

FactDev

0.1

Generated by Doxygen 1.8.6

Sat Mar 28 2015 15:28:54

Contents

1	FactDev documentation	1
2	README	3
3	QTestRunner	5
4	Namespace Index	7
4.1	Namespace List	7
5	Hierarchical Index	9
5.1	Class Hierarchy	9
6	Class Index	13
6.1	Class List	13
7	Namespace Documentation	17
7.1	Databases Namespace Reference	17
7.1.1	Detailed Description	17
7.2	Gui Namespace Reference	17
7.2.1	Detailed Description	18
7.3	Gui::Widgets Namespace Reference	18
7.3.1	Detailed Description	18
7.4	Models Namespace Reference	18
7.4.1	Detailed Description	19
8	Class Documentation	21
8.1	Databases::AccessDatabase Class Reference	21
8.2	Gui::Dialogs::AddProjectDialog Class Reference	21
8.2.1	Detailed Description	22
8.2.2	Constructor & Destructor Documentation	22
8.2.2.1	AddProjectDialog	22
8.3	Gui::Dialogs::AddQuoteDialog Class Reference	22
8.3.1	Detailed Description	23
8.3.2	Constructor & Destructor Documentation	23

8.3.2.1	AddQuoteDialog	23
8.3.3	Member Function Documentation	24
8.3.3.1	fillQuoteBilling	24
8.3.3.2	getCopy	24
8.3.3.3	getIdCustomer	24
8.3.3.4	getNumber	24
8.3.3.5	setCopy	24
8.3.3.6	setQuoteIdNumber	25
8.4	Models::Billing Class Reference	25
8.4.1	Detailed Description	27
8.4.2	Constructor & Destructor Documentation	27
8.4.2.1	Billing	27
8.4.3	Member Function Documentation	27
8.4.3.1	addContributory	27
8.4.3.2	getContributories	27
8.4.3.3	getDataMap	27
8.4.3.4	getDate	27
8.4.3.5	getDescription	28
8.4.3.6	getFilename	28
8.4.3.7	getFolder	28
8.4.3.8	getItem	28
8.4.3.9	getNumber	28
8.4.3.10	getPath	28
8.4.3.11	getSumQuantity	28
8.4.3.12	getSumRate	29
8.4.3.13	getTitle	29
8.4.3.14	hydrat	29
8.4.3.15	isBilling	29
8.4.3.16	isPaid	29
8.4.3.17	operator!=	29
8.4.3.18	operator<	29
8.4.3.19	operator==	30
8.4.3.20	setDate	30
8.4.3.21	setDescription	30
8.4.3.22	setIsBilling	30
8.4.3.23	setIsPaid	30
8.4.3.24	setNumber	30
8.4.3.25	setTitle	31
8.5	Databases::BillingDatabase Class Reference	31
8.5.1	Detailed Description	32

8.5.2	Member Function Documentation	32
8.5.2.1	addBilling	32
8.5.2.2	addBillingProject	32
8.5.2.3	getAllBillingsOfProject	33
8.5.2.4	getAllBillingsOnly	33
8.5.2.5	getBilling	33
8.5.2.6	getBilling	33
8.5.2.7	getBillings	33
8.5.2.8	getBillingsBetweenDates	34
8.5.2.9	getBillingsTable	34
8.5.2.10	getMaxBillingNumber	34
8.5.2.11	getMaxBillingNumberOfCustomer	34
8.5.2.12	getMaxQuoteNumber	35
8.5.2.13	getMaxQuoteNumberOfCustomer	35
8.5.2.14	instance	35
8.5.2.15	isBillingPaid	35
8.5.2.16	removeBilling	35
8.5.2.17	removeBillingProject	36
8.6	BillingDatabaseTest Class Reference	36
8.7	BillingModelTest Class Reference	36
8.8	Gui::Widgets::WdgModels::BillingsTableModel Class Reference	37
8.8.1	Detailed Description	37
8.8.2	Constructor & Destructor Documentation	38
8.8.2.1	BillingsTableModel	38
8.8.3	Member Function Documentation	39
8.8.3.1	append	39
8.8.3.2	columnCount	39
8.8.3.3	count	39
8.8.3.4	data	39
8.8.3.5	flags	39
8.8.3.6	getBillings	40
8.8.3.7	headerData	40
8.8.3.8	remove	40
8.8.3.9	rowCount	40
8.8.3.10	setData	40
8.9	Gui::Widgets::CheckFields::CheckCity Class Reference	41
8.9.1	Detailed Description	41
8.9.2	Constructor & Destructor Documentation	41
8.9.2.1	CheckCity	41
8.10	Gui::Widgets::CheckFields::CheckCountry Class Reference	41

8.10.1 Detailed Description	42
8.10.2 Constructor & Destructor Documentation	42
8.10.2.1 CheckCountry	42
8.11 Gui::Widgets::CheckFields::CheckEmail Class Reference	42
8.11.1 Detailed Description	43
8.11.2 Constructor & Destructor Documentation	43
8.11.2.1 CheckEmail	43
8.11.3 Member Function Documentation	43
8.11.3.1 check	43
8.12 Gui::Widgets::CheckFields::CheckFieldsLetters Class Reference	44
8.12.1 Detailed Description	44
8.12.2 Constructor & Destructor Documentation	44
8.12.2.1 CheckFieldsLetters	44
8.12.3 Member Function Documentation	44
8.12.3.1 check	44
8.13 Gui::Widgets::CheckFields::CheckFieldsNumbers Class Reference	45
8.13.1 Detailed Description	45
8.13.2 Constructor & Destructor Documentation	45
8.13.2.1 CheckFieldsNumbers	45
8.13.3 Member Function Documentation	46
8.13.3.1 check	46
8.14 Gui::Widgets::CheckFields::CheckIpAddress Class Reference	46
8.14.1 Detailed Description	46
8.14.2 Constructor & Destructor Documentation	47
8.14.2.1 CheckIpAddress	47
8.14.3 Member Function Documentation	47
8.14.3.1 check	47
8.15 Gui::Widgets::CheckFields::CheckLogin Class Reference	47
8.15.1 Detailed Description	48
8.15.2 Constructor & Destructor Documentation	48
8.15.2.1 CheckLogin	48
8.15.3 Member Function Documentation	48
8.15.3.1 check	48
8.16 Gui::Widgets::CheckFields::CheckName Class Reference	49
8.16.1 Detailed Description	49
8.16.2 Constructor & Destructor Documentation	49
8.16.2.1 CheckName	49
8.17 Gui::Widgets::CheckFields::CheckPhone Class Reference	49
8.17.1 Detailed Description	50
8.17.2 Constructor & Destructor Documentation	50

8.17.2.1	CheckPhone	50
8.17.3	Member Function Documentation	50
8.17.3.1	check	50
8.17.3.2	getCountry	51
8.17.3.3	setCountry	51
8.18	Gui::Widgets::CheckFields::CheckPortNumber Class Reference	51
8.18.1	Detailed Description	51
8.18.2	Constructor & Destructor Documentation	52
8.18.2.1	CheckPortNumber	52
8.18.3	Member Function Documentation	52
8.18.3.1	check	52
8.19	Gui::Widgets::CheckFields::CheckPostalCode Class Reference	52
8.19.1	Detailed Description	53
8.19.2	Constructor & Destructor Documentation	53
8.19.2.1	CheckPostalCode	53
8.19.3	Member Function Documentation	53
8.19.3.1	check	53
8.19.3.2	getCountry	53
8.19.3.3	setCountry	53
8.20	Gui::Widgets::CheckFields::CheckQLineEdit Class Reference	54
8.20.1	Detailed Description	54
8.20.2	Constructor & Destructor Documentation	54
8.20.2.1	CheckQLineEdit	54
8.20.3	Member Function Documentation	55
8.20.3.1	getBtnValid	55
8.20.3.2	isValid	55
8.20.3.3	setBtnValid	55
8.21	Gui::Widgets::CheckFields::CheckSiretNumber Class Reference	55
8.21.1	Detailed Description	56
8.21.2	Constructor & Destructor Documentation	56
8.21.2.1	CheckSiretNumber	56
8.21.3	Member Function Documentation	56
8.21.3.1	check	56
8.22	Gui::Widgets::CheckFields::CheckUntilField Class Reference	56
8.22.1	Detailed Description	57
8.22.2	Constructor & Destructor Documentation	57
8.22.2.1	CheckUntilField	57
8.22.3	Member Function Documentation	57
8.22.3.1	check	57
8.23	Gui::Widgets::CheckFields::CheckValidField Class Reference	57

8.23.1 Detailed Description	58
8.23.2 Constructor & Destructor Documentation	58
8.23.2.1 CheckValidField	58
8.23.3 Member Function Documentation	58
8.23.3.1 check	58
8.24 Gui::Widgets::Delegates::ComboBoxDelegate Class Reference	59
8.24.1 Detailed Description	59
8.24.2 Constructor & Destructor Documentation	59
8.24.2.1 ComboBoxDelegate	59
8.24.3 Member Function Documentation	60
8.24.3.1 createEditor	60
8.24.3.2 paint	60
8.24.3.3 setEditorData	60
8.24.3.4 setModelData	60
8.24.3.5 updateEditorGeometry	61
8.25 Gui::Widgets::ComboBoxModelWidget Class Reference	61
8.25.1 Detailed Description	61
8.25.2 Constructor & Destructor Documentation	61
8.25.2.1 ComboBoxModelWidget	61
8.26 Gui::Dialogs::ComputeTurnoverDialog Class Reference	62
8.26.1 Detailed Description	62
8.26.2 Member Function Documentation	62
8.26.2.1 beginDateControl	62
8.26.2.2 endDateControl	63
8.26.2.3 fillLabels	63
8.27 Mustache::Context Class Reference	63
8.27.1 Detailed Description	64
8.27.2 Constructor & Destructor Documentation	64
8.27.2.1 Context	64
8.27.3 Member Function Documentation	64
8.27.3.1 canEval	64
8.27.3.2 eval	64
8.27.3.3 isFalse	64
8.27.3.4 listCount	64
8.27.3.5 partialResolver	64
8.27.3.6 partialValue	64
8.27.3.7 pop	65
8.27.3.8 push	65
8.27.3.9 stringValue	65
8.28 ContributoriesDatabaseTest Class Reference	65

8.29 Models::ContributoriesList Class Reference	65
8.29.1 Detailed Description	66
8.29.2 Member Function Documentation	66
8.29.2.1 addContributory	66
8.29.2.2 addProject	67
8.29.2.3 getAllContributories	67
8.29.2.4 getContributories	67
8.29.2.5 getCustomer	67
8.29.2.6 getDataMap	67
8.29.2.7 getIdBilling	67
8.29.2.8 getNbProjects	68
8.29.2.9 getProjects	68
8.29.2.10 getRate	68
8.29.2.11 getSumQuantity	68
8.29.2.12 getSumRate	68
8.29.2.13 isInsert	68
8.29.2.14 setAllIdContributories	68
8.29.2.15 setIdBilling	69
8.29.2.16 setInsert	69
8.30 Gui::Widgets::WdgModels::ContributoriesTableModel Class Reference	69
8.30.1 Detailed Description	70
8.30.2 Constructor & Destructor Documentation	70
8.30.2.1 ContributoriesTableModel	70
8.30.3 Member Function Documentation	70
8.30.3.1 append	70
8.30.3.2 columnCount	70
8.30.3.3 count	71
8.30.3.4 data	71
8.30.3.5 flags	71
8.30.3.6 getContributories	71
8.30.3.7 getSumQuantity	71
8.30.3.8 headerData	71
8.30.3.9 remove	72
8.30.3.10 rowCount	72
8.30.3.11 setData	72
8.31 Gui::Widgets::ContributoriesWidget Class Reference	72
8.31.1 Detailed Description	73
8.31.2 Constructor & Destructor Documentation	73
8.31.2.1 ContributoriesWidget	73
8.31.3 Member Function Documentation	74

8.31.3.1	add	74
8.31.3.2	addProject	74
8.31.3.3	count	74
8.31.3.4	getContributories	74
8.32	Models::Contributory Class Reference	74
8.32.1	Detailed Description	75
8.32.2	Constructor & Destructor Documentation	76
8.32.2.1	Contributory	76
8.32.3	Member Function Documentation	77
8.32.3.1	getDataMap	77
8.32.3.2	getDescription	77
8.32.3.3	getNbHours	77
8.32.3.4	getProject	77
8.32.3.5	hydrat	77
8.32.3.6	operator!=	77
8.32.3.7	operator==	78
8.32.3.8	setDescription	78
8.32.3.9	setNbHours	78
8.32.3.10	setProject	78
8.33	Databases::ContributoryDatabase Class Reference	78
8.33.1	Detailed Description	79
8.33.2	Member Function Documentation	79
8.33.2.1	addContributory	79
8.33.2.2	getContributoriesByBilling	80
8.33.2.3	getContributoriesByBillingAndProject	81
8.33.2.4	getContributory	81
8.33.2.5	getContributory	81
8.33.2.6	instance	81
8.33.2.7	removeContributory	82
8.34	ContributoryListTest Class Reference	83
8.35	ContributoryModelTest Class Reference	83
8.36	CounterContext Class Reference	83
8.36.1	Member Function Documentation	84
8.36.1.1	canEval	84
8.36.1.2	canEval	84
8.36.1.3	eval	84
8.36.1.4	eval	84
8.36.1.5	stringValue	85
8.36.1.6	stringValue	85
8.37	Models::Customer Class Reference	85

8.37.1 Detailed Description	86
8.37.2 Constructor & Destructor Documentation	86
8.37.2.1 Customer	86
8.37.3 Member Function Documentation	86
8.37.3.1 getDataMap	86
8.37.3.2 getNameFolder	86
8.37.3.3 getPath	86
8.37.3.4 getTurnover	87
8.37.3.5 hydrat	87
8.37.3.6 isArchived	87
8.37.3.7 setIsArchived	87
8.38 Gui::Widgets::CustomerContextualMenu Class Reference	87
8.38.1 Detailed Description	88
8.38.2 Constructor & Destructor Documentation	88
8.38.2.1 CustomerContextualMenu	88
8.39 Databases::CustomerDatabase Class Reference	88
8.39.1 Detailed Description	89
8.39.2 Member Function Documentation	89
8.39.2.1 addCustomer	89
8.39.2.2 getCustomer	89
8.39.2.3 getCustomer	90
8.39.2.4 getCustomersTable	90
8.39.2.5 getItemCustomer	90
8.39.2.6 getItemProject	90
8.39.2.7 getItemRoot	91
8.39.2.8 getNbCustomers	91
8.39.2.9 getTree	91
8.39.2.10 instance	91
8.39.2.11 removeCustomer	91
8.39.2.12 updateCustomer	92
8.40 CustomerDatabaseTest Class Reference	92
8.41 Gui::Widgets::CustomerDataWidget Class Reference	92
8.41.1 Detailed Description	93
8.41.2 Constructor & Destructor Documentation	93
8.41.2.1 CustomerDataWidget	93
8.41.3 Member Function Documentation	93
8.41.3.1 printInformations	93
8.42 CustomerModelTest Class Reference	93
8.43 Gui::Widgets::WdgModels::CustomersTableModel Class Reference	94
8.43.1 Detailed Description	94

8.43.2	Constructor & Destructor Documentation	95
8.43.2.1	CustomersTableModel	95
8.43.3	Member Function Documentation	96
8.43.3.1	append	96
8.43.3.2	columnCount	96
8.43.3.3	count	96
8.43.3.4	data	96
8.43.3.5	flags	96
8.43.3.6	getCustomers	97
8.43.3.7	headerData	97
8.43.3.8	remove	97
8.43.3.9	rowCount	97
8.43.3.10	setData	97
8.44	Databases::Database Class Reference	98
8.44.1	Detailed Description	99
8.44.2	Member Function Documentation	99
8.44.2.1	changeDatabase	99
8.44.2.2	executeFile	99
8.44.2.3	instance	99
8.44.2.4	lastError	100
8.44.2.5	setDatabase	101
8.44.2.6	value	101
8.45	Gui::Widgets::DatabaseSettingsWidget Class Reference	101
8.45.1	Detailed Description	102
8.45.2	Constructor & Destructor Documentation	102
8.45.2.1	DatabaseSettingsWidget	102
8.45.3	Member Function Documentation	102
8.45.3.1	getDatabaseName	102
8.45.3.2	getDomainNameOrIP	103
8.45.3.3	getLogin	103
8.45.3.4	getPassword	103
8.45.3.5	getPort	103
8.45.3.6	isValid	103
8.46	Exceptions::DbException Class Reference	103
8.46.1	Detailed Description	104
8.46.2	Constructor & Destructor Documentation	104
8.46.2.1	DbException	104
8.46.3	Member Function Documentation	104
8.46.3.1	popupMessage	104
8.47	Gui::Dialogs::DialogAddCustomer Class Reference	105

8.47.1 Detailed Description	105
8.47.2 Constructor & Destructor Documentation	105
8.47.2.1 DialogAddCustomer	105
8.48 Utils::Directories Class Reference	106
8.48.1 Member Function Documentation	106
8.48.1.1 makeDirectory	106
8.49 Gui::Widgets::Delegates::DoubleSpinBoxDelegate Class Reference	106
8.49.1 Detailed Description	107
8.49.2 Constructor & Destructor Documentation	107
8.49.2.1 DoubleSpinBoxDelegate	107
8.49.3 Member Function Documentation	107
8.49.3.1 createEditor	107
8.49.3.2 setEditorData	107
8.49.3.3 setModelData	108
8.49.3.4 updateEditorGeometry	109
8.50 Exceptions::FileException Class Reference	109
8.50.1 Detailed Description	109
8.50.2 Constructor & Destructor Documentation	109
8.50.2.1 FileException	109
8.50.3 Member Function Documentation	110
8.50.3.1 popupMessage	110
8.51 Generation Class Reference	110
8.52 Utils::HierarchicalSystem Class Reference	110
8.52.1 Detailed Description	111
8.52.2 Member Function Documentation	111
8.52.2.1 addBillingToProject	111
8.52.2.2 addProjectToCustomer	111
8.52.2.3 getCustomers	111
8.52.2.4 getProjects	112
8.53 Gui::Widgets::CheckFields::!CheckField Class Reference	112
8.53.1 Detailed Description	112
8.53.2 Member Function Documentation	112
8.53.2.1 check	112
8.54 Models::!Model Class Reference	113
8.54.1 Detailed Description	113
8.54.2 Member Function Documentation	114
8.54.2.1 getDataMap	114
8.54.2.2 getIId	114
8.54.2.3 hydrat	114
8.54.2.4 isToRemoved	114

8.54.2.5	setId	114
8.54.2.6	setToRemoved	114
8.55	Utils::ItemType Class Reference	115
8.55.1	Detailed Description	115
8.55.2	Constructor & Destructor Documentation	115
8.55.2.1	ItemType	115
8.55.3	Member Function Documentation	115
8.55.3.1	getModel	115
8.55.3.2	getName	116
8.55.3.3	getType	116
8.55.3.4	setName	116
8.55.3.5	setType	116
8.56	ItemTypeTest Class Reference	116
8.57	Utils::Log Class Reference	117
8.57.1	Detailed Description	117
8.57.2	Member Function Documentation	117
8.57.2.1	instance	117
8.57.2.2	write	117
8.57.3	Friends And Related Function Documentation	118
8.57.3.1	operator<<	118
8.58	Gui::MainWindow Class Reference	118
8.58.1	Detailed Description	120
8.58.2	Constructor & Destructor Documentation	120
8.58.2.1	MainWindow	120
8.58.3	Member Function Documentation	120
8.58.3.1	addBill	120
8.58.3.2	addDoc	120
8.58.3.3	addProject	121
8.58.3.4	addQuote	121
8.58.3.5	editUser	121
8.58.3.6	getCurrentCustomerId	121
8.58.3.7	getCurrentCustomerName	121
8.58.3.8	getCurrentProjectId	121
8.58.3.9	getCurrentProjectName	121
8.58.3.10	getCurrentQuoteId	122
8.58.3.11	resizeEvent	122
8.58.3.12	rootTree	122
8.58.3.13	search	122
8.58.3.14	treeLevel	122
8.59	Gui::Dialogs::MessageBox Class Reference	122

8.59.1 Detailed Description	123
8.59.2 Constructor & Destructor Documentation	123
8.59.2.1 MessageBox	123
8.59.3 Member Function Documentation	123
8.59.3.1 setImage	123
8.59.3.2 setText	124
8.60 Parameters Class Reference	124
8.60.1 Detailed Description	124
8.60.2 Member Data Documentation	124
8.60.2.1 DB_FILENAME	124
8.60.2.2 VERSION	124
8.61 Mustache::PartialFileLoader Class Reference	125
8.61.1 Detailed Description	125
8.61.2 Member Function Documentation	125
8.61.2.1 getPartial	125
8.62 Mustache::PartialMap Class Reference	125
8.62.1 Detailed Description	126
8.62.2 Member Function Documentation	126
8.62.2.1 getPartial	126
8.63 Mustache::PartialResolver Class Reference	126
8.63.1 Detailed Description	126
8.63.2 Member Function Documentation	126
8.63.2.1 getPartial	126
8.64 Generator::PdfGenerator Class Reference	127
8.64.1 Detailed Description	127
8.64.2 Constructor & Destructor Documentation	127
8.64.2.1 PdfGenerator	127
8.64.3 Member Function Documentation	127
8.64.3.1 generate	127
8.65 Models::People Class Reference	127
8.65.1 Detailed Description	129
8.65.2 Member Function Documentation	129
8.65.2.1 getAddress	129
8.65.2.2 getCity	129
8.65.2.3 getCompany	129
8.65.2.4 getCountry	129
8.65.2.5 getEmail	130
8.65.2.6 getFax	130
8.65.2.7 getFirstname	130
8.65.2.8 getLastname	130

8.65.2.9	getMobilePhone	130
8.65.2.10	getPhone	130
8.65.2.11	getPostalCode	130
8.65.2.12	operator!=	131
8.65.2.13	operator==	132
8.65.2.14	setAddress	132
8.65.2.15	setCity	132
8.65.2.16	setCompany	132
8.65.2.17	setCountry	132
8.65.2.18	setEmail	132
8.65.2.19	setFax	133
8.65.2.20	setFirstname	133
8.65.2.21	setLastname	133
8.65.2.22	setMobilePhone	133
8.65.2.23	setPhone	133
8.65.2.24	setPostalCode	133
8.66	Utils::pointers Class Reference	134
8.67	PointersTest Class Reference	134
8.68	Gui::Widgets::Popup Class Reference	134
8.68.1	Detailed Description	134
8.69	Models::Project Class Reference	135
8.69.1	Detailed Description	136
8.69.2	Constructor & Destructor Documentation	136
8.69.2.1	Project	136
8.69.3	Member Function Documentation	136
8.69.3.1	getBeginDate	136
8.69.3.2	getCost	137
8.69.3.3	getCost	137
8.69.3.4	getCustomer	137
8.69.3.5	getDailyRate	137
8.69.3.6	getDataMap	137
8.69.3.7	getDescription	137
8.69.3.8	getEndDate	138
8.69.3.9	getName	138
8.69.3.10	hydrat	138
8.69.3.11	operator!=	138
8.69.3.12	operator<	138
8.69.3.13	operator==	138
8.69.3.14	setBeginDate	139
8.69.3.15	setCost	139

8.69.3.16	setCustomer	139
8.69.3.17	setDailyRate	139
8.69.3.18	setDescription	139
8.69.3.19	setEndDate	139
8.69.3.20	setName	140
8.70	Gui::Widgets::Delegates::ProjectComboDelegate Class Reference	140
8.70.1	Detailed Description	141
8.70.2	Constructor & Destructor Documentation	141
8.70.2.1	ProjectComboDelegate	141
8.70.3	Member Function Documentation	141
8.70.3.1	createEditor	141
8.70.3.2	getProjects	141
8.70.3.3	isLocked	141
8.70.3.4	paint	141
8.70.3.5	setLocked	142
8.71	Gui::Widgets::WdgModels::ProjectContributoriesTableModel Class Reference	142
8.71.1	Detailed Description	143
8.71.2	Constructor & Destructor Documentation	143
8.71.2.1	ProjectContributoriesTableModel	143
8.71.3	Member Function Documentation	143
8.71.3.1	allProjectsChosen	143
8.71.3.2	append	143
8.71.3.3	columnCount	144
8.71.3.4	data	144
8.71.3.5	flags	144
8.71.3.6	getProject	144
8.71.3.7	getProjects	144
8.71.3.8	getSelectedProjects	145
8.71.3.9	headerData	145
8.71.3.10	remove	145
8.71.3.11	rowCount	145
8.71.3.12	setData	145
8.72	Databases::ProjectDatabase Class Reference	146
8.72.1	Detailed Description	147
8.72.2	Member Function Documentation	147
8.72.2.1	addProject	147
8.72.2.2	getAllProjects	147
8.72.2.3	getCostProjects	147
8.72.2.4	getNbProjects	147
8.72.2.5	getNbProjectsForACustomer	147

8.72.2.6	getProject	148
8.72.2.7	getProject	148
8.72.2.8	getProjects	148
8.72.2.9	getProjectsOfCustomer	148
8.72.2.10	getProjectsTable	149
8.72.2.11	instance	150
8.72.2.12	removeProject	150
8.72.2.13	updateProject	150
8.73	ProjectDatabaseTest Class Reference	150
8.74	ProjectModelTest Class Reference	151
8.75	Gui::Widgets::WdgModels::ProjectsTableModel Class Reference	151
8.75.1	Detailed Description	152
8.75.2	Constructor & Destructor Documentation	152
8.75.2.1	ProjectsTableModel	152
8.75.3	Member Function Documentation	152
8.75.3.1	append	152
8.75.3.2	columnCount	152
8.75.3.3	count	152
8.75.3.4	data	152
8.75.3.5	flags	153
8.75.3.6	getProjects	153
8.75.3.7	headerData	153
8.75.3.8	remove	153
8.75.3.9	rowCount	154
8.75.3.10	setData	154
8.76	Gui::Widgets::ProjectsWidget Class Reference	154
8.76.1	Detailed Description	155
8.76.2	Constructor & Destructor Documentation	155
8.76.2.1	ProjectsWidget	155
8.76.3	Member Function Documentation	155
8.76.3.1	updateBtn	155
8.77	Mustache::QtVariantContext Class Reference	155
8.77.1	Detailed Description	156
8.77.2	Member Typedef Documentation	156
8.77.2.1	fn_t	156
8.77.3	Member Function Documentation	156
8.77.3.1	canEval	156
8.77.3.2	eval	156
8.77.3.3	isFalse	157
8.77.3.4	listCount	157

8.77.3.5	pop	157
8.77.3.6	push	157
8.77.3.7	stringValue	157
8.78	Models::Rate Class Reference	157
8.78.1	Detailed Description	158
8.78.2	Constructor & Destructor Documentation	158
8.78.2.1	Rate	158
8.78.2.2	Rate	158
8.78.3	Member Function Documentation	158
8.78.3.1	getDailyRate	158
8.78.3.2	getHourlyRate	159
8.78.3.3	getLegalRate	159
8.78.3.4	getNbDailyHours	159
8.79	Databases::RateDatabase Class Reference	159
8.79.1	Member Function Documentation	160
8.79.1.1	addRateProject	160
8.79.1.2	getRate	160
8.79.1.3	instance	160
8.79.1.4	updateRateProject	160
8.80	RateModelTest Class Reference	161
8.81	Gui::Widgets::RateWidget Class Reference	161
8.81.1	Detailed Description	162
8.81.2	Constructor & Destructor Documentation	162
8.81.2.1	RateWidget	162
8.81.3	Member Function Documentation	162
8.81.3.1	getDailyRate	162
8.81.3.2	getHourlyRate	162
8.81.3.3	setDailyRate	162
8.81.3.4	setHourlyRate	162
8.81.3.5	setWidgetDailyRateValue	162
8.81.3.6	setWidgetHourlyRateValue	163
8.82	Mustache::Renderer Class Reference	163
8.82.1	Detailed Description	163
8.82.2	Member Function Documentation	163
8.82.2.1	error	163
8.82.2.2	errorPartial	163
8.82.2.3	errorPos	163
8.82.2.4	render	164
8.82.2.5	setTagMarkers	164
8.83	Models::Search Class Reference	164

8.83.1 Detailed Description	165
8.83.2 Member Function Documentation	165
8.83.2.1 filterOnBillsOrQuotes	165
8.83.2.2 filterOnCompany	165
8.83.2.3 filterOnContributories	165
8.83.2.4 filterOnCustomersWithoutProject	166
8.83.2.5 filterOnNumberElements	166
8.83.2.6 filterOnProjects	166
8.83.2.7 filterOnReferentLastname	166
8.83.2.8 filterOnVarcharElements	166
8.83.2.9 getFilter	167
8.83.2.10 getGroupFilter	167
8.83.2.11 getSearchInBillsQuotes	167
8.83.2.12 getSearchInCompanies	167
8.83.2.13 getSearchInProjects	167
8.83.2.14 getSearchInReferentLastname	167
8.83.2.15 getText	167
8.83.2.16 searchInContributories	168
8.83.2.17 setGroupFilter	168
8.83.2.18 setSearchInBillsQuotes	168
8.83.2.19 setSearchInCompanies	168
8.83.2.20 setSearchInContributories	168
8.83.2.21 setSearchInProjects	168
8.83.2.22 setSearchInReferentLastname	168
8.83.2.23 setText	169
8.84 Gui::Docks::SearchDock Class Reference	169
8.84.1 Detailed Description	169
8.84.2 Constructor & Destructor Documentation	170
8.84.2.1 SearchDock	170
8.84.3 Member Function Documentation	171
8.84.3.1 search	171
8.84.3.2 textChanged	171
8.85 searchTest Class Reference	171
8.86 Gui::Widgets::searchWidget Class Reference	171
8.86.1 Detailed Description	172
8.86.2 Constructor & Destructor Documentation	172
8.86.2.1 searchWidget	172
8.86.3 Member Function Documentation	173
8.86.3.1 getCurrentCustomerId	173
8.86.3.2 getIdCustomer	173

8.86.3.3	isCustomerSelected	173
8.86.3.4	search	173
8.86.3.5	selectCustomer	173
8.86.3.6	setIdCustomer	173
8.87	Gui::Dialogs::StartedWindowsDialog Class Reference	174
8.87.1	Detailed Description	175
8.87.2	Constructor & Destructor Documentation	175
8.87.2.1	StartedWindowsDialog	175
8.87.3	Member Function Documentation	175
8.87.3.1	databaseTypeChanged	175
8.87.3.2	getImage	175
8.87.3.3	isDatabaseCentralized	175
8.87.3.4	isDatabaseTypeValid	176
8.88	Models::Statistics Class Reference	176
8.88.1	Member Function Documentation	176
8.88.1.1	getTurnoverBetweenDates	176
8.89	Utils::String Class Reference	176
8.89.1	Detailed Description	177
8.89.2	Member Function Documentation	177
8.89.2.1	firstLetterToUpper	177
8.90	StringTest Class Reference	177
8.91	Mustache::Tag Struct Reference	177
8.91.1	Detailed Description	178
8.91.2	Member Enumeration Documentation	178
8.91.2.1	Type	178
8.92	TestAdder< T > Class Template Reference	178
8.93	testadder Class Reference	178
8.94	TestMustache Class Reference	179
8.95	TestRunner Class Reference	179
8.96	Generator::TexGenerator Class Reference	179
8.96.1	Detailed Description	179
8.96.2	Constructor & Destructor Documentation	180
8.96.2.1	TexGenerator	180
8.96.3	Member Function Documentation	181
8.96.3.1	generate	181
8.97	Gui::Widgets::Delegates::TextareaDelegate Class Reference	181
8.97.1	Detailed Description	181
8.97.2	Constructor & Destructor Documentation	182
8.97.2.1	TextareaDelegate	182
8.97.3	Member Function Documentation	182

8.97.3.1	createEditor	182
8.97.3.2	setEditorData	182
8.97.3.3	setModelData	182
8.97.3.4	updateEditorGeometry	182
8.98	Gui::Widgets::Delegates::UnitComboDelegate Class Reference	183
8.98.1	Detailed Description	183
8.98.2	Constructor & Destructor Documentation	183
8.98.2.1	UnitComboDelegate	183
8.98.3	Member Function Documentation	184
8.98.3.1	createEditor	184
8.98.3.2	paint	184
8.99	Models::User Class Reference	184
8.99.1	Detailed Description	185
8.99.2	Constructor & Destructor Documentation	186
8.99.2.1	User	186
8.99.3	Member Function Documentation	187
8.99.3.1	getDataMap	187
8.99.3.2	getNoSiret	187
8.99.3.3	getTitle	187
8.99.3.4	getWorkspaceName	187
8.99.3.5	getWorkspacePath	187
8.99.3.6	hydrat	187
8.99.3.7	operator!=	188
8.99.3.8	operator==	188
8.99.3.9	setNoSiret	188
8.99.3.10	setTitle	188
8.99.3.11	setWorkspaceName	188
8.99.3.12	setWorkspacePath	189
8.99.3.13	updateFolders	189
8.100	Databases::UserDatabase Class Reference	189
8.100.1	Detailed Description	190
8.100.2	Member Function Documentation	190
8.100.2.1	getUser	190
8.100.2.2	instance	190
8.101	UserDatabaseTest Class Reference	190
8.102	Gui::Dialogs::UserDataDialog Class Reference	191
8.102.1	Detailed Description	191
8.102.2	Constructor & Destructor Documentation	191
8.102.2.1	UserDataDialog	191
8.103	UserModelTest Class Reference	192

Chapter 1

FactDev documentation

This website contains the documentation of FactDev, a billings and quotes software.

Team

FACT team is a team for University Project. Members of this team are :

- Florent Berbie ([KraTuX31](#))
- Antoine de Roquemaurel ([aroquemaurel](#))
- Cédric Rohaut ([Oxynos](#))
- Manantsoa Andriamihary Razanajatovo ([manantsoa](#))

For more informations, you can go to <http://fact-team.github.io>.

What is FactDev ?

FactDev is a software for quotes and billings developped by FACT team for university project in Toulouse III University – Paul Sabatier.

This software is developed with C++ and Qt framework, and there is differents features :

- Customers database
- Differents projects for customers
- Quotes
- Billings

Documentation

You can access to the documentation generated by Doxygen here :

- [HTML Documentation](#)
- [PDF Documentation](#)

Installation and using

- [User Manual](#)

Chapter 2

README

! [Build Status] (<https://travis-ci.org/robertknight/qt-mustache.svg?branch=master>)

Qt Mustache

qt-mustache is a simple library for rendering **Mustache templates**.

Example Usage

```
““cpp #include "mustache.h"
```

```
QVariantHash contact; contact["name"] = "John Smith"; contact["email"] = "john.smith@gmail.com";
```

```
QString contactTemplate = "<b>{{name}}</b> <a href='mailto:{{email}}'>{{email}}";
```

```
Mustache::Renderer renderer; Mustache::QtVariantContext context(contact);
```

```
QTextStream output(stdout); output << renderer.render(contactTemplate, &context); ““
```

Outputs: John Smith john.smith@gmail.com

For further examples, see the tests in `test_mustache.cpp`

Building

- To build the tests, run `qmake` followed by `make`
- To use qt-mustache in your project, just add the `mustache.h` and `mustache.cpp` files to your project.

License

qt-mustache is licensed under the BSD license.

Dependencies

qt-mustache depends on the QtCore library. It is compatible with Qt 4 and Qt 5.

Usage

Syntax

qt-mustache uses the standard Mustache syntax. See the [Mustache manual](#) for details.

Data Sources

qt-mustache expands Mustache tags using values from a `Mustache::Context`. `Mustache::QtVariantContext` is a simple context implementation which wraps a `QVariantHash` or `QVariantMap`. If you want to render a template using a custom data source, you can either create a `QVariantHash` which mirrors the data source or you can re-implement `Mustache::Context`.

Partials

When a `{{>partial}}` Mustache tag is encountered, qt-mustache will attempt to load the partial using a `Mustache::PartialResolver` provided by the context. `Mustache::PartialMap` is a simple resolver which takes a `QHash<QString,QString>` map of partial names to values and looks up partials in that map. `Mustache::PartialFileLoader` is another simple resolver which fetches partials from `<partial name>.mustache` files in a specified directory.

You can re-implement the `Mustache::PartialResolver` interface if you want to load partials from a custom source (eg. a database).

Error Handling

If an error occurs when rendering a template, `Mustache::Renderer::errorPosition()` is set to non-negative value and template rendering stops. If the error occurs whilst rendering a partial template, `errorPartial()` contains the name of the partial.

Lambdas

The [Mustache manual](#) provides a mechanism to customize rendering of template sections by setting the value for a tag to a callable object (eg. a lambda in Ruby or Javascript), which takes the unrendered block of text for a template section and renders it itself. qt-mustache supports this via the `Context::canEval()` and `Context::eval()` methods.

Chapter 3

QTestRunner

A repo for Increase QTest productivity

Thanks to <https://marcoarena.wordpress.com/2012/06/23/increase-your-qtest-productivity/>!

The QTestLib framework is a tool for unit testing Qt based applications and libraries. I find it precious and simple to use, though it lacks some important features, supported, for example, by GTest. I'm not talking about mocking – for this you generally need an out-and-out framework (like GMock) – instead, I'm referring to simple things like fast deploying. Suppose you have written a test class like this:

This repo contains only the code in this article.

Chapter 4

Namespace Index

4.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Databases		
	Class for Database , contains queries	??
Gui	??
Gui::Widgets		
	Contains Widgets classes	??
Models		
	Models classes	??

Chapter 5

Hierarchical Index

5.1 Class Hierarchy

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

Databases::AccessDatabase	??
Mustache::Context	??
Mustache::QtVariantContext	??
CounterContext	??
CounterContext	??
Databases::Database	??
Databases::BillingDatabase	??
Databases::ContributoryDatabase	??
Databases::CustomerDatabase	??
Databases::ProjectDatabase	??
Databases::RateDatabase	??
Databases::UserDatabase	??
Utils::Directories	??
exception	
Exceptions::DbException	??
Exceptions::FileException	??
Utils::HierarchicalSystem	??
Gui::Widgets::CheckFields::ICheckField	??
Gui::Widgets::CheckFields::CheckQLineEdit	??
Gui::Widgets::CheckFields::CheckEmail	??
Gui::Widgets::CheckFields::CheckPortNumber	??
Gui::Widgets::CheckFields::CheckSiretNumber	??
Gui::Widgets::CheckFields::CheckUntilField	??
Gui::Widgets::CheckFields::CheckFieldsLetters	??
Gui::Widgets::CheckFields::CheckCity	??
Gui::Widgets::CheckFields::CheckCountry	??
Gui::Widgets::CheckFields::CheckName	??
Gui::Widgets::CheckFields::CheckFieldsNumbers	??
Gui::Widgets::CheckFields::CheckIpAddress	??
Gui::Widgets::CheckFields::CheckLogin	??
Gui::Widgets::CheckFields::CheckPhone	??
Gui::Widgets::CheckFields::CheckPostalCode	??
Gui::Widgets::CheckFields::CheckValidField	??
Models::IModel	??
Models::Billing	??
Models::Contributory	??
Models::Customer	??

Models::Project	??
Models::User	??
Utils::ItemType	??
Utils::Log	??
Parameters	??
Mustache::PartialResolver	??
Mustache::PartialFileLoader	??
Mustache::PartialMap	??
Generator::PdfGenerator	??
Models::People	??
Models::Customer	??
Models::User	??
Utils::pointers	??
Gui::Widgets::Popup	??
QAbstractTableModel	
Gui::Widgets::WdgModels::BillingsTableModel	??
Gui::Widgets::WdgModels::ContributoriesTableModel	??
Gui::Widgets::WdgModels::CustomersTableModel	??
Gui::Widgets::WdgModels::ProjectContributoriesTableModel	??
Gui::Widgets::WdgModels::ProjectsTableModel	??
QDialog	
Gui::Dialogs::AddProjectDialog	??
Gui::Dialogs::AddQuoteDialog	??
Gui::Dialogs::ComputeTurnoverDialog	??
Gui::Dialogs::DialogAddCustomer	??
Gui::Dialogs::MessageBox	??
Gui::Dialogs::StartedWindowsDialog	??
Gui::Dialogs::UserDataDialog	??
QDockWidget	
Gui::Docks::SearchDock	??
QItemDelegate	
Gui::Widgets::Delegates::ComboBoxDelegate	??
Gui::Widgets::Delegates::ProjectComboDelegate	??
Gui::Widgets::Delegates::UnitComboDelegate	??
Gui::Widgets::Delegates::DoubleSpinBoxDelegate	??
Gui::Widgets::Delegates::TextareaDelegate	??
QLineEdit	
Gui::Widgets::CheckFields::CheckQLineEdit	??
QMainWindow	
Gui::MainWindow	??
QMap	
Models::ContributoriesList	??
QMenu	
Gui::Widgets::CustomerContextualMenu	??
QObject	
BillingDatabaseTest	??
BillingModelTest	??
ContributoriesDatabaseTest	??
ContributoryListTest	??
ContributoryModelTest	??
CustomerDatabaseTest	??
CustomerModelTest	??
Generation	??
ItemTypeTest	??
PointersTest	??
ProjectDatabaseTest	??
ProjectModelTest	??
RateModelTest	??

searchTest	??
StringTest	??
TestMustache	??
TestMustache	??
UserDatabaseTest	??
UserModelTest	??
QWidget	
Gui::Widgets::ComboBoxModelWidget	??
Gui::Widgets::ContributoriesWidget	??
Gui::Widgets::CustomerDataWidget	??
Gui::Widgets::DatabaseSettingsWidget	??
Gui::Widgets::ProjectsWidget	??
Gui::Widgets::RateWidget	??
Gui::Widgets::searchWidget	??
Models::Rate	??
Mustache::Renderer	??
Models::Search	??
Models::Statistics	??
Utils::String	??
Mustache::Tag	??
TestAdder< T >	??
testadder	??
TestRunner	??
Generator::TexGenerator	??

Chapter 6

Class Index

6.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Databases::AccessDatabase	??
Gui::Dialogs::AddProjectDialog	
Windows to add a new Project	??
Gui::Dialogs::AddQuoteDialog	
Window to add or modify a Quote	??
Models::Billing	
: Billing or Quote of a Customer	??
Databases::BillingDatabase	
The BillingDatabase class Billing (or Quote) table database	??
BillingDatabaseTest	??
BillingModelTest	??
Gui::Widgets::WdgModels::BillingsTableModel	
For a Billing table	??
Gui::Widgets::CheckFields::CheckCity	
Line Edit of City with a check icon	??
Gui::Widgets::CheckFields::CheckCountry	
CheckCountry::CheckCountry Line Edit of country with a check icon	??
Gui::Widgets::CheckFields::CheckEmail	
Line Edit of email with a check icon	??
Gui::Widgets::CheckFields::CheckFieldsLetters	
Field with only letters (no numbers)	??
Gui::Widgets::CheckFields::CheckFieldsNumbers	
Line Edit of number with a check icon	??
Gui::Widgets::CheckFields::CheckIpAddress	
Line Edit of IP address with a check icon	??
Gui::Widgets::CheckFields::CheckLogin	
Line Edit of login with a check icon	??
Gui::Widgets::CheckFields::CheckName	
Line edit of name with a check icon	??
Gui::Widgets::CheckFields::CheckPhone	
Line Edit of Phone number with a check icon	??
Gui::Widgets::CheckFields::CheckPortNumber	
The CheckFieldsNumbers class Line Edit of number with a check icon	??
Gui::Widgets::CheckFields::CheckPostalCode	
Line Edit of postal code with a check icon	??
Gui::Widgets::CheckFields::CheckQLineEdit	
LineEdit custom with a check of text inputed	??

Gui::Widgets::CheckFields::CheckSiretNumber	
Line Edit with a check icon	??
Gui::Widgets::CheckFields::CheckUntilField	
The CheckUntilField class	??
Gui::Widgets::CheckFields::CheckValidField	
Check field not required	??
Gui::Widgets::Delegates::ComboBoxDelegate	
The ComboBoxDelegate class	??
Gui::Widgets::ComboBoxModelWidget	
Model of ComboBox	??
Gui::Dialogs::ComputeTurnoverDialog	
Window to compute a turnover with a period	??
Mustache::Context	??
ContributoriesDatabaseTest	??
Models::ContributoriesList	
List of contributories	??
Gui::Widgets::WdgModels::ContributoriesTableModel	
For a custom table for contributories widget	??
Gui::Widgets::ContributoriesWidget	
Widget of Contributories	??
Models::Contributory	
The Contributory class	??
Databases::ContributoryDatabase	
The ContributoryDatabase class Contributory (or Quote) table database	??
ContributoryListTest	??
ContributoryModelTest	??
CounterContext	??
Models::Customer	
Customer	??
Gui::Widgets::CustomerContextualMenu	
Display contextual menu on a customer	??
Databases::CustomerDatabase	
The CustomerDatabase class Customer table database	??
CustomerDatabaseTest	??
Gui::Widgets::CustomerDataWidget	
Class for display info of a customer	??
CustomerModelTest	??
Gui::Widgets::WdgModels::CustomersTableModel	
For a customer table	??
Databases::Database	
The Database class Master class for all database access	??
Gui::Widgets::DatabaseSettingsWidget	
Windows of database settings	??
Exceptions::DbException	
For database exception : queries, db file,	??
Gui::Dialogs::DialogAddCustomer	
Window to add or modify a Customer	??
Utils::Directories	??
Gui::Widgets::Delegates::DoubleSpinBoxDelegate	
The DoubleSpinBoxDelegate class	??
Exceptions::FileException	
For file/access file exception	??
Generation	??
Utils::HierarchicalSystem	
Create class which contains hierarchical system of FactDev	??
Gui::Widgets::CheckFields::ICheckField	
Interface to check fields validity	??

Models::IModel	
The IModel class	??
Utils::ItemType	
Item type model	??
ItemTypeTest	??
Utils::Log	
For Simple management of log	??
Gui::MainWindow	
Main Window of the software	??
Gui::Dialogs::MessageBox	
Information window with message	??
Parameters	
Class for simple user parameters	??
Mustache::PartialFileLoader	??
Mustache::PartialMap	??
Mustache::PartialResolver	??
Generator::PdfGenerator	
Generator of PDF files	??
Models::People	
People	??
Utils::pointers	??
PointersTest	??
Gui::Widgets::Popup	
Class for display popup quickly	??
Models::Project	
: Project linked to a Customer	??
Gui::Widgets::Delegates::ProjectComboDelegate	
The ProjectComboDelegate class	??
Gui::Widgets::WdgModels::ProjectContributoriesTableModel	
Table model of contributories linked to projets	??
Databases::ProjectDatabase	
Project table database	??
ProjectDatabaseTest	??
ProjectModelTest	??
Gui::Widgets::WdgModels::ProjectsTableModel	
For a Project table	??
Gui::Widgets::ProjectsWidget	
Actions on Project	??
Mustache::QtVariantContext	??
Models::Rate	
Rate of a prestation	??
Databases::RateDatabase	??
RateModelTest	??
Gui::Widgets::RateWidget	
Class for display Rate	??
Mustache::Rendered	??
Models::Search	
The Search class	??
Gui::Docks::SearchDock	
Dock which contains search bar	??
searchTest	??
Gui::Widgets::searchWidget	
Class for search in database	??
Gui::Dialogs::StartedWindowsDialog	
Construct a Windows for the first begin	??
Models::Statistics	??
Utils::String	
The Utils class	??

StringTest	??
Mustache::Tag	??
TestAdder< T >	??
testadder	??
TestMustache	??
TestRunner	??
Generator::TexGenerator	
Generate a LaTeX file	??
Gui::Widgets::Delegates::TextareaDelegate	
The TextareaDelegate class	??
Gui::Widgets::Delegates::UnitComboDelegate	
The UnitComboDelegate class	??
Models::User	
User of it application	??
Databases::UserDatabase	
Access to User data in the the table User of the Database	??
UserDatabaseTest	??
Gui::Dialogs::UserDataDialog	
Window to fill user data	??
UserModelTest	??

Chapter 7

Namespace Documentation

7.1 Databases Namespace Reference

Class for [Database](#), contains queries.

Classes

- class [AccessDatabase](#)
- class [BillingDatabase](#)
The [BillingDatabase](#) class Billing (or Quote) table database.
- class [ContributoryDatabase](#)
The [ContributoryDatabase](#) class Contributory (or Quote) table database.
- class [CustomerDatabase](#)
The [CustomerDatabase](#) class Customer table database.
- class [Database](#)
The [Database](#) class Master class for all database access.
- class [ProjectDatabase](#)
The [ProjectDatabase](#) class Project table database.
- class [RateDatabase](#)
- class [UserDatabase](#)
The [UserDatabase](#) class Access to User data in the the table User of the [Database](#)

Enumerations

- enum **DbType** { [SQLITE](#), [MYSQL](#) }

7.1.1 Detailed Description

Class for [Database](#), contains queries.

7.2 Gui Namespace Reference

Namespaces

- [Widgets](#)
Contains [Widgets](#) classes.

Classes

- class [MainWindow](#)
The [MainWindow](#) class Main Window of the software.

7.2.1 Detailed Description

Classes for Graphical User Interface

7.3 Gui::Widgets Namespace Reference

Contains [Widgets](#) classes.

Classes

- class [ComboBoxModelWidget](#)
The [ComboBoxModelWidget](#) class Model of ComboBox.
- class [ContributoriesWidget](#)
The [ContributoriesWidget](#) class Widget of Contributories.
- class [CustomerContextualMenu](#)
Display contextual menu on a customer.
- class [CustomerDataWidget](#)
Class for display info of a customer.
- class [DatabaseSettingsWidget](#)
The [DatabaseSettingsWidget](#) class Windows of database settings.
- class [Popup](#)
Class for display popup quickly.
- class [ProjectsWidget](#)
The [ProjectsWidget](#) class Actions on Project.
- class [RateWidget](#)
Class for display Rate.
- class [searchWidget](#)
Class for search in database.

7.3.1 Detailed Description

Contains [Widgets](#) classes.

7.4 Models Namespace Reference

[Models](#) classes.

Classes

- class [Billing](#)
The [Billing](#) class : [Billing](#) or Quote of a [Customer](#).
- class [ContributoriesList](#)
The [ContributoriesList](#) class List of contributories.

- class [Contributory](#)
The [Contributory](#) class.
- class [Customer](#)
The [Customer](#) class [Customer](#).
- class [IModel](#)
The [IModel](#) class.
- class [People](#)
The [People](#) class [People](#).
- class [Project](#)
The [Project](#) class : [Project](#) linked to a [Customer](#).
- class [Rate](#)
The [Rate](#) class [Rate](#) of a prestation.
- class [Search](#)
The [Search](#) class.
- class [Statistics](#)
- class [User](#)
The [User](#) class [User](#) of it application.

7.4.1 Detailed Description

[Models](#) classes.

Chapter 8

Class Documentation

8.1 Databases::AccessDatabase Class Reference

Static Public Member Functions

- static void **init** ()
- static void **commit** ()

Static Public Attributes

- static QString **_address**
- static int **_port**
- static QString **_userDb**
- static QString **_password**
- static QString **_dbName**
- static Databases::DbType **_dbType**
- static bool **_exists**

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

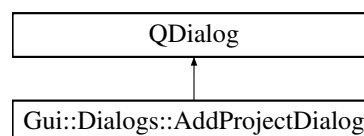
- /home/travis/build/FACT-Team/FactDev/src/database/accessdatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/accessdatabase.cpp

8.2 Gui::Dialogs::AddProjectDialog Class Reference

The [AddProjectDialog](#) class Windows to add a new Project.

```
#include <addprojectdialog.h>
```

Inheritance diagram for Gui::Dialogs::AddProjectDialog:



Public Slots

- void [checkFields](#) ()
AddProjectDialog::checkFields Check if fields are valid.

Public Member Functions

- [AddProjectDialog](#) (int idProject=0, int noRowCustomer=0, QWidget *parent=0)
AddProjectDialogAddProjectDialog Construct a windows according an idCustomer and, optionnaly, an idProject
- void [accept](#) ()
AddProjectDialog::accept Valid data inputed by user and add these data in Database.
- void [reject](#) ()
AddProjectDialog::reject Cancel the operation and close the windows.
- void [fillFields](#) ()
AddProjectDialog::fillFields Fill the differents fields of the current windows according the Project data existing As a project requires to be linked to a Customer, the Customer selection part may be disable.

8.2.1 Detailed Description

The [AddProjectDialog](#) class Windows to add a new Project.

Author

Florent Berbie

See Also

[Project](#)

8.2.2 Constructor & Destructor Documentation

8.2.2.1 [Gui::Dialogs::AddProjectDialog::AddProjectDialog](#) (int *idProject* = 0, int *noRowCustomer* = 0, QWidget * *parent* = 0) [explicit]

[AddProjectDialogAddProjectDialog](#) Construct a windows according an *idCustomer* and, optionnaly, an *idProject*

Parameters

<i>noRowCustomer</i>	Row number of the Customer
<i>idProject</i>	Project identify
<i>parent</i>	QWidget of the current windows

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

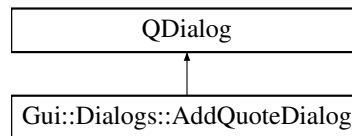
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/addprojectdialog.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/addprojectdialog.cpp

8.3 [Gui::Dialogs::AddQuoteDialog](#) Class Reference

The [AddQuoteDialog](#) class Window to add or modify a Quote.

```
#include <addquotedialog.h>
```

Inheritance diagram for [Gui::Dialogs::AddQuoteDialog](#):



Public Slots

- void **updateBtn** (void)
- void **changeDocType** ()

Public Member Functions

- [AddQuoteDialog](#) (bool isBilling, int idCustomer=0, int id=0, bool copy=false, QWidget *parent=0)
[AddQuoteDialog::AddQuoteDialog](#) Construct a windows [AddQuoteDialog](#).
- void [fillFields](#) ()
[AddQuoteDialog::fillFields](#) Fill line edits with the data of the quote.
- int [getNumber](#) ()
[AddQuoteDialog::getNumber](#) return the number of bill or quote.
- void [accept](#) ()
[AddQuoteDialog::accept](#) Valid data inputed by user and add these data in Database.
- void [reject](#) ()
[AddQuoteDialog::reject](#) Cancel the operation and close the windows.
- bool [getCopy](#) () const
[AddQuoteDialog::getCopy](#) return if [AddQuoteDialog](#) is an add by copy of Billing/quote or if it's edition or add of Billing/quote.
- void [setCopy](#) (bool copy)
[AddQuoteDialog::setCopy](#) Change the `_copy` value to define if it's a copy of a Billing/quote or if it's a new Billing or a Billing edition.
- int [getIdCustomer](#) () const
[AddQuoteDialog::getIdCustomer](#) get the id of the customer changing his quote/billing.
- void [fillQuoteBilling](#) (bool isBilling)
[AddQuoteDialog::fillQuoteBilling](#) fill fields for a Billing or a quote.
- void [fillQuoteBillingCopy](#) (bool isBilling)
[AddQuoteDialog::fillQuoteBilling](#) fill fields if it's a Billing or a quote for a copy.
- void [setQuoteIdNumber](#) (int id, int idCustomer, bool isBilling)
[AddQuoteDialog::setQuoteIdNumber](#) set id and number of the `_quote`

8.3.1 Detailed Description

The [AddQuoteDialog](#) class Window to add or modify a Quote.

Author

8.3.2 Constructor & Destructor Documentation

- 8.3.2.1 [Gui::Dialogs::AddQuoteDialog::AddQuoteDialog](#) (bool *isBilling*, int *idCustomer* = 0, int *id* = 0, bool *copy* = false, QWidget * *parent* = 0) [explicit]

[AddQuoteDialog::AddQuoteDialog](#) Construct a windows [AddQuoteDialog](#).

Parameters

<i>isBilling</i>	displaying of a quote or a Billing
<i>idCustomer</i>	the id of the Customer
<i>id</i>	the id of the quote or the billing of the Customer's Project
<i>edit</i>	if it's an edition or a copy of a quote/Billing
<i>parent</i>	QWidget of the current windows

8.3.3 Member Function Documentation

8.3.3.1 void Gui::Dialogs::AddQuoteDialog::fillQuoteBilling (bool *isBilling*)

[AddQuoteDialog::fillQuoteBilling](#) fill fields for a Billing or a quote.

Parameters

<i>isBilling</i>	if it's a quote or a Billing
------------------	------------------------------

8.3.3.2 bool Gui::Dialogs::AddQuoteDialog::getCopy () const

[AddQuoteDialog::getCopy](#) return if [AddQuoteDialog](#) is an add by copy of Billing/quote or if it's edition or add of Billing/quote.

Returns

if it's a copy or not

8.3.3.3 int Gui::Dialogs::AddQuoteDialog::getIdCustomer () const

[AddQuoteDialog::getIdCustomer](#) get the id of the customer changing his quote/billing.

Returns

the id of the customer

8.3.3.4 int Gui::Dialogs::AddQuoteDialog::getNumber ()

[AddQuoteDialog::getNumber](#) return the number of bill or quote.

Returns

int

8.3.3.5 void Gui::Dialogs::AddQuoteDialog::setCopy (bool *copy*)

[AddQuoteDialog::setCopy](#) Change the `_copy` value to define if it's a copy of a Billing/quote or if it's a new Billing or a Billing edition.

Parameters

<i>copy</i>	
-------------	--

8.3.3.6 void Gui::Dialogs::AddQuoteDialog::setQuoteIdNumber (int *id*, int *idCustomer*, bool *isBilling*)

AddQuoteDialog::setQuoteIdNumber set id and number of the *_quote*

Parameters

<i>id</i>	the id of the Billing
<i>idCustomer</i>	the id of the customer to set the <i>number</i>
<i>isBilling</i>	if it's a quote or a Billing

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

- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/addquotedialog.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/addquotedialog.cpp

8.4 Models::Billing Class Reference

The **Billing** class : **Billing** or Quote of a **Customer**.

```
#include <billing.h>
```

Inheritance diagram for Models::Billing:



Public Member Functions

- **Billing** ()
Billing::Billing. Construct a *Billing*.
- **Billing** (int *id*)
Billing::Billing. Construct a *Billing* or quote.
- **~Billing** ()
destruct a billing object
- void **commit** ()
Billing::commit. Insert a modification in *Billing* table on the database.
- void **hydrat** (int *getId*)
Billing::hydrat. Update of the *Billing* which is specified by *getId*
- void **remove** ()
Billing::remove. Remove a *Billing*.
- QVariantHash **getDataMap** ()
Billing::getDataMap Get all data of model with a HashMap key/value.
- void **generateTex** ()
Billing::generateTex Generate a *.tex* file for the billing.
- void **generatePdf** ()
Billing::generatePdf Generate a *.pdf* file for the billing.
- QString **getPath** ()

- Billing::getPath* Return the path of billing filename (without extension)

 - QString **getFolder** ()

Billing::getFolder Return the directory of billing.
- QString **getFilename** ()

Billing::getFilename Return the filename of billing (without extension)
- **ContributoriesList** & **getContributories** ()

Billing::getContributories. Return a map of **Contributory** for each **Project** of the **Billing**
- void **addContributory** (**Contributory** &c)

Billing::addContributories Add a new contributory for project p.
- double **getSumRate** ()

Billing::getSumRate. return **Billing Rate**.
- double **getSumQuantity** ()

Billing::getSumQuantity. return **Billing** total by time.
- QString **getTitle** () const

Billing::getTitle. return title of **Billing**
- void **setTitle** (const QString &**getTitle**)

Billing::setTitle. Modify the title of **Billing**
- QString **getDescription** () const

Billing::getDescription. return description of **Billing**
- void **setDescription** (const QString &**getDescription**)

Billing::setDescription. Modify the description of **Billing**
- int **getNumber** () const

Billing::getNumber. Return number of the **Billing**.
- void **setNumber** (int **getNumber**)

Billing::setNumber. Modify _number of **Billing**.
- bool **isBilling** () const

Billing::isBilling. Return if it's a billing or a quote.
- void **setIsBilling** (bool **isBilling**)

Billing::setIsBilling. Modify isBilling of **Billing**.
- QDate **getDate** () const

Billing::getDate. return date of the **Billing**
- void **setDate** (const QDate &**getDate**)

Billing::setDate. Modify date of the **Billing**
- bool **isPaid** () const

Billing::isPaid Return TRUE if thee current billing is paid else return FALSE.
- void **setIsPaid** (bool **isPaid**)

Billing::setIsPaid Define the current billing according the argument isPaid
- bool **operator==** (const **Billing** &b)

Billing::operator == define the operator "==" to compare two billings and to see if they are the same.
- bool **operator!=** (const **Billing** &b)

Billing::operator != defines the operator "!=" to compare two **Billing** and to see if they are different.
- void **setContributories** (const **ContributoriesList** &contributories)
- bool **operator<** (const **Billing** &b) const

Billing::operator < defines the operator "<" to compare two **Billing** and to see if the first is anterior to the second.
- QStandardItem * **getItem** ()

Billing::getItem Return the bill/quote item.

Additional Inherited Members

8.4.1 Detailed Description

The [Billing](#) class : [Billing](#) or Quote of a [Customer](#).

Author

Antoine de Roquemaurel
Florent Berbie

8.4.2 Constructor & Destructor Documentation

8.4.2.1 Models::Billing::Billing (int *id*)

[Billing::Billing](#). Construct a [Billing](#) or quote.

Parameters

<i>int</i>	id
------------	----

8.4.3 Member Function Documentation

8.4.3.1 void Models::Billing::addContributory (Contributory & *c*)

[Billing::addContributories](#) Add a new contributory for project p.

Parameters

<i>p</i>	The Project who contain Contributory
<i>c</i>	The new Contributory

8.4.3.2 ContributoriesList & Models::Billing::getContributories ()

[Billing::getContributories](#). Return a map of [Contributory](#) for each [Project](#) of the [Billing](#)

Returns

QMap<[Project](#), QList<Contributory>>

8.4.3.3 QVariantHash Models::Billing::getDataMap () [virtual]

[Billing::getDataMap](#) Get all data of model with a HashMap key/value.

Returns

Model's data

Implements [Models::IModel](#).

8.4.3.4 QDate Models::Billing::getDate () const

[Billing::getDate](#). return date of the [Billing](#)

Returns

date of [Billing](#)

8.4.3.5 QString Models::Billing::getDescription () const

[Billing::getDescription](#). return description of [Billing](#)

Returns

description of [Billing](#)

8.4.3.6 QString Models::Billing::getFilename ()

[Billing::getFilename](#) Return the filename of billing (without extension)

Returns

Filename of Bulling

8.4.3.7 QString Models::Billing::getFolder ()

[Billing::getFolder](#) Return the directory of billing.

Returns

[Billing](#) directory

8.4.3.8 QStandardItem * Models::Billing::getItem ()

[Billing::getItem](#) Return the bill/quote item.

Returns

QStandardItem an item for QTree (level/depth 3)

8.4.3.9 int Models::Billing::getNumber () const

[Billing::getNumber](#). Return number of the [Billing](#).

Returns

_number of [Billing](#)

8.4.3.10 QString Models::Billing::getPath ()

[Billing::getPath](#) Return the path of billing filename (without extension)

Returns

billing path

8.4.3.11 double Models::Billing::getSumQuantity ()

[Billing::getSumQuantity](#). return [Billing](#) total by time.

Returns

time in hours

8.4.3.12 double Models::Billing::getSumRate ()

[Billing::getSumRate](#). return **Billing Rate**.

Returns

8.4.3.13 QString Models::Billing::getTitle () const

[Billing::getTitle](#). return title of **Billing**

Returns

title of **Billing**

8.4.3.14 void Models::Billing::hydrat (int *getId*) [virtual]

[Billing::hydrat](#). Update of the **Billing** which is specified by *getId*

Parameters

<i>getId</i>	
--------------	--

Implements [Models::IModel](#).

8.4.3.15 bool Models::Billing::isBilling () const

[Billing::isBilling](#). Return if it's a billing or a quote.

Returns

if it's billing or a quote

8.4.3.16 bool Models::Billing::isPaid () const

[Billing::isPaid](#) Return TRUE if thee current billing is paid else return FALSE.

Returns

Boolean

8.4.3.17 bool Models::Billing::operator!= (const **Billing** & *b*)

[Billing::operator !=](#) defines the operator "!=" to compare two **Billing** and to see if they are different.

Parameters

<i>b</i>	the Billing to compare with the current Billing
----------	---

Returns

true if the **Billing** are different else false

8.4.3.18 bool Models::Billing::operator< (const **Billing** & *b*) const

[Billing::operator <](#) defines the operator "<" to compare two **Billing** and to see if the first is anterior to the second.

Parameters

<i>b</i>	the Billing to compare with the current Billing
----------	---

Returns

true if the [Billing](#) are different else false

8.4.3.19 `bool Models::Billing::operator== (const Billing & b)`

[Billing::operator ==](#) define the operator "==" to compare two billings and to see if they are the same.

Parameters

<i>b</i>	the Billing to compare with the current Billing
----------	---

Returns

true if they are the same billings else false

8.4.3.20 `void Models::Billing::setDate (const QDate & getDate)`

[Billing::setDate](#). Modify *date* of the [Billing](#)

Parameters

<i>getDate</i>	the new date of the Billing
----------------	---

8.4.3.21 `void Models::Billing::setDescription (const QString & getDescription)`

[Billing::setDescription](#). Modify the description of [Billing](#)

Parameters

<i>getDescription</i>	Modify the description with <i>getDescription</i>
-----------------------	---

8.4.3.22 `void Models::Billing::setIsBilling (bool isBilling)`

[Billing::setIsBilling](#). Modify *isBilling* of [Billing](#).

Parameters

<i>isBilling</i>	
------------------	--

8.4.3.23 `void Models::Billing::setIsPaid (bool isPaid)`

[Billing::setIsPaid](#) Define the current billing according the argument *isPaid*

Parameters

<i>isPaid</i>	Boolean
---------------	---------

8.4.3.24 `void Models::Billing::setNumber (int getNumber)`

[Billing::setNumber](#). Modify *_number* of [Billing](#).

Parameters

<code>getNumber</code>	the new number of the Billing
------------------------	---

8.4.3.25 void Models::Billing::setTitle (const QString & getTitle)

[Billing::setTitle](#). Modify the title of [Billing](#)

Parameters

<code>getTitle</code>	Modify the title with <code>getTitle</code>
-----------------------	---

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

- /home/travis/build/FACT-Team/FactDev/src/models/billing.h
- /home/travis/build/FACT-Team/FactDev/src/models/billing.cpp

8.5 Databases::BillingDatabase Class Reference

The [BillingDatabase](#) class Billing (or Quote) table database.

```
#include <billingdatabase.h>
```

Inheritance diagram for Databases::BillingDatabase:



Public Member Functions

- [Models::Billing](#) * [getBilling](#) (const int pld)
BillingDatabase::getCustomer get informations about the billing identified by pld
- [WdgModels::BillingsTableModel](#) * [getBillingsTable](#) (const int idProject) throw (DbException*)
BillingDatabase::getBillingsTable Return an item model of billings for QTableView.
- int [addBilling](#) (const [Models::Billing](#) &)
BillingDatabase::addBilling Add the billing pBilling to the database.
- void [updateBilling](#) (const [Models::Billing](#) &)
BillingDatabase::updateCustomer Update informations about the billing pCustomer
- void [removeBilling](#) (const int pld)
BillingDatabase::removeCustomer Remove the billing with the id pld
- void [addBillingProject](#) (const int idProject, const int idBilling, const int idContributory)
BillingDatabase::addBillingProject Link a project, a billing and a contributory in the table BillingProject.
- bool [isBillingPaid](#) (const int pld)
BillingDatabase::isBillingPaid Return TRUE if the id pld correspond to a Billing and not quote (isBilling = 1) and if this billing is paid (isPaid = 1) else return FALSE.
- void [removeBillingProject](#) (const int idProject, const int idBilling, const int idContributory)
BillingDatabase::removeBillingProject remove a link between a project, a billing and a contributory in the table BillingProject.
- int [getMaxBillingNumber](#) ()
getMaxBillingNumber Get the last number of a billing

- int [getMaxQuoteNumber](#) ()
getMaxQuoteNumber Get the last number of a quote
- int [getMaxBillingNumberOfCustomer](#) (const int idCustomer)
BillingDatabase::getMaxBillingNumberOfCustomer Get the last number of a **Customer's** Billing.
- int [getMaxQuoteNumberOfCustomer](#) (const int idCustomer)
BillingDatabase::getMaxQuoteNumberOfCustomer Get the last number of a **Customer's** quote.
- QSharedPointer< [Models::Billing](#) > [getBilling](#) (QStringQuery &q)
BillingDatabase::getBilling Add the element of the q request and return their.
- QMap< [Project](#) *, [Billing](#) * > [getAllBillingsOfProject](#) ()
BillingDatabase::getAllBillingsOfProject Return a map with the project id as key linked to the billing.
- QList< [Billing](#) > [getBillings](#) (const int projectId)
BillingDatabase::getBillings get bills by project.
- QList< [Billing](#) > [getAllBillingsOnly](#) (const int idProject)
BillingDatabase::getAllBillingsOnly get all billings which aren't quotes for the project p
- QList< [Billing](#) > [getBillingsBetweenDates](#) (QDate begin, QDate end)
BillingDatabase::getBillingsBetweenDates get billings in the list between begin and end dates.

Static Public Member Functions

- static [BillingDatabase](#) * [instance](#) () throw (DbException*)
BillingDatabase::getInstance Return an instance of [BillingDatabase](#)

Additional Inherited Members

8.5.1 Detailed Description

The [BillingDatabase](#) class Billing (or Quote) table database.

Author

See Also

[Database](#)
Billing/Quote

8.5.2 Member Function Documentation

8.5.2.1 int Databases::BillingDatabase::addBilling (const Models::Billing & pBilling)

[BillingDatabase::addBilling](#) Add the billing *pBilling* to the database.

Returns

billing id

8.5.2.2 void Databases::BillingDatabase::addBillingProject (const int idProject, const int idBilling, const int idContributory)

[BillingDatabase::addBillingProject](#) Link a project, a billing and a contributory in the table BillingProject.

Parameters

<i>idProject</i>	Project id
<i>idBilling</i>	Billing id
<i>idContributory</i>	Contributory id

8.5.2.3 QMap< Project *, Billing * > Databases::BillingDatabase::getAllBillingsOfProject ()

[BillingDatabase::getAllBillingsOfProject](#) Return a map with the project id as key linked to the billing.

Returns

Map with projects and Billing

8.5.2.4 QList< Billing > Databases::BillingDatabase::getAllBillingsOnly (const int *idProject*)

[BillingDatabase::getAllBillingsOnly](#) get all billings which aren't quotes for the project *p*

Parameters

<i>p</i>	
----------	--

Returns

the list of **Billing**

8.5.2.5 Models::Billing * Databases::BillingDatabase::getBilling (const int *pld*)

[BillingDatabase::getCustomer](#) get informations about the billing identified by *pld*

Parameters

<i>pld</i>	billing id
------------	------------

Returns

the Billing

8.5.2.6 QSharedPointer< Billing > Databases::BillingDatabase::getBilling (QSqlQuery & *q*)

[BillingDatabase::getBilling](#) Add the element of the *q* request and return their.

Parameters

<i>q</i>	SQL request
----------	-------------

Returns

a billing formed according to QSharedPointer

8.5.2.7 QList< Billing > Databases::BillingDatabase::getBillings (const int *projectId*)

[BillingDatabase::getBillings](#) get bills by project.

Parameters

<i>projectId</i>	
------------------	--

Returns

List with bills

8.5.2.8 `QList< Billing > Databases::BillingDatabase::getBillingsBetweenDates (QDate begin, QDate end)`

[BillingDatabase::getBillingsBetweenDates](#) get billings in the list between *begin* and *end* dates.

Parameters

<i>begin</i>	
<i>end</i>	

Returns

the list of **Billing**

8.5.2.9 `WdgModels::BillingsTableModel * Databases::BillingDatabase::getBillingsTable (const int idProject) throw DbException *`

[BillingDatabase::getBillingsTable](#) Return an item model of billings for QTableView.

Parameters

<i>pid</i>	the project id of the billings returned
------------	---

Exceptions

<i>DbException</i>	
--------------------	--

Returns

QStandardItemModel an item model for QTableView

8.5.2.10 `int Databases::BillingDatabase::getMaxBillingNumber ()`

`getMaxBillingNumber` Get the last number of a billing

Returns

The max number

8.5.2.11 `int Databases::BillingDatabase::getMaxBillingNumberOfCustomer (const int idCustomer)`

[BillingDatabase::getMaxBillingNumberOfCustomer](#) Get the last number of a **Customer's** Billing.

Parameters

--

<i>idCustomer</i>	Customer id
-------------------	-------------

Returns

the last number of the **Customer's** Billing

8.5.2.12 `int Databases::BillingDatabase::getMaxQuoteNumber ()`

`getMaxQuoteNumber` Get the last number of a quote

Returns

The last number

8.5.2.13 `int Databases::BillingDatabase::getMaxQuoteNumberOfCustomer (const int idCustomer)`

[BillingDatabase::getMaxQuoteNumberOfCustomer](#) Get the last number of a **Customer's** quote.

Parameters

<i>idCustomer</i>	Customer id
-------------------	-------------

Returns

the last number of the **Customer's** quote

8.5.2.14 `BillingDatabase * Databases::BillingDatabase::instance () throw DbException *` [static]

`BillingDatabase::getInstance` Return an instance of [BillingDatabase](#)

See Also

[DbException](#)

Returns

Instance of [BillingDatabase](#)

8.5.2.15 `bool Databases::BillingDatabase::isBillingPaid (const int pld)`

[BillingDatabase::isBillingPaid](#) Return TRUE if the id *pld* correspond to a Billing and not quote (isBilling = 1) and if this billing is paid (isPaid = 1) else return FALSE.

Parameters

<i>pld</i>	Billing id
------------	------------

Returns

TRUE if billing is paid

8.5.2.16 `void Databases::BillingDatabase::removeBilling (const int pld)`

`BillingDatabase::removeCustomer` Remove the billing with the id *pld*

Parameters

<i>pId</i>	billing id
------------	------------

8.5.2.17 void Databases::BillingDatabase::removeBillingProject (const int *idProject*, const int *idBilling*, const int *idContributory*)

[BillingDatabase::removeBillingProject](#) remove a link between a project, a billing and a contributory in the table BillingProject.

Parameters

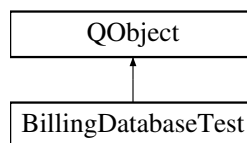
<i>idProject</i>	Project id
<i>idBilling</i>	Billing id
<i>idContributory</i>	Contributory id

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

- /home/travis/build/FACT-Team/FactDev/src/database/billingdatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/billingdatabase.cpp

8.6 BillingDatabaseTest Class Reference

Inheritance diagram for BillingDatabaseTest:

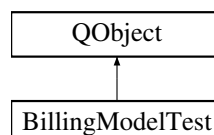


The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/tests/database/billingdatabasetest.h
- /home/travis/build/FACT-Team/FactDev/tests/database/billingdatabasetest.cpp

8.7 BillingModelTest Class Reference

Inheritance diagram for BillingModelTest:



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

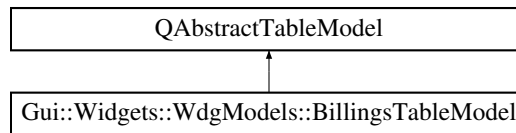
- /home/travis/build/FACT-Team/FactDev/tests/models/billingmodeltest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/billingmodeltest.cpp

8.8 Gui::Widgets::WdgModels::BillingsTableModel Class Reference

The [BillingsTableModel](#) class for a Billing table.

```
#include <billingstablemodel.h>
```

Inheritance diagram for Gui::Widgets::WdgModels::BillingsTableModel:



Public Member Functions

- [BillingsTableModel](#) ()
BillingsTableModel::BillingsTableModel Construct a *BillingsTableModel*.
- int [rowCount](#) (const QModelIndex &) const
BillingsTableModel::rowCount Number of billings row.
- int [columnCount](#) (const QModelIndex &) const
BillingsTableModel::columnCount Number of column of a Billing.
- QVariant [data](#) (const QModelIndex &index, int role=Qt::DisplayRole) const
BillingsTableModel::data Obtains data of a specify cell.
- QVariant [headerData](#) (int section, Qt::Orientation orientation, int role=Qt::DisplayRole) const
BillingsTableModel::headerData Obtains header title of table.
- bool [setData](#) (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)
BillingsTableModel::setData Change data of a cell.
- void [append](#) (const [Billing](#) &billing)
BillingsTableModel::append Add a new line in table.
- void [remove](#) (const int i)
BillingsTableModel::remove Remove a line.
- Qt::ItemFlags [flags](#) (const QModelIndex &index) const
BillingsTableModel::flags Differents table flags.
- int [count](#) ()
BillingsTableModel::count Number of billings in table.
- QList< [Billing](#) > [getBillings](#) () const
BillingsTableModel::getbillings Return the list of billings.

8.8.1 Detailed Description

The [BillingsTableModel](#) class for a Billing table.

Author

Florent Berbie

See Also

[Billing](#)

8.8.2 Constructor & Destructor Documentation

8.8.2.1 Gui::Widgets::WdgModels::BillingsTableModel::BillingsTableModel ()

[BillingsTableModel::BillingsTableModel](#) Construct a [BillingsTableModel](#).

Parameters

<i>parent</i>	Parent widget
---------------	---------------

8.8.3 Member Function Documentation

8.8.3.1 void Gui::Widgets::WdgModels::BillingsTableModel::append (const Billing & *billing*)

[BillingsTableModel::append](#) Add a new line in table.

Parameters

<i>Billing</i>	The new Billing
----------------	-----------------

8.8.3.2 int Gui::Widgets::WdgModels::BillingsTableModel::columnCount (const QModelIndex &) const

[BillingsTableModel::columnCount](#) Number of column of a Billing.

Returns

The number of column

8.8.3.3 int Gui::Widgets::WdgModels::BillingsTableModel::count ()

[BillingsTableModel::count](#) Number of billings in table.

Returns

The number of billings

8.8.3.4 QVariant Gui::Widgets::WdgModels::BillingsTableModel::data (const QModelIndex & *index*, int *role* = Qt::DisplayRole) const

[BillingsTableModel::data](#) Obtains data of a specify cell.

Parameters

<i>index</i>	The cell who we want data
<i>role</i>	The role of set

Returns

The data of cell

8.8.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::BillingsTableModel::flags (const QModelIndex & *index*) const

[BillingsTableModel::flags](#) Differents table flags.

Parameters

<i>index</i>	The cell who we want to know flags
--------------	------------------------------------

Returns

Flags

8.8.3.6 `QList< Billing > Gui::Widgets::WdgModels::BillingsTableModel::getBillings () const`

`BillingsTableModel::getbillings` Return the list of billings.

Returns

list of billings

8.8.3.7 `QVariant Gui::Widgets::WdgModels::BillingsTableModel::headerData (int section, Qt::Orientation orientation, int role = Qt::DisplayRole) const`

`BillingsTableModel::headerData` Obtains header title of table.

Parameters

<i>section</i>	The number of column
<i>orientation</i>	The table orientation
<i>role</i>	

Returns

The Title header of column

8.8.3.8 `void Gui::Widgets::WdgModels::BillingsTableModel::remove (const int i)`

`BillingsTableModel::remove` Remove a line.

Parameters

<i>i</i>	The number of line to remove
----------	------------------------------

8.8.3.9 `int Gui::Widgets::WdgModels::BillingsTableModel::rowCount (const QModelIndex &) const`

`BillingsTableModel::rowCount` Number of billings row.

Returns

The number of billings

8.8.3.10 `bool Gui::Widgets::WdgModels::BillingsTableModel::setData (const QModelIndex & index, const QVariant & value, int role = Qt::EditRole)`

`BillingsTableModel::setData` Change data of a cell.

Parameters

<i>index</i>	The cell to change data
<i>value</i>	The new value
<i>role</i>	The role of cell

Returns

True if we could edit

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

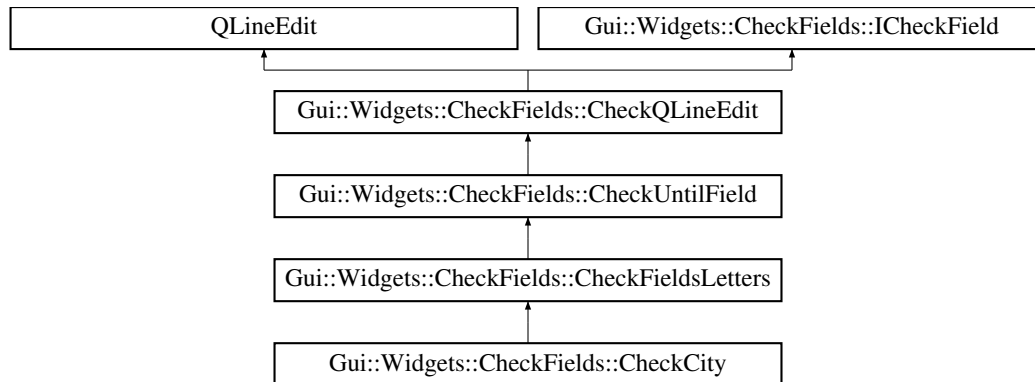
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetmodels/billingstablemodel.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetmodels/billingstablemodel.cpp

8.9 Gui::Widgets::CheckFields::CheckCity Class Reference

The [CheckCity](#) class Line Edit of City with a check icon.

```
#include <checkcity.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckCity:



Public Member Functions

- [CheckCity](#) (QWidget *w=0, QPushButton *btn=0)
[CheckCity::CheckCity](#) Construct a [CheckCity](#).

Additional Inherited Members

8.9.1 Detailed Description

The [CheckCity](#) class Line Edit of City with a check icon.

8.9.2 Constructor & Destructor Documentation

8.9.2.1 Gui::Widgets::CheckFields::CheckCity::CheckCity (QWidget * w = 0, QPushButton * btn = 0)

[CheckCity::CheckCity](#) Construct a [CheckCity](#).

Parameters

<i>w</i>	QWidget linked to CheckCity
----------	---

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

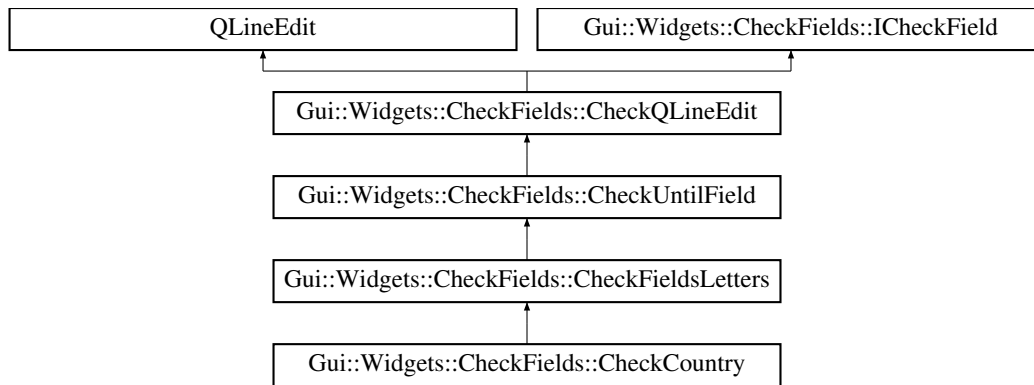
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkcity.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkcity.cpp

8.10 Gui::Widgets::CheckFields::CheckCountry Class Reference

[CheckCountry::CheckCountry](#) Line Edit of country with a check icon.

```
#include <checkcountry.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckCountry:



Public Member Functions

- [CheckCountry](#) (QWidget *w=0, QPushButton *btn=0)
CheckCountry::CheckCountry Construct a [CheckCountry](#).

Additional Inherited Members

8.10.1 Detailed Description

[CheckCountry::CheckCountry](#) Line Edit of country with a check icon.

8.10.2 Constructor & Destructor Documentation

8.10.2.1 [Gui::Widgets::CheckFields::CheckCountry::CheckCountry](#) (QWidget * *w* = 0, QPushButton * *btn* = 0)

[CheckCountry::CheckCountry](#) Construct a [CheckCountry](#).

Parameters

<i>w</i>	QWidget linked to CheckCountry
----------	--

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

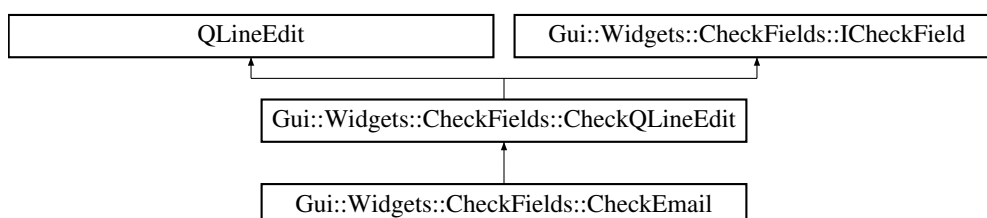
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkcountry.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkcountry.cpp

8.11 Gui::Widgets::CheckFields::CheckEmail Class Reference

The [CheckEmail](#) class Line Edit of email with a check icon.

```
#include <checkemail.h>
```

Inheritance diagram for [Gui::Widgets::CheckFields::CheckEmail](#):



Public Member Functions

- [CheckEmail](#) (QWidget *w=0, QPushButton *btn=0)
[CheckEmail::CheckEmail](#) Construct a CheckMail.
- bool [check](#) (const QString text)
*[CheckEmail::check](#) Check if the field email is valid. To be valid, an email address should be under this form: **me@me.** – **xx** An email address need:*

Additional Inherited Members

8.11.1 Detailed Description

The [CheckEmail](#) class Line Edit of email with a check icon.

8.11.2 Constructor & Destructor Documentation

8.11.2.1 [Gui::Widgets::CheckFields::CheckEmail::CheckEmail](#) ([QWidget](#) * *w* = 0, [QPushButton](#) * *btn* = 0)

[CheckEmail::CheckEmail](#) Construct a CheckMail.

Parameters

<i>w</i>	QWidget linked to CheckEmail
----------	--

8.11.3 Member Function Documentation

8.11.3.1 bool [Gui::Widgets::CheckFields::CheckEmail::check](#) (const [QString](#) *text*) `[virtual]`

[CheckEmail::check](#) Check if the field email is valid. To be valid, an email address should be under this form: **me@me.** **xx** An email address need:

- 1 character [A-Z] or [a-z] minimum before the character `</i>`
- the character '@'
- 1 character [A-Z] or [a-z] after the character`</i>` minimum and before the character .
- 1 character [A-Z] or [a-z] minimum afer the character . Return TRUE if email address is valid, else FALSE

Parameters

<i>text</i>	<i>Text to check</i>
-------------	----------------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkemail.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkemail.cpp

8.12 Gui::Widgets::CheckFields::CheckFieldsLetters Class Reference

The [CheckFieldsLetters](#) class Field with only letters (no numbers)

```
#include <checkfieldsletters.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckFieldsLetters:



Public Member Functions

- [CheckFieldsLetters](#) (QWidget *w=0, QPushButton *btn=0)
[CheckFieldsLetters::CheckFieldsLetters](#) Construct a [CheckFieldsLetters](#).
- bool [check](#) (QString text)
[CheckFieldsLetters::check](#) Check if the field contains only letters.

Additional Inherited Members

8.12.1 Detailed Description

The [CheckFieldsLetters](#) class Field with only letters (no numbers)

8.12.2 Constructor & Destructor Documentation

8.12.2.1 Gui::Widgets::CheckFields::CheckFieldsLetters::CheckFieldsLetters (QWidget * w = 0, QPushButton * btn = 0)

[CheckFieldsLetters::CheckFieldsLetters](#) Construct a [CheckFieldsLetters](#).

Parameters

<i>w</i>	QWidget linked to CheckFieldsLetters
----------	--

8.12.3 Member Function Documentation

8.12.3.1 bool Gui::Widgets::CheckFields::CheckFieldsLetters::check (QString *text*) [virtual]

[CheckFieldsLetters::check](#) Check if the field contains only letters.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

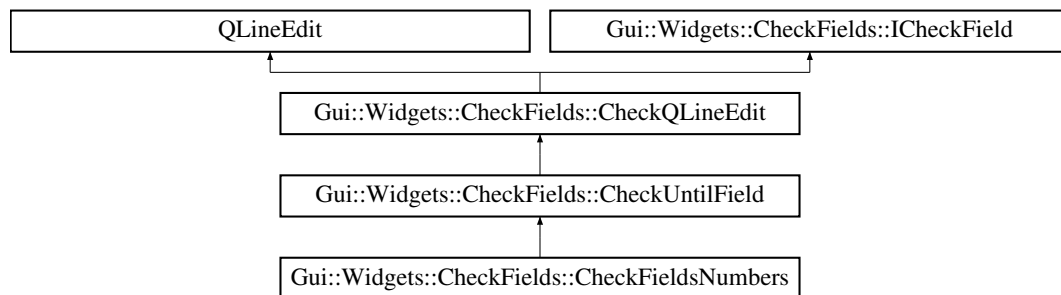
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkfieldsletters.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkfieldsletters.cpp

8.13 Gui::Widgets::CheckFields::CheckFieldsNumbers Class Reference

The [CheckFieldsNumbers](#) class Line Edit of number with a check icon.

```
#include <checkfieldsnumbers.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckFieldsNumbers:

**Public Member Functions**

- [CheckFieldsNumbers](#) (QWidget *w=0, QPushButton *btn=0)
CheckFieldsNumbers::CheckFieldsNumbers Construct a [CheckFieldsNumbers](#).
- bool [check](#) (QString text)
CheckFieldsNumbers::check Check if the field contains only numbers.

Additional Inherited Members

8.13.1 Detailed Description

The [CheckFieldsNumbers](#) class Line Edit of number with a check icon.

Author

Florent BERBIE

See Also

[CheckQLineEdit](#)
[CheckUntilField](#)

8.13.2 Constructor & Destructor Documentation

8.13.2.1 [Gui::Widgets::CheckFields::CheckFieldsNumbers::CheckFieldsNumbers](#) ([QWidget](#) * *w* = 0, [QPushButton](#) * *btn* = 0)

[CheckFieldsNumbers::CheckFieldsNumbers](#) Construct a [CheckFieldsNumbers](#).

Parameters

<i>w</i>	QWidget linked to CheckFieldsNumbers
----------	--

8.13.3 Member Function Documentation

8.13.3.1 `bool Gui::Widgets::CheckFields::CheckFieldsNumbers::check (QString text) [virtual]`

[CheckFieldsNumbers::check](#) Check if the field contains only numbers.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

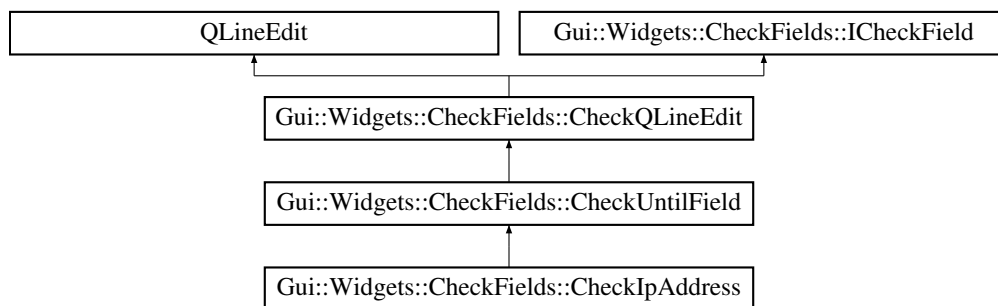
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkfieldsnumbers.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkfieldsnumbers.cpp

8.14 Gui::Widgets::CheckFields::CheckIpAddress Class Reference

The [CheckIpAddress](#) class Line Edit of IP address with a check icon.

```
#include <checkipaddress.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckIpAddress:



Public Member Functions

- [CheckIpAddress](#) (QWidget *w=0, QPushButton *btn=0)
[CheckIpAddress::CheckIpAddress](#) Construct a [CheckIpAddress](#).
- bool [check](#) (QString text)
[CheckIpAddress::check](#) Check if the field contains an IP address or domain name valid.

Additional Inherited Members

8.14.1 Detailed Description

The [CheckIpAddress](#) class Line Edit of IP address with a check icon.

Author

Florent BERBIE

See Also

[CheckQLineEdit](#)
[CheckUntilField](#)

8.14.2 Constructor & Destructor Documentation

8.14.2.1 `Gui::Widgets::CheckFields::CheckIpAddress::CheckIpAddress (QWidget * w = 0, QPushButton * btn = 0)`[CheckIpAddress::CheckIpAddress](#) Construct a [CheckIpAddress](#).

Parameters

<i>w</i>	QWidget linked to CheckIpAddress
----------	--

8.14.3 Member Function Documentation

8.14.3.1 `bool Gui::Widgets::CheckFields::CheckIpAddress::check (QString text) [virtual]`[CheckIpAddress::check](#) Check if the field contains an IP address or domain name valid.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

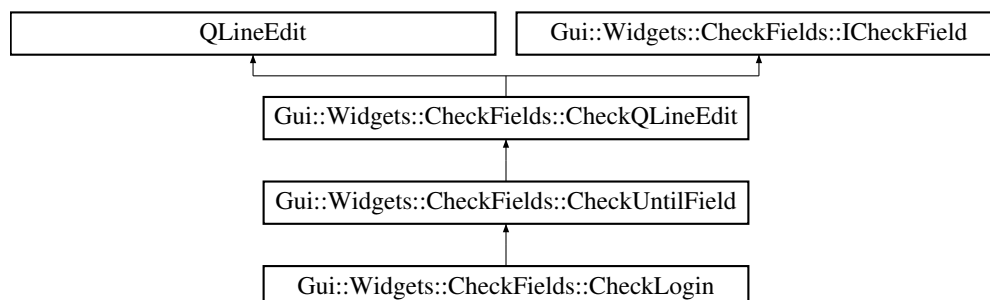
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkipaddress.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkipaddress.cpp

8.15 Gui::Widgets::CheckFields::CheckLogin Class Reference

The [CheckLogin](#) class Line Edit of login with a check icon.

#include <checklogin.h>

Inheritance diagram for Gui::Widgets::CheckFields::CheckLogin:



Public Slots

- void [passwordPreviousInputed](#) (const QString &text)
[CheckLogin::fieldTextChanged](#) For each new characater inputed or removed, displays an icon to show if the field is valid or not.

Public Member Functions

- [CheckLogin](#) (QWidget *w=0, QPushButton *btn=0)
[CheckLogin::CheckLogin](#) Construct a [CheckLogin](#).
- bool [check](#) (QString text)
[CheckLogin::check](#) Check if the field contains only numbers.

8.15.1 Detailed Description

The [CheckLogin](#) class Line Edit of login with a check icon.

Author

Florent BERBIE

See Also

[CheckQLineEdit](#)
[CheckUntilField](#)

8.15.2 Constructor & Destructor Documentation

8.15.2.1 [Gui::Widgets::CheckFields::CheckLogin::CheckLogin](#) ([QWidget](#) * *w* = 0, [QPushButton](#) * *btn* = 0)

[CheckLogin::CheckLogin](#) Construct a [CheckLogin](#).

Parameters

<i>w</i>	QWidget linked to CheckIpAddress
----------	--

8.15.3 Member Function Documentation

8.15.3.1 bool [Gui::Widgets::CheckFields::CheckLogin::check](#) ([QString](#) *text*) [virtual]

[CheckLogin::check](#) Check if the field contains only numbers.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

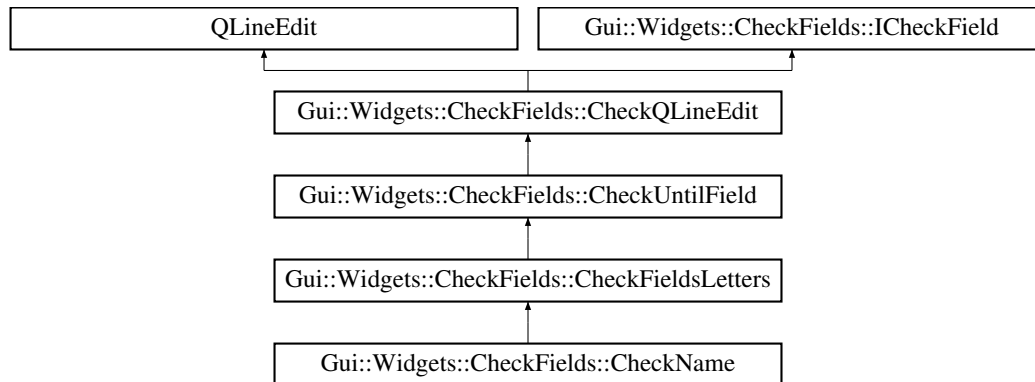
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checklogin.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checklogin.cpp

8.16 Gui::Widgets::CheckFields::CheckName Class Reference

The [CheckName](#) class Line edit of name with a check icon.

```
#include <checkname.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckName:



Public Member Functions

- [CheckName](#) (QWidget *w=0, QPushButton *btn=0)
[CheckName::CheckName](#) Construct a [CheckName](#).

Additional Inherited Members

8.16.1 Detailed Description

The [CheckName](#) class Line edit of name with a check icon.

8.16.2 Constructor & Destructor Documentation

8.16.2.1 Gui::Widgets::CheckFields::CheckName::CheckName (QWidget * w = 0, QPushButton * btn = 0)

[CheckName::CheckName](#) Construct a [CheckName](#).

Parameters

<i>w</i>	QWidget linked to CheckName
----------	---

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

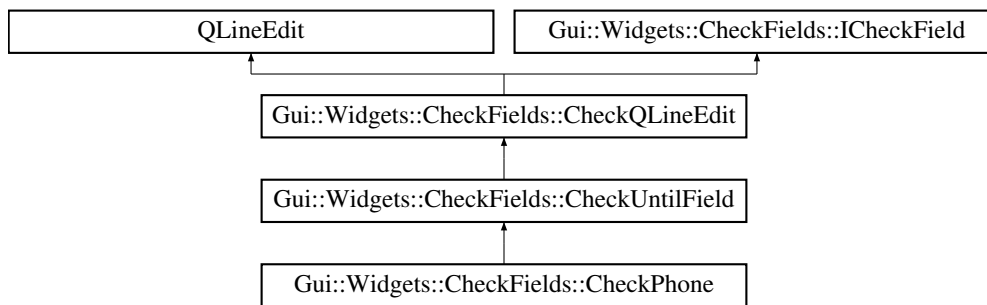
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkname.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkname.cpp

8.17 Gui::Widgets::CheckFields::CheckPhone Class Reference

The [CheckPhone](#) class Line Edit of Phone number with a check icon.

```
#include <checkphone.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckPhone:



Public Member Functions

- [CheckPhone](#) (QWidget *w=0, QPushButton *btn=0)
[CheckPhone::CheckPhone](#) Construct a [CheckPhone](#).
- bool [check](#) (QString text)
[CheckPhone::check](#) Check if the field is valid. To be valid, a name should be composed of a character.
- QString [getCountry](#) () const
[CheckPhone::getCountry](#) Return the country linked to current field.
- void [setCountry](#) (const QString &country)
[CheckPhone::setCountry](#) Modify the country linked to field.

Additional Inherited Members

8.17.1 Detailed Description

The [CheckPhone](#) class Line Edit of Phone number with a check icon.

8.17.2 Constructor & Destructor Documentation

8.17.2.1 [Gui::Widgets::CheckFields::CheckPhone::CheckPhone](#) ([QWidget](#) * *w* = 0, [QPushButton](#) * *btn* = 0)

[CheckPhone::CheckPhone](#) Construct a [CheckPhone](#).

Parameters

<i>w</i>	QWidget linked to CheckPhone
----------	--

8.17.3 Member Function Documentation

8.17.3.1 bool [Gui::Widgets::CheckFields::CheckPhone::check](#) ([QString](#) *text*) [virtual]

[CheckPhone::check](#) Check if the field is valid. To be valid, a name should be composed of a character.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

8.17.3.2 QString Gui::Widgets::CheckFields::CheckPhone::getCountry () const

[CheckPhone::getCountry](#) Return the country linked to current field.

Returns

8.17.3.3 void Gui::Widgets::CheckFields::CheckPhone::setCountry (const QString & country)

[CheckPhone::setCountry](#) Modify the *country* linked to field.

Parameters

<i>country</i>	New country
----------------	-------------

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

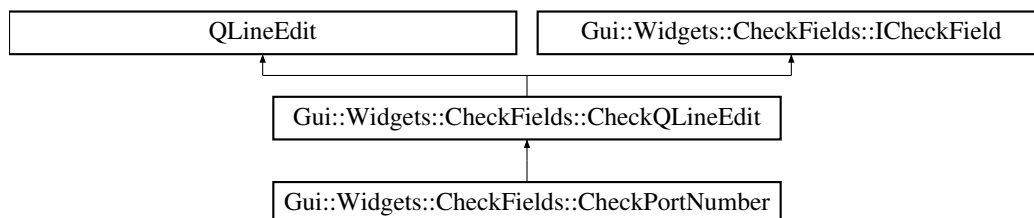
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkphone.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkphone.cpp

8.18 Gui::Widgets::CheckFields::CheckPortNumber Class Reference

The [CheckFieldsNumbers](#) class Line Edit of number with a check icon.

```
#include <checkportnumber.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckPortNumber:



Public Member Functions

- [CheckPortNumber](#) (QWidget *w=0, QPushButton *btn=0)
CheckPortNumber.
- bool [check](#) (QString text)
CheckPortNumber::check Check if the field contains only numbers or an empty text.

Additional Inherited Members

8.18.1 Detailed Description

The [CheckFieldsNumbers](#) class Line Edit of number with a check icon.

Author

Florent BERBIE

See Also

[CheckQLineEdit](#)
[CheckUntilField](#)

8.18.2 Constructor & Destructor Documentation

8.18.2.1 `Gui::Widgets::CheckFields::CheckPortNumber::CheckPortNumber (QWidget * w = 0, QPushButton * btn = 0)`

[CheckPortNumber.](#)

Parameters

<i>w</i>	Widget parent
<i>btn</i>	Button parretrn

8.18.3 Member Function Documentation

8.18.3.1 `bool Gui::Widgets::CheckFields::CheckPortNumber::check (QString text)` `[virtual]`

[CheckPortNumber::check](#) Check if the field contains only numbers or an empty text.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

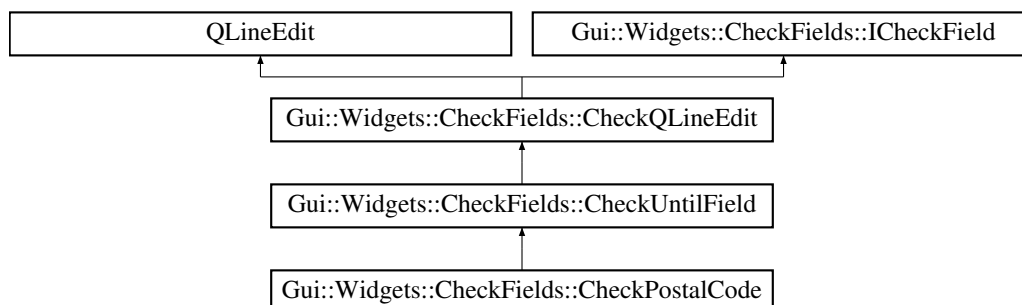
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkportnumber.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkportnumber.cpp

8.19 Gui::Widgets::CheckFields::CheckPostalCode Class Reference

The [CheckPostalCode](#) class Line Edit of postal code with a check icon.

```
#include <checkpostalcode.h>
```

Inheritance diagram for `Gui::Widgets::CheckFields::CheckPostalCode`:



Public Member Functions

- [CheckPostalCode](#) (QWidget *w=0, QPushButton *btn=0)
[CheckPostalCode::CheckPostalCode](#) Construct a [CheckPostalCode](#).
- bool [check](#) (QString text)
[CheckPostalCode::check](#) Check if the field is valid. To be valid, a name should be composed of a character.
- QString [getCountry](#) () const
[CheckPostalCode::getCountry](#) Return the country linked to current field.
- void [setCountry](#) (const QString &country)
[CheckPostalCode::setCountry](#) Modify the country linked to field.

Additional Inherited Members

8.19.1 Detailed Description

The [CheckPostalCode](#) class Line Edit of postal code with a check icon.

8.19.2 Constructor & Destructor Documentation

8.19.2.1 [Gui::Widgets::CheckFields::CheckPostalCode::CheckPostalCode](#) ([QWidget](#) * *w* = 0, [QPushButton](#) * *btn* = 0)

[CheckPostalCode::CheckPostalCode](#) Construct a [CheckPostalCode](#).

Parameters

<i>w</i>	QWidget linked to CheckPostalCode
----------	---

8.19.3 Member Function Documentation

8.19.3.1 bool [Gui::Widgets::CheckFields::CheckPostalCode::check](#) ([QString](#) *text*) [\[virtual\]](#)

[CheckPostalCode::check](#) Check if the field is valid. To be valid, a name should be composed of a character.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

8.19.3.2 [QString](#) [Gui::Widgets::CheckFields::CheckPostalCode::getCountry](#) () const

[CheckPostalCode::getCountry](#) Return the country linked to current field.

Returns

country Country of the field

8.19.3.3 void [Gui::Widgets::CheckFields::CheckPostalCode::setCountry](#) (const [QString](#) & *country*)

[CheckPostalCode::setCountry](#) Modify the *country* linked to field.

Parameters

<i>country</i>	New country
----------------	-------------

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

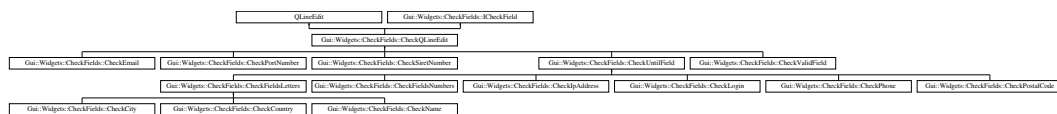
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkpostalcode.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkpostalcode.cpp

8.20 Gui::Widgets::CheckFields::CheckQLineEdit Class Reference

The [CheckQLineEdit](#) class QLineEdit custom with a check of text inputed.

```
#include <checkqlineEdit.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckQLineEdit:



Public Slots

- void [fieldTextChanged](#) (const QString &text)
[CheckQLineEdit::fieldTextChanged](#) For each new characater inputed or removed, displays an icon to show if the field is valid or not.

Public Member Functions

- [CheckQLineEdit](#) (QWidget *parent=0, QPushButton *btn=0)
[CheckQLineEdit::CheckQLineEdit](#) Construct a [CheckQLineEdit](#).
- void [displayCheckValidFieldIcon](#) ()
[CheckQLineEdit::displayCheckValidFieldIcon](#) Display a valid icon into the field.
- void [displayCheckNoValidFieldIcon](#) ()
[CheckQLineEdit::displayCheckNoValidFieldIcon](#) Display a "no valid" icon into the field.
- QPushButton * [getBtnValid](#) () const
[CheckQLineEdit::getBtnValid](#).
- void [setBtnValid](#) (QPushButton *getBtnValid)
[CheckQLineEdit::setBtnValid](#).
- bool [isValid](#) ()
[isValid](#) Return true if the current field if valid

8.20.1 Detailed Description

The [CheckQLineEdit](#) class QLineEdit custom with a check of text inputed.

8.20.2 Constructor & Destructor Documentation

- 8.20.2.1 [Gui::Widgets::CheckFields::CheckQLineEdit::CheckQLineEdit](#) (QWidget * *parent* = 0, QPushButton * *btn* = 0)
[explicit]

[CheckQLineEdit::CheckQLineEdit](#) Construct a [CheckQLineEdit](#).

Parameters

<i>parent</i>	
---------------	--

8.20.3 Member Function Documentation

8.20.3.1 `QPushButton * Gui::Widgets::CheckFields::CheckQLineEdit::getBtnValid () const`

[CheckQLineEdit::getBtnValid.](#)

Returns

a

8.20.3.2 `bool Gui::Widgets::CheckFields::CheckQLineEdit::isValid ()`

isValid Return true if the current field if valid

Returns

boolean

8.20.3.3 `void Gui::Widgets::CheckFields::CheckQLineEdit::setBtnValid (QPushButton * getBtnValid)`

[CheckQLineEdit::setBtnValid.](#)

Parameters

<i>getBtnValid</i>	
--------------------	--

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

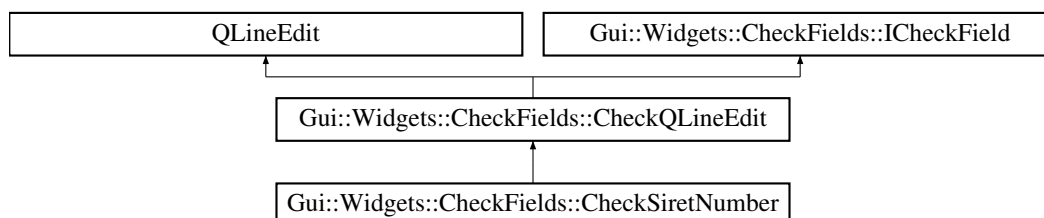
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkqlineEdit.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkqlineEdit.cpp

8.21 Gui::Widgets::CheckFields::CheckSiretNumber Class Reference

The [CheckSiretNumber](#) class Line Edit with a check icon.

```
#include <checksiretnumber.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::CheckSiretNumber:



Public Member Functions

- [CheckSiretNumber](#) (QWidget *w=0, QPushButton *btn=0)

[CheckSiretNumber::CheckSiretNumber](#) Construct a [CheckSiretNumber](#).

- bool [check](#) (QString text)

[CheckSiretNumber::check](#) Check if the field noSiret is valid. To be valid, a SIRET number should be composed of numbers.

Additional Inherited Members

8.21.1 Detailed Description

The [CheckSiretNumber](#) class Line Edit with a check icon.

8.21.2 Constructor & Destructor Documentation

8.21.2.1 `Gui::Widgets::CheckFields::CheckSiretNumber::CheckSiretNumber (QWidget * w = 0, QPushButton * btn = 0)`

[CheckSiretNumber::CheckSiretNumber](#) Construct a [CheckSiretNumber](#).

Parameters

<i>w</i>	QWidget linked to CheckSiretNumber
----------	--

8.21.3 Member Function Documentation

8.21.3.1 `bool Gui::Widgets::CheckFields::CheckSiretNumber::check (QString text) [virtual]`

[CheckSiretNumber::check](#) Check if the field noSiret is valid. To be valid, a SIRET number should be composed of numbers.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

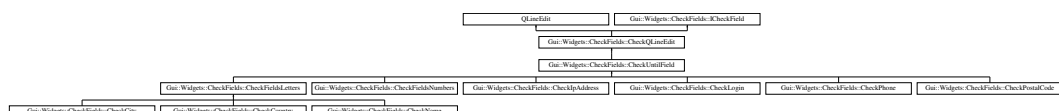
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checksiretnumber.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checksiretnumber.cpp

8.22 Gui::Widgets::CheckFields::CheckUntilField Class Reference

The [CheckUntilField](#) class.

```
#include <checkuntilfield.h>
```

Inheritance diagram for [Gui::Widgets::CheckFields::CheckUntilField](#):



Public Member Functions

- [CheckUntilField](#) (QWidget *w=0, QPushButton *btn=0)
[CheckUntilField::CheckUntilField](#) Construct a [CheckUntilField](#).
- bool [check](#) (QString text)
[CheckUntilField::check](#) Check if the field is valid. To be valid, a name should be composed of a character.

Additional Inherited Members

8.22.1 Detailed Description

The [CheckUntilField](#) class.

8.22.2 Constructor & Destructor Documentation

8.22.2.1 [Gui::Widgets::CheckFields::CheckUntilField::CheckUntilField](#) ([QWidget](#) * *w* = 0, [QPushButton](#) * *btn* = 0)

[CheckUntilField::CheckUntilField](#) Construct a [CheckUntilField](#).

Parameters

<i>w</i>	QWidget linked to CheckUntilField
----------	---

8.22.3 Member Function Documentation

8.22.3.1 bool [Gui::Widgets::CheckFields::CheckUntilField::check](#) ([QString](#) *text*) [\[virtual\]](#)

[CheckUntilField::check](#) Check if the field is valid. To be valid, a name should be composed of a character.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

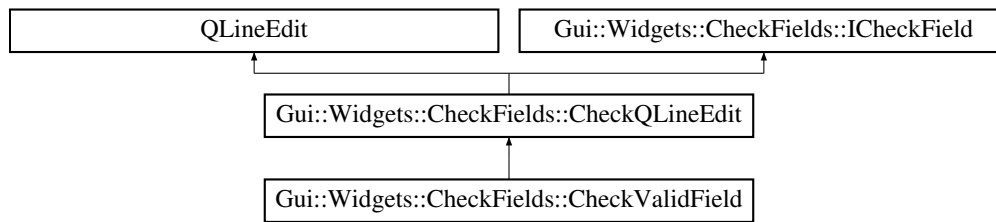
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkuntilfield.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkuntilfield.cpp

8.23 Gui::Widgets::CheckFields::CheckValidField Class Reference

The [CheckValidField](#) class Check field not required.

```
#include <checkvalidfield.h>
```

Inheritance diagram for [Gui::Widgets::CheckFields::CheckValidField](#):



Public Member Functions

- [CheckValidField](#) (QWidget *w=0, QPushButton *btn=0)
[CheckValidField::CheckValidField](#).
- bool [check](#) (QString text)
[CheckValidField::check](#) Return TRUE : the field is not required.

Additional Inherited Members

8.23.1 Detailed Description

The [CheckValidField](#) class Check field not required.

8.23.2 Constructor & Destructor Documentation

8.23.2.1 [Gui::Widgets::CheckFields::CheckValidField::CheckValidField](#) (QWidget * *w* = 0, QPushButton * *btn* = 0)

[CheckValidField::CheckValidField](#).

Parameters

<i>w</i>	QWidget linked to CheckValidField
----------	---

8.23.3 Member Function Documentation

8.23.3.1 bool [Gui::Widgets::CheckFields::CheckValidField::check](#) (QString *text*) [virtual]

[CheckValidField::check](#) Return TRUE : the field is not required.

Parameters

<i>text</i>	Text to check
-------------	---------------

Returns

boolean Validity of the text

Implements [Gui::Widgets::CheckFields::ICheckField](#).

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

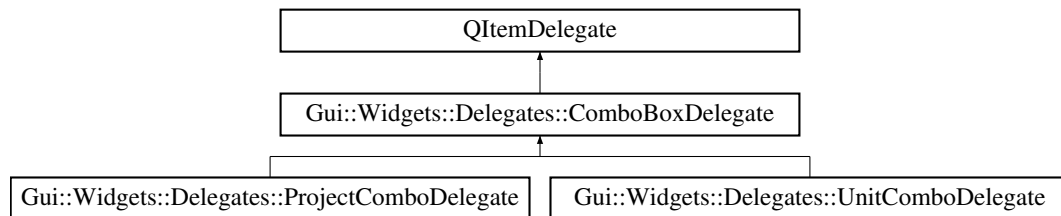
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkvalidfield.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/checkvalidfield.cpp

8.24 Gui::Widgets::Delegates::ComboBoxDelegate Class Reference

The [ComboBoxDelegate](#) class.

```
#include <comboboxdelegate.h>
```

Inheritance diagram for Gui::Widgets::Delegates::ComboBoxDelegate:



Public Member Functions

- [ComboBoxDelegate](#) (QObject *parent=0)
ComboBoxDelegate::ComboBoxDelegate Construct a [ComboBoxDelegate](#).
- virtual QWidget * [createEditor](#) (QWidget *parent, const QStyleOptionViewItem &option, const QModelIndex &index) const =0
ComboBoxDelegate::createEditor Return a *ComboBox* specified by *index* item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void [paint](#) (QPainter *painter, const QStyleOptionViewItem &option, const QModelIndex &index) const =0
ComboBoxDelegate::paint Renders the delegate using the given painter and style option for the item specified by *index*
- void [setEditorData](#) (QWidget *editor, const QModelIndex &index) const
ComboBoxDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index
- void [setModelData](#) (QWidget *editor, QAbstractItemModel *model, const QModelIndex &index) const
ComboBoxDelegate::setModelData Gets data from the editor widget and stores it in the specified model at the item index
- void [updateEditorGeometry](#) (QWidget *editor, const QStyleOptionViewItem &option, const QModelIndex &index) const
ComboBoxDelegate::updateEditorGeometry Update the editor for the item specified by index according to the style option given.

8.24.1 Detailed Description

The [ComboBoxDelegate](#) class.

Author

Antoine de Roquemaurel

8.24.2 Constructor & Destructor Documentation

8.24.2.1 Gui::Widgets::Delegates::ComboBoxDelegate::ComboBoxDelegate (QObject * parent = 0)

[ComboBoxDelegate::ComboBoxDelegate](#) Construct a [ComboBoxDelegate](#).

Parameters

<i>parent</i>	Object parent
---------------	---------------

8.24.3 Member Function Documentation

8.24.3.1 `virtual QWidget* Gui::Widgets::Delegates::ComboBoxDelegate::createEditor (QWidget * parent, const QStyleOptionViewItem & option, const QModelIndex & index) const` [pure virtual]

[ComboBoxDelegate::createEditor](#) Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Returns

ComboBox

Implemented in [Gui::Widgets::Delegates::UnitComboDelegate](#), and [Gui::Widgets::Delegates::ProjectComboDelegate](#).

8.24.3.2 `void Gui::Widgets::Delegates::ComboBoxDelegate::paint (QPainter * painter, const QStyleOptionViewItem & option, const QModelIndex & index) const` [pure virtual]

[ComboBoxDelegate::paint](#) Renders the delegate using the given *painter* and style *option* for the item specified by *index*

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Implemented in [Gui::Widgets::Delegates::UnitComboDelegate](#), and [Gui::Widgets::Delegates::ProjectComboDelegate](#).

8.24.3.3 `void Gui::Widgets::Delegates::ComboBoxDelegate::setEditorData (QWidget * editor, const QModelIndex & index) const`

[ComboBoxDelegate::setEditorData](#) Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

<i>editor</i>	Data edited
<i>index</i>	Index of the model to edit

8.24.3.4 `void Gui::Widgets::Delegates::ComboBoxDelegate::setModelData (QWidget * editor, QAbstractItemModel * model, const QModelIndex & index) const`

[ComboBoxDelegate::setModelData](#) Gets data from the *editor* widget and stores it in the specified *model* at the item *index*

Parameters

<i>editor</i>	Editor Widget
<i>model</i>	Model to store data
<i>index</i>	Item index

8.24.3.5 void Gui::Widgets::Delegates::ComboBoxDelegate::updateEditorGeometry (QWidget * *editor*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const

[ComboBoxDelegate::updateEditorGeometry](#) Update the *editor* for the item specified by *index* according to the style *option* given.

Parameters

<i>editor</i>	Editor widget to update
<i>option</i>	Style option
<i>index</i>	Item index

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

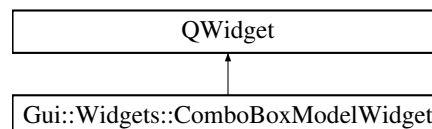
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/comboboxdelegate.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/comboboxdelegate.cpp

8.25 Gui::Widgets::ComboBoxModelWidget Class Reference

The [ComboBoxModelWidget](#) class Model of ComboBox.

```
#include <comboboxmodelwidget.h>
```

Inheritance diagram for Gui::Widgets::ComboBoxModelWidget:



Public Member Functions

- [ComboBoxModelWidget](#) (QWidget *parent=0)
[ComboBoxModelWidget::ComboBoxModelWidget](#) Construct a [ComboBoxModelWidget](#).

8.25.1 Detailed Description

The [ComboBoxModelWidget](#) class Model of ComboBox.

8.25.2 Constructor & Destructor Documentation

8.25.2.1 Gui::Widgets::ComboBoxModelWidget::ComboBoxModelWidget (QWidget * *parent* = 0) [explicit]

[ComboBoxModelWidget::ComboBoxModelWidget](#) Construct a [ComboBoxModelWidget](#).

Parameters

<i>parent</i>	QWidget parent
---------------	----------------

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

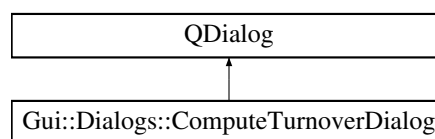
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/comboboxmodelwidget.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/comboboxmodelwidget.cpp

8.26 Gui::Dialogs::ComputeTurnoverDialog Class Reference

The [ComputeTurnoverDialog](#) class window to compute a turnover with a period.

```
#include <computeturnoverdialog.h>
```

Inheritance diagram for Gui::Dialogs::ComputeTurnoverDialog:



Public Slots

- void [computeTurnover](#) ()
ComputeTurnoverDialog::computeTurnover compute the turnover between chosen dates in the window.
- void [endDateControl](#) (const QDateTime end)
ComputeTurnoverDialog::endDateControl controls if the end date field is valid.
- void [beginDateControl](#) (const QDateTime begin)
ComputeTurnoverDialog::beginDateControl controls if the begin date field is valid.

Public Member Functions

- **ComputeTurnoverDialog** (QWidget *parent=0)
- void [fillLabels](#) (const int nbBillings, const int turnover)
ComputeTurnoverDialog::fillLabels Fills the labels with nbBillings and turnover

8.26.1 Detailed Description

The [ComputeTurnoverDialog](#) class window to compute a turnover with a period.

Author

Manantsoa Razanajatovo

8.26.2 Member Function Documentation

8.26.2.1 void [Gui::Dialogs::ComputeTurnoverDialog::beginDateControl](#) (const QDateTime *begin*) [slot]

[ComputeTurnoverDialog::beginDateControl](#) controls if the *begin* date field is valid.

Parameters

<i>begin</i>	
--------------	--

8.26.2.2 void Gui::Dialogs::ComputeTurnoverDialog::endDateControl (const QDate *end*) [slot]

[ComputeTurnoverDialog::endDateControl](#) controls if the *end* date field is valid.

Parameters

<i>end</i>	
------------	--

8.26.2.3 void Gui::Dialogs::ComputeTurnoverDialog::fillLabels (const int *nbBillings*, const int *turnover*)

[ComputeTurnoverDialog::fillLabels](#) Fills the labels with *nbBillings* and *turnover*

Parameters

<i>nbBillings</i>	the number of Billings
<i>turnover</i>	the turnover computed

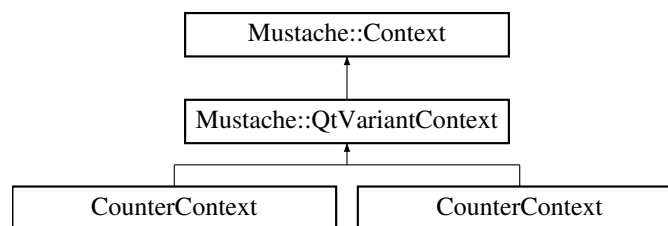
The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/computeturnoverdialog.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/computeturnoverdialog.cpp

8.27 Mustache::Context Class Reference

```
#include <mustache.h>
```

Inheritance diagram for Mustache::Context:



Public Member Functions

- [Context](#) ([PartialResolver](#) *resolver=0)
- virtual QString [stringValue](#) (const QString &key) const =0
- virtual bool [isFalse](#) (const QString &key) const =0
- virtual int [listCount](#) (const QString &key) const =0
- virtual void [push](#) (const QString &key, int index=-1)=0
- virtual void [pop](#) ()=0
- QString [partialValue](#) (const QString &key) const
- [PartialResolver](#) * [partialResolver](#) () const
- virtual bool [canEval](#) (const QString &key) const
- virtual QString [eval](#) (const QString &key, const QString &_template, [Renderer](#) *renderer)

8.27.1 Detailed Description

[Context](#) is an interface that [Mustache::Renderer::render\(\)](#) uses to fetch substitutions for template tags.

8.27.2 Constructor & Destructor Documentation

8.27.2.1 Context::Context ([PartialResolver](#) * *resolver* = 0) [explicit]

Create a context. *resolver* is used to fetch the expansions for any `{{>partial}}` tags which appear in a template.

8.27.3 Member Function Documentation

8.27.3.1 bool Context::canEval (const [QString](#) & *key*) const [virtual]

Returns true if [eval\(\)](#) should be used to render section tags using *key*. If [canEval\(\)](#) returns true for a key, the renderer will pass the literal, unrendered block of text for the section to [eval\(\)](#) and replace the section with the result.

[canEval\(\)](#) and [eval\(\)](#) are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented in [CounterContext](#), [CounterContext](#), and [Mustache::QtVariantContext](#).

8.27.3.2 [QString](#) Context::eval (const [QString](#) & *key*, const [QString](#) & *_template*, [Renderer](#) * *renderer*) [virtual]

Callback used to render a template section with the given *key*. *renderer* will substitute the original section tag with the result of [eval\(\)](#).

The default implementation returns an empty string.

Reimplemented in [CounterContext](#), [CounterContext](#), and [Mustache::QtVariantContext](#).

8.27.3.3 virtual bool Mustache::Context::isFalse (const [QString](#) & *key*) const [pure virtual]

Returns true if the value for *key* is 'false' or an empty list. 'False' values typically include empty strings, the boolean value false etc.

When processing a section Mustache tag, the section is not rendered if the key is false, or for an inverted section tag, the section is only rendered if the key is false.

Implemented in [Mustache::QtVariantContext](#).

8.27.3.4 virtual int Mustache::Context::listCount (const [QString](#) & *key*) const [pure virtual]

Returns the number of items in the list value for *key* or 0 if the value for *key* is not a list.

Implemented in [Mustache::QtVariantContext](#).

8.27.3.5 [PartialResolver](#) * Context::partialResolver () const

Returns the partial resolver passed to the constructor.

8.27.3.6 [QString](#) Context::partialValue (const [QString](#) & *key*) const

Returns the partial template for a given *key*.

8.27.3.7 `virtual void Mustache::Context::pop () [pure virtual]`

Exit the current context.

Implemented in [Mustache::QtVariantContext](#).

8.27.3.8 `virtual void Mustache::Context::push (const QString & key, int index = -1) [pure virtual]`

Set the current context to the value for `key`. If `index` is ≥ 0 , set the current context to the `index`'th value in the list value for `key`.

Implemented in [Mustache::QtVariantContext](#).

8.27.3.9 `virtual QString Mustache::Context::stringValue (const QString & key) const [pure virtual]`

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

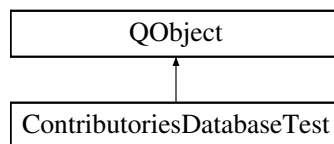
Implemented in [CounterContext](#), [CounterContext](#), and [Mustache::QtVariantContext](#).

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

- `/home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h`
- `/home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.cpp`

8.28 ContributoriesDatabaseTest Class Reference

Inheritance diagram for ContributoriesDatabaseTest:



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

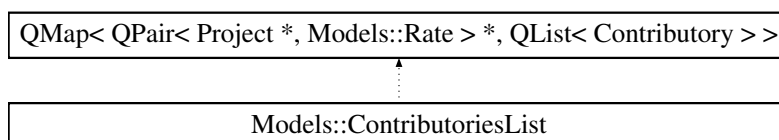
- `/home/travis/build/FACT-Team/FactDev/tests/database/contributoriesdatabasetest.h`
- `/home/travis/build/FACT-Team/FactDev/tests/database/contributoriesdatabasetest.cpp`

8.29 Models::ContributoriesList Class Reference

The [ContributoriesList](#) class List of contributories.

```
#include <contributorieslist.h>
```

Inheritance diagram for Models::ContributoriesList:



Public Member Functions

- [ContributoriesList](#) ()
ContributoriesList::ContributoriesList Construct a [ContributoriesList](#).
- virtual void [commit](#) ()
ContributoriesList::commit Update or insert data into the database.
- void [addContributory](#) ([Models::Contributory](#) &contributory)
ContributoriesList::addContributory Add a new contributory
- void [addProject](#) ([Project](#) *p, [Models::Rate](#) rate)
ContributoriesList::addProject Add a [Project](#) p and it rate
- [QList< Contributory >](#) & [getContributories](#) ([Project](#) *p)
ContributoriesList::getContributories Return a list of Contributories for the [Project](#) p
- int [getIdBilling](#) () const
ContributoriesList::getIdBilling Return the [Billing](#) ID.
- void [setIdBilling](#) (int idBilling)
ContributoriesList::setIdBilling Change the [Billing](#) id by the new idBilling
- void [setAllIdContributories](#) (int idContributory)
ContributoriesList::setAllIdContributories Change all [Contributory](#) id with the same id.
- bool [isInsert](#) () const
ContributoriesList::isInsert Return TRUE if an element is inserting else FALSE.
- void [setInsert](#) (bool insert)
ContributoriesList::setInsert Change the state of insertion.
- int [getNbProjects](#) ()
ContributoriesList::getNbProjects Return the number of projects.
- double [getSumRate](#) ()
ContributoriesList::getSumRate Return the sum of the contributories rates.
- double [getSumQuantity](#) ()
ContributoriesList::getSumQuantity Return the sum of quantity (number of hours) of the Contributories.
- [QSharedPointer< Customer >](#) [getCustomer](#) ()
ContributoriesList::getCustomer Return the Customers linked to theses contributories.
- [QList< Project * >](#) [getProjects](#) ()
ContributoriesList::getProjects List of Projects.
- [QList< Contributory >](#) * [getAllContributories](#) ()
ContributoriesList::getAllContributories List of all contributories (all contributories from all projects)
- [Models::Rate](#) [getRate](#) ([Models::Project](#) *project)
ContributoriesList::getRate.
- [QVariantList](#) [getDataMap](#) ()
ContributoriesList::getDataMap Return a list of [Billing](#) and it value linked which indicates if it is inserting or not.

8.29.1 Detailed Description

The [ContributoriesList](#) class List of contributories.

Author

Antoine de Roquemaurel

8.29.2 Member Function Documentation

8.29.2.1 void [Models::ContributoriesList::addContributory](#) ([Models::Contributory](#) & contributory)

[ContributoriesList::addContributory](#) Add a new contributory

Parameters

<i>contributory</i>	Contributory to add
---------------------	-------------------------------------

8.29.2.2 void Models::ContributorList::addProject ([Project](#) * *p*, Models::Rate *rate*)

[ContributorList::addProject](#) Add a [Project](#) *p* and it *rate*

Parameters

<i>p</i>	Project to add
<i>rate</i>	Rate of the project

8.29.2.3 QList< [Contributory](#) > * Models::ContributorList::getAllContributories ()

[ContributorList::getAllContributories](#) List of all contributories (all contributories from all projects)

Returns

List of all contributories

8.29.2.4 QList< [Contributory](#) > & Models::ContributorList::getContributories ([Project](#) * *p*)

[ContributorList::getContributories](#) Return a list of ContributorList for the [Project](#) *p*

Parameters

<i>p</i>	Project
----------	-------------------------

Returns

List of ContributorList for a project

8.29.2.5 QSharedPointer< [Customer](#) > Models::ContributorList::getCustomer ()

[ContributorList::getCustomer](#) Return the Customers linked to theses contributories.

Returns

[Customer](#)

8.29.2.6 QVariantList Models::ContributorList::getDataMap ()

[ContributorList::getDataMap](#) Return a list of [Billing](#) and it value linked which indicates if it is inserting or not.

Returns

List of billing and value linked

8.29.2.7 int Models::ContributorList::getIdBilling () const

[ContributorList::getIdBilling](#) Return the [Billing](#) ID.

Returns

[Billing](#) id

8.29.2.8 `int Models::ContributoriesList::getNbProjects ()`

[ContributoriesList::getNbProjects](#) Return the number of projects.

Returns

Count number of project

8.29.2.9 `QList< Project * > Models::ContributoriesList::getProjects (void)`

[ContributoriesList::getProjects](#) List of Projects.

Returns

List of Projects

8.29.2.10 `Models::Rate Models::ContributoriesList::getRate (Models::Project * project)`

[ContributoriesList::getRate](#).

Parameters

<i>project</i>	
----------------	--

Returns**8.29.2.11** `double Models::ContributoriesList::getSumQuantity ()`

[ContributoriesList::getSumQuantity](#) Return the sum of quantity (number of hours) of the Contributories.

Returns

sum of quantity

8.29.2.12 `double Models::ContributoriesList::getSumRate ()`

[ContributoriesList::getSumRate](#) Return the sum of the contributories rates.

Returns

Sum of contributories rates

8.29.2.13 `bool Models::ContributoriesList::isInsert () const`

[ContributoriesList::isInsert](#) Return TRUE if an element is inserting else FALSE.

Returns

boolean

8.29.2.14 `void Models::ContributoriesList::setAllIdContributories (int idContributory)`

[ContributoriesList::setAllIdContributories](#) Change all [Contributory](#) id with the same id.

Parameters

<i>idContributory</i>	the new Contributory id
-----------------------	---

8.29.2.15 void Models::ContributoriesList::setIdBilling (int *idBilling*)

[ContributoriesList::setIdBilling](#) Change the [Billing](#) id by the new *idBilling*

Parameters

<i>idBilling</i>	Billing id
------------------	------------

8.29.2.16 void Models::ContributoriesList::setInsert (bool *insert*)

[ContributoriesList::setInsert](#) Change the state of insertion.

Parameters

<i>insert</i>	Boolean
---------------	---------

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

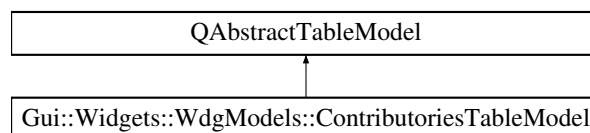
- /home/travis/build/FACT-Team/FactDev/src/models/contributorieslist.h
- /home/travis/build/FACT-Team/FactDev/src/models/contributorieslist.cpp

8.30 Gui::Widgets::WdgModels::ContributoriesTableModel Class Reference

The [ContributoriesTableModel](#) class for a custom table for contributories widget.

```
#include <contributoriestablemodel.h>
```

Inheritance diagram for Gui::Widgets::WdgModels::ContributoriesTableModel:



Public Member Functions

- [ContributoriesTableModel](#) (QObject *parent=0)
[ContributoriesTableModel::ContributoriesTableModel](#) Construct a [ContributoriesTableModel](#).
- int [rowCount](#) (const QModelIndex &) const
[ContributoriesTableModel::rowCount](#) Number of contributories row.
- int [columnCount](#) (const QModelIndex &) const
[ContributoriesTableModel::columnCount](#) Number of column of a contributory.
- QVariant [data](#) (const QModelIndex &index, int role) const
[ContributoriesTableModel::data](#) Obtains data of a specify cell.
- QVariant [headerData](#) (int section, Qt::Orientation orientation, int role) const
[ContributoriesTableModel::headerData](#) Obtains header title of table.
- bool [setData](#) (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)
[ContributoriesTableModel::setData](#) Change data of a cell.
- void [append](#) (const [Contributory](#) &contributory)

- [*ContributoriesTableModel::append*](#) Add a new line in table.
- void [*remove*](#) (const int i)
- [*ContributoriesTableModel::remove*](#) Remove a line.
- Qt::ItemFlags [*flags*](#) (const QModelIndex &index) const
- [*ContributoriesTableModel::flags*](#) Differents table flags.
- QList< [*Contributory*](#) > [*getContributories*](#) ()
- [*ContributoriesTableModel::getContributories*](#) Get all contributories of table.
- int [*count*](#) ()
- [*ContributoriesTableModel::count*](#) Number of contributories in table.
- double [*getSumQuantity*](#) () const
- [*ContributoriesTableModel::getSumQuantity*](#) Return the sum of the hours of all contributories added.
- void [*clear*](#) ()
- [*ContributoriesTableModel::clear*](#) Remove all contributories.

8.30.1 Detailed Description

The [*ContributoriesTableModel*](#) class for a custom table for contributories widget.

Author

Antoine de Roquemaurel

See Also

[*Contributory*](#)

8.30.2 Constructor & Destructor Documentation

8.30.2.1 [*Gui::Widgets::WdgModels::ContributoriesTableModel::ContributoriesTableModel \(QObject * parent = 0 \)*](#)

[*ContributoriesTableModel::ContributoriesTableModel*](#) Construct a [*ContributoriesTableModel*](#).

Parameters

<i>parent</i>	Parent widget
---------------	---------------

8.30.3 Member Function Documentation

8.30.3.1 void [*Gui::Widgets::WdgModels::ContributoriesTableModel::append \(const Contributory & contributory \)*](#)

[*ContributoriesTableModel::append*](#) Add a new line in table.

Parameters

<i>contributory</i>	The new contributory
---------------------	----------------------

8.30.3.2 int [*Gui::Widgets::WdgModels::ContributoriesTableModel::columnCount \(const QModelIndex & \)*](#) const

[*ContributoriesTableModel::columnCount*](#) Number of column of a contributory.

Returns

The number of column

8.30.3.3 `int Gui::Widgets::WdgModels::ContributoriesTableModel::count ()`

[ContributoriesTableModel::count](#) Number of contributories in table.

Returns

The number of contributories

8.30.3.4 `QVariant Gui::Widgets::WdgModels::ContributoriesTableModel::data (const QModelIndex & index, int role) const`

[ContributoriesTableModel::data](#) Obtains data of a specify cell.

Parameters

<i>index</i>	The cell who we want data
<i>role</i>	The role of set

Returns

The data of cell

8.30.3.5 `Qt::ItemFlags Gui::Widgets::WdgModels::ContributoriesTableModel::flags (const QModelIndex & index) const`

[ContributoriesTableModel::flags](#) Differents table flags.

Parameters

<i>index</i>	The cell who we want to know flags
--------------	------------------------------------

Returns

Flags

8.30.3.6 `QList< Contributory > Gui::Widgets::WdgModels::ContributoriesTableModel::getContributories ()`

[ContributoriesTableModel::getContributories](#) Get all contributories of table.

Returns

The contributory list

8.30.3.7 `double Gui::Widgets::WdgModels::ContributoriesTableModel::getSumQuantity () const`

[ContributoriesTableModel::getSumQuantity](#) Return the sum of the hours of all contriburoies added.

Returns

sum quantity

8.30.3.8 `QVariant Gui::Widgets::WdgModels::ContributoriesTableModel::headerData (int section, Qt::Orientation orientation, int role) const`

[ContributoriesTableModel::headerData](#) Obtains header title of table.

Parameters

<i>section</i>	The number of column
<i>orientation</i>	The table orientation
<i>role</i>	

Returns

The Title header of column

8.30.3.9 `void Gui::Widgets::WdgModels::ContributoriesTableModel::remove (const int i)`

[ContributoriesTableModel::remove](#) Remove a line.

Parameters

<i>i</i>	The number of line to remove
----------	------------------------------

8.30.3.10 `int Gui::Widgets::WdgModels::ContributoriesTableModel::rowCount (const QModelIndex &) const`

[ContributoriesTableModel::rowCount](#) Number of contributories row.

Returns

The number of contributories

8.30.3.11 `bool Gui::Widgets::WdgModels::ContributoriesTableModel::setData (const QModelIndex & index, const QVariant & value, int role = Qt::EditRole)`

[ContributoriesTableModel::setData](#) Change data of a cell.

Parameters

<i>index</i>	The cell to change data
<i>value</i>	The new value
<i>role</i>	The role of cell

Returns

True if we could edit

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

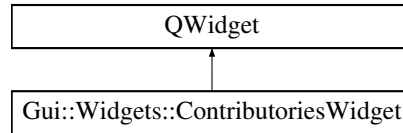
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetsmodels/contributoriesstablemodel.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetsmodels/contributoriesstablemodel.cpp

8.31 Gui::Widgets::ContributoriesWidget Class Reference

The [ContributoriesWidget](#) class Widget of Contributories.

```
#include <contributorieswidget.h>
```

Inheritance diagram for Gui::Widgets::ContributoriesWidget:



Public Slots

- void [add](#) (void)
[ContributoriesWidget::add](#) Add a new empty contributory.
- void [remove](#) (void)
[ContributoriesWidget::remove](#) Remove the current contributory.
- void [addProject](#) (QPair< [Project](#) *, [Rate](#) > *p=0)
[ContributoriesWidget::addProject](#) Add a Projet and it rate p
- void [removeProject](#) (void)
[ContributoriesWidget::removeProject](#) Remove the current Project.
- void [changeProject](#) (void)
[ContributoriesWidget::changeProject](#) Change the current Project.
- void [editing](#) (void)
[ContributoriesWidget::editing](#) Remove the current Project in the combobox not used.
- void [updateUi](#) (void)
[ContributoriesWidget::updateUi](#) Update the User Interface.
- void [updatePrice](#) (void)
[ContributoriesWidget::updatePrice](#) Update total price.

Signals

- void [contributoryChanged](#) ()
[ContributoriesWidget::contributoryChanged](#) Signal that a contributory has changed.

Public Member Functions

- [ContributoriesWidget](#) (QSharedPointer< [Customer](#) > c, QWidget *parent=0)
[ContributoriesWidget::ContributoriesWidget](#) Construct a [ContributoriesWidget](#).
- [ContributoriesList](#) * [getContributories](#) () const
[ContributoriesWidget::getContributories](#) Get contributories List.
- int [count](#) ()
[ContributoriesWidget::count](#) Numbers of contributories.
- void [add](#) ([ContributoriesList](#) &list)
[ContributoriesWidget::add](#) Add contributorieslist list in the model.

8.31.1 Detailed Description

The [ContributoriesWidget](#) class Widget of Contributories.

8.31.2 Constructor & Destructor Documentation

- 8.31.2.1 [Gui::Widgets::ContributoriesWidget::ContributoriesWidget](#) (QSharedPointer< [Customer](#) > c, QWidget * parent = 0) [explicit]

[ContributoriesWidget::ContributoriesWidget](#) Construct a [ContributoriesWidget](#).

Parameters

<i>c</i>	Customer
<i>parent</i>	Widget parent

8.31.3 Member Function Documentation

8.31.3.1 void Gui::Widgets::ContributoriesWidget::add (ContributoriesList & *list*)

[ContributoriesWidget::add](#) Add contributorieslist *list* in the model.

Parameters

<i>list</i>	the ContributoriesList
-------------	-------------------------------

8.31.3.2 void Gui::Widgets::ContributoriesWidget::addProject (QPair< Project *, Rate > * *p* = 0) [slot]

[ContributoriesWidget::addProject](#) Add a Projet and it rate *p*

Parameters

<i>p</i>	Rate linked to Project
----------	------------------------

8.31.3.3 int Gui::Widgets::ContributoriesWidget::count ()

[ContributoriesWidget::count](#) Numbers of contributories.

Returns

Numbers of contributories

8.31.3.4 ContributoriesList * Gui::Widgets::ContributoriesWidget::getContributories () const

[ContributoriesWidget::getContributories](#) Get contributories List.

Returns

ContributoriesList

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

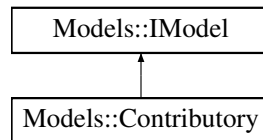
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/contributorieswidget.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/contributorieswidget.cpp

8.32 Models::Contributory Class Reference

The [Contributory](#) class.

```
#include <contributory.h>
```

Inheritance diagram for Models::Contributory:



Public Member Functions

- [Contributory](#) ()
Contributory::Contributory Construct a [Contributory](#).
- [Contributory](#) (int id)
Contributory::Contributory Construct a [Contributory](#) and get data in database.
- [~Contributory](#) ()
Destroy an contributory object.
- void [commit](#) ()
Contributory::commit Update or insert a contributory to the database.
- void [hydrat](#) (int id)
Contributory::hydrat Get data about the [Contributory](#) which is specified by the identify id
- void [remove](#) ()
Contributory::remove Remove the current [Contributory](#).
- QVariantHash [getDataMap](#) ()
getDataMap Get all data of model with a HashMap key/value
- [Project](#) * [getProject](#) () const
Contributory::getProject Return the project linked to this [Contributory](#).
- void [setProject](#) ([Project](#) *id)
Contributory::setProject Modify the identify id of the [Project](#) linked to this [Contributory](#).
- double [getNbHours](#) () const
getNbHours Number of work hour of a contributory
- void [setNbHours](#) (double value)
setNbHours Change nbHours
- QString [getDescription](#) () const
getDescription Description of a contributory
- void [setDescription](#) (const QString &[getDescription](#))
setDescription Change the contributory description
- bool [operator==](#) (const [Contributory](#) &c)
operator == define the operator "==" to compare two [Contributory](#)
- bool [operator!=](#) (const [Contributory](#) &c)
operator != define the operator "!=" to compare two [Contributory](#)
- QString [getLongDescription](#) () const
- void [setLongDescription](#) (const QString &getLongDescription)

Additional Inherited Members

8.32.1 Detailed Description

The [Contributory](#) class.

Author

8.32.2 Constructor & Destructor Documentation

8.32.2.1 Models::Contributory::Contributory (int *id*)

[Contributory::Contributory](#) Construct a [Contributory](#) and get data in database.

Parameters

<i>id</i>	Contributory 's id
-----------	------------------------------------

8.32.3 Member Function Documentation

8.32.3.1 QVariantHash Models::Contributory::getDataMap () [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements [Models::IModel](#).

8.32.3.2 QString Models::Contributory::getDescription () const

getDescription Description of a contributory

Returns

The description

8.32.3.3 double Models::Contributory::getNbHours () const

getNbHours Number of work hour of a contributory

Returns

Then number of hours

8.32.3.4 Project * Models::Contributory::getProject () const

[Contributory::getProject](#) Return the project linked to this [Contributory](#).

Returns

[Project](#) linked to this [Contributory](#)

8.32.3.5 void Models::Contributory::hydrat (int *id*) [virtual]

[Contributory::hydrat](#) Get data about the [Contributory](#) which is specified by the identify *id*

Parameters

<i>id</i>	Contributory identify
-----------	---------------------------------------

Implements [Models::IModel](#).

8.32.3.6 bool Models::Contributory::operator!= (const Contributory & c)

operator != define the operator "!=" to compare two [Contributory](#)

Parameters

<i>c</i>	the Contributory to compare with the current Contributory
----------	---

Returns

true if the [Contributory](#) are different else false

8.32.3.7 bool Models::Contributory::operator== (const Contributory & c)

operator == define the operator "==" to compare two [Contributory](#)

Parameters

<i>c</i>	the Contributory to compare with the current Contributory
----------	---

Returns

true if the [Contributory](#) are equals else false

8.32.3.8 void Models::Contributory::setDescription (const QString & getDescription)

setDescription Change the contributory description

Parameters

<i>getDescription</i>	The new description
-----------------------	---------------------

8.32.3.9 void Models::Contributory::setNbHours (double value)

setNbHours Change nbHours

Parameters

<i>value</i>	The new value of nbHours
--------------	--------------------------

8.32.3.10 void Models::Contributory::setProject (Project * id)

[Contributory::setProject](#) Modify the identify *id* of the [Project](#) linked to this [Contributory](#).

Parameters

<i>id</i>	Project Identify
-----------	----------------------------------

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

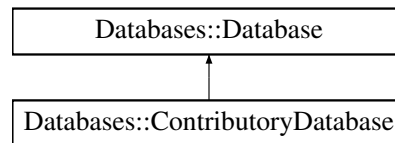
- /home/travis/build/FACT-Team/FactDev/src/models/contributory.h
- /home/travis/build/FACT-Team/FactDev/src/models/contributory.cpp

8.33 Databases::ContributoryDatabase Class Reference

The [ContributoryDatabase](#) class Contributory (or Quote) table database.

```
#include <contributorydatabase.h>
```

Inheritance diagram for Databases::ContributoryDatabase:



Public Member Functions

- [Models::Contributory](#) * [getContributory](#) (const int idContributory)
ContributoryDatabase::getCustomer get informations about the Contributory identified by pld
- [Models::ContributoriesList](#) [getContributoriesByBilling](#) (const int billingId)
ContributoryDatabase::getContributoriesByBilling get informations about the Contributory identified by Billing
- int [addContributory](#) (const [Models::Contributory](#) &)
ContributoryDatabase::addContributory Add the Contributory pContributory to the database.
- void [updateContributory](#) (const [Models::Contributory](#) &)
ContributoryDatabase::updateCustomer Update informations about the Contributory pCustomer
- void [removeContributory](#) (const int pld)
ContributoryDatabase::removeCustomer Remove the Contributory with the id pld
- [Models::Contributory](#) * [getContributory](#) (QString &q)
getContributory Obtain a contributory without new query
- [Models::ContributoriesList](#) [getContributoriesByBillingAndProject](#) (const int billingId, const int projectId)
getContributory Get contributories list by project and billing

Static Public Member Functions

- static [ContributoryDatabase](#) * [instance](#) () throw (DbException*)
ContributoryDatabase::getInstance Return an instance of [ContributoryDatabase](#)

Additional Inherited Members

8.33.1 Detailed Description

The [ContributoryDatabase](#) class Contributory (or Quote) table database.

Author

See Also

[Database](#)
[Contributory/Quote](#)

8.33.2 Member Function Documentation

8.33.2.1 int Databases::ContributoryDatabase::addContributory (const [Models::Contributory](#) & pContributory)

[ContributoryDatabase::addContributory](#) Add the Contributory pContributory to the database.

Returns

Contributory id

8.33.2.2 **Models::ContributoriesList** Databases::ContributoryDatabase::getContributoriesByBilling (const int *billingId*)

[ContributoryDatabase::getContributoriesByBilling](#) get informations about the Contributory identified by *Billing*

Parameters

<i>idBilling</i>	Contributory id
------------------	-----------------

Returns

the Contributory

8.33.2.3 Models::ContributoriesList Databases::ContributoryDatabase::getContributoriesByBillingAndProject (const int *billingId*, const int *projectId*)

getContributory Get contributories list by project and billing

Parameters

<i>billingId</i>	
<i>projectId</i>	

Returns

The contributories list by project and billing

8.33.2.4 Models::Contributory * Databases::ContributoryDatabase::getContributory (const int *idContributory*)

ContributoryDatabase::getCustomer get informations about the Contributory identified by *pId*

Parameters

<i>idContributory</i>	Contributory id
-----------------------	-----------------

Returns

the Contributory

8.33.2.5 Models::Contributory * Databases::ContributoryDatabase::getContributory (QSqlQuery & *q*)

getContributory Obtain a contributory without new query

Parameters

<i>q</i>	The query to use
----------	------------------

Returns

The contributory linked to *q*

8.33.2.6 ContributoryDatabase * Databases::ContributoryDatabase::instance () throw DbException *) [static]

ContributoryDatabase::getInstance Return an instance of [ContributoryDatabase](#)

See Also

[DbException](#)

Returns

Instance of [ContributoryDatabase](#)

8.33.2.7 void Databases::ContributoryDatabase::removeContributory (const int *pld*)

ContributoryDatabase::removeCustomer Remove the Contributory with the id *pld*

Parameters

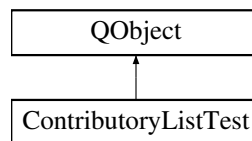
<i>pld</i>	Contributory id
------------	-----------------

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

- /home/travis/build/FACT-Team/FactDev/src/database/contributorydatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/contributorydatabase.cpp

8.34 ContributoryListTest Class Reference

Inheritance diagram for ContributoryListTest:

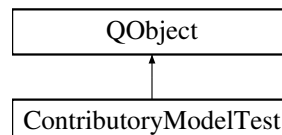


The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/tests/models/contributorylisttest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/contributorylisttest.cpp

8.35 ContributoryModelTest Class Reference

Inheritance diagram for ContributoryModelTest:

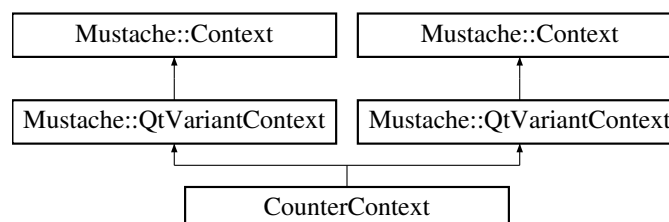


The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/tests/models/contributorymodeltest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/contributorymodeltest.cpp

8.36 CounterContext Class Reference

Inheritance diagram for CounterContext:



Public Member Functions

- **CounterContext** (const QVariantHash &map)
- virtual bool [canEval](#) (const QString &key) const
- virtual QString [eval](#) (const QString &key, const QString &_template, [Mustache::Renderer](#) *renderer)
- virtual QString [stringValue](#) (const QString &key) const
- **CounterContext** (const QVariantHash &map)
- virtual bool [canEval](#) (const QString &key) const
- virtual QString [eval](#) (const QString &key, const QString &_template, [Mustache::Renderer](#) *renderer)
- virtual QString [stringValue](#) (const QString &key) const

Public Attributes

- int **counter**

Additional Inherited Members

8.36.1 Member Function Documentation

8.36.1.1 virtual bool CounterContext::canEval (const QString & key) const [inline], [virtual]

Returns true if [eval\(\)](#) should be used to render section tags using `key`. If [canEval\(\)](#) returns true for a key, the renderer will pass the literal, unrendered block of text for the section to [eval\(\)](#) and replace the section with the result.

[canEval\(\)](#) and [eval\(\)](#) are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented from [Mustache::QtVariantContext](#).

8.36.1.2 virtual bool CounterContext::canEval (const QString & key) const [inline], [virtual]

Returns true if [eval\(\)](#) should be used to render section tags using `key`. If [canEval\(\)](#) returns true for a key, the renderer will pass the literal, unrendered block of text for the section to [eval\(\)](#) and replace the section with the result.

[canEval\(\)](#) and [eval\(\)](#) are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented from [Mustache::QtVariantContext](#).

8.36.1.3 virtual QString CounterContext::eval (const QString & key, const QString & _template, [Mustache::Renderer](#) * *renderer*) [inline], [virtual]

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of [eval\(\)](#).

The default implementation returns an empty string.

Reimplemented from [Mustache::QtVariantContext](#).

8.36.1.4 virtual QString CounterContext::eval (const QString & key, const QString & _template, [Mustache::Renderer](#) * *renderer*) [inline], [virtual]

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of [eval\(\)](#).

The default implementation returns an empty string.

Reimplemented from [Mustache::QtVariantContext](#).

8.36.1.5 virtual QString CounterContext::stringValue (const QString & key) const [inline], [virtual]

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

Reimplemented from [Mustache::QtVariantContext](#).

8.36.1.6 virtual QString CounterContext::stringValue (const QString & key) const [inline], [virtual]

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

Reimplemented from [Mustache::QtVariantContext](#).

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

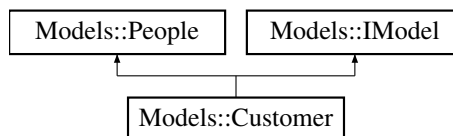
- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/tests/test_mustache.cpp

8.37 Models::Customer Class Reference

The [Customer](#) class [Customer](#).

```
#include <customer.h>
```

Inheritance diagram for Models::Customer:



Public Member Functions

- [Customer](#) ()
Customer::Customer Construct a [Customer](#).
- [Customer](#) (int id)
Customer::Customer Construct a [Customer](#) who is specified by id
- void [commit](#) ()
Customer::commit Update customer data on the database.
- void [hydrat](#) (int id)
Customer::hydrat Insert into database informations related to the [Customer](#) who is specified by id
- void [remove](#) ()
Customer::remove Remove the current customer.
- QVariantHash [getDataMap](#) ()
getDataMap Get all data of model with a HashMap key/value
- QString [getPath](#) () const
Customer::getPath Return the path of the workspace for the current [Customer](#).
- QString [getNameFolder](#) () const
Customer::getNameFolder Return the name of the current [Customer](#)'s folder in the workspace.
- double [getTurnover](#) () const
Customer::getTurnover Return the turnover of the customer money that customer pay, revenue sales.
- bool [isArchived](#) () const
Customer::isArchived Return if the [Customer](#) is archived.

- void [setIsArchived](#) (const bool [isArchived](#))
[Customer::setIsArchived](#) set the *isArchived* parameter.

Additional Inherited Members

8.37.1 Detailed Description

The [Customer](#) class [Customer](#).

Author

Antoine de Roquemaurel
 Florent Berbie

8.37.2 Constructor & Destructor Documentation

8.37.2.1 Models::Customer::Customer (int *id*)

[Customer::Customer](#) Construct a [Customer](#) who is specified by *id*

Parameters

<i>id</i>	Customer identify
-----------	-----------------------------------

8.37.3 Member Function Documentation

8.37.3.1 QVariantHash Models::Customer::getDataMap () [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements [Models::IModel](#).

8.37.3.2 QString Models::Customer::getNameFolder () const

[Customer::getNameFolder](#) Return the name of the current [Customer](#)'s folder in the workspace.

Returns

name of the [Customer](#)'s folder

8.37.3.3 QString Models::Customer::getPath () const

[Customer::getPath](#) Return the path of the workspace for the current [Customer](#).

Returns

workspace path

8.37.3.4 `double Models::Customer::getTurnover () const`

[Customer::getTurnover](#) Return the turnover of the customer money that customer pay, revenue sales.

Returns

turnover

8.37.3.5 `void Models::Customer::hydrat (int id) [virtual]`

[Customer::hydrat](#) Insert into database informations related to the [Customer](#) who is specified by *id*

Parameters

<i>id</i>	Customer identify
-----------	-----------------------------------

Implements [Models::IModel](#).

8.37.3.6 `bool Models::Customer::isArchived () const`

[Customer::isArchived](#) Return if the [Customer](#) is archived.

Returns

true or false

8.37.3.7 `void Models::Customer::setIsArchived (const bool isArchived)`

[Customer::setIsArchived](#) set the *isArchived* parameter.

Parameters

<i>isArchived</i>	
-------------------	--

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

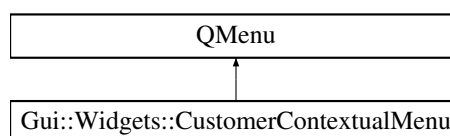
- /home/travis/build/FACT-Team/FactDev/src/models/customer.h
- /home/travis/build/FACT-Team/FactDev/src/models/customer.cpp

8.38 Gui::Widgets::CustomerContextualMenu Class Reference

Display contextual menu on a customer.

```
#include <customercontextualmenu.h>
```

Inheritance diagram for Gui::Widgets::CustomerContextualMenu:



Public Member Functions

- [CustomerContextualMenu](#) (QWidget *w=0)
CustomerContextualMenu::CustomerContextualMenu Construct a new contextual menu.
- [~CustomerContextualMenu](#) ()
CustomerContextualMenu::~Destruct the contextual menu.

8.38.1 Detailed Description

Display contextual menu on a customer.

Author

Antoine de Roquemaurel

8.38.2 Constructor & Destructor Documentation

8.38.2.1 Gui::Widgets::CustomerContextualMenu::CustomerContextualMenu (QWidget * w = 0)

[CustomerContextualMenu::CustomerContextualMenu](#) Construct a new contextual menu.

Parameters

<i>w</i>	Parent
----------	--------

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

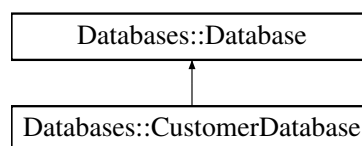
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/customercontextualmenu.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/customercontextualmenu.cpp

8.39 Databases::CustomerDatabase Class Reference

The [CustomerDatabase](#) class Customer table database.

```
#include <customerdatabase.h>
```

Inheritance diagram for Databases::CustomerDatabase:



Public Member Functions

- [WdgModels::CustomersTableModel * getCustomersTable](#) (QString filter="") throw (DbException*)
CustomerDatabase::getCustomersTable Return an item model of customers for QTableView.
- [QStandardItemModel * getTree](#) (QString filter="") throw (DbException*)
CustomerDatabase::getTree Return an item model of customers for QTree.
- [QSharedPointer< Models::Customer > getCustomer](#) (const int pld)
CustomerDatabase::getCustomer get informations about the customer identified by pld
- [int addCustomer](#) (const [Models::Customer](#) &)
CustomerDatabase::addCustomer Add the customer pCustomer to the database.

- void [updateCustomer](#) ([Customer](#) &)
*[CustomerDatabase::updateCustomer](#) Update informations about the customer *pCustomer**
- void [removeCustomer](#) (const int *pld*)
*[CustomerDatabase::removeCustomer](#) Remove the customer with the id *pld**
- int [getNbCustomers](#) ()
[CustomerDatabase::getNbCustomers](#) Return the number of customers existing.
- QStandardItem * [getItemRoot](#) ()
[CustomerDatabase::getItemRoot](#) Return the first item for the QStandardItemModel.
- QStandardItem * [getItemCustomer](#) (QStringQuery *q1*)
[CustomerDatabase::getItemCustomer](#) Return the customer item for the QStandardItemModel.
- QStandardItem * [getItemProject](#) (QStringQuery *q2*)
[CustomerDatabase::getItemProject](#) Return the project item for the QStandardItemModel.
- QSharedPointer< [Models::Customer](#) > [getCustomer](#) (QStringQuery &*q*)
*[CustomerDatabase::getCustomer](#) Add the element of the *q* request and return their.*
- void [updateCustomer](#) (QStringQuery &*q*, [Customer](#) &*pCustomer*)
*[CustomerDatabase::updateCustomer](#) Update customer data according to the request *q**

Static Public Member Functions

- static [CustomerDatabase](#) * [instance](#) () throw (DbException*)
[CustomerDatabase::instance](#) Return an instance of [CustomerDatabase](#)

Additional Inherited Members

8.39.1 Detailed Description

The [CustomerDatabase](#) class Customer table database.

Author

Antoine de Roquemaurel
Manantsoa Razanajatovo
Florent Berbie

See Also

[Database](#)
[Customer](#)

8.39.2 Member Function Documentation

8.39.2.1 int [Databases::CustomerDatabase::addCustomer](#) (const [Models::Customer](#) & *pCustomer*)

[CustomerDatabase::addCustomer](#) Add the customer *pCustomer* to the database.

Returns

customer id

8.39.2.2 QSharedPointer< [Models::Customer](#) > [Databases::CustomerDatabase::getCustomer](#) (const int *pld*)

[CustomerDatabase::getCustomer](#) get informations about the customer identified by *pld*

Parameters

<i>pId</i>	customer id
------------	-------------

Returns

the Customer

8.39.2.3 QSharedPointer< Models::Customer > Databases::CustomerDatabase::getCustomer (QSqlQuery & *q*)

[CustomerDatabase::getCustomer](#) Add the element of the *q* request and return their.

Parameters

<i>q</i>	SQL request
----------	-------------

Returns

a customer formed according to QSharedPointer

8.39.2.4 WdgModels::CustomersTableModel * Databases::CustomerDatabase::getCustomersTable (QString *filter* = " ") throw DbException *)

[CustomerDatabase::getCustomersTable](#) Return an item model of customers for QTableView.

Parameters

<i>filter</i>	Select only customers who are specified by <i>filter</i>
---------------	--

Exceptions

<i>DbException</i>	
--------------------	--

Returns

QStandardItemModel an item model for QTableView

8.39.2.5 QStandardItem * Databases::CustomerDatabase::getItemCustomer (QSqlQuery *q1*)

[CustomerDatabase::getItemCustomer](#) Return the customer item for the QStandardItemModel.

Parameters

<i>q1</i>	the row of the sql query for customers
-----------	--

Returns

QStandardItem an item for QTree (level/depth 1)

8.39.2.6 QStandardItem * Databases::CustomerDatabase::getItemProject (QSqlQuery *q2*)

[CustomerDatabase::getItemProject](#) Return the project item for the QStandardItemModel.

Parameters

<i>q2</i>	the row of the sql query for projects
-----------	---------------------------------------

Returns

QStandardItem an item for QTree (level/depth 2)

8.39.2.7 QStandardItem * Databases::CustomerDatabase::getItemRoot ()

[CustomerDatabase::getItemRoot](#) Return the first item for the QStandardItemModel.

Returns

QStandardItem an item for QTree (level/depth 0)

8.39.2.8 int Databases::CustomerDatabase::getNbCustomers ()

[CustomerDatabase::getNbCustomers](#) Return the number of customers existing.

Returns

number of customers

8.39.2.9 QStandardItemModel * Databases::CustomerDatabase::getTree (QString *filter* = " ") throw DbException *)

[CustomerDatabase::getTree](#) Return an item model of customers for QTree.

Parameters

<i>filter</i>	Select only customers who are specified by <i>filter</i>
---------------	--

Exceptions

<i>DbException</i>	
--------------------	--

Returns

QStandardItemModel an item model for QTreeView

8.39.2.10 CustomerDatabase * Databases::CustomerDatabase::instance () throw DbException *) [static]

[CustomerDatabase::instance](#) Return an instance of [CustomerDatabase](#)

See Also

DbException

Returns

Instance of [CustomerDatabase](#)

8.39.2.11 void Databases::CustomerDatabase::removeCustomer (const int *pld*)

[CustomerDatabase::removeCustomer](#) Remove the customer with the id *pld*

Parameters

<i>pId</i>	customer id
------------	-------------

8.39.2.12 void Databases::CustomerDatabase::updateCustomer (QSqlQuery & *q*, Customer & *pCustomer*)

[CustomerDatabase::updateCustomer](#) Update customer data according to the request *q*

Parameters

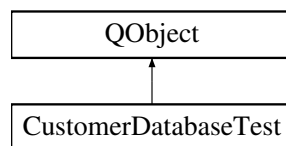
<i>q</i>	SQL request
----------	-------------

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

- /home/travis/build/FACT-Team/FactDev/src/database/customerdatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/customerdatabase.cpp

8.40 CustomerDatabaseTest Class Reference

Inheritance diagram for CustomerDatabaseTest:



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

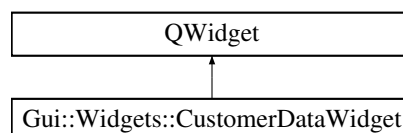
- /home/travis/build/FACT-Team/FactDev/tests/database/customerdatabasetest.h
- /home/travis/build/FACT-Team/FactDev/tests/database/customerdatabasetest.cpp

8.41 Gui::Widgets::CustomerDataWidget Class Reference

Class for display info of a customer.

```
#include <customerdatawidget.h>
```

Inheritance diagram for Gui::Widgets::CustomerDataWidget:



Public Member Functions

- [CustomerDataWidget](#) (QWidget *parent=0)
[CustomerDataWidget::CustomerDataWidget](#) Construct a [CustomerDataWidget](#).
- void [printUserData](#) ()
[CustomerDataWidget::printUserData](#) Print Data of current user.
- void [printInformations](#) (int id)
[CustomerDataWidget::printInformations](#) Print Data of customer id.

8.41.1 Detailed Description

Class for display info of a customer.

Author

8.41.2 Constructor & Destructor Documentation

8.41.2.1 `Gui::Widgets::CustomerDataWidget::CustomerDataWidget (QWidget * parent = 0) [explicit]`

[CustomerDataWidget::CustomerDataWidget](#) Construct a [CustomerDataWidget](#).

Parameters

<i>parent</i>	Widget parent
---------------	---------------

8.41.3 Member Function Documentation

8.41.3.1 `void Gui::Widgets::CustomerDataWidget::printInformations (int id)`

[CustomerDataWidget::printInformations](#) Print Data of customer id.

Parameters

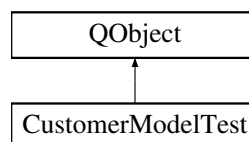
<i>id</i>	of customer to print
-----------	----------------------

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

- `/home/travis/build/FACT-Team/FactDev/src/gui/widgets/customerdatawidget.h`
- `/home/travis/build/FACT-Team/FactDev/src/gui/widgets/customerdatawidget.cpp`

8.42 CustomerModelTest Class Reference

Inheritance diagram for CustomerModelTest:



Public Member Functions

- `void setup ()`

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

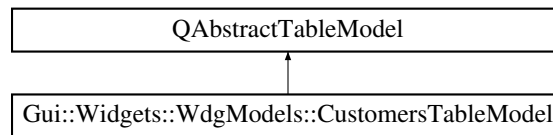
- `/home/travis/build/FACT-Team/FactDev/tests/models/customermodeltest.h`
- `/home/travis/build/FACT-Team/FactDev/tests/models/customermodeltest.cpp`

8.43 Gui::Widgets::WdgModels::CustomersTableModel Class Reference

The [CustomersTableModel](#) class for a customer table.

```
#include <customerstablemodel.h>
```

Inheritance diagram for Gui::Widgets::WdgModels::CustomersTableModel:



Public Member Functions

- [CustomersTableModel](#) (QObject *parent=0)
CustomersTableModel::CustomersTableModel Construct a [CustomersTableModel](#).
- int [rowCount](#) (const QModelIndex &) const
CustomersTableModel::rowCount Number of customers row.
- int [columnCount](#) (const QModelIndex &) const
CustomersTableModel::columnCount Number of column of a customer.
- QVariant [data](#) (const QModelIndex &index, int role=Qt::DisplayRole) const
CustomersTableModel::data Obtains data of a specify cell.
- QVariant [headerData](#) (int section, Qt::Orientation orientation, int role=Qt::DisplayRole) const
CustomersTableModel::headerData Obtains header title of table.
- bool [setData](#) (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)
CustomersTableModel::setData Change data of a cell.
- void [append](#) (const [Customer](#) &customer)
CustomersTableModel::append Add a new line in table.
- void [remove](#) (const int i)
CustomersTableModel::remove Remove a line.
- Qt::ItemFlags [flags](#) (const QModelIndex &index) const
CustomersTableModel::flags Differents table flags.
- int [count](#) ()
CustomersTableModel::count Number of customers in table.
- QList< [Customer](#) > [getCustomers](#) () const
CustomersTableModel::getCustomers Return the list of customers.

8.43.1 Detailed Description

The [CustomersTableModel](#) class for a customer table.

Author

Florent Berbie

See Also

[Customer](#)

8.43.2 Constructor & Destructor Documentation

8.43.2.1 Gui::Widgets::WdgModels::CustomersTableModel::CustomersTableModel (*QObject* * *parent* = 0)

[CustomersTableModel::CustomersTableModel](#) Construct a [CustomersTableModel](#).

Parameters

<i>parent</i>	Parent widget
---------------	---------------

8.43.3 Member Function Documentation

8.43.3.1 void Gui::Widgets::WdgModels::CustomersTableModel::append (const Customer & *customer*)

[CustomersTableModel::append](#) Add a new line in table.

Parameters

<i>Customer</i>	The new customer
-----------------	------------------

8.43.3.2 int Gui::Widgets::WdgModels::CustomersTableModel::columnCount (const QModelIndex &) const

[CustomersTableModel::columnCount](#) Number of column of a customer.

Returns

The number of column

8.43.3.3 int Gui::Widgets::WdgModels::CustomersTableModel::count ()

[CustomersTableModel::count](#) Number of customers in table.

Returns

The number of customers

8.43.3.4 QVariant Gui::Widgets::WdgModels::CustomersTableModel::data (const QModelIndex & *index*, int *role* = Qt::DisplayRole) const

[CustomersTableModel::data](#) Obtains data of a specify cell.

Parameters

<i>index</i>	The cell who we want data
<i>role</i>	The role of set

Returns

The data of cell

8.43.3.5 Qt::ItemFlags Gui::Widgets::WdgModels::CustomersTableModel::flags (const QModelIndex & *index*) const

[CustomersTableModel::flags](#) Differents table flags.

Parameters

<i>index</i>	The cell who we want to know flags
--------------	------------------------------------

Returns

Flags

8.43.3.6 `QList< Customer > Gui::Widgets::WdgModels::CustomersTableModel::getCustomers () const`

[CustomersTableModel::getCustomers](#) Return the list of customers.

Returns

list of Customers

8.43.3.7 `QVariant Gui::Widgets::WdgModels::CustomersTableModel::headerData (int section, Qt::Orientation orientation, int role = Qt::DisplayRole) const`

[CustomersTableModel::headerData](#) Obtains header title of table.

Parameters

<i>section</i>	The number of column
<i>orientation</i>	The table orientation
<i>role</i>	

Returns

The Title header of column

8.43.3.8 `void Gui::Widgets::WdgModels::CustomersTableModel::remove (const int i)`

[CustomersTableModel::remove](#) Remove a line.

Parameters

<i>i</i>	The number of line to remove
----------	------------------------------

8.43.3.9 `int Gui::Widgets::WdgModels::CustomersTableModel::rowCount (const QModelIndex &) const`

[CustomersTableModel::rowCount](#) Number of customers row.

Returns

The number of customers

8.43.3.10 `bool Gui::Widgets::WdgModels::CustomersTableModel::setData (const QModelIndex & index, const QVariant & value, int role = Qt::EditRole)`

[CustomersTableModel::setData](#) Change data of a cell.

Parameters

<i>index</i>	The cell to change data
<i>value</i>	The new value
<i>role</i>	The role of cell

Returns

True if we could edit

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

- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetsmodels/customerstablemodel.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetsmodels/customerstablemodel.cpp

8.44 Databases::Database Class Reference

The **Database** class Master class for all database access.

```
#include <database.h>
```

Inheritance diagram for Databases::Database:



Public Member Functions

- QString **lastError** (const QSqlQuery &q) const
Database::lastError Return an error message on the last error occurred during the SQL request q
- void **testCases** ()
Database::testCases Realise a test cases.
- void **executeFile** (QString pName)
Database::executerFichier Exeute a specified file named pName
- void **openTransaction** ()
Database::openTransaction Open new transaction.
- void **closeTransaction** ()
Database::closeTransaction Close current transaction.
- void **close** ()
Database::close Close database access.
- void **open** ()
Database::open Open database.
- **~Database** ()
Database::~Database Suppression object, and close database access.
- void **setDatabase** (QSqlDatabase sql)
Database::setDatabase Set database.
- void **updateBillingNumber** ()
Database::updateBillingNumber Update the billing number.
- void **cleanDatabase** ()
Database::clearDatabase Drop alls tables of *Database* WARNING: We can't restore data after.
- void **changeDatabase** (Databases::DbType dbType)
changeDatabase Change the current database : mysql to sqlite or sqlite to mysql

Static Public Member Functions

- static Database * **instance** (bool tests=false) throw (DbException*)
Database::getInstance Return an instance of *Database*.

Protected Member Functions

- Database (bool tests=false) throw (DbException*)
Database::Database *Database* is a singleton.
- QVariant **value** (const QSqlQuery &q, const QString &champ) const
Database::valeur Value of database field.

Protected Attributes

- QSettings * [_settings](#)
settings
- QSqlDatabase [mDatabase](#)
contains Database
- QList< [Database](#) * > [_instances](#)
List of instances.

Static Protected Attributes

- static [Database](#) * [_instance](#) = 0
Instance.
- static bool [_dbInstance](#) = 0
an instance of db is open
- static bool [isOpen](#) = false
[Database](#) is open.
- static bool [_isMysql](#) = false

8.44.1 Detailed Description

The [Database](#) class Master class for all database access.

Author

Antoine de Roquemaurel

8.44.2 Member Function Documentation

8.44.2.1 void Databases::Database::changeDatabase (Databases::DbType *dbType*)

changeDatabase Change the current database : mysql to sqlite or sqlite to mysql

Parameters

<i>dbType</i>	: The new database type, Sqlite or Mysql
---------------	--

8.44.2.2 void Databases::Database::executeFile (QString *pName*)

Database::executerFichier Exeute a specified file named *pName*

Parameters

<i>pNom</i>	File name
-------------	-----------

8.44.2.3 Database * Databases::Database::instance (bool *tests* = false) throw DbException * [static]

Database::getInstance Return an instance of [Database](#).

Returns

Instance of [Database](#)

8.44.2.4 QString Databases::Database::lastError (const QSqlQuery & *q*) const [inline]

[Database::lastError](#) Return an error message on the last error occurred during the SQL request *q*

Parameters

<i>q</i>	SQL request
----------	-------------

Returns

an error message

8.44.2.5 void Databases::Database::setDatabase (QSqlDatabase *sql*)

[Database::setDatabase](#) Set database.

Parameters

<i>sql</i>	The new database
------------	------------------

8.44.2.6 QVariant Databases::Database::value (const QSqlQuery & *q*, const QString & *champ*) const [protected]

Database::valeur Value of database field.

Parameters

<i>q</i>	Query
<i>champ</i>	Field

Returns

The value

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

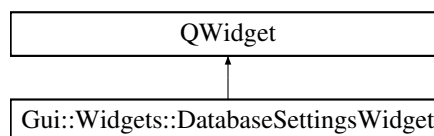
- /home/travis/build/FACT-Team/FactDev/src/database/database.h
- /home/travis/build/FACT-Team/FactDev/src/database/database.cpp

8.45 Gui::Widgets::DatabaseSettingsWidget Class Reference

The [DatabaseSettingsWidget](#) class Windows of database settings.

```
#include <databasesettingswidget.h>
```

Inheritance diagram for Gui::Widgets::DatabaseSettingsWidget:



Public Slots

- bool [isValid](#) ()
DatabaseSettingsWidget::isValid Return TRUE if all fields are correctly inputed else FALSE.
- void [checkRepeatPassword](#) (QString text)
DatabaseSettingsWidget::checkRepeatLogin Check if the second login field is the same than the first.
- void [userInterfaceChanged](#) ()
DatabaseSettingsWidget::userInterfaceChanged User interface has changed.

Signals

- void [textfieldChanged](#) ()
[DatabaseSettingsWidget::textfieldChanged](#) Signal which indicates if a fieldtext has changed.

Public Member Functions

- [DatabaseSettingsWidget](#) (QWidget *parent=0)
[DatabaseSettingsWidget::DatabaseSettingsWidget](#) Construct a [DatabaseSettingsWidget](#).
- void [fillFields](#) ()
[DatabaseSettingsWidget::fillFields](#) Complete fields with a default value for field Database name, Username, IP address and port.
- QString [getDatabaseName](#) ()
[DatabaseSettingsWidget::getDatabaseName](#) Return the database name.
- QString [getLogin](#) ()
[DatabaseSettingsWidget::getLogin](#) Return the login of the user.
- QString [getPassword](#) ()
[DatabaseSettingsWidget::getPassword](#) Return the password of the user.
- QString [getDomainNameOrIP](#) ()
[DatabaseSettingsWidget::getDomainNameOrIP](#) Return the domain name or IP address on the machine where is the database.
- QString [getPort](#) ()
[DatabaseSettingsWidget::getPort](#) Return the port of the database.

8.45.1 Detailed Description

The [DatabaseSettingsWidget](#) class Windows of database settings.

Author

8.45.2 Constructor & Destructor Documentation

8.45.2.1 [Gui::Widgets::DatabaseSettingsWidget::DatabaseSettingsWidget](#) ([QWidget](#) * *parent* = 0) [explicit]

[DatabaseSettingsWidget::DatabaseSettingsWidget](#) Construct a [DatabaseSettingsWidget](#).

Parameters

<i>parent</i>	Parent widget of this windows
---------------	-------------------------------

8.45.3 Member Function Documentation

8.45.3.1 [QString](#) [Gui::Widgets::DatabaseSettingsWidget::getDatabaseName](#) ()

[DatabaseSettingsWidget::getDatabaseName](#) Return the database name.

Returns

Database name

8.45.3.2 QString Gui::Widgets::DatabaseSettingsWidget::getDomainNameOrIP ()

[DatabaseSettingsWidget::getDomainNameOrIP](#) Return the domain name or IP address on the machine where is the database.

Returns

Domain name or IP address

8.45.3.3 QString Gui::Widgets::DatabaseSettingsWidget::getLogin ()

[DatabaseSettingsWidget::getLogin](#) Return the login of the user.

Returns

Login of the user

8.45.3.4 QString Gui::Widgets::DatabaseSettingsWidget::getPassword ()

[DatabaseSettingsWidget::getPassword](#) Return the password of the user.

Returns

User password

8.45.3.5 QString Gui::Widgets::DatabaseSettingsWidget::getPort ()

[DatabaseSettingsWidget::getPort](#) Return the port of the database.

Returns

Database port

8.45.3.6 bool Gui::Widgets::DatabaseSettingsWidget::isValid () [slot]

[DatabaseSettingsWidget::isValid](#) Return TRUE if all fields are correctly inputed else FALSE.

Returns

boolean

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

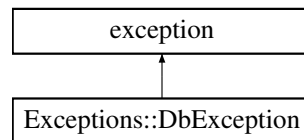
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/databasesettingswidget.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/databasesettingswidget.cpp

8.46 Exceptions::DbException Class Reference

The [DbException](#) class for database exception : queries, db file, ...

```
#include <dbexception.h>
```

Inheritance diagram for Exceptions::DbException:



Public Member Functions

- [DbException](#) (const QString fct, const QString fctName, const QString logError, float errorCode)
[Exceptions::DbException](#). Construct a [DbException](#).
- virtual [~DbException](#) () throw ()
[~DbException](#)
- void [popupMessage](#) (QWidget *parent)
[DbException::popupMessage](#). Display a popup message with the message error.

8.46.1 Detailed Description

The [DbException](#) class for database exception : queries, db file, ...

Author

Antoine de Roquemaurel

8.46.2 Constructor & Destructor Documentation

8.46.2.1 [Exceptions::DbException::DbException](#) (const QString *fct*, const QString *fctName*, const QString *logError*, float *errorCode*)

[DbException::DbException](#). Construct a [DbException](#).

Parameters

<i>userError</i>	ClassName of error
<i>fctName</i>	Function name
<i>logError</i>	Message error
<i>errorCode</i>	Code of error

8.46.3 Member Function Documentation

8.46.3.1 void [Exceptions::DbException::popupMessage](#) (QWidget * *parent*)

[DbException::popupMessage](#). Display a popup message with the message error.

Parameters

<i>parent</i>	
---------------	--

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

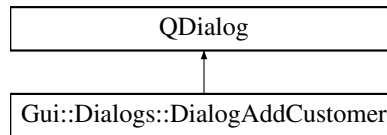
- /home/travis/build/FACT-Team/FactDev/src/exceptions/dbexception.h
- /home/travis/build/FACT-Team/FactDev/src/exceptions/dbexception.cpp

8.47 Gui::Dialogs::DialogAddCustomer Class Reference

The [DialogAddCustomer](#) class Window to add or modify a Customer.

```
#include <dialogaddcustomer.h>
```

Inheritance diagram for Gui::Dialogs::DialogAddCustomer:



Public Slots

- void [checkFields](#) ()
[DialogAddCustomer::checkFields](#) Check if fields are valid.

Public Member Functions

- [DialogAddCustomer](#) (int id=0, QWidget *parent=0)
[DialogAddCustomer::DialogAddCustomer](#) Construct a window to add/modify a Customer.
- void [fillFields](#) ()
[DialogAddCustomer::fillFields](#) If the Customer exists, fill line edits with the data of the current Customer.
- void [accept](#) ()
[DialogAddCustomer::accept](#) Valid data inputed by user and add these data in Database.
- void [reject](#) ()
[DialogAddCustomer::reject](#) Cancel the operation and close the windows.

8.47.1 Detailed Description

The [DialogAddCustomer](#) class Window to add or modify a Customer.

Author

8.47.2 Constructor & Destructor Documentation

8.47.2.1 [Gui::Dialogs::DialogAddCustomer::DialogAddCustomer](#) (int *id* = 0, QWidget * *parent* = 0) [explicit]

[DialogAddCustomer::DialogAddCustomer](#) Construct a window to add/modify a Customer.

Parameters

<i>id</i>	Customer id
<i>parent</i>	QWidget parent

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

- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/dialogaddcustomer.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/dialogaddcustomer.cpp

8.48 Utils::Directories Class Reference

Static Public Member Functions

- static QString [makeDirectory](#) (QDir &directory, const QString path, const QString folder) throw (Exceptions::FileException*)

MainWindow::makeDirectory If not exists make a new directory folder

8.48.1 Member Function Documentation

8.48.1.1 QString [Utils::Directories::makeDirectory](#) (QDir & *directory*, const QString *path*, const QString *folder*) throw Exceptions::FileException *) [static]

MainWindow::makeDirectory If not exists make a new directory *folder*

Parameters

<i>path</i>	Return the path of the folder just created
<i>folder</i>	Folder name to create

Returns

Path of the folder just created

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

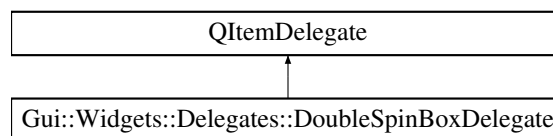
- /home/travis/build/FACT-Team/FactDev/src/Utils/directories.h
- /home/travis/build/FACT-Team/FactDev/src/Utils/directories.cpp

8.49 Gui::Widgets::Delegates::DoubleSpinBoxDelegate Class Reference

The [DoubleSpinBoxDelegate](#) class.

```
#include <doublespinboxdelegate.h>
```

Inheritance diagram for Gui::Widgets::Delegates::DoubleSpinBoxDelegate:



Public Member Functions

- [DoubleSpinBoxDelegate](#) (QObject *parent=0)
DoubleSpinBoxDelegate::DoubleSpinBoxDelegate.
- QWidget * [createEditor](#) (QWidget *parent, const QStyleOptionViewItem &option, const QModelIndex &index) const
DoubleSpinBoxDelegate::createEditor Return a ComboBox specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void [setEditorData](#) (QWidget *editor, const QModelIndex &index) const
DoubleSpinBoxDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index

- void [setModelData](#) (QWidget *editor, QAbstractItemModel *model, const QModelIndex &index) const
DoubleSpinBoxDelegate::setEditorData Sets the data to be displayed and edited by the editor from the data model item specified by the model index
- void [updateEditorGeometry](#) (QWidget *editor, const QStyleOptionViewItem &option, const QModelIndex &index) const
DoubleSpinBoxDelegate::updateEditorGeometry Update the editor for the item specified by index according to the style option given.

8.49.1 Detailed Description

The [DoubleSpinBoxDelegate](#) class.

Author

Florent Berbie

8.49.2 Constructor & Destructor Documentation

8.49.2.1 [Gui::Widgets::Delegates::DoubleSpinBoxDelegate::DoubleSpinBoxDelegate](#) (QObject * *parent* = 0)

[DoubleSpinBoxDelegate::DoubleSpinBoxDelegate](#).

Parameters

<i>parent</i>	
---------------	--

8.49.3 Member Function Documentation

8.49.3.1 [QWidget * Gui::Widgets::Delegates::DoubleSpinBoxDelegate::createEditor](#) (QWidget * *parent*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const

[DoubleSpinBoxDelegate::createEditor](#) Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Returns

[DoubleSpinBoxDelegate](#)

8.49.3.2 [void Gui::Widgets::Delegates::DoubleSpinBoxDelegate::setEditorData](#) (QWidget * *editor*, const QModelIndex & *index*) const

[DoubleSpinBoxDelegate::setEditorData](#) Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

<i>editor</i>	Data edited
<i>index</i>	Index of the model to edit

8.49.3.3 void Gui::Widgets::Delegates::DoubleSpinBoxDelegate::setModelData (QWidget * *editor*, QAbstractItemModel * *model*, const QModelIndex & *index*) const

[DoubleSpinBoxDelegate::setEditorData](#) Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

<i>editor</i>	Data edited
<i>index</i>	Index of the model to edit

8.49.3.4 void Gui::Widgets::Delegates::DoubleSpinBoxDelegate::updateEditorGeometry (QWidget * *editor*, const QStyleOptionViewItem & *option*, const QModelIndex & *index*) const

[DoubleSpinBoxDelegate::updateEditorGeometry](#) Update the *editor* for the item specified by *index* according to the style *option* given.

Parameters

<i>editor</i>	Editor widget to update
<i>option</i>	Style option
<i>index</i>	Item index

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

- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/doublespinboxdelegate.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/doublespinboxdelegate.cpp

8.50 Exceptions::FileException Class Reference

The [FileException](#) class for file/access file exception.

```
#include <fileexception.h>
```

Public Member Functions

- [FileException](#) (const QString userError, const QString fctName, const QString logError, float errorCode)
FileException::FileException. Construct a *FileException*.
- void [popupMessage](#) (QWidget *parent)
FileException::popupMessage. Display a popup message with the message error.

8.50.1 Detailed Description

The [FileException](#) class for file/access file exception.

Author

Florent Berbie

8.50.2 Constructor & Destructor Documentation

8.50.2.1 Exceptions::FileException::FileException (const QString *userError*, const QString *fctName*, const QString *logError*, float *errorCode*)

[FileException::FileException](#). Construct a [FileException](#).

Parameters

<i>userError</i>	ClassName of error
<i>fctName</i>	Function name
<i>logError</i>	Message error
<i>errorCode</i>	Code of error

8.50.3 Member Function Documentation

8.50.3.1 void Exceptions::FileException::popupMessage (QWidget * *parent*)

[FileException::popupMessage](#). Display a popup message with the message error.

Parameters

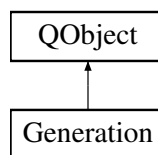
<i>parent</i>	
---------------	--

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

- /home/travis/build/FACT-Team/FactDev/src/exceptions/fileexception.h
- /home/travis/build/FACT-Team/FactDev/src/exceptions/fileexception.cpp

8.51 Generation Class Reference

Inheritance diagram for Generation:



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

- /home/travis/build/FACT-Team/FactDev/tests/generation.h
- /home/travis/build/FACT-Team/FactDev/tests/generation.cpp

8.52 Utils::HierarchicalSystem Class Reference

The [HierarchicalSystem](#) class Create class which contains hierarchical system of FactDev.

```
#include <hierarchicalsistem.h>
```

Public Member Functions

- [HierarchicalSystem](#) ()
HierarchicalSystem::HierarchicalSystem Construct a [HierarchicalSystem](#).
- void [getAllProjects](#) ()
HierarchicalSystem::getAllProjects Get all projects and add each project to Customer linked.
- void [getAllBillings](#) ()
HierarchicalSystem::getAllBillings Get all billings and add each billing to Project linked.
- void [updateData](#) ()

- [*HierarchicalSystem::updateData*](#) Update data on Customers, Projects, Billings.
- void [*addProjectToCustomer*](#) ([Project](#) *p, [Customer](#) c)
[*HierarchicalSystem::addProjectToCustomer*](#) Add the Project p to the Customer c
- void [*addBillingToProject*](#) ([Billing](#) *b, [Project](#) *p)
[*HierarchicalSystem::addBillingToProject*](#) Add the Billing b to the Project p
- [QMap](#)< [Project](#) *, [Customer](#) > [*getCustomers*](#) () const
[*HierarchicalSystem::getCustomers*](#) Return all customers and these projects linked.
- [QMap](#)< [Billing](#) *, [Project](#) * > [*getProjects*](#) () const
[*HierarchicalSystem::getProjects*](#) Return all projects and these billing linked.

8.52.1 Detailed Description

The [HierarchicalSystem](#) class Create class which contains hierarchical system of FactDev.

Author

Florent Berbie

See Also

[Customer](#)
[Project](#)
[Billing](#)

8.52.2 Member Function Documentation

8.52.2.1 void Utils::HierarchicalSystem::addBillingToProject ([Billing](#) * b, [Project](#) * p)

[*HierarchicalSystem::addBillingToProject*](#) Add the Billing b to the Project p

Parameters

<i>b</i>	Billing
<i>p</i>	Project

8.52.2.2 void Utils::HierarchicalSystem::addProjectToCustomer ([Project](#) * p, [Customer](#) c)

[*HierarchicalSystem::addProjectToCustomer*](#) Add the Project p to the Customer c

Parameters

<i>p</i>	Project
<i>c</i>	Customer

8.52.2.3 [QMap](#)< [Project](#) *, [Customer](#) > Utils::HierarchicalSystem::getCustomers () const

[*HierarchicalSystem::getCustomers*](#) Return all customers and these projects linked.

Returns

Projects linked to Customers

8.52.2.4 QMap< Billing *, Project * > Utils::HierarchicalSystem::getProjects (void) const

[HierarchicalSystem::getProjects](#) Return all projects and these billing linked.

Returns

Billing linked to Projects

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

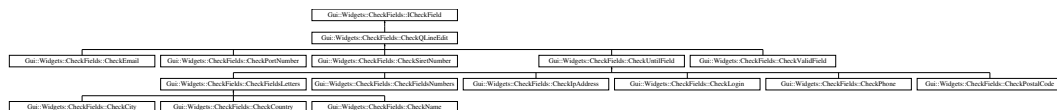
- /home/travis/build/FACT-Team/FactDev/src/utls/hierarchicalsistem.h
- /home/travis/build/FACT-Team/FactDev/src/utls/hierarchicalsistem.cpp

8.53 Gui::Widgets::CheckFields::ICheckField Class Reference

The [ICheckField](#) class Interface to check fields validity.

```
#include <icheckfield.h>
```

Inheritance diagram for Gui::Widgets::CheckFields::ICheckField:



Public Member Functions

- virtual bool [check](#) (QString text)=0
ICheckField::check Check if the field (line edit) is valid Return TRUE if the field is valid, else FALSE.

8.53.1 Detailed Description

The [ICheckField](#) class Interface to check fields validity.

8.53.2 Member Function Documentation

8.53.2.1 virtual bool Gui::Widgets::CheckFields::ICheckField::check (QString text) [pure virtual]

[ICheckField::check](#) Check if the field (line edit) is valid Return TRUE if the field is valid, else FALSE.

Returns

boolean Validity of the text

Implemented in [Gui::Widgets::CheckFields::CheckEmail](#), [Gui::Widgets::CheckFields::CheckPortNumber](#), [Gui::Widgets::CheckFields::CheckFieldsNumbers](#), [Gui::Widgets::CheckFields::CheckIpAddress](#), [Gui::Widgets::CheckFields::CheckLogin](#), [Gui::Widgets::CheckFields::CheckPhone](#), [Gui::Widgets::CheckFields::CheckPostalCode](#), [Gui::Widgets::CheckFields::CheckSiretNumber](#), [Gui::Widgets::CheckFields::CheckUntilField](#), [Gui::Widgets::CheckFields::CheckFieldsLetters](#), and [Gui::Widgets::CheckFields::CheckValidField](#).

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

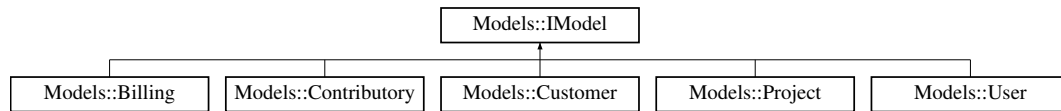
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/checkfields/icheckfield.h

8.54 Models::IModel Class Reference

The `IModel` class.

```
#include <imodel.h>
```

Inheritance diagram for Models::IModel:



Public Member Functions

- virtual `~IModel()`
~IModel Remove an instance of IModel
- virtual void `commit()`=0
IModel::commit Update or insert data into the database.
- virtual void `hydrat(int id)`=0
IModel::hydrat Get data of the element which is specified by the identify id from the database.
- virtual void `remove()`=0
IModel::remove Remove the current element in the database.
- virtual QVariantHash `getDataMap()`=0
getDataMap Get all data of model with a HashMap key/value
- int `getId()` const
IModel::getId Return the identify of the element of the database.
- void `setId(int id)`
IModel::setId Replace the current identify by id
- bool `isToRemoved()` const
toRemoved return if object must be removed.
- void `setToRemoved(bool toRemoved)`
setToRemoved Change the flag for removed object

Protected Attributes

- int `_id`
Element identify.
- bool `_toRemoved`
Flag to know if the object must be removed.

8.54.1 Detailed Description

The `IModel` class.

Author

Antoine de Roquemaurel

8.54.2 Member Function Documentation

8.54.2.1 `virtual QVariantHash Models::IModel::getDataMap () [pure virtual]`

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implemented in [Models::Billing](#), [Models::Project](#), [Models::User](#), [Models::Contributory](#), and [Models::Customer](#).

8.54.2.2 `int Models::IModel::getId () const [inline]`

[IModel::getId](#) Return the identify of the element of the database.

Returns

identity

8.54.2.3 `virtual void Models::IModel::hydrat (int id) [pure virtual]`

[IModel::hydrat](#) Get data of the element which is specified by the identify *id* from the database.

Parameters

<i>id</i>	
-----------	--

Implemented in [Models::Billing](#), [Models::Project](#), [Models::User](#), [Models::Contributory](#), and [Models::Customer](#).

8.54.2.4 `bool Models::IModel::isToRemoved () const [inline]`

toRemoved return if object must be removed.

Returns

boolean

8.54.2.5 `void Models::IModel::setId (int id) [inline]`

[IModel::setId](#) Replace the current identify by *id*

Parameters

<i>id</i>	New identify
-----------	--------------

8.54.2.6 `void Models::IModel::setToRemoved (bool toRemoved) [inline]`

setToRemoved Change the flag for removed object

Parameters

<i>toRemoved</i>	The new flag
------------------	--------------

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

- /home/travis/build/FACT-Team/FactDev/src/models/imodel.h

8.55 Utils::ItemType Class Reference

The [ItemType](#) class Item type model.

```
#include <itemtype.h>
```

Public Member Functions

- [ItemType](#) (int type, QString name)
ItemType::ItemType Construct an Item type.
- QString [getName](#) () const
ItemType::getName Get item name.
- [Models::IModel](#) * [getModel](#) (int id)
ItemType::getModel Get the databasemodel of the [ItemType](#) according to this identity id
- void [setName](#) (const QString &name)
ItemType::setName Modify the item name.
- int [getType](#) () const
ItemType::getType Get the type of the current item.
- void [setType](#) (int type)
ItemType::setType Modify the type of the current item.

Static Public Attributes

- static const int [CUSTOMER](#) = 0
constant value assigned to Customer
- static const int [PROJECT](#) = 1
constant value assigned to Project
- static const int [BILLING](#) = 2
constant value assigned to Billing
- static const int [QUOTE](#) = 3
constant value assigned to Quote

8.55.1 Detailed Description

The [ItemType](#) class Item type model.

8.55.2 Constructor & Destructor Documentation

8.55.2.1 Utils::ItemType::ItemType (int type, QString name)

[ItemType::ItemType](#) Construct an Item type.

Parameters

<i>type</i>	Type of the item
<i>name</i>	Name of the item

8.55.3 Member Function Documentation

8.55.3.1 Models::IModel * Utils::ItemType::getModel (int *id*)

[ItemType::getModel](#) Get the databasemodel of the [ItemType](#) according to this identity *id*

Parameters

<i>id</i>	Item type identity
-----------	--------------------

Returns

database model

8.55.3.2 QString Utils::ItemType::getName () const

[ItemType::getName](#) Get item name.

Returns

item name

8.55.3.3 int Utils::ItemType::getType () const

[ItemType::getType](#) Get the type of the current item.

Returns

type of the current item

8.55.3.4 void Utils::ItemType::setName (const QString & *name*)

[ItemType::setName](#) Modify the item name.

Parameters

<i>name</i>	New Item name
-------------	---------------

8.55.3.5 void Utils::ItemType::setType (int *type*)

[ItemType::setType](#) Modify the type of the current item.

Parameters

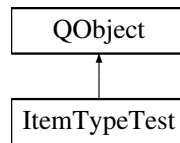
<i>type</i>	New item type
-------------	---------------

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

- /home/travis/build/FACT-Team/FactDev/src/utils/itemtype.h
- /home/travis/build/FACT-Team/FactDev/src/utils/itemtype.cpp

8.56 ItemTypeTest Class Reference

Inheritance diagram for ItemTypeTest:



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

- /home/travis/build/FACT-Team/FactDev/tests/utils/itemtypetest.h
- /home/travis/build/FACT-Team/FactDev/tests/utils/itemtypetest.cpp

8.57 Utils::Log Class Reference

The [Log](#) class for Simple management of log.

```
#include <log.h>
```

Public Member Functions

- [~Log](#) ()
Log::~~Log.
- void [write](#) (const QString text)
Log::write. Write log message in file.
- [Log](#) ()
Log::Log. Log is a singleton.

Static Public Member Functions

- static [Log](#) & [instance](#) (TypeLog type=INFO)
Log::instance. Return the instance of logger.

Friends

- [Log](#) & [operator<<](#) ([Log](#) &logger, const QString &text)
operator << for log writing

8.57.1 Detailed Description

The [Log](#) class for Simple management of log.

8.57.2 Member Function Documentation

8.57.2.1 [Log](#) & [Utils::Log::instance](#) (TypeLog *type* = INFO) [static]

[Log::instance](#). Return the instance of logger.

Parameters

<i>type</i>	Type of log : WARNING, INFO, ERROR
-------------	------------------------------------

Returns

Instance of logger.

8.57.2.2 void Utils::Log::write (const QString text)

[Log::write](#). Write log message in file.

Parameters

<i>text</i>	
-------------	--

8.57.3 Friends And Related Function Documentation

8.57.3.1 Log& operator<< (Log & logger, const QString & text) [friend]

operator << for log writing

Parameters

<i>logger</i>	Instance of Logger
<i>text</i>	Text to write

Returns

New logger.

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

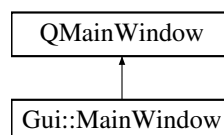
- /home/travis/build/FACT-Team/FactDev/src/utils/log.h
- /home/travis/build/FACT-Team/FactDev/src/utils/log.cpp

8.58 Gui::MainWindow Class Reference

The [MainWindow](#) class Main Window of the software.

```
#include <mainwindow.h>
```

Inheritance diagram for Gui::MainWindow:



Public Slots

- void [addCustomer](#) ()
[MainWindow::addCustomer](#) open window to add a new customer.
- void [editCustomer](#) ()

- MainWindow::editCustomer* open window to modify a customer.

 - void `removeCustomer` ()

MainWindow::removeCustomer open a popup to confirm the deletion of a customer, if ok remove the customer.
- void `archiveCustomer` ()

MainWindow::archiveCustomer open a pop-up to confirm the archiving of the customer, if ok archive the customer.
- void `addQuote` ()

MainWindow::addQuote open window to add a new quote.
- void `addBill` ()

MainWindow::addBill open window to add a new bill.
- void `billingsIsPaid` ()

MainWindow::billingsIsPaid Define the current billing as "paid".
- void `editUser` ()

MainWindow::editUser modify the user.
- void `search` (QString s)

MainWindow::search launch a new search.
- void `addProject` ()

MainWindow::addProject Create a new project for a customer.
- void `removeProject` (void)

MainWindow::removeProject Remove a project for a customer.
- void `editProject` (void)

MainWindow::editProject Modify the customer project.
- void `aboutQt` ()

MainWindow::aboutQt show Qt's details.
- void `aboutFact` ()

MainWindow::aboutFact show FACT's details (FACT team)
- void `aboutFactDev` ()

MainWindow::aboutFactDev() show FactDev's details (FactDev Software)
- void `aboutIcons` ()

MainWindow::aboutIcons() show icons's details.
- void `updateButtons` (void)

updateButton Update all button to disable or enabled its
- void `editDoc` ()

MainWindow::editDoc Edit the quote or bill of the project.
- void `removeDoc` ()

MainWindow::removeDoc Remove the quote or bill of the project.
- void `copyDoc` ()

MainWindow::copyDoc Copy all elements of a quote or a bill and Display these elements in a new quote or bill.
- void `openPdf` ()

MainWindow::openPdf Open the PDF file of the current Quote or Billing selected in the TableView.
- void `computeTurnover` ()

MainWindow::computeTurnover open window to allow computation of a period turnover.

Public Member Functions

- `MainWindow` (QWidget *parent=0)
- MainWindow*: Construct a window.
- int `getCurrentCustomerId` ()
- MainWindow::getCurrentCustomerId* get the selected customer.
- int `getCurrentProjectId` ()
- MainWindow::getCurrentProjectId* get the selected project id.

- int [getCurrentQuoteId](#) ()
MainWindow::getCurrentQuoteId get the selected quote id.
- QString [getCurrentCustomerName](#) ()
MainWindow::getCurrentCustomerName get the selected customer name in the customers' table.
- QString [getCurrentProjectName](#) ()
MainWindow::getCurrentProjectName get the selected project name in the table of projects.
- int [treeLevel](#) ()
MainWindow::treeLevel return the level of the node selected in the tree.
- QModelIndex [rootTree](#) ()
MainWindow::rootTree return the root of the tree "Tous les clients".
- void [addDoc](#) (bool isBilling)
MainWindow::addDoc open window to add a new document.
- void [resizeEvent](#) (QResizeEvent *event)
MainWindow::resizeEvent Resize central TableView when you resize the **MainWindow**
- void [responsiveCustomerTable](#) ()
MainWindow::responsiveCustomerTable Resize the Customer TableView according it resolution.
- void [responsiveProjectTable](#) ()
MainWindow::responsiveProjectTable Resize the Project TableView according it resolution.
- void [responsiveBillingTable](#) ()
MainWindow::responsiveBillingTable Resize the Billing TableView according it resolution.

8.58.1 Detailed Description

The [MainWindow](#) class Main Window of the software.

Author

Everybody

8.58.2 Constructor & Destructor Documentation

8.58.2.1 [Gui::MainWindow::MainWindow](#) (QWidget * *parent* = 0) [explicit]

[MainWindow](#): Construct a window.

Parameters

<i>parent</i>	
---------------	--

8.58.3 Member Function Documentation

8.58.3.1 void [Gui::MainWindow::addBill](#) () [slot]

[MainWindow::addBill](#) open window to add a new bill.

See Also

AddQuoteDialog

8.58.3.2 void [Gui::MainWindow::addDoc](#) (bool *isBilling*)

[MainWindow::addDoc](#) open window to add a new document.

Parameters

<i>bool</i>	quote or bill
-------------	---------------

See Also

[addBill](#) [addQuote](#)

8.58.3.3 void Gui::MainWindow::addProject () [slot]

[MainWindow::addProject](#) Create a new project for a customer.

See Also

AddProjectDialog

8.58.3.4 void Gui::MainWindow::addQuote () [slot]

[MainWindow::addQuote](#) open window to add a new quote.

See Also

AddQuoteDialog

8.58.3.5 void Gui::MainWindow::editUser () [slot]

[MainWindow::editUser](#) modify the user.

See Also

UserDataDialog

8.58.3.6 int Gui::MainWindow::getCurrentCustomerId ()

[MainWindow::getCurrentCustomerId](#) get the selected customer.

Returns

id of the selected customer

8.58.3.7 QString Gui::MainWindow::getCurrentCustomerName ()

[MainWindow::getCurrentCustomerName](#) get the selected customer name in the customers' table.

Returns

name of the selected customer

8.58.3.8 int Gui::MainWindow::getCurrentProjectId ()

[MainWindow::getCurrentProjectId](#) get the selected project id.

Returns

id of the selected project

8.58.3.9 QString Gui::MainWindow::getCurrentProjectName ()

[MainWindow::getCurrentProjectName](#) get the selected project name in the table of projects.

Returns

name of the selected project

8.58.3.10 int Gui::MainWindow::getCurrentQuoteId ()

[MainWindow::getCurrentQuoteId](#) get the selected quote id.

Returns

id of the selected quote

8.58.3.11 void Gui::MainWindow::resizeEvent (QResizeEvent * *event*)

[MainWindow::resizeEvent](#) Resize central TableView when you resize the [MainWindow](#)

Parameters

<i>event</i>	Resize event
--------------	--------------

8.58.3.12 QModelIndex Gui::MainWindow::rootTree ()

[MainWindow::rootTree](#) return the root of the tree "Tous les clients".

Returns

QModelIndex

8.58.3.13 void Gui::MainWindow::search (QString *s*) [slot]

[MainWindow::search](#) launch a new search.

Parameters

<i>s</i>	text in field
----------	---------------

8.58.3.14 int Gui::MainWindow::treeLevel ()

[MainWindow::treeLevel](#) return the level of the node selected in the tree.

Returns

integer, depth of the item in tree

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

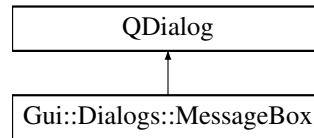
- /home/travis/build/FACT-Team/FactDev/src/gui/mainwindow/mainwindow.h
- /home/travis/build/FACT-Team/FactDev/src/gui/mainwindow/mainwindow.cpp

8.59 Gui::Dialogs::MessageBox Class Reference

The [MessageBox](#) class Information window with message.

```
#include <messagebox.h>
```

Inheritance diagram for Gui::Dialogs::MessageBox:



Public Member Functions

- [MessageBox](#) (QWidget *parent=0)
[MessageBox::MessageBox](#) Construt a [MessageBox](#)
- void [aboutFact](#) ()
[MessageBox::aboutFact](#) Defines FACT team information.
- void [aboutFactDev](#) ()
[MessageBox::aboutFactDev](#) Defines FactDev software information.
- void [aboutIcons](#) ()
[MessageBox::aboutIcons](#) Defines icons theme information.
- void [setImage](#) (QString img, int width=128, int height=128)
[MessageBox::setImage](#) Add the icon img to the current window.
- void [setText](#) (QString txt)
[MessageBox::setText](#) Add the text txt to the current window.

Static Public Member Functions

- static void [showAboutFact](#) ()
[MessageBox::showAboutFact](#) Shows window about FACT team.
- static void [showAboutFactDev](#) ()
[MessageBox::showAboutFactDev](#) Shows window about FactDev software.
- static void [showAboutIcons](#) ()
[MessageBox::showAboutIcons](#) Shows about icons theme of FactDev software.

8.59.1 Detailed Description

The [MessageBox](#) class Information window with message.

Author

Florent Berbie

8.59.2 Constructor & Destructor Documentation

8.59.2.1 [Gui::Dialogs::MessageBox::MessageBox \(QWidget * parent = 0 \)](#) [explicit]

[MessageBox::MessageBox](#) Construt a [MessageBox](#)

Parameters

<i>parent</i>	
---------------	--

8.59.3 Member Function Documentation

8.59.3.1 void Gui::Dialogs::MessageBox::setImage (QString *img*, int *width* = 128, int *height* = 128)

[MessageBox::setImage](#) Add the icon *img* to the current window.

Parameters

<i>img</i>	Icon
<i>width</i>	Icon width (default: 128)
<i>height</i>	Icon height (default: 128)

8.59.3.2 void Gui::Dialogs::MessageBox::setText (QString *txt*)

[MessageBox::setText](#) Add the text *txt* to the current window.

Parameters

<i>txt</i>	Text inside the current window
------------	--------------------------------

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

- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/messagebox.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/messagebox.cpp

8.60 Parameters Class Reference

The [Parameters](#) class Class for simple user parameters.

```
#include <parameters.h>
```

Static Public Attributes

- static const QString [DB_FILENAME](#) = "database.db"
DB_FILENAME The database sqlite file name.
- static const double [VERSION](#) = 1.0
VERSION Version number of software.

8.60.1 Detailed Description

The [Parameters](#) class Class for simple user parameters.

Author

Antoine de Roquemaurel

8.60.2 Member Data Documentation

8.60.2.1 const QString Parameters::DB_FILENAME = "database.db" [static]

DB_FILENAME The database sqlite file name.

Database file name

8.60.2.2 `const double Parameters::VERSION = 1.0` `[static]`

VERSION Version number of software.

Application version

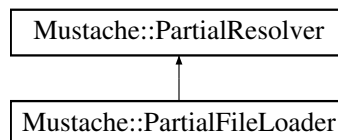
The documentation for this class was generated from the following files:

- `/home/travis/build/FACT-Team/FactDev/src/parameters.h`
- `/home/travis/build/FACT-Team/FactDev/src/parameters.cpp`

8.61 Mustache::PartialFileLoader Class Reference

```
#include <mustache.h>
```

Inheritance diagram for Mustache::PartialFileLoader:



Public Member Functions

- **PartialFileLoader** (const QString &basePath)
- virtual QString [getPartial](#) (const QString &name)

8.61.1 Detailed Description

A partial fetcher when loads templates from '`<name>.mustache`' files in a given directory.

Once a partial has been loaded, it is cached for future use.

8.61.2 Member Function Documentation

8.61.2.1 `QString PartialFileLoader::getPartial (const QString & name)` `[virtual]`

Returns the partial template with a given `name`.

Implements [Mustache::PartialResolver](#).

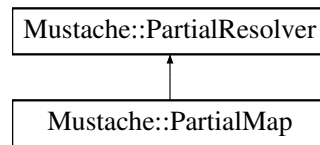
The documentation for this class was generated from the following files:

- `/home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h`
- `/home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.cpp`

8.62 Mustache::PartialMap Class Reference

```
#include <mustache.h>
```

Inheritance diagram for Mustache::PartialMap:



Public Member Functions

- **PartialMap** (const QHash< QString, QString > &partials)
- virtual QString [getPartial](#) (const QString &name)

8.62.1 Detailed Description

A simple partial fetcher which returns templates from a map of (partial name -> template)

8.62.2 Member Function Documentation

8.62.2.1 QString PartialMap::getPartial (const QString & name) [virtual]

Returns the partial template with a given `name`.

Implements [Mustache::PartialResolver](#).

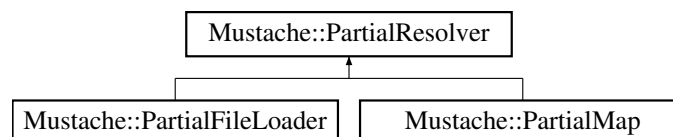
The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h
- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.cpp

8.63 Mustache::PartialResolver Class Reference

```
#include <mustache.h>
```

Inheritance diagram for Mustache::PartialResolver:



Public Member Functions

- virtual QString [getPartial](#) (const QString &name)=0

8.63.1 Detailed Description

Interface for fetching template partials.

8.63.2 Member Function Documentation

8.63.2.1 virtual QString Mustache::PartialResolver::getPartial (const QString & *name*) [pure virtual]

Returns the partial template with a given *name*.

Implemented in [Mustache::PartialFileLoader](#), and [Mustache::PartialMap](#).

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

- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h

8.64 Generator::PdfGenerator Class Reference

The [PdfGenerator](#) class Generator of PDF files.

```
#include <pdfgenerator.h>
```

Public Member Functions

- [PdfGenerator](#) (QString pdflatexPath="pdflatex")
PdfGenerator::PdfGenerator Construct a *PdfGenerator*.
- void [generate](#) (QString inputDir, QString filename)
PdfGenerator::generate Generate a PDF of the file named *filename* into the directory *inputDir*

8.64.1 Detailed Description

The [PdfGenerator](#) class Generator of PDF files.

8.64.2 Constructor & Destructor Documentation

8.64.2.1 Generator::PdfGenerator::PdfGenerator (QString *pdflatexPath* = "pdflatex")

[PdfGenerator::PdfGenerator](#) Construct a *PdfGenerator*.

Parameters

<i>pdflatexPath</i>	Path to the command to generate PDF file
---------------------	--

8.64.3 Member Function Documentation

8.64.3.1 void Generator::PdfGenerator::generate (QString *inputDir*, QString *filename*)

[PdfGenerator::generate](#) Generate a PDF of the file named *filename* into the directory *inputDir*

Parameters

<i>inputDir</i>	Directory where is store the PDF generated
<i>filename</i>	File name

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

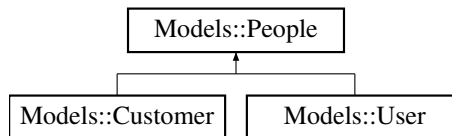
- /home/travis/build/FACT-Team/FactDev/src/generator/pdfgenerator.h
- /home/travis/build/FACT-Team/FactDev/src/generator/pdfgenerator.cpp

8.65 Models::People Class Reference

The `People` class `People`.

```
#include <people.h>
```

Inheritance diagram for Models::People:



Public Member Functions

- `People ()`
People::People Construct a *People*.
- `QString getFirstname () const`
People::getFirstname Return the *People* firstname.
- `void setFirstname (const QString &firstname)`
People::setFirstname Modify the *People* firstname
- `QString getLastname () const`
People::getLastname Return the *People* lastname.
- `void setLastname (const QString &lastname)`
People::setLastname Modify the *People* lastname
- `QString getCompany () const`
People::getCompany Return the *People* company.
- `void setCompany (const QString &company)`
People::setCompany Modify the *People* company name.
- `QString getAddress () const`
People::getAddress Return the company address (Number and name of street)
- `void setAddress (const QString &address)`
People::setAddress Modify the *People* company address
- `QString getPostalCode () const`
People::getPostalCode Return the postal code.
- `void setPostalCode (const QString &postalCode)`
People::setPostalCode Modify the postal code postalCode
- `QString getCity () const`
People::getCity Return the city.
- `void setCity (const QString &city)`
People::setCity Modify the city
- `QString getCountry () const`
People::getCountry Return the country of the *People*.
- `void setCountry (const QString &country)`
People::setCountry Replace the country of the *People* by country
- `QString getEmail () const`
People::getEmail Return the *People* professional email
- `void setEmail (const QString &email)`
People::setEmail Modify the *People* professional email
- `QString getPhone () const`
People::getPhone Return the number of the desktop phone.

- void [setPhone](#) (const QString &phone)
People::setPhone Modify the number of the desktop phone
- QString [getMobilePhone](#) () const
People::getMobilePhone Return the number of the professional mobile phone.
- void [setMobilePhone](#) (const QString &mobilePhone)
People::setMobilePhone Modify the number of the professional [People](#) mobile phone mobilePhone
- QString [getFax](#) () const
People::getFax Return the fax number.
- void [setFax](#) (const QString &fax)
People::setFax Replace the current fax number by fax
- bool [operator==](#) (const [People](#) &c)
People::operator == Re-define the operator "==" to compare if the current [People](#) is the same to the other [People](#) c
Return TRUE if both Peoples are the same, else FALSE.
- bool [operator!=](#) (const [People](#) &c)
People::operator != Re-define the operator "!=" to compare if the current [People](#) is differnt to the other [People](#) c
Return TRUE if both Peoples are different, else FALSE.

8.65.1 Detailed Description

The [People](#) class [People](#).

8.65.2 Member Function Documentation

8.65.2.1 QString Models::People::getAddress () const

[People::getAddress](#) Return the company address (Number and name of street)

Returns

Address company

8.65.2.2 QString Models::People::getCity () const

[People::getCity](#) Return the city.

Returns

city

8.65.2.3 QString Models::People::getCompany () const

[People::getCompany](#) Return the [People](#) company.

Returns

New company name

8.65.2.4 QString Models::People::getCountry () const

[People::getCountry](#) Return the country of the [People](#).

Returns

country of the [People](#)

8.65.2.5 QString Models::People::getEmail () const

[People::getEmail](#) Return the [People](#) professional *email*

Returns

professional email

8.65.2.6 QString Models::People::getFax () const

[People::getFax](#) Return the fax number.

Returns

fax number

8.65.2.7 QString Models::People::getFirstname () const

[People::getFirstname](#) Return the [People](#) firstname.

Returns

[People](#) firstname

8.65.2.8 QString Models::People::getLastname () const

[People::getLastname](#) Return the [People](#) lastname.

Returns

[People](#) lastname

8.65.2.9 QString Models::People::getMobilePhone () const

[People::getMobilePhone](#) Return the number of the professional mobile phone.

Returns

number of mobile phone

8.65.2.10 QString Models::People::getPhone () const

[People::getPhone](#) Return the number of the desktop phone.

Returns

number of the desktop phone

8.65.2.11 QString Models::People::getPostalCode () const

[People::getPostalCode](#) Return the postal code.

Returns

postal code

8.65.2.12 `bool Models::People::operator!=(const People & c)`

`People::operator ==` Re-define the operator "!=" to compare if the current `People` is different to the other `People` `c`
Return TRUE if both Peoples are different, else FALSE.

Parameters

<i>c</i>	People to compare
----------	-----------------------------------

Returns

boolean

8.65.2.13 bool Models::People::operator== (const [People](#) & *c*)

[People::operator ==](#) Re-define the operator "==" to compare if the current [People](#) is the same to the other [People](#) *c*
Return TRUE if both Peoples are the same, else FALSE.

Parameters

<i>c</i>	People to compare
----------	-----------------------------------

Returns

boolean

8.65.2.14 void Models::People::setAddress (const QString & *address*)

[People::setAddress](#) Modify the [People](#) company *address*

Parameters

<i>address</i>	Company address (name and number of street)
----------------	---

8.65.2.15 void Models::People::setCity (const QString & *city*)

[People::setCity](#) Modify the *city*

Parameters

<i>city</i>	Company city address
-------------	----------------------

8.65.2.16 void Models::People::setCompany (const QString & *company*)

[People::setCompany](#) Modify the [People](#) *company* name.

Parameters

<i>company</i>	New People company name
----------------	---

8.65.2.17 void Models::People::setCountry (const QString & *country*)

[People::setCountry](#) Replace the country of the [People](#) by *country*

Parameters

<i>country</i>	New country of the People
----------------	---

8.65.2.18 void Models::People::setEmail (const QString & *email*)

[People::setEmail](#) Modify the [People](#) professional *email*

Parameters

<i>email</i>	The People professional email
--------------	---

8.65.2.19 void Models::People::setFax (const QString & *fax*)

[People::setFax](#) Replace the current fax number by *fax*

Parameters

<i>fax</i>	new fax number
------------	----------------

8.65.2.20 void Models::People::setFirstname (const QString & *firstname*)

[People::setFirstname](#) Modify the [People](#) *firstname*

Parameters

<i>firstname</i>	New People firstname
------------------	--------------------------------------

8.65.2.21 void Models::People::setLastname (const QString & *lastname*)

[People::setLastname](#) Modify the [People](#) *lastname*

Parameters

<i>lastname</i>	New People lastname
-----------------	-------------------------------------

8.65.2.22 void Models::People::setMobilePhone (const QString & *mobilePhone*)

[People::setMobilePhone](#) Modify the number of the professional [People](#) mobile phone *mobilePhone*

Parameters

<i>mobilePhone</i>	Number of the professional mobile phone
--------------------	---

8.65.2.23 void Models::People::setPhone (const QString & *phone*)

[People::setPhone](#) Modify the number of the desktop *phone*

Parameters

<i>phone</i>	Number of the desktop phone
--------------	-----------------------------

8.65.2.24 void Models::People::setPostalCode (const QString & *postalCode*)

[People::setPostalCode](#) Modify the postal code *postalCode*

Parameters

<i>postalCode</i>	New postal code
-------------------	-----------------

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

- /home/travis/build/FACT-Team/FactDev/src/models/people.h
- /home/travis/build/FACT-Team/FactDev/src/models/people.cpp

8.66 Utils::pointers Class Reference

Static Public Member Functions

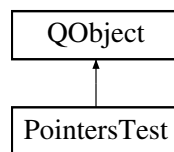
- static void **deleteIfNotNull** (QObject *p)

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

- /home/travis/build/FACT-Team/FactDev/src/utils/pointers.h
- /home/travis/build/FACT-Team/FactDev/src/utils/pointers.cpp

8.67 PointersTest Class Reference

Inheritance diagram for PointersTest:



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

- /home/travis/build/FACT-Team/FactDev/tests/utils/pointerstest.h
- /home/travis/build/FACT-Team/FactDev/tests/utils/pointerstest.cpp

8.68 Gui::Widgets::Popup Class Reference

Class for display popup quickly.

```
#include <popup.h>
```

Static Public Member Functions

- static void **toImplement** (QString, QWidget *)
Popup::toImplement Method to display a critical message : feature is not implemented now.

8.68.1 Detailed Description

Class for display popup quickly.

Author

Antoine de Roquemaurel

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

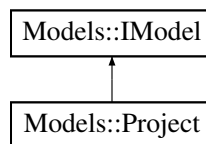
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/popup.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/popup.cpp

8.69 Models::Project Class Reference

The [Project](#) class : [Project](#) linked to a [Customer](#).

```
#include <project.h>
```

Inheritance diagram for Models::Project:



Public Member Functions

- [Project](#) ()
Project::Project Construct a [Project](#).
- [Project](#) (QString name)
Project::project Construct a project with a name.
- [Project](#) (int id)
Project::Project Construct a [Project](#) which is specified by an id
- virtual [~Project](#) ()
~Project Desctruct project object
- void [commit](#) ()
Project::commit Update project data in the database.
- void [hydrat](#) (int id)
Project::hydrat Insert project data which is specified by id in the database.
- void [remove](#) ()
Project::remove Remove the current project.
- QVariantHash [getDataMap](#) ()
getDataMap Get all data of model with a HashMap key/value
- QString [getName](#) () const
Project::getName Return the project name.
- void [setName](#) (const QString &name)
Project::setName Modify the project name
- QString [getDescription](#) () const
Project::getDescription Return a project description.
- void [setDescription](#) (const QString &description)
Project::setDescription Modify the project description
- QDate [getBeginDate](#) () const
Project::getBeginDate return the date of creation of the [Project](#)
- void [setBeginDate](#) (QDate beginDate)

- Project::setBeginDate* Modify beginDate of a **Project**
- QDate **getEndDate** () const
 - Project::getEndDate* Return the endDate of the **Project**
- void **setEndDate** (QDate endDate)
 - Project::setEndDate* Modify endDate of **Project**
- double **getCost** () const
 - Project::getCost* Return the **Project** cost
- void **setCost** (double cost)
 - Project::setCost* Modify the **Project** cost
- double **getDailyRate** () const
 - Project::getDailyRate* Return the daily rate estimated for this project.
- void **setDailyRate** (double dailyRate)
 - Project::setDailyRate* Modify the daily rate dailyRate of the current project.
- QSharedPointer< **Customer** > **getCustomer** () const
 - Project::getCustomer* Return the reference to the customer linked to this project.
- void **setCustomer** (QSharedPointer< **Customer** > customer)
 - Project::setCustomer* Modify the customer linked to this project.
- bool **operator==** (const **Project** &p)
 - Project::operator ==* Re-define the operator "==" to compare if the current project is the same to the other **Project** p
Return TRUE if both projects are the same, else FALSE.
- bool **operator<** (const **Project** &p) const
 - Project::operator <* defines the operator "<" to compare two **Project** and to see if the first is anterior to the second.
- bool **operator!=** (const **Project** &p)
 - Project::operator !=* Re-define the operator "!=" to compare if the current project is different to the other **Project** p
Return TRUE if both projects are different, else FALSE.
- double **getCost** ()
 - Project::costCompute* compute the **Project** cost

Additional Inherited Members

8.69.1 Detailed Description

The **Project** class : **Project** linked to a **Customer**.

Author

Florent Berbie

See Also

IModel

8.69.2 Constructor & Destructor Documentation

8.69.2.1 Models::Project::Project (int id)

Project::Project Construct a **Project** which is specified by an *id*

Parameters

<i>id</i>	
-----------	--

8.69.3 Member Function Documentation

8.69.3.1 QDate Models::Project::getBeginDate () const

[Project::getBeginDate](#) return the date of creation of the [Project](#)

Returns

the begin date of the [Project](#)

8.69.3.2 double Models::Project::getCost () const

[Project::getCost](#) Return the [Project](#) *cost*

Returns

the project cost

8.69.3.3 double Models::Project::getCost ()

[Project::costCompute](#) compute the [Project](#) *cost*

Returns

the project cost

8.69.3.4 QSharedPointer< Customer > Models::Project::getCustomer () const

[Project::getCustomer](#) Return the reference to the customer linked to this project.

Returns

customer linked to this project

8.69.3.5 double Models::Project::getDailyRate () const

[Project::getDailyRate](#) Return the daily rate estimated for this project.

Returns

the daily rate linket to the current project

8.69.3.6 QVariantHash Models::Project::getDataMap () [virtual]

[getDataMap](#) Get all data of model with a HashMap key/value

Returns

Model's data

Implements [Models::IModel](#).

8.69.3.7 QString Models::Project::getDescription () const

[Project::getDescription](#) Return a project description.

Returns

project description

8.69.3.8 QDate Models::Project::getEndDate () const

[Project::getEndDate](#) Return the *endDate* of the **Project**

Returns

the end date of the project

8.69.3.9 QString Models::Project::getName () const

[Project::getName](#) Return the project name.

Returns

project name

8.69.3.10 void Models::Project::hydrat (int *id*) [virtual]

[Project::hydrat](#) Insert project data which is specified by *id* in the database.

Parameters

<i>id</i>	Project identify
-----------	----------------------------------

Implements [Models::IModel](#).

8.69.3.11 bool Models::Project::operator!= (const Project & *p*)

[Project::operator ==](#) Re-define the operator "!=" to compare if the current project is differnt to the other **Project** *p*
Return TRUE if both projects are different, else FALSE.

Parameters

<i>c</i>	Project to compare
----------	------------------------------------

Returns

boolean

8.69.3.12 bool Models::Project::operator< (const Project & *p*) const

[Project::operator <](#) defines the operator "<" to compare two **Project** and to see if the first is anterior to the second.

Parameters

<i>b</i>	the Project to compare with the current Project
----------	---

Returns

true if the **Project** are different else false

8.69.3.13 `bool Models::Project::operator==(const Project & p)`

Project::operator == Re-define the operator "==" to compare if the current project is the same to the other **Project** *p*
Return TRUE if both projects are the same, else FALSE.

Parameters

<i>c</i>	Project to compare
----------	---------------------------

Returns

boolean

8.69.3.14 `void Models::Project::setBeginDate (QDate beginDate)`

Project::setBeginDate Modify *beginDate* of a **Project**

Parameters

<i>beginDate</i>	the new date of creation of the project
------------------	---

8.69.3.15 `void Models::Project::setCost (double cost)`

Project::setCost Modify the **Project** *cost*

Parameters

<i>cost</i>	the projectCost
-------------	-----------------

8.69.3.16 `void Models::Project::setCustomer (QSharedPointer< Customer > customer)`

Project::setCustomer Modify the *customer* linked to this project.

Parameters

<i>customer</i>	New customer associated to this project
-----------------	---

8.69.3.17 `void Models::Project::setDailyRate (double dailyRate)`

Project::setDailyRate Modify the daily rate *dailyRate* of the current project.

Parameters

<i>dailyRate</i>	New daily rate associated to the current project
------------------	--

8.69.3.18 `void Models::Project::setDescription (const QString & description)`

Project::setDescription Modify the project *description*

Parameters

<i>description</i>	New project description
--------------------	-------------------------

8.69.3.19 void Models::Project::setEndDate (QDate *endDate*)

[Project::setEndDate](#) Modify *endDate* of [Project](#)

Parameters

<i>endDate</i>	the new end date of the project
----------------	---------------------------------

8.69.3.20 void Models::Project::setName (const QString & *name*)

[Project::setName](#) Modify the project *name*

Parameters

<i>name</i>	Project name
-------------	------------------------------

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

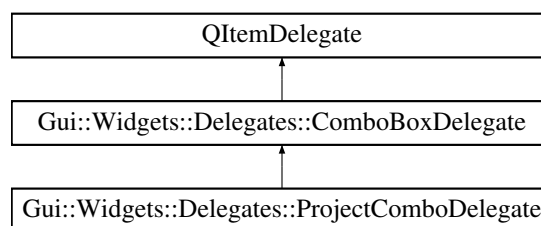
- /home/travis/build/FACT-Team/FactDev/src/models/project.h
- /home/travis/build/FACT-Team/FactDev/src/models/project.cpp

8.70 Gui::Widgets::Delegates::ProjectComboDelegate Class Reference

The [ProjectComboDelegate](#) class.

```
#include <projectcombodelegate.h>
```

Inheritance diagram for Gui::Widgets::Delegates::ProjectComboDelegate:



Public Member Functions

- [ProjectComboDelegate](#) (QSharedPointer< [Models::Customer](#) > c, QObject *parent=0)
[ProjectComboDelegate::ProjectComboDelegate](#).
- QWidget * [createEditor](#) (QWidget *parent, const QStyleOptionViewItem &option, const QModelIndex &index) const
[ProjectComboDelegate::createEditor](#) Return a *ComboBox* specified by *index* item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void [paint](#) (QPainter *painter, const QStyleOptionViewItem &option, const QModelIndex &index) const
[ProjectComboDelegate::paint](#) Renders the delegate using the given painter and style option for the item specified by *index*
- void [removeInCombo](#) (QList< int > &l)
[ProjectComboDelegate::removeInCombo](#) Remove the items contained in the list *l* into the current *ComboBox*.

- QMap< int, [Models::Project](#) > [getProjects](#) () const
[ProjectComboDelegate::getProjects](#) Return a list of projects linked to a Customer ID.
- bool [isLocked](#) () const
[ProjectComboDelegate::isLocked](#) Return TRUE if is locked else FALSE.
- void [setLocked](#) (bool locked)
[ProjectComboDelegate::setLocked](#) Change the state of the lock by a new value locked

8.70.1 Detailed Description

The [ProjectComboDelegate](#) class.

8.70.2 Constructor & Destructor Documentation

- 8.70.2.1 [Gui::Widgets::Delegates::ProjectComboDelegate::ProjectComboDelegate](#) ([QSharedPointer](#)< [Models::Customer](#) > *c*, [QObject](#) * *parent* = 0)

[ProjectComboDelegate::ProjectComboDelegate](#).

Parameters

<i>c</i>	
<i>parent</i>	

8.70.3 Member Function Documentation

- 8.70.3.1 [QWidget](#) * [Gui::Widgets::Delegates::ProjectComboDelegate::createEditor](#) ([QWidget](#) * *parent*, const [QStyleOptionViewItem](#) & *option*, const [QModelIndex](#) & *index*) const [virtual]

[ProjectComboDelegate::createEditor](#) Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Returns

ComboBox

Implements [Gui::Widgets::Delegates::ComboBoxDelegate](#).

- 8.70.3.2 [QMap](#)< int, [Models::Project](#) > [Gui::Widgets::Delegates::ProjectComboDelegate::getProjects](#) () const

[ProjectComboDelegate::getProjects](#) Return a list of projects linked to a Customer ID.

Returns

Projets of a Customer

8.70.3.3 `bool Gui::Widgets::Delegates::ProjectComboDelegate::isLocked () const`

[ProjectComboDelegate::isLocked](#) Return TRUE if is locked else FALSE.

Returns

boolean

8.70.3.4 `void Gui::Widgets::Delegates::ProjectComboDelegate::paint (QPainter * painter, const QStyleOptionViewItem & option, const QModelIndex & index) const` [virtual]

[ProjectComboDelegate::paint](#) Renders the delegate using the given *painter* and style *option* for the item specified by *index*

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Implements [Gui::Widgets::Delegates::ComboBoxDelegate](#).

8.70.3.5 `void Gui::Widgets::Delegates::ProjectComboDelegate::setLocked (bool locked)`

[ProjectComboDelegate::setLocked](#) Change the state of the lock by a new value *locked*

Parameters

<i>locked</i>

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

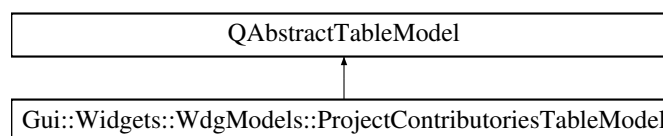
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/projectcombodelegate.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/projectcombodelegate.cpp

8.71 Gui::Widgets::WdgModels::ProjectContributoriesTableModel Class Reference

The [ProjectContributoriesTableModel](#) class Table model of contributories linked to projects.

```
#include <projectcontributoriesstablemodel.h>
```

Inheritance diagram for Gui::Widgets::WdgModels::ProjectContributoriesTableModel:



Public Member Functions

- [ProjectContributoriesTableModel](#) (QObject *parent=0)
[ProjectContributoriesTableModel::ProjectContributoriesTableModel](#) Construct a [ProjectContributoriesTableModel](#).
- int [rowCount](#) (const QModelIndex &) const
[ProjectContributoriesTableModel::rowCount](#) Number of contributories row.
- int [columnCount](#) (const QModelIndex &) const

- [*ProjectContributoriesTableModel::columnCount*](#) Number of column of a contributory.
QVariant [data](#) (const QModelIndex &index, int role) const
[*ProjectContributoriesTableModel::data*](#) Obtains data of a specify cell.
- QVariant [headerData](#) (int section, Qt::Orientation orientation, int role) const
[*ProjectContributoriesTableModel::headerData*](#) Obtains header title of table.
- bool [setData](#) (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)
[*ProjectContributoriesTableModel::setData*](#) Change data of a cell.
- Qt::ItemFlags [flags](#) (const QModelIndex &index) const
[*ProjectContributoriesTableModel::flags*](#) Differents table flags.
- void [append](#) (QPair< [Models::Project](#) *, [Models::Rate](#) > p)
[*ProjectContributoriesTableModel::append*](#) Add a new rate p to a Project.
- void [append](#) ()
[*ProjectContributoriesTableModel::append*](#) Add the current element to the list.
- bool [allProjectsChosen](#) ()
[*ProjectContributoriesTableModel::allProjectsChosen*](#) Return TRUE if all projects have been selected else FALSE.
- QList< int > & [getSelectedProjects](#) ()
[*ProjectContributoriesTableModel::getSelectedProjects*](#) Return the list of the Project id which are selected.
- void [remove](#) (int index)
[*ProjectContributoriesTableModel::remove*](#) Remove the element to the index
- QPair< [Models::Project](#) *, [Models::Rate](#) > [getProject](#) (const int row)
[*ProjectContributoriesTableModel::getProject*](#) Return the project and it rate linked which correspond to the row
- QList< QPair< [Models::Project](#) *, [Models::Rate](#) > > [getProjects](#) (void)
[*ProjectContributoriesTableModel::getProjects*](#) Return a list of the set of projects and their rates linked.

8.71.1 Detailed Description

The [ProjectContributoriesTableModel](#) class Table model of contributories linked to projects.

Author

Antoine de Roquemaurel

8.71.2 Constructor & Destructor Documentation

8.71.2.1 [Gui::Widgets::WdgModels::ProjectContributoriesTableModel::ProjectContributoriesTableModel \(QObject * parent = 0 \)](#)

[ProjectContributoriesTableModel::ProjectContributoriesTableModel](#) Construct a [ProjectContributoriesTableModel](#).

Parameters

<i>parent</i>	Parent widget
---------------	---------------

8.71.3 Member Function Documentation

8.71.3.1 [bool Gui::Widgets::WdgModels::ProjectContributoriesTableModel::allProjectsChosen \(\)](#)

[ProjectContributoriesTableModel::allProjectsChosen](#) Return TRUE if all projects have been selected else FALSE.

Returns

boolean All projects selected

8.71.3.2 void Gui::Widgets::WdgModels::ProjectContributoriesTableModel::append (QPair< Models::Project *,
Models::Rate > p)

[ProjectContributoriesTableModel::append](#) Add a new rate p to a Project.

Parameters

<i>p</i>	
----------	--

8.71.3.3 `int Gui::Widgets::WdgModels::ProjectContributoriesTableModel::columnCount (const QModelIndex &) const`

[ProjectContributoriesTableModel::columnCount](#) Number of column of a contributory.

Returns

The number of column

8.71.3.4 `QVariant Gui::Widgets::WdgModels::ProjectContributoriesTableModel::data (const QModelIndex & index, int role) const`

[ProjectContributoriesTableModel::data](#) Obtains data of a specify cell.

Parameters

<i>index</i>	The cell who we want data
<i>role</i>	The role of set

Returns

The data of cell

8.71.3.5 `Qt::ItemFlags Gui::Widgets::WdgModels::ProjectContributoriesTableModel::flags (const QModelIndex & index) const`

[ProjectContributoriesTableModel::flags](#) Differents table flags.

Parameters

<i>index</i>	The cell who we want to know flags
--------------	------------------------------------

Returns

Flags

8.71.3.6 `QPair< Models::Project *, Models::Rate > Gui::Widgets::WdgModels::ProjectContributoriesTableModel::getProject (const int row)`

[ProjectContributoriesTableModel::getProject](#) Return the project and it rate linked which correspond to the *row*

Parameters

<i>row</i>	Row of the project to get
------------	---------------------------

Returns

a Project and it rate

8.71.3.7 `QList< QPair< Models::Project *, Models::Rate > > Gui::Widgets::WdgModels::ProjectContributoriesTableModel::getProjects (void)`

[ProjectContributoriesTableModel::getProjects](#) Return a list of the set of projects and their rates linked.

Returns

List of Projects and Rates linked

8.71.3.8 `QList< int > & Gui::Widgets::WdgModels::ProjectContributoriesTableModel::getSelectedProjects ()`

[ProjectContributoriesTableModel::getSelectedProjects](#) Return the list of the Project id which are selected.

Returns

List of ID Project

8.71.3.9 `QVariant Gui::Widgets::WdgModels::ProjectContributoriesTableModel::headerData (int section, Qt::Orientation orientation, int role) const`

[ProjectContributoriesTableModel::headerData](#) Obtains header title of table.

Parameters

<i>section</i>	The number of column
<i>orientation</i>	The table orientation
<i>role</i>	

Returns

The Title header of column

8.71.3.10 `void Gui::Widgets::WdgModels::ProjectContributoriesTableModel::remove (int index)`

[ProjectContributoriesTableModel::remove](#) Remove the element to the *index*

Parameters

<i>index</i>	Index of the element to remove
--------------	--------------------------------

8.71.3.11 `int Gui::Widgets::WdgModels::ProjectContributoriesTableModel::rowCount (const QModelIndex &) const`

[ProjectContributoriesTableModel::rowCount](#) Number of contributories row.

Returns

The number of contributories

8.71.3.12 `bool Gui::Widgets::WdgModels::ProjectContributoriesTableModel::setData (const QModelIndex & index, const QVariant & value, int role = Qt::EditRole)`

[ProjectContributoriesTableModel::setData](#) Change data of a cell.

Parameters

<i>index</i>	The cell to change data
<i>value</i>	The new value
<i>role</i>	The role of cell

Returns

True if we could edit

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

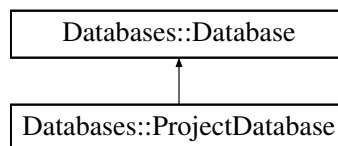
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetmodels/projectcontributoriestablemodel.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetmodels/projectcontributoriestablemodel.cpp

8.72 Databases::ProjectDatabase Class Reference

The [ProjectDatabase](#) class Project table database.

```
#include <projectdatabase.h>
```

Inheritance diagram for Databases::ProjectDatabase:



Public Member Functions

- [Models::Project](#) * [getProject](#) (const int pld)
ProjectDatabase::getProject Get informations about the project identified by 'pld'.
- [Models::Project](#) * [getProject](#) (QSqlQuery &q)
ProjectDatabase::getProject.
- int [addProject](#) (const [Models::Project](#) &)
ProjectDatabase::addProject Add the project 'pProject' to the database.
- void [updateProject](#) (const [Models::Project](#) &)
ProjectDatabase::updateProject Update informations about the project.
- void [removeProject](#) (const int pld)
removeProject Remove the project with the id 'pld'
- int [getNbProjects](#) ()
ProjectDatabase::getNbProjects Return the number of projects existing.
- int [getNbProjectsForACustomer](#) (const int pld)
ProjectDatabase::getNbProjectsForACustomer Return the number of projects existing for an identify customer pld
- QMap< int, [Models::Project](#) > [getProjectsOfCustomer](#) (QSharedPointer< [Models::Customer](#) > c)
getProjectsOfCustomer Return all projects of a customer
- QList< [Models::Project](#) > [getProjects](#) (const int customerId)
getProjects Return all projects of a customer
- [WdgModels::ProjectsTableModel](#) * [getProjectsTable](#) (const int pld) throw (DbException*)
getProjectsTable Return all projects of a customer in QStandardItemModel
- QSharedPointer< [Project](#) > [updateProject](#) (QSqlQuery &q)
getProject Obtain a project without new query

- `QList< Project * > getAllProjects ()`
`Project::getAllProjects` Return the list of projects.
- `double getCostProjects (QList< Project * > projects)`
`ProjectDatabase::getCostProjects` compute the cost of the project list given in parameter.

Static Public Member Functions

- `static ProjectDatabase * instance () throw (DbException*)`
`ProjectDatabase::getInstance` Return an instance of [ProjectDatabase](#).

Additional Inherited Members

8.72.1 Detailed Description

The [ProjectDatabase](#) class Project table database.

Author

Florent Berbie

See Also

[Database](#)
[Project](#)

8.72.2 Member Function Documentation

8.72.2.1 `int Databases::ProjectDatabase::addProject (const Models::Project & pProject)`

[ProjectDatabase::addProject](#) Add the project 'pProject' to the database.

Returns

project id

8.72.2.2 `QList< Project * > Databases::ProjectDatabase::getAllProjects ()`

`Project::getAllProjects` Return the list of projects.

Returns

Projects list

8.72.2.3 `double Databases::ProjectDatabase::getCostProjects (QList< Project * > projects)`

[ProjectDatabase::getCostProjects](#) compute the cost of the project list given in parameter.

Parameters

<i>projects</i>	the list of projects
-----------------	----------------------

Returns

the cost of the list given

8.72.2.4 `int Databases::ProjectDatabase::getNbProjects ()`

[ProjectDatabase::getNbProjects](#) Return the number of projects existing.

Returns

number of projects

8.72.2.5 `int Databases::ProjectDatabase::getNbProjectsForACustomer (const int pld)`

[ProjectDatabase::getNbProjectsForACustomer](#) Return the number of projects existing for an identify customer *pld*

Parameters

<i>pld</i>	Project id
------------	------------

Returns

number of projects

8.72.2.6 `Models::Project * Databases::ProjectDatabase::getProject (const int pld)`

[ProjectDatabase::getProject](#) Get informations about the project identified by 'pld'.

Parameters

<i>pld</i>	project
------------	---------

Returns

the project

8.72.2.7 `Models::Project * Databases::ProjectDatabase::getProject (QSqlQuery & q)`

[ProjectDatabase::getProject](#).

Parameters

<i>q</i>	
----------	--

Returns**8.72.2.8** `QList< Project > Databases::ProjectDatabase::getProjects (const int customerId)`

[getProjects](#) Return all projects of a customer

Parameters

<i>customerId</i>	The customer id
-------------------	-----------------

Returns

List with all projects of customer

8.72.2.9 `QMap< int, Models::Project > Databases::ProjectDatabase::getProjectsOfCustomer (QSharedPointer< Models::Customer > c)`

getProjectsOfCustomer Return all projects of a customer

Parameters

<i>c</i>	The customer
----------	--------------

Returns

All projects of *c* with id in key

8.72.2.10 WdgModels::ProjectsTableModel * Databases::ProjectDatabase::getProjectsTable (const int *pld*) throw DbException *)

getProjectsTable Return all projects of a customer in QStandardItemModel

Parameters

<i>filter</i>	Select only projects who are specified by <i>filter</i>
---------------	---

Returns

QStandardItemModel an item model for QTableView

8.72.2.11 ProjectDatabase * Databases::ProjectDatabase::instance () throw DbException *) [static]

ProjectDatabase::getInstance Return an instance of [ProjectDatabase](#).

Returns

Instance of [ProjectDatabase](#)

8.72.2.12 void Databases::ProjectDatabase::removeProject (const int *pld*)

removeProject Remove the project with the id 'pld'

Parameters

<i>pld</i>	project id
------------	------------

8.72.2.13 QSharedPointer< Models::Project > Databases::ProjectDatabase::updateProject (QSqlQuery & *q*)

getProject Obtain a project without new query

Parameters

<i>q</i>	The query to use
----------	------------------

Returns

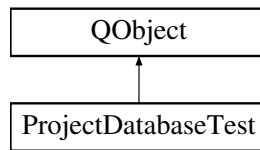
The project linked to *q*

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

- /home/travis/build/FACT-Team/FactDev/src/database/projectdatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/projectdatabase.cpp

8.73 ProjectDatabaseTest Class Reference

Inheritance diagram for ProjectDatabaseTest:

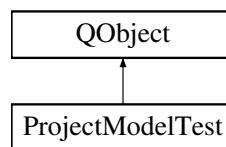


The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/tests/database/projectdatabasetest.h
- /home/travis/build/FACT-Team/FactDev/tests/database/projectdatabasetest.cpp

8.74 ProjectModelTest Class Reference

Inheritance diagram for ProjectModelTest:



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

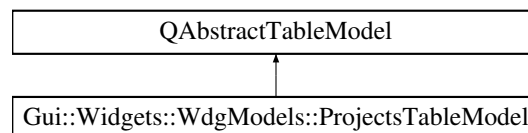
- /home/travis/build/FACT-Team/FactDev/tests/models/projectmodeltest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/projectmodeltest.cpp

8.75 Gui::Widgets::WdgModels::ProjectsTableModel Class Reference

The [ProjectsTableModel](#) class for a Project table.

```
#include <projectstablemodel.h>
```

Inheritance diagram for Gui::Widgets::WdgModels::ProjectsTableModel:



Public Member Functions

- [ProjectsTableModel](#) ()
ProjectsTableModel::ProjectsTableModel Construct a *ProjectsTableModel*.
- int [rowCount](#) (const QModelIndex &) const
ProjectsTableModel::rowCount Number of projects row.

- int [columnCount](#) (const QModelIndex &) const
ProjectsTableModel::columnCount Number of column of a Project.
- QVariant [data](#) (const QModelIndex &index, int role=Qt::DisplayRole) const
ProjectsTableModel::data Obtains data of a specify cell.
- QVariant [headerData](#) (int section, Qt::Orientation orientation, int role=Qt::DisplayRole) const
ProjectsTableModel::headerData Obtains header title of table.
- bool [setData](#) (const QModelIndex &index, const QVariant &value, int role=Qt::EditRole)
ProjectsTableModel::setData Change data of a cell.
- void [append](#) (const [Project](#) &project)
ProjectsTableModel::append Add a new line in table.
- void [remove](#) (const int i)
ProjectsTableModel::remove Remove a line.
- Qt::ItemFlags [flags](#) (const QModelIndex &index) const
ProjectsTableModel::flags Differents table flags.
- int [count](#) ()
ProjectsTableModel::count Number of projects in table.
- QList< [Project](#) > [getProjects](#) () const
ProjectsTableModel::getProjects Return the list of projects.

8.75.1 Detailed Description

The [ProjectsTableModel](#) class for a Project table.

Author

Florent Berbie

See Also

[Project](#)

8.75.2 Constructor & Destructor Documentation

8.75.2.1 Gui::Widgets::WdgModels::ProjectsTableModel::ProjectsTableModel ()

[ProjectsTableModel::ProjectsTableModel](#) Construct a [ProjectsTableModel](#).

Parameters

<i>parent</i>	Parent widget
---------------	---------------

8.75.3 Member Function Documentation

8.75.3.1 void Gui::Widgets::WdgModels::ProjectsTableModel::append (const [Project](#) & *project*)

[ProjectsTableModel::append](#) Add a new line in table.

Parameters

<i>Project</i>	The new Project
----------------	-----------------

8.75.3.2 `int Gui::Widgets::WdgModels::ProjectsTableModel::columnCount (const QModelIndex &) const`

[ProjectsTableModel::columnCount](#) Number of column of a Project.

Returns

The number of column

8.75.3.3 `int Gui::Widgets::WdgModels::ProjectsTableModel::count ()`

[ProjectsTableModel::count](#) Number of projects in table.

Returns

The number of projects

8.75.3.4 `QVariant Gui::Widgets::WdgModels::ProjectsTableModel::data (const QModelIndex & index, int role = Qt::DisplayRole) const`

[ProjectsTableModel::data](#) Obtains data of a specify cell.

Parameters

<i>index</i>	The cell who we want data
<i>role</i>	The role of set

Returns

The data of cell

8.75.3.5 `Qt::ItemFlags Gui::Widgets::WdgModels::ProjectsTableModel::flags (const QModelIndex & index) const`

[ProjectsTableModel::flags](#) Differents table flags.

Parameters

<i>index</i>	The cell who we want to know flags
--------------	------------------------------------

Returns

Flags

8.75.3.6 `QList< Project > Gui::Widgets::WdgModels::ProjectsTableModel::getProjects (void) const`

[ProjectsTableModel::getProjects](#) Return the list of projects.

Returns

list of projects

8.75.3.7 `QVariant Gui::Widgets::WdgModels::ProjectsTableModel::headerData (int section, Qt::Orientation orientation, int role = Qt::DisplayRole) const`

[ProjectsTableModel::headerData](#) Obtains header title of table.

Parameters

<i>section</i>	The number of column
<i>orientation</i>	The table orientation
<i>role</i>	

Returns

The Title header of column

8.75.3.8 void Gui::Widgets::WdgModels::ProjectsTableModel::remove (const int *i*)

[ProjectsTableModel::remove](#) Remove a line.

Parameters

<i>i</i>	The number of line to remove
----------	------------------------------

8.75.3.9 int Gui::Widgets::WdgModels::ProjectsTableModel::rowCount (const QModelIndex &) const

[ProjectsTableModel::rowCount](#) Number of projects row.

Returns

The number of projects

8.75.3.10 bool Gui::Widgets::WdgModels::ProjectsTableModel::setData (const QModelIndex & *index*, const QVariant & *value*, int *role* = Qt::EditRole)

[ProjectsTableModel::setData](#) Change data of a cell.

Parameters

<i>index</i>	The cell to change data
<i>value</i>	The new value
<i>role</i>	The role of cell

Returns

True if we could edit

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

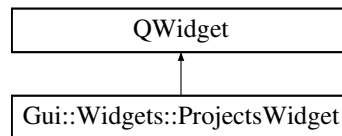
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetmodels/projectstablemodel.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/widgetmodels/projectstablemodel.cpp

8.76 Gui::Widgets::ProjectsWidget Class Reference

The [ProjectsWidget](#) class Actions on Project.

```
#include <projectswidget.h>
```

Inheritance diagram for Gui::Widgets::ProjectsWidget:



Public Slots

- void [newProject](#) ()
[ProjectsWidget::addProject](#) Event which sends a signal to add a new project.
- void [editSelectedProject](#) ()
[ProjectsWidget::editSelectedProject](#) Event which sends a signal to edit the project selected.
- void [removeSelectedProject](#) ()
[ProjectsWidget::removeSelectedProject](#) Event which sends a signal to remove the project selected.
- void [updateBtn](#) (bool b)
[ProjectsWidget::updateBtn](#) Update the toolbar in tblProjects.

Signals

- void [addProject](#) ()
[ProjectsWidget::addProject](#) Add a new project to the current Customer.
- void [editProject](#) ()
[ProjectsWidget::editProject](#) Edit the current Customer selected.
- void [removeProject](#) ()
[ProjectsWidget::removeProject](#) Remove the current Customer selected.

Public Member Functions

- [ProjectsWidget](#) (QWidget *parent=0)
[ProjectsWidget::ProjectsWidget](#) Construct a [ProjectsWidget](#).

8.76.1 Detailed Description

The [ProjectsWidget](#) class Actions on Project.

Author

Florent Berbie

8.76.2 Constructor & Destructor Documentation

8.76.2.1 [Gui::Widgets::ProjectsWidget::ProjectsWidget](#) ([QWidget](#) * *parent* = 0) [explicit]

[ProjectsWidget::ProjectsWidget](#) Construct a [ProjectsWidget](#).

Parameters

<i>parent</i>	
---------------	--

8.76.3 Member Function Documentation

8.76.3.1 void Gui::Widgets::ProjectsWidget::updateBtn (bool *b*) [slot]

[ProjectsWidget::updateBtn](#) Update the toolbar in tblProjects.

Parameters

<i>boolean</i>	if a row is selected
----------------	----------------------

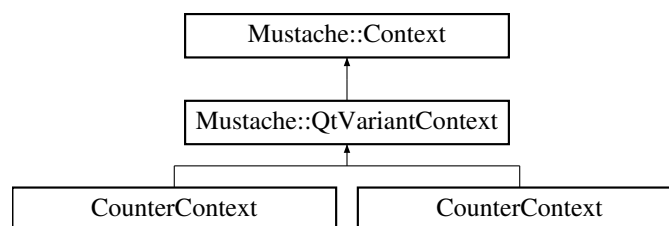
The documentation for this class was generated from the following files:

- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/projectswidget.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/projectswidget.cpp

8.77 Mustache::QtVariantContext Class Reference

```
#include <mustache.h>
```

Inheritance diagram for Mustache::QtVariantContext:



Public Types

- typedef QString(* [fn_t](#))(const QString &, [Mustache::Renderer](#) *, [Mustache::Context](#) *)

Public Member Functions

- **QtVariantContext** (const QVariant &root, [PartialResolver](#) *resolver=0)
- virtual QString [stringValue](#) (const QString &key) const
- virtual bool [isFalse](#) (const QString &key) const
- virtual int [listCount](#) (const QString &key) const
- virtual void [push](#) (const QString &key, int index=-1)
- virtual void [pop](#) ()
- virtual bool [canEval](#) (const QString &key) const
- virtual QString [eval](#) (const QString &key, const QString &_template, [Mustache::Renderer](#) *renderer)

8.77.1 Detailed Description

A context implementation which wraps a QVariantHash or QVariantMap.

8.77.2 Member Typedef Documentation

8.77.2.1 typedef QString(* [Mustache::QtVariantContext::fn_t](#))(const QString &, [Mustache::Renderer](#) *, [Mustache::Context](#) *)

Construct a [QtVariantContext](#) which wraps a dictionary in a QVariantHash or a QVariantMap.

8.77.3 Member Function Documentation

8.77.3.1 `bool QtVariantContext::canEval (const QString & key) const [virtual]`

Returns true if `eval()` should be used to render section tags using `key`. If `canEval()` returns true for a key, the renderer will pass the literal, unrendered block of text for the section to `eval()` and replace the section with the result.

`canEval()` and `eval()` are equivalents for callable objects (eg. lambdas) in other Mustache implementations.

The default implementation always returns false.

Reimplemented from [Mustache::Context](#).

Reimplemented in [CounterContext](#), and [CounterContext](#).

8.77.3.2 `QString QtVariantContext::eval (const QString & key, const QString & _template, Mustache::Renderer * renderer) [virtual]`

Callback used to render a template section with the given `key`. `renderer` will substitute the original section tag with the result of `eval()`.

The default implementation returns an empty string.

Reimplemented from [Mustache::Context](#).

Reimplemented in [CounterContext](#), and [CounterContext](#).

8.77.3.3 `bool QtVariantContext::isFalse (const QString & key) const [virtual]`

Returns true if the value for `key` is 'false' or an empty list. 'False' values typically include empty strings, the boolean value false etc.

When processing a section Mustache tag, the section is not rendered if the key is false, or for an inverted section tag, the section is only rendered if the key is false.

Implements [Mustache::Context](#).

8.77.3.4 `int QtVariantContext::listCount (const QString & key) const [virtual]`

Returns the number of items in the list value for `key` or 0 if the value for `key` is not a list.

Implements [Mustache::Context](#).

8.77.3.5 `void QtVariantContext::pop () [virtual]`

Exit the current context.

Implements [Mustache::Context](#).

8.77.3.6 `void QtVariantContext::push (const QString & key, int index = -1) [virtual]`

Set the current context to the value for `key`. If `index` is ≥ 0 , set the current context to the `index`'th value in the list value for `key`.

Implements [Mustache::Context](#).

8.77.3.7 `QString QtVariantContext::stringValue (const QString & key) const [virtual]`

Returns a string representation of the value for `key` in the current context. This is used to replace a Mustache value tag.

Implements [Mustache::Context](#).

Reimplemented in [CounterContext](#), and [CounterContext](#).

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

- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h
- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.cpp

8.78 Models::Rate Class Reference

The [Rate](#) class [Rate](#) of a prestation.

```
#include <rate.h>
```

Public Member Functions

- [Rate](#) ()
[Rate::Rate](#) Construct a [Rate](#).
- [Rate](#) (double hourly)
[Rate::Rate](#) Construct a [Rate](#) with an hourly rate.
- [Rate](#) (int idBilling, int idProject)
[Rate::Rate](#) Construct a [Rate](#) for the [Billing](#) with idBilling as identify and from the project identified by idProject
- void [setDailyRate](#) (const double &)
[Rate::setDailyRate](#) Modify the daily rate with the new value.
- void [setHourlyRate](#) (const double &)
[Rate::setHourlyRate](#) Modify the hourly rate with the new value.
- double [getDailyRate](#) () const
[Rate::getDailyRate](#) Return the daily rate.
- double [getHourlyRate](#) () const
[Rate::getHourlyRate](#) Return the hourly rate.
- int [getNbDailyHours](#) () const
[Rate::getNbDailyHours](#) Return the number of daily hours.
- double [getLegalRate](#) () const
[Rate::getLegalRate](#) Return the legal rate.

8.78.1 Detailed Description

The [Rate](#) class [Rate](#) of a prestation.

Author

Antoine de Roquemaurel

8.78.2 Constructor & Destructor Documentation

8.78.2.1 Models::Rate::Rate (double hourly)

[Rate::Rate](#) Construct a [Rate](#) with an hourly rate.

Parameters

<i>hourly</i>	Hourly rate
---------------	-------------

8.78.2.2 Models::Rate::Rate (int *idBilling*, int *idProject*)

[Rate::Rate](#) Construct a [Rate](#) for the [Billing](#) with *idBilling* as identify and from the project identified by *idProject*

Parameters

<i>idBilling</i>	Billing identify
<i>idProject</i>	Project identify

8.78.3 Member Function Documentation

8.78.3.1 double Models::Rate::getDailyRate () const

[Rate::getDailyRate](#) Return the daily rate.

Returns

8.78.3.2 double Models::Rate::getHourlyRate () const

[Rate::getHourlyRate](#) Return the hourly rate.

Returns

8.78.3.3 double Models::Rate::getLegalRate () const

[Rate::getLegalRate](#) Return the legal rate.

Returns

8.78.3.4 int Models::Rate::getNbDailyHours () const

[Rate::getNbDailyHours](#) Return the number of daily hours.

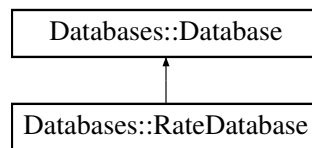
Returns

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

- /home/travis/build/FACT-Team/FactDev/src/models/rate.h
- /home/travis/build/FACT-Team/FactDev/src/models/rate.cpp

8.79 Databases::RateDatabase Class Reference

Inheritance diagram for Databases::RateDatabase:



Public Member Functions

- void [addRateProject](#) (int idProject, int idBilling, double hourlyRate)
RateDatabase::addRateProject Add a new rate *hourlyRate* to the Project identified by *idProject* of the Billing with *idBilling* as identify.
- double [getRate](#) (const int idBilling, const int idProject) const
RateDatabase::getRate Return the rate of the Project identified by *idProject* of the Billing with *idBilling* as identify.
- void [updateRateProject](#) (int idProject, int idBilling, double hourlyRate)
RateDatabase::updateRateProject Update the current rate by the new *hourlyRate* of the Project identified by *idProject* of the Billing with *idBilling* as identify.

Static Public Member Functions

- static [RateDatabase](#) * [instance](#) () throw (DbException*)
RateDatabase::getInstance Return an instance of [RateDatabase](#).

Additional Inherited Members

8.79.1 Member Function Documentation

8.79.1.1 void Databases::RateDatabase::addRateProject (int *idProject*, int *idBilling*, double *hourlyRate*)

[RateDatabase::addRateProject](#) Add a new rate *hourlyRate* to the Project identified by *idProject* of the Billing with *idBilling* as identify.

Parameters

<i>idProject</i>	Project identify
<i>idBilling</i>	Billing identify
<i>hourlyRate</i>	New hourly rate

8.79.1.2 double Databases::RateDatabase::getRate (const int *idBilling*, const int *idProject*) const

[RateDatabase::getRate](#) Return the rate of the Project identified by *idProject* of the Billing with *idBilling* as identify.

Parameters

<i>idBilling</i>	Billing identify
<i>idProject</i>	Project identify

Returns

Rate of the Project of a Billing

8.79.1.3 **RateDatabase** * Databases::RateDatabase::instance () throw DbException *) [static]

RateDatabase::getInstance Return an instance of [RateDatabase](#).

Returns

Instance of [RateDatabase](#)

8.79.1.4 void Databases::RateDatabase::updateRateProject (int *idProject*, int *idBilling*, double *hourlyRate*)

[RateDatabase::updateRateProject](#) Update the current rate by the new *hourlyRate* of the Project identified by *id-Project* of the Billing with *idBilling* as identify.

Parameters

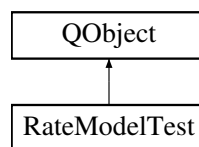
<i>idProject</i>	Project identify
<i>idBilling</i>	Billing identify
<i>hourlyRate</i>	New hourly rate

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

- /home/travis/build/FACT-Team/FactDev/src/database/ratedatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/ratedatabase.cpp

8.80 RateModelTest Class Reference

Inheritance diagram for RateModelTest:



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

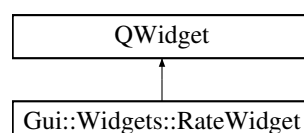
- /home/travis/build/FACT-Team/FactDev/tests/models/ratemodeltest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/ratemodeltest.cpp

8.81 Gui::Widgets::RateWidget Class Reference

Class for display Rate.

```
#include <ratewidget.h>
```

Inheritance diagram for Gui::Widgets::RateWidget:



Public Slots

- void [setDailyRate](#) ()
setDailyRate Set a new value for the daily rate
- void [setHourlyRate](#) ()
setHourlyRate Set a new value for the hourly rate

Public Member Functions

- [RateWidget](#) (QWidget *parent=0)
RateWidget::RateWidget Construct a rate widget.
- void [initRate](#) ()
RateWidget::initRate Initialize the rate.
- void [setWidgetDailyRateValue](#) (double value)
RateWidget::setWidgetDailyRateValue Modify the value of the daily rate spin box component.
- double [getDailyRate](#) ()
RateWidget::getDailyRate Get the daily rate.
- double [getHourlyRate](#) ()
RateWidget::getHourlyRate Get the hourly rate.
- void [setWidgetHourlyRateValue](#) (double value)
RateWidget::setWidgetHourlyRateValue Modify the value of the hourly rate spin box component.
- void [updateConversionRate](#) ()
updateConversionRate Update daily rate or hourly rate

8.81.1 Detailed Description

Class for display Rate.

Author

Florent Berbie

8.81.2 Constructor & Destructor Documentation

8.81.2.1 [Gui::Widgets::RateWidget::RateWidget](#) ([QWidget * parent = 0](#)) [[explicit](#)]

[RateWidget::RateWidget](#) Construct a rate widget.

Parameters

<i>parent</i>	The QWidget parent
---------------	--------------------

8.81.3 Member Function Documentation

8.81.3.1 double [Gui::Widgets::RateWidget::getDailyRate](#) ()

[RateWidget::getDailyRate](#) Get the daily rate.

Returns

The daily rate

8.81.3.2 `double Gui::Widgets::RateWidget::getHourlyRate ()`

[RateWidget::getHourlyRate](#) Get the hourly rate.

Returns

The hourly rate

8.81.3.3 `void Gui::Widgets::RateWidget::setDailyRate () [slot]`

`setDailyRate` Set a new value for the daily rate

Parameters

<i>dailyRate</i>	The new daily rate
------------------	--------------------

8.81.3.4 `void Gui::Widgets::RateWidget::setHourlyRate () [slot]`

`setHourlyRate` Set a new value for the hourly rate

Parameters

<i>hourlyRate</i>	The new hourly rate
-------------------	---------------------

8.81.3.5 `void Gui::Widgets::RateWidget::setWidgetDailyRateValue (double value)`

[RateWidget::setWidgetDailyRateValue](#) Modify the *value* of the daily rate spin box component.

Parameters

<i>value</i>	New Value
--------------	-----------

8.81.3.6 `void Gui::Widgets::RateWidget::setWidgetHourlyRateValue (double value)`

[RateWidget::setWidgetHourlyRateValue](#) Modify the *value* of the hourly rate spin box component.

Parameters

<i>value</i>	New value
--------------	-----------

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

- `/home/travis/build/FACT-Team/FactDev/src/gui/widgets/ratewidget.h`
- `/home/travis/build/FACT-Team/FactDev/src/gui/widgets/ratewidget.cpp`

8.82 Mustache::Renderer Class Reference

```
#include <mustache.h>
```

Public Member Functions

- `QString` [render](#) (const `QString` &`_template`, [Context](#) *`context`)
- `QString` [error](#) () const
- `int` [errorPos](#) () const

- QString [errorPartial](#) () const
- void [setTagMarkers](#) (const QString &startMarker, const QString &endMarker)

8.82.1 Detailed Description

Renders Mustache templates, replacing mustache tags with values from a provided context.

8.82.2 Member Function Documentation

8.82.2.1 QString [Renderer::error](#) () const

Returns a message describing the last error encountered by the previous [render\(\)](#) call.

8.82.2.2 QString [Renderer::errorPartial](#) () const

Returns the name of the partial where the error occurred, or an empty string if the error occurred in the main template.

8.82.2.3 int [Renderer::errorPos](#) () const

Returns the position in the template where the last error occurred when rendering the template or -1 if no error occurred.

If the error occurred in a partial template, the returned position is the offset in the partial template.

8.82.2.4 QString [Renderer::render](#) (const QString &_template, Context * context)

Render a Mustache template, using `context` to fetch the values used to replace Mustache tags.

8.82.2.5 void [Renderer::setTagMarkers](#) (const QString &startMarker, const QString &endMarker)

Sets the default tag start and end markers. This can be overridden within a template.

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

- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h
- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.cpp

8.83 Models::Search Class Reference

The [Search](#) class.

```
#include <search.h>
```

Public Member Functions

- [Search](#) ()
[Search::Search](#) Construct a search.
- [~Search](#) ()
[Search::Search](#) Destruct a search.
- QString [getFilter](#) ()

- Search::getFilter* Return the search filter.
- void **filterOnVarcharElements** (QString &filter, const QStringList list, QString element)
 - Search::filterOnVarcharElements* Search, for each word of the list from the filter, if it corresponds to the element in the database.
 - void **filterOnNumberElements** (QString &filter, const QStringList list, QString element)
 - Search::filterOnNumberElements* Search, for each number of the list from the filter, if it corresponds to the element in the database.
 - void **filterOnCompany** (QString &filter, const QStringList list)
 - Search::filterOnCompany* Search all companies contained in the list of the filter
 - void **filterOnReferentLastname** (QString &filter, const QStringList list)
 - Search::filterOnReferentLastname* Search all referents last name contained in the list of the filter
 - void **filterOnProjects** (QString &filter, const QStringList list)
 - Search::filterOnProjects* Search all projects contained in the list of the filter
 - void **filterOnContributories** (QString &filter, const QStringList list)
 - Search::filterOnContributories* Search all contributories contained in the list of the filter
 - void **filterOnBillsOrQuotes** (QString &filter, const QStringList list)
 - Search::filterOnBillsOrQuotes* Search all bills or quotes which are contained in the list of the filter
 - void **filterOnCustomersWithoutProject** (QString &filter, const QStringList list)
 - Search::filterOnCustomersWithoutProject* Search all customers which are contained in the list of the filter
- bool **getSearchInCompanies** () const
 - Search::getSearchInCompanies* Return if we search a company.
 - void **setSearchInCompanies** (bool searchInCompanies)
 - Search::setSearchInCompanies* Modify the filter of companies search.
 - bool **getSearchInReferentLastname** () const
 - Search::getSearchInReferentLastname* Return if we search a Last name referent.
 - void **setSearchInReferentLastname** (bool searchInReferentLastname)
 - Search::setSearchInReferentLastname* Modify the filter of referents last name search.
 - bool **getSearchInProjects** () const
 - Search::getSearchInProjects* Return TRUE if it exists one or more projets in database.
 - void **setSearchInProjects** (bool searchInProjects)
 - Search::setSearchInProjects* Modify the filter of projects search.
 - bool **searchInContributories** () const
 - Search::getSearchInContributories* Return TRUE if it exists one or more projets in database.
 - void **setSearchInContributories** (bool **searchInContributories**)
 - Search::setSearchInContributories* Modify the filter of contributories search.
 - bool **getSearchInBillsQuotes** () const
 - Search::searchInBillsQuotes* Return TRUE if it exists one or more bills or quotes in database.
 - void **setSearchInBillsQuotes** (bool searchInBillsQuotes)
 - Search::setSearchInBillsQuotes* Modify the filter of bills and quotes search.
 - bool **getGroupFilter** () const
 - Search::getGroupFilter* Return if the filter is activated.
 - void **setGroupFilter** (bool **getGroupFilter**)
 - Search::setGroupFilter* Modify if we active search filter.
 - QString **getText** () const
 - Search::getText* Return sql portion of filter.
 - void **setText** (const QString &**getText**)
 - Search::setText* Modify sql portion.

8.83.1 Detailed Description

The [Search](#) class.

Author

Antoine de Roquemaurel
Florent Berbie

8.83.2 Member Function Documentation

8.83.2.1 void Models::Search::filterOnBillsOrQuotes (QString & *filter*, const QStringList *list*)

[Search::filterOnBillsOrQuotes](#) [Search](#) all bills or quotes which are contained in the *list* of the *filter*

Parameters

<i>filter</i>	Text inputed in searchbar
<i>list</i>	List of bills or quotes

8.83.2.2 void Models::Search::filterOnCompany (QString & *filter*, const QStringList *list*)

[Search::filterOnCompany](#) [Search](#) all companies contained in the *list* of the *filter*

Parameters

<i>filter</i>	Text inputed in searchbar
<i>list</i>	List of companies

8.83.2.3 void Models::Search::filterOnContributories (QString & *filter*, const QStringList *list*)

[Search::filterOnContributories](#) [Search](#) all contributories contained in the *list* of the *filter*

Parameters

<i>filter</i>	Text inputed in searchbar
<i>list</i>	List of contributories

8.83.2.4 void Models::Search::filterOnCustomersWithoutProject (QString & *filter*, const QStringList *list*)

[Search::filterOnCustomersWithoutProject](#) [Search](#) all customers which are contained in the *list* of the *filter*

Parameters

<i>filter</i>	Text inputed in searchbar
<i>list</i>	List of bills or quotes

8.83.2.5 void Models::Search::filterOnNumberElements (QString & *filter*, const QStringList *list*, QString *element*)

[Search::filterOnNumberElements](#) [Search](#), for each number of the *list* from the *filter*, if it corresponds to the *element* in the database.

Parameters

<i>filter</i>	Text inputed in the searchbar
<i>list</i>	List of numbers from the <i>filter</i>
<i>element</i>	Attribute name into the database

8.83.2.6 void Models::Search::filterOnProjects (QString & *filter*, const QStringList *list*)

[Search::filterOnProjects](#) Search all projects contained in the *list* of the *filter*

Parameters

<i>filter</i>	Text inputed in searchbar
<i>list</i>	List of projects

8.83.2.7 void Models::Search::filterOnReferentLastname (QString & *filter*, const QStringList *list*)

[Search::filterOnReferentLastname](#) Search all referents last name contained in the *list* of the *filter*

Parameters

<i>filter</i>	Text inputed in searchbar
<i>list</i>	List of referent last name

8.83.2.8 void Models::Search::filterOnVarcharElements (QString & *filter*, const QStringList *list*, QString *element*)

[Search::filterOnVarcharElements](#) Search, for each word of the *list* from the *filter*, if it corresponds to the *element* in the database.

Parameters

<i>filter</i>	Text inputed in the searchbar
<i>list</i>	List of words from the <i>filter</i>
<i>element</i>	Attribute name into the database

8.83.2.9 QString Models::Search::getFilter ()

[Search::getFilter](#) Return the search filter.

Returns

filter selected (sql portion)

8.83.2.10 bool Models::Search::getGroupFilter () const

[Search::getGroupFilter](#) Return if the filter is activated.

Returns

boolean if search filter is activated

8.83.2.11 bool Models::Search::getSearchInBillsQuotes () const

[Search::searchInBillsQuotes](#) Return TRUE if it exists one or more bills or quotes in database.

Returns

boolean if bills or quotes are existing

8.83.2.12 `bool Models::Search::getSearchInCompanies () const`

[Search::getSearchInCompanies](#) Return if we search a company.

Returns

boolean if we search a company

8.83.2.13 `bool Models::Search::getSearchInProjects () const`

[Search::getSearchInProjects](#) Return TRUE if it exists one or more projets in database.

Returns

boolean if project are existing

8.83.2.14 `bool Models::Search::getSearchInReferentLastname () const`

[Search::getSearchInReferentLastname](#) Return if we search a Last name referent.

Returns

boolean if search concerns the last name of referent

8.83.2.15 `QString Models::Search::getText () const`

[Search::getText](#) Return sql portion of filter.

Returns

QString the sql portion

8.83.2.16 `bool Models::Search::searchInContributories () const`

[Search::getSearchInContributories](#) Return TRUE if it exists one or more projets in database.

Returns

boolean if contributories are existing

8.83.2.17 `void Models::Search::setGroupFilter (bool getGroupFilter)`

[Search::setGroupFilter](#) Modify if we active search filter.

Parameters

<i>getGroupFilter</i>	Get if filter is actived
-----------------------	--------------------------

8.83.2.18 void Models::Search::setSearchInBillsQuotes (bool *searchInBillsQuotes*)

[Search::setSearchInBillsQuotes](#) Modify the filter of bills and quotes search.

Parameters

<i>searchInBills-Quotes</i>	Search in bills or quotes which are concerned
-----------------------------	---

8.83.2.19 void Models::Search::setSearchInCompanies (bool *searchInCompanies*)

[Search::setSearchInCompanies](#) Modify the filter of companies search.

Parameters

<i>getSearchIn-Companies</i>	Search in companies is concerned
------------------------------	--

8.83.2.20 void Models::Search::setSearchInContributories (bool *searchInContributories*)

[Search::setSearchInContributories](#) Modify the filter of contributories search.

Parameters

<i>searchIn-Contributories</i>	Search in contributories which are concerned
--------------------------------	--

8.83.2.21 void Models::Search::setSearchInProjects (bool *searchInProjects*)

[Search::setSearchInProjects](#) Modify the filter of projects search.

Parameters

<i>searchInProjects</i>	Search in projects which are concerned
-------------------------	--

8.83.2.22 void Models::Search::setSearchInReferentLastname (bool *searchInReferentLastname*)

[Search::setSearchInReferentLastname](#) Modify the filter of referents last name search.

Parameters

<i>searchIn-Referent-Lastname</i>	Search in referents last name which are concerned
-----------------------------------	---

8.83.2.23 void Models::Search::setText (const QString & *getText*)

[Search::setText](#) Modify sql portion.

Parameters

<i>getText</i>	Get sql portion
----------------	-----------------

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

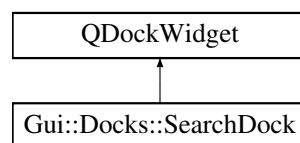
- /home/travis/build/FACT-Team/FactDev/src/models/search.h
- /home/travis/build/FACT-Team/FactDev/src/models/search.cpp

8.84 Gui::Docks::SearchDock Class Reference

The [SearchDock](#) class Dock which contains search bar.

```
#include <searchdock.h>
```

Inheritance diagram for Gui::Docks::SearchDock:



Public Slots

- void [search](#) (QString text)
[SearchDock::search](#) Search elements in database which correspond to the text

Signals

- void [textChanged](#) (QString text)
[SearchDock::textChanged](#) Current text changed by text

Public Member Functions

- [SearchDock](#) (QWidget *parent=0, Qt::WindowFlags flags=0)
[SearchDock::SearchDock](#) Construct a [SearchDock](#).

8.84.1 Detailed Description

The [SearchDock](#) class Dock which contains search bar.

8.84.2 Constructor & Destructor Documentation

8.84.2.1 [Gui::Docks::SearchDock::SearchDock](#) ([QWidget](#) * *parent* = 0, [Qt::WindowFlags](#) *flags* = 0) [explicit]

[SearchDock::SearchDock](#) Construct a [SearchDock](#).

Parameters

<i>parent</i>	Widget Parent
<i>flags</i>	Window flag

8.84.3 Member Function Documentation

8.84.3.1 void Gui::Docks::SearchDock::search (QString *text*) [slot]

[SearchDock::search](#) Search elements in database which correspond to the *text*

Parameters

<i>text</i>	Element to search
-------------	-------------------

8.84.3.2 void Gui::Docks::SearchDock::textChanged (QString *text*) [signal]

[SearchDock::textChanged](#) Current text changed by *text*

Parameters

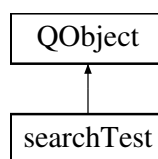
<i>text</i>	New text
-------------	----------

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

- /home/travis/build/FACT-Team/FactDev/src/gui/docks/searchdock.h
- /home/travis/build/FACT-Team/FactDev/src/gui/docks/searchdock.cpp

8.85 searchTest Class Reference

Inheritance diagram for searchTest:



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

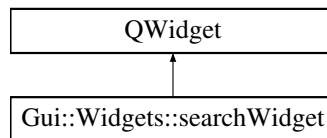
- /home/travis/build/FACT-Team/FactDev/tests/models/searchtest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/searchtest.cpp

8.86 Gui::Widgets::searchWidget Class Reference

Class for search in database.

```
#include <searchwidget.h>
```

Inheritance diagram for Gui::Widgets::searchWidget:



Public Slots

- void [search](#) (QString toSearch)
searchWidget::launch a search
- void [getCustomerData](#) ()
searchWidget::getCustomerData Return data on the customer selected in the QTableView and display this data (First-name, Lastname, Company)

Signals

- void [selectCustomer](#) ()
searchWidget::selectCustomer Signal that the Customer selected has changed

Public Member Functions

- [searchWidget](#) (QWidget *parent=0)
searchWidget::searchWidget Construct a search widget
- int [getCurrentCustomerId](#) ()
searchWidget::getCurrentCustomerId Return the id of the customer selected in the table
- bool [isCustomerSelected](#) () const
searchWidget::isCustomerSelected Return TRUE if a customer is selected, else FALSE
- void [selectCustomer](#) (int id)
searchWidget::selectCustomer Select the Customer with it id
- int [getIdCustomer](#) () const
searchWidget::getIdCustomer Return the Customer id of the Customer line selected
- void [setIdCustomer](#) (int idCustomer)
searchWidget::setIdCustomer Change the customer id by the new idCustomer

8.86.1 Detailed Description

Class for search in database.

Author

Antoine de Roquemaurel

8.86.2 Constructor & Destructor Documentation

8.86.2.1 Gui::Widgets::searchWidget::searchWidget (QWidget * parent = 0) [explicit]

[searchWidget::searchWidget](#) Construct a search widget

Parameters

<i>parent</i>	The QWidget parent
---------------	--------------------

8.86.3 Member Function Documentation

8.86.3.1 `int Gui::Widgets::searchWidget::getCurrentCustomerId ()`

[searchWidget::getCurrentCustomerId](#) Return the id of the customer selected in the table

Returns

id of the current customer

8.86.3.2 `int Gui::Widgets::searchWidget::getIdCustomer () const`

[searchWidget::getIdCustomer](#) Return the Customer id of the Customer line selected

Returns

The Customer ID

8.86.3.3 `bool Gui::Widgets::searchWidget::isCustomerSelected () const`

[searchWidget::isCustomerSelected](#) Return TRUE if a customer is selected, else FALSE

Returns

boolean

8.86.3.4 `void Gui::Widgets::searchWidget::search (QString toSearch) [slot]`

`searchWidget::launch` a search

Parameters

<i>toSearch</i>	The value to search
-----------------	---------------------

8.86.3.5 `void Gui::Widgets::searchWidget::selectCustomer (int id)`

[searchWidget::selectCustomer](#) Select the Customer with it *id*

Parameters

<i>id</i>	Customer ID
-----------	-------------

8.86.3.6 `void Gui::Widgets::searchWidget::setIdCustomer (int idCustomer)`

[searchWidget::setIdCustomer](#) Change the customer id by the new *idCustomer*

Parameters

<i>idCustomer</i>	Customer id
-------------------	-------------

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

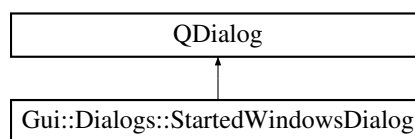
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/searchwidget.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/searchwidget.cpp

8.87 Gui::Dialogs::StartedWindowsDialog Class Reference

The [StartedWindowsDialog](#) class Construct a Windows for the first begin.

```
#include <startedwindowsdialog.h>
```

Inheritance diagram for Gui::Dialogs::StartedWindowsDialog:



Public Slots

- void [checkFields](#) ()
StartedWindowsDialog::checkFields Check all fields of dialog components.
- void [backToPage2](#) ()
StartedWindowsDialog::backToPage2 Return to the second page.
- void [nextToPage2](#) ()
StartedWindowsDialog::nextToPage2 Go to the second page.
- void [nextToPage3](#) ()
StartedWindowsDialog::nextToPage3 Go to the third page.
- void [databaseTypeChanged](#) (const int index)
StartedWindowsDialog::databaseTypeChanged Listener on the Database type combobox.
- void [updateNextButton](#) ()
StartedWindowsDialog::updateNextButton Check if the next button is enabled.
- void [accept](#) ()
StartedWindowsDialog::accept Valid data inputed by user and add these data in Database.

Public Member Functions

- [StartedWindowsDialog](#) (QWidget *parent=0)
StartedWindowsDialog::StartedWindowsDialog Construct a [StartedWindowsDialog](#).
- void [fillFields](#) ()
StartedWindowsDialog::fillFields Fill line edits with the data of the user.
- QPixmap [getImage](#) (QString path, int width=256, int height=256)
StartedWindowsDialog::getImage Return a scaled image from the icon specified by it path. The image returned has a resolution of width*height (default 256*256)
- bool [isDatabaseTypeValid](#) ()
StartedWindowsDialog::isDatabaseTypeValid Return TRUE if settings of database type is valid else return FALSE.
- bool [isDatabaseCentralized](#) ()
StartedWindowsDialog::isDatabaseCentralized Return TRUE if the current type of database is centralized else FALSE.

8.87.1 Detailed Description

The [StartedWindowsDialog](#) class Construct a Windows for the first begin.

Author

Florent Berbie

See Also

[DatabaseSettingsWidget](#)

8.87.2 Constructor & Destructor Documentation

8.87.2.1 [Gui::Dialogs::StartedWindowsDialog::StartedWindowsDialog \(QWidget * *parent* = 0 \)](#) `[explicit]`

[StartedWindowsDialog::StartedWindowsDialog](#) Construct a [StartedWindowsDialog](#).

Parameters

<i>parent</i>	QWidget parent
---------------	----------------

8.87.3 Member Function Documentation

8.87.3.1 `void Gui::Dialogs::StartedWindowsDialog::databaseTypeChanged (const int index)` `[slot]`

[StartedWindowsDialog::databaseTypeChanged](#) Listener on the Database type combobox.

Parameters

<i>index</i>	Current index selected
--------------	------------------------

8.87.3.2 `QPixmap Gui::Dialogs::StartedWindowsDialog::getImage (QString path, int width = 256, int height = 256)`

[StartedWindowsDialog::getImage](#) Return a scaled image from the icon specified by it *path*. The image returned has a resolution of *width***height* (default 256*256)

Parameters

<i>path</i>	Icon path
<i>width</i>	Icon width
<i>height</i>	Icon height

Returns

Scaled image

8.87.3.3 `bool Gui::Dialogs::StartedWindowsDialog::isDatabaseCentralized ()`

[StartedWindowsDialog::isDatabaseCentralized](#) Return TRUE if the current type of database is centralized else FALSE.

Returns

boolean

8.87.3.4 bool Gui::Dialogs::StartedWindowsDialog::isDatabaseTypeValid ()

[StartedWindowsDialog::isDatabaseTypeValid](#) Return TRUE if settings of database type is valid else return FALSE.

Returns

boolean on the database type validity

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

- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/startedwindowsdialog.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/startedwindowsdialog.cpp

8.88 Models::Statistics Class Reference

Static Public Member Functions

- static QPair< int, double > [getTurnoverBetweenDates](#) (QDate begin, QDate end)
getTurnoverBetweenDates Compute the turnover between 2 dates

8.88.1 Member Function Documentation

8.88.1.1 QPair< int, double > Statistics::getTurnoverBetweenDates (QDate *begin*, QDate *end*) [static]

[getTurnoverBetweenDates](#) Compute the turnover between 2 dates

Parameters

<i>begin</i>	The beginning date
<i>end</i>	The ending date

Returns

A Qpair who contains the number of billing and their costs

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

- /home/travis/build/FACT-Team/FactDev/src/models/statistics.h
- /home/travis/build/FACT-Team/FactDev/src/models/statistics.cpp

8.89 Utils::String Class Reference

The Utils class.

```
#include <string.h>
```

Static Public Member Functions

- static QString [firstLetterToUpper](#) (QString s)
firstLetterToUpper Put the first letter of a string in capslock

8.89.1 Detailed Description

The Utils class.

Author

Antoine de Roquemaurel

8.89.2 Member Function Documentation

8.89.2.1 QString Utils::String::firstLetterToUpper (QString s) [static]

firstLetterToUpper Put the first letter of a string in capslock

Parameters

s	The string to display
---	-----------------------

Returns

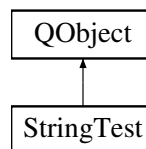
The new string with caps

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

- /home/travis/build/FACT-Team/FactDev/src/Utils/string.h
- /home/travis/build/FACT-Team/FactDev/src/Utils/string.cpp

8.90 StringTest Class Reference

Inheritance diagram for StringTest:



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

- /home/travis/build/FACT-Team/FactDev/tests/Utils/stringtest.h
- /home/travis/build/FACT-Team/FactDev/tests/Utils/stringtest.cpp

8.91 Mustache::Tag Struct Reference

```
#include <mustache.h>
```

Public Types

- enum `Type` { `Null`, `Value`, `SectionStart`, `InvertedSectionStart`, `SectionEnd`, `Partial`, `Comment`, `SetDelimiter` }
- enum `EscapeMode` { `Escape`, `Unescape`, `Raw` }

Public Attributes

- **Type** `type`
- `QString` `key`
- `int` `start`
- `int` `end`
- `EscapeMode` `escapeMode`

8.91.1 Detailed Description

Holds properties of a tag in a mustache template.

8.91.2 Member Enumeration Documentation

8.91.2.1 `enum Mustache::Tag::Type`

Enumerator

SectionStart A `{{key}}` or `{{{key}}}` tag.

InvertedSectionStart A `{{#section}}` tag.

SectionEnd An `{{^inverted-section}}` tag.

Partial A `{{/section}}` tag.

Comment A `{{^partial}}` tag.

SetDelimiter A `{{! comment }}` tag. A `{{=<% %>=}}` tag

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

- `/home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/src/mustache.h`

8.92 `TestAdder< T >` Class Template Reference

Public Member Functions

- **TestAdder** (`const QString &name`)

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

- `/home/travis/build/FACT-Team/FactDev/tests/QStringRunner/testadder.cpp`

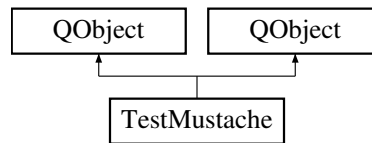
8.93 `testadder` Class Reference

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

- `/home/travis/build/FACT-Team/FactDev/tests/QStringRunner/testadder.h`

8.94 TestMustache Class Reference

Inheritance diagram for TestMustache:



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

- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/tests/test_mustache.h
- /home/travis/build/FACT-Team/FactDev/src/libs/qt-mustache/tests/test_mustache.cpp

8.95 TestRunner Class Reference

Public Member Functions

- `template<typename T >`
char **RegisterTest** (QString name)
- int **RunAll** ()

Static Public Member Functions

- static [TestRunner](#) & **Instance** ()

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

- /home/travis/build/FACT-Team/FactDev/tests/QTestRunner/testrunner.h
- /home/travis/build/FACT-Team/FactDev/tests/QTestRunner/testrunner.cpp

8.96 Generator::TexGenerator Class Reference

The [TexGenerator](#) class Generate a LaTeX file.

```
#include <texgenerator.h>
```

Public Member Functions

- [TexGenerator](#) (QString tpl)
[TexGenerator::TexGenerator](#) Construct a [TexGenerator](#).
- void [generate](#) (QVariantHash data, QString path)
[TexGenerator::generate](#) Generate a LaTeX file into a file specified by the path and which contains data

8.96.1 Detailed Description

The [TexGenerator](#) class Generate a LaTeX file.

8.96.2 Constructor & Destructor Documentation

8.96.2.1 Generator::TexGenerator::TexGenerator (QString *tpl*)

[TexGenerator::TexGenerator](#) Construct a [TexGenerator](#).

Parameters

<i>tpl</i>	Template LaTeX to apply
------------	-------------------------

8.96.3 Member Function Documentation

8.96.3.1 void Generator::TexGenerator::generate (QVariantHash *data*, QString *path*)

[TexGenerator::generate](#) Generate a LaTeX file into a file specified by the *path* and which contains *data*

Parameters

<i>data</i>	Data to integrate in the file
<i>path</i>	Path of the out file

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

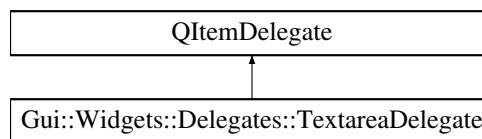
- /home/travis/build/FACT-Team/FactDev/src/generator/texgenerator.h
- /home/travis/build/FACT-Team/FactDev/src/generator/texgenerator.cpp

8.97 Gui::Widgets::Delegates::TextareaDelegate Class Reference

The [TextareaDelegate](#) class.

```
#include <textareadelegate.h>
```

Inheritance diagram for Gui::Widgets::Delegates::TextareaDelegate:



Public Member Functions

- [TextareaDelegate](#) (QWidget *parent=0)
[TextareaDelegate::TextareaDelegate](#).
- void [setModelData](#) (QWidget *editor, QAbstractItemModel *model, const QModelIndex &index) const
[TextareaDelegate::setModelData](#) Sets the data model from content in editor.
- void [updateEditorGeometry](#) (QWidget *editor, const QStyleOptionViewItem &option, const QModelIndex &) const
[TextareaDelegate::updateEditorGeometry](#) Update the editor for the item specified by index according to the style option given.
- QWidget * [createEditor](#) (QWidget *parent, const QStyleOptionViewItem &, const QModelIndex &) const
[TextareaDelegate::createEditor](#) Return a QTextEdit specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void [setEditorData](#) (QWidget *editor, const QModelIndex &index) const
[TextareaDelegate::setEditorData](#) Sets the data to be displayed and edited by the editor from the data model item specified by the model index

8.97.1 Detailed Description

The [TextareaDelegate](#) class.

Author

Antoine de Roquemaurel

8.97.2 Constructor & Destructor Documentation

8.97.2.1 Gui::Widgets::Delegates::TextareaDelegate::TextareaDelegate (QWidget * *parent* = 0)

[TextareaDelegate::TextareaDelegate](#).

Parameters

<i>parent</i>	
---------------	--

8.97.3 Member Function Documentation

8.97.3.1 QWidget * Gui::Widgets::Delegates::TextareaDelegate::createEditor (QWidget * *parent*, const QStyleOptionViewItem & , const QModelIndex &) const

[TextareaDelegate::createEditor](#) Return a QTextEdit specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Returns

[TextareaDelegate](#)

8.97.3.2 void Gui::Widgets::Delegates::TextareaDelegate::setEditorData (QWidget * *editor*, const QModelIndex & *index*) const

[TextareaDelegate::setEditorData](#) Sets the data to be displayed and edited by the *editor* from the data model item specified by the model *index*

Parameters

<i>editor</i>	Data edited
<i>index</i>	Index of the model to edit

8.97.3.3 void Gui::Widgets::Delegates::TextareaDelegate::setModelData (QWidget * *editor*, QAbstractItemModel * *model*, const QModelIndex & *index*) const

[TextareaDelegate::setModelData](#) Sets the data model from content in editor.

Parameters

<i>editor</i>	Data edited
<i>index</i>	Index of the model to edit

8.97.3.4 void Gui::Widgets::Delegates::TextareaDelegate::updateEditorGeometry (QWidget * *editor*, const QStyleOptionViewItem & *option*, const QModelIndex &) const

[TextareaDelegate::updateEditorGeometry](#) Update the *editor* for the item specified by *index* according to the style *option* given.

Parameters

<i>editor</i>	Editor widget to update
<i>option</i>	Style option
<i>index</i>	Item index

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

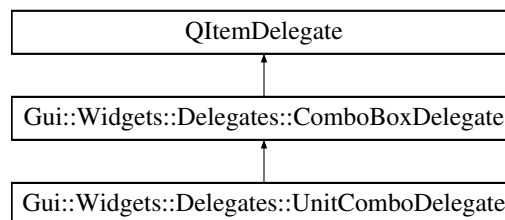
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/textareadelegate.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/textareadelegate.cpp

8.98 Gui::Widgets::Delegates::UnitComboDelegate Class Reference

The [UnitComboDelegate](#) class.

```
#include <unitcombodelegate.h>
```

Inheritance diagram for Gui::Widgets::Delegates::UnitComboDelegate:



Public Member Functions

- [UnitComboDelegate](#) (QObject *parent=0)
[UnitComboDelegate::UnitComboDelegate](#).
- QWidget * [createEditor](#) (QWidget *parent, const QStyleOptionViewItem &option, const QModelIndex &index) const
[UnitComboDelegate::createEditor](#) Return a ComboBox specified by index item defined by the parent widget and style option which are used to control how the editor widgets appears.
- void [paint](#) (QPainter *painter, const QStyleOptionViewItem &option, const QModelIndex &index) const
[UnitComboDelegate::paint](#) Renders the delegate using the given painter and style option for the item specified by index

8.98.1 Detailed Description

The [UnitComboDelegate](#) class.

8.98.2 Constructor & Destructor Documentation

8.98.2.1 Gui::Widgets::Delegates::UnitComboDelegate::UnitComboDelegate (QObject * parent = 0)

[UnitComboDelegate::UnitComboDelegate](#).

Parameters

<i>parent</i>	
---------------	--

8.98.3 Member Function Documentation

8.98.3.1 `QWidget * Gui::Widgets::Delegates::UnitComboDelegate::createEditor (QWidget * parent, const QStyleOptionViewItem & option, const QModelIndex & index) const` [virtual]

[UnitComboDelegate::createEditor](#) Return a ComboBox specified by *index* item defined by the *parent* widget and style *option* which are used to control how the editor widgets appears.

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Returns

ComboBox

Implements [Gui::Widgets::Delegates::ComboBoxDelegate](#).

8.98.3.2 `void Gui::Widgets::Delegates::UnitComboDelegate::paint (QPainter * painter, const QStyleOptionViewItem & option, const QModelIndex & index) const` [virtual]

[UnitComboDelegate::paint](#) Renders the delegate using the given *painter* and style *option* for the item specified by *index*

Parameters

<i>parent</i>	Widget parent
<i>option</i>	Option style
<i>index</i>	Index for editing

Implements [Gui::Widgets::Delegates::ComboBoxDelegate](#).

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

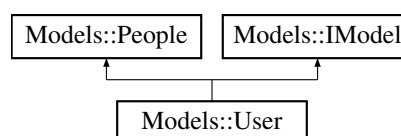
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/unitcombodelegate.h
- /home/travis/build/FACT-Team/FactDev/src/gui/widgets/delegates/unitcombodelegate.cpp

8.99 Models::User Class Reference

The [User](#) class [User](#) of it application.

```
#include <user.h>
```

Inheritance diagram for Models::User:



Public Member Functions

- [User](#) ()
User::User. Construct an [User](#).
- [User](#) (int id)
User::User. Construct a [User](#) with the identify id
- void [commit](#) ()
User::commit Update user data in [User](#) table on the database.
- void [hydrat](#) (int id=1)
User::hydrat Get data of the user who is specified by id from the database.
- void [remove](#) ()
remove Remove the current [User](#)
- QVariantHash [getDataMap](#) ()
getDataMap Get all data of model with a HashMap key/value
- void [updateFolders](#) (void)
MainWindow::updateFolders Make directories which contain quotes and billings. Directories are the same than theirs of the Tree organisation (without Projects).
- QString [getTitle](#) () const
User::getTitle Return a short description of [User](#) (company) activity.
- void [setTitle](#) (const QString &title)
User::setTitle Modify the user/company activities description
- QString [getNoSiret](#) () const
User::getNoSiret Return the SIRET number (company registration number)
- void [setNoSiret](#) (const QString &noSiret)
User::setNoSiret Modify the SIRET number (company registration number) noSiret
- QString [getWorkspaceName](#) () const
User::getWorkspaceName Return the name of the workspace user.
- void [setWorkspaceName](#) (const QString &workspaceName)
User::setWorkspaceName Change the current workspace name by the new workspaceName
- QString [getWorkspacePath](#) () const
User::getWorkspacePath Return the path of the workspace user.
- void [setWorkspacePath](#) (const QString &workspacePath)
User::setWorkspacePath Change the current workspace path by the new workspacePath
- bool [operator==](#) (const [User](#) &u)
User::operator == Re-define the operator "==" to compare if the current [User](#) is the same to the other [User](#) c Return TRUE if both Users are the same, else FALSE.
- bool [operator!=](#) (const [User](#) &u)
User::operator != Re-define the operator "!=" to compare if the current [User](#) is differnt to the other [User](#) c Return TRUE if both Users are different, else FALSE.

Additional Inherited Members

8.99.1 Detailed Description

The [User](#) class [User](#) of it application.

Author

Florent Berbie

8.99.2 Constructor & Destructor Documentation

8.99.2.1 Models::User::User (int *id*)

[User::User](#). Construct a [User](#) with the identify *id*

Parameters

<i>id</i>	User id
-----------	-------------------------

8.99.3 Member Function Documentation

8.99.3.1 QVariantHash Models::User::getDataMap () [virtual]

getDataMap Get all data of model with a HashMap key/value

Returns

Model's data

Implements [Models::IModel](#).

8.99.3.2 QString Models::User::getNoSiret () const

[User::getNoSiret](#) Return the SIRET number (company registration number)

Returns

SIRET number

8.99.3.3 QString Models::User::getTitle () const

[User::getTitle](#) Return a short description of [User](#) (company) activity.

Returns

a short description of user (company) activity

8.99.3.4 QString Models::User::getWorkspaceName () const

[User::getWorkspaceName](#) Return the name of the workspace user.

Returns

workspace name

8.99.3.5 QString Models::User::getWorkspacePath () const

[User::getWorkspacePath](#) Return the path of the workspace user.

Returns

workspace path

8.99.3.6 void Models::User::hydrat (int *id* = 1) [virtual]

[User::hydrat](#) Get data of the user who is specified by *id* from the database.

Parameters

<i>id</i>	User identify
-----------	-------------------------------

Implements [Models::IModel](#).

8.99.3.7 bool Models::User::operator!= (const User & *u*)

[User::operator !=](#) Re-define the operator "!=" to compare if the current [User](#) is different to the other [User](#) c Return TRUE if both Users are different, else FALSE.

Parameters

<i>u</i>	User to compare
----------	---------------------------------

Returns

boolean

8.99.3.8 bool Models::User::operator== (const User & *u*)

[User::operator ==](#) Re-define the operator "==" to compare if the current [User](#) is the same to the other [User](#) c Return TRUE if both Users are the same, else FALSE.

Parameters

<i>u</i>	User to compare
----------	---------------------------------

Returns

boolean

8.99.3.9 void Models::User::setNoSiret (const QString & *noSiret*)

[User::setNoSiret](#) Modify the SIRET number (company registration number) *noSiret*

Parameters

<i>noSiret</i>	SIRET number
----------------	--------------

8.99.3.10 void Models::User::setTitle (const QString & *title*)

[User::setTitle](#) Modify the user/company activities *description*

Parameters

<i>title</i>	Short description on activity(ies) of User company
--------------	--

8.99.3.11 void Models::User::setWorkspaceName (const QString & *workspaceName*)

[User::setWorkspaceName](#) Change the current workspace name by the new *workspaceName*

Parameters

<i>workspaceName</i>	
----------------------	--

8.99.3.12 void Models::User::setWorkspacePath (const QString & *workspacePath*)

[User::setWorkspacePath](#) Change the current workspace path by the new *workspacePath*

Parameters

<i>workspacePath</i>	
----------------------	--

8.99.3.13 void Models::User::updateFolders (void)

MainWindow::updateFolders Make directories which contain quotes and billings. Directories are the same than theirs of the Tree organisation (without Projects).

Organisation of folders are formed like this:

- COMPANY CustomerLastname CustomerFirstname/
 - Quotes/
 - * quote1 ...
 - Billings/
 - * billing1 ...

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

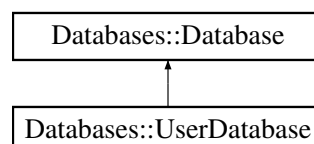
- /home/travis/build/FACT-Team/FactDev/src/models/user.h
- /home/travis/build/FACT-Team/FactDev/src/models/user.cpp

8.100 Databases::UserDatabase Class Reference

The [UserDatabase](#) class Access to User data in the the table User of the [Database](#)

```
#include <userdatabase.h>
```

Inheritance diagram for Databases::UserDatabase:



Public Member Functions

- [Models::User](#) * [getUser](#) (const int pld=1)
 [UserDatabase::getUser](#) Get informations about the user (identified by 'pld')
- void [updateUser](#) (const [Models::User](#) &)
 [UserDatabase::updateUser](#) Update informations about the user.

Static Public Member Functions

- static [UserDatabase](#) * [instance](#) () throw (DbException*)
UserDatabase::getInstance Return an instance of [UserDatabase](#).

Additional Inherited Members

8.100.1 Detailed Description

The [UserDatabase](#) class Access to User data in the the table User of the [Database](#)

Author

Florent Berbie

See Also

[Database](#)
[User](#)

8.100.2 Member Function Documentation

8.100.2.1 [Models::User](#) * [Databases::UserDatabase::getUser](#) (const int *pld* = 1)

[UserDatabase::getUser](#) Get informations about the user (identified by 'pld')

Parameters

<i>pld</i>	user id (1 default)
------------	---------------------

Returns

the user

8.100.2.2 [UserDatabase](#) * [Databases::UserDatabase::instance](#) () throw [DbException](#) *) [static]

[UserDatabase::getInstance](#) Return an instance of [UserDatabase](#).

Returns

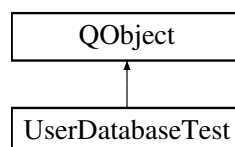
Instance of [UserDatabase](#)

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

- /home/travis/build/FACT-Team/FactDev/src/database/userdatabase.h
- /home/travis/build/FACT-Team/FactDev/src/database/userdatabase.cpp

8.101 UserDatabaseTest Class Reference

Inheritance diagram for UserDatabaseTest:



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

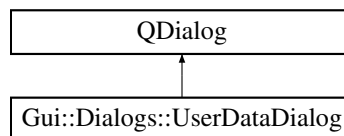
- /home/travis/build/FACT-Team/FactDev/tests/database/userdatabasetest.h
- /home/travis/build/FACT-Team/FactDev/tests/database/userdatabasetest.cpp

8.102 Gui::Dialogs::UserDataDialog Class Reference

The [UserDataDialog](#) class Window to fill user data.

```
#include <userdatadialog.h>
```

Inheritance diagram for Gui::Dialogs::UserDataDialog:



Public Slots

- void [checkFields](#) ()
[UserDataDialog::checkFields](#) Check all fields of dialog components.
- void [browseWorkspacePath](#) ()
[UserDataDialog::browseWorkspacePath](#) Open a new window to define the workspace path of the user.

Public Member Functions

- [UserDataDialog](#) (QWidget *parent=0)
[UserDataDialog::UserDataDialog](#) Construct a window with user data.
- void [fillFields](#) ()
[UserDataDialog::fillFields](#) Fill line edits with the data of the user.
- void [accept](#) ()
[UserDataDialog::accept](#) Valid data inputed by user and add these data in Database.
- void [reject](#) ()
[UserDataDialog::reject](#) Cancel the operation and close the windows.

8.102.1 Detailed Description

The [UserDataDialog](#) class Window to fill user data.

Author

Florent Berbie

See Also

Project

8.102.2 Constructor & Destructor Documentation

8.102.2.1 Gui::Dialogs::UserDataDialog::UserDataDialog (QWidget * parent = 0) [explicit]

[UserDataDialog::UserDataDialog](#) Construct a window with user data.

Parameters

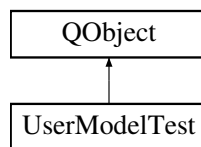
<i>parent</i>	
---------------	--

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

- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/userdatadialog.h
- /home/travis/build/FACT-Team/FactDev/src/gui/dialogs/userdatadialog.cpp

8.103 UserModelTest Class Reference

Inheritance diagram for UserModelTest:



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

- /home/travis/build/FACT-Team/FactDev/tests/models/usermodeltest.h
- /home/travis/build/FACT-Team/FactDev/tests/models/usermodeltest.cpp

Index

- add
 - Gui::Widgets::ContributoriesWidget, [74](#)
- addBill
 - Gui::MainWindow, [120](#)
- addBilling
 - Databases::BillingDatabase, [32](#)
- addBillingProject
 - Databases::BillingDatabase, [32](#)
- addBillingToProject
 - Utils::HierarchicalSystem, [111](#)
- addContributory
 - Databases::ContributoryDatabase, [79](#)
 - Models::Billing, [27](#)
 - Models::ContributoriesList, [66](#)
- addCustomer
 - Databases::CustomerDatabase, [89](#)
- addDoc
 - Gui::MainWindow, [120](#)
- addProject
 - Databases::ProjectDatabase, [147](#)
 - Gui::MainWindow, [120](#)
 - Gui::Widgets::ContributoriesWidget, [74](#)
 - Models::ContributoriesList, [67](#)
- AddProjectDialog
 - Gui::Dialogs::AddProjectDialog, [22](#)
- addProjectToCustomer
 - Utils::HierarchicalSystem, [111](#)
- addQuote
 - Gui::MainWindow, [121](#)
- AddQuoteDialog
 - Gui::Dialogs::AddQuoteDialog, [23](#)
- addRateProject
 - Databases::RateDatabase, [160](#)
- allProjectsChosen
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, [143](#)
- append
 - Gui::Widgets::WdgModels::BillingsTableModel, [39](#)
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, [70](#)
 - Gui::Widgets::WdgModels::CustomersTableModel, [96](#)
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, [143](#)
 - Gui::Widgets::WdgModels::ProjectsTableModel, [152](#)
- beginDateControl
 - Gui::Dialogs::ComputeTurnoverDialog, [62](#)
- Billing
 - Models::Billing, [27](#)
- BillingDatabaseTest, [36](#)
- BillingModelTest, [36](#)
- BillingsTableModel
 - Gui::Widgets::WdgModels::BillingsTableModel, [38](#)
- canEval
 - CounterContext, [84](#)
 - Mustache::Context, [64](#)
 - Mustache::QtVariantContext, [156](#)
- changeDatabase
 - Databases::Database, [99](#)
- check
 - Gui::Widgets::CheckFields::CheckEmail, [43](#)
 - Gui::Widgets::CheckFields::CheckFieldsLetters, [44](#)
 - Gui::Widgets::CheckFields::CheckFieldsNumbers, [46](#)
 - Gui::Widgets::CheckFields::CheckIpAddress, [47](#)
 - Gui::Widgets::CheckFields::CheckLogin, [48](#)
 - Gui::Widgets::CheckFields::CheckPhone, [50](#)
 - Gui::Widgets::CheckFields::CheckPortNumber, [52](#)
 - Gui::Widgets::CheckFields::CheckPostalCode, [53](#)
 - Gui::Widgets::CheckFields::CheckSiretNumber, [56](#)
 - Gui::Widgets::CheckFields::CheckUntilField, [57](#)
 - Gui::Widgets::CheckFields::CheckValidField, [58](#)
 - Gui::Widgets::CheckFields::ICheckField, [112](#)
- CheckCity
 - Gui::Widgets::CheckFields::CheckCity, [41](#)
- CheckCountry
 - Gui::Widgets::CheckFields::CheckCountry, [42](#)
- CheckEmail
 - Gui::Widgets::CheckFields::CheckEmail, [43](#)
- CheckFieldsLetters
 - Gui::Widgets::CheckFields::CheckFieldsLetters, [44](#)
- CheckFieldsNumbers
 - Gui::Widgets::CheckFields::CheckFieldsNumbers, [45](#)
- CheckIpAddress
 - Gui::Widgets::CheckFields::CheckIpAddress, [47](#)
- CheckLogin
 - Gui::Widgets::CheckFields::CheckLogin, [48](#)
- CheckName
 - Gui::Widgets::CheckFields::CheckName, [49](#)
- CheckPhone
 - Gui::Widgets::CheckFields::CheckPhone, [50](#)
- CheckPortNumber
 - Gui::Widgets::CheckFields::CheckPortNumber, [52](#)
- CheckPostalCode
 - Gui::Widgets::CheckFields::CheckPostalCode, [53](#)
- CheckQLineEdit

- Gui::Widgets::CheckFields::CheckQLineEdit, 54
- CheckSiretNumber
 - Gui::Widgets::CheckFields::CheckSiretNumber, 56
- CheckUntilField
 - Gui::Widgets::CheckFields::CheckUntilField, 57
- CheckValidField
 - Gui::Widgets::CheckFields::CheckValidField, 58
- columnCount
 - Gui::Widgets::WdgModels::BillingsTableModel, 39
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 70
 - Gui::Widgets::WdgModels::CustomersTableModel, 96
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 143
 - Gui::Widgets::WdgModels::ProjectsTableModel, 152
- ComboBoxDelegate
 - Gui::Widgets::Delegates::ComboBoxDelegate, 59
- ComboBoxModelWidget
 - Gui::Widgets::ComboBoxModelWidget, 61
- Comment
 - Mustache::Tag, 178
- Context
 - Mustache::Context, 64
- ContributoriesDatabaseTest, 65
- ContributoriesTableModel
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 70
- ContributoriesWidget
 - Gui::Widgets::ContributoriesWidget, 73
- Contributory
 - Models::Contributory, 76
- ContributoryListTest, 83
- ContributoryModelTest, 83
- count
 - Gui::Widgets::ContributoriesWidget, 74
 - Gui::Widgets::WdgModels::BillingsTableModel, 39
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 70
 - Gui::Widgets::WdgModels::CustomersTableModel, 96
 - Gui::Widgets::WdgModels::ProjectsTableModel, 152
- CounterContext, 83
 - canEval, 84
 - eval, 84
 - stringValue, 84, 85
- createEditor
 - Gui::Widgets::Delegates::ComboBoxDelegate, 60
 - Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 107
 - Gui::Widgets::Delegates::ProjectComboDelegate, 141
 - Gui::Widgets::Delegates::TextareaDelegate, 182
 - Gui::Widgets::Delegates::UnitComboDelegate, 184
- Customer
 - Models::Customer, 86
- CustomerContextualMenu
 - Gui::Widgets::CustomerContextualMenu, 88
- CustomerDataWidget
 - Gui::Widgets::CustomerDataWidget, 93
- CustomerDatabaseTest, 92
- CustomerModelTest, 93
- CustomersTableModel
 - Gui::Widgets::WdgModels::CustomersTableModel, 95
- DB_FILENAME
 - Parameters, 124
- data
 - Gui::Widgets::WdgModels::BillingsTableModel, 39
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 71
 - Gui::Widgets::WdgModels::CustomersTableModel, 96
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 144
 - Gui::Widgets::WdgModels::ProjectsTableModel, 152
- DatabaseSettingsWidget
 - Gui::Widgets::DatabaseSettingsWidget, 102
- databaseTypeChanged
 - Gui::Dialogs::StartedWindowsDialog, 175
- Databases, 17
- Databases::AccessDatabase, 21
- Databases::BillingDatabase, 31
 - addBilling, 32
 - addBillingProject, 32
 - getAllBillingsOfProject, 33
 - getAllBillingsOnly, 33
 - getBilling, 33
 - getBillings, 33
 - getBillingsBetweenDates, 34
 - getBillingsTable, 34
 - getMaxBillingNumber, 34
 - getMaxBillingNumberOfCustomer, 34
 - getMaxQuoteNumber, 35
 - getMaxQuoteNumberOfCustomer, 35
 - instance, 35
 - isBillingPaid, 35
 - removeBilling, 35
 - removeBillingProject, 36
- Databases::ContributoryDatabase, 78
 - addContributory, 79
 - getContributoriesByBilling, 79
 - getContributoriesByBillingAndProject, 81
 - getContributory, 81
 - instance, 81
 - removeContributory, 81
- Databases::CustomerDatabase, 88
 - addCustomer, 89
 - getCustomer, 89, 90
 - getCustomersTable, 90
 - getItemCustomer, 90
 - getItemProject, 90
 - getItemRoot, 91

- getNbCustomers, 91
 - getTree, 91
 - instance, 91
 - removeCustomer, 91
 - updateCustomer, 92
- Databases::Database, 98
 - changeDatabase, 99
 - executeFile, 99
 - instance, 99
 - lastError, 99
 - setDatabase, 101
 - value, 101
- Databases::ProjectDatabase, 146
 - addProject, 147
 - getAllProjects, 147
 - getCostProjects, 147
 - getNbProjects, 147
 - getNbProjectsForACustomer, 147
 - getProject, 148
 - getProjects, 148
 - getProjectsOfCustomer, 148
 - getProjectsTable, 148
 - instance, 150
 - removeProject, 150
 - updateProject, 150
- Databases::RateDatabase, 159
 - addRateProject, 160
 - getRate, 160
 - instance, 160
 - updateRateProject, 160
- Databases::UserDatabase, 189
 - getUser, 190
 - instance, 190
- DbException
 - Exceptions::DbException, 104
- DialogAddCustomer
 - Gui::Dialogs::DialogAddCustomer, 105
- DoubleSpinBoxDelegate
 - Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 107
- editUser
 - Gui::MainWindow, 121
- endDateControl
 - Gui::Dialogs::ComputeTurnoverDialog, 63
- error
 - Mustache::Renderer, 163
- errorPartial
 - Mustache::Renderer, 163
- errorPos
 - Mustache::Renderer, 163
- eval
 - CounterContext, 84
 - Mustache::Context, 64
 - Mustache::QtVariantContext, 156
- Exceptions::DbException, 103
 - DbException, 104
 - popupMessage, 104
- Exceptions::FileException, 109
 - FileException, 109
 - popupMessage, 110
- executeFile
 - Databases::Database, 99
- FileException
 - Exceptions::FileException, 109
- fillLabels
 - Gui::Dialogs::ComputeTurnoverDialog, 63
- fillQuoteBilling
 - Gui::Dialogs::AddQuoteDialog, 24
- filterOnBillsOrQuotes
 - Models::Search, 165
- filterOnCompany
 - Models::Search, 165
- filterOnContributories
 - Models::Search, 165
- filterOnCustomersWithoutProject
 - Models::Search, 166
- filterOnNumberElements
 - Models::Search, 166
- filterOnProjects
 - Models::Search, 166
- filterOnReferentLastname
 - Models::Search, 166
- filterOnVarcharElements
 - Models::Search, 166
- firstLetterToUpper
 - Utils::String, 177
- flags
 - Gui::Widgets::WdgModels::BillingsTableModel, 39
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 71
 - Gui::Widgets::WdgModels::CustomersTableModel, 96
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 144
 - Gui::Widgets::WdgModels::ProjectsTableModel, 153
- fn_t
 - Mustache::QtVariantContext, 156
- generate
 - Generator::PdfGenerator, 127
 - Generator::TexGenerator, 181
- Generation, 110
 - Generator::PdfGenerator, 127
 - generate, 127
 - PdfGenerator, 127
- Generator::TexGenerator, 179
 - generate, 181
 - TexGenerator, 180
- getAddress
 - Models::People, 129
- getAllBillingsOfProject
 - Databases::BillingDatabase, 33
- getAllBillingsOnly
 - Databases::BillingDatabase, 33
- getAllContributories

- Models::ContributoriesList, 67
- getAllProjects
 - Databases::ProjectDatabase, 147
- getBeginDate
 - Models::Project, 136
- getBilling
 - Databases::BillingDatabase, 33
- getBillings
 - Databases::BillingDatabase, 33
 - Gui::Widgets::WdgModels::BillingsTableModel, 39
- getBillingsBetweenDates
 - Databases::BillingDatabase, 34
- getBillingsTable
 - Databases::BillingDatabase, 34
- getBtnValid
 - Gui::Widgets::CheckFields::CheckQLineEdit, 55
- getCity
 - Models::People, 129
- getCompany
 - Models::People, 129
- getContributories
 - Gui::Widgets::ContributoriesWidget, 74
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 71
 - Models::Billing, 27
 - Models::ContributoriesList, 67
- getContributoriesByBilling
 - Databases::ContributoryDatabase, 79
- getContributoriesByBillingAndProject
 - Databases::ContributoryDatabase, 81
- getContributory
 - Databases::ContributoryDatabase, 81
- getCopy
 - Gui::Dialogs::AddQuoteDialog, 24
- getCost
 - Models::Project, 136, 137
- getCostProjects
 - Databases::ProjectDatabase, 147
- getCountry
 - Gui::Widgets::CheckFields::CheckPhone, 50
 - Gui::Widgets::CheckFields::CheckPostalCode, 53
 - Models::People, 129
- getCurrentCustomerId
 - Gui::MainWindow, 121
 - Gui::Widgets::searchWidget, 173
- getCurrentCustomerName
 - Gui::MainWindow, 121
- getCurrentProjectId
 - Gui::MainWindow, 121
- getCurrentProjectName
 - Gui::MainWindow, 121
- getCurrentQuoteId
 - Gui::MainWindow, 121
- getCustomer
 - Databases::CustomerDatabase, 89, 90
 - Models::ContributoriesList, 67
 - Models::Project, 137
- getCustomers
 - Gui::Widgets::WdgModels::CustomersTableModel, 96
 - Utils::HierarchicalSystem, 111
- getCustomersTable
 - Databases::CustomerDatabase, 90
- getDailyRate
 - Gui::Widgets::RateWidget, 162
 - Models::Project, 137
 - Models::Rate, 158
- getDataMap
 - Models::Billing, 27
 - Models::ContributoriesList, 67
 - Models::Contributory, 77
 - Models::Customer, 86
 - Models::IModel, 114
 - Models::Project, 137
 - Models::User, 187
- getDatabaseName
 - Gui::Widgets::DatabaseSettingsWidget, 102
- getDate
 - Models::Billing, 27
- getDescription
 - Models::Billing, 27
 - Models::Contributory, 77
 - Models::Project, 137
- getDomainNameOrIP
 - Gui::Widgets::DatabaseSettingsWidget, 102
- getEmail
 - Models::People, 129
- getEndDate
 - Models::Project, 137
- getFax
 - Models::People, 130
- getFilename
 - Models::Billing, 28
- getFilter
 - Models::Search, 166
- getFirstname
 - Models::People, 130
- getFolder
 - Models::Billing, 28
- getGroupFilter
 - Models::Search, 167
- getHourlyRate
 - Gui::Widgets::RateWidget, 162
 - Models::Rate, 158
- getId
 - Models::IModel, 114
- getIdBilling
 - Models::ContributoriesList, 67
- getIdCustomer
 - Gui::Dialogs::AddQuoteDialog, 24
 - Gui::Widgets::searchWidget, 173
- getImage
 - Gui::Dialogs::StartedWindowsDialog, 175
- getItem
 - Models::Billing, 28
- getItemCustomer

- Databases::CustomerDatabase, 90
- getItemProject
 - Databases::CustomerDatabase, 90
- getItemRoot
 - Databases::CustomerDatabase, 91
- getLastName
 - Models::People, 130
- getLegalRate
 - Models::Rate, 159
- getLogin
 - Gui::Widgets::DatabaseSettingsWidget, 103
- getMaxBillingNumber
 - Databases::BillingDatabase, 34
- getMaxBillingNumberOfCustomer
 - Databases::BillingDatabase, 34
- getMaxQuoteNumber
 - Databases::BillingDatabase, 35
- getMaxQuoteNumberOfCustomer
 - Databases::BillingDatabase, 35
- getMobilePhone
 - Models::People, 130
- getModel
 - Utils::ItemType, 115
- getName
 - Models::Project, 138
 - Utils::ItemType, 116
- getNameFolder
 - Models::Customer, 86
- getNbCustomers
 - Databases::CustomerDatabase, 91
- getNbDailyHours
 - Models::Rate, 159
- getNbHours
 - Models::Contributory, 77
- getNbProjects
 - Databases::ProjectDatabase, 147
 - Models::ContributoriesList, 67
- getNbProjectsForACustomer
 - Databases::ProjectDatabase, 147
- getNoSiret
 - Models::User, 187
- getNumber
 - Gui::Dialogs::AddQuoteDialog, 24
 - Models::Billing, 28
- getPartial
 - Mustache::PartialFileLoader, 125
 - Mustache::PartialMap, 126
 - Mustache::PartialResolver, 126
- getPassword
 - Gui::Widgets::DatabaseSettingsWidget, 103
- getPath
 - Models::Billing, 28
 - Models::Customer, 86
- getPhone
 - Models::People, 130
- getPort
 - Gui::Widgets::DatabaseSettingsWidget, 103
- getPostalCode
 - Models::People, 130
- getProject
 - Databases::ProjectDatabase, 148
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 144
 - Models::Contributory, 77
- getProjects
 - Databases::ProjectDatabase, 148
 - Gui::Widgets::Delegates::ProjectComboDelegate, 141
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 144
 - Gui::Widgets::WdgModels::ProjectsTableModel, 153
 - Models::ContributoriesList, 68
 - Utils::HierarchicalSystem, 111
- getProjectsOfCustomer
 - Databases::ProjectDatabase, 148
- getProjectsTable
 - Databases::ProjectDatabase, 148
- getRate
 - Databases::RateDatabase, 160
 - Models::ContributoriesList, 68
- getSearchInBillsQuotes
 - Models::Search, 167
- getSearchInCompanies
 - Models::Search, 167
- getSearchInProjects
 - Models::Search, 167
- getSearchInReferentLastname
 - Models::Search, 167
- getSelectedProjects
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 144
- getSumQuantity
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 71
 - Models::Billing, 28
 - Models::ContributoriesList, 68
- getSumRate
 - Models::Billing, 28
 - Models::ContributoriesList, 68
- getText
 - Models::Search, 167
- getTitle
 - Models::Billing, 29
 - Models::User, 187
- getTree
 - Databases::CustomerDatabase, 91
- getTurnover
 - Models::Customer, 86
- getTurnoverBetweenDates
 - Models::Statistics, 176
- getType
 - Utils::ItemType, 116
- getUser
 - Databases::UserDatabase, 190
- getWorkspaceName

- Models::User, 187
- getWorkspacePath
 - Models::User, 187
- Gui, 17
- Gui::Dialogs::AddProjectDialog, 21
 - AddProjectDialog, 22
- Gui::Dialogs::AddQuoteDialog, 22
 - AddQuoteDialog, 23
 - fillQuoteBilling, 24
 - getCopy, 24
 - getCustomerId, 24
 - getNumber, 24
 - setCopy, 24
 - setQuoteIdNumber, 25
- Gui::Dialogs::ComputeTurnoverDialog, 62
 - beginDateControl, 62
 - endDateControl, 63
 - fillLabels, 63
- Gui::Dialogs::DialogAddCustomer, 105
 - DialogAddCustomer, 105
- Gui::Dialogs::MessageBox, 122
 - MessageBox, 123
 - setImage, 123
 - setText, 124
- Gui::Dialogs::StartedWindowsDialog, 174
 - databaseTypeChanged, 175
 - getImage, 175
 - isDatabaseCentralized, 175
 - isDatabaseTypeValid, 175
 - StartedWindowsDialog, 175
- Gui::Dialogs::UserDataDialog, 191
 - UserDataDialog, 191
- Gui::Docks::SearchDock, 169
 - search, 171
 - SearchDock, 170
 - textChanged, 171
- Gui::MainWindow, 118
 - addBill, 120
 - addDoc, 120
 - addProject, 120
 - addQuote, 121
 - editUser, 121
 - getCurrentCustomerId, 121
 - getCurrentCustomerName, 121
 - getCurrentProjectId, 121
 - getCurrentProjectName, 121
 - getCurrentQuoteId, 121
 - MainWindow, 120
 - resizeEvent, 122
 - rootTree, 122
 - search, 122
 - treeLevel, 122
- Gui::Widgets, 18
- Gui::Widgets::CheckFields::CheckCity, 41
 - CheckCity, 41
- Gui::Widgets::CheckFields::CheckCountry, 41
 - CheckCountry, 42
- Gui::Widgets::CheckFields::CheckEmail, 42
 - check, 43
 - CheckEmail, 43
- Gui::Widgets::CheckFields::CheckFieldsLetters, 44
 - check, 44
 - CheckFieldsLetters, 44
- Gui::Widgets::CheckFields::CheckFieldsNumbers, 45
 - check, 46
 - CheckFieldsNumbers, 45
- Gui::Widgets::CheckFields::CheckIpAddress, 46
 - check, 47
 - CheckIpAddress, 47
- Gui::Widgets::CheckFields::CheckLogin, 47
 - check, 48
 - CheckLogin, 48
- Gui::Widgets::CheckFields::CheckName, 49
 - CheckName, 49
- Gui::Widgets::CheckFields::CheckPhone, 49
 - check, 50
 - CheckPhone, 50
 - getCountry, 50
 - setCountry, 51
- Gui::Widgets::CheckFields::CheckPortNumber, 51
 - check, 52
 - CheckPortNumber, 52
- Gui::Widgets::CheckFields::CheckPostalCode, 52
 - check, 53
 - CheckPostalCode, 53
 - getCountry, 53
 - setCountry, 53
- Gui::Widgets::CheckFields::CheckQLineEdit, 54
 - CheckQLineEdit, 54
 - getBtnValid, 55
 - isValid, 55
 - setBtnValid, 55
- Gui::Widgets::CheckFields::CheckSiretNumber, 55
 - check, 56
 - CheckSiretNumber, 56
- Gui::Widgets::CheckFields::CheckUntilField, 56
 - check, 57
 - CheckUntilField, 57
- Gui::Widgets::CheckFields::CheckValidField, 57
 - check, 58
 - CheckValidField, 58
- Gui::Widgets::CheckFields::ICheckField, 112
 - check, 112
- Gui::Widgets::ComboBoxModelWidget, 61
 - ComboBoxModelWidget, 61
- Gui::Widgets::ContributoriesWidget, 72
 - add, 74
 - addProject, 74
 - ContributoriesWidget, 73
 - count, 74
 - getContributories, 74
- Gui::Widgets::CustomerContextualMenu, 87
 - CustomerContextualMenu, 88
- Gui::Widgets::CustomerDataWidget, 92
 - CustomerDataWidget, 93
 - printInformations, 93

- Gui::Widgets::DatabaseSettingsWidget, 101
 - DatabaseSettingsWidget, 102
 - getDatabaseName, 102
 - getDomainNameOrIP, 102
 - getLogin, 103
 - getPassword, 103
 - getPort, 103
 - isValid, 103
- Gui::Widgets::Delegates::ComboBoxDelegate, 59
 - ComboBoxDelegate, 59
 - createEditor, 60
 - paint, 60
 - setEditorData, 60
 - setModelData, 60
 - updateEditorGeometry, 61
- Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 106
 - createEditor, 107
 - DoubleSpinBoxDelegate, 107
 - setEditorData, 107
 - setModelData, 107
 - updateEditorGeometry, 109
- Gui::Widgets::Delegates::ProjectComboDelegate, 140
 - createEditor, 141
 - getProjects, 141
 - isLocked, 141
 - paint, 141
 - ProjectComboDelegate, 141
 - setLocked, 142
- Gui::Widgets::Delegates::TextareaDelegate, 181
 - createEditor, 182
 - setEditorData, 182
 - setModelData, 182
 - TextareaDelegate, 182
 - updateEditorGeometry, 182
- Gui::Widgets::Delegates::UnitComboDelegate, 183
 - createEditor, 184
 - paint, 184
 - UnitComboDelegate, 183
- Gui::Widgets::Popup, 134
- Gui::Widgets::ProjectsWidget, 154
 - ProjectsWidget, 155
 - updateBtn, 155
- Gui::Widgets::RateWidget, 161
 - getDailyRate, 162
 - getHourlyRate, 162
 - RateWidget, 162
 - setDailyRate, 162
 - setHourlyRate, 162
 - setWidgetDailyRateValue, 162
 - setWidgetHourlyRateValue, 163
- Gui::Widgets::WdgModels::BillingsTableModel, 37
 - append, 39
 - BillingsTableModel, 38
 - columnCount, 39
 - count, 39
 - data, 39
 - flags, 39
 - getBillings, 39
 - headerData, 40
 - remove, 40
 - rowCount, 40
 - setData, 40
- Gui::Widgets::WdgModels::ContributoriesTableModel, 69
 - append, 70
 - columnCount, 70
 - ContributoriesTableModel, 70
 - count, 70
 - data, 71
 - flags, 71
 - getContributories, 71
 - getSumQuantity, 71
 - headerData, 71
 - remove, 72
 - rowCount, 72
 - setData, 72
- Gui::Widgets::WdgModels::CustomersTableModel, 94
 - append, 96
 - columnCount, 96
 - count, 96
 - CustomersTableModel, 95
 - data, 96
 - flags, 96
 - getCustomers, 96
 - headerData, 97
 - remove, 97
 - rowCount, 97
 - setData, 97
- Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 142
 - allProjectsChosen, 143
 - append, 143
 - columnCount, 143
 - data, 144
 - flags, 144
 - getProject, 144
 - getProjects, 144
 - getSelectedProjects, 144
 - headerData, 145
 - ProjectContributoriesTableModel, 143
 - remove, 145
 - rowCount, 145
 - setData, 145
- Gui::Widgets::WdgModels::ProjectsTableModel, 151
 - append, 152
 - columnCount, 152
 - count, 152
 - data, 152
 - flags, 153
 - getProjects, 153
 - headerData, 153
 - ProjectsTableModel, 152
 - remove, 153
 - rowCount, 153
 - setData, 154
- Gui::Widgets::searchWidget, 171

- getCurrentCustomerId, 173
 - getCustomerId, 173
 - isCustomerSelected, 173
 - search, 173
 - searchWidget, 172
 - selectCustomer, 173
 - setCustomerId, 173
- headerData
 - Gui::Widgets::WdgModels::BillingsTableModel, 40
 - Gui::Widgets::WdgModels::ContributoriesTableModel, 71
 - Gui::Widgets::WdgModels::CustomersTableModel, 97
 - Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 145
 - Gui::Widgets::WdgModels::ProjectsTableModel, 153
- hydrat
 - Models::Billing, 29
 - Models::Contributory, 77
 - Models::Customer, 87
 - Models::IModel, 114
 - Models::Project, 138
 - Models::User, 187
- instance
 - Databases::BillingDatabase, 35
 - Databases::ContributoryDatabase, 81
 - Databases::CustomerDatabase, 91
 - Databases::Database, 99
 - Databases::ProjectDatabase, 150
 - Databases::RateDatabase, 160
 - Databases::UserDatabase, 190
 - Utils::Log, 117
- InvertedSectionStart
 - Mustache::Tag, 178
- isArchived
 - Models::Customer, 87
- isBilling
 - Models::Billing, 29
- isBillingPaid
 - Databases::BillingDatabase, 35
- isCustomerSelected
 - Gui::Widgets::searchWidget, 173
- isDatabaseCentralized
 - Gui::Dialogs::StartedWindowsDialog, 175
- isDatabaseTypeValid
 - Gui::Dialogs::StartedWindowsDialog, 175
- isFalse
 - Mustache::Context, 64
 - Mustache::QtVariantContext, 157
- isInsert
 - Models::ContributoriesList, 68
- isLocked
 - Gui::Widgets::Delegates::ProjectComboDelegate, 141
- isPaid
 - Models::Billing, 29
- isToRemoved
 - Models::IModel, 114
- isValid
 - Gui::Widgets::CheckFields::CheckQLineEdit, 55
 - Gui::Widgets::DatabaseSettingsWidget, 103
- ItemType
 - Utils::ItemType, 115
- ItemTypeTest, 116
- lastError
 - Databases::Database, 99
- listCount
 - Mustache::Context, 64
 - Mustache::QtVariantContext, 157
- MainWindow
 - Gui::MainWindow, 120
- makeDirectory
 - Utils::Directories, 106
- MessageBox
 - Gui::Dialogs::MessageBox, 123
- Models, 18
- Models::Billing, 25
 - addContributory, 27
 - Billing, 27
 - getContributories, 27
 - getDataMap, 27
 - getDate, 27
 - getDescription, 27
 - getFilename, 28
 - getFolder, 28
 - getItem, 28
 - getNumber, 28
 - getPath, 28
 - getSumQuantity, 28
 - getSumRate, 28
 - getTitle, 29
 - hydrat, 29
 - isBilling, 29
 - isPaid, 29
 - operator<, 29
 - operator==, 30
 - setDate, 30
 - setDescription, 30
 - setIsBilling, 30
 - setIsPaid, 30
 - setNumber, 30
 - setTitle, 31
- Models::ContributoriesList, 65
 - addContributory, 66
 - addProject, 67
 - getAllContributories, 67
 - getContributories, 67
 - getCustomer, 67
 - getDataMap, 67
 - getCustomerId, 67
 - getNbProjects, 67
 - getProjects, 68
 - getRate, 68

- getSumQuantity, 68
- getSumRate, 68
- isInsert, 68
- setAllIdContributories, 68
- setIdBilling, 69
- setInsert, 69
- Models::Contributory, 74
 - Contributory, 76
 - getDataMap, 77
 - getDescription, 77
 - getNbHours, 77
 - getProject, 77
 - hydrat, 77
 - operator==, 78
 - setDescription, 78
 - setNbHours, 78
 - setProject, 78
- Models::Customer, 85
 - Customer, 86
 - getDataMap, 86
 - getNameFolder, 86
 - getPath, 86
 - getTurnover, 86
 - hydrat, 87
 - isArchived, 87
 - setIsArchived, 87
- Models::IModel, 113
 - getDataMap, 114
 - getId, 114
 - hydrat, 114
 - isToRemoved, 114
 - setId, 114
 - setToRemoved, 114
- Models::People, 127
 - getAddress, 129
 - getCity, 129
 - getCompany, 129
 - getCountry, 129
 - getEmail, 129
 - getFax, 130
 - getFirstname, 130
 - getLastName, 130
 - getMobilePhone, 130
 - getPhone, 130
 - getPostalCode, 130
 - operator==, 132
 - setAddress, 132
 - setCity, 132
 - setCompany, 132
 - setCountry, 132
 - setEmail, 132
 - setFax, 133
 - setFirstname, 133
 - setLastName, 133
 - setMobilePhone, 133
 - setPhone, 133
 - setPostalCode, 133
- Models::Project, 135
 - getBeginDate, 136
 - getCost, 136, 137
 - getCustomer, 137
 - getDailyRate, 137
 - getDataMap, 137
 - getDescription, 137
 - getEndDate, 137
 - getName, 138
 - hydrat, 138
 - operator<, 138
 - operator==, 138
 - Project, 136
 - setBeginDate, 139
 - setCost, 139
 - setCustomer, 139
 - setDailyRate, 139
 - setDescription, 139
 - setEndDate, 139
 - setName, 140
- Models::Rate, 157
 - getDailyRate, 158
 - getHourlyRate, 158
 - getLegalRate, 159
 - getNbDailyHours, 159
 - Rate, 158
- Models::Search, 164
 - filterOnBillsOrQuotes, 165
 - filterOnCompany, 165
 - filterOnContributories, 165
 - filterOnCustomersWithoutProject, 166
 - filterOnNumberElements, 166
 - filterOnProjects, 166
 - filterOnReferentLastname, 166
 - filterOnVarcharElements, 166
 - getFilter, 166
 - getGroupFilter, 167
 - getSearchInBillsQuotes, 167
 - getSearchInCompanies, 167
 - getSearchInProjects, 167
 - getSearchInReferentLastname, 167
 - getText, 167
 - searchInContributories, 167
 - setGroupFilter, 168
 - setSearchInBillsQuotes, 168
 - setSearchInCompanies, 168
 - setSearchInContributories, 168
 - setSearchInProjects, 168
 - setSearchInReferentLastname, 168
 - setText, 169
- Models::Statistics, 176
 - getTurnoverBetweenDates, 176
- Models::User, 184
 - getDataMap, 187
 - getNoSiret, 187
 - getTitle, 187
 - getWorkspaceName, 187
 - getWorkspacePath, 187
 - hydrat, 187

- operator==, 188
- setNoSiret, 188
- setTitle, 188
- setWorkspaceName, 188
- setWorkspacePath, 189
- updateFolders, 189
- User, 186
- Mustache::Tag
 - Comment, 178
 - InvertedSectionStart, 178
 - Partial, 178
 - SectionEnd, 178
 - SectionStart, 178
 - SetDelimiter, 178
- Mustache::Context, 63
 - canEval, 64
 - Context, 64
 - eval, 64
 - isFalse, 64
 - listCount, 64
 - partialResolver, 64
 - partialValue, 64
 - pop, 64
 - push, 65
 - stringValue, 65
- Mustache::PartialFileLoader, 125
 - getPartial, 125
- Mustache::PartialMap, 125
 - getPartial, 126
- Mustache::PartialResolver, 126
 - getPartial, 126
- Mustache::QtVariantContext, 155
 - canEval, 156
 - eval, 156
 - fn_t, 156
 - isFalse, 157
 - listCount, 157
 - pop, 157
 - push, 157
 - stringValue, 157
- Mustache::Renderer, 163
 - error, 163
 - errorPartial, 163
 - errorPos, 163
 - render, 163
 - setTagMarkers, 164
- Mustache::Tag, 177
 - Type, 178
- operator<
 - Models::Billing, 29
 - Models::Project, 138
- operator<<
 - Utils::Log, 118
- operator==
 - Models::Billing, 30
 - Models::Contributory, 78
 - Models::People, 132
 - Models::Project, 138
 - Models::User, 188
- paint
 - Gui::Widgets::Delegates::ComboBoxDelegate, 60
 - Gui::Widgets::Delegates::ProjectComboDelegate, 141
 - Gui::Widgets::Delegates::UnitComboDelegate, 184
- Parameters, 124
 - DB_FILENAME, 124
 - VERSION, 124
- Partial
 - Mustache::Tag, 178
- partialResolver
 - Mustache::Context, 64
- partialValue
 - Mustache::Context, 64
- PdfGenerator
 - Generator::PdfGenerator, 127
- PointersTest, 134
- pop
 - Mustache::Context, 64
 - Mustache::QtVariantContext, 157
- popupMessage
 - Exceptions::DbException, 104
 - Exceptions::FileException, 110
- printInformations
 - Gui::Widgets::CustomerDataWidget, 93
- Project
 - Models::Project, 136
- ProjectComboDelegate
 - Gui::Widgets::Delegates::ProjectComboDelegate, 141
- ProjectContributoriesTableModel
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 143
- ProjectDatabaseTest, 150
- ProjectModelTest, 151
- ProjectsTableModel
 - Gui::Widgets::WdgModels::ProjectsTableModel, 152
- ProjectsWidget
 - Gui::Widgets::ProjectsWidget, 155
- push
 - Mustache::Context, 65
 - Mustache::QtVariantContext, 157
- Rate
 - Models::Rate, 158
- RateModelTest, 161
- RateWidget
 - Gui::Widgets::RateWidget, 162
- remove
 - Gui::Widgets::WdgModels::BillingsTableModel, 40
 - Gui::Widgets::WdgModels::ContributoriesTable-Model, 72
 - Gui::Widgets::WdgModels::CustomersTableModel, 97
 - Gui::Widgets::WdgModels::ProjectContributories-TableModel, 145

- Gui::Widgets::WdgModels::ProjectsTableModel, 153
- removeBilling
 - Databases::BillingDatabase, 35
- removeBillingProject
 - Databases::BillingDatabase, 36
- removeContributory
 - Databases::ContributoryDatabase, 81
- removeCustomer
 - Databases::CustomerDatabase, 91
- removeProject
 - Databases::ProjectDatabase, 150
- render
 - Mustache::Renderer, 163
- resizeEvent
 - Gui::MainWindow, 122
- rootTree
 - Gui::MainWindow, 122
- rowCount
 - Gui::Widgets::WdgModels::BillingsTableModel, 40
 - Gui::Widgets::WdgModels::ContributoriesTableModel, 72
 - Gui::Widgets::WdgModels::CustomersTableModel, 97
 - Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 145
 - Gui::Widgets::WdgModels::ProjectsTableModel, 153
- search
 - Gui::Docks::SearchDock, 171
 - Gui::MainWindow, 122
 - Gui::Widgets::searchWidget, 173
- SearchDock
 - Gui::Docks::SearchDock, 170
- searchInContributories
 - Models::Search, 167
- searchTest, 171
- searchWidget
 - Gui::Widgets::searchWidget, 172
- SectionEnd
 - Mustache::Tag, 178
- SectionStart
 - Mustache::Tag, 178
- selectCustomer
 - Gui::Widgets::searchWidget, 173
- SetDelimiter
 - Mustache::Tag, 178
- setAddress
 - Models::People, 132
- setAllIdContributories
 - Models::ContributoriesList, 68
- setBeginDate
 - Models::Project, 139
- setBtnValid
 - Gui::Widgets::CheckFields::CheckQLineEdit, 55
- setCity
 - Models::People, 132
- setCompany
 - Models::People, 132
- setCopy
 - Gui::Dialogs::AddQuoteDialog, 24
- setCost
 - Models::Project, 139
- setCountry
 - Gui::Widgets::CheckFields::CheckPhone, 51
 - Gui::Widgets::CheckFields::CheckPostalCode, 53
 - Models::People, 132
- setCustomer
 - Models::Project, 139
- setDailyRate
 - Gui::Widgets::RateWidget, 162
 - Models::Project, 139
- setData
 - Gui::Widgets::WdgModels::BillingsTableModel, 40
 - Gui::Widgets::WdgModels::ContributoriesTableModel, 72
 - Gui::Widgets::WdgModels::CustomersTableModel, 97
 - Gui::Widgets::WdgModels::ProjectContributoriesTableModel, 145
 - Gui::Widgets::WdgModels::ProjectsTableModel, 154
- setDatabase
 - Databases::Database, 101
- setDate
 - Models::Billing, 30
- setDescription
 - Models::Billing, 30
 - Models::Contributory, 78
 - Models::Project, 139
- setEditorData
 - Gui::Widgets::Delegates::ComboBoxDelegate, 60
 - Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 107
 - Gui::Widgets::Delegates::TextareaDelegate, 182
- setEmail
 - Models::People, 132
- setEndDate
 - Models::Project, 139
- setFax
 - Models::People, 133
- setFirstname
 - Models::People, 133
- setGroupFilter
 - Models::Search, 168
- setHourlyRate
 - Gui::Widgets::RateWidget, 162
- setId
 - Models::IModel, 114
- setIdBilling
 - Models::ContributoriesList, 69
- setIdCustomer
 - Gui::Widgets::searchWidget, 173
- setImage
 - Gui::Dialogs::MessageBox, 123
- setInsert

- Models::ContributoriesList, 69
- setIsArchived
 - Models::Customer, 87
- setIsBilling
 - Models::Billing, 30
- setIsPaid
 - Models::Billing, 30
- setLastname
 - Models::People, 133
- setLocked
 - Gui::Widgets::Delegates::ProjectComboDelegate, 142
- setMobilePhone
 - Models::People, 133
- setModelData
 - Gui::Widgets::Delegates::ComboBoxDelegate, 60
 - Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 107
 - Gui::Widgets::Delegates::TextareaDelegate, 182
- setName
 - Models::Project, 140
 - Utils::ItemType, 116
- setNbHours
 - Models::Contributory, 78
- setNoSiret
 - Models::User, 188
- setNumber
 - Models::Billing, 30
- setPhone
 - Models::People, 133
- setPostalCode
 - Models::People, 133
- setProject
 - Models::Contributory, 78
- setQuoteIdNumber
 - Gui::Dialogs::AddQuoteDialog, 25
- setSearchInBillsQuotes
 - Models::Search, 168
- setSearchInCompanies
 - Models::Search, 168
- setSearchInContributories
 - Models::Search, 168
- setSearchInProjects
 - Models::Search, 168
- setSearchInReferentLastname
 - Models::Search, 168
- setTagMarkers
 - Mustache::Renderer, 164
- setText
 - Gui::Dialogs::MessageBox, 124
 - Models::Search, 169
- setTitle
 - Models::Billing, 31
 - Models::User, 188
- setToRemoved
 - Models::IModel, 114
- setType
 - Utils::ItemType, 116
- setWidgetDailyRateValue
 - Gui::Widgets::RateWidget, 162
- setWidgetHourlyRateValue
 - Gui::Widgets::RateWidget, 163
- setWorkspaceName
 - Models::User, 188
- setWorkspacePath
 - Models::User, 189
- StartedWindowsDialog
 - Gui::Dialogs::StartedWindowsDialog, 175
- StringTest, 177
- stringValue
 - CounterContext, 84, 85
 - Mustache::Context, 65
 - Mustache::QtVariantContext, 157
- TestAdder< T >, 178
- TestMustache, 179
- TestRunner, 179
- testadder, 178
- TexGenerator
 - Generator::TexGenerator, 180
- textChanged
 - Gui::Docks::SearchDock, 171
- TextareaDelegate
 - Gui::Widgets::Delegates::TextareaDelegate, 182
- treeLevel
 - Gui::MainWindow, 122
- Type
 - Mustache::Tag, 178
- UnitComboDelegate
 - Gui::Widgets::Delegates::UnitComboDelegate, 183
- updateBtn
 - Gui::Widgets::ProjectsWidget, 155
- updateCustomer
 - Databases::CustomerDatabase, 92
- updateEditorGeometry
 - Gui::Widgets::Delegates::ComboBoxDelegate, 61
 - Gui::Widgets::Delegates::DoubleSpinBoxDelegate, 109
 - Gui::Widgets::Delegates::TextareaDelegate, 182
- updateFolders
 - Models::User, 189
- updateProject
 - Databases::ProjectDatabase, 150
- updateRateProject
 - Databases::RateDatabase, 160
- User
 - Models::User, 186
- UserDataDialog
 - Gui::Dialogs::UserDataDialog, 191
- UserDatabaseTest, 190
- UserModelTest, 192
- Utils::Directories, 106
 - makeDirectory, 106
- Utils::HierarchicalSystem, 110
 - addBillingToProject, 111
 - addProjectToCustomer, 111

- getCustomers, [111](#)
- getProjects, [111](#)
- Utils::ItemType, [115](#)
 - getModel, [115](#)
 - getName, [116](#)
 - getType, [116](#)
 - ItemType, [115](#)
 - setName, [116](#)
 - setType, [116](#)
- Utils::Log, [117](#)
 - instance, [117](#)
 - operator<<, [118](#)
 - write, [117](#)
- Utils::String, [176](#)
 - firstLetterToUpper, [177](#)
- Utils::pointers, [134](#)
- VERSION
 - Parameters, [124](#)
- value
 - Databases::Database, [101](#)
- write
 - Utils::Log, [117](#)