

iLearnRW

0.0.1

Generated by Doxygen 1.8.4

Tue Aug 27 2013 15:25:20

Contents

1	Main Page	1
2	Doxygen	3
3	DataLogger Log format	5
4	Todo List	7
5	Module Documentation	9
5.1	DolphinTodo_20130619	9
5.2	OnlineResourceBankTodo	10
6	Namespace Documentation	11
6.1	Package ilearnrw	11
6.2	Package ilearnrw.application	11
6.3	Package ilearnrw.datalogger	11
6.3.1	Detailed Description	12
6.4	Package ilearnrw.games	12
6.5	Package ilearnrw.online	12
6.6	Package ilearnrw.online.resourcebank	12
6.6.1	Detailed Description	12
6.7	Package ilearnrw.prototype	15
6.8	Package ilearnrw.prototype.application	15
6.9	Package ilearnrw.user	16
6.10	Package ilearnrw.user.mockup	16
6.11	Package ilearnrw.user.problems	16
6.12	Package ilearnrw.user.test	16
7	Class Documentation	17
7.1	ilearnrw.application.ApplicationId Class Reference	17
7.1.1	Detailed Description	18
7.1.2	Constructor & Destructor Documentation	18
7.1.2.1	ApplicationId	18
7.1.3	Member Function Documentation	18

7.1.3.1	equals	18
7.1.3.2	getId	19
7.1.4	Member Data Documentation	19
7.1.4.1	mId	19
7.1.4.2	serialVersionUID	19
7.2	ilearnrw.datalogger.DataLogger Class Reference	19
7.2.1	Detailed Description	21
7.2.2	Member Function Documentation	21
7.2.2.1	addUserProblem	21
7.2.2.2	discardPendingChanges	21
7.2.2.3	getActions	21
7.2.2.4	getUser	21
7.2.2.5	getUserAdministration	22
7.2.2.6	hasPendingChanges	22
7.2.2.7	loadUserActions	22
7.2.2.8	loadUserStore	22
7.2.2.9	logAction	22
7.2.2.10	moveToOldSessions	23
7.2.2.11	registerUserSession	23
7.2.2.12	removeUserProblem	23
7.2.2.13	setCurrentUser	23
7.2.2.14	setDirty	23
7.2.2.15	setPreference	23
7.2.2.16	setUserDetail	23
7.2.2.17	setUserLanguage	23
7.2.2.18	writePendingChanges	24
7.2.3	Member Data Documentation	24
7.2.3.1	mMarker	24
7.2.3.2	mUser	24
7.2.3.3	mUserActions	24
7.2.3.4	mUserIsDirty	24
7.2.3.5	mUserStore	24
7.3	ilearnrw.datalogger.IDataLogger Interface Reference	24
7.3.1	Detailed Description	26
7.3.2	Member Function Documentation	26
7.3.2.1	getActions	26
7.3.2.2	logAction	26
7.4	ilearnrw.datalogger.ILoginProvider Interface Reference	27
7.4.1	Member Function Documentation	28
7.4.1.1	getUser	28

7.4.1.2	getUserAdministration	28
7.5	ilearnrw.datalogger.IProfileAccessUpdater Interface Reference	28
7.5.1	Detailed Description	31
7.5.2	Member Function Documentation	31
7.5.2.1	addUserProblem	31
7.5.2.2	discardPendingChanges	31
7.5.2.3	hasPendingChanges	31
7.5.2.4	moveToOldSessions	31
7.5.2.5	registerUserSession	31
7.5.2.6	removeUserProblem	32
7.5.2.7	setCurrentUser	32
7.5.2.8	setPreference	32
7.5.2.9	setUserDetail	32
7.5.2.10	setUserLanguage	32
7.5.2.11	writePendingChanges	32
7.6	ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable Class Reference	33
7.6.1	Detailed Description	34
7.6.2	Constructor & Destructor Documentation	34
7.6.2.1	PendingChangesAvailable	34
7.6.2.2	PendingChangesAvailable	34
7.6.3	Member Function Documentation	34
7.6.3.1	getMessage	34
7.6.4	Member Data Documentation	34
7.6.4.1	serialVersionUID	34
7.7	ilearnrw.datalogger.IProfileAccessUpdater.PermissionException Class Reference	34
7.7.1	Detailed Description	36
7.7.2	Constructor & Destructor Documentation	36
7.7.2.1	PermissionException	36
7.7.2.2	PermissionException	36
7.7.3	Member Function Documentation	36
7.7.3.1	getMessage	36
7.7.4	Member Data Documentation	36
7.7.4.1	serialVersionUID	36
7.8	ilearnrw.datalogger.IUserAdministration Interface Reference	36
7.8.1	Detailed Description	38
7.8.2	Member Function Documentation	38
7.8.2.1	authenticateAdmin	38
7.8.2.2	changeUsername	39
7.8.2.3	createUser	39
7.8.2.4	deleteUser	39

7.8.2.5	getAllUsers	39
7.8.2.6	isAuthenticated	39
7.8.2.7	updatePassword	39
7.9	ilearnrw.datalogger.IUserAdministration.AuthenticationException Class Reference	39
7.9.1	Detailed Description	41
7.9.2	Constructor & Destructor Documentation	41
7.9.2.1	AuthenticationException	41
7.9.2.2	AuthenticationException	41
7.9.3	Member Function Documentation	41
7.9.3.1	getMessage	41
7.9.4	Member Data Documentation	41
7.9.4.1	serialVersionUID	41
7.10	ilearnrw.datalogger.UserActionFilter Class Reference	42
7.10.1	Detailed Description	43
7.10.2	Constructor & Destructor Documentation	43
7.10.2.1	UserActionFilter	43
7.10.3	Member Function Documentation	43
7.10.3.1	getApplicationId	43
7.10.3.2	getEndTime	43
7.10.3.3	getStartTime	43
7.10.3.4	getTags	43
7.10.3.5	getUserId	43
7.10.4	Member Data Documentation	43
7.10.4.1	mApplicationId	43
7.10.4.2	mTags	43
7.10.4.3	mTimeEnd	43
7.10.4.4	mTimeStart	43
7.10.4.5	mUserId	43
7.11	ilearnrw.datalogger.UserStore Class Reference	43
7.11.1	Detailed Description	46
7.11.2	Constructor & Destructor Documentation	46
7.11.2.1	UserStore	46
7.11.3	Member Function Documentation	46
7.11.3.1	authenticateAdmin	46
7.11.3.2	changeUsername	47
7.11.3.3	checkAdminAuth	47
7.11.3.4	create	47
7.11.3.5	createUser	47
7.11.3.6	deepCopy	47
7.11.3.7	delete	47

7.11.3.8	deleteUser	48
7.11.3.9	getAllUsers	48
7.11.3.10	getUser	48
7.11.3.11	getUserAdministration	48
7.11.3.12	getUserIds	48
7.11.3.13	isAuthenticated	48
7.11.3.14	load	48
7.11.3.15	read	48
7.11.3.16	read	48
7.11.3.17	read_noCopy	48
7.11.3.18	save	48
7.11.3.19	update	49
7.11.3.20	updatePassword	49
7.11.4	Member Data Documentation	49
7.11.4.1	mFilePath	49
7.11.4.2	mHasAdminAuth	49
7.11.4.3	mLoadedUsers	49
7.11.4.4	mLogger	49
7.12	ilearnrw.games.GameAddTwoNumbers Class Reference	50
7.12.1	Constructor & Destructor Documentation	51
7.12.1.1	GameAddTwoNumbers	51
7.12.2	Member Function Documentation	51
7.12.2.1	printOutActions	51
7.12.2.2	update	51
7.12.3	Member Data Documentation	51
7.12.3.1	mApplicationId	51
7.12.3.2	mLogger	51
7.12.3.3	mUser	51
7.12.3.4	mUserActions	51
7.12.3.5	mUserId	51
7.12.3.6	mWelcomeMessage	51
7.13	ilearnrw.games.NumberProblems Class Reference	52
7.13.1	Constructor & Destructor Documentation	53
7.13.1.1	NumberProblems	53
7.13.2	Member Function Documentation	53
7.13.2.1	getProblem	53
7.13.2.2	initialize	53
7.14	ilearnrw.games.NumbersToText Class Reference	54
7.14.1	Constructor & Destructor Documentation	56
7.14.1.1	NumbersToText	56

7.14.2	Member Function Documentation	56
7.14.2.1	onSelected	56
7.14.2.2	printOutActions	56
7.14.3	Member Data Documentation	56
7.14.3.1	APPID	56
7.14.3.2	FAILED_TAG	56
7.14.3.3	mDataLogger	56
7.14.3.4	mUser	56
7.14.3.5	OK_TAG	56
7.15	ilearnrw.games.NumbersToTextHelper Class Reference	56
7.15.1	Detailed Description	57
7.15.2	Member Function Documentation	57
7.15.2.1	main	57
7.15.3	Member Data Documentation	57
7.15.3.1	processor	57
7.15.3.2	SCALE	57
7.15.3.3	SCALE_UNITS	57
7.16	ilearnrw.games.NumbersToTextHelper.AbstractProcessor Class Reference	58
7.16.1	Member Function Documentation	59
7.16.1.1	getDigits	59
7.16.1.2	getName	59
7.16.1.3	getName	59
7.16.1.4	getName	59
7.16.2	Member Data Documentation	59
7.16.2.1	NO_VALUE	59
7.16.2.2	SEPARATOR	59
7.17	ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor Class Reference	60
7.17.1	Constructor & Destructor Documentation	62
7.17.1.1	CompositeBigProcessor	62
7.17.2	Member Function Documentation	62
7.17.2.1	getHighProcessor	62
7.17.2.2	getLowProcessor	62
7.17.2.3	getName	62
7.17.2.4	getPartDivider	62
7.17.2.5	getToken	62
7.17.3	Member Data Documentation	62
7.17.3.1	exponent	62
7.17.3.2	hundredProcessor	62
7.17.3.3	lowProcessor	62
7.18	ilearnrw.games.NumbersToTextHelper.DefaultProcessor Class Reference	63

7.18.1	Member Function Documentation	64
7.18.1.1	getName	64
7.18.2	Member Data Documentation	65
7.18.2.1	MINUS	65
7.18.2.2	processor	65
7.18.2.3	UNION_AND	65
7.18.2.4	ZERO_TOKEN	65
7.19	ilearnrw.games.NumbersToTextHelper.HundredProcessor Class Reference	65
7.19.1	Member Function Documentation	67
7.19.1.1	getName	67
7.19.2	Member Data Documentation	67
7.19.2.1	EXPONENT	67
7.19.2.2	tensProcessor	67
7.19.2.3	unitProcessor	67
7.20	ilearnrw.games.NumbersToTextHelper.Scale Enum Reference	67
7.20.1	Member Function Documentation	67
7.20.1.1	getName	67
7.20.2	Member Data Documentation	67
7.20.2.1	SHORT	67
7.21	ilearnrw.games.NumbersToTextHelper.ScaleUnit Class Reference	68
7.21.1	Constructor & Destructor Documentation	68
7.21.1.1	ScaleUnit	68
7.21.2	Member Function Documentation	68
7.21.2.1	getExponent	68
7.21.2.2	getName	68
7.21.3	Member Data Documentation	68
7.21.3.1	exponent	68
7.21.3.2	names	68
7.22	ilearnrw.games.NumbersToTextHelper.TensProcessor Class Reference	69
7.22.1	Member Function Documentation	71
7.22.1.1	getName	71
7.22.2	Member Data Documentation	71
7.22.2.1	TOKENS	71
7.22.2.2	UNION_SEPARATOR	71
7.22.2.3	unitProcessor	71
7.23	ilearnrw.games.NumbersToTextHelper.UnitProcessor Class Reference	71
7.23.1	Member Function Documentation	72
7.23.1.1	getName	72
7.23.2	Member Data Documentation	72
7.23.2.1	TOKENS	72

7.24	ilearnrw.games.UsersProvider Class Reference	73
7.24.1	Member Function Documentation	73
7.24.1.1	main	73
7.25	ilearnrw.online.resourcebank.IResource Interface Reference	73
7.26	ilearnrw.online.resourcebank.IResourceBank Interface Reference	74
7.27	ilearnrw.online.resourcebank.IResourceCollection Interface Reference	74
7.28	ilearnrw.online.resourcebank.ITeachingStrategy Interface Reference	74
7.29	ilearnrw.prototype.application.ConsoleMenu Class Reference	75
7.29.1	Constructor & Destructor Documentation	75
7.29.1.1	ConsoleMenu	75
7.29.2	Member Function Documentation	76
7.29.2.1	doModalMenu	76
7.29.2.2	in	76
7.29.2.3	out	76
7.29.2.4	showMenu	76
7.29.2.5	subMenu	76
7.29.3	Member Data Documentation	76
7.29.3.1	mIn	76
7.29.3.2	mMenuActions	76
7.29.3.3	mMenuTitle	76
7.29.3.4	mOut	76
7.29.3.5	mScanner	76
7.30	ilearnrw.prototype.application.ConsoleMenu.EConsoleMenuActionResult Enum Reference	76
7.30.1	Member Data Documentation	76
7.30.1.1	menuBreak	76
7.30.1.2	showThisMenuAgain	77
7.31	ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction Interface Reference	77
7.31.1	Member Function Documentation	77
7.31.1.1	getText	77
7.31.1.2	onSelected	77
7.32	ilearnrw.prototype.application.ConsoleMenuAction Class Reference	78
7.32.1	Constructor & Destructor Documentation	79
7.32.1.1	ConsoleMenuAction	79
7.32.2	Member Function Documentation	79
7.32.2.1	getText	79
7.32.2.2	onSelected	79
7.32.3	Member Data Documentation	79
7.32.3.1	mText	79
7.33	ilearnrw.prototype.application.DatabaseManager Class Reference	80
7.33.1	Constructor & Destructor Documentation	81

7.33.1.1 DatabaseManager	81
7.33.2 Member Function Documentation	81
7.33.2.1 onSelected	82
7.34 ilearnrw.prototype.application.DatabaseManager.ISelectUser Interface Reference	82
7.34.1 Member Function Documentation	82
7.34.1.1 selectUser	82
7.35 ilearnrw.prototype.application.Program Class Reference	83
7.35.1 Member Function Documentation	83
7.35.1.1 getDataLogger	83
7.35.1.2 getLoginProvider	83
7.35.1.3 getProfileAccessUpdater	84
7.35.1.4 getStringArg	84
7.35.1.5 main	84
7.35.2 Member Data Documentation	84
7.35.2.1 sDataLogger	84
7.35.2.2 sWelcomeMessage	84
7.36 ilearnrw.prototype.application.UserActions Class Reference	85
7.36.1 Detailed Description	86
7.36.2 Constructor & Destructor Documentation	86
7.36.2.1 UserActions	86
7.36.3 Member Function Documentation	86
7.36.3.1 addUserAction	86
7.36.3.2 getAction	86
7.36.3.3 getActions	86
7.36.3.4 getActions	86
7.36.3.5 load	86
7.36.3.6 save	86
7.36.4 Member Data Documentation	86
7.36.4.1 mActions	86
7.36.4.2 mFilePath	86
7.36.4.3 mLogger	86
7.37 ilearnrw.prototype.application.UserActionsLog Class Reference	87
7.37.1 Constructor & Destructor Documentation	88
7.37.1.1 UserActionsLog	88
7.37.2 Member Function Documentation	88
7.37.2.1 onSelected	89
7.37.2.2 printOutActions	89
7.38 ilearnrw.prototype.application.UserGames Class Reference	89
7.38.1 Constructor & Destructor Documentation	90
7.38.1.1 UserGames	90

7.38.2	Member Function Documentation	90
7.38.2.1	onSelected	91
7.38.3	Member Data Documentation	91
7.38.3.1	mUser	91
7.39	ilearnrw.prototype.application.UserLogin Class Reference	91
7.39.1	Detailed Description	92
7.39.2	Constructor & Destructor Documentation	92
7.39.2.1	UserLogin	92
7.39.3	Member Function Documentation	92
7.39.3.1	onSelected	93
7.40	ilearnrw.prototype.application.UserProblems Class Reference	93
7.40.1	Constructor & Destructor Documentation	95
7.40.1.1	UserProblems	95
7.40.2	Member Function Documentation	95
7.40.2.1	addProblemDefinitionByCategory	95
7.40.2.2	addProblemDefinitionByLanguage	95
7.40.2.3	onSelected	95
7.40.3	Member Data Documentation	95
7.40.3.1	mUser	95
7.41	ilearnrw.user.GameAction Enum Reference	95
7.41.1	Member Data Documentation	95
7.41.1.1	DIFFICULT_WORD	95
7.42	ilearnrw.user.IlearnException Class Reference	96
7.42.1	Constructor & Destructor Documentation	97
7.42.1.1	IlearnException	97
7.42.1.2	IlearnException	97
7.42.2	Member Function Documentation	97
7.42.2.1	getMessage	97
7.43	ilearnrw.user.LanguageCode Enum Reference	97
7.43.1	Member Function Documentation	97
7.43.1.1	getEnglishCode	97
7.43.1.2	getGreekCode	97
7.43.2	Member Data Documentation	97
7.43.2.1	EN	97
7.44	ilearnrw.user.mockup.ProblemDefinitionIndexMockup Class Reference	98
7.44.1	Constructor & Destructor Documentation	99
7.44.1.1	ProblemDefinitionIndexMockup	99
7.44.2	Member Function Documentation	99
7.44.2.1	getProblemsByCategory	99
7.44.2.2	getProblemsByLanguage	99

7.44.3	Member Data Documentation	99
7.44.3.1	index	99
7.45	ilearnrw.user.problems.Category Class Reference	100
7.45.1	Constructor & Destructor Documentation	101
7.45.1.1	Category	101
7.45.2	Member Function Documentation	101
7.45.2.1	equals	101
7.45.2.2	getUrl	102
7.45.2.3	isSubCategory	102
7.45.2.4	setUrl	102
7.45.2.5	toString	102
7.45.3	Member Data Documentation	102
7.45.3.1	serialVersionUID	102
7.45.3.2	url	102
7.46	ilearnrw.user.problems.ProblemDefinition Class Reference	102
7.46.1	Constructor & Destructor Documentation	104
7.46.1.1	ProblemDefinition	104
7.46.2	Member Function Documentation	104
7.46.2.1	addProblemNode	104
7.46.2.2	getAvailableLanguages	104
7.46.2.3	getProblemNode	104
7.46.2.4	getProblemNodes	104
7.46.2.5	getScoreUpperBound	104
7.46.2.6	getType	104
7.46.2.7	getURI	104
7.46.2.8	setAvailableLanguages	104
7.46.2.9	setProblemNodes	104
7.46.2.10	setScoreUpperBound	104
7.46.2.11	setType	104
7.46.2.12	setURI	104
7.46.2.13	toString	104
7.46.3	Member Data Documentation	104
7.46.3.1	availableLanguages	104
7.46.3.2	problemNodes	104
7.46.3.3	scoreUpperBound	104
7.46.3.4	serialVersionUID	104
7.46.3.5	type	105
7.46.3.6	URI	105
7.47	ilearnrw.user.problems.ProblemDefinitionIndex Class Reference	105
7.47.1	Constructor & Destructor Documentation	106

7.47.1.1	ProblemDefinitionIndex	106
7.47.2	Member Function Documentation	106
7.47.2.1	addProblemDefinition	106
7.47.2.2	getAllProblems	106
7.47.2.3	getProblemDefinition	106
7.47.2.4	getProblemsByCategory	107
7.47.2.5	getProblemsByLanguage	107
7.47.3	Member Data Documentation	107
7.47.3.1	index	107
7.48	ilearnrw.user.problems.ProblemDefinitionIndexApi Interface Reference	107
7.48.1	Member Function Documentation	108
7.48.1.1	getProblemsByCategory	108
7.48.1.2	getProblemsByLanguage	108
7.49	ilearnrw.user.problems.ProblemNode Class Reference	109
7.49.1	Constructor & Destructor Documentation	110
7.49.1.1	ProblemNode	110
7.49.2	Member Function Documentation	111
7.49.2.1	getChildren	111
7.49.2.2	getNodeTitle	111
7.49.2.3	setChildren	111
7.49.2.4	setNodeTitle	111
7.49.3	Member Data Documentation	111
7.49.3.1	children	111
7.49.3.2	nodeTitle	111
7.49.3.3	serialVersionUID	111
7.50	ilearnrw.user.test.MockupTests Class Reference	112
7.50.1	Member Function Documentation	112
7.50.1.1	testIndex	112
7.51	ilearnrw.user.test.UserTests Class Reference	112
7.51.1	Member Function Documentation	112
7.51.1.1	createProblems	112
7.51.1.2	createUserProfile	113
7.52	ilearnrw.user.Tester Class Reference	113
7.52.1	Member Function Documentation	113
7.52.1.1	main	113
7.53	ilearnrw.user.User Class Reference	114
7.53.1	Constructor & Destructor Documentation	116
7.53.1.1	User	116
7.53.1.2	User	116
7.53.1.3	User	116

7.53.2	Member Function Documentation	116
7.53.2.1	getDetails	116
7.53.2.2	getOldSessions	116
7.53.2.3	getProfile	116
7.53.2.4	getSession	116
7.53.2.5	getUserId	116
7.53.2.6	setDetails	116
7.53.2.7	setOldSessions	116
7.53.2.8	setProfile	116
7.53.2.9	setSession	116
7.53.3	Member Data Documentation	116
7.53.3.1	details	116
7.53.3.2	oldSessions	116
7.53.3.3	profile	116
7.53.3.4	serialVersionUID	116
7.53.3.5	session	116
7.53.3.6	userId	116
7.54	ilearnrw.user.UserAction Class Reference	117
7.54.1	Detailed Description	119
7.54.2	Constructor & Destructor Documentation	119
7.54.2.1	UserAction	119
7.54.3	Member Function Documentation	119
7.54.3.1	createRandomDate	119
7.54.3.2	getApplicationId	119
7.54.3.3	getTag	119
7.54.3.4	getText	119
7.54.3.5	getTimeStamp	119
7.54.3.6	getUserId	120
7.54.3.7	randBetween	120
7.54.3.8	setApplicationId	120
7.54.3.9	setUserId	120
7.54.4	Member Data Documentation	120
7.54.4.1	mApplicationId	120
7.54.4.2	mTag	120
7.54.4.3	mTimeStamp	120
7.54.4.4	mUserId	120
7.54.4.5	mValue	120
7.54.4.6	serialVersionUID	120
7.55	ilearnrw.user.UserDetails Class Reference	121
7.55.1	Constructor & Destructor Documentation	123

7.55.1.1	UserDetails	123
7.55.1.2	UserDetails	123
7.55.1.3	UserDetails	123
7.55.2	Member Function Documentation	123
7.55.2.1	checkPassword	123
7.55.2.2	getLanguage	123
7.55.2.3	getUsername	123
7.55.2.4	hash	123
7.55.2.5	hasPassword	123
7.55.2.6	passwordIsSet	123
7.55.2.7	setLanguage	123
7.55.2.8	setLanguage	123
7.55.2.9	setPassword	124
7.55.2.10	setPasswordHash	124
7.55.2.11	setUsername	124
7.55.3	Member Data Documentation	124
7.55.3.1	desiredKeyLen	124
7.55.3.2	iterations	124
7.55.3.3	language	124
7.55.3.4	mPasswordHash	124
7.55.3.5	mPasswordSalt	124
7.55.3.6	saltLen	124
7.55.3.7	serialVersionUID	124
7.55.3.8	username	124
7.56	ilearnrw.user.UserPreferences Class Reference	125
7.56.1	Constructor & Destructor Documentation	126
7.56.1.1	UserPreferences	126
7.56.1.2	UserPreferences	126
7.56.2	Member Function Documentation	126
7.56.2.1	getFontSize	126
7.56.2.2	setFontSize	126
7.56.3	Member Data Documentation	126
7.56.3.1	fontSize	126
7.56.3.2	serialVersionUID	126
7.57	ilearnrw.user.UserProblem Class Reference	127
7.57.1	Constructor & Destructor Documentation	129
7.57.1.1	UserProblem	129
7.57.1.2	UserProblem	129
7.57.2	Member Function Documentation	129
7.57.2.1	getProblem	129

7.57.2.2	getProblemNode	129
7.57.2.3	setProblem	129
7.57.2.4	toString	129
7.57.3	Member Data Documentation	129
7.57.3.1	index	129
7.57.3.2	problem	129
7.57.3.3	serialVersionUID	129
7.58	ilearnrw.user.UserProblemsList Class Reference	129
7.58.1	Constructor & Destructor Documentation	130
7.58.1.1	UserProblemsList	130
7.58.2	Member Function Documentation	130
7.58.2.1	getList	130
7.58.2.2	loadTestProblems	130
7.58.2.3	setList	130
7.58.3	Member Data Documentation	131
7.58.3.1	list	131
7.58.3.2	serialVersionUID	131
7.59	ilearnrw.user.UserProfile Class Reference	131
7.59.1	Constructor & Destructor Documentation	133
7.59.1.1	UserProfile	133
7.59.1.2	UserProfile	133
7.59.1.3	UserProfile	133
7.59.1.4	UserProfile	133
7.59.2	Member Function Documentation	133
7.59.2.1	getLanguage	133
7.59.2.2	getPreferences	133
7.59.2.3	getProblemsList	133
7.59.2.4	setLanguage	133
7.59.2.5	setPreferences	133
7.59.2.6	setProblemsList	133
7.59.3	Member Data Documentation	133
7.59.3.1	language	133
7.59.3.2	preferences	133
7.59.3.3	problems	133
7.59.3.4	serialVersionUID	133
7.60	ilearnrw.user.UserSession Class Reference	134
7.60.1	Detailed Description	135
7.60.2	Member Function Documentation	135
7.60.2.1	insertAction	135
7.60.2.2	startSession	135

7.60.2.3	stopSession	135
7.60.3	Member Data Documentation	135
7.60.3.1	end	135
7.60.3.2	isActive	135
7.60.3.3	serialVersionUID	135
7.60.3.4	start	135
 Index		 136

Chapter 1

Main Page

Source Control

The projects source is managed in a private repository at github.

Project url: <https://github.com/mpakarlsson/ilearnrw>

Chapter 2

Doxygen

A Doxyfile for generating documentation can be found at `docs/Doxygen`.

Installation

Linux

todo: Provide installation documentation.

Windows

To build the documentation a couple of dependencies are required:

Name	Version	Url	Exe	Note
Doxygen	1.8.4	http://doxygen.org	doxygen.exe	-
Miktex	2.9.3759	http://miktex.org/download	tex.exe	Should work with other tex packages as well.
ghostscript	9.0.7	http://sourceforge.net/projects/ghostscript/	?	-
Graphviz (dot)	2.8	http://www.graphviz.org/	dot.exe	-

The exe files listed must be available in the windows `PATH%` variable.

Build documentation

Windows

To build the documentation run the `make_doc.bat` file located at `docs/make_doc.bat`.

The batchfile will run doxygen and then tex. The output `bin/doxygen/latex/refman.pdf` will be copied to `docs/doxygen.pdf`.

Chapter 3

DataLogger Log format

Following is a quick description about how data is, or will be, stored.

Users

Users are stored in the [UserStore](#). Initially the [UserStore](#) just uses Java serialization to write the User objects to disk. This requires that any objects that has a relationship (via class members) to the User object has to implement the `java.io.Serializable` interface.

This could be changed to a SQL backend in the future.

User Actions

The User actions are envisioned to be stored in a SQL database with a similar setup as described below.

Table UserActions:

Type	Name
PK	actionId
FK	userId
FK	marker
FK	applicationId
TimeStamp	timestamp
VARCHAR()	tag
VARCHAR()	value

This would result in SQL output similar to:

Id	Profile	Marker	Application	TimeStamp	Tag	Value
1	child1	Teach1	some_id	2013-06-14	ADD	cat
2	child1	child1	some_id	2013-06-14	REMOVE	cat

Using the `getUserActions()` function of the [IDataLogger](#) interface the database can be queried. To filter the result a [UserActionFilter](#) object is provided.

Chapter 4

Todo List

Group **DolphinTodo_20130619**

Implement IProfileAccessUpdater

DONE: Move interfaces to DataLogger

Implement all usecases provided by IProfileAccessUpdater into console application.

DONE: cleanup the code

DONE: Merge with the main branch.

Class **ilearnrw.datalogger.UserActionFilter**

Add getters/setters, appropriate functions etc.

Class **ilearnrw.datalogger.UserStore**

Replace storage backed with SQL database.

Enforce unique usernames.

Member **ilearnrw.datalogger.UserStore.update (User user)**

Don't allow multiple users with the same username.

Member **ilearnrw.games.NumbersToText.onSelected (ConsoleMenu menu)**

: it's the responsibility of the datalogger to save the actions

Class **ilearnrw.prototype.application.UserActions**

getActions(filter) should not return null? It's easier to return just an empty list

Class **ilearnrw.user.UserAction**

Add some documentation about this class.

Add get/setters

Member **ilearnrw.user.UserAction.createRandomDate (int startYear, int endYear)**

: remove these after testing UserActions with UserActionFilter enough

Member **ilearnrw.user.UserDetails.username**

Something to think about is whether this should be case-sensitive and if unicode should be supported. Unicode can create some headaches, there was a blog post about this recently (of which i only read the headline) see:

<http://labs.spotify.com/2013/06/18/creative-usernames/>

Group **OnlineResourceBankTodo**

: Could a resource consist of multiple items of the same type?

Chapter 5

Module Documentation

5.1 DolphinTodo_20130619

Todo Implement [IProfileAccessUpdater](#)

DONE: Move interfaces to [DataLogger](#)

Implement all usecases provided by [IProfileAccessUpdater](#) into console application.

DONE: cleanup the code

DONE: Merge with the main branch.

5.2 OnlineResourceBankTodo

Todo : Could a resource consist of multiple items of the same type?

Chapter 6

Namespace Documentation

6.1 Package ilearnrw

Packages

- package [application](#)
- package [datalogger](#)
[DataLogger](#) package.
- package [games](#)
- package [online](#)
- package [prototype](#)
- package [user](#)

6.2 Package ilearnrw.application

Classes

- class [ApplicationId](#)
[ApplicationId](#) identifies an application in different parts of the system.

6.3 Package ilearnrw.datalogger

[DataLogger](#) package.

Classes

- class [DataLogger](#)
- interface [IDataLogger](#)
Provides an interface for applications to log UserActions.
- interface [ILoginProvider](#)
- interface [IProfileAccessUpdater](#)
Interface to update profiles in the iLearnRW system.
- interface [IUserAdministration](#)
Provides User administration tasks.
- class [UserActionFilter](#)
- class [UserStore](#)
The [UserStore](#) is used to create and store user objects persistantly.

6.3.1 Detailed Description

[DataLogger](#) package. Four main interfaces has been defined at the moment.

1. [IDataLogger](#)
2. [ILoginProvider](#)
3. [IProfileAccessUpdater](#)
4. [IUserAdministration](#)

The [DataLogger](#) object implements [IDataLogger](#), [ILoginProvider](#) and [IProfileAccessUpdater](#).

See their respective documentation for details.

6.4 Package ilearnrw.games

Classes

- class [GameAddTwoNumbers](#)
- class [NumberProblems](#)
- class [NumbersToText](#)
- class [NumbersToTextHelper](#)
- class [UsersProvider](#)

6.5 Package ilearnrw.online

Packages

- package [resourcebank](#)

6.6 Package ilearnrw.online.resourcebank

Classes

- interface [IResource](#)
- interface [IResourceBank](#)
- interface [IResourceCollection](#)
- interface [ITeachingStrategy](#)

6.6.1 Detailed Description

Purpose

Excerpt from DOW ILearnRW (318803) 2012-06-22.pdf Page 49, point #7

On-line resource bank. A learning tool is useless when no appropriate learning material is available for it. The learning material should be readily available and accessible from everywhere, i.e., it should be on-line. The material should contain not only "isolated" titles of text. It should contain coherent collections of data which support specific teaching strategies and constitute well structured learning/intervention programs, forming a resource bank valuable to learners, educators and dyslexia professionals. The resource bank should be also to offer its customized views of its material to each

learner, based on the learner's profile, i.e., the resource bank should be profile-sensitive. That means, that the user will be able to browse the resource bank and view its content sorted according to the suitability/ appropriateness to her profile. In addition, the user should be able to issue search requests with her profile as one search argument, making in this way the reading activities more enjoyable.

Requirements

Needed resources

- A server.
- A database.

Required libraries

Functionality

The Online Resource Bank contains a number of Resources and a number of Teaching strategies. The Teaching strategies maps UserProblems to Resources.

A Resource combines multiple DataResources such as text, images and audio into a single resource.

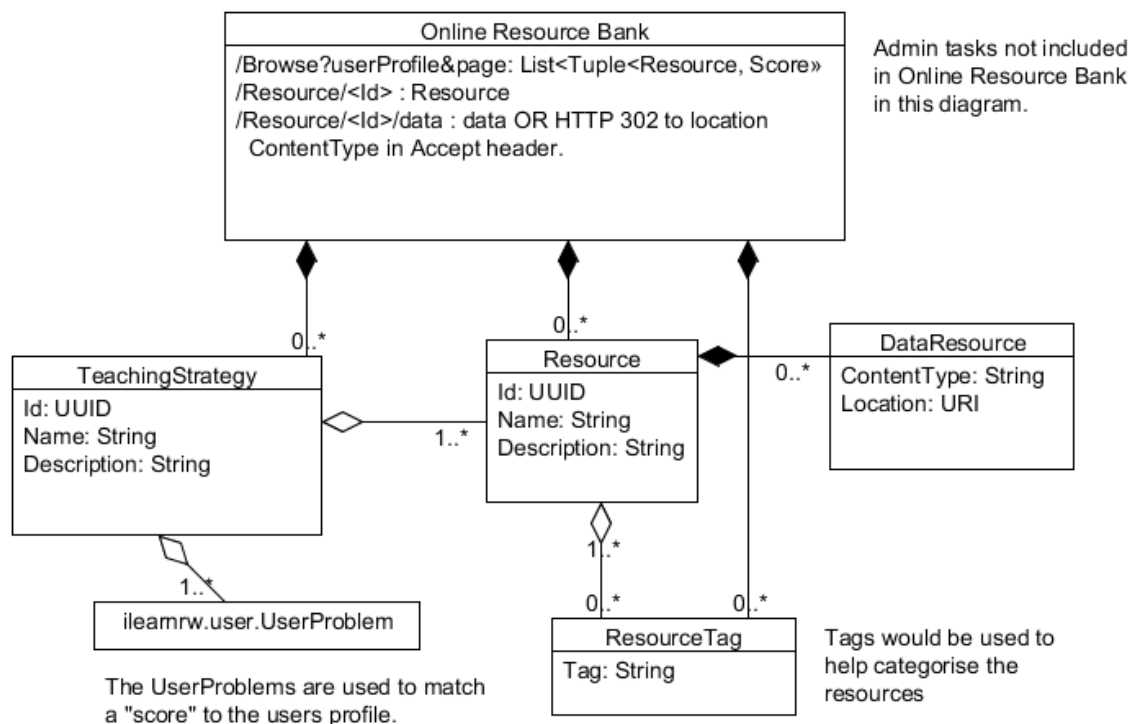


Figure 6.1: Online Resource Bank

RESTful API

The Online Resource Bank is designed to be a RESTful service. This means that no state is ever stored at the server and standard HTTP headers are utilized.

The clients connect to the service using standard HTTP. Since some information might be sensitive all requests sent to the server should always be encrypted.

Authorization

A cookie named `Token` would have to be set to authenticate the current user.

```
Cookie: Token=<token>
```

Token

The token would be one of 2 things

1. An id in a lookup table on the Authorization server
2. Signed user information (with potential extra information)

Option 1 would require the Online Resource Bank to contact the authorization server to verify the token. In option 2 as long as the signature can be verified the token can be trusted to be valid.

The token could be stored in a cookie on the client and sent to the server in each request.

Tokens are excluded from the rest of the documentation but must be available in the implementation.

API

Note that how to administer the system has been ignored at the moment but i would have the service also expose html pages which can be used to update the system.

Browse

```
GET /browse?userProfile=<userProfile>&page=<pageNumber>&SupportedContentTypes=plain%2Ftext%2C%20audio%2Fmp
```

Returns a list of resources suited for the particular user. Note that we need to know what content types the client can handle. Accepted format is `application/json`

GET of browse:

```
GET /browse?userProfile=<userProfile>&page=1&SupportedContentTypes=plain%2Ftext%2C%20audio%2Fmpeg
```

Response:

```
{
  "page": 1,
  "result": [
    {
      "name": "Resource 1",
      "description": "Resource 1 description",
      "score": 100
      "data": [
        ["plain/text; charset=\\"utf8\\" , 1233],
        ["audio/mpeg", 19393949],
      ]
    },
    ... (more resource objects)
  ]
}
```

Resources

```
GET /resource/<resource-id>
GET /resource/<resource-id>/data
```

Identifies the resource itself. This could be a piece of text, an audio clip etc. GET of resource:

```
GET /resource/<resource-id>
Accept: application/json
```


Response:

```

    ContentType: application/json

    {
      "name": "Resource 1",
      "description": "Resource 1 description",
      "data": [
        ["plain/text; charset=\"utf8\"", 1233],
        ["audio/mpeg", 19393949],
        ["image/png", 29393]
      ]
    }

```

To fetch the data of the resource in audio format:

```

GET /Resource/<resource-id>/data
Accept: audio/mpeg

```

Response:

```

HTTP/1.1 200 OK
Content-Type: audio/mpeg
Content-Length: xxx

```

Alternatively:

```

HTTP/1.1 302
Location: http://someServer/audio/someclip.mp3

```

Implementations for CREATE, PUT, POST and DELETE would allow administrative tasks to be done via an application or a web interface. Although the CREATE would just use `/Resource` as URI.

6.7 Package ilearnrw.prototype

Packages

- package [application](#)

6.8 Package ilearnrw.prototype.application

Classes

- class [ConsoleMenu](#)
- class [ConsoleMenuAction](#)
- class [DatabaseManager](#)
- class [Program](#)
- class [UserActions](#)
- class [UserActionsLog](#)
- class [UserGames](#)
- class [UserLogin](#)

Main menu of a logged in user.

- class [UserProblems](#)

6.9 Package ilearnrw.user

Packages

- package [mockup](#)
- package [problems](#)
- package [test](#)

Classes

- enum [GameAction](#)
- class [IlearnException](#)
- enum [LanguageCode](#)
- class [Tester](#)
- class [User](#)
- class [UserAction](#)
- class [UserDetails](#)
- class [UserPreferences](#)
- class [UserProblem](#)
- class [UserProblemsList](#)
- class [UserProfile](#)
- class [UserSession](#)

6.10 Package ilearnrw.user.mockup

Classes

- class [ProblemDefinitionIndexMockup](#)

6.11 Package ilearnrw.user.problems

Classes

- class [Category](#)
- class [ProblemDefinition](#)
- class [ProblemDefinitionIndex](#)
- interface [ProblemDefinitionIndexApi](#)
- class [ProblemNode](#)

6.12 Package ilearnrw.user.test

Classes

- class [MockupTests](#)
- class [UserTests](#)

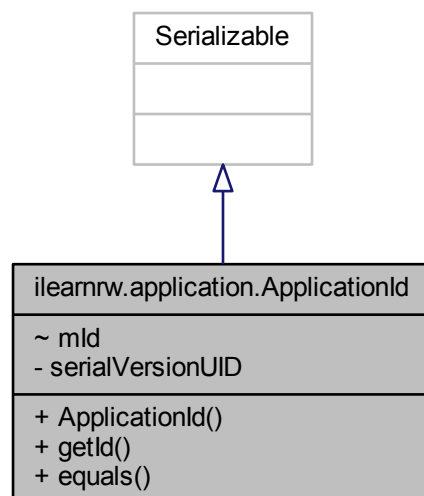
Chapter 7

Class Documentation

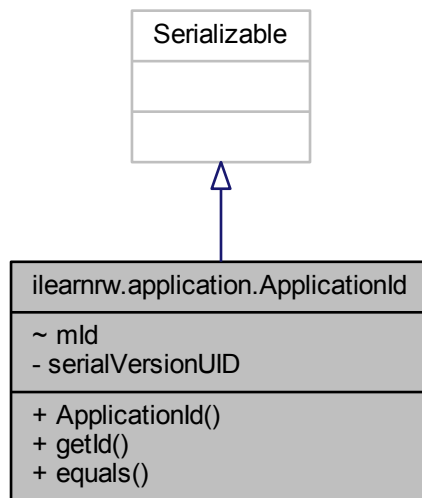
7.1 ilearnrw.application.ApplicationId Class Reference

[ApplicationId](#) identifies an application in different parts of the system.

Inheritance diagram for ilearnrw.application.ApplicationId:



Collaboration diagram for ilearnrw.application.ApplicationId:



Public Member Functions

- [ApplicationId](#) (String id)
- String [getId](#) ()
- boolean [equals](#) ([ApplicationId](#) id)

Package Attributes

- String [mld](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.1.1 Detailed Description

[ApplicationId](#) identifies an application in different parts of the system.

It should be a unique identifier

7.1.2 Constructor & Destructor Documentation

7.1.2.1 ilearnrw.application.ApplicationId.ApplicationId (String id)

7.1.3 Member Function Documentation

7.1.3.1 boolean ilearnrw.application.ApplicationId.equals (ApplicationId id)

7.1.3.2 String ilearnrw.application.ApplicationId.getId ()

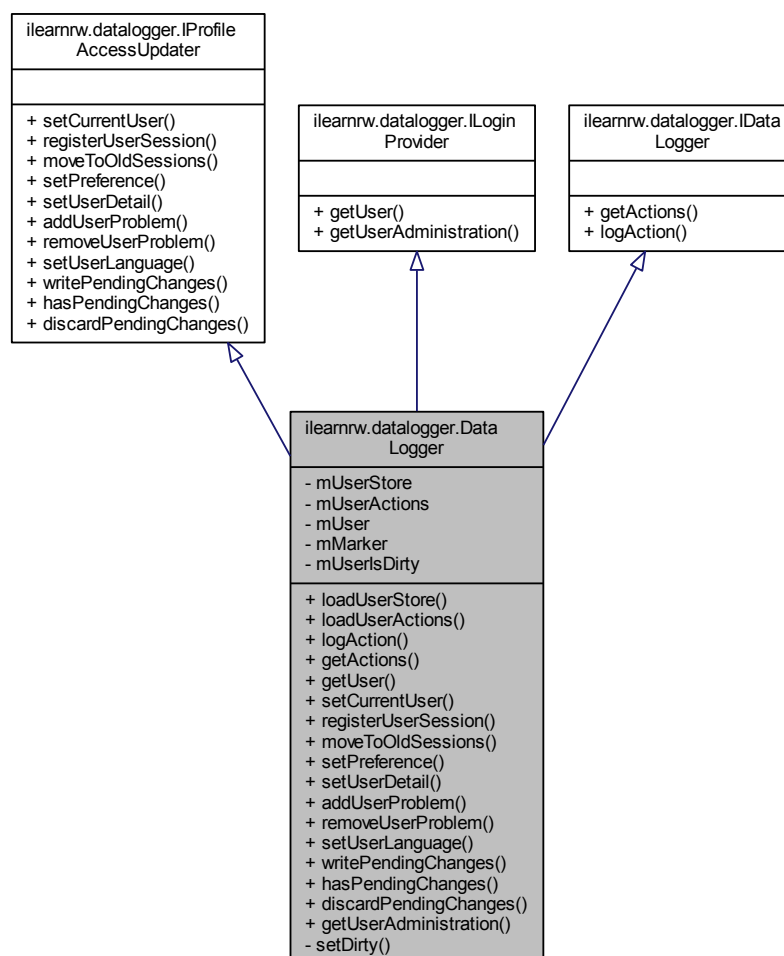
7.1.4 Member Data Documentation

7.1.4.1 String ilearnrw.application.ApplicationId.mld [package]

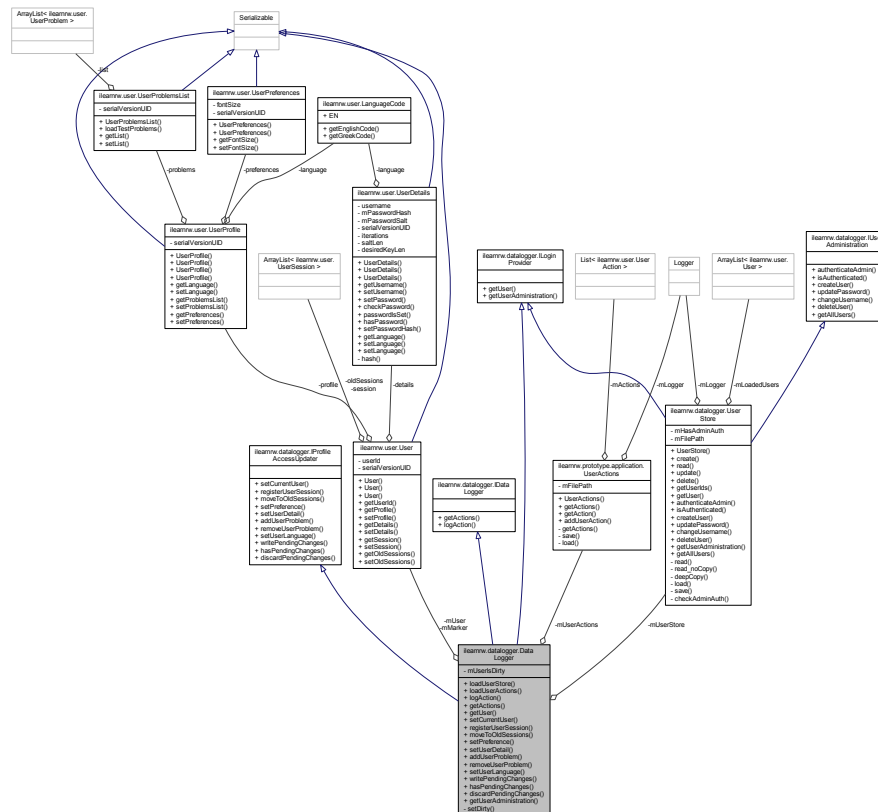
7.1.4.2 final long ilearnrw.application.ApplicationId.serialVersionUID = 1L [static],[private]

7.2 ilearnrw.datalogger.DataLogger Class Reference

Inheritance diagram for ilearnrw.datalogger.DataLogger:



Collaboration diagram for `ilearnrw.datalogger.DataLogger`:



Public Member Functions

- boolean [loadUserStore](#) (String filePath)
- boolean [loadUserActions](#) (String filePath)
- void [logAction](#) ([UserAction](#) action)

The Timestamp of the UserAction should be updated before the actions is stored. The timestamp should be local time of the server when this function is called. (Added some temporary code that generates random years within an interval since I needed to test the [UserActionFilter](#) date handling, remove it when done)

- List< [UserAction](#) > [getActions](#) ([UserActionFilter](#) filter)
- [User](#) [getUser](#) (String username, String password)
- void [setCurrentUser](#) ([User](#) user, [User](#) marker) throws [PermissionException](#), [PendingChangesAvailable](#)

Sets the userprofile to work with and who will be the `marker` for the changes.

- boolean [registerUserSession](#) ([UserSession](#) session)

Registers a new User session in the log.

- boolean [moveToOldSessions](#) ()

Adds current sessions to old sessions.

- boolean [setPreference](#) ([UserPreferences](#) preferences)

Registers a change to the users preferences.

- boolean [setUserDetail](#) ([UserDetails](#) userDetails)

Updates a UserDetail.

- boolean [addUserProblem](#) ([UserProblem](#) userProblem)

Adds a predefined ProblemDefinition to the user profile.

- boolean [removeUserProblem](#) ([UserProblem](#) userProblem)

Removes a ProblemDefinition.

- boolean [setUserLanguage](#) ([LanguageCode](#) language)
Sets the userlanguage.
- [User](#) [writePendingChanges](#) ()
Writes unwritten changes to the back end (persistent storage).
- boolean [hasPendingChanges](#) ()
Checks if there are any changes pending.
- void [discardPendingChanges](#) ()
- [IUserAdministration](#) [getUserAdministration](#) ()

Private Member Functions

- void [setDirty](#) ()

Private Attributes

- [UserStore](#) [mUserStore](#) = null
- [UserActions](#) [mUserActions](#) = null
- [User](#) [mUser](#) = null
- [User](#) [mMarker](#) = null
- boolean [mUserIsDirty](#) = false

7.2.1 Detailed Description

[DataLogger](#) class.

The Datalogger is the main object that Applications will use to query and update the user object and it's properties.

7.2.2 Member Function Documentation

7.2.2.1 boolean ilearnrw.datalogger.DataLogger.addUserProblem ([UserProblem](#) *userProblem*)

Adds a predefined ProblemDefinition to the user profile.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.2 void ilearnrw.datalogger.DataLogger.discardPendingChanges ()

Will discard the temporary user object.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.3 List<[UserAction](#)> ilearnrw.datalogger.DataLogger.getActions ([UserActionFilter](#) *filter*)

The list of user actions if you only are interested in the list of user actions

Implements [ilearnrw.datalogger.IDataLogger](#).

7.2.2.4 [User](#) ilearnrw.datalogger.DataLogger.getUser ([String](#) *username*, [String](#) *password*)

Gets a User object from a username and password.

Returns

A User object if successful. Otherwise null.

Implements [ilearnrw.datalogger.ILoginProvider](#).

7.2.2.5 IUserAdministration ilearnrw.datalogger.DataLogger.getUserAdministration ()

Implements [ilearnrw.datalogger.ILoginProvider](#).

7.2.2.6 boolean ilearnrw.datalogger.DataLogger.hasPendingChanges ()

Checks if there are any changes pending.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.7 boolean ilearnrw.datalogger.DataLogger.loadUserActions (String filePath)

Loads the UserActions from disk.

This has to be called during initialization.

Note

filePath does not have to exist. But the directory it is in has to.

Parameters

<i>filePath</i>	FilePath to a disk copy.
-----------------	--------------------------

Returns

true if successful. false if already loaded.

7.2.2.8 boolean ilearnrw.datalogger.DataLogger.loadUserStore (String filePath)

Loads the [UserStore](#) from disk.

This has to be called during initialization.

Note

filePath does not have to exist. But the directory it is in has to.

Parameters

<i>filePath</i>	FilePath to a disk copy.
-----------------	--------------------------

Returns

true if successful. false if already loaded.

7.2.2.9 void ilearnrw.datalogger.DataLogger.logAction (UserAction action)

The Timestamp of the UserAction should be updated before the actions is stored. The timestamp should be local time of the server when this function is called. (Added some temporary code that generates random years within an interval since I needed to test the [UserActionFilter](#) date handling, remove it when done)

Implements [ilearnrw.datalogger.IDataLogger](#).

7.2.2.10 boolean ilearnrw.datalogger.DataLogger.moveToOldSessions ()

Adds current sessions to old sessions.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.11 boolean ilearnrw.datalogger.DataLogger.registerUserSession (*UserSession session*)

Registers a new User session in the log.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.12 boolean ilearnrw.datalogger.DataLogger.removeUserProblem (*UserProblem userProblem*)

Removes a ProblemDefinition.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.13 void ilearnrw.datalogger.DataLogger.setCurrentUser (*User user*, *User marker*) throws **PermissionException**, **PendingChangesAvailable**

Sets the userprofile to work with and who will be the `marker` for the changes.

Doing this once saves the application from passing the userprofile to all functions.

The `marker` can be the same as the profile to be edited.

Exceptions

<i>PermissionException</i>	Thrown when the admin profile does not have permissions to update the profile.
<i>PendingChangesAvailable</i>	Thrown when there are pending changes are available.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.14 void ilearnrw.datalogger.DataLogger.setDirty () [private]**7.2.2.15** boolean ilearnrw.datalogger.DataLogger.setPreference (*UserPreferences preferences*)

Registers a change to the users preferences.

Should changes to user preferences be logged so that teachers/experts can be notified/see that a preference has been changed?

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.16 boolean ilearnrw.datalogger.DataLogger.setUserDetail (*UserDetails userDetails*)

Updates a UserDetail.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.17 boolean ilearnrw.datalogger.DataLogger.setUserLanguage (*LanguageCode Language*)

Sets the userlanguage.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.2.18 User ilearnrw.datalogger.DataLogger.writePendingChanges ()

Writes unwritten changes to the back end (persistent storage).

This will flush all pending operations in the same order as they are added. This is important if for example the same preference is set twice.

Returns

The new user after the changes has been applied.

Implements [ilearnrw.datalogger.IProfileAccessUpdater](#).

7.2.3 Member Data Documentation

7.2.3.1 User ilearnrw.datalogger.DataLogger.mMarker = null [private]

The `marker` to use when logging changes.

7.2.3.2 User ilearnrw.datalogger.DataLogger.mUser = null [private]

The current user to be updated.

7.2.3.3 UserActions ilearnrw.datalogger.DataLogger.mUserActions = null [private]

7.2.3.4 boolean ilearnrw.datalogger.DataLogger.mUserIsDirty = false [private]

Flag to keep track on whether `mUser` is dirty;

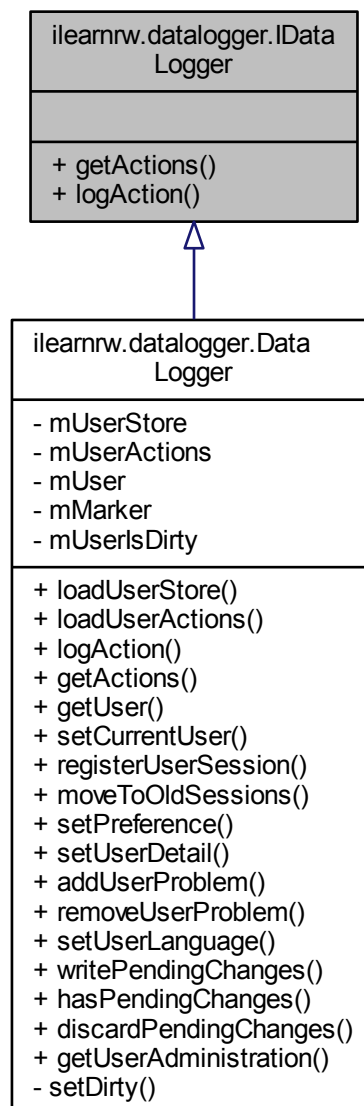
7.2.3.5 UserStore ilearnrw.datalogger.DataLogger.mUserStore = null [private]

The object that store users.

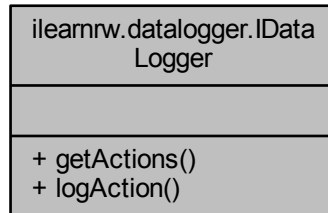
7.3 ilearnrw.datalogger.IDataLogger Interface Reference

Provides an interface for applications to log `UserActions`.

Inheritance diagram for ilearnrw.datalogger.IDataLogger:



Collaboration diagram for `ilearnrw.datalogger.IDataLogger`:



Public Member Functions

- List< [UserAction](#) > [getActions](#) ([UserActionFilter](#) filter)
- void [logAction](#) ([UserAction](#) action)

The Timestamp of the UserAction should be updated before the actions is stored. The timestamp should be local time of the server when this function is called. (Added some temporary code that generates random years within an interval since I needed to test the [UserActionFilter](#) date handling, remove it when done)

7.3.1 Detailed Description

Provides an interface for applications to log UserActions.

7.3.2 Member Function Documentation

7.3.2.1 List<UserAction> ilearnrw.datalogger.IDataLogger.getActions (UserActionFilter filter)

Given a [UserActionFilter](#) this function can return a list of UserActions.

Implemented in [ilearnrw.datalogger.DataLogger](#).

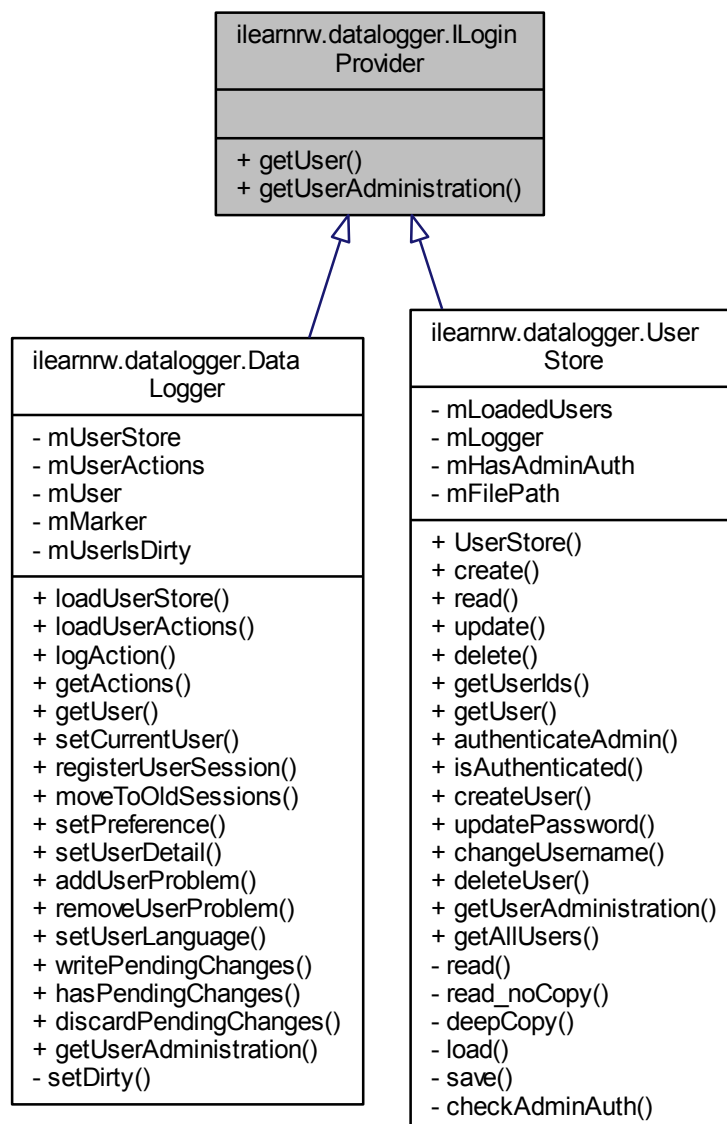
7.3.2.2 void ilearnrw.datalogger.IDataLogger.logAction (UserAction action)

The Timestamp of the UserAction should be updated before the actions is stored. The timestamp should be local time of the server when this function is called. (Added some temporary code that generates random years within an interval since I needed to test the [UserActionFilter](#) date handling, remove it when done)

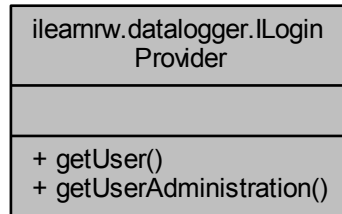
Implemented in [ilearnrw.datalogger.DataLogger](#).

7.4 ilearnrw.datalogger.ILoginProvider Interface Reference

Inheritance diagram for ilearnrw.datalogger.ILoginProvider:



Collaboration diagram for `ilearnrw.datalogger.ILoginProvider`:



Public Member Functions

- [User](#) `getUser` (String username, String password)

Authenticates a User.

- [IUserAdministration](#) `getUserAdministration` ()

7.4.1 Member Function Documentation

7.4.1.1 User `ilearnrw.datalogger.ILoginProvider.getUser` (String username, String password)

Authenticates a User.

Implemented in [ilearnrw.datalogger.UserStore](#), and [ilearnrw.datalogger.DataLogger](#).

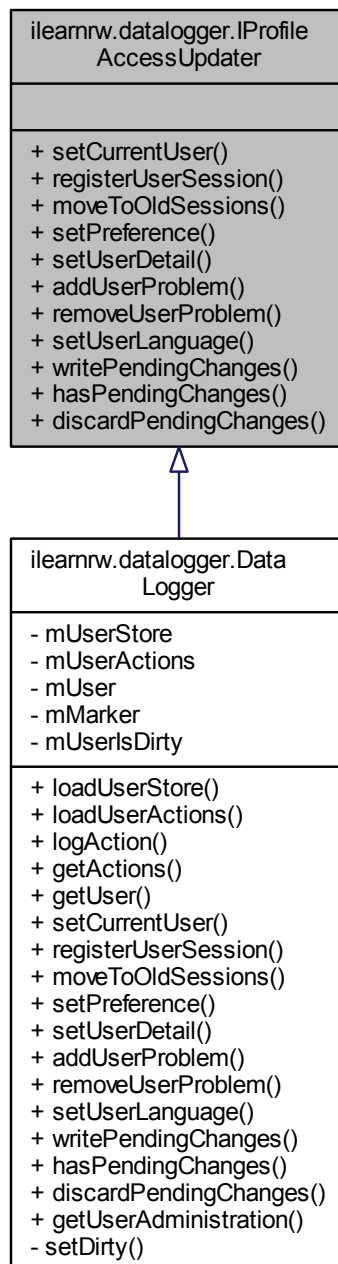
7.4.1.2 IUserAdministration `ilearnrw.datalogger.ILoginProvider.getUserAdministration` ()

Implemented in [ilearnrw.datalogger.UserStore](#), and [ilearnrw.datalogger.DataLogger](#).

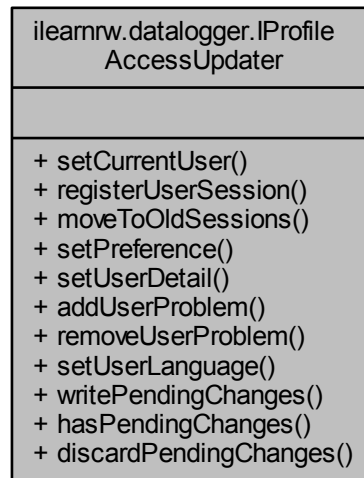
7.5 ilearnrw.datalogger.IProfileAccessUpdater Interface Reference

Interface to update profiles in the iLearnRW system.

Inheritance diagram for ilearnrw.datalogger.IProfileAccessUpdater:



Collaboration diagram for ilearnrw.datalogger.IProfileAccessUpdater:



Classes

- class [PendingChangesAvailable](#)
Indicates that an action cannot be performed due to pending changes.
- class [PermissionException](#)
The marker does not have permission to perform the requested change.

Public Member Functions

- void [setCurrentUser](#) ([User](#) user, [User](#) marker) throws [PermissionException](#), [PendingChangesAvailable](#)
*Sets the userprofile to work with and who will be the *marker* for the changes.*
- boolean [registerUserSession](#) ([UserSession](#) session)
Registers a new User session in the log.
- boolean [moveToOldSessions](#) ()
Adds current sessions to old sessions.
- boolean [setPreference](#) ([UserPreferences](#) preferences)
Registers a change to the users preferences.
- boolean [setUserDetail](#) ([UserDetails](#) userDetails)
Updates a UserDetail.
- boolean [addUserProblem](#) ([UserProblem](#) userProblem)
Adds a predefined ProblemDefinition to the user profile.
- boolean [removeUserProblem](#) ([UserProblem](#) userProblem)
Removes a ProblemDefinition.
- boolean [setUserLanguage](#) ([LanguageCode](#) Language)
Sets the userlanguage.
- [User](#) [writePendingChanges](#) ()
Writes unwritten changes to the back end (persistent storage).
- boolean [hasPendingChanges](#) ()

Checks if there are any changes pending.

- void [discardPendingChanges](#) ()

Discards all pending changes and resets the internal state.

7.5.1 Detailed Description

Interface to update profiles in the iLearnRW system.

This interface assumes that `Apps` would get a read-only copy of the users profile and would then have to use this interface to make changes to the profile. When the profile has been updated a new readonly copy would have to be fetched.

A nice side-effect is that we can make sure to log all changes to the profile.

Updater also manages

Registry - Handles/stores users/profiles.

Note

This is the interface that the application would use. Not an interface appropriate to a webservice. The implementer of this interface could still use a webservice to store the information in.

UserAdd/delete etc.. Get user (given userid, password)

7.5.2 Member Function Documentation

7.5.2.1 boolean ilearnrw.datalogger.IProfileAccessUpdater.addUserProblem (`UserProblem` *userProblem*)

Adds a predefined ProblemDefinition to the user profile.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.2 void ilearnrw.datalogger.IProfileAccessUpdater.discardPendingChanges ()

Discards all pending changes and resets the internal state.

In order to add new changes `setCurrentUser` has to be called again.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.3 boolean ilearnrw.datalogger.IProfileAccessUpdater.hasPendingChanges ()

Checks if there are any changes pending.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.4 boolean ilearnrw.datalogger.IProfileAccessUpdater.moveToOldSessions ()

Adds current sessions to old sessions.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.5 boolean ilearnrw.datalogger.IProfileAccessUpdater.registerUserSession (`UserSession` *session*)

Registers a new User session in the log.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.6 boolean ilearnrw.datalogger.IProfileAccessUpdater.removeUserProblem (*UserProblem userProblem*)

Removes a ProblemDefinition.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.7 void ilearnrw.datalogger.IProfileAccessUpdater.setCurrentUser (*User user*, *User marker*) throws *PermissionException*, *PendingChangesAvailable*

Sets the userprofile to work with and who will be the `marker` for the changes.

Doing this once saves the application from passing the userprofile to all functions.

The `marker` can be the same as the profile to be edited.

Exceptions

PermissionException	Thrown when the admin profile does not have permissions to update the profile.
PendingChangesAvailable	Thrown when there are pending changes are available.

Parameters

<i>user</i>	Profile of the user to update
<i>marker</i>	Profile of the user who is doing the update

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.8 boolean ilearnrw.datalogger.IProfileAccessUpdater.setPreference (*UserPreferences preferences*)

Registers a change to the users preferences.

Should changes to user preferences be logged so that teachers/experts can be notified/see that a preference has been changed?

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.9 boolean ilearnrw.datalogger.IProfileAccessUpdater.setUserDetail (*UserDetails userDetails*)

Updates a UserDetail.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.10 boolean ilearnrw.datalogger.IProfileAccessUpdater.setUserLanguage (*LanguageCode language*)

Sets the userlanguage.

Implemented in [ilearnrw.datalogger.DataLogger](#).

7.5.2.11 User ilearnrw.datalogger.IProfileAccessUpdater.writePendingChanges ()

Writes unwritten changes to the back end (persistent storage).

This will flush all pending operations in the same order as they are added. This is important if for example the same preference is set twice.

Returns

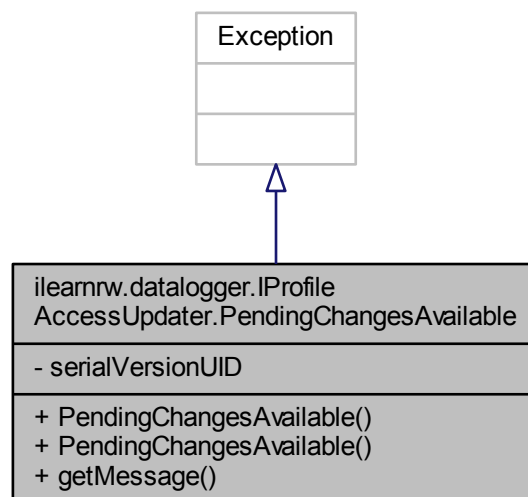
The new user after the changes has been applied.

Implemented in [ilearnrw.datalogger.DataLogger](#).

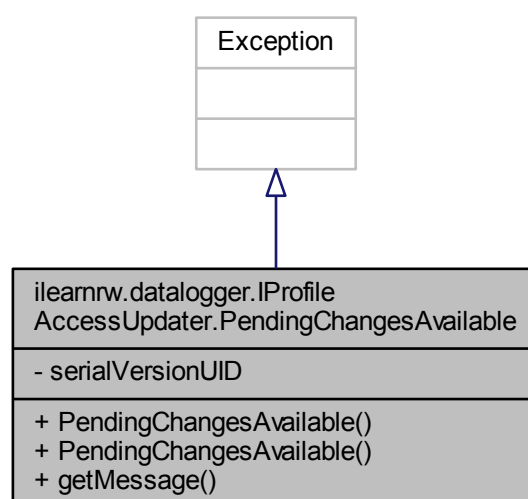
7.6 ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable Class Reference

Indicates that an action cannot be performed due to pending changes.

Inheritance diagram for ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable:



Collaboration diagram for ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable:



Public Member Functions

- [PendingChangesAvailable](#) (String message)
- [PendingChangesAvailable](#) (String message, Throwable throwable)
- String [getMessage](#) ()

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.6.1 Detailed Description

Indicates that an action cannot be performed due to pending changes.

If the action was to be performed the pending changes would be lost.

7.6.2 Constructor & Destructor Documentation

7.6.2.1 `ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable.PendingChangesAvailable (String message)`

7.6.2.2 `ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable.PendingChangesAvailable (String message, Throwable throwable)`

7.6.3 Member Function Documentation

7.6.3.1 `String ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable.getMessage ()`

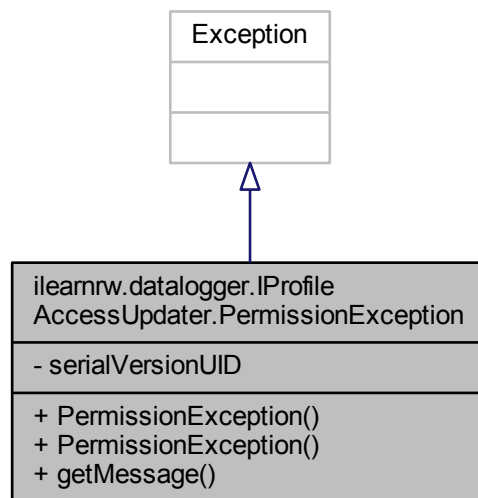
7.6.4 Member Data Documentation

7.6.4.1 `final long ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable.serialVersionUID = 1L` `[static], [private]`

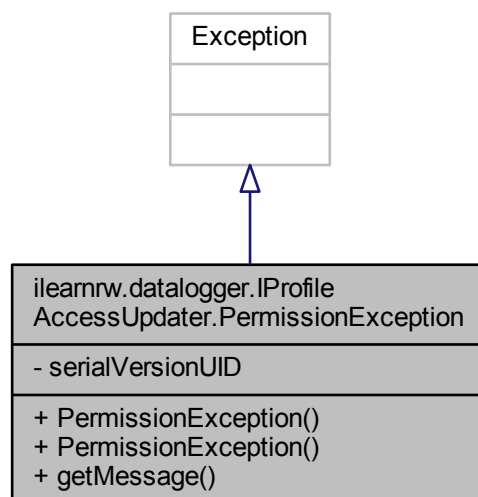
7.7 ilearnrw.datalogger.IProfileAccessUpdater.PermissionException Class Reference

The marker does not have permission to perform the requested change.

Inheritance diagram for ilearnrw.datalogger.IProfileAccessUpdater.PermissionException:



Collaboration diagram for ilearnrw.datalogger.IProfileAccessUpdater.PermissionException:



Public Member Functions

- [PermissionException](#) (String message)
- [PermissionException](#) (String message, Throwable throwable)
- String [getMessage](#) ()

Static Private Attributes

- static final long `serialVersionUID` = 1L

7.7.1 Detailed Description

The marker does not have permission to perform the requested change.

7.7.2 Constructor & Destructor Documentation

7.7.2.1 `ilearnrw.datalogger.IProfileAccessUpdater.PermissionException.PermissionException (String message)`

7.7.2.2 `ilearnrw.datalogger.IProfileAccessUpdater.PermissionException.PermissionException (String message, Throwable throwable)`

7.7.3 Member Function Documentation

7.7.3.1 `String ilearnrw.datalogger.IProfileAccessUpdater.PermissionException.getMessage ()`

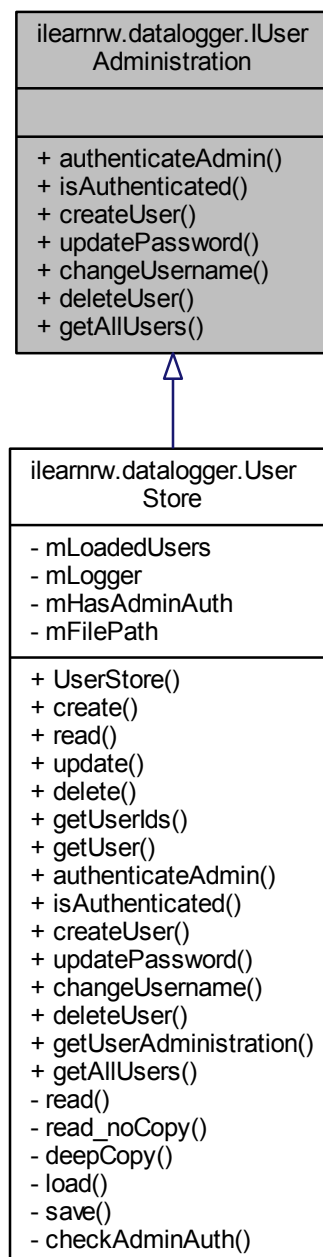
7.7.4 Member Data Documentation

7.7.4.1 `final long ilearnrw.datalogger.IProfileAccessUpdater.PermissionException.serialVersionUID = 1L` `[static],`
`[private]`

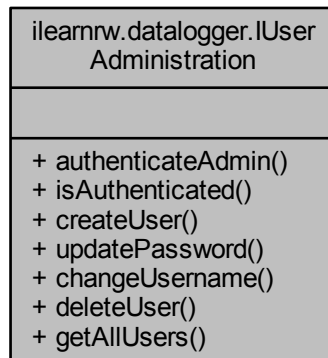
7.8 ilearnrw.datalogger.IUserAdministration Interface Reference

Provides User administration tasks.

Inheritance diagram for ilearnrw.datalogger.IUserAdministration:



Collaboration diagram for `ilearnrw.datalogger.IUserAdministration`:



Classes

- class [AuthenticationException](#)

The marker does not have permission to perform the requested change.

Public Member Functions

- void [authenticateAdmin](#) (String adminPassword) throws AuthenticationException
- boolean [isAuthenticated](#) ()
- [User](#) [createUser](#) (String username, String password) throws AuthenticationException
- [User](#) [updatePassword](#) (String username, String newPassword) throws AuthenticationException
- [User](#) [changeUsername](#) (String oldUsername, String newUsername) throws AuthenticationException
- boolean [deleteUser](#) (String username) throws AuthenticationException
- List< [User](#) > [getAllUsers](#) () throws AuthenticationException

7.8.1 Detailed Description

Provides User administration tasks.

7.8.2 Member Function Documentation

7.8.2.1 void `ilearnrw.datalogger.IUserAdministration.authenticateAdmin (String adminPassword)` throws `AuthenticationException`

Authenticates an Administrator by requiring a password.

Parameters

<i>adminPassword</i>	The administrator password. This could be set in a configuration file.
----------------------	--

Implemented in [ilearnrw.datalogger.UserStore](#).

7.8.2.2 User ilearnrw.datalogger.IUserAdministration.changeUsername (String *oldUsername*, String *newUsername*) throws AuthenticationException

Implemented in [ilearnrw.datalogger.UserStore](#).

7.8.2.3 User ilearnrw.datalogger.IUserAdministration.createUser (String *username*, String *password*) throws AuthenticationException

Creates a new User.

Parameters

<i>username</i>	
<i>password</i>	

Returns

The new User.

Implemented in [ilearnrw.datalogger.UserStore](#).

7.8.2.4 boolean ilearnrw.datalogger.IUserAdministration.deleteUser (String *username*) throws AuthenticationException

Implemented in [ilearnrw.datalogger.UserStore](#).

7.8.2.5 List<User> ilearnrw.datalogger.IUserAdministration.getAllUsers () throws AuthenticationException

Implemented in [ilearnrw.datalogger.UserStore](#).

7.8.2.6 boolean ilearnrw.datalogger.IUserAdministration.isAuthenticated ()

Implemented in [ilearnrw.datalogger.UserStore](#).

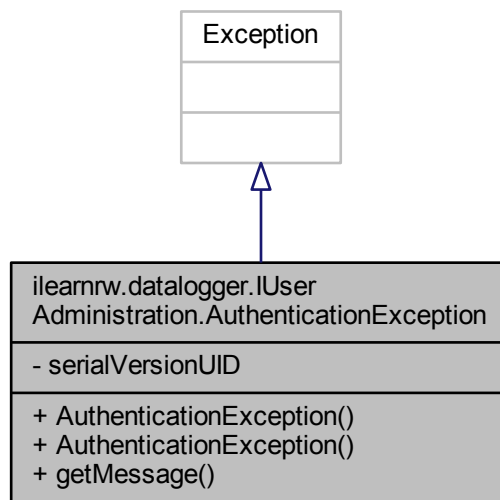
7.8.2.7 User ilearnrw.datalogger.IUserAdministration.updatePassword (String *username*, String *newPassword*) throws AuthenticationException

Implemented in [ilearnrw.datalogger.UserStore](#).

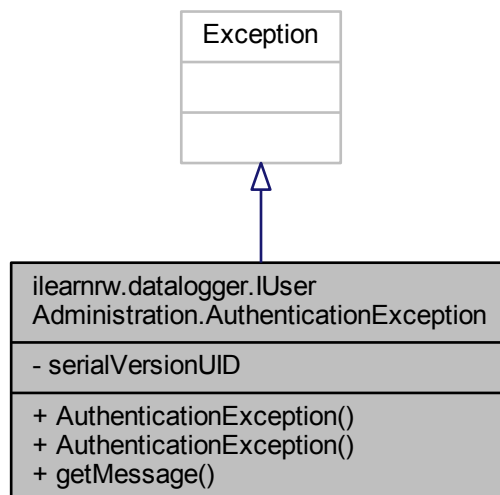
7.9 ilearnrw.datalogger.IUserAdministration.AuthenticationException Class Reference

The marker does not have permission to perform the requested change.

Inheritance diagram for `ilearnrw.datalogger.IUserAdministration.AuthenticationException`:



Collaboration diagram for `ilearnrw.datalogger.IUserAdministration.AuthenticationException`:



Public Member Functions

- [AuthenticationException](#) (String message)
- [AuthenticationException](#) (String message, Throwable throwable)
- String [getMessage](#) ()

Static Private Attributes

- static final long `serialVersionUID` = 1L

7.9.1 Detailed Description

The marker does not have permission to perform the requested change.

7.9.2 Constructor & Destructor Documentation

7.9.2.1 `ilearnrw.datalogger.IUserAdministration.AuthenticationException.AuthenticationException (String message)`

7.9.2.2 `ilearnrw.datalogger.IUserAdministration.AuthenticationException.AuthenticationException (String message, Throwable throwable)`

7.9.3 Member Function Documentation

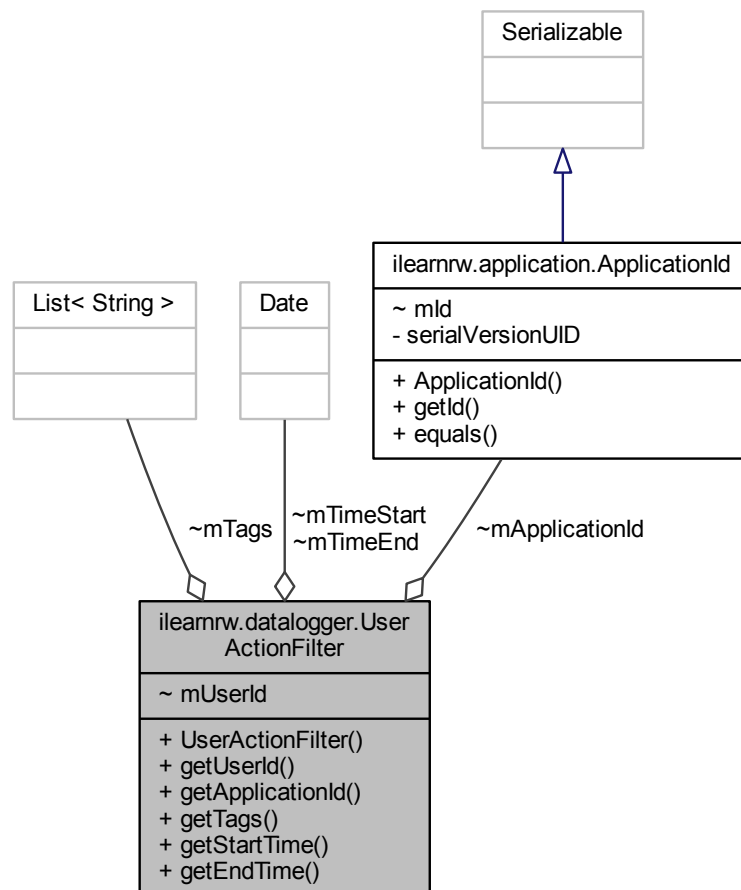
7.9.3.1 `String ilearnrw.datalogger.IUserAdministration.AuthenticationException.getMessage ()`

7.9.4 Member Data Documentation

7.9.4.1 `final long ilearnrw.datalogger.IUserAdministration.AuthenticationException.serialVersionUID = 1L` `[static],`
`[private]`

7.10 ilearnrw.datalogger.UserActionFilter Class Reference

Collaboration diagram for ilearnrw.datalogger.UserActionFilter:



Public Member Functions

- [UserActionFilter](#) (int userId, [ApplicationId](#) appld, Date start, Date end, List< String > tags)
- int [getUserId](#) ()
- [ApplicationId](#) [getApplicationId](#) ()
- List< String > [getTags](#) ()
- Date [getStartTime](#) ()
- Date [getEndTime](#) ()

Package Attributes

- int [mUserId](#)
- [ApplicationId](#) [mApplicationId](#)
- List< String > [mTags](#)
- Date [mTimeStart](#)
- Date [mTimeEnd](#)

7.10.1 Detailed Description

The [UserActionFilter](#) is used to query the datalogger for information

Todo Add getters/setters, appropriate functions etc.

7.10.2 Constructor & Destructor Documentation

7.10.2.1 `ilearnrw.datalogger.UserActionFilter.UserActionFilter (int userId, ApplicationId appld, Date start, Date end, List<String> tags)`

Creates a filter that is used on the UserAction list

Parameters

--	--

7.10.3 Member Function Documentation

7.10.3.1 `ApplicationId ilearnrw.datalogger.UserActionFilter.getApplicationId ()`

7.10.3.2 `Date ilearnrw.datalogger.UserActionFilter.getEndTime ()`

7.10.3.3 `Date ilearnrw.datalogger.UserActionFilter.getStartTime ()`

7.10.3.4 `List<String> ilearnrw.datalogger.UserActionFilter.getTags ()`

7.10.3.5 `int ilearnrw.datalogger.UserActionFilter.getUserId ()`

7.10.4 Member Data Documentation

7.10.4.1 `ApplicationId ilearnrw.datalogger.UserActionFilter.mApplicationId` [package]

7.10.4.2 `List<String> ilearnrw.datalogger.UserActionFilter.mTags` [package]

7.10.4.3 `Date ilearnrw.datalogger.UserActionFilter.mTimeEnd` [package]

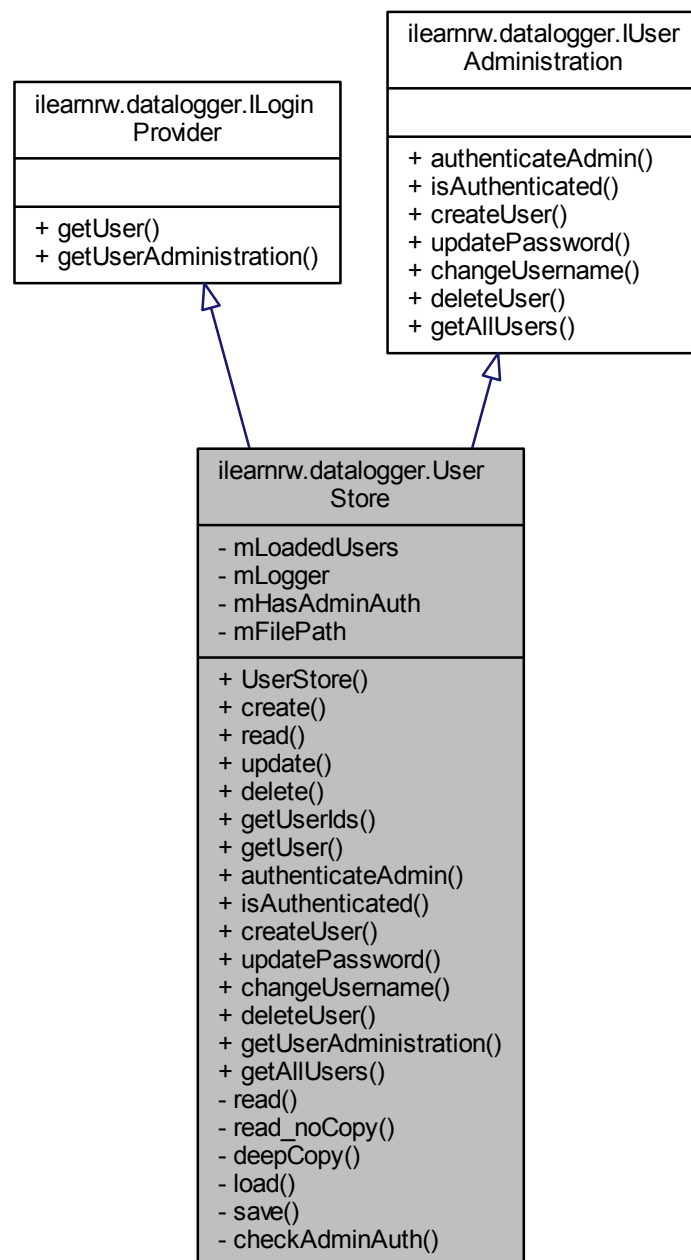
7.10.4.4 `Date ilearnrw.datalogger.UserActionFilter.mTimeStart` [package]

7.10.4.5 `int ilearnrw.datalogger.UserActionFilter.mUserId` [package]

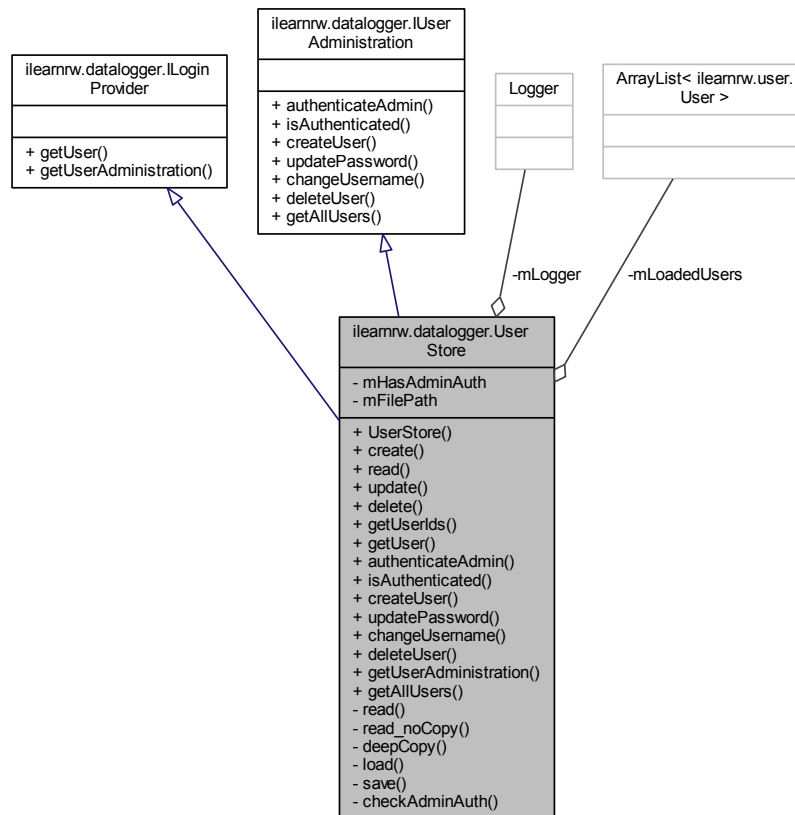
7.11 ilearnrw.datalogger.UserStore Class Reference

The [UserStore](#) is used to create and store user objects persistantly.

Inheritance diagram for ilearnrw.datalogger.UserStore:



Collaboration diagram for ilearnrw.datalogger.UserStore:



Public Member Functions

- [UserStore](#) (String filePath)
- [User create](#) ()
- [User read](#) (int userId)
- [User update](#) ([User](#) user)
- boolean [delete](#) (int userId)
- LinkedList< Integer > [getUserIds](#) ()
- [User getUser](#) (String username, String password)
Authenticates a User.
- void [authenticateAdmin](#) (String adminPassword) throws AuthenticationException
- boolean [isAuthenticated](#) ()
- [User createUser](#) (String username, String password) throws AuthenticationException
- [User updatePassword](#) (String username, String newPassword) throws AuthenticationException
- [User changeUsername](#) (String oldUsername, String newUsername) throws AuthenticationException
- boolean [deleteUser](#) (String username) throws AuthenticationException
- [IUserAdministration getUserAdministration](#) ()
- List< [User](#) > [getAllUsers](#) () throws AuthenticationException

Private Member Functions

- [User read](#) (String username)

- [User read_noCopy](#) (int userId)
- [User deepCopy](#) ([User](#) user)
- void [load](#) ()
Loads the serialized objects in `mFilePath` into `mLoadedUsers`.
- void [save](#) ()
Serializes `mLoadedUsers` into `mFilePath`
- void [checkAdminAuth](#) () throws `AuthenticationException`

Private Attributes

- `ArrayList< User > mLoadedUsers`
- Logger [mLogger](#)
- boolean [mHasAdminAuth](#) = false

Static Private Attributes

- static String [mFilePath](#)
The complete path to the file to read/write the user information from/to.

7.11.1 Detailed Description

The [UserStore](#) is used to create and store user objects persistantly.

Todo Replace storage backed with SQL database.
Enforce unique usernames.

Note

At the moment Java's built in serialization is used to store users to disk. This requires all objects to be stored to implement the `Serializable` interface.

Implements a CRUD interface for users and save/load functions to store the data persistently.

7.11.2 Constructor & Destructor Documentation

7.11.2.1 `ilearnrw.datalogger.UserStore.UserStore (String filePath)`

Parameters

<i>filePath</i>	The full path to the file to use to store the user profiles etc.
-----------------	--

7.11.3 Member Function Documentation

7.11.3.1 `void ilearnrw.datalogger.UserStore.authenticateAdmin (String adminPassword)` throws `AuthenticationException`

Authenticates an Administrator by requiring a password.

Parameters

<i>adminPassword</i>	The administrator password. This could be set in a configuration file.
----------------------	--

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.3.2 User ilearnrw.datalogger.UserStore.changeUsername (String *oldUsername*, String *newUsername*) throws AuthenticationException

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.3.3 void ilearnrw.datalogger.UserStore.checkAdminAuth () throws AuthenticationException [private]

7.11.3.4 User ilearnrw.datalogger.UserStore.create ()

Create a new user.

Will generate a new UserId for the user.

Returns

A new User object.

7.11.3.5 User ilearnrw.datalogger.UserStore.createUser (String *username*, String *password*) throws AuthenticationException

Creates a new User.

Parameters

<i>username</i>	
<i>password</i>	

Returns

The new User.

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.3.6 User ilearnrw.datalogger.UserStore.deepCopy (User *user*) [private]

Returns a copy of a User object

This is called from functions returning a user in this class since we do not want to give out the actual reference stored in the `_loadedUsers` list.

7.11.3.7 boolean ilearnrw.datalogger.UserStore.delete (int *userId*)

Deletes a user from the store.

Parameters

<i>userId</i>	userId of the user to delete
---------------	------------------------------

Returns

true if successful.

7.11.3.8 **boolean** `ilearnrw.datalogger.UserStore.deleteUser (String username)` throws **AuthenticationException**

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.3.9 **List<User>** `ilearnrw.datalogger.UserStore.getAllUsers ()` throws **AuthenticationException**

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.3.10 **User** `ilearnrw.datalogger.UserStore.getUser (String username, String password)`

Authenticates a User.

Implements [ilearnrw.datalogger.ILoginProvider](#).

7.11.3.11 **IUserAdministration** `ilearnrw.datalogger.UserStore.getUserAdministration ()`

Implements [ilearnrw.datalogger.ILoginProvider](#).

7.11.3.12 **LinkedList<Integer>** `ilearnrw.datalogger.UserStore.getUserIds ()`

Returns

A list of all stored userids.

7.11.3.13 **boolean** `ilearnrw.datalogger.UserStore.isAuthenticated ()`

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.3.14 **void** `ilearnrw.datalogger.UserStore.load ()` [private]

Loads the serialized objects in `mFilePath` into `mLoadedUsers`.

7.11.3.15 **User** `ilearnrw.datalogger.UserStore.read (int userId)`

Reads a User from the [UserStore](#)

Parameters

<i>userId</i>	The userId of the user to load
---------------	--------------------------------

Returns

The user with `userId == userId`

7.11.3.16 **User** `ilearnrw.datalogger.UserStore.read (String username)` [private]

7.11.3.17 **User** `ilearnrw.datalogger.UserStore.read_noCopy (int userId)` [private]

7.11.3.18 **void** `ilearnrw.datalogger.UserStore.save ()` [private]

Serializes `mLoadedUsers` into `mFilePath`

7.11.3.19 User ilearnrw.datalogger.UserStore.update (User user)

Updates the user currently stored with the one supplied.

Todo Don't allow multiple users with the same username.

Parameters

<i>user</i>	The new user copy to store.
-------------	-----------------------------

Returns

A new copy of the user object.

7.11.3.20 User ilearnrw.datalogger.UserStore.updatePassword (String username, String newPassword) throws AuthenticationException

Implements [ilearnrw.datalogger.IUserAdministration](#).

7.11.4 Member Data Documentation

7.11.4.1 String ilearnrw.datalogger.UserStore.mFilePath [static], [private]

The complete path to the file to read/write the user information from/to.

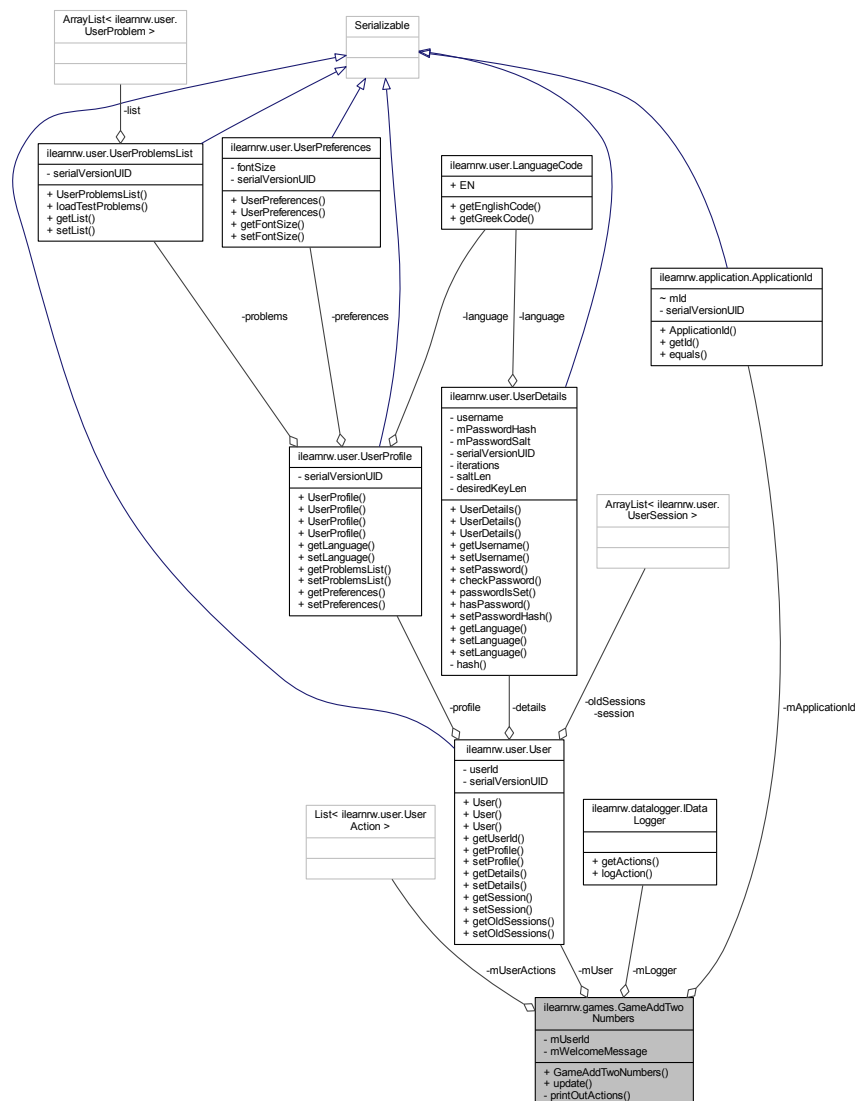
7.11.4.2 boolean ilearnrw.datalogger.UserStore.mHasAdminAuth = false [private]

7.11.4.3 ArrayList<User> ilearnrw.datalogger.UserStore.mLoadedUsers [private]

7.11.4.4 Logger ilearnrw.datalogger.UserStore.mLogger [private]

7.12 ilearnrw.games.GameAddTwoNumbers Class Reference

Collaboration diagram for ilearnrw.games.GameAddTwoNumbers:



Public Member Functions

- [GameAddTwoNumbers](#) ([User](#) user, [IDataLogger](#) logger)
- void [update](#) ()

Private Member Functions

- void [printOutActions](#) (List< [UserAction](#) > actions, [PrintStream](#) out)

Private Attributes

- [IDataLogger](#) mLogger

- [User mUser](#)
- [ApplicationId mApplicationId](#) = new [ApplicationId](#)("AddTwoNumbers")
- int [mUserId](#)
- String [mWelcomeMessage](#) = "Welcome, you are now playing 'AddTwoNumbers'"
- List< [UserAction](#) > [mUserActions](#) = null

7.12.1 Constructor & Destructor Documentation

7.12.1.1 `ilearnrw.games.GameAddTwoNumbers.GameAddTwoNumbers (User user, IDataLogger logger)`

7.12.2 Member Function Documentation

7.12.2.1 `void ilearnrw.games.GameAddTwoNumbers.printOutActions (List< UserAction > actions, PrintStream out)`
[private]

7.12.2.2 `void ilearnrw.games.GameAddTwoNumbers.update ()`

7.12.3 Member Data Documentation

7.12.3.1 `ApplicationId ilearnrw.games.GameAddTwoNumbers.mApplicationId = new ApplicationId("AddTwoNumbers")`
[private]

7.12.3.2 `IDataLogger ilearnrw.games.GameAddTwoNumbers.mLogger` [private]

7.12.3.3 `User ilearnrw.games.GameAddTwoNumbers.mUser` [private]

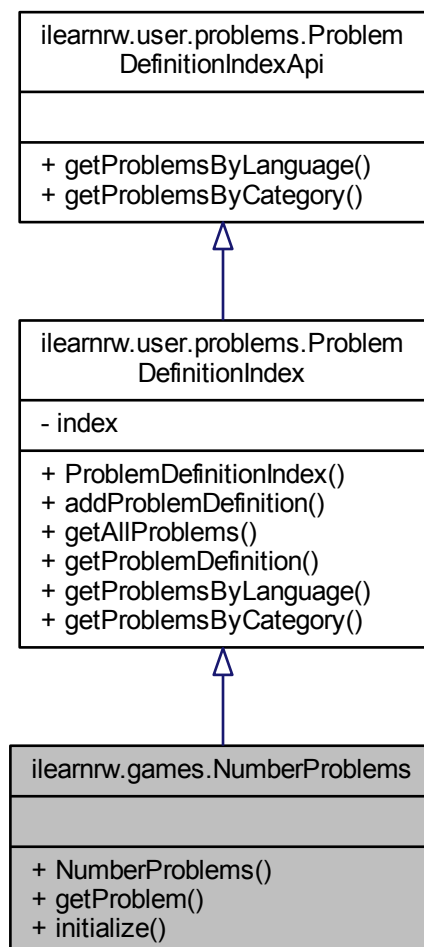
7.12.3.4 `List<UserAction> ilearnrw.games.GameAddTwoNumbers.mUserActions = null` [private]

7.12.3.5 `int ilearnrw.games.GameAddTwoNumbers.mUserId` [private]

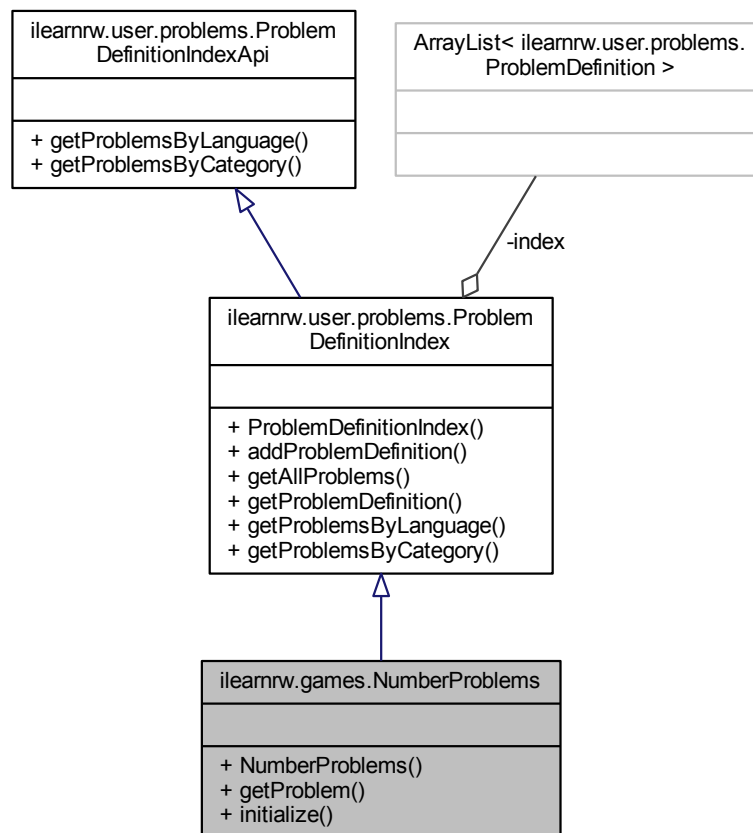
7.12.3.6 `String ilearnrw.games.GameAddTwoNumbers.mWelcomeMessage = "Welcome, you are now playing 'AddTwoNumbers'"`
[private]

7.13 ilearnrw.games.NumberProblems Class Reference

Inheritance diagram for ilearnrw.games.NumberProblems:



Collaboration diagram for ilearnrw.games.NumberProblems:



Public Member Functions

- [NumberProblems](#) ()
- void [getProblem](#) (int idx)
- void [initialize](#) ()

7.13.1 Constructor & Destructor Documentation

7.13.1.1 `ilearnrw.games.NumberProblems.NumberProblems ()`

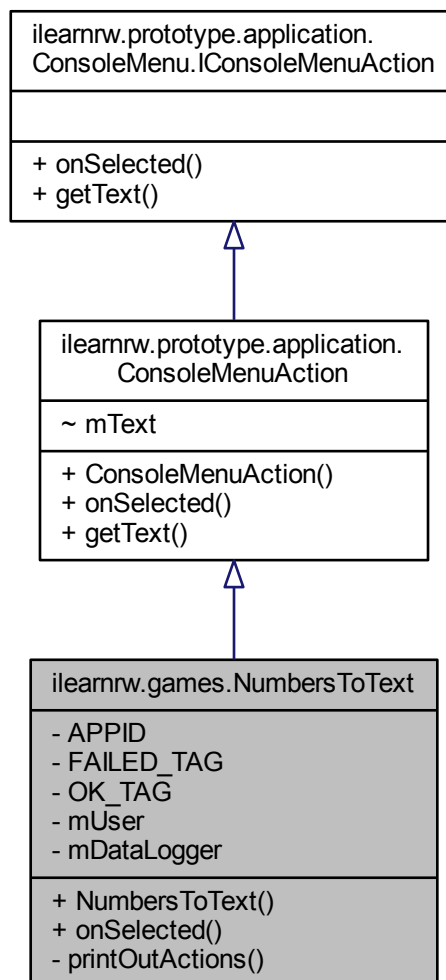
7.13.2 Member Function Documentation

7.13.2.1 `void ilearnrw.games.NumberProblems.getProblem (int idx)`

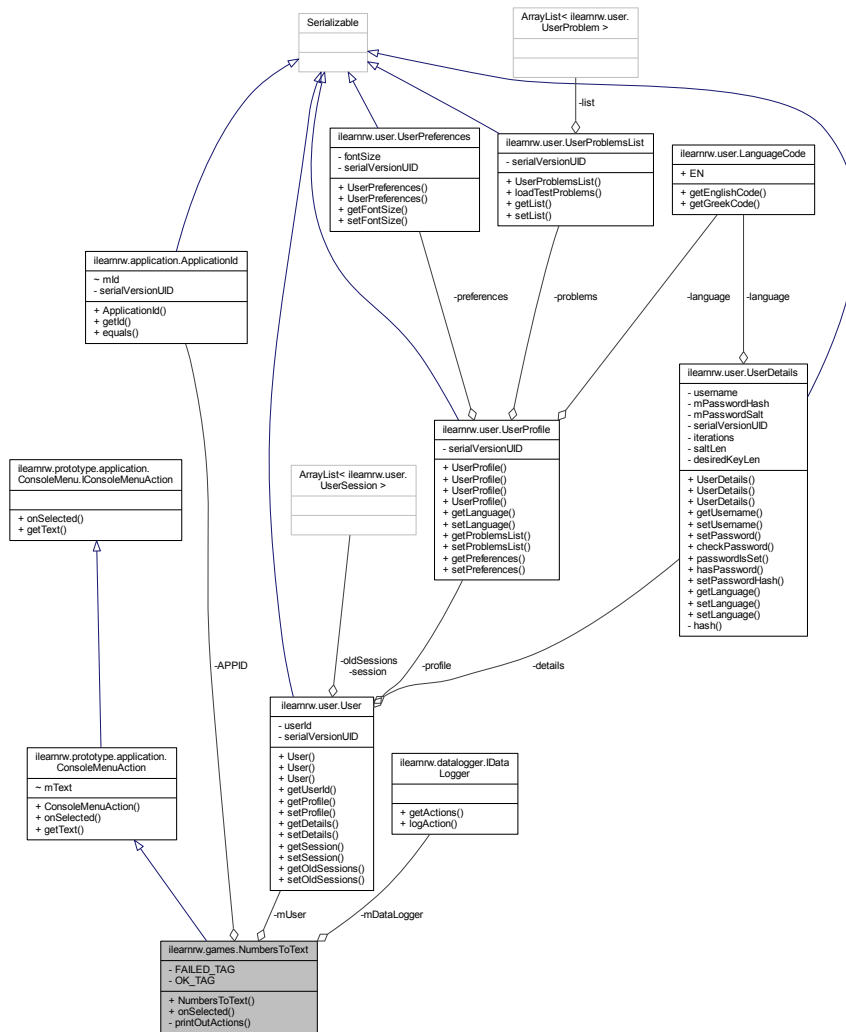
7.13.2.2 `void ilearnrw.games.NumberProblems.initialize ()`

7.14 ilearnrw.games.NumbersToText Class Reference

Inheritance diagram for ilearnrw.games.NumbersToText:



Collaboration diagram for ilearnrw.games.NumbersToText:



Public Member Functions

- [NumbersToText](#) (String text, [User](#) user, [IDataLogger](#) dataLogger)
- [EConsoleMenuActionResult](#) onSelected ([ConsoleMenu](#) menu)

Private Member Functions

- void [printOutActions](#) (List< [UserAction](#) > actions, [PrintStream](#) out)

Private Attributes

- final [ApplicationId](#) APPID = new [ApplicationId](#)("NUMBERSTOTEXT")
- final String FAILED_TAG = "FAILED"
- final String OK_TAG = "OK"
- [User](#) mUser
- [IDataLogger](#) mDataLogger

Additional Inherited Members

7.14.1 Constructor & Destructor Documentation

7.14.1.1 `ilearnrw.games.NumbersToText.NumbersToText (String text, User user, IDataLogger dataLogger)`

7.14.2 Member Function Documentation

7.14.2.1 `EConsoleMenuActionResult ilearnrw.games.NumbersToText.onSelected (ConsoleMenu menu)`

Todo : it's the responsibility of the datalogger to save the actions

Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

7.14.2.2 `void ilearnrw.games.NumbersToText.printOutActions (List< UserAction > actions, PrintStream out)`
[private]

7.14.3 Member Data Documentation

7.14.3.1 `final ApplicationId ilearnrw.games.NumbersToText.APPID = new ApplicationId("NUMBERSTOTEXT")`
[private]

7.14.3.2 `final String ilearnrw.games.NumbersToText.FAILED_TAG = "FAILED"` [private]

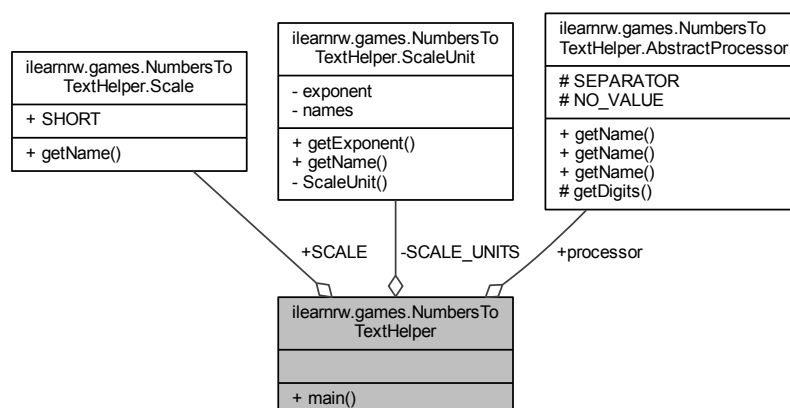
7.14.3.3 `IDataLogger ilearnrw.games.NumbersToText.mDataLogger` [private]

7.14.3.4 `User ilearnrw.games.NumbersToText.mUser` [private]

7.14.3.5 `final String ilearnrw.games.NumbersToText.OK_TAG = "OK"` [private]

7.15 ilearnrw.games.NumbersToTextHelper Class Reference

Collaboration diagram for ilearnrw.games.NumbersToTextHelper:



Classes

- class [AbstractProcessor](#)
- class [CompositeBigProcessor](#)
- class [DefaultProcessor](#)
- class [HundredProcessor](#)
- enum [Scale](#)
- class [ScaleUnit](#)
- class [TensProcessor](#)
- class [UnitProcessor](#)

Static Public Member Functions

- static void [main](#) (String...args)

Static Public Attributes

- static [Scale](#) [SCALE](#) = [Scale.SHORT](#)
- static [AbstractProcessor](#) [processor](#)

Static Private Attributes

- static [ScaleUnit](#)[] [SCALE_UNITS](#)

7.15.1 Detailed Description

This class will convert numeric values into an english representation

For units, see : <http://www.jimloy.com/math/billion.htm>

Author

yanick.rochon@gmail.com

7.15.2 Member Function Documentation

7.15.2.1 static void `ilearnrw.games.NumbersToTextHelper.main (String... args)` `[static]`

7.15.3 Member Data Documentation

7.15.3.1 `AbstractProcessor` `ilearnrw.games.NumbersToTextHelper.processor` `[static]`

7.15.3.2 `Scale` `ilearnrw.games.NumbersToTextHelper.SCALE = Scale.SHORT` `[static]`

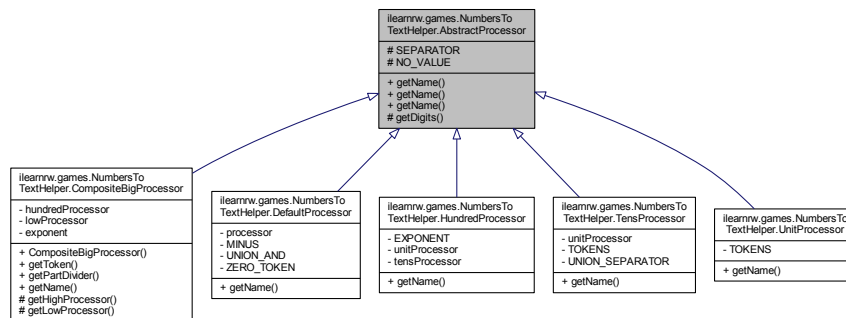
Change this scale to support American and modern British value (short scale) or Traditional British value (long scale)

7.15.3.3 `ScaleUnit []` `ilearnrw.games.NumbersToTextHelper.SCALE_UNITS` `[static], [private]`

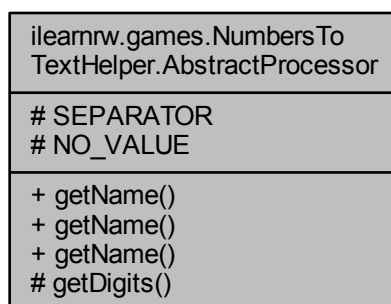
See http://www.wordiq.com/definition/Names_of_large_numbers

7.16 ilearnrw.games.NumbersToTextHelper.AbstractProcessor Class Reference

Inheritance diagram for ilearnrw.games.NumbersToTextHelper.AbstractProcessor:



Collaboration diagram for ilearnrw.games.NumbersToTextHelper.AbstractProcessor:



Public Member Functions

- String `getName` (long value)
- String `getName` (double value)
- abstract String `getName` (String value)

Protected Member Functions

- List< Integer > `getDigits` (long value)

Static Protected Attributes

- static final String `SEPARATOR` = " "
- static final int `NO_VALUE` = -1

7.16.1 Member Function Documentation

7.16.1.1 `List<Integer> ilearnrw.games.NumbersToTextHelper.AbstractProcessor.getDigits (long value)` [protected]

7.16.1.2 `String ilearnrw.games.NumbersToTextHelper.AbstractProcessor.getName (long value)`

7.16.1.3 `String ilearnrw.games.NumbersToTextHelper.AbstractProcessor.getName (double value)`

7.16.1.4 `abstract String ilearnrw.games.NumbersToTextHelper.AbstractProcessor.getName (String value)` [pure virtual]

Implemented in [ilearnrw.games.NumbersToTextHelper.DefaultProcessor](#), [ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor](#), [ilearnrw.games.NumbersToTextHelper.HundredProcessor](#), [ilearnrw.games.NumbersToTextHelper.TensProcessor](#), and [ilearnrw.games.NumbersToTextHelper.UnitProcessor](#).

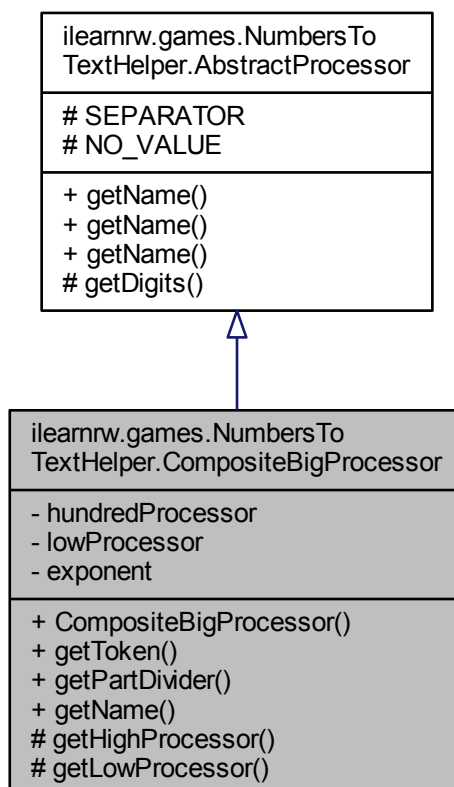
7.16.2 Member Data Documentation

7.16.2.1 `final int ilearnrw.games.NumbersToTextHelper.AbstractProcessor.NO_VALUE = -1` [static], [protected]

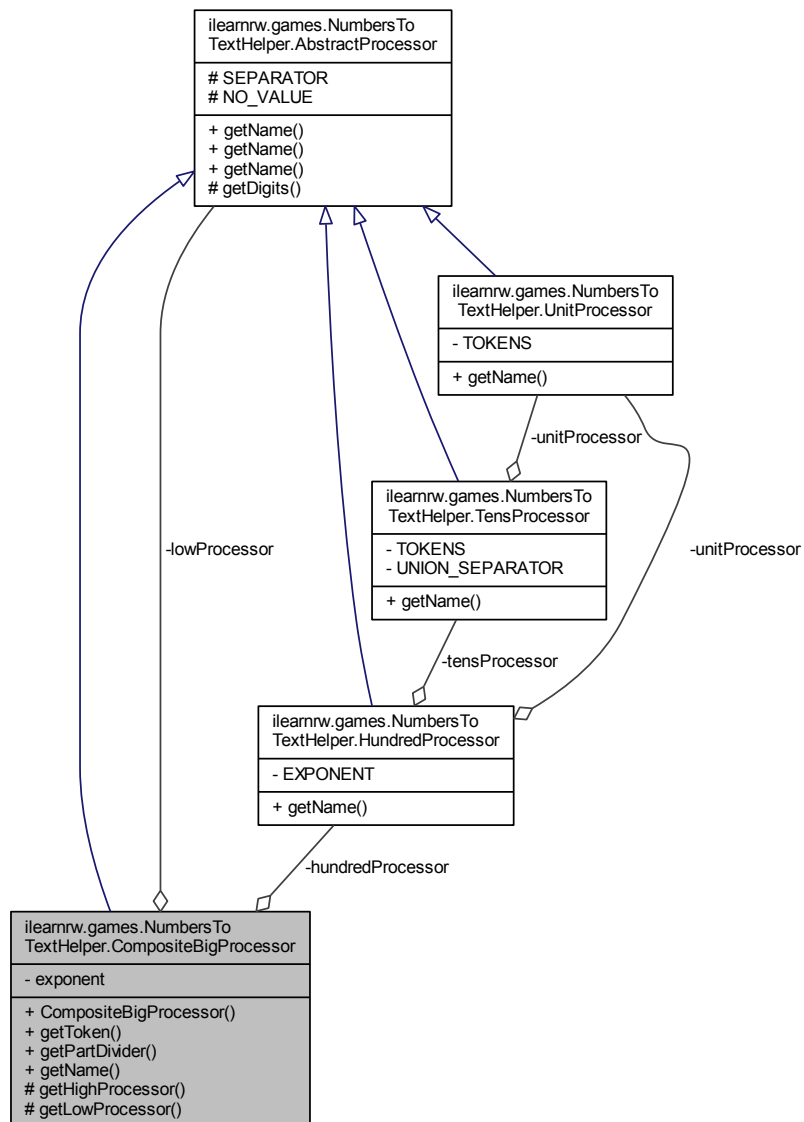
7.16.2.2 `final String ilearnrw.games.NumbersToTextHelper.AbstractProcessor.SEPARATOR = " "` [static], [protected]

7.17 ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor Class Reference

Inheritance diagram for ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor:



Collaboration diagram for ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor:



Public Member Functions

- [CompositeBigProcessor](#) (int [exponent](#))
- String [getToken](#) ()
- int [getPartDivider](#) ()
- String [getName](#) (String value)

Protected Member Functions

- [AbstractProcessor](#) [getHighProcessor](#) ()
- [AbstractProcessor](#) [getLowProcessor](#) ()

Private Attributes

- `HundredProcessor hundredProcessor = new HundredProcessor()`
- `AbstractProcessor lowProcessor`
- `int exponent`

Additional Inherited Members

7.17.1 Constructor & Destructor Documentation

7.17.1.1 `ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.CompositeBigProcessor (int exponent)`

7.17.2 Member Function Documentation

7.17.2.1 `AbstractProcessor ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.getHighProcessor ()`
[protected]

7.17.2.2 `AbstractProcessor ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.getLowProcessor ()`
[protected]

7.17.2.3 `String ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.getName (String value)` [virtual]

Implements `ilearnrw.games.NumbersToTextHelper.AbstractProcessor`.

7.17.2.4 `int ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.getPartDivider ()`

7.17.2.5 `String ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.getToken ()`

7.17.3 Member Data Documentation

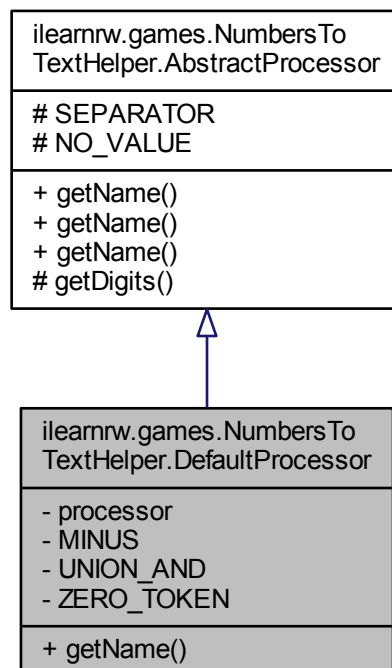
7.17.3.1 `int ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.exponent` [private]

7.17.3.2 `HundredProcessor ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.hundredProcessor = new HundredProcessor()` [private]

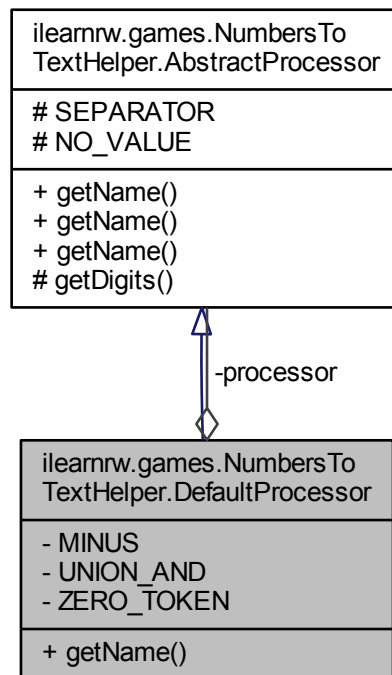
7.17.3.3 `AbstractProcessor ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor.lowProcessor` [private]

7.18 ilearnrw.games.NumbersToTextHelper.DefaultProcessor Class Reference

Inheritance diagram for ilearnrw.games.NumbersToTextHelper.DefaultProcessor:



Collaboration diagram for `ilearnrw.games.NumbersToTextHelper.DefaultProcessor`:



Public Member Functions

- String `getName` (String value)

Private Attributes

- `AbstractProcessor processor` = new `CompositeBigProcessor`(63)

Static Private Attributes

- static String `MINUS` = "minus"
- static String `UNION_AND` = "and"
- static String `ZERO_TOKEN` = "zero"

Additional Inherited Members

7.18.1 Member Function Documentation

7.18.1.1 String `ilearnrw.games.NumbersToTextHelper.DefaultProcessor.getName` (String *value*) [virtual]

Implements `ilearnrw.games.NumbersToTextHelper.AbstractProcessor`.

7.18.2 Member Data Documentation

7.18.2.1 `String ilearnrw.games.NumbersToTextHelper.DefaultProcessor.MINUS = "minus"` `[static]`, `[private]`

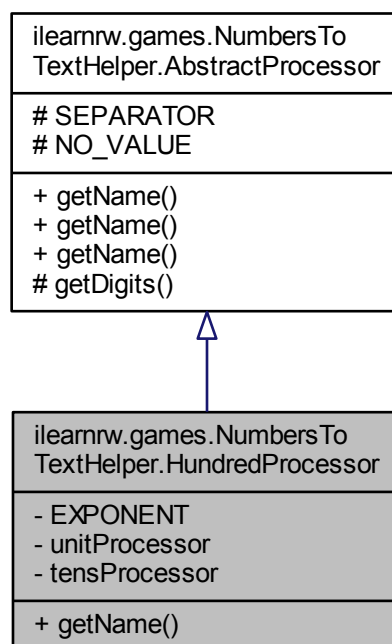
7.18.2.2 `AbstractProcessor ilearnrw.games.NumbersToTextHelper.DefaultProcessor.processor = new CompositeBigProcessor(63)` `[private]`

7.18.2.3 `String ilearnrw.games.NumbersToTextHelper.DefaultProcessor.UNION_AND = "and"` `[static]`, `[private]`

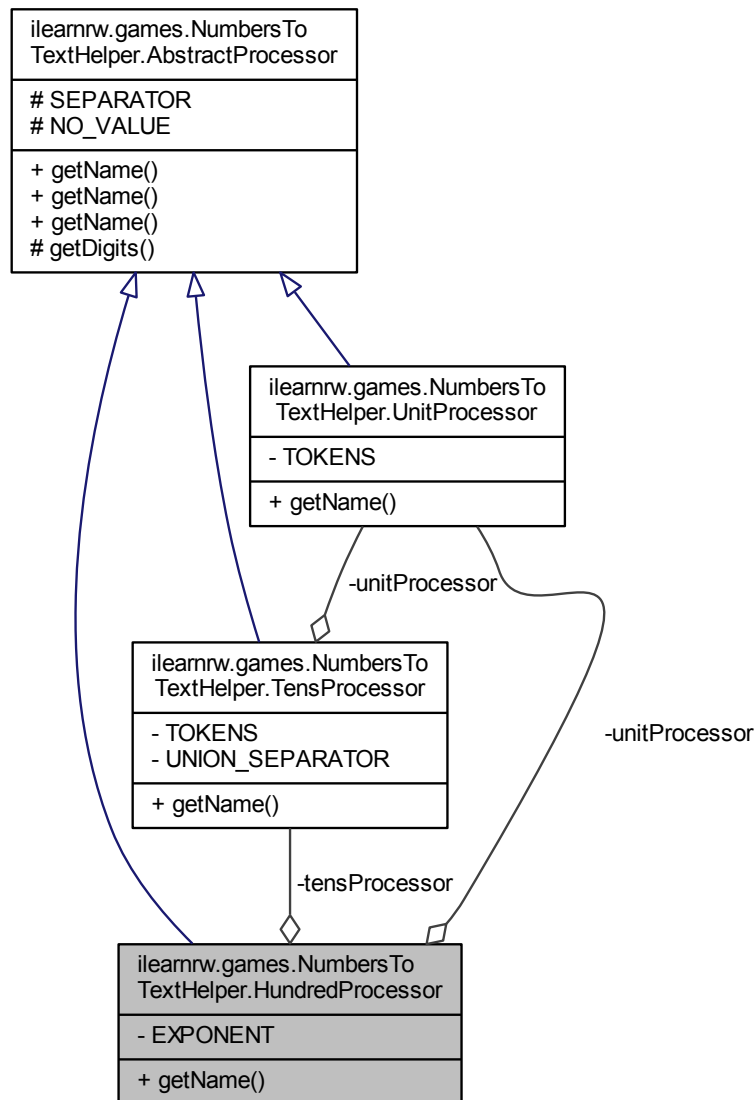
7.18.2.4 `String ilearnrw.games.NumbersToTextHelper.DefaultProcessor.ZERO_TOKEN = "zero"` `[static]`, `[private]`

7.19 ilearnrw.games.NumbersToTextHelper.HundredProcessor Class Reference

Inheritance diagram for `ilearnrw.games.NumbersToTextHelper.HundredProcessor`:



Collaboration diagram for `ilearnrw.games.NumbersToTextHelper.HundredProcessor`:



Public Member Functions

- String `getName` (String value)

Private Attributes

- int `EXPONENT` = 2
- `UnitProcessor unitProcessor` = new `UnitProcessor`()
- `TensProcessor tensProcessor` = new `TensProcessor`()

Additional Inherited Members

7.19.1 Member Function Documentation

7.19.1.1 String ilearnrw.games.NumbersToTextHelper.HundredProcessor.getName (String *value*) [virtual]

Implements [ilearnrw.games.NumbersToTextHelper.AbstractProcessor](#).

7.19.2 Member Data Documentation

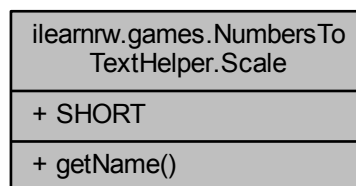
7.19.2.1 int ilearnrw.games.NumbersToTextHelper.HundredProcessor.EXPONENT = 2 [private]

7.19.2.2 **TensProcessor** ilearnrw.games.NumbersToTextHelper.HundredProcessor.tensProcessor = new **TensProcessor**() [private]

7.19.2.3 **UnitProcessor** ilearnrw.games.NumbersToTextHelper.HundredProcessor.unitProcessor = new **UnitProcessor**() [private]

7.20 ilearnrw.games.NumbersToTextHelper.Scale Enum Reference

Collaboration diagram for ilearnrw.games.NumbersToTextHelper.Scale:



Public Member Functions

- String [getName](#) (int exponent)

Public Attributes

- [SHORT](#)

7.20.1 Member Function Documentation

7.20.1.1 String ilearnrw.games.NumbersToTextHelper.Scale.getName (int *exponent*)

7.20.2 Member Data Documentation

7.20.2.1 ilearnrw.games.NumbersToTextHelper.Scale.SHORT

7.21 ilearnrw.games.NumbersToTextHelper.ScaleUnit Class Reference

Collaboration diagram for ilearnrw.games.NumbersToTextHelper.ScaleUnit:

ilearnrw.games.NumbersToTextHelper.ScaleUnit
- exponent - names
+ getExponent() + getName() - ScaleUnit()

Public Member Functions

- int [getExponent](#) ()
- String [getName](#) (int index)

Private Member Functions

- [ScaleUnit](#) (int [exponent](#), String...names)

Private Attributes

- int [exponent](#)
- String[] [names](#)

7.21.1 Constructor & Destructor Documentation

7.21.1.1 ilearnrw.games.NumbersToTextHelper.ScaleUnit.ScaleUnit (int *exponent*, String... *names*) [private]

7.21.2 Member Function Documentation

7.21.2.1 int ilearnrw.games.NumbersToTextHelper.ScaleUnit.getExponent ()

7.21.2.2 String ilearnrw.games.NumbersToTextHelper.ScaleUnit.getName (int *index*)

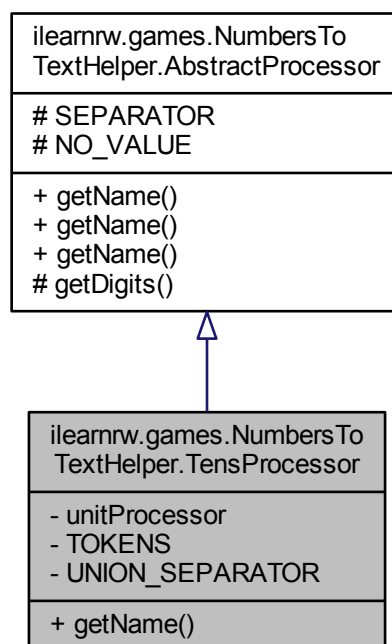
7.21.3 Member Data Documentation

7.21.3.1 int ilearnrw.games.NumbersToTextHelper.ScaleUnit.exponent [private]

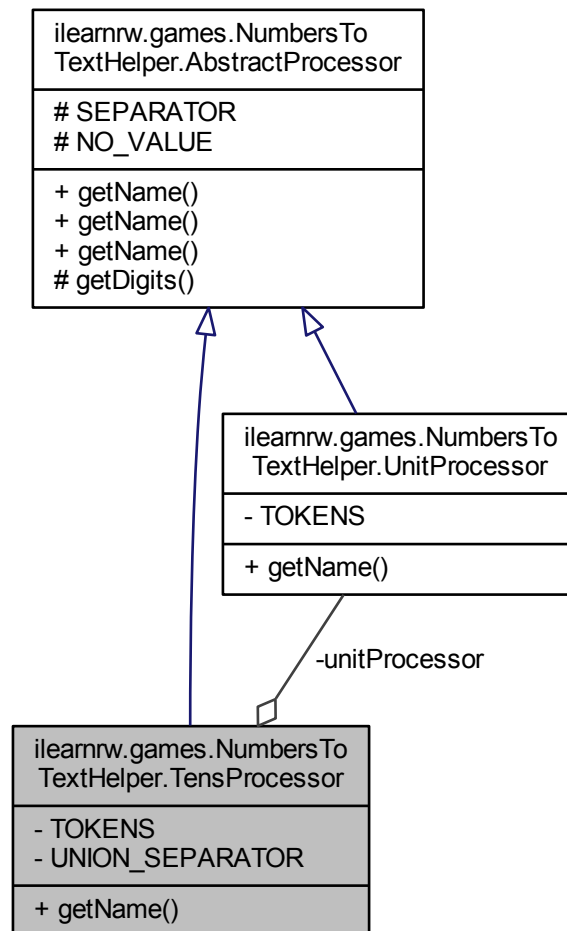
7.21.3.2 String [] ilearnrw.games.NumbersToTextHelper.ScaleUnit.names [private]

7.22 ilearnrw.games.NumbersToTextHelper.TensProcessor Class Reference

Inheritance diagram for ilearnrw.games.NumbersToTextHelper.TensProcessor:



Collaboration diagram for `ilearnrw.games.NumbersToTextHelper.TensProcessor`:



Public Member Functions

- String `getName` (String value)

Private Attributes

- `UnitProcessor unitProcessor = new UnitProcessor()`

Static Private Attributes

- static final String[] `TOKENS`
- static final String `UNION_SEPARATOR` = "-"

Additional Inherited Members

7.22.1 Member Function Documentation

7.22.1.1 `String ilearnrw.games.NumbersToTextHelper.TensProcessor.getName (String value)` [virtual]

Implements [ilearnrw.games.NumbersToTextHelper.AbstractProcessor](#).

7.22.2 Member Data Documentation

7.22.2.1 `final String [] ilearnrw.games.NumbersToTextHelper.TensProcessor.TOKENS` [static], [private]

Initial value:

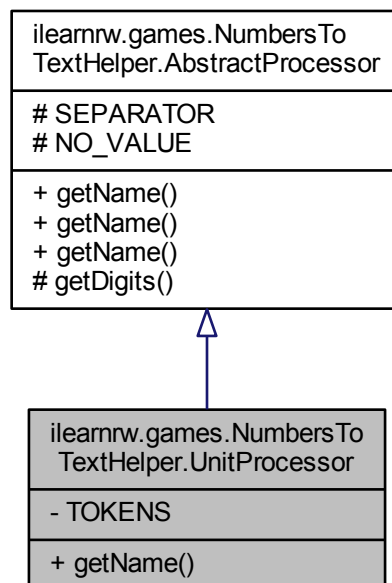
```
= new String[] {
    "twenty", "thirty", "fourty", "fifty", "sixty", "seventy", "eighty", "ninety"
}
```

7.22.2.2 `final String ilearnrw.games.NumbersToTextHelper.TensProcessor.UNION_SEPARATOR = "-"` [static], [private]

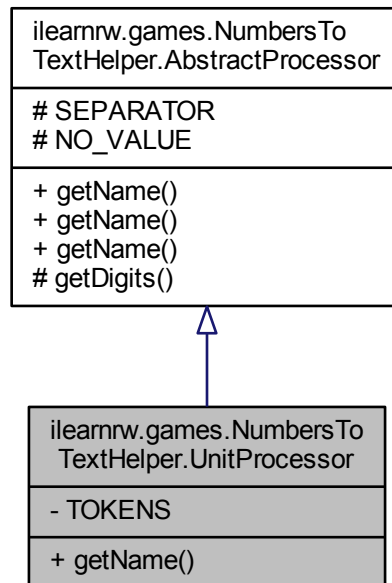
7.22.2.3 `UnitProcessor ilearnrw.games.NumbersToTextHelper.TensProcessor.unitProcessor = new UnitProcessor()` [private]

7.23 ilearnrw.games.NumbersToTextHelper.UnitProcessor Class Reference

Inheritance diagram for `ilearnrw.games.NumbersToTextHelper.UnitProcessor`:



Collaboration diagram for `ilearnrw.games.NumbersToTextHelper.UnitProcessor`:



Public Member Functions

- String [getName](#) (String value)

Static Private Attributes

- static final String[] [TOKENS](#)

Additional Inherited Members

7.23.1 Member Function Documentation

7.23.1.1 String `ilearnrw.games.NumbersToTextHelper.UnitProcessor.getName (String value)` [virtual]

Implements [ilearnrw.games.NumbersToTextHelper.AbstractProcessor](#).

7.23.2 Member Data Documentation

7.23.2.1 final String [] `ilearnrw.games.NumbersToTextHelper.UnitProcessor.TOKENS` [static],[private]

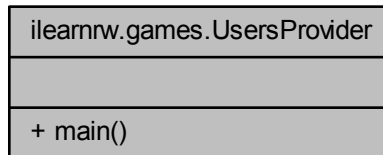
Initial value:

```

= new String[] {
    "one", "two", "three", "four", "five", "six", "seven", "eight", "nine",
    "ten", "eleven", "twelve", "thirteen", "fourteen", "fifteen", "sixteen", "seventeen", "eighteen",
    "nineteen"
}
  
```

7.24 ilearnrw.games.UsersProvider Class Reference

Collaboration diagram for ilearnrw.games.UsersProvider:



Static Public Member Functions

- static void `main` (String[] args)

7.24.1 Member Function Documentation

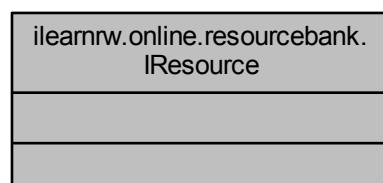
7.24.1.1 static void `ilearnrw.games.UsersProvider.main` (String[] *args*) [static]

Parameters

<i>args</i>	
-------------	--

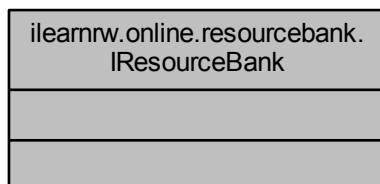
7.25 ilearnrw.online.resourcebank.IResource Interface Reference

Collaboration diagram for ilearnrw.online.resourcebank.IResource:



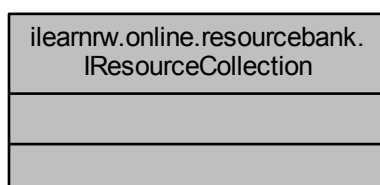
7.26 ilearnrw.online.resourcebank.IResourceBank Interface Reference

Collaboration diagram for ilearnrw.online.resourcebank.IResourceBank:



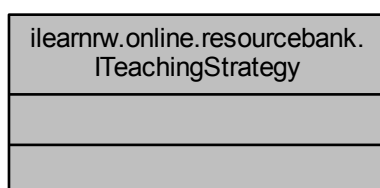
7.27 ilearnrw.online.resourcebank.IResourceCollection Interface Reference

Collaboration diagram for ilearnrw.online.resourcebank.IResourceCollection:



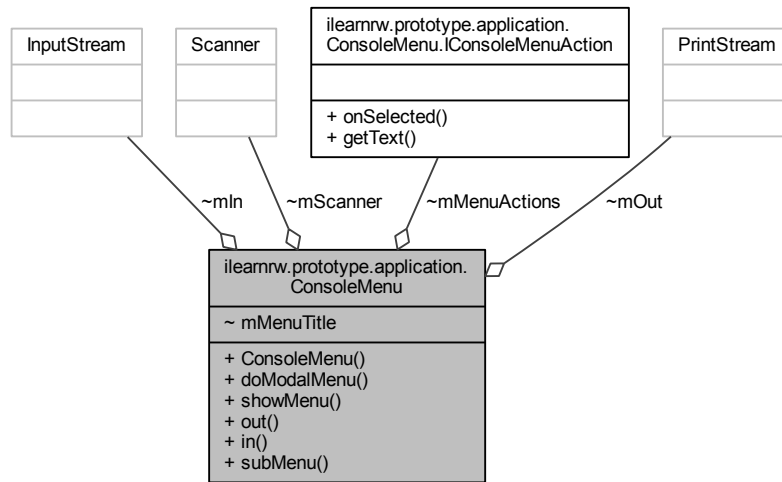
7.28 ilearnrw.online.resourcebank.ITeachingStrategy Interface Reference

Collaboration diagram for ilearnrw.online.resourcebank.ITeachingStrategy:



7.29 ilearnrw.prototype.application.ConsoleMenu Class Reference

Collaboration diagram for ilearnrw.prototype.application.ConsoleMenu:



Classes

- enum [EConsoleMenuActionResult](#)
- interface [IConsoleMenuAction](#)

Public Member Functions

- [ConsoleMenu](#) (PrintStream outputStream, InputStream inputStream, String menuTitle, [IConsoleMenuAction](#)[] menuActions)
- void [doModalMenu](#) ()
- [EConsoleMenuActionResult](#) [showMenu](#) ()
- PrintStream [out](#) ()
- Scanner [in](#) ()
- [ConsoleMenu](#) [subMenu](#) (String menuTitle, [IConsoleMenuAction](#)[] menuActions)

Package Attributes

- PrintStream [mOut](#)
- InputStream [mIn](#)
- Scanner [mScanner](#)
- String [mMenuTitle](#)
- [IConsoleMenuAction](#)[] [mMenuActions](#)

7.29.1 Constructor & Destructor Documentation

- 7.29.1.1 [ilearnrw.prototype.application.ConsoleMenu.ConsoleMenu](#) (PrintStream *outStream*, InputStream *inStream*, String *menuTitle*, [IConsoleMenuAction](#)[] *menuActions*)

7.29.2 Member Function Documentation

7.29.2.1 void `ilearnrw.prototype.application.ConsoleMenu.doModalMenu` ()

7.29.2.2 Scanner `ilearnrw.prototype.application.ConsoleMenu.in` ()

7.29.2.3 `PrintStream` `ilearnrw.prototype.application.ConsoleMenu.out` ()

7.29.2.4 `EConsoleMenuActionResult` `ilearnrw.prototype.application.ConsoleMenu.showMenu` ()

7.29.2.5 `ConsoleMenu` `ilearnrw.prototype.application.ConsoleMenu.subMenu` (*String menuTitle*, `IConsoleMenuAction[] menuActions`)

7.29.3 Member Data Documentation

7.29.3.1 `InputStream` `ilearnrw.prototype.application.ConsoleMenu.mIn` [package]

7.29.3.2 `IConsoleMenuAction []` `ilearnrw.prototype.application.ConsoleMenu.mMenuActions` [package]

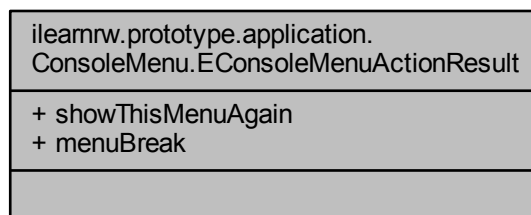
7.29.3.3 `String` `ilearnrw.prototype.application.ConsoleMenu.mMenuTitle` [package]

7.29.3.4 `PrintStream` `ilearnrw.prototype.application.ConsoleMenu.mOut` [package]

7.29.3.5 Scanner `ilearnrw.prototype.application.ConsoleMenu.mScanner` [package]

7.30 ilearnrw.prototype.application.ConsoleMenu.EConsoleMenuActionResult Enum Reference

Collaboration diagram for `ilearnrw.prototype.application.ConsoleMenu.EConsoleMenuActionResult`:



Public Attributes

- [showThisMenuAgain](#)
- [menuBreak](#)

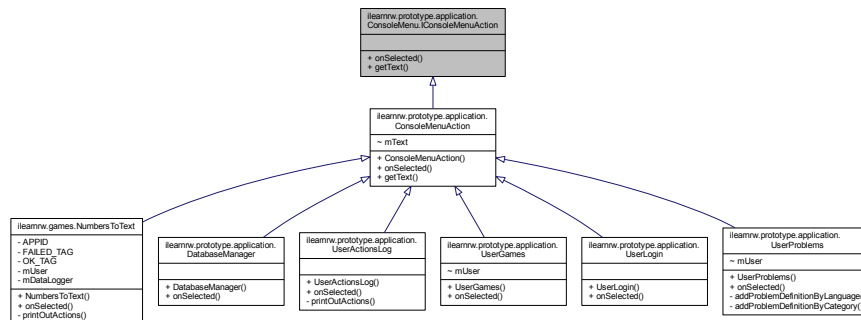
7.30.1 Member Data Documentation

7.30.1.1 `ilearnrw.prototype.application.ConsoleMenu.EConsoleMenuActionResult.menuBreak`

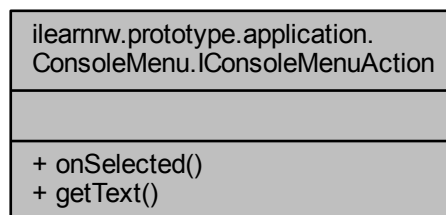
7.30.1.2 ilearnrw.prototype.application.ConsoleMenu.EConsoleMenuActionResult.showThisMenuAgain

7.31 ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction Interface Reference

Inheritance diagram for ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction:



Collaboration diagram for ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction:



Public Member Functions

- [EConsoleMenuActionResult onSelected \(ConsoleMenu menu\)](#)
- [String getText \(\)](#)

7.31.1 Member Function Documentation

7.31.1.1 String ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction.getText ()

Implemented in [ilearnrw.prototype.application.ConsoleMenuAction](#).

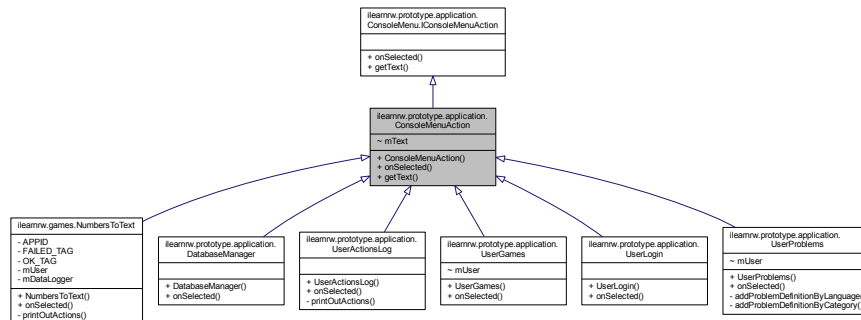
7.31.1.2 EConsoleMenuActionResult ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction.onSelected (ConsoleMenu menu)

Implemented in [ilearnrw.games.NumbersToText](#), [ilearnrw.prototype.application.UserProblems](#), [ilearnrw.prototype.application.UserActionsLog](#), [ilearnrw.prototype.application.DatabaseManager](#), [ilearnrw.prototype.application.UserGames](#), [ilearnrw.prototype.application.UserLogin](#), and [ilearnrw.prototype.application.UserProblems](#).

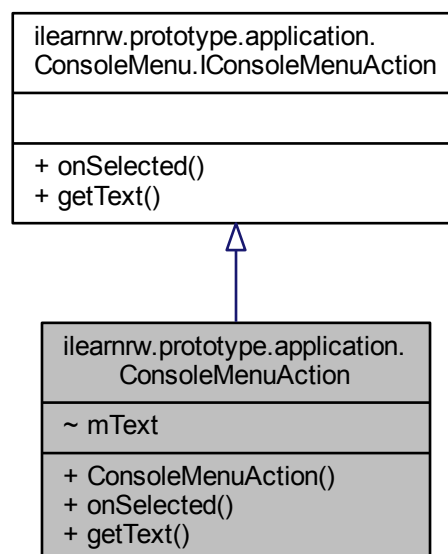
Games, [ilearnrw.prototype.application.UserLogin](#), and [ilearnrw.prototype.application.ConsoleMenuAction](#).

7.32 ilearnrw.prototype.application.ConsoleMenuAction Class Reference

Inheritance diagram for [ilearnrw.prototype.application.ConsoleMenuAction](#):



Collaboration diagram for [ilearnrw.prototype.application.ConsoleMenuAction](#):



Public Member Functions

- [ConsoleMenuAction](#) (String text)
- [EConsoleMenuActionResult](#) [onSelected](#) ([ConsoleMenu](#) menu)
- String [getText](#) ()

Package Attributes

- String [mText](#)

7.32.1 Constructor & Destructor Documentation

7.32.1.1 `ilearnrw.prototype.application.ConsoleMenuAction.ConsoleMenuAction (String text)`

7.32.2 Member Function Documentation

7.32.2.1 `String ilearnrw.prototype.application.ConsoleMenuAction.getText ()`

Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

7.32.2.2 `EConsoleMenuActionResult ilearnrw.prototype.application.ConsoleMenuAction.onSelected (ConsoleMenu menu)`

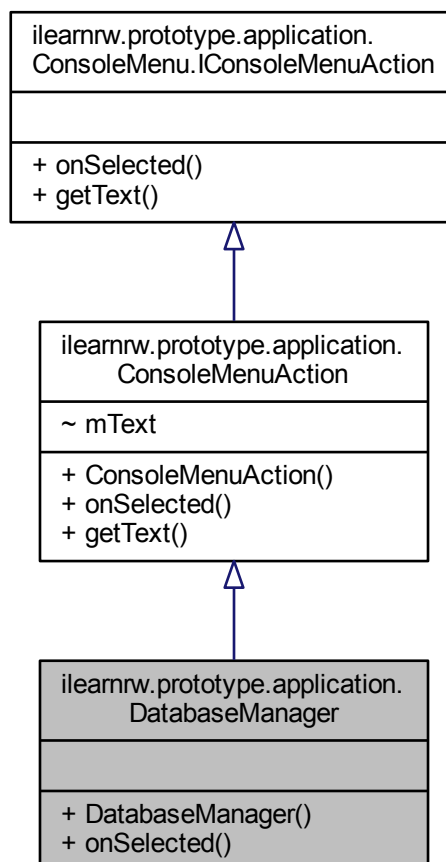
Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

7.32.3 Member Data Documentation

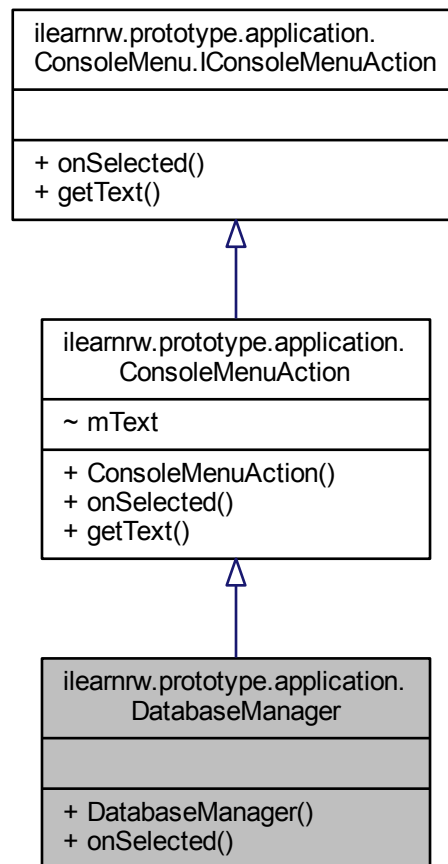
7.32.3.1 `String ilearnrw.prototype.application.ConsoleMenuAction.mText` [package]

7.33 ilearnrw.prototype.application.DatabaseManager Class Reference

Inheritance diagram for ilearnrw.prototype.application.DatabaseManager:



Collaboration diagram for ilearnrw.prototype.application.DatabaseManager:



Classes

- interface [ISelectUser](#)

Public Member Functions

- [DatabaseManager](#) (String text)
- [EConsoleMenuActionResult onSelected](#) ([ConsoleMenu](#) menu)

Additional Inherited Members

7.33.1 Constructor & Destructor Documentation

7.33.1.1 `ilearnrw.prototype.application.DatabaseManager.DatabaseManager (String text)`

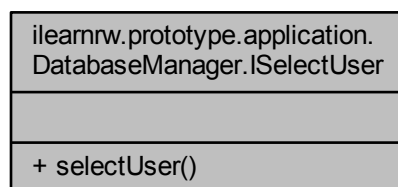
7.33.2 Member Function Documentation

7.33.2.1 **EConsoleMenuActionResult** `ilearnrw.prototype.application.DatabaseManager.onSelected (ConsoleMenu menu)`

Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

7.34 `ilearnrw.prototype.application.DatabaseManager.ISelectUser` Interface Reference

Collaboration diagram for `ilearnrw.prototype.application.DatabaseManager.ISelectUser`:



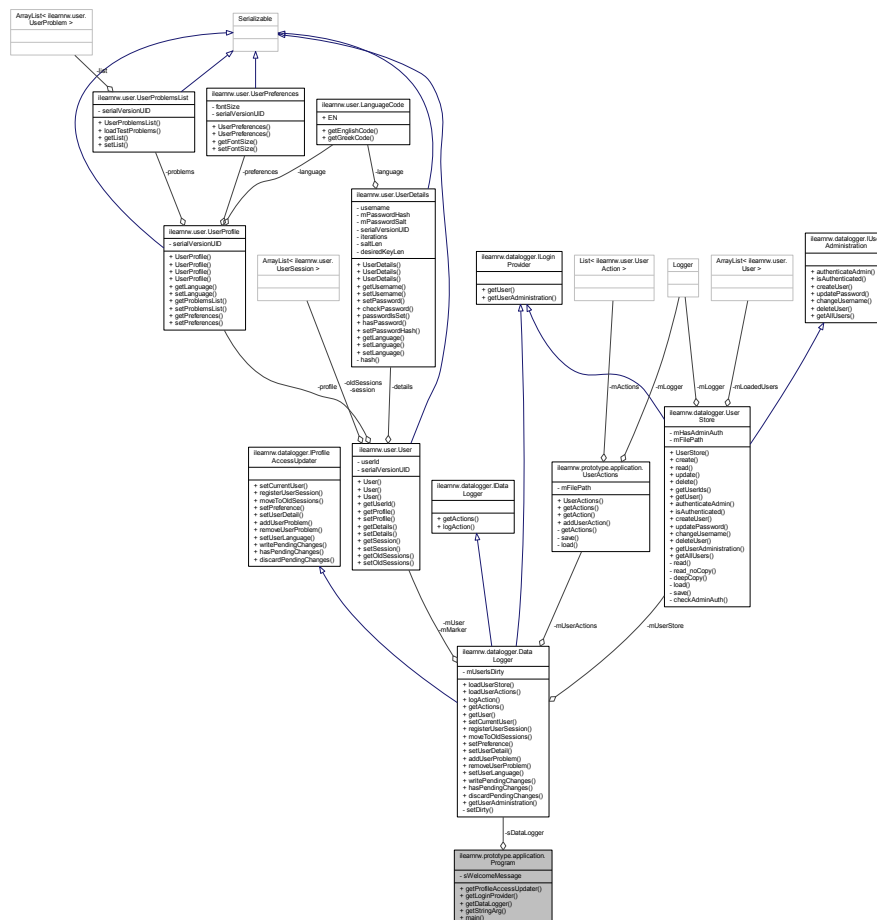
Public Member Functions

- [User selectUser](#) ([ConsoleMenu menu](#)) throws `AuthenticationException`

7.34.1 Member Function Documentation

7.34.1.1 **User** `ilearnrw.prototype.application.DatabaseManager.ISelectUser.selectUser (ConsoleMenu menu)` throws `AuthenticationException`

Collaboration diagram for ilearnrw.prototype.application.Program:



- static `IProfileAccessUpdater` `getProfileAccessUpdater` ()
- static `ILoginProvider` `getLoginProvider` ()
- static `IDataLogger` `getDataLogger` ()
- static `String` `getStringArg` (`String` name, `String[]` args) throws `Exception`
- static void `main` (`String[]` args)

- static `ILogger sDataLogger = new DataLogger()`
- static `String sWelcomeMessage`

7.35.1.1 static IDataLogger ilearnrw.prototype.application.Program.getLogger () [static]

Generated on Tue Aug 27 2013 15:25:20 for iLearnRW by Doxygen

7.35.1.3 **static** `IProfileAccessUpdater` `ilearnrw.prototype.application.Program.getProfileAccessUpdater ()` `[static]`

7.35.1.4 **static** `String` `ilearnrw.prototype.application.Program.getStringArg (String name, String[] args)` throws `Exception` `[static]`

7.35.1.5 **static void** `ilearnrw.prototype.application.Program.main (String[] args)` `[static]`

Parameters

<i>args</i>	Commandline arguments.
-------------	------------------------

Commandline arguments:

-db <filePath> Database file to use (required)

7.35.2 Member Data Documentation

7.35.2.1 **DataLogger** `ilearnrw.prototype.application.Program.sDataLogger = new DataLogger()` `[static]`, `[private]`

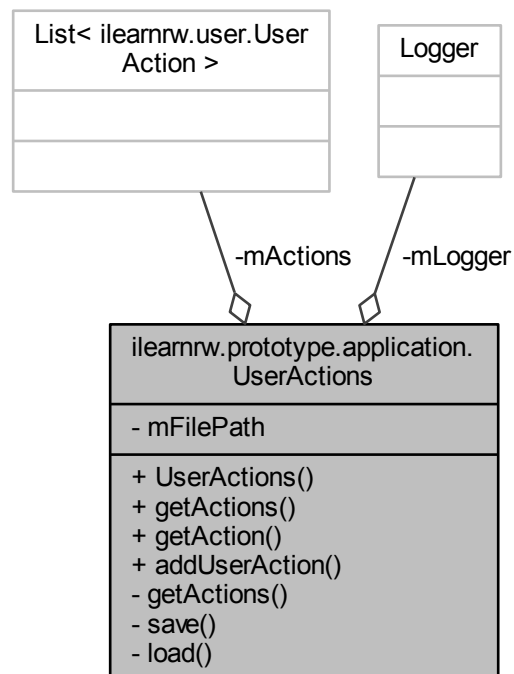
7.35.2.2 `String` `ilearnrw.prototype.application.Program.sWelcomeMessage` `[static]`, `[private]`

Initial value:

```
= "Welcome to the iLearnRW prototype application\n"
+ "\n"
+ "Developer notes:\n"
+ "\tUsernames are not unique.\n"
+ "\tUsernames cannot contain spaces.\n"
+ "\n"
```

7.36 ilearnrw.prototype.application.UserActions Class Reference

Collaboration diagram for ilearnrw.prototype.application.UserActions:



Public Member Functions

- [UserActions](#) (String filePath)
- List< [UserAction](#) > [getActions](#) ([UserActionFilter](#) filter)
- [UserAction](#) [getAction](#) (int i)
- void [addUserAction](#) ([UserAction](#) action)

Private Member Functions

- List< [UserAction](#) > [getActions](#) ()
- void [save](#) ()
- void [load](#) ()

Private Attributes

- List< [UserAction](#) > [mActions](#)
- Logger [mLogger](#)
- String [mFilePath](#)

7.36.1 Detailed Description

Todo `getActions(filter)` should not return null? It's easier to return just an empty list

7.36.2 Constructor & Destructor Documentation

7.36.2.1 `ilearnrw.prototype.application.UserActions.UserActions (String filePath)`

7.36.3 Member Function Documentation

7.36.3.1 `void ilearnrw.prototype.application.UserActions.addAction (UserAction action)`

7.36.3.2 `UserAction ilearnrw.prototype.application.UserActions.getAction (int i)`

7.36.3.3 `List<UserAction> ilearnrw.prototype.application.UserActions.getActions () [private]`

7.36.3.4 `List<UserAction> ilearnrw.prototype.application.UserActions.getActions (UserActionFilter filter)`

7.36.3.5 `void ilearnrw.prototype.application.UserActions.load () [private]`

7.36.3.6 `void ilearnrw.prototype.application.UserActions.save () [private]`

7.36.4 Member Data Documentation

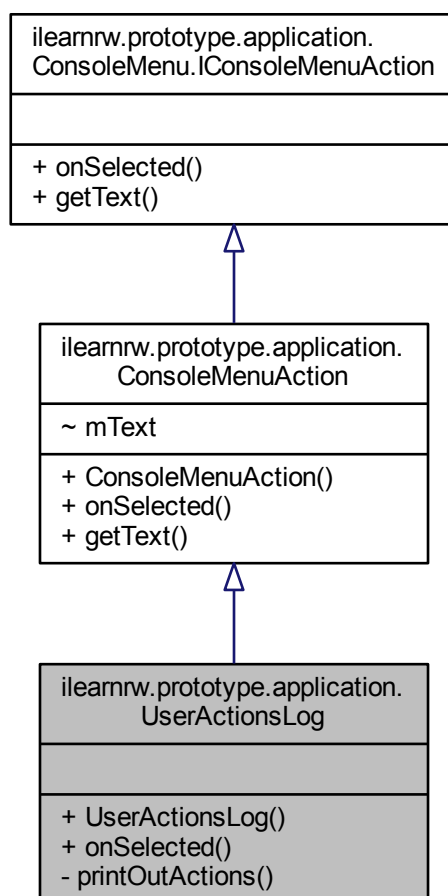
7.36.4.1 `List<UserAction> ilearnrw.prototype.application.UserActions.mActions [private]`

7.36.4.2 `String ilearnrw.prototype.application.UserActions.mFilePath [private]`

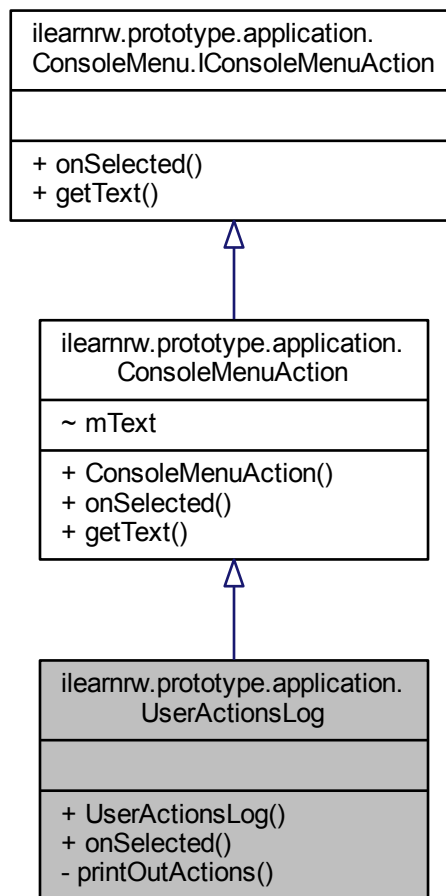
7.36.4.3 `Logger ilearnrw.prototype.application.UserActions.mLogger [private]`

7.37 ilearnrw.prototype.application.UserActionsLog Class Reference

Inheritance diagram for ilearnrw.prototype.application.UserActionsLog:



Collaboration diagram for `ilearnrw.prototype.application.UserActionsLog`:



Public Member Functions

- [UserActionsLog](#) (String text)
- [EConsoleMenuActionResult onSelected](#) ([ConsoleMenu](#) menu)

Private Member Functions

- void [printOutActions](#) (List< [UserAction](#) > actions, PrintStream out)

Additional Inherited Members

7.37.1 Constructor & Destructor Documentation

7.37.1.1 `ilearnrw.prototype.application.UserActionsLog.UserActionsLog (String text)`

7.37.2 Member Function Documentation

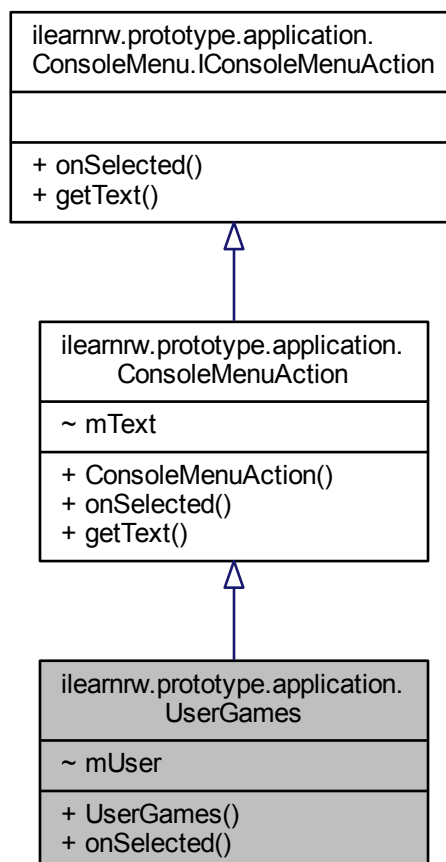
7.37.2.1 **EConsoleMenuActionResult** ilearnrw.prototype.application.UserActionsLog.onSelected (ConsoleMenu menu)

Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

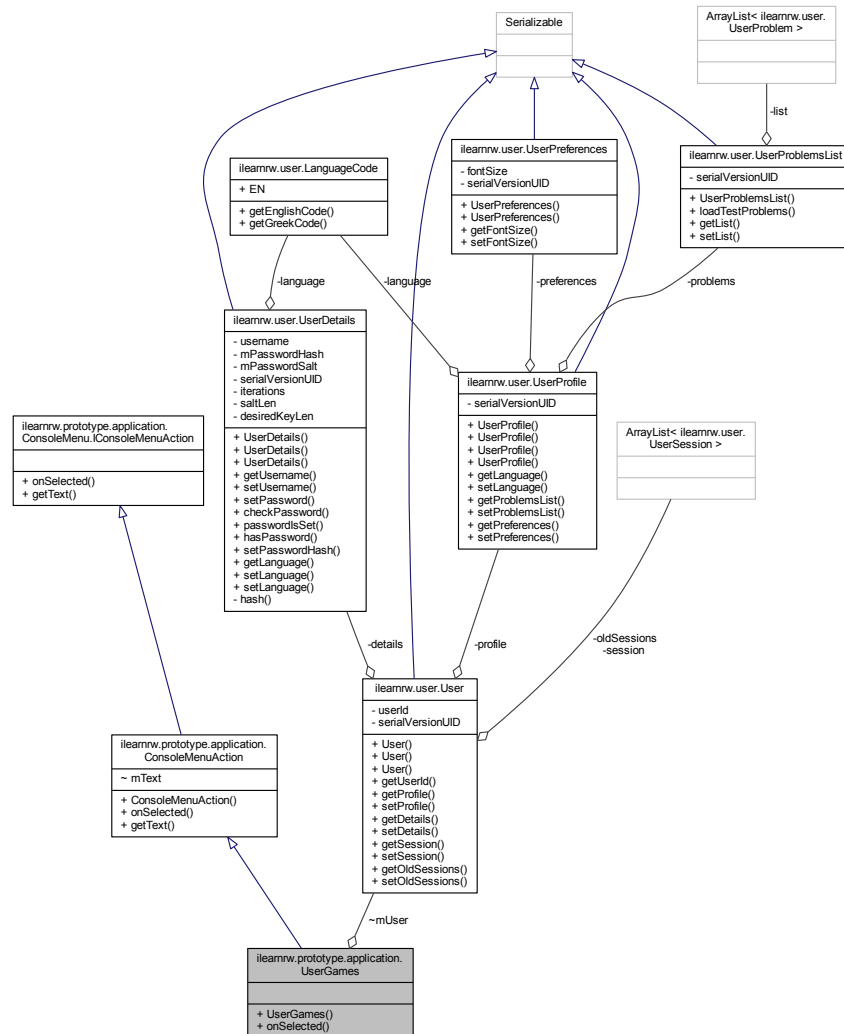
7.37.2.2 void ilearnrw.prototype.application.UserActionsLog.printOutActions (List< UserAction > actions, PrintStream out) [private]

7.38 ilearnrw.prototype.application.UserGames Class Reference

Inheritance diagram for ilearnrw.prototype.application.UserGames:



Collaboration diagram for `ilearnrw.prototype.application.UserGames`:



Public Member Functions

- [UserGames](#) (String text, [User](#) user)
- [EConsoleMenuActionResult](#) onSelected ([ConsoleMenu](#) menu)

Package Attributes

- [User](#) mUser

7.38.1 Constructor & Destructor Documentation

7.38.1.1 `ilearnrw.prototype.application.UserGames.UserGames (String text, User user)`

7.38.2 Member Function Documentation

7.38.2.1 **EConsoleMenuActionResult** ilearnrw.prototype.application.UserGames.onSelected (**ConsoleMenu** menu)

Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

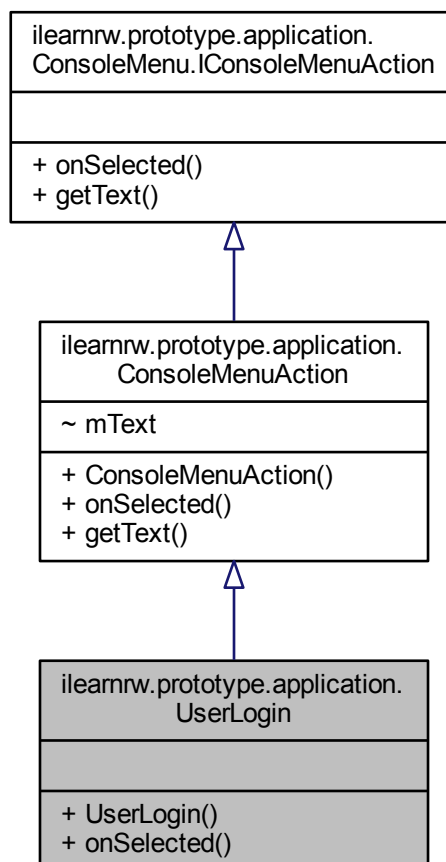
7.38.3 Member Data Documentation

7.38.3.1 **User** ilearnrw.prototype.application.UserGames.mUser [package]

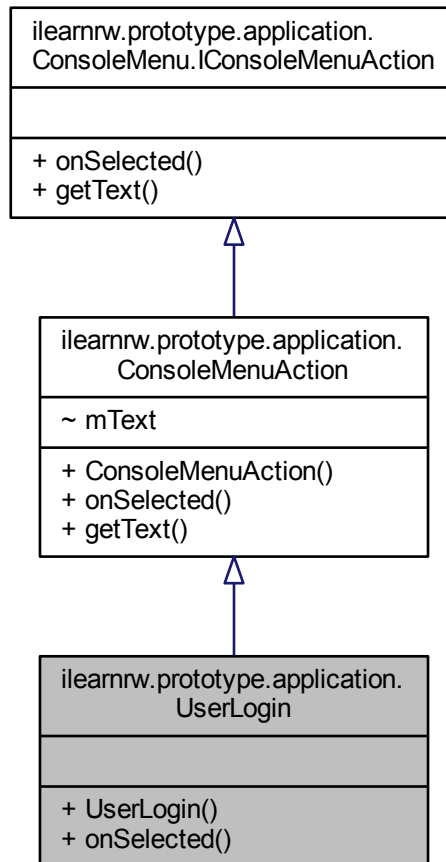
7.39 ilearnrw.prototype.application.UserLogin Class Reference

Main menu of a logged in user.

Inheritance diagram for ilearnrw.prototype.application.UserLogin:



Collaboration diagram for `ilearnrw.prototype.application.UserLogin`:



Public Member Functions

- [UserLogin](#) (String text)
- [EConsoleMenuActionResult onSelected](#) ([ConsoleMenu](#) menu)

Additional Inherited Members

7.39.1 Detailed Description

Main menu of a logged in user.

7.39.2 Constructor & Destructor Documentation

7.39.2.1 `ilearnrw.prototype.application.UserLogin.UserLogin (String text)`

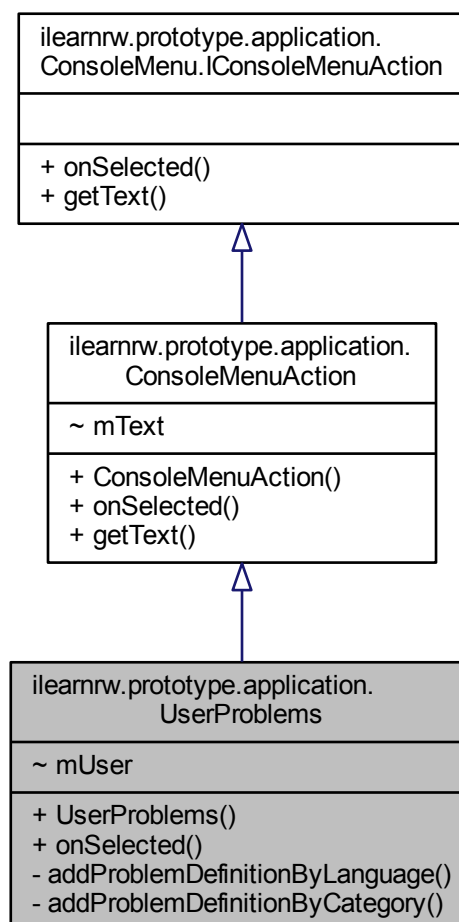
7.39.3 Member Function Documentation

7.39.3.1 **EConsoleMenuActionResult** ilearnrw.prototype.application.UserLogin.onSelected (**ConsoleMenu** *menu*)

Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

7.40 ilearnrw.prototype.application.UserProblems Class Reference

Inheritance diagram for ilearnrw.prototype.application.UserProblems:



7.40.1 Constructor & Destructor Documentation

7.40.1.1 `ilearnrw.prototype.application.UserProblems.UserProblems (String text, User user)`

7.40.2 Member Function Documentation

7.40.2.1 `void ilearnrw.prototype.application.UserProblems.addProblemDefinitionByCategory (ConsoleMenu menu)`
[private]

7.40.2.2 `void ilearnrw.prototype.application.UserProblems.addProblemDefinitionByLanguage (ConsoleMenu menu)`
[private]

7.40.2.3 `EConsoleMenuActionResult ilearnrw.prototype.application.UserProblems.onSelected (ConsoleMenu menu)`

Should display a menu where you can:

- List problem definitions
- Add problem definitions to a profile from the ProblemDefinitionIndex.
- Remove problem definitions

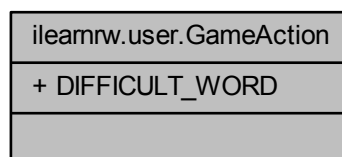
Implements [ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction](#).

7.40.3 Member Data Documentation

7.40.3.1 `User ilearnrw.prototype.application.UserProblems.mUser = null` [package]

7.41 ilearnrw.user.GameAction Enum Reference

Collaboration diagram for ilearnrw.user.GameAction:



Public Attributes

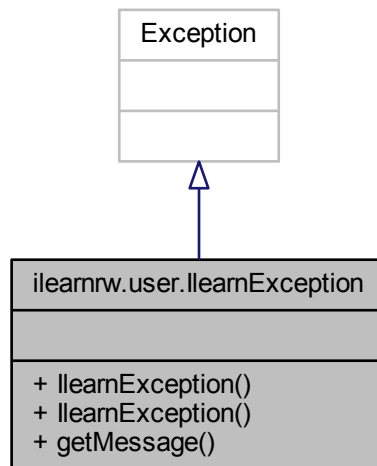
- [DIFFICULT_WORD](#)

7.41.1 Member Data Documentation

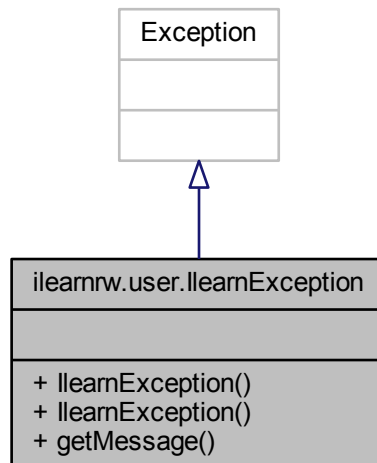
7.41.1.1 `ilearnrw.user.GameAction.DIFFICULT_WORD`

7.42 ilearnrw.user.IlearnException Class Reference

Inheritance diagram for ilearnrw.user.IlearnException:



Collaboration diagram for ilearnrw.user.IlearnException:



Public Member Functions

- [IlearnException](#) (String message)
- [IlearnException](#) (String message, Throwable throwable)
- String [getMessage](#) ()

7.42.1 Constructor & Destructor Documentation

7.42.1.1 `ilearnrw.user.IlearnException.IlearnException (String message)`

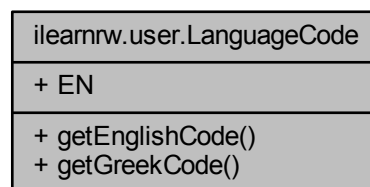
7.42.1.2 `ilearnrw.user.IlearnException.IlearnException (String message, Throwable throwable)`

7.42.2 Member Function Documentation

7.42.2.1 `String ilearnrw.user.IlearnException.getMessage ()`

7.43 ilearnrw.user.LanguageCode Enum Reference

Collaboration diagram for `ilearnrw.user.LanguageCode`:



Static Public Member Functions

- static byte [getEnglishCode](#) ()
- static byte [getGreekCode](#) ()

Public Attributes

- [EN](#)

7.43.1 Member Function Documentation

7.43.1.1 `static byte ilearnrw.user.LanguageCode.getEnglishCode () [static]`

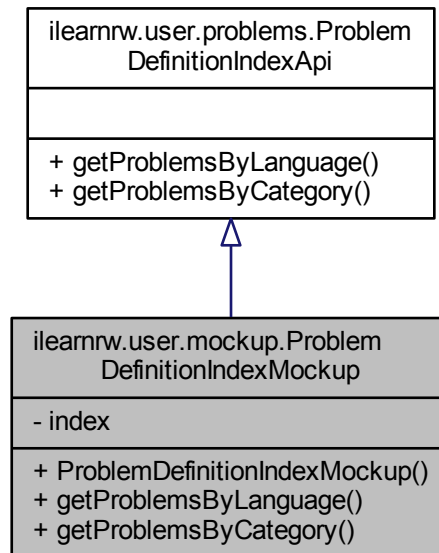
7.43.1.2 `static byte ilearnrw.user.LanguageCode.getGreekCode () [static]`

7.43.2 Member Data Documentation

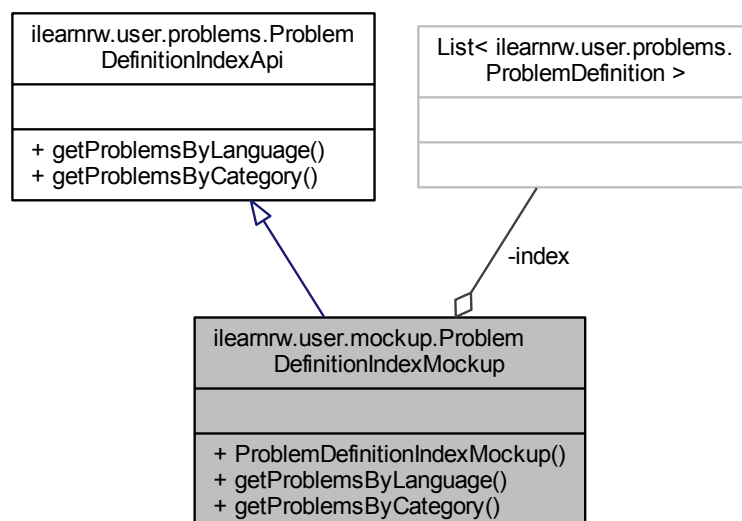
7.43.2.1 `ilearnrw.user.LanguageCode.EN`

7.44 ilearnrw.user.mockup.ProblemDefinitionIndexMockup Class Reference

Inheritance diagram for ilearnrw.user.mockup.ProblemDefinitionIndexMockup:



Collaboration diagram for ilearnrw.user.mockup.ProblemDefinitionIndexMockup:



Public Member Functions

- [ProblemDefinitionIndexMockup](#) ()
- [ArrayList< ProblemDefinition > getProblemsByLanguage](#) ([LanguageCode](#) x)
- [ArrayList< ProblemDefinition > getProblemsByCategory](#) ([Category](#) x)

Private Attributes

- [List< ProblemDefinition > index](#) = new [ArrayList<ProblemDefinition>\(\)](#)

7.44.1 Constructor & Destructor Documentation

7.44.1.1 [ilearnrw.user.mockup.ProblemDefinitionIndexMockup.ProblemDefinitionIndexMockup](#) ()

7.44.2 Member Function Documentation

7.44.2.1 [ArrayList<ProblemDefinition> ilearnrw.user.mockup.ProblemDefinitionIndexMockup.getProblemsByCategory](#) ([Category](#) x)

Implements [ilearnrw.user.problems.ProblemDefinitionIndexApi](#).

7.44.2.2 [ArrayList<ProblemDefinition> ilearnrw.user.mockup.ProblemDefinitionIndexMockup.getProblemsByLanguage](#) ([LanguageCode](#) x)

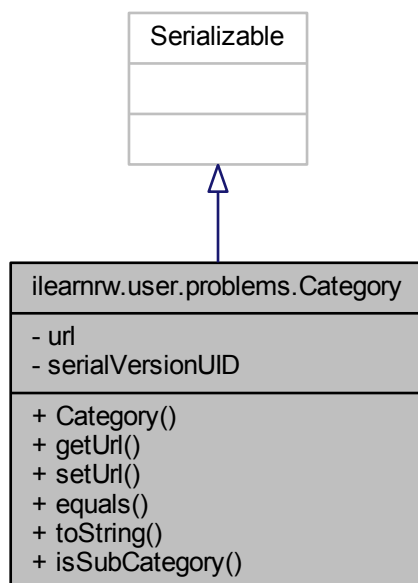
Implements [ilearnrw.user.problems.ProblemDefinitionIndexApi](#).

7.44.3 Member Data Documentation

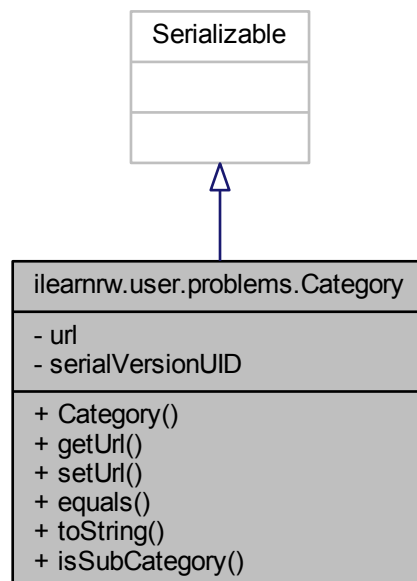
7.44.3.1 [List<ProblemDefinition> ilearnrw.user.mockup.ProblemDefinitionIndexMockup.index](#) = new [ArrayList<ProblemDefinition>\(\)](#) [private]

7.45 ilearnrw.user.problems.Category Class Reference

Inheritance diagram for ilearnrw.user.problems.Category:



Collaboration diagram for ilearnrw.user.problems.Category:



Public Member Functions

- [Category](#) (String *url*)
- String [getUrl](#) ()
- void [setUrl](#) (String *url*)
- boolean [equals](#) (Object *obj*)
- String [toString](#) ()
- boolean [isSubCategory](#) ([Category](#) *c*)

Private Attributes

- String [url](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.45.1 Constructor & Destructor Documentation

7.45.1.1 `ilearnrw.user.problems.Category.Category (String url)`

7.45.2 Member Function Documentation

7.45.2.1 `boolean ilearnrw.user.problems.Category.equals (Object obj)`

7.45.2.2 `String ilearnrw.user.problems.Category.getUrl ()`

7.45.2.3 `boolean ilearnrw.user.problems.Category.isSubCategory (Category c)`

7.45.2.4 `void ilearnrw.user.problems.Category.setUrl (String url)`

7.45.2.5 `String ilearnrw.user.problems.Category.toString ()`

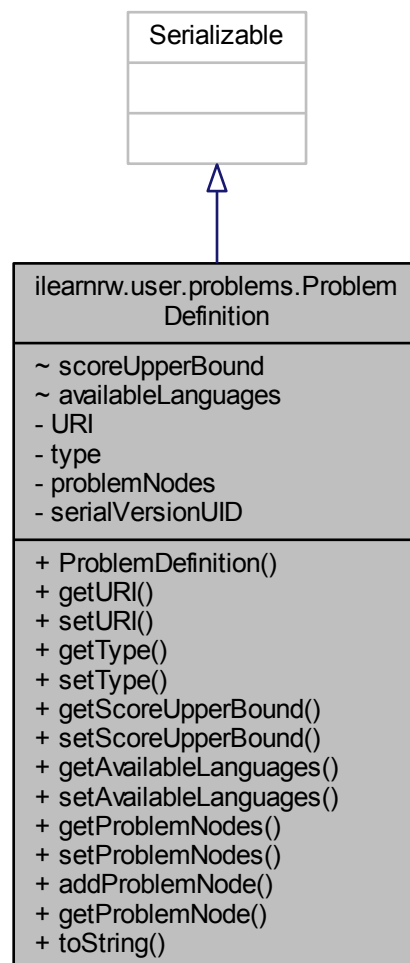
7.45.3 Member Data Documentation

7.45.3.1 `final long ilearnrw.user.problems.Category.serialVersionUID = 1L` `[static],[private]`

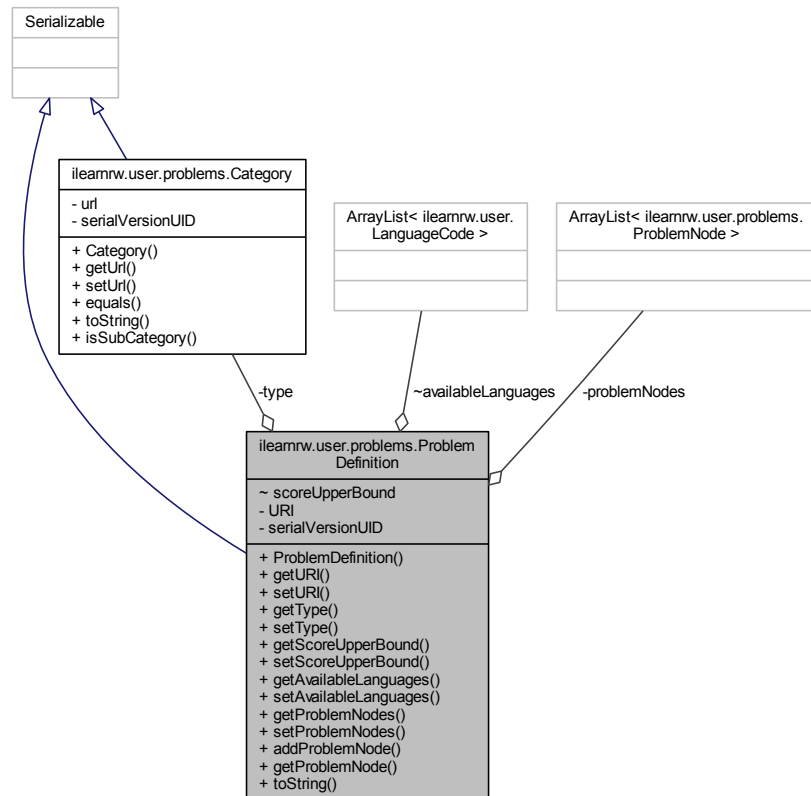
7.45.3.2 `String ilearnrw.user.problems.Category.url` `[private]`

7.46 ilearnrw.user.problems.ProblemDefinition Class Reference

Inheritance diagram for ilearnrw.user.problems.ProblemDefinition:



Collaboration diagram for ilearnrw.user.problems.ProblemDefinition:



Public Member Functions

- [ProblemDefinition](#) (String [URI](#), [Category](#) [type](#), int [scoreUpperBound](#), ArrayList< [LanguageCode](#) > [availableLanguages](#))
- String [getURI](#) ()
- void [setURI](#) (String [uRI](#))
- [Category](#) [getType](#) ()
- void [setType](#) ([Category](#) [type](#))
- int [getScoreUpperBound](#) ()
- void [setScoreUpperBound](#) (int [scoreUpperBound](#))
- ArrayList< [LanguageCode](#) > [getAvailableLanguages](#) ()
- void [setAvailableLanguages](#) (ArrayList< [LanguageCode](#) > [availableLanguages](#))
- ArrayList< [ProblemNode](#) > [getProblemNodes](#) ()
- void [setProblemNodes](#) (ArrayList< [ProblemNode](#) > [problemNodes](#))
- void [addProblemNode](#) ([ProblemNode](#) [node](#))
- [ProblemNode](#) [getProblemNode](#) (int [idx](#)) throws [IlearnException](#)
- String [toString](#) ()

Package Attributes

- int [scoreUpperBound](#)
- ArrayList< [LanguageCode](#) > [availableLanguages](#)

Private Attributes

- String [URI](#)
- [Category](#) type
- ArrayList< [ProblemNode](#) > *problemNodes*

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.46.1 Constructor & Destructor Documentation

- 7.46.1.1 `ilearnrw.user.problems.ProblemDefinition.ProblemDefinition (String URI, Category type, int scoreUpperBound, ArrayList< LanguageCode > availableLanguages)`

7.46.2 Member Function Documentation

- 7.46.2.1 `void ilearnrw.user.problems.ProblemDefinition.addProblemNode (ProblemNode node)`
- 7.46.2.2 `ArrayList<LanguageCode> ilearnrw.user.problems.ProblemDefinition.getAvailableLanguages ()`
- 7.46.2.3 `ProblemNode ilearnrw.user.problems.ProblemDefinition.getProblemNode (int idx)` throws `ILearnException`
- 7.46.2.4 `ArrayList<ProblemNode> ilearnrw.user.problems.ProblemDefinition.getProblemNodes ()`
- 7.46.2.5 `int ilearnrw.user.problems.ProblemDefinition.getScoreUpperBound ()`
- 7.46.2.6 `Category ilearnrw.user.problems.ProblemDefinition.getType ()`
- 7.46.2.7 `String ilearnrw.user.problems.ProblemDefinition.getURI ()`
- 7.46.2.8 `void ilearnrw.user.problems.ProblemDefinition.setAvailableLanguages (ArrayList< LanguageCode > availableLanguages)`
- 7.46.2.9 `void ilearnrw.user.problems.ProblemDefinition.setProblemNodes (ArrayList< ProblemNode > problemNodes)`
- 7.46.2.10 `void ilearnrw.user.problems.ProblemDefinition.setScoreUpperBound (int scoreUpperBound)`
- 7.46.2.11 `void ilearnrw.user.problems.ProblemDefinition.setType (Category type)`
- 7.46.2.12 `void ilearnrw.user.problems.ProblemDefinition.setURI (String uRI)`
- 7.46.2.13 `String ilearnrw.user.problems.ProblemDefinition.toString ()`

7.46.3 Member Data Documentation

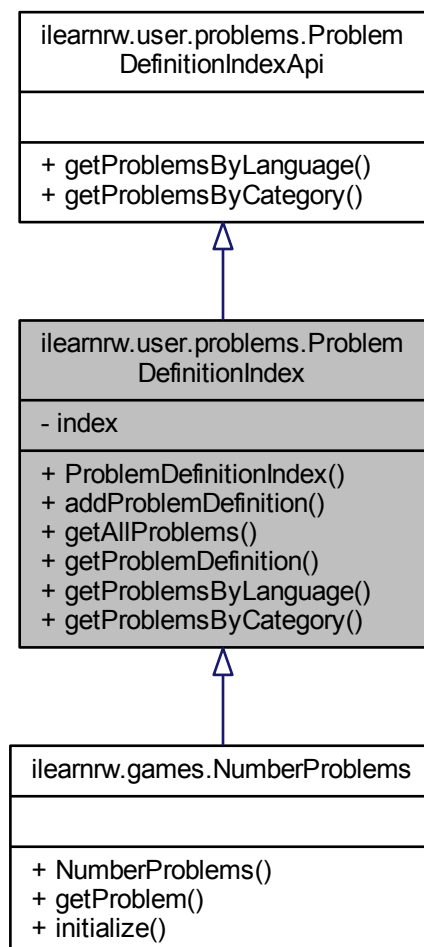
- 7.46.3.1 `ArrayList<LanguageCode> ilearnrw.user.problems.ProblemDefinition.availableLanguages` [package]
- 7.46.3.2 `ArrayList<ProblemNode> ilearnrw.user.problems.ProblemDefinition.problemNodes` [private]
- 7.46.3.3 `int ilearnrw.user.problems.ProblemDefinition.scoreUpperBound` [package]
- 7.46.3.4 `final long ilearnrw.user.problems.ProblemDefinition.serialVersionUID = 1L` [static], [private]

7.46.3.5 Category ilearnrw.user.problems.ProblemDefinition.type [private]

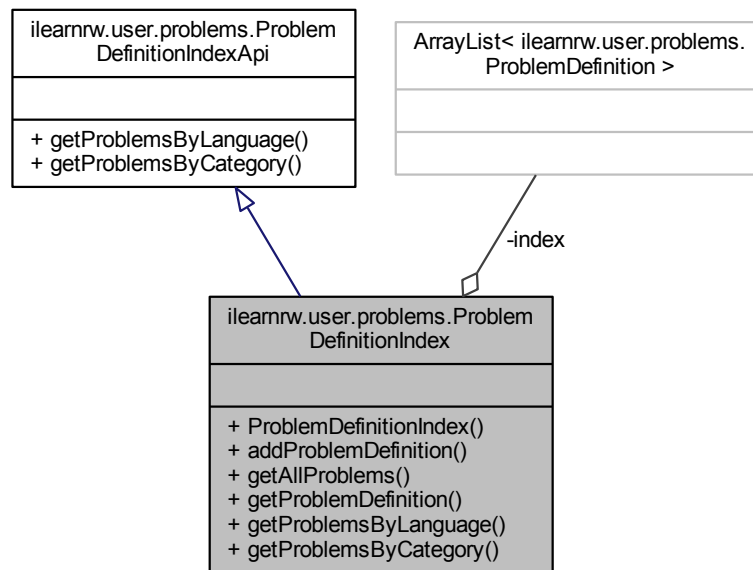
7.46.3.6 String ilearnrw.user.problems.ProblemDefinition.URI [private]

7.47 ilearnrw.user.problems.ProblemDefinitionIndex Class Reference

Inheritance diagram for ilearnrw.user.problems.ProblemDefinitionIndex:



Collaboration diagram for `ilearnrw.user.problems.ProblemDefinitionIndex`:



Public Member Functions

- [ProblemDefinitionIndex](#) ()
- void [addProblemDefinition](#) ([ProblemDefinition](#) prob)
- [ArrayList< ProblemDefinition > getAllProblems](#) ()
- [ProblemDefinition getProblemDefinition](#) (int idx) throws [IlearnException](#)
- [ArrayList< ProblemDefinition > getProblemsByLanguage](#) ([LanguageCode](#) x)
- [ArrayList< ProblemDefinition > getProblemsByCategory](#) ([Category](#) x)

Private Attributes

- [ArrayList< ProblemDefinition > index](#)

7.47.1 Constructor & Destructor Documentation

7.47.1.1 `ilearnrw.user.problems.ProblemDefinitionIndex.ProblemDefinitionIndex ()`

7.47.2 Member Function Documentation

7.47.2.1 void `ilearnrw.user.problems.ProblemDefinitionIndex.addProblemDefinition (ProblemDefinition prob)`

7.47.2.2 `ArrayList<ProblemDefinition> ilearnrw.user.problems.ProblemDefinitionIndex.getAllProblems ()`

7.47.2.3 `ProblemDefinition ilearnrw.user.problems.ProblemDefinitionIndex.getProblemDefinition (int idx)` throws [IlearnException](#)

7.47.2.4 `ArrayList<ProblemDefinition> ilearnrw.user.problems.ProblemDefinitionIndex.getProblemsByCategory (Category x)`

Implements [ilearnrw.user.problems.ProblemDefinitionIndexApi](#).

7.47.2.5 `ArrayList<ProblemDefinition> ilearnrw.user.problems.ProblemDefinitionIndex.getProblemsByLanguage (LanguageCode x)`

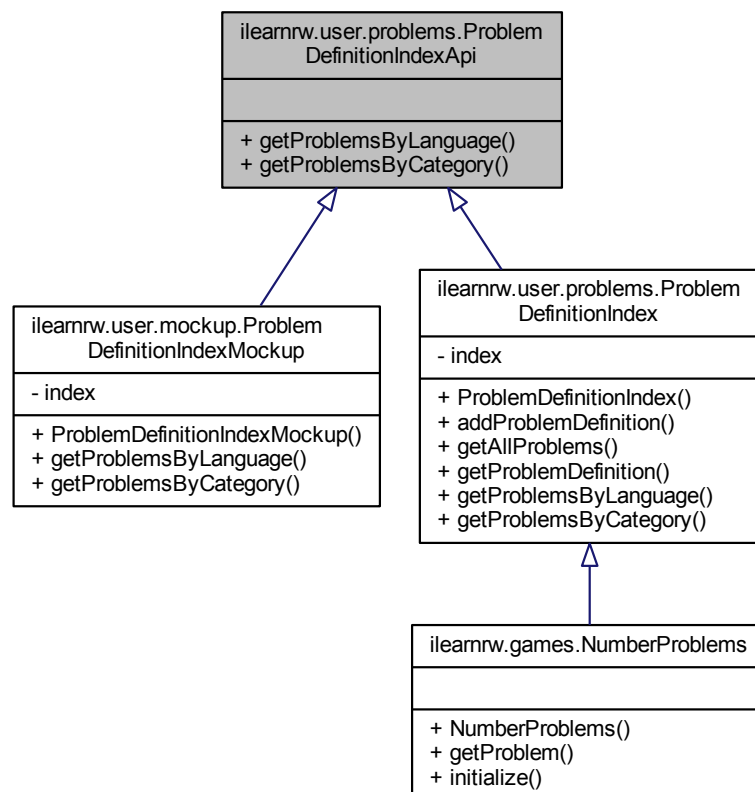
Implements [ilearnrw.user.problems.ProblemDefinitionIndexApi](#).

7.47.3 Member Data Documentation

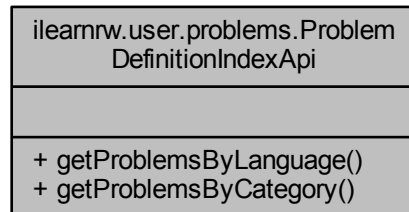
7.47.3.1 `ArrayList<ProblemDefinition> ilearnrw.user.problems.ProblemDefinitionIndex.index` `[private]`

7.48 ilearnrw.user.problems.ProblemDefinitionIndexApi Interface Reference

Inheritance diagram for `ilearnrw.user.problems.ProblemDefinitionIndexApi`:



Collaboration diagram for `ilearnrw.user.problems.ProblemDefinitionIndexApi`:



Public Member Functions

- `ArrayList< ProblemDefinition > getProblemsByLanguage (LanguageCode x)`
- `ArrayList< ProblemDefinition > getProblemsByCategory (Category x)`

7.48.1 Member Function Documentation

7.48.1.1 `ArrayList<ProblemDefinition> ilearnrw.user.problems.ProblemDefinitionIndexApi.getProblemsByCategory (Category x)`

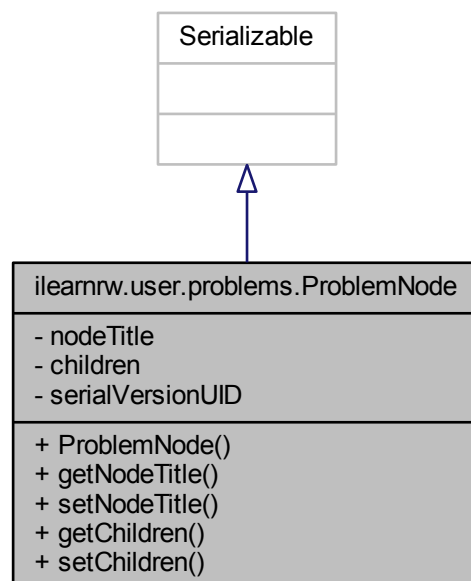
Implemented in [ilearnrw.user.mockup.ProblemDefinitionIndexMockup](#), and [ilearnrw.user.problems.ProblemDefinitionIndex](#).

7.48.1.2 `ArrayList<ProblemDefinition> ilearnrw.user.problems.ProblemDefinitionIndexApi.getProblemsByLanguage (LanguageCode x)`

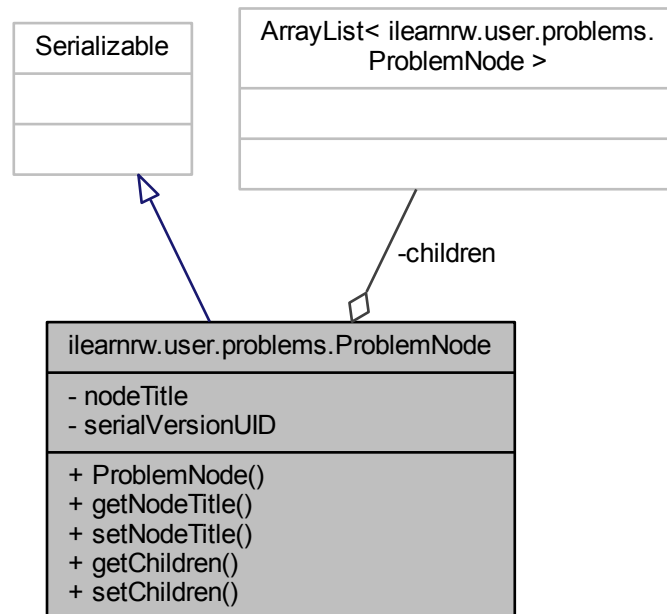
Implemented in [ilearnrw.user.mockup.ProblemDefinitionIndexMockup](#), and [ilearnrw.user.problems.ProblemDefinitionIndex](#).

7.49 ilearnrw.user.problems.ProblemNode Class Reference

Inheritance diagram for ilearnrw.user.problems.ProblemNode:



Collaboration diagram for `ilearnrw.user.problems.ProblemNode`:



Public Member Functions

- `ProblemNode` (String `nodeTitle`, ArrayList< `ProblemNode` > `children`)
- String `getNodeTitle` ()
- void `setNodeTitle` (String `nodeTitle`)
- ArrayList< `ProblemNode` > `getChildren` ()
- void `setChildren` (ArrayList< `ProblemNode` > `children`)

Private Attributes

- String `nodeTitle`
- ArrayList< `ProblemNode` > `children`

Static Private Attributes

- static final long `serialVersionUID` = 1L

7.49.1 Constructor & Destructor Documentation

7.49.1.1 `ilearnrw.user.problems.ProblemNode.ProblemNode` (String `nodeTitle`, ArrayList< `ProblemNode` > `children`)

Parameters

<i>nodeTitle</i>	the title of the problem
<i>children</i>	a list of subproblems

7.49.2 Member Function Documentation

7.49.2.1 `ArrayList<ProblemNode> ilearnrw.user.problems.ProblemNode.getChildren ()`

Returns

the arraylist with the subproblems

7.49.2.2 `String ilearnrw.user.problems.ProblemNode.getNodeTitle ()`

Returns

the nodeTitle

7.49.2.3 `void ilearnrw.user.problems.ProblemNode.setChildren (ArrayList< ProblemNode > children)`

Parameters

<i>the</i>	children to set
------------	-----------------

7.49.2.4 `void ilearnrw.user.problems.ProblemNode.setNodeTitle (String nodeTitle)`

Parameters

<i>nodeTitle</i>	the nodeTitle to set
------------------	----------------------

7.49.3 Member Data Documentation

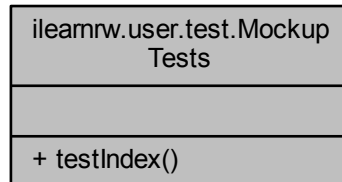
7.49.3.1 `ArrayList<ProblemNode> ilearnrw.user.problems.ProblemNode.children` [private]

7.49.3.2 `String ilearnrw.user.problems.ProblemNode.nodeTitle` [private]

7.49.3.3 `final long ilearnrw.user.problems.ProblemNode.serialVersionUID = 1L` [static], [private]

7.50 ilearnrw.user.test.MockupTests Class Reference

Collaboration diagram for ilearnrw.user.test.MockupTests:



Public Member Functions

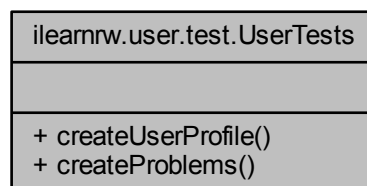
- void [testIndex](#) ()

7.50.1 Member Function Documentation

7.50.1.1 void `ilearnrw.user.test.MockupTests.testIndex` ()

7.51 ilearnrw.user.test.UserTests Class Reference

Collaboration diagram for ilearnrw.user.test.UserTests:



Public Member Functions

- void [createUserProfile](#) ()
- void [createProblems](#) ()

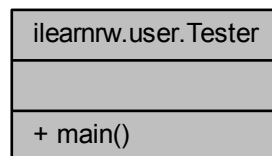
7.51.1 Member Function Documentation

7.51.1.1 void `ilearnrw.user.test.UserTests.createProblems` ()

7.51.1.2 void ilearnrw.user.test.UserTests.createUserProfile ()

7.52 ilearnrw.user.Tester Class Reference

Collaboration diagram for ilearnrw.user.Tester:



Static Public Member Functions

- static void `main` (String[] args)

7.52.1 Member Function Documentation

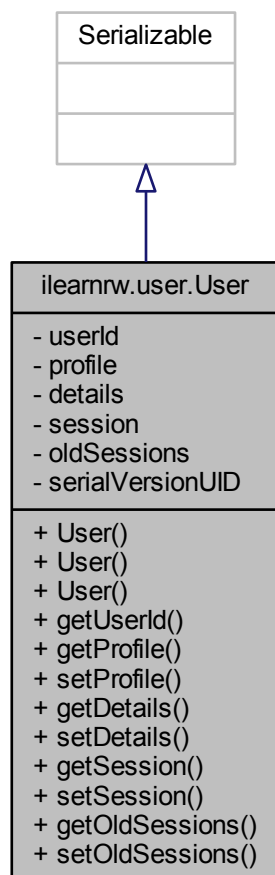
7.52.1.1 static void ilearnrw.user.Tester.main (String[] args) [static]

Parameters

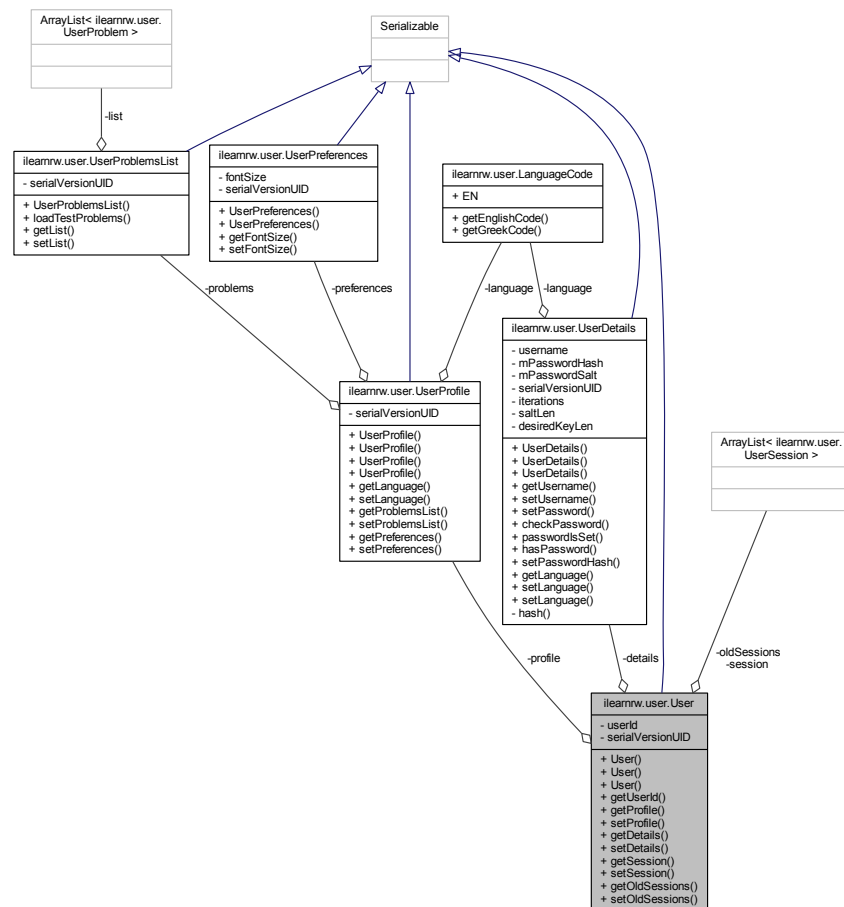
<i>args</i>	
-------------	--

7.53 ilearnrw.user.User Class Reference

Inheritance diagram for ilearnrw.user.User:



Collaboration diagram for ilearnrw.user.User:



Public Member Functions

- [User](#) (int [userId](#), [UserProfile](#) [profile](#), [UserDetails](#) [details](#), ArrayList< [UserSession](#) > [oldSessions](#))
- [User](#) ([LanguageCode](#) [lc](#))
- [User](#) (int [userId](#))
- int [getUserId](#) ()
- [UserProfile](#) [getProfile](#) ()
- void [setProfile](#) ([UserProfile](#) [profile](#))
- [UserDetails](#) [getDetails](#) ()
- void [setDetails](#) ([UserDetails](#) [details](#))
- ArrayList< [UserSession](#) > [getSession](#) ()
- void [setSession](#) (ArrayList< [UserSession](#) > [session](#))
- ArrayList< [UserSession](#) > [getOldSessions](#) ()
- void [setOldSessions](#) (ArrayList< [UserSession](#) > [oldSessions](#))

Private Attributes

- int [userId](#)
- [UserProfile](#) [profile](#)
- [UserDetails](#) [details](#)
- ArrayList< [UserSession](#) > [session](#)
- ArrayList< [UserSession](#) > [oldSessions](#)

Static Private Attributes

- static final long `serialVersionUID` = 1L

7.53.1 Constructor & Destructor Documentation

7.53.1.1 `ilearnrw.user.User.User (int userId, UserProfile profile, UserDetails details, ArrayList< UserSession > oldSessions)`

7.53.1.2 `ilearnrw.user.User.User (LanguageCode lc)`

7.53.1.3 `ilearnrw.user.User.User (int userId)`

7.53.2 Member Function Documentation

7.53.2.1 **UserDetails** `ilearnrw.user.User.getDetails ()`

7.53.2.2 **ArrayList<UserSession>** `ilearnrw.user.User.getOldSessions ()`

7.53.2.3 **UserProfile** `ilearnrw.user.User.getProfile ()`

7.53.2.4 **ArrayList<UserSession>** `ilearnrw.user.User.getSession ()`

7.53.2.5 **int** `ilearnrw.user.User.getUserId ()`

7.53.2.6 **void** `ilearnrw.user.User.setDetails (UserDetails details)`

7.53.2.7 **void** `ilearnrw.user.User.setOldSessions (ArrayList< UserSession > oldSessions)`

7.53.2.8 **void** `ilearnrw.user.User.setProfile (UserProfile profile)`

7.53.2.9 **void** `ilearnrw.user.User.setSession (ArrayList< UserSession > session)`

7.53.3 Member Data Documentation

7.53.3.1 **UserDetails** `ilearnrw.user.User.details` [private]

7.53.3.2 **ArrayList<UserSession>** `ilearnrw.user.User.oldSessions` [private]

7.53.3.3 **UserProfile** `ilearnrw.user.User.profile` [private]

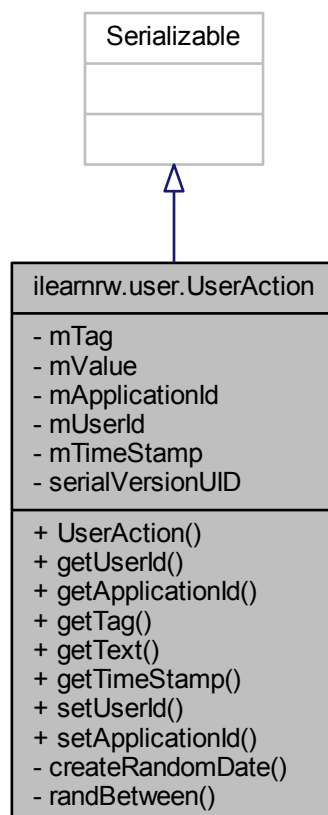
7.53.3.4 **final long** `ilearnrw.user.User.serialVersionUID = 1L` [static], [private]

7.53.3.5 **ArrayList<UserSession>** `ilearnrw.user.User.session` [private]

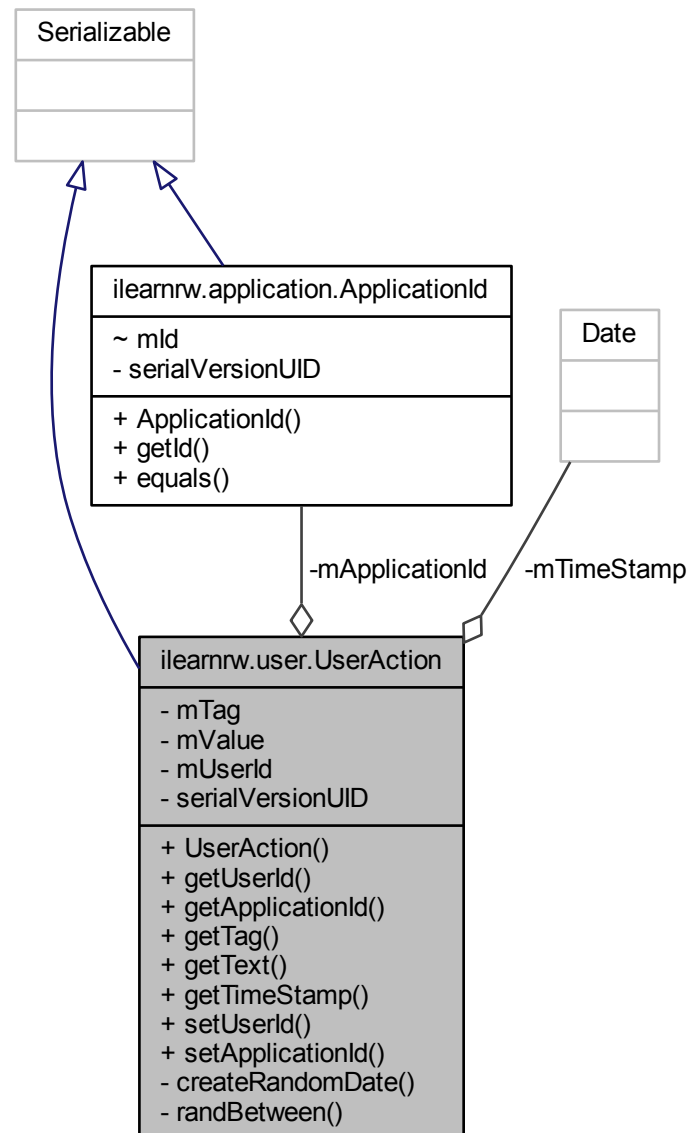
7.53.3.6 **int** `ilearnrw.user.User.userId` [private]

7.54 ilearnrw.user.UserAction Class Reference

Inheritance diagram for ilearnrw.user.UserAction:



Collaboration diagram for `ilearnrw.user.UserAction`:



Public Member Functions

- `UserAction` (String tag, String value, [ApplicationId](#) applicationId, int userId)
- int `getUserId` ()
- [ApplicationId](#) `getApplicationId` ()
- String `getTag` ()
- String `getText` ()
- Date `getTimeStamp` ()
- void `setUserId` (int userId)
- void `setApplicationId` ([ApplicationId](#) applicationId)

Private Member Functions

- Calendar [createRandomDate](#) (int startYear, int endYear)

Static Private Member Functions

- static int [randBetween](#) (int start, int end)

Private Attributes

- String [mTag](#)
- String [mValue](#)
- [ApplicationId](#) [mApplicationId](#)
- int [mUserId](#)
- Date [mTimeStamp](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 2L

7.54.1 Detailed Description

Todo Add some documentation about this class.

Add get/setters

7.54.2 Constructor & Destructor Documentation

7.54.2.1 `ilearnrw.user.UserAction.UserAction (String tag, String value, ApplicationId applicationId, int userId)`

Default constructor used by applications when logging a user action.

Parameters

<i>tag</i>	
<i>value</i>	
<i>applicationId</i>	
<i>userId</i>	

7.54.3 Member Function Documentation

7.54.3.1 `Calendar ilearnrw.user.UserAction.createRandomDate (int startYear, int endYear) [private]`

Todo : remove these after testing UserActions with UserActionFilter enough

7.54.3.2 `ApplicationId ilearnrw.user.UserAction.getApplicationId ()`

7.54.3.3 `String ilearnrw.user.UserAction.getTag ()`

7.54.3.4 `String ilearnrw.user.UserAction.getText ()`

7.54.3.5 `Date ilearnrw.user.UserAction.getTimeStamp ()`

7.54.3.6 `int ilearnrw.user.UserAction.getUserId ()`

7.54.3.7 `static int ilearnrw.user.UserAction.randBetween (int start, int end)` `[static], [private]`

7.54.3.8 `void ilearnrw.user.UserAction.setApplicationId (ApplicationId applicationId)`

7.54.3.9 `void ilearnrw.user.UserAction.setUserId (int userId)`

7.54.4 Member Data Documentation

7.54.4.1 `ApplicationId ilearnrw.user.UserAction.mApplicationId` `[private]`

7.54.4.2 `String ilearnrw.user.UserAction.mTag` `[private]`

7.54.4.3 `Date ilearnrw.user.UserAction.mTimeStamp` `[private]`

Not sure about the datatype Date. Does it contain the time as well? Might have to change.

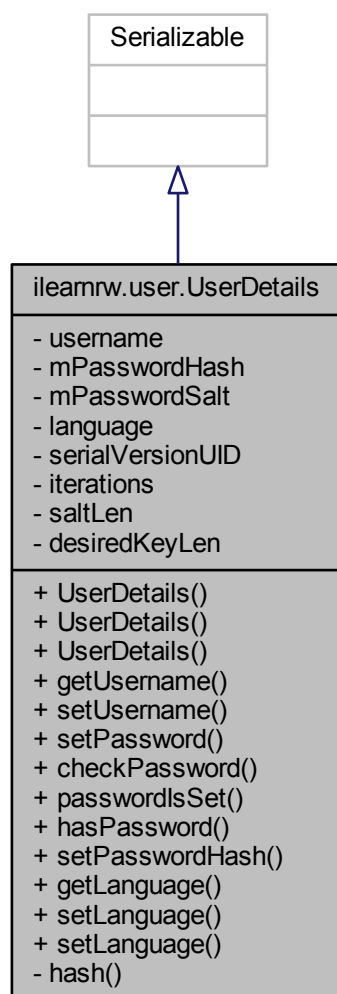
7.54.4.4 `int ilearnrw.user.UserAction.mUserId` `[private]`

7.54.4.5 `String ilearnrw.user.UserAction.mValue` `[private]`

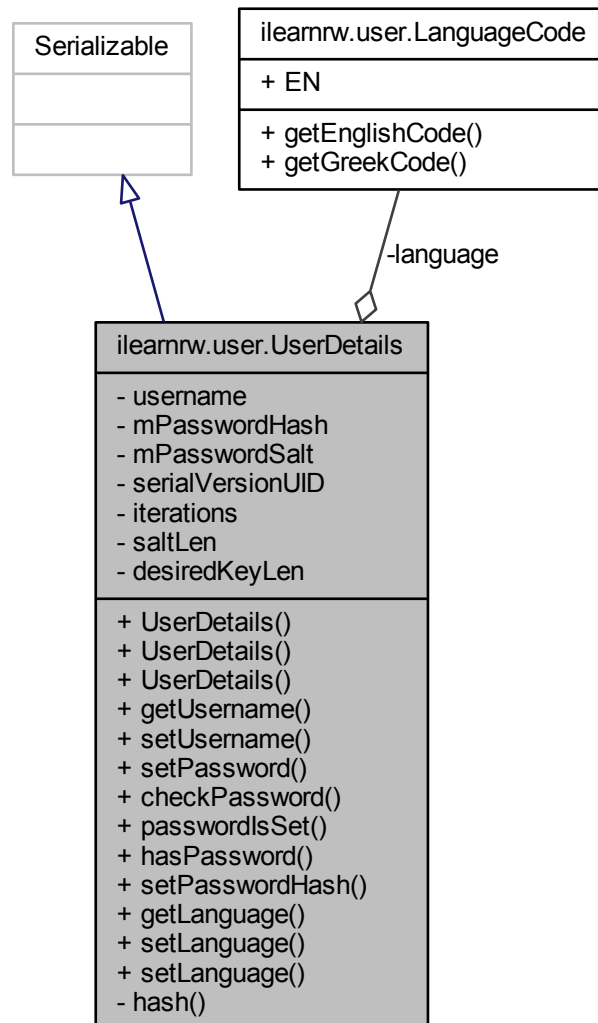
7.54.4.6 `final long ilearnrw.user.UserAction.serialVersionUID = 2L` `[static], [private]`

7.55 ilearnrw.user.UserDetails Class Reference

Inheritance diagram for ilearnrw.user.UserDetails:



Collaboration diagram for ilearnrw.user.UserDetails:



Public Member Functions

- [UserDetails](#) (String [username](#), int password, [LanguageCode](#) [language](#))
- [UserDetails](#) ([LanguageCode](#) [language](#))
- [UserDetails](#) (int [userId](#))
- String [getUsername](#) ()
- void [setUsername](#) (String [username](#))
- final void [setPassword](#) (String plaintextPassword)
- final boolean [checkPassword](#) (String plaintextPassword)
- boolean [passwordIsSet](#) ()
- boolean [hasPassword](#) ()
- final void [setPasswordHash](#) (byte[] [hash](#), byte[] salt)
- [LanguageCode](#) [getLanguage](#) ()
- void [setLanguage](#) ([LanguageCode](#) [language](#))
- void [setLanguage](#) (byte languageCode)

Static Private Member Functions

- static byte[] [hash](#) (String plaintextPassword, byte[] salt) throws NoSuchAlgorithmException, InvalidKeySpecException

Private Attributes

- String [username](#)
- byte[] [mPasswordHash](#)
Stores hash(plainTextPassword + mPasswordSalt)
- byte[] [mPasswordSalt](#)
Stores the salt used to hash the password.
- [LanguageCode](#) [language](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 1L
- static final int [iterations](#) = 10*1024
- static final int [saltLen](#) = 32
- static final int [desiredKeyLen](#) = 256

7.55.1 Constructor & Destructor Documentation

7.55.1.1 `ilearnrw.user.UserDetails.UserDetails (String username, int password, LanguageCode language)`

7.55.1.2 `ilearnrw.user.UserDetails.UserDetails (LanguageCode language)`

7.55.1.3 `ilearnrw.user.UserDetails.UserDetails (int userId)`

7.55.2 Member Function Documentation

7.55.2.1 `final boolean ilearnrw.user.UserDetails.checkPassword (String plaintextPassword)`

7.55.2.2 `LanguageCode ilearnrw.user.UserDetails.getLanguage ()`

7.55.2.3 `String ilearnrw.user.UserDetails.getUsername ()`

7.55.2.4 `static byte [] ilearnrw.user.UserDetails.hash (String plaintextPassword, byte[] salt) throws
NoSuchAlgorithmException, InvalidKeySpecException [static],[private]`

7.55.2.5 `boolean ilearnrw.user.UserDetails.hasPassword ()`

7.55.2.6 `boolean ilearnrw.user.UserDetails.passwordIsSet ()`

7.55.2.7 `void ilearnrw.user.UserDetails.setLanguage (LanguageCode language)`

Parameters

<i>the</i>	language code as a enum item
------------	------------------------------

7.55.2.8 `void ilearnrw.user.UserDetails.setLanguage (byte languageCode)`

Parameters

<i>the</i>	language code as a enum item
------------	------------------------------

7.55.2.9 `final void ilearnrw.user.UserDetails.setPassword (String plaintextPassword)`

7.55.2.10 `final void ilearnrw.user.UserDetails.setPasswordHash (byte[] hash, byte[] salt)`

7.55.2.11 `void ilearnrw.user.UserDetails.setUsername (String username)`

7.55.3 Member Data Documentation

7.55.3.1 `final int ilearnrw.user.UserDetails.desiredKeyLen = 256` `[static], [private]`

7.55.3.2 `final int ilearnrw.user.UserDetails.iterations = 10*1024` `[static], [private]`

7.55.3.3 `LanguageCode ilearnrw.user.UserDetails.language` `[private]`

7.55.3.4 `byte [] ilearnrw.user.UserDetails.mPasswordHash` `[private]`

Stores hash(plainTextPassword + mPasswordSalt)

Note

In the future we might want to change this to store the value in a base64 encoded String;

7.55.3.5 `byte [] ilearnrw.user.UserDetails.mPasswordSalt` `[private]`

Stores the salt used to hash the password.

Note

In the future we might want to change this to store the value in a base64 encoded String;

7.55.3.6 `final int ilearnrw.user.UserDetails.saltLen = 32` `[static], [private]`

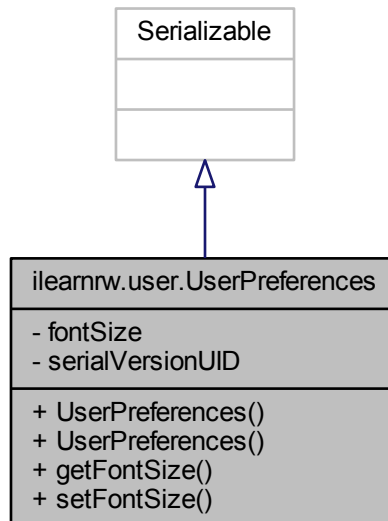
7.55.3.7 `final long ilearnrw.user.UserDetails.serialVersionUID = 1L` `[static], [private]`

7.55.3.8 `String ilearnrw.user.UserDetails.username` `[private]`

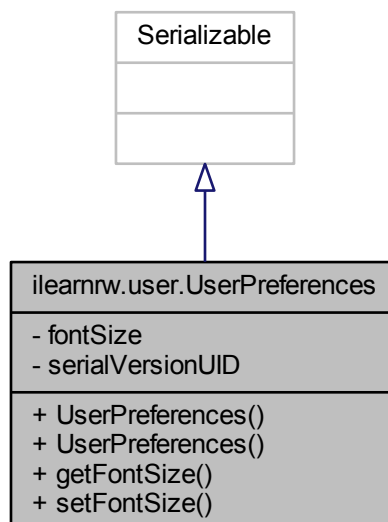
Todo Something to think about is whether this should be case-sensitive and if unicode should be supported. Unicode can create some headaches, there was a blog post about this recently (of which i only read the headline) see: <http://labs.spotify.com/2013/06/18/creative-usernames/>

7.56 ilearnrw.user.UserPreferences Class Reference

Inheritance diagram for ilearnrw.user.UserPreferences:



Collaboration diagram for ilearnrw.user.UserPreferences:



Public Member Functions

- [UserPreferences](#) ()
- [UserPreferences](#) (int [fontSize](#))
- int [getFontSize](#) ()
- void [setFontSize](#) (int [fontSize](#))

Private Attributes

- int [fontSize](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.56.1 Constructor & Destructor Documentation

7.56.1.1 `ilearnrw.user.UserPreferences.UserPreferences ()`

7.56.1.2 `ilearnrw.user.UserPreferences.UserPreferences (int fontSize)`

7.56.2 Member Function Documentation

7.56.2.1 `int ilearnrw.user.UserPreferences.getFontSize ()`

7.56.2.2 `void ilearnrw.user.UserPreferences.setFontSize (int fontSize)`

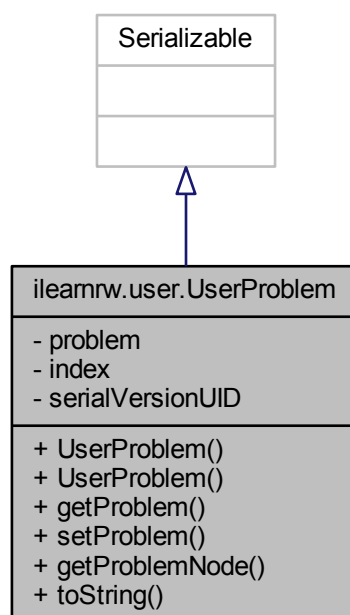
7.56.3 Member Data Documentation

7.56.3.1 `int ilearnrw.user.UserPreferences.fontSize` `[private]`

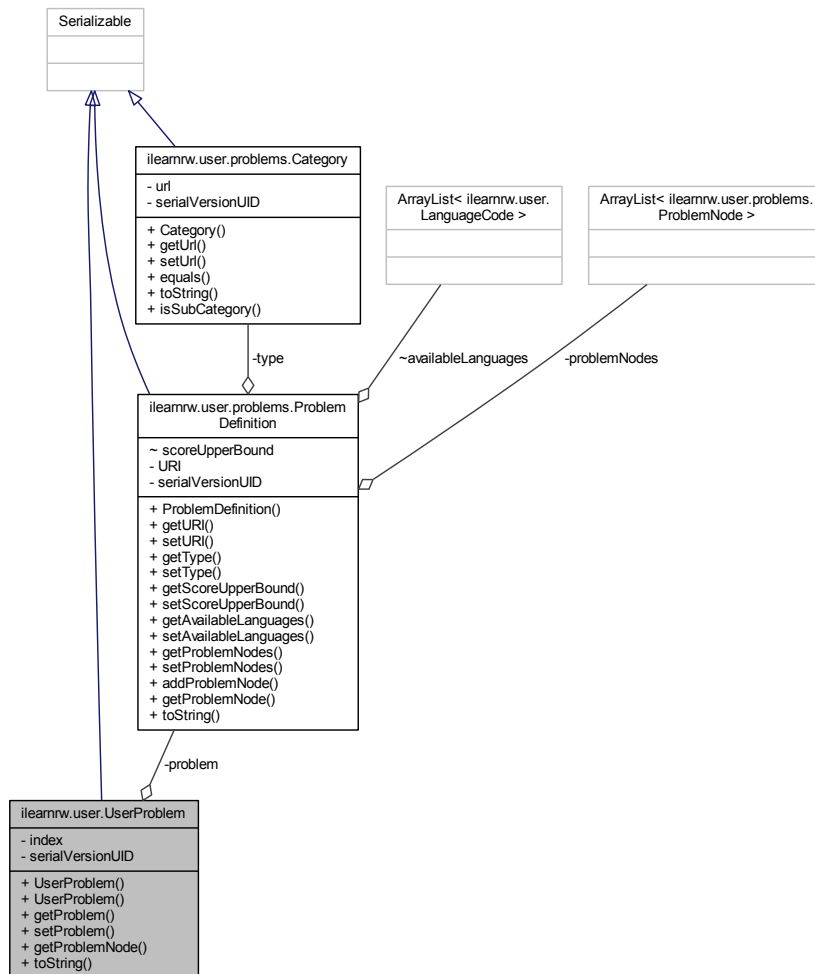
7.56.3.2 `final long ilearnrw.user.UserPreferences.serialVersionUID = 1L` `[static], [private]`

7.57 ilearnrw.user.UserProblem Class Reference

Inheritance diagram for ilearnrw.user.UserProblem:



Collaboration diagram for `ilearnrw.user.UserProblem`:



Public Member Functions

- `UserProblem` (`ProblemDefinition problem`, `int index`)
- `UserProblem` (`ProblemDefinition problem`)
- `ProblemDefinition` `getProblem` ()
- `void` `setProblem` (`ProblemDefinition problem`)
- `ProblemNode` `getProblemNode` () throws `ILearnException`
- `String` `toString` ()

Private Attributes

- `ProblemDefinition` `problem`
- `int` `index`

Static Private Attributes

- `static final long` `serialVersionUID` = 1L

7.57.1 Constructor & Destructor Documentation

7.57.1.1 `ilearnrw.user.UserProblem.UserProblem (ProblemDefinition problem, int index)`

7.57.1.2 `ilearnrw.user.UserProblem.UserProblem (ProblemDefinition problem)`

7.57.2 Member Function Documentation

7.57.2.1 **ProblemDefinition** `ilearnrw.user.UserProblem.getProblem ()`

Returns

the ProblemDefintion of the user's problem

7.57.2.2 **ProblemNode** `ilearnrw.user.UserProblem.getProblemNode ()` throws `ILearnException`

7.57.2.3 `void ilearnrw.user.UserProblem.setProblem (ProblemDefinition problem)`

7.57.2.4 `String ilearnrw.user.UserProblem.toString ()`

7.57.3 Member Data Documentation

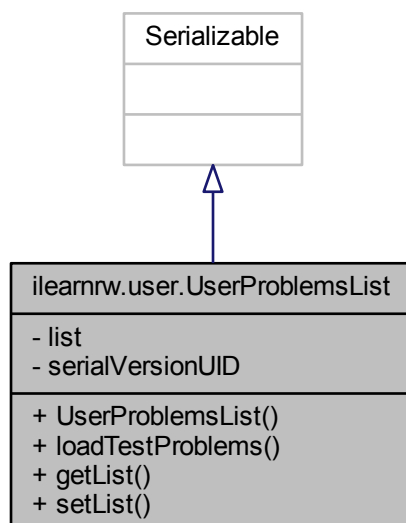
7.57.3.1 `int ilearnrw.user.UserProblem.index` `[private]`

7.57.3.2 **ProblemDefinition** `ilearnrw.user.UserProblem.problem` `[private]`

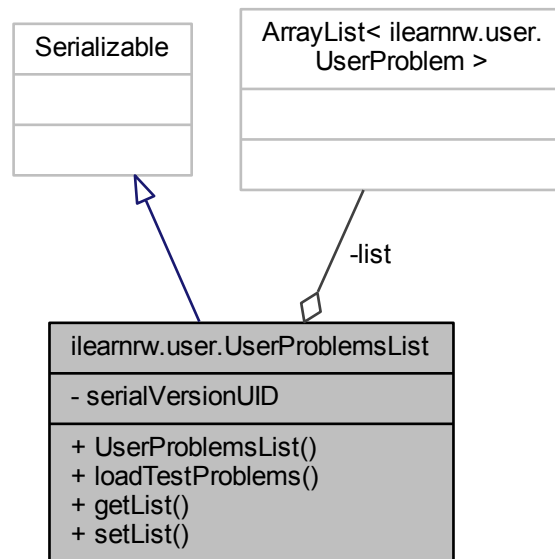
7.57.3.3 `final long ilearnrw.user.UserProblem.serialVersionUID = 1L` `[static],[private]`

7.58 ilearnrw.user.UserProblemsList Class Reference

Inheritance diagram for `ilearnrw.user.UserProblemsList`:



Collaboration diagram for `ilearnrw.user.UserProblemsList`:



Public Member Functions

- [UserProblemsList](#) ()
- void [loadTestProblems](#) ()
- [ArrayList< UserProblem > getList](#) ()
- void [setList](#) ([ArrayList< UserProblem > list](#))

Private Attributes

- [ArrayList< UserProblem > list](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.58.1 Constructor & Destructor Documentation

7.58.1.1 `ilearnrw.user.UserProblemsList.UserProblemsList ()`

7.58.2 Member Function Documentation

7.58.2.1 `ArrayList<UserProblem> ilearnrw.user.UserProblemsList.getList ()`

7.58.2.2 `void ilearnrw.user.UserProblemsList.loadTestProblems ()`

7.58.2.3 `void ilearnrw.user.UserProblemsList.setList (ArrayList< UserProblem > list)`

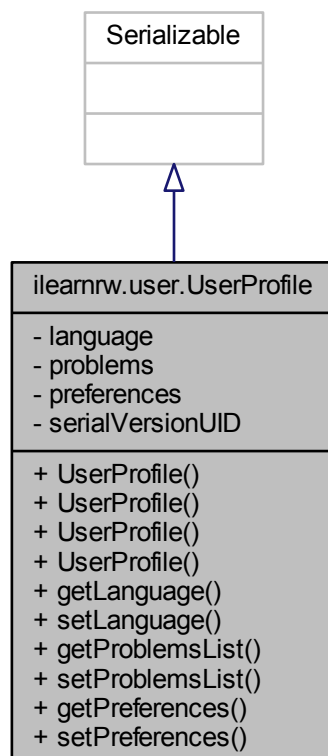
7.58.3 Member Data Documentation

7.58.3.1 `ArrayList<UserProblem> ilearnrw.user.UserProblemsList.list` `[private]`

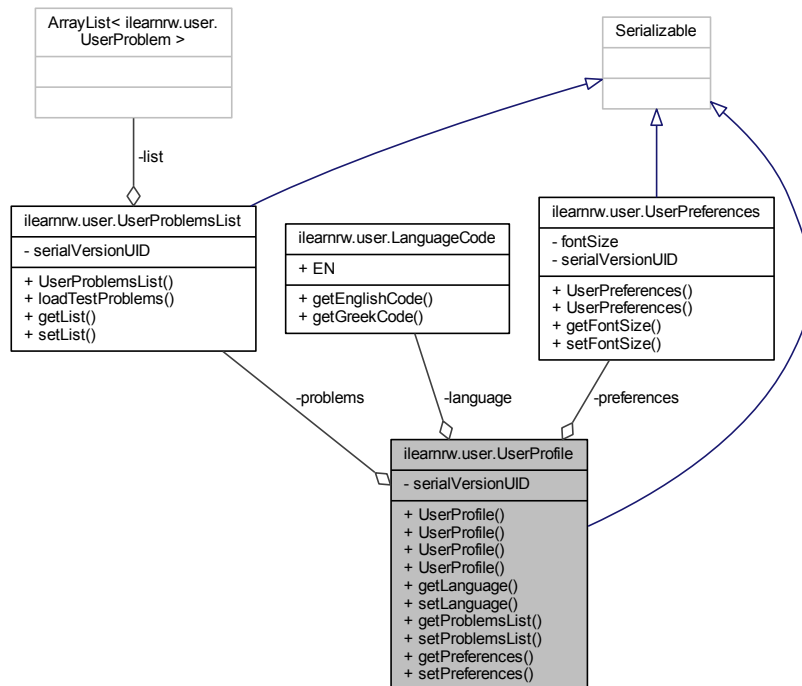
7.58.3.2 `final long ilearnrw.user.UserProblemsList.serialVersionUID = 1L` `[static], [private]`

7.59 ilearnrw.user.UserProfile Class Reference

Inheritance diagram for ilearnrw.user.UserProfile:



Collaboration diagram for ilearnrw.user.UserProfile:



Public Member Functions

- `UserProfile (LanguageCode language, UserProblemsList problems, UserPreferences preferences)`
- `UserProfile (LanguageCode language)`
- `UserProfile ()`
- `UserProfile (int userId)`
- `LanguageCode getLanguage ()`
- `void setLanguage (LanguageCode language)`
- `UserProblemsList getProblemsList ()`
- `void setProblemsList (UserProblemsList problems)`
- `UserPreferences getPreferences ()`
- `void setPreferences (UserPreferences preferences)`

Private Attributes

- `LanguageCode language`
- `UserProblemsList problems`
- `UserPreferences preferences`

Static Private Attributes

- static final long `serialVersionUID` = 1L

7.59.1 Constructor & Destructor Documentation

7.59.1.1 `ilearnrw.user.UserProfile.UserProfile (LanguageCode language, UserProblemsList problems, UserPreferences preferences)`

7.59.1.2 `ilearnrw.user.UserProfile.UserProfile (LanguageCode language)`

7.59.1.3 `ilearnrw.user.UserProfile.UserProfile ()`

7.59.1.4 `ilearnrw.user.UserProfile.UserProfile (int userId)`

7.59.2 Member Function Documentation

7.59.2.1 `LanguageCode ilearnrw.user.UserProfile.getLanguage ()`

7.59.2.2 `UserPreferences ilearnrw.user.UserProfile.getPreferences ()`

7.59.2.3 `UserProblemsList ilearnrw.user.UserProfile.getProblemsList ()`

7.59.2.4 `void ilearnrw.user.UserProfile.setLanguage (LanguageCode language)`

7.59.2.5 `void ilearnrw.user.UserProfile.setPreferences (UserPreferences preferences)`

7.59.2.6 `void ilearnrw.user.UserProfile.setProblemsList (UserProblemsList problems)`

7.59.3 Member Data Documentation

7.59.3.1 `LanguageCode ilearnrw.user.UserProfile.language [private]`

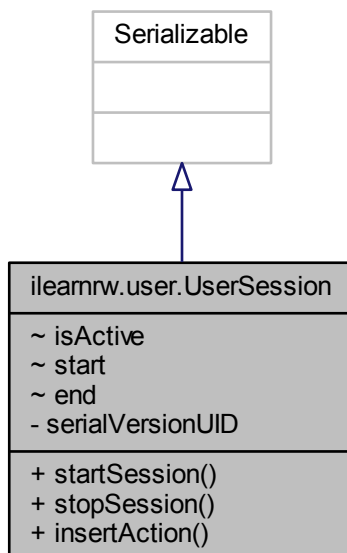
7.59.3.2 `UserPreferences ilearnrw.user.UserProfile.preferences [private]`

7.59.3.3 `UserProblemsList ilearnrw.user.UserProfile.problems [private]`

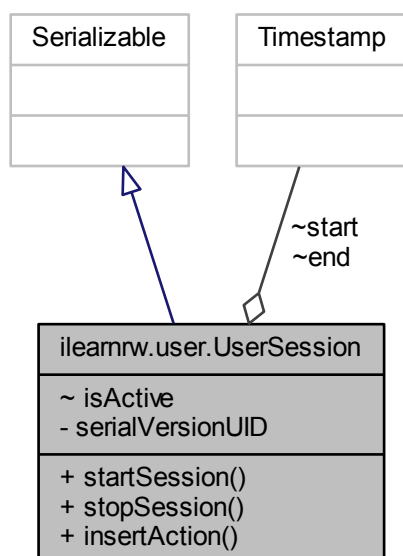
7.59.3.4 `final long ilearnrw.user.UserProfile.serialVersionUID = 1L [static], [private]`

7.60 ilearnrw.user.UserSession Class Reference

Inheritance diagram for ilearnrw.user.UserSession:



Collaboration diagram for ilearnrw.user.UserSession:



Public Member Functions

- void [startSession](#) ()
- void [stopSession](#) ()
- void [insertAction](#) ([UserAction](#) x) throws [IlearnException](#)

Package Attributes

- boolean [isActive](#)
- Timestamp [start](#)
- Timestamp [end](#)

Static Private Attributes

- static final long [serialVersionUID](#) = 1L

7.60.1 Detailed Description

Note

At the moment there will be issues if a user has 2 concurrent sessions. The reason for this is that the User-Actions are not tied directly to a session. To fix: Add deviceid to session and actions.

7.60.2 Member Function Documentation

7.60.2.1 void [ilearnrw.user.UserSession.insertAction](#) ([UserAction](#) x) throws [IlearnException](#)

7.60.2.2 void [ilearnrw.user.UserSession.startSession](#) ()

sets the timestamp and turns the boolean variable 'isActive' to true!

7.60.2.3 void [ilearnrw.user.UserSession.stopSession](#) ()

sets the timestamp and turns the boolean variable 'isActive' to true!

7.60.3 Member Data Documentation

7.60.3.1 Timestamp [ilearnrw.user.UserSession.end](#) [package]

7.60.3.2 boolean [ilearnrw.user.UserSession.isActive](#) [package]

7.60.3.3 final long [ilearnrw.user.UserSession.serialVersionUID](#) = 1L [static],[private]

7.60.3.4 Timestamp [ilearnrw.user.UserSession.start](#) [package]

Index

APPID

ilearnrw::games::NumbersToText, [56](#)

addProblemDefinition

ilearnrw::user::problems::ProblemDefinitionIndex, [106](#)

addProblemDefinitionByCategory

ilearnrw::prototype::application::UserProblems, [95](#)

addProblemDefinitionByLanguage

ilearnrw::prototype::application::UserProblems, [95](#)

addProblemNode

ilearnrw::user::problems::ProblemDefinition, [104](#)

addUserAction

ilearnrw::prototype::application::UserActions, [86](#)

addUserProblem

ilearnrw::datalogger::DataLogger, [21](#)

ilearnrw::datalogger::IProfileAccessUpdater, [31](#)

ApplicationId

ilearnrw::application::ApplicationId, [18](#)

authenticateAdmin

ilearnrw::datalogger::IUserAdministration, [38](#)

ilearnrw::datalogger::UserStore, [46](#)

AuthenticationException

ilearnrw::datalogger::IUserAdministration::AuthenticationException, [41](#)

availableLanguages

ilearnrw::user::problems::ProblemDefinition, [104](#)

Category

ilearnrw::user::problems::Category, [101](#)

changeUsername

ilearnrw::datalogger::IUserAdministration, [38](#)

ilearnrw::datalogger::UserStore, [47](#)

checkAdminAuth

ilearnrw::datalogger::UserStore, [47](#)

checkPassword

ilearnrw::user::UserDetails, [123](#)

children

ilearnrw::user::problems::ProblemNode, [111](#)

CompositeBigProcessor

ilearnrw::games::NumbersToTextHelper::CompositeBigProcessor, [62](#)

ConsoleMenu

ilearnrw::prototype::application::ConsoleMenu, [75](#)

ConsoleMenuAction

ilearnrw::prototype::application::ConsoleMenuAction, [79](#)

create

ilearnrw::datalogger::UserStore, [47](#)

createProblems

ilearnrw::user::test::UserTests, [112](#)

createRandomDate

ilearnrw::user::UserAction, [119](#)

createUser

ilearnrw::datalogger::IUserAdministration, [39](#)

ilearnrw::datalogger::UserStore, [47](#)

createUserProfile

ilearnrw::user::test::UserTests, [112](#)

DIFFICULT_WORD

ilearnrw::user::GameAction, [95](#)

DatabaseManager

ilearnrw::prototype::application::DatabaseManager, [81](#)

deepCopy

ilearnrw::datalogger::UserStore, [47](#)

delete

ilearnrw::datalogger::UserStore, [47](#)

deleteUser

ilearnrw::datalogger::IUserAdministration, [39](#)

ilearnrw::datalogger::UserStore, [47](#)

desiredKeyLen

ilearnrw::user::UserDetails, [124](#)

details

ilearnrw::user::User, [116](#)

discardPendingChanges

ilearnrw::datalogger::DataLogger, [21](#)

ilearnrw::datalogger::IProfileAccessUpdater, [31](#)

doModalMenu

ilearnrw::prototype::application::ConsoleMenu, [76](#)

DolphinTodo_20130619, 9

EN

ilearnrw::user::LanguageCode, [97](#)

EXPONENT

ilearnrw::games::NumbersToTextHelper::Hundred-Processor, [67](#)

end

ilearnrw::user::UserSession, [135](#)

equals

ilearnrw::application::ApplicationId, [18](#)

ilearnrw::user::problems::Category, [101](#)

exponent

ilearnrw::games::NumbersToTextHelper::CompositeBigProcessor, [62](#)

ilearnrw::games::NumbersToTextHelper::ScaleUnit, [68](#)

FAILED_TAG

ilearnrw::games::NumbersToText, [56](#)

fontSize

- ilearnrw::user::UserPreferences, 126
- GameAddTwoNumbers
 - ilearnrw::games::GameAddTwoNumbers, 51
- getAction
 - ilearnrw::prototype::application::UserActions, 86
- getActions
 - ilearnrw::datalogger::DataLogger, 21
 - ilearnrw::datalogger::IDataLogger, 26
 - ilearnrw::prototype::application::UserActions, 86
- getAllProblems
 - ilearnrw::user::problems::ProblemDefinitionIndex, 106
- getAllUsers
 - ilearnrw::datalogger::IUserAdministration, 39
 - ilearnrw::datalogger::UserStore, 48
- getApplicationId
 - ilearnrw::datalogger::UserActionFilter, 43
 - ilearnrw::user::UserAction, 119
- getAvailableLanguages
 - ilearnrw::user::problems::ProblemDefinition, 104
- getChildren
 - ilearnrw::user::problems::ProblemNode, 111
- getDataLogger
 - ilearnrw::prototype::application::Program, 83
- getDetails
 - ilearnrw::user::User, 116
- getDigits
 - ilearnrw::games::NumbersToTextHelper::Abstract-Processor, 59
- getEndTime
 - ilearnrw::datalogger::UserActionFilter, 43
- getEnglishCode
 - ilearnrw::user::LanguageCode, 97
- getExponent
 - ilearnrw::games::NumbersToTextHelper::Scale-Unit, 68
- getFontSize
 - ilearnrw::user::UserPreferences, 126
- getGreekCode
 - ilearnrw::user::LanguageCode, 97
- getHighProcessor
 - ilearnrw::games::NumbersToTextHelper::Composite-BigProcessor, 62
- getId
 - ilearnrw::application::ApplicationId, 18
- getLanguage
 - ilearnrw::user::UserDetails, 123
 - ilearnrw::user::UserProfile, 133
- getList
 - ilearnrw::user::UserProblemsList, 130
- getLoginProvider
 - ilearnrw::prototype::application::Program, 83
- getLowProcessor
 - ilearnrw::games::NumbersToTextHelper::Composite-BigProcessor, 62
- getMessage
 - ilearnrw::datalogger::IProfileAccessUpdater::PendingChangesAvailable, 34
- ilearnrw::datalogger::IProfileAccessUpdater::PermissionException, 36
- ilearnrw::datalogger::IUserAdministration::Authentication-Exception, 41
- ilearnrw::user::IlearnException, 97
- getName
 - ilearnrw::games::NumbersToTextHelper::Abstract-Processor, 59
 - ilearnrw::games::NumbersToTextHelper::Composite-BigProcessor, 62
 - ilearnrw::games::NumbersToTextHelper::Default-Processor, 64
 - ilearnrw::games::NumbersToTextHelper::Hundred-Processor, 67
 - ilearnrw::games::NumbersToTextHelper::Scale, 67
 - ilearnrw::games::NumbersToTextHelper::Scale-Unit, 68
 - ilearnrw::games::NumbersToTextHelper::Tens-Processor, 71
 - ilearnrw::games::NumbersToTextHelper::Unit-Processor, 72
- getNodeTitle
 - ilearnrw::user::problems::ProblemNode, 111
- getOldSessions
 - ilearnrw::user::User, 116
- getPartDivider
 - ilearnrw::games::NumbersToTextHelper::Composite-BigProcessor, 62
- getPreferences
 - ilearnrw::user::UserProfile, 133
- getProblem
 - ilearnrw::games::NumberProblems, 53
 - ilearnrw::user::UserProblem, 129
- getProblemDefinition
 - ilearnrw::user::problems::ProblemDefinitionIndex, 106
- getProblemNode
 - ilearnrw::user::problems::ProblemDefinition, 104
 - ilearnrw::user::UserProblem, 129
- getProblemNodes
 - ilearnrw::user::problems::ProblemDefinition, 104
- getProblemsByCategory
 - ilearnrw::user::mockup::ProblemDefinitionIndex-Mockup, 99
 - ilearnrw::user::problems::ProblemDefinitionIndex, 106
 - ilearnrw::user::problems::ProblemDefinitionIndex-API, 108
- getProblemsByLanguage
 - ilearnrw::user::mockup::ProblemDefinitionIndex-Mockup, 99
 - ilearnrw::user::problems::ProblemDefinitionIndex, 107
 - ilearnrw::user::problems::ProblemDefinitionIndex-API, 108
- getProblemsList
 - ilearnrw::user::UserProfile, 133
- getProfile

- ilearnrw::user::User, 116
- getProfileAccessUpdater
 - ilearnrw::prototype::application::Program, 83
- getScoreUpperBound
 - ilearnrw::user::problems::ProblemDefinition, 104
- getSession
 - ilearnrw::user::User, 116
- getStartTime
 - ilearnrw::datalogger::UserActionFilter, 43
- getStringArg
 - ilearnrw::prototype::application::Program, 84
- getTag
 - ilearnrw::user::UserAction, 119
- getTags
 - ilearnrw::datalogger::UserActionFilter, 43
- getText
 - ilearnrw::prototype::application::ConsoleMenu::IConsoleMenuAction, 77
 - ilearnrw::prototype::application::ConsoleMenuAction, 79
 - ilearnrw::user::UserAction, 119
- getTimeStamp
 - ilearnrw::user::UserAction, 119
- getToken
 - ilearnrw::games::NumbersToTextHelper::CompositeBigProcessor, 62
- getType
 - ilearnrw::user::problems::ProblemDefinition, 104
- getURI
 - ilearnrw::user::problems::ProblemDefinition, 104
- getUrl
 - ilearnrw::user::problems::Category, 101
- getUser
 - ilearnrw::datalogger::DataLogger, 21
 - ilearnrw::datalogger::ILoginProvider, 28
 - ilearnrw::datalogger::UserStore, 48
- getUserAdministration
 - ilearnrw::datalogger::DataLogger, 22
 - ilearnrw::datalogger::ILoginProvider, 28
 - ilearnrw::datalogger::UserStore, 48
- getUserId
 - ilearnrw::datalogger::UserActionFilter, 43
 - ilearnrw::user::User, 116
 - ilearnrw::user::UserAction, 119
- getUserIds
 - ilearnrw::datalogger::UserStore, 48
- getUsername
 - ilearnrw::user::UserDetails, 123
- hasPassword
 - ilearnrw::user::UserDetails, 123
- hasPendingChanges
 - ilearnrw::datalogger::DataLogger, 22
 - ilearnrw::datalogger::IProfileAccessUpdater, 31
- hash
 - ilearnrw::user::UserDetails, 123
- hundredProcessor
 - ilearnrw::games::NumbersToTextHelper::CompositeBigProcessor, 62
- IlearnException
 - ilearnrw::user::IlearnException, 97
- ilearnrw, 11
 - ilearnrw.application, 11
 - ilearnrw.application.ApplicationId, 17
 - ilearnrw.datalogger, 11
 - ilearnrw.datalogger.DataLogger, 19
 - ilearnrw.datalogger.IDataLogger, 24
 - ilearnrw.datalogger.ILoginProvider, 27
 - ilearnrw.datalogger.IProfileAccessUpdater, 28
 - ilearnrw.datalogger.IProfileAccessUpdater.PendingChangesAvailable, 33
 - ilearnrw.datalogger.IProfileAccessUpdater.PermissionException, 34
 - ilearnrw.datalogger.IUserAdministration, 36
 - ilearnrw.datalogger.IUserAdministration.AuthenticationException, 39
 - ilearnrw.datalogger.UserActionFilter, 42
 - ilearnrw.datalogger.UserStore, 43
 - ilearnrw.games, 12
 - ilearnrw.games.GameAddTwoNumbers, 50
 - ilearnrw.games.NumberProblems, 52
 - ilearnrw.games.NumbersToText, 54
 - ilearnrw.games.NumbersToTextHelper, 56
 - ilearnrw.games.NumbersToTextHelper.AbstractProcessor, 58
 - ilearnrw.games.NumbersToTextHelper.CompositeBigProcessor, 60
 - ilearnrw.games.NumbersToTextHelper.DefaultProcessor, 63
 - ilearnrw.games.NumbersToTextHelper.HundredProcessor, 65
 - ilearnrw.games.NumbersToTextHelper.Scale, 67
 - ilearnrw.games.NumbersToTextHelper.ScaleUnit, 68
 - ilearnrw.games.NumbersToTextHelper.TensProcessor, 69
 - ilearnrw.games.NumbersToTextHelper.UnitProcessor, 71
 - ilearnrw.games.UsersProvider, 73
 - ilearnrw.online, 12
 - ilearnrw.online.resourcebank, 12
 - ilearnrw.online.resourcebank.IResource, 73
 - ilearnrw.online.resourcebank.IResourceBank, 74
 - ilearnrw.online.resourcebank.IResourceCollection, 74
 - ilearnrw.online.resourcebank.ITeachingStrategy, 74
 - ilearnrw.prototype, 15
 - ilearnrw.prototype.application, 15
 - ilearnrw.prototype.application.ConsoleMenu, 75
 - ilearnrw.prototype.application.ConsoleMenu.EConsoleMenuActionResult, 76
 - ilearnrw.prototype.application.ConsoleMenu.IConsoleMenuAction, 77
 - ilearnrw.prototype.application.ConsoleMenuAction, 78
 - ilearnrw.prototype.application.DatabaseManager, 80
 - ilearnrw.prototype.application.DatabaseManager.ISelectUser, 82
 - ilearnrw.prototype.application.Program, 83
 - ilearnrw.prototype.application.UserActions, 85

- ilearnrw.prototype.application.UserActionsLog, 87
- ilearnrw.prototype.application.UserGames, 89
- ilearnrw.prototype.application.UserLogin, 91
- ilearnrw.prototype.application.UserProblems, 93
- ilearnrw.user, 16
- ilearnrw.user.GameAction, 95
- ilearnrw.user.IlearnException, 96
- ilearnrw.user.LanguageCode, 97
- ilearnrw.user.mockup, 16
- ilearnrw.user.mockup.ProblemDefinitionIndexMockup, 98
- ilearnrw.user.problems, 16
- ilearnrw.user.problems.Category, 100
- ilearnrw.user.problems.ProblemDefinition, 102
- ilearnrw.user.problems.ProblemDefinitionIndex, 105
- ilearnrw.user.problems.ProblemDefinitionIndexApi, 107
- ilearnrw.user.problems.ProblemNode, 109
- ilearnrw.user.test, 16
- ilearnrw.user.test.MockupTests, 112
- ilearnrw.user.test.UserTests, 112
- ilearnrw.user.Tester, 113
- ilearnrw.user.User, 114
- ilearnrw.user.UserAction, 117
- ilearnrw.user.UserDetails, 121
- ilearnrw.user.UserPreferences, 125
- ilearnrw.user.UserProblem, 127
- ilearnrw.user.UserProblemsList, 129
- ilearnrw.user.UserProfile, 131
- ilearnrw.user.UserSession, 134
- ilearnrw::application::ApplicationId
 - ApplicationId, 18
 - equals, 18
 - getId, 18
 - mId, 19
 - serialVersionUID, 19
- ilearnrw::datalogger::DataLogger
 - addUserProblem, 21
 - discardPendingChanges, 21
 - getActions, 21
 - getUser, 21
 - getUserAdministration, 22
 - hasPendingChanges, 22
 - loadUserActions, 22
 - loadUserStore, 22
 - logAction, 22
 - mMarker, 24
 - mUser, 24
 - mUserActions, 24
 - mUserIsDirty, 24
 - mUserStore, 24
 - moveToOldSessions, 22
 - registerUserSession, 23
 - removeUserProblem, 23
 - setCurrentUser, 23
 - setDirty, 23
 - setPreference, 23
 - setUserDetail, 23
 - setUserLanguage, 23
 - writePendingChanges, 23
- ilearnrw::datalogger::IDataLogger
 - getActions, 26
 - logAction, 26
- ilearnrw::datalogger::ILoginProvider
 - getUser, 28
 - getUserAdministration, 28
- ilearnrw::datalogger::IProfileAccessUpdater
 - addUserProblem, 31
 - discardPendingChanges, 31
 - hasPendingChanges, 31
 - moveToOldSessions, 31
 - registerUserSession, 31
 - removeUserProblem, 31
 - setCurrentUser, 32
 - setPreference, 32
 - setUserDetail, 32
 - setUserLanguage, 32
 - writePendingChanges, 32
- ilearnrw::datalogger::IProfileAccessUpdater::Pending-ChangesAvailable
 - getMessage, 34
 - PendingChangesAvailable, 34
 - serialVersionUID, 34
- ilearnrw::datalogger::IProfileAccessUpdater::Permission-Exception
 - getMessage, 36
 - PermissionException, 36
 - serialVersionUID, 36
- ilearnrw::datalogger::IUserAdministration
 - authenticateAdmin, 38
 - changeUsername, 38
 - createUser, 39
 - deleteUser, 39
 - getAllUsers, 39
 - isAuthenticated, 39
 - updatePassword, 39
- ilearnrw::datalogger::IUserAdministration::Authentication-Exception
 - AuthenticationException, 41
 - getMessage, 41
 - serialVersionUID, 41
- ilearnrw::datalogger::UserActionFilter
 - getApplicationId, 43
 - getEndTime, 43
 - getStartTime, 43
 - getTags, 43
 - getUserId, 43
 - mApplicationId, 43
 - mTags, 43
 - mTimeEnd, 43
 - mTimeStart, 43
 - mUserId, 43
 - UserActionFilter, 43
- ilearnrw::datalogger::UserStore
 - authenticateAdmin, 46
 - changeUsername, 47
 - checkAdminAuth, 47

- create, 47
- createUser, 47
- deepCopy, 47
- delete, 47
- deleteUser, 47
- getAllUsers, 48
- getUser, 48
- getUserAdministration, 48
- getUserIds, 48
- isAuthenticated, 48
- load, 48
- mFilePath, 49
- mHasAdminAuth, 49
- mLoadedUsers, 49
- mLogger, 49
- read, 48
- read_noCopy, 48
- save, 48
- update, 48
- updatePassword, 49
- UserStore, 46
- ilearnrw::games::GameAddTwoNumbers
 - GameAddTwoNumbers, 51
 - mApplicationId, 51
 - mLogger, 51
 - mUser, 51
 - mUserActions, 51
 - mUserId, 51
 - mWelcomeMessage, 51
 - printOutActions, 51
 - update, 51
- ilearnrw::games::NumberProblems
 - getProblem, 53
 - initialize, 53
 - NumberProblems, 53
- ilearnrw::games::NumbersToText
 - APPID, 56
 - FAILED_TAG, 56
 - mDataLogger, 56
 - mUser, 56
 - NumbersToText, 56
 - OK_TAG, 56
 - onSelected, 56
 - printOutActions, 56
- ilearnrw::games::NumbersToTextHelper
 - main, 57
 - processor, 57
 - SCALE, 57
 - SCALE_UNITS, 57
- ilearnrw::games::NumbersToTextHelper::Abstract-Processor
 - getDigits, 59
 - getName, 59
 - NO_VALUE, 59
 - SEPARATOR, 59
- ilearnrw::games::NumbersToTextHelper::CompositeBig-Processor
 - CompositeBigProcessor, 62
- exponent, 62
- getHighProcessor, 62
- getLowProcessor, 62
- getName, 62
- getPartDivider, 62
- getToken, 62
- hundredProcessor, 62
- lowProcessor, 62
- ilearnrw::games::NumbersToTextHelper::Default-Processor
 - getName, 64
 - MINUS, 65
 - processor, 65
 - UNION_AND, 65
 - ZERO_TOKEN, 65
- ilearnrw::games::NumbersToTextHelper::Hundred-Processor
 - EXPONENT, 67
 - getName, 67
 - tensProcessor, 67
 - unitProcessor, 67
- ilearnrw::games::NumbersToTextHelper::Scale
 - getName, 67
 - SHORT, 67
- ilearnrw::games::NumbersToTextHelper::ScaleUnit
 - exponent, 68
 - getExponent, 68
 - getName, 68
 - names, 68
 - ScaleUnit, 68
- ilearnrw::games::NumbersToTextHelper::TensProcessor
 - getName, 71
 - TOKENS, 71
 - unitProcessor, 71
- ilearnrw::games::NumbersToTextHelper::UnitProcessor
 - getName, 72
 - TOKENS, 72
- ilearnrw::games::UsersProvider
 - main, 73
- ilearnrw::prototype::application::ConsoleMenu
 - ConsoleMenu, 75
 - doModalMenu, 76
 - in, 76
 - mIn, 76
 - mMenuActions, 76
 - mMenuTitle, 76
 - mOut, 76
 - mScanner, 76
 - out, 76
 - showMenu, 76
 - subMenu, 76
- ilearnrw::prototype::application::ConsoleMenu::E-ConsoleMenuActionResult
 - menuBreak, 76
 - showThisMenuAgain, 76
- ilearnrw::prototype::application::ConsoleMenu::IConsole-MenuAction
 - getText, 77

- onSelected, 77
- ilearnrw::prototype::application::ConsoleMenuAction
 - ConsoleMenuAction, 79
 - getText, 79
 - mText, 79
 - onSelected, 79
- ilearnrw::prototype::application::DatabaseManager
 - DatabaseManager, 81
 - onSelected, 81
- ilearnrw::prototype::application::DatabaseManager::I-
 - SelectUser
 - selectUser, 82
- ilearnrw::prototype::application::Program
 - getDataLogger, 83
 - getLoginProvider, 83
 - getProfileAccessUpdater, 83
 - getStringArg, 84
 - main, 84
 - sDataLogger, 84
 - sWelcomeMessage, 84
- ilearnrw::prototype::application::UserActions
 - addUserAction, 86
 - getAction, 86
 - getActions, 86
 - load, 86
 - mActions, 86
 - mFilePath, 86
 - mLogger, 86
 - save, 86
 - UserActions, 86
- ilearnrw::prototype::application::UserActionsLog
 - onSelected, 88
 - printOutActions, 89
 - UserActionsLog, 88
- ilearnrw::prototype::application::UserGames
 - mUser, 91
 - onSelected, 90
 - UserGames, 90
- ilearnrw::prototype::application::UserLogin
 - onSelected, 92
 - UserLogin, 92
- ilearnrw::prototype::application::UserProblems
 - addProblemDefinitionByCategory, 95
 - addProblemDefinitionByLanguage, 95
 - mUser, 95
 - onSelected, 95
 - UserProblems, 95
- ilearnrw::user::GameAction
 - DIFFICULT_WORD, 95
- ilearnrw::user::IlearnException
 - getMessage, 97
 - IlearnException, 97
- ilearnrw::user::LanguageCode
 - EN, 97
 - getEnglishCode, 97
 - getGreekCode, 97
- ilearnrw::user::Tester
 - main, 113
- ilearnrw::user::User
 - details, 116
 - getDetails, 116
 - getOldSessions, 116
 - getProfile, 116
 - getSession, 116
 - getUserId, 116
 - oldSessions, 116
 - profile, 116
 - serialVersionUID, 116
 - session, 116
 - setDetails, 116
 - setOldSessions, 116
 - setProfile, 116
 - setSession, 116
 - User, 116
 - userId, 116
- ilearnrw::user::UserAction
 - createRandomDate, 119
 - getApplicationId, 119
 - getTag, 119
 - getText, 119
 - getTimeStamp, 119
 - getUserId, 119
 - mApplicationId, 120
 - mTag, 120
 - mTimeStamp, 120
 - mUserId, 120
 - mValue, 120
 - randBetween, 120
 - serialVersionUID, 120
 - setApplicationId, 120
 - setUserId, 120
 - UserAction, 119
- ilearnrw::user::UserDetails
 - checkPassword, 123
 - desiredKeyLen, 124
 - getLanguage, 123
 - getUsername, 123
 - hasPassword, 123
 - hash, 123
 - iterations, 124
 - language, 124
 - mPasswordHash, 124
 - mPasswordSalt, 124
 - passwordIsSet, 123
 - saltLen, 124
 - serialVersionUID, 124
 - setLanguage, 123
 - setPassword, 124
 - setPasswordHash, 124
 - setUsername, 124
 - UserDetails, 123
 - username, 124
- ilearnrw::user::UserPreferences
 - fontSize, 126
 - getFontSize, 126
 - serialVersionUID, 126

- setFontSize, 126
 - UserPreferences, 126
- ilearnrw::user::UserProblem
 - getProblem, 129
 - getProblemNode, 129
 - index, 129
 - problem, 129
 - serialVersionUID, 129
 - setProblem, 129
 - toString, 129
 - UserProblem, 129
- ilearnrw::user::UserProblemsList
 - getList, 130
 - list, 131
 - loadTestProblems, 130
 - serialVersionUID, 131
 - setList, 130
 - UserProblemsList, 130
- ilearnrw::user::UserProfile
 - getLanguage, 133
 - getPreferences, 133
 - getProblemsList, 133
 - language, 133
 - preferences, 133
 - problems, 133
 - serialVersionUID, 133
 - setLanguage, 133
 - setPreferences, 133
 - setProblemsList, 133
 - UserProfile, 133
- ilearnrw::user::UserSession
 - end, 135
 - insertAction, 135
 - isActive, 135
 - serialVersionUID, 135
 - start, 135
 - startSession, 135
 - stopSession, 135
- ilearnrw::user::mockup::ProblemDefinitionIndexMockup
 - getProblemsByCategory, 99
 - getProblemsByLanguage, 99
 - index, 99
 - ProblemDefinitionIndexMockup, 99
- ilearnrw::user::problems::Category
 - Category, 101
 - equals, 101
 - getUrl, 101
 - isSubCategory, 102
 - serialVersionUID, 102
 - setUrl, 102
 - toString, 102
 - url, 102
- ilearnrw::user::problems::ProblemDefinition
 - addProblemNode, 104
 - availableLanguages, 104
 - getAvailableLanguages, 104
 - getProblemNode, 104
 - getProblemNodes, 104
 - getScoreUpperBound, 104
 - getType, 104
 - getUri, 104
 - ProblemDefinition, 104
 - problemNodes, 104
 - scoreUpperBound, 104
 - serialVersionUID, 104
 - setAvailableLanguages, 104
 - setProblemNodes, 104
 - setScoreUpperBound, 104
 - setType, 104
 - setUri, 104
 - toString, 104
 - type, 104
 - URI, 105
- ilearnrw::user::problems::ProblemDefinitionIndex
 - addProblemDefinition, 106
 - getAllProblems, 106
 - getProblemDefinition, 106
 - getProblemsByCategory, 106
 - getProblemsByLanguage, 107
 - index, 107
 - ProblemDefinitionIndex, 106
- ilearnrw::user::problems::ProblemDefinitionIndexApi
 - getProblemsByCategory, 108
 - getProblemsByLanguage, 108
- ilearnrw::user::problems::ProblemNode
 - children, 111
 - getChildren, 111
 - getNodeTitle, 111
 - nodeTitle, 111
 - ProblemNode, 110
 - serialVersionUID, 111
 - setChildren, 111
 - setNodeTitle, 111
- ilearnrw::user::test::MockupTests
 - testIndex, 112
- ilearnrw::user::test::UserTests
 - createProblems, 112
 - createUserProfile, 112
- in
 - ilearnrw::prototype::application::ConsoleMenu, 76
- index
 - ilearnrw::user::mockup::ProblemDefinitionIndexMockup, 99
 - ilearnrw::user::problems::ProblemDefinitionIndex, 107
 - ilearnrw::user::UserProblem, 129
- initialize
 - ilearnrw::games::NumberProblems, 53
- insertAction
 - ilearnrw::user::UserSession, 135
- isActive
 - ilearnrw::user::UserSession, 135
- isAuthenticated
 - ilearnrw::datalogger::IUserAdministration, 39
 - ilearnrw::datalogger::UserStore, 48
- isSubCategory

- ilearnrw::user::problems::Category, 102
- iterations
 - ilearnrw::user::UserDetails, 124
- language
 - ilearnrw::user::UserDetails, 124
 - ilearnrw::user::UserProfile, 133
- list
 - ilearnrw::user::UserProblemsList, 131
- load
 - ilearnrw::datalogger::UserStore, 48
 - ilearnrw::prototype::application::UserActions, 86
- loadTestProblems
 - ilearnrw::user::UserProblemsList, 130
- loadUserActions
 - ilearnrw::datalogger::DataLogger, 22
- loadUserStore
 - ilearnrw::datalogger::DataLogger, 22
- logAction
 - ilearnrw::datalogger::DataLogger, 22
 - ilearnrw::datalogger::IDataLogger, 26
- lowProcessor
 - ilearnrw::games::NumbersToTextHelper::Composite-BigProcessor, 62
- mActions
 - ilearnrw::prototype::application::UserActions, 86
- mApplicationId
 - ilearnrw::datalogger::UserActionFilter, 43
 - ilearnrw::games::GameAddTwoNumbers, 51
 - ilearnrw::user::UserAction, 120
- mDataLogger
 - ilearnrw::games::NumbersToText, 56
- mFilePath
 - ilearnrw::datalogger::UserStore, 49
 - ilearnrw::prototype::application::UserActions, 86
- mHasAdminAuth
 - ilearnrw::datalogger::UserStore, 49
- MINUS
 - ilearnrw::games::NumbersToTextHelper::Default-Processor, 65
- mId
 - ilearnrw::application::ApplicationId, 19
- mIn
 - ilearnrw::prototype::application::ConsoleMenu, 76
- mLoadedUsers
 - ilearnrw::datalogger::UserStore, 49
- mLogger
 - ilearnrw::datalogger::UserStore, 49
 - ilearnrw::games::GameAddTwoNumbers, 51
 - ilearnrw::prototype::application::UserActions, 86
- mMarker
 - ilearnrw::datalogger::DataLogger, 24
- mMenuActions
 - ilearnrw::prototype::application::ConsoleMenu, 76
- mMenuTitle
 - ilearnrw::prototype::application::ConsoleMenu, 76
- mOut
 - ilearnrw::prototype::application::ConsoleMenu, 76
- mPasswordHash
 - ilearnrw::user::UserDetails, 124
- mPasswordSalt
 - ilearnrw::user::UserDetails, 124
- mScanner
 - ilearnrw::prototype::application::ConsoleMenu, 76
- mTag
 - ilearnrw::user::UserAction, 120
- mTags
 - ilearnrw::datalogger::UserActionFilter, 43
- mText
 - ilearnrw::prototype::application::ConsoleMenu-Action, 79
- mTimeEnd
 - ilearnrw::datalogger::UserActionFilter, 43
- mTimeStamp
 - ilearnrw::user::UserAction, 120
- mTimeStart
 - ilearnrw::datalogger::UserActionFilter, 43
- mUser
 - ilearnrw::datalogger::DataLogger, 24
 - ilearnrw::games::GameAddTwoNumbers, 51
 - ilearnrw::games::NumbersToText, 56
 - ilearnrw::prototype::application::UserGames, 91
 - ilearnrw::prototype::application::UserProblems, 95
- mUserActions
 - ilearnrw::datalogger::DataLogger, 24
 - ilearnrw::games::GameAddTwoNumbers, 51
- mUserId
 - ilearnrw::datalogger::UserActionFilter, 43
 - ilearnrw::games::GameAddTwoNumbers, 51
 - ilearnrw::user::UserAction, 120
- mUserIsDirty
 - ilearnrw::datalogger::DataLogger, 24
- mUserStore
 - ilearnrw::datalogger::DataLogger, 24
- mValue
 - ilearnrw::user::UserAction, 120
- mWelcomeMessage
 - ilearnrw::games::GameAddTwoNumbers, 51
- main
 - ilearnrw::games::NumbersToTextHelper, 57
 - ilearnrw::games::UsersProvider, 73
 - ilearnrw::prototype::application::Program, 84
 - ilearnrw::user::Tester, 113
- menuBreak
 - ilearnrw::prototype::application::ConsoleMenu::E-ConsoleMenuActionResult, 76
- moveToOldSessions
 - ilearnrw::datalogger::DataLogger, 22
 - ilearnrw::datalogger::IProfileAccessUpdater, 31
- NO_VALUE
 - ilearnrw::games::NumbersToTextHelper::Abstract-Processor, 59
- names
 - ilearnrw::games::NumbersToTextHelper::Scale-Unit, 68
- nodeTitle

- ilearnrw::user::problems::ProblemNode, 111
- NumberProblems
 - ilearnrw::games::NumberProblems, 53
- NumbersToText
 - ilearnrw::games::NumbersToText, 56
- OK_TAG
 - ilearnrw::games::NumbersToText, 56
- oldSessions
 - ilearnrw::user::User, 116
- onSelected
 - ilearnrw::games::NumbersToText, 56
 - ilearnrw::prototype::application::ConsoleMenu::I-
ConsoleMenuAction, 77
 - ilearnrw::prototype::application::ConsoleMenu-
Action, 79
 - ilearnrw::prototype::application::DatabaseManager,
81
 - ilearnrw::prototype::application::UserActionsLog,
88
 - ilearnrw::prototype::application::UserGames, 90
 - ilearnrw::prototype::application::UserLogin, 92
 - ilearnrw::prototype::application::UserProblems, 95
- OnlineResourceBankTodo, 10
- out
 - ilearnrw::prototype::application::ConsoleMenu, 76
- passwordIsSet
 - ilearnrw::user::UserDetails, 123
- PendingChangesAvailable
 - ilearnrw::datalogger::IProfileAccessUpdater::-
PendingChangesAvailable, 34
- PermissionException
 - ilearnrw::datalogger::IProfileAccessUpdater::-
PermissionException, 36
- preferences
 - ilearnrw::user::UserProfile, 133
- printOutActions
 - ilearnrw::games::GameAddTwoNumbers, 51
 - ilearnrw::games::NumbersToText, 56
 - ilearnrw::prototype::application::UserActionsLog,
89
- problem
 - ilearnrw::user::UserProblem, 129
- ProblemDefinition
 - ilearnrw::user::problems::ProblemDefinition, 104
- ProblemDefinitionIndex
 - ilearnrw::user::problems::ProblemDefinitionIndex,
106
- ProblemDefinitionIndexMockup
 - ilearnrw::user::mockup::ProblemDefinitionIndex-
Mockup, 99
- ProblemNode
 - ilearnrw::user::problems::ProblemNode, 110
- problemNodes
 - ilearnrw::user::problems::ProblemDefinition, 104
- problems
 - ilearnrw::user::UserProfile, 133
- processor
 - ilearnrw::games::NumbersToTextHelper, 57
 - ilearnrw::games::NumbersToTextHelper::Default-
Processor, 65
- profile
 - ilearnrw::user::User, 116
- randBetween
 - ilearnrw::user::UserAction, 120
- read
 - ilearnrw::datalogger::UserStore, 48
- read_noCopy
 - ilearnrw::datalogger::UserStore, 48
- registerUserSession
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 31
- removeUserProblem
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 31
- SCALE
 - ilearnrw::games::NumbersToTextHelper, 57
- SCALE_UNITS
 - ilearnrw::games::NumbersToTextHelper, 57
- sDataLogger
 - ilearnrw::prototype::application::Program, 84
- SEPARATOR
 - ilearnrw::games::NumbersToTextHelper::Abstract-
Processor, 59
- SHORT
 - ilearnrw::games::NumbersToTextHelper::Scale, 67
- sWelcomeMessage
 - ilearnrw::prototype::application::Program, 84
- saltLen
 - ilearnrw::user::UserDetails, 124
- save
 - ilearnrw::datalogger::UserStore, 48
 - ilearnrw::prototype::application::UserActions, 86
- ScaleUnit
 - ilearnrw::games::NumbersToTextHelper::Scale-
Unit, 68
- scoreUpperBound
 - ilearnrw::user::problems::ProblemDefinition, 104
- selectUser
 - ilearnrw::prototype::application::DatabaseManager-
::ISelectUser, 82
- serialVersionUID
 - ilearnrw::application::ApplicationId, 19
 - ilearnrw::datalogger::IProfileAccessUpdater::-
PendingChangesAvailable, 34
 - ilearnrw::datalogger::IProfileAccessUpdater::-
PermissionException, 36
 - ilearnrw::datalogger::IUserAdministration::Authentication-
Exception, 41
 - ilearnrw::user::problems::Category, 102
 - ilearnrw::user::problems::ProblemDefinition, 104
 - ilearnrw::user::problems::ProblemNode, 111
 - ilearnrw::user::User, 116
 - ilearnrw::user::UserAction, 120
 - ilearnrw::user::UserDetails, 124

- ilearnrw::user::UserPreferences, 126
 - ilearnrw::user::UserProblem, 129
 - ilearnrw::user::UserProblemsList, 131
 - ilearnrw::user::UserProfile, 133
 - ilearnrw::user::UserSession, 135
- session
 - ilearnrw::user::User, 116
- setApplicationId
 - ilearnrw::user::UserAction, 120
- setAvailableLanguages
 - ilearnrw::user::problems::ProblemDefinition, 104
- setChildren
 - ilearnrw::user::problems::ProblemNode, 111
- setCurrentUser
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 32
- setDetails
 - ilearnrw::user::User, 116
- setDirty
 - ilearnrw::datalogger::DataLogger, 23
- setFontSize
 - ilearnrw::user::UserPreferences, 126
- setLanguage
 - ilearnrw::user::UserDetails, 123
 - ilearnrw::user::UserProfile, 133
- setList
 - ilearnrw::user::UserProblemsList, 130
- setNodeTitle
 - ilearnrw::user::problems::ProblemNode, 111
- setOldSessions
 - ilearnrw::user::User, 116
- setPassword
 - ilearnrw::user::UserDetails, 124
- setPasswordHash
 - ilearnrw::user::UserDetails, 124
- setPreference
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 32
- setPreferences
 - ilearnrw::user::UserProfile, 133
- setProblem
 - ilearnrw::user::UserProblem, 129
- setProblemNodes
 - ilearnrw::user::problems::ProblemDefinition, 104
- setProblemsList
 - ilearnrw::user::UserProfile, 133
- setProfile
 - ilearnrw::user::User, 116
- setScoreUpperBound
 - ilearnrw::user::problems::ProblemDefinition, 104
- setSession
 - ilearnrw::user::User, 116
- setType
 - ilearnrw::user::problems::ProblemDefinition, 104
- setURI
 - ilearnrw::user::problems::ProblemDefinition, 104
- setUrl
 - ilearnrw::user::problems::Category, 102
- setUserDetail
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 32
- setUserId
 - ilearnrw::user::UserAction, 120
- setUserLanguage
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 32
- setUsername
 - ilearnrw::user::UserDetails, 124
- showMenu
 - ilearnrw::prototype::application::ConsoleMenu, 76
- showThisMenuAgain
 - ilearnrw::prototype::application::ConsoleMenu::E-ConsoleMenuActionResult, 76
- start
 - ilearnrw::user::UserSession, 135
- startSession
 - ilearnrw::user::UserSession, 135
- stopSession
 - ilearnrw::user::UserSession, 135
- subMenu
 - ilearnrw::prototype::application::ConsoleMenu, 76
- TOKENS
 - ilearnrw::games::NumbersToTextHelper::Tens-Processor, 71
 - ilearnrw::games::NumbersToTextHelper::Unit-Processor, 72
- tensProcessor
 - ilearnrw::games::NumbersToTextHelper::Hundred-Processor, 67
- testIndex
 - ilearnrw::user::test::MockupTests, 112
- toString
 - ilearnrw::user::problems::Category, 102
 - ilearnrw::user::problems::ProblemDefinition, 104
 - ilearnrw::user::UserProblem, 129
- type
 - ilearnrw::user::problems::ProblemDefinition, 104
- UNION_AND
 - ilearnrw::games::NumbersToTextHelper::Default-Processor, 65
- UNION_SEPARATOR
 - ilearnrw::games::NumbersToTextHelper::Tens-Processor, 71
- URI
 - ilearnrw::user::problems::ProblemDefinition, 105
- unitProcessor
 - ilearnrw::games::NumbersToTextHelper::Hundred-Processor, 67
 - ilearnrw::games::NumbersToTextHelper::Tens-Processor, 71
- update
 - ilearnrw::datalogger::UserStore, 48
 - ilearnrw::games::GameAddTwoNumbers, 51
- updatePassword
 - ilearnrw::datalogger::IUserAdministration, 39

- ilearnrw::datalogger::UserStore, 49
- url
 - ilearnrw::user::problems::Category, 102
- User
 - ilearnrw::user::User, 116
- UserAction
 - ilearnrw::user::UserAction, 119
- UserActionFilter
 - ilearnrw::datalogger::UserActionFilter, 43
- UserActions
 - ilearnrw::prototype::application::UserActions, 86
- UserActionsLog
 - ilearnrw::prototype::application::UserActionsLog, 88
- UserDetails
 - ilearnrw::user::UserDetails, 123
- UserGames
 - ilearnrw::prototype::application::UserGames, 90
- userId
 - ilearnrw::user::User, 116
- UserLogin
 - ilearnrw::prototype::application::UserLogin, 92
- UserPreferences
 - ilearnrw::user::UserPreferences, 126
- UserProblem
 - ilearnrw::user::UserProblem, 129
- UserProblems
 - ilearnrw::prototype::application::UserProblems, 95
- UserProblemsList
 - ilearnrw::user::UserProblemsList, 130
- UserProfile
 - ilearnrw::user::UserProfile, 133
- UserStore
 - ilearnrw::datalogger::UserStore, 46
- username
 - ilearnrw::user::UserDetails, 124
- writePendingChanges
 - ilearnrw::datalogger::DataLogger, 23
 - ilearnrw::datalogger::IProfileAccessUpdater, 32
- ZERO_TOKEN
 - ilearnrw::games::NumbersToTextHelper::Default-Processor, 65