
SGAD



protech.unipd@gmail.com

DEFINIZIONE DI PRODOTTO v2.00

Nome del documento	Definizione di Prodotto
Versione del documento	2.00
Data redazione	2014/01/31
Redazione	Biancucci Maurizio Nessi Alberto Gallo Francesco
Verifica	Segantin Fabio
Approvazione	Esterno
Uso	Prof. Vardanega Tullio Prof. Cardin Riccardo FunGo Studios S.r.l.
Distribuzione	

Sommario

Questo documento vuol definire l'architettura in dettaglio del prodotto SGAD.

Diario delle modifiche

Modifica	Autore	Ruolo	Data	Versione
<i>Approvazione del documento</i>	Segantin Fabio	Project Manager	2014/03/13	v 2.00
<i>Verifica del documento</i>	Gallo Francesco	Verificatore	2014/03/13	v 1.09
<i>Approfondito riferimento informativo di Spray</i>	Biancucci Maurizio	Amministratore	2014/03/12	v 1.08
<i>Corretto diagramma di sequenza "Gestione degli eventi"</i>	Nessi Alberto	Progettista	2014/03/12	v 1.07
<i>Approfondimento delle descrizioni di varie classi e componenti, in particolare la componente operation</i>	Nessi Alberto	Progettista	2014/03/12	v 1.06
<i>Corretto uso del termine "segnatura" con il termine "firma"</i>	Nessi Alberto	Progettista	2014/03/12	v 1.05
<i>Aggiunti metodi di utilità interna ad alcune classi del pacchetto operation</i>	Nessi Alberto	Progettista	2014/03/08	v 1.04
<i>Aggiunta classe: InternalRequester al pacchetto usermanager. Aggiunte classi ToInternalRequester-Request e ToPublisherAndUser-InternalRequest al pacchetto messages</i>	Nessi Alberto	Progettista	2014/03/08	v 1.03

<i>Aggiornata firma del metodo execute in tutte le classi del pacchetto operation che posseggono il metodo. Rimosse le classi LoadAnotherUser e ReceiveLoadAnotherUser. Aggiunto parametro alla classe STimeout</i>	Nessi Alberto	Progettista	2014/03/08	v 1.02
<i>Rimosso il parametro piggy dalla classe Logic e spostato in UserData, al quale sono stati aggiunti metodi</i>	Nessi Alberto	Progettista	2014/03/08	v 1.01
<i>Approvazione del documento</i>	Nessi Alberto	Project Manager	2014/02/05	v 1.00
<i>Verifica del documento</i>	Gatto Francesco	Progettista	2014/02/03	v 0.47
<i>Inserita sezione "Schema della base di dati"</i>	Biancucci Maurizio	Progettista	2014/02/28	v 0.46
<i>Stesa sezione "sgad::clienttier::model::userdatamanager"</i>	Biancucci Maurizio	Progettista	2014/02/28	v 0.45
<i>Stesa sezione "sgad::clienttier::controller::backupmanager"</i>	Battistella Stefano	Progettista	2014/02/26	v 0.44
<i>Verifica del documento</i>	Gatto Francesco	Verificatore	2014/02/24	v 0.43
<i>Correzione degli errori individuati</i>	Segantin Fabio	Progettista	2014/02/24	v 0.42
<i>Verifica del documento</i>	Gatto Francesco	Verificatore	2014/02/23	v 0.41
<i>Stesa sezione "sgad::servertier::application"</i>	Segantin Fabio	Progettista	2014/02/22	v 0.40
<i>Stesa sezione "sgad::servertier::presentation::cluster"</i>	Segantin Fabio	Progettista	2014/02/22	v 0.39
<i>Stesa sezione "sgad::servertier::presentation::httpresponder"</i>	Segantin Fabio	Progettista	2014/02/22	v 0.38
<i>Stesa sezione "sgad::servertier::presentation::usermanager"</i>	Segantin Fabio	Progettista	2014/02/21	v 0.37

<i>Stesa sezione “sgad::-servertier::presentation:::-messages”</i>	Segantin Fabio	Progettista	2014/02/21	v 0.36
<i>Stesa sezione “sgad::-servertier::presentation:::-timeout”</i>	Segantin Fabio	Progettista	2014/02/20	v 0.35
<i>Stesa sezione “sgad::-servertier::presentation:::-pagemanager”</i>	Segantin Fabio	Progettista	2014/02/20	v 0.34
<i>Stesa sezione “sgad::-servertier::businesslogic:::-logic”</i>	Segantin Fabio	Progettista	2014/02/20	v 0.33
<i>Stesa sezione “sgad::-servertier::businesslogic:::-operations”</i>	Segantin Fabio	Progettista	2014/02/19	v 0.32
<i>Stesa sezione “sgad::-servertier::dataaccess:::-databaseaccess:::-userdatadao”</i>	Gallo Francesco	Progettista	2014/02/19	v 0.31
<i>Stesa sezione “sgad::-servertier::dataaccess:::-databaseaccess:::-shareddatadao”</i>	Gallo Francesco	Progettista	2014/02/19	v 0.30
<i>Stesa sezione “sgad::-servertier::dataaccess:::-databaseaccess:::-databasemanager”</i>	Gallo Francesco	Progettista	2014/02/19	v 0.29
<i>Stesa sezione “sgad::-servertier::dataaccess:::-databaseaccess”</i>	Segantin Fabio	Progettista	2014/02/18	v 0.28
<i>Stesa sezione “sgad::-servertier::dataaccess:::-data::userdata”</i>	Segantin Fabio	Progettista	2014/02/17	v 0.27
<i>Stesa sezione “sgad::-servertier::dataaccess:::-data::shareddata”</i>	Segantin Fabio	Progettista	2014/02/16	v 0.26
<i>Stesa sezione “sgad::-clienttier::controller:::-actions”</i>	Segantin Fabio	Progettista	2014/02/15	v 0.25

<i>Stesa sezione “sgad::clienttier::controller::menufactory”</i>	Gallo Francesco	Progettista	2014/02/15	v 0.24
<i>Stesa sezione “sgad::clienttier::view::context”</i>	Gallo Francesco	Progettista	2014/02/14	v 0.23
<i>Verifica del documento</i>	Gallo Francesco	Progettista	2014/02/13	v 0.22
<i>Correzione degli errori individuati</i>	Gallo Francesco	Progettista	2014/02/12	v 0.21
<i>Verifica del documento</i>	Gallo Francesco	Progettista	2014/02/11	v 0.20
<i>Stesa sezione “Tracciamento”</i>	Gallo Francesco	Progettista	2014/02/11	v 0.19
<i>Stesa sezione “Diagrammi di sequenza”</i>	Gallo Francesco	Progettista	2014/02/09	v 0.18
<i>Stesa sezione “sgad::clienttier::view::commands”</i>	Gallo Francesco	Progettista	2014/02/08	v 0.17
<i>Stesa sezione “sgad::clienttier::view::graphicobjects::components::filter”</i>	Segantin Fabio	Progettista	2014/02/07	v 0.16
<i>Stesa sezione “sgad::clienttier::view::graphicobjects::components::worldcomponent”</i>	Segantin Fabio	Progettista	2014/02/06	v 0.15
<i>Stesa sezione “sgad::clienttier::view::graphicobjects::components::worldcomponentshape”</i>	Segantin Fabio	Progettista	2014/02/05	v 0.14
<i>Stesa sezione “sgad::clienttier::view::graphicobjects::collection”</i>	Segantin Fabio	Progettista	2014/02/05	v 0.13
<i>Stesa sezione “sgad::clienttier::view::graphicobjects::bound”</i>	Gallo Francesco	Progettista	2014/02/04	v 0.12
<i>Stesa sezione “sgad::clienttier::view::graphicobjects::graphicobject”</i>	Segantin Fabio	Progettista	2014/02/04	v 0.11

<i>Stesa sezione “sgad::clienttier::view:::graphicobjects::shape”</i>	Segantin Fabio	Progettista	2014/02/04	v 0.10
<i>Stesa sezione “sgad::clienttier::view:::graphicobjects::point”</i>	Segantin Fabio	Progettista	2014/02/04	v 0.09
<i>Stesa sezione “sgad::clienttier::model:::personaldata”</i>	Segantin Fabio	Progettista	2014/02/03	v 0.08
<i>Stesa sezione “sgad::clienttier::model:::generaldata”</i>	Segantin Fabio	Progettista	2014/02/03	v 0.07
<i>Stesa sezione “sgad::clienttier::model:::observer”</i>	Gallo Francesco	Progettista	2014/02/02	v 0.06
<i>Stesa sezione “sgad::clienttier::controller:::requester”</i>	Segantin Fabio	Progettista	2014/02/02	v 0.05
<i>Stesa sezione “sgad::clienttier::controller:::messageinterpreter”</i>	Segantin Fabio	Progettista	2014/02/01	v 0.04
<i>Inizio stesura sezione “Specifiche dei componenti”</i>	Segantin Fabio	Progettista	2014/02/01	v 0.03
<i>Stesura della sezione “Standard di progetto”</i>	Gallo Francesco	Progettista	2014/02/01	v 0.02
<i>Inizio stesura documento con impostazione generale dello scheletro del documento</i>	Segantin Fabio	Progettista	2014/01/31	v 0.01

Indice

1	Introduzione	15
1.1	Scopo del documento	15
1.2	Scopo del Prodotto	15
1.3	Glossario	15
1.4	Riferimenti	15
1.4.1	Normativi	15
1.4.2	Informativi	15
2	Standard di progetto	17
2.1	Standard di progettazione architetturale	17
2.2	Standard di documentazione del codice	17
2.3	Standard di denominazione di entità e relazioni	17
2.4	Standard di programmazione	17
2.5	Strumenti di lavoro	17
3	Specifiche dei componenti	18
3.1	Metodo e formalismo di specifica	18
3.2	sgad	18
3.2.1	Informazioni sul package	18
3.3	sgad::clienttier	19
3.3.1	Informazioni sul package	19
3.4	sgad::clienttier::controller	20
3.4.1	Informazioni sul package	20
3.5	sgad::clienttier::controller::actions	21
3.5.1	Informazioni sul package	21
3.5.2	Classi	23
3.5.2.1	Action	23
3.5.2.2	ActionListener	25
3.5.2.3	BackToVillage	26
3.5.2.4	BuildConstruction	26
3.5.2.5	ChangePasswordAction	28
3.5.2.6	CloseContextualMenu	28
3.5.2.7	DeleteAccountAction	29
3.5.2.8	DemolishBuilding	29
3.5.2.9	DismissUnits	30
3.5.2.10	EnableRightClick	31
3.5.2.11	HarvestResources	32
3.5.2.12	LoadAnotherUser	33
3.5.2.13	LoadGeneralData	34
3.5.2.14	LoadPersonalData	36
3.5.2.15	Logout	38
3.5.2.16	ReceiveStealResources	38
3.5.2.17	ReloadPageAction	39
3.5.2.18	RemoveGraphicObjectAction	40

3.5.2.19	RequestAttackUser	40
3.5.2.20	RequestBuildConstruction	41
3.5.2.21	RequestChangePassword	42
3.5.2.22	RequestDeleteAccount	44
3.5.2.23	RequestDemolishBuilding	44
3.5.2.24	RequestDismissUnits	45
3.5.2.25	RequestHarvestResources	46
3.5.2.26	RequestLoadGeneralData	46
3.5.2.27	RequestLoadPersonalData	47
3.5.2.28	RequestStoleResources	48
3.5.2.29	RequestTrainUnits	49
3.5.2.30	RequestUpgradeBuilding	50
3.5.2.31	SetAllVoidFilter	50
3.5.2.32	SetBuildModeFilter	51
3.5.2.33	ShowAccountDeletedMenu	52
3.5.2.34	ShowAccountManagerMenu	52
3.5.2.35	ShowAnotherUserMenu	53
3.5.2.36	ShowAttackMenu	54
3.5.2.37	ShowAttackResultMenu	55
3.5.2.38	ShowBuildConstructionMenu	56
3.5.2.39	ShowBuildingContextualMenu	56
3.5.2.40	ShowDemolishMenu	57
3.5.2.41	ShowDismissUnitMenu	58
3.5.2.42	ShowInteractionMenu	59
3.5.2.43	ShowOperationFailureMenu	60
3.5.2.44	ShowPasswordChangedMenu	60
3.5.2.45	ShowResourceMenu	61
3.5.2.46	ShowTileContextualMenu	62
3.5.2.47	ShowTrainUnitMenu	62
3.5.2.48	ShowUnitSelectionMenu	63
3.5.2.49	ShowUpgradeMenu	64
3.5.2.50	ShowUserListMenu	65
3.5.2.51	StealResources	66
3.5.2.52	TrainUnit	66
3.5.2.53	UpgradeBuilding	68
3.5.2.54	RequestUserList	69
3.5.2.55	ShowLogMenu	69
3.6	sgad::clienttier::controller::backupmanager	70
3.6.1	Informazioni sul package	70
3.6.2	Classi	70
3.6.2.1	BackupManager	70
3.7	sgad::clienttier::controller::menufactory	72
3.7.1	Informazioni sul package	72
3.7.2	Classi	72
3.7.2.1	MenuFactory	72
3.7.2.2	AccountManagerMenuFactory	73

3.7.2.3	AnotherUserMenuFactory	74
3.7.2.4	AttackResultMenuFactory	75
3.7.2.5	BuildingContextualMenuFactory	77
3.7.2.6	ConfirmMenuFactory	79
3.7.2.7	InteractionMenuFactory	80
3.7.2.8	LogMenuFactory	80
3.7.2.9	NotifyMenuFactory	81
3.7.2.10	ResourceMenuFactory	82
3.7.2.11	TileContextualMenuFactory	84
3.7.2.12	UnitSelectionMenu	85
3.7.2.13	UserListMenuFactory	86
3.8	sgad::clienttier::controller::messageinterpreter	88
3.8.1	Informazioni sul package	88
3.8.2	Classi	88
3.8.2.1	MessageInterpreter	88
3.9	sgad::clienttier::controller::requester	89
3.9.1	Informazioni sul package	89
3.9.2	Classi	90
3.9.2.1	AJAXRequester	90
3.10	sgad::clienttier::model	92
3.10.1	Informazioni sul package	92
3.11	sgad::clienttier::model::generaldata	93
3.11.1	Informazioni sul package	93
3.11.2	Classi	94
3.11.2.1	Bonus	94
3.11.2.2	BuildingWithLevel	95
3.11.2.3	Cost	98
3.11.2.4	DataFactory	99
3.11.2.5	ProducedResource	101
3.11.2.6	QuantityResource	102
3.11.2.7	Resource	103
3.11.2.8	Unit	104
3.12	sgad::clienttier::model::observer	106
3.12.1	Informazioni sul package	106
3.12.2	Classi	107
3.12.2.1	Observable	107
3.12.2.2	Observer	108
3.13	sgad::clienttier::model::personaldata	109
3.13.1	Informazioni sul package	109
3.13.2	Classi	110
3.13.2.1	AuthenticationData	110
3.13.2.2	BuildingPossession	111
3.13.2.3	OwnedResource	113
3.13.2.4	Position	115
3.13.2.5	UnitPossession	116
3.13.2.6	UnitInProgress	117

3.13.2.7	UserData	120
3.14	sgad::clienttier::model::userdatamanager	122
3.14.1	Informazioni sul package	122
3.14.2	Classi	122
3.14.2.1	UserDataManager	122
3.15	sgad::clienttier::view	124
3.15.1	Informazioni sul package	124
3.16	sgad::clienttier::view::commands	125
3.16.1	Informazioni sul package	125
3.16.2	Classi	125
3.16.2.1	Command	125
3.16.2.2	Click	126
3.16.2.3	DragAndDrop	127
3.16.2.4	RightClick	128
3.17	sgad::clienttier::view::context	129
3.17.1	Informazioni sul package	129
3.17.2	Classi	130
3.17.2.1	Context	130
3.18	sgad::clienttier::view::graphicobjects	133
3.18.1	Informazioni sul package	133
3.19	sgad::clienttier::view::graphicobjects::bound	134
3.19.1	Informazioni sul package	134
3.19.2	Classi	135
3.19.2.1	Bound	135
3.20	sgad::clienttier::view::graphicobjects::collection	137
3.20.1	Informazioni sul package	137
3.20.2	Classi	138
3.20.2.1	Collection	138
3.20.2.2	GraphicObjectCollection	138
3.20.2.3	Iterator	140
3.20.2.4	GraphicObjectIterator	141
3.21	sgad::clienttier::view::graphicobjects::components	143
3.21.1	Informazioni sul package	143
3.22	sgad::clienttier::view::graphicobjects::components::filter	144
3.22.1	Informazioni sul package	144
3.22.2	Classi	144
3.22.2.1	GraphicFilter	144
3.22.2.2	BuildModeFilter	145
3.22.2.3	VoidFilter	146
3.23	sgad::clienttier::view::graphicobjects::components::worldcomponent	148
3.23.1	Informazioni sul package	148
3.23.2	Classi	148
3.23.2.1	WorldComponent	148
3.23.2.2	BuildingComponent	149
3.23.2.3	TileComponent	152
3.24	sgad::clienttier::view::graphicobjects::components::worldcomponentshape	154

3.24.1	Informazioni sul package	154
3.24.2	Classi	155
3.24.2.1	WorldComponentShapeFactory	155
3.24.2.2	WorldComponentShapeImg	155
3.24.2.3	BarracksL1Shape	157
3.24.2.4	BarracksL2Shape	157
3.24.2.5	BarracksL3Shape	158
3.24.2.6	InConstructionShape	158
3.24.2.7	MineL1Shape	159
3.24.2.8	MineL2Shape	160
3.24.2.9	MineL3Shape	160
3.24.2.10	SchoolOfMagicL1Shape	161
3.24.2.11	SchoolOfMagicL2Shape	161
3.24.2.12	SchoolOfMagicL3Shape	162
3.24.2.13	StableL1Shape	162
3.24.2.14	StableL2Shape	163
3.24.2.15	StableL3Shape	163
3.24.2.16	TileShape	164
3.24.2.17	WizardTowerL1Shape	164
3.24.2.18	WizardTowerL2Shape	165
3.24.2.19	WizardTowerL3Shape	166
3.25	sgad::clienttier::view::graphicobjects::graphicobject	167
3.25.1	Informazioni sul package	167
3.25.2	Classi	167
3.25.2.1	GraphicObject	167
3.26	sgad::clienttier::view::graphicobjects::point	171
3.26.1	Informazioni sul package	171
3.26.2	Classi	171
3.26.2.1	Point2D	171
3.27	sgad::clienttier::view::graphicobjects::shape	173
3.27.1	Informazioni sul package	173
3.27.2	Classi	173
3.27.2.1	Shape	173
3.28	sgad::clienttier::view::graphicobjects::widget	175
3.28.1	Informazioni sul package	175
3.28.2	Classi	176
3.28.2.1	Widget	176
3.28.2.2	ButtonWidget	179
3.28.2.3	FrameWidget	181
3.28.2.4	ImageWidget	184
3.28.2.5	TextWidget	186
3.29	sgad::servertier	190
3.29.1	Informazioni sul package	190
3.30	sgad::servertier::application	191
3.30.1	Informazioni sul package	191
3.30.2	Classi	191

3.30.2.1	Application	191
3.31	sgad::servertier::businesslogic	192
3.31.1	Informazioni sul package	192
3.32	sgad::servertier::businesslogic::logic	193
3.32.1	Informazioni sul package	193
3.32.2	Classi	193
3.32.2.1	Logic	193
3.33	sgad::servertier::businesslogic::operations	196
3.33.1	Informazioni sul package	196
3.33.2	Classi	198
3.33.2.1	Operation	198
3.33.2.2	Attack	200
3.33.2.3	BuildConstruction	201
3.33.2.4	ChangeAccountData	202
3.33.2.5	DeleteAccount	203
3.33.2.6	DemolishBuilding	204
3.33.2.7	DismissUnit	205
3.33.2.8	DonateResources	206
3.33.2.9	DonateUnits	207
3.33.2.10	GetAllServerData	208
3.33.2.11	GetServerData	209
3.33.2.12	HarvestResource	210
3.33.2.13	InternalLogin	211
3.33.2.14	LoadGlobalData	212
3.33.2.15	LoadVillage	213
3.33.2.16	LoadUserList	214
3.33.2.17	Login	215
3.33.2.18	Logout	216
3.33.2.19	ReceiveAttack	216
3.33.2.20	ReceiveStealResource	219
3.33.2.21	Registration	220
3.33.2.22	SaveUser	221
3.33.2.23	StealResource	222
3.33.2.24	TrainUnit	223
3.33.2.25	UpdateUserData	224
3.33.2.26	UpgradeBuilding	226
3.33.2.27	OperationFactory	227
3.34	sgad::servertier::dataaccess	228
3.34.1	Informazioni sul package	228
3.35	sgad::servertier::dataaccess::data	228
3.35.1	Informazioni sul package	228
3.36	sgad::servertier::dataaccess::data::shareddata	230
3.36.1	Informazioni sul package	230
3.36.2	Classi	231
3.36.2.1	Bonus	231
3.36.2.2	BuildingWithLevel	232

3.36.2.3	Cost	235
3.36.2.4	DataFactory	237
3.36.2.5	ProducedResource	238
3.36.2.6	QuantityResource	240
3.36.2.7	Resource	241
3.36.2.8	Unit	242
3.37	sgad::servertier::dataaccess::data::userdata	244
3.37.1	Informazioni sul package	244
3.37.2	Classi	245
3.37.2.1	AuthenticationData	245
3.37.2.2	BuildingPossession	247
3.37.2.3	OwnedResource	249
3.37.2.4	Position	250
3.37.2.5	UnitPossession	252
3.37.2.6	UnitInProgress	253
3.37.2.7	UserData	254
3.38	sgad::servertier::dataaccess::databaseaccess	258
3.38.1	Informazioni sul package	258
3.39	sgad::servertier::dataaccess::databaseaccess::databasemanager	260
3.39.1	Informazioni sul package	260
3.39.2	Classi	261
3.39.2.1	DataBaseManager	261
3.40	sgad::servertier::dataaccess::databaseaccess::shareddatadao	264
3.40.1	Informazioni sul package	264
3.40.2	Classi	265
3.40.2.1	BonusDAO	265
3.40.2.2	BuildingWithLevelDAO	265
3.40.2.3	CostDAO	266
3.40.2.4	ProducedResourceDAO	267
3.40.2.5	QuantityResourceDAO	268
3.40.2.6	ResourceDAO	268
3.40.2.7	SharedDataDAO	269
3.40.2.8	UnitDAO	270
3.41	sgad::servertier::dataaccess::databaseaccess::userdatadao	271
3.41.1	Informazioni sul package	271
3.41.2	Classi	272
3.41.2.1	AuthenticationDataDAO	272
3.41.2.2	BuildingPossessionDAO	272
3.41.2.3	OwnedResourceDAO	273
3.41.2.4	PositionDAO	274
3.41.2.5	UnitInProgressDAO	274
3.41.2.6	UnitPossessionDAO	275
3.41.2.7	UserDataDAO	276
3.42	sgad::servertier::presentation	278
3.42.1	Informazioni sul package	278
3.43	sgad::servertier::presentation::cluster	279

3.43.1	Informazioni sul package	279
3.43.2	Classi	279
3.43.2.1	ClusterListener	279
3.44	sgad::servertier::presentation::httpresponder	280
3.44.1	Informazioni sul package	280
3.44.2	Classi	281
3.44.2.1	ResponderActor	281
3.44.2.2	WorkerActor	282
3.45	sgad::servertier::presentation::messages	283
3.45.1	Informazioni sul package	283
3.45.2	Classi	284
3.45.2.1	ToInternalRequesterRequest	284
3.45.2.2	ToLoginActorRequest	285
3.45.2.3	ToPublisherAndUserInternalRequest	286
3.45.2.4	ToPublisherAndUserIsAliveRequest	287
3.45.2.5	ToPublisherAndUserRequest	288
3.45.2.6	ToRegistrationActorRequest	289
3.45.2.7	ToRequesterIsAliveRequest	290
3.45.2.8	ToWorkerLoginRequest	290
3.45.2.9	ToWorkerRegistrationRequest	292
3.45.2.10	ToWorkerUserRequest	293
3.45.2.11	UserActorTimeoutMessage	294
3.45.2.12	WorkerActorTimeoutMessage	294
3.46	sgad::servertier::presentation::pagemanager	295
3.46.1	Informazioni sul package	295
3.46.2	Classi	296
3.46.2.1	PageFactory	296
3.47	sgad::servertier::presentation::timeout	299
3.47.1	Informazioni sul package	299
3.47.2	Classi	299
3.47.2.1	STimeout	299
3.48	sgad::servertier::presentation::usermanager	302
3.48.1	Informazioni sul package	302
3.48.2	Classi	303
3.48.2.1	InternalRequester	303
3.48.2.2	IsUserActorAliveRequester	304
3.48.2.3	IsUserActorAliveResponder	305
3.48.2.4	LoginActor	306
3.48.2.5	PublisherActor	307
3.48.2.6	RegistrationActor	307
3.48.2.7	UserActor	308
4	Schema base di dati	311
5	Diagrammi di sequenza	312
5.1	Gestione degli eventi	312
5.2	Visualizzazione dei menu	313

5.3	Interazione con il server	314
5.4	Gestione di una richiesta di registrazione	315
5.5	Gestione di una richiesta di login	317
5.6	Gestione di una richiesta di gioco	319
6	Tracciamento	320
6.1	Tracciamento Classi-Requisiti	320
6.2	Tracciamento Requisiti-Classi	370

1 Introduzione

1.1 Scopo del documento

Il presente documento ha lo scopo di definire in dettaglio la struttura e le relazioni tra le componenti del *prodotto_{|g|}* SGAD, riprendendo e approfondendo quanto già descritto nel documento “*Specifica Tecnica v3.00*”. Tale documento servirà da guida per i Programmatori, fornendo loro le direttive per l’implementazione del *sistema_{|g|}* e l’attività di codifica.

1.2 Scopo del Prodotto

Il prodotto denominato SGAD si propone di fornire un sistema per il gioco online avente architettura distribuita. Il sistema permette all’utente, tramite una semplice interfaccia web, di utilizzare un *social game_{|g|}* di tipo strategico. Esso intende offrire all’utente un’esperienza di gioco nella quale si possano collezionare risorse, costruire edifici, ampliare la propria squadra ed eventualmente combattere con altri utenti.

1.3 Glossario

Al fine di evitare ogni ambiguità relativa al linguaggio e ai termini utilizzati nei documenti formali, viene allegato il “*Glossario v2.00*”. In tale documento vengono definiti e descritti tutti i termini evidenziati con l’apposita marcatura.

1.4 Riferimenti

1.4.1 Normativi

- “*Analisi dei Requisiti v6.00*”
- “*Norme di Progetto v5.00*”
- “*Specifica Tecnica v3.00*”

1.4.2 Informativi

- Scala for the Impatient (Cay Horstmann, Addison-Wesley, 2012)
- Effective Akka - Patterns and Best Practices (Jamie Allen, O'Reilly Media, 2013)
- Documentazione Akka-actors: <http://doc.akka.io/docs/akka/2.2.3/scala/actors.html>
- Documentazione Akka-cluster: <http://doc.akka.io/docs/akka/2.2.3/common/cluster.html>
- HTML5 Candidate Recommendation: <http://www.w3.org/TR/html5/>
- CSS, direttive W3C: <http://www.w3.org/TR/CSS/>
- Standard ECMA-262 di JavaScript: <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf>

- Standard ECMA-404 di JSON: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>
- Documentazione API jQuery: <http://api.jquery.com/>
- Documentazione spray: <http://spray.io/documentation/1.2.0/>
- Documentazione MongoDB: <http://docs.mongodb.org/manual/>
- Documentazione Casbah: <http://mongodb.github.io/casbah/>
- SWEBOK - Chapter 3: Software Design: <http://www.computer.org/portal/web/swebok/html/ch3>
- SWEBOK - Chapter 11: Software Quality: <http://www.computer.org/portal/web/swebok/html/ch11>
- Design Pattern - Elementi per il riuso di software ad oggetti (Gamma, Helm, Johnson, Vlissides, editore Pearson, 2002)
- Software Engineering - Chapter 6: Architectural design - Ian Sommerville - 9th ed. (2010)

2 Standard di progetto

2.1 Standard di progettazione architetturale

Gli standard di progettazione architetturale seguiti sono definiti nel documento “*Specifica Tecnica v3.00*”. Si faccia riferimento ad esso per approfondimenti.

2.2 Standard di documentazione del codice

Gli standard di documentazione del codice sono definiti e descritti nel documento “*Norme di Progetto v5.00*”. Si faccia riferimento ad esso per approfondimenti.

2.3 Standard di denominazione di entità e relazioni

Tutti gli elementi definiti nel seguente documento, siano essi $package_{|g|}$, classi, metodi o attributi, devono avere una denominazione chiara e concisa. La chiarezza del nome sarà anteposta alla sua lunghezza, che potrà venire appositamente abbreviata.

Sono ammesse abbreviazioni quando:

- non si altera l'immediata comprensibilità del nome;
- non si incorre nel rischio di essere ambigui.

Si faccia riferimento al documento “*Norme di Progetto v5.00*” per tutte le regole tipografiche adottate.

2.4 Standard di programmazione

Gli standard di programmazione sono definiti e descritti nel documento “*Norme di Progetto v5.00*”. Si rimanda ad esso per le regole da seguire durante la codifica.

2.5 Strumenti di lavoro

Tutti gli strumenti di lavoro e le procedure da seguire per la corretta realizzazione del prodotto sono definiti nel documento “*Norme di Progetto v5.00*”. Si faccia riferimento ad esso per approfondimenti.

3 Specifica dei componenti

3.1 Metodo e formalismo di specifica

L'esposizione dell'architettura dell'applicazione procede con un approccio di tipo top-down. Si descrive quindi l'architettura partendo dal generale, decomponendo le componenti fino a raggiungere il particolare. Si inizia quindi con la descrizione delle componenti. Per ogni componente si analizzeranno in dettaglio le singole classi, specificando per ognuna il tipo, l'obiettivo, la funzione, gli attributi e i metodi.

Per i diagrammi delle componenti, di classe e di attività si utilizza il formalismo *UML_g* 2.0. Per permettere di distinguere facilmente le classi e le componenti presenti in *librerie_g* e *framework_g* esterni utilizzati dall'applicazione, questi verranno rappresentati con un colore verde. Le classi e le componenti dell'applicazione sono invece raffigurati con un colore azzurro.

3.2 Componente sgad

3.2.1 Informazioni sul package

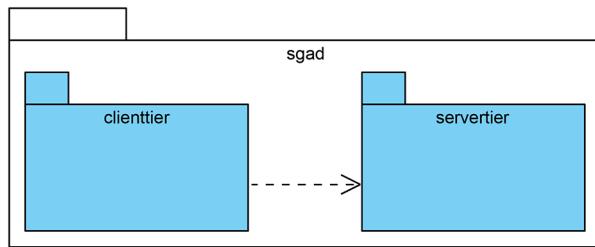


Figura 1: Diagramma della componente **sgad**

- **Descrizione:** componente che contiene il prodotto SGAD.

- **Package contenuti**

- **clienttier**: componente globale per il *front end_g* del prodotto. Nel *Three-Tier_g* del sistema rappresenta il livello Client *Tier_g*. Le relazioni tra le sottocomponenti **model**, **view** e **controller** rappresentano le interazioni tipiche che intercorrono tra le componenti del *design pattern_g* MVC. Esso si occupa di visualizzare i dati all'utente e di inoltrare le richieste al *back end_g*.
- **servertier**: componente globale per il *back end* del prodotto. Le relazioni tra le componenti **presentation**, **businesslogic** e **dataaccess** rappresentano delle interazioni tipiche. Esse intercorrono tra le componenti del design pattern Three-Tier.

3.3 Componente sgad::clienttier

3.3.1 Informazioni sul package

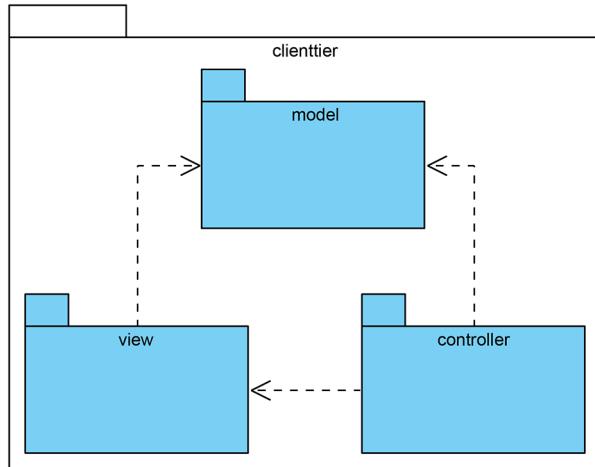


Figura 2: Diagramma della componente `sgad::clienttier`

- **Descrizione:** componente globale per il front end del prodotto. Nel Three-Tier del sistema rappresenta il livello Client Tier. Le relazioni tra le sottocomponenti **model**, **view** e **controller** rappresentano le interazioni tipiche che intercorrono tra le componenti del design pattern MVC. Esso si occupa di visualizzare i dati all'utente e di inoltrare le richieste al back end.
- **Padre:** `sgad`
- **Interazioni con altri componenti**
 - **servertier**: componente globale per il back end del prodotto. Le relazioni tra le componenti **presentation**, **businesslogic** e **dataaccess** rappresentano delle interazioni tipiche. Esse intercorrono tra le componenti del design pattern Three-Tier.
- **Package contenuti**
 - **controller**: componente Controller dell'architettura $MVC_{|g|}$. Essa esegue le operazioni che l'utente ha richiesto tramite la view agendo se necessario sul $model_{|g|}$ data.
 - **model**: componente Model dell'architettura MVC. Essa memorizza tutti i dati di gioco dell'utente su cui la view si basa per disegnarsi.
 - **view**: componente View dell'architettura MVC. Essa gestisce la visualizzazione del gioco e l'input utente all'interno del $canvas_{|g|}$ $HTML5_{|g|}$.

3.4 Componente sgad::clienttier::controller

3.4.1 Informazioni sul package

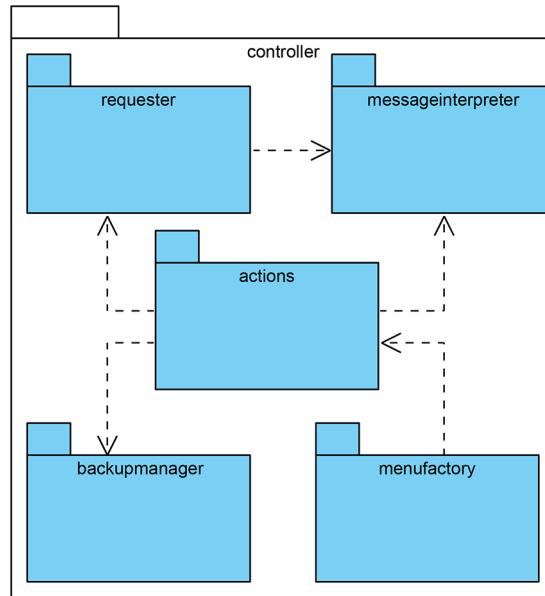


Figura 3: Diagramma della componente `sgad::clienttier::controller`

- **Descrizione:** componente Controller dell'architettura MVC. Essa esegue le operazioni che l'utente ha richiesto tramite la view agendo se necessario sul model data.
- **Padre:** `clienttier`
- **Interazioni con altri componenti**
 - `model`: componente Model dell'architettura MVC. Essa memorizza tutti i dati di gioco dell'utente su cui la view si basa per disegnarsi.
 - `view`: componente View dell'architettura MVC. Essa gestisce la visualizzazione del gioco e l'input utente all'interno del canvas HTML5.
- **Package contenuti**
 - `actions`: componente per la gestione delle possibili azioni che un utente può intraprendere durante l'interazione con il mondo di gioco.
 - `backupmanager`: componente che si occupa di gestire il mantenimento dei dati e delle viste dell'utente proprietario dell'account.
 - `menufactory`: componente per la gestione dei vari menu che possono essere visualizzati durante le interazioni con il mondo di gioco.
 - `messageinterpreter`: componente per l'interpretazione delle reply ricevute dal server.
 - `requester`: componente per la gestione dell'invio di richieste al server.

3.5 Componente sgad::clienttier::controller::actions

3.5.1 Informazioni sul package

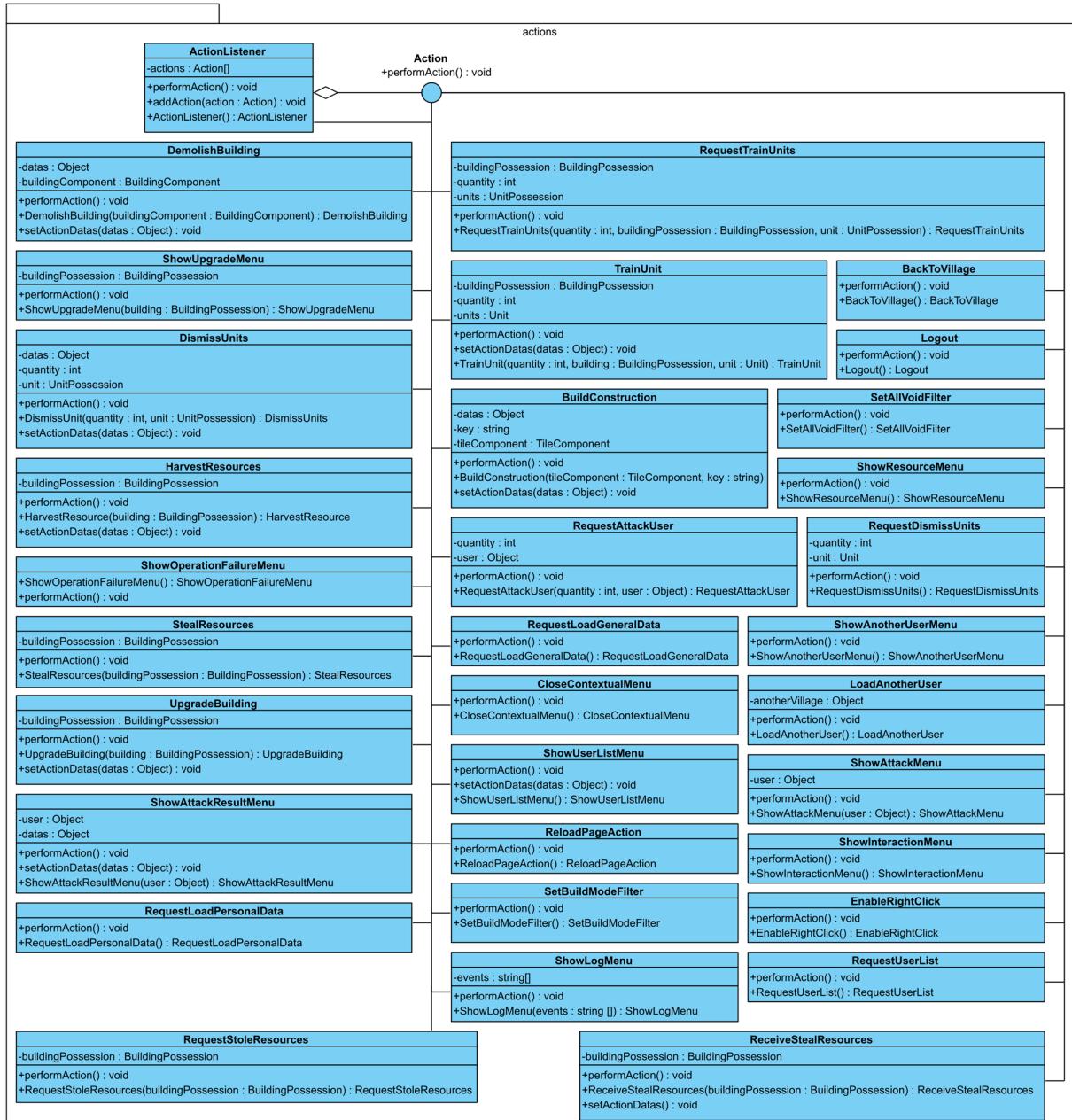


Figura 4: Diagramma della componente `sgad::clienttier::controller::actions`

3 Specifica dei componenti

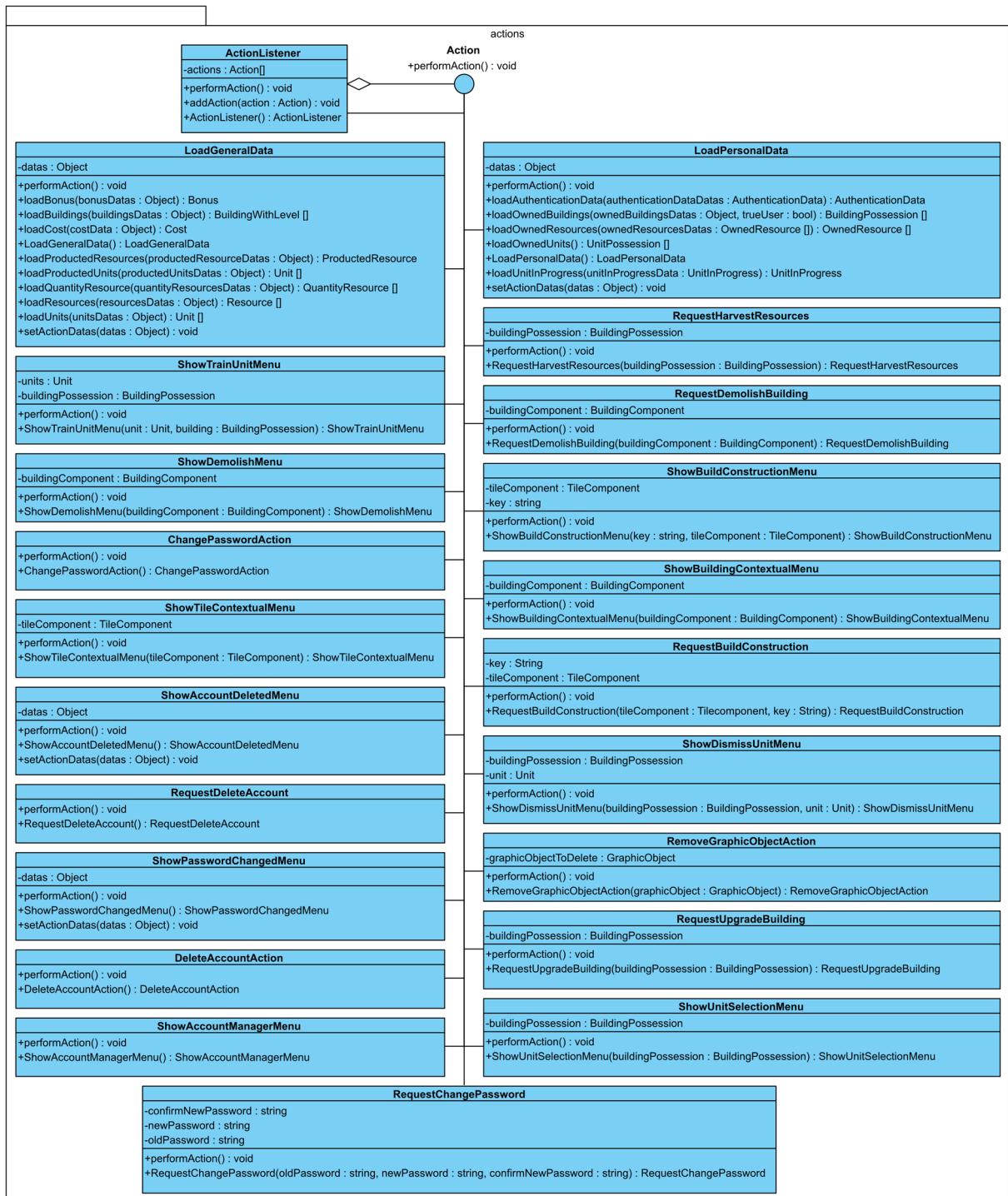


Figura 5: Diagramma della componente `sgad::clienttier::controller::actions`

- **Descrizione:** componente per la gestione delle possibili azioni che un utente può intraprendere durante l'interazione con il mondo di gioco.
- **Padre:** controller
- **Interazioni con altri componenti**
 - messageinterpreter: componente per l'interpretazione delle reply ricevute dal server.
 - requester: componente per la gestione dell'invio di richieste al server.
 - graphicobjects: componente per la gestione degli oggetti grafici che vengono rappresentati nel mondo di gioco quali widget e immagini rappresentanti il mondo di gioco (esempio: edificio).

3.5.2 Classi

3.5.2.1 sgad::clienttier::controller::actions::Action



Figura 6: Diagramma della classe sgad::clienttier::controller::actions::Action

- **Descrizione:** interfaccia per le possibili azioni che possono essere eseguite sull'interfaccia grafica o sul modello dei dati.
- **Utilizzo:** viene utilizzata per fornire una firma di metodo standard per qualsiasi tipo di azione che è possibile eseguire. Le classi che realizzano l'interfaccia si occuperanno di ridefinire il metodo per eseguire la particolare azione. Permette di stabilire un punto di comunicazione tra la view e il model. Inoltre, facilita l'associazione di un'azione ad un elemento della view.

Per un maggior approfondimento, si vedano le sezioni Visualizzazione dei menu e Interazione con il server.

- **Sottoclassi**

- ActionListener
- BackToVillage
- BuildConstruction
- ChangePasswordAction
- CloseContextualMenu
- DeleteAccountAction
- DemolishBuilding
- DismissUnits

- EnableRightClick
- HarvestResources
- LoadAnotherUser
- LoadGeneralData
- LoadPersonalData
- Logout
- ReceiveStealResources
- ReloadPageAction
- RemoveGraphicObjectAction
- RequestAttackUser
- RequestBuildConstruction
- RequestChangePassword
- RequestDeleteAccount
- RequestDemolishBuilding
- RequestDismissUnits
- RequestHarvestResources
- RequestLoadGeneralData
- RequestLoadPersonalData
- RequestStoleResources
- RequestTrainUnits
- RequestUpgradeBuilding
- SetAllVoidFilter
- SetBuildModeFilter
- ShowAccountDeletedMenu
- ShowAccountManagerMenu
- ShowAnotherUserMenu
- ShowAttackMenu
- ShowAttackResultMenu
- ShowBuildConstructionMenu
- ShowBuildingContextualMenu
- ShowDemolishMenu
- ShowDismissUnitMenu
- ShowInteractionMenu
- ShowOperationFailureMenu
- ShowPasswordChangedMenu
- ShowResourceMenu
- ShowTileContextualMenu

- ShowTrainUnitMenu
 - ShowUnitSelectionMenu
 - ShowUpgradeMenu
 - ShowUserListMenu
 - StealResources
 - TrainUnit
 - UpgradeBuilding
- Metodi
 - `+performAction() : void`
Esegue l'azione.

3.5.2.2 sgad::clienttier::controller::actions::ActionListener

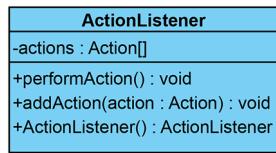


Figura 7: Diagramma della classe `sgad::clienttier::controller::actions::-ActionListener`

- **Descrizione:** classe per permettere di memorizzare una coda di azioni da eseguire in successione. Queste verranno invocate quando viene eseguito il metodo per eseguire l'azione su tale classe.
- **Utilizzo:** viene utilizzata per mantenere una lista di azioni. Ciò permette di comporre una sequenza di azioni senza dover definire una nuova classe che raggruppa tali azioni esplicitamente. Fornisce quindi la possibilità di associare un insieme di azioni ad un elemento dell'interfaccia grafica.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-actions : Action[]`
La collezione di azioni da eseguire.
- Metodi
 - `+performAction() : void`
Esegue la lista di azioni che è stata precedentemente memorizzata.

– `+addAction(action : Action) : void`

Inserisce una nuova azione nella lista delle azioni da eseguire. L'azione viene eseguita dopo tutte le azioni già inserite.

Argomenti:

* `action` : l'azione da inserire.

– `+ActionListener() : ActionListener`

Costruttore della classe.

3.5.2.3 sgad::clienttier::controller::actions::BackToVillage

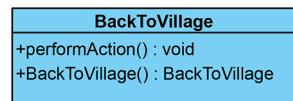


Figura 8: Diagramma della classe `sgad::clienttier::controller::actions::-BackToVillage`

- **Descrizione:** classe per permettere di ritornare al proprio villaggio nella mappa.
- **Utilizzo:** viene utilizzata per permettere all'utente di ritornare al proprio villaggio una volta che ha saccheggiato il villaggio di un altro utente.
- **Classi ereditate**
 - `Action`
- **Metodi**
 - `+performAction() : void`
Ripristina le viste e il modello dei dati relativi all'account dell'utente giocatore autenticato.
 - `+BackToVillage() : BackToVillage`
Costruttore della classe.

3.5.2.4 sgad::clienttier::controller::actions::BuildConstruction

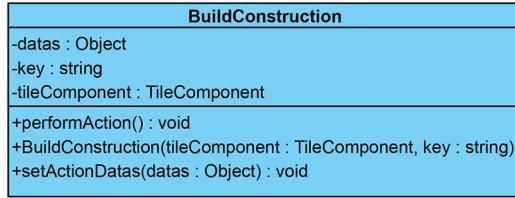


Figura 9: Diagramma della classe `sgad::clienttier::controller::actions::-BuildConstruction`

- **Descrizione:** classe per la gestione della costruzione di un nuovo edificio nel villaggio dell’utente.

- **Utilizzo:** viene utilizzata per permettere all’utente di costruire un nuovo edificio nel villaggio.

- **Classi ereditate**
 - Action

- **Attributi**
 - `-datas : Object`
Dati relativi all’edificio da costruire.
 - `-key : string`
Tipo di edificio da rappresentare.
 - `-tileComponent : TileComponent`
La casella della mappa dove costruire l’edificio.

- **Metodi**
 - `+BuildConstruction(key : string, tileComponent : TileComponent) :- BuildConstruction`
Costruttore della classe.

Argomenti:

* `key` : tipo di edificio da rappresentare.

* `tileComponent` : la casella della mappa dove costruire l’edificio.

- `+performAction() : void`

Esegue la lista di azioni necessaria per costruire il nuovo edificio.

- `+setActionDatas(datas : Object) : void`

Imposta i dati relativi all’edificio da costruire.

Argomenti:

* `datas` : i dati da impostare.

3.5.2.5 sgad::clienttier::controller::actions::ChangePasswordAction

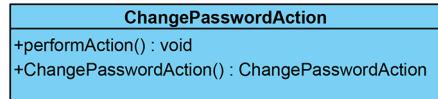


Figura 10: Diagramma della classe sgad::clienttier::controller::actions::- ChangePasswordAction

- **Descrizione:** classe per permettere di modificare la password di un account.
- **Utilizzo:** viene utilizzata per modificare la password associata all'account.
- **Classi ereditate**
 - Action
- **Metodi**
 - +performAction() : void
Esegue la lista di azioni necessarie per la modifica della password.
 - +ChangePasswordAction() : ChangePasswordAction
Costruttore della classe.

3.5.2.6 sgad::clienttier::controller::actions::CloseContextualMenu

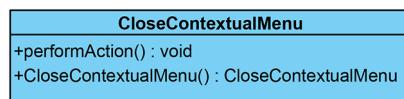


Figura 11: Diagramma della classe sgad::clienttier::controller::actions::- CloseContextualMenu

- **Descrizione:** classe per la gestione della chiusura del menu contestuale visualizzato alla pressione del pulsante destro del mouse.
- **Utilizzo:** viene utilizzata per gestire la chiusura del menu contestuale correntemente visualizzato.
- **Classi ereditate**
 - Action
- **Metodi**

- `+CloseContextualMenu() : CloseContextualMenu`
Costruttore della classe.

- `+performAction() : void`
Chiude il menu contestuale attualmente visualizzato.

3.5.2.7 sgad::clienttier::controller::actions::DeleteAccountAction

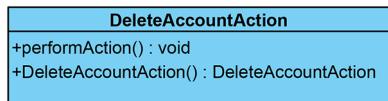


Figura 12: Diagramma della classe `sgad::clienttier::controller::actions::DeleteAccountAction`

- **Descrizione:** la classe viene utilizzata per l'eliminazione di un account utente.
- **Utilizzo:** viene utilizzata per eliminare i dati dell'account.
- **Classi ereditate**
 - `Action`
- **Metodi**
 - `+DeleteAccountAction() : DeleteAccountAction`
Costruttore della classe.

 - `+performAction() : void`
Esegue la lista di azioni per eliminare l'account.

3.5.2.8 sgad::clienttier::controller::actions::DemolishBuilding

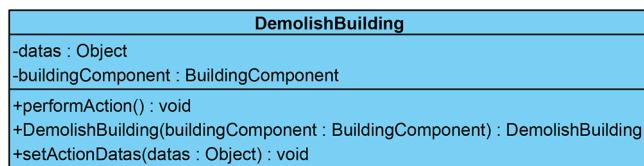


Figura 13: Diagramma della classe `sgad::clienttier::controller::actions::DemolishBuilding`

- **Descrizione:** classe per la gestione della demolizione di un edificio già presente nel villaggio.
- **Utilizzo:** viene utilizzata per permettere all'utente di demolire un edificio che ha precedentemente costruito.
- **Classi ereditate**
 - Action
- **Attributi**
 - -buildingComponent : BuildingComponent
L'edificio da demolire.
 - -datas : Object
Dati relativi all'edificio da demolire.
- **Metodi**
 - +DemolishBuilding(buildingComponent : BuildingComponent) : DemolishBuilding
Costruttore della classe.

Argomenti:

 - * buildingComponent : l'edificio da demolire.
 - +performAction() : void
Demolisce l'edificio indicato.
 - +setActionDatas(datas : Object) : void
Imposta i dati relativi all'edificio da demolire.

Argomenti:

 - * datas : i dati da impostare.

3.5.2.9 sgad::clienttier::controller::actions::DismissUnits

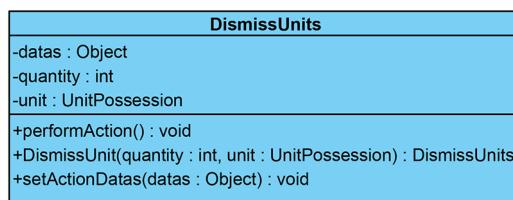


Figura 14: Diagramma della classe sgad::clienttier::controller::actions::-DismissUnits

- **Descrizione:** classe per la gestione del congedo di unità arruolate in precedenza dall'utente.

- **Utilizzo:** viene utilizzata per permettere all'utente di congedare un determinato numero di unità di un certo tipo.

- **Classi ereditate**

- **Action**

- **Attributi**

- **-datas : Object**

Dati da impostare per eseguire l'azione.

- **-quantity : int**

La quantità di unità da congedare.

- **-unit : UnitPossession**

Il tipo di unità da congedare.

- **Metodi**

- **+DismissUnit(quantity : int, unit : Unit) : DismissUnit**
Costruttore della classe.

Argomenti:

- * **quantity** : la quantità di unità da congedare.

- * **unit** : il tipo di unità che di cui è stato richiesto il congedo.

- **+performAction() : void**

Rimuove le unità da dimettere.

- **+setActionDatas(datas : Object) : void**

Imposta i dati necessari per eseguire l'azione.

Argomenti:

- * **datas** : i dati da impostare.

3.5.2.10 sgad::clienttier::controller::actions::EnableRightClick

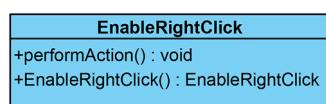


Figura 15: Diagramma della classe `sgad::clienttier::controller::actions::-EnableRightClick`

- **Descrizione:** classe per permettere di abilitare la gestione dell'evento click destro del mouse.
- **Utilizzo:** viene utilizzata dai vari menu per riabilitare la gestione dell'evento rightclick. Ciò serve per evitare di continuare ad interagire con il mondo di gioco senza aver chiuso il menu visualizzato.
- **Classi ereditate**
 - Action
- **Metodi**
 - `+EnableRightClick() : EnableRightClick`
Costruttore della classe.
 - `+performAction() : void`
Abilita il click destro del mouse.

3.5.2.11 sgad::clienttier::controller::actions::HarvestResources

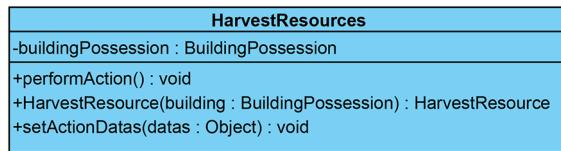


Figura 16: Diagramma della classe `sgad::clienttier::controller::actions::-HarvestResources`

- **Descrizione:** classe per la gestione della raccolta di risorse dagli edifici di produzione costruiti nel villaggio.
- **Utilizzo:** viene utilizzata per permettere all'utente di raccogliere le risorse prodotte dagli edifici di produzione.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio da cui raccogliere risorse.
- **Metodi**

- `+HarvestResource(building : BuildingPossession) : HarvestResource`
Costruttore della classe.

Argomenti:

- * `building` : l'edificio dal quale è stato richiesta la raccolta di risorse.
- `+performAction() : void`
Esegue le istruzioni necessarie per raccogliere le risorse.
- `+setActionDatas(datas : Object) : void`
Imposta i dati relativi alle risorse da raccogliere.

Argomenti:

- * `datas` : i dati da impostare.

3.5.2.12 sgad::clienttier::controller::actions::LoadAnotherUser

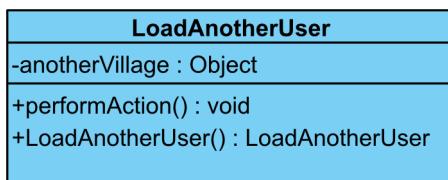


Figura 17: Diagramma della classe `sgad::clienttier::controller::actions::LoadAnotherUser`

- **Descrizione:** classe per il caricamento di un altro utente.
- **Utilizzo:** viene utilizzata per caricare i dati di un altro utente. Si occupa quindi di salvare i dati dell'utente attualmente visualizzato per poterli ripristinare una volta che si ritorna al proprio villaggio.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-anotherVillage : Object`
Il villaggio del giocatore attaccato.
- **Metodi**
 - `+LoadAnotherUser() : LoadAnotherUser`
Costruttore della classe.

- `+performAction() : void`

Esegue le istruzioni necessarie per il caricamento di un utente differente da quello attuale.

3.5.2.13 sgad::clienttier::controller::actions::LoadGeneralData

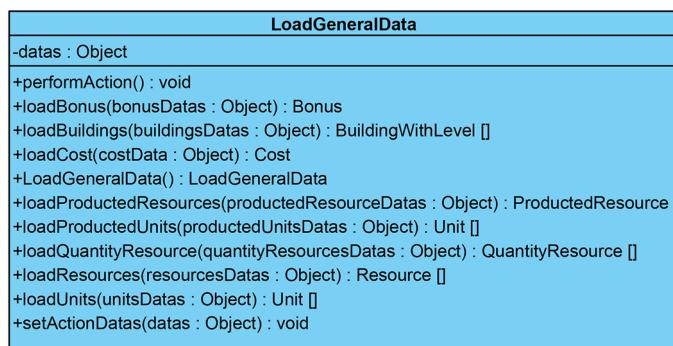


Figura 18: Diagramma della classe `sgad::clienttier::controller::actions::LoadGeneralData`

- **Descrizione:** classe per la gestione del caricamento dei dati che non dipendono dai particolari dati dell’utente.
- **Utilizzo:** viene utilizzata per caricare i dati globali all’avvio di una nuova sessione. Il caricamento di tali dati è quindi unico. Viene effettuato ad ogni accesso al mondo di gioco.
- **Classi ereditate**
 - `Action`
- **Attributi**
 - `-datas : Object`
I dati dal quale caricare il modello dei dati.
- **Metodi**
 - `+LoadGeneralData() : LoadGeneralData`
Costruttore della classe.
 - `+performAction() : void`
Carica il modello dei dati in base all’oggetto contenente i dati memorizzato.
 - `+loadResources(resourcesDatas : Object) : Array<Resource>`
Carica le risorse indicate dal frammento dei dati.

Argomenti:

* `resourcesDatas` : i dati contenenti le risorse.

- `+loadUnits(unitsDatas : Object) : Array<Unit>`
Carica le unità indicate nel frammento dei dati.

Argomenti:

* `unitsDatas` : i dati contenenti le unità.

- `+loadCost(costData : Object) : Cost`
Carico il costo indicato nel frammento dei dati.

Argomenti:

* `costData` : i dati contenenti il costo.

- `+loadQuantityResource(quantityResourcesDatas : Object) : Array<QuantityResource>`
Carica le quantità di risorse indicate nel frammento dei dati.

Argomenti:

* `quantityResourcesDatas` : i dati contenenti le quantità di risorse.

- `+loadBuildings(buildingsDatas : Object) : Array<BuildingWithLevel>`
Carica gli edifici indicati nel frammento dei dati.

Argomenti:

* `buildingsDatas` : i dati contenenti gli edifici.

- `+loadProducedResources(productedResourceDatas : Object) : ProducedResource`
Carica le risorse producibili nel frammento dei dati.

Argomenti:

* `producedResourceDatas` : i dati contenenti le risorse producibili.

- `+loadProducedUnits(productedUnitsDatas : Object) : Array<Unit>`
Carica le unità producibili nel frammento dei dati.

Argomenti:

* `producedUnitsDatas` : i dati contenenti le unità producibili.

- `+loadBonus(bonusDatas : Object) : Bonus`
Carica i bonus nel frammento dei dati.

Argomenti:

- * **bonusDatas** : i dati contenenti i bonus.
- **+setActionDatas(datas : Object) : void**
Imposta i dati da cui verrà caricare il modello.

Argomenti:

- * **datas** : i dati da cui caricare il modello.

3.5.2.14 sgad::clienttier::controller::actions::LoadPersonalData

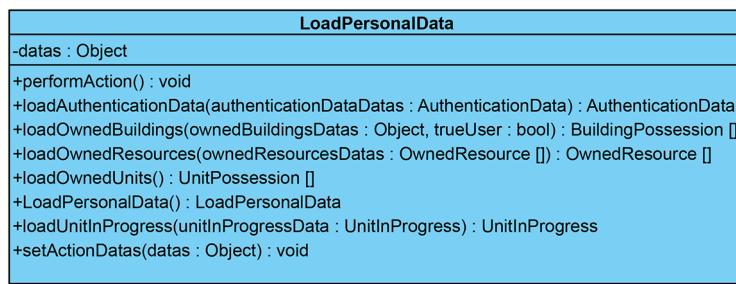


Figura 19: Diagramma della classe `sgad::clienttier::controller::actions::-LoadPersonalData`

- **Descrizione:** classe per la gestione del caricamento dei dati personali dell’utente.
- **Utilizzo:** viene utilizzata per caricare i dati dell’utente all’avvio di una nuova sessione. Il caricamento viene effettuato ad ogni accesso al mondo di gioco. Si occupa inoltre di creare gli elementi di interfaccia grafica che rispecchieranno i dati del mondo di gioco.
- **Classi ereditate**
 - Action
- **Attributi**
 - **-datas : Object**
I dati dal quale caricare il modello dei dati.
- **Metodi**
 - **+LoadPersonalData() : LoadPersonalData**
Costruttore della classe.

– `+performAction() : void`
Carica i dati personali dell’utente.

– `+loadAuthenticationData(authenticationDataDatas : AuthenticationData)`
`: AuthenticationData`
Ritorna i dati di accesso dell’utente.

Argomenti:

* `authenticationDataDatas` : i dati di accesso dell’utente.

– `+loadOwnedResources(ownedResourcesDatas : Array<OwnedResource>)` :-
`Array<OwnedResource>`
Carica i dati relativi alle risorse possedute dall’utente.

Argomenti:

* `ownedResourcesDatas` : le risorse possedute dall’utente.

– `+loadOwnedUnits() : Array<UnitPossession>`
Ritorna le unità possedute dall’utente.

– `+loadOwnedBuildings(ownedBuildingsDatas : Object, trueUser : bool)`
`: Array<BuildingPossession>`
Ritorna gli edifici posseduti dall’utente.

Argomenti:

* `ownedBuildingsDatas` : gli edifici posseduti dall’utente.

* `trueUser` : indica se i dati si riferiscono a questo utente o al destinatario dell’attacco.

– `+loadUnitInProgress(unitInProgressData : UnitInProgress) : UnitInProgress`
Carica le unità che sono in fase di creazione.

Argomenti:

* `unitInProgressData` : le unità che sono in fase di creazione.

– `+setActionDatas(datas : Object) : void`
Imposta i dati personali dell’utente.

Argomenti:

* `datas` : i dati personali dell’utente.

3.5.2.15 sgad::clienttier::controller::actions::Logout

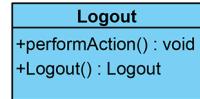


Figura 20: Diagramma della classe sgad::clienttier::controller::actions::Logout

- **Descrizione:** classe per la gestione del logout dell'utente.
- **Utilizzo:** viene utilizzata per eseguire il logout una volta che l'utente ha terminato di interagire con il mondo di gioco.
- **Classi ereditate**
 - Action
- **Metodi**
 - **+Logout()** : Logout
Costruttore della classe.
 - **+performAction()** : void
Esegue le istruzioni necessarie per effettuare il logout.

3.5.2.16 sgad::clienttier::controller::actions::ReceiveStealResources

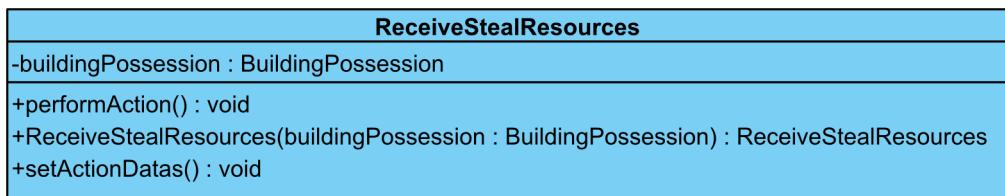


Figura 21: Diagramma della classe sgad::clienttier::controller::actions::-ReceiveStealResources

- **Descrizione:** classe per la ricezione delle risorse rubate
- **Utilizzo:** gestisce le risorse rubate
- **Classi ereditate**
 - Action
- **Attributi**

- `-buildingPossession : BuildingPossession`
L'edificio dal quale devono essere rimosse le risorse prodotte.
- Metodi
 - `+performAction() : void`
Esegue l'azione per rimuovere le risorse rubate.
 - `+setActionDatas() : void`
Imposta i dati necessari ad eseguire l'azione.
 - `+ReceiveStealResources(buildingPossession : BuildingPossession) :- ReceiveStealResources`
Costruttore della classe.

Argomenti:

* `buildingPossession` : l'edificio dal quale sono state rubate le risorse.

3.5.2.17 sgad::clienttier::controller::actions::ReloadPageAction

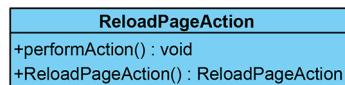


Figura 22: Diagramma della classe `sgad::clienttier::controller::actions::- ReloadPageAction`

- **Descrizione:** azione per il ricaricamento della pagina.
- **Utilizzo:** viene utilizzata per richiedere il ricaricamento di una pagina in seguito ad un fallimento di una richiesta al server.
- **Classi ereditate**
 - Action
- Metodi
 - `+ReloadPageAction() : ReloadPageAction`
Costruttore della classe.
 - `+performAction() : void`
Ricarica la pagina.

3.5.2.18 sgad::clienttier::controller::actions::RemoveGraphicObjectAction

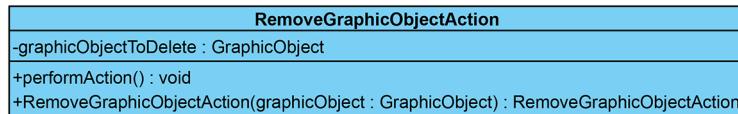


Figura 23: Diagramma della classe `sgad::clienttier::controller::actions::RemoveGraphicObjectAction`

- **Descrizione:** classe per la rimozione di un oggetto grafico visualizzato nell'area di gioco.
- **Utilizzo:** viene utilizzata per rimuovere un oggetto grafico che rappresenta un elemento dell'interfaccia grafica. Tale azione non conosce il tipo effettivo dell'oggetto che andrà a rimuovere. Tuttavia richiede che l'oggetto da rimuovere sia un sottotipo di `GraphicObject`.
- **Classi ereditate**
 - `Action`
- **Attributi**
 - `-graphicObjectToDelete : GraphicObject`
Oggetto grafico da rimuovere.
- **Metodi**
 - `+RemoveGraphicObjectAction(graphicObject : GraphicObject) : RemoveGraphicObjectAction`
Costruttore della classe.

Argomenti:

 - * `graphicObject` : l'oggetto grafico da rimuovere
 - `+performAction() : void`
Rimuove l'oggetto grafico indicato.

3.5.2.19 sgad::clienttier::controller::actions::RequestAttackUser

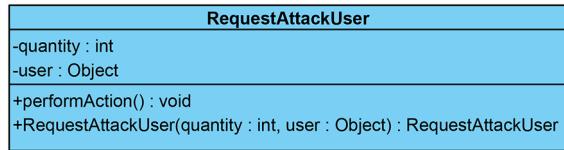


Figura 24: Diagramma della classe `sgad::clienttier::controller::actions::-RequestAttackUser`

- **Descrizione:** classe per la gestione della richiesta di attacco di un utente desiderato.
- **Utilizzo:** viene utilizzata per richiedere al server di attaccare l'utente desiderato.
- **Classi ereditate**

– Action

- **Attributi**

– `-quantity : int`

La quantità di unità da mandare all'attacco.

– `-user : Object`

L'utente da attaccare.

- **Metodi**

– `+RequestAttackUser (quantity : int[], user : Object) : RequestAttackUser`

Costruttore della classe.

Argomenti:

* `quantity` : la quantità di truppe da mandare all'attacco. Deve essere un array associativo dove la chiave è quella del tipo di unità.

* `user` : l'utente da attaccare.

– `+performAction() : void`

Invia la richiesta di attacco dell'utente al server.

3.5.2.20 sgad::clienttier::controller::actions::RequestBuildConstruction

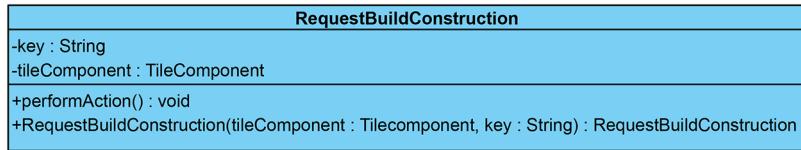


Figura 25: Diagramma della classe `sgad::clienttier::controller::actions::RequestBuildConstruction`

- **Descrizione:** classe per la richiesta di costruzione di un edificio.
- **Utilizzo:** viene utilizzata per inviare al server una richiesta per la costruzione dell'edificio desiderato.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-key : String`
Tipo di edificio da rappresentare.
 - `-tileComponent : TileComponent`
La casella della mappa dove costruire l'edificio.
- **Metodi**
 - `+RequestBuildConstruction(key : String, tileComponent : TileComponent)`
: `RequestBuildConstruction`
Costruttore della classe.
 - Argomenti:**
 - * `key` : la chiave dell'edificio richiesto dalla costruzione.
 - * `tileComponent` : la casella in cui è stata richiesta la costruzione di un edificio.
 - `+performAction() : void`
Gestisce le operazioni per la richiesta di costruzione dell'edificio.

3.5.2.21 sgad::clienttier::controller::actions::RequestChangePassword

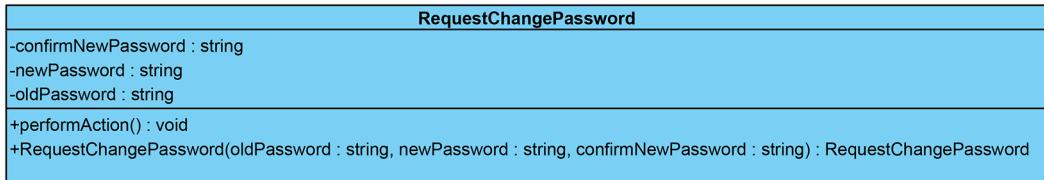


Figura 26: Diagramma della classe `sgad::clienttier::controller::actions::-RequestChangePassword`

- **Descrizione:** classe per la gestione della richiesta di cambiamento password.
- **Utilizzo:** viene utilizzata per richiedere al server la modifica della password dell'utente.
- **Classi ereditate**
 - Action

- **Attributi**

- `-confirmNewPassword : string`
La vecchia password.
- `-newPassword : string`
La nuova password.
- `-oldPassword : string`
La nuova password.

- **Metodi**

- `+RequestChangePassword(confirmNewPassword : string, newPassword : string, oldPassword : string) : RequestChangePassword`
Costruttore della classe.

Argomenti:

- * `confirmNewPassword` : la nuova password.
- * `newPassword` : la nuova password.
- * `oldPassword` : la vecchia password.
- `+performAction() : void`
Invia la richiesta di modifica della password.

3.5.2.22 sgad::clienttier::controller::actions::RequestDeleteAccount



Figura 27: Diagramma della classe `sgad::clienttier::controller::actions::RequestDeleteAccount`

- **Descrizione:** classe per la gestione della richiesta di eliminazione dell'account.
- **Utilizzo:** viene utilizzata per richiedere al server di eliminare l'account dell'utente.
- **Classi ereditate**
 - Action
- **Metodi**
 - `+performAction() : void`
Invia la richiesta di di eliminazione dell'account.
 - `+RequestDeleteAccount() : RequestDeleteAccount`
Costruttore della classe.

3.5.2.23 sgad::clienttier::controller::actions::RequestDemolishBuilding

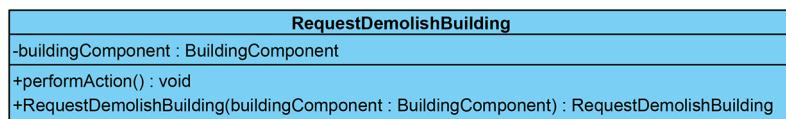


Figura 28: Diagramma della classe `sgad::clienttier::controller::actions::RequestDemolishBuilding`

- **Descrizione:** classe per la richiesta di demolizione di un edificio.
- **Utilizzo:** viene utilizzata per inviare al server una richiesta per demolire l'edificio desiderato.
- **Classi ereditate**
 - Action
- **Attributi**

- `-buildingComponent : BuildingComponent`
L’edificio da demolire.

- **Metodi**

- `+RequestDemolishBuilding (buildingComponent : BuildingComponent) :- RequestDemolishBuilding`
Costruttore della classe.

Argomenti:

- * `buildingComponent` : l’edificio da demolire.
- `+performAction() : void`
Gestisce la richiesta di demolizione dell’edificio.

3.5.2.24 sgad::clienttier::controller::actions::RequestDismissUnits

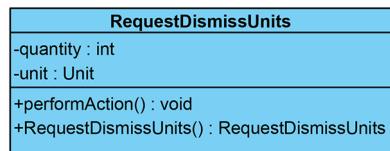


Figura 29: Diagramma della classe `sgad::clienttier::controller::actions::RequestDismissUnits`

- **Descrizione:** classe per la richiesta di dimissione di unità possedute.
- **Utilizzo:** viene utilizzata per inviare una richiesta al server per congedare le unità desiderate.
- **Classi ereditate**
 - `Action`
- **Attributi**
 - `-quantity : int`
La quantità di unità da dimettere.
 - `-unit : Unit`
Il tipo di unità da dimettere.
- **Metodi**
 - `+RequestDismissUnits() : RequestDismissUnits`
Costruttore della classe.

- `+performAction() : void`
Invia la richiesta di caricamento dei dati al server.

3.5.2.25 sgad::clienttier::controller::actions::RequestHarvestResources



Figura 30: Diagramma della classe `sgad::clienttier::controller::actions::RequestHarvestResources`

- **Descrizione:** classe per la richiesta di raccolta di risorse.
- **Utilizzo:** viene utilizzata per inviare al server una richiesta per raccogliere le risorse dell'edificio selezionato.
- **Classi ereditate**

- Action

- **Attributi**

- `-buildingPossession : BuildingPossession`
L'edificio da cui raccogliere.

- **Metodi**

- `+RequestHarvestResources(buildingPossession : BuildingPossession) :- RequestHarvestResources`
Costruttore della classe.

Argomenti:

- * `buildingPossession` : l'edificio da cui raccogliere.
- `+performAction() : void`
Gestisce la richiesta di demolizione dell'edificio.

3.5.2.26 sgad::clienttier::controller::actions::RequestLoadGeneralData

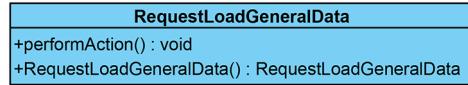


Figura 31: Diagramma della classe `sgad::clienttier::controller::actions::-RequestLoadGeneralData`

- **Descrizione:** classe per la gestione della richiesta per il caricamento dei dati globali.
- **Utilizzo:** viene utilizzata per generare la richiesta che avvierà il caricamento dei dati globali. Viene quindi imposto, come azione al successo della richiesta per il caricamento effettivo dei dati globali, un oggetto di tipo `LoadGeneralData`. Viene poi eseguita la richiesta per il caricamento dei dati personali dell'utente.
- **Classi ereditate**
 - `Action`
- **Metodi**
 - `+performAction() : void`
Invia la richiesta di caricamento dei dati al server.
 - `+RequestLoadGeneralData() : RequestLoadGeneralData`
Costruttore della classe.

3.5.2.27 `sgad::clienttier::controller::actions::RequestLoadPersonalData`

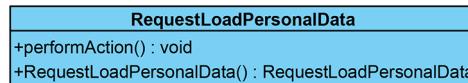


Figura 32: Diagramma della classe `sgad::clienttier::controller::actions::-RequestLoadPersonalData`

- **Descrizione:** classe per la gestione della richiesta per il caricamento dei dati dell'utente.
- **Utilizzo:** viene utilizzata per generare la richiesta che avvierà il caricamento dei dati dell'utente. Viene quindi imposto, come azione al successo della richiesta per il caricamento effettivo dei dati personali, un oggetto di tipo `LoadPersonalData`
- **Classi ereditate**
 - `Action`
- **Metodi**

- `+performAction() : void`
Invia la richiesta di caricamento dei dati al server.
- `+RequestLoadPersonalData() : RequestLoadPersonalData`
Costruttore della classe.

3.5.2.28 sgad::clienttier::controller::actions::RequestStoleResources

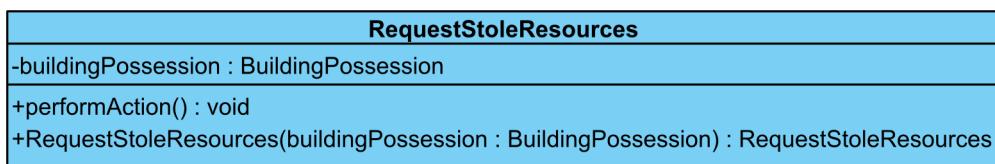


Figura 33: Diagramma della classe `sgad::clienttier::controller::actions::RequestStoleResources`

- **Descrizione:** classe per la richiesta di furto di risorse
 - **Utilizzo:** gestisce l'azione del furto di risorse
 - **Classi ereditate**
 - Action
 - **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio da cui rubare le risorse
 - **Metodi**
 - `+performAction() : void`
Invia la richiesta di furto al server.
 - `+RequestStoleResources(buildingPossession : BuildingPossession) :- RequestStoleResources`
Costruttore della classe.
- Argomenti:**
- * `buildingPossession` : l'edificio dal quale si vogliono rubare le risorse.

3.5.2.29 sgad::clienttier::controller::actions::RequestTrainUnits

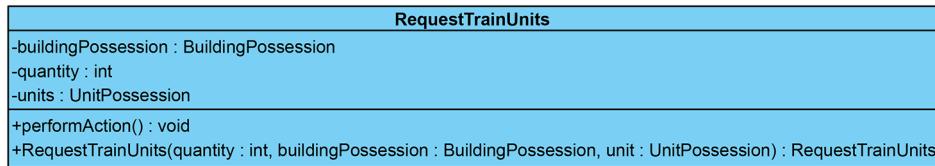


Figura 34: Diagramma della classe `sgad::clienttier::controller::actions::RequestTrainUnits`

- **Descrizione:** classe per la gestione della richiesta per il caricamento dei dati globali.

- **Utilizzo:** viene utilizzata per inviare al server la richiesta di addestramento delle unità selezionate nell'edificio desiderato.

- **Classi ereditate**
 - Action

- **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio a cui aggiungere le unità.
 - `-quantity : int`
La quantità di unità da aggiungere.
 - `-units : UnitPossession`
Il tipo di unità da aggiungere.

- **Metodi**
 - `+performAction() : void`
Invia la richiesta di caricamento dei dati al server.
 - `+RequestTrainUnits (buildingPossession : BuildingPossession, quantity : int, unit : UnitPossession) : RequestTrainUnits`
Costruttore della classe.

Argomenti:

- * `buildingPossession` : l'edificio a cui aggiungere le unità.
- * `quantity` : la quantità di unità da aggiungere.
- * `unit` : il tipo di unità da aggiungere.

3.5.2.30 sgad::clienttier::controller::actions::RequestUpgradeBuilding

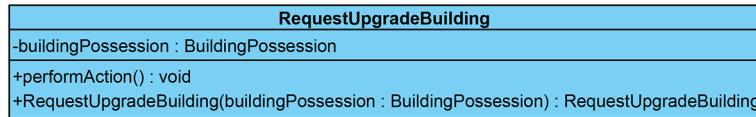


Figura 35: Diagramma della classe `sgad::clienttier::controller::actions::RequestUpgradeBuilding`

- **Descrizione:** classe per la gestione della richiesta di miglioramento di un edificio.
- **Utilizzo:** viene utilizzata per inviare al server la richiesta di miglioramento dell'edificio desiderato.
- **Classi ereditate**
 - `Action`
- **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio da migliorare.
- **Metodi**
 - `+RequestUpgradeBuilding(buildingPossession : BuildingPossession) :- RequestUpgradeBuilding`
Costruttore della classe.

Argomenti:

 - * `buildingPossession` : l'edificio da migliorare.
 - `+performAction() : void`
Invia la richiesta di miglioramento dell'edificio al server.

3.5.2.31 sgad::clienttier::controller::actions::SetAllVoidFilter



Figura 36: Diagramma della classe `sgad::clienttier::controller::actions::SetAllVoidFilter`

- **Descrizione:** classe per la gestione di un filtro standard.
- **Utilizzo:** viene utilizzata per permettere all'utente di ritornare ad una visualizzazione normale. Viene quindi impostato il filtro standard agli elementi di interfaccia grafica del villaggio dell'utente.
- **Classi ereditate**
 - Action
- **Metodi**
 - `+performAction() : void`
Gestisce il filtro.
 - `+SetAllVoidFilter() : SetAllVoidFilter`
Costruttore della classe.

3.5.2.32 sgad::clienttier::controller::actions::SetBuildModeFilter

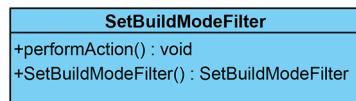


Figura 37: Diagramma della classe `sgad::clienttier::controller::actions::SetBuildModeFilter`

- **Descrizione:** classe per la gestione dei filtri per la modalità costruzione.
- **Utilizzo:** viene utilizzata per impostare come visualizzazione la modalità di costruzione.
- **Classi ereditate**
 - Action
- **Metodi**
 - `+performAction() : void`
Imposta il filtro.
 - `+SetBuildModeFilter() : SetBuildModeFilter`
Costruttore della classe.

3.5.2.33 sgad::clienttier::controller::actions::ShowAccountDeletedMenu

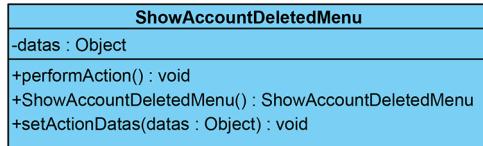


Figura 38: Diagramma della classe `sgad::clienttier::controller::actions::ShowAccountDeletedMenu`

- **Descrizione:** classe per la visualizzazione di un menu indicante l'esito dell'operazione di eliminazione dell'account.
- **Utilizzo:** visualizza un menu per indicare l'esito della cancellazione dell'account dell'utente.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-datas : Object`
I dati da impostare per la visualizzazione del menu.
- **Metodi**
 - `+performAction() : void`
Si occupa di visualizzare il menu all'utente.
 - `+ShowAccountDeletedMenu() : ShowAccountDeletedMenu`
Costruttore della classe.
 - `+setActionDatas(datas : Object) : void`
Imposta i dati per la visualizzazione del menu all'utente.

Argomenti:

* `datas` : i dati da impostare.

3.5.2.34 sgad::clienttier::controller::actions::ShowAccountManagerMenu

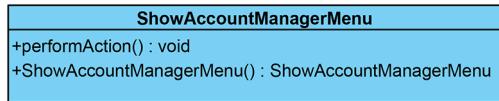


Figura 39: Diagramma della classe `sgad::clienttier::controller::actions::ShowAccountManagerMenu`

- **Descrizione:** classe per la creazione di un menu per la gestione dell'account.
- **Utilizzo:** viene utilizzata per visualizzare il menu per modificare i dati dell'account utente.
- **Classi ereditate**
 - Action
- **Metodi**
 - `+performAction() : void`
Visualizza il menu relativo all'account.
 - `+ShowAccountManagerMenu() : ShowAccountManagerMenu`
Costruttore della classe.

3.5.2.35 sgad::clienttier::controller::actions::ShowAnotherUserMenu

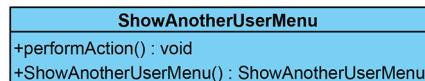


Figura 40: Diagramma della classe `sgad::clienttier::controller::actions::ShowAnotherUserMenu`

- **Descrizione:** classe per la visualizzazione del menu per ritornare al proprio villaggio una volta che si è nel villaggio di un altro utente.
- **Utilizzo:** viene utilizzata per visualizzare il menu una volta che si sta visualizzando il villaggio di un altro utente.
- **Classi ereditate**
 - Action
- **Metodi**

- `+performAction() : void`
Visualizza il menu per ritornare al proprio villaggio una volta che si è nel villaggio di un altro utente.
- `+ShowAnotherUserMenu() : ShowAnotherUserMenu`
Costruttore della classe.

3.5.2.36 sgad::clienttier::controller::actions::ShowAttackMenu

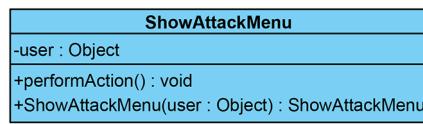


Figura 41: Diagramma della classe `sgad::clienttier::controller::actions::ShowAttackMenu`

- **Descrizione:** classe per la visualizzazione del menu per l'attacco.
- **Utilizzo:** viene utilizzata per visualizzare il menu per richiedere quali truppe si desidera inviare per l'attacco.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-user : Object`
L'utente a cui visualizzare il menu.
- **Metodi**
 - `+ShowAttackMenu(user : Object) : ShowAttackMenu`
Costruttore della classe.

Argomenti:

 - * `user` : l'utente a cui visualizzare il menu.
 - `+performAction() : void`
Si occupa di mostrare all'utente il menu.

3.5.2.37 sgad::clienttier::controller::actions::ShowAttackResultMenu

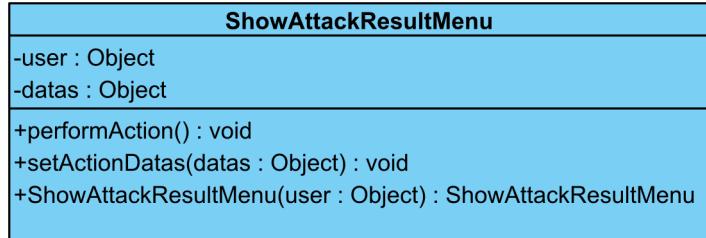


Figura 42: Diagramma della classe sgad::clienttier::controller::actions::-ShowAttackResultMenu

- **Descrizione:** classe per la visualizzazione del menu che visualizza il risultato dell'attacco.
- **Utilizzo:** viene utilizzata per visualizzare l'esito dello scontro con l'utente attaccato una volta inviato l'attacco.
- **Classi ereditate**
 - Action
- **Attributi**
 - -datas : Object
Dati sulle unità riguardo il risultato della battaglia.
 - -user : Object
L'utente a cui visualizzare il risultato dell'attacco.
- **Metodi**
 - +ShowAttackResultMenu (user : Object) : ShowAttackResultMenu
Costruttore della classe.

Argomenti:

* **user** : l'utente a cui visualizzare il risultato dell'attacco.

– +performAction() : void
Si occupa di mostrare all'utente il menu.

– +setActionDatas(datas : Object) : void
Imposta i dati sulle unità rispetto alla battaglia.

Argomenti:

* **datas** : i dati sulle unità rispetto alla battaglia.

3.5.2.38 sgad::clienttier::controller::actions::ShowBuildConstructionMenu

ShowBuildConstructionMenu
-tileComponent : TileComponent
-key : string
+performAction() : void
+ShowBuildConstructionMenu(key : string, tileComponent : TileComponent) : ShowBuildConstructionMenu

Figura 43: Diagramma della classe `sgad::clienttier::controller::actions::ShowBuildConstructionMenu`

- **Descrizione:** classe per la richiesta della conferma della costruzione.
- **Utilizzo:** viene utilizzata per richiedere all'utente la conferma della costruzione.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-key : String`
Tipo di edificio da costruire.
 - `-tileComponent : TileComponent`
La casella della mappa dove costruire l'edificio.

- **Metodi**

- `+ShowBuildConstructionMenu(key : String, tileComponent : TileComponent) : ShowBuildConstructionMenu`
Costruttore della classe.

Argomenti:

- * `key` : tipo di edificio da costruire.
- * `tileComponent` : la casella della mappa dove costruire l'edificio.
- `+performAction() : void`
Si occupa di mostrare all'utente il menu.

3.5.2.39 sgad::clienttier::controller::actions::ShowBuildingContextualMenu

ShowBuildingContextualMenu
-buildingComponent : BuildingComponent
+performAction() : void
+ShowBuildingContextualMenu(buildingComponent : BuildingComponent) : ShowBuildingContextualMenu

Figura 44: Diagramma della classe `sgad::clienttier::controller::actions::-ShowBuildingContextualMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu contestuale per le possibili azioni che si possono eseguire su un edificio.
- **Utilizzo:** viene utilizzata per permettere all'utente di visualizzare un menu contestuale per le possibili azioni che si possono eseguire su un edificio.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-buildingComponent : BuildingComponent`
L'edificio di cui visualizzare il menu contestuale.
- **Metodi**
 - `+ShowBuildingContextualMenu(buildingComponent : BuildingComponent)`
: `ShowBuildingContextualMenu`
Costruttore della classe.

Argomenti:

- * `buildingComponent` : l'edificio di cui visualizzare il menu contestuale.
- `+performAction() : void`
Si occupa di mostrare all'utente il menu.

3.5.2.40 sgad::clienttier::controller::actions::ShowDemolishMenu

ShowDemolishMenu
-buildingComponent : BuildingComponent
+performAction() : void
+ShowDemolishMenu(buildingComponent : BuildingComponent) : ShowDemolishMenu

Figura 45: Diagramma della classe `sgad::clienttier::controller::actions::-ShowDemolishMenu`

- **Descrizione:** classe che visualizza il menu per la demolizione di un edificio.

- **Utilizzo:** viene utilizzata per richiedere la conferma dell'azione di demolizione dell'edificio richiesto.
- **Classi ereditate**
 - Action
- **Attributi**
 - -buildingComponent : BuildingComponent
L'edificio da demolire.
- **Metodi**
 - +ShowDemolishMenu(buildingComponent : BuildingComponent) : ShowDemolishMenu
Costruttore della classe.

Argomenti:

 - * buildingComponent : l'edificio da demolire.
 - +performAction() : void
Si occupa di mostrare all'utente il menu.

3.5.2.41 sgad::clienttier::controller::actions::ShowDismissUnitMenu

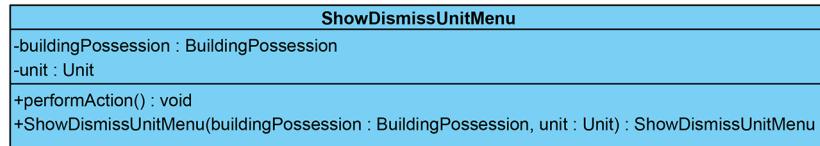


Figura 46: Diagramma della classe `sgad::clienttier::controller::actions::-ShowDismissUnitMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu per il congedo di unità.
- **Utilizzo:** viene utilizzata per richiedere la conferma dell'azione di congedo di unità.
- **Classi ereditate**
 - Action
- **Attributi**
 - -buildingPossession : BuildingPossession
L'edificio da cui richiedere la dimissione delle unità.

- `-unit : Unit`
Il tipo di unità da dimettere.

- **Metodi**

- `+ShowDismissUnitMenu(buildingPossession : BuildingPossession, unit : Unit) : ShowDismissUnitMenu`
Costruttore della classe.

Argomenti:

- * `buildingPossession` : l'edificio da cui richiedere la dimissione delle unità.
- * `unit` : il tipo di unità da dimettere.
- `+performAction() : void`
Visualizza il menu per richiedere la quantità da inserire e la conferma della dimissione.

3.5.2.42 sgad::clienttier::controller::actions::ShowInteractionMenu

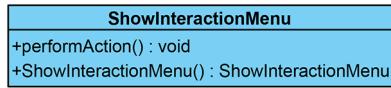


Figura 47: Diagramma della classe `sgad::clienttier::controller::actions::-ShowInteractionMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu per entrare in modalità costruzione o per la visualizzazione della lista degli utenti.
- **Utilizzo:** viene utilizzata per visualizzare un menu che l'utente può utilizzare per entrare in modalità costruzione o per visualizzare la lista degli utenti.
- **Classi ereditate**
 - `Action`
- **Metodi**
 - `+ShowInteractionMenu() : ShowInteractionMenu`
Costruttore della classe.
 - `+performAction() : void`
Si occupa di mostrare all'utente il menu.

3.5.2.43 sgad::clienttier::controller::actions::ShowOperationFailureMenu

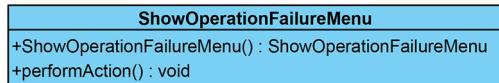


Figura 48: Diagramma della classe `sgad::clienttier::controller::actions::-ShowOperationFailureMenu`

- **Descrizione:** classe per la visualizzazione di un menu al fallimento di un'operazione.
- **Utilizzo:** viene utilizzata dai vari gestori delle richieste per notificare che la connessione con il server ha comportato il fallimento dell'operazione.
- **Classi ereditate**
 - Action
- **Metodi**
 - +ShowOperationFailureMenu() : ShowOperationFailureMenu
Costruttore della classe.
 - +performAction() : void
Visualizza un menu per indicare che l'operazione non è andata a buon fine.

3.5.2.44 sgad::clienttier::controller::actions::ShowPasswordChangedMenu

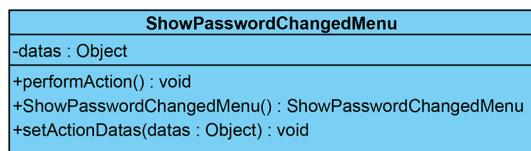


Figura 49: Diagramma della classe `sgad::clienttier::controller::actions::-ShowPasswordChangedMenu`

- **Descrizione:** classe per la visualizzazione di un menu di conferma del cambiamento della password.
- **Utilizzo:** viene utilizzata per visualizzare l'esito del cambiamento della password.
- **Classi ereditate**
 - Action

- **Attributi**

- `-datas : Object`

I dati da impostare per la visualizzazione del menu.

- **Metodi**

- `+ShowPasswordChangedMenu() : ShowPasswordChangedMenu`

Costruttore della classe.

- `+setActionDatas(datas : Object) : void`

Imposta i dati per la visualizzazione del menu all'utente.

Argomenti:

- * `datas` : i dati da impostare.

- `+performAction() : void`

Si occupa di visualizzare il menu all'utente.

3.5.2.45 sgad::clienttier::controller::actions::ShowResourceMenu

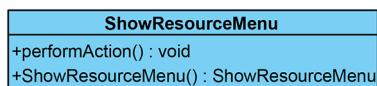


Figura 50: Diagramma della classe `sgad::clienttier::controller::actions::-ShowResourceMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu per selezionare le risorse da donare ad un altro utente.

- **Utilizzo:** viene utilizzata per visualizzare il menu per le risorse da selezionare per essere donate all'utente.

- **Classi ereditate**

- `Action`

- **Metodi**

- `+performAction() : void`

Visualizza il menu delle risorse da donare.

- `+ShowResourceMenu() : ShowResourceMenu`

Costruttore della classe.

3.5.2.46 sgad::clienttier::controller::actions::ShowTileContextualMenu



Figura 51: Diagramma della classe `sgad::clienttier::controller::actions::ShowTileContextualMenu`

- **Descrizione:** classe per la gestione della visualizzazione della barra riepilogativa delle risorse e delle unità possedute dall'utente.
- **Utilizzo:** viene utilizzata per visualizzare il menu riepilogativo delle risorse e delle unità possedute dall'utente.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-tileComponent : TileComponent`
La casella di cui visualizzare il menu contestuale.
- **Metodi**
 - `+performAction() : void`
Visualizza il menu contestuale relativo alla casella indicata.
 - `+ShowTileContextualMenu(tileComponent : TileComponent) : ShowTileContextualMenu`
Costruttore della classe.
- Argomenti:**
 - * `tileComponent` : la casella di cui visualizzare il menu contestuale.

3.5.2.47 sgad::clienttier::controller::actions::ShowTrainUnitMenu

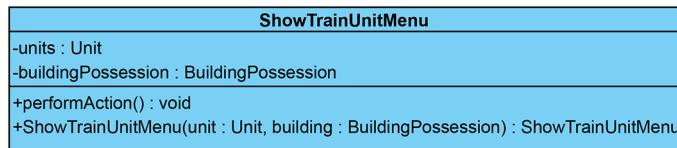


Figura 52: Diagramma della classe `sgad::clienttier::controller::actions::ShowTrainUnitMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu che permette all'utente di addestrare le unità.
- **Utilizzo:** viene utilizzata per la visualizzazione di un menu. Esso permette all'utente di inserire i dati necessari per eseguire poi l'azione di addestramento delle unità desiderate.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio di cui richiedere l'addestramento delle unità.
 - `-units : Unit`
Il tipo di unità da addestrare.
- **Metodi**
 - `+performAction() : void`
Visualizza il menu per richiedere la quantità da inserire e la conferma dell'addestramento.
 - `+ShowTrainUnitMenu(building : BuildingPossession, unit : Unit) :- ShowTrainUnitMenu`
Costruttore della classe.

Argomenti:

 - * `building` : l'edificio in cui deve essere addestrata l'unità.
 - * `unit` : il tipo di unità che deve essere addestrata.

3.5.2.48 sgad::clienttier::controller::actions::ShowUnitSelectionMenu

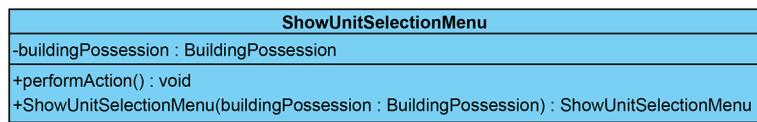


Figura 53: Diagramma della classe `sgad::clienttier::controller::actions::-ShowUnitSelectionMenu`

- **Descrizione:** classe per la visualizzazione di un menu per le unità addestrabili.
- **Utilizzo:** viene utilizzata per la visualizzazione di un menu per la selezione delle unità da mandare in battaglia.

- **Classi ereditate**

- Action

- **Attributi**

- -buildingPossession : BuildingPossession

L'edificio di cui richiedere l'addestramento delle unità.

- **Metodi**

- +performAction() : void

Visualizza il menu delle unità addestrabili.

- +ShowUnitSelectionMenu(buildingPossession : BuildingPossession) : ShowUnitSelectionMenu

Costruttore della classe.

Argomenti:

- * `buildingPossession` : l'edificio di cui richiedere l'addestramento delle unità.

3.5.2.49 sgad::clienttier::controller::actions::ShowUpgradeMenu

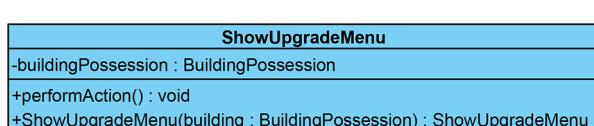


Figura 54: Diagramma della classe `sgad::clienttier::controller::actions::ShowUpgradeMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu che permette all'utente di confermare il miglioramento di un edificio.

- **Utilizzo:** viene utilizzata per la visualizzazione di un menu. Esso permette all'utente di confermare, o annullare, la richiesta di miglioramento dell'edificio desiderato.

- **Classi ereditate**

- Action

- **Attributi**

- -buildingPossession : BuildingPossession

L'edificio di cui richiedere la conferma di miglioramento.

- **Metodi**

- `+ShowUpgradeMenu(building : BuildingPossession) : ShowUpgradeMenu`
Costruttore della classe.

Argomenti:

- * `building` : l'edificio di cui richiedere la conferma di miglioramento.
- `+performAction() : void`
Visualizza il menu per richiedere la conferma del miglioramento.

3.5.2.50 sgad::clienttier::controller::actions::ShowUserListMenu

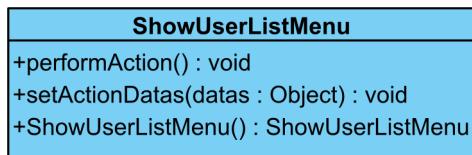


Figura 55: Diagramma della classe `sgad::clienttier::controller::actions::ShowUserListMenu`

- **Descrizione:** classe per la gestione della visualizzazione di un menu che permette all'utente di visualizzare la lista degli utenti con i quali può interagire.

- **Utilizzo:** viene utilizzata per la visualizzazione di un menu. Esso permette all'utente di selezionare un utente con il quale interagire.

- **Classi ereditate**

- Action

- **Attributi**

- `-datas : Object`
La lista degli altri utenti da poter attaccare.

- **Metodi**

- `+ShowUserListMenu() : ShowUserListMenu`
Costruttore della classe.

- `+performAction() : void`
Si occupa di mostrare all'utente il menu.

- `+setActionDatas(datas : Object) : void`
Carica la lista degli altri utenti.

Argomenti:

- * **datas** : carica la lista degli altri utenti.

3.5.2.51 sgad::clienttier::controller::actions::StealResources

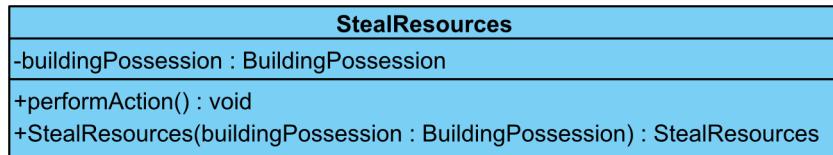


Figura 56: Diagramma della classe `sgad::clienttier::controller::actions::StealResources`

- **Descrizione:** classe per la gestione del saccheggio di un edificio produttivo di un altro utente.
- **Utilizzo:** viene usata per eseguire l'effettivo saccheggio di un edificio di produzione desiderato dall'utente vincitore dell'attacco.
- **Classi ereditate**
 - Action
- **Attributi**
 - -buildingPossession : BuildingPossession
L'edificio da cui saccheggiare le risorse.
- **Metodi**
 - +StealResources(**buildingPossession** : BuildingPossession) : StealResources
Costruttore della classe.

Argomenti:

- * **buildingPossession** : l'edificio da cui saccheggiare le risorse.

- +performAction() : void
Si occupa di attuare il saccheggio.

3.5.2.52 sgad::clienttier::controller::actions::TrainUnit

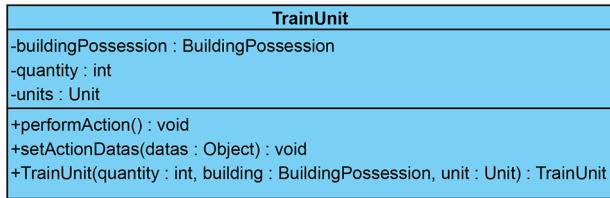


Figura 57: Diagramma della classe `sgad::clienttier::controller::actions::TrainUnit`

- **Descrizione:** classe per la gestione dell’addestramento di unità in un edificio.
- **Utilizzo:** viene utilizzata per eseguire l’effettivo addestramento delle unità desiderate dall’utente.
- **Classi ereditate**

– `Action`

- **Attributi**

- `-buildingPossession : BuildingPossession`
L’edificio a cui aggiungere le unità.
- `-quantity : int`
La quantità di unità da aggiungere.
- `-units : Unit`
Il tipo di unità da aggiungere.

- **Metodi**

- `+TrainUnit(building : BuildingPossession, quantity : int, unit : Unit) : TrainUnit`
Classe per l’aggiunta di nuove unità ad un edificio.

Argomenti:

- * `building` : l’edificio a cui aggiungere le unità.
- * `quantity` : la quantità di unità da aggiungere.
- * `unit` : il tipo di unità da aggiungere.

- `+performAction() : void`
Aggiunge le unità richieste.

- `+setActionDatas(datas : Object) : void`
Imposta i dati necessari.

Argomenti:

* **datas** : i dati da impostare.

3.5.2.53 sgad::clienttier::controller::actions::UpgradeBuilding

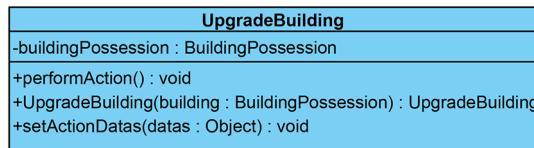


Figura 58: Diagramma della classe `sgad::clienttier::controller::actions::UpgradeBuilding`

- **Descrizione:** classe per la gestione del miglioramento di un edificio desiderato dall'utente.
- **Utilizzo:** viene utilizzata per eseguire l'effettivo miglioramento di un edificio desiderato dall'utente.
- **Classi ereditate**
 - Action
- **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio da migliorare.
- **Metodi**
 - `+UpgradeBuilding(building : BuildingPossession) : UpgradeBuilding`
Costruttore della classe.

Argomenti:

* **building** : l'edificio da migliorare.

- `+performAction() : void`
Migliora l'edificio indicato.
- `+setActionDatas(datas : Object) : void`
Imposta i dati per il miglioramento dell'edificio.

Argomenti:

* **datas** : i dati per il miglioramento.

3.5.2.54 sgad::clienttier::controller::actions::RequestUserList

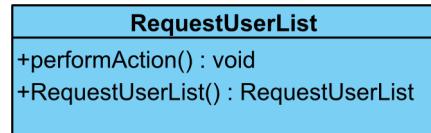


Figura 59: Diagramma della classe `sgad::clienttier::controller::actions::RequestUserList`

- **Descrizione:** classe per la richiesta della lista di utenti presenti
- **Utilizzo:** gestisce la richiesta della lista di utenti presenti
- **Metodi**
 - `+performAction() : void`
Esegue l'azione per richiede la lista degli utenti.
 - `+RequestUserList() : RequestUserList`
Costruttore della classe.

3.5.2.55 sgad::clienttier::controller::actions::ShowLogMenu

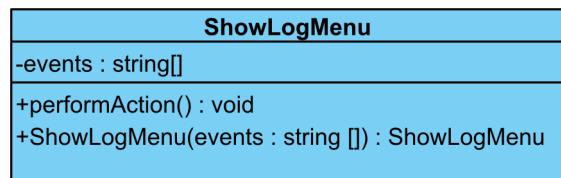


Figura 60: Diagramma della classe `sgad::clienttier::controller::actions::ShowLogMenu`

- **Descrizione:** classe per la visualizzazione di un menu di gestione account
- **Utilizzo:** permette la visualizzazione di un menu di gestione account
- **Attributi**
 - `-events : string[]`
L'insieme di eventi che si sono verificati.
- **Metodi**

– `+performAction() : void`
 Esegue l'azione per visualizzare l'elenco degli eventi verificatesi.

– `+ShowLogMenu(events : string[]) : ShowLogMenu`
 Costruttore della classe.

Argomenti:

* `events` : la lista degli eventi verificatesi.

3.6 Componente sgad::clienttier::controller::backupmanager

3.6.1 Informazioni sul package

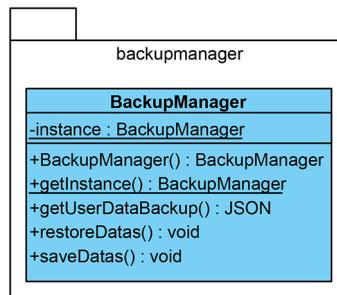


Figura 61: Diagramma della componente sgad::clienttier::controller::- backupmanager

- **Descrizione:** componente che si occupa di gestire il mantenimento dei dati e delle viste dell'utente proprietario dell'account.
- **Padre:** controller

3.6.2 Classi

3.6.2.1 sgad::clienttier::controller::backupmanager::BackupManager

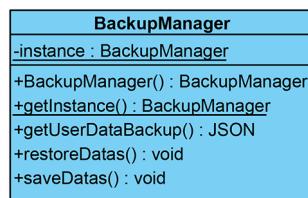


Figura 62: Diagramma della classe sgad::clienttier::controller::backupmanager::- BackupManager

- **Descrizione:** classe che si occupa di gestire il mantenimento dei dati e delle viste dell'utente proprietario dell'account.
- **Utilizzo:** viene utilizzata per memorizzare i dati e le viste dell'utente senza necessariamente richiederle, e quindi ricostruire.
- **Attributi**
 - `-instance : BackupManager`
Unica istanza dell'oggetto creato.
- **Metodi**
 - `+BackupManager() : BackupManager`
Costruttore della classe.
 - `+getInstance() : BackupManager`
Restituisce l'unica istanza dell'oggetto già creata, nel qual caso l'oggetto sia già stato richiesto.
 - `+getUserDataBackup() : JSON`
Fornisce i dati dell'utente che sono stati salvati.
 - `+saveDatas() : void`
Salva i dati dell'utente giocatore.
 - `+restoreDatas() : void`
Ripristina i dati dell'utente giocatore.

3.7 Componente sgad::clienttier::controller::menufactory

3.7.1 Informazioni sul package

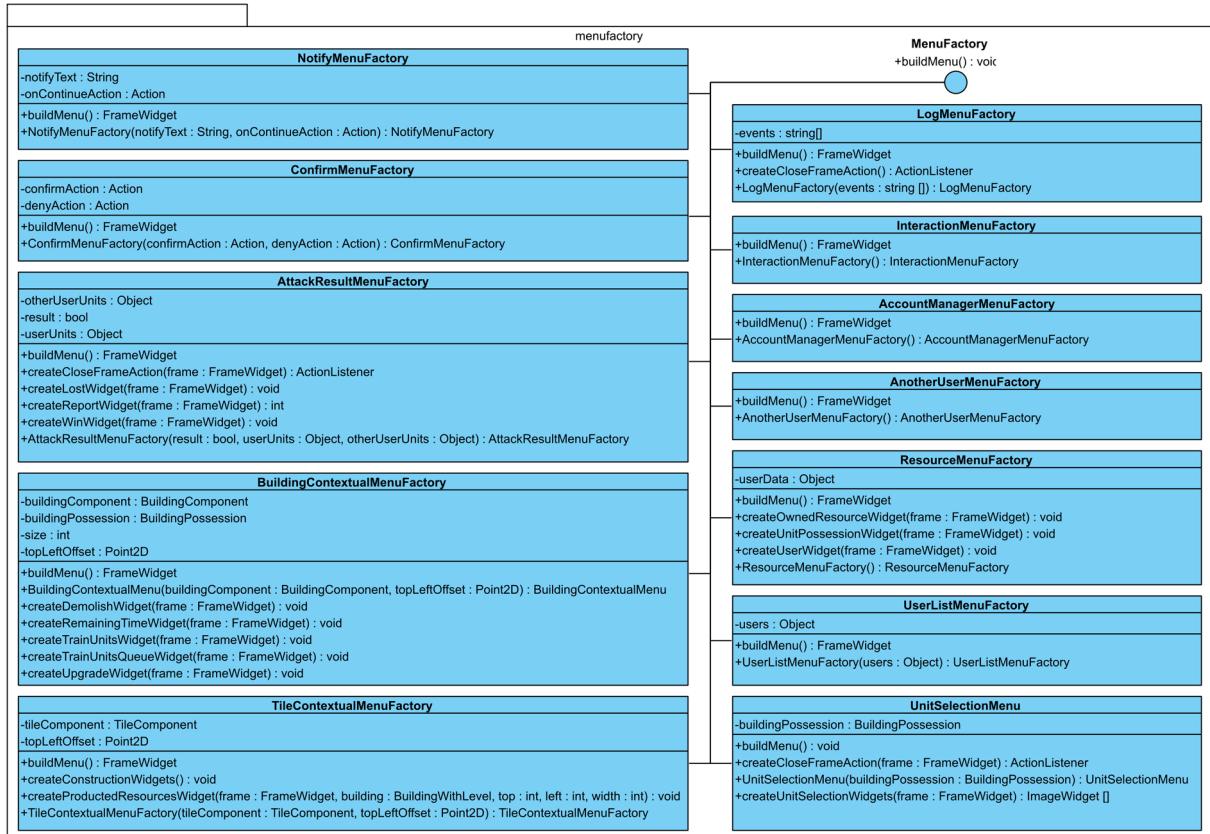


Figura 63: Diagramma della componente `sgad::clienttier::controller::menufactory`

- **Descrizione:** componente per la gestione dei vari menu che possono essere visualizzati durante le interazioni con il mondo di gioco.
- **Padre:** controller
- **Interazioni con altri componenti**
 - **components:** componente per la gestione degli oggetti del mondo di gioco e della relativa visualizzazione.
 - **widget:** componente per la gestione dei widget per permettere all’utente di interagire con il mondo di gioco attraverso menu e finestre.

3.7.2 Classi

3.7.2.1 sgad::clienttier::controller::menufactory::MenuFactory



Figura 64: Diagramma della classe `sgad::clienttier::controller::menufactory::-MenuFactory`

- **Descrizione:** interfaccia per i possibili menu che possono essere visualizzati durante l’interazione dell’utente con il mondo di gioco.
- **Utilizzo:** viene utilizzata per fornire una firma di metodo standard per qualsiasi tipo di menu che è possibile costruire. Le classi che realizzano l’interfaccia si occuperanno di ridefinire il metodo per costruire il menu effettivo. Permette di disaccoppiare l’effettivo menu che viene creato dall’oggetto che richiede la costruzione di un menu.
Per un maggior approfondimento, si veda la sezione Visualizzazione dei menu.
- **Sottoclassi**
 - `AccountManagerMenuFactory`
 - `AnotherUserMenuFactory`
 - `AttackResultMenuFactory`
 - `BuildingContextualMenuFactory`
 - `ConfirmMenuFactory`
 - `InteractionMenuFactory`
 - `LogMenuFactory`
 - `NotifyMenuFactory`
 - `ResourceMenuFactory`
 - `TileContextualMenuFactory`
 - `UnitSelectionMenu`
 - `UserListMenuFactory`
- **Metodi**
 - `+buildMenu() : void`
Costruisce il menu.

3.7.2.2 `sgad::clienttier::controller::menufactory::AccountManagerMenuFactory`

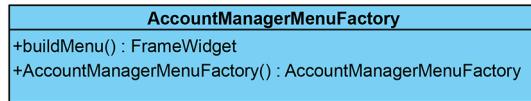


Figura 65: Diagramma della classe `sgad::clienttier::controller::menufactory::-AccountManagerMenuFactory`

- **Descrizione:** classe per la costruzione di un menu che permetta all'utente di modificare il proprio account.
- **Utilizzo:** viene utilizzata per costruire un menu che permetta all'utente di modificare il proprio account.
- **Classi ereditate**
 - `MenuFactory`
- **Metodi**
 - `+buildMenu() : FrameWidget`
Costruisce il menu che permette all'utente di gestire il proprio account.
 - `+AccountManagerMenuFactory() : AccountManagerMenuFactory`
Costruttore della classe.

3.7.2.3 sgad::clienttier::controller::menufactory::AnotherUserMenuFactory



Figura 66: Diagramma della classe `sgad::clienttier::controller::menufactory::-AnotherUserMenuFactory`

- **Descrizione:** classe per la costruzione di un menu per il ritorno al villaggio dell'utente.
- **Utilizzo:** viene utilizzata per creare un menu che richieda l'inserimento di un dato numerico da parte dell'utente. Si occupa quindi di istanziare oggetti di tipo `Widget` ai quali assocerà oggetti di tipo `Action` indicando a che evento associare tali azioni. Richiede di conoscere eventuali limiti da impostare al numero inserito.
- **Classi ereditate**
 - `MenuFactory`
- **Metodi**

- `+AnotherUserMenuFactory()` : `AnotherUserMenuFactory`
Costruttore della classe.

- `+buildMenu()` : `FrameWidget`
Fornisce il menu che permette all'utente di poter tornare al proprio villaggio.

3.7.2.4 sgad::clienttier::controller::menufactory::AttackResultMenuFactory

AttackResultMenuFactory
<code>-otherUserUnits : Object</code> <code>-result : bool</code> <code>-userUnits : Object</code>
<code>+buildMenu() : FrameWidget</code> <code>+createCloseFrameAction(frame : FrameWidget) : ActionListener</code> <code>+createLostWidget(frame : FrameWidget) : void</code> <code>+createReportWidget(frame : FrameWidget) : int</code> <code>+createWinWidget(frame : FrameWidget) : void</code> <code>+AttackResultMenuFactory(result : bool, userUnits : Object, otherUserUnits : Object) : AttackResultMenuFactory</code>

Figura 67: Diagramma della classe `sgad::clienttier::controller::menufactory::-AttackResultMenuFactory`

- **Descrizione:** classe per la creazione di un menu che permetta all'utente di visualizzare l'esito dello scontro.
- **Utilizzo:** viene utilizzata per creare un menu che permetta all'utente di visualizzare l'esito dello scontro. Permette inoltre di passare al villaggio dell'utente se l'esito della battaglia è positivo.
- **Classi ereditate**
 - `MenuFactory`
- **Attributi**
 - `-avversaryVillage : Object`
Il villaggio del giocatore attaccato.
 - `-otherUserUnits : Object`
Le unità perse dal giocatore attaccato.
 - `-result : bool`
Indica se la battaglia è andata a buon fine o meno.
 - `-userUnits : Object`
Le unità perse dall'utente.

- Metodi

– `+AttackResultMenuFactory(avversaryVillage : Object, otherUserUnits : Object, result : bool, userUnits : Object) : AttackResultMenuFactory`
Costruttore della classe.

Argomenti:

- * `avversaryVillage` : il villaggio del giocatore attaccato.
- * `otherUserUnits` : le unità perse dal giocatore attaccato.
- * `result` : indica se la battaglia è andata a buon fine o meno.
- * `userUnits` : le unità perse dall'utente.

– `+buildMenu() : FrameWidget`
Fornisce il frame contenente il menu per visualizzare all'utente l'esito della battaglia.

– `+createWinWidget(frame : FrameWidget) : void`
Crea il frame col menu del risultato per il vincitore.

Argomenti:

- * `frame` : il frame che contiene il menu.

– `+createLostWidget(frame : FrameWidget) : void`
Crea il frame col menu del risultato per il perdente.

Argomenti:

- * `frame` : il frame che contiene il menu.

– `+createReportWidget(frame : FrameWidget) : int`
Crea il frame col menu che mostra il report finale della battaglia.

Argomenti:

- * `frame` : il frame che contiene il menu.

– `+createCloseFrameAction(frame : FrameWidget) : ActionListener`
Crea l'azione associata alla chiusura del menu generale col risultato della battaglia.

Argomenti:

- * `frame` : il frame che contiene il menu.

3.7.2.5 sgad::clienttier::controller::menufactory::BuildingContextualMenuFactory

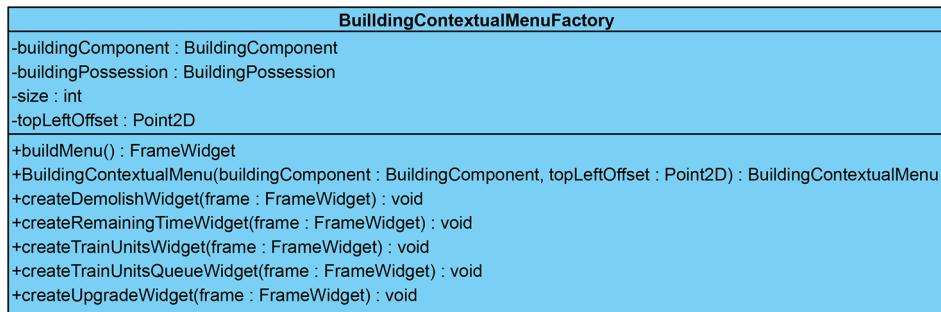


Figura 68: Diagramma della classe `sgad::clienttier::controller::menufactory::BuildingContextualMenuFactory`

- **Descrizione:** classe per la creazione di un menu di tipo contestuale relativo ad una costruzione.
- **Utilizzo:** viene utilizzata per creare dinamicamente un menu adatto secondo le caratteristiche dell'edificio richiesto. Si occupa quindi di istanziare oggetti di tipo `Widget` ai quali assocerà oggetti di tipo `Action` indicando a che evento associare tali azioni.
- **Classi ereditate**
 - `MenuFactory`
- **Attributi**
 - `-buildingComponent : BuildingComponent`
L'edificio di riferimento dal quale recuperare i dati.
 - `-buildingPossession : BuildingPossession`
Il tipo dell'edificio dal quale recuperare i dati.
 - `-size : int`
Grandezza dell'edificio.
 - `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.
- **Metodi**
 - `+BuildingContextualMenuFactory(buildingComponent : BuildingComponent, topLeftOffset : Point2D) : BuildingContextualMenuFactory`
Costruttore della classe.

Argomenti:

- * **buildingComponent** : l'edificio di riferimento dal quale recuperare i dati.
- * **topLeftOffset** : la posizione del punto in alto a sinistra.
- **+buildMenu()** : **FrameWidget**
Fornisce il menu contestuale relativo all'edificio con il quale l'utente desidera interagire.
- **+createUpgradeWidget(frame : FrameWidget) : void**
Costruisce un widget per il miglioramento di un edificio.

Argomenti:

- * **frame** : il frame nel quale inserire il widget.
- **+createDemolishWidget(frame : FrameWidget) : void**
Costruisce un widget per demolire l'edificio.

Argomenti:

- * **frame** : il frame nel quale inserire il widget.
- **+createTrainUnitsWidget(frame : FrameWidget) : void**
Costruisce un widget per addestrare unità.

Argomenti:

- * **frame** : il frame al quale aggiungere il widget.
- **+createTrainUnitsQueueWidget(frame : FrameWidget) : void**
Costruisce un widget per visualizzare le unità in coda.

Argomenti:

- * **frame** : il frame al quale aggiungere il widget.
- **+createRemainingTimeWidget(frame : FrameWidget) : void**
Costruisce un widget per visualizzare il tempo rimanente.

Argomenti:

- * **frame** : il frame nel quale inserire il widget.
- **+createPreconditionsWidget(frame : FrameWidget) : void**
Costruisce un widget per visualizzare una precondizione.

Argomenti:

- * **frame** : il frame nel quale inserire il widget.
- **+createProductedResourcesWidget(frame : FrameWidget) : void**
Costruisce un widget per visualizzare le risorse prodotte.

Argomenti:

- * **frame** : il frame nel quale inserire il widget.

3.7.2.6 sgad::clienttier::controller::menufactory::ConfirmMenuFactory

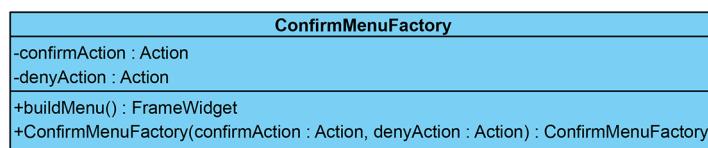


Figura 69: Diagramma della classe **sgad::clienttier::controller::menufactory::-ConfirmMenuFactory**

- **Descrizione:** classe per la creazione di un menu che permetta di confermare una generica azione.
- **Utilizzo:** viene utilizzata per creare un menu standard che permetta all'utente di confermare o meno un'operazione. Si occupa quindi di istanziare oggetti di tipo **Widget** ai quali assocerà oggetti di tipo **Action** indicando a che evento associare tali azioni. Richiede che vengano specificate quali siano le azioni da associare alla conferma e all'annullamento dell'operazione al momento della sua creazione.
- **Classi ereditate**
 - **MenuFactory**
- **Attributi**
 - **-confirmAction : Action**
L'azione da eseguire alla conferma.
 - **-denyAction : Action**
L'azione da eseguire all'annullamento.
- **Metodi**
 - **+ConfirmMenuFactory(confirmAction : Action, denyAction : Action) :- ConfirmMenuFactory**
Costruttore della classe.

Argomenti:

- * `confirmAction` : l'azione da eseguire alla conferma.
- * `denyAction` : l'azione da eseguire all'annullamento.
- `+buildMenu() : FrameWidget`
Fornisce il menu che permette di richiedere una conferma da parte dell'utente per proseguire con un'operazione.

3.7.2.7 sgad::clienttier::controller::menufactory::InteractionMenuFactory

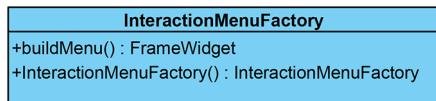


Figura 70: Diagramma della classe `sgad::clienttier::controller::menufactory::-InteractionMenuFactory`

- **Descrizione:** classe per la creazione di un menu per interagire con l'intero mondo di gioco.
- **Utilizzo:** viene utilizzata per costruire un menu che permetta all'utente di eseguire interazioni legate all'intero mondo di gioco.
- **Classi ereditate**
 - `MenuFactory`
- **Metodi**
 - `+InteractionMenuFactory() : InteractionMenuFactory`
Costruttore della classe.
 - `+buildMenu() : FrameWidget`
Fornisce il menu che permette all'utente di effettuare delle operazioni per poter interagire a livello globale con il mondo di gioco.

3.7.2.8 sgad::clienttier::controller::menufactory::LogMenuFactory

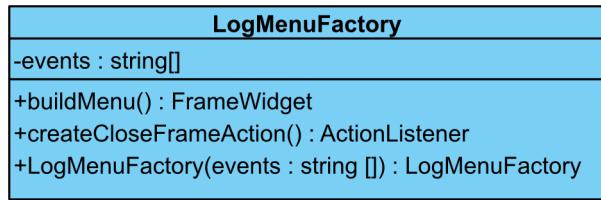


Figura 71: Diagramma della classe sgad::clienttier::controller::menufactory::- LogMenuFactory

- **Descrizione:** classe per la costruzione di un menu che permetta all'utente di modificare il proprio account.

- **Utilizzo:** gestisce la costruzione di un menu che permetta all'utente di modificare il proprio account.

- **Classi ereditate**

- MenuFactory

- **Attributi**

- -events : string[]
Il log degli eventi.

- **Metodi**

- +buildMenu() : FrameWidget
Costruisce il menu per visualizzare gli eventi verificatesi.
 - +createCloseFrameAction() : ActionListener
Costruisce l'elenco delle azioni necessarie per rimuovere i vari oggetti grafici.
 - +LogMenuFactory(events : string[]) : LogMenuFactory
Costruttore della classe.

Argomenti:

- * events : il log degli eventi.

3.7.2.9 sgad::clienttier::controller::menufactory::NotifyMenuFactory

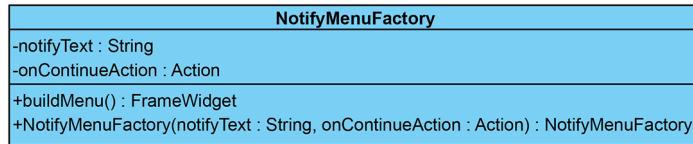


Figura 72: Diagramma della classe `sgad::clienttier::controller::menufactory::-NotifyMenuFactory`

- **Descrizione:** classe per la costruzione di un menu per la visualizzazione di una notifica.
- **Utilizzo:** viene utilizzata dalle varie azioni per notificare un'evento che impedisce il normale proseguimento dell'azione.
- **Classi ereditate**
 - `MenuFactory`
- **Attributi**
 - `-notifyText : String`
Il testo da visualizzare come contenuto della notifica.
 - `-onContinueAction : Action`
L'azione da eseguire al proseguimento della notifica.
- **Metodi**
 - `+NotifyMenuFactory (notifyText : String, onContinueAction : Action)`
: `NotifyMenuFactory`
Costruttore della classe.
 - Argomenti:**
 - * `notifyText` : il testo da visualizzare come contenuto della notifica.
 - * `onContinueAction` : l'azione da eseguire al proseguimento della notifica.
 - `+buildMenu() : FrameWidget`
Fornisce il menu per notificare all'utente qualche avviso in merito all'operazione attualmente in corso.

3.7.2.10 `sgad::clienttier::controller::menufactory::ResourceMenuFactory`

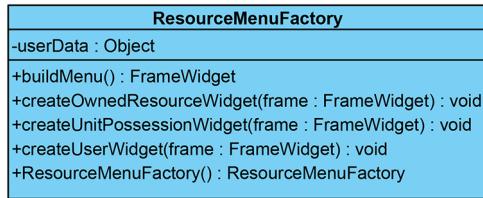


Figura 73: Diagramma della classe `sgad::clienttier::controller::menufactory::-ResourceMenuFactory`

- **Descrizione:** classe per la creazione di un menu che visualizzi le risorse e le unità possedute dall'utente.

- **Utilizzo:** viene utilizzata per costruire il menu delle risorse possedute dall'utente e delle unità.

- **Classi ereditate**

- `MenuFactory`

- **Attributi**

- `-userData : Object`
I dati dell'utente.

- **Metodi**

- `+ResourceMenuFactory() : ResourceMenuFactory`
Costruttore della classe.

- `+buildMenu() : FrameWidget`
Fornisce il menu per poter visualizzare le unità e le risorse possedute dall'utente.

- `+createOwnedResourceWidget(frame : FrameWidget) : void`
Crea un widget per le risorse possedute dall'utente.

Argomenti:

- * `frame` : crea un widget per le risorse possedute dall'utente.

- `+createUnitPossessionWidget(frame : FrameWidget) : void`
Crea un widget per le unità possedute dall'utente.

Argomenti:

- * `frame` : il frame al quale aggiungere il widget.

- `+createUserWidget(frame : FrameWidget) : void`
Crea un widget per il nome dell'utente.

Argomenti:

- * `frame` : il frame al quale aggiungere il widget.

3.7.2.11 sgad::clienttier::controller::menufactory::TileContextualMenuFactory

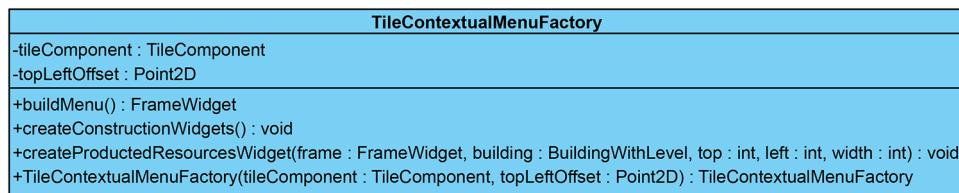


Figura 74: Diagramma della classe `sgad::clienttier::controller::menufactory::-TileContextualMenuFactory`

- **Descrizione:** classe per la creazione di un menu che visualizzi gli edifici costruibili in una casella.
- **Utilizzo:** viene utilizzata per costruire un menu contestuale che permetta all'utente di costruire un edificio nella casella selezionata.
- **Classi ereditate**

- `MenuFactory`

- **Attributi**

- `-tileComponent : TileComponent`
La casella nella quale verrà costruito l'edificio.
- `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.

- **Metodi**

- `+TileContextualMenuFactory(tileComponent : TileComponent, topLeftOffset : Point2D) : TileContextualMenuFactory`
Costruttore della classe.

Argomenti:

- * `tileComponent` : la casella nella quale verrà costruito l'edificio.

- * `topLeftOffset` : la posizione del punto in alto a sinistra.
- `+buildMenu() : FrameWidget`
Costruisce il menu contestuale per permettere all'utente di scegliere la costruzione da avviare relativamente alla casella selezionata.
- `+createConstructionWidgets() : void`
Costruisce il menu che contiene gli edifici che è possibile creare nel gioco.
- `+createProducedResourcesWidget(building : BuildingWithLevel, frame : FrameWidget, left : int, top : int, width : int) : void`
Costruisce il widget contenente le informazioni riguardo le risorse che vengono prodotte da un determinato edificio.

Argomenti:

- * `building` : l'edificio che produce risorse.
- * `frame` : il frame contenente gli edifici costruibili.
- * `left` : coordinata y di partenza del widget.
- * `top` : coordinata x di partenza del widget.
- * `width` : larghezza dell'area del widget.

3.7.2.12 sgad::clienttier::controller::menufactory::UnitSelectionMenu

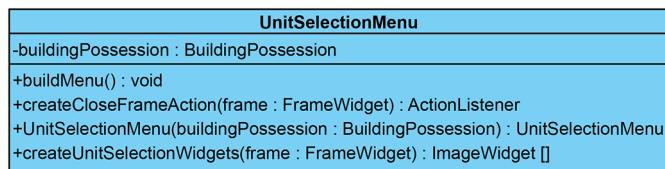


Figura 75: Diagramma della classe `sgad::clienttier::controller::menufactory::-UnitSelectionMenu`

- **Descrizione:** classe per la creazione di un menu che visualizzi le unità producibili in un edificio.
- **Utilizzo:** viene utilizzata per costruire un menu che permetta all'utente di selezionare quali unità desidera addestrare o congedare.
- **Classi ereditate**

- MenuFactory

- **Attributi**

- -buildingPossession : BuildingPossession
L'edificio associato alla produzione di unità.

- **Metodi**

- +UnitSelectionMenu(buildingPossession : BuildingPossession) : UnitSelectionMenu
Costruttore della classe.

Argomenti:

- * buildingPossession : l'edificio associato alla produzione di unità.

- +buildMenu() : FrameWidget
Fornisce il menu che permette all'utente di selezionare il tipo di unità da addestrare o da congedare.
- +createUnitSelectionWidgets(frame : FrameWidget) : Array<ImageWidget>
Crea dei widgets per visualizzare le unità addestrabili.

Argomenti:

- * frame : il frame che contiene il menu.

- +createCloseFrameAction(frame : FrameWidget) : ActionListener
Costruisce l'azione per la chiusura del menu.

Argomenti:

- * frame : il frame che contiene il menu.

3.7.2.13 sgad::clienttier::controller::menufactory::UserListMenuFactory

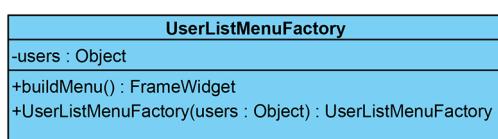


Figura 76: Diagramma della classe sgad::clienttier::controller::menufactory::-UserListMenuFactory

- **Descrizione:** classe per la visualizzazione della lista di altri giocatori connessi.

- **Utilizzo:** viene utilizzata per costruire un menu che visualizzi i vari utenti con i quali l'utente può interagire.

- **Classi ereditate**

- MenuFactory

- **Attributi**

- -users : Object

La lista degli utenti giocatori connessi.

- **Metodi**

- +UserListMenuFactory(users : Object) : UserListMenuFactory

Costruttore della classe.

Argomenti:

- * users : la lista degli utenti giocatori connessi.

- +buildMenu() : FrameWidget

Fornisce un menu per visualizzare una lista di giocatori con i quali interagire.

3.8 Componente sgad::clienttier::controller::messageinterpreter

3.8.1 Informazioni sul package

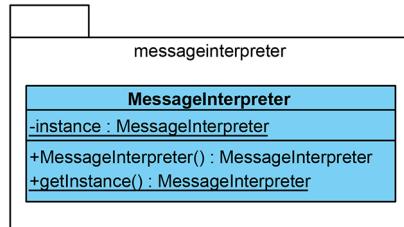


Figura 77: Diagramma della componente `sgad::clienttier::controller::messageinterpreter`

- **Descrizione:** componente per l'interpretazione delle reply ricevute dal server.
- **Padre:** controller
- **Interazioni con altri componenti**
 - **actions:** componente per la gestione delle possibili azioni che un utente può intraprendere durante l'interazione con il mondo di gioco.
 - **requester:** componente per la gestione dell'invio di richieste al server.

3.8.2 Classi

3.8.2.1 sgad::clienttier::controller::messageinterpreter::MessageInterpreter

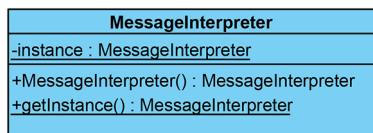


Figura 78: Diagramma della classe `sgad::clienttier::controller::messageinterpreter::MessageInterpreter`

- **Descrizione:** classe per l'interpretazione di una risposta che è stata ricevuta dal server in seguito ad una determinata richiesta.
- **Utilizzo:** viene utilizzata per analizzare un messaggio ricevuto dal server dopo che è stata inviata una richiesta. La classe analizza se il messaggio ricevuto ha al suo interno un ulteriore messaggio in *piggybacking_[g]*. Ciò può verificarsi quando sono state apportate modifiche ai dati che risiedono sul server non in merito ad operazioni eseguite dall'utente stesso (esempio: interazione con un altro utente). La classe invocherà determinate azioni per riportare le modifiche anche sul client.

Per un maggior approfondimento, si veda la sezione Interazione con il server.

- Attributi

- `-instance : MessageInterpreter`

Unica istanza dell'oggetto di tipo `MessageInterpreter` creato.

- Metodi

- `+MessageInterpreter() : MessageInterpreter`

Costruttore della classe.

- `+getInstance() : MessageInterpreter`

Restituisce l'unica istanza dell'oggetto già creata, nel qual caso l'oggetto sia già stato richiesto.

- `+analyzeMessage(json : Object) : JSON`

Analizza un oggetto in formato `JSON[g]` per determinare se all'interno dello stesso compare più di un singolo messaggio. Una volta analizzato l'oggetto, il metodo fornisce i dati relativi all'operazione originaria.

Argomenti:

- * `json` : jSON che deve essere analizzato.

3.9 Componente `sgad::clienttier::controller::requester`

3.9.1 Informazioni sul package

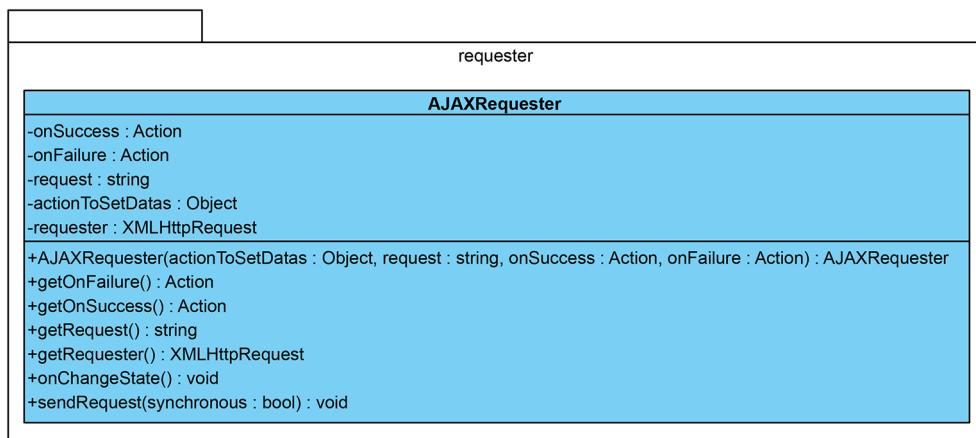


Figura 79: Diagramma della componente `sgad::clienttier::controller::requester`

- **Descrizione:** componente per la gestione dell'invio di richieste al server.
- **Padre:** controller

- **Interazioni con altri componenti**

- **actions**: componente per la gestione delle possibili azioni che un utente può intraprendere durante l'interazione con il mondo di gioco.
- **messageinterpreter**: componente per l'interpretazione delle reply ricevute dal server.
- **httpresponder**: componente per la gestione delle richieste $HTTP_{|g|}$ ricevute dai vari client.

3.9.2 Classi

3.9.2.1 sgad::clienttier::controller::requester::AJAXRequester

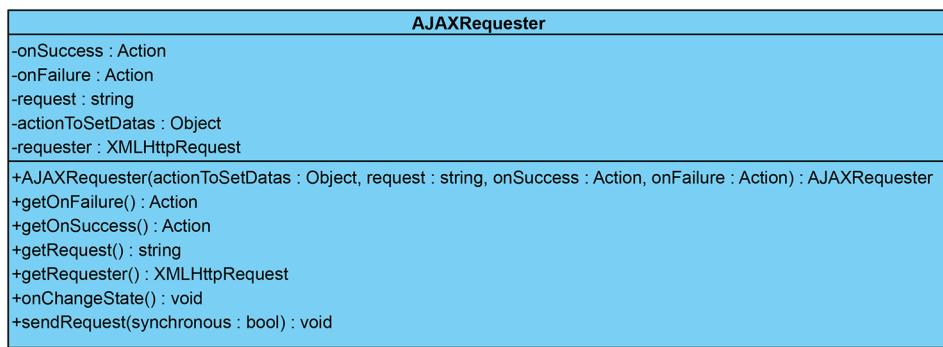


Figura 80: Diagramma della classe `sgad::clienttier::controller::requester::AJAXRequester`

- **Descrizione:** classe per la gestione dell'invio delle richieste al server da parte del client.
- **Utilizzo:** viene utilizzata per inviare una richiesta al server. Tale classe si occuperà di eseguire determinate azioni in base a determinati errori nella comunicazione. Essa implementerà le richieste tramite la tecnologia $AJAX_{|g|}$. Necessita di conoscere le azioni da eseguire nel caso in cui la richiesta venga soddisfatta correttamente o meno.

Per un maggior approfondimento, si veda la sezione Interazione con il server.

- **Attributi**

- `-actionToSetDatas : Object`
Azioni da impostare.
- `-onFailure : Action`
Azione da eseguire al fallimento della richiesta.
- `-onSuccess : Action`
Azione da eseguire al successo della richiesta.
- `-request : String`
Richiesta che deve essere trasmessa al server.

- `-requester : XMLHttpRequest`
Oggetto che gestisce la richiesta.

- Metodi

- `+AJAXRequester(actionToSetDatas : Object, onFailure : Action, onSuccess : Action, request : string) : AJAXRequester`
Costruttore della classe.

Argomenti:

- * `actionToSetDatas` : azioni da impostare.
- * `onFailure` : l'azione che deve essere eseguita al fallimento della richiesta.
- * `onSuccess` : l'azione che deve essere eseguita al successo della richiesta.
- * `request` : la richiesta che dovrà essere inviata.

- `+sendRequest(synchronous : bool) : void`
Invia la richiesta al server.

Argomenti:

- * `synchronous` : indica se la richiesta deve essere sincrona, valore pari a true, o asincrona, valore pari a false.

- `+onChangeState() : void`
Gestisce la modifica dello stato della richiesta inviata.

- `+getRequest() : string`
Fornisce la richiesta.

- `+getRequester() : XMLHttpRequest`
Fornisce l'oggetto che gestisce la richiesta.

- `+getOnSuccess() : Action`
Fornisce l'azione da eseguire al successo della richiesta.

- `+getOnFailure() : Action`
Fornisce l'azione da eseguire al fallimento della richiesta.

3.10 Componente sgad::clienttier::model

3.10.1 Informazioni sul package

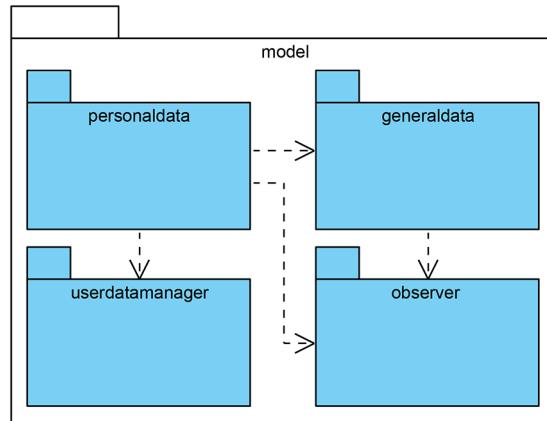


Figura 81: Diagramma della componente `sgad::clienttier::model`

- **Descrizione:** componente Model dell'architettura MVC. Essa memorizza tutti i dati di gioco dell'utente su cui la view si basa per disegnarsi.
- **Padre:** `clienttier`
- **Interazioni con altri componenti**
 - **controller:** componente Controller dell'architettura MVC. Essa esegue le operazioni che l'utente ha richiesto tramite la view agendo se necessario sul model data.
 - **view:** componente View dell'architettura MVC. Essa gestisce la visualizzazione del gioco e l'input utente all'interno del canvas HTML5.
- **Package contenuti**
 - **generaldata:** componente del Model per la raccolta dei dati non specifici dell'utente ma relativi all'intero gioco.
 - **observer:** componente per raccogliere le classi che realizzeranno il pattern Observer tipico dell'architettura MVC.
 - **personaldata:** componente del model che gestisce le classi per la memorizzazione dei dati legati all'utente.
 - **userdatamanager:** componente per il mantenimento di un accesso globale ai dati dell'utente di cui si sta attualmente visualizzando il villaggio.

3.11 Componente sgad::clienttier::model::generaldata

3.11.1 Informazioni sul package

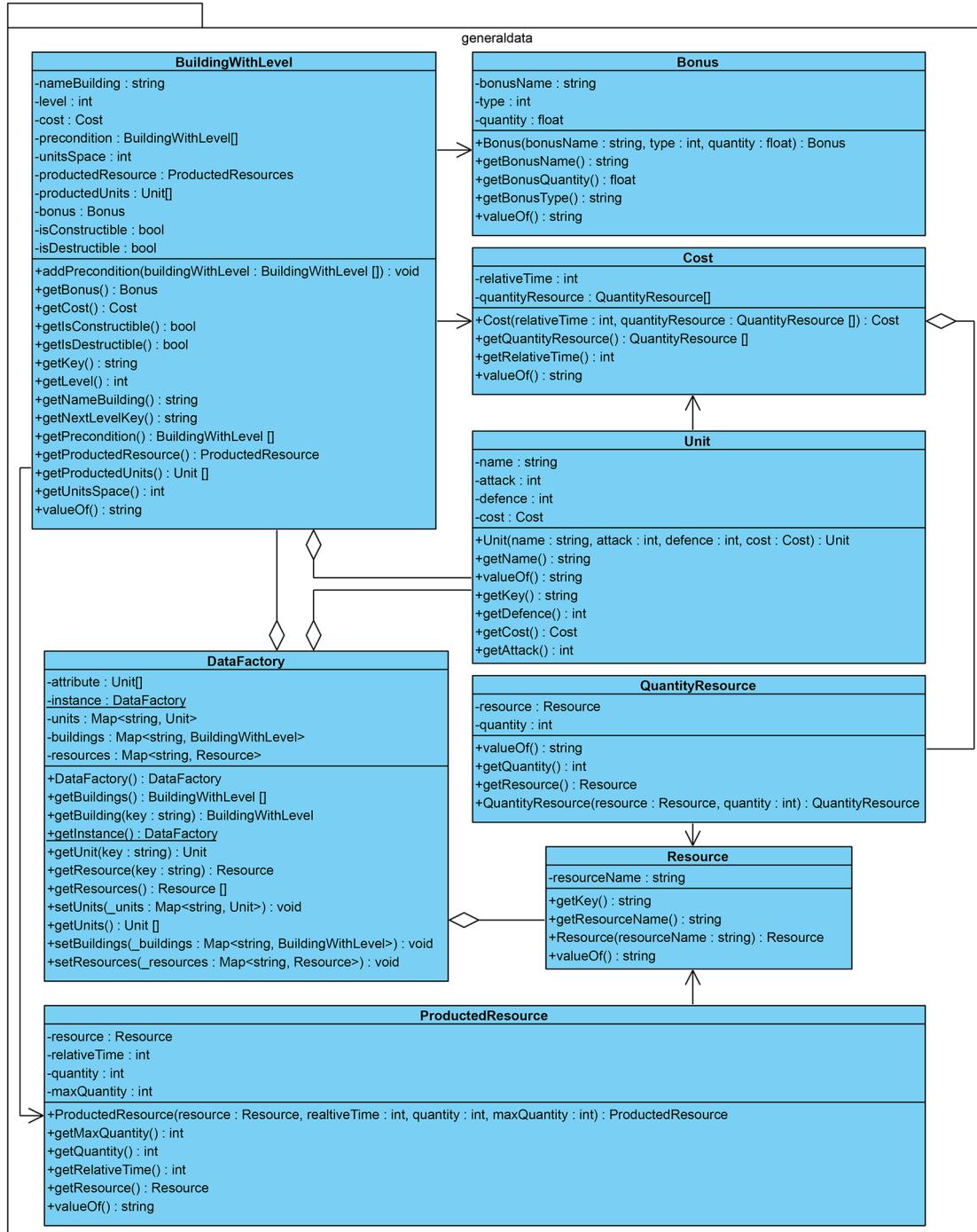


Figura 82: Diagramma della componente `sgad::clienttier::model::generaldata`

- **Descrizione:** componente del Model per la raccolta dei dati non specifici dell'utente ma relativi all'intero gioco.
- **Padre:** model
- **Interazioni con altri componenti**
 - **observer:** componente per raccogliere le classi che realizzeranno il pattern Observer tipico dell'architettura MVC.
 - **personaldata:** componente del model che gestisce le classi per la memorizzazione dei dati legati all'utente.

3.11.2 Classi

3.11.2.1 sgad::clienttier::model::generaldata::Bonus

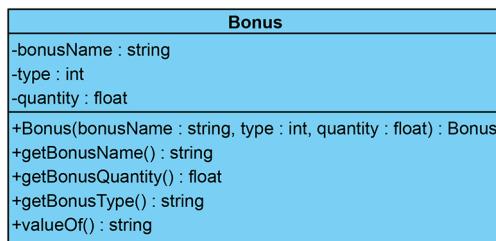


Figura 83: Diagramma della classe sgad::clienttier::model::generaldata::Bonus

- **Descrizione:** classe per la gestione dei bonus eventualmente disponibili per certi edifici.
- **Utilizzo:** viene utilizzata per mantenere l'informazione sul nome del bonus, sul tipo e sulla quantità dello stesso.
- **Attributi**
 - `-bonusName : string`
Rappresenta il nome del bonus.
 - `-quantity : float`
Rappresenta la potenza del bonus.
 - `-type : int`
Codice identificativo che rappresenta il tipo di bonus e di conseguenza i parametri che influenza con potenza pari al parametro quantity.
- **Metodi**
 - `+Bonus(bonusName : string, quantity : float, type : int) : Bonus`
Costruttore della classe.

Argomenti:

- * **bonusName** : il nome del bonus.
- * **quantity** : il codice identificativo del bonus, il quale determina quale caratteristica del BuildingWithLevel influenza.
- * **type** : la potenza del bonus, cioè quanto influenza la caratteristica del BuildingWithLevel.
- **+getBonusName() : string**
Controlla l'accesso al nome del bonus.
- **+getBonusQuantity() : float**
Controlla l'accesso alla quantità del bonus.
- **+getBonusType() : string**
Controlla l'accesso al tipo del bonus.
- **+valueOf() : string**
Ridefinizione del metodo valueOf di Bonus

3.11.2.2 sgad::clienttier::model::generaldata::BuildingWithLevel



Figura 84: Diagramma della classe `sgad::clienttier::model::generaldata::BuildingWithLevel`

- **Descrizione:** classe per la gestione di un edificio ad un particolare livello.
- **Utilizzo:** viene utilizzata per memorizzare tutte le informazioni che descrivono un edificio ad un particolare livello.
- **Attributi**

- `-bonus : Bonus`
Bonus a disposizione della costruzione.
- `-cost : Cost`
Costo per costruire la costruzione.
- `-isConstructible : bool`
Indica se l'edificio è direttamente edificabile su una cella dell'area di gioco.
- `-isDestructible : bool`
Indica se l'edificio è rimovibile
- `-level : int`
Livello della costruzione.
- `-nameBuilding : string`
Nome della costruzione.
- `-precondition : BuildingWithLevel[]`
Costruzioni necessarie per poter costruire questo edificio.
- `-producedResource : ProductedResources`
Risorsa che questo edificio produce.
- `-producedUnits : Unit[]`
Unità che questo edificio può produrre.
- `-unitsSpace : int`
Numero di posti per le unità che questo edificio offre.

- **Metodi**

- `+BuildingWithLevel(bonus : Bonus, cost : Cost, isConstructible : bool, isDestructible : bool, level : int, nameBuilding : string, precondition : BuildingWithLevel [], producedResource : ProductedResource, producedUnits : Unit [], unitsSpace : int) : BuildingWithLevel`
Costruttore della classe.

Argomenti:

- * `bonus` : il bonus a disposizione della costruzione.
- * `cost` : il costo per costruire la costruzione.
- * `isConstructible` : indica se l'edificio è direttamente edificabile su una cella dell'area di gioco.

- * `isDestructible` : indica se l'edificio è rimuovibile
 - * `level` : il livello della costruzione.
 - * `nameBuilding` : il nome della costruzione.
 - * `precondition` : le costruzioni necessarie per poter costruire questo edificio.
 - * `productedException` : la risorsa che questo edificio produce.
 - * `productedUnits` : le unità che questo edificio può produrre.
 - * `unitsSpace` : i posti per le unità che questo edificio offre.
- `+getKey() : string`
Fornisce la chiave univoca che identifica l'edificio.
- `+getNextLevelKey() : string`
Fornisce la chiave univoca che identifica il livello successivo dell'edificio.
- `+valueOf() : string`
Ridefinizione del metodo `valueOf` di `BuildingWithLevel`
- `+getUnitsSpace() : int`
Controlla l'accesso al numero di posti per le unità che offre questo edificio.
- `+getProductedUnits() : Array<Unit>`
Controlla l'accesso alla collezione di unità che questo edificio può produrre.
- `+getProductedException() : ProductedException`
Controlla l'accesso alla risorsa che viene prodotta da questo edificio.
- `+getPrecondition() : Array<BuildingWithLevel>`
Controlla l'accesso alla collezione di costruzioni necessarie per costruire questo edificio.
- `+getNameBuilding() : string`
Controlla l'accesso al nome della costruzione.
- `+getLevel() : int`
Controlla l'accesso al livello della costruzione.

- `+getCost() : Cost`
Controlla l'accesso al costo della costruzione.
- `+getBonus() : Bonus`
Controlla l'accesso al bonus a disposizione della costruzione.
- `+getIsConstructible() : bool`
Metodo getter per l'attributo isConstructible.
- `+getIsDestructible() : bool`
Metodo getter per l'attributo isDestructible
- `+addPrecondition(buildingWithLevel : Array<BuildingWithLevel>) : void`
Inserisce una nuova precondizione da soddisfare per poter costruire l'edificio.

Argomenti:

- * `buildingWithLevel` : lista di edifici da costruire soddisfacendo la precondizione.

3.11.2.3 sgad::clienttier::model::generaldata::Cost

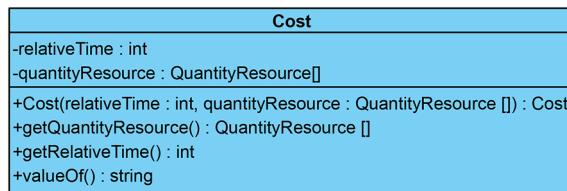


Figura 85: Diagramma della classe `sgad::clienttier::model::generaldata::Cost`

- **Descrizione:** classe per la gestione di un costo in termini di risorse e tempo.
- **Utilizzo:** viene utilizzata per la gestione di un costo in termini di quantità di risorse e tempo.
- **Attributi**
 - `-quantityResource : QuantityResource []`
Le quantità e il tipo di risorse necessarie.
 - `-relativeTime : int`
Il tempo necessario espresso in secondi.
- **Metodi**

– `+Cost(quantityResource : QuantityResource[], relativeTime : int) :- Cost`

Costruttore della classe.

Argomenti:

* `quantityResource` : le quantità e il tipo di risorse necessarie.

* `relativeTime` : il tempo necessario espresso in secondi.

– `+valueOf() : string`

Ridefinizione del metodo valueOf di Cost

– `+getRelativeTime() : int`

Controlla l'accesso alle quantità e al tipo di risorse necessarie.

– `+getQuantityResource() : Array<QuantityResource>`

Controlla l'accesso al tempo necessario espresso in secondi.

3.11.2.4 sgad::clienttier::model::generaldata::DataFactory

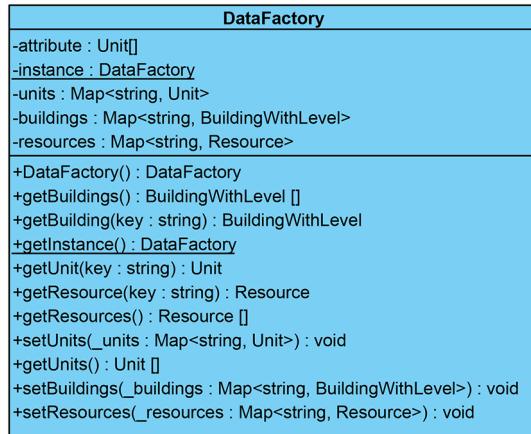


Figura 86: Diagramma della classe `sgad::clienttier::model::generaldata::DataFactory`

- **Descrizione:** classe che ritorna un'istanza delle componenti di gioco.
- **Utilizzo:** viene utilizzata per ottenere riferimenti agli oggetti che compongono la sua stessa componente.
- **Attributi**

- `-buildings : Map<string, BuildingWithLevel>`
Le BuildingWithLevel disponibili nel gioco.
- `-instance : DataFactory`
L'unica istanza dell'oggetto di tipo DataFactory.
- `-resources : Map<string, Resource>`
Le Resource disponibili nel gioco.
- `-units : Map<string, Unit>`
Le Unit disponibili nel gioco.

- Metodi

- `+getBuilding(key : string) : BuildingWithLevel`
Fornisce un oggetto della classe BuildingWithLevel corrispondente alla key passata.

Argomenti:

* `key` : l'identificativo dell'edificio richiesto.

- `+getUnit(key : string) : Unit`
Ritorna la Unit corrispondente alla key in input.

Argomenti:

* `key` : l'identificativo dell'unità richiesta.

- `+getInstance() : DataFactory`
Ritorna l'unica istanza dell'oggetto.
- `+getResource(key : string) : Resource`
Ritorna la Resource corrispondente alla key in input.

Argomenti:

* `key` : chiave identificativa della Resource desiderata.

- `+setBuildings(_buildings : Map<string,BuildingWithLevel>) : void`
Setta le BuldingWithLevel disponibili nel gioco.

Argomenti:

* `_buildings` : le costruzioni disponibili nel gioco.

- `+setResources(_resources : Map<string, Resource>) : void`
Setta le Resource disponibili nel gioco.

Argomenti:

- * `_resources` : le risorse disponibili nel gioco.
 - `+setUnits(_units : Map<string, Unit>) : void`
Setta le Unit disponibili nel gioco.
- Argomenti:**
- * `_units` : le unità disponibili nel gioco.
 - `+getBuildings() : Array<BuildingWithLevel>`
Controlla l'accesso alle BuildingWithLevel disponibili nel gioco.
 - `+getResources() : Array<Resource>`
Controlla l'accesso alle Resource disponibili nel gioco.
 - `+getUnits() : Array<Unit>`
Controlla l'accesso alle Unità disponibili nel gioco.
 - `+DataFactory() : DataFactory`
Costruttore della classe.

3.11.2.5 sgad::clienttier::model::generaldata::ProducedResource

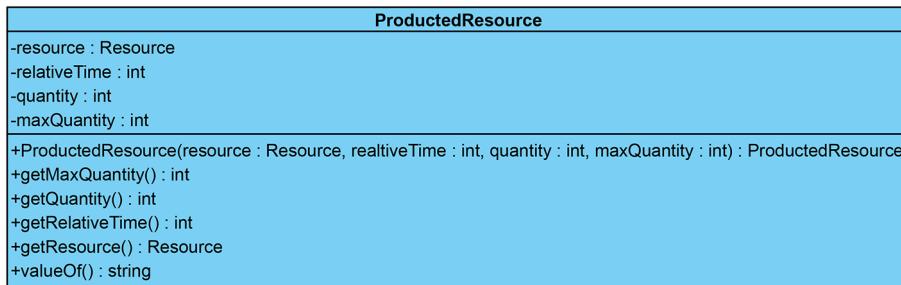


Figura 87: Diagramma della classe `sgad::clienttier::model::generaldata::ProducedResource`

- **Descrizione:** classe per la gestione delle informazioni relative alla risorsa prodotta da un edificio.
- **Utilizzo:** viene utilizzata per la memorizzazione delle informazioni relative alla risorsa prodotta da un edificio.
- **Attributi**

- `-maxQuantity : int`
Il massimo numero di unità di risorse accumulabili nell'edificio.

- `-quantity : int`
Il numero di risorse prodotte ogni multiplo di `relativeTime` che passa.
- `-relativeTime : int`
Indica ogni quanto l'edificio produce `quantity` unità di risorsa `resource`.
- `-resource : Resource`
La risorsa prodotta.

- Metodi

- `+ProducedResource(maxQuantity : int, quantity : int, relativeTime : int, resource : Resource) : ProducedResource`
Costruttore della classe.

Argomenti:

- * `maxQuantity` : il massimo numero di unità di risorse accumulabili nell'edificio.
- * `quantity` : il numero di risorse prodotte ogni multiplo di `relativeTime` che passa.
- * `relativeTime` : indica ogni quanto l'edificio produce `quantity` unità di risorsa `resource`.
- * `resource` : la risorsa prodotta.

- `+getMaxQuantity() : int`
Controlla l'accesso alla quantità massima possibile della risorsa.
- `+getQuantity() : int`
Controlla l'accesso alla quantità della risorsa.
- `+getRelativeTime() : int`
Controlla l'accesso all'unità di tempo necessaria per la produzione della risorsa.
- `+getResource() : Resource`
Controlla l'accesso alla risorsa.
- `+valueOf() : string`
Ridefinizione del metodo `valueOf` di `ProducedResource`

3.11.2.6 sgad::clienttier::model::generaldata::QuantityResource

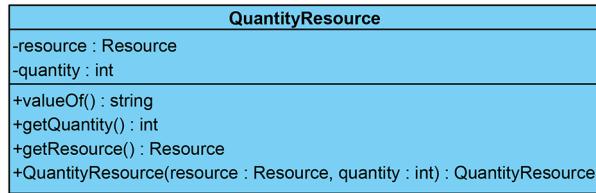


Figura 88: Diagramma della classe `sgad::clienttier::model::generaldata::QuantityResource`

- **Descrizione:** classe per la gestione della quantità di una particolare risorsa.

- **Utilizzo:** viene utilizzata per associare una quantità ad una risorsa.

- **Attributi**

- `-quantity : int`
Il numero di unità della data risorsa.
- `-resource : Resource`
La risorsa.

- **Metodi**

- `+QuantityResource(quantity : int, resource : Resource) : QuantityResource`
Costruttore della classe.

Argomenti:

- * `quantity` : il numero di unità della data risorsa.
- * `resource` : il nome della risorsa.
- `+getQuantity() : int`
Metodo getter della quantità della risorsa
- `+getResource() : Resource`
Metodo getter della risorsa
- `+valueOf() : string`
Ridefinizione del metodo `valueOf` di `QuantityResource`

3.11.2.7 sgad::clienttier::model::generaldata::Resource

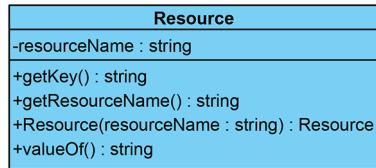


Figura 89: Diagramma della classe sgad::clienttier::model::generaldata::Resource

- **Descrizione:** classe che rappresenta una risorsa.
- **Utilizzo:** viene utilizzata per memorizzare le informazioni su una risorsa.
- **Attributi**

- `-resourceName : string`
Il nome della risorsa.

- **Metodi**

- `+Resource(resourceName : string) : Resource`
Costruttore della classe.

Argomenti:

- * `resourceName` : il nome della risorsa.
- `+getKey() : string`
Fornisce la chiave che identifica la risorsa.
- `+get resourceName() : string`
Controlla l'accesso al nome della risorsa.
- `+valueOf() : string`
Ridefinizione del metodo valueOf di Resource

3.11.2.8 sgad::clienttier::model::generaldata::Unit

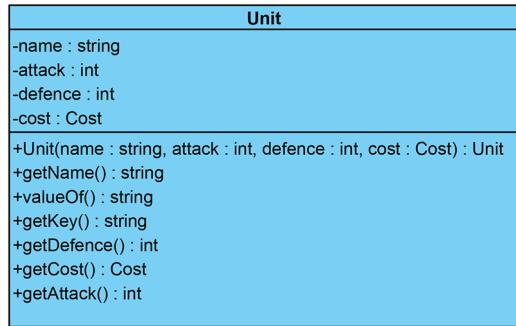


Figura 90: Diagramma della classe `sgad::clienttier::model::generaldata::Unit`

- **Descrizione:** classe per la gestione delle informazioni di una unità.
- **Utilizzo:** viene utilizzata per la memorizzazione delle informazioni sulle unità.
- **Attributi**

- `-attack : int`
La forza di attacco dell’unità.
- `-cost : Cost`
Il costo di produzione dell’unità.
- `-defence : int`
La forza di difesa dell’unità.
- `-name : string`
Il nome dell’unità.

- **Metodi**

- `+Unit(attack : int, cost : Cost, defence : int, name : string) :- Unit`
Costruttore della classe.

Argomenti:

- * `attack` : la forza di attacco dell’unità.
- * `cost` : il costo di produzione dell’unità.
- * `defence` : la forza di difesa dell’unità.
- * `name` : il nome dell’unità.

- `+getKey() : string`
Fornisce la chiave identificativa dell’unità.

- `+valueOf() : string`
Ridefinizione del metodo valueOf di Unit
- `+getName() : string`
Metodo getter del nome dell'unità
- `+getAttack() : int`
Metodo getter del valore di attacco dell'unità
- `+getDefence() : int`
Metodo getter del valore di difesa dell'unità
- `+getCost() : Cost`
Metodo getter del costo dell'unità

3.12 Componente sgad::clienttier::model::observer

3.12.1 Informazioni sul package

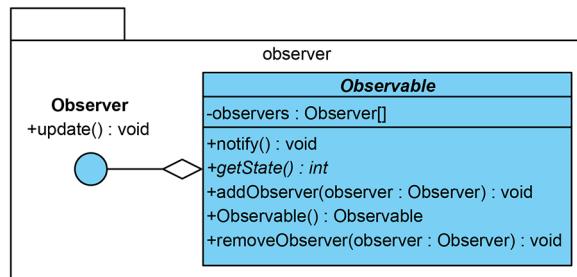


Figura 91: Diagramma della componente `sgad::clienttier::model::observer`

- **Descrizione:** componente per raccogliere le classi che realizzeranno il pattern Observer tipico dell'architettura MVC.
- **Padre:** `model`
- **Interazioni con altri componenti**
 - `generaldata`: componente del Model per la raccolta dei dati non specifici dell'utente ma relativi all'intero gioco.
 - `personaldata`: componente del model che gestisce le classi per la memorizzazione dei dati legati all'utente.
 - `context`: componente che raccoglie la classe che gestisce l'oggetto che rappresenterà l'ambiente generale di gioco.

- `graphicobjects`: componente per la gestione degli oggetti grafici che vengono rappresentati nel mondo di gioco quali widget e immagini rappresentanti il mondo di gioco (esempio: edificio).

3.12.2 Classi

3.12.2.1 sgad::clienttier::model::observer::Observable

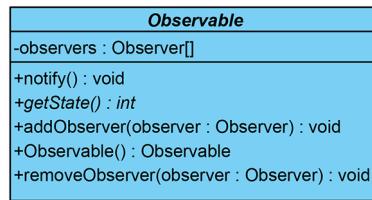


Figura 92: Diagramma della classe `sgad::clienttier::model::observer::Observable`

- **Descrizione:** classe astratta per la gestione di oggetti osservabili. Essi, al cambiamento di stato, notificheranno tutti gli oggetti osservatori che richiedono di conoscere tali cambiamenti.
 - **Utilizzo:** viene utilizzata da alcune delle classi della componente `model` per notificare un cambiamento dello stato del model a cui gli oggetti della componente `view` sono interessati per potersi rappresentare correttamente.
 - **Sottoclassi**
 - `BuildingPossession`
 - `OwnedResource`
 - `UnitInProgress`
 - `UnitPossession`
 - **Attributi**
 - `-observers : Observer[]`
Lista di osservatori che richiedono di essere notificati ad un cambiamento di stato.
 - **Metodi**
 - `+addObserver(observer : Observer) : void`
Aggiunge un osservatore alla lista degli osservatori.
- Argomenti:**
- * `observer` : osservatore che deve essere aggiunto.

- `+removeObserver(observer : Observer) : void`
Rimuove un osservatore dalla lista degli osservatori.

Argomenti:

- * `observer` : osservatore che deve essere rimosso.
- `+notify() : void`
Notifica a tutti gli osservatori che è avvenuto un aggiornamento.
- `+getState() : int`
Fornisce un valore numerico rappresentante una quantità.
- `+Observable() : Observable`
Costruttore della classe.

3.12.2.2 sgad::clienttier::model::observer::Observer

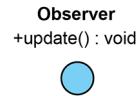


Figura 93: Diagramma della classe `sgad::clienttier::model::observer::Observer`

- **Descrizione:** interfaccia per la gestione di oggetti che osservano altri oggetti.
- **Utilizzo:** viene utilizzata ereditando questa classe e definendo il metodo esposto quando si vuole che questo oggetto sia aggiornato alla modifica di un oggetto di tipo `Observable`.
- **Sottoclassi**
 - `BuildingComponent`
 - `TextWidget`
- **Metodi**
 - `+update() : void`
Aggiorna lo stato dell'osservatore.

3.13 Componente sgad::clienttier::model::personaldata

3.13.1 Informazioni sul package

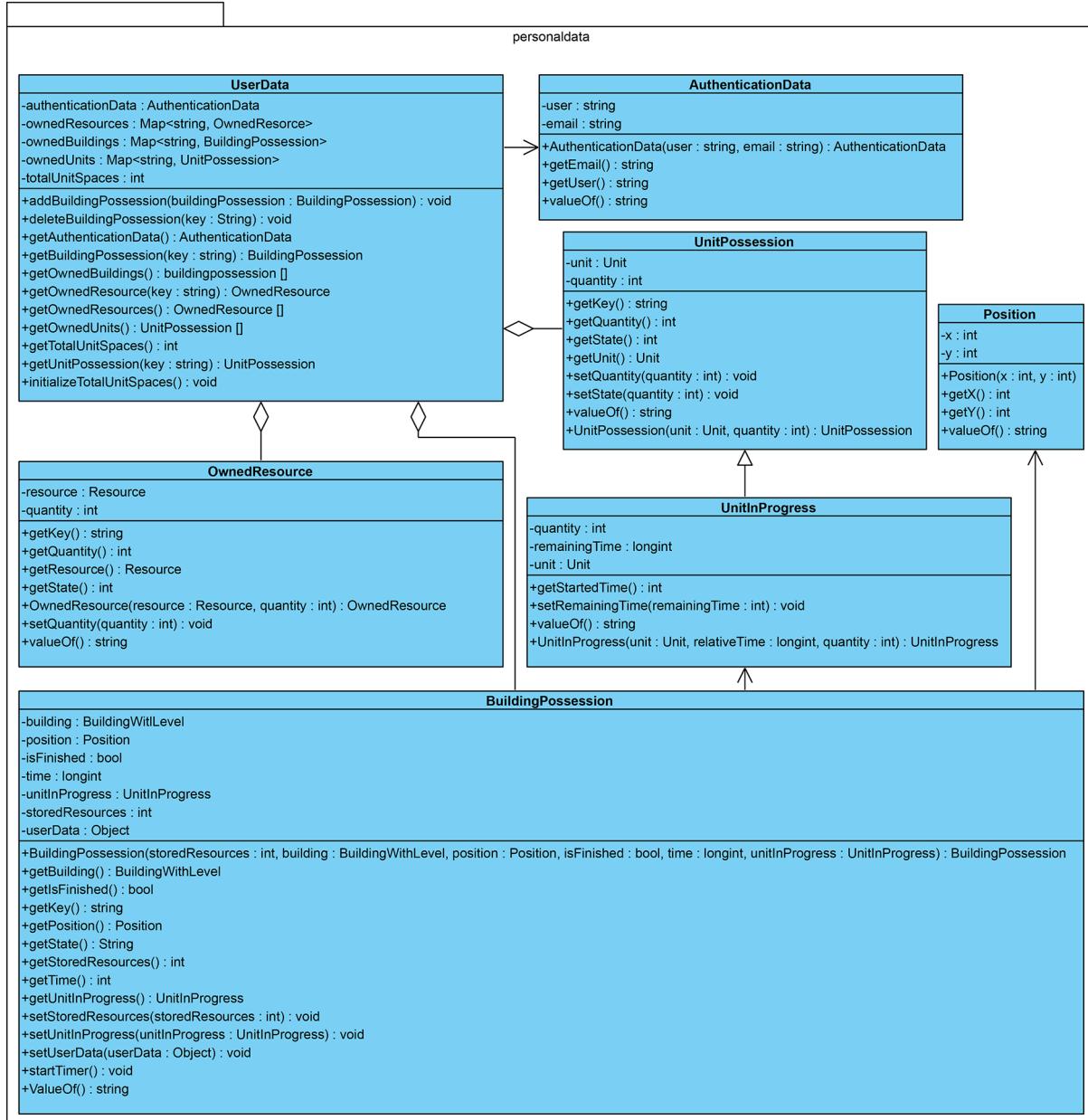


Figura 94: Diagramma della componente sgad::clienttier::model::personaldata

- Descrizione:** componente del model che gestisce le classi per la memorizzazione dei dati legati all'utente.

- **Padre:** model
- **Interazioni con altri componenti**

- **generaldata:** componente del Model per la raccolta dei dati non specifici dell'utente ma relativi all'intero gioco.
- **observer:** componente per raccogliere le classi che realizzeranno il pattern Observer tipico dell'architettura MVC.

3.13.2 Classi

3.13.2.1 sgad::clienttier::model::personaldata::AuthenticationData

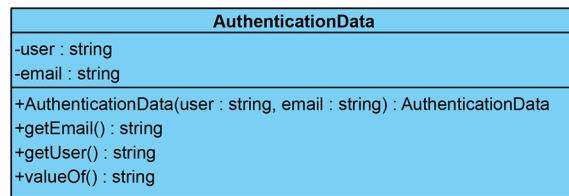


Figura 95: Diagramma della classe sgad::clienttier::model::personaldata::AuthenticationData

- **Descrizione:** classe per la gestione delle informazioni personali e di accesso di un utente.
- **Utilizzo:** viene utilizzata istanziando un oggetto che memorizza le informazioni.
- **Attributi**
 - **-email :** string
L'email dell'utente.
 - **-user :** string
Il nome utente univoco dell'utente.
- **Metodi**
 - **+AuthenticationData(email : string, user : string) : AuthenticationData**
Costruttore della classe.

Argomenti:

 - * **email** : l'email dell'utente.
 - * **user** : il nome utente univoco dell'utente.
 - **+getEmail() : string**
Metodo getter per email

- `+getUser() : string`
Metodo getter per l'identificativo utente

- `+valueOf() : string`
Ridefinizione del metodo valueOf per AuthenticationData

3.13.2.2 sgad::clienttier::model::personaldata::BuildingPossession

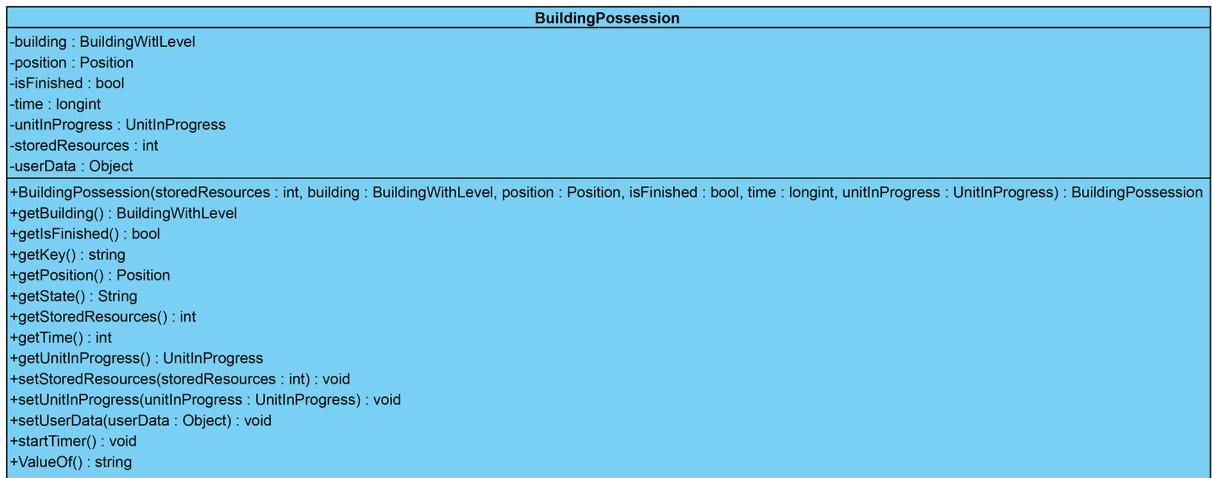


Figura 96: Diagramma della classe `sgad::clienttier::model::personaldata::BuildingPossession`

- **Descrizione:** classe per la gestione di un edificio posseduto da un utente.
- **Utilizzo:** viene utilizzata per la memorizzazione delle informazioni riguardanti un edificio posseduto da un utente.
- **Classi ereditate**
 - `Observable`
- **Attributi**
 - `-building : BuildingWithLevel`
Il tipo di costruzione.
 - `-isFinished : bool`
Indica se la costruzione dell'edificio è completa.
 - `-position : Position`
Indica le coordinate in cui l'edificio è posizionato nel mondo di gioco.
 - `-storedResources : int`
Indica la quantità di risorse immagazzinate nell'edificio.

- `-time : longint`
Indica una data con orario preciso al secondo. Se l'edificio è completamente costruito indica il momento in cui è stata effettuata l'ultima raccolta delle risorse prodotte. Se l'edificio è in costruzione indica il momento in cui la costruzione è iniziata.
- `-unitInProgress : UnitInProgress`
Indica le eventuali unità in costruzione presso l'edificio.
- `-userData : Object`
Indica i dati dell'utente.

- Metodi

- `+BuildingPossession(building : BuildingWithLevel, isFinished : bool, position : Position, storedResources : int, time : longint, unitInProgress : UnitInProgress) : BuildingPossession`
Costruttore della classe.

Argomenti:

- * `building` : il tipo di costruzione.
- * `isFinished` : indica se la costruzione dell'edificio è completa.
- * `position` : indica le coordinate in cui l'edificio è posizionato nel mondo di gioco.
- * `storedResources` : indica la quantità di risorse immagazzinate nell'edificio.
- * `time` : indica una data con orario preciso al secondo. Se l'edificio è completamente costruito indica il momento in cui è stata effettuata l'ultima raccolta delle risorse prodotte. Se l'edificio è in costruzione indica il momento in cui la costruzione è iniziata.
- * `unitInProgress` : indica le eventuali unità in costruzione presso l'edificio.

- `+getBuilding() : BuildingWithLevel`
Metodo getter per building.
- `+getIsFinished() : boolean`
Metodo getter per isFinished.
- `+getPosition() : Position`
Metodo getter per position.
- `+getTime() : int`
Metodo getter per time.

- `+getUnitInProgress() : UnitInProgress`
Metodo getter per unitinprogress.

- `+valueOf() : string`
Ridefinizione del metodo valueOf per BuildingPossession

- `+getKey() : string`
Ritorna la chiave univoca.

- `+setStoredResources(storedResources : int) : void`
Metodo setter per productedResources.

Argomenti:

- * `storedResources` : la quantità di risorse da impostare.

- `+getStoreResources() : int`
Metodo getter per productedResources.

- `+getState() : string`
Ritorna la data con orario preciso al secondo.

- `+setUnitInProgress(unitInProgress : UnitInProgress) : void`
Imposta le unità in coda nell'edificio.

Argomenti:

- * `unitInProgress` : le unità da impostare nella coda.

- `+setUserData(userData : Object) : void`
Imposta i dati dell'utente.

Argomenti:

- * `userData` : i dati dell'utente.

- `+startTimer() : void`
Avvia il timer relativo all'edificio.

3.13.2.3 sgad::clienttier::model::personaldata::OwnedResource

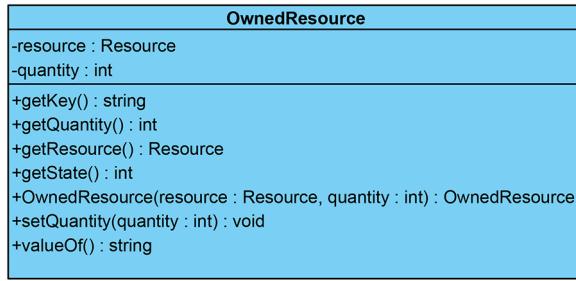


Figura 97: Diagramma della classe `sgad::clienttier::model::personaldata::-OwnedResource`

- **Descrizione:** classe che rappresenta una risorsa posseduta da un utente.
- **Utilizzo:** viene utilizzata per la memorizzazione di una risorsa posseduta da un utente.
- **Classi ereditate**
 - `Observable`
- **Attributi**
 - `-quantity : int`
La quantità di risorsa posseduta.
 - `-resource : Resource`
Il tipo di risorsa.
- **Metodi**
 - `+OwnedResource(quantity : int, resource : Resource) : OwnedResource`
Costruttore della classe.
 - Argomenti:**
 - * `quantity` : la quantità di risorsa posseduta.
 - * `resource` : il tipo di risorsa.
 - `+getQuantity() : int`
Metodo getter per quantity
 - `+getResource() : Resource`
Metodo getter per resource
 - `+valueOf() : string`
Ridefinizione del metodo `valueOf` per `OwnedResource`

– `+getKey() : string`
 Ritorna la chiave univoca della classe.

– `+setQuantity(quantity : int) : void`
 Metodo setter per quantity.

Argomenti:

* `quantity` : la quantità da impostare.

– `+getState() : int`
 Fornisce lo stato della risorsa.

3.13.2.4 sgad::clienttier::model::personaldata::Position

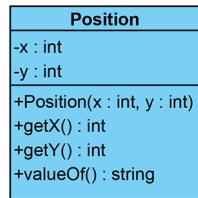


Figura 98: Diagramma della classe `sgad::clienttier::model::personaldata::Position`

- **Descrizione:** classe per la gestione di una posizione sulla griglia del villaggio.
- **Utilizzo:** viene utilizzata per memorizzare la posizione di un edificio posseduto da un utente.
- **Attributi**

– `-x : int`
 La coordinata x.

– `-y : int`
 La coordinata y.

- **Metodi**

– `+Position(x : int, y : int) : Position`
 Costruttore della classe.

Argomenti:

* `x` : la coordinata x.

* `y` : la coordinata y.

- `+getX() : int`
Metodo getter per x.
- `+getY() : int`
Metodo getter per y.
- `+valueOf() : string`
Ridefinizione del metodo valueOf per Position.

3.13.2.5 sgad::clienttier::model::personaldata::UnitPossession

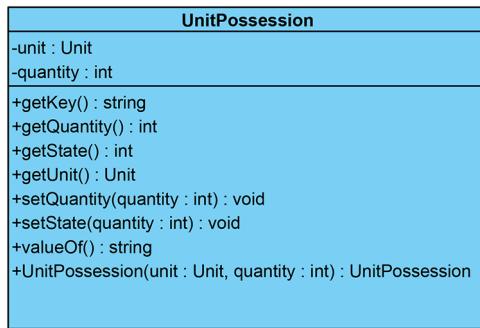


Figura 99: Diagramma della classe `sgad::clienttier::model::personaldata::-UnitPossession`

- **Descrizione:** classe per la rappresentazione di un'unità con associata una quantità.
- **Utilizzo:** viene utilizzata per memorizzare per ogni tipo di unità la quantità di queste possedute da un utente.
- **Classi ereditate**
 - `Observable`
- **Sottoclassi**
 - `UnitInProgress`
- **Attributi**
 - `-quantity : int`
La quantità di unità.
 - `-unit : Unit`
Il tipo di unità.
- **Metodi**

- `+UnitPossession(quantity : int, unit : Unit) : UnitPossession`
Costruttore della classe.

Argomenti:

* `quantity` : la quantità di unità.

* `unit` : il tipo di unità.

- `+getQuantity() : int`
Metodo getter per la quantità di unità

- `+getUnit() : Unit`
Metodo getter per il tipo di unità

- `+valueOf() : string`
Ridefinizione del metodo valueOf di UnitPossession

- `+getKey() : string`
Ritorna la chiave univoca.

- `+setState(quantity : int) : void`
Imposta il valore numerico rappresentante la quantità.

Argomenti:

* `quantity` : la quantità da impostare.

- `+setQuantity(quantity : int) : void`
Metodo setter per la quantità di unità

Argomenti:

* `quantity` : la quantità da impostare.

- `+getState() : int`
Fornisce un valore numerico rappresentante una quantità.

3.13.2.6 sgad::clienttier::model::personaldata::UnitInProgress

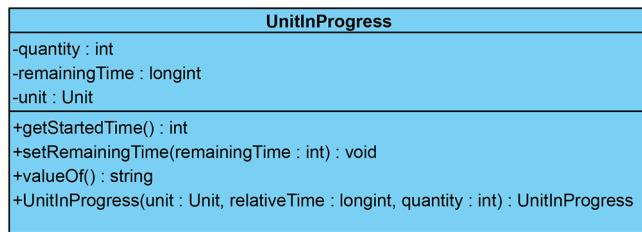


Figura 100: Diagramma della classe `sgad::clienttier::model::personaldata::-UnitInProgress`

- **Descrizione:** classe per la gestione della coda di costruzione di unità di un particolare edificio.
- **Utilizzo:** viene utilizzata per gestire le code di costruzione degli edifici completi posseduti da un particolare utente.
- **Classi ereditate**
 - `Observable`
 - `UnitPossession`
- **Attributi**
 - `-quantity : int`
La quantità di unità
 - `-remainingTime : longint`
Il tempo rimanente alla costruzione della prossima unità.
 - `-unit : Unit`
Il tipo di unità.
- **Metodi**
 - `+UnitInProgress(quantity : int, relativeTime : longint, unit : Unit) : UnitInProgress`
Costruttore della classe.

Argomenti:

 - * `quantity` : la quantità di unità.
 - * `relativeTime` : il tempo rimanente alla costruzione della prossima unità.
 - * `unit` : il tipo di unità.
 - `+getStartTime() : int`
Metodo getter per startTime

– `+valueOf() : string`
Ridefinizione del metodo valueOf per UnitInProgress

– `+setRemainingTime(remainingTime : int) : void`
Metodo setter per remainingTime.

Argomenti:

* `remainingTime` : il tempo rimanente per la prossima unità.

– `+getRemainingTime() : int`
Metodo getter per remainingTime.

3.13.2.7 sgad::clienttier::model::personaldata::UserData

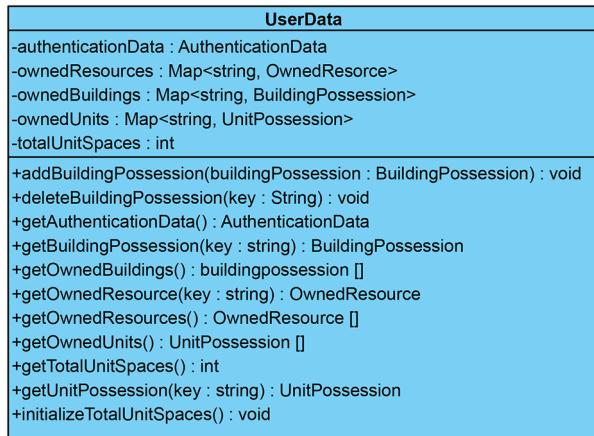


Figura 101: Diagramma della classe `sgad::clienttier::model::personaldata::-UserData`

- **Descrizione:** classe per la gestione dei dati di un utente.
- **Utilizzo:** viene utilizzata per la gestione dei dati di un utente.
- **Attributi**

- `-authenticationData : AuthenticationData`
I dati personali dell’utente.
- `-ownedBuildings : Map<string, BuildingPossession>`
Gli edifici presenti nel villaggio dell’utente.
- `-ownedResources : Map<string, OwnedResource>`
Le risorse possedute dall’utente.
- `-ownedUnits : Map<string, UnitPossession>`
Le unità possedute dall’utente.
- `-totalUnitSpaces : int`
Spazio totale utilizzabile per le unità.

- **Metodi**

- `+UserData(authenticationData : AuthenticationData, ownedBuildings : Map<string, BuildingPossession>, ownedResources : Map<string, OwnedResource>, ownedUnits : Map<string, UnitPossession>) : UserData`
Costruttore della classe.

Argomenti:

- * `authenticationData` : i dati personali dell’utente.

- * `ownedBuildings` : gli edifici presenti nel villaggio dell'utente.
 - * `ownedResources` : le risorse possedute dall'utente.
 - * `ownedUnits` : le unità possedute dall'utente.
- `+getOwnedResource(key : string) : OwnedResource`
Fornisce la risorsa posseduta in base alla chiave determinata.

Argomenti:

- * `key` : chiave identificativa della risorsa posseduta.
- `+getUnitPossession(key : string) : UnitPossession`
Ritorna la UnitPossession corrispondente alla key in input.

Argomenti:

- * `key` : chiave identificativa dell'unità posseduta.
- `+getBuildingPossession(key : string) : BuildingPossession`
Fornisce l'edificio posseduto in base alla chiave determinata.

Argomenti:

- * `key` : chiave identificativa dell'edificio posseduto.
- `+initializeTotalUnitSpaces() : void`
Inizializza il valore dei posti disponibili per le unità effettuando il calcolo.
- `+getTotalUnitSpaces() : int`
Metodo getter per l'attributo TotalUnitSpaces.
- `+getAuthenticationData() : AuthenticationData`
Controlla l'accesso ai dati personali dell'utente.
- `+getOwnedBuildings() : Array<BuildingPossession>`
Controlla l'accesso agli edifici presenti nel villaggio dell'utente.
- `+getOwnedResources() : Array<OwnedResource>`
Controlla l'accesso alle risorse possedute dall'utente.
- `+getOwnedUnits() : Array<UnitPossession>`
Controlla l'accesso alle unità possedute dall'utente.

- `+deleteBuildingPossession(key : String) : void`
Demolisce l'edificio identificato dalla chiave.

Argomenti:

- * `key` : la chiave.

- `+addBuildingPossession(buildingPossession : BuildingPossession) :- void`
Aggiunge l'edificio identificato da buildingPossession.

Argomenti:

- * `buildingPossession` : l'edificio identificato da buildingPossession.

3.14 Componente sgad::clienttier::model::userdatamanager

3.14.1 Informazioni sul package

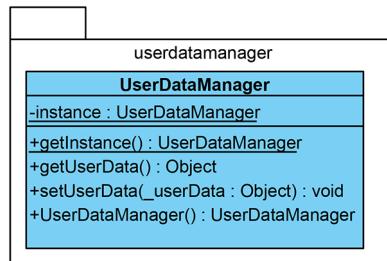


Figura 102: Diagramma della componente `sgad::clienttier::model::userdatamanager`

- **Descrizione:** componente per il mantenimento di un accesso globale ai dati dell'utente di cui si sta attualmente visualizzando il villaggio.
- **Padre:** `model`

3.14.2 Classi

3.14.2.1 `sgad::clienttier::model::userdatamanager::UserDataManager`

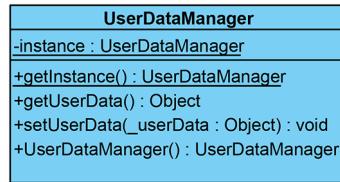


Figura 103: Diagramma della classe `sgad::clienttier::model::userdatamanager::-UserDataManager`

- **Descrizione:** classe che si occupa di mantenere l'accesso globale ai dati dell'utente di cui si sta attualmente visualizzando il villaggio.

- **Utilizzo:** viene utilizzata per accedere globalmente ai dati dell'utente. Permette inoltre di sostituire i dati dell'utente con i dati di un altro utente per modificare velocemente la rappresentazione del mondo di gioco.

- **Attributi**

- `-instance : UserDataManager`
Unica istanza dell'oggetto.

- **Metodi**

- `+UserDataManager() : UserDataManager`
Costruttore della classe.
- `+getUserData() : Object`
I dati dell'utente corrente.
- `+setUserData(_userData : Object) : void`
Imposta i dati dell'utente corrente.

Argomenti:

- * `_userData` : i dati da impostare.
- `+getInstance() : UserDataManager`
Restituisce l'istanza dell'oggetto già creata nel qual caso l'oggetto sia già stato richiesto.

3.15 Componente sgad::clienttier::view

3.15.1 Informazioni sul package

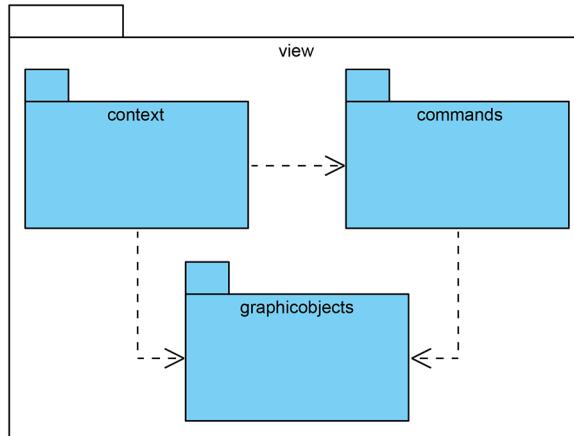


Figura 104: Diagramma della componente sgad::clienttier::view

- **Descrizione:** componente View dell'architettura MVC. Essa gestisce la visualizzazione del gioco e l'input utente all'interno del canvas HTML5.
- **Padre:** clienttier
- **Interazioni con altri componenti**
 - **controller:** componente Controller dell'architettura MVC. Essa esegue le operazioni che l'utente ha richiesto tramite la view agendo se necessario sul model data.
 - **model:** componente Model dell'architettura MVC. Essa memorizza tutti i dati di gioco dell'utente su cui la view si basa per disegnarsi.
- **Package contenuti**
 - **commands:** componente per la gestione dei comandi che vengono ricevuti dall'oggetto in cui viene rappresentato il mondo di gioco.
 - **context:** componente che raccoglie la classe che gestisce l'oggetto che rappresenterà l'ambiente generale di gioco.
 - **graphicobjects:** componente per la gestione degli oggetti grafici che vengono rappresentati nel mondo di gioco quali widget e immagini rappresentanti il mondo di gioco (esempio: edificio).

3.16 Componente sgad::clienttier::view::commands

3.16.1 Informazioni sul package

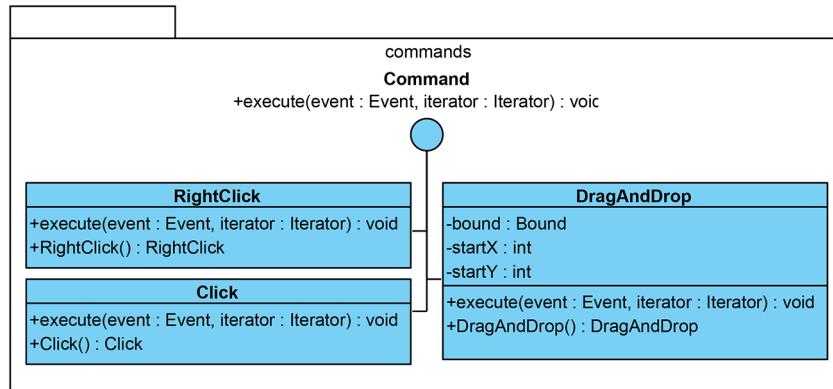


Figura 105: Diagramma della componente `sgad::clienttier::view::commands`

- **Descrizione:** componente per la gestione dei comandi che vengono ricevuti dall'oggetto in cui viene rappresentato il mondo di gioco.
- **Padre:** `view`
- **Interazioni con altri componenti**
 - `context`: componente che raccoglie la classe che gestisce l'oggetto che rappresenterà l'ambiente generale di gioco.
 - `graphicobjects`: componente per la gestione degli oggetti grafici che vengono rappresentati nel mondo di gioco quali widget e immagini rappresentanti il mondo di gioco (esempio: edificio).

3.16.2 Classi

3.16.2.1 `sgad::clienttier::view::commands::Command`



Figura 106: Diagramma della classe `sgad::clienttier::view::commands::Command`

- **Descrizione:** interfaccia per la gestione di un evento nell'area di gioco. Le sottoclassi determineranno quale evento gestire.

- **Utilizzo:** viene utilizzata per fornire un metodo comune a qualsiasi tipo di eventi che possono verificarsi nell'interazione tra utente e mondo di gioco. Le classi che realizzano l'interfaccia si occuperanno di ridefinire il metodo per gestire l'evento.

- **Sottoclassi**

- Click
- DragAndDrop
- RightClick

- **Metodi**

- `+execute(event : Event, iterator : Iterator) : void`
Gestisce il verificarsi di un determinato evento.

Argomenti:

- * `event` : l'evento verificatosi.
- * `iterator` : un iteratore per delegare l'evento all'oggetto a cui punta.
- `+Command() : Command`
Costruttore della classe.

3.16.2.2 sgad::clienttier::view::commands::Click



Figura 107: Diagramma della classe `sgad::clienttier::view::commands::Click`

- **Descrizione:** classe per la gestione di un evento, di tipo click sinistro del mouse, nell'area di gioco.
- **Utilizzo:** viene utilizzata per incapsulare la gestione di un evento di tipo click sinistro del mouse. Tale evento deve essere delegato all'effettivo oggetto grafico sul quale è avvenuto. Una volta individuato l'oggetto, verrà invocato su di esso il metodo appropriato.

Per un maggior approfondimento, si veda la sezione Gestione degli eventi.

- **Classi ereditate**

- Command

- **Metodi**

- `+execute(event : Event, iterator : Iterator) : void`
 Individua l'oggetto con minore profondità sul quale si è verificato l'evento.
 Delega poi ad esso la gestione.

Argomenti:

- * `event` : l'evento contenente le coordinate nel quale si è verificato.
- * `iterator` : l'iteratore che permette di scorrersi la collezione di elementi.
- `+Click() : Click`
 Costruttore della classe.

3.16.2.3 sgad::clienttier::view::commands::DragAndDrop



Figura 108: Diagramma della classe `sgad::clienttier::view::commands::DragAndDrop`

- **Descrizione:** classe per la gestione di un evento, di tipo drag and drop, nell'area di gioco.
- **Utilizzo:** viene utilizzata per incapsulare la gestione di un evento di tipo drag and drop. Tale evento prevede di spostare la telecamera sul mondo di gioco sfruttando la relativa azione.
- **Classi ereditate**
 - `Command`
- **Attributi**
 - `-bound : Bound`
 Mantiene i limiti alla traslazione della telecamera.
 - `-startX : int`
 La posizione x iniziale dalla quale si inizia a traslare.
 - `-startY : int`
 La posizione y iniziale dalla quale si inizia a traslare.
- **Metodi**

- `+execute(event : Event, iterator : Iterator) : void`
Trasla la posizione del bound di un spostamento pari allo spostamento del mouse.

Argomenti:

- * `event` : l'evento contenente le coordinate nel quale si è verificato.
- * `iterator` : l'iteratore può essere imposto a null. Il metodo non necessita di iteratori.
- `+DragAndDrop() : DragAndDrop`
Costruttore della classe.

3.16.2.4 sgad::clienttier::view::commands::RightClick



Figura 109: Diagramma della classe `sgad::clienttier::view::commands::RightClick`

- **Descrizione:** classe per la gestione di un evento, di tipo click destro del mouse, nell'area di gioco.
- **Utilizzo:** viene utilizzata per incapsulare la gestione di un evento di tipo click destro del mouse. Tale evento deve essere delegato all'effettivo oggetto grafico sul quale è avvenuto. Una volta individuato l'oggetto, verrà invocato su di esso il metodo appropriato.

- **Classi ereditate**

- `Command`

- **Metodi**

- `+execute(event : Event, iterator : Iterator) : void`
Individua l'oggetto con minore profondità sul quale si è verificato l'evento.
Delega poi ad esso la gestione.

Argomenti:

- * `event` : l'evento contenente le coordinate nel quale si è verificato.

- * `iterator` : l'iteratore che permette di scorrersi la collezione di elementi.

- `+RightClick() : RightClick`
Costruttore della classe.

3.17 Componente sgad::clienttier::view::context

3.17.1 Informazioni sul package

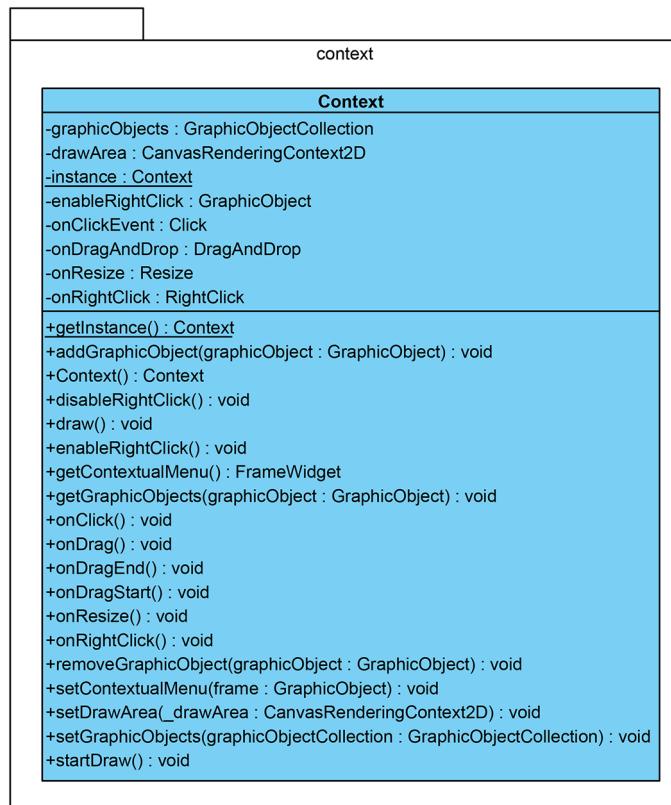


Figura 110: Diagramma della componente `sgad::clienttier::view::context`

- **Descrizione:** componente che raccoglie la classe che gestisce l'oggetto che rappresenterà l'ambiente generale di gioco.
- **Padre:** `view`
- **Interazioni con altri componenti**
 - `observer`: componente per raccogliere le classi che realizzeranno il pattern Observer tipico dell'architettura MVC.
 - `commands`: componente per la gestione dei comandi che vengono ricevuti dall'oggetto in cui viene rappresentato il mondo di gioco.
 - `graphicobject`: componente per la gestione dell'interfaccia per qualsiasi oggetto che verrà visualizzato nell'area di gioco.

3.17.2 Classi

3.17.2.1 sgad::clienttier::view::context::Context



Figura 111: Diagramma della classe `sgad::clienttier::view::context::Context`

- **Descrizione:** classe per la gestione dell'area del mondo di gioco. La classe permetterà di trasferire gli eventi verificatisi durante l'interazione con il mondo di gioco ai vari oggetti grafici.
- **Utilizzo:** viene utilizzata come pannello sul quale viene rappresentato il mondo di gioco. Quando si verifica un evento su tale oggetto, l'evento verrà trasferito ad un gestore opportuno. La classe prevede un metodo opportuno per mantenere aggiornata costantemente la rappresentazione del mondo di gioco.

Per un maggior approfondimento, si vedano le sezioni Gestione degli eventi e Visualizzazione dei menu.

- **Attributi**

- `-drawArea : CanvasRenderingContext2D`

L'area sulla quale vengono rappresentati gli oggetti del mondo di gioco.

- `-enableRightClick : GraphicObject`

Abilita il click destro.

- `-graphicObjects : GraphicObjectCollection`

La collezione degli oggetti grafici per rappresentare il mondo di gioco.

- `-instance : Context`
L'unica istanza dell'oggetto.
- `-onClickEvent : Click`
Gestisce l'evento click.
- `-onDragAndDrop : DragAndDrop`
Gestisce l'evento draganddrop.
- `-onResize : Resize`
Gestisce l'evento resize.
- `-onRightClick : RightClick`
Gestisce l'evento rightclick.

- Metodi

- `+onClick() : void`
Gestisce l'evento onClick sull'area rappresentante il mondo di gioco.
- `+onRightClick() : void`
Gestisce l'evento onRightClick sull'area rappresentante il mondo di gioco.
- `+setContextualMenu(frame : GraphicObject) : void`
Imposta il menu contestuale attualmente visualizzato.

Argomenti:

* `frame` : il frame rappresentante il menu contestuale.

- `+addGraphicObject(graphicObject : GraphicObject) : void`
Aggiunge un oggetto grafico da visualizzare.

Argomenti:

* `graphicObject` : l'oggetto grafico da aggiungere.

- `+onDragStart() : void`
Gestisce l'evento onDragStart sull'area rappresentante il mondo di gioco.
- `+onDragEnd() : void`
Gestisce l'evento onDragEnd sull'area rappresentante il mondo di gioco.
- `+onDrag() : void`
Gestisce l'evento onDrag sull'area rappresentante il mondo di gioco.
- `+onResize() : void`
Gestisce l'evento onResize sull'area rappresentante il mondo di gioco.

- `+setDrawArea(_drawArea : CanvasRenderingContext2D) : void`
Imposta l'area nella quale verrà rappresentato il mondo di gioco.

Argomenti:

 - * `_drawArea` : l'area nella quale verrà rappresentato il mondo di gioco.

- `+Context() : Context`
Costruttore della classe.

- `+setGraphicObjects(graphicObjectCollection : GraphicObjectCollection) : void`
Imposta l'elenco degli oggetti grafici attualmente visualizzati.

Argomenti:

 - * `graphicObjectCollection` : gli oggetti grafici attualmente visualizzati.

- `+removeGraphicObject(graphicObject : GraphicObject) : void`
Rimuove un oggetto grafico visualizzato.

Argomenti:

 - * `graphicObject` : l'oggetto grafico da rimuovere.

- `+getContextualMenu() : FrameWidget`
Fornisce il menu contestuale.

- `+getGraphicObjects(graphicObject : GraphicObject) : void`
Fornisce il menu contestuale.

Argomenti:

 - * `graphicObject` : l'oggetto grafico da rimuovere.

- `+startDraw() : void`
Richiede la rappresentazione del mondo di gioco.

- `+enableRightClick() : void`
Disabilita la gestione del click destro.

- `+disableRightClick() : void`
Abilita la gestione del click destro.

- `+getInstance() : Context`
Restituisce l'istanza dell'oggetto già creata nel qual caso l'oggetto sia già stato richiesto. Oppure la crea al momento se è la prima volta.

- `+draw() : void`
Rappresenta il mondo di gioco.

3.18 Componente sgad::clienttier::view::graphicobjects

3.18.1 Informazioni sul package

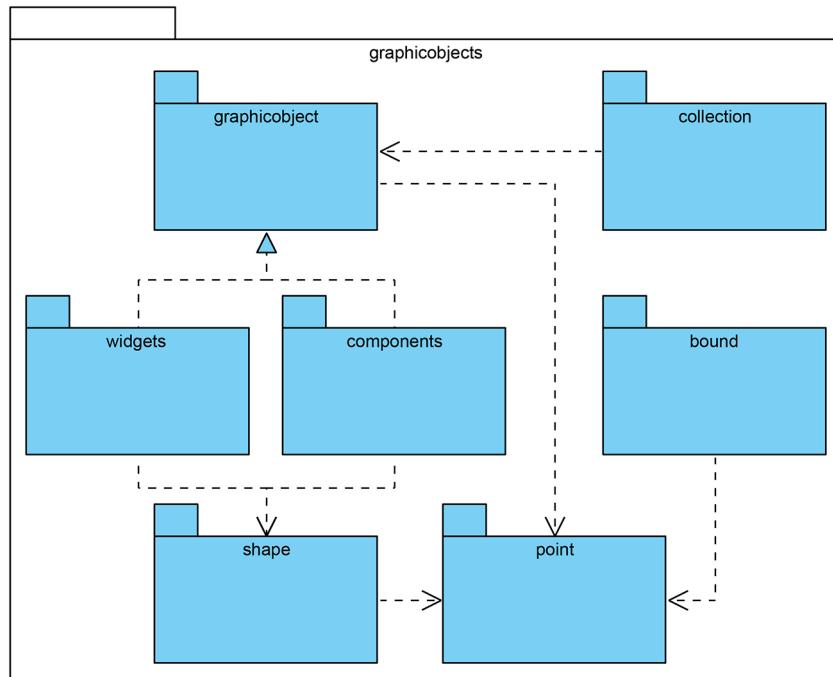


Figura 112: Diagramma della componente `sgad::clienttier::view::graphicobjects`

- **Descrizione:** componente per la gestione degli oggetti grafici che vengono rappresentati nel mondo di gioco quali widget e immagini rappresentanti il mondo di gioco (esempio: edificio).
- **Padre:** `view`
- **Interazioni con altri componenti**
 - `actions`: componente per la gestione delle possibili azioni che un utente può intraprendere durante l'interazione con il mondo di gioco.
 - `observer`: componente per raccogliere le classi che realizzeranno il pattern Observer tipico dell'architettura MVC.
 - `commands`: componente per la gestione dei comandi che vengono ricevuti dall'oggetto in cui viene rappresentato il mondo di gioco.
- **Package contenuti**

- **bound:** componente per la gestione dell'area visualizzata dall'utente rispetto all'intero mondo di gioco.
- **collection:** componente per la raccolta delle classi che gestiranno le collezioni di oggetti grafici del mondo di gioco.
- **components:** componente per la gestione degli oggetti del mondo di gioco e della relativa visualizzazione.
- **graphicobject:** componente per la gestione dell'interfaccia per qualsiasi oggetto che verrà visualizzato nell'area di gioco.
- **point:** componente per la gestione dei punti rappresentabili con due coordinate.
- **shape:** componente per la gestione delle forme degli oggetti grafici.
- **widget:** componente per la gestione dei widget per permettere all'utente di interagire con il mondo di gioco attraverso menu e finestre.

3.19 Componente sgad::clienttier::view::graphicobjects::bound

3.19.1 Informazioni sul package

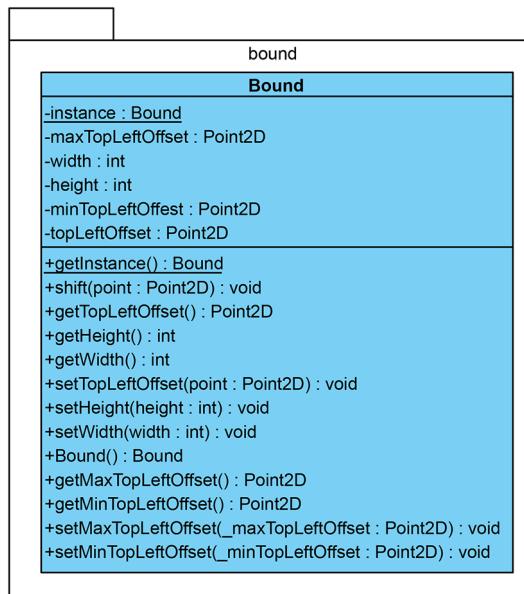


Figura 113: Diagramma della componente `sgad::clienttier::view::graphicobjects::-bound`

- **Descrizione:** componente per la gestione dell'area visualizzata dall'utente rispetto all'intero mondo di gioco.
- **Padre:** `graphicobjects`

3.19.2 Classi

3.19.2.1 sgad::clienttier::view::graphicobjects::bound::Bound

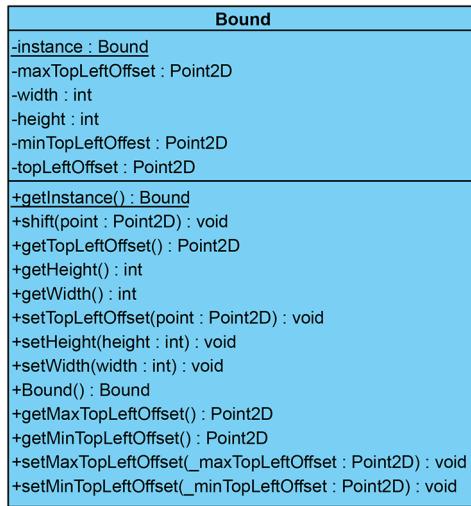


Figura 114: Diagramma della classe sgad::clienttier::view::graphicobjects::bound::Bound

- **Descrizione:** classe per la gestione dei limiti della porzione dell'area di gioco effettivamente visualizzata.
- **Utilizzo:** viene utilizzata gestire i limiti entro i quali devono essere rappresentati gli oggetti grafici del mondo di gioco.
- **Attributi**

- `-height : int`
Altezza del rettangolo.
- `-instance : Bound`
Unica istanza dell'oggetto di tipo Bound.
- `-maxTopLeftOffset : Point2D`
La posizione massima del punto in alto a sinistra.
- `-minTopLeftOffset : Point2D`
La posizione minima del punto in alto a sinistra.
- `-topLeftOffset : Point2D`
Posizione del punto in alto a sinistra del rettangolo.
- `-width : int`
Larghezza del rettangolo.

- **Metodi**

– `+getInstance() : Bound`
Fornisce l'unica istanza dell'oggetto di tipo Bound.

– `+shift(point : Point2D) : void`
Trasla il rettangolo.

Argomenti:

* `point` : punto rappresentante le coordinate di quanto verrà traslato il rettangolo.

– `+getTopLeftOffset() : Point2D`
Fornisce la posizione del punto in alto a sinistra del rettangolo.

– `+setTopLeftOffset(point : Point2D) : void`
Imposta la posizione del punto in alto a sinistra del rettangolo

Argomenti:

* `point` : la nuova posizione del punto in alto a sinistra del rettangolo.

– `+getHeight() : int`
Fornisce l'altezza del rettangolo.

– `+setHeight(height : int) : void`
Imposta l'altezza del rettangolo.

Argomenti:

* `height` : la nuova altezza del rettangolo.

– `+setWidth(width : int) : void`
Imposta la larghezza del rettangolo.

Argomenti:

* `width` : la nuova larghezza del rettangolo.

– `+getWidth() : int`
Ottiene la larghezza del rettangolo.

– `+getMinTopLeftOffset() : Point2D`
Fornisce la posizione minima del punto in alto a sinistra.

– `+setMinTopLeftOffset(_minTopLeftOffset : Point2D) : void`
Imposta la posizione minima del punto in alto a sinistra.

Argomenti:

- * `_minTopLeftOffset` : la posizione minima del punto in alto a sinistra.
- `+setMaxTopLeftOffset(_maxTopLeftOffset : Point2D) : void`
Imposta la posizione massima del punto in alto a sinistra.

Argomenti:

- * `_maxTopLeftOffset` : la posizione massima del punto in alto a sinistra.
- `+getMaxTopLeftOffset() : Point2D`
Fornisce la posizione massima del punto in alto a sinistra.
- `+Bound() : Bound`
Costruttore della classe.

3.20 Componente sgad::clienttier::view::graphicobjects::collection

3.20.1 Informazioni sul package

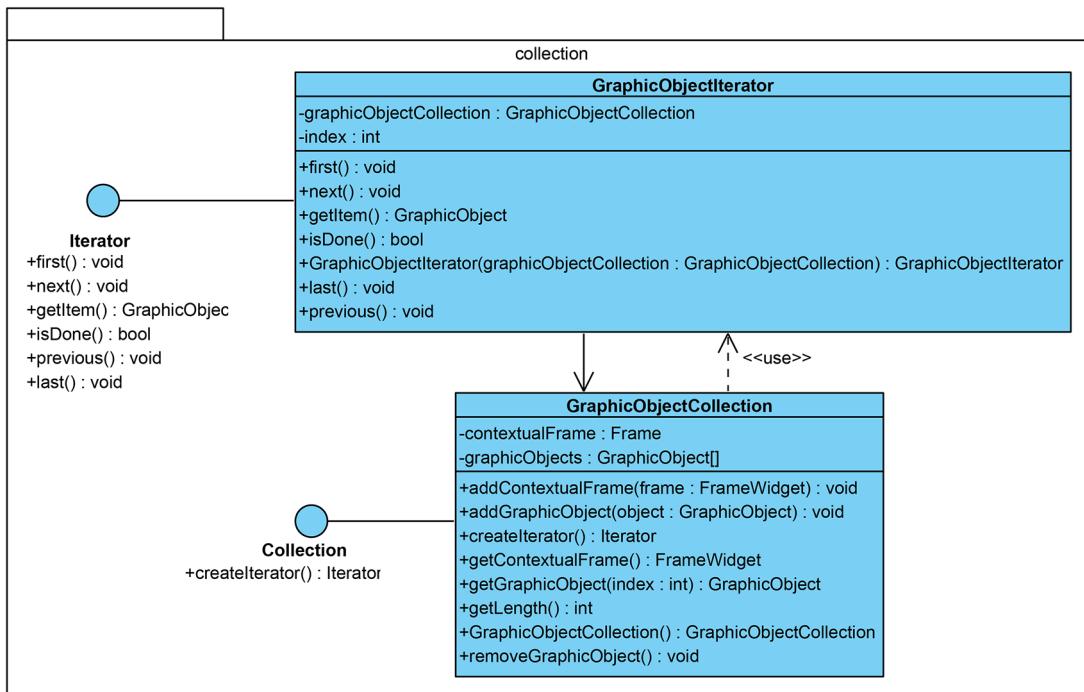


Figura 115: Diagramma della componente `sgad::clienttier::view::graphicobjects::collection`

- **Descrizione:** componente per la raccolta delle classi che gestiranno le collezioni di oggetti grafici del mondo di gioco.

- **Padre:** `graphicobjects`
- **Interazioni con altri componenti**
 - `graphicobject`: componente per la gestione dell’interfaccia per qualsiasi oggetto che verrà visualizzato nell’area di gioco.

3.20.2 Classi

3.20.2.1 `sgad::clienttier::view::graphicobjects::collection::Collection`



Figura 116: Diagramma della classe `sgad::clienttier::view::graphicobjects::collection::Collection`

- **Descrizione:** interfaccia per le classi che rappresentano collezioni di oggetti
- **Utilizzo:** viene utilizzata per fornire dei metodi standard per qualsiasi classe che rappresenta una collezione di oggetti. Le classi che realizzano l’interfaccia si occuperanno di ridefinire i metodi per accedere alla collezione.
- **Sottoclassi**
 - `GraphicObjectCollection`
- **Metodi**
 - `+createIterator() : Iterator`
Fornisce un oggetto per accedere alla collezione.

3.20.2.2 `sgad::clienttier::view::graphicobjects::collection::GraphicObjectCollection`

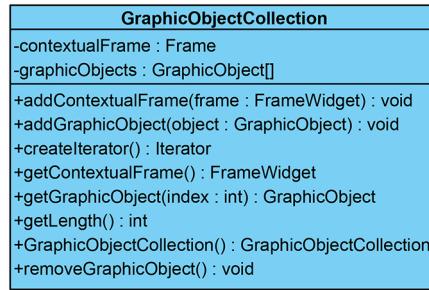


Figura 117: Diagramma della classe `sgad::clienttier::view::graphicobjects::collection::GraphicObjectCollection`

- **Descrizione:** classe per organizzare l'insieme degli oggetti grafici da rappresentare nel mondo di gioco.
- **Utilizzo:** viene utilizzata per organizzare la struttura dati che raccoglierà l'insieme degli oggetti grafici da rappresentare. Ciò permette di nascondere l'effettiva struttura dati alle classi che utilizzano questa classe. Tale astrazione è necessaria per permettere di gestire facilmente l'ordine di rappresentazione degli oggetti grafici. La collezione prevede di essere ordinata secondo la profondità decrescente dell'oggetto.

Per un maggior approfondimento, si veda la sezione Gestione degli eventi.

- **Classi ereditate**

 - `Collection`

- **Attributi**

 - `-contextualFrame : Frame`

Il frame rappresentante il menu contestuale attualmente visualizzato.

 - `-graphicObjects : GraphicObject[]`

La collezione degli oggetti grafici. Tale collezione deve essere mantenuta in ordine secondo la profondità degli oggetti in essa contenuti.

- **Metodi**

 - `+addContextualFrame(frame : FrameWidget) : void`

Imposta un frame come menu contestuale.

Argomenti:

 - * `frame` : il frame da impostare come menu contestuale.

 - `+addGraphicObject(object : GraphicObject) : void`

Aggiunge un oggetto grafico alla collezione.

Argomenti:

* `object` : l'oggetto grafico da aggiungere.

– `+getGraphicObject(index : int) : GraphicObject`

Fornisce l'oggetto grafico memorizzato nell'indice indicato.

Argomenti:

* `index` : la posizione dell'oggetto grafico.

– `+getLength() : int`

Fornisce il numero di oggetti contenuti nella collezione.

– `+removeGraphicObject() : void`

Rimuove un oggetto grafico dalla collezione.

– `+getContextualFrame() : FrameWidget`

Fornisce il frame del menu contestuale.

– `+GraphicObjectCollection() : GraphicObjectCollection`

Costruttore della classe.

– `+createIterator() : Iterator`

Fornisce un oggetto per accedere alla collezione.

3.20.2.3 `sgad::clienttier::view::graphicobjects::collection::Iterator`

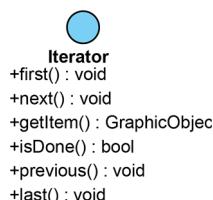


Figura 118: Diagramma della classe `sgad::clienttier::view::graphicobjects::collection::Iterator`

- **Descrizione:** interfaccia per i possibili iteratori che possono essere realizzati per accedere a classi per la gestione di aggregati di dati.
- **Utilizzo:** viene utilizzata per fornire dei metodi standard per qualsiasi tipo di iteratore. Le classi che realizzano l'interfaccia si occuperanno di ridefinire i metodi per utilizzare l'iteratore.
- **Sottoclassi**

- `GraphicObjectIterator`
- Metodi
 - `+first() : void`
Sposta l'iteratore al primo elemento della collezione.
 - `+next() : void`
Sposta l'iteratore al successivo elemento della collezione.
 - `+getItem() : GraphicObject`
Fornisce l'elemento associato alla posizione corrente.
 - `+isDone() : bool`
Indica se si è arrivato al termine dell'aggregato.
 - `+last() : void`
Sposta l'iteratore all'ultimo elemento della collezione.
 - `+previous() : void`
Sposta l'iteratore al precedente elemento della collezione.

3.20.2.4 sgad::clienttier::view::graphicobjects::collection::GraphicObjectIterator

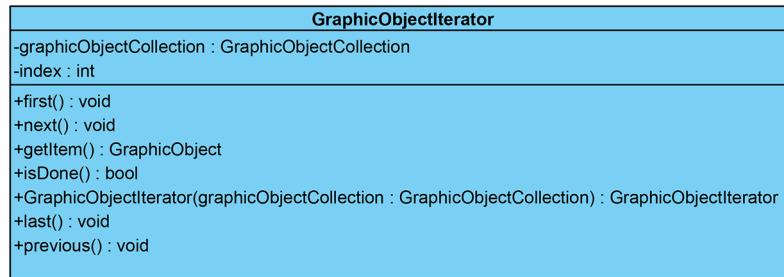


Figura 119: Diagramma della classe `sgad::clienttier::view::graphicobjects::collection::GraphicObjectIterator`

- **Descrizione:** classe per la gestione di un iteratore per accedere ad una collezione di oggetti grafici.
- **Utilizzo:** viene utilizzata per permettere di accedere ad una collezione di elementi grafici senza conoscere l'effettiva struttura di tale collezione.
Per un maggior approfondimento, si veda la sezione Gestione degli eventi.

- Classi ereditate

- Iterator

- Attributi

- -graphicObjectCollection : GraphicObjectCollection
Aggregato associato all'iteratore.

- -index : int
Posizione attuale dell'iteratore.

- Metodi

- +first() : void
Sposta l'iteratore al primo elemento della collezione. Il primo elemento sarà l'oggetto grafico con minore profondità.

- +getItem() : GraphicObject
Fornisce l'elemento associato alla posizione corrente.

- +isDone() : bool
Indica se si è arrivato al termine dell'aggregato.

- +last() : void
Sposta l'iteratore all'ultimo elemento della collezione. L'ultimo elemento dovrà essere l'elemento con maggiore profondità.

- +next() : void
Sposta l'iteratore al successivo elemento della collezione. L'elemento successivo deve essere l'elemento con profondità immediatamente successiva all'attuale elemento.

- +GraphicObjectIterator(graphicObjectCollection : GraphicObjectCollection) : GraphicObjectIterator
Costruttore della classe.

Argomenti:

- * graphicObjectCollection : la collezione di oggetti grafici sulla quale deve scorrere l'iteratore.

- +previous() : void
Sposta l'iteratore al precedente elemento della collezione. L'elemento precedente deve essere l'elemento con profondità immediatamente precedente all'attuale elemento.

3.21 Componente sgad::clienttier::view::graphicobjects::components

3.21.1 Informazioni sul package

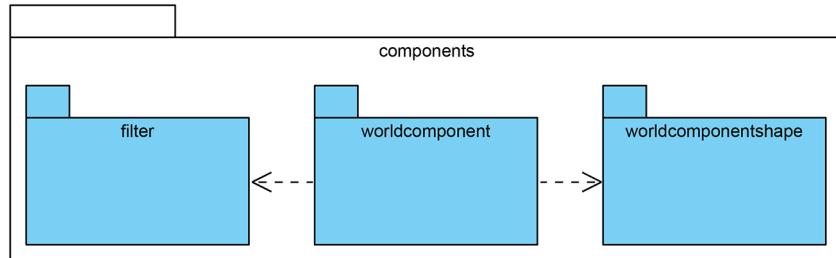


Figura 120: Diagramma della componente sgad::clienttier::view::graphicobjects::components

- **Descrizione:** componente per la gestione degli oggetti del mondo di gioco e della relativa visualizzazione.
- **Padre:** graphicobjects
- **Interazioni con altri componenti**
 - menufactory: componente per la gestione dei vari menu che possono essere visualizzati durante le interazioni con il mondo di gioco.
 - graphicobject: componente per la gestione dell'interfaccia per qualsiasi oggetto che verrà visualizzato nell'area di gioco.
 - shape: componente per la gestione delle forme degli oggetti grafici.
- **Package contenuti**
 - filter: componente per la gestione dei vari filtri con i quali vengono visualizzati gli oggetti del mondo di gioco.
 - worldcomponent: componente per la gestione dei vari oggetti, quali edifici e unità, del mondo di gioco.
 - worldcomponentshape: componente per la gestione delle forme e delle immagini per gli oggetti del mondo di gioco.

3.22 Componente sgad::clienttier::view::graphicobjects::components::filter

3.22.1 Informazioni sul package

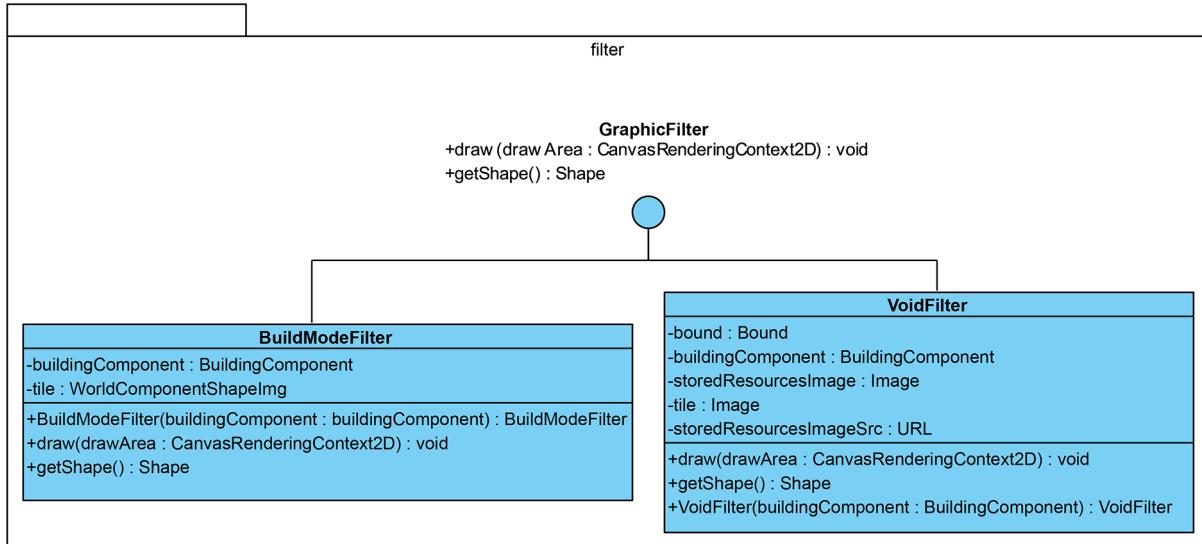


Figura 121: Diagramma della componente `sgad::clienttier::view::graphicobjects::components::filter`

- **Descrizione:** componente per la gestione dei vari filtri con i quali vengono visualizzati gli oggetti del mondo di gioco.
- **Padre:** components

3.22.2 Classi

3.22.2.1 sgad::clienttier::view::graphicobjects::components::filter::GraphicFilter



Figura 122: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::filter::GraphicFilter`

- **Descrizione:** interfaccia per i possibili filtri che possono essere applicati ad un edificio.

- **Utilizzo:** viene utilizzata per fornire un metodo comune a qualsiasi tipo di filtro che può essere utilizzato. Le classi che realizzano l'interfaccia si occuperanno di ridefinire il metodo per gestire il filtro.

- **Sottoclassi**

- `BuildModeFilter`
- `VoidFilter`

- **Metodi**

- `+draw(drawArea : CanvasRenderingContext2D) : void`
Rappresenta l'oggetto grafico all'interno del mondo di gioco.

Argomenti:

- * `drawArea` : area di disegno sulla quale verrà disegnato l'oggetto.
- `+getShape() : Shape`
Fornisce il poligono che rappresenta l'oggetto grafico.

3.22.2.2 sgad::clienttier::view::graphicobjects::components::filter::BuildModeFilter

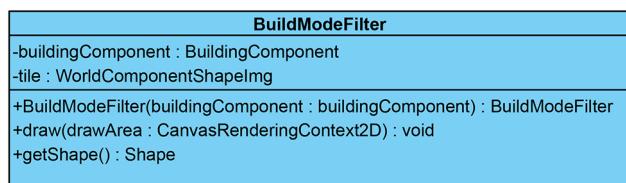


Figura 123: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::filter::BuildModeFilter`

- **Descrizione:** classe per la gestione del filtro quando viene selezionata la modalità costruzione.
- **Utilizzo:** viene utilizzata per encapsulare l'algoritmo per la rappresentazione di una costruzione quando l'utente entra in modalità costruzione.
- **Classi ereditate**
 - `GraphicFilter`
- **Attributi**
 - `-buildingComponent : BuildingComponent`
Oggetto grafico rappresentante l'edificio all'interno del mondo di gioco.

- `-tile : WorldComponentShapeImg`

Il poligono e l'immagine rappresentante l'edificio in modo tale da non ostruire la visuale.

- Metodi

- `+draw(drawArea : CanvasRenderingContext2D) : void`

Rappresenta l'oggetto grafico nel mondo di gioco in modo tale da non ostruire la visuale delle caselle.

Argomenti:

* `drawArea` : area di disegno sulla quale verrà disegnato l'oggetto.

- `+getShape() : Shape`

Fornisce il poligono rappresentante l'oggetto grafico.

- `+BuildModeFilter(buildingComponent : buildingComponent) : BuildModeFilter`

Costruttore della classe.

Argomenti:

* `buildingComponent` : l'oggetto grafico rappresentante l'edificio nel mondo di gioco.

3.22.2.3 sgad::clienttier::view::graphicobjects::components::filter::VoidFilter

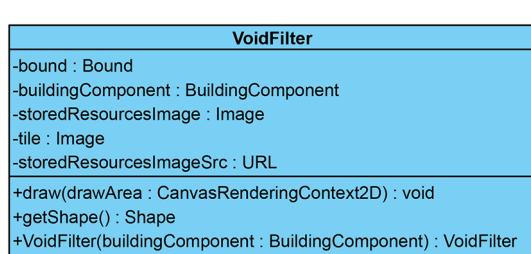


Figura 124: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::filter::VoidFilter`

- **Descrizione:** classe per la gestione di un filtro che non deve modificare la rappresentazione dell'edificio.
- **Utilizzo:** viene utilizzata per gestire la rappresentazione standard di un edificio nell'area di gioco.
- **Classi ereditate**

- `GraphicFilter`
- **Attributi**
 - `-bound : Bound`
L'unica istanza dell'oggetto di tipo `Bound`.
 - `-buildingComponent : BuildingComponent`
L'oggetto grafico rappresentante un edificio all'interno dell'area grafica.
 - `-storedResourcesImage : Image`
L'immagine rappresentante le risorse collezionate.
 - `-storedResourcesImageSrc : URL`
Il path dove reperire l'immagine.
 - `-tile : Image`
L'immagine rappresentante l'oggetto grafico associato alla casella.
- **Metodi**
 - `+draw(drawArea : CanvasRenderingContext2D) : void`
Rappresenta l'oggetto grafico nel mondo di gioco senza applicare alcun filtro.
 - **Argomenti:**
 - * `drawArea` : l'area in cui deve essere rappresentato l'oggetto grafico.
 - `+getShape() : Shape`
Fornisce la forma dell'oggetto grafico senza alcuna deformazione o cambiamento.
 - `+VoidFilter(buildingComponent : BuildingComponent) : VoidFilter`
Costruttore della classe.
 - **Argomenti:**
 - * `buildingComponent` : l'oggetto grafico rappresentante l'edificio nel mondo di gioco.

3.23 Componente sgad::clienttier::view::graphicobjects::components::worldcomponent

3.23.1 Informazioni sul package

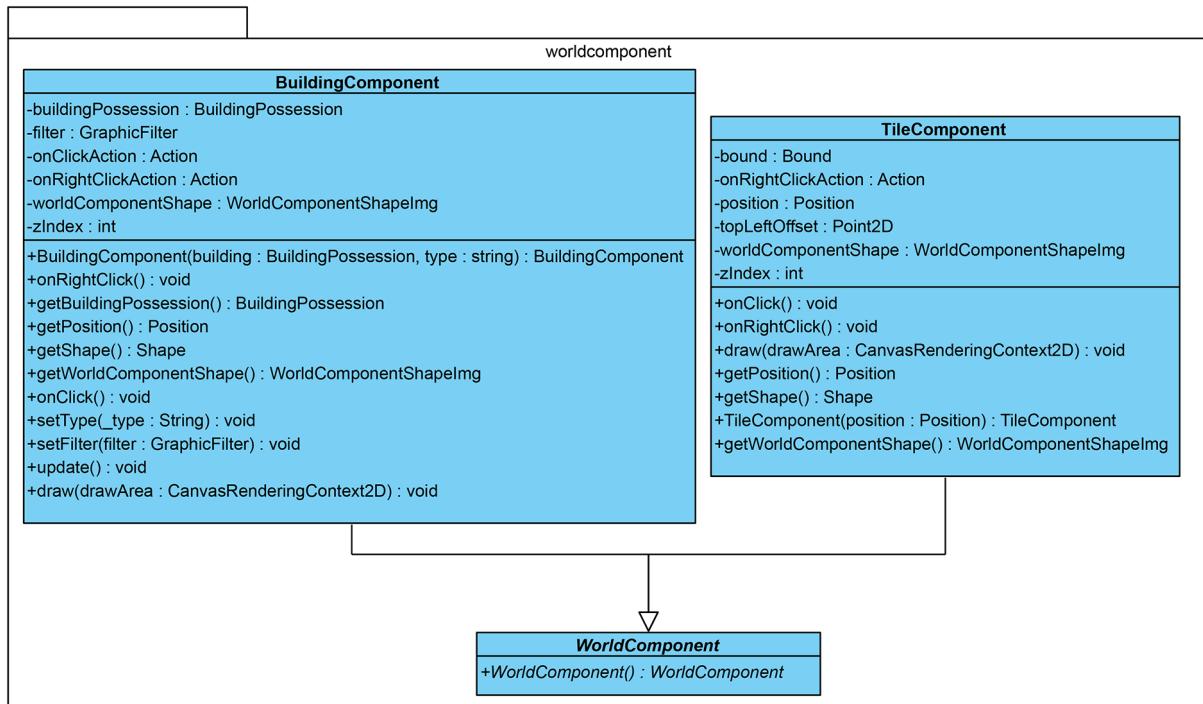


Figura 125: Diagramma della componente `sgad::clienttier::view::graphicobjects::components::worldcomponent`

- **Descrizione:** componente per la gestione dei vari oggetti, quali edifici e unità, del mondo di gioco.
- **Padre:** `components`
- **Interazioni con altri componenti**
 - `worldcomponentshape`: componente per la gestione delle forme e delle immagini per gli oggetti del mondo di gioco.

3.23.2 Classi

3.23.2.1 `sgad::clienttier::view::graphicobjects::components::worldcomponent::WorldComponent`

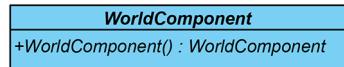


Figura 126: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponent::WorldComponent`

- **Descrizione:** classe astratta per la gestione di un qualsiasi edificio o casella all'interno del mondo di gioco.
- **Utilizzo:** viene utilizzata per fornire dei metodi comuni a tutti gli edifici e caselle che verranno creati all'interno del mondo di gioco. Tale classe contiene il riferimento ad un oggetto di tipo `WorldComponentShapeFactory` per ottenere degli oggetti di tipo `WorldComponentShapeImg`.
- **Classi ereditate**
 - `GraphicObject`
- **Sottoclassi**
 - `BuildingComponent`
 - `TileComponent`
- **Metodi**
 - `+WorldComponent() : WorldComponent`
Costruttore della classe.

3.23.2.2 `sgad::clienttier::view::graphicobjects::components::worldcomponent::BuildingComponent`

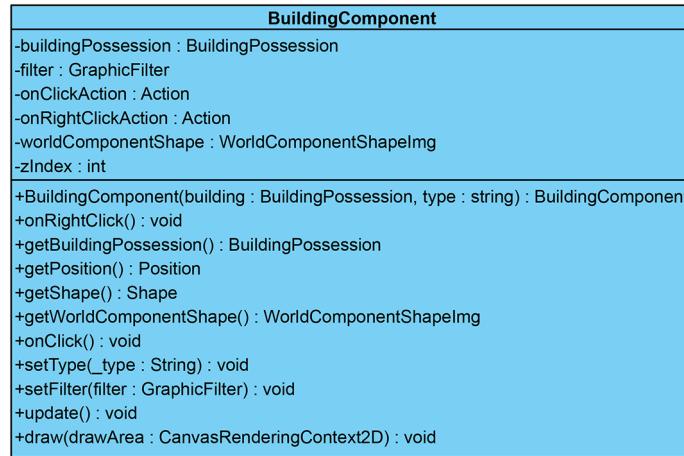


Figura 127: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponent::BuildingComponent`

- **Descrizione:** classe per la rappresentazione di un edificio generico all'interno dell'area di gioco.
- **Utilizzo:** viene utilizzata per fornire un comportamento comune ai possibili edifici del mondo di gioco. Utilizza il modello dei dati per determinare quale sia il tipo di edificio da rappresentare e il comportamento associato.
- **Classi ereditate**
 - `Observer`
 - `WorldComponent`
- **Attributi**
 - `-buildingPossession : BuildingPossession`
L'edificio che deve essere rappresentato.
 - `-filter : GraphicFilter`
Il filtro grafico che deve essere applicato alla rappresentazione.
 - `-onClickAction : Action`
L'azione da eseguire al verificarsi del click sulla componente.
 - `-onRightClickAction : Action`
L'azione da eseguire al verificarsi del rightclick sulla componente.
 - `-worldComponentShape : WorldComponentShapeImg`
La forma della componente che deve essere rappresentata.
 - `-zIndex : int`
Coordinata z dell'edificio.
- **Metodi**

- `+BuildingComponent(building : BuildingPossession, type : string) :- BuildingComponent`
Costruttore della classe.

Argomenti:

* `building` : l'edificio che deve essere rappresentato.

* `type` : tipo di edificio da rappresentare.

- `+setFilter(filter : GraphicFilter) : void`
Imposta il filtro che deve essere applicato all'oggetto rappresentante un edificio.

Argomenti:

* `filter` : il filtro da impostare.

- `+draw(drawArea : CanvasRenderingContext2D) : void`
Rappresenta l'oggetto grafico nel mondo di gioco.

Argomenti:

* `drawArea` : l'area in cui deve essere rappresentato l'oggetto grafico.

- `+onClick() : void`
Esegue l'azione associata al verificarsi dell'evento click.

- `+getBuildingPossession() : BuildingPossession`
Fornisce i dati dell'edificio che viene rappresentato.

- `+onRightClick() : void`
Esegue l'azione associata al verificarsi dell'evento click destro.

- `+getShape() : Shape`
Fornisce il poligono che rappresenta l'oggetto grafico.

- `+getWorldComponentShape() : WorldComponentShapeImg`
Fornisce l'immagine e il poligono per rappresentare l'oggetto del mondo di gioco.

- `+setType(_type : String) : void`
Imposta il tipo della componente.

Argomenti:

* `_type` : il tipo della classe.

- `+getPosition() : Position`
Fornisce la posizione dell'edificio che viene rappresentato.
- `+update() : void`
Aggiorna il tipo di edificio.

3.23.2.3 sgad::clienttier::view::graphicobjects::components::worldcomponent::-TileComponent

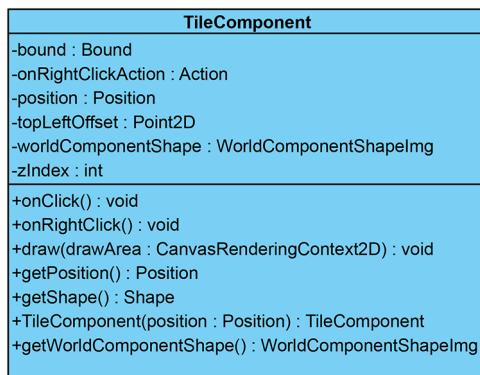


Figura 128: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponent::-TileComponent`

- **Descrizione:** classe per la rappresentazione di una casella del mondo di gioco sulla quale non è stato costruito alcun edificio.
- **Utilizzo:** viene utilizzata per rappresentare una casella libera nel mondo di gioco.
- **Classi ereditate**
 - `WorldComponent`
- **Attributi**
 - `-bound : Bound`
Il riferimento alle informazioni della telecamera.
 - `-onRightClickAction : Action`
L'azione da eseguire al verificarsi dell'evento rightclick.
 - `-position : Position`
La posizione associata alla casella.
 - `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.
 - `-worldComponentShape : WorldComponentShapeImg`
La forma della casella da rappresentare nel mondo di gioco.

- `-zIndex : int`
Coordinata z della casella.

- **Metodi**

- `+TileComponent(position : Position) : TileComponent`
Costruttore della classe.

Argomenti:

- * `position` : la posizione in cui deve essere costruita la casella.

- `+draw(drawArea : CanvasRenderingContext2D) : void`
Rappresenta l'oggetto grafico nel mondo di gioco.

Argomenti:

- * `drawArea` : l'area in cui rappresentare l'oggetto grafico.

- `+onClick() : void`
Invoca la relativa azione associata all'evento click del mouse.

- `+onRightClick() : void`
Invoca l'azione da eseguire alla pressione del tasto destro del mouse sull'oggetto grafico.

- `+getShape() : Shape`
Fornisce il poligono rappresentante la casella come oggetto grafico.

- `+getWorldComponentShape() : WorldComponentShapeImg`
Fornisce il poligono e l'immagine rappresentante la casella nel mondo di gioco.

- `+getPosition() : Position`
Fornisce la posizione associata alla casella.

3.24 Componente sgad::clienttier::view::graphicobjects::components::worldcomponentshape

3.24.1 Informazioni sul package

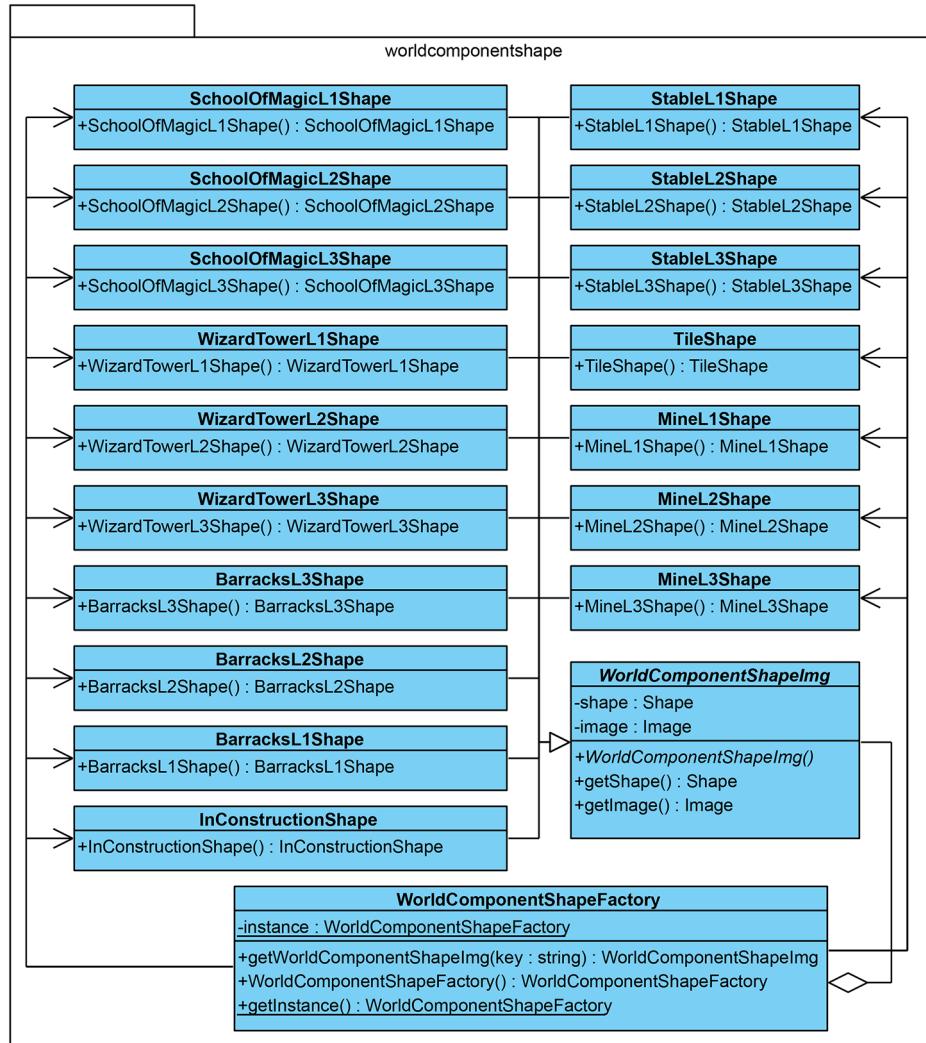


Figura 129: Diagramma della componente `sgad::clienttier::view::graphicobjects::components::worldcomponentshape`

- **Descrizione:** componente per la gestione delle forme e delle immagini per gli oggetti del mondo di gioco.
- **Padre:** `components`
- **Interazioni con altri componenti**
 - `worldcomponent`: componente per la gestione dei vari oggetti, quali edifici e unità, del mondo di gioco.

3.24.2 Classi

3.24.2.1 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WorldComponentShapeFactory

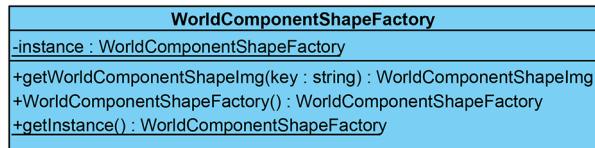


Figura 130: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WorldComponentShapeFactory

- **Descrizione:** classe per la gestione del controllo ai vari oggetti di tipo WorldComponentShapeImg.
- **Utilizzo:** viene utilizzata per fornire un determinato oggetto di tipo WorldComponentShapeImg restituendo un riferimento a tale oggetto se già esistente oppure creandolo al momento.
- **Attributi**

- -instance : WorldComponentShapeFactory
Unica istanza dell'oggetto creato.

- **Metodi**

- +getWorldComponentShapeImg(key : string) : WorldComponentShapeImg
Fornisce l'oggetto associato alla relativa chiave.

Argomenti:

- * **key** : chiave per accedere all'oggetto.
- +getInstance() : WorldComponentShapeFactory
Restituisce l'istanza dell'oggetto già creata nel qual caso l'oggetto sia già stato richiesto. Oppure la crea al momento se è la prima volta.
- +WorldComponentShapeFactory() : WorldComponentShapeFactory
Costruttore della classe.

3.24.2.2 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WorldComponentShapeImg

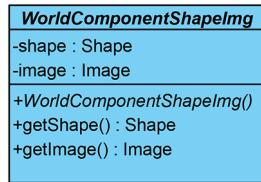


Figura 131: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WorldComponentShapeImg`

- **Descrizione:** classe astratta per le possibili forme degli edifici e delle unità che possono essere rappresentate all'interno del mondo di gioco.
- **Utilizzo:** viene utilizzata per fornire un metodo comune a tutti i tipi di forme che possono essere rappresentate. Le classi che ereditano dalla classe si occuperanno di ridefinire il metodo per essere rappresentate.
- **Sottoclassi**
 - `BarracksL1Shape`
 - `BarracksL2Shape`
 - `BarracksL3Shape`
 - `InConstructionShape`
 - `MineL1Shape`
 - `MineL2Shape`
 - `MineL3Shape`
 - `SchoolOfMagicL1Shape`
 - `SchoolOfMagicL2Shape`
 - `SchoolOfMagicL3Shape`
 - `StableL1Shape`
 - `StableL2Shape`
 - `StableL3Shape`
 - `TileShape`
 - `WizardTowerL1Shape`
 - `WizardTowerL2Shape`
 - `WizardTowerL3Shape`
- **Attributi**
 - `-image : Image`
L'immagine rappresentante l'oggetto grafico.
 - `-shape : Shape`
Il poligono che rappresenta l'oggetto grafico.

- Metodi

- `+getShape() : Shape`
Fornisce il poligono rappresentante l'oggetto grafico.
- `+WorldComponentShapeImg() : WorldComponentShapeImg`
Costruttore della classe.
- `+getImage() : Image`
Fornisce l'immagine rappresentante l'oggetto grafico.

3.24.2.3 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL1Shape

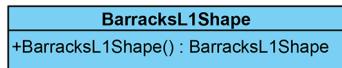


Figura 132: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL1Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una caserma di livello 1.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una caserma di livello 1.
- **Classi ereditate**
 - `WorldComponentShapeImg`
- **Metodi**
 - `+BarracksL1Shape() : BarracksL1Shape`
Costruttore della classe.

3.24.2.4 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL2Shape

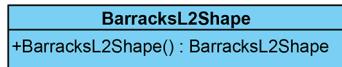


Figura 133: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL2Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una caserma di livello 2.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una caserma di livello 2.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +BarracksL2Shape() : BarracksL2Shape
Costruttore della classe.

3.24.2.5 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL3Shape

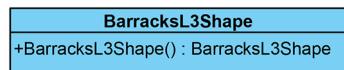


Figura 134: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL3Shape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una caserma di livello 3.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una caserma di livello 3.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +BarracksL3Shape() : BarracksL3Shape
Costruttore della classe.

3.24.2.6 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::InConstructionShape

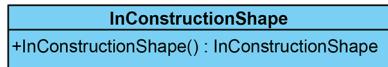


Figura 135: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::InConstructionShape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di un edificio in costruzione.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare un edificio in costruzione.
- **Classi ereditate**
 - `WorldComponentShapeImg`
- **Metodi**
 - `+InConstructionShape() : InConstructionShape`
Costruttore della classe.

3.24.2.7 `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL1Shape`

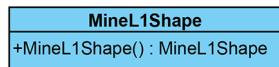


Figura 136: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL1Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una miniera di livello 1.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una miniera di livello 1.
- **Classi ereditate**
 - `WorldComponentShapeImg`
- **Metodi**
 - `+MineL1Shape() : MineL1Shape`
Costruttore della classe.

3.24.2.8 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL2Shape

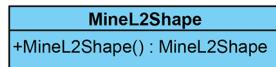


Figura 137: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL2Shape

- **Descrizione:** classe per la memorizzazione dell’immagine e del poligono per la rappresentazione di una miniera di livello 2.
- **Utilizzo:** viene utilizzata per mantenere un’unica volta in tutto il sistema i dati comuni per rappresentare una miniera di livello 2.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +MineL2Shape() : MineL2Shape
Costruttore della classe.

3.24.2.9 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL3Shape



Figura 138: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL3Shape

- **Descrizione:** classe per la memorizzazione dell’immagine e del poligono per la rappresentazione di una miniera di livello 3.
- **Utilizzo:** viene utilizzata per mantenere un’unica volta in tutto il sistema i dati comuni per rappresentare una miniera di livello 3.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**

- `+MineL3Shape()` : `MineL3Shape`
Costruttore della classe.

3.24.2.10 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::SchoolOfMagicL1Shape

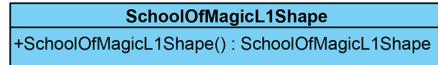


Figura 139: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::SchoolOfMagicL1Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una scuola di magia di livello 1.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una scuola di magia di livello 1.
- **Classi ereditate**
 - `WorldComponentShapeImg`
- **Metodi**
 - `+SchoolOfMagicL1Shape()` : `SchoolOfMagicL1Shape`
Costruttore della classe.

3.24.2.11 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::SchoolOfMagicL2Shape

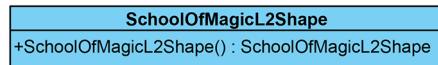


Figura 140: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::SchoolOfMagicL2Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una scuola di magia di livello 2.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una scuola di magia di livello 2.
- **Classi ereditate**

- WorldComponentShapeImg
- Metodi
 - +SchoolOfMagicL2Shape() : SchoolOfMagicL2Shape
Costruttore della classe.

3.24.2.12 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::SchoolOfMagicL3Shape



Figura 141: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::SchoolOfMagicL3Shape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una scuola di magia di livello 3.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una scuola di magia di livello 3.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +SchoolOfMagicL3Shape() : SchoolOfMagicL3Shape
Costruttore della classe.

3.24.2.13 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::StableL1Shape

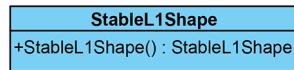


Figura 142: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::StableL1Shape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una stalla di livello 1.

- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una stalla di livello 1.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +StableL1Shape() : StableL1Shape
Costruttore della classe.

3.24.2.14 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::StableL2Shape

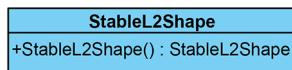


Figura 143: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::StableL2Shape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una stalla di livello 2.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una stalla di livello 2.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +StableL2Shape() : StableL2Shape
Costruttore della classe.

3.24.2.15 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::StableL3Shape

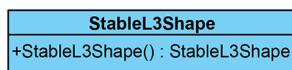


Figura 144: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::StableL3Shape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una stalla di livello 3.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una stalla di livello 3.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +StableL3Shape() : StableL3Shape
Costruttore della classe.

3.24.2.16 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::TileShape



Figura 145: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::TileShape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una casella.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una casella.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +TileShape() : TileShape
Costruttore della classe.

3.24.2.17 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WizardTowerL1Shape

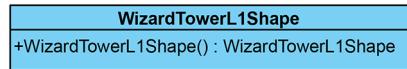


Figura 146: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WizardTowerL1Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una torre dello stregone di livello 1.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una torre dello stregone di livello 1.
- **Classi ereditate**
 - `WorldComponentShapeImg`
- **Metodi**
 - `+WizardTowerL1Shape() : WizardTowerL1Shape`
Costruttore della classe.

3.24.2.18 `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WizardTowerL2Shape`

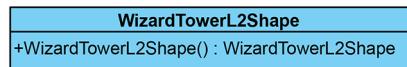


Figura 147: Diagramma della classe `sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WizardTowerL2Shape`

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una torre dello stregone di livello 2.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una torre dello stregone di livello 2.
- **Classi ereditate**
 - `WorldComponentShapeImg`
- **Metodi**
 - `+WizardTowerL2Shape() : WizardTowerL2Shape`
Costruttore della classe.

3.24.2.19 sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WizardTowerL3Shape

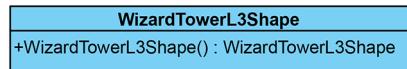


Figura 148: Diagramma della classe sgad::clienttier::view::graphicobjects::components::worldcomponentshape::WizardTowerL3Shape

- **Descrizione:** classe per la memorizzazione dell'immagine e del poligono per la rappresentazione di una torre dello stregone di livello 3.
- **Utilizzo:** viene utilizzata per mantenere un'unica volta in tutto il sistema i dati comuni per rappresentare una torre dello stregone di livello 3.
- **Classi ereditate**
 - WorldComponentShapeImg
- **Metodi**
 - +WizardTowerL3Shape() : WizardTowerL3Shape
Costruttore della classe.

3.25 Componente sgad::clienttier::view::graphicobjects::graphicobject

3.25.1 Informazioni sul package

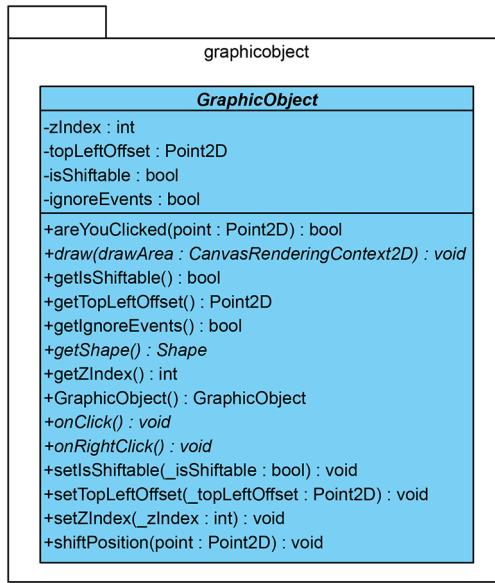


Figura 149: Diagramma della componente `sgad::clienttier::view::graphicobjects::-graphicobject`

- **Descrizione:** componente per la gestione dell’interfaccia per qualsiasi oggetto che verrà visualizzato nell’area di gioco.
- **Padre:** `graphicobjects`
- **Interazioni con altri componenti**
 - `context`: componente che raccoglie la classe che gestisce l’oggetto che rappresenterà l’ambiente generale di gioco.
 - `collection`: componente per la raccolta delle classi che gestiranno le collezioni di oggetti grafici del mondo di gioco.
 - `components`: componente per la gestione degli oggetti del mondo di gioco e della relativa visualizzazione.
 - `widget`: componente per la gestione dei widget per permettere all’utente di interagire con il mondo di gioco attraverso menu e finestre.

3.25.2 Classi

3.25.2.1 `sgad::clienttier::view::graphicobjects::graphicobject::GraphicObject`

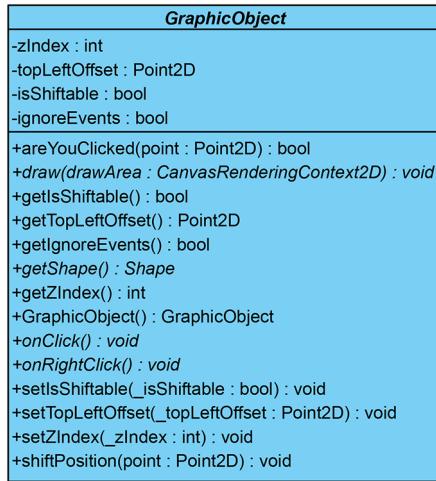


Figura 150: Diagramma della classe `sgad::clienttier::view::graphicobjects::graphicobject::GraphicObject`

- **Descrizione:** classe astratta per la rappresentazione di un qualsiasi oggetto grafico all'interno dell'area di gioco.
- **Utilizzo:** viene utilizzata per fornire dei metodi comuni per interagire con un oggetto grafico. Le varie sottoclassi si occuperanno di realizzare tali metodi.

Per un maggior approfondimento, si veda la sezione Gestione degli eventi.

- **Sottoclassi**

- `WorldComponent`
- `Widget`

- **Attributi**

- `-ignoreEvents : bool`
Indica se l'oggetto ignora gli eventi, true, o se li cattura, false.
- `-isShiftable : bool`
Indica se l'oggetto può essere traslato o meno.
- `-topLeftOffset : Point2D`
Indica la posizione in alto a sinistra del rettangolo che contiene l'oggetto grafico.
- `-zIndex : int`
Profondità dell'oggetto grafico.

- **Metodi**

- `+areYouClicked(point : Point2D) : bool`
Controlla se l'oggetto è stato cliccato.

Argomenti:

* `point` : punto rappresentante la posizione del mouse dove è avvenuto l'evento.

– `+shiftPosition(point : Point2D) : void`

Trasla l'oggetto di uno spostamento pari alla nuova posizione. Viene traslato solamente se può essere traslato.

Argomenti:

* `point` : posizione indicante lo spostamento della traslazione.

– `+draw(drawArea : CanvasRenderingContext2D) : void`

Disegna l'oggetto nell'area di disegno richiesta.

Argomenti:

* `drawArea` : area di disegno sulla quale verrà disegnato l'oggetto.

– `+setIsShiftable(_isShiftable : bool) : void`

Imposta se l'oggetto può essere traslato o meno.

Argomenti:

* `_isShiftable` : valore per indicare se l'oggetto può essere traslato o meno.

– `+onClick() : void`

Esegue un'azione al verificarsi dell'evento click sull'oggetto.

– `+onRightClick() : void`

Esegue un'azione al verificarsi dell'evento rightclick sull'oggetto.

– `+getZIndex() : int`

Fornisce la profondità dell'oggetto.

– `+getShape() : Shape`

Fornisce il poligono che rappresenta l'oggetto grafico.

– `+getTopLeftOffset() : Point2D`

Fornisce la posizione in alto a sinistra del rettangolo che contiene l'oggetto grafico.

– `+getIsShiftable() : bool`

Fornisce se l'oggetto può essere traslato o meno.

– `+setZIndex(_zIndex : int) : void`

Imposta la profondità dell'oggetto grafico.

Argomenti:

- * `_zIndex` : la profondità dell'oggetto grafico.
- `+setTopLeftOffset(_topLeftOffset : Point2D) : void`
Imposta la posizione in alto a sinistra del rettangolo che contiene l'oggetto grafico.

Argomenti:

- * `_topLeftOffset` : la posizione in alto a sinistra del rettangolo che contiene l'oggetto grafico.
- `+getIgnoreEvents() : bool`
Fornisce se l'oggetto ignora gli eventi o li cattura.
- `+GraphicObject() : GraphicObject`
Costruttore della classe.

3.26 Componente sgad::clienttier::view::graphicobjects::point

3.26.1 Informazioni sul package

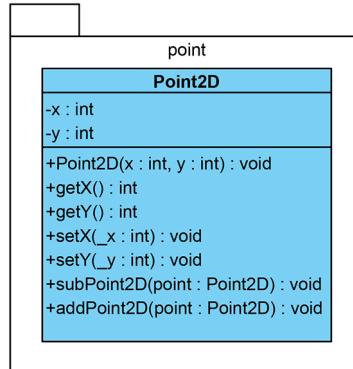


Figura 151: Diagramma della componente sgad::clienttier::view::graphicobjects::-point

- **Descrizione:** componente per la gestione dei punti rappresentabili con due coordinate.
- **Padre:** graphicobjects

3.26.2 Classi

3.26.2.1 sgad::clienttier::view::graphicobjects::point::Point2D

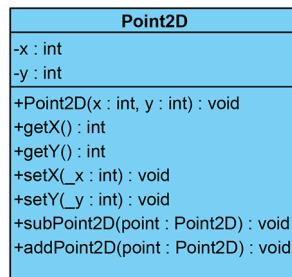


Figura 152: Diagramma della classe sgad::clienttier::view::graphicobjects::-point::Point2D

- **Descrizione:** classe per la rappresentazione di un punto avente due coordinate.
- **Utilizzo:** viene utilizzata per rappresentare un punto indicato dalle coordinate x e y.
- **Attributi**

— **-x :** int

Valore rappresentante la coordinata x del punto.

- `-y` : `int`
Valore rappresentante la coordinata y del punto.

- Metodi

- `+getX()` : `int`
Ottiene la coordinata x del punto.
- `+getY()` : `int`
Ottiene la coordinata y del punto.
- `+subPoint2D(point : Point2D) : void`
Rimuove il punto passato come parametro.

Argomenti:

- * `point` : punto da rimuovere.
- `+addPoint2D(point : Point2D) : void`
Aggiunge il punto passato come parametro.

Argomenti:

- * `point` : punto che si vuole aggiungere.
- `+setX(_x : int) : void`
Imposta la coordinata x del punto.

Argomenti:

- * `_x` : la nuova coordinata x del punto.
- `+setY(_y : int) : void`
Imposta la coordinata y del punto.

Argomenti:

- * `_y` : la nuova coordinata y del punto.
- `+Point2D(x : int, y : int) : Point2D`
Costruttore della classe.

Argomenti:

- * `x` : coordinata x del punto.
- * `y` : coordinata y del punto.

3.27 Componente sgad::clienttier::view::graphicobjects::shape

3.27.1 Informazioni sul package

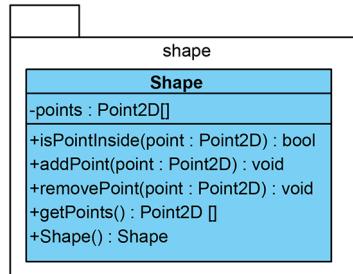


Figura 153: Diagramma della componente sgad::clienttier::view::graphicobjects::shape

- **Descrizione:** componente per la gestione delle forme degli oggetti grafici.
- **Padre:** graphicobjects
- **Interazioni con altri componenti**
 - **components:** componente per la gestione degli oggetti del mondo di gioco e della relativa visualizzazione.
 - **widget:** componente per la gestione dei widget per permettere all'utente di interagire con il mondo di gioco attraverso menu e finestre.

3.27.2 Classi

3.27.2.1 sgad::clienttier::view::graphicobjects::shape::Shape

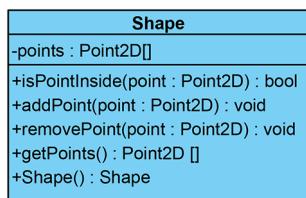


Figura 154: Diagramma della classe sgad::clienttier::view::graphicobjects::shape::Shape

- **Descrizione:** classe per la gestione di un insieme di punti rappresentanti un poligono.
- **Utilizzo:** viene utilizzata per rappresentare un insieme di punti. Le sottoclassi di **GraphicObject** usano oggetti di tale classe per avere un'indicazione su quale sia il poligono che le rappresentano.

Per un maggior approfondimento, si veda la sezione Gestione degli eventi.

- Attributi

- `-points : Point2D[]`

Lista ordinata di punti rappresentanti i vertici del poligono. Almeno uno dei punti deve essere posizionato sull'asse y e un altro deve esserlo sull'asse x.

- Metodi

- `+isPointInside(point : Point2D) : bool`

Controlla che il punto richiesto sia all'interno o meno del poligono che l'oggetto rappresenta.

Argomenti:

- * `point` : punto in 2 dimensioni sul quale deve essere effettuato il controllo.
Il punto non deve aver subito traslazioni. Nel caso, deve essere riportato allo stato originario.

- `+addPoint(point : Point2D) : void`

Aggiunge un punto al poligono in ultima posizione.

Argomenti:

- * `point` : il punto da aggiungere.

- `+removePoint(point : Point2D) : void`

Rimuove il punto dal poligono.

Argomenti:

- * `point` : il punto da rimuovere.

- `+Shape() : Shape`

Costruttore della classe.

- `+getPoints() : Array<Point2D>`

Controlla l'accesso alla lista dei punti.

3.28 Componente sgad::clienttier::view::graphicobjects::widget

3.28.1 Informazioni sul package

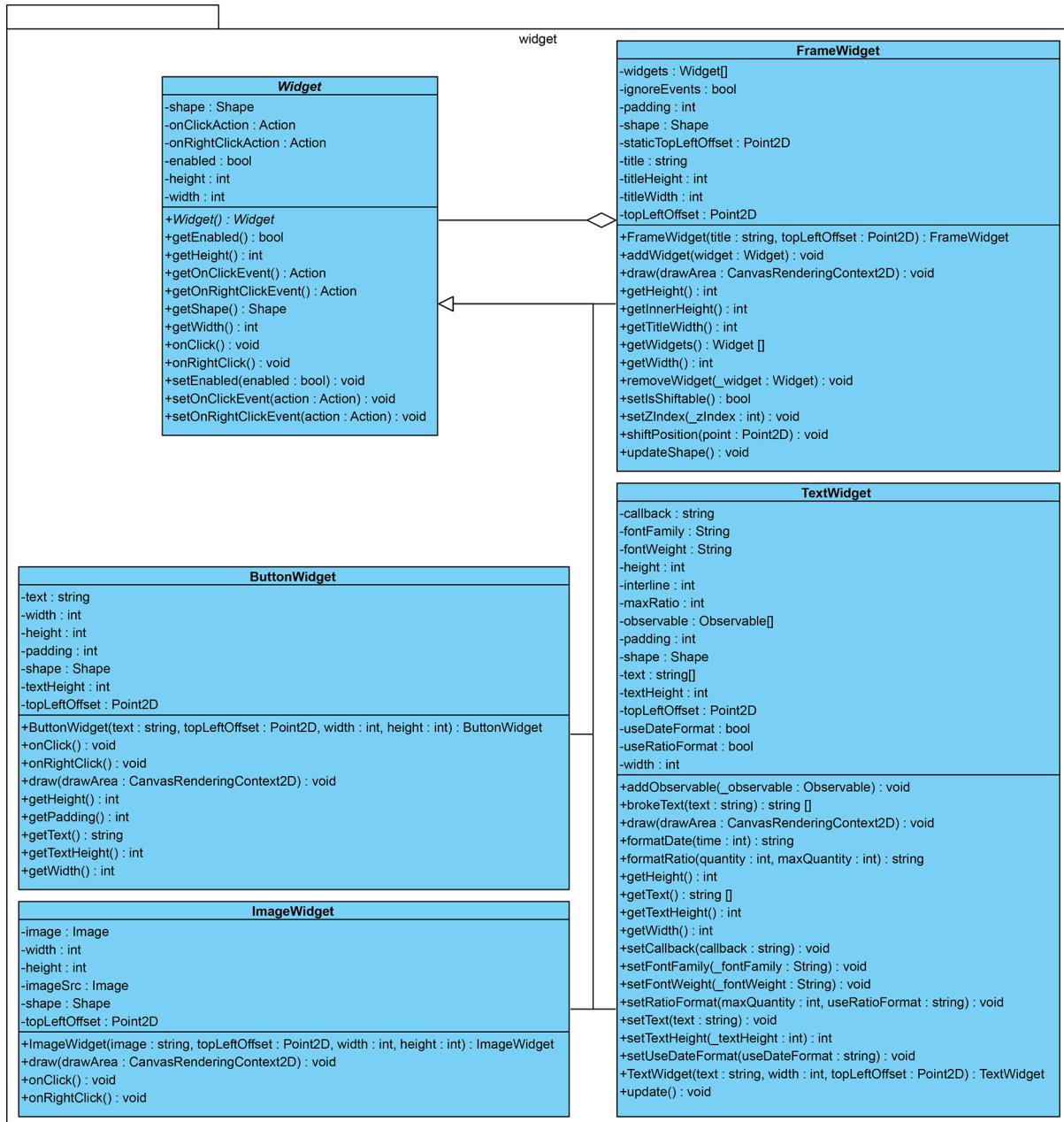


Figura 155: Diagramma della componente sgad::clienttier::view::graphicobjects::widget

- **Descrizione:** componente per la gestione dei widget per permettere all'utente di interagire con il mondo di gioco attraverso menu e finestre.
- **Padre:** `graphicobjects`
- **Interazioni con altri componenti**
 - `menufactory`: componente per la gestione dei vari menu che possono essere visualizzati durante le interazioni con il mondo di gioco.
 - `graphicobject`: componente per la gestione dell'interfaccia per qualsiasi oggetto che verrà visualizzato nell'area di gioco.
 - `shape`: componente per la gestione delle forme degli oggetti grafici.

3.28.2 Classi

3.28.2.1 sgad::clienttier::view::graphicobjects::widget::Widget



Figura 156: Diagramma della classe `sgad::clienttier::view::graphicobjects::widget::Widget`

- **Descrizione:** classe astratta per i possibili widget che possono essere rappresentati per interagire con il mondo di gioco.
- **Utilizzo:** viene utilizzata per fornire dei metodi comuni a qualsiasi tipo di widget che possono essere rappresentati. Le classi che ereditano da tale classe si occuperanno di ridefinire i metodi astratti. Mantiene un riferimento ad un oggetto di tipo `Shape` per rappresentare la forma del widget.

Per un maggior approfondimento, si veda la sezione Visualizzazione dei menu.

- **Classi ereditate**

- `GraphicObject`

- Sottoclassi

- ButtonWidget
- FrameWidget
- ImageWidget
- TextWidget

- Attributi

- `-enabled : bool`
Indica se si può interagire con l'oggetto o meno.
- `-height : int`
Altezza dell'oggetto widget
- `-onClickAction : Action`
L'azione da eseguire al verificarsi dell'evento click.
- `-onRightClickAction : Action`
L'azione da eseguire al verificarsi dell'evento rightclick.
- `-shape : Shape`
Il poligono che rappresenta l'oggetto grafico.
- `-width : int`
Larghezza dell'oggetto widget

- Metodi

- `+setOnClickEvent(action : Action) : void`
Imposta l'azione da eseguire al verificarsi dell'evento click.

Argomenti:

- * `action` : l'azione da impostare.

- `+setOnRightClickEvent(action : Action) : void`
Imposta l'azione da eseguire al verificarsi dell'evento rightclick.

Argomenti:

- * `action` : l'azione da impostare.

- `+setEnabled(enabled : bool) : void`
Imposta se si possa interagire con l'oggetto o meno.

Argomenti:

- * `enabled` : indica se si può interagire con l'oggetto, valore pari a true, o meno, valore pari a false.

- `+onClick() : void`
Esegue l'azione impostata nell'attributo `onClickAction`.
- `+onRightClick() : void`
Esegue l'azione impostata nell'attributo `onRightClickAction`.
- `+getShape() : Shape`
Fornisce la forma che rappresenta l'oggetto grafico.
- `+getWidth() : int`
Ritorna la larghezza del widget.
- `+getHeight() : int`
Ritorna l'altezza del widget
- `+getEnabled() : bool`
Ritorna se si può interagire con l'oggetto o meno.
- `+getOnClickEvent() : Action`
Imposta l'azione da eseguire al verificarsi dell'evento click.
- `+getOnRightClickEvent() : Action`
Imposta l'azione da eseguire al verificarsi dell'evento rightclick.
- `+Widget() : Widget`
Costruttore della classe.

3.28.2.2 sgad::clienttier::view::graphicobjects::widget::ButtonWidget

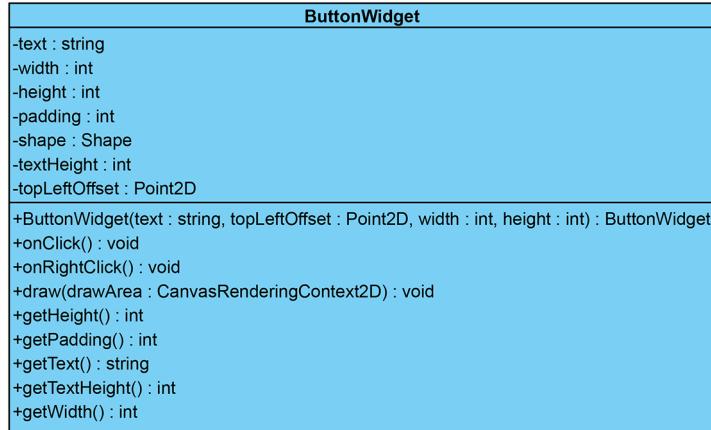


Figura 157: Diagramma della classe `sgad::clienttier::view::graphicobjects::widget::ButtonWidget`

- **Descrizione:** classe per la gestione di un generico bottone nell’interfaccia grafica.
- **Utilizzo:** viene utilizzata per rappresentare un bottone fornendo metodi comuni a qualsiasi possibile bottone che viene costruito. Le sottoclassi di `MenuFactory` determineranno quali saranno le azioni da eseguire al verificarsi di determinati eventi.
- **Classi ereditate**
 - `Widget`
- **Attributi**
 - `-height : int`
Altezza del bottone.
 - `-padding : int`
Padding del bottone.
 - `-shape : Shape`
Il poligono del bottone.
 - `-text : string`
Testo da visualizzare all’interno del bottone.
 - `-textHeight : int`
Altezza del testo del bottone.
 - `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.
 - `-width : int`
Larghezza del bottone.

- Metodi

– `+ButtonWidget(height : int, text : string, topLeftOffset : Point2D, width : int) : ButtonWidget`
Costruttore della classe.

Argomenti:

* `height` : l'altezza dell'oggetto grafico.

* `text` : il testo da visualizzare all'interno del bottone.

* `topLeftOffset` : la posizione del punto in alto a sinistra dell'oggetto grafico.

* `width` : la larghezza dell'oggetto grafico.

– `+draw(drawArea : CanvasRenderingContext2D) : void`
Disegna l'oggetto nell'area indicata.

Argomenti:

* `drawArea` : l'area nella quale deve essere rappresentato l'oggetto.

– `+onClick() : void`
Invoca l'azione da eseguire al verificarsi dell'evento click.

– `+onRightClick() : void`
Invoca l'azione da eseguire al verificarsi dell'evento rightclick.

– `+getText() : string`
Fornisce il testo del bottone.

– `+getWidth() : int`
Fornisce la larghezza del bottone.

– `+getHeight() : int`
Fornisce l'altezza del bottone.

– `+getPadding() : int`
Fornisce il padding del bottone.

– `+getTextHeight() : int`
Fornisce l'altezza del testo del bottone.

3.28.2.3 sgad::clienttier::view::graphicobjects::widget::FrameWidget

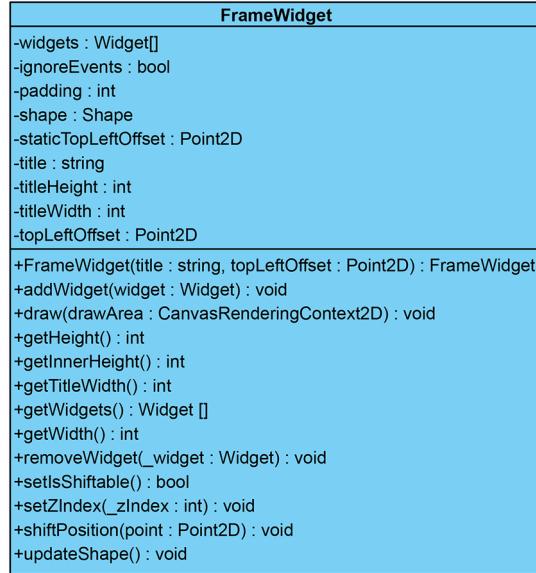


Figura 158: Diagramma della classe sgad::clienttier::view::graphicobjects::widget::FrameWidget

- **Descrizione:** classe per la gestione di un generico frame nell’interfaccia grafica.
- **Utilizzo:** viene utilizzata per rappresentare un frame fornendo metodi comuni a qualsiasi possibile frame che viene costruito. Le sottoclassi di MenuFactory determineranno quali saranno le azioni da eseguire al verificarsi di determinati eventi. Stabiliranno inoltre quali altri oggetti di tipo `Widget` il frame dovrà contenere.

Per un maggior approfondimento, si veda la sezione Visualizzazione dei menu.

- **Classi ereditate**

- `Widget`

- **Attributi**

- `-ignoreEvents : bool`
Booleano per indicare se ignorare o meno l’evento.
 - `-padding : int`
Il padding del frame.
 - `-shape : Shape`
Il poligono del frame.
 - `-staticTopLeftOffset : Point2D`
Il punto iniziale in cui è stato creato il frame.
 - `-title : string`
Titolo del frame. Se viene impostato a null allora non sarà visualizzato.

- `-titleHeight : int`
L'altezza del titolo del frame. Se il titolo non è presente, l'altezza sarà pari a 0.
- `-titleWidth : int`
La larghezza del titolo.
- `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.
- `-widgets : Array<Widget>`
L'insieme dei widget contenuti nel frame.

- Metodi

- `+FrameWidget(title : string, topLeftOffset : Point2D) : FrameWidget`
Costruttore della classe.

Argomenti:

- * `title` : il titolo del frame che deve essere visualizzato. Se il valore è pari a null allora non verrà visualizzato.

- * `topLeftOffset` : la posizione del punto in alto a sinistra dell'oggetto grafico.

- `+addWidget(widget : Widget) : void`
Aggiunge il widget indicato.

Argomenti:

- * `widget` : il widget da aggiungere.

- `+getHeight() : int`
Fornisce l'altezza del frame basandosi sull'oggetto che occupa la posizione più a destra.

- `+getWidth() : int`
Fornisce la larghezza del frame basandosi sull'oggetto che occupa la posizione più a destra.

- `+updateShape() : void`
Aggiorna la forma del poligono in base alla larghezza e all'altezza attuale.

- `+removeWidget(_widget : Widget) : void`
Rimuove il widget indicato.

Argomenti:

- * `_widget` : il widget da rimuovere.

– `+shiftPosition(point : Point2D) : void`

Trasla il frame in una determinata posizione. Sposta anche le posizioni di tutti gli oggetti interni al widget.

Argomenti:

* `point` : il punto indicante lo spostamento del frame.

– `+getInnerHeight() : int`

Fornisce l'altezza interna del frame basandosi sull'oggetto che occupa la posizione più in basso.

– `+getWidgets() : Array<Widget>`

Fornisce i widgets inclusi nel frame.

– `+setZIndex(_zIndex : int) : void`

Imposta la profondità del frame.

Argomenti:

* `_zIndex` : la profondità del frame.

– `+setIsShiftable() : bool`

Imposta il frame traslabilo, valore pari a true, non traslabilo, valore pari a false.

– `+getTitleWidth() : int`

Ritorna la larghezza del titolo.

– `+draw(drawArea : CanvasRenderingContext2D) : void`

Disegna l'oggetto nell'area indicata.

Argomenti:

* `drawArea` : l'area nella quale deve essere rappresentato l'oggetto.

3.28.2.4 sgad::clienttier::view::graphicobjects::widget::ImageWidget

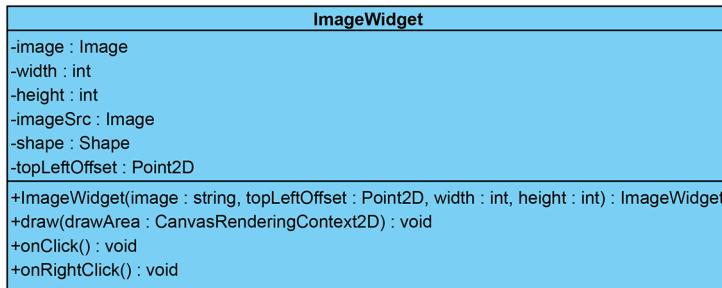


Figura 159: Diagramma della classe `sgad::clienttier::view::graphicobjects::widget::ImageWidget`

- **Descrizione:** classe per la gestione di una generica immagine nell’interfaccia grafica.
- **Utilizzo:** viene utilizzata per rappresentare un’immagine fornendo metodi comuni a qualsiasi possibile immagine che viene costruita. Le sottoclassi di `MenuFactory` determineranno quali saranno le azioni da eseguire al verificarsi di determinati eventi e quale effettiva immagine dovrà essere visualizzata.

- **Classi ereditate**

- `Widget`

- **Attributi**

- `-height : int`
L’altezza dell’oggetto grafico.
- `-image : Image`
Percorso dove è stata memorizzata l’immagine.
- `-imageSrc : Image`
Immagine rappresentante il widget.
- `-shape : Shape`
Il poligono dell’immagine.
- `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.
- `-width : int`
Larghezza dell’oggetto grafico.

- **Metodi**

- `+ImageWidget(height : int, image : string, topLeftOffset : Point2D, width : int) : ImageWidget`
Costruttore della classe.

Argomenti:

- * **height** : l'altezza dell'oggetto grafico.
 - * **image** : la stringa rappresentante il percorso dell'immagine.
 - * **topLeftOffset** : la posizione del punto in alto a sinistra dell'oggetto grafico.
 - * **width** : la larghezza dell'oggetto grafico.
- **+draw(drawArea : CanvasRenderingContext2D) : void**
Disegna l'oggetto nell'area indicata.

Argomenti:

- * **drawArea** : l'area nella quale deve essere rappresentato l'oggetto.
- **+onClick() : void**
Invoca l'azione da eseguire al verificarsi dell'evento click.
- **+onRightClick() : void**
Invoca l'azione da eseguire al verificarsi dell'evento rightclick.

3.28.2.5 sgad::clienttier::view::graphicobjects::widget::TextWidget



Figura 160: Diagramma della classe `sgad::clienttier::view::graphicobjects::widget::TextWidget`

- **Descrizione:** classe per la gestione di un generico testo nell’interfaccia grafica.
- **Utilizzo:** viene utilizzata per rappresentare un testo fornendo metodi comuni a qualsiasi possibile testo che viene costruito. Le sottoclassi di `MenuFactory` determineranno quale sarà l’effettivo testo da visualizzare. Stabiliranno inoltre se tale testo dovrà essere collegato ad oggetto di tipo `Observable` per mantenere aggiornato il testo.
- **Classi ereditate**
 - `Observer`
 - `Widget`
- **Attributi**
 - `-callback : string`
La funzione da invocare all’aggiornamento dello stato in seguito ad una notifica di un observer.

- `-fontFamily : String`
Il tipo di font da utilizzare.
- `-fontWeight : String`
Il peso del font da utilizzare.
- `-height : int`
Altezza dell'area testuale.
- `-interline : int`
L'interlinea tra le righe.
- `-maxRatio : int`
Indica il rapporto massimo.
- `-observable : Array<Observable>`
Oggetto che deve essere osservato.
- `-padding : int`
Padding dell'area testuale.
- `-shape : Shape`
Il poligono dell'area testuale.
- `-text : string[]`
Testo da visualizzare all'interno dell'area testuale. Viene spezzato in un vettore di righe.
- `-textHeight : int`
Altezza del testo dell'area testuale.
- `-topLeftOffset : Point2D`
La posizione del punto in alto a sinistra.
- `-useDateFormat : bool`
Indica se si usa o meno il formato data.
- `-useRatioFormat : bool`
Indica se si usa o meno il formato rapporto.
- `-width : int`
Larghezza dell'area testuale.

- Metodi

- `+TextWidget(text : string, topLeftOffset : Point2D, width : int) : TextWidget`
Costruttore della classe.

Argomenti:

- * `text` : il testo dell'area di testo da visualizzare.
- * `topLeftOffset` : la posizione del punto in alto a sinistra dell'oggetto grafico.
- * `width` : la larghezza dell'area di testo.

- `+draw(drawArea : CanvasRenderingContext2D) : void`
Disegna l'oggetto nell'area indicata.

Argomenti:

* `drawArea` : l'area nella quale deve essere rappresentato l'oggetto.

- `+getHeight() : int`
Fornisce l'altezza dell'area testuale.

- `+update() : void`
Aggiorna il testo visualizzato.

- `+getWidth() : int`
Fornisce la larghezza dell'area testuale.

- `+getTextHeight() : int`
Fornisce l'altezza del testo dell'area testuale.

- `+brokeText(text : string) : string[]`
Spezza il testo se troppo lungo in più righe.

Argomenti:

* `text` : il testo da analizzare.

- `+getText() : string[]`
Fornisce il testo dell'area testuale.

- `+setText(text : string) : void`
Imposta il testo dell'area testuale.

Argomenti:

* `text` : il testo dell'area testuale.

- `+setTextHeight(_textHeight : int) : int`
Imposta l'altezza del testo dell'area testuale.

Argomenti:

* `_textHeight` : l'altezza del testo dell'area testuale.

- `+addObservable(_observable : Observable) : void`
Aggiunge l'oggetto che deve essere osservato per poter aggiornare il testo.

Argomenti:

* `_observable` : l'oggetto da osservare.

– `+setFontFamily(_fontFamily : String) : void`

Imposta il tipo di font da utilizzare.

Argomenti:

* `_fontFamily` : il tipo di font da impostare.

– `+setFontSize(_fontSize : String) : void`

Imposta il dimensione del font da utilizzare.

Argomenti:

* `_fontSize` : la dimensione del font da impostare.

– `+formatRatio(maxQuantity : int, quantity : int) : string`

Ritorna la quantità attuale rapportata alla quantità massima.

Argomenti:

* `maxQuantity` : quantità massima possibile.

* `quantity` : quantità corrente.

– `+formatDate(time : int) : string`

Ritorna un timer nel formato hh/mm/ss.

Argomenti:

* `time` : tempo in millisecondi.

– `+setUseDateFormat(useDateFormat : string) : void`

Imposta il timer.

Argomenti:

* `useDateFormat` : il timer da impostare.

– `+setRatioFormat(maxQuantity : int, useRatioFormat : string) : void`

Imposta il rapporto quantità corrente su quantità massima possibile.

Argomenti:

* `maxQuantity` : quantità massima possibile.

* `useRatioFormat` : il rapporto da impostare.

– `+setCallback(callback : string) : void`

Imposta la funzione che deve essere invocata quando si riceve un update.

Argomenti:

- * `callback` : il nome della funzione da invocare.

3.29 Componente `sgad::servertier`

3.29.1 Informazioni sul package

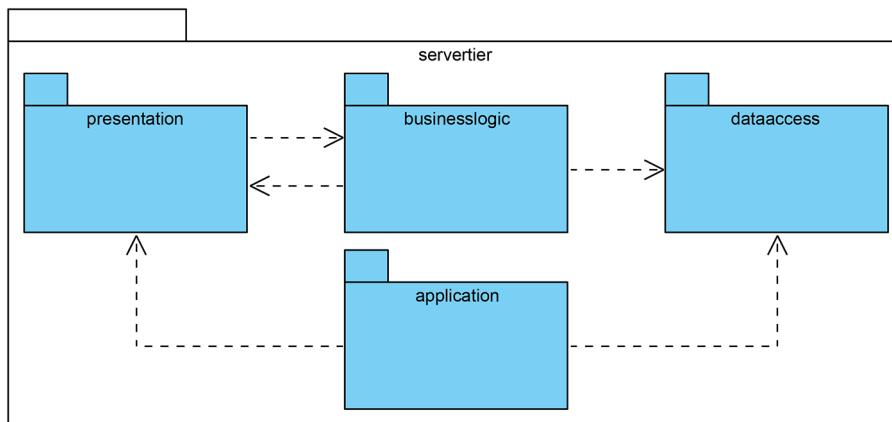


Figura 161: Diagramma della componente `sgad::servertier`

- **Descrizione:** componente globale per il back end del prodotto. Le relazioni tra le componenti `presentation`, `businesslogic` e `dataaccess` rappresentano delle interazioni tipiche. Esse intercorrono tra le componenti del design pattern Three-Tier.

- **Padre:** `sgad`

- **Interazioni con altri componenti**

- `clienttier`: componente globale per il front end del prodotto. Nel Three-Tier del sistema rappresenta il livello Client Tier. Le relazioni tra le sottocomponenti `model`, `view` e `controller` rappresentano le interazioni tipiche che intercorrono tra le componenti del design pattern MVC. Esso si occupa di visualizzare i dati all'utente e di inoltrare le richieste al back end.

- **Package contenuti**

- `application`: componente che si occupa del `bootstrapg` dell'applicazione.
- `businesslogic`: componente corrispondente al livello business logic dell'architettura Three-Tier. Essa si occupa di effettuare le operazioni richieste dagli utenti. Nel caso le operazioni siano relative a interazioni tra più utenti, comunicherà

con la componente **presentation** per instradare correttamente la richiesta verso gli altri client coinvolti.

- **dataaccess**: componente per la gestione dell'accesso al $database_{|g|}$ e della creazione degli oggetti delle componenti shareddata e userdata.
- **presentation**: componente per la gestione della comunicazione con i client. Rappresenta il livello superiore e visibile dall'esterno dell'architettura Three-Tier. Si occupa di smistare le richieste in arrivo alle corrette componenti della logica.

3.30 Componente sgad::servertier::application

3.30.1 Informazioni sul package

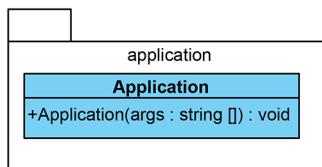


Figura 162: Diagramma della componente `sgad::servertier::application`

- **Descrizione**: componente che si occupa del bootstrap dell'applicazione.
- **Padre**: `servertier`
- **Interazioni con altri componenti**
 - **databaseaccess**: componente corrispondente al livello data access dell'architettura Three-Tier. Essa si occupa del mantenimento dei dati e della comunicazione con il database.
 - **presentation**: componente per la gestione della comunicazione con i client. Rappresenta il livello superiore e visibile dall'esterno dell'architettura Three-Tier. Si occupa di smistare le richieste in arrivo alle corrette componenti della logica.
 - **cluster**: componente per la gestione e la partecipazione del singolo server a $cluster_{|g|}$ di server.

3.30.2 Classi

3.30.2.1 sgad::servertier::application::Application



Figura 163: Diagramma della classe `sgad::servertier::application::Application`

- **Descrizione:** classe per la gestione del bootstrap dell'applicativo lato server.
 - **Utilizzo:** viene utilizzata per la creazione, l'avvio e il caricamento dei vari moduli che compongono l'applicativo di back end.
 - **Metodi**
 - `+Application(args : string[]) : void`
Corrisponde al main del programma.
- Argomenti:**
- * `args` : dati in input all'avvio del programma.

3.31 Componente `sgad::servertier::businesslogic`

3.31.1 Informazioni sul package

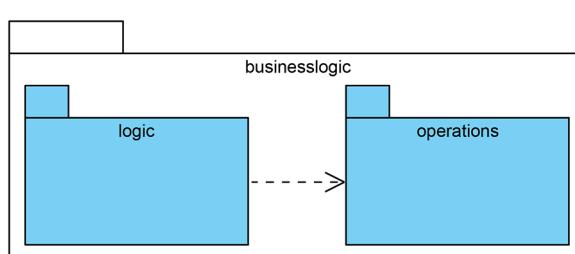


Figura 164: Diagramma della componente `sgad::servertier::businesslogic`

- **Descrizione:** componente corrispondente al livello business logic dell'architettura Three-Tier. Essa si occupa di effettuare le operazioni richieste dagli utenti. Nel caso le operazioni siano relative a interazioni tra più utenti, comunicherà con la componente `presentation` per instradare correttamente la richiesta verso gli altri client coinvolti.
- **Padre:** `servertier`
- **Interazioni con altri componenti**
 - `databaseaccess`: componente corrispondente al livello data access dell'architettura Three-Tier. Essa si occupa del mantenimento dei dati e della comunicazione con il database.
 - `presentation`: componente per la gestione della comunicazione con i client. Rappresenta il livello superiore e visibile dall'esterno dell'architettura Three-Tier. Si occupa di smistare le richieste in arrivo alle corrette componenti della logica.
- **Package contenuti**
 - `logic`: componente che gestisce il controllo delle operazioni richieste dal client.
 - `operations`: componente per la gestione dell'insieme delle possibili operazioni che un utente può effettuare.

3.32 Componente sgad::servertier::businesslogic::logic

3.32.1 Informazioni sul package

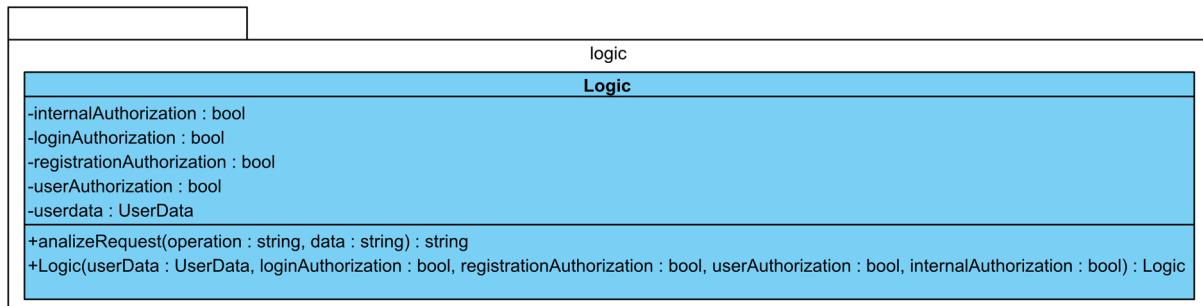


Figura 165: Diagramma della componente sgad::servertier::businesslogic::logic

- **Descrizione:** componente che gestisce il controllo delle operazioni richieste dal client.
- **Padre:** businesslogic
- **Interazioni con altri componenti**
 - **operations:** componente per la gestione dell'insieme delle possibili operazioni che un utente può effettuare.
 - **databasemanager:** componente per la gestione dell'interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l'unico punto di accesso con il database.
 - **userManager:** componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore associato alla richiesta.

3.32.2 Classi

3.32.2.1 sgad::servertier::businesslogic::logic::Logic

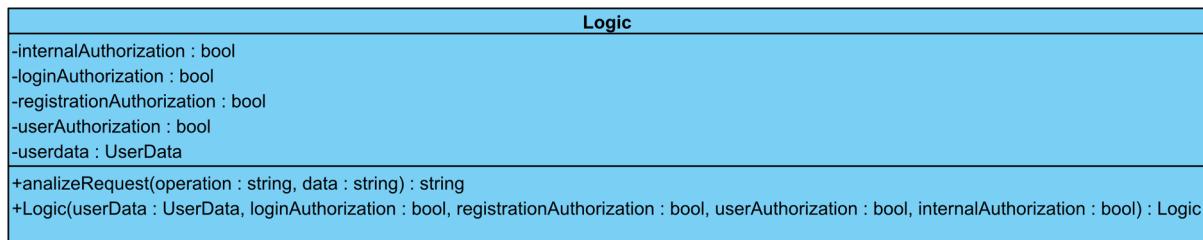


Figura 166: Diagramma della classe sgad::servertier::businesslogic::logic::Logic

- **Descrizione:** classe per la gestione dell'operazione richiesta dall'utente. Essa si occupa di instraradare la richiesta tramite l'invocazione dell'operazione giusta.
- **Utilizzo:** viene utilizzata per la scelta della giusta operazione da eseguire come risposta alla richiesta del client.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione, Gestione di una richiesta di login e Gestione di una richiesta di gioco.

- **Attributi**

- `-internalAuthorization : bool`
Autorizzazione a operare richieste interne. Di default è false.
- `-loginAuthorization : bool`
Autorizzazione a operare richieste di login. Di default è false.
- `-registrationAuthorization : bool`
Autorizzazione a operare richieste di registrazione. Di default è false.
- `-userAuthorization : bool`
Autorizzazione a operare richieste di user. Di default è false.
- `-userdata : UserData`
I dati dell'utente a cui la classe Logic è associata.

- **Metodi**

- `+analyzeRequest(data : string, operation : string) : string`
Analizza ed esegue una richiesta.

Argomenti:

- * `data` : i dati in input alla richiesta.

- * `operation` : il tipo di richiesta desiderata.

- `+Logic(internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : Logic`
Costruttore della classe.

Argomenti:

- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.

- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.

- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.

* **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.

* **userData** : i dati dell'utente a cui la classe Logic è associata.

3.33 Componente sgad::servtier::businesslogic::operations

3.33.1 Informazioni sul package

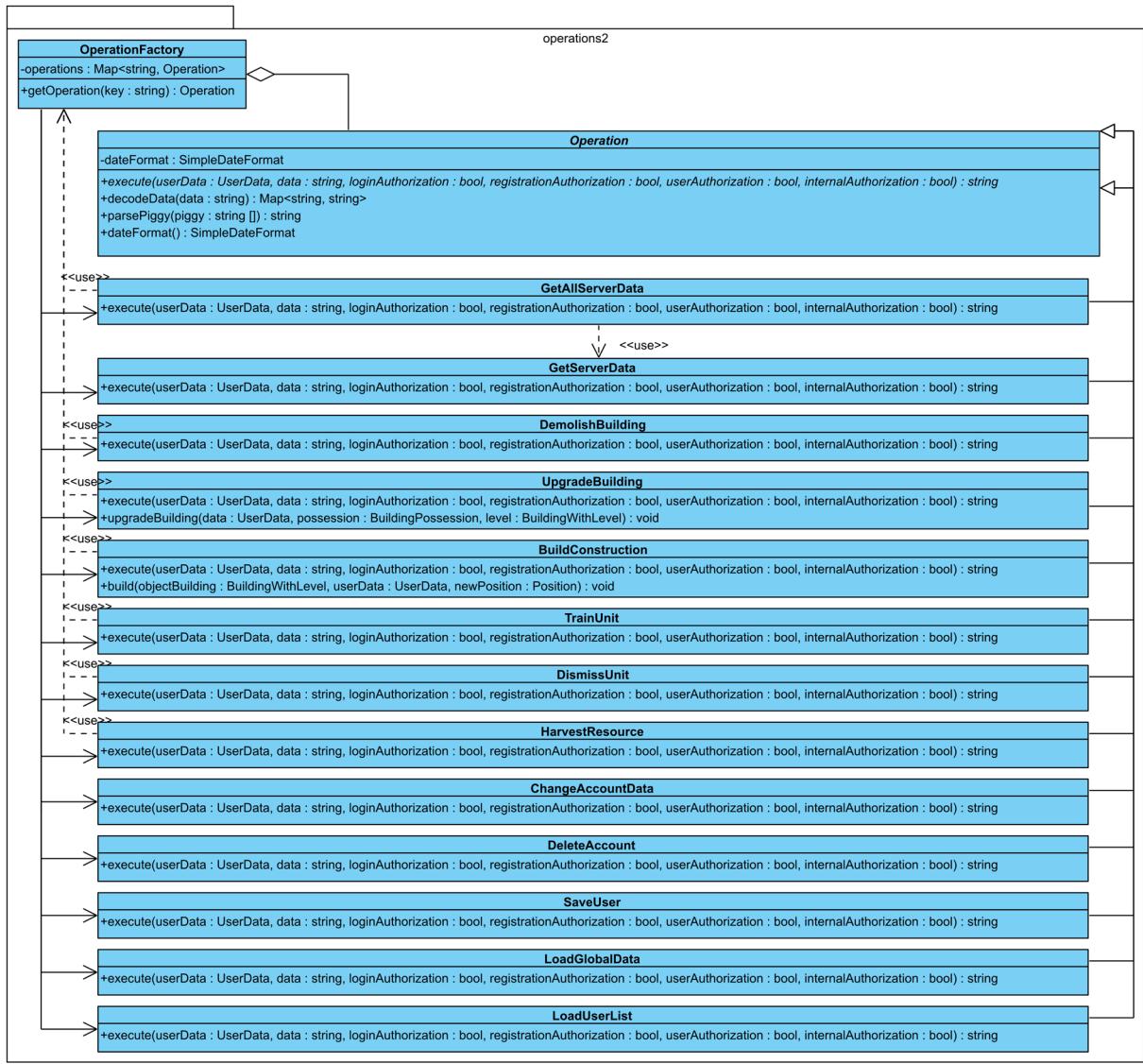


Figura 167: Diagramma della componente `sgad::servtier::businesslogic::operations`

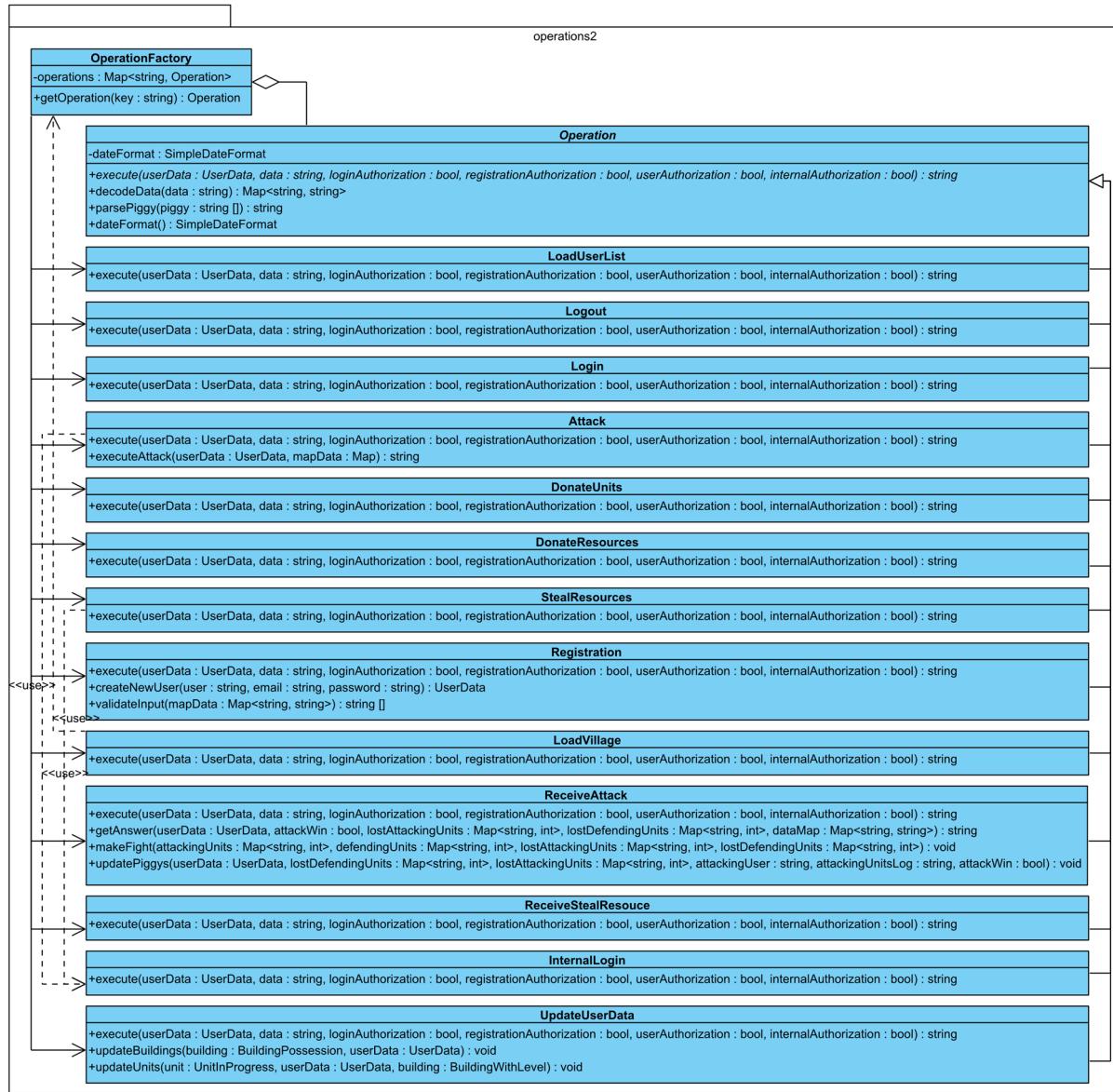


Figura 168: Diagramma della componente `sgad::servtier::businesslogic::-operations`

- **Descrizione:** componente per la gestione dell'insieme delle possibili operazioni che un utente può effettuare.
- **Padre:** `businesslogic`
- **Interazioni con altri componenti**
 - `logic`: componente che gestisce il controllo delle operazioni richieste dal client.

- **data:** componente che gestisce l'insieme dei dati necessari per mantenere memorizzata, sul server, la sessione dell'utente.

3.33.2 Classi

3.33.2.1 sgad::servertier::businesslogic::operations::Operation

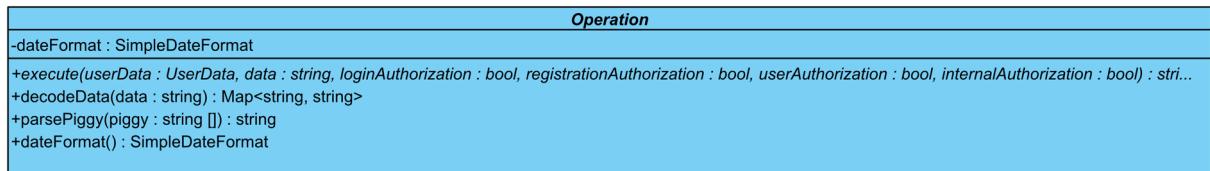


Figura 169: Diagramma della classe `sgad::servertier::businesslogic::operations::Operation`

- **Descrizione:** classe astratta per l'esecuzione di un'operazione.
- **Utilizzo:** viene utilizzata mediante un riferimento dalla classe `Logic` per l'esecuzione di un'operazione sul model.

Per un maggior approfondimento, si veda la sezione Gestione di una richiesta di gioco.

- **Sottoclassi**

- `Attack`
- `BuildConstruction`
- `ChangeAccountData`
- `DeleteAccount`
- `DemolishBuilding`
- `DismissUnit`
- `DonateResources`
- `DonateUnits`
- `GetAllServerData`
- `GetServerData`
- `HarvestResource`
- `InternalLogin`
- `LoadGlobalData`
- `LoadVillage`
- `LoadUserList`
- `Login`
- `Logout`

- ReceiveAttack
- ReceiveStealResource
- Registration
- SaveUser
- StealResource
- TrainUnit
- UpdateUserData
- UpgradeBuilding

- **Attributi**

- `-dateFormat : SimpleDateFormat`

Attributo utilizzato per restituire la data corrente formattata in maniera adeguata.

- **Metodi**

- `+decodeData(data : string) : Map<string, string>`

Decodifica i dati in input restituendo una mappa facile da utilizzare.

Argomenti:

* `data` : stringa dei dati in input da decodificare.

- `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`

Metodo astratto che rappresenta l'esecuzione di un'operazione.

Argomenti:

* `data` : dati accompagnatori alla richiesta dell'operazione.

* `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.

* `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.

* `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.

* `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.

* `userData` : dati dell'utente su cui verrà effettuata l'operazione.

– `+parsePiggy(piggy : string[]) : string`

Metodo utilizzato per fare il parsing dei messaggi PiggyBack per restituirli al client.

Argomenti:

* `piggy` : il riferimento all'arraybuffer dei messaggi.

– `+dateFormat() : SimpleDateFormat`

Restituisce il valore del parametro dateFormat.

3.33.2.2 sgad::servertier::businesslogic::operations::Attack

Attack
<code>+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string</code>
<code>+executeAttack(userData : UserData, mapData : Map) : string</code>

Figura 170: Diagramma della classe `sgad::servertier::businesslogic::operations::Attack`

- **Descrizione:** classe che rappresenta l'operazione di lancio attacco ad un altro utente.
 - **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di attacco. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
 - **Classi ereditate**
 - `Operation`
 - **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
 - Metodo che esegue l'operazione di lancio di un attacco ad un altro utente.
- Argomenti:**
- * `data` : dati accompagnatori alla richiesta dell'operazione.
 - * `internalAuthorization` : autorizzazione ad operare richieste interne. Di default è false.
 - * `loginAuthorization` : autorizzazione ad operare richieste di login. Di default è false.

- * `registrationAuthorization` : autorizzazione ad operare richieste di registrazione. Di default è false.
 - * `userAuthorization` : autorizzazione ad operare richieste di user. Di default è false.
 - * `userData` : dati dell'utente su cui verrà effettuata l'operazione.
- `-executeAttack(mapData : Map<string, string='></string,>, userData : UserData) : string`
 Esegue l'attacco vero e proprio fornendo una risposta da dare al client.

Argomenti:

- * `mapData` : dati in input alla operazione già codificati.
- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.3 sgad::servertier::businesslogic::operations::BuildConstruction

BuildConstruction
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string
+build(objectBuilding : BuildingWithLevel, userData : UserData, newPosition : Position) : void

Figura 171: Diagramma della classe `sgad::servertier::businesslogic::operations::-BuildConstruction`

- **Descrizione:** classe che rappresenta l'operazione di costruzione di un edificio.
- **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di costruzione di un edificio. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `-build(newPosition : Position, objectBuilding : BuildingWithLevel, userData : UserData) : void`
 Metodo utilizzato dalla classe per effettuare la costruzione, il metodo richiede che i test di consistenza con i dati siano stati eseguiti.

Argomenti:

- * **newPosition** : È la posizione in cui si costruisce la costruzione.
- * **objectBuilding** : È la BuildingWithLevel da costruire.
- * **userData** : i dati utente di chi richieda la costruzione.

– **+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string**
 Esecuzione della costruzione.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.4 sgad::servertier::businesslogic::operations::ChangeAccountData

ChangeAccountData
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 172: Diagramma della classe `sgad::servertier::businesslogic::operations::- ChangeAccountData`

- **Descrizione:** classe che rappresenta l'operazione di modifica dei dati dell'account utente.
- **Utilizzo:** la classe viene utilizzata per la modifica dei dati dell'account di un utente.
- **Classi ereditate**

- Operation
- Metodi
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`

Implementa l'operazione di modifica dei dati dell'account.

Argomenti:

 - * `data` : dati accompagnatori alla richiesta dell'operazione.
 - * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
 - * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
 - * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
 - * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
 - * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.5 sgad::servertier::businesslogic::operations::DeleteAccount

DeleteAccount
<code>+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string</code>

Figura 173: Diagramma della classe `sgad::servertier::businesslogic::operations::DeleteAccount`

- **Descrizione:** classe che rappresenta l'operazione di eliminazione di un account utente.
- **Utilizzo:** la classe viene utilizzata per l'eliminazione di un account utente.
- **Classi ereditate**
 - Operation
- Metodi

```

  - +execute(data : string, internalAuthorization : bool, loginAuthorization
    : bool, registrationAuthorization : bool, userAuthorization : bool,
    userData : UserData) : string
  
```

Metodo che implementa l'eliminazione di un account utente.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.6 sgad::servertier::businesslogic::operations::DemolishBuilding

DemolishBuilding
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 174: Diagramma della classe `sgad::servertier::businesslogic::operations::- DemolishBuilding`

- **Descrizione:** classe che rappresenta la demolizione di un edificio.
- **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di demolizione di un edificio. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - Operation
- **Metodi**

```

    - +execute(data : string, internalAuthorization : bool, loginAuthorization
      : bool, registrationAuthorization : bool, userAuthorization : bool,
      userData : UserData) : string
  
```

Metodo di esecuzione della demolizione.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.7 sgad::servertier::businesslogic::operations::DismissUnit

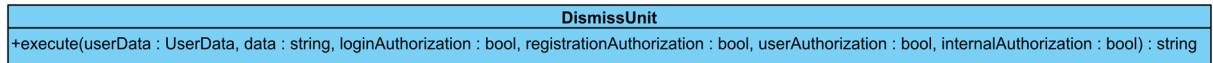


Figura 175: Diagramma della classe `sgad::servertier::businesslogic::operations::DismissUnit`

- **Descrizione:** classe che rappresenta l'operazione di congedo di unità.
- **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di congedo di unità. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - Operation
- **Metodi**

```

  - +execute(data : string, internalAuthorization : bool, loginAuthorization
    : bool, registrationAuthorization : bool, userAuthorization : bool,
    userData : UserData) : string
  
```

Metodo per eseguire la distruzione di unità.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.8 sgad::servertier::businesslogic::operations::DonateResources



Figura 176: Diagramma della classe `sgad::servertier::businesslogic::operations::-DonateResources`

- **Descrizione:** classe che rappresenta l'operazione di dono di risorse ad un altro utente.
- **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di dono di risorse ad un altro utente.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`

Metodo di esecuzione per la donazione di risorse.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.
- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
- * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.9 sgad::servertier::businesslogic::operations::DonateUnits



Figura 177: Diagramma della classe `sgad::servertier::businesslogic::operations::-DonateUnits`

- **Descrizione:** classe che rappresenta il dono di unità ad un altro utente.
- **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di dono di unità ad un altro utente.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Metodo di esecuzione per la donazione di unità.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.
- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
- * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.10 sgad::servertier::businesslogic::operations::GetAllServerData

GetAllServerData
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 178: Diagramma della classe `sgad::servertier::businesslogic::operations::GetAllServerData`

- **Descrizione:** classe per la fornitura delle informazioni sullo stato di tutto il cluster.
- **Utilizzo:** viene utilizzata per fornire al client tutte le informazioni di stato sul cluster.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Recupera le informazioni di tutti i server.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.

- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
- * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.11 sgad::servtier::businesslogic::operations::GetServerData

GetServerData
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 179: Diagramma della classe `sgad::servtier::businesslogic::operations::GetServerData`

- **Descrizione:** classe per la fornitura dei dati di stato sul server.
- **Utilizzo:** viene utilizzata per fornire i dati di stato sul server.
- **Classi ereditate**
 - Operation
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Restituisce le informazioni di un singolo server.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.
- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.

- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.12 sgad::servtier::businesslogic::operations::HarvestResource



Figura 180: Diagramma della classe `sgad::servtier::businesslogic::operations::- HarvestResource`

- **Descrizione:** classe per la gestione dell'operazione di raccolta risorse da un edificio.
- **Utilizzo:** viene utilizzata per la raccolta delle risorse prodotte da un edificio. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - Operation
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Metodo di esecuzione della raccolta risorse.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.

- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.13 sgad::servtier::businesslogic::operations::InternalLogin

InternalLogin
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 181: Diagramma della classe sgad::servtier::businesslogic::operations::InternalLogin

- **Descrizione:** classe che rappresenta l'operazione di login.
- **Utilizzo:** la classe viene utilizzata per effettuare login interni allo scopo di permettere ad altri utente di comunicare con utenti non loggati in quel momento.
- **Classi ereditate**
 - Operation
- **Metodi**
 - +execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userdata : UserData) : string
Metodo di esecuzione del login interno senza controllo della password.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.

- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userdata** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.14 sgad::servtier::businesslogic::operations::LoadGlobalData

LoadGlobalData
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 182: Diagramma della classe `sgad::servtier::businesslogic::operations::-LoadGlobalData`

- **Descrizione:** classe per la gestione del caricamento da parte del client dei dati globali di gioco.
- **Utilizzo:** viene utilizzata per dare al client le informazioni sui dati generali di gioco.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userdata : UserData) : string`
Metodo di load dei dati generali di gioco.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.

* **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.

* **userdata** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.15 sgad::servertier::businesslogic::operations::LoadVillage

LoadVillage
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 183: Diagramma della classe `sgad::servertier::businesslogic::operations::LoadVillage`

- **Descrizione:** classe per la gestione dell'operazione di caricamento del villaggio richiesta del client.

- **Utilizzo:** viene utilizzata per dare al client i dati del villaggio dell'utente. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.

- **Classi ereditate**

- Operation

- **Metodi**

- `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`

Metodo di esecuzione di caricamento dei dati di gioco dell'user.

Argomenti:

* **data** : dati accompagnatori alla richiesta dell'operazione.

* **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.

* **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.

* **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.

- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.16 sgad::servertier::businesslogic::operations::LoadUserList



Figura 184: Diagramma della classe `sgad::servertier::businesslogic::operations::LoadUserList`

- **Descrizione:** classe per la gestione dell'operazione di scelta degli utenti con i quali un utente può interagire.
- **Utilizzo:** viene utilizzata per dire al client quali sono gli utenti con i quali può interagire.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Metodo per il caricamento della lista utenti da far visualizzare all'utente.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.
- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.

- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.17 sgad::servertier::businesslogic::operations::Login



Figura 185: Diagramma della classe `sgad::servertier::businesslogic::operations::- Login`

- **Descrizione:** classe che rappresenta l'operazione di login.
 - **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di login richiesta da un client.
Per un maggior approfondimento, si rimanda alla sezione Gestione di una richiesta di login.
 - **Classi ereditate**
 - `Operation`
 - **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userdata : UserData) : string`
Metodo per il login di un utente.
- Argomenti:**
- * `data` : dati accompagnatori alla richiesta dell'operazione.
 - * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
 - * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
 - * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
 - * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
 - * `userdata` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.18 sgad::servertier::businesslogic::operations::Logout

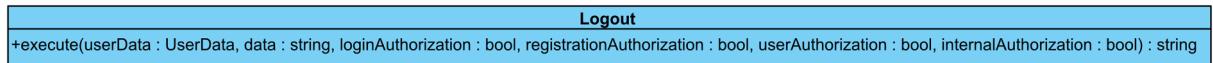


Figura 186: Diagramma della classe sgad::servertier::businesslogic::operations::Logout

- **Descrizione:** classe che rappresenta l'operazione di logout.
 - **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di logout.
 - **Classi ereditate**
 - Operation
 - **Metodi**
 - +execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string
Metodo per il logout di un utente.
- Argomenti:**
- * **data** : dati accompagnatori alla richiesta dell'operazione.
 - * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.
 - * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
 - * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
 - * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
 - * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.19 sgad::servertier::businesslogic::operations::ReceiveAttack



Figura 187: Diagramma della classe `sgad::servertier::businesslogic::operations::-ReceiveAttack`

- **Descrizione:** classe che gestisce la ricezione dell'attacco ed il computo del risultato della battaglie.
- **Utilizzo:** classe utilizzata per la gestione della ricezione dell'attacco e per la computazione del risultato delle battaglie. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Metodo che gestite la ricezione dell'attacco e il computo del risultato della battaglie.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.
- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
- * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

– `+makeFight(attackUnits : Map<string, int>, defendingUnits : Map<string, int>, lostAttackUnits : Map<string, int>, lostDefendUnits : Map<string, int>) : void`

Metodo ausiliario per calcolare l'esito della battaglia.

Argomenti:

- * `attackUnits` : una mappa contenente la quantità mappata con la chiave delle unità in attacco.

- * `defendingUnits` : una mappa contenente la quantità mappata con la chiave delle unità in difesa.

- * `lostAttackUnits` : una mappa in cui mappare le unità perse durante lo scontro per l'attaccante.

- * `lostDefendUnits` : una mappa in cui mappare le unità perse durante lo scontro dalla difesa.

– `+updatePiggys(attackUnitsLog : string, attackUser : string, attackWin : bool, lostAttackUnits : Map<string, int>, lostDefendUnits : Map<string, int>, userData : UserData) : void`

Metodo utilizzato per aggiornare i piggyBack di un user durante un attacco.

Argomenti:

- * `attackUnitsLog` : le unità attaccanti già formattate per il log.

- * `attackUser` : nome dell'attaccante.

- * `attackWin` : booleano per identificare se si è vinto o perso lo scontro.

- * `lostAttackUnits` : mappa contenente il numero di unità perdute dall'attaccante.

- * `lostDefendUnits` : mappa contenente il numero di unità perdute dal difensore.

- * `userData` : riferimento ai dati utente.

– `-getAnswer(attackWin : bool, dataMap : Map<string, string>, lostAttackUnits : Map<string, int>, lostDefendUnits : Map<string, int>, userData : UserData) : string`

Metodo utilizzato per formattare correttamente la risposta.

Argomenti:

- * `attackWin` : un booleano relativo all'esito dello scontro.
- * `dataMap` : una mappa dei dati in ingresso all'operazione.
- * `lostAttackingUnits` : una mappa che identifica le unità perse dall'attaccante.
- * `lostDefendingUnits` : una mappa che identifica le unità perse dal difensore.
- * `userData` : i dati dell'utente.

3.33.2.20 sgad::servertier::businesslogic::operations::ReceiveStealResource

ReceiveStealResource
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string

Figura 188: Diagramma della classe `sgad::servertier::businesslogic::operations::-ReceiveStealResource`

- **Descrizione:** classe che rappresenta l'operazione interna di ricezione di un attacco e computo dell'esito dello scontro.
- **Utilizzo:** classe utilizzata per rappresentare l'operazione interna di ricezione di un attacco e computazione dell'esito dello scontro. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - `Operation`
- **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
 - Metodo astratto che rappresenta l'esecuzione di un'operazione.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.
- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.

- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.
- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.
- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.
- * **userData** : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.21 sgad::servertier::businesslogic::operations::Registration



Figura 189: Diagramma della classe `sgad::servertier::businesslogic::operations::Registration`

- **Descrizione:** classe per la gestione dell'operazione di registrazione.
- **Utilizzo:** viene utilizzata per registrare nel database un nuovo utente.
Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione e Gestione di una richiesta di login.
- **Classi ereditate**
 - Operation
- **Metodi**
 - `-createNewUser(email : string, password : string, user : string)`
Crea i dati di un nuovo utente con le costruzioni e le risorse standard.

Argomenti:

- * **email** : email del nuovo utente.
- * **password** : password del nuovo utente.
- * **user** : username del nuovo utente.

```
— +execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userdata : UserData) : string
```

Metodo che esegue la registrazione di un nuovo utente.

Argomenti:

- * **data** : dati accompagnatori alla richiesta dell'operazione.

- * **internalAuthorization** : autorizzazione a operare richieste interne. Di default è false.

- * **loginAuthorization** : autorizzazione a operare richieste di login. Di default è false.

- * **registrationAuthorization** : autorizzazione a operare richieste di registrazione. Di default è false.

- * **userAuthorization** : autorizzazione a operare richieste di user. Di default è false.

- * **userdata** : dati dell'utente su cui verrà effettuata l'operazione.

```
— +validateInput(mapData : Map<string, string>) : string[]
```

Controlla i dati in input e li valida restituendo la lista degli errori.

Argomenti:

- * **mapData** : mappa contenete i parametri in input alla registrazione.

3.33.2.22 sgad::servertier::businesslogic::operations::SaveUser



Figura 190: Diagramma della classe `sgad::servertier::businesslogic::operations::SaveUser`

- **Descrizione:** classe che rappresenta l'operazione di salvataggio sul database i dati aggiornati dell'utente in input.
- **Utilizzo:** viene utilizzata per salvare i dati dell'utente aggiornati nel database.
- **Classi ereditate**

- Operation
 - Metodi
 - `+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string`
 Metodo che esegue il salvataggio sul database dei nuovi dati dell’utente.
- Argomenti:**
- * `userData` : dati accompagnatori alla richiesta dell’operazione.
 - * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
 - * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
 - * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
 - * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
 - * `userData` : dati dell’utente su cui verrà effettuata l’operazione.

3.33.2.23 sgad::servertier::businesslogic::operations::StealResource

StealResource
<code>+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string</code>

Figura 191: Diagramma della classe `sgad::servertier::businesslogic::operations::-StealResource`

- **Descrizione:** classe che rappresenta l’operazione di saccheggio di risorse da un edificio di un altro utente.
- **Utilizzo:** viene utilizzata per l’esecuzione dell’operazione di saccheggio di risorse da un edificio di un altro utente. Questa operazione, per poter portare a termine il proprio compito, utilizza un’altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**

- Operation
 - Metodi
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Esegue l'operazione di saccheggio dopo aver vinto una battaglia.
- Argomenti:**
- * `data` : dati accompagnatori alla richiesta dell'operazione.
 - * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
 - * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
 - * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
 - * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
 - * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.24 sgad::servertier::businesslogic::operations::TrainUnit



Figura 192: Diagramma della classe `sgad::servertier::businesslogic::operations::TrainUnit`

- **Descrizione:** classe che rappresenta l'operazione di arruolamento di unità.
- **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di arruolamento di unità. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
- **Classi ereditate**
 - Operation

- Metodi

- `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`

Metodo per l'esecuzione dell'addestramento delle unità.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.

- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.

- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.

- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.

- * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.

- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

3.33.2.25 sgad::servertier::businesslogic::operations::UpdateUserData

UpdateUserData
<code>+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string</code>
<code>+updateBuildings(building : BuildingPossession, userData : UserData) : void</code>
<code>+updateUnits(unit : UnitInProgress, userData : UserData, building : BuildingWithLevel) : void</code>

Figura 193: Diagramma della classe `sgad::servertier::businesslogic::operations::-UpdateUserData`

- **Descrizione:** mantiene aggiornati i dati dell'utente in merito alle costruzioni e alle unità in coda di produzione.
- **Utilizzo:** classe utilizzata per aggiornare i dati dell'utente riguardanti le costruzioni e le unità in coda di produzione.
- **Classi ereditate**
 - `Operation`

- Metodi

- `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`

Metodo per eseguire l'update dei dati dell'utente riguardanti le costruzioni e le unità in coda di produzione.

Argomenti:

- * `data` : dati accompagnatori alla richiesta dell'operazione.
- * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
- * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
- * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
- * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
- * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

- `-updateBuildings(building : BuildingPossession, userData : UserData) : void`

Update degli edifici.

Argomenti:

- * `building` : indica la BuildingPossession da aggiornare.
- * `userData` : sono i dati dell'utente che richiede l'operazione.

- `-updateUnits(building : BuildingWithLevel, unit : UnitInProgress, userData : UserData) : void`

Update delle unità.

Argomenti:

- * `building` : il tipo di edificio che produce le unità da aggiornare.
- * `unit` : indica la unitInProgress da aggiornare.
- * `userData` : l'userdata a cui le unità appartengono.

3.33.2.26 sgad::servertier::businesslogic::operations::UpgradeBuilding

UpgradeBuilding
+execute(userData : UserData, data : string, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, internalAuthorization : bool) : string
+upgradeBuilding(data : UserData, possession : BuildingPossession, level : BuildingWithLevel) : void

Figura 194: Diagramma della classe `sgad::servertier::businesslogic::operations::- UpgradeBuilding`

- **Descrizione:** classe che rappresenta l'operazione di avanzamento di livello di un edificio posseduto dall'utente.
 - **Utilizzo:** viene utilizzata per l'esecuzione dell'operazione di avanzamento di livello di un edificio posseduto dall'utente. Questa operazione, per poter portare a termine il proprio compito, utilizza un'altra operazione ad autorizzazione interna. Questa viene recuperata ed eseguita tramite OperationFactory.
 - **Classi ereditate**
 - `Operation`
 - **Metodi**
 - `+execute(data : string, internalAuthorization : bool, loginAuthorization : bool, registrationAuthorization : bool, userAuthorization : bool, userData : UserData) : string`
Metodo per l'esecuzione di un upgrade di un edificio.
- Argomenti:**
- * `data` : dati accompagnatori alla richiesta dell'operazione.
 - * `internalAuthorization` : autorizzazione a operare richieste interne. Di default è false.
 - * `loginAuthorization` : autorizzazione a operare richieste di login. Di default è false.
 - * `registrationAuthorization` : autorizzazione a operare richieste di registrazione. Di default è false.
 - * `userAuthorization` : autorizzazione a operare richieste di user. Di default è false.
 - * `userData` : dati dell'utente su cui verrà effettuata l'operazione.

– `+upgradeBuilding(data : UserData, level : BuildingWithLevel, possession : BuildingPossession) : void`

Metodo utilizzato dalla classe per effettuare l'upgrade, il metodo richiede che i test di consistenza con i dati siano stati eseguiti.

Argomenti:

* `data` : i dati utente di chi richieda l'upgrade.

* `level` : il modello della costruzione di livello superiore.

* `possession` : la BuildingPossession corrente.

3.33.2.27 sgad::servertier::businesslogic::operations::OperationFactory

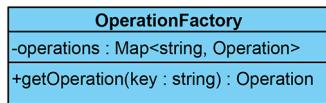


Figura 195: Diagramma della classe `sgad::servertier::businesslogic::operations::OperationFactory`

- **Descrizione:** classe per la gestione delle classi che realizzano `Operation` concretamente.

- **Utilizzo:** viene utilizzato per ottenere un riferimento all'operazione concreta.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione, Gestione di una richiesta di login e Gestione di una richiesta di gioco.

- **Attributi**

– `-operations : Map<string, Operation>`

Contiene le tutte le operazioni effettuabili dalla Logic.

- **Metodi**

– `+getOperation(key : string) : Operation`

Ritorna l'operazione identificata dalla stringa in input.

Argomenti:

* `key` : stringa identificativa dell'operazione richiesta.

3.34 Componente sgad::servertier::dataaccess

3.34.1 Informazioni sul package

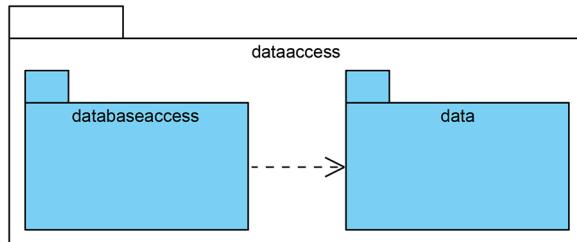


Figura 196: Diagramma della componente `sgad::servertier::dataaccess`

- **Descrizione:** componente per la gestione dell'accesso ad database e della creazione degli oggetti delle componenti shareddata e userdata.
- **Padre:** `servertier`
- **Package contenuti**
 - `data`: componente che gestisce l'insieme dei dati necessari per mantenere memorizzata, sul server, la sessione dell'utente.
 - `databaseaccess`: componente corrispondente al livello data access dell'architettura Three-Tier. Essa si occupa del mantenimento dei dati e della comunicazione con il database.

3.35 Componente sgad::servertier::dataaccess::data

3.35.1 Informazioni sul package

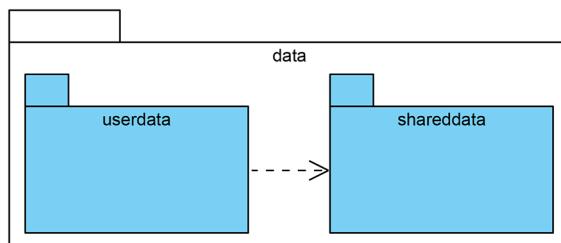


Figura 197: Diagramma della componente `sgad::servertier::dataaccess::data`

- **Descrizione:** componente che gestisce l'insieme dei dati necessari per mantenere memorizzata, sul server, la sessione dell'utente.
- **Padre:** `dataaccess`

- **Interazioni con altri componenti**

- **operations**: componente per la gestione dell'insieme delle possibili operazioni che un utente può effettuare.
- **databasemanager**: componente per la gestione dell'interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l'unico punto di accesso con il database.

- **Package contenuti**

- **shareddata**: componente per la gestione dei dati che non dipendono dall'utente ma dall'implementazione del mondo di gioco.
- **userdata**: componente per la gestione dei dati relativi alla sessione dell'utente mantenuta dal server.

3.36 Componente sgad::servtier::dataaccess::data::shareddata

3.36.1 Informazioni sul package

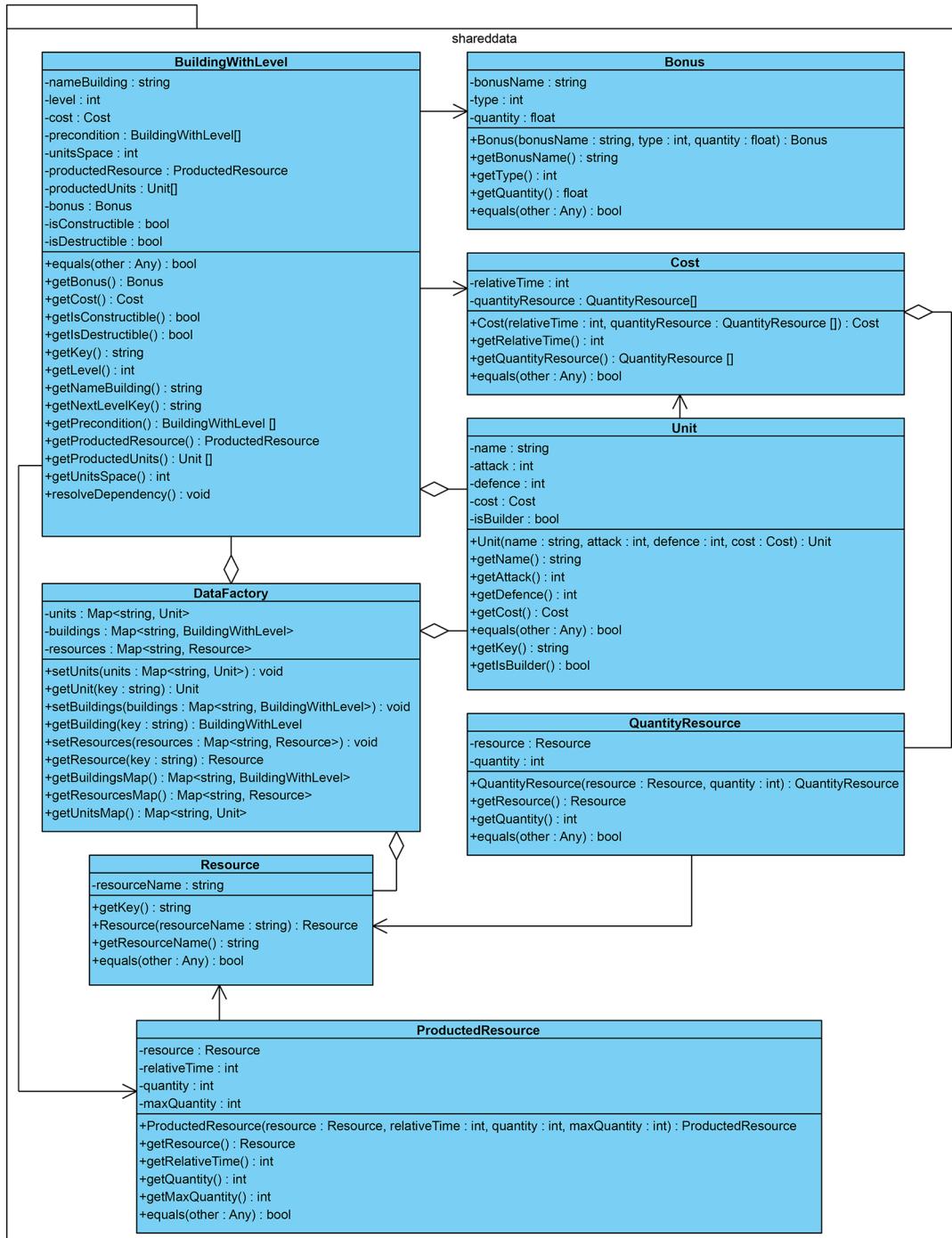


Figura 198: Diagramma della componente `sgad::servtier::dataaccess::data::shareddata`

- **Descrizione:** componente per la gestione dei dati che non dipendono dall'utente ma dall'implementazione del mondo di gioco.
- **Padre:** data
- **Interazioni con altri componenti**
 - **userdata:** componente per la gestione dei dati relativi alla sessione dell'utente mantenuta dal server.
 - **databasemanager:** componente per la gestione dell'interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l'unico punto di accesso con il database.
 - **shareddatadao:** componente per l'istanziazione delle classi del pacchetto shareddata a partire dai dati letti dal database.
 - **userdatadao:** componente per l'istanziazione delle classi del pacchetto userdata a partire dai dati letti dal database.

3.36.2 Classi

3.36.2.1 sgad::servertier::dataaccess::data::shareddata::Bonus

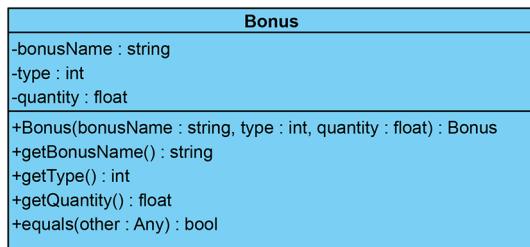


Figura 199: Diagramma della classe `sgad::servertier::dataaccess::data::shareddata::Bonus`

- **Descrizione:** classe per la gestione dei bonus eventualmente disponibili per certi edifici.
- **Utilizzo:** viene utilizzata per mantenere l'informazione sul nome del bonus, sul tipo e sulla quantità dello stesso.
- **Attributi**
 - `-bonusName : string`
Il nome del bonus.
 - `-quantity : float`
La potenza del bonus, cioè quanto influenza la caratteristica del Building-WithLevel.

– `-type : int`

Il codice identificativo del bonus, il quale determina quale caratteristica del BuildingWithLevel influenza.

• Metodi

– `+Bonus(bonusName : string, quantity : float, type : int) : Bonus`
Costruttore della classe.

Argomenti:

* `bonusName` : il nome del bonus.

* `quantity` : la potenza del bonus, cioè quanto influenza la caratteristica del BuildingWithLevel.

* `type` : il codice identificativo del bonus, il quale determina quale caratteristica del BuildingWithLevel influenza.

– `+getBonusName() : string`

Metodo getter per l'attributo bonusName.

– `+getType() : int`

Metodo getter per l'attributo type.

– `+getQuantity() : float`

Metodo getter per l'attributo quantity.

– `+equals(other : Any) : bool`

Metodo per il confronto.

Argomenti:

* `other` : l'altro oggetto con cui fare il confronto.

3.36.2.2 sgad::servertier::dataaccess::data::shareddata::BuildingWithLevel

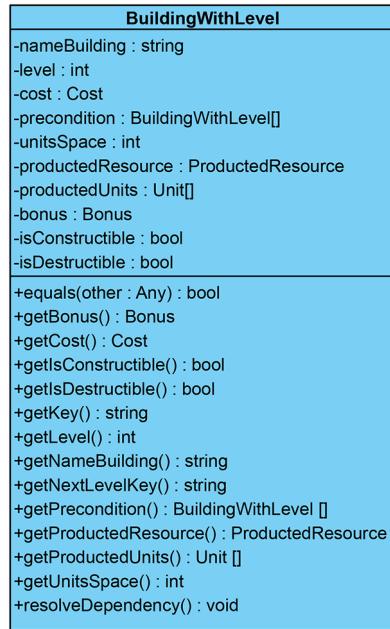


Figura 200: Diagramma della classe `sgad::servertier::dataaccess::data::-shareddata::BuildingWithLevel`

- **Descrizione:** classe per la gestione di un edificio ad un particolare livello.
- **Utilizzo:** viene utilizzata per memorizzare tutte le informazioni che descrivono un edificio ad un particolare livello.
- **Attributi**
 - `-bonus : Bonus`
Il bonus a disposizione della costruzione.
 - `-cost : Cost`
Le quantità ed il tipo di risorse necessarie.
 - `-isConstructible : bool`
Indica se l'edificio è direttamente edificabile su una cella dell'area di gioco.
 - `-isDestructible : bool`
Indica se l'edificio è rimovibile.
 - `-level : int`
Il livello della costruzione.
 - `-nameBuilding : string`
Il nome della costruzione.
 - `-precondition : BuildingWithLevel[]`
Le costruzioni necessarie per poter costruire questo edificio.
 - `-producedResource : ProductedResource`
La risorsa che questo edificio produce.

- `-productedUnits : Unit[]`
Le unità che questo edificio può produrre.

- `-unitsSpace : int`
I posti per le unità che questo edificio offre.

- Metodi

- `+getKey() : string`
Fornisce la chiave identificativa dell'edificio.

- `+getNextLevelKey() : string`
Fornisce la chiave identificativa dell'edificio di livello successivo.

- `+getNameBuilding() : string`
Metodo getter per l'attributo nameBuilding.

- `+getLevel() : int`
Metodo getter per l'attributo level.

- `+getCost() : Cost`
Metodo getter per l'attributo cost.

- `+getPrecondition() : BuildingWithLevel[]`
Metodo getter per l'attributo precondition.

- `+getUnitsSpace() : int`
Metodo getter per l'attributo unitsSpace.

- `+getProductedResource() : ProductedResource`
Metodo getter per l'attributo productedResource.

- `+getProductedUnits() : Unit[]`
Metodo getter per l'attributo productedUnits.

- `+getBonus() : Bonus`
Metodo getter per l'attributo bonus.

- `+getIsConstructible() : bool`
Metodo getter per l'attributo isConstructible.

- `+equals(other : Any) : bool`
Metodo equals che confronta i due oggetti dicendo se sono uguali.

Argomenti:

- * **other** : l'altro oggetto con cui si fa il confronto.
- **-resolveDependency()** : **void**
Risolve le dipendenze della Building, per ogni dipendenza prende dal DataFactory un riferimento all'oggetto al posto di tenere una stringa.
- **+getIsDestructible()** : **bool**
Ritorna vero o falso a seconda che l'edificio sia o non sia distruggibile.
- **+BuildingWithLevel(bonus : Bonus, cost : Cost, isConstructible : bool, isDestructible : bool, level : int, nameBuilding : string, precondition : Object[], producedResource : ProductedResource, producedUnits : Unit[], unitsSpace : int) : BuildingWithLevel**
Costruttore della classe.

Argomenti:

- * **bonus** : il bonus a disposizione della costruzione.
- * **cost** : le quantità ed il tipo di risorse necessarie.
- * **isConstructible** : indica se l'edificio è direttamente edificabile su una cella dell'area di gioco.
- * **isDestructible** : indica se l'edificio è rimovibile.
- * **level** : il livello della costruzione.
- * **nameBuilding** : il nome della costruzione.
- * **precondition** : le costruzioni necessarie per poter costruire questo edificio.
- * **producedResource** : la risorsa che questo edificio produce.
- * **producedUnits** : le unità che questo edificio può produrre.
- * **unitsSpace** : i posti per le unità che questo edificio offre.

3.36.2.3 sgad::servertier::dataaccess::data::shareddata::Cost

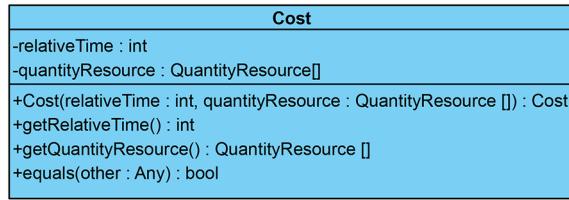


Figura 201: Diagramma della classe `sgad::servtier::dataaccess::data::shareddata::Cost`

- **Descrizione:** classe per la gestione di un costo in termini di risorse e tempo.
- **Utilizzo:** viene utilizzata per la gestione di un costo in termini di quantità di risorse.
- **Attributi**

- `-quantityResource : QuantityResource[]`
Le quantità e il tipo di risorse necessarie.
- `-relativeTime : int`
Il tempo necessario espresso in secondi.

- **Metodi**

- `+Cost(quantityResource : QuantityResource[], relativeTime : int) :- Cost`
Costruttore della classe.

Argomenti:

- * `quantityResource` : le quantità e il tipo di risorse necessarie.
- * `relativeTime` : il tempo necessario espresso in secondi.

- `+getRelativeTime() : int`
Metodo getter per l'attributo relativeTime.

- `+getQuantityResource() : QuantityResource[]`
Metodo getter per l'attributo quantityResource

- `+equals(other : Any) : bool`
Metodo per controllare l'uguaglianza di due Costi.

Argomenti:

- * `other` : il costo con cui confrontarsi.

3.36.2.4 sgad::servertier::dataaccess::data::shareddata::DataFactory

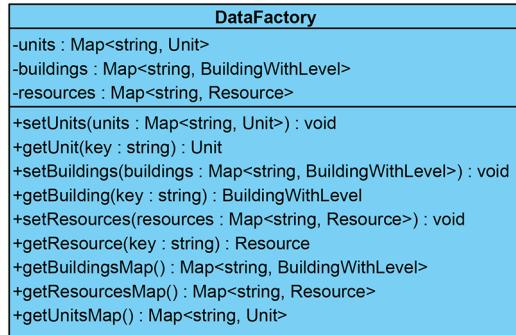


Figura 202: Diagramma della classe `sgad::servertier::dataaccess::data::shareddata::DataFactory`

- **Descrizione:** classe che ritorna un'istanza delle componenti di gioco.
- **Utilizzo:** viene utilizzata per ottenere riferimenti agli oggetti che compongono la sua stessa componente.
- **Attributi**
 - `-buildings : Map<string, BuildingWithLevel>`
Le BuildingWithLevel disponibili nel gioco.
 - `-resources : Map<string, Resource>`
Le Resource disponibili nel gioco.
 - `-units : Map<string, Unit>`
Le Unit disponibili nel gioco.
- **Metodi**
 - `+getBuilding(key : string) : BuildingWithLevel`
Fornisce un oggetto della classe BuildingWithLevel corrispondente alla key passata.

Argomenti:

 - * `key` : chiave identificativa della BuildingWithLevel desiderata.
 - `+getUnit(key : string) : Unit`
Ritorna la Unit corrispondente alla key in input.

Argomenti:

 - * `key` : l'identificativo dell'unità richiesta.

- `+getResource(key : string) : Resource`
Ritorna la Resource corrispondente alla key in input.

Argomenti:

* `key` : chiave identificativa della Resource desiderata.

- `+setUnits(units : Map<string, Unit>) : void`
Setta le Unit disponibili nel gioco.

Argomenti:

* `units` : le unità disponibili nel gioco.

- `+setResources(resources : Map<string, Resource>) : void`
Setta le Resource disponibili nel gioco.

Argomenti:

* `resources` : le risorse disponibili nel gioco.

- `+setBuildings(buildings : Map<string, BuildingWithLevel>) : void`
Setta le BuldingWithLevel disponibili nel gioco.

Argomenti:

* `buildings` : le costruzioni disponibili nel gioco.

- `+getResourcesMap() : Map<string, Resource>`
Metodo get utilizzato dai $DAO_{|g|}$ per ricevere l'intera mappa di risorse.

- `+getUnitsMap() : Map<string, Unit>`
Metodo utilizzato dal DAO per recuperare la mappa delle unità.

- `-getBuildingsMap() : Map<string, BuildingWithLevel>`
Metodo get utilizzato dai DAO per ricevere l'intera mappa di edifici.

3.36.2.5 sgad::servertier::dataaccess::data::shareddata::ProducedResource

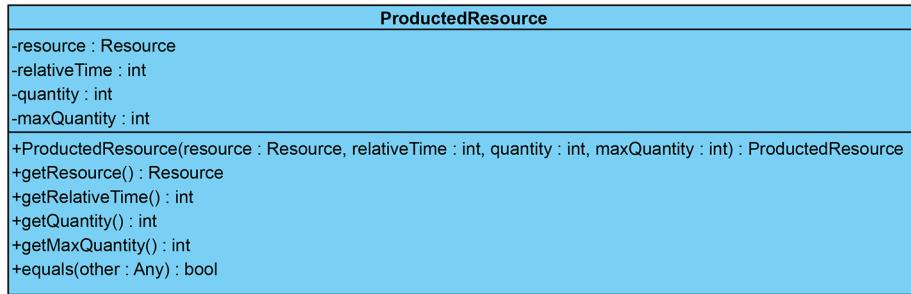


Figura 203: Diagramma della classe `sgad::servertier::dataaccess::data::shareddata::ProducedResource`

- **Descrizione:** classe per la gestione delle informazioni relative alla risorsa prodotta da un edificio.

- **Utilizzo:** viene utilizzata per la memorizzazione delle informazioni relative alla risorsa prodotta da un edificio.

- **Attributi**

- `-maxQuantity : int`
Il massimo numero di unità di risorse accumulabili nell'edificio.
- `-quantity : int`
Il numero di risorse prodotte ogni multiplo di `relativeTime` che passa.
- `-relativeTime : int`
Indica ogni quanto l'edificio produce `quantity` unità di risorsa `resource`.
- `-resource : Resource`
La risorsa prodotta.

- **Metodi**

- `+ProducedResource(maxQuantity : int, quantity : int, relativeTime : int, resource : Resource) : ProducedResource`
Costruttore della classe.

Argomenti:

- * `maxQuantity` : il massimo numero di unità di risorse accumulabili nell'edificio.
- * `quantity` : il numero di risorse prodotte ogni multiplo di `relativeTime` che passa.
- * `relativeTime` : indica ogni quanto l'edificio produce `quantity` unità di risorsa `resource`.

- * `resource` : la risorsa prodotta.
- `+getResource() : Resource`
Metodo getter per l'attributo resource.
- `+getRelativeTime() : int`
Metodo getter per l'attributo relativeTime.
- `+getQuantity() : int`
Metodo getter per l'attributo quantity.
- `+getMaxQuantity() : int`
Metodo getter per l'attributo maxQuantity.
- `+equals(other : Any) : bool`
Metodo per controllare l'uguaglianza.

Argomenti:

- * `other` : un ProductedResource da controllare.

3.36.2.6 sgad::servtier::dataaccess::data::shareddata::QuantityResource

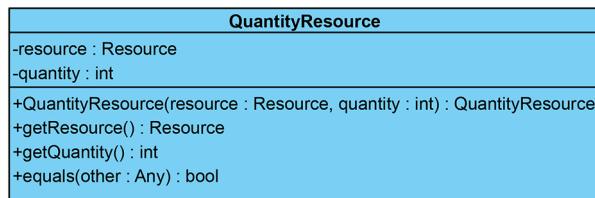


Figura 204: Diagramma della classe `sgad::servtier::dataaccess::data::shareddata::QuantityResource`

- **Descrizione:** classe per la gestione della quantità di una particolare risorsa.
- **Utilizzo:** viene utilizzata per associare una quantità ad una risorsa.
- **Attributi**
 - `-quantity : int`
Il numero di unità della data risorsa.
 - `-resource : Resource`
La risorsa.
- **Metodi**

- `+QuantityResource(quantity : int, resource : Resource) : QuantityResource`
Costruttore della classe.

Argomenti:

- * `quantity` : il numero di unità della data risorsa.
- * `resource` : la risorsa di cui si vuole memorizzare la quantità.
- `+getQuantity() : int`
Metodo getter per l'attributo quantity.
- `+getResource() : Resource`
Metodo getter per l'attributo resource.
- `+equals(other : Any) : bool`
Metodo per controllare l'uguaglianza.

Argomenti:

- * `other` : un'altra quantità.

3.36.2.7 sgad::servtier::dataaccess::data::shareddata::Resource

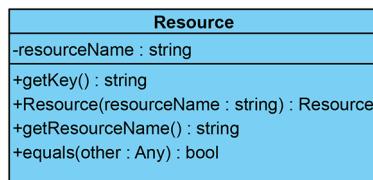


Figura 205: Diagramma della classe `sgad::servtier::dataaccess::data::shareddata::Resource`

- **Descrizione:** classe che rappresenta una risorsa.
- **Utilizzo:** viene utilizzata per memorizzare le informazioni su una risorsa.
- **Attributi**
 - `-resourceName` : string
Il nome della risorsa.
- **Metodi**
 - `+Resource(resourceName : string) : Resource`
Costruttore della classe.

Argomenti:

- * **resourceName** : il nome della risorsa.
- **+equals(risorsa : Any) : bool**
Verifica se due risorse sono uguali.

Argomenti:

- * **risorsa** : la risorsa con cui verificare l'uguaglianza.
- **+get resourceName() : string**
Metodo getter per l'attributo resourceName.
- **+getKey() : string**
Fornisce la chiave identificativa della risorsa.

3.36.2.8 sgad::servertier::dataaccess::data::shareddata::Unit

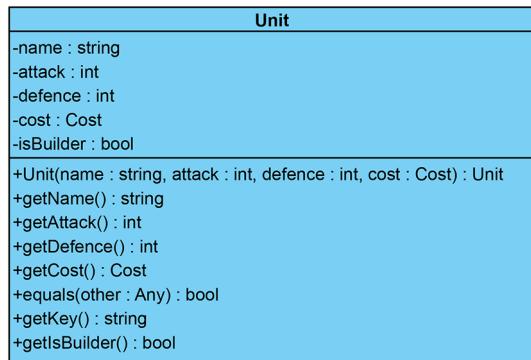


Figura 206: Diagramma della classe `sgad::servertier::dataaccess::data::shareddata::Unit`

- **Descrizione:** classe per la gestione delle informazioni di una unità.
- **Utilizzo:** viene utilizzata per la memorizzazione delle informazioni sulle unità.
- **Attributi**
 - **-attack : int**
La forza di attacco dell'unità.
 - **-cost : Cost**
Il costo di produzione dell'unità.

- `-defence : int`
La forza di difesa dell'unità.
- `-isBuilder : bool`
Identifica le unità che permettono di costruire.
- `-name : string`
Il nome dell'unità.

- Metodi

- `+Unit(attack : int, cost : Cost, defence : int, name : string) :- Unit`
Costruttore della classe.

Argomenti:
 - * `attack` : la forza di attacco dell'unità.
 - * `cost` : il costo di produzione dell'unità.
 - * `defence` : la forza di difesa dell'unità.
 - * `name` : il nome dell'unità.
- `+getName() : string`
Metodo getter per l'attributo name.
- `+getCost() : Cost`
Metodo getter per l'attributo cost.
- `+getDefence() : int`
Metodo getter per l'attributo defence.
- `+getAttack() : int`
Metodo getter per l'attributo attack.
- `+getKey() : string`
Fornisce la chiave identificativa dell'unità.
- `+equals(other : Any) : bool`
Metodo per controllare l'uguaglianza.

Argomenti:

- * `other` : l'altro oggetto con cui si fa il confronto.

3 Specifica dei componenti

– `+getIsBuilder() : bool`

Metodo getter per l'attributo isBuilder.

3.37 Componente `sgad::servertier::dataaccess::data::userdata`

3.37.1 Informazioni sul package

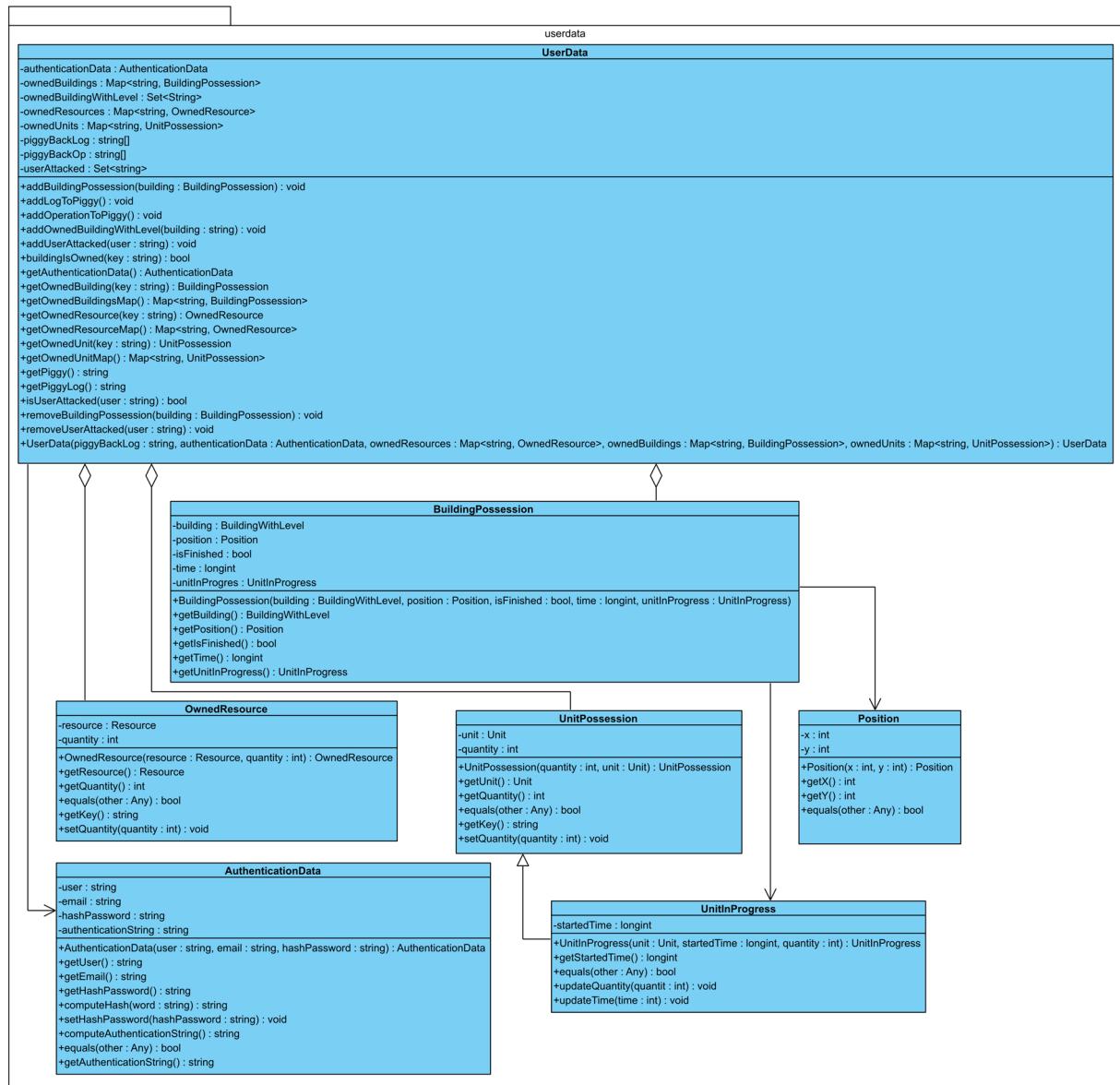


Figura 207: Diagramma della componente `sgad::servertier::dataaccess::data::userdata`

- **Descrizione:** componente per la gestione dei dati relativi alla sessione dell’utente mantenuta dal server.
- **Padre:** data
- **Interazioni con altri componenti**
 - **shareddata:** componente per la gestione dei dati che non dipendono dall’utente ma dall’implementazione del mondo di gioco.
 - **userdatadao:** componente per l’istanziazione delle classi del pacchetto userdata a partire dai dati letti dal database.

3.37.2 Classi

3.37.2.1 sgad::servertier::dataaccess::data::userdata::AuthenticationData

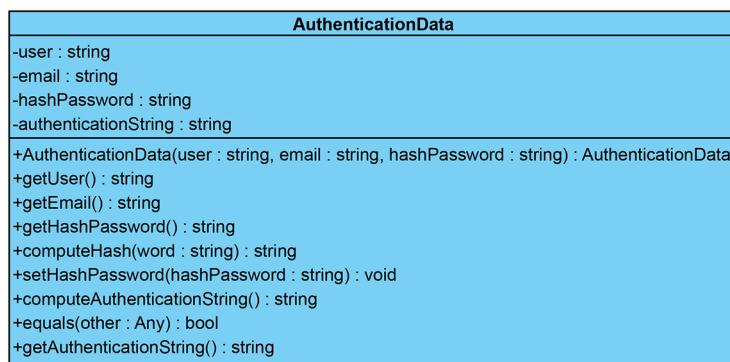


Figura 208: Diagramma della classe `sgad::servertier::dataaccess::data::userdata::AuthenticationData`

- **Descrizione:** classe per la gestione delle informazioni personali e di accesso di un utente.
- **Utilizzo:** viene utilizzata istanziando un oggetto che memorizza le informazioni.
- **Attributi**
 - `-authenticationString : string`
La stringa di autenticazione che il client utilizzerà per autenticare le richieste di gioco.
 - `-email : string`
L'email dell'utente.
 - `-hashPassword : string`
L'hash della password dell'utente.
 - `-user : string`
Il nome utente univoco dell'utente.

- Metodi

- `+AuthenticationData(email : string, hashPassword : string, user : string) : AuthenticationData`
Costruttore della classe.

Argomenti:

- * `email` : l'email dell'utente.
 - * `hashPassword` : l'hash della password dell'utente.
 - * `user` : il nome utente univoco dell'utente.

- `+getUser() : string`
Metodo getter per l'attributo user.

- `+getEmail() : string`
Metodo getter per l'attributo email.

- `+getHashPassword() : string`
Metodo getter per l'attributo hashPassword.

- `+setHashPassword(hashPassword : string) : void`
Metodo setter per l'attributo hashPassword.

Argomenti:

- * `hashPassword` : nuovo hash della password di accesso dell'utente.
 - `+computeHash(word : string) : string`
Il metodo calcola l'hash SHA-1 della stringa in input.

Argomenti:

- * `word` : la stringa input della quale il metodo calcola l'hash SHA-1.
 - `-computeAuthenticationString() : string`
La stringa di autenticazione che il client utilizzerà per autenticare le richieste.

- `+equals(other : Any) : bool`
Metodo per il confronto.

Argomenti:

- * `other` : altro oggetto da controllare.

– `+getAuthenticationString() : string`

Metodo getter per l'attributo authenticationString. Ritorna la stringa di autenticazione che il client utilizzerà per autenticare le richieste.

3.37.2.2 sgad::servertier::dataaccess::data::userdata::BuildingPossession

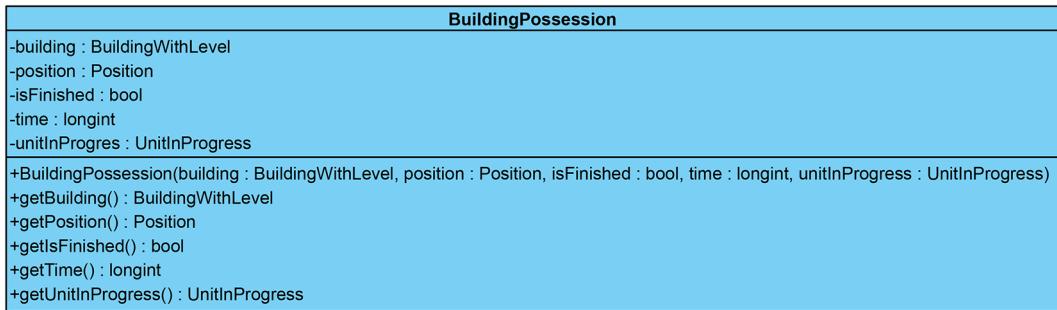


Figura 209: Diagramma della classe `sgad::servertier::dataaccess::data::userdata::BuildingPossession`

- **Descrizione:** classe per la gestione di un edificio posseduto da un utente.
- **Utilizzo:** viene utilizzata per la memorizzazione delle informazioni riguardanti un edificio posseduto da un utente.
- **Attributi**

– `-building : BuildingWithLevel`

Il tipo di costruzione.

– `-isFinished : bool`

Indica se la costruzione dell'edificio è completa.

– `-position : Position`

Indica le coordinate in cui l'edificio è posizionato nel mondo di gioco.

– `-time : longint`

Indica una data con orario preciso al secondo. Se l'edificio è completamente costruito indica il momento in cui è stata effettuata l'ultima raccolta delle risorse prodotte. Se l'edificio è in costruzione indica il momento in cui la costruzione è iniziata.

– `-unitInProgress : UnitInProgress`

Indica le eventuali unità in costruzione presso l'edificio.

- **Metodi**

– `+BuildingPossession(building : BuildingWithLevel, isFinished : bool, position : Position, time : longint, unitInProgress : UnitInProgress) : BuildingPossession`

Costruttore della classe.

Argomenti:

- * **building** : il tipo di costruzione.
- * **isFinished** : indica se la costruzione dell'edificio è completa.
- * **position** : indica le coordinate in cui l'edificio è posizionato nel mondo di gioco.
- * **time** : indica una data con orario preciso al secondo. Se l'edificio è completamente costruito indica il momento in cui è stata effettuata l'ultima raccolta delle risorse prodotte. Se l'edificio è in costruzione indica il momento in cui la costruzione è iniziata.
- * **unitInProgress** : indica le eventuali unità in costruzione presso l'edificio.

– **+getBuilding()** : **BuildingWithLevel**
Metodo getter per l'attributo building.

– **+getPosition()** : **Position**
Metodo getter per l'attributo position.

– **+getIsFinished()** : **bool**
Metodo getter per l'attributo isFinished.

– **+getTime()** : **longint**
Metodo getter per l'attributo time.

– **+getUnitInProgress()** : **UnitInProgress**
Metodo getter per l'attributo unitInProgress.

– **+equals(other : Any)** : **bool**
Metodo di confronto.

Argomenti:

- * **other** : oggetto con cui fare il confronto.
- **+getKey()** : **string**
Ritorna la chiave univoca.
- **+setTime(newTime : long)** : **void**
Metodo setter per l'attributo Time.

Argomenti:

* `newTime` : il tempo, che deve essere maggiore di quello corrente.

– `+setIsFinished(stato : bool) : void`

Metodo setter per l'attributo `isFinished`.

Argomenti:

* `stato` : metodo setter per l'attributo `isFinished`.

– `+addUnit(unitInProgress : UnitInProgress) : bool`

Metodo per aggiungere unità alla coda di produzione.

Argomenti:

* `unitInProgress` : le unità da aggiungere.

– `+setBuilding(newBuilding : BuildingWithLevel) : void`

Metodo setter per l'attributo `building`.

Argomenti:

* `newBuilding` : la nuova costruzione.

3.37.2.3 sgad::servertier::dataaccess::data::userdata::OwnedResource

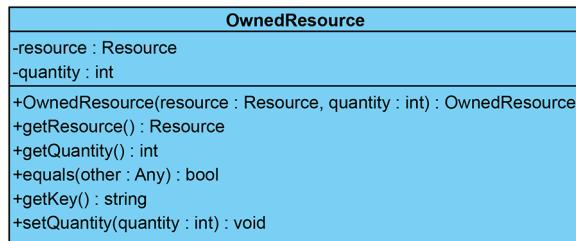


Figura 210: Diagramma della classe `sgad::servertier::dataaccess::data::userdata::OwnedResource`

- **Descrizione:** classe che rappresenta una risorsa posseduta da un utente.
- **Utilizzo:** viene utilizzata per la memorizzazione di una risorsa posseduta da un utente.
- **Attributi**
 - `-quantity : int`
La quantità di risorsa posseduta.

- **-resource : Resource**
Il tipo di risorsa.

- **Metodi**

- **+OwnedResource(quantity : int, resource : Resource) : OwnedResource**
Costruttore della classe.

Argomenti:

- * **quantity** : la quantità di risorsa posseduta.
- * **resource** : il tipo di risorsa.

- **+getResource() : Resource**
Metodo getter per l'attributo resource.

- **+getQuantity() : int**
Metodo getter per l'attributo quantity.

- **+setQuantity(quantity : int) : void**
Metodo setter per l'attributo quantity.

Argomenti:

- * **quantity** : nuovo valore per l'attributo, deve essere maggiore di zero.

- **+equals(other : Any) : bool**
Metodo per il confronto.

Argomenti:

- * **other** : l'oggetto con cui fare il confronto.

- **+getKey() : string**
Ritorna la chiave univoca.

3.37.2.4 sgad::servertier::dataaccess::data::userdata::Position

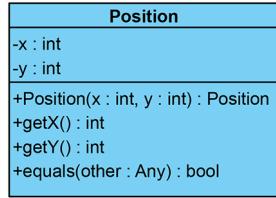


Figura 211: Diagramma della classe `sgad::servertier::data::-userdata::Position`

- **Descrizione:** classe per la gestione di una posizione sulla griglia del villaggio.
- **Utilizzo:** viene utilizzata per memorizzare la posizione di un edificio posseduto da un utente.
- **Attributi**

– `-x` : `int`
 La coordinata x.

– `-y` : `int`
 La coordinata y.

- **Metodi**

– `+Position(x : int, y : int) : Position`
 Costruttore della classe.

Argomenti:

* `x` : la coordinata x.

* `y` : la coordinata y.

– `+getX() : int`
 Metodo getter per l'attributo x.

– `+getY() : int`
 Metodo getter per l'attributo y.

– `+equals(other : Any) : bool`
 Metodo di confronto.

Argomenti:

* `other` : altro oggetto con cui fare il confronto.

3.37.2.5 sgad::servertier::dataaccess::data::userdata::UnitPossession

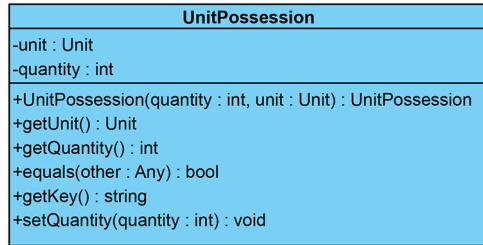


Figura 212: Diagramma della classe `sgad::servertier::dataaccess::data::userdata::UnitPossession`

- **Descrizione:** classe per la rappresentazione di un'unità con associata una quantità.
- **Utilizzo:** viene utilizzata per memorizzare per ogni tipo di unità la quantità di queste possedute da un utente.
- **Sottoclassi**
 - `UnitInProgress`
- **Attributi**
 - `-quantity : int`
La quantità di unità.
 - `-unit : Unit`
Il tipo di unità.
- **Metodi**
 - `+UnitPossession(quantity : int, unit : Unit) : UnitPossession`
Costruttore della classe.
 - Argomenti:**
 - * `quantity` : la quantità di unità.
 - * `unit` : il tipo di unità.
 - `+getUnit() : Unit`
Metodo getter per l'attributo `unit`.
 - `+getQuantity() : int`
Metodo getter per l'attributo `quantity`.

– `+setQuantity(quantity : int) : void`
 Metodo setter per l'attributo quantity.

Argomenti:

* `quantity` : la quantità di unità.

– `+equals(other : Any) : bool`
 Metodo di confronto.

Argomenti:

* `other` : oggetto con cui fare il confronto.

– `+getKey() : string`
 Ritorna la chiave univoca.

3.37.2.6 sgad::servertier::dataaccess::data::userdata::UnitInProgress

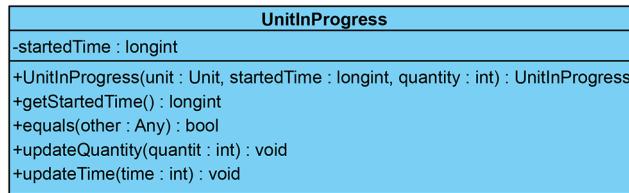


Figura 213: Diagramma della classe `sgad::servertier::dataaccess::data::userdata::UnitInProgress`

- **Descrizione:** classe per la gestione della coda di costruzione di unità di un particolare edificio.
- **Utilizzo:** viene utilizzata per gestire le code di costruzione degli edifici completi posseduti da un particolare utente.
- **Classi ereditate**
 - `UnitPossession`
- **Attributi**
 - `-startedTime : longint`
 La data con orario preciso al secondo in cui la costruzione di unità è iniziata.
- **Metodi**

- `+UnitInProgress(quantity : int, startTime : longint, unit : Unit)`
: `UnitInProgress`
Costruttore della classe.

Argomenti:

- * `quantity` : la quantità di unità.
 - * `startTime` : la data con orario preciso al secondo in cui la costruzione di unità è iniziata.
 - * `unit` : il tipo di unità.
- `+getStartTime() : longint`
Metodo getter per l'attributo startTime.
 - `+equals(other : Any) : bool`
Metodo per il confronto.

Argomenti:

- * `other` : l'oggetto da confrontare.
- `+updateTime(time : int) : void`
Metodo utilizzato per fare l'update della quantità rimanente.

Argomenti:

- * `time` : tempo da aggiungere a startTime.
- `+updateQuantity(quantity : int) : void`
Metodo utilizzato per fare l'update della quantità rimanente.

Argomenti:

- * `quantity` : la quantità di unità da sottrarre, deve essere maggiore di zero.

3.37.2.7 sgad::servertier::dataaccess::data::userdata::UserData

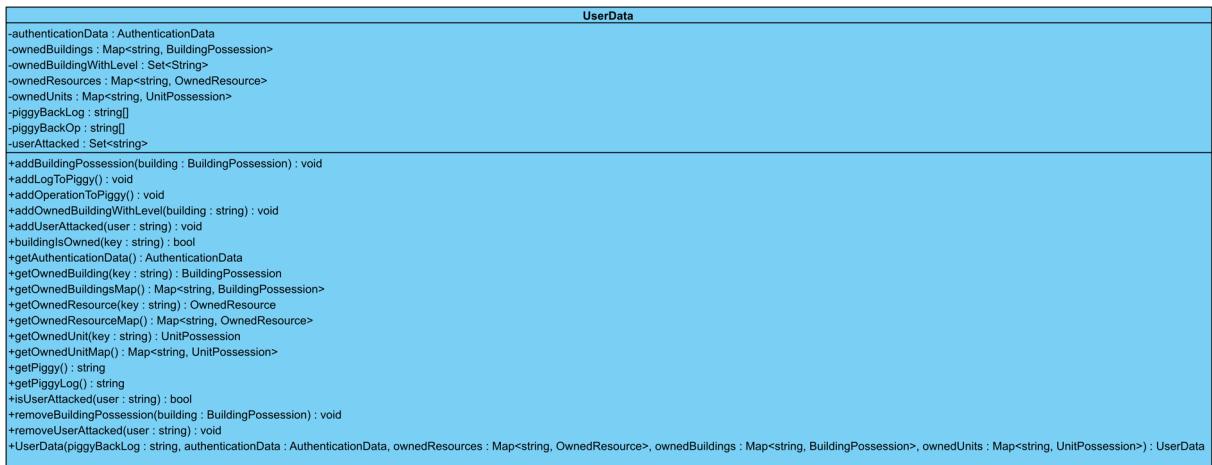


Figura 214: Diagramma della classe `sgad::servtier::dataaccess::data::-userdata::UserData`

- **Descrizione:** classe per la gestione dei dati di un utente.

- **Utilizzo:** viene utilizzata per la gestione dei dati di un utente.

- **Attributi**

- `-authenticationData : AuthenticationData`
I dati personali dell’utente.
- `-ownedBuildings : Map<string, BuildingPossession>`
Gli edifici presenti nel villaggio dell’utente.
- `-ownedBuildingWithLevel : Set<String>`
Attributo di possessione del tipo di costruzione.
- `-ownedResources : Map<string, OwnedResource>`
Le risorse possedute dall’utente.
- `-ownedUnits : Map<string, UnitPossession>`
Le unità possedute dall’utente.
- `-piggyBackLog : string[]`
Attributo relativo ai piggyBack di log.
- `-piggyBackOp : string[]`
Attributo relativo ai piggyBack di operazioni.
- `-userAttacked : Set<string>`
Rappresenta l’insieme degli utenti attaccati con successo.

- **Metodi**

- `+getOwnedResource(key : string) : OwnedResource`
Fornisce la risorsa posseduta in base alla chiave determinata.

Argomenti:

* `key` : chiave identificativa della risorsa posseduta.

– `+getOwnedUnit(key : string) : UnitPossession`

Ritorna la UnitPossession corrispondente alla key in input.

Argomenti:

* `key` : chiave identificativa dell'unità posseduta.

– `+getOwnedBuilding(key : string) : BuildingPossession`

Fornisce l'edificio posseduto in base alla chiave determinata.

Argomenti:

* `key` : chiave identificativa dell'edificio posseduto.

– `+buildingIsOwned(key : string) : bool`

Metodo per controllare la possessione di un edificio.

Argomenti:

* `key` : la chiave dell'edificio in questione.

– `+removeBuildingPossession(building : BuildingPossession) : void`

Metodo per rimuovere una Building da quelle possedute.

Argomenti:

* `building` : la Costruzione da distruggere.

– `+addOwnedBuildingWithLevel(building : string) : void`

Metodo per aggiungere un elemento ai tipi di edificio posseduti.

Argomenti:

* `building` : la nuova costruzione.

– `+getAuthenticationData() : AuthenticationData`

Il relativo AuthenticationData.

– `+getOwnedBuildingsMap() : Map<string, BuildingPossession>`

Metodo getter per il DAO che restituisce la collezione ownedBuildings.

– `+getOwnedResourceMap() : Map<string, OwnedResource>`

Metodo getter per il DAO che restituisce la collezione ownedResources.

– `+getOwnedUnitMap() : Map<string, UnitPossession>`

Metodo getter per il DAO che restituisce la collezione ownedUnits.

- `+addBuildingPossession(building : BuildingPossession) : void`
Metodo per aggiungere una Building a quelle possedute.

Argomenti:

* `building` : la nuova Costruzione.

- `+addUserAttacked(user : string) : void`
Aggiunge all'insieme degli utenti attaccati con successo l'utente in input.

Argomenti:

* `user` : utente attaccato con successo.

- `+removeUserAttacked(user : string) : void`
Rimuove un utente dalla lista degli utenti attaccati con successo.

Argomenti:

* `user` : utente da rimuovere.

- `+isUserAttacked(user : string) : bool`
Verifica se l'utente in input è stato attaccato con successo.

Argomenti:

* `user` : utente da verificare se attaccato con successo.

- `+getPiggy() : string[]`
Metodo utilizzato per ricevere la mappa dei messaggi di piggy.

- `+getPiggyLog() : string[]`
Metodo di get per ricevere solo i piggy di Log.

- `+addOperationToPiggy(operation : string) : void`
Metodo utilizzato per aggiungere operazioni al piggy.

Argomenti:

* `operation` : l'operazione da aggiungere al piggy.

- `+addLogToPiggy(log : string) : void`
Metodo utilizzato per aggiungere un log al piggy.

Argomenti:

* `log` : una operazione di log.

```

  - +UserData(authenticationData : AuthenticationData, ownedBuildings :
    Map<string, BuildingPossession>, ownedResources : Map<string, OwnedResource>,
    ownedUnits : Map<string, UnitPossession>, piggyBackLog : string[])
    : UserData
      Costruttore della classe.
  
```

Argomenti:

- * `authenticationData` : i dati personali dell'utente.
- * `ownedBuildings` : gli edifici presenti nel villaggio dell'utente.
- * `ownedResources` : le risorse possedute dall'utente.
- * `ownedUnits` : le unità possedute dall'utente.
- * `piggyBackLog` : attributo relativo ai piggyBack di log.

3.38 Componente sgad::servtier::dataaccess::databaseaccess

3.38.1 Informazioni sul package

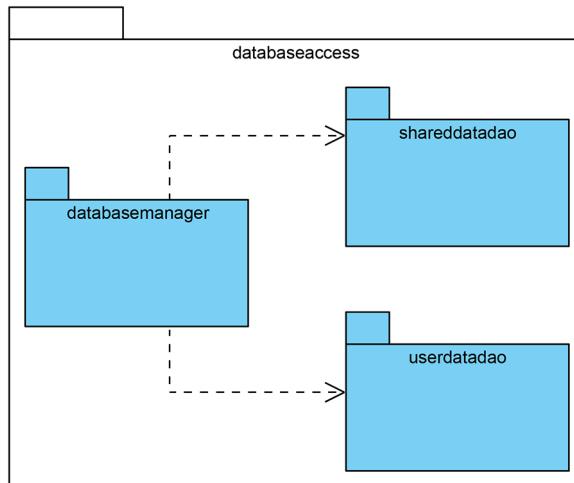


Figura 215: Diagramma della componente `sgad::servtier::dataaccess::-databaseaccess`

- **Descrizione:** componente corrispondente al livello data access dell'architettura Three-Tier. Essa si occupa del mantenimento dei dati e della comunicazione con il database.
- **Padre:** `dataaccess`

- **Interazioni con altri componenti**

- **casbah**.
- **application**: componente che si occupa del bootstrap dell'applicazione.
- **businesslogic**: componente corrispondente al livello business logic dell'architettura Three-Tier. Essa si occupa di effettuare le operazioni richieste dagli utenti. Nel caso le operazioni siano relative a interazioni tra più utenti, comunicherà con la componente **presentation** per instradare correttamente la richiesta verso gli altri client coinvolti.

- **Package contenuti**

- **databasemanager**: componente per la gestione dell'interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l'unico punto di accesso con il database.
- **shareddatadao**: componente per l'istanziazione delle classi del pacchetto shareddata a partire dai dati letti dal database.
- **userdatadao**: componente per l'istanziazione delle classi del pacchetto userdata a partire dai dati letti dal database.

3.39 Componente sgad::servertier::dataaccess::databaseaccess::database- manager

3.39.1 Informazioni sul package



Figura 216: Diagramma della componente sgad::servertier::dataaccess::-databaseaccess::databasemanager

- **Descrizione:** componente per la gestione dell’interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l’unico punto di accesso con il database.
- **Padre:** databaseaccess
- **Interazioni con altri componenti**
 - casbah.
 - logic: componente che gestisce il controllo delle operazioni richieste dal client.
 - data: componente che gestisce l’insieme dei dati necessari per mantenere memorizzata, sul server, la sessione dell’utente.
 - shareddata: componente per la gestione dei dati che non dipendono dall’utente ma dall’implementazione del mondo di gioco.
 - shareddatadao: componente per l’istanziazione delle classi del pacchetto shareddata a partire dai dati letti dal database.

- `userdatadao`: componente per l’istanziazione delle classi del pacchetto `userdata` a partire dai dati letti dal database.

3.39.2 Classi

3.39.2.1 `sgad::servtier::dataaccess::databaseaccess::databasemanager::DataBaseManager`



Figura 217: Diagramma della classe `sgad::servtier::dataaccess::databaseaccess::databasemanager::DataBaseManager`

- **Descrizione:** classe per la gestione del database. Essa astrae l’utilizzo di un particolare database al resto dell’applicativo.
- **Utilizzo:** viene utilizzata per la connessione al database e l’esecuzione di operazioni $CRUD_{|g|}$ su di esso. Astrae l’implementazione della base di dati al resto dell’applicazione.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione e Gestione di una richiesta di login.

- **Attributi**

- `-connectionTimeoutMS : int`

Valore di default per il timeout connessione.

- `-databaseName : string`

Nome del database di default per la connessione al database.

- `-databaseUrl : string`
Socket di default (sintassi: nomeHost:numeroPorta) del database di default per la connessione al database.
- `-db : MongoDB`
Riferimento al database specificato.
- `-fileName : string`
Stringa contenente il nome del file che contiene le impostazioni da caricare.
- `-inputStream : FileInputStream`
Oggetto InputStream nel quale caricare il file di configurazione.
- `-logger : Logger`
Parametro per la gestione dei messaggi di log.
- `-mongoClient : MongoClient`
Riferimento al DBMS_{|g|}.
- `-networkErrorLog : string`
Messaggio di errore.
- `-password : string`
Password di default per la connessione al database.
- `-properties : Properties`
Oggetto Properties per la lettura delle impostazioni dal file di configurazione.
- `-socketTimeoutMS : int`
Valore di default per il timeout socket.
- `-uri : string`
Stringa contenente l'URI di connessione al database.
- `-userName : string`
Nome utente di default per la connessione al database.

- Metodi

- `+save(userdata : UserData) : bool`
Salva sul database i dati aggiornati di un utente.

Argomenti:

* `userdata` : oggetto da salvare sul database.

- `+connect() : bool`
Avvia la connessione al database leggendo i dati generali di gioco e settandoli nel DataFactory.
- `-loadResources() : bool`
Esegue una `query|g|` sul database e setta le risorse in DataFactory.

- `-loadUnits() : bool`
Esegue una query sul database e setta le unità in DataFactory.
- `-loadBuildings() : bool`
Esegue una query sul database e setta le building in DataFactory.
- `-checkConfigurationFile(prop : Properties) : bool`
Controlla che il nome del database ed il suo socket indicati nel file di configurazione non siano nulli.

Argomenti:

- * `prop` : il file di configurazione in input che il metodo dovrà verificare.
- `+internalLoadUserData(user : string) : Set<UserData, int>`
Esegue una query sul database e ritorna un oggetto UserData corrispondente all'utente in input senza controllare la password.

Argomenti:

- * `user` : username dell'utente da caricare.
- `+loadUserData(hashPassword : string, user : string) : Set<UserData, int>`
Esegue una query sul database e ritorna un oggetto UserData corrispondente all'utente in input.

Argomenti:

- * `hashPassword` : hash della password dell'utente da caricare.
- * `user` : username dell'utente da caricare.
- `+loadRandomUsers(number : int, user : string) : List<String>`
Esegue una query sul database ritornando una lista di utenti random.

Argomenti:

- * `number` : numero di utenti random che si vogliono visualizzare.
- * `user` : username dell'utente che invoca il metodo.

- `+insertNewUser(userdata : UserData) : int`
Inserisce sul database i dati di un nuovo utente.

Argomenti:

- * `userdata` : oggetto da salvare nel database.

- `+deleteUser(user : string) : bool`
Elimina definitivamente un utente dal database.

Argomenti:

* `user` : username identificativo dell'utente da eliminare.

3.40 Componente sgad::servertier::dataaccess::databaseaccess::shareddataadao

3.40.1 Informazioni sul package

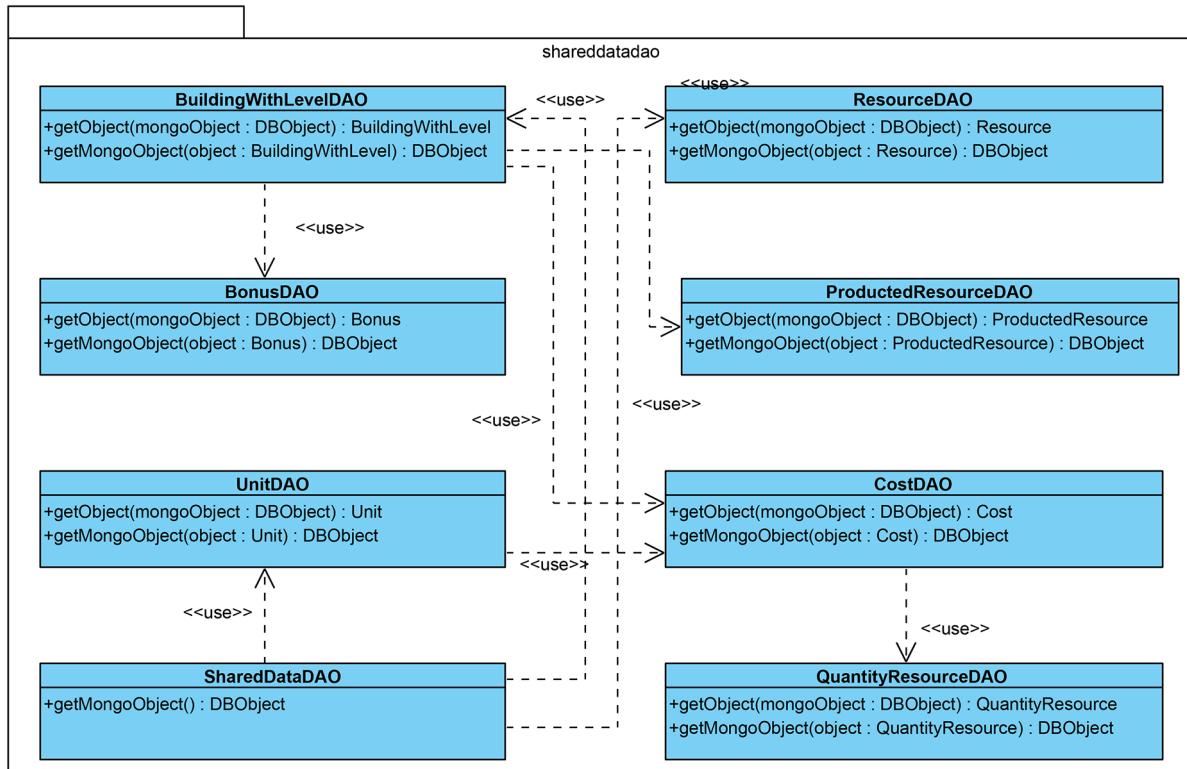


Figura 218: Diagramma della componente `sgad::servertier::dataaccess::-databaseaccess::shareddataadao`

- **Descrizione:** componente per l'istanziazione delle classi del pacchetto shareddata a partire dai dati letti dal database.
- **Padre:** `databaseaccess`
- **Interazioni con altri componenti**
 - `casbah`.

- **shareddata**: componente per la gestione dei dati che non dipendono dall’utente ma dall’implementazione del mondo di gioco.
- **databasemanager**: componente per la gestione dell’interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l’unico punto di accesso con il database.

3.40.2 Classi

3.40.2.1 sgad::servertier::dataaccess::databaseaccess::shareddatadao::BonusDAO

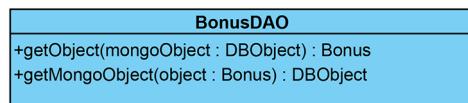


Figura 219: Diagramma della classe sgad::servertier::dataaccess::databaseaccess::shareddatadao::BonusDAO

- **Descrizione:** classe per l’interazione con il database di una classe di tipo Bonus.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe Bonus a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

- **+getObject(mongoObject : DBObject) : Bonus**
Ritorna il corrispettivo Bonus istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* **mongoObject** : l’oggetto da cui parte l’istanziazione della classe ritornata.

- **+getMongoObject(object : Bonus) : DBObject**
Fornisce il corrispettivo MongoDBObject della classe Bonus fornita in input.

Argomenti:

* **object** : l’oggetto da cui parte l’istanziazione della classe ritornata.

3.40.2.2 sgad::servertier::dataaccess::databaseaccess::shareddatadao::BuildingWithLevelDAO

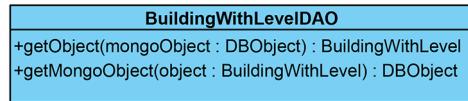


Figura 220: Diagramma della classe sgad::servertier::dataaccess::databaseaccess::shareddatadao::BuildingWithLevelDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo BuildingWithLevel.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe BuildingWithLevel a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : BuildingWithLevel`

Ritorna il corrispettivo BuildingWithLevel istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

– `+getMongoObject(object : BuildingWithLevel) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe BuildingWithLevel fornita in input.

Argomenti:

* `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.40.2.3 sgad::servertier::dataaccess::databaseaccess::shareddatadao::CostDAO

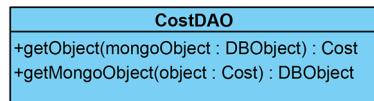


Figura 221: Diagramma della classe sgad::servertier::dataaccess::databaseaccess::shareddatadao::CostDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo Cost.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe Cost a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.

- Metodi

- `+getObject(mongoObject : DBObject) : Cost`

Ritorna il corrispettivo Cost istanziato a partire da un oggetto MongoDBObject.

Argomenti:

- * `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

- `+getMongoObject(object : Cost) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe Cost fornita in input.

Argomenti:

- * `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.40.2.4 sgad::servertier::dataaccess::databaseaccess::shareddatadao::ProductedResourceDAO

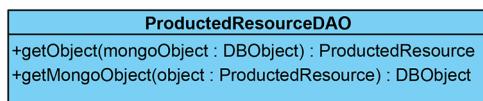


Figura 222: Diagramma della classe `sgad::servertier::dataaccess::databaseaccess::shareddatadao::ProductedResourceDAO`

- **Descrizione:** classe per l'interazione con il database di una classe di tipo ProductedResource.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe ProductedResource a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.

- Metodi

- `+getObject(mongoObject : DBObject) : ProductedResource`

Ritorna il corrispettivo ProductedResource istanziato a partire da un oggetto MongoDBObject.

Argomenti:

- * `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

- `+getMongoObject(object : ProductedResource) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe ProductedResource fornita in input.

Argomenti:

* **object** : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.40.2.5 sgad::servertier::dataaccess::databaseaccess::shareddatadao::Quantity-ResourceDAO

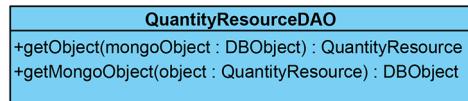


Figura 223: Diagramma della classe sgad::servertier::dataaccess::-databaseaccess::shareddatadao::QuantityResourceDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo Quantity-Resource.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe QuantityResource a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– **+getObject(mongoObject : DBObject) : QuantityResource**
 Ritorna il corrispettivo QuantityResource istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* **mongoObject** : l'oggetto da cui parte l'istanziazione della classe ritornata.

– **+getMongoObject(object : QuantityResource) : DBObject**
 Fornisce il corrispettivo MongoDBObject della classe QuantityResource fornita in input.

Argomenti:

* **object** : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.40.2.6 sgad::servertier::dataaccess::databaseaccess::shareddatadao::Resource-DAO

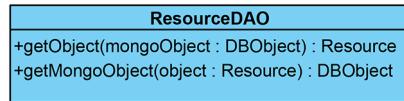


Figura 224: Diagramma della classe `sgad::servertier::dataaccess::databaseaccess::shareddatadao::ResourceDAO`

- **Descrizione:** classe per l'interazione con il database di una classe di tipo Resource.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe Resource a partire da oggetti `MongoDBObject` propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : Resource`

Ritorna il corrispettivo Resource istanziato a partire da un oggetto `MongoDBObject`.

Argomenti:

* `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

– `+getMongoObject(object : Resource) : DBObject`

Fornisce il corrispettivo `MongoDBObject` della classe Resource fornita in input.

Argomenti:

* `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.40.2.7 `sgad::servertier::dataaccess::databaseaccess::shareddatadao::SharedDataDAO`

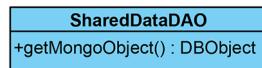


Figura 225: Diagramma della classe `sgad::servertier::dataaccess::databaseaccess::shareddatadao::SharedDataDAO`

- **Descrizione:** classe per la trasformazione dei dati di gioco generali contenuti nel DataFactory in formato JSON.
- **Utilizzo:** viene usata per ottenere il JSON dei dati di gioco generali.
- **Metodi**

– `+getMongoObject() : DBObject`

Fornisce il corrispettivo DBObject dei dati di gioco generali contenuti nel DataFactory.

3.40.2.8 sgad::servertier::dataaccess::databaseaccess::shareddatadao::UnitDAO

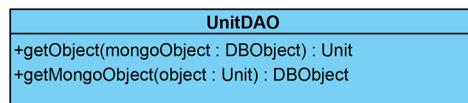


Figura 226: Diagramma della classe `sgad::servertier::dataaccess::databaseaccess::shareddatadao::UnitDAO`

- **Descrizione:** classe per l'interazione con il database di una classe di tipo Unit.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe Unit a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : Unit`

Ritorna il corrispettivo Unit istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

– `+getMongoObject(object : Unit) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe Unit fornita in input.

Argomenti:

* `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41 Componente sgad::servertier::dataaccess::databaseaccess::userdatadao

3.41.1 Informazioni sul package

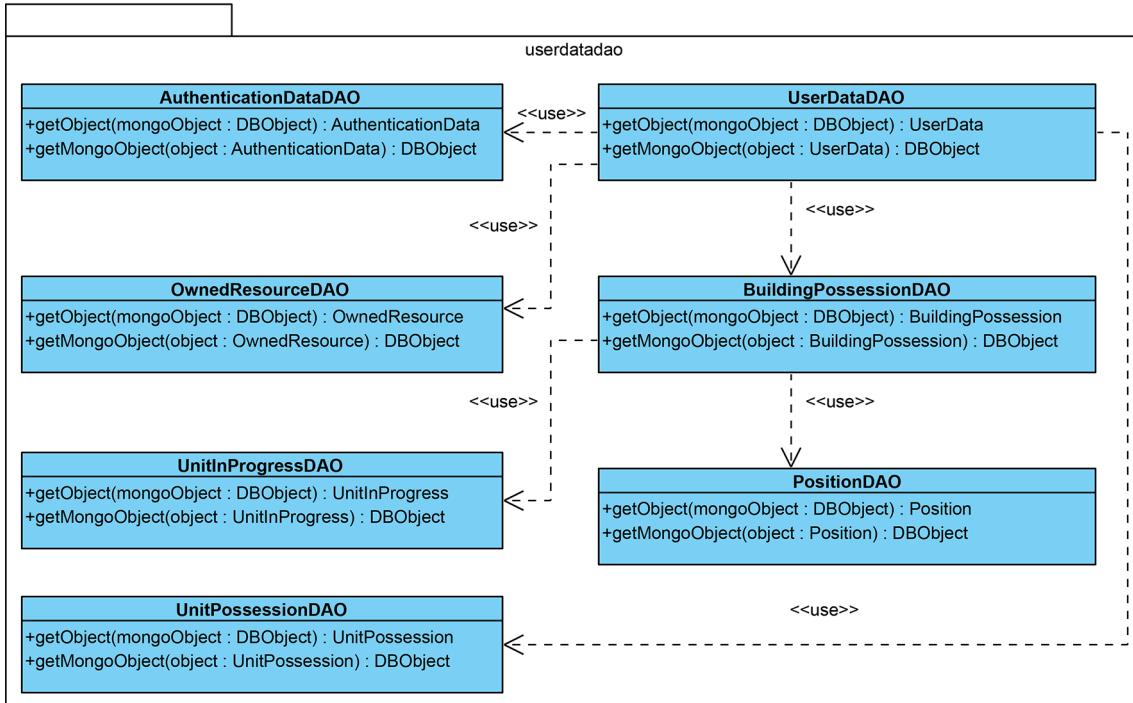


Figura 227: Diagramma della componente `sgad::servertier::dataaccess::databaseaccess::userdatadao`

- **Descrizione:** componente per l'istanziazione delle classi del pacchetto userdata a partire dai dati letti dal database.
- **Padre:** `databaseaccess`
- **Interazioni con altri componenti**
 - `casbah`.
 - `shareddata`: componente per la gestione dei dati che non dipendono dall'utente ma dall'implementazione del mondo di gioco.
 - `userdata`: componente per la gestione dei dati relativi alla sessione dell'utente mantenuta dal server.
 - `databasemanager`: componente per la gestione dell'interazione tra il server e il database, esegue azioni come il salvataggio e il caricamento dei dati. Esso è l'unico punto di accesso con il database.

3.41.2 Classi

3.41.2.1 sgad::servtier::dataaccess::databaseaccess::userdatadao::AuthenticationDataDAO



Figura 228: Diagramma della classe sgad::servtier::dataaccess::databaseaccess::userdatadao::AuthenticationDataDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo AuthenticationData.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe AuthenticationData a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : AuthenticationData`

Ritorna il corrispettivo AuthenticationData istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

– `+getMongoObject(object : AuthenticationData) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe AuthenticationData fornita in input.

Argomenti:

* `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41.2.2 sgad::servtier::dataaccess::databaseaccess::userdatadao::BuildingPossessionDAO

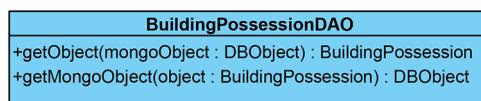


Figura 229: Diagramma della classe sgad::servtier::dataaccess::databaseaccess::userdatadao::BuildingPossessionDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo BuildingPossession.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe BuildingPossession a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : BuildingPossession`

Ritorna il corrispettivo BuildingPossession istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

– `+getMongoObject(object : BuildingPossession) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe BuildingPossession fornita in input.

Argomenti:

* `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41.2.3 sgad::servtier::dataaccess::databaseaccess::userdatadao::OwnedResourceDAO



Figura 230: Diagramma della classe `sgad::servtier::dataaccess::databaseaccess::userdatadao::OwnedResourceDAO`

- **Descrizione:** classe per l'interazione con il database di una classe di tipo OwnedResource.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe OwnedResource a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : OwnedResource`

Ritorna il corrispettivo OwnedResource istanziato a partire da un oggetto MongoDBObject.

Argomenti:

- * `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.
- `+getMongoObject(object : OwnedResource) : DBObject`
Fornisce il corrispettivo MongoDBObject della classe OwnedResource fornita in input.

Argomenti:

- * `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41.2.4 sgad::servertier::dataaccess::databaseaccess::userdatadao::PositionDAO



Figura 231: Diagramma della classe `sgad::servertier::dataaccess::databaseaccess::userdatadao::PositionDAO`

- **Descrizione:** classe per l'interazione con il database di una classe di tipo Position.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe Position a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

- `+getObject(mongoObject : DBObject) : Position`
Ritorna il corrispettivo Position istanziato a partire da un oggetto MongoDBObject.

Argomenti:

- * `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

- `+getMongoObject(object : Position) : DBObject`
Fornisce il corrispettivo MongoDBObject della classe Position fornita in input.

Argomenti:

- * `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41.2.5 sgad::servertier::dataaccess::databaseaccess::userdatadao::UnitInProgressDAO



Figura 232: Diagramma della classe sgad::servertier::dataaccess::databaseaccess::userdatadao::UnitInProgressDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo UnitInProgress.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe UnitInProgress a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.
- **Metodi**

– `+getObject(mongoObject : DBObject) : UnitInProgress`

Ritorna il corrispettivo UnitInProgress istanziato a partire da un oggetto MongoDBObject.

Argomenti:

* `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

– `+getMongoObject(object : UnitInProgress) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe UnitInProgress fornita in input.

Argomenti:

* `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41.2.6 sgad::servertier::dataaccess::databaseaccess::userdatadao::UnitPossessionDAO

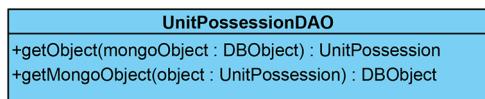


Figura 233: Diagramma della classe sgad::servertier::dataaccess::databaseaccess::userdatadao::UnitPossessionDAO

- **Descrizione:** classe per l'interazione con il database di una classe di tipo UnitPossession.
- **Utilizzo:** viene utilizzata per istanziare oggetti della classe UnitPossession a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.

- Metodi

- `+getObject(mongoObject : DBObject) : UnitPossession`

Ritorna il corrispettivo UnitPossession istanziato a partire da un oggetto MongoDBObject.

Argomenti:

- * `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

- `+getMongoObject(object : UnitPossession) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe UnitPossession fornita in input.

Argomenti:

- * `object` : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.41.2.7 sgad::servtier::dataaccess::databaseaccess::userdatadao::UserDataDAO

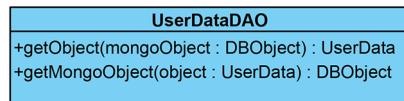


Figura 234: Diagramma della classe `sgad::servtier::dataaccess::databaseaccess::userdatadao::UserDataDAO`

- **Descrizione:** classe per l'interazione con il database di una classe di tipo UserData.

- **Utilizzo:** viene utilizzata per istanziare oggetti della classe UserData a partire da oggetti MongoDBObject propri della tecnologia di persistenza adottata.

- Metodi

- `+getObject(mongoObject : DBObject) : UserData`

Ritorna il corrispettivo UserData istanziato a partire da un oggetto MongoDBObject.

Argomenti:

- * `mongoObject` : l'oggetto da cui parte l'istanziazione della classe ritornata.

- `+getMongoObject(object : UserData) : DBObject`

Fornisce il corrispettivo MongoDBObject della classe UserData fornita in input.

Argomenti:

* **object** : l'oggetto da cui parte l'istanziazione della classe ritornata.

3.42 Componente sgad::servertier::presentation

3.42.1 Informazioni sul package

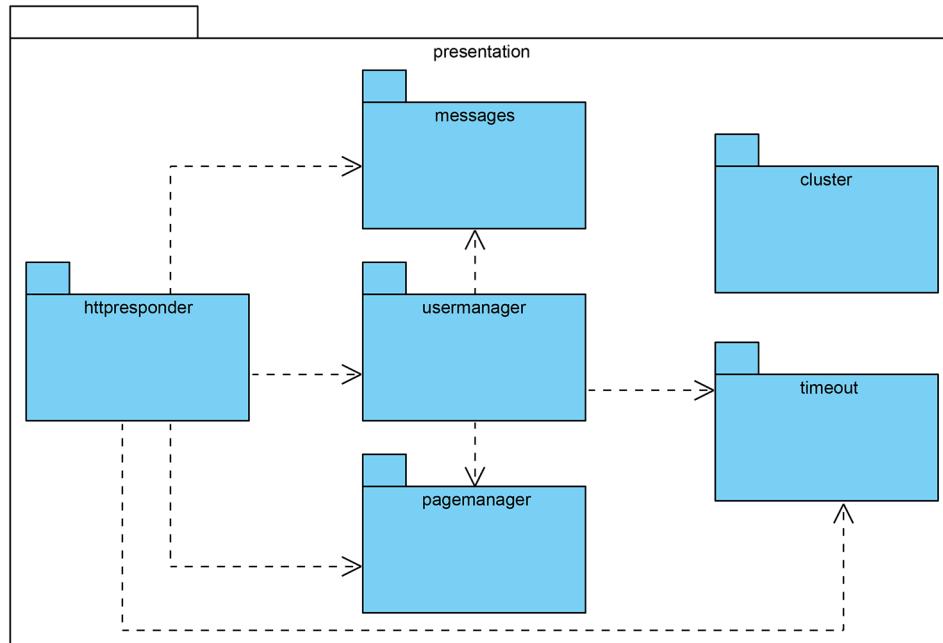


Figura 235: Diagramma della componente sgad::servertier::presentation

- **Descrizione:** componente per la gestione della comunicazione con i client. Rappresenta il livello superiore e visibile dall'esterno dell'architettura Three-Tier. Si occupa di smistare le richieste in arrivo alle corrette componenti della logica.
- **Padre:** servertier
- **Interazioni con altri componenti**
 - akka::actors.
 - akka::contrib::pattern.
 - application: componente che si occupa del bootstrap dell'applicazione.
 - businesslogic: componente corrispondente al livello business logic dell'architettura Three-Tier. Essa si occupa di effettuare le operazioni richieste dagli utenti. Nel caso le operazioni siano relative a interazioni tra più utenti, comunicherà con la componente presentation per instradare correttamente la richiesta verso gli altri client coinvolti.
 - spray.
- **Package contenuti**
 - cluster: componente per la gestione e la partecipazione del singolo server a cluster di server.

- **httpresponder**: componente per la gestione delle richieste HTTP ricevute dai vari client.
- **messages**: componente per i messaggi scambiati tra gli attori del sistema.
- **pagemanager**: componente per la gestione delle pagine $HTML_{|g|}$, $CSS_{|g|}$ e sorgenti $JavaScript_{|g|}$.
- **timeout**: componente per la gestione dei timeout.
- **userManager**: componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore associato alla richiesta.

3.43 Componente sgad::servertier::presentation::cluster

3.43.1 Informazioni sul package

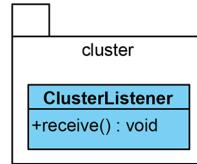


Figura 236: Diagramma della componente `sgad::servertier::presentation::cluster`

- **Descrizione**: componente per la gestione e la partecipazione del singolo server a cluster di server.
- **Padre**: `presentation`
- **Interazioni con altri componenti**
 - `application`: componente che si occupa del bootstrap dell'applicazione.

3.43.2 Classi

3.43.2.1 sgad::servertier::presentation::cluster::ClusterListener



Figura 237: Diagramma della classe `sgad::servertier::presentation::cluster::ClusterListener`

- **Descrizione**: classe che gestisce l'interazione del server con gli altri server del cluster.
- **Utilizzo**: viene utilizzato per gestire i messaggi in arrivo dagli eventi del cluster.

- Metodi
 - `+receive() : void`
Metodo per il parsing dei messaggi in arrivo.

3.44 Componente sgad::servertier::presentation::httpresponder

3.44.1 Informazioni sul package

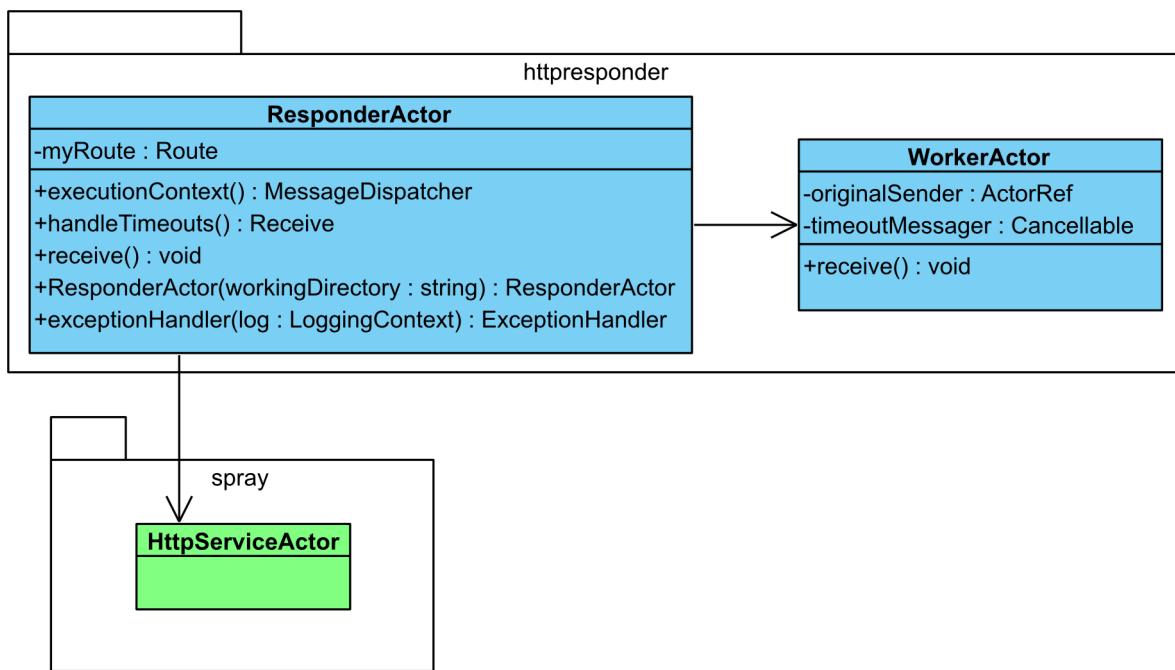


Figura 238: Diagramma della componente `sgad::servertier::presentation::httpresponder`

- **Descrizione:** componente per la gestione delle richieste HTTP ricevute dai vari client.
- **Padre:** `presentation`
- **Interazioni con altri componenti**

- `requester`: componente per la gestione dell'invio di richieste al server.
- `messages`: componente per i messaggi scambiati tra gli attori del sistema.
- `pagemanager`: componente per la gestione delle pagine HTML, CSS e sorgenti JavaScript.
- `timeout`: componente per la gestione dei timeout.

- **usermanager**: componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore associato alla richiesta.
- **spray**.

3.44.2 Classi

3.44.2.1 sgad::servertier::presentation::httpresponder::ResponderActor

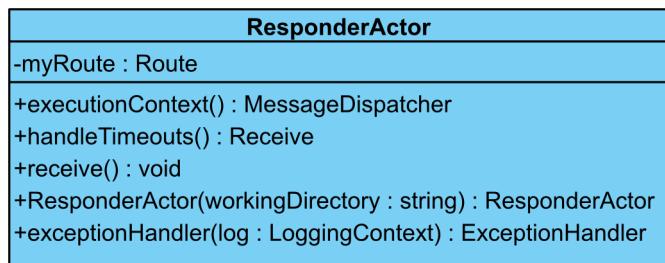


Figura 239: Diagramma della classe sgad::servertier::presentation::httpresponder::ResponderActor

- **Descrizione:** classe per la gestione degli *end point*_[g] per le richieste HTTP inviate dal client.

- **Utilizzo:** viene utilizzato per poter esporre un pin di connessione ai client.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione, Gestione di una richiesta di login e Gestione di una richiesta di gioco.

- **Classi ereditate**

- spray::HttpServiceActor

- **Attributi**

- -myRoute : Route

Definisce gli end point HTTP e il modo in cui viene gestita ogni richiesta.

- **Metodi**

- +receive() : void

Risponde alle richieste HTTP con l'utilizzo dell'attributo myRoute.

- +ResponderActor(workingDirectory : string) : ResponderActor

Costruttore della classe.

Argomenti:

* **workingDirectory** : percorso assoluto da cui l'applicazione è stata lanciata.

- `+executionContext() : MessageDispatcher`
Il dispatcher dell'actor system che poter creare a gestire i Timeout delle richieste.
- `+handleTimeouts() : Receive`
Gestisce gli errori del server durante l'elaborazione di una risposta ad una richiesta HTTP.
- `+exceptionHandler(log : LoggingContext) : ExceptionHandler`
Metodo per la gestione degli errori durante il routing.

Argomenti:

- * `log` : riferimento implicito al log.

3.44.2.2 sgad::servertier::presentation::httpresponder::WorkerActor

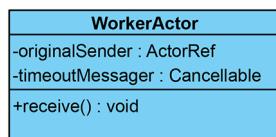


Figura 240: Diagramma della classe `sgad::servertier::presentation::httpresponder::WorkerActor`

- **Descrizione:** classe per la gestione asincrona delle richieste in arrivo dai client.
- **Utilizzo:** viene utilizzata dal ResponderActor per rendere asincrone le risposte ai client.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione, Gestione di una richiesta di login e Gestione di una richiesta di gioco.

- **Attributi**

- `-originalSender : ActorRef`
Sender iniziale che ha richiesto il servizio a cui si risponderà ottenuta la risposta.
- `-timeoutMessager : Cancellable`
Timeout che rappresenta il tempo massimo di una richiesta, dopo di che il worker va stoppato.

- **Metodi**

- `+receive() : void`
Gestisce le richieste di login, registrazione e di gioco e chiede ad altri attori la risposta. Ottiene la risposta e la manda all'attore che ha chiesto il servizio.

3.45 Componente sgad::servertier::presentation::messages

3.45.1 Informazioni sul package

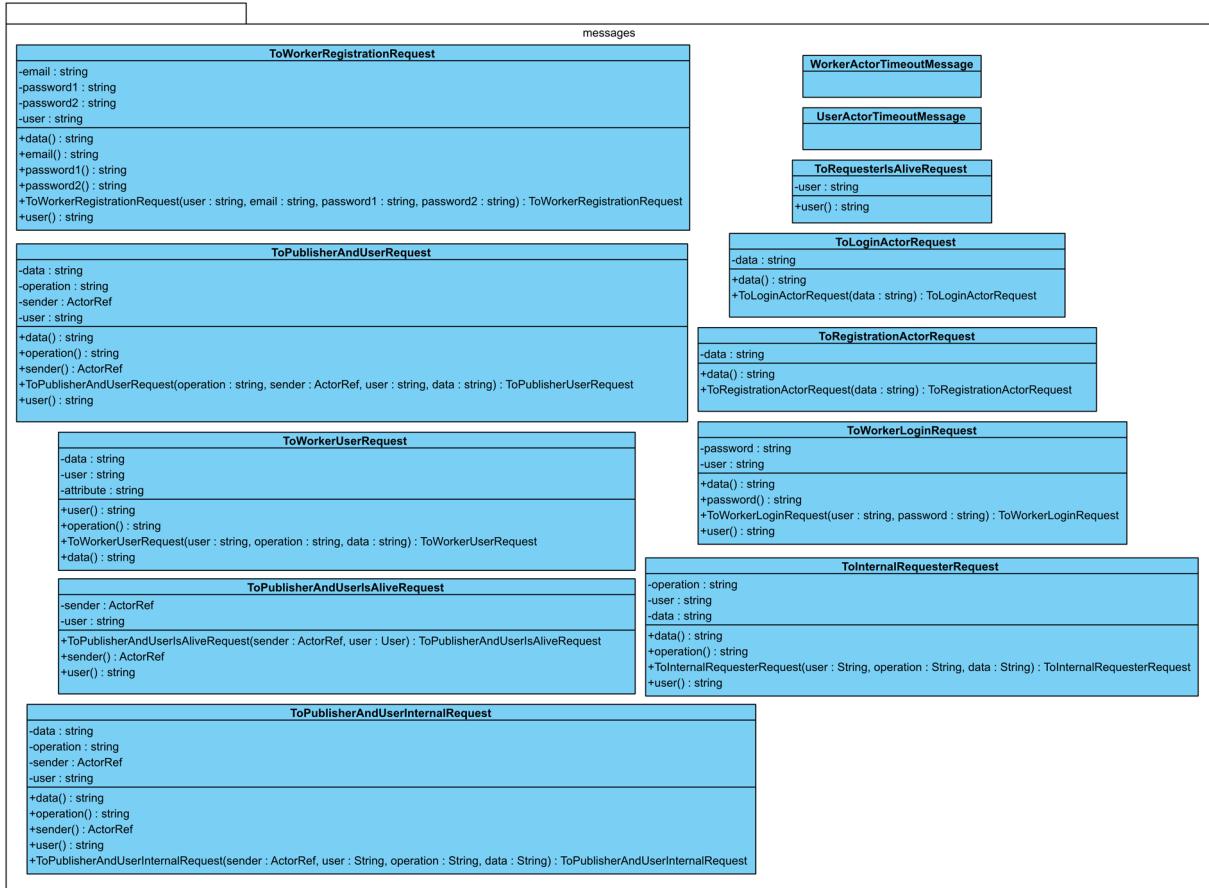


Figura 241: Diagramma della componente `sgad::servertier::presentation::messages`

- **Descrizione:** componente per i messaggi scambiati tra gli attori del sistema.
- **Padre:** `presentation`
- **Interazioni con altri componenti**
 - `httpresponder`: componente per la gestione delle richieste HTTP ricevute dai vari client.
 - `usermanager`: componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore associato alla richiesta.

3.45.2 Classi

3.45.2.1 sgad::servertier::presentation::messages::ToInternalRequesterRequest

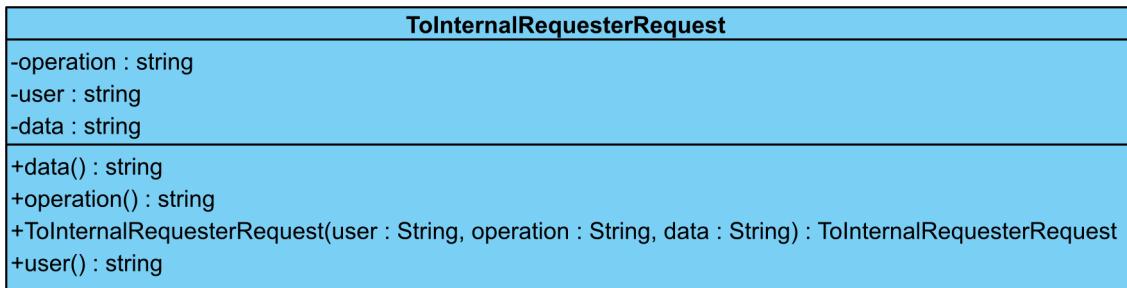


Figura 242: Diagramma della classe `sgad::servertier::presentation::messages::ToInternalRequesterRequest`

- **Descrizione:** classe per una richiesta interna verso il InternalRequester.
- **Utilizzo:** viene utilizzata per inviare all'internalRequester una richiesta di gestione di un messaggio interno.
- **Attributi**

- `-data : string`
Dati della richiesta.
- `-operation : string`
Operazione richiesta.
- `-user : string`
User destinatario della richiesta.

- **Metodi**

- `+ToInternalRequesterRequest(data : String, operation : String, user : String) : ToInternalRequesterRequest`
Costruttore della classe.

Argomenti:

- * `data` : dati della richiesta.
- * `operation` : operazione richiesta.
- * `user` : user destinatario della richiesta.

– **+user()** : **string**
 Ritorna il valore dell'attributo user.

– **+data()** : **string**
 Ritorna il valore dell'attributo data.

– **+operation()** : **string**
 Ritorna il valore dell'attributo operation.

3.45.2.2 sgad::servtier::presentation::messages::ToLoginActorRequest

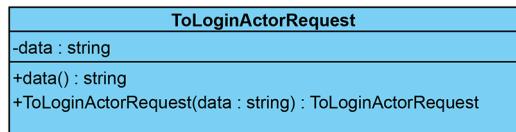


Figura 243: Diagramma della classe `sgad::servtier::presentation::messages::ToLoginActorRequest`

- **Descrizione:** classe per il messaggio di elaborazione di una richiesta di login al LoginActor.

- **Utilizzo:** viene utilizzata dal WorkerActor per ottenere una risposta alla richiesta di login ricevuta.

- **Attributi**

– **-data** : **string**
 Dati di accompagnamento alla richiesta.

- **Metodi**

– **+ToLoginActorRequest(data : string)** : **ToLoginActorRequest**
 Costruttore della classe.

Argomenti:

* **data** : dati di accompagnamento alla richiesta.

– **+data()** : **string**
 Ritorna il valore dell'attributo data.

3.45.2.3 sgad::servertier::presentation::messages::ToPublisherAndUserInternalRequest

ToPublisherAndUserInternalRequest
<pre>-data : string -operation : string -sender : ActorRef -user : string +data() : string +operation() : string +sender() : ActorRef +user() : string +ToPublisherAndUserInternalRequest(sender : ActorRef, user : String, operation : String, data : String) : ToPublisherAndUserInternalRequest</pre>

Figura 244: Diagramma della classe sgad::servertier::presentation::messages::-ToPublisherAndUserInternalRequest

- **Descrizione:** classe per la gestione delle richieste interne tra user.
- **Utilizzo:** viene utilizzata per mandare un messaggio di richiesta interna tra UserActor.
- **Attributi**

- `-data : string`
Dati associati alla richiesta.
- `-operation : string`
Operazione richiesta.
- `-sender : ActorRef`
Sender della richiesta a cui inviare la risposta.
- `-user : string`
User destinatario delle richieste.

- **Metodi**

- `+ToPublisherAndUserInternalRequest(data : String, operation : String, sender : ActorRef, user : String) : ToPublisherAndUserInternalRequest`
Costruttore della classe.

Argomenti:

- * `data` : dati associati alla richiesta.
- * `operation` : operazione richiesta.
- * `sender` : sender della richiesta a cui inviare la risposta.

- * `user` : user destinatario delle richiesta.
- `+user() : string`
Ritorna il valore dell'attributo user.
- `+data() : string`
Ritorna il valore dell'attributo data.
- `+operation() : string`
Ritorna il valore dell'attributo operation.
- `+sender() : ActorRef`
Ritorna il valore dell'attributo sender.

3.45.2.4 sgad::servertier::presentation::messages::ToPublisherAndUserIsAliveRequest

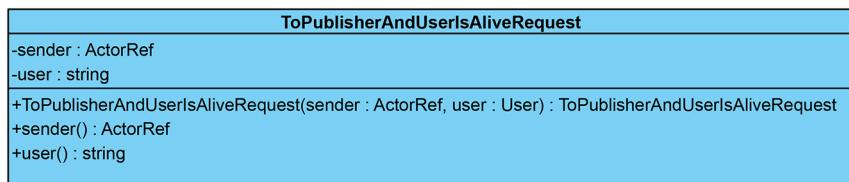


Figura 245: Diagramma della classe `sgad::servertier::presentation::messages::-ToPublisherAndUserIsAliveRequest`

- **Descrizione:** classe per il messaggio isAlive?.
- **Utilizzo:** viene utilizzata per scoprire se un attore è attivo.
- **Attributi**

- `-sender : ActorRef`
Sender della richiesta a cui inviare la risposta.
- `-user : string`
User destinatario delle richiesta.

- **Metodi**

- `+IsAliveRequest(sender : ActorRef, user : User) : IsAliveRequest`
Costruttore della classe

Argomenti:

- * **sender** : sender della richiesta a cui inviare la risposta.
- * **user** : destinatario delle richiesta.
- **+sender()** : **ActorRef**
Ritorna il valore dell'attributo sender.
- **+user()** : **string**
Ritorna il valore dell'attributo user.

3.45.2.5 sgad::servertier::presentation::messages::ToPublisherAndUserRequest

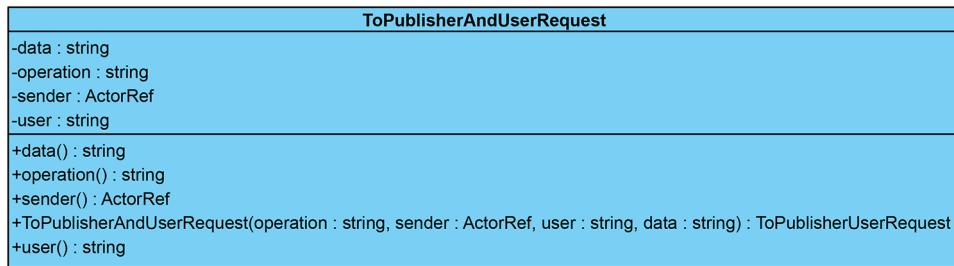


Figura 246: Diagramma della classe `sgad::servertier::presentation::messages::-ToPublisherAndUserRequest`

- **Descrizione:** classe per il messaggio di elaborazione di una richiesta di gioco al PublisherActor.
- **Utilizzo:** viene utilizzata dal WorkerActor per ottenere una risposta alla richiesta di gioco ricevuta.

- **Attributi**

- **-data** : **string**
Dati di accompagnamento alla richiesta.
- **-operation** : **string**
Operazione richiesta.
- **-sender** : **ActorRef**
Sender della richiesta a cui inviare la risposta.
- **-user** : **string**
User destinatario delle richiesta.

- **Metodi**

– `+ToPublisherAndUserRequest(data : string, operation : string, sender : ActorRef, user : string) : ToPublisherAndUserRequest`
 Costruttore della classe.

Argomenti:

- * `data` : dati di accompagnamento alla richiesta.
- * `operation` : operazione richiesta.
- * `sender` : sender della richiesta a cui inviare la risposta.
- * `user` : user destinatario delle richieste.

– `+sender() : ActorRef`
 Ritorna il valore dell'attributo sender.

– `+user() : string`
 Ritorna il valore dell'attributo user.

– `+operation() : string`
 Ritorna il valore dell'attributo operation.

– `+data() : string`
 Ritorna il valore dell'attributo data.

3.45.2.6 sgad::servertier::presentation::messages::ToRegistrationActorRequest

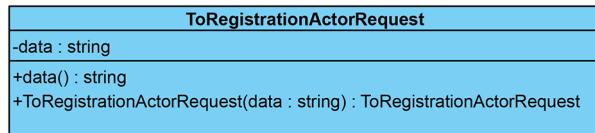


Figura 247: Diagramma della classe `sgad::servertier::presentation::messages::ToRegistrationActorRequest`

- **Descrizione:** classe per il messaggio di elaborazione di una richiesta di registrazione al RegistrationActor.
- **Utilizzo:** viene utilizzata dal WorkerActor per ottenere una risposta alla richiesta di registrazione ricevuta.
- **Attributi**

- `-data : string`
Dati di accompagnamento alla richiesta.

- **Metodi**

- `+ToRegistrationActorRequest (data : string) : ToRegistrationActorRequest`
Costruttore della classe.

Argomenti:

- * `data` : dati di accompagnamento alla richiesta.
- `+data() : string`
Ritorna il valore dell'attributo data.

3.45.2.7 sgad::servertier::presentation::messages::ToRequesterIsAliveRequest



Figura 248: Diagramma della classe `sgad::servertier::presentation::messages::ToRequesterIsAliveRequest`

- **Descrizione:** classe per la richiesta di esistenza dell'UserActor associato al user in input.
- **Utilizzo:** la classe viene utilizzata per richiedere l'esistenza dell'UserActor associato ad un utente.
- **Attributi**

- `-user : string`
Username destinatario delle richieste.

- **Metodi**

- `+user() : string`
Ritorna il valore per l'attributo user.

3.45.2.8 sgad::servertier::presentation::messages::ToWorkerLoginRequest

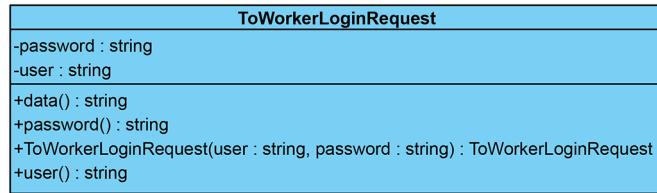


Figura 249: Diagramma della classe `sgad::servertier::presentation::messages::ToWorkerLoginRequest`

- **Descrizione:** classe per il messaggio di elaborazione di una richiesta di login al WorkerActor.

- **Utilizzo:** viene utilizzata ogni qualvolta il server ottiene dal client una richiesta di login.

- **Attributi**

- `-password : string`
Dati di accompagnamento alla richiesta.

- `-user : string`
Username dell’utente.

- **Metodi**

- `+ToWorkerLoginRequest (password : string, user : string) : ToWorkerLoginRequest`
Costruttore della classe.

Argomenti:

- * `password` : password dell’utente.

- * `user` : username dell’utente.

- `+password() : string`
Ritorna il valore dell’attributo password.

- `+user() : string`
Ritorna il valore dell’attributo user.

- `+data() : string`
Converte la classe nel formato di dati compatibili per le operations.

3.45.2.9 sgad::servertier::presentation::messages::ToWorkerRegistrationRequest

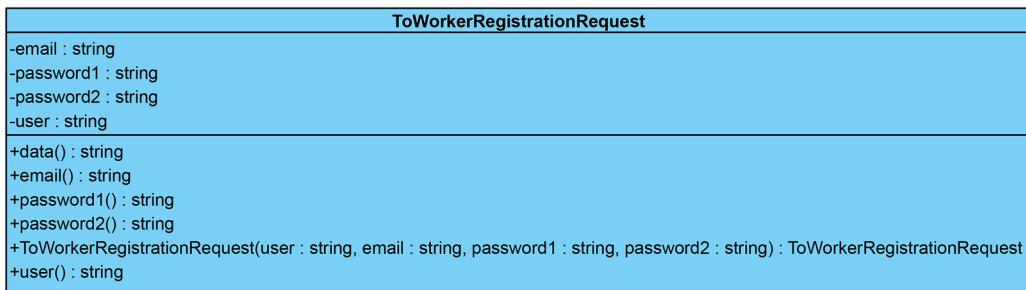


Figura 250: Diagramma della classe `sgad::servertier::presentation::messages::ToWorkerRegistrationRequest`

- **Descrizione:** classe per il messaggio di elaborazione di una richiesta di registrazione al WorkerActor.
- **Utilizzo:** viene utilizzata ogni qualvolta il server ottiene dal client una richiesta di registrazione.
- **Attributi**

- `-email : string`
Email dell’utente.
- `-password1 : string`
Password dell’utente.
- `-password2 : string`
Password dell’utente ripetuta per doppia conferma.
- `-user : string`
Username dell’utente.

- **Metodi**

- `+email() : string`
Ritorna il valore dell’attributo email.
- `+user() : string`
Ritorna il valore dell’attributo user.
- `+password1() : string`
Ritorna il valore dell’attributo password.
- `+data() : string`
Converte la classe nel formato di dati compatibili per le operations.

- `+password2() : string`
Ritorna il valore dell'attributo password2.

- `+ToWorkerRegistrationRequest(email : string, password1 : string, password2 : string, user : string) : ToWorkerRegistrationRequest`
Costruttore della classe.

Argomenti:

- * `email` : email dell'utente.
- * `password1` : password dell'utente.
- * `password2` : password dell'utente ripetuta per doppia conferma.
- * `user` : username dell'utente.

3.45.2.10 sgad::servtier::presentation::messages::ToWorkerUserRequest

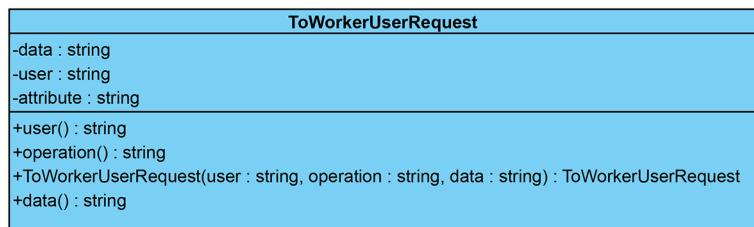


Figura 251: Diagramma della classe `sgad::servtier::presentation::messages::-ToWorkerUserRequest`

- **Descrizione:** classe per il messaggio di elaborazione di una richiesta di gioco al WorkerActor.
- **Utilizzo:** viene utilizzata ogni qualvolta il server ottiene dal client una richiesta di gioco.
- **Attributi**

- `-data : string`
Dati di accompagnamento alla richiesta.
- `-operation : string`
Operazione richiesta.
- `-user : string`
User destinatario delle richieste.

- Metodi

- `+ToWorkerUserRequest (data : string, operation : string, user : string)`
`: ToWorkerUserRequest`
 Costruttore della classe.
- Argomenti:**
- * `data` : dati di accompagnamento alla richiesta.
 - * `operation` : operazione richiesta.
 - * `user` : user destinatario della richiesta.
- `+data() : string`
 Ritorna il valore dell'attributo data.
 - `+operation() : string`
 Ritorna il valore dell'attributo operation.
 - `+user() : string`
 Ritorna il valore dell'attributo user.

3.45.2.11 sgad::servertier::presentation::messages::UserActorTimeoutMessage

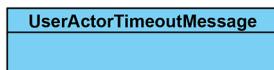


Figura 252: Diagramma della classe `sgad::servertier::presentation::messages::UserActorTimeoutMessage`

- **Descrizione:** classe che rappresenta il messaggio di timeout verso l'UserActor a cui l'attore deve reagire uccidendosi.
- **Utilizzo:** viene utilizzata per rappresentare il messaggio di timeout diretto all'UserActor a cui l'attore deve reagire.

3.45.2.12 sgad::servertier::presentation::messages::WorkerActorTimeoutMessage



Figura 253: Diagramma della classe `sgad::servertier::presentation::messages::-WorkerActorTimeoutMessage`

- **Descrizione:** classe che rappresenta il messaggio di timeout verso il WorkerActor a cui l'attore deve reagire uccidendosi.
- **Utilizzo:** viene utilizzata per rappresentare il messaggio di timeout diretto al WorkerActor a cui l'attore deve reagire.

3.46 Componente `sgad::servertier::presentation::pagemanager`

3.46.1 Informazioni sul package

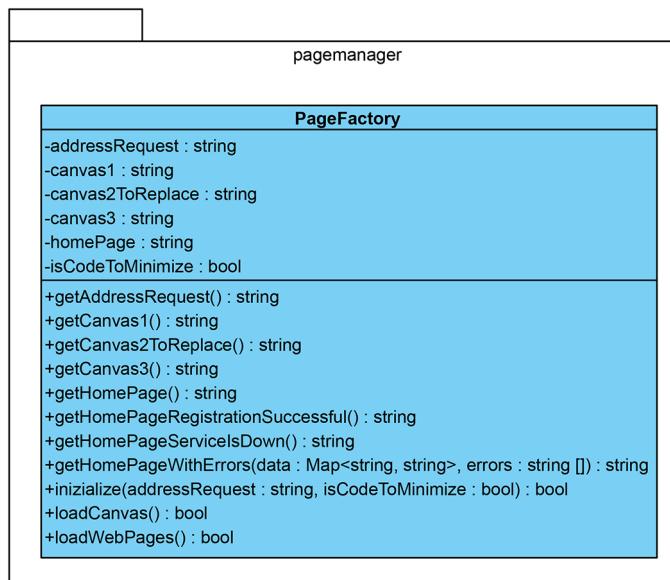


Figura 254: Diagramma della componente `sgad::servertier::presentation::-pagemanager`

- **Descrizione:** componente per la gestione delle pagine HTML, CSS e sorgenti JavaScript.
- **Padre:** `presentation`
- **Interazioni con altri componenti**
 - `httpresponder`: componente per la gestione delle richieste HTTP ricevute dai vari client.

- **userManager**: componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore associato alla richiesta.

3.46.2 Classi

3.46.2.1 sgad::servtier::presentation::pagemanager::PageFactory

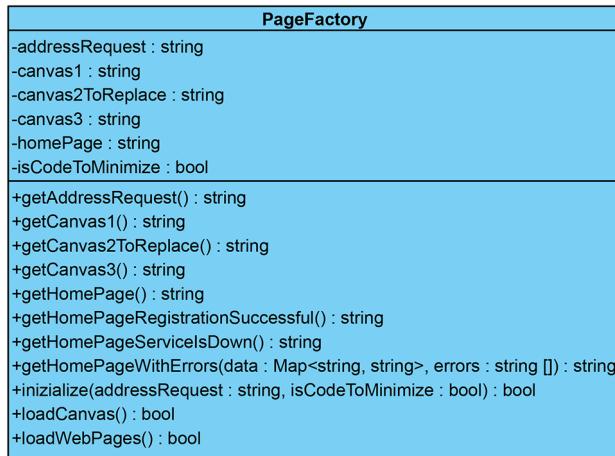


Figura 255: Diagramma della classe sgad::servtier::presentation::pagemanager::PageFactory

- **Descrizione:** classe che gestisce le pagine HTML da inviare ai client.
- **Utilizzo:** viene usata per ottenere le pagine HTML personalizzate e non da inviare ai client.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione e Gestione di una richiesta di login.

- **Attributi**

- **-addressRequest : string**
Indirizzo su cui il server HTTP è reperibile.
- **-canvas1 : string**
La prima parte del codice della pagina di gioco.
- **-canvas2ToReplace : string**
La seconda parte della pagina di gioco, localizza un punto dove iniettare le personalizzazioni nel codice JavaScript lato client.
- **-canvas3 : string**
Terza ed ultima parte della pagina di gioco.
- **-HomePage : string**
Contiene il codice HTML della home page.

– `-isCodeToMinimize : bool`

Decide se il codice di gioco lato client viene minimizzato al caricamento.

- Metodi

– `+getAddressRequest() : string`

Metodo getter per l'attributo addressRequest.

– `+inizialize(addressRequest : string, isCodeToMinimize : bool) : bool`

Esegue l'inizializzazione caricando le risorse esterne come le pagine del sito ed il codice di gioco lato client, eventualmente minimizzandolo.

Argomenti:

* `addressRequest` : l'indirizzo su cui l'applicazione risponde alle richieste HTTP.

* `isCodeToMinimize` : decide se il codice di gioco lato client viene minimizzato al caricamento.

– `-loadCanvas() : bool`

Carica il codice di gioco in memoria ed eventualmente lo minimizza in base all'attributo isCodeToMinimize.

– `+getCanvas1() : string`

Metodo getter per l'attributo canvas1.

– `+getCanvas2ToReplace() : string`

Metodo getter per l'attributo canvas2ToReplace.

– `+getCanvas3() : string`

Metodo getter per l'attributo canvas3.

– `-loadWebPages() : bool`

Carica in memoria tutto il codice delle web pages del sito del gioco.

– `+getHomePage() : string`

Metodo getter che ritorna la homepage del sito.

– `+getHomePageWithErrors(data : Map<string,string>, errors : string[]) : string`

Metodo utilizzato dal pageManager per poter restituire la Homepage con gli errori.

Argomenti:

- * **data** : i dati passati dalla pagina.
- * **errors** : una lista di errori riscontrati.
- **+getHomePageRegistrationSuccessful()** : **string**
Metodo per ricevere la pagina di registrazione avvenuta.
- **+getHomePageServiceIsDown()** : **string**
Metodo per ricevere la pagina ricevuta quando il servizio non è disponibile.

3.47 Componente sgad::servertier::presentation::timeout

3.47.1 Informazioni sul package

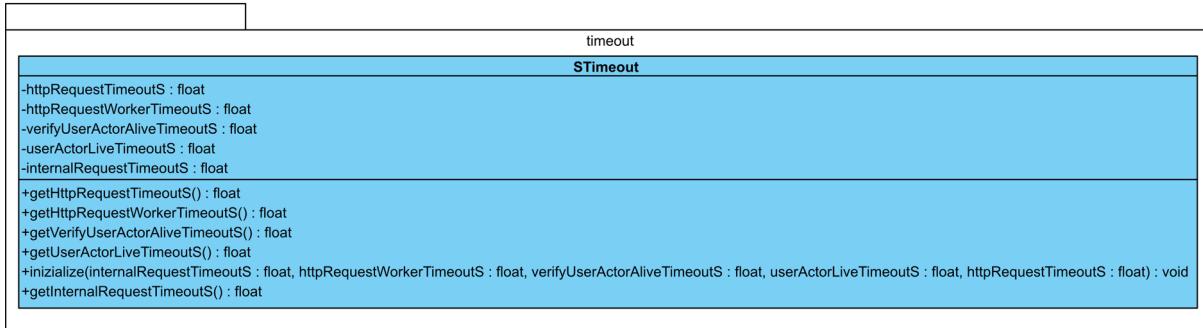


Figura 256: Diagramma della componente sgad::servertier::presentation::timeout

- **Descrizione:** componente per la gestione dei timeout.
- **Padre:** presentation
- **Interazioni con altri componenti**
 - **httpresponder:** componente per la gestione delle richieste HTTP ricevute dai vari client.
 - **usermanager:** componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore associato alla richiesta.

3.47.2 Classi

3.47.2.1 sgad::servertier::presentation::timeout::STimeout

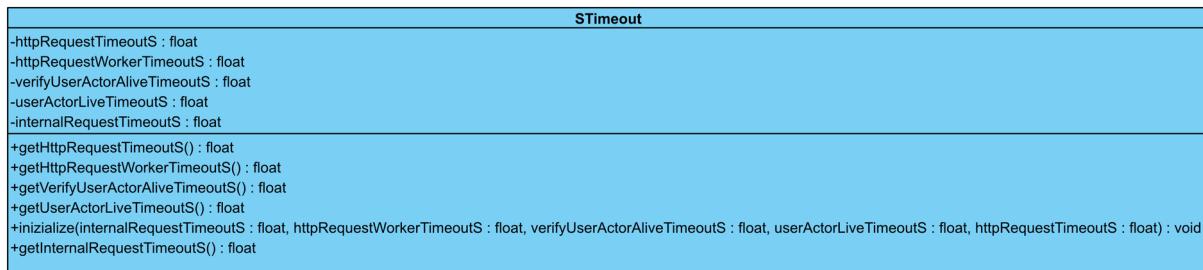


Figura 257: Diagramma della classe sgad::servertier::presentation::timeout::-STimeout

- **Descrizione:** classe che rappresenta i timeout delle richieste all'interno dell'applicazione.

- **Utilizzo:** viene utilizzata per rappresentare i timeout delle richieste.

- **Attributi**

- `-httpRequestTimeoutS : float`
Timeout della richiesta HTTP in secondi.
- `-httpRequestWorkerTimeoutS : float`
Timeout della vita di un worker. Dovrebbe essere più alto di quello della richiesta HTTP in secondi.
- `-internalRequestTimeoutS : float`
Timeout di una richiesta interna fra user.
- `-verifyUserActorAliveTimeoutS : float`
Timeout della ricerca se l'attore di un utente è vivo. Ogni richiesta di login impiega almeno questo tempo in secondi.
- `-userActorLiveTimeoutS : float`
Tempo di vita di una sessione utente senza che l'utente non interagisca con il server in secondi.

- **Metodi**

- `+inizialize(httpRequestTimeoutS : float, httpRequestWorkerTimeoutS : float, internalRequestTimeoutS : float, verifyUserActorAliveTimeoutS : float, userActorLiveTimeoutS : float) : void`
Inizializza i parametri dei timeout con valori diversi da quelli di default.

Argomenti:

- * `httpRequestTimeoutS` : timeout della richiesta HTTP in secondi.
 - * `httpRequestWorkerTimeoutS` : timeout della vita di un worker. Dovrebbe essere più alto di quello della richiesta HTTP in secondi.
 - * `internalRequestTimeoutS` : timeout di una richiesta interna fra user.
 - * `verifyUserActorAliveTimeoutS` : timeout della ricerca se l'attore di un utente è vivo. Ogni richiesta di login impiega almeno questo tempo in secondi.
 - * `userActorLiveTimeoutS` : tempo di vita di una sessione utente senza che l'utente non interagisca con il server in secondi.
-
- `+getHttpRequestTimeoutS() : float`
Metodo getter per l'attributo httpRequestTimeoutS

- `+getHttpRequestWorkerTimeoutS() : float`
Metodo getter per l'attributo httpRequestWorkerTimeoutS.

- `+getVerifyUserActorAliveTimeoutS() : float`
Metodo getter per l'attributo verifyUserActorAliveTimeoutS.

- `+getUserActorLiveTimeoutS() : float`
Metodo getter per l'attributo userActorLiveTimeoutS.

- `+getInternalRequestTimeoutS() : float`
Metodo getter per l'attributo internalRequestTimeoutS.

3.48 Componente sgad::servertier::presentation::usermanager

3.48.1 Informazioni sul package

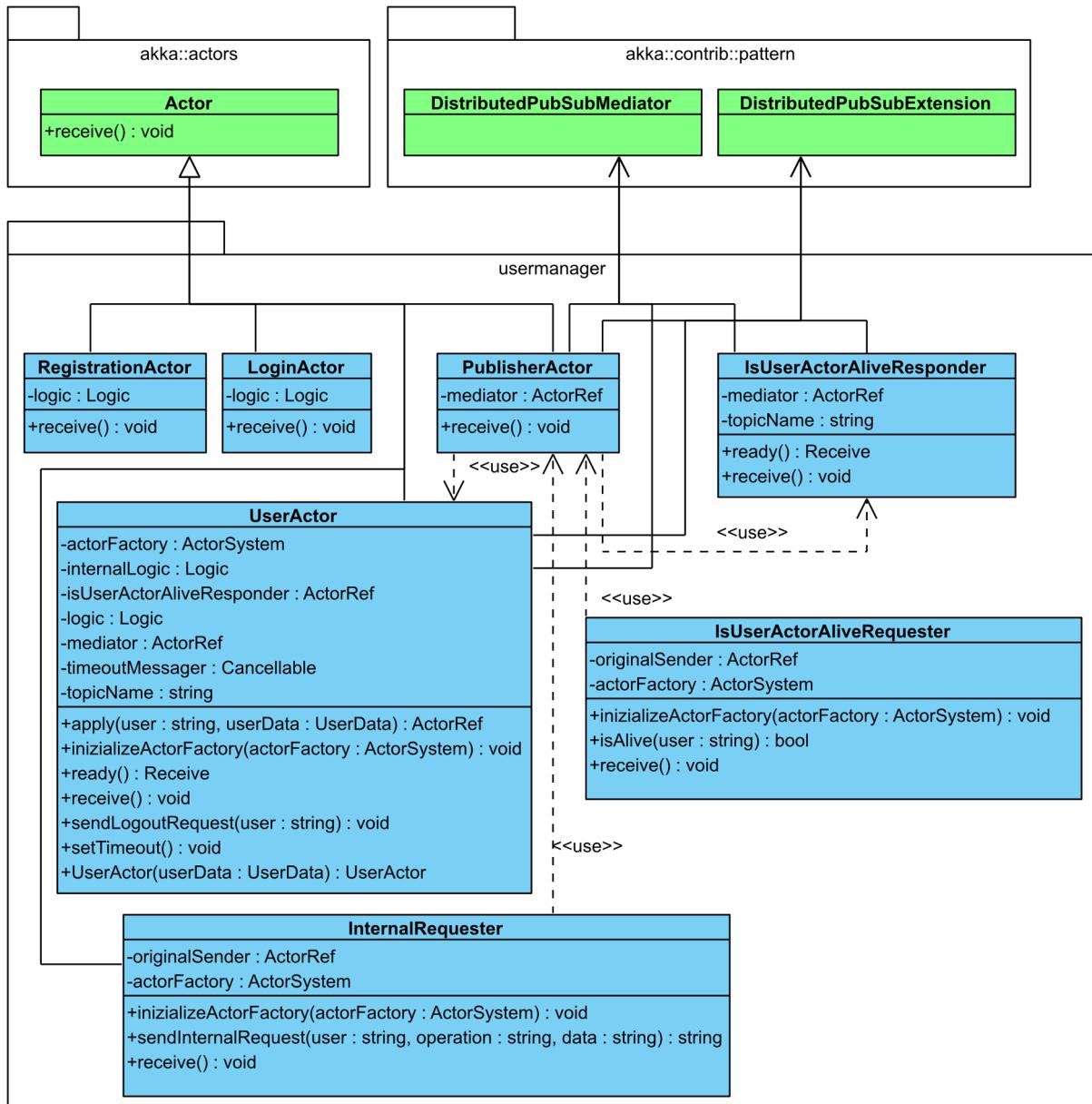


Figura 258: Diagramma della componente `sgad::servertier::presentation::usermanager`

- **Descrizione:** componente per la gestione degli attori relativi alla gestione delle richieste generate dai client. Si occupa di ricercare in tutto il cluster di server l'attore

associato alla richiesta.

- **Padre:** presentation
- **Interazioni con altri componenti**
 - akka::actors.
 - akka::contrib::pattern.
 - logic: componente che gestisce il controllo delle operazioni richieste dal client.
 - httpresponder: componente per la gestione delle richieste HTTP ricevute dai vari client.
 - messages: componente per i messaggi scambiati tra gli attori del sistema.
 - pagemanager: componente per la gestione delle pagine HTML, CSS e sorgenti JavaScript.
 - timeout: componente per la gestione dei timeout.

3.48.2 Classi

3.48.2.1 sgad::servertier::presentation::usermanager::InternalRequester

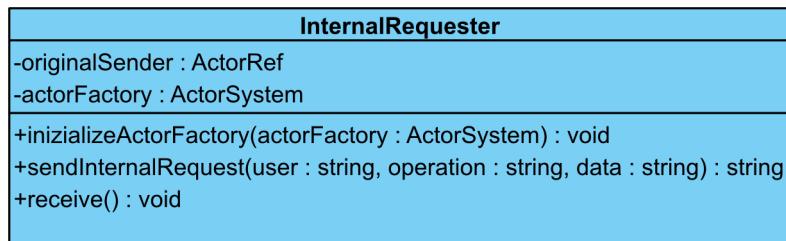


Figura 259: Diagramma della classe sgad::servertier::presentation::usermanager::InternalRequester

- **Descrizione:** classe di gestione della richiesta di user ad autorizzazione interna incapsulandone l'algoritmo.
- **Utilizzo:** viene utilizzata per mandare richieste interne fra attori.
- **Classi ereditate**
 - akka::actors::Actor
- **Attributi**
 - `-actorFactory : ActorSystem`
ActorSystem del sistema con il quale creare nuovi attori.
 - `-originalSender : ActorRef`
Sender iniziale che ha richiesto il servizio a cui si risponderà ottenuta la risposta.

- **Metodi**

- `+receive() : void`
Riceve la richiesta e spedisce la risposta al sender.
- `+sendInternalRequest(data : string, operation : string, user : string) : string`
Controlla se l'UserActor associato all'attore in input è attivo nel cluster.

Argomenti:

- * `data` : dati associati alla richiesta.

- * `operation` : operazione richiesta.

- * `user` : nome dell'username per il quale controllare che esista l'attore associato.

- `+inizializeActorFactory(actorFactory : ActorSystem) : void`
Inizializza la classe settando l'actorSystem con cui questa potrà creare attori.

Argomenti:

- * `actorFactory` : l'actorSystem del sistema con cui verranno creati gli attori.

3.48.2.2 sgad::servertier::presentation::usermanager::IsUserActorAliveRequester

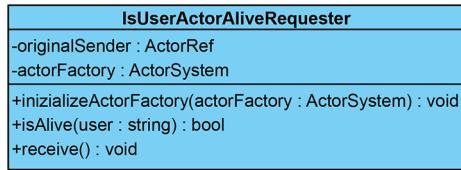


Figura 260: Diagramma della classe `sgad::servertier::presentation::usermanager::IsUserActorAliveRequester`

- **Descrizione:** attore che gestisce la risoluzione della richiesta di esistenza di altri UserActor.
- **Utilizzo:** la classe viene utilizzata per verificare l'esistenza di altri UserActor.
Per un maggior approfondimento, si rimanda alla sezione Gestione di una richiesta di login.
- **Attributi**

- `-actorFactory : ActorSystem`
ActorSystem del sistema con il quale creare nuovi attori.
- `-originalSender : ActorRef`
Sender iniziale che ha richiesto il servizio a cui si risponderà ottenuta la risposta.

- **Metodi**

- `+receive() : void`
Gestisce le richieste e risponde al sender della richiesta.
- `+isAlive(user : string) : bool`
Controlla se l'UserActor associato all'attore in input è attivo nel cluster.

Argomenti:

- * `user` : nome dell'username per il quale controllare che esista l'attore associato.
- `+inizializeActorFactory(actorFactory : ActorSystem) : void`
Inizializza la classe settando l'actorSystem con cui questa potrà creare attori.

Argomenti:

- * `actorFactory` : l'actorSystem del sistema con cui verranno creati gli attori.

3.48.2.3 sgad::servertier::presentation::usermanager::IsUserActorAliveResponder

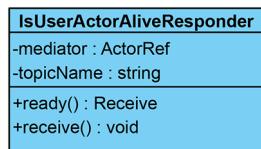


Figura 261: Diagramma della classe `sgad::servertier::presentation::usermanager::IsUserActorAliveResponder`

- **Descrizione:** classe per la risposta della richiesta di esistenza dell'UserActor in modo asincrono.
- **Utilizzo:** la classe viene utilizzata per verificare in modo asincrono l'esistenza dell'UserActor.
- **Attributi**

- `-mediator : ActorRef`
Riferimento all'attore mediatore del sistema fornito da $Akka_{|g|}$.
- `-topicName : string`
Nome del topic a cui l'attore si iscrive, uguale al nome dell'utente associato
- `IsUserActorAliveResponder`.

- **Metodi**

- `+receive() : void`
Riceve ed elabora il messaggio di confermata sottoscrizione al topic set-tando come nuovo receive dei messaggi il metodo ready.
- `+ready() : Receive`
Elabora i messaggi in arrivo all'attore dopo che questo si è sottoscritto con successo al topic.

3.48.2.4 `sgad::servertier::presentation::usermanager::LoginActor`

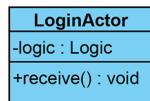


Figura 262: Diagramma della classe `sgad::servertier::presentation::usermanager::-LoginActor`

- **Descrizione:** classe per la gestione delle richieste di login.
- **Utilizzo:** viene utilizzata per controllare i dati di una richiesta di login. Nel caso di login positivo essa istanzia un attore per l'utente e lo registra presso `PublisherActor` del server corretto.
Per un maggior approfondimento, si rimanda alla sezione Gestione di una richiesta di login.
- **Classi ereditate**

 - `akka::actors::Actor`

- **Attributi**

 - `-logic : Logic`
Oggetto di logica che esegue tutte le operazioni richieste.

- **Metodi**

 - `+receive() : void`
Elabora i messaggi di login.

3.48.2.5 sgad::servertier::presentation::usermanager::PublisherActor



Figura 263: Diagramma della classe `sgad::servertier::presentation::usermanager::PublisherActor`

- **Descrizione:** classe che gestisce l'inoltro dei messaggi alle istanze giuste delle classi (attori Akka) del pacchetto `usermanager`.

- **Utilizzo:** viene utilizzato da per l'inoltro della richiesta del client all'attore corretto in grado di gestirla e rispondere. È in grado di capire dove trovare un attore indipendentemente da quale server lo ospita.

Per un maggior approfondimento, si veda la sezione Gestione di una richiesta di gioco.

- **Classi ereditate**

- `akka::actors::Actor`

- **Attributi**

- `-mediator : ActorRef`

Attore di Akka usato per inviare i messaggi agli attori senza sapere su quale server essi siano locati.

- **Metodi**

- `+receive() : void`

Riceve i messaggi di tipo `ToPublisherUserRequest` e li inoltra all'user corretto.

3.48.2.6 sgad::servertier::presentation::usermanager::RegistrationActor

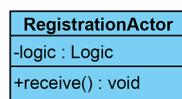


Figura 264: Diagramma della classe `sgad::servertier::presentation::usermanager::RegistrationActor`

- **Descrizione:** classe per la gestione delle richieste di registrazione.

- **Utilizzo:** viene utilizzata per la gestione delle richieste di registrazione. Essa controllerà la validità dei dati e nel caso procederà con la scrittura nel database.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di registrazione e Gestione di una richiesta di login.

- **Classi ereditate**

- `akka::actors::Actor`

- **Attributi**

- `-logic : Logic`

Oggetto di logica che esegue tutte le operazioni richieste.

- **Metodi**

- `+receive() : void`

Elabora i messaggi di registrazione.

3.48.2.7 `sgad::servertier::presentation::usermanager::UserActor`

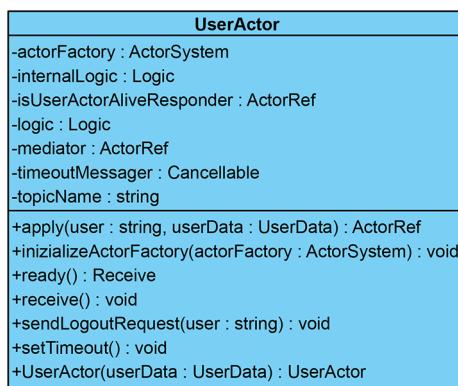


Figura 265: Diagramma della classe `sgad::servertier::presentation::usermanager::UserActor`

- **Descrizione:** classe per la gestione della sessione di un particolare utente. Essa è l'unico accesso per le richieste di manipolazione ai dati di quell'utente.
- **Utilizzo:** viene usata per effettuare le operazioni richieste sui dati di ogni utente. Essa si occupa di caricare i dati dell'utente, di tenerli aggiornati e infine salvarli sul database. Unificando la via di accesso ai dati dell'utente si eliminano potenziali inconsistenze a causa di modifiche concorrenti. La sua logica interna si occuperà anche di controllare la fonte e la validità delle richieste in arrivo prima di eseguirle.

Per un maggior approfondimento, si vedano le sezioni Gestione di una richiesta di login e Gestione di una richiesta di gioco.

- Classi ereditate

- akka::actors::Actor

- Attributi

- -actorFactory : ActorSystem

ActorSystem del sistema con il quale creare nuovi attori.

- -internalLogic : Logic

Oggetto di logica interna che esegue le operazioni interne.

- -isUserActorAliveResponder : ActorRef

Attore che risponderà alle richieste di isAlive.

- -logic : Logic

Oggetto di logica che esegue tutte le operazioni richieste.

- -mediator : ActorRef

Riferimento all'attore mediatore del sistema fornito da Akka.

- -timeoutMessenger : Cancellable

Timeout che rappresenta il tempo massimo di vita dopo ogni richiesta, dopo di che lo UserActor va stoppato salvando prima i dati di gioco sul database.

- -topicName : string

Nome del topic a cui l'attore si iscrive, uguale al nome dell'utente associato.

- Metodi

- +UserActor(userData : UserData) : UserActor

Costruttore della classe.

Argomenti:

- * userData : dati dell'utente associati all'attore.

- +receive() : void

Riceve ed elabora il messaggio di confermata sottoscrizione al topic settando come nuovo receive dei messaggi il metodo ready.

- +ready() : Receive

Elabora i messaggi in arrivo all'attore dopo che questo si è sottoscritto con successo al topic.

- -setTimeout() : void

Cancella il vecchio timeout e ne crea uno nuovo.

- +apply(user : string, userData : UserData) : ActorRef

Crea e fa partire un attore di tipo UserActor.

Argomenti:

- * `user` : nome del nuovo attore.
- * `userData` : dati dell'utente associato.
- `+sendLogoutRequest(user : string) : void`
Manda il messaggio di Timeout per richiedere lo stop all'UserActor in input. La richiesta deve essere inviata dallo stesso server in cui risiede l'attore.

Argomenti:

- * `user` : username dell'utente a cui l'UserActor è associato.
- `+inizializeActorFactory(actorFactory : ActorSystem) : void`
Inizializza la classe settando l'actorSystem con cui questa potrà creare attori.

Argomenti:

- * `actorFactory` : l'actorSystem del sistema con cui verranno creati gli attori.

4 Schema base di dati

Anche se il database utilizzato è *schema-less*_{|g|}, si è deciso di memorizzare comunque i dati con uno schema. Viene di seguito presentato un diagramma.

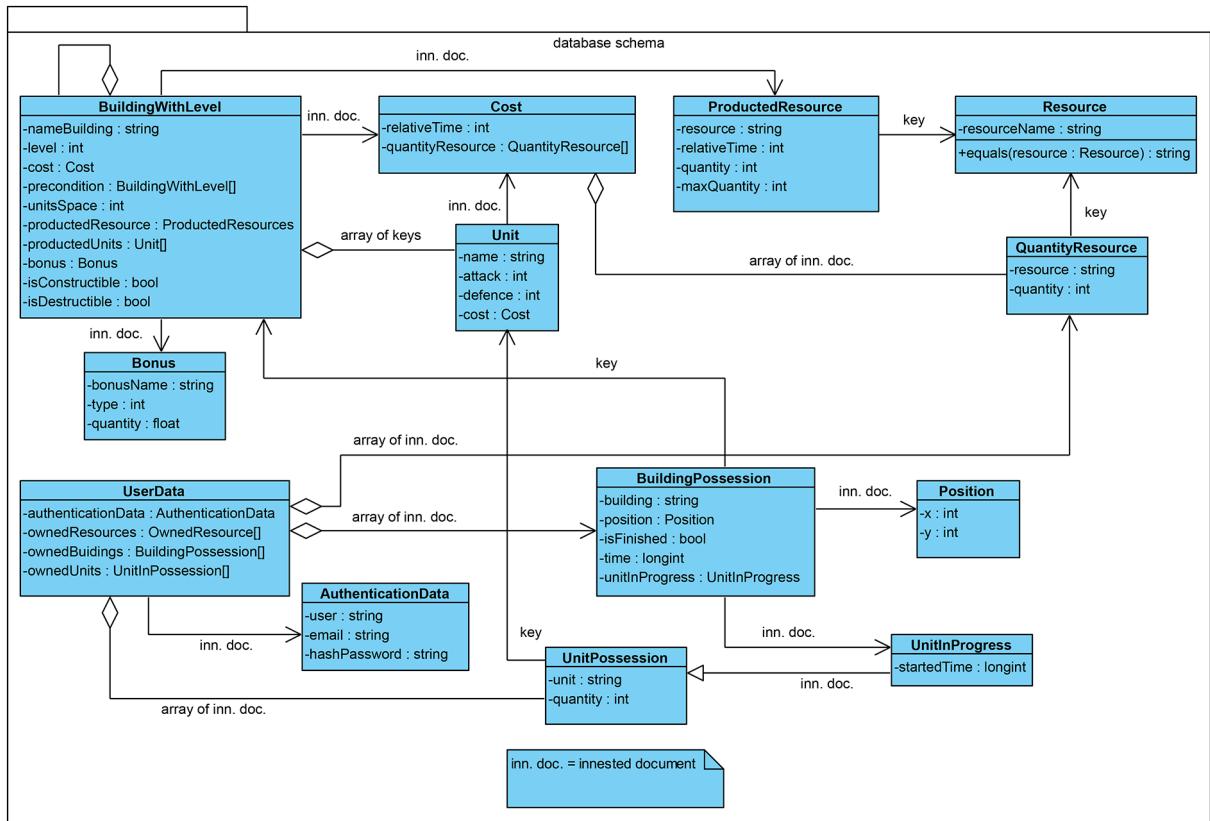


Figura 266: Diagramma dello schema utilizzato per la memorizzazione dei dati sul database

La figura indica l'implementazione delle relazioni in *MongoDB*_{|g|}. Lo schema evidenzia il legame con i pacchetti del model server e client. L'architettura ha lo scopo di memorizzare una singola volta i dati di gioco e agganciare riferimenti di questi agli oggetti degli utenti. Questo permette un buon utilizzo dello spazio evitando sprechi. È così inoltre possibile cambiare i dati di gioco a livello di database e automaticamente tutte le nuove sessioni di gioco si comporteranno seguendo i nuovi dati.

5 Diagrammi di sequenza

Vengono qui riportati i diagrammi di sequenza delle operazioni principali dell'applicazione.

5.1 Gestione degli eventi

Il diagramma in figura 267 illustra la gestione di un generico evento verificatosi canvas rappresentante il mondo di gioco.

La sequenza di azioni viene avviata una volta che l'utente genera un evento, come può essere il click, sul mondo di gioco. Inizialmente l'oggetto **Context** crea una collezione di oggetti grafici e il gestore dell'evento che si verificherà. Una volta che l'evento viene catturato dall'oggetto **Context**, questo richiede all'oggetto **GraphicObject** un oggetto di tipo **Iterator**, mediante il metodo **createIterator()** per poter scorrere la lista di oggetti grafici. Di conseguenza **GraphicObject** creerà un nuovo oggetto di tipo **GraphicObjectIterator** che è un sottotipo di **Iterator** e lo fornirà all'oggetto **Context**.

Una volta ottenuto l'iteratore, l'oggetto **Context** invocherà il metodo **execute(event : Event, iterator : Iterator)** sull'oggetto **onClickEvent**, creato in precedenza, passando come parametri l'evento che si è verificato e l'iteratore sul quale scorrere la lista.

L'oggetto **onClickEvent** richiede quindi all'iteratore di posizionarsi sull'ultimo elemento della collezione di oggetti grafici. L'ultimo elemento è l'elemento con profondità minore, quindi quello più vicino. Di conseguenza, l'oggetto **onClickEvent** sposterà l'iteratore sull'elemento precedente finché non verrà individuato l'oggetto su cui si è effettivamente verificato l'evento.

Per determinare se l'evento si è verificato sull'oggetto individuato dall'iteratore, viene invocato il metodo **areYouClicked(point : Point2D)** dell'oggetto grafico. Tale metodo individua se il punto in cui si è verificato l'evento sia all'interno o meno del poligono che rappresenta l'oggetto grafico. Invoca di conseguenza il metodo **isPointInside(point : Point2D)** sull'oggetto **shape** di tipo **Shape** che memorizza i punti rappresentanti il poligono.

Sull'oggetto di tipo **GraphicObject** che viene puntato dall'iteratore, una volta che la condizione si è verificata, viene invocato il relativo evento su tale oggetto che si occuperà di gestirlo.

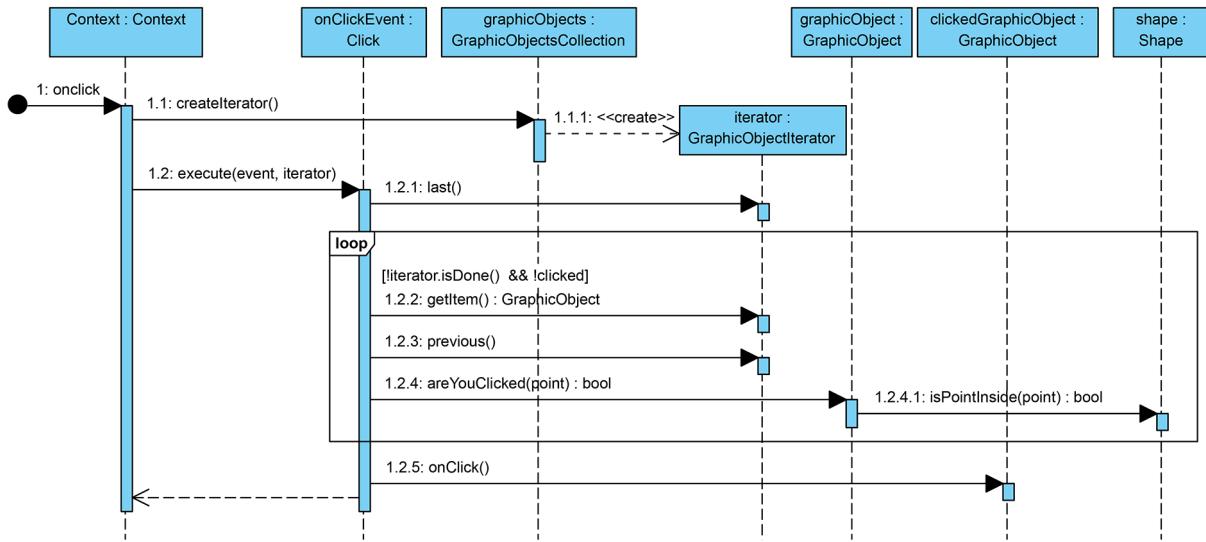


Figura 267: Diagramma di sequenza per la gestione di un evento

5.2 Visualizzazione dei menu

Il diagramma in figura 268 illustra la creazione di un menu.

La sequenza di azioni viene avviata una volta che un'azione, venendo eseguita, crea un menu da visualizzare nell'interfaccia grafica.

L'azione si occuperà inizialmente di creare un oggetto con supertipo `MenuFactory`. Durante la creazione vengono quindi forniti i parametri necessari all'oggetto per costruire il menu adatto.

Viene quindi invocato il metodo `buildMenu()` che richiede al menu di fornire un oggetto di tipo `FrameWidget` contenente tutti gli oggetti grafici del menu. Il metodo crea oggetti di supertipo `Widget` necessari alla visualizzazione del menu a cui vengono associati altri oggetti di supertipo `Action` mediante il metodo `setOnClickEvent(action : Action)` oppure `setOnRightClickAction(action : Action)`. Di conseguenza, una volta che l'utente interagirà con gli oggetti grafici del menu, verranno eseguite le relative azioni.

Una volta creati gli oggetti grafici, questi vengono inseriti in un oggetto di tipo `FrameWidget` che verrà restituito all'oggetto che ha richiesto la creazione del menu.

Vengono poi ottenuti gli oggetti di supertipo `Widget` creati dall'oggetto di tipo `FrameWidget` fornito. Questi vengono quindi aggiunti all'oggetto `Context` che si occuperà di visualizzarli nel mondo di gioco.

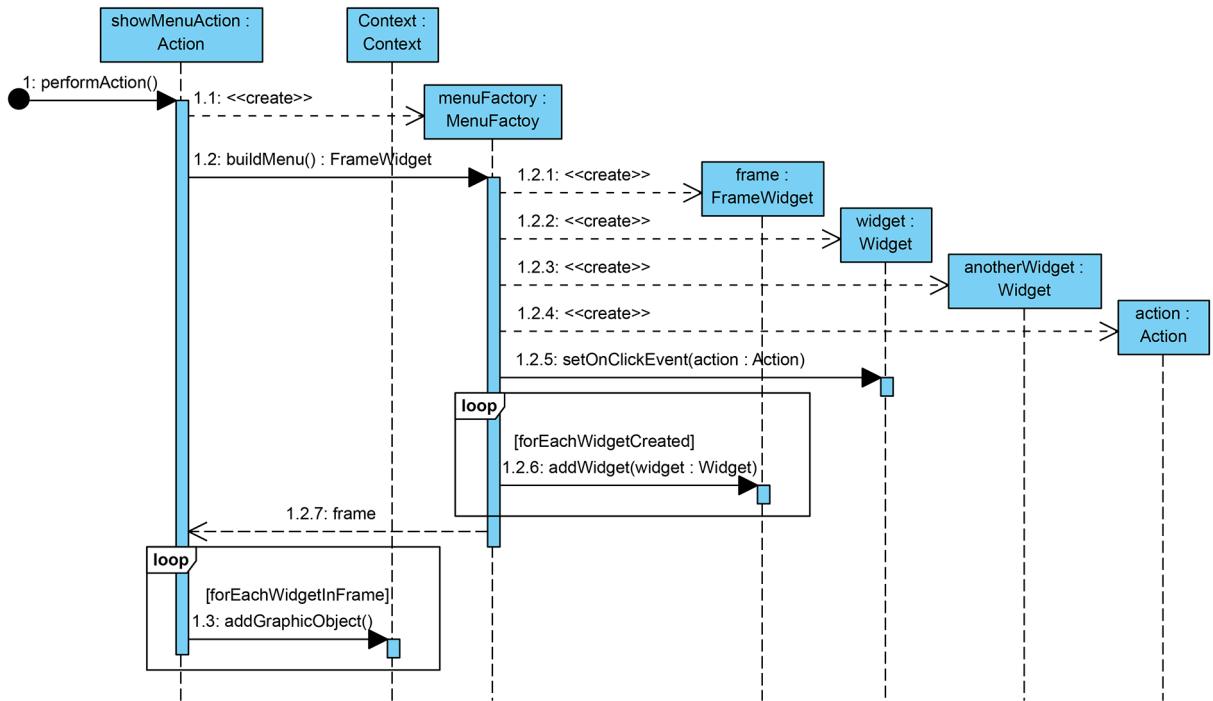


Figura 268: Diagramma di sequenza per la visualizzazione di un menu

5.3 Interazione con il server

Il diagramma in figura 269 illustra l'interazione del client con il server.

La sequenza di azioni viene avviata una volta che un oggetto di tipo `AJAXRequester` invia una richiesta al server.

In seguito all'invio della richiesta, possono verificarsi due casi. Il primo caso prevede una risposta corretta da parte del server, impostando l'attributo `requestStatus` con valore pari a 200. Nel secondo caso è stato impostato l'attributo `requestStatus` con un valore diverso da 200, indicante che la richiesta non è stata soddisfatta correttamente.

Nel caso in cui la richiesta sia stata soddisfatta correttamente, la risposta fornita viene gestita da un oggetto di tipo `MessageInterpreter` che si occuperà di analizzarla per determinare l'eventuale presenza di messaggi in piggybacking.

Per ogni ulteriore messaggio, l'oggetto `MessageInterpreter` crea l'azione indicata nel messaggio. A tale azione vengono quindi impostati i dati associati al messaggio analizzato. In seguito l'azione viene eseguita per rendere consistente il modello del client con quello del server. Infine, vengono forniti all'invocatore, i dati della richiesta inviata.

Una volta ottenuti i dati, questi vengono passati all'oggetto `onSuccess` di supertipo `Action`. Viene poi invocato il metodo `performAction()` di tale oggetto per eseguire l'azione.

Se invece la richiesta non ha avuto successo, viene direttamente eseguito il metodo `performAction` dell'oggetto `onFailureAction` che si occuperà di visualizzare l'avvenuto errore di comunicazione con il server.

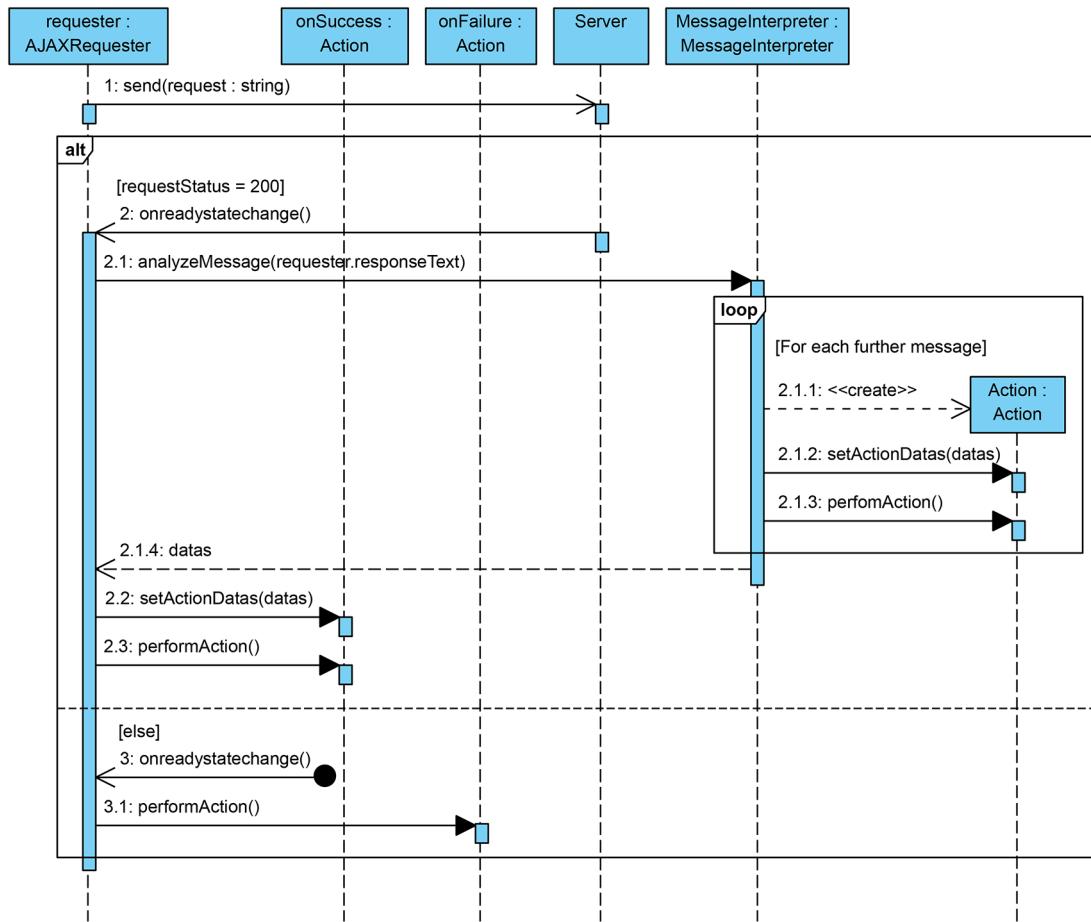


Figura 269: Diagramma di sequenza per l'interazione con il server

5.4 Gestione di una richiesta di registrazione

Il diagramma in figura 270 illustra la gestione da parte del server di una richiesta di registrazione.

La sequenza di azioni viene avviata quando un client tramite un *browser_g* web invia una richiesta in “post” contenente i campi: *user*, *email*, *password1* e *password2*.

La richiesta con i campi corretti viene raccolta dal *ResponderActor* il quale crea un nuovo attore di tipo *WorkerActor* al quale lancia una *ask* di tipo *ToWorkerLoginRequest* contenente i dati della richiesta. Essa ritorna un *Future[String]*. Questo *future_g* di tipo stringa è passato alla gestione interna del framework *spray_g* che si occuperà di risolverlo e rispondere al client lasciando il responder actor libero di rispondere ad altre richieste. Viene creato un nuovo *WorkerActor* per ogni richiesta allo scopo di ottenere il massimo parallelismo nel *ResponderActor*.

Il *WorkerActor* ricevuto il messaggio un crea attore di tipo *RegistrationActor* allo scopo di non serializzare tutte le richieste di registrazione. Manda poi al nuovo attore un messaggio con un oggetto di tipo *ToRegistrationActorRequest* contenente i dati convertiti nel formato standard interpretabile dalla logica e si mette in attesa di ricevere

risposta.

Il `RegistrationActor` ottenuto il messaggio ne affida la gestione all'oggetto `logic` di tipo `Logic` creato in fase di costruzione.

L'oggetto `logic` gestisce la richiesta ottenendo un oggetto di tipo `Registration` dall'`OperationFactory` e affidando a essa la gestione.

L'oggetto di tipo `Registration` gestisce la richiesta dapprima controllando la validità dei dati in input decodificati grazie alla chiamata interna `decodeData()`, se non validano risponde tramite la richiesta `getHomePageWithErrors(errors)` al `PageFactory`. Se i dati validano allora crea un nuovo `UserData` con i dati del nuovo utente e tenta di inserirlo nel database grazie alla chiamata `insertNewUser()` del `DataBaseManager`. Esso garantirà l'unicità dei campi "user" ed "email". Se il risultato della chiamata va a buon fine risponde con la richiesta `getHomePageRegistrationSuccessful()` al `PageFactory`. Se l'inserimento fallisce allora risponde con la richiesta `getHomePageWithErrors(errors)`. A loro volta tutti gli oggetti rispondono fino a far arrivare la risposta al client.

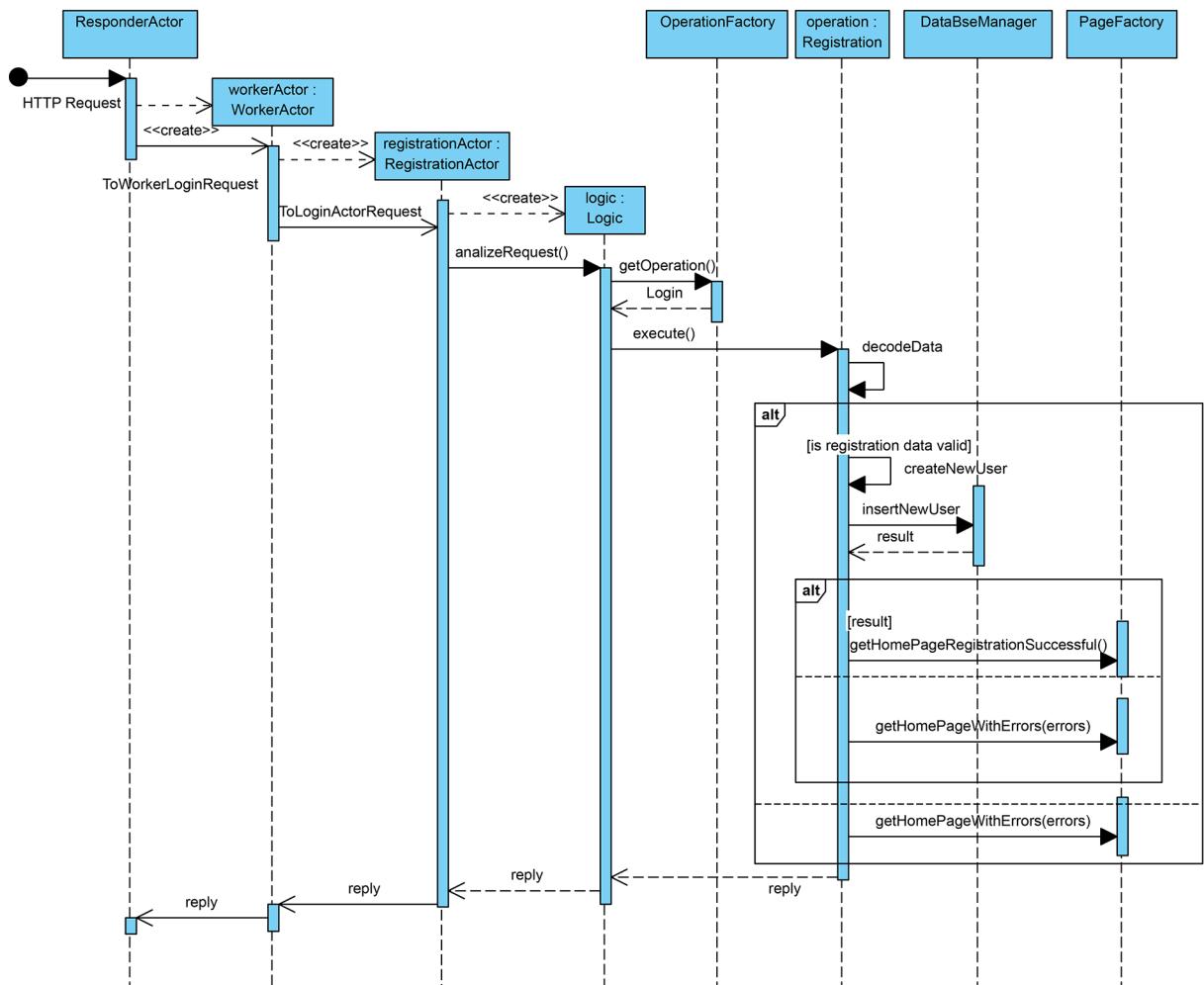


Figura 270: Diagramma di sequenza per la gestione di una richiesta di registrazione

5.5 Gestione di una richiesta di login

Il diagramma in figura 271 illustra la gestione da parte del server di una richiesta di login.

La sequenza di azioni viene avviata una volta che un client tramite un browser web invia una richiesta in post contenente i campi: **user** e **password**.

La richiesta con i campi corretti viene raccolta dal **ResponderActor** il quale crea un nuovo attore di tipo **WorkerActor** al quale lancia una ask di tipo **ToWorkerLoginRequest** contenente i dati della richiesta. Essa ritorna un **Future[String]**. Questo future di tipo stringa è passato alla gestione interna del framework spray che si occuperà di risolverlo e rispondere al client lasciando il responder actor libero di rispondere ad altre richieste. Viene creato un nuovo **WorkerActor** per ogni richiesta allo scopo di ottenere il massimo parallelismo nel **ResponderActor**.

Il **WorkerActor** ricevuto il messaggio crea un attore di **LoginActor** allo scopo di non

5 Diagrammi di sequenza

serializzare tutte le richieste di registrazione. Manda poi al nuovo attore un messaggio con un oggetto di tipo `ToLoginActorRequest` contenente i dati convertiti nel formato standard interpretabile dalla logica e si mette in attesa di ricevere risposta.

Il `LoginActor` ottenuto il messaggio ne affida la gestione all'oggetto `logic` di tipo `Logic` creato in fase di costruzione.

L'oggetto `logic` gestisce la richiesta ottenendo un oggetto di tipo `Login` dall'`OperationFactory` e affidando a essa la gestione.

L'oggetto di tipo `Registration` gestisce la richiesta tentando di caricare i dati dell'utente dal database grazie alla chiamata `loadUserData()` del `DataBaseManager` con i dati in input decodificati grazie alla chiamata interna `decodeData()`. Se il risultato della chiamata va a buon fine controlla se l'`UserActor` associato all'utente è già presente nel cluster tramite la chiamata `isAlive()` al `IsUserActorAliveRequester`. Se l'attore non è presente allora ne richiede la creazione. Infine grazie alle richieste `getCanvas1()`, `getCanvas2ToReplace()` e `getCanvas3()` al `PageFactory` ottiene i dati di gioco da mandare al client. Collega e inserisce i dati corretti nel `canvas2` e spedisce al mittente il risultato.

Se invece il login fallisce risponde con la richiesta `getHomePageWithErrors(errors)`. A loro volta tutti gli oggetti rispondono fino a far arrivare la risposta al client.

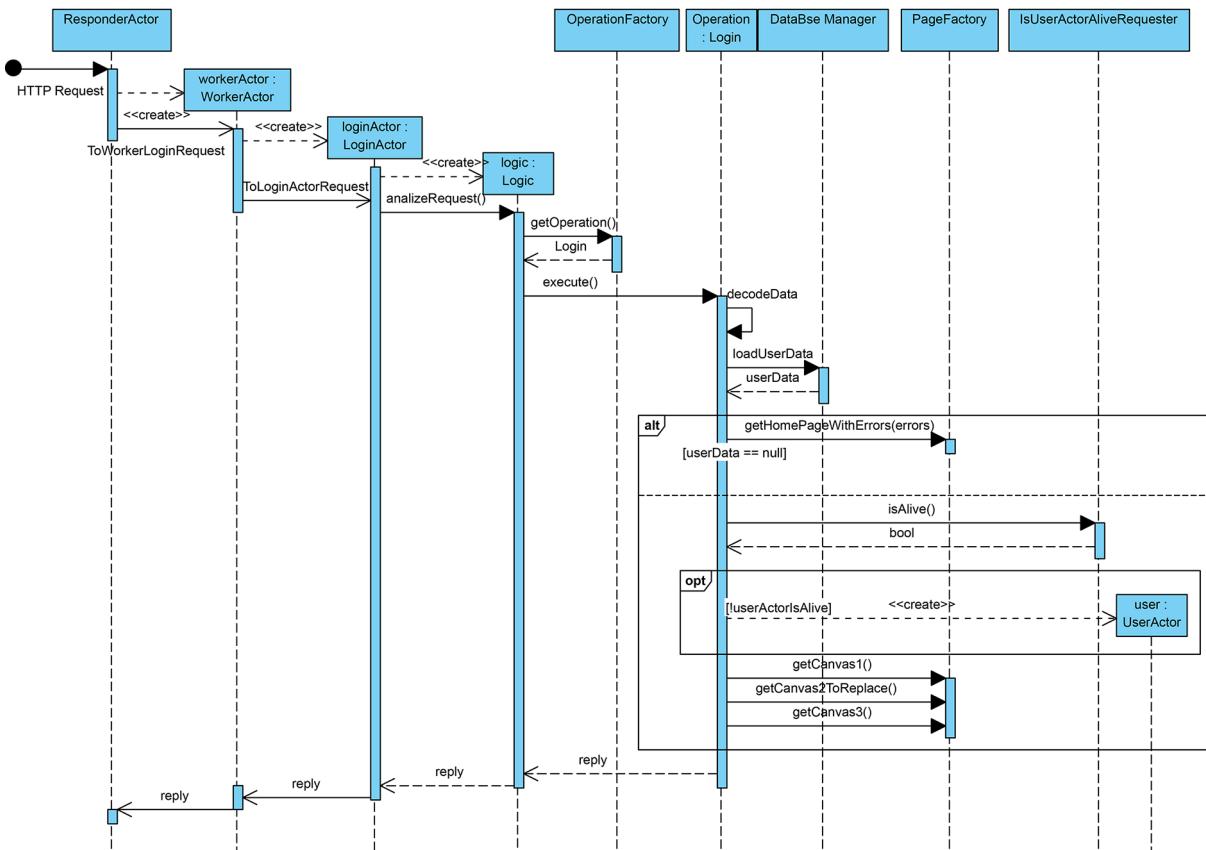


Figura 271: Diagramma di sequenza per la gestione di una richiesta di login

5.6 Gestione di una richiesta di gioco

Il diagramma in figura 272 illustra la gestione da parte del server di una richiesta di gioco.

La sequenza di azioni viene avviata una volta che un client tramite un browser web ed il codice di gioco lato client invia una richiesta in post contenente i campi: `user`, `operation` e `data`.

La richiesta con i campi corretti viene raccolta dal `ResponderActor` il quale crea un nuovo attore di tipo `WorkerActor` al quale lancia una ask di tipo `ToWorkerUserRequest` contenente i dati della richiesta. Essa ritorna un `Future[String]`. Questo future di tipo stringa è passato alla gestione interna del framework spray che si occuperà di risolverlo e rispondere al client lasciando il responder actor libero di rispondere ad altre richieste. Viene creato un nuovo `WorkerActor` per ogni richiesta allo scopo di ottenere il massimo parallelismo nel `ResponderActor`.

Il `WorkerActor` ricevuto il messaggio manda un messaggio al `PublisherActor` con un oggetto di tipo `ToPublisherAndUserRequest` contenente i dati convertiti nel formato standard interpretabile dalla logica e si mette in attesa di ricevere risposta.

Il `PublisherActor` inoltra rapidamente il messaggio all'`UserActor` senza sapere in quale server del cluster esso sia locato.

Lo `UserActor` ottenuto il messaggio ne affida la gestione all'oggetto `logic` di tipo `Logic` creato in fase di costruzione.

L'oggetto `logic` gestisce la richiesta ottenendo un oggetto di tipo `ConcreteOperation` dall'`OperationFactory` e affidando a essa la gestione. Il vero tipo dell'oggetto `ConcreteOperation` dipende dall'operazione richiesta dal client.

L'oggetto di tipo `ConcreteOperation` gestisce la richiesta ottenendo i dati grazie alla chiamata interna `decodeData()` e modificando se corretti i dati dell'utente. Elabora infine una risposta. A loro volta tutti gli oggetti rispondono fino a far arrivare la risposta al client.

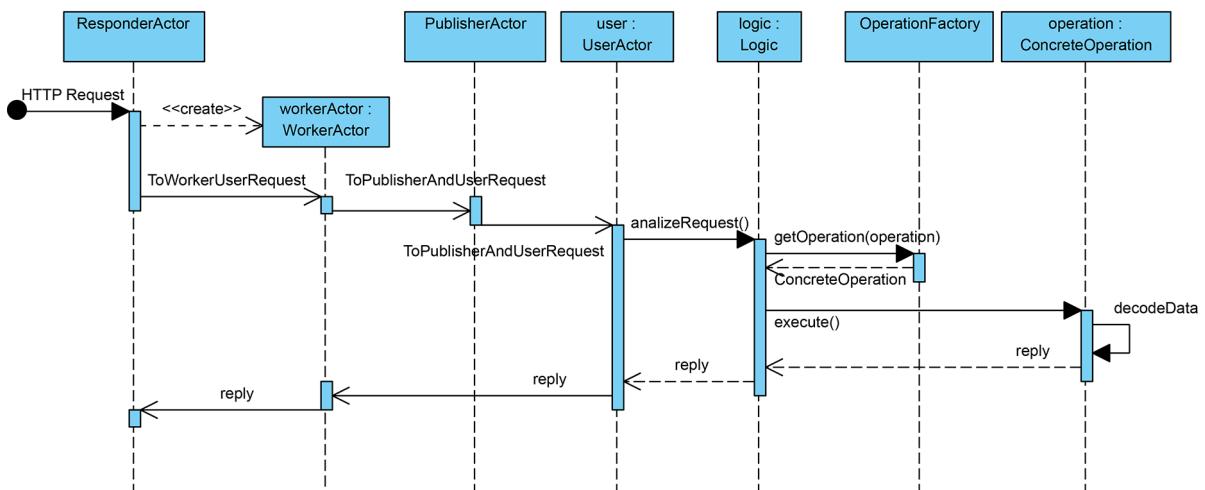


Figura 272: Diagramma di sequenza per la gestione di una richiesta di gioco

6 Tracciamento

6.1 Tracciamento Classi-Requisiti

Classi	Requisiti
akka::actors::Actor	R0F5 R0F5.1
akka::contrib::pattern::DistributedPubSubMediator	R0F5 R0F5.1
casbah::DB	R0F5 R0F5.1
casbah::DBCollection	R0F5 R0F5.1
casbah::DBObject	R0F5 R0F5.1
casbah::MongoClient	R0F5 R0F5.1
sgad::clienttier::controller::actions::Action	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ActionListener	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::BackToVillage	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2

	R2F5.2.1
sgad::clienttier::controller::actions::BuildConstruction	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.3.2.1 R0F5.1.4 R0F5.1.4.3 R0F5.1.4.3.2 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- ChangePasswordAction	R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2
sgad::clienttier::controller::actions::- CloseContextualMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::DeleteAccountAction	R0F4.3
sgad::clienttier::controller::actions::DemolishBuilding	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.5.2 R1F5.1.5.2.1 R1F5.1.5.2.2 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.1

	R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::DismissUnits	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.1.6.3 R1F5.1.6.4 R1F5.1.6.5 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::EnableRightClick	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::HarvestResources	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.1.3 R0F5.1.1.6 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::LoadAnotherUser	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1

	R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::LoadGeneralData	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::LoadPersonalData	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::Logout	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ReloadPageAction	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- RemoveGraphicObjectAction	R0F5 R0F5.1 R0F5.1.1

	R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::RequestAttackUser	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.2
sgad::clienttier::controller::actions::-RequestBuildConstruction	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.3 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-RequestChangePassword	R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2
sgad::clienttier::controller::actions::-RequestDeleteAccount	R0F4.3
sgad::clienttier::controller::actions::-RequestDemolishBuilding	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4

	R1F5.1.5 R1F5.1.5.2 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::RequestDismissUnits	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-RequestHarvestResources	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-RequestLoadGeneralData	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-RequestLoadPersonalData	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5

	R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::RequestTrainUnits	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.4 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-RequestUpgradeBuilding	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::SetAllVoidFilter	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::SetBuildModeFilter	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1

sgad::clienttier::controller::actions::- ShowAccountDeletedMenu	R0F4.3
sgad::clienttier::controller::actions::- ShowAnotherUserMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-ShowAttackMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.1.1.1
sgad::clienttier::controller::actions::- ShowAttackResultMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.1.1.1 R2F5.2.1.3
sgad::clienttier::controller::actions::- ShowBuildConstructionMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1

	R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- ShowBuildingContextualMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ShowDemolishMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- ShowDismissUnitMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.1.6.2 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ShowInteractionMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5

	R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- ShowOperationFailureMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- ShowPasswordChangedMenu	R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.2.1 R0F4.2.2
sgad::clienttier::controller::actions::ShowResourceMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::- ShowTileContextualMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.1 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ShowTrainUnitMenu	R0F5

	R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.3 R0F5.1.2.3.1 R0F5.1.2.4.1 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::-ShowUnitSelectionMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ShowUpgradeMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::ShowUserListMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::StoleResources	R0F5

	R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.4 R2F5.2.1.4.1
sgad::clienttier::controller::actions::TrainUnit	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.4 R0F5.1.2.5 R0F5.1.2.7.1 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::actions::UpgradeBuilding	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.3.2 R0F5.1.3.2.1 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::backupmanager::-BackupManager	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2

	R2F5.2.1
sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3
sgad::clienttier::controller::menufactory::- AnotherUserMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- AttackResultMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.3
sgad::clienttier::controller::menufactory::- BuildingContextualMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.3.1.1 R0F5.1.4 R1F5.1.5 R1F5.1.5.1 R1F5.1.5.1.1

	R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- ConfirmMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- InteractionMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::MenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.3 R0F5.1.3.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- NotifyMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1

	R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- ResourceMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- TileContextualMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.1.1 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- UnitSelectionMenu	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.7 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::menufactory::- UserListMenuFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4

	R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.1
sgad::clienttier::controller::messageinterpreter::-MessageInterpreter	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.2 R0F5.1.2.4 R0F5.1.3 R0F5.1.3.2 R0F5.1.4 R0F5.1.4.3 R1F5.1.5 R1F5.1.5.2 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::controller::requester::AJAXRequester	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.2 R0F5.1.2.4 R0F5.1.3 R0F5.1.3.2 R0F5.1.4 R0F5.1.4.3 R1F5.1.5 R1F5.1.5.2 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::Bonus	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.4.3 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2

	R2F5.2.1
sgad::clienttier::model::generaldata::BuildingWithLevel	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.1.3 R0F5.1.1.5 R0F5.1.2 R0F5.1.2.1.1 R0F5.1.2.2.1 R0F5.1.2.5 R0F5.1.2.7 R0F5.1.2.7.2 R0F5.1.3 R0F5.1.3.1.1 R0F5.1.3.1.3 R0F5.1.3.2 R0F5.1.3.2.1 R0F5.1.3.2.2 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.1.1 R0F5.1.4.1.2 R0F5.1.4.3.2 R0F5.1.4.3.3 R1F5.1.5 R1F5.1.5.1.1 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::Cost	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.4.1 R0F5.1.2.4.3 R0F5.1.2.6 R0F5.1.3 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.3.2.2 R0F5.1.4 R0F5.1.4.1.1 R0F5.1.4.3.2

	R0F5.1.4.3.3 R1F5.1.5 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::DataFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.1.1 R0F5.1.4.1.2 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::ProducedResource	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.3 R0F5.1.1.4 R0F5.1.1.5 R0F5.1.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.3.1 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::QuantityResource	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.4.1 R0F5.1.2.6 R0F5.1.3 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.4 R0F5.1.4.1.1 R0F5.1.4.3.2

	R1F5.1.5 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::Resource	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.1.5 R0F5.1.2 R0F5.1.2.4.1 R0F5.1.2.6 R0F5.1.3 R0F5.1.4 R0F5.1.4.1.1 R0F5.1.4.3.1 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::generaldata::Unit	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.2.1 R0F5.1.2.4.1 R0F5.1.2.4.3 R0F5.1.2.6 R0F5.1.2.7 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.1.1 R2F5.2.1.1.2
sgad::clienttier::model::observer::Observable	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5

	R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::observer::Observer	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::personaldata::-AuthenticationData	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::personaldata::BuildingPossession	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.1.3 R0F5.1.1.4 R0F5.1.2 R0F5.1.2.4.2 R0F5.1.2.4.3 R0F5.1.2.5 R0F5.1.2.7.1

	R0F5.1.3 R0F5.1.3.1.1 R0F5.1.3.1.3 R0F5.1.3.2 R0F5.1.3.2.2 R0F5.1.4 R0F5.1.4.3 R0F5.1.4.3.1 R0F5.1.4.3.3 R1F5.1.5 R1F5.1.5.2 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::personaldata::OwnedResource	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.2 R0F5.1.2.4.1 R0F5.1.3 R0F5.1.3.1.3 R0F5.1.4 R0F5.1.4.1.1 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.4.1
sgad::clienttier::model::personaldata::Position	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::personaldata::UnitInProgress	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.4 R0F5.1.2.4.3 R0F5.1.2.5

	R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::personaldata::UnitPossession	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::model::personaldata::UserData	R0F3 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.2 R0F5.1.2.3.1 R0F5.1.2.4 R0F5.1.2.4.1 R0F5.1.2.5 R0F5.1.2.7 R0F5.1.3 R0F5.1.3.1.1 R0F5.1.3.2.1 R0F5.1.4 R0F5.1.4.1.1 R0F5.1.4.3.2 R1F5.1.5 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.1

	R1F5.2 R2F5.2.1 R2F5.2.1.1.1 R2F5.2.1.4.1
sgad::clienttier::model::userdatamanager::- UserDataManager	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.2 R0F5.1.2 R0F5.1.2.3.1 R0F5.1.2.4 R0F5.1.2.4.1 R0F5.1.2.5 R0F5.1.2.7 R0F5.1.3 R0F5.1.3.1.1 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.4 R0F5.1.4.1.1 R0F5.1.4.3.2 R1F5.1.5 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.1 R1F5.2 R2F5.2.1 R2F5.2.1.1.1 R2F5.2.1.4.1
sgad::clienttier::view::commands::Click	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.1 R0F5.1.2

	R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.2.1 R1F5.1.5 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.1 R2F5.2.1.4 R2F5.2.1.4.2
sgad::clienttier::view::commands::Command	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::commands::DragAndDrop	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::commands::Resize	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::commands::RightClick	R0F5 R0F5.1

	R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.2 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::context::Context	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.2 R0F5.1.4.2.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.5.2.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.1 R2F5.2.1.4 R2F5.2.1.4.2

sgad::clienttier::view::graphicobjects::bound::Bound	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::collection::-Collection	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R0F5.1.4.1 R1F5.1.5 R1F5.1.5.1

	R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.1 R2F5.2.1.4 R2F5.2.1.4.2
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R0F5.1.4.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.1 R2F5.2.1.4 R2F5.2.1.4.2
sgad::clienttier::view::graphicobjects::collection::Iterator	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::filter::-BuildModeFilter	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5

	R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::filter::-GraphicFilter	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::filter::-VoidFilter	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponent::BuildingComponent	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.5.2 R1F5.1.5.2.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.4.2
sgad::clienttier::view::graphicobjects::components::-worldcomponent::TileComponent	R0F5 R0F5.1 R0F5.1.1

	R0F5.1.2 R0F5.1.3 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.2 R0F5.1.4.2.1 R1F5.1.5 R1F5.1.5.2.1 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponent::WorldComponent	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL1Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL2Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1

sgad::clienttier::view::graphicobjects::components::worldcomponentshape::BarracksL3Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::worldcomponentshape::InConstructionShape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL1Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL2Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::worldcomponentshape::MineL3Shape	R0F5

	R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::SchoolOfMagicL1Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::SchoolOfMagicL2Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::SchoolOfMagicL3Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::StableL1Shape	R0F5 R0F5.1 R0F5.1.1

	R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::StableL2Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::StableL3Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::TileShape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::WizardTowerL1Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3

	R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::WizardTowerL2Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::WizardTowerL3Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::WorldComponentShapeFactory	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::WorldComponentShapeImg	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5

	R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::graphicobject::-GraphicObject	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::point::Point2D	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::shape::Shape	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::clienttier::view::graphicobjects::widget::-ButtonWidget	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3

	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R0F5.1.4.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.3
sgad::clienttier::view::graphicobjects::widget::: FrameWidget	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.3.1.1 R0F5.1.4 R0F5.1.4.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1

	R2F5.2.1.3
sgad::clienttier::view::graphicobjects::widget::-ImageWidget	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.3.1.1 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.1.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1 R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.3
sgad::clienttier::view::graphicobjects::widget::-TextWidget	R0F3 R0F3.1 R0F3.2 R0F3.3 R0F4 R0F4.1 R0F4.1.1 R0F4.1.2 R0F4.1.3 R0F4.2 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.3 R0F5.1.3.1 R0F5.1.4 R0F5.1.4.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.6.1
	R1F5.1.6.1

	R1F5.1.6.1.1 R1F5.2 R2F5.2.1 R2F5.2.1.3
sgad::clienttier::view::graphicobjects::widget::Widget	R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R1F5.1.6.1 R1F5.2 R2F5.2.1
sgad::servtier::application::Application	R0F5 R0F5.1 R1F5.2
sgad::servtier::businesslogic::logic::Logic	R0F2 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servtier::businesslogic::operations::Attack	R2F5.2.1 R2F5.2.1.1 R2F5.2.1.1.1 R2F5.2.1.2 R2F5.2.1.3
sgad::servtier::businesslogic::operations::- AttackUserVillage	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::- BuildConstruction	R0F5 R0F5.1 R0F5.1.4 R0F5.1.4.1 R0F5.1.4.1.1 R0F5.1.4.2 R0F5.1.4.2.1 R0F5.1.4.3 R0F5.1.4.3.2 R0F5.1.4.3.4

	R1F5.1.5 R2F5.2.1
sgad::servtier::businesslogic::operations::- ChangeAccountData	R0F4 R0F4.1 R0F4.1.1 R0F4.1.1.1 R0F4.1.2 R0F4.2 R0F4.2.1 R0F4.2.2
sgad::servtier::businesslogic::operations::DeleteAccount	R0F4 R0F4.1.1 R0F4.3
sgad::servtier::businesslogic::operations::- DemolishBuilding	R0F5 R0F5.1 R1F5.1.5 R1F5.1.5.1 R1F5.1.5.2.2 R1F5.1.5.2.3
sgad::servtier::businesslogic::operations::DismissUnit	R0F5 R0F5.1 R1F5.1.6.1 R1F5.1.6.2 R1F5.1.6.2.1 R1F5.1.6.3 R1F5.1.6.4 R1F5.1.6.5
sgad::servtier::businesslogic::operations::- DonateResources	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::DonateUnits	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::- GetAllServerData	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::- GetServerData	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::- HarvestResource	R0F5 R0F5.1 R0F5.1.1 R0F5.1.1.1 R0F5.1.1.3

	R0F5.1.1.6 R0F5.1.4.3.1
sgad::servtier::businesslogic::operations::InternalLogin	R2F5.2.1
sgad::servtier::businesslogic::operations::-LoadFinishedBuilding	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::-LoadGlobalData	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::LoadVillage	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::LoadUserList	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::Login	R0F1 R0F2 R0F2.3.1 R0F2.3.2 R0F4 R0F4.1 R0F5 R0F5.1 R0F5.1.2
sgad::servtier::businesslogic::operations::Logout	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::Operation	R0F1 R0F1.1.1 R0F1.2.1 R0F1.2.2 R0F1.3.1 R0F1.5.3 R0F2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servtier::businesslogic::operations::-OperationFactory	R0F5 R0F5.1

sgad::servtier::businesslogic::operations::ReceiveAttack	R2F5.2.1 R2F5.2.1.1.1 R2F5.2.1.1.1.1
sgad::servtier::businesslogic::operations::- ReceiveStealResouce	R2F5.2.1.4
sgad::servtier::businesslogic::operations::Registration	R0F1 R0F1.1.1 R0F1.2.1 R0F1.2.2 R0F1.3.1 R0F1.4 R0F1.4.1 R0F1.4.2 R0F1.4.3 R0F1.4.4 R0F1.4.5 R0F1.5 R0F1.5.1 R0F1.5.2 R0F5 R0F5.1
sgad::servtier::businesslogic::operations::SaveUser	R0F5 R0F5.1
sgad::servtier::businesslogic::operations::StealResource	R2F5.2.1 R2F5.2.1.4
sgad::servtier::businesslogic::operations::StealResources	R0F5 R0F5.1 R2F5.2.1 R2F5.2.1.4 R2F5.2.1.4.1 R2F5.2.1.4.2
sgad::servtier::businesslogic::operations::TrainUnit	R0F5 R0F5.1 R0F5.1.2 R0F5.1.2.1 R0F5.1.2.2 R0F5.1.2.3 R0F5.1.2.4 R0F5.1.2.4.1 R0F5.1.2.5 R0F5.1.2.7 R0F5.1.2.7.1
sgad::servtier::businesslogic::operations::- UpdateUserData	R0F5 R0F5.1 R0F5.1.4.1.1

sgad::servtier::businesslogic::operations::- UpgradeBuilding	R0F5 R0F5.1 R0F5.1.3 R0F5.1.3.1 R0F5.1.3.1.2
sgad::servtier::dataaccess::data::shareddata::Bonus	R0F1 R0F1.4 R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel	R0F1 R0F1.4 R0F1.4.1 R0F1.4.2 R0F1.4.3 R0F2 R0F5 R0F5.1 R0F5.1.1.2 R0F5.1.1.3 R0F5.1.1.5 R0F5.1.2.1 R0F5.1.2.1.1 R0F5.1.2.2.1 R0F5.1.2.5 R0F5.1.2.7.2 R0F5.1.3.1.1 R0F5.1.3.1.2 R0F5.1.3.1.3 R0F5.1.3.2 R0F5.1.3.2.1 R0F5.1.3.2.2 R0F5.1.4.1.1 R0F5.1.4.3.1 R0F5.1.4.3.2 R0F5.1.4.3.3 R1F5.1.5.1.1 R1F5.1.5.2.3
sgad::servtier::dataaccess::data::shareddata::Cost	R0F1 R0F1.4 R0F2 R0F5 R0F5.1 R0F5.1.2.4.1

	R0F5.1.2.6 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.3.2.2 R0F5.1.4.1.1 R0F5.1.4.3.2 R0F5.1.4.3.3 R1F5.1.5.2.3
sgad::servtier::dataaccess::data::shareddata::- DataFactory	R0F1 R0F1.4 R0F1.4.1 R0F1.4.2 R0F1.4.3 R0F1.4.4 R0F1.4.5 R0F2 R0F5 R0F5.1 R0F5.1.3.1.1 R0F5.1.4.1.2
sgad::servtier::dataaccess::data::shareddata::- ProducedResource	R0F1 R0F1.4 R0F2 R0F5 R0F5.1 R0F5.1.1.3 R0F5.1.1.4 R0F5.1.1.5 R0F5.1.4.3.1
sgad::servtier::dataaccess::data::shareddata::- QuantityResource	R0F1 R0F1.4 R0F2 R0F5 R0F5.1 R0F5.1.2.4.1 R0F5.1.2.6 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.4.1.1 R0F5.1.4.3.2 R1F5.1.5.2.3
sgad::servtier::dataaccess::data::shareddata::Resource	R0F1 R0F1.4

	R0F1.4.5 R0F2 R0F5 R0F5.1 R0F5.1.1.2 R0F5.1.1.5 R0F5.1.2.4.1 R0F5.1.2.6 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.4.3.1 R0F5.1.4.3.2 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R2F5.2.1 R2F5.2.1.4.1
sgad::servtier::dataaccess::data::shareddata::Unit	R0F1 R0F1.4 R0F1.4.4 R0F2 R0F5 R0F5.1 R0F5.1.2.1 R0F5.1.2.1.1 R0F5.1.2.2.1 R0F5.1.2.4 R0F5.1.2.4.1 R0F5.1.2.6 R0F5.1.4.3.4 R1F5.1.6.3 R2F5.2.1 R2F5.2.1.1.1.1 R2F5.2.1.1.1.2 R2F5.2.1.2
sgad::servtier::dataaccess::data::userdata::-AuthenticationData	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1

	R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servertier::dataaccess::data::userdata::- BuildingPossession	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F1.4.1 R0F1.4.2 R0F1.4.3 R0F2 R0F5 R0F5.1 R0F5.1.1.2 R0F5.1.1.3 R0F5.1.1.4 R0F5.1.2.1 R0F5.1.2.4 R0F5.1.2.4.2 R0F5.1.2.5 R0F5.1.2.7 R0F5.1.2.7.1 R0F5.1.3.1.1 R0F5.1.3.1.2 R0F5.1.3.2 R0F5.1.3.2.2 R0F5.1.4.1.1 R0F5.1.4.2 R0F5.1.4.3 R0F5.1.4.3.1 R0F5.1.4.3.3 R0F5.1.4.3.4 R1F5.1.5.2 R1F5.1.5.2.1 R1F5.1.5.2.3
sgad::servertier::dataaccess::data::userdata::- OwnedResource	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F1.4.5 R0F2

	R0F5 R0F5.1 R0F5.1.1.2 R0F5.1.2.4.1 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.4.1.1 R0F5.1.4.3.2 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R2F5.2.1 R2F5.2.1.4.1
sgad::servtier::dataaccess::data::userdata::Position	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F2 R0F5 R0F5.1 R0F5.1.4.2
sgad::servtier::dataaccess::data::userdata::- UnitInProgress	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F2 R0F5 R0F5.1 R0F5.1.2.4 R0F5.1.2.5 R0F5.1.2.7
sgad::servtier::dataaccess::data::userdata::- UnitPossession	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F1.4.4 R0F2 R0F5 R0F5.1 R0F5.1.2.3.1 R0F5.1.2.7 R0F5.1.4.3.4 R1F5.1.6.3 R2F5.2.1 R2F5.2.1.1.1

	R2F5.2.1.2
sgad::servtier::dataaccess::data::userdata::UserData	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F1.4.1 R0F1.4.2 R0F1.4.3 R0F1.4.4 R0F1.4.5 R0F2 R0F3.1 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1.2 R0F5.1.2.4 R0F5.1.2.4.1 R0F5.1.2.5 R0F5.1.2.7 R0F5.1.3.1.1 R0F5.1.3.1.2 R0F5.1.3.1.3 R0F5.1.3.2.1 R0F5.1.4.3 R0F5.1.4.3.2 R1F5.1.5.2 R1F5.1.5.2.1 R1F5.1.5.2.3 R1F5.1.5.2.3.1 R1F5.1.6.3 R2F5.2.1 R2F5.2.1.1.1.1 R2F5.2.1.2 R2F5.2.1.4.1
sgad::servtier::dataaccess::databaseaccess::- databasemanager:: DataBaseManager	R0F1 R0F1.1.1 R0F1.2.2 R0F1.4 R0F2 R0F5 R0F5.1

sgad::servtier::dataaccess::databaseaccess::- shareddatadao::BonusDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::BuildingWithLevelDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::CostDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::ProducedResourceDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::QuantityResourceDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::ResourceDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::SharedDataDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- shareddatadao::UnitDAO	R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::AuthenticationDataDAO	R0F1 R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::BuildingPossessionDAO	R0F1 R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::OwnedResourceDAO	R0F1 R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::PositionDAO	R0F1 R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::UnitInProgressDAO	R0F1

	R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::UnitPossessionDAO	R0F1 R0F2 R0F5 R0F5.1
sgad::servtier::dataaccess::databaseaccess::- userdatadao::UserDataDAO	R0F1 R0F2 R0F5 R0F5.1
sgad::servtier::presentation::httpresponder::- ResponderActor	R0F1 R0F1.1 R0F1.2 R0F1.3 R0F1.5 R0F2 R0F2.1 R0F2.2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R0P1 R1F5.1.5 R2F5.2.1
sgad::servtier::presentation::httpresponder::- WorkerActor	R0F1 R0F2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2

	R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servtier::presentation::messages::IsAliveRequest	R0F5 R0F5.1
sgad::servtier::presentation::messages::- ToLoginActorRequest	R0F2 R0F5 R0F5.1
sgad::servtier::presentation::messages::- ToPublisherAndUserRequest	R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servtier::presentation::messages::- ToRegistrationActorRequest	R0F1 R0F5 R0F5.1
sgad::servtier::presentation::messages::- ToWorkerLoginRequest	R0F2 R0F5 R0F5.1
sgad::servtier::presentation::messages::- ToWorkerRegistrationRequest	R0F1 R0F5 R0F5.1
sgad::servtier::presentation::messages::- ToWorkerUserRequest	R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4

	R1F5.1.5 R2F5.2.1
sgad::servertier::presentation::pagemanager::PageFactory	R0F1 R0F1.5 R0F2 R0F2.3 R0F5 R0F5.1 R0P1
sgad::servertier::presentation::timeout::STimeout	R0F1 R0F2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servertier::presentation::usermanager::- IsUserActorAliveRequester	R0F2
sgad::servertier::presentation::usermanager::- IsUserActorAliveResponder	R0F2
sgad::servertier::presentation::usermanager::LoginActor	R0F2 R0F5 R0F5.1
sgad::servertier::presentation::usermanager::- PublisherActor	R0F2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1
sgad::servertier::presentation::usermanager::- RegistrationActor	R0F1 R0F5 R0F5.1

sgad::servtier::presentation::usermanager::UserActor	R0F2 R0F4 R0F4.1 R0F4.1.1 R0F4.3 R0F5 R0F5.1 R0F5.1.1 R0F5.1.2 R0F5.1.3 R0F5.1.4 R1F5.1.5 R2F5.2.1 R2F5.2.1.4
spray::HttpRequester	R0F5 R0F5.1

Tabella 2: Tracciamento Classi-Requisiti

6.2 Tracciamento Requisiti-Classi

Requisiti	Classi
R0F1	sgad::servtier::businesslogic::operations::Login sgad::servtier::businesslogic::operations::Operation sgad::servtier::businesslogic::operations::Registration sgad::servtier::dataaccess::data::shareddata::Bonus sgad::servtier::dataaccess::data::shareddata::-BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::Cost sgad::servtier::dataaccess::data::shareddata::-DataFactory sgad::servtier::dataaccess::data::shareddata::-ProducedResource sgad::servtier::dataaccess::data::shareddata::-QuantityResource sgad::servtier::dataaccess::data::shareddata::Resource sgad::servtier::dataaccess::data::shareddata::Unit sgad::servtier::dataaccess::data::userdata::-AuthenticationData sgad::servtier::dataaccess::data::userdata::-BuildingPossession sgad::servtier::dataaccess::data::userdata::-OwnedResource sgad::servtier::dataaccess::data::userdata::Position

	sgad::servertier::dataaccess::data::userdata::- UnitInProgress sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::dataaccess::databaseaccess::- databasemanager::DataBaseManager sgad::servertier::dataaccess::databaseaccess::- userdatadao::AuthenticationDataDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::BuildingPossessionDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::OwnedResourceDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::PositionDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::UnitInProgressDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::UnitPossessionDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::UserDataDAO sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToRegistrationActorRequest sgad::servertier::presentation::messages::- ToWorkerRegistrationRequest sgad::servertier::presentation::pagemanager::PageFactory sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- RegistrationActor
R0F1.1	sgad::servertier::presentation::httpresponder::- ResponderActor
R0F1.1.1	sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::Registration sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::Position sgad::servertier::dataaccess::data::userdata::- UnitInProgress

	sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::dataaccess::databaseaccess::- databasemanager::DataBaseManager
R0F1.2	sgad::servertier::presentation::httpresponder::- ResponderActor
R0F1.2.1	sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::Registration
R0F1.2.2	sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::Registration sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::Position sgad::servertier::dataaccess::data::userdata::- UnitInProgress sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::dataaccess::databaseaccess::- databasemanager::DataBaseManager
R0F1.3	sgad::servertier::presentation::httpresponder::- ResponderActor
R0F1.3.1	sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::Registration
R0F1.4	sgad::servertier::businesslogic::operations::Registration sgad::servertier::dataaccess::data::shareddata::Bonus sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- DataFactory sgad::servertier::dataaccess::data::shareddata::- ProducedResource sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::- BuildingPossession

	sgad::servtier::dataaccess::data::userdata::- OwnedResource sgad::servtier::dataaccess::data::userdata::Position sgad::servtier::dataaccess::data::userdata::- UnitInProgress sgad::servtier::dataaccess::data::userdata::- UnitPossession sgad::servtier::dataaccess::data::userdata::UserData sgad::servtier::dataaccess::databaseaccess::- databasemanager::DataBaseManager
R0F1.4.1	sgad::servtier::businesslogic::operations::Registration sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::userdata::- BuildingPossession sgad::servtier::dataaccess::data::userdata::UserData
R0F1.4.2	sgad::servtier::businesslogic::operations::Registration sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::userdata::- BuildingPossession sgad::servtier::dataaccess::data::userdata::UserData
R0F1.4.3	sgad::servtier::businesslogic::operations::Registration sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::userdata::- BuildingPossession sgad::servtier::dataaccess::data::userdata::UserData
R0F1.4.4	sgad::servtier::businesslogic::operations::Registration sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::shareddata::Unit sgad::servtier::dataaccess::data::userdata::- UnitPossession sgad::servtier::dataaccess::data::userdata::UserData
R0F1.4.5	sgad::servtier::businesslogic::operations::Registration sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::shareddata::Resource

	sgad::servtier::dataaccess::data::userdata::- OwnedResource sgad::servtier::dataaccess::data::userdata::UserData sgad::servtier::businesslogic::operations::Registration sgad::servtier::presentation::httpresponder::- ResponderActor sgad::servtier::presentation::pagemanager::PageFactory
R0F1.5	sgad::servtier::businesslogic::operations::Registration sgad::servtier::businesslogic::operations::Registration sgad::servtier::businesslogic::operations::Operation
R0F1.5.1	sgad::servtier::businesslogic::operations::Registration
R0F1.5.2	sgad::servtier::businesslogic::operations::Registration
R0F1.5.3	sgad::servtier::businesslogic::operations::Operation
R0F2	sgad::servtier::businesslogic::logic::Logic sgad::servtier::businesslogic::operations::Login sgad::servtier::businesslogic::operations::Operation sgad::servtier::dataaccess::data::shareddata::Bonus sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::Cost sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::shareddata::- ProducedResource sgad::servtier::dataaccess::data::shareddata::- QuantityResource sgad::servtier::dataaccess::data::shareddata::Resource sgad::servtier::dataaccess::data::shareddata::Unit sgad::servtier::dataaccess::data::userdata::- AuthenticationData sgad::servtier::dataaccess::data::userdata::- BuildingPossession sgad::servtier::dataaccess::data::userdata::- OwnedResource sgad::servtier::dataaccess::data::userdata::Position sgad::servtier::dataaccess::data::userdata::- UnitInProgress sgad::servtier::dataaccess::data::userdata::- UnitPossession sgad::servtier::dataaccess::data::userdata::UserData sgad::servtier::dataaccess::databaseaccess::- databasemanager:: DataBaseManager sgad::servtier::dataaccess::databaseaccess::- userdatadao:: AuthenticationDataDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao:: BuildingPossessionDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao:: OwnedResourceDAO

	sgad::servertier::dataaccess::databaseaccess::- userdatadao::PositionDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::UnitInProgressDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::UnitPossessionDAO sgad::servertier::dataaccess::databaseaccess::- userdatadao::UserDataDAO sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToLoginActorRequest sgad::servertier::presentation::messages::- ToWorkerLoginRequest sgad::servertier::presentation::pagemanager::PageFactory sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- IsUserActorAliveRequester sgad::servertier::presentation::usermanager::- IsUserActorAliveResponder sgad::servertier::presentation::usermanager::LoginActor sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F2.1	sgad::servertier::presentation::httpresponder::- ResponderActor
R0F2.2	sgad::servertier::presentation::httpresponder::- ResponderActor
R0F2.3	sgad::servertier::presentation::pagemanager::PageFactory
R0F2.3.1	sgad::servertier::businesslogic::operations::Login
R0F2.3.2	sgad::servertier::businesslogic::operations::Login
R0F3	sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget

	sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget
R0F3.1	sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::dataaccess::data::userdata::UserData
R0F3.2	sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget
R0F3.3	sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection

	sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget
R0F4	sgad::clienttier::controller::actions::- ChangePasswordAction sgad::clienttier::controller::actions::- RequestChangePassword sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::- ChangeAccountData sgad::servertier::businesslogic::operations::DeleteAccount sgad::servertier::businesslogic::operations::Login sgad::servertier::businesslogic::operations::Operation sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout

	sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F4.1	sgad::clienttier::controller::actions::- ChangePasswordAction sgad::clienttier::controller::actions::- RequestChangePassword sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::- ChangeAccountData sgad::servertier::businesslogic::operations::Login sgad::servertier::businesslogic::operations::Operation sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F4.1.1	sgad::clienttier::controller::actions::- ChangePasswordAction

	sgad::clienttier::controller::actions::- RequestChangePassword sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::logic::Logic sgad::servertier::businesslogic::operations::- ChangeAccountData sgad::servertier::businesslogic::operations::DeleteAccount sgad::servertier::businesslogic::operations::Operation sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F4.1.1.1	sgad::servertier::businesslogic::operations::- ChangeAccountData
R0F4.1.2	sgad::clienttier::controller::actions::- ChangePasswordAction sgad::clienttier::controller::actions::- RequestChangePassword

	<pre> sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::- ChangeAccountData </pre>
R0F4.1.3	<pre> sgad::clienttier::controller::actions::- ChangePasswordAction sgad::clienttier::controller::actions::- RequestChangePassword sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::clienttier::controller::menufactory::- AccountManagerMenuFactory sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget </pre>
R0F4.2	<pre> sgad::clienttier::controller::actions::- ChangePasswordAction sgad::clienttier::controller::actions::- RequestChangePassword </pre>

	<pre> sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::- ChangeAccountData </pre>
R0F4.2.1	<pre> sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::servertier::businesslogic::operations::- ChangeAccountData </pre>
R0F4.2.2	<pre> sgad::clienttier::controller::actions::- ShowPasswordChangedMenu sgad::servertier::businesslogic::operations::- ChangeAccountData </pre>
R0F4.3	<pre> sgad::clienttier::controller::actions::DeleteAccountAction sgad::clienttier::controller::actions::- RequestDeleteAccount sgad::clienttier::controller::actions::- ShowAccountDeletedMenu sgad::clienttier::model::personaldata::- AuthenticationData sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::logic::Logic </pre>

	sgad::servertier::businesslogic::operations::DeleteAccount sgad::servertier::businesslogic::operations::Operation sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F5	akka::actors::Actor akka::contrib::pattern::DistributedPubSubMediator casbah::DB casbah::DBCollection casbah::DBObject casbah::MongoClient sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- CloseContextualMenu sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::DismissUnits sgad::clienttier::controller::actions::EnableRightClick sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::LoadAnotherUser sgad::clienttier::controller::actions::LoadGeneralData sgad::clienttier::controller::actions::LoadPersonalData sgad::clienttier::controller::actions::Logout sgad::clienttier::controller::actions::ReloadPageAction sgad::clienttier::controller::actions::- RemoveGraphicObjectAction sgad::clienttier::controller::actions::RequestAttackUser sgad::clienttier::controller::actions::- RequestBuildConstruction sgad::clienttier::controller::actions::- RequestDemolishBuilding sgad::clienttier::controller::actions::RequestDismissUnits

```
sgad::clienttier::controller::actions::-  
RequestHarvestResources  
sgad::clienttier::controller::actions::-  
RequestLoadGeneralData  
sgad::clienttier::controller::actions::-  
RequestLoadPersonalData  
sgad::clienttier::controller::actions::RequestTrainUnits  
sgad::clienttier::controller::actions::-  
RequestUpgradeBuilding  
sgad::clienttier::controller::actions::SetAllVoidFilter  
sgad::clienttier::controller::actions::SetBuildModeFilter  
sgad::clienttier::controller::actions::-  
ShowAnotherUserMenu  
sgad::clienttier::controller::actions::ShowAttackMenu  
sgad::clienttier::controller::actions::-  
ShowAttackResultMenu  
sgad::clienttier::controller::actions::-  
ShowBuildConstructionMenu  
sgad::clienttier::controller::actions::-  
ShowBuildingContextualMenu  
sgad::clienttier::controller::actions::ShowDemolishMenu  
sgad::clienttier::controller::actions::-  
ShowDismissUnitMenu  
sgad::clienttier::controller::actions::ShowInteractionMenu  
sgad::clienttier::controller::actions::-  
ShowOperationFailureMenu  
sgad::clienttier::controller::actions::ShowResourceMenu  
sgad::clienttier::controller::actions::-  
ShowTileContextualMenu  
sgad::clienttier::controller::actions::ShowTrainUnitMenu  
sgad::clienttier::controller::actions::-  
ShowUnitSelectionMenu  
sgad::clienttier::controller::actions::ShowUpgradeMenu  
sgad::clienttier::controller::actions::ShowUserListMenu  
sgad::clienttier::controller::actions::StoleResources  
sgad::clienttier::controller::actions::TrainUnit  
sgad::clienttier::controller::actions::UpgradeBuilding  
sgad::clienttier::controller::backupmanager::-  
BackupManager  
sgad::clienttier::controller::menufactory::-  
AnotherUserMenuFactory  
sgad::clienttier::controller::menufactory::-  
AttackResultMenuFactory  
sgad::clienttier::controller::menufactory::-  
BuildingContextualMenuFactory
```

```
sgad::clienttier::controller::menufactory::-
ConfirmMenuFactory
sgad::clienttier::controller::menufactory::-
InteractionMenuFactory
sgad::clienttier::controller::menufactory::MenuFactory
sgad::clienttier::controller::menufactory::-
NotifyMenuFactory
sgad::clienttier::controller::menufactory::-
ResourceMenuFactory
sgad::clienttier::controller::menufactory::-
TileContextualMenuFactory
sgad::clienttier::controller::menufactory::-
UnitSelectionMenu
sgad::clienttier::controller::menufactory::-
UserListMenuFactory
sgad::clienttier::controller::messageinterpreter::-
MessageInterpreter
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-
AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-
UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
```

```
sgad::clienttier::view::graphicobjects::collection::-  
Collection  
sgad::clienttier::view::graphicobjects::collection::-  
GraphicObjectCollection  
sgad::clienttier::view::graphicobjects::collection::-  
GraphicObjectIterator  
sgad::clienttier::view::graphicobjects::collection::Iterator  
sgad::clienttier::view::graphicobjects::components::filter::-  
BuildModeFilter  
sgad::clienttier::view::graphicobjects::components::filter::-  
GraphicFilter  
sgad::clienttier::view::graphicobjects::components::filter::-  
VoidFilter  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponent::BuildingComponent  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponent::TileComponent  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponent::WorldComponent  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::BarracksL1Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::BarracksL2Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::BarracksL3Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::InConstructionShape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::MineL1Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::MineL2Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::MineL3Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::SchoolOfMagicL1Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::SchoolOfMagicL2Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::SchoolOfMagicL3Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::StableL1Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::StableL2Shape  
sgad::clienttier::view::graphicobjects::components::-  
worldcomponentshape::StableL3Shape
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::application::Application
sgad::servertier::businesslogic::logic::Logic
sgad::servertier::businesslogic::operations::-
AttackUserVillage
sgad::servertier::businesslogic::operations::-
BuildConstruction
sgad::servertier::businesslogic::operations::-
DemolishBuilding
sgad::servertier::businesslogic::operations::DismissUnit
sgad::servertier::businesslogic::operations::-
DonateResources
sgad::servertier::businesslogic::operations::DonateUnits
sgad::servertier::businesslogic::operations::-
GetAllServerData
sgad::servertier::businesslogic::operations::-
GetServerData
sgad::servertier::businesslogic::operations::-
HarvestResource
sgad::servertier::businesslogic::operations::-
LoadFinishedBuilding
```

```
sgad::servertier::businesslogic::operations::-
LoadGlobalData
sgad::servertier::businesslogic::operations::LoadVillage
sgad::servertier::businesslogic::operations::LoadUserList
sgad::servertier::businesslogic::operations::Login
sgad::servertier::businesslogic::operations::Logout
sgad::servertier::businesslogic::operations::Operation
sgad::servertier::businesslogic::operations::-
OperationFactory
sgad::servertier::businesslogic::operations::Registration
sgad::servertier::businesslogic::operations::SaveUser
sgad::servertier::businesslogic::operations::StealResources
sgad::servertier::businesslogic::operations::TrainUnit
sgad::servertier::businesslogic::operations::-
UpdateUserData
sgad::servertier::businesslogic::operations::-
UpgradeBuilding
sgad::servertier::dataaccess::data::shareddata::Bonus
sgad::servertier::dataaccess::data::shareddata::-
BuildingWithLevel
sgad::servertier::dataaccess::data::shareddata::Cost
sgad::servertier::dataaccess::data::shareddata::-
DataFactory
sgad::servertier::dataaccess::data::shareddata::-
ProducedResource
sgad::servertier::dataaccess::data::shareddata::-
QuantityResource
sgad::servertier::dataaccess::data::shareddata::Resource
sgad::servertier::dataaccess::data::shareddata::Unit
sgad::servertier::dataaccess::data::userdata::-
AuthenticationData
sgad::servertier::dataaccess::data::userdata::-
BuildingPossession
sgad::servertier::dataaccess::data::userdata::-
OwnedResource
sgad::servertier::dataaccess::data::userdata::Position
sgad::servertier::dataaccess::data::userdata::-
UnitInProgress
sgad::servertier::dataaccess::data::userdata::-
UnitPossession
sgad::servertier::dataaccess::data::userdata::UserData
sgad::servertier::dataaccess::databaseaccess::-
databasemanager:: DataBaseManager
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao:: BonusDAO
```

```
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::BuildingWithLevelDAO  
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::CostDAO  
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::ProducedResourceDAO  
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::QuantityResourceDAO  
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::ResourceDAO  
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::SharedDataDAO  
sgad::servertier::dataaccess::databaseaccess::-  
shareddatadao::UnitDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::AuthenticationDataDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::BuildingPossessionDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::OwnedResourceDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::PositionDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::UnitInProgressDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::UnitPossessionDAO  
sgad::servertier::dataaccess::databaseaccess::-  
userdatadao::UserDataDAO  
sgad::servertier::presentation::httpresponder::-  
ResponderActor  
sgad::servertier::presentation::httpresponder::-  
WorkerActor  
sgad::servertier::presentation::messages::IsAliveRequest  
sgad::servertier::presentation::messages::  
ToLoginActorRequest  
sgad::servertier::presentation::messages::  
ToPublisherAndUserRequest  
sgad::servertier::presentation::messages::  
ToRegistrationActorRequest  
sgad::servertier::presentation::messages::  
ToWorkerLoginRequest  
sgad::servertier::presentation::messages::  
ToWorkerRegistrationRequest  
sgad::servertier::presentation::messages::  
ToWorkerUserRequest  
sgad::servertier::presentation::pagemanager::PageFactory
```

	sgad::servertier::presentation::usermanager::LoginActor sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::- RegistrationActor sgad::servertier::presentation::usermanager::UserActor spray::HttpRequester
R0F5.1	akka::actors::Actor akka::contrib::pattern::DistributedPubSubMediator casbah::DB casbah::DBCollection casbah::DBObject casbah::MongoClient sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- CloseContextualMenu sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::DismissUnits sgad::clienttier::controller::actions::EnableRightClick sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::LoadAnotherUser sgad::clienttier::controller::actions::LoadGeneralData sgad::clienttier::controller::actions::LoadPersonalData sgad::clienttier::controller::actions::Logout sgad::clienttier::controller::actions::ReloadPageAction sgad::clienttier::controller::actions::- RemoveGraphicObjectAction sgad::clienttier::controller::actions::RequestAttackUser sgad::clienttier::controller::actions::- RequestBuildConstruction sgad::clienttier::controller::actions::- RequestDemolishBuilding sgad::clienttier::controller::actions::RequestDismissUnits sgad::clienttier::controller::actions::- RequestHarvestResources sgad::clienttier::controller::actions::- RequestLoadGeneralData sgad::clienttier::controller::actions::- RequestLoadPersonalData sgad::clienttier::controller::actions::RequestTrainUnits sgad::clienttier::controller::actions::- RequestUpgradeBuilding sgad::clienttier::controller::actions::SetAllVoidFilter

```
sgad::clienttier::controller::actions::SetBuildModeFilter
sgad::clienttier::controller::actions::-
ShowAnotherUserMenu
sgad::clienttier::controller::actions::ShowAttackMenu
sgad::clienttier::controller::actions::-
ShowAttackResultMenu
sgad::clienttier::controller::actions::-
ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-
ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-
ShowDismissUnitMenu
sgad::clienttier::controller::actions::ShowInteractionMenu
sgad::clienttier::controller::actions::-
ShowOperationFailureMenu
sgad::clienttier::controller::actions::ShowResourceMenu
sgad::clienttier::controller::actions::-
ShowTileContextualMenu
sgad::clienttier::controller::actions::ShowTrainUnitMenu
sgad::clienttier::controller::actions::-
ShowUnitSelectionMenu
sgad::clienttier::controller::actions::ShowUpgradeMenu
sgad::clienttier::controller::actions::ShowUserListMenu
sgad::clienttier::controller::actions::StoleResources
sgad::clienttier::controller::actions::TrainUnit
sgad::clienttier::controller::actions::UpgradeBuilding
sgad::clienttier::controller::backupmanager::-
BackupManager
sgad::clienttier::controller::menufactory::-
AnotherUserMenuFactory
sgad::clienttier::controller::menufactory::-
AttackResultMenuFactory
sgad::clienttier::controller::menufactory::-
BuildingContextualMenuFactory
sgad::clienttier::controller::menufactory::-
ConfirmMenuFactory
sgad::clienttier::controller::menufactory::-
InteractionMenuFactory
sgad::clienttier::controller::menufactory::MenuFactory
sgad::clienttier::controller::menufactory::-
NotifyMenuFactory
sgad::clienttier::controller::menufactory::-
ResourceMenuFactory
sgad::clienttier::controller::menufactory::-
TileContextualMenuFactory
```

```
sgad::clienttier::controller::menufactory::-
UnitSelectionMenu
sgad::clienttier::controller::menufactory::-
UserListMenuFactory
sgad::clienttier::controller::messageinterpreter::-
MessageInterpreter
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-
AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-
UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-
Collection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-
BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-
GraphicFilter
```

```
sgad::clienttier::view::graphicobjects::components::filter::-
VoidFilter
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
```

```
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::application::Application
sgad::servertier::businesslogic::logic::Logic
sgad::servertier::businesslogic::operations::-
AttackUserVillage
sgad::servertier::businesslogic::operations::-
BuildConstruction
sgad::servertier::businesslogic::operations::-
DemolishBuilding
sgad::servertier::businesslogic::operations::DismissUnit
sgad::servertier::businesslogic::operations::-
DonateResources
sgad::servertier::businesslogic::operations::DonateUnits
sgad::servertier::businesslogic::operations::-
GetAllServerData
sgad::servertier::businesslogic::operations::-
GetServerData
sgad::servertier::businesslogic::operations::-
HarvestResource
sgad::servertier::businesslogic::operations::-
LoadFinishedBuilding
sgad::servertier::businesslogic::operations::-
LoadGlobalData
sgad::servertier::businesslogic::operations::LoadVillage
sgad::servertier::businesslogic::operations::LoadUserList
sgad::servertier::businesslogic::operations::Login
sgad::servertier::businesslogic::operations::Logout
sgad::servertier::businesslogic::operations::Operation
sgad::servertier::businesslogic::operations::-
OperationFactory
sgad::servertier::businesslogic::operations::Registration
sgad::servertier::businesslogic::operations::SaveUser
sgad::servertier::businesslogic::operations::StealResources
sgad::servertier::businesslogic::operations::TrainUnit
```

```
sgad::servertier::businesslogic::operations::-
UpdateUserData
sgad::servertier::businesslogic::operations::-
UpgradeBuilding
sgad::servertier::dataaccess::data::shareddata::Bonus
sgad::servertier::dataaccess::data::shareddata::-
BuildingWithLevel
sgad::servertier::dataaccess::data::shareddata::Cost
sgad::servertier::dataaccess::data::shareddata::-
DataFactory
sgad::servertier::dataaccess::data::shareddata::-
ProducedResource
sgad::servertier::dataaccess::data::shareddata::-
QuantityResource
sgad::servertier::dataaccess::data::shareddata::Resource
sgad::servertier::dataaccess::data::shareddata::Unit
sgad::servertier::dataaccess::data::userdata::-
AuthenticationData
sgad::servertier::dataaccess::data::userdata::-
BuildingPossession
sgad::servertier::dataaccess::data::userdata::-
OwnedResource
sgad::servertier::dataaccess::data::userdata::Position
sgad::servertier::dataaccess::data::userdata::-
UnitInProgress
sgad::servertier::dataaccess::data::userdata::-
UnitPossession
sgad::servertier::dataaccess::data::userdata::UserData
sgad::servertier::dataaccess::databaseaccess::-
databasemanager::DataBaseManager
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::BonusDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::BuildingWithLevelDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::CostDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::ProducedResourceDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::QuantityResourceDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::ResourceDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::SharedDataDAO
sgad::servertier::dataaccess::databaseaccess::-
shareddatadao::UnitDAO
```

	sgad::servtier::dataaccess::databaseaccess::- userdatadao::AuthenticationDataDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao::BuildingPossessionDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao::OwnedResourceDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao::PositionDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao::UnitInProgressDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao::UnitPossessionDAO sgad::servtier::dataaccess::databaseaccess::- userdatadao::UserDataDAO sgad::servtier::presentation::httpresponder::- ResponderActor sgad::servtier::presentation::httpresponder::- WorkerActor sgad::servtier::presentation::messages::IsAliveRequest sgad::servtier::presentation::messages:: ToLoginActorRequest sgad::servtier::presentation::messages:: ToPublisherAndUserRequest sgad::servtier::presentation::messages:: ToRegistrationActorRequest sgad::servtier::presentation::messages:: ToWorkerLoginRequest sgad::servtier::presentation::messages:: ToWorkerRegistrationRequest sgad::servtier::presentation::messages:: ToWorkerUserRequest sgad::servtier::presentation::pagemanager::PageFactory sgad::servtier::presentation::usermanager::LoginActor sgad::servtier::presentation::usermanager::- PublisherActor sgad::servtier::presentation::usermanager::- RegistrationActor sgad::servtier::presentation::usermanager::UserActor spray::HttpRequester
R0F5.1.1	sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- CloseContextualMenu sgad::clienttier::controller::actions::DemolishBuilding

```
sgad::clienttier::controller::actions::DismissUnits
sgad::clienttier::controller::actions::EnableRightClick
sgad::clienttier::controller::actions::HarvestResources
sgad::clienttier::controller::actions::LoadAnotherUser
sgad::clienttier::controller::actions::LoadGeneralData
sgad::clienttier::controller::actions::LoadPersonalData
sgad::clienttier::controller::actions::Logout
sgad::clienttier::controller::actions::ReloadPageAction
sgad::clienttier::controller::actions::-
RemoveGraphicObjectAction
sgad::clienttier::controller::actions::RequestAttackUser
sgad::clienttier::controller::actions::-
RequestBuildConstruction
sgad::clienttier::controller::actions::-
RequestDemolishBuilding
sgad::clienttier::controller::actions::RequestDismissUnits
sgad::clienttier::controller::actions::-
RequestHarvestResources
sgad::clienttier::controller::actions::-
RequestLoadGeneralData
sgad::clienttier::controller::actions::-
RequestLoadPersonalData
sgad::clienttier::controller::actions::RequestTrainUnits
sgad::clienttier::controller::actions::-
RequestUpgradeBuilding
sgad::clienttier::controller::actions::SetAllVoidFilter
sgad::clienttier::controller::actions::SetBuildModeFilter
sgad::clienttier::controller::actions::-
ShowAnotherUserMenu
sgad::clienttier::controller::actions::ShowAttackMenu
sgad::clienttier::controller::actions::-
ShowAttackResultMenu
sgad::clienttier::controller::actions::-
ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-
ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-
ShowDismissUnitMenu
sgad::clienttier::controller::actions::ShowInteractionMenu
sgad::clienttier::controller::actions::-
ShowOperationFailureMenu
sgad::clienttier::controller::actions::ShowResourceMenu
sgad::clienttier::controller::actions::-
ShowTileContextualMenu
sgad::clienttier::controller::actions::ShowTrainUnitMenu
```

```
sgad::clienttier::controller::actions::-  
ShowUnitSelectionMenu  
sgad::clienttier::controller::actions::-ShowUpgradeMenu  
sgad::clienttier::controller::actions::-ShowUserListMenu  
sgad::clienttier::controller::actions::-StoleResources  
sgad::clienttier::controller::actions::-TrainUnit  
sgad::clienttier::controller::actions::-UpgradeBuilding  
sgad::clienttier::controller::backupmanager::-  
BackupManager  
sgad::clienttier::controller::menufactory::-  
AnotherUserMenuFactory  
sgad::clienttier::controller::menufactory::-  
AttackResultMenuFactory  
sgad::clienttier::controller::menufactory::-  
BuildingContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
ConfirmMenuFactory  
sgad::clienttier::controller::menufactory::-  
InteractionMenuFactory  
sgad::clienttier::controller::menufactory::MenuFactory  
sgad::clienttier::controller::menufactory::-  
NotifyMenuFactory  
sgad::clienttier::controller::menufactory::-  
ResourceMenuFactory  
sgad::clienttier::controller::menufactory::-  
TileContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
UnitSelectionMenu  
sgad::clienttier::controller::menufactory::-  
UserListMenuFactory  
sgad::clienttier::controller::messageinterpreter::-  
MessageInterpreter  
sgad::clienttier::controller::requester::AJAXRequester  
sgad::clienttier::model::generaldata::Bonus  
sgad::clienttier::model::generaldata::BuildingWithLevel  
sgad::clienttier::model::generaldata::Cost  
sgad::clienttier::model::generaldata::DataFactory  
sgad::clienttier::model::generaldata::ProducedResource  
sgad::clienttier::model::generaldata::QuantityResource  
sgad::clienttier::model::generaldata::Resource  
sgad::clienttier::model::generaldata::Unit  
sgad::clienttier::model::observer::Observable  
sgad::clienttier::model::observer::Observer  
sgad::clienttier::model::personaldata::-  
AuthenticationData  
sgad::clienttier::model::personaldata::BuildingPossession
```

```
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-
UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-
Collection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-
BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-
GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-
VoidFilter
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::businesslogic::logic::Logic
sgad::servertier::businesslogic::operations::-
HarvestResource
sgad::servertier::businesslogic::operations::Operation
sgad::servertier::dataaccess::data::userdata::-
AuthenticationData
```

	<pre> sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor </pre>
R0F5.1.1.1	<pre> sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::- GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::- worldcomponent::BuildingComponent sgad::servertier::businesslogic::operations::- HarvestResource </pre>
R0F5.1.1.2	<pre> sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::- RequestHarvestResources sgad::clienttier::controller::messageinterpreter::- MessageInterpreter sgad::clienttier::controller::requester::AJAXRequester sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Resource sgad::clienttier::model::personaldata::BuildingPossession sgad::clienttier::model::personaldata::OwnedResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::UserData </pre>
R0F5.1.1.3	<pre> sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::ProducedResource </pre>

	sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::businesslogic::operations::- HarvestResource sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::- ProductedResource sgad::servertier::dataaccess::data::userdata::- BuildingPossession
R0F5.1.1.4	sgad::clienttier::model::generaldata::ProducedResource sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::dataaccess::data::shareddata::- ProductedResource sgad::servertier::dataaccess::data::userdata::- BuildingPossession
R0F5.1.1.5	sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::ProducedResource sgad::clienttier::model::generaldata::Resource sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::- ProductedResource sgad::servertier::dataaccess::data::shareddata::Resource
R0F5.1.1.6	sgad::clienttier::controller::actions::HarvestResources sgad::servertier::businesslogic::operations::- HarvestResource
R0F5.1.2	sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- CloseContextMenu sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::DismissUnits sgad::clienttier::controller::actions::EnableRightClick sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::LoadAnotherUser sgad::clienttier::controller::actions::LoadGeneralData sgad::clienttier::controller::actions::LoadPersonalData sgad::clienttier::controller::actions::Logout sgad::clienttier::controller::actions::ReloadPageAction sgad::clienttier::controller::actions::- RemoveGraphicObjectAction sgad::clienttier::controller::actions::RequestAttackUser sgad::clienttier::controller::actions::- RequestBuildConstruction

```
sgad::clienttier::controller::actions::-  
RequestDemolishBuilding  
sgad::clienttier::controller::actions::-RequestDismissUnits  
sgad::clienttier::controller::actions::-  
RequestHarvestResources  
sgad::clienttier::controller::actions::-  
RequestLoadGeneralData  
sgad::clienttier::controller::actions::-  
RequestLoadPersonalData  
sgad::clienttier::controller::actions::-RequestTrainUnits  
sgad::clienttier::controller::actions::-  
RequestUpgradeBuilding  
sgad::clienttier::controller::actions::-SetAllVoidFilter  
sgad::clienttier::controller::actions::-SetBuildModeFilter  
sgad::clienttier::controller::actions::-  
ShowAnotherUserMenu  
sgad::clienttier::controller::actions::-ShowAttackMenu  
sgad::clienttier::controller::actions::-  
ShowAttackResultMenu  
sgad::clienttier::controller::actions::-  
ShowBuildConstructionMenu  
sgad::clienttier::controller::actions::-  
ShowBuildingContextualMenu  
sgad::clienttier::controller::actions::-ShowDemolishMenu  
sgad::clienttier::controller::actions::-  
ShowDismissUnitMenu  
sgad::clienttier::controller::actions::-ShowInteractionMenu  
sgad::clienttier::controller::actions::-  
ShowOperationFailureMenu  
sgad::clienttier::controller::actions::-ShowResourceMenu  
sgad::clienttier::controller::actions::-  
ShowTileContextualMenu  
sgad::clienttier::controller::actions::-ShowTrainUnitMenu  
sgad::clienttier::controller::actions::-  
ShowUnitSelectionMenu  
sgad::clienttier::controller::actions::-ShowUpgradeMenu  
sgad::clienttier::controller::actions::-ShowUserListMenu  
sgad::clienttier::controller::actions::-StoleResources  
sgad::clienttier::controller::actions::-TrainUnit  
sgad::clienttier::controller::actions::-UpgradeBuilding  
sgad::clienttier::controller::backupmanager::-  
BackupManager  
sgad::clienttier::controller::menufactory::-  
AnotherUserMenuFactory  
sgad::clienttier::controller::menufactory::-  
AttackResultMenuFactory
```

```
sgad::clienttier::controller::menufactory::-
BuildingContextualMenuFactory
sgad::clienttier::controller::menufactory::-
ConfirmMenuFactory
sgad::clienttier::controller::menufactory::-
InteractionMenuFactory
sgad::clienttier::controller::menufactory::MenuFactory
sgad::clienttier::controller::menufactory::-
NotifyMenuFactory
sgad::clienttier::controller::menufactory::-
ResourceMenuFactory
sgad::clienttier::controller::menufactory::-
TileContextualMenuFactory
sgad::clienttier::controller::menufactory::-
UnitSelectionMenu
sgad::clienttier::controller::menufactory::-
UserListMenuFactory
sgad::clienttier::controller::messageinterpreter::-
MessageInterpreter
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-
AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-
UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
```

```
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-
Collection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-
BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-
GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-
VoidFilter
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
```

	sgad::clienttier::view::graphicobjects::components::- worldcomponentshape::TileShape sgad::clienttier::view::graphicobjects::components::- worldcomponentshape::WizardTowerL1Shape sgad::clienttier::view::graphicobjects::components::- worldcomponentshape::WizardTowerL2Shape sgad::clienttier::view::graphicobjects::components::- worldcomponentshape::WizardTowerL3Shape sgad::clienttier::view::graphicobjects::components::- worldcomponentshape::WorldComponentShapeFactory sgad::clienttier::view::graphicobjects::components::- worldcomponentshape::WorldComponentShapeImg sgad::clienttier::view::graphicobjects::graphicobject::- GraphicObject sgad::clienttier::view::graphicobjects::point::Point2D sgad::clienttier::view::graphicobjects::shape::Shape sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::clienttier::view::graphicobjects::widget::Widget sgad::servertier::businesslogic::operations::Login sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F5.1.2.1	sgad::clienttier::controller::actions::- ShowBuildingContextualMenu sgad::clienttier::controller::menufactory::MenuFactory sgad::clienttier::view::commands::Click

	sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::-worldcomponent::BuildingComponent sgad::clienttier::view::graphicobjects::widget::-ButtonWidget sgad::clienttier::view::graphicobjects::widget::-FrameWidget sgad::clienttier::view::graphicobjects::widget::-ImageWidget sgad::clienttier::view::graphicobjects::widget::-TextWidget sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::shareddata::-BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::-BuildingPossession
R0F5.1.2.1.1	sgad::clienttier::model::generaldata::BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::-BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Unit
R0F5.1.2.2	sgad::clienttier::controller::actions::-ShowUnitSelectionMenu sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator sgad::clienttier::view::graphicobjects::widget::-ButtonWidget sgad::clienttier::view::graphicobjects::widget::-FrameWidget sgad::clienttier::view::graphicobjects::widget::-ImageWidget sgad::clienttier::view::graphicobjects::widget::-TextWidget sgad::servertier::businesslogic::operations::TrainUnit
R0F5.1.2.2.1	sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Unit sgad::servertier::dataaccess::data::shareddata::-BuildingWithLevel

	sgad::servertier::dataaccess::data::shareddata::Unit
R0F5.1.2.3	sgad::clienttier::controller::actions::ShowTrainUnitMenu sgad::servertier::businesslogic::operations::TrainUnit
R0F5.1.2.3.1	sgad::clienttier::controller::actions::ShowTrainUnitMenu sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::dataaccess::data::userdata::- UnitPossession
R0F5.1.2.4	sgad::clienttier::controller::actions::RequestTrainUnits sgad::clienttier::controller::actions::TrainUnit sgad::clienttier::controller::messageinterpreter::- MessageInterpreter sgad::clienttier::controller::requester::AJAXRequester sgad::clienttier::model::personaldata::UnitInProgress sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- UnitInProgress sgad::servertier::dataaccess::data::userdata::UserData
R0F5.1.2.4.1	sgad::clienttier::controller::actions::ShowTrainUnitMenu sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::generaldata::Resource sgad::clienttier::model::generaldata::Unit sgad::clienttier::model::personaldata::OwnedResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::UserData
R0F5.1.2.4.2	sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::dataaccess::data::userdata::- BuildingPossession

R0F5.1.2.4.3	sgad::clienttier::model::generaldata::Bonus sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::Unit sgad::clienttier::model::personaldata::BuildingPossession sgad::clienttier::model::personaldata::UnitInProgress
R0F5.1.2.5	sgad::clienttier::controller::actions::TrainUnit sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::personaldata::BuildingPossession sgad::clienttier::model::personaldata::UnitInProgress sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- UnitInProgress sgad::servertier::dataaccess::data::userdata::UserData
R0F5.1.2.6	sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::generaldata::Resource sgad::clienttier::model::generaldata::Unit sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::shareddata::Unit
R0F5.1.2.7	sgad::clienttier::controller::menufactory::- UnitSelectionMenu sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Unit sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- UnitInProgress sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData
R0F5.1.2.7.1	sgad::clienttier::controller::actions::TrainUnit sgad::clienttier::model::personaldata::BuildingPossession

	sgad::servertier::businesslogic::operations::TrainUnit sgad::servertier::dataaccess::data::userdata::-BuildingPossession
R0F5.1.2.7.2	sgad::clienttier::model::generaldata::BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::-BuildingWithLevel
R0F5.1.3	sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::-CloseContextualMenu sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::DismissUnits sgad::clienttier::controller::actions::EnableRightClick sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::LoadAnotherUser sgad::clienttier::controller::actions::LoadGeneralData sgad::clienttier::controller::actions::LoadPersonalData sgad::clienttier::controller::actions::Logout sgad::clienttier::controller::actions::ReloadPageAction sgad::clienttier::controller::actions::-RemoveGraphicObjectAction sgad::clienttier::controller::actions::RequestAttackUser sgad::clienttier::controller::actions::-RequestBuildConstruction sgad::clienttier::controller::actions::-RequestDemolishBuilding sgad::clienttier::controller::actions::RequestDismissUnits sgad::clienttier::controller::actions::-RequestHarvestResources sgad::clienttier::controller::actions::-RequestLoadGeneralData sgad::clienttier::controller::actions::-RequestLoadPersonalData sgad::clienttier::controller::actions::RequestTrainUnits sgad::clienttier::controller::actions::-RequestUpgradeBuilding sgad::clienttier::controller::actions::SetAllVoidFilter sgad::clienttier::controller::actions::SetBuildModeFilter sgad::clienttier::controller::actions::-ShowAnotherUserMenu sgad::clienttier::controller::actions::ShowAttackMenu sgad::clienttier::controller::actions::-ShowAttackResultMenu

```
sgad::clienttier::controller::actions::-
ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-
ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-
ShowDismissUnitMenu
sgad::clienttier::controller::actions::ShowInteractionMenu
sgad::clienttier::controller::actions::-
ShowOperationFailureMenu
sgad::clienttier::controller::actions::ShowResourceMenu
sgad::clienttier::controller::actions::-
ShowTileContextualMenu
sgad::clienttier::controller::actions::ShowTrainUnitMenu
sgad::clienttier::controller::actions::-
ShowUnitSelectionMenu
sgad::clienttier::controller::actions::ShowUpgradeMenu
sgad::clienttier::controller::actions::ShowUserListMenu
sgad::clienttier::controller::actions::StoleResources
sgad::clienttier::controller::actions::TrainUnit
sgad::clienttier::controller::actions::UpgradeBuilding
sgad::clienttier::controller::backupmanager::-
BackupManager
sgad::clienttier::controller::menufactory::-
AnotherUserMenuFactory
sgad::clienttier::controller::menufactory::-
AttackResultMenuFactory
sgad::clienttier::controller::menufactory::-
BuildingContextualMenuFactory
sgad::clienttier::controller::menufactory::-
ConfirmMenuFactory
sgad::clienttier::controller::menufactory::-
InteractionMenuFactory
sgad::clienttier::controller::menufactory::MenuFactory
sgad::clienttier::controller::menufactory::-
NotifyMenuFactory
sgad::clienttier::controller::menufactory::-
ResourceMenuFactory
sgad::clienttier::controller::menufactory::-
TileContextualMenuFactory
sgad::clienttier::controller::menufactory::-
UnitSelectionMenu
sgad::clienttier::controller::menufactory::-
UserListMenuFactory
sgad::clienttier::controller::messageinterpreter::-
MessageInterpreter
```

```
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-Collection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-VoidFilter
sgad::clienttier::view::graphicobjects::components::-worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::TileComponent
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
```

	sgad::clienttier::view::graphicobjects::widget::-FrameWidget sgad::clienttier::view::graphicobjects::widget::-ImageWidget sgad::clienttier::view::graphicobjects::widget::-TextWidget sgad::clienttier::view::graphicobjects::widget::Widget sgad::servertier::businesslogic::logic::Logic sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::-UpgradeBuilding sgad::servertier::dataaccess::data::userdata::-AuthenticationData sgad::servertier::presentation::httpresponder::-ResponderActor sgad::servertier::presentation::httpresponder::-WorkerActor sgad::servertier::presentation::messages::-ToPublisherAndUserRequest sgad::servertier::presentation::messages::-ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::-PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F5.1.3.1	sgad::clienttier::controller::actions::-ShowBuildingContextualMenu sgad::clienttier::controller::menufactory::-BuildingContextualMenuFactory sgad::clienttier::view::commands::RightClick sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::-worldcomponent::WorldComponent sgad::clienttier::view::graphicobjects::widget::-ButtonWidget sgad::clienttier::view::graphicobjects::widget::-FrameWidget sgad::clienttier::view::graphicobjects::widget::-ImageWidget sgad::clienttier::view::graphicobjects::widget::-TextWidget

	sgad::servtier::businesslogic::operations::- UpgradeBuilding
R0F5.1.3.1.1	sgad::clienttier::controller::menufactory::- BuildingContextualMenuFactory sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::personaldata::BuildingPossession sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::- DataFactory sgad::servtier::dataaccess::data::userdata::- BuildingPossession sgad::servtier::dataaccess::data::userdata::UserData
R0F5.1.3.1.2	sgad::servtier::businesslogic::operations::- UpgradeBuilding sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::userdata::- BuildingPossession sgad::servtier::dataaccess::data::userdata::UserData
R0F5.1.3.1.3	sgad::clienttier::controller::menufactory::MenuFactory sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::personaldata::BuildingPossession sgad::clienttier::model::personaldata::OwnedResource sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servtier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servtier::dataaccess::data::shareddata::Cost sgad::servtier::dataaccess::data::shareddata::- QuantityResource sgad::servtier::dataaccess::data::shareddata::Resource sgad::servtier::dataaccess::data::userdata::- OwnedResource sgad::servtier::dataaccess::data::userdata::UserData
R0F5.1.3.2	sgad::clienttier::controller::actions::UpgradeBuilding

	<pre> sgad::clienttier::controller::messageinterpreter::- MessageInterpreter sgad::clienttier::controller::requester::AJAXRequester sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::userdata::- BuildingPossession </pre>
R0F5.1.3.2.1	<pre> sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::UpgradeBuilding sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::UserData </pre>
R0F5.1.3.2.2	<pre> sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::userdata::- BuildingPossession </pre>
R0F5.1.4	<pre> sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- CloseContextualMenu sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::DismissUnits sgad::clienttier::controller::actions::EnableRightClick sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::LoadAnotherUser sgad::clienttier::controller::actions::LoadGeneralData </pre>

```
sgad::clienttier::controller::actions::LoadPersonalData
sgad::clienttier::controller::actions::Logout
sgad::clienttier::controller::actions::ReloadPageAction
sgad::clienttier::controller::actions::-RemoveGraphicObjectAction
sgad::clienttier::controller::actions::RequestAttackUser
sgad::clienttier::controller::actions::-RequestBuildConstruction
sgad::clienttier::controller::actions::-RequestDemolishBuilding
sgad::clienttier::controller::actions::RequestDismissUnits
sgad::clienttier::controller::actions::-RequestHarvestResources
sgad::clienttier::controller::actions::-RequestLoadGeneralData
sgad::clienttier::controller::actions::-RequestLoadPersonalData
sgad::clienttier::controller::actions::RequestTrainUnits
sgad::clienttier::controller::actions::-RequestUpgradeBuilding
sgad::clienttier::controller::actions::SetAllVoidFilter
sgad::clienttier::controller::actions::SetBuildModeFilter
sgad::clienttier::controller::actions::-ShowAnotherUserMenu
sgad::clienttier::controller::actions::ShowAttackMenu
sgad::clienttier::controller::actions::-ShowAttackResultMenu
sgad::clienttier::controller::actions::-ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-ShowDismissUnitMenu
sgad::clienttier::controller::actions::ShowInteractionMenu
sgad::clienttier::controller::actions::-ShowOperationFailureMenu
sgad::clienttier::controller::actions::ShowResourceMenu
sgad::clienttier::controller::actions::-ShowTileContextualMenu
sgad::clienttier::controller::actions::ShowTrainUnitMenu
sgad::clienttier::controller::actions::-ShowUnitSelectionMenu
sgad::clienttier::controller::actions::ShowUpgradeMenu
sgad::clienttier::controller::actions::ShowUserListMenu
sgad::clienttier::controller::actions::StoleResources
```

```
sgad::clienttier::controller::actions::TrainUnit
sgad::clienttier::controller::actions::UpgradeBuilding
sgad::clienttier::controller::backupmanager::-
BackupManager
sgad::clienttier::controller::menufactory::-
AnotherUserMenuFactory
sgad::clienttier::controller::menufactory::-
AttackResultMenuFactory
sgad::clienttier::controller::menufactory::-
BuildingContextualMenuFactory
sgad::clienttier::controller::menufactory::-
ConfirmMenuFactory
sgad::clienttier::controller::menufactory::-
InteractionMenuFactory
sgad::clienttier::controller::menufactory::MenuFactory
sgad::clienttier::controller::menufactory::-
NotifyMenuFactory
sgad::clienttier::controller::menufactory::-
ResourceMenuFactory
sgad::clienttier::controller::menufactory::-
TileContextualMenuFactory
sgad::clienttier::controller::menufactory::-
UnitSelectionMenu
sgad::clienttier::controller::menufactory::-
UserListMenuFactory
sgad::clienttier::controller::messageinterpreter::-
MessageInterpreter
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-
AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
```

```
sgad::clienttier::model::userdatamanager::-
UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-
Collection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-
BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-
GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-
VoidFilter
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::businesslogic::logic::Logic
sgad::servertier::businesslogic::operations::-
BuildConstruction
sgad::servertier::businesslogic::operations::Operation
sgad::servertier::dataaccess::data::userdata::-
AuthenticationData
sgad::servertier::presentation::httpresponder::-
ResponderActor
sgad::servertier::presentation::httpresponder::-
WorkerActor
sgad::servertier::presentation::messages::-
ToPublisherAndUserRequest
```

	sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor
R0F5.1.4.1	sgad::clienttier::controller::actions::- ShowTileContextualMenu sgad::clienttier::controller::menufactory::- TileContextualMenuFactory sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::DataFactory sgad::clienttier::view::commands::Click sgad::clienttier::view::commands::RightClick sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::- GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::- worldcomponent::TileComponent sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::- BuildConstruction
R0F5.1.4.1.1	sgad::clienttier::controller::menufactory::- TileContextualMenuFactory sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::DataFactory sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::generaldata::Resource sgad::clienttier::model::personaldata::OwnedResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::servertier::businesslogic::operations::- BuildConstruction

	<pre> sgad::servertier::businesslogic::operations::- UpdateUserData sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- OwnedResource </pre>
R0F5.1.4.1.2	<pre> sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::DataFactory sgad::servertier::dataaccess::data::shareddata::- DataFactory </pre>
R0F5.1.4.2	<pre> sgad::clienttier::view::commands::RightClick sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::components::- worldcomponent::TileComponent sgad::servertier::businesslogic::operations::- BuildConstruction sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::Position </pre>
R0F5.1.4.2.1	<pre> sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::components::- worldcomponent::TileComponent sgad::servertier::businesslogic::operations::- BuildConstruction </pre>
R0F5.1.4.3	<pre> sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- RequestBuildConstruction sgad::clienttier::controller::messageinterpreter::- MessageInterpreter sgad::clienttier::controller::requester::AJAXRequester sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::businesslogic::operations::- BuildConstruction sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::UserData </pre>
R0F5.1.4.3.1	<pre> sgad::clienttier::model::generaldata::ProducedResource sgad::clienttier::model::generaldata::Resource sgad::clienttier::model::personaldata::BuildingPossession </pre>

	sgad::servertier::businesslogic::operations::- HarvestResource sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::- ProductedResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::userdata::- BuildingPossession
R0F5.1.4.3.2	sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::- BuildConstruction sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::UserData
R0F5.1.4.3.3	sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::personaldata::BuildingPossession sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::userdata::- BuildingPossession
R0F5.1.4.3.4	sgad::servertier::businesslogic::operations::- BuildConstruction sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- UnitPossession
R0P1	sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::pagemanager::PageFactory
R1F5.1.5	sgad::clienttier::controller::actions::Action

```
sgad::clienttier::controller::actions::ActionListener
sgad::clienttier::controller::actions::BackToVillage
sgad::clienttier::controller::actions::BuildConstruction
sgad::clienttier::controller::actions::-CloseContextualMenu
sgad::clienttier::controller::actions::DemolishBuilding
sgad::clienttier::controller::actions::DismissUnits
sgad::clienttier::controller::actions::EnableRightClick
sgad::clienttier::controller::actions::HarvestResources
sgad::clienttier::controller::actions::LoadAnotherUser
sgad::clienttier::controller::actions::LoadGeneralData
sgad::clienttier::controller::actions::LoadPersonalData
sgad::clienttier::controller::actions::Logout
sgad::clienttier::controller::actions::ReloadPageAction
sgad::clienttier::controller::actions::-RemoveGraphicObjectAction
sgad::clienttier::controller::actions::RequestAttackUser
sgad::clienttier::controller::actions::-RequestBuildConstruction
sgad::clienttier::controller::actions::-RequestDemolishBuilding
sgad::clienttier::controller::actions::RequestDismissUnits
sgad::clienttier::controller::actions::-RequestHarvestResources
sgad::clienttier::controller::actions::-RequestLoadGeneralData
sgad::clienttier::controller::actions::-RequestLoadPersonalData
sgad::clienttier::controller::actions::RequestTrainUnits
sgad::clienttier::controller::actions::-RequestUpgradeBuilding
sgad::clienttier::controller::actions::SetAllVoidFilter
sgad::clienttier::controller::actions::SetBuildModeFilter
sgad::clienttier::controller::actions::-ShowAnotherUserMenu
sgad::clienttier::controller::actions::ShowAttackMenu
sgad::clienttier::controller::actions::-ShowAttackResultMenu
sgad::clienttier::controller::actions::-ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-ShowDismissUnitMenu
sgad::clienttier::controller::actions::ShowInteractionMenu
```

```
sgad::clienttier::controller::actions::-  
ShowOperationFailureMenu  
sgad::clienttier::controller::actions::-ShowResourceMenu  
sgad::clienttier::controller::actions::-  
ShowTileContextualMenu  
sgad::clienttier::controller::actions::-ShowTrainUnitMenu  
sgad::clienttier::controller::actions::-  
ShowUnitSelectionMenu  
sgad::clienttier::controller::actions::-ShowUpgradeMenu  
sgad::clienttier::controller::actions::-ShowUserListMenu  
sgad::clienttier::controller::actions::StoleResources  
sgad::clienttier::controller::actions::TrainUnit  
sgad::clienttier::controller::actions::UpgradeBuilding  
sgad::clienttier::controller::backupmanager::-  
BackupManager  
sgad::clienttier::controller::menufactory::-  
AnotherUserMenuFactory  
sgad::clienttier::controller::menufactory::-  
AttackResultMenuFactory  
sgad::clienttier::controller::menufactory::-  
BuildingContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
ConfirmMenuFactory  
sgad::clienttier::controller::menufactory::-  
InteractionMenuFactory  
sgad::clienttier::controller::menufactory::MenuFactory  
sgad::clienttier::controller::menufactory::-  
NotifyMenuFactory  
sgad::clienttier::controller::menufactory::-  
ResourceMenuFactory  
sgad::clienttier::controller::menufactory::-  
TileContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
UnitSelectionMenu  
sgad::clienttier::controller::menufactory::-  
UserListMenuFactory  
sgad::clienttier::controller::messageinterpreter::-  
MessageInterpreter  
sgad::clienttier::controller::requester::AJAXRequester  
sgad::clienttier::model::generaldata::Bonus  
sgad::clienttier::model::generaldata::BuildingWithLevel  
sgad::clienttier::model::generaldata::Cost  
sgad::clienttier::model::generaldata::DataFactory  
sgad::clienttier::model::generaldata::ProducedResource  
sgad::clienttier::model::generaldata::QuantityResource  
sgad::clienttier::model::generaldata::Resource
```

```
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-Collection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-VoidFilter
sgad::clienttier::view::graphicobjects::components::-worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL3Shape
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::businesslogic::logic::Logic
```

	<pre> sgad::servertier::businesslogic::operations::- BuildConstruction sgad::servertier::businesslogic::operations::- DemolishBuilding sgad::servertier::businesslogic::operations::Operation sgad::servertier::dataaccess::data::userdata::- AuthenticationData sgad::servertier::presentation::httpresponder::- ResponderActor sgad::servertier::presentation::httpresponder::- WorkerActor sgad::servertier::presentation::messages::- ToPublisherAndUserRequest sgad::servertier::presentation::messages::- ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::- PublisherActor sgad::servertier::presentation::usermanager::UserActor </pre>
R1F5.1.5.1	<pre> sgad::clienttier::controller::actions::- ShowBuildingContextualMenu sgad::clienttier::controller::menufactory::- BuildingContextualMenuFactory sgad::clienttier::view::commands::RightClick sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::- GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::- worldcomponent::WorldComponent sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::- DemolishBuilding </pre>
R1F5.1.5.1.1	<pre> sgad::clienttier::controller::menufactory::- BuildingContextualMenuFactory sgad::clienttier::model::generaldata::BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel </pre>

R1F5.1.5.2	<pre> sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::- RequestDemolishBuilding sgad::clienttier::controller::messageinterpreter::- MessageInterpreter sgad::clienttier::controller::requester::AJAXRequester sgad::clienttier::model::personaldata::BuildingPossession sgad::clienttier::view::graphicobjects::components::- worldcomponent::BuildingComponent sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::UserData </pre>
R1F5.1.5.2.1	<pre> sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::components::- worldcomponent::BuildingComponent sgad::clienttier::view::graphicobjects::components::- worldcomponent::TileComponent sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::UserData </pre>
R1F5.1.5.2.2	<pre> sgad::clienttier::controller::actions::DemolishBuilding sgad::servertier::businesslogic::operations::- DemolishBuilding </pre>
R1F5.1.5.2.3	<pre> sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::model::generaldata::BuildingWithLevel sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::- DemolishBuilding sgad::servertier::dataaccess::data::shareddata::- BuildingWithLevel sgad::servertier::dataaccess::data::shareddata::Cost sgad::servertier::dataaccess::data::shareddata::- QuantityResource sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::userdata::- BuildingPossession sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::UserData </pre>
R1F5.1.5.2.3.1	<pre> sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::model::generaldata::BuildingWithLevel </pre>

	sgad::clienttier::model::generaldata::Cost sgad::clienttier::model::generaldata::QuantityResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::userdata::- OwnedResource sgad::servertier::dataaccess::data::userdata::UserData
R1F5.1.6.1	sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener sgad::clienttier::controller::actions::BackToVillage sgad::clienttier::controller::actions::BuildConstruction sgad::clienttier::controller::actions::- CloseContextualMenu sgad::clienttier::controller::actions::DemolishBuilding sgad::clienttier::controller::actions::DismissUnits sgad::clienttier::controller::actions::EnableRightClick sgad::clienttier::controller::actions::HarvestResources sgad::clienttier::controller::actions::LoadAnotherUser sgad::clienttier::controller::actions::LoadGeneralData sgad::clienttier::controller::actions::LoadPersonalData sgad::clienttier::controller::actions::Logout sgad::clienttier::controller::actions::ReloadPageAction sgad::clienttier::controller::actions::- RemoveGraphicObjectAction sgad::clienttier::controller::actions::RequestAttackUser sgad::clienttier::controller::actions::- RequestBuildConstruction sgad::clienttier::controller::actions::- RequestDemolishBuilding sgad::clienttier::controller::actions::RequestDismissUnits sgad::clienttier::controller::actions::- RequestHarvestResources sgad::clienttier::controller::actions::- RequestLoadGeneralData sgad::clienttier::controller::actions::- RequestLoadPersonalData sgad::clienttier::controller::actions::RequestTrainUnits sgad::clienttier::controller::actions::- RequestUpgradeBuilding sgad::clienttier::controller::actions::SetAllVoidFilter sgad::clienttier::controller::actions::SetBuildModeFilter sgad::clienttier::controller::actions::- ShowAnotherUserMenu sgad::clienttier::controller::actions::ShowAttackMenu

```
sgad::clienttier::controller::actions::-  
ShowAttackResultMenu  
sgad::clienttier::controller::actions::-  
ShowBuildConstructionMenu  
sgad::clienttier::controller::actions::-  
ShowBuildingContextualMenu  
sgad::clienttier::controller::actions:::ShowDemolishMenu  
sgad::clienttier::controller::actions::-  
ShowDismissUnitMenu  
sgad::clienttier::controller::actions:::ShowInteractionMenu  
sgad::clienttier::controller::actions::-  
ShowOperationFailureMenu  
sgad::clienttier::controller::actions:::ShowResourceMenu  
sgad::clienttier::controller::actions::-  
ShowTileContextualMenu  
sgad::clienttier::controller::actions:::ShowTrainUnitMenu  
sgad::clienttier::controller::actions::-  
ShowUnitSelectionMenu  
sgad::clienttier::controller::actions:::ShowUpgradeMenu  
sgad::clienttier::controller::actions:::ShowUserListMenu  
sgad::clienttier::controller::actions:::StoleResources  
sgad::clienttier::controller::actions:::TrainUnit  
sgad::clienttier::controller::actions:::UpgradeBuilding  
sgad::clienttier::controller::backupmanager::-  
BackupManager  
sgad::clienttier::controller::menufactory::-  
AnotherUserMenuFactory  
sgad::clienttier::controller::menufactory::-  
AttackResultMenuFactory  
sgad::clienttier::controller::menufactory::-  
BuildingContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
ConfirmMenuFactory  
sgad::clienttier::controller::menufactory::-  
InteractionMenuFactory  
sgad::clienttier::controller::menufactory:::MenuFactory  
sgad::clienttier::controller::menufactory::-  
NotifyMenuFactory  
sgad::clienttier::controller::menufactory::-  
ResourceMenuFactory  
sgad::clienttier::controller::menufactory::-  
TileContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
UnitSelectionMenu  
sgad::clienttier::controller::menufactory::-  
UserListMenuFactory
```

```
sgad::clienttier::controller::messageinterpreter::-
MessageInterpreter
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-
AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-
UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-
Collection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-
GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-
BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-
GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-
VoidFilter
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::BuildingComponent
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::BarracksL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
```

	sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::clienttier::view::graphicobjects::widget::Widget sgad::servertier::businesslogic::operations::DismissUnit
R1F5.1.6.1.1	sgad::clienttier::controller::actions::- ShowBuildingContextualMenu sgad::clienttier::controller::menufactory::- UnitSelectionMenu sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::- GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::- worldcomponent::BuildingComponent sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget
R1F5.1.6.2	sgad::clienttier::controller::actions::- ShowDismissUnitMenu sgad::servertier::businesslogic::operations::DismissUnit
R1F5.1.6.2.1	sgad::servertier::businesslogic::operations::DismissUnit
R1F5.1.6.3	sgad::clienttier::controller::actions::DismissUnits sgad::servertier::businesslogic::operations::DismissUnit sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData
R1F5.1.6.4	sgad::clienttier::controller::actions::DismissUnits sgad::servertier::businesslogic::operations::DismissUnit
R1F5.1.6.5	sgad::clienttier::controller::actions::DismissUnits sgad::servertier::businesslogic::operations::DismissUnit
R1F5.2	sgad::clienttier::controller::actions::Action sgad::clienttier::controller::actions::ActionListener

```
sgad::clienttier::controller::actions::BackToVillage
sgad::clienttier::controller::actions::BuildConstruction
sgad::clienttier::controller::actions::-CloseContextualMenu
sgad::clienttier::controller::actions::DemolishBuilding
sgad::clienttier::controller::actions::DismissUnits
sgad::clienttier::controller::actions::EnableRightClick
sgad::clienttier::controller::actions::HarvestResources
sgad::clienttier::controller::actions::LoadAnotherUser
sgad::clienttier::controller::actions::LoadGeneralData
sgad::clienttier::controller::actions::LoadPersonalData
sgad::clienttier::controller::actions::Logout
sgad::clienttier::controller::actions::ReloadPageAction
sgad::clienttier::controller::actions::-RemoveGraphicObjectAction
sgad::clienttier::controller::actions::RequestAttackUser
sgad::clienttier::controller::actions::-RequestBuildConstruction
sgad::clienttier::controller::actions::-RequestDemolishBuilding
sgad::clienttier::controller::actions::RequestDismissUnits
sgad::clienttier::controller::actions::-RequestHarvestResources
sgad::clienttier::controller::actions::-RequestLoadGeneralData
sgad::clienttier::controller::actions::-RequestLoadPersonalData
sgad::clienttier::controller::actions::RequestTrainUnits
sgad::clienttier::controller::actions::-RequestUpgradeBuilding
sgad::clienttier::controller::actions::SetAllVoidFilter
sgad::clienttier::controller::actions::SetBuildModeFilter
sgad::clienttier::controller::actions::-ShowAnotherUserMenu
sgad::clienttier::controller::actions::ShowAttackMenu
sgad::clienttier::controller::actions::-ShowAttackResultMenu
sgad::clienttier::controller::actions::-ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-ShowDismissUnitMenu
sgad::clienttier::controller::actions::ShowInteractionMenu
```

```
sgad::clienttier::controller::actions::-  
ShowOperationFailureMenu  
sgad::clienttier::controller::actions::-ShowResourceMenu  
sgad::clienttier::controller::actions::-  
ShowTileContextualMenu  
sgad::clienttier::controller::actions::-ShowTrainUnitMenu  
sgad::clienttier::controller::actions::-  
ShowUnitSelectionMenu  
sgad::clienttier::controller::actions::-ShowUpgradeMenu  
sgad::clienttier::controller::actions::-ShowUserListMenu  
sgad::clienttier::controller::actions::StoleResources  
sgad::clienttier::controller::actions::TrainUnit  
sgad::clienttier::controller::actions::UpgradeBuilding  
sgad::clienttier::controller::backupmanager::-  
BackupManager  
sgad::clienttier::controller::menufactory::-  
AnotherUserMenuFactory  
sgad::clienttier::controller::menufactory::-  
AttackResultMenuFactory  
sgad::clienttier::controller::menufactory::-  
BuildingContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
ConfirmMenuFactory  
sgad::clienttier::controller::menufactory::-  
InteractionMenuFactory  
sgad::clienttier::controller::menufactory::MenuFactory  
sgad::clienttier::controller::menufactory::-  
NotifyMenuFactory  
sgad::clienttier::controller::menufactory::-  
ResourceMenuFactory  
sgad::clienttier::controller::menufactory::-  
TileContextualMenuFactory  
sgad::clienttier::controller::menufactory::-  
UnitSelectionMenu  
sgad::clienttier::controller::menufactory::-  
UserListMenuFactory  
sgad::clienttier::controller::messageinterpreter::-  
MessageInterpreter  
sgad::clienttier::controller::requester::AJAXRequester  
sgad::clienttier::model::generaldata::Bonus  
sgad::clienttier::model::generaldata::BuildingWithLevel  
sgad::clienttier::model::generaldata::Cost  
sgad::clienttier::model::generaldata::DataFactory  
sgad::clienttier::model::generaldata::ProducedResource  
sgad::clienttier::model::generaldata::QuantityResource  
sgad::clienttier::model::generaldata::Resource
```

```
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-Collection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-VoidFilter
sgad::clienttier::view::graphicobjects::components::-worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL3Shape
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::application::Application
```

R2F5.2.1

```

sgad::clienttier::controller::actions::Action
sgad::clienttier::controller::actions::ActionListener
sgad::clienttier::controller::actions::BackToVillage
sgad::clienttier::controller::actions::BuildConstruction
sgad::clienttier::controller::actions::-CloseContextualMenu
sgad::clienttier::controller::actions::DemolishBuilding
sgad::clienttier::controller::actions::DismissUnits
sgad::clienttier::controller::actions::EnableRightClick
sgad::clienttier::controller::actions::HarvestResources
sgad::clienttier::controller::actions::LoadAnotherUser
sgad::clienttier::controller::actions::LoadGeneralData
sgad::clienttier::controller::actions::LoadPersonalData
sgad::clienttier::controller::actions::Logout
sgad::clienttier::controller::actions::ReloadPageAction
sgad::clienttier::controller::actions::-RemoveGraphicObjectAction
sgad::clienttier::controller::actions::RequestAttackUser
sgad::clienttier::controller::actions::-RequestBuildConstruction
sgad::clienttier::controller::actions::-RequestDemolishBuilding
sgad::clienttier::controller::actions::RequestDismissUnits
sgad::clienttier::controller::actions::-RequestHarvestResources
sgad::clienttier::controller::actions::-RequestLoadGeneralData
sgad::clienttier::controller::actions::-RequestLoadPersonalData
sgad::clienttier::controller::actions::RequestTrainUnits
sgad::clienttier::controller::actions::-RequestUpgradeBuilding
sgad::clienttier::controller::actions::SetAllVoidFilter
sgad::clienttier::controller::actions::SetBuildModeFilter
sgad::clienttier::controller::actions::-ShowAnotherUserMenu
sgad::clienttier::controller::actions::ShowAttackMenu
sgad::clienttier::controller::actions::-ShowAttackResultMenu
sgad::clienttier::controller::actions::-ShowBuildConstructionMenu
sgad::clienttier::controller::actions::-ShowBuildingContextualMenu
sgad::clienttier::controller::actions::ShowDemolishMenu
sgad::clienttier::controller::actions::-ShowDismissUnitMenu
  
```

```
sgad::clienttier::controller::actions::ShowInteractionMenu
sgad::clienttier::controller::actions::-ShowOperationFailureMenu
sgad::clienttier::controller::actions::ShowResourceMenu
sgad::clienttier::controller::actions::-ShowTileContextualMenu
sgad::clienttier::controller::actions::ShowTrainUnitMenu
sgad::clienttier::controller::actions::-ShowUnitSelectionMenu
sgad::clienttier::controller::actions::ShowUpgradeMenu
sgad::clienttier::controller::actions::ShowUserListMenu
sgad::clienttier::controller::actions::StoleResources
sgad::clienttier::controller::actions::TrainUnit
sgad::clienttier::controller::actions::UpgradeBuilding
sgad::clienttier::controller::backupmanager::-BackupManager
sgad::clienttier::controller::menufactory::-AnotherUserMenuFactory
sgad::clienttier::controller::menufactory::-AttackResultMenuFactory
sgad::clienttier::controller::menufactory::-BuildingContextualMenuFactory
sgad::clienttier::controller::menufactory::-ConfirmMenuFactory
sgad::clienttier::controller::menufactory::-InteractionMenuFactory
sgad::clienttier::controller::menufactory::MenuFactory
sgad::clienttier::controller::menufactory::-NotifyMenuFactory
sgad::clienttier::controller::menufactory::-ResourceMenuFactory
sgad::clienttier::controller::menufactory::-TileContextualMenuFactory
sgad::clienttier::controller::menufactory::-UnitSelectionMenu
sgad::clienttier::controller::menufactory::-UserListMenuFactory
sgad::clienttier::controller::messageinterpreter::-MessageInterpreter
sgad::clienttier::controller::requester::AJAXRequester
sgad::clienttier::model::generaldata::Bonus
sgad::clienttier::model::generaldata::BuildingWithLevel
sgad::clienttier::model::generaldata::Cost
sgad::clienttier::model::generaldata::DataFactory
sgad::clienttier::model::generaldata::ProducedResource
sgad::clienttier::model::generaldata::QuantityResource
```

```
sgad::clienttier::model::generaldata::Resource
sgad::clienttier::model::generaldata::Unit
sgad::clienttier::model::observer::Observable
sgad::clienttier::model::observer::Observer
sgad::clienttier::model::personaldata::-AuthenticationData
sgad::clienttier::model::personaldata::BuildingPossession
sgad::clienttier::model::personaldata::OwnedResource
sgad::clienttier::model::personaldata::Position
sgad::clienttier::model::personaldata::UnitInProgress
sgad::clienttier::model::personaldata::UnitPossession
sgad::clienttier::model::personaldata::UserData
sgad::clienttier::model::userdatamanager::-UserDataManager
sgad::clienttier::view::commands::Click
sgad::clienttier::view::commands::Command
sgad::clienttier::view::commands::DragAndDrop
sgad::clienttier::view::commands::Resize
sgad::clienttier::view::commands::RightClick
sgad::clienttier::view::context::Context
sgad::clienttier::view::graphicobjects::bound::Bound
sgad::clienttier::view::graphicobjects::collection::-Collection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection
sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator
sgad::clienttier::view::graphicobjects::collection::Iterator
sgad::clienttier::view::graphicobjects::components::filter::-BuildModeFilter
sgad::clienttier::view::graphicobjects::components::filter::-GraphicFilter
sgad::clienttier::view::graphicobjects::components::filter::-VoidFilter
sgad::clienttier::view::graphicobjects::components::-worldcomponent::BuildingComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::TileComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponent::WorldComponent
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL1Shape
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL2Shape
sgad::clienttier::view::graphicobjects::components::-worldcomponentshape::BarracksL3Shape
```

```
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::InConstructionShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::MineL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::SchoolOfMagicL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::StableL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::TileShape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL1Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL2Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WizardTowerL3Shape
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeFactory
sgad::clienttier::view::graphicobjects::components::-
worldcomponentshape::WorldComponentShapeImg
sgad::clienttier::view::graphicobjects::graphicobject::-
GraphicObject
sgad::clienttier::view::graphicobjects::point::Point2D
sgad::clienttier::view::graphicobjects::shape::Shape
sgad::clienttier::view::graphicobjects::widget::-
ButtonWidget
sgad::clienttier::view::graphicobjects::widget::-
FrameWidget
sgad::clienttier::view::graphicobjects::widget::-
ImageWidget
sgad::clienttier::view::graphicobjects::widget::-
TextWidget
sgad::clienttier::view::graphicobjects::widget::Widget
sgad::servertier::businesslogic::logic::Logic
```

	sgad::servertier::businesslogic::operations::Attack sgad::servertier::businesslogic::operations::-BuildConstruction sgad::servertier::businesslogic::operations::InternalLogin sgad::servertier::businesslogic::operations::Operation sgad::servertier::businesslogic::operations::ReceiveAttack sgad::servertier::businesslogic::operations::StealResource sgad::servertier::businesslogic::operations::StealResources sgad::servertier::dataaccess::data::shareddata::Resource sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::-AuthenticationData sgad::servertier::dataaccess::data::userdata::-OwnedResource sgad::servertier::dataaccess::data::userdata::-UnitPossession sgad::servertier::dataaccess::data::userdata::UserData sgad::servertier::presentation::httpresponder::-ResponderActor sgad::servertier::presentation::httpresponder::-WorkerActor sgad::servertier::presentation::messages::-ToPublisherAndUserRequest sgad::servertier::presentation::messages::-ToWorkerUserRequest sgad::servertier::presentation::timeout::STimeout sgad::servertier::presentation::usermanager::-PublisherActor sgad::servertier::presentation::usermanager::UserActor
R2F5.2.1.1	sgad::clienttier::controller::actions::ShowUserListMenu sgad::clienttier::controller::menufactory::-UserListMenuFactory sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::-GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::-GraphicObjectIterator sgad::servertier::businesslogic::operations::Attack
R2F5.2.1.1.1	sgad::servertier::businesslogic::operations::ReceiveAttack
R2F5.2.1.1.1.1	sgad::clienttier::controller::actions::ShowAttackMenu sgad::clienttier::controller::actions::-ShowAttackResultMenu sgad::clienttier::model::generaldata::Unit sgad::clienttier::model::personaldata::UserData

	<pre> sgad::clienttier::model::userdatamanager::- UserDataManager sgad::servertier::businesslogic::operations::Attack sgad::servertier::businesslogic::operations::ReceiveAttack sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData </pre>
R2F5.2.1.1.2	<pre> sgad::clienttier::model::generaldata::Unit sgad::servertier::dataaccess::data::shareddata::Unit </pre>
R2F5.2.1.2	<pre> sgad::clienttier::controller::actions::RequestAttackUser sgad::servertier::businesslogic::operations::Attack sgad::servertier::dataaccess::data::shareddata::Unit sgad::servertier::dataaccess::data::userdata::- UnitPossession sgad::servertier::dataaccess::data::userdata::UserData </pre>
R2F5.2.1.3	<pre> sgad::clienttier::controller::actions::- ShowAttackResultMenu sgad::clienttier::controller::menufactory::- AttackResultMenuFactory sgad::clienttier::view::graphicobjects::widget::- ButtonWidget sgad::clienttier::view::graphicobjects::widget::- FrameWidget sgad::clienttier::view::graphicobjects::widget::- ImageWidget sgad::clienttier::view::graphicobjects::widget::- TextWidget sgad::servertier::businesslogic::operations::Attack </pre>
R2F5.2.1.4	<pre> sgad::clienttier::controller::actions::StoleResources sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::- GraphicObjectIterator sgad::servertier::businesslogic::operations::- ReceiveStealResouce sgad::servertier::businesslogic::operations::StealResource sgad::servertier::businesslogic::operations::StealResources sgad::servertier::presentation::usermanager::UserActor </pre>
R2F5.2.1.4.1	<pre> sgad::clienttier::controller::actions::StoleResources sgad::clienttier::model::personaldata::OwnedResource sgad::clienttier::model::personaldata::UserData sgad::clienttier::model::userdatamanager::- UserDataManager </pre>

	sgad::servtier::businesslogic::operations::StealResources sgad::servtier::dataaccess::data::shareddata::Resource sgad::servtier::dataaccess::data::userdata::- OwnedResource sgad::servtier::dataaccess::data::userdata::UserData
R2F5.2.1.4.2	sgad::clienttier::view::commands::Click sgad::clienttier::view::context::Context sgad::clienttier::view::graphicobjects::collection::- GraphicObjectCollection sgad::clienttier::view::graphicobjects::collection::- GraphicObjectIterator sgad::clienttier::view::graphicobjects::components::- worldcomponent::BuildingComponent sgad::servtier::businesslogic::operations::StealResources

Tabella 3: Tracciamento Requisiti-Classi