

gino CMS
2.4.0

Generated by Doxygen 1.8.8

Thu Nov 12 2015 12:33:31

Contents

1	gino CMS	1
2	Todo List	3
3	Namespace Index	5
3.1	Namespace List	5
4	Hierarchical Index	7
4.1	Class Hierarchy	7
5	Data Structure Index	11
5.1	Data Structures	11
6	File Index	17
6.1	File List	17
7	Namespace Documentation	23
7.1	Gino Namespace Reference	23
7.1.1	Detailed Description	30
7.1.2	Function Documentation	31
7.1.2.1	absolutePath	31
7.1.2.2	arrayToObject	31
7.1.2.3	attributeVar	31
7.1.2.4	baseFileName	31
7.1.2.5	browsers	31
7.1.2.6	checkEmail	32
7.1.2.7	clean_sequence	32
7.1.2.8	cleanVar	32
7.1.2.9	cleanVarEditor	33
7.1.2.10	codeDb	33
7.1.2.11	codeInput	33
7.1.2.12	codeParser	33
7.1.2.13	codeToDB	34
7.1.2.14	convertToDatabase	34

7.1.2.15	convertToHtml	34
7.1.2.16	cryptMethod	35
7.1.2.17	cutHtmlText	35
7.1.2.18	cutString	35
7.1.2.19	dateDiff	36
7.1.2.20	dateToDbDate	37
7.1.2.21	dbDatetimeToDate	37
7.1.2.22	dbDatetimeToTime	37
7.1.2.23	dbDateToDate	37
7.1.2.24	dbNumberToNumber	38
7.1.2.25	dbTimeToTime	38
7.1.2.26	decode_params	38
7.1.2.27	deleteFileDir	38
7.1.2.28	download	39
7.1.2.29	emailPolicy	39
7.1.2.30	enabledPng	39
7.1.2.31	enabledZip	39
7.1.2.32	enclosedField	39
7.1.2.33	encode_params	40
7.1.2.34	extension	40
7.1.2.35	extensionFile	40
7.1.2.36	get_browser_info	40
7.1.2.37	getDateDiff	41
7.1.2.38	gOpt	41
7.1.2.39	htmlChars	41
7.1.2.40	htmlCharsText	42
7.1.2.41	htmlInput	42
7.1.2.42	htmlInputEditor	42
7.1.2.43	htmlToEntities	42
7.1.2.44	icon	43
7.1.2.45	is_browser	45
7.1.2.46	is_browser_platform	45
7.1.2.47	is_browser_version	45
7.1.2.48	is_robot	45
7.1.2.49	isNumeric	45
7.1.2.50	isValid	46
7.1.2.51	jsVar	46
7.1.2.52	listProv	46
7.1.2.53	numberToDB	46
7.1.2.54	obj_serialize	47

7.1.2.55	obj_unserialize	47
7.1.2.56	platforms	47
7.1.2.57	preCodeParser	47
7.1.2.58	relativePath	48
7.1.2.59	remove_attributes	48
7.1.2.60	replaceChar	48
7.1.2.61	robots	48
7.1.2.62	searchNameFile	48
7.1.2.63	share	49
7.1.2.64	shareAll	49
7.1.2.65	slimboxReplace	49
7.1.2.66	strip_invisible_tags	49
7.1.2.67	strip_selected_tags	50
7.1.2.68	strip_tags_attributes	50
7.1.2.69	stripAll	50
7.1.2.70	stripEditor	51
7.1.2.71	timeDiff	52
7.1.2.72	timeToDbTime	52
7.1.2.73	traslitterazione	52
7.1.2.74	utf8_urldecode	52
7.1.2.75	verifyExtension	53
7.1.3	Variable Documentation	53
7.1.3.1	APP_DIR	53
7.1.3.2	CACHE_DIR	53
7.1.3.3	CLASSES_DIR	53
7.1.3.4	CONTENT_DIR	53
7.1.3.5	CONTENT_WWW	53
7.1.3.6	CONTROLLER_CLASS_PREFIX	53
7.1.3.7	CORE_CLASS_PREFIX	53
7.1.3.8	CSS_DIR	54
7.1.3.9	CSS_WWW	54
7.1.3.10	FIELDS_DIR	54
7.1.3.11	FONTS_DIR	54
7.1.3.12	GIMAGE_DIR	54
7.1.3.13	GRAPHICS_DIR	54
7.1.3.14	HOME_FILE	54
7.1.3.15	LIB_DIR	54
7.1.3.16	MODEL_CLASS_PREFIX	54
7.1.3.17	OS	54
7.1.3.18	PLUGIN_DIR	54

7.1.3.19	SITE_APP	54
7.1.3.20	SITE_CUSTOM_CKEDITOR	54
7.1.3.21	SITE_GRAPHICS	55
7.1.3.22	SITE_IMG	55
7.1.3.23	SITE_JS	55
7.1.3.24	SITE_LIB	55
7.1.3.25	SITE_ROOT	55
7.1.3.26	SITE_WWW	55
7.1.3.27	TBL_GROUP	55
7.1.3.28	TBL_GROUP_PERMISSION	55
7.1.3.29	TBL_LANGUAGE	55
7.1.3.30	TBL_LOG_ACCESS	55
7.1.3.31	TBL_MODULE	55
7.1.3.32	TBL_MODULE_APP	55
7.1.3.33	TBL_NATION	56
7.1.3.34	TBL_PERMISSION	56
7.1.3.35	TBL_SYS_CONF	56
7.1.3.36	TBL_TRANSLATION	56
7.1.3.37	TBL_USER	56
7.1.3.38	TBL_USER_ADD	56
7.1.3.39	TBL_USER_GROUP	56
7.1.3.40	TBL_USER_PERMISSION	56
7.1.3.41	TMP_DIR	56
7.1.3.42	TPL_DIR	56
7.1.3.43	VIEWS_DIR	56
7.2	Gino\App Namespace Reference	56
7.3	Gino\App\Attachment Namespace Reference	57
7.3.1	Detailed Description	58
7.4	Gino\App\Auth Namespace Reference	58
7.4.1	Detailed Description	58
7.4.2	Variable Documentation	58
7.4.2.1	LDAP_APP_PASSWORD	58
7.4.2.2	LDAP_APP_USERNAME	58
7.4.2.3	LDAP_BASE_DN	58
7.4.2.4	LDAP_DOMAIN	58
7.4.2.5	LDAP_FILTER_SEARCH	58
7.4.2.6	LDAP_HOST	59
7.4.2.7	LDAP_PORT	59
7.4.2.8	LDAP_PROTOCOL_VERSION	59
7.4.2.9	LDAP_SEARCH_DN	59

7.5	Gino\App\Graphics Namespace Reference	59
7.5.1	Detailed Description	59
7.6	Gino\App\Index Namespace Reference	59
7.6.1	Detailed Description	59
7.7	Gino\App\Instruments Namespace Reference	59
7.7.1	Detailed Description	59
7.8	Gino\App\Language Namespace Reference	60
7.8.1	Detailed Description	60
7.9	Gino\App\Layout Namespace Reference	60
7.9.1	Detailed Description	60
7.10	Gino\App\Menu Namespace Reference	60
7.10.1	Detailed Description	60
7.11	Gino\App\Module Namespace Reference	60
7.11.1	Detailed Description	61
7.12	Gino\App\Page Namespace Reference	61
7.12.1	Detailed Description	61
7.13	Gino\App\PhpModuleView Namespace Reference	61
7.13.1	Detailed Description	61
7.14	Gino\App\SearchSite Namespace Reference	62
7.14.1	Detailed Description	62
7.15	Gino\App\Statistics Namespace Reference	62
7.15.1	Detailed Description	62
7.16	Gino\App\SysClass Namespace Reference	62
7.16.1	Detailed Description	62
7.17	Gino\App\Sysconf Namespace Reference	63
7.17.1	Detailed Description	63
7.18	Gino\App\Sysfunc Namespace Reference	63
7.18.1	Detailed Description	63
7.19	Gino\Exception Namespace Reference	63
7.19.1	Detailed Description	63
7.20	Gino\Http Namespace Reference	64
7.20.1	Detailed Description	64
7.21	Gino\Plugin Namespace Reference	64
7.21.1	Detailed Description	64
7.22	Gino\Test Namespace Reference	64
7.22.1	Detailed Description	65
7.22.2	Function Documentation	65
7.22.2.1	setPage	65

8.1	Access Class Reference	67
8.1.1	Detailed Description	68
8.1.2	Constructor & Destructor Documentation	68
8.1.2.1	__construct	68
8.1.3	Member Function Documentation	68
8.1.3.1	Authentication	68
8.1.3.2	AuthenticationMethod	69
8.1.3.3	logAccess	69
8.1.3.4	loginError	69
8.1.3.5	loginSuccess	69
8.1.3.6	requireAdmin	70
8.1.3.7	requirePerm	70
8.1.4	Field Documentation	70
8.1.4.1	\$_db	70
8.1.4.2	\$_home	70
8.1.4.3	\$_session	70
8.2	AdminTable Class Reference	70
8.2.1	Detailed Description	73
8.2.2	Constructor & Destructor Documentation	74
8.2.2.1	__construct	74
8.2.3	Member Function Documentation	75
8.2.3.1	addWhereClauses	75
8.2.3.2	addWhereExtra	75
8.2.3.3	addWhereJoin	75
8.2.3.4	adminDelete	75
8.2.3.5	adminForm	76
8.2.3.6	adminList	76
8.2.3.7	backOffice	78
8.2.3.8	clean	78
8.2.3.9	dump	78
8.2.3.10	editUrl	79
8.2.3.11	formFilters	79
8.2.3.12	formFiltersAdd	79
8.2.3.13	hidden	80
8.2.3.14	m2mthroughAction	80
8.2.3.15	modelAction	80
8.2.3.16	modelForm	82
8.2.3.17	permission	84
8.2.3.18	readFile	84
8.2.3.19	restore	85

8.2.3.20	setSessionSearch	85
8.2.3.21	setSessionSearchAdd	85
8.2.4	Field Documentation	85
8.2.4.1	\$_allow_insertion	85
8.2.4.2	\$_controller	85
8.2.4.3	\$_db	85
8.2.4.4	\$_delete_deny	85
8.2.4.5	\$_edit_deny	86
8.2.4.6	\$_filter_add	86
8.2.4.7	\$_filter_fields	86
8.2.4.8	\$_filter_join	86
8.2.4.9	\$_form	86
8.2.4.10	\$_hidden	86
8.2.4.11	\$_ifp	86
8.2.4.12	\$_list_display	86
8.2.4.13	\$_list_remove	86
8.2.4.14	\$_registry	86
8.2.4.15	\$_request	86
8.2.4.16	\$_session	86
8.2.4.17	\$_view	86
8.3	AdminTable_AuthUser Class Reference	86
8.3.1	Detailed Description	87
8.3.2	Member Function Documentation	87
8.3.2.1	modelAction	87
8.4	attachment Class Reference	89
8.4.1	Detailed Description	90
8.4.2	Constructor & Destructor Documentation	90
8.4.2.1	__construct	90
8.4.3	Member Function Documentation	90
8.4.3.1	downloader	90
8.4.3.2	editorAttachmentList	90
8.4.3.3	editorList	91
8.4.3.4	getDataDir	91
8.4.3.5	getDataWWW	91
8.4.3.6	jsonImageList	91
8.4.3.7	manageAttachment	91
8.4.3.8	manageCategory	92
8.4.3.9	manageItem	92
8.4.4	Field Documentation	92
8.4.4.1	\$_title	92

8.5	AttachmentCtg Class Reference	92
8.5.1	Detailed Description	94
8.5.2	Constructor & Destructor Documentation	94
8.5.2.1	__construct	94
8.5.3	Member Function Documentation	94
8.5.3.1	__toString	94
8.5.3.2	delete	94
8.5.3.3	getForSelect	94
8.5.3.4	path	95
8.5.3.5	structure	95
8.5.4	Field Documentation	95
8.5.4.1	\$table	95
8.6	AttachmentItem Class Reference	95
8.6.1	Detailed Description	97
8.6.2	Constructor & Destructor Documentation	97
8.6.2.1	__construct	97
8.6.3	Member Function Documentation	98
8.6.3.1	__toString	98
8.6.3.2	deleteFromCtg	98
8.6.3.3	path	98
8.6.3.4	pathDownload	98
8.6.3.5	pathView	98
8.6.3.6	previewAdminList	99
8.6.3.7	previewLink	99
8.6.3.8	structure	99
8.6.3.9	type	99
8.6.4	Field Documentation	99
8.6.4.1	\$_doc_extension	99
8.6.4.2	\$_img_extension	99
8.6.4.3	\$_pdf_extension	100
8.6.4.4	\$_xls_extension	100
8.6.4.5	\$table	100
8.7	AttachmentItemAdminTable Class Reference	100
8.7.1	Detailed Description	101
8.7.2	Member Function Documentation	101
8.7.2.1	modelAction	101
8.8	auth Class Reference	101
8.8.1	Detailed Description	104
8.8.2	Constructor & Destructor Documentation	104
8.8.2.1	__construct	104

8.8.3	Member Function Documentation	104
8.8.3.1	actionJoinGroupPermission	104
8.8.3.2	actionJoinUserPermission	105
8.8.3.3	changePassword	105
8.8.3.4	checkAuthenticationUser	105
8.8.3.5	checkUsername	106
8.8.3.6	formGroup	106
8.8.3.7	formPermission	106
8.8.3.8	getAddPath	106
8.8.3.9	getBasePath	107
8.8.3.10	getCheckUser	107
8.8.3.11	getClassElements	107
8.8.3.12	joiningrouppermission	108
8.8.3.13	joinUserPermission	108
8.8.3.14	login	108
8.8.3.15	manageAuth	108
8.8.3.16	manageGroup	108
8.8.3.17	managePermission	109
8.8.3.18	manageUser	109
8.8.3.19	outputFunctions	109
8.8.3.20	passwordRules	109
8.8.3.21	returnJoinLink	109
8.8.4	Field Documentation	110
8.8.4.1	\$_aut_pwd	110
8.8.4.2	\$_aut_pwd_length	110
8.8.4.3	\$_label_field1	110
8.8.4.4	\$_label_field2	110
8.8.4.5	\$_label_field3	110
8.8.4.6	\$_ldap_auth	110
8.8.4.7	\$_ldap_auth_only	110
8.8.4.8	\$_ldap_auth_password	110
8.8.4.9	\$_ldap_single_user	110
8.8.4.10	\$_options	110
8.8.4.11	\$_optionsLabels	110
8.8.4.12	\$_other_field1	110
8.8.4.13	\$_other_field2	110
8.8.4.14	\$_other_field3	110
8.8.4.15	\$_pwd_length_max	110
8.8.4.16	\$_pwd_length_min	110
8.8.4.17	\$_pwd_numeric_number	110

8.8.4.18	<code>\$_self_registration</code>	110
8.8.4.19	<code>\$_self_registration_active</code>	110
8.8.4.20	<code>\$_title</code>	110
8.8.4.21	<code>\$_user_more</code>	110
8.8.4.22	<code>\$_user_view</code>	110
8.8.4.23	<code>\$_username_as_email</code>	110
8.8.4.24	<code>\$_users_for_page</code>	110
8.9	BooleanField Class Reference	111
8.9.1	Detailed Description	112
8.9.2	Constructor & Destructor Documentation	112
8.9.2.1	<code>__construct</code>	112
8.9.3	Member Function Documentation	112
8.9.3.1	<code>__toString</code>	112
8.9.3.2	<code>formElement</code>	113
8.9.3.3	<code>getDefault</code>	113
8.9.3.4	<code>getEnum</code>	113
8.9.3.5	<code>setDefault</code>	113
8.9.3.6	<code>setEnum</code>	113
8.9.4	Field Documentation	114
8.9.4.1	<code>\$_default</code>	114
8.9.4.2	<code>\$_enum</code>	114
8.10	Cache Class Reference	114
8.10.1	Detailed Description	115
8.10.2	Constructor & Destructor Documentation	115
8.10.2.1	<code>__construct</code>	115
8.10.3	Member Function Documentation	115
8.10.3.1	<code>delete</code>	115
8.10.3.2	<code>getFilename</code>	116
8.10.3.3	<code>isCached</code>	116
8.10.3.4	<code>read</code>	116
8.10.3.5	<code>write</code>	116
8.10.4	Field Documentation	117
8.10.4.1	<code>\$_ds</code>	117
8.10.4.2	<code>\$_enabled</code>	117
8.10.4.3	<code>\$_fld</code>	117
8.10.4.4	<code>\$_grp</code>	117
8.10.4.5	<code>\$_prefix</code>	117
8.10.4.6	<code>\$_registry</code>	117
8.10.4.7	<code>\$_tc</code>	117
8.10.4.8	<code>\$id</code>	117

8.11	Captcha Class Reference	117
8.11.1	Detailed Description	118
8.11.2	Constructor & Destructor Documentation	118
8.11.2.1	__construct	118
8.11.3	Member Function Documentation	118
8.11.3.1	check	118
8.11.3.2	checkRequirements	119
8.11.3.3	generateStrings	119
8.11.3.4	hex2RGB	119
8.11.3.5	render	119
8.11.4	Field Documentation	119
8.11.4.1	\$_allow_numbers	119
8.11.4.2	\$_font_file	120
8.11.4.3	\$_height	120
8.11.4.4	\$_letters	120
8.11.4.5	\$_name	120
8.11.4.6	\$_numbers	120
8.11.4.7	\$_width	120
8.12	CharField Class Reference	120
8.12.1	Detailed Description	121
8.12.2	Constructor & Destructor Documentation	122
8.12.2.1	__construct	122
8.12.3	Member Function Documentation	122
8.12.3.1	filterWhereClause	122
8.12.3.2	formElement	122
8.12.3.3	getTrnsl	122
8.12.3.4	setTrnsl	123
8.12.4	Field Documentation	124
8.12.4.1	\$_trnsl	124
8.13	Compressor Class Reference	124
8.13.1	Detailed Description	125
8.13.2	Constructor & Destructor Documentation	125
8.13.2.1	__construct	125
8.13.3	Member Function Documentation	125
8.13.3.1	addCss	125
8.13.3.2	addJs	125
8.13.3.3	convertRelativePath	125
8.13.3.4	filePath	126
8.13.3.5	mergeCss	126
8.13.3.6	mergeJs	126

8.13.3.7	minifyCss	126
8.13.3.8	minifyJs	127
8.13.3.9	moveCss	127
8.13.3.10	shouldUpdate	127
8.13.4	Field Documentation	127
8.13.4.1	\$_css	127
8.13.4.2	\$_js	127
8.14	Conf Class Reference	128
8.14.1	Detailed Description	129
8.14.2	Constructor & Destructor Documentation	129
8.14.2.1	__construct	129
8.14.3	Member Function Documentation	129
8.14.3.1	__toString	129
8.14.3.2	structure	129
8.14.4	Field Documentation	129
8.14.4.1	\$table	129
8.15	Controller Class Reference	130
8.15.1	Detailed Description	132
8.15.2	Constructor & Destructor Documentation	132
8.15.2.1	__construct	132
8.15.3	Member Function Documentation	132
8.15.3.1	__call	132
8.15.3.2	deleteInstance	133
8.15.3.3	getBaseAbsPath	133
8.15.3.4	getBasePath	133
8.15.3.5	getClassElements	133
8.15.3.6	getClassName	133
8.15.3.7	getInstance	134
8.15.3.8	getInstanceName	134
8.15.3.9	link	134
8.15.3.10	linkAdmin	134
8.15.3.11	manageFrontend	134
8.15.3.12	manageOptions	134
8.15.3.13	requirePerm	135
8.15.3.14	setInstanceProperties	135
8.15.3.15	setOption	135
8.15.3.16	setPaths	135
8.15.3.17	userHasPerm	135
8.15.4	Field Documentation	136
8.15.4.1	\$_access	136

8.15.4.2	\$_class_img	136
8.15.4.3	\$_class_name	136
8.15.4.4	\$_class_www	136
8.15.4.5	\$_data_dir	136
8.15.4.6	\$_data_www	136
8.15.4.7	\$_db	136
8.15.4.8	\$_home	136
8.15.4.9	\$_instance	136
8.15.4.10	\$_instance_label	136
8.15.4.11	\$_instance_name	136
8.15.4.12	\$_locale	136
8.15.4.13	\$_permissions	136
8.15.4.14	\$_plink	136
8.15.4.15	\$_registry	136
8.15.4.16	\$_session	136
8.15.4.17	\$_trd	136
8.15.4.18	\$_view_dir	136
8.16	Core Class Reference	136
8.16.1	Detailed Description	137
8.16.2	Constructor & Destructor Documentation	137
8.16.2.1	__construct	137
8.16.3	Member Function Documentation	138
8.16.3.1	answer	138
8.16.3.2	detectMobile	138
8.16.3.3	initMobile	138
8.16.3.4	initRegistry	138
8.16.4	Field Documentation	138
8.16.4.1	\$_db	138
8.16.4.2	\$_registry	138
8.17	Css Class Reference	139
8.17.1	Detailed Description	140
8.17.2	Constructor & Destructor Documentation	141
8.17.2.1	__construct	141
8.17.3	Member Function Documentation	141
8.17.3.1	actionCssLayout	141
8.17.3.2	actionDelCssLayout	141
8.17.3.3	cssFileName	142
8.17.3.4	formCssLayout	142
8.17.3.5	formDelCssLayout	142
8.17.3.6	getAll	142

8.17.3.7	getFromFilename	142
8.17.3.8	layoutInfo	142
8.17.3.9	setFilename	143
8.17.4	Field Documentation	144
8.17.4.1	\$_class	144
8.17.4.2	\$_css_list	144
8.17.4.3	\$_instance_class	144
8.17.4.4	\$_interface	144
8.17.4.5	\$_label	144
8.17.4.6	\$_mdlLink	144
8.17.4.7	\$_module	144
8.17.4.8	\$_name	144
8.17.4.9	\$_tbl_data	144
8.17.4.10	\$_tbl_module	144
8.17.4.11	\$table	144
8.18	DataCache Class Reference	144
8.18.1	Detailed Description	145
8.18.2	Constructor & Destructor Documentation	145
8.18.2.1	__construct	145
8.18.3	Member Function Documentation	145
8.18.3.1	get	145
8.18.3.2	save	146
8.19	DateField Class Reference	146
8.19.1	Detailed Description	147
8.19.2	Constructor & Destructor Documentation	147
8.19.2.1	__construct	147
8.19.3	Member Function Documentation	148
8.19.3.1	__toString	148
8.19.3.2	clean	148
8.19.3.3	filterWhereClause	148
8.19.3.4	formElement	148
8.20	DatetimeField Class Reference	149
8.20.1	Detailed Description	150
8.20.2	Constructor & Destructor Documentation	150
8.20.2.1	__construct	150
8.20.3	Member Function Documentation	151
8.20.3.1	__toString	151
8.20.3.2	clean	151
8.20.3.3	filterWhereClause	151
8.20.3.4	formElement	151

8.20.3.5	getAutoNow	152
8.20.3.6	getAutoNowAdd	152
8.20.3.7	setAutoNow	152
8.20.3.8	setAutoNowAdd	152
8.20.4	Field Documentation	152
8.20.4.1	\$_auto_now	152
8.20.4.2	\$_auto_now_add	152
8.21	Db Class Reference	153
8.21.1	Detailed Description	154
8.21.2	Member Function Documentation	154
8.21.2.1	instance	154
8.21.3	Field Documentation	154
8.21.3.1	\$_db_charset	154
8.21.3.2	\$_db_dbname	154
8.21.3.3	\$_db_host	154
8.21.3.4	\$_db_pass	154
8.21.3.5	\$_db_schema	154
8.21.3.6	\$_db_user	154
8.22	DbManager Interface Reference	154
8.22.1	Detailed Description	157
8.22.2	Constructor & Destructor Documentation	157
8.22.2.1	__construct	157
8.22.3	Member Function Documentation	157
8.22.3.1	actionquery	157
8.22.3.2	affected	157
8.22.3.3	autoIncValue	157
8.22.3.4	begin	158
8.22.3.5	closeConnection	158
8.22.3.6	columnHasValue	158
8.22.3.7	commit	158
8.22.3.8	concat	159
8.22.3.9	conformType	159
8.22.3.10	delete	159
8.22.3.11	distinct	160
8.22.3.12	drop	160
8.22.3.13	dump	160
8.22.3.14	dumpDatabase	161
8.22.3.15	escapeString	161
8.22.3.16	fieldInformations	161
8.22.3.17	freeresult	161

8.22.3.18 getFieldFromId	162
8.22.3.19 getFieldsName	162
8.22.3.20 getInfoQuery	162
8.22.3.21 getlastid	162
8.22.3.22 getNumRecords	163
8.22.3.23 getTableStructure	163
8.22.3.24 insert	164
8.22.3.25 join	165
8.22.3.26 limit	165
8.22.3.27 multiActionquery	165
8.22.3.28 openConnection	166
8.22.3.29 query	166
8.22.3.30 restore	166
8.22.3.31 resultselect	167
8.22.3.32 rollback	167
8.22.3.33 select	167
8.22.3.34 selectquery	168
8.22.3.35 tableexists	168
8.22.3.36 union	168
8.22.3.37 update	169
8.23 DirectoryField Class Reference	169
8.23.1 Detailed Description	171
8.23.2 Constructor & Destructor Documentation	171
8.23.2.1 __construct	171
8.23.3 Member Function Documentation	172
8.23.3.1 checkName	172
8.23.3.2 clean	172
8.23.3.3 defaultName	172
8.23.3.4 delete	173
8.23.3.5 getPath	173
8.23.3.6 getPrefix	173
8.23.3.7 setPath	173
8.23.3.8 setPrefix	173
8.23.3.9 validate	173
8.23.4 Field Documentation	174
8.23.4.1 \$_default_name	174
8.23.4.2 \$_path	174
8.23.4.3 \$_prefix	174
8.24 Document Class Reference	174
8.24.1 Detailed Description	175

8.24.2	Constructor & Destructor Documentation	175
8.24.2.1	__construct	175
8.24.3	Member Function Documentation	176
8.24.3.1	__invoke	176
8.24.3.2	checkOutputFunctionPermissions	176
8.24.3.3	errorMessages	176
8.24.3.4	footLine	176
8.24.3.5	getSkin	176
8.24.3.6	google_analytics	176
8.24.3.7	headLine	177
8.24.3.8	modClass	177
8.24.3.9	modPage	177
8.24.3.10	modUrl	177
8.24.3.11	parseModules	177
8.24.3.12	render	178
8.24.3.13	renderModule	178
8.24.3.14	renderNave	178
8.24.3.15	setHeadVariables	179
8.24.3.16	setIfEmpty	179
8.24.4	Field Documentation	179
8.24.4.1	\$_registry	179
8.24.4.2	\$_request	179
8.24.4.3	\$_url_content	179
8.25	EmailField Class Reference	179
8.25.1	Detailed Description	180
8.25.2	Constructor & Destructor Documentation	181
8.25.2.1	__construct	181
8.25.3	Member Function Documentation	181
8.25.3.1	clean	181
8.25.3.2	validate	181
8.26	EnumField Class Reference	182
8.26.1	Detailed Description	183
8.26.2	Constructor & Destructor Documentation	183
8.26.2.1	__construct	183
8.26.3	Member Function Documentation	183
8.26.3.1	__toString	183
8.26.3.2	formElement	184
8.26.3.3	getDefault	184
8.26.3.4	getEnum	184
8.26.3.5	setDefault	184

8.26.3.6	setEnum	184
8.26.4	Field Documentation	185
8.26.4.1	\$_default	185
8.26.4.2	\$_enum	185
8.27	Error Class Reference	185
8.27.1	Detailed Description	185
8.27.2	Member Function Documentation	186
8.27.2.1	codeMessages	186
8.27.2.2	errorMessage	186
8.27.2.3	getErrorMessage	186
8.27.2.4	warning	186
8.28	EventDispatcher Class Reference	187
8.28.1	Detailed Description	188
8.28.2	Member Function Documentation	188
8.28.2.1	emit	188
8.28.2.2	listen	188
8.28.2.3	listenEmitter	188
8.28.3	Field Documentation	189
8.28.3.1	\$_emitter_listeners	189
8.28.3.2	\$_listeners	189
8.29	Exception403 Class Reference	189
8.29.1	Detailed Description	190
8.29.2	Constructor & Destructor Documentation	190
8.29.2.1	__construct	190
8.29.3	Member Function Documentation	191
8.29.3.1	httpResponse	191
8.29.4	Field Documentation	191
8.29.4.1	\$_redirect	191
8.30	Exception404 Class Reference	191
8.30.1	Detailed Description	192
8.30.2	Constructor & Destructor Documentation	192
8.30.2.1	__construct	192
8.30.3	Member Function Documentation	193
8.30.3.1	httpResponse	193
8.31	Exception500 Class Reference	193
8.31.1	Detailed Description	194
8.31.2	Constructor & Destructor Documentation	194
8.31.2.1	__construct	194
8.31.3	Member Function Documentation	194
8.31.3.1	httpResponse	194

8.32 Export Class Reference	194
8.32.1 Detailed Description	196
8.32.2 Member Function Documentation	196
8.32.2.1 exportCsv	196
8.32.2.2 exportData	196
8.32.2.3 getData	197
8.32.2.4 getHeadFields	197
8.32.2.5 removeBOM	197
8.32.2.6 setData	197
8.32.2.7 setFields	197
8.32.2.8 setHead	198
8.32.2.9 setOrder	198
8.32.2.10 setRids	198
8.32.2.11 setSeparator	199
8.32.2.12 setTable	200
8.32.2.13 writeFile	200
8.32.3 Field Documentation	201
8.32.3.1 \$_data	201
8.32.3.2 \$_fields	201
8.32.3.3 \$_head	201
8.32.3.4 \$_order	201
8.32.3.5 \$_rids	201
8.32.3.6 \$_s	201
8.32.3.7 \$_table	201
8.33 Field Class Reference	201
8.33.1 Detailed Description	204
8.33.2 Constructor & Destructor Documentation	204
8.33.2.1 __construct	204
8.33.3 Member Function Documentation	205
8.33.3.1 __toString	205
8.33.3.2 adminListOrder	205
8.33.3.3 canBeOrdered	205
8.33.3.4 clean	206
8.33.3.5 filterWhereClause	206
8.33.3.6 formElement	206
8.33.3.7 formElementWidget	207
8.33.3.8 getAutoIncrement	208
8.33.3.9 getDefault	208
8.33.3.10 getLabel	209
8.33.3.11 getLenght	209

8.33.3.12	getName	209
8.33.3.13	getPrimaryKey	209
8.33.3.14	getRequired	209
8.33.3.15	getUniqueKey	209
8.33.3.16	getValue	209
8.33.3.17	getValueType	210
8.33.3.18	getWidget	210
8.33.3.19	setAutoIncrement	210
8.33.3.20	setDefault	210
8.33.3.21	setLabel	210
8.33.3.22	setLenght	210
8.33.3.23	setName	211
8.33.3.24	setPrimaryKey	211
8.33.3.25	setRequired	211
8.33.3.26	setUniqueKey	211
8.33.3.27	setValue	212
8.33.3.28	setValueType	213
8.33.3.29	setWidget	213
8.33.3.30	validate	213
8.33.4	Field Documentation	213
8.33.4.1	\$_auto_increment	213
8.33.4.2	\$_default	213
8.33.4.3	\$_default_widget	213
8.33.4.4	\$_label	213
8.33.4.5	\$_lenght	214
8.33.4.6	\$_model	214
8.33.4.7	\$_name	214
8.33.4.8	\$_primary_key	214
8.33.4.9	\$_required	214
8.33.4.10	\$_table	214
8.33.4.11	\$_unique_key	214
8.33.4.12	\$_value	214
8.33.4.13	\$_value_type	214
8.34	FileField Class Reference	214
8.34.1	Detailed Description	216
8.34.2	Constructor & Destructor Documentation	217
8.34.2.1	__construct	217
8.34.3	Member Function Documentation	217
8.34.3.1	checkFilename	217
8.34.3.2	clean	218

8.34.3.3	conformPath	218
8.34.3.4	delete	218
8.34.3.5	formElement	218
8.34.3.6	getAddPath	218
8.34.3.7	getCheckType	219
8.34.3.8	getDirectory	219
8.34.3.9	getExtensions	219
8.34.3.10	getMaxFileSize	219
8.34.3.11	getPath	219
8.34.3.12	getPrefix	219
8.34.3.13	getTypesAllowed	219
8.34.3.14	pathToFile	220
8.34.3.15	saveFile	221
8.34.3.16	setAddPath	221
8.34.3.17	setCheckType	221
8.34.3.18	setDirectory	221
8.34.3.19	setExtensions	222
8.34.3.20	setMaxFileSize	222
8.34.3.21	setPath	222
8.34.3.22	setPrefix	222
8.34.3.23	setTypesAllowed	223
8.34.3.24	validate	224
8.34.4	Field Documentation	224
8.34.4.1	\$_check_type	224
8.34.4.2	\$_delete_file	224
8.34.4.3	\$_directory	224
8.34.4.4	\$_extensions	224
8.34.4.5	\$_max_file_size	224
8.34.4.6	\$_path_abs	224
8.34.4.7	\$_path_add	224
8.34.4.8	\$_prefix	224
8.34.4.9	\$_types_allowed	224
8.35	FloatField Class Reference	224
8.35.1	Detailed Description	226
8.35.2	Constructor & Destructor Documentation	226
8.35.2.1	__construct	226
8.35.3	Member Function Documentation	226
8.35.3.1	__toString	226
8.35.3.2	clean	227
8.35.3.3	formElement	227

8.35.3.4	getDecimalDigits	227
8.35.3.5	getIntDigits	227
8.35.3.6	setDecimalDigits	227
8.35.3.7	setIntDigits	227
8.35.4	Field Documentation	228
8.35.4.1	\$_decimal_digits	228
8.35.4.2	\$_int_digits	228
8.36	ForeignKeyField Class Reference	228
8.36.1	Detailed Description	229
8.36.2	Constructor & Destructor Documentation	229
8.36.2.1	__construct	229
8.36.3	Member Function Documentation	230
8.36.3.1	__toString	230
8.36.3.2	formElement	230
8.36.3.3	getEnum	230
8.36.4	Field Documentation	231
8.36.4.1	\$_add_related	231
8.36.4.2	\$_add_related_url	231
8.36.4.3	\$_enum	231
8.36.4.4	\$_foreign	231
8.36.4.5	\$_foreign_order	231
8.36.4.6	\$_foreign_where	231
8.37	Form Class Reference	231
8.37.1	Detailed Description	234
8.37.2	Constructor & Destructor Documentation	234
8.37.2.1	__construct	234
8.37.3	Member Function Documentation	235
8.37.3.1	arequired	235
8.37.3.2	captcha	235
8.37.3.3	checkbox	235
8.37.3.4	cfile	236
8.37.3.5	checkbox	236
8.37.3.6	checkCaptcha	237
8.37.3.7	checkDefaultCaptcha	237
8.37.3.8	checkFilename	237
8.37.3.9	checkReCaptcha	238
8.37.3.10	cinput	238
8.37.3.11	cinput_date	239
8.37.3.12	close	239
8.37.3.13	countEqualName	239

8.37.3.14	cradio	239
8.37.3.15	createImage	240
8.37.3.16	cselect	240
8.37.3.17	ctextarea	241
8.37.3.18	defaultCaptcha	241
8.37.3.19	dimensionFile	242
8.37.3.20	dirUpload	242
8.37.3.21	editorHtml	242
8.37.3.22	fcktextarea	243
8.37.3.23	formFieldTranslation	243
8.37.3.24	generateFormToken	244
8.37.3.25	hidden	244
8.37.3.26	imagePreviewer	244
8.37.3.27	input	244
8.37.3.28	isImage	245
8.37.3.29	jsCountCharText	245
8.37.3.30	label	245
8.37.3.31	load	246
8.37.3.32	manageFile	246
8.37.3.33	multipleCheckbox	247
8.37.3.34	noinput	248
8.37.3.35	open	249
8.37.3.36	option	249
8.37.3.37	radio	250
8.37.3.38	reCaptcha	251
8.37.3.39	resizeImage	251
8.37.3.40	retvar	251
8.37.3.41	save	253
8.37.3.42	savelImage	253
8.37.3.43	select	253
8.37.3.44	setMethod	254
8.37.3.45	setOptions	254
8.37.3.46	setValidation	254
8.37.3.47	textarea	255
8.37.3.48	upload	255
8.37.3.49	verifyFormToken	255
8.37.4	Field Documentation	256
8.37.4.1	\$_div_field_width	256
8.37.4.2	\$_div_label_width	256
8.37.4.3	\$_extension_denied	256

8.37.4.4	<code>\$_fckeditor_field</code>	256
8.37.4.5	<code>\$_form_name</code>	256
8.37.4.6	<code>\$_ico_calendar_path</code>	256
8.37.4.7	<code>\$_input_field</code>	256
8.37.4.8	<code>\$_input_max</code>	256
8.37.4.9	<code>\$_input_size</code>	256
8.37.4.10	<code>\$_lng_trl</code>	256
8.37.4.11	<code>\$_max_file_size</code>	256
8.37.4.12	<code>\$_method</code>	256
8.37.4.13	<code>\$_multi_language</code>	256
8.37.4.14	<code>\$_options</code>	256
8.37.4.15	<code>\$_prefix_file</code>	256
8.37.4.16	<code>\$_prefix_thumb</code>	256
8.37.4.17	<code>\$_registry</code>	256
8.37.4.18	<code>\$_request</code>	256
8.37.4.19	<code>\$_requestVar</code>	256
8.37.4.20	<code>\$_tbl_attachment</code>	256
8.37.4.21	<code>\$_tbl_attachment_ctg</code>	256
8.37.4.22	<code>\$_textarea_field</code>	256
8.37.4.23	<code>\$_trd</code>	256
8.37.4.24	<code>\$_trsl_id</code>	257
8.37.4.25	<code>\$_trsl_table</code>	257
8.37.4.26	<code>\$_validation</code>	257
8.37.4.27	<code>\$session</code>	257
8.37.4.28	<code>_IMAGE_GIF_</code>	257
8.37.4.29	<code>_IMAGE_JPG_</code>	257
8.37.4.30	<code>_IMAGE_PNG_</code>	257
8.38	Frontend Class Reference	257
8.38.1	Detailed Description	258
8.38.2	Constructor & Destructor Documentation	258
8.38.2.1	<code>__construct</code>	258
8.38.3	Member Function Documentation	258
8.38.3.1	<code>actionModuleFile</code>	258
8.38.3.2	<code>fileName</code>	259
8.38.3.3	<code>formModuleFile</code>	259
8.38.3.4	<code>manageFrontend</code>	259
8.38.3.5	<code>moduleList</code>	259
8.38.3.6	<code>pathToFile</code>	260
8.38.3.7	<code>setLists</code>	261
8.38.4	Field Documentation	261

8.38.4.1	\$_class	261
8.38.4.2	\$_css_list	261
8.38.4.3	\$_mdlLink	261
8.38.4.4	\$_module	261
8.38.4.5	\$_module_id	261
8.38.4.6	\$_registry	261
8.38.4.7	\$_view_list	261
8.39	FuncTest Class Reference	261
8.39.1	Detailed Description	262
8.39.2	Member Function Documentation	263
8.39.2.1	test_absolutePath	263
8.39.2.2	test_arrayToObject	263
8.39.2.3	test_baseFilename	263
8.39.2.4	test_checkEmail	263
8.39.2.5	test_cutHtmlText	263
8.39.2.6	test_cutString	263
8.39.2.7	test_dateDiff	263
8.39.2.8	test_dateToDbDate	263
8.39.2.9	test_dbDatetimeToDate	263
8.39.2.10	test_dbDatetimeToTime	263
8.39.2.11	test_dbDateToDate	263
8.39.2.12	test_dbNumberToNumber	263
8.39.2.13	test_extension	263
8.39.2.14	test_getDateDiff	263
8.39.2.15	test_gOpt	263
8.39.2.16	test_isValid	263
8.39.2.17	test_numberToDB	263
8.39.2.18	test_relativePath	263
8.39.2.19	test_searchNameFile	263
8.39.2.20	test_timeDiff	263
8.39.2.21	test_timeToDbTime	263
8.39.2.22	test_traslitterazione	263
8.40	GImage Class Reference	263
8.40.1	Detailed Description	265
8.40.2	Constructor & Destructor Documentation	265
8.40.2.1	__construct	265
8.40.3	Member Function Documentation	265
8.40.3.1	blackThresholdImage	265
8.40.3.2	cloneImage	266
8.40.3.3	crop	266

8.40.3.4	cropCenter	266
8.40.3.5	cropEntropy	266
8.40.3.6	croplImage	267
8.40.3.7	croplImageEntropy	267
8.40.3.8	getEntropy	267
8.40.3.9	getHeight	268
8.40.3.10	getImageHistogram	268
8.40.3.11	getPath	268
8.40.3.12	getResource	268
8.40.3.13	getThumbFromKey	268
8.40.3.14	getWidth	269
8.40.3.15	grayscaleEntropy	269
8.40.3.16	makeThumb	269
8.40.3.17	resize	269
8.40.3.18	resizeImage	269
8.40.3.19	save	270
8.40.3.20	saveTmpImage	270
8.40.3.21	slice	270
8.40.3.22	stream	270
8.40.3.23	thumb	271
8.40.3.24	toKey	271
8.40.4	Field Documentation	271
8.40.4.1	\$_abspath	271
8.40.4.2	\$_dir	271
8.40.4.3	\$_height	271
8.40.4.4	\$_image	271
8.40.4.5	\$_image_type	271
8.40.4.6	\$_table	271
8.40.4.7	\$_tmp_image	271
8.40.4.8	\$_width	272
8.41	graphics Class Reference	272
8.41.1	Detailed Description	273
8.41.2	Constructor & Destructor Documentation	274
8.41.2.1	__construct	274
8.41.3	Member Function Documentation	274
8.41.3.1	getClassElements	274
8.41.3.2	isHeader	274
8.41.3.3	manageGraphics	274
8.41.3.4	outputFunctions	274
8.41.3.5	printFooterAdhoc	275

8.41.3.6	printFooterAdmin	275
8.41.3.7	printFooterMobile	275
8.41.3.8	printFooterPrivate	275
8.41.3.9	printFooterPublic	275
8.41.3.10	printHeaderAdhoc	275
8.41.3.11	printHeaderAdmin	275
8.41.3.12	printHeaderMobile	276
8.41.3.13	printHeaderPrivate	276
8.41.3.14	printHeaderPublic	276
8.41.3.15	render	276
8.41.4	Field Documentation	276
8.41.4.1	\$_title	276
8.42	GraphicsItem Class Reference	276
8.42.1	Detailed Description	278
8.42.2	Constructor & Destructor Documentation	278
8.42.2.1	__construct	278
8.42.3	Member Function Documentation	278
8.42.3.1	__toString	278
8.42.3.2	structure	278
8.42.4	Field Documentation	278
8.42.4.1	\$_extension_img	278
8.42.4.2	\$table	278
8.43	Group Class Reference	278
8.43.1	Detailed Description	280
8.43.2	Constructor & Destructor Documentation	280
8.43.2.1	__construct	280
8.43.3	Member Function Documentation	280
8.43.3.1	__toString	280
8.43.3.2	getList	280
8.43.3.3	getMergeValue	280
8.43.3.4	getPermissions	281
8.43.3.5	setMergeValue	281
8.43.4	Field Documentation	281
8.43.4.1	\$table	281
8.43.4.2	\$table_group_perm	281
8.44	GTag Class Reference	281
8.44.1	Detailed Description	282
8.44.2	Member Function Documentation	282
8.44.2.1	getAllTags	282
8.44.2.2	getContentTags	282

8.44.2.3	getRelatedContents	283
8.44.2.4	getTagsHistogram	284
8.44.2.5	saveContentTags	284
8.44.3	Field Documentation	284
8.44.3.1	\$_table_tag	284
8.44.3.2	\$_table_tag_taggeditem	284
8.45	ImageField Class Reference	284
8.45.1	Detailed Description	287
8.45.2	Constructor & Destructor Documentation	287
8.45.2.1	__construct	287
8.45.3	Member Function Documentation	288
8.45.3.1	delete	288
8.45.3.2	formElement	288
8.45.3.3	getHeighth	288
8.45.3.4	getPrefixFile	288
8.45.3.5	getPrefixThumb	288
8.45.3.6	getResize	289
8.45.3.7	getThumb	289
8.45.3.8	getThumbHeighth	289
8.45.3.9	getThumbWidth	289
8.45.3.10	getWidth	289
8.45.3.11	resizeImage	289
8.45.3.12	saveFile	290
8.45.3.13	saveImage	290
8.45.3.14	setHeight	290
8.45.3.15	setPrefixFile	290
8.45.3.16	setPrefixThumb	291
8.45.3.17	setResize	291
8.45.3.18	setThumb	291
8.45.3.19	setThumbHeight	291
8.45.3.20	setThumbWidth	292
8.45.3.21	setWidth	293
8.45.4	Field Documentation	293
8.45.4.1	\$_height	293
8.45.4.2	\$_prefix_file	293
8.45.4.3	\$_prefix_thumb	293
8.45.4.4	\$_resize	293
8.45.4.5	\$_thumb	293
8.45.4.6	\$_thumb_height	293
8.45.4.7	\$_thumb_width	293

8.45.4.8	<code>\$_width</code>	293
8.45.4.9	<code>_IMAGE_GIF_</code>	293
8.45.4.10	<code>_IMAGE_JPG_</code>	293
8.45.4.11	<code>_IMAGE_PNG_</code>	293
8.46	index Class Reference	293
8.46.1	Detailed Description	294
8.46.2	Constructor & Destructor Documentation	295
8.46.2.1	<code>__construct</code>	295
8.46.3	Member Function Documentation	295
8.46.3.1	<code>admin_page</code>	295
8.46.3.2	<code>modulesManageArray</code>	295
8.46.3.3	<code>outputFunctions</code>	295
8.46.3.4	<code>sysModulesManageArray</code>	296
8.47	InputForm Class Reference	296
8.47.1	Detailed Description	297
8.47.2	Constructor & Destructor Documentation	297
8.47.2.1	<code>__construct</code>	297
8.47.3	Member Function Documentation	297
8.47.3.1	<code>checkbox</code>	297
8.47.3.2	<code>date</code>	297
8.47.3.3	<code>editor</code>	298
8.47.3.4	<code>email</code>	298
8.47.3.5	<code>file</code>	298
8.47.3.6	<code>hidden</code>	299
8.47.3.7	<code>multicheck</code>	299
8.47.3.8	<code>noinput</code>	299
8.47.3.9	<code>password</code>	300
8.47.3.10	<code>radio</code>	300
8.47.3.11	<code>select</code>	301
8.47.3.12	<code>text</code>	301
8.47.3.13	<code>textarea</code>	302
8.47.4	Field Documentation	302
8.47.4.1	<code>\$_form</code>	302
8.48	instruments Class Reference	302
8.48.1	Detailed Description	303
8.48.2	Constructor & Destructor Documentation	304
8.48.2.1	<code>__construct</code>	304
8.48.3	Member Function Documentation	304
8.48.3.1	<code>links</code>	304
8.48.3.2	<code>manageInstruments</code>	304

8.48.3.3	mimeTypes	304
8.49	IntegerField Class Reference	304
8.49.1	Detailed Description	305
8.49.2	Constructor & Destructor Documentation	306
8.49.2.1	__construct	306
8.49.3	Member Function Documentation	306
8.49.3.1	formElement	306
8.50	Javascript Class Reference	306
8.50.1	Detailed Description	307
8.50.2	Member Function Documentation	307
8.50.2.1	abiMapLib	307
8.50.2.2	analytics	307
8.50.2.3	onLoadFunction	307
8.50.2.4	sharethis	308
8.50.2.5	slimboxLib	308
8.50.2.6	vendor	308
8.51	JSMin Class Reference	308
8.51.1	Detailed Description	309
8.51.2	Constructor & Destructor Documentation	310
8.51.2.1	__construct	310
8.51.3	Member Function Documentation	310
8.51.3.1	action	310
8.51.3.2	get	311
8.51.3.3	isAlphaNum	311
8.51.3.4	min	311
8.51.3.5	minify	311
8.51.3.6	next	311
8.51.3.7	peek	312
8.51.4	Field Documentation	312
8.51.4.1	\$a	312
8.51.4.2	\$b	312
8.51.4.3	\$input	312
8.51.4.4	\$inputIndex	312
8.51.4.5	\$inputLength	312
8.51.4.6	\$lookAhead	312
8.51.4.7	\$output	312
8.51.4.8	ACTION_DELETE_A	312
8.51.4.9	ACTION_DELETE_A_B	312
8.51.4.10	ACTION_KEEP_A	312
8.51.4.11	ORD_LF	312

8.51.4.12	ORD_SPACE	312
8.52	JSMInException Class Reference	312
8.52.1	Detailed Description	313
8.53	Lang Class Reference	313
8.53.1	Detailed Description	315
8.53.2	Constructor & Destructor Documentation	315
8.53.2.1	__construct	315
8.53.3	Member Function Documentation	315
8.53.3.1	__toString	315
8.53.3.2	code	315
8.53.3.3	countryCodes	316
8.53.3.4	getMainLang	316
8.53.3.5	languageCodes	316
8.53.3.6	structure	316
8.53.4	Field Documentation	316
8.53.4.1	\$table	316
8.54	language Class Reference	317
8.54.1	Detailed Description	318
8.54.2	Constructor & Destructor Documentation	318
8.54.2.1	__construct	318
8.54.3	Member Function Documentation	318
8.54.3.1	choiceLanguage	318
8.54.3.2	manageLang	319
8.54.3.3	manageLanguage	320
8.54.3.4	outputFunctions	320
8.54.4	Field Documentation	320
8.54.4.1	\$_flag_language	320
8.54.4.2	\$_options	320
8.54.4.3	\$_optionsLabels	320
8.54.4.4	\$_title	320
8.54.4.5	FLAG_PREFIX	320
8.54.4.6	FLAG_SUFFIX	320
8.55	layout Class Reference	320
8.55.1	Detailed Description	322
8.55.2	Constructor & Destructor Documentation	323
8.55.2.1	__construct	323
8.55.3	Member Function Documentation	323
8.55.3.1	actionCss	323
8.55.3.2	actionDelCss	323
8.55.3.3	actionDelSkin	324

8.55.3.4	actionDelTemplate	324
8.55.3.5	actionFiles	324
8.55.3.6	actionSkin	325
8.55.3.7	actionTemplate	325
8.55.3.8	cssList	325
8.55.3.9	formFiles	325
8.55.3.10	info	326
8.55.3.11	manageCss	326
8.55.3.12	manageLayout	326
8.55.3.13	manageSkin	326
8.55.3.14	manageTemplate	326
8.55.3.15	manageView	327
8.55.3.16	modulesCodeList	327
8.55.3.17	modulesList	327
8.55.3.18	skinList	327
8.55.3.19	templateList	328
8.55.3.20	viewList	328
8.56	Ldap Class Reference	328
8.56.1	Detailed Description	329
8.56.2	Constructor & Destructor Documentation	329
8.56.2.1	__construct	329
8.56.3	Member Function Documentation	329
8.56.3.1	authentication	329
8.56.3.2	binding	330
8.56.3.3	checkValidUsername	330
8.56.3.4	connection	330
8.56.3.5	printError	330
8.56.3.6	search	330
8.56.4	Field Documentation	331
8.56.4.1	\$_auth	331
8.56.4.2	\$_debug	331
8.56.4.3	\$_filter_search	331
8.56.4.4	\$_justthese_search	331
8.56.4.5	\$_ldap_base_dn	331
8.56.4.6	\$_ldap_domain	331
8.56.4.7	\$_ldap_ds	331
8.56.4.8	\$_ldap_host	331
8.56.4.9	\$_ldap_log	331
8.56.4.10	\$_ldap_password	331
8.56.4.11	\$_ldap_port	331

8.56.4.12	<code>\$_ldap_protocol_version</code>	331
8.56.4.13	<code>\$_ldap_username</code>	331
8.56.4.14	<code>\$_test_anonymous</code>	331
8.56.4.15	<code>\$_test_search</code>	331
8.56.4.16	<code>\$_user_name</code>	331
8.56.4.17	<code>\$_user_password</code>	331
8.57	Loader Class Reference	331
8.57.1	Detailed Description	332
8.57.2	Member Function Documentation	332
8.57.2.1	<code>import</code>	332
8.57.2.2	<code>load</code>	332
8.57.2.3	<code>singleton</code>	333
8.58	Locale Class Reference	334
8.58.1	Detailed Description	335
8.58.2	Constructor & Destructor Documentation	337
8.58.2.1	<code>__construct</code>	337
8.58.3	Member Function Documentation	338
8.58.3.1	<code>detectCodes</code>	338
8.58.3.2	<code>get</code>	338
8.58.3.3	<code>get_languages</code>	338
8.58.3.4	<code>init</code>	339
8.58.3.5	<code>initGettext</code>	339
8.58.3.6	<code>pathToFile</code>	339
8.58.3.7	<code>setLanguage</code>	339
8.58.3.8	<code>userLanguage</code>	339
8.58.4	Field Documentation	340
8.58.4.1	<code>\$_session</code>	340
8.58.4.2	<code>\$_strings</code>	340
8.59	LogAccess Class Reference	340
8.59.1	Detailed Description	341
8.59.2	Constructor & Destructor Documentation	341
8.59.2.1	<code>__construct</code>	341
8.59.3	Member Function Documentation	341
8.59.3.1	<code>getCountForUser</code>	341
8.59.3.2	<code>getLastForUser</code>	341
8.59.4	Field Documentation	342
8.59.4.1	<code>\$table</code>	342
8.60	Logger Class Reference	342
8.60.1	Detailed Description	342
8.60.2	Member Function Documentation	343

8.60.2.1	exceptionReportAdmins	343
8.60.2.2	manageException	343
8.60.2.3	messageReportAdmins	343
8.60.2.4	stackTraceHtml	343
8.60.2.5	systemVariablesHtml	343
8.61	ManyToManyField Class Reference	344
8.61.1	Detailed Description	345
8.61.2	Constructor & Destructor Documentation	345
8.61.2.1	__construct	345
8.61.3	Member Function Documentation	346
8.61.3.1	__toString	346
8.61.3.2	clean	346
8.61.3.3	filterWhereClause	347
8.61.3.4	formElement	347
8.61.3.5	getJoinTable	347
8.61.3.6	getJoinTableId	348
8.61.3.7	getJoinTableM2mId	348
8.61.4	Field Documentation	348
8.61.4.1	\$_add_related	348
8.61.4.2	\$_add_related_url	348
8.61.4.3	\$_enum	348
8.61.4.4	\$_join_table	348
8.61.4.5	\$_join_table_id	348
8.61.4.6	\$_join_table_m2m_id	348
8.61.4.7	\$_m2m	348
8.61.4.8	\$_m2m_order	348
8.61.4.9	\$_m2m_where	348
8.62	ManyToManyInlineField Class Reference	348
8.62.1	Detailed Description	350
8.62.2	Constructor & Destructor Documentation	350
8.62.2.1	__construct	350
8.62.3	Member Function Documentation	351
8.62.3.1	__toString	351
8.62.3.2	clean	351
8.62.3.3	filterWhereClause	351
8.62.3.4	formElement	351
8.62.3.5	getEnum	352
8.62.4	Field Documentation	352
8.62.4.1	\$_enum	352
8.62.4.2	\$_m2m	352

8.62.4.3	<code>\$_m2m_controller</code>	352
8.62.4.4	<code>\$_m2m_order</code>	352
8.62.4.5	<code>\$_m2m_where</code>	352
8.63	ManyToManyThroughField Class Reference	352
8.63.1	Detailed Description	354
8.63.2	Constructor & Destructor Documentation	354
8.63.2.1	<code>__construct</code>	354
8.63.3	Member Function Documentation	354
8.63.3.1	<code>__toString</code>	354
8.63.3.2	<code>clean</code>	354
8.63.3.3	<code>filterWhereClause</code>	355
8.63.3.4	<code>formElement</code>	355
8.63.3.5	<code>getController</code>	355
8.63.3.6	<code>getM2m</code>	356
8.63.3.7	<code>getModelTableId</code>	356
8.63.3.8	<code>getTable</code>	356
8.63.4	Field Documentation	356
8.63.4.1	<code>\$_controller</code>	356
8.63.4.2	<code>\$_enum</code>	356
8.63.4.3	<code>\$_m2m</code>	356
8.63.4.4	<code>\$_m2m_controller</code>	356
8.63.4.5	<code>\$_model_table_id</code>	356
8.64	menu Class Reference	356
8.64.1	Detailed Description	358
8.64.2	Constructor & Destructor Documentation	359
8.64.2.1	<code>__construct</code>	359
8.64.3	Member Function Documentation	359
8.64.3.1	<code>actionDelMenuVoice</code>	359
8.64.3.2	<code>actionMenuVoice</code>	359
8.64.3.3	<code>actionUpdateOrder</code>	359
8.64.3.4	<code>breadCrumbs</code>	360
8.64.3.5	<code>deleteInstance</code>	360
8.64.3.6	<code>formMenuVoice</code>	360
8.64.3.7	<code>getClassElements</code>	360
8.64.3.8	<code>getTree</code>	360
8.64.3.9	<code>jsSearchModulesLib</code>	361
8.64.3.10	<code>jsSortLib</code>	361
8.64.3.11	<code>listMenu</code>	361
8.64.3.12	<code>manageDoc</code>	361
8.64.3.13	<code>outputFunctions</code>	362

8.64.3.14	pathToSelectedVoice	362
8.64.3.15	printItemsClass	362
8.64.3.16	printItemsList	362
8.64.3.17	printItemsPage	363
8.64.3.18	render	363
8.64.3.19	renderMenuAdmin	363
8.64.3.20	searchModules	364
8.64.4	Field Documentation	364
8.64.4.1	\$_cache	364
8.64.4.2	\$_ico_more	364
8.64.4.3	\$_menu_functions_list	364
8.64.4.4	\$_options	364
8.64.4.5	\$_optionsLabels	364
8.64.4.6	\$_tbl_opt	364
8.64.4.7	\$_title	364
8.65	MenuVoice Class Reference	364
8.65.1	Detailed Description	366
8.65.2	Constructor & Destructor Documentation	366
8.65.2.1	__construct	366
8.65.3	Member Function Documentation	366
8.65.3.1	deleteInstanceVoices	366
8.65.3.2	deleteVoice	367
8.65.3.3	formVoice	367
8.65.3.4	get	367
8.65.3.5	getChildren	367
8.65.3.6	getSelectedVoice	367
8.65.3.7	initOrderList	368
8.65.3.8	updateOrderList	368
8.65.3.9	userCanSee	368
8.65.4	Field Documentation	368
8.65.4.1	\$_gform	368
8.65.4.2	\$_tbl_user_role	368
8.65.4.3	\$tbl_voices	368
8.66	MobileDetect Class Reference	368
8.66.1	Detailed Description	369
8.66.2	Constructor & Destructor Documentation	369
8.66.2.1	__construct	369
8.66.3	Member Function Documentation	369
8.66.3.1	__call	369
8.66.3.2	isDevice	369

8.66.3.3	isMobile	370
8.66.4	Field Documentation	370
8.66.4.1	\$accept	370
8.66.4.2	\$devices	370
8.66.4.3	\$isAndroid	370
8.66.4.4	\$isBlackberry	370
8.66.4.5	\$isGeneric	370
8.66.4.6	\$isIphone	370
8.66.4.7	\$isMobile	370
8.66.4.8	\$isOpera	370
8.66.4.9	\$isPalm	370
8.66.4.10	\$isWindows	370
8.66.4.11	\$userAgent	370
8.67	Model Class Reference	371
8.67.1	Detailed Description	374
8.67.2	Constructor & Destructor Documentation	374
8.67.2.1	__construct	374
8.67.3	Member Function Documentation	375
8.67.3.1	__get	375
8.67.3.2	__set	375
8.67.3.3	__toString	375
8.67.3.4	addm2m	375
8.67.3.5	addm2mthrough	376
8.67.3.6	checkIsConstraint	376
8.67.3.7	dataType	376
8.67.3.8	delete	376
8.67.3.9	deleteDbData	377
8.67.3.10	deleteInstance	377
8.67.3.11	deletem2m	377
8.67.3.12	deletem2mthrough	377
8.67.3.13	deletem2mthroughField	377
8.67.3.14	fieldLabel	377
8.67.3.15	getController	378
8.67.3.16	getFromSlug	378
8.67.3.17	getModelLabel	378
8.67.3.18	getSelectOptionsFromObjects	378
8.67.3.19	getStructure	378
8.67.3.20	getTable	379
8.67.3.21	isConstraintError	379
8.67.3.22	m2mtObject	379

8.67.3.23	ml	379
8.67.3.24	objects	379
8.67.3.25	save	380
8.67.3.26	savem2m	380
8.67.3.27	setChecksConstraint	380
8.67.3.28	setIsConstraint	380
8.67.3.29	structure	381
8.67.3.30	updateStructure	381
8.67.4	Field Documentation	382
8.67.4.1	\$_check_is_constraint	382
8.67.4.2	\$_chgP	382
8.67.4.3	\$_controller	382
8.67.4.4	\$_db	382
8.67.4.5	\$_fields_label	382
8.67.4.6	\$_is_constraint	382
8.67.4.7	\$_lng_dft	382
8.67.4.8	\$_lng_nav	382
8.67.4.9	\$_locale	382
8.67.4.10	\$_m2m	382
8.67.4.11	\$_m2mt	382
8.67.4.12	\$_model_label	382
8.67.4.13	\$_p	382
8.67.4.14	\$_registry	382
8.67.4.15	\$_request	382
8.67.4.16	\$_structure	382
8.67.4.17	\$_tbl_data	382
8.67.4.18	\$_trd	382
8.68	module Class Reference	382
8.68.1	Detailed Description	384
8.68.2	Constructor & Destructor Documentation	384
8.68.2.1	__construct	384
8.68.3	Member Function Documentation	384
8.68.3.1	actionEditModule	384
8.68.3.2	actionEditModuleActive	385
8.68.3.3	actionInsertModule	385
8.68.3.4	actionModule	385
8.68.3.5	actionRemoveModule	386
8.68.3.6	createMdlFolders	386
8.68.3.7	deleteModuleInstance	386
8.68.3.8	formActivateModule	387

8.68.3.9	formModule	388
8.68.3.10	formRemoveModule	388
8.68.3.11	listModule	388
8.68.3.12	manageModule	388
8.69	ModuleApp Class Reference	389
8.69.1	Detailed Description	390
8.69.2	Constructor & Destructor Documentation	390
8.69.2.1	__construct	390
8.69.3	Member Function Documentation	390
8.69.3.1	__toString	390
8.69.3.2	className	391
8.69.3.3	classNameNs	391
8.69.3.4	getFromName	391
8.69.4	Field Documentation	391
8.69.4.1	\$table	391
8.70	ModuleInstance Class Reference	391
8.70.1	Detailed Description	392
8.70.2	Constructor & Destructor Documentation	393
8.70.2.1	__construct	393
8.70.3	Member Function Documentation	393
8.70.3.1	className	393
8.70.3.2	classNameNs	393
8.70.3.3	getFromModuleApp	393
8.70.3.4	getFromName	393
8.70.3.5	moduleApp	394
8.70.4	Field Documentation	394
8.70.4.1	\$table	394
8.71	mysql Class Reference	394
8.71.1	Detailed Description	396
8.71.2	Constructor & Destructor Documentation	397
8.71.2.1	__construct	397
8.71.3	Member Function Documentation	397
8.71.3.1	actionquery	397
8.71.3.2	affected	397
8.71.3.3	autoIncValue	397
8.71.3.4	begin	398
8.71.3.5	closeConnection	398
8.71.3.6	columnHasValue	398
8.71.3.7	commit	398
8.71.3.8	concat	398

8.71.3.9	conformType	398
8.71.3.10	delete	399
8.71.3.11	distinct	399
8.71.3.12	drop	399
8.71.3.13	dump	399
8.71.3.14	dumpDatabase	399
8.71.3.15	escapeString	400
8.71.3.16	execQuery	400
8.71.3.17	fieldInformations	400
8.71.3.18	freeresult	400
8.71.3.19	getFieldFromId	400
8.71.3.20	getFieldsName	400
8.71.3.21	getInfoQuery	401
8.71.3.22	getlastid	401
8.71.3.23	getNumRecords	401
8.71.3.24	getTableStructure	401
8.71.3.25	insert	401
8.71.3.26	join	401
8.71.3.27	limit	401
8.71.3.28	multiActionquery	402
8.71.3.29	openConnection	402
8.71.3.30	query	402
8.71.3.31	restore	402
8.71.3.32	resultselect	402
8.71.3.33	rollback	402
8.71.3.34	select	402
8.71.3.35	selectquery	403
8.71.3.36	setconnection	403
8.71.3.37	setnumberrows	403
8.71.3.38	setsql	403
8.71.3.39	setUtf8	403
8.71.3.40	tableexists	403
8.71.3.41	union	403
8.71.3.42	update	403
8.71.4	Field Documentation	403
8.71.4.1	\$_affected	403
8.71.4.2	\$_cache	403
8.71.4.3	\$_cnt	403
8.71.4.4	\$_connection	403
8.71.4.5	\$_db_charset	404

8.71.4.6	<code>\$_db_host</code>	404
8.71.4.7	<code>\$_db_name</code>	404
8.71.4.8	<code>\$_db_password</code>	404
8.71.4.9	<code>\$_db_user</code>	404
8.71.4.10	<code>\$_dbconn</code>	404
8.71.4.11	<code>\$_dbresults</code>	404
8.71.4.12	<code>\$_debug</code>	404
8.71.4.13	<code>\$_enable_cache</code>	404
8.71.4.14	<code>\$_lastid</code>	404
8.71.4.15	<code>\$_numberrows</code>	404
8.71.4.16	<code>\$_qry</code>	404
8.71.4.17	<code>\$_rows</code>	404
8.71.4.18	<code>\$_sql</code>	404
8.72	Options Class Reference	404
8.72.1	Detailed Description	405
8.72.2	Constructor & Destructor Documentation	405
8.72.2.1	<code>__construct</code>	405
8.72.3	Member Function Documentation	406
8.72.3.1	<code>actionOptions</code>	406
8.72.3.2	<code>editableField</code>	406
8.72.3.3	<code>manageDoc</code>	406
8.72.3.4	<code>setData</code>	406
8.72.4	Field Documentation	407
8.72.4.1	<code>\$_action</code>	407
8.72.4.2	<code>\$_class</code>	407
8.72.4.3	<code>\$_class_prefix</code>	407
8.72.4.4	<code>\$_db</code>	407
8.72.4.5	<code>\$_home</code>	407
8.72.4.6	<code>\$_instance</code>	407
8.72.4.7	<code>\$_module_app</code>	407
8.72.4.8	<code>\$_tbl_options</code>	407
8.72.4.9	<code>\$_title</code>	407
8.73	OutputCache Class Reference	407
8.73.1	Detailed Description	408
8.73.2	Constructor & Destructor Documentation	409
8.73.2.1	<code>__construct</code>	409
8.73.3	Member Function Documentation	409
8.73.3.1	<code>start</code>	409
8.73.3.2	<code>stop</code>	409
8.74	page Class Reference	409

8.74.1	Detailed Description	412
8.74.2	Constructor & Destructor Documentation	414
8.74.2.1	__construct	414
8.74.3	Member Function Documentation	414
8.74.3.1	accessPage	414
8.74.3.2	actionComment	414
8.74.3.3	box	415
8.74.3.4	checkSlug	415
8.74.3.5	commentNotification	415
8.74.3.6	explanationTemplate	415
8.74.3.7	feedRSS	416
8.74.3.8	formComment	417
8.74.3.9	getAddPath	417
8.74.3.10	getBasePath	417
8.74.3.11	getClassElements	417
8.74.3.12	getUrlPage	417
8.74.3.13	manageComment	418
8.74.3.14	manageCtg	418
8.74.3.15	manageEntry	418
8.74.3.16	managePage	418
8.74.3.17	outputFunctions	419
8.74.3.18	parseTemplate	419
8.74.3.19	permissions	419
8.74.3.20	relatedContentsList	419
8.74.3.21	replaceTplVar	419
8.74.3.22	searchSite	420
8.74.3.23	searchSiteResult	420
8.74.3.24	showcase	420
8.74.3.25	systemNewsletterList	420
8.74.3.26	systemNewsletterRender	421
8.74.3.27	view	422
8.74.4	Field Documentation	422
8.74.4.1	\$_box_tpl_code	422
8.74.4.2	\$_comment_moderation	422
8.74.4.3	\$_comment_notification	422
8.74.4.4	\$_entry_tpl_code	422
8.74.4.5	\$_newsletter_entries_number	422
8.74.4.6	\$_newsletter_tpl_code	422
8.74.4.7	\$_showcase_auto_interval	423
8.74.4.8	\$_showcase_auto_start	423

8.74.4.9	\$_showcase_number	423
8.74.4.10	\$_showcase_title	423
8.74.4.11	\$_showcase_tpl_code	423
8.74.4.12	\$_tbl_opt	423
8.75	PageCategory Class Reference	423
8.75.1	Detailed Description	424
8.75.2	Constructor & Destructor Documentation	424
8.75.2.1	__construct	424
8.75.3	Member Function Documentation	425
8.75.3.1	__toString	425
8.75.4	Field Documentation	425
8.75.4.1	\$table	425
8.76	PageComment Class Reference	425
8.76.1	Detailed Description	426
8.76.2	Constructor & Destructor Documentation	427
8.76.2.1	__construct	427
8.76.3	Member Function Documentation	427
8.76.3.1	__toString	427
8.76.3.2	deleteFromEntry	427
8.76.3.3	getCountFromEntry	427
8.76.3.4	getTree	428
8.76.3.5	notifyComment	429
8.76.3.6	save	429
8.76.3.7	structure	429
8.76.4	Field Documentation	429
8.76.4.1	\$table	429
8.77	PageEntry Class Reference	430
8.77.1	Detailed Description	431
8.77.2	Constructor & Destructor Documentation	431
8.77.2.1	__construct	431
8.77.3	Member Function Documentation	432
8.77.3.1	__toString	432
8.77.3.2	accessWhere	432
8.77.3.3	delete	432
8.77.3.4	get	432
8.77.3.5	getCount	433
8.77.3.6	getFromSlug	433
8.77.3.7	getUrl	434
8.77.3.8	imgPath	434
8.77.3.9	save	434

8.77.3.10 structure	434
8.77.4 Field Documentation	435
8.77.4.1 \$_extension_img	435
8.77.4.2 \$table	435
8.78 Paginator Class Reference	435
8.78.1 Detailed Description	436
8.78.2 Constructor & Destructor Documentation	436
8.78.2.1 __construct	436
8.78.3 Member Function Documentation	436
8.78.3.1 getCurrentPage	436
8.78.3.2 limit	436
8.78.3.3 limitQuery	436
8.78.3.4 navigator	437
8.78.3.5 pages	437
8.78.3.6 pagination	437
8.78.3.7 setCurrentPage	437
8.78.3.8 summary	437
8.78.3.9 urlPage	437
8.78.4 Field Documentation	438
8.78.4.1 \$_current_page	438
8.78.4.2 \$_interval	438
8.78.4.3 \$_items_for_page	438
8.78.4.4 \$_items_number	438
8.78.4.5 \$_pages_number	438
8.79 PaginatorTest Class Reference	438
8.79.1 Detailed Description	439
8.79.2 Member Function Documentation	439
8.79.2.1 test_getCurrentPage	439
8.79.2.2 test_limit	439
8.79.2.3 test_limitQuery	439
8.79.2.4 test_pages	439
8.79.2.5 test_summary	440
8.80 Permission Class Reference	440
8.80.1 Detailed Description	441
8.80.2 Constructor & Destructor Documentation	442
8.80.2.1 __construct	442
8.80.3 Member Function Documentation	443
8.80.3.1 __toString	443
8.80.3.2 getForMulticheck	443
8.80.3.3 getFromFullCode	443

8.80.3.4	getGroupedPermissions	443
8.80.3.5	getList	443
8.80.3.6	structure	444
8.80.4	Field Documentation	444
8.80.4.1	\$table	444
8.80.4.2	\$table_perm_user	444
8.81	PhpModule Class Reference	444
8.81.1	Detailed Description	445
8.81.2	Constructor & Destructor Documentation	445
8.81.2.1	__construct	445
8.81.3	Member Function Documentation	446
8.81.3.1	actionPhpModule	446
8.81.3.2	formPhpModule	446
8.81.3.3	setContent	446
8.81.3.4	setInstance	446
8.81.3.5	structure	447
8.81.4	Field Documentation	447
8.81.4.1	\$_interface	447
8.81.4.2	\$_tbl_data	447
8.81.4.3	\$table	447
8.82	phpModuleView Class Reference	447
8.82.1	Detailed Description	449
8.82.2	Constructor & Destructor Documentation	449
8.82.2.1	__construct	449
8.82.3	Member Function Documentation	449
8.82.3.1	deleteInstance	449
8.82.3.2	getClassElements	449
8.82.3.3	info	450
8.82.3.4	manageDoc	450
8.82.3.5	outputFunctions	450
8.82.3.6	permissions	450
8.82.3.7	viewList	450
8.82.4	Field Documentation	450
8.82.4.1	\$_blackList	450
8.82.4.2	\$_options	450
8.82.4.3	\$_optionsLabels	450
8.82.4.4	\$_tbl_opt	450
8.82.4.5	\$_title	450
8.82.4.6	\$_title_visible	451
8.83	ReCaptchaResponse Class Reference	451

8.83.1 Detailed Description	451
8.83.2 Field Documentation	451
8.83.2.1 \$error	451
8.83.2.2 \$is_valid	451
8.84 Redirect Class Reference	451
8.84.1 Detailed Description	452
8.84.2 Constructor & Destructor Documentation	453
8.84.2.1 __construct	453
8.84.3 Member Function Documentation	453
8.84.3.1 redirectUrl	453
8.84.3.2 send	453
8.84.3.3 sendHeaders	453
8.84.4 Field Documentation	453
8.84.4.1 \$_url	453
8.85 Registry Class Reference	454
8.85.1 Detailed Description	455
8.85.2 Member Function Documentation	455
8.85.2.1 __get	455
8.85.2.2 __set	456
8.85.2.3 addCss	456
8.85.2.4 addHeadLink	456
8.85.2.5 addJs	456
8.85.2.6 addMeta	457
8.85.2.7 propertyExists	457
8.85.2.8 variables	457
8.85.3 Field Documentation	458
8.85.3.1 \$vars	458
8.86 Request Class Reference	458
8.86.1 Detailed Description	459
8.86.2 Constructor & Destructor Documentation	460
8.86.2.1 __construct	460
8.86.3 Member Function Documentation	460
8.86.3.1 checkGETKey	460
8.86.3.2 checkPOSTKey	460
8.86.3.3 isSecure	460
8.86.3.4 updateUrl	460
8.86.3.5 valueOrNull	461
8.86.4 Field Documentation	461
8.86.4.1 \$absolute_url	461
8.86.4.2 \$COOKIES	461

8.86.4.3	\$FILES	461
8.86.4.4	\$GET	461
8.86.4.5	\$META	461
8.86.4.6	\$method	461
8.86.4.7	\$path	461
8.86.4.8	\$POST	461
8.86.4.9	\$query_string	461
8.86.4.10	\$REQUEST	461
8.86.4.11	\$request_uri	461
8.86.4.12	\$root_absolute_url	461
8.86.4.13	\$session	461
8.86.4.14	\$url	461
8.86.4.15	\$user	461
8.87	Response Class Reference	461
8.87.1	Detailed Description	463
8.87.2	Constructor & Destructor Documentation	463
8.87.2.1	__construct	463
8.87.3	Member Function Documentation	463
8.87.3.1	__invoke	463
8.87.3.2	getContent	464
8.87.3.3	send	464
8.87.3.4	sendContent	464
8.87.3.5	sendHeaders	464
8.87.3.6	setContent	464
8.87.3.7	setContentType	464
8.87.3.8	setEncoding	465
8.87.3.9	setHeaders	465
8.87.3.10	setStatus	465
8.87.4	Field Documentation	465
8.87.4.1	\$_content	465
8.87.4.2	\$_content_type	465
8.87.4.3	\$_encoding	465
8.87.4.4	\$_headers	465
8.87.4.5	\$_request	465
8.87.4.6	\$_status_code	466
8.87.4.7	\$_status_text	466
8.88	ResponseFile Class Reference	466
8.88.1	Detailed Description	467
8.88.2	Constructor & Destructor Documentation	467
8.88.2.1	__construct	467

8.88.3	Member Function Documentation	467
8.88.3.1	sendHeaders	467
8.88.3.2	setDispositionType	468
8.88.4	Field Documentation	468
8.88.4.1	\$_content_disposition	468
8.88.4.2	\$_filename	468
8.89	ResponseForbidden Class Reference	468
8.89.1	Detailed Description	469
8.89.2	Constructor & Destructor Documentation	469
8.89.2.1	__construct	469
8.89.3	Member Function Documentation	470
8.89.3.1	sendContent	470
8.90	ResponseJson Class Reference	470
8.90.1	Detailed Description	471
8.90.2	Constructor & Destructor Documentation	471
8.90.2.1	__construct	471
8.91	ResponseNotFound Class Reference	471
8.91.1	Detailed Description	472
8.91.2	Constructor & Destructor Documentation	472
8.91.2.1	__construct	472
8.91.3	Member Function Documentation	473
8.91.3.1	sendContent	473
8.92	ResponseServerError Class Reference	473
8.92.1	Detailed Description	474
8.92.2	Constructor & Destructor Documentation	474
8.92.2.1	__construct	474
8.92.3	Member Function Documentation	474
8.92.3.1	sendContent	474
8.93	Router Class Reference	474
8.93.1	Detailed Description	476
8.93.2	Constructor & Destructor Documentation	476
8.93.2.1	__construct	476
8.93.3	Member Function Documentation	476
8.93.3.1	isRoute	476
8.93.3.2	link	477
8.93.3.3	rewritePathInfo	477
8.93.3.4	rewriteQueryString	477
8.93.3.5	route	478
8.93.3.6	setUrlParams	478
8.93.3.7	transformPathQueryString	478

8.93.3.8	urlRewrite	478
8.93.4	Field Documentation	478
8.93.4.1	\$_controller_view	478
8.93.4.2	\$_registry	478
8.93.4.3	\$_request	478
8.93.4.4	\$_url_class	478
8.93.4.5	\$_url_instance	478
8.93.4.6	\$_url_method	479
8.93.4.7	EVT_NAME	479
8.94	Search Class Reference	479
8.94.1	Detailed Description	479
8.94.2	Constructor & Destructor Documentation	480
8.94.2.1	__construct	480
8.94.3	Member Function Documentation	480
8.94.3.1	clearSearchString	480
8.94.3.2	getKeywords	480
8.94.3.3	getSearchResults	480
8.94.3.4	makeQuery	481
8.94.4	Field Documentation	481
8.94.4.1	\$_table	481
8.95	searchSite Class Reference	481
8.95.1	Detailed Description	483
8.95.2	Constructor & Destructor Documentation	483
8.95.2.1	__construct	483
8.95.3	Member Function Documentation	483
8.95.3.1	checkOptions	483
8.95.3.2	form	484
8.95.3.3	getClassElements	484
8.95.3.4	info	484
8.95.3.5	manageSearchSite	484
8.95.3.6	outputFunctions	484
8.95.3.7	results	484
8.95.4	Field Documentation	485
8.95.4.1	\$_inst_mdl	485
8.95.4.2	\$_options	485
8.95.4.3	\$_optionsLabels	485
8.95.4.4	\$_sys_mdl	485
8.95.4.5	\$_title	485
8.95.4.6	\$_view_dir	485
8.96	Session Class Reference	485

8.96.1	Detailed Description	487
8.96.2	Constructor & Destructor Documentation	487
8.96.2.1	__construct	487
8.96.3	Member Function Documentation	487
8.96.3.1	__get	487
8.96.3.2	__isset	487
8.96.3.3	__set	488
8.96.3.4	__unset	488
8.96.3.5	destroy	488
8.96.3.6	startSession	488
8.96.4	Field Documentation	488
8.96.4.1	\$_vars	488
8.96.4.2	\$sessionState	488
8.96.4.3	SESSION_NOT_STARTED	489
8.96.4.4	SESSION_STARTED	489
8.97	Singleton Class Reference	489
8.97.1	Detailed Description	490
8.97.2	Constructor & Destructor Documentation	490
8.97.2.1	__construct	490
8.97.3	Member Function Documentation	490
8.97.3.1	__clone	490
8.97.3.2	__sleep	490
8.97.3.3	__wakeup	490
8.97.3.4	instance	490
8.97.3.5	instance_to_class	491
8.97.4	Field Documentation	491
8.97.4.1	\$_instances	491
8.98	Skin Class Reference	491
8.98.1	Detailed Description	493
8.98.2	Constructor & Destructor Documentation	493
8.98.2.1	__construct	493
8.98.3	Member Function Documentation	493
8.98.3.1	actionDelSkin	493
8.98.3.2	actionSkin	494
8.98.3.3	formDelSkin	494
8.98.3.4	formSkin	494
8.98.3.5	getSkin	494
8.98.3.6	layoutInfo	495
8.98.3.7	newSkinPriority	495
8.98.3.8	removeCss	495

8.98.3.9	<code>removeTemplate</code>	495
8.98.3.10	<code>sortUp</code>	495
8.98.4	Field Documentation	496
8.98.4.1	<code>\$_interface</code>	496
8.98.4.2	<code>\$_tbl_data</code>	496
8.98.4.3	<code>\$table</code>	496
8.99	SlugField Class Reference	496
8.99.1	Detailed Description	497
8.99.2	Constructor & Destructor Documentation	497
8.99.2.1	<code>__construct</code>	497
8.99.3	Member Function Documentation	498
8.99.3.1	<code>formElement</code>	498
8.99.4	Field Documentation	499
8.99.4.1	<code>\$_autofill</code>	499
8.99.4.2	<code>\$_js</code>	499
8.100	sqlcode Class Reference	499
8.100.1	Detailed Description	499
8.100.2	Constructor & Destructor Documentation	500
8.100.2.1	<code>__construct</code>	500
8.100.3	Member Function Documentation	500
8.100.3.1	<code>admin_actionUser</code>	500
8.100.4	Field Documentation	500
8.100.4.1	<code>\$_dbms</code>	500
8.101	sqlsrv Class Reference	500
8.101.1	Detailed Description	503
8.101.2	Constructor & Destructor Documentation	503
8.101.2.1	<code>__construct</code>	503
8.101.3	Member Function Documentation	504
8.101.3.1	<code>actionquery</code>	504
8.101.3.2	<code>affected</code>	504
8.101.3.3	<code>autoIncValue</code>	504
8.101.3.4	<code>begin</code>	504
8.101.3.5	<code>closeConnection</code>	504
8.101.3.6	<code>columnHasValue</code>	504
8.101.3.7	<code>commit</code>	504
8.101.3.8	<code>concat</code>	505
8.101.3.9	<code>conformType</code>	505
8.101.3.10	<code>delete</code>	506
8.101.3.11	<code>distinct</code>	506
8.101.3.12	<code>drop</code>	506

8.101.3.13dump	506
8.101.3.14dumpDatabase	506
8.101.3.15escapeString	506
8.101.3.16execQuery	507
8.101.3.17fieldInformations	507
8.101.3.18freeresult	507
8.101.3.19getCheckConstraint	507
8.101.3.20getConstraintType	508
8.101.3.21getDataType	508
8.101.3.22getFieldFromId	508
8.101.3.23getFieldLength	508
8.101.3.24getFieldsName	509
8.101.3.25getInfoQuery	509
8.101.3.26getInformationKey	509
8.101.3.27getlastid	509
8.101.3.28getNumRecords	510
8.101.3.29getTableStructure	510
8.101.3.30insert	510
8.101.3.31join	510
8.101.3.32limit	510
8.101.3.33limitQuery	510
8.101.3.34listTables	510
8.101.3.35multiActionquery	510
8.101.3.36openConnection	511
8.101.3.37query	511
8.101.3.38restore	511
8.101.3.39resultselect	511
8.101.3.40rollback	511
8.101.3.41select	511
8.101.3.42selectquery	512
8.101.3.43setconnection	512
8.101.3.44setnumberrows	512
8.101.3.45setsql	512
8.101.3.46tableexists	512
8.101.3.47union	512
8.101.3.48update	512
8.101.4 Field Documentation	512
8.101.4.1 \$_affected	512
8.101.4.2 \$_cache	513
8.101.4.3 \$_cnt	513

8.101.4.4 <code>\$_connection</code>	513
8.101.4.5 <code>\$_db_charset</code>	513
8.101.4.6 <code>\$_db_host</code>	513
8.101.4.7 <code>\$_db_name</code>	513
8.101.4.8 <code>\$_db_password</code>	513
8.101.4.9 <code>\$_db_user</code>	513
8.101.4.10 <code>\$_dbconn</code>	513
8.101.4.11 <code>\$_dbresults</code>	513
8.101.4.12 <code>\$_debug</code>	513
8.101.4.13 <code>\$_enable_cache</code>	513
8.101.4.14 <code>\$_lastid</code>	513
8.101.4.15 <code>\$_numberrows</code>	513
8.101.4.16 <code>\$_qry</code>	513
8.101.4.17 <code>\$_rows</code>	513
8.101.4.18 <code>\$_sql</code>	513
8.102statistics Class Reference	513
8.102.1 Detailed Description	515
8.102.2 Constructor & Destructor Documentation	515
8.102.2.1 <code>__construct</code>	515
8.102.3 Member Function Documentation	515
8.102.3.1 <code>export</code>	515
8.102.3.2 <code>info</code>	515
8.102.3.3 <code>manageStatistics</code>	515
8.102.3.4 <code>viewlogAccess</code>	516
8.102.4 Field Documentation	516
8.102.4.1 <code>\$_tbl_log_access</code>	516
8.103sysClass Class Reference	516
8.103.1 Detailed Description	518
8.103.2 Constructor & Destructor Documentation	518
8.103.2.1 <code>__construct</code>	518
8.103.3 Member Function Documentation	518
8.103.3.1 <code>actionEditSysClass</code>	518
8.103.3.2 <code>actionEditSysClassActive</code>	518
8.103.3.3 <code>actionInsertSysClass</code>	519
8.103.3.4 <code>actionManualSysClass</code>	519
8.103.3.5 <code>actionRemoveSysClass</code>	519
8.103.3.6 <code>actionUpgradeSysClass</code>	520
8.103.3.7 <code>formActivateSysClass</code>	520
8.103.3.8 <code>formEditSysClass</code>	520
8.103.3.9 <code>formInsertSysClass</code>	520

8.103.3.10formManualSysClass	521
8.103.3.11formRemoveSysClass	521
8.103.3.12formUpgradeSysClass	521
8.103.3.13info	521
8.103.3.14manageSysClass	521
8.103.3.15sysClassList	521
8.103.3.16upgradeFolders	522
8.103.4 Field Documentation	522
8.103.4.1 \$_title	522
8.103.4.2 PACKAGE_EXTENSION	522
8.104sysconf Class Reference	522
8.104.1 Detailed Description	523
8.104.2 Constructor & Destructor Documentation	524
8.104.2.1 __construct	524
8.104.3 Member Function Documentation	524
8.104.3.1 manageSysconf	524
8.105sysfunc Class Reference	524
8.105.1 Detailed Description	525
8.105.2 Constructor & Destructor Documentation	526
8.105.2.1 __construct	526
8.105.3 Member Function Documentation	526
8.105.3.1 page403	526
8.105.3.2 page404	526
8.105.3.3 page500	527
8.106TagField Class Reference	527
8.106.1 Detailed Description	528
8.106.2 Constructor & Destructor Documentation	529
8.106.2.1 __construct	529
8.106.3 Member Function Documentation	529
8.106.3.1 clean	529
8.106.3.2 formElement	529
8.106.3.3 save	529
8.106.3.4 tagCloud	530
8.106.4 Field Documentation	530
8.106.4.1 \$_model_controller_class	530
8.106.4.2 \$_model_controller_instance	530
8.107Template Class Reference	530
8.107.1 Detailed Description	533
8.107.2 Constructor & Destructor Documentation	533
8.107.2.1 __construct	533

8.107.3 Member Function Documentation	533
8.107.3.1 actionCopyTemplate	533
8.107.3.2 actionDelTemplate	533
8.107.3.3 actionFreeTemplate	534
8.107.3.4 actionTemplate	534
8.107.3.5 addBlockForm	534
8.107.3.6 cellCtrl	535
8.107.3.7 createTemplate	535
8.107.3.8 deleteBlocks	535
8.107.3.9 formBlock	535
8.107.3.10 formCopyTemplate	536
8.107.3.11 formData	536
8.107.3.12 formDelTemplate	536
8.107.3.13 formFreeTemplate	536
8.107.3.14 formOutline	536
8.107.3.15 formTemplate	537
8.107.3.16 nitBlocksProperties	537
8.107.3.17 layoutInfo	537
8.107.3.18 manageTemplate	537
8.107.3.19 printBlock	537
8.107.3.20 renderNave	538
8.107.3.21 saveBlock	538
8.107.3.22 setFilename	538
8.107.3.23 plBlockForm	539
8.107.4 Field Documentation	539
8.107.4.1 \$_align_dict	539
8.107.4.2 \$_blocks_number	539
8.107.4.3 \$_blocks_properties	539
8.107.4.4 \$_home	539
8.107.4.5 \$_interface	539
8.107.4.6 \$_tbl_data	539
8.107.4.7 \$_um_dict	539
8.107.4.8 \$table	539
8.107.4.9 \$table_block	539
8.108 TextField Class Reference	539
8.108.1 Detailed Description	541
8.108.2 Constructor & Destructor Documentation	541
8.108.2.1 __construct	541
8.108.3 Member Function Documentation	541
8.108.3.1 canBeOrdered	541

8.108.3.2 clean	542
8.108.3.3 filterWhereClause	542
8.108.3.4 formElement	542
8.108.3.5 getTrnsl	542
8.108.3.6 setTrnsl	542
8.108.4 Field Documentation	543
8.108.4.1 \$_trnsl	543
8.109TimeField Class Reference	543
8.109.1 Detailed Description	544
8.109.2 Constructor & Destructor Documentation	544
8.109.2.1 __construct	544
8.109.3 Member Function Documentation	544
8.109.3.1 __toString	544
8.109.3.2 clean	545
8.109.3.3 formElement	545
8.110Translation Class Reference	545
8.110.1 Detailed Description	546
8.110.2 Constructor & Destructor Documentation	546
8.110.2.1 __construct	546
8.110.3 Member Function Documentation	546
8.110.3.1 actionTranslation	546
8.110.3.2 deleteTranslations	547
8.110.3.3 formTranslation	547
8.110.3.4 listItemOrdered	547
8.110.3.5 manageTranslation	547
8.110.3.6 selectTXT	548
8.110.4 Field Documentation	548
8.110.4.1 \$_db	548
8.110.4.2 \$_lng	548
8.110.4.3 \$_lngDft	548
8.110.4.4 \$_registry	548
8.110.4.5 \$_tbl_translation	548
8.111User Class Reference	548
8.111.1 Detailed Description	550
8.111.2 Constructor & Destructor Documentation	551
8.111.2.1 __construct	551
8.111.3 Member Function Documentation	552
8.111.3.1 __toString	552
8.111.3.2 checkEmail	552
8.111.3.3 checkPassword	552

8.111.3.4 checkUsername	553
8.111.3.5 delete	553
8.111.3.6 deleteMoreInfo	553
8.111.3.7 formPassword	554
8.111.3.8 generatePassword	554
8.111.3.9 getFromUserPwd	554
8.111.3.10getMergeValue	555
8.111.3.11getPermissions	555
8.111.3.12getUsersFromPermissions	555
8.111.3.13hasAdminPerm	555
8.111.3.14hasPerm	556
8.111.3.15savePassword	556
8.111.3.16setMergeValue	556
8.111.3.17setMoreInfo	557
8.111.3.18setPassword	557
8.111.3.19structure	557
8.111.4 Field Documentation	557
8.111.4.1 \$extension_media	557
8.111.4.2 \$table	557
8.111.4.3 \$table_groups	557
8.111.4.4 \$table_more	557
8.112View Class Reference	558
8.112.1 Detailed Description	558
8.112.2 Constructor & Destructor Documentation	558
8.112.2.1 __construct	558
8.112.3 Member Function Documentation	559
8.112.3.1 assign	559
8.112.3.2 render	559
8.112.3.3 setViewTpl	559
8.112.4 Field Documentation	559
8.112.4.1 \$_data	559
8.112.4.2 \$_dft_view_folder	560
8.112.4.3 \$_registry	560
8.112.4.4 \$_view_folder	560
8.112.4.5 \$_view_tpl	560
8.113YearField Class Reference	560
8.113.1 Detailed Description	561
8.113.2 Constructor & Destructor Documentation	561
8.113.2.1 __construct	561
8.113.3 Member Function Documentation	562

8.113.3.1 formElement	562
9 File Documentation	563
9.1 app/attachment/class.AttachmentCtg.php File Reference	563
9.1.1 Detailed Description	563
9.2 app/attachment/class.AttachmentItem.php File Reference	563
9.2.1 Detailed Description	564
9.3 app/attachment/class.AttachmentItemAdminTable.php File Reference	564
9.3.1 Detailed Description	564
9.4 app/attachment/class_attachment.php File Reference	564
9.4.1 Detailed Description	565
9.5 app/auth/class.AdminTable_AuthUser.php File Reference	565
9.5.1 Detailed Description	565
9.6 app/auth/class.Group.php File Reference	566
9.6.1 Detailed Description	566
9.7 app/auth/class.Ldap.php File Reference	566
9.7.1 Detailed Description	566
9.8 app/auth/class.Permission.php File Reference	567
9.8.1 Detailed Description	567
9.9 app/auth/class.User.php File Reference	567
9.9.1 Detailed Description	568
9.10 app/auth/class_auth.php File Reference	568
9.10.1 Detailed Description	568
9.11 app/auth/config.Ldap.php File Reference	568
9.11.1 Detailed Description	569
9.12 app/auth/views/login.php File Reference	569
9.12.1 Detailed Description	569
9.13 app/graphics/class.GraphicsItem.php File Reference	569
9.13.1 Detailed Description	570
9.14 app/graphics/class_graphics.php File Reference	570
9.14.1 Detailed Description	570
9.15 app/graphics/views/render.php File Reference	571
9.16 app/menu/views/render.php File Reference	571
9.17 app/index/class_index.php File Reference	571
9.17.1 Detailed Description	571
9.18 app/instruments/class_instruments.php File Reference	571
9.18.1 Detailed Description	572
9.19 app/language/class.Lang.php File Reference	572
9.19.1 Detailed Description	572
9.20 app/language/class_language.php File Reference	572

9.20.1 Detailed Description	573
9.21 app/language/views/choice_language.php File Reference	573
9.21.1 Detailed Description	573
9.22 app/layout/class_layout.php File Reference	574
9.22.1 Detailed Description	574
9.23 app/menu/class.MenuVoice.php File Reference	574
9.23.1 Detailed Description	575
9.24 app/menu/class_menu.php File Reference	575
9.24.1 Detailed Description	575
9.25 app/menu/views/render_mainMenu.php File Reference	575
9.25.1 Detailed Description	576
9.26 app/menu/views/render_menu_admin.php File Reference	576
9.26.1 Detailed Description	576
9.27 app/module/class.ModuleInstance.php File Reference	577
9.27.1 Detailed Description	577
9.28 app/module/class_module.php File Reference	577
9.28.1 Detailed Description	578
9.29 app/page/class.PageCategory.php File Reference	578
9.29.1 Detailed Description	578
9.30 app/page/class.PageComment.php File Reference	578
9.30.1 Detailed Description	579
9.31 app/page/class.PageEntry.php File Reference	579
9.31.1 Detailed Description	579
9.32 app/page/class_page.php File Reference	579
9.32.1 Detailed Description	580
9.33 app/page/language/en_US/page_lang.php File Reference	580
9.34 app/page/language/it_IT/page_lang.php File Reference	580
9.35 app/page/views/box.php File Reference	580
9.35.1 Detailed Description	580
9.36 app/page/views/showcase.php File Reference	581
9.36.1 Detailed Description	581
9.37 app/page/views/view.php File Reference	581
9.37.1 Detailed Description	582
9.38 app/phpModuleView/class.PhpModule.php File Reference	582
9.38.1 Detailed Description	582
9.39 app/phpModuleView/class_phpModuleView.php File Reference	583
9.39.1 Detailed Description	583
9.40 app/searchSite/class_searchSite.php File Reference	583
9.40.1 Detailed Description	583
9.41 app/searchSite/views/form.php File Reference	584

9.41.1 Detailed Description	584
9.42 app/searchSite/views/results.php File Reference	584
9.42.1 Detailed Description	584
9.43 app/statistics/class.LogAccess.php File Reference	585
9.43.1 Detailed Description	585
9.44 app/statistics/class_statistics.php File Reference	585
9.44.1 Detailed Description	586
9.45 app/sysClass/class.ModuleApp.php File Reference	586
9.45.1 Detailed Description	586
9.46 app/sysClass/class_sysClass.php File Reference	586
9.46.1 Detailed Description	587
9.47 app/sysconf/class.Conf.php File Reference	587
9.47.1 Detailed Description	587
9.48 app/sysconf/class_sysconf.php File Reference	587
9.48.1 Detailed Description	588
9.49 app/sysfunc/class_sysfunc.php File Reference	588
9.49.1 Detailed Description	588
9.50 configuration.php File Reference	589
9.50.1 Detailed Description	589
9.50.2 Variable Documentation	589
9.50.2.1 ADMINS	589
9.50.2.2 DB_CHARSET	590
9.50.2.3 DB_DBNAME	590
9.50.2.4 DB_HOST	590
9.50.2.5 DB_PASSWORD	590
9.50.2.6 DB_PORT	590
9.50.2.7 DB_SCHEMA	590
9.50.2.8 DB_USER	590
9.50.2.9 DBMS	590
9.50.2.10 DEBUG	590
9.50.2.11 LDAP_AUTH	590
9.50.2.12 MAX_FILE_SIZE	590
9.50.2.13 SESSION_NAME	590
9.51 index.php File Reference	591
9.51.1 Detailed Description	591
9.52 lib/classes/class.Access.php File Reference	591
9.52.1 Detailed Description	592
9.53 lib/classes/class.AdminTable.php File Reference	592
9.53.1 Detailed Description	592
9.54 lib/classes/class.Cache.php File Reference	592

9.54.1 Detailed Description	593
9.55 lib/classes/class.Captcha.php File Reference	593
9.55.1 Detailed Description	593
9.56 lib/classes/class.Compressor.php File Reference	593
9.56.1 Detailed Description	594
9.57 lib/classes/class.Core.php File Reference	594
9.57.1 Detailed Description	594
9.58 lib/classes/class.Css.php File Reference	595
9.58.1 Detailed Description	595
9.59 lib/classes/class.Db.php File Reference	595
9.59.1 Detailed Description	595
9.60 lib/classes/class.Document.php File Reference	596
9.60.1 Detailed Description	596
9.61 lib/classes/class.Error.php File Reference	596
9.61.1 Detailed Description	597
9.62 lib/classes/class.EventDispatcher.php File Reference	597
9.62.1 Detailed Description	597
9.63 lib/classes/class.Export.php File Reference	597
9.63.1 Detailed Description	598
9.64 lib/classes/class.Form.php File Reference	598
9.64.1 Detailed Description	598
9.65 lib/classes/class.Frontend.php File Reference	598
9.65.1 Detailed Description	599
9.66 lib/classes/class.GImage.php File Reference	599
9.66.1 Detailed Description	599
9.67 lib/classes/class.GTag.php File Reference	600
9.67.1 Detailed Description	600
9.68 lib/classes/class.InputForm.php File Reference	600
9.68.1 Detailed Description	600
9.69 lib/classes/class.Javascript.php File Reference	601
9.69.1 Detailed Description	601
9.70 lib/classes/class.Loader.php File Reference	601
9.70.1 Detailed Description	602
9.70.2 Function Documentation	602
9.70.2.1 __autoload	602
9.71 lib/classes/class.Locale.php File Reference	602
9.71.1 Detailed Description	602
9.72 lib/classes/class.Logger.php File Reference	603
9.72.1 Detailed Description	603
9.73 lib/classes/class.MobileDetect.php File Reference	603

9.73.1 Detailed Description	603
9.74 lib/classes/class.Options.php File Reference	603
9.74.1 Detailed Description	604
9.75 lib/classes/class.Paginator.php File Reference	604
9.75.1 Detailed Description	604
9.76 lib/classes/class.Registry.php File Reference	605
9.76.1 Detailed Description	605
9.77 lib/classes/class.Router.php File Reference	605
9.77.1 Detailed Description	605
9.78 lib/classes/class.Search.php File Reference	606
9.78.1 Detailed Description	606
9.79 lib/classes/class.Session.php File Reference	606
9.79.1 Detailed Description	607
9.80 lib/classes/class.Singleton.php File Reference	607
9.80.1 Detailed Description	607
9.81 lib/classes/class.Skin.php File Reference	607
9.81.1 Detailed Description	608
9.82 lib/classes/class.Template.php File Reference	608
9.82.1 Detailed Description	608
9.83 lib/classes/class.Translation.php File Reference	608
9.83.1 Detailed Description	609
9.84 lib/classes/exceptions/class.Exception403.php File Reference	609
9.84.1 Detailed Description	609
9.85 lib/classes/exceptions/class.Exception404.php File Reference	610
9.85.1 Detailed Description	610
9.86 lib/classes/exceptions/class.Exception500.php File Reference	610
9.86.1 Detailed Description	610
9.87 lib/classes/fields/class.BooleanField.php File Reference	611
9.87.1 Detailed Description	611
9.88 lib/classes/fields/class.CharField.php File Reference	611
9.88.1 Detailed Description	611
9.89 lib/classes/fields/class.DateField.php File Reference	612
9.89.1 Detailed Description	612
9.90 lib/classes/fields/class.DatetimeField.php File Reference	612
9.90.1 Detailed Description	613
9.91 lib/classes/fields/class.DirectoryField.php File Reference	613
9.91.1 Detailed Description	613
9.92 lib/classes/fields/class.EmailField.php File Reference	613
9.92.1 Detailed Description	614
9.93 lib/classes/fields/class.EnumField.php File Reference	614

9.93.1 Detailed Description	614
9.94 lib/classes/fields/class.Field.php File Reference	614
9.94.1 Detailed Description	615
9.95 lib/classes/fields/class.FileField.php File Reference	615
9.95.1 Detailed Description	615
9.96 lib/classes/fields/class.FloatField.php File Reference	616
9.96.1 Detailed Description	616
9.97 lib/classes/fields/class.ForeignKeyField.php File Reference	616
9.97.1 Detailed Description	616
9.98 lib/classes/fields/class.ImageField.php File Reference	617
9.98.1 Detailed Description	617
9.99 lib/classes/fields/class.IntegerField.php File Reference	617
9.99.1 Detailed Description	618
9.100lib/classes/fields/class.ManyToManyField.php File Reference	618
9.100.1 Detailed Description	618
9.101lib/classes/fields/class.ManyToManyInlineField.php File Reference	618
9.101.1 Detailed Description	619
9.102lib/classes/fields/class.ManyToManyThroughField.php File Reference	619
9.102.1 Detailed Description	619
9.103lib/classes/fields/class.SlugField.php File Reference	619
9.103.1 Detailed Description	620
9.104lib/classes/fields/class.TagField.php File Reference	620
9.104.1 Detailed Description	620
9.105lib/classes/fields/class.TextField.php File Reference	621
9.105.1 Detailed Description	621
9.106lib/classes/fields/class.TimeField.php File Reference	621
9.106.1 Detailed Description	621
9.107lib/classes/fields/class.YearField.php File Reference	622
9.107.1 Detailed Description	622
9.108lib/classes/http/class.Redirect.php File Reference	622
9.108.1 Detailed Description	623
9.109lib/classes/http/class.Request.php File Reference	623
9.109.1 Detailed Description	623
9.110lib/classes/http/class.Response.php File Reference	623
9.110.1 Detailed Description	624
9.111lib/classes/http/class.ResponseFile.php File Reference	624
9.111.1 Detailed Description	624
9.112lib/classes/http/class.ResponseForbidden.php File Reference	624
9.112.1 Detailed Description	625
9.113lib/classes/http/class.ResponseJson.php File Reference	625

9.113.1 Detailed Description	625
9.114lib/classes/http/class.ResponseNotFound.php File Reference	626
9.114.1 Detailed Description	626
9.115lib/classes/http/class.ResponseServerError.php File Reference	626
9.115.1 Detailed Description	626
9.116lib/classes/mvc/class.Controller.php File Reference	627
9.116.1 Detailed Description	627
9.117lib/classes/mvc/class.Model.php File Reference	627
9.117.1 Detailed Description	627
9.118lib/classes/mvc/class.View.php File Reference	628
9.118.1 Detailed Description	628
9.119lib/const.php File Reference	628
9.119.1 Detailed Description	629
9.120lib/func.browser.php File Reference	629
9.120.1 Detailed Description	630
9.121lib/func.file.php File Reference	630
9.121.1 Detailed Description	631
9.122lib/func.php File Reference	631
9.122.1 Detailed Description	633
9.123lib/func.var.php File Reference	633
9.123.1 Detailed Description	634
9.124lib/global.php File Reference	634
9.124.1 Detailed Description	635
9.124.2 Function Documentation	635
9.124.2.1 get_app_name_class_ns	635
9.124.2.2 get_app_namespace	635
9.124.2.3 get_model_app_name_class_ns	636
9.124.2.4 get_name_class	636
9.124.2.5 set_name_class	636
9.124.3 Variable Documentation	636
9.124.3.1 else	636
9.125lib/plugin/plugin.mysql.php File Reference	637
9.125.1 Detailed Description	637
9.126lib/plugin/plugin.sqlsrv.php File Reference	637
9.126.1 Detailed Description	637
9.127lib/recaptchalib.php File Reference	638
9.127.1 Detailed Description	638
9.127.2 Function Documentation	638
9.127.2.1 _recaptcha_aes_encrypt	638
9.127.2.2 _recaptcha_aes_pad	638

9.127.2.3 _recaptcha_http_post	638
9.127.2.4 _recaptcha_mailhide_email_parts	639
9.127.2.5 _recaptcha_mailhide_urlbase64	639
9.127.2.6 _recaptcha_qsencode	639
9.127.2.7 recaptcha_check_answer	639
9.127.2.8 recaptcha_get_html	639
9.127.2.9 recaptcha_get_signup_url	640
9.127.2.10recaptcha_mailhide_html	640
9.127.2.11recaptcha_mailhide_url	640
9.127.3 Variable Documentation	640
9.127.3.1 RECAPTCHA_API_SECURE_SERVER	640
9.127.3.2 RECAPTCHA_API_SERVER	640
9.127.3.3 RECAPTCHA_VERIFY_SERVER	640
9.128lib/sqlcode.php File Reference	640
9.128.1 Detailed Description	641
9.129README.md File Reference	641
9.130settings.php File Reference	641
9.130.1 Detailed Description	642
9.131templates/_popup.php File Reference	642
9.132templates/admin_home.php File Reference	642
9.133templates/admin_page.php File Reference	642
9.134templates/default_free_tpl.php File Reference	643
9.135templates/home.php File Reference	643
9.136templates/home_mobile.php File Reference	643
9.137templates/page.php File Reference	643
9.138templates/pages_mobile.php File Reference	643
9.139test/class.Paginator_test.php File Reference	643
9.139.1 Detailed Description	643
9.140test/func_test.php File Reference	643
9.140.1 Detailed Description	644
9.141test/include.php File Reference	644
9.141.1 Detailed Description	644
9.141.2 Variable Documentation	644
9.141.2.1 SITE_ROOT	644
9.141.2.2 SITE_WWW	644
9.142views/403.php File Reference	644
9.142.1 Detailed Description	645
9.143views/404.php File Reference	645
9.143.1 Detailed Description	645
9.144views/500.php File Reference	646

9.144.1 Detailed Description	646
9.145views/admin_table_form.php File Reference	646
9.145.1 Detailed Description	646
9.146views/admin_table_list.php File Reference	647
9.146.1 Detailed Description	647
9.147views/logger_stack_trace.php File Reference	647
9.147.1 Detailed Description	648
9.148views/logger_system_variables.php File Reference	648
9.148.1 Detailed Description	648
9.149views/paginator_navigator.php File Reference	649
9.149.1 Detailed Description	649
9.150views/paginator_pagination.php File Reference	649
9.150.1 Detailed Description	649
9.151views/related_contents_list.php File Reference	650
9.151.1 Detailed Description	650
9.152views/section.php File Reference	650
9.152.1 Detailed Description	651
9.153views/tab.php File Reference	651
9.153.1 Detailed Description	651
9.154views/table.php File Reference	652
9.154.1 Detailed Description	652
Index	655

Chapter 1

gino CMS

gino è un CMS framework scritto in PHP che fornisce tutti gli strumenti necessari a creare un sito web e gestirne i contenuti in modo semplice ed efficace, svincolando l'amministratore da conoscenze tecniche di programmazione.

Caratteristiche

- utilizzo di patterns di programmazione, MVC, Mediator, Factory Method, Singleton
- gestione multilingua
- gestione utenti e gruppi di utenti
- gestione layout libero e a blocchi con interfaccia di manipolazione
- gestione di layout legati a singoli url o gruppi di url attraverso espressioni regolari
- gestione cache dei contenuti e delle query
- modulo pagine
- modulo allegati
- modulo gestione header/footer
- modulo lingue
- modulo menu
- modulo per la gestione di codice custom php
- modulo statistiche di accesso
- gestione errori http 404, 403, 500
- ricerca full-text
- installazione di pacchetti aggiuntivi con un click
- integrazione bootstrap 3 (less) e font-awesome, possibilità di creare layout responsive
- integrazione mootools > 1.4
- gestione dispositivi mobile separata se necessario
- moduli aggiuntivi disponibili sull'[account github di Otto](#)

Releases

2014/12/23 | v 2.0.0 | Nuova versione 2015/02/02 | v 2.0.1 | Hotfix 2015/03/03 | v 2.1.0 | Feature 2015/03/06 | v 2.1.1 | Hotfix 2015/03/13 | v 2.1.2 | Hotfix 2015/03/27 | v 2.2.0 | Feature 2015/04/01 | v 2.2.1 | Hotfix 2015/06/03 | v 2.3.0 | Feature 2015/06/30 | v 2.3.1 | Hotfix 2015/11/11 | v 2.4.0 | Feature

Copyright

Copyright © 2005-2015 [Otto srl](#), [MIT License](#)

Chapter 2

Todo List

Global **AdminTable::permission** (\$options, \$fname)

Implementare il metodo che restituisce TRUE se l'utente ha il permesso di agire sul campo, FALSE altrimenti.

Parameters

array	<i>\$options</i>	array associativo di opzioni
string	<i>\$fname</i>	nome del campo

Returns

TRUE

Global **AdminTable::readFile** (\$model, \$path_to_file, \$options=array())

Implementare la funzionalità di importazione del file

Parameters

\Gino\↔ Model	<i>\$model</i>	Gino.Model
string	<i>\$path_to_file</i>	
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• verify_items (array): valori da verificare nel processo di importazione, nel formato array(nome_campo=>valore[,])• dump (boolean): effettua il dump della tabella prima dell'importazione• dump_path (string): percorso del file di dump

Returns

string (log dell'importazione)

Global **AdminTable::restore** (\$table, \$filename, \$options=array())

Implementare la funzionalità di restore di un file

See also

Gino.Db::restore()

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

string	<i>\$filename</i>	nome del file da importare
array	<i>\$options</i>	arra

Returns

boolean

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Gino	Namespace principale di gino	23
Gino\App	56
Gino\App\Attachment	Namespace dell'applicazione Attachment , che gestisce allegati categorizzati	57
Gino\App\Auth	Namespace dell'applicazione Auth , per la gestione di utenti, gruppi e permessi	58
Gino\App\Graphics	Namespace dell'applicazione Graphics , che gestisce in maniera semplificata header e footer	59
Gino\App\Index	Namespace dell'applicazione Index , che gestisce la home amministrativa	59
Gino\App\Instruments	Namespace dell'applicazione Instruments , per la visualizzazione di utilità quali percorsi e mime types	59
Gino\App\Language	Namespace dell'applicazione Language , che gestisce le lingue installate sul sistema e disponibili per le traduzioni	60
Gino\App\Layout	Namespace dell'applicazione Layout , che gestisce i layout del sistema	60
Gino\App\Menu	Namespace dell'applicazione Menu , che gestisce istanze di menu	60
Gino\App\Module	Namespace dell'applicazione Module , che gestisce la creazione/rimozione di istanze di moduli di sistema	60
Gino\App\Page	Namespace dell'applicazione Page , per la gestione di pagine categorizzate	61
Gino\App\PhpModuleView	Namespace dell'applicazione PhpModuleView , che gestisce output generati direttamente da codice php salvato da database	61
Gino\App\SearchSite	Namespace dell'applicazione SearchSite , che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca	62
Gino\App\Statistics	Namespace dell'applicazione Statistics , che gestisce le statistiche di accesso all'area privata	62
Gino\App\SysClass	Namespace dell'applicazione SysClass , che gestisce l'installazione/upgrade/rimozione di classi di sistema	62

Gino\App\Sysconf	
Namespace dell'applicazione Sysconf , che gestisce le impostazioni di sistema	63
Gino\App\Sysfunc	
Namespace dell'applicazione Sysfunc , per la gestione della visualizzazione ddi pagine di errore	63
Gino\Exception	
Namespace che comprende tutte le classi di tipo Exception	63
Gino\Http	
Namespace che comprende tutte le classi per la gestione di richieste e risposte http	64
Gino\Plugin	
Namespace che comprende classi di tipo plugin	64
Gino\Test	
Namespace che comprende gli UNIT TEST di classi e funzioni di gino	64

Chapter 4

Hierarchical Index

4.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Access	67
AdminTable	70
AttachmentItemAdminTable	100
AdminTable_AuthUser	86
Build	??
BooleanBuild	??
CharBuild	??
DateBuild	??
DatetimeBuild	??
DirectoryBuild	??
EmailBuild	??
EnumBuild	??
FileBuild	??
ImageBuild	??
FloatBuild	??
ForeignKeyBuild	??
IntegerBuild	??
YearBuild	??
ManyToManyBuild	??
ManyToManyThroughBuild	??
MulticheckBuild	??
SlugBuild	??
TagBuild	??
TextBuild	??
TimeBuild	??
cache	
DataCache	144
Cache	114
OutputCache	407
Captcha	117
Compressor	124
Controller	130
attachment	89
auth	101
graphics	272
index	293
instruments	302

language	317
layout	320
menu	356
module	382
page	409
phpModuleView	447
searchSite	481
statistics	513
sysClass	516
sysconf	522
sysfunc	524
Core	136
DbManager	154
mysql	394
pdo	??
pdo_mysql	??
pdo_sqlsrv	??
sqlsrv	500
Document	174
Error	185
Exception	
Exception403	189
Exception404	191
Exception500	193
JSMinException	312
Export	194
Field	201
BooleanField	111
CharField	120
DateField	146
DatetimeField	149
DirectoryField	169
EmailField	179
EnumField	182
FileField	214
ImageField	284
FloatField	224
ForeignKeyField	228
IntegerField	304
YearField	560
ManyToManyField	344
ManyToManyThroughField	352
MulticheckField	??
SlugField	496
TagField	527
TextField	539
TimeField	543
Form	231
Frontend	257
GImage	263
GTag	281
Javascript	306
JSMin	308
Ldap	328
Loader	331
Logger	342
MobileDetect	368

Model	371
AttachmentCtg	92
AttachmentItem	95
Group	278
Permission	440
RegistrationProfile	??
RegistrationRequest	??
User	548
GraphicsItem	276
Lang	313
MenuVoice	364
ModuleInstance	391
PageCategory	423
PageComment	425
PageEntry	430
PhpModule	444
LogAccess	340
ModuleApp	389
Conf	128
Css	139
Skin	491
Template	530
Options	404
Paginator	435
PHPUnit_Framework_TestCase	
FuncTest	261
PaginatorTest	438
plugin_phpfastcache	??
ReCaptchaResponse	451
Response	461
Redirect	451
ResponseFile	466
ResponseForbidden	468
ResponseJson	470
ResponseNotFound	471
ResponsePdf	??
ResponseServerError	473
Search	479
Singleton	489
EventDispatcher	187
Request	458
Locale	334
Router	474
Session	485
singleton	
Db	153
Registry	454
sqlcode	499
SqlParse	??
Translation	545
View	558
Widget	??
CheckboxWidget	??
ConstantWidget	??
DatetimeWidget	??
DateWidget	??
EditorWidget	??

EmailWidget	??
FileWidget	??
FloatWidget	??
HiddenWidget	??
ImageWidget	??
MulticheckWidget	??
PasswordWidget	??
RadioWidget	??
SelectWidget	??
TextareaWidget	??
TextWidget	??
TimeWidget	??
UnitWidget	??

Chapter 5

Data Structure Index

5.1 Data Structures

Here are the data structures with brief descriptions:

Access	Classe per la gestione dell'autenticazione ed accesso alla funzionalità	67
AdminTable	Gestisce l'interfaccia di amministrazione di un modello con inserimento, modifica ed eliminazione	70
AdminTable_AuthUser	Sovrascrive la classe AdminTable	86
attachment	Classe controller del modulo di gestione di archivi di file categorizzati	89
AttachmentCtg	Classe di tipo Gino.Model che rappresenta una categoria di allegati	92
AttachmentItem	Classe di tipo Gino.Model che rappresenta una singolo allegato	95
AttachmentItemAdminTable	Estende la classe Gino.AdminTable per gestire il salvataggio dell'allegato quando viene modificata la directory (categoria)	100
auth	Classe di tipo Gino.Controller per la gestione degli utenti, gruppi e permessi	101
BooleanBuild	Gestisce i campi di tipo BOOLEAN	??
BooleanField	Rappresenta campi di tipo BOOLEAN	111
Build	Gestisce i campi del modello	??
Cache	Classe che viene estesa da Gino.OutputCache() e Gino.DataCache()	114
Captcha	Classe per la generazione di immagini captcha	117
CharBuild	Gestisce i campi di tipo stringa (CHAR, VARCHAR)	??
CharField	Campi di tipo stringa (CHAR, VARCHAR)	120
CheckboxWidget	Campi checkbox nei form	??
Compressor	Classe per la compressione di css e js (merge e minify)	124
Conf	Classe di tipo Gino.Model che gestisce le impostazioni del sistema	128

ConstantWidget	
Campi di tipo costante nei form	??
Controller	
Classe astratta primitiva di tipo Controller (MVC), dalla quale tutti i controller delle singole app discendono	130
Core	
Gestisce una Gino.Http.Request ed invia una Gino.Http.Response adeguata	136
Css	
Libreria per la gestione dei file css dei singoli moduli e dei file css del layout (da associare alle skin)	139
DataCache	
Memorizza le strutture dati scrivendo su file	144
DateBuild	
Gestisce i campi di tipo data	??
DateField	
Campo di tipo data	146
DatetimeBuild	
Gestisce i campi di tipo DATETIME	??
DatetimeField	
Campo di tipo DATETIME	149
DatetimeWidget	
Campi di tipo datetime nei form	??
DateWidget	
Campi di tipo date nei form	??
Db	
Classe Factory e Singleton usata per creare oggetti che si interfacciano al database	153
DbManager	
Interfaccia per le librerie di connessione al database	154
DirectoryBuild	
Gestisce i campi di tipo DIRECTORY	??
DirectoryField	
Campo di tipo DIRECTORY	169
Document	
Crea il documento html da inviare come corpo della risposta HTTP	174
EditorWidget	
Campi di tipo editor nei form	??
EmailBuild	
Campo di tipo EMAIL	??
EmailField	
Campo di tipo EMAIL	179
EmailWidget	
Campi di tipo email nei form	??
EnumBuild	
Gestisce i campi di tipo ENUM	??
EnumField	
Campo di tipo ENUM	182
Error	
Classe per la gestione di errori da mostrare all'utente E' possibile anche mostrare dei warning per facilitare il debug	185
EventDispatcher	
Classe di tipo Mediator e Singleton per la gestione di segnali (eventi) tra oggetti	187
Exception403	
Classe per la gestione di eccezioni 403	189
Exception404	
Classe per la gestione di eccezioni 404	191
Exception500	
Classe per la gestione di eccezioni 500	193

Export	Libreria per l'esportazione di tabelle o dati	194
Field	Gestisce le caratteristiche del tipo di campo (colonne)	201
FileBuild	Campo di tipo FILE	??
FileField	Campo di tipo FILE	214
FileWidget	Campi di tipo file nei form	??
FloatBuild	Gestisce campi di tipo decimale (FLOAT, DOUBLE, DECIMAL)	??
FloatField	Campo di tipo decimale (FLOAT, DOUBLE, DECIMAL)	224
FloatWidget	Campi di tipo float nei form	??
ForeignKeyBuild	Campo di tipo chiave esterna	??
ForeignKeyField	Campo di tipo chiave esterna	228
Form	Classe per la creazione ed il salvataggio dati di un form	231
Frontend	Libreria per la gestione dei file di front-end dei singoli moduli (css e viste)	257
FuncTest	Classe di tipo PHPUnit_Framework_TestCase per testare il file func.php	261
GImage	Classe per il trattamento di immagini	263
graphics	Classe di tipo Gino.Controller per la gestione personalizzata degli header e footer del sistema	272
GraphicsItem	Classe di tipo Gino.Model che rappresenta un header/footer	276
Group	Classe tipo Gino.Model che rappresenta un gruppo di utenti	278
GTag	Classe per il trattamento di campi di tipo tag	281
HiddenWidget	Campi di tipo hidden nei form	??
ImageBuild	Gestisce campi di tipo IMMAGINE	??
ImageField	Campo di tipo IMMAGINE	284
ImageWidget	Campi di tipo immagine nei form	??
index	Classe di tipo Gino.Controller del modulo che gestisce la home amministrativa	293
instruments	Classe di tipo Gino.Controller per la gestione di strumenti aggiuntivi di gino	302
IntegerBuild	Campo di tipo INTERO	??
IntegerField	Campo di tipo INTERO	304
Javascript	Contiene i metodi per includere alcuni javascript	306
JSMIn	PHP implementation of Douglas Crockford's JSMIn	308
JSMInException	JSMInException	312

Lang	Classe di tipo Gino.Model che rappresenta una lingua installata sul sistema	313
language	Classe di tipo Gino.Controller per la gestione delle lingue disponibili per le traduzioni	317
layout	Gestisce il layout dell'applicazione raggruppando le funzionalità fornite dalle librerie dei css, template e skin	320
Ldap	Libreria di connessione ai server Ldap	328
Loader	Loader di classi di tipo Gino.Model e classi di sistema	331
Locale	Libreria per la gestione delle traduzioni che non utilizzano le librerie gettext	334
LogAccess	Modello che rappresenta un login sul sistema	340
Logger	Classe per la notifica di log di sistema	342
ManyToManyBuild	Gestisce i campi di tipo many to many	??
ManyToManyField	Campo di tipo many to many	344
ManyToManyThroughBuild	Gestisce i campi di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive	??
ManyToManyThroughField	Campo di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive	352
menu	Classe di tipo Gino.Controller per la gestione dei menu	356
MenuVoice	Classe di tipo Gino.Model che rappresenta una voce di menu	364
MobileDetect	Verifica se una richiesta HTTP è stata effettuata da un dispositivo mobile	368
Model	Classe astratta che definisce un modello, cioè un oggetto che rappresenta una tabella su database	371
module	Classe di tipo Gino.Controller per la gestione di istanze di moduli di sistema	382
ModuleApp	Classe di tipo Gino.Model che rappresenta un modulo di sistema	389
ModuleInstance	Classe di tipo Gino.Model che rappresenta un'istanza di un modulo di sistema	391
MulticheckBuild	Gestisce i campi di tipo multicheck	??
MulticheckField	Campo di tipo multicheck	??
MulticheckWidget	Campi di tipo multicheck nei form	??
mysql	Libreria di connessione ai database MySQL	394
Options	Gestisce le opzioni di classe, costruendo il form ed effettuando l'action	404
OutputCache	Memorizza gli output (text, html, xml) scrivendo su file	407
page	Classe di tipo Gino.Controller per la gestione delle pagine	409
PageCategory	Classe tipo Gino.Model che rappresenta una categoria di pagine	423

PageComment	
Classe tipo Gino.Model che rappresenta un commento ad una pagina	425
PageEntry	
Classe tipo Gino.Model che rappresenta una pagina	430
Paginator	
Gestisce la paginazione di elementi dati il numero di elementi totali ed il numero di elementi per pagina, ricava i limiti per creare il sottoinsieme di elementi da mostrare e gestisce la navigazione tra le pagine	435
PaginatorTest	
Classe di tipo PHPUnit_Framework_TestCase per testare la classe Gino.Paginator	438
PasswordWidget	
Campi di tipo password nei form	??
pdo	
Libreria di connessione ai database	??
pdo_mysql	
Driver specifico per la connessione a un database MYSQL attraverso la libreria PDO	??
pdo_sqlsrv	
Driver specifico per la connessione a un database SQL Server attraverso la libreria PDO	??
Permission	
Classe tipo Gino.Model che rappresenta un permesso	440
PhpModule	
Classe di tipo Gino.Model che rappresenta un output scritto in codice php	444
phpModuleView	
Permette la creazione di moduli di classe in grado di eseguire codice php completamente personalizzabile	447
plugin_phpfastcache	
Interfaccia alla libreria phpfastcache	??
RadioWidget	
Campi di tipo radio button nei form	??
ReCaptchaResponse	
A ReCaptchaResponse is returned from recaptcha_check_answer()	451
Redirect	
Subclass di Gino.Http.Response per gestire reindirizzamenti	451
RegistrationProfile	
Classe tipo Gino.Model che rappresenta un profilo di registrazione utenti	??
RegistrationRequest	
Classe tipo Gino.Model che rappresenta una richiesta di registrazione	??
Registry	
Registro di gino	454
Request	
Wrapper di una richiesta HTTP	458
Response	
Wrapper di una risposta HTTP	461
ResponseFile	
Subclass di per gestire stream di file	466
ResponseForbidden	
Subclass di Gino.Http.Response per gestire risposte a seguito di errori 403	468
ResponseJson	
Subclass di per gestire risposte in formato json	470
ResponseNotFound	
Subclass di Gino.Http.Response per gestire risposte a seguito di errori 404	471
ResponsePdf	
Subclass di per gestire risposte in formato pdf	??
ResponseServerError	
Subclass di Gino.Http.Response per gestire risposte a seguito di errori interni (code 500)	473
Router	
Gestisce il routing di una request HTTP, chiamando la classe e metodo che devono fornire risposta	474

Search	Libreria per ricerche full text pesate sulle tabelle	479
searchSite	Gestisce le ricerche full text sui contenuti dell'applicazione	481
SelectWidget	Campi di tipo select nei form	??
Session	Classe per la gestione delle variabili di sessione	485
Singleton	Classe astratta per implementazione del Singleton pattern	489
Skin	Libreria per la gestione delle Skin	491
SlugBuild	Getisce i campi di tipo SLUG (CHAR, VARCHAR)	??
SlugField	Campo di tipo SLUG (CHAR, VARCHAR)	496
sqlcode	Contiene query e codice sql personalizzato	499
SqlParse	Classe per gestire il parser dei file sql (Code freely adapted from phpBB Group)	??
sqlsrv	Libreria di connessione ai database SQL Server	500
statistics	Statistiche degli accessi all'area privata	513
sysClass	Classe di tipo Gino.Controller per la gestione dei moduli di sistema	516
sysconf	Classe di tipo Gino.Controller per la gestione delle principali impostazioni di sistema	522
sysfunc	Classe di tipo Gino.Controller che gestisce metodi che stampano pagine di errore	524
TagBuild	Gestisce i campi per inserimento tag	??
TagField	Classe per la gestione di campi per inserimento tag	527
Template	Libreria per la gestione dei template del documento html da associare alle Gino	530
TextareaWidget	Campi di tipo textarea nei form	??
TextBuild	Campo di tipo TEXT	??
TextField	Campo di tipo TEXT	539
TextWidget	Campi di tipo testo nei form	??
TimeBuild	Gestisce i campi di tipo TIME	??
TimeField	Campo di tipo TIME	543
TimeWidget	Campi di tipo orario nei form	??
Translation	Classe per la gestione delle traduzioni	545
UnitWidget	Insieme di campi di un modello	??
User	Classe di tipo Gino.Model che rappresenta un utente	548
View	Gestisce le viste, impostando il template e ritornando l'output	558

Widget	Gestisce i campi nei form	??
YearBuild	Gestisce i campi di tipo ANNO	??
YearField	Campo di tipo ANNO	560

Chapter 6

File Index

6.1 File List

Here is a list of all files with brief descriptions:

configuration.php	File di configurazione	589
index.php	Entry point di gino CMS	591
settings.php	Definizione dei percorsi dell'applicazione	641
app/attachment/class.AttachmentCtg.php	Contiene la definizione ed implementazione della classe Gino.App.Attachment.AttachmentCtg	563
app/attachment/class.AttachmentItem.php	Contiene la definizione ed implementazione della classe Gino.App.Attachment.AttachmentItem	563
app/attachment/class.AttachmentItemAdminTable.php	Contiene la definizione ed implementazione della classe Gino.App.Attachment.AttachmentItemAdminTable	564
app/attachment/class_attachment.php	Contiene la definizione ed implementazione della classe Gino.App.Attachment.attachment	564
app/auth/class.AdminTable_AuthUser.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.AdminTable_AuthUser	565
app/auth/class.Group.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.Group	566
app/auth/class.Ldap.php	Contiene la definizione ed implementazione della class Gino.App.Auth.Ldap	566
app/auth/class.Permission.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.Permission	567
app/auth/class.RegistrationProfile.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.RegistrationProfile	??
app/auth/class.RegistrationRequest.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.RegistrationRequest	??
app/auth/class.User.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.User	567
app/auth/class_auth.php	Contiene la definizione ed implementazione della classe Gino.App.Auth.auth	568
app/auth/config.Ldap.php	File di configurazione per l'accesso a LDAP	568
app/auth/views/activate_profile.php	Template registrazione nuovo profilo per utente già registrato	??
app/auth/views/activation_email_message.php	Template messaggio mail inviata a seguito di conferma indirizzo email	??

app/auth/views/activation_email_object.php	
Template oggetto mail inviata a seguito di conferma indirizzo email	??
app/auth/views/confirmation_result.php	
Template risultato conferma registrazione	??
app/auth/views/data_recovery_request.php	
Template richiesta di recupero credenziali	??
app/auth/views/data_recovery_request_processed.php	
Template processing richiesta di recupero credenziali	??
app/auth/views/data_recovery_success.php	
Template successo richiesta di recupero credenziali	??
app/auth/views/login.php	
Template form di login	569
app/auth/views/profile.php	??
app/auth/views/registration.php	
Template form di registrazione	??
app/auth/views/registration_email_message.php	
Template corpo della mail inviata a seguito di registrazione	??
app/auth/views/registration_email_object.php	
Template oggetto mail inviata a seguito di registrazione	??
app/auth/views/registration_result.php	
Template risultato registrazione	??
app/graphics/class.GraphicsItem.php	
Contiene la definizione ed implementazione della classe Gino.App.Graphics.GraphicsItem	569
app/graphics/class_graphics.php	
Contiene la definizione ed implementazione della classe Gino.App.Graphics.graphics	570
app/graphics/views/render.php	571
app/index/class_index.php	
Contiene la definizione ed implementazione della classe Gino.App.Index.index	571
app/instruments/class_instruments.php	
Contiene la definizione ed implementazione della classe Gino.App.Instruments.instruments	571
app/language/class.Lang.php	
Contiene la classe definizione ed implementazione della classe Gino.App.Language.Lang	572
app/language/class_language.php	
Contiene la classe definizione ed implementazione della classe Gino.App.Language.language	572
app/language/views/choice_language.php	
Template della vista scelta lingua	573
app/layout/class_layout.php	
Contiene la definizione ed implementazione della classe Gino.App.Layout.layout	574
app/layout/layout.js	??
app/layout/layout_single.js	??
app/menu/class.MenuVoice.php	
Contiene la definizione ed implementazione della classe Gino.App.Menu.MenuVoice	574
app/menu/class_menu.php	
Contiene la definizione ed implementazione della classe Gino.App.Menu.menu	575
app/menu/views/render.php	571
app/menu/views/render_mainMenu.php	
Template visualizzazione menu	575
app/menu/views/render_menu_admin.php	
Template visualizzazione menu	576
app/module/class.ModuleInstance.php	
Contiene la definizione ed implementazione della classe Gino.App.Module.ModuleInstance	577
app/module/class_module.php	
Contiene la definizione ed implementazione della classe Gino.App.Module.module	577
app/page/class.PageCategory.php	
Contiene la definizione ed implementazione della classe Gino.App.Page.PageCategory	578
app/page/class.PageComment.php	
Contiene la definizione ed implementazione della classe Gino.App.Page.PageComment	578

app/page/class.PageEntry.php	Contiene la definizione ed implementazione della classe Gino.App.Page.PageEntry	579
app/page/class_page.php	Contiene la definizione ed implementazione della classe Gino.App.Page.page	579
app/page/page.js		??
app/page/prettify.js		??
app/page/language/en_US/page_lang.php		580
app/page/language/it_IT/page_lang.php		580
app/page/views/box.php	Template per la vista delle pagine inserite nel template	580
app/page/views/showcase.php	Template per la vista vetrina	581
app/page/views/view.php	Template per la vista delle pagine	581
app/phpModuleView/class.PhpModule.php	Contiene la definizione ed implementazione della classe Gino.App.PhpModuleView.PhpModule	582
app/phpModuleView/class_phpModuleView.php	Contiene la definizione ed implementazione della classe Gino.App.PhpModuleView.phpModuleView	583
app/searchSite/class_searchSite.php	Contiene la definizione ed implementazione della classe Gino.App.SearchSite.searchSite	583
app/searchSite/searchSite.js		??
app/searchSite/views/form.php	Template form di ricerca nel sito	584
app/searchSite/views/results.php	Template risultati ricerca nel sito	584
app/statistics/class.LogAccess.php	Contiene la definizione ed implementazione della classe	585
app/statistics/class_statistics.php	Contiene la definizione ed implementazione della classe statistics	585
app/sysClass/class.ModuleApp.php	Contiene la definizione ed implementazione della classe Gino.App.SysClass.ModuleApp	586
app/sysClass/class_sysClass.php	Contiene la definizione ed implementazione della classe Gino.App.SysClass.sysClass	586
app/sysconf/class.Conf.php	Contiene la definizione ed implementazione della classe Gino.App.Sysconf.Conf	587
app/sysconf/class_sysconf.php	Contiene la definizione ed implementazione della classe Gino.App.Sysconf.sysconf	587
app/sysfunc/class_sysfunc.php	Contiene la definizione ed implementazione della classe Gino.App.Sysfunc.sysfunc	588
lib/const.php	Alcune costanti utilizzate dal sistema	628
lib/func.browser.php	Racchiude funzioni per il detect di browser, piattaforma etc..	629
lib/func.file.php	Racchiude le funzioni generali di manipolazione file e directory utilizzate da gino	630
lib/func.php	Racchiude funzioni generali utilizzate da gino	631
lib/func.var.php	Racchiude le librerie per il trattamento di stringhe e dei valori da e per un input/database	633
lib/global.php	Set di funzioni definite nel ROOT namespace	634
lib/recaptchalib.php	Contiene la libreria che gestisce la chiamata del reCaptcha	638
lib/sqlcode.php	Contiene la classe sqlcode	640
lib/classes/class.Access.php	Contiene la definizione ed implementazione della classe Gino.Access	591

lib/classes/class.AdminTable.php	
Contiene la definizione ed implementazione della classe Gino.AdminTable	592
lib/classes/class.Cache.php	
Contiene la definizione ed implementazione delle classi Gino.Cache , Gino.OutputCache , Gino.DataCache	592
lib/classes/class.Captcha.php	
Contiene la definizione ed implementazione della classe Gino.Captcha	593
lib/classes/class.Compressor.php	
Contiene la definizione ed implementazione della classe Gino.Compressor , Gino.JSMin e Gino.JSMinException	593
lib/classes/class.Core.php	
Contiene la definizione ed implementazione della classe Gino.Core	594
lib/classes/class.Css.php	
Contiene la definizione ed implementazione della classe Gino.Css	595
lib/classes/class.Db.php	
Contiene l'interfaccia Gino.DbManager e le classi Gino.Db e Gino.SqlParse	595
lib/classes/class.Document.php	
Contiene la definizione ed implementazione della class Gino.Document	596
lib/classes/class.Error.php	
Contiene la definizione ed implementazione della classe Gino.Error	596
lib/classes/class.EventDispatcher.php	
Contiene la definizione ed implementazione della classe Gino.EventDispatcher	597
lib/classes/class.Export.php	
Contiene la definizione ed implementazione della classe Gino.Export	597
lib/classes/class.Form.php	
Contiene la definizione ed implementazione della classe Gino.Form	598
lib/classes/class.Frontend.php	
Contiene la definizione ed implementazione della classe Gino.Frontend	598
lib/classes/class.GImage.php	
Contiene la definizione ed implementazione della classe Gino.GImage	599
lib/classes/class.GTag.php	
Contiene la definizione ed implementazione della classe Gino.GTag	600
lib/classes/class.Javascript.php	
Contiene la definizione ed implementazione della classe Gino.Javascript	601
lib/classes/class.Loader.php	
Contiene la definizione e l'implementazione della classe Gino.Loader ed il metodo magic <code>__autoload</code>	601
lib/classes/class.Locale.php	
Contiene la definizione ed implementazione della classe Gino.Locale	602
lib/classes/class.Logger.php	
Contiene la definizione ed implementazione della classe Gino.Logger	603
lib/classes/class.MobileDetect.php	
Contiene la definizione ed implementazione della classe Gino.MobileDetect	603
lib/classes/class.Options.php	
Contiene la definizione ed implementazione della classe Gino.Options	603
lib/classes/class.Paginator.php	
Contiene la definizione ed implementazione della classe Gino.Paginator	604
lib/classes/class.Registry.php	
Contiene la definizione ed implementazione della classe Gino.Registry	605
lib/classes/class.Router.php	
Contiene la definizione ed implementazione della class Gino.Router	605
lib/classes/class.Search.php	
Contiene la definizione ed implementazione della classe Gino.Search	606
lib/classes/class.Session.php	
Contiene la definizione ed implementazione della classe Gino.Session	606
lib/classes/class.Singleton.php	
Contiene la definizione ed implemetazione della classe Gino.Singleton	607

lib/classes/class.Skin.php	Contiene la definizione ed implementazione della classe Gino.Skin	607
lib/classes/class.Template.php	Contiene la definizione ed implementazione della classe Gino.Template	608
lib/classes/class.Translation.php	Contiene la definizione ed implementazione della classe Gino.Translation	608
lib/classes/build/class.BooleanBuild.php	Contiene la definizione ed implementazione della classe Gino.BooleanBuild	??
lib/classes/build/class.Build.php	Contiene la definizione ed implementazione della classe Gino.Build	??
lib/classes/build/class.CharBuild.php	Contiene la definizione ed implementazione della classe Gino.CharBuild	??
lib/classes/build/class.DateBuild.php	Contiene la definizione ed implementazione della classe Gino.DateBuild	??
lib/classes/build/class.DatetimeBuild.php	Contiene la definizione ed implementazione della classe Gino.DatetimeBuild	??
lib/classes/build/class.DirectoryBuild.php	Contiene la definizione ed implementazione della classe Gino.DirectoryBuild	??
lib/classes/build/class.EmailBuild.php	Contiene la definizione ed implementazione della classe Gino.EmailBuild	??
lib/classes/build/class.EnumBuild.php	Contiene la definizione ed implementazione della classe Gino.EnumBuild	??
lib/classes/build/class.FileBuild.php	Contiene la definizione ed implementazione della classe Gino.FileBuild	??
lib/classes/build/class.FloatBuild.php	Contiene la definizione ed implementazione della classe Gino.FloatField	??
lib/classes/build/class.ForeignKeyBuild.php	Contiene la definizione ed implementazione della classe Gino.ForeignKeyBuild	??
lib/classes/build/class.ImageBuild.php	Contiene la definizione ed implementazione della classe Gino.ImageBuild	??
lib/classes/build/class.IntegerBuild.php	Contiene la definizione ed implementazione della classe Gino.IntegerBuild	??
lib/classes/build/class.ManyToManyBuild.php	Contiene la definizione ed implementazione della classe Gino.ManyToManyField	??
lib/classes/build/class.ManyToManyThroughBuild.php	Contiene la definizione ed implementazione della classe Gino.ManyToManyThroughBuild	??
lib/classes/build/class.MulticheckBuild.php	Contiene la definizione ed implementazione della classe Gino.MulticheckBuild	??
lib/classes/build/class.SlugBuild.php	Contiene la definizione ed implementazione della classe Gino.SlugBuild	??
lib/classes/build/class.TagBuild.php	Contiene la definizione ed implementazione della classe Gino.TagBuild	??
lib/classes/build/class.TextBuild.php	Contiene la definizione ed implementazione della classe Gino.TextBuild	??
lib/classes/build/class.TimeBuild.php	Contiene la definizione ed implementazione della classe Gino.TimeBuild	??
lib/classes/build/class.YearBuild.php	Contiene la definizione ed implementazione della classe Gino.YearBuild	??
lib/classes/exceptions/class.Exception403.php	Contiene la definizione ed implementazione della classe Gino.Exception.Exception403	609
lib/classes/exceptions/class.Exception404.php	Contiene la definizione ed implementazione della classe Gino.Exception.Exception404	610
lib/classes/exceptions/class.Exception500.php	Contiene la definizione ed implementazione della classe Gino.Exception.Exception500	610
lib/classes/fields/class.BooleanField.php	Contiene la definizione ed implementazione della classe Gino.BooleanField	611
lib/classes/fields/class.CharField.php	Contiene la definizione ed implementazione della classe Gino.CharField	611

lib/classes/fields/class.DateField.php		
Contiene la definizione ed implementazione della classe	Gino.DateField	612
lib/classes/fields/class.DatetimeField.php		
Contiene la definizione ed implementazione della classe	Gino.DatetimeField	612
lib/classes/fields/class.DirectoryField.php		
Contiene la definizione ed implementazione della classe	Gino.DirectoryField	613
lib/classes/fields/class.EmailField.php		
Contiene la definizione ed implementazione della classe	Gino.EmailField	613
lib/classes/fields/class.EnumField.php		
Contiene la definizione ed implementazione della classe	Gino.EnumField	614
lib/classes/fields/class.Field.php		
Contiene la definizione ed implementazione della classe	Gino.Field	614
lib/classes/fields/class.FileField.php		
Contiene la definizione ed implementazione della classe	Gino.FileField	615
lib/classes/fields/class.FloatField.php		
Contiene la definizione ed implementazione della classe	Gino.FloatField	616
lib/classes/fields/class.ForeignKeyField.php		
Contiene la definizione ed implementazione della classe	Gino.ForeignKeyField	616
lib/classes/fields/class.ImageField.php		
Contiene la definizione ed implementazione della classe	Gino.ImageField	617
lib/classes/fields/class.IntegerField.php		
Contiene la definizione ed implementazione della classe	Gino.IntegerField	617
lib/classes/fields/class.ManyToManyField.php		
Contiene la definizione ed implementazione della classe	Gino.ManyToManyField	618
lib/classes/fields/class.ManyToManyThroughField.php		
Contiene la definizione ed implementazione della classe	Gino.ManyToManyThroughField	619
lib/classes/fields/class.MulticheckField.php		
Contiene la definizione ed implementazione della classe	Gino.MulticheckField	??
lib/classes/fields/class.SlugField.php		
Contiene la definizione ed implementazione della classe	Gino.SlugField	619
lib/classes/fields/class.TagField.php		
Contiene la definizione ed implementazione della classe	Gino.TagField	620
lib/classes/fields/class.TextField.php		
Contiene la definizione ed implementazione della classe	Gino.TextField	621
lib/classes/fields/class.TimeField.php		
Contiene la definizione ed implementazione della classe	Gino.TimeField	621
lib/classes/fields/class.YearField.php		
Contiene la definizione ed implementazione della classe	Gino.YearField	622
lib/classes/http/class.Redirect.php		
Contiene la definizione ed implementazione della classe	Gino.Http.Redirect	622
lib/classes/http/class.Request.php		
Contiene la definizione ed implementazione della classe	Gino.Http.Request	623
lib/classes/http/class.Response.php		
Contiene la definizione ed implementazione della classe	Gino.Http.Response	623
lib/classes/http/class.ResponseFile.php		
Contiene la definizione ed implementazione della classe	Gino.Http.ResponseFile	624
lib/classes/http/class.ResponseForbidden.php		
Contiene la definizione ed implementazione della classe	Gino.Http.ResponseForbidden	624
lib/classes/http/class.ResponseJson.php		
Contiene la definizione ed implementazione della classe	Gino.Http.ResponseJson	625
lib/classes/http/class.ResponseNotFound.php		
Contiene la definizione ed implementazione della classe	Gino.Http.ResponseNotFound	626
lib/classes/http/class.ResponsePdf.php		
Contiene la definizione ed implementazione della classe	Gino.Http.ResponsePdf	??
lib/classes/http/class.ResponseServerError.php		
Contiene la definizione ed implementazione della classe	Gino.Http.ResponseServerError	626
lib/classes/mvc/class.Controller.php		
Contiene la definizione ed implementazione della classe	Gino.Controller	627

lib/classes/mvc/class.Model.php		
Contiene la definizione ed implementazione della classe Gino.Model	627	
lib/classes/mvc/class.View.php		
Contiene la definizione ed implementazione della classe Gino.View	628	
lib/classes/widget/class.CheckboxWidget.php		
Contiene la definizione ed implementazione della classe Gino.CheckboxWidget	??	
lib/classes/widget/class.ConstantWidget.php		
Contiene la definizione ed implementazione della classe Gino.ConstantWidget	??	
lib/classes/widget/class.DatetimeWidget.php		
Contiene la definizione ed implementazione della classe Gino.DatetimeWidget	??	
lib/classes/widget/class.DateWidget.php		
Contiene la definizione ed implementazione della classe Gino.DateWidget	??	
lib/classes/widget/class.EditorWidget.php		
Contiene la definizione ed implementazione della classe Gino.EditorWidget	??	
lib/classes/widget/class.EmailWidget.php		
Contiene la definizione ed implementazione della classe Gino.EmailWidget	??	
lib/classes/widget/class.FileWidget.php		
Contiene la definizione ed implementazione della classe Gino.FileWidget	??	
lib/classes/widget/class.FloatWidget.php		
Contiene la definizione ed implementazione della classe Gino.FloatWidget	??	
lib/classes/widget/class.HiddenWidget.php		
Contiene la definizione ed implementazione della classe Gino.HiddenWidget	??	
lib/classes/widget/class.ImageWidget.php		
Contiene la definizione ed implementazione della classe Gino.ImageWidget	??	
lib/classes/widget/class.MulticheckWidget.php		
Contiene la definizione ed implementazione della classe Gino.MulticheckWidget	??	
lib/classes/widget/class.PasswordWidget.php		
Contiene la definizione ed implementazione della classe Gino.PasswordWidget	??	
lib/classes/widget/class.RadioWidget.php		
Contiene la definizione ed implementazione della classe Gino.RadioWidget	??	
lib/classes/widget/class.SelectWidget.php		
Contiene la definizione ed implementazione della classe Gino.SelectWidget	??	
lib/classes/widget/class.TextareaWidget.php		
Contiene la definizione ed implementazione della classe Gino.TextareaWidget	??	
lib/classes/widget/class.TextWidget.php		
Contiene la definizione ed implementazione della classe Gino.TextWidget	??	
lib/classes/widget/class.TimeWidget.php		
Contiene la definizione ed implementazione della classe Gino.TimeWidget	??	
lib/classes/widget/class.UnitWidget.php		
Contiene la definizione ed implementazione della classe Gino.UnitWidget	??	
lib/classes/widget/class.Widget.php		
Contiene la definizione ed implementazione della classe Gino.Widget	??	
lib/custom_ckeditor/config.js		??
lib/js/abiMap.js		??
lib/js/gino-min.js		??
lib/js/modernizr.js		??
lib/js/MooComplete.js		??
lib/js/mootools-1.4.0-yc.js		??
lib/js/respond.js		??
lib/js/slimbox.js		??
lib/js/CodeMirror/clike.js		??
lib/js/CodeMirror/codemirror.js		??
lib/js/CodeMirror/css.js		??
lib/js/CodeMirror/htmlmixed.js		??
lib/js/CodeMirror/javascript.js		??
lib/js/CodeMirror/matchbrackets.js		??
lib/js/CodeMirror/php.js		??

lib/js/CodeMirror/xml.js	??
lib/plugin/plugin.mysql.php	
Contiene la classe mysql	637
lib/plugin/plugin.pdo.php	
Contiene la classe pdo	??
lib/plugin/plugin.pdo_mysql.php	
Contiene la classe pdo_mysql	??
lib/plugin/plugin.pdo_sqlsrv.php	
Contiene la classe pdo_sqlsrv	??
lib/plugin/plugin.phpfastcache.php	
Contiene la classe plugin_phpfastcache	??
lib/plugin/plugin.sqlsrv.php	
Contiene la classe sqlsrv	637
templates/_popup.php	642
templates/admin_home.php	642
templates/admin_page.php	642
templates/default_free_tpl.php	643
templates/home.php	643
templates/home_mobile.php	643
templates/page.php	643
templates/pages_mobile.php	643
test/class.Paginator_test.php	
Contiene la definizione ed implementazione della classe Gino.Test.PaginatorTest	643
test/func_test.php	
Contiene la definizione ed implementazione della classe Gino.Test.FuncTest	643
test/include.php	
Include classi e funzioni per l'esecuzione di unit tests	644
views/403.php	
Template della pagina 403, Forbidden	644
views/404.php	
Template della pagina 404, Not Found	645
views/500.php	
Template della pagina 500, Internal Server Error	646
views/admin_table_form.php	
Template della pagina di inserimento/modifica record in area amministrativa	646
views/admin_table_list.php	
Template della lista record in area amministrativa	647
views/logger_stack_trace.php	
Template che stampa lo stack trace di una Exception	647
views/logger_system_variables.php	
Template che stampa un dump delle variabili di sistema (\$_SERVER, \$_SESSION, \$_REQUEST)	648
views/paginator_navigator.php	
Template dei controller per la navigazione delle pagine, paginazione	649
views/paginator_pagination.php	
Template completo paginazione, include navigazione pagine e sommario	649
views/related_contents_list.php	
Template che stampa la lista di contenuti correlati	650
views/section.php	
Template di una section	650
views/tab.php	
Template che stampa una tab html	651
views/table.php	
Template utilizzato per visualizzare le tabelle	652

Chapter 7

Namespace Documentation

7.1 Gino Namespace Reference

Namespace principale di gino.

Namespaces

- [App](#)
- [Exception](#)
Namespace che comprende tutte le classi di tipo [Exception](#)
- [Http](#)
Namespace che comprende tutte le classi per la gestione di richieste e risposte http
- [Plugin](#)
Namespace che comprende classi di tipo plugin
- [Test](#)
Namespace che comprende gli UNIT TEST di classi e funzioni di gino

Data Structures

- class [Access](#)
Classe per la gestione dell'autenticazione ed accesso alla funzionalità
- class [AdminTable](#)
Gestisce l'interfaccia di amministrazione di un modello con inserimento, modifica ed eliminazione.
- class [BooleanBuild](#)
Gestisce i campi di tipo BOOLEAN.
- class [BooleanField](#)
Rappresenta campi di tipo BOOLEAN.
- class [Build](#)
Gestisce i campi del modello.
- class [Cache](#)
Classe che viene estesa da [Gino.OutputCache\(\)](#) e [Gino.DataCache\(\)](#)
- class [Captcha](#)
Classe per la generazione di immagini captcha.
- class [CharBuild](#)
Gestisce i campi di tipo stringa (CHAR, VARCHAR)
- class [CharField](#)
Campi di tipo stringa (CHAR, VARCHAR)

- class [CheckboxWidget](#)
Campi checkbox nei form.
- class [Compressor](#)
Classe per la compressione di css e js (merge e minify)
- class [ConstantWidget](#)
Campi di tipo costante nei form.
- class [Controller](#)
Classe astratta primitiva di tipo [Controller](#) (MVC), dalla quale tutti i controller delle singole app discendono.
- class [Core](#)
Gestisce una [Gino.Http.Request](#) ed invia una [Gino.Http.Response](#) adeguata.
- class [Css](#)
Libreria per la gestione dei file css dei singoli moduli e dei file css del layout (da associare alle skin)
- class [DataCache](#)
Memorizza le strutture dati scrivendo su file.
- class [DateBuild](#)
Gestisce i campi di tipo data.
- class [DateField](#)
Campo di tipo data.
- class [DatetimeBuild](#)
Gestisce i campi di tipo DATETIME.
- class [DatetimeField](#)
Campo di tipo DATETIME.
- class [DatetimeWidget](#)
Campi di tipo datetime nei form.
- class [DateWidget](#)
Campi di tipo date nei form.
- class [Db](#)
Classe Factory e [Singleton](#) usata per creare oggetti che si interfacciano al database.
- interface [DbManager](#)
Interfaccia per le librerie di connessione al database.
- class [DirectoryBuild](#)
Gestisce i campi di tipo DIRECTORY.
- class [DirectoryField](#)
Campo di tipo DIRECTORY.
- class [Document](#)
Crea il documento html da inviare come corpo della risposta HTTP.
- class [EditorWidget](#)
Campi di tipo editor nei form.
- class [EmailBuild](#)
Campo di tipo EMAIL.
- class [EmailField](#)
Campo di tipo EMAIL.
- class [EmailWidget](#)
Campi di tipo email nei form.
- class [EnumBuild](#)
Gestisce i campi di tipo ENUM.
- class [EnumField](#)
Campo di tipo ENUM.
- class [Error](#)
Classe per la gestione di errori da mostrare all'utente E' possibile anche mostrare dei warning per facilitare il debug.
- class [EventDispatcher](#)

- Classe di tipo Mediator e [Singleton](#) per la gestione di segnali (eventi) tra oggetti.*
- class [Export](#)
Libreria per l'esportazione di tabelle o dati.
 - class [Field](#)
Gestisce le caratteristiche del tipo di campo (colonne)
 - class [FileBuild](#)
Campo di tipo FILE.
 - class [FileField](#)
Campo di tipo FILE.
 - class [FileWidget](#)
Campi di tipo file nei form.
 - class [FloatBuild](#)
Gestisce campi di tipo decimale (FLOAT, DOUBLE, DECIMAL)
 - class [FloatField](#)
Campo di tipo decimale (FLOAT, DOUBLE, DECIMAL)
 - class [FloatWidget](#)
Campi di tipo float nei form.
 - class [ForeignKeyBuild](#)
Campo di tipo chiave esterna.
 - class [ForeignKeyField](#)
Campo di tipo chiave esterna.
 - class [Form](#)
Classe per la creazione ed il salvataggio dati di un form.
 - class [Frontend](#)
Libreria per la gestione dei file di front-end dei singoli moduli (css e viste)
 - class [GImage](#)
Classe per il trattamento di immagini.
 - class [GTag](#)
Classe per il trattamento di campi di tipo tag.
 - class [HiddenWidget](#)
Campi di tipo hidden nei form.
 - class [ImageBuild](#)
Gestisce campi di tipo IMMAGINE.
 - class [ImageField](#)
Campo di tipo IMMAGINE.
 - class [ImageWidget](#)
Campi di tipo immagine nei form.
 - class [IntegerBuild](#)
Campo di tipo INTERO.
 - class [IntegerField](#)
Campo di tipo INTERO.
 - class [Javascript](#)
Contiene i metodi per includere alcuni javascript.
 - class [JSMIn](#)
PHP implementation of Douglas Crockford's [JSMIn](#).
 - class [JSMInException](#)
[JSMInException](#).
 - class [Loader](#)
[Loader](#) di classi di tipo [Gino.Model](#) e classi di sistema.
 - class [Locale](#)
Libreria per la gestione delle traduzioni che non utilizzano le librerie gettext.

- class [Logger](#)
Classe per la notifica di log di sistema.
- class [ManyToManyBuild](#)
Gestisce i campi di tipo many to many.
- class [ManyToManyField](#)
Campo di tipo many to many.
- class [ManyToManyThroughBuild](#)
Gestisce i campi di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.
- class [ManyToManyThroughField](#)
Campo di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.
- class [MobileDetect](#)
Verifica se una richiesta HTTP è stata effettuata da un dispositivo mobile.
- class [Model](#)
Classe astratta che definisce un modello, cioè un oggetto che rappresenta una tabella su database.
- class [MulticheckBuild](#)
Gestisce i campi di tipo multicheck.
- class [MulticheckField](#)
Campo di tipo multicheck.
- class [MulticheckWidget](#)
Campi di tipo multicheck nei form.
- class [Options](#)
Gestisce le opzioni di classe, costruendo il form ed effettuando l'action.
- class [OutputCache](#)
Memorizza gli output (text, html, xml) scrivendo su file.
- class [Paginator](#)
Gestisce la paginazione di elementi dati il numero di elementi totali ed il numero di elementi per pagina, ricava i limiti per creare il sottoinsieme di elementi da mostrare e gestisce la navigazione tra le pagine.
- class [PasswordWidget](#)
Campi di tipo password nei form.
- class [RadioWidget](#)
Campi di tipo radio button nei form.
- class [Registry](#)
Registro di gino.
- class [Router](#)
Gestisce il routing di una request HTTP, chiamando la classe e metodo che devono fornire risposta.
- class [Search](#)
Libreria per ricerche full text pesate sulle tabelle.
- class [SelectWidget](#)
Campi di tipo select nei form.
- class [Session](#)
Classe per la gestione delle variabili di sessione.
- class [Singleton](#)
Classe astratta per implementazione del [Singleton](#) pattern.
- class [Skin](#)
Libreria per la gestione delle [Skin](#).
- class [SlugBuild](#)
Getisce i campi di tipo SLUG (CHAR, VARCHAR)
- class [SlugField](#)
Campo di tipo SLUG (CHAR, VARCHAR)
- class [sqlcode](#)
Contiene query e codice sql personalizzato.

- class [SqlParse](#)
Classe per gestire il parser dei file sql (Code freely adapted from phpBB Group)
- class [TagBuild](#)
Gestisce i campi per inserimento tag.
- class [TagField](#)
Classe per la gestione di campi per inserimento tag.
- class [Template](#)
Libreria per la gestione dei template del documento html da associare alle [Gino](#).
- class [TextareaWidget](#)
Campi di tipo textarea nei form.
- class [TextBuild](#)
Campo di tipo TEXT.
- class [TextField](#)
Campo di tipo TEXT.
- class [TextWidget](#)
Campi di tipo testo nei form.
- class [TimeBuild](#)
Gestisce i campi di tipo TIME.
- class [TimeField](#)
Campo di tipo TIME.
- class [TimeWidget](#)
Campi di tipo orario nei form.
- class [Translation](#)
Classe per la gestione delle traduzioni.
- class [UnitWidget](#)
Insieme di campi di un modello.
- class [View](#)
Gestisce le viste, impostando il template e ritornando l'output.
- class [Widget](#)
Gestisce i campi nei form.
- class [YearBuild](#)
Gestisce i campi di tipo ANNO.
- class [YearField](#)
Campo di tipo ANNO.

Functions

- [browsers](#) ()
Lista di browser popolari.
- [robots](#) ()
Lista di web robots popolari.
- [platforms](#) ()
Lista di piattaforme popolari.
- [get_browser_info](#) (\$arg=", \$agent=")
Informazioni browser.
- [is_browser](#) (\$name)
Verifica se il browser è quello dato come argomento.
- [is_browser_version](#) (\$version)
Verifica la versione del browser.
- [is_browser_platform](#) (\$platform)

- Verifica la piattaforma del browser.*

 - [is_robot](#) ()

Verifica se il client è un robot.
- [searchNameFile](#) (\$dir)

Lista files contenuti in una directory.
- [download](#) (\$full_path)

Forza il download di un file.
- [extension](#) (\$filename, \$extensions)

Controlla le estensioni dei file.
- [baseFileName](#) (\$filename)

Ricava il nome del file senza l'estensione.
- [deleteFileDir](#) (\$dir, \$delete_dir=TRUE)

Elimina ricorsivamente i file e le directory.
- [extensionFile](#) (\$filename)

Estensione di un file.
- [verifyExtension](#) (\$filename, \$extensions)

Controlla se l'estensione di un file è valida.
- [relativePath](#) (\$abspath)

Ricava il percorso relativo a partire da un percorso assoluto.
- [absolutePath](#) (\$relpath)

Ricava il percorso assoluto a partire da un percorso relativo.
- [gOpt](#) (\$opt_name, \$opt_array, \$default)

Restituisce l'elemento di un array corrispondente alla chiave data oppure un valore di default.
- [arrayToObject](#) (array \$array)

Trasforma un array in un oggetto.
- [checkEmail](#) (\$value, \$regexp=null)

Verifica la validità dell'indirizzo email.
- [dateToDbDate](#) (\$date, \$s='/')

Formatta la data per il database (YYYY-MM-DD)
- [dbDateToDate](#) (\$db_date, \$s='/', \$num_year=4)

Converte il formato della data da database (campo DATE) in un formato di facile visualizzazione (DD/MM/YYYY)
- [dbDatetimeToDate](#) (\$datetime, \$s='/', \$num_year=4)

Converte il formato della data da database (campo DATETIME) in un formato di facile visualizzazione (DD/MM/Y↔YYY)
- [dbDatetimeToTime](#) (\$datetime)

Riporta l'orario di un campo DATETIME (HH:MM:SS)
- [dbTimeToTime](#) (\$db_time, \$seconds=false)

Mostra l'orario (HH:MM[:SS])
- [timeToDbTime](#) (\$time)

Formatta l'orario per il database (HH:MM:SS)
- [dbNumberToNumber](#) (\$number, \$decimals=2)

Formatta un numero con il raggruppamento delle centinaia.
- [numberToDB](#) (\$number)

Formatta un numero per il database (il separatore decimale è il punto)
- [isNumeric](#) (\$variable)

Controlla se una variabile è un numero o una stringa numerica.
- [timeDiff](#) (\$firstTime, \$lastTime)

Calcola l'intervallo di tempo in secondi tra due valori datetime.
- [dateDiff](#) (\$interval, \$datefrom, \$dateto, \$using_timestamps=false)

Calcola la differenza di tempo tra due datetime in più formati.
- [getDateDiff](#) (\$start_date, \$end_date=null, \$options=array())

- Calcola la differenza di tempo tra due datetime in più formati.*

 - **isValid** (\$type, \$var)

Verifica se il valore della variabile è conforme al tipo di controllo indicato.
- **cutHtmlText** (\$html, \$length, \$ending, \$strip_tags, \$cut_words, \$cut_images, \$options=null)

Accorcia un testo HTML alla lunghezza desiderata (length)
- **cutString** (\$string, \$max_char, \$word_complete=true, \$file=false)

Limita i caratteri di una stringa.
- **listProv** ()

Elenco delle province.
- **share** (\$site, \$url, \$title=null, \$description=null)

Codice html di condivisione social network.
- **shareAll** (\$social, \$url, \$title=null, \$description=null)

Codice condivisione social network con integrazione sharethis.
- **traslitterazione** (\$numero, \$decimale=false)

Converte un numero in cifre.
- **convertSize** (\$size)

Converte la dimensione di un file in un formato leggibile (KB,GB,...)
- **enabledPng** ()

Verifica la validità del supporto PNG.
- **enabledZip** ()

Verifica la validità della classe ZipArchive.
- **cryptMethod** (\$string, \$crypt)

Crypta la password dell'utente.
- **emailPolicy** ()

Testo della policy di una email.
- **obj_serialize** (\$instanceName, \$object)

Operazione di serializzazione.
- **obj_unserialize** (\$instanceName)

Operazione di deserializzazione.
- **icon** (\$name, \$options=array())

Icone.
- **convertToHtml** (\$value)

Converte l'encoding di un valore preso da un campo di un database (non UTF-8) nella codifica UTF-8.
- **convertToDatabase** (\$value, \$character_set=null)

Converte l'encoding di un valore da html (es UTF-8) a un encoding valido per il database.
- **replaceChar** (\$text)

Sostituzione caratteri "strani" di word.
- **strip_tags_attributes** (\$text, \$strip_js, \$strip_attributes)

Rimuove gli attributi javascript insicuri nei tag html.
- **remove_attributes** (\$text)

Rimuove l'attributo class quando vuoto.
- **strip_selected_tags** (\$text, \$tags="", \$stripContent=false)

Rimuove i tag indicati.
- **strip_invisible_tags** (\$text)

Rimuove script, stili e altro testo indesiderato (e invisibile) tra i tag.
- **stripAll** (\$text)

Sostituzione entities con il carattere corrispondente.
- **stripEditor** (\$text)

Sostituisce alcuni tag utilizzati dall'editor html con i tag standard.
- **cleanVar** (\$method, \$name, \$type, \$strip_tags= "", \$options=array())

Modifica il valore presente in un campo del form per inserirlo nel database.

- **clean_sequence** (\$text, \$strip_tags, \$options)
Esegue una serie di operazioni sul testo per renderlo compatibile con gino e col database.
- **cleanVarEditor** (\$method, \$name, \$strip_tags)
Modifica il valore presente in un campo del form di tipo editor html per inserirlo nel database.
- **utf8_urldecode** (\$str)
Conversione dei dati Unicode \$_GET/\$_POST, generati dalla funzione javascript escape(), in UTF8 per il processo server-side.
- **codeDb** (\$text)
Prepara il testo da inserire nel database senza rimuovere il codice html.
- **codeToDB** (\$method, \$name, \$options=array())
Modifica il valore presente in un campo testo del form per inserirlo nel database.
- **htmlChars** (\$string, \$id="", \$options=array())
Modifica il valore di un campo testo del database per visualizzarlo in HTML.
- **preCodeParser** (\$string)
Modifica il valore di un campo testo di tipo "codice" del database per visualizzarlo in HTML.
- **codeParser** (\$string, \$id="")
Modifica il valore di un campo testo del database per visualizzarlo in HTML.
- **slimboxReplace** (\$string, \$id)
Modifica il valore di un campo testo del database per attivare le librerie slimbox.
- **htmlCharsText** (\$string)
Mostra il valore di un campo testo del database formattato come "solo testo".
- **htmlInput** (\$string)
Modifica il valore di un campo testo del database per visualizzarlo in un input form.
- **htmlInputEditor** (\$string)
Modifica il valore di un campo testo del database per visualizzarlo in un input form di tipo editor html.
- **codeInput** (\$string)
Modifica il valore di un campo testo del database per visualizzarlo in un input form di tipo "solo testo".
- **enclosedField** (\$string)
Racchiude il testo tra virgolette singole.
- **jsVar** (\$string, \$newline=false)
Escape testo che deve essere passato come variabile javascript.
- **attributeVar** (\$string)
Escape testo che deve essere racchiuso in attributi html.
- **htmlToEntities** (\$string)
Converte le entities HTML, ma non i tag.
- **encode_params** (\$params)
Codifica i parametri url.
- **decode_params** (\$params)
Decodifica i parametri url.

Variables

- const **SITE_ROOT** dirname(realpath(__FILE__))
SITE_ROOT Percorso assoluto alla root directory
- const **SITE_WWW** \$site_www
SITE_WWW Percorso relativo dell'applicazione a partire dalla root directory
- const **GIMAGE_DIR** CONTENT_DIR.OS.'gimage'
- const **TBL_MODULE** 'sys_module'
Nome della tabella che contiene le istanze di moduli.
- const **TBL_MODULE_APP** 'sys_module_app'
Nome della tabella che contiene i moduli installati.

- const [TBL_SYS_CONF](#) 'sys_conf'
Nome della tabella che contiene la configurazione del sistema.
- const [TBL_LOG_ACCESS](#) 'sys_log_access'
Nome della tabella che contiene il log degli accessi.
- const [TBL_USER](#) 'auth_user'
Nome della tabella utenti.
- const [TBL_USER_ADD](#) 'auth_user_add'
Nome della tabella informazioni aggiuntive utenti.
- const [TBL_USER_GROUP](#) 'auth_user_group'
Nome della tabella di join utenti - gruppi.
- const [TBL_GROUP](#) 'auth_group'
Nome della tabella dei gruppi di utenti.
- const [TBL_GROUP_PERMISSION](#) 'auth_group_perm'
Nome della tabella di join gruppi - permessi.
- const [TBL_PERMISSION](#) 'auth_permission'
Nome della tabella dei permessi.
- const [TBL_USER_PERMISSION](#) 'auth_user_perm'
Nome della tabella di join utenti - permessi.
- const [TBL_REGISTRATION_PROFILE](#) 'auth_registration_profile'
Nome della tabella dei profili di registrazione utenti.
- const [TBL_REGISTRATION_PROFILE_GROUP](#) 'auth_registration_profile_group'
Nome della tabella di associazione profili di registrazione utenti - gruppi.
- const [TBL_REGISTRATION_REQUEST](#) 'auth_registration_request'
Nome della tabella delle richieste di registrazione.
- const [TBL_LANGUAGE](#) 'language'
Nome della tabella lingue.
- const [TBL_TRANSLATION](#) 'language_translation'
Nome della tabella traduzioni.
- const [TBL_NATION](#) 'nation'
Nome della tabella nazioni.
- const [OS](#) DIRECTORY_SEPARATOR
Separatore di directory definito dal sistema operativo.
- const [CACHE_DIR](#) SITE_ROOT.OS.'cache'
Percorso assoluto alla directory cache.
- const [LIB_DIR](#) SITE_ROOT.OS.'lib'
Percorso assoluto alla directory lib.
- const [CLASSES_DIR](#) LIB_DIR.OS.'classes'
Percorso assoluto alla directory che contiene le classi che non fanno parte di moduli.
- const [FIELDS_DIR](#) CLASSES_DIR.OS.'fields'
Percorso assoluto alla directory che contiene le classi che descrivono campi di db.
- const [APP_DIR](#) SITE_ROOT.OS.'app'
Percorso assoluto alla directory che contiene le directory dei moduli.
- const [PLUGIN_DIR](#) LIB_DIR.OS.'plugin'
Percorso assoluto alla directory che contiene plugins.
- const [CSS_DIR](#) SITE_ROOT.OS.'css'
Percorso assoluto alla directory che contiene css.
- const [TPL_DIR](#) SITE_ROOT.OS.'templates'
Percorso assoluto alla directory che contiene templates.
- const [VIEWS_DIR](#) SITE_ROOT.OS.'views'
Percorso assoluto alla directory che contiene viste generiche di sistema.
- const [GRAPHICS_DIR](#) SITE_ROOT.OS.'graphics'

- Percorso assoluto alla directory che contiene file di grafica.*
- const [CONTENT_DIR](#) SITE_ROOT.OS.'contents'
- Percorso assoluto alla directory che contiene upload degli utenti.*
- const [FONTS_DIR](#) SITE_ROOT.OS.'fonts'
- Percorso assoluto alla directory che contiene fonts files.*
- const [TMP_DIR](#) '/tmp'
- Percorso assoluto alla directory per la creazione di file temporanei.*
- const [HOME_FILE](#) SITE_WWW.'/\$home_file'
- Percorso relativo dello SCRIPT_FILE.*
- const [CSS_WWW](#) SITE_WWW.'/css'
- Percorso relativo alla directory dei css.*
- const [SITE_APP](#) SITE_WWW.'/app'
- Percorso relativo alla directory che contiene i moduli.*
- const [SITE_IMG](#) SITE_WWW.'/img'
- Percorso relativo alla directory che contiene immagini si sistema.*
- const [SITE_GRAPHICS](#) SITE_WWW.'/graphics'
- Percorso relativo alla directory che contiene file di grafica.*
- const [SITE_LIB](#) SITE_WWW.'/lib'
- Percorso relativo alla directory lib.*
- const [SITE_JS](#) SITE_LIB.'/js'
- Percorso relativo alla directory che contiene librerie javascript.*
- const [SITE_CUSTOM_CKEDITOR](#) SITE_LIB.'/custom_ckeditor'
- Percorso relativo alla directory che contiene file di customizzazione ckeditor.*
- const [CONTENT_WWW](#) SITE_WWW.'/contents'
- Percorso relativo alla directory che contiene file uploadati dall'utente.*
- const [CONTROLLER_CLASS_PREFIX](#) 'class_'
- Prefisso classi di tipo [Gino.Controller](#).*
- const [MODEL_CLASS_PREFIX](#) 'class.'
- Prefisso classi di tipo [Gino.Model](#).*
- const [CORE_CLASS_PREFIX](#) 'class.'
- Prefisso classi non interne a moduli.*

7.1.1 Detailed Description

Namespace principale di gino.

Contiene tutte le funzioni e classi proprie di gino al suo interno o in uno dei suoi sotto namespaces.

7.1.2 Function Documentation

7.1.2.1 [Gino\absolutePath \(\\$relpath \)](#)

Ricava il percorso assoluto a partire da un percorso relativo.

Parameters

string	<i>\$relpath</i>	percorso relativo
--------	------------------	-------------------

Returns

percorso assoluto

7.1.2.2 Gino\arrayToObject (array \$array)

Trasforma un array in un oggetto.

Parameters

array	<i>\$array</i>	
-------	----------------	--

Returns

object

7.1.2.3 Gino\attributeVar (*\$string*)

Escape testo che deve essere racchiuso in attributi html.

Parameters

string	<i>\$string</i>	
--------	-----------------	--

Returns

testo escaped

7.1.2.4 Gino\baseFileName (*\$filename*)

Ricava il nome del file senza l'estensione.

Parameters

string	<i>\$filename</i>	nome del file
--------	-------------------	---------------

Returns

nome file senza estensione

7.1.2.5 Gino\browsers ()

Lista di browser popolari.

Returns

array

7.1.2.6 Gino\checkEmail (*\$value*, *\$regexp* = null)

Verifica la validità dell'indirizzo email.

Di default verifica la corrispondenza dell'indirizzo email alle specifiche dello standard RFC-2822.

Parameters

string	<i>\$value</i>	indirizzo email
mixed	<i>\$regexp</i>	se presente verifica la corrispondenza di un indirizzo con una espressione regolare <ul style="list-style-type: none">• tipo boolean, se vero attiva una espressione regolare restrittiva• tipo string, espressione regolare personalizzata

Returns

boolean

7.1.2.7 Gino\clean_sequence (*\$text*, *\$strip_tags*, *\$options*)

Esegue una serie di operazioni sul testo per renderlo compatibile con gino e col database.

Parameters

string	<i>\$text</i>	testo
string	<i>\$strip_tags</i>	stringa con i tag da rimuovere, ad esempio "<a><p><quote>"
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • escape (boolean)

Returns

testo ripulito

7.1.2.8 Gino\cleanVar (*\$method*, *\$name*, *\$type*, *\$strip_tags* = ' ', *\$options* = array())

Modifica il valore presente in un campo del form per inserirlo nel database.

Imposta il tipo del testo

See also

[clean_sequence\(\)](#)

Parameters

string	<i>\$method</i>	metodo utilizzato (GET, POST, REQUEST)
string	<i>\$name</i>	nome della variabile
string	<i>\$type</i>	tipo di variabile (bool,int,float,string,array,object,null)
string	<i>\$strip_tags</i>	stringa con i tag da rimuovere, ad esempio "<a><p><quote>"
array	<i>\$options</i>	array associativo di opzioni

Returns

testo ripulito

7.1.2.9 Gino\cleanVarEditor (*\$method*, *\$name*, *\$strip_tags*)

Modifica il valore presente in un campo del form di tipo editor html per inserirlo nel database.

Parameters

string	<i>\$method</i>	metodo utilizzato (GET, POST, REQUEST)
string	<i>\$name</i>	nome della variabile
string	<i>\$strip_tags</i>	stringa con i tag da rimuovere, ad esempio "<a><p><quote>"

Returns

testo ripulito

7.1.2.10 GinoIcodeDb (*\$text*)

Prepara il testo da inserire nel database senza rimuovere il codice html.

Parameters

string	<i>\$text</i>	testo
--------	---------------	-------

Returns

testo

7.1.2.11 Gino\codeInput (*\$string*)

Modifica il valore di un campo testo del database per visualizzarlo in un input form di tipo "solo testo".

Parameters

string	<i>\$string</i>	
--------	-----------------	--

Returns

testo modificato

7.1.2.12 Gino\codeParser (*\$string*, *\$id* = ' ')

Modifica il valore di un campo testo del database per visualizzarlo in HTML.

Si tratta di testo di tipo "codice", ovvero inserito con la funzione [codeToDB\(\)](#). Crea un blocco che racchiude il codice e ne evidenzia le righe.

See also

[codeToDB\(\)](#)
[preCodeParser\(\)](#)

Parameters

string	<i>\$string</i>	
string	<i>\$id</i>	codice che raggruppa un insieme di immagini da visualizzare con le librerie slimbox

Returns

testo modificato

7.1.2.13 Gino\codeToDB (*\$method*, *\$name*, *\$options* = array())

Modifica il valore presente in un campo testo del form per inserirlo nel database.

Si tratta di testo di tipo "codice", nel quale non viene rimosso il codice html

See also

[codeDB\(\)](#)

Parameters

string	<i>\$method</i>	metodo utilizzato (GET, POST, REQUEST)
string	<i>\$name</i>	nome della variabile
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • width (string): sovrascrive la larghezza di visualizzazione di una immagine • height (string): sovrascrive l'altezza di visualizzazione di una immagine

Returns

testo modificato

7.1.2.14 Gino\convertSize (*\$size*)

Converte la dimensione di un file in un formato leggibile (KB,GB,...)

Parameters

integer	<i>\$size</i>	
---------	---------------	--

Returns

string

7.1.2.15 Gino\convertToDatabase (*\$value*, *\$character_set* = null)

Converte l'encoding di un valore da html (es UTF-8) a un encoding valido per il database.

Parameters

string	<i>\$value</i>	valore da convertire
string	<i>\$character_set</i>	set di caratteri del database <ul style="list-style-type: none"> • <i>CP1252</i>, per SQL Server

Returns

valore convertito

7.1.2.16 Gino\convertToHtml (*\$value*)

Converte l'encoding di un valore preso da un campo di un database (non UTF-8) nella codifica UTF-8.

Parameters

string	<i>\$value</i>	valore da convertire
--------	----------------	----------------------

Returns

valore convertito

7.1.2.17 Gino\cryptMethod (*\$string*, *\$crypt*)

Crypta la password dell'utente.

Parameters

string	<i>\$string</i>	
string	<i>\$crypt</i>	metodo di criptazione; default: proprietà <i>_crypt</i> (impostazioni di sistema)

Returns

password criptata

7.1.2.18 Gino\cutHtmlText (*\$html*, *\$length*, *\$ending*, *\$strip_tags*, *\$cut_words*, *\$cut_images*, *\$options* = null)

Accorcia un testo HTML alla lunghezza desiderata (*length*)

Sostituisce l'ultimo carattere con il valore *ending* se il testo è più lungo di *length*. Può strappare i TAG.

Parameters

string	<i>\$html</i>	stringa HTML da accorciare
integer	<i>\$length</i>	lunghezza della stringa da riportare, incluse le ellissi
string	<i>\$ending</i>	finale da aggiungere alla stringa accorciata
boolean	<i>\$strip_tags</i>	se vero, i TAG HTML saranno sostituiti da niente
boolean	<i>\$cut_words</i>	se falso, l'ultima parola della stringa non sarà tagliata
boolean	<i>\$cut_images</i>	se vero, la stringa non conterrà immagini
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • endingPosition (string) [in out]: posizionamento dei caratteri dell'<i>ending</i> nella struttura html o al di fuori della struttura html (dopo che sono stati chiusi tutti i TAG)

Returns

stringa accorciata

7.1.2.19 Gino\cutString (*\$string*, *\$max_char*, *\$word_complete* = true, *\$file* = false)

Limita i caratteri di una stringa.

Parameters

string	<i>\$string</i>	testo da accorciare
integer	<i>\$max_char</i>	numero massimo di caratteri
boolean	<i>\$word_complete</i>	se vero, mantiene l'ultima parola completa (utile nei select)
boolean	<i>\$file</i>	se vero, mostra l'estensione finale del file

Returns

stringa ridotta

7.1.2.20 Gino\dateDiff (*\$interval*, *\$datefrom*, *\$dateto*, *\$using_timestamps* = false)

Calcola la differenza di tempo tra due datetime in più formati.

Parameters

string	<i>\$interval</i>	indica il tipo di numero da ricavare, accetta i valori: <ul style="list-style-type: none"> • yyyy - numero totale di anni • q - numero totale di quarti • m - numero totale di mesi • y - differenza tra numero di giorni (ad es. 1st Jan 2004 è "1", il primo giorno. 2nd Feb 2003 è "33". La differenza è "-32".) • d - numero totale di giorni • w - numero totale di giorni della settimana • ww - numero totale di settimane • h - numero totale di ore • n - numero totale di minuti • s - numero totale di secondi (default)
string	<i>\$datefrom</i>	datetime iniziale
string	<i>\$dateto</i>	datetime finale
boolean	<i>\$using_↔ timestamps</i>	indica se i valori di \$datefrom e \$dateto sono in formato timestamp (default false)

Returns

differenza di tempo

7.1.2.21 Gino\dateToDbDate (*\$date*, *\$s* = ' / ')

Formatta la data per il database (YYYY-MM-DD)

Parameters

string	<i>\$date</i>	valore della data (DD/MM/YYYY), generalmente da input form
string	<i>\$s</i>	separatore utilizzato nella data

Returns

string

7.1.2.22 Gino\dbDatetimeToDate (*\$datetime*, *\$s* = ' / ', *\$num_year* = 4)

Converte il formato della data da database (campo DATETIME) in un formato di facile visualizzazione (DD/MM[↔YYYY])

Parameters

mixed	<i>\$datetime</i>	valore del campo datetime; string (YYYY-MM-DD HH:MM:SS) or object(Date↔Time)
string	<i>\$s</i>	separatore utilizzato nella data
integer	<i>\$num_year</i>	numero di cifre dell'anno da mostrare

Returns

string

7.1.2.23 Gino\dbDatetimeToTime (*\$datetime*)

Riporta l'orario di un campo DATETIME (HH:MM:SS)

Parameters

mixed	<i>\$datetime</i>	valore del campo datetime; string (YYYY-MM-DD HH:MM:SS) or object(DateTime)
-------	-------------------	---

Returns

string

7.1.2.24 Gino\dbDateToDate (*\$db_date*, *\$s* = ' / ', *\$num_year* = 4)

Converte il formato della data da database (campo DATE) in un formato di facile visualizzazione (DD/MM[/YYYY])

Parameters

mixed	<i>\$db_date</i>	valore del campo date; string (YYYY-MM-DD) or object(DateTime)
string	<i>\$s</i>	separatore utilizzato nella data
integer	<i>\$num_year</i>	numero di cifre dell'anno da mostrare

Returns

string

7.1.2.25 Gino\dbNumberToNumber (*\$number*, *\$decimals* = 2)

Formatta un numero con il raggruppamento delle centinaia.

```
1 number_format(float $number, int $decimals = 0, string $dec_point = ',', string $thousands_sep = '.')
```

Parameters

float	<i>\$number</i>	numero
int	<i>\$decimals</i>	numero di decimali

Returns

numero formattato

7.1.2.26 Gino\dbTimeToTime (*\$db_time*, *\$seconds* = false)

Mostra l'orario (HH:MM[:SS])

Parameters

mixed	<i>\$db_time</i>	valore del campo time o dell'output della funzione dbDatetimeToTime; string (HH:MM:SS) or object(DateTime)
boolean	<i>\$seconds</i>	visualizzazione dei secondi

Returns

string

7.1.2.27 Gino\decode_params (*\$params*)

Decodifica i parametri url.

Parameters

string	<i>\$params</i>	parametri url
--------	-----------------	---------------

Returns

stringa decodificata

7.1.2.28 Gino\deleteFileDir (*\$dir*, *\$delete_dir* = TRUE)

Elimina ricorsivamente i file e le directory.

Parameters

string	<i>\$dir</i>	percorso assoluto alla directory
boolean	<i>\$delete_dir</i>	per eliminare anche le directory (default true)

Returns

true

7.1.2.29 Gino\download (*\$full_path*)

Forza il download di un file.

Parameters

string	<i>\$full_path</i>	percorso del file
--------	--------------------	-------------------

Returns

stream del file o FALSE se il file non si apre in lettura

7.1.2.30 Gino\emailPolicy ()

Testo della policy di una email.

Returns

policy

7.1.2.31 Gino\enabledPng ()

Verifica la validità del supporto PNG.

Returns

TRUE se supportato, FALSE altrimenti

7.1.2.32 GinoEnabledZip ()

Verifica la validità della classe *ZipArchive*.

Returns

TRUE se abilitato, FALSE altrimenti

7.1.2.33 GinoEnclosedField (*\$string*)

Racchiude il testo tra virgolette singole.

La funzione viene utilizzata ad esempio per racchiudere i campi nelle email e nelle esportazioni di file

Parameters

string	<i>\$string</i>	
--------	-----------------	--

Returns

testo racchiuso tra virgolette

7.1.2.34 GinoEncodeParams (*\$params*)

Codifica i parametri url.

Parameters

string	<i>\$params</i>	parametri url
--------	-----------------	---------------

Returns

stringa codificata

7.1.2.35 GinoExtension (*\$filename*, *\$extensions*)

Controlla le estensioni dei file.

Verifica se il file ha una estensione valida, ovvero presente nell'elenco delle estensioni.

Parameters

string	<i>\$filename</i>	nome del file
array	<i>\$extensions</i>	elenco delle estensioni valide

Returns

TRUE se *\$extension* è vuoto o se il file ha un'estensione valida

7.1.2.36 GinoExtensionFile (*\$filename*)

Estensione di un file.

Parameters

string	<i>\$filename</i>	nome del file
--------	-------------------	---------------

Returns

estensione

7.1.2.37 Gino\get_browser_info (*\$arg* = ' ', *\$agent* = ' ')

Informazioni browser.

Parameters

string	<i>\$arg</i>	proprietà da ritornare (invece di un array completo)
string	<i>\$agent</i>	HTTP AGENT, default \$_SERVER['HTTP_USER_AGENT']

Returns

informazioni browser, una sola proprietà o un array (agent, name, version, is_bot, platform)

7.1.2.38 Gino\getDateDiff (*\$start_date*, *\$end_date* = null, *\$options* = array())

Calcola la differenza di tempo tra due datetime in più formati.

Parameters

string	<i>\$start_date</i>	datetime iniziale
string	<i>\$end_date</i>	datetime finale (default now)
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • diff (string): tipo di output <ul style="list-style-type: none"> – <i>s</i>, differenza in secondi (default) – <i>i</i>, differenza in minuti – <i>h</i>, differenza in ore

Returns

differenza di tempo

Utilizza la classe .

7.1.2.39 Gino\gOpt (*\$opt_name*, *\$opt_array*, *\$default*)

Restituisce l'elemento di un array corrispondente alla chiave data oppure un valore di default.

Parameters

string	<i>\$opt_name</i>	nome della chiave
array	<i>\$opt_array</i>	array associativo

mixed	<i>\$default</i>	valore di default
-------	------------------	-------------------

Returns

l'elemento corrispondente alla chiave data oppure il default

7.1.2.40 GinoHtmlChars (*\$string*, *\$id* = ' ', *\$options* = array())

Modifica il valore di un campo testo del database per visualizzarlo in HTML.

Parameters

string	<i>\$string</i>	testo
string	<i>\$id</i>	codice che raggruppa un insieme di immagini da visualizzare con le librerie slimbox
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • newline (boolean): inserisce dei tag BR prima di ogni nuova linea in una stringa (sostituisce i caratteri newline)

Returns

stringa modificata

7.1.2.41 GinoHtmlCharsText (*\$string*)

Mostra il valore di un campo testo del database formattato come "solo testo".

Inserisce dei tag BR prima di ogni nuova linea in una stringa (sostituisce i
)

Parameters

string	<i>\$string</i>
--------	-----------------

Returns

testo modificato

7.1.2.42 GinoHtmlInput (*\$string*)

Modifica il valore di un campo testo del database per visualizzarlo in un input form.

Parameters

string	<i>\$string</i>
--------	-----------------

Returns

testo modificato

7.1.2.43 GinoHtmlInputEditor (*\$string*)

Modifica il valore di un campo testo del database per visualizzarlo in un input form di tipo editor html.

Parameters

string	<i>\$string</i>	
--------	-----------------	--

Returns

testo modificato

7.1.2.44 GinoHtmlToEntities (*\$string*)

Converte le entities HTML, ma non i tag.

Parameters

string	<i>\$string</i>	
--------	-----------------	--

Returns

testo convertito

7.1.2.45 GinoIcon (*\$name*, *\$options* = array())

Icone.

Parameters

string	<i>\$name</i>	codice dell'icona <ul style="list-style-type: none"> • admin • attach • back • cart • check • close • config • content • duplicate • css • delete • detail • download • email • export • feed • group • help • home • input • insert • language • layout • link • list • minimize • modify • new • newpdf • palette • password • pdf • permission • print
Generated on Thu Nov 12 2015 12:33:31 for gino CMS by Doxygen		• return <ul style="list-style-type: none"> • revision • search

Returns

codice html icona

7.1.2.46 Ginolis_browser (*\$name*)

Verifica se il browser è quello dato come argomento.

Parameters

string	<i>\$name</i>	nome browser
--------	---------------	--------------

Returns

bool

7.1.2.47 Ginolis_browser_platform (*\$platform*)

Verifica la piattaforma del browser.

Parameters

string	<i>\$platform</i>	
--------	-------------------	--

Returns

bool

7.1.2.48 Ginolis_browser_version (*\$version*)

Verifica la versione del browser.

Parameters

string	<i>\$version</i>	
--------	------------------	--

Returns

bool

7.1.2.49 Ginolis_robot ()

Verifica se il client è un robot.

Returns

bool

7.1.2.50 GinolisNumeric (*\$variable*)

Controlla se una variabile è un numero o una stringa numerica.

Parameters

mixed	<i>\$variable</i>	valore della variabile (string integer float)
-------	-------------------	---

Returns

boolean

7.1.2.51 GinoIsValid (*\$type*, *\$var*)

Verifica se il valore della variabile è conforme al tipo di controllo indicato.

Parameters

string	<i>\$type</i>	tipo di controllo da eseguire <ul style="list-style-type: none">• IP: indirizzo IP• URL: indirizzo URL• Email: indirizzo email• ISBN: codice ISBN• Date: data• Time: orario• HexColor: codice colore esadecimale
string	<i>\$var</i>	valore della variabile

Returns

boolean

7.1.2.52 GinoJsVar (*\$string*, *\$newline* = false)

Escape testo che deve essere passato come variabile javascript.

Parameters

string	<i>\$string</i>	
boolean	<i>\$newline</i>	mantiene gli 'a capo' (default false)

Returns

testo escaped

7.1.2.53 GinoListProv ()

Elenco delle province.

Returns

array (sigla=>capoluogo)

7.1.2.54 GinoNumberToDB (*\$number*)

Formatta un numero per il database (il separatore decimale è il punto)

Parameters

string	<i>\$number</i>	numero
--------	-----------------	--------

Returns

numero formattato

7.1.2.55 Gino\obj_serialize (*\$instanceName*, *\$object*)

Operazione di serializzazione.

Viene creato nella directory dei contenuti dell'istanza il file *ser_nomeistanza.txt*

Parameters

string	<i>\$instanceName</i>	nome dell'istanza
object	<i>\$object</i>	oggetto da serializzare

Returns

void

7.1.2.56 Gino\obj_unserialize (*\$instanceName*)

Operazione di deserializzazione.

Parameters

string	<i>\$instanceName</i>	nome dell'istanza
--------	-----------------------	-------------------

Returns

object

7.1.2.57 Gino\platforms ()

Lista di piattaforme popolari.

Returns

array

7.1.2.58 Gino\preCodeParser (*\$string*)

Modifica il valore di un campo testo di tipo "codice" del database per visualizzarlo in HTML.

Si tratta di testo di tipo "codice", ovvero inserito con la funzione [codeToDB\(\)](#). Crea un blocco che racchiude il codice.

See also

[codeToDB\(\)](#)

Parameters

string	<i>\$string</i>	
--------	-----------------	--

Returns

testo modificato

7.1.2.59 Gino\relativePath (*\$abspath*)

Ricava il percorso relativo a partire da un percorso assoluto.

Parameters

string	<i>\$abspath</i>	percorso assoluto
--------	------------------	-------------------

Returns

percorso relativo

7.1.2.60 Gino\remove_attributes (*\$text*)

Rimuove l'attributo class quando vuoto.

Parameters

string	<i>\$text</i>	testo
--------	---------------	-------

Returns

testo ripulito

7.1.2.61 Gino\replaceChar (*\$text*)

Sostituzione caratteri "strani" di word.

Parameters

string	<i>\$text</i>	
--------	---------------	--

Returns

stringa ripulita

7.1.2.62 Gino\robots ()

Lista di web robots popolari.

Returns

array

7.1.2.63 Gino\searchNameFile (*\$dir*)

Lista files contenuti in una directory.

Parameters

string	<i>\$dir</i>	percorso della directory (se <i>dir</i> è un percorso relativo, verrà aperta la directory relativa alla directory corrente)
--------	--------------	---

Returns

array di nomi di file

7.1.2.64 Gino\share (\$site, \$url, \$title = null, \$description = null)

Codice html di condivisione social network.

Parameters

string	<i>\$site</i>	tipo di condivisione (facebook, twitter, linkedin, googleplus)
string	<i>\$url</i>	indirizzo da condividere
string	<i>\$title</i>	titolo della condivisione
string	<i>\$description</i>	descrizione

Returns

codice

7.1.2.65 Gino\shareAll (\$social, \$url, \$title = null, \$description = null)

Codice condivisione social network con integrazione sharethis.

Parameters

array	<i>\$social</i>	elenco delle tipologie di condivisione (facebook, twitter, linkedin, googleplus, digg); col valore <i>all</i> vengono mostrate tutte le condivisioni
string	<i>\$url</i>	indirizzo da condividere
string	<i>\$title</i>	titolo della condivisione
string	<i>\$description</i>	descrizione

Returns

codice

7.1.2.66 Gino\slimboxReplace (\$string, \$id)

Modifica il valore di un campo testo del database per attivare le librerie slimbox.

Parameters

string	<i>\$string</i>	testo
string	<i>\$id</i>	codice che raggruppa un insieme di immagini da visualizzare con le librerie slimbox

Returns

testo modificato

7.1.2.67 Gino\strip_invisible_tags (\$text)

Rimuove script, stili e altro testo indesiderato (e invisibile) tra i tag.

Parameters

string	<i>\$text</i>	testo
--------	---------------	-------

Returns

string

7.1.2.68 Gino\strip_selected_tags (*\$text*, *\$tags* = ' ', *\$stripContent* = false)

Rimuove i tag indicati.

Simile a strip_tags(), ma mentre strip_tags rimuove tutti i tag, questa funzione li preserva

Parameters

string	<i>\$text</i>	testo
string	<i>\$tags</i>	stringa con i tag da rimuovere, ad esempio "<a><p><quote>"
boolean	<i>\$stripContent</i>	rimuove anche il testo contenuto tra l'apertura e la chiusura del tag

Returns

stringa ripulita

7.1.2.69 Gino\strip_tags_attributes (*\$text*, *\$strip_js*, *\$strip_attributes*)

Rimuove gli attributi javascript insicuri nei tag html.

See also

[remove_attributes\(\)](#)

Parameters

string	<i>\$text</i>	testo
boolean	<i>\$strip_js</i>	rimuove gli attributi javascript
boolean	<i>\$strip_attributes</i>	rimuove alcuni attributi html

Returns

testo ripulito

7.1.2.70 Gino\stripAll (*\$text*)

Sostituzione entities con il carattere corrispondente.

Questa funzione viene utilizzata per "ripulire" il testo che arriva dall'editor html (CK Editor)

Parameters

string	<i>\$text</i>	testo
--------	---------------	-------

Returns

testo ripulito

7.1.2.71 Gino\stripEditor (*\$text*)

Sostituisce alcuni tag utilizzati dall'editor html con i tag standard.

Questa funzione viene utilizzata per "ripulire" il testo che arriva dall'editor html (CK Editor)

Parameters

string	<i>\$text</i>	testo
--------	---------------	-------

Returns

testo ripulito

7.1.2.72 Gino\timeDiff (*\$firstTime*, *\$lastTime*)

Calcola l'intervallo di tempo in secondi tra due valori datetime.

Parameters

string	<i>\$firstTime</i>	datetime iniziale
string	<i>\$lastTime</i>	datetime finale

Returns

intervallo di tempo

7.1.2.73 Gino\timeToDbTime (*\$time*)

Formatta l'orario per il database (HH:MM:SS)

Parameters

string	<i>\$time</i>	orario ([00][{, :}00][{, :}00])
--------	---------------	---------------------------------

Returns

string

7.1.2.74 Gino\traslitterazione (*\$numero*, *\$decimale* = *false*)

Converte un numero in cifre.

Parameters

mixed	<i>\$numero</i>	valore da convertire (float integer)
boolean	<i>\$decimale</i>	se vero, mostra il decimale ([/00])

Returns

numero in cifre

7.1.2.75 Gino\utf8_urldecode (*\$str*)

Conversione dei dati Unicode \$_GET/\$_POST, generati dalla funzione javascript escape(), in UTF8 per il processo server-side.

Parameters

string	<i>\$str</i>	
--------	--------------	--

Returns

stringa decodificata

7.1.2.76 Gino\verifyExtension (\$filename, \$extensions)

Controlla se l'estensione di un file è valida.

Parameters

string	<i>\$filename</i>	nome del file
array	<i>\$extensions</i>	elenco dei formati di file permessi

Returns

TRUE se l'estensione è compresa in quelle date, FALSE altrimenti

7.1.3 Variable Documentation**7.1.3.1 const APP_DIR SITE_ROOT.OS.'app'**

Percorso assoluto alla directory che contiene le directory dei moduli.

7.1.3.2 const CACHE_DIR SITE_ROOT.OS.'cache'

Percorso assoluto alla directory cache.

7.1.3.3 const CLASSES_DIR LIB_DIR.OS.'classes'

Percorso assoluto alla directory che contiene le classi che non fanno parte di moduli.

7.1.3.4 const CONTENT_DIR SITE_ROOT.OS.'contents'

Percorso assoluto alla directory che contiene upload degli utenti.

7.1.3.5 const CONTENT_WWW SITE_WWW.'/contents'

Percorso relativo alla directory che contiene file uploadati dall'utente.

7.1.3.6 const CONTROLLER_CLASS_PREFIX 'class_'

Prefisso classi di tipo [Gino.Controller](#).

7.1.3.7 const CORE_CLASS_PREFIX 'class.'

Prefisso classi non interne a moduli.

7.1.3.8 `const CSS_DIR SITE_ROOT.OS.'css'`

Percorso assoluto alla directory che contiene css.

7.1.3.9 `const CSS_WWW SITE_WWW.'/css'`

Percorso relativo alla directory dei css.

7.1.3.10 `const FIELDS_DIR CLASSES_DIR.OS.'fields'`

Percorso assoluto alla directory che contiene le classi che descrivono campi di db.

7.1.3.11 `const FONTS_DIR SITE_ROOT.OS.'fonts'`

Percorso assoluto alla directory che contiene fonts files.

7.1.3.12 `const GIMAGE_DIR CONTENT_DIR.OS.'gimage'`

7.1.3.13 `const GRAPHICS_DIR SITE_ROOT.OS.'graphics'`

Percorso assoluto alla directory che contiene file di grafica.

7.1.3.14 `const HOME_FILE SITE_WWW.'/.'$home_file`

Percorso relativo dello SCRIPT_FILE.

7.1.3.15 `const LIB_DIR SITE_ROOT.OS.'lib'`

Percorso assoluto alla directory lib.

7.1.3.16 `const MODEL_CLASS_PREFIX 'class.'`

Prefisso classi di tipo [Gino.Model](#).

7.1.3.17 `const OS DIRECTORY_SEPARATOR`

Separatore di directory definito dal sistema operativo.

7.1.3.18 `const PLUGIN_DIR LIB_DIR.OS.'plugin'`

Percorso assoluto alla directory che contiene plugins.

7.1.3.19 `const SITE_APP SITE_WWW.'/app'`

Percorso relativo alla directory che contiene i moduli.

7.1.3.20 `const SITE_CUSTOM_CKEDITOR SITE_LIB.'/custom_ckeditor'`

Percorso relativo alla directory che contiene file di customizzazione ckeditor.

7.1.3.21 `const SITE_GRAPHICS SITE_WWW.'/graphics'`

Percorso relativo alla directory che contiene file di grafica.

7.1.3.22 `const SITE_IMG SITE_WWW.'/img'`

Percorso relativo alla directory che contiene immagini si sistema.

7.1.3.23 `const SITE_JS SITE_LIB.'/js'`

Percorso relativo alla directory che contiene librerie javascript.

7.1.3.24 `const SITE_LIB SITE_WWW.'/lib'`

Percorso relativo alla directory lib.

7.1.3.25 `const SITE_ROOT dirname(realpath(__FILE__))`

SITE_ROOT Percorso assoluto alla root directory

7.1.3.26 `const SITE_WWW $site_www`

SITE_WWW Percorso relativo dell'applicazione a partire dalla root directory

7.1.3.27 `const TBL_GROUP 'auth_group'`

Nome della tabella dei gruppi di utenti.

7.1.3.28 `const TBL_GROUP_PERMISSION 'auth_group_perm'`

Nome della tabella di join gruppi - permessi.

7.1.3.29 `const TBL_LANGUAGE 'language'`

Nome della tabella lingue.

7.1.3.30 `const TBL_LOG_ACCESS 'sys_log_access'`

Nome della tabella che contiene il log degli accessi.

7.1.3.31 `const TBL_MODULE 'sys_module'`

Nome della tabella che contiene le istanze di moduli.

7.1.3.32 `const TBL_MODULE_APP 'sys_module_app'`

Nome della tabella che contiene i moduli installati.

7.1.3.33 `const TBL_NATION 'nation'`

Nome della tabella nazioni.

7.1.3.34 `const TBL_PERMISSION 'auth_permission'`

Nome della tabella dei permessi.

7.1.3.35 `const TBL_REGISTRATION_PROFILE 'auth_registration_profile'`

Nome della tabella dei profili di registrazione utenti.

7.1.3.36 `const TBL_REGISTRATION_PROFILE_GROUP 'auth_registration_profile_group'`

Nome della tabella di associazione profili di registrazione utenti - gruppi.

7.1.3.37 `const TBL_REGISTRATION_REQUEST 'auth_registration_request'`

Nome della tabella delle richieste di registrazione.

7.1.3.38 `const TBL_SYS_CONF 'sys_conf'`

Nome della tabella che contiene la configurazione del sistema.

7.1.3.39 `const TBL_TRANSLATION 'language_translation'`

Nome della tabella traduzioni.

7.1.3.40 `const TBL_USER 'auth_user'`

Nome della tabella utenti.

7.1.3.41 `const TBL_USER_ADD 'auth_user_add'`

Nome della tabella informazioni aggiuntive utenti.

7.1.3.42 `const TBL_USER_GROUP 'auth_user_group'`

Nome della tabella di join utenti - gruppi.

7.1.3.43 `const TBL_USER_PERMISSION 'auth_user_perm'`

Nome della tabella di join utenti - permessi.

7.1.3.44 `const TMP_DIR '/tmp'`

Percorso assoluto alla directory per la creazione di file temporanei.

7.1.3.45 `const TPL_DIR SITE_ROOT.OS.'templates'`

Percorso assoluto alla directory che contiene templates.

7.1.3.46 `const VIEWS_DIR SITE_ROOT.OS.'views'`

Percorso assoluto alla directory che contiene viste generiche di sistema.

7.2 Gino\App Namespace Reference

Namespaces

- [Attachment](#)
Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati
- [Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi
- [Graphics](#)
Namespace dell'applicazione [Graphics](#), che gestisce in maniera semplificata header e footer
- [Index](#)
Namespace dell'applicazione [Index](#), che gestisce la home amministrativa
- [Instruments](#)
Namespace dell'applicazione [Instruments](#), per la visualizzazione di utilità quali percorsi e mime types
- [Language](#)
Namespace dell'applicazione [Language](#), che gestisce le lingue installate sul sistema e disponibili per le traduzioni
- [Layout](#)
Namespace dell'applicazione [Layout](#), che gestisce i layout del sistema.
- [Menu](#)
Namespace dell'applicazione [Menu](#), che gestisce istanze di menu
- [Module](#)
Namespace dell'applicazione [Module](#), che gestisce la creazione/rimozione di istanze di moduli di sistema
- [Page](#)
Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate
- [PhpModuleView](#)
Namespace dell'applicazione [PhpModuleView](#), che gestisce output generati direttamente da codice php salvato da database
- [SearchSite](#)
Namespace dell'applicazione [SearchSite](#), che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca
- [Statistics](#)
Namespace dell'applicazione [Statistics](#), che gestisce le statistiche di accesso all'area privata
- [SysClass](#)
Namespace dell'applicazione [SysClass](#), che gestisce l'installazione/upgrade/rimozione di classi di sistema
- [Sysconf](#)
Namespace dell'applicazione [Sysconf](#), che gestisce le impostazioni di sistema
- [Sysfunc](#)
Namespace dell'applicazione [Sysfunc](#), per la gestione della visualizzazione di pagine di errore

7.3 Gino\App\Attachment Namespace Reference

Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati

Data Structures

- class [attachment](#)
Classe controller del modulo di gestione di archivi di file categorizzati.
- class [AttachmentCtg](#)
Classe di tipo [Gino.Model](#) che rappresenta una categoria di allegati.
- class [AttachmentItem](#)
Classe di tipo [Gino.Model](#) che rappresenta un singolo allegato.
- class [AttachmentItemAdminTable](#)
Estende la classe [Gino.AdminTable](#) per gestire il salvataggio dell'allegato quando viene modificata la directory (categoria)

7.3.1 Detailed Description

Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati

7.4 Gino\App\Auth Namespace Reference

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

Data Structures

- class [AdminTable_AuthUser](#)
Sovrascrive la classe [AdminTable](#).
- class [auth](#)
Classe di tipo [Gino.Controller](#) per la gestione degli utenti, gruppi e permessi.
- class [Group](#)
Classe tipo [Gino.Model](#) che rappresenta un gruppo di utenti.
- class [Ldap](#)
Libreria di connessione ai server [Ldap](#).
- class [Permission](#)
Classe tipo [Gino.Model](#) che rappresenta un permesso.
- class [RegistrationProfile](#)
Classe tipo [Gino.Model](#) che rappresenta un profilo di registrazione utenti.
- class [RegistrationRequest](#)
Classe tipo [Gino.Model](#) che rappresenta una richiesta di registrazione.
- class [User](#)
Classe di tipo [Gino.Model](#) che rappresenta un utente.

Variables

- const [LDAP_HOST](#) "
- const [LDAP_PORT](#) '389'
- const [LDAP_BASE_DN](#) "
- const [LDAP_SEARCH_DN](#) "
- const [LDAP_APP_USERNAME](#) "
- const [LDAP_APP_PASSWORD](#) "
- const [LDAP_DOMAIN](#) "
- const [LDAP_PROTOCOL_VERSION](#) "
- const [LDAP_FILTER_SEARCH](#) "

7.4.1 Detailed Description

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

7.4.2 Variable Documentation

7.4.2.1 `const LDAP_APP_PASSWORD "`

7.4.2.2 `const LDAP_APP_USERNAME "`

7.4.2.3 `const LDAP_BASE_DN "`

7.4.2.4 `const LDAP_DOMAIN "`

7.4.2.5 `const LDAP_FILTER_SEARCH "`

7.4.2.6 `const LDAP_HOST "`

7.4.2.7 `const LDAP_PORT '389'`

7.4.2.8 `const LDAP_PROTOCOL_VERSION "`

7.4.2.9 `const LDAP_SEARCH_DN "`

7.5 Gino\App\Graphics Namespace Reference

Namespace dell'applicazione [Graphics](#), che gestisce in maniera semplificata header e footer

Data Structures

- class [graphics](#)
Classe di tipo [Gino.Controller](#) per la gestione personalizzata degli header e footer del sistema.
- class [GraphicsItem](#)
Classe di tipo [Gino.Model](#) che rappresenta un header/footer.

7.5.1 Detailed Description

Namespace dell'applicazione [Graphics](#), che gestisce in maniera semplificata header e footer

7.6 Gino\App\Index Namespace Reference

Namespace dell'applicazione [Index](#), che gestisce la home amministrativa

Data Structures

- class [index](#)
Classe di tipo [Gino.Controller](#) del modulo che gestisce la home amministrativa.

7.6.1 Detailed Description

Namespace dell'applicazione [Index](#), che gestisce la home amministrativa

7.7 Gino\App\Instruments Namespace Reference

Namespace dell'applicazione [Instruments](#), per la visualizzazione di utilità quali percorsi e mime types

Data Structures

- class [instruments](#)

Classe di tipo [Gino.Controller](#) per la gestione di strumenti aggiuntivi di gino.

7.7.1 Detailed Description

Namespace dell'applicazione [Instruments](#), per la visualizzazione di utilità quali percorsi e mime types

7.8 Gino\App\Language Namespace Reference

Namespace dell'applicazione [Language](#), che gestisce le lingue installate sul sistema e disponibili per le traduzioni

Data Structures

- class [Lang](#)

Classe di tipo [Gino.Model](#) che rappresenta una lingua installata sul sistema.

- class [language](#)

Classe di tipo [Gino.Controller](#) per la gestione delle lingue disponibili per le traduzioni.

7.8.1 Detailed Description

Namespace dell'applicazione [Language](#), che gestisce le lingue installate sul sistema e disponibili per le traduzioni

7.9 Gino\App\Layout Namespace Reference

Namespace dell'applicazione [Layout](#), che gestisce i layout del sistema.

Data Structures

- class [layout](#)

Gestisce il layout dell'applicazione raggruppando le funzionalità fornite dalle librerie dei css, template e skin.

7.9.1 Detailed Description

Namespace dell'applicazione [Layout](#), che gestisce i layout del sistema.

Consente la modifica dei template, css e viste generali utilizzate da tutto il sistema.

7.10 Gino\App\Menu Namespace Reference

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

Data Structures

- class [menu](#)
Classe di tipo [Gino.Controller](#) per la gestione dei menu.
- class [MenuVoice](#)
Classe di tipo [Gino.Model](#) che rappresenta una voce di menu.

7.10.1 Detailed Description

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

7.11 Gino\App\Module Namespace Reference

Namespace dell'applicazione [Module](#), che gestisce la creazione/rimozione di istanze di moduli di sistema

Data Structures

- class [module](#)
Classe di tipo [Gino.Controller](#) per la gestione di istanze di moduli di sistema.
- class [ModuleInstance](#)
Classe di tipo [Gino.Model](#) che rappresenta un'istanza di un modulo di sistema.

7.11.1 Detailed Description

Namespace dell'applicazione [Module](#), che gestisce la creazione/rimozione di istanze di moduli di sistema

7.12 Gino\App\Page Namespace Reference

Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

Data Structures

- class [page](#)
Classe di tipo [Gino.Controller](#) per la gestione delle pagine.
- class [PageCategory](#)
Classe tipo [Gino.Model](#) che rappresenta una categoria di pagine.
- class [PageComment](#)
Classe tipo [Gino.Model](#) che rappresenta un commento ad una pagina.
- class [PageEntry](#)
Classe tipo [Gino.Model](#) che rappresenta una pagina.

7.12.1 Detailed Description

Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

7.13 Gino\App\PhpModuleView Namespace Reference

Namespace dell'applicazione [PhpModuleView](#), che gestisce output generati direttamente da codice php salvato da database

Data Structures

- class [PhpModule](#)
Classe di tipo [Gino.Model](#) che rappresenta un output scritto in codice php.
- class [phpModuleView](#)
Permette la creazione di moduli di classe in grado di eseguire codice php completamente personalizzabile.

7.13.1 Detailed Description

Namespace dell'applicazione [PhpModuleView](#), che gestisce output generati direttamente da codice php salvato da database

7.14 Gino\App\SearchSite Namespace Reference

Namespace dell'applicazione [SearchSite](#), che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca

Data Structures

- class [searchSite](#)
Gestisce le ricerche full text sui contenuti dell'applicazione.

7.14.1 Detailed Description

Namespace dell'applicazione [SearchSite](#), che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca

7.15 Gino\App\Statistics Namespace Reference

Namespace dell'applicazione [Statistics](#), che gestisce le statistiche di accesso all'area privata

Data Structures

- class [LogAccess](#)
Modello che rappresenta un login sul sistema.
- class [statistics](#)
Statistiche degli accessi all'area privata.

7.15.1 Detailed Description

Namespace dell'applicazione [Statistics](#), che gestisce le statistiche di accesso all'area privata

7.16 Gino\App\SysClass Namespace Reference

Namespace dell'applicazione [SysClass](#), che gestisce l'installazione/upgrade/rimozione di classi di sistema

Data Structures

- class [ModuleApp](#)
Classe di tipo [Gino.Model](#) che rappresenta un modulo di sistema.
- class [sysClass](#)
Classe di tipo [Gino.Controller](#) per la gestione dei moduli di sistema.

7.16.1 Detailed Description

Namespace dell'applicazione [SysClass](#), che gestisce l'installazione/upgrade/rimozione di classi di sistema

7.17 Gino\App\Sysconf Namespace Reference

Namespace dell'applicazione [Sysconf](#), che gestisce le impostazioni di sistema

Data Structures

- class [Conf](#)
Classe di tipo [Gino.Model](#) che gestisce le impostazioni del sistema.
- class [sysconf](#)
Classe di tipo [Gino.Controller](#) per la gestione delle principali impostazioni di sistema.

7.17.1 Detailed Description

Namespace dell'applicazione [Sysconf](#), che gestisce le impostazioni di sistema

7.18 Gino\App\Sysfunc Namespace Reference

Namespace dell'applicazione [Sysfunc](#), per la gestione della visualizzazione ddi pagine di errore

Data Structures

- class [sysfunc](#)
Classe di tipo [Gino.Controller](#) che gestisce metodi che stampano pagine di errore.

7.18.1 Detailed Description

Namespace dell'applicazione [Sysfunc](#), per la gestione della visualizzazione ddi pagine di errore

7.19 Gino\Exception Namespace Reference

Namespace che comprende tutte le classi di tipo [Exception](#)

Data Structures

- class [Exception403](#)
Classe per la gestione di eccezioni 403.
- class [Exception404](#)
Classe per la gestione di eccezioni 404.
- class [Exception500](#)
Classe per la gestione di eccezioni 500.

7.19.1 Detailed Description

Namespace che comprende tutte le classi di tipo [Exception](#)

7.20 Gino\Http Namespace Reference

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

Data Structures

- class [Redirect](#)
Subclass di [Gino.Http.Response](#) per gestire reindirizzamenti.
- class [Request](#)
Wrapper di una richiesta HTTP.
- class [Response](#)
Wrapper di una risposta HTTP.
- class [ResponseFile](#)
Subclass di per gestire stream di file.
- class [ResponseForbidden](#)
Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 403.
- class [ResponseJson](#)
Subclass di per gestire risposte in formato json.
- class [ResponseNotFound](#)
Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 404.
- class [ResponsePdf](#)
Subclass di per gestire risposte in formato pdf.
- class [ResponseServerError](#)
Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori interni (code 500)

7.20.1 Detailed Description

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

7.21 Gino\Plugin Namespace Reference

Namespace che comprende classi di tipo plugin

Data Structures

- class [mysql](#)
Libreria di connessione ai database MySQL.
- class [pdo](#)
Libreria di connessione ai database.
- class [pdo_mysql](#)
Driver specifico per la connessione a un database MYSQL attraverso la libreria PDO.
- class [pdo_sqlsrv](#)
Driver specifico per la connessione a un database SQL Server attraverso la libreria PDO.
- class [plugin_phpfastcache](#)
Interfaccia alla libreria phpfastcache.
- class [sqlsrv](#)
Libreria di connessione ai database SQL Server.

7.21.1 Detailed Description

Namespace che comprende classi di tipo plugin

7.22 Gino\Test Namespace Reference

Namespace che comprende gli UNIT TEST di classi e funzioni di gino

Data Structures

- class [FuncTest](#)
Classe di tipo PHPUnit_Framework_TestCase per testare il file [func.php](#).
- class [PaginatorTest](#)
Classe di tipo PHPUnit_Framework_TestCase per testare la classe [Gino.Paginator](#).

Functions

- [setPage](#) ([\\$page](#))

7.22.1 Detailed Description

Namespace che comprende gli UNIT TEST di classi e funzioni di gino

7.22.2 Function Documentation

7.22.2.1 Gino\Test\setPage (*\$page*)

Chapter 8

Data Structure Documentation

8.1 Access Class Reference

Classe per la gestione dell'autenticazione ed accesso alla funzionalità

Public Member Functions

- [__construct](#) ()
Costruttore.
- [Authentication](#) (\Gino\Http\Request \$request)
Autenticazione all'applicazione.
- [requireAdmin](#) ()
Verifica che l'utente si amministratore del sito.
- [requirePerm](#) (\$class, \$perm, \$instance=0)
Verifica se l'utente non ha almeno uno dei permessi dati.

Data Fields

- [\\$_session](#)

Protected Attributes

- [\\$_home](#)
- [\\$_db](#)

Private Member Functions

- [loginError](#) (\Gino\Http\Request \$request)
Autenticazione errata.
- [loginSuccess](#) (\Gino\Http\Request \$request)
Autenticazione valida.
- [AuthenticationMethod](#) (\$username, \$password)
Imposta le variabili di sessione (user_id, user_name) e logga l'accesso ([logAccess\(\)](#))
- [logAccess](#) (\$userid)
Registra il log dell'accesso all'applicazione.

8.1.1 Detailed Description

Classe per la gestione dell'autenticazione ed accesso alla funzionalità

La classe gestisce il processo di autenticazione e l'accesso al sito e alle sue funzionalità

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.1.2 Constructor & Destructor Documentation

8.1.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.Access](#)

8.1.3 Member Function Documentation

8.1.3.1 Authentication (`\Gino\Http\Request $request`)

Autenticazione all'applicazione.

Parametri POST:

- *action* (string), con valore auth (procedura di autenticazione)
- *user* (string), lo username
- *pwd* (string), la password

Parametri GET:

- *action* (string), con valore logout (procedura di logout)

See also

[AuthenticationMethod\(\)](#)
[loginSuccess\(\)](#)
[loginError\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#) o FALSE se non avvengono autenticazione e logout

8.1.3.2 AuthenticationMethod (*\$username*, *\$password*) [private]

Imposta le variabili di sessione (user_id, user_name) e logga l'accesso ([logAccess\(\)](#))

See also

[::checkAuthenticationUser\(\)](#)

Parameters

string	<i>\$username</i>	
string	<i>\$password</i>	

Returns

risultato autenticazione, bool

8.1.3.3 logAccess (*\$userid*) [private]

Registra il log dell'accesso all'applicazione.

Parameters

integer	<i>\$userid</i>	valore ID dell'utente
---------	-----------------	-----------------------

Returns

risultato operazione, bool

8.1.3.4 loginError (\Gino\Http\Request *\$request*) [private]

Autenticazione errata.

Setta l'errore in sessione e ritorna una [Gino.Http.Redirect](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#) alla pagina di autenticazione

8.1.3.5 loginSuccess (\Gino\Http\Request *\$request*) [private]

Autenticazione valida.

Reindirizza alla home page o all'url impostato in sessione (auth_redirect).

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.1.3.6 requireAdmin ()

Verifica che l'utente si amministratore del sito.

Se la condizione non è verificata getta una [Gino.Exception.Exception403](#)

Returns

void

8.1.3.7 requirePerm (*\$class*, *\$perm*, *\$instance* = 0)

Verifica se l'utente non ha almeno uno dei permessi dati.

Se la condizione non è verificata getta una [Gino.Exception.Exception403](#) se l'utente è autenticato, altrimenti reindirizza alla pagina di login e ferma l'esecuzione

Parameters

string	<i>\$class</i>	nome classe senza namespace
string	<i>\$perm</i>	codice permesso
int	<i>\$instance</i>	id istanza modulo

Returns

void

8.1.4 Field Documentation

8.1.4.1 *\$_db* [protected]8.1.4.2 *\$_home* [protected]8.1.4.3 *\$_session*

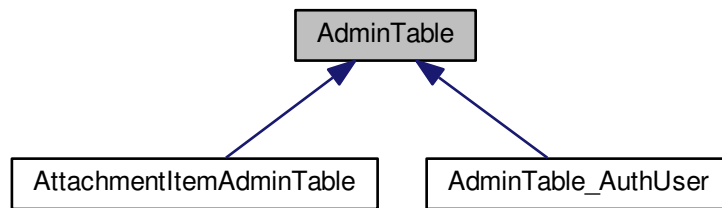
The documentation for this class was generated from the following file:

- lib/classes/[class.Access.php](#)

8.2 AdminTable Class Reference

Gestisce l'interfaccia di amministrazione di un modello con inserimento, modifica ed eliminazione.

Inheritance diagram for AdminTable:



Public Member Functions

- [__construct](#) (\$controller, \$opts=array())
Costruttore.
- [hidden](#) (\$hidden=array())
Setta la proprietà _hidden (campi hidden del form)
- [modelForm](#) (\$model, \$options=array(), \$inputs=array())
Generazione automatica del form di inserimento/modifica di un [Gino.Model](#).
- [modelAction](#) (\$model, \$options=array(), \$options_element=array())
Salvataggio dei dati a seguito del submit di un form di inserimento/modifica.
- [backOffice](#) (\$model_class_name, \$options_view=array(), \$options_form=array(), \$inputs=array())
Gestisce il backoffice completo di un modello (wrapper)
- [adminDelete](#) (\$model, \$options_form)
Eliminazione di un record (richiesta di conferma soltanto da javascript)
- [adminForm](#) (\$model_obj, \$options_form, \$inputs)
Wrapper per mostrare e processare il form.
- [adminList](#) (\$model, \$options_view=array())
Lista dei record del modello.
- [permission](#) (\$options, \$fname)
Permessi di modifica dei campo.

Data Fields

- [\\$_request](#)
- [\\$_db](#)
- [\\$_session](#)
- [\\$_view](#)
- [\\$_hidden](#)
- [\\$_edit_deny](#)
- [\\$_delete_deny](#)
- [\\$_list_remove](#)

Protected Member Functions

- [setSessionSearch](#) (\$model)
Setta le variabili di sessione usate per filtrare i record nella lista amministrativa.
- [setSessionSearchAdd](#) (\$model, \$filters)
Setta le variabili di sessione usate per filtrare i record nella lista amministrativa (riferimento ai filtri non automatici)
- [addWhereClauses](#) (&\$query_where, \$model)
Aggiunge le condizioni where usate per filtrare i record nella admin list all'argomento \$query_where passato per reference.
- [formFilters](#) (\$model, \$options)
Form per filtraggio record.
- [editUrl](#) (\$add_params=array(), \$remove_params=array())
Costruisce il percorso per il reindirizzamento.
- [readFile](#) (\$model, \$path_to_file, \$options=array())
Legge il file e ne importa il contenuto.
- [restore](#) (\$table, \$filename, \$options=array())
Restore di un file.
- [dump](#) (\$table, \$filename, \$options=array())
Dump di una tabella.

Protected Attributes

- [\\$_controller](#)
- [\\$_registry](#)
- [\\$_form](#)
- [\\$_allow_insertion](#)
- [\\$_filter_fields](#)
- [\\$_filter_join](#)
- [\\$_filter_add](#)
- [\\$_list_display](#)
- [\\$_ifp](#)

Private Member Functions

- [adminListOrder](#) (\$order_dir, \$table, \$name)
Definisce l'ordinamento della query.
- [m2mThroughAction](#) (\$m2m_name, \$m2m_object, \$model, \$options=array())
Salvataggio dei campi [Gino.ManyToManyThroughField](#).
- [addWhereJoin](#) (\$model, \$class_name, \$fname)
Elementi che concorrono a determinare le condizioni di ricerca dei campi automatici.
- [addWhereExtra](#) (\$class_name)
Definizione delle condizioni di ricerca aggiuntive a quelle sui campi automatici.
- [formFiltersAdd](#) (\$filters, \$fname, \$class_name, \$gform)
Input form dei filtri aggiuntivi.
- [clean](#) (\$name, \$options=null)
Ripulisce un input per l'inserimento in database.

8.2.1 Detailed Description

Gestisce l'interfaccia di amministrazione di un modello con inserimento, modifica ed eliminazione.

Fornisce gli strumenti per gestire la parte amministrativa di un modulo, mostrando gli elementi e interagendo con loro (inserimento, modifica, eliminazione).

Nel metodo [backOffice\(\)](#) viene ricercato automaticamente il parametro 'id' come identificatore del record sul quale interagire. Non utilizzare il parametro 'id' per altri riferimenti.

Il campo di nome *instance* viene trattato diversamente dagli altri campi: non compare nel form e il valore gli viene passato direttamente dall'istanza.

Per attivare i filtri di ricerca nella pagina di visualizzazione dei record occorre indicare i campi sui quali applicare il filtro nella chiave *filter_fields* (opzioni della vista).

Nella tabella di visualizzazione dei record i campi sui quali è possibile ordinare i risultati sono quelli per i quali la tipologia è "ordinabile", ovvero il metodo *Gino.Field::canBeOrdered()* ritorna il valore *true*.

Per attivare l'importazione dei file utilizzare l'opzione *import_file* come specificato nel metodo [modelAction\(\)](#) e sovrascrivere il metodo [readFile\(\)](#).

Gestione dei permessi

La gestione delle autorizzazioni a operare sulle funzionalità del modulo avviene impostando opportunamente le opzioni *allow_insertion*, *edit_deny*, *delete_deny* quando si istanzia la classe *adminTable()*.

Esempio:

```
// se gruppo1 $edit_deny = 'all'
// se gruppo2 $edit_deny = array(2);
// altrimenti $edit_deny = null;
$admin_table = new adminTable($this, array('allow_insertion'=>true, 'delete_deny'=>'all'));
```

La gestione fine delle autorizzazioni a operare sui singoli campi della tabella avviene indicando i gruppi autorizzati nell'array delle opzioni della funzionalità utilizzando la chiave *permission*.

Il formato è il seguente:

```
$buffer = $admin_table->backOffice('elearningCtg',
    array(
        'list_display' => array('id', 'name'),
        'add_params_url'=>array('block'=>'ctg')
    ),
    array(
        'permission'=>array(
            'view'=>group,
            'fields'=>array(
                'field1'=>group,
                'field2'=>group
            )
        )
    )
);
```

dove *group* (mixed) indica il o i gruppi autorizzati a una determinata funzione/campo.

La chiave *view* contiene il permesso di accedere alla singola funzionalità (view, edit, delete), e per il momento non viene utilizzata.

Tabella delle associazioni del tipo di campo con il tipo input di default

Classe	Tipo di campo	Widget principale
charField()	CHAR, VARCHAR	text
dateField()	DATE	date
datetimeField()	DATETIME	null
directoryField()	CHAR, VARCHAR	text
enumField()	ENUM	radio
fileField()	FILE	file
floatField()	FLOAT, DOUBLE, DECIMAL	float
foreignKeyField()	FOREIGN_KEY	select
imageField()	IMAGE	image
integerField()	TINYINT, SMALLINT, MEDIUMINT, INT	text
manyToManyField()	CHAR, VARCHAR	multicheck
textField()	TEXT	textarea
timeField()	TIME	time
yearField()	YEAR	text

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.2.2 Constructor & Destructor Documentation**8.2.2.1 __construct (\$controller, \$opts = array())**

Costruttore.

Parameters

\Gino\↔ Controller	\$controller	istanza di Gino.Controller che gestisce il backend
array	\$opts	array associativo di opzioni <ul style="list-style-type: none"> • view_folder (string): percorso della directory contenente la vista da caricare • allow_insertion (boolean): indica se permettere o meno l'inserimento di nuovi record • edit_deny (mixed): indica quali sono gli ID dei record che non possono essere modificati <ul style="list-style-type: none"> – string, 'all' -> tutti – array, elenco ID • delete_deny (mixed): indica quali sono gli ID dei record che non possono essere eliminati <ul style="list-style-type: none"> – string, 'all' -> tutti – array, elenco ID

Returns

istanza di [Gino.AdminTable](#)

8.2.3 Member Function Documentation

8.2.3.1 addWhereClauses (& \$query_where, \$model) [protected]

Aggiunge le condizioni where usate per filtrare i record nella admin list all'argomento \$query_where passato per reference.

See also

`self::addWhereJoin()`
`self::addWhereExtra()`

Parameters

array	<i>\$query_where</i>	reference all'array di where clauses già impostate
\Gino\↔Model	<i>\$model</i>	istanza di Gino.Model

Returns

void

8.2.3.2 addWhereExtra (\$class_name) [private]

Definizione delle condizioni di ricerca aggiuntive a quelle sui campi automatici.

La condizione da inserire nella query di ricerca viene definita nel metodo indicato come valore della chiave *filter*. Il metodo deve essere creato di volta in volta.

Parameters

string	<i>\$class_name</i>	nome della classe
--------	---------------------	-------------------

Returns

array di condizioni

8.2.3.3 addWhereJoin (\$model, \$class_name, \$fname) [private]

Elementi che concorrono a determinare le condizioni di ricerca dei campi automatici.

Ci può essere una solo campo input di tipo join.

Parameters

object	<i>\$model</i>	modello
string	<i>\$class_name</i>	nome della classe
string	<i>\$fname</i>	nome del campo della tabella al quale associare le condizioni aggiuntive

Returns

array di condizioni o null

8.2.3.4 adminDelete (*\$model*, *\$options_form*)

Eliminazione di un record (richiesta di conferma soltanto da javascript)

See also

`self::backOffice()`

Parameters

<code>\Gino\↔ Model</code>	<i>\$model</i>	modello da eliminare
array	<i>\$options_form</i>	array associativo di opzioni <ul style="list-style-type: none"> • link_delete (string): indirizzo al quale si viene rimandati dopo la procedura di eliminazione del record (se non presente viene costruito automaticamente)

Returns

[Gino.Http.Redirect](#)

8.2.3.5 adminForm (*\$model_obj*, *\$options_form*, *\$inputs*)

Wrapper per mostrare e processare il form.

See also

`self::modelForm()`
`self::modelAction()`
`self::editUrl()`

Parameters

<code>\Gino\↔ Model</code>	<i>\$model_obj</i>	istanza di Gino.Model da inserire/modificare
array	<i>\$options_form</i>	array associativo di opzioni <ul style="list-style-type: none"> • link_return (string): indirizzo al quale si viene rimandati dopo un esito positivo del form (se non presente viene costruito automaticamente) • form_description (string): testo che compare tra il titolo ed il form
array	<i>\$inputs</i>	

Returns

[Gino.Http.Redirect](#) se viene richiesta una action o si verifica un errore, form html altrimenti

8.2.3.6 adminList (*\$model*, *\$options_view* = `array()`)

Lista dei record del modello.

See also

`self::backOffice()`

Parameters

object	<i>\$model</i>	
array	<i>\$options_view</i>	<p>array associativo di opzioni</p> <ul style="list-style-type: none"> • filter_fields (array): campi sui quali applicare il filtro per la ricerca automatica • filter_join (array): contiene le proprietà degli input form da associare ai campi ai quali viene applicato il filtro; i valori in arrivo da questi input concorrono alla definizione delle condizioni dei campi ai quali sono associati <ul style="list-style-type: none"> – <i>field</i> (string): nome del campo di riferimento; l'input form viene posizionato dopo questo campo – <i>name</i> (string): nome dell'input – <i>label</i> (string): nome della label – <i>data</i> (array): elementi che compongono gli input form radio e select – <i>default</i> (string): valore di default – <i>input</i> (string): tipo di input form, valori validi: radio (default), select – <i>where_clause</i> (string): nome della chiave da passare alle opzioni del metodo addWhereClauses(); per i campi data: <i>operator</i> inoltre contiene le opzioni da passare al metodo clean – <i>value_type</i> (string): tipo di dato (default string) – <i>method</i> (array): default <code>\$this->_request->POST</code> – <i>escape</i> (boolean): default true <p>Esempio:</p> <pre>array('field'=>'date_end', 'label'=>'', 'name'=>'op', 'data'=>array(1=>'<=', 2=>'=', 3=>'>='), 'where_clause'=>'operator')</pre> <ul style="list-style-type: none"> • filter_add (array): contiene le proprietà degli input form che vengono aggiunti come filtro per la ricerca automatica <ul style="list-style-type: none"> – <i>field</i> (string): nome del campo che precede l'input form aggiuntivo nel form di ricerca – <i>name</i> (string): nome dell'input – <i>label</i> (string): nome della label – <i>data</i> (array): elementi che compongono gli input form radio e select – <i>default</i> (string): valore di default – <i>input</i> (string): tipo di input form, valori validi: radio (default), select – <i>filter</i> (string): nome del metodo da richiamare per la condizione aggiuntiva; il metodo dovrà essere creato in una classe che estende <i>adminTable()</i> inoltre contiene le opzioni da passare al metodo clean – <i>value_type</i> (string): tipo di dato (default string) – <i>method</i> (array): default <code>\$this->_request->POST</code> – <i>escape</i> (boolean): default true <p>Esempio:</p> <pre>array('field'=>'date_end', 'label'=>'_("Scaduto")', 'name'=>'expired', 'data'=>array('no'=>'_("no")', 'yes'=>'_("si)'), 'filter'=>'filterWhereExpired'), array('field'=>'date_end', 'label'=>'_("Filiali")', 'name'=>'cod_filiale', 'data'=>\$array_filiali, 'input'=>'select', </pre>

Returns

lista record paginata e ordinabile

8.2.3.7 adminListOrder (\$order_dir, \$table, \$name) [private]

Definisce l'ordinamento della query.

Parameters

string	<i>\$order_dir</i>	
string	<i>\$table</i>	
string	<i>\$name</i>	

Returns

order clause

8.2.3.8 backOffice (\$model_class_name, \$options_view = array(), \$options_form = array(), \$inputs = array())

Gestisce il backoffice completo di un modello (wrapper)

See also

```
self::permission()
self::adminForm()
self::adminDelete()
self::adminList()
```

Parameters

string	<i>\$model_class_name</i>	nome della classe del modello
array	<i>\$options_view</i>	opzioni della vista (comprese le autorizzazioni a visualizzare singoli campi)
array	<i>\$options_form</i>	opzioni del form (comprese le autorizzazioni a mostrare l'input di singoli campi e a salvarli)
array	<i>\$inputs</i>	opzioni degli elementi nel form

Returns

interfaccia di amministrazione

8.2.3.9 clean (\$name, \$options = null) [private]

Ripulisce un input per l'inserimento in database.

Parameters

string	<i>\$name</i>	nome dell'input form
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • value_type (string) • method (array) • escape (boolean)

Returns

valore ripulito

8.2.3.10 `dump($table, $filename, $options = array())` [protected]

Dump di una tabella.

See also

Gino.Db::dump()

Parameters

string	<i>\$table</i>	
string	<i>\$filename</i>	nome del file completo di percorso
array	<i>\$options</i>	array associativo di opzioni

Returns

string (nome del file di dump)

8.2.3.11 `editUrl($add_params = array(), $remove_params = array())` [protected]

Costruisce il percorso per il reindirizzamento.

Parameters

array	<i>\$add_params</i>	elenco parametri da aggiungere al path (Gino.Http.Request::path) (formato chiave=>valore)
array	<i>\$remove_params</i>	elenco parametri da rimuovere dal path (Gino.Http.Request::path)

Returns

url ricostruito

8.2.3.12 `formFilters($model, $options)` [protected]

[Form](#) per filtraggio record.

See also

self::permission()
self::formFiltersAdd()

Parameters

<code>\Gino\↔ Model</code>	<i>\$model</i>	istanza di Gino.Model
array	<i>\$options</i>	autorizzazioni alla visualizzazione dei singoli campi

Returns

form html

8.2.3.13 `formFiltersAdd($filters, $fname, $class_name, $gform)` [private]

Input form dei filtri aggiuntivi.

Parameters

array	<i>\$filters</i>	elenco dei filtri
string	<i>\$fname</i>	nome del campo della tabella al quale far seguire gli eventuali filtri aggiuntivi
string	<i>\$class_name</i>	nome della classe
\Gino\Form	<i>\$gform</i>	istanza di Gino.Form

Returns

elementi del form in html

8.2.3.14 `hidden ($hidden = array())`

Setta la proprietà `_hidden` (campi hidden del form)

Parameters

array	<i>\$hidden</i>	array delle accoppiate nome-valore dei campi hidden non impostati automaticamente
-------	-----------------	---

Returns

void

8.2.3.15 `m2mThroughAction ($m2m_name, $m2m_object, $model, $options = array()) [private]`

Salvataggio dei campi [Gino.ManyToManyThroughField](#).

Il salvataggio di questi tipi di campi avviene in automatico utilizzando la class [Gino.AdminTable](#). Non è gestito dalla classe [Gino.Model](#).

Parameters

string	<i>\$m2m_name</i>	nome del campo ManyToManyThroughField
\Gino\↔ ManyTo↔ Many↔ Through↔ Field	<i>\$m2m_object</i>	istanza della classe di tipo Gino.Field che rappresenta il campo
\Gino\↔ Model	<i>\$model</i>	istanza del model cui appartiene il campo
	<i>\$options</i>	array associativo di opzioni

Returns

risultato operazione, bool o errori

8.2.3.16 `modelAction ($model, $options = array(), $options_element = array())`

Salvataggio dei dati a seguito del submit di un form di inserimento/modifica.

Per recuperare i dati dal form vengono impostati i seguenti parametri di default:

- **formId**, valore generato
- **method**, post
- **validation**, true

- **session_value**, valore generato

Esempio:

```
$id = cleanVar($this->_request->POST, 'id', 'int', '');
$start = cleanVar($this->_request->POST, 'start', 'int', '');

$item = new eventItem($id, $this);
$options_form = array(
    'removeFields'=>null
);
$options_element = array('code'=>array('value_type'=>'int'));

$admin_table = new adminTable();
$admin_table->modelAction($item, $options_form, $options_element);
```

See also

[Gino.Model::save\(\)](#)
[Gino.Build::clean\(\)](#)

Parameters

object	<i>\$model</i>	
array	<i>\$options</i>	<ul style="list-style-type: none"> • opzioni per il recupero dei dati dal form • opzioni per selezionare gli elementi da recuperare dal form <ul style="list-style-type: none"> – removeFields (array): elenco dei campi non presenti nel form – viewFields (array): elenco dei campi presenti nel form • import_file (array): attivare l'importazione di un file (richiama il metodo readFile()) <ul style="list-style-type: none"> – <i>field_import</i> (string): nome del campo del file di importazione – <i>field_verify</i> (array): valori da verificare nel processo di importazione, nel formato array(nome_campo=>valore[,]) – <i>field_log</i> (string): nome del campo del file di log – <i>dump</i> (boolean): per eseguire il dump della tabella prima di importare il file – <i>dump_path</i> (string): percorso del file di dump
array	<i>\$options</i> ↔ <i>element</i>	opzioni per formattare uno o più elementi da inserire nel database

Returns

risultato operazione, bool o errori

8.2.3.17 modelForm (*\$model*, *\$options* = array(), *\$inputs* = array())

Generazione automatica del form di inserimento/modifica di un [Gino.Model](#).

Cicla sulla struttura del modello e per ogni field costruisce l'elemento del form.

L'elemento del form di base lo prende da `$model->structure[nome_campo]->formElement(opzioni);`

Nella costruzione del form vengono impostati i seguenti parametri di default:

- **formId**, valore generato

- **method**, post
- **validation**, true
- **trnsl_table**, il nome della tabella viene recuperato dal modello
- **trnsl_id**, il valore del record id viene recuperato dal modello
- **session_value**, valore generato
- **upload**, viene impostato a TRUE se l'oggetto di un campo del form appartiene almeno a una classe `fileField()` o `imageField()`
- **required**, l'elenco dei campi obbligatori viene costruito controllando il valore della proprietà `$_required` dell'oggetto del campo
- **s_name**, il nome del submit è 'submit'
- **s_value**, il valore del submit è 'salva'

Le opzioni degli elementi input sono formattate nel seguente modo: `nome_campo=>array(opzione=>valore[,...])`

E' possibile rimuovere gli elementi input dalla struttura del form (*removeFields*) oppure selezionare gli elementi da mostrare (*viewFields*).

E' inoltre possibile aggiungere degli elementi input all'interno della struttura del form indicando come chiave il nome del campo prima del quale inserire ogni elemento (*addCell*).

Il campo *instance* non viene mostrato nel form, neanche come campo nascosto.

Esempio:

```
$gform = new Form('', '', '');
$addCell = array(
    'lng'=>$gform->cinput('map_address', 'text', '', array(_("Indirizzo evento"), _("es: torino, via
    mazzini 37)), array("size">40, "maxlength">200, "id">"map_address"))
    .$gform->cinput('map_coord', 'button', _("converti"), '', array("id">"map_coord", "classField">"
    generic", "js">$onclick))
);

$removeFields = array('instance');
if(!$manage_ctg) $removeFields[] = 'ctg';

$options_form = array(
    'formId'=>'eform',
    'f_action'=>$this->_home.">?evt[$this->_instanceName-actionEvent]",
    's_name'=>'submit_action',
    'removeFields'=>$removeFields,
    'viewFields'=>null,
    'addCell'=>$addCell
);

$admin_table = new adminTable($this);
$admin_table->hidden(array('ref'=>$reference));
$form = $admin_table->modelForm(
    $model,
    $options_form,
    array(
        'ctg'=>array('required'=>true),
        'name'=>array('required'=>false),
        'lat'=>array('id'=>'lat', 'trnsl'=>false),
        'lng'=>array('id'=>'lng', 'trnsl'=>false),
        'image'=>array('preview'=>true),
        'description'=>array(
            'widget'=>'editor',
            'notes'=>false,
            'img_preview'=>true,
            'fck_toolbar'=>self::$_fck_toolbar,
            'fck_height'=>100),
        'summary'=>array(
            'maxlength'=>$maxlength_summary,
            'id'=>'summary',
            'rows'=>6,
            'cols'=>55)
    )
);
```

See also

[permission\(\)](#)

Parameters

object array	<i>\$model</i> <i>\$options</i>	<p>oggetto dell'elemento (record da processare)</p> <ul style="list-style-type: none"> • opzioni del form • opzioni per modificare la struttura del form <ul style="list-style-type: none"> – removeFields (array): elenco dei campi da non mostrare nel form – viewFields (array): elenco dei campi da mostrare nel form – addCell (array): elementi html da mostrare nel form in aggiunta agli input form generati dalla struttura. Le chiavi dell'array sono i nomi dei campi che seguono gli elementi aggiuntivi, mentre i valori sono altri array che hanno come chiavi: <ul style="list-style-type: none"> * <i>name</i>, nome dell'elemento da aggiungere (nome dell'input form o altro) * <i>field</i>, codice da aggiungere Riassunto, la struttura di addCell è la seguente: <pre> array('field_table'=>array('name'=>'name_item_add', 'field'=>'content_item_add')) </pre> • permission (array): raggruppa le autorizzazioni ad accedere a una determinata funzione/campo
array	<i>\$inputs</i>	<p>opzioni degli elementi input che vengono passate al metodo formElement() opzioni valide</p> <ul style="list-style-type: none"> • opzioni dei metodi della classe Form() • widget (string): tipo di input • enum (mixed): recupera gli elementi che popolano gli input radio, select, multichexk <ul style="list-style-type: none"> – <i>string</i>, query per recuperare gli elementi (select di due campi) – <i>array</i>, elenco degli elementi (key=>value) • seconds (boolean): mostra i secondi • default (mixed): valore di default (input radio) • checked (boolean): valore selezionato (input checkbox)

Returns

form di inserimento/modifica

8.2.3.18 permission (*\$options*, *\$fname*)

Permessi di modifica dei campo.

Todo Implementare il metodo che restituisce TRUE se l'utente ha il permesso di agire sul campo, FALSE altrimenti.

Parameters

array	<i>\$options</i>	array associativo di opzioni
string	<i>\$fname</i>	nome del campo

Returns

TRUE

8.2.3.19 `readFile ($model, $path_to_file, $options = array())` [protected]

Legge il file e ne importa il contenuto.

Todo Implementare la funzionalità di importazione del file

Parameters

\Gino\↵ Model	<i>\$model</i>	Gino.Model
string	<i>\$path_to_file</i>	
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • verify_items (array): valori da verificare nel processo di importazione, nel formato array(nome_campo=>valore[,]) • dump (boolean): effettua il dump della tabella prima dell'importazione • dump_path (string): percorso del file di dump

Returns

string (log dell'importazione)

8.2.3.20 `restore ($table, $filename, $options = array())` [protected]

Restore di un file.

Todo Implementare la funzionalità di restore di un file

See also

Gino.Db::restore()

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$filename</i>	nome del file da importare
array	<i>\$options</i>	arra

Returns

boolean

8.2.3.21 `setSessionSearch ($model)` [protected]

Setta le variabili di sessione usate per filtrare i record nella lista amministrativa.

Parameters

\Gino\↔ Model	<i>\$model</i>	istanza di Gino.Model
------------------	----------------	---------------------------------------

Returns

void

8.2.3.22 setSessionSearchAdd (*\$model*, *\$filters*) [protected]

Setta le variabili di sessione usate per filtrare i record nella lista amministrativa (riferimento ai filtri non automatici)

Parameters

\Gino\↔ Model	<i>\$model</i>	istanza di Gino.Model
array	<i>\$filters</i>	elenco dei filtri

Returns

void

8.2.4 Field Documentation

8.2.4.1 *\$_allow_insertion* [protected]8.2.4.2 *\$_controller* [protected]8.2.4.3 *\$_db*8.2.4.4 *\$_delete_deny*8.2.4.5 *\$_edit_deny*8.2.4.6 *\$_filter_add* [protected]8.2.4.7 *\$_filter_fields* [protected]8.2.4.8 *\$_filter_join* [protected]8.2.4.9 *\$_form* [protected]8.2.4.10 *\$_hidden*8.2.4.11 *\$_ifp* [protected]8.2.4.12 *\$_list_display* [protected]8.2.4.13 *\$_list_remove*8.2.4.14 *\$_registry* [protected]8.2.4.15 *\$_request*8.2.4.16 *\$_session*

8.2.4.17 \$_view

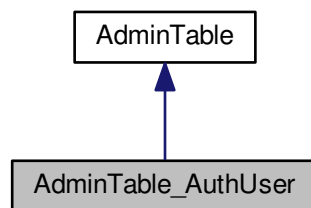
The documentation for this class was generated from the following file:

- lib/classes/[class.AdminTable.php](#)

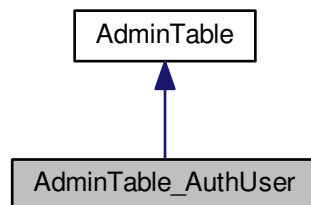
8.3 AdminTable_AuthUser Class Reference

Sovrascrive la classe [AdminTable](#).

Inheritance diagram for AdminTable_AuthUser:



Collaboration diagram for AdminTable_AuthUser:



Public Member Functions

- [modelAction](#) (\$model, \$options=array(), \$options_element=array())

Additional Inherited Members

8.3.1 Detailed Description

Sovrascrive la classe [AdminTable](#).

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.3.2 Member Function Documentation

8.3.2.1 `modelAction ($model, $options = array(), $options_element = array())`

See also

`::modelAction()`

Customizations:

- `username_as_email`
- `user_more_info`
- `aut_password`
- `aut_password_length`
- `pwd_length_min`
- `pwd_length_max`
- `pwd_numeric_number`
- `ldap_auth`
- `ldap_auth_password`

See also

[User::checkEmail\(\)](#)
[User::checkPassword\(\)](#)
[User::checkUsername\(\)](#)
[User::setPassword\(\)](#)
[User::setMoreInfo\(\)](#)

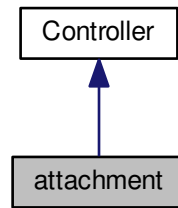
The documentation for this class was generated from the following file:

- [app/auth/class.AdminTable_AuthUser.php](#)

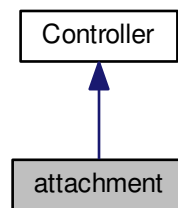
8.4 attachment Class Reference

Classe controller del modulo di gestione di archivi di file categorizzati.

Inheritance diagram for attachment:



Collaboration diagram for attachment:



Public Member Functions

- [__construct](#) ()
- [getDataDir](#) ()
 - Percorso assoluto alla directory dei contenuti.*
- [getDataWWW](#) ()
 - Percorso relativo alla directory dei contenuti.*
- [getPath](#) (\$ctg_id)
 - Percorso della directory di una categoria di allegati.*
- [downloader](#) (\Gino\Http\Request \$request)
 - Download il un allegato.*
- [manageAttachment](#) (\Gino\Http\Request \$request)
 - Interfaccia backoffice del modulo.*
- [editorList](#) (\Gino\Http\Request \$request)
 - Interfaccia per l'inserimento di allegati all'interno dell'editor CKEDITOR.*
- [editorAttachmentList](#) (\Gino\Http\Request \$request)
 - Tabella di allegati con funzionalità per i drag and drop all'interno di CKEDITOR.*
- [jsonImageList](#) ()
 - Lista immagini in formato json Il json restituito è nella forma array(0 => array('image' => image_rel_path, 'folder' => ctg_name), 1 => ...)*

Private Member Functions

- [manageCategory](#) (\$request)
Interfaccia di amministrazione delle categorie.
- [managetItem](#) ()
Interfaccia di amministrazione dei file.

Private Attributes

- [\\$_title](#)
titolo del modulo

Additional Inherited Members

8.4.1 Detailed Description

Classe controller del modulo di gestione di archivi di file categorizzati.

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.4.2 Constructor & Destructor Documentation

8.4.2.1 __construct ()

8.4.3 Member Function Documentation

8.4.3.1 downloader (\Gino\Http\Request \$request)

Download il un allegato.

Exceptions

Gino.Exception ↔ Exception404	se l'allegato non è recuperabile
--	----------------------------------

Parameters

\Gino\Http\Request ↔	<i>\$request</i>	istanza di Gino.Http.Request
--------------------------------------	------------------	--

Returns

[Gino.Http.ResponseFile](#)

8.4.3.2 editorAttachmentList (\Gino\Http\Request \$request)

Tabella di allegati con funzionalità per i drag and drop all'interno di CKEDITOR.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) tabella allegati

8.4.3.3 editorList (\Gino\Http\Request \$request)

Interfaccia per l'inserimento di allegati all'interno dell'editor CKEDITOR.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#), lista allegati

8.4.3.4 getDataDir ()

Percorso assoluto alla directory dei contenuti.

Returns

percorso assoluto

8.4.3.5 getDataWWW ()

Percorso relativo alla directory dei contenuti.

Returns

percorso relativo

8.4.3.6 getPath (\$ctg_id)

Percorso della directory di una categoria di allegati.

Parameters

integer	<i>\$ctg_id</i>	valore id della categoria
---------	-----------------	---------------------------

Returns

percorso

8.4.3.7 jsonImageList ()

Lista immagini in formato json Il json restituito è nella forma array(0 => array('image' => image_rel_path, 'folder' => ctg_name), 1 => ...)

Returns

[Gino.Http.ResponseJson](#)

8.4.3.8 manageAttachment (\Gino\Http\Request \$request)

Interfaccia backoffice del modulo.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) backend di amministrazione del modulo

8.4.3.9 manageCategory (\$request) [private]

Interfaccia di amministrazione delle categorie.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) o html form

8.4.3.10 manageItem () [private]

Interfaccia di amministrazione dei file.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) o html form

8.4.4 Field Documentation

8.4.4.1 \$_title [private]

titolo del modulo

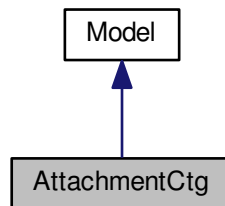
The documentation for this class was generated from the following file:

- [app/attachment/class_attachment.php](#)

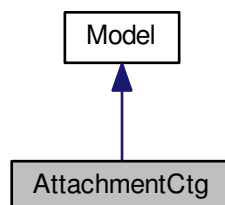
8.5 AttachmentCtg Class Reference

Classe di tipo [Gino.Model](#) che rappresenta una categoria di allegati.

Inheritance diagram for AttachmentCtg:



Collaboration diagram for AttachmentCtg:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [path](#) (\$type)
Percorso alla directory con OS o slash finale.
- [delete](#) ()
Eliminazione modello Elimina tutti gli allegati della categoria.

Static Public Member Functions

- static [columns](#) ()

Struttura dei campi della tabella di un modello.

- static `getForSelect` (\$options=array())

Array di categorie nella forma id=>name Utilizzato per popolare un input select.

Static Public Attributes

- static `$table` = 'attachment_ctg'
- static `$columns`

Additional Inherited Members

8.5.1 Detailed Description

Classe di tipo `Gino.Model` che rappresenta una categoria di allegati.

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.5.2 Constructor & Destructor Documentation

8.5.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<code>\$id</code>	valore ID del record
---------	-------------------	----------------------

Returns

istanza di `Gino.App.Attachment.AttachmentCtg`

8.5.3 Member Function Documentation

8.5.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

nome categoria

8.5.3.2 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.5.3.3 delete ()

Eliminazione modello Elimina tutti gli allegati della categoria.

Returns

TRUE

8.5.3.4 static getForSelect (\$options = array()) [static]

Array di categorie nella forma id=>name Utilizzato per popolare un input select.

Parameters

array	\$options	array associativo di opzioni (where, order e limit)
-------	-----------	---

Returns

array associativo id=>name

8.5.3.5 path (\$type)

Percorso alla directory con OS o slash finale.

Parameters

string	\$type	tipo di percorso: <ul style="list-style-type: none"> • abs: assoluto • rel: relativo alla DOCUMENT ROOT • view: relativo alla ROOT • url: url assoluto
--------	--------	--

Returns

percorso directory

8.5.4 Field Documentation

8.5.4.1 \$columns [static]

8.5.4.2 \$table = 'attachment_ctg' [static]

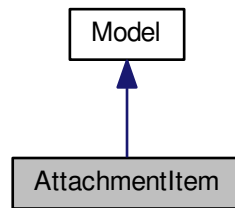
The documentation for this class was generated from the following file:

- app/attachment/[class.AttachmentCtg.php](#)

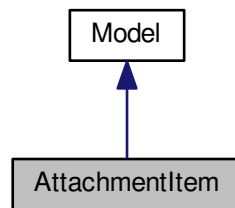
8.6 AttachmentItem Class Reference

Classe di tipo [Gino.Model](#) che rappresenta una singolo allegato.

Inheritance diagram for AttachmentItem:



Collaboration diagram for AttachmentItem:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa del modello.
- [type](#) ()
Tipologia di allegato (img, xls, doc, pdf, other)
- [pathView](#) ()
Percorso dell'allegato relativo alla ROOT.
- [pathDownload](#) ()
Percorso dell'allegato per il download.
- [path](#) (\$type)
Percorso dell'allegato.
- [previewLink](#) (\$label= 'filename')
Link al preview dell'allegato.
- [previewAdminList](#) ()

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.
- static [deleteFromCtg](#) (\$ctg_id)
Elimina tutti gli allegati di una categoria.

Static Public Attributes

- static [\\$table](#) = 'attachment'
tabella del modello
- static [\\$columns](#)

Static Protected Member Functions

- static [properties](#) (\$model, \$controller)

Private Attributes

- [\\$_img_extension](#)
estensioni associate a icone immagini
- [\\$_xls_extension](#)
estensioni associate a icona spreadsheets
- [\\$_pdf_extension](#)
estensioni associate a icona pdf
- [\\$_doc_extension](#)
estensioni associate a icona doc

Additional Inherited Members

8.6.1 Detailed Description

Classe di tipo [Gino.Model](#) che rappresenta una singolo allegato.

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.6.2 Constructor & Destructor Documentation

8.6.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.App.Attachment.AttachmentItem](#)

8.6.3 Member Function Documentation

8.6.3.1 __toString ()

Rappresentazione a stringa del modello.

Returns

nome file

8.6.3.2 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.6.3.3 static deleteFromCtg (\$ctg_id) [static]

Elimina tutti gli allegati di una categoria.

Parameters

int	<i>\$ctg_id</i>	id della categoria
-----	-----------------	--------------------

Returns

TRUE

8.6.3.4 path (\$type)

Percorso dell'allegato.

Parameters

string	<i>\$type</i>	tipo di percorso: <ul style="list-style-type: none">• abs: assoluto• rel: relativo alla DOCUMENT ROOT• view: relativo alla ROOT• url: url assoluto• download: url relativo per il download
--------	---------------	--

Returns

string il percorso dell'allegato

8.6.3.5 pathDownload ()

Percorso dell'allegato per il download.

Returns

string il percorso dell'allegato

8.6.3.6 pathView ()

Percorso dell'allegato relativo alla ROOT.

Returns

string il percorso dell'allegato

8.6.3.7 previewAdminList ()

8.6.3.8 previewLink (*\$label* = 'filename')

Link al preview dell'allegato.

Lightbox per le immagini e download per altri tipi di file

See also

[path\(\)](#) string *\$label* etichetta da mostrare nel link, *i* possibili valori sono filename | [path](#)

Returns

link preview

8.6.3.9 static properties (*\$model*, *\$controller*) [static],[protected]

See also

[Gino.Model::properties\(\)](#)

8.6.3.10 type ()

Tipologia di allegato (img, xls, doc, pdf, other)

Returns

string tipologia dell'allegato (img, xls, doc, pdf, other)

8.6.4 Field Documentation

8.6.4.1 *\$_doc_extension* [private]

estensioni associate a icona doc

8.6.4.2 `$img_extension` [private]

estensioni associate a icone immagini

8.6.4.3 `$pdf_extension` [private]

estensioni associate a icona pdf

8.6.4.4 `$xls_extension` [private]

estensioni associate a icona spreadsheets

8.6.4.5 `$columns` [static]

8.6.4.6 `$table = 'attachment'` [static]

tabella del modello

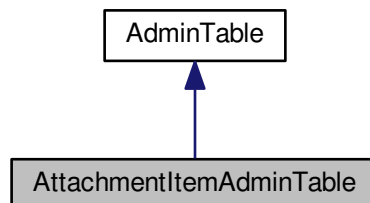
The documentation for this class was generated from the following file:

- [app/attachment/class.AttachmentItem.php](#)

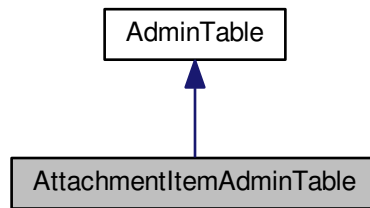
8.7 AttachmentItemAdminTable Class Reference

Estende la classe [Gino.AdminTable](#) per gestire il salvataggio dell'allegato quando viene modificata la directory (categoria)

Inheritance diagram for AttachmentItemAdminTable:



Collaboration diagram for AttachmentItemAdminTable:



Public Member Functions

- [modelAction](#) ([\\$model](#), [\\$options=array\(\)](#), [\\$options_element=array\(\)](#))
rispetto al parent consente di salvare il file all'interno della directory selezionata

Additional Inherited Members

8.7.1 Detailed Description

Estende la classe [Gino.AdminTable](#) per gestire il salvataggio dell'allegato quando viene modificata la directory (categoria)

See also

[Gino.AdminTable](#)

8.7.2 Member Function Documentation

8.7.2.1 [modelAction](#) ([\\$model](#), [\\$options = array\(\)](#), [\\$options_element = array\(\)](#))

rispetto al parent consente di salvare il file all'interno della directory selezionata

See also

[Gino.AdminTable::modelAction\(\)](#)

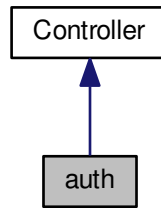
The documentation for this class was generated from the following file:

- [app/attachment/class.AttachmentItemAdminTable.php](#)

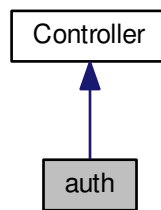
8.8 auth Class Reference

Classe di tipo [Gino.Controller](#) per la gestione degli utenti, gruppi e permessi.

Inheritance diagram for auth:



Collaboration diagram for auth:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [getBasePath](#) (\$path= 'abs')
Percorso base della directory dei contenuti.
- [getAddPath](#) (\$id)
Percorso della directory dei contenuti (una directory per ogni utente)
- [checkAuthenticationUser](#) (\$username, \$password)
Verifica utente/password nel processo di autenticazione.
- [activateProfile](#) (\Gino\Http\Request \$request)
Attivazione di altri profili per utente già registrato ed attivo.
- [registration](#) (\Gino\Http\Request \$request)
Registrazione utente.
- [registrationResult](#) (\Gino\Http\Request \$request)
Vista risultato registrazione.
- [confirmRegistration](#) (\Gino\Http\Request \$request)
Conferma indirizzo email registrazione.
- [activateRegistrationUser](#) (\Gino\Http\Request \$request)
Creazione ed attivazione di un utente a seguito di richiesta di registrazione.

- [profile](#) (\Gino\Http\Request \$request)
Vista profilo utente.
- [deleteAccount](#) (\Gino\Http\Request \$request)
Eliminazione account.
- [manageAuth](#) (\Gino\Http\Request \$request)
Interfaccia di amministrazione modulo.
- [checkUsername](#) (\Gino\Http\Request \$request)
Controlla se lo username è disponibile.
- [checkUsernameJson](#) (\Gino\Http\Request \$request)
Controlla se lo username è disponibile e ritorna un json.
- [checkPassowrdJson](#) (\Gino\Http\Request \$request)
Controlla che la password abbia le caratteristiche richieste.
- [actionJoinUserPermission](#) (\Gino\Http\Request \$request)
Processa il form di associazione degli utenti ai permessi.
- [actionJoinGroupPermission](#) (\Gino\Http\Request \$request)
Processa il form di associazione gruppo-permessi.
- [login](#) (\Gino\Http\Request \$request)
Pagina di login.
- [dataRecovery](#) (\Gino\Http\Request \$request)
Pagina di recupero username e password.
- [dataRecoverySuccess](#) (\Gino\Http\Request \$request)

Static Public Member Functions

- static [outputFunctions](#) ()
Definizione dei metodi pubblici che forniscono un output per il front-end.
- static [getClassElements](#) ()
Restituisce alcune proprietà della classe.

Data Fields

- [\\$_optionsLabels](#)
- [\\$_user_view](#)
- [\\$_aut_pwd_length](#)
- [\\$_pwd_length_min](#)
- [\\$_pwd_length_max](#)
- [\\$_pwd_numeric_number](#)
- [\\$_ldap_auth_only](#)
- [\\$_ldap_single_user](#)
- [\\$_ldap_auth_password](#)
- [\\$_other_field1](#)
- [\\$_other_field2](#)
- [\\$_other_field3](#)
- [\\$_label_field2](#)
- [\\$_label_field3](#)

Private Member Functions

- [getCheckUser](#) (\$username, \$options=array())
Verifica la validità di un utente.
- [createAndActivateUser](#) (\Gino\App\Auth\RegistrationRequest \$registration_request)
Crea ed attiva un utente, invia una mail di attivazione.
- [manageRegistrationRequest](#) ()
Interfaccia di amministrazione richieste di registrazione.
- [manageRegistrationProfile](#) ()
Interfaccia di amministrazione profili di registrazione.
- [manageUser](#) (\Gino\Http\Request \$request)
Interfaccia di amministrazione utenti.
- [passwordRules](#) (\$id=null)
Descrizione delle regole alle quali è sottoposta la password.
- [passwordStrength](#) (\$password)
Calcola la robustezza di una password, da 0 a 10.
- [changePassword](#) (\Gino\Http\Request \$request)
Interfaccia di sostituzione della password.
- [manageGroup](#) ()
Interfaccia di amministrazione dei gruppi.
- [managePermission](#) ()
Interfaccia di amministrazione dei permessi.
- [returnJoinLink](#) (\$block, \$option, \$ref_id)
Reindirizza le operazione di join tra utenti/gruppi/permessi.
- [joinUserPermission](#) (\Gino\Http\Request \$request)
Interfaccia di associazione utente-permessi.
- [joinGroupPermission](#) (\$request)
Form di associazione gruppo-permessi.
- [formPermission](#) (\$obj_form, \$checked=array())
Imposta il multichexbox sui permessi.
- [formGroup](#) (\$obj_form, \$checked=array())
Imposta il multichexbox sui gruppi.

Private Attributes

- [\\$_options](#)
- [\\$_users_for_page](#)
- [\\$_user_more](#)
- [\\$_username_as_email](#)
- [\\$_aut_pwd](#)
- [\\$_ldap_auth](#)
- [\\$_label_field1](#)

Additional Inherited Members

8.8.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione degli utenti, gruppi e permessi.

I permessi delle applicazioni sono definiti nella tabella *auth_permission*. Il campo *admin* indica se il permesso necessita dell'accesso all'area amministrativa.

Ogni utente può essere associato a un permesso definito nella tabella *auth_permission*, e tale associazione viene

registrata nella tabella `auth_user_perm`.

La tabella `auth_user_perm` registra il valore id dell'utente, del permesso e dell'istanza relativa all'applicazione del permesso.

Questo implica che nell'interfaccia di associazione utente/permessi occorre mostrare i permessi relativi a ogni applicazione (classe) per tutte le istanze presenti.

I gruppi sono definiti nella tabella `auth_group`. I gruppi possono essere associati ai permessi e alle istanze (`auth_group_perm`) e gli utenti ai gruppi (`auth_group_user`).

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.8.2 Constructor & Destructor Documentation

8.8.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.App.Auth.auth](#)

8.8.3 Member Function Documentation

8.8.3.1 `actionJoinGroupPermission (\Gino\Http\Request $request)`

Processa il form di associazione gruppo-permessi.

Parametri POST:

- id (integer), valore ID del gruppo
- perm (array), permessi selezionati

Parameters

<code>\Gino\Http\Request</code>	<code>\$request</code>	
---------------------------------	------------------------	--

Returns

[Gino.Http.Redirect](#)

8.8.3.2 `actionJoinUserPermission (\Gino\Http\Request $request)`

Processa il form di associazione degli utenti ai permessi.

parametri post:

- id (integer), valore id dell'utente
- perm (array), permessi selezionati

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.8.3.3 activateProfile (\Gino\Http\Request \$request)

Attivazione di altri profili per utente già registrato ed attivo.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

8.8.3.4 activateRegistrationUser (\Gino\Http\Request \$request)

Creazione ed attivazione di un utente a seguito di richiesta di registrazione.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.8.3.5 changePassword (\Gino\Http\Request \$request) [private]

Interfaccia di sostituzione della password.

Parametri GET (per il form):

- ref (integer), valore ID dell'utente
- c (integer), riporta se la password è stata correttamente aggiornata

Parametri POST (per l'action del form):

- id (integer), valore ID dell'utente

See also

[User::savePassword\(\)](#)
[User::formPassword\(\)](#)
[passwordRules\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
--	------------------	--

Returns

html

8.8.3.6 checkAuthenticationUser (*\$username*, *\$password*)

Verifica utente/password nel processo di autenticazione.

See also

[Ldap::getCheckUser\(\)](#)

Parameters

string	<i>\$username</i>	
string	<i>\$password</i>	

Returns

user object

8.8.3.7 checkPassowrdJson ([\Gino\Http\Request](#) *\$request*)

Controlla che la password abbia le caratteristiche richieste.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
--	------------------	--

Returns

[Gino.Http.ResponseJson](#)

8.8.3.8 checkUsername ([\Gino\Http\Request](#) *\$request*)

Controlla se lo username è disponibile.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
--	------------------	--

Returns

[Gino.Http.Response](#)

8.8.3.9 checkUsernameJson (\Gino\Http\Request \$request)

Controlla se lo username è disponibile e ritorna un json.

Parameters

\Gino\← Http\← Request	\$request	
------------------------------	-----------	--

Returns

[Gino.Http.ResponseJson](#)

8.8.3.10 confirmRegistration (\Gino\Http\Request \$request)

Conferma indirizzo email registrazione.

Parameters

\Gino\← Http\← Request	\$request	
------------------------------	-----------	--

Returns

[Gino.Http.Response](#)

8.8.3.11 createAndActivateUser (\Gino\App\Auth\RegistrationRequest \$registration_request) [private]

Crea ed attiva un utente, invia una mail di attivazione.

Parameters

\Gino\← App\← Auth\← Registration← Request	\$registration_← request	richiesta di registrazione
--	-----------------------------	----------------------------

Returns

bool, risultato

8.8.3.12 dataRecovery (\Gino\Http\Request \$request)

Pagina di recupero username e password.

Parameters

\Gino\← Http\← Request	\$request	
------------------------------	-----------	--

Returns

[Gino.Http.Response](#)

8.8.3.13 `dataRecoverySuccess (\Gino\Http\Request $request)`

8.8.3.14 `deleteAccount (\Gino\Http\Request $request)`

Eliminazione account.

Parameters

\Gino\↵ Http\↵ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.8.3.15 `formGroup ($obj_form, $checked = array()) [private]`

Imposta il multichexbox sui gruppi.

See also

[Gino.App.Auth.Group::getList\(\)](#)

[Gino.Form::multipleCheckbox\(\)](#)

Parameters

\Gino\Form	<i>\$obj_form</i>	istanza di Gino.Form
array	<i>\$checked</i>	array di id di permessi selezionati

Returns

html, multichex

8.8.3.16 `formPermission ($obj_form, $checked = array()) [private]`

Imposta il multichexbox sui permessi.

Returns

html, multichex

8.8.3.17 `getAddPath ($id)`

Percorso della directory dei contenuti (una directory per ogni utente)

Parameters

integer	<i>\$id</i>	valore ID dell'utente
---------	-------------	-----------------------

Returns

path directory

8.8.3.18 `getBasePath ($path = 'abs')`

Percorso base della directory dei contenuti.

Parameters

string	<i>\$path</i>	tipo di percorso (default abs) <ul style="list-style-type: none">• abs, assoluto• rel, relativo
--------	---------------	--

Returns

percorso

8.8.3.19 `getCheckUser ($username, $options = array())` [private]

Verifica la validità di un utente.

See also

[User::getFromUserPwd\(\)](#)

Parameters

string	<i>\$username</i>	
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• user_pwd (string): password per autenticazione di gino• auth_ldap (boolean): indica se è stata effettuata l'autenticazione ldap

Returns

user object

8.8.3.20 `static getClassElements ()` [static]

Restituisce alcune proprietà della classe.

Returns

array associativo contenente le tabelle, viste e struttura directory contenuti

8.8.3.21 `joinGroupPermission ($request)` [private]

[Form](#) di associazione gruppo-permessi.

parametri get:

- ref (integer), valore id del gruppo

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

html, form

8.8.3.22 joinUserPermission (\Gino\Http\Request \$request) [private]

Interfaccia di associazione utente-permessi.

Parametri GET:

- ref (integer), valore ID dell'utente

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

html

8.8.3.23 login (\Gino\Http\Request \$request)

Pagina di login.

See also[Gino.Access::Authentication\(\)](#)**Returns**[Gino.Http.Response](#)**8.8.3.24 manageAuth (\Gino\Http\Request \$request)**

Interfaccia di amministrazione modulo.

Parameters

--	--

8.8.3.25 manageGroup () [private]

Interfaccia di amministrazione dei gruppi.

Returns[Gino.Http.Redirect](#) oppure html

8.8.3.26 `managePermission () [private]`

Interfaccia di amministrazione dei permessi.

Returns

[Gino.Http.Redirect](#) oppure html

8.8.3.27 `manageRegistrationProfile () [private]`

Interfaccia di amministrazione profili di registrazione.

Returns

html oppure

8.8.3.28 `manageRegistrationRequest () [private]`

Interfaccia di amministrazione richieste di registrazione.

Returns

html oppure

8.8.3.29 `manageUser (\Gino\Http\Request $request) [private]`

Interfaccia di amministrazione utenti.

Parameters

--	--

8.8.3.30 `static outputFunctions () [static]`

Definizione dei metodi pubblici che forniscono un output per il front-end.

Questo metodo viene letto dal motore di generazione dei layout (prende i metodi non presenti nel file ini) e dal motore di generazione di voci di menu (presenti nel file ini) per presentare una lista di output associati all'istanza di classe.

Returns

array associativo metodi pubblici metodo => array('label' => label, 'permissions' => permissions)

8.8.3.31 `passwordRules ($id = null) [private]`

Descrizione delle regole alle quali è sottoposta la password.

Parameters

integer	<i>\$id</i>	valore ID dell'utente
---------	-------------	-----------------------

Returns

string, regole password

8.8.3.32 `passwordStrength ($password) [private]`

Calcola la robustezza di una password, da 0 a 10.

Parameters

string	<i>\$password</i>	
--------	-------------------	--

Returns

int, robustezza password da 0 a 10

8.8.3.33 profile (\Gino\Http\Request \$request)

Vista profilo utente.

Tutti gli utenti hanno un profilo dove poter modificare la password ed alcune informazioni personali. Solamente gli utenti creati a seguito di procedura registrazione possono associarsi ad altri profili di registrazione

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

8.8.3.34 registration (\Gino\Http\Request \$request)

Registrazione utente.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

8.8.3.35 registrationResult (\Gino\Http\Request \$request)

Vista risultato registrazione.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

8.8.3.36 returnJoinLink (\$block, \$option, \$ref_id) [private]

Reindirizza le operazione di join tra utenti/gruppi/permessi.

Parameters

string	<i>\$block</i>	
string	<i>\$option</i>	
integer	<i>\$ref_id</i>	valore ID del riferimento (utente o gruppo)

Returns

redirect

8.8.4 Field Documentation

8.8.4.1 `$_aut_pwd` [private]

8.8.4.2 `$_aut_pwd_length`

8.8.4.3 `$_label_field1` [private]

8.8.4.4 `$_label_field2`

8.8.4.5 `$_label_field3`

8.8.4.6 `$_ldap_auth` [private]

8.8.4.7 `$_ldap_auth_only`

8.8.4.8 `$_ldap_auth_password`

8.8.4.9 `$_ldap_single_user`

8.8.4.10 `$_options` [private]

8.8.4.11 `$_optionsLabels`

8.8.4.12 `$_other_field1`

8.8.4.13 `$_other_field2`

8.8.4.14 `$_other_field3`

8.8.4.15 `$_pwd_length_max`

8.8.4.16 `$_pwd_length_min`

8.8.4.17 `$_pwd_numeric_number`

8.8.4.18 `$_user_more` [private]

8.8.4.19 `$_user_view`

8.8.4.20 `$_username_as_email` [private]

8.8.4.21 `$_users_for_page` [private]

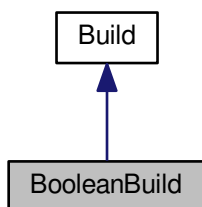
The documentation for this class was generated from the following file:

- app/auth/[class_auth.php](#)

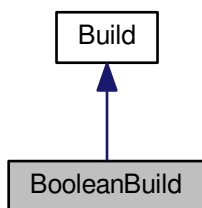
8.9 BooleanBuild Class Reference

Gestisce i campi di tipo BOOLEAN.

Inheritance diagram for BooleanBuild:



Collaboration diagram for BooleanBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getChoice](#) ()
Getter della proprietà enum.
- [setChoice](#) (\$value)
Setter della proprietà enum.
- [retrieveValue](#) ()

Protected Attributes

- [\\$_choice](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.9.1 Detailed Description

Gestisce i campi di tipo BOOLEAN.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.9.2 Constructor & Destructor Documentation

8.9.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.9.3 Member Function Documentation

8.9.3.1 `getChoice ()`

Getter della proprietà enum.

Returns

proprietà enum

8.9.3.2 `retrieveValue ()`

See also

[Gino.Build::retrieveValue\(\)](#)

8.9.3.3 `setChoice ($value)`

Setter della proprietà enum.

Parameters

array	<i>\$value</i>	
-------	----------------	--

Returns

void

8.9.4 Field Documentation

8.9.4.1 \$_choice [protected]

Proprietà dei campi specifiche del tipo di campo.

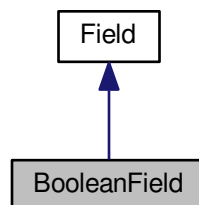
The documentation for this class was generated from the following file:

- lib/classes/build/[class.BooleanBuild.php](#)

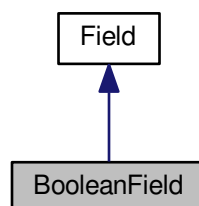
8.10 BooleanField Class Reference

Rappresenta campi di tipo BOOLEAN.

Inheritance diagram for BooleanField:



Collaboration diagram for BooleanField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)
- [valueToDb](#) (\$value)

Protected Attributes

- [\\$_choice](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.10.1 Detailed Description

Rappresenta campi di tipo BOOLEAN.

Tipologie di input associabili: radio

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.10.2 Constructor & Destructor Documentation

8.10.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – choice (array): elenco degli elementi di scelta
-------	------------------	---

Returns

istanza di [Gino.BooleanField](#)

8.10.3 Member Function Documentation

8.10.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.10.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or boolean

8.10.3.3 `valueToDb ($value)`

See also

[Gino.Field::valueToDb\(\)](#)

Returns

null or integer[1|0]

8.10.4 Field Documentation

8.10.4.1 `$_choice` [`protected`]

Proprietà dei campi specifiche del tipo di campo.

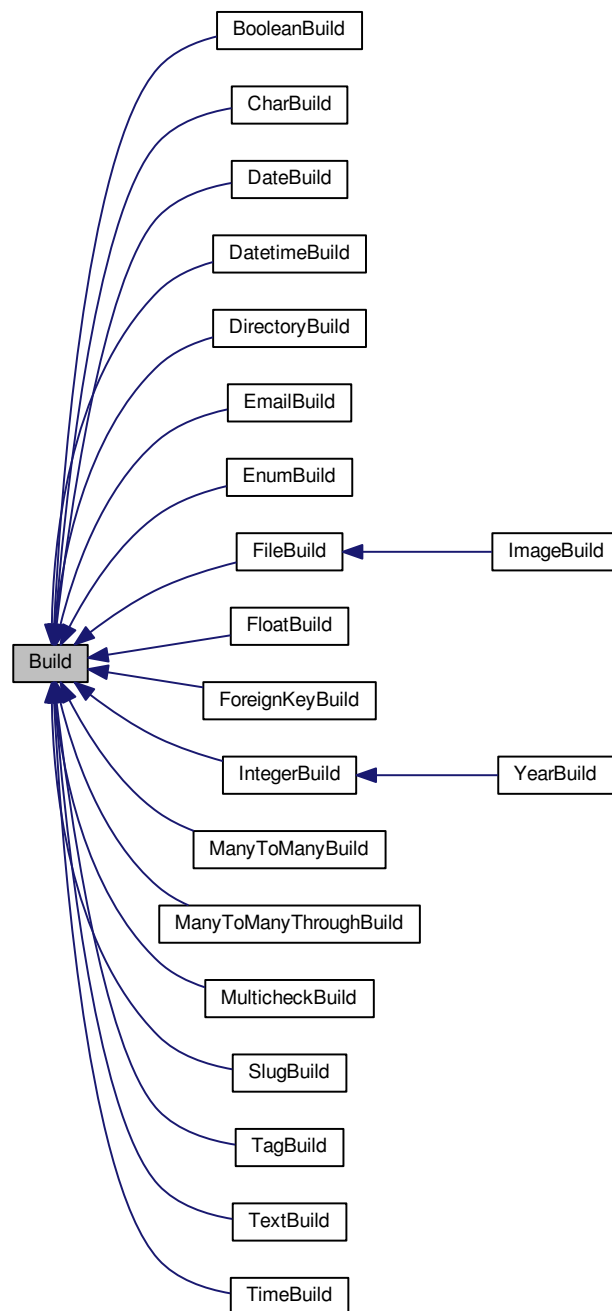
The documentation for this class was generated from the following file:

- [lib/classes/fields/class.BooleanField.php](#)

8.11 Build Class Reference

Gestisce i campi del modello.

Inheritance diagram for Build:



Public Member Functions

- `__construct` (\$options)
Costruttore.
- `__toString` ()
Rappresentazione a stringa dell'oggetto.
- `canBeOrdered` ()

Indica se il campo può essere utilizzato come ordinamento nella lista della sezione amministrativa.

- `getName ()`
Getter della proprietà name.
- `setName ($value)`
Setter della proprietà name.
- `getValue ()`
Getter della proprietà value.
- `setValue ($value)`
Setter della proprietà value.
- `getTable ()`
Getter della proprietà table.
- `setTable ($value)`
Setter della proprietà table.
- `getViewInput ()`
Getter della proprietà value.
- `setViewInput ($value)`
Setter della proprietà value.
- `formElement (\Gino\Form $form, $options)`
Stampa un elemento del form facendo riferimento al valore della chiave widget.
- `formFilter (\Gino\Form $form, $options)`
Stampa un elemento del form di filtri area amministrativa.
- `filterWhereClause ($value, $options=array())`
Definisce la condizione WHERE per il campo.
- `cleanFilter ($options)`
Ripulisce un input usato come filtro in area amministrativa.
- `clean ($options=null, $id=null)`
Ripulisce un input per l'inserimento del valore in database.
- `retrieveValue ()`
Valore del campo predisposto per l'output html.

Data Fields

- `$_label`
- `$_default`
- `$_length`
- `$_auto_increment`
- `$_primary_key`
- `$_unique_key`
- `$_required`
- `$_widget`
- `$_value_type`
- `$_int_digits`
- `$_decimal_digits`

Protected Attributes

- `$_name`
Proprietà dei campi.
- `$_model`
- `$_field_object`
- `$_table`
- `$_view_input`
- `$_value`

Private Member Functions

- [properties \(\)](#)

Elenco delle proprietà generali e specifiche di un campo.

Private Attributes

- [\\$_options](#)

8.11.1 Detailed Description

Gestisce i campi del modello.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.11.2 Constructor & Destructor Documentation

8.11.2.1 `__construct ($options)`

Costruttore.

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo di una tabella opzioni delle colonne (caratteristiche del tipo di campo) opzioni del modello <ul style="list-style-type: none"> • model (object): • value (mixed):
-------	------------------	---

8.11.3 Member Function Documentation

8.11.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

valore del campo

8.11.3.2 `canBeOrdered ()`

Indica se il campo può essere utilizzato come ordinamento nella lista della sezione amministrativa.

Returns

TRUE se puo' essere utilizzato per l'ordinamento, FALSE altrimenti

8.11.3.3 `clean ($options = null, $id = null)`

Ripulisce un input per l'inserimento del valore in database.

See also

[Gino.cleanVar\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • value_type (string): tipo di valore • method (array): metodo di recupero degli elementi del form • escape (boolean): evita che venga eseguito il <code>mysql_real_escape_string</code> sul valore del campo
integer	<i>\$id</i>	valore id del record

Returns

valore ripulito dell'input

8.11.3.4 `cleanFilter ($options)`

Ripulisce un input usato come filtro in area amministrativa.

Parameters

<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • escape (boolean): evita che venga eseguito il <code>mysql_real_escape_string</code> sul valore del campo
------------------	--

Returns

input ripulito

8.11.3.5 `filterWhereClause ($value, $options = array())`

Definisce la condizione WHERE per il campo.

Parameters

mixed	<i>\$value</i>	
array	<i>\$options</i>	array associativo di opzioni specifiche dei tipi di campo

Returns

where clause

8.11.3.6 `formElement (\Gino\Form $form, $options)`

Stampa un elemento del form facendo riferimento al valore della chiave *widget*.

Nella chiamata del form occorre definire la chiave *widget* nell'array degli elementi input.

Nel caso in cui la chiave *widget* non sia definita, verrà presa la chiave di default specificata nelle proprietà del modello.

Esempio

```
array(
  'ctg'=>array('required'=>true),
  'field_text1'=>array(
    'widget'=>'editor',
    'notes'=>false,
    'img_preview'=>false,
    'fck_height'=>100),
  'field_text2'=>array('maxlength'=>$maxlength_summary, 'id'=>'summary', 'rows'=>6, 'cols'=>55)
)
```

See also

[Gino.Widget::printInputForm\(\)](#)

Parameters

\Gino\Form	<i>\$form</i>	istanza di Gino.Form
array	<i>\$options</i>	<p>opzioni dell'elemento del form</p> <ul style="list-style-type: none"> • opzioni dei metodi della classe Form • opzioni che sovrascrivono le impostazioni del campo/modello <p>– widget (string): tipo di input form; può assumere uno dei seguenti valori</p> <ul style="list-style-type: none"> * <i>hidden</i> * <i>constant</i> * <i>select</i> * <i>radio</i> * <i>checkbox</i> * <i>multicheck</i> * <i>editor</i> * <i>textarea</i> * <i>float</i> * <i>date</i> * <i>datetime</i> * <i>time</i> * <i>password</i> * <i>file</i> * <i>image</i> * <i>email</i> * <i>unit</i> <p>– required (boolean): campo obbligatorio</p>

Returns

controllo del campo, html

8.11.3.7 formFilter (\Gino\Form \$form, \$options)

Stampa un elemento del form di filtri area amministrativa.

Parameters

\Gino\Form	<i>\$form</i>	istanza di Gino\Form
array	<i>\$options</i>	<ul style="list-style-type: none">• default (mixed): valore di default

Returns

controllo del campo, html

8.11.3.8 `getName ()`

Getter della proprietà name.

Returns

nome del campo

8.11.3.9 `getTable ()`

Getter della proprietà table.

Returns

nome della tabella

8.11.3.10 `getValue ()`

Getter della proprietà value.

Returns

valore del campo

8.11.3.11 `getViewInput ()`

Getter della proprietà value.

Returns

valore del campo

8.11.3.12 `properties ()` [*private*]

Elenco delle proprietà generali e specifiche di un campo.

Returns

array

8.11.3.13 `retrieveValue ()`

Valore del campo predisposto per l'output html.

Parameters

mixed	<i>\$value</i>	
-------	----------------	--

Returns

mixed

8.11.3.14 setName (*\$value*)

Setter della proprietà name.

Parameters

mixed	<i>\$value</i>	
-------	----------------	--

Returns

void

8.11.3.15 setTable (*\$value*)

Setter della proprietà table.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.11.3.16 setValue (*\$value*)

Setter della proprietà value.

Parameters

mixed	<i>\$value</i>	
-------	----------------	--

Returns

void

8.11.3.17 setViewInput (*\$value*)

Setter della proprietà value.

Parameters

boolean	<i>\$value</i>	
---------	----------------	--

Returns

void

8.11.4 Field Documentation

8.11.4.1 `$_auto_increment`

8.11.4.2 `$_decimal_digits`

8.11.4.3 `$_default`

8.11.4.4 `$_field_object` `[protected]`

8.11.4.5 `$_int_digits`

8.11.4.6 `$_label`

8.11.4.7 `$_lenght`

8.11.4.8 `$_model` `[protected]`

8.11.4.9 `$_name` `[protected]`

Proprietà dei campi.

8.11.4.10 `$_options` `[private]`

8.11.4.11 `$_primary_key`

8.11.4.12 `$_required`

8.11.4.13 `$_table` `[protected]`

8.11.4.14 `$_unique_key`

8.11.4.15 `$_value` `[protected]`

8.11.4.16 `$_value_type`

8.11.4.17 `$_view_input` `[protected]`

8.11.4.18 `$_widget`

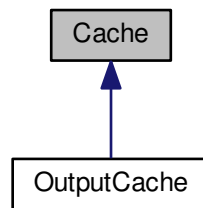
The documentation for this class was generated from the following file:

- `lib/classes/build/class.Build.php`

8.12 Cache Class Reference

Classe che viene estesa da [Gino.OutputCache\(\)](#) e [Gino.DataCache\(\)](#)

Inheritance diagram for Cache:



Public Member Functions

- `__construct ()`
Costruttore.
- `delete ($grp, $id)`
Elimina il file di cache corrispondente al gruppo e id dati.

Data Fields

- `$_fld`
- `$_prefix`
- `$id`
- `$_tc`

Protected Member Functions

- `write ($data)`
Scrive i dati sul file `self::getFilename()`
- `read ()`
Legge il contenuto del file `self::getFilename()`
- `getFilename ()`
Nome file di cache.
- `isCached ()`
Controlla se il file di cache esiste ed è valido.

Protected Attributes

- `$_registry`
- `$_ds`
- `$_grp`
- `$_enabled`

8.12.1 Detailed Description

Classe che viene estesa da [Gino.OutputCache\(\)](#) e [Gino.DataCache\(\)](#)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.12.2 Constructor & Destructor Documentation

8.12.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.Cache](#)

8.12.3 Member Function Documentation

8.12.3.1 `delete ($grp, $id)`

Elimina il file di cache corrispondente al gruppo e id dati.

Parameters

string	<i>\$grp</i>	gruppo
string	<i>\$id</i>	

Returns

risultato operazione, bool

8.12.3.2 `getFilename ()` [protected]

Nome file di cache.

Returns

nome file

8.12.3.3 `isCached ()` [protected]

Controlla se il file di cache esiste ed è valido.

Returns

TRUE se il file di cache è presente e valido, FALSE altrimenti (e lo elimina se presente)

8.12.3.4 `read ()` [protected]

Legge il contenuto del file `self::getFilename()`

See also

`self::getFilename()`

Returns

contenuto file

8.12.3.5 `write ($data)` [protected]

Scrive i dati sul file `self::getFilename()`

See also

`self::getFilename()`

Parameters

string	<i>\$data</i>	
--------	---------------	--

Returns

void

8.12.4 Field Documentation

8.12.4.1 `$_ds` [protected]

8.12.4.2 `$_enabled` [protected]

8.12.4.3 `$_fld`

8.12.4.4 `$_grp` [protected]

8.12.4.5 `$_prefix`

8.12.4.6 `$_registry` [protected]

8.12.4.7 `$_tc`

8.12.4.8 `$id`

The documentation for this class was generated from the following file:

- [lib/classes/class.Cache.php](#)

8.13 Captcha Class Reference

Classe per la generazione di immagini captcha.

Public Member Functions

- `__construct` (\$name, \$opts=null)
Costruttore.
- `render` (\$opts=null)
Renderizza l'immagine e l'input per l'inserimento del codice.
- `check` ()
Controlla che il codice inserito dall'utente sia corretto.

Private Member Functions

- `checkRequirements` ()
Verifica se l'immagine captcha può essere generata.
- `generateStrings` ()
Genera due stringhe alfanumeriche random.
- `hex2RGB` (\$hexStr)
Converte un colore da formato hex a formato array rgb.

Private Attributes

- `$_name`
name of the captcha field
- `$_width`
captcha image width
- `$_height`
captcha image height
- `$_font_file`
absolute path of the font ttf file
- `$_letters`
letters allowed for the captcha
- `$_numbers`
numbers allowed for the captcha
- `$_allow_numbers`
whether to allow number in captcha or not

8.13.1 Detailed Description

Classe per la generazione di immagini captcha.

`imagettftext()` richiede le GD library e le FreeType library.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.13.2 Constructor & Destructor Documentation

8.13.2.1 `__construct ($name, $opts = null)`

Costruttore.

Parameters

string	<i>\$name</i>	nome del campo captcha
array	<i>\$opts</i>	array associativo di opzioni <ul style="list-style-type: none">• allow_numbers (boolean): permettere o meno caratteri di tipo numero, default FALSE

Returns

istanza di [Gino.Captcha](#)

8.13.3 Member Function Documentation

8.13.3.1 `check ()`

Controlla che il codice inserito dall'utente sia corretto.

Returns

risultato controllo, bool

8.13.3.2 `checkRequirements ()` [private]

Verifica se l'immagine captcha può essere generata.

Returns

risultato verifica, bool

8.13.3.3 `generateStrings ()` [private]

Genera due stringhe alfanumeriche random.

Returns

array(stringa, stringa)

8.13.3.4 `hex2RGB ($hexStr)` [private]

Converte un colore da formato hex a formato array rgb.

Parameters

string	<i>\$hexStr</i>	formato hex
--------	-----------------	-------------

Returns

array associativo contenente i valori rgb (red, blue, green)

8.13.3.5 `render ($opts = null)`

Renderizza l'immagine e l'input per l'inserimento del codice.

See also

`self::checkRequirements()`

Parameters

mixed	<i>\$opts</i>	array associativo di opzioni <ul style="list-style-type: none"> • bkg_color (string): hex code del colore del background, default #00ff00 • color (string): hex code del colore dei caratteri, default #000000
-------	---------------	--

Returns

immagine captcha e form input

8.13.4 Field Documentation

8.13.4.1 `$allow_numbers` [private]

whether to allow number in captcha or not

8.13.4.2 `$font_file` [private]

absolute path of the font ttf file

8.13.4.3 `$height` [private]

captcha image height

8.13.4.4 `$letters` [private]

letters allowed for the captcha

8.13.4.5 `$name` [private]

name of the captcha field

8.13.4.6 `$numbers` [private]

numbers allowed for the captcha

8.13.4.7 `$width` [private]

captcha image width

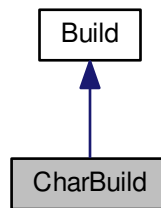
The documentation for this class was generated from the following file:

- lib/classes/[class.Captcha.php](#)

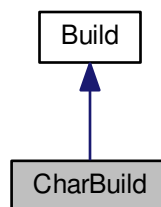
8.14 CharBuild Class Reference

Gestisce i campi di tipo stringa (CHAR, VARCHAR)

Inheritance diagram for CharBuild:



Collaboration diagram for CharBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getTrnsl](#) ()
Getter della proprietà trnsl (indica se il campo necessita di traduzione o no)
- [setTrnsl](#) (\$value)
Setter della proprietà trnsl.
- [filterWhereClause](#) (\$value)
- [formElement](#) (\Gino\Form \$form, \$options)

Protected Attributes

- [\\$_trnsl](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.14.1 Detailed Description

Gestisce i campi di tipo stringa (CHAR, VARCHAR)

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.14.2 Constructor & Destructor Documentation

8.14.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.14.3 Member Function Documentation

8.14.3.1 filterWhereClause (*\$value*)

See also

[Gino.Build::filterWhereClause\(\)](#)

8.14.3.2 formElement (\Gino\Form *\$form*, *\$options*)

See also

[Gino.Build::formElement\(\)](#)

8.14.3.3 getTrnsl ()

Getter della proprietà trnsl (indica se il campo necessita di traduzione o no)

Returns

proprietà trnsl

8.14.3.4 setTrnsl (*\$value*)

Setter della proprietà trnsl.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.14.4 Field Documentation

8.14.4.1 `$_trnsI` [protected]

Proprietà dei campi specifiche del tipo di campo.

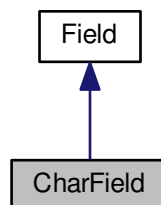
The documentation for this class was generated from the following file:

- lib/classes/build/[class.CharBuild.php](#)

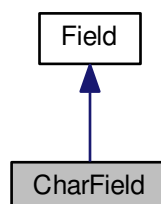
8.15 CharField Class Reference

Campi di tipo stringa (CHAR, VARCHAR)

Inheritance diagram for CharField:



Collaboration diagram for CharField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)
- [valueToDb](#) (\$value)

Protected Attributes

- [\\$_trnsI](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.15.1 Detailed Description

Campi di tipo stringa (CHAR, VARCHAR)

Tipologie di input associabili: testo, campo nascosto, textarea

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.15.2 Constructor & Destructor Documentation

8.15.2.1 [__construct](#) (*\$options*)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Field()• opzioni specifiche del tipo di campo<ul style="list-style-type: none">– trnsI (boolean): campo con traduzioni
-------	------------------	--

Returns

istanza di [Gino.CharField](#)

8.15.3 Member Function Documentation

8.15.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.15.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.15.3.3 `valueToDb ($value)`

See also

[Gino.Field::valueToDb\(\)](#)

Returns

null or string

8.15.4 Field Documentation

8.15.4.1 `$_trns1` [protected]

Proprietà dei campi specifiche del tipo di campo.

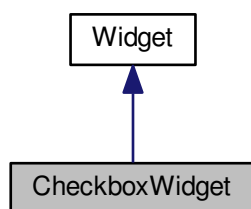
The documentation for this class was generated from the following file:

- [lib/classes/fields/class.CharField.php](#)

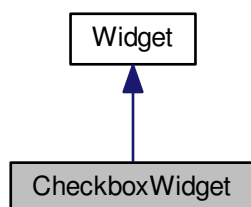
8.16 CheckboxWidget Class Reference

Campi checkbox nei form.

Inheritance diagram for CheckboxWidget:



Collaboration diagram for CheckboxWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.16.1 Detailed Description

Campi checkbox nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.16.2 Member Function Documentation

8.16.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

The documentation for this class was generated from the following file:

- lib/classes/widget/[class.CheckboxWidget.php](#)

8.17 Compressor Class Reference

Classe per la compressione di css e js (merge e minify)

Public Member Functions

- [__construct](#) (\$params=array())
Costruttore.
- [addCss](#) (\$paths)
Aggiunge nuovi path css.
- [addJs](#) (\$paths)
Aggiunge nuovi path js.
- [mergeJs](#) (\$options=array())
Merge di script js in un unico file.
- [mergeCss](#) (\$options=array())
Merge di stylesheet in un unico file.

Data Fields

- [\\$_js](#) = array()

Private Member Functions

- [minifyJs](#) (\$content)
Esegue il minify del contenuto js.
- [shouldUpdate](#) (\$type, \$paths_string, \$paths)
Verifica se il file compresso deve essere ricreato.
- [filePath](#) (\$type, \$paths_string)
Percorso del file compresso.
- [moveCss](#) (\$source, \$destination, \$content)
Modifica i path all'interno dei css per rispecchiare il cambio di directory del css.
- [convertRelativePath](#) (\$path, \$from, \$to)
Converte un path relativo ad una folder in un path relativo ad un'altra folder.
- [minifyCss](#) (\$content)
Esegue il minify del contenuto css.

Private Attributes

- [\\$_css](#) = array()

8.17.1 Detailed Description

Classe per la compressione di css e js (merge e minify)

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.17.2 Constructor & Destructor Documentation

8.17.2.1 `__construct ($params = array())`

Costruttore.

Parameters

array	<i>\$params</i>	Array associativo di parametri <ul style="list-style-type: none">css: path o array di path relativi di cssjs: path o array di path relativi di js
-------	-----------------	--

Returns

istanza di [Gino.Compressor](#)

8.17.3 Member Function Documentation

8.17.3.1 `addCss ($paths)`

Aggiunge nuovi path css.

Returns

void

8.17.3.2 `addJs ($paths)`

Aggiunge nuovi path js.

Returns

void

8.17.3.3 `convertRelativePath ($path, $from, $to)` [private]

Converte un path relativo ad una folder in un path relativo ad un'altra folder.

Parameters

string	<i>\$path</i>	percorso da convertire
string	<i>\$from</i>	folder attuale
string	<i>\$to</i>	folder finale

Returns

path convertito

8.17.3.4 `filePath ($type, $paths_string)` [private]

Percorso del file compresso.

Returns

path

8.17.3.5 `mergeCss ($options = array())`

Merge di stylesheet in un unico file.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">minify: eseguire il minify dei file oppure no
-------	------------------	--

Returns

path del file unificato

8.17.3.6 `mergeJs ($options = array())`

Merge di script js in un unico file.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">minify: eseguire il minify dei file oppure no
-------	------------------	--

Returns

path del file unificato

8.17.3.7 `minifyCss ($content)` [private]

Esegue il minify del contenuto css.

Parameters

string	<i>\$content</i>	contenuto css
--------	------------------	---------------

Returns

contenuto minificato

8.17.3.8 minifyJs (*\$content*) [private]

Esegue il minify del contenuto js.

Parameters

string	<i>\$content</i>	contenuto js
--------	------------------	--------------

See also

[JSMIn](#)

Returns

contenuto minificato

8.17.3.9 moveCss (*\$source*, *\$destination*, *\$content*) [private]

Modifica i path all'interno dei css per rispecchiare il cambio di directory del css.

Parameters

string	<i>\$source</i>	path del css sorgente
string	<i>\$destination</i>	path del css di destinazione
string	<i>\$content</i>	contenuto del file sorgente

Returns

contenuto con i path aggiornati

8.17.3.10 shouldUpdate (*\$type*, *\$paths_string*, *\$paths*) [private]

Verifica se il file compresso deve essere ricreato.

Returns

vero o falso

8.17.4 Field Documentation**8.17.4.1 \$_css = array()** [private]**8.17.4.2 \$_js = array()**

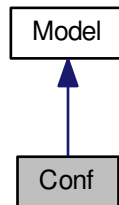
The documentation for this class was generated from the following file:

- lib/classes/[class.Compressor.php](#)

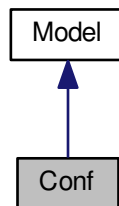
8.18 Conf Class Reference

Classe di tipo [Gino.Model](#) che gestisce le impostazioni del sistema.

Inheritance diagram for Conf:



Collaboration diagram for Conf:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.

Static Public Attributes

- static [\\$table](#) = "sys_conf"
- static [\\$columns](#)

Additional Inherited Members

8.18.1 Detailed Description

Classe di tipo [Gino.Model](#) che gestisce le impostazioni del sistema.

Le impostazioni sono salvate in una riga di tabella corrispondente ad ID 1

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.18.2 Constructor & Destructor Documentation

8.18.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.App.Sysconf.Conf](#)

8.18.3 Member Function Documentation

8.18.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

string

8.18.3.2 `static columns ()` [*static*]

Struttura dei campi della tabella di un modello.

Returns

array

8.18.4 Field Documentation

8.18.4.1 `$columns` [*static*]

8.18.4.2 `$table = "sys_conf"` [*static*]

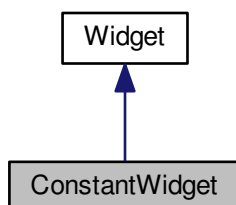
The documentation for this class was generated from the following file:

- [app/sysconf/class.Conf.php](#)

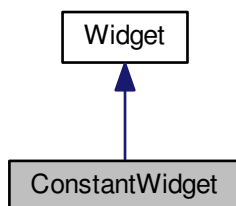
8.19 ConstantWidget Class Reference

Campi di tipo costante nei form.

Inheritance diagram for ConstantWidget:



Collaboration diagram for ConstantWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.19.1 Detailed Description

Campi di tipo costante nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.19.2 Member Function Documentation

8.19.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

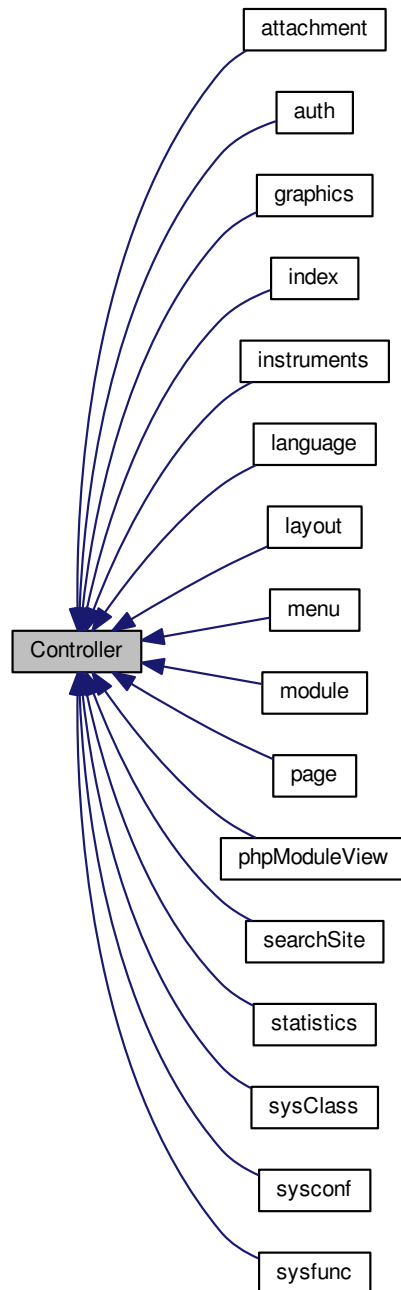
The documentation for this class was generated from the following file:

- `lib/classes/widget/class.ConstantWidget.php`

8.20 Controller Class Reference

Classe astratta primitiva di tipo [Controller](#) (MVC), dalla quale tutti i controller delle singole app discendono.

Inheritance diagram for Controller:



Public Member Functions

- [__construct](#) (\$instance_id=0)
Inizializza il controller.
- [__call](#) (\$name, \$arguments)
Funzione chiamata quando si cerca di chiamare un metodo inaccessibile.
- [getInstance](#) ()

- *Espone l'id valore dell'istanza.*
- [getInstanceName](#) ()
- *Espone il nome dell'istanza.*
- [getClassName](#) ()
- *Espone il nome della classe.*
- [getBaseAbsPath](#) ()
- *Percorso assoluto alla cartella dei contenuti.*
- [getBasePath](#) ()
- *Percorso relativo alla cartella dei contenuti.*
- [requirePerm](#) (\$perm)
- *Richiama il metodo anonimo di [Access](#) passando in automatico classe e istanza.*
- [userHasPerm](#) (\$perm)
- *Richiama il metodo anonimo di [User](#) passando in automatico classe e istanza.*
- [link](#) (\$instance_name, \$method, array \$params=array(), \$query_string= "", array \$kwargs=array())
- *Shortcut link per classi di tipo [Gino.Controller](#).*
- [linkAdmin](#) (array \$params=array(), \$query_string= "", array \$kwargs=array())
- *Shortcut link area amministrativa per classi di tipo [Gino.Controller](#).*
- [manageOptions](#) ()
- *Interfaccia per la gestione delle opzioni dei moduli.*
- [manageFrontend](#) ()
- *Interfaccia per la gestione delle viste e css (frontend)*
- [manageLocale](#) ()
- *Interfaccia per la gestione dei file delle traduzioni.*
- [deleteInstance](#) ()
- *Eliminazione istanza del modulo Questo metodo deve essere sovrascritto dalle classi istanziabili per permettere l'eliminazione delle istanze.*

Static Public Member Functions

- static [getClassElements](#) ()
- *Restituisce alcune proprietà della classe.*

Data Fields

- [\\$_db](#)
- [\\$_access](#)
- [\\$_session](#)
- [\\$_plink](#)
- [\\$_trd](#)
- [\\$_locale](#)
- [\\$_permissions](#)
- [\\$_class_name](#)
- [\\$_instance](#)
- [\\$_instance_name](#)
- [\\$_instance_label](#)
- [\\$_class_www](#)
- [\\$_class_img](#)
- [\\$_data_dir](#)
- [\\$_data_www](#)
- [\\$_view_dir](#)
- [\\$_home](#)

Protected Member Functions

- [setPaths](#) ()
Setta i percorsi di base dell'app.
- [setOption](#) (\$option, \$options=FALSE)
Opzioni di classe.

Protected Attributes

- [\\$_registry](#)

Private Member Functions

- [setInstanceProperties](#) (\$instance_id)
Setta le proprietà legate all'istanza.

8.20.1 Detailed Description

Classe astratta primitiva di tipo [Controller](#) (MVC), dalla quale tutti i controller delle singole app discendono.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.20.2 Constructor & Destructor Documentation

8.20.2.1 `__construct ($instance_id = 0)`

Inizializza il controller.

Parameters

int	<i>\$instance_id</i>	id modulo, se diverso da zero il modulo è un'istanza di una classe, altrimenti è la classe di sistema
-----	----------------------	---

Returns

istanza di [Gino.Controller](#)

8.20.3 Member Function Documentation

8.20.3.1 `__call ($name, $arguments)`

Funzione chiamata quando si cerca di chiamare un metodo inaccessibile.

Parameters

string	<i>\$name</i>	nome metodo
array	<i>\$arguments</i>	argomenti

Returns

[Exception](#)

8.20.3.2 deleteInstance ()

Eliminazione istanza del modulo Questo metodo deve essere sovrascritto dalle classi istanziabili per permettere l'eliminazione delle istanze.

Se non sovrascritto viene chiamato e getta una [Exception](#)

8.20.3.3 getBaseAbsPath ()

Percorso assoluto alla cartella dei contenuti.

Returns

percorso assoluto

8.20.3.4 getBasePath ()

Percorso relativo alla cartella dei contenuti.

Returns

percorso relativo

8.20.3.5 static getClassElements () [static]

Restituisce alcune proprietà della classe.

Le informazioni vengono utilizzate per creare o eliminare istanze. Questo metodo dev'essere sovrascritto da tutte le classi figlie.

Returns

lista delle proprietà utilizzate per la creazione di istanze di tipo pagina

8.20.3.6 getClassName ()

Espone il nome della classe.

Returns

nome classe

8.20.3.7 getInstance ()

Espone l'id valore dell'istanza.

Returns

id

8.20.3.8 getInstanceName ()

Espone il nome dell'istanza.

Returns

nome istanza

8.20.3.9 link (\$instance_name, \$method, array \$params = array(), \$query_string = ' ', array \$kwargs = array())

Shortcut link per classi di tipo [Gino.Controller](#).

See also

[Gino.Router::link](#)

8.20.3.10 linkAdmin (array \$params = array(), \$query_string = ' ', array \$kwargs = array())

Shortcut link area amministrativa per classi di tipo [Gino.Controller](#).

See also

[Gino.Router::link](#)

8.20.3.11 manageFrontend ()

Interfaccia per la gestione delle viste e css (frontend)

See also

[Frontend::manageFrontend\(\)](#)

Returns

interfaccia di amministrazione frontend (viste, css...)

8.20.3.12 manageLocale ()

Interfaccia per la gestione dei file delle traduzioni.

See also

[Locale::manageLocale\(\)](#)

Returns

interfaccia di amministrazione traduzioni

8.20.3.13 `manageOptions ()`

Interfaccia per la gestione delle opzioni dei moduli.

See also

[Gino.Options::manageDoc\(\)](#)

Returns

interfaccia di amministrazione opzioni

8.20.3.14 `requirePerm ($perm)`

Richiama il metodo anonimo di [Access](#) passando in automatico classe e istanza.

See also

[Access::requirePerm](#)

Returns

void

8.20.3.15 `setInstanceProperties ($instance_id)` [private]

Setta le proprietà legate all'istanza.

Parameters

int	<i>\$id</i>	id dell'istanza
-----	-------------	-----------------

Returns

void

8.20.3.16 `setOption ($option, $options = FALSE)` [protected]

Opzioni di classe.

Parameters

string	<i>\$option</i>	nome del campo dell'opzione di classe
mixed	<i>\$options</i>	default FALSE. <ul style="list-style-type: none"> • (array): chiavi value (valore di default), translation (traduzione) • (boolean): indica se è prevista la traduzione (compatibilità con precedenti versioni di gino)

Returns

mixed

8.20.3.17 setPaths () [protected]

Setta i percorsi di base dell'app.

Returns

void

8.20.3.18 userHasPerm (\$perm)

Richiama il metodo anonimo di User passando in automatico classe e istanza.

See also

User:hasPerm

Returns

void

8.20.4 Field Documentation**8.20.4.1 \$_access****8.20.4.2 \$_class_img****8.20.4.3 \$_class_name****8.20.4.4 \$_class_www****8.20.4.5 \$_data_dir****8.20.4.6 \$_data_www****8.20.4.7 \$_db****8.20.4.8 \$_home****8.20.4.9 \$_instance****8.20.4.10 \$_instance_label****8.20.4.11 \$_instance_name****8.20.4.12 \$_locale****8.20.4.13 \$_permissions**

8.20.4.14 `$_plink`

8.20.4.15 `$_registry` [protected]

8.20.4.16 `$_session`

8.20.4.17 `$_trd`

8.20.4.18 `$_view_dir`

The documentation for this class was generated from the following file:

- lib/classes/mvc/[class.Controller.php](#)

8.21 Core Class Reference

Gestisce una [Gino.Http.Request](#) ed invia una [Gino.Http.Response](#) adeguata.

Public Member Functions

- [__construct](#) ()
Costruttore Include le classi fondamentali per il funzionamento del sistema, setta il locale corretto ed inizializza il registro.
- [answer](#) ()
Invia la risposta HTTP al client e chiude la connessione al DB Se la risposta ricevuta dal [Gino](#) non è una [Gino.Http.<→Response](#) invia una [Gino.Http.ResponseNotFound](#) (404)

Data Fields

- [\\$_db](#)

Private Member Functions

- [initRegistry](#) ()
Inizializza il registro di sistema.
- [initMobile](#) ()
Esegue operazioni relative ai dispositivi mobile Controlla ed imposta la variabile di sessione che gestisce risposte adatte per il mobile.
- [detectMobile](#) (\Gino\Session \$session)
Esegue il detect di dispositivi mobile, setta una variabile di sessione se il detect è positivo.

Private Attributes

- [\\$_registry](#)

8.21.1 Detailed Description

Gestisce una [Gino.Http.Request](#) ed invia una [Gino.Http.Response](#) adeguata.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.21.2 Constructor & Destructor Documentation

8.21.2.1 `__construct ()`

Costruttore Include le classi fondamentali per il funzionamento del sistema, setta il locale corretto ed inizializza il registro.

8.21.3 Member Function Documentation

8.21.3.1 `answer ()`

Invia la risposta HTTP al client e chiude la connessione al DB Se la risposta ricevuta dal [Gino](#) non è una [Gino](#)↔[Http.Response](#) invia una [Gino.Http.ResponseNotFound](#) (404)

Returns

void

8.21.3.2 `detectMobile (\Gino\Session $session)` [private]

Esegue il detect di dispositivi mobile, setta una variabile di sessione se il detect è positivo.

Parameters

\Gino ↔ Session	<code>\$session</code>	istanza di Gino.Session
------------------------------------	------------------------	---

Returns

void

8.21.3.3 `initMobile ()` [private]

Esegue operazioni relative ai dispositivi mobile Controlla ed imposta la variabile di sessione che gestisce risposte adatte per il mobile.

Returns

void

8.21.3.4 `initRegistry ()` [private]

Inizializza il registro di sistema.

Returns

void

8.21.4 Field Documentation

8.21.4.1 \$_db

8.21.4.2 \$_registry [private]

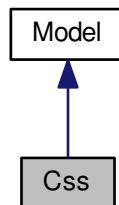
The documentation for this class was generated from the following file:

- lib/classes/[class.Core.php](#)

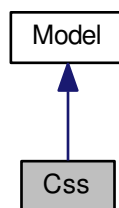
8.22 Css Class Reference

Libreria per la gestione dei file css dei singoli moduli e dei file css del layout (da associare alle skin)

Inheritance diagram for Css:



Collaboration diagram for Css:



Public Member Functions

- [__construct](#) ([\\$type](#), [\\$params=array\(\)](#))
Costruttore.
- [setFilename](#) ([\\$value](#))
Setta il nome del file.
- [formCssLayout](#) ()

Form per la creazione e la modifica di un file css (layout)

- [actionCssLayout](#) (\Gino\Http\Request \$request)

Processa il form di inserimento/modifica css.

- [formDelCssLayout](#) ()

Form per l'eliminazione di un file css (layout)

- [actionDelCssLayout](#) (\Gino\Http\Request \$request)

Processa il form di eliminazione css.

Static Public Member Functions

- static [columns](#) ()

Struttura dei campi della tabella di un modello.

- static [getAll](#) (\$order= 'label')

Lista oggetti.

- static [getFromFilename](#) (\$filename)

Recupera l'oggetto a partire dal nome del file.

- static [layoutInfo](#) ()

Descrizione della procedura.

Data Fields

- [\\$_module](#)
- [\\$_name](#)
- [\\$_label](#)
- [\\$_css_list](#)

Static Public Attributes

- static [\\$table](#) = 'sys_layout_css'
- static [\\$columns](#)

Protected Attributes

- [\\$_tbl_data](#)

Private Member Functions

- [cssFileName](#) (\$css_file)

Ricava il nome del file css dell'istanza di un modulo.

Private Attributes

- [\\$_class](#)
- [\\$_instance_class](#)
- [\\$_mdlLink](#)
- [\\$_interface](#)
- [\\$_tbl_module](#)

Additional Inherited Members

8.22.1 Detailed Description

Libreria per la gestione dei file css dei singoli moduli e dei file css del layout (da associare alle skin)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.22.2 Constructor & Destructor Documentation

8.22.2.1 `__construct ($type, $params = array())`

Costruttore.

I nomi dei file CSS del modulo vengono recuperati richiamando il metodo `getClassElements()` che ritorna un array con la chiave *css* (array).

Se il modulo non è istanziabile il metodo `getClassElements()` dovrà riportare anche la chiave *instance* con valore *false*.

See also

`getClassElements()`

Parameters

string	<i>\$type</i>	tipo di utilizzo <ul style="list-style-type: none"> • module • layout
array	<i>\$params</i>	array associativo di opzioni <ul style="list-style-type: none"> • id (integer): valore ID del record • class (string): nome della classe • module (integer): valore ID del modulo • name (string): nome del modulo • label (string): etichetta del modulo

Returns

istanza di [Gino.Css](#)

8.22.3 Member Function Documentation

8.22.3.1 `actionCssLayout (\Gino\Http\Request $request)`

Processa il form di inserimento/modifica css.

See also

`self::formCssLayout()`

Parameters

--	--

8.22.3.2 `actionDelCssLayout (\Gino\Http\Request $request)`

Processa il form di eliminazione css.

See also

`self::formDelCssLayout()`

Parameters

--	--

8.22.3.3 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.22.3.4 `cssFileName ($css_file) [private]`

Ricava il nome del file css dell'istanza di un modulo.

Returns

nome file

8.22.3.5 `formCssLayout ()`

[Form](#) per la creazione e la modifica di un file css (layout)

Returns

codice html form

8.22.3.6 `formDelCssLayout ()`

[Form](#) per l'eliminazione di un file css (layout)

Returns

codice html form

8.22.3.7 `static getAll ($order = 'label') [static]`

Lista oggetti.

Parameters

string	<i>\$order</i>	campo di ordinamento risultati
--------	----------------	--------------------------------

Returns

array di oggetti [Gino.Css](#)

8.22.3.8 static getFromFilename (*\$filename*) [static]

Recupera l'oggetto a partire dal nome del file.

Returns

[Gino.Css](#) o null se non lo trova

8.22.3.9 static layoutInfo () [static]

Descrizione della procedura.

Returns

informazioni, codice html

8.22.3.10 setFilename (*\$value*)

Setta il nome del file.

Parameters

string	<i>\$value</i>	nome file
--------	----------------	-----------

Returns

TRUE

8.22.4 Field Documentation

8.22.4.1 *\$_class* [private]

8.22.4.2 *\$_css_list*

8.22.4.3 *\$_instance_class* [private]

8.22.4.4 *\$_interface* [private]

8.22.4.5 *\$_label*

8.22.4.6 *\$_mdlLink* [private]

8.22.4.7 *\$_module*

8.22.4.8 *\$_name*

8.22.4.9 `$_tbl_data` [`protected`]

8.22.4.10 `$_tbl_module` [`private`]

8.22.4.11 `$columns` [`static`]

8.22.4.12 `$table = 'sys_layout_css'` [`static`]

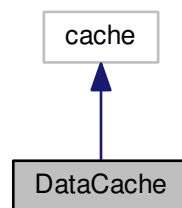
The documentation for this class was generated from the following file:

- `lib/classes/class.Css.php`

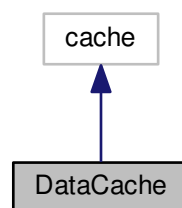
8.23 DataCache Class Reference

Memorizza le strutture dati scrivendo su file.

Inheritance diagram for DataCache:



Collaboration diagram for DataCache:



Public Member Functions

- `__construct` (`$enabled=TRUE`)
- `get` (`$grp`, `$id`, `$tc`)

Fornisce i dati in cache o ritorna FALSE se non presenti o scaduti.

- `save ($data)`

Salva i dati serializzati su file se la cache è abilitata.

8.23.1 Detailed Description

Memorizza le strutture dati scrivendo su file.

Esempio di utilizzo

```
$cache = new dataCache();
if (!$data = $cache->get('group_name', 'id', 3600)) {
    $data = someCalculations();
    $cache->save($data);
}
```

Se i dati sono in cache sono ritornati dal metodo `self::get` e l'if statement viene skippato, altrimenti i dati sono ricavati e salvati nella cache

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.23.2 Constructor & Destructor Documentation

8.23.2.1 `__construct ($enabled = TRUE)`

8.23.3 Member Function Documentation

8.23.3.1 `get ($grp, $id, $tc)`

Fornisce i dati in cache o ritorna FALSE se non presenti o scaduti.

Parameters

string	<i>\$grp</i>	gruppo
string	<i>\$id</i>	

Returns

dati deserializzati o FALSE

8.23.3.2 `save ($data)`

Salva i dati serializzati su file se la cache è abilitata.

Returns

void

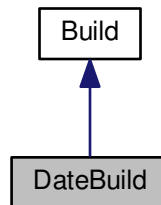
The documentation for this class was generated from the following file:

- `lib/classes/class.Cache.php`

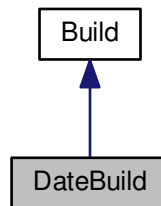
8.24 DateBuild Class Reference

Gestisce i campi di tipo data.

Inheritance diagram for DateBuild:



Collaboration diagram for DateBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [formElement](#) (\Gino\Form \$form, \$options)
- [filterWhereClause](#) (\$value, \$options=array())
- [clean](#) (\$options=null)

Additional Inherited Members

8.24.1 Detailed Description

Gestisce i campi di tipo data.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.24.2 Constructor & Destructor Documentation

8.24.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.24.3 Member Function Documentation

8.24.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

valore del campo

8.24.3.2 clean (*\$options* = null)

See also

[Gino.Build::clean\(\)](#)

8.24.3.3 filterWhereClause (*\$value*, *\$options* = array())

See also

[Gino.Build::filterWhereClause\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">operator (string): operatore di confronto della data
-------	------------------	--

8.24.3.4 formElement (\Gino\Form \$form, \$options)

See also

[Gino.Build::formElement\(\)](#)

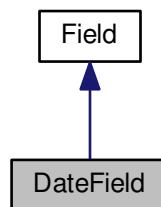
The documentation for this class was generated from the following file:

- [lib/classes/build/class.DateBuild.php](#)

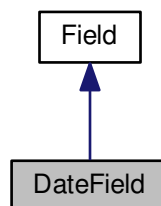
8.25 DateField Class Reference

Campo di tipo data.

Inheritance diagram for DateField:



Collaboration diagram for DateField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.

Additional Inherited Members

8.25.1 Detailed Description

Campo di tipo data.

Tipologie di input associabili: testo in formato data

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.25.2 Constructor & Destructor Documentation

8.25.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Field()
-------	------------------	---

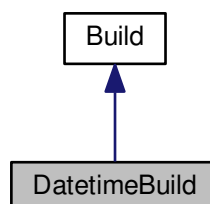
The documentation for this class was generated from the following file:

- lib/classes/fields/[class.DateField.php](#)

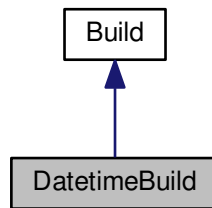
8.26 DatetimeBuild Class Reference

Gestisce i campi di tipo DATETIME.

Inheritance diagram for DatetimeBuild:



Collaboration diagram for DatetimeBuild:



Public Member Functions

- `__construct` (\$options)
Costruttore.
- `__toString` ()
Rappresentazione a stringa dell'oggetto.
- `filterWhereClause` (\$value, \$options=array())
- `clean` (\$options=null)

Data Fields

- `$_auto_now_add`
- `$_view_input`

Protected Attributes

- `$_auto_now`
Proprietà dei campi specifiche del tipo di campo.

8.26.1 Detailed Description

Gestisce i campi di tipo DATETIME.

Impostando opportunamente le proprietà `$_auto_now_add` e `$_auto_now` è possibile gestire il campo datetime in modo che venga impostato soltanto quando viene creato l'oggetto oppure ogni volta che l'oggetto viene salvato.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.26.2 Constructor & Destructor Documentation

8.26.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.26.3 Member Function Documentation

8.26.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

valore del campo

8.26.3.2 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#)

8.26.3.3 `filterWhereClause ($value, $options = array())`

See also

[Gino.Build::filterWhereClause\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• operator (string): operatore di confronto della data
-------	------------------	--

8.26.4 Field Documentation

8.26.4.1 `$_auto_now` [*protected*]

Proprietà dei campi specifiche del tipo di campo.

8.26.4.2 `$_auto_now_add`

8.26.4.3 `$_view_input`

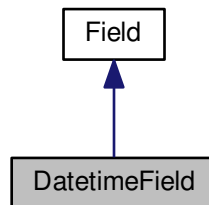
The documentation for this class was generated from the following file:

- `lib/classes/build/class.DatetimeBuild.php`

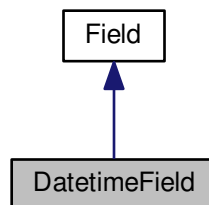
8.27 DatetimeField Class Reference

Campo di tipo DATETIME.

Inheritance diagram for DatetimeField:



Collaboration diagram for DatetimeField:



Public Member Functions

- `__construct` (`$options`)
Costruttore.
- `getProperties` ()
- `getAutoNow` ()
Getter della proprietà `auto_now` (update ad ogni modifica del record)
- `setAutoNow` (`$value`)
Setter della proprietà `auto_now`.

- `getAutoNowAdd ()`
Getter della proprietà auto_now_add (update in inserimento record)
- `setAutoNowAdd ($value)`
Setter della proprietà auto_now_add.
- `valueFromDb ($value)`

Data Fields

- `$_auto_now_add`
- `$_view_input`

Protected Attributes

- `$_auto_now`
Proprietà dei campi specifiche del tipo di campo.

8.27.1 Detailed Description

Campo di tipo DATETIME.

Tipologie di input associabili: nessun input, testo nascosto, testo in formato datetime (YYYY-MM-DD HH:MM:SS).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.27.2 Constructor & Destructor Documentation

8.27.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	<p>array associativo di opzioni del campo del database</p> <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – auto_now (boolean): imposta automaticamente il valore del campo al tempo/ora corrente ogni volta che l'oggetto viene salvato (default: true) – auto_now_add (boolean): imposta automaticamente il valore del campo al tempo/ora corrente la prima volta che l'oggetto viene creato (default: true) – view_input (boolean): per visualizzare l'input nel form (default false)
-------	------------------	---

8.27.3 Member Function Documentation

8.27.3.1 getAutoNow ()

Getter della proprietà `auto_now` (update ad ogni modifica del record)

Returns

proprietà `auto_now`

8.27.3.2 getAutoNowAdd ()

Getter della proprietà `auto_now_add` (update in inserimento record)

Returns

proprietà `auto_now_add`

8.27.3.3 getProperties ()

See also

[Gino.Field::getProperties\(\)](#)

8.27.3.4 setAutoNow (\$value)

Setter della proprietà `auto_now`.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.27.3.5 setAutoNowAdd (*\$value*)

Setter della proprietà auto_now_add.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.27.3.6 valueFromDb (*\$value*)

See also

[Gino.Field::valueFromDb\(\)](#)

8.27.4 Field Documentation

8.27.4.1 `$_auto_now` [protected]

Proprietà dei campi specifiche del tipo di campo.

8.27.4.2 `$_auto_now_add`8.27.4.3 `$_view_input`

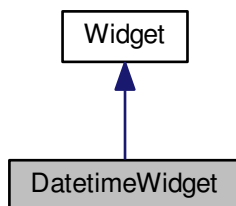
The documentation for this class was generated from the following file:

- lib/classes/fields/[class.DatetimeField.php](#)

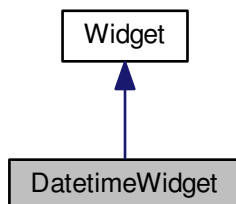
8.28 DatetimeWidget Class Reference

Campi di tipo datetime nei form.

Inheritance diagram for DateTimeWidget:



Collaboration diagram for DateTimeWidget:



Public Member Functions

- [printInputForm](#) (`$form`, `$options`)

Additional Inherited Members

8.28.1 Detailed Description

Campi di tipo datetime nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.28.2 Member Function Documentation

8.28.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

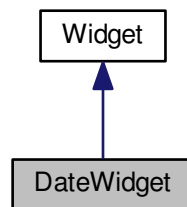
The documentation for this class was generated from the following file:

- lib/classes/widget/class.DatetimeWidget.php

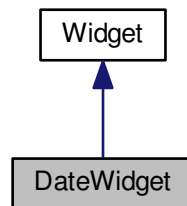
8.29 DateWidget Class Reference

Campi di tipo date nei form.

Inheritance diagram for DateWidget:



Collaboration diagram for DateWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.29.1 Detailed Description

Campi di tipo date nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.29.2 Member Function Documentation

8.29.2.1 printInputForm (*\$form*, *\$options*)

See also

[Gino.Widget::printInputForm\(\)](#)

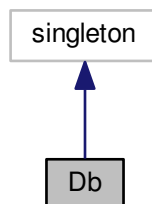
The documentation for this class was generated from the following file:

- lib/classes/widget/class.DateWidget.php

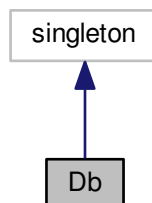
8.30 Db Class Reference

Classe Factory e [Singleton](#) usata per creare oggetti che si interfacciano al database.

Inheritance diagram for Db:



Collaboration diagram for Db:



Static Public Member Functions

- static [instance](#) ()

Istanzia la classe che si occupa della connessione al database Garantisce che tutti gli oggetti ricevano sempre la stessa istanza [Singleton](#).

Static Private Attributes

- static [\\$_dbms](#) = DBMS
- static [\\$_db_host](#) = DB_HOST
- static [\\$_db_user](#) = DB_USER
- static [\\$_db_pass](#) = DB_PASSWORD
- static [\\$_db_dbname](#) = DB_DBNAME
- static [\\$_db_charset](#) = DB_CHARSET
- static [\\$_db_schema](#) = DB_SCHEMA

8.30.1 Detailed Description

Classe Factory e [Singleton](#) usata per creare oggetti che si interfacciano al database.

Le librerie di connessione al database sono sottoclassi di questa (che funziona come "scheletro") e vengono istanziate nel metodo [instance\(\)](#)

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.30.2 Member Function Documentation

8.30.2.1 static instance () [static]

Istanzia la classe che si occupa della connessione al database Garantisce che tutti gli oggetti ricevano sempre la stessa istanza [Singleton](#).

Returns

object

8.30.3 Field Documentation

8.30.3.1 [\\$_db_charset](#) = DB_CHARSET [static],[private]

8.30.3.2 [\\$_db_dbname](#) = DB_DBNAME [static],[private]

8.30.3.3 [\\$_db_host](#) = DB_HOST [static],[private]

8.30.3.4 [\\$_db_pass](#) = DB_PASSWORD [static],[private]

8.30.3.5 [\\$_db_schema](#) = DB_SCHEMA [static],[private]

8.30.3.6 `$_db_user = DB_USER` `[static]`, `[private]`

8.30.3.7 `$_dbms = DBMS` `[static]`, `[private]`

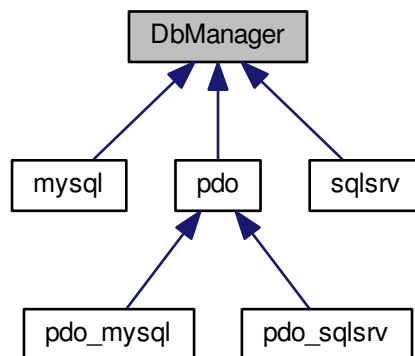
The documentation for this class was generated from the following file:

- `lib/classes/class.Db.php`

8.31 DbManager Interface Reference

Interfaccia per le librerie di connessione al database.

Inheritance diagram for DbManager:



Public Member Functions

- `__construct` (`$params`)
Costruttore.
- `getInfoQuery` ()
Restituisce informazioni sull'esecuzione delle query.
- `openConnection` ()
Apri una connessione al database.
- `closeConnection` ()
Chiude la connessione al database.
- `begin` ()
Istruzione begin.
- `rollback` ()
Istruzione rollback.
- `commit` ()
Istruzione commit.
- `multiActionQuery` (`$file_content`)
Esegue una o più query concatenate dal punto e virgola.
- `freeresult` (`$result`)
Libera tutta la memoria utilizzata dal set di risultati.

- **getLastId** (\$table)
Valore dell'ultimo ID generato da una colonna Auto Increment a seguito di una istruzione INSERT o UPDATE.
- **autoIncValue** (\$table)
Valore di Auto Increment.
- **getFieldFromId** (\$table, \$field, \$field_id, \$id, \$options)
Ricava il valore di un campo a una data condizione.
- **tableexists** (\$table)
Verifica l'esistenza di una tabella.
- **fieldInformations** (\$table)
Recupera le informazioni sui campi di una tabella.
- **conformFieldType** (\$type)
Uniforma i tipi di dato dei campi.
- **limit** (\$range, \$offset)
Istruzione per limitare i risultati di una query (LIMIT)
- **distinct** (\$fields, \$options)
Distinct keyword.
- **concat** (\$sequence)
Istruzione per concatenare i campi.
- **dumpDatabase** (\$file)
Effettua il dump del database.
- **getTableStructure** (\$table)
Informazioni sulla struttura di una tabella del database.
- **changeFieldType** (\$data_type, \$value)
Reimposta il corretto tipo di dato di un campo quando il valore recuperato da una istruzione select è di un tipo non corrispondente (vedi PDO_SQLSRV)
- **getFieldsName** (\$table)
Recupera il nome dei campi di una tabella.
- **getNumRecords** (\$table, \$where, \$field, \$options)
Numero di record interessati da una query di selezione.
- **query** (\$fields, \$tables, \$where, \$options)
Costruisce una query di selezione.
- **execCustomQuery** (\$query, \$options)
Esegue una query di selezione passando direttamente la query.
- **select** (\$fields, \$tables, \$where, \$options)
Esegue una query di selezione passando i parametri di costruzione della query.
- **insert** (\$fields, \$table, \$debug)
Costruisce ed esegue una query di inserimento.
- **update** (\$fields, \$table, \$where, \$debug)
Costruisce ed esegue una query di aggiornamento.
- **delete** (\$table, \$where, \$debug)
Costruisce ed esegue una query di eliminazione.
- **drop** (\$table)
Eliminazione di una tabella.
- **columnHasValue** (\$table, \$field, \$value, \$options=array())
Controlla che il valore del campo non sia già presente in tabella.
- **join** (\$table, \$condition, \$option)
Definizione delle Join.
- **union** (\$queries, \$options=array())
Permette di combinare delle istruzioni SELECT.
- **restore** (\$table, \$filename, \$options=array())
Restore di un file (ad es.

- **dump** (\$table, \$path_to_file, \$options=array())

Dump di una tabella.

- **escapeString** (\$string)

Aggiunge le sequenze di escape ai caratteri speciali in una stringa per l'uso in una istruzione SQL, tenendo conto dell'attuale set di caratteri della connessione.

8.31.1 Detailed Description

Interfaccia per le librerie di connessione al database.

Definisce i metodi che le librerie di connessione al database devono implementare.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.31.2 Constructor & Destructor Documentation

8.31.2.1 __construct (\$params)

Costruttore.

Parameters

array	\$params	parametri di connessione al db
-------	----------	--------------------------------

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3 Member Function Documentation

8.31.3.1 autoIncValue (\$table)

Valore di Auto Increment.

Parameters

string	\$table	nome della tabella
--------	---------	--------------------

Returns

auto increment

Implemented in [pdo](#), [mysql](#), [sqlsrv](#), [pdo_mysql](#), and [pdo_sqlsrv](#).

8.31.3.2 begin ()

Istruzione begin.

Returns

bool, risultato istruzione

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3.3 `changeFieldType ($data_type, $value)`

Reimposta il corretto tipo di dato di un campo quando il valore recuperato da una istruzione `select` è di un tipo non corrispondente (vedi `PDO_SQLSRV`)

Parameters

string	<i>\$data_type</i>	tipo di dato come definito da Gino.Model::dataType()
mixed	<i>\$value</i>	valore del campo di un record (Gino.Model::\$_p)

Returns

mixed

Implemented in [pdo](#), [sqlsrv](#), [mysql](#), and [pdo_sqlsrv](#).

8.31.3.4 closeConnection ()

Chiude la connessione al database.

Returns

void

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3.5 columnHasValue (*\$table*, *\$field*, *\$value*, *\$options* = array())

Controlla che il valore del campo non sia già presente in tabella.

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$field</i>	campo da cercare
string	<i>\$value</i>	valore da confrontare
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">except_id (integer): valore ID del record per il quale non effettuare il controllo

Returns

TRUE se presente, FALSE altrimenti

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.6 commit ()

Istruzione commit.

Returns

bool risultato istruzione

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3.7 concat (*\$sequence*)

Istruzione per concatenare i campi.

Esempi:

```
$this->_db->concat(array("label", "' ('", "server", "')'"));
$this->_db->concat(array("lastname", "' '", "firstname"))
```

Parameters

mixed	<i>\$sequence</i>	elenco di campi da concatenare
-------	-------------------	--------------------------------

Returns

CONCAT statement

Implemented in [pdo](#), [sqlsrv](#), [mysql](#), [pdo_mysql](#), and [pdo_sqlsrv](#).

8.31.3.8 conformFieldType (*\$type*)

Uniforma i tipi di dato dei campi.

Elenco dei tipi di dato validi nella definizione di un campo

- *char*, input form
- *text*, textarea form
- *int*, input form (se length>1) o radio button (length=1)
- *bool*, radio button
- *date*, input form di tipo data

Parameters

mixed	<i>\$type</i>	tipo di dato come definito dalle singole librerie
-------	---------------	---

Returns

string (char|text|int|bool|date)

Implemented in [pdo](#), [pdo_sqlsrv](#), [sqlsrv](#), [mysql](#), and [pdo_mysql](#).

8.31.3.9 delete (*\$table*, *\$where*, *\$debug*)

Costruisce ed esegue una query di eliminazione.

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$where</i>	condizione della query
boolean	<i>\$debug</i>	se vero stampa a video la query

Returns

risultato operazione, bool

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.10 distinct (*\$fields*, *\$options*)

Distinct keyword.

Parameters

string	<i>\$fields</i>	nome/nomi dei campi separati da virgola
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • alias (string): nome dell'alias del distinct • remove_table (boolean): rimuove il nome della tabella dalla definizione del campo

Returns

DISTINCT statement

Implemented in [pdo](#), [sqlsrv](#), [mysql](#), [pdo_mysql](#), and [pdo_sqlsrv](#).

8.31.3.11 drop (*\$table*)

Eliminazione di una tabella.

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

risultato dell'operazione, bool

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.12 dump (*\$table*, *\$path_to_file*, *\$options* = array())

Dump di una tabella.

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$path_to_file</i>	nome del file completo di percorso
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • where (string): condizioni della query • delim (string): stringa che viene usata per separare tra loro i valori dei campi • enclosed (string): carattere utilizzato per racchiudere i valori di tipo stringa

Returns

stringa (nome del file di dump)

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.13 dumpDatabase (*\$file*)

Effettua il dump del database.

Parameters

string	<i>\$file</i>	percorso completo del file da scrivere
--------	---------------	--

Returns

risultato operazione, bool

Implemented in [pdo](#), [sqlsrv](#), [mysql](#), [pdo_sqlsrv](#), and [pdo_mysql](#).

8.31.3.14 `escapeString ($string)`

Aggiunge le sequenze di escape ai caratteri speciali in una stringa per l'uso in una istruzione SQL, tenendo conto dell'attuale set di caratteri della connessione.

Parameters

mixed	<i>\$string</i>	
-------	-----------------	--

Returns

stringa

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.15 `execCustomQuery ($query, $options)`

Esegue una query di selezione passando direttamente la query.

Parameters

string	<i>\$query</i>	istruzione sql da eseguire
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • statement (string): tipologia di query <ul style="list-style-type: none"> – <i>select</i> (default) – <i>action</i> • opzioni del metodo select()

Returns

array (select statement) or boolean

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.16 `fieldInformations ($table)`

Recupera le informazioni sui campi di una tabella.

Utilizzato dalla classe [Gino.Options](#) per costruire il form delle opzioni di una classe.

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

informazioni in array

Implemented in [pdo](#), [sqlsrv](#), and [mysql](#).

8.31.3.17 freeresult (*\$result*)

Libera tutta la memoria utilizzata dal set di risultati.

Parameters

array	<i>\$result</i>	risultato della query
-------	-----------------	-----------------------

Returns

bool, risultato

Implemented in [pdo](#), [sqlsrv](#), and [mysql](#).

8.31.3.18 getFieldFromId (*\$table*, *\$field*, *\$field_id*, *\$id*, *\$options*)

Ricava il valore di un campo a una data condizione.

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$field</i>	nome del campo del quale occorre ricavare il valore
string	<i>\$field_id</i>	nome del campo condizione (where)
mixed	<i>\$id</i>	valore del campo condizione (where)
array	<i>\$options</i>	

Returns

valore del campo

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3.19 getFieldsName (*\$table*)

Recupera il nome dei campi di una tabella.

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

array con i nomi dei campi

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.20 getInfoQuery ()

Restituisce informazioni sull'esecuzione delle query.

Returns

informazioni

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3.21 getLastId (\$table)

Valore dell'ultimo ID generato da una colonna Auto Increment a seguito di una istruzione INSERT o UPDATE.

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

ultimo id generato

Implemented in [pdo](#), [sqlsrv](#), and [mysql](#).

8.31.3.22 getNumRecords (\$table, \$where, \$field, \$options)

Numero di record interessati da una query di selezione.

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$where</i>	condizione della query
string	<i>\$field</i>	nome del campo di selezione per il conteggio dei record
array	<i>\$options</i>	opzioni del metodo select()

Returns

numero di record

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.23 getTableStructure (\$table)

Informazioni sulla struttura di una tabella del database.

L'array deve essere così strutturato:

```
array(  
    'primary_key' => string 'primary_key_name'  
    'keys' =>  
        array (key_name[, ...])  
    'fields' =>  
        array (field_1=>array (size=10), field_2=>array (size=10)[, ...])  
)
```

Per ogni campo vengono definite le chiavi:

- **order** (string): the ordinal position
- **default** (null or mixed): the default value
- **null** (string): whether the field is nullable or not ('NO' or 'YES')
- **type** (string): the field type (varchar, int, text, ...); must return a compatible value with those defined in `Model::dataType()`
- **max_length** (null or string): the field max length (es. '200' in varchar field)
- **n_int** (string): the number of int digits
- **n_precision** (integer): the number of decimal digits

- **key** (string): the field key if set (ex. 'PRI' for primary key, 'UNI' for unique key)
- **extra** (string): extra information (ex. auto_increment for an auto-increment field)
- **enum** (null or string): valori di un campo enumerazione (es. "yes','no")

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

array di informazioni

Implemented in [pdo](#), [sqlsrv](#), [mysql](#), [pdo_sqlsrv](#), and [pdo_mysql](#).

8.31.3.24 insert (*\$fields*, *\$table*, *\$debug*)

Costruisce ed esegue una query di inserimento.

Parameters

array	<i>\$fields</i>	elenco dei campi con i loro valori; se il valore di un campo è un array, con la chiave <i>sql</i> è possibile passare una istruzione SQL <ul style="list-style-type: none">• array(field1=>value1, field2=>value2), esempio: <pre>array('name'=>\$name, 'description'=>\$description, 'room'=>\$room)</pre>• array(field1=>array('sql'=>sql_command), field2=>value2), esempio: <pre>array('label'=>\$label, 'orderList'=>array('sql'=>"orderList-1"))</pre>
string	<i>\$table</i>	nome della tabella
boolean	<i>\$debug</i>	se vero stampa a video la query

Returns

risultato operazione, bool

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.25 join (\$table, \$condition, \$option)

Definizione delle Join.

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$condition</i>	condizione della join
string	<i>\$option</i>	opzione della join <ul style="list-style-type: none">• left• right• outer• inner• left outer• right outer

Returns

join clause

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.26 limit (\$range, \$offset)

Istruzione per limitare i risultati di una query (LIMIT)

Esempio

```
$this->_db->limit(1, 0);
```

Parameters

integer	<i>\$range</i>	numero di elementi da mostrare
integer	<i>\$offset</i>	record di partenza (key)

Returns

LIMIT condition

Implemented in [pdo](#), [sqlsrv](#), [mysql](#), [pdo_mysql](#), and [pdo_sqlsrv](#).

8.31.3.27 multiActionQuery (\$file_content)

Esegue una o più query concatenate dal punto e virgola.

Il metodo viene utilizzato per l'installazione dei pacchetti.

Parameters

string	<i>\$file_content</i>	contenuto del file sql
--------	-----------------------	------------------------

Returns

bool, risultato

Implemented in [pdo](#), [mysql](#), [sqlsrv](#), [pdo_mysql](#), and [pdo_sqlsrv](#).

8.31.3.28 openConnection ()

Apri una connessione al database.

Returns

TRUE in caso di successo, genera un'eccezione altrimenti

Implemented in [mysql](#), and [sqlsrv](#).

8.31.3.29 query (\$fields, \$tables, \$where, \$options)

Costruisce una query di selezione.

Parameters

mixed	<i>\$fields</i>	elenco dei campi
mixed	<i>\$tables</i>	elenco delle tabelle
string	<i>\$where</i>	condizione della query
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • order (string): ordinamento • group_by (string): elenco dei campi da raggruppare (istruzione GROUP BY) • distinct (string): nome/nomi dei campi sui quali applicare la keyword DISTINCT • limit (mixed): limitazione degli elementi <ul style="list-style-type: none"> – string, condizione di limitazione degli elementi – array, valori per il range di limitazione (array(offset, range)) • debug (boolean): se vero stampa a video la query

Returns

query

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.30 restore (\$table, \$filename, \$options = array())

Restore di un file (ad es.

di un backup)

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$filename</i>	nome del file da importare
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • fields (array): nomi dei campi • delim (string): stringa che viene usata per separare tra loro i valori dei campi • enclosed (string): stringa utilizzata per racchiudere i valori di tipo stringa • escaped (string): carattere di escape, cioè quello utilizzato prima dei caratteri speciali • lineend (string): stringa utilizzata come separatore tra i record • hasheader (boolean): indica se il file comincia con una riga contenente i nomi dei campi

Returns

risultato dell'operazione, bool

Implemented in [sqlsrv](#), [pdo](#), [mysql](#), [pdo_sqlsrv](#), and [pdo_mysql](#).

8.31.3.31 rollback ()

Istruzione rollback.

Returns

bool, risultato istruzione

Implemented in [pdo](#), [mysql](#), and [sqlsrv](#).

8.31.3.32 select (*\$fields*, *\$tables*, *\$where*, *\$options*)

Esegue una query di selezione passando i parametri di costruzione della query.

See also

[query\(\)](#)

Parameters

mixed	<i>\$fields</i>	elenco dei campi
mixed	<i>\$tables</i>	elenco delle tabelle
string	<i>\$where</i>	condizione della query
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • custom_query (string): query completa • cache (boolean): indica se salvare in cache (se abilitata) i risultati della query (default true) • identity_keyword (string): codice identificativo dei dati in cache • time_caching (integer): tempo di durata della cache • opzioni del metodo query()

Returns

array di risultati

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.33 tableexists (*\$table*)

Verifica l'esistenza di una tabella.

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

bool

Implemented in [pdo](#), [mysql](#), [sqlsrv](#), [pdo_mysql](#), and [pdo_sqlsrv](#).

8.31.3.34 union (*\$queries*, *\$options* = array())

Permette di combinare delle istruzioni SELECT.

Le regole di base per la combinazione dei set di risultati di più query tramite l'istruzione UNION:

- Tutte le query devono includere lo stesso numero di colonne nello stesso ordine.
- I tipi di dati devono essere compatibili.

See also

[select\(\)](#)

Parameters

array	<i>\$queries</i>	query da unire (viene seguito l'ordine nell'array)
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• debug (boolean): se vero stampa a video la query• instruction (string): istruzione (default UNION)• cache (boolean): indica se salvare in cache (se abilitata) i risultati della query (default true)

Returns

array di risultati

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

8.31.3.35 update (*\$fields*, *\$table*, *\$where*, *\$debug*)

Costruisce ed esegue una query di aggiornamento.

Parameters

array	<i>\$fields</i>	<p>elenco dei campi con i loro valori; se il valore di un campo è un array, con la chiave <i>sql</i> è possibile passare una istruzione SQL</p> <ul style="list-style-type: none"> array(field1=>value1, field2=>value2), esempio: <pre>array('name'=>\$name, 'description'=>\$description, 'room'=>\$room)</pre> array(field1=>array('sql'=>sql_command), field2=>value2), esempio: <pre>array('label'=>\$label, 'orderList'=>array('sql'=>"orderList-1"))</pre>
string	<i>\$table</i>	nome della tabella
string	<i>\$where</i>	condizione della query
boolean	<i>\$debug</i>	se vero stampa a video la query

Returns

risultato operazione, bool

Implemented in [sqlsrv](#), [pdo](#), and [mysql](#).

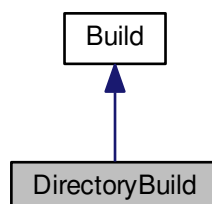
The documentation for this interface was generated from the following file:

- lib/classes/[class.Db.php](#)

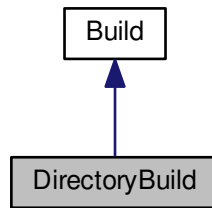
8.32 DirectoryBuild Class Reference

Gestisce i campi di tipo DIRECTORY.

Inheritance diagram for DirectoryBuild:



Collaboration diagram for DirectoryBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getPath](#) ()
Getter della proprietà path.
- [setPath](#) (\$value)
Setter della proprietà path.
- [getPrefix](#) ()
Getter della proprietà prefix.
- [setPrefix](#) (\$value)
Setter della proprietà prefix.
- [clean](#) (\$options=null)
- [delete](#) ()
Eliminazione della directory.

Data Fields

- [\\$_prefix](#)
- [\\$_default_name](#)

Protected Attributes

- [\\$_path](#)
Proprietà dei campi specifiche del tipo di campo.

Private Member Functions

- [defaultName](#) (\$options)
Nome di default della directory.
- [checkName](#) (\$name_dir)
Sostituisce nel nome di una directory i caratteri diversi da [a-zA-Z0-9_].

8.32.1 Detailed Description

Gestisce i campi di tipo DIRECTORY.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.32.2 Constructor & Destructor Documentation

8.32.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.32.3 Member Function Documentation

8.32.3.1 `checkName ($name_dir) [private]`

Sostituisce nel nome di una directory i caratteri diversi da [a-zA-Z0-9_.

-] con il carattere underscore (_)

Se il nome della directory è presente lo salva aggiungendogli un numero progressivo

Parameters

string	<i>\$name_dir</i>	nome della directory
--------	-------------------	----------------------

Returns

nome directory 'friendly'

8.32.3.2 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#) Crea la directory se non esiste

8.32.3.3 `defaultName ($options)` [private]

Nome di default della directory.

Returns

nome directory o null

8.32.3.4 `delete ()`

Eliminazione della directory.

Returns

TRUE o errore

8.32.3.5 `getPath ()`

Getter della proprietà path.

Returns

proprietà path

8.32.3.6 `getPrefix ()`

Getter della proprietà prefix.

Returns

proprietà prefix

8.32.3.7 `setPath ($value)`

Setter della proprietà path.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.32.3.8 `setPrefix ($value)`

Setter della proprietà prefix.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.32.4 Field Documentation

8.32.4.1 `$_default_name`

8.32.4.2 `$_path` `[protected]`

Proprietà dei campi specifiche del tipo di campo.

8.32.4.3 `$_prefix`

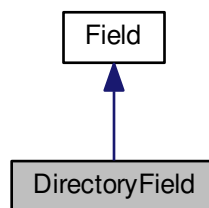
The documentation for this class was generated from the following file:

- `lib/classes/build/class.DirectoryBuild.php`

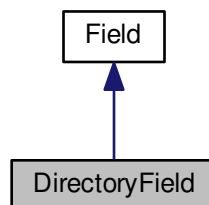
8.33 DirectoryField Class Reference

Campo di tipo DIRECTORY.

Inheritance diagram for DirectoryField:



Collaboration diagram for DirectoryField:



Public Member Functions

- `__construct` (`$options`)

Costruttore.

- [getProperties](#) ()
- [valueFromDb](#) (\$value)

Data Fields

- [\\$_prefix](#)
- [\\$_default_name](#)

Protected Attributes

- [\\$_path](#)

Proprietà dei campi specifiche del tipo di campo.

8.33.1 Detailed Description

Campo di tipo DIRECTORY.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.33.2 Constructor & Destructor Documentation

8.33.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – path (string): percorso assoluto della directory superiore – prefix (string): prefisso da aggiungere al nome della directory – default_name (array): valori per il nome di default <ul style="list-style-type: none"> * <i>field</i> (string): nome dell'input dal quale ricavare il nome della directory (default id) * <i>maxlenth</i> (integer): numero di caratteri da considerare nel nome dell'input (default 15) * <i>value_type</i> (string): tipo di valore (default string)
-------	------------------	---

8.33.3 Member Function Documentation

8.33.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.33.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.33.4 Field Documentation

8.33.4.1 `$_default_name`

8.33.4.2 `$_path` [*protected*]

Proprietà dei campi specifiche del tipo di campo.

8.33.4.3 `$_prefix`

The documentation for this class was generated from the following file:

- `lib/classes/fields/class.DirectoryField.php`

8.34 Document Class Reference

Crea il documento html da inviare come corpo della risposta HTTP.

Public Member Functions

- [__construct](#) (`$url_content`)
Costruttore.
- [__invoke](#) ()
Risposta HTTP con il documento.
- [render](#) ()
Crea il corpo della risposta HTTP.

Data Fields

- [\\$_request](#)
- [\\$_url_content](#)

Private Member Functions

- [getSkin \(\)](#)
Recupera la skin da utilizzare per generare il documento.
- [setHeadVariables \(\\$skin\)](#)
Imposta le variabili del [Gino.Registry](#) utilizzate all'interno del template.
- [parseModules \(\\$m\)](#)
Parserizza un placeholder del template ritornando la pagina o istanza/metodo corrispondenti.
- [errorMessages \(\)](#)
Codice javascript che mostra, se presente, l'errore in sessione.
- [headLine \(\\$skin\)](#)
Headline per template non free.
- [footLine \(\)](#)
Footline per template non free.
- [renderNave \(\\$matches\)](#)
Gestisce gli elementi del layout ricavati dal file di template non free.
- [renderModule \(\\$mdlMarker\)](#)
Gestisce il tipo di elemento da richiamare.
- [modPage \(\\$mdlId\)](#)
Contenuto modulo di tipo pagina.
- [modClass \(\\$mdlId, \\$mdlFunc, \\$mdlType\)](#)
Contenuto modulo di tipo classe.
- [checkOutputFunctionPermissions \(\\$perms, \\$class_name, \\$instance\)](#)
Check dei permessi di accesso ad un metodo di tipo output.
- [modUrl \(\)](#)
Contenuto modulo/pagina richiamato da url.
- [google_analytics \(\)](#)
Codice google analytics.

Private Attributes

- [\\$_registry](#)

8.34.1 Detailed Description

Crea il documento html da inviare come corpo della risposta HTTP.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.34.2 Constructor & Destructor Documentation

8.34.2.1 `__construct ($url_content)`

Costruttore.

Parameters

string	<i>\$url_content</i>	contenuto fornito dal metodo chiamato via url
--------	----------------------	---

Returns

istanza di [Gino.Document](#)

8.34.3 Member Function Documentation

8.34.3.1 `__invoke ()`

Risposta HTTP con il documento.

Returns

[Gino.Http.Response](#) il cui corpo è il documento

8.34.3.2 `checkOutputFunctionPermissions ($perms, $class_name, $instance)` [private]

Check dei permessi di accesso ad un metodo di tipo output.

formati: sysclass_name.perm_name => cerca il permesso perm_name della classe sysclass_name con istanza 0 (compresa la sysclass fittizia 'core') perm_name => cerca il permesso perm_name della classe che definisce outputFunctions con istanza corrente (0 se si tratta di una sysclass)

Returns

bool

8.34.3.3 `errorMessages ()` [private]

Codice javascript che mostra, se presente, l'errore in sessione.

Returns

codice javascript

8.34.3.4 `footLine ()` [private]

Footline per template non free.

Returns

footline

8.34.3.5 `getSkin ()` [private]

Recupera la skin da utilizzare per generare il documento.

La scelta della skin dipende dall'url

Returns

[Gino.Skin](#) oppure una [Exception](#) se la skin non viene trovata

8.34.3.6 `google_analytics()` [private]

Codice google analytics.

Returns

codice javascript

8.34.3.7 `headLine($skin)` [private]

Headline per template non free.

Returns

headline

8.34.3.8 `modClass($mdlId, $mdlFunc, $mdlType)` [private]

Contenuto modulo di tipo classe.

Parameters

int	<i>\$mdlId</i>	id istanza/classe
string	<i>\$mdlFunc</i>	metodo
string	<i>\$mdlType</i>	tipo modulo (sysclass class)

Returns

contenuto

8.34.3.9 `modPage($mdlId)` [private]

Contenuto modulo di tipo pagina.

See also

[Gino.App.Page.page::box\(\)](#)

Parameters

int	<i>\$mdlId</i>	valore ID della pagina
-----	----------------	------------------------

Returns

contenuto pagina

8.34.3.10 `modUrl()` [private]

Contenuto modulo/pagina richiamato da url.

Returns

contenuto

8.34.3.11 `parseModules($m)` [private]

Parserizza un placeholder del template ritornando la pagina o istanza/metodo corrispondenti.

Parameters

string	<i>\$m</i>	placeholder
--------	------------	-------------

Returns

contenuto corrispondente

8.34.3.12 `render ()`

Crea il corpo della risposta HTTP.

Returns

documento html

8.34.3.13 `renderModule ($mdlMarker) [private]`

Gestisce il tipo di elemento da richiamare.

See also

[modPage\(\)](#)
[modClass\(\)](#)
[modUrl\(\)](#)

Parameters

string	<i>\$mdlMarker</i>	placeholder
--------	--------------------	-------------

Returns

contenuto o [Exception](#) se il modulo non viene riconosciuto

8.34.3.14 `renderNave ($matches) [private]`

Gestisce gli elementi del layout ricavati dal file di template non free.

See also

[renderModule\(\)](#)

Parameters

array	<i>\$matches</i>	<ul style="list-style-type: none">• \$matches[0] complete matching• \$matches[1] match open tag, es.• \$matches[3] match div id, es. nav_1_1• \$matches[4] match div content, es. {module classid=20 func=blockList}• \$matches[5] match close tag, es.
-------	------------------	--

Returns

string

8.34.3.15 setHeadVariables (*\$skin*) [private]

Imposta le variabili del [Gino.Registry](#) utilizzate all'interno del template.

Il registro contiene informazioni per i meta tag, css e javascript

Parameters

\Gino\Skin	<i>\$skin</i>	
------------	---------------	--

Returns

void

8.34.4 Field Documentation

8.34.4.1 \$_registry [private]

8.34.4.2 \$_request

8.34.4.3 \$_url_content

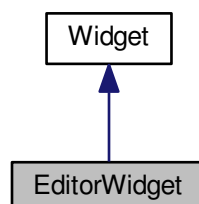
The documentation for this class was generated from the following file:

- lib/classes/[class.Document.php](#)

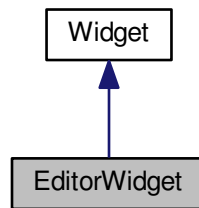
8.35 EditorWidget Class Reference

Campi di tipo editor nei form.

Inheritance diagram for EditorWidget:



Collaboration diagram for EditorWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.35.1 Detailed Description

Campi di tipo editor nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.35.2 Member Function Documentation

8.35.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

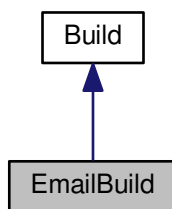
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.EditorWidget.php](#)

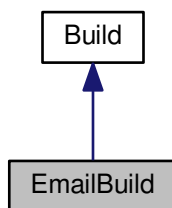
8.36 EmailBuild Class Reference

Campo di tipo EMAIL.

Inheritance diagram for EmailBuild:



Collaboration diagram for EmailBuild:



Public Member Functions

- `__construct` (\$options)
Costruttore.
- `clean` (\$options=array())

Protected Attributes

- `$_trnsI`
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.36.1 Detailed Description

Campo di tipo EMAIL.

Tipologie di input associabili: testo in formato email.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.36.2 Constructor & Destructor Documentation

8.36.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.36.3 Member Function Documentation

8.36.3.1 `clean ($options = array())`

See also

[Gino.Build::clean\(\)](#)

8.36.4 Field Documentation

8.36.4.1 `$_trnsI` [protected]

Proprietà dei campi specifiche del tipo di campo.

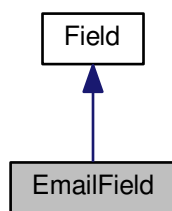
The documentation for this class was generated from the following file:

- lib/classes/build/[class.EmailBuild.php](#)

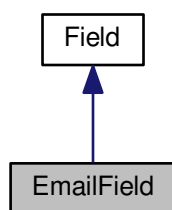
8.37 EmailField Class Reference

Campo di tipo EMAIL.

Inheritance diagram for EmailField:



Collaboration diagram for EmailField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)

Protected Attributes

- [\\$_trnsI](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.37.1 Detailed Description

Campo di tipo EMAIL.

Tipologie di input associabili: testo in formato email.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.37.2 Constructor & Destructor Documentation

8.37.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Field()
-------	------------------	---

Returns

istanza di [Gino.EmailField](#)

8.37.3 Member Function Documentation

8.37.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.37.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.37.4 Field Documentation

8.37.4.1 `$_trnsi` [protected]

Proprietà dei campi specifiche del tipo di campo.

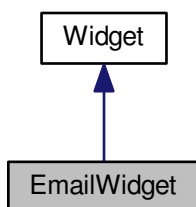
The documentation for this class was generated from the following file:

- lib/classes/fields/[class.EmailField.php](#)

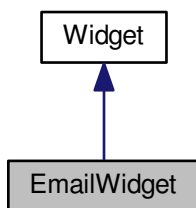
8.38 EmailWidget Class Reference

Campi di tipo email nei form.

Inheritance diagram for EmailWidget:



Collaboration diagram for EmailWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.38.1 Detailed Description

Campi di tipo email nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.38.2 Member Function Documentation

8.38.2.1 printInputForm (\$form, \$options)

See also

[Gino.Widget::printInputForm\(\)](#)

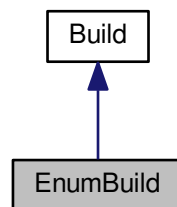
The documentation for this class was generated from the following file:

- lib/classes/widget/class.EmailWidget.php

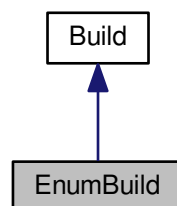
8.39 EnumBuild Class Reference

Gestisce i campi di tipo ENUM.

Inheritance diagram for EnumBuild:



Collaboration diagram for EnumBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.

- [getChoice](#) ()
Getter della proprietà choice.
- [setChoice](#) (\$value)
Setter della proprietà choice.
- [formElement](#) (\Gino\Form \$form, \$options)

Protected Attributes

- [\\$_choice](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.39.1 Detailed Description

Gestisce i campi di tipo ENUM.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.39.2 Constructor & Destructor Documentation

8.39.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.39.3 Member Function Documentation

8.39.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

valore del campo

8.39.3.2 formElement (\Gino\Form \$form, \$options)

See also

[Gino.Build::formElement\(\)](#)

8.39.3.3 getChoice ()

Getter della proprietà choice.

Returns

proprietà choice

8.39.3.4 setChoice (\$value)

Setter della proprietà choice.

Parameters

array	<i>\$value</i>	
-------	----------------	--

Returns

void

8.39.4 Field Documentation

8.39.4.1 \$_choice [protected]

Proprietà dei campi specifiche del tipo di campo.

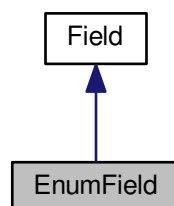
The documentation for this class was generated from the following file:

- lib/classes/build/[class.EnumBuild.php](#)

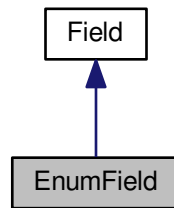
8.40 EnumField Class Reference

Campo di tipo ENUM.

Inheritance diagram for EnumField:



Collaboration diagram for EnumField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)

Protected Attributes

- [\\$_choice](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.40.1 Detailed Description

Campo di tipo ENUM.

Tipologie di input associabili: radio, select

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.40.2 Constructor & Destructor Documentation

8.40.2.1 [__construct](#) (*\$options*)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – choice (array): elenco degli elementi di scelta
-------	------------------	---

8.40.3 Member Function Documentation

8.40.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)8.40.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

8.40.4 Field Documentation

8.40.4.1 `$_choice` [protected]

Proprietà dei campi specifiche del tipo di campo.

The documentation for this class was generated from the following file:

- lib/classes/fields/[class.EnumField.php](#)

8.41 Error Class Reference

Classe per la gestione di errori da mostrare all'utente E' possibile anche mostrare dei warning per facilitare il debug.

Static Public Member Functions

- static [codeMessages](#) ()
Elenco dei codici di errore numerici.
- static [errorMessage](#) (\$message, \$link)
Gestione dell'errore con reindirizzamento a un indirizzo indicato L'errore viene salvato in sessione, avviene quindi il reindirizzamento.
- static [warning](#) (\$message)
Stampa warning se si è in DEBUG TRUE.
- static [getErrorMessage](#) ()
Recupera il messaggio di errore e lo toglie dalla sessione.

8.41.1 Detailed Description

Classe per la gestione di errori da mostrare all'utente E' possibile anche mostrare dei warning per facilitare il debug. I warning vengono mostrati solamente se la costante DEBUG in [configuration.php](#) è settata a FALSE.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.41.2 Member Function Documentation

8.41.2.1 static codeMessages () [static]

Elenco dei codici di errore numerici.

Returns

array associativo codice-errore => stringa-errore

8.41.2.2 static errorMessage (\$message, \$link) [static]

Gestione dell'errore con reindirizzamento a un indirizzo indicato L'errore viene salvato in sessione, avviene quindi il reindirizzamento.

L'errore viene mostrato al nuovo url e poi tolto dalla sessione per evitare ripetizioni. Esempio

```
return \Gino\Error::errorMessage(array('error'=>1), $error_redirect);
```

Parameters

array	<i>\$message</i>	array associativo di opzioni <ul style="list-style-type: none">• error (mixed)<ul style="list-style-type: none">– <i>string</i>: testo personalizzato dell'errore– <i>integer</i>: codice di errore• hint (string): testo dei suggerimenti
string	<i>\$link</i>	collegamento al quale reindirizzare a seguito dell'errore

Returns

[Gino.Http.Redirect](#) alla pagina di errore

8.41.2.3 static getErrorMessage () [static]

Recupera il messaggio di errore e lo toglie dalla sessione.

Returns

messaggio di errore se presente, stringa vuota altrimenti

8.41.2.4 static warning (*\$message*) [static]

Stampa warning se si è in DEBUG TRUE.

Esempio

```
error::warning('modello inesistente');
```

Parameters

string	<i>\$message</i>	messaggio
--------	------------------	-----------

Returns

void

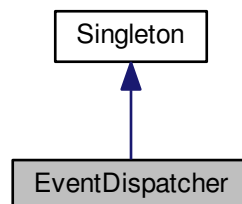
The documentation for this class was generated from the following file:

- lib/classes/[class.Error.php](#)

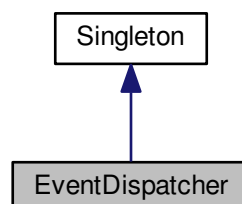
8.42 EventDispatcher Class Reference

Classe di tipo Mediator e [Singleton](#) per la gestione di segnali (eventi) tra oggetti.

Inheritance diagram for EventDispatcher:



Collaboration diagram for EventDispatcher:



Public Member Functions

- [listenEmitter](#) (\$emitter, \$event_name, \$callable)
Registra un listener per l'ascolto di un evento notificato da un emitter.
- [listen](#) (\$event_name, \$callable)
Registra un listener per l'ascolto di un evento notificato da qualunque emitter.
- [emit](#) (\$emitter, \$event_name, \$params=null)
Notifica di un evento da parte di un emitter a tutti i listener.

Private Attributes

- [\\$_listeners](#) = array()
- [\\$_emitter_listeners](#) = array()

Additional Inherited Members

8.42.1 Detailed Description

Classe di tipo Mediator e [Singleton](#) per la gestione di segnali (eventi) tra oggetti.

See also

[Singleton](#)

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.42.2 Member Function Documentation

8.42.2.1 `emit ($emitter, $event_name, $params = null)`

Notifica di un evento da parte di un emitter a tutti i listener.

Parameters

string	<i>\$emitter</i>	nome classe dell'emitter
string	<i>\$event_name</i>	nome evento
mixed	<i>\$params</i>	argomenti ulteriori da passare al listener

8.42.2.2 `listen ($event_name, $callable)`

Registra un listener per l'ascolto di un evento notificato da qualunque emitter.

Parameters

string	<i>\$event_name</i>	nome evento
	<i>callable</i>	un callable, vedi <code>call_user_func</code> . Riceve come parametri l'oggetto che emette, il nome dell'evento ed un parametro passato dall'emitter

Returns

void

8.42.2.3 `listenEmitter ($emitter, $event_name, $callable)`

Registra un listener per l'ascolto di un evento notificato da un emitter.

Parameters

mixed	<i>\$emitter</i>	oggetto che emette il segnale
string	<i>\$event_name</i>	nome evento
	<i>callable</i>	un callable, vedi <code>call_user_func</code> . Riceve come parametri il nome dell'evento ed un parametro passato dall'emitter

Returns

void

8.42.3 Field Documentation

8.42.3.1 `$_emitter_listeners = array()` [private]

8.42.3.2 `$_listeners = array()` [private]

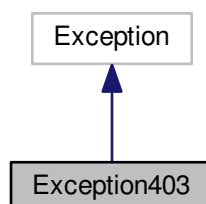
The documentation for this class was generated from the following file:

- [lib/classes/class.EventDispatcher.php](#)

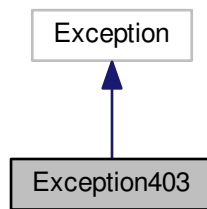
8.43 Exception403 Class Reference

Classe per la gestione di eccezioni 403.

Inheritance diagram for Exception403:



Collaboration diagram for Exception403:



Public Member Functions

- [__construct](#) (array \$kwargs=array())
Costruttore.
- [httpResponse](#) ()
Response con contenuto 403.

Private Attributes

- [\\$_redirect](#)

8.43.1 Detailed Description

Classe per la gestione di eccezioni 403.

Definisce un metodo che fornisce una [Gino.Http.Response](#) con il contenuto 403

See also

[Gino.Http.ResponseForbidden](#)

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.43.2 Constructor & Destructor Documentation

8.43.2.1 `__construct (array $kwargs = array())`

Costruttore.

Parameters

array	<i>\$kwargs</i>	array associativo <ul style="list-style-type: none">• redirect (bool), default FALSE. Se TRUE il metodo Gino.Exception.Exception403::httpResponse() ritorna un Gino.Http.Redirect alla pagina di login
-------	-----------------	--

Returns

istanza [Gino.Exception.Exception403](#)

8.43.3 Member Function Documentation

8.43.3.1 httpResponse ()

Response con contenuto 403.

Returns

[Gino.Http.ResponseForbidden](#)

8.43.4 Field Documentation

8.43.4.1 \$_redirect [private]

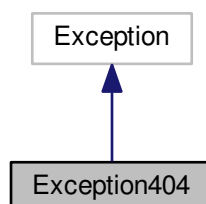
The documentation for this class was generated from the following file:

- lib/classes/exceptions/[class.Exception403.php](#)

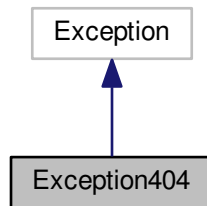
8.44 Exception404 Class Reference

Classe per la gestione di eccezioni 404.

Inheritance diagram for Exception404:



Collaboration diagram for Exception404:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [httpResponse](#) ()
Response con contenuto 404.

8.44.1 Detailed Description

Classe per la gestione di eccezioni 404.

Definisce un metodo che fornisce una [Gino.Http.Response](#) con il contenuto 404

See also

[Gino.Http.ResponseNotFound](#)

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.44.2 Constructor & Destructor Documentation

8.44.2.1 __construct ()

Costruttore.

Returns

istanza [Gino.Exception.Exception404](#)

8.44.3 Member Function Documentation

8.44.3.1 httpResponse ()

Response con contenuto 404.

Returns

[Gino.Http.ResponseNotFound](#)

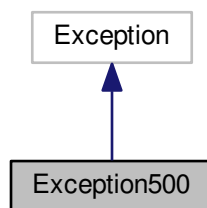
The documentation for this class was generated from the following file:

- [lib/classes/exceptions/class.Exception404.php](#)

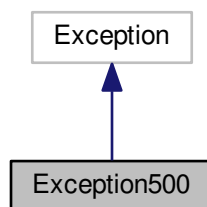
8.45 Exception500 Class Reference

Classe per la gestione di eccezioni 500.

Inheritance diagram for Exception500:



Collaboration diagram for Exception500:



Public Member Functions

- [__construct \(\)](#)
Costruttore.

- [httpResponse](#) ()

Response con contenuto 500.

8.45.1 Detailed Description

Classe per la gestione di eccezioni 500.

Definisce un metodo che fornisce una [Gino.Http.Response](#) con il contenuto 500

See also

[Gino.Http.ResponseServerError](#)

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.45.2 Constructor & Destructor Documentation

8.45.2.1 __construct ()

Costruttore.

Returns

istanza [Gino.Exception.Exception500](#)

8.45.3 Member Function Documentation

8.45.3.1 httpResponse ()

Response con contenuto 500.

Returns

[Gino.Http.ResponseServerError](#)

The documentation for this class was generated from the following file:

- [lib/classes/exceptions/class.Exception500.php](#)

8.46 Export Class Reference

Libreria per l'esportazione di tabelle o dati.

Public Member Functions

- [setTable](#) (\$table)
Imposta la proprietà _table.
- [setSeparator](#) (\$s)
Imposta la proprietà _s.
- [setFields](#) (\$fields)
Imposta la proprietà _fields.
- [setHead](#) (\$head)
Imposta la proprietà _head.
- [setRids](#) (\$rids)
Imposta la proprietà _rids.
- [setOrder](#) (\$order)
Imposta la proprietà _order.
- [setData](#) (\$data)
Imposta la proprietà _data.
- [exportData](#) (\$filename, \$extension, \$output='stream')
Esporta il file.

Protected Member Functions

- [writeFile](#) (\$filename, \$content, \$type)
Crea un file con caratteristiche specifiche di encoding.
- [removeBOM](#) (\$str="")
Rimuove il BOM (Byte Order Mark)

Private Member Functions

- [exportCsv](#) (\$filename, \$output)
Esporta un file csv.
- [getData](#) ()
Dati da esportare in formato array.
- [getHeadFields](#) ()
array di intestazioni delle colonne

Private Attributes

- [\\$_s](#) = ","
- [\\$_table](#)
- [\\$_head](#) = TRUE
- [\\$_fields](#) = '*'
- [\\$_rids](#) = '*'
- [\\$_order](#)
- [\\$_data](#)

8.46.1 Detailed Description

Libreria per l'esportazione di tabelle o dati.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

UTILIZZO

L'utilizzo della libreria prevede l'inclusione del file

```
require_once(CLASSES_DIR.OS.'class.export.php');
```

ESEMPIO

```
$items = array(
    array('value1', 'value2', 'value3'),
    array('value4', 'value5', 'value6')
);

$obj_export = new export();
$obj_export->setData($items);
$obj_export->exportData('export.csv', 'csv');
```

8.46.2 Member Function Documentation

8.46.2.1 exportCsv (\$filename, \$output) [private]

Esporta un file csv.

Parameters

string	<i>\$filename</i>	the name of the file written (the absolute path if the output is file)
string	<i>\$output</i>	(file stream)

Returns

se il file viene inviato in output || TRUE se il file viene salvato su fs

8.46.2.2 exportData (\$filename, \$extension, \$output = 'stream')

Esporta il file.

Attualmente è prevista soltanto l'esportazione di file CSV

See also

`self::exportCsv()`

Parameters

string	<i>\$filename</i>	the name of the file written (the absolute path if the output is file)
string	<i>\$extension</i>	the file extension
string	<i>\$output</i>	(file stream)

Returns

se il file viene inviato in output || TRUE se il file viene salvato su fs

8.46.2.3 `getData ()` [private]

Dati da esportare in formato array.

Returns

array di dati

8.46.2.4 `getHeadFields ()` [private]

array di intestazioni delle colonne

Returns

array intestazioni

8.46.2.5 `removeBOM ($str = ' ')` [protected]

Rimuove il BOM (Byte Order Mark)

Parameters

string	<i>\$str</i>	
--------	--------------	--

Returns

stringa senza BOM

8.46.2.6 `setData ($data)`

Imposta la proprietà `_data`.

Parameters

array	<i>\$data</i>	dati da esportare (parametro competitivo a <code>_table</code>): <pre>array(0=>array("head1", "head2", "head3"), 1=>array("value1 record 1", "value 2 record 1", "value 3 record 1"), 2=>array("value1 record 2", "value 2 record 2", "value 3 record 2"))</pre>
-------	---------------	--

Returns

void

8.46.2.7 `setFields ($fields)`

Imposta la proprietà `_fields`.

Parameters

mixed	<i>\$fields</i>	<div>campi da esportare<ul style="list-style-type: none">• *: tutti i campi• * -(field1,field2): tutti i campi eccetto field1 e field2• field1,field2: solamente i campi field1 e field2• array("field1", "field2"): solamente i campi field1 e field2</div>
-------	-----------------	---

Returns

void

8.46.2.8 setHead (*\$head*)

Imposta la proprietà *_head*.

Parameters

boolean	<i>\$head</i>	indica se mostrare o meno l'intestazione delle colonne
---------	---------------	--

Returns

void

8.46.2.9 setOrder (*\$order*)

Imposta la proprietà *_order*.

Parameters

string	<i>\$order</i>	il campo per l'ordinamento dei risultati
--------	----------------	--

Returns

void

8.46.2.10 setRids (*\$rids*)

Imposta la proprietà *_rids*.

Parameters

mixed	<i>\$rids</i>	<div>id dei record da esportare:<ul style="list-style-type: none">• *: tutti i record• 1,3,5: records con id=1, id=3 e id=5• array(1,3,5): records con id=1, id=3 e id=5</div>
-------	---------------	---

Returns

void

8.46.2.11 setSeparator (\$s)

Imposta la proprietà `_s`.

Parameters

string	<i>\$s</i>	separatore dei campi, default ','
--------	------------	-----------------------------------

Returns

void

8.46.2.12 `setTable ($table)`

Imposta la proprietà `_table`.

Parameters

string	<i>\$table</i>	nome della tabella da esportare
--------	----------------	---------------------------------

Returns

void

8.46.2.13 `writeFile ($filename, $content, $type)` [protected]

Crea un file con caratteristiche specifiche di encoding.

– Procedura di esportazione di un file

1. I valori da database devono passare attraverso le funzioni `utf8_encode()` e `enclosedField()`:

```
$firstname = enclosedField(utf8_encode($b['firstname'])); //-> TESTO
$date = utf8_encode($b['date']); //-> DATA
$number = $b['number']; //-> NUMERO
```

1. Creare il file sul filesystem:

```
$filename = $this->_doc_dir.'/'.$filename;
if (file_exists($filename)) unlink($filename);
$this->writeFile($filename, $output, 'csv');
```

1. Effettuare il download del file:

```
$filename = 'export.csv';
header("Content-type: application/csv \r \n");
header("Content-Disposition: inline; filename=$filename");
echo $output;
exit();
```

Parameters

string	<i>\$filename</i>	percorso assoluto al file
string	<i>\$content</i>	contenuto del file
string	<i>\$type</i>	tipologia di file <ul style="list-style-type: none"> • utf8 • iso8859 • csv: in questo caso utilizzare la funzione <code>utf8_encode()</code> sui valori da DB

Returns

void

8.46.3 Field Documentation

8.46.3.1 `$_data` [private]

8.46.3.2 `$_fields = '*'` [private]

8.46.3.3 `$_head = TRUE` [private]

8.46.3.4 `$_order` [private]

8.46.3.5 `$_rids = '*'` [private]

8.46.3.6 `$_s = ","` [private]

8.46.3.7 `$_table` [private]

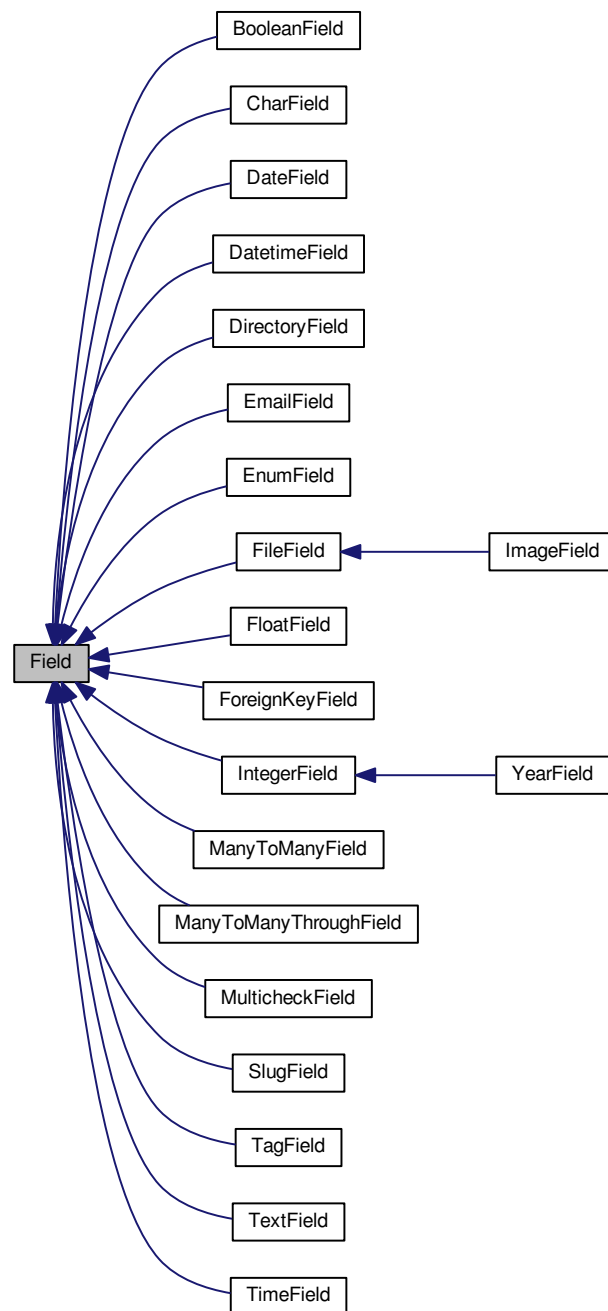
The documentation for this class was generated from the following file:

- [lib/classes/class.Export.php](#)

8.47 Field Class Reference

Gestisce le caratteristiche del tipo di campo (colonne)

Inheritance diagram for Field:



Public Member Functions

- `__construct` (\$options)
Costruttore.
- `__toString` ()
Rappresentazione a stringa dell'oggetto.
- `getName` ()

- Getter della proprietà name.*

 - [setName](#) (\$name)

Setter della proprietà name.
- [getLabel](#) ()

Getter della proprietà label.

 - [setLabel](#) (\$label)

Setter della proprietà label.
- [getDefault](#) ()

Getter della proprietà default.

 - [setDefault](#) (\$default)

Setter della proprietà default.
- [getLenght](#) ()

Getter della proprietà length.

 - [setLenght](#) (\$length)

Setter della proprietà length.
- [getAutoIncrement](#) ()

Getter della proprietà auto_increment.

 - [setAutoIncrement](#) (\$value)

Setter della proprietà auto_increment.
- [getPrimaryKey](#) ()

Getter della proprietà primary_key.

 - [setPrimaryKey](#) (\$value)

Setter della proprietà primary_key.
- [getUniqueKey](#) ()

Getter della proprietà unique_key.

 - [setUniqueKey](#) (\$value)

Setter della proprietà unique_key.
- [getRequired](#) ()

Getter della proprietà required.

 - [setRequired](#) (\$value)

Setter della proprietà required.
- [getWidget](#) ()

Getter della proprietà widget.

 - [setWidget](#) (\$value)

Setter della proprietà widget.
- [getValueType](#) ()

Getter della proprietà value_type.

 - [setValueType](#) (\$value)

Setter della proprietà value_type.
- [getIntDigits](#) ()

Getter della proprietà int_digits (cifre intere)

 - [setIntDigits](#) (\$value)

Setter della proprietà int_digits.
- [getDecimalDigits](#) ()

Getter della proprietà decimal_digits (cifre decimali)

 - [setDecimalDigits](#) (\$value)

Setter della proprietà decimal_digits.
- [getProperties](#) ()

Proprietà definite per il campo.
- [valueFromDb](#) (\$value)

Valore del campo del modello nel formato richiesto.
- [valueToDb](#) (\$value)

Imposta il valore recuperato dal form e ripulito con [Gino.Build::clean\(\)](#).

Data Fields

- [\\$_default](#)
- [\\$_lenght](#)
- [\\$_auto_increment](#)
- [\\$_primary_key](#)
- [\\$_unique_key](#)
- [\\$int_digits](#)
- [\\$decimal_digits](#)

Protected Attributes

- [\\$_name](#)

Proprietà dei campi Vengono esposte dai relativi metodi `__get` e `__set`.

- [\\$_required](#)
- [\\$_widget](#)
- [\\$_default_widget](#)
- [\\$_value_type](#)

8.47.1 Detailed Description

Gestisce le caratteristiche del tipo di campo (colonne)

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.47.2 Constructor & Destructor Documentation

8.47.2.1 __construct (*\$options*)

Costruttore.

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• name (string): nome del campo• label (string): label del campo• default (mixed): valore di default del campo• max_lenght (integer): lunghezza massima del campo• auto_increment (boolean): campo auto_increment• primary_key (boolean): campo chiave primaria• unique_key (boolean): campo chiave unica• required (boolean): valore indicatore del campo obbligatorio• int_digits (integer): numero di cifre intere di un campo float• decimal_digits (integer): numero di cifre decimali di un campo float
-------	------------------	--

8.47.3 Member Function Documentation

8.47.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

valore del campo

8.47.3.2 getAutoIncrement ()

Getter della proprietà auto_increment.

Returns

TRUE se il campo è autoincrement, FALSE altrimenti

8.47.3.3 getDecimalDigits ()

Getter della proprietà decimal_digits (cifre decimali)

Returns

integer decimal digits

8.47.3.4 getDefault ()

Getter della proprietà default.

Returns

valore di default del campo

8.47.3.5 getIntDigits ()

Getter della proprietà int_digits (cifre intere)

Returns

integer

8.47.3.6 getLabel ()

Getter della proprietà label.

Returns

etichetta del campo

8.47.3.7 getLenght ()

Getter della proprietà length.

Returns

lunghezza del campo

8.47.3.8 getName ()

Getter della proprietà name.

Returns

nome del campo

8.47.3.9 getPrimaryKey ()

Getter della proprietà primary_key.

Returns

TRUE se il campo è una chiave primaria, FALSE altrimenti

8.47.3.10 `getProperties ()`

Proprietà definite per il campo.

Returns

array

8.47.3.11 `getRequired ()`

Getter della proprietà `required`.

Returns

TRUE se il campo è obbligatorio, FALSE altrimenti

8.47.3.12 `getUniqueKey ()`

Getter della proprietà `unique_key`.

Returns

TRUE se il campo ha chiave unica, FALSE altrimenti

8.47.3.13 `getValueType ()`

Getter della proprietà `value_type`.

Returns

tipo di dato

8.47.3.14 `getWidget ()`

Getter della proprietà `widget`.

Returns

widget

8.47.3.15 `setAutoIncrement ($value)`

Setter della proprietà `auto_increment`.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.47.3.16 `setDecimalDigits ($value)`

Setter della proprietà `decimal_digits`.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.47.3.17 setDefault (*\$default*)

Setter della proprietà default.

Parameters

mixed	<i>\$default</i>	
-------	------------------	--

Returns

void

8.47.3.18 setIntDigits (*\$value*)

Setter della proprietà int_digits.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.47.3.19 setLabel (*\$label*)

Setter della proprietà label.

Parameters

string	<i>\$label</i>	
--------	----------------	--

Returns

void

8.47.3.20 setLenght (*\$length*)

Setter della proprietà length.

Parameters

int	<i>\$length</i>	
-----	-----------------	--

Returns

void

8.47.3.21 setName (*\$name*)

Setter della proprietà name.

Parameters

string	<i>\$name</i>	
--------	---------------	--

Returns

void

8.47.3.22 setPrimaryKey (*\$value*)

Setter della proprietà primary_key.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.47.3.23 setRequired (*\$value*)

Setter della proprietà required.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.47.3.24 setUniqueKey (*\$value*)

Setter della proprietà unique_key.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.47.3.25 setValueType (*\$value*)

Setter della proprietà value_type.

Parameters

string	<i>\$value</i>	tipo di dato
--------	----------------	--------------

Returns

void

8.47.3.26 `setWidget ($value)`

Setter della proprietà widget.

Parameters

string null	<i>\$value</i>	
---------------	----------------	--

Returns

void

8.47.3.27 valueFromDb (*\$value*)

Valore del campo del modello nel formato richiesto.

See also

[Gino.Model::getProperties\(\)](#)[Gino.Model::fetchColumns\(\)](#)

Parameters

mixed	<i>\$value</i>	
-------	----------------	--

Returns

mixed

8.47.3.28 valueToDb (*\$value*)

Imposta il valore recuperato dal form e ripulito con [Gino.Build::clean\(\)](#).

Il valore viene poi utilizzato per la definizione della query e la gestione dei ManyToMany (

See also

[Gino.Model::__set\(\)](#).

Parameters

mixed	<i>\$value</i>	
-------	----------------	--

Returns

mixed

8.47.4 Field Documentation

8.47.4.1 \$_auto_increment

8.47.4.2 \$_default

8.47.4.3 \$_default_widget [protected]

8.47.4.4 \$_lenght

8.47.4.5 \$_name [protected]

Proprietà dei campi Vengono esposte dai relativi metodi `__get` e `__set`.

8.47.4.6 `$_primary_key`

8.47.4.7 `$_required` `[protected]`

8.47.4.8 `$_unique_key`

8.47.4.9 `$_value_type` `[protected]`

8.47.4.10 `$_widget` `[protected]`

8.47.4.11 `$decimal_digits`

8.47.4.12 `$int_digits`

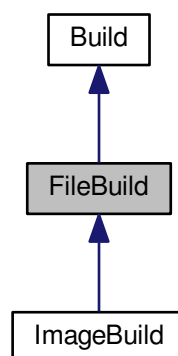
The documentation for this class was generated from the following file:

- `lib/classes/fields/class.Field.php`

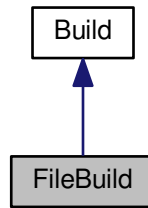
8.48 FileBuild Class Reference

Campo di tipo FILE.

Inheritance diagram for FileBuild:



Collaboration diagram for FileBuild:



Public Member Functions

- [__construct](#) (\$options=array())
Costruttore.
- [getExtensions](#) ()
Getter della proprietà extensions (estensioni accettate)
- [setExtensions](#) (\$value)
Setter della proprietà extensions.
- [getPath](#) ()
Getter della proprietà path_abs (percorso assoluto)
- [setPath](#) (\$value)
Setter della proprietà path.
- [getAddPath](#) ()
Getter della proprietà path_add.
- [setAddPath](#) (\$value)
Setter della proprietà path_add.
- [getPrefix](#) ()
Getter della proprietà prefix.
- [setPrefix](#) (\$value)
Setter della proprietà prefix.
- [getCheckType](#) ()
Getter della proprietà check_type (controllare o meno il mime type)
- [setCheckType](#) (\$value)
Setter della proprietà check_type.
- [getTypesAllowed](#) ()
Getter della proprietà types_allowed (mime types consentiti)
- [setTypesAllowed](#) (\$value)
Setter della proprietà types_allowed.
- [getMaxFileSize](#) ()
Getter della proprietà max_file_size.
- [setMaxFileSize](#) (\$value)
Setter della proprietà max_file_size.
- [getDirectory](#) ()
Getter della proprietà directory.
- [setDirectory](#) (\$value)

Setter della proprietà direcotry.

- `delete` ()

Eliminazione diretta del file.

- `formFilter` (\Gino\Form \$form, \$options=array())
- `filterWhereClause` (\$value)
- `cleanFilter` (\$options)
- `formElement` (\Gino\Form \$form, \$options)
- `clean` (\$options=null)

Data Fields

- `$_path`
- `$_add_path`
- `$_prefix`
- `$_check_type`
- `$_filesize_field`
- `$_types_allowed`
- `$_max_file_size`

Protected Member Functions

- `saveFile` (\$filename, \$filename_tmp)
Salva il file uploadato.
- `pathToFile` (\$options=array())
Ricostruisce il percorso a un file.

Protected Attributes

- `$_extensions`
Proprietà dei campi specifiche del modello.
- `$_directory`
- `$_delete_file`

Private Member Functions

- `checkDeleteFile` (\$input_file, \$check_delete)
Verifica se il file presente è da eliminare.
- `conformPath` (\$directory)
Imposta il separatore di directory come ultimo carattere.
- `checkFilename` (\$filename, \$prefix, \$options=null)
Sostituisce nel nome di un file i caratteri diversi da [a-zA-Z0-9_].

8.48.1 Detailed Description

Campo di tipo FILE.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.48.2 Constructor & Destructor Documentation

8.48.2.1 `__construct ($options = array())`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.48.3 Member Function Documentation

8.48.3.1 `checkDeleteFile ($input_file, $check_delete)` [private]

Verifica se il file presente è da eliminare.

Parameters

string	<i>\$input_file</i>	
boolean	<i>\$check_delete</i>	

Returns

boolean

8.48.3.2 `checkFilename ($filename, $prefix, $options = null)` [private]

Sostituisce nel nome di un file i caratteri diversi da [a-zA-Z0-9_.

-] con il carattere underscore (_)

Se il nome del file è presente lo salva aggiungendogli un numero progressivo

Parameters

string	<i>\$filename</i>	nome del file
string	<i>\$prefix</i>	prefisso da aggiungere al nome del file
array	<i>\$options</i>	array associativo di opzioni in aggiunta a quelle del metodo clean() <ul style="list-style-type: none"> • add_index (boolean) <ul style="list-style-type: none"> – true, aggiunge un numero progressivo al nome del file, ad esempio da foo.1.txt a foo.1.2.txt – false (default), incrementa il numero senza aggiungerlo, ad esempio da foo.1.txt a foo.2.txt

Returns

nome file

8.48.3.3 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#) Effettua l'upload del file

Returns

string or [Exception](#)

8.48.3.4 `cleanFilter ($options)`

See also

[Gino.Build::cleanFilter\(\)](#)

8.48.3.5 `conformPath ($directory)` [private]

Imposta il separatore di directory come ultimo carattere.

Parameters

string	<i>\$directory</i>	nome della directory
--------	--------------------	----------------------

Returns

path directory

8.48.3.6 `delete ()`

Eliminazione diretta del file.

Returns

bool (true) or array (error)

8.48.3.7 `filterWhereClause ($value)`

See also

[Gino.Build::filterWhereClause\(\)](#)

8.48.3.8 `formElement (\Gino\Form $form, $options)`

See also

[Gino.Build::formElement\(\)](#)

8.48.3.9 `formFilter (\Gino\Form $form, $options = array())`

See also

[Gino.Build::formFilter\(\)](#)

8.48.3.10 `getAddPath ()`

Getter della proprietà `path_add`.

Returns

proprietà `path_add`

8.48.3.11 `getCheckType ()`

Getter della proprietà `check_type` (controllare o meno il mime type)

Returns

proprietà `check_type`

8.48.3.12 `getDirectory ()`

Getter della proprietà `directory`.

Returns

proprietà `direcotry`

8.48.3.13 `getExtensions ()`

Getter della proprietà `extensions` (estensioni accettate)

Returns

proprietà `extensions`

8.48.3.14 `getMaxFileSize ()`

Getter della proprietà `max_file_size`.

Returns

proprietà `max_file_size`

8.48.3.15 `getPath ()`

Getter della proprietà `path_abs` (percorso assoluto)

Returns

proprietà `path_abs`

8.48.3.16 `getPrefix ()`

Getter della proprietà `prefix`.

Returns

proprietà `prefix`

8.48.3.17 `getTypesAllowed ()`

Getter della proprietà `types_allowed` (mime types consentiti)

Returns

proprietà `types_allowed`

8.48.3.18 `pathToFile ($options = array())` [protected]

Ricostruisce il percorso a un file.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• type (string): tipo di percorso<ul style="list-style-type: none">– <i>abs</i>: assoluto– <i>rel</i>: relativo• thumb_file (boolean): file thumbnail• complete (boolean): percorso completo col nome del file
-------	------------------	---

Returns

percorso

8.48.3.19 `saveFile ($filename, $filename_tmp)` [protected]

Salva il file uploadato.

Parameters

string	<i>\$filename</i>	nome file
resource	<i>\$filename_tmp</i>	file temporaneo

Returns

bool (true) or array (error)

8.48.3.20 `setAddPath ($value)`

Setter della proprietà `path_add`.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.48.3.21 `setCheckType ($value)`

Setter della proprietà `check_type`.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.48.3.22 setDirectory (*\$value*)

Setter della proprietà direcotry.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.48.3.23 setExtensions (*\$value*)

Setter della proprietà extensions.

Parameters

array	<i>\$value</i>	
-------	----------------	--

Returns

void

8.48.3.24 setMaxFileSize (*\$value*)

Setter della proprietà max_file_size.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.48.3.25 setPath (*\$value*)

Setter della proprietà path.

Parameters

mixed	<i>\$value</i>	<ul style="list-style-type: none">• string• array, array(object controller, string method_name, array method_↔ params)
-------	----------------	---

Returns

void

8.48.3.26 setPrefix (\$value)

Setter della proprietà prefix.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.48.3.27 setTypesAllowed (\$value)

Setter della proprietà types_allowed.

Parameters

array	<i>\$value</i>	
-------	----------------	--

Returns

void

8.48.4 Field Documentation**8.48.4.1 \$_add_path****8.48.4.2 \$_check_type****8.48.4.3 \$_delete_file** [protected]**8.48.4.4 \$_directory** [protected]**8.48.4.5 \$_extensions** [protected]

Proprietà dei campi specifiche del modello.

8.48.4.6 \$_filesize_field**8.48.4.7 \$_max_file_size****8.48.4.8 \$_path****8.48.4.9 \$_prefix****8.48.4.10 \$_types_allowed**

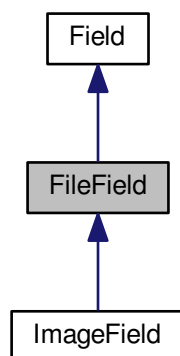
The documentation for this class was generated from the following file:

- lib/classes/build/[class.FileBuild.php](#)

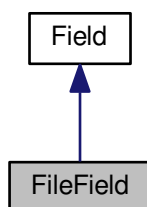
8.49 FileField Class Reference

Campo di tipo FILE.

Inheritance diagram for FileField:



Collaboration diagram for FileField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)

Data Fields

- [\\$_path](#)
- [\\$_add_path](#)
- [\\$_prefix](#)
- [\\$_check_type](#)

- [\\$_types_allowed](#)
- [\\$_max_file_size](#)

Protected Attributes

- [\\$_extensions](#)

Proprietà dei campi specifiche del modello.

8.49.1 Detailed Description

Campo di tipo FILE.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.49.2 Constructor & Destructor Documentation

8.49.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	<p>array associativo di opzioni del campo del database</p> <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – extensions (array): estensioni lecite di file – path (mixed) <ul style="list-style-type: none"> * string, percorso assoluto fino a prima del valore del record ID <pre>\$controller = new auth(); 'path' => \$controller->getBasePath(),</pre> * array <pre>'path' => array('Gino\App\Attachment\AttachmentItem', 'getPath'),</pre> – add_path (string): parte del percorso assoluto dal parametro <i>path</i> fino a prima del file – prefix (string) – check_type (boolean) – filesize_field (string): nome dell'eventuale campo nel quale viene registrata la dimensione del file – types_allowed(array) – max_file_size (integer)
-------	------------------	--

8.49.3 Member Function Documentation

8.49.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.49.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.49.4 Field Documentation

8.49.4.1 `$_add_path`

8.49.4.2 `$_check_type`

8.49.4.3 `$_extensions` `[protected]`

Proprietà dei campi specifiche del modello.

8.49.4.4 `$_max_file_size`

8.49.4.5 `$_path`

8.49.4.6 `$_prefix`

8.49.4.7 `$_types_allowed`

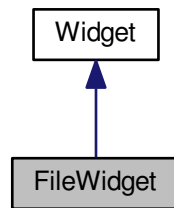
The documentation for this class was generated from the following file:

- `lib/classes/fields/class.FileField.php`

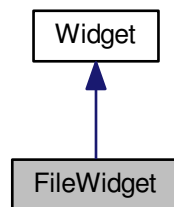
8.50 FileWidget Class Reference

Campi di tipo file nei form.

Inheritance diagram for FileWidget:



Collaboration diagram for FileWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.50.1 Detailed Description

Campi di tipo file nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.50.2 Member Function Documentation

8.50.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

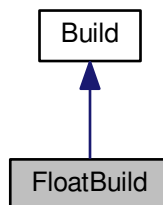
The documentation for this class was generated from the following file:

- [lib/classes/widget/class.FileWidget.php](#)

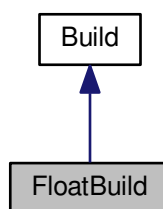
8.51 FloatBuild Class Reference

Gestisce campi di tipo decimale (FLOAT, DOUBLE, DECIMAL)

Inheritance diagram for FloatBuild:



Collaboration diagram for FloatBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [formElement](#) (\Gino\Form \$form, \$options)

Additional Inherited Members

8.51.1 Detailed Description

Gestisce campi di tipo decimale (FLOAT, DOUBLE, DECIMAL)

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.51.2 Constructor & Destructor Documentation

8.51.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.51.3 Member Function Documentation

8.51.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

valore campo

8.51.3.2 `formElement (\Gino\Form $form, $options)`

See also

[Gino.Build::formElement\(\)](#)

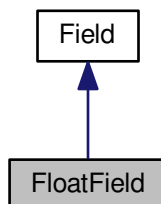
The documentation for this class was generated from the following file:

- lib/classes/build/[class.FloatBuild.php](#)

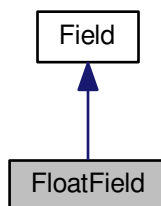
8.52 FloatField Class Reference

Campo di tipo decimale (FLOAT, DOUBLE, DECIMAL)

Inheritance diagram for FloatField:



Collaboration diagram for FloatField:



Public Member Functions

- `__construct` (\$options)
Costruttore.

Additional Inherited Members

8.52.1 Detailed Description

Campo di tipo decimale (FLOAT, DOUBLE, DECIMAL)

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.52.2 Constructor & Destructor Documentation

8.52.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Field()
-------	------------------	---

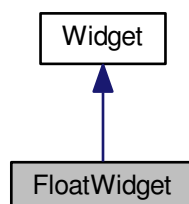
The documentation for this class was generated from the following file:

- lib/classes/fields/[class.FloatField.php](#)

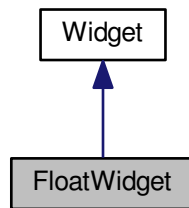
8.53 FloatWidget Class Reference

Campi di tipo float nei form.

Inheritance diagram for FloatWidget:



Collaboration diagram for FloatWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.53.1 Detailed Description

Campi di tipo float nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.53.2 Member Function Documentation

8.53.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

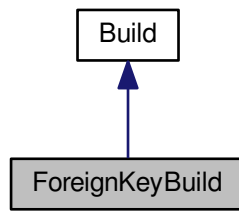
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.FloatWidget.php](#)

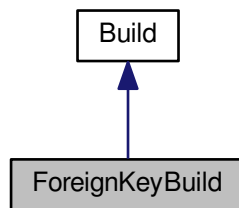
8.54 ForeignKeyBuild Class Reference

Campo di tipo chiave esterna.

Inheritance diagram for ForeignKeyBuild:



Collaboration diagram for ForeignKeyBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [formElement](#) (\Gino\Form \$form, \$options)
- [retrieveValue](#) ()

Data Fields

- [\\$_foreign_where](#)
- [\\$_foreign_order](#)
- [\\$_foreign_controller](#)
- [\\$_add_related](#)
- [\\$_add_related_url](#)

Protected Attributes

- [\\$_foreign](#)
Proprietà dei campi.

8.54.1 Detailed Description

Campo di tipo chiave esterna.

I valori da associare al campo risiedono in una tabella esterna e i parametri per accedervi devono essere definiti nelle opzioni del campo.

Tipologie di input associabili: select, radio

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.54.2 Constructor & Destructor Documentation

8.54.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.54.3 Member Function Documentation

8.54.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

valore campo

8.54.3.2 formElement (\Gino\Form *\$form*, *\$options*)

See also

[Gino.Build::formElement\(\)](#)

8.54.3.3 retrieveValue ()

See also

[Gino.Build::retrieveValue\(\)](#)

Returns

object

8.54.4 Field Documentation

8.54.4.1 `$_add_related`

8.54.4.2 `$_add_related_url`

8.54.4.3 `$_foreign` `[protected]`

Proprietà dei campi.

8.54.4.4 `$_foreign_controller`

8.54.4.5 `$_foreign_order`

8.54.4.6 `$_foreign_where`

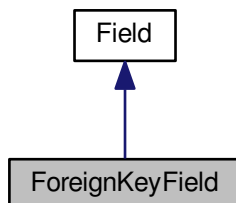
The documentation for this class was generated from the following file:

- `lib/classes/build/class.ForeignKeyBuild.php`

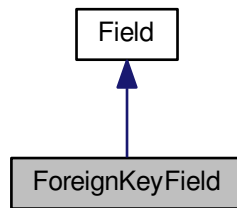
8.55 ForeignKeyField Class Reference

Campo di tipo chiave esterna.

Inheritance diagram for ForeignKeyField:



Collaboration diagram for ForeignKeyField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueToDb](#) (\$value)

Data Fields

- [\\$_foreign_where](#)
- [\\$_foreign_order](#)
- [\\$_add_related_url](#)

Protected Attributes

- [\\$_foreign](#)
Proprietà dei campi specifiche del tipo di campo.
- [\\$_add_related](#)

8.55.1 Detailed Description

Campo di tipo chiave esterna.

I valori da associare al campo risiedono in una tabella esterna e i parametri per accedervi devono essere definiti nelle opzioni del campo.

Tipologie di input associabili: select, radio

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.55.2 Constructor & Destructor Documentation

8.55.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	<p>array associativo di opzioni del campo del database</p> <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe field() <ul style="list-style-type: none"> – foreign (string): nome della classe della chiave esterna – foreign_where (mixed): condizioni della query <ul style="list-style-type: none"> * <i>string</i>, es. "cond1='\$cond1' AND cond2='\$cond2'" * <i>array</i>, es. array("cond1='\$cond1'", "cond2='\$cond2'") – foreign_order (string): ordinamento dei valori (es. name ASC); default 'id' – foreign_controller (object): oggetto del controller della classe della chiave esterna – add_related (boolean) – add_related_url (string)
-------	------------------	---

Returns

istanza di [Gino.ForeignKeyField](#)

8.55.3 Member Function Documentation

8.55.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.55.3.2 `valueToDb ($value)`

See also

[Gino.Field::valueToDb\(\)](#)

Returns

object

8.55.4 Field Documentation

8.55.4.1 `$_add_related` [protected]

8.55.4.2 `$_add_related_url`

8.55.4.3 `$_foreign` [protected]

Proprietà dei campi specifiche del tipo di campo.

8.55.4.4 `$_foreign_order`

8.55.4.5 `$_foreign_where`

The documentation for this class was generated from the following file:

- `lib/classes/fields/class.ForeignKeyField.php`

8.56 Form Class Reference

Classe per la creazione ed il salvataggio dati di un form.

Public Member Functions

- `__construct` (\$formId, \$method, \$validation, \$options=null)
Costruttore.
- `load` (\$session_value, \$clear=TRUE)
Recupera i dati dalla sessione del form.
- `save` (\$session_value)
Salva i valori dei campi del form in una variabile di sessione.
- `retvar` (\$name, \$default= "")
Recupera il valore di un campo del form precedentemente salvato.
- `open` (\$action, \$upload, \$list_required, \$options=null)
Parte iniziale del form, FORM TAG, TOKEN, REQUIRED.
- `close` ()
Chiusura form, /FORM TAG.
- `editorHtml` (\$name, \$value, \$options=null)
Inizializza l'editor visuale CKEditor.
- `label` (\$name, \$text, \$required)
TAG LABEL.
- `arequired` ()
Controlla la compilazione dei campi obbligatori.
- `captcha` (\$options=null)
Widget Captcha.
- `checkCaptcha` ()
Verifica del captcha.
- `noinput` (\$label, \$value, \$options=null)
Simula un campo ma senza input.
- `hidden` (\$name, \$value, \$options=null)
Input hidden.
- `input` (\$name, \$type, \$value, \$options=null)
Input tag.

- **cinput** (\$name, \$type, \$value, \$label, \$options)
Input con label.
- **cinput_date** (\$name, \$value, \$label, \$options)
Input di tipo data con label.
- **ctextarea** (\$name, \$value, \$label, \$options=null)
Tectarea con label.
- **textarea** (\$name, \$value, \$options)
Textarea.
- **cradio** (\$name, \$value, \$data, \$default, \$label, \$options=null)
Input radio con label.
- **radio** (\$name, \$value, \$data, \$default, \$options)
Input radio.
- **ccheckbox** (\$name, \$checked, \$value, \$label, \$options=null)
Input checkbox con label.
- **checkbox** (\$name, \$checked, \$value, \$options)
Input checkbox.
- **multipleCheckbox** (\$name, \$checked, \$data, \$label, \$options=null)
Input checkbox multiplo (many to many) con label.
- **cselect** (\$name, \$value, \$data, \$label, \$options=null)
Input select con label.
- **select** (\$name, \$selected, \$data, \$options)
Input select.
- **cfile** (\$name, \$value, \$label, \$options)
Input file con label.
- **manageFile** (\$name, \$old_file, \$resize, \$valid_extension, \$directory, \$link_error, \$table, \$field, \$idName, \$id, \$options=null)
Gestisce l'upload di un file.
- **savelImage** (\$filename, \$directory, \$prefix_file, \$prefix_thumb, \$new_width, \$new_height, \$thumb_width, \$thumb_height)
Salva le immagini eventualmente ridimensionandole.
- **createlImage** (\$filename, \$directory, \$options=array())
Ridimensiona e crea il thumbnail di una immagine già caricata.
- **jsCountCharText** ()
Funzione javascript che conta il numero dei caratteri ancora disponibili all'interno di un textarea.

Data Fields

- **const** `_IMAGE_GIF_` = 1
- **const** `_IMAGE_JPG_` = 2
- **const** `_IMAGE_PNG_` = 3
- **\$**`_requestVar`
- **\$**`_trd`
- **\$**`_multi_language`
- **\$**`_trnsId`
- **\$**`_input_max`
- **\$**`_prefix_thumb`
- **\$**`_div_field_width`
- **\$**`_textarea_field`
- **\$**`_ckeditor_field`

Private Member Functions

- [setOptions](#) (\$options)
Imposta le opzioni.
- [option](#) (\$opt)
Valore di un'opzione.
- [setMethod](#) (\$method)
Setter della proprietà method.
- [setValidation](#) (\$validation)
Setter della proprietà validation (eseguire o meno la validazione del form)
- [generateFormToken](#) (\$form_name)
Genera un token per prevenire attacchi CSRF.
- [verifyFormToken](#) (\$form_name)
Verifica il token per prevenire attacchi CSRF.
- [reCaptcha](#) (\$public_key, \$options=null)
Captcha widget attraverso la libreria RECAPTCHA.
- [defaultCaptcha](#) (\$options)
Captcha widget attraverso la libreria di gino.
- [checkReCaptcha](#) (\$public_key, \$private_key)
Verifica captcha utilizzando la libreria RECAPTCHA.
- [checkDefaultCaptcha](#) ()
Verifica captcha utilizzando la libreria di gino.
- [imagePreviewer](#) ()
Codice per la visualizzazione allegati contestualmente all'editor CKEDITOR.
- [isImage](#) (\$path)
- [countEqualName](#) (\$file_new, \$file_old, \$resize, \$prefix_file, \$prefix_thumb, \$directory)
Conteggio di file con stesso nome all'interno di \$directory.
- [upload](#) (\$file_tmp, \$file_name, \$uploadaddir)
Upload del file temporaneo nella directory di destinazione.
- [dirUpload](#) (\$directory)
Imposta il carattere '/' come ultimo carattere della directory.
- [checkFilename](#) (\$filename, \$prefix)
Sostituisce nel nome di un file i caratteri diversi da [a-zA-Z0-9_].
- [resizeImage](#) (\$new_width, \$new_height, \$im_width, \$im_height)
Calcola le dimensioni alle quali deve essere ridimensionata una immagine.
- [dimensionFile](#) (\$dimension, \$im_width, \$im_height)
Ricalcola le dimensioni di un'immagine, dimensionando rispetto al lato lungo.
- [formFieldTranslation](#) (\$type, \$tbl, \$field, \$id_value, \$width, \$toolbar="")
Interfaccia che apre o chiude il form per l'inserimento e la modifica delle traduzioni.

Private Attributes

- [\\$_registry](#)
- [\\$_request](#)
- [\\$_method](#)
- [\\$_validation](#)
- [\\$_form_name](#)
- [\\$_options](#)
- [\\$_max_file_size](#)
- [\\$session](#)
- [\\$_lng_trl](#)

- [\\$_trnsI_table](#)
- [\\$_input_size](#)
- [\\$_prefix_file](#)
- [\\$_tbl_attachment_ctg](#)
- [\\$_tbl_attachment](#)
- [\\$_extension_denied](#)
- [\\$_div_label_width](#)
- [\\$_input_field](#)
- [\\$_ico_calendar_path](#)

8.56.1 Detailed Description

Classe per la creazione ed il salvataggio dati di un form.

Fornisce gli strumenti per generare gli elementi del form e per gestire l'upload di file

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.56.2 Constructor & Destructor Documentation

8.56.2.1 __construct (\$formId, \$method, \$validation, \$options = null)

Costruttore.

Parameters

mixed	<i>\$formId</i>	valore ID del form
string	<i>\$method</i>	metodo del form (get/post)
boolean	<i>\$validation</i>	attiva il controllo di validazione tramite javascript
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • trnsI_table (string): nome della tabella per le traduzioni • trnsI_id (integer): riferimento da passare alla tabella per le traduzioni • verifyToken (boolean): verifica il token (contro gli attacchi CSFR) • form_label_width (string): larghezza (%) della colonna con il tag label (default FORM_LABEL_WIDTH) • form_field_width (string): larghezza (%) della colonna con il tag input (default FORM_FIELD_WIDTH)

Exceptions

Exception	se viene rilevato un attacco CSRF
---------------------------	-----------------------------------

Returns

istanza di [Gino.Form](#)

8.56.3 Member Function Documentation

8.56.3.1 arequired ()

Controlla la compilazione dei campi obbligatori.

Returns

numero campi obbligatori non compilati

8.56.3.2 captcha (*\$options* = null)

[Widget Captcha.](#)

Sono previsti due controlli captcha:

1. con le librerie reCAPTCHA (attivo automaticamente se sono state inserite le chiavi pubbliche e private re↔
Captcha nelle 'Impostazioni di sistema')
2. con la classe captcha di gino

See also

self::reCaptcha()
self::defaultCaptcha()

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• classLabel (string): valore CLASS del tag SPAN in <label>• text_add (string): testo che segue il controllo
-------	------------------	--

Returns

widget captcha

8.56.3.3 ccheckbox (*\$name*, *\$checked*, *\$value*, *\$label*, *\$options* = null)

Input checkbox con label.

Esempio

```
$buffer = $gform->ccheckbox('public', $value=='yes'?TRUE:FALSE, 'yes', _("Pubblico"));
```

See also

self::label()
self::checkbox()

Parameters

string	<i>\$name</i>	nome input
boolean	<i>\$checked</i>	valore selezionato
mixed	<i>\$value</i>	valore del tag input
string	<i>\$label</i>	testo <label>
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo checkbox()) <ul style="list-style-type: none"> • required (boolean): campo obbligatorio • classLabel (string): valore CLASS del tag SPAN in <label> • text_add (string): testo da aggiungere dopo il checkbox

Returns

codice html riga form, input + label

8.56.3.4 cfile (*\$name*, *\$value*, *\$label*, *\$options*)

Input file con label.

Integra il checkbox di eliminazione del file e non è gestita l'obbligatorietà del campo.

```
$obj->cfile('image', $filename, _("testo label"), array("extensions"=>array('jpg', ...), "preview"=>TRUE, "
    previewSrc"=>/path/to/image);
```

See also

self::label()

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	nome del file
string	<i>\$label</i>	testo <label>
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo input()) <ul style="list-style-type: none"> • extensions (array): estensioni valide • classLabel (string): valore CLASS del tag SPAN in <label> • preview (boolean): mostra l'anteprima di una immagine • previewSrc (string): percorso relativo dell'immagine • text_add (string): testo da aggiungere in coda al tag input

Returns

codice html riga form, input file + label

8.56.3.5 checkbox (*\$name*, *\$checked*, *\$value*, *\$options*)

Input checkbox.

Parameters

string	<i>\$name</i>	nome input
boolean	<i>\$checked</i>	valore selezionato
mixed	<i>\$value</i>	valore del tag input
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • id (string): valore ID del tag input • classField (string): nome della classe del tag input • js (string): javascript • other (string): altro nel tag

Returns

widget html

8.56.3.6 `checkCaptcha ()`

Verifica del captcha.

See also

`self::checkReCaptcha()`
[self `checkDefaultCaptcha\(\)`](#)

Returns

risultato verifica, bool

8.56.3.7 `checkDefaultCaptcha ()` `[private]`

Verifica captcha utilizzando la libreria di gino.

See also

[Gino.Captcha::check\(\)](#)

Returns

risultato della verifica, bool

8.56.3.8 `checkFilename ($filename, $prefix)` `[private]`

Sostituisce nel nome di un file i caratteri diversi da [a-zA-Z0-9_._-] con il carattere underscore (_)

Parameters

string	<i>\$filename</i>	nome del file
string	<i>\$prefix</i>	prefisso da aggiungere al nome del file

Returns

nome file normalizzato

8.56.3.9 `checkReCaptcha ($public_key, $private_key)` [private]

Verifica captcha utilizzando la libreria RECAPTCHA.

Parameters

string	<i>\$public_key</i>	
string	<i>\$private_key</i>	

Returns

risultato verifica, bool

8.56.3.10 `cinput ($name, $type, $value, $label, $options)`

Input con label.

See also

`self::label()`
`self::formFieldTranslation()`

Parameters

string	<i>\$name</i>	nome input
string	<i>\$type</i>	valore della proprietà <i>type</i> (text)
string	<i>\$value</i>	valore attivo
mixed	<i>\$label</i>	testo <label>
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo input()) <ul style="list-style-type: none"> • required (boolean): campo obbligatorio • classLabel (string): valore CLASS del tag SPAN in <label> • trnsI (boolean): attiva la traduzione • trnsI_table (string): nome della tabella con il campo da tradurre • trnsI_id (integer): valore dell'ID del record di riferimento per la traduzione • field (string): nome del campo con il testo da tradurre • size (integer): lunghezza del tag • text_add (string): testo dopo il tag input

Returns

codice html riga form, label + input

8.56.3.11 `cinput_date ($name, $value, $label, $options)`

Input di tipo data con label.

See also

`self::label()`
`self::input()`

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	valore attivo
mixed	<i>\$label</i>	testo <label>
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo input()) <ul style="list-style-type: none"> • required (boolean): campo obbligatorio • classLabel (string): valore CLASS del tag SPAN in <label> • inputClickEvent (boolean): per attivare l'evento sulla casella di testo • text_add (string): testo dopo il tag input

Returns

codice html riga form, input + label

8.56.3.12 close ()

Chiusura form, /FORM TAG.

Returns

chiusura form, html

8.56.3.13 countEqualName (*\$file_new*, *\$file_old*, *\$resize*, *\$prefix_file*, *\$prefix_thumb*, *\$directory*) [private]

Conteggio di file con stesso nome all'interno di \$directory.

Parameters

string	<i>\$file_new</i>	nome nuovo file
string	<i>\$file_old</i>	nome file precedente
bool	<i>\$resize</i>	
string	<i>\$prefix_file</i>	
string	<i>\$prefix_thumb</i>	
string	<i>\$directory</i>	path

Returns

numero files con stesso nome

8.56.3.14 cradio (*\$name*, *\$value*, *\$data*, *\$default*, *\$label*, *\$options = null*)

Input radio con label.

See also

self::label()

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	valore attivo
array	<i>\$data</i>	elementi dei pulsanti radio (array(value=>text[,]))
mixed	<i>\$default</i>	valore di default
mixed	<i>\$label</i>	testo <label>
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo radio()) <ul style="list-style-type: none"> • required (boolean): campo obbligatorio • classLabel (string): valore CLASS del tag SPAN in <label> • text_add (boolean): testo aggiuntivo stampato sotto il box

Returns

codice html riga form, input radio + label

8.56.3.15 createImage (*\$filename*, *\$directory*, *\$options* = array())

Ridimensiona e crea il thumbnail di una immagine già caricata.

Parameters

string	<i>\$filename</i>	nome del file
string	<i>\$directory</i>	percorso della directory del file
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • prefix_file (string): prefisso del file • prefix_thumb (string): prefisso del thumbnail • width (integer): dimensione in pixel alla quale ridimensionare il file (larghezza) • thumb_width (integer): dimensione in pixel alla quale creare il thumbnail (larghezza)

Returns

risultato operazione, bool o errore

8.56.3.16 cselect (*\$name*, *\$value*, *\$data*, *\$label*, *\$options* = null)

Input select con label.

See also

self::label()

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	elemento selezionato (ad es. valore da 'modifica')
mixed	<i>\$data</i>	elementi del select
mixed	<i>\$label</i>	testo del tag label
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo select()) <ul style="list-style-type: none"> • required (boolean): campo obbligatorio • text_add (string): testo dopo il select • classLabel (string): valore CLASS del tag SPAN in <label>

Returns

codice html riga form, select + label

8.56.3.17 `ctextarea ($name, $value, $label, $options = null)`

Tectarea con label.

See also

`self::label()`
`self::formFieldTranslation()`

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	valore attivo
string	<i>\$label</i>	testo del tag label
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo textarea()) <ul style="list-style-type: none"> • classLabel (string): nome della classe del tag span nel tag label • text_add (string): testo aggiuntivo stampato sotto il box • trnsI (boolean): attiva la traduzione • trnsI_id (integer): valore dell'ID del record di riferimento per la traduzione • trnsI_table (string): nome della tabella con il campo da tradurre • field (string): nome del campo da tradurre • cols (integer): numero di colonne

Returns

codice html riga form, textarea + label

8.56.3.18 `defaultCaptcha ($options)` [`private`]

[Captcha](#) widget attraverso la libreria di gino.

See also

[Gino.Captcha::render\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni
-------	------------------	------------------------------

Returns

widget captcha

8.56.3.19 dimensionFile (*\$dimension*, *\$im_width*, *\$im_height*) [private]

Ricalcola le dimensioni di un'immagine, dimensionando rispetto al lato lungo.

Parameters

int	<i>\$dimension</i>	dimensione ridimensionamento
int	<i>\$im_width</i>	larghezza immagine
int	<i>\$im_height</i>	altezza immagine

Returns

array(larghezza, altezza)

8.56.3.20 dirUpload (*\$directory*) [private]

Imposta il carattere '/' come ultimo carattere della directory.

Parameters

string	<i>\$directory</i>	nome della directory
--------	--------------------	----------------------

Returns

path directory

8.56.3.21 editorHtml (*\$name*, *\$value*, *\$options* = null)

Inizializza l'editor visuale CKEditor.

Parameters

string	<i>\$name</i>	
string	<i>\$value</i>	
array	<i>\$options</i>	<ul style="list-style-type: none"> • toolbar (string): nome della toolbar • width (string): larghezza dell'editor (pixel o %) • height (integer): altezza dell'editor (pixel)

Returns

string, script js

8.56.3.22 `formFieldTranslation ($type, $tbl, $field, $id_value, $width, $toolbar = ' ') [private]`

Interfaccia che apre o chiude il form per l'inserimento e la modifica delle traduzioni.

Viene richiamato nei metodi della classe `Gino.Form`: `cinput()`, `ctextarea()`, `textarea()`

See also

[gino-min.js](#)

Parameters

string	<i>\$type</i>	tipologia di input (input, textarea, fckeditor)
string	<i>\$tbl</i>	nome della tabella con il campo da tradurre
string	<i>\$field</i>	nome del campo con il testo da tradurre
integer	<i>\$id_value</i>	valore dell'ID del record di riferimento per la traduzione
integer	<i>\$width</i>	lunghezza del tag input o numero di colonne (textarea)
string	<i>\$toolbar</i>	nome della toolbar dell'editor html

Returns

codice html interfaccia

8.56.3.23 `generateFormToken ($form_name) [private]`

Genera un token per prevenire attacchi CSRF.

Parameters

string	<i>\$form_name</i>	
--------	--------------------	--

Returns

token

8.56.3.24 `hidden ($name, $value, $options = null)`

Input hidden.

Parameters

string	<i>\$name</i>	nome del tag
mixed	<i>\$value</i>	valore del tag
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> id (string): valore ID del tag

Returns

widget html

8.56.3.25 `imagePreviewer () [private]`

Codice per la visualizzazione allegati contestualmente all'editor CKEDITOR.

See also

[Gino.App.Attachment.attachment::editorList\(\)](#)

Returns

codice html

8.56.3.26 `input ($name, $type, $value, $options = null)`

Input tag.

Parameters

string	<i>\$name</i>	nome input
string	<i>\$type</i>	valore della proprietà <i>type</i> (text)
string	<i>\$value</i>	valore attivo
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• id (string): valore ID del tag• pattern (string): espressione regolare che verifica il valore dell'elemento input• hint (string): placeholder• size (integer): lunghezza del tag• maxlength (integer): numero massimo di caratteri consentito• classField (string): nome della classe del tag• js (string): javascript• readonly (boolean): campo di sola lettura• other (string): altro nel tag

Returns

widget html

8.56.3.27 isImage (\$path) [private]**8.56.3.28 jsCountCharText ()**

Funzione javascript che conta il numero dei caratteri ancora disponibili all'interno di un textarea.

```
$buffer = "<script type=\"text/javascript\" language=\"javascript\">initCounter($('id_elemento'),
    {$this->option('maxlength')})</script>";
```

Returns

codice html

8.56.3.29 label (\$name, \$text, \$required)**TAG LABEL.****Parameters**

string	<i>\$name</i>	nome dell'etichetta
mixed	<i>\$text</i>	testo dell'etichetta, testo o array (array-> array('label'=>_(""..."), 'description'=>_(""...")))
boolean	<i>\$required</i>	campo obbligatorio

Returns

label tag, html

8.56.3.30 load (\$session_value, \$clear = TRUE)

Recupera i dati dalla sessione del form.

Permette di mostrare i campi correttamente compilati a seguito di errore

Parameters

array	<i>\$session_value</i>	nome della variabile di sessione nella quale sono salvati i valori degli input
boolean	<i>\$clear</i>	distrugge la sessione

Returns

global

8.56.3.31 manageFile (\$name, \$old_file, \$resize, \$valid_extension, \$directory, \$link_error, \$table, \$field, \$idName, \$id, \$options = null)

Gestisce l'upload di un file.

Verifica la conformità del file ed effettua l'upload. Eventualmente crea la directory e ridimensiona l'immagine.

See also

```

self::dirUpload()
self::countEqualName()
self::upload()
self::saveImage()

```

Parameters

string	<i>\$name</i>	nome input
string	<i>\$old_file</i>	nome del file esistente
boolean	<i>\$resize</i>	ridimensionamento del file
array	<i>\$valid_extension</i>	estensioni lecite di file
string	<i>\$directory</i>	directory di upload (/path/to/directory/)
string	<i>\$link_error</i>	parametri da aggiungere al reindirizzamento
string	<i>\$table</i>	tabella da aggiornare inserendo il nome del file (UPDATE); se NULL non viene effettuata la query di UPDATE
string	<i>\$field</i>	nome del campo del file; se NULL non viene effettuata la query di UPDATE
string	<i>\$idName</i>	nome del campo ID; se NULL non viene effettuata la query di UPDATE
string	<i>\$id</i>	valore del campo ID; se NULL non viene effettuata la query di UPDATE
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • check_type (boolean): attiva l'opzione <i>types_allowed</i> (TRUE, o 1 per compatibilità => controlla il tipo di file, FALSE => non controllare) • types_allowed (array): array per alcuni tipi di file (mime types) • max_file_size (integer): dimensione massima di un upload (bytes) • thumb (boolean): attiva i thumbnail • prefix (string): per fornire un prefisso a prescindere dal ridimensionamento • prefix_file (string): nel caso <i>resize=TRUE</i> • prefix_thumb (string): nel caso <i>resize=TRUE</i> • width (integer): larghezza alla quale ridimensionare l'immagine • height (integer): altezza alla quale ridimensionare l'immagine • thumb_width (integer): larghezza del thumbnail • thumb_height (integer): altezza del thumbnail • ftp (boolean): permette di inserire il nome del file qualora questo risulti di dimensione superiore al consentito. Il file fisico deve essere poi inserito via FTP • errorQuery (string): query di eliminazione del record qualora non vada a buon fine l'upload del file (INSERT)

Returns

risultato operazione, bool o errori

8.56.3.32 multipleCheckbox (*\$name*, *\$checked*, *\$data*, *\$label*, *\$options* = null)

Input checkbox multiplo (many to many) con label.

Parameters

string	<i>\$name</i>	nome input
array	<i>\$checked</i>	valori degli elementi selezionati
mixed	<i>\$data</i>	<ul style="list-style-type: none"> • string, query • array, elementi del checkbox (value_check=>text)
string	<i>\$label</i>	testo <label>
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • id (string) • classField (string) • readonly (boolean) • js (string) • other (string) • required (string) • classLabel (string) • checkPosition (stringa): posizionamento del checkbox (left) • table (string): nome della tabella con il campo da tradurre • field (mixed): nome o nomi dei campi da recuperare <ul style="list-style-type: none"> – string: nome del campo con il testo da tradurre – array: nomi dei campi da concatenare • idName (string): nome del campo di riferimento • encode_html (boolean): attiva la conversione del testo dal database ad html (default TRUE)

Returns

codice html riga form, input multichexk + label

8.56.3.33 noinput (*\$label*, *\$value*, *\$options* = null)

Simula un campo ma senza input.

Parameters

string	<i>\$label</i>	contenuto della prima colonna <ul style="list-style-type: none"> • string • array, ad esempio array(_("etichetta"), _("spiegazione"))
string	<i>\$value</i>	contenuto della seconda colonna
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • id (string): ID del tag TD della label • style (string): stile del tag TR • other (string): altro nel tag TD della label • class_label (string): classe del tag TD della label • class (string): classe dello span della label

Returns

codice html riga del form

8.56.3.34 `open ($action, $upload, $list_required, $options = null)`

Parte iniziale del form, FORM TAG, TOKEN, REQUIRED.

Per attivare le opzioni **func_confirm** e **text_confirm** occorre istanziare la classe [Form](#) con il parametro validation (TRUE)

Parameters

string	<i>\$action</i>	indirizzo dell'action
boolean	<i>\$upload</i>	attiva l'upload di file
string	<i>\$list_required</i>	lista di elementi obbligatori (separati da virgola)
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • func_confirm (string): nome della funzione js da chiamare (es. window.confirmSend()) • text_confirm (string): testo del messaggio che compare nel box di conferma • generateToken (boolean): costruisce l'input hidden token (contro gli attacchi CSFR)

Returns

parte iniziale del form, html

8.56.3.35 `option ($opt) [private]`

Valore di un'opzione.

Parameters

string	<i>\$opt</i>	opzione da recuperare
--------	--------------	-----------------------

Returns

valore opzione

8.56.3.36 radio (*\$name*, *\$value*, *\$data*, *\$default*, *\$options*)

Input radio.

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	valore attivo
array	<i>\$data</i>	elementi dei pulsanti radio (array(value=>text[,]))
mixed	<i>\$default</i>	valore di default
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • aspect (string): col valore 'v' gli elementi vengono messi uno sotto l'altro • id (string): valore ID del tag <input> • classField (string): valore CLASS del tag <input> • js (string): javascript • other (string): altro nel tag

Returns

widget html

8.56.3.37 reCaptcha (*\$public_key*, *\$options = null*) [private]

Captcha widget attraverso la libreria RECAPTCHA.

Nelle Impostazioni di sistema devono essere state inserite le chiavi pubbliche e private reCaptcha

Parameters

string	<i>\$public_key</i>	
array	<i>\$options</i>	array associativo di opzioni

Returns

widget captcha

8.56.3.38 resizeImage (*\$new_width*, *\$new_height*, *\$im_width*, *\$im_height*) [private]

Calcola le dimensioni alle quali deve essere ridimensionata una immagine.

Parameters

integer	<i>\$new_width</i>	
integer	<i>\$new_height</i>	
integer	<i>\$im_width</i>	
integer	<i>\$im_height</i>	

Returns

array (larghezza, altezza)

8.56.3.39 `retvar ($name, $default = ' ')`

Recupera il valore di un campo del form precedentemente salvato.

See also

[Gino.Form::load\(\)](#)

[Gino.Form::save\(\)](#)

Parameters

string	<i>\$name</i>	nome del campo
mixed	<i>\$default</i>	valore di default

Returns

valore campo

8.56.3.40 `save ($session_value)`

Salva i valori dei campi del form in una variabile di sessione.

Parameters

string	<i>\$session_value</i>	nome della variabile di sessione, come definito nel metodo load()
--------	------------------------	---

Returns

void

8.56.3.41 `savelmage ($filename, $directory, $prefix_file, $prefix_thumb, $new_width, $new_height, $thumb_width, $thumb_height)`

Salva le immagini eventualmente ridimensionandole.

Se **thumb_width** e **thumb_height** sono nulli, il thumbnail non viene generato

Parameters

string	<i>\$filename</i>	nome del file
string	<i>\$directory</i>	percorso della directory del file
string	<i>\$prefix_file</i>	prefisso da aggiungere al file
string	<i>\$prefix_thumb</i>	prefisso da aggiungere al thumbnail
integer	<i>\$new_width</i>	larghezza dell'immagine
integer	<i>\$new_height</i>	altezza dell'immagine
integer	<i>\$thumb_width</i>	larghezza del thumbnail
integer	<i>\$thumb_height</i>	altezza del thumbnail

Returns

risultato operazione, bool

8.56.3.42 `select ($name, $selected, $data, $options)`

Input select.

Parameters

string	<i>\$name</i>	nome input
mixed	<i>\$selected</i>	elemento selezionato
mixed	<i>\$data</i>	elementi del select (query-> recupera due campi, array-> key=>value)
array	<i>\$options</i>	array associativo di opzioni (aggiungere quelle del metodo select()) <ul style="list-style-type: none"> • id (string): ID del tag select • classField (string): nome della classe del tag select • size (integer) • multiple (boolean): scelta multipla di elementi • js (string): utilizzare per eventi javascript (ad es. onchange="jump") • other (string): altro da inserire nel tag select • noFirst (boolean): FALSE-> mostra la prima voce vuota • firstVoice (string): testo del primo elemento • firstValue (mixed): valore del primo elemento • maxChars (integer): numero massimo di caratteri del testo • cutWords (boolean): taglia l'ultima parola se la stringa supera il numero massimo di caratteri

Returns

widget html

8.56.3.43 `setMethod ($method)` [private]

Setter della proprietà method.

Parameters

string	<i>\$method</i>	'post' o 'get'
--------	-----------------	----------------

Returns

void

8.56.3.44 `setOptions ($options)` [private]

Imposta le opzioni.

Parameters

array	<i>\$options</i>	
-------	------------------	--

Returns

void

8.56.3.45 `setValidation ($validation) [private]`

Setter della proprietà validation (eseguire o meno la validazione del form)

Parameters

bool	<i>\$validation</i>	
------	---------------------	--

Returns

void

8.56.3.46 `textarea ($name, $value, $options)`

Textarea.

See also

[imagePreviewer\(\)](#)
[editorHtml\(\)](#)

Parameters

string	<i>\$name</i>	nome input
string	<i>\$value</i>	valore attivo
array	<i>\$options</i>	<p>array associativo di opzioni opzioni del textarea</p> <ul style="list-style-type: none"> • id (string): valore della proprietà id del tag • required (boolean): campo obbligatorio • classField (string): nome della classe del tag • rows (integer): numero di righe • cols (integer): numero di colonne • readonly (boolean): campo di sola lettura • js (string): javascript • other (string): altro nel tag • maxlength (integer): numero massimo di caratteri consentiti • helptext (array) <ul style="list-style-type: none"> – <i>title</i> – <i>text</i> opzioni del tag label • classLabel (string): nome della classe del tag label opzioni dell'editor html • label (string): label • ckeditor (boolean): attiva l'editor html • ckeditor_toolbar (string): nome della toolbar dell'editor html • ckeditor_container (boolean): racchiude l'input editor in un contenitore div • width (string): larghezza dell'editor (pixel o %) • height (integer): altezza dell'editor (pixel) • notes (boolean): mostra le note • img_preview (boolean): mostra il browser di immagini di sistema • text_add (boolean): testo aggiuntivo • trnsI (boolean): attiva la traduzione • trnsI_table (string): nome della tabella con il campo da tradurre • trnsI_id (integer): valore dell'ID del record di riferimento per la traduzione • field (string): nome del campo con il testo da tradurre

Returns

string, codice html

8.56.3.47 `upload ($file_tmp, $file_name, $uploaddir)` [private]

Upload del file temporaneo nella directory di destinazione.

Parameters

resource	<i>\$file_tmp</i>	
string	<i>\$file_name</i>	
string	<i>\$uploaddir</i>	directory di destinazione

Returns

risultato operazione, bool

8.56.3.48 verifyFormToken (*\$form_name*) [private]

Verifica il token per prevenire attacchi CSRF.

Parameters

string	<i>\$form_name</i>	
--------	--------------------	--

Returns

risultato verifica, bool

8.56.4 Field Documentation

8.56.4.1 *\$_div_field_width*8.56.4.2 *\$_div_label_width* [private]8.56.4.3 *\$_extension_denied* [private]8.56.4.4 *\$_fckeditor_field*8.56.4.5 *\$_form_name* [private]8.56.4.6 *\$_ico_calendar_path* [private]8.56.4.7 *\$_input_field* [private]8.56.4.8 *\$_input_max*8.56.4.9 *\$_input_size* [private]8.56.4.10 *\$_lng_trl* [private]8.56.4.11 *\$_max_file_size* [private]8.56.4.12 *\$_method* [private]8.56.4.13 *\$_multi_language*8.56.4.14 *\$_options* [private]8.56.4.15 *\$_prefix_file* [private]8.56.4.16 *\$_prefix_thumb*

- 8.56.4.17 `$_registry` [private]
- 8.56.4.18 `$_request` [private]
- 8.56.4.19 `$_requestVar`
- 8.56.4.20 `$_tbl_attachment` [private]
- 8.56.4.21 `$_tbl_attachment_ctg` [private]
- 8.56.4.22 `$_textarea_field`
- 8.56.4.23 `$_trd`
- 8.56.4.24 `$_trnsI_id`
- 8.56.4.25 `$_trnsI_table` [private]
- 8.56.4.26 `$_validation` [private]
- 8.56.4.27 `$session` [private]
- 8.56.4.28 `const _IMAGE_GIF_ = 1`
- 8.56.4.29 `const _IMAGE_JPG_ = 2`
- 8.56.4.30 `const _IMAGE_PNG_ = 3`

The documentation for this class was generated from the following file:

- [lib/classes/class.Form.php](#)

8.57 Frontend Class Reference

Libreria per la gestione dei file di front-end dei singoli moduli (css e viste)

Public Member Functions

- [__construct](#) (\$controller)
Costruttore.
- [manageFrontend](#) ()
Interfaccia per la gestione dei file di front-end dei moduli.

Data Fields

- [\\$_class](#)
- [\\$_module_id](#)
- [\\$_view_list](#)

Private Member Functions

- [setLists](#) ()

Imposta le liste di file css e viste.

- [fileName](#) (\$file, \$ext)

Nome del file.

- [pathToFile](#) (\$code)

Percorso assoluto della directory dei file di front-end.

- [moduleList](#) (\$code)

Tabella con lista elementi del modulo, css o viste.

- [formModuleFile](#) (\$code, \$request)

Form di modifica file.

- [actionModuleFile](#) (\$code, \$request)

Processa il form di modifica di un file.

Private Attributes

- [\\$_registry](#)
- [\\$_module](#)
- [\\$_css_list](#)
- [\\$_mdlLink](#)

8.57.1 Detailed Description

Libreria per la gestione dei file di front-end dei singoli moduli (css e viste)

Possono essere selezionati e modificati i file con le seguenti caratteristiche:

- i file css definiti nel metodo `getClassElements()` della classe del modulo
- i file delle viste presenti nella directory `views` presente nella directory dell'applicazione

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.57.2 Constructor & Destructor Documentation

8.57.2.1 `__construct ($controller)`

Costruttore.

Parameters

<code>\Gino\↔ Controller</code>	<code>\$controller</code>	istanza della classe di tipo Gino.Controller
-------------------------------------	---------------------------	--

Returns

istanza di [Gino.Frontend](#)

8.57.3 Member Function Documentation

8.57.3.1 `actionModuleFile ($code, $request)` [private]

Processa il form di modifica di un file.

Parameters

string	<i>\$code</i>	'css' o 'view'

8.57.3.2 fileName (*\$file*, *\$ext*) [private]

Nome del file.

I [Css](#) e le viste vengono copiate quando si crea una nuova istanza di un modulo. Il nome viene modificato per comprendere anche il nome dell'istanza.

Parameters

string	<i>\$file</i>	nome del file
string	<i>\$ext</i>	estensione del file

Returns

nome file

8.57.3.3 formModuleFile (*\$code*, *\$request*) [private]

[Form](#) di modifica file.

Parameters

string	<i>\$code</i>	'css' o 'view'

8.57.3.4 manageFrontend ()

Interfaccia per la gestione dei file di front-end dei moduli.

See also

```
self::moduleList()
self::formModuleFile()
self::actionModuleFile()
```

Returns

interfaccia

8.57.3.5 moduleList (*\$code*) [private]

Tabella con lista elementi del modulo, css o viste.

Utilizza la libreria javascript CodeMirror

Parameters

string	<i>\$code</i>	'css' o 'view'
--------	---------------	----------------

Returns

codice html

8.57.3.6 `pathToFile ($code)` [private]

Percorso assoluto della directory dei file di front-end.

Parameters

string	<i>\$code</i>	
--------	---------------	--

Returns

percorso

8.57.3.7 setLists () [private]

Imposta le liste di file css e viste.

Returns

void

8.57.4 Field Documentation

8.57.4.1 \$_class

8.57.4.2 \$_css_list [private]

8.57.4.3 \$_mdlLink [private]

8.57.4.4 \$_module [private]

8.57.4.5 \$_module_id

8.57.4.6 \$_registry [private]

8.57.4.7 \$_view_list

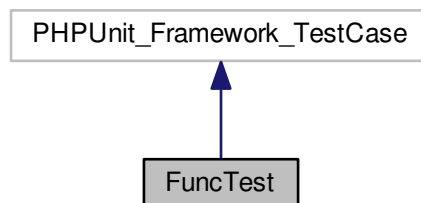
The documentation for this class was generated from the following file:

- lib/classes/[class.Frontend.php](#)

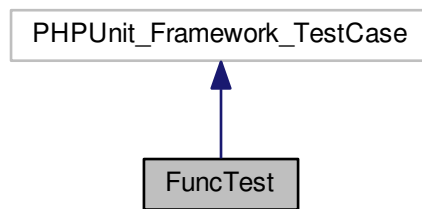
8.58 FuncTest Class Reference

Classe di tipo PHPUnit_Framework_TestCase per testare il file [func.php](#).

Inheritance diagram for FuncTest:



Collaboration diagram for FuncTest:



Public Member Functions

- [test_relativePath \(\)](#)
- [test_absolutePath \(\)](#)
- [test_gOpt \(\)](#)
- [test_arrayToObject \(\)](#)
- [test_searchNameFile \(\)](#)
- [test_extension \(\)](#)
- [test_checkEmail \(\)](#)
- [test_dateToDbDate \(\)](#)
- [test_dbDateToDate \(\)](#)
- [test_dbDatetimeToDate \(\)](#)
- [test_dbDatetimeToTime \(\)](#)
- [test_timeToDbTime \(\)](#)
- [test_dbNumberToNumber \(\)](#)
- [test_numberToDB \(\)](#)
- [test_timeDiff \(\)](#)
- [test_dateDiff \(\)](#)
- [test_getDateDiff \(\)](#)
- [test_isValid \(\)](#)
- [test_cutHtmlText \(\)](#)
- [test_cutString \(\)](#)
- [test_baseFilename \(\)](#)
- [test_traslitterazione \(\)](#)

8.58.1 Detailed Description

Classe di tipo PHPUnit_Framework_TestCase per testare il file [func.php](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.58.2 Member Function Documentation

8.58.2.1 `test_absolutePath ()`

8.58.2.2 `test_arrayToObject ()`

8.58.2.3 `test_baseFilename ()`

8.58.2.4 `test_checkEmail ()`

8.58.2.5 `test_cutHtmlText ()`

8.58.2.6 `test_cutString ()`

8.58.2.7 `test_dateDiff ()`

8.58.2.8 `test_dateToDbDate ()`

8.58.2.9 `test_dbDatetimeToDate ()`

8.58.2.10 `test_dbDatetimeToTime ()`

8.58.2.11 `test_dbDateToDate ()`

8.58.2.12 `test_dbNumberToNumber ()`

8.58.2.13 `test_extension ()`

8.58.2.14 `test_getDateDiff ()`

8.58.2.15 `test_gOpt ()`

8.58.2.16 `test_isValid ()`

8.58.2.17 `test_numberToDB ()`

8.58.2.18 `test_relativePath ()`

8.58.2.19 `test_searchNameFile ()`

8.58.2.20 `test_timeDiff ()`

8.58.2.21 `test_timeToDbTime ()`

8.58.2.22 `test_traslitterazione ()`

The documentation for this class was generated from the following file:

- [test/func_test.php](#)

8.59 GImage Class Reference

Classe per il trattamento di immagini.

Public Member Functions

- [__construct](#) (\$abspath)
Costruttore.
- [getPath](#) ()
Ritorna il percorso relativo dell'immagine (da usare come attributo src del tag img)
- [getWidth](#) ()
Ritorna la larghezza dell'immagine.
- [getHeight](#) ()
Ritorna l'altezza dell'immagine.
- [getResource](#) ()
Ritorna la resource dell'immagine.
- [save](#) (\$abspath=null, \$compression=75, \$permissions=null)
Salva l'immagine su filesystem.
- [stream](#) (\$compression=75)
Output dell'immagine.
- [resize](#) (\$width, \$height, \$options)
- [crop](#) (\$width, \$height, \$xo, \$yo, \$options=array())
Crop dell'immagine con larghezza, altezza e punto iniziali dati.
- [cropCenter](#) (\$width, \$height, \$options=array())
Crop centrale dell'immagine con larghezza e altezza dati.
- [cropEntropy](#) (\$width, \$height, \$options=array())
Crop dell'immagine con larghezza e altezza dati nella zona a massima entropia.
- [thumb](#) (\$width, \$height, \$options=array())
Genera una thumb delle dimensioni Le thumb sono generate al volo e tenute in cache su db.
- [resizeImage](#) (\$image, \$width, \$height, \$options=array())
Resize dell'immagine alle dimensioni fornite.

Data Fields

- [\\$_dir](#) = GIMAGE_DIR
- [\\$_abspath](#)
- [\\$_image](#)
- [\\$_width](#)
- [\\$_height](#)
- [\\$_image_type](#)
- [\\$_tmp_image](#)

Private Member Functions

- [toKey](#) (\$abspath, \$width, \$height, \$options)
Genera una key univoca per un'operazione eseguita su un'immagine.
- [getThumbFromKey](#) (\$key)
Recupera il path dell'immagine se già stata sottoposta alla stessa operazione (cache)
- [makeThumb](#) (\$key, \$width, \$height, \$options)
Genera una thumb delle dimensioni fornite Se viene fornita solo una dimensione oppure entrambe le dimensioni fornite rispecchiano il ratio dell'immagine originale viene effettuato un resize.
- [saveTmpImage](#) (\$key, \$compression=75)
Salva l'immagine temporanea su filesystem e su db.
- [cropImageEntropy](#) (\$image, \$width, \$height, \$options)
Crop dell'immagine nella parte con maggiore entropia.

- **blackThresholdImage** (\$image, \$rt, \$gt, \$bt)
Converte ogni px con rgb maggiore di una soglia a nero.
- **cloneImage** (\$image)
Clona una risorsa immagine.
- **slice** (\$image, \$target_size, \$axis)
slice
- **cropImage** (\$image, \$width, \$height, \$x0, \$y0, \$options=array())
Brief Ritorna il crop di un'immagine.
- **grayscaleEntropy** (\$image)
Calcola l'entropia di un'immagine.
- **getImageHistogram** (\$image)
Ricava una array di frequenze di tonalità di grigio dell'immagine.
- **getEntropy** (\$histogram, \$area)
Calcola l'entropia dato l'istogramma di frequenze di colori.

Private Attributes

- **\$_table** = 'sys_gimage'

8.59.1 Detailed Description

Classe per il trattamento di immagini.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.59.2 Constructor & Destructor Documentation

8.59.2.1 __construct (\$abspath)

Costruttore.

Parameters

string	<i>\$abspath</i>	percorso assoluto del file
--------	------------------	----------------------------

Returns

istanza di [Gino.GImage](#)

8.59.3 Member Function Documentation

8.59.3.1 blackThresholdImage (\$image, \$rt, \$gt, \$bt) [private]

Converte ogni px con rgb maggiore di una soglia a nero.

Parameters

resource	<i>\$image</i>	image resource
int	<i>\$rt</i>	red threshold
int	<i>\$gt</i>	green threshold
int	<i>\$bt</i>	blue threshold

Returns

void

8.59.3.2 cloneImage (*\$image*) [private]

Clona una risorsa immagine.

Parameters

resource	<i>\$image</i>	risorsa
----------	----------------	---------

Returns

resource clone

8.59.3.3 crop (*\$width*, *\$height*, *\$xo*, *\$y0*, *\$options* = array())

Crop dell'immagine con larghezza, altezza e punto iniziali dati.

Parameters

int	<i>\$width</i>	Larghezza crop
int	<i>\$height</i>	Altezza crop
int	<i>\$xo</i>	Coordinata x punto top left di taglio
int	<i>\$yo</i>	Coordinata y punto top left di taglio
array	<i>\$options</i>	Opzioni.

Returns

void

8.59.3.4 cropCenter (*\$width*, *\$height*, *\$options* = array())

Crop centrale dell'immagine con larghezza e altezza dati.

Parameters

int	<i>\$width</i>	Larghezza crop
int	<i>\$height</i>	Altezza crop
array	<i>\$options</i>	Opzioni.

Returns

void

8.59.3.5 cropEntropy (*\$width*, *\$height*, *\$options* = array())

Crop dell'immagine con larghezza e altezza dati nella zona a massima entropia.

Parameters

int	<i>\$width</i>	Larghezza crop
int	<i>\$height</i>	Altezza crop
array	<i>\$options</i>	Opzioni.

Returns

void

8.59.3.6 `cropImage ($image, $width, $height, $x0, $y0, $options = array())` [private]

Brief Ritorna il crop di un'immagine.

Parameters

resource	<i>\$image</i>	Immagine da croppare
int	<i>\$width</i>	larghezza immagine croppata
int	<i>\$height</i>	altezza immagine croppata
int	<i>\$x0</i>	coordinata x dalla quale partire a tagliare
int	<i>\$y0</i>	coordinata y dalla quale partire a tagliare
array	<i>\$options</i>	Opzioni.

Returns

resource immagine croppata

8.59.3.7 `cropImageEntropy ($image, $width, $height, $options)` [private]

Crop dell'immagine nella parte con maggiore entropia.

Parameters

resource	<i>\$image</i>	resource immagine
int	<i>\$width</i>	larghezza crop
int	<i>\$height</i>	altezza crop
array	<i>\$options</i>	Opzioni.

Returns

resource immagine

8.59.3.8 `getEntropy ($histogram, $area)` [private]

Calcola l'entropia dato l'istogramma di frequenze di colori.

Parameters

array	<i>\$histogram</i>	istogramma di frequenze di colori
int	<i>\$area</i>	area dell'immagine

Returns

float entropia

8.59.3.9 getHeight ()

Ritorna l'altezza dell'immagine.

Returns

altezza immagine in px

8.59.3.10 getImageHistogram (\$image) [private]

Ricava una array di frequenze di tonalità di grigio dell'immagine.

Parameters

resource	<i>\$image</i>	resource dell'immagine
----------	----------------	------------------------

Returns

array istogramma

8.59.3.11 getPath ()

Ritorna il percorso relativo dell'immagine (da usare come attributo src del tag img)

See also

[Gino.relativePath](#)

Returns

path relativo immagine

8.59.3.12 getResource ()

Ritorna la resource dell'immagine.

Returns

resource immagine

8.59.3.13 getThumbFromKey (\$key) [private]

Recupera il path dell'immagine se già stata sottoposta alla stessa operazione (cache)

Parameters

string	<i>\$key</i>	chiave univoca dell'operazione
--------	--------------	--------------------------------

Returns

[Gino.GImage](#)|FALSE oggetto [GImage](#) dell'immagine in cache o false

8.59.3.14 `getWidth ()`

Ritorna la larghezza dell'immagine.

Returns

larghezza immagine in px

8.59.3.15 `grayscaleEntropy ($image)` `[private]`

Calcola l'entropia di un'immagine.

Parameters

resource	<i>\$image</i>	resource immagine
----------	----------------	-------------------

Returns

float entropia

8.59.3.16 `makeThumb ($key, $width, $height, $options)` `[private]`

Genera una thumb delle dimensioni fornite. Se viene fornita solo una dimensione oppure entrambe le dimensioni fornite rispecchiano il ratio dell'immagine originale viene effettuato un resize.

Altrimenti viene effettuato un resize prima ed un crop dopo, cercando di tagliare la parte di immagine con maggiore entropia.

Parameters

string	<i>\$key</i>	Chiave univoca che identifica la thumb da generare
int null	<i>\$width</i>	Larghezza della thumb
int null	<i>\$height</i>	Altezza della thumb
array	<i>\$options</i>	Opzioni. Array associativo di opzioni: <ul style="list-style-type: none"> 'allow_enlarge': default false. Consente l'allargamento di immagini per soddisfare le dimensioni richieste.

Returns

[Gino.GImage](#) nuovo oggetto [GImage](#) della thumb generata

8.59.3.17 `resize ($width, $height, $options)`8.59.3.18 `resizeImage ($image, $width, $height, $options = array())`

Resize dell'immagine alle dimensioni fornite.

Parameters

resource	<i>\$image</i>	resource dell'immagine
int	<i>\$width</i>	Larghezza della thumb
int	<i>\$height</i>	Altezza della thumb
array	<i>\$options</i>	Opzioni.

Returns

resource immagine ridimensionata

8.59.3.19 `save ($abspath = null, $compression = 75, $permissions = null)`

Salva l'immagine su filesystem.

Parameters

string	<i>\$abspath</i>	percorso (default il percorso originale dell'immagine)
int	<i>\$compression</i>	compressione, default 75
string	<i>\$permission</i>	permessi

Returns

void

8.59.3.20 `saveTmpImage ($key, $compression = 75)` [private]

Salva l'immagine temporanea su filesystem e su db.

Returns

[Gino.GImage](#) oggetto [GImage](#) della nuova immagine salvata

8.59.3.21 `slice ($image, $target_size, $axis)` [private]

slice

Parameters

resource	<i>\$image</i>	
int	<i>\$target_size</i>	dimensione finale
string	<i>\$axis</i>	asse h=orizzontale, v=verticale

Returns

coordinata dalla quale tagliare

8.59.3.22 `stream ($compression = 75)`

Output dell'immagine.

```
ob_start();
$image->stream();
$i = ob_get_clean();
echo "<img src='data:image/jpeg;base64,'" . base64_encode( $i ) . "'>";
```

Parameters

int	<i>\$compression</i>	compressione, default 75
-----	----------------------	--------------------------

Returns

stream immagine

8.59.3.23 `thumb ($width, $height, $options = array())`

Genera una thumb delle dimensioni Le thumb sono generate al volo e tenute in cache su db.

Se viene richiesta una thumb già creata viene direttamente restituita. Altrimenti viene creata.

Parameters

int null	<i>\$width</i>	Larghezza della thumb
int null	<i>\$height</i>	Altezza della thumb
array	<i>\$options</i>	Opzioni. Array associativo di opzioni: <ul style="list-style-type: none"> 'allow_enlarge': default false. Consente l'allargamento di immagini per soddisfare le dimensioni richieste.

Returns

[Gino.GImage](#) nuovo oggetto immagine wrapper della thumb generata

8.59.3.24 toKey (\$abspath, \$width, \$height, \$options) [private]

Genera una key univoca per un'operazione eseguita su un'immagine.

Parameters

string	<i>\$abspath</i>	percorso assoluto dell'immagine
int	<i>\$width</i>	Larghezza dell'immagine dopo l'operazione
int	<i>\$height</i>	Altezza dell'immagine dopo l'operazione array \$options Opzioni

Returns

chiave univoca

8.59.4 Field Documentation

8.59.4.1 \$_abspath

8.59.4.2 \$_dir = GIMAGE_DIR

8.59.4.3 \$_height

8.59.4.4 \$_image

8.59.4.5 \$_image_type

8.59.4.6 \$_table = 'sys_gimage' [private]

8.59.4.7 \$_tmp_image

8.59.4.8 \$_width

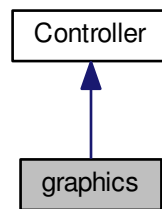
The documentation for this class was generated from the following file:

- lib/classes/[class.GImage.php](#)

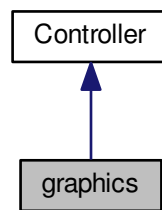
8.60 graphics Class Reference

Classe di tipo [Gino.Controller](#) per la gestione personalizzata degli header e footer del sistema.

Inheritance diagram for graphics:



Collaboration diagram for graphics:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [printHeaderPublic](#) ()
Header con valore ID 1.
- [printHeaderPrivate](#) ()
Header con valore ID 2.
- [printHeaderAdmin](#) ()
Header con valore ID 3.
- [printHeaderMobile](#) ()
Header con valore ID 4.
- [printHeaderAdhoc](#) ()
Header con valore ID 5.
- [printFooterPublic](#) ()
Header con valore ID 6.
- [printFooterPrivate](#) ()
Footer con valore ID 7.
- [printFooterAdmin](#) ()
Footer con valore ID 8.

- [printFooterMobile](#) ()
Footer con valore ID 9.
- [printFooterAdhoc](#) ()
Footer con valore ID 10.
- [manageGraphics](#) (\Gino\Http\Request \$request)
Interfaccia di amministrazione modulo.

Static Public Member Functions

- static [getClassElements](#) ()
Restituisce alcune proprietà della classe.
- static [outputFunctions](#) ()
Elenco dei metodi che possono essere richiamati dal menu e dal template.

Private Member Functions

- [isHeader](#) (\$id)
Dice se l'oggetto rappresenta un header o no.
- [render](#) (\$id)
Stampa l'header/footer.

Private Attributes

- [\\$_title](#)

Additional Inherited Members

8.60.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione personalizzata degli header e footer del sistema.

Sono disponibili 5 header e 5 footer completamente personalizzabili ed utilizzabili nella composizione del layout. Un header/footer può essere di due tipologie:

- grafica, prevede il caricamento di una immagine
- codice, prevede l'inserimento di codice html

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.60.2 Constructor & Destructor Documentation

8.60.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.App.Graphics.graphics](#)

8.60.3 Member Function Documentation

8.60.3.1 `static getClassElements () [static]`

Restituisce alcune proprietà della classe.

Returns

array associativo contenente le viste

8.60.3.2 `isHeader ($id) [private]`

Dice se l'oggetto rappresenta un header o no.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

bool

8.60.3.3 `manageGraphics (\Gino\Http\Request $request)`

Interfaccia di amministrazione modulo.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<i>\$request</i>	istanza di Gino.Http.Request
---	------------------	--

Returns

8.60.3.4 `static outputFunctions () [static]`

Elenco dei metodi che possono essere richiamati dal menu e dal template.

Returns

array, elenco metodi nella forma nome_metodo => array(label => string, permissions => array())

8.60.3.5 printFooterAdhoc ()

Footer con valore ID 10.

Returns

html, footer

8.60.3.6 printFooterAdmin ()

Footer con valore ID 8.

Returns

html, footer

8.60.3.7 printFooterMobile ()

Footer con valore ID 9.

Returns

html, footer

8.60.3.8 printFooterPrivate ()

Footer con valore ID 7.

Returns

html, footer

8.60.3.9 printFooterPublic ()

Header con valore ID 6.

Returns

html, header

8.60.3.10 printHeaderAdhoc ()

Header con valore ID 5.

Returns

html, header

8.60.3.11 printHeaderAdmin ()

Header con valore ID 3.

Returns

html, header

8.60.3.12 printHeaderMobile ()

Header con valore ID 4.

Returns

html, header

8.60.3.13 printHeaderPrivate ()

Header con valore ID 2.

Returns

html, header

8.60.3.14 printHeaderPublic ()

Header con valore ID 1.

Returns

html, header

8.60.3.15 render(\$id) [private]

Stampa l'header/footer.

Parameters

integer	\$id	valore ID del record
---------	------	----------------------

Returns

html

8.60.4 Field Documentation

8.60.4.1 \$_title [private]

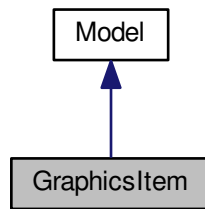
The documentation for this class was generated from the following file:

- [app/graphics/class_graphics.php](#)

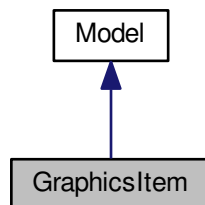
8.61 GraphicsItem Class Reference

Classe di tipo [Gino.Model](#) che rappresenta un header/footer.

Inheritance diagram for GraphicsItem:



Collaboration diagram for GraphicsItem:



Public Member Functions

- `__construct ($id)`
Costruttore.
- `__toString ()`
Rappresentazione a stringa dell'oggetto.

Static Public Member Functions

- static `columns ()`
Struttura dei campi della tabella di un modello.

Static Public Attributes

- static `$table` = "sys_graphics"
- static `$columns`

Static Private Attributes

- static `$_extension_img` = array("jpg", "png", "gif")

Additional Inherited Members

8.61.1 Detailed Description

Classe di tipo [Gino.Model](#) che rappresenta un header/footer.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.61.2 Constructor & Destructor Documentation

8.61.2.1 `__construct ($id)`

Costruttore.

Returns

istanza di [Gino.App.Graphics.GraphicsItem](#)

8.61.3 Member Function Documentation

8.61.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

descrizione

8.61.3.2 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.61.4 Field Documentation

8.61.4.1 `$_extension_img = array("jpg", "png", "gif") [static], [private]`

8.61.4.2 `$columns [static]`

8.61.4.3 `$table = "sys_graphics" [static]`

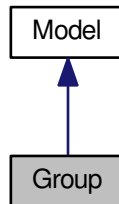
The documentation for this class was generated from the following file:

- [app/graphics/class.GraphicsItem.php](#)

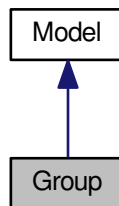
8.62 Group Class Reference

Classe tipo [Gino.Model](#) che rappresenta un gruppo di utenti.

Inheritance diagram for Group:



Collaboration diagram for Group:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [getPermissions](#) ()
Elenco dei permessi di un gruppo.

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.
- static [getList](#) ()
Elenco dei gruppi.
- static [setMergeValue](#) (\$permission_id, \$instance_id)
Valore che raggruppa permesso e istanza.

- static [getMergeValue](#) (\$value)
Splitta i valori di permesso e istanza.

Static Public Attributes

- static [\\$table](#) = TBL_GROUP
- static [\\$table_group_perm](#) = TBL_GROUP_PERMISSION
- static [\\$columns](#)

Additional Inherited Members

8.62.1 Detailed Description

Classe tipo [Gino.Model](#) che rappresenta un gruppo di utenti.

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.62.2 Constructor & Destructor Documentation

8.62.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.App.Auth.Group](#)

8.62.3 Member Function Documentation

8.62.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

nome gruppo

8.62.3.2 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.62.3.3 static getList () [static]

Elenco dei gruppi.

Returns

array array(id, name, description)

8.62.3.4 static getMergeValue (\$value) [static]

Splitta i valori di permesso e istanza.

Parameters

string	<i>\$value</i>	valore da splittare
--------	----------------	---------------------

Returns

array array(permission_id, instance_id)

8.62.3.5 getPermissions ()

Elenco dei permessi di un gruppo.

Parameters

integer	<i>\$id</i>	valore ID del gruppo
---------	-------------	----------------------

Returns

array

8.62.3.6 static setMergeValue (\$permission_id, \$instance_id) [static]

Valore che raggruppa permesso e istanza.

Parameters

integer	<i>\$permission_id</i>	valore ID del permesso
integer	<i>\$instance_id</i>	valore ID dell'istanza

Returns

string

8.62.4 Field Documentation

8.62.4.1 \$columns [static]

8.62.4.2 \$table = TBL_GROUP [static]

8.62.4.3 \$table_group_perm = TBL_GROUP_PERMISSION [static]

The documentation for this class was generated from the following file:

- app/auth/[class.Group.php](#)

8.63 GTag Class Reference

Classe per il trattamento di campi di tipo tag.

Static Public Member Functions

- static [saveContentTags](#) (\$content_controller_class, \$content_controller_instance, \$content_class, \$content_id, \$tags)
Salva i tag su db, sia nella tabella tag che nella tabella di associazione ai contenuti.
- static [getContentTags](#) ()
Ritorna un array di tag associati al contenuto dato Non è necessario inserire nella where clause anche i campi relativi al controller, perché comunque gli oggetti sono unici per id e nome classe.
- static [getAllTags](#) ()
Array di tutti i tag presenti nel sistema.
- static [getRelatedContents](#) (\$content_class, \$content_id)
Fornisce contenuti correlati basandosi su corrispondenza di tag.
- static [getTagsHistogram](#) ()
Isogramma dei tag (array tag->frequenza) Utile per la scrittura di una tag cloud.
- static [deleteTaggedItem](#) (\$controller_class, \$controller_instance, \$model_class, \$model_id)
Elimina le associazioni dei tag ai contenuti.

Static Public Attributes

- static [\\$_table_tag_taggeditem](#) = 'sys_tag_taggeditem'

Static Private Attributes

- static [\\$_table_tag](#) = 'sys_tag'

8.63.1 Detailed Description

Classe per il trattamento di campi di tipo tag.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.63.2 Member Function Documentation

8.63.2.1 static deleteTaggedItem (\$controller_class, \$controller_instance, \$model_class, \$model_id) [static]

Elimina le associazioni dei tag ai contenuti.

Parameters

string	<i>\$controller_class</i>	nome della classe controller del modello cui i tag sono associati
integer	<i>\$controller_↔instance</i>	valore id dell'istanza della classe controller del modello cui i tag sono associati
string	<i>\$model_class</i>	nome della classe del modello cui i tag sono associati
integer	<i>\$model_id</i>	valore id dell'oggetto cui i tag sono associati

Returns

boolean

Example in the model

```
GTag::deleteTaggedItem($this->_controller->getClassName(), $this->_controller->
getInstance(), get_name_class($this), $this->id);
```

8.63.2.2 static getAllTags () [static]

Array di tutti i tag presenti nel sistema.

Returns

array di tag

8.63.2.3 static getContentTags () [static]

Ritorna un array di tag associati al contenuto dato Non è necessario inserire nella where clause anche i campi relativi al controller, perché comunque gli oggetti sono unici per id e nome classe.

Parameters

string	<i>\$content_class</i>	la classe del modello cui i tag sono associati
int	<i>\$content_id</i>	l'id del oggetto cui i tag sono associati

Returns

array di tag

8.63.2.4 static getRelatedContents (\$content_class, \$content_id) [static]

Fornisce contenuti correlati basandosi su corrispondenza di tag.

Parameters

string	<i>\$content_class</i>	la classe dell'oggetto per il quale cercare contenuti correlati
string	<i>\$content_id</i>	la id dell'oggetto per il quale cercare contenuti correlati

Returns

contenuti correlati

8.63.2.5 static getTagsHistogram () [static]

Isrogramma dei tag (array tag->frequeunza) Utile per la scrittura di una tag cloud.

Returns

istogramma tags

8.63.2.6 static saveContentTags (\$content_controller_class, \$content_controller_instance, \$content_class, \$content_id, \$tags) [static]

Salva i tag su db, sia nella tabella tag che nella tabella di associazione ai contenuti.

Parameters

string	<i>\$content_controller_class</i>	nome della classe controller del modello cui i tag sono associati
string	<i>\$content_controller_instance</i>	id dell'istanza della classe controller del modello cui i tag sono associati
string	<i>\$content_class</i>	la classe del modello cui i tag sono associati
int	<i>\$content_id</i>	l'id del oggetto cui i tag sono associati
string	<i>\$tags</i>	stringa di tag separati da virgole

Returns

TRUE

8.63.3 Field Documentation

8.63.3.1 \$_table_tag = 'sys_tag' [static],[private]

8.63.3.2 \$_table_tag_taggeditem = 'sys_tag_taggeditem' [static]

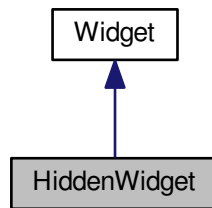
The documentation for this class was generated from the following file:

- lib/classes/[class.GTag.php](#)

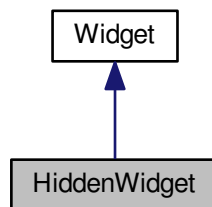
8.64 HiddenWidget Class Reference

Campi di tipo hidden nei form.

Inheritance diagram for HiddenWidget:



Collaboration diagram for HiddenWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.64.1 Detailed Description

Campi di tipo hidden nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.64.2 Member Function Documentation

8.64.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

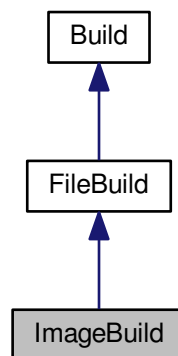
The documentation for this class was generated from the following file:

- [lib/classes/widget/class.HiddenWidget.php](#)

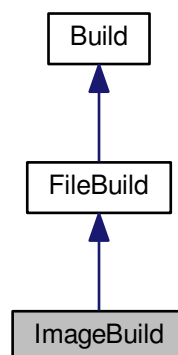
8.65 ImageBuild Class Reference

Gestisce campi di tipo IMMAGINE.

Inheritance diagram for ImageBuild:



Collaboration diagram for ImageBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getResize](#) ()
Getter della proprietà resize (ridimensionare immagine)
- [setResize](#) (\$value)
Setter della proprietà resize.
- [getThumb](#) ()
Getter della proprietà thumb (creazione thumb)
- [setThumb](#) (\$value)
Setter della proprietà thumb.
- [getPrefixFile](#) ()
Getter della proprietà prefix_file.
- [setPrefixFile](#) (\$value)
Setter della proprietà prefix_file.
- [getPrefixThumb](#) ()
Getter della proprietà prefix_thumb.
- [setPrefixThumb](#) (\$value)
Setter della proprietà prefix_thumb.
- [getWidth](#) ()
Getter della proprietà width (larghezza di ridimensionamento)
- [setWidth](#) (\$value)
Setter della proprietà width.
- [getHeighth](#) ()
Getter della proprietà height (altezza di ridimensionamento)
- [setHeight](#) (\$value)
Setter della proprietà height.
- [getThumbWidth](#) ()
Getter della proprietà thumb_width (larghezza thumb)
- [setThumbWidth](#) (\$value)
Setter della proprietà thumb_width.
- [getThumbHeighth](#) ()
Getter della proprietà thumb_height (altezza thumb)
- [setThumbHeight](#) (\$value)
Setter della proprietà thumb_height.
- [formElement](#) (\Gino\Form \$form, \$options)
- [delete](#) ()
Eliminazione immagine.

Data Fields

- const [_IMAGE_GIF_](#) = 1
- const [_IMAGE_JPG_](#) = 2
- const [_IMAGE_PNG_](#) = 3
- [\\$_thumb](#)
- [\\$_prefix_file](#)
- [\\$_prefix_thumb](#)
- [\\$_width](#)
- [\\$_height](#)
- [\\$_thumb_width](#)
- [\\$_thumb_height](#)

Protected Member Functions

- [saveFile](#) (\$filename, \$filename_tmp)
Salvataggio immagine.
- [saveImage](#) (\$filename, \$prefix_file, \$prefix_thumb, \$new_width, \$new_height, \$thumb_width, \$thumb_height)
Salva le immagini eventualmente ridimensionandole.

Protected Attributes

- [\\$_resize](#)
Proprietà dei campi specifiche del modello.

Private Member Functions

- [resizeImage](#) (\$new_width, \$new_height, \$im_width, \$im_height)
Calcola le dimensioni alle quali deve essere ridimensionata una immagine.

8.65.1 Detailed Description

Gestisce campi di tipo IMMAGINE.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.65.2 Constructor & Destructor Documentation

8.65.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Build() • opzioni generali definite come proprietà nella classe FileBuild()
-------	------------------	--

8.65.3 Member Function Documentation

8.65.3.1 `delete ()`

Eliminazione immagine.

See also

[Gino.FileBuild::delete\(\)](#)

8.65.3.2 `formElement (\Gino\Form $form, $options)`

See also

[Gino.Build::formElement\(\)](#)

8.65.3.3 `getHeighth ()`

Getter della proprietà height (altezza di ridimensionamento)

Returns

proprietà height

8.65.3.4 `getPrefixFile ()`

Getter della proprietà prefix_file.

Returns

proprietà prefix_file

8.65.3.5 `getPrefixThumb ()`

Getter della proprietà prefix_thumb.

Returns

proprietà prefix_thumb

8.65.3.6 `getResize ()`

Getter della proprietà resize (ridimensionare immagine)

Returns

proprietà resize

8.65.3.7 `getThumb ()`

Getter della proprietà thumb (creazione thumb)

Returns

proprietà thumb

8.65.3.8 getThumbHeighth ()

Getter della proprietà thumb_height (altezza thumb)

Returns

proprietà thumb_height

8.65.3.9 getThumbWidth ()

Getter della proprietà thumb_width (larghezza thumb)

Returns

proprietà thumb_width

8.65.3.10 getWidth ()

Getter della proprietà width (larghezza di ridimensionamento)

Returns

proprietà width

8.65.3.11 resizeImage (\$new_width, \$new_height, \$im_width, \$im_height) [private]

Calcola le dimensioni alle quali deve essere ridimensionata una immagine.

Parameters

integer	<i>\$new_width</i>	
integer	<i>\$new_height</i>	
integer	<i>\$im_width</i>	
integer	<i>\$im_height</i>	

Returns

array (larghezza, altezza)

8.65.3.12 saveFile (\$filename, \$filename_tmp) [protected]

Salvataggio immagine.

See also

[Gino.FileBuild::saveFile\(\)](#)

8.65.3.13 saveImage (\$filename, \$prefix_file, \$prefix_thumb, \$new_width, \$new_height, \$thumb_width, \$thumb_height) [protected]

Salva le immagini eventualmente ridimensionandole.

Se **thumb_width** e **thumb_height** sono nulli, il thumbnail non viene generato

Parameters

string	<i>\$filename</i>	nome del file
string	<i>\$prefix_file</i>	prefisso da aggiungere al file
string	<i>\$prefix_thumb</i>	prefisso da aggiungere al thumbnail
integer	<i>\$new_width</i>	larghezza dell'immagine
integer	<i>\$new_height</i>	altezza dell'immagine
integer	<i>\$thumb_width</i>	larghezza del thumbnail
integer	<i>\$thumb_height</i>	altezza del thumbnail

Returns

risultato operazione, bool

8.65.3.14 setHeight (*\$value*)

Setter della proprietà height.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.65.3.15 setPrefixFile (*\$value*)

Setter della proprietà prefix_file.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.65.3.16 setPrefixThumb (*\$value*)

Setter della proprietà prefix_thumb.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.65.3.17 setResize (*\$value*)

Setter della proprietà resize.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.65.3.18 setThumb (*\$value*)

Setter della proprietà thumb.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.65.3.19 setThumbHeight (*\$value*)

Setter della proprietà thumb_height.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.65.3.20 setThumbWidth (*\$value*)

Setter della proprietà thumb_width.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.65.3.21 setWidth (*\$value*)

Setter della proprietà width.

Parameters

int	<i>\$value</i>	
-----	----------------	--

Returns

void

8.65.4 Field Documentation

8.65.4.1 `$_height`

8.65.4.2 `$_prefix_file`

8.65.4.3 `$_prefix_thumb`

8.65.4.4 `$_resize` `[protected]`

Proprietà dei campi specifiche del modello.

8.65.4.5 `$_thumb`

8.65.4.6 `$_thumb_height`

8.65.4.7 `$_thumb_width`

8.65.4.8 `$_width`

8.65.4.9 `const _IMAGE_GIF_ = 1`

8.65.4.10 `const _IMAGE_JPG_ = 2`

8.65.4.11 `const _IMAGE_PNG_ = 3`

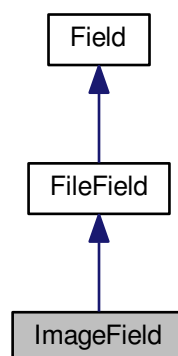
The documentation for this class was generated from the following file:

- `lib/classes/build/class.ImageBuild.php`

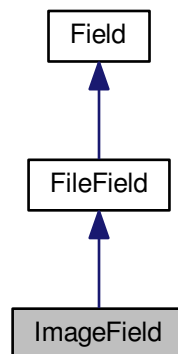
8.66 ImageField Class Reference

Campo di tipo IMMAGINE.

Inheritance diagram for ImageField:



Collaboration diagram for ImageField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()

Data Fields

- [\\$_thumb](#)
- [\\$_prefix_file](#)
- [\\$_prefix_thumb](#)
- [\\$_width](#)
- [\\$_height](#)
- [\\$_thumb_width](#)
- [\\$_thumb_height](#)

Protected Attributes

- [\\$_resize](#)
Proprietà dei campi specifiche del modello.

8.66.1 Detailed Description

Campo di tipo IMMAGINE.

Tipologie di input associabili: input file

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.66.2 Constructor & Destructor Documentation

8.66.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni generali definite come proprietà nella classe FileField() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – resize (boolean) – thumb (boolean) – prefix_file (string) – prefix_thumb (string) – width (integer) – height (integer) – thumb_width (integer) – thumb_height (integer)
-------	------------------	--

8.66.3 Member Function Documentation

8.66.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.66.4 Field Documentation

8.66.4.1 `$_height`

8.66.4.2 `$_prefix_file`

8.66.4.3 `$_prefix_thumb`

8.66.4.4 `$_resize` [protected]

Proprietà dei campi specifiche del modello.

8.66.4.5 `$_thumb`

8.66.4.6 `$_thumb_height`

8.66.4.7 `$_thumb_width`

8.66.4.8 `$_width`

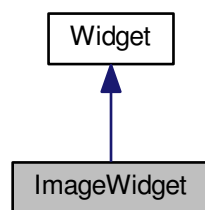
The documentation for this class was generated from the following file:

- `lib/classes/fields/class.ImageField.php`

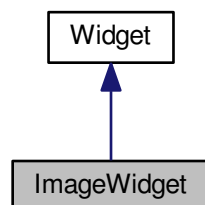
8.67 ImageWidget Class Reference

Campi di tipo immagine nei form.

Inheritance diagram for ImageWidget:



Collaboration diagram for ImageWidget:



Public Member Functions

- `printInputForm` (`$form`, `$options`)

Additional Inherited Members

8.67.1 Detailed Description

Campi di tipo immagine nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.67.2 Member Function Documentation

8.67.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

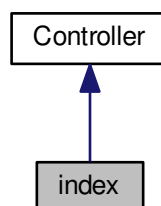
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.ImageWidget.php](#)

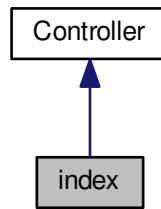
8.68 index Class Reference

Classe di tipo [Gino.Controller](#) del modulo che gestisce la home amministrativa.

Inheritance diagram for index:



Collaboration diagram for index:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [admin_page](#) (\Gino\Http\Request \$request)
Home page amministrazione.
- [sysModulesManageArray](#) (\$request)
Elenco dei moduli di sistema visualizzabili nell'area amministrativa.
- [modulesManageArray](#) (\$request)
Elenco dei moduli non di sistema visualizzabili nell'area amministrativa.

Static Public Member Functions

- static [outputFunctions](#) ()
Elenco dei metodi che possono essere richiamati dal menu.

Additional Inherited Members

8.68.1 Detailed Description

Classe di tipo [Gino.Controller](#) del modulo che gestisce la home amministrativa.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.68.2 Constructor & Destructor Documentation

8.68.2.1 [__construct](#) ()

Costruttore.

Returns

istanza di [Gino.App.Index.index](#)

8.68.3 Member Function Documentation**8.68.3.1 admin_page (\Gino\Http\Request \$request)**

Home page amministrazione.

Parameters

\Gino\↔ Http↔ Request	\$request	istanza di Gino.Http.Request
-----------------------------	-----------	--

Returns

[Gino.Http.Response](#) home page amministrazione

8.68.3.2 modulesManageArray (\$request)

Elenco dei moduli non di sistema visualizzabili nell'area amministrativa.

Returns

array associativo contenente informazioni sui moduli: <module_id> => array(<label>, <name>, <class>, <decription>)

8.68.3.3 static outputFunctions () [static]

Elenco dei metodi che possono essere richiamati dal menu.

Returns

array di metodi pubblici

8.68.3.4 sysModulesManageArray (\$request)

Elenco dei moduli di sistema visualizzabili nell'area amministrativa.

Returns

array associativo contenente informazioni sui moduli: <module_id> => array(<label>, <name>, <decription>)

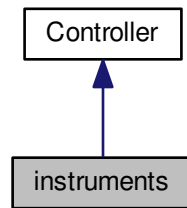
The documentation for this class was generated from the following file:

- [app/index/class_index.php](#)

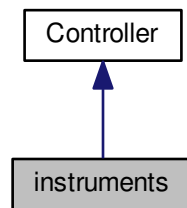
8.69 instruments Class Reference

Classe di tipo [Gino.Controller](#) per la gestione di strumenti aggiuntivi di gino.

Inheritance diagram for instruments:



Collaboration diagram for instruments:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [manageInstruments](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa alla gestione degli strumenti.

Private Member Functions

- [links](#) ()
Strumento - mostra l'elenco delle risorse disponibili (con i relativi link)
- [mimeTypes](#) (\Gino\Http\Request \$request)
Strumento - mostra l'elenco dei mime type (include come iframe il file mime-type-table.html)

Additional Inherited Members

8.69.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione di strumenti aggiuntivi di gino.

Adesso sono disponibili l'elenco delle risorse disponibili (con i relativi link) e dei mime type.
Per aggiungere uno strumento è necessario:

- creare un record nella tabella *instruments*
- associare nel metodo viewItem() il valore del campo id dello strumento con un suo metodo personalizzato (ad es. itemNew)
- creare il metodo personalizzato (ad es. itemNew)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.69.2 Constructor & Destructor Documentation

8.69.2.1 __construct ()

Costruttore.

Returns

[Gino.App.Instruments.instruments](#)

8.69.3 Member Function Documentation

8.69.3.1 links () [private]

Strumento - mostra l'elenco delle risorse disponibili (con i relativi link)

Returns

html, elenco risposte con link

8.69.3.2 manageInstruments (\Gino\Http\Request \$request)

Interfaccia amministrativa alla gestione degli strumenti.

Parameters

\Gino\↔ Http\↔ Request	\$request	istanza di Gino.App.Instruments.instruments
------------------------------	-----------	---

Returns

interfaccia di amministrazione

8.69.3.3 mimeTypees (\Gino\Http\Request \$request) [private]

Strumento - mostra l'elenco dei mime type (include come iframe il file mime-type-table.html)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

html, elenco mime type

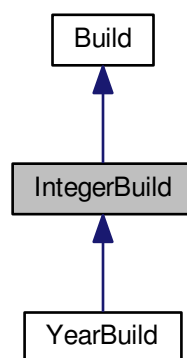
The documentation for this class was generated from the following file:

- app/instruments/[class_instruments.php](#)

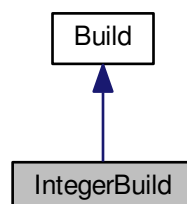
8.70 IntegerBuild Class Reference

Campo di tipo INTERO.

Inheritance diagram for IntegerBuild:



Collaboration diagram for IntegerBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [formElement](#) (\Gino\Form \$form, \$options)

Additional Inherited Members

8.70.1 Detailed Description

Campo di tipo INTERO.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.70.2 Constructor & Destructor Documentation

8.70.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.70.3 Member Function Documentation

8.70.3.1 formElement (\Gino\Form \$form, \$options)

See also

[Gino.Build::formElement\(\)](#)

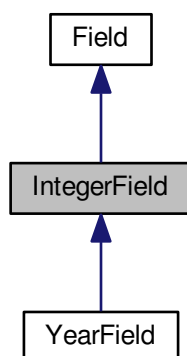
The documentation for this class was generated from the following file:

- lib/classes/build/[class.IntegerBuild.php](#)

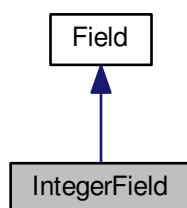
8.71 IntegerField Class Reference

Campo di tipo INTERO.

Inheritance diagram for IntegerField:



Collaboration diagram for IntegerField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [valueFromDb](#) (\$value)
- [valueToDb](#) (\$value)

Additional Inherited Members

8.71.1 Detailed Description

Campo di tipo INTERO.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.71.2 Constructor & Destructor Documentation

8.71.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Field()
-------	------------------	---

Returns

istanza di [Gino.IntegerField](#)

8.71.3 Member Function Documentation

8.71.3.1 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or integer

8.71.3.2 `valueToDb ($value)`

See also

[Gino.Field::valueToDb\(\)](#)

Returns

null or integer

The documentation for this class was generated from the following file:

- lib/classes/fields/[class.IntegerField.php](#)

8.72 Javascript Class Reference

Contiene i metodi per includere alcuni javascript.

Static Public Member Functions

- static [abiMapLib](#) ()
Include il file javascript con la libreria delle mappe.
- static [slimboxLib](#) ()
Include il file javascript con la libreria slimbox.
- static [onLoadFunction](#) (\$skinObj=null)
Funzioni javascript caricate all'interno della sezione HEAD dell'html.
- static [vendor](#) ()
Include librerie js di terze parti.
- static [analytics](#) ()
Include il codice per attivare google analytics.

Static Private Member Functions

- static [sharethis](#) (\$key)
Codice html inclusione libreria sharethis (sharethis.com)

8.72.1 Detailed Description

Contiene i metodi per includere alcuni javascript.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.72.2 Member Function Documentation

8.72.2.1 static [abiMapLib](#) () [static]

Include il file javascript con la libreria delle mappe.

Returns

codice html

8.72.2.2 static [analytics](#) () [static]

Include il codice per attivare google analytics.

Returns

codice html

8.72.2.3 static [onLoadFunction](#) (\$skinObj = null) [static]

Funzioni javascript caricate all'interno della sezione HEAD dell'html.

Parameters

\Gino\Skin	<i>\$skinObj</i>	istanza di Gino.Skin associata alla pagina
------------	------------------	--

Returns

codice html

8.72.2.4 static `sharethis ($key) [static],[private]`

Codice html inclusione libreria sharethis (sharethis.com)

Returns

codice html

8.72.2.5 static `slimboxLib () [static]`

Include il file javascript con la libreria slimbox.

Returns

codice html

8.72.2.6 static `vendor () [static]`

Include librerie js di terze parti.

Returns

codice html

The documentation for this class was generated from the following file:

- lib/classes/[class.Javascript.php](#)

8.73 JSMin Class Reference

PHP implementation of Douglas Crockford's [JSMin](#).

Public Member Functions

- [__construct](#) (\$input)
Constructor.

Static Public Member Functions

- static [minify](#) (\$s)
Minify Javascript.

Data Fields

- const `ORD_LF` = 10
- const `ORD_SPACE` = 32
- const `ACTION_KEEP_A` = 1
- const `ACTION_DELETE_A` = 2
- const `ACTION_DELETE_A_B` = 3

Protected Member Functions

- `action` (\$command)
Action – do something! What to do is determined by the \$command argument.
- `get` ()
Get next char.
- `isAlphaNum` (\$c)
Is \$c a letter, digit, underscore, dollar sign, or non-ASCII character.
- `min` ()
Perform minification, return result.
- `next` ()
Get the next character, skipping over comments.
- `peek` ()
Get next char.

Protected Attributes

- `$a` = ""
- `$b` = ""
- `$input` = ""
- `$inputIndex` = 0
- `$inputLength` = 0
- `$lookAhead` = null
- `$output` = ""

8.73.1 Detailed Description

PHP implementation of Douglas Crockford's [JSMIn](#).

This is pretty much a direct port of jsmin.c to PHP with just a few PHP-specific performance tweaks. Also, whereas jsmin.c reads from stdin and outputs to stdout, this library accepts a string as input and returns another string as output.

PHP 5 or higher is required.

Permission is hereby granted to use this version of the library under the same terms as jsmin.c, which has the following license:

Copyright (c) 2002 Douglas Crockford (www.crockford.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

SOFTWARE.

Author

Ryan Grove ryan@wonko.com

Copyright

2002 Douglas Crockford douglas@crockford.com (jsmin.c)

2008 Ryan Grove ryan@wonko.com (PHP port)

2012 Adam Goforth aag@adamgoforth.com (Updates) <http://opensource.org/licenses/mit-license.php> MIT License

Version

1.1.2 (2012-05-01) <https://github.com/rgrove/jsmin-php>

8.73.2 Constructor & Destructor Documentation

8.73.2.1 `__construct ($input)`

Constructor.

Parameters

string	<i>\$input</i>	Javascript to be minified
--------	----------------	---

8.73.3 Member Function Documentation

8.73.3.1 `action ($command)` [`protected`]

Action – do something! What to do is determined by the `$command` argument.

action treats a string as a single character. Wow! action recognizes a regular expression if it is preceded by (or , or =.

[next\(\)](#) [get\(\)](#)

Exceptions

JSMinException	<p>If parser errors are found:</p> <ul style="list-style-type: none"> • Unterminated string literal • Unterminated regular expression set in regex literal • Unterminated regular expression literal
--------------------------------	---

Parameters

int	<i>\$command</i>	One of class constants: ACTION_KEEP_A Output A. Copy B to A. Get the next B. ACTION_DELETE_A Copy B to A. Get the next B. (Delete A). ACTION_DELETE_A_B Get the next B. (Delete B).
-----	------------------	---

8.73.3.2 `get ()` [protected]

Get next char.

Convert ctrl char to space.

Returns

string|null

8.73.3.3 `isAlphaNum ($c)` [protected]

Is \$c a letter, digit, underscore, dollar sign, or non-ASCII character.

Returns

bool

8.73.3.4 `min ()` [protected]

Perform minification, return result.

[action\(\)](#) [isAlphaNum\(\)](#) [get\(\)](#) [peek\(\)](#)

Returns

string

8.73.3.5 `static minify ($js)` [static]

Minify [Javascript](#).

[__construct\(\)](#) [min\(\)](#)

Parameters

string	<i>\$js</i>	Javascript to be minified
--------	-------------	---

Returns

string

8.73.3.6 `next ()` [protected]

Get the next character, skipping over comments.

[peek\(\)](#) is used to see if a '/' is followed by a '/' or '*'.

[get\(\)](#) [peek\(\)](#)

Exceptions

JSMinException	On unterminated comment.
--------------------------------	--------------------------

Returns

string

8.73.3.7 peek () [protected]

Get next char.

If is ctrl character, translate to a space or newline.

[get\(\)](#)

Returns

string|null

8.73.4 Field Documentation**8.73.4.1 \$a = "** [protected]**8.73.4.2 \$b = "** [protected]**8.73.4.3 \$input = "** [protected]**8.73.4.4 \$inputIndex = 0** [protected]**8.73.4.5 \$inputLength = 0** [protected]**8.73.4.6 \$lookAhead = null** [protected]**8.73.4.7 \$output = "** [protected]**8.73.4.8 const ACTION_DELETE_A = 2****8.73.4.9 const ACTION_DELETE_A_B = 3****8.73.4.10 const ACTION_KEEP_A = 1****8.73.4.11 const ORD_LF = 10****8.73.4.12 const ORD_SPACE = 32**

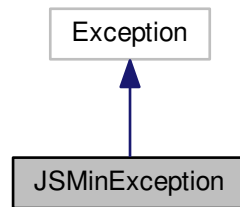
The documentation for this class was generated from the following file:

- lib/classes/[class.Compressor.php](#)

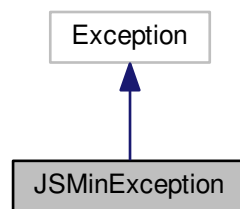
8.74 JSMinException Class Reference

[JSMinException](#).

Inheritance diagram for JSMinException:



Collaboration diagram for JSMinException:



8.74.1 Detailed Description

[JSMinException](#).

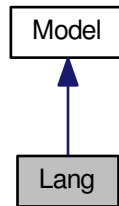
The documentation for this class was generated from the following file:

- [lib/classes/class.Compressor.php](#)

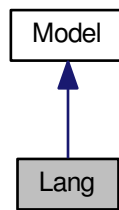
8.75 Lang Class Reference

Classe di tipo [Gino.Model](#) che rappresenta una lingua installata sul sistema.

Inheritance diagram for Lang:



Collaboration diagram for Lang:



Public Member Functions

- `__construct ($id=null)`
Costruttore.
- `__toString ()`
Rappresentazione a stringa dell'oggetto.
- `code ()`
Codice completo lingua es it_IT.

Static Public Member Functions

- static `columns ()`
Struttura dei campi della tabella di un modello.
- static `getMainLang ()`
Lingua di default.

Static Public Attributes

- static `$table` = "language"
- static `$columns`

Static Private Member Functions

- static [languageCodes](#) ()
Codici lingua.
- static [countryCodes](#) ()
Codici stato.

Additional Inherited Members

8.75.1 Detailed Description

Classe di tipo [Gino.Model](#) che rappresenta una lingua installata sul sistema.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.75.2 Constructor & Destructor Documentation

8.75.2.1 `__construct ($id = null)`

Costruttore.

Parameters

int	<i>\$id</i>	id lingua
-----	-------------	-----------

Returns

istanza di [Gino.App.Language.Lang](#)

8.75.3 Member Function Documentation

8.75.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

nome lingua

8.75.3.2 `code ()`

Codice completo lingua es it_IT.

Returns

codice lingua

8.75.3.3 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.75.3.4 `static countryCodes () [static],[private]`

Codici stato.

Returns

array di codici stato codice => stato

8.75.3.5 `static getMainLang () [static]`

Lingua di default.

La lingua di default viene impostata nelle Impostazioni di sistema [Gino](#)

Returns

string (codice della lingua)

8.75.3.6 `static languageCodes () [static],[private]`

Codici lingua.

Returns

array di codici lingua codice => lingua

8.75.4 Field Documentation

8.75.4.1 `$columns [static]`

8.75.4.2 `$table = "language" [static]`

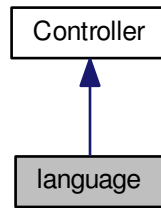
The documentation for this class was generated from the following file:

- [app/language/class.Lang.php](#)

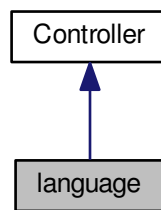
8.76 language Class Reference

Classe di tipo [Gino.Controller](#) per la gestione delle lingue disponibili per le traduzioni.

Inheritance diagram for language:



Collaboration diagram for language:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [choiceLanguage](#) ()
Vista scelta lingua (includibile in template) Se l'impostazione multilingua è FALSE ritorna una risposta di contenuto vuoto.
- [manageLanguage](#) (\Gino\Http\Request \$request)
Interfaccia di amministrazione modulo.

Static Public Member Functions

- static [outputFunctions](#) ()
Elenco dei metodi che possono essere richiamati dal menu e dal template.

Data Fields

- const [FLAG_PREFIX](#) = "flag_"
- const [FLAG_SUFFIX](#) = ".gif"
- [\\$_optionsLabels](#)

Private Member Functions

- [manageLang](#) ([\Gino\Http\Request](#) \$request)
Interfaccia di amministrazione lingue installate nel sistema.

Private Attributes

- [\\$_options](#)
- [\\$_title](#)
- [\\$_flag_language](#)

Additional Inherited Members

8.76.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione delle lingue disponibili per le traduzioni.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.76.2 Constructor & Destructor Documentation

8.76.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.App.Language.language](#)

8.76.3 Member Function Documentation

8.76.3.1 `choiceLanguage ()`

Vista scelta lingua (includibile in template) Se l'impostazione multilingua è FALSE ritorna una risposta di contenuto vuoto.

Returns

html

8.76.3.2 `manageLang (\Gino\Http\Request $request)` [private]

Interfaccia di amministrazione lingue installate nel sistema.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

html, interfaccia

8.76.3.3 `manageLanguage (\Gino\Http\Request $request)`

Interfaccia di amministrazione modulo.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Response](#)

8.76.3.4 `static outputFunctions () [static]`

Elenco dei metodi che possono essere richiamati dal menu e dal template.

Returns

array, elenco metodi nella forma nome_metodo => array(label => string, permissions => array())

8.76.4 Field Documentation

8.76.4.1 `$_flag_language` [private]

8.76.4.2 `$_options` [private]

8.76.4.3 `$_optionsLabels`

8.76.4.4 `$_title` [private]

8.76.4.5 `const FLAG_PREFIX = "flag_"`

8.76.4.6 `const FLAG_SUFFIX = ".gif"`

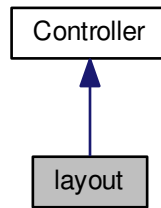
The documentation for this class was generated from the following file:

- [app/language/class_language.php](#)

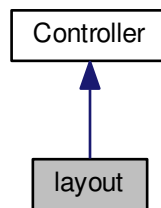
8.77 layout Class Reference

Gestisce il layout dell'applicazione raggruppando le funzionalità fornite dalle librerie dei css, template e skin.

Inheritance diagram for layout:



Collaboration diagram for layout:



Public Member Functions

- [__construct](#) ()
- [manageLayout](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa per la gestione del layout.
- [actionSkin](#) (\Gino\Http\Request \$request)
Processa il form di inserimento/modifica skin.
- [actionDelSkin](#) (\Gino\Http\Request \$request)
Processa il form di eliminazione skin.
- [actionCss](#) (\Gino\Http\Request \$request)
Processa il form di inserimento/modifica css.
- [actionDelCss](#) (\Gino\Http\Request \$request)
Processa il form di eliminazione css.
- [actionTemplate](#) (\Gino\Http\Request \$request)
Processa il form di inserimento/modifica template.
- [actionDelTemplate](#) (\Gino\Http\Request \$request)
Processa il form di eliminazione template.
- [modulesCodeList](#) ()
Elenco dei moduli di sistema, istanze di moduli di sistema e delle pagine disponibili come blocchi all'interno del template.

- [modulesList](#) (\Gino\Http\Request \$request)

Elenco dei moduli di sistema, istanze di moduli di sistema e delle pagine disponibili come blocchi all'interno del template.

- [actionFiles](#) (\Gino\Http\Request \$request)

Processa il form di modifica dei file.

Private Member Functions

- [manageTemplate](#) (\Gino\Http\Request \$request)

Interfaccia di amministrazione dei template.

- [manageSkin](#) (\Gino\Http\Request \$request)

Interfaccia di amministrazione delle skin.

- [manageCss](#) (\$request)

Interfaccia di amministrazione dei CSS.

- [manageView](#) (\$request)

Interfaccia di amministrazione delle viste.

- [skinList](#) ()

Lista delle skin.

- [templateList](#) ()

Lista dei template.

- [cssList](#) ()

Lista dei css.

- [viewList](#) ()

Lista delle viste.

- [info](#) ()

Informazioni modulo.

- [formFiles](#) (\$filename, \$code)

Form di modifica files (css, viste)

Additional Inherited Members

8.77.1 Detailed Description

Gestisce il layout dell'applicazione raggruppando le funzionalità fornite dalle librerie dei css, template e skin.

See also

[Gino.Css](#)
[Gino.Template](#)
[Gino.Skin](#)

Fornisce le interfacce per la modifica dei file di frontend generali di gino:

- file css presenti nella directory *css*
- file delle viste presenti nella directory *views*

PROCESSO DI GESTIONE DEL LAYOUT A BLOCCHI

Lo schema del layout viene stampato dal metodo `template::manageTemplate()` che legge il file di template e identifica porzioni di codice tipo:

```
<div id="nav_3_1_1" style="width:100%;">
{module sysclassid=8 func=printFooterPublic}
</div>
```

Queste porzioni di codice vengono passate con la funzione `preg_replace_callback()` al metodo [Gino.Template::renderNave\(\)](#) che recupera il tipo di blocco nello schema del template utilizzando delle funzioni di `preg_match()`. L'elenco dei moduli/pagine disponibili viene gestito dal metodo [Gino.App.Layout.layout::modulesList\(\)](#).

LAYOUT FREE

I layout free sono gestiti direttamente editando il file php del template. Anche in questo caso si usa un meta linguaggio per inserire output di moduli nelle posizioni desiderate.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.77.2 Constructor & Destructor Documentation

8.77.2.1 __construct ()

8.77.3 Member Function Documentation

8.77.3.1 actionCss (\Gino\Http\Request \$request)

Processa il form di inserimento/modifica css.

See also

[Gino.Css::formCssLayout\(\)](#)

Parameters

\Gino\← Http\← Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.2 actionDelCss (\Gino\Http\Request \$request)

Processa il form di eliminazione css.

See also

[Gino.Css::formDelCssLayout\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.3 actionDelSkin (\Gino\Http\Request \$request)

Processa il form di eliminazione skin.

See also

[Gino.Skin::formDelSkin\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.4 actionDelTemplate (\Gino\Http\Request \$request)

Processa il form di eliminazione template.

See also

[Gino.Template::formDelTemplate\(\)](#)
[Gino.Template::actionDelTemplate\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.5 actionFiles (\Gino\Http\Request \$request)

Processa il form di modifica dei file.

See also

[self::formFiles\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.6 actionSkin (\Gino\Http\Request \$request)

Processa il form di inserimento/modifica skin.

See also

[Gino.Skin::formSkin\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.7 actionTemplate (\Gino\Http\Request \$request)

Processa il form di inserimento/modifica template.

See also

[Gino.Template::formTemplate\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.77.3.8 cssList() [private]

Lista dei css.

Returns

html, lista css

8.77.3.9 formFiles (\$filename, \$code) [private]

[Form](#) di modifica files (css, viste)

Parameters

string	<i>\$filename</i>	
string	<i>\$code</i>	css view

8.77.3.10 info () [private]

Informazioni modulo.

Returns

html, informazioni

8.77.3.11 manageCss (\$request) [private]

Interfaccia di amministrazione dei CSS.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) o html

8.77.3.12 manageLayout (\Gino\Http\Request \$request)

Interfaccia amministrativa per la gestione del layout.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

8.77.3.13 manageSkin (\Gino\Http\Request \$request) [private]

Interfaccia di amministrazione delle skin.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) o html

8.77.3.14 manageTemplate (\Gino\Http\Request \$request) [private]

Interfaccia di amministrazione dei template.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) o html

8.77.3.15 `manageView ($request) [private]`

Interfaccia di amministrazione delle viste.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) o html

8.77.3.16 `modulesCodeList ()`

Elenco dei moduli di sistema, istanze di moduli di sistema e delle pagine disponibili come blocchi all'interno del template.

I metodi il cui output è disponibile all'inserimento nel template sono quelli definiti nel metodo `outputFunctions` della classe di tipo [Gino.Controller](#) e che non sono presenti nel file `<nome_modulo>.ini`, e quindi non richiamabili da url. Questo metodo è richiamato nell'interfaccia di gestione di template di tipo free

Returns

[Gino.Http.Response](#)

8.77.3.17 `modulesList (\Gino\Http\Request $request)`

Elenco dei moduli di sistema, istanze di moduli di sistema e delle pagine disponibili come blocchi all'interno del template.

I metodi il cui output è disponibile all'inserimento nel template sono quelli definiti nel metodo `outputFunctions` della classe di tipo [Gino.Controller](#) e che non sono presenti nel file `<nome_modulo>.ini`, e quindi non richiamabili da url. Questo metodo è richiamato nell'interfaccia di gestione di template di tipo a blocchi, e consente l'inserimento dell'output all'interno della struttura con un click.

Returns

[Gino.Http.Response](#)

8.77.3.18 `skinList () [private]`

Lista delle skin.

Returns

html, lista skin

8.77.3.19 `templateList()` [private]

Lista dei template.

Returns

html, lista template

8.77.3.20 `viewList()` [private]

Lista delle viste.

Returns

html, lista viste

The documentation for this class was generated from the following file:

- [app/layout/class_layout.php](#)

8.78 Ldap Class Reference

Libreria di connessione ai server [Ldap](#).

Public Member Functions

- [__construct](#) (\$username, \$password, \$options=array())
Costruttore.
- [authentication](#) ()
Autenticazione [Ldap](#).

Data Fields

- [\\$_ldap_protocol_version](#)
- [\\$_filter_search](#)
- [\\$_justthese_search](#)

Private Member Functions

- [checkValidUsername](#) ()
Verifica se il nome utente è in un formato valido.
- [printError](#) (\$string=null)
Stampa le informazioni sulle procedure di connessione.
- [connection](#) ()
Connessione [Ldap](#).
- [binding](#) ()
- [search](#) ()
Ricerca [Ldap](#).

Private Attributes

- [\\$_ldap_host](#)
- [\\$_ldap_port](#)
- [\\$_ldap_base_dn](#)
- [\\$_ldap_ds](#)
- [\\$_ldap_username](#)
- [\\$_ldap_password](#)
- [\\$_ldap_log](#)
- [\\$_ldap_domain](#)
- [\\$_user_name](#)
- [\\$_user_password](#)
- [\\$_debug](#)
- [\\$_auth](#)
- [\\$_test_anonymous](#)
- [\\$_test_search](#)

8.78.1 Detailed Description

Libreria di connessione ai server [Ldap](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.78.2 Constructor & Destructor Documentation

8.78.2.1 `__construct ($username, $password, $options = array())`

Costruttore.

Parameters

string	<i>\$username</i>	username di accesso
string	<i>\$password</i>	password dell'utente
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • debug (boolean)

Returns

void

8.78.3 Member Function Documentation

8.78.3.1 `authentication ()`

Autenticazione [Ldap](#).

See also

[checkValidUsername\(\)](#)
[printError\(\)](#)
[connection\(\)](#)
[binding\(\)](#)

Returns

boolean

8.78.3.2 `binding ()` [private]**8.78.3.3** `checkValidUsername ()` [private]

Verifica se il nome utente è in un formato valido.

Uno username inserito nella forma "dominio\username" ritorna un errore quando viene utilizzato come filtro di ricerca.

Returns

boolean

8.78.3.4 `connection ()` [private]

Connessione [Ldap](#).

Returns

void or string (log)

8.78.3.5 `printError ($string = null)` [private]

Stampa le informazioni sulle procedure di connessione.

Se il debug è attivo stampa tutti i log.

Parameters

string	<i>\$string</i>	se presente ritorna il log fino al momento della chiamata
--------	-----------------	---

8.78.3.6 `search ()` [private]

Ricerca [Ldap](#).

Returns

boolean (authentication) or string (log)

Esempio di output di `Ldap_get_entries`

```
Array
(
    [count] => 1
    [0] => Array
        (
            [cn] => Array
```

```

    (
        [count] => 1
        [0] => A006471
    )

    [0] => cn
    [count] => 1
    [dn] => CN=A006471,CN=Administration,CN=fgadam,DC=fg,DC=local
)
)

```

8.78.4 Field Documentation

8.78.4.1 `$_auth` [private]

8.78.4.2 `$_debug` [private]

8.78.4.3 `$_filter_search`

8.78.4.4 `$_justthese_search`

8.78.4.5 `$_ldap_base_dn` [private]

8.78.4.6 `$_ldap_domain` [private]

8.78.4.7 `$_ldap_ds` [private]

8.78.4.8 `$_ldap_host` [private]

8.78.4.9 `$_ldap_log` [private]

8.78.4.10 `$_ldap_password` [private]

8.78.4.11 `$_ldap_port` [private]

8.78.4.12 `$_ldap_protocol_version`

8.78.4.13 `$_ldap_username` [private]

8.78.4.14 `$_test_anonymous` [private]

8.78.4.15 `$_test_search` [private]

8.78.4.16 `$_user_name` [private]

8.78.4.17 `$_user_password` [private]

The documentation for this class was generated from the following file:

- [app/auth/class.Ldap.php](#)

8.79 Loader Class Reference

[Loader](#) di classi di tipo [Gino.Model](#) e classi di sistema.

Static Public Member Functions

- static [import](#) (\$app, \$classes)
Importa le classi di sistema o i modelli specificati Le classi di sistema sono quelle che risiedono nella directory lib/classes, le classi model sono proprie delle app.
- static [load](#) (\$class, \$args=array(), \$namespace='\Gino\\')
Restituisce un'istanza della classe di sistema richiesta.
- static [singleton](#) (\$class)
Restituisce l'istanza [Singleton](#) di una classe.

8.79.1 Detailed Description

[Loader](#) di classi di tipo [Gino.Model](#) e classi di sistema.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.79.2 Member Function Documentation

8.79.2.1 static import (\$app, \$classes) [static]

Importa le classi di sistema o i modelli specificati Le classi di sistema sono quelle che risiedono nella directory lib/classes, le classi model sono proprie delle app.

Per importare classi di sistema, l'argomento \$app deve avere la forma [class | class/subdir] dove subdir è la subdir di lib/classes in cui si trova la classe da importare, mentre le classi da importare, argomento \$classes, devono essere complete di namespace. Per importare classi di tipo [Gino.Model](#) invece l'argomento \$app deve essere il nome del [Controller](#) (ed anche della directory della app), mentre le classi nell'argomento \$classes devono essere specificate senza namespace.

Parameters

string	\$app	nome dell'app dei models oppure stringa 'class[/subdir]' per le classi di sistema
array string	\$classes	la classe o le classi da importare (comprehensive di eventuale namespace)

8.79.2.2 static load (\$class, \$args = array(), \$namespace = '\Gino\\') [static]

Restituisce un'istanza della classe di sistema richiesta.

Parameters

string	\$class	nome della classe di sistema da istanziare (senza namespace)
array	\$args	argomenti del costruttore
string	\$namespace	nome del namespace (default Gino)

Returns

object

8.79.2.3 `static singleton ($class) [static]`

Restituisce l'istanza [Singleton](#) di una classe.

Parameters

string	<i>\$class</i>	nome della classe comprensivo di eventuale namespace
--------	----------------	--

Returns

object

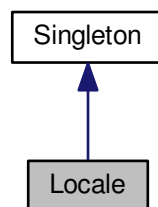
The documentation for this class was generated from the following file:

- lib/classes/[class.Loader.php](#)

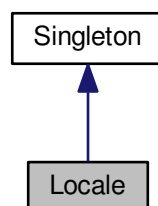
8.80 Locale Class Reference

Libreria per la gestione delle traduzioni che non utilizzano le librerie gettext.

Inheritance diagram for Locale:



Collaboration diagram for Locale:



Public Member Functions

- [get](#) (\$key)
Recupera la traduzione corrispondente alla chiave data.
- [manageLocale](#) (\$controller)
Interfaccia per la gestione dei file delle traduzioni dei moduli.

Static Public Member Functions

- static `init ()`
Setta la lingua del client Quando viene richiesto un cambio di lingua viene effettuato un refresh della pagina per poter reimpostare le librerie gettext.
- static `initGettext ()`
Inizializza le librerie gettext.

Protected Member Functions

- `__construct ($class_name)`
Costruttore.

Private Member Functions

- `pathToFile ()`
Percorso file traduzioni della classe o quello di default se non presente.
- `pathToBaseDir ($code=null)`
Percorso assoluto della directory dei file delle traduzioni.
- `fileName ()`
Nome del file delle traduzioni.
- `setProperties ($controller)`
Imposta alcune proprietà
- `listLocaleFile ()`
Lista dei file delle traduzioni.
- `moduleList ()`
Tabella con l'elenco dei file delle traduzioni del modulo.
- `formModuleFile ($request)`
Form di modifica file.
- `actionModuleFile ($request)`
Processa il form di modifica di un file.
- `formCreateFile ($request)`
Form di creazione di un file delle traduzioni.
- `actionCreateFile ($request)`
Processa il form di creazione di un file.

Static Private Member Functions

- static `setLanguage ()`
Setta la lingua di navigazione e di default in sessione.
- static `userLanguage ()`
Lingua dello User Agent.
- static `get_languages ()`
Recupera le caratteristiche di tutti i locale.
- static `detectCodes ()`
Elenco dei codici lingua associati alle nazioni.

Private Attributes

- [\\$_session](#)
- [\\$_strings](#)
- [\\$_class_name](#)
- [\\$_file_list](#)

Additional Inherited Members

8.80.1 Detailed Description

Libreria per la gestione delle traduzioni che non utilizzano le librerie gettext.

Meccanismi per la gestione delle traduzioni

In gino sono previsti due meccanismi per gestire le traduzioni:

- utilizzo delle librerie gettext
- file di stringhe localizzate

La classe locale si prende carico della gestione dei file di stringhe.

La classe locale viene inclusa nel file [class.Core.php](#) e viene istanziata come singleton:

- nella classe [Gino.Controller](#) per le classi applicative (che risiedono nella directory app)
- nelle classi modello delle classi applicative che estendono la classe [Model](#)
- nelle classi non applicative

Esempio di richiamo in una classe modello

```
$this->_locale = locale::instance_to_class($this->_controller->getClassName());
```

Esempio di richiamo in una classe non applicativa

```
$locale = locale::instance_to_class(get_class());
```

Directory dei file

La definizione delle directory dei file avviene nel metodo [pathToFile\(\)](#).

Mentre il nome del file è comunque sempre nella forma `[nome_classe]_lang.php`, come ad esempio

```
app/user/language/en_US/user_lang.php
```

la directory dove risiedono questi file non è univoca.

Se esiste la directory `app/[nome_classe]`, il file viene cercato nel percorso:

```
app/[nome_classe]/language/[codice_lingua]/
```

In caso contrario nel percorso:

```
languages/[codice_lingua]/
```

Richiamare le stringhe

Per richiamare una stringa si utilizza il metodo *get* passandogli il nome della chiave che identifica la stringa, ad esempio:

```
$this->_locale->get('label_phone')
```

I file contenenti le stringhe sono così costruiti:

```
// versione inglese
return array(
    'label_name' => 'Name',
    'label_comments' => 'Enabled comments'
);
// versione italiana
return array(
    'label_name' => 'Nome',
    'label_comments' => 'Abilita i commenti'
);
```

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.80.2 Constructor & Destructor Documentation

8.80.2.1 __construct (*\$class_name*) [protected]

Costruttore.

Parameters

string	<i>\$class_name</i>	nome della classe che istanzia
--------	---------------------	--------------------------------

Returns

void

8.80.3 Member Function Documentation

8.80.3.1 actionCreateFile (*\$request*) [private]

Processa il form di creazione di un file.

Parameters

--	--

8.80.3.2 actionModuleFile (*\$request*) [private]

Processa il form di modifica di un file.

Parameters

--	--

8.80.3.3 static detectCodes () [static],[private]

Elenco dei codici lingua associati alle nazioni.

Returns

array codice_stato=>nome_stato

8.80.3.4 fileName () [private]

Nome del file delle traduzioni.

Returns

nome file

8.80.3.5 formCreateFile (\$request) [private]

[Form](#) di creazione di un file delle traduzioni.

Parameters

--	--

8.80.3.6 formModuleFile (\$request) [private]

[Form](#) di modifica file.

Parameters

--	--

8.80.3.7 get (\$key)

Recupera la traduzione corrispondente alla chiave data.

Parameters

string	\$key	chiave
--------	-------	--------

Returns

traduzione o la chiave stessa se la traduzione non è presente

8.80.3.8 static get_languages () [static],[private]

Recupera le caratteristiche di tutti i locale.

Ritorna un array con i seguenti 4 elementi per ogni lingua che l'OS supporta:

1. abbreviazione completa, es. en-ca

2. codice lingua primario, es. en
3. stringa completa lingua, es. English (Canada)
4. stringa lingua primaria, es. English

Esempio

```
$_SERVER["HTTP_ACCEPT_LANGUAGE"] :  
en-gb,en;q=0.5 [Firefox]  
it-it,it;q=0.8,en-us;q=0.5,en;q=0.3 [Firefox]  
it [IE7]
```

See also

[detectCodes\(\)](#)

Returns

array

8.80.3.9 static init() [static]

Setta la lingua del client Quando viene richiesto un cambio di lingua viene effettuato un refresh della pagina per poter reimpostare le librerie gettext.

Returns

TRUE

8.80.3.10 static initGettext() [static]

Inizializza le librerie gettext.

See also

[lib/global.php](#)

Returns

TRUE

8.80.3.11 listLocaleFile() [private]

Lista dei file delle traduzioni.

See also

[self::pathToBaseDir\(\)](#)

Returns

void

8.80.3.12 `manageLocale ($controller)`

Interfaccia per la gestione dei file delle traduzioni dei moduli.

See also

```
self::setProperties()
self::moduleList()
self::formModuleFile()
self::actionModuleFile()
```

Parameters

object	<i>\$controller</i>	controller
--------	---------------------	------------

Returns

interfaccia

8.80.3.13 `moduleList ()` [private]

Tabella con l'elenco dei file delle traduzioni del modulo.

Utilizza la libreria javascript CodeMirror

Returns

codice html

8.80.3.14 `pathToBaseDir ($code = null)` [private]

Percorso assoluto della directory dei file delle traduzioni.

Parameters

string	<i>\$code</i>	codice della lingua
--------	---------------	---------------------

Returns

percorso

8.80.3.15 `pathToFile ()` [private]

Percorso file traduzioni della classe o quello di default se non presente.

See also

```
self::fileName()
self::pathToBaseDir()
```

Parameters

string	<i>\$class_name</i>	nome classe
--------	---------------------	-------------

Returns

percorso

8.80.3.16 static `setLanguage ()` [static],[private]

Setta la lingua di navigazione e di default in sessione.

See also

[Gino.App.Language.Lang::getMainLang\(\)](#)

Returns

void

8.80.3.17 `setProperties ($controller)` [private]

Imposta alcune proprietà

See also

`self::listLocaleFile()`

Parameters

object	<i>\$controller</i>	
--------	---------------------	--

Returns

void

8.80.3.18 static `userLanguage ()` [static],[private]

Lingua dello User Agent.

See also

[get_languages\(\)](#)

Returns

lingua user agent o FALSE se non trovata

8.80.4 Field Documentation

8.80.4.1 `$_class_name` [private]

8.80.4.2 `$_file_list` [private]

8.80.4.3 `$_session` [private]

8.80.4.4 `$_strings` [private]

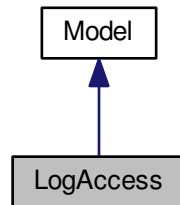
The documentation for this class was generated from the following file:

- lib/classes/[class.Locale.php](#)

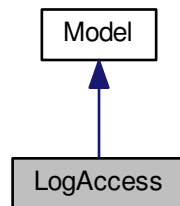
8.81 LogAccess Class Reference

Modello che rappresenta un login sul sistema.

Inheritance diagram for LogAccess:



Collaboration diagram for LogAccess:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.
- static [getCountForUser](#) (\$user_id)
Totale di accessi utente.
- static [getLastForUser](#) (\$user_id)
Ultimo accesso di un utente.

Static Public Attributes

- static [\\$table](#) = TBL_LOG_ACCESS
- static [\\$columns](#)

Additional Inherited Members

8.81.1 Detailed Description

Modello che rappresenta un login sul sistema.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.81.2 Constructor & Destructor Documentation

8.81.2.1 `__construct ($id)`

Costruttore.

Returns

istanza di [Gino.App.Statistics.LogAccess](#)

8.81.3 Member Function Documentation

8.81.3.1 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.81.3.2 `static getCountForUser ($user_id) [static]`

Totale di accessi utente.

Parameters

int	<i>\$user_id</i>	id utente
-----	------------------	-----------

Returns

numero di accessi

8.81.3.3 `static getLastForUser ($user_id) [static]`

Ultimo accesso di un utente.

Parameters

int	<code>\$user_id</code>	id utente
-----	------------------------	-----------

Returns

[Gino.App.Statistics.LogAccess](#) dell'ultimo accesso o null

8.81.4 Field Documentation

8.81.4.1 `$columns` [static]

8.81.4.2 `$table = TBL_LOG_ACCESS` [static]

The documentation for this class was generated from the following file:

- [app/statistics/class.LogAccess.php](#)

8.82 Logger Class Reference

Classe per la notifica di log di sistema.

Static Public Member Functions

- static [messageReportAdmins](#) (\$subject, \$message)
Invia un messaggio agli amministratori del sistema se la costante DEBUG in [configuration.php](#) è FALSE.
- static [exceptionReportAdmins](#) (\$exception)
Invia lo stack trace di una exception agli amministratori del sistema se la costante DEBUG in [configuration.php](#) è FALSE.
- static [manageException](#) (\$exception)
Gestore di eccezioni.

Static Private Member Functions

- static [systemVariablesHtml](#) ()
Html che mostra il valore di variabili di sistema \$_SESSION, \$_SERVER, \$_REQUEST.
- static [stackTraceHtml](#) (\$exception)
Html che mostra lo stack trace di una [Exception](#).

8.82.1 Detailed Description

Classe per la notifica di log di sistema.

Gestisce un logger per errori e warning. Se la costante DEBUG definita in [configuration.php](#) è settata a TRUE stampa a video errori e warnings. Se DEBUG è impostata a FALSE invia una mail agli amministratori di sistema definiti dalla costante ADMIN in [configuration.php](#). Gestisce il comportamento a seguito del throw di una exception. La risposta si differenzia a seconda del valore della costante DEBUG.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.82.2 Member Function Documentation

8.82.2.1 static exceptionReportAdmins (*\$exception*) [static]

Invia lo stack trace di una exception agli amministratori del sistema se la costante DEBUG in [configuration.php](#) è FALSE.

Parameters

Exception	<i>\$exception</i>	oggetto Exception
---------------------------	--------------------	-----------------------------------

Returns

TRUE se la mail è stata correttamente spedita, FALSE altrimenti

8.82.2.2 static manageException (*\$exception*) [static]

Gestore di eccezioni.

Con DEBUG attivo stampa a video il trace, in produzione invia una mail con il trace agli ADMINS definiti nel file [configuration.php](#) e ritorna una [Gino.Http.Response](#) definita dalla classe [Exception](#) oppure una [Gino.Http.<ResponseServerError](#)

Parameters

\Exception	<i>\$exception</i>	oggetto Exception return void
----------------------------	--------------------	---

8.82.2.3 static messageReportAdmins (*\$subject*, *\$message*) [static]

Invia un messaggio agli amministratori del sistema se la costante DEBUG in [configuration.php](#) è FALSE.

Parameters

string	<i>\$subject</i>	titolo messaggio da inviare
string	<i>\$message</i>	corpo messaggio da inviare

Returns

vero se la mail è stata correttamente spedita, falso altrimenti

8.82.2.4 static stackTraceHtml (*\$exception*) [static],[private]

Html che mostra lo stack trace di una [Exception](#).

Parameters

\Exception	<i>\$exception</i>	
----------------------------	--------------------	--

Returns

documento html con stack trace

8.82.2.5 static systemVariablesHtml () [static],[private]

Html che mostra il valore di variabili di sistema \$_SESSION, \$_SERVER, \$_REQUEST.

Returns

html

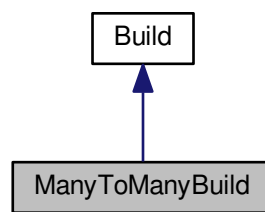
The documentation for this class was generated from the following file:

- lib/classes/[class.Logger.php](#)

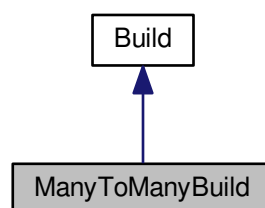
8.83 ManyToManyBuild Class Reference

Gestisce i campi di tipo many to many.

Inheritance diagram for ManyToManyBuild:



Collaboration diagram for ManyToManyBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()

Rappresentazione a stringa dell'oggetto.

- [getJoinTable](#) ()

Getter della proprietà join_table.

- [getJoinTableId](#) ()

Getter della proprietà join_table_id (nome della chiave esterna nella tabella di join del modello che ha il campo) proprietà join_table_id.

- [getJoinTableM2mId](#) ()

Getter della proprietà join_table_m2m_id (nome della chiave esterna nella tabella di join del modello m2m) proprietà join_table_m2m_id.

- [formElement](#) (\Gino\Form \$form, \$options)
- [clean](#) (\$options=null)
- [filterWhereClause](#) (\$value)
- [retrieveValue](#) ()

Data Fields

- [\\$_m2m_order](#)
- [\\$_m2m_where](#)
- [\\$_m2m_controller](#)
- [\\$_join_table](#)
- [\\$_add_related](#)
- [\\$_add_related_url](#)
- [\\$_join_table_m2m_id](#)

Protected Attributes

- [\\$_m2m](#)

Proprietà dei campi specifiche del tipo di campo.

- [\\$_join_table_id](#)

Private Member Functions

- [getJoinM2mValues](#) ()

Valori dei record della tabella m2m associati al modello.

8.83.1 Detailed Description

Gestisce i campi di tipo many to many.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.83.2 Constructor & Destructor Documentation

8.83.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.83.3 Member Function Documentation

8.83.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

rappresentazione a stringa dei modelli associati separati da virgola

8.83.3.2 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#)

Returns

array

8.83.3.3 `filterWhereClause ($value)`

See also

[Gino.Build::filterWhereClause\(\)](#)

Parameters

array	<i>\$value</i>	
-------	----------------	--

8.83.3.4 `formElement (\Gino\Form $form, $options)`

See also

[Gino.Build::formElement\(\)](#)

8.83.3.5 `getJoinM2mValues ()` [private]

Valori dei record della tabella m2m associati al modello.

Returns

array (id values)

8.83.3.6 `getJoinTable ()`

Getter della proprietà `join_table`.

Returns

proprietà `join_table`

8.83.3.7 `getJoinTableId ()`

Getter della proprietà `join_table_id` (nome della chiave esterna nella tabella di join del modello che ha il campo) proprietà `join_table_id`.

8.83.3.8 `getJoinTableM2mId ()`

Getter della proprietà `join_table_m2m_id` (nome della chiave esterna nella tabella di join del modello m2m) proprietà `join_table_m2m_id`.

8.83.3.9 `retrieveValue ()`

See also

[Gino.Build::retrieveValue\(\)](#)

8.83.4 Field Documentation

8.83.4.1 `$_add_related`

8.83.4.2 `$_add_related_url`

8.83.4.3 `$_join_table`

8.83.4.4 `$_join_table_id` [protected]

8.83.4.5 `$_join_table_m2m_id`

8.83.4.6 `$_m2m` [protected]

Proprietà dei campi specifiche del tipo di campo.

8.83.4.7 `$_m2m_controller`

8.83.4.8 `$_m2m_order`

8.83.4.9 \$_m2m_where

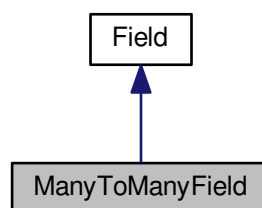
The documentation for this class was generated from the following file:

- lib/classes/build/[class.ManyToManyBuild.php](#)

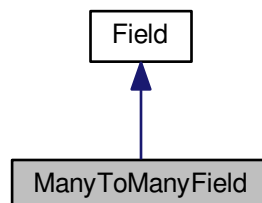
8.84 ManyToManyField Class Reference

Campo di tipo many to many.

Inheritance diagram for ManyToManyField:



Collaboration diagram for ManyToManyField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueToDb](#) (\$value)

Data Fields

- [\\$_m2m_order](#)
- [\\$_m2m_where](#)
- [\\$_add_related_url](#)

Protected Attributes

- [\\$_m2m](#)
Proprietà dei campi specifiche del tipo di campo.
- [\\$_join_table](#)
- [\\$_add_related](#)

8.84.1 Detailed Description

Campo di tipo many to many.

I valori da associare al campo risiedono in una tabella esterna e i parametri per accedervi devono essere definiti nelle opzioni del campo.

Tipologie di input associabili: multichек

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.84.2 Constructor & Destructor Documentation

8.84.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	<p>array associativo di opzioni del campo del database</p> <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – add_related (bool): includere o meno un bottone che permetta l'inserimento di nuovi modelli correlati nello stesso contesto – add_related_url (string): path alla vista per inserimento modello correlato – m2m (object): classe del modello correlato (nome completo di namespace) – m2m_where (mixed): condizioni della query per filtrare i possibili modelli da associare <ul style="list-style-type: none"> * <i>string</i>, es. "cond1='\$cond1' AND cond2='\$cond2'" * <i>array</i>, es. array("cond1='\$cond1'", "cond2='\$cond2'") – m2m_order (string): ordinamento dei valori (es. name ASC) – m2m_controller (): classe Controller del modello m2m, essenziale se il modello appartiene ad un modulo istanziabile – join_table (string): nome tabella di join
-------	------------------	---

8.84.3 Member Function Documentation

8.84.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.84.3.2 `valueToDb ($value)`

See also

[Gino.Field::valueToDb\(\)](#)

8.84.4 Field Documentation

8.84.4.1 `$_add_related` `[protected]`

8.84.4.2 `$_add_related_url`

8.84.4.3 `$_join_table` `[protected]`

8.84.4.4 `$_m2m` `[protected]`

Proprietà dei campi specifiche del tipo di campo.

8.84.4.5 `$_m2m_order`

8.84.4.6 `$_m2m_where`

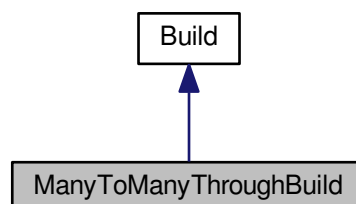
The documentation for this class was generated from the following file:

- [lib/classes/fields/class.ManyToManyField.php](#)

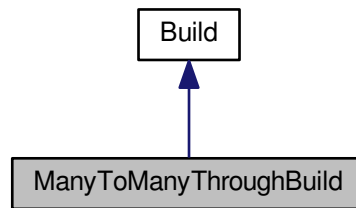
8.85 ManyToManyThroughBuild Class Reference

Gestisce i campi di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.

Inheritance diagram for ManyToManyThroughBuild:



Collaboration diagram for ManyToManyThroughBuild:



Public Member Functions

- `__construct` (\$options)
Costruttore.
- `__toString` ()
Rappresentazione a stringa dell'oggetto.
- `getValues` ()
Valori dei record di associazione.
- `getM2m` ()
Restituisce la classe del m2m.
- `getController` ()
Restituisce il controller del modello cui appartiene il campo.
- `getM2mTable` ()
Restituisce la tabella dati della classe m2m.
- `getModelTableId` ()
Restituisce il nome del campo che immagazzina l'id del modello che ha la relazione m2m.
- `formElement` (\Gino\Form \$form, \$options)
- `filterWhereClause` (\$value)

Data Fields

- `$_m2m_controller`
- `$_controller`

Protected Attributes

- `$_m2m`
Proprietà dei campi specifiche del tipo di campo.
- `$_remove_fields`
- `$_model_table_id`

8.85.1 Detailed Description

Gestisce i campi di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.85.2 Constructor & Destructor Documentation

8.85.2.1 __construct (*\$options*)

Costruttore.

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database
		<ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()• opzioni definite come proprietà specifiche del modello <p>– remove_fields (array)</p>

Returns

void

Ridefinisce il valore della proprietà `$_value` come array dei valori id dei record associati.

8.85.3 Member Function Documentation

8.85.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

lista rappresentazioni a stringa dei modelli correlati separati da virgola

8.85.3.2 filterWhereClause (*\$value*)

See also

[Gino.Build::filterWhereClause\(\)](#)

8.85.3.3 formElement (\Gino\Form *\$form*, *\$options*)

See also

[Gino.Build::formElement\(\)](#)

8.85.3.4 `getController ()`

Restituisce il controller del modello cui appartiene il campo.

Returns

[Gino.Controller](#) controller

8.85.3.5 `getM2m ()`

Restituisce la classe del m2m.

See also

[Gino.Model::m2mtObject\(\)](#)

Returns

string, nome della classe del modello m2m

8.85.3.6 `getM2mTable ()`

Restituisce la tabella dati della classe m2m.

Returns

nome tabella

8.85.3.7 `getModelTableId ()`

Restituisce il nome del campo che immagazzina l'id del modello che ha la relazione m2m.

Returns

nome del campo

8.85.3.8 `getValues ()`

Valori dei record di associazione.

Returns

array

8.85.4 Field Documentation

8.85.4.1 `$_controller`

8.85.4.2 `$_m2m` `[protected]`

Proprietà dei campi specifiche del tipo di campo.

8.85.4.3 `$_m2m_controller`

8.85.4.4 `$_model_table_id` `[protected]`

8.85.4.5 `$_remove_fields` `[protected]`

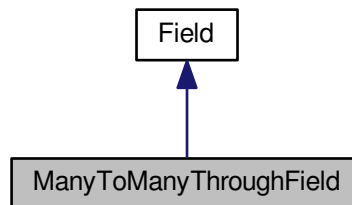
The documentation for this class was generated from the following file:

- `lib/classes/build/class.ManyToManyThroughBuild.php`

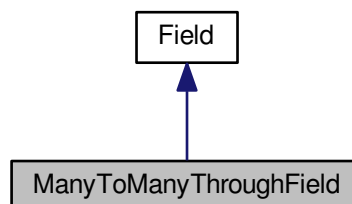
8.86 ManyToManyThroughField Class Reference

Campo di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.

Inheritance diagram for ManyToManyThroughField:



Collaboration diagram for ManyToManyThroughField:



Public Member Functions

- `__construct` (`$options`)
Costruttore.
- `getProperties` ()
- `valueFromDb` (`$value`)

Data Fields

- [\\$_m2m_controller](#)
- [\\$_controller](#)

Protected Attributes

- [\\$_m2m](#)

Proprietà dei campi specifiche del tipo di campo.

8.86.1 Detailed Description

Campo di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

Il prefisso dei nomi degli input che compongono il [ManyToManyThroughField](#) viene impostato dal javascript (m2mt↔_).

8.86.2 Constructor & Destructor Documentation

8.86.2.1 __construct (\$options)

Costruttore.

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – controller (object): controller del modello cui appartiene il campo – m2m (string): classe attraverso la quale si esprime la relazione molti a molti (nome completo di namespace) – m2m_controller (object): oggetto controller da passare eventualmente al costruttore della classe m2m
-------	------------------	--

Returns

void

8.86.3 Member Function Documentation

8.86.3.1 getProperties ()

See also

[Gino.Field::getProperties\(\)](#)

8.86.3.2 valueFromDb (*\$value*)

See also

[Gino.Field::valueFromDb\(\)](#)

Parameters

integer	<i>\$value</i>	valore id del record
---------	----------------	----------------------

Returns

null or array (valori id dei record di associazione)

8.86.4 Field Documentation

8.86.4.1 \$_controller

8.86.4.2 \$_m2m [protected]

Proprietà dei campi specifiche del tipo di campo.

8.86.4.3 \$_m2m_controller

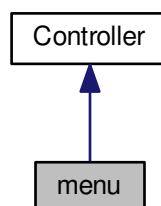
The documentation for this class was generated from the following file:

- lib/classes/fields/[class.ManyToManyThroughField.php](#)

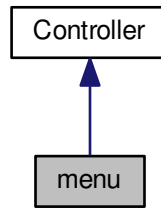
8.87 menu Class Reference

Classe di tipo [Gino.Controller](#) per la gestione dei menu.

Inheritance diagram for menu:



Collaboration diagram for menu:



Public Member Functions

- [__construct](#) (\$instance)
Costruttore.
- [deleteInstance](#) ()
Eliminazione di una istanza.
- [render](#) ()
Visualizzazione menu.
- [breadCrumbs](#) ()
Briciole di pane.
- [manageDoc](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa per la gestione del menu.
- [actionUpdateOrder](#) (\Gino\Http\Request \$request)
Aggiorna l'ordinamento delle voci di menu.
- [actionMenuVoice](#) (\Gino\Http\Request \$request)
Processa il form di inserimento/modifica voce di menu.
- [actionDelMenuVoice](#) (\$request)
Eliminazione di una voce di menu.
- [searchModules](#) ()
Form di ricerca moduli e pagine collegabili a voci di menu.
- [printItemsList](#) (\Gino\Http\Request \$request)
Mostra le interfacce che le classi mettono a disposizione del menu e le pagine.

Static Public Member Functions

- static [getClassElements](#) ()
Restituisce alcune proprietà della classe.
- static [outputFunctions](#) ()
Definizione dei metodi pubblici che forniscono un output per il front-end.

Data Fields

- [\\$_optionsLabels](#)

Private Member Functions

- [getTree](#) (\$parent=0)
Costruisce il tree delle voci di menu.
- [pathToSelectedVoice](#) ()
Percorso alla voce di menu selezionata.
- [listMenu](#) ()
Lista voci di menu aria amministrativa.
- [renderMenuAdmin](#) (\$parent=0)
Voci di menu con gli strumenti per la loro modifica, area amministrativa.
- [formMenuVoice](#) (\$voice, \$parent)
Form inserimento/modifica voce di menu.
- [jsSortLib](#) ()
Libreria javascript per l'ordinamento delle voci di menu.
- [jsSearchModulesLib](#) ()
Libreria javascript per la ricerca dei moduli.
- [printItemsPage](#) (\$pages)
Elenco pagine che è possibile collegare a una voce di menu.
- [printItemsClass](#) (\$modules_app, \$modules)
Interfacce che le classi dei moduli mettono a disposizione del menu.

Private Attributes

- [\\$_tbl_opt](#)
- [\\$_options](#)
- [\\$_title](#)
- [\\$_cache](#)
- [\\$_ico_more](#)

Static Private Attributes

- static [\\$_menu_functions_list](#) = 'menuFunctionsList'

Additional Inherited Members

8.87.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione dei menu.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.87.2 Constructor & Destructor Documentation

8.87.2.1 [__construct](#) (*\$instance*)

Costruttore.

Parameters

int	<i>\$instance</i>	id istanza
-----	-------------------	------------

Returns

istanza di [Gino.App.Menu.menu](#)

8.87.3 Member Function Documentation

8.87.3.1 actionDelMenuVoice (*\$request*)

Eliminazione di una voce di menu.

Parameters

\Gino\← Http\← Request	<i>\$request</i>	istanza di Gino.Http.Request
--	------------------	--

Returns

8.87.3.2 actionMenuVoice ([\Gino\Http\Request](#) *\$request*)

Processa il form di inserimento/modifica voce di menu.

Parameters

\Gino\← Http\← Request	<i>\$request</i>	istanza di Gino.Http.Request
--	------------------	--

Returns

8.87.3.3 actionUpdateOrder ([\Gino\Http\Request](#) *\$request*)

Aggiorna l'ordinamento delle voci di menu.

Parameters

\Gino\← Http\← Request	<i>\$request</i>	istanza di Gino.Http.Request
--	------------------	--

Returns

, risultato ordinamento

8.87.3.4 breadCrumbs ()

Briciole di pane.

Returns

html, briciole di pane

8.87.3.5 deleteInstance ()

Eliminazione di una istanza.

Returns

risultato operazione, bool

8.87.3.6 formMenuVoice (\$voice, \$parent) [private]

Form inserimento/modifica voce di menu.

Parameters

--	--

8.87.3.7 static getClassElements () [static]

Restituisce alcune proprietà della classe.

Returns

array associativo contenente le tabelle, viste e struttura directory contenuti

8.87.3.8 getTree (\$parent = 0) [private]

Costruisce il tree delle voci di menu.

Parameters

int	<i>\$parent</i>	id della voce parent
-----	-----------------	----------------------

Returns

array con il tree, chiavi: id, type, url, label, sub (altro tree, ricorsivo)

8.87.3.9 jsSearchModulesLib () [private]

Libreria javascript per la ricerca dei moduli.

Chiamate Ajax:

- [printItemsList\(\)](#)

Returns

html, codice js

8.87.3.10 jsSortLib () [private]

Libreria javascript per l'ordinamento delle voci di menu.

Chiamate Ajax:

- [actionUpdateOrder\(\)](#)

See also

[actionUpdateOrder\(\)](#)

Returns

html, codice js

8.87.3.11 listMenu () [private]

Lista voci di menu aria amministrativa.

Returns

html

8.87.3.12 manageDoc (\Gino\Http\Request \$request)

Interfaccia amministrativa per la gestione del menu.

Parameters

\Gino\← Http\← Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

8.87.3.13 static outputFunctions () [static]

Definizione dei metodi pubblici che forniscono un output per il front-end.

Questo metodo viene letto dal motore di generazione dei layout (prende i metodi non presenti nel file ini) e dal motore di generazione di voci di menu (presenti nel file ini) per presentare una lista di output associati all'istanza di classe.

Returns

array associativo metodi pubblici metodo => array('label' => label, 'permissions' => permissions)

8.87.3.14 pathToSelectedVoice () [private]

Percorso alla voce di menu selezionata.

See also

`self::breadCrumbs()`

Returns

html

8.87.3.15 `printItemsClass ($modules_app, $modules)` [private]

Interfacce che le classi dei moduli mettono a disposizione del menu.

Si richiamano i metodi [outputFunctions\(\)](#) delle classi dei moduli e dei moduli di sistema

See also

[Gino.App.Auth.Permission::getFromFullCode\(\)](#)

Parameters

array	<i>\$array_search</i>	array di array con le chiavi id, name, label, role1
-------	-----------------------	---

Returns

html

8.87.3.16 `printItemsList (\Gino\Http\Request $request)`

Mostra le interfacce che le classi mettono a disposizione del menu e le pagine.

See also

`self::printItemsClass()`

`self::printItemsPage()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns**8.87.3.17** `printItemsPage ($pages)` [private]

Elenco pagine che è possibile collegare a una voce di menu.

See also

[Gino.App.Auth.Permission::getFromFullCode\(\)](#)

Parameters

array	<i>\$array_search</i>	la chiave è il valore ID e il valore il titolo della pagina
-------	-----------------------	---

Returns

html

8.87.3.18 `render ()`

Visualizzazione menu.

See also

[Gino.App.Menu.MenuVoice::getSelectedVoice\(\)](#)

Returns

html, menu

8.87.3.19 `renderMenuAdmin ($parent = 0)` `[private]`

Voci di menu con gli strumenti per la loro modifica, area amministrativa.

See also

[jsSortLib\(\)](#)

Parameters

integer	<i>\$parent</i>	valore ID della voce di menu alla quale la voce corrente è collegata
---------	-----------------	--

Returns

html

8.87.3.20 `searchModules ()`

[Form](#) di ricerca moduli e pagine collegabili a voci di menu.

Returns

8.87.4 Field Documentation

8.87.4.1 `$_cache` `[private]`

8.87.4.2 `$_ico_more` `[private]`

8.87.4.3 `$_menu_functions_list = 'menuFunctionsList'` `[static], [private]`

8.87.4.4 `$_options` `[private]`

8.87.4.5 `$_optionsLabels`

8.87.4.6 `$_tbl_opt` `[private]`

8.87.4.7 `$_title` `[private]`

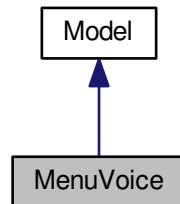
The documentation for this class was generated from the following file:

- [app/menu/class_menu.php](#)

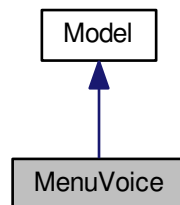
8.88 MenuVoice Class Reference

Classe di tipo [Gino.Model](#) che rappresenta una voce di menu.

Inheritance diagram for MenuVoice:



Collaboration diagram for MenuVoice:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [initOrderList](#) ()
Recupera la prima posizione disponibile per l'ordinamento.
- [updateOrderList](#) ()
Query di aggiornamento della sequenza delle voci di menu.
- [deleteVoice](#) ()
Elimina una voce di menu e tutte le sue discendenti (ricorsivo)
- [userCanSee](#) ()
Controlla se l'utente può vedere la voce di menu.
- [formVoice](#) (\$formation, \$parent)
Form di inserimento e modifica di una voce di menu.

Static Public Member Functions

- static [deleteInstanceVoices](#) (\$instance)
Elimina le voci di menu di una istanza (nella procedura di eliminazione di una istanza)
- static [get](#) (\$options=null)
Recupera oggetti di tipo [Gino.App.Menu.MenuVoice](#).
- static [getSelectedVoice](#) (\$instance)
Ricava la voce di menu corrispondente all'url corrente.
- static [columns](#) ()
Struttura dei campi della tabella di un modello.

Static Public Attributes

- static [\\$tbl_voices](#) = "sys_menu_voices"
- static [\\$columns](#)

Private Member Functions

- [getChildren](#) ()
Figli di primo livello della voce di menu.

Private Attributes

- [\\$_gform](#)

Static Private Attributes

- static [\\$_tbl_user_role](#) = "user_role"

Additional Inherited Members

8.88.1 Detailed Description

Classe di tipo [Gino.Model](#) che rappresenta una voce di menu.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.88.2 Constructor & Destructor Documentation

8.88.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID della voce di menu
---------	-------------	------------------------------

Returns

istanza di [Gino.App.Menu.MenuVoice](#)

8.88.3 Member Function Documentation

8.88.3.1 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.88.3.2 static deleteInstanceVoices (\$instance) [static]

Elimina le voci di menu di una istanza (nella procedura di eliminazione di una istanza)

Parameters

integer	<i>\$instance</i>	valore id dell'istanza
---------	-------------------	------------------------

Returns

TRUE

8.88.3.3 deleteVoice ()

Elimina una voce di menu e tutte le sue discendenti (ricorsivo)

Returns

void

8.88.3.4 formVoice (\$formation, \$parent)

[Form](#) di inserimento e modifica di una voce di menu.

Parameters

string	<i>\$formation</i>	indirizzo dell'azione
integer	<i>\$parent</i>	valore ID della voce parent

Returns

html, form

8.88.3.5 static get (\$options = null) [static]

Recupera oggetti di tipo [Gino.App.Menu.MenuVoice](#).

Parameters

array	<i>\$options</i>	array associativo di opzioni: <ul style="list-style-type: none">• where: where clause• order: ordinamento
-------	------------------	--

8.88.3.6 `getChildren ()` [*private*]

Figli di primo livello della voce di menu.

Returns

array id => [Gino.App.Menu.MenuVoice](#)

8.88.3.7 `static getSelectedVoice ($instance)` [*static*]

Ricava la voce di menu corrispondente all'url corrente.

Parameters

integer	<i>\$instance</i>	valore ID dell'istanza
---------	-------------------	------------------------

Returns

id voce selezionata o null

8.88.3.8 `initOrderList ()`

Recupera la prima posizione disponibile per l'ordinamento.

Returns

numero posizione ordinamento

8.88.3.9 `updateOrderList ()`

Query di aggiornamento della sequenza delle voci di menu.

Returns

risultato query, bool

8.88.3.10 `userCanSee ()`

Controlla se l'utente può vedere la voce di menu.

Returns

TRUE se la può vedere, FALSE altrimenti

8.88.4 Field Documentation

8.88.4.1 `$_gform` `[private]`

8.88.4.2 `$_tbl_user_role = "user_role"` `[static]`, `[private]`

8.88.4.3 `$columns` `[static]`

8.88.4.4 `$tbl_voices = "sys_menu_voices"` `[static]`

The documentation for this class was generated from the following file:

- [app/menu/class.MenuVoice.php](#)

8.89 MobileDetect Class Reference

Verifica se una richiesta HTTP è stata effettuata da un dispositivo mobile.

Public Member Functions

- [__construct](#) ()
Verifica se la richiesta proviene da un dispositivo mobile e sovrascrive la proprietà isMobile.
- [__call](#) (\$name, \$arguments)
Overloads isAndroid() | isBlackberry() | isOpera() | isPalm() | isWindows() | isGeneric() through isDevice()
- [isMobile](#) ()
Verifica se il dispositivo connesso è mobile.

Protected Member Functions

- [isDevice](#) (\$device)
Verifica se il dispositivo è un particolare device.

Protected Attributes

- [\\$accept](#)
- [\\$userAgent](#)
- [\\$isMobile](#) = FALSE
- [\\$isAndroid](#) = null
- [\\$isBlackberry](#) = null
- [\\$isIphone](#) = null
- [\\$isOpera](#) = null
- [\\$isPalm](#) = null
- [\\$isWindows](#) = null
- [\\$isGeneric](#) = null
- [\\$devices](#)

8.89.1 Detailed Description

Verifica se una richiesta HTTP è stata effettuata da un dispositivo mobile.

<http://www.opensource.org/licenses/mit-license.php> The MIT License

Version

SVN:

Id

Mobile_Detect.php 4 2011-05-26 08:04:20Z vic.stanciu@gmail.com

8.89.2 Constructor & Destructor Documentation

8.89.2.1 __construct ()

Verifica se la richiesta proviene da un dispositivo mobile e sovrascrive la proprietà *isMobile*.

Returns

void

8.89.3 Member Function Documentation

8.89.3.1 __call (\$name, \$arguments)

Overloads isAndroid() | isBlackberry() | isOpera() | isPalm() | isWindows() | isGeneric() through [isDevice\(\)](#)

Parameters

string	<i>\$name</i>	
array	<i>\$arguments</i>	

Returns

bool

8.89.3.2 isDevice (\$device) [protected]

Verifica se il dispositivo è un particolare device.

Parameters

string	<i>\$device</i>	nome device (android, opera, blackberry, ...)
--------	-----------------	---

Returns

risultato verifica, bool

8.89.3.3 isMobile ()

Verifica se il dispositivo connesso è mobile.

Returns

TRUE se il dispositivo è mobile, FALSE altrimenti

8.89.4 Field Documentation

8.89.4.1 **\$accept** [protected]

8.89.4.2 **\$devices** [protected]

Initial value:

```
= array(
    "android"      => "android",
    "blackberry"   => "blackberry",
    "iphone"       => "(iphone|ipod)",
    "opera"        => "opera mini",
    "palm"         => "(avantgo|blazer|elaine|hiptop|palm|plucker|xiino)",
    "windows"      => "windows ce; (iemobile|ppc|smartphone)",
    "generic"      => "
(kindle|mobile|mmp|midp|o2|pda|pocket|psp|symbian|smartphone|treo|up.browser|up.link|vodafone|wap)"
)
```

8.89.4.3 **\$isAndroid = null** [protected]

8.89.4.4 **\$isBlackberry = null** [protected]

8.89.4.5 **\$isGeneric = null** [protected]

8.89.4.6 **\$isIphone = null** [protected]

8.89.4.7 **\$isMobile = FALSE** [protected]

8.89.4.8 **\$isOpera = null** [protected]

8.89.4.9 **\$isPalm = null** [protected]

8.89.4.10 **\$isWindows = null** [protected]

8.89.4.11 **\$userAgent** [protected]

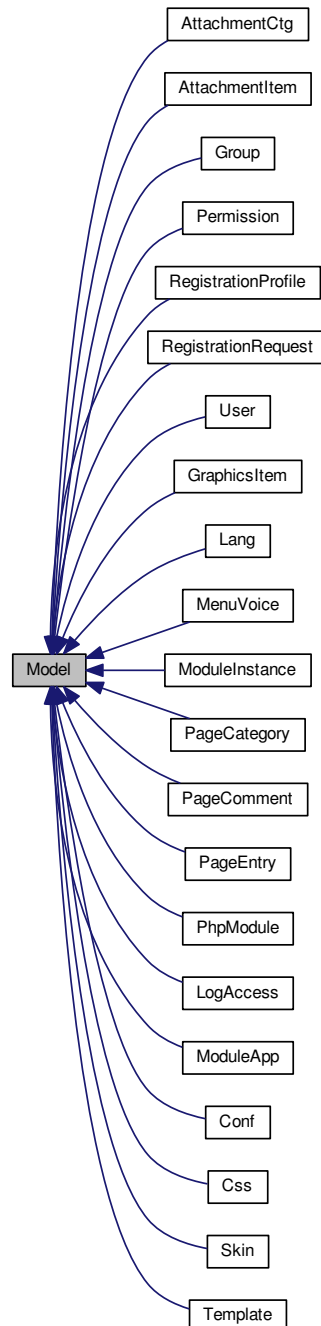
The documentation for this class was generated from the following file:

- lib/classes/[class.MobileDetect.php](#)

8.90 Model Class Reference

Classe astratta che definisce un modello, cioè un oggetto che rappresenta una tabella su database.

Inheritance diagram for Model:



Public Member Functions

- `__construct ($id=null)`
Costruttore.
- `__toString ()`

Rappresentazione a stringa dell'oggetto Sovrascrivere questo metodo nella classe figlia per restituire un valore parlante.

- **fieldLabel** (\$field)
Etichetta del campo.
- **setIsConstraint** (\$is_constraint)
Setter per la proprietà che contiene le informazioni per il check dei constraint Esempio:
- **__get** (\$pName)
Metodo richiamato ogni volta che qualcuno prova a ottenere una proprietà dell'oggetto non definita.
- **__set** (\$pName, \$pValue)
Metodo richiamato ogni volta che qualcuno prova a impostare una proprietà dell'oggetto non definita (\$this->{fieldname})
- **m2mtObject** (\$m2mt_field, \$id)
Ritorna l'oggetto ManyToMany through model.
- **ml** (\$pName)
Recupera le proprietà con la traduzione.
- **getStructure** ()
*Struttura dei campi del modello Un array associativo che contiene tutti i campi come chiavi e le relative classi di tipo **Field** come valore.*
- **getRecordValues** ()
Valori di un record.
- **getController** ()
***Controller** del modello.*
- **save** (\$options=array())
Salva il modello su db Salva sia i campi della tabella sia i m2m.
- **savem2m** (\$m2m)
Salvataggio dei ManyToMany.
- **deleteDbData** ()
Elimina le proprietà su db del modello e le traduzioni.
- **delete** ()
Elimina l'oggetto Elimina i dati su db, le traduzioni, e le associazioni m2m e m2mt Controlla che non ci siano regole di constraint che impediscano l'eliminazione, in caso ce ne fossero di non rispettate ritorna un elenco di regole che impediscono l'eliminazione.
- **deletem2m** ()
Elimina le associazioni m2m.
- **deletem2mthrough** ()
Elimina le associazioni m2mt.
- **deletem2mthroughField** (\$field_name)
Elimina le associazioni di un campo m2mt.
- **getModelLabel** ()
Etichetta del modello.
- **getTable** ()
Tabella principale dei dati.
- **getProperties** (\$field_obj)
Imposta le opzioni "model, field_object, value, table" e recupera le proprietà del campo dipendenti dai valori del record.
- **build** (\$field_obj)
*Classe **Build** del campo di tabella.*
- **shows** (\$field_obj)
Valore da mostrare in output.
- **fetchColumns** (\$id)
Recupera i valori del record e li carica nella proprietà _p Il valore dei campi di tipo ManyToMany è un array che racchiude i valori id dei record della tabella di join associata al modello.
- **refreshModel** ()
Refresh del modello (da chiamare manualmente)

Static Public Member Functions

- static [deleteInstance](#) (\$controller)
Eliminazione di tutti i record legati all'istanza del controller passato come argomento.
- static [getSelectOptionsFromObjects](#) (\$objects)
Array associativo id => rappresentazione a stringa a partire da array di oggetti.
- static [objects](#) (\$controller=null, \$options=array())
Metodo generico statico per ricavare oggetti.
- static [getFromSlug](#) (\$slug, \$controller=null)
Recupera l'oggetto a partire dallo slug.
- static [columns](#) ()
Recupera l'oggetto del tipo di campo di un modello.

Data Fields

- [\\$_request](#)
- [\\$_db](#)
- [\\$_lng_nav](#)

Protected Member Functions

- [setCheckIsConstraint](#) (\$check)
Setter per la variabile di controllo del check constraint.
- [isConstraintError](#) (\$res)
Errore conseguente ad una violazione delle constraint in eliminazione.
- [checkIsConstraint](#) ()
Controllo delle regole constraint.

Static Protected Member Functions

- static [properties](#) (\$model, \$controller=null)
Proprietà specifiche di un modello dipendenti dai valori del record (ad esempio dal valore id)

Protected Attributes

- [\\$_registry](#)
- [\\$_tbl_data](#)
- [\\$_model_label](#)
- [\\$_controller](#)
- [\\$_locale](#)
- [\\$_p](#) = array()
- [\\$_is_constraint](#) = array()
- [\\$_check_is_constraint](#) = true
- [\\$_lng_dft](#)

Private Member Functions

- [checkM2m](#) (\$field_obj)
Verifica se il tipo di campo di un modello è un oggetto ManyToMany.

Private Attributes

- [\\$_structure](#)
- [\\$_trd](#)

8.90.1 Detailed Description

Classe astratta che definisce un modello, cioè un oggetto che rappresenta una tabella su database.

La classe permette di descrivere la struttura dei dati del modello. Sono supportati molti tipi di dati (

See also

[lib/classes/fields/*](#)), compresi le relazioni molti [a](#) molti, comprensivi, nel caso, di campi aggiuntivi. La classe gestisce il salvataggio del modello su db [e](#) l'eliminazione, controllando, se specificato, che le constraint siano rispettate. Sono presenti metodi di generico utilizzo quali un selettore di oggetti, un selettore attraverso slug.

Le proprietà su DB possono essere lette attraverso il metodo `__get`, ma possono anche essere protette costruendo una funzione `get` personalizzata all'interno della classe.

Le proprietà su DB possono essere impostate attraverso il metodo `__set`; in aggiunta possono essere definiti setter specifici definendo dei metodi `setFieldName`.

La classe figlia che istanzia il parent passa il valore ID del record dell'oggetto direttamente nel costruttore:

```
parent::__construct ($id);
```

Criteri di costruzione di un modello/tabella per la definizione della struttura

Le tabelle che si riferiscono alle applicazioni possono essere gestite in modo automatico attraverso la classe [Gino](#).↔
[AdminTable](#).

Ognuna di queste tabelle viene definita in un modello che estende la classe [Gino.Model](#), e in particolare il modello definisce i propri campi nel metodo statico `columns()`. In questo metodo vengono definiti tutti i campi del modello utilizzando le opzioni generali dei campi (

See also

[Gino.Field::__construct\(\)](#) e quelle specifiche del tipo di campo.

Le tabelle devono essere coerenti con la definizione del modello per cui, ad esempio, i campi obbligatori devono essere 'not null' e gli eventuali valori di default devono essere indicati anche nel campo della tabella.

Copyright

2014-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.90.2 Constructor & Destructor Documentation

8.90.2.1 `__construct ($id = null)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record dell'oggetto
---------	-------------	-----------------------------------

Returns

istanza di [Gino.Model](#)

8.90.3 Member Function Documentation

8.90.3.1 `__get ($pName)`

Metodo richiamato ogni volta che qualcuno prova a ottenere una proprietà dell'oggetto non definita.

L'output è il metodo get specifico per questa proprietà (se esiste), altrimenti è la proprietà.

Per i campi su tabella principale la proprietà ritornata è uguale al valore salvato sul db.

Per i m2m la proprietà è uguale ad un array con gli id dei modelli correlati.

Per i m2mt la proprietà è uguale ad un array con gli id dei modelli correlati.

Parameters

string	<i>\$pName</i>	nome della proprietà
--------	----------------	----------------------

Returns

valore proprietà

8.90.3.2 `__set ($pName, $pValue)`

Metodo richiamato ogni volta che qualcuno prova a impostare una proprietà dell'oggetto non definita (`$this->{fieldname}`)

Parameters

string	<i>\$pName</i>	nome della proprietà
mixed	<i>\$pValue</i>	valore da impostare

Returns

void

8.90.3.3 `__toString ()`

Rappresentazione a stringa dell'oggetto Sovrascrivere questo metodo nella classe figlia per restituire un valore parlante.

Returns

id

8.90.3.4 `build ($field_obj)`

Classe [Build](#) del campo di tabella.

Le eventuali proprietà del modello dipendenti dai valori del record sovrascrivono le proprietà del campo

Parameters

object	<i>\$field_obj</i>	oggetto della classe del tipo di campo
--------	--------------------	--

Returns

object

8.90.3.5 `checkIsConstraint ()` [protected]

Controllo delle regole constraint.

Returns

true oppure lista delle regole violate

8.90.3.6 `checkM2m ($field_obj)` [private]

Verifica se il tipo di campo di un modello è un oggetto ManyToMany.

Parameters

object	<i>\$field</i>	oggetto del tipo di campo
--------	----------------	---------------------------

Returns

boolean

8.90.3.7 `static columns ()` [static]

Recupera l'oggetto del tipo di campo di un modello.

Parameters

string	<i>\$field_name</i>	nome del campo
--------	---------------------	----------------

Returns

object or null Struttura dei campi della tabella di un modello
array

Il formato degli elementi dell'array è il seguente:

```
field_name = new \Gino\Type\Field(array(  
    'name' => string,  
    'label' => string,  
    'primary_key' => bool,  
    'unique_key' => bool,  
    'auto_increment' => bool,  
    'default' => mixed,  
    'max_lenght' => integer,  
    'required' => boolean,  
    'int_digits' => integer,  
    'decimal_digits' => integer,  
));
```

8.90.3.8 delete ()

Elimina l'oggetto Elimina i dati su db, le traduzioni, e le associazioni m2m e m2mt Controlla che non ci siano regole di constraint che impediscano l'eliminazione, in caso ce ne fossero di non rispettate ritorna un elenco di regole che impediscono l'eliminazione.

Returns

bool (true) or array (error)

8.90.3.9 deleteDbData ()

Elimina le proprietà su db del modello e le traduzioni.

Returns

risultato dell'operazione, bool

8.90.3.10 static deleteInstance (*\$controller*) [static]

Eliminazione di tutti i record legati all'istanza del controller passato come argomento.

Parameters

mixed	<i>\$controller</i>	istanza del controller
-------	---------------------	------------------------

Returns

TRUE

8.90.3.11 deleteM2m ()

Elimina le associazioni m2m.

Returns

risultato dell'operazione, bool

8.90.3.12 deleteM2mthrough ()

Elimina le associazioni m2mt.

Returns

risultato dell'operazione, bool

8.90.3.13 deleteM2mthroughField (*\$field_name*)

Elimina le associazioni di un campo m2mt.

Parameters

string	<i>\$field_name</i>	nome campo
--------	---------------------	------------

Returns

risultato dell'operazione, bool

8.90.3.14 fetchColumns (*\$id*)

Recupera i valori del record e li carica nella proprietà `_p`. Il valore dei campi di tipo `ManyToOne` è un array che racchiude i valori id dei record della tabella di join associata al modello.

Parameters

integer	<i>\$id</i>	valore id del record
---------	-------------	----------------------

Exceptions

--	--

8.90.3.15 fieldLabel (*\$field*)

Etichetta del campo.

Parameters

string	<i>\$field</i>	nome campo
--------	----------------	------------

Returns

etichetta

8.90.3.16 getController ()

[Controller](#) del modello.

Returns

istanza del controller o null se non esiste istanza

8.90.3.17 static getFromSlug (*\$slug*, *\$controller = null*) [static]

Recupera l'oggetto a partire dallo slug.

Parameters

string	<i>\$slug</i>	slug
mixed	<i>\$controller</i>	istanza del controller.

Returns

oggetto che matcha lo slug dato

8.90.3.18 getModelLabel ()

Etichetta del modello.

Returns

label

8.90.3.19 getProperties (\$field_obj)

Imposta le opzioni "model, field_object, value, table" e recupera le proprietà del campo dipendenti dai valori del record.

Parameters

object	<i>\$field_obj</i>	oggetto della classe del tipo di campo
--------	--------------------	--

Returns

array

8.90.3.20 getRecordValues ()

Valori di un record.

Parameters

string	<i>\$id</i>	valore id del record
--------	-------------	----------------------

Returns

multitype:array,null

8.90.3.21 static getSelectOptionsFromObjects (\$objects) [static]

Array associativo id => rappresentazione a stringa a partire da array di oggetti.

Parameters

array	<i>\$objects</i>	
-------	------------------	--

Returns

array associativo id=>stringa

8.90.3.22 getStructure ()

Struttura dei campi del modello Un array associativo che contiene tutti i campi come chiavi e le relative classi di tipo [Field](#) come valore.

Returns

struttura dati

8.90.3.23 `getTable ()`

Tabella principale dei dati.

Returns

nome tabella

8.90.3.24 `isConstraintError ($res)` [protected]

Errore conseguente ad una violazione delle constraint in eliminazione.

Parameters

array	<i>\$res</i>	array delle regole constraint violate
-------	--------------	---------------------------------------

Returns

html errore

8.90.3.25 `m2mtObject ($m2mt_field, $id)`

Ritorna l'oggetto ManyToMany through model.

Parameters

string	<i>\$m2mt_field</i>	nome del campo m2mt
int	<i>\$id</i>	id del record

Returns

oggetto

8.90.3.26 `ml ($pName)`

Recupera le proprietà con la traduzione.

Parameters

string	<i>\$pName</i>	nome proprietà
--------	----------------	----------------

Returns

string traduzione

8.90.3.27 `static objects ($controller = null, $options = array())` [static]

Metodo generico statico per ricavare oggetti.

Parameters

mixed	<i>\$controller</i>	istanza del controller
array	<i>\$options</i>	array associativo di opzioni: <ul style="list-style-type: none">• where: where clause• order: ordinamento• limit: limite risultati

Returns

array di oggetti ricavati

8.90.3.28 static properties (*\$model*, *\$controller* = null) [static],[protected]

Proprietà specifiche di un modello dipendenti dai valori del record (ad esempio dal valore id)

Parameters

object	<i>\$model</i>	
object	<i>\$controller</i>	

Returns

array

8.90.3.29 refreshModel ()

Refresh del modello (da chiamare manualmente)

Quando ad esempio si modificano gli m2mt e si vogliono vederne gli effetti prima del ricaricamento pagina. Modificando gli m2mt, questi vengono aggiornati sul db, ma il modello che ha tali m2mt continua a referenziare i vecchi, questo perché il salvataggio viene gestito da [Gino.AdminTable](#) e non da modello stesso che quindi ne è quasi all'oscuro. Ora questo metodo viene anche chiamato da [AdminTable](#) e quindi le modifiche si riflettono immediatamente anche sul modello. Chiamarlo manualmente se la modifica agli m2mt viene fatta in modo diverso dall'uso di [Gino.AdminTable::modelAction\(\)](#).

See also

[Gino.AdminTable::m2mthroughAction\(\)](#)

Returns

void

8.90.3.30 save (*\$options* = array())

Salva il modello su db Salva sia i campi della tabella sia i m2m.

I m2mt devono essere salvati manualmente, la classe [AdminTable](#) lo fa in maniera automatica. Quando il salvataggio avviene con successo viene emesso un segnale 'post_save' da parte del modello.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • only_update (string array): nome o nomi dei campi da aggiornare <ul style="list-style-type: none"> – string, nome o nomi dei campi separati da virgola • no_update (array): elenco dei campi da non impostare in una istruzione di update; di default vengono aggiunti i campi 'id', 'instance'
-------	------------------	---

Returns

il risultato dell'operazione o errori

8.90.3.31 `savem2m ($m2m)`

Salvataggio dei ManyToMany.

Parameters

array	<i>\$m2m</i>	campi m2m del modello (field_name => (array) join_table_id_values)
-------	--------------	--

Returns

true

8.90.3.32 setCheckIsConstraint (*\$check*) [protected]

Setter per la variabile di controllo del check constraint.

Parameters

bool	<i>\$check</i>	
------	----------------	--

Returns

void

8.90.3.33 setIsConstraint (*\$is_constraint*)

Setter per la proprietà che contiene le informazioni per il check dei constraint Esempio:

```
$is_constraint = array(
    'MyModelClass'=>'field_name',
    'MyModelClassWithInstanceController'=>array('field' => 'field_name', 'controller' => new mycontroller()
    )
);
```

Parameters

array	<i>\$is_constraint</i>	
-------	------------------------	--

Returns

void

8.90.3.34 shows (*\$field_obj*)

Valore da mostrare in output.

See also

[Gino.Build::retrieveValue\(\)](#)

Parameters

object	<i>\$field_obj</i>	oggetto della classe del tipo di campo
--------	--------------------	--

Returns

mixed

8.90.4 Field Documentation

8.90.4.1 `$_check_is_constraint = true` [protected]

8.90.4.2 `$_controller` [protected]

8.90.4.3 `$_db`

8.90.4.4 `$_is_constraint = array()` [protected]

8.90.4.5 `$_lng_dft` [protected]

8.90.4.6 `$_lng_nav`

8.90.4.7 `$_locale` [protected]

8.90.4.8 `$_model_label` [protected]

8.90.4.9 `$_p = array()` [protected]

8.90.4.10 `$_registry` [protected]

8.90.4.11 `$_request`

8.90.4.12 `$_structure` [private]

8.90.4.13 `$_tbl_data` [protected]

8.90.4.14 `$_trd` [private]

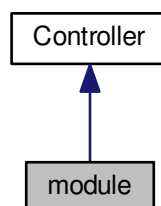
The documentation for this class was generated from the following file:

- lib/classes/mvc/[class.Model.php](#)

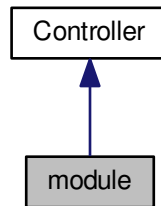
8.91 module Class Reference

Classe di tipo [Gino.Controller](#) per la gestione di istanze di moduli di sistema.

Inheritance diagram for module:



Collaboration diagram for module:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [manageModule](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa per la gestione dei moduli.
- [actionRemoveModule](#) (\Gino\Http\Request \$request)
Processa il form di eliminazione di un modulo.
- [actionModule](#) (\Gino\Http\Request \$request)
Processa il form di inserimento/modifica modulo.
- [actionEditModuleActive](#) (\Gino\Http\Request \$request)
Processa il form di attivazione e disattivazione di un modulo.

Private Member Functions

- [listModule](#) ()
Elenco dei moduli.
- [formRemoveModule](#) (\$module)
Form di eliminazione di un modulo.
- [deleteModuleInstance](#) (\$module)
Elimina automaticamente associazioni con permessi, css, viste e contenuti di un'istanza di modulo.
- [formModule](#) (\$module)
Form di inserimento/modifica di un modulo.
- [actionInsertModule](#) (\Gino\Http\Request \$request)
Processa il form di inserimento di un modulo.
- [createMdlFolders](#) (\$pdir, \$nsdir)
Creazione cartelle contenuti modulo.
- [actionEditModule](#) (\Gino\Http\Request \$request)
Processa il form di modifica di un modulo.
- [formActivateModule](#) (\$module)
Form di attivazione e disattivazione di un modulo.

Additional Inherited Members

8.91.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione di istanze di moduli di sistema.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.91.2 Constructor & Destructor Documentation

8.91.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.App.Module.module](#)

8.91.3 Member Function Documentation

8.91.3.1 `actionEditModule (\Gino\Http\Request $request)` [`private`]

Processa il form di modifica di un modulo.

See also

`self::formModule()`
`self::actionModule()`

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Redirect](#)

8.91.3.2 `actionEditModuleActive (\Gino\Http\Request $request)`

Processa il form di attivazione e disattivazione di un modulo.

See also

`self::formActivateModule()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.91.3.3 `actionInsertModule (\Gino\Http\Request $request)` [private]

Processa il form di inserimento di un modulo.

See also

`self::formModule()`
`self::actionModule()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.91.3.4 `actionModule (\Gino\Http\Request $request)`

Processa il form di inserimento/modifica modulo.

See also

`self::formModule()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.91.3.5 `actionRemoveModule (\Gino\Http\Request $request)`

Processa il form di eliminazione di un modulo.

See also

`self::formRemoveModule()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.91.3.6 createMdlFolders (*\$pdir*, *\$nsdir*) [private]

Creazione cartelle contenuti modulo.

Parameters

string	<i>\$pdir</i>	directory di riferimento
array	<i>\$nsdir</i>	array di sottodirectory della direcotry di riferimento

Returns

TRUE

8.91.3.7 deleteModuleInstance (*\$module*) [private]

Elimina automaticamente associazioni con permessi, css, viste e contenuti di un'istanza di modulo.

Parameters

\Gino\↔ App\↔ Module\↔ Module↔ Instance	<i>\$module</i>	istanza di Gino.App.Module.ModuleInstance
---	-----------------	---

Returns

void

8.91.3.8 formActivateModule (*\$module*) [private]

[Form](#) di attivazione e disattivazione di un modulo.

Parameters

\Gino\↔ App\↔ Module\↔ Module↔ Instance	<i>\$module</i>	istanza di Gino.App.Module.ModuleInstance
---	-----------------	---

Returns

html, form

8.91.3.9 formModule (*\$module*) [private]

[Form](#) di inserimento/modifica di un modulo.

Parameters

\Gino\↵ App\↵ Module\↵ Module↵ Instance	<i>\$module</i>	istanza di Gino.App.Module.ModuleInstance
---	-----------------	---

Returns

html, form

8.91.3.10 `formRemoveModule ($module)` [private]

Form di eliminazione di un modulo.

Parameters

\Gino\↵ App\↵ Module\↵ Module↵ Instance	<i>\$module</i>	istanza di Gino.App.Module.ModuleInstance
---	-----------------	---

Returns

html, form

8.91.3.11 `listModule ()` [private]

Elenco dei moduli.

Parameters

integer	<i>\$sel_id</i>	valore ID del modulo selezionato
---------	-----------------	----------------------------------

Returns

html, lista moduli

8.91.3.12 `manageModule (\Gino\Http\Request $request)`

Interfaccia amministrativa per la gestione dei moduli.

Parameters

\Gino\↵ Http\↵ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

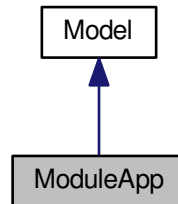
The documentation for this class was generated from the following file:

- `app/module/class_module.php`

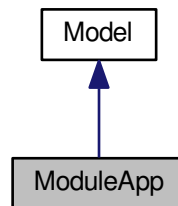
8.92 ModuleApp Class Reference

Classe di tipo [Gino.Model](#) che rappresenta un modulo di sistema.

Inheritance diagram for ModuleApp:



Collaboration diagram for ModuleApp:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [className](#) ()
Nome della classe di sistema.
- [classNameNs](#) ()
Nome della classe di sistema con namespace completo.

Static Public Member Functions

- static [getFromName](#) (\$name)
Recupera l'oggetto dato il nome della classe di sistema.
- static [columns](#) ()
Struttura dei campi della tabella di un modello.

Static Public Attributes

- static `$table` = `TBL_MODULE_APP`
- static `$columns`

Additional Inherited Members

8.92.1 Detailed Description

Classe di tipo `Gino.Model` che rappresenta un modulo di sistema.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.92.2 Constructor & Destructor Documentation

8.92.2.1 `__construct ($id)`

Costruttore.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

istanza di `Gino.App.SysClass.ModuleApp`

8.92.3 Member Function Documentation

8.92.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

label modulo

8.92.3.2 `className ()`

Nome della classe di sistema.

Returns

nome classe (senza namespace)

8.92.3.3 `classNameNs ()`

Nome della classe di sistema con namespace completo.

Returns

nome classe (con namespace)

8.92.3.4 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.92.3.5 `static getFromName ($name) [static]`

Recupera l'oggetto dato il nome della classe di sistema.

Parameters

string	<i>\$name</i>	nome classe senza namespace
--------	---------------	-----------------------------

Returns

istanza di [Gino.App.SysClass.ModuleApp](#)

8.92.4 Field Documentation

8.92.4.1 `$columns [static]`

8.92.4.2 `$table = TBL_MODULE_APP [static]`

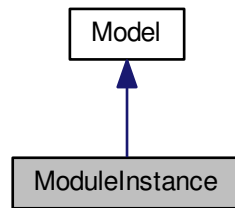
The documentation for this class was generated from the following file:

- [app/sysClass/class.ModuleApp.php](#)

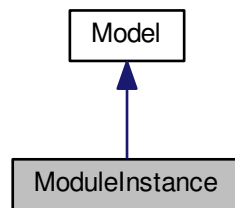
8.93 ModuleInstance Class Reference

Classe di tipo [Gino.Model](#) che rappresenta un'istanza di un modulo di sistema.

Inheritance diagram for ModuleInstance:



Collaboration diagram for ModuleInstance:



Public Member Functions

- `__construct` (\$id)
Costruttore.
- `className` ()
Nome della classe.
- `classNameNs` (\$ns=true)
Nome della classe con namespace completo.
- `moduleApp` ()
Modulo di sistema dell'istanza.

Static Public Member Functions

- static `getFromName` (\$name)
Recupera l'oggetto dato il nome dell'istanza.
- static `getFromModuleApp` (\$module_app_id)
Recupera le istanze dato l'id classe di sistema.
- static `columns` ()
Struttura dei campi della tabella di un modello.

Static Public Attributes

- static `$table` = `TBL_MODULE`
- static `$columns`

Additional Inherited Members

8.93.1 Detailed Description

Classe di tipo `Gino.Model` che rappresenta un'istanza di un modulo di sistema.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.93.2 Constructor & Destructor Documentation

8.93.2.1 `__construct ($id)`

Costruttore.

Parameters

int	<i>\$id</i>	
-----	-------------	--

Returns

istanza di `Gino.App.Module.ModuleInstance`

8.93.3 Member Function Documentation

8.93.3.1 `className ()`

Nome della classe.

Returns

nome classe

8.93.3.2 `classNameNs ($ns = true)`

Nome della classe con namespace completo.

Returns

nome classe con namespace

8.93.3.3 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.93.3.4 static getFromModuleApp (\$module_app_id) [static]

Recupera le istanze dato l'id classe di sistema.

Parameters

int	<i>\$module_app_id</i>	
-----	------------------------	--

Returns

array di istanze di [Gino.App.Module.ModuleInstance](#)

8.93.3.5 static getFromName (\$name) [static]

Recupera l'oggetto dato il nome dell'istanza.

Parameters

string	<i>\$name</i>	nome istanza senza namespace
--------	---------------	------------------------------

Returns

istanza di [Gino.App.Module.ModuleInstance](#)

8.93.3.6 moduleApp ()

Modulo di sistema dell'istanza.

Returns

istanza di [Gino.App.SysClass.ModuleApp](#)

8.93.4 Field Documentation

8.93.4.1 \$columns [static]

8.93.4.2 \$table = TBL_MODULE [static]

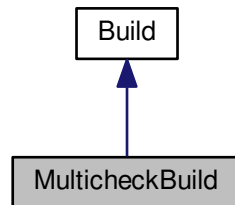
The documentation for this class was generated from the following file:

- app/module/[class.ModuleInstance.php](#)

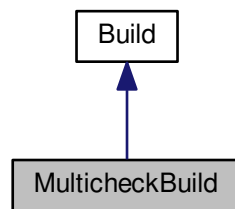
8.94 MulticheckBuild Class Reference

Gestisce i campi di tipo multicheck.

Inheritance diagram for MulticheckBuild:



Collaboration diagram for MulticheckBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [getChoice](#) ()
Getter della proprietà choice.
- [formElement](#) (\Gino\Form \$form, \$options)
- [clean](#) (\$options=null)
- [filterWhereClause](#) (\$value)
- [retrieveValue](#) ()

Data Fields

- [\\$_refmodel_order](#)
- [\\$_refmodel_where](#)
- [\\$_refmodel_controller](#)

Protected Attributes

- [\\$_refmodel](#)
Proprietà dei campi specifiche del tipo di campo.
- [\\$_choice](#)

8.94.1 Detailed Description

Gestisce i campi di tipo multichcek.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.94.2 Constructor & Destructor Documentation

8.94.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.94.3 Member Function Documentation

8.94.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

rappresentazione a stringa dei modelli associati separati da virgola

8.94.3.2 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#)

Returns

string

8.94.3.3 filterWhereClause (*\$value*)

See also

[Gino.Build::filterWhereClause\(\)](#)

Parameters

array	<i>\$value</i>	
-------	----------------	--

8.94.3.4 formElement (\Gino\Form *\$form*, *\$options*)

See also

[Gino.Build::formElement\(\)](#)

8.94.3.5 getChoice ()

Getter della proprietà choice.

Returns

proprietà choice

8.94.3.6 retrieveValue ()

See also

[Gino.Build::retrieveValue\(\)](#)

8.94.4 Field Documentation

8.94.4.1 `$_choice` [protected]

8.94.4.2 `$_refmodel` [protected]

Proprietà dei campi specifiche del tipo di campo.

8.94.4.3 `$_refmodel_controller`

8.94.4.4 `$_refmodel_order`

8.94.4.5 `$_refmodel_where`

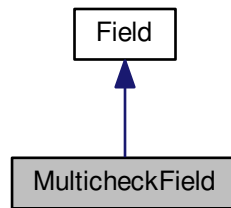
The documentation for this class was generated from the following file:

- lib/classes/build/[class.MulticheckBuild.php](#)

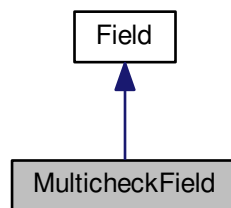
8.95 MulticheckField Class Reference

Campo di tipo multicheck.

Inheritance diagram for MulticheckField:



Collaboration diagram for MulticheckField:



Public Member Functions

- `__construct` (\$options)
Costruttore.
- `getProperties` ()

Data Fields

- `$_refmodel_where`
- `$_refmodel_order`
- `$_refmodel_controller`

Protected Attributes

- `$_refmodel`
Proprietà dei campi specifiche del tipo di campo.
- `$_choice`

8.95.1 Detailed Description

Campo di tipo multicheck.

I valori da associare al campo risiedono in un'altra tabella e i parametri per accedervi devono essere definiti nelle opzioni del campo.

Tipologie di input associabili: multicheck

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.95.2 Constructor & Destructor Documentation

8.95.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – refmodel (object): classe del modello correlato (nome completo di namespace) – refmodel_where (mixed): condizioni della query per filtrare i possibili modelli da associare <ul style="list-style-type: none"> * <i>string</i>, es. "cond1='\$cond1' AND cond2='\$cond2'" * <i>array</i>, es. array("cond1='\$cond1'", "cond2='\$cond2'") – refmodel_order (string): ordinamento dei valori (es. name ASC) – refmodel_controller (): classe Controller del modello refmodel, essenziale se il modello appartiene ad un modulo istanziabile – choice (array): elenco degli elementi di scelta
-------	------------------	---

8.95.3 Member Function Documentation

8.95.3.1 getProperties ()

See also

[Gino.Field::getProperties\(\)](#)

8.95.4 Field Documentation

8.95.4.1 `$_choice` `[protected]`

8.95.4.2 `$_refmodel` `[protected]`

Proprietà dei campi specifiche del tipo di campo.

8.95.4.3 `$_refmodel_controller`

8.95.4.4 `$_refmodel_order`

8.95.4.5 `$_refmodel_where`

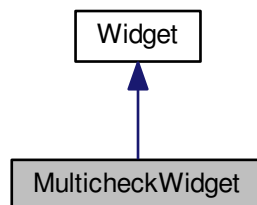
The documentation for this class was generated from the following file:

- `lib/classes/fields/class.MulticheckField.php`

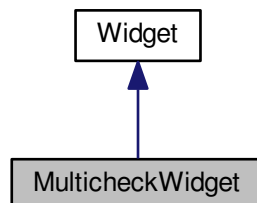
8.96 MulticheckWidget Class Reference

Campi di tipo multicheck nei form.

Inheritance diagram for MulticheckWidget:



Collaboration diagram for MulticheckWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.96.1 Detailed Description

Campi di tipo multichек nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.96.2 Member Function Documentation

8.96.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

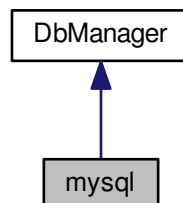
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.MultichекWidget.php](#)

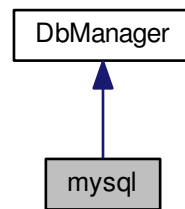
8.97 mysql Class Reference

Libreria di connessione ai database MySQL.

Inheritance diagram for mysql:



Collaboration diagram for mysql:



Public Member Functions

- [__construct](#) (\$params)
Costruttore.
- [getInfoQuery](#) ()
- [openConnection](#) ()
- [closeConnection](#) ()
Chiude le connessioni non persistenti a un server MySQL.
- [begin](#) ()
Per tabelle innodb.
- [rollback](#) ()
Per tabelle innodb.
- [commit](#) ()
Per tabelle innodb.
- [multiActionQuery](#) (\$file_content)
- [freeresult](#) (\$res=null)
- [getLastId](#) (\$table)
Il valore della funzione SQL LAST_INSERT_ID() di MySQL contiene sempre il più recente valore AUTO_INCREMENT generato e non è azzerato dalle query.
- [autoIncValue](#) (\$table)
Ottiene il valore del campo AUTO_INCREMENT.
- [getFieldFromId](#) (\$table, \$field, \$field_id, \$id, \$options=array())
- [tableexists](#) (\$table)
- [fieldInformations](#) (\$table)
- [conformFieldType](#) (\$type)
- [limit](#) (\$range, \$offset)
- [distinct](#) (\$fields, \$options=array())
- [concat](#) (\$sequence)
- [dumpDatabase](#) (\$file)
- [getTableStructure](#) (\$table)
- [changeFieldType](#) (\$data_type, \$value)
- [getFieldsName](#) (\$table)
- [getNumRecords](#) (\$table, \$where=null, \$field='id', \$options=array())
- [query](#) (\$fields, \$tables, \$where=null, \$options=array())
- [execCustomQuery](#) (\$query, \$options=array())
- [select](#) (\$fields, \$tables, \$where=null, \$options=array())
- [insert](#) (\$fields, \$table, \$debug=false)

- [update](#) (\$fields, \$table, \$where, \$debug=false)
- [delete](#) (\$table, \$where, \$debug=false)
- [drop](#) (\$table)
- [columnHasValue](#) (\$table, \$field, \$value, \$options=array())
- [join](#) (\$table, \$condition, \$option)
- [union](#) (\$queries, \$options=array())
- [restore](#) (\$table, \$filename, \$options=array())
- [dump](#) (\$table, \$filename, \$options=array())
- [escapeString](#) (\$string)

Data Fields

- [\\$_db_name](#)
- [\\$_db_user](#)
- [\\$_db_password](#)
- [\\$_db_charset](#)
- [\\$_dbconn](#)

Private Member Functions

- [setParamsCache](#) ()
Parametri inerenti la cache delle query presenti nelle Impostazioni di sistema.
- [setsql](#) (\$sql_query)
Imposta la query come proprietà
- [setNumberRows](#) (\$numberresults)
- [setconnection](#) (\$connection)
- [queryResults](#) (\$query, \$options=array())
Esegue la query e ne calcola i tempi di esecuzione.
- [setUtf8](#) ()

Private Attributes

- [\\$_db_host](#)
- [\\$_sql](#)
- [\\$_qry](#)
- [\\$_numberrows](#)
- [\\$_connection](#)
- [\\$_rows](#)
- [\\$_affected](#)
- [\\$_lastid](#)
- [\\$_dbresults](#) = array()
- [\\$_show_stats](#)
- [\\$_info_queries](#)
- [\\$_cnt](#)
- [\\$_time_queries](#)
- [\\$_query_cache](#)
- [\\$_query_cache_time](#)
- [\\$_cache](#)

8.97.1 Detailed Description

Libreria di connessione ai database MySQL.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

CACHE QUERY

La proprietà `self::$_query_cache` indica se è stata abilita la cache delle query.

Le query che vengono salvate in cache sono quelle che passano dal metodo `select()` ed `execCustomQuery()`, e non riguardano quindi le query di struttura, quali quelle presenti nei metodi:

- `fieldInformations()`
- `getTableStructure()`
- `getFieldsName()`

Qualora non si desideri caricare in cache una determinata query è sufficiente passare l'opzione `cache=false` ai metodi `select()` e `execCustomQuery()`.

La cache delle query viene svuotata ogni volta che viene effettuata una istruzione di tipo action.

INFORMAZIONI SULLE QUERY

La proprietà `self::$_show_stats` attiva la raccolta di informazioni sulle prestazioni delle chiamate al database.

Le query di tipo select alimentano i contatori `self::$_cnt` e `self::$_time_queries`.

8.97.2 Constructor & Destructor Documentation

8.97.2.1 `__construct ($params)`

Costruttore.

Parameters

array	<i>\$params</i>	parametri di connessione al database <ul style="list-style-type: none"> • host (string): nome del server • db_name (string): nome del database • user (string): utente che si connette • password (string): password dell'utente che si connette • charset (string): encoding • connect (boolean): attiva la connessione
-------	-----------------	--

Returns

void

Implements [DbManager](#).

8.97.3 Member Function Documentation

8.97.3.1 autoIncValue (*\$table*)

Ottiene il valore del campo AUTO_INCREMENT.

See also

[DbManager::autoIncValue\(\)](#)

Implements [DbManager](#).

8.97.3.2 begin ()

Per tabelle innodb.

See also

[DbManager::begin\(\)](#)

Implements [DbManager](#).

8.97.3.3 changeFieldType (*\$data_type*, *\$value*)

See also

[Gino.DbManager::changeFieldType\(\)](#)

Implements [DbManager](#).

8.97.3.4 closeConnection ()

Chiude le connessioni non persistenti a un server MySQL.

See also

[DbManager::closeConnection\(\)](#)

Implements [DbManager](#).

8.97.3.5 columnHasValue (*\$table*, *\$field*, *\$value*, *\$options* = array())

See also

[DbManager::columnHasValue\(\)](#)

Implements [DbManager](#).

8.97.3.6 `commit ()`

Per tabella innodb.

See also

[DbManager::commit\(\)](#)

Implements [DbManager](#).

8.97.3.7 `concat ($sequence)`

See also

[DbManager::concat\(\)](#)

Implements [DbManager](#).

8.97.3.8 `conformFieldType ($type)`

See also

[DbManager::conformFieldType\(\)](#)

Implements [DbManager](#).

8.97.3.9 `delete ($table, $where, $debug = false)`

See also

[DbManager::delete\(\)](#)

Implements [DbManager](#).

8.97.3.10 `distinct ($fields, $options = array())`

See also

[DbManager::distinct\(\)](#)

Implements [DbManager](#).

8.97.3.11 `drop ($table)`

See also

[DbManager::drop\(\)](#)

Implements [DbManager](#).

8.97.3.12 `dump ($table, $filename, $options = array())`

See also

[DbManager::dump\(\)](#)

Per poter effettuare questa operazione occorre:

- assegnare il permesso FILE all'utente del database: `GRANT FILE ON *.* TO 'dbuser'@'localhost';`
- la directory di salvataggio deve avere i permessi 777, oppure deve avere come proprietario l'utente di sistema mysql (gruppo mysql)

Implements [DbManager](#).

8.97.3.13 dumpDatabase (*\$file*)

See also

[DbManager::dumpDatabase\(\)](#)

Implements [DbManager](#).

8.97.3.14 escapeString (*\$string*)

See also

[DbManager::escapeString\(\)](#)

Implements [DbManager](#).

8.97.3.15 execCustomQuery (*\$query*, *\$options* = array())

See also

[Gino.DbManager::execCustomQuery\(\)](#)

Implements [DbManager](#).

8.97.3.16 fieldInformations (*\$table*)

See also

[DbManager::fieldInformations\(\)](#)
[conformFieldType\(\)](#)

Implements [DbManager](#).

8.97.3.17 freeresult (*\$res* = null)

See also

[DbManager::freeresult\(\)](#)

Implements [DbManager](#).

8.97.3.18 `getFieldFromId ($table, $field, $field_id, $id, $options = array())`

See also

[DbManager::getFieldFromId\(\)](#)

Implements [DbManager](#).

8.97.3.19 `getFieldsName ($table)`

See also

[DbManager::getFieldsName\(\)](#)
[freeresult\(\)](#)

Implements [DbManager](#).

8.97.3.20 `getInfoQuery ()`

See also

[DbManager::getInfoQuery\(\)](#)

Implements [DbManager](#).

8.97.3.21 `getLastId ($table)`

Il valore della funzione SQL `LAST_INSERT_ID()` di MySQL contiene sempre il più recente valore `AUTO_INCREMENT` generato e non è azzerato dalle query.

See also

[DbManager::getLastId\(\)](#)

Implements [DbManager](#).

8.97.3.22 `getNumRecords ($table, $where = null, $field = 'id', $options = array())`

See also

[DbManager::getNumRecords\(\)](#)

Implements [DbManager](#).

8.97.3.23 `getTableStructure ($table)`

See also

[DbManager::getTableStructure\(\)](#)

Implements [DbManager](#).

8.97.3.24 `insert ($fields, $table, $debug = false)`

See also

[DbManager::insert\(\)](#)

Implements [DbManager](#).

8.97.3.25 `join ($table, $condition, $option)`

See also

[DbManager::join\(\)](#)

Implements [DbManager](#).

8.97.3.26 `limit ($range, $offset)`

See also

[DbManager::limit\(\)](#)

Implements [DbManager](#).

8.97.3.27 `multiActionQuery ($file_content)`

See also

[DbManager::multiActionQuery\(\)](#)

Implements [DbManager](#).

8.97.3.28 `openConnection ()`

See also

[DbManager::openConnection\(\)](#)

Implements [DbManager](#).

8.97.3.29 `query ($fields, $tables, $where = null, $options = array())`

See also

[DbManager::query\(\)](#)

Implements [DbManager](#).

8.97.3.30 `queryResults ($query, $options = array())` [private]

Esegue la query e ne calcola i tempi di esecuzione.

Parameters

string	<i>\$query</i>	query da eseguire
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• statement (string): tipologia di query<ul style="list-style-type: none">– <i>select</i> (default)– <i>action</i>

Returns

mixed

8.97.3.31 `restore ($table, $filename, $options = array())`

See also

[DbManager::restore\(\)](#)

Implements [DbManager](#).

8.97.3.32 `rollback ()`

Per tabelle innodb.

See also

[DbManager::rollback\(\)](#)

Implements [DbManager](#).

8.97.3.33 `select ($fields, $tables, $where = null, $options = array())`

See also

[DbManager::select\(\)](#)

Implements [DbManager](#).

8.97.3.34 `setconnection ($connection)` [private]

8.97.3.35 `setNumberRows ($numberresults)` [private]

8.97.3.36 `setParamsCache ()` [private]

Parametri inerenti la cache delle query presenti nelle Impostazioni di sistema.

Returns

void

8.97.3.37 `setsql ($sql_query)` [private]

Imposta la query come proprietà

Parameters

string	<i>\$sql_query</i>	query
--------	--------------------	-------

8.97.3.38 `setUtf8 ()` [private]

8.97.3.39 `tableexists ($table)`

See also

[DbManager::tableexists\(\)](#)

Implements [DbManager](#).

8.97.3.40 `union ($queries, $options = array())`

See also

[DbManager::union\(\)](#)

Implements [DbManager](#).

8.97.3.41 `update ($fields, $table, $where, $debug = false)`

See also

[DbManager::update\(\)](#)

Implements [DbManager](#).

8.97.4 Field Documentation

8.97.4.1 `$_affected` [private]

8.97.4.2 `$_cache` [private]

8.97.4.3 `$_cnt` [private]

8.97.4.4 `$_connection` [private]

8.97.4.5 `$_db_charset`

8.97.4.6 `$_db_host` [private]

8.97.4.7 `$_db_name`

8.97.4.8 `$_db_password`

8.97.4.9 `$_db_user`

8.97.4.10 `$_dbconn`

8.97.4.11 `$_dbresults = array()` [private]

8.97.4.12 `$_info_queries` [private]

8.97.4.13 `$_lastid` [private]

8.97.4.14 `$_numberrows` [private]

8.97.4.15 `$_qry` [private]

8.97.4.16 `$_query_cache` [private]

8.97.4.17 `$_query_cache_time` [private]

8.97.4.18 `$_rows` [private]

8.97.4.19 `$_show_stats` [private]

8.97.4.20 `$_sql` [private]

8.97.4.21 `$_time_queries` [private]

The documentation for this class was generated from the following file:

- lib/plugin/[plugin.mysql.php](#)

8.98 Options Class Reference

Gestisce le opzioni di classe, costruendo il form ed effettuando l'action.

Public Member Functions

- [__construct](#) (\$controller)
Costruttore.
- [manageDoc](#) ()
Interfaccia per la gestione delle opzioni di una istanza/modulo ([Form](#))
- [actionOptions](#) ()
Processa il form di opzioni.

Data Fields

- [\\$_module_app](#)
- [\\$_class_prefix](#)

Private Member Functions

- [setData](#) (\$controller)
Imposta alcune variabili utilizzate dalla classe.
- [editableField](#) (\$field)
Controlla se il campo è editabile.

Private Attributes

- [\\$_db](#)
- [\\$_class](#)
- [\\$_tbl_options](#)
- [\\$_instance](#)
- [\\$_title](#)
- [\\$_home](#)
- [\\$_action](#)

8.98.1 Detailed Description

Gestisce le opzioni di classe, costruendo il form ed effettuando l'action.

Le opzioni che possono essere associate a ciascun campo sono:

- **label** (string): nome della label

- **value** (mixed): valore di default
- **required** (boolean): campo obbligatorio
- **trnsi** (boolean): campo che necessita di traduzione
- **section** (boolean): segnala l'inizio di un blocco di opzioni
- **section_title** (string): nome del blocco di opzioni
- **section_description** (string): descrizione del blocco di opzioni

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.98.2 Constructor & Destructor Documentation

8.98.2.1 __construct (*\$controller*)

Costruttore.

Parameters

\Gino\↔ Controller	<i>\$controller</i>	istanza di Gino.Controller
-----------------------	---------------------	--

Returns

istanza di [Gino.Options](#)

8.98.3 Member Function Documentation

8.98.3.1 actionOptions ()

Processa il form di opzioni.

Returns

Gino.Gtpp.Redirect

8.98.3.2 editableField (*\$field*) [private]

Controlla se il campo è editabile.

Returns

True se editabile, FALSE altrimenti

8.98.3.3 `manageDoc ()`

Interfaccia per la gestione delle opzioni di una istanza/modulo ([Form](#))

Come informazioni sui campi sono necessarie:

- **name** (string): nome del campo
- **type** (string): tipo di campo
- **length** (integer): numero massimo di caratteri

See also

`Gino.Db::fieldInformations()`

Returns

interfaccia

8.98.3.4 `setData ($controller)` [private]

Imposta alcune variabili utilizzate dalla classe.

Exceptions

--	--

8.98.4 Field Documentation

8.98.4.1 `$_action` [private]

8.98.4.2 `$_class` [private]

8.98.4.3 `$_class_prefix`

8.98.4.4 `$_db` [private]

8.98.4.5 `$_home` [private]

8.98.4.6 `$_instance` [private]

8.98.4.7 `$_module_app`

8.98.4.8 `$_tbl_options` [private]

8.98.4.9 `$_title` [private]

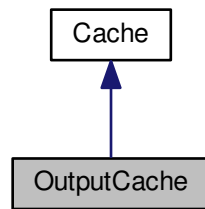
The documentation for this class was generated from the following file:

- lib/classes/[class.Options.php](#)

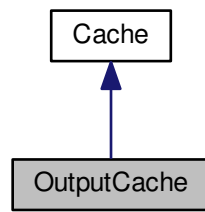
8.99 OutputCache Class Reference

Memorizza gli output (text, html, xml) scrivendo su file.

Inheritance diagram for OutputCache:



Collaboration diagram for OutputCache:



Public Member Functions

- `__construct` (&\$buffer, \$enabled=TRUE)
Costruttore.
- `start` (\$grp, \$id, \$tc)
Inizia il processo di cache. Se il contenuto è in cache lo accoda a \$buffer e ritorna FALSE, altrimenti ritorna TRUE (e si entra nell'if statement)
- `stop` (\$data)
Finalizza il processo di cache. Se la cache è abilitata salva il contenuto in un file, appende il contenuto alla variabile \$buffer.

Additional Inherited Members

8.99.1 Detailed Description

Memorizza gli output (text, html, xml) scrivendo su file.

Esempio di utilizzo

```
$GINO = "previous text-";  
$cache = new outputCache($GINO);  
if($cache->start("group_name", "id", 3600)) {
```

```

    $buffer = "some content-";

    $cache->stop($buffer);
}
$GINO .= "next text";

```

→ result: \$GINO = "previous text-somec content-next text";

se il contenuto è in cache l'if statement viene saltato ed il contenuto è concatenato alla variabile \$buffer (attraverso il metodo self::stop) se il contenuto non è in cache viene eseguito il codice interno all'if statement, il contenuto viene preparato e salvato in cache e quindi aggiunto alla variabile \$buffer (attraverso il metodo self::stop)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.99.2 Constructor & Destructor Documentation

8.99.2.1 __construct (& \$buffer, \$enabled = TRUE)

Costruttore.

Parameters

string	<i>\$buffer</i>	contenuto al quale appendere il contenuto da mettere/prelevare dalla cache
bool	<i>\$enabled</i>	abilitazione cache, default TRUE

Returns

istanza di [Gino.OutputCache](#)

8.99.3 Member Function Documentation

8.99.3.1 start (\$grp, \$id, \$tc)

Inizia il processo di cache Se il contenuto è in cache lo accoda a \$buffer e ritorna FALSE, altrimenti ritorna TRUE (e si entra nell'if statement)

Parameters

string	<i>\$grp</i>	gruppo
string	<i>\$id</i>	

Returns

TRUE se il file è in cache, FALSE altrimenti

8.99.3.2 stop (\$data)

Finalizza il processo di cache Se la cache è abilitata salva il contenuto in un file, appende il contenuto alla variabile \$buffer.

Returns

void

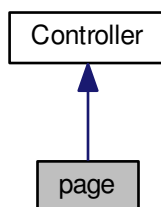
The documentation for this class was generated from the following file:

- lib/classes/[class.Cache.php](#)

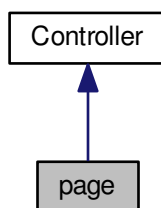
8.100 page Class Reference

Classe di tipo [Gino.Controller](#) per la gestione delle pagine.

Inheritance diagram for page:



Collaboration diagram for page:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [commentNotification](#) ()
Getter dell'opzione comment_notification.
- [getBasePath](#) (\$path='abs')
Percorso base alla directory dei contenuti.
- [getAddPath](#) (\$id)

- Percorso della directory di una pagina a partire dal percorso base.*
- [accessPage](#) (\$options=array())
Accesso alla visualizzazione delle pagine.
- [showcase](#) ()
Front end vetrina pagine più lette.
- [box](#) (\$id=null)
Front end pagina inserita nel template del layout.
- [view](#) (\Gino\Http\Request \$request)
Front end pagina.
- [actionComment](#) (\Gino\Http\Request \$request)
Processa il form di inserimento commento.
- [relatedContentsList](#) (\$page_entry)
Lista di contenuti correlati per tag.
- [managePage](#) (\Gino\Http\Request \$request)
Interfaccia di amministrazione del modulo.
- [checkSlug](#) (\Gino\Http\Request \$request)
Controlla l'unicità del valore dello slug.
- [searchSite](#) ()
Metodo per la definizione di parametri da utilizzare per il modulo "Ricerca nel sito".
- [searchSiteResult](#) (\$results)
Definisce la presentazione del singolo item trovato a seguito di ricerca (modulo "Ricerca nel sito")
- [systemNewsletterList](#) ()
Adattatore per la classe newsletter.
- [systemNewsletterRender](#) (\$id)
Contenuto di una pagina quanto inserita in una newsletter.
- [feedRSS](#) (\Gino\Http\Request \$request)
Genera un feed RSS standard che presenta gli ultimi 50 post pubblicati.

Static Public Member Functions

- static [getClassElements](#) ()
Restituisce alcune proprietà della classe.
- static [outputFunctions](#) ()
Definizione dei metodi pubblici che forniscono un output per il front-end.
- static [explanationTemplate](#) ()
Linee guida sulla costruzione di un template.
- static [getUrlPage](#) (\$id, \$box=FALSE)
Indirizzo pagina.

Private Member Functions

- [formComment](#) (\$entry)
Form di inserimento commento.
- [parseTemplate](#) (\$entry, \$tpl, \$matches)
Parserizzazione dei template inseriti da opzioni.
- [replaceTplVar](#) (\$property, \$filter, \$obj)
Replace delle variabili del template.
- [manageEntry](#) (\$request)
Interfaccia di amministrazione delle pagine.
- [manageCtg](#) ()
Interfaccia di amministrazione delle categorie.
- [manageComment](#) ()
Interfaccia di amministrazione dei commenti.

Private Attributes

- [\\$_showcase_title](#)
Titolo vista vetrina.
- [\\$_showcase_number](#)
Numero post in vetrina.
- [\\$_showcase_tpl_code](#)
Template elemento in vista vetrina.
- [\\$_showcase_auto_start](#)
Avvio automatico animazione vetrina.
- [\\$_showcase_auto_interval](#)
Intervallo animazione automatica vetrina.
- [\\$_entry_tpl_code](#)
Template pagina.
- [\\$_box_tpl_code](#)
Template pagina inserita nel template.
- [\\$_comment_moderation](#)
Moderazione dei commenti.
- [\\$_comment_notification](#)
Notifica commenti.
- [\\$_newsletter_entries_number](#)
Numero di post proposti per la newsletter.
- [\\$_newsletter_tpl_code](#)
Template elemento quando inserito in newsletter.
- [\\$_tbl_opt](#)
Tabella di opzioni.

Additional Inherited Members

8.100.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione delle pagine.

CARATTERISTICHE

Modulo di gestione pagine, con feed RSS e predisposizione contenuti per ricerca nel sito e newsletter.
Per ogni singola pagina è possibile abilitare l'inserimento dei commenti (il form include un controllo captcha).

PERMESSI

- visualizzazione pagine private (*can_view_private*)
- redazione (*can_edit*)
- pubblicazione (*can_publish*)
- amministrazione modulo (*can_admin*)

Il metodo utilizzato per verificare le condizioni di accesso alle pagine è [accessPage\(\)](#). Questo metodo è pubblico e può essere utilizzato anche dalle altre classi e applicazioni, come ad esempio la classe [Gino.App.Menu.menu](#).

POLITICHE DI VISUALIZZAZIONE

Alla visualizzazione di una pagina concorrono i seguenti elementi:

- pubblicazione della pagina (campo *published*)
- visualizzazione a utenti con permesso 'visualizza pagine private' (campo *private*)
- visualizzazione a utenti specifici (campo *users*)

Per un utente non autenticato una pagina è quindi visibile nel caso in cui:

- è pubblicata (*published*=1)
- non è associata a specifici utenti (*users*="")
- non è privata (*private*=0)

OPZIONI CONFIGURABILI

- titolo vetrina pagine più lette + numero + template singolo elemento
- template pagina
- moderazione commenti
- notifica commenti
- template per newsletter

OUTPUT

- vetrina pagine (più lette)
- pagina
- feed RSS

##DIRECTORY DEI CONTENUTI

I contenuti non testuali delle pagine sono strutturati in directory secondo lo schema:

- contents/
- page/
- page/
- [page_id]/

TEMPLATE

Quando una pagina viene richiamata da URL viene chiamato il metodo [view\(\)](#).

In questo caso il template di default è quello che corrisponde al campo *entry_tpl_code* della tabella delle opzioni ('[Template](#) vista dettaglio pagina' nelle opzioni vista pagina).

Questo template può essere sovrascritto compilando il campo "Template pagina intera" () nel form della pagina.

Quando una pagina viene richiamata nel template del layout viene chiamato il metodo [box\(\)](#).

In questo caso il template di default è quello che corrisponde al campo *box_tpl_code* della tabella delle opzioni ('[Template](#) vista dettaglio pagina' nelle opzioni vista pagina inserita nel template).

Questo template può essere sovrascritto compilando il campo "Template box" () nel form della pagina.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.100.2 Constructor & Destructor Documentation

8.100.2.1 __construct ()

Costruttore.

Returns

istanza di [Gino.App.Page.page](#)

8.100.3 Member Function Documentation

8.100.3.1 accessPage (*\$options* = array())

Accesso alla visualizzazione delle pagine.

Parameters

<i>array</i>	options array associativo di opzioni <ul style="list-style-type: none"> • page_obj (object): oggetto della pagina • page_id (integer): valore ID della pagina
--------------	---

Returns

accesso, bool

8.100.3.2 actionComment (\Gino\Http\Request *\$request*)

Processa il form di inserimento commento.

Parameters

--	--

8.100.3.3 box (*\$id* = null)

Front end pagina inserita nel template del layout.

Il template di default impostato nelle opzioni della libreria può essere sovrascritto da un template personalizzato (campo *box_tpl_code*).

Parametri GET:

- id (integer), valore ID della pagina

See also

self::accessPage()
self::parseTemplate()

Parameters

int	<i>\$id</i>	valore ID della pagina
-----	-------------	------------------------

Returns

html

8.100.3.4 checkSlug (\Gino\Http\Request \$request)

Controlla l'unicità del valore dello slug.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#) (disponibile / non disponibile)

8.100.3.5 commentNotification ()

Getter dell'opzione comment_notification.

Returns

proprietà comment_notification

8.100.3.6 static explanationTemplate () [static]

Linee guida sulla costruzione di un template.

Returns

html

8.100.3.7 feedRSS (\Gino\Http\Request \$request)

Genera un feed RSS standard che presenta gli ultimi 50 post pubblicati.

Parameters

--	--

8.100.3.8 formComment (\$entry) [private]

[Form](#) di inserimento commento.

Returns

html, form

8.100.3.9 `getAddPath ($id)`

Percorso della directory di una pagina a partire dal percorso base.

Parameters

integer	<i>\$id</i>	valore id della pagina
---------	-------------	------------------------

Returns

percorso

8.100.3.10 `getBasePath ($path = 'abs')`

Percorso base alla directory dei contenuti.

Parameters

string	<i>\$path</i>	tipo di percorso (default abs) <ul style="list-style-type: none">• abs, assoluto• rel, relativo
--------	---------------	--

Returns

percorso

8.100.3.11 `static getClassElements ()` [static]

Restituisce alcune proprietà della classe.

Returns

lista delle proprietà utilizzate per la creazione di istanze di tipo pagina

8.100.3.12 `static getUrlPage ($id, $box = FALSE)` [static]

Indirizzo pagina.

Viene utilizzato per le operazione interne a gino (ad es. menu e layout)

Parameters

integer	<i>\$id</i>	valore ID della pagina
boolean	<i>\$box</i>	se TRUE restituisce l'indirizzo per una pagina inserita nel template del layout

Returns

url

8.100.3.13 `manageComment ()` [private]

Interfaccia di amministrazione dei commenti.

Returns

[Gino.Http.Redirect](#) oppure html, interfaccia di back office dei commenti,

8.100.3.14 `manageCtg () [private]`

Interfaccia di amministrazione delle categorie.

Returns

[Gino.Http.Redirect](#) oppure html, interfaccia di back office delle categorie,

8.100.3.15 `manageEntry ($request) [private]`

Interfaccia di amministrazione delle pagine.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Redirect](#) oppure html, interfaccia di back office delle pagine,

8.100.3.16 `managePage (\Gino\Http\Request $request)`

Interfaccia di amministrazione del modulo.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Response](#) interfaccia di back office

8.100.3.17 `static outputFunctions () [static]`

Definizione dei metodi pubblici che forniscono un output per il front-end.

Questo metodo viene letto dal motore di generazione dei layout (prende i metodi non presenti nel file ini) e dal motore di generazione di voci di menu (presenti nel file ini) per presentare una lista di output associati all'istanza di classe.

Returns

array, METHOD_NAME => array('label' => (string) [label], 'permissions' => (array) [permissions list in the format classname.code_perm])

8.100.3.18 `parseTemplate ($entry, $tpl, $matches) [private]`

Parserizzazione dei template inseriti da opzioni.

Parameters

\Gino\↔ App\↔ Page\↔ PageEntry	<i>\$entry</i>	istanza di Gino.App.Page.PageEntry
string	<i>\$tpl</i>	codice del template
array	<i>\$matches</i>	matches delle variabili da sostituire

Returns

template parserizzato

8.100.3.19 relatedContentsList (*\$page_entry*)

Lista di contenuti correlati per tag.

Parameters

\Gino\↔ App\↔ Page\↔ PageEntry	<i>\$page_entry</i>	istanza di Gino.App.Page.PageEntry
---	---------------------	--

Returns

lista contenuti correlati

8.100.3.20 replaceTplVar (*\$property*, *\$filter*, *\$obj*) [private]

Replace delle variabili del template.

Parameters

string	<i>\$property</i>	proprietà da sostituire
string	<i>\$filter</i>	filtro applicato
\Gino\↔ App\↔ Page\↔ PageEntry	<i>\$obj</i>	istanza di Gino.App.Page.PageEntry

Returns

replace del parametro proprietà

8.100.3.21 searchSite ()

Metodo per la definizione di parametri da utilizzare per il modulo "Ricerca nel sito".

Il modulo "Ricerca nel sito" di [Gino](#) base chiama questo metodo per ottenere informazioni riguardo alla tabella, campi, pesi etc... per effettuare la ricerca dei contenuti.

Returns

array[string]mixed array associativo contenente i parametri per la ricerca

8.100.3.22 searchSiteResult (*\$results*)

Definisce la presentazione del singolo item trovato a seguito di ricerca (modulo "Ricerca nel sito")

See also

[Gino.App.SearchSite](#)

[Gino.Search](#)

Parameters

array	<i>\$results</i>	array associativo contenente i risultati della ricerca
-------	------------------	--

Returns

html, presentazione item tra i risultati della ricerca

8.100.3.23 showcase ()

Front end vetrina pagine più lette.

Returns

html

8.100.3.24 systemNewsletterList ()

Adattatore per la classe newsletter.

Returns

array di elementi esportabili nella newsletter

8.100.3.25 systemNewsletterRender (*\$id*)

Contenuto di una pagina quanto inserita in una newsletter.

Parameters

int	<i>\$id</i>	identificativo della pagina
-----	-------------	-----------------------------

Returns

contenuto pagina

8.100.3.26 view (\Gino\Http\Request *\$request*)

Front end pagina.

Il template di default impostato nelle opzioni della libreria può essere sovrascritto da un template personalizzato (campo *tpl_code*).

See also

`self::accessPage()`

`self::parseTemplate()`

`self::formComment()`

[Gino.App.Page.PageComment::getTree\(\)](#)

Parameters

--	--

8.100.4 Field Documentation

8.100.4.1 `$_box_tpl_code` [private]

[Template](#) pagina inserita nel template.

8.100.4.2 `$_comment_moderation` [private]

Moderazione dei commenti.

8.100.4.3 `$_comment_notification` [private]

Notifica commenti.

8.100.4.4 `$_entry_tpl_code` [private]

[Template](#) pagina.

8.100.4.5 `$_newsletter_entries_number` [private]

Numero di post proposti per la newsletter.

8.100.4.6 `$_newsletter_tpl_code` [private]

[Template](#) elemento quando inserito in newsletter.

8.100.4.7 `$_showcase_auto_interval` [private]

Intervallo animazione automatica vetrina.

8.100.4.8 `$_showcase_auto_start` [private]

Avvio automatico animazione vetrina.

8.100.4.9 `$_showcase_number` [private]

Numero post in vetrina.

8.100.4.10 `$_showcase_title` [private]

Titolo vista vetrina.

8.100.4.11 `$_showcase_tpl_code` [private]

[Template](#) elemento in vista vetrina.

8.100.4.12 `$_tbl_opt` [private]

Tabella di opzioni.

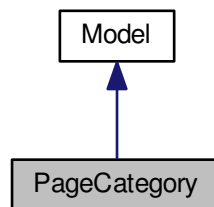
The documentation for this class was generated from the following file:

- [app/page/class_page.php](#)

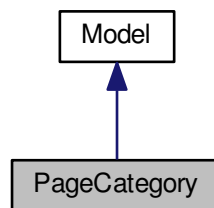
8.101 PageCategory Class Reference

Classe tipo [Gino.Model](#) che rappresenta una categoria di pagine.

Inheritance diagram for PageCategory:



Collaboration diagram for PageCategory:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.

Static Public Member Functions

- static [columns](#) ()

Struttura dei campi della tabella di un modello.

Static Public Attributes

- static `$table` = "page_category"
- static `$columns`

Additional Inherited Members

8.101.1 Detailed Description

Classe tipo `Gino.Model` che rappresenta una categoria di pagine.

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.101.2 Constructor & Destructor Documentation

8.101.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<code>\$id</code>	valore ID del record
---------	-------------------	----------------------

Returns

istanza di `Gino.App.Page.PageCategory`

8.101.3 Member Function Documentation

8.101.3.1 `__toString ()`

Rappresentazione a stringa dell'oggetto.

Returns

nome categoria

8.101.3.2 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.101.4 Field Documentation

8.101.4.1 `$columns` `[static]`

8.101.4.2 `$table = "page_category"` `[static]`

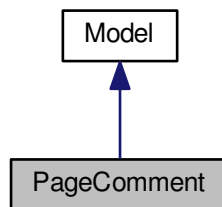
The documentation for this class was generated from the following file:

- [app/page/class.PageCategory.php](#)

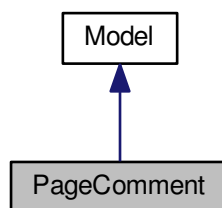
8.102 PageComment Class Reference

Classe tipo [Gino.Model](#) che rappresenta un commento ad una pagina.

Inheritance diagram for PageComment:



Collaboration diagram for PageComment:



Public Member Functions

- [__construct](#) (`$id`)
Costruttore.
- [__toString](#) ()
Rappresentazione testuale dell'oggetto.
- [save](#) ()
Salva i cambiamenti fatti sull'oggetto modificando o inserendo un nuovo record su DB.

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.
- static [getCountFromEntry](#) (\$entry_id)
Numero totale di commenti per la pagina.
- static [getTree](#) (\$entry_id, \$reply=0, \$tree=array(), \$recursion=0)
Albero dei commenti ad una pagina.
- static [deleteFromEntry](#) (\$entry_id)
Eliminazione commenti legati ad una pagina.

Static Public Attributes

- static [\\$table](#) = 'page_comment'
- static [\\$columns](#)

Static Protected Member Functions

- static [properties](#) (\$model)

Private Member Functions

- [notifyComment](#) ()
Notifica gli utenti riguardo l'aggiunta di un commento.

Additional Inherited Members

8.102.1 Detailed Description

Classe tipo [Gino.Model](#) che rappresenta un commento ad una pagina.

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.102.2 Constructor & Destructor Documentation

8.102.2.1 __construct (\$id)

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.App.Page.PageComment](#)

8.102.3 Member Function Documentation

8.102.3.1 `__toString ()`

Rappresentazione testuale dell'oggetto.

Returns

data, autore

8.102.3.2 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.102.3.3 `static deleteFromEntry ($entry_id) [static]`

Eliminazione commenti legati ad una pagina.

Parameters

int	<i>\$entry_id</i>	identificativo della pagina
-----	-------------------	-----------------------------

Returns

risultato dell'operazione, bool

8.102.3.4 `static getCountFromEntry ($entry_id) [static]`

Numero totale di commenti per la pagina.

Parameters

integer	<i>\$entry_id</i>	identificativo della pagina
---------	-------------------	-----------------------------

Returns

numero di commenti

8.102.3.5 `static getTree ($entry_id, $reply = 0, $tree = array(), $recursion = 0) [static]`

Albero dei commenti ad una pagina.

Parameters

integer	<i>\$entry_id</i>	identificativo della pagina
integer	<i>\$reply</i>	identificativo del commento al quale risponde

Returns

array di array associativi in ordine ad albero. id=>id commento, recursion=>indentazione

8.102.3.6 notifyComment () [private]

Notifica gli utenti riguardo l'aggiunta di un commento.

Returns

TRUE

8.102.3.7 static properties (\$model) [static],[protected]

8.102.3.8 save ()

Salva i cambiamenti fatti sull'oggetto modificando o inserendo un nuovo record su DB.

Invia email di notifica dell'avvenuta pubblicazione di un commento all'autore del post se l'opzione `comment_notification` è attiva, ed agli utenti che commentando hanno scelto di essere notificati nel caso fossero stati postati altri commenti

See also

[Gino.Model::save\(\)](#)

Returns

boolean

8.102.4 Field Documentation

8.102.4.1 \$columns [static]

8.102.4.2 \$table = 'page_comment' [static]

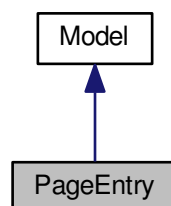
The documentation for this class was generated from the following file:

- [app/page/class.PageComment.php](#)

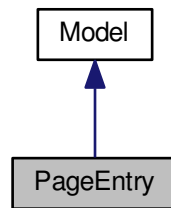
8.103 PageEntry Class Reference

Classe tipo [Gino.Model](#) che rappresenta una pagina.

Inheritance diagram for PageEntry:



Collaboration diagram for PageEntry:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [getUrl](#) ()
Url relativo pagina.
- [save](#) (\$options=array())
- [delete](#) ()
- [imgPath](#) ()
Path relativo dell'immagine associata.

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.
- static [getFromSlug](#) (\$slug, \$controller=null)
Restituisce l'istanza [Gino.App.Page.PageEntry](#) a partire dallo slug/ID fornito.
- static [get](#) (\$options=null)
Restituisce oggetti di tipo [Gino.App.Page.PageEntry](#).
- static [getCount](#) (\$options=null)
Restituisce il numero di oggetti [Gino.App.Page.PageEntry](#) selezionati.

Static Public Attributes

- static [\\$table](#) = 'page_entry'
- static [\\$columns](#)

Static Protected Member Functions

- static [properties](#) (\$model, \$controller)

Static Protected Attributes

- static [\\$_extension_img](#) = array('jpg', 'jpeg', 'png')

Private Member Functions

- [deleteDir](#) ()
Elimina la directory della pagina.

Static Private Member Functions

- static [accessWhere](#) (\$access_user, \$access_private)
Definisce le condizioni di accesso a una pagina integrando le condizioni del WHERE.

Additional Inherited Members

8.103.1 Detailed Description

Classe tipo [Gino.Model](#) che rappresenta una pagina.

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.103.2 Constructor & Destructor Documentation

8.103.2.1 __construct (\$id)

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
object	<i>\$instance</i>	istanza del controller

8.103.3 Member Function Documentation

8.103.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

titolo

8.103.3.2 static accessWhere (\$access_user, \$access_private) [static],[private]

Definisce le condizioni di accesso a una pagina integrando le condizioni del WHERE.

Parameters

integer	<i>\$access_user</i>	valore ID dell'utente
boolean	<i>\$access_private</i>	indica se l'utente appartiene al gruppo "utenti pagine private"

Returns

where clause

8.103.3.3 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.103.3.4 `delete ()`

See also

[Gino.Model::delete\(\)](#)

8.103.3.5 `deleteDir () [private]`

Elimina la directory della pagina.

Exceptions

--	--

8.103.3.6 `static get ($options = null) [static]`

Restituisce oggetti di tipo Gino.App.Page.PageEntry.

See also

[accessWhere\(\)](#)

Parameters

object	<i>\$controller</i>	istanza del controller
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • published (boolean) • tag (integer) • category (integer) • order (string) • limit (string) • access_user (integer): valore ID dell'utente in sessione (per l'accesso limitato a specifici utenti) • access_private (boolean): identifica se l'utente in sessione appartiene al gruppo che può accedere alle pagine private

Returns

array di istanze di tipo [Gino.App.Page.PageEntry](#)

8.103.3.7 static getCount (*\$options* = null) [static]

Restituisce il numero di oggetti [Gino.App.Page.PageEntry](#) selezionati.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• published (boolean)• tag (integer)• category (integer)• access_user (integer): valore ID dell'utente in sessione (per l'accesso limitato a specifici utenti)• access_private (boolean): identifica se l'utente in sessione appartiene al gruppo che può accedere alle pagine private
-------	------------------	--

Returns

numero di pagine

8.103.3.8 static getFromSlug (*\$slug*, *\$controller* = null) [static]

Restituisce l'istanza [Gino.App.Page.PageEntry](#) a partire dallo slug/ID fornito.

Parameters

mixed	<i>\$slug</i>	lo slug oppure il valore ID della pagina
null	<i>\$controller</i>	per compatibilità con il metodo Gino.Model::getFromSlug

Returns

istanza di [Gino.App.Page.PageEntry](#)

8.103.3.9 getUrl ()

Url relativo pagina.

Returns

url

8.103.3.10 imgPath ()

Path relativo dell'immagine associata.

Returns

path relativo dell'immagine

8.103.3.11 `static properties ($model, $controller)` `[static]`, `[protected]`

See also

[Gino.Model::properties\(\)](#)

8.103.3.12 `save ($options = array())`

See also

[Gino.Model::save\(\)](#) Sovrascrive il metodo di [Gino.Model](#) per salvare l'autore della pagina

8.103.4 Field Documentation

8.103.4.1 `$_extension_img = array('jpg', 'jpeg', 'png')` `[static]`, `[protected]`

8.103.4.2 `$columns` `[static]`

8.103.4.3 `$table = 'page_entry'` `[static]`

The documentation for this class was generated from the following file:

- [app/page/class.PageEntry.php](#)

8.104 Paginator Class Reference

Gestisce la paginazione di elementi dati il numero di elementi totali ed il numero di elementi per pagina, ricava i limiti per creare il sottoinsieme di elementi da mostrare e gestisce la navigazione tra le pagine.

Public Member Functions

- [__construct](#) (\$items_number, \$items_for_page, array \$kwargs=array())
Costruttore.
- [getCurrentPage](#) ()
- [limit](#) ()
Limiti items selezionati, 1 based.
- [limitQuery](#) ()
LIMIT CLAUSE, 0 based.
- [summary](#) ()
Riassunto elementi pagina corrente.
- [navigator](#) ()
Controllo per la navigazione delle pagine.
- [pages](#) ()
Ricava le pagine da mostrare nella navigazione e le mette in un array inserisce anche i 3 punti '...' quando due pagine non sono consecutive.
- [pagination](#) (\$options=null)
Codice html completo della paginazione Include la navigazione ed il sommario.
- [goToPage](#) (\$opt=null)
Interfaccia di invio a una pagina specifica.

Data Fields

- [\\$_items_for_page](#)
- [\\$_pages_number](#)
- [\\$_interval](#)
- [\\$_current_page](#)

Private Member Functions

- [setCurrentPage\(\)](#)
Imposta la pagina corrente.
- [urlPage\(\\$p\)](#)
Url che porta alla pagina data.

Private Attributes

- [\\$_items_number](#)

8.104.1 Detailed Description

Gestisce la paginazione di elementi dati il numero di elementi totali ed il numero di elementi per pagina, ricava i limiti per creare il sottoinsieme di elementi da mostrare e gestisce la navigazione tra le pagine.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.104.2 Constructor & Destructor Documentation

8.104.2.1 `__construct($items_number, $items_for_page, array $kwargs = array())`

Costruttore.

Parameters

int	<i>\$items_number</i>	numero totale di item
int	<i>\$items_for_page</i>	numero di items per pagina
array	<i>\$kwargs</i>	array associativo <ul style="list-style-type: none"> • interval: int, default 2. Numero di pagine da mostrare nella navigazione nell'intorno di quella corrente

Returns

istanza di [Gino.Paginator](#)

8.104.3 Member Function Documentation

8.104.3.1 `getCurrentPage ()`

8.104.3.2 `goToPage ($opt = null)`

Interfaccia di invio a una pagina specifica.

Parameters

string	<i>\$url</i>	indirizzo da richiamare
array	<i>\$opt</i>	array associativo di opzioni <ul style="list-style-type: none"> • url (string): indirizzo della pagina • label (string): testo da anteporre all'input (se la chiave non esiste viene mostrato un testo di default) • add_params (mixed): variabili da aggiungere all'indirizzo <ul style="list-style-type: none"> – string, es. p1=var1&p2=var2 – array, nel formato parametro=>valore

Returns

string

8.104.3.3 limit ()

Limiti items selezionati, 1 based.

Returns

array(limite inferiore, numero superiore), il limite inferiore parte da 1

8.104.3.4 limitQuery ()

LIMIT CLAUSE, 0 based.

Returns

array(limite inferiore, numero items per pagina), il limite inferiore parte da 0

8.104.3.5 navigator ()

Controllo per la navigazione delle pagine.

Returns

codice html

8.104.3.6 pages ()

Ricava le pagine da mostrare nella navigazione e le mette in un array inserisce anche i 3 punti '...' quando due pagine non sono consecutive.

Returns

array di pagine

8.104.3.7 pagination (\$options = null)

Codice html completo della paginazione Include la navigazione ed il sommario.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • summary_label (string): label del sommario • gotopage (array): parametri dell'interfaccia di rimando a una pagina specifica <ul style="list-style-type: none"> – <i>view</i> (boolean), indica se mostrare l'interfaccia – parametri del metodo goToPage()
-------	------------------	---

Returns

codice html paginazione

8.104.3.8 `setCurrentPage ()` [private]

Imposta la pagina corrente.

Returns

void

8.104.3.9 `summary ()`

Riassunto elementi pagina corrente.

Returns

codice html riassunto, es 10-20 di 100

8.104.3.10 `urlPage ($p)` [private]

Url che porta alla pagina data.

Parameters

int	<i>\$p</i>	numero pagina
-----	------------	---------------

Returns

url o null se è la pagina corrente

8.104.4 Field Documentation

8.104.4.1 `$_current_page`8.104.4.2 `$_interval`8.104.4.3 `$_items_for_page`8.104.4.4 `$_items_number` [private]

8.104.4.5 `$_pages_number`

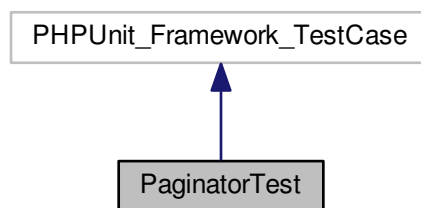
The documentation for this class was generated from the following file:

- lib/classes/[class.Paginator.php](#)

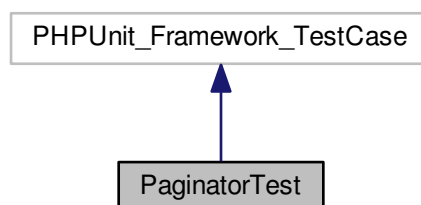
8.105 PaginatorTest Class Reference

Classe di tipo PHPUnit_Framework_TestCase per testare la classe [Gino.Paginator](#).

Inheritance diagram for PaginatorTest:



Collaboration diagram for PaginatorTest:



Public Member Functions

- [test_getCurrentPage \(\)](#)
Test metodo getCurrentPage Il metodo restituisce il numero di pagina corrente.
- [test_limit \(\)](#)
Test metodo limit Il metodo limit restituisce gli estremi di selezione a partire da 1.
- [test_limitQuery \(\)](#)
Test metodo limitQuery Il metodo limitQuery restituisce l'estremo inferiore della selezione (a partire da 0) ed il numero di elementi da selezionare Viene utilizzato per la condizione LIMIT delle query.
- [test_summary \(\)](#)
Test metodo summary Il metdo summary stampa il sommario dei risultati slezionati, es 1-10 di 100.

- [test_pages](#) ()

Test metodo *pages* Il metdo *pages* ricava le pagine da mostrare nella navigazione, inserendo i tre puntini '...' tra pagine non contigue.

8.105.1 Detailed Description

Classe di tipo PHPUnit_Framework_TestCase per testare la classe [Gino.Paginator](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.105.2 Member Function Documentation

8.105.2.1 test_getCurrentPage ()

Test metodo *getCurrentPage* Il metodo restituisce il numero di pagina corrente.

8.105.2.2 test_limit ()

Test metodo *limit* Il metodo *limit* restituisce gli estremi di selezione a partire da 1.

8.105.2.3 test_limitQuery ()

Test metodo *limitQuery* Il metodo *limitQuery* restituisce l'estremo inferiore della selezione (a partire da 0) ed il numero di elementi da selezionare Viene utilizzato per la condizione LIMIT delle query.

8.105.2.4 test_pages ()

Test metodo *pages* Il metdo *pages* ricava le pagine da mostrare nella navigazione, inserendo i tre puntini '...' tra pagine non contigue.

Di default l'intorno mostrato della pagina corrente è di 4 pagine, due per lato.

8.105.2.5 test_summary ()

Test metodo *summary* Il metdo *summary* stampa il sommario dei risultati selezionati, es 1-10 di 100.

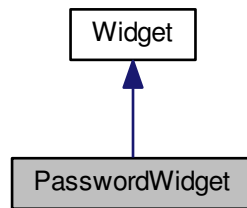
The documentation for this class was generated from the following file:

- [test/class.Paginator_test.php](#)

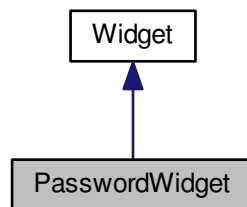
8.106 PasswordWidget Class Reference

Campi di tipo password nei form.

Inheritance diagram for PasswordWidget:



Collaboration diagram for PasswordWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.106.1 Detailed Description

Campi di tipo password nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.106.2 Member Function Documentation

8.106.2.1 printInputForm (\$form, \$options)

See also

[Gino.Widget::printInputForm\(\)](#)

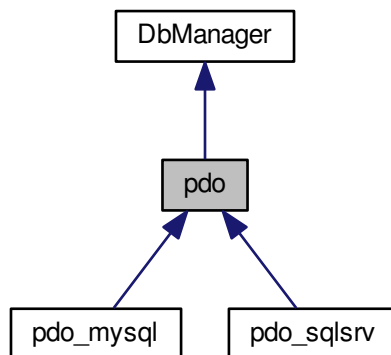
The documentation for this class was generated from the following file:

- [lib/classes/widget/class.PasswordWidget.php](#)

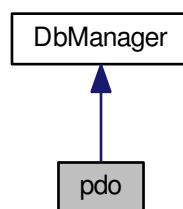
8.107 pdo Class Reference

Libreria di connessione ai database.

Inheritance diagram for pdo:



Collaboration diagram for pdo:



Public Member Functions

- [__construct](#) (\$params)
Costruttore.
- [getNumberRows](#) ()

Ritorna il numero di righe risultanti da una select query.

- [getNumberCols](#) ()

Ritorna il numero di colonne richiamate in una istruzione SELECT.

- [getAffectedRows](#) ()

Ritorna il numero di righe interessate da una istruzione INSERT, UPDATE o DELETE.

- [getInfoQuery](#) ()
- [openConnection](#) (\$opt=array())
- [closeConnection](#) ()
- [begin](#) ()
- [rollback](#) ()
- [commit](#) ()
- [multiActionQuery](#) (\$queries)
- [freeresult](#) (\$result=null)
- [getLastId](#) (\$table)
- [autoIncValue](#) (\$table)
- [getFieldFromId](#) (\$table, \$field, \$field_id, \$id, \$options=array())
- [tableexists](#) (\$table)
- [fieldInformations](#) (\$table)
- [conformFieldType](#) (\$meta)
- [limit](#) (\$range, \$offset)
- [distinct](#) (\$fields, \$options=array())
- [concat](#) (\$sequence)
- [dumpDatabase](#) (\$file)
- [getTableStructure](#) (\$table)
- [changeFieldType](#) (\$data_type, \$value)
- [getFieldsName](#) (\$table)
- [getNumRecords](#) (\$table, \$where=null, \$field='id', \$options=array())
- [query](#) (\$fields, \$tables, \$where=null, \$options=array())
- [execCustomQuery](#) (\$query, \$options=array())
- [select](#) (\$fields, \$tables, \$where=null, \$options=array())
- [insert](#) (\$fields, \$table, \$debug=false)
- [update](#) (\$fields, \$table, \$where, \$debug=false)
- [delete](#) (\$table, \$where, \$debug=false)
- [drop](#) (\$table)
- [columnHasValue](#) (\$table, \$field, \$value, \$options=array())
- [join](#) (\$table, \$condition, \$option)
- [union](#) (\$queries, \$options=array())
- [restore](#) (\$table, \$filename, \$options=array())
- [dump](#) (\$table, \$path_to_file, \$options=array())
- [escapeString](#) (\$string)

Data Fields

- [\\$_db_host](#)
- [\\$_db_name](#)
- [\\$_db_user](#)
- [\\$_db_password](#)
- [\\$_db_charset](#)

Protected Member Functions

- [checkRowsFromSelect](#) (\$result)
Verifica se una istruzione SELECT ha avuto esito positivo.
- [queryResults](#) (\$query, \$options=array())
Esegue la query e ne calcola i tempi di esecuzione.
- [fetch](#) (\$results=null, \$options=array())
Prende e rende fruibili i risultati della query.
- [setDSN](#) ()
Imposta la stringa di connessione al database.
- [setAttribute](#) ()
Attributi da associare alla connessione.
- [setCharacterSet](#) ()
Imposta il character set del database.
- [buildQuery](#) (\$options=array())
Costruisce la query (personalizzata per ogni driver)
- [setFieldName](#) (\$field)
Imposta il nome del campo in una query.
- [setFieldValue](#) (\$value)
Imposta il valore del campo in una query.
- [arrayFieldsToString](#) (\$a_fields)
Formatta l'elenco dei campi in una istruzione insert.
- [SQLForFieldInformations](#) (\$table)
SQL code specifico del driver per il metodo [fieldInformations\(\)](#)
- [SQLForGetFieldsName](#) (\$table)
SQL code specifico del driver per il metodo [getFieldsName\(\)](#)
- [SQLForDump](#) (\$table, \$path_to_file, \$delimiter, \$enclosed, \$where)
SQL code specifico del driver per il metodo [dump\(\)](#)

Static Protected Member Functions

- static [getCharset](#) ()
Elenco dei character set accettati.

Protected Attributes

- [\\$_dbms](#)
- [\\$_numberrows](#)
- [\\$_affected](#)
- [\\$_numbercols](#)
- [\\$_connection](#)
- [\\$_pdo](#)
- [\\$_result](#)
- [\\$_show_stats](#)
- [\\$_info_queries](#)
- [\\$_cnt](#)
- [\\$_time_queries](#)
- [\\$_query_cache](#)
- [\\$_query_cache_time](#)
- [\\$_cache](#)

Private Member Functions

- [setParamsCache \(\)](#)
Parametri inerenti la cache delle query presenti nelle Impostazioni di sistema.
- [setNumberRows \(\\$numberresults\)](#)
Imposta il numero di righe risultanti da una select query.
- [setNumberCols \(\\$number\)](#)
Imposta il numero di colonne richiamate in una istruzione SELECT.
- [setConnection \(\\$connection\)](#)
- [setAffectedRows \(\\$number\)](#)
Imposta il numero di righe interessate da una istruzione INSERT, UPDATE o DELETE.
- [getError \(\)](#)
Recupera il tipo di errore.
- [setFetchMode \(\\$results, \\$options=array\(\)\)](#)
Imposta la modalità di recupero dei risultati della query.
- [getAttribute \(\\$connection\)](#)
Stampa il valore degli attributi associati alla connessione.
- [setWhereItems \(\\$where\)](#)
Imposta la condizione where e i valori per le query con variabili parametrizzate.

8.107.1 Detailed Description

Libreria di connessione ai database.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

DRIVER

L'interfaccia PDO attualmente è implementata da 12 driver (<http://php.net/manual/en/pdo.drivers.php>) che supportano diversi tipi di database.

Ogni driver deve essere gestito da una classe che estende la classe [Gino.Plugin.pdo](#) e che definisce alcune specificità di un database.

Il nome della classe del driver deve essere "pdo_[valore della costante DBMS]" mentre il nome del file "plugin.[nome della classe].php".

CACHE QUERY

La proprietà `self::$_query_cache` indica se è stata abilita la cache delle query.

Le query che vengono salvate in cache sono quelle che passano dal metodo [select\(\)](#) ed [execCustomQuery\(\)](#), e non riguardano quindi le query di struttura, quali quelle presenti nei metodi:

- [fieldInformations\(\)](#)
- [getTableStructure\(\)](#)
- [getFieldsName\(\)](#)

Qualora non si desideri caricare in cache una determinata query è sufficiente passare l'opzione *cache=false* ai metodi [select\(\)](#) e [execCustomQuery\(\)](#).

La cache delle query viene svuotata ogni volta che viene eseguita una query di tipo *action* (

See also

[queryResults\(\)](#).

INFORMAZIONI SULLE QUERY

La proprietà *self::\$_show_stats* attiva la raccolta di informazioni sulle prestazioni delle chiamate al database.

Le query di tipo select alimentano i contatori *self::\$_cnt* e *self::\$_time_queries*.

8.107.2 Constructor & Destructor Documentation

8.107.2.1 __construct (\$params)

Costruttore.

Parameters

array	<i>\$params</i>	parametri di connessione al database <ul style="list-style-type: none"> • dbms (string): nome del database management system • host (string): nome del server • db_name (string): nome del database • user (string): utente che si connette • password (string): password dell'utente che si connette • charset (string): encoding • connect (boolean): attiva la connessione
-------	-----------------	---

Returns

void

Implements [DbManager](#).

8.107.3 Member Function Documentation

8.107.3.1 arrayFieldsToString (\$a_fields) [protected]

Formatta l'elenco dei campi in una istruzione insert.

Parameters

array	<i>\$a_fields</i>	elenco dei campi
-------	-------------------	------------------

Returns

string

8.107.3.2 autoIncValue (*\$table*)

See also

[Gino.DbManager::autoIncValue\(\)](#)
driver

Implements [DbManager](#).

8.107.3.3 begin ()

See also

[Gino.DbManager::begin\(\)](#)

Implements [DbManager](#).

8.107.3.4 buildQuery (*\$options* = array()) [protected]

Costruisce la query (personalizzata per ogni driver)

See also

driver

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• statement (string): tipo di istruzione sql (default <i>select</i>)• fields (string)• tables (string)• where (string)• group_by (string)• order (string)• limit (array string)• debug (boolean)
-------	------------------	--

Returns

string

8.107.3.5 changeFieldType (*\$data_type*, *\$value*)

See also

[Gino.DbManager::changeFieldType\(\)](#)

Implements [DbManager](#).

8.107.3.6 checkRowsFromSelect (*\$result*) [protected]

Verifica se una istruzione SELECT ha avuto esito positivo.

Parameters

object	<i>\$result</i>	PDOStatement object
--------	-----------------	---------------------

Returns

integer (0 negative)

8.107.3.7 `closeConnection ()`

See also

[Gino.DbManager::closeConnection\(\)](#)

Implements [DbManager](#).

8.107.3.8 `columnHasValue ($table, $field, $value, $options = array())`

See also

[Gino.DbManager::columnHasValue\(\)](#)

Implements [DbManager](#).

8.107.3.9 `commit ()`

See also

[Gino.DbManager::commit\(\)](#)

Implements [DbManager](#).

8.107.3.10 `concat ($sequence)`

See also

[Gino.DbManager::concat\(\)](#)
driver

Implements [DbManager](#).

8.107.3.11 `conformFieldType ($meta)`

See also

[Gino.DbManager::conformFieldType\(\)](#)
driver

Implements [DbManager](#).

8.107.3.12 `delete ($table, $where, $debug = false)`

See also

[Gino.DbManager::delete\(\)](#)

Implements [DbManager](#).

8.107.3.13 `distinct ($fields, $options = array())`

See also

[Gino.DbManager::distinct\(\)](#)
driver

Implements [DbManager](#).

8.107.3.14 `drop ($table)`

See also

[Gino.DbManager::drop\(\)](#)

Implements [DbManager](#).

8.107.3.15 `dump ($table, $path_to_file, $options = array())`

See also

[Gino.DbManager::dump\(\)](#)
[SQLForDump\(\)](#)

Implements [DbManager](#).

8.107.3.16 `dumpDatabase ($file)`

See also

[Gino.DbManager::dumpDatabase\(\)](#)
driver

Implements [DbManager](#).

8.107.3.17 `escapeString ($string)`

See also

[Gino.DbManager::escapeString\(\)](#)

Implements [DbManager](#).

8.107.3.18 `execCustomQuery ($query, $options = array())`

See also

[Gino.DbManager::execCustomQuery\(\)](#)

Implements [DbManager](#).

8.107.3.19 `fetch ($results = null, $options = array())` [protected]

Prende e rende fruibili i risultati della query.

Parameters

object	<i>\$results</i>	oggetto PDOStatement
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • set_mode (boolean): indica se impostate il fetch mode (default true) • fetchAll (boolean): indica se utilizzare il metodo fetchAll per prendere i risultati (default false) • opzioni del metodo setFetchMode()

Returns

array (fetchAll=false) or object (fetchAll=true)

8.107.3.20 fieldInformations (*\$table*)

See also

[Gino.DbManager::fieldInformations\(\)](#)
[SQLForFieldInformations\(\)](#)

Implements [DbManager](#).

8.107.3.21 freeresult (*\$result = null*)

See also

[Gino.DbManager::freeresult\(\)](#)

Implements [DbManager](#).

8.107.3.22 getAffectedRows ()

Ritorna il numero di righe interessate da una istruzione INSERT, UPDATE o DELETE.

Returns

integer

8.107.3.23 getAttribute (*\$connection*) [private]

Stampa il valore degli attributi associati alla connessione.

Parameters

object	<i>\$connection</i>	PDO object
--------	---------------------	------------

8.107.3.24 static getCharset () [static],[protected]

Elenco dei character set accettati.

Returns

array

8.107.3.25 `getError () [private]`

Recupera il tipo di errore.

Returns

string

A prescindere dalla modalità che è impostato, c'è un codice di errore interno che viene impostato e che si può controllare utilizzando i metodi `errorCode()` e `errorInfo()` degli oggetti `POD` e `PDOStatement`.

`errorCode()` restituisce una stringa di 5 caratteri, come definito nella ANSI SQL-92.

`errorInfo()` è generalmente più utile in quanto restituisce un array che comprende un messaggio di errore in aggiunta al codice a 5 caratteri.

8.107.3.26 `getFieldFromId ($table, $field, $field_id, $id, $options = array())`**See also**

[Gino.DbManager::getFieldFromId\(\)](#)

Implements [DbManager](#).

8.107.3.27 `getFieldsName ($table)`**See also**

[Gino.DbManager::getFieldsName\(\)](#)

[SQLForGetFieldsName\(\)](#)

Implements [DbManager](#).

8.107.3.28 `getInfoQuery ()`**See also**

[Gino.DbManager::getInfoQuery\(\)](#)

Implements [DbManager](#).

8.107.3.29 `getLastId ($table)`**See also**

[Gino.DbManager::getLastId\(\)](#)

Implements [DbManager](#).

8.107.3.30 `getNumberCols ()`

Ritorna il numero di colonne richiamate in una istruzione `SELECT`.

Returns

integer

8.107.3.31 `getNumberRows ()`

Ritorna il numero di righe risultanti da una select query.

Returns

integer

8.107.3.32 `getNumRecords ($table, $where = null, $field = 'id', $options = array())`

See also

[Gino.DbManager::getNumRecords\(\)](#)

Implements [DbManager](#).

8.107.3.33 `getTableStructure ($table)`

See also

[Gino.DbManager::getTableStructure\(\)](#)
driver

Implements [DbManager](#).

8.107.3.34 `insert ($fields, $table, $debug = false)`

See also

[Gino.DbManager::insert\(\)](#)

Implements [DbManager](#).

8.107.3.35 `join ($table, $condition, $option)`

See also

[Gino.DbManager::join\(\)](#)

Implements [DbManager](#).

8.107.3.36 `limit ($range, $offset)`

See also

[Gino.DbManager::limit\(\)](#)
driver

Implements [DbManager](#).

8.107.3.37 `multiActionQuery ($queries)`

See also

[Gino.DbManager::multiActionQuery\(\)](#)
driver

Implements [DbManager](#).

8.107.3.38 `openConnection ($opt = array())`

See also

[Gino.DbManager::openConnection\(\)](#)

Parameters

array	<i>\$opt</i>	array associativo di opzioni <ul style="list-style-type: none"> • persistent (boolean): connessione persistente (default true) • get_attribute (boolean): recupera i valori degli attributi della connessione (default false)
-------	--------------	---

8.107.3.39 `query ($fields, $tables, $where = null, $options = array())`

See also

[Gino.DbManager::query\(\)](#)
[buildQuery\(\)](#)

Implements [DbManager](#).

8.107.3.40 `queryResults ($query, $options = array())` [protected]

Esegue la query e ne calcola i tempi di esecuzione.

Parameters

string	<i>\$query</i>	query da eseguire
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • statement (string): tipologia di query <ul style="list-style-type: none"> – <i>select</i> (default) – <i>action</i> • parameters (boolean): indica se la query è parametrizzata <ul style="list-style-type: none"> – <i>true</i>, prima prepara e poi esegue la query – <i>false</i> (default), esegue la query • values (array): elenco dei valori da sostituire alle variabili parametrizzate

Returns

PDOStatement object

8.107.3.41 `restore ($table, $filename, $options = array())`

See also

[Gino.DbManager::restore\(\)](#)

Implements [DbManager](#).

8.107.3.42 `rollback ()`

See also

[Gino.DbManager::rollback\(\)](#)

Implements [DbManager](#).

8.107.3.43 `select ($fields, $tables, $where = null, $options = array())`

See also

[Gino.DbManager::select\(\)](#)
[queryResults\(\)](#)
[fetch\(\)](#)

Implements [DbManager](#).

8.107.3.44 `setAffectedRows ($number)` [private]

Imposta il numero di righe interessate da una istruzione INSERT, UPDATE o DELETE.

Parameters

integer	<i>\$number</i>	
---------	-----------------	--

8.107.3.45 `setAttribute ()` [protected]

Attributi da associare alla connessione.

Returns

array

8.107.3.46 `setCharacterSet ()` [protected]

Imposta il character set del database.

Nota: non è necessario se lo si specifica nella stringa della connessione

8.107.3.47 `setConnection ($connection)` [private]**8.107.3.48** `setDSN ()` [protected]

Imposta la stringa di connessione al database.

Returns

string

8.107.3.49 `setFetchMode ($results, $options = array())` [private]

Imposta la modalità di recupero dei risultati della query.

Parameters

object	<i>\$results</i>	oggetto PDOStatement
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • mode (string): modalità di recupero dei risultati della query <ul style="list-style-type: none"> – ASSOC (default) – NUM – OBJ – CLASS • classname (string): nome della classe da richiamare (modalità CLASS) • construct_args (array): argomenti da passare al costruttore della classe da richiamare (modalità CLASS)

8.107.3.50 **setFieldName** (*\$field*) [protected]

Imposta il nome del campo in una query.

Parameters

string	<i>\$field</i>	nome del campo
--------	----------------	----------------

Returns

string

8.107.3.51 **setFieldValue** (*\$value*) [protected]

Imposta il valore del campo in una query.

Parameters

mixed	<i>\$value</i>	valore del campo
-------	----------------	------------------

Returns

string

8.107.3.52 **setNumberCols** (*\$number*) [private]

Imposta il numero di colonne richiamate in una istruzione SELECT.

Parameters

<i>integer</i>	
----------------	--

8.107.3.53 **setNumberRows** (*\$numberresults*) [private]

Imposta il numero di righe risultanti da una select query.

Parameters

<i>integer</i>	
----------------	--

8.107.3.54 **setParamsCache** () [private]

Parametri inerenti la cache delle query presenti nelle Impostazioni di sistema.

Returns

void

8.107.3.55 **setWhereItems** (*\$where*) [private]

Imposta la condizione where e i valori per le query con variabili parametrizzate.

Parameters

mixed	<i>\$where</i>	<ul style="list-style-type: none"> • <i>string</i>, condizione where non parametrizzata • <i>array</i>, condizione where con variabili parametrizzate, nel formato <pre>array(0 => (string) where condition with named parameters or ? character 1 => (array) binding parameters, array(param=>value[,...]) or array(value[,...]))</pre>
-------	----------------	--

Returns

array(where_condition[string], parameter_values[array], parameter_query[bool])

Esempi di input where per una query con variabili parametrizzate:

```
$where = array(
    "username = :username AND email = :email AND last_login > :last_login",
    array(':username'=>'test', ':email'=>$mail, ':last_login'=>time() - 3600)
)

$where = array(
    "id=? AND name=?",
    array($id, $name)
)
```

8.107.3.56 **SQLForDump** (*\$table*, *\$path_to_file*, *\$delimiter*, *\$enclosed*, *\$where*) [protected]

SQL code specifico del driver per il metodo [dump\(\)](#)

Parameters

string	<i>\$table</i>	nome della tabella
string	<i>\$path_to_file</i>	nome del file completo di percorso
string	<i>\$delimiter</i>	stringa che viene usata per separare tra loro i valori dei campi
string	<i>\$enclosed</i>	carattere utilizzato per racchiudere i valori di tipo stringa
string	<i>\$where</i>	condizioni della query

Returns

string

8.107.3.57 SQLForFieldInformations (*\$table*) [protected]

SQL code specifico del driver per il metodo [fieldInformations\(\)](#)

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

string

8.107.3.58 `SQLForGetFieldsName ($table)` `[protected]`

SQL code specifico del driver per il metodo [getFieldsName\(\)](#)

Parameters

string	<i>\$table</i>	nome della tabella
--------	----------------	--------------------

Returns

string

8.107.3.59 `tableexists ($table)`

See also

[Gino.DbManager::tableexists\(\)](#)
driver

Implements [DbManager](#).

8.107.3.60 `union ($queries, $options = array())`

See also

[Gino.DbManager::union\(\)](#)

In mssql possono essere utilizzati gli operatori:

- UNION, elimina le righe duplicate dai risultati combinati delle istruzioni SELECT
- UNION ALL, mostra i record duplicati

Implements [DbManager](#).

8.107.3.61 `update ($fields, $table, $where, $debug = false)`

See also

[Gino.DbManager::update\(\)](#)

Implements [DbManager](#).

8.107.4 Field Documentation

8.107.4.1 `$affected` [protected]

8.107.4.2 `$cache` [protected]

8.107.4.3 `$cnt` [protected]

8.107.4.4 `$connection` [protected]

8.107.4.5 `$db_charset`

8.107.4.6 `$db_host`

8.107.4.7 `$db_name`

8.107.4.8 `$db_password`

8.107.4.9 `$db_user`

8.107.4.10 `$dbms` [protected]

8.107.4.11 `$info_queries` [protected]

8.107.4.12 `$numbercols` [protected]

8.107.4.13 `$numberrows` [protected]

8.107.4.14 `$pdo` [protected]

8.107.4.15 `$query_cache` [protected]

8.107.4.16 `$query_cache_time` [protected]

8.107.4.17 `$result` [protected]

8.107.4.18 `$show_stats` [protected]

8.107.4.19 `$time_queries` [protected]

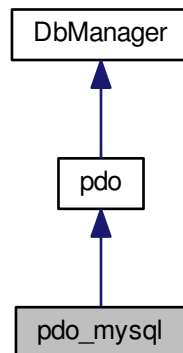
The documentation for this class was generated from the following file:

- lib/plugin/[plugin.pdo.php](#)

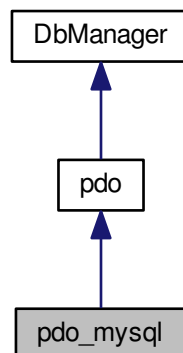
8.108 pdo_mysql Class Reference

Driver specifico per la connessione a un database MYSQL attraverso la libreria PDO.

Inheritance diagram for pdo_mysql:



Collaboration diagram for pdo_mysql:



Public Member Functions

- [multiActionQuery](#) (\$file_content)
- [autoIncValue](#) (\$table)
- [tableexists](#) (\$table)
- [limit](#) (\$range, \$offset)
- [distinct](#) (\$fields, \$options=array())
- [concat](#) (\$sequence)
- [dumpDatabase](#) (\$file)
- [getTableStructure](#) (\$table)
- [restore](#) (\$table, \$filename, \$options=array())
- [conformFieldType](#) (\$meta)

Protected Member Functions

- [setDSN](#) ()
- [setAttribute](#) ()
- [setCharacterSet](#) ()
- [buildQuery](#) (\$options=array())
- [setFieldName](#) (\$field)
- [setFieldValue](#) (\$value)
- [arrayFieldsToString](#) (\$a_fields)
- [SQLForFieldInformations](#) (\$table)
- [SQLForGetFieldsName](#) (\$table)
- [SQLForDump](#) (\$table, \$path_to_file, \$delimiter, \$enclosed, \$where)

Additional Inherited Members

8.108.1 Detailed Description

Driver specifico per la connessione a un database MYSQL attraverso la libreria PDO.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

NOTE

Metodo [dump\(\)](#)

Per poter effettuare il dump di un tabase MySQL occorre:

- assegnare il permesso FILE all'utente del database: GRANT FILE ON *.* TO 'dbuser'@'localhost';
- la directory di salvataggio deve avere i permessi read/write/execute, oppure deve avere come proprietario l'utente di sistema del database

8.108.2 Member Function Documentation

8.108.2.1 [arrayFieldsToString](#) (*\$a_fields*) [[protected](#)]

See also

[Gino.Plugin.pdo::arrayFieldsToString\(\)](#)

8.108.2.2 [autoIncValue](#) (*\$table*)

See also

[Gino.Plugin.pdo::autoIncValue\(\)](#)

Implements [DbManager](#).

8.108.2.3 buildQuery (*\$options* = array()) [protected]

See also

[Gino.Plugin.pdo::buildQuery\(\)](#)

8.108.2.4 concat (*\$sequence*)

See also

[Gino.Plugin.pdo::concat\(\)](#)

Implements [DbManager](#).

8.108.2.5 conformFieldType (*\$meta*)

See also

[Gino.Plugin.pdo::conformFieldType\(\)](#)

Implements [DbManager](#).

8.108.2.6 distinct (*\$fields*, *\$options* = array())

See also

[Gino.Plugin.pdo::distinct\(\)](#)

Implements [DbManager](#).

8.108.2.7 dumpDatabase (*\$file*)

See also

[Gino.Plugin.pdo::dumpDatabase\(\)](#)

Implements [DbManager](#).

8.108.2.8 getTableStructure (*\$table*)

See also

[Gino.Plugin.pdo::getTableStructure\(\)](#)

In MySQL:

```
field_float          'n_int' => int 0      'n_precision' => int 0
field_float(10,2)    'n_int' => string '10' 'n_precision' => string '2'
field_decimal(10,2)  'n_int' => string '10' 'n_precision' => string '2'
```

Implements [DbManager](#).

8.108.2.9 limit (*\$range*, *\$offset*)

See also

[Gino.Plugin.pdo::limit\(\)](#)

Implements [DbManager](#).

8.108.2.10 multiActionQuery (*\$file_content*)

See also

[Gino.Plugin.pdo::multiActionQuery\(\)](#)

Implements [DbManager](#).

8.108.2.11 restore (*\$table*, *\$filename*, *\$options* = array())

See also

[Gino.Plugin.pdo::restore\(\)](#)

Implements [DbManager](#).

8.108.2.12 setAttribute () [protected]

See also

[Gino.Plugin.pdo::setAttribute\(\)](#)

8.108.2.13 setCharacterSet () [protected]

See also

[Gino.Plugin.pdo::setCharacterSet\(\)](#)

8.108.2.14 setDSN () [protected]

See also

[Gino.Plugin.pdo::setDSN\(\)](#)

Example of configuration parameters

```
define("DB_HOST", "localhost");  
define("DB_PORT", "3306");
```

8.108.2.15 setFieldName (*\$field*) [protected]

See also

[Gino.Plugin.pdo::setFieldName\(\)](#)

8.108.2.16 setFieldValue (*\$value*) [protected]

See also

[Gino.Plugin.pdo::setFieldValue\(\)](#)

8.108.2.17 SQLForDump (*\$table*, *\$path_to_file*, *\$delimiter*, *\$enclosed*, *\$where*) [protected]

See also

[Gino.Plugin.pdo::SQLForDump\(\)](#)

8.108.2.18 SQLForFieldInformations (*\$table*) [protected]

See also

[Gino.Plugin.pdo::SQLForFieldInformations\(\)](#)

8.108.2.19 SQLForGetFieldsName (*\$table*) [protected]

See also

[Gino.Plugin.pdo::SQLForGetFieldsName\(\)](#)

8.108.2.20 tableexists (*\$table*)

See also

[Gino.Plugin.pdo::tableexists\(\)](#)

Implements [DbManager](#).

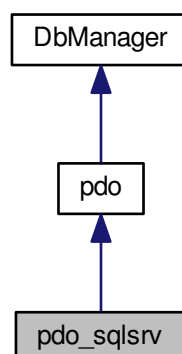
The documentation for this class was generated from the following file:

- lib/plugin/[plugin.pdo_mysql.php](#)

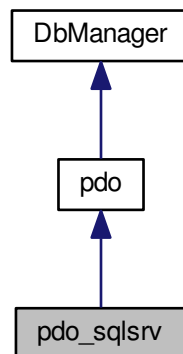
8.109 pdo_sqlsrv Class Reference

Driver specifico per la connessione a un database SQL Server attraverso la libreria PDO.

Inheritance diagram for pdo_sqlsrv:



Collaboration diagram for pdo_sqlsrv:



Public Member Functions

- [multiActionQuery](#) (\$file_content)
- [autoIncValue](#) (\$table)
- [tableexists](#) (\$table)
- [limit](#) (\$range, \$offset=0)
- [distinct](#) (\$fields, \$options=array())
- [concat](#) (\$sequence)
- [dumpDatabase](#) (\$file)
- [getTableStructure](#) (\$table)
- [changeFieldType](#) (\$data_type, \$value)
- [restore](#) (\$table, \$filename, \$options=array())
- [conformFieldType](#) (\$meta)

Protected Member Functions

- [setDSN](#) ()
- [setAttribute](#) ()
- [checkRowsFromSelect](#) (\$result)
- [buildQuery](#) (\$options=array())
- [setFieldName](#) (\$field)
- [setFieldValue](#) (\$value)
- [arrayFieldsToString](#) (\$a_fields)
- [SQLForFieldInformations](#) (\$table)
- [SQLForGetFieldsName](#) (\$table)
- [SQLForDump](#) (\$table, \$path_to_file, \$delimiter, \$enclosed, \$where)

Private Member Functions

- [listTables](#) ()
- [getDataType](#) (\$info)

Ricava il tipo di dato.

- [getFieldLength](#) (\$info)
Ricava il numero di caratteri di un campo.
- [getConstraintType](#) (\$column, \$table)
Verifica se una colonna è una chiave.
- [getCheckConstraint](#) (\$column, \$table)
Verifica se una colonna ha un vincolo CHECK e ne ricava il vincolo.
- [getInformationKey](#) (\$column, \$table, \$key=null)
Verifica se un campo è una particolare chiave e, in caso positivo, ne recupera il nome.
- [setQueryUsingPaging](#) (\$fields, \$tables, \$where=null, \$options=array())
Imposta una query con paginazione.

Additional Inherited Members

8.109.1 Detailed Description

Driver specifico per la connessione a un database SQL Server attraverso la libreria PDO.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.109.2 Member Function Documentation

8.109.2.1 `arrayFieldsToString ($a_fields)` [protected]

See also

[Gino.Plugin.pdo::arrayFieldsToString\(\)](#)

8.109.2.2 `autoIncValue ($table)`

See also

[Gino.Plugin.pdo::autoIncValue\(\)](#)

Implements [DbManager](#).

8.109.2.3 `buildQuery ($options = array())` [protected]

See also

[Gino.Plugin.pdo::buildQuery\(\)](#)

8.109.2.4 `changeFieldType ($data_type, $value)`

See also

[Gino.Plugin.pdo::changeFieldType\(\)](#)

Implements [DbManager](#).

8.109.2.5 `checkRowsFromSelect ($result)` `[protected]`

See also

`::checkRowsFromSelect()`

8.109.2.6 `concat ($sequence)`

See also

[Gino.Plugin.pdo::concat\(\)](#)

Implements [DbManager](#).

8.109.2.7 `conformFieldType ($meta)`

See also

[Gino.Plugin.pdo::conformFieldType\(\)](#)

Implements [DbManager](#).

8.109.2.8 `distinct ($fields, $options = array())`

See also

[Gino.Plugin.pdo::distinct\(\)](#)

Implements [DbManager](#).

8.109.2.9 `dumpDatabase ($file)`

See also

[Gino.Plugin.pdo::dumpDatabase\(\)](#)
[listTables\(\)](#)

Implements [DbManager](#).

8.109.2.10 `getCheckConstraint ($column, $table)` `[private]`

Verifica se una colonna ha un vincolo CHECK e ne ricava il vincolo.

Parameters

string	<i>\$column</i>	
string	<i>\$table</i>	

Returns

string, ([no_admin]='no' OR [no_admin]='yes')

8.109.2.11 `getConstraintType ($column, $table)` `[private]`

Verifica se una colonna è una chiave.

Parameters

string	<i>\$column</i>	
string	<i>\$table</i>	

Returns

array(key, name)

Recupera il nome della chiave:

- *PRIMARY* KEY
- *UNIQUE*
- *FOREIGN* KEY e il suo nome (ad es. PK__page_ent__3213E83FAC2C67F3, UQ__page_ent__32DD1E4C35F37AB2)

8.109.2.12 `getDataType ($info)` [private]

Ricava il tipo di dato.

Il tipo di dato deve essere compatibile con quelli definiti in `Model::dataType()`.

Parameters

array	<i>\$info</i>	
-------	---------------	--

Returns

string

8.109.2.13 `getFieldLength ($info)` [private]

Ricava il numero di caratteri di un campo.

Parameters

array	<i>\$info</i>	
-------	---------------	--

Returns

integer

8.109.2.14 `getInformationKey ($column, $table, $key = null)` [private]

Verifica se un campo è una particolare chiave e, in caso positivo, ne recupera il nome.

Parameters

string	<i>\$column</i>	
string	<i>\$table</i>	
string	<i>\$key</i>	nome della chiave da ricercare <ul style="list-style-type: none"> • <i>PRI</i>, primary key • <i>UNI</i>, unique key • <i>FOR</i>, foreign key

Returns

mixed

8.109.2.15 `getTableStructure ($table)`

See also

[Gino.Plugin.pdo::getTableStructure\(\)](#)
[getConstraintType\(\)](#)
[getDataType\(\)](#)
[getFieldLength\(\)](#)
[getCheckConstraint\(\)](#)
[getInformationKey\(\)](#)

Implements [DbManager](#).

8.109.2.16 `limit ($range, $offset = 0)`

See also

[Gino.Plugin.pdo::limit\(\)](#)

Examples

```
//Returning the first 100 rows from a table called employee:  
select top 100 * from employee  
//Returning the top 20% of rows from a table called employee:  
select top 20 percent * from employee
```

Implements [DbManager](#).

8.109.2.17 `listTables ()` [private]8.109.2.18 `multiActionQuery ($file_content)`

See also

[Gino.Plugin.pdo::multiActionQuery\(\)](#)
[Gino.SqlParse::getQueries\(\)](#)

Implements [DbManager](#).

8.109.2.19 `restore ($table, $filename, $options = array())`

See also

[Gino.Plugin.pdo::restore\(\)](#)

Implements [DbManager](#).

8.109.2.20 `setAttribute ()` [protected]

See also

[Gino.Plugin.pdo::setAttribute\(\)](#)

8.109.2.21 setDSN () [protected]

See also

[Gino.Plugin.pdo::setDSN\(\)](#)

Example of configuration parameters

```
define("DB_HOST", "(local)\SQLEXPRESS");    // (local): host name
define("DB_PORT", "1433");
```

8.109.2.22 setFieldName (\$field) [protected]

See also

[Gino.Plugin.pdo::setFieldName\(\)](#)

8.109.2.23 setFieldValue (\$value) [protected]

See also

[Gino.Plugin.pdo::setFieldValue\(\)](#)

8.109.2.24 setQueryUsingPaging (\$fields, \$tables, \$where = null, \$options = array()) [private]

Imposta una query con paginazione.

Parameters

string array	<i>\$fields</i>	
string array	<i>\$tables</i>	
string	<i>\$where</i>	
array	<i>\$options</i>	<ul style="list-style-type: none">• order (string)• limit (array)• debug (boolean)• distinct (string)

Returns

string

8.109.2.25 SQLForDump (\$table, \$path_to_file, \$delimiter, \$enclosed, \$where) [protected]

See also

[Gino.Plugin.pdo::SQLForDump\(\)](#)

8.109.2.26 `SQLForFieldInformations ($table)` [protected]

See also

[Gino.Plugin.pdo::SQLForFieldInformations\(\)](#)

8.109.2.27 `SQLForGetFieldsName ($table)` [protected]

See also

[Gino.Plugin.pdo::SQLForGetFieldsName\(\)](#)

8.109.2.28 `tableexists ($table)`

See also

[Gino.Plugin.pdo::tableexists\(\)](#)

Implements [DbManager](#).

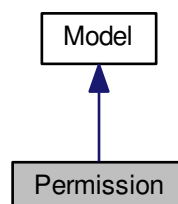
The documentation for this class was generated from the following file:

- [lib/plugin/plugin.pdo_sqlsrv.php](#)

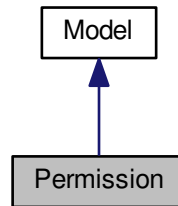
8.110 Permission Class Reference

Classe tipo [Gino.Model](#) che rappresenta un permesso.

Inheritance diagram for Permission:



Collaboration diagram for Permission:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.

Static Public Member Functions

- static [columns](#) ()
Struttura dei campi della tabella di un modello.
- static [getFromFullCode](#) (\$code)
Restituisce un oggetto dato il codice completo.
- static [getForMulticheck](#) ()
Restituisce un array da utilizzare con un widget multicheck.
- static [getList](#) ()
Elenco dei permessi.

Static Public Attributes

- static [\\$table](#) = TBL_PERMISSION
- static [\\$table_perm_user](#) = TBL_USER_PERMISSION
- static [\\$columns](#)

Static Private Member Functions

- static [getGroupedPermissions](#) ()
Restituisce i permessi raggruppati per modulo.

Additional Inherited Members

8.110.1 Detailed Description

Classe tipo [Gino.Model](#) che rappresenta un permesso.

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

I permessi sono relativi ad una classe. Nel momento in cui vengono associati a gruppi o utenti si aggiunge l'informazione istanza, che vale 0 per classi non istanziabili. Quando richiedo il controllo bloccante di un permesso ([Gino.Access::requirePerm\(\)](#)) devo dare classe, codice e istanza. Il codice puo' essere un array di codici di permessi tutti appartenenti alla stessa classe e istanza. Se voglio permettere l'accesso con permessi di classi o istanze diverse uso più volte [Gino.Access::requirePerm\(\)](#).

I [Controller](#) hanno una shortcut di [Gino.Access::requirePerm\(\)](#) dove classe e istanza sono passati direttamente e coincidono con le proprietà del [Controller](#) stesso.

La stessa cosa per il controllo non bloccante, con i metodi [Gino.App.Auth.User::hasPerm\(\)](#) e lo shortcut [Gino.<Controller::userHasPerm\(\)](#).

Esistono permessi di sistema (hanno class='core') che sono generici. La classe [Gino.Core](#) non esiste tra le ModuleApp, ma l'eccezione viene gestita interamente all'interno della classe [Permission](#), per tutti gli altri si tratta di permessi analoghi agli altri e non vedono alcuna differenza. Il nome della classe è core e non system siccome core è una classe esistente e non si può creare un modulo di uguale nome.

I [Controller](#) devono definire i permessi per le outputFunction(), in quel contesto si considerano i permessi di classe, non di istanza. Servono per mostrare i permessi quando si trattano i metodi di output (menu, layout etc...). In tale contesto l'indicazione del permesso deve essere necessariamente nel formato classe.codice, in questo modo si possono attribuire permessi di tipo core alla visualizzazione dei metodi, oppure permessi di altre classi.

Non è stata utilizzata la stessa notazione per i controlli in quanto in questo caso bisogna specificare anche l'istanza e diventerebbe complicato, ed inoltre non si potrebbero definire gli shortcut che nel 90% dei casi sono sufficienti.

Per recuperare i permessi di tutti i moduli (istanziabili e non) ed i permessi core da utilizzare in un form per una associazione, utilizzare [Gino.App.Auth.Permission::getForMulticheck\(\)](#).

8.110.2 Constructor & Destructor Documentation

8.110.2.1 __construct (\$id)

Costruttore.

Parameters

int	<i>\$id</i>	id permesso
-----	-------------	-------------

Returns

istanza di [Gino.App.Auth.Permission](#)

8.110.3 Member Function Documentation

8.110.3.1 __toString ()

Rappresentaazione a stringa dell'oggetto.

Returns

etichetta permesso

8.110.3.2 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.110.3.3 static getForMulticheck () [static]

Restituisce un array da utilizzare con un widget multicheck.

Returns

array

8.110.3.4 static getFromFullCode (\$code) [static]

Restituisce un oggetto dato il codice completo.

Parameters

string	<i>\$code</i>	codice nel formato classname.code_perm
--------	---------------	--

Returns

istanza di [Gino.App.Auth.Permission](#) o null se non lo trova

8.110.3.5 static getGroupedPermissions () [static],[private]

Restituisce i permessi raggruppati per modulo.

Returns

array nome_modulo.modulo_id => Gino.Auth.App.Permission

8.110.3.6 static getList () [static]

Elenco dei permessi.

Vengono mostrati:

- per i moduli non istanziabili -> i loro permessi
- per i moduli istanziabili -> i permessi dei moduli per ogni loro istanza

Returns

array array(perm_id, name, label)

8.110.4 Field Documentation

8.110.4.1 `$columns` `[static]`

8.110.4.2 `$table = TBL_PERMISSION` `[static]`

8.110.4.3 `$table_perm_user = TBL_USER_PERMISSION` `[static]`

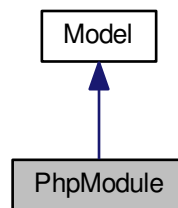
The documentation for this class was generated from the following file:

- `app/auth/class.Permission.php`

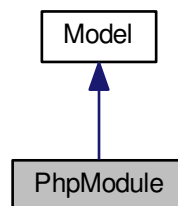
8.111 PhpModule Class Reference

Classe di tipo `Gino.Model` che rappresenta un output scritto in codice php.

Inheritance diagram for PhpModule:



Collaboration diagram for PhpModule:



Public Member Functions

- `__construct` (`$id`, `$controller`)
Costruttore.
- `setInstance` (`$value`)

Setter della proprietà instance.

- [setContent](#) (\$value)

Setter della proprietà content (codice php)

- [formPhpModule](#) ()

Form di inserimento e modifica del codice php.

- [actionPhpModule](#) (\Gino\Http\Request \$request)

Processa il form di modifica/inserimento codice php.

Static Public Member Functions

- static [columns](#) ()

Struttura dei campi della tabella di un modello.

- static [getFromInstance](#) (\$controller)

Recupera l'oggetto a aprtire dall'istanza del [Controller](#).

Static Public Attributes

- static [\\$table](#) = 'php_module'
- static [\\$columns](#)

Protected Attributes

- [\\$_tbl_data](#)

Private Attributes

- [\\$_interface](#)

Additional Inherited Members

8.111.1 Detailed Description

Classe di tipo [Gino.Model](#) che rappresenta un output scritto in codice php.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.111.2 Constructor & Destructor Documentation

8.111.2.1 `__construct` (\$id, \$controller)

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID dell'istanza
	Gino.App\Php↔ ModuleView.↔ phpModuleView	\$controller controller

Returns

void

8.111.3 Member Function Documentation

8.111.3.1 actionPhpModule (\Gino\Http\Request \$request)

Processa il form di modifica/inserimento codice php.

See also

self::formPhpModule()

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

8.111.3.2 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.111.3.3 formPhpModule ()

[Form](#) di inserimento e modifica del codice php.

Returns

html, form

8.111.3.4 static getFromInstance (\$controller) [static]

Recupera l'oggetto a aprtire dall'istanza del [Controller](#).

Parameters

Gino.App.PhpModuleView.phpModuleView	\$controller istanza controller Gino.App.PhpModuleView.PhpModuleView
--	--

Returns

[Gino.App.PhpModuleView.PhpModule](#)

8.111.3.5 setContent (\$value)

Setter della proprietà content (codice php)

Parameters

string	\$value	
--------	---------	--

Returns

TRUE

8.111.3.6 setInstance (\$value)

Setter della proprietà instance.

Parameters

int	\$value	
-----	---------	--

Returns

TRUE

8.111.4 Field Documentation

8.111.4.1 `$_interface` [private]

8.111.4.2 `$_tbl_data` [protected]

8.111.4.3 `$columns` [static]

8.111.4.4 `$table = 'php_module'` [static]

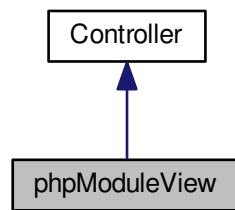
The documentation for this class was generated from the following file:

- [app/phpModuleView/class.PhpModule.php](#)

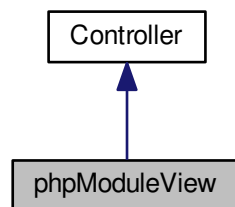
8.112 phpModuleView Class Reference

Permette la creazione di moduli di classe in grado di eseguire codice php completamente personalizzabile.

Inheritance diagram for phpModuleView:



Collaboration diagram for phpModuleView:



Public Member Functions

- [__construct](#) (\$mdlId)
- [deleteInstance](#) ()
Eliminazione di una istanza.
- [viewList](#) ()
Visualizzazione del modulo.
- [manageDoc](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa per la gestione dei moduli di classe '[phpModuleView](#)'.

Static Public Member Functions

- static [getClassElements](#) ()
Restituisce alcune proprietà della classe.
- static [outputFunctions](#) ()
Definizione dei metodi pubblici che forniscono un output per il front-end.

Data Fields

- [\\$_title_visible](#)
- [\\$_optionsLabels](#)

Private Member Functions

- [info \(\)](#)

Informazioni modulo.

Private Attributes

- [\\$_tbl_opt](#)
- [\\$_blackList](#)
- [\\$_title](#)
- [\\$_options](#)

Additional Inherited Members

8.112.1 Detailed Description

Permette la creazione di moduli di classe in grado di eseguire codice php completamente personalizzabile.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

PROCEDURA

- si crea un modulo di classe selezionando la classe [phpModuleView](#)
- il modulo diventa visibile nella sezione moduli
- selezionando il modulo creato si accede alle funzionalità della classe [phpModuleView](#)
- nella sezione *Contenuto* è possibile scrivere direttamente il codice php

Per precauzione tutte le funzioni di php che permettono di eseguire programmi direttamente sulla macchina sono vietate. Nel caso in cui venisse rilevata la presenza di una di queste funzioni il codice non verrebbe eseguito e l'output risultante sarebbe nullo.

Per una corretta integrazione dell'output prodotto all'interno del layout del sito, si consiglia di non utilizzare le funzioni per la stampa diretta *echo* e *print*, ma di immagazzinare tutto l'output all'interno della variabile *\$buffer*, che verrà stampata all'interno del layout. Si consiglia di fare molta attenzione perché nonostante l'accesso alle funzionalità più pericolose del php sia proibito, si ha un controllo completo sulle variabili, ed in caso di cattivo uso del modulo si potrebbe seriamente compromettere la visualizzazione del modulo o dell'intero sito.

PERMESSI

- `can_admin`, amministrazione completa modulo

8.112.2 Constructor & Destructor Documentation

8.112.2.1 `__construct ($moduleId)`

8.112.3 Member Function Documentation

8.112.3.1 `deleteInstance ()`

Eliminazione di una istanza.

Returns

risultato operazione, bool

8.112.3.2 `static getClassElements () [static]`

Restituisce alcune proprietà della classe.

Returns

array associativo contenente le tabelle, viste e struttura directory contenuti

8.112.3.3 `info () [private]`

Informazioni modulo.

Returns

html, informazioni

8.112.3.4 `manageDoc (\Gino\Http\Request $request)`

Interfaccia amministrativa per la gestione dei moduli di classe '[phpModuleView](#)'.

Parameters

<code>\Gino\↵ Http\↵ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Response](#)

8.112.3.5 `static outputFunctions () [static]`

Definizione dei metodi pubblici che forniscono un output per il front-end.

Questo metodo viene letto dal motore di generazione dei layout (prende i metodi non presenti nel file ini) e dal motore di generazione di voci di menu (presenti nel file ini) per presentare una lista di output associati all'istanza di classe.

Returns

array associativo metodi pubblici metodo => array('label' => label, 'permissions' => permissions)

8.112.3.6 viewList ()

Visualizzazione del modulo.

Returns

[Gino.Http.Response](#)

8.112.4 Field Documentation

8.112.4.1 `$_blackList` [private]

8.112.4.2 `$_options` [private]

8.112.4.3 `$_optionsLabels`

8.112.4.4 `$_tbl_opt` [private]

8.112.4.5 `$_title` [private]

8.112.4.6 `$_title_visible`

The documentation for this class was generated from the following file:

- [app/phpModuleView/class_phpModuleView.php](#)

8.113 plugin_phpfastcache Class Reference

Interfaccia alla libreria phpfastcache.

Public Member Functions

- [__construct](#) (\$options=array())
Constructor.
- [getIdentityKeyword](#) ()
Getter della proprietà identity_keyword.
- [setIdentityKeyword](#) (\$value)
Setter della proprietà identity_keyword.
- [get](#) (\$identity_keyword=null)
Try to get from cache.
- [set](#) (\$results, \$options=array())
Set to cache.
- [delete](#) (\$identity_keyword)
Delete from cache.
- [clean](#) ()
Clean all from cache.
- [getInfo](#) ()

Static Private Member Functions

- static [cacheType](#) ()
Tipologie di cache supportate.

Private Attributes

- `$_obj_cache`
- `$_content`
- `$_identity_keyword` =null
- `$_cache_time`

8.113.1 Detailed Description

Interfaccia alla libreria phpfastcache.

See also

<http://www.phpfastcache.com/>

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.113.2 Constructor & Destructor Documentation

8.113.2.1 `__construct ($options = array())`

Constructor.

Parameters

array	<i>\$options</i>	array associativo di opzioni che sovrascrivono i valori delle costanti definite nel file configuration.php <ul style="list-style-type: none">• cache_time (integer): tempo di durata della cache (default 30')• cache_path (string): directory di salvataggio della cache• cache_type (string): tipologia di cache (
-------	------------------	---

See also

[cacheType\(\)](#)

- **cache_server** (array): parametri di connessione necessari per alcune tipologie di cache
- **cache_fallback** (string): tipologia di fallback della cache

Returns

void

8.113.3 Member Function Documentation

8.113.3.1 `static cacheType ()` [*static*],[*private*]

Tipologie di cache supportate.

Returns

array

8.113.3.2 clean ()

Clean all from cache.

8.113.3.3 delete (*\$identity_keyword*)

Delete from cache.

8.113.3.4 get (*\$identity_keyword* = null)

Try to get from cache.

Parameters

string	<i>\$identity_↔ keyword</i>	
--------	---------------------------------	--

Returns

mixed

8.113.3.5 getIdentityKeyword ()

Getter della proprietà identity_keyword.

Returns

string

8.113.3.6 getInfo ()

8.113.3.7 set (*\$results*, *\$options* = array())

Set to cache.

Parameters

mixed	<i>\$results</i>	
array	<i>\$options</i>	

8.113.3.8 setIdentityKeyword (*\$value*)

Setter della proprietà identity_keyword.

Parameters

string	<i>\$value</i>	
--------	----------------	--

Returns

void

8.113.4 Field Documentation

8.113.4.1 `$_cache_time` [private]

8.113.4.2 `$_content` [private]

8.113.4.3 `$_identity_keyword = null` [private]

8.113.4.4 `$_obj_cache` [private]

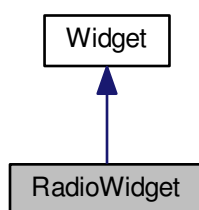
The documentation for this class was generated from the following file:

- lib/plugin/[plugin.phpfastcache.php](#)

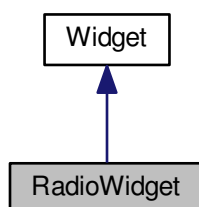
8.114 RadioWidget Class Reference

Campi di tipo radio button nei form.

Inheritance diagram for RadioWidget:



Collaboration diagram for RadioWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.114.1 Detailed Description

Campi di tipo radio button nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.114.2 Member Function Documentation

8.114.2.1 printInputForm (*\$form*, *\$options*)

See also

[Gino.Widget::printInputForm\(\)](#)

The documentation for this class was generated from the following file:

- lib/classes/widget/[class.RadioWidget.php](#)

8.115 ReCaptchaResponse Class Reference

A [ReCaptchaResponse](#) is returned from [recaptcha_check_answer\(\)](#)

Data Fields

- [\\$is_valid](#)
- [\\$error](#)

8.115.1 Detailed Description

A [ReCaptchaResponse](#) is returned from [recaptcha_check_answer\(\)](#)

8.115.2 Field Documentation

8.115.2.1 \$error

8.115.2.2 \$is_valid

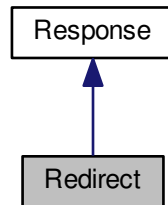
The documentation for this class was generated from the following file:

- lib/[recaptchalib.php](#)

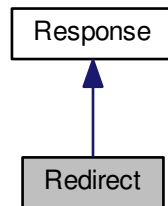
8.116 Redirect Class Reference

Subclass di [Gino.Http.Response](#) per gestire reindirizzamenti.

Inheritance diagram for Redirect:



Collaboration diagram for Redirect:



Public Member Functions

- [__construct](#) (\$url, array \$kwargs=array())
Costruttore.
- [send](#) ()
Invia la risposta HTTP Invia solamente un header per il redirect.

Protected Member Functions

- [sendHeaders](#) ()
Invia gli header della richiesta HTTP Invia un header per il redirect.

Private Member Functions

- [redirectUrl](#) (\$url)
Redirect url.

Private Attributes

- [\\$_url](#)

Additional Inherited Members

8.116.1 Detailed Description

Subclass di [Gino.Http.Response](#) per gestire reindirizzamenti.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.116.2 Constructor & Destructor Documentation

8.116.2.1 `__construct ($url, array $kwargs = array())`

Costruttore.

Parameters

string	<i>\$url</i>	redirect url
array	<i>\$kwargs</i>	

Returns

istanza di [Gino.Http.Redirect](#)

8.116.3 Member Function Documentation

8.116.3.1 `redirectUrl ($url) [private]`

Redrect url.

Ricava un url adatto per il redirect. Se l'url è assoluto oppure relativo a partire dalla document root lo restituisce invariato, altrimenti aggiunge il path di differenza tra document root e site root

Parameters

string	<i>\$url</i>	
--------	--------------	--

Returns

url

8.116.3.2 `send ()`

Invia la risposta HTTP Invia solamente un header per il redirect.

Returns

void

8.116.3.3 `sendHeaders ()` [protected]

Invia gli header della richiesta HTTP Invia un header per il redirect.

Returns

void

8.116.4 Field Documentation

8.116.4.1 `$_url` [private]

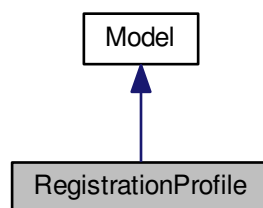
The documentation for this class was generated from the following file:

- lib/classes/http/[class.Redirect.php](#)

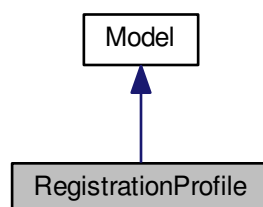
8.117 RegistrationProfile Class Reference

Classe tipo [Gino.Model](#) che rappresenta un profilo di registrazione utenti.

Inheritance diagram for RegistrationProfile:



Collaboration diagram for RegistrationProfile:



Public Member Functions

- `__construct ($id)`
Costruttore.
- `__toString ()`
Rappresentazione a stringa dell'oggetto.
- `getUrl ()`
Url pagina di registrazione profilo.
- `informationApp ()`
Istanza del modulo utilizzato per gestire informazioni aggiuntive.

Static Public Member Functions

- static `columns ()`
Struttura dei campi della tabella di un modello.

Data Fields

- const `MODULE_TYPE_SYS = 1`
- const `MODULE_TYPE_INS = 2`

Static Public Attributes

- static `$table = TBL_REGISTRATION_PROFILE`
- static `$table_groups = TBL_REGISTRATION_PROFILE_GROUP`
- static `$columns`

Additional Inherited Members

8.117.1 Detailed Description

Classe tipo `Gino.Model` che rappresenta un profilo di registrazione utenti.

Copyright

2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.117.2 Constructor & Destructor Documentation

8.117.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.App.Auth.RegistrationProfile](#)

8.117.3 Member Function Documentation

8.117.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

descrizione profilo

8.117.3.2 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.117.3.3 getUrl ()

Url pagina di registrazione profilo.

Returns

url

8.117.3.4 informationApp ()

Istanza del modulo utilizzato per gestire informazioni aggiuntive.

Returns

istanza modulo, object

8.117.4 Field Documentation

8.117.4.1 \$columns [static]

8.117.4.2 \$table = TBL_REGISTRATION_PROFILE [static]

8.117.4.3 \$table_groups = TBL_REGISTRATION_PROFILE_GROUP [static]

8.117.4.4 const MODULE_TYPE_INS = 2

8.117.4.5 const MODULE_TYPE_SYS = 1

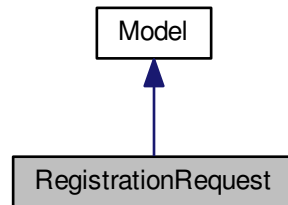
The documentation for this class was generated from the following file:

- app/auth/[class.RegistrationProfile.php](#)

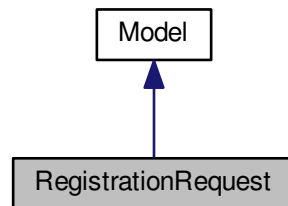
8.118 RegistrationRequest Class Reference

Classe tipo [Gino.Model](#) che rappresenta una richiesta di registrazione.

Inheritance diagram for RegistrationRequest:



Collaboration diagram for RegistrationRequest:



Public Member Functions

- `__construct ($id)`
Costruttore.
- `getOrActivateUser ()`
Nome utente o link per sua creazione e attivazione.

Static Public Member Functions

- static `columns ()`
Struttura dei campi della tabella di un modello.

Static Public Attributes

- static `$table = TBL_REGISTRATION_REQUEST`
- static `$columns`

Additional Inherited Members

8.118.1 Detailed Description

Classe tipo [Gino.Model](#) che rappresenta una richiesta di registrazione.

Copyright

2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.118.2 Constructor & Destructor Documentation

8.118.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.App.Auth.RegistrationProfile](#)

8.118.3 Member Function Documentation

8.118.3.1 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.118.3.2 `getOrActivateUser ()`

Nome utente o link per sua creazione e attivazione.

Returns

string

8.118.4 Field Documentation

8.118.4.1 `$columns [static]`

8.118.4.2 `$table = TBL_REGISTRATION_REQUEST [static]`

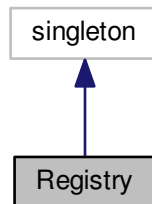
The documentation for this class was generated from the following file:

- [app/auth/class.RegistrationRequest.php](#)

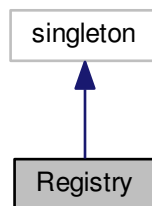
8.119 Registry Class Reference

Registro di gino.

Inheritance diagram for Registry:



Collaboration diagram for Registry:



Public Member Functions

- [__set](#) (\$index, \$value)
Imposta il valore di una variabile.
- [__get](#) (\$index)
Ritorna il valore di una variabile.
- [propertyExists](#) (\$prop)
Controlla se è stata definita una proprietà del registry.
- [addCss](#) (\$css)
Aggiunge un file css.
- [addJs](#) (\$js)
Aggiunge un file javascript.
- [addCustomJs](#) (\$js, \$options=array())
Aggiunge un file javascript con eventuali opzioni.
- [addMeta](#) (\$meta)
Aggiunge un META TAG.
- [addHeadLink](#) (\$link)

Aggiunge un LINK TAG.

- **variables** (\$var)

Codice HTML da inserire nell'HEAD del documento della variabile fornita (CSS, JS, META, ...)

Private Attributes

- **\$vars** = array()

Array associativo che contiene le chiavi del registro private.

8.119.1 Detailed Description

Registro di gino.

Il registro è un [Gino.Singleton](#) che può conservare dati accessibili e modificabili da tutte le classi di gino. Le variabili sono chiavi di un array privato della classe e sono accessibili direttamente grazie all'utilizzo dei metodi `__get` e `__set`.

In particolare conserva alcune variabili che vengono utilizzate nella creazione dei template:

- **title** (valore del tag meta con name 'title')
- **description** (valore del tag meta con name 'description')
- **keywords** (valore del tag meta con name 'keywords')
- **favicon** (percorso relativo della favicon)
- **css** (array di path di file css)
- **js** (array di path di file javascript)
- **custom_js** (array di path di file javascript con eventuali opzioni)
- **meta** (tag meta aggiuntivi)
- **head_links** (link)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

Caricamento [Javascript](#)

I file Js vengono trattati in diversi modi a seconda del valore della costante `DEBUG` e del tipo di caricamento nel registro, con `Registry::addJS()` oppure `Registry::addCustomJS()`.

I file caricati con `Registry::addCustomJS()` vengono accodati a quelli caricati con `Registry::addJS()`.

`Registry::addJS()`

Con `Debug True` i file vengono caricati uno per uno nella versione originale, mentre con `Debug False` i file vengono minificati e compressi in unico file ([Compressor::mergeJs\(\)](#)).

Il file che viene generato viene salvato nella directory `cache/js`, e viene soltanto recuperato se risulta già presente.

Registry::addCustomJS()

Con Debug True i file vengono caricati uno per uno nella versione originale, mentre con Debug False si devono verificare i valori delle opzioni *compress* e *minify* di ogni file.

Se il valore di *compress* è True il file viene inglobato nel file generale che comprende i file caricati con Registry::addJS(). Naturalmente il file compresso viene anche minificato.

Se il valore di *compress* è False e il valore di *minify* è True il file viene minificato, altrimenti viene caricata la sua versione originale.

8.119.2 Member Function Documentation**8.119.2.1 __get (\$index)**

Ritorna il valore di una variabile.

Esempio

```
echo $instance->foo;
```

Parameters

string	<i>\$index</i>	nome della variabile
--------	----------------	----------------------

Returns

valore variabile o null se non definita

8.119.2.2 __set (\$index, \$value)

Imposta il valore di una variabile.

Esempio

```
$instance->foo = 'bar';
```

Parameters

string	<i>\$index</i>	nome della variabile
mixed	<i>\$value</i>	valore della variabile

Returns

void

8.119.2.3 addCss (\$css)

Aggiunge un file css.

Esempio

```
$instance->addCss(CSS_WWW."/file.css");
```

Parameters

string	<i>\$css</i>	percorso relativo al file css
--------	--------------	-------------------------------

Returns

void

8.119.2.4 addCustomJs (*\$js*, *\$options* = array())

Aggiunge un file javascript con eventuali opzioni.

Esempio

```
$instance->addCustomCss(SITE_JS."/file.js", array('compress'=>false, 'minify'=>false));
```

Parameters

string	<i>\$js</i>	percorso relativo al file javascript
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• compress (boolean): indica se inglobare il contenuto del file nel file generale che comprende i file caricati con Registry::addJS()• minify (boolean): indica se effettuare la minificazione del file

Returns

void

8.119.2.5 addHeadLink (*\$link*)

Aggiunge un LINK TAG.

Esempio

```
$instance->addHeadLink(array('rel'=>'external', 'title'=>'foo', 'href'=>''));
```

Parameters

array	<i>\$link</i>	elementi di un tag link (rel, type, title, href)
-------	---------------	--

Returns

void

8.119.2.6 addJs (*\$js*)

Aggiunge un file javascript.

Esempio

```
$instance->addCss(SITE_JS."/file.js");
```

Parameters

string	<i>\$js</i>	percorso relativo al file javascript
--------	-------------	--------------------------------------

Returns

void

8.119.2.7 addMeta (*\$meta*)

Aggiunge un META TAG.

Esempio

```
$instance->addMeta(array('name'=>'bar', 'property'=>'foo'));
```

Parameters

array	<i>\$meta</i>	elementi di un tag meta (name, property, content)
-------	---------------	---

Returns

void

8.119.2.8 propertyExists (*\$prop*)

Controlla se è stata definita una proprietà del registry.

Parameters

<i>\$prop</i>	nome della proprietà
---------------	----------------------

Returns

TRUE se è definita, FALSE altrimenti

8.119.2.9 variables (*\$var*)

Codice HTML da inserire nell'HEAD del documento della variabile fornita (CSS, JS, META, ...)

Si possono fornire le seguenti chiavi:

- title
- description
- keywords
- favicon
- css
- js
- custom_js
- meta
- head_links

I [Css](#) ed i [Js](#) sono trattati dalla classe [Gino.Compressor](#) che permette la minificazione e merge dei file quando la costante `DEBUG` definita in [configuration.php](#) è settata a `FALSE`.

See also

[Gino.Compressor](#)

Parameters

string	<i>\$var</i>	nome della chiave che deve essere recuperata dal registro
--------	--------------	---

Returns

codice html

I file Js vengono trattati in diversi modi a seconda del valore della costante `DEBUG` e del tipo di caricamento nel registro, con `Registry::addJS()` oppure `Registry::addCustomJS()`.

I file caricati con `Registry::addCustomJS()` vengono accodati a quelli caricati con `Registry::addJS()`.

`Registry::addJS()`

Con `Debug True` i file vengono caricati uno per uno nella versione originale, mentre con `Debug False` i file vengono minificati e compressi in unico file ([Compressor::mergeJs\(\)](#)).

Il file che viene generato viene salvato nella directory `cache/js`, e viene soltanto recuperato se risulta già presente.

`Registry::addCustomJS()`

Con `Debug True` i file vengono caricati uno per uno nella versione originale, mentre con `Debug False` si devono verificare i valori delle opzioni `compress` e `minify` di ogni file.

Se il valore di `compress` è `True` il file viene inglobato nel file generale che comprende i file caricati con `Registry::addJS()`. Naturalmente il file compresso viene anche minificato.

Se il valore di `compress` è `False` e il valore di `minify` è `True` il file viene minificato, altrimenti viene caricata la sua versione originale.

8.119.3 Field Documentation

8.119.3.1 `$vars = array()` [`private`]

Array associativo che contiene le chiavi del registro private.

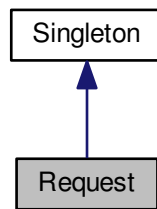
The documentation for this class was generated from the following file:

- `lib/classes/class.Registry.php`

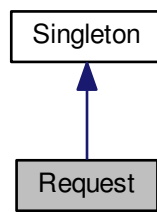
8.120 Request Class Reference

Wrapper di una richiesta HTTP.

Inheritance diagram for Request:



Collaboration diagram for Request:



Public Member Functions

- [updateUrl \(\)](#)
Calcola l'url nella forma espansa a partire dai parametri GET Quando viene effettuato url rewriting da parte di [Gino](#).↔ [Router](#) viene chiamato questo metodo per calcolare l'url non espanso utilizzando la proprietà GET che è stata opportunamente modificata.
- [isSecure \(\)](#)
Connessione sicura https.
- [checkGETKey \(\\$key, \\$value\)](#)
Controlla se la chiave key di GET ha valore value.
- [checkPOSTKey \(\\$key, \\$value\)](#)
Controlla se la chiave key di POST ha valore value.

Data Fields

- [\\$GET](#)
- [\\$POST](#)
- [\\$REQUEST](#)
- [\\$COOKIES](#)
- [\\$FILES](#)
- [\\$META](#)

- [\\$method](#)
- [\\$path](#)
- [\\$url](#)
- [\\$request_uri](#)
- [\\$query_string](#)
- [\\$absolute_url](#)
- [\\$root_absolute_url](#)
- [\\$session](#)
- [\\$user](#) = null

Protected Member Functions

- [__construct](#) ()
Costruttore Il costruttore è protetto in modo da garantire il pattern [Singleton](#).

Private Member Functions

- [valueOrNull](#) (array \$array, \$key)
Valore associato alla chiave data di un array o null.

Additional Inherited Members

8.120.1 Detailed Description

Wrapper di una richiesta HTTP.

Contiene tutte le informazioni importanti di una richiesta HTTP. La classe è un singleton quindi tutte le classi che la utilizzano si scambiano la stessa istanza. Le proprietà sono pubbliche e aperte in lettura e scrittura.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.120.2 Constructor & Destructor Documentation

8.120.2.1 [__construct](#) () [[protected](#)]

Costruttore Il costruttore è protetto in modo da garantire il pattern [Singleton](#).

Returns

nuova istanza di [Gino.Http.Request](#)

8.120.3 Member Function Documentation

8.120.3.1 [checkGETKey](#) ([\\$key](#), [\\$value](#))

Controlla se la chiave key di GET ha valore value.

Parameters

string	<i>\$key</i>	chiave
mixed	<i>\$value</i>	valore

Returns

TRUE se GET[key] = value, FALSE altrimenti

8.120.3.2 checkPOSTKey (*\$key*, *\$value*)

Controlla se la chiave key di POST ha valore value.

Parameters

string	<i>\$key</i>	chiave
mixed	<i>\$value</i>	valore

Returns

TRUE se POST[key] = value, FALSE altrimenti

8.120.3.3 isSecure ()

Connessione sicura https.

Returns

TRUE se la connessione è sicura, FALSE altrimenti

8.120.3.4 updateUrl ()

Calcola l'url nella forma espansa a partire dai parametri GET. Quando viene effettuato url rewriting da parte di [Gino.Router](#) viene chiamato questo metodo per calcolare l'url non espanso utilizzando la proprietà GET che è stata opportunamente modificata.

L'url parte dalla site root senza / iniziale, es. [index.php](#)?evt[page-view]&id=test

Returns

void

8.120.3.5 valueOrNull (array *\$array*, *\$key*) [private]

Valore associato alla chiave data di un array o null.

Parameters

array	<i>\$array</i>	
string	<i>\$key</i>	

Returns

valore associato alla \$key data o null

8.120.4 Field Documentation

8.120.4.1 \$absolute_url

8.120.4.2 \$COOKIES

8.120.4.3 \$FILES

8.120.4.4 \$GET

8.120.4.5 \$META

8.120.4.6 \$method

8.120.4.7 \$path

8.120.4.8 \$POST

8.120.4.9 \$query_string

8.120.4.10 \$REQUEST

8.120.4.11 \$request_uri

8.120.4.12 \$root_absolute_url

8.120.4.13 \$session

8.120.4.14 \$url

8.120.4.15 \$user = null

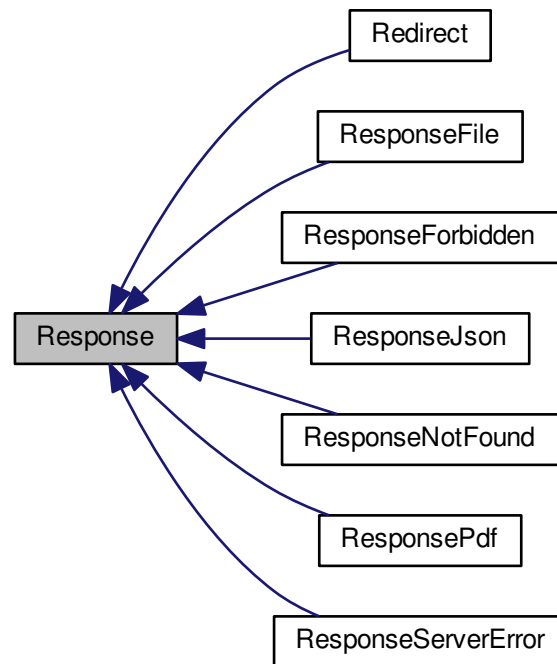
The documentation for this class was generated from the following file:

- lib/classes/http/[class.Request.php](#)

8.121 Response Class Reference

Wrapper di una risposta HTTP.

Inheritance diagram for Response:



Public Member Functions

- `__construct` (\$content, array \$kwargs=array())
Costruttore.
- `__invoke` ()
Invia la risposta HTTP Metodo chiamato quando si usa l'oggetto come funzione.
- `setContent` (\$content)
Setter del contenuto
- `getContent` ()
Getter del contenuto
- `setStatus` (\$code, \$text)
Setter dello status della risposta.
- `setContentType` (\$content_type)
Setter del content type
- `setEncoding` (\$encoding)
Setter dell'encoding
- `setHeaders` (\$headers)
Setter di headers ulteriori
- `send` ()
Invia la risposta HTTP.

Data Fields

- [\\$_content](#)
- [\\$_status_code](#)
- [\\$_status_text](#)
- [\\$_content_type](#)
- [\\$_encoding](#)
- [\\$_headers](#)

Protected Member Functions

- [sendHeaders\(\)](#)
Invia gli header della richiesta HTTP.
- [sendContent\(\)](#)
Invia il corpo della richiesta HTTP.

Protected Attributes

- [\\$_request](#)

8.121.1 Detailed Description

Wrapper di una risposta HTTP.

Tutti i metodi dei [Gino](#) eseguiti da [Gino](#) in risposta ad un url, ritornano un oggetto [Gino.Http.Response](#) o una sua sottoclasse. Questo oggetto si preoccupa di settare gli header e di inviare il contenuto della risposta HTTP

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.121.2 Constructor & Destructor Documentation

8.121.2.1 `__construct ($content, array $kwargs = array())`

Costruttore.

Parameters

string	<i>\$content</i>	contenuto della risposta
array	<i>\$kwargs</i>	array associativo di argomenti

Returns

istanza di [Gino.Http.Response](#)

8.121.3 Member Function Documentation

8.121.3.1 `__invoke ()`

Invia la risposta HTTP Metodo chiamato quando si usa l'oggetto come funzione.

Returns

void

8.121.3.2 `getContent ()`

Getter del contenuto

Returns

contenuto della risposta http

8.121.3.3 `send ()`

Invia la risposta HTTP.

Returns

void

8.121.3.4 `sendContent ()` [protected]

Invia il corpo della richiesta HTTP.

Returns

void

8.121.3.5 `sendHeaders ()` [protected]

Invia gli header della richiesta HTTP.

Returns

void

8.121.3.6 `setContent ($content)`

Setter del contenuto

Parameters

string	<i>\$content</i>	contenuto della risposta http
--------	------------------	-------------------------------

Returns

void

8.121.3.7 `setContentType ($content_type)`

Setter del content type

Parameters

string	<i>\$content_type</i>	
--------	-----------------------	--

Returns

void

8.121.3.8 setEncoding (*\$encoding*)

Setter dell'encoding

Parameters

string	<i>\$encoding</i>	
--------	-------------------	--

Returns

void

8.121.3.9 setHeaders (*\$headers*)

Setter di headers ulteriori

Parameters

<i>array</i>	headers array associativo di headers nella forma chiave=>valore	
--------------	---	--

Returns

void

8.121.3.10 setStatus (*\$code*, *\$text*)

Setter dello status della risposta.

Parameters

int	<i>\$code</i>	codice risposta
	<i>string</i>	text testo dello status

Returns

void

8.121.4 Field Documentation

8.121.4.1 *\$_content*8.121.4.2 *\$_content_type*8.121.4.3 *\$_encoding*8.121.4.4 *\$_headers*

8.121.4.5 `$_request` [protected]

8.121.4.6 `$_status_code`

8.121.4.7 `$_status_text`

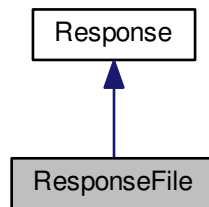
The documentation for this class was generated from the following file:

- lib/classes/http/[class.Response.php](#)

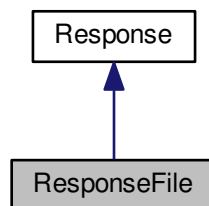
8.122 ResponseFile Class Reference

Subclass di per gestire stream di file.

Inheritance diagram for ResponseFile:



Collaboration diagram for ResponseFile:



Public Member Functions

- [__construct](#) (\$file, \$content_type, \$filename, array \$kwargs=array())
Costruttore.
- [setDispositionType](#) (\$disposition_type)
Setter del disposition type

Data Fields

- [\\$_filename](#)

Protected Member Functions

- [sendHeaders\(\)](#)
Invia gli header della richiesta HTTP.

Protected Attributes

- [\\$_content_disposition](#)

8.122.1 Detailed Description

Subclass di [File](#) per gestire stream di file.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.122.2 Constructor & Destructor Documentation

8.122.2.1 `__construct ($file, $content_type, $filename, array $kwargs = array())`

Costruttore.

Parameters

string	<i>\$file</i>	path assoluto al file o contenuto del file (in tal caso settare a TRUE ← E <code>kwargs['file_is_content']</code>)
string	<i>\$content_type</i>	
string	<i>\$filename</i>	
array	<i>\$kwargs</i>	array associativo <ul style="list-style-type: none"> • <code>disposition_type</code>: string, disposition type header • <code>file_is_content</code>: bool, se TRUE il parametro <code>\$file</code> viene considerato essere il contenuto del file e non il path assoluto

Returns

istanza di

8.122.3 Member Function Documentation

8.122.3.1 `sendHeaders ()` `[protected]`

Invia gli header della richiesta HTTP.

Returns

void

8.122.3.2 `setDispositionType ($disposition_type)`

Setter del disposition type

Parameters

string	<i>\$disposition_type</i>	
--------	---------------------------	--

Returns

void

8.122.4 **Field Documentation****8.122.4.1** `$_content_disposition` [protected]**8.122.4.2** `$_filename`

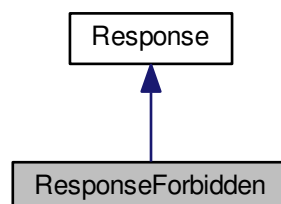
The documentation for this class was generated from the following file:

- lib/classes/http/[class.ResponseFile.php](#)

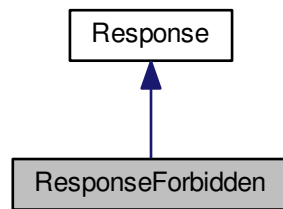
8.123 **ResponseForbidden Class Reference**

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 403.

Inheritance diagram for ResponseForbidden:



Collaboration diagram for ResponseForbidden:



Public Member Functions

- [__construct](#) (array \$kwargs=array())
Costruttore.

Protected Member Functions

- [sendContent](#) ()
Corpo della risposta HTTP Mostra la pagina 403 di gino.

Additional Inherited Members

8.123.1 Detailed Description

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 403.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.123.2 Constructor & Destructor Documentation

8.123.2.1 [__construct](#) (array \$kwargs = array())

Costruttore.

Parameters

array	<i>\$kwargs</i>	
-------	-----------------	--

Returns

istanza di [Gino.Http.ResponseForbidden](#)

8.123.3 Member Function Documentation

8.123.3.1 `sendContent ()` [protected]

Corpo della risposta HTTP Mostra la pagina 403 di gino.

Returns

void

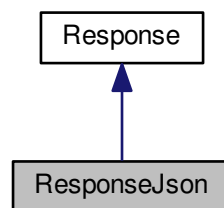
The documentation for this class was generated from the following file:

- lib/classes/http/[class.ResponseForbidden.php](#)

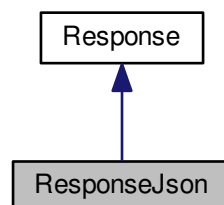
8.124 ResponseJson Class Reference

Subclass di per gestire risposte in formato json.

Inheritance diagram for ResponseJson:



Collaboration diagram for ResponseJson:



Public Member Functions

- [__construct](#) (\$content, array \$kwargs=array())
Costruttore.

Additional Inherited Members

8.124.1 Detailed Description

Subclass di per gestire risposte in formato json.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.124.2 Constructor & Destructor Documentation

8.124.2.1 __construct (*\$content*, array *\$kwargs* = array())

Costruttore.

Parameters

mixed	<i>\$content</i>	contenuto della risposta. Se diverso da stringa viene codificato in json
array	<i>\$kwargs</i>	

Returns

istanza di

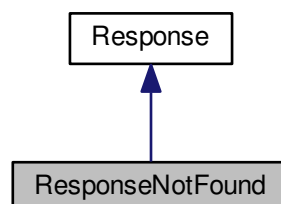
The documentation for this class was generated from the following file:

- lib/classes/http/[class.ResponseJson.php](#)

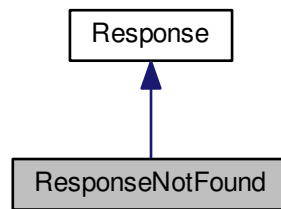
8.125 ResponseNotFound Class Reference

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 404.

Inheritance diagram for ResponseNotFound:



Collaboration diagram for ResponseNotFound:



Public Member Functions

- [__construct](#) (array \$kwargs=array())
Costruttore.

Protected Member Functions

- [sendContent](#) ()
Corpo della risposta HTTP Mostra la pagina 404 di gino.

Additional Inherited Members

8.125.1 Detailed Description

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 404.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.125.2 Constructor & Destructor Documentation

8.125.2.1 __construct (array \$kwargs = array())

Costruttore.

Parameters

array	<i>\$kwargs</i>	
-------	-----------------	--

Returns

istanza di [Gino.Http.ResponseNotFound](#)

8.125.3 Member Function Documentation

8.125.3.1 `sendContent ()` [protected]

Corpo della risposta HTTP Mostra la pagina 404 di gino.

Returns

void

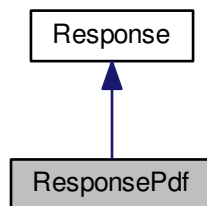
The documentation for this class was generated from the following file:

- lib/classes/http/[class.ResponseNotFound.php](#)

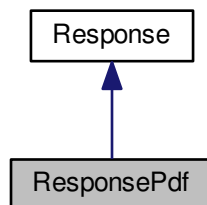
8.126 ResponsePdf Class Reference

Subclass di per gestire risposte in formato pdf.

Inheritance diagram for ResponsePdf:



Collaboration diagram for ResponsePdf:



Public Member Functions

- [__construct](#) (\$content, array \$kwargs=array())
Costruttore.

Additional Inherited Members

8.126.1 Detailed Description

Subclass di per gestire risposte in formato pdf.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.126.2 Constructor & Destructor Documentation

8.126.2.1 __construct (*\$content*, array *\$kwargs* = array())

Costruttore.

Parameters

mixed	<i>\$content</i>	contenuto della risposta. Se diverso da stringa viene codificato in json
array	<i>\$kwargs</i>	array associativo di opzioni <ul style="list-style-type: none">• di ::pdfFromPage()• di ::setPhpParams()

Returns

istanza di

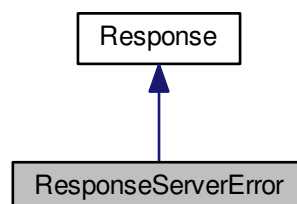
The documentation for this class was generated from the following file:

- lib/classes/http/[class.ResponsePdf.php](#)

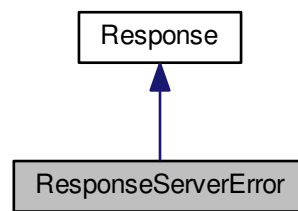
8.127 ResponseServerError Class Reference

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori interni (code 500)

Inheritance diagram for ResponseServerError:



Collaboration diagram for ResponseServerError:



Public Member Functions

- [__construct](#) (array \$kwargs=array())
Costruttore.

Protected Member Functions

- [sendContent](#) ()
Corpo della risposta HTTP Mostra la pagina 500 di gino.

Additional Inherited Members

8.127.1 Detailed Description

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori interni (code 500)

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.127.2 Constructor & Destructor Documentation

8.127.2.1 __construct (array \$kwargs = array())

Costruttore.

Parameters

array	<i>\$kwargs</i>	
-------	-----------------	--

Returns

istanza di [Gino.Http.ResponseServerError](#)

8.127.3 Member Function Documentation

8.127.3.1 `sendContent()` [protected]

Corpo della risposta HTTP Mostra la pagina 500 di gino.

Returns

void

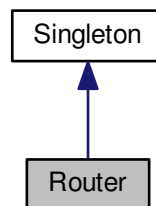
The documentation for this class was generated from the following file:

- lib/classes/http/[class.ResponseServerError.php](#)

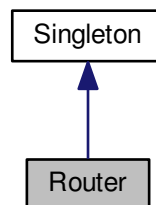
8.128 Router Class Reference

Gestisce il routing di una request HTTP, chiamando la classe e metodo che devono fornire risposta.

Inheritance diagram for Router:



Collaboration diagram for Router:



Public Member Functions

- [isRoute](#) (\$callable)

Controlla se la callable fornita è quella chiamata da url.

- [route](#) ()
Esegue il route della request HTTP Passa la [Gino](#) alla callable che deve gestirla e ritornare una [Gino.Http.Response](#).
- [link](#) (\$instance_name, \$method, array \$params=array(), \$query_string= "", array \$kwargs=array())
Url che linka un metodo di una istanza di controller con parametri dati.
- [transformPathQueryString](#) (array \$add=array(), array \$remove=array())
Trasformazione di un path con aggiunta o rimozione di parametri dalla query string.

Data Fields

- const [EVT_NAME](#) = 'evt'
- [\\$_request](#)
- [\\$_url_class](#)
- [\\$_url_instance](#)
- [\\$_url_method](#)
- [\\$_controller_view](#)

Protected Member Functions

- [__construct](#) ()
Costruttore Esegue l'url rewriting quando si utilizzano pretty urls e setta le variabili che contengono le informazioni della classe e metodo chiamati da url.

Private Member Functions

- [urlRewrite](#) ()
Url rewriting Se l'url non è nella forma pretty riscrive le proprietà GET e REQUEST dell'oggetto [Gino](#) parserizzando l'url.
- [rewritePathInfo](#) (array \$paths)
Riscrittura URL path info.
- [rewriteQueryString](#) (array \$pairs)
Riscrittura URL della query_string.
- [setUrlParams](#) ()
Setta le proprietà che contengono le informazioni della classe e metodo chiamati da url Se i parametri ricavati dall'url tentano di chiamare una callable (classe + metodo) non chiamabile per qualunque motivo, viene generata una [Gino](#).

Private Attributes

- [\\$_registry](#)

Additional Inherited Members

8.128.1 Detailed Description

Gestisce il routing di una request HTTP, chiamando la classe e metodo che devono fornire risposta.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.128.2 Constructor & Destructor Documentation

8.128.2.1 `__construct ()` `[protected]`

Costruttore Esegue l'url rewriting quando si utilizzano pretty urls e setta le variabili che contengono le informazioni della classe e metodo chiamati da url.

8.128.3 Member Function Documentation

8.128.3.1 `isRoute ($callable)`

Controlla se la callable fornita è quella chiamata da url.

Parameters

callable	<i>\$callable</i>	
----------	-------------------	--

Returns

bool

8.128.3.2 `link ($instance_name, $method, array $params = array() , $query_string = ' ', array $kwargs = array())`

Url che linka un metodo di una istanza di controller con parametri dati.

Parameters

string ↳ Gino\↳ Controller	<i>\$instance_name</i>	nome istanza o istanza del Gino
string	<i>\$method</i>	nome metodo
array	<i>\$params</i>	parametri da aggiungere come path info nel caso di pretty url
array string	<i>\$query_string</i>	parametri aggiuntivi da trattare come query_string in entrambi i casi (pretty, espanso)
array	<i>\$kwargs</i>	array associativo <ul style="list-style-type: none"> pretty: bool, default TRUE. Creare un pretty url o un url espanso abs: bool, default FALSE. Se TRUE viene ritornato un url assoluto

Returns

url

8.128.3.3 `rewritePathInfo (array $paths)` `[private]`

Riscrittura URL path info.

Parameters

array	<i>\$paths</i>	parti del path info
-------	----------------	---------------------

Returns

TRUE

<http://example.com/admin>

8.128.3.4 `rewriteQueryString (array $pairs)` [`private`]

Riscrittura URL della `query_string`.

Parameters

array	<i>\$pairs</i>	coppie chiave-valore
-------	----------------	----------------------

Returns

void

8.128.3.5 route ()

Esegue il route della request HTTP Passa la [Gino](#) alla callable che deve gestirla e ritornare una [Gino.Http.Response](#).

Se non è definita una callable, ritorna una [Gino.Http.Response](#) con contenuto vuoto

Returns

[Gino.Http.Response](#)

8.128.3.6 setUrlParams () [private]

Setta le proprietà che contengono le informazioni della classe e metodo chiamati da url Se i parametri ricavati dall'url tentano di chiamare una callable (classe + metodo) non chiamabile per qualunque motivo, viene generata una [Gino](#).

Returns

TRUE

8.128.3.7 transformPathQueryString (array \$add = array(), array \$remove = array())

Trasformazione di un path con aggiunta o rimozione di parametri dalla query string.

Parameters

array	<i>\$add</i>	parametri da aggiungere nella forma parametro => valore
array	<i>\$remove</i>	parametri da rimuovere

Returns

path trasformato

8.128.3.8 urlRewrite () [private]

Url rewriting Se l'url non è nella forma pretty riscrive le proprietà GET e REQUEST dell'oggetto [Gino](#) parserizzando l'url.

Chiama [Gino.Http.Request](#) per fare un update della proprietà url.

Returns

void

8.128.4 Field Documentation

8.128.4.1 `$_controller_view`

8.128.4.2 `$_registry` [`private`]

8.128.4.3 `$_request`

8.128.4.4 `$_url_class`

8.128.4.5 `$_url_instance`

8.128.4.6 `$_url_method`

8.128.4.7 `const EVT_NAME = 'evt'`

The documentation for this class was generated from the following file:

- [lib/classes/class.Router.php](#)

8.129 Search Class Reference

Libreria per ricerche full text pesate sulle tabelle.

Public Member Functions

- [__construct](#) (`$table`, `$opts=array()`)
Costruttore.
- [makeQuery](#) (`$selected_fields`, `$required_clauses`, `$weight_clauses`)
Costruisce la query di una ricerca full text.
- [getSearchResults](#) (`$dbObj`, `$selected_fields`, `$required_clauses`, `$weight_clauses`)
Risultati di una ricerca full text.

Private Member Functions

- [clearSearchString](#) (`$search_string`)
Ripulisce la stringa di ricerca Elimina parole con poco significato.
- [getKeywords](#) (`$search_string`)
Ricava le parole chiave da una stringa di ricerca.

Private Attributes

- [\\$_table](#)

8.129.1 Detailed Description

Libreria per ricerche full text pesate sulle tabelle.

Codice SQL da eseguire sul database MySQL

```

DELIMITER $$

DROP FUNCTION IF EXISTS 'replace_ci'$$
CREATE FUNCTION 'replace_ci' ( str TEXT,needle CHAR(255),str_rep CHAR(255))
RETURNS TEXT
DETERMINISTIC
BEGIN
DECLARE return_str TEXT;
SELECT replace(lower(str),lower(needle),str_rep) INTO return_str;
RETURN return_str;
END$$

DELIMITER ;

```

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.129.2 Constructor & Destructor Documentation

8.129.2.1 `__construct ($table, $opts = array())`

Costruttore.

Parameters

string	<i>\$table</i>	testo del FROM in una query SQL (ad esempio: page AS p, page_block AS pb)
array	<i>\$opts</i>	array associativo di opzioni <ul style="list-style-type: none"> highlight_range (integer)

Returns

istanza di [Gino.Search](#)

8.129.3 Member Function Documentation

8.129.3.1 `clearSearchString ($search_string)` [private]

Ripulisce la stringa di ricerca Elimina parole con poco significato.

Parameters

string	<i>\$search_string</i>	
--------	------------------------	--

Returns

stringa ripulita

8.129.3.2 `getKeywords ($search_string)` [private]

Ricava le parole chiave da una stringa di ricerca.

Parameters

string	<i>\$search_string</i>	
--------	------------------------	--

Returns

array di parole chiave

8.129.3.3 `getSearchResults ($dbObj, $selected_fields, $required_clauses, $weight_clauses)`

Risultati di una ricerca full text.

See also

`self::makeQuery()`

Parameters

object	<i>\$dbObj</i>	istanza del database (<code>db::instance()</code>)
array	<i>\$selected_fields</i>	campi da selezionare nella ricerca
array	<i>\$required_clauses</i>	tipo di ricerca sul testo
array	<i>\$weight_clauses</i>	

Returns

array di risultati, ciascun risultato è un array con chiavi `relevance`, `occurrences`, `<field-name>`

8.129.3.4 `makeQuery ($selected_fields, $required_clauses, $weight_clauses)`

Costruisce la query di una ricerca full text.

Parameters

array	<i>\$selected_fields</i>	campi da selezionare nella ricerca (costruzione del SELECT), ad esempio <code>array("p.item_id", array("highlight"=>true, "field"=>"p.title"), array("highlight"=>true, "p.subtitle"), array("highlight"=>true, "field"=>"pb.text"))</code>
array	<i>\$required_clauses</i>	tipo di ricerca sul testo (costruzione del WHERE), ad esempio <code>array("p.item_id"=>array("field"=>true, "value"=>"pb.item"))</code>
array	<i>\$weight_clauses</i>	(costruzione del WHERE e rilevanza dei risultati), ad esempio <code>array("p.title"=>array("weight"=>3), "p.subtitle"=>array("weight"=>2), "pb.text"=>array("weight"=>1))</code>

Returns

query

8.129.4 Field Documentation

8.129.4.1 `$_table` [`private`]

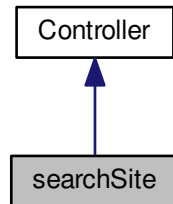
The documentation for this class was generated from the following file:

- `lib/classes/class.Search.php`

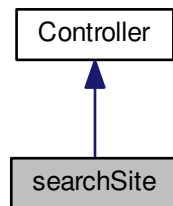
8.130 searchSite Class Reference

Gestisce le ricerche full text sui contenuti dell'applicazione.

Inheritance diagram for searchSite:



Collaboration diagram for searchSite:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [manageSearchSite](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa per la gestione delle ricerche.
- [form](#) ()
Form di ricerca.
- [results](#) (\Gino\Http\Request \$request)
Stampa i risultati di una ricerca.
- [info](#) ()
Informazioni modulo.

Static Public Member Functions

- static [getClassElements](#) ()
Restituisce alcune proprietà della classe.

- static [outputFunctions](#) ()

Definizione dei metodi pubblici che forniscono un output per il front-end.

Data Fields

- [\\$_optionsLabels](#)
- [\\$_sys_mdl](#)
- [\\$_inst_mdl](#)
- [\\$_title](#)

Protected Attributes

- [\\$_view_dir](#)

Private Member Functions

- [checkOptions](#) ()

Pannello con le opzioni di ricerca.

Private Attributes

- [\\$_options](#)

Additional Inherited Members

8.130.1 Detailed Description

Gestisce le ricerche full text sui contenuti dell'applicazione.

Copyright

2005 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.130.2 Constructor & Destructor Documentation

8.130.2.1 `__construct` ()

Costruttore.

Returns

istanza di [Gino.App.SearchSite.searchSite](#)

8.130.3 Member Function Documentation

8.130.3.1 `checkOptions ()` [`private`]

Pannello con le opzioni di ricerca.

Returns

html, pannello

8.130.3.2 `form ()`

Form di ricerca.

Returns

html, form di ricerca

8.130.3.3 `static getClassElements ()` [`static`]

Restituisce alcune proprietà della classe.

Returns

array associativo di proprietà utilizzate per la creazione di istanze di tipo pagina (tabelle, css, viste)

8.130.3.4 `info ()`

Informazioni modulo.

Returns

html, informazioni

8.130.3.5 `manageSearchSite (\Gino\Http\Request $request)`

Interfaccia amministrativa per la gestione delle ricerche.

Parameters

--	--

8.130.3.6 `static outputFunctions ()` [`static`]

Definizione dei metodi pubblici che forniscono un output per il front-end.

Questo metodo viene letto dal motore di generazione dei layout (prende i metodi non presenti nel file ini) e dal motore di generazione di voci di menu (presenti nel file ini) per presentare una lista di output associati all'istanza di classe.

Returns

array associativo metodi pubblici metodo => array('label' => label, 'permissions' => permissions)

8.130.3.7 results (\Gino\Http\Request \$request)

Stampa i risultati di una ricerca.

La ricerca viene effettuata sui moduli nei quali sono stati definiti i metodi *searchSite()* e *searchSiteResult()*

See also

[Gino.Search::getSearchResults\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	\$request	istanza di Gino.Http.Request
------------------------------	-----------	--

Returns

[Gino.Http.Response](#)

8.130.4 Field Documentation

8.130.4.1 \$_inst_mdl

8.130.4.2 \$_options [private]

8.130.4.3 \$_optionsLabels

8.130.4.4 \$_sys_mdl

8.130.4.5 \$_title

8.130.4.6 \$_view_dir [protected]

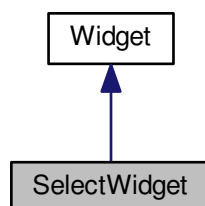
The documentation for this class was generated from the following file:

- app/searchSite/[class_searchSite.php](#)

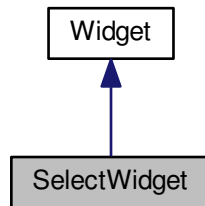
8.131 SelectWidget Class Reference

Campi di tipo select nei form.

Inheritance diagram for SelectWidget:



Collaboration diagram for SelectWidget:



Public Member Functions

- [printInputForm](#) (`$form`, `$options`)

Additional Inherited Members

8.131.1 Detailed Description

Campi di tipo select nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.131.2 Member Function Documentation

8.131.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

Parameters

array	<i>\$options</i>	<ul style="list-style-type: none">• choice (mixed): elementi del select
-------	------------------	--

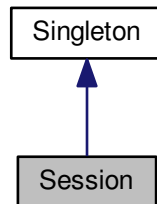
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.SelectWidget.php](#)

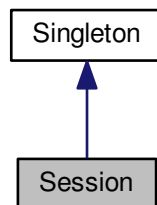
8.132 Session Class Reference

Classe per la gestione delle variabili di sessione.

Inheritance diagram for Session:



Collaboration diagram for Session:



Public Member Functions

- [startSession](#) (\$session_name="")
(Ri)avvia la sessione
- [__set](#) (\$name, \$value)
Imposta il valore di una variabile di sessione.
- [__get](#) (\$name)
Ritorna il valore di una variabile di sessione.
- [__isset](#) (\$name)
Verifica se una variabile di sessione esiste.
- [__unset](#) (\$name)
Distrugge una variabile di sessione.
- [destroy](#) ()
Distrugge la sessione corrente.

Data Fields

- const `SESSION_STARTED` = TRUE
- const `SESSION_NOT_STARTED` = FALSE

Protected Member Functions

- `__construct()`
Costruttore Avvia la sessione.

Private Attributes

- `$sessionState` = self::SESSION_NOT_STARTED
- `$_vars`

Additional Inherited Members

8.132.1 Detailed Description

Classe per la gestione delle variabili di sessione.

La classe è di tipo [Gino.Singleton](#) per garantire l'esistenza di una sola istanza a runtime.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.132.2 Constructor & Destructor Documentation

8.132.2.1 `__construct()` [protected]

Costruttore Avvia la sessione.

8.132.3 Member Function Documentation

8.132.3.1 `__get($name)`

Ritorna il valore di una variabile di sessione.

Esempio

```
echo $instance->foo;
```


Parameters

string	<i>\$name</i>	nome della variabile di sessione
--------	---------------	----------------------------------

Returns

valore variabile di sessione o null

8.132.3.2 `__isset ($name)`

Verifica se una variabile di sessione esiste.

Parameters

string	<i>\$name</i>	nome della variabile di sessione
--------	---------------	----------------------------------

Returns

TRUE se esiste, FALSE altrimenti

8.132.3.3 `__set ($name, $value)`

Imposta il valore di una variabile di sessione.

Esempio

```
$instance->foo = 'bar';
```

Parameters

string	<i>\$name</i>	nome della variabile di sessione
mixed	<i>\$value</i>	valore della variabile di sessione

Returns

void

8.132.3.4 `__unset ($name)`

Distrugge una variabile di sessione.

Parameters

string	<i>\$name</i>	nome della variabile di sessione
--------	---------------	----------------------------------

Returns

void

8.132.3.5 `destroy ()`

Distrugge la sessione corrente.

Returns

FALSE

8.132.3.6 `startSession ($session_name = ' ')`

(Ri)avvia la sessione

Parameters

string	<code>\$session_name</code>	nome della sessione (se non indicato viene impostato all'esterno della classe)
--------	-----------------------------	--

Returns

boolean TRUE se le sessione è stata inizializzata, altrimenti FALSE

8.132.4 Field Documentation

8.132.4.1 `$_vars` [private]

8.132.4.2 `$sessionState = self::SESSION_NOT_STARTED` [private]

8.132.4.3 `const SESSION_NOT_STARTED = FALSE`

8.132.4.4 `const SESSION_STARTED = TRUE`

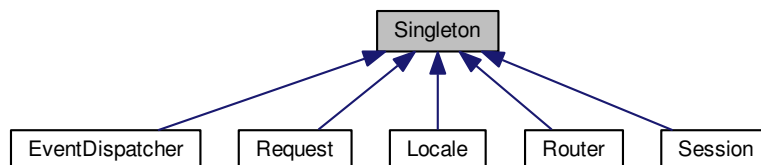
The documentation for this class was generated from the following file:

- lib/classes/[class.Session.php](#)

8.133 Singleton Class Reference

Classe astratta per implementazione del [Singleton](#) pattern.

Inheritance diagram for Singleton:



Public Member Functions

- [__clone](#) ()
I [Singleton](#) non possono essere clonati.
- [__sleep](#) ()
I [Singleton](#) non possono essere serializzati.
- [__wakeup](#) ()
I [Singleton](#) non possono essere serializzati.

Static Public Member Functions

- static [instance](#) ()
Metodo per recuperare istanze [Singleton](#).

- static [instance_to_class](#) (\$main_class)

Metodo per recuperare istanze [Singleton](#) rispetto alla classe fornita.

Protected Member Functions

- [__construct](#) ()

Costruttore Il costruttore è definito come metodo protetto in modo che classi client non possano ottenere nuove istanze di una class [Gino.Singleton](#) attraverso di esso.

Static Protected Attributes

- static [\\$_instances](#) = array()

8.133.1 Detailed Description

Classe astratta per implementazione del [Singleton](#) pattern.

Garantisce che che non vengano create istanze multiple delle classi che la estendono. Quando una classe richiede un'istanza di una classe di tipo [Gino.Singleton](#) riceve sempre la stessa istanza, uguale a quella che ricevono tutte le altre classi. Il pattern [Singleton](#) permette di implementare al meglio le classi che rappresentano il controllo del database, la http request, il registro di sistema e la sessione.

See also

[Gino.Registry](#)
[Gino.Session](#)
[Gino.Db](#)
[Gino.Http.Request](#)

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.133.2 Constructor & Destructor Documentation

8.133.2.1 [__construct](#) () [protected]

Costruttore Il costruttore è definito come metodo protetto in modo che classi client non possano ottenere nuove istanze di una class [Gino.Singleton](#) attraverso di esso.

8.133.3 Member Function Documentation

8.133.3.1 [__clone](#) ()

I [Singleton](#) non possono essere clonati.

8.133.3.2 `__sleep ()`

I [Singleton](#) non possono essere serializzati.

8.133.3.3 `__wakeup ()`

I [Singleton](#) non possono essere serializzati.

8.133.3.4 `static instance ()` `[static]`

Metodo per recuperare istanze [Singleton](#).

Per ogni classe [Singleton](#) ritorna sempre la stessa istanza

Returns

object

8.133.3.5 `static instance_to_class ($main_class)` `[static]`

Metodo per recuperare istanze [Singleton](#) rispetto alla classe fornita.

Per ogni classe [Singleton](#) e classe fornita ritorna sempre la stessa istanza

Parameters

string	<i>\$main_class</i>	nome della classe che richiede l'istanza Singleton
--------	---------------------	--

Returns

object

8.133.4 Field Documentation

8.133.4.1 `$_instances = array()` `[static]`, `[protected]`

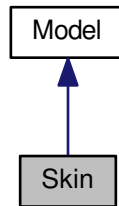
The documentation for this class was generated from the following file:

- [lib/classes/class.Singleton.php](#)

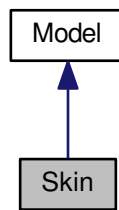
8.134 Skin Class Reference

Libreria per la gestione delle [Skin](#).

Inheritance diagram for Skin:



Collaboration diagram for Skin:



Public Member Functions

- [__construct](#) (\$id)
Costruttore.
- [sortUp](#) ()
Aumenta di 1 la priorità della skin e scala le altre.
- [formSkin](#) ()
Form per la creazione e la modifica di una skin.
- [actionSkin](#) (\Gino\Http\Request \$request)
Processa il form di inserimento e modifica di una skin.
- [formDelSkin](#) ()
Form per l'eliminazione di una skin.
- [actionDelSkin](#) (\Gino\Http\Request \$request)
Processa il form di eliminazione di una skin

Static Public Member Functions

- static [columns](#) ()
- static [getSkin](#) (\Gino\Http\Request \$request)
Recupera la skin corrispondente all'url della request.

- static [removeCss](#) (\$id)
Elimina dalla tabella delle skin il riferimento a uno specifico file css.
- static [removeTemplate](#) (\$id)
Elimina dalla tabella delle skin il riferimento a uno specifico template.
- static [newSkinPriority](#) ()
Priorità di una nuova skin.
- static [layoutInfo](#) ()
Descrizione della procedura.

Static Public Attributes

- static [\\$table](#) = 'sys_layout_skin'
- static [\\$columns](#)

Private Attributes

- [\\$_interface](#)

Additional Inherited Members

8.134.1 Detailed Description

Libreria per la gestione delle [Skin](#).

Le [Skin](#) sono l'unione di un template, un css (opzionale), e delle rules che permettono di associarle ad un url. Dato un url il sistema ricava la skin associata ed utilizza il template per generare il documento html completo.

See also

[Gino.App.Layout](#)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.134.2 Constructor & Destructor Documentation

8.134.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.Skin](#)

8.134.3 Member Function Documentation

8.134.3.1 `actionDelSkin (\Gino\Http\Request $request)`

Processa il form di eliminazione di una skin

See also

`self::formDelSkin()`

Parameters

<code>\Gino\← Http\← Request</code>	<code>\$request</code>	istanza di Gino.Request
---	------------------------	-------------------------

Returns

[Gino.Http.Response](#)

8.134.3.2 `actionSkin (\Gino\Http\Request $request)`

Processa il form di inserimento e modifica di una skin.

See also

`self::formSkin()`

Parameters

<code>\Gino\← Http\← Request</code>	<code>\$request</code>	istanza di Gino.Request
---	------------------------	-------------------------

Returns

[Gino.Http.Response](#)

8.134.3.3 `static columns () [static]`

8.134.3.4 `formDelSkin ()`

[Form](#) per l'eliminazione di una skin.

Returns

codice html form

8.134.3.5 `formSkin ()`

[Form](#) per la creazione e la modifica di una skin.

Returns

codice html form

8.134.3.6 `static getSkin (\Gino\Http\Request $request) [static]`

Recupera la skin corrispondente all'url della request.

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	
------------------------------	------------------	--

Returns

skin trovata oppure FALSE

Variabile di sessione -> urls -> rexp

Urls

Rexp

8.134.3.7 static layoutInfo () [static]

Descrizione della procedura.

Returns

informazioni, codice html

8.134.3.8 static newSkinPriority () [static]

Priorità di una nuova skin.

Returns

priorità

8.134.3.9 static removeCss (\$id) [static]

Elimina dalla tabella delle skin il riferimento a uno specifico file css.

Parameters

integer	<i>\$id</i>	valore ID del css associato alla skin
---------	-------------	---------------------------------------

Returns

risultato operazione, bool

8.134.3.10 static removeTemplate (\$id) [static]

Elimina dalla tabella delle skin il riferimento a uno specifico template.

Parameters

integer	<i>\$id</i>	valore ID del template associato alla skin
---------	-------------	--

Returns

risultato operazione, bool

8.134.3.11 `sortUp ()`

Aumenta di 1 la priorità della skin e scala le altre.

Returns

risultato operazione, bool

8.134.4 Field Documentation

8.134.4.1 `$_interface` [`private`]

8.134.4.2 `$columns` [`static`]

8.134.4.3 `$table = 'sys_layout_skin'` [`static`]

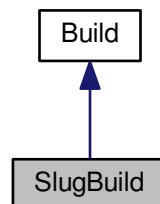
The documentation for this class was generated from the following file:

- [lib/classes/class.Skin.php](#)

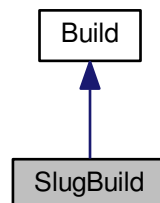
8.135 SlugBuild Class Reference

Getisce i campi di tipo SLUG (CHAR, VARCHAR)

Inheritance diagram for SlugBuild:



Collaboration diagram for SlugBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [formElement](#) (\Gino\Form \$form, \$options)
- [clean](#) (\$options=null, \$id=null)

Data Fields

- [\\$_trnsI](#)

Protected Attributes

- [\\$_autofill](#)
Proprietà dei campi specifiche del tipo di campo.

8.135.1 Detailed Description

Getisce i campi di tipo SLUG (CHAR, VARCHAR)

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.135.2 Constructor & Destructor Documentation

8.135.2.1 [__construct](#) (*\$options*)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.135.3 Member Function Documentation

8.135.3.1 [clean](#) (*\$options = null*, *\$id = null*)

See also

[Gino.Build::clean\(\)](#) Controlla la preesistenza del valore nei record della tabella

8.135.3.2 formElement (\Gino\Form \$form, \$options)

See also

[Gino.Build::formElement\(\)](#) Aggiunge il codice javascript che permette l'autoriempimento del campo se è stata passata l'opzione autofill.

8.135.4 Field Documentation

8.135.4.1 \$_autofill [protected]

Proprietà dei campi specifiche del tipo di campo.

8.135.4.2 \$_trnsi

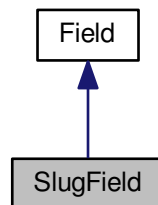
The documentation for this class was generated from the following file:

- lib/classes/build/class.SlugBuild.php

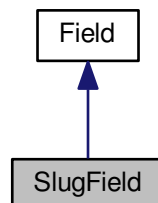
8.136 SlugField Class Reference

Campo di tipo SLUG (CHAR, VARCHAR)

Inheritance diagram for SlugField:



Collaboration diagram for SlugField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)

Data Fields

- [\\$_trnsI](#)

Protected Attributes

- [\\$_autofill](#)
Proprietà dei campi specifiche del tipo di campo.

8.136.1 Detailed Description

Campo di tipo SLUG (CHAR, VARCHAR)

I campi slug sono utilizzati per l'inserimento della parte caratterizzante di un pretty url, in genere utilizzato al posto dell'id per identificare un oggetto

Copyright

2014-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.136.2 Constructor & Destructor Documentation

8.136.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – autofill (string array): nome o array di nomi dei campi da utilizzare per calcolare lo slug. Se vengono dati più campi vengono concatenati con un dash '-'
-------	------------------	--

8.136.3 Member Function Documentation

8.136.3.1 `getProperties ()`

See also

[Gino.Field::getProperties\(\)](#)

8.136.3.2 `valueFromDb ($value)`

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.136.4 Field Documentation

8.136.4.1 `$_autofill` [`protected`]

Proprietà dei campi specifiche del tipo di campo.

8.136.4.2 `$_trnsI`

The documentation for this class was generated from the following file:

- `lib/classes/fields/class.SlugField.php`

8.137 sqlcode Class Reference

Contiene query e codice sql personalizzato.

Public Member Functions

- `__construct ()`
- `admin_actionUser` (`$table1`, `$table2`, `$value1`, `$value2`, `$value3`, `$value4`)

Private Attributes

- `$_dbms`

8.137.1 Detailed Description

Contiene query e codice sql personalizzato.

Copyright

2013 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

Ogni query e ogni porzione di codice sql deve essere inserita in un metodo il cui nome, per praticità, riporta le sue coordinate ed è così composto:

nome_classe + _ + nome_metodo

Esempio

```
include_once(LIB_DIR.OS."sqlcode.php");
$obj = new sqlcode();
$call = get_class().'_methodName';
$query = $obj->$call($param1, $param2, $param3);
```

8.137.2 Constructor & Destructor Documentation**8.137.2.1 __construct ()****8.137.3 Member Function Documentation****8.137.3.1 admin_actionUser (\$table1, \$table2, \$value1, \$value2, \$value3, \$value4)****8.137.4 Field Documentation****8.137.4.1 \$dbms [private]**

The documentation for this class was generated from the following file:

- [lib/sqlcode.php](#)

8.138 SqlParse Class Reference

Classe per gestire il parser dei file sql (Code freely adapted from phpBB Group)

Static Public Member Functions

- static [remove_comments](#) (&\$output)
Strip the sql comment lines out of an uploaded sql file specifically for mssql and postgres type files in the install.
- static [remove_remarks](#) (\$sql)
Strip the sql comment lines out of an uploaded sql file.
- static [split_sql_file](#) (\$sql, \$delimiter)
Split an uploaded sql file into single sql statements.
- static [getQueries](#) (\$options=array())
Elenco delle singole query presenti in un file sql.

8.138.1 Detailed Description

Classe per gestire il parser dei file sql (Code freely adapted from phpBB Group)

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.138.2 Member Function Documentation

8.138.2.1 static `getQueries ($options = array())` [static]

Elenco delle singole query presenti in un file sql.

Parameters

string	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• file_schema (string): percorso al file sql• content_schema (string): contenuto del file sql
--------	------------------	---

Returns

array (query)

8.138.2.2 static `remove_comments (& $output)` [static]

Strip the sql comment lines out of an uploaded sql file specifically for mssql and postgres type files in the install.

Parameters

string	<i>\$output</i>	
--------	-----------------	--

Returns

string

8.138.2.3 static `remove_remarks ($sql)` [static]

Strip the sql comment lines out of an uploaded sql file.

Parameters

string	<i>\$sql</i>	file contents
--------	--------------	---------------

Returns

array

8.138.2.4 static `split_sql_file ($sql, $delimiter)` [static]

Split an uploaded sql file into single sql statements.

Note: expects trim() to have already been run on \$sql.

Parameters

string	<i>\$sql</i>	file contents
string	<i>\$delimiter</i>	delimiter

Returns

array

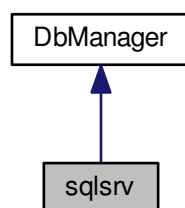
The documentation for this class was generated from the following file:

- lib/classes/[class.Db.php](#)

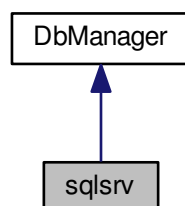
8.139 sqlsrv Class Reference

Libreria di connessione ai database SQL Server.

Inheritance diagram for sqlsrv:



Collaboration diagram for sqlsrv:



Public Member Functions

- [__construct](#) (\$params)
Costruttore.
- [getInfoQuery](#) ()

- [openConnection \(\)](#)
- [closeConnection \(\)](#)
- [begin \(\)](#)
- [rollback \(\)](#)

Per tabelle innodb.

- [commit \(\)](#)
- *Per tabelle innodb.*
- [multiActionQuery \(\\$file_content\)](#)
- [freeresult \(\\$res=null\)](#)
- [getLastId \(\\$table\)](#)
- [autoIncValue \(\\$table\)](#)

Ottiene il valore del campo AUTO_INCREMENT.

- [getFieldFromId \(\\$table, \\$field, \\$field_id, \\$id, \\$options=array\(\)\)](#)
- [tableexists \(\\$table\)](#)
- [fieldInformations \(\\$table\)](#)
- [conformFieldType \(\\$type\)](#)

Uniforma i tipi di dato dei campi.

- [limit \(\\$range, \\$offset=0\)](#)
- [distinct \(\\$fields, \\$options=array\(\)\)](#)
- [concat \(\\$sequence\)](#)
- [dumpDatabase \(\\$file\)](#)
- [getTableStructure \(\\$table\)](#)
- [changeFieldType \(\\$data_type, \\$value\)](#)
- [getFieldsName \(\\$table\)](#)
- [getNumRecords \(\\$table, \\$where=null, \\$field='id', \\$options=array\(\)\)](#)
- [query \(\\$fields, \\$tables, \\$where=null, \\$options=array\(\)\)](#)
- [execCustomQuery \(\\$query, \\$options=array\(\)\)](#)
- [select \(\\$fields, \\$tables, \\$where=null, \\$options=array\(\)\)](#)
- [insert \(\\$fields, \\$table, \\$debug=false\)](#)
- [update \(\\$fields, \\$table, \\$where, \\$debug=false\)](#)
- [delete \(\\$table, \\$where, \\$debug=false\)](#)
- [drop \(\\$table\)](#)
- [columnHasValue \(\\$table, \\$field, \\$value, \\$options=array\(\)\)](#)
- [join \(\\$table, \\$condition, \\$option\)](#)
- [union \(\\$queries, \\$options=array\(\)\)](#)
- [restore \(\\$table, \\$filename, \\$options=array\(\)\)](#)
- [dump \(\\$table, \\$filename, \\$options=array\(\)\)](#)
- [escapeString \(\\$string\)](#)

Data Fields

- [\\$_db_name](#)
- [\\$_db_user](#)
- [\\$_db_password](#)
- [\\$_db_charset](#)
- [\\$_dbconn](#)

Private Member Functions

- [setParamsCache](#) ()
Parametri inerenti la cache delle query presenti nelle Impostazioni di sistema.
- [setSql](#) (\$sql_query)
Imposta la query come proprietà
- [setNumberRows](#) (\$numberresults)
- [setconnection](#) (\$connection)
- [queryResults](#) (\$query, \$options=array())
Esegue la query e ne calcola i tempi di esecuzione.
- [listTables](#) ()
- [getDataType](#) (\$info)
Ricava il tipo di dato.
- [getFieldLength](#) (\$info)
Ricava il numero di caratteri di un campo.
- [getConstraintType](#) (\$column, \$table)
Verifica se una colonna è una chiave.
- [getCheckConstraint](#) (\$column, \$table)
Verifica se una colonna ha un vincolo CHECK e ne ricava il vincolo.
- [getInformationKey](#) (\$column, \$table, \$key=null)
Verifica se un campo è una particolare chiave e, in caso positivo, ne recupera il nome.
- [limitQuery](#) (\$fields, \$tables, \$where=null, \$options=array())

Private Attributes

- [\\$_db_host](#)
- [\\$_sql](#)
- [\\$_qry](#)
- [\\$_numberrows](#)
- [\\$_connection](#)
- [\\$_rows](#)
- [\\$_affected](#)
- [\\$_lastid](#)
- [\\$_dbresults](#) = array()
- [\\$_show_stats](#)
- [\\$_info_queries](#)
- [\\$_cnt](#)
- [\\$_time_queries](#)
- [\\$_query_cache](#)
- [\\$_query_cache_time](#)
- [\\$_cache](#)

8.139.1 Detailed Description

Libreria di connessione ai database SQL Server.

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

GESTIONE CODIFICA UTF8

In SQL Server occorre gestire la codifica UTF8 dei dati.

GESTIONE DATABASE/HTML

Dal database alla visualizzazione

In questo caso si passa attraverso il metodo [Gino.convertToHtml\(\)](#) richiamato dai metodi `htmlChars`, `htmlCharsText`, `htmlInput`, `htmlInputEditor` presenti nel file `func.var.php`.

Dal form al database

I dati passano attraverso il metodo [Gino.convertToDatabase\(\)](#) (file `func.var.php`) richiamato direttamente dalle librerie di connessione al database.

8.139.2 Constructor & Destructor Documentation

8.139.2.1 `__construct ($params)`

Costruttore.

Parameters

array	<i>\$params</i>	parametri di connessione al database <ul style="list-style-type: none"> • host (string): nome del server • db_name (string): nome del database • user (string): utente che si connette • password (string): password dell'utente che si connette • charset (string): encoding • connect (boolean): attiva la connessione
-------	-----------------	--

Returns

void

Implements [DbManager](#).

8.139.3 Member Function Documentation

8.139.3.1 `autoIncValue ($table)`

Ottiene il valore del campo `AUTO_INCREMENT`.

See also

[DbManager::autoIncValue\(\)](#)

Implements [DbManager](#).

8.139.3.2 begin ()

See also

[DbManager::begin\(\)](#)

Implements [DbManager](#).

8.139.3.3 changeFieldType (*\$data_type*, *\$value*)

See also

[Gino.DbManager::changeFieldType\(\)](#)

Implements [DbManager](#).

8.139.3.4 closeConnection ()

See also

[DbManager::closeConnection\(\)](#)

Implements [DbManager](#).

8.139.3.5 columnHasValue (*\$table*, *\$field*, *\$value*, *\$options* = `array()`)

See also

[DbManager::columnHasValue\(\)](#)

Implements [DbManager](#).

8.139.3.6 commit ()

Per tabella innodb.

See also

[DbManager::commit\(\)](#)

Implements [DbManager](#).

8.139.3.7 concat (*\$sequence*)

See also

[DbManager::concat\(\)](#)

Implements [DbManager](#).

8.139.3.8 conformFieldType (*\$type*)

Uniforma i tipi di dato dei campi.

Parameters

integer	<i>\$type</i>
---------	---------------

Returns

string

Tipi di dato riportati dalla funzione `sqlsrv_field_metadata()`

Per l'elenco fare riferimento alla documentazione ufficiale Microsoft:

<http://msdn.microsoft.com/en-us/library/cc296197.aspx>

bigint	-5
binary	-2
bit	-7
char	1
date	91
datetime	93
datetime2	93
datetimeoffset	-155
decimal	3
float	6
image	-4
int	4
money	3
nchar	-8
ntext	-10
numeric	2
nvarchar	-9
real	7
smalldatetime	93
smallint	5
Smallmoney	3
text	-1
time	-154
timestamp	-2
tinyint	-6
udt	-151
uniqueidentifier	-11
varbinary	-3
varchar	12
xml	-152

Implements [DbManager](#).

8.139.3.9 `delete ($table, $where, $debug = false)`

See also

[DbManager::delete\(\)](#)

Implements [DbManager](#).

8.139.3.10 `distinct ($fields, $options = array())`

See also

[DbManager::distinct\(\)](#)

Implements [DbManager](#).

8.139.3.11 `drop ($table)`

See also

[DbManager::drop\(\)](#)

Implements [DbManager](#).

8.139.3.12 `dump ($table, $filename, $options = array())`

See also

[DbManager::dump\(\)](#)

Implements [DbManager](#).

8.139.3.13 `dumpDatabase ($file)`

See also

[DbManager::dumpDatabase\(\)](#)
[listTables\(\)](#)

Implements [DbManager](#).

8.139.3.14 `escapeString ($string)`

See also

[DbManager::escapeString\(\)](#)

Implements [DbManager](#).

8.139.3.15 `execCustomQuery ($query, $options = array())`

See also

[Gino.DbManager::execCustomQuery\(\)](#)

Implements [DbManager](#).

8.139.3.16 `fieldInformations ($table)`

See also

[DbManager::fieldInformations\(\)](#)
[conformFieldType\(\)](#)

La funzione `sqlsrv_field_metadata()` ritorna i riferimenti:

Name	The name of the field.
Type	The numeric value for the SQL type.
Size	The number of characters for fields of character type, the number of bytes for fields of binary type, or NULL for other types.
Precision	The precision for types of variable precision, NULL for other types.
Scale	The scale for types of variable scale, NULL for other types.
Nullable	An enumeration indicating whether the column is nullable, not nullable, or if it is not known.

Implements [DbManager](#).

8.139.3.17 `freeresult ($res = null)`

See also

[DbManager::freeresult\(\)](#)

Implements [DbManager](#).

8.139.3.18 `getCheckConstraint ($column, $table)` [private]

Verifica se una colonna ha un vincolo CHECK e ne ricava il vincolo.

Parameters

string	<i>\$column</i>	
string	<i>\$table</i>	

Returns

string, ([no_admin]='no' OR [no_admin]='yes')

8.139.3.19 `getConstraintType ($column, $table)` [private]

Verifica se una colonna è una chiave.

Parameters

string	<i>\$column</i>	
string	<i>\$table</i>	

Returns

array(key, name)

Recupera il nome della chiave:

- *PRIMARYKEY*

- *UNIQUE*
- *FOREIGN KEY* e il suo nome (ad es. PK__page_ent__3213E83FAC2C67F3, UQ__page_ent__32DD1E4C35F37AB2)

8.139.3.20 `getDataType ($info)` [private]

Ricava il tipo di dato.

Il tipo di dato deve essere compatibile con quelli definiti in `Model::dataType()`.

Parameters

array	<i>\$info</i>	
-------	---------------	--

Returns

string

8.139.3.21 `getFieldFromId ($table, $field, $field_id, $id, $options = array())`

See also

[DbManager::getFieldFromId\(\)](#)

Implements [DbManager](#).

8.139.3.22 `getFieldLength ($info)` [private]

Ricava il numero di caratteri di un campo.

Parameters

array	<i>\$info</i>	
-------	---------------	--

Returns

integer

8.139.3.23 `getFieldsName ($table)`

See also

[DbManager::getFieldsName\(\)](#)
[freeresult\(\)](#)

Implements [DbManager](#).

8.139.3.24 `getInfoQuery ()`

See also

[DbManager::getInfoQuery\(\)](#)

Implements [DbManager](#).

8.139.3.25 `getInformationKey ($column, $table, $key = null)` `[private]`

Verifica se un campo è una particolare chiave e, in caso positivo, ne recupera il nome.

Parameters

string	<i>\$column</i>	
string	<i>\$table</i>	
string	<i>\$key</i>	nome della chiave da ricercare <ul style="list-style-type: none"> • <i>PRI</i>, primary key • <i>UNI</i>, unique key • <i>FOR</i>, foreign key

Returns

mixed

8.139.3.26 getLastId (*\$table*)

See also

[DbManager::getLastId\(\)](#)Implements [DbManager](#).8.139.3.27 getNumRecords (*\$table*, *\$where* = null, *\$field* = 'id', *\$options* = array())

See also

[DbManager::getNumRecords\(\)](#)Implements [DbManager](#).8.139.3.28 getTableStructure (*\$table*)

See also

[DbManager::getTableStructure\(\)](#)
[getConstraintType\(\)](#)
[getDataType\(\)](#)
[getFieldLength\(\)](#)
[getCheckConstraint\(\)](#)
[getInformationKey\(\)](#)

Implements [DbManager](#).8.139.3.29 insert (*\$fields*, *\$table*, *\$debug* = false)

See also

[DbManager::insert\(\)](#)Implements [DbManager](#).8.139.3.30 join (*\$table*, *\$condition*, *\$option*)

See also

[DbManager::join\(\)](#)

Implements [DbManager](#).

8.139.3.31 `limit ($range, $offset = 0)`

See also

[Gino.DbManager::limit\(\)](#)

Examples

```
//Returning the first 100 rows from a table called employee:  
select top 100 * from employee  
//Returning the top 20% of rows from a table called employee:  
select top 20 percent * from employee
```

Implements [DbManager](#).

8.139.3.32 `limitQuery ($fields, $tables, $where = null, $options = array())` [private]

8.139.3.33 `listTables ()` [private]

8.139.3.34 `multiActionQuery ($file_content)`

See also

[Gino.DbManager::multiActionQuery\(\)](#)

[Gino.SqlParse::getQueries\(\)](#)

Implements [DbManager](#).

8.139.3.35 `openConnection ()`

See also

[DbManager::openConnection\(\)](#)

L'estensione SQLSRV restituisce oggetti DateTime.

È possibile disabilitare la funzione utilizzando l'opzione di connessione ReturnDatesAsStrings.

See also

<https://msdn.microsoft.com/en-us/library/ff628167.aspx>

Implements [DbManager](#).

8.139.3.36 `query ($fields, $tables, $where = null, $options = array())`

See also

[DbManager::query\(\)](#)

[limitQuery\(\)](#)

Implements [DbManager](#).

8.139.3.37 `queryResults ($query, $options = array())` [private]

Esegue la query e ne calcola i tempi di esecuzione.

Parameters

string	<i>\$query</i>	query da eseguire
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • statement (string): tipologia di query <ul style="list-style-type: none"> – <i>select</i> (default) – <i>action</i>

Returns

mixed

See also

<http://msdn.microsoft.com/en-us/library/hh487160.aspx>
8.139.3.38 `restore ($table, $filename, $options = array())`

See also

[DbManager::restore\(\)](#)
Implements [DbManager](#).
8.139.3.39 `rollback ()`

Per tabelle innodb.

See also

[DbManager::rollback\(\)](#)
Implements [DbManager](#).
8.139.3.40 `select ($fields, $tables, $where = null, $options = array())`

See also

[DbManager::select\(\)](#)
Implements [DbManager](#).
8.139.3.41 `setconnection ($connection)` [private]

8.139.3.42 `setNumberRows ($numberresults)` [private]

8.139.3.43 `setParamsCache ()` [private]

Parametri inerenti la cache delle query presenti nelle Impostazioni di sistema.

Returns

void

8.139.3.44 `setsql($sql_query)` [private]

Imposta la query come proprietà

Parameters

string	<i>\$sql_query</i>	query
--------	--------------------	-------

8.139.3.45 `tableexists ($table)`

See also

[DbManager::tableexists\(\)](#)Implements [DbManager](#).8.139.3.46 `union ($queries, $options = array())`

See also

[DbManager::union\(\)](#)

In mssql possono essere utilizzati gli operatori:

- UNION, elimina le righe duplicate dai risultati combinati delle istruzioni SELECT
- UNION ALL, mostra i record duplicati

Implements [DbManager](#).8.139.3.47 `update ($fields, $table, $where, $debug = false)`

See also

[DbManager::update\(\)](#)Implements [DbManager](#).

8.139.4 Field Documentation

8.139.4.1 `$_affected` [private]8.139.4.2 `$_cache` [private]8.139.4.3 `$_cnt` [private]8.139.4.4 `$_connection` [private]8.139.4.5 `$_db_charset`8.139.4.6 `$_db_host` [private]8.139.4.7 `$_db_name`8.139.4.8 `$_db_password`8.139.4.9 `$_db_user`

8.139.4.10 `$_dbconn`

8.139.4.11 `$_dbresults = array()` [private]

8.139.4.12 `$_info_queries` [private]

8.139.4.13 `$_lastid` [private]

8.139.4.14 `$_numberrows` [private]

8.139.4.15 `$_qry` [private]

8.139.4.16 `$_query_cache` [private]

8.139.4.17 `$_query_cache_time` [private]

8.139.4.18 `$_rows` [private]

8.139.4.19 `$_show_stats` [private]

8.139.4.20 `$_sql` [private]

8.139.4.21 `$_time_queries` [private]

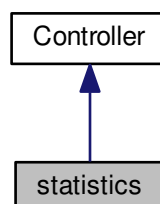
The documentation for this class was generated from the following file:

- [lib/plugin/plugin.sqlsrv.php](#)

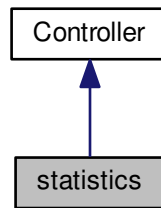
8.140 statistics Class Reference

Statistiche degli accessi all'area privata.

Inheritance diagram for statistics:



Collaboration diagram for statistics:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [manageStatistics](#) (\Gino\Http\Request \$request)
Interfaccia di visualizzazione delle statistiche.
- [export](#) ()
Esportazione delle statistiche sugli accessi all'area privata.

Private Member Functions

- [viewlogAccess](#) ()
Statistiche sugli accessi all'area privata.
- [info](#) ()
Pagina di informazioni.

Private Attributes

- [\\$_tbl_log_access](#)

Additional Inherited Members

8.140.1 Detailed Description

Statistiche degli accessi all'area privata.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.140.2 Constructor & Destructor Documentation

8.140.2.1 `__construct ()`

Costruttore.

Returns

istanza di [Gino.App.Statistics.statistics](#)

8.140.3 Member Function Documentation

8.140.3.1 `export ()`

Esportazione delle statistiche sugli accessi all'area privata.

Returns

[Gino.Http.ResponseFile](#)

8.140.3.2 `info ()` `[private]`

Pagina di informazioni.

Returns

html, informazioni

8.140.3.3 `manageStatistics (\Gino\Http\Request $request)`

Interfaccia di visualizzazione delle statistiche.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Response](#)

8.140.3.4 `viewlogAccess ()` `[private]`

Statistiche sugli accessi all'area privata.

Returns

html, statistiche

8.140.4 Field Documentation

8.140.4.1 `$_tbl_log_access` `[private]`

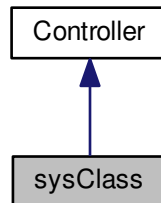
The documentation for this class was generated from the following file:

- [app/statistics/class_statistics.php](#)

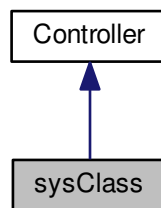
8.141 sysClass Class Reference

Classe di tipo [Gino.Controller](#) per la gestione dei moduli di sistema.

Inheritance diagram for sysClass:



Collaboration diagram for sysClass:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [manageSysClass](#) (\Gino\Http\Request \$request)
Interfaccia amministrativa per la gestione dei moduli di sistema.
- [actionInsertSysClass](#) (\Gino\Http\Request \$request)
Processa il form di installazione modulo di sistema Il modulo viene attivato durante l'installazione.
- [actionManualSysClass](#) (\Gino\Http\Request \$request)
Processa il form di installazione manuale di un modulo di sistema.
- [actionEditSysClass](#) (\Gino\Http\Request \$request)
Processa il form di modifica di un modulo di sistema.
- [actionEditSysClassActive](#) (\Gino\Http\Request \$request)
Processa il form di modifica dello stato dell'attivazione del modulo.
- [actionUpgradeSysClass](#) (\Gino\Http\Request \$request)
Processa il form di aggiornamento del modulo.
- [actionRemoveSysClass](#) (\Gino\Http\Request \$request)
Processa il form di eliminazione di un modulo di sistema.

Data Fields

- const `PACKAGE_EXTENSION` = 'zip'

Private Member Functions

- `sysClassList ()`
Elenco dei moduli di sistema.
- `formInsertSysClass ()`
Form di installazione di un modulo di sistema.
- `formManualSysClass ()`
Form per l'installazione manuale di un modulo di sistema.
- `formEditSysClass ($id)`
Form di modifica di un modulo di sistema.
- `formActivateSysClass ($id)`
Form di attivazione del modulo.
- `formUpgradeSysClass ($id)`
Form di aggiornamento del modulo.
- `upgradeFolders ($files_dir, $module_dir, $noCopyFiles)`
Aggiunge e sovrascrive files nella directory del modulo durante il processo di upgrade.
- `formRemoveSysClass ($id)`
Form di eliminazione di un modulo di sistema.
- `info ()`
Informazioni modulo.

Private Attributes

- `$_title`

Additional Inherited Members

8.141.1 Detailed Description

Classe di tipo `Gino.Controller` per la gestione dei moduli di sistema.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.141.2 Constructor & Destructor Documentation

8.141.2.1 `__construct ()`

Costruttore.

Returns

istanza di `Gino.App.SysClass.sysclass`

8.141.3 Member Function Documentation

8.141.3.1 `actionEditSysClass (\Gino\Http\Request $request)`

Processa il form di modifica di un modulo di sistema.

See also

`self::formEditSysClass()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.141.3.2 `actionEditSysClassActive (\Gino\Http\Request $request)`

Processa il form di modifica dello stato dell'attivazione del modulo.

See also

`self::formActivateSysClass()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.141.3.3 `actionInsertSysClass (\Gino\Http\Request $request)`

Processa il form di installazione modulo di sistema Il modulo viene attivato durante l'installazione.

See also

`self::formInsertSysClass()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.141.3.4 `actionManualSysClass (\Gino\Http\Request $request)`

Processa il form di installazione manuale di un modulo di sistema.

Il modulo viene attivato nell'installazione

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.141.3.5 actionRemoveSysClass (\Gino\Http\Request *\$request*)

Processa il form di eliminazione di un modulo di sistema.

See also

`self::formRemoveSysClass()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.141.3.6 actionUpgradeSysClass (\Gino\Http\Request *\$request*)

Processa il form di aggiornamento del modulo.

See also

`self::formUpgradeSysClass()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Redirect](#)

8.141.3.7 formActivateSysClass (*\$id*) [private]

[Form](#) di attivazione del modulo.

Parameters

int	<i>\$id</i>	valore ID del modulo
-----	-------------	----------------------

Returns

html, form

8.141.3.8 `formEditSysClass ($id) [private]`

[Form](#) di modifica di un modulo di sistema.

Parameters

integer	<i>\$id</i>	valore ID del modulo
---------	-------------	----------------------

Returns

html, form

8.141.3.9 formInsertSysClass () [private]

[Form](#) di installazione di un modulo di sistema.

Returns

html, form

8.141.3.10 formManualSysClass () [private]

[Form](#) per l'installazione manuale di un modulo di sistema.

Returns

html, form

8.141.3.11 formRemoveSysClass (\$id) [private]

[Form](#) di eliminazione di un modulo di sistema.

Parameters

int	<i>\$id</i>	valore ID del modulo
-----	-------------	----------------------

Returns

html, form

8.141.3.12 formUpgradeSysClass (\$id) [private]

[Form](#) di aggiornamento del modulo.

Parameters

int	<i>\$id</i>	valore ID del modulo
-----	-------------	----------------------

Returns

html, form

8.141.3.13 info () [private]

Informazioni modulo.

Returns

html, informazioni

8.141.3.14 `manageSysClass (\Gino\Http\Request $request)`

Interfaccia amministrativa per la gestione dei moduli di sistema.

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Response](#)

8.141.3.15 sysClassList () [private]

Elenco dei moduli di sistema.

Parameters

integer	<code>\$sel_id</code>	valore ID del modulo selezionato
---------	-----------------------	----------------------------------

Returns

html, elenco moduli

8.141.3.16 upgradeFolders (\$files_dir, \$module_dir, \$noCopyFiles) [private]

Aggiunge e sovrascrive files nella directory del modulo durante il processo di upgrade.

Parameters

string	<code>\$files_dir</code>	
string	<code>\$module_dir</code>	
array	<code>\$noCopyFiles</code>	array di nommi di files da non copiare/sovrascrivere

Returns

risultato operazione, bool

8.141.4 Field Documentation

8.141.4.1 \$_title [private]

8.141.4.2 const PACKAGE_EXTENSION = 'zip'

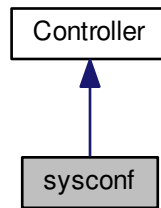
The documentation for this class was generated from the following file:

- [app/sysClass/class_sysClass.php](#)

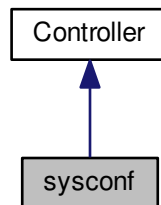
8.142 sysconf Class Reference

Classe di tipo [Gino.Controller](#) per la gestione delle principali impostazioni di sistema.

Inheritance diagram for sysconf:



Collaboration diagram for sysconf:



Public Member Functions

- [__construct](#) ()
Costruttore.
- [manageSysconf](#) (\Gino\Http\Request \$request)
Interfaccia di amministrazione modulo.

Additional Inherited Members

8.142.1 Detailed Description

Classe di tipo [Gino.Controller](#) per la gestione delle principali impostazioni di sistema.

Le impostazioni di sistema sono:

- Gestione lingue
- Lingua di default (se la gestione lingue è disattivata)
- Log degli accessi
- Metodo criptazione password

- Descrizione sito
- Parole chiave sito
- Titolo sito
- Abilita la cache di contenuti e dati
- Abilita la cache query
- Codice google analytics (es. UA-1234567-1)
- Chiave pubblica reCAPTCHA
- Chiave privata reCAPTCHA
- Chiave pubblica ShareThis
- Shortname Disqus
- Email amministratore di sistema
- Email invio automatico comunicazioni
- Ottimizzazione per dispositivi mobili (Palmari, Iphone)
- Contenuto file robots.txt

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.142.2 Constructor & Destructor Documentation

8.142.2.1 __construct ()

Costruttore.

Returns

istanza di [Gino.App.Sysconf.sysconf](#)

8.142.3 Member Function Documentation

8.142.3.1 manageSysconf (\Gino\Http\Request \$request)

Interfaccia di amministrazione modulo.

Parameters

\Gino\↔ Http\↔ Request	\$request	istanza di Gino.Http.Request
------------------------------	-----------	--

Returns

[Gino.Http.Response](#)

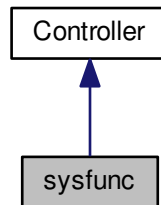
The documentation for this class was generated from the following file:

- [app/sysconf/class_sysconf.php](#)

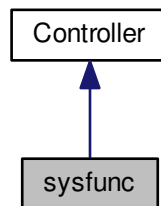
8.143 sysfunc Class Reference

Classe di tipo [Gino.Controller](#) che gestisce metodi che stampano pagine di errore.

Inheritance diagram for sysfunc:



Collaboration diagram for sysfunc:



Public Member Functions

- [__construct](#) ()
Costruttore.

Static Public Member Functions

- static [page404](#) (\$title= "", \$message= "")
Pagina di errore contenuto non disponibile.
- static [page403](#) (\$title= "", \$message= "")
Pagina di errore contenuto forbidden.
- static [page500](#) (\$title= "", \$message= "")
Pagina di errore server error.

Additional Inherited Members

8.143.1 Detailed Description

Classe di tipo [Gino.Controller](#) che gestisce metodi che stampano pagine di errore.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.143.2 Constructor & Destructor Documentation

8.143.2.1 `__construct ()`

Costruttore.

Returns

[Gino.App.Sysfunc.sysfunc](#)

8.143.3 Member Function Documentation

8.143.3.1 `static page403 ($title = ' ', $message = ' ') [static]`

Pagina di errore contenuto forbidden.

See also

[Gino.Http.ResponseForbidden](#)
[Gino.Exception.Exception403](#)

Parameters

string	<i>\$title</i>	titolo della pagina
string	<i>\$message</i>	messaggio mostrato public

Returns

pagina di errore

8.143.3.2 `static page404 ($title = ' ', $message = ' ') [static]`

Pagina di errore contenuto non disponibile.

See also

[Gino.Http.ResponseNotFound](#)
[Gino.Exception.Exception404](#)

Parameters

string	<i>\$title</i>	titolo della pagina
string	<i>\$message</i>	messaggio mostrato

Returns

pagina di errore

8.143.3.3 `static page500 ($title = ' ', $message = ' ') [static]`

Pagina di errore server error.

See also

[Gino.Http.ResponseServerError](#)
[Gino.Exception.Exception500](#)

Parameters

string	<i>\$title</i>	titolo della pagina
string	<i>\$message</i>	messaggio mostrato public

Returns

pagina di errore

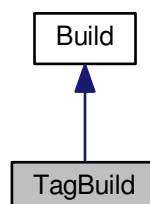
The documentation for this class was generated from the following file:

- [app/sysfunc/class_sysfunc.php](#)

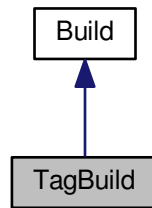
8.144 TagBuild Class Reference

Gestisce i campi per inserimento tag.

Inheritance diagram for TagBuild:



Collaboration diagram for TagBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [formElement](#) (\Gino\Form \$form, \$options)
- [clean](#) (\$options=null)
- [save](#) (\$event_name, \$params)
Salva i tag nelle tabelle dei tag e quella di associazione ai contenuti.
- [tagCloud](#) ()
Tag cloud.

Data Fields

- [\\$_model_controller_instance](#)

Protected Attributes

- [\\$_model_controller_class](#)
Proprietà dei campi specifiche del tipo di campo.

8.144.1 Detailed Description

Gestisce i campi per inserimento tag.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.144.2 Constructor & Destructor Documentation

8.144.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.144.3 Member Function Documentation

8.144.3.1 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#) Ripulisce l'input e registra un listener per salvare i tag quando il modello è stato correttamente salvato

8.144.3.2 `formElement (\Gino\Form $form, $options)`

See also

[Gino.Build::formElement\(\)](#)

8.144.3.3 `save ($event_name, $params)`

Salva i tag nelle tabelle dei tag e quella di associazione ai contenuti.

Parameters

string	<i>\$event_name</i>	nome evento
array	<i>\$param</i>	array associativo. La chiave model ha il modello appena salvato

Returns

void

8.144.3.4 `tagCloud ()`

Tag cloud.

Returns

tag cloud

8.144.4 Field Documentation

8.144.4.1 `$_model_controller_class` [protected]

Proprietà dei campi specifiche del tipo di campo.

8.144.4.2 `$_model_controller_instance`

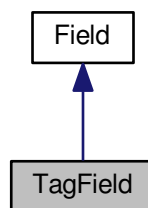
The documentation for this class was generated from the following file:

- `lib/classes/build/class.TagBuild.php`

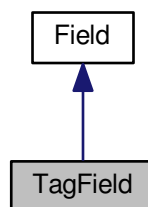
8.145 TagField Class Reference

Classe per la gestione di campi per inserimento tag.

Inheritance diagram for TagField:



Collaboration diagram for TagField:



Public Member Functions

- `__construct` (`$options`)
Costruttore.
- `getProperties` ()
- `valueFromDb` (`$value`)

Data Fields

- `$_model_controller_instance`

Protected Attributes

- [\\$_model_controller_class](#)

Proprietà dei campi specifiche del tipo di campo.

8.145.1 Detailed Description

Classe per la gestione di campi per inserimento tag.

Copyright

2014-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.145.2 Constructor & Destructor Documentation

8.145.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – model_controller_class: nome classe del controller – model_controller_instance: id istanza del controller
-------	------------------	---

8.145.3 Member Function Documentation

8.145.3.1 getProperties ()

See also

[Gino.Field::getProperties\(\)](#)

8.145.3.2 valueFromDb (\$value)

See also

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.145.4 Field Documentation

8.145.4.1 `$_model_controller_class` [protected]

Proprietà dei campi specifiche del tipo di campo.

8.145.4.2 `$_model_controller_instance`

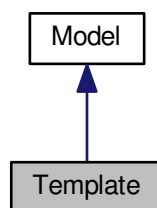
The documentation for this class was generated from the following file:

- `lib/classes/fields/class.TagField.php`

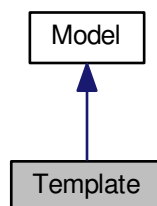
8.146 Template Class Reference

Libreria per la gestione dei template del documento html da associare alle [Gino](#).

Inheritance diagram for Template:



Collaboration diagram for Template:



Public Member Functions

- `__construct` (\$id)
Costruttore.
- `setFilename` (\$value)

- Imposta il nome del file.*
- [formFreeTemplate](#) ()
 - Form di inserimento/modifica template di tipo free.*
- [actionFreeTemplate](#) (\Gino\Http\Request \$request)
 - Processa il form di inserimento/modifica template free.*
- [formTemplate](#) ()
 - Form di inserimento/modifica di un template a blocchi.*
- [formOutline](#) ()
 - Form che introduce alla modifica dello schema dei template a blocchi.*
- [formCopyTemplate](#) ()
 - Form di duplicazione di un template a blocchi.*
- [tplBlockForm](#) (\$request=null)
 - Parte del form template a blocchi per inserimento/modifica di un blocco.*
- [addBlockForm](#) (\$ref=null, \$request=null)
 - Form di aggiunta blocchi.*
- [formDelTemplate](#) ()
 - Form di eliminazione di un template.*
- [actionDelTemplate](#) (\Gino\Http\Request \$request)
 - Processa il form di eliminazione di un template.*
- [manageTemplate](#) (\$css, \$tpl_id=0)
 - Schema e modifica interattiva del template a blocchi.*
- [actionTemplate](#) (\$request)
 - Processa il form di inserimento/modifica template.*
- [actionCopyTemplate](#) (\Gino\Http\Request \$request)
 - Processa il form di duplicazione template a blocchi.*

Static Public Member Functions

- static [columns](#) ()
 - Struttura dei campi della tabella di un modello.*
- static [layoutInfo](#) ()
 - Descrizione della procedura.*

Data Fields

- [\\$_interface](#)
- [\\$_blocks_properties](#)

Static Public Attributes

- static [\\$table](#) = 'sys_layout_tpl'
- static [\\$columns](#)

Private Member Functions

- [initBlocksProperties](#) ()
 - Imposta le proprietà dei blocchi*
- [formData](#) (\$gform, \$free=FALSE)
 - For di inserimento/modifica dati template.*
- [formBlock](#) (\$gform)

Parte del form di template a blocchi per la scelta/visualizzazione del numero di blocchi.

- [createTemplate](#) (\$blocks_number, \$template=)
Creazione template interfaccia interattiva.
- [printBlock](#) (\$num, \$align, \$rows, \$cols, \$um, \$width, \$pos=0, \$template=)
Creazione blocco nell'interfaccia interattiva di gestione template a blocchi.
- [renderNave](#) (\$matches)
Crea una navata nell'interfaccia interattiva di gestione template a blocchi.
- [cellCtrl](#) (\$id)
Controlli di una cella nell'interfaccia interattiva di gestione template a blocchi.
- [saveBlock](#) (\$id, \$position, \$width, \$um, \$align, \$rows, \$cols)
Salvataggio di un blocco.
- [deleteBlocks](#) ()
Eliminazione blocchi template.

Private Attributes

- [\\$_home](#)
- [\\$_blocks_number](#)
- [\\$_align_dict](#)
- [\\$_um_dict](#)

Static Private Attributes

- static [\\$table_block](#) = 'sys_layout_tpl_block'

Additional Inherited Members

8.146.1 Detailed Description

Libreria per la gestione dei template del documento html da associare alle [Gino](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.146.2 Constructor & Destructor Documentation

8.146.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.Template](#)

8.146.3 Member Function Documentation

8.146.3.1 `actionCopyTemplate (\Gino\Http\Request $request)`

Processa il form di duplicazione template a blocchi.

See also

`self::formCopyTemplate()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

8.146.3.2 `actionDelTemplate (\Gino\Http\Request $request)`

Processa il form di eliminazione di un template.

See also

`self::formDelTemplate()`
[Gino.Skin::removeTemplate\(\)](#)

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Http.Request
------------------------------	------------------	--

Returns

[Gino.Http.Response](#)

8.146.3.3 `actionFreeTemplate (\Gino\Http\Request $request)`

Processa il form di inserimento/modifica template free.

See also

`self::formFreeTemplate()`

Parameters

\Gino\↔ Http\↔ Request	<i>\$request</i>	istanza di Gino.Request
------------------------------	------------------	---

Returns

[Gino.Http.Response](#)

8.146.3.4 `actionTemplate ($request)`

Processa il form di inserimento/modifica template.

See also

[formTemplate\(\)](#)

Parameters

<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request
---	------------------------	--

Returns

[Gino.Http.Response](#)

8.146.3.5 `addBlockForm ($ref = null, $request = null)`

[Form](#) di aggiunta blocchi.

Parameters

<code>integer</code>	<code>\$ref</code>	numero del blocco nella sequenza corretta, default null
<code>\Gino\↔ Http\↔ Request</code>	<code>\$request</code>	istanza di Gino.Http.Request , default null

Returns

html

8.146.3.6 `cellCtrl ($id) [private]`

Controlli di una cella nell'interfaccia interattiva di gestione template a blocchi.

Parameters

<code>int</code>	<code>\$id</code>	id cella
------------------	-------------------	----------

Returns

html

8.146.3.7 `static columns () [static]`

Struttura dei campi della tabella di un modello.

Returns

array

8.146.3.8 `createTemplate ($blocks_number, $template = ' ') [private]`

Creazione template interfaccia interattiva.

Parameters

int	<i>\$blocks_number</i>	numero blocchi string \$template
-----	------------------------	----------------------------------

Returns

html

8.146.3.9 deleteBlocks () [private]

Eliminazione blocchi template.

Returns

risultato operazione, bool

8.146.3.10 formBlock (\$gform) [private]

Parte del form di template a blocchi per la scelta/visualizzazione del numero di blocchi.

See also

[tplBlockForm\(\)](#)

Parameters

\Gino\Form	<i>\$gform</i>	istanza di Gino.Form
------------	----------------	--------------------------------------

Returns

html

8.146.3.11 formCopyTemplate ()

[Form](#) di duplicazione di un template a blocchi.

Returns

html, form

8.146.3.12 formData (\$gform, \$free = FALSE) [private]

For di inserimento/modifica dati template.

Parameters

\Gino\Form	<i>\$gform</i>	istanza di Gino.Form
bool	<i>\$free</i>	indica se il template è di tipo free (TRUE) o a blocchi

Returns

html, form

8.146.3.13 formDelTemplate ()

[Form](#) di eliminazione di un template.

Returns

html, form

8.146.3.14 formFreeTemplate ()

[Form](#) di inserimento/modifica template di tipo free.

Returns

html, form

8.146.3.15 formOutline ()

[Form](#) che introduce alla modifica dello schema dei template a blocchi.

Returns

html, form

8.146.3.16 formTemplate ()

[Form](#) di inserimento/modifica di un template a blocchi.

See also

`self::formBlock()`

Returns

html, form

8.146.3.17 initBlocksProperties () [private]

Imposta le proprietà dei blocchi

Returns

void

8.146.3.18 static layoutInfo () [static]

Descrizione della procedura.

Returns

html, informazioni

8.146.3.19 `manageTemplate ($css, $tpl_id = 0)`

Schema e modifica interattiva del template a blocchi.

La creazione e la ricostruzione del template sono i due casi in cui si creano e si modificano i blocchi. Il metodo che lavora sui blocchi è [createTemplate\(\)](#); nel caso della modifica del template viene letto direttamente il file.

See also

`self::renderNave()`

Parameters

<code>\Gino\Css</code>	<code>\$css</code>	istanza di Gino.Css
integer	<code>\$tpl_id</code>	valore ID del template

Returns

html, interfaccia di modifica interattiva del template

8.146.3.20 `printBlock ($num, $align, $rows, $cols, $um, $width, $pos = 0, $template = ' ')` `[private]`

Creazione blocco nell'interfaccia interattiva di gestione template a blocchi.

Parameters

int	<code>\$num</code>	numero blocco
string	<code>\$align</code>	allineamento (1: sinistra, 2: centrato, 3: destra)
int	<code>\$rows</code>	numero righe
int	<code>\$cols</code>	numero colonne
string	<code>\$um</code>	unita di misura (1: px, 2: %)
int	<code>\$width</code>	larghezza
int	<code>\$pos</code>	posizione
string	<code>\$template</code>	

Returns

html

8.146.3.21 `renderNave ($matches)` `[private]`

Crea una navata nell'interfaccia interattiva di gestione template a blocchi.

Parameters

array	<code>\$matches</code>	<ul style="list-style-type: none"> • <code>\$matches[0]</code> complete matching • <code>\$matches[1]</code> match open tag, es. • <code>\$matches[3]</code> match div id, es. <code>nav_1_1</code> • <code>\$matches[4]</code> match div content, es. <code>{module classid=20 func=blockList}</code> • <code>\$matches[5]</code> match close tag, es.
-------	------------------------	--

Returns

html

8.146.3.22 `saveBlock ($id, $position, $width, $um, $align, $rows, $cols)` [private]

Salvataggio di un blocco.

Parameters

int	<i>\$id</i>	id blocco
int	<i>\$position</i>	posizione
int	<i>\$width</i>	larghezza
int	<i>\$um</i>	unità di misura (1: px, 2: %)
int	<i>\$align</i>	allineamento (1: sinistra, 2: centrato, 3: destra)
int	<i>\$rows</i>	numero righe
int	<i>\$cols</i>	numero colonne

Returns

risultato, bool

8.146.3.23 `setFilename ($value)`

Imposta il nome del file.

Parameters

string	<i>\$value</i>	nome del file dei template
--------	----------------	----------------------------

Returns

TRUE

8.146.3.24 `tplBlockForm ($request = null)`

Parte del form template a blocchi per inserimento/modifica di un blocco.

Parameters

<code>\Gino\↵ Http\↵ Request</code>	<i>\$request</i>	istanza di Gino.Http.Request default null
---	------------------	---

Returns

html

8.146.4 Field Documentation

8.146.4.1 `$_align_dict` [private]8.146.4.2 `$_blocks_number` [private]

8.146.4.3 `$_blocks_properties`

8.146.4.4 `$_home` `[private]`

8.146.4.5 `$_interface`

8.146.4.6 `$_um_dict` `[private]`

8.146.4.7 `$columns` `[static]`

8.146.4.8 `$table = 'sys_layout_tpl'` `[static]`

8.146.4.9 `$table_block = 'sys_layout_tpl_block'` `[static]`, `[private]`

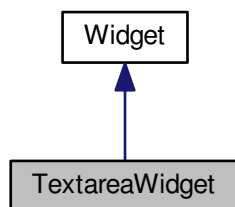
The documentation for this class was generated from the following file:

- [lib/classes/class.Template.php](#)

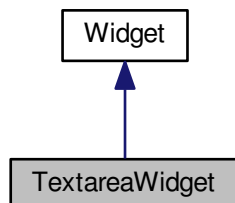
8.147 TextareaWidget Class Reference

Campi di tipo textarea nei form.

Inheritance diagram for TextareaWidget:



Collaboration diagram for TextareaWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.147.1 Detailed Description

Campi di tipo textarea nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com

abidibo abidibo@gmail.com

8.147.2 Member Function Documentation

8.147.2.1 [printInputForm](#) (*\$form*, *\$options*)

See also

[Gino.Widget::printInputForm\(\)](#)

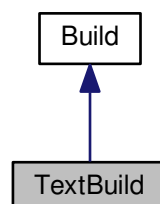
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.TextareaWidget.php](#)

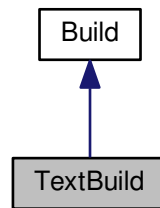
8.148 TextBuild Class Reference

Campo di tipo TEXT.

Inheritance diagram for TextBuild:



Collaboration diagram for TextBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getTrnsl](#) ()
Getter della proprietà trnsl (indica se il campo necessita di traduzione o no)
- [setTrnsl](#) (\$value)
Setter della proprietà trnsl.
- [canBeOrdered](#) ()
- [filterWhereClause](#) (\$value)
- [formElement](#) (\Gino\Form \$form, \$options)
- [clean](#) (\$options=null)

Protected Attributes

- [\\$_trnsl](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.148.1 Detailed Description

Campo di tipo TEXT.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.148.2 Constructor & Destructor Documentation

8.148.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.148.3 Member Function Documentation

8.148.3.1 `canBeOrdered ()`

See also

[Gino.Build::canBeOrdered\(\)](#)

Returns

FALSE

8.148.3.2 `clean ($options = null)`

See also

[Gino.Build::clean\(\)](#)

8.148.3.3 `filterWhereClause ($value)`

See also

[Gino.Build::filterWhereClause\(\)](#)

8.148.3.4 `formElement (\Gino\Form $form, $options)`

See also

[Gino.Build::formElement\(\)](#)

8.148.3.5 `getTrnsl ()`

Getter della proprietà trnsl (indica se il campo necessita di traduzione o no)

Returns

proprietà trnsl

8.148.3.6 setTrnsl (*\$value*)

Setter della proprietà trnsl.

Parameters

bool	<i>\$value</i>	
------	----------------	--

Returns

void

8.148.4 Field Documentation

8.148.4.1 \$_trnsI [protected]

Proprietà dei campi specifiche del tipo di campo.

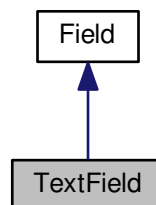
The documentation for this class was generated from the following file:

- lib/classes/build/[class.TextBuild.php](#)

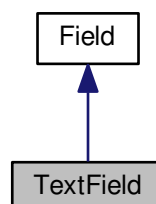
8.149 TextField Class Reference

Campo di tipo TEXT.

Inheritance diagram for TextField:



Collaboration diagram for TextField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [getProperties](#) ()
- [valueFromDb](#) (\$value)
- [valueToDb](#) (\$value)

Protected Attributes

- [\\$_trnsI](#)
Proprietà dei campi specifiche del tipo di campo.

Additional Inherited Members

8.149.1 Detailed Description

Campo di tipo TEXT.

Tipologie di input associabili: textarea, testo, editor html

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.149.2 Constructor & Destructor Documentation

8.149.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Field() • opzioni specifiche del tipo di campo <ul style="list-style-type: none"> – trnsI (boolean): campo con traduzioni
-------	------------------	---

Returns

istanza di [Gino.TextField](#)

8.149.3 Member Function Documentation**8.149.3.1** `getProperties ()`**See also**

[Gino.Field::getProperties\(\)](#)

8.149.3.2 `valueFromDb ($value)`**See also**

[Gino.Field::valueFromDb\(\)](#)

Returns

null or string

8.149.3.3 `valueToDb ($value)`**See also**

[Gino.Field::valueToDb\(\)](#)

Returns

null or string

8.149.4 Field Documentation**8.149.4.1** `$_trnsI` [protected]

Proprietà dei campi specifiche del tipo di campo.

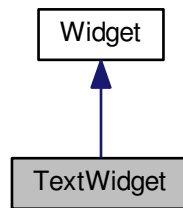
The documentation for this class was generated from the following file:

- [lib/classes/fields/class.TextField.php](#)

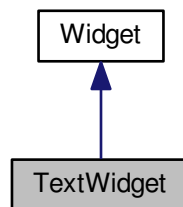
8.150 TextWidget Class Reference

Campi di tipo testo nei form.

Inheritance diagram for TextWidget:



Collaboration diagram for TextWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.150.1 Detailed Description

Campi di tipo testo nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.150.2 Member Function Documentation

8.150.2.1 `printInputForm ($form, $options)`

See also

[Gino.Widget::printInputForm\(\)](#)

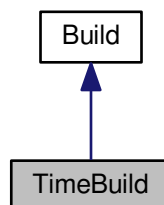
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.TextWidget.php](#)

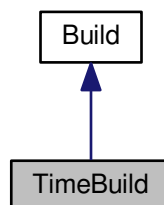
8.151 TimeBuild Class Reference

Gestisce i campi di tipo TIME.

Inheritance diagram for TimeBuild:



Collaboration diagram for TimeBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [__toString](#) ()
Rappresentazione a stringa dell'oggetto.
- [formElement](#) (\Gino\Form \$form, \$options)
- [clean](#) (\$options=null)

Additional Inherited Members

8.151.1 Detailed Description

Gestisce i campi di tipo TIME.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.151.2 Constructor & Destructor Documentation

8.151.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Build()
-------	------------------	---

8.151.3 Member Function Documentation

8.151.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

valore del campo

8.151.3.2 clean (*\$options = null*)

See also

[Gino.Build::clean\(\)](#)

8.151.3.3 formElement (\Gino\Form *\$form*, *\$options*)

See also

[Gino.Build::formElement\(\)](#)

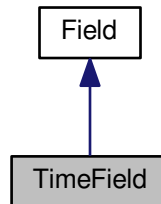
The documentation for this class was generated from the following file:

- lib/classes/build/[class.TimeBuild.php](#)

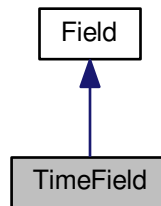
8.152 TimeField Class Reference

Campo di tipo TIME.

Inheritance diagram for TimeField:



Collaboration diagram for TimeField:



Public Member Functions

- `__construct` (\$options)
Costruttore.

Additional Inherited Members

8.152.1 Detailed Description

Campo di tipo TIME.

Tipologie di input associabili: testo in formato ora.

L'orario può essere mostrato con o senza i secondi utilizzando la chiave *seconds*.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.152.2 Constructor & Destructor Documentation

8.152.2.1 `__construct ($options)`

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">• opzioni generali definite come proprietà nella classe Field()
-------	------------------	---

Returns

istanza di [Gino.TimeField](#)

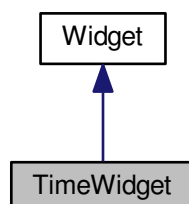
The documentation for this class was generated from the following file:

- [lib/classes/fields/class.TimeField.php](#)

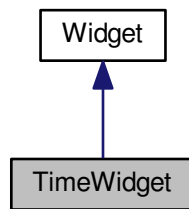
8.153 TimeWidget Class Reference

Campi di tipo orario nei form.

Inheritance diagram for TimeWidget:



Collaboration diagram for TimeWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.153.1 Detailed Description

Campi di tipo orario nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.153.2 Member Function Documentation

8.153.2.1 printInputForm (\$form, \$options)

See also

[Gino.Widget::printInputForm\(\)](#)

Parameters

array	<i>\$options</i>	opzioni dell'elemento del form <ul style="list-style-type: none"> • opzioni dei metodi input() e cinput() della classe Form • seconds (boolean): per mostrare i secondi
-------	------------------	--

The documentation for this class was generated from the following file:

- lib/classes/widget/class.TimeWidget.php

8.154 Translation Class Reference

Classe per la gestione delle traduzioni.

Public Member Functions

- [__construct](#) (\$language, \$language_dft)
Costruttore.
- [selectTXT](#) (\$table, \$field, \$reference, \$id_name='id')
Traduzione di un campo di tabella.
- [manageTranslation](#) (\$request)
Gestisce le traduzioni nei form.
- [listItemOrdered](#) (\$query, \$id_name, \$tbl, \$ord_field, \$ord_type)
Ordina i risultati di una query facendo riferimento ai testi tradotti.
- [formTranslation](#) ()
Form per l'inserimento e la modifica delle traduzioni.
- [actionTranslation](#) (\Gino\Http\Request \$request)
Inserimento e modifica delle traduzioni.

Static Public Member Functions

- static [deleteTranslations](#) (\$tbl, \$tbl_id)
Eliminazione traduzione.

Private Attributes

- [\\$_registry](#)
- [\\$_db](#)
- [\\$_lng](#)
Lingua valida in sessione.
- [\\$_lngDft](#)
Lingua di default (se non viene trovata \$_lng)
- [\\$_tbl_translation](#)
tabella che conserva le traduzioni

8.154.1 Detailed Description

Classe per la gestione delle traduzioni.

Le traduzioni vengono cercate nella lingua di navigazione. Se non presenti viene restituita la traduzione nella lingua di default. La lingua di navigazione è quella ricavata dallo user agent del client, oppure impostata in sessione a seguito di scelta dell'utente. La lingua di default è quella impostata come tale da interfaccia.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.154.2 Constructor & Destructor Documentation

8.154.2.1 `__construct ($language, $language_dft)`

Costruttore.

Parameters

string	<i>\$language</i>	lingua di navigazione
string	<i>\$language_dft</i>	lingua di default

Returns

istanza di [Gino.Translation](#)

8.154.3 Member Function Documentation

8.154.3.1 `actionTranslation (\Gino\Http\Request $request)`

Inserimento e modifica delle traduzioni.

Returns

risultato operazione

8.154.3.2 `static deleteTranslations ($tbl, $tbl_id) [static]`

Eliminazione traduzione.

Parameters

string	<i>\$tbl</i>	nome della tabella con il campo da tradurre
integer	<i>\$tbl_id</i>	valore dell'ID del record di riferimento per la traduzione

Returns

risultato operazione, bool

8.154.3.3 `formTranslation ()`

[Form](#) per l'inserimento e la modifica delle traduzioni.

Il metodo viene richiamato da una request ajax avviata dalla funzione javascript `prepareTrlForm()`.

Returns

form inserimento traduzione

8.154.3.4 listItemOrdered (*\$query*, *\$id_name*, *\$tbl*, *\$ord_field*, *\$ord_type*)

Ordina i risultati di una query facendo riferimento ai testi tradotti.

Parameters

string	<i>\$query</i>	query
string	<i>\$id_name</i>	nome del campo di riferimento
string	<i>\$tbl</i>	nome della tabella del testo da tradurre
string	<i>\$ord_field</i>	nome del campo da tradurre e in base al quale ordinare
string	<i>\$ord_type</i>	tipo di ordinamento (asc, desc)

Returns

risultati ordinati

8.154.3.5 manageTranslation (*\$request*)

Gestisce le traduzioni nei form.

Parameters

object	<i>\$request</i>	oggetto Request
--------	------------------	-----------------

Returns

oggetto Response o null

8.154.3.6 selectTXT (*\$table*, *\$field*, *\$reference*, *\$id_name* = 'id')

Traduzione di un campo di tabella.

Se non è presente la traduzione viene mostrato il testo nella lingua di default

Parameters

string	<i>\$table</i>	nome della tabella del testo da tradurre
string	<i>\$field</i>	nome del campo da tradurre
mixed	<i>\$reference</i>	valore del campo di riferimento
string	<i>\$id_name</i>	nome del campo di riferimento

Returns

traduzione

8.154.4 Field Documentation

8.154.4.1 *\$_db* [private]8.154.4.2 *\$_lng* [private]

Lingua valida in sessione.

8.154.4.3 *\$_lngDft* [private]

Lingua di default (se non viene trovata *\$_lng*)

8.154.4.4 `$_registry` [private]

8.154.4.5 `$_tbl_translation` [private]

tabella che conserva le traduzioni

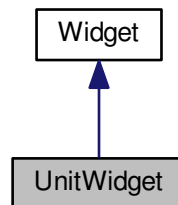
The documentation for this class was generated from the following file:

- [lib/classes/class.Translation.php](#)

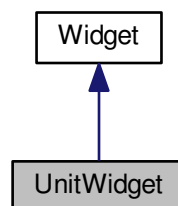
8.155 UnitWidget Class Reference

Insieme di campi di un modello.

Inheritance diagram for UnitWidget:



Collaboration diagram for UnitWidget:



Public Member Functions

- [printInputForm](#) (\$form, \$options)

Additional Inherited Members

8.155.1 Detailed Description

Insieme di campi di un modello.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.155.2 Member Function Documentation

8.155.2.1 printInputForm (*\$form*, *\$options*)

See also

[Gino.Widget::printInputForm\(\)](#)

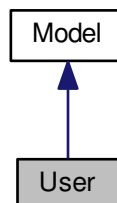
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.UnitWidget.php](#)

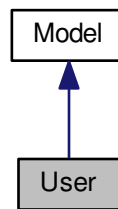
8.156 User Class Reference

Classe di tipo [Gino.Model](#) che rappresenta un utente.

Inheritance diagram for User:



Collaboration diagram for User:



Public Member Functions

- `__construct ($id)`
Costruttore.
- `__toString ()`
Rappresentazione a stringa dell'oggetto.
- `formPassword ($options=array())`
Form per cambiare la password.
- `savePassword ($options=array())`
Processa il form cambio password.
- `hasPerm ($class_name, $perms, $instance=0)`
Verifica se l'utente ha uno dei permessi della classe.
- `hasAdminPerm ($class_name, $instance=0)`
Verifica se l'utente ha uno dei permessi amministrativi della classe.
- `getPermissions ()`
Elenco dei permessi di un utente.
- `delete ()`
Eliminazione utente.

Static Public Member Functions

- static `columns ()`
Struttura dei campi della tabella di un modello.
- static `setPassword ($password, $options=array())`
Imposta la password (dal form di inserimento utente)
- static `checkPassword ($options=array())`
Verifica la conformità di una password.
- static `generatePassword ($options=array())`
Genera una password (random)
- static `checkEmail ($id=null)`
Verifica la conformità e validità di un indirizzo email.
- static `checkUsername ($options)`
Verifica l'unicità dello username.
- static `getFromUserPwd ($username, $password, $auth_idap)`
Restituisce l'utente dati i valori username e password.

- static [getFromEmail](#) (\$email, \$active=TRUE)
Restituisce l'utente legato all'email data.
- static [getUsersFromPermissions](#) (\$code, \$controller)
Elenco degli utenti associati a uno o più permessi.
- static [setMergeValue](#) (\$permission_id, \$instance_id)
Valore che raggruppa permesso e istanza.
- static [getMergeValue](#) (\$value)
Splitta i valori di permesso e istanza.
- static [setMoreInfo](#) (\$id)
Gestisce i record della tabella aggiuntiva degli utenti.
- static [deleteMoreInfo](#) (\$id)
Elimina i record della tabella aggiuntiva degli utenti.

Static Public Attributes

- static [\\$table](#) = TBL_USER
- static [\\$table_groups](#) = TBL_USER_GROUP
- static [\\$table_more](#) = TBL_USER_ADD
- static [\\$columns](#)

Static Protected Member Functions

- static [properties](#) (\$model)

Static Private Member Functions

- static [getLanguageNation](#) ()
Codice lingua per l'elenco nazioni nella tabella nation.

Static Private Attributes

- static [\\$extension_media](#)
- static [\\$lng_nav](#)
- static [\\$lng_dft](#)

Additional Inherited Members

8.156.1 Detailed Description

Classe di tipo [Gino.Model](#) che rappresenta un utente.

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.156.2 Constructor & Destructor Documentation

8.156.2.1 `__construct ($id)`

Costruttore.

Parameters

integer	<i>\$id</i>	valore ID del record
---------	-------------	----------------------

Returns

istanza di [Gino.App.Auth.User](#)

8.156.3 Member Function Documentation

8.156.3.1 __toString ()

Rappresentazione a stringa dell'oggetto.

Returns

nome e cognome

8.156.3.2 static checkEmail (*\$id* = null) [static]

Verifica la conformità e validità di un indirizzo email.

Parametri POST:

- email (string)
- check_email (string)

See also

[Gino.checkEmail\(\)](#)

Parameters

integer	<i>\$id</i>	valore ID del record sul quale non si effettua il controllo (per le operazioni di modifica)
---------	-------------	---

Returns

array (error) or true

8.156.3.3 static checkPassword (*\$options* = array()) [static]

Verifica la conformità di una password.

Parametri POST:

- userpwd (string)
- check_userpwd (string)

Se è presente l'input form *check_userpwd* viene controllata la corrispondenza con l'input form *userpwd*.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• password (string): valore della password (se non indicato viene recuperato il valore dal POST)• check_password (string): valore di controllo della password (se non indicato viene recuperato il valore dal POST)• pwd_length_min (integer): numero minimo di caratteri della password• pwd_length_max (integer): numero massimo di caratteri della password• pwd_numeric_number (integer): numero di caratteri numerici da inserire nella password
-------	------------------	---

Returns

array (errore) oppure TRUE

8.156.3.4 static checkUsername (*\$options*) [static]

Verifica l'unicità dello username.

Parametri POST:

- username (string)
- email (string)

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• username_as_email (boolean): indica se come username viene utilizzata l'email (default false)
-------	------------------	---

Returns

array (error) or true

8.156.3.5 static columns () [static]

Struttura dei campi della tabella di un modello.

Returns

array

8.156.3.6 delete ()

Eliminazione utente.

See also

[Model::delete\(\)](#)

8.156.3.7 static deleteMoreInfo (*\$id*) [static]

Elimina i record della tabella aggiuntiva degli utenti.

Parameters

integer	<i>\$id</i>	valore ID dell'utente
---------	-------------	-----------------------

Returns

boolean

8.156.3.8 formPassword (*\$options* = array())

[Form](#) per cambiare la password.

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• form_action (string): indirizzo del form action• rules (string): descrizione delle regole alle quali è sottoposta la password• maxlength (integer): numero massimo di caratteri
-------	------------------	---

Returns

html, form

8.156.3.9 static generatePassword (*\$options* = array()) [static]

Genera una password (random)

Parameters

array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none">• aut_password_length (integer): numero di caratteri della password automatica
-------	------------------	--

Returns

password

8.156.3.10 static getFromEmail (*\$email*, *\$active* = TRUE) [static]

Restituisce l'utente legato all'email data.

Parameters

string	<i>\$email</i>	
bool	<i>\$active</i>	considera solo utenti attivi, default TRUE

Returns

[Gino.App.Auth.User](#) oggetto utente

8.156.3.11 `static getUserPwd ($username, $password, $auth_ldap) [static]`

Restituisce l'utente dati i valori username e password.

See also

[Gino.Access::Authentication\(\)](#)

Parameters

string	<i>\$username</i>	lo username
string	<i>\$password</i>	la password
boolean	<i>\$auth_ldap</i>	risultato dell'autenticazione ldap

Returns

mixed [Gino.App.Auth.User](#) ricavato oppure null

8.156.3.12 `static getLanguageNation () [static],[private]`

Codice lingua per l'elenco nazioni nella tabella nation.

Returns

string

8.156.3.13 `static getMergeValue ($value) [static]`

Splitta i valori di permesso e istanza.

Parameters

string	<i>\$value</i>	valore da splittare
--------	----------------	---------------------

Returns

array array(permission_id, instance_id)

8.156.3.14 `getPermissions ()`

Elenco dei permessi di un utente.

See also

[setMergeValue\(\)](#)

Parameters

integer	<i>\$id</i>	valore ID dell'utente
---------	-------------	-----------------------

Returns

array permessi

8.156.3.15 static getUsersFromPermissions (*\$code*, *\$controller*) [static]

Elenco degli utenti associati a uno o più permessi.

Parameters

	<i>string!array</i>	<i>\$code</i> codice o codici del permesso
object	<i>\$controller</i>	controller

Returns

array (valori id)

8.156.3.16 hasAdminPerm (*\$class_name*, *\$instance* = 0)

Verifica se l'utente ha uno dei permessi amministrativi della classe.

Parameters

string	<i>\$class_name</i>	il nome della classe
int	<i>\$instance</i>	istanza della classe (0 per classi non istanziabili)

Returns

bool

8.156.3.17 hasPerm (*\$class_name*, *\$perms*, *\$instance* = 0)

Verifica se l'utente ha uno dei permessi della classe.

Parameters

string	<i>\$class_name</i>	nome della classe
int array	<i>\$perms</i>	id o array di id dei permessi da verificare
int	<i>\$instance</i>	istanza della classe (0 per classi non istanziabili)

Returns

bool

8.156.3.18 static properties (*\$model*) [static], [protected]

See also

[Gino.Model::properties\(\)](#)

8.156.3.19 `savePassword ($options = array())`

Processa il form cambio password.

See also

`self::formPassword()`

Parameters

array	<i>\$options</i>	array associativo di opzioni (per il metodo checkPassword()) <ul style="list-style-type: none"> • pwd_length_min (integer): numero minimo di caratteri della password • pwd_length_max (integer): numero massimo di caratteri della password • pwd_numeric_number (integer): numero di caratteri numerici da inserire nella password
-------	------------------	---

Returns

boolean

8.156.3.20 `static setMergeValue ($permission_id, $instance_id) [static]`

Valore che raggruppa permesso e istanza.

Parameters

integer	<i>\$permission_id</i>	valore ID del permesso
integer	<i>\$instance_id</i>	valore ID dell'istanza

Returns

string

8.156.3.21 `static setMoreInfo ($id) [static]`

Gestisce i record della tabella aggiuntiva degli utenti.

Parameters

integer	<i>\$id</i>	valore ID dell'utente
---------	-------------	-----------------------

Returns

boolean

8.156.3.22 `static setPassword ($password, $options = array()) [static]`

Imposta la password (dal form di inserimento utente)

See also

[generatePassword\(\)](#)
[pub::cryptMethod\(\)](#)

Parameters

string	<i>\$password</i>	
array	<i>\$options</i>	array associativo di opzioni <ul style="list-style-type: none"> • aut_password (boolean): la password è generata dal sistema (default false)

Returns

string (password) oppure array (error)

8.156.4 Field Documentation

8.156.4.1 **\$columns** [static]

8.156.4.2 **\$extension_media** [static], [private]

8.156.4.3 **\$lng_dft** [static], [private]

8.156.4.4 **\$lng_nav** [static], [private]

8.156.4.5 **\$table = TBL_USER** [static]

8.156.4.6 **\$table_groups = TBL_USER_GROUP** [static]

8.156.4.7 **\$table_more = TBL_USER_ADD** [static]

The documentation for this class was generated from the following file:

- app/auth/[class.User.php](#)

8.157 View Class Reference

Gestisce le viste, impostando il template e ritornando l'output.

Public Member Functions

- [__construct](#) (\$view_folder=null, \$tpl=null)
Costruttore.
- [setViewTpl](#) (\$view_name, \$opts=null)
Setta il template della view.
- [assign](#) (\$name, \$value)
Aggiunge variabili al context.
- [render](#) (\$data=null)
Ritorna l'output generato dalla vista.

Protected Attributes

- [\\$_data](#)
oggetto che contiene il context della view
- [\\$_registry](#)

- istanze del registry*
- [\\$_view_folder](#)
percorso alla cartella contenente le view specifiche del modulo
- [\\$_dft_view_folder](#)
percorso alla cartella che contiene le view di sistema
- [\\$_view_tpl](#)
path alla view correntemente in uso

8.157.1 Detailed Description

Gestisce le viste, impostando il template e ritornando l'output.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

8.157.2 Constructor & Destructor Documentation

8.157.2.1 __construct (\$view_folder = null, \$tpl = null)

Costruttore.

Parameters

string	<i>\$view_folder</i>	path alla directory in cui cercare il template, default null. Se ha valore null il template viene cercato all'interno della directory dei template generali di gino.
string	<i>\$tpl</i>	nome template, default null

Returns

istanza di [Gino.View](#)

8.157.3 Member Function Documentation

8.157.3.1 assign (\$name, \$value)

Aggiunge variabili al context.

Parameters

string	<i>\$name</i>	il nome della variabile da utilizzare nella vista
mixed	<i>\$value</i>	il valore della variabile

Returns

void

8.157.3.2 render (\$data = null)

Ritorna l'output generato dalla vista.

Parameters

<i>array</i>	data dizionario delle variabili da passare
--------------	--

Returns

l'output della vista

8.157.3.3 `setViewTpl ($view_name, $opts = null)`

Setta il template della view.

Cerca il template nella directory che contiene le view specifiche e se non lo trova prosegue la ricerca nella directory delle view di sistema

Parameters

string	<i>\$view_name</i>	il nome del template
array	<i>\$opts</i>	

Returns

void o [Exception](#) se il template non viene trovato

8.157.4 Field Documentation

8.157.4.1 `$_data` [`protected`]

oggetto che contiene il context della view

8.157.4.2 `$_dft_view_folder` [`protected`]

percorso alla cartella che contiene le view di sistema

8.157.4.3 `$_registry` [`protected`]

istanze del registry

8.157.4.4 `$_view_folder` [`protected`]

percorso alla cartella contenente le view specifiche del modulo

8.157.4.5 `$_view_tpl` [`protected`]

path alla view correntemente in uso

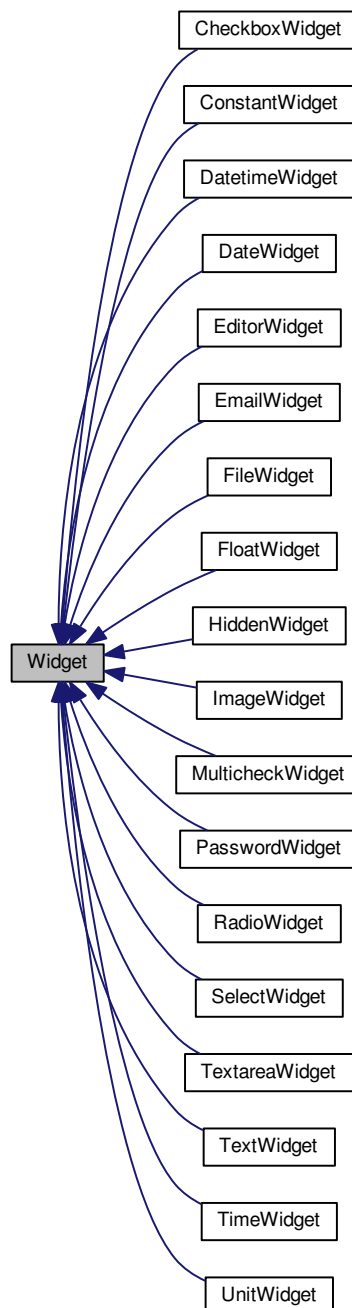
The documentation for this class was generated from the following file:

- `lib/classes/mvc/class.View.php`

8.158 Widget Class Reference

Gestisce i campi nei form.

Inheritance diagram for Widget:



Public Member Functions

- `__construct ()`
Costruttore.
- `printInputForm ($form, $options)`

Stampa l'input form associato al widget. In particolare imposta le proprietà principali del widget.

Data Fields

- [\\$_label](#)
- [\\$_value](#)
- [\\$_default](#)

Protected Attributes

- [\\$_name](#)
- [\\$_form](#)

8.158.1 Detailed Description

Gestisce i campi nei form.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.158.2 Constructor & Destructor Documentation

8.158.2.1 `__construct ()`

Costruttore.

8.158.3 Member Function Documentation

8.158.3.1 `printInputForm ($form, $options)`

Stampa l'input form associato al widget In particolare imposta le proprietà principali del widget.

È poi il singolo widget che stampa l'input form richiamando prima le proprietà.

Parameters

object	<i>\$form</i>	oggetto Gino.Form
array	<i>\$options</i>	array associativo di opzioni del campo/modello

Returns

string

8.158.4 Field Documentation

8.158.4.1 `$_default`

8.158.4.2 `$_form` [protected]

8.158.4.3 `$_label`

8.158.4.4 `$_name` [protected]

8.158.4.5 `$_value`

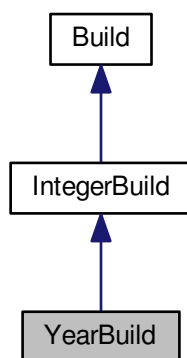
The documentation for this class was generated from the following file:

- lib/classes/widget/[class.Widget.php](#)

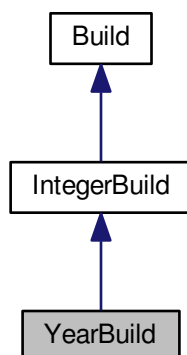
8.159 YearBuild Class Reference

Gestisce i campi di tipo ANNO.

Inheritance diagram for YearBuild:



Collaboration diagram for YearBuild:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.
- [formElement](#) (\Gino\Form \$form, \$options)

Additional Inherited Members

8.159.1 Detailed Description

Gestisce i campi di tipo ANNO.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.159.2 Constructor & Destructor Documentation

8.159.2.1 __construct (\$options)

Costruttore.

See also

[Gino.Build::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none"> • opzioni generali definite come proprietà nella classe Build() e Integer↔ Build()
-------	------------------	--

8.159.3 Member Function Documentation

8.159.3.1 formElement (\Gino\Form \$form, \$options)

See also

[Gino.IntegerBuild::formElement\(\)](#)

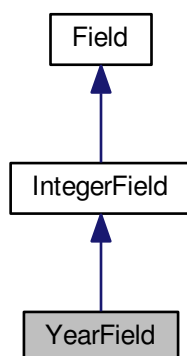
The documentation for this class was generated from the following file:

- lib/classes/build/[class.YearBuild.php](#)

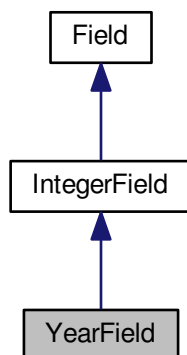
8.160 YearField Class Reference

Campo di tipo ANNO.

Inheritance diagram for YearField:



Collaboration diagram for YearField:



Public Member Functions

- [__construct](#) (\$options)
Costruttore.

Additional Inherited Members

8.160.1 Detailed Description

Campo di tipo ANNO.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

8.160.2 Constructor & Destructor Documentation

8.160.2.1 __construct (*\$options*)

Costruttore.

See also

[Gino.Field::__construct\(\)](#)

Parameters

array	<i>\$options</i>	array associativo di opzioni del campo del database <ul style="list-style-type: none">opzioni generali definite come proprietà nella classe Field() e Integer↔Field()
-------	------------------	---

Returns

istanza di [Gino.YearField](#)

The documentation for this class was generated from the following file:

- lib/classes/fields/[class.YearField.php](#)

Chapter 9

File Documentation

9.1 app/attachment/class.AttachmentCtg.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.AttachmentCtg](#).

Data Structures

- class [AttachmentCtg](#)
Classe di tipo [Gino.Model](#) che rappresenta una categoria di allegati.

Namespaces

- [Gino\App\Attachment](#)
Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati

9.1.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.AttachmentCtg](#).

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.2 app/attachment/class.AttachmentItem.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.AttachmentItem](#).

Data Structures

- class [AttachmentItem](#)
Classe di tipo [Gino.Model](#) che rappresenta una singolo allegato.

Namespaces

- [Gino\App\Attachment](#)

Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati

9.2.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.AttachmentItem](#).

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.3 app/attachment/class.AttachmentItemAdminTable.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.AttachmentItemAdminTable](#).

Data Structures

- class [AttachmentItemAdminTable](#)

Estende la classe [Gino.AdminTable](#) per gestire il salvataggio dell'allegato quando viene modificata la directory (categoria)

Namespaces

- [Gino\App\Attachment](#)

Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati

9.3.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.AttachmentItemAdminTable](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.4 app/attachment/class_attachment.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.attachment](#).

Data Structures

- class [attachment](#)

Classe controller del modulo di gestione di archivi di file categorizzati.

Namespaces

- [Gino\App\Attachment](#)

Namespace dell'applicazione [Attachment](#), che gestisce allegati categorizzati

9.4.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Attachment.attachment](#).

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.5 app/auth/class.AdminTable_AuthUser.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.AdminTable_AuthUser](#).

Data Structures

- class [AdminTable_AuthUser](#)

Sovrascrive la classe [AdminTable](#).

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.5.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.AdminTable_AuthUser](#).

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.6 app/auth/class.Group.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.Group](#).

Data Structures

- class [Group](#)
Classe tipo [Gino.Model](#) che rappresenta un gruppo di utenti.

Namespaces

- [Gino\App\Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.6.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.Group](#).

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.7 app/auth/class.Ldap.php File Reference

Contiene la definizione ed implementazione della class [Gino.App.Auth.Ldap](#).

Data Structures

- class [Ldap](#)
Libreria di connessione ai server [Ldap](#).

Namespaces

- [Gino\App\Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.7.1 Detailed Description

Contiene la definizione ed implementazione della class [Gino.App.Auth.Ldap](#).

Copyright

2013 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

UTILIZZO

Per implementare l'autenticazione attraverso Ldap occorre creare un utente da associare a quello Ldap, che può essere:

1. l'utente con lo username indicato nelle opzioni Ldap (nel caso dell'utente unico)
2. un utente di gino con lo username uguale a quello dell'autenticazione Ldap (nel caso di utente non unico)

9.8 app/auth/class.Permission.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.Permission](#).

Data Structures

- class [Permission](#)
Classe tipo [Gino.Model](#) che rappresenta un permesso.

Namespaces

- [Gino\App\Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.8.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.Permission](#).

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.9 app/auth/class.RegistrationProfile.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.RegistrationProfile](#).

Data Structures

- class [RegistrationProfile](#)
Classe tipo [Gino.Model](#) che rappresenta un profilo di registrazione utenti.

Namespaces

- [Gino\App\Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.9.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.RegistrationProfile](#).

Copyright

2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.10 app/auth/class.RegistrationRequest.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.RegistrationRequest](#).

Data Structures

- class [RegistrationRequest](#)
Classe tipo [Gino.Model](#) che rappresenta una richiesta di registrazione.

Namespaces

- [Gino\App\Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.10.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.RegistrationRequest](#).

Copyright

2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.11 app/auth/class.User.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.User](#).

Data Structures

- class [User](#)
Classe di tipo [Gino.Model](#) che rappresenta un utente.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.11.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.User](#).

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.12 app/auth/class_auth.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Auth.auth](#).

Data Structures

- class [auth](#)

Classe di tipo [Gino.Controller](#) per la gestione degli utenti, gruppi e permessi.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.12.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Auth.auth](#).

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.13 app/auth/config.ldap.php File Reference

File di configurazione per l'accesso a LDAP.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

Variables

- const [LDAP_HOST](#) ""
- const [LDAP_PORT](#) '389'
- const [LDAP_BASE_DN](#) ""
- const [LDAP_SEARCH_DN](#) ""
- const [LDAP_APP_USERNAME](#) ""
- const [LDAP_APP_PASSWORD](#) ""
- const [LDAP_DOMAIN](#) ""
- const [LDAP_PROTOCOL_VERSION](#) ""
- const [LDAP_FILTER_SEARCH](#) ""

9.13.1 Detailed Description

File di configurazione per l'accesso a LDAP.

Contiene le impostazioni per la ricerca degli utenti ldap e i parametri di connessione

Copyright

2013 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.14 app/auth/views/activate_profile.php File Reference

Template registrazione nuovo profilo per utente già registrato.

9.14.1 Detailed Description

Template registrazione nuovo profilo per utente già registrato.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione
- \$form: html, form di registrazione

9.15 app/auth/views/activation_email_message.php File Reference

Template messaggio mail inviata a seguito di conferma indirizzo email.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.15.1 Detailed Description

Template messaggio mail inviata a seguito di conferma indirizzo email.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione
- \$user: [Gino.App.Auth.User](#), utente attivato
- \$login_url: string, URL login
- \$profile_url: string, URL pagina profilo

9.16 app/auth/views/activation_email_object.php File Reference

Template oggetto mail inviata a seguito di conferma indirizzo email.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.16.1 Detailed Description

Template oggetto mail inviata a seguito di conferma indirizzo email.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione

Le variabili a disposizione sono:

- \$user: [Gino.App.Auth.User](#), utente
- \$data: array. Array di informazioni aggiuntive legate ai profili. 'description': descrizione profilo 'content': informazioni 'update_url': url update informazioni
- \$form_password: html, form di modifica password

9.17 app/auth/views/confirmation_result.php File Reference

Template risultato conferma registrazione.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.17.1 Detailed Description

Template risultato conferma registrazione.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione
- \$registration_request: [Gino.App.Auth.RegistrationRequest](#) richiesta di registrazione
- \$mail_sent: bool, invio mail con attivazione automatica riuscito

9.18 app/auth/views/data_recovery_request.php File Reference

Template richiesta di recupero credenziali.

9.18.1 Detailed Description

Template richiesta di recupero credenziali.

Le variabili a disposizione sono:

- \$form: html, form invio email

9.19 app/auth/views/data_recovery_request_processed.php File Reference

Template processing richiesta di recupero credenziali.

9.19.1 Detailed Description

Template processing richiesta di recupero credenziali.

Le variabili a disposizione sono:

- \$error: bool, True se la mail inviata non corrisponde ad alcun utente

9.20 app/auth/views/data_recovery_success.php File Reference

Template successo richiesta di recupero credenziali.

9.20.1 Detailed Description

Template successo richiesta di recupero credenziali.

Le variabili a disposizione sono:

- \$profile_url: string, url pagina profilo utente

9.21 app/auth/views/login.php File Reference

Template form di login.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.21.1 Detailed Description

Template form di login.

Le variabili a disposizione sono:

- \$title: string, titolo
- \$form: html, form di login

9.22 app/auth/views/profile.php File Reference

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.23 app/auth/views/registration.php File Reference

Template form di registrazione.

9.23.1 Detailed Description

Template form di registrazione.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione
- \$form: html, form di registrazione

9.24 app/auth/views/registration_email_message.php File Reference

Template corpo della mail inviata a seguito di registrazione.

Namespaces

- [Gino\App\Auth](#)

Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.24.1 Detailed Description

Template corpo della mail inviata a seguito di registrazione.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione
- \$request: [Gino.App.Auth.RegistrationRequest](#), richiesta di registrazione
- \$confirmation_url: string, URL per conferma indirizzo email

9.25 app/auth/views/registration_email_object.php File Reference

Template oggetto mail inviata a seguito di registrazione.

Namespaces

- [Gino\App\Auth](#)
Namespace dell'applicazione [Auth](#), per la gestione di utenti, gruppi e permessi

9.25.1 Detailed Description

Template oggetto mail inviata a seguito di registrazione.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione

9.26 app/auth/views/registration_result.php File Reference

Template risultato registrazione.

Variables

- `if($error!==FALSE)('Registrazione non riuscita.')` ?></p><p><? else = \$error

9.26.1 Detailed Description

Template risultato registrazione.

Le variabili a disposizione sono:

- \$profile: [Gino.App.Auth.RegistrationProfile](#), profilo di registrazione
- \$error: mixed, bool FALSE, oppure string messaggio errore
- \$request: [Gino.App.Auth.RegistrationRequest](#), richiesta di registrazione

9.26.2 Variable Documentation

9.26.2.1 `if ($error !== FALSE) ('Registrazione non riuscita.') ?></p><p><? else = $error`

9.27 app/graphics/class.GraphicsItem.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Graphics.GraphicsItem](#).

Data Structures

- class [GraphicsItem](#)
Classe di tipo [Gino.Model](#) che rappresenta un header/footer.

Namespaces

- [Gino\App\Graphics](#)
Namespace dell'applicazione [Graphics](#), che gestisce in maniera semplificata header e footer

9.27.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Graphics.GraphicsItem](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.28 app/graphics/class_graphics.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Graphics.graphics](#).

Data Structures

- class [graphics](#)
Classe di tipo [Gino.Controller](#) per la gestione personalizzata degli header e footer del sistema.

Namespaces

- [Gino\App\Graphics](#)
Namespace dell'applicazione [Graphics](#), che gestisce in maniera semplificata header e footer

9.28.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Graphics.graphics](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.29 app/graphics/views/render.php File Reference

Namespaces

- [Gino\App\Graphics](#)

Namespace dell'applicazione [Graphics](#), che gestisce in maniera semplificata header e footer

9.30 app/menu/views/render.php File Reference

Namespaces

- [Gino\App\Menu](#)

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

9.31 app/index/class_index.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Index.index](#).

Data Structures

- class [index](#)

Classe di tipo [Gino.Controller](#) del modulo che gestisce la home amministrativa.

Namespaces

- [Gino\App\Index](#)

Namespace dell'applicazione [Index](#), che gestisce la home amministrativa

9.31.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Index.index](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.32 app/instruments/class_instruments.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Instruments.instruments](#).

Data Structures

- class [instruments](#)

Classe di tipo [Gino.Controller](#) per la gestione di strumenti aggiuntivi di gino.

Namespaces

- [Gino\App\Instruments](#)

Namespace dell'applicazione [Instruments](#), per la visualizzazione di utilità quali percorsi e mime types

9.32.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Instruments.instruments](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.33 app/language/class.Lang.php File Reference

Contiene la classe definizione ed implementazione della classe [Gino.App.Language.Lang](#).

Data Structures

- class [Lang](#)

Classe di tipo [Gino.Model](#) che rappresenta una lingua installata sul sistema.

Namespaces

- [Gino\App\Language](#)

Namespace dell'applicazione [Language](#), che gestisce le lingue installate sul sistema e disponibili per le traduzioni

9.33.1 Detailed Description

Contiene la classe definizione ed implementazione della classe [Gino.App.Language.Lang](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.34 app/language/class_language.php File Reference

Contiene la classe definizione ed implementazione della classe [Gino.App.Language.language](#).

Data Structures

- class [language](#)

Classe di tipo [Gino.Controller](#) per la gestione delle lingue disponibili per le traduzioni.

Namespaces

- [Gino\App\Language](#)

Namespace dell'applicazione [Language](#), che gestisce le lingue installate sul sistema e disponibili per le traduzioni

9.34.1 Detailed Description

Contiene la classe definizione ed implementazione della classe [Gino.App.Language.language](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.35 app/language/views/choice_language.php File Reference

Template della vista scelta lingua.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.35.1 Detailed Description

Template della vista scelta lingua.

Variabili disponibili:

- **languages**: array, lingue attive, oggetti [Gino.App.Language.Lang](#)
- **flag**: bool, opzione visualizza bandiere al posto delle etichette
- **flag_prefix**: string, prefisso flag img
- **flag_suffix**: string, suffisso flag img
- **lng_support**: bool, indica se la lingua di navigazione del client è supportata dal sistema
- **lng**: string, codice lingua di navigazione client
- **lng_dft**: string, codice lingua di default
- **router**: , istanza di [Gino.Router](#)

Il path delle immagini è il seguente: SITE_IMG / flag_prefix \$language->label flag_suffix

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.36 app/layout/class_layout.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Layout.layout](#).

Data Structures

- class [layout](#)

Gestisce il layout dell'applicazione raggruppando le funzionalità fornite dalle librerie dei css, template e skin.

Namespaces

- [Gino\App\Layout](#)

Namespace dell'applicazione [Layout](#), che gestisce i layout del sistema.

9.36.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Layout.layout](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.37 app/layout/layout.js File Reference

9.38 app/layout/layout_single.js File Reference

9.39 app/menu/class.MenuVoice.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Menu.MenuVoice](#).

Data Structures

- class [MenuVoice](#)

Classe di tipo [Gino.Model](#) che rappresenta una voce di menu.

Namespaces

- [Gino\App\Menu](#)

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

9.39.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Menu.MenuVoice](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.40 app/menu/class_menu.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Menu.menu](#).

Data Structures

- class [menu](#)

Classe di tipo [Gino.Controller](#) per la gestione dei menu.

Namespaces

- [Gino\App\Menu](#)

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

9.40.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Menu.menu](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.41 app/menu/views/render_mainMenu.php File Reference

Template visualizzazione menu.

Namespaces

- [Gino\App\Menu](#)

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

9.41.1 Detailed Description

Template visualizzazione menu.

Variabili disponibili:

- **instance_name**: nome istanza menu
- **title**: titolo menu
- **selected**: id voce selezionata
- **tree**: array che contiene il tree delle voci di menu. Ciascuna voce è un array associativo con chiavi:
 - id: int, id voce
 - sub: array, tree sottovoci (ricorsivo)
 - label: string, voce di menu
 - type: string, int|ext tipo link,
 - url: string, url

Copyright

2005-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.42 app/menu/views/render_menu_admin.php File Reference

Template visualizzazione menu.

Namespaces

- [Gino\App\Menu](#)

Namespace dell'applicazione [Menu](#), che gestisce istanze di menu

9.42.1 Detailed Description

Template visualizzazione menu.

Variabili disponibili:

- **instance_name**: nome istanza menu
- **title**: titolo menu
- **selected**: id voce selezionata

- **tree**: array che contiene il tree delle voci di menu. Ciascuna voce è un array associativo con chiavi:
 - id: int, id voce
 - sub: array, tree sottovoci (ricorsivo)
 - label: string, voce di menu
 - type: string, int|ext tipo link,
 - url: string, url

Copyright

2005-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.43 app/module/class.ModuleInstance.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Module.ModuleInstance](#).

Data Structures

- class [ModuleInstance](#)
Classe di tipo [Gino.Model](#) che rappresenta un'istanza di un modulo di sistema.

Namespaces

- [Gino\App\Module](#)
Namespace dell'applicazione [Module](#), che gestisce la creazione/rimozione di istanze di moduli di sistema

9.43.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Module.ModuleInstance](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.44 app/module/class_module.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Module.module](#).

Data Structures

- class [module](#)

Classe di tipo [Gino.Controller](#) per la gestione di istanze di moduli di sistema.

Namespaces

- [Gino\App\Module](#)

Namespace dell'applicazione [Module](#), che gestisce la creazione/rimozione di istanze di moduli di sistema

9.44.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Module.module](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.45 app/page/class.PageCategory.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Page.PageCategory](#).

Data Structures

- class [PageCategory](#)

Classe tipo [Gino.Model](#) che rappresenta una categoria di pagine.

Namespaces

- [Gino\App\Page](#)

Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.45.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Page.PageCategory](#).

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.46 app/page/class.PageComment.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Page.PageComment](#).

Data Structures

- class [PageComment](#)
Classe tipo [Gino.Model](#) che rappresenta un commento ad una pagina.

Namespaces

- [Gino\App\Page](#)
Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.46.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Page.PageComment](#).

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.47 app/page/class.PageEntry.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Page.PageEntry](#).

Data Structures

- class [PageEntry](#)
Classe tipo [Gino.Model](#) che rappresenta una pagina.

Namespaces

- [Gino\App\Page](#)
Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.47.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Page.PageEntry](#).

Copyright

2013-2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.48 app/page/class_page.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Page.page](#).

Data Structures

- class [page](#)
Classe di tipo [Gino.Controller](#) per la gestione delle pagine.

Namespaces

- [Gino\App\Page](#)
Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.48.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Page.page](#).

Version

1.0

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.49 app/page/language/en_US/page_lang.php File Reference

9.50 app/page/language/it_IT/page_lang.php File Reference

9.51 app/page/page.js File Reference

Functions

- [window addEvent](#) ('load', function(){if(typeof [prettyPrint](#)== 'function') [prettyPrint](#)();})

Variables

- var [PageSlider](#)

9.52.2.3 window PR_SHOULD_USE_CONTINUATION =!0

9.52.2.4 window prettyPrint

Initial value:

```
=function(a){function m(){for(var e=window.PR_SHOULD_USE_CONTINUATION?1.now()+250:Infinity;
    p<h.length&&l.now()<e;p++){var n=h[p],k=n.className;if(k.indexOf("prettyprint")>=0){var k=k.match(
    g),f,b;if(b=
    !k){b=n;for(var o=void 0,c=b.firstChild;c=c.nextSibling)var i=c.nodeType,
    o=i===1?o?b:c:i===3?N.test(c.nodeValue)?b:o:o;b=(f=o===b?void 0:o)&&"CODE"===
    f.tagName)b&&(k=f.className.match(g));k&&(k=k[1]);b=!1;for(o=n.parentNode;o;
    o=o.parentNode)if((o.tagName==="pre"||o.tagName==="code"||o.tagName==="xmp")&&
    o.className&&o.className.indexOf("prettyprint")>=0){b=!0;break}b||(b=(b=n.className.match(/\b(?:\d+)\b/))?(b[1]&&b[1].length?b[1]:!0:!1)&&D(n,b),d={g:k,h:n,i:b},E(
    d))}p<h.length?setTimeout(m,
    250):a&&a()}for(var e=[document.getElementsByTagName("pre"),document.
    getElementsByTagName("code"),document.getElementsByTagName("xmp")],h=[],k=0;k<e.length;++k)for(var
    t=0,s=e[k].length;t<s;++t)h.push(e[k][t]);var e=q,l=Date,l.now||l={now:function(){return+
    new Date}};var p=0,d,g=/\b(?:\d+)\b/;g?-(["\w.]+)(?!\\S)/;m() }
```

9.52.2.5 window prettyPrintOne =function(a,m,e){var h=document.createElement("PRE");h.innerHTML←
L=a;e&&D(h,e);E({g:m,i:e,h:h});return h.innerHTML}

9.52.2.6 var q =null

9.53 app/page/views/box.php File Reference

Template per la vista delle pagine inserite nel template.

Namespaces

- [Gino\App\Page](#)

Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.53.1 Detailed Description

Template per la vista delle pagine inserite nel template.

Variabili disponibili:

- **section_id**: attributo id del tag section
- **tpl**: template del post deciso da opzioni

Copyright

2012-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

9.54 app/page/views/showcase.php File Reference

Template per la vista vetrina.

Namespaces

- [Gino\App\Page](#)

Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.54.1 Detailed Description

Template per la vista vetrina.

Variabili disponibili:

- **section_id**: attributo id del tag section
- **title**: titolo della vista
- **feed**: link ai feed rss
- **wrapper_id**: attributo id del wrapper che contiene gli elementi della vetrina
- **ctrl_begin**: parte iniziale dell'attributo id dei controller dello slide (l'attributo viene completato con l'indice)
- **items**: elementi della vetrina
- **ctrls**: controlli della vetrina (per lo slide)
- **options**: opzioni da passare alla classe javascript che gestisce lo slide

Copyright

2012-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.55 app/page/views/view.php File Reference

Template per la vista delle pagine.

Namespaces

- [Gino\App\Page](#)

Namespace dell'applicazione [Page](#), per la gestione di pagine categorizzate

9.55.1 Detailed Description

Template per la vista delle pagine.

Variabili disponibili:

- **section_id**: attributo id del tag section
- **page**: oggetto pageEntry
- **tpl**: template del post deciso da opzioni
- **enable_comments**: abilitazione commenti
- **form_comment**: form inserimento commento

Copyright

2012-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.56 app/phpModuleView/class.PhpModule.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.PhpModuleView.PhpModule](#).

Data Structures

- class [PhpModule](#)
Classe di tipo [Gino.Model](#) che rappresenta un output scritto in codice php.

Namespaces

- [Gino\App\PhpModuleView](#)
Namespace dell'applicazione [PhpModuleView](#), che gestisce output generati direttamente da codice php salvato du database

9.56.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.PhpModuleView.PhpModule](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.57 app/phpModuleView/class_phpModuleView.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.PhpModuleView.phpModuleView](#).

Data Structures

- class [phpModuleView](#)
Permette la creazione di moduli di classe in grado di eseguire codice php completamente personalizzabile.

Namespaces

- [Gino\App\PhpModuleView](#)
Namespace dell'applicazione [PhpModuleView](#), che gestisce output generati direttamente da codice php salvato du database

9.57.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.PhpModuleView.phpModuleView](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

9.58 app/searchSite/class_searchSite.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.SearchSite.searchSite](#).

Data Structures

- class [searchSite](#)
Gestisce le ricerche full text sui contenuti dell'applicazione.

Namespaces

- [Gino\App\SearchSite](#)
Namespace dell'applicazione [SearchSite](#), che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca

9.58.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.SearchSite.searchSite](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

9.59 app/searchSite/searchSite.js File Reference

Functions

- [window](#) [addEvent](#) ('domready', function(){var field=\$('#search_site');if(typeof \$('#search_site_check')!= 'undefined'){\$('#search_site_check').addEvent('click', [viewCheckOptions.bind](#)(\$('#search_site_check')));}}
 function viewCheckOptions(height_or
- [if](#) (optionLayer.getStyle('visibility')== 'hidden')
- [setTimeout](#) (function(){optionLayer.setStyle('visibility', 'hidden');}, 500)

Variables

- [window width_or](#)
- [else](#)

9.59.1 Function Documentation

9.59.1.1 `window addEvent ('domready', function(){var field=$('#search_site');if(typeof $('#search_site_check')!= 'undefined'){$('#search_site_check').addEvent('click', viewCheckOptions.bind($('#search_site_check')));}}`

9.59.1.2 `if (optionLayer.getStyle('visibility' == 'hidden')`

9.59.1.3 `setTimeout (function(){optionLayer.setStyle('visibility', 'hidden');}, 500)`

9.59.2 Variable Documentation

9.59.2.1 `else`

Initial value:

```
{
    optionLayer.fade('out')
```

9.59.2.2 `window width_or`

Initial value:

```
{
    var optionLayer = $('#search_site_check_options')
```

9.60 app/searchSite/views/form.php File Reference

Template form di ricerca nel sito.

Namespaces

- [Gino\App\SearchSite](#)

Namespace dell'applicazione [SearchSite](#), che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca

9.60.1 Detailed Description

Template form di ricerca nel sito.

Le variabili a disposizione sono:

- `$form_action`: l'url dell'attributo action del form per accedere ai risultati della ricerca
- `$choices`: bool, dice se sono possibili delle scelte di ricerca

`check_options`

- `se`

choices è vero mostra il pannello con le scelte di ricerca. Nota bene! La vista carica un javascript che gestisce la comparsa/scomparsa del pannello con le opzioni di ricerca, per funzionare gli id del bottone di check deve essere 'search_site_check'. E naturalmente deve essere stampata la variabile `$check_options`.

9.61 app/searchSite/views/results.php File Reference

Template risultati ricerca nel sito.

Namespaces

- [Gino\App\SearchSite](#)

Namespace dell'applicazione [SearchSite](#), che gestisce ricerche full text su tutti i moduli installati che supportano la ricerca

9.61.1 Detailed Description

Template risultati ricerca nel sito.

Le variabili a disposizione sono:

- `$title`: bool, titolo
- `$results_num`: int, numero risultati
- `$content`: html, risultati

9.62 app/statistics/class.LogAccess.php File Reference

Contiene la definizione ed implementazione della classe .

Data Structures

- class [LogAccess](#)

Modello che rappresenta un login sul sistema.

Namespaces

- [Gino\App\Statistics](#)

Namespace dell'applicazione [Statistics](#), che gestisce le statistiche di accesso all'area privata

9.62.1 Detailed Description

Contiene la definizione ed implementazione della classe .

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.63 app/statistics/class_statistics.php File Reference

Contiene la definizione ed implementazione della classe statistics.

Data Structures

- class [statistics](#)

Statistiche degli accessi all'area privata.

Namespaces

- [Gino\App\Statistics](#)

Namespace dell'applicazione [Statistics](#), che gestisce le statistiche di accesso all'area privata

9.63.1 Detailed Description

Contiene la definizione ed implementazione della classe statistics.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.64 app/sysClass/class.ModuleApp.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.SysClass.ModuleApp](#).

Data Structures

- class [ModuleApp](#)

Classe di tipo [Gino.Model](#) che rappresenta un modulo di sistema.

Namespaces

- [Gino\App\SysClass](#)

Namespace dell'applicazione [SysClass](#), che gestisce l'installazione/upgrade/rimozione di classi di sistema

9.64.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.SysClass.ModuleApp](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.65 app/sysClass/class_sysClass.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.SysClass.sysClass](#).

Data Structures

- class [sysClass](#)
Classe di tipo [Gino.Controller](#) per la gestione dei moduli di sistema.

Namespaces

- [Gino\App\SysClass](#)
Namespace dell'applicazione [SysClass](#), che gestisce l'installazione/upgrade/rimozione di classi di sistema

9.65.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.SysClass.sysClass](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.66 app/sysconf/class.Conf.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Sysconf.Conf](#).

Data Structures

- class [Conf](#)
Classe di tipo [Gino.Model](#) che gestisce le impostazioni del sistema.

Namespaces

- [Gino\App\Sysconf](#)

Namespace dell'applicazione [Sysconf](#), che gestisce le impostazioni di sistema

9.66.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Sysconf.Conf](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.67 app/sysconf/class_sysconf.php File Reference

Contiene la definizione ed implementazione della classe [Gino.App.Sysconf.sysconf](#).

Data Structures

- class [sysconf](#)

Classe di tipo [Gino.Controller](#) per la gestione delle principali impostazioni di sistema.

Namespaces

- [Gino\App\Sysconf](#)

Namespace dell'applicazione [Sysconf](#), che gestisce le impostazioni di sistema

9.67.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.App.Sysconf.sysconf](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.68 app/sysfunc/class_sysfunc.php File Reference

Contiene la definizione ed implemetazione della classe [Gino.App.Sysfunc.sysfunc](#).

Data Structures

- class [sysfunc](#)
Classe di tipo [Gino.Controller](#) che gestisce metodi che stampano pagine di errore.

Namespaces

- [Gino\App\Sysfunc](#)
Namespace dell'applicazione [Sysfunc](#), per la gestione della visualizzazione ddi pagine di errore

9.68.1 Detailed Description

Contiene la definizione ed implemetazione della classe [Gino.App.Sysfunc.sysfunc](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.69 configuration.php File Reference

File di configurazione.

Variables

- const [DEBUG](#) TRUE
- const [SHOW_STATS](#) FALSE
- const [ADMINS](#) serialize(array('marco.guidotti@otto.to.it', 'stefano.contini@otto.to.it'))
Amministratori sistema Ricevono notifiche di errori di sistema.
- const [USE_PDO](#) false
Utilizzo della libreria PDO.
- const [DBMS](#) 'mysql'
Tipo di database.
- const [DB_HOST](#) "localhost"
Nome del server database.
- const [DB_PORT](#) "3306"
Numero della porta di connessione del server database.
- const [DB_DBNAME](#) "dbgino"
Nome del database.
- const [DB_USER](#) "root"
- const [DB_PASSWORD](#) ""
Password dell'utente del database.
- const [DB_SCHEMA](#) ""
Schema del database.
- const [DB_CHARSET](#) 'utf-8'
Codifica del database.

- const `QUERY_CACHE_TYPE` 'auto'
- const `QUERY_CACHE_SERVER` null
- const `QUERY_CACHE_PATH` null
- const `QUERY_CACHE_FALLBACK` null
- const `SESSION_NAME` 'GINO_SESSID'
Nome della sessione.
- const `MAX_FILE_SIZE` 5242880
*Dimensione massima dei file per l'upload (2Mb => 2*1024*1024)*

9.69.1 Detailed Description

File di configurazione.

Contiene i parametri dell'applicazione

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.69.2 Variable Documentation

9.69.2.1 `const ADMINS serialize(array('marco.guidotti@otto.to.it', 'stefano.contini@otto.to.it'))`

Amministratori sistema Ricevono notifiche di errori di sistema.

Il valore deve essere necessariamente un array serializzato

9.69.2.2 `const DB_CHARSET 'utf-8'`

Codifica del database.

9.69.2.3 `const DB_DBNAME "dbgino"`

Nome del database.

9.69.2.4 `const DB_HOST "localhost"`

Nome del server database.

9.69.2.5 `const DB_PASSWORD ""`

Password dell'utente del database.

9.69.2.6 `const DB_PORT "3306"`

Numero della porta di connessione del server database.

9.69.2.7 `const DB_SCHEMA ""`

Schema del database.

9.69.2.8 `const DB_USER "root"`

9.69.2.9 `const DBMS 'mysql'`

Tipo di database.

See also

`db::instance()`

9.69.2.10 `const DEBUG TRUE`

9.69.2.11 `const MAX_FILE_SIZE 5242880`

Dimensione massima dei file per l'upload (2Mb => 2*1024*1024)

9.69.2.12 `const QUERY_CACHE_FALLBACK null`

9.69.2.13 `const QUERY_CACHE_PATH null`

9.69.2.14 `const QUERY_CACHE_SERVER null`

9.69.2.15 `const QUERY_CACHE_TYPE 'auto'`

9.69.2.16 `const SESSION_NAME 'GINO_SESSID'`

Nome della sessione.

9.69.2.17 `const SHOW_STATS FALSE`

9.69.2.18 `const USE_PDO false`

Utilizzo della libreria PDO.

See also

`db::instance()`

9.70 index.php File Reference

Entry point di gino CMS.

Namespaces

- [Gino](#)

Namespace principale di gino.

Variables

- const [SITE_ROOT](#) `dirname(realpath(__FILE__))`
SITE_ROOT Percorso assoluto alla root directory
- const [SITE_WWW](#) `$site_www`
SITE_WWW Percorso relativo dell'applicazione a partire dalla root directory

9.70.1 Detailed Description

Entry point di gino CMS.

Entry point dell'applicazione. Definisce i percorsi alla root directory ed inizializza la classe [Gino.Core](#) che fornisce la risposta HTTP. Raccoglie eventuali eccezioni e le tratta attraverso la classe [Gino.Logger](#) che ha comportamenti diversi a seconda del valore della costante DEBUG impostata nel file [configuration.php](#)

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.71 lib/classes/build/class.BooleanBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.BooleanBuild](#).

Data Structures

- class [BooleanBuild](#)
Gestisce i campi di tipo BOOLEAN.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.71.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.BooleanBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.72 lib/classes/build/class.Build.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Build](#).

Data Structures

- class [Build](#)
Gestisce i campi del modello.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.72.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Build](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.73 lib/classes/build/class.CharBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.CharBuild](#).

Data Structures

- class [CharBuild](#)
Gestisce i campi di tipo stringa (CHAR, VARCHAR)

Namespaces

- [Gino](#)
Namespace principale di gino.

9.73.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.CharBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.74 lib/classes/build/class.DateBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DateBuild](#).

Data Structures

- class [DateBuild](#)
Gestisce i campi di tipo data.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.74.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DateBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.75 lib/classes/build/class.DatetimeBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DatetimeBuild](#).

Data Structures

- class [DatetimeBuild](#)
Gestisce i campi di tipo DATETIME.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.75.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DatetimeBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.76 lib/classes/build/class.DirectoryBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DirectoryBuild](#).

Data Structures

- class [DirectoryBuild](#)
Gestisce i campi di tipo DIRECTORY.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.76.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DirectoryBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.77 lib/classes/build/class.EmailBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EmailBuild](#).

Data Structures

- class [EmailBuild](#)
Campo di tipo EMAIL.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.77.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EmailBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.78 lib/classes/build/class.EnumBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EnumBuild](#).

Data Structures

- class [EnumBuild](#)
Gestisce i campi di tipo ENUM.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.78.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EnumBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.79 lib/classes/build/class.FileBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.FileBuild](#).

Data Structures

- class [FileBuild](#)
Campo di tipo FILE.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.79.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.FileBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.80 lib/classes/build/class.FloatBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.FloatField](#).

Data Structures

- class [FloatBuild](#)
Gestisce campi di tipo decimale (FLOAT, DOUBLE, DECIMAL)

Namespaces

- [Gino](#)
Namespace principale di gino.

9.80.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.FloatField](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.81 lib/classes/build/class.ForeignKeyBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ForeignKeyBuild](#).

Data Structures

- class [ForeignKeyBuild](#)
Campo di tipo chiave esterna.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.81.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ForeignKeyBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.82 lib/classes/build/class.ImageBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ImageBuild](#).

Data Structures

- class [ImageBuild](#)
Gestisce campi di tipo IMMAGINE.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.82.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ImageBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.83 lib/classes/build/class.IntegerBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.IntegerBuild](#).

Data Structures

- class [IntegerBuild](#)
Campo di tipo INTERO.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.83.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.IntegerBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.84 lib/classes/build/class.ManyToManyBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ManyToManyField](#).

Data Structures

- class [ManyToManyBuild](#)
Gestisce i campi di tipo many to many.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.84.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ManyToManyField](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.85 lib/classes/build/class.ManyToManyThroughBuild.php File Reference

Contiene la definizione ed implementation della classe [Gino.ManyToManyThroughBuild](#).

Data Structures

- class [ManyToManyThroughBuild](#)
Gestisce i campi di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.85.1 Detailed Description

Contiene la definizione ed implementation della classe [Gino.ManyToManyThroughBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.86 lib/classes/build/class.MulticheckBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.MulticheckBuild](#).

Data Structures

- class [MulticheckBuild](#)
Gestisce i campi di tipo multicheck.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.86.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.MulticheckBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.87 lib/classes/build/class.SlugBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.SlugBuild](#).

Data Structures

- class [SlugBuild](#)
Getisce i campi di tipo SLUG (CHAR, VARCHAR)

Namespaces

- [Gino](#)
Namespace principale di gino.

9.87.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.SlugBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.88 lib/classes/build/class.TagBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TagBuild](#).

Data Structures

- class [TagBuild](#)
Gestisce i campi per inserimento tag.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.88.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TagBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.89 lib/classes/build/class.TextBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TextBuild](#).

Data Structures

- class [TextBuild](#)
Campo di tipo TEXT.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.89.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TextBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.90 lib/classes/build/class.TimeBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TimeBuild](#).

Data Structures

- class [TimeBuild](#)
Gestisce i campi di tipo TIME.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.90.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TimeBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.91 lib/classes/build/class.YearBuild.php File Reference

Contiene la definizione ed implementazione della classe [Gino.YearBuild](#).

Data Structures

- class [YearBuild](#)
Gestisce i campi di tipo ANNO.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.91.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.YearBuild](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.92 lib/classes/class.Access.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Access](#).

Data Structures

- class [Access](#)

Classe per la gestione dell'autenticazione ed accesso alla funzionalità

Namespaces

- [Gino](#)

Namespace principale di gino.

9.92.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Access](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.93 lib/classes/class.AdminTable.php File Reference

Contiene la definizione ed implementazione della classe [Gino.AdminTable](#).

Data Structures

- class [AdminTable](#)

Gestisce l'interfaccia di amministrazione di un modello con inserimento, modifica ed eliminazione.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.93.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.AdminTable](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.94 lib/classes/class.Cache.php File Reference

Contiene la definizione ed implementazione delle classi [Gino.Cache](#), [Gino.OutputCache](#), [Gino.DataCache](#).

Data Structures

- class [Cache](#)
Classe che viene estesa da [Gino.OutputCache\(\)](#) e [Gino.DataCache\(\)](#)
- class [OutputCache](#)
Memorizza gli output (text, html, xml) scrivendo su file.
- class [DataCache](#)
Memorizza le strutture dati scrivendo su file.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.94.1 Detailed Description

Contiene la definizione ed implementazione delle classi [Gino.Cache](#), [Gino.OutputCache](#), [Gino.DataCache](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.95 lib/classes/class.Captcha.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Captcha](#).

Data Structures

- class [Captcha](#)
Classe per la generazione di immagini captcha.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.95.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Captcha](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.96 lib/classes/class.Compressor.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Compressor](#), [Gino.JSMin](#) e [Gino.JSMinException](#).

Data Structures

- class [Compressor](#)
Classe per la compressione di css e js (merge e minify)
- class [JSMin](#)
PHP implementation of Douglas Crockford's [JSMin](#).
- class [JSMinException](#)
[JSMinException](#).

Namespaces

- [Gino](#)
Namespace principale di gino.

9.96.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Compressor](#), [Gino.JSMin](#) e [Gino.JSMinException](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.97 lib/classes/class.Core.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Core](#).

Data Structures

- class [Core](#)

Gestisce una [Gino.Http.Request](#) ed invia una [Gino.Http.Response](#) adeguata.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.97.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Core](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.98 lib/classes/class.Css.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Css](#).

Data Structures

- class [Css](#)

Libreria per la gestione dei file css dei singoli moduli e dei file css del layout (da associare alle skin)

Namespaces

- [Gino](#)

Namespace principale di gino.

9.98.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Css](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.99 lib/classes/class.Db.php File Reference

Contiene l'interfaccia [Gino.DbManager](#) e le classi [Gino.Db](#) e [Gino.SqlParse](#).

Data Structures

- interface [DbManager](#)
Interfaccia per le librerie di connessione al database.
- class [Db](#)
Classe Factory e [Singleton](#) usata per creare oggetti che si interfacciano al database.
- class [SqlParse](#)
Classe per gestire il parser dei file sql (Code freely adapted from phpBB Group)

Namespaces

- [Gino](#)
Namespace principale di gino.

9.99.1 Detailed Description

Contiene l'interfaccia [Gino.DbManager](#) e le classi [Gino.Db](#) e [Gino.SqlParse](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.100 lib/classes/class.Document.php File Reference

Contiene la definizione ed implementazione della class [Gino.Document](#).

Data Structures

- class [Document](#)
Crea il documento html da inviare come corpo della risposta HTTP.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.100.1 Detailed Description

Contiene la definizione ed implementazione della class [Gino.Document](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.101 lib/classes/class.Error.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Error](#).

Data Structures

- class [Error](#)

Classe per la gestione di errori da mostrare all'utente E' possibile anche mostrare dei warning per facilitare il debug.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.101.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Error](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.102 lib/classes/class.EventDispatcher.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EventDispatcher](#).

Data Structures

- class [EventDispatcher](#)

Classe di tipo Mediator e [Singleton](#) per la gestione di segnali (eventi) tra oggetti.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.102.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EventDispatcher](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.103 lib/classes/class.Export.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Export](#).

Data Structures

- class [Export](#)

Libreria per l'esportazione di tabelle o dati.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.103.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Export](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.104 lib/classes/class.Form.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Form](#).

Data Structures

- class [Form](#)

Classe per la creazione ed il salvataggio dati di un form.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.104.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Form](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.105 lib/classes/class.Frontend.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Frontend](#).

Data Structures

- class [Frontend](#)

Libreria per la gestione dei file di front-end dei singoli moduli (css e viste)

Namespaces

- [Gino](#)

Namespace principale di gino.

9.105.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Frontend](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.106 lib/classes/class.GImage.php File Reference

Contiene la definizione ed implementazione della classe [Gino.GImage](#).

Data Structures

- class [GImage](#)

Classe per il trattamento di immagini.

Namespaces

- [Gino](#)

Namespace principale di gino.

Variables

- const [GIMAGE_DIR](#) CONTENT_DIR.OS.'gimage'

9.106.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.GImage](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.107 lib/classes/class.GTag.php File Reference

Contiene la definizione ed implementazione della classe [Gino.GTag](#).

Data Structures

- class [GTag](#)

Classe per il trattamento di campi di tipo tag.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.107.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.GTag](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.108 lib/classes/class.Javascript.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Javascript](#).

Data Structures

- class [Javascript](#)

Contiene i metodi per includere alcuni javascript.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.108.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Javascript](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.109 lib/classes/class.Loader.php File Reference

Contiene la definizione e l'implementazione della classe [Gino.Loader](#) ed il metodo magic `__autoload`.

Data Structures

- class [Loader](#)

[Loader](#) di classi di tipo [Gino.Model](#) e classi di sistema.

Namespaces

- [Gino](#)

Namespace principale di gino.

Functions

- [__autoload](#) (\$class)

__autoload

9.109.1 Detailed Description

Contiene la definizione e l'implementazione della classe [Gino.Loader](#) ed il metodo magic `__autoload`.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.109.2 Function Documentation

9.109.2.1 `__autoload ($class)`

`__autoload`

Carica in maniera automatica le classi di tipo [Gino.Controller](#)

Parameters

string	<i>\$class</i>	nome classe con o senza namespace
--------	----------------	-----------------------------------

Returns

void

9.110 lib/classes/class.Locale.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Locale](#).

Data Structures

- class [Locale](#)

Libreria per la gestione delle traduzioni che non utilizzano le librerie gettext.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.110.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Locale](#).

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.111 lib/classes/class.Logger.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Logger](#).

Data Structures

- class [Logger](#)
Classe per la notifica di log di sistema.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.111.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Logger](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.112 lib/classes/class.MobileDetect.php File Reference

Contiene la definizione ed implementazione della classe [Gino.MobileDetect](#).

Data Structures

- class [MobileDetect](#)
Verifica se una richiesta HTTP è stata effettuata da un dispositivo mobile.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.112.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.MobileDetect](#).

9.113 lib/classes/class.Options.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Options](#).

Data Structures

- class [Options](#)

Gestisce le opzioni di classe, costruendo il form ed effettuando l'action.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.113.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Options](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.114 lib/classes/class.Paginator.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Paginator](#).

Data Structures

- class [Paginator](#)

Gestisce la paginazione di elementi dati il numero di elementi totali ed il numero di elementi per pagina, ricava i limiti per creare il sottoinsieme di elementi da mostrare e gestisce la navigazione tra le pagine.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.114.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Paginator](#).

Copyright

© 2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.115 lib/classes/class.Registry.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Registry](#).

Data Structures

- class [Registry](#)

Registro di gino.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.115.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Registry](#).

Copyright

© 2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.116 lib/classes/class.Router.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Router](#).

Data Structures

- class [Router](#)

Gestisce il routing di una request HTTP, chiamando la classe e metodo che devono fornire risposta.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.116.1 Detailed Description

Contiene la definizione ed implementazione della class [Gino.Router](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.117 lib/classes/class.Search.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Search](#).

Data Structures

- class [Search](#)

Libreria per ricerche full text pesate sulle tabelle.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.117.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Search](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.118 lib/classes/class.Session.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Session](#).

Data Structures

- class [Session](#)

Classe per la gestione delle variabili di sessione.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.118.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Session](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.119 lib/classes/class.Singleton.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Singleton](#).

Data Structures

- class [Singleton](#)

Classe astratta per implementazione del [Singleton](#) pattern.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.119.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Singleton](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.120 lib/classes/class.Skin.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Skin](#).

Data Structures

- class [Skin](#)
Libreria per la gestione delle [Skin](#).

Namespaces

- [Gino](#)
Namespace principale di gino.

9.120.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Skin](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.121 lib/classes/class.Template.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Template](#).

Data Structures

- class [Template](#)
Libreria per la gestione dei template del documento html da associare alle [Gino](#).

Namespaces

- [Gino](#)
Namespace principale di gino.

9.121.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Template](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.122 lib/classes/class.Translation.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Translation](#).

Data Structures

- class [Translation](#)
Classe per la gestione delle traduzioni.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.122.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Translation](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.123 lib/classes/exceptions/class.Exception403.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Exception.Exception403](#).

Data Structures

- class [Exception403](#)
Classe per la gestione di eccezioni 403.

Namespaces

- [Gino\Exception](#)

Namespace che comprende tutte le classi di tipo [Exception](#)

9.123.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Exception.Exception403](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.124 lib/classes/exceptions/class.Exception404.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Exception.Exception404](#).

Data Structures

- class [Exception404](#)

Classe per la gestione di eccezioni 404.

Namespaces

- [Gino\Exception](#)

Namespace che comprende tutte le classi di tipo [Exception](#)

9.124.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Exception.Exception404](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.125 lib/classes/exceptions/class.Exception500.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Exception.Exception500](#).

Data Structures

- class [Exception500](#)
Classe per la gestione di eccezioni 500.

Namespaces

- [Gino\Exception](#)
Namespace che comprende tutte le classi di tipo [Exception](#)

9.125.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Exception.Exception500](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.126 lib/classes/fields/class.BooleanField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.BooleanField](#).

Data Structures

- class [BooleanField](#)
Rappresenta campi di tipo BOOLEAN.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.126.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.BooleanField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.127 lib/classes/fields/class.CharField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.CharField](#).

Data Structures

- class [CharField](#)
Campi di tipo stringa (CHAR, VARCHAR)

Namespaces

- [Gino](#)
Namespace principale di gino.

9.127.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.CharField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.128 lib/classes/fields/class.DateField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DateField](#).

Data Structures

- class [DateField](#)
Campo di tipo data.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.128.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DateField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.129 lib/classes/fields/class.DatetimeField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DatetimeField](#).

Data Structures

- class [DatetimeField](#)
Campo di tipo DATETIME.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.129.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DatetimeField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.130 lib/classes/fields/class.DirectoryField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DirectoryField](#).

Data Structures

- class [DirectoryField](#)
Campo di tipo DIRECTORY.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.130.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DirectoryField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.131 lib/classes/fields/class.EmailField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EmailField](#).

Data Structures

- class [EmailField](#)
Campo di tipo EMAIL.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.131.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EmailField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.132 lib/classes/fields/class.EnumField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EnumField](#).

Data Structures

- class [EnumField](#)
Campo di tipo ENUM.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.132.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EnumField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.133 lib/classes/fields/class.Field.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Field](#).

Data Structures

- class [Field](#)

Gestisce le caratteristiche del tipo di campo (colonne)

Namespaces

- [Gino](#)

Namespace principale di gino.

9.133.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Field](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.134 lib/classes/fields/class.FileField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.FileField](#).

Data Structures

- class [FileField](#)

Campo di tipo FILE.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.134.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.FileField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.135 lib/classes/fields/class.FloatField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.FloatField](#).

Data Structures

- class [FloatField](#)

Campo di tipo decimale (FLOAT, DOUBLE, DECIMAL)

Namespaces

- [Gino](#)

Namespace principale di gino.

9.135.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.FloatField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.136 lib/classes/fields/class.ForeignKeyField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ForeignKeyField](#).

Data Structures

- class [ForeignKeyField](#)
Campo di tipo chiave esterna.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.136.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ForeignKeyField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.137 lib/classes/fields/class.ImageField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ImageField](#).

Data Structures

- class [ImageField](#)
Campo di tipo IMMAGINE.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.137.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ImageField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.138 lib/classes/fields/class.IntegerField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.IntegerField](#).

Data Structures

- class [IntegerField](#)
Campo di tipo INTERO.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.138.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.IntegerField](#).

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.139 lib/classes/fields/class.ManyToManyField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ManyToManyField](#).

Data Structures

- class [ManyToManyField](#)
Campo di tipo many to many.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.139.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ManyToManyField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.140 lib/classes/fields/class.ManyToManyThroughField.php File Reference

Contiene la definizione ed implementation della classe [Gino.ManyToManyThroughField](#).

Data Structures

- class [ManyToManyThroughField](#)

Campo di tipo many to many con associazione attraverso un modello che porta informazioni aggiuntive.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.140.1 Detailed Description

Contiene la definizione ed implementation della classe [Gino.ManyToManyThroughField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.141 lib/classes/fields/class.MulticheckField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.MulticheckField](#).

Data Structures

- class [MulticheckField](#)

Campo di tipo multicheck.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.141.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.MulticheckField](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.142 lib/classes/fields/class.SlugField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.SlugField](#).

Data Structures

- class [SlugField](#)

Campo di tipo SLUG (CHAR, VARCHAR)

Namespaces

- [Gino](#)

Namespace principale di gino.

9.142.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.SlugField](#).

Copyright

2014-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.143 lib/classes/fields/class.TagField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TagField](#).

Data Structures

- class [TagField](#)

Classe per la gestione di campi per inserimento tag.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.143.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TagField](#).

Copyright

2014-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.144 lib/classes/fields/class.TextField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TextField](#).

Data Structures

- class [TextField](#)

Campo di tipo TEXT.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.144.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TextField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.145 lib/classes/fields/class.TimeField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TimeField](#).

Data Structures

- class [TimeField](#)
Campo di tipo TIME.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.145.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TimeField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.146 lib/classes/fields/class.YearField.php File Reference

Contiene la definizione ed implementazione della classe [Gino.YearField](#).

Data Structures

- class [YearField](#)
Campo di tipo ANNO.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.146.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.YearField](#).

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.147 lib/classes/http/class.Redirect.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.Redirect](#).

Data Structures

- class [Redirect](#)

Subclass di [Gino.Http.Response](#) per gestire reindirizzamenti.

Namespaces

- [Gino\Http](#)

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.147.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.Redirect](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.148 lib/classes/http/class.Request.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.Request](#).

Data Structures

- class [Request](#)

Wrapper di una richiesta HTTP.

Namespaces

- [Gino\Http](#)

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.148.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.Request](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.149 lib/classes/http/class.Response.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.Response](#).

Data Structures

- class [Response](#)
Wrapper di una risposta HTTP.

Namespaces

- [Gino\Http](#)
Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.149.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.Response](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.150 lib/classes/http/class.ResponseFile.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseFile](#).

Data Structures

- class [ResponseFile](#)
Subclass di per gestire stream di file.

Namespaces

- [Gino\Http](#)

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.150.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseFile](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.151 lib/classes/http/class.ResponseForbidden.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseForbidden](#).

Data Structures

- class [ResponseForbidden](#)

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 403.

Namespaces

- [Gino\Http](#)

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.151.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseForbidden](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.152 lib/classes/http/class.ResponseJson.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseJson](#).

Data Structures

- class [ResponseJson](#)

Subclass di per gestire risposte in formato json.

Namespaces

- [Gino\Http](#)

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.152.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseJson](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.153 lib/classes/http/class.ResponseNotFound.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseNotFound](#).

Data Structures

- class [ResponseNotFound](#)

Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori 404.

Namespaces

- [Gino\Http](#)

Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.153.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseNotFound](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.154 lib/classes/http/class.ResponsePdf.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.ResponsePdf](#).

Data Structures

- class [ResponsePdf](#)
Subclass di per gestire risposte in formato pdf.

Namespaces

- [Gino\Http](#)
Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.154.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.ResponsePdf](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.155 lib/classes/http/class.ResponseServerError.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseServerError](#).

Data Structures

- class [ResponseServerError](#)
Subclass di [Gino.Http.Response](#) per gestire risposte a seguito di errori interni (code 500)

Namespaces

- [Gino\Http](#)
Namespace che comprende tutte le classi per la gestione di richieste e risposte http

9.155.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Http.ResponseServerError](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.156 lib/classes/mvc/class.Controller.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Controller](#).

Data Structures

- class [Controller](#)

Classe astratta primitiva di tipo [Controller](#) (MVC), dalla quale tutti i controller delle singole app discendono.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.156.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Controller](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.157 lib/classes/mvc/class.Model.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Model](#).

Data Structures

- class [Model](#)

Classe astratta che definisce un modello, cioè un oggetto che rappresenta una tabella su database.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.157.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Model](#).

Copyright

2014-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.158 lib/classes/mvc/class.View.php File Reference

Contiene la definizione ed implementazione della classe [Gino.View](#).

Data Structures

- class [View](#)

Gestisce le viste, impostando il template e ritornando l'output.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.158.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.View](#).

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.159 lib/classes/widget/class.CheckboxWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.CheckboxWidget](#).

Data Structures

- class [CheckboxWidget](#)

Campi checkbox nei form.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.159.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.CheckboxWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.160 lib/classes/widget/class.ConstantWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ConstantWidget](#).

Data Structures

- class [ConstantWidget](#)
Campi di tipo costante nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.160.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ConstantWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.161 lib/classes/widget/class.DatetimeWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DatetimeWidget](#).

Data Structures

- class [DatetimeWidget](#)
Campi di tipo datetime nei form.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.161.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DatetimeWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.162 lib/classes/widget/class.DateWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.DateWidget](#).

Data Structures

- class [DateWidget](#)

Campi di tipo date nei form.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.162.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.DateWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.163 lib/classes/widget/class.EditorWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EditorWidget](#).

Data Structures

- class [EditorWidget](#)

Campi di tipo editor nei form.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.163.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EditorWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.164 lib/classes/widget/class.EmailWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.EmailWidget](#).

Data Structures

- class [EmailWidget](#)

Campi di tipo email nei form.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.164.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.EmailWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.165 lib/classes/widget/class.FileWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.FileWidget](#).

Data Structures

- class [FileWidget](#)
Campi di tipo file nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.165.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.FileWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.166 lib/classes/widget/class.FloatWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.FloatWidget](#).

Data Structures

- class [FloatWidget](#)
Campi di tipo float nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.166.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.FloatWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.167 lib/classes/widget/class.HiddenWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.HiddenWidget](#).

Data Structures

- class [HiddenWidget](#)
Campi di tipo hidden nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.167.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.HiddenWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.168 lib/classes/widget/class.ImageWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.ImageWidget](#).

Data Structures

- class [ImageWidget](#)
Campi di tipo immagine nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.168.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.ImageWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.169 lib/classes/widget/class.MulticheckWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.MulticheckWidget](#).

Data Structures

- class [MulticheckWidget](#)
Campi di tipo multicheck nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.169.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.MulticheckWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.170 lib/classes/widget/class.PasswordWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.PasswordWidget](#).

Data Structures

- class [PasswordWidget](#)
Campi di tipo password nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.170.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.PasswordWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.171 lib/classes/widget/class.RadioWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.RadioWidget](#).

Data Structures

- class [RadioWidget](#)
Campi di tipo radio button nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.171.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.RadioWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.172 lib/classes/widget/class.SelectWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.SelectWidget](#).

Data Structures

- class [SelectWidget](#)
Campi di tipo select nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.172.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.SelectWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.173 lib/classes/widget/class.TextareaWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TextareaWidget](#).

Data Structures

- class [TextareaWidget](#)
Campi di tipo textarea nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.173.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TextareaWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.174 lib/classes/widget/class.TextWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TextWidget](#).

Data Structures

- class [TextWidget](#)
Campi di tipo testo nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.174.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TextWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.175 lib/classes/widget/class.TimeWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.TimeWidget](#).

Data Structures

- class [TimeWidget](#)
Campi di tipo orario nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.175.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.TimeWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.176 lib/classes/widget/class.UnitWidget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.UnitWidget](#).

Data Structures

- class [UnitWidget](#)
Insieme di campi di un modello.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.176.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.UnitWidget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.177 lib/classes/widget/class.Widget.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Widget](#).

Data Structures

- class [Widget](#)
Gestisce i campi nei form.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.177.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Widget](#).

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.178 lib/const.php File Reference

Alcune costanti utilizzate dal sistema.

Namespaces

- [Gino](#)
Namespace principale di gino.

Variables

- const [TBL_MODULE](#) 'sys_module'
Nome della tabella che contiene le istanze di moduli.
- const [TBL_MODULE_APP](#) 'sys_module_app'
Nome della tabella che contiene i moduli installati.
- const [TBL_SYS_CONF](#) 'sys_conf'
Nome della tabella che contiene la configurazione del sistema.
- const [TBL_LOG_ACCESS](#) 'sys_log_access'
Nome della tabella che contiene il log degli accessi.
- const [TBL_USER](#) 'auth_user'
Nome della tabella utenti.
- const [TBL_USER_ADD](#) 'auth_user_add'

- Nome della tabella informazioni aggiuntive utenti.*
- const **TBL_USER_GROUP** 'auth_user_group'
- Nome della tabella di join utenti - gruppi.*
- const **TBL_GROUP** 'auth_group'
- Nome della tabella dei gruppi di utenti.*
- const **TBL_GROUP_PERMISSION** 'auth_group_perm'
- Nome della tabella di join gruppi - permessi.*
- const **TBL_PERMISSION** 'auth_permission'
- Nome della tabella dei permessi.*
- const **TBL_USER_PERMISSION** 'auth_user_perm'
- Nome della tabella di join utenti - permessi.*
- const **TBL_REGISTRATION_PROFILE** 'auth_registration_profile'
- Nome della tabella dei profili di registrazione utenti.*
- const **TBL_REGISTRATION_PROFILE_GROUP** 'auth_registration_profile_group'
- Nome della tabella di associazione profili di registrazione utenti - gruppi.*
- const **TBL_REGISTRATION_REQUEST** 'auth_registration_request'
- Nome della tabella delle richieste di registrazione.*
- const **TBL_LANGUAGE** 'language'
- Nome della tabella lingue.*
- const **TBL_TRANSLATION** 'language_translation'
- Nome della tabella traduzioni.*
- const **TBL_NATION** 'nation'
- Nome della tabella nazioni.*

9.178.1 Detailed Description

Alcune costanti utilizzate dal sistema.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

9.179 lib/custom_ckeditor/config.js File Reference

Variables

- CKEDITOR [editorConfig](#)
- Copyright (c) 2003-2014, CKSource - Frederico Knabben.*

9.179.1 Variable Documentation

9.179.1.1 CKEDITOR editorConfig

Initial value:

```
= function( config ) {  
  
    config.removePlugins = 'forms';  
  
    config.addStylesSet = [  
    ];  
  
    var base_href = $$('base')[0].get('href');  
    config.extraPlugins = 'imagebrowser';  
    config.imageBrowser_listUrl = base_href + 'attachment/jsonImageList';  
}
```

Copyright (c) 2003-2014, CKSource - Frederico Knabben.

All rights reserved. For licensing, see LICENSE.md or <http://ckeditor.com/license>

9.180 lib/func.browser.php File Reference

Racchiude funzioni per il detect di browser, piattaforma etc...

Namespaces

- [Gino](#)
Namespace principale di gino.

Functions

- [browsers](#) ()
Lista di browser popolari.
- [robots](#) ()
Lista di web robots popolari.
- [platforms](#) ()
Lista di piattaforme popolari.
- [get_browser_info](#) (\$arg=", \$agent=")
Informazioni browser.
- [is_browser](#) (\$name)
Verifica se il browser è quello dato come argomento.
- [is_browser_version](#) (\$version)
Verifica la versione del browser.
- [is_browser_platform](#) (\$platform)
Verifica la piattaforma del browser.
- [is_robot](#) ()
Verifica se il client è un robot.

9.180.1 Detailed Description

Racchiude funzioni per il detect di browser, piattaforma etc...

Funzione originale di Linesh Jose, <http://lineshjose.com>

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

Linesh Jose lineshjose@gmail.com
marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.181 lib/func.file.php File Reference

Racchiude le funzioni generali di manipolazione file e directory utilizzate da gino.

Namespaces

- [Gino](#)
Namespace principale di gino.

Functions

- [searchNameFile](#) (\$dir)
Lista files contenuti in una directory.
- [download](#) (\$full_path)
Forza il download di un file.
- [extension](#) (\$filename, \$extensions)
Controlla le estensioni dei file.
- [baseFileName](#) (\$filename)
Ricava il nome del file senza l'estensione.
- [deleteFileDir](#) (\$dir, \$delete_dir=TRUE)
Elimina ricorsivamente i file e le directory.
- [extensionFile](#) (\$filename)
Estensione di un file.
- [verifyExtension](#) (\$filename, \$extensions)
Controlla se l'estensione di un file è valida.

9.181.1 Detailed Description

Racchiude le funzioni generali di manipolazione file e directory utilizzate da gino.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.182 lib/func.php File Reference

Racchiude funzioni generali utilizzate da gino.

Namespaces

- [Gino](#)
Namespace principale di gino.

Functions

- [relativePath](#) (\$abspath)
Ricava il percorso relativo a partire da un percorso assoluto.
- [absolutePath](#) (\$relpath)
Ricava il percorso assoluto a partire da un percorso relativo.
- [gOpt](#) (\$opt_name, \$opt_array, \$default)
Restituisce l'elemento di un array corrispondente alla chiave data oppure un valore di default.
- [arrayToObject](#) (array \$array)
Trasforma un array in un oggetto.
- [checkEmail](#) (\$value, \$regexp=null)
Verifica la validità dell'indirizzo email.
- [dateToDbDate](#) (\$date, \$s='/')
Formatta la data per il database (YYYY-MM-DD)
- [dbDateToDate](#) (\$db_date, \$s='/', \$num_year=4)
Converte il formato della data da database (campo DATE) in un formato di facile visualizzazione (DD/MM/YYYY)
- [dbDatetimeToDate](#) (\$datetime, \$s='/', \$num_year=4)
Converte il formato della data da database (campo DATETIME) in un formato di facile visualizzazione (DD/MM/YYYY)
- [dbDatetimeToTime](#) (\$datetime)
Riporta l'orario di un campo DATETIME (HH:MM:SS)
- [dbTimeToTime](#) (\$db_time, \$seconds=false)
Mostra l'orario (HH:MM[:SS])
- [timeToDbTime](#) (\$time)
Formatta l'orario per il database (HH:MM:SS)
- [dbNumberToNumber](#) (\$number, \$decimals=2)
Formatta un numero con il raggruppamento delle centinaia.
- [numberToDB](#) (\$number)
Formatta un numero per il database (il separatore decimale è il punto)
- [isNumeric](#) (\$variable)
Controlla se una variabile è un numero o una stringa numerica.
- [timeDiff](#) (\$firstTime, \$lastTime)
Calcola l'intervallo di tempo in secondi tra due valori datetime.
- [dateDiff](#) (\$interval, \$datefrom, \$dateto, \$using_timestamps=false)
Calcola la differenza di tempo tra due datetime in più formati.
- [getDateDiff](#) (\$start_date, \$end_date=null, \$options=array())
Calcola la differenza di tempo tra due datetime in più formati.
- [isValid](#) (\$type, \$var)
Verifica se il valore della variabile è conforme al tipo di controllo indicato.
- [cutHtmlText](#) (\$html, \$length, \$sending, \$strip_tags, \$cut_words, \$cut_images, \$options=null)
Accorcia un testo HTML alla lunghezza desiderata (length)

- [cutString](#) (\$string, \$max_char, \$word_complete=true, \$file=false)
Limita i caratteri di una stringa.
- [listProv](#) ()
Elenco delle province.
- [share](#) (\$site, \$url, \$title=null, \$description=null)
Codice html di condivisione social network.
- [shareAll](#) (\$social, \$url, \$title=null, \$description=null)
Codice condivisione social network con integrazione sharethis.
- [traslitterazione](#) (\$numero, \$decimale=false)
Converte un numero in cifre.
- [convertSize](#) (\$size)
Converte la dimensione di un file in un formato leggibile (KB,GB,...)
- [enabledPng](#) ()
Verifica la validità del supporto PNG.
- [enabledZip](#) ()
Verifica la validità della classe ZipArchive.
- [cryptMethod](#) (\$string, \$crypt)
Crypta la password dell'utente.
- [emailPolicy](#) ()
Testo della policy di una email.
- [obj_serialize](#) (\$instanceName, \$object)
Operazione di serializzazione.
- [obj_unserialize](#) (\$instanceName)
Operazione di deserializzazione.
- [icon](#) (\$name, \$options=array())
Icone.

9.182.1 Detailed Description

Racchiude funzioni generali utilizzate da gino.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.183 lib/func.var.php File Reference

Racchiude le librerie per il trattamento di stringhe e dei valori da e per un input/database.

Namespaces

- [Gino](#)
Namespace principale di gino.

Functions

- [convertToHtml](#) (\$value)
Converte l'encoding di un valore preso da un campo di un database (non UTF-8) nella codifica UTF-8.
- [convertToDatabase](#) (\$value, \$character_set=null)
Converte l'encoding di un valore da html (es UTF-8) a un encoding valido per il database.
- [replaceChar](#) (\$text)
Sostituzione caratteri "strani" di word.
- [strip_tags_attributes](#) (\$text, \$strip_js, \$strip_attributes)
Rimuove gli attributi javascript insicuri nei tag html.
- [remove_attributes](#) (\$text)
Rimuove l'attributo class quando vuoto.
- [strip_selected_tags](#) (\$text, \$tags="", \$stripContent=false)
Rimuove i tag indicati.
- [strip_invisible_tags](#) (\$text)
Rimuove script, stili e altro testo indesiderato (e invisibile) tra i tag.
- [stripAll](#) (\$text)
Sostituzione entities con il carattere corrispondente.
- [stripEditor](#) (\$text)
Sostituisce alcuni tag utilizzati dall'editor html con i tag standard.
- [cleanVar](#) (\$method, \$name, \$type, \$strip_tags= "", \$options=array())
Modifica il valore presente in un campo del form per inserirlo nel database.
- [clean_sequence](#) (\$text, \$strip_tags, \$options)
Esegue una serie di operazioni sul testo per renderlo compatibile con gino e col database.
- [cleanVarEditor](#) (\$method, \$name, \$strip_tags)
Modifica il valore presente in un campo del form di tipo editor html per inserirlo nel database.
- [utf8_urldecode](#) (\$str)
Conversione dei dati Unicode \$_GET/\$_POST, generati dalla funzione javascript escape(), in UTF8 per il processo server-side.
- [codeDb](#) (\$text)
Prepara il testo da inserire nel database senza rimuovere il codice html.
- [codeToDB](#) (\$method, \$name, \$options=array())
Modifica il valore presente in un campo testo del form per inserirlo nel database.
- [htmlChars](#) (\$string, \$id="", \$options=array())
Modifica il valore di un campo testo del database per visualizzarlo in HTML.
- [preCodeParser](#) (\$string)
Modifica il valore di un campo testo di tipo "codice" del database per visualizzarlo in HTML.
- [codeParser](#) (\$string, \$id="")
Modifica il valore di un campo testo del database per visualizzarlo in HTML.
- [slimboxReplace](#) (\$string, \$id)
Modifica il valore di un campo testo del database per attivare le librerie slimbox.
- [htmlCharsText](#) (\$string)
Mostra il valore di un campo testo del database formattato come "solo testo".
- [htmlInput](#) (\$string)
Modifica il valore di un campo testo del database per visualizzarlo in un input form.
- [htmlInputEditor](#) (\$string)
Modifica il valore di un campo testo del database per visualizzarlo in un input form di tipo editor html.
- [codeInput](#) (\$string)
Modifica il valore di un campo testo del database per visualizzarlo in un input form di tipo "solo testo".
- [enclosedField](#) (\$string)
Racchiude il testo tra virgolette singole.

- `jsVar` (\$string, \$newline=false)
Escape testo che deve essere passato come variabile javascript.
- `attributeVar` (\$string)
Escape testo che deve essere racchiuso in attributi html.
- `htmlToEntities` (\$string)
Converte le entities HTML, ma non i tag.
- `encode_params` (\$params)
Codifica i parametri url.
- `decode_params` (\$params)
Decodifica i parametri url.

9.183.1 Detailed Description

Racchiude le librerie per il trattamento di stringhe e dei valori da e per un input/database.

Copyright

2005-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

9.184 lib/global.php File Reference

Set di funzioni definite nel ROOT namespace.

Functions

- `get_name_class` (\$class)
Recupera il nome della classe senza namespace.
- `set_name_class` (\$class, \$namespace= "Gino\\")
Imposta il nome della classe col suo namespace.
- `get_app_namespace` (\$controller_name)
Nome del namespace di una classe di tipo `Gino.Controller`.
- `get_app_name_class_ns` (\$controller_name)
Nome della classe di tipo `Gino.Controller` con namespace completo.
- `get_model_app_name_class_ns` (\$controller_name, \$model_name)
Nome della classe di tipo `Gino.Model` con namespace completo.

Variables

- `if(!extension_loaded('gettext')) else`

9.184.1 Detailed Description

Set di funzioni definite nel ROOT namespace.

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.184.2 Function Documentation

9.184.2.1 `get_app_name_class_ns ($controller_name)`

Nome della classe di tipo [Gino.Controller](#) con namespace completo.

Parameters

string	<i>\$controller_name</i>	nome della classe controller
--------	--------------------------	------------------------------

Returns

nome classe completo di namespace

9.184.2.2 `get_app_namespace ($controller_name)`

Nome del namespace di una classe di tipo [Gino.Controller](#).

Parameters

string	<i>\$controller_name</i>	nome della classe controller
--------	--------------------------	------------------------------

Returns

namespace

9.184.2.3 `get_model_app_name_class_ns ($controller_name, $model_name)`

Nome della classe di tipo [Gino.Model](#) con namespace completo.

Parameters

string	<i>\$controller_name</i>	nome della classe controller
string	<i>\$model_name</i>	nome del modello

Returns

nome classe completo di namespace

9.184.2.4 `get_name_class ($class)`

Recupera il nome della classe senza namespace.

Parameters

object string	<i>\$class</i>	oggetto o nome completo della classe
--------------------	----------------	--------------------------------------

Returns

nome classe senza namespace

9.184.2.5 `set_name_class ($class, $namespace = '\Gino\')`

Imposta il nome della classe col suo namespace.

Parameters

string	<i>\$class</i>	nome della classe
string	<i>\$namespace</i>	nome del namespace, default

Returns

nome classe completo di namespace

9.184.3 Variable Documentation

9.184.3.1 `if (!extension_loaded('gettext')) else`

Initial value:

```
{
    $domain='messages'
```

9.185 lib/js/abiMap.js File Reference

9.186 lib/js/CodeMirror/clike.js File Reference

Functions

- CodeMirror [defineMode](#) ("clike", function(config, parserConfig){var indentUnit=config.indentUnit, statement←IndentUnit=parserConfig.statementIndentUnit||indentUnit, dontAlignCalls=parserConfig.dontAlignCalls, keywords=parserConfig.keywords||[], builtin=parserConfig.builtin||[], blockKeywords=parserConfig.block←Keywords||[], atoms=parserConfig.atoms||[], hooks=parserConfig.hooks||[], multiLineStrings=parser←Config.multiLineStrings;var isOperatorChar=/[+\-*&%=<>!?|\/]/;var curPunc;function tokenBase(stream, state){var ch=stream.next();if(hooks[ch]){var result=hooks[ch](stream, state);if(result!==false) return result;}if(ch==" " || ch=="\n") { state.tokenize = tokenString(ch); return state.tokenize(stream, state); } if (/([\[\]\{\}\(\)\;\:,\.\.\/]/.test(ch)) { curPunc = ch; return null; } if (/^\/\s/.test(ch)) { stream.eatWhile(/[\w\._]/); return "number"; } if (ch == "/") { if (stream.eat("*")) { state.tokenize = tokenComment; return tokenComment(stream, state); } if (stream.eat("/")) { stream.skipToEnd(); return "comment"; } } if (isOperatorChar.test(ch)) { stream.eatWhile(isOperatorChar); return "operator"; } stream.eatWhile(/[\w\\$_]/); var cur = stream.current(); if (keywords.propertyIsEnumerable(cur)) { if (blockKeywords.propertyIsEnumerable(cur)) curPunc = "new-statement"; return "keyword"; } if (builtin.propertyIsEnumerable(cur)) { if (blockKeywords.propertyIsEnumerable(cur)) curPunc = "newstatement"; return "builtin"; } if (atoms.propertyIsEnumerable(cur)) return "atom"; return "variable"; } function tokenString(quote) { return function(stream, state) { var escaped = false, next, end = false; while ((next = stream.next()) != null) { if (next == quote && !escaped) { end = true; break; } escaped = !escaped && next == "\""; } if (end || !(escaped || multiLineStrings)) state.tokenize = null; return "string"; }; } function tokenComment(stream, state) { var maybeEnd = false, ch; while (ch = stream.next()) { if

```
(ch == "/" && maybeEnd) { state.tokenize = null; break; } maybeEnd = (ch == "*"); } return "comment"; } function Context(indented, column, type, align, prev) { this.indented = indented; this.column = column; this.type = type; this.align = align; this.prev = prev; } function pushContext(state, col, type) { var indent = state.indented; if (state.context && state.context.type == "statement") indent = state.context.indented; return state.context = new Context(indent, col, type, null, state.context); } function popContext(state) { var t = state.context.type; if (t == ")" || t == "]" || t == ";") state.indented = state.context.indented; return state.context = state.context.prev; } return { startState: function(basecolumn) { return { tokenize: null, context: new Context((basecolumn || 0) - indentUnit, 0, "top", false), indented: 0, startOfLine: true }; }, token: function(stream, state) { var ctx = state.context; if (stream.sol()) { if (ctx.align == null) ctx.align = false; state.indented = stream.indentation(); state.startOfLine = true; } if (stream.eatSpace()) return null; curPunc = null; var style = (state.tokenize || tokenBase)(stream, state); if (style == "comment" || style == "meta") return style; if (ctx.align == null) ctx.align = true; if ((curPunc == ";" || curPunc == ":" || curPunc == ",") && ctx.type == "statement") popContext(state); else if (curPunc == "{") pushContext(state, stream.column(), ""); else if (curPunc == "[") pushContext(state, stream.column(), ""); else if (curPunc == "(") pushContext(state, stream.column(), ""); else if (curPunc == ")") { while (ctx.type == "statement") ctx = popContext(state); if (ctx.type == ";") ctx = popContext(state); while (ctx.type == "statement") ctx = popContext(state); } else if (curPunc == ctx.type) popContext(state); else if (((ctx.type == ";" || ctx.type == "top") && curPunc != ';') || (ctx.type == "statement" && curPunc == "newstatement")) pushContext(state, stream.column(), "statement"); state.startOfLine = false; return style; }, indent: function(state, textAfter) { if (state.tokenize != tokenBase && state.tokenize != null) return CodeMirror.Pass; var ctx = state.context, firstChar = textAfter && textAfter.charAt(0); if (ctx.type == "statement" && firstChar == ";") ctx = ctx.prev; var closing = firstChar == ctx.type; if (ctx.type == "statement") return ctx.indented + (firstChar == "{" ? 0 : statementIndentUnit); else if (ctx.align && (!dontAlignCalls || ctx.type != ";")) return ctx.column + (closing ? 0 : 1); else if (ctx.type == ";") && !closing return ctx.indented + statementIndentUnit; else return ctx.indented + (closing ? 0 : indentUnit); }, electricChars: "{}", blockCommentStart: "/*", blockCommentEnd: "*/", lineComment: "//", fold: "brace" }; }
```

- function `cppHook` (stream, state)
- function `tokenAtString` (stream, state)
- function `mimes` (ms, mode)
- `mimes` (["text/x-csrc", "text/x-c", "text/x-chdr"], {name: "clike", keywords: words(cKeywords), blockKeywords: words("case do else for if switch while struct"), atoms: words("null"), hooks: {"#": `cppHook`}})
- `CodeMirror.defineMIME` ("text/x-java", {name: "clike", keywords: words("abstract assert boolean break byte case catch char class const continue default "+"do double else enum extends final finally float for goto if implements import "+"instanceof int interface long native new package private protected public "+"return short static strictfp super switch synchronized this throw throws transient "+"try void volatile while"), blockKeywords: words("catch class do else finally for if switch try while"), atoms: words("true false null"), hooks: {"@": function(stream){stream.eatWhile(/[w\$_]/);return "meta";}}})

9.186.1 Function Documentation

9.186.1.1 function `cppHook` (stream, state)

9.186.1.2 `CodeMirror.defineMIME` ("text/x-java" , {name: "clike", keywords: words("abstract assert boolean break byte case catch char class const continue default "+"do double else enum extends final finally float for goto if implements import "+"instanceof int interface long native new package private protected public "+"return short static strictfp super switch synchronized this throw throws transient "+"try void volatile while"), blockKeywords: words("catch class do else finally for if switch try while"), atoms: words("true false null"), hooks: {"@": function(stream){stream.eatWhile(/[w\$_]/);return "meta";}} })

```

9.186.1.3 CodeMirror.defineMode( "clike", function(config, parserConfig){var indentUnit=config.indentUnit,
statementIndentUnit=parserConfig.statementIndentUnit||indentUnit, dontAlignCalls=parserConfig.dontAlignCalls,
keywords=parserConfig.keywords||[], builtin=parserConfig.builtin||[],
blockKeywords=parserConfig.blockKeywords||[], atoms=parserConfig.atoms||[], hooks=parserConfig.hooks||[],
multiLineStrings=parserConfig.multiLineStrings;var isOperatorChar=/[\+\-\*\&%=<>!\?|\]/;var curPunc;function
tokenBase(stream, state){var ch=stream.next();if(hooks[ch]){var result=hooks[ch](stream, state);if(result!==false)
return result;}if(ch=="'"||ch=="") { state.tokenize = tokenString(ch); return state.tokenize(stream, state);
} if (/[\[\]\{\}\(\),;\:\.]/.test(ch)) { curPunc = ch; return null; } if (/[/]/.test(ch)) { stream.eatWhile(/[\w\$_]/); return
"number"; } if (ch == "/" ) { if (stream.eat("*")) { state.tokenize = tokenComment; return tokenComment(stream,
state); } if (stream.eat("/")) { stream.skipToEnd(); return "comment"; } } if (isOperatorChar.test(ch)) {
stream.eatWhile(isOperatorChar); return "operator"; } stream.eatWhile(/[\w\$_]/); var cur = stream.current(); if
(keywords.propertyIsEnumerable(cur)) { if (blockKeywords.propertyIsEnumerable(cur)) curPunc = "newstatement";
return "keyword"; } if (builtin.propertyIsEnumerable(cur)) { if (blockKeywords.propertyIsEnumerable(cur)) curPunc =
"newstatement"; return "builtin"; } if (atoms.propertyIsEnumerable(cur)) return "atom"; return "variable"; }
function tokenString(quote) { return function(stream, state) { var escaped = false, next, end = false; while
((next = stream.next()) != null) { if (next == quote && !escaped) {end = true; break;} escaped = !escaped &&
next == "\""; } if (end || !(escaped || multiLineStrings)) state.tokenize = null; return "string"; }; } function
tokenComment(stream, state) { var maybeEnd = false, ch; while (ch = stream.next()) { if (ch == "/" && maybeEnd)
{ state.tokenize = null; break; } maybeEnd = (ch == "*"); } return "comment"; } function Context(indented,
column, type, align, prev) { this.indented = indented; this.column = column; this.type = type; this.align =
align; this.prev = prev; } function pushContext(state, col, type) { var indent = state.indented; if (state.context &&
state.context.type == "statement") indent = state.context.indented; return state.context = new Context(indent,
col, type, null, state.context); } function popContext(state) { var t = state.context.type; if (t == ")" || t ==
"]" || t == "}") state.indented = state.context.indented; return state.context = state.context.prev; } return {
startState: function(basecolumn) { return { tokenize: null, context: new Context((basecolumn || 0) - indentUnit,
0, "top", false), indented: 0, startOfLine: true }; }, token: function(stream, state) { var ctx = state.context; if
(stream.sol()) { if (ctx.align == null) ctx.align = false; state.indented = stream.indentation(); state.startOfLine =
true; } if (stream.eatSpace()) return null; curPunc = null; var style = (state.tokenize || tokenBase)(stream,
state); if (style == "comment" || style == "meta") return style; if (ctx.align == null) ctx.align = true; if ((curPunc ==
";" || curPunc == ":" || curPunc == ",") && ctx.type == "statement") popContext(state); else if (curPunc ==
"{") pushContext(state, stream.column(), "}"); else if (curPunc == "[") pushContext(state, stream.column(), "]");
else if (curPunc == "(") pushContext(state, stream.column(), ")"); else if (curPunc == "}") { while (ctx.type ==
"statement") ctx = popContext(state); if (ctx.type == "}") ctx = popContext(state); while (ctx.type == "statement")
ctx = popContext(state); } else if (curPunc == ctx.type) popContext(state); else if (((ctx.type == "}" || ctx.type
== "top") && curPunc != ";") || (ctx.type == "statement" && curPunc == "newstatement")) pushContext(state,
stream.column(), "statement"); state.startOfLine = false; return style; }, indent: function(state,
textAfter){if(state.tokenize!=tokenBase &&state.tokenize!=null) return CodeMirror.Pass;var ctx=state.context,
firstChar=textAfter &&textAfter.charAt(0);if(ctx.type=="statement"&&firstChar=="}")

```

```

9.186.1.4 function mimes ( ms, mode )

```

```

9.186.1.5 mimes ( {name:"clike", keywords:words(cKeywords), blockKeywords:words("case do else for if switch while
struct"), atoms:words("null"), hooks:{"#":"cppHook"}} )

```

```

9.186.1.6 function tokenAtString ( stream, state )

```

9.187 lib/js/CodeMirror/codemirror.js File Reference

9.188 lib/js/CodeMirror/css.js File Reference

9.189 lib/js/CodeMirror/htmlmixed.js File Reference

Functions

- CodeMirror [defineMode](#) ("htmlmixed", function(config, parserConfig){var scriptStartRegex=parserConfig.scriptStartRegex||/^<%/i, scriptEndRegex=parserConfig.scriptEndRegex||/^%>/i;var scriptingMode, htmlMixedMode;function htmlDispatch(stream, state){if(stream.match(scriptStartRegex, false)){state.token=scriptingDispatch;return scriptingMode.token(stream, state.scriptState);}else return htmlMixedMode.token(stream, state.htmlState);}function scriptingDispatch(stream, state){if(stream.match(scriptEndRegex, false)){state.token=htmlDispatch;return htmlMixedMode.token(stream, state.htmlState);}else return scriptingMode.token(stream, state.scriptState);}return{startState:function(){scriptingMode=scriptingMode||CodeMirror.getMode(config, parserConfig.scriptingModeSpec);htmlMixedMode=htmlMixedMode||CodeMirror.getMode(config,"htmlmixed");return{token:parserConfig.startOpen?scriptingDispatch:htmlDispatch, htmlState:CodeMirror.startState(htmlMixedMode), scriptState:CodeMirror.startState(scriptingMode)};}, token:function(stream, state){return state.token(stream, state);}, indent:function(state, textAfter){if(state.token==htmlDispatch) return htmlMixedMode.indent(state.htmlState, textAfter);else if(scriptingMode.indent) return scriptingMode.indent(state.scriptState, textAfter);}, copyState:function(state){return{token:state.token, htmlState:CodeMirror.copyState(htmlMixedMode, state.htmlState), scriptState:CodeMirror.copyState(scriptingMode, state.scriptState)};}, electricChars:"/{}:", innerMode:function(state){if(state.token==scriptingDispatch) return{state:state.scriptState, mode:scriptingMode};else return{state:state.htmlState, mode:htmlMixedMode};}};,"htmlmixed")
- CodeMirror [defineMIME](#) ("application/x-ejs",{name:"htmlmixed", scriptingModeSpec:"javascript"})

9.189.1 Function Documentation

9.189.1.1 CodeMirror [defineMIME](#) ("application/x-ejs" , {name:"htmlmixed", scriptingModeSpec:"javascript"})

9.189.1.2 CodeMirror [defineMode](#) ("htmlmixed" , function(config, parserConfig){var scriptStartRegex=parserConfig.scriptStartRegex||/^<%/i, scriptEndRegex=parserConfig.scriptEndRegex||/^%>/i;var scriptingMode, htmlMixedMode;function htmlDispatch(stream, state){if(stream.match(scriptStartRegex, false)){state.token=scriptingDispatch;return scriptingMode.token(stream, state.scriptState);}else return htmlMixedMode.token(stream, state.htmlState);}function scriptingDispatch(stream, state){if(stream.match(scriptEndRegex, false)){state.token=htmlDispatch;return htmlMixedMode.token(stream, state.htmlState);}else return scriptingMode.token(stream, state.scriptState);}return{startState:function(){scriptingMode=scriptingMode||CodeMirror.getMode(config, parserConfig.scriptingModeSpec);htmlMixedMode=htmlMixedMode||CodeMirror.getMode(config,"htmlmixed");return{token:parserConfig.startOpen?scriptingDispatch:htmlDispatch, htmlState:CodeMirror.startState(htmlMixedMode), scriptState:CodeMirror.startState(scriptingMode)};}, token:function(stream, state){return state.token(stream, state);}, indent:function(state, textAfter){if(state.token==htmlDispatch) return htmlMixedMode.indent(state.htmlState, textAfter);else if(scriptingMode.indent) return scriptingMode.indent(state.scriptState, textAfter);}, copyState:function(state){return{token:state.token, htmlState:CodeMirror.copyState(htmlMixedMode, state.htmlState), scriptState:CodeMirror.copyState(scriptingMode, state.scriptState)};}, electricChars:"/{}:", innerMode:function(state){if(state.token==scriptingDispatch) return{state:state.scriptState, mode:scriptingMode};else return{state:state.htmlState, mode:htmlMixedMode};}}; , "htmlmixed")

9.190 lib/js/CodeMirror/htmlmixed.js File Reference

9.191 lib/js/CodeMirror/javascript.js File Reference

Functions

- CodeMirror [defineMode](#) ("javascript", function(config, parserConfig){var indentUnit=config.indentUnit;var statementIndent=parserConfig.statementIndent;var jsonMode=parserConfig.json;var isTS=parserConfig.typescript;var keywords=function(){function kw(type){return{type:type, style:"keyword"}};var A=kw("keyword a"), B=kw("keyword b"), C=kw("keyword c");var operator=kw("operator"), atom={type:"atom", style:"atom"};var jsKeywords={"if":kw("if"),"while":A,"with":A,"else":B,"do":B,"try":B,"finally":B,"return":C,"break":


```

:C,"continue":C,"new":C,"delete":C,"throw":C,"var":kw("var"),"const":kw("var"),"let":kw("var"),"function"↵
:kw("function"),"catch":kw("catch"),"for":kw("for"),"switch":kw("switch"),"case":kw("case"),"default":kw("default"),"in"↵
:operator,"typeof":operator,"instanceof":operator,"true":atom,"false":atom,"null":atom,"undefined":atom,"Na↵
N":atom,"Infinity":atom,"this":kw("this"));if(isTS){var type={type:"variable", style:"variable-3"};var ts↵
Keywords={"interface":kw("interface"),"class":kw("class"),"extends":kw("extends"),"constructor":kw("constructor"),"public"↵
:kw("public"),"private":kw("private"),"protected":kw("protected"),"static":kw("static"),"super":kw("super"),"string"↵
:type,"number":type,"bool":type,"any":type};for(var attr in tsKeywords){jsKeywords[attr]=tsKeywords[attr];}return
jsKeywords;}();var isOperatorChar=/[+\-*&%=<>!?|~^]/;function chain(stream, state, f){state.tokenize=f;return
f(stream, state);}function nextUntilUnescaped(stream, end){var escaped=false, next;while((next=stream.↵
next())!=null){if(next==end &&!escaped) return false;escaped=!escaped &&next=="\\";}return escaped;}var
type, content;function ret(tp, style, cont){type=tp;content=cont;return style;}function jsTokenBase(stream,
state){var ch=stream.next();if(ch==" " || ch=="") return chain(stream, state, jsTokenString(ch)); else if (ch ==
"." && stream.match(/^\d+(?:[eE][+-]?(\d+)?/)) return ret("number", "number"); else if (/[\[\]\{\}\(\)\;\,\:\.\]/.test(ch))
return ret(ch); else if (ch == "0" && stream.eat(/[x]/i)) { stream.eatWhile(/[\da-f]/i); return ret("number", "num-
ber"); } else if (/^[\d]/.test(ch)) { stream.match(/^\d*(?:\.\d*)?(?:[eE][+-]?(\d+)?/); return ret("number", "num-
ber"); } else if (ch == "/") { if (stream.eat("*")) { return chain(stream, state, jsTokenComment); } else if
(stream.eat("/")) { stream.skipToEnd(); return ret("comment", "comment"); } else if (state.lastType == "op-
erator" || state.lastType == "keyword c" || /^[\[\]\{\}\(\)\;\,\:\.\$]/.test(state.lastType)) { nextUntilUnescaped(stream,
"/"); stream.eatWhile(/[\gimyl]/); return ret("regexp", "string-2"); } else { stream.eatWhile(isOperatorChar);
return ret("operator", null, stream.current()); } } else if (ch == "#") { stream.skipToEnd(); return ret("error",
"error"); } else if (isOperatorChar.test(ch)) { stream.eatWhile(isOperatorChar); return ret("operator", null,
stream.current()); } else { stream.eatWhile(/[\w\$_]/); var word = stream.current(), known = keywords.↵
propertyIsEnumerable(word) && keywords[word]; return (known && state.lastType != ".") ? ret(known.type,
known.style, word) : ret("variable", "variable", word); } }

function jsTokenString(quote) { return function(stream, state) { if (!nextUntilUnescaped(stream, quote))
state.tokenize = jsTokenBase; return ret("string", "string"); }; }

function jsTokenComment(stream, state) { var maybeEnd = false, ch; while (ch = stream.next()) { if (ch ==
"/" && maybeEnd) { state.tokenize = jsTokenBase; break; } maybeEnd = (ch == "*"); } return ret("comment",
"comment"); }

var atomicTypes = {"atom": true, "number": true, "variable": true, "string": true, "regexp": true, "this": true};

function JSLexical(indented, column, type, align, prev, info) { this.indented = indented; this.column = column;
this.type = type; this.prev = prev; this.info = info; if (align != null) this.align = align; }

function inScope(state, varname) { for (var v = state.localVars; v; v = v.next) if (v.name == varname) return
true; }

function parseJS(state, style, type, content, stream) { var cc = state.cc;

cx.state = state; cx.stream = stream; cx.marked = null, cx.cc = cc;

if (!state.lexical.hasOwnProperty("align")) state.lexical.align = true;

while(true) { var combinator = cc.length ? cc.pop() : jsonMode ? expression : statement; if (combinator(type,
content)) { while(cc.length && cc[cc.length - 1].lex) cc.pop(); if (cx.marked) return cx.marked; if (type ==
"variable" && inScope(state, content)) return "variable-2"; return style; } }

var cx = {state: null, column: null, marked: null, cc: null}; function pass() { for (var i = arguments.length -
1; i >= 0; i--) cx.cc.push(arguments[i]); } function cont() { pass.apply(null, arguments); return true; } function
register(varname) { function inList(list) { for (var v = list; v; v = v.next) if (v.name == varname) return true;
return false; } var state = cx.state; if (state.context) { cx.marked = "def"; if (inList(state.localVars)) return;
state.local↵
Vars = {name: varname, next: state.localVars}; } else { if (inList(state.globalVars)) return; state.globalVars =
{name: varname, next: state.globalVars}; } }

var defaultVars = {name: "this", next: {name: "arguments"}}; function pushcontext() { cx.state.context =
{prev: cx.state.context, vars: cx.state.localVars}; cx.state.localVars = defaultVars; } function popcontext() {
cx.state.localVars = cx.state.context.vars; cx.state.context = cx.state.context.prev; } function pushlex(type,
info) { var result = function() { var state = cx.state, indent = state.indented; if (state.lexical.type == "stat")
indent = state.lexical.indented; state.lexical = new JSLexical(indent, cx.stream.column(), type, null, state.↵
lexical, info); }; result.lex = true; return result; } function poplex() { var state = cx.state; if (state.lexical.prev) { if
(state.lexical.type == "") state.indented = state.lexical.indented; state.lexical = state.lexical.prev; } } poplex.lex
= true;

```



```
function expect(wanted) { return function(type) { if (type == wanted) return cont(); else if (wanted == ";") return pass(); else return cont(arguments.callee); }; }
```

```
function statement(type) { if (type == "var") return cont(pushlex("vardef"), vardef1, expect(";", poplex); if (type == "keyword a") return cont(pushlex("form"), expression, statement, poplex); if (type == "keyword b") return cont(pushlex("form"), statement, poplex); if (type == "{") return cont(pushlex("{}"), block, poplex); if (type == ";") return cont(); if (type == "if") return cont(pushlex("form"), expression, statement, poplex maybeelse); if (type == "function") return cont(functiondef); if (type == "for") return cont(pushlex("form"), expect("("), pushlex("{}"), forspec1, expect(")"), poplex, statement, poplex); if (type == "variable") return cont(pushlex("stat"), maybelabel); if (type == "switch") return cont(pushlex("form"), expression, pushlex("{}"), "switch", expect("{}"), block, poplex, poplex); if (type == "case") return cont(expression, expect(":")); if (type == "default") return cont(expect(":")); if (type == "catch") return cont(pushlex("form"), pushcontext, expect("("), funarg, expect(")"), statement, poplex, popcontext); return pass(pushlex("stat"), expression, expect(";", poplex); } function expression(type) { return expressionInner(type, false); } function expressionNoComma(type) { return expressionInner(type, true); } function expressionInner(type, noComma) { var maybeop = noComma ? maybeoperatorNoComma : maybeoperatorComma; if (atomicTypes.hasOwnProperty(type)) return cont(maybeop); if (type == "function") return cont(functiondef); if (type == "keyword c") return cont(noComma ? maybeexpressionNoComma : maybeexpression); if (type == "(") return cont(pushlex("("), maybeexpression, expect(")"), poplex, maybeop); if (type == "operator") return cont(noComma ? expressionNoComma : expression); if (type == "[") return cont(pushlex("["), commasep(expressionNoComma, "]", poplex, maybeop); if (type == "{") return cont(pushlex("{}"), commasep(objprop, "}", poplex, maybeop); return cont(); } function maybeexpression(type) { if (type.match(/[{}()\[\]`]/)) return pass(); return pass(expression); } function maybeexpressionNoComma(type) { if (type.match(/[{}()\[\]`]/)) return pass()
```

- `return pass` (`expressionNoComma`)
- function `maybeoperatorComma` (`type`, `value`)
- function `maybeoperatorNoComma` (`type`, `value`, `noComma`)
- function `maybelabel` (`type`)
- function `property` (`type`)
- function `objprop` (`type`, `value`)
- function `getterSetter` (`type`)
- function `commasep` (`what`, `end`)
- function `block` (`type`)
- function `maybetype` (`type`)
- function `typedef` (`type`)
- function `vardef1` (`type`, `value`)
- function `vardef2` (`type`, `value`)
- function `maybeelse` (`type`, `value`)
- function `forspec1` (`type`)
- function `formaybein` (`_type`, `value`)
- function `forspec2` (`type`, `value`)
- function `forspec3` (`type`)
- function `functiondef` (`type`, `value`)
- function `funarg` (`type`, `value`)
- CodeMirror `defineMIME` ("`text/javascript`", "`javascript`")

Variables

- return

9.191.1 Function Documentation

9.191.1.1 function block (*type*)

9.191.1.2 function commasep (*what*, *end*)

9.191.1.3 CodeMirror defineMIME ("text/javascript" , "javascript")

Generated on Thu Nov 12 2015 12:33:31 for girlCMS by Doxygen = "sv

- 9.191.1.5 function formaybein (*_type*, *value*)
- 9.191.1.6 function forspec1 (*type*)
- 9.191.1.7 function forspec2 (*type*, *value*)
- 9.191.1.8 function forspec3 (*type*)
- 9.191.1.9 function funarg (*type*, *value*)
- 9.191.1.10 function functiondef (*type*, *value*)
- 9.191.1.11 function getterSetter (*type*)
- 9.191.1.12 function maybeelse (*type*, *value*)
- 9.191.1.13 function maybelabel (*type*)
- 9.191.1.14 function maybeoperatorComma (*type*, *value*)
- 9.191.1.15 function maybeoperatorNoComma (*type*, *value*, *noComma*)
- 9.191.1.16 function maybetype (*type*)
- 9.191.1.17 function objprop (*type*, *value*)
- 9.191.1.18 return pass (*expressionNoComma*)
- 9.191.1.19 function property (*type*)
- 9.191.1.20 function typedef (*type*)
- 9.191.1.21 function vardef1 (*type*, *value*)
- 9.191.1.22 function vardef2 (*type*, *value*)

9.191.2 Variable Documentation

- 9.191.2.1 return

9.192 lib/js/CodeMirror/matchbrackets.js File Reference

Functions

- function [findMatchingBracket](#) (cm, where, strict)
- function [matchBrackets](#) (cm, autoclear)
- function [doMatchBrackets](#) (cm)
- CodeMirror [defineOption](#) ("matchBrackets", false, function(cm, val, old){if(old &&old!=CodeMirror.Init) cm.off("cursorActivity", doMatchBrackets);if(val){cm.state.matchBrackets=typeof val=="object"?val:{};cm.↵ on("cursorActivity", doMatchBrackets);}})
- CodeMirror [defineExtension](#) ("matchBrackets", function(){[matchBrackets](#)(this, true);})

Variables

- var [Pos](#) = CodeMirror.Pos

- var `matching` = {"(": ">", ")": "<", "[": ">", "]": "<", "{": ">", "}": "<"}
- var `currentlyHighlighted` = null

9.192.1 Function Documentation

9.192.1.1 `CodeMirror.defineExtension ("matchBrackets" , function(){matchBrackets(this, true);})`

9.192.1.2 `CodeMirror.defineOption ("matchBrackets" , false , function(cm, val, old){if(old && old!=CodeMirror.Init) cm.off("cursorActivity", doMatchBrackets);if(val){cm.state.matchBrackets=typeof val=="object"?val:{};cm.on("cursorActivity", doMatchBrackets);}}`)

9.192.1.3 `function doMatchBrackets (cm)`

9.192.1.4 `function findMatchingBracket (cm, where, strict)`

9.192.1.5 `function matchBrackets (cm, autoclear)`

9.192.2 Variable Documentation

9.192.2.1 `var currentlyHighlighted = null`

9.192.2.2 `var matching = {"(": ">", ")": "<", "[": ">", "]": "<", "{": ">", "}": "<"}`

9.192.2.3 `var Pos = CodeMirror.Pos`

9.193 lib/js/CodeMirror/php.js File Reference

9.194 lib/js/CodeMirror/xml.js File Reference

9.195 lib/js/gino-min.js File Reference

Functions

- `document.createElement` ('header')
- `function window addEvent` ('load', function(){\$\$('button[data-toggle=collapse]).each(function(btn){var target=\$(btn.get('data-target'));btn.addEvent('click', function(){target.toggle();});});} gino.html_unescape

Variables

- var `gino` = {}
- `function gino sameDomain`
- `gino loading` = "<img src='img/ajax-loader.gif' alt='loading...'"
- `gino loading_element` = new Element('img[src=img/ajax-loader.gif']).setProperty('alt', 'loading...')
- `gino requestCache` = new `Array`()
- `gino ajaxRequest`
- `gino printCalendar`
- `gino hScrollingList`
- `gino vScrollingList`
- `gino filterMulticheck`

9.195.1 Function Documentation

9.195.1.1 `function window addEvent ('load' , function(){$$('button[data-toggle=collapse]').each(function(btn){var target=$(btn.get('data-target'));btn.addEvent('click', function(){target.toggle();});}) }`

9.195.1.2 `document createElement ('header')`

9.195.2 Variable Documentation

9.195.2.1 `gino ajaxRequest`

9.195.2.2 `gino filterMulticheck`

9.195.2.3 `var gino = {}`

9.195.2.4 `gino hScrollingList`

9.195.2.5 `gino loading = ""`

9.195.2.6 `gino loading_element = new Element('img[src=img/ajax-loader.gif]').setProperty('alt', 'loading...')`

9.195.2.7 `gino printCalendar`

9.195.2.8 `gino requestCache = new Array()`

9.195.2.9 `function gino sameDomain`

Initial value:

```
= function(win) {
  var H=location.href,
  local= H.substring(0, H.indexOf(location.pathname));
  try {
    win=win.document;
    return win && win.URL && win.URL.indexOf(local)== 0;
  }
  catch(er) {
    return false;
  }
}

document.createElement('hgroup')
```

9.195.2.10 `gino vScrollingList`

9.196 lib/js/modernizr.js File Reference

Variables

- `window Modernizr` =function(a,b,c){function D(a){j.cssText=a}function E(a,b){return D(n.join(a+";")+ (b||""))}function F(a,b){return typeof a===b}function G(a,b){return!!~(""+a).indexOf(b)}function H(a,b){for(var d in a){var e=a[d];if(!G(e,"-")&&j[e]!==c)return b=="pfx"?e:!0}return!1}function I(a,b,d){for(var e in a){var f=b[a[e]];if(f!==c)return d===!1?a[e]:F(f,"function")?f.bind(d||b):f}return!1}function J(a,b,c){var d=a.charAt(0).toUpperCase()+a.slice(1),e=(a+" "+p.join(d+" "+d)).split(" ");return F(b,"string")||F(b,"undefined")?H(e,b):(e=(a+" "+q.join(d+" "+d)).split(" "),I(e,b,c))}function K(){e.input=function(c){for(var d=0,e=c.length;d<e;d++)u[c[d]]=c[d]in k;return u.list&&(u.list!==b.createElement("datalist")&&!a.HTMLDataListElement),u}("autocomplete autofocus list placeholder max min multiple pattern required step".split(" ")),e.inputtypes=function(a){for(var d=0,e,f,h,i=a.length;d<i;d++)k.setAttribute("type",f=a[d]),e=k.type!="text",e&&(k.value=l,k.style.cssText="position: absolute; visibility: hidden;","/^range\$/".test(f)&&k.style.WebkitAppearance!==c?(g.appendChild(k),h=b.↵

```

defaultView,e=h.getComputedStyle&&h.getComputedStyle(k,null).WebkitAppearance!=="textfield"&&k.↵
offsetHeight!=="0,g.removeChild(k):/^ (search|tel)$/.test(f)||/^ (url|email)$/.test(f)?e=k.checkValidity&&k.↵
checkValidity()===!1:e=k.value!=!1),t[a[d]]=!e;return t}("search tel url email datetime date month week time
datetime-local number range color".split(" "))var d="2.7.1",e={},f=!0,g=b.documentElement,h="modernizr",i=b.↵
createElement(h),j=i.style,k=b.createElement("input"),l="",m={}.toString,n=" -webkit- -moz- -o- -ms-
".split(" "),o="Webkit Moz O ms",p=o.split(" "),q=o.toLowerCase().split(" "),r={svg:"http://www.w3.↵
org/2000/svg"},s={},t={},u={},v=[],w=v.slice,x,y=function(a,c,d,e){var f,i,j,k,l=b.createElement("div"),m=b.↵
body,n=m||b.createElement("body");if(parseInt(d,10))while(d--){j=b.createElement("div"),j.id=e[e[d]:h+(d+1)],l.↵
appendChild(j);return f=["&#173;","<style id='s',h,'">','a,'"</style>"].join(""),l.id=h,(m?!:n).innerHTM↵
L+=f,n.appendChild(l),m||(n.style.background="",n.style.overflow="hidden",k=g.style.overflow,g.style.↵
overflow="hidden",g.appendChild(n)),i=c(l,a),m?!:parentNode.removeChild(l):(n.parentNode.remove↵
Child(n),g.style.overflow=k,!i),z=function(b){var c=a.matchMedia||a.msMatchMedia;if(c)return c(b).matches;var
d;return y("@media "+b+" { #'+h+' { position: absolute; } }",function(b){d=(a.getComputedStyle?get↵
ComputedStyle(b,null):b.currentStyle)["position"]=="absolute"},d),A=function(){function d(d,e){e=e||b.↵
createElement(a[d]||"div"),d="on"+d;var f=d in e;return f||(e.setAttribute||(e=b.createElement("div")),e.↵
setAttribute&&e.removeAttribute&&(e.setAttribute(d,""),f=F(e[d],"function"),F(e[d],"undefined"))||(e[d]=c),e.↵
removeAttribute(d))),e=null,f;var a={select:"input",change:"input",submit:"form",reset:"form",error:"img",load↵
:"img",abort:"img"};return d}(),B={}.hasOwnProperty,C;!F(B,"undefined")&&F(B.call,"undefined")?C=function(a,b){return
B.call(a,b):C=function(a,b){return b in a&&F(a.constructor.prototype[b],"undefined")},Function.prototype.↵
bind||(Function.prototype.bind=function(b){var c=this;if(typeof c!="function")throw new TypeError;var
d=w.call(arguments,1),e=function(){if(this instanceof e){var a=function(){};a.prototype=c.prototype;var
f=new a,g=c.apply(f,d.concat(w.call(arguments)));return Object(g)===g?g:f}return c.apply(b,d.concat(w.↵
call(arguments))));return e}),s.flexbox=function(){return J("flexWrap")},s.flexboxlegacy=function(){return
J("boxDirection")},s.canvas=function(){var a=b.createElement("canvas");return!!a.getContext&&!!a.get↵
Context("2d")},s.canvastext=function(){return!!e.canvas&&!!F(b.createElement("canvas").getContext("2d").↵
fillText,"function")},s.webgl=function(){return!!a.WebGLRenderingContext},s.touch=function(){var c;return"ontouchstart"in
a||a.DocumentTouch&&b instanceof DocumentTouch?c=!0:y(["@media (" ,n.join("touch-enabled"),("," ,h,""),"," ,{"#modernizr{top↵
:9px;position:absolute}}"].join(""),function(a){c=a.offsetTop===9}),c},s.geolocation=function(){return"geolocation"in
navigator},s.postMessage=function(){return!!a.postMessage},s.websqlDatabase=function(){return!!a.open↵
Database},s.indexedDB=function(){return!!J("indexedDB",a)},s.hashchange=function(){return A("hashchange",a)&&(b.↵
documentMode===c||b.documentMode>7)},s.history=function(){return!!a.history&&!!history.pushState},s.↵
draganddrop=function(){var a=b.createElement("div");return"draggable"in a||"ondragstart"in a&&"ondrop"in
a},s.websockets=function(){return"WebSocket"in a||"MozWebSocket"in a},s.rgb=function(){return D("background-
color:rgb(150,255,150,5)"),G(j.backgroundColor,"rgb")},s.hsl=function(){return D("background-color↵
:hsl(120,40%,100%,.5)"),G(j.backgroundColor,"rgb")||G(j.backgroundColor,"hsl")},s.multiplebgs=function(){return
D("background:url(https://),url(https://),red url(https://)"),/(url\s*(.?.*)?){3}/.test(j.backgroundColor)},s.backgroundsize=function(){return
J("backgroundSize")},s.borderimage=function(){return J("borderImage")},s.borderradius=function(){return
J("borderRadius")},s.boxshadow=function(){return J("boxShadow")},s.textshadow=function(){return b.↵
createElement("div").style.textShadow===""},s.opacity=function(){return E("opacity:.55"),/^0.55$/test(j.↵
opacity)},s.cssanimations=function(){return J("animationName")},s.csscolumns=function(){return J("column↵
Count")},s.cssgradients=function(){var a="background-image:" ,b="gradient(linear,left top,right bottom,from(#9f9),to(white));",c="
gradient(left top,#9f9, white);";return D((a+"-webkit- ".split(" ").join(b+a)+n.join(c+a)).slice(0,-a.length)),G(j.↵
backgroundImage,"gradient")},s.cssreflections=function(){return J("boxReflect")},s.csstransforms=function(){return!!↵
J("transform")},s.csstransforms3d=function(){var a=!J("perspective");return a&&"webkitPerspective"in g.↵
style&&y("@media (transform-3d),(-webkit-transform-3d){#modernizr{left:9px;position:absolute;height↵
:3px;}}",function(b,c){a=b.offsetLeft===9&&b.offsetHeight===3}),a},s.csstransitions=function(){return J("transition")},s.↵
fontface=function(){var a;return y("@font-face {font-family:'font';src:url('https://')}",function(c,d){var e=b.↵
getElementById("smodernizr"),f=e.sheet||e.styleSheet,g=f?f.cssRules&&f.cssRules[0]?f.cssRules[0].css↵
Text:f.cssText||"":"";a=/src/i.test(g)&&g.indexOf(d.split(" ")[0])===0}),a},s.generatedcontent=function(){var
a;return y(["#",h,"{font:0/0 a}#",h,":after{content:'"+l+"';visibility:hidden;font:3px/1 a}"].join(""),function(b){a=b.↵
offsetHeight>=3}),a},s.video=function(){var a=b.createElement("video"),c=!1;try{if(c=!a.canPlayType)c=new
Boolean(c),c.ogg=a.canPlayType('video/ogg; codecs="theora"').replace(/^no$/,""),c.h264=a.canPlay↵
Type('video/mp4; codecs="avc1.42E01E"').replace(/^no$/,""),c.webm=a.canPlayType('video/webm; codecs="vp8,
vorbis"').replace(/^no$/,"")}catch(d){return c},s.audio=function(){var a=b.createElement("audio"),c=!1;try{if(c=!a.↵
canPlayType)c=new Boolean(c),c.ogg=a.canPlayType('audio/ogg; codecs="vorbis"').replace(/^no$/,""),c.↵
mp3=a.canPlayType('audio/mpeg;').replace(/^no$/,""),c.wav=a.canPlayType('audio/wav; codecs="1"').↵
replace(/^no$/,""),c.m4a=(a.canPlayType('audio/x-m4a;')||a.canPlayType('audio/aac;')).replace(/^no$/,"")}catch(d){return
c},s.localstorage=function(){try{return localStorage.setItem(h,h),localStorage.removeItem(h,!0)}catch(a){return!1}},s.↵

```



```

sessionStorage=function(){try{return sessionStorage.setItem(h,h),sessionStorage.removeItem(h,!0)}catch(a){return!1}};s.↵
webworkers=function(){return!!a.Worker},s.applicationCache=function(){return!!a.applicationCache},s.↵
svg=function(){return!!b.createElementNS&&!!b.createElementNS(r.svg,"svg").createElementNS(r.svg,"svgRect"),s.inlinesvg=function(){var
a=b.createElement("div");return a.innerHTML="<svg/>",(a.firstChild&&a.firstChild.namespaceURI)==r.↵
svg},s.smil=function(){return!!b.createElementNS&&/SVGAnimate/.test(m.call(b.createElementNS(r.↵
svg,"animate"))),s.svgclippaths=function(){return!!b.createElementNS&&/SVGClipPath/.test(m.call(b.↵
createElementNS(r.svg,"clipPath")));for(var L in s)C(s,L)&&(x=L.toLowerCase(),e[x]=s[L](),v.push((e[x]?""↵
:"no-")+x));return e.input||K(),e.addTest=function(a,b){if(typeof a=="object")for(var d in a)C(a,d)&&e.add↵
Test(d,a[d]);else{a=a.toLowerCase();if(e[a]!=c)return e;b=typeof b=="function"?b():b,typeof f!="undefined"&&f&&(g.↵
className+=" "+(b?"":"no-")+a),e[a]=b)return e},D(""),i=k=null,function(a,b){function l(a,b){var c=a.↵
createElement("p"),d=a.getElementsByName("head")[0]||a.documentElement;return c.innerHTML=↵
L="x<style>"+b+"</style>",d.insertBefore(c.lastChild,d.firstChild)}function m(){var a=s.elements;return
typeof a=="string"?a.split(" "):a}function n(a){var b=j[a[h]];return b||(b={},i++,a[h]=i,j[i]=b),b}function
o(a,c,d){c||(c=b);if(k)return c.createElement(a);d||(d=n(c));var g;return d.cache[a]?g=d.cache[a].clone↵
Node():f.test(a)?g=(d.cache[a]=d.createElement(a)).cloneNode():g=d.createElement(a),g.canHaveChildren&&le.↵
test(a)&&!g.tagName?d.frag.appendChild(g):g}function p(a,c){a||(a=b);if(k)return a.createDocument↵
Fragment();c=c||n(a);var d=c.frag.cloneNode(),e=0,f=m(),g=f.length;for(;e<g;e++)d.createElement(f[e]);return
d}function q(a,b){b.cache||(b.cache={},b.createElement=a.createElement,b.createFrag=a.createDocument↵
Fragment,b.frag=b.createFrag()),a.createElement=function(c){return s.shivMethods?o(c,a,b):b.create↵
Elem(c)},a.createDocumentFragment=function(h,f){return function(){var n=f.cloneNode(),c=n.create↵
Element;h.shivMethods&&(" "+m()).join().replace(/[\w-]+/g,function(a){return b.createElem(a),b.frag.create↵
Element(a),"c"+" "+a+"")});return n})(s,b.frag)}function r(a){a||(a=b);var c=n(a);return s.shivCSS&&lg&&lc.↵
hasCSS&&(c.hasCSS=!!(a,"article,aside,dialog,figcaption,figure,footer,header,hgroup,main,nav,section{display↵
:block}mark{background:#FF0;color:#000}template{display:none}")),k||q(a,c),a}var c="3.7.0",d=a.html5||[],e=/^<|<(?↵
:button|map|select|textarea|object|iframe|option|optgroup)$/i,f=/^(?<a|b|code|div|fieldset|h1|h2|h3|h4|h5|h6|i|label|li|ol|p|q|spa
_html5shiv",i=0,j={},k;(function(){try{var a=b.createElement("a");a.innerHTML="<xyz></xyz>",g="hidden"in
a,k=a.childNodes.length==1||function(){b.createElement("a");var a=b.createDocumentFragment();return
typeof a.cloneNode=="undefined"||typeof a.createDocumentFragment=="undefined"||typeof a.create↵
Element=="undefined"}})catch(c){g=!0,k=!0}});var s={elements:d.elements||"abbr article aside audio bdi
canvas data datalist details dialog figcaption figure footer header hgroup main mark meter nav output progress
section summary template time video",version:c,shivCSS:d.shivCSS!==!1,supportsUnknownElements:↵
:k,shivMethods:d.shivMethods!==!1,type:"default",shivDocument:r,createElement:o,createDocument↵
Fragment:p},a.html5=s,r(b))(this,b),e._version=d,e._prefixes=n,e._domPrefixes=q,e._cssomPrefixes=p,e.↵
mq=z,e.hasEvent=A,e.testProp=function(a){return H([a])},e.testAllProps=J,e.testStyles=y,g.className=g.↵
className.replace(/(^\s|no-js(\s|$))/,"$1$2")+("f?" js "+v.join(" ").":""),e)(this,this.document)

```

- function **a** =navigator.userAgent.toLowerCase()
- function **b** {var **a**=}
- function **c** {function **d(a)**{return"object Function"==o.call(a)}function **e(a)**{return"string"==typeof a}function **f()**{function **g(a)**{return!a||"loaded"==a||"complete"==a||"uninitialized"==a}function **h()**{var **a**=p.shift(),**q**=1,**a?****a?****t?**m(function(){{"c"==a.t?**B.injectCss:B.injectJs**}(a.s,0,a.a,a.x,a.e,1),0):(a(),h()):q=0}function **i(a,c,d,e,f,i,j)**{function **k(b)**{if(!o&&g(l.readyState)&&(u.r=o=1,lq&&h()),l.onload=l.onreadystatechange=null,b)}{"img"!a&&m(function(){t.↵
removeChild(l),50);for(var d in y[c])y[c].hasOwnProperty(d)&&y[c][d].onload())var j=j||**B.errorTimeout**,l=b.↵
createElement(a),o=0,r=0,u={t:d,s:c,e:f,a:i,x:j};1===y[c]&&(r=1,y[c]=[]),"object"==a?l.data=c:(l.src=c,l.↵
type=a),l.width=l.height="0",l.onerror=l.onload=l.onreadystatechange=function(){k.call(this,r)},p.splice(e,0,u),"img"!a&&(r||2==↵
insertBefore(l,s?null:n),m(k,j)):y[c].push(l)}function j(a,b,c,d,f){return q=0,b=b||"j",e(a)?i("c"==b?v:u,a,b,this.↵
i++,c,d,f):(p.splice(this.i++,0,a),1==p.length&&h()),this}function k(){var **a**=**B**;return a.loader=[load;j,i:0],a}var
l=b.documentElement,m=a.setTimeout,n=b.getElementsByTagName("script")[0],o={}.toString,p=[],q=0,r="Moz↵
Appearance"in l.style,s=r&&!!b.createRange().compareNode,t=s?!n.parentNode,l=a.opera&&"[object
Opera]"==o.call(a.opera),l=!!b.attachEvent&&!!u,r?"object":l?"script":l?"img",v=l?"script":u,w=Array.is↵
Array||function(a){return"object Array"==o.call(a)},x=[],y={},z={timeout:function(a,b){return b.length&&(a.↵
timeout=b[0]),a}},A,B
- B** =function(a){function **b(a)**{var **a**=a.split("!"),b=x.length,c=a.pop(),d=a.length,c={url:c,origUrl:c,prefixes↵
:a},e,f,g;for(f=0;f<d;f++)g=a[f].split("="),(e=z[g.shift()])&&(c=e(c,g));for(f=0;f<b;f++)c=x[f](c);return c}function
g(a,e,f,g,h){var i=b(a),j=i.autoCallback,i.url.split(".").pop().split("?").shift(),i.bypass||(e&&(e=d(e))?e:e[a]||e[g]||e[a.↵
split("").pop().split("?")[0]],i.instead?i.instead(a,e,f,g,h):(y[i.url]?i.noexec=!0:y[i.url]=1,f.load(i.url,i.forceC↵
SS||li.forceJS&&"css"==i.url.split(".").pop().split("?").shift()?"c":i.noexec,i.attrs,i.timeout),(d(e)||d(j))&&f.↵
load(function(){k(),e&&e(i.origUrl,h,g),j&&j(i.origUrl,h,g),y[i.url]=2})))function **h(a,b)**{function **c(a,c){if(a){if(e(a))c||(j=function(){var
a=[].slice.call(arguments);k.apply(this,a,l()),g(a,j,b,0,h);else if(Object(a)===a)for(n in m=function(){var**


```

b=0,c;for(c in a)a.hasOwnProperty(c)&&b++;return b}(),a)a.hasOwnProperty(n)&&(!c&&!--m&&(d(j)?j=function(){var
a=[].slice.call(arguments);k.apply(this,a),l();j[n]=function(a){return function(){var b=[].slice.call(arguments);a&&a.↵
apply(this,b,l())}(k[n])),g(a[n],j,b,n,h))}else!c&&l())var h=!a.test,i=a.load||a.both,j=a.callback||f,k=j,l=a.↵
complete||f,m,n;c(h?a.yep:a.nope,!i),i&&c(i))var i,j,l=this.yepnope.loader;if(e(a))g(a,0,l,0);else if(w(a))for(i=0;i<a.↵
length;i++)j=a[i],e(j)?g(j,0,l,0):w(j)?B(j):Object(j)===j&&h(j,l);else Object(a)===a&&h(a,l)}

```

- **B addPrefix** =function(a,b){z[a]=b}
- **B B addFilter** =function(a){x.push(a)}
- **B B B errorTimeout** =1e4
- **B B B null** ==b.readyState&&b.addEventListener&&(b.readyState="loading",b.addEventListener("DOM↵
ContentLoaded",A=function(){b.removeEventListener("DOMContentLoaded",A,0),b.readyState="complete"},0))
- **B B B a yepnope** =k()
- **B B B a a yepnope executeStack** =h
- **B B B a a yepnope a yepnope injectJs** =function(a,c,d,e,i,j){var k=b.createElement("script"),l,o,e=e||B.error↵
Timeout;k.src=a;for(o in d)k.setAttribute(o,d[o]);c=j?h:c||f,k.onreadystatechange=k.onload=function(){!l&&g(k.↵
readyState)&&(l=1,c),k.onload=k.onreadystatechange=null}},m(function(){l||l=1,c(1)}),e,i?k.onload():n.↵
parentNode.insertBefore(k,n)}
- **B B B a a yepnope a yepnope a yepnope injectCss** =function(a,c,d,e,g,i){var e=b.createElement("link"),j,c=i?h↵
:c||f,e.href=a,e.rel="stylesheet",e.type="text/css";for(j in d)e.setAttribute(j,d[j]);g||(n.parentNode.insert↵
Before(e,n),m(c,0))}(this,document),Modernizr.load=function(){yepnope.apply(window,[].slice.call(arguments,0))}

9.196.1 Variable Documentation

9.196.1.1 **var a** =navigator.userAgent.toLowerCase()

9.196.1.2 **B B addFilter** =function(a){x.push(a)}

9.196.1.3 **B addPrefix** =function(a,b){z[a]=b}

9.196.1.4 **B** =function(a){function b(a){var a=a.split("!"),b=x.length,c=a.pop(),d=a.length,c={url:c,origUrl:c,prefixes↵
:a},e,f,g;for(f=0;f<d;f++)g=a[f].split("="),(e=z[g.shift()])&&(c=e(c,g));for(f=0;f<b;f++)c=x[f](c);return
c}function g(a,e,f,g,h){var i=b(a),j=i.autoCallback;i.url.split(".").pop().split("?").shift(),i.bypass||(e&&(e=d(e)?e↵
:e[a])|e[g])|e[a.split("/").pop().split("?")[0]],i.instead?i.instead(a,e,f,g,h):(y[i.url]?i.noexec=!0:y[i.url]=1,f.↵
load(i.url,i.forceCSS||i.forceJS&&"css"==i.url.split(".").pop().split("?").shift()?"c":c,i.noexec,i.attrs,i.↵
timeout),(d(e)||d(j))&&f.load(function(){k(i,e&&e(i.origUrl,h,g),j&&j(i.origUrl,h,g),y[i.url]?2)}))}function
h(a,b){function c(a,c){if(a){if(e(a))c||(j=function(){var a=[].slice.call(arguments);k.↵
apply(this,a,l())},g(a,j,b,0,h);else if(Object(a)===a)for(n in m=function(){var b=0,c;for(c in
a)a.hasOwnProperty(c)&&b++;return b}(),a)a.hasOwnProperty(n)&&(!c&&!--m&&(d(j)?j=function(){var
a=[].slice.call(arguments);k.apply(this,a),l();j[n]=function(a){return function(){var b=[].slice.call(arguments);a&&a.↵
apply(this,b,l())}(k[n])),g(a[n],j,b,n,h))}else!c&&l())var
h=!a.test,i=a.load||a.both,j=a.callback||f,k=j,l=a.complete||f,m,n;c(h?a.yep:a.nope,!i),i&&c(i))var
i,j,l=this.yepnope.loader;if(e(a))g(a,0,l,0);else if(w(a))for(i=0;i<a.length;i++)j=a[i],e(j)?g(j,0,l,0):w(j)?B(j)↵
:Object(j)===j&&h(j,l);else Object(a)===a&&h(a,l)}

9.196.1.5 **var b** {var a={}

```

9.196.1.6 var c {function d(a){return"[object Function]"==o.call(a)}function e(a){return"string"==typeof a}function
f(){function g(a){return!a||"loaded"==a||"complete"==a||"uninitialized"==a}function h(){var a=p.↵
shift();q=1,a?a.t?m(function(){{"c"==a.t?B.injectCss:B.injectJs}(a.s,0,a.a,a.x,a.e,1),0):(a(),h()):q=0}function
i(a,c,d,e,f,i,j){function k(b){if(!o&&g(l.readyState)&&(u.r=o=1,!q&&h()),l.onload=l.↵
onreadystatechange=null,b)){!img!=a&&m(function(){t.removeChild(l),50);for(var d in
y[c])y[c].hasOwnProperty(d)&&y[c][d].onload()}}var j=j||B.errorTimeout,l=b.create↵
Element(a,o=0,r=0,u={t:d,s:c,e:f,a:i,x:j};1===y[c]&&(r=1,y[c]=[ ]),"object"==a?l.data=c:(l.src=c,l.↵
type=a),l.width=l.height="0",l.onerror=l.onload=l.onreadystatechange=function(){k.call(this,r)},p.↵
splice(e,0,u),"img"!=a&&(r||2===y[c]?t.insertBefore(l,s?null:n),m(k,j)):y[c].push(l))}function
j(a,b,c,d,f){return q=0,b=b||"j",e(a)?i("c"==b?v:u,a,b,this.i++,c,d,f):(p.splice(this.↵
i++,0,a),1==p.length&&h()),this}function k(){var a=B;return a.loader={load:j,i:0},a}var
l=b.documentElement,m=a.setTimeout,n=b.getElementsByTagName("script")[0],o={}.toString,p=[ ],q=0,r="Moz↵
Appearance"in l.style,s=r&&!b.createRange().compareNode,t=s?!n.parentNode,l=a.opera&&"[object
Opera]"==o.call(a.opera),l=!!b.attachEvent&&!u=r?"object":l?"script":"img",v=l?"script":u,w=Array.↵
isArray||function(a){return"[object Array]"==o.call(a)},x=[ ],y={},z={timeout:function(a,b){return
b.length&&(a.timeout=b[0]),a}},A,B

```

9.196.1.7 B B B errorTimeout =1e4

9.196.1.8 B B B a a yepnope executeStack =h

```

9.196.1.9 B B B a a yepnope a yepnope a yepnope injectCss =function(a,c,d,e,g,i){var
e=b.createElement("link"),j,c=i?h:c||f;e.href=a,e.rel="stylesheet",e.type="text/css";for(j in
d)e.setAttribute(j,d[j]);g||(n.parentNode.insertBefore(e,n),m(c,0))}(this,document),Modernizr.↵
load=function(){yepnope.apply(window,[ ].slice.call(arguments,0))}

```

```

9.196.1.10 B B B a a yepnope a yepnope injectJs =function(a,c,d,e,i,j){var k=b.create↵
Element("script"),l,o,e=||B.errorTimeout;k.src=a;for(o in d)k.setAttribute(o,d[o]);c=j?h↵
:c||f,k.onreadystatechange=k.onload=function(){!!&&g(k.readyState)&&(l=1,c(),k.onload=k.↵
onreadystatechange=null)},m(function(){l||(l=1,c(1)),e,i?k.onload():n.parentNode.insertBefore(k,n)}

```

Generated on Thu Nov 12 2015 12:33:34 for gino-GMS by Doxygen

9.196.1.12 **B B B** null ==b.readyState&&b.addEventListener&&(b.readyState="loading",b.addEventListener("DOMContentLoaded",A=function(){b.removeEventListener("DOMContentLoaded",A,0),b.readyState="complete"},0))

9.196.1.13 **B B B** a.yepnope=k()

9.197 lib/js/MooComplete.js File Reference

Functions

- function [MooComplete](#) (element, options)

9.197.1 Function Documentation

9.197.1.1 function [MooComplete](#) (*element*, *options*)

9.198 lib/js/mootools-1.4.0-yc.js File Reference

Functions

- [for](#) (var [q](#) in{toString:1})
- [if](#) ([r](#))
- [f](#) [implement](#) ({hide:function(){this.\$hidden=true;return this;}, protect:function(){this.\$protected=true;return this;}})
- new [k](#) ("Type", [k](#))
- [c](#) ("String", String,[["charAt"](#),["charCodeAt"](#),["concat"](#),["indexOf"](#),["lastIndexOf"](#),["match"](#),["quote"](#),["replace"](#),["search"](#),["slice"](#),["split"](#),["substring"](#),["toLowerCase"](#),["toUpperCase"](#)])(["Array"](#))
- [Date](#) [extend](#) ("now", function(){return+(new [Date](#));})
- [Array](#) [implement](#) ("clone", function(){var v=this.length, w=new [Array](#)(v);while(v--){w[v]=s(this[v]);}return w;})
- [Object](#) [extend](#) ({merge:function(C, y, x){if(e(y)=="string"){return a(C, y, x);}for(var B=1, w=arguments.length;B< w;B++){var z=arguments[B];for(var A in z){a(C, A, z[A]);}}return C;}, clone:function(i){var w={};for(var v in i){w[v]=s(i[v]);}return w;}, append:function(z){for(var y=1, w=arguments.length;y< w;y++){var v=arguments[y]||{};for(var x in v){z[x]=v[x];}}return z;})
- [each](#) (function(i){new [k](#)(i);})
- [g](#) [alias](#) ("each","forEach")
- [if](#) (Object.create==Function.prototype.create)
- Hash [alias](#) ({indexOf:"keyOf", contains:"hasValue"})
- [\\$uid](#) ([j](#))
- [Window](#) [mirror](#) (function([e](#), [q](#)){j[e]=q;})
- [if](#) (!l.head)
- [if](#) (l.execCommand)
- [if](#) (this.attachEvent &&!this.addEventListener)
- [catch](#) ([h](#))
- [if](#) (p.Platform.ios)
- [if](#) (p.ie)
- [if](#) (p.safari||p.chrome)
- [a](#) [defineKeys](#) ({["38"](#):["up"](#),["40"](#):["down"](#),["37"](#):["left"](#),["39"](#):["right"](#),["27"](#):["esc"](#),["32"](#):["space"](#),["8"](#):["backspace"](#),["9"](#):["tab"](#),["46"](#):["delete"](#),["13"](#):["enter"](#)})
- [a](#) [implement](#) ("implement", f.overloadSetter())
- function [b](#) ([x](#), [s](#), [D](#), [z](#), [r](#), [C](#), [q](#), [B](#), [A](#), [y](#), [u](#), [F](#), [G](#), [v](#), [p](#), [w](#))
- [if](#) (!this.Slick)
- [apply](#) ((typeof exports!="undefined"?exports:this)
- [k](#) [addListener](#) ("DOMContentLoaded", [g](#))
- [if](#) (d.doScroll &&!h())

- `if (k.readyState)`
- `if ("onreadystatechange" in k)`
- `i addEvent ("load", function(){f=true;})`
- `Events implement (Events.Pseudos(b, a.addEvent, a.removeEvent))`
- `if (this.Fx)`
- `if (this.Element &&Element.implement &&this.Fx)`
- `Object append (f, new Events)`
- `Object append (c, f,{setLanguage:f.use, getCurrentLanguage:function(){var g=f.getCurrent();return(g)?g.↵ name:null;}, set:function(){f.define.apply(this, arguments);return this;}, get:function(i, h, g){if(h){i+= "."+h;}return f.get(i, g);})`
- `Locale define ("en-US", "Date",{months:["January","February","March","April","May","June","July","August","September","October", months_abbr:["Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec"], days:["Sunday","Monday","Tuesday","Wednesday", days_abbr:["Sun","Mon","Tue","Wed","Thu","Fri","Sat"], dateOrder:["month","date","year"], shortDate↵ :"%m/%d/%Y", shortTime:"%l:%M%p", AM:"AM", PM:"PM", firstDayOfWeek:0, ordinal:function(a){return(a >3 &&a< 21)?"th":["th","st","nd","rd","th"][Math.min(a%10, 4)];}, lessThanMinuteAgo:"less than a minute ago", minuteAgo:"about a minute ago", minutesAgo:{delta} minutes ago", hourAgo:"about an hour ago", hoursAgo:"about {delta} hours ago", dayAgo:"1 day ago", daysAgo:{delta} days ago", weekAgo:"1 week ago", weeksAgo:{delta} weeks ago", monthAgo:"1 month ago", monthsAgo:{delta} months ago", yearAgo:"1 year ago", yearsAgo:{delta} years ago", lessThanMinuteUntil:"less than a minute from now", minute↵ Until:"about a minute from now", minutesUntil:{delta} minutes from now", hourUntil:"about an hour from now", hoursUntil:"about {delta} hours from now", dayUntil:"1 day from now", daysUntil:{delta} days from now", weekUntil:"1 week from now", weeksUntil:{delta} weeks from now", monthUntil:"1 month from now", monthsUntil:{delta} months from now", yearUntil:"1 year from now", yearsUntil:{delta} years from now"))`
- `a defineParsers ("%Y([-./]%m([-./]%d((T|)%X)?)?)?" "%Y%m%d(T%H(%M%S)?)?" "%x(%X)?" "%d%o(%b(%Y)?)?(%X)?" "%b(%d%o)(%Y)(%X)?" "%Y %b(%d%o(%X)?)?" "%o %b %d %X %z %Y" "%T" "%H:%M(?%p)?"`
- `Locale Locale getCurrent ()`
- `defineParsers ({re:/^(?:tod|tom|yes)/i, handler:function(a){var b=new Date().clearTime();switch(a[0]){case"tom"↵ :return b.increment();case"yes":return b.decrement();default:return b;}}},{re:/^(next|last)([a-z]+)$/i, handler↵ :function(e){var f=new Date().clearTime();var b=f.getDay();var c=Date.parseDay(e[2], true);var a=c-b;if(c<=b){a+=7;}if(e[1]=="last"){a-=7;}return f.set("date", f.getDate()+a);}).alias("timeAgoInWords"`
- `function while (b--)`
- `invoke ("implement", Events.Pseudos(d, a.addEvent, a.removeEvent))`
- `function DOMEvent definePseudo ("keys", function(d, e, c){var g=c[0], f=[], h=this.retrieve(a,[]);f.append(d.↵ value.replace("++", function(){f.push("+");return"";}).split("+"));h.include(g.key);if(f.every(function(j){return h.contains(j)})){e.apply(this, c);this.store(a, h);if(!this.retrieve(b)){var i=function(j){(function(){h=this.↵ retrieve(a,[]).erase(j.key);this.store(a, h);}).delay(0, this);this.store(b, i).addEvent("keyup", i);})`
- `if (!window.Form)`
- `Object append (Form.Validator, Form.Validator.adders)`
- `Form Validator implement (Form.Validator.adders)`
- `Form Validator add ("IsEmpty",{errorMsg:false, test:function(a){if(a.type=="select-one"||a.type=="select"){return!(a.↵ selectedIndex >=0 &&a.options[a.selectedIndex].value!="");}else{return((a.get("value")==null)|| (a.↵ get("value").length==0));}}})`
- `Form Validator addAllThese ([["required",{errorMsg:function(){return Form.Validator.getMsg("required");}, test:function(a){return!Form.Validator.getValidator("IsEmpty").test(a);}},["length",{errorMsg:function(a, b){if(typeof(b.length)!="null"){return Form.Validator.getMsg("length").substitute({length:b.length, el↵ Length:a.get("value").length});}else{return"";}}, test:function(a, b){if(typeof(b.length)!="null"){return(a.↵ get("value").length==b.length||a.get("value").length==0);}else{return true;}}},["minLength",{errorMsg↵ :function(a, b){if(typeof(b.minLength)!="null"){return Form.Validator.getMsg("minLength").substitute({min↵ Length:b.minLength, length:a.get("value").length});}else{return"";}}, test:function(a, b){if(typeof(b.min↵ Length)!="null"){return(a.get("value").length >=(b.minLength||0));}else{return true;}}},["maxLength",{error↵ Msg:function(a, b){if(typeof(b.maxLength)!="null"){return Form.Validator.getMsg("maxLength").substitute({max↵ Length:b.maxLength, length:a.get("value").length});}else{return"";}}, test:function(a, b){return a.get("value").length<=(b.↵ maxLength||10000);}},["validate-integer",{errorMsg:Form.Validator.getMsg.pass("integer"), test:function(a){return`

- ```

Form.Validator.getValidator("IsEmpty").test(a)||(/^(?![1-9]d*[0$]).test(a.get("value")));},["validate-numeric",{error←
Msg:Form.Validator.getMsg.pass("numeric"), test:function(a){return Form.Validator.getValidator("Is←
Empty").test(a)||(/^(?:0 $0(?:d *\.|[1-9]0)d *(\.d+)?$/).test(a.get("value")));}},["validate-digits",{error←
Msg:Form.Validator.getMsg.pass("digits"), test:function(a){return Form.Validator.getValidator("IsEmpty").test(a)||(/^[d()←
:\-+#+$/.test(a.get("value")));}},["validate-alpha",{errorMsg:Form.Validator.getMsg.pass("alpha"), test←
:function(a){return Form.Validator.getValidator("IsEmpty").test(a)||(/^[a-zA-Z]+$/).test(a.get("value")));}},["validate-
alphanum",{errorMsg:Form.Validator.getMsg.pass("alphanum"), test:function(a){return Form.Validator.get←
Validator("IsEmpty").test(a)||!(\W/).test(a.get("value")));}},["validate-date",{errorMsg:function(a, b){if(Date.←
parse){var c=b.dateFormat("%x");return Form.Validator.getMsg("dateSuchAs").substitute({date:new Date().format(c)});}else{retu←
Form.Validator.getMsg("dateInFormatMDY");}}, test:function(e, g){if(Form.Validator.getValidator("Is←
Empty").test(e)){return true;}var a=Locale.getCurrent().sets.Date, b=new RegExp([a.days, a.days_←
abbr, a.months, a.months_abbr].flatten().join("|"), "i"), i=e.get("value"), f=i.match(/([a-z]+/gi);if(f &&f.←
every(b.exec, b)){return false;}var c=Date.parse(i), h=g.dateFormat("%x", d=c.format(h);if(d!="invalid
date"){e.set("value", d);return c.isValid();}},["validate-email",{errorMsg:Form.Validator.getMsg.pass("email"),
test:function(a){return Form.Validator.getValidator("IsEmpty").test(a)||(/^(?:[a-z0-9!#$%&'*=+V=?^_←
'{}~\-.]?{0, 63}[a-z0-9!#$%&'*=+V=?^_{}~\-.]@(?:[a-z0-9](?:[a-z0-9-]{0, 61}[a-z0-9])?.)*[a-z0-9](?←
:[a-z0-9-]{0, 61}[a-z0-9])?|[(?:25[0-5]|2[0-4][0-9][01]?[0-9][0-9]?)\.){3}(?:25[0-5]|2[0-4][0-9][01]?[0-
9][0-9]?)\.}$/i).test(a.get("value")));}},["validate-url",{errorMsg:Form.Validator.getMsg.pass("url"), test←
:function(a){return Form.Validator.getValidator("IsEmpty").test(a)||(/^(https?|ftp|rtmp|mms):\/\/([A-Z0-9][A-
Z0-9_]*)(\.[A-Z0-9][A-Z0-9_]*+)(:(\d+)?V?/i).test(a.get("value")));}},["validate-currency-dollar",{error←
Msg:Form.Validator.getMsg.pass("currencyDollar"), test:function(a){return Form.Validator.getValidator("←
IsEmpty").test(a)||(/^\$?-?([1-9]{1}[0-9]{0, 2}\.?[0-9]{3})?(\.[0-9]{0, 2})?[1-9]{1}d *(\.[0-9]{0, 2})?[0\.[0-
9]{0, 2})?(\.[0-9]{1, 2})?$/).test(a.get("value")));}},["validate-one-required",{errorMsg:Form.Validator.←
getMsg.pass("oneRequired"), test:function(a, b){var c=document.id(b["validate-one-required"])|a.get←
Parent(b["validate-one-required"]);return c.getElements("input").some(function(d){if(["checkbox","radio"].←
contains(d.get("type"))){return d.get("checked");}return d.get("value");}})}]]

```
- Form **Validator** **addAllThese** ([["validate-enforce-oncheck",{test:function(a, b){var c=a.getParent("form").retrieve("validator");if(!c) true;}(b.toEnforce|document.id(b.enforceChildrenOf).getElements("input, select, textarea")).map(function(d){if(a.← checked){c.enforceField(d);}else{c.ignoreField(d);c.resetField(d);}});return true;}}],["validate-ignore-oncheck",{test← :function(a, b){var c=a.getParent("form").retrieve("validator");if(!c){return true;}(b.toIgnore|document.id(b.← ignoreChildrenOf).getElements("input, select, textarea")).each(function(d){if(a.checked){c.ignoreField(d);c.← resetField(d);}else{c.enforceField(d);}});return true;}}],["validate-nospace",{errorMsg:function(){return Form.← Validator.getMsg("noSpace");}, test:function(a, b){return!a.get("value").test(/s/);}},["validate-toggle-oncheck",{test← :function(b, c){var d=b.getParent("form").retrieve("validator");if(!d){return true;}var a=c.toToggle|document.← id(c.toToggleChildrenOf).getElements("input, select, textarea");if(!b.checked){a.each(function(e){d.← ignoreField(e);d.resetField(e);}};else{a.each(function(e){d.enforceField(e);});return true;}}],["validate- reqchk-byname",{errorMsg:function(){return Form.Validator.getMsg("reqChkByNode");}, test:function(a, b){return(document.id(b.nodeId).getElements(b.selector|["input[type=checkbox], input[type=radio]").some(function(c){return c.checked;}})}],["validate-required-check",{errorMsg:function(a, b){return b.useTitle?a.get("title"):Form.← Validator.getMsg("requiredChk");}, test:function(a, b){return!!a.checked;}}],["validate-reqchk-byname",{error← Msg:function(a, b){return Form.Validator.getMsg("reqChkByName").substitute({label:b.label|a.get("type")});}, test:function(b, d){var c=d.groupName|b.get("name");var a=\$\$(document.getElementsByName(c)).some(function(g, f){return g.checked;});var e=b.getParent("form").retrieve("validator");if(a &&e){e.resetField(b);return a;}}],["validate-match",{errorMsg:function(a, b){return Form.Validator.getMsg("match").substitute({match← Name:b.matchName|document.id(b.matchInput).get("name")});}, test:function(b, c){var d=b.get("value");var a=document.id(c.matchInput)&&document.id(c.matchInput).get("value");return d &&a?d==a:true;}}],["validate- after-date",{errorMsg:function(a, b){return Form.Validator.getMsg("afterDate").substitute({label:b.after← Label|b.afterElement?Form.Validator.getMsg("startDate"):Form.Validator.getMsg("currentDate")});}, test← :function(b, c){var d=document.id(c.afterElement)?Date.parse(document.id(c.afterElement).get("value"))← :new Date();var a=Date.parse(b.get("value"));return a &&d?a >=d:true;}}],["validate-before-date",{error← Msg:function(a, b){return Form.Validator.getMsg("beforeDate").substitute({label:b.beforeLabel|b.before← Element?Form.Validator.getMsg("endDate"):Form.Validator.getMsg("currentDate")});}, test:function(b, c){var d=Date.parse(b.get("value"));var a=document.id(c.beforeElement)?Date.parse(document.id(c.before← Element).get("value")):new Date();return a &&d?a >=d:true;}}],["validate-custom-required",{errorMsg← :function(){return Form.Validator.getMsg("required");}, test:function(a, b){return a.get("value")!=b.empty← Value;}}],["validate-same-month",{errorMsg:function(a, b){var c=document.id(b.sameMonthAs)&&document.← id(b.sameMonthAs).get("value");var d=a.get("value");if(d!=""){return Form.Validator.getMsg(c?"same← Month":"startMonth");}, test:function(a, b){var d=Date.parse(a.get("value"));var c=Date.parse(document.←



- ```
id(b.sameMonthAs)&&document.id(b.sameMonthAs).get("value");return d &&c?d.format("%B")==c.↵
format("%B"):true;}},"validate-cc-num",{errorMsg:function(a){var b=a.get("value").replace(/^[0-9]/g,"");return
Form.Validator.getMsg("creditcard").substitute({length:b.length});}, test:function(c){if(Form.Validator.get↵
Validator("IsEmpty").test(c)){return true;}var g=c.get("value");g=g.replace(/^[0-9]/g,"");var a=false;if(g.↵
test(/^[40-9]{12}([0-9]{3})?$/)){a="Visa";}else{if(g.test(/^[51-5]([0-9]{14})$/)){a="Master Card";}else{if(g.↵
test(/^[347][0-9]{13}$/)){a="American Express";}else{if(g.test(/^[60110-9]{12}$/)){a="Discover";}}}}if(a){var
d=0;var e=0;for(var b=g.length-1;b >=0;--b){e=g.charAt(b).toInt();if(e==0){continue;}if((g.length-b)%2==0){e+=e;}if(e
>9){e=e.toString().charAt(0).toInt()+e.toString().charAt(1).toInt();d+=e;}if((d%10)==0){return true;}}var
f="";while(g!=""){f+=" "+g.substr(0, 4);g=g.substr(4);c.getParent("form").retrieve("validator").ignore↵
Field(c);c.set("value", f.clean());c.getParent("form").retrieve("validator").enforceField(c);return false;}}}]
• Object append (OverText,{each:function(a){return OverText.instances.each(function(c, b){if(c.element &&c.↵
text){a.call(OverText, c, b);}});}, update:function(){return OverText.each(function(a){return a.reposition();});},
hideAll:function(){return OverText.each(function(a){return a.hide(true, true);});}, showAll:function(){return
OverText.each(function(a){return a.show();});}})
• function a (b)
• Hash each (Hash.prototype, function(b, a){if(typeof b=="function"){Hash.Cookie.implement(a, function(){var
c=b.apply(this.hash, arguments);if(this.options.autoSave){this.save();}return c;}})})
• document addEvents ({keyup:d, keydown:d})
• Keyboard prototype options nonParsedEvents combine (["rebound","onrebound"])
```

Variables

- var e
- var u
- var f =this.Function
- var r =true
- f prototype overloadSetter
- f prototype overloadGetter
- f prototype extend =function(i,v){this[i]=v;}.overloadSetter()
- f prototype implement
- var o =Array.prototype.slice
- f from =function(i){return(e(i)=="function")?i:function(){return i;}}
- var k
- var p =Object.prototype.toString
- k isEnumerable
- var b ={}
 - var d =function(i){var v=e(i.prototype);return b[v]||(b[v]=[]);}
- var h
 - var t =function(i,w){if(w&&w.\$hidden){return;}var v=this[i];if(v==null||!v.\$protected){this[i]=w;}}
- var c
- Array
- Number
- Function
- RegExp
- Object
- Date
- Number prototype \$family
- var l =Object.prototype.hasOwnProperty
- Object each =Object.forEach
- var s =function(i){switch(e(i)){case"array":return i.clone();case"object":return Object.clone(i);default:return i;}}
- var a
- var j =Date.now()
- var g
- Object type =k.isObject
- var n =this.Native=function(i){return new k(i.name,i.initialize);}

- var `m` =`Array.type`
- this `$A` =`function(i){return Array.from(i).slice();}`
- this `$arguments`
- this `$chk` =`function(i){return !(i||i===0);}`
- this `$clear` =`function(i){clearTimeout(i);clearInterval(i);return null;}`
- this `$defined`
- this `$each` =`function(w,v,x){var i=e(w);((i=="arguments"||i=="collection"||i=="array"||i=="elements")?Array:↵
Object).each(w,v,x);}`
- this `$empty` =`function(){}`
- this `$extend` =`function(v,i){return Object.append(v,i);}`
- this `$H` =`function(i){return new g(i);}`
- this `$merge`
- this `$lambda` =`f.from`
- this `$mixin` =`Object.merge`
- this `$random` =`Number.random`
- this `$splat` =`Array.from`
- this `$time` =`Date.now`
- this `$type` =`function(i){var v=e(i);if(v=="elements"){return"array";}return(v=="null")?false:v;}`
- this `$unlink`
- var `$pick` =`function(){return Array.from(arguments).pick();}`
- delete `Function` prototype `bind`
- var `$try` =`Function.attempt`
- this `$uid` =`(j.ActiveXObject)?function(e){return(e.uid||(e.uid=[b++]))[0];}:function(e){return e.uid||(e.uid=b++);}`
- `p Platform` [`p.Platform.name`] =`true`
- `p Request`
- `p Features xhr` =`!(p.Request)`
- var `i`
- `p Plugins Flash` =`{version:Number(i[0]||"0."+i[1])||0,build:Number(i[2])||0}`
- `p exec`
- this `Window` =`this.$constructor=new Type("Window",function(){})`
- this `Document` =`l.$constructor=new Type("Document",function(){})`
- `l html` =`l.documentElement`
- `try {n(l.html.childNodes)`
- `p Engine` =`{}`
- this `$exec` =`p.exec`
- `a defineKey` =`function(d,c){b[d]=c;return this;}`
- `a defineKeys` =`a.defineKey.overloadSetter(true)`
- var `Event` =`DOMEvent`
- `Event Keys` =`{}`
- `a Mutators`
- this `Events`
- this `Options`
- `d parse`
- `d escapeRegExp` =`f`
- `function k isNativeCode` =`function(c){return (/^\s*[native code]\s*/).test(""+c);}`
- `k isXML`
- `k setDocument`
- `k search`
- `k uidx` =`1`
- `k uidk` =`"slick-uniqueid"`
- `k getUIDXML`
- `k getUIDHTML` =`function(c){return c.uniqueNumber||(c.uniqueNumber=this.uidx++);}`
- `k sort`
- `k cacheNTH` =`{}`
- `k matchNTH` =`/^([+-]?[d*])?([a-z]+)?([+-]?[d+])?$/`

- [k parseNTHArgument](#)
- [k createNTHPseudo](#)
- [k pushArray](#)
- [k pushUID](#) =function(q,c,s,p,n,r){var o=this.getUID(q);if(!this.uniques[o]&&this.matchSelector(q,c,s,p,n,r)){this.uniques[o]=true;this.found.push(q);}}
- [k matchNode](#)
- [Request JSON](#)
- [var Cookie](#)
- [Cookie write](#) =function(b,c,a){return new [Cookie\(b,a\).write\(c\);}](#)
- [Cookie read](#) =function(a){return new [Cookie\(a\).read\(\);}](#)
- [Cookie dispose](#)
- [else {c=true](#)
- [Element Events domready](#) ={onAdd:function(m){if(l){m.call(this);}}}
- [Element Events load](#)
- [window](#)
- [document](#)
- [function Swiff CallBacks](#) ={}
- [Swiff remote](#)
- [MooTools More](#) ={version:"1.4.0.1",build:"a4244edf2aa97ac8a196fc96082dd35af1abab87"}
- [Events definePseudo](#) =function(c,d){b[c]=d;return this;}
- [Events lookupPseudo](#) =function(c){return b[c];}
- [Class refactor](#)
- [Class Mutators Binds](#)
- [Class Mutators initialize](#)
- [Class Occlude](#)
- [f Set](#)
- [j S](#) =j.M
- [var q](#)
- [timeDiffInWords](#)
- [a regs](#) ={endSlash:/\\$/ ,scheme:/^\(w+):/,directoryDot:/\.\\$/}
- [a base](#) =new [a\(Array.from\(document.getElements\("base\[href\]",true\)\).getLast\(\),{base:document.location}\)](#)
- [URI](#)
- [Element prototype position](#)
- [var IframeShim](#)
- [var Mask](#)
- [Element Properties mask](#)
- [var Spinner](#)
- [Element Properties spinner](#)
- [function Fx Reveal](#)
- [Element Properties reveal](#)
- [Element Properties dissolve](#) =[Element.Properties.reveal](#)
- [Form Request Append](#)
- [var InputValidator](#)
- [Element Properties validators](#) ={get:function(){return(this.get("data-validators")||this.className).clean().split("");}}
- [Element Properties validatorProps](#)
- [Form Validator](#)
- [Form Validator getMsg](#)
- [Form Validator adders](#)
- [Element Properties validator](#)
- [var FormValidator](#) =[Form.Validator](#)
- [Form Validator Inline](#)
- [var OverText](#)
- [OverText instances](#) =[]
- [Fx Elements](#)

- Fx [Accordion](#)
- Fx [Move](#)
- Element Properties [move](#)
- Fx [Slide](#)
- Element Properties [slide](#)
- var [SmoothScroll](#)
- Fx [Sort](#)
- var [Drag](#)
- var [Slider](#)
- var [Sortable](#)s
- [Request JSONP](#)
- [Request JSONP](#) counter =0
- [Request JSONP](#) request_map ={}
- [Request Queue](#)
- var [Asset](#)
- this \$RGB =function(e,d,c){return new a([e,d,c],"rgb");}
- this \$HSB =function(e,d,c){return new a([e,d,c],"hsb");}
- this \$HEX
- var [HtmlTable](#)
- [HtmlTable](#) ParserPriority =["date","input-checked","input-value","float","number"]
- [HtmlTable](#) Parsers
- [HtmlTable](#) defineParsers
- a [stop](#)
- a [manager](#) =new a({active:true})
- a [trace](#)
- Keyboard [rebind](#)
- Keyboard [getActiveShortcuts](#)
- Keyboard [getShortcut](#)
- Keyboard [getShortcuts](#) =function(b,a){return Keyboard.getShortcut(b,a,{many:true});}
- var [Scroller](#)
- function this [Tips](#)

9.198.1 Function Documentation

9.198.1.1 \$uid (j)

9.198.1.2 function a (b)

9.198.1.3 Form Validator add ("IsEmpty", {errorMsg:false, test:function(a){if(a.type=="select-one"||a.type=="select"){return!(a.selectedIndex >=0 &&a.options[a.selectedIndex].value!="");}else{return((a.get("value")==null)||((a.get("value").length==0)));}} })

9.198.1.4 Form Validator addAllThese (["validate-integer",{errorMsg:Form.Validator.getMsg.pass("integer"), test:function(a){return Form.Validator.getValidator("IsEmpty").test(a)||(!^(?[-1-9] \d *| 0)

9.198.1.5 Form Validator addAllThese (input [type=radio])

9.198.1.6 i addEvent ("load", function(){f=true;})

9.198.1.7 document addEvents ({keyup:d, keydown:d})

9.198.1.8 k addListener ("DOMContentLoaded", g)

9.198.1.9 Element alias ("each", "forEach")

- 9.198.1.10 `a alias ({indexOf:"keyOf", contains:"hasValue"})`
- 9.198.1.11 `Object append (f , new Events)`
- 9.198.1.12 `Object append (c , f , {setLanguage:f.use, getCurrentLanguage:function(){var g=f.get←
Current();return(g)?g.name:null;}, set:function(){f.define.apply(this, arguments);return this;}, get:function(i, h,
g){if(h){i+="."+h;}return f.get(i, g);}})`
- 9.198.1.13 `Object append (Form. Validator, Form.Validator. adders)`
- 9.198.1.14 `Object append (OverText , {each:function(a){return OverText.instances.each(function(c, b){if(c.element
&&c.text){a.call(OverText, c, b);}});}, update:function(){return OverText.each(function(a){return
a.reposition();});}, hideAll:function(){return OverText.each(function(a){return a.hide(true, true);});},
showAll:function(){return OverText.each(function(a){return a.show();});}})`
- 9.198.1.15 `apply ((typeof exports!="undefined"?exports:this)`
- 9.198.1.16 `function b (x, s, D, z, r, C, q, B, A, y, u, F, G, v, p, w)`
- 9.198.1.17 `c ("String", String)`
- 9.198.1.18 `catch (h)`
- 9.198.1.19 `Keyboard prototype options nonParsedEvents combine ()`
- 9.198.1.20 `Locale define ("en-US" , "Date" , {months←
:["January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"],
months_abbr:["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"], days←
:["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"],
days_abbr:["Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"], dateOrder:["month", "date", "year"],
shortDate:"%m/%d/%Y", shortTime:"%l:%M%p", AM:"AM", PM:"PM", firstDayOfWeek:0,
ordinal:function(a){return(a >3 &&a < 21)?"th":["th", "st", "nd", "rd", "th"][Math.min(a%10, 4)];},
lessThanMinuteAgo:"less than a minute ago", minuteAgo:"about a minute ago", minutesAgo:"{delta} minutes ago",
hourAgo:"about an hour ago", hoursAgo:"about {delta} hours ago", dayAgo:"1 day ago", daysAgo:"{delta} days
ago", weekAgo:"1 week ago", weeksAgo:"{delta} weeks ago", monthAgo:"1 month ago", monthsAgo:"{delta} months
ago", yearAgo:"1 year ago", yearsAgo:"{delta} years ago", lessThanMinuteUntil:"less than a minute from now",
minuteUntil:"about a minute from now", minutesUntil:"{delta} minutes from now", hourUntil:"about an hour
from now", hoursUntil:"about {delta} hours from now", dayUntil:"1 day from now", daysUntil:"{delta} days from
now", weekUntil:"1 week from now", weeksUntil:"{delta} weeks from now", monthUntil:"1 month from now",
monthsUntil:"{delta} months from now", yearUntil:"1 year from now", yearsUntil:"{delta} years from now"})`
- 9.198.1.21 `a defineKeys ({"38":"up", "40":"down", "37":"left", "39":"right", "27":"esc", "32":"space", "8":"backspace", "9"←
:"tab", "46":"delete", "13":"enter"})`
- 9.198.1.22 `a defineParsers ("%Y([-/]%m([-/]%d((T|)%X)?)?)", "%Y%m%d(T%H(%M%S)?)?", "%x(%X)?", "%d%o(%b(
%Y)?)?(%X)?", "%b(%d%o)?(%Y)?(%X)?", "%Y %b(%d%o(%X)?)", "%o %b %d %X %z %Y", "%T",
"%H:%M(%p)?")`
- 9.198.1.23 `defineParsers ({re:/^(?:tod|tom|yes)/i, handler:function(a){var b=new Date().clearTime();switch(a[0]){case"tom"←
:return b.increment();case"yes":return b.decrement();default:return b;}}, {re:/^(next|last)([a-z]+)$/i,
handler:function(e){var f=new Date().clearTime();var b=f.getDay();var c=Date.parseDay(e[2], true);var
a=c-b;if(c<=b){a+=7;}if(e[1]=="last"){a=7;}return f.set("date", f.getDate()+a);}})`

- 9.198.1.24 `function DOMEvent definePseudo ("keys", function(d, e, c){var g=c[0], f=[], h=this.retrieve(a,[]);f.append(d.↵
value.replace("++", function(){f.push("+");return"";}).split("+"));h.include(g.key);if(f.every(function(j){return
h.contains(j)})){e.apply(this, c);this.store(a, h);if(!this.retrieve(b)){var i=function(j){(function(){h=this.↵
retrieve(a,[]).erase(j.key);this.store(a, h);}).delay(0, this);this.store(b, i).addEvent("keyup", i);}}
)`
- 9.198.1.25 `each (function(i){new k(i);})`
- 9.198.1.26 `Hash each (Hash. prototype, function(b, a){if(typeof b=="function"){Hash.Cookie.implement(a, function(){var
c=b.apply(this.hash, arguments);if(this.options.autoSave){this.save();}return c;});}})`
- 9.198.1.27 `Date extend ("now", function(){return+(new Date);})`
- 9.198.1.28 `Object extend ({merge:function(C, y, x){if(e(y)=="string"){return a(C, y, x);}for(var B=1, w=arguments.length;B<
w;B++){var z=arguments[B];for(var A in z){a(C, A, z[A]);}}return C;}, clone:function(i){var w={};for(var
v in i){w[v]=s(i[v]);}return w;}, append:function(z){for(var y=1, w=arguments.length;y< w;y++){var
v=arguments[y]||{};for(var x in v){z[x]=v[x];}}return z;})`
- 9.198.1.29 `for (var q in{toString:1})`
- 9.198.1.30 `Locale Locale getCurrent ()`
- 9.198.1.31 `if (r)`
- 9.198.1.32 `if (Object. create ==Function.prototype.create)`
- 9.198.1.33 `if (!l. head)`
- 9.198.1.34 `if (l. execCommand)`
- 9.198.1.35 `if (this.attachEvent &&!this. addEventListener)`
- 9.198.1.36 `if (p.Platform. ios)`
- 9.198.1.37 `if (p. ie == "unknown")`
- 9.198.1.38 `if (p.safari||p. chrome)`
- 9.198.1.39 `if (!this. Slick)`
- 9.198.1.40 `if (d.doScroll &&! h())`
- 9.198.1.41 `if ("onreadystatechange" in k)`
- 9.198.1.42 `if (k. readyState)`
- 9.198.1.43 `function if (this. Fx)`

Initial value:

```
{var a=this.Table=function(){this.length=0;var c=[],b=[];this.set=function(e,
g){var d=c.indexOf(e);
if(d== -1){var f=c.length;c[f]=e;b[f]=g;this.length++;}else{b[d]=g;}return this;};this.get=function
(e){var d=c.indexOf(e);return(d== -1)?null:b[d];};this.erase=function(e){var d=c.indexOf(
e);
if(d!=-1){this.length--;c.splice(d,1);return b.splice(d,1)[0];}return null;};this.
each=this.forEach=function(f,g){for(var e=0,d=this.length;e<d;e++){f.call(
g,c[e],b[e],this);
}};}
```

- 9.198.1.44 `if (this.Element &&Element.implement &&this.Fx)`
- 9.198.1.45 `if (!window.Form)`
- 9.198.1.46 `f.implement ({hide:function(){this.$hidden=true;return this;}, protect:function(){this.$protected=true;return this;}}`
`)`
- 9.198.1.47 `Array.implement ("clone" , function(){var v=this.length, w=new Array(v);while(v--){w[v]=s(this[v]);}return w; })`
- 9.198.1.48 `a.implement ("implement" , f.overloadSetter())`
- 9.198.1.49 `Events.implement (Events.Pseudosb, a.addEvent, a.removeEvent)`
- 9.198.1.50 `FormValidator.implement (Form.Validator.adders)`
- 9.198.1.51 `invoke ("implement" , Events.Pseudosd, a.addEvent, a.removeEvent)`
- 9.198.1.52 `new k ("Type" , k)`
- 9.198.1.53 `Document.mirror (function(e, q){j[e]=q;})`
- 9.198.1.54 `function while (b--)`

9.198.2 Variable Documentation

- 9.198.2.1 `this.$A=function(i){return Array.from(i).slice();}`
- 9.198.2.2 `this.$arguments`

Initial value:

```
=function(v){return function(){return arguments[v];};}
```

- 9.198.2.3 `this.$chk=function(i){return !(i||i===0);}`
- 9.198.2.4 `this.$clear=function(i){clearTimeout(i);clearInterval(i);return null;}`
- 9.198.2.5 `this.$defined`

Initial value:

```
=function(i){return (i!=null);}
```

- 9.198.2.6 `this.$each=function(w,v,x){var i=e(w);((i=="arguments"||i=="collection"||i=="array"||i=="elements")?Array:↵
Object).each(w,v,x);}`
- 9.198.2.7 `this.$empty=function(){}`
- 9.198.2.8 `this.$exec=p.exec`
- 9.198.2.9 `this.$extend=function(v,i){return Object.append(v,i);}`

9.198.2.10 I \$family

Initial value:

```
=function() {return isFinite(this)?"number":"null";
}.hide()
```

9.198.2.11 function this \$H =function(i){return new g(i);}

9.198.2.12 this \$HEX

Initial value:

```
=function(b) {return new a(b, "hex");
}
```

9.198.2.13 this \$HSB =function(e,d,c){return new a([e,d,c], "hsb");}

9.198.2.14 this \$lambda =f.from

9.198.2.15 this \$merge

Initial value:

```
=function() {var i=Array.slice(arguments);i.unshift({});
return Object.merge.apply(null,i); }
```

9.198.2.16 this \$mixin =Object.merge

9.198.2.17 var \$pick =function(){return Array.from(arguments).pick();}

9.198.2.18 this \$random =Number.random

9.198.2.19 this \$RGB =function(e,d,c){return new a([e,d,c], "rgb");}

9.198.2.20 this \$splat =Array.from

9.198.2.21 this \$time =Date.now

9.198.2.22 var \$try =Function.attempt

9.198.2.23 this \$type =function(i){var v=e(i);if(v=="elements"){return "array";}return(v=="null"?false:v);}

9.198.2.24 \$uid =(j.ActiveXObject)?function(e){return(e.uid||(e.uid=[b++]))[0];}:function(e){return e.uid||(e.uid=b++);}

9.198.2.25 this \$unlink

Initial value:

```
=function(i) {switch (e(i)) {case "object":return Object.clone(i);
case "array":return Array.clone(i); case "hash":return new g(i); default: return
i; }}
```

9.198.2.26 var a

Initial value:

```
=function(v,i,w){switch(e(w)){case"object":if(e(v[i])=="object"){Object.merge(v[i],w);}else{v[i]=Object.clone(w);}}break;case"array":v[i]=w.clone();break;default:v[i]=w;}return v;}
```

9.198.2.27 var Accordion

Initial value:

```
=new Class({Extends:Fx.Elements,options:{fixedHeight:false,fixedWidth:false,display:0,show:false,height:true,width:false,opacity:true,alwaysHide:false,trigger:"click",initialDisplayFx:true,resetHeight:true},initialize:function(){var g=function(h){return h!=null;};var f=Array.link(arguments,{container:Type.isElement,options:Type.isObject,togglers:g,elements:g});this.parent(f.elements,f.options);var b=this.options,e=this.togglers==$(f.togglers);this.previous=-1;this.internalChain=new Chain();if(b.alwaysHide){this.options.link="chain";if(b.show||this.options.show==0){b.display=false;this.previous=b.show;}if(b.start){b.display=false;b.show=false;}var d=this.effects={};if(b.opacity){d.opacity="fullOpacity";if(b.width){d.width=b.fixedWidth?"fullWidth":"offsetWidth";}if(b.height){d.height=b.fixedHeight?"fullHeight":"scrollHeight";}for(var c=0,a=e.length;c<a;c++){this.addSection(e[c],this.elements[c]);}this.elements.each(function(j,h){if(b.show==h){this.fireEvent("active",[e[h],j]);}else{for(var k in d){j.setStyle(k,0);}},this);if(b.display||b.display==0||b.initialDisplayFx==false){this.display(b.display,b.initialDisplayFx);}if(b.fixedHeight!=false){b.resetHeight=false;}this.addEvent("complete",this.internalChain.callChain.bind(this.internalChain));},addSection:function(g,d){g=document.id(g);d=document.id(d);this.togglers.include(g);this.elements.include(d);var f=this.togglers,c=this.options,h=f.contains(g),a=f.indexOf(g),b=this.display.pass(a,this);g.store("accordion:display",b).addEvent(c.trigger,b);if(c.height){d.setStyles({"padding-top":0,"border-top":"none","padding-bottom":0,"border-bottom":"none"});if(c.width){d.setStyles({"padding-left":0,"border-left":"none","padding-right":0,"border-right":"none"});d.fullOpacity=1;if(c.fixedWidth){d.fullWidth=c.fixedWidth;}if(c.fixedHeight){d.fullHeight=c.fixedHeight;}d.setStyle("overflow","hidden");if(!h){for(var e in this.effects){d.setStyle(e,0);}}return this;},removeSection:function(f,b){var e=this.togglers,a=e.indexOf(f),c=this.elements[a];var d=function(){e.erase(f);this.elements.erase(c);this.detach(f);}.bind(this);if(this.now==a||b!=null){this.display(b!=null?b:(a-1>0?a-1:0)).chain(d);}else{d();}return this;},detach:function(b){var a=function(c){c.removeEvent(this.options.trigger,c.retrieve("accordion:display"))};.bind(this);if(!b){this.togglers.each(a);}else{a(b);}return this;},display:function(b,c){if(!this.check(b,c)){return this;}var h={},g=this.elements,a=this.options,f=this.effects;if(c==null){c=true;}if(typeof(b)=="element"){b=g.indexOf(b);}if(b==this.previous&&!a.alwaysHide){return this;}if(a.resetHeight){var e=g[this.previous];if(e&&!this.selfHidden){for(var d in f){e.setStyle(d,e[f[d]]);}}if((this.timer&&a.link=="chain")||b===this.previous&&!a.alwaysHide){return this;}this.previous=b;this.selfHidden=false;g.each(function(l,k){h[k]={};var j;if(k!=b){j=true;}else{if(a.alwaysHide&&(l.offsetHeight>0&&a.height)||l.offsetWidth>0&&a.width){j=true;}this.selfHidden=true;}}this.fireEvent(j?"background":"active",[this.togglers[k],l]);for(var m in f){h[k][m]=j?0:1[f[m]]};if(!c&&!j&&a.resetHeight){h[k].height="auto";}},this);this.internalChain.clearChain();this.internalChain.chain(function(){if(a.resetHeight&&!this.selfHidden){var i=g[b];if(i){i.setStyle("height","auto");}}}).bind(this);return c?this.start(h):this.set(h).internalChain.callChain();})
```

9.198.2.28 Form Validator adds

Initial value:

```
=({validators:{},add:function(b,a){this.validators[b]=new InputValidator(b,a);if(!this.initialize){this.implement({validators:this.validators});}},addAllThese:function(a){Array.from(a).each(function(b){this.add(b[0],b[1]);},this);},getValidator:function(a){return this.validators[a.split(":")[0];}}}
```

9.198.2.29 Form Request Append

Initial value:

```
=new Class({Extends:Form.Request,options:{useReveal:true,revealOptions:{},inject:"bottom"},makeRequest:
function(){this.request=new Request.HTML(Object.merge({url:this.element.get("action"),method:this.e
lement.get("method")||"post",spinnerTarget:this.element},this.options.requestOptions,{evalScripts:false})).
addEvents({success:function(b,g,f,a){var c;
var d=Elements.from(f);if(d.length==1){c=d[0];}else{c=new Element("div",{styles:{display:"none"}})
.adopt(d);}c.inject(this.target,this.options.inject);
if(this.options.requestOptions.evalScripts){Browser.exec(a);}this.fireEvent("beforeEffect",c);var
e=function(){this.fireEvent("success",[c,this.target,b,g,f,a]);}
}.bind(this);if(this.options.useReveal){c.set("reveal",this.options.revealOptions).get("reveal").chain(e);c
.reveal();}else{e();}.bind(this),failure:function(a){this.fireEvent("failure",
a);}
}.bind(this));this.attachReset();}}
```

9.198.2.30 Array

9.198.2.31 var Asset

Initial value:

```
=(javascript:function(d,b){if(!b){b={};}var a=new Element("script",{src:d,
type:"text/javascript"}),e=b.document||document,c=b.onload||b.onLoad;
delete b.onload;delete b.onLoad;delete b.document;if(c){if(typeof a.onreadystatechange!="undefined"){a.
addEvent("readystatechange",function(){if(["loaded","complete"].contains(this.readyState)){
c.call(this);}
}}};else{a.addEvent("load",c);}return a.set(b).inject(e.head);},css:function(
d,a){if(!a){a={};}var b=new Element("link",{rel:"stylesheet",media:"screen",
type:"text/css",href:d});
var c=a.onload||a.onLoad,e=a.document||document;delete a.onload;delete a.onLoad;delete a.document
;if(c){b.addEvent("load",c);}return b.set(a).inject(e.head);
},image:function(c,b){if(!b){b={};}var d=new Image(),a=document.id(d)||new Element("img");["load
","abort","error"].each(function(e){var g="on"+e,f="on"+e.capitalize(),h=b[
g]||b[f]||function(){};
delete b[f];delete b[g];d[g]=function(){if(!d){return;}if(!a.parentNode){a.width=d.width;a.height=d.
height;}d=d.onload=d.onabort=d.onerror=null;h.delay(1,a,a);
a.fireEvent(e,a,1);}});d.src=a.src=c;if(d&&d.complete){d.onload.delay(1);}return a.set(b);},images:
function(c,b){c=Array.from(c);var d=function(){},a=0;
b=Object.merge({onComplete:d,onProgress:d,onError:d,properties:{},b);return new
Elements(c.map(function(f,e){return Asset.image(f,Object.append(b.properties,{onload:function(){a+
+;
b.onProgress.call(this,a,e,f);if(a==c.length){b.onComplete();}},onerror:function(){a++;b.onError.call(this,
a,e,f);if(a==c.length){b.onComplete();}}}}));
}));}
```

9.198.2.32 var b={}

9.198.2.33 a base =new a(Array.from(document.getElements("base[href]",true)).getLast(),{base:document.location})

9.198.2.34 delete Function prototype bind

9.198.2.35 Class Mutators Binds

Initial value:

```
=function(a){if(!this.prototype.initialize){this.implement("initialize",function(){});}return
Array.from(a).concat(this.prototype.Binds||[]);
}
```

9.198.2.36 var c

Initial value:

```
=function(v,z,x){var w=(z!=Object),D=z.prototype;if(w){z=new k(v,z);
}for(var A=0,y=x.length;A<y;A++){var E=x[A],C=z[E],B=D[E];if(C){C.protect();}if(w&&B){delete D[E];D[E]=B.
protect();}if(w){z.implement(D);}return c;}
```


9.198.2.37 **k** cacheNTH = {}

9.198.2.38 **function** Swiff CallBacks = {}

9.198.2.39 **function** Hash Cookie

Initial value:

```
=new Class({Implements:Options,options:{path:"/",domain:false,duration:false,secure:false,document:document,encode:true},initialize:function(b,a){this.key=b;this.setOptions(a);},write:function(b){if(this.options.encode){b=encodeURIComponent(b);}if(this.options.domain){b+=" ";domain="+this.options.domain;}if(this.options.path){b+=" ";path="+this.options.path;}if(this.options.duration){var a=new Date();a.setTime(a.getTime()+this.options.duration*24*60*60*1000);b+=" ";expires="+a.toGMTString();}if(this.options.secure){b+=" ";secure="";}this.options.document.cookie=this.key+"="+b;return this;},read:function(){var a=this.options.document.cookie.match("(?:^|;)\s*" + this.key.escapeRegExp() + "=(.*)");return (a)?decodeURIComponent(a[1]):null;},dispose:function(){new Cookie(this.key,Object.merge({},this.options,{duration:-1})).write("");return this;}})
```

9.198.2.40 **Request** JSONP counter = 0

9.198.2.41 **k** createNTHPseudo

Initial value:

```
=function(p,n,c,o){return function(s,q){var u=this.getUID(s);if(!this[c][u]){var A=s.parentNode;if(!A){return false;}var r=A[p],t=1;if(o){var z=s.nodeName;do{if(r.nodeName!=z){continue;}}this[c][this.getUID(r)]=t++;}while((r=r[n]));}else{do{if(r.nodeType!=1){continue;}}this[c][this.getUID(r)]=t++;}while((r=r[n]));}q=q||"n";var v=this.cacheNTH[q]||this.parseNTHArgument(q);if(!v){return false;}var y=v.a,x=v.b,w=this[c][u];if(y==0){return x==w;}if(y>0){if(w<x){return false;}else{if(x<w){return false;}}return((w-x)%y)==0;}}}
```

9.198.2.42 **var** d = function(i){var v=e(i.prototype);return b[v]||(b[v]=[]);}

9.198.2.43 **Date**

9.198.2.44 **a** defineKey = function(d,c){b[d]=c;return this;}

9.198.2.45 **DOMEvent** defineKeys = a.defineKey.overloadSetter(true)

9.198.2.46 **HtmlTable** defineParsers

Initial value:

```
=function(a){HtmlTable.Parsers=Object.append(HtmlTable.Parsers,a);for(var b in a){HtmlTable.ParserPriority.unshift(b);}}
```

9.198.2.47 **DOMEvent** definePseudo = function(c,d){b[c]=d;return this;}

9.198.2.48 **Cookie** dispose

Initial value:

```
=function(b,a){return new Cookie(b,a).dispose();}
```

9.198.2.49 **Element Properties** `dissolve =Element.Properties.reveal`

9.198.2.50 `this Document =!$.constructor=new Type("Document",function({}))`

9.198.2.51 `document`

9.198.2.52 **Element Events** `domready =(onAdd:function(m){if(!){m.call(this);}}`

9.198.2.53 `var Drag`

Initial value:

```
=new Class({Implements:[Events,Options],options:{snap:6,unit:"px",grid:false,style:true,limit:
  false,handle:false,invert:false,preventDefault:false,stopPropagation:false,modifiers:{x:"left",y:"top"}},
  initialize:function(){var b=Array.link(arguments,{options:Type.isObject,element:function(
    c){return c!=null;
  }}});this.element=document.id(b.element);this.document=this.element.getDocument();this.
  setOptions(b.options||{});var a=typeof(this.options.handle);this.handles=(a=="array"||a=="collection")?$(this.
  options.handle):document.id(this.options.handle)||this.element;
this.mouse={now:{},pos:{}};this.value={start:{},now:{}};this.selection=(Browser.ie)?"selectstart":"
  mousedown";if(Browser.ie&&!Drag.ondragstartFixed){document.ondragstart=
  Function.from(false);
  Drag.ondragstartFixed=true;}this.bound={start:this.start.bind(this),check:this.check.bind(this),drag:
  this.drag.bind(this),stop:this.stop.bind(this),cancel:this.cancel.bind(this),eventStop:
  Function.from(false)};
this.attach();,attach:function(){this.handles.addEvent("mousedown",this.bound.start);return this;},detach:
  function(){this.handles.removeEvent("mousedown",this.bound.start);
  return this;},start:function(a){var j=this.options;if(a.rightClick){return;}if(j.preventDefault){a.
  preventDefault();}if(j.stopPropagation){a.stopPropagation();
  }this.mouse.start=a.page;this.fireEvent("beforeStart",this.element);var c=j.limit;this.limit={x:[],y:[]};
  var e,g;for(e in j.modifiers){if(!j.modifiers[e]){continue;
  }var b=this.element.getStyle(j.modifiers[e]);if(b&&!b.match(/px$/)){if(!g){g=this.element.getCoordinates(
  this.element.getOffsetParent());}b=g[j.modifiers[e]];
  }if(j.style){this.value.now[e]=(b||0).toInt();}else{this.value.now[e]=this.element[j.modifiers[
  e]];}if(j.invert){this.value.now[e]*=-1;}this.mouse.pos[e]=a.page[e]-this.value.now[
  e];
  }if(c&&c[e]){var d=2;while(d--){var f=c[e][d];if(f||f==0){this.limit[e][d]=(typeof f=="function")?
  f():f;}}}if(typeof(this.options.grid=="number"){this.options.grid={x:this.options.grid,y:this.options.
  grid};
  }var h={mousemove:this.bound.check,mouseup:this.bound.cancel};h[this.selection]=this.bound.eventStop;this.
  document.addEvents(h);,check:function(a){if(this.options.preventDefault){a.preventDefault();
  }var b=Math.round(Math.sqrt(Math.pow(a.page.x-this.mouse.start.x,2)+Math.pow(a.page.y-this.mouse.start.y,2)
  ));if(b>this.options.snap){this.cancel();}this.document.addEvents({mousemove:this.bound.drag,mouseup:
  this.bound.stop});
  this.fireEvent("start",[this.element,a]).fireEvent("snap",this.element);},drag:function(
  b){var a=this.options;if(a.preventDefault){b.preventDefault();
  }this.mouse.now=b.page;for(var c in a.modifiers){if(!a.modifiers[c]){continue;}this.value.now[
  c]=this.mouse.now[c]-this.mouse.pos[c];if(a.invert){this.value.now[c]*=-1;
  }if(a.limit&&this.limit[c]){if((this.limit[c][1]||this.limit[c][1]==0)&&(this.value.now[c]>this.limit[c][1]
  )){this.value.now[c]=this.limit[c][1];}else{if((this.limit[c][0]||this.limit[c][0]==0)&&(this.value.now[
  c]<this.limit[c][0])){this.value.now[c]=this.limit[c][0];
  }}}if(a.grid[c]){this.value.now[c]-=(this.value.now[c]-(this.limit[c][0]||0))%a.grid[
  c];}if(a.style){this.element.setStyle(a.modifiers[c],this.value.now[c]+a.unit);
  }else{this.element[a.modifiers[c]]=this.value.now[c];}this.fireEvent("drag",[this.element,b]);},cancel:
  function(a){this.document.removeEvents({mousemove:this.bound.check,mouseup:this.bound.cancel});
  if(a){this.document.removeEvent(this.selection,this.bound.eventStop);this.fireEvent("cancel",this.
  element);},stop:function(b){var a={mousemove:this.bound.drag,mouseup:this.bound.stop};
  a[this.selection]=this.bound.eventStop;this.document.removeEvents(a);if(b){this.fireEvent("complete
 ",[this.element,b]);}}}
```

9.198.2.54 `var e`

Initial value:

```
=this.typeOf=function(i){if(i==null){return"null";}if(i.$family){return i.$family();
  }if(i.nodeName){if(i.nodeType==1){return"element";}if(i.nodeType==3){return(/\
  S/).test(i.nodeValue)?"textnode":"whitespace";}else{if(typeof i.length=="number"){if(
  i.caller){return"arguments";
  }if("item" in i){return"collection";}}return typeof i;
  }
```

9.198.2.55 `a each =Object.forEach`

9.198.2.56 Fx Elements

Initial value:

```
=new Class({Extends:Fx.CSS, initialize:function(b,a){this.elements=this.subject=$$(
  b);this.parent(a);}, compute:function(g,h,j){var c={};
  for(var d in g){var a=g[d],e=h[d],f=c[d]={};for(var b in a){f[b]=this.parent(a[
    b],e[b],j);}}return c;}, set:function(b){for(var c in b){if(!this.elements[c]){continue;
  }var a=b[c];for(var d in a){this.render(this.elements[c],d,a[d],this.options.unit);}}return this;},
  start:function(c){if(!this.check(c)){return this;};var h={},j={};
  for(var d in c){if(!this.elements[d]){continue;};var f=c[d],a=h[d]={},g=j[
    d]={};for(var b in f){var e=this.prepare(this.elements[d],b,f[b]);a[b]=e.from;
  g[b]=e.to;}}return this.parent(h,j);}}
```

9.198.2.57 else {c=true

9.198.2.58 p Engine ={}

9.198.2.59 d escapeRegExp =f

9.198.2.60 var Event =DOMEvent

9.198.2.61 this Events

Initial value:

```
=new Class({$events:{}, addEvent:function(d,c,b){d=a(d);
  if(c==empty){return this;};this.$events[d]=(this.$events[d]||[]).include(c);if(
    b){c.internal=true;return this;}, addEvents:function(b){for(var c in
    b){this.addEvent(c,b[c]);}
  }return this;}, fireEvent:function(e,c,b){e=a(e);var d=this.$events[e];if(!d){return this;};c=
  Array.from(c);d.each(function(f){if(b){f.delay(b,this,c);}else{f.apply(this,c);
  }},this);return this;}, removeEvent:function(e,d){e=a(e);var c=this.$events[
  e];if(c&&!d.internal){var b=c.indexOf(d);if(b!=-1){delete c[b];}}return this;
  }, removeEvents:function(d){var e;if(typeof d=="object"){for(e in d){this.removeEvent(e,d[e]);}return
  this;};if(d){d=a(d);}for(e in this.$events){if(d&&d!=e){continue;
  }var c=this.$events[e];for(var b=c.length;b--){if(b in c){this.removeEvent(e,c[b]);}}return this;}}
```

9.198.2.62 p exec

Initial value:

```
=function(q){if(!q){return q;};if(j.execScript){j.execScript(q);}else{var e=
  1.createElement("script");e.setAttribute("type","text/javascript");e.text=q;
  1.head.appendChild(e);1.head.removeChild(e);}return q;}
```

9.198.2.63 a extend =function(i,v){this[i]=v;}.overloadSetter()

9.198.2.64 var f =this.Function

9.198.2.65 p Plugins Flash ={version:Number(i[0]||"0."+i[1])||0,build:Number(i[2])||0}

9.198.2.66 var FormValidator =Form.Validator

9.198.2.67 Elements from =function(i){return(e(i)=="function"?i:function(){return i;};}

9.198.2.68 Function

9.198.2.69 var g

Initial value:

```
=this.Hash=new k("Hash",function(i){if(e(i)=="hash"){i=Object.clone(i.getClean());}
for(var v in i){this[v]=i[v];}return this;});
```

9.198.2.70 Keyboard getActiveShortcuts

Initial value:

```
=function(b){var a=[],c=[];Keyboard.each(b,[],.push.bind(a));a.each(function(
d){c.extend(d.getShortcuts());}
});return c;}
```

9.198.2.71 Form Validator getMsg

Initial value:

```
=function(a){return Locale.get("FormValidator."+a);
}
```

9.198.2.72 Keyboard getShortcut

Initial value:

```
=function(c,b,d){d=d||{};var a=d.many?[]:null,e=d.many?function(g){var
f=g.getShortcut(c);if(f){a.push(f);}:function(f){if(!a){a=f.getShortcut(c);}
};Keyboard.each(b,e);return a;}
```

9.198.2.73 Keyboard getShortcuts =function(b,a){return Keyboard.getShortcut(b,a,{many:true});}

9.198.2.74 k getUIDHTML =function(c){return c.uniqueNumber||(c.uniqueNumber=this.uidx++);}

9.198.2.75 k getUIDXML

Initial value:

```
=function(n){var c=n.getAttribute(this.uidk);
if(!c){c=this.uidx++;n.setAttribute(this.uidk,c);}return c;}
```

9.198.2.76 var h

Initial value:

```
=function(w,A){if(A&&A.$hidden){return;}var v=d(this);for(var x=0;x<v.length;
x++){var z=v[x];if(e(z)=="type"){h.call(z,w,A);}else{z.call(this,w,A);}}var y=this.prototype[
w];if(y==null||!y.$protected){this.prototype[w]=A;if(this[w]==null&&
e(A)=="function"){t.call(this,w,function(i){return A.apply(i,o.call(arguments,1));}
)}};}
```

9.198.2.77 I html =I.documentElement

9.198.2.78 HtmlTable

Initial value:

```

=new Class({Implements:[Options,Events,Class.Occlude],options:{properties:{cellpadding:0,
cellspacing:0,border:0},rows:[],headers:[],footers:[],property:"HtmlTable",initialize:function(){var
a=Array.link(arguments,{options:Type.isObject,table:Type.isElement,id:Type.isString});
this.setOptions(a.options);if(!a.table&&a.id){a.table=document.id(a.id);}this.element=a.table||new
Element("table",this.options.properties);if(this.occlude()){return this.occluded;
}this.build();,build:function(){this.element.store("HtmlTable",this);this.body=
document.id(this.element.tBodies[0])||new Element("tbody").inject(this.element);
$$this.body.rows;if(this.options.headers.length){this.setHeaders(this.options.headers);}else{this.tthead=
document.id(this.element.tHead);}if(this.tthead){this.head=this.getHead();
}if(this.options.footers.length){this.setFooters(this.options.footers);}this.ttfoot=
document.id(this.element.tFoot);if(this.ttfoot){this.foot=document.id(this.ttfoot.rows[0]);
}this.options.rows.each(function(a){this.push(a);,this);},toElement:function(){return this.element;},empty
:function(){this.body.empty();return this;},set:function(e,a){var d=(e=="headers")?"tHead":"tfoot",
b=d.toLowercase();
this[b]=(document.id(this.element[d])||new Element(b).inject(this.element,"top")).empty();var
c=this.push(a,{,this[b],e=="headers"?"th":"td");if(e=="headers"){this.head=this.getHead();
}else{this.foot=this.getHead();}return c;},getHead:function(){var a=this.tthead.rows;return a.length>1?$$
a:a.length?document.id(a[0]):false;},setHeaders:function(a){this.set("headers",a);
return this;},setFooters:function(a){this.set("footers",a);return this;},update:function(
d,e,a){var b=d.getChildren(a||"td"),c=b.length-1;e.each(function(i,f){var
j=b[f]||new Element(a||"td").inject(d),h=(i?i.content:"")||i,g=typeof(h);
if(i&&i.properties){j.set(i.properties);}if(/(element(s?)|array|collection)/.test(
g)){j.empty().adopt(h);}else{j.set("html",h);}if(f>c){b.push(j);}else{b[f]=
j;
}}});return{tr:d,tds:b};},push:function(e,c,d,a,b){if(typeof(e)=="element"&&e.get("tag")=="tr"){e.
inject(d||this.body,b);return{tr:e,tds:e.getChildren("td")};
}return this.update(new Element("tr",c).inject(d||this.body,b),e,a);},pushMany:function(
d,c,e,a,b){return d.map(function(f){return this.push(f,c,e,a,b);
},this);});

```

9.198.2.79 vari

Initial value:

```

=(Function.attempt(function(){return navigator.plugins["Shockwave Flash"].description;
},function(){return new ActiveXObject("ShockwaveFlash.ShockwaveFlash").GetVariable("$version");})||"0 r0").
match(/\d+/g)

```

9.198.2.80 var IFrameShim

Initial value:

```

=new Class({Implements:[Options,Events,Class.Occlude],options:{className:"iframeShim",src:'
javascript:false;document.write("");',display:false,zIndex:null,margin:0,offset:{x:0,y:0},
browsers:(Browser.ie6||(Browser.firefox&&Browser.version<3&&Browser.Platform.mac))},property:"
IframeShim",initialize:function(b,a){this.element=document.id(b);
if(this.occlude()){return this.occluded;}this.setOptions(a);this.makeShim();return this;},makeShim:
function(){if(this.options.browsers){var c=this.element.getStyle("zIndex").toInt();
if(!c){c=1;var b=this.element.getStyle("position");if(b=="static"||!b){this.element.setStyle("position",
"relative");this.element.setStyle("zIndex",c);
}c=(this.options.zIndex!=null||this.options.zIndex===0)&&c>this.options.zIndex?this.options.zIndex:c-
1;if(c<0){c=1;}this.shim=new Element("iframe",{src:this.options.src,scrolling:"no",frameborder:0,styles:{
zIndex:c,position:"absolute",border:"none",filter:"
progid:DXImageTransform.Microsoft.Alpha(style=0,opacity=0)},"class":this.options.className}).store("IframeShim",this);
var a=(function(){this.shim.inject(this.element,"after");this[this.options.display?"show":"hide"]();this.
fireEvent("inject");}).bind(this);if(!IFrameShim.ready){window.addEvent("load",a);
}else{a();}else{this.position=this.hide=this.show=this.dispose=
Function.from(this);},position:function(){if(!IFrameShim.ready||!this.shim){
return this;
}var a=this.element.measure(function(){return this.getSize();});if(this.options.margin!=undefined){a.x=a.x-
(this.options.margin*2);a.y=a.y-(this.options.margin*2);
this.options.offset.x+=this.options.margin;this.options.offset.y+=this.options.margin;}this.shim.set({width
:a.x,height:a.y}).position({relativeTo:this.element,offset:this.options.offset});
return this;},hide:function(){if(this.shim){this.shim.setStyle("display","none");return this;},show:
function(){if(this.shim){this.shim.setStyle("display","block");
}return this.position();},dispose:function(){if(this.shim){this.shim.dispose();}return this;
},destroy:function(){if(this.shim){this.shim.destroy();}return this;
}})

```

9.198.2.81 Keyboard implement

Initial value:

```
=function(i,v){this.prototype[i]=v;
}.overloadSetter()
```

9.198.2.82 Class Mutators initialize

Initial value:

```
=function(a){return function(){Array.from(this.Binds).each(function(b){var
c=this[b];if(c){this[b]=c.bind(this);}},this);return a.apply(this,arguments);
}};
```

9.198.2.83 Form Validator Inline

Initial value:

```
=new Class({Extends:Form.Validator,options:{showError:function(a){if(a.reveal){
a.reveal();}else{a.setStyle("display","block");}},hideError:function(a){if(a.dissolve){a.dissolve();}else{a.setStyle("display","none");}},
scrollToErrorsOnSubmit:true,scrollToErrorsOnBlur:false,scrollToErrorsOnChange:false,scrollFxOptions:{transition:"
quad:out",offset:{y:-20}}},initialize:function(b,a){this.parent(b,a);
this.addEvent("onElementValidate",function(g,f,e,h){var d=this.getValidator(
e);if(!g&&d.getError(f)){if(h){f.addClass("warning");}var c=this.makeAdvice(
e,f,d.getError(f),h);
this.insertAdvice(c,f);this.showAdvice(e,f);}else{this.hideAdvice(e,f);}});},makeAdvice:function(
d,f,c,g){var e=(g)?this.warningPrefix:this.errorPrefix;
e+=(this.options.useTitles)?f.title|c:c;var a=(g)?"warning-advice":"validation-advice";var
b=this.getAdvice(d,f);if(b){b=b.set("html",e);}else{b=new Element("div",{html:
e,styles:{display:"none"},id:"advice-"+d.split(":")[0]+"-"+this.getFieldId(f)}).addClass(
a);
}f.store("$moo:advice-"+d,b);return b;},getFieldId:function(a){return a.id?a.id:
a.id="input_"+a.name;},showAdvice:function(b,c){var a=this.getAdvice(b,c);
if(a&&!c.retrieve("$moo:"+this.getPropName(b))&&(a.getStyle("display")=="none"||a.getStyle("visibility")=="
hidden"||a.getStyle("opacity")==0)){c.store("$moo:"+this.getPropName(b),true);
this.options.showError(a);this.fireEvent("showAdvice",[c,a,b]);}},hideAdvice:function(
b,c){var a=this.getAdvice(b,c);if(a&&c.retrieve("$moo:"+this.getPropName(b))){c.store("$moo:"+this.
getPropName(b),false);
this.options.hideError(a);this.fireEvent("hideAdvice",[c,a,b]);}},getPropName:function(
a){return"advice"+a;},resetField:function(a){a=document.id(a);if(!a){return this;
}}this.parent(a);a.get("validators").each(function(b){this.hideAdvice(b,a);},this);return this;},
getAllAdviceMessages:function(d,c){var b=[];if(d.hasClass("ignoreValidation")&&!c){return
b;
}var a=d.get("validators").some(function(g){var e=g.test("^warn-")||d.hasClass("warnOnly");if(e){
g=g.replace(/^warn-/,"");var f=this.getValidator(g);if(!f){return
}b.push({message:f.getError(d),warnOnly:e,passed:f.test(),validator:f});},this);return
b;},getAdvice:function(a,b){return b.retrieve("$moo:advice-"+a);
},insertAdvice:function(a,c){var b=c.get("validatorProps");if(!b.msgPos||!
document.id(b.msgPos)){if(c.type&&c.type.toLowerCase()=="radio"){c.getParent().adopt(a);
}else{a.inject(document.id(c),"after");}}else{document.id(b.msgPos).grab(a);}},
validateField:function(g,f,b){var a=this.parent(g,f);if(((this.options.scrollToErrorsOnSubmit&&b==
null)||b)&&!a){var c=document.id(this).getElement(".validation-failed");
var d=document.id(this).getParent();while(d!=document.body&&d.getScrollSize().y==d.getSize(
).y){d=d.getParent();}var e=d.retrieve("$moo:fvScroller");if(!e&&window.Fx&&Fx.Scroll){e=new Fx.Scroll
(d,this.options.scrollFxOptions);
d.store("$moo:fvScroller",e);}if(c){if(e){e.toElement(c);}else{d.scrollTo(d.getScroll().x,c.getPosition(d).
y-20);}}return a;},watchFields:function(a){a.each(function(b){if(this.options.evaluateFieldsOnBlur){b.add
Event("blur",this.validationMonitor.pass([b,false,this.options.scrollToErrorsOnBlur],this));
}if(this.options.evaluateFieldsOnChange){b.addEvent("change",this.validationMonitor.pass([b,true,this.optio
ns.scrollToErrorsOnChange],this));}},this);}});
```

9.198.2.84 var InputValidator

Initial value:

```
=this.InputValidator=new Class({Implements:[Options],options:{errorMsg:"Validation
Failed.",test:Function.from(true)},initialize:function(b,a){this.setOptions(
a);
this.className=b;},test:function(b,a){b=document.id(b);return(b)?this.options.test(
b,a||this.getProps(b):false;},getError:function(c,a){c=document.id(
c);
var b=this.options.errorMsg;if(typeof(b)=="function"){b=b(c,a||this.getProps(
c));}return b;},getProps:function(a){a=document.id(a);return(a)?a.get("validatorProps"):{};
}});
```

9.198.2.85 OverText instances =[]

9.198.2.86 k isEnumerable

Initial value:

```
=function(i){return(i!=null&&typeof i.length=="number"&&p.call(i)!="[object Function]");}
}
```

9.198.2.87 function k isNativeCode =function(c){return(/^(?!\s*\[native code\]\s*)\$/).test(""+c);}

9.198.2.88 k isXML

Initial value:

```
=function(c){return(!!c.xmlVersion)||(!c.xml)||(!d.call(c)=="[object XMLDocument]")||
(c.nodeType==9&&c.documentElement.nodeName!="HTML");}
}
```

9.198.2.89 var j =Date.now()

9.198.2.90 Request JSON

Initial value:

```
=new Class({Extends:Request,options:{secure:true},initialize:function(
a){this.parent(a);Object.append(this.headers,{Accept:"application/json","X-Request":"JSON"});
},success:function(c){var b;try{b=this.response.json=JSON.decode(c,this.options.secure);}catch(
a){this.fireEvent("error",[c,a]);return;}if(b==null){this.onFailure();}
}else{this.onSuccess(b,c);}}})
```

9.198.2.91 Request JSONP

Initial value:

```
=new Class({Implements:[Chain,Events,Options],options:{onRequest:function(
a){if(this.options.log&&window.console&&console.log){console.log("JSONP retrieving script with url:"
+a);}
},onError:function(a){if(this.options.log&&window.console&&console.warn){console.warn("JSONP "+
a+" will fail in Internet Explorer, which enforces a 2083 bytes length limit on URIs");}
},url:"",callbackKey:"callback",injectScript:document.head,data:"",link:"ignore",timeout:0,log:
false},initialize:function(a){this.setOptions(a);},send:function(c){if(!
Request.prototype.check.call(this,c)){return this;
}this.running=true;var d=typeof(c);if(d=="string"||d=="element"){c={data:c;}}
c=Object.merge(this.options,c||{});var e=c.data;switch(typeof(e)){case"element":e=
document.id(e).toQueryString();
break;case"object":case"hash":e=Object.toQueryString(e);}var b=this.index=
Request.JSONP.counter++;var f=c.url+(c.url.test("\\?")?"&":"?")+c.callbackKey+"
=Request.JSONP.request_map.request_"+b+(e?"&"+e:"");
if(f.length>2083){this.fireEvent("error",f);}Request.JSONP.request_map["request_"+
b]=function(){this.success(arguments,b);}.bind(this);var a=this.getScript(f).inject(
c.injectScript);
this.fireEvent("request",[f,a]);if(c.timeout){this.timeout.delay(c.timeout,this);return this;},getScript
:function(a){if(!this.script){this.script=new Element("script",{type:"text/javascript",async:true,src:a
});}
return this.script;},success:function(b,a){if(!this.running){return;}this.clear().fireEvent("complete",b
).fireEvent("success",b).callChain();},cancel:function(){if(this.running){this.clear().fireEvent("cancel");}
return this;},isRunning:function(){return !!this.running;},clear:function(){this.running=false;if(this.
script){this.script.destroy();this.script=null;
}return this;},timeout:function(){if(this.running){this.running=false;this.fireEvent("timeout",[this.script
.get("src"),this.script]).fireEvent("failure").cancel();
}return this;}}})
```

9.198.2.92 var k**Initial value:**

```
=this.Type=function(x,w){if(x){var v=x.toLowerCase();var i=function(y){return(
e(y)==v)};k["is"+x]=i;if(w!=null){w.prototype.$family=(function(){return v;
}).hide();w.type=i;}}if(w==null){return null;}w.extend(this);w.$constructor=
k;w.prototype.$constructor=w;return w;}
```

9.198.2.93 Event Keys ={}**9.198.2.94 var l =Object.prototype.hasOwnProperty****9.198.2.95 Element Events load****Initial value:**

```
={base:"load",onAdd:function(m){if(f&&this==i){m.call(this);
}},condition:function(){if(this==i){g();delete Element.Events.load;}return true;}}
```

9.198.2.96 Events lookupPseudo =function(c){return b[c];}**9.198.2.97 j m =Array.type****9.198.2.98 a manager =new a({active:true})****9.198.2.99 var Mask****Initial value:**

```
=new Class({Implements:[Options,Events],Binds:["position"],options:{style:{}, "class": "
mask",maskMargins:false,useIframeShim:true,iframeShimOptions:{},initialize:function(
b,a){this.target=document.id(b)||document.id(document.body);
this.target.store("mask",this);this.setOptions(a);this.render();this.inject();},render:function(){this.
element=new Element("div",{ "class":this.options["class"],id:this.options.id||"mask-"+String.uniqueID(),styles:
Object.merge({},this.options.style,{display:"none"}),events:{click:function(
a){this.fireEvent("click",a);
if(this.options.hideOnClick){this.hide();}.bind(this)}});this.hidden=true;},toElement:function(){return
this.element;},inject:function(b,a){a=a||(this.options.inject?this.options.inject.where:"")||this.target==
document.body?"inside":"after";
b=b||(this.options.inject&&this.options.inject.target)||this.target;this.element.inject(
b,a);if(this.options.useIframeShim){this.shim=new IframeShim(this.element,this.options.
iframeShimOptions);
this.addEvents({show:this.shim.show.bind(this.shim),hide:this.shim.hide.bind(this.shim),destroy:
this.shim.destroy.bind(this.shim)});},position:function(){this.resize(this.options.width,this.options.
height);
this.element.position({relativeTo:this.target,position:"topLeft",ignoreMargins:!this.options.
maskMargins,ignoreScroll:this.target==document.body});return this;
},resize:function(a,e){var b={styles:["padding","border"]};if(this.options.maskMargins){b.styles.push("
margin");}var d=this.target.getComputedSize(b);if(this.target==document.body){this.element.setStyles
({width:0,height:0});
var c=window.getScrollSize();if(d.totalHeight<c.y){d.totalHeight=c.y;}if(d.totalWidth<c.x){d.
totalWidth=c.x;}this.element.setStyles({width:Array.pick([a,d.totalWidth,d.x]),height:
Array.pick([e,d.totalHeight,d.y])});
return this;},show:function(){if(!this.hidden){return this;}window.addEvent("resize",this.
position);this.position();this.showMask.apply(this,arguments);
return this;},showMask:function(){this.element.setStyle("display","block");this.hidden=false;this.fireEvent
("show");},hide:function(){if(this.hidden){return this;}
}window.removeEvent("resize",this.position);this.hideMask.apply(this,arguments);if(this.
options.destroyOnHide){return this.destroy();}return this;},hideMask:function(){this.element.setStyle("display","
none");
this.hidden=true;this.fireEvent("hide");},toggle:function(){this[this.hidden?"show":"hide"]();},destroy:
function(){this.hide();this.element.destroy();this.fireEvent("destroy");
this.target.eliminate("mask");}}
```


9.198.2.100 Element Properties mask**Initial value:**

```
= {set: function(b) {var a=this.retrieve("mask"); if(a) {a.destroy();} return this.eliminate("mask").store("mask:options",b);}, get: function() {var a=this.retrieve("mask"); if(!a) {a=new Mask(this,this.retrieve("mask:options")); this.store("mask",a);} return a;}}
```

9.198.2.101 k matchNode**9.198.2.102 k matchNTH** `=/^([+]?[d*])?([a-z]+)?([+]?[d+])?$/`**9.198.2.103 MooTools More** `= {version:"1.4.0.1",build:"a4244edf2aa97ac8a196fc96082dd35af1abab87"}`**9.198.2.104 Drag Move****Initial value:**

```
= new Class({Extends:Fx.Morph,options:{relativeTo:document.body,position:"center",edge:false,offset:{x:0,y:0}},start:function(a){var b=this.element,c=b.getStyles("top","left"); if(c.top=="auto"||c.left=="auto"){b.setPosition(b.getPosition(b.getOffsetParent()));} return this.parent(b.position(Object.merge({},this.options,a,{returnPos:true})));}}
```

9.198.2.105 Element Properties move**Initial value:**

```
= {set: function(a) {this.get("move").cancel().setOptions(a); return this;}, get: function() {var a=this.retrieve("move"); if(!a) {a=new Fx.Move(this,{link:"cancel"}); this.store("move",a);} return a;}}
```

9.198.2.106 a Mutators**Initial value:**

```
= {Extends: function(g) {this.parent=g; this.prototype=d(g);}, Implements: function(g) {Array.from(g).each(function(j) {var h=new j; for(var i in h) {f.call(this,i,h[i],true);}}, this);}}
```

9.198.2.107 var n `= this.Native=function(i){return new k(i.name,i.initialize);}`**9.198.2.108 Number****9.198.2.109 var o** `= Array.prototype.slice`**9.198.2.110 Object****9.198.2.111 Class Occlude****Initial value:**

```
= new Class({occlude: function(c,b) {b=document.id(b)||this.element; var a=b.retrieve(c)||this.property; if(a&&!this.occluded) {return (this.occluded=a);} this.occluded=false; b.store(c)||this.property,this; return this.occluded;}}
```

9.198.2.112 this Options

Initial value:

```
=new Class({setOptions:function(){var b=this.options=Object.merge.apply(
    null,[{}],this.options).append(arguments));
if(this.addEvent){for(var c in b){if(typeof(b[c])!="function"||!(/^on[A-Z]/).test(c)){continue;}
    this.addEvent(c,b[c]);delete b[c];}}return this;}}
```

9.198.2.113 f prototype overloadGetter

Initial value:

```
=function(v){var i=this;
return function(x){var y,w;if(v||typeof x!="string"){y=x;}else{if(arguments.length>1){y=arguments;}}if(y){
    w={};for(var z=0;z<y.length;z++){w[y[z]]=i.call(this,y[z]);}
}}else{w=i.call(this,x);}return w;}}
```

9.198.2.114 f prototype overloadSetter

Initial value:

```
=function(v){var i=this;return function(x,w){if(x==null){return this;}if(v||typeof x!="string"){for(
    var y in x){i.call(this,y,x[y]);}
}if(r){for(var z=r.length;z--){y=r[z];if(x.hasOwnProperty(y)){i.call(this,y,x[y]);}}}else{i.call(this,
    x,w);}return this;}}
```

9.198.2.115 var OverText

Initial value:

```
=new Class({Implements:[Options,Events,Class.Occlude],Binds:["reposition","assert","focus",
    "hide"],options:{element:"label",labelClass:"overTxtLabel",positionOptions:{
    position:"upperLeft",edge:"upperLeft",offset:{x:4,y:2}},poll:false,pollInterval:250,wrap:false},
    property:"OverText",initialize:function(b,a){b=this.element=document.id(
    b);
if(this.occlude()){return this.occluded;}this.setOptions(a);this.attach(b);
    OverText.instances.push(this);if(this.options.poll){this.poll();}},toElement:function(){return
    this.element;
},attach:function(){var b=this.element,a=this.options,c=a.textOverride||b.get("alt")||b.get("title");if(
    !c){return this;}var d=this.text=new Element(a.element,{"class":a.labelClass,styles:{lineHeight:"normal",po
    sition:"absolute",cursor:"text"},html:c,events:{click:this.hide.pass(a.element=="label",this)}}).inject(
    b,"after");
if(a.element=="label"){if(!b.get("id")){b.set("id","input_"+String.uniqueID());}d.set("for",b.get("id"));}
    if(a.wrap){this.textHolder=new Element("div.overTxtWrapper",{styles:{lineHeight:"normal",
    position:"relative"}}).grab(d).inject(b,"before");
}return this.enable();},destroy:function(){this.element.eliminate(this.property);this.disable();if(
    this.text){this.text.destroy();}if(this.textHolder){this.textHolder.destroy();}
}return this;},disable:function(){this.element.removeEvents({focus:this.focus,blur:this.assert,change:this.
    assert});window.removeEvent("resize",this.reposition);
this.hide(true,true);return this;},enable:function(){this.element.addEvents({focus:this.focus,blur:this.
    assert,change:this.assert});window.addEvent("resize",this.reposition);
this.reposition();return this;},wrap:function(){if(this.options.element=="label"){if(!this.element.get("id")
    ){this.element.set("id","input_"+String.uniqueID());}
}this.text.set("for",this.element.get("id"));},startPolling:function(){this.pollingPaused=false;return
    this.poll();},poll:function(a){if(this.poller&&!a){return this;
}if(a){clearInterval(this.poller);}else{this.poller=(function(){if(!this.pollingPaused){this.assert(true);}
    }).periodical(this.options.pollInterval,this);
}return this;},stopPolling:function(){this.pollingPaused=true;return this.poll(true);},focus:function(){if(
    this.text&&!this.text.isDisplayed()||this.element.get("disabled")){return this;
}return this.hide();},hide:function(c,a){if(this.text&&(this.text.isDisplayed()&&(!this.element.get("
    disabled")||a))){this.text.hide();this.fireEvent("textHide",[this.text,this.element]);
this.pollingPaused=true;if(!c){try{this.element.fireEvent("focus");this.element.focus();}catch(b){}}return
    this;},show:function(){if(this.text&&(this.text.isDisplayed()||this.text.show();
this.reposition();this.fireEvent("textShow",[this.text,this.element]);this.pollingPaused=false;return this
    ;},test:function(){return !this.element.get("value");
},assert:function(a){return this[this.test()?"show":"hide"](a);},reposition:function(){this.assert(true);
    if(!this.element.isVisible()){return this.stopPolling().hide();
}if(this.text&&this.test()){this.text.position(Object.merge(this.options.positionOptions,{relativeTo:
    this.element}));return this;}}
```

9.198.2.116 `var p=Object.prototype.toString`

9.198.2.117 `a parse`

Initial value:

```
=function(o){return e(o);
}
```

9.198.2.118 `k parseNTHArgument`

Initial value:

```
=function(q){var o=q.match(this.matchNTH);
if(!o){return false;}var p=o[2]||false;var n=o[1]||1;if(n=="-"){n=-1;}var c=+o[3]||0;o=(p=="n")?{
a:n,b:c}:{(p=="odd")?{a:2,b:1}:{(p=="even")?{a:2,b:0}:{a:0,b:n};
return(this.cacheNTH[q]=o);}
```

9.198.2.119 `HtmlTable ParserPriority=["date","input-checked","input-value","float","number"]`

9.198.2.120 `HtmlTable Parsers`

Initial value:

```
={date:{match:/^\d{2}[-\/ ]\d{2}[-\/ ]\d{2,4}$/ ,convert:function(){var a=
Date.parse(this.get("text").stripTags());
return (typeof(a)=="date")?a.format("db"):"";},type:"date"},input-checked:{match:/
type="(radio|checkbox)" / ,convert:function(){return this.getElement("input").checked;
}},input-value:{match:/<input/,convert:function(){return this.getElement("input").value;}},number:{match:
/^\d+[\d.,]*$/,convert:function(){return this.get("text").stripTags().toInt();
}},number:true,numberLax:{match:/^[^\d]+\d+$/,convert:function(){return this.get("text").replace(/^[^-?^
0-9]/,"").stripTags().toInt();},number:true},float:{match:/^[^\d]+\.[^\d]+/,convert:function(){return
this.get("text").replace(/^[^-?^d.]/,"").stripTags().toFloat();
}},number:true},floatLax:{match:/^[^\d]+[\d]+\.[^\d]+$/,convert:function(){return this.get("text").
replace(/^[^-?^d.]/,"").stripTags();},number:true},string:{match:null,convert:function(){return this.get("
text").stripTags().toLowerCase();
}},title:{match:null,convert:function(){return this.title;}}}
```

9.198.2.121 `p Platform[p.Platform.name]=true`

9.198.2.122 `Element prototype position`

9.198.2.123 `k pushArray`

Initial value:

```
=function(p,c,r,o,n,q){if(this.matchSelector(p,c,r,o,n,q)){this.found.push(
p);
}}
```

9.198.2.124 `k pushUID =function(q,c,s,p,n,r){var o=this.getUID(q);if(!this.uniques[o]&&this.matchSelector(q,c,s,p,n,r)){this.uniques[o]=true;this.found.push(q);}}`

9.198.2.125 `var q`

Initial value:

```
=function(s,t){if(!t){return this;}switch(s){case"a":case"A":return this.set("day",
    a.parseDay(t,true));case"b":case"B":return this.set("mo",a.parseMonth(t,true));
case"d":return this.set("date",t);case"H":case"I":return this.set("hr",t);case"m":return this.set("mo",
    t-1);case"M":return this.set("min",t);case"p":return this.set("ampm",t.replace(/\./
    g,""));
case"S":return this.set("sec",t);case"s":return this.set("ms",("0."+t)*1000);case"w":return this.set("day",
    t);case"Y":return this.set("year",t);case"y":t+=t;
if(t<100){t+=i+(t<0?100:0);}return this.set("year",t);case"z":if(t=="Z"){
    t="+00";}var u=t.match(/([+-])(\d{2})?:?(\d{2})?/);u=(u[1]+"1")*(u[2]*60+(u[3]||0))+this.
    getTimezoneOffset();
return this.set("time",this-u*60000);}return this;}
```

9.198.2.126 Request Queue

Initial value:

```
=new Class({Implements:[Options,Events],Binds:["attach","request","complete","cancel","
    success","failure","exception"],options:{stopOnFailure:true,autoAdvance:true,concurrent:1,requests:{}},
    initialize:function(a){var b;
if(a){b=a.requests;delete a.requests;}this.setOptions(a);this.requests={};this.
    queue=[];this.reqBinders={};if(b){this.addRequests(b);},addRequest:function(
    a,b){this.requests[a]=b;
this.attach(a,b);return this;},addRequests:function(a){Object.each(a,function(
    c,b){this.addRequest(b,c);},this);return this;},getName:function(a){return
    Object.keyOf(this.requests,a);
},attach:function(a,b){if(b._groupSend){return this;}[ "request","complete","cancel","success","failure","
    exception"].each(function(c){if(!this.reqBinders[a]){this.reqBinders[a]={};
}this.reqBinders[a][c]=function(){this["on"+c.capitalize()].apply(this,[a,b].
    append(arguments));}).bind(this);b.addEvent(c,this.reqBinders[a][c]);,this);
b._groupSend=b.send;b.send=function(c){this.send(a,c);return b;}.bind(this);return this;},removeRequest:
    function(b){var a=typeof(b)=="object"?this.getName(b):b;
if(!a&&typeof(a)!="string"){return this;}b=this.requests[a];if(!b){return this;}[ "request","complete","
    cancel","success","failure","exception"].each(function(c){b.removeEvent(c,this.reqBinders[a][
    c]);
},this);b.send=b._groupSend;delete b._groupSend;return this;},getRunning:function(){return
    Object.filter(this.requests,function(a){return a.running;});
},isRunning:function(){return !(Object.keys(this.getRunning()).length);},send:function(
    b,a){var c=function(){this.requests[b]._groupSend(a);this.queue.erase(c);
}.bind(this);c.name=b;if(Object.keys(this.getRunning()).length>=this.options.concurrent||(this.error
    &&this.options.stopOnFailure)){this.queue.push(c);}else{c();
}return this;},hasNext:function(a){return(!a)?!!this.queue.length:!!this.
    queue.filter(function(b){return b.name==a;}).length;},resume:function(){this.error=false;
(this.options.concurrent-Object.keys(this.getRunning()).length).times(this.runNext,this);return this;
},runNext:function(a){if(!this.queue.length){return this;
}if(!a){this.queue[0]();}else{var b;this.queue.each(function(c){if(!b&&c.name==a){b=true;c();}})
;return this;},runAll:function(){this.queue.each(function(a){a();
});return this;},clear:function(a){if(!a){this.queue.empty();}else{this.
    queue=this.queue.map(function(b){if(b.name!=a){return b;}else{return false;}}).filter(function(b)
    {return b;
});return this;},cancel:function(a){this.requests[a].cancel();return this;},onRequest:function(){this.
    fireEvent("request",arguments);},onComplete:function(){this.fireEvent("complete",arguments);
if(!this.queue.length){this.fireEvent("end");},onCancel:function(){if(this.options.autoAdvance&&!this
    .error){this.runNext();}this.fireEvent("cancel",arguments);
},onSuccess:function(){if(this.options.autoAdvance&&!this.error){this.runNext();}this.fireEvent("success",
    arguments);},onFailure:function(){this.error=true;
if(!this.options.stopOnFailure&&this.options.autoAdvance){this.runNext();}this.fireEvent("failure",
    arguments);},onException:function(){this.error=true;
if(!this.options.stopOnFailure&&this.options.autoAdvance){this.runNext();}this.fireEvent("exception",
    arguments);}})
```

9.198.2.127 var r=true

9.198.2.128 Cookie read =function(a){return new Cookie(a).read();}

9.198.2.129 Keyboard rebound

Initial value:

```
=function(b,a){Array.from(a).each(function(c){c.getKeyboard().removeEvent(
    c.keys,c.handler);c.getKeyboard().addEvent(b,c.handler);c.keys=b;
c.getKeyboard().fireEvent("rebound");});}
```

9.198.2.130 Class refactor

Initial value:

```
=function(b,a){Object.each(a,function(e,d){var c=b.prototype[d];
c=(c&&c.$origin)||c||function(){b.implement(d,(typeof e=="function"?function(){var
f=this.previous;this.previous=c;var g=e.apply(this,arguments);this.previous=
f;
return g;}:e));});return b;}
```

9.198.2.131 RegExp

9.198.2.132 a regs ={endSlash:/\$/,scheme:/^\(w+):/,directoryDot:/\.\\$/}

9.198.2.133 Swiff remote

Initial value:

```
=function(obj,fn){var rs=obj.CallFunction('<invoke name="'+fn+'" returnType="javascript">'+
__flash__argumentsToXML(arguments,2)+"</invoke>");
return eval(rs);}
```

9.198.2.134 Request

Initial value:

```
=(function(){var r=function(){return new XMLHttpRequest();};
var q=function(){return new ActiveXObject("MSXML2.XMLHTTP");};var e=function(){return new ActiveXObject
("Microsoft.XMLHTTP");};return Function.attempt(function(){r();
return r;},function(){q();return q;},function(){e();return e;});})();
```

9.198.2.135 Request JSONP request_map={}

9.198.2.136 function Fx Reveal

Initial value:

```
=new Class({Extends:Fx.Morph,options:{link:"cancel",styles:["padding","border","margin"],transitionOpacity:
!Browser.ie6,mode:"vertical",display:function(){return this.element.get("tag")!="tr"?block:"table-row";
},opacity:1,hideInputs:Browser.ie?"select,input,textarea,object,embed":null,
dissolve:function(){if(!this.hiding&&!this.showing){if(this.element.getStyle("display")!="none"){
this.hiding=true;
this.showing=false;this.hidden=true;this.cssText=this.element.style.cssText;var
d=this.element.getComputedSize({styles:this.options.styles,mode:this.options.mode});
if(this.options.transitionOpacity){d.opacity=this.options.opacity;}var c={};
Object.each(d,function(f,e){c[e]=[f,0];});this.element.setStyles({display:
Function.from(this.options.display).call(this),overflow:"hidden"});
var b=a(this);if(b){b.setStyle("visibility","hidden");}this.$chain.unshift(function(){if(this.hidden){
this.hiding=false;this.element.style.cssText=this.cssText;
this.element.setStyle("display","none");if(b){b.setStyle("visibility","visible");}}this.fireEvent("hide",
this.element);this.callChain();}.bind(this));this.start(c);
}else{this.callChain.delay(10,this);this.fireEvent("complete",this.element);this.fireEvent("hide",this.
element);}}else{if(this.options.link=="chain"){this.chain(this.dissolve.bind(this));
}else{if(this.options.link=="cancel"&&!this.hiding){this.cancel();this.dissolve();}}return this;},
reveal:function(){if(!this.showing&&!this.hiding){if(this.element.getStyle("display")=="none"){this.
hiding=false;
this.showing=true;this.hidden=false;this.cssText=this.element.style.cssText;var
d=this.element.measure(function(){d=this.element.getComputedSize({styles:this.options.styles,mode:this.
options.mode});
}.bind(this));if(this.options.heightOverride!=null){d.height=this.options.heightOverride.toInt();}if(
this.options.widthOverride!=null){d.width=this.options.widthOverride.toInt();}
if(this.options.transitionOpacity){this.element.setStyle("opacity",0);d.opacity=this.options.opacity;}var
c={height:0,display:Function.from(this.options.display).call(this);
Object.each(d,function(f,e){c[e]=0;});c.overflow="hidden";this.element.setStyles(c);var b=
a(this);if(b){b.setStyle("visibility","hidden");}this.$chain.unshift(function(){this.element.style.cssText
=this.cssText;
```

```

this.element.setStyle("display",Function.from(this.options.display).call(this));if(!this.hidden){
    this.showing=false;if(b){b.setStyle("visibility","visible");
}this.callChain();this.fireEvent("show",this.element);}.bind(this);this.start(d);}else{this.callChain();
    this.fireEvent("complete",this.element);this.fireEvent("show",this.element);
}}else{if(this.options.link=="chain"){this.chain(this.reveal.bind(this));}else{if(this.options.link=="cancel"&&!this.showing){this.cancel();this.reveal();
}}}return this;},toggle:function(){if(this.element.getStyle("display")=="none"){this.
    reveal();}else{this.dissolve();}return this;},cancel:function(){this.parent.apply(this,
    arguments);
if(this.cssText!=null){this.element.style.cssText=this.cssText;}this.hiding=false;this.showing=false;
    return this;}})

```

9.198.2.137 Element Properties reveal

Initial value:

```

={set:function(b){this.get("reveal").cancel().setOptions(b);
return this;},get:function(){var b=this.retrieve("reveal");if(!b){b=new Fx.Reveal(this);this.store("reveal",b);}return b;}}

```

9.198.2.138 `var s=function(i){switch(e(i)){case"array":return i.clone();case"object":return Object.clone(i);default:return i;}}`

9.198.2.139 `J S=J.M`

9.198.2.140 `var Scroller`

Initial value:

```

=new Class({Implements:[Events,Options],options:{area:20,velocity:1,onChange:function(
    a,b){this.element.scrollTo(a,b);
},fps:50},initialize:function(b,a){this.setOptions(a);this.element=
    document.id(b);this.docBody=document.id(this.element.getDocument().body);this.listener=(
    typeof(this.element)!="element"?this.docBody:this.element;
this.timer=null;this.bound={attach:this.attach.bind(this),detach:this.detach.bind(this),getCoords:this.
    getCoords.bind(this)};},start:function(){this.listener.addEvents({mouseover:this.bound.attach,mouseleave:
    this.bound.detach});
return this;},stop:function(){this.listener.removeEvents({mouseover:this.bound.attach,mouseleave:this.
    bound.detach});this.detach();this.timer=clearInterval(this.timer);
return this;},attach:function(){this.listener.addEvent("mousemove",this.bound.getCoords);},detach:function(
    ){this.listener.removeEvent("mousemove",this.bound.getCoords);
this.timer=clearInterval(this.timer);},getCoords:function(a){this.page=(this.listener.get("tag")=="body"?
    a.client:a.page;if(!this.timer){this.timer=this.scroll.periodical(Math.round(1000/this.options.fps),this)
    };
},scroll:function(){var c=this.element.getSize(),a=this.element.getScroll(),h=this.element!=this.
    docBody?this.element.getOffsets():{x:0,y:0},d=this.element.getScrollSize(),g={x:0,y:0},
    e=this.options.area.top||this.options.area,b=this.options.area.bottom||this.options.area;
for(var f in this.page){if(this.page[f]<(e+h[f])&&a[f]!=0){g[f]=(this.page[
    f]-e-h[f])*this.options.velocity;}else{if(this.page[f]>b+(c[f]+h[f])&&a[f]+c[
    f]!=d[f]){g[f]=(this.page[f]-c[f]+b-h[f])*this.options.velocity;
}}g[f]=g[f].round();}if(g.y||g.x){this.fireEvent("change",[a.x+g.x,a.y+
    g.y]);}}}

```

9.198.2.141 `k search`

Initial value:

```

=function(U,z,H,s){var p=this.found=(s)?null:(H||[]);if(!U){return p;}else{if(U.navigator){U=U.
    document;}else{if(!U.nodeType){return p;}}var F,O,V=this.uniques={},I=!!(H&&H.length),y=(U.nodeType==9);
if(this.document!=y?U:U.ownerDocument){this.setDocument(U);}if(I){for(O=p.length;O--;
    ){V[this.getUID(p[O])]=true;}if(typeof z=="string"){var r=z.match(f);
simpleSelectors:if(r){var u=r[1],v=r[2],A,E;if(!u){if(v=="*"&&this.brokenStarGEBTN){break simpleSelectors;
    }E=U.getElementsByTagName(v);if(s){return E[0]||null;
}for(O=0;A=E[O++];){if(!(I&&V[this.getUID(A)])){p.push(A);}}else{if(u=="#"){if(!this.isHTMLDocument||!y){
    break simpleSelectors;A=U.getElementById(v);
if(!A){return p;}if(this.idGetName&&A.getAttributeNode("id").nodeValue!=v){break simpleSelectors;}if(
    s){return A||null;}if(!(I&&V[this.getUID(A)])){p.push(A);
}}else{if(u=="")if(!this.isHTMLDocument||!(U.getElementsByClassName||this.brokenGEBCN)&&U.
    querySelectorAll)){break simpleSelectors;}if(U.getElementsByClassName&&!this.brokenGEBCN){E=U.getElementsByClassName(v);
if(s){return E[0]||null;}for(O=0;A=E[O++];){if(!(I&&V[this.getUID(A)])){p.push(A);}}else{var T=new

```

```

    RegExp ("^|\\s" + e.escapeRegExp (v) + "(\\s|$)"); E=U.getElementsByTagName ("*");
    for (O=0; A=E[O++];) {className=A.className; if (! (className&&T.test (className))) {continue; } if (
    s) {return A; } if (! (I&&V[this.getUID (A)])) {p.push (A); }} } if (I) {this.sort (p); }
    } return (s)?null:p; } querySelector: if (U.querySelectorAll) {if (!this.isHTMLDocument ||
    g[z] || this.brokenMixedCaseQSA || (this.brokenCheckedQSA&&z.indexOf (":checked")>-1) || (this.
    brokenEmptyAttributeQSA&&h.test (z)) || (!y&&z.indexOf ("")>-1) || e.disableQSA) {break querySelector;
    } var S=z, x=U; if (!y) {var C=x.getAttribute ("id"), t="slickid__"; x.setAttribute ("id",
    t); S="#"+t+" "+S; U=x.parentNode; } try {if (s) {return U.querySelector (S) || null;
    } else {E=U.querySelectorAll (S); } } catch (Q) {g[z]=1; break querySelector; } finally {if (!y) {if (C) {x.setAttribute ("
    id", C); } else {x.removeAttribute ("id"); } U=x; } } if (this.starSelectsClosedQSA) {for (O=0;
    A=E[O++];) {if (A.nodeName>"@&&! (I&&V[this.getUID (A)])) {p.push (A); } } } else {for (O=0; A=E[O++];) {if (! (I&&V[this.
    getUID (A)])) {p.push (A); } } } if (I) {this.sort (p);
    } return p; } F=this.Slick.parse (z); if (!F.length) {return p; } else {if (z==null) {return
    p; } else {if (z.Slick) {F=z; } else {if (this.contains (U.documentElement || U, z)) { (p)?p.push (z):p=z;
    return p; } else {return p; } } } this.posNTH= {}; this.posNTHLast= {}; this.posNTHType= {}; this.posNTHTypeLast= {};
    this.push= (!I&&(s || (F.length==1&&F.expressions[0].length==1)))?this.pushArray:this.
    pushUID;
    if (p==null) {p= []; } var M, L, K; var B, J, D, c, q, G, W; var N, P, o, w, R=F.expressions;
    search: for (O=0; (P=R[O]); O++) {for (M=0; (o=P[M]); M++) {B="combinator:" + o.combinator;
    if (!this[B]) {continue search; } J= (this.isXMLDocument)?o.tag:o.tag.toUpperCase (); D=o.id; c=o.classList; q
    =o.classes; G=o.attributes; W=o.pseudos; w= (M=== (P.length-1));
    this.bitUniques= {}; if (w) {this.uniques=V; this.found=p; } else {this.uniques= {}; this.found= []; } if (M===0) {this[
    B] (U, J, D, q, G, W, c); if (s&&w&&p.length) {break search;
    } } else {if (s&&w) {for (L=0, K=N.length; L<K; L++) {this[B] (N[L], J, D, q, G, W, c); if (p.length) {break
    search; } } } else {for (L=0, K=N.length; L<K; L++) {this[B] (N[L], J, D, q, G, W, c);
    } } } N=this.found; if (I || (F.expressions.length>1)) {this.sort (p); } return (s)?(p[0] ||
    null):p; }

```

9.198.2.142 f Set

Initial value:

```

=new Class ({sets: {}, inherits: {locales: [], sets: {}}, initialize: function (
    g) {this.name=g || "";
    }, define: function (j, h, i) {var g=this.sets[j]; if (!g) {g= {}; } if (h) {if (typeof (h)=="object") {g=
    Object.merge (g, h); } else {g[h]=i; } } this.sets[j]=g; return this; }, get: function (
    s, k, r) {var q=Object.getFromPath (this.sets, s);
    if (q!=null) {var n=typeof (q); if (n=="function") {q=q.apply (null, Array.from (k)); } else {if (n=="
    object") {q=Object.clone (q); } } return q; } var j=s.indexOf ("."); p=j<0?s:s.substr (0, j),
    m= (this.inherits.sets[p] || []). combine (this.inherits.locales). include ("en-US");
    if (!r) {r= []; } for (var h=0, g=m.length; h<g; h++) {if (r.contains (m[h])) {continue; }
    r. include (m[h]); var o=a[m[h]]; if (!o) {continue; } q=o.get (s, k, r); if (q!=null) {return
    q;
    } } return ""; }, inherit: function (h, i) {h=Array.from (h); if (i&&!this.inherits.sets[
    i]) {this.inherits.sets[i]= []; } var g=h.length; while (g--) { (i?this.inherits.sets[
    i]:this.inherits.locales). unshift (h[g]);
    } return this; })

```

9.198.2.143 k setDocument

Initial value:

```

=function (x) {var u=x.nodeType; if (u==9) { } else {if (u) {x=x.ownerDocument; } else {if (x.navigator) {x=x.document; }
    else {return; } } } if (this.document===x) {return;
    } this.document=x; var z=x.documentElement, v=this.getUIDXML (z), p=
    m[v], B; if (p) {for (B in p) {this[B]=p[B]; } return; } p=m[v]= {}; p.root=z; p.isXMLDocument=this.
    isXML (x);
    p.brokenStarGEBTN=p.starSelectsClosedQSA=p.idGetsName=p.brokenMixedCaseQSA=p.brokenGEBCN=
    p.brokenCheckedQSA=p.brokenEmptyAttributeQSA=p.isHTMLDocument=p.nativeMatchesSelector=false;
    var n, o, y, r, s; var t, c="slick_uniqueid"; var A=x.createElement ("div"); var q=x.body || x.
    getElementsByTagName ("body") [0] || z; q.appendChild (A); try {A.innerHTML='<a id="'+c+' "></a>';
    p.isHTMLDocument=!x.getElementById (c); } catch (w) {if (p.isHTMLDocument) {A.style.display="none"; A.
    appendChild (x.createComment ("")); o= (A.getElementsByTagName ("*").length>1);
    try {A.innerHTML="foo</foo>"; t=A.getElementsByTagName ("*"); n= (t&&!t.length&&t[0].nodeName.charAt (0)==" /");
    } catch (w) { (p.brokenStarGEBTN=o || n); try {A.innerHTML='<a name="'+c+' "><b id="'+c+' "></b>';
    p.idGetsName=x.getElementById (c)===A.firstChild; } catch (w) {if (A.getElementsByClassName) {try {A.innerHTML='
    <a class="f"></a><a class="b"></a>'; A.getElementsByClassName ("b").length;
    A.firstChild.className="b"; r= (A.getElementsByClassName ("b").length!=2); } catch (w) {try {A.innerHTML='<a
    class="a"></a><a class="f b a"></a>'; y= (A.getElementsByClassName ("a").length!=2);
    } catch (w) { (p.brokenGEBCN=r || y); if (A.querySelectorAll) {try {A.innerHTML="foo</foo>"; t=A.querySelectorAll ("*
    "); p.starSelectsClosedQSA= (t&&!t.length&&t[0].nodeName.charAt (0)==" /");
    } catch (w) {try {A.innerHTML='<a class="MiX"></a>'; p.brokenMixedCaseQSA=!A.querySelectorAll ("MiX").length;
    } catch (w) {try {A.innerHTML='<select><option selected="selected">a</option></select>';
    p.brokenCheckedQSA= (A.querySelectorAll (":checked").length==0); } catch (w) {try {A.innerHTML='<a
    class=" "></a>'; p.brokenEmptyAttributeQSA= (A.querySelectorAll ('[class=""]').length!=0);
    } catch (w) {try {A.innerHTML='<form action="s"><input id="action"/></form>'; s= (A.firstChild.getAttribute ("
    action")!="s"); } catch (w) {p.nativeMatchesSelector=z.matchesSelector || z.mozMatchesSelector || z.

```



```

        webkitMatchesSelector;
    if (p.nativeMatchesSelector) {try{p.nativeMatchesSelector.call(z, ":slick");p.nativeMatchesSelector=
        null; }catch(w){}} }try{z.slick_expando=1; delete z.slick_expando;
    p.getUID=this.getUIDHTML; }catch(w){p.getUID=this.getUIDXML; }q.removeChild(A); A=t=q=
        null; p.getAttribute=(p.isHTMLDocument&&s)?function(E,C){var F=this.attributeGetters[C];
    if (F) {return F.call(E); }var D=E.getAttributeNode(C); return (D)?D.nodeValue:null; } :function(D,C){var E=
        this.attributeGetters[C]; return (E)?E.call(D):D.getAttribute(C);
    }; p.hasAttribute=(z&&this.isNativeCode(z.hasAttribute))?function(D,C){return D.hasAttribute(C)
        }; :function(D,C){D=D.getAttributeNode(C); return !! (D&&(D.specified||D.nodeValue));
    }; p.contains=(z&&this.isNativeCode(z.contains))?function(C,D){return C.contains(D); } : (z&&z.
        compareDocumentPosition)?function(C,D){return C===D||!! (C.compareDocumentPosition(D)&16);
    }; :function(C,D){if (D){do{if (D===C){return true; } }while ((D=D.parentNode)); }return false; };
    p.documentSorter=(z.compareDocumentPosition)?function(D,C){if (!D.compareDocumentPosition||!C.
        compareDocumentPosition){return 0;
    }return D.compareDocumentPosition(C)&4?-1:D===C?0:1; } : ("sourceIndex" in z)?function(D,C){if (!D.sourceIndex
        ||!C.sourceIndex){return 0; }return D.sourceIndex-C.sourceIndex;
    }; (x.createRange)?function(F,D){if (!F.ownerDocument||!D.ownerDocument){return 0; }var E=F.ownerDocument.
        createRange(), C=D.ownerDocument.createRange(); E.setStart(F,0);
    E.setEnd(F,0); C.setStart(D,0); C.setEnd(D,0); return E.compareBoundaryPoints(Range.START_TO_END,C); } :
        null; z=null; for (B in p){this[B]=p[B]; } }

```

9.198.2.144 Fx Slide

Initial value:

```

=new Class({Extends:Fx, options:{mode:"vertical", wrapper:false, hideOverflow:true, resetHeight:false},
    initialize:function(b,a){b=this.element=this.subject=document.id(
        b);
    this.parent(a); a=this.options; var d=b.retrieve("wrapper"), c=b.getStyles("margin", "position", "overflow"
        ); if (a.hideOverflow){c=Object.append(c, {overflow:"hidden"});
    } if (a.wrapper){d=document.id(a.wrapper).setStyles(c); } if (!d){d=new Element("div", {styles:
        c}).wraps(b); } b.store("wrapper", d).setStyle("margin", 0); if (b.getStyle("overflow")=="visible"){
        b.setStyle("overflow", "hidden");
    } this.now=[]; this.open=true; this.wrapper=d; this.addEvent("complete", function(){this.open=(d["
        offset"+this.layout.capitalize()]!=0); if (this.open&&this.options.resetHeight){d.setStyle("height", "");
    } }, true); }, vertical:function(){this.margin="margin-top"; this.layout="height"; this.offset=this.element.
        offsetHeight; }, horizontal:function(){this.margin="margin-left";
    this.layout="width"; this.offset=this.element.offsetWidth; }, set:function(a){this.element.setStyle(this.
        margin, a[0]); this.wrapper.setStyle(this.layout, a[1]);
    return this; }, compute:function(c,b,a){return [0,1].map(function(d){return Fx.compute(
        c[d], b[d], a); } ), start:function(b,e){if (!this.check(b,e)){return this;
    } this[e]|this.options.mode|(); var d=this.element.getStyle(this.margin).toInt(),
        c=this.wrapper.getStyle(this.layout).toInt(), a=[[d,c], [0, this.offset]], g=[[
        d,c], [-this.offset, 0]], f;
    switch (b){case "in": f=a; break; case "out": f=g; break; case "toggle": f=(c==0)?a:g; } return this.parent(f[0], f[1
        ]); }, slideIn:function(a){return this.start("in", a);
    }, slideOut:function(a){return this.start("out", a); }, hide:function(a){this[a]|this.options.mode|(); this.
        open=false; return this.set([-this.offset, 0]); }, show:function(a){this[a]|this.options.mode|();
    this.open=true; return this.set([0, this.offset]); }, toggle:function(a){return this.start("toggle",
        a); } })

```

9.198.2.145 Element Properties slide

Initial value:

```

={set:function(a){this.get("slide").cancel().setOptions(a);
    return this; }, get:function(){var a=this.retrieve("slide"); if (!a){a=new Fx.Slide(this, {link:"cancel"});
        this.store("slide", a); } return a; } }

```

9.198.2.146 var Slider

Initial value:

```

=new Class({Implements:[Events, Options], Binds:["clickedElement", "draggedKnob", "
    scrolledElement"], options:{onTick:function(a){this.setKnobPosition(a);
    }, initialStep:0, snap:false, offset:0, range:false, wheel:false, steps:100, mode:"horizontal"},
    initialize:function(f,a,e){this.setOptions(e); e=this.options; this.element=
        document.id(f);
    a=this.knob=document.id(a); this.previousChange=this.previousEnd=this.step=-1; var
        b={}; d={x:false, y:false}; switch (e.mode){case "vertical": this.axis="y"; this.property="top";
    this.offset="offsetHeight"; break; case "horizontal": this.axis="x"; this.property="left"; this.offset="
        offsetWidth"; } this.setSliderDimensions(); this.setRange(e.range);

```



```

if (a.getStyle("position")=="static") {a.setStyle("position", "relative");}a.setStyle(this.
    property, -e.offset);d[this.axis]=this.property;b[this.axis]=[-
    e.offset, this.full-e.offset];
var c={snap:0, limit:b, modifiers:d, onDrag:this.draggedKnob, onStart:this.draggedKnob, onBeforeStart:(
    function() {this.isDragging=true;}).bind(this), onCancel:function() {this.isDragging=false;
    }.bind(this), onComplete:function() {this.isDragging=false;this.draggedKnob();this.end();}.bind(this)};if (
    e.snap) {this.setSnap(c);}this.drag=new Drag(a, c);
this.attach();if (e.initialStep!=null) {this.set(e.initialStep);}, attach:function() {this.element.
    addEvent("mousedown", this.clickedElement);if (this.options.wheel) {this.element.addEvent("mousewheel", this.
    scrolledElement);}
this.drag.attach();return this;}, detach:function() {this.element.removeEvent("mousedown", this.
    clickedElement).removeEvent("mousewheel", this.scrolledElement);
this.drag.detach();return this;}, autosize:function() {this.setSliderDimensions().setKnobPosition(this.
    toPosition(this.step));this.drag.options.limit[this.axis]=[-this.options.offset, this.full-this.options.offset];
if (this.options.snap) {this.setSnap();}return this;}, setSnap:function(a) {if (!a) {a=this.drag.options;}.a.grid
    =Math.ceil(this.stepWidth);a.limit[this.axis][1]=this.full;
return this;}, setKnobPosition:function(a) {if (this.options.snap) {a=this.toPosition(this.step);}this.knob.
    setStyle(this.property, a);return this;}, setSliderDimensions:function() {this.full=this.element.measure(
    function() {this.half=this.knob[this.offset]/2;
return this.element[this.offset]-this.knob[this.offset]+(this.options.offset*2);}).bind(this);return this;
    }, set:function(a) {if (!((this.range>0)^(a<this.min))) {a=this.min;
    }if (!((this.range>0)^(a>this.max))) {a=this.max;}this.step=Math.round(a);return this.checkStep().fireEvent("
    tick", this.toPosition(this.step)).end();}, setRange:function(a,b) {this.min=Array.pick([a[0], 0]);
this.max=Array.pick([a[1], this.options.steps]);this.range=this.max-this.min;this.steps=this.options.
    steps||this.full;this.stepSize=Math.abs(this.range)/this.steps;
this.stepWidth=this.stepSize*this.full/Math.abs(this.range);if (a) {this.set(Array.pick([b, this.step])).
    floor(this.min).max(this.max);}return this;}, clickedElement:function(c) {if (this.isDragging||c.target==this.
    knob) {return;
    }var b=this.range<0?-1:1, a=c.page[this.axis]-this.element.getPosition()[this.axis]-this.half; a=a.limit(-
    this.options.offset, this.full-this.options.offset);
this.step=Math.round(this.min+b*this.toStep(a));this.checkStep().fireEvent("tick", a).end();},
    scrolledElement:function(a) {var b=(this.options.mode=="horizontal")?(a.wheel<0):(a.wheel>0);
this.set(this.step+(b?-1:1)*this.stepSize);a.stop();}, draggedKnob:function() {var b=this.range<0?-1:1, a=
    this.drag.value.now[this.axis]; a=a.limit(-this.options.offset, this.full-this.options.offset);
this.step=Math.round(this.min+b*this.toStep(a));this.checkStep();}, checkStep:function() {var a=this.step; if (
    this.previousChange!=a) {this.previousChange=a;
this.fireEvent("change", a);}return this;}, end:function() {var a=this.step; if (this.previousEnd!=a) {this.
    previousEnd=a;this.fireEvent("complete", a+"");}return this;
    }, toStep:function(a) {var b=(a+this.options.offset)*this.stepSize/this.full*this.steps;return this.options.
    steps?Math.round(b-b*this.stepSize):b;}, toPosition:function(a) {return (this.full*Math.abs(this.min-a))/(
    this.steps*this.stepSize)-this.options.offset;
    })
})

```

9.198.2.147 var SmoothScroll

Initial value:

```

=Fx.SmoothScroll=new Class({Extends:Fx.Scroll, options:{axes:["x", "y"]},
    initialize:function(c,d) {d=d||document;
this.doc=d.getDocument();this.parent(this.doc, c);var e=d.getWindow(), a=e.location.href.match(/^[\^#]*/)
    [0]+"#", b=$$(this.options.links||this.doc.links);
b.each(function(g) {if (g.href.indexOf(a)!=0) {return;}var f=g.href.substr(a.length);if (f) {this.useLink
    (g, f);}}, this);this.addEvent("complete", function() {e.location.hash=this.anchor;
this.element.scrollTo(this.to[0], this.to[1]);}, true);}, useLink:function(b,a) {b.addEvent("click", function
    (d) {var c=document.id(a)||this.doc.getElement("a[name="+a+"]");
if (!c) {return;}d.preventDefault();this.toElement(c, this.options.axes).chain(function() {this.fireEvent("
    scrolledTo", [b, c]);}).bind(this);this.anchor=a;}).bind(this);
return this;});

```

9.198.2.148 k sort

Initial value:

```

=function(c) {if (!this.documentSorter) {return c;
    }c.sort(this.documentSorter);return c;}

```

9.198.2.149 Fx Sort

Initial value:

```

=new Class({Extends:Fx.Elements, options:{mode:"vertical"}, initialize:function(
    b,a) {this.parent(b, a);this.elements.each(function(c) {if (c.getStyle("position")=="static") {

```

```

    c.setStyle("position","relative");
  });this.setDefaultOrder();},setDefaultOrder:function(){this.currentOrder=this.elements.map(function(
    b,a){return a;});},sort:function(){if(!this.check(arguments)){return this;
  }var e=Array.flatten(arguments);var i=0,a=0,c={},h={},d=this.options.mode=="vertical";var
    f=this.elements.map(function(m,k){var l=m.getComputedSize({styles:["border","padding","margin"]});
  var n;if(d){n={top:i,margin:l["margin-top"],height:l.totalHeight;i+=n.height-l["margin-top"]};}else{n={
    left:a,margin:l["margin-left"],width:l.totalWidth};
  a+=n.width;}var j=d?"top":"left";h[k]={};var o=m.getStyle(j).toInt();h[k][
    j]=o||0;return n;},this);this.set(h);e=e.map(function(j){return j.toInt();});
  if(e.length!=this.elements.length){this.currentOrder.each(function(j){if(!e.contains(j)){e.push(j);});if(e
    .length>this.elements.length){e.splice(this.elements.length-1,e.length-this.elements.length);
  }var b=0;i=a=0;e.each(function(k){var j={};if(d){j.top=i-f[k].top-b;i+=f[
    k].height;}else{j.left=a-f[k].left;a+=f[k].width;b=b+f[k].margin;c[k]=j;},this);
  var g={};Array.clone(e).sort().each(function(j){g[j]=c[j];});this.start(g);this.currentOrder=
    e;return this;},rearrangeDOM:function(a){a=a||this.currentOrder;
  var b=this.elements[0].getParent();var c=[];this.elements.setStyle("opacity",0);
    a.each(function(d){c.push(this.elements[d].inject(b).setStyles({top:0,left:0}));
  },this);this.elements.setStyle("opacity",1);this.elements=$$(c);this.setDefaultOrder();return this;},
    getDefaultValue:function(){return this.elements.map(function(b,a){return a;
  });},getCurrentOrder:function(){return this.currentOrder;},forward:function(){return this.
    sort(this.getDefaultOrder());},backward:function(){return this.sort(this.getDefaultOrder().reverse(
    ));
  },reverse:function(){return this.sort(this.currentOrder.reverse());},sortByElements:function(
    a){return this.sort(a.map(function(b){return this.elements.indexOf(b);
  },this));},swap:function(c,b){if(typeof(c)=="element"){c=this.elements.indexOf(c);}if(typeof(b)=="element
    ")b=this.elements.indexOf(b);}var a=Array.clone(this.currentOrder);
  a[this.currentOrder.indexOf(c)]=b;a[this.currentOrder.indexOf(b)]=c;return this.
    sort(a);})

```

9.198.2.150 var Sortables

Initial value:

```

=new Class({Implements:[Events,Options],options:{opacity:1,clone:false,revert:false,handle:
  false,dragOptions:{},snap:4,constrain:false,preventDefault:false},initialize:function(
    a,b){this.setOptions(b);
  this.elements=[];this.lists=[];this.idle=true;this.addLists($$(document.id(
    a)||a));if(!this.options.clone){this.options.revert=false;if(this.options.revert){this.effect=new Fx.
    Morph(null,Object.merge({duration:250,link:"cancel"},this.options.revert));
  },attach:function(){this.addLists(this.lists);return this;},detach:function(){this.lists=this.removeLists(
    this.lists);return this;},addItem:function(){Array.flatten(arguments).each(function(
    a){this.elements.push(a);
  var b=a.retrieve("sortables:start",function(c){this.start.call(this,c,a);}).bind(this);this.options.
    handle?a.getElement(this.options.handle)||a:a).addEvent("mousedown",b);
  },this);return this;},addLists:function(){Array.flatten(arguments).each(function(
    a){this.lists.include(a);this.addItem(a.getChildren());},this);return this;
  },removeItems:function(){return $(Array.flatten(arguments).map(function(a){this.elements.erase(
    a);var b=a.retrieve("sortables:start");this.options.handle?a.getElement(this.options.handle)||
    a:a).removeEvent("mousedown",b);
  return a;},this));},removeLists:function(){return $(Array.flatten(arguments).map(function(
    a){this.lists.erase(a);this.removeItem(a.getChildren());return a;
  },this));},getClone:function(b,a){if(!this.options.clone){return new Element(a.tagName).inject(
    document.body);}if(typeof(this.options.clone)=="function"){return this.options.clone.call(this,b,
    a,this.list);
  }var c=a.clone(true).setStyles({margin:0,position:"absolute",visibility:"hidden",width:
    a.getStyle("width")}).addEvent("mousedown",function(d){a.fireEvent("mousedown",
    d);
  });if(c.get("html").test("radio")){c.getElements("input[type=radio]").each(function(
    d,e){d.set("name","clone_"+e);if(d.get("checked")){a.getElements("input[type=radio") [
    e].set("checked",true);
  });}return c.inject(this.list).setPosition(a.getPosition(a.getOffsetParent()));},getDroppables:function(
    ){var a=this.list.getChildren().erase(this.clone).erase(this.element);
  if(!this.options.constrain){a.append(this.lists).erase(this.list);return a;},insert:function(
    c,b){var a="inside";if(this.lists.contains(b)){this.list=b;
  this.drag.droppables=this.getDroppables();}else{a=this.element.getAllPrevious().contains(b)?"before":"after
    ";this.element.inject(b,a);this.fireEvent("sort",[this.element,this.clone]);
  },start:function(b,a){if(!this.idle||b.rightClick||["button","input","a","textarea"].contains(b.target.
    get("tag"))){return;}this.idle=false;this.element=a;
  this.opacity=a.get("opacity");this.list=a.getParent();this.clone=this.getClone(b,a);this.drag=new
    Drag.Move(this.clone,Object.merge({preventDefault:this.options.preventDefault,snap:this.options.s
    nap,container:this.options.constrain&this.element.getParent(),droppables:this.getDroppables(),this.options
    .dragOptions}).addEvents({onSnap:function(){b.stop();
  this.clone.setStyle("visibility","visible");this.element.set("opacity",this.options.opacity||0);this.
    fireEvent("start",[this.element,this.clone]);}.bind(this),onEnter:this.insert.bind(this),onCancel:this.end.bind(
    this),onComplete:this.end.bind(this));
  this.clone.inject(this.element,"before");this.drag.start(b);},end:function(){this.drag.detach();this.
    element.set("opacity",this.opacity);if(this.effect){var b=this.element.getStyles("width","height"),
    d=this.clone,c=d.computePosition(this.element.getPosition(this.clone.getOffsetParent()));
  var a=function(){this.removeEvent("cancel",a);d.destroy();};this.effect.element=
    d;this.effect.start({top:c.top,left:c.left,width:b.width,height:b.height,opacity:0.25}).
    addEvent("cancel",a).chain(a);
  }else{this.clone.destroy();this.reset();},reset:function(){this.idle=true;this.fireEvent("complete",this.

```

```

        element));,serialize:function(){var c=Array.link(arguments,{modifier:Type.isFunction,index:function(d){return d!=null;}});var b=this.lists.map(function(d){return d.getChildren().map(c.modifier||function(e){return e.get("id");},this);},this);var a=c.index;if(this.lists.length==1){a=0;}return(a||a==0)&&a>0&&a<this.lists.length?b[a]:b;}}

```

9.198.2.151 var Spinner

Initial value:

```

=new Class({Extends:Mask,Implements:Chain,options:{class:"spinner",containerPosition:{},content:"class":"spinner-content",messageContainer:{class:"spinner-msg",img:{class:"spinner-img"},fxOptions:{link:"chain"}},initialize:function(c,a){this.target=document.id(c)||document.id(document.body);this.target.store("spinner",this);this.setOptions(a);this.render();this.inject();var b=function(){this.active=false;}.bind(this);this.addEvents({hide:b,show:b});},render:function(){this.parent();this.element.set("id",this.options.id||"spinner-"+String.uniqueID());this.content=document.id(this.options.content)||new Element("div",this.options.content);this.content.inject(this.element);if(this.options.message){this.msg=document.id(this.options.message)||new Element("p",this.options.messageContainer).appendText(this.options.message);this.msg.inject(this.content);}if(this.options.img){this.img=document.id(this.options.img)||new Element("div",this.options.img);this.img.inject(this.content);}this.element.set("tween",this.options.fxOptions);},show:function(a){if(this.active){return this.chain(this.show.bind(this));}if(!this.hidden){this.callChain.delay(20,this);return this;}this.active=true;return this.parent(a);},showMask:function(a){var b=function(){this.content.position(Object.merge({relativeTo:this.element},this.options.containerPosition));}.bind(this);if(a){this.parent();b();}else{if(!this.options.style.opacity){this.options.style.opacity=this.element.getStyle("opacity").toFloat();}this.element.setStyle({display:"block",opacity:0}).tween("opacity",this.options.style.opacity);}b();this.hidden=false;this.fireEvent("show");this.callChain();},hide:function(a){if(this.active){return this.chain(this.hide.bind(this));}if(this.hidden){this.callChain.delay(20,this);return this;}this.active=true;return this.parent(a);},hideMask:function(a){if(a){return this.parent();}this.element.tween("opacity",0).get("tween").chain(function(){this.element.setStyle("display","none");});this.hidden=true;this.fireEvent("hide");this.callChain();}.bind(this));},destroy:function(){this.content.destroy();this.parent();this.target.eliminate("spinner");}}

```

9.198.2.152 Element Properties spinner

Initial value:

```

={set:function(a){var b=this.retrieve("spinner");if(b){b.destroy();}return this.eliminate("spinner").store("spinner:options",a);},get:function(){var a=this.retrieve("spinner");if(!a){a=new Spinner(this,this.retrieve("spinner:options"));this.store("spinner",a);}return a;}}

```

9.198.2.153 a stop

Initial value:

```

=function(f){f.preventDefaultPropagation=true;}

```

9.198.2.154 var t=function(i,w){if(w&&w.\$hidden){return;}var v=this[i];if(v==null||!v.\$protected){this[i]=w;}}

9.198.2.155 timeDiffInWords

9.198.2.156 function this Tips

Initial value:

```

=new Class({Implements:[Events,Options],options:{onShow:function(){this.tip.setStyle("display"
    ,"block");},onHide:function(){this.tip.setStyle("display","none");
},title:"title",text:function(b){return b.get("rel")||b.get("href");},showDelay:100,hideDelay:100,
    className:"tip-wrap",offset:{x:16,y:16},windowPadding:{x:0,y:0},fixed:false,waiAria:true},
    initialize:function(){var b=Array.link(arguments,{options:Type.isObject,elements:function(
    c){return c!=null;
}});this.setOptions(b.options);if(b.elements){this.attach(b.elements);}this.container=new Element("div",{
    class:"tip"});if(this.options.id){this.container.set("id",this.options.id);
if(this.options.waiAria){this.attachWaiAria();}},toElement:function(){if(this.tip){return this.tip;}this.
    tip=new Element("div",{class:this.options.className,styles:{position:"absolute",top:0,left:0}}).
    adopt(new Element("div",{class:"tip-top"}),this.container,new Element("div",{class:"tip-bottom"}));
return this.tip;},attachWaiAria:function(){var b=this.options.id;this.container.set("role","tooltip");if(!
    this.waiAria){this.waiAria={show:function(c){if(b){c.set("aria-describedby",b);
}this.container.set("aria-hidden","false");},hide:function(c){if(b){c.erase("aria-describedby");}this.
    container.set("aria-hidden","true");}};this.addEvents(this.waiAria);
},detachWaiAria:function(){if(this.waiAria){this.container.erase("role");this.container.erase("aria-hidden"
    );this.removeEvents(this.waiAria);},attach:function(b){$(b).each(function(
    d){var f=a(this.options.title,d),e=a(this.options.text,d);
d.set("title","").store("tip:native",f).retrieve("tip:title",f);d.retrieve("tip:text",
    e);this.fireEvent("attach",[d]);var c=["enter","leave"];if(!this.options.fixed){c.push("move");
}c.each(function(h){var g=d.retrieve("tip:"+h);if(!g){g=function(i){this["element"+
    h.capitalize()].apply(this,[i,d]);}.bind(this);}d.store("tip:"+h,g).addEvent("mouse"+
    h,g);
},this);},this);return this;},detach:function(b){$(b).each(function(d){["enter","leave","move"].
    each(function(e){d.removeEvent("mouse"+e,d.retrieve("tip:"+e)).eliminate("tip:"+
    e);
});this.fireEvent("detach",[d]);if(this.options.title=="title"){var c=d.retrieve("tip:native");if(c){
    d.set("title",c);}},this);return this;},elementEnter:function(c,b){clearTimeout(this.timer);
this.timer=function(){this.container.empty();["title","text"].each(function(e){var
    d=b.retrieve("tip:"+e);var f=this["_"+e+"Element"]=new Element("div",{class:"tip-"+
    e}).inject(this.container);
if(d){this.fill(f,d);},this);this.show(b);this.position((this.options.fixed)?{page:b.getPosition()
    };c);}).delay(this.options.showDelay,this);},elementLeave:function(c,b){clearTimeout(this.timer);
this.timer=this.hide.delay(this.options.hideDelay,this,b);this.fireForParent(c,b);},setTitle:function(
    b){if(this._titleElement){this._titleElement.empty();
this.fill(this._titleElement,b);return this;},setText:function(b){if(this._textElement){this._textElement
    .empty();this.fill(this._textElement,b);return this;
},fireForParent:function(c,b){b=b.getParent();if(!b||b==document.body){return;}if(b.retrieve("
    tip:enter")){b.fireEvent("mouseenter",c);}else{this.fireForParent(c,b);
}},elementMove:function(c,b){this.position(c);},position:function(
    f){if(!this.tip){document.id(this);}var c=window.getSize(),b=
    window.getScroll(),g={x:this.tip.offsetWidth,y:this.tip.offsetHeight,d={x:"left",y:"top"},
    e={y:false,x2:false,y2:false,x:false},h={};
for(var i in d){h[d[i]]=f.page[i]+this.options.offset[i];if(h[d[i]]<0){e
    i=true;if((h[d[i]]+g[i]-b[i])>c[i]-this.options.windowPadding[i]){h[d[i]]=f.page[
    i]-this.options.offset[i]-g[i];
e[i+"2"]=true;}}this.fireEvent("bound",e);this.tip.setStyles(h);},fill:function(
    b,c){if(typeof c=="string"){b.set("html",c);}else{b.adopt(c);},show:function(
    b){if(!this.tip){document.id(this);
}if(!this.tip.getParent()){this.tip.inject(document.body);}this.fireEvent("show",[this.tip,b]);
hide:function(b){if(!this.tip){document.id(this);}this.fireEvent("hide",[this.tip,b]);
}})

```

9.198.2.157 a trace

Initial value:

```

=function(f){f=f||a.manager;var g=window.console&&console.log;if(g){console.log("the following
    items have focus: ");
}a.each(f,function(h){if(g){console.log(document.id(h.wiget)||h.wiget||
    h);}});}

```

9.198.2.158 try(n(1.html.childNodes)

9.198.2.159 Array type =k.isObject

9.198.2.160 var u

Initial value:

```

=this.instanceOf=function(w,i){if(w==null){return false;}var v=w.$constructor||
    w.constructor;
while(v){if(v===i){return true;}v=v.parent;}return w instanceof i;}

```

9.198.2.161 k uidk="slick-uniqueid"

9.198.2.162 k uidx=1

9.198.2.163 URI

Initial value:

```
=Class.refactor(URI,{combine:function(f,e){if(!e||f.scheme!=e.scheme||f.host!=e.host||f.port!=e.port){return this.previous.apply(this,arguments);}var a=f.file+(f.query?"?"+f.query:"")+ (f.fragment?"#"+f.fragment:"");if(!e.directory){return(f.directory||(f.file?"":"/")+a);}var d=e.directory.split("/"),c=f.directory.split("/"),g="",h;var b=0;for(h=0;h<d.length&&h<c.length&&d[h]==c[h];h++){for(b=0;b<d.length-h-1;b++){g+="./";}for(b=h;b<c.length-1;b++){g+=c[b]+"/";}return(g||(f.file?"":"/")+a);},toAbsolute:function(a){a=new URI(a);if(a){a.set("directory","").set("file","");return this.toRelative(a);},toRelative:function(a){return this.get("value",new URI(a));}}})
```

9.198.2.164 Form Validator

Initial value:

```
=new Class({Implements:[Options,Events],Binds:["onSubmit"],options:{fieldSelectors:"input, select, textarea",ignoreHidden:true,ignoreDisabled:true,useTitles:false,evaluateOnSubmit:true,evaluateFieldsOnBlur:true,evaluateFieldsOnChange:true,serial:true,stopOnFailure:true,warningPrefix:function(){return Form.Validator.getMsg("warningPrefix")}||"Warning: ";},errorPrefix:function(){return Form.Validator.getMsg("errorPrefix")}||"Error: ";},initialize:function(b,a){this.setOptions(a);this.element=document.id(b);this.element.store("validator",this);this.warningPrefix=Function.from(this.options.warningPrefix)();this.errorPrefix=Function.from(this.options.errorPrefix)();if(this.options.evaluateOnSubmit){this.element.addEvent("submit",this.onSubmit);}if(this.options.evaluateFieldsOnBlur||this.options.evaluateFieldsOnChange){this.watchFields(this.getFields());},toElement:function(){return this.element;},getFields:function(){return(this.fields=this.element.getElements(this.options.fieldSelectors));},watchFields:function(a){a.each(function(b){if(this.options.evaluateFieldsOnBlur){b.addEvent("blur",this.validationMonitor.pass([b,false],this));}if(this.options.evaluateFieldsOnChange){b.addEvent("change",this.validationMonitor.pass([b,true],this));},this);},validationMonitor:function(){clearTimeout(this.timer);this.timer=this.validateField.delay(50,this,arguments);},onSubmit:function(a){if(this.validate(a)){this.reset();},reset:function(){this.getFields().each(this.resetField,this);return this;},validate:function(b){var a=this.getFields().map(function(c){return this.validateField(c,true);},this).every(function(c){return c;});this.fireEvent("formValidate",[a,this.element,b]);if(this.options.stopOnFailure&&!a&&b){b.preventDefault();}return a;},validateField:function(j,b){if(this.paused){return true;}j=document.id(j);var f=!j.hasClass("validation-failed");var g,i;if(this.options.serial&&!b){g=this.element.getElement(".validation-failed");i=this.element.getElement(".warning");}if(j&&(!g||b||j.hasClass("validation-failed"))||(g&&!this.options.serial)){var a=j.get("validators");var d=a.some(function(k){return this.getValidator(k);},this);var h=[];a.each(function(k){if(k&&!this.test(k,j)){h.include(k);},this);f=h.length===0;if(d&&!this.hasValidator(j,"warnOnly")){if(f){j.addClass("validation-passed").removeClass("validation-failed");this.fireEvent("elementPass",[j]);}else{j.addClass("validation-failed").removeClass("validation-passed");this.fireEvent("elementFail",[j,h]);}if(!i){var e=a.some(function(k){if(k.test("^warn-","")){return this.getValidator(k.replace(/^warn-/,""));}}else{return null;},this);j.removeClass("warning");var c=a.map(function(k){if(k.test("^warn-","")){return this.test(k.replace(/^warn-/,""),j,true);}else{return null;},this);}}return f;},test:function(b,d,e){d=document.id(d);if((this.options.ignoreHidden&&!d.isVisible())|| (this.options.ignoreDisabled&&d.get("disabled"))){return true;}var a=this.getValidator(b);if(e!=null){e=false;if(this.hasValidator(d,"warnOnly")){e=true;}var c=this.hasValidator(d,"ignoreValidation")|| (a?a.test(d):true);if(a&&d.isVisible()){this.fireEvent("elementValidate",[c,d,b,e]);}if(e){return true;}return c;},hasValidator:function(b,a){return b.get("validators").contains(a);},resetField:function(a){a=document.id(a);if(a){a.get("validators").each(function(b){if(b.test("^warn-","")){b=b.replace(/^warn-/,"");a.removeClass("validation-failed");a.removeClass("warning");a.removeClass("validation-passed");},this);}return this;},stop:function(){this.paused=true;return this;},start:function(){this.paused=false;return this;},ignoreField:function(a,b){a=document.id(a);if(a){this.enforceField(a);if(b){a.addClass("warnOnly");}else{a.addClass("ignoreValidation");}}return this;},enforceField:function(a){a=document.id(a);if(a){a.removeClass("warnOnly").removeClass("ignoreValidation");}return this;}}})
```

9.198.2.165 Element Properties validator

Initial value:

```
=(set:function(a){this.get("validator").setOptions(a);},get:function(){var a=this.retrieve("validator");
    if(!a){a=new Form.Validator(this);
    this.store("validator",a);}return a;})
```

9.198.2.166 Element Properties validatorProps

Initial value:

```
=(set:function(a){return this.eliminate("$moo:validatorProps").store("$moo:validatorProps",
    a);
},get:function(a){if(a){this.set(a);}if(this.retrieve("$moo:validatorProps")){return this.retrieve("
    $moo:validatorProps");}if(this.getProperty("data-validator-properties")||this.getProperty("validatorProps")){try
    {this.store("$moo:validatorProps",JSON.decode(this.getProperty("validatorProps")||this.getProperty("
    data-validator-properties")));}
}catch(c){return{}};}else{var b=this.get("validators").filter(function(d){return
    d.test(":");});if(!b.length){this.store("$moo:validatorProps",{});}else{a={};
    b.each(function(d){var f=d.split(":");if(f[1]){try{a[f[0]]=JSON.decode(f[1]);}catch(
    g){}}});this.store("$moo:validatorProps",a);}return this.retrieve("$moo:validatorProps");
})
```

9.198.2.167 Element Properties validators =`{get:function(){return(this.get("data-validators")||this.className).clean().split("
");}}`

9.198.2.168 this Window =`this.$constructor=new Type("Window",function({})`

9.198.2.169 window

9.198.2.170 Cookie write =`function(b,c,a){return new Cookie(b,a).write(c);}`

9.198.2.171 p Features xhr =`!(p.Request)`

9.199 lib/js/respond.js File Reference

Variables

- function `w`
- var `respond` = {}
- `respond update` = `function({})`
- var `requestQueue` = []
- var `xmlHttp`
- var `ajax`
- var `isUnsupportedMediaQuery`
- `respond queue` = `requestQueue`
- `respond unsupportedmq` = `isUnsupportedMediaQuery`

9.199.1 Variable Documentation

9.199.1.1 respond ajax

Initial value:

```

= function( url, callback ) {
    var req = xmlhttp();
    if (!req){
        return;
    }
    req.open( "GET", url, true );
    req.onreadystatechange = function () {
        if ( req.readyState !== 4 || req.status !== 200 && req.status !== 304 ){
            return;
        }
        callback( req.responseText );
    };
    if ( req.readyState === 4 ){
        return;
    }
    req.send( null );
}

```

9.199.1.2 var isUnsupportedMediaQuery

Initial value:

```

= function( query ) {
    return query.replace( respond.regex.minmaxwh, '' ).match(
        respond.regex.other );
}

```

9.199.1.3 respond queue = requestQueue

9.199.1.4 var requestQueue = []

9.199.1.5 w respond = {}

9.199.1.6 respond unsupportedmq = isUnsupportedMediaQuery

9.199.1.7 respond update = function({})

9.199.1.8 function w

Initial value:

```

{
    "use strict"
}

```

9.199.1.9 var xmlhttp

Initial value:

```

= (function() {
    var xmlhttpmethod = false;
    try {
        xmlhttpmethod = new w.XMLHttpRequest();
    }
    catch( e ){
        xmlhttpmethod = new w.ActiveXObject( "Microsoft.XMLHTTP" );
    }
    return function() {
        return xmlhttpmethod;
    };
})();

```


9.200 lib/js/slimbox.js File Reference

Functions

- [window addEvent](#) ("domready", Slimbox.scanPage)

Variables

- var [Slimbox](#) =(function(){var F=[window](#),n=Browser.Engine.trident4,u,g,G=-1,o,w,E,v,y,M,s,m={},t=new Image(),K=new Image(),l,a,h,q,J,e,H,c,A,L,x,i,d,C;F.addEvent("domready",function(){\$(document.body).adopt(\$\$(l=new Element("div",{id:"lbOverlay",events:{click:D}}),a=new Element("div",{id:"lbCenter"}),H=new Element("div",{id:"lbBottomContainer"})).setStyle("display","none");h=new Element("div",{id:"lbImage"}).injectInside(a).adopt(q=new Element("div",{styles:{position:"relative"}}).adopt(J=new Element("a",{id:"lbPrevLink",href:"#",events:{click:B}}),e=new Element("a",{id:"lbNextLink",href:"#",events:{click:f}})),c=new Element("div",{id:"lbBottom"}).injectInside(H).adopt(new Element("a",{id:"lbCloseLink",href:"#",events:{click:D}}),A=new Element("div",{id:"lbCaption"}),L=new Element("div",{id:"lbNumber"}),new Element("div",{styles:{clear:"both"}}));function z(){var N=F.getScroll(),O=F.getSize();\$\$a,H).setStyle("left",N.x+(O.x/2));if(v){l.setStyle({left:N.x,top:N.y,width:O.x,height:O.y})}function l(N){["object",n?"select":"embed"].forEach(function(P){Array.forEach(document.getElementsByTagName(P),function(Q){if(N){Q._slimbox=Q.style.visibility;Q.style.visibility=N?"hidden":Q._slimbox});l.style.display=N?"":"none";var O=N?"addEvent":"removeEvent";F[O]("scroll",z)[O]("resize",z);document[O]("keyp(O){var N=O.code;return u.closeKeys.contains(N)?D():u.nextKeys.contains(N)?f():u.previousKeys.contains(N)?B():false}function B(){return b(w)}function f(){return b(E)}function b(N){if(N>=0){G=N;o=g[N][0];w=(G|(u.loop?g.length:0))-1;E=((G+1)%g.length)|(u.loop?0:-1);r();a.className="lbLoading";m=new Image();m.onload=k;m.src=o}return false}function k(){a.className="";d.set(0);h.setStyle({backgroundImage:"url('"+o+"')",display:"");q.setStyle("width",m.width);\$\$q,J,e).setStyle("height",m.height);A.set("html",g[G][1]||"");L.set("html",(((g.length>1)&&u.counterText)||"").replace(/x/,G+1).replace(/y/,g.length));if(w>=0){t.src=g[w][0];if(E>=0){K.src=g[E][0];M=h.offsetWidth;s=h.offsetHeight;var P=Math.max(0,y-(s/2)),N=0,O;if(a.offsetHeight!=s){N=i.start({height:s,top:P});if(a.offsetWidth!=M){N=i.start({width:M,marginLeft:-M/2})}O=function(){H.setStyle({width:M,top:P+s,marginLeft:-M/2,visibility:"hidden",display:"");d.start(1);if(N){i.chain(O)}else{O()}}function j(){if(w>=0){J.style.display=""}}if(E>=0){e.style.display=""}}C.set(-c.offsetHeight).start(0);H.style.visibility=""}}function r(){m.onload=\$empty;m.src=t.src=K.src=o;i.cancel();d.cancel();C.cancel();\$\$J,e,h,H).setStyle("display","none");function D(){if(G>=0){r();G=w=E=-1;a.style.display="none";x.cancel().chain(l).start(0)}return false}Element.implement({slimbox:function(N,O){\$(this).slimbox(N,O);return this}});Elements.implement({slimbox:function(N,Q,P){Q=Q||function(R){return[R.href,R.title]};P=P||function(){return true};var O=this;O.removeEvents("click").addEvent("click",function(){var R=O.filter(P,this);return Slimbox.open(R.map(Q),R.indexOf(this),N)});return O});return{open:function(P,O,N){u=\$extend({loop:false,overlayOpacity:0.8,overlayFadeDuration:400,resizeDuration:400,resizeTransition:false,initialWidth:250,initialHeight:250,imageFadeDuration:400,captionAnimationDuration:400,counterText:"Image {x} of {y}",closeKeys:[27,88,67],previousKeys:[37,80],nextKeys:[39,78],N||{});x=new Fx.Tween(l,{property:"opacity",duration:u.overlayFadeDuration});i=new Fx.Morph(a,\$extend({duration:u.resizeDuration,link:"chain"},u.resizeTransition?{transition:u.resizeTransition}:{});d=new Fx.Tween(h,{property:"opacity",duration:u.imageFadeDuration,onComplete:j});C=new Fx.Tween(c,{property:"margin-top",duration:u.captionAnimationDuration});if(typeof P=="string"){P=[P,O];O=0}y=F.getScrollTop()+(F.getHeight()/2);M=u.initialWidth;s=u.initialHeight;a.setStyle({top:Math.max(0,y-(s/2)),width:M,height:s,marginLeft:-M/2,display:"");v=n||(l.currentStyle&&(l.currentStyle.position!="fixed"));if(v){l.style.position="absolute"}x.set(0).start(u.overlayOpacity);z();l(1);g=P;u.loop=u.loop&&(g.length>1);return b(O)}}}()
- [Slimbox scanPage](#)

9.200.1 Function Documentation

9.200.1.1 window addEvent ("domready" , Slimbox. scanPage)

9.200.2 Variable Documentation

9.200.2.1 Slimbox scanPage

Initial value:

```

= function() {
    $(document.links).filter(function(el) {
        return el.rel && el.rel.test(/^lightbox/i);
    }).slimbox({}, null, function(el) {
        return (this == el) || ((this.rel.length > 8) && (this.rel == el.rel));
    });
}

```

```

9.200.2.2 var Slimbox =(function(){var F=window,n=Browser.Engine.trident4,u,g,G=-1,o,w,E,v,y,M,s,m={},t=new
Image(),K=new Image(),l,a,h,q,j,e,H,c,A,L,x,i,d,C;F.addEvent("domready",function(){$(document.
body).adopt($(l=new Element("div",{id:"lbOverlay",events:{click:D}}),a=new Element("div",{id:"lbCenter"}),H=new
Element("div",{id:"lbBottomContainer"})).setStyle("display","none");h=new Element("div",{id:
:"lbImage"}).injectInside(a).adopt(q=new Element("div",{styles:{position:"relative"}}).adopt(J=new
Element("a",{id:"lbPrevLink",href:"#",events:{click:B}}),e=new Element("a",{id:"lbNext
Link",href:"#",events:{click:f}}));c=new Element("div",{id:"lbBottom"}).injectInside(H).adopt(new
Element("a",{id:"lbCloseLink",href:"#",events:{click:D}}),A=new Element("div",{id:"lbCaption"}),L=new
Element("div",{id:"lbNumber"}),new Element("div",{styles:{clear:"both"}}));function z(){var N=F.getScroll(),O=F.
getSize();$(a,H).setStyle("left",N.x+(O.x/2));if(v){l.setStyle({left:N.x,top:N.y,width:O.x,height:O.y})}function
l(N){["object","n?"+"select":"embed"].forEach(function(P){Array.forEach(document.
getElementsByName(P),function(Q){if(N){Q._slimbox=Q.style.visibility}Q.style.
visibility=N?"hidden":Q._slimbox});l.style.display=N?"":"none";var O=N?"addEvent":"remove
Event";F(O)("scroll",z)[O]("resize",z);document[O]("keydown",p)}function p(O){var N=O.code;return
u.closeKeys.contains(N)?D():u.nextKeys.contains(N)?f():u.previousKeys.contains(N)?B():false}function
B(){return b(w)}function f(){return b(E)}function b(N){if(N>=0){G=N;o=g[N][0];w=(G|(u.
loop?g.length:0))-1;E=((G+1)%g.length)|(u.loop?0:-1);r();a.className="lbLoading";m=new
Image();m.onload=k;m.src=o}return false}function k(){a.className="";d.set(0);h.setStyles({background:
Image:"url("+o+")",display:""});q.setStyle("width",m.width);$(q,j,e).setStyle("height",m.height);A.
set("html",g[G][1]|| "");L.set("html",((g.length>1)&&u.counterText)|| "").replace(/x/,G+1).replace(/y/,g.
length));if(w>=0){t.src=g[w][0];if(E>=0){K.src=g[E][0];M=h.offsetWidth;s=h.offsetHeight;var
P=Math.max(0,y-(s/2)),N=0,O;if(a.offsetHeight!=s){N=i.start({height:s,top:P});if(a.offset
Width!=M){N=i.start({width:M;marginLeft:-M/2})}O=function(){H.setStyles({width:M,top:
P+s;marginLeft:-M/2,visibility:"hidden",display:""});d.start(1);if(N){i.chain(O)}else{O()}}function
j(){if(w>=0){J.style.display=""}if(E>=0){e.style.display=""}C.set(-c.offsetHeight).start(0);H.style.visibility=""}}function
r(){m.onload=$empty;m.src=t.src=K.src=o;i.cancel();d.cancel();C.cancel();$(J,e,h,H).set
Style("display","none")}function D(){if(G>=0){r();G=w=E=-1;a.style.display="none";x.cancel().chain(l).start(0)}return
false}Element.implement({slimbox:function(N,O){$(this).slimbox(N,O);return this}});Elements.
implement({slimbox:function(N,Q,P){Q=Q||function(R){return[R.href,R.title];P=P||function(){return
true};var O=this;O.removeEvents("click").addEvent("click",function(){var R=O.filter(P,this);return
Slimbox.open(R.map(Q),R.indexOf(this,N));return O});return {open:function(P,O,N){u=$extend({loop:
false,overlayOpacity:0.8,overlayFadeDuration:400,resizeDuration:400,resizeTransition:false,initial
Width:250,initialHeight:250,imageFadeDuration:400,captionAnimationDuration:400,counterText:"Image
{x} of {y}",closeKeys:[27,88,67],previousKeys:[37,80],nextKeys:[39,78],N||{};x=new
Fx.Tween(l,{property:"opacity",duration:u.overlayFadeDuration});i=new Fx.Morph(a,$extend({duration:
u.resizeDuration,link:"chain"},u.resizeTransition?{transition:u.resizeTransition}:{}));d=new
Fx.Tween(h,{property:"opacity",duration:u.imageFadeDuration,onComplete:j});C=new Fx.Tween(c,{property:
:"margin-top",duration:u.captionAnimationDuration});if(typeof P=="string"){P=[P,O];O=0}y=F.getScroll
Top()+(F.getHeight()/2);M=u.initialWidth;s=u.initialHeight;a.setStyles({top:Math.max(0,y-(s/2)),width:M,height:
s;marginLeft:-M/2,display:""});v=n||(l.currentStyle&&(l.currentStyle.position!="fixed"));if(v){l.style.
position="absolute"}x.set(0).start(u.overlayOpacity);z();l(1);g=P;u.loop=u.loop&&(g.length>1);return
b(O)}}})();

```

9.201 lib/plugin/plugin.mysql.php File Reference

Contiene la classe mysql.

Data Structures

- class [mysql](#)

Libreria di connessione ai database MySQL.

Namespaces

- [Gino\Plugin](#)

Namespace che comprende classi di tipo plugin

9.201.1 Detailed Description

Contiene la classe mysql.

Copyright

2005-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.202 lib/plugin/plugin.pdo.php File Reference

Contiene la classe pdo.

Data Structures

- class [pdo](#)

Libreria di connessione ai database.

Namespaces

- [Gino\Plugin](#)

Namespace che comprende classi di tipo plugin

9.202.1 Detailed Description

Contiene la classe pdo.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.203 lib/plugin/plugin.pdo_mysql.php File Reference

Contiene la classe pdo_mysql.

Data Structures

- class [pdo_mysql](#)

Driver specifico per la connessione a un database MYSQL attraverso la libreria PDO.

Namespaces

- [Gino\Plugin](#)

Namespace che comprende classi di tipo plugin

9.203.1 Detailed Description

Contiene la classe pdo_mysql.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.204 lib/plugin/plugin.pdo_sqlsrv.php File Reference

Contiene la classe pdo_sqlsrv.

Data Structures

- class [pdo_sqlsrv](#)

Driver specifico per la connessione a un database SQL Server attraverso la libreria PDO.

Namespaces

- [Gino\Plugin](#)

Namespace che comprende classi di tipo plugin

9.204.1 Detailed Description

Contiene la classe pdo_sqlsrv.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.205 lib/plugin/plugin.phpfastcache.php File Reference

Contiene la classe plugin_phpfastcache.

Data Structures

- class [plugin_phpfastcache](#)
Interfaccia alla libreria phpfastcache.

Namespaces

- [Gino\Plugin](#)
Namespace che comprende classi di tipo plugin

9.205.1 Detailed Description

Contiene la classe plugin_phpfastcache.

Copyright

2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.206 lib/plugin/plugin.sqlsrv.php File Reference

Contiene la classe sqlsrv.

Data Structures

- class [sqlsrv](#)
Libreria di connessione ai database SQL Server.

Namespaces

- [Gino\Plugin](#)
Namespace che comprende classi di tipo plugin

9.206.1 Detailed Description

Contiene la classe sqlsrv.

Copyright

2013-2015 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.207 lib/recaptchalib.php File Reference

Contiene la libreria che gestisce la chiamata del reCaptcha.

Data Structures

- class [ReCaptchaResponse](#)

A [ReCaptchaResponse](#) is returned from [recaptcha_check_answer\(\)](#)

Functions

- [_recaptcha_qsencode](#) (\$data)
Encodes the given data into a query string format.
- [_recaptcha_http_post](#) (\$host, \$path, \$data, \$port=80)
Submits an HTTP POST to a reCAPTCHA server.
- [recaptcha_get_html](#) (\$pubkey, \$error=null, \$use_ssl=false)
Gets the challenge HTML (javascript and non-javascript version).
- [recaptcha_check_answer](#) (\$privkey, \$remoteip, \$challenge, \$response, \$extra_params=array())
Calls an HTTP POST function to verify if the user's guess was correct.
- [recaptcha_get_signup_url](#) (\$domain=null, \$appname=null)
gets a URL where the user can sign up for reCAPTCHA.
- [_recaptcha_aes_pad](#) (\$val)
- [_recaptcha_aes_encrypt](#) (\$val, \$ky)
- [_recaptcha_mailhide_urlbase64](#) (\$x)
- [recaptcha_mailhide_url](#) (\$pubkey, \$privkey, \$email)
- [_recaptcha_mailhide_email_parts](#) (\$email)
gets the parts of the email to expose to the user.
- [recaptcha_mailhide_html](#) (\$pubkey, \$privkey, \$email)
Gets html to display an email address given a public an private key.

Variables

- const [RECAPTCHA_API_SERVER](#) "http://www.google.com/recaptcha/api"
The reCAPTCHA server URL's.
- const [RECAPTCHA_API_SECURE_SERVER](#) "https://www.google.com/recaptcha/api"
- const [RECAPTCHA_VERIFY_SERVER](#) "www.google.com"

9.207.1 Detailed Description

Contiene la libreria che gestisce la chiamata del reCaptcha.

9.207.2 Function Documentation

9.207.2.1 `_recaptcha_aes_encrypt ($val, $ky)`

9.207.2.2 `_recaptcha_aes_pad ($val)`

9.207.2.3 `_recaptcha_http_post ($host, $path, $data, $port = 80)`

Submits an HTTP POST to a reCAPTCHA server.

Parameters

string	<i>\$host</i>	
string	<i>\$path</i>	
array	<i>\$data</i>	
	<i>int</i>	port

Returns

array response

9.207.2.4 `_recaptcha_mailhide_email_parts ($email)`

gets the parts of the email to expose to the user.

eg, given johndoe(at)example.com return ["john", "example.com"]. the email is then displayed as john...(at)example.↵com

9.207.2.5 `_recaptcha_mailhide_urlbase64 ($x)`9.207.2.6 `_recaptcha_qsencode ($data)`

Encodes the given data into a query string format.

Parameters

<i>\$data</i>	- array of string elements to be encoded
---------------	--

Returns

string - encoded request

9.207.2.7 `recaptcha_check_answer ($privkey, $remoteip, $challenge, $response, $extra_params = array())`

Calls an HTTP POST function to verify if the user's guess was correct.

Parameters

string	<i>\$privkey</i>	
string	<i>\$remoteip</i>	
string	<i>\$challenge</i>	
string	<i>\$response</i>	
array	<i>\$extra_params</i>	an array of extra variables to post to the server

Returns

[ReCaptchaResponse](#)

9.207.2.8 `recaptcha_get_html ($pubkey, $error = null, $use_ssl = false)`

Gets the challenge HTML (javascript and non-javascript version).

This is called from the browser, and the resulting reCAPTCHA HTML widget is embedded within the HTML form it was called from.

Parameters

string	<i>\$pubkey</i>	A public key for reCAPTCHA
string	<i>\$error</i>	The error given by reCAPTCHA (optional, default is null)
boolean	<i>\$use_ssl</i>	Should the request be made over ssl? (optional, default is false)

Returns

string - The HTML to be embedded in the user's form.

9.207.2.9 `recaptcha_get_signup_url ($domain = null, $appname = null)`

gets a URL where the user can sign up for reCAPTCHA.

If your application has a configuration page where you enter a key, you should provide a link using this function.

Parameters

string	<i>\$domain</i>	The domain where the page is hosted
string	<i>\$appname</i>	The name of your application

9.207.2.10 `recaptcha_mailhide_html ($pubkey, $privkey, $email)`

Gets html to display an email address given a public an private key.

to get a key, go to:

<http://www.google.com/recaptcha/mailhide/apikey>

9.207.2.11 `recaptcha_mailhide_url ($pubkey, $privkey, $email)`

9.207.3 Variable Documentation

9.207.3.1 `const RECAPTCHA_API_SECURE_SERVER "https://www.google.com/recaptcha/api"`

9.207.3.2 `const RECAPTCHA_API_SERVER "http://www.google.com/recaptcha/api"`

The reCAPTCHA server URL's.

9.207.3.3 `const RECAPTCHA_VERIFY_SERVER "www.google.com"`

9.208 `lib/sqlcode.php` File Reference

Contiene la classe sqlcode.

Data Structures

- class `sqlcode`
Contiene query e codice sql personalizzato.

Namespaces

- `Gino`
Namespace principale di gino.

9.208.1 Detailed Description

Contiene la classe sqlcode.

Copyright

2013 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.209 README.md File Reference

9.210 settings.php File Reference

Definizione dei percorsi dell'applicazione.

Namespaces

- [Gino](#)

Namespace principale di gino.

Variables

- const [OS](#) DIRECTORY_SEPARATOR
Separatore di directory definito dal sistema operativo.
- const [CACHE_DIR](#) SITE_ROOT.OS.'cache'
Percorso assoluto alla directory cache.
- const [LIB_DIR](#) SITE_ROOT.OS.'lib'
Percorso assoluto alla directory lib.
- const [CLASSES_DIR](#) LIB_DIR.OS.'classes'
Percorso assoluto alla directory che contiene le classi che non fanno parte di moduli.
- const [FIELDS_DIR](#) CLASSES_DIR.OS.'fields'
Percorso assoluto alla directory che contiene le classi che descrivono campi di db.
- const [APP_DIR](#) SITE_ROOT.OS.'app'
Percorso assoluto alla directory che contiene le directory dei moduli.
- const [PLUGIN_DIR](#) LIB_DIR.OS.'plugin'
Percorso assoluto alla directory che contiene plugins.
- const [CSS_DIR](#) SITE_ROOT.OS.'css'
Percorso assoluto alla directory che contiene css.
- const [TPL_DIR](#) SITE_ROOT.OS.'templates'
Percorso assoluto alla directory che contiene templates.
- const [VIEWS_DIR](#) SITE_ROOT.OS.'views'
Percorso assoluto alla directory che contiene viste generiche di sistema.
- const [GRAPHICS_DIR](#) SITE_ROOT.OS.'graphics'
Percorso assoluto alla directory che contiene file di grafica.
- const [CONTENT_DIR](#) SITE_ROOT.OS.'contents'
Percorso assoluto alla directory che contiene upload degli utenti.

- const **FONTS_DIR** SITE_ROOT.OS.'fonts'
Percorso assoluto alla directory che contiene fonts files.
- const **TMP_DIR** '/tmp'
Percorso assoluto alla directory per la creazione di file temporanei.
- const **HOME_FILE** SITE_WWW.'/\$home_file'
Percorso relativo dello SCRIPT_FILE.
- const **CSS_WWW** SITE_WWW.'/css'
Percorso relativo alla directory dei css.
- const **SITE_APP** SITE_WWW.'/app'
Percorso relativo alla directory che contiene i moduli.
- const **SITE_IMG** SITE_WWW.'/img'
Percorso relativo alla directory che contiene immagini si sistema.
- const **SITE_GRAPHICS** SITE_WWW.'/graphics'
Percorso relativo alla directory che contiene file di grafica.
- const **SITE_LIB** SITE_WWW.'/lib'
Percorso relativo alla directory lib.
- const **SITE_JS** SITE_LIB.'/js'
Percorso relativo alla directory che contiene librerie javascript.
- const **SITE_CUSTOM_CKEDITOR** SITE_LIB.'/custom_ckeditor'
Percorso relativo alla directory che contiene file di customizzazione ckeditor.
- const **CONTENT_WWW** SITE_WWW.'/contents'
Percorso relativo alla directory che contiene file uploadati dall'utente.
- const **CONTROLLER_CLASS_PREFIX** 'class_'
Prefisso classi di tipo [Gino.Controller](#).
- const **MODEL_CLASS_PREFIX** 'class.'
Prefisso classi di tipo [Gino.Model](#).
- const **CORE_CLASS_PREFIX** 'class.'
Prefisso classi non interne a moduli.

9.210.1 Detailed Description

Definizione dei percorsi dell'applicazione.

Copyright

2005-20014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
 abidibo abidibo@gmail.com

9.211 templates/_popup.php File Reference

9.212 templates/admin_home.php File Reference

9.213 templates/admin_page.php File Reference

9.214 templates/default_free_tpl.php File Reference

9.215 templates/home.php File Reference

9.216 templates/home_mobile.php File Reference

9.217 templates/page.php File Reference

9.218 templates/pages_mobile.php File Reference

9.219 test/class.Paginator_test.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Test.PaginatorTest](#).

Data Structures

- class [PaginatorTest](#)
Classe di tipo `PHPUnit_Framework_TestCase` per testare la classe [Gino.Paginator](#).

Namespaces

- [Gino\Test](#)
Namespace che comprende gli UNIT TEST di classi e funzioni di gino

Functions

- [setPage](#) (\$page)

9.219.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Test.PaginatorTest](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.220 test/func_test.php File Reference

Contiene la definizione ed implementazione della classe [Gino.Test.FuncTest](#).

Data Structures

- class [FuncTest](#)
Classe di tipo `PHPUnit_Framework_TestCase` per testare il file [func.php](#).

Namespaces

- [Gino\Test](#)

Namespace che comprende gli UNIT TEST di classi e funzioni di gino

9.220.1 Detailed Description

Contiene la definizione ed implementazione della classe [Gino.Test.FuncTest](#).

Copyright

2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.221 test/include.php File Reference

Include classi e funzioni per l'esecuzione di unit tests.

Variables

- const [SITE_ROOT](#) realpath(dirname(__FILE__).DIRECTORY_SEPARATOR.'..')
- const [SITE_WWW](#) "

9.221.1 Detailed Description

Include classi e funzioni per l'esecuzione di unit tests.

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Author

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.221.2 Variable Documentation

9.221.2.1 const [SITE_ROOT](#) realpath(dirname(__FILE__).DIRECTORY_SEPARATOR.'..')

9.221.2.2 const [SITE_WWW](#) "

9.222 views/403.php File Reference

Template della pagina 403, Forbidden.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.222.1 Detailed Description

Template della pagina 403, Forbidden.

Variabili disponibili:

- **title**: string, titolo
- **message**: string, messaggio

See also

[Gino.Exception.Exception403](#)

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.223 views/404.php File Reference

Template della pagina 404, Not Found.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.223.1 Detailed Description

Template della pagina 404, Not Found.

Variabili disponibili:

- **title**: string, titolo
- **message**: string, messaggio

See also

[Gino.Exception.Exception404](#)

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.224 views/500.php File Reference

Template della pagina 500, Internal Server Error.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.224.1 Detailed Description

Template della pagina 500, Internal Server Error.

Variabili disponibili:

- **title**: string, titolo
- **message**: string, messaggio

See also

[Gino.Exception.Exception500](#)

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.225 views/admin_table_form.php File Reference

Template della pagina di inserimento/modifica record in area amministrativa.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.225.1 Detailed Description

Template della pagina di inserimento/modifica record in area amministrativa.

Variabili disponibili:

- **title**: string, titolo
- **form_description**: html, testo spiegazioni
- **form** html, il form

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.226 views/admin_table_list.php File Reference

Template della lista record in area amministrativa.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.226.1 Detailed Description

Template della lista record in area amministrativa.

Variabili disponibili:

- **title**: string, titolo.
- **search_icon**: html, icona ricerca
- **form_filters_title**: string, titolo form filtri
- **form_filters**: html, form filtri
- **link_insert**: html, link inserimento nuovo record
- **link_export**: html, link esportazione record
- **description**: html, testo informativo
- **table**: html, tabella con i record ed i bottoni di manipolazione
- **tot_records**: int, numero di record
- **pagination**: html, paginazione (sommario e navigazione)

See also

- [Gino.Logger](#)

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.227 views/logger_stack_trace.php File Reference

Template che stampa lo stack trace di una Exception.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.227.1 Detailed Description

Template che stampa lo stack trace di una Exception.

Quando la costante DEBUG in [configuration.php](#) è settata a TRUE viene mostrato a video, altrimenti viene inviato per mail agli amministratori, costante ADMIN.

Variabili disponibili:

- **registry**: , istanza di [Gino.Registry](#)
- **exception**: , istanza dell'Exception
- **system_variables_html**: html, dump delle variabili di sistema (\$_SERVER, \$_SESSION, \$_REQUEST)

See also

[Gino.Logger](#)

Copyright

2015 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.228 views/logger_system_variables.php File Reference

Template che stampa un dump delle variabili di sistema (\$_SERVER, \$_SESSION, \$_REQUEST).

Namespaces

- [Gino](#)

Namespace principale di gino.

9.228.1 Detailed Description

Template che stampa un dump delle variabili di sistema (\$_SERVER, \$_SESSION, \$_REQUEST).

See also

[Gino.Logger](#)

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.229 views/paginator_navigator.php File Reference

Template dei controller per la navigazione delle pagine, paginazione.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.229.1 Detailed Description

Template dei controller per la navigazione delle pagine, paginazione.

Variabili disponibili:

- **pages**: array di pagine, ciascun elemento puo' essere:
 - un array del tipo array(numero_pagina => url)
 - un array del tipo array(numero_pagina => null) che indica la pagina attiva
 - una stringa, i puntini (...) che separano pagine non consecutive
- next: url pagina successiva, null se inesistente
- prev: url pagina precedente, null se inesistente

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.230 views/paginator_pagination.php File Reference

Template completo paginazione, include navigazione pagine e sommario.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.230.1 Detailed Description

Template completo paginazione, include navigazione pagine e sommario.

Variabili disponibili:

- **summary**: string, sommario
- **summary_label**: string, label del sommario
- **navigator**: html, controller di navigazione pagine [paginator_navigator.php](#)
- **gotopage**: html, interfaccia di invio a una pagina specifica

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.231 views/related_contents_list.php File Reference

Template che stampa la lista di contenuti correlati.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.231.1 Detailed Description

Template che stampa la lista di contenuti correlati.

Variabili disponibili:

- **related_contents**: array. Un array associativo \$content_type => \$links dove:
 - content_type: string, tipo di contenuto (pagine, news...)
 - links: array, array di link completi di tag a

Copyright

2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.232 views/section.php File Reference

Template di una section.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.232.1 Detailed Description

Template di una section.

Variabili disponibili:

- **class**: string, classe css della section
- **id**: string, attributo id della section
- **pre_header**: html, elementi da mostrare prima del titolo
- **header_links**: array, array di link
- **post_header**: html, elementi da mostrare dopo il titolo
- **header_class**: string, classe css del tag h1
- **title**: string, titolo section
- **content**: html, contenuto
- **footer**: html, contenuto del footer della section

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.233 views/tab.php File Reference

Template che stampa una tab html.

Namespaces

- [Gino](#)
Namespace principale di gino.

9.233.1 Detailed Description

Template che stampa una tab html.

Variabili disponibili:

- **link_position**: string, left|right. Posizione dei link.
- **id**: string. Attributo id del container.
- **title**: string. Titolo pagina.
- **tab_title**: string. Titolo tab.
- **links**: array|html. Link o array di link completi di tag a.
- **selected_link**: html. Link selezionato, completo di tag a, da confrontare con links.
- **content**: html, contenuto tab selezionata

Copyright

2013-2014 Otto srl MIT License <http://www.opensource.org/licenses/mit-license.php>

Authors

Marco Guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

9.234 views/table.php File Reference

Template utilizzato per visualizzare le tabelle.

Namespaces

- [Gino](#)

Namespace principale di gino.

9.234.1 Detailed Description

Template utilizzato per visualizzare le tabelle.

Copyright

2013-2014 Otto srl (<http://www.opensource.org/licenses/mit-license.php>) The MIT License

Author

marco guidotti guidottim@gmail.com
abidibo abidibo@gmail.com

Variabili disponibili:

- **form_start**: (string) apertura form (se presente)
- **form_end**: (string) chiusura form (se presente)
- **class**: classe css della tabella
- **caption**: (string) caption, opzionale
- **tr_class**: (string) classe css di una riga evidenziata
- **multiple_heads**: (bool) molteplici righe di intestazione
- **heads**: (array) array di intestazioni o array di array di intestazioni (**multiple_heads**), ciascun elemento può essere una stringa (testo intestazione) o un array associativo:
 - **class**: classe css da applicare al tag th
 - **text**: testo intestazione
- **rows**: (array) array di righe di tabella. Ciascuna riga è una array di celle. Ciascuna cella può essere una stringa (testo della cella) o un array associativo:
 - **evidence**: (bool) se TRUE la riga usa la classe **tr_class**
 - **header**: (bool) se TRUE la cella è da considerarsi intestazione (th)

- **colspan:** (int) attributo colspan
- **title:** (string) attributo title
- **class:** classe css della cella
- **text:** testo della cella
- **foots:** (string|array) table footer, può essere un array di testi (uno per ogni cella), o una stringa (unica cella con colspan adatto)

Index

- \$COOKIES
 - Gino::Http::Request, [461](#)
- \$FILES
 - Gino::Http::Request, [461](#)
- \$GET
 - Gino::Http::Request, [461](#)
- \$META
 - Gino::Http::Request, [461](#)
- \$POST
 - Gino::Http::Request, [461](#)
- \$REQUEST
 - Gino::Http::Request, [461](#)
- \$_abspath
 - Gino::GImage, [271](#)
- \$_access
 - Gino::Controller, [136](#)
- \$_action
 - Gino::Options, [407](#)
- \$_add_related
 - Gino::ForeignKeyField, [231](#)
 - Gino::ManyToManyField, [348](#)
- \$_add_related_url
 - Gino::ForeignKeyField, [231](#)
 - Gino::ManyToManyField, [348](#)
- \$_affected
 - Gino::Plugin::mysql, [403](#)
 - Gino::Plugin::sqlsrv, [512](#)
- \$_align_dict
 - Gino::Template, [539](#)
- \$_allow_insertion
 - Gino::AdminTable, [85](#)
- \$_allow_numbers
 - Gino::Captcha, [119](#)
- \$_aut_pwd
 - Gino::App::Auth::auth, [110](#)
- \$_aut_pwd_length
 - Gino::App::Auth::auth, [110](#)
- \$_auth
 - Gino::App::Auth::Ldap, [331](#)
- \$_auto_increment
 - Gino::Field, [213](#)
- \$_auto_now
 - Gino::DatetimeField, [152](#)
- \$_auto_now_add
 - Gino::DatetimeField, [152](#)
- \$_autofill
 - Gino::SlugField, [499](#)
- \$_blackList
 - Gino::App::PhpModuleView::phpModuleView, [450](#)
- \$_blocks_number
 - Gino::Template, [539](#)
- \$_blocks_properties
 - Gino::Template, [539](#)
- \$_box_tpl_code
 - Gino::App::Page::page, [422](#)
- \$_cache
 - Gino::App::Menu::menu, [364](#)
 - Gino::Plugin::mysql, [403](#)
 - Gino::Plugin::sqlsrv, [512](#)
- \$_check_is_constraint
 - Gino::Model, [382](#)
- \$_check_type
 - Gino::FileField, [224](#)
- \$_chgP
 - Gino::Model, [382](#)
- \$_class
 - Gino::Css, [144](#)
 - Gino::Frontend, [261](#)
 - Gino::Options, [407](#)
- \$_class_img
 - Gino::Controller, [136](#)
- \$_class_name
 - Gino::Controller, [136](#)
- \$_class_prefix
 - Gino::Options, [407](#)
- \$_class_www
 - Gino::Controller, [136](#)
- \$_cnt
 - Gino::Plugin::mysql, [403](#)
 - Gino::Plugin::sqlsrv, [513](#)
- \$_comment_moderation
 - Gino::App::Page::page, [422](#)
- \$_comment_notification
 - Gino::App::Page::page, [422](#)
- \$_connection
 - Gino::Plugin::mysql, [403](#)
 - Gino::Plugin::sqlsrv, [513](#)
- \$_content
 - Gino::Http::Response, [465](#)
- \$_content_disposition
 - Gino::Http::ResponseFile, [468](#)
- \$_content_type
 - Gino::Http::Response, [465](#)
- \$_controller
 - Gino::AdminTable, [85](#)
 - Gino::ManyToManyThroughField, [356](#)
 - Gino::Model, [382](#)
- \$_controller_view

- Gino::Router, 478
- \$_css
 - Gino::Compressor, 127
- \$_css_list
 - Gino::Css, 144
 - Gino::Frontend, 261
- \$_current_page
 - Gino::Paginator, 438
- \$_data
 - Gino::Export, 201
 - Gino::View, 559
- \$_data_dir
 - Gino::Controller, 136
- \$_data_www
 - Gino::Controller, 136
- \$_db
 - Gino::Access, 70
 - Gino::AdminTable, 85
 - Gino::Controller, 136
 - Gino::Core, 138
 - Gino::Model, 382
 - Gino::Options, 407
 - Gino::Translation, 548
- \$_db_charset
 - Gino::Db, 154
 - Gino::Plugin::mysql, 403
 - Gino::Plugin::sqlsrv, 513
- \$_db_dbname
 - Gino::Db, 154
- \$_db_host
 - Gino::Db, 154
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_db_name
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_db_pass
 - Gino::Db, 154
- \$_db_password
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_db_schema
 - Gino::Db, 154
- \$_db_user
 - Gino::Db, 154
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_dbconn
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_dbms
 - Gino::sqlcode, 500
- \$_dbresults
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_debug
 - Gino::App::Auth::Ldap, 331
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_decimal_digits
 - Gino::FloatField, 228
- \$_default
 - Gino::BooleanField, 114
 - Gino::EnumField, 185
 - Gino::Field, 213
- \$_default_name
 - Gino::DirectoryField, 174
- \$_default_widget
 - Gino::Field, 213
- \$_delete_deny
 - Gino::AdminTable, 85
- \$_delete_file
 - Gino::FileField, 224
- \$_dft_view_folder
 - Gino::View, 559
- \$_dir
 - Gino::GImage, 271
- \$_directory
 - Gino::FileField, 224
- \$_div_field_width
 - Gino::Form, 256
- \$_div_label_width
 - Gino::Form, 256
- \$_doc_extension
 - Gino::App::Attachment::AttachmentItem, 99
- \$_ds
 - Gino::Cache, 117
- \$_edit_deny
 - Gino::AdminTable, 85
- \$_emitter_listeners
 - Gino::EventDispatcher, 189
- \$_enable_cache
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_enabled
 - Gino::Cache, 117
- \$_encoding
 - Gino::Http::Response, 465
- \$_entry_tpl_code
 - Gino::App::Page::page, 422
- \$_enum
 - Gino::BooleanField, 114
 - Gino::EnumField, 185
 - Gino::ForeignKeyField, 231
 - Gino::ManyToManyField, 348
 - Gino::ManyToManyInlineField, 352
 - Gino::ManyToManyThroughField, 356
- \$_extension_denied
 - Gino::Form, 256
- \$_extension_img
 - Gino::App::Graphics::GraphicsItem, 278
 - Gino::App::Page::PageEntry, 435
- \$_extensions
 - Gino::FileField, 224
- \$_fckeditor_field
 - Gino::Form, 256

- `$_fields`
 - Gino::Export, 201
- `$_fields_label`
 - Gino::Model, 382
- `$_filename`
 - Gino::Http::ResponseFile, 468
- `$_filter_add`
 - Gino::AdminTable, 86
- `$_filter_fields`
 - Gino::AdminTable, 86
- `$_filter_join`
 - Gino::AdminTable, 86
- `$_filter_search`
 - Gino::App::Auth::Ldap, 331
- `$_flag_language`
 - Gino::App::Language::language, 320
- `$_fld`
 - Gino::Cache, 117
- `$_font_file`
 - Gino::Captcha, 119
- `$_foreign`
 - Gino::ForeignKeyField, 231
- `$_foreign_order`
 - Gino::ForeignKeyField, 231
- `$_foreign_where`
 - Gino::ForeignKeyField, 231
- `$_form`
 - Gino::AdminTable, 86
 - Gino::InputForm, 302
- `$_form_name`
 - Gino::Form, 256
- `$_gform`
 - Gino::App::Menu::MenuVoice, 368
- `$_grp`
 - Gino::Cache, 117
- `$_head`
 - Gino::Export, 201
- `$_headers`
 - Gino::Http::Response, 465
- `$_height`
 - Gino::Captcha, 120
 - Gino::GImage, 271
 - Gino::ImageField, 293
- `$_hidden`
 - Gino::AdminTable, 86
- `$_home`
 - Gino::Access, 70
 - Gino::Controller, 136
 - Gino::Options, 407
 - Gino::Template, 539
- `$_ico_calendar_path`
 - Gino::Form, 256
- `$_ico_more`
 - Gino::App::Menu::menu, 364
- `$_ifp`
 - Gino::AdminTable, 86
- `$_image`
 - Gino::GImage, 271
- `$_image_type`
 - Gino::GImage, 271
- `$_img_extension`
 - Gino::App::Attachment::AttachmentItem, 99
- `$_input_field`
 - Gino::Form, 256
- `$_input_max`
 - Gino::Form, 256
- `$_input_size`
 - Gino::Form, 256
- `$_inst_md1`
 - Gino::App::SearchSite::searchSite, 485
- `$_instance`
 - Gino::Controller, 136
 - Gino::Options, 407
- `$_instance_class`
 - Gino::Css, 144
- `$_instance_label`
 - Gino::Controller, 136
- `$_instance_name`
 - Gino::Controller, 136
- `$_instances`
 - Gino::Singleton, 491
- `$_int_digits`
 - Gino::FloatField, 228
- `$_interface`
 - Gino::App::PhpModuleView::PhpModule, 447
 - Gino::Css, 144
 - Gino::Skin, 496
 - Gino::Template, 539
- `$_interval`
 - Gino::Paginator, 438
- `$_is_constraint`
 - Gino::Model, 382
- `$_items_for_page`
 - Gino::Paginator, 438
- `$_items_number`
 - Gino::Paginator, 438
- `$_join_table`
 - Gino::ManyToManyField, 348
- `$_join_table_id`
 - Gino::ManyToManyField, 348
- `$_join_table_m2m_id`
 - Gino::ManyToManyField, 348
- `$_js`
 - Gino::Compressor, 127
 - Gino::SlugField, 499
- `$_justthese_search`
 - Gino::App::Auth::Ldap, 331
- `$_label`
 - Gino::Css, 144
 - Gino::Field, 213
- `$_label_field1`
 - Gino::App::Auth::auth, 110
- `$_label_field2`
 - Gino::App::Auth::auth, 110
- `$_label_field3`
 - Gino::App::Auth::auth, 110

- `$_lastid`
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- `$_ldap_auth`
 - Gino::App::Auth::auth, 110
- `$_ldap_auth_only`
 - Gino::App::Auth::auth, 110
- `$_ldap_auth_password`
 - Gino::App::Auth::auth, 110
- `$_ldap_base_dn`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_domain`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_ds`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_host`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_log`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_password`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_port`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_protocol_version`
 - Gino::App::Auth::Ldap, 331
- `$_ldap_single_user`
 - Gino::App::Auth::auth, 110
- `$_ldap_username`
 - Gino::App::Auth::Ldap, 331
- `$_lenght`
 - Gino::Field, 213
- `$_letters`
 - Gino::Captcha, 120
- `$_list_display`
 - Gino::AdminTable, 86
- `$_list_remove`
 - Gino::AdminTable, 86
- `$_listeners`
 - Gino::EventDispatcher, 189
- `$_lng`
 - Gino::Translation, 548
- `$_lngDft`
 - Gino::Translation, 548
- `$_lng_dft`
 - Gino::Model, 382
- `$_lng_nav`
 - Gino::Model, 382
- `$_lng_trl`
 - Gino::Form, 256
- `$_locale`
 - Gino::Controller, 136
 - Gino::Model, 382
- `$_m2m`
 - Gino::ManyToManyField, 348
 - Gino::ManyToManyInlineField, 352
 - Gino::ManyToManyThroughField, 356
 - Gino::Model, 382
- `$_m2m_controller`
 - Gino::ManyToManyInlineField, 352
 - Gino::ManyToManyThroughField, 356
- `$_m2m_order`
 - Gino::ManyToManyField, 348
 - Gino::ManyToManyInlineField, 352
- `$_m2m_where`
 - Gino::ManyToManyField, 348
 - Gino::ManyToManyInlineField, 352
- `$_m2mt`
 - Gino::Model, 382
- `$_max_file_size`
 - Gino::FormField, 224
 - Gino::Form, 256
- `$_mdlLink`
 - Gino::Css, 144
 - Gino::Frontend, 261
- `$_menu_functions_list`
 - Gino::App::Menu::menu, 364
- `$_method`
 - Gino::Form, 256
- `$_model`
 - Gino::Field, 214
- `$_model_controller_class`
 - Gino::TagField, 530
- `$_model_controller_instance`
 - Gino::TagField, 530
- `$_model_label`
 - Gino::Model, 382
- `$_model_table_id`
 - Gino::ManyToManyThroughField, 356
- `$_module`
 - Gino::Css, 144
 - Gino::Frontend, 261
- `$_module_app`
 - Gino::Options, 407
- `$_module_id`
 - Gino::Frontend, 261
- `$_multi_language`
 - Gino::Form, 256
- `$_name`
 - Gino::Captcha, 120
 - Gino::Css, 144
 - Gino::Field, 214
- `$_newsletter_entries_number`
 - Gino::App::Page::page, 422
- `$_newsletter_tpl_code`
 - Gino::App::Page::page, 422
- `$_numberrows`
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- `$_numbers`
 - Gino::Captcha, 120
- `$_options`
 - Gino::App::Auth::auth, 110
 - Gino::App::Language::language, 320
 - Gino::App::Menu::menu, 364
 - Gino::App::PhpModuleView::phpModuleView, 450
 - Gino::App::SearchSite::searchSite, 485

- Gino::Form, 256
- \$_optionsLabels
 - Gino::App::Auth::auth, 110
 - Gino::App::Language::language, 320
 - Gino::App::Menu::menu, 364
 - Gino::App::PhpModuleView::phpModuleView, 450
 - Gino::App::SearchSite::searchSite, 485
- \$_order
 - Gino::Export, 201
- \$_other_field1
 - Gino::App::Auth::auth, 110
- \$_other_field2
 - Gino::App::Auth::auth, 110
- \$_other_field3
 - Gino::App::Auth::auth, 110
- \$_p
 - Gino::Model, 382
- \$_pages_number
 - Gino::Paginator, 438
- \$_path
 - Gino::DirectoryField, 174
- \$_path_abs
 - Gino::FileField, 224
- \$_path_add
 - Gino::FileField, 224
- \$_pdf_extension
 - Gino::App::Attachment::AttachmentItem, 99
- \$_permissions
 - Gino::Controller, 136
- \$_plink
 - Gino::Controller, 136
- \$_prefix
 - Gino::Cache, 117
 - Gino::DirectoryField, 174
 - Gino::FileField, 224
- \$_prefix_file
 - Gino::Form, 256
 - Gino::ImageField, 293
- \$_prefix_thumb
 - Gino::Form, 256
 - Gino::ImageField, 293
- \$_primary_key
 - Gino::Field, 214
- \$_pwd_length_max
 - Gino::App::Auth::auth, 110
- \$_pwd_length_min
 - Gino::App::Auth::auth, 110
- \$_pwd_numeric_number
 - Gino::App::Auth::auth, 110
- \$_qry
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_redirect
 - Gino::Exception::Exception403, 191
- \$_registry
 - Gino::AdminTable, 86
 - Gino::Cache, 117
 - Gino::Controller, 136
 - Gino::Core, 138
 - Gino::Document, 179
 - Gino::Form, 256
 - Gino::Frontend, 261
 - Gino::Model, 382
 - Gino::Router, 478
 - Gino::Translation, 548
 - Gino::View, 560
- \$_request
 - Gino::AdminTable, 86
 - Gino::Document, 179
 - Gino::Form, 256
 - Gino::Http::Response, 465
 - Gino::Model, 382
 - Gino::Router, 478
- \$_requestVar
 - Gino::Form, 256
- \$_required
 - Gino::Field, 214
- \$_resize
 - Gino::ImageField, 293
- \$_rids
 - Gino::Export, 201
- \$_rows
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_s
 - Gino::Export, 201
- \$_self_registration
 - Gino::App::Auth::auth, 110
- \$_self_registration_active
 - Gino::App::Auth::auth, 110
- \$_session
 - Gino::Access, 70
 - Gino::AdminTable, 86
 - Gino::Controller, 136
 - Gino::Locale, 340
- \$_showcase_auto_interval
 - Gino::App::Page::page, 422
- \$_showcase_auto_start
 - Gino::App::Page::page, 423
- \$_showcase_number
 - Gino::App::Page::page, 423
- \$_showcase_title
 - Gino::App::Page::page, 423
- \$_showcase_tpl_code
 - Gino::App::Page::page, 423
- \$_sql
 - Gino::Plugin::mysql, 404
 - Gino::Plugin::sqlsrv, 513
- \$_status_code
 - Gino::Http::Response, 465
- \$_status_text
 - Gino::Http::Response, 466
- \$_strings
 - Gino::Locale, 340
- \$_structure
 - Gino::Model, 382

- `$_sys_mdl`
 - Gino::App::SearchSite::searchSite, 485
- `$_table`
 - Gino::Export, 201
 - Gino::Field, 214
 - Gino::GImage, 271
 - Gino::Search, 481
- `$_table_tag`
 - Gino::GTag, 284
- `$_table_tag_taggeditem`
 - Gino::GTag, 284
- `$_tbl_attachment`
 - Gino::Form, 256
- `$_tbl_attachment_ctg`
 - Gino::Form, 256
- `$_tbl_data`
 - Gino::App::PhpModuleView::PhpModule, 447
 - Gino::Css, 144
 - Gino::Model, 382
 - Gino::Skin, 496
 - Gino::Template, 539
- `$_tbl_log_access`
 - Gino::App::Statistics::statistics, 516
- `$_tbl_module`
 - Gino::Css, 144
- `$_tbl_opt`
 - Gino::App::Menu::menu, 364
 - Gino::App::Page::page, 423
 - Gino::App::PhpModuleView::phpModuleView, 450
- `$_tbl_options`
 - Gino::Options, 407
- `$_tbl_translation`
 - Gino::Translation, 548
- `$_tbl_user_role`
 - Gino::App::Menu::MenuVoice, 368
- `$_tc`
 - Gino::Cache, 117
- `$_test_anonymous`
 - Gino::App::Auth::Ldap, 331
- `$_test_search`
 - Gino::App::Auth::Ldap, 331
- `$_textarea_field`
 - Gino::Form, 256
- `$_thumb`
 - Gino::ImageField, 293
- `$_thumb_height`
 - Gino::ImageField, 293
- `$_thumb_width`
 - Gino::ImageField, 293
- `$_title`
 - Gino::App::Attachment::attachment, 92
 - Gino::App::Auth::auth, 110
 - Gino::App::Graphics::graphics, 276
 - Gino::App::Language::language, 320
 - Gino::App::Menu::menu, 364
 - Gino::App::PhpModuleView::phpModuleView, 450
 - Gino::App::SearchSite::searchSite, 485
 - Gino::App::SysClass::sysClass, 522
 - Gino::Options, 407
- `$_title_visible`
 - Gino::App::PhpModuleView::phpModuleView, 450
- `$_tmp_image`
 - Gino::GImage, 271
- `$_trd`
 - Gino::Controller, 136
 - Gino::Form, 256
 - Gino::Model, 382
- `$_trnsI`
 - Gino::CharField, 124
 - Gino::TextField, 543
- `$_trnsI_id`
 - Gino::Form, 256
- `$_trnsI_table`
 - Gino::Form, 257
- `$_types_allowed`
 - Gino::FileField, 224
- `$_um_dict`
 - Gino::Template, 539
- `$_unique_key`
 - Gino::Field, 214
- `$_url`
 - Gino::Http::Redirect, 453
- `$_url_class`
 - Gino::Router, 478
- `$_url_content`
 - Gino::Document, 179
- `$_url_instance`
 - Gino::Router, 478
- `$_url_method`
 - Gino::Router, 478
- `$_user_more`
 - Gino::App::Auth::auth, 110
- `$_user_name`
 - Gino::App::Auth::Ldap, 331
- `$_user_password`
 - Gino::App::Auth::Ldap, 331
- `$_user_view`
 - Gino::App::Auth::auth, 110
- `$_username_as_email`
 - Gino::App::Auth::auth, 110
- `$_users_for_page`
 - Gino::App::Auth::auth, 110
- `$_validation`
 - Gino::Form, 257
- `$_value`
 - Gino::Field, 214
- `$_value_type`
 - Gino::Field, 214
- `$_vars`
 - Gino::Session, 488
- `$_view`
 - Gino::AdminTable, 86
- `$_view_dir`
 - Gino::App::SearchSite::searchSite, 485
 - Gino::Controller, 136
- `$_view_folder`

- Gino::View, 560
- \$_view_list
 - Gino::Frontend, 261
- \$_view_tpl
 - Gino::View, 560
- \$_width
 - Gino::Captcha, 120
 - Gino::GImage, 271
 - Gino::ImageField, 293
- \$_xls_extension
 - Gino::App::Attachment::AttachmentItem, 100
- \$a
 - Gino::JSMin, 312
- \$absolute_url
 - Gino::Http::Request, 461
- \$accept
 - Gino::MobileDetect, 370
- \$b
 - Gino::JSMin, 312
- \$devices
 - Gino::MobileDetect, 370
- \$error
 - ReCaptchaResponse, 451
- \$extension_media
 - Gino::App::Auth::User, 557
- \$id
 - Gino::Cache, 117
- \$input
 - Gino::JSMin, 312
- \$inputIndex
 - Gino::JSMin, 312
- \$inputLength
 - Gino::JSMin, 312
- \$isAndroid
 - Gino::MobileDetect, 370
- \$isBlackberry
 - Gino::MobileDetect, 370
- \$isGeneric
 - Gino::MobileDetect, 370
- \$isIphone
 - Gino::MobileDetect, 370
- \$isMobile
 - Gino::MobileDetect, 370
- \$isOpera
 - Gino::MobileDetect, 370
- \$isPalm
 - Gino::MobileDetect, 370
- \$isWindows
 - Gino::MobileDetect, 370
- \$is_valid
 - ReCaptchaResponse, 451
- \$lookAhead
 - Gino::JSMin, 312
- \$method
 - Gino::Http::Request, 461
- \$output
 - Gino::JSMin, 312
- \$path
 - Gino::Http::Request, 461
- \$query_string
 - Gino::Http::Request, 461
- \$request_uri
 - Gino::Http::Request, 461
- \$root_absolute_url
 - Gino::Http::Request, 461
- \$session
 - Gino::Form, 257
 - Gino::Http::Request, 461
- \$sessionState
 - Gino::Session, 488
- \$table
 - Gino::App::Attachment::AttachmentCtg, 95
 - Gino::App::Attachment::AttachmentItem, 100
 - Gino::App::Auth::Group, 281
 - Gino::App::Auth::Permission, 444
 - Gino::App::Auth::User, 557
 - Gino::App::Graphics::GraphicsItem, 278
 - Gino::App::Language::Lang, 316
 - Gino::App::Module::ModuleInstance, 394
 - Gino::App::Page::PageCategory, 425
 - Gino::App::Page::PageComment, 429
 - Gino::App::Page::PageEntry, 435
 - Gino::App::PhpModuleView::PhpModule, 447
 - Gino::App::Statistics::LogAccess, 342
 - Gino::App::SysClass::ModuleApp, 391
 - Gino::App::Sysconf::Conf, 129
 - Gino::Css, 144
 - Gino::Skin, 496
 - Gino::Template, 539
- \$table_block
 - Gino::Template, 539
- \$table_group_perm
 - Gino::App::Auth::Group, 281
- \$table_groups
 - Gino::App::Auth::User, 557
- \$table_more
 - Gino::App::Auth::User, 557
- \$table_perm_user
 - Gino::App::Auth::Permission, 444
- \$tbl_voices
 - Gino::App::Menu::MenuVoice, 368
- \$url
 - Gino::Http::Request, 461
- \$user
 - Gino::Http::Request, 461
- \$userAgent
 - Gino::MobileDetect, 370
- \$vars
 - Gino::Registry, 458
- _IMAGE_GIF_
 - Gino::Form, 257
 - Gino::ImageField, 293
- _IMAGE_JPG_
 - Gino::Form, 257
 - Gino::ImageField, 293
- _IMAGE_PNG_
 - Gino::Form, 257
 - Gino::ImageField, 293

- Gino::Form, 257
- Gino::ImageField, 293
- __autoload
 - class.Loader.php, 602
- __call
 - Gino::Controller, 132
 - Gino::MobileDetect, 369
- __clone
 - Gino::Singleton, 490
- __construct
 - Gino::Access, 68
 - Gino::AdminTable, 74
 - Gino::App::Attachment::AttachmentCtg, 94
 - Gino::App::Attachment::AttachmentItem, 97
 - Gino::App::Attachment::attachment, 90
 - Gino::App::Auth::Group, 280
 - Gino::App::Auth::Ldap, 329
 - Gino::App::Auth::Permission, 442
 - Gino::App::Auth::User, 551
 - Gino::App::Auth::auth, 104
 - Gino::App::Graphics::GraphicsItem, 278
 - Gino::App::Graphics::graphics, 274
 - Gino::App::Index::index, 295
 - Gino::App::Instruments::instruments, 304
 - Gino::App::Language::Lang, 315
 - Gino::App::Language::language, 318
 - Gino::App::Layout::layout, 323
 - Gino::App::Menu::MenuVoice, 366
 - Gino::App::Menu::menu, 359
 - Gino::App::Module::ModuleInstance, 393
 - Gino::App::Module::module, 384
 - Gino::App::Page::PageCategory, 424
 - Gino::App::Page::PageComment, 427
 - Gino::App::Page::PageEntry, 431
 - Gino::App::Page::page, 414
 - Gino::App::PhpModuleView::PhpModule, 445
 - Gino::App::PhpModuleView::phpModuleView, 449
 - Gino::App::SearchSite::searchSite, 483
 - Gino::App::Statistics::LogAccess, 341
 - Gino::App::Statistics::statistics, 515
 - Gino::App::SysClass::ModuleApp, 390
 - Gino::App::SysClass::sysClass, 518
 - Gino::App::Sysconf::Conf, 129
 - Gino::App::Sysconf::sysconf, 524
 - Gino::App::Sysfunc::sysfunc, 526
 - Gino::BooleanField, 112
 - Gino::Cache, 115
 - Gino::Captcha, 118
 - Gino::CharField, 122
 - Gino::Compressor, 125
 - Gino::Controller, 132
 - Gino::Core, 137
 - Gino::Css, 141
 - Gino::DataCache, 145
 - Gino::DateField, 147
 - Gino::DatetimeField, 150
 - Gino::DbManager, 157
 - Gino::DirectoryField, 171
 - Gino::Document, 175
 - Gino::EmailField, 181
 - Gino::EnumField, 183
 - Gino::Exception::Exception403, 190
 - Gino::Exception::Exception404, 192
 - Gino::Exception::Exception500, 194
 - Gino::Field, 204
 - Gino::FileField, 217
 - Gino::FloatField, 226
 - Gino::ForeignKeyField, 229
 - Gino::Form, 234
 - Gino::Frontend, 258
 - Gino::GImage, 265
 - Gino::Http::Redirect, 453
 - Gino::Http::Request, 460
 - Gino::Http::Response, 463
 - Gino::Http::ResponseFile, 467
 - Gino::Http::ResponseForbidden, 469
 - Gino::Http::ResponseJson, 471
 - Gino::Http::ResponseNotFound, 472
 - Gino::Http::ResponseServerError, 474
 - Gino::ImageField, 287
 - Gino::InputForm, 297
 - Gino::IntegerField, 306
 - Gino::JSMIn, 310
 - Gino::Locale, 337
 - Gino::ManyToManyField, 345
 - Gino::ManyToManyInlineField, 350
 - Gino::ManyToManyThroughField, 354
 - Gino::MobileDetect, 369
 - Gino::Model, 374
 - Gino::Options, 405
 - Gino::OutputCache, 409
 - Gino::Paginator, 436
 - Gino::Plugin::mysql, 397
 - Gino::Plugin::sqlsrv, 503
 - Gino::Router, 476
 - Gino::Search, 480
 - Gino::Session, 487
 - Gino::Singleton, 490
 - Gino::Skin, 493
 - Gino::SlugField, 497
 - Gino::TagField, 529
 - Gino::Template, 533
 - Gino::TextField, 541
 - Gino::TimeField, 544
 - Gino::Translation, 546
 - Gino::View, 558
 - Gino::YearField, 561
 - Gino::sqlcode, 500
 - __get
 - Gino::Model, 375
 - Gino::Registry, 455
 - Gino::Session, 487
 - __invoke
 - Gino::Document, 176
 - Gino::Http::Response, 463
 - __isset

- Gino::Session, [487](#)
- __set
 - Gino::Model, [375](#)
 - Gino::Registry, [455](#)
 - Gino::Session, [487](#)
- __sleep
 - Gino::Singleton, [490](#)
- __toString
 - Gino::App::Attachment::AttachmentCtg, [94](#)
 - Gino::App::Attachment::AttachmentItem, [98](#)
 - Gino::App::Auth::Group, [280](#)
 - Gino::App::Auth::Permission, [443](#)
 - Gino::App::Auth::User, [552](#)
 - Gino::App::Graphics::GraphicsItem, [278](#)
 - Gino::App::Language::Lang, [315](#)
 - Gino::App::Page::PageCategory, [425](#)
 - Gino::App::Page::PageComment, [427](#)
 - Gino::App::Page::PageEntry, [432](#)
 - Gino::App::SysClass::ModuleApp, [390](#)
 - Gino::App::Sysconf::Conf, [129](#)
 - Gino::BooleanField, [112](#)
 - Gino::DateField, [148](#)
 - Gino::DatetimeField, [151](#)
 - Gino::EnumField, [183](#)
 - Gino::Field, [205](#)
 - Gino::FloatField, [226](#)
 - Gino::ForeignKeyField, [230](#)
 - Gino::ManyToManyField, [346](#)
 - Gino::ManyToManyInlineField, [351](#)
 - Gino::ManyToManyThroughField, [354](#)
 - Gino::Model, [375](#)
 - Gino::TimeField, [544](#)
- __unset
 - Gino::Session, [488](#)
- __wakeup
 - Gino::Singleton, [490](#)
- _recaptcha_aes_encrypt
 - recaptchalib.php, [638](#)
- _recaptcha_aes_pad
 - recaptchalib.php, [638](#)
- _recaptcha_http_post
 - recaptchalib.php, [638](#)
- _recaptcha_mailhide_email_parts
 - recaptchalib.php, [639](#)
- _recaptcha_mailhide_urlbase64
 - recaptchalib.php, [639](#)
- _recaptcha_qsencode
 - recaptchalib.php, [639](#)
- ACTION_DELETE_A
 - Gino::JSMin, [312](#)
- ACTION_DELETE_A_B
 - Gino::JSMin, [312](#)
- ACTION_KEEP_A
 - Gino::JSMin, [312](#)
- ADMINS
 - configuration.php, [589](#)
- APP_DIR
 - Gino, [53](#)
- abiMapLib
 - Gino::Javascript, [307](#)
- absolutePath
 - Gino, [31](#)
- Access, [67](#)
- accessPage
 - Gino::App::Page::page, [414](#)
- accessWhere
 - Gino::App::Page::PageEntry, [432](#)
- action
 - Gino::JSMin, [310](#)
- actionComment
 - Gino::App::Page::page, [414](#)
- actionCopyTemplate
 - Gino::Template, [533](#)
- actionCss
 - Gino::App::Layout::layout, [323](#)
- actionCssLayout
 - Gino::Css, [141](#)
- actionDelCss
 - Gino::App::Layout::layout, [323](#)
- actionDelCssLayout
 - Gino::Css, [141](#)
- actionDelMenuVoice
 - Gino::App::Menu::menu, [359](#)
- actionDelSkin
 - Gino::App::Layout::layout, [324](#)
 - Gino::Skin, [493](#)
- actionDelTemplate
 - Gino::App::Layout::layout, [324](#)
 - Gino::Template, [533](#)
- actionEditModule
 - Gino::App::Module::module, [384](#)
- actionEditModuleActive
 - Gino::App::Module::module, [385](#)
- actionEditSysClass
 - Gino::App::SysClass::sysClass, [518](#)
- actionEditSysClassActive
 - Gino::App::SysClass::sysClass, [518](#)
- actionFiles
 - Gino::App::Layout::layout, [324](#)
- actionFreeTemplate
 - Gino::Template, [534](#)
- actionInsertModule
 - Gino::App::Module::module, [385](#)
- actionInsertSysClass
 - Gino::App::SysClass::sysClass, [519](#)
- actionJoinGroupPermission
 - Gino::App::Auth::auth, [104](#)
- actionJoinUserPermission
 - Gino::App::Auth::auth, [104](#)
- actionManualSysClass
 - Gino::App::SysClass::sysClass, [519](#)
- actionMenuVoice
 - Gino::App::Menu::menu, [359](#)
- actionModule
 - Gino::App::Module::module, [385](#)
- actionModuleFile

- Gino::Frontend, 258
- actionOptions
 - Gino::Options, 406
- actionPhpModule
 - Gino::App::PhpModuleView::PhpModule, 446
- actionRemoveModule
 - Gino::App::Module::module, 386
- actionRemoveSysClass
 - Gino::App::SysClass::sysClass, 519
- actionSkin
 - Gino::App::Layout::layout, 325
 - Gino::Skin, 494
- actionTemplate
 - Gino::App::Layout::layout, 325
 - Gino::Template, 534
- actionTranslation
 - Gino::Translation, 546
- actionUpdateOrder
 - Gino::App::Menu::menu, 359
- actionUpgradeSysClass
 - Gino::App::SysClass::sysClass, 520
- actionquery
 - Gino::DbManager, 157
 - Gino::Plugin::mysql, 397
 - Gino::Plugin::sqlsrv, 504
- addBlockForm
 - Gino::Template, 534
- addCss
 - Gino::Compressor, 125
 - Gino::Registry, 456
- addHeadLink
 - Gino::Registry, 456
- addJs
 - Gino::Compressor, 125
 - Gino::Registry, 456
- addMeta
 - Gino::Registry, 457
- addWhereClauses
 - Gino::AdminTable, 75
- addWhereExtra
 - Gino::AdminTable, 75
- addWhereJoin
 - Gino::AdminTable, 75
- addm2m
 - Gino::Model, 375
- addm2mthrough
 - Gino::Model, 376
- admin_actionUser
 - Gino::sqlcode, 500
- admin_page
 - Gino::App::Index::index, 295
- adminDelete
 - Gino::AdminTable, 75
- adminForm
 - Gino::AdminTable, 76
- adminList
 - Gino::AdminTable, 76
- adminListOrder
 - Gino::Field, 205
- AdminTable, 70
- AdminTable_AuthUser, 86
- affected
 - Gino::DbManager, 157
 - Gino::Plugin::mysql, 397
 - Gino::Plugin::sqlsrv, 504
- analytics
 - Gino::Javascript, 307
- answer
 - Gino::Core, 138
- app/attachment/class.AttachmentCtg.php, 563
- app/attachment/class.AttachmentItem.php, 563
- app/attachment/class.AttachmentItemAdminTable.php, 564
- app/attachment/class_attachment.php, 564
- app/auth/class.AdminTable_AuthUser.php, 565
- app/auth/class.Group.php, 566
- app/auth/class.Ldap.php, 566
- app/auth/class.Permission.php, 567
- app/auth/class.User.php, 567
- app/auth/class_auth.php, 568
- app/auth/config.Ldap.php, 568
- app/auth/views/login.php, 569
- app/graphics/class.GraphicsItem.php, 569
- app/graphics/class_graphics.php, 570
- app/graphics/views/render.php, 571
- app/index/class_index.php, 571
- app/instruments/class_instruments.php, 571
- app/language/class.Lang.php, 572
- app/language/class_language.php, 572
- app/language/views/choice_language.php, 573
- app/layout/class_layout.php, 574
- app/menu/class.MenuVoice.php, 574
- app/menu/class_menu.php, 575
- app/menu/views/render.php, 571
- app/menu/views/render_mainMenu.php, 575
- app/menu/views/render_menu_admin.php, 576
- app/module/class.ModuleInstance.php, 577
- app/module/class_module.php, 577
- app/page/class.PageCategory.php, 578
- app/page/class.PageComment.php, 578
- app/page/class.PageEntry.php, 579
- app/page/class_page.php, 579
- app/page/language/en_US/page_lang.php, 580
- app/page/language/it_IT/page_lang.php, 580
- app/page/views/box.php, 580
- app/page/views/showcase.php, 581
- app/page/views/view.php, 581
- app/phpModuleView/class.PhpModule.php, 582
- app/phpModuleView/class_phpModuleView.php, 583
- app/searchSite/class_searchSite.php, 583
- app/searchSite/views/form.php, 584
- app/searchSite/views/results.php, 584
- app/statistics/class.LogAccess.php, 585
- app/statistics/class_statistics.php, 585
- app/sysClass/class.ModuleApp.php, 586
- app/sysClass/class_sysClass.php, 586

- app/sysconf/class.Conf.php, [587](#)
- app/sysconf/class_sysconf.php, [587](#)
- app/sysfunc/class_sysfunc.php, [588](#)
- arequired
 - Gino::Form, [235](#)
- arrayToObject
 - Gino, [31](#)
- assign
 - Gino::View, [559](#)
- attachment, [89](#)
- AttachmentCtg, [92](#)
- AttachmentItem, [95](#)
- AttachmentItemAdminTable, [100](#)
- attributeVar
 - Gino, [31](#)
- auth, [101](#)
- Authentication
 - Gino::Access, [68](#)
- authentication
 - Gino::App::Auth::Ldap, [329](#)
- AuthenticationMethod
 - Gino::Access, [69](#)
- autoIncValue
 - Gino::DbManager, [157](#)
 - Gino::Plugin::mysql, [397](#)
 - Gino::Plugin::sqlsrv, [504](#)
- backOffice
 - Gino::AdminTable, [78](#)
- baseFileName
 - Gino, [31](#)
- begin
 - Gino::DbManager, [158](#)
 - Gino::Plugin::mysql, [397](#)
 - Gino::Plugin::sqlsrv, [504](#)
- binding
 - Gino::App::Auth::Ldap, [330](#)
- blackThresholdImage
 - Gino::GImage, [265](#)
- BooleanField, [111](#)
- box
 - Gino::App::Page::page, [415](#)
- breadCrumbs
 - Gino::App::Menu::menu, [360](#)
- browsers
 - Gino, [31](#)
- CACHE_DIR
 - Gino, [53](#)
- CLASSES_DIR
 - Gino, [53](#)
- CONTENT_DIR
 - Gino, [53](#)
- CONTENT_WWW
 - Gino, [53](#)
- CONTROLLER_CLASS_PREFIX
 - Gino, [53](#)
- CORE_CLASS_PREFIX
 - Gino, [53](#)
- CSS_DIR
 - Gino, [53](#)
- CSS_WWW
 - Gino, [54](#)
- Cache, [114](#)
- canBeOrdered
 - Gino::Field, [205](#)
 - Gino::TextField, [541](#)
- Captcha, [117](#)
- captcha
 - Gino::Form, [235](#)
- ccheckbox
 - Gino::Form, [235](#)
- cellCtrl
 - Gino::Template, [535](#)
- cfile
 - Gino::Form, [236](#)
- changePassword
 - Gino::App::Auth::auth, [105](#)
- CharField, [120](#)
- check
 - Gino::Captcha, [118](#)
- checkAuthenticationUser
 - Gino::App::Auth::auth, [105](#)
- checkCaptcha
 - Gino::Form, [237](#)
- checkDefaultCaptcha
 - Gino::Form, [237](#)
- checkEmail
 - Gino, [31](#)
 - Gino::App::Auth::User, [552](#)
- checkFilename
 - Gino::FileField, [217](#)
 - Gino::Form, [237](#)
- checkGETKey
 - Gino::Http::Request, [460](#)
- checkIsConstraint
 - Gino::Model, [376](#)
- checkName
 - Gino::DirectoryField, [172](#)
- checkOptions
 - Gino::App::SearchSite::searchSite, [483](#)
- checkOutputFunctionPermissions
 - Gino::Document, [176](#)
- checkPOSTKey
 - Gino::Http::Request, [460](#)
- checkPassword
 - Gino::App::Auth::User, [552](#)
- checkReCaptcha
 - Gino::Form, [238](#)
- checkRequirements
 - Gino::Captcha, [118](#)
- checkSlug
 - Gino::App::Page::page, [415](#)
- checkUsername
 - Gino::App::Auth::User, [553](#)
 - Gino::App::Auth::auth, [106](#)
- checkValidUsername

- Gino::App::Auth::Ldap, 330
- checkbox
 - Gino::Form, 236
 - Gino::InputForm, 297
- choiceLanguage
 - Gino::App::Language::language, 318
- cinput
 - Gino::Form, 238
- cinput_date
 - Gino::Form, 238
- class.Loader.php
 - __autoload, 602
- className
 - Gino::App::Module::ModuleInstance, 393
 - Gino::App::SysClass::ModuleApp, 390
- classNameNs
 - Gino::App::Module::ModuleInstance, 393
 - Gino::App::SysClass::ModuleApp, 391
- clean
 - Gino::AdminTable, 78
 - Gino::DateField, 148
 - Gino::DatetimeField, 151
 - Gino::DirectoryField, 172
 - Gino::EmailField, 181
 - Gino::Field, 205
 - Gino::FileField, 217
 - Gino::FloatField, 226
 - Gino::ManyToManyField, 346
 - Gino::ManyToManyInlineField, 351
 - Gino::ManyToManyThroughField, 354
 - Gino::TagField, 529
 - Gino::TextField, 541
 - Gino::TimeField, 545
- clean_sequence
 - Gino, 32
- cleanVar
 - Gino, 32
- cleanVarEditor
 - Gino, 33
- clearSearchString
 - Gino::Search, 480
- cloneImage
 - Gino::GImage, 266
- close
 - Gino::Form, 239
- closeConnection
 - Gino::DbManager, 158
 - Gino::Plugin::mysql, 398
 - Gino::Plugin::sqlsrv, 504
- code
 - Gino::App::Language::Lang, 315
- codeDb
 - Gino, 33
- codeInput
 - Gino, 33
- codeMessages
 - Gino::Error, 186
- codeParser
 - Gino, 33
- codeToDB
 - Gino, 34
- columnHasValue
 - Gino::DbManager, 158
 - Gino::Plugin::mysql, 398
 - Gino::Plugin::sqlsrv, 504
- commentNotification
 - Gino::App::Page::page, 415
- commit
 - Gino::DbManager, 158
 - Gino::Plugin::mysql, 398
 - Gino::Plugin::sqlsrv, 504
- Compressor, 124
- concat
 - Gino::DbManager, 158
 - Gino::Plugin::mysql, 398
 - Gino::Plugin::sqlsrv, 505
- Conf, 128
- configuration.php, 589
 - ADMINS, 589
 - DB_CHARSET, 589
 - DB_DBNAME, 590
 - DB_HOST, 590
 - DB_PASSWORD, 590
 - DB_PORT, 590
 - DB_SCHEMA, 590
 - DB_USER, 590
 - DBMS, 590
 - DEBUG, 590
 - LDAP_AUTH, 590
 - MAX_FILE_SIZE, 590
 - SESSION_NAME, 590
- conformPath
 - Gino::FileField, 218
- conformType
 - Gino::DbManager, 159
 - Gino::Plugin::mysql, 398
 - Gino::Plugin::sqlsrv, 505
- connection
 - Gino::App::Auth::Ldap, 330
- Controller, 130
- convertRelativePath
 - Gino::Compressor, 125
- convertToDatabase
 - Gino, 34
- convertToHtml
 - Gino, 34
- Core, 136
- countEqualName
 - Gino::Form, 239
- countryCodes
 - Gino::App::Language::Lang, 315
- cradio
 - Gino::Form, 239
- createImage
 - Gino::Form, 240
- createMdlFolders

- Gino::App::Module::module, 386
- createTemplate
 - Gino::Template, 535
- crop
 - Gino::GImage, 266
- cropCenter
 - Gino::GImage, 266
- cropEntropy
 - Gino::GImage, 266
- croplImage
 - Gino::GImage, 267
- croplImageEntropy
 - Gino::GImage, 267
- cryptMethod
 - Gino, 35
- cselect
 - Gino::Form, 240
- Css, 139
- cssFileName
 - Gino::Css, 142
- cssList
 - Gino::App::Layout::layout, 325
- ctextarea
 - Gino::Form, 241
- cutHtmlText
 - Gino, 35
- cutString
 - Gino, 35
- DB_CHARSET
 - configuration.php, 589
- DB_DBNAME
 - configuration.php, 590
- DB_HOST
 - configuration.php, 590
- DB_PASSWORD
 - configuration.php, 590
- DB_PORT
 - configuration.php, 590
- DB_SCHEMA
 - configuration.php, 590
- DB_USER
 - configuration.php, 590
- DBMS
 - configuration.php, 590
- DEBUG
 - configuration.php, 590
- DataCache, 144
- dataType
 - Gino::Model, 376
- date
 - Gino::InputForm, 297
- dateDiff
 - Gino, 36
- DateField, 146
- dateToDbDate
 - Gino, 37
- DatetimeField, 149
- Db, 153
- dbDateToDate
 - Gino, 37
- dbDatetimeToDate
 - Gino, 37
- dbDatetimeToTime
 - Gino, 37
- DbManager, 154
- dbNumberToNumber
 - Gino, 38
- dbTimeToTime
 - Gino, 38
- decode_params
 - Gino, 38
- defaultCaptcha
 - Gino::Form, 241
- defaultName
 - Gino::DirectoryField, 172
- delete
 - Gino::App::Attachment::AttachmentCtg, 94
 - Gino::App::Auth::User, 553
 - Gino::App::Page::PageEntry, 432
 - Gino::Cache, 115
 - Gino::DbManager, 159
 - Gino::DirectoryField, 172
 - Gino::FileField, 218
 - Gino::ImageField, 288
 - Gino::Model, 376
 - Gino::Plugin::mysql, 399
 - Gino::Plugin::sqlsrv, 506
- deleteBlocks
 - Gino::Template, 535
- deleteDbData
 - Gino::Model, 376
- deleteFileDir
 - Gino, 38
- deleteFromCtg
 - Gino::App::Attachment::AttachmentItem, 98
- deleteFromEntry
 - Gino::App::Page::PageComment, 427
- deleteInstance
 - Gino::App::Menu::menu, 360
 - Gino::App::PhpModuleView::phpModuleView, 449
 - Gino::Controller, 133
 - Gino::Model, 377
- deleteInstanceVoices
 - Gino::App::Menu::MenuVoice, 366
- deleteModuleInstance
 - Gino::App::Module::module, 386
- deleteMoreInfo
 - Gino::App::Auth::User, 553
- deleteTranslations
 - Gino::Translation, 547
- deleteVoice
 - Gino::App::Menu::MenuVoice, 366
- deletem2m
 - Gino::Model, 377
- deletem2mthrough
 - Gino::Model, 377

- deletem2mthroughField
 - Gino::Model, 377
- destroy
 - Gino::Session, 488
- detectCodes
 - Gino::Locale, 338
- detectMobile
 - Gino::Core, 138
- dimensionFile
 - Gino::Form, 242
- dirUpload
 - Gino::Form, 242
- DirectoryField, 169
- distinct
 - Gino::DbManager, 160
 - Gino::Plugin::mysql, 399
 - Gino::Plugin::sqlsrv, 506
- Document, 174
- download
 - Gino, 39
- downloader
 - Gino::App::Attachment::attachment, 90
- drop
 - Gino::DbManager, 160
 - Gino::Plugin::mysql, 399
 - Gino::Plugin::sqlsrv, 506
- dump
 - Gino::AdminTable, 78
 - Gino::DbManager, 160
 - Gino::Plugin::mysql, 399
 - Gino::Plugin::sqlsrv, 506
- dumpDatabase
 - Gino::DbManager, 161
 - Gino::Plugin::mysql, 399
 - Gino::Plugin::sqlsrv, 506
- EVT_NAME
 - Gino::Router, 479
- editUrl
 - Gino::AdminTable, 79
- editableField
 - Gino::Options, 406
- editor
 - Gino::InputForm, 298
- editorAttachmentList
 - Gino::App::Attachment::attachment, 90
- editorHtml
 - Gino::Form, 242
- editorList
 - Gino::App::Attachment::attachment, 91
- else
 - global.php, 636
- email
 - Gino::InputForm, 298
- EmailField, 179
- emailPolicy
 - Gino, 39
- emit
 - Gino::EventDispatcher, 188
- enabledPng
 - Gino, 39
- enabledZip
 - Gino, 39
- enclosedField
 - Gino, 39
- encode_params
 - Gino, 40
- EnumField, 182
- Error, 185
- errorMessage
 - Gino::Error, 186
- errorMessages
 - Gino::Document, 176
- escapeString
 - Gino::DbManager, 161
 - Gino::Plugin::mysql, 399
 - Gino::Plugin::sqlsrv, 506
- EventDispatcher, 187
- Exception403, 189
- Exception404, 191
- Exception500, 193
- exceptionReportAdmins
 - Gino::Logger, 343
- execQuery
 - Gino::Plugin::mysql, 400
 - Gino::Plugin::sqlsrv, 506
- explanationTemplate
 - Gino::App::Page::page, 415
- Export, 194
- export
 - Gino::App::Statistics::statistics, 515
- exportCsv
 - Gino::Export, 196
- exportData
 - Gino::Export, 196
- extension
 - Gino, 40
- extensionFile
 - Gino, 40
- FIELDS_DIR
 - Gino, 54
- FLAG_PREFIX
 - Gino::App::Language::language, 320
- FLAG_SUFFIX
 - Gino::App::Language::language, 320
- FONTS_DIR
 - Gino, 54
- fcktextarea
 - Gino::Form, 242
- feedRSS
 - Gino::App::Page::page, 415
- Field, 201
- fieldInformations
 - Gino::DbManager, 161
 - Gino::Plugin::mysql, 400
 - Gino::Plugin::sqlsrv, 507
- fieldLabel

- Gino::Model, [377](#)
- file
 - Gino::InputForm, [298](#)
- FileField, [214](#)
- fileName
 - Gino::Frontend, [259](#)
- filePath
 - Gino::Compressor, [126](#)
- filterWhereClause
 - Gino::CharField, [122](#)
 - Gino::DateField, [148](#)
 - Gino::DatetimeField, [151](#)
 - Gino::Field, [206](#)
 - Gino::ManyToManyField, [347](#)
 - Gino::ManyToManyInlineField, [351](#)
 - Gino::ManyToManyThroughField, [355](#)
 - Gino::TextField, [542](#)
- FloatField, [224](#)
- footLine
 - Gino::Document, [176](#)
- ForeignKeyField, [228](#)
- Form, [231](#)
- form
 - Gino::App::SearchSite::searchSite, [483](#)
- formActivateModule
 - Gino::App::Module::module, [386](#)
- formActivateSysClass
 - Gino::App::SysClass::sysClass, [520](#)
- formBlock
 - Gino::Template, [535](#)
- formComment
 - Gino::App::Page::page, [417](#)
- formCopyTemplate
 - Gino::Template, [536](#)
- formCssLayout
 - Gino::Css, [142](#)
- formData
 - Gino::Template, [536](#)
- formDelCssLayout
 - Gino::Css, [142](#)
- formDelSkin
 - Gino::Skin, [494](#)
- formDelTemplate
 - Gino::Template, [536](#)
- formEditSysClass
 - Gino::App::SysClass::sysClass, [520](#)
- formElement
 - Gino::BooleanField, [112](#)
 - Gino::CharField, [122](#)
 - Gino::DateField, [148](#)
 - Gino::DatetimeField, [151](#)
 - Gino::EnumField, [183](#)
 - Gino::Field, [206](#)
 - Gino::FileField, [218](#)
 - Gino::FloatField, [227](#)
 - Gino::ForeignKeyField, [230](#)
 - Gino::ImageField, [288](#)
 - Gino::IntegerField, [306](#)
 - Gino::ManyToManyField, [347](#)
 - Gino::ManyToManyInlineField, [351](#)
 - Gino::ManyToManyThroughField, [355](#)
 - Gino::SlugField, [498](#)
 - Gino::TagField, [529](#)
 - Gino::TextField, [542](#)
 - Gino::TimeField, [545](#)
 - Gino::YearField, [562](#)
- formElementWidget
 - Gino::Field, [207](#)
- formFieldTranslation
 - Gino::Form, [243](#)
- formFiles
 - Gino::App::Layout::layout, [325](#)
- formFilters
 - Gino::AdminTable, [79](#)
- formFiltersAdd
 - Gino::AdminTable, [79](#)
- formFreeTemplate
 - Gino::Template, [536](#)
- formGroup
 - Gino::App::Auth::auth, [106](#)
- formInsertSysClass
 - Gino::App::SysClass::sysClass, [520](#)
- formManualSysClass
 - Gino::App::SysClass::sysClass, [520](#)
- formMenuVoice
 - Gino::App::Menu::menu, [360](#)
- formModule
 - Gino::App::Module::module, [388](#)
- formModuleFile
 - Gino::Frontend, [259](#)
- formOutline
 - Gino::Template, [536](#)
- formPassword
 - Gino::App::Auth::User, [554](#)
- formPermission
 - Gino::App::Auth::auth, [106](#)
- formPhpModule
 - Gino::App::PhpModuleView::PhpModule, [446](#)
- formRemoveModule
 - Gino::App::Module::module, [388](#)
- formRemoveSysClass
 - Gino::App::SysClass::sysClass, [521](#)
- formSkin
 - Gino::Skin, [494](#)
- formTemplate
 - Gino::Template, [536](#)
- formTranslation
 - Gino::Translation, [547](#)
- formUpgradeSysClass
 - Gino::App::SysClass::sysClass, [521](#)
- formVoice
 - Gino::App::Menu::MenuVoice, [367](#)
- freeresult
 - Gino::DbManager, [161](#)
 - Gino::Plugin::mysql, [400](#)
 - Gino::Plugin::sqlsrv, [507](#)

- Frontend, [257](#)
- FuncTest, [261](#)
- GIMAGE_DIR
 - Gino, [54](#)
- GImage, [263](#)
- gOpt
 - Gino, [41](#)
- GRAPHICS_DIR
 - Gino, [54](#)
- GTag, [281](#)
- generateFormToken
 - Gino::Form, [244](#)
- generatePassword
 - Gino::App::Auth::User, [554](#)
- generateStrings
 - Gino::Captcha, [119](#)
- get
 - Gino::App::Menu::MenuVoice, [367](#)
 - Gino::App::Page::PageEntry, [432](#)
 - Gino::DataCache, [145](#)
 - Gino::JSMin, [311](#)
 - Gino::Locale, [338](#)
- get_app_name_class_ns
 - global.php, [635](#)
- get_app_namespace
 - global.php, [635](#)
- get_browser_info
 - Gino, [40](#)
- get_languages
 - Gino::Locale, [338](#)
- get_model_app_name_class_ns
 - global.php, [636](#)
- get_name_class
 - global.php, [636](#)
- getAddPath
 - Gino::App::Auth::auth, [106](#)
 - Gino::App::Page::page, [417](#)
 - Gino::FileField, [218](#)
- getAll
 - Gino::Css, [142](#)
- getAllTags
 - Gino::GTag, [282](#)
- getAutoIncrement
 - Gino::Field, [208](#)
- getAutoNow
 - Gino::DatetimeField, [151](#)
- getAutoNowAdd
 - Gino::DatetimeField, [152](#)
- getBaseAbsPath
 - Gino::Controller, [133](#)
- getBasePath
 - Gino::App::Auth::auth, [107](#)
 - Gino::App::Page::page, [417](#)
 - Gino::Controller, [133](#)
- getCheckConstraint
 - Gino::Plugin::sqlsrv, [507](#)
- getCheckType
 - Gino::FileField, [218](#)
- getCheckUser
 - Gino::App::Auth::auth, [107](#)
- getChildren
 - Gino::App::Menu::MenuVoice, [367](#)
- getClassElements
 - Gino::App::Auth::auth, [107](#)
 - Gino::App::Graphics::graphics, [274](#)
 - Gino::App::Menu::menu, [360](#)
 - Gino::App::Page::page, [417](#)
 - Gino::App::PhpModuleView::phpModuleView, [449](#)
 - Gino::App::SearchSite::searchSite, [484](#)
 - Gino::Controller, [133](#)
- getClassName
 - Gino::Controller, [133](#)
- getConstraintType
 - Gino::Plugin::sqlsrv, [508](#)
- getContent
 - Gino::Http::Response, [463](#)
- getContentTags
 - Gino::GTag, [282](#)
- getController
 - Gino::ManyToManyThroughField, [355](#)
 - Gino::Model, [378](#)
- getCount
 - Gino::App::Page::PageEntry, [433](#)
- getCountForUser
 - Gino::App::Statistics::LogAccess, [341](#)
- getCountFromEntry
 - Gino::App::Page::PageComment, [427](#)
- getCurrentPage
 - Gino::Paginator, [436](#)
- getData
 - Gino::Export, [197](#)
- getDataDir
 - Gino::App::Attachment::attachment, [91](#)
- getDataType
 - Gino::Plugin::sqlsrv, [508](#)
- getDataWWW
 - Gino::App::Attachment::attachment, [91](#)
- getDateDiff
 - Gino, [41](#)
- getDecimalDigits
 - Gino::FloatField, [227](#)
- getDefault
 - Gino::BooleanField, [113](#)
 - Gino::EnumField, [184](#)
 - Gino::Field, [208](#)
- getDirectory
 - Gino::FileField, [219](#)
- getEntropy
 - Gino::GImage, [267](#)
- getEnum
 - Gino::BooleanField, [113](#)
 - Gino::EnumField, [184](#)
 - Gino::ForeignKeyField, [230](#)
 - Gino::ManyToManyInlineField, [352](#)
- getErrorMessage
 - Gino::Error, [186](#)

- getExtensions
 - Gino::FileField, [219](#)
- getFieldFromId
 - Gino::DbManager, [162](#)
 - Gino::Plugin::mysql, [400](#)
 - Gino::Plugin::sqlsrv, [508](#)
- getFieldLength
 - Gino::Plugin::sqlsrv, [508](#)
- getFieldsName
 - Gino::DbManager, [162](#)
 - Gino::Plugin::mysql, [400](#)
 - Gino::Plugin::sqlsrv, [509](#)
- getFilename
 - Gino::Cache, [116](#)
- getForMulticheck
 - Gino::App::Auth::Permission, [443](#)
- getForSelect
 - Gino::App::Attachment::AttachmentCtg, [94](#)
- getFromFilename
 - Gino::Css, [142](#)
- getFromFullCode
 - Gino::App::Auth::Permission, [443](#)
- getFromModuleApp
 - Gino::App::Module::ModuleInstance, [393](#)
- getFromName
 - Gino::App::Module::ModuleInstance, [393](#)
 - Gino::App::SysClass::ModuleApp, [391](#)
- getFromSlug
 - Gino::App::Page::PageEntry, [433](#)
 - Gino::Model, [378](#)
- getFromUserPwd
 - Gino::App::Auth::User, [554](#)
- getGroupedPermissions
 - Gino::App::Auth::Permission, [443](#)
- getHeadFields
 - Gino::Export, [197](#)
- getHeight
 - Gino::GImage, [267](#)
- getHeighth
 - Gino::ImageField, [288](#)
- getImageHistogram
 - Gino::GImage, [268](#)
- getInfoQuery
 - Gino::DbManager, [162](#)
 - Gino::Plugin::mysql, [400](#)
 - Gino::Plugin::sqlsrv, [509](#)
- getInformationKey
 - Gino::Plugin::sqlsrv, [509](#)
- getInstance
 - Gino::Controller, [133](#)
- getInstanceName
 - Gino::Controller, [134](#)
- getIntDigits
 - Gino::FloatField, [227](#)
- getJoinTable
 - Gino::ManyToManyField, [347](#)
- getJoinTableId
 - Gino::ManyToManyField, [347](#)
- getJoinTableM2mId
 - Gino::ManyToManyField, [348](#)
- getKeywords
 - Gino::Search, [480](#)
- getLabel
 - Gino::Field, [208](#)
- getLastForUser
 - Gino::App::Statistics::LogAccess, [341](#)
- getLenght
 - Gino::Field, [209](#)
- getList
 - Gino::App::Auth::Group, [280](#)
 - Gino::App::Auth::Permission, [443](#)
- getM2m
 - Gino::ManyToManyThroughField, [355](#)
- getMainLang
 - Gino::App::Language::Lang, [316](#)
- getMaxFileSize
 - Gino::FileField, [219](#)
- getMergeValue
 - Gino::App::Auth::Group, [280](#)
 - Gino::App::Auth::User, [555](#)
- getModelLabel
 - Gino::Model, [378](#)
- getModelTableId
 - Gino::ManyToManyThroughField, [356](#)
- getName
 - Gino::Field, [209](#)
- getNumRecords
 - Gino::DbManager, [163](#)
 - Gino::Plugin::mysql, [401](#)
 - Gino::Plugin::sqlsrv, [509](#)
- getPath
 - Gino::DirectoryField, [173](#)
 - Gino::FileField, [219](#)
 - Gino::GImage, [268](#)
- getPermissions
 - Gino::App::Auth::Group, [281](#)
 - Gino::App::Auth::User, [555](#)
- getPrefix
 - Gino::DirectoryField, [173](#)
 - Gino::FileField, [219](#)
- getPrefixFile
 - Gino::ImageField, [288](#)
- getPrefixThumb
 - Gino::ImageField, [288](#)
- getPrimaryKey
 - Gino::Field, [209](#)
- getRelatedContents
 - Gino::GTag, [282](#)
- getRequired
 - Gino::Field, [209](#)
- getResize
 - Gino::ImageField, [288](#)
- getResource
 - Gino::GImage, [268](#)
- getSearchResults
 - Gino::Search, [480](#)

- getSelectOptionsFromObjects
 - Gino::Model, 378
- getSelectedVoice
 - Gino::App::Menu::MenuVoice, 367
- getSkin
 - Gino::Document, 176
 - Gino::Skin, 494
- getStructure
 - Gino::Model, 378
- getTable
 - Gino::ManyToManyThroughField, 356
 - Gino::Model, 378
- getTableStructure
 - Gino::DbManager, 163
 - Gino::Plugin::mysql, 401
 - Gino::Plugin::sqlsrv, 510
- getTagsHistogram
 - Gino::GTag, 284
- getThumb
 - Gino::ImageField, 289
- getThumbFromKey
 - Gino::GImage, 268
- getThumbHeighth
 - Gino::ImageField, 289
- getThumbWidth
 - Gino::ImageField, 289
- getTree
 - Gino::App::Menu::menu, 360
 - Gino::App::Page::PageComment, 427
- getTrnsl
 - Gino::CharField, 122
 - Gino::TextField, 542
- getTypesAllowed
 - Gino::FileField, 219
- getUniqueKey
 - Gino::Field, 209
- getUrl
 - Gino::App::Page::PageEntry, 434
- getUrlPage
 - Gino::App::Page::page, 417
- getUsersFromPermissions
 - Gino::App::Auth::User, 555
- getValue
 - Gino::Field, 209
- getValueType
 - Gino::Field, 209
- getWidget
 - Gino::Field, 210
- getWidth
 - Gino::GImage, 268
 - Gino::ImageField, 289
- getlastid
 - Gino::DbManager, 162
 - Gino::Plugin::mysql, 401
 - Gino::Plugin::sqlsrv, 509
- Gino, 23
 - APP_DIR, 53
 - absolutePath, 31
 - arrayToObject, 31
 - attributeVar, 31
 - baseFileName, 31
 - browsers, 31
 - CACHE_DIR, 53
 - CLASSES_DIR, 53
 - CONTENT_DIR, 53
 - CONTENT_WWW, 53
 - CONTROLLER_CLASS_PREFIX, 53
 - CORE_CLASS_PREFIX, 53
 - CSS_DIR, 53
 - CSS_WWW, 54
 - checkEmail, 31
 - clean_sequence, 32
 - cleanVar, 32
 - cleanVarEditor, 33
 - codeDb, 33
 - codeInput, 33
 - codeParser, 33
 - codeToDB, 34
 - convertToDatabase, 34
 - convertToHtml, 34
 - cryptMethod, 35
 - cutHtmlText, 35
 - cutString, 35
 - dateDiff, 36
 - dateToDbDate, 37
 - dbDateToDate, 37
 - dbDatetimeToDate, 37
 - dbDatetimeToTime, 37
 - dbNumberToNumber, 38
 - dbTimeToTime, 38
 - decode_params, 38
 - deleteFileDir, 38
 - download, 39
 - emailPolicy, 39
 - enabledPng, 39
 - enabledZip, 39
 - enclosedField, 39
 - encode_params, 40
 - extension, 40
 - extensionFile, 40
 - FIELDS_DIR, 54
 - FONTS_DIR, 54
 - GIMAGE_DIR, 54
 - gOpt, 41
 - GRAPHICS_DIR, 54
 - get_browser_info, 40
 - getDateDiff, 41
 - HOME_FILE, 54
 - htmlChars, 41
 - htmlCharsText, 42
 - htmlInput, 42
 - htmlInputEditor, 42
 - htmlToEntities, 42
 - icon, 43
 - is_browser, 45
 - is_browser_platform, 45

- is_browser_version, [45](#)
- is_robot, [45](#)
- isNumeric, [45](#)
- isValid, [46](#)
- jsVar, [46](#)
- LIB_DIR, [54](#)
- listProv, [46](#)
- MODEL_CLASS_PREFIX, [54](#)
- numberToDB, [46](#)
- OS, [54](#)
- obj_serialize, [47](#)
- obj_unserialize, [47](#)
- PLUGIN_DIR, [54](#)
- platforms, [47](#)
- preCodeParser, [47](#)
- relativePath, [48](#)
- remove_attributes, [48](#)
- replaceChar, [48](#)
- robots, [48](#)
- SITE_APP, [54](#)
- SITE_CUSTOM_CKEDITOR, [54](#)
- SITE_GRAPHICS, [54](#)
- SITE_IMG, [55](#)
- SITE_JS, [55](#)
- SITE_LIB, [55](#)
- SITE_ROOT, [55](#)
- SITE_WWW, [55](#)
- searchNameFile, [48](#)
- share, [49](#)
- shareAll, [49](#)
- slimboxReplace, [49](#)
- strip_invisible_tags, [49](#)
- strip_selected_tags, [50](#)
- strip_tags_attributes, [50](#)
- stripAll, [50](#)
- stripEditor, [50](#)
- TBL_GROUP, [55](#)
- TBL_GROUP_PERMISSION, [55](#)
- TBL_LANGUAGE, [55](#)
- TBL_LOG_ACCESS, [55](#)
- TBL_MODULE, [55](#)
- TBL_MODULE_APP, [55](#)
- TBL_NATION, [55](#)
- TBL_PERMISSION, [56](#)
- TBL_SYS_CONF, [56](#)
- TBL_TRANSLATION, [56](#)
- TBL_USER, [56](#)
- TBL_USER_ADD, [56](#)
- TBL_USER_GROUP, [56](#)
- TBL_USER_PERMISSION, [56](#)
- TMP_DIR, [56](#)
- TPL_DIR, [56](#)
- timeDiff, [52](#)
- timeToDbTime, [52](#)
- traslitterazione, [52](#)
- utf8_urldecode, [52](#)
- VIEWS_DIR, [56](#)
- verifyExtension, [53](#)
- Gino::Access
 - \$_db, [70](#)
 - \$_home, [70](#)
 - \$_session, [70](#)
 - __construct, [68](#)
 - Authentication, [68](#)
 - AuthenticationMethod, [69](#)
 - logAccess, [69](#)
 - loginError, [69](#)
 - loginSuccess, [69](#)
 - requireAdmin, [70](#)
 - requirePerm, [70](#)
- Gino::AdminTable
 - \$_allow_insertion, [85](#)
 - \$_controller, [85](#)
 - \$_db, [85](#)
 - \$_delete_deny, [85](#)
 - \$_edit_deny, [85](#)
 - \$_filter_add, [86](#)
 - \$_filter_fields, [86](#)
 - \$_filter_join, [86](#)
 - \$_form, [86](#)
 - \$_hidden, [86](#)
 - \$_ifp, [86](#)
 - \$_list_display, [86](#)
 - \$_list_remove, [86](#)
 - \$_registry, [86](#)
 - \$_request, [86](#)
 - \$_session, [86](#)
 - \$_view, [86](#)
 - __construct, [74](#)
 - addWhereClauses, [75](#)
 - addWhereExtra, [75](#)
 - addWhereJoin, [75](#)
 - adminDelete, [75](#)
 - adminForm, [76](#)
 - adminList, [76](#)
 - backOffice, [78](#)
 - clean, [78](#)
 - dump, [78](#)
 - editUrl, [79](#)
 - formFilters, [79](#)
 - formFiltersAdd, [79](#)
 - hidden, [80](#)
 - m2mthroughAction, [80](#)
 - modelAction, [80](#)
 - modelForm, [82](#)
 - permission, [84](#)
 - readFile, [84](#)
 - restore, [84](#)
 - setSessionSearch, [85](#)
 - setSessionSearchAdd, [85](#)
- Gino::App::Attachment::AttachmentCtg
 - \$table, [95](#)
 - __construct, [94](#)
 - __toString, [94](#)
 - delete, [94](#)
 - getForSelect, [94](#)

- path, 95
- structure, 95
- Gino::App::Attachment::AttachmentItem
 - \$_doc_extension, 99
 - \$_img_extension, 99
 - \$_pdf_extension, 99
 - \$_xls_extension, 100
 - \$table, 100
 - __construct, 97
 - __toString, 98
 - deleteFromCtg, 98
 - path, 98
 - pathDownload, 98
 - pathView, 98
 - previewAdminList, 99
 - previewLink, 99
 - structure, 99
 - type, 99
- Gino::App::Attachment::AttachmentItemAdminTable
 - modelAction, 101
- Gino::App::Attachment::attachment
 - \$_title, 92
 - __construct, 90
 - downloader, 90
 - editorAttachmentList, 90
 - editorList, 91
 - getDataDir, 91
 - getDataWWW, 91
 - jsonImageList, 91
 - manageAttachment, 91
 - manageCategory, 92
 - manageItem, 92
- Gino::App::Auth
 - LDAP_APP_PASSWORD, 58
 - LDAP_APP_USERNAME, 58
 - LDAP_BASE_DN, 58
 - LDAP_DOMAIN, 58
 - LDAP_FILTER_SEARCH, 58
 - LDAP_HOST, 58
 - LDAP_PORT, 59
 - LDAP_PROTOCOL_VERSION, 59
 - LDAP_SEARCH_DN, 59
- Gino::App::Auth::AdminTable_AuthUser
 - modelAction, 87
- Gino::App::Auth::Group
 - \$table, 281
 - \$table_group_perm, 281
 - __construct, 280
 - __toString, 280
 - getList, 280
 - getMergeValue, 280
 - getPermissions, 281
 - setMergeValue, 281
- Gino::App::Auth::Ldap
 - \$_auth, 331
 - \$_debug, 331
 - \$_filter_search, 331
 - \$_justthese_search, 331
 - \$_ldap_base_dn, 331
 - \$_ldap_domain, 331
 - \$_ldap_ds, 331
 - \$_ldap_host, 331
 - \$_ldap_log, 331
 - \$_ldap_password, 331
 - \$_ldap_port, 331
 - \$_ldap_protocol_version, 331
 - \$_ldap_username, 331
 - \$_test_anonymous, 331
 - \$_test_search, 331
 - \$_user_name, 331
 - \$_user_password, 331
 - __construct, 329
 - authentication, 329
 - binding, 330
 - checkValidUsername, 330
 - connection, 330
 - printError, 330
 - search, 330
- Gino::App::Auth::Permission
 - \$table, 444
 - \$table_perm_user, 444
 - __construct, 442
 - __toString, 443
 - getForMulticheck, 443
 - getFromFullCode, 443
 - getGroupedPermissions, 443
 - getList, 443
 - structure, 444
- Gino::App::Auth::User
 - \$extension_media, 557
 - \$table, 557
 - \$table_groups, 557
 - \$table_more, 557
 - __construct, 551
 - __toString, 552
 - checkEmail, 552
 - checkPassword, 552
 - checkUsername, 553
 - delete, 553
 - deleteMoreInfo, 553
 - formPassword, 554
 - generatePassword, 554
 - getFromUserPwd, 554
 - getMergeValue, 555
 - getPermissions, 555
 - getUsersFromPermissions, 555
 - hasAdminPerm, 555
 - hasPerm, 556
 - savePassword, 556
 - setMergeValue, 556
 - setMoreInfo, 557
 - setPassword, 557
 - structure, 557
- Gino::App::Auth::auth
 - \$_aut_pwd, 110
 - \$_aut_pwd_length, 110

- [\\$_label_field1, 110](#)
- [\\$_label_field2, 110](#)
- [\\$_label_field3, 110](#)
- [\\$_ldap_auth, 110](#)
- [\\$_ldap_auth_only, 110](#)
- [\\$_ldap_auth_password, 110](#)
- [\\$_ldap_single_user, 110](#)
- [\\$_options, 110](#)
- [\\$_optionsLabels, 110](#)
- [\\$_other_field1, 110](#)
- [\\$_other_field2, 110](#)
- [\\$_other_field3, 110](#)
- [\\$_pwd_length_max, 110](#)
- [\\$_pwd_length_min, 110](#)
- [\\$_pwd_numeric_number, 110](#)
- [\\$_self_registration, 110](#)
- [\\$_self_registration_active, 110](#)
- [\\$_title, 110](#)
- [\\$_user_more, 110](#)
- [\\$_user_view, 110](#)
- [\\$_username_as_email, 110](#)
- [\\$_users_for_page, 110](#)
- [__construct, 104](#)
- [actionJoinGroupPermission, 104](#)
- [actionJoinUserPermission, 104](#)
- [changePassword, 105](#)
- [checkAuthenticationUser, 105](#)
- [checkUsername, 106](#)
- [formGroup, 106](#)
- [formPermission, 106](#)
- [getAddPath, 106](#)
- [getBasePath, 107](#)
- [getCheckUser, 107](#)
- [getClassElements, 107](#)
- [joinUserPermission, 108](#)
- [joingrouppermission, 107](#)
- [login, 108](#)
- [manageAuth, 108](#)
- [manageGroup, 108](#)
- [managePermission, 108](#)
- [manageUser, 109](#)
- [outputFunctions, 109](#)
- [passwordRules, 109](#)
- [returnJoinLink, 109](#)
- [Gino::App::Graphics::GraphicsItem](#)
 - [\\$_extension_img, 278](#)
 - [\\$table, 278](#)
 - [__construct, 278](#)
 - [__toString, 278](#)
 - [structure, 278](#)
- [Gino::App::Graphics::graphics](#)
 - [\\$_title, 276](#)
 - [__construct, 274](#)
 - [getClassElements, 274](#)
 - [isHeader, 274](#)
 - [manageGraphics, 274](#)
 - [outputFunctions, 274](#)
 - [printFooterAdhoc, 274](#)
 - [printFooterAdmin, 275](#)
 - [printFooterMobile, 275](#)
 - [printFooterPrivate, 275](#)
 - [printFooterPublic, 275](#)
 - [printHeaderAdhoc, 275](#)
 - [printHeaderAdmin, 275](#)
 - [printHeaderMobile, 275](#)
 - [printHeaderPrivate, 276](#)
 - [printHeaderPublic, 276](#)
 - [render, 276](#)
- [Gino::App::Index::index](#)
 - [__construct, 295](#)
 - [admin_page, 295](#)
 - [modulesManageArray, 295](#)
 - [outputFunctions, 295](#)
 - [sysModulesManageArray, 295](#)
- [Gino::App::Instruments::instruments](#)
 - [__construct, 304](#)
 - [links, 304](#)
 - [manageInstruments, 304](#)
 - [mimeTypes, 304](#)
- [Gino::App::Language::Lang](#)
 - [\\$table, 316](#)
 - [__construct, 315](#)
 - [__toString, 315](#)
 - [code, 315](#)
 - [countryCodes, 315](#)
 - [getMainLang, 316](#)
 - [languageCodes, 316](#)
 - [structure, 316](#)
- [Gino::App::Language::language](#)
 - [\\$_flag_language, 320](#)
 - [\\$_options, 320](#)
 - [\\$_optionsLabels, 320](#)
 - [\\$_title, 320](#)
 - [__construct, 318](#)
 - [choiceLanguage, 318](#)
 - [FLAG_PREFIX, 320](#)
 - [FLAG_SUFFIX, 320](#)
 - [manageLang, 318](#)
 - [manageLanguage, 320](#)
 - [outputFunctions, 320](#)
- [Gino::App::Layout::layout](#)
 - [__construct, 323](#)
 - [actionCss, 323](#)
 - [actionDelCss, 323](#)
 - [actionDelSkin, 324](#)
 - [actionDelTemplate, 324](#)
 - [actionFiles, 324](#)
 - [actionSkin, 325](#)
 - [actionTemplate, 325](#)
 - [cssList, 325](#)
 - [formFiles, 325](#)
 - [info, 326](#)
 - [manageCss, 326](#)
 - [manageLayout, 326](#)
 - [manageSkin, 326](#)
 - [manageTemplate, 326](#)

- manageView, [327](#)
- modulesCodeList, [327](#)
- modulesList, [327](#)
- skinList, [327](#)
- templateList, [327](#)
- viewList, [328](#)
- Gino::App::Menu::MenuVoice
 - \$_gform, [368](#)
 - \$_tbl_user_role, [368](#)
 - \$_tbl_voices, [368](#)
 - __construct, [366](#)
 - deleteInstanceVoices, [366](#)
 - deleteVoice, [366](#)
 - formVoice, [367](#)
 - get, [367](#)
 - getChildren, [367](#)
 - getSelectedVoice, [367](#)
 - initOrderList, [367](#)
 - updateOrderList, [368](#)
 - userCanSee, [368](#)
- Gino::App::Menu::menu
 - \$_cache, [364](#)
 - \$_ico_more, [364](#)
 - \$_menu_functions_list, [364](#)
 - \$_options, [364](#)
 - \$_optionsLabels, [364](#)
 - \$_tbl_opt, [364](#)
 - \$_title, [364](#)
 - __construct, [359](#)
 - actionDelMenuVoice, [359](#)
 - actionMenuVoice, [359](#)
 - actionUpdateOrder, [359](#)
 - breadCrumbs, [360](#)
 - deleteInstance, [360](#)
 - formMenuVoice, [360](#)
 - getClassElements, [360](#)
 - getTree, [360](#)
 - jsSearchModulesLib, [360](#)
 - jsSortLib, [361](#)
 - listMenu, [361](#)
 - manageDoc, [361](#)
 - outputFunctions, [361](#)
 - pathToSelectedVoice, [362](#)
 - printItemsClass, [362](#)
 - printItemsList, [362](#)
 - printItemsPage, [363](#)
 - render, [363](#)
 - renderMenuAdmin, [363](#)
 - searchModules, [363](#)
- Gino::App::Module::ModuleInstance
 - \$table, [394](#)
 - __construct, [393](#)
 - className, [393](#)
 - classNameNs, [393](#)
 - getFromModuleApp, [393](#)
 - getFromName, [393](#)
 - moduleApp, [393](#)
- Gino::App::Module::module
 - __construct, [384](#)
 - actionEditModule, [384](#)
 - actionEditModuleActive, [385](#)
 - actionInsertModule, [385](#)
 - actionModule, [385](#)
 - actionRemoveModule, [386](#)
 - createMdlFolders, [386](#)
 - deleteModuleInstance, [386](#)
 - formActivateModule, [386](#)
 - formModule, [388](#)
 - formRemoveModule, [388](#)
 - listModule, [388](#)
 - manageModule, [388](#)
- Gino::App::Page::PageCategory
 - \$table, [425](#)
 - __construct, [424](#)
 - __toString, [425](#)
- Gino::App::Page::PageComment
 - \$table, [429](#)
 - __construct, [427](#)
 - __toString, [427](#)
 - deleteFromEntry, [427](#)
 - getCountFromEntry, [427](#)
 - getTree, [427](#)
 - notifyComment, [429](#)
 - save, [429](#)
 - structure, [429](#)
- Gino::App::Page::PageEntry
 - \$_extension_img, [435](#)
 - \$table, [435](#)
 - __construct, [431](#)
 - __toString, [432](#)
 - accessWhere, [432](#)
 - delete, [432](#)
 - get, [432](#)
 - getCount, [433](#)
 - getFromSlug, [433](#)
 - getUrl, [434](#)
 - imgPath, [434](#)
 - save, [434](#)
 - structure, [434](#)
- Gino::App::Page::page
 - \$_box_tpl_code, [422](#)
 - \$_comment_moderation, [422](#)
 - \$_comment_notification, [422](#)
 - \$_entry_tpl_code, [422](#)
 - \$_newsletter_entries_number, [422](#)
 - \$_newsletter_tpl_code, [422](#)
 - \$_showcase_auto_interval, [422](#)
 - \$_showcase_auto_start, [423](#)
 - \$_showcase_number, [423](#)
 - \$_showcase_title, [423](#)
 - \$_showcase_tpl_code, [423](#)
 - \$_tbl_opt, [423](#)
 - __construct, [414](#)
 - accessPage, [414](#)
 - actionComment, [414](#)
 - box, [415](#)

- checkSlug, [415](#)
- commentNotification, [415](#)
- explanationTemplate, [415](#)
- feedRSS, [415](#)
- formComment, [417](#)
- getAddPath, [417](#)
- getBasePath, [417](#)
- getClassElements, [417](#)
- getUrlPage, [417](#)
- manageComment, [418](#)
- manageCtg, [418](#)
- manageEntry, [418](#)
- managePage, [418](#)
- outputFunctions, [418](#)
- parseTemplate, [419](#)
- permissions, [419](#)
- relatedContentsList, [419](#)
- replaceTplVar, [419](#)
- searchSite, [420](#)
- searchSiteResult, [420](#)
- showcase, [420](#)
- systemNewsletterList, [420](#)
- systemNewsletterRender, [420](#)
- view, [422](#)
- Gino::App::PhpModuleView::PhpModule
 - [\\$_interface](#), [447](#)
 - [\\$_tbl_data](#), [447](#)
 - [\\$table](#), [447](#)
 - [__construct](#), [445](#)
 - [actionPhpModule](#), [446](#)
 - [formPhpModule](#), [446](#)
 - [setContent](#), [446](#)
 - [setInstance](#), [446](#)
 - [structure](#), [447](#)
- Gino::App::PhpModuleView::phpModuleView
 - [\\$_blackList](#), [450](#)
 - [\\$_options](#), [450](#)
 - [\\$_optionsLabels](#), [450](#)
 - [\\$_tbl_opt](#), [450](#)
 - [\\$_title](#), [450](#)
 - [\\$_title_visible](#), [450](#)
 - [__construct](#), [449](#)
 - [deleteInstance](#), [449](#)
 - [getClassElements](#), [449](#)
 - [info](#), [449](#)
 - [manageDoc](#), [450](#)
 - [outputFunctions](#), [450](#)
 - [permissions](#), [450](#)
 - [viewList](#), [450](#)
- Gino::App::SearchSite::searchSite
 - [\\$_inst_md1](#), [485](#)
 - [\\$_options](#), [485](#)
 - [\\$_optionsLabels](#), [485](#)
 - [\\$_sys_md1](#), [485](#)
 - [\\$_title](#), [485](#)
 - [\\$_view_dir](#), [485](#)
 - [__construct](#), [483](#)
 - [checkOptions](#), [483](#)
 - [form](#), [483](#)
 - [getClassElements](#), [484](#)
 - [info](#), [484](#)
 - [manageSearchSite](#), [484](#)
 - [outputFunctions](#), [484](#)
 - [results](#), [484](#)
- Gino::App::Statistics::LogAccess
 - [\\$table](#), [342](#)
 - [__construct](#), [341](#)
 - [getCountForUser](#), [341](#)
 - [getLastForUser](#), [341](#)
- Gino::App::Statistics::statistics
 - [\\$_tbl_log_access](#), [516](#)
 - [__construct](#), [515](#)
 - [export](#), [515](#)
 - [info](#), [515](#)
 - [manageStatistics](#), [515](#)
 - [viewlogAccess](#), [515](#)
- Gino::App::SysClass::ModuleApp
 - [\\$table](#), [391](#)
 - [__construct](#), [390](#)
 - [__toString](#), [390](#)
 - [className](#), [390](#)
 - [classNameNs](#), [391](#)
 - [getFromName](#), [391](#)
- Gino::App::SysClass::sysClass
 - [\\$_title](#), [522](#)
 - [__construct](#), [518](#)
 - [actionEditSysClass](#), [518](#)
 - [actionEditSysClassActive](#), [518](#)
 - [actionInsertSysClass](#), [519](#)
 - [actionManualSysClass](#), [519](#)
 - [actionRemoveSysClass](#), [519](#)
 - [actionUpgradeSysClass](#), [520](#)
 - [formActivateSysClass](#), [520](#)
 - [formEditSysClass](#), [520](#)
 - [formInsertSysClass](#), [520](#)
 - [formManualSysClass](#), [520](#)
 - [formRemoveSysClass](#), [521](#)
 - [formUpgradeSysClass](#), [521](#)
 - [info](#), [521](#)
 - [manageSysClass](#), [521](#)
 - [PACKAGE_EXTENSION](#), [522](#)
 - [sysClassList](#), [521](#)
 - [upgradeFolders](#), [522](#)
- Gino::App::Sysconf::Conf
 - [\\$table](#), [129](#)
 - [__construct](#), [129](#)
 - [__toString](#), [129](#)
 - [structure](#), [129](#)
- Gino::App::Sysconf::sysconf
 - [__construct](#), [524](#)
 - [manageSysconf](#), [524](#)
- Gino::App::Sysfunc::sysfunc
 - [__construct](#), [526](#)
 - [page403](#), [526](#)
 - [page404](#), [526](#)
 - [page500](#), [527](#)

- Gino::BooleanField
 - `$_default`, 114
 - `$_enum`, 114
 - `__construct`, 112
 - `__toString`, 112
 - `formElement`, 112
 - `getDefault`, 113
 - `getEnum`, 113
 - `setDefault`, 113
 - `setEnum`, 113
- Gino::Cache
 - `$_ds`, 117
 - `$_enabled`, 117
 - `$_fld`, 117
 - `$_grp`, 117
 - `$_prefix`, 117
 - `$_registry`, 117
 - `$_tc`, 117
 - `$id`, 117
 - `__construct`, 115
 - `delete`, 115
 - `getFilename`, 116
 - `isCached`, 116
 - `read`, 116
 - `write`, 116
- Gino::Captcha
 - `$_allow_numbers`, 119
 - `$_font_file`, 119
 - `$_height`, 120
 - `$_letters`, 120
 - `$_name`, 120
 - `$_numbers`, 120
 - `$_width`, 120
 - `__construct`, 118
 - `check`, 118
 - `checkRequirements`, 118
 - `generateStrings`, 119
 - `hex2RGB`, 119
 - `render`, 119
- Gino::CharField
 - `$_trnsI`, 124
 - `__construct`, 122
 - `filterWhereClause`, 122
 - `formElement`, 122
 - `getTrnsI`, 122
 - `setTrnsI`, 122
- Gino::Compressor
 - `$_css`, 127
 - `$_js`, 127
 - `__construct`, 125
 - `addCss`, 125
 - `addJs`, 125
 - `convertRelativePath`, 125
 - `filePath`, 126
 - `mergeCss`, 126
 - `mergeJs`, 126
 - `minifyCss`, 126
 - `minifyJs`, 127
 - `moveCss`, 127
 - `shouldUpdate`, 127
- Gino::Controller
 - `$_access`, 136
 - `$_class_img`, 136
 - `$_class_name`, 136
 - `$_class_www`, 136
 - `$_data_dir`, 136
 - `$_data_www`, 136
 - `$_db`, 136
 - `$_home`, 136
 - `$_instance`, 136
 - `$_instance_label`, 136
 - `$_instance_name`, 136
 - `$_locale`, 136
 - `$_permissions`, 136
 - `$_plink`, 136
 - `$_registry`, 136
 - `$_session`, 136
 - `$_trd`, 136
 - `$_view_dir`, 136
 - `__call`, 132
 - `__construct`, 132
 - `deleteInstance`, 133
 - `getBaseAbsPath`, 133
 - `getBasePath`, 133
 - `getClassElements`, 133
 - `getClassName`, 133
 - `getInstance`, 133
 - `getInstanceName`, 134
 - `link`, 134
 - `linkAdmin`, 134
 - `manageFrontend`, 134
 - `manageOptions`, 134
 - `requirePerm`, 134
 - `setInstanceProperties`, 135
 - `setOption`, 135
 - `setPaths`, 135
 - `userHasPerm`, 135
- Gino::Core
 - `$_db`, 138
 - `$_registry`, 138
 - `__construct`, 137
 - `answer`, 138
 - `detectMobile`, 138
 - `initMobile`, 138
 - `initRegistry`, 138
- Gino::Css
 - `$_class`, 144
 - `$_css_list`, 144
 - `$_instance_class`, 144
 - `$_interface`, 144
 - `$_label`, 144
 - `$_mdlLink`, 144
 - `$_module`, 144
 - `$_name`, 144
 - `$_tbl_data`, 144
 - `$_tbl_module`, 144

- \$table, [144](#)
 - __construct, [141](#)
 - actionCssLayout, [141](#)
 - actionDelCssLayout, [141](#)
 - cssFileName, [142](#)
 - formCssLayout, [142](#)
 - formDelCssLayout, [142](#)
 - getAll, [142](#)
 - getFromFilename, [142](#)
 - layoutInfo, [142](#)
 - setFilename, [142](#)
- Gino::DataCache
 - __construct, [145](#)
 - get, [145](#)
 - save, [146](#)
- Gino::DateField
 - __construct, [147](#)
 - __toString, [148](#)
 - clean, [148](#)
 - filterWhereClause, [148](#)
 - formElement, [148](#)
- Gino::DatetimeField
 - \$_auto_now, [152](#)
 - \$_auto_now_add, [152](#)
 - __construct, [150](#)
 - __toString, [151](#)
 - clean, [151](#)
 - filterWhereClause, [151](#)
 - formElement, [151](#)
 - getAutoNow, [151](#)
 - getAutoNowAdd, [152](#)
 - setAutoNow, [152](#)
 - setAutoNowAdd, [152](#)
- Gino::Db
 - \$_db_charset, [154](#)
 - \$_db_dbname, [154](#)
 - \$_db_host, [154](#)
 - \$_db_pass, [154](#)
 - \$_db_schema, [154](#)
 - \$_db_user, [154](#)
 - instance, [154](#)
- Gino::DbManager
 - __construct, [157](#)
 - actionquery, [157](#)
 - affected, [157](#)
 - autoIncValue, [157](#)
 - begin, [158](#)
 - closeConnection, [158](#)
 - columnHasValue, [158](#)
 - commit, [158](#)
 - concat, [158](#)
 - conformType, [159](#)
 - delete, [159](#)
 - distinct, [160](#)
 - drop, [160](#)
 - dump, [160](#)
 - dumpDatabase, [161](#)
 - escapeString, [161](#)
 - fieldInformations, [161](#)
 - freeresult, [161](#)
 - getFieldFromId, [162](#)
 - getFieldsName, [162](#)
 - getInfoQuery, [162](#)
 - getNumRecords, [163](#)
 - getTableStructure, [163](#)
 - getlastid, [162](#)
 - insert, [164](#)
 - join, [165](#)
 - limit, [165](#)
 - multiActionquery, [165](#)
 - openConnection, [166](#)
 - query, [166](#)
 - restore, [166](#)
 - resultselect, [167](#)
 - rollback, [167](#)
 - select, [167](#)
 - selectquery, [168](#)
 - tableexists, [168](#)
 - union, [168](#)
 - update, [169](#)
- Gino::DirectoryField
 - \$_default_name, [174](#)
 - \$_path, [174](#)
 - \$_prefix, [174](#)
 - __construct, [171](#)
 - checkName, [172](#)
 - clean, [172](#)
 - defaultName, [172](#)
 - delete, [172](#)
 - getPath, [173](#)
 - getPrefix, [173](#)
 - setPath, [173](#)
 - setPrefix, [173](#)
 - validate, [173](#)
- Gino::Document
 - \$_registry, [179](#)
 - \$_request, [179](#)
 - \$_url_content, [179](#)
 - __construct, [175](#)
 - __invoke, [176](#)
 - checkOutputFunctionPermissions, [176](#)
 - errorMessages, [176](#)
 - footLine, [176](#)
 - getSkin, [176](#)
 - google_analytics, [176](#)
 - headLine, [176](#)
 - modClass, [177](#)
 - modPage, [177](#)
 - modUrl, [177](#)
 - parseModules, [177](#)
 - render, [178](#)
 - renderModule, [178](#)
 - renderNave, [178](#)
 - setHeadVariables, [178](#)
 - setIfEmpty, [179](#)
- Gino::EmailField

- __construct, 181
 - clean, 181
 - validate, 181
- Gino::EnumField
 - \$_default, 185
 - \$_enum, 185
 - __construct, 183
 - __toString, 183
 - formElement, 183
 - getDefault, 184
 - getEnum, 184
 - setDefault, 184
 - setEnum, 184
- Gino::Error
 - codeMessages, 186
 - errorMessage, 186
 - getErrorMessage, 186
 - warning, 186
- Gino::EventDispatcher
 - \$_emitter_listeners, 189
 - \$_listeners, 189
 - emit, 188
 - listen, 188
 - listenEmitter, 188
- Gino::Exception::Exception403
 - \$_redirect, 191
 - __construct, 190
 - httpResponse, 191
- Gino::Exception::Exception404
 - __construct, 192
 - httpResponse, 193
- Gino::Exception::Exception500
 - __construct, 194
 - httpResponse, 194
- Gino::Export
 - \$_data, 201
 - \$_fields, 201
 - \$_head, 201
 - \$_order, 201
 - \$_rids, 201
 - \$_s, 201
 - \$_table, 201
 - exportCsv, 196
 - exportData, 196
 - getData, 197
 - getHeadFields, 197
 - removeBOM, 197
 - setData, 197
 - setFields, 197
 - setHead, 198
 - setOrder, 198
 - setRids, 198
 - setSeparator, 198
 - setTable, 200
 - writeFile, 200
- Gino::Field
 - \$_auto_increment, 213
 - \$_default, 213
 - \$_default_widget, 213
 - \$_label, 213
 - \$_lenght, 213
 - \$_model, 214
 - \$_name, 214
 - \$_primary_key, 214
 - \$_required, 214
 - \$_table, 214
 - \$_unique_key, 214
 - \$_value, 214
 - \$_value_type, 214
 - __construct, 204
 - __toString, 205
 - adminListOrder, 205
 - canBeOrdered, 205
 - clean, 205
 - filterWhereClause, 206
 - formElement, 206
 - formElementWidget, 207
 - getAutoIncrement, 208
 - getDefault, 208
 - getLabel, 208
 - getLenght, 209
 - getName, 209
 - getPrimaryKey, 209
 - getRequired, 209
 - getUniqueKey, 209
 - getValue, 209
 - getValueType, 209
 - getWidget, 210
 - setAutoIncrement, 210
 - setDefault, 210
 - setLabel, 210
 - setLenght, 210
 - setName, 211
 - setPrimaryKey, 211
 - setRequired, 211
 - setUniqueKey, 211
 - setValue, 211
 - setValueType, 213
 - setWidget, 213
 - validate, 213
- Gino::FileField
 - \$_check_type, 224
 - \$_delete_file, 224
 - \$_directory, 224
 - \$_extensions, 224
 - \$_max_file_size, 224
 - \$_path_abs, 224
 - \$_path_add, 224
 - \$_prefix, 224
 - \$_types_allowed, 224
 - __construct, 217
 - checkFilename, 217
 - clean, 217
 - conformPath, 218
 - delete, 218
 - formElement, 218

- getAddPath, [218](#)
- getCheckType, [218](#)
- getDirectory, [219](#)
- getExtensions, [219](#)
- getMaxFileSize, [219](#)
- getPath, [219](#)
- getPrefix, [219](#)
- getTypesAllowed, [219](#)
- pathToFile, [219](#)
- saveFile, [221](#)
- setAddPath, [221](#)
- setCheckType, [221](#)
- setDirectory, [221](#)
- setExtensions, [222](#)
- setMaxFileSize, [222](#)
- setPath, [222](#)
- setPrefix, [222](#)
- setTypesAllowed, [222](#)
- validate, [224](#)
- Gino::FloatField
 - [\\$_decimal_digits](#), [228](#)
 - [\\$_int_digits](#), [228](#)
 - [__construct](#), [226](#)
 - [__toString](#), [226](#)
 - [clean](#), [226](#)
 - [formElement](#), [227](#)
 - [getDecimalDigits](#), [227](#)
 - [getIntDigits](#), [227](#)
 - [setDecimalDigits](#), [227](#)
 - [setIntDigits](#), [227](#)
- Gino::ForeignKeyField
 - [\\$_add_related](#), [231](#)
 - [\\$_add_related_url](#), [231](#)
 - [\\$_enum](#), [231](#)
 - [\\$_foreign](#), [231](#)
 - [\\$_foreign_order](#), [231](#)
 - [\\$_foreign_where](#), [231](#)
 - [__construct](#), [229](#)
 - [__toString](#), [230](#)
 - [formElement](#), [230](#)
 - [getEnum](#), [230](#)
- Gino::Form
 - [\\$_div_field_width](#), [256](#)
 - [\\$_div_label_width](#), [256](#)
 - [\\$_extension_denied](#), [256](#)
 - [\\$_fckeditor_field](#), [256](#)
 - [\\$_form_name](#), [256](#)
 - [\\$_ico_calendar_path](#), [256](#)
 - [\\$_input_field](#), [256](#)
 - [\\$_input_max](#), [256](#)
 - [\\$_input_size](#), [256](#)
 - [\\$_lng_trl](#), [256](#)
 - [\\$_max_file_size](#), [256](#)
 - [\\$_method](#), [256](#)
 - [\\$_multi_language](#), [256](#)
 - [\\$_options](#), [256](#)
 - [\\$_prefix_file](#), [256](#)
 - [\\$_prefix_thumb](#), [256](#)
 - [\\$_registry](#), [256](#)
 - [\\$_request](#), [256](#)
 - [\\$_requestVar](#), [256](#)
 - [\\$_tbl_attachment](#), [256](#)
 - [\\$_tbl_attachment_ctg](#), [256](#)
 - [\\$_textarea_field](#), [256](#)
 - [\\$_trd](#), [256](#)
 - [\\$_trns_id](#), [256](#)
 - [\\$_trns_table](#), [257](#)
 - [\\$_validation](#), [257](#)
 - [\\$session](#), [257](#)
 - [_IMAGE_GIF_](#), [257](#)
 - [_IMAGE_JPG_](#), [257](#)
 - [_IMAGE_PNG_](#), [257](#)
 - [__construct](#), [234](#)
 - [arequired](#), [235](#)
 - [captcha](#), [235](#)
 - [ccheckbox](#), [235](#)
 - [cfile](#), [236](#)
 - [checkCaptcha](#), [237](#)
 - [checkDefaultCaptcha](#), [237](#)
 - [checkFilename](#), [237](#)
 - [checkReCaptcha](#), [238](#)
 - [checkbox](#), [236](#)
 - [cinput](#), [238](#)
 - [cinput_date](#), [238](#)
 - [close](#), [239](#)
 - [countEqualName](#), [239](#)
 - [cradio](#), [239](#)
 - [createImage](#), [240](#)
 - [cselect](#), [240](#)
 - [ctextarea](#), [241](#)
 - [defaultCaptcha](#), [241](#)
 - [dimensionFile](#), [242](#)
 - [dirUpload](#), [242](#)
 - [editorHtml](#), [242](#)
 - [fcktextarea](#), [242](#)
 - [formFieldTranslation](#), [243](#)
 - [generateFormToken](#), [244](#)
 - [hidden](#), [244](#)
 - [imagePreviewer](#), [244](#)
 - [input](#), [244](#)
 - [isImage](#), [245](#)
 - [jsCountCharText](#), [245](#)
 - [label](#), [245](#)
 - [load](#), [245](#)
 - [manageFile](#), [246](#)
 - [multipleCheckbox](#), [247](#)
 - [noinput](#), [248](#)
 - [open](#), [249](#)
 - [option](#), [249](#)
 - [radio](#), [249](#)
 - [reCaptcha](#), [251](#)
 - [resizeImage](#), [251](#)
 - [retvar](#), [251](#)
 - [save](#), [253](#)
 - [saveImage](#), [253](#)
 - [select](#), [253](#)

- setMethod, 254
- setOptions, 254
- setValidation, 254
- textarea, 255
- upload, 255
- verifyFormToken, 255
- Gino::Frontend
 - \$_class, 261
 - \$_css_list, 261
 - \$_mdlLink, 261
 - \$_module, 261
 - \$_module_id, 261
 - \$_registry, 261
 - \$_view_list, 261
 - __construct, 258
 - actionModuleFile, 258
 - fileName, 259
 - formModuleFile, 259
 - manageFrontend, 259
 - moduleList, 259
 - pathToFile, 259
 - setLists, 261
- Gino::GImage
 - \$_abspath, 271
 - \$_dir, 271
 - \$_height, 271
 - \$_image, 271
 - \$_image_type, 271
 - \$_table, 271
 - \$_tmp_image, 271
 - \$_width, 271
 - __construct, 265
 - blackThresholdImage, 265
 - cloneImage, 266
 - crop, 266
 - cropCenter, 266
 - cropEntropy, 266
 - cropImage, 267
 - cropImageEntropy, 267
 - getEntropy, 267
 - getHeight, 267
 - getImageHistogram, 268
 - getPath, 268
 - getResource, 268
 - getThumbFromKey, 268
 - getWidth, 268
 - grayscaleEntropy, 269
 - makeThumb, 269
 - resize, 269
 - resizeImage, 269
 - save, 270
 - saveTmpImage, 270
 - slice, 270
 - stream, 270
 - thumb, 271
 - toKey, 271
- Gino::GTag
 - \$_table_tag, 284
 - \$_table_tag_taggeditem, 284
 - getAllTags, 282
 - getContentTags, 282
 - getRelatedContents, 282
 - getTagsHistogram, 284
 - saveContentTags, 284
- Gino::Http::Redirect
 - \$_url, 453
 - __construct, 453
 - redirectUrl, 453
 - send, 453
 - sendHeaders, 453
- Gino::Http::Request
 - \$COOKIES, 461
 - \$FILES, 461
 - \$GET, 461
 - \$META, 461
 - \$POST, 461
 - \$REQUEST, 461
 - \$absolute_url, 461
 - \$method, 461
 - \$path, 461
 - \$query_string, 461
 - \$request_uri, 461
 - \$root_absolute_url, 461
 - \$session, 461
 - \$url, 461
 - \$user, 461
 - __construct, 460
 - checkGETKey, 460
 - checkPOSTKey, 460
 - isSecure, 460
 - updateUrl, 460
 - valueOrNull, 461
- Gino::Http::Response
 - \$_content, 465
 - \$_content_type, 465
 - \$_encoding, 465
 - \$_headers, 465
 - \$_request, 465
 - \$_status_code, 465
 - \$_status_text, 466
 - __construct, 463
 - __invoke, 463
 - getContent, 463
 - send, 464
 - sendContent, 464
 - sendHeaders, 464
 - setContent, 464
 - setContentType, 464
 - setEncoding, 465
 - setHeaders, 465
 - setStatus, 465
- Gino::Http::ResponseFile
 - \$_content_disposition, 468
 - \$_filename, 468
 - __construct, 467
 - sendHeaders, 467

- setDispositionType, [468](#)
- Gino::Http::ResponseForbidden
 - __construct, [469](#)
 - sendContent, [470](#)
- Gino::Http::ResponseJson
 - __construct, [471](#)
- Gino::Http::ResponseNotFound
 - __construct, [472](#)
 - sendContent, [473](#)
- Gino::Http::ResponseServerError
 - __construct, [474](#)
 - sendContent, [474](#)
- Gino::ImageField
 - \$_height, [293](#)
 - \$_prefix_file, [293](#)
 - \$_prefix_thumb, [293](#)
 - \$_resize, [293](#)
 - \$_thumb, [293](#)
 - \$_thumb_height, [293](#)
 - \$_thumb_width, [293](#)
 - \$_width, [293](#)
 - _IMAGE_GIF_, [293](#)
 - _IMAGE_JPG_, [293](#)
 - _IMAGE_PNG_, [293](#)
 - __construct, [287](#)
 - delete, [288](#)
 - formElement, [288](#)
 - getHeighth, [288](#)
 - getPrefixFile, [288](#)
 - getPrefixThumb, [288](#)
 - getResize, [288](#)
 - getThumb, [289](#)
 - getThumbHeighth, [289](#)
 - getThumbWidth, [289](#)
 - getWidth, [289](#)
 - resizeImage, [289](#)
 - saveFile, [290](#)
 - savelImage, [290](#)
 - setHeight, [290](#)
 - setPrefixFile, [290](#)
 - setPrefixThumb, [291](#)
 - setResize, [291](#)
 - setThumb, [291](#)
 - setThumbHeight, [291](#)
 - setThumbWidth, [291](#)
 - setWidth, [293](#)
- Gino::InputForm
 - \$_form, [302](#)
 - __construct, [297](#)
 - checkbox, [297](#)
 - date, [297](#)
 - editor, [298](#)
 - email, [298](#)
 - file, [298](#)
 - hidden, [299](#)
 - multicheck, [299](#)
 - noinput, [299](#)
 - password, [300](#)
 - radio, [300](#)
 - select, [301](#)
 - text, [301](#)
 - textarea, [301](#)
- Gino::IntegerField
 - __construct, [306](#)
 - formElement, [306](#)
- Gino::JSMIn
 - \$a, [312](#)
 - \$b, [312](#)
 - \$input, [312](#)
 - \$inputIndex, [312](#)
 - \$inputLength, [312](#)
 - \$lookAhead, [312](#)
 - \$output, [312](#)
 - __construct, [310](#)
 - ACTION_DELETE_A, [312](#)
 - ACTION_DELETE_A_B, [312](#)
 - ACTION_KEEP_A, [312](#)
 - action, [310](#)
 - get, [311](#)
 - isAlphaNum, [311](#)
 - min, [311](#)
 - minify, [311](#)
 - next, [311](#)
 - ORD_LF, [312](#)
 - ORD_SPACE, [312](#)
 - peek, [312](#)
- Gino::Javascript
 - abiMapLib, [307](#)
 - analytics, [307](#)
 - onLoadFunction, [307](#)
 - sharethis, [308](#)
 - slimboxLib, [308](#)
 - vendor, [308](#)
- Gino::Loader
 - import, [332](#)
 - load, [332](#)
 - singleton, [332](#)
- Gino::Locale
 - \$_session, [340](#)
 - \$_strings, [340](#)
 - __construct, [337](#)
 - detectCodes, [338](#)
 - get, [338](#)
 - get_languages, [338](#)
 - init, [338](#)
 - initGettext, [339](#)
 - pathToFile, [339](#)
 - setLanguage, [339](#)
 - userLanguage, [339](#)
- Gino::Logger
 - exceptionReportAdmins, [343](#)
 - manageException, [343](#)
 - messageReportAdmins, [343](#)
 - stackTraceHtml, [343](#)
 - systemVariablesHtml, [343](#)
- Gino::ManyToManyField

- [\\$_add_related](#), 348
- [\\$_add_related_url](#), 348
- [\\$_enum](#), 348
- [\\$_join_table](#), 348
- [\\$_join_table_id](#), 348
- [\\$_join_table_m2m_id](#), 348
- [\\$_m2m](#), 348
- [\\$_m2m_order](#), 348
- [\\$_m2m_where](#), 348
- [__construct](#), 345
- [__toString](#), 346
- [clean](#), 346
- [filterWhereClause](#), 347
- [formElement](#), 347
- [getJoinTable](#), 347
- [getJoinTableId](#), 347
- [getJoinTableM2mId](#), 348
- Gino::ManyToManyInlineField
 - [\\$_enum](#), 352
 - [\\$_m2m](#), 352
 - [\\$_m2m_controller](#), 352
 - [\\$_m2m_order](#), 352
 - [\\$_m2m_where](#), 352
 - [__construct](#), 350
 - [__toString](#), 351
 - [clean](#), 351
 - [filterWhereClause](#), 351
 - [formElement](#), 351
 - [getEnum](#), 352
- Gino::ManyToManyThroughField
 - [\\$_controller](#), 356
 - [\\$_enum](#), 356
 - [\\$_m2m](#), 356
 - [\\$_m2m_controller](#), 356
 - [\\$_model_table_id](#), 356
 - [__construct](#), 354
 - [__toString](#), 354
 - [clean](#), 354
 - [filterWhereClause](#), 355
 - [formElement](#), 355
 - [getController](#), 355
 - [getM2m](#), 355
 - [getModelTableId](#), 356
 - [getTable](#), 356
- Gino::MobileDetect
 - [\\$accept](#), 370
 - [\\$devices](#), 370
 - [\\$isAndroid](#), 370
 - [\\$isBlackberry](#), 370
 - [\\$isGeneric](#), 370
 - [\\$isIphone](#), 370
 - [\\$isMobile](#), 370
 - [\\$isOpera](#), 370
 - [\\$isPalm](#), 370
 - [\\$isWindows](#), 370
 - [\\$userAgent](#), 370
 - [__call](#), 369
 - [__construct](#), 369
 - [isDevice](#), 369
 - [isMobile](#), 370
- Gino::Model
 - [\\$_check_is_constraint](#), 382
 - [\\$_chgP](#), 382
 - [\\$_controller](#), 382
 - [\\$_db](#), 382
 - [\\$_fields_label](#), 382
 - [\\$_is_constraint](#), 382
 - [\\$_lng_dft](#), 382
 - [\\$_lng_nav](#), 382
 - [\\$_locale](#), 382
 - [\\$_m2m](#), 382
 - [\\$_m2mt](#), 382
 - [\\$_model_label](#), 382
 - [\\$_p](#), 382
 - [\\$_registry](#), 382
 - [\\$_request](#), 382
 - [\\$_structure](#), 382
 - [\\$_tbl_data](#), 382
 - [\\$_trd](#), 382
 - [__construct](#), 374
 - [__get](#), 375
 - [__set](#), 375
 - [__toString](#), 375
 - [addm2m](#), 375
 - [addm2mthrough](#), 376
 - [checkIsConstraint](#), 376
 - [dataType](#), 376
 - [delete](#), 376
 - [deleteDbData](#), 376
 - [deleteInstance](#), 377
 - [deletem2m](#), 377
 - [deletem2mthrough](#), 377
 - [deletem2mthroughField](#), 377
 - [fieldLabel](#), 377
 - [getController](#), 378
 - [getFromSlug](#), 378
 - [getModelLabel](#), 378
 - [getSelectOptionsFromObjects](#), 378
 - [getStructure](#), 378
 - [getTable](#), 378
 - [isConstraintError](#), 379
 - [m2mtObject](#), 379
 - [ml](#), 379
 - [objects](#), 379
 - [save](#), 380
 - [savem2m](#), 380
 - [setCheckIsConstraint](#), 380
 - [setIsConstraint](#), 380
 - [structure](#), 381
 - [updateStructure](#), 381
- Gino::Options
 - [\\$_action](#), 407
 - [\\$_class](#), 407
 - [\\$_class_prefix](#), 407
 - [\\$_db](#), 407
 - [\\$_home](#), 407

- [\\$_instance](#), 407
- [\\$_module_app](#), 407
- [\\$_tbl_options](#), 407
- [\\$_title](#), 407
- [__construct](#), 405
- [actionOptions](#), 406
- [editableField](#), 406
- [manageDoc](#), 406
- [setData](#), 406
- Gino::OutputCache
 - [__construct](#), 409
 - [start](#), 409
 - [stop](#), 409
- Gino::Paginator
 - [\\$_current_page](#), 438
 - [\\$_interval](#), 438
 - [\\$_items_for_page](#), 438
 - [\\$_items_number](#), 438
 - [\\$_pages_number](#), 438
 - [__construct](#), 436
 - [getCurrentPage](#), 436
 - [limit](#), 436
 - [limitQuery](#), 436
 - [navigator](#), 436
 - [pages](#), 437
 - [pagination](#), 437
 - [setCurrentPage](#), 437
 - [summary](#), 437
 - [urlPage](#), 437
- Gino::Plugin::mysql
 - [\\$_affected](#), 403
 - [\\$_cache](#), 403
 - [\\$_cnt](#), 403
 - [\\$_connection](#), 403
 - [\\$_db_charset](#), 403
 - [\\$_db_host](#), 404
 - [\\$_db_name](#), 404
 - [\\$_db_password](#), 404
 - [\\$_db_user](#), 404
 - [\\$_dbconn](#), 404
 - [\\$_dbresults](#), 404
 - [\\$_debug](#), 404
 - [\\$_enable_cache](#), 404
 - [\\$_lastid](#), 404
 - [\\$_numberrows](#), 404
 - [\\$_qry](#), 404
 - [\\$_rows](#), 404
 - [\\$_sql](#), 404
 - [__construct](#), 397
 - [actionquery](#), 397
 - [affected](#), 397
 - [autoIncValue](#), 397
 - [begin](#), 397
 - [closeConnection](#), 398
 - [columnHasValue](#), 398
 - [commit](#), 398
 - [concat](#), 398
 - [conformType](#), 398
 - [delete](#), 399
 - [distinct](#), 399
 - [drop](#), 399
 - [dump](#), 399
 - [dumpDatabase](#), 399
 - [escapeString](#), 399
 - [execQuery](#), 400
 - [fieldInformations](#), 400
 - [freeresult](#), 400
 - [getFieldFromId](#), 400
 - [getFieldsName](#), 400
 - [getInfoQuery](#), 400
 - [getNumRecords](#), 401
 - [getTableStructure](#), 401
 - [getlastid](#), 401
 - [insert](#), 401
 - [join](#), 401
 - [limit](#), 401
 - [multiActionquery](#), 401
 - [openConnection](#), 402
 - [query](#), 402
 - [restore](#), 402
 - [resultselect](#), 402
 - [rollback](#), 402
 - [select](#), 402
 - [selectquery](#), 402
 - [setUtf8](#), 403
 - [setconnection](#), 403
 - [setnumberrows](#), 403
 - [setsql](#), 403
 - [tableexists](#), 403
 - [union](#), 403
 - [update](#), 403
- Gino::Plugin::sqlsrv
 - [\\$_affected](#), 512
 - [\\$_cache](#), 512
 - [\\$_cnt](#), 513
 - [\\$_connection](#), 513
 - [\\$_db_charset](#), 513
 - [\\$_db_host](#), 513
 - [\\$_db_name](#), 513
 - [\\$_db_password](#), 513
 - [\\$_db_user](#), 513
 - [\\$_dbconn](#), 513
 - [\\$_dbresults](#), 513
 - [\\$_debug](#), 513
 - [\\$_enable_cache](#), 513
 - [\\$_lastid](#), 513
 - [\\$_numberrows](#), 513
 - [\\$_qry](#), 513
 - [\\$_rows](#), 513
 - [\\$_sql](#), 513
 - [__construct](#), 503
 - [actionquery](#), 504
 - [affected](#), 504
 - [autoIncValue](#), 504
 - [begin](#), 504
 - [closeConnection](#), 504

- columnHasValue, 504
- commit, 504
- concat, 505
- conformType, 505
- delete, 506
- distinct, 506
- drop, 506
- dump, 506
- dumpDatabase, 506
- escapeString, 506
- execQuery, 506
- fieldInformations, 507
- freeresult, 507
- getCheckConstraint, 507
- getConstraintType, 508
- getDataType, 508
- getFieldFromId, 508
- getFieldLength, 508
- getFieldsName, 509
- getInfoQuery, 509
- getInformationKey, 509
- getNumRecords, 509
- getTableStructure, 510
- getlastid, 509
- insert, 510
- join, 510
- limit, 510
- limitQuery, 510
- listTables, 510
- multiActionquery, 510
- openConnection, 511
- query, 511
- restore, 511
- resultselect, 511
- rollback, 511
- select, 511
- selectquery, 511
- setconnection, 512
- setnumberrows, 512
- setsql, 512
- tableexists, 512
- union, 512
- update, 512
- Gino::Registry
 - \$vars, 458
 - __get, 455
 - __set, 455
 - addCss, 456
 - addHeadLink, 456
 - addJs, 456
 - addMeta, 457
 - propertyExists, 457
 - variables, 457
- Gino::Router
 - \$_controller_view, 478
 - \$_registry, 478
 - \$_request, 478
 - \$_url_class, 478
 - \$_url_instance, 478
 - \$_url_method, 478
 - __construct, 476
 - EVT_NAME, 479
 - isRoute, 476
 - link, 477
 - rewritePathInfo, 477
 - rewriteQueryString, 477
 - route, 477
 - setUrlParams, 478
 - transformPathQueryString, 478
 - urlRewrite, 478
- Gino::Search
 - \$_table, 481
 - __construct, 480
 - clearSearchString, 480
 - getKeywords, 480
 - getSearchResults, 480
 - makeQuery, 481
- Gino::Session
 - \$_vars, 488
 - \$sessionState, 488
 - __construct, 487
 - __get, 487
 - __isset, 487
 - __set, 487
 - __unset, 488
 - destroy, 488
 - SESSION_NOT_STARTED, 488
 - SESSION_STARTED, 489
 - startSession, 488
- Gino::Singleton
 - \$_instances, 491
 - __clone, 490
 - __construct, 490
 - __sleep, 490
 - __wakeup, 490
 - instance, 490
 - instance_to_class, 490
- Gino::Skin
 - \$_interface, 496
 - \$_tbl_data, 496
 - \$table, 496
 - __construct, 493
 - actionDelSkin, 493
 - actionSkin, 494
 - formDelSkin, 494
 - formSkin, 494
 - getSkin, 494
 - layoutInfo, 495
 - newSkinPriority, 495
 - removeCss, 495
 - removeTemplate, 495
 - sortUp, 495
- Gino::SlugField
 - \$_autofill, 499
 - \$_js, 499
 - __construct, 497

- formElement, [498](#)
- Gino::TagField
 - \$_model_controller_class, [530](#)
 - \$_model_controller_instance, [530](#)
 - __construct, [529](#)
 - clean, [529](#)
 - formElement, [529](#)
 - save, [529](#)
 - tagCloud, [530](#)
- Gino::Template
 - \$_align_dict, [539](#)
 - \$_blocks_number, [539](#)
 - \$_blocks_properties, [539](#)
 - \$_home, [539](#)
 - \$_interface, [539](#)
 - \$_tbl_data, [539](#)
 - \$_um_dict, [539](#)
 - \$table, [539](#)
 - \$table_block, [539](#)
 - __construct, [533](#)
 - actionCopyTemplate, [533](#)
 - actionDelTemplate, [533](#)
 - actionFreeTemplate, [534](#)
 - actionTemplate, [534](#)
 - addBlockForm, [534](#)
 - cellCtrl, [535](#)
 - createTemplate, [535](#)
 - deleteBlocks, [535](#)
 - formBlock, [535](#)
 - formCopyTemplate, [536](#)
 - formData, [536](#)
 - formDelTemplate, [536](#)
 - formFreeTemplate, [536](#)
 - formOutline, [536](#)
 - formTemplate, [536](#)
 - initBlocksProperties, [537](#)
 - layoutInfo, [537](#)
 - manageTemplate, [537](#)
 - printBlock, [537](#)
 - renderNave, [538](#)
 - saveBlock, [538](#)
 - setFilename, [538](#)
 - tplBlockForm, [539](#)
- Gino::Test
 - setPage, [65](#)
- Gino::Test::FuncTest
 - test_absolutePath, [263](#)
 - test_arrayToObject, [263](#)
 - test_baseFilename, [263](#)
 - test_checkEmail, [263](#)
 - test_cutHtmlText, [263](#)
 - test_cutString, [263](#)
 - test_dateDiff, [263](#)
 - test_dateToDbDate, [263](#)
 - test_dbDateToDate, [263](#)
 - test_dbDatetimeToDate, [263](#)
 - test_dbDatetimeToTime, [263](#)
 - test_dbNumberToNumber, [263](#)
 - test_extension, [263](#)
 - test_gOpt, [263](#)
 - test_getDateDiff, [263](#)
 - test_isValid, [263](#)
 - test_numberToDB, [263](#)
 - test_relativePath, [263](#)
 - test_searchNameFile, [263](#)
 - test_timeDiff, [263](#)
 - test_timeToDbTime, [263](#)
 - test_traslitterazione, [263](#)
- Gino::Test::PaginatorTest
 - test_getCurrentPage, [439](#)
 - test_limit, [439](#)
 - test_limitQuery, [439](#)
 - test_pages, [439](#)
 - test_summary, [439](#)
- Gino::TextField
 - \$_trnsl, [543](#)
 - __construct, [541](#)
 - canBeOrdered, [541](#)
 - clean, [541](#)
 - filterWhereClause, [542](#)
 - formElement, [542](#)
 - getTrnsl, [542](#)
 - setTrnsl, [542](#)
- Gino::TimeField
 - __construct, [544](#)
 - __toString, [544](#)
 - clean, [545](#)
 - formElement, [545](#)
- Gino::Translation
 - \$_db, [548](#)
 - \$_lng, [548](#)
 - \$_lngDft, [548](#)
 - \$_registry, [548](#)
 - \$_tbl_translation, [548](#)
 - __construct, [546](#)
 - actionTranslation, [546](#)
 - deleteTranslations, [547](#)
 - formTranslation, [547](#)
 - listItemOrdered, [547](#)
 - manageTranslation, [547](#)
 - selectTXT, [548](#)
- Gino::View
 - \$_data, [559](#)
 - \$_dft_view_folder, [559](#)
 - \$_registry, [560](#)
 - \$_view_folder, [560](#)
 - \$_view_tpl, [560](#)
 - __construct, [558](#)
 - assign, [559](#)
 - render, [559](#)
 - setViewTpl, [559](#)
- Gino::YearField
 - __construct, [561](#)
 - formElement, [562](#)
- Gino::sqlcode
 - \$_dbms, [500](#)

- __construct, 500
 - admin_actionUser, 500
- Gino\App, 56
- Gino\App\Attachment, 57
- Gino\App\Auth, 58
- Gino\App\Graphics, 59
- Gino\App\Index, 59
- Gino\App\Instruments, 59
- Gino\App\Language, 60
- Gino\App\Layout, 60
- Gino\App\Menu, 60
- Gino\App\Module, 60
- Gino\App\Page, 61
- Gino\App\PhpModuleView, 61
- Gino\App\SearchSite, 62
- Gino\App\Statistics, 62
- Gino\App\SysClass, 62
- Gino\App\Sysconf, 63
- Gino\App\Sysfunc, 63
- Gino\Exception, 63
- Gino\Http, 64
- Gino\Plugin, 64
- Gino\Test, 64
- global.php
 - else, 636
 - get_app_name_class_ns, 635
 - get_app_namespace, 635
 - get_model_app_name_class_ns, 636
 - get_name_class, 636
 - set_name_class, 636
- google_analytics
 - Gino::Document, 176
- graphics, 272
- GraphicsItem, 276
- grayscaleEntropy
 - Gino::GImage, 269
- Group, 278
- HOME_FILE
 - Gino, 54
- hasAdminPerm
 - Gino::App::Auth::User, 555
- hasPerm
 - Gino::App::Auth::User, 556
- headLine
 - Gino::Document, 176
- hex2RGB
 - Gino::Captcha, 119
- hidden
 - Gino::AdminTable, 80
 - Gino::Form, 244
 - Gino::InputForm, 299
- htmlChars
 - Gino, 41
- htmlCharsText
 - Gino, 42
- htmlInput
 - Gino, 42
- htmlInputEditor
 - Gino, 42
- htmlToEntities
 - Gino, 42
- httpResponse
 - Gino::Exception::Exception403, 191
 - Gino::Exception::Exception404, 193
 - Gino::Exception::Exception500, 194
- icon
 - Gino, 43
- ImageField, 284
- imagePreviewer
 - Gino::Form, 244
- imgPath
 - Gino::App::Page::PageEntry, 434
- import
 - Gino::Loader, 332
- include.php
 - SITE_ROOT, 644
 - SITE_WWW, 644
- index, 293
- index.php, 591
- info
 - Gino::App::Layout::layout, 326
 - Gino::App::PhpModuleView::phpModuleView, 449
 - Gino::App::SearchSite::searchSite, 484
 - Gino::App::Statistics::statistics, 515
 - Gino::App::SysClass::sysClass, 521
- init
 - Gino::Locale, 338
- initBlocksProperties
 - Gino::Template, 537
- initGettext
 - Gino::Locale, 339
- initMobile
 - Gino::Core, 138
- initOrderList
 - Gino::App::Menu::MenuVoice, 367
- initRegistry
 - Gino::Core, 138
- input
 - Gino::Form, 244
- InputForm, 296
- insert
 - Gino::DbManager, 164
 - Gino::Plugin::mysql, 401
 - Gino::Plugin::sqlsrv, 510
- instance
 - Gino::Db, 154
 - Gino::Singleton, 490
- instance_to_class
 - Gino::Singleton, 490
- instruments, 302
- IntegerField, 304
- is_browser
 - Gino, 45
- is_browser_platform
 - Gino, 45
- is_browser_version

- Gino, [45](#)
- is_robot
 - Gino, [45](#)
- isAlphaNum
 - Gino::JSMIn, [311](#)
- isCached
 - Gino::Cache, [116](#)
- isConstraintError
 - Gino::Model, [379](#)
- isDevice
 - Gino::MobileDetect, [369](#)
- isHeader
 - Gino::App::Graphics::graphics, [274](#)
- isImage
 - Gino::Form, [245](#)
- isMobile
 - Gino::MobileDetect, [370](#)
- isNumeric
 - Gino, [45](#)
- isRoute
 - Gino::Router, [476](#)
- isSecure
 - Gino::Http::Request, [460](#)
- isValid
 - Gino, [46](#)
- JSMIn, [308](#)
- JSMInException, [312](#)
- Javascript, [306](#)
- join
 - Gino::DbManager, [165](#)
 - Gino::Plugin::mysql, [401](#)
 - Gino::Plugin::sqlsrv, [510](#)
- joinUserPermission
 - Gino::App::Auth::auth, [108](#)
- joingrouppermission
 - Gino::App::Auth::auth, [107](#)
- jsCountCharText
 - Gino::Form, [245](#)
- jsSearchModulesLib
 - Gino::App::Menu::menu, [360](#)
- jsSortLib
 - Gino::App::Menu::menu, [361](#)
- jsVar
 - Gino, [46](#)
- jsonImageList
 - Gino::App::Attachment::attachment, [91](#)
- LDAP_APP_PASSWORD
 - Gino::App::Auth, [58](#)
- LDAP_APP_USERNAME
 - Gino::App::Auth, [58](#)
- LDAP_AUTH
 - configuration.php, [590](#)
- LDAP_BASE_DN
 - Gino::App::Auth, [58](#)
- LDAP_DOMAIN
 - Gino::App::Auth, [58](#)
- LDAP_FILTER_SEARCH
 - Gino::App::Auth, [58](#)
- LDAP_HOST
 - Gino::App::Auth, [58](#)
- LDAP_PORT
 - Gino::App::Auth, [59](#)
- LDAP_PROTOCOL_VERSION
 - Gino::App::Auth, [59](#)
- LDAP_SEARCH_DN
 - Gino::App::Auth, [59](#)
- LIB_DIR
 - Gino, [54](#)
- label
 - Gino::Form, [245](#)
- Lang, [313](#)
- language, [317](#)
- languageCodes
 - Gino::App::Language::Lang, [316](#)
- layout, [320](#)
- layoutInfo
 - Gino::Css, [142](#)
 - Gino::Skin, [495](#)
 - Gino::Template, [537](#)
- Ldap, [328](#)
- lib/classes/class.Access.php, [591](#)
- lib/classes/class.AdminTable.php, [592](#)
- lib/classes/class.Cache.php, [592](#)
- lib/classes/class.Captcha.php, [593](#)
- lib/classes/class.Compressor.php, [593](#)
- lib/classes/class.Core.php, [594](#)
- lib/classes/class.Css.php, [595](#)
- lib/classes/class.Db.php, [595](#)
- lib/classes/class.Document.php, [596](#)
- lib/classes/class.Error.php, [596](#)
- lib/classes/class.EventDispatcher.php, [597](#)
- lib/classes/class.Export.php, [597](#)
- lib/classes/class.Form.php, [598](#)
- lib/classes/class.Frontend.php, [598](#)
- lib/classes/class.GImage.php, [599](#)
- lib/classes/class.GTag.php, [600](#)
- lib/classes/class.InputForm.php, [600](#)
- lib/classes/class.Javascript.php, [601](#)
- lib/classes/class.Loader.php, [601](#)
- lib/classes/class.Locale.php, [602](#)
- lib/classes/class.Logger.php, [603](#)
- lib/classes/class.MobileDetect.php, [603](#)
- lib/classes/class.Options.php, [603](#)
- lib/classes/class.Paginator.php, [604](#)
- lib/classes/class.Registry.php, [605](#)
- lib/classes/class.Router.php, [605](#)
- lib/classes/class.Search.php, [606](#)
- lib/classes/class.Session.php, [606](#)
- lib/classes/class.Singleton.php, [607](#)
- lib/classes/class.Skin.php, [607](#)
- lib/classes/class.Template.php, [608](#)
- lib/classes/class.Translation.php, [608](#)
- lib/classes/exceptions/class.Exception403.php, [609](#)
- lib/classes/exceptions/class.Exception404.php, [610](#)
- lib/classes/exceptions/class.Exception500.php, [610](#)

- lib/classes/fields/class.BooleanField.php, 611
- lib/classes/fields/class.CharField.php, 611
- lib/classes/fields/class.DateField.php, 612
- lib/classes/fields/class.DatetimeField.php, 612
- lib/classes/fields/class.DirectoryField.php, 613
- lib/classes/fields/class.EmailField.php, 613
- lib/classes/fields/class.EnumField.php, 614
- lib/classes/fields/class.Field.php, 614
- lib/classes/fields/class.FileField.php, 615
- lib/classes/fields/class.FloatField.php, 616
- lib/classes/fields/class.ForeignKeyField.php, 616
- lib/classes/fields/class.ImageField.php, 617
- lib/classes/fields/class.IntegerField.php, 617
- lib/classes/fields/class.ManyToManyField.php, 618
- lib/classes/fields/class.ManyToManyInlineField.php, 618
- lib/classes/fields/class.ManyToManyThroughField.php, 619
- lib/classes/fields/class.SlugField.php, 619
- lib/classes/fields/class.TagField.php, 620
- lib/classes/fields/class.TextField.php, 621
- lib/classes/fields/class.TimeField.php, 621
- lib/classes/fields/class.YearField.php, 622
- lib/classes/http/class.Redirect.php, 622
- lib/classes/http/class.Request.php, 623
- lib/classes/http/class.Response.php, 623
- lib/classes/http/class.ResponseFile.php, 624
- lib/classes/http/class.ResponseForbidden.php, 624
- lib/classes/http/class.ResponseJson.php, 625
- lib/classes/http/class.ResponseNotFound.php, 626
- lib/classes/http/class.ResponseServerError.php, 626
- lib/classes/mvc/class.Controller.php, 627
- lib/classes/mvc/class.Model.php, 627
- lib/classes/mvc/class.View.php, 628
- lib/const.php, 628
- lib/func.browser.php, 629
- lib/func.file.php, 630
- lib/func.php, 631
- lib/func.var.php, 633
- lib/global.php, 634
- lib/plugin/plugin.mysql.php, 637
- lib/plugin/plugin.sqlsrv.php, 637
- lib/recaptchalib.php, 638
- lib/sqlcode.php, 640
- limit
 - Gino::DbManager, 165
 - Gino::Paginator, 436
 - Gino::Plugin::mysql, 401
 - Gino::Plugin::sqlsrv, 510
- limitQuery
 - Gino::Paginator, 436
 - Gino::Plugin::sqlsrv, 510
- link
 - Gino::Controller, 134
 - Gino::Router, 477
- linkAdmin
 - Gino::Controller, 134
- links
 - Gino::App::Instruments::instruments, 304
- listItemOrdered
 - Gino::Translation, 547
- listMenu
 - Gino::App::Menu::menu, 361
- listModule
 - Gino::App::Module::module, 388
- listProv
 - Gino, 46
- listTables
 - Gino::Plugin::sqlsrv, 510
- listen
 - Gino::EventDispatcher, 188
- listenEmitter
 - Gino::EventDispatcher, 188
- load
 - Gino::Form, 245
 - Gino::Loader, 332
- Loader, 331
- Locale, 334
- LogAccess, 340
- logAccess
 - Gino::Access, 69
- Logger, 342
- login
 - Gino::App::Auth::auth, 108
- loginError
 - Gino::Access, 69
- loginSuccess
 - Gino::Access, 69
- m2mtObject
 - Gino::Model, 379
- m2mthroughAction
 - Gino::AdminTable, 80
- MAX_FILE_SIZE
 - configuration.php, 590
- MODEL_CLASS_PREFIX
 - Gino, 54
- makeQuery
 - Gino::Search, 481
- makeThumb
 - Gino::GImage, 269
- manageAttachment
 - Gino::App::Attachment::attachment, 91
- manageAuth
 - Gino::App::Auth::auth, 108
- manageCategory
 - Gino::App::Attachment::attachment, 92
- manageComment
 - Gino::App::Page::page, 418
- manageCss
 - Gino::App::Layout::layout, 326
- manageCtg
 - Gino::App::Page::page, 418
- manageDoc
 - Gino::App::Menu::menu, 361
 - Gino::App::PhpModuleView::phpModuleView, 450
 - Gino::Options, 406
- manageEntry

- Gino::App::Page::page, 418
- manageException
 - Gino::Logger, 343
- manageFile
 - Gino::Form, 246
- manageFrontend
 - Gino::Controller, 134
 - Gino::Frontend, 259
- manageGraphics
 - Gino::App::Graphics::graphics, 274
- manageGroup
 - Gino::App::Auth::auth, 108
- manageInstruments
 - Gino::App::Instruments::instruments, 304
- manageItem
 - Gino::App::Attachment::attachment, 92
- manageLang
 - Gino::App::Language::language, 318
- manageLanguage
 - Gino::App::Language::language, 320
- manageLayout
 - Gino::App::Layout::layout, 326
- manageModule
 - Gino::App::Module::module, 388
- manageOptions
 - Gino::Controller, 134
- managePage
 - Gino::App::Page::page, 418
- managePermission
 - Gino::App::Auth::auth, 108
- manageSearchSite
 - Gino::App::SearchSite::searchSite, 484
- manageSkin
 - Gino::App::Layout::layout, 326
- manageStatistics
 - Gino::App::Statistics::statistics, 515
- manageSysClass
 - Gino::App::SysClass::sysClass, 521
- manageSysconf
 - Gino::App::Sysconf::sysconf, 524
- manageTemplate
 - Gino::App::Layout::layout, 326
 - Gino::Template, 537
- manageTranslation
 - Gino::Translation, 547
- manageUser
 - Gino::App::Auth::auth, 109
- manageView
 - Gino::App::Layout::layout, 327
- ManyToManyField, 344
- ManyToManyInlineField, 348
- ManyToManyThroughField, 352
- menu, 356
- MenuVoice, 364
- mergeCss
 - Gino::Compressor, 126
- mergeJs
 - Gino::Compressor, 126
- messageReportAdmins
 - Gino::Logger, 343
- mimeTypes
 - Gino::App::Instruments::instruments, 304
- min
 - Gino::JSMIn, 311
- minify
 - Gino::JSMIn, 311
- minifyCss
 - Gino::Compressor, 126
- minifyJs
 - Gino::Compressor, 127
- ml
 - Gino::Model, 379
- MobileDetect, 368
- modClass
 - Gino::Document, 177
- modPage
 - Gino::Document, 177
- modUrl
 - Gino::Document, 177
- Model, 371
- modelAction
 - Gino::AdminTable, 80
 - Gino::App::Attachment::AttachmentItemAdmin↔
Table, 101
 - Gino::App::Auth::AdminTable_AuthUser, 87
- modelForm
 - Gino::AdminTable, 82
- module, 382
- ModuleApp, 389
- moduleApp
 - Gino::App::Module::ModuleInstance, 393
- ModuleInstance, 391
- moduleList
 - Gino::Frontend, 259
- modulesCodeList
 - Gino::App::Layout::layout, 327
- modulesList
 - Gino::App::Layout::layout, 327
- modulesManageArray
 - Gino::App::Index::index, 295
- moveCss
 - Gino::Compressor, 127
- multiActionquery
 - Gino::DbManager, 165
 - Gino::Plugin::mysql, 401
 - Gino::Plugin::sqlsrv, 510
- multicheck
 - Gino::InputForm, 299
- multipleCheckbox
 - Gino::Form, 247
- mysql, 394
- navigator
 - Gino::Paginator, 436
- newSkinPriority
 - Gino::Skin, 495
- next

- Gino::JSMin, 311
- noinput
 - Gino::Form, 248
 - Gino::InputForm, 299
- notifyComment
 - Gino::App::Page::PageComment, 429
- numberToDB
 - Gino, 46
- ORD_LF
 - Gino::JSMin, 312
- ORD_SPACE
 - Gino::JSMin, 312
- OS
 - Gino, 54
- obj_serialize
 - Gino, 47
- obj_unserialize
 - Gino, 47
- objects
 - Gino::Model, 379
- onLoadFunction
 - Gino::Javascript, 307
- open
 - Gino::Form, 249
- openConnection
 - Gino::DbManager, 166
 - Gino::Plugin::mysql, 402
 - Gino::Plugin::sqlsrv, 511
- option
 - Gino::Form, 249
- Options, 404
- OutputCache, 407
- outputFunctions
 - Gino::App::Auth::auth, 109
 - Gino::App::Graphics::graphics, 274
 - Gino::App::Index::index, 295
 - Gino::App::Language::language, 320
 - Gino::App::Menu::menu, 361
 - Gino::App::Page::page, 418
 - Gino::App::PhpModuleView::phpModuleView, 450
 - Gino::App::SearchSite::searchSite, 484
- PACKAGE_EXTENSION
 - Gino::App::SysClass::sysClass, 522
- PLUGIN_DIR
 - Gino, 54
- page, 409
- page403
 - Gino::App::Sysfunc::sysfunc, 526
- page404
 - Gino::App::Sysfunc::sysfunc, 526
- page500
 - Gino::App::Sysfunc::sysfunc, 527
- PageCategory, 423
- PageComment, 425
- PageEntry, 430
- pages
 - Gino::Paginator, 437
- pagination
 - Gino::Paginator, 437
- Paginator, 435
- PaginatorTest, 438
- parseModules
 - Gino::Document, 177
- parseTemplate
 - Gino::App::Page::page, 419
- password
 - Gino::InputForm, 300
- passwordRules
 - Gino::App::Auth::auth, 109
- path
 - Gino::App::Attachment::AttachmentCtg, 95
 - Gino::App::Attachment::AttachmentItem, 98
- pathDownload
 - Gino::App::Attachment::AttachmentItem, 98
- pathToFile
 - Gino::FileField, 219
 - Gino::Frontend, 259
 - Gino::Locale, 339
- pathToSelectedVoice
 - Gino::App::Menu::menu, 362
- pathView
 - Gino::App::Attachment::AttachmentItem, 98
- peek
 - Gino::JSMin, 312
- Permission, 440
- permission
 - Gino::AdminTable, 84
- permissions
 - Gino::App::Page::page, 419
 - Gino::App::PhpModuleView::phpModuleView, 450
- PhpModule, 444
- phpModuleView, 447
- platforms
 - Gino, 47
- preCodeParser
 - Gino, 47
- previewAdminList
 - Gino::App::Attachment::AttachmentItem, 99
- previewLink
 - Gino::App::Attachment::AttachmentItem, 99
- printBlock
 - Gino::Template, 537
- printError
 - Gino::App::Auth::Ldap, 330
- printFooterAdhoc
 - Gino::App::Graphics::graphics, 274
- printFooterAdmin
 - Gino::App::Graphics::graphics, 275
- printFooterMobile
 - Gino::App::Graphics::graphics, 275
- printFooterPrivate
 - Gino::App::Graphics::graphics, 275
- printFooterPublic
 - Gino::App::Graphics::graphics, 275
- printHeaderAdhoc

- Gino::App::Graphics::graphics, 275
- printHeaderAdmin
 - Gino::App::Graphics::graphics, 275
- printHeaderMobile
 - Gino::App::Graphics::graphics, 275
- printHeaderPrivate
 - Gino::App::Graphics::graphics, 276
- printHeaderPublic
 - Gino::App::Graphics::graphics, 276
- printItemsClass
 - Gino::App::Menu::menu, 362
- printItemsList
 - Gino::App::Menu::menu, 362
- printItemsPage
 - Gino::App::Menu::menu, 363
- propertyExists
 - Gino::Registry, 457
- query
 - Gino::DbManager, 166
 - Gino::Plugin::mysql, 402
 - Gino::Plugin::sqlsrv, 511
- README.md, 641
- RECAPTCHA_API_SECURE_SERVER
 - recaptchalib.php, 640
- RECAPTCHA_API_SERVER
 - recaptchalib.php, 640
- RECAPTCHA_VERIFY_SERVER
 - recaptchalib.php, 640
- radio
 - Gino::Form, 249
 - Gino::InputForm, 300
- reCaptcha
 - Gino::Form, 251
- ReCaptchaResponse, 451
 - \$error, 451
 - \$is_valid, 451
- read
 - Gino::Cache, 116
- readFile
 - Gino::AdminTable, 84
- recaptcha_check_answer
 - recaptchalib.php, 639
- recaptcha_get_html
 - recaptchalib.php, 639
- recaptcha_get_signup_url
 - recaptchalib.php, 640
- recaptcha_mailhide_html
 - recaptchalib.php, 640
- recaptcha_mailhide_url
 - recaptchalib.php, 640
- recaptchalib.php
 - _recaptcha_aes_encrypt, 638
 - _recaptcha_aes_pad, 638
 - _recaptcha_http_post, 638
 - _recaptcha_mailhide_email_parts, 639
 - _recaptcha_mailhide_urlbase64, 639
 - _recaptcha_qlencode, 639
 - RECAPTCHA_API_SECURE_SERVER, 640
 - RECAPTCHA_API_SERVER, 640
 - RECAPTCHA_VERIFY_SERVER, 640
 - recaptcha_check_answer, 639
 - recaptcha_get_html, 639
 - recaptcha_get_signup_url, 640
 - recaptcha_mailhide_html, 640
 - recaptcha_mailhide_url, 640
- Redirect, 451
- redirectUrl
 - Gino::Http::Redirect, 453
- Registry, 454
- relatedContentsList
 - Gino::App::Page::page, 419
- relativePath
 - Gino, 48
- remove_attributes
 - Gino, 48
- removeBOM
 - Gino::Export, 197
- removeCss
 - Gino::Skin, 495
- removeTemplate
 - Gino::Skin, 495
- render
 - Gino::App::Graphics::graphics, 276
 - Gino::App::Menu::menu, 363
 - Gino::Captcha, 119
 - Gino::Document, 178
 - Gino::View, 559
- renderMenuAdmin
 - Gino::App::Menu::menu, 363
- renderModule
 - Gino::Document, 178
- renderNave
 - Gino::Document, 178
 - Gino::Template, 538
- replaceChar
 - Gino, 48
- replaceTplVar
 - Gino::App::Page::page, 419
- Request, 458
- requireAdmin
 - Gino::Access, 70
- requirePerm
 - Gino::Access, 70
 - Gino::Controller, 134
- resize
 - Gino::GImage, 269
- resizeImage
 - Gino::Form, 251
 - Gino::GImage, 269
 - Gino::ImageField, 289
- Response, 461
- ResponseFile, 466
- ResponseForbidden, 468
- ResponseJson, 470
- ResponseNotFound, 471

- ResponseServerError, [473](#)
- restore
 - Gino::AdminTable, [84](#)
 - Gino::DbManager, [166](#)
 - Gino::Plugin::mysql, [402](#)
 - Gino::Plugin::sqlsrv, [511](#)
- results
 - Gino::App::SearchSite::searchSite, [484](#)
- resultselect
 - Gino::DbManager, [167](#)
 - Gino::Plugin::mysql, [402](#)
 - Gino::Plugin::sqlsrv, [511](#)
- returnJoinLink
 - Gino::App::Auth::auth, [109](#)
- retvar
 - Gino::Form, [251](#)
- rewritePathInfo
 - Gino::Router, [477](#)
- rewriteQueryString
 - Gino::Router, [477](#)
- robots
 - Gino, [48](#)
- rollback
 - Gino::DbManager, [167](#)
 - Gino::Plugin::mysql, [402](#)
 - Gino::Plugin::sqlsrv, [511](#)
- route
 - Gino::Router, [477](#)
- Router, [474](#)
- SESSION_NAME
 - configuration.php, [590](#)
- SESSION_NOT_STARTED
 - Gino::Session, [488](#)
- SESSION_STARTED
 - Gino::Session, [489](#)
- SITE_APP
 - Gino, [54](#)
- SITE_CUSTOM_CKEDITOR
 - Gino, [54](#)
- SITE_GRAPHICS
 - Gino, [54](#)
- SITE_IMG
 - Gino, [55](#)
- SITE_JS
 - Gino, [55](#)
- SITE_LIB
 - Gino, [55](#)
- SITE_ROOT
 - Gino, [55](#)
 - include.php, [644](#)
- SITE_WWW
 - Gino, [55](#)
 - include.php, [644](#)
- save
 - Gino::App::Page::PageComment, [429](#)
 - Gino::App::Page::PageEntry, [434](#)
 - Gino::DataCache, [146](#)
 - Gino::Form, [253](#)
 - Gino::GImage, [270](#)
 - Gino::Model, [380](#)
 - Gino::TagField, [529](#)
- saveBlock
 - Gino::Template, [538](#)
- saveContentTags
 - Gino::GTag, [284](#)
- saveFile
 - Gino::FormField, [221](#)
 - Gino::ImageField, [290](#)
- savelImage
 - Gino::Form, [253](#)
 - Gino::ImageField, [290](#)
- savePassword
 - Gino::App::Auth::User, [556](#)
- saveTplImage
 - Gino::GImage, [270](#)
- savem2m
 - Gino::Model, [380](#)
- Search, [479](#)
- search
 - Gino::App::Auth::Ldap, [330](#)
- searchModules
 - Gino::App::Menu::menu, [363](#)
- searchNameFile
 - Gino, [48](#)
- searchSite, [481](#)
 - Gino::App::Page::page, [420](#)
- searchSiteResult
 - Gino::App::Page::page, [420](#)
- select
 - Gino::DbManager, [167](#)
 - Gino::Form, [253](#)
 - Gino::InputForm, [301](#)
 - Gino::Plugin::mysql, [402](#)
 - Gino::Plugin::sqlsrv, [511](#)
- selectTXT
 - Gino::Translation, [548](#)
- selectquery
 - Gino::DbManager, [168](#)
 - Gino::Plugin::mysql, [402](#)
 - Gino::Plugin::sqlsrv, [511](#)
- send
 - Gino::Http::Redirect, [453](#)
 - Gino::Http::Response, [464](#)
- sendContent
 - Gino::Http::Response, [464](#)
 - Gino::Http::ResponseForbidden, [470](#)
 - Gino::Http::ResponseNotFound, [473](#)
 - Gino::Http::ResponseServerError, [474](#)
- sendHeaders
 - Gino::Http::Redirect, [453](#)
 - Gino::Http::Response, [464](#)
 - Gino::Http::ResponseFile, [467](#)
- Session, [485](#)
- set_name_class
 - global.php, [636](#)
- setAddPath

- Gino::FileField, [221](#)
- setAutoIncrement
 - Gino::Field, [210](#)
- setAutoNow
 - Gino::DatetimeField, [152](#)
- setAutoNowAdd
 - Gino::DatetimeField, [152](#)
- setCheckIsConstraint
 - Gino::Model, [380](#)
- setCheckType
 - Gino::FileField, [221](#)
- setContent
 - Gino::App::PhpModuleView::PhpModule, [446](#)
 - Gino::Http::Response, [464](#)
- setContentType
 - Gino::Http::Response, [464](#)
- setCurrentPage
 - Gino::Paginator, [437](#)
- setData
 - Gino::Export, [197](#)
 - Gino::Options, [406](#)
- setDecimalDigits
 - Gino::FloatField, [227](#)
- setDefault
 - Gino::BooleanField, [113](#)
 - Gino::EnumField, [184](#)
 - Gino::Field, [210](#)
- setDirectory
 - Gino::FileField, [221](#)
- setDispositionType
 - Gino::Http::ResponseFile, [468](#)
- setEncoding
 - Gino::Http::Response, [465](#)
- setEnum
 - Gino::BooleanField, [113](#)
 - Gino::EnumField, [184](#)
- setExtensions
 - Gino::FileField, [222](#)
- setFields
 - Gino::Export, [197](#)
- setFilename
 - Gino::Css, [142](#)
 - Gino::Template, [538](#)
- setHead
 - Gino::Export, [198](#)
- setHeadVariables
 - Gino::Document, [178](#)
- setHeaders
 - Gino::Http::Response, [465](#)
- setHeight
 - Gino::ImageField, [290](#)
- setIfEmpty
 - Gino::Document, [179](#)
- setInstance
 - Gino::App::PhpModuleView::PhpModule, [446](#)
- setInstanceProperties
 - Gino::Controller, [135](#)
- setIntDigits
 - Gino::FloatField, [227](#)
- setIsConstraint
 - Gino::Model, [380](#)
- setLabel
 - Gino::Field, [210](#)
- setLanguage
 - Gino::Locale, [339](#)
- setLenght
 - Gino::Field, [210](#)
- setLists
 - Gino::Frontend, [261](#)
- setMaxFileSize
 - Gino::FileField, [222](#)
- setMergeValue
 - Gino::App::Auth::Group, [281](#)
 - Gino::App::Auth::User, [556](#)
- setMethod
 - Gino::Form, [254](#)
- setMoreInfo
 - Gino::App::Auth::User, [557](#)
- setName
 - Gino::Field, [211](#)
- setOption
 - Gino::Controller, [135](#)
- setOptions
 - Gino::Form, [254](#)
- setOrder
 - Gino::Export, [198](#)
- setPage
 - Gino::Test, [65](#)
- setPassword
 - Gino::App::Auth::User, [557](#)
- setPath
 - Gino::DirectoryField, [173](#)
 - Gino::FileField, [222](#)
- setPaths
 - Gino::Controller, [135](#)
- setPrefix
 - Gino::DirectoryField, [173](#)
 - Gino::FileField, [222](#)
- setPrefixFile
 - Gino::ImageField, [290](#)
- setPrefixThumb
 - Gino::ImageField, [291](#)
- setPrimaryKey
 - Gino::Field, [211](#)
- setRequired
 - Gino::Field, [211](#)
- setResize
 - Gino::ImageField, [291](#)
- setRids
 - Gino::Export, [198](#)
- setSeparator
 - Gino::Export, [198](#)
- setSessionSearch
 - Gino::AdminTable, [85](#)
- setSessionSearchAdd
 - Gino::AdminTable, [85](#)

- setStatus
 - Gino::Http::Response, 465
- setTable
 - Gino::Export, 200
- setThumb
 - Gino::ImageField, 291
- setThumbHeight
 - Gino::ImageField, 291
- setThumbWidth
 - Gino::ImageField, 291
- setTrnsI
 - Gino::CharField, 122
 - Gino::TextField, 542
- setTypesAllowed
 - Gino::FileField, 222
- setUniqueKey
 - Gino::Field, 211
- setUrlParams
 - Gino::Router, 478
- setUtf8
 - Gino::Plugin::mysql, 403
- setValidation
 - Gino::Form, 254
- setValue
 - Gino::Field, 211
- setValueType
 - Gino::Field, 213
- setViewTpl
 - Gino::View, 559
- setWidget
 - Gino::Field, 213
- setWidth
 - Gino::ImageField, 293
- setconnection
 - Gino::Plugin::mysql, 403
 - Gino::Plugin::sqlsrv, 512
- setnumberrows
 - Gino::Plugin::mysql, 403
 - Gino::Plugin::sqlsrv, 512
- setsql
 - Gino::Plugin::mysql, 403
 - Gino::Plugin::sqlsrv, 512
- settings.php, 641
- share
 - Gino, 49
- shareAll
 - Gino, 49
- sharethis
 - Gino::Javascript, 308
- shouldUpdate
 - Gino::Compressor, 127
- showcase
 - Gino::App::Page::page, 420
- Singleton, 489
- singleton
 - Gino::Loader, 332
- Skin, 491
- skinList
 - Gino::App::Layout::layout, 327
- slice
 - Gino::GImage, 270
- slimboxLib
 - Gino::Javascript, 308
- slimboxReplace
 - Gino, 49
- SlugField, 496
- sortUp
 - Gino::Skin, 495
- sqlcode, 499
- sqlsrv, 500
- stackTraceHtml
 - Gino::Logger, 343
- start
 - Gino::OutputCache, 409
- startSession
 - Gino::Session, 488
- statistics, 513
- stop
 - Gino::OutputCache, 409
- stream
 - Gino::GImage, 270
- strip_invisible_tags
 - Gino, 49
- strip_selected_tags
 - Gino, 50
- strip_tags_attributes
 - Gino, 50
- stripAll
 - Gino, 50
- stripEditor
 - Gino, 50
- structure
 - Gino::App::Attachment::AttachmentCtg, 95
 - Gino::App::Attachment::AttachmentItem, 99
 - Gino::App::Auth::Permission, 444
 - Gino::App::Auth::User, 557
 - Gino::App::Graphics::GraphicsItem, 278
 - Gino::App::Language::Lang, 316
 - Gino::App::Page::PageComment, 429
 - Gino::App::Page::PageEntry, 434
 - Gino::App::PhpModuleView::PhpModule, 447
 - Gino::App::Sysconf::Conf, 129
 - Gino::Model, 381
- summary
 - Gino::Paginator, 437
- sysClass, 516
- sysClassList
 - Gino::App::SysClass::sysClass, 521
- sysModulesManageArray
 - Gino::App::Index::index, 295
- sysconf, 522
- sysfunc, 524
- systemNewsletterList
 - Gino::App::Page::page, 420
- systemNewsletterRender
 - Gino::App::Page::page, 420

- systemVariablesHtml
 - Gino::Logger, [343](#)
- TBL_GROUP
 - Gino, [55](#)
- TBL_GROUP_PERMISSION
 - Gino, [55](#)
- TBL_LANGUAGE
 - Gino, [55](#)
- TBL_LOG_ACCESS
 - Gino, [55](#)
- TBL_MODULE
 - Gino, [55](#)
- TBL_MODULE_APP
 - Gino, [55](#)
- TBL_NATION
 - Gino, [55](#)
- TBL_PERMISSION
 - Gino, [56](#)
- TBL_SYS_CONF
 - Gino, [56](#)
- TBL_TRANSLATION
 - Gino, [56](#)
- TBL_USER
 - Gino, [56](#)
- TBL_USER_ADD
 - Gino, [56](#)
- TBL_USER_GROUP
 - Gino, [56](#)
- TBL_USER_PERMISSION
 - Gino, [56](#)
- TMP_DIR
 - Gino, [56](#)
- TPL_DIR
 - Gino, [56](#)
- tableexists
 - Gino::DbManager, [168](#)
 - Gino::Plugin::mysql, [403](#)
 - Gino::Plugin::sqlsrv, [512](#)
- tagCloud
 - Gino::TagField, [530](#)
- TagField, [527](#)
- Template, [530](#)
- templateList
 - Gino::App::Layout::layout, [327](#)
- templates/_popup.php, [642](#)
- templates/admin_home.php, [642](#)
- templates/admin_page.php, [642](#)
- templates/default_free_tpl.php, [643](#)
- templates/home.php, [643](#)
- templates/home_mobile.php, [643](#)
- templates/page.php, [643](#)
- templates/pages_mobile.php, [643](#)
- test/class.Paginator_test.php, [643](#)
- test/func_test.php, [643](#)
- test/include.php, [644](#)
- test_absolutePath
 - Gino::Test::FuncTest, [263](#)
- test_arrayToObject
 - Gino::Test::FuncTest, [263](#)
- test_baseFilename
 - Gino::Test::FuncTest, [263](#)
- test_checkEmail
 - Gino::Test::FuncTest, [263](#)
- test_cutHtmlText
 - Gino::Test::FuncTest, [263](#)
- test_cutString
 - Gino::Test::FuncTest, [263](#)
- test_dateDiff
 - Gino::Test::FuncTest, [263](#)
- test_dateToDbDate
 - Gino::Test::FuncTest, [263](#)
- test_dbDateToDate
 - Gino::Test::FuncTest, [263](#)
- test_dbDatetimeToDate
 - Gino::Test::FuncTest, [263](#)
- test_dbDatetimeToTime
 - Gino::Test::FuncTest, [263](#)
- test_dbNumberToNumber
 - Gino::Test::FuncTest, [263](#)
- test_extension
 - Gino::Test::FuncTest, [263](#)
- test_gOpt
 - Gino::Test::FuncTest, [263](#)
- test_getCurrentPage
 - Gino::Test::PaginatorTest, [439](#)
- test_getDateDiff
 - Gino::Test::FuncTest, [263](#)
- test_isValid
 - Gino::Test::FuncTest, [263](#)
- test_limit
 - Gino::Test::PaginatorTest, [439](#)
- test_limitQuery
 - Gino::Test::PaginatorTest, [439](#)
- test_numberToDB
 - Gino::Test::FuncTest, [263](#)
- test_pages
 - Gino::Test::PaginatorTest, [439](#)
- test_relativePath
 - Gino::Test::FuncTest, [263](#)
- test_searchNameFile
 - Gino::Test::FuncTest, [263](#)
- test_summary
 - Gino::Test::PaginatorTest, [439](#)
- test_timeDiff
 - Gino::Test::FuncTest, [263](#)
- test_timeToDbTime
 - Gino::Test::FuncTest, [263](#)
- test_traslitterazione
 - Gino::Test::FuncTest, [263](#)
- text
 - Gino::InputForm, [301](#)
- TextField, [539](#)
- textarea
 - Gino::Form, [255](#)
 - Gino::InputForm, [301](#)
- thumb

- Gino::GImage, 271
- timeDiff
 - Gino, 52
- TimeField, 543
- timeToDbTime
 - Gino, 52
- toKey
 - Gino::GImage, 271
- tplBlockForm
 - Gino::Template, 539
- transformPathQueryString
 - Gino::Router, 478
- Translation, 545
- traslitterazione
 - Gino, 52
- type
 - Gino::App::Attachment::AttachmentItem, 99
- union
 - Gino::DbManager, 168
 - Gino::Plugin::mysql, 403
 - Gino::Plugin::sqlsrv, 512
- update
 - Gino::DbManager, 169
 - Gino::Plugin::mysql, 403
 - Gino::Plugin::sqlsrv, 512
- updateOrderList
 - Gino::App::Menu::MenuVoice, 368
- updateStructure
 - Gino::Model, 381
- updateUrl
 - Gino::Http::Request, 460
- upgradeFolders
 - Gino::App::SysClass::sysClass, 522
- upload
 - Gino::Form, 255
- urlPage
 - Gino::Paginator, 437
- urlRewrite
 - Gino::Router, 478
- User, 548
- userCanSee
 - Gino::App::Menu::MenuVoice, 368
- userHasPerm
 - Gino::Controller, 135
- userLanguage
 - Gino::Locale, 339
- utf8_urldecode
 - Gino, 52
- VIEWS_DIR
 - Gino, 56
- validate
 - Gino::DirectoryField, 173
 - Gino::EmailField, 181
 - Gino::Field, 213
 - Gino::FileField, 224
- valueOrNull
 - Gino::Http::Request, 461
- variables
 - Gino::Registry, 457
- vendor
 - Gino::Javascript, 308
- verifyExtension
 - Gino, 53
- verifyFormToken
 - Gino::Form, 255
- View, 558
- view
 - Gino::App::Page::page, 422
- viewList
 - Gino::App::Layout::layout, 328
 - Gino::App::PhpModuleView::phpModuleView, 450
- viewlogAccess
 - Gino::App::Statistics::statistics, 515
- views/403.php, 644
- views/404.php, 645
- views/500.php, 646
- views/admin_table_form.php, 646
- views/admin_table_list.php, 647
- views/logger_stack_trace.php, 647
- views/logger_system_variables.php, 648
- views/paginator_navigator.php, 649
- views/paginator_pagination.php, 649
- views/related_contents_list.php, 650
- views/section.php, 650
- views/tab.php, 651
- views/table.php, 652
- warning
 - Gino::Error, 186
- write
 - Gino::Cache, 116
- writeFile
 - Gino::Export, 200
- YearField, 560