

Model Report

Version •



Date/Time Generated:

11/10/2017 10:20:34 AM

Author:

Luís Melo

EA Repository : L:\IdeaProjects\player-finder\docs\PPES_joined.EAP

Table of Contents

1 Model	8
1.1 LogicView	8
1.1.1 Class Diagram diagram	8
1.1.2 AbstractAuditing	9
1.1.3 Authority	10
1.1.4 Game	10
1.1.4.1 GameStateMachine	12
1.1.4.1.1 GameStateMachine diagram	13
1.1.4.1.2 State	13
1.1.4.1.2.1 CANCELLED	14
1.1.4.1.2.2 FINISHED	14
1.1.4.1.3 State	14
1.1.4.1.3.1 Initial State	15
1.1.4.1.3.2 ACTIVE	15
1.1.4.1.3.3 FULL	16
1.1.4.1.4 # of players reached limit	16
1.1.4.1.5 after 1 week	16
1.1.4.1.6 after 1 week	16
1.1.4.1.7 Game creation	17
1.1.4.1.8 owner cancels the game	17
1.1.4.1.9 owner cancels the game	17
1.1.4.1.10 when actual date > game.endTime	17
1.1.4.1.11 when actual date > game.endTime	17
1.1.4.1.12 ARCHIEVED	17
1.1.4.1.13 PENDING	18
1.1.4.1.14 StateMachine1	18
1.1.4.1.14.1 StateMachine1 diagram	18
1.1.4.1.15 Initial	19
1.1.4.1.16 Synch	19
1.1.4.1.17 Terminate	19
1.1.5 GameType	19
1.1.6 Location	20
1.1.7 User	21
1.1.8 GameStatus	22
1.1.9 TypeOfGame	23
1.2 Use Case View	24
1.2.1 Create Game Activity Diagram diagram	24
1.2.2 FlowFinal	24
1.2.3 Logged In	25
1.2.4 Synchronization	25
1.2.5 Synchronization	25
1.2.6 Game Sequence Diagram diagram	26
1.2.7 Select Game Activity Diagram diagram	31
1.2.8 FlowFinal	31
1.2.9 FlowFinal	32
1.2.10 FlowFinal	32
1.2.11 Logged In	32
1.2.12 Use Case Diagram diagram	32
1.2.13 PlayerFinder Application	33

1.2.14	Note	33
1.2.15	Administrator	34
1.2.16	User	34
1.2.17	Visitor	35
1.2.18	Cancel Game	35
1.2.19	Create Game	36
1.2.19.1	Create Game_SequenceDiagram	37
1.2.19.1.1	System	37
1.2.19.1.2	User	38
1.2.20	Create new Location	38
1.2.21	Edit Game	39
1.2.22	Edit Profile	39
1.2.23	Introduce a Day, Hour for Stranting and Finishing	39
1.2.24	Introduce Location	40
1.2.25	Introduce Number of Slots	40
1.2.26	Introduce Other Information	40
1.2.27	Join Game	41
1.2.28	Manage Game	41
1.2.29	Manage the System	42
1.2.30	Manage Types of Games	42
1.2.31	Register in system	42
1.2.31.1	Register in system social Account	43
1.2.31.1.1	Click register with some social account	44
1.2.31.1.2	End	44
1.2.31.1.3	Start	44
1.2.31.2	Register in system Social Account	44
1.2.31.2.1	Click register with some social account	45
1.2.31.2.2	End	45
1.2.31.2.3	Start	45
1.2.31.3	Register in system_SequenceDiagram	45
1.2.31.3.1	System	46
1.2.31.3.2	System	46
1.2.31.3.3	System12	46
1.2.31.3.4	Visitor	46
1.2.31.4	Object1	46
1.2.32	Search Games	47
1.2.33	Use Case Model diagram	47
1.2.34	Note	48
1.2.35	Note	48
1.2.36	Note	49
1.2.37	Actors	49
1.2.38	Primary Use Cases	49
1.2.39	\$help://actor.htm	49
1.2.40	\$help://use_case_model_pattern.htm	49
1.2.41	\$help://usecasediagram.htm	50
1.2.42	Visitor Login Sequence Diagram diagram	50
1.2.43	Actors	52
1.2.43.1	Actors diagram	52
1.2.43.2	Actor1	53
1.2.43.3	Administrator	53
1.2.43.4	System	53
1.2.43.5	User	54
1.2.43.6	Visitor	55
1.2.44	Primary Use Cases	56

1.2.44.1	Cancel Game	56
1.2.44.2	Create Game	56
1.2.44.2.1	Create Game_SequenceDiagram	57
1.2.44.2.1.1	CreateGame_Sequence diagram	57
1.2.44.2.1.2	System	59
1.2.44.2.1.3	User	59
1.2.44.3	Create new Location	60
1.2.44.4	Edit Game	60
1.2.44.5	Edit Profile	60
1.2.44.6	Introduce Location	61
1.2.44.7	Introduce Number of Slots	61
1.2.44.8	Introduce Other Information	61
1.2.44.9	Introduce a Day, Hour for Stranting and Finishing	62
1.2.44.10	Join Game	62
1.2.44.11	Login in platform	62
1.2.44.12	Logout	62
1.2.44.13	Manage Game	63
1.2.44.14	Manage Types of Games	63
1.2.44.15	Manage the System	64
1.2.44.16	Register in system	64
1.2.44.16.1	Use Case1 diagram	65
1.2.44.16.2	User	65
1.2.44.16.3	Register in system social Account	67
1.2.44.16.3.1	Register in system Social Account diagram	67
1.2.44.16.3.2	Click register with some social account	67
1.2.44.16.3.3	End	68
1.2.44.16.3.4	Start	68
1.2.44.16.4	Register in system Social Account	68
1.2.44.16.4.1	Register in system Social Account diagram	68
1.2.44.16.4.2	Click register with some social account	69
1.2.44.16.4.3	End	69
1.2.44.16.4.4	Start	69
1.2.44.16.5	Register in system_SequenceDiagram	70
1.2.44.16.5.1	5b_User inputs invalid data diagram	70
1.2.44.16.5.2	5a_Social Register diagram	71
1.2.44.16.5.3	1_Basic_Path diagram	71
1.2.44.16.5.4	System	71
1.2.44.16.5.5	System	72
1.2.44.16.5.6	System12	72
1.2.44.16.5.7	Visitor	72
1.2.44.16.6	Object1	72
1.2.44.17	Search Games	73
1.2.44.18	Object	73
1.2.45	User	73
1.2.46	Visitor	74
1.2.47	Game joined	74
1.2.48	Object1	74
1.2.49	Activity1	74
1.2.50	Back To Create Game	75
1.2.51	Back To Home Page	75
1.2.52	Back To Home Page	75
1.2.53	Change Information about the Game	75
1.2.54	Change Status	76
1.2.55	Consult Location	76

1.2.56	Create Game	76
1.2.57	Delete Game	77
1.2.58	Edit Game	77
1.2.58.1	ObjectNode1	77
1.2.59	Introduce Finish Date	77
1.2.60	Introduce Name	78
1.2.61	Introduce Number Of Players	78
1.2.62	Introduce Number of Open Slots	78
1.2.62.1	ObjectNode2	79
1.2.63	Introduce Start Date	79
1.2.64	Join Game	79
1.2.65	Select Game	80
1.2.66	Select Location	80
1.2.67	FlowFinal	80
1.2.68	FlowFinal	81
1.2.69	FlowFinal	81
1.2.70	FlowFinal	81
1.2.71	DecisionNode	81
1.2.72	InteractionFragment	81
1.2.73	ActivityFinal	82
1.2.74	Are you sure?	82
1.2.75	Are you sure?	82
1.2.76	Are you the Game Owner?	82
1.2.77	Database	83
1.2.78	Delete Game	84
1.2.79	Information Invalid?	84
1.2.80	InputData	85
1.2.81	Interface	85
1.2.82	Logged In	86
1.2.83	Logged In	87
1.2.84	Login Interface	87
1.2.85	Object1	87
1.2.86	Register	87
1.2.87	Service	88
1.2.88	Update Game	89
1.2.89	User	89
1.2.90	Validation	89
1.2.91	Verification	90
1.2.92	Visitor Not Register	90
1.2.93	dd	90
1.3	Component Model	91
1.3.1	Component Model diagram	91
1.3.2	Note	91
1.3.3	Note	91
1.3.4	Note	92
1.3.5	Note	92
1.3.6	Components	92
1.3.7	Connections	92
1.3.8	Internal Structures	93
1.3.9	\$help://component_model_pattern.htm	93
1.3.10	\$help://componentdiagram.htm	93
1.3.11	Components Diagram diagram	93
1.3.12	Note	94
1.3.13	Angular SPA	94

1.3.14	External API	94
1.3.15	Java Spring RESTful Web API	95
1.3.16	Components	95
1.3.16.1	Components diagram	95
1.3.16.2	Component1	95
1.3.16.2.1	Part1	96
1.3.16.2.2	ProvidedInterface1	96
1.3.16.3	Component2	97
1.3.16.3.1	RequiredInterface	97
1.3.17	Connections	98
1.3.17.1	Connections diagram	98
1.3.17.2	Component1	98
1.3.17.2.1	Part1	98
1.3.17.2.2	ProvidedInterface1	99
1.3.17.3	Component2	99
1.3.17.3.1	RequiredInterface	99
1.3.17.4	ProvidedInterface1	100
1.3.17.5	RequiredInterface	100
1.3.17.6	Note	100
1.3.17.7	Note	101
1.3.17.8	Note	101
1.3.18	External API	102
1.3.19	Internal Structures	103
1.3.19.1	Internal Structures diagram	103
1.3.19.2	Component1	103
1.3.19.2.1	Part1	103
1.3.19.2.2	ProvidedInterface1	104
1.3.19.3	Part1	104
1.3.19.4	ProvidedInterface1	105
1.3.19.5	Note	105
1.3.19.6	Note	105
1.3.20	Package1	107
1.3.20.1	Package1 diagram	107
1.3.21	Angular SPA	108
1.3.22	Component1	109
1.3.23	External API	110
1.3.24	Java Spring RESTful Web API	111
1.3.25	Interface1	111
1.3.26	Interface2	111
1.3.27	Component	111
1.3.28	Component	111
1.3.28.1	Controller	111
1.3.28.2	Model	112
1.3.28.3	View	112
1.3.29	Controller	113
1.3.30	Database	113
1.3.31	Domain	113
1.3.32	Repository	113
1.3.33	Resource	114
1.3.34	Security	114
1.3.35	Service	115
1.4	Deployment Model	116
1.4.1	Deployment Diagram diagram	116
1.4.2	Note	116

1.4.3	Deployment Model diagram	117
1.4.4	Note	117
1.4.5	Note	117
1.4.6	Note	118
1.4.7	Note	118
1.4.8	\$help://deployment_model_pattern.htm	118
1.4.9	\$help://deploymentdiagram.htm	118
1.4.10	Nodes	118
1.4.11	Artifacts	119
1.4.12	Topology	119
1.4.13	Nodes	119
1.4.13.1	Nodes diagram	119
1.4.13.2	Clients	119
1.4.13.2.1	Clients diagram	120
1.4.13.3	Devices	121
1.4.13.3.1	Devices diagram	121
1.4.13.4	Servers	122
1.4.13.4.1	Servers diagram	122
1.4.14	Artifacts	123
1.4.14.1	Artifacts diagram	123
1.4.15	Topology	124
1.4.15.1	Network diagram	124
1.4.16	Node1	124
1.4.17	Component1	124
1.4.18	Google Server	124
1.4.19	Client Machine (Computer)	124
1.4.19.1	Browser	125
1.4.20	Client Machine (Mobile Phone)	125
1.4.20.1	Browser	125
1.4.21	Deployment Machine	126
1.4.21.1	Apache Tomcat	126
1.4.21.1.1	Angular SPA	127
1.4.21.1.2	Java Spring RESTful Web API	127
1.4.21.2	Java	128
1.4.21.3	MySQL	128
1.4.21.3.1	Database	128
1.4.22	Device1	128
1.4.23	Device2	129
1.4.24	Facebook Authentication API	129
1.4.25	Google Authentication API	129
1.4.26	Google Maps Server	129
1.4.27	Twitter Authentication API	130

1 Model

Package in package "

Model
Version Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1 LogicView

Package in package 'Model'

LogicView
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/3/2017

1.1.1 Class Diagram diagram

Class diagram in package 'LogicView'

Class Diagram
Version 1.0
Luís Melo created on 11/3/2017. Last modified 11/10/2017

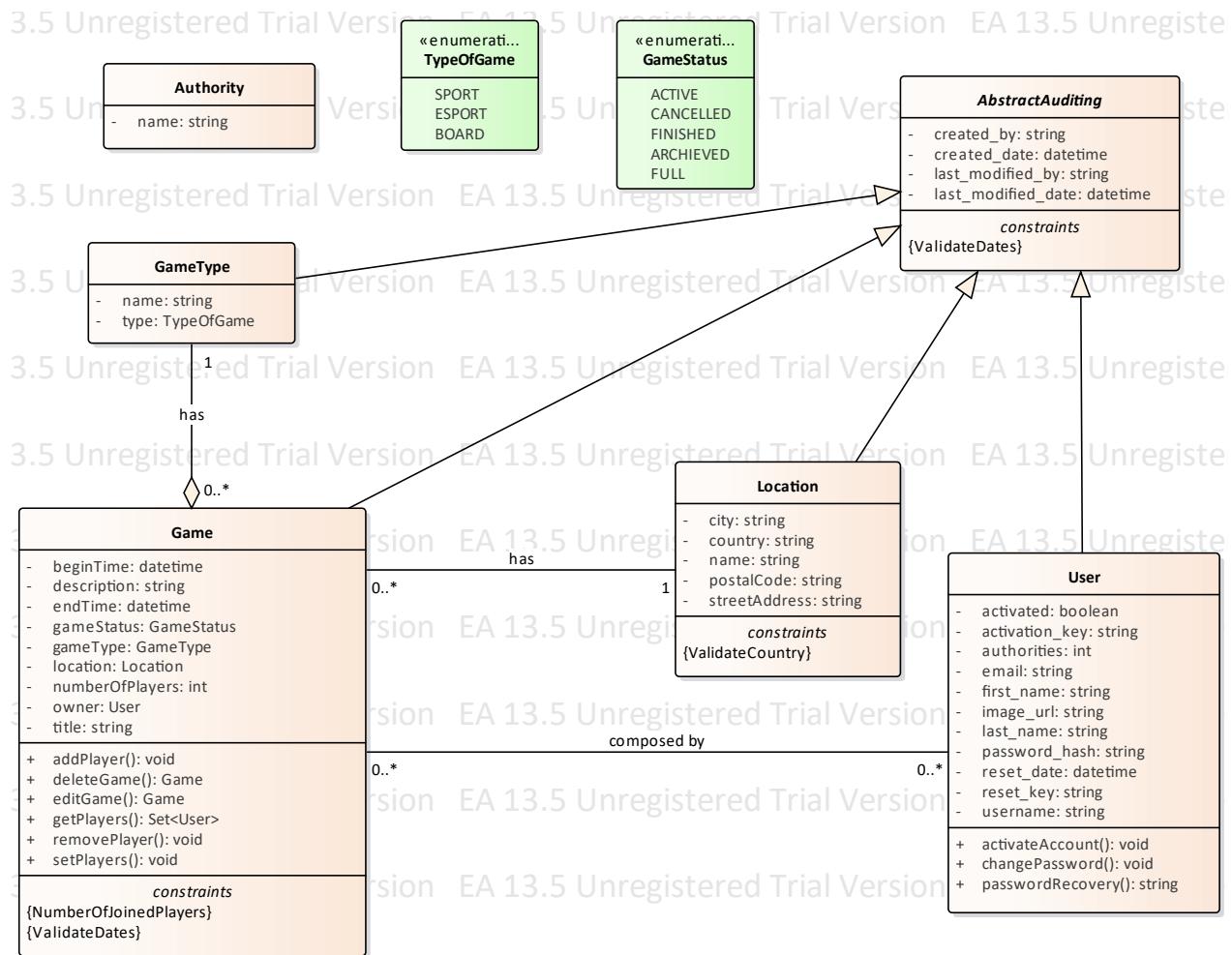


Figure 1: Class Diagram

1.1.2 AbstractAuditing

Class in package 'LogicView'

This model is responsible to auditing all operations.

Because these are recurring operations, all other classes will inherit the attributes and operations of this class.

Constraints:

ValidateDates: update dates should always be after the date of creation

AbstractAuditing

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/3/2017. Last modified 11/10/2017

CONSTRAINTS

⌚ OCL. ValidateDates

inv Auditing: self.last_modified_date < self.created_date

[Approved, Weight is 0.]

INCOMING STRUCTURAL RELATIONSHIPS

➡ Generalization from Location to AbstractAuditing

[Direction is 'Source -> Destination'.]

➡ Generalization from User to AbstractAuditing

[Direction is 'Source -> Destination'.]

➡ Generalization from Game to AbstractAuditing

[Direction is 'Source -> Destination'.]

➡ Generalization from GameType to AbstractAuditing

[Direction is 'Source -> Destination'.]

ATTRIBUTES

◆ created_by : string Private

[Is static False. Containment is Not Specified.]

◆ created_date : datetime Private

[Is static False. Containment is Not Specified.]

◆ last_modified_by : string Private

[Is static False. Containment is Not Specified.]

◆ last_modified_date : datetime Private

[Is static False. Containment is Not Specified.]

1.1.3 Authority

Class in package 'LogicView'

Authority
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/10/2017

ATTRIBUTES

◆ name : string Private

[Is static False. Containment is Not Specified.]

1.1.4 Game

Class in package 'LogicView'

Game entity will model all games that were and will be created

Constraints:

NumberOfJoinedPlayers: Players joined must be less than the allowed

ValidateDates: update dates should always be after the date of creation

Game
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/10/2017
Extends AbstractAuditing

ELEMENTS OWNED BY Game

█ GameStateMachine : StateMachine

CONSTRAINTS

⌚ OCL. NumberOfJoinedPlayers

inv Game: self.getPlayers()>size()<=self.numberOfPlayers

[Approved, Weight is 0.]

⌚ OCL. ValidateDates

inv Game: self.beginTime< self.endTime

[Approved, Weight is 1.]

OUTGOING STRUCTURAL RELATIONSHIPS

◀ Generalization from Game to AbstractAuditing

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Aggregation from GameType to Game

[Name is has. Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ beginTime : datetime	Private [Is static False. Containment is Not Specified.]
◆ description : string	Private [Is static False. Containment is Not Specified.]
◆ endTime : datetime	Private [Is static False. Containment is Not Specified.]
◆ gameStatus : GameStatus	Private [Is static False. Containment is Not Specified.]
◆ gameType : GameType	Private [Is static False. Containment is Not Specified.]
◆ location : Location	Private [Is static False. Containment is Not Specified.]
◆ numberOfPlayers : int	Private [Is static False. Containment is Not Specified.]
◆ owner : User	Private [Is static False. Containment is Not Specified.]
◆ title : string	Private [Is static False. Containment is Not Specified.]

ASSOCIATIONS	
✓ Association (direction: Unspecified) has	Source: Public (Class) Game Cardinality: [0..*]
	Target: Public (Class) Location Cardinality: [1]
✓ Association (direction: Unspecified) composed by	Source: Public (Class) Game Cardinality: [0..*]
	Target: Public (Class) User Cardinality: [0..*]

OPERATIONS	
◆ addPlayer () : void	Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ deleteGame () : Game	Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ editGame () : Game	Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

OPERATIONS	
◆ getPlayers () : Set<User>	Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ removePlayer () : void	Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ setPlayers () : void	Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

1.1.4.1 **GameStateMachine**

StateMachine owned by 'Game', in package 'LogicView'

GameStateMachine
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/4/2017. Last modified 11/4/2017

ELEMENTS OWNED BY GameStateMachine	
█ State : State	
█ State : State	
█ # of players reached limit : Trigger	
█ after 1 week : Trigger	
█ after 1 week : Trigger	
█ Game creation : Trigger	
█ owner cancels the game : Trigger	
█ owner cancels the game : Trigger	
█ when actual date > game.endTime : Trigger	
█ when actual date > game.endTime : Trigger	
█ ARCHIEVED : State	
█ PENDING : State	
█ StateMachine1 : StateMachine	

ELEMENTS OWNED BY GameStateMachine	
Initial : Initial State	
Synch : Synch State	
Terminate : Terminate	

1.1.4.1.1 GameStateMachine diagram

StateMachine diagram in package 'LogicView'

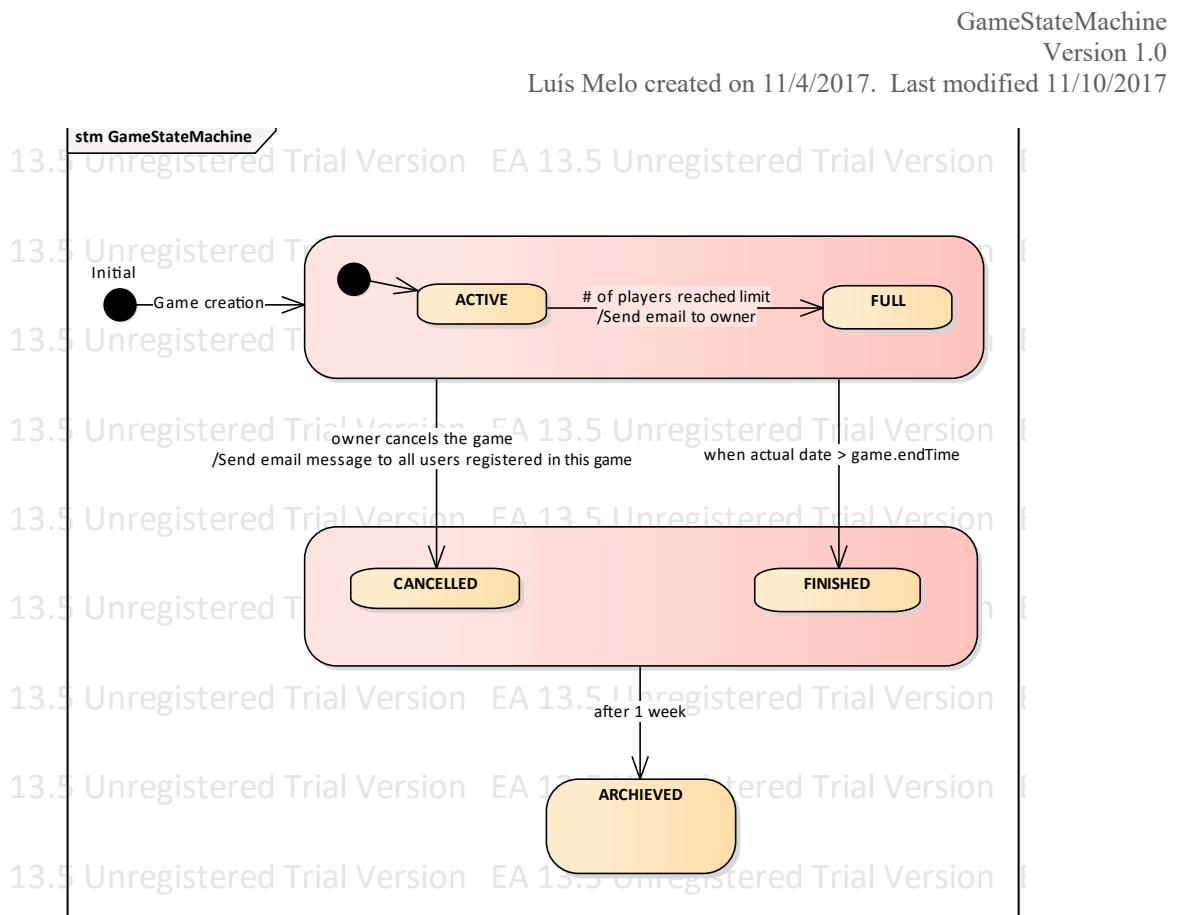


Figure 2: GameStateMachine

1.1.4.1.2 State

State owned by 'GameStateMachine', in package 'LogicView'

State
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

ELEMENTS OWNED BY	
Initial : Initial State	

ELEMENTS OWNED BY	
 CANCELLED : State	
 FINISHED : State	

OUTGOING BEHAVIORAL RELATIONSHIPS	
 Transition from State to ARCHIEVED	
Triggers: after 1 week Time	

1.1.4.1.2.1 CANCELLED

State in package 'LogicView'

CANCELLED

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/4/2017. Last modified 11/10/2017

INCOMING BEHAVIORAL RELATIONSHIPS	
 Transition from State to CANCELLED	
Effect: Send email message to all users registered in this game	
Triggers: owner cancels the game Signal	
 Transition from PENDING to CANCELLED	
Guard: owner cancels the game	

1.1.4.1.2.2 FINISHED

State in package 'LogicView'

FINISHED

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/4/2017. Last modified 11/10/2017

INCOMING BEHAVIORAL RELATIONSHIPS	
 Transition from State to FINISHED	
Triggers: when actual date > game.endTime Time	

1.1.4.1.3 State

State owned by 'GameStateMachine', in package 'LogicView'

State

ELEMENTS OWNED BY	
 Initial State	: Initial State
 ACTIVE	: State
 FULL	: State

OUTGOING BEHAVIORAL RELATIONSHIPS	
 Transition	from State to CANCELLED
Effect:	Send email message to all users registered in this game
Triggers:	
	owner cancels the game Signal
 Transition	from State to FINISHED
Triggers:	
	when actual date > game.endTime Time

INCOMING BEHAVIORAL RELATIONSHIPS	
 Transition	from Initial to State
Triggers:	
	Game creation Call

1.1.4.1.3.1 Initial State

Initial State in package 'LogicView'

OUTGOING BEHAVIORAL RELATIONSHIPS	
 Transition	from Initial State to ACTIVE

1.1.4.1.3.2 ACTIVE

State in package 'LogicView'

OUTGOING BEHAVIORAL RELATIONSHIPS	
 Transition	from ACTIVE to Synch

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Transition from ACTIVE to FULL

Effect: Send email to owner

Triggers:

of players reached limit Call

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Transition from Initial State to ACTIVE

⇒ Transition from PENDING to ACTIVE

Guard: owner makes the game public

1.1.4.1.3.3 FULL

State in package 'LogicView'

FULL

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/4/2017. Last modified 11/10/2017

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Transition from ACTIVE to FULL

Effect: Send email to owner

Triggers:

of players reached limit Call

1.1.4.1.4 # of players reached limit

Trigger owned by 'GameStateMachine', in package 'LogicView'

of players reached limit

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.5 after 1 week

Trigger owned by 'GameStateMachine', in package 'LogicView'

after 1 week

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.6 after 1 week

Trigger owned by 'GameStateMachine', in package 'LogicView'

after 1 week

Version 1.0 Phase 1.0 Proposed

1.1.4.1.7 Game creation

Trigger owned by 'GameStateMachine', in package 'LogicView'

Game creation
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.8 owner cancels the game

Trigger owned by 'GameStateMachine', in package 'LogicView'

owner cancels the game
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.9 owner cancels the game

Trigger owned by 'GameStateMachine', in package 'LogicView'

owner cancels the game
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.10 when actual date > game.endTime

Trigger owned by 'GameStateMachine', in package 'LogicView'

when actual date > game.endTime
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.11 when actual date > game.endTime

Trigger owned by 'GameStateMachine', in package 'LogicView'

when actual date > game.endTime
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/10/2017. Last modified 11/10/2017

1.1.4.1.12 ARCHIEVED

State owned by 'GameStateMachine', in package 'LogicView'

ARCHIEVED
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/4/2017. Last modified 11/4/2017

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Transition from State to ARCHIEVED

INCOMING BEHAVIORAL RELATIONSHIPS
Triggers: after 1 week Time

1.1.4.1.13 PENDING

State owned by 'GameStateMachine', in package 'LogicView'

PENDING
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/4/2017. Last modified 11/4/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
↳ Transition from PENDING to CANCELLED Guard: owner cancels the game
↳ Transition from PENDING to ACTIVE Guard: owner makes the game public

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Transition from Initial to PENDING
Guard: Game Creation

1.1.4.1.14 StateMachine1

StateMachine owned by 'GameStateMachine', in package 'LogicView'

StateMachine1
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/4/2017. Last modified 11/4/2017

1.1.4.1.14.1 StateMachine1 diagram

StateMachine diagram in package 'LogicView'

StateMachine1
Version 1.0
Luís Melo created on 11/4/2017. Last modified 11/4/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 3: StateMachine1

1.1.4.1.15 Initial

Initial State owned by 'GameStateMachine', in package 'LogicView'

Initial State of one Game after a certain user created it.

After it is created, the game is available to public and will remain active state until its begin date or after cancellation by the owner.

The game accepts new players to join until reach the target number of players required by the owner.

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Transition from Initial to State

Triggers:
Game creation Call

↳ Transition from Initial to PENDING

Guard: Game Creation

1.1.4.1.16 Synch

Synch State owned by 'GameStateMachine', in package 'LogicView'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Transition from ACTIVE to Synch

1.1.4.1.17 Terminate

Terminate owned by 'GameStateMachine', in package 'LogicView'

1.1.5 GameType

Class in package 'LogicView'

GameType

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 11/3/2017. Last modified 11/10/2017

Extends AbstractAuditing

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Aggregation from GameType to Game

[Name is has. Direction is 'Source -> Destination'.]

↳ Generalization from GameType to AbstractAuditing

[Direction is 'Source -> Destination'.]

ATTRIBUTES

ATTRIBUTES	
◆ name : string Private	[Is static False. Containment is Not Specified.]
◆ type : TypeOfGame Private	[Is static False. Containment is Not Specified.]

1.1.6 Location

Class in package 'LogicView'

Location entity will model all locations that were and will be used to host games.

Constraints:

ValidateCountry: each country must be 3 a 3 letter string following ISO 3166-1 alfa-3 standard

Location
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/10/2017
Extends AbstractAuditing

CONSTRAINTS	
◆ OCL. ValidateCountry	
inv ValidateCountry: self.country->length = 3	[Approved, Weight is 0.]

OUTGOING STRUCTURAL RELATIONSHIPS	
↳ Generalization from Location to AbstractAuditing	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
◆ city : string Private	[Is static False. Containment is Not Specified.]
◆ country : string Private	[Is static False. Containment is Not Specified.]
◆ name : string Private	[Is static False. Containment is Not Specified.]
◆ postalCode : string Private	[Is static False. Containment is Not Specified.]
◆ streetAddress : string Private	[Is static False. Containment is Not Specified.]

ASSOCIATIONS	
--------------	--

ASSOCIATIONS	
Association (direction: Unspecified) has	
Source: Public (Class) Game Cardinality: [0..*]	Target: Public (Class) Location Cardinality: [1]

1.1.7 User

Class in package 'LogicView'

User
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/10/2017
Extends AbstractAuditing

OUTGOING STRUCTURAL RELATIONSHIPS	
Generalization from User to AbstractAuditing	[Direction is 'Source -> Destination'.]

ATTRIBUTES	
activated : boolean Private	[Is static False. Containment is Not Specified.]
activation_key : string Private	[Is static False. Containment is Not Specified.]
authorities : int Private	[Is static False. Containment is Not Specified.]
email : string Private	[Is static False. Containment is Not Specified.]
first_name : string Private	[Is static False. Containment is Not Specified.]
image_url : string Private	[Is static False. Containment is Not Specified.]
last_name : string Private	[Is static False. Containment is Not Specified.]
password_hash : string Private	[Is static False. Containment is Not Specified.]
reset_date : datetime Private	[Is static False. Containment is Not Specified.]
reset_key : string Private	[Is static False. Containment is Not Specified.]

ATTRIBUTES
◆ username : string Private [Is static False. Containment is Not Specified.]
ASSOCIATIONS
✓ Association (direction: Unspecified) composed by Source: Public (Class) Game Cardinality: [0..*] Target: Public (Class) User Cardinality: [0..*]
OPERATIONS
◆ activateAccount () : void Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ changePassword () : void Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]
◆ passwordRecovery () : string Public [Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False.]

1.1.8 GameStatus

Enumeration in package 'LogicView'

GameStatus
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/10/2017

ATTRIBUTES
◆ ACTIVE : Public [Stereotype is «enum». Is static False. Containment is Not Specified.]
◆ CANCELLED : Public [Stereotype is «enum». Is static False. Containment is Not Specified.]
◆ FINISHED : Public [Stereotype is «enum». Is static False. Containment is Not Specified.]
◆ ARCHIEVED : Public [Stereotype is «enum». Is static False. Containment is Not Specified.]
◆ FULL : Public [Stereotype is «enum». Is static False. Containment is Not Specified.]

1.1.9 TypeOfGame

Enumeration in package 'LogicView'

TypeOfGame
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 11/3/2017. Last modified 11/10/2017

ATTRIBUTES	
◆ SPORT : Public	[Stereotype is «enum». Is static False. Containment is Not Specified.]
◆ ESPORT : Public	[Stereotype is «enum». Is static False. Containment is Not Specified.]
◆ BOARD : Public	[Stereotype is «enum». Is static False. Containment is Not Specified.]

1.2 Use Case View

Package in package 'Model'

Use Case View
Version Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

1.2.1 Create Game Activity Diagram diagram

Activity diagram in package 'Use Case View'

Create Game Activity Diagram
Version 1.0
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

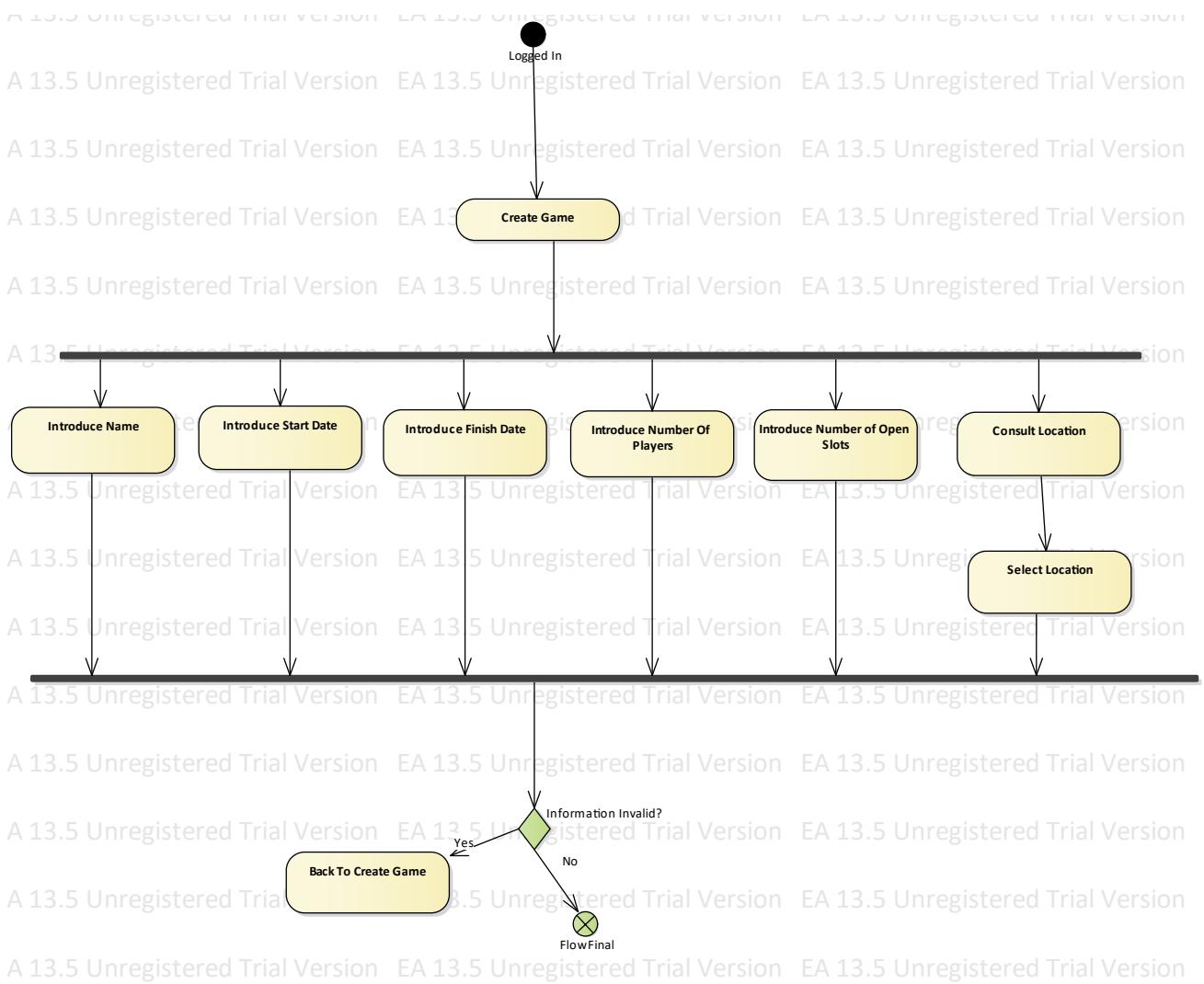


Figure 4: Create Game Activity Diagram

1.2.2 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS

Name: No

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Information Invalid? to FlowFinal

1.2.3 Logged In

ActivityInitial in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Logged In to Create Game

1.2.4 Synchronization

Synchronization in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Synchronization to Consult Location

⇐ ControlFlow from Synchronization to Introduce Start Date

⇐ ControlFlow from Synchronization to Introduce Number of Open Slots

⇐ ControlFlow from Synchronization to Introduce Number Of Players

⇐ ControlFlow from Synchronization to Introduce Name

⇐ ControlFlow from Synchronization to Introduce Finish Date

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Create Game to Synchronization

1.2.5 Synchronization

Synchronization in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Synchronization to Information Invalid?

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Introduce Finish Date to Synchronization

INCOMING BEHAVIORAL RELATIONSHIPS
⇒ ControlFlow from Introduce Name to Synchronization
⇒ ControlFlow from Introduce Number of Open Slots to Synchronization
⇒ Object Flow from ObjectNode2 to Synchronization
⇒ ControlFlow from Select Location to Synchronization
⇒ ControlFlow from Introduce Number Of Players to Synchronization
⇒ ControlFlow from Introduce Start Date to Synchronization

1.2.6 Game Sequence Diagram diagram

Interaction diagram in package 'Use Case View'

Game Sequence Diagram
Version 1.0
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

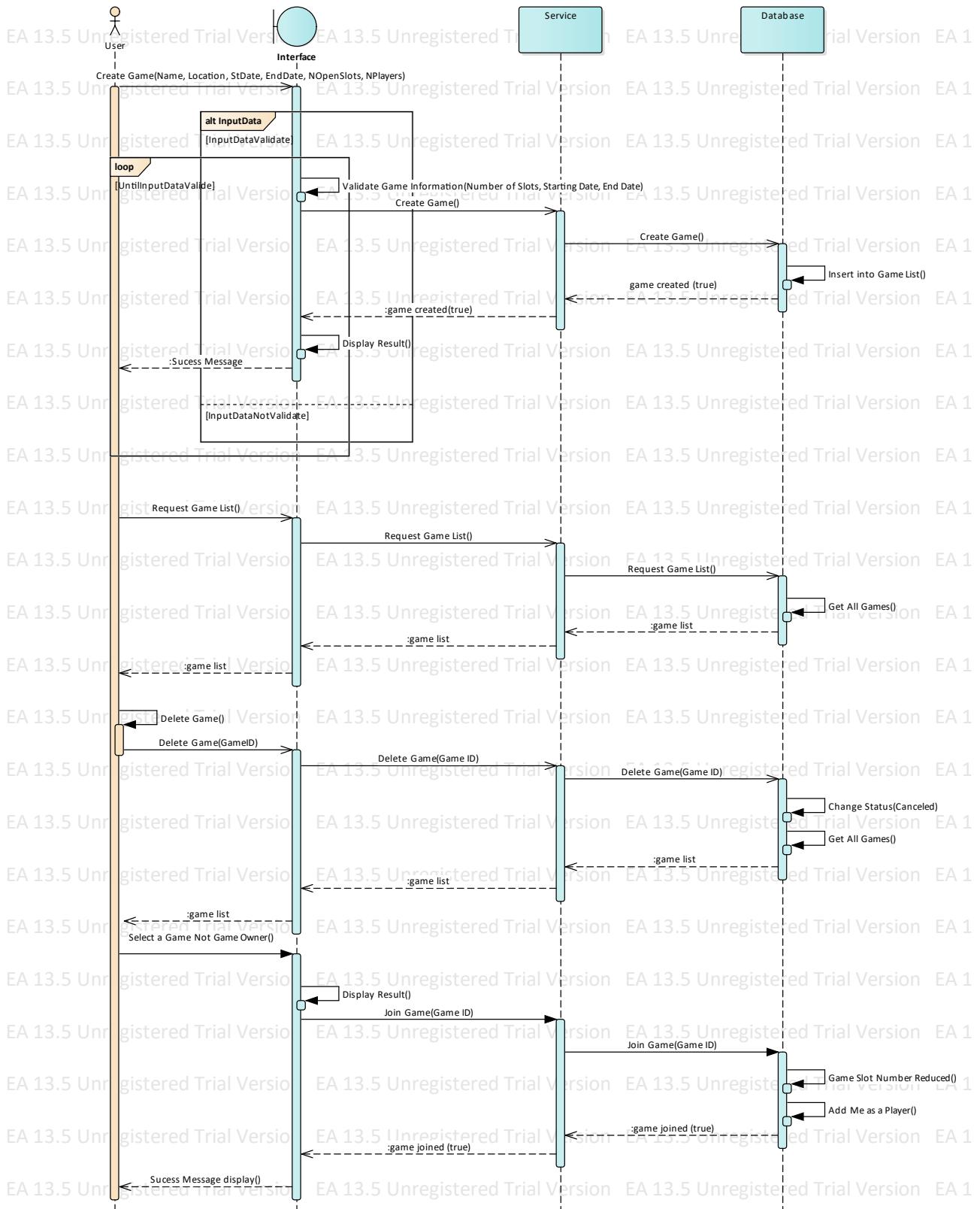


Figure 5: Game Sequence Diagram

INTERACTION MESSAGES

✉ 1.0 'Create Game' from 'User' sent to 'Interface'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.1 'Validate Game Information' from 'Interface' sent to 'Interface'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.2 'Create Game' from 'Interface' sent to 'Service'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.3 'Create Game' from 'Service' sent to 'Database'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.4 'Insert into Game List' from 'Database' sent to 'Database'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.5 'game created (true)' from 'Database' sent to 'Service'.

Asynchronous Call. Returns void.

[Return is True. Iteration is False. New group is False.]

✉ 1.6 '' from 'Service' sent to 'Interface'.

Asynchronous Call. Returns game created(true).

[Return is True. Iteration is False. New group is False.]

✉ 1.7 'Display Result' from 'Interface' sent to 'Interface'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.8 '' from 'Interface' sent to 'User'.

Asynchronous Call. Returns Success Message.

[Return is True. Iteration is False. New group is False.]

✉ 1.9 'Request Game List()' from 'User' sent to 'Interface'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.10 'Request Game List' from 'Interface' sent to 'Service'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.11 'Request Game List' from 'Service' sent to 'Database'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.12 'Get All Games' from 'Database' sent to 'Database'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.13 '' from 'Database' sent to 'Service'.

Asynchronous Call. Returns game list.

[Return is True. Iteration is False. New group is False.]

✉ 1.14 '' from 'Service' sent to 'Interface'.

Asynchronous Call. Returns game list.

[Return is True. Iteration is False. New group is False.]

✉ 1.15 '' from 'Interface' sent to 'User'.

Asynchronous Call. Returns game list.

[Return is True. Iteration is False. New group is False.]

✉ 1.16 'Delete Game' from 'User' sent to 'User'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.17 'Delete Game' from 'User' sent to 'Interface'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.18 'Delete Game' from 'Interface' sent to 'Service'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.19 'Delete Game' from 'Service' sent to 'Database'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.20 'Change Status' from 'Database' sent to 'Database'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.21 'Get All Games' from 'Database' sent to 'Database'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.22 '' from 'Database' sent to 'Service'.

Synchronous Call. Returns game list.

[Return is True. Iteration is False. New group is False.]

<p>✉ 1.23 " from 'Service' sent to 'Interface'.</p>
<p>Asynchronous Call. Returns game list. [Return is True. Iteration is False. New group is False.]</p>
<p>✉ 1.24 " from 'Interface' sent to 'User'.</p>
<p>Asynchronous Call. Returns game list. [Return is True. Iteration is False. New group is False.]</p>
<p>✉ 1.25 'Select a Game Not Game Owner' from 'User' sent to 'Interface'.</p>
<p>Synchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]</p>
<p>✉ 1.26 'Display Result' from 'Interface' sent to 'Interface'.</p>
<p>Synchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]</p>
<p>✉ 1.27 'Join Game' from 'Interface' sent to 'Service'.</p>
<p>Synchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]</p>
<p>✉ 1.28 'Join Game' from 'Service' sent to 'Database'.</p>
<p>Synchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]</p>
<p>✉ 1.29 'Game Slot Number Reduced' from 'Database' sent to 'Database'.</p>
<p>Synchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]</p>
<p>✉ 1.30 'Add Me as a Player' from 'Database' sent to 'Database'.</p>
<p>Synchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]</p>
<p>✉ 1.31 " from 'Database' sent to 'Service'.</p>
<p>Asynchronous Call. Returns game joined (true). [Return is True. Iteration is False. New group is False.]</p>
<p>✉ 1.32 " from 'Service' sent to 'Interface'.</p>
<p>Asynchronous Call. Returns game joined (true). [Return is True. Iteration is False. New group is False.]</p>
<p>✉ 1.33 'Sucess Message display' from 'Interface' sent to 'User'.</p>
<p>Asynchronous Call. Returns void. [Return is True. Iteration is False. New group is False.]</p>

1.2.7 Select Game Activity Diagram diagram

Activity diagram in package 'Use Case View'

Select Game Activity Diagram
Version 1.0

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

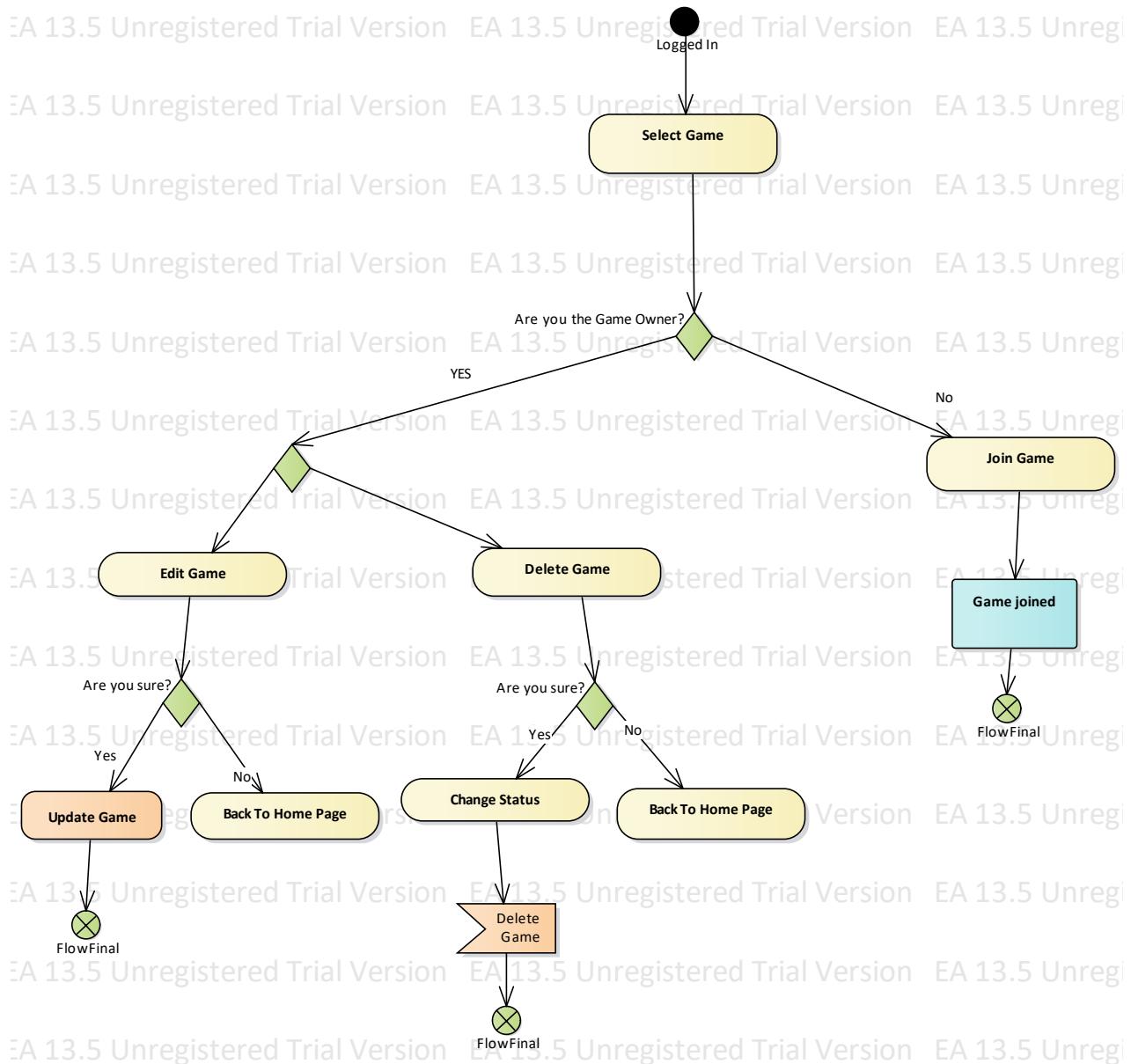


Figure 6: Select Game Activity Diagram

1.2.8 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS

Object Flow from Game joined to FlowFinal

1.2.9 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS
➡ ControlFlow from Delete Game to FlowFinal

1.2.10 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS
➡ ControlFlow from Update Game to FlowFinal

1.2.11 Logged In

ActivityInitial in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS
➡ ControlFlow from Logged In to Select Game

1.2.12 Use Case Diagram diagram

Use Case diagram in package 'Use Case View'

Use Case Diagram
Version 1.0

Luís Melo created on 10/27/2017. Last modified 11/5/2017

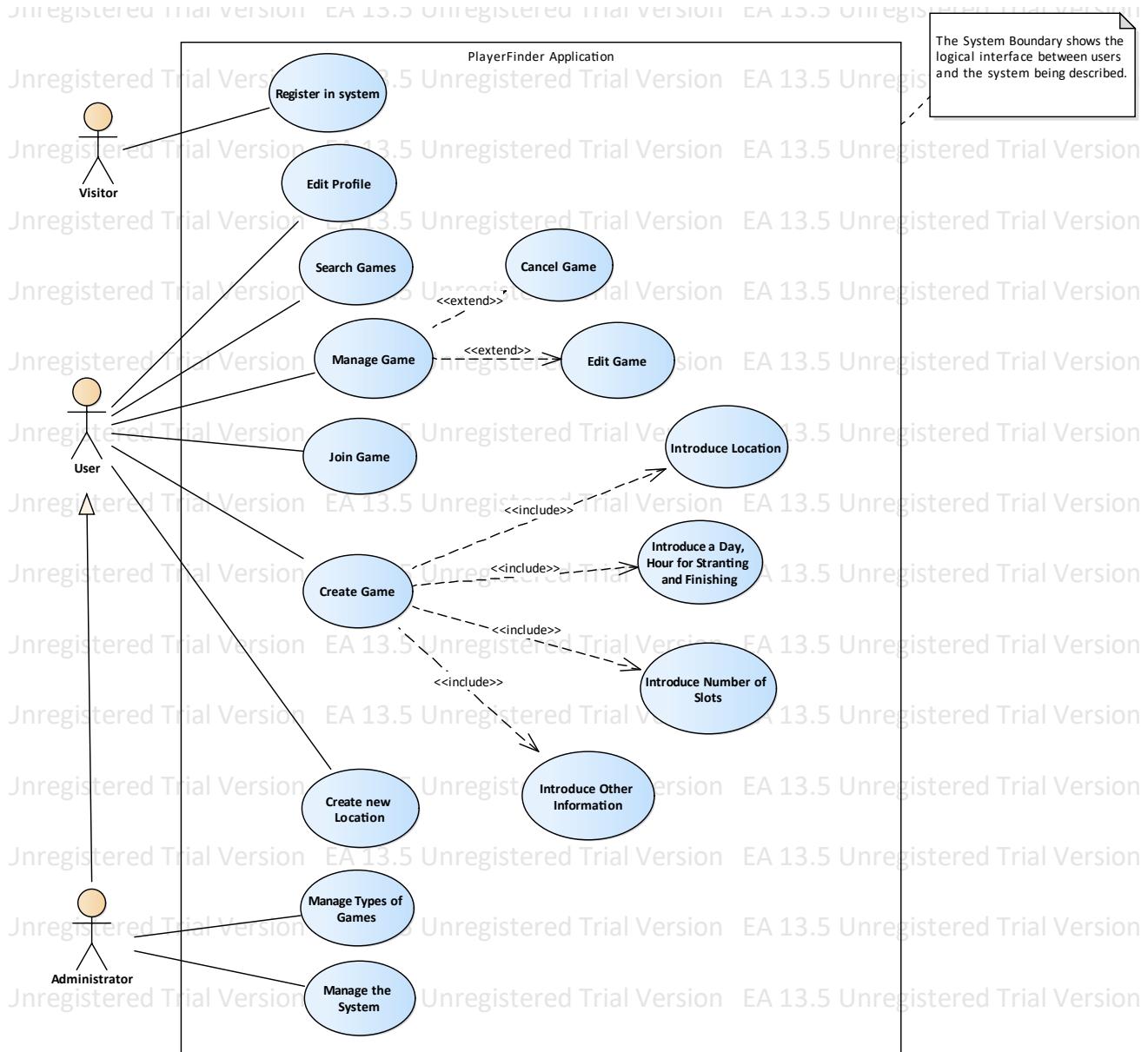


Figure 7: Use Case Diagram

1.2.13 PlayerFinder Application

Boundary in package 'Use Case View'

PlayerFinder Application
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017
Extends

1.2.14 Note

Note in package 'Use Case View'

The System Boundary shows the logical interface between users and the system being described.

Note
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017
Extends

1.2.15 Administrator

Actor in package 'Actors'

Responsible for managing the whole system.

Administrator

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Generalization from Administrator to User

[Direction is 'Source -> Destination'.]

ASSOCIATIONS

✍ Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage Types of Games

✍ Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage the System

1.2.16 User

Actor in package 'Actors'

A person that has registered in the application. Can join or create games.

User

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ Sequence from User to Object1

INCOMING STRUCTURAL RELATIONSHIPS

➡ Generalization from Administrator to User

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ UseCaseLink Source -> Destination

From: User : Actor, Public

To: Login in platform : UseCase, Public

ASSOCIATIONS

ASSOCIATIONS	
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Manage Game
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Edit Profile
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Search Games
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Logout
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Join Game
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Create new Location
✓ Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Create Game

1.2.17 Visitor

Actor in package 'Actors'

A person who is not yet registered and has access only to register, login and home page. He has not access to any functionality of the platform.

Visitor
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

ASSOCIATIONS	
✓ Association (direction: Unspecified)	
Source: Public (Actor) Visitor	Target: Public (UseCase) Register in system

1.2.18 Cancel Game

UseCase in package 'Primary Use Cases'

If the User who created the game wants to cancel a already game created, for any reason such as weather conditions, he/she can cancel it.

Cancel Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

 **Dependency <<extend>>** Source -> Destination

From: Manage Game : UseCase, Public
To: Cancel Game : UseCase, Public

1.2.19 Create Game

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the user can follow the mandatory steps for creating a game:

- 1 - Introduce a Location
- 2 - Introduce a Day and Hour for starting and finishing the game
- 3- Introduce the Number of Slots
- 4 - Introduce other information such as name, total of players.

Create Game

Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Create Game

 Create Game_SequenceDiagram : Interaction

SCENARIOS

 Basic Path. Basic Path

1. The User navigates to the System's game creation page
2. The System shows to the User the game creation form
3. The User inserts in the System the game's data
4. The User clicks in the submit button on the System
5. The System shows to the User the success of the operation
6. The System shows to the User the list of all games

CONSTRAINTS

 Invariant. Pre-condition(s)

Valid credentials

At least one location created.

[Approved, Weight is 0.]

CONNECTORS

 **Dependency <<include>>** Source -> Destination

From: Create Game : UseCase, Public
To: Introduce Number of Slots : UseCase, Public

CONNECTORS

 **Dependency <<include>>** Source -> Destination

From: Create Game : UseCase, Public
To: Introduce Location : UseCase, Public

 **Dependency <<include>>** Source -> Destination

From: Create Game : UseCase, Public
To: Introduce a Day, Hour for Stranting and Finishing : UseCase, Public

 **Dependency <<include>>** Source -> Destination

From: Create Game : UseCase, Public
To: Introduce Other Information : UseCase, Public

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Create Game

1.2.19.1 Create Game_SequenceDiagram

Interaction owned by 'Create Game', in package 'Primary Use Cases'

Create Game_SequenceDiagram

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

ELEMENTS OWNED BY Create Game_SequenceDiagram

 System : Sequence

 User : Sequence

1.2.19.1.1 System

Sequence owned by 'Create Game_SequenceDiagram', in package 'Primary Use Cases'

System

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: 5.The System shows to the User the success of the operation

 Sequence from System to User

Name: 6.The System shows to the User the list of all games

 Sequence from System to User

Name: 2.The System shows to the User the game creation form

 Sequence from System to User

OUTGOING BEHAVIORAL RELATIONSHIPS
INCOMING BEHAVIORAL RELATIONSHIPS
<p>Name: 3.The User inserts in the System the game's data  Sequence from User to System</p> <p>Name: 1.The User navigates to the System's game creation page  Sequence from User to System</p> <p>Name: 4.The User clicks in the submit button on the System  Sequence from User to System</p>

1.2.19.1.2 User

Sequence owned by 'Create Game_SequenceDiagram', in package 'Primary Use Cases'

User
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
<p>Name: 3.The User inserts in the System the game's data  Sequence from User to System</p> <p>Name: 1.The User navigates to the System's game creation page  Sequence from User to System</p> <p>Name: 4.The User clicks in the submit button on the System  Sequence from User to System</p>
INCOMING BEHAVIORAL RELATIONSHIPS
<p>Name: 5.The System shows to the User the success of the operation  Sequence from System to User</p> <p>Name: 6.The System shows to the User the list of all games  Sequence from System to User</p> <p>Name: 2.The System shows to the User the game creation form  Sequence from System to User</p>

1.2.20 Create new Location

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the user can create a new location for game purpose.

Create new Location

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Create new Location

1.2.21 Edit Game

UseCase in package 'Primary Use Cases'

This Use Case will occur if a doctor wants to visualize some games already created or updated any information about the game.

Edit Game

Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

Dependency <>extend>> Source -> Destination

From: Manage Game : UseCase, Public
To: Edit Game : UseCase, Public

1.2.22 Edit Profile

UseCase in package 'Primary Use Cases'

After login, the user information can be edit and updated.

Edit Profile

Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Edit Profile

1.2.23 Introduce a Day, Hour for Stranting and Finishing

UseCase in package 'Primary Use Cases'

Information about the day and hour for starting and finishing the game.

Introduce a Day, Hour for Stranting and Finishing

Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

CONNECTORS

↗ **Dependency <<include>>** Source -> Destination
From: Create Game : UseCase, Public
To: Introduce a Day, Hour for Stranting and Finishing : UseCase, Public

1.2.24 Introduce Location

UseCase in package 'Primary Use Cases'

Information about the location where the game is taking place.

Introduce Location
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

↗ **Dependency <<include>>** Source -> Destination
From: Create Game : UseCase, Public
To: Introduce Location : UseCase, Public

1.2.25 Introduce Number of Slots

UseCase in package 'Primary Use Cases'

Information about the number of open slots for the game.

Introduce Number of Slots
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

↗ **Dependency <<include>>** Source -> Destination
From: Create Game : UseCase, Public
To: Introduce Number of Slots : UseCase, Public

1.2.26 Introduce Other Information

UseCase in package 'Primary Use Cases'

Information about the name and total of players.

Introduce Other Information
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

CONNECTORS



 **Dependency** <<include>> Source -> Destination
From: Create Game : UseCase, Public
To: Introduce Other Information : UseCase, Public

1.2.27 Join Game

UseCase in package 'Primary Use Cases'

User can access the games created by others in real time and join a game if we wanted.

Join Game
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS



 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Join Game

1.2.28 Manage Game

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the user can manage the game we created.

Manage Game
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

CONSTRAINTS



Invariant. Pre-Condition(s)

At least one game created.
Valid credentials

[Approved, Weight is 0.]

CONNECTORS



Dependency <<extend>> Source -> Destination
From: Manage Game : UseCase, Public
To: Edit Game : UseCase, Public



 **Dependency** <<extend>> Source -> Destination
From: Manage Game : UseCase, Public
To: Cancel Game : UseCase, Public

ASSOCIATIONS



 Association (direction: Unspecified)

ASSOCIATIONS

Source: Public (Actor) User

Target: Public (UseCase) Manage Game

1.2.29 Manage the System

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the administrator can manage the system such as performance, metrics, etc.

Manage the System

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage the System

1.2.30 Manage Types of Games

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the administrator can manage different types of games, such as football, basket, etc.

Manage Types of Games

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage Types of Games

1.2.31 Register in system

UseCase in package 'Primary Use Cases'

This Use Case will occur if a visitor wants register in the application and use it.

Register in system

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Register in system

Register in system social Account : Activity

Register in system Social Account : Activity

ELEMENTS OWNED BY Register in system	
■	Register in system_SequenceDiagram : Interaction
■	Object1 : Sequence

SCENARIOS

Basic Path. Normal Scenario

The user inputs manually all the information needed to register in the system.

1. The visitor access application in the system
2. The system shows the home page
3. The visitor clicks in register button on the system
4. The system shows register form
5. Visitor insert username, email and password into the System
 - Alternate: 5a. Social Register
 - Exception: 5b. User inputs invalid data
6. System shows success message

Alternate. Social Register

1. Click the register social account button

Exception. User inputs invalid data

When registering in the system if the passwords don't match the system shows a message error.

1. The system shows the visitor an error message.

ASSOCIATIONS

1.2.31.1 Register in system social Account

Activity owned by 'Register in system', in package 'Primary Use Cases'

Register in system social Account
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Register in system social Account

- Click register with some social account : Activity

- End : ActivityFinal

- Start : ActivityInitial

1.2.31.1.1 Click register with some social account

Activity owned by 'Register in system social Account', in package 'Primary Use Cases'

Click register with some social account

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Click register with some social account to End

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Start to Click register with some social account

1.2.31.1.2 End

ActivityFinal owned by 'Register in system social Account', in package 'Primary Use Cases'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Click register with some social account to End

1.2.31.1.3 Start

ActivityInitial owned by 'Register in system social Account', in package 'Primary Use Cases'

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Start to Click register with some social account

1.2.31.2 Register in system Social Account

Activity owned by 'Register in system', in package 'Primary Use Cases'

Register in system Social Account

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Register in system Social Account

█ Click register with some social account : Action «RuleTask»

█ End : ActivityFinal

█ Start : ActivityInitial

1.2.31.2.1 Click register with some social account

Action «RuleTask» owned by 'Register in system Social Account', in package 'Primary Use Cases'

Click register with some social account

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from «RuleTask» Click register with some social account to End

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Start to «RuleTask» Click register with some social account

1.2.31.2.2 End

ActivityFinal owned by 'Register in system Social Account', in package 'Primary Use Cases'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from «RuleTask» Click register with some social account to End

1.2.31.2.3 Start

ActivityInitial owned by 'Register in system Social Account', in package 'Primary Use Cases'

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Start to «RuleTask» Click register with some social account

1.2.31.3 Register in system_SequenceDiagram

Interaction owned by 'Register in system', in package 'Primary Use Cases'

Register in system_SequenceDiagram

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

ELEMENTS OWNED BY Register in system_SequenceDiagram

█ System : Sequence

█ System : Sequence

ELEMENTS OWNED BY Register in system_SequenceDiagram	
 System12 : Sequence	
 Visitor : Sequence	

1.2.31.3.1 System

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.31.3.2 System

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
<p>Name: 1.The system shows the visitor an error message.  Sequence from System to Visitor</p>

1.2.31.3.3 System12

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

System12
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.31.3.4 Visitor

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

Visitor
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

INCOMING BEHAVIORAL RELATIONSHIPS
<p>Name: 1.The system shows the visitor an error message.  Sequence from System to Visitor</p>

1.2.31.4 Object1

Sequence owned by 'Register in system', in package 'Primary Use Cases'

Object1

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

INCOMING BEHAVIORAL RELATIONSHIPS

➡ Sequence from User to Object1

1.2.32 Search Games

UseCase in package 'Primary Use Cases'

This Use Case will occur if a user wants to visualize some games already created.
Search will be searching in all database and results should be showing if something exist.

Search Games

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

↙ Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Search Games

1.2.33 Use Case Model diagram

Use Case diagram in package 'Use Case View'

Use Case Model

Version 1.0

Luís Melo created on 10/27/2017. Last modified 10/27/2017

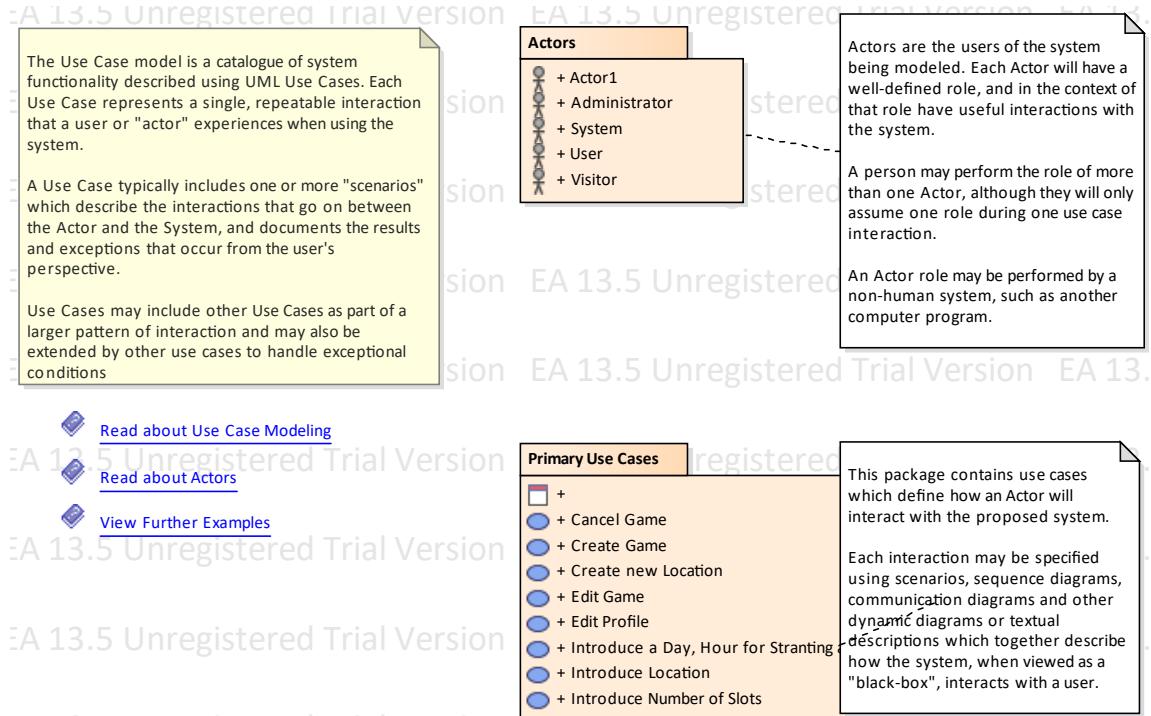


Figure 8: Use Case Model

1.2.34 Note

Note in package 'Use Case View'

Actors are the users of the system being modeled. Each Actor will have a well-defined role, and in the context of that role have useful interactions with the system.

A person may perform the role of more than one Actor, although they will only assume one role during one use case interaction.

An Actor role may be performed by a non-human system, such as another computer program.

Note
 Version 1.0 Phase 1.0 Proposed
 Luís Melo created on 10/27/2017. Last modified 10/27/2017
 Extends

1.2.35 Note

Note in package 'Use Case View'

This package contains use cases which define how an Actor will interact with the proposed system.

Each interaction may be specified using scenarios, sequence diagrams, communication diagrams and other dynamic diagrams or textual descriptions which together describe how the system, when viewed as a "black-box", interacts with a user.

Note
 Version 1.0 Phase 1.0 Proposed
 Luís Melo created on 10/27/2017. Last modified 10/27/2017
 Extends

1.2.36 Note

Note in package 'Use Case View'

The Use Case model is a catalogue of system functionality described using UML Use Cases. Each Use Case represents a single, repeatable interaction that a user or "actor" experiences when using the system.

A Use Case typically includes one or more "scenarios" which describe the interactions that go on between the Actor and the System, and documents the results and exceptions that occur from the user's perspective.

Use Cases may include other Use Cases as part of a larger pattern of interaction and may also be extended by other use cases to handle exceptional conditions

Note
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017
Extends

1.2.37 Actors

Package in package 'Use Case View'

Actors
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.38 Primary Use Cases

Package in package 'Use Case View'

Primary Use Cases
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.39 \$help://actor.htm

Text in package 'Use Case View'

\$help://actor.htm
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017
Alias Read about Actors
Extends

1.2.40 \$help://use_case_model_pattern.htm

Text in package 'Use Case View'

\$help://use_case_model_pattern.htm
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017
Alias Read about Use Case Modeling
Extends

1.2.41 \$help://usecasediagram.htm

Text in package 'Use Case View'

\$help://usecasediagram.htm
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017
Alias View Further Examples
Extends

1.2.42 Visitor Login Sequence Diagram diagram

Interaction diagram in package 'Use Case View'

Visitor Login Sequence Diagram
Version 1.0
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

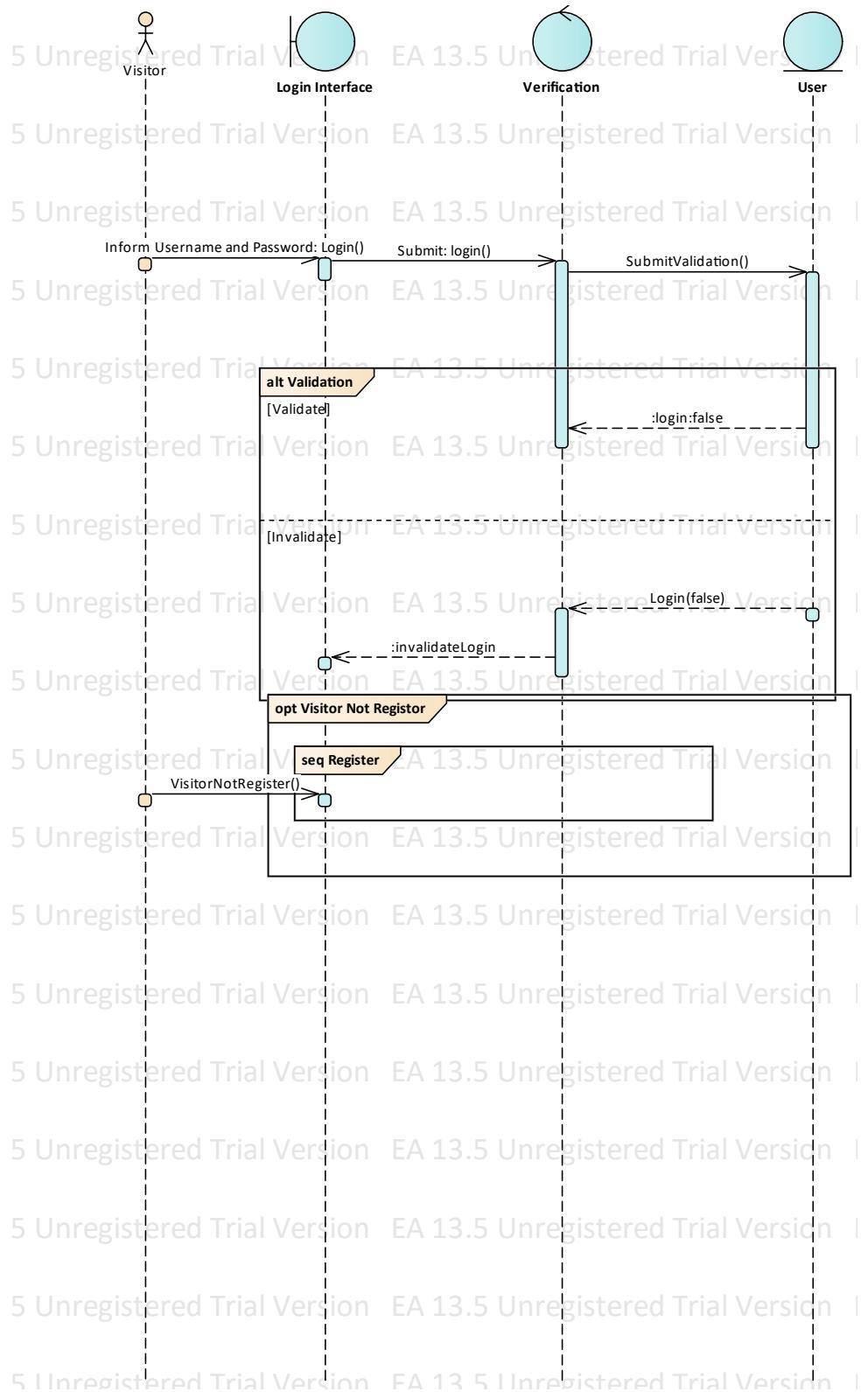


Figure 9: Visitor Login Sequence Diagram

INTERACTION MESSAGES

1.0 'Inform Username and Password: Login' from 'Visitor' sent to 'Login Interface'.

Asynchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

✉ 1.1 'Submit: login()' from 'Login Interface' sent to 'Verification'.
Asynchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]
✉ 1.2 'SubmitValidation()' from 'Verification' sent to 'User'.
Asynchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]
✉ 1.3 '' from 'User' sent to 'Verification'.
Asynchronous Call. Returns login:false. [Return is True. Iteration is False. New group is False.]
✉ 1.4 'Login' from 'User' sent to 'Verification'.
Asynchronous Call. Returns void. [Return is True. Iteration is False. New group is False.]
✉ 1.5 '' from 'Verification' sent to 'Login Interface'.
Asynchronous Call. Returns invalidateLogin. [Return is True. Iteration is False. New group is False.]
✉ 1.6 'VisitorNotRegister' from 'Visitor' sent to 'Login Interface'.
Asynchronous Call. Returns void. [Return is False. Iteration is False. New group is False.]

1.2.43 Actors

Package in package 'Use Case View'

Actors
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.43.1 Actors diagram

Use Case diagram in package 'Actors'

Actors: Administrator, User and Visitor

Actors
Version 1.0
Luís Melo created on 10/27/2017. Last modified 10/27/2017

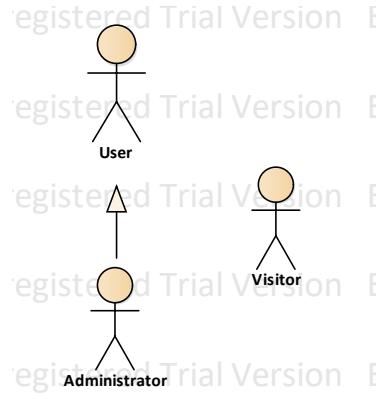


Figure 10: Actors

1.2.43.2 Actor1

Actor in package 'Actors'

Actor1
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.43.3 Administrator

Actor in package 'Actors'

Responsible for managing the whole system.

Administrator
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING STRUCTURAL RELATIONSHIPS

↳ Generalization from Administrator to User

[Direction is 'Source -> Destination'.]

ASSOCIATIONS

↙ Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage Types of Games

↙ Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage the System

1.2.43.4 System

Actor in package 'Actors'

System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.43.5 User*Actor in package 'Actors'*

A person that has registered in the application. Can join or create games.

User
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

 Sequence from User to Object1

INCOMING STRUCTURAL RELATIONSHIPS

 Generalization from Administrator to User

[Direction is 'Source -> Destination'.]

CONNECTORS

 **UseCaseLink** Source -> Destination

From: User : Actor, Public

To: Login in platform : UseCase, Public

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Manage Game

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Edit Profile

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Search Games

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Logout

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Join Game

 Association (direction: Unspecified)

ASSOCIATIONS	
Source: Public (Actor) User	Target: Public (UseCase) Create new Location
Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Create Game

1.2.43.6 Visitor

Actor in package 'Actors'

A person who is not yet registered and has access only to register, login and home page. He has not access to any functionality of the platform.

Visitor
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

ASSOCIATIONS	
Association (direction: Unspecified)	
Source: Public (Actor) Visitor	Target: Public (UseCase) Register in system

1.2.44 Primary Use Cases

Package in package 'Use Case View'

Primary Use Cases

Version 1.0 Phase 1.0 Proposed

Luis Melo created on 10/27/2017. Last modified 10/27/2017

1.2.44.1 Cancel Game

UseCase in package 'Primary Use Cases'

If the User who created the game wants to cancel a already game created, for any reason such as weather conditions, he/she can cancel it.

Cancel Game

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

 **Dependency <<extend>>** Source -> Destination
 From: Manage Game : UseCase, Public
 To: Cancel Game : UseCase, Public

1.2.44.2 Create Game

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the user can follow the mandatory steps for creating a game:

- 1 - Introduce a Location
- 2 - Introduce a Day and Hour for starting and finishing the game
- 3- Introduce the Number of Slots
- 4 - Introduce other information such as name, total of players.

Create Game

Version 1.0 Phase 1.0 Proposed

Luis Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Create Game

 Create Game_SequenceDiagram : Interaction

SCENARIOS

 Basic Path. Basic Path

1. The User navigates to the System's game creation page
2. The System shows to the User the game creation form
3. The User inserts in the System the game's data
4. The User clicks in the submit button on the System
5. The System shows to the User the success of the operation
6. The System shows to the User the list of all games

CONSTRAINTS

CONSTRAINTS

 Invariant. Pre-condition(s)

Valid credentials
At least one location created.

[Approved, Weight is 0.]

CONNECTORS

 **Dependency** <<include>> Source -> Destination
From: Create Game : UseCase, Public
To: Introduce Number of Slots : UseCase, Public

 **Dependency** <<include>> Source -> Destination
From: Create Game : UseCase, Public
To: Introduce Location : UseCase, Public

 **Dependency** <<include>> Source -> Destination
From: Create Game : UseCase, Public
To: Introduce a Day, Hour for Stranting and Finishing : UseCase, Public

 **Dependency** <<include>> Source -> Destination
From: Create Game : UseCase, Public
To: Introduce Other Information : UseCase, Public

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Create Game

1.2.44.2.1 Create Game_SequenceDiagram

Interaction owned by 'Create Game', in package 'Primary Use Cases'

Create Game_SequenceDiagram
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

ELEMENTS OWNED BY Create Game_SequenceDiagram

 System : Sequence

 User : Sequence

1.2.44.2.1.1 CreateGame_Sequence diagram

Interaction diagram in package 'Primary Use Cases'

CreateGame_Sequence
Version 1.0

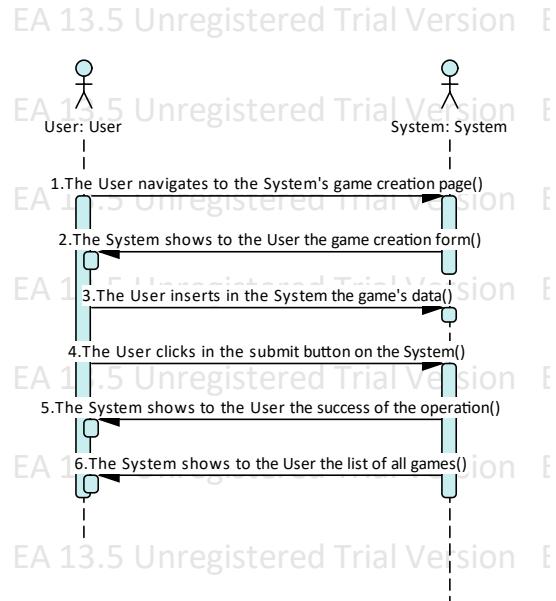


Figure 11: CreateGame_Sequence

INTERACTION MESSAGES

- ✉ 1.0 '1.The User navigates to the System's game creation page'** from 'User: User' sent to 'System: System'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

- ✉ 1.1 '2.The System shows to the User the game creation form'** from 'System: System' sent to 'User: User'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

- ✉ 1.2 '3.The User inserts in the System the game's data'** from 'User: User' sent to 'System: System'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

- ✉ 1.3 '4.The User clicks in the submit button on the System'** from 'User: User' sent to 'System: System'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

- ✉ 1.4 '5.The System shows to the User the success of the operation'** from 'System: System' sent to 'User: User'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

- ✉ 1.5 '6.The System shows to the User the list of all games'** from 'System: System' sent to 'User: User'.

Synchronous Call. Returns void.

[Return is False. Iteration is False. New group is False.]

1.2.44.2.1.2 System

Sequence owned by 'Create Game_SequenceDiagram', in package 'Primary Use Cases'

System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: 5.The System shows to the User the success of the operation
 Sequence from System to User

Name: 6.The System shows to the User the list of all games
 Sequence from System to User

Name: 2.The System shows to the User the game creation form
 Sequence from System to User

INCOMING BEHAVIORAL RELATIONSHIPS

Name: 3.The User inserts in the System the game's data
 Sequence from User to System

Name: 1.The User navigates to the System's game creation page
 Sequence from User to System

Name: 4.The User clicks in the submit button on the System
 Sequence from User to System

1.2.44.2.1.3 User

Sequence owned by 'Create Game_SequenceDiagram', in package 'Primary Use Cases'

User
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: 3.The User inserts in the System the game's data
 Sequence from User to System

Name: 1.The User navigates to the System's game creation page
 Sequence from User to System

Name: 4.The User clicks in the submit button on the System
 Sequence from User to System

INCOMING BEHAVIORAL RELATIONSHIPS

Name: 5.The System shows to the User the success of the operation

INCOMING BEHAVIORAL RELATIONSHIPS	
 Sequence from System to User	
Name: 6.The System shows to the User the list of all games	
 Sequence from System to User	
Name: 2.The System shows to the User the game creation form	
 Sequence from System to User	

1.2.44.3 Create new Location

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the user can create a new location for game purpose.

Create new Location

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS	
 Association (direction: Unspecified)	
Source: Public (Actor) User	Target: Public (UseCase) Create new Location

1.2.44.4 Edit Game

UseCase in package 'Primary Use Cases'

This Use Case will occur if a doctor wants to visualize some games already created or updated any information about the game.

Edit Game

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS	
 Dependency <<extend>> Source -> Destination	

From: Manage Game : UseCase, Public

To: Edit Game : UseCase, Public

1.2.44.5 Edit Profile

UseCase in package 'Primary Use Cases'

After login, the user information can be edit and updated.

Edit Profile

Version 1.0 Phase 1.0 Proposed

Luis Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Edit Profile

1.2.44.6 Introduce Location

UseCase in package 'Primary Use Cases'

Information about the location where the game is taking place.

Introduce Location

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

 Dependency <<include>> Source -> Destination

From: Create Game : UseCase, Public

To: Introduce Location : UseCase, Public

1.2.44.7 Introduce Number of Slots

UseCase in package 'Primary Use Cases'

Information about the number of open slots for the game.

Introduce Number of Slots

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

 Dependency <<include>> Source -> Destination

From: Create Game : UseCase, Public

To: Introduce Number of Slots : UseCase, Public

1.2.44.8 Introduce Other Information

UseCase in package 'Primary Use Cases'

Information about the name and total of players.

Introduce Other Information

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS

 Dependency <<include>> Source -> Destination

From: Create Game : UseCase, Public

To: Introduce Other Information : UseCase, Public

1.2.44.9 Introduce a Day, Hour for Stranting and Finishing

UseCase in package 'Primary Use Cases'

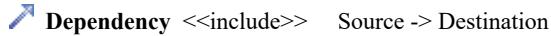
Information about the day and hour for starting and finishing the game.

Introduce a Day, Hour for Stranting and Finishing

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

CONNECTORS



Dependency <<include>> Source -> Destination

From: Create Game : UseCase, Public

To: Introduce a Day, Hour for Stranting and Finishing : UseCase, Public

1.2.44.10 Join Game

UseCase in package 'Primary Use Cases'

User can access the games created by others in real time and join a game if we wanted.

Join Game

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS



Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Join Game

1.2.44.11 Login in platform

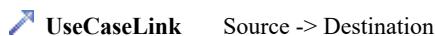
UseCase in package 'Primary Use Cases'

Login in platform

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

CONNECTORS



UseCaseLink Source -> Destination

From: User : Actor, Public

To: Login in platform : UseCase, Public

1.2.44.12 Logout

UseCase in package 'Primary Use Cases'

Logout

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Logout

1.2.44.13 Manage Game*UseCase in package 'Primary Use Cases'*

Use Case (U.S.) where the user can manage the game we created.

Manage Game

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

CONSTRAINTS

Invariant. Pre-Condition(s)

At least one game created.

Valid credentials

[Approved, Weight is 0.]

CONNECTORS

Dependency <<extend>> Source -> Destination

From: Manage Game : UseCase, Public

To: Edit Game : UseCase, Public

Dependency <<extend>> Source -> Destination

From: Manage Game : UseCase, Public

To: Cancel Game : UseCase, Public

ASSOCIATIONS

Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Manage Game

1.2.44.14 Manage Types of Games*UseCase in package 'Primary Use Cases'*

Use Case (U.S.) where the administrator can manage different types of games, such as football, basket, etc.

Manage Types of Games

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage Types of Games

1.2.44.15 Manage the System

UseCase in package 'Primary Use Cases'

Use Case (U.S.) where the administrator can manage the system such as performance, metrics, etc.

Manage the System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) Administrator

Target: Public (UseCase) Manage the System

1.2.44.16 Register in system

UseCase in package 'Primary Use Cases'

This Use Case will occur if a visitor wants register in the application and use it.

Register in system
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Register in system

 Register in system social Account : Activity

 Register in system Social Account : Activity

 Register in system_SequenceDiagram : Interaction

 Object1 : Sequence

SCENARIOS

 Basic Path. Normal Scenario

The user inputs manually all the information needed to register in the system.

1. The visitor access application in the system
 2. The system shows the home page
 3. The visitor clicks in register button on the system
 4. The system shows register form
 5. Visitor insert username, email and password into the System
- Alternate: 5a. Social Register

SCENARIOS
Exception: 5b. User inputs invalid data 6. System shows success message
Alternate. Social Register 1. Click the register social account button
Exception. User inputs invalid data When registering in the system if the passwords don't match the system shows a message error. 1. The system shows the visitor an error message.
ASSOCIATIONS
Association (direction: Unspecified) Source: Public (Actor) Visitor Target: Public (UseCase) Register in system

1.2.44.16.1 Use Case1 diagram

Interaction diagram in package 'Primary Use Cases'

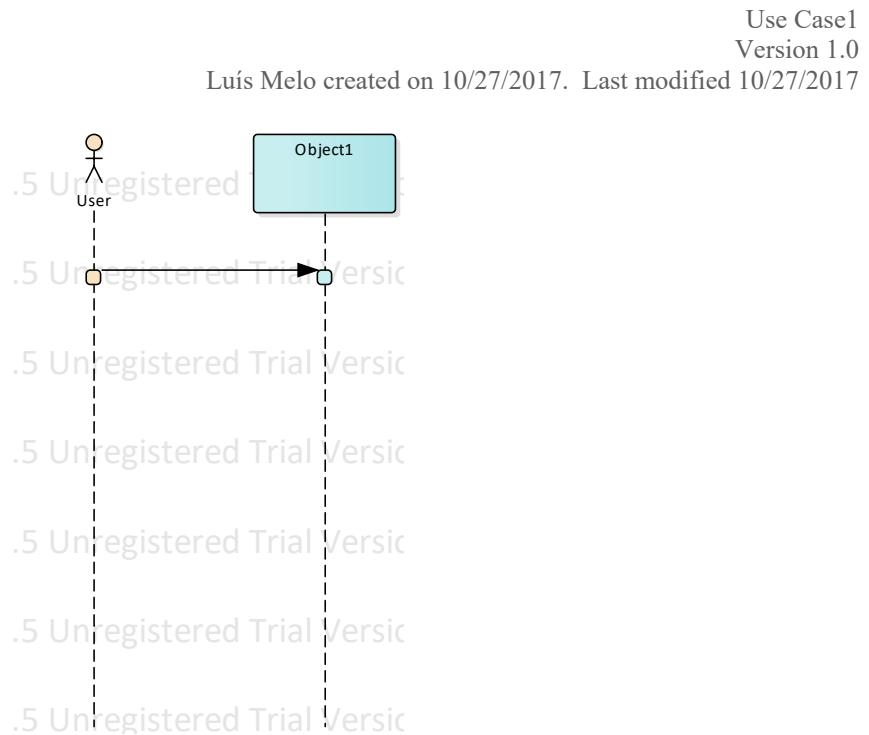


Figure 12: Use Case1

1.2.44.16.2 User

Actor in package 'Actors'

A person that has registered in the application. Can join or create games.

User
Version 1.0 Phase 1.0 Proposed

OUTGOING BEHAVIORAL RELATIONSHIPS Sequence from User to Object1**INCOMING STRUCTURAL RELATIONSHIPS** Generalization from Administrator to User

[Direction is 'Source -> Destination'.]

CONNECTORS **UseCaseLink** Source -> Destination

From: User : Actor, Public

To: Login in platform : UseCase, Public

ASSOCIATIONS Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Manage Game

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Edit Profile

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Search Games

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Logout

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Join Game

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Create new Location

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Create Game

INTERACTION MESSAGES

✉ 1.0 " from 'User' sent to 'Object1'.

Synchronous Call.

[Return is False. Iteration is False. New group is False.]

1.2.44.16.3 Register in system social Account

Activity owned by 'Register in system', in package 'Primary Use Cases'

Register in system social Account

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Register in system social Account

▣ Click register with some social account : Activity

▣ End : ActivityFinal

▣ Start : ActivityInitial

1.2.44.16.3.1 Register in system Social Account diagram

Activity diagram in package 'Primary Use Cases'

Register in system Social Account

Version 1.0

Luís Melo created on 10/27/2017. Last modified 11/5/2017

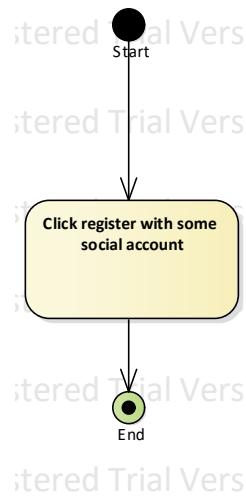


Figure 13: Register in system Social Account

1.2.44.16.3.2 Click register with some social account

Activity owned by 'Register in system social Account', in package 'Primary Use Cases'

Click register with some social account
 Version 1.0 Phase 1.0 Proposed
 Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Click register with some social account to End

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Start to Click register with some social account

1.2.44.16.3.3 End

ActivityFinal owned by 'Register in system social Account', in package 'Primary Use Cases'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Click register with some social account to End

1.2.44.16.3.4 Start

ActivityInitial owned by 'Register in system social Account', in package 'Primary Use Cases'

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Start to Click register with some social account

1.2.44.16.4 Register in system Social Account

Activity owned by 'Register in system', in package 'Primary Use Cases'

Register in system Social Account
 Version 1.0 Phase 1.0 Proposed
 Luís Melo created on 10/27/2017. Last modified 11/5/2017

ELEMENTS OWNED BY Register in system Social Account

█ Click register with some social account : Action «RuleTask»

█ End : ActivityFinal

█ Start : ActivityInitial

1.2.44.16.4.1 Register in system Social Account diagram

Activity diagram in package 'Primary Use Cases'

Register in system Social Account

Version 1.0

Luís Melo created on 10/27/2017. Last modified 11/5/2017

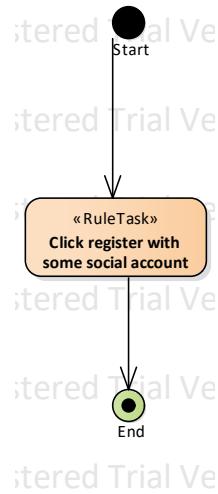


Figure 14: Register in system Social Account

1.2.44.16.4.2 Click register with some social account

Action «RuleTask» owned by 'Register in system Social Account', in package 'Primary Use Cases'

Click register with some social account

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from «RuleTask» Click register with some social account to End

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Start to «RuleTask» Click register with some social account

1.2.44.16.4.3 End

ActivityFinal owned by 'Register in system Social Account', in package 'Primary Use Cases'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from «RuleTask» Click register with some social account to End

1.2.44.16.4.4 Start

ActivityInitial owned by 'Register in system Social Account', in package 'Primary Use Cases'

OUTGOING BEHAVIORAL RELATIONSHIPS

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Start to «RuleTask» Click register with some social account

1.2.44.16.5 Register in system_SequenceDiagram

Interaction owned by 'Register in system', in package 'Primary Use Cases'

Register in system_SequenceDiagram

Version 1.0 Phase 1.0 Proposed

Luís Melo created on 10/27/2017. Last modified 10/27/2017

ELEMENTS OWNED BY Register in system_SequenceDiagram

System : Sequence

System : Sequence

System12 : Sequence

Visitor : Sequence

1.2.44.16.5.1 5b_User inputs invalid data diagram

Interaction diagram in package 'Primary Use Cases'

5b_User inputs invalid data

Version 1.0

Luís Melo created on 10/27/2017. Last modified 10/27/2017

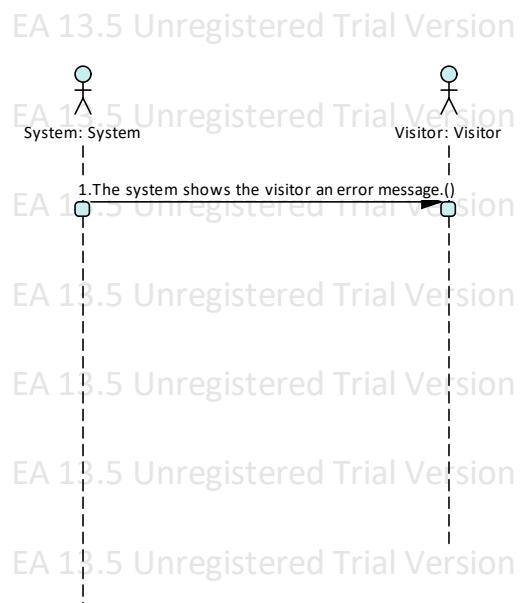


Figure 15: 5b_User inputs invalid data

INTERACTION MESSAGES
<p>✉ 1.0 '1.The system shows the visitor an error message.' from 'System: System' sent to 'Visitor: Visitor'.</p> <p>Synchronous Call. Returns void.</p> <p>[Return is False. Iteration is False. New group is False.]</p>

1.2.44.16.5.2 5a_Social Register diagram

Interaction diagram in package 'Primary Use Cases'

5a_Social Register
Version 1.0

Luís Melo created on 10/27/2017. Last modified 10/27/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 16: 5a_Social Register

1.2.44.16.5.3 1_Basic_Path diagram

Interaction diagram in package 'Primary Use Cases'

1_Basic_Path
Version 1.0

Luís Melo created on 10/27/2017. Last modified 10/27/2017



Figure 17: 1_Basic_Path

1.2.44.16.5.4 System

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.44.16.5.5 System

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

System
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: 1.The system shows the visitor an error message.
 Sequence from System to Visitor

1.2.44.16.5.6 System12

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

System12
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.44.16.5.7 Visitor

Sequence owned by 'Register in system_SequenceDiagram', in package 'Primary Use Cases'

Visitor
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

INCOMING BEHAVIORAL RELATIONSHIPS

Name: 1.The system shows the visitor an error message.
 Sequence from System to Visitor

1.2.44.16.6 Object1

Sequence owned by 'Register in system', in package 'Primary Use Cases'

Object1
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

INCOMING BEHAVIORAL RELATIONSHIPS

 Sequence from User to Object1

1.2.44.17 Search Games

UseCase in package 'Primary Use Cases'

This Use Case will occur if a user wants to visualize some games already created.
Search will be searching in all database and results should be showing if something exist.

Search Games
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 11/5/2017

ASSOCIATIONS

 Association (direction: Unspecified)

Source: Public (Actor) User

Target: Public (UseCase) Search Games

1.2.44.18 Object

Object in package 'Primary Use Cases'

Object
Version 1.0 Phase 1.0 Proposed
Luís Melo created on 10/27/2017. Last modified 10/27/2017

1.2.45 User

Actor in package 'Use Case View'

User
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Delete Game

 Sequence from User to User

Name: Request Game List()

 Sequence from User to «boundary» Interface

Name: Delete Game

 Sequence from User to «boundary» Interface

Name: Create Game

 Sequence from User to «boundary» Interface

Name: Select a Game Not Game Owner

 Sequence from User to «boundary» Interface

INCOMING BEHAVIORAL RELATIONSHIPS

Name: Delete Game

 Sequence from User to User

INCOMING BEHAVIORAL RELATIONSHIPS
➡ Sequence from «boundary» Interface to User
➡ Sequence from «boundary» Interface to User
➡ Sequence from «boundary» Interface to User
Name: Sucess Message display ➡ Sequence from «boundary» Interface to User

1.2.46 Visitor

Actor in package 'Use Case View'

Visitor

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
Name: VisitorNotRegister ↳ Sequence from Visitor to «boundary» Login Interface
Name: Inform Username and Password: Login ↳ Sequence from Visitor to «boundary» Login Interface

1.2.47 Game joined

Object in package 'Use Case View'

Game joined

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.48 Object1

Object in package 'Use Case View'

Object1

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.49 Activity1

Activity in package 'Use Case View'

Activity1

Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.50 Back To Create Game

Activity in package 'Use Case View'

Back To Create Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

INCOMING BEHAVIORAL RELATIONSHIPS

Name: Yes
⇒ ControlFlow from Information Invalid? to Back To Create Game

1.2.51 Back To Home Page

Activity in package 'Use Case View'

Back To Home Page
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

INCOMING BEHAVIORAL RELATIONSHIPS

Name: No
⇒ ControlFlow from Are you sure? to Back To Home Page

1.2.52 Back To Home Page

Activity in package 'Use Case View'

Back To Home Page
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

INCOMING BEHAVIORAL RELATIONSHIPS

Name: No
⇒ ControlFlow from Are you sure? to Back To Home Page

1.2.53 Change Information about the Game

Activity in package 'Use Case View'

Change Information about the Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.54 Change Status

Activity in package 'Use Case View'

Change Status
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Change Status to Delete Game

INCOMING BEHAVIORAL RELATIONSHIPS

Name: Yes

⇒ ControlFlow from Are you sure? to Change Status

1.2.55 Consult Location

Activity in package 'Use Case View'

Consult Location
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Consult Location to Select Location

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Consult Location

1.2.56 Create Game

Activity in package 'Use Case View'

Create Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Create Game to Synchronization

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Logged In to Create Game

1.2.57 Delete Game

Activity in package 'Use Case View'

Delete Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Delete Game to Are you sure?

INCOMING BEHAVIORAL RELATIONSHIPS

↗ ControlFlow from DecisionNode to Delete Game

1.2.58 Edit Game

Activity in package 'Use Case View'

Edit Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

STRUCTURAL PART OF Edit Game

❶ ObjectNode1 : ActionPin

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Edit Game to Are you sure?

INCOMING BEHAVIORAL RELATIONSHIPS

↗ ControlFlow from DecisionNode to Edit Game

1.2.58.1 ObjectNode1

ActionPin owned by 'Edit Game', in package 'Use Case View'

1.2.59 Introduce Finish Date

Activity in package 'Use Case View'

Introduce Finish Date
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Introduce Finish Date to Synchronization

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Introduce Finish Date

1.2.60 Introduce Name

Activity in package 'Use Case View'

Introduce Name
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Introduce Name to Synchronization

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Introduce Name

1.2.61 Introduce Number Of Players

Activity in package 'Use Case View'

Introduce Number Of Players
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Introduce Number Of Players to Synchronization

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Synchronization to Introduce Number Of Players

1.2.62 Introduce Number of Open Slots

Activity in package 'Use Case View'

Introduce Number of Open Slots
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

STRUCTURAL PART OF Introduce Number of Open Slots
 ObjectNode2 : ActionPin
OUTGOING BEHAVIORAL RELATIONSHIPS
 ControlFlow from Introduce Number of Open Slots to Synchronization
INCOMING BEHAVIORAL RELATIONSHIPS
 ControlFlow from Synchronization to Introduce Number of Open Slots
1.2.62.1 ObjectNode2
ActionPin owned by 'Introduce Number of Open Slots', in package 'Use Case View'
OUTGOING BEHAVIORAL RELATIONSHIPS
 Object Flow from ObjectNode2 to Synchronization
1.2.63 Introduce Start Date
Activity in package 'Use Case View'

Introduce Start Date
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
 ControlFlow from Introduce Start Date to Synchronization
INCOMING BEHAVIORAL RELATIONSHIPS
 ControlFlow from Synchronization to Introduce Start Date
1.2.64 Join Game
Activity in package 'Use Case View'

Join Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
 Object Flow from Join Game to Game joined

INCOMING BEHAVIORAL RELATIONSHIPS

Name: No
⇒ ControlFlow from Are you the Game Owner? to Join Game

1.2.65 Select Game

Activity in package 'Use Case View'

Select Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Select Game to Are you the Game Owner?

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Logged In to Select Game

1.2.66 Select Location

Activity in package 'Use Case View'

Select Location
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

⇐ ControlFlow from Select Location to Synchronization

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Consult Location to Select Location

1.2.67 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS

Name: No
⇒ ControlFlow from Information Invalid? to FlowFinal

1.2.68 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Update Game to FlowFinal

1.2.69 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Delete Game to FlowFinal

1.2.70 FlowFinal

FlowFinal in package 'Use Case View'

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ Object Flow from Game joined to FlowFinal

1.2.71 DecisionNode

DecisionNode in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from DecisionNode to Edit Game

⇒ ControlFlow from DecisionNode to Delete Game

INCOMING BEHAVIORAL RELATIONSHIPS

Name: YES

⇒ ControlFlow from Are you the Game Owner? to DecisionNode

1.2.72 InteractionFragment

InteractionFragment in package 'Use Case View'

InteractionFragment

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.73 ActivityFinal

ActivityFinal in package 'Use Case View'

1.2.74 Are you sure?

DecisionNode in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Yes

↳ ControlFlow from Are you sure? to Update Game

Name: No

↳ ControlFlow from Are you sure? to Back To Home Page

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Edit Game to Are you sure?

1.2.75 Are you sure?

DecisionNode in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Yes

↳ ControlFlow from Are you sure? to Change Status

Name: No

↳ ControlFlow from Are you sure? to Back To Home Page

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Delete Game to Are you sure?

1.2.76 Are you the Game Owner?

DecisionNode in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: No

↳ ControlFlow from Are you the Game Owner? to Join Game

Name: YES

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Are you the Game Owner? to DecisionNode

INCOMING BEHAVIORAL RELATIONSHIPS

⇒ ControlFlow from Select Game to Are you the Game Owner?

1.2.77 Database

Sequence in package 'Use Case View'

Database
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: game created (true)

↳ Sequence from Database to Service

Name: Add Me as a Player

↳ Sequence from Database to Database

Name: Get All Games

↳ Sequence from Database to Database

↳ Sequence from Database to Service

Name: Game Slot Number Reduced

↳ Sequence from Database to Database

↳ Sequence from Database to Service

Name: Change Status

↳ Sequence from Database to Database

Name: Get All Games

↳ Sequence from Database to Database

Name: Insert into Game List

↳ Sequence from Database to Database

↳ Sequence from Database to Service

INCOMING BEHAVIORAL RELATIONSHIPS

Name: Add Me as a Player

⇒ Sequence from Database to Database

INCOMING BEHAVIORAL RELATIONSHIPS	
Name: Join Game	⇒ Sequence from Service to Database
Name: Create Game	⇒ Sequence from Service to Database
Name: Get All Games	⇒ Sequence from Database to Database
Name: Game Slot Number Reduced	⇒ Sequence from Database to Database
Name: Change Status	⇒ Sequence from Database to Database
Name: Get All Games	⇒ Sequence from Database to Database
Name: Request Game List	⇒ Sequence from Service to Database
Name: Insert into Game List	⇒ Sequence from Database to Database
Name: Delete Game	⇒ Sequence from Service to Database

1.2.78 Delete Game

AcceptEventAction in package 'Use Case View'

Delete Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS	
⇒ ControlFlow from Delete Game to FlowFinal	

INCOMING BEHAVIORAL RELATIONSHIPS	
⇒ ControlFlow from Change Status to Delete Game	

1.2.79 Information Invalid?

DecisionNode in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS
Name: No ↳ ControlFlow from Information Invalid? to FlowFinal
Name: Yes ↳ ControlFlow from Information Invalid? to Back To Create Game

INCOMING BEHAVIORAL RELATIONSHIPS
⇒ ControlFlow from Synchronization to Information Invalid?

1.2.80 InputData

InteractionFragment in package 'Use Case View'

InputData
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.81 Interface

Sequence «boundary» in package 'Use Case View'

Interface
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
Name: Request Game List ↳ Sequence from «boundary» Interface to Service
Name: Display Result ↳ Sequence from «boundary» Interface to «boundary» Interface
Name: Delete Game ↳ Sequence from «boundary» Interface to Service
Name: Create Game ↳ Sequence from «boundary» Interface to Service
Name: Validate Game Information ↳ Sequence from «boundary» Interface to «boundary» Interface
Name: Join Game ↳ Sequence from «boundary» Interface to Service
↳ Sequence from «boundary» Interface to User

OUTGOING BEHAVIORAL RELATIONSHIPS
↳ Sequence from «boundary» Interface to User
↳ Sequence from «boundary» Interface to User
Name: Success Message display ↳ Sequence from «boundary» Interface to User
Name: Display Result ↳ Sequence from «boundary» Interface to «boundary» Interface

INCOMING BEHAVIORAL RELATIONSHIPS
⇒ Sequence from Service to «boundary» Interface
Name: Display Result ⇒ Sequence from «boundary» Interface to «boundary» Interface
Name: Request Game List() ⇒ Sequence from User to «boundary» Interface
⇒ Sequence from Service to «boundary» Interface
Name: Validate Game Information ⇒ Sequence from «boundary» Interface to «boundary» Interface
Name: Delete Game ⇒ Sequence from User to «boundary» Interface
⇒ Sequence from Service to «boundary» Interface
Name: Create Game ⇒ Sequence from User to «boundary» Interface
⇒ Sequence from Service to «boundary» Interface
Name: Display Result ⇒ Sequence from «boundary» Interface to «boundary» Interface
Name: Select a Game Not Game Owner ⇒ Sequence from User to «boundary» Interface

1.2.82 Logged In

ActivityInitial in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS
--

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Logged In to Create Game

1.2.83 Logged In

ActivityInitial in package 'Use Case View'

OUTGOING BEHAVIORAL RELATIONSHIPS

↳ ControlFlow from Logged In to Select Game

1.2.84 Login Interface

Sequence «boundary» in package 'Use Case View'

Login Interface
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: Submit: login()
↳ Sequence from «boundary» Login Interface to «control» Verification

INCOMING BEHAVIORAL RELATIONSHIPS

Name: VisitorNotRegister
⇒ Sequence from Visitor to «boundary» Login Interface

Name: Inform Username and Password: Login
⇒ Sequence from Visitor to «boundary» Login Interface

⇒ Sequence from «control» Verification to «boundary» Login Interface

1.2.85 Object1

Sequence in package 'Use Case View'

Object1
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.86 Register

InteractionFragment in package 'Use Case View'

Register

1.2.87 Service

Sequence in package 'Use Case View'

Service
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS
↳ Sequence from Service to «boundary» Interface
↳ Sequence from Service to «boundary» Interface
Name: Join Game ↳ Sequence from Service to Database
Name: Create Game ↳ Sequence from Service to Database
↳ Sequence from Service to «boundary» Interface
↳ Sequence from Service to «boundary» Interface
Name: Request Game List ↳ Sequence from Service to Database
Name: Delete Game ↳ Sequence from Service to Database

INCOMING BEHAVIORAL RELATIONSHIPS
Name: Request Game List ⇒ Sequence from «boundary» Interface to Service
Name: game created (true) ⇒ Sequence from Database to Service
Name: Delete Game ⇒ Sequence from «boundary» Interface to Service
Name: Create Game ⇒ Sequence from «boundary» Interface to Service
Name: Join Game ⇒ Sequence from «boundary» Interface to Service

INCOMING BEHAVIORAL RELATIONSHIPS	
⇒ Sequence from Database to Service	
⇒ Sequence from Database to Service	
⇒ Sequence from Database to Service	

1.2.88 Update Game

Action in package 'Use Case View'

Update Game
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS	
↳ ControlFlow from Update Game to FlowFinal	
INCOMING BEHAVIORAL RELATIONSHIPS	
Name: Yes	
⇒ ControlFlow from Are you sure? to Update Game	

1.2.89 User

Sequence «entity» in package 'Use Case View'

User
Version 1.0 Phase 1.0 Proposed
ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS	
↳ Sequence from «entity» User to «control» Verification	
INCOMING BEHAVIORAL RELATIONSHIPS	
Name: Login	
↳ Sequence from «entity» User to «control» Verification	

INCOMING BEHAVIORAL RELATIONSHIPS	
Name: SubmitValidation() ⇒ Sequence from «control» Verification to «entity» User	

1.2.90 Validation

InteractionFragment in package 'Use Case View'

Validation

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.91 Verification

Sequence «control» in package 'Use Case View'

Verification

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

OUTGOING BEHAVIORAL RELATIONSHIPS

Name: SubmitValidation()

↳ Sequence from «control» Verification to «entity» User

↳ Sequence from «control» Verification to «boundary» Login Interface

INCOMING BEHAVIORAL RELATIONSHIPS

Name: Submit: login()

⇒ Sequence from «boundary» Login Interface to «control» Verification

⇒ Sequence from «entity» User to «control» Verification

Name: Login

⇒ Sequence from «entity» User to «control» Verification

1.2.92 Visitor Not Register

InteractionFragment in package 'Use Case View'

Visitor Not Register

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.2.93 dd

Gate in package 'Use Case View'

dd

Version 1.0 Phase 1.0 Proposed

ahgoncalves created on 11/5/2017. Last modified 11/5/2017

1.3 Component Model

Package in package 'Model'

Component Model
Version Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.1 Component Model diagram

Component diagram in package 'Component Model'

Component Model
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

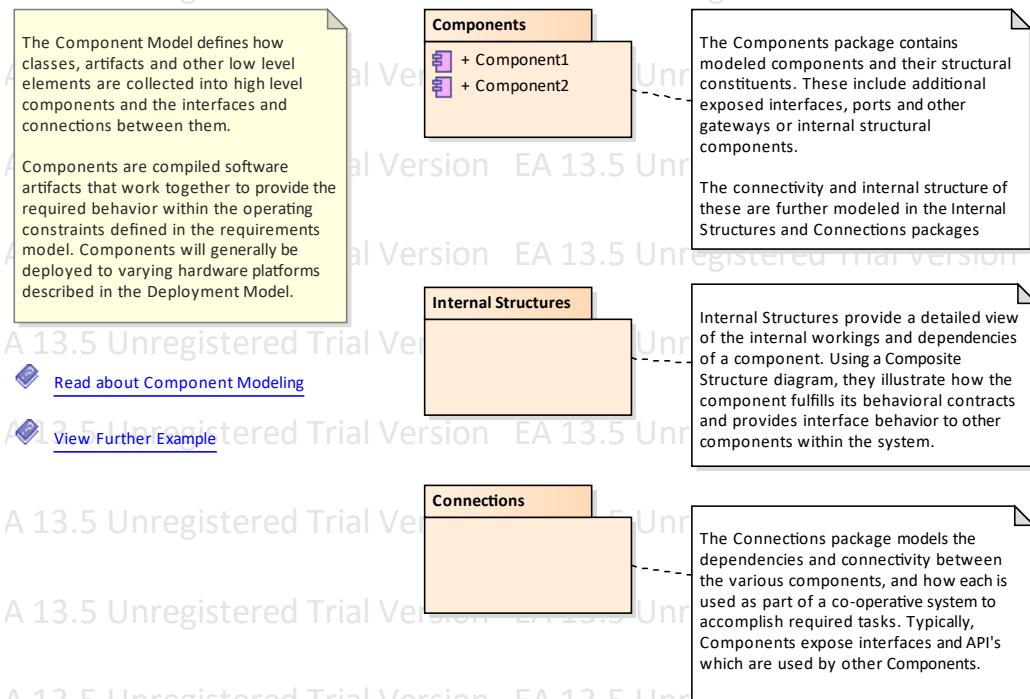


Figure 18: Component Model

1.3.2 Note

Note in package 'Component Model'

The Components package contains modeled components and their structural constituents. These include additional exposed interfaces, ports and other gateways or internal structural components.

The connectivity and internal structure of these are further modeled in the Internal Structures and Connections packages

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.3 Note

Note in package 'Component Model'

Internal Structures provide a detailed view of the internal workings and dependencies of a component. Using a Composite Structure diagram, they illustrate how the component fulfills its behavioral contracts and provides interface behavior to other components within the system.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.4 Note

Note in package 'Component Model'

The Connections package models the dependencies and connectivity between the various components, and how each is used as part of a co-operative system to accomplish required tasks. Typically, Components expose interfaces and API's which are used by other Components.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.5 Note

Note in package 'Component Model'

The Component Model defines how classes, artifacts and other low level elements are collected into high level components and the interfaces and connections between them.

Components are compiled software artifacts that work together to provide the required behavior within the operating constraints defined in the requirements model. Components will generally be deployed to varying hardware platforms described in the Deployment Model.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.6 Components

Package in package 'Component Model'

Components
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.7 Connections

Package in package 'Component Model'

Connections
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.8 Internal Structures

Package in package 'Component Model'

Internal Structures
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.9 \$help://component_model_pattern.htm

Text in package 'Component Model'

\$help://component_model_pattern.htm
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Alias Read about Component Modeling
Extends

1.3.10 \$help://componentdiagram.htm

Text in package 'Component Model'

\$help://componentdiagram.htm
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Alias View Further Example
Extends

1.3.11 Components Diagram diagram

Component diagram in package 'Component Model'

Components Diagram
Version 1.0
Me created on 11/8/2017. Last modified 11/9/2017

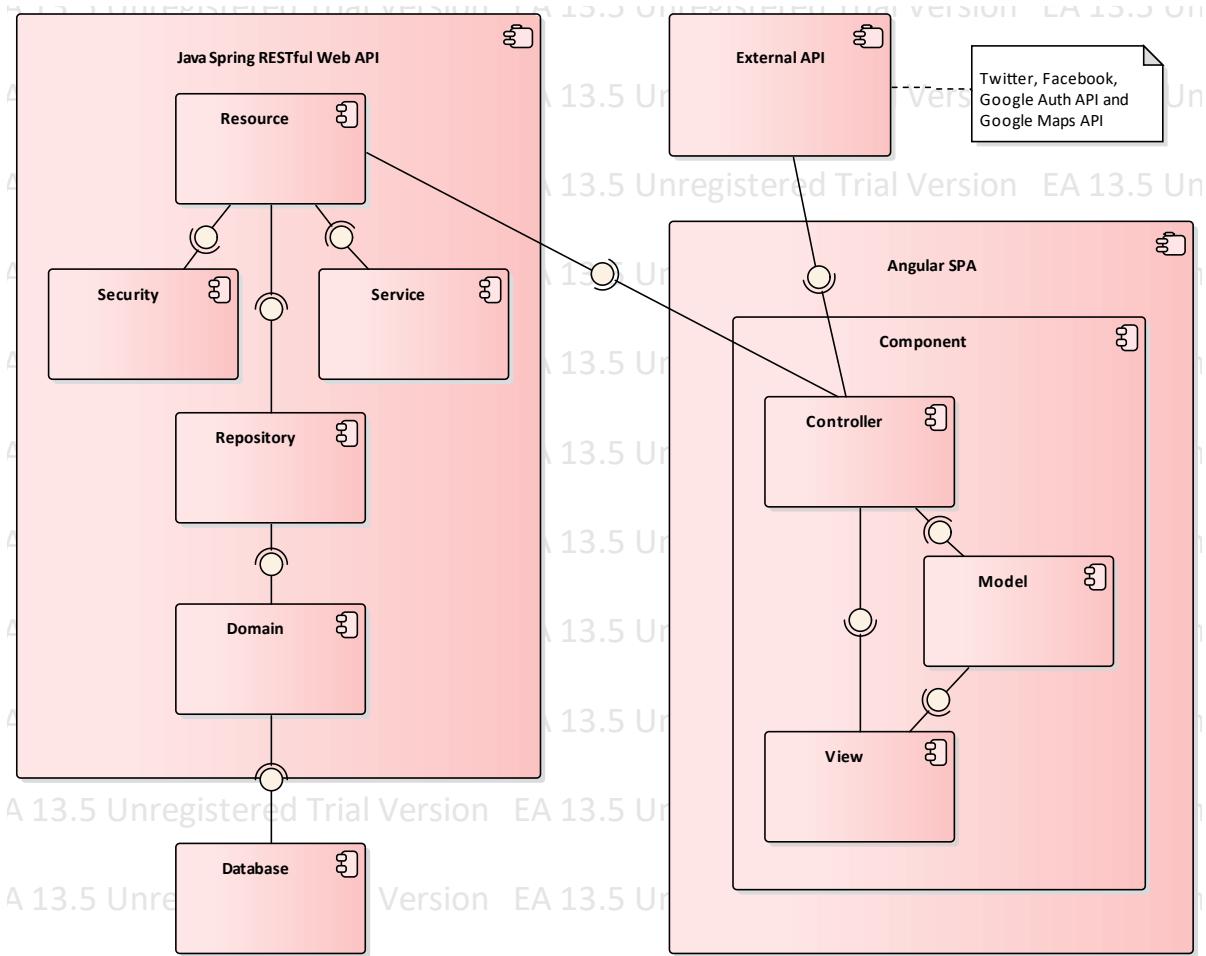


Figure 19: Components Diagram

1.3.12 Note

Note in package 'Component Model'

Twitter, Facebook, Google Auth API and Google Maps API

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.13 Angular SPA

Component in package 'Component Model'

Angular SPA
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.14 External API

Component in package 'Component Model'

External API
Version 1.0 Phase 1.0 Proposed

Me created on 11/8/2017. Last modified 11/8/2017

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Assembly from Controller to External API

[Direction is 'Source -> Destination'.]

1.3.15 Java Spring RESTful Web API*Component in package 'Component Model'*

Java Spring RESTful Web API
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/8/2017

1.3.16 Components*Package in package 'Component Model'*

Components
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/8/2017

1.3.16.1 Components diagram*Component diagram in package 'Components'*

Components
 Version 1.0
 Luís Melo created on 11/8/2017. Last modified 11/8/2017

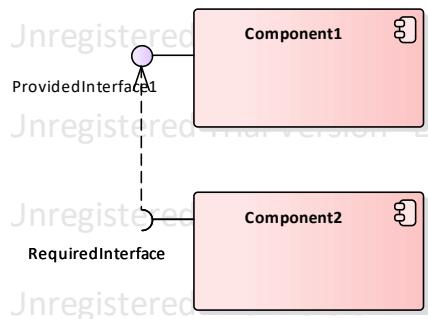


Figure 20: Components

1.3.16.2 Component1*Component in package 'Components'*

Component1
 Version 1.0 Phase 1.0 Proposed

STRUCTURAL PART OF Component1

 Part1 : Property

 ProvidedInterface1 : ProvidedInterface

1.3.16.2.1 Part1

Property owned by 'Component1', in package 'Components'

Part1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

INCOMING STRUCTURAL RELATIONSHIPS

 Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

 **Delegate** Source -> Destination

From: ProvidedInterface1 : ProvidedInterface, Public
To: Part1 : Property, Public

1.3.16.2.2 ProvidedInterface1

ProvidedInterface owned by 'Component1', in package 'Components'

ProvidedInterface1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

OUTGOING STRUCTURAL RELATIONSHIPS

 Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

 **Delegate** Source -> Destination

From: ProvidedInterface1 : ProvidedInterface, Public
To: Part1 : Property, Public

 **Dependency** Source -> Destination

From: RequiredInterface : RequiredInterface, Public
To: ProvidedInterface1 : ProvidedInterface, Public

1.3.16.3 Component2

Component in package 'Components'

Component2
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

STRUCTURAL PART OF Component2

 RequiredInterface : RequiredInterface

1.3.16.3.1 RequiredInterface

RequiredInterface owned by 'Component2', in package 'Components'

RequiredInterface
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

CONNECTORS

 **Dependency** Source -> Destination
From: RequiredInterface : RequiredInterface, Public
To: ProvidedInterface1 : ProvidedInterface, Public

1.3.17 Connections

Package in package 'Component Model'

Connections
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.17.1 Connections diagram

Component diagram in package 'Connections'

Connections
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

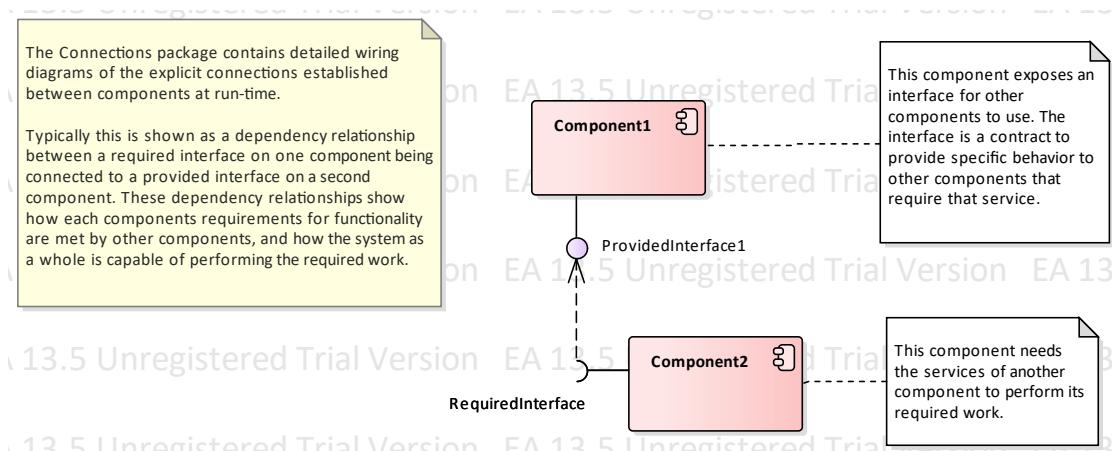


Figure 21: Connections

1.3.17.2 Component1

Component in package 'Components'

Component1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

STRUCTURAL PART OF Component1

Part1 : Property

ProvidedInterface1 : ProvidedInterface

1.3.17.2.1 Part1

Property owned by 'Component1', in package 'Components'

Part1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

INCOMING STRUCTURAL RELATIONSHIPS

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Delegate** Source -> Destination

From: ProvidedInterface1 : ProvidedInterface, Public
To: Part1 : Property, Public

1.3.17.2.2 ProvidedInterface1

ProvidedInterface owned by 'Component1', in package 'Components'

ProvidedInterface1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

OUTGOING STRUCTURAL RELATIONSHIPS

↙ Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Delegate** Source -> Destination

From: ProvidedInterface1 : ProvidedInterface, Public
To: Part1 : Property, Public

↗ **Dependency** Source -> Destination

From: RequiredInterface : RequiredInterface, Public
To: ProvidedInterface1 : ProvidedInterface, Public

1.3.17.3 Component2

Component in package 'Components'

Component2
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

STRUCTURAL PART OF Component2

✖ RequiredInterface : RequiredInterface

1.3.17.3.1 RequiredInterface

RequiredInterface owned by 'Component2', in package 'Components'

RequiredInterface

CONNECTORS

 **Dependency** Source -> Destination
 From: RequiredInterface : RequiredInterface, Public
 To: ProvidedInterface1 : ProvidedInterface, Public

1.3.17.4 ProvidedInterface1

ProvidedInterface owned by 'Component1', in package 'Components'

ProvidedInterface1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

OUTGOING STRUCTURAL RELATIONSHIPS

 Delegate from ProvidedInterface1 to Part1
 [Direction is 'Source -> Destination'.]

CONNECTORS

 **Delegate** Source -> Destination
 From: ProvidedInterface1 : ProvidedInterface, Public
 To: Part1 : Property, Public

 **Dependency** Source -> Destination
 From: RequiredInterface : RequiredInterface, Public
 To: ProvidedInterface1 : ProvidedInterface, Public

1.3.17.5 RequiredInterface

RequiredInterface owned by 'Component2', in package 'Components'

RequiredInterface
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

CONNECTORS

 **Dependency** Source -> Destination
 From: RequiredInterface : RequiredInterface, Public
 To: ProvidedInterface1 : ProvidedInterface, Public

1.3.17.6 Note

Note in package 'Connections'

This component exposes an interface for other components to use. The interface is a contract to provide specific behavior to other components that require that service.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.17.7 Note

Note in package 'Connections'

This component needs the services of another component to perform its required work.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.17.8 Note

Note in package 'Connections'

The Connections package contains detailed wiring diagrams of the explicit connections established between components at run-time.

Typically this is shown as a dependency relationship between a required interface on one component being connected to a provided interface on a second component. These dependency relationships show how each components requirements for functionality are met by other components, and how the system as a whole is capable of performing the required work.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.3.18 External API

Package in package 'Component Model'

External API
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.19 Internal Structures

Package in package 'Component Model'

Internal Structures
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.19.1 Internal Structures diagram

CompositeStructure diagram in package 'Internal Structures'

Internal Structures
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

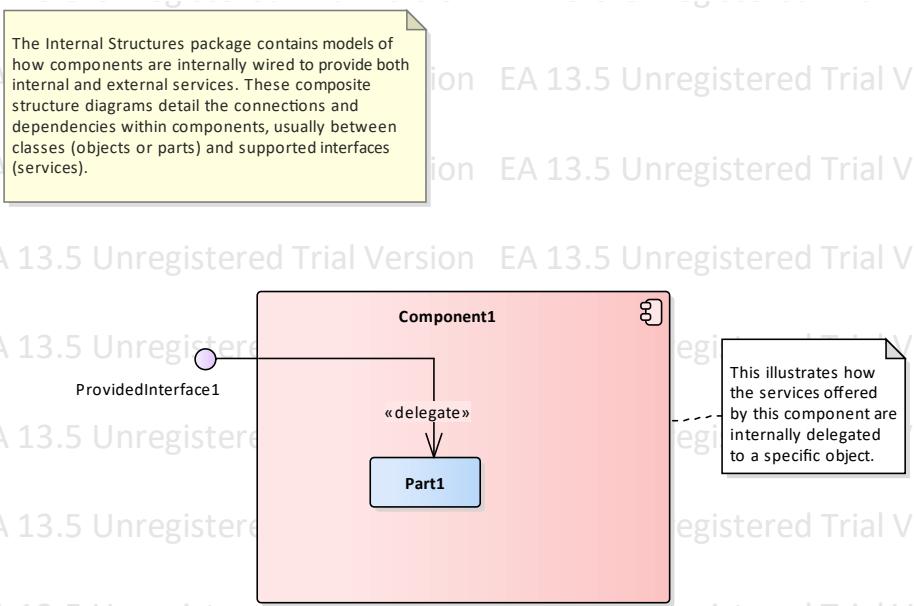


Figure 22: Internal Structures

1.3.19.2 Component1

Component in package 'Components'

Component1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

STRUCTURAL PART OF Component1	
✿	Part1 : Property
✿	ProvidedInterface1 : ProvidedInterface

1.3.19.2.1 Part1

Property owned by 'Component1', in package 'Components'

Part1

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Delegate** Source -> Destination

From: ProvidedInterface1 : ProvidedInterface, Public
To: Part1 : Property, Public

1.3.19.2.2 ProvidedInterface1

ProvidedInterface owned by 'Component1', in package 'Components'

ProvidedInterface1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

OUTGOING STRUCTURAL RELATIONSHIPS

↙ Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Delegate** Source -> Destination

From: ProvidedInterface1 : ProvidedInterface, Public
To: Part1 : Property, Public

↗ **Dependency** Source -> Destination

From: RequiredInterface : RequiredInterface, Public
To: ProvidedInterface1 : ProvidedInterface, Public

1.3.19.3 Part1

Property owned by 'Component1', in package 'Components'

Part1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Delegate from ProvidedInterface1 to Part1

[Direction is 'Source -> Destination'.]

CONNECTORS

CONNECTORS

 **Delegate** Source -> Destination
 From: ProvidedInterface1 : ProvidedInterface, Public
 To: Part1 : Property, Public

1.3.19.4 ProvidedInterface1

ProvidedInterface owned by 'Component1', in package 'Components'

ProvidedInterface1
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/8/2017

OUTGOING STRUCTURAL RELATIONSHIPS

 Delegate from ProvidedInterface1 to Part1
 [Direction is 'Source -> Destination'.]

CONNECTORS

 **Delegate** Source -> Destination
 From: ProvidedInterface1 : ProvidedInterface, Public
 To: Part1 : Property, Public

 **Dependency** Source -> Destination
 From: RequiredInterface : RequiredInterface, Public
 To: ProvidedInterface1 : ProvidedInterface, Public

1.3.19.5 Note

Note in package 'Internal Structures'

This illustrates how the services offered by this component are internally delegated to a specific object.

Note
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017
 Extends

1.3.19.6 Note

Note in package 'Internal Structures'

The Internal Structures package contains models of how components are internally wired to provide both internal and external services. These composite structure diagrams detail the connections and dependencies within components, usually between classes (objects or parts) and supported interfaces (services).

Note
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017
 Extends

1.3.20 Package1

Package in package 'Component Model'

Package1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.20.1 Package1 diagram

Component diagram in package 'Package1'

Package1
Version 1.0
Me created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 23: Package1

1.3.21 Angular SPA

Package in package 'Component Model'

Angular SPA
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.22 Component1

Package in package 'Component Model'

Component1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.23 External API

Package in package 'Component Model'

External API
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

LINKS TO OTHER PACKAGES	
 Assembly from 'Controller' Component to 'External API' Component	Direction: Source -> Destination

1.3.24 Java Spring RESTful Web API

Package in package 'Component Model'

Java Spring RESTful Web API
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.3.25 Interface1

Interface in package 'Component Model'

Interface1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.3.26 Interface2

Interface in package 'Component Model'

Interface2
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.3.27 Component

Component in package 'Component Model'

Component
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.3.28 Component

Component in package 'Component Model'

Component
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

ELEMENTS OWNED BY Component	
■	Controller : Component
■	Model : Component
■	View : Component

1.3.28.1 Controller

Component owned by 'Component', in package 'Component Model'

Controller
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Assembly from Controller to External API
[Direction is 'Source -> Destination'.]
- ↳ Assembly from Controller to Resource
[Direction is 'Source -> Destination'.]
- ↳ Assembly from Controller to Model
[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Assembly from View to Controller
[Direction is 'Source -> Destination'.]

1.3.28.2 Model

Component owned by 'Component', in package 'Component Model'

Model
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Assembly from View to Model
[Direction is 'Source -> Destination'.]
- ⇒ Assembly from Controller to Model
[Direction is 'Source -> Destination'.]

1.3.28.3 View

Component owned by 'Component', in package 'Component Model'

View
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

OUTGOING STRUCTURAL RELATIONSHIPS

- ↳ Assembly from View to Model
[Direction is 'Source -> Destination'.]

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Assembly from View to Controller

[Direction is 'Source -> Destination'.]

1.3.29 Controller

Component in package 'Component Model'

Controller
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.3.30 Database

Component in package 'Component Model'

Database
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

INCOMING STRUCTURAL RELATIONSHIPS

➡ Assembly from Domain to Database

[Direction is 'Source -> Destination'.]

1.3.31 Domain

Component in package 'Component Model'

Domain
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Assembly from Domain to Database

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

➡ Assembly from Repository to Domain

[Direction is 'Source -> Destination'.]

1.3.32 Repository

Component in package 'Component Model'

Repository
Version 1.0 Phase 1.0 Proposed

Me created on 11/8/2017. Last modified 11/10/2017

OUTGOING STRUCTURAL RELATIONSHIPS Assembly from Repository to Domain

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS Assembly from Resource to Repository

[Direction is 'Source -> Destination'.]

1.3.33 Resource

Component in package 'Component Model'

Resource

Version 1.0 Phase 1.0 Proposed

Me created on 11/8/2017. Last modified 11/10/2017

OUTGOING STRUCTURAL RELATIONSHIPS Assembly from Resource to Repository

[Direction is 'Source -> Destination'.]

 Assembly from Resource to Service

[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS Assembly from Security to Resource

[Direction is 'Source -> Destination'.]

 Assembly from Controller to Resource

[Direction is 'Source -> Destination'.]

1.3.34 Security

Component in package 'Component Model'

Security

Version 1.0 Phase 1.0 Proposed

Me created on 11/8/2017. Last modified 11/10/2017

OUTGOING STRUCTURAL RELATIONSHIPS Assembly from Security to Resource

[Direction is 'Source -> Destination'.]

1.3.35 Service

Component in package 'Component Model'

Service
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Assembly from Resource to Service

[Direction is 'Source -> Destination'.]

1.4 Deployment Model

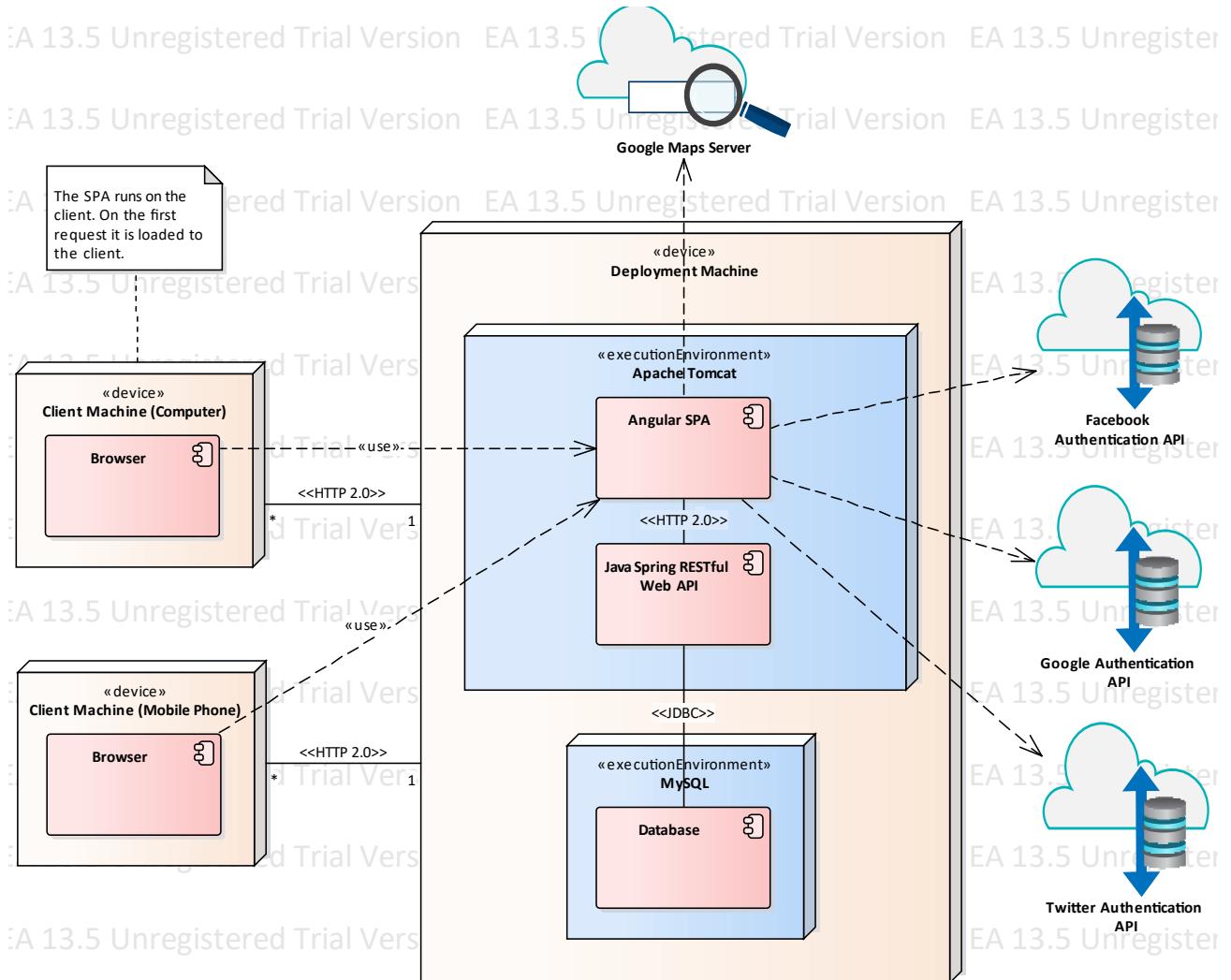
Package in package 'Model'

Deployment Model
Version Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.1 Deployment Diagram diagram

Deployment diagram in package 'Deployment Model'

Deployment Diagram
Version 1.0
Me created on 11/8/2017. Last modified 11/10/2017



Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017
 Extends

1.4.3 Deployment Model diagram

Deployment diagram in package 'Deployment Model'

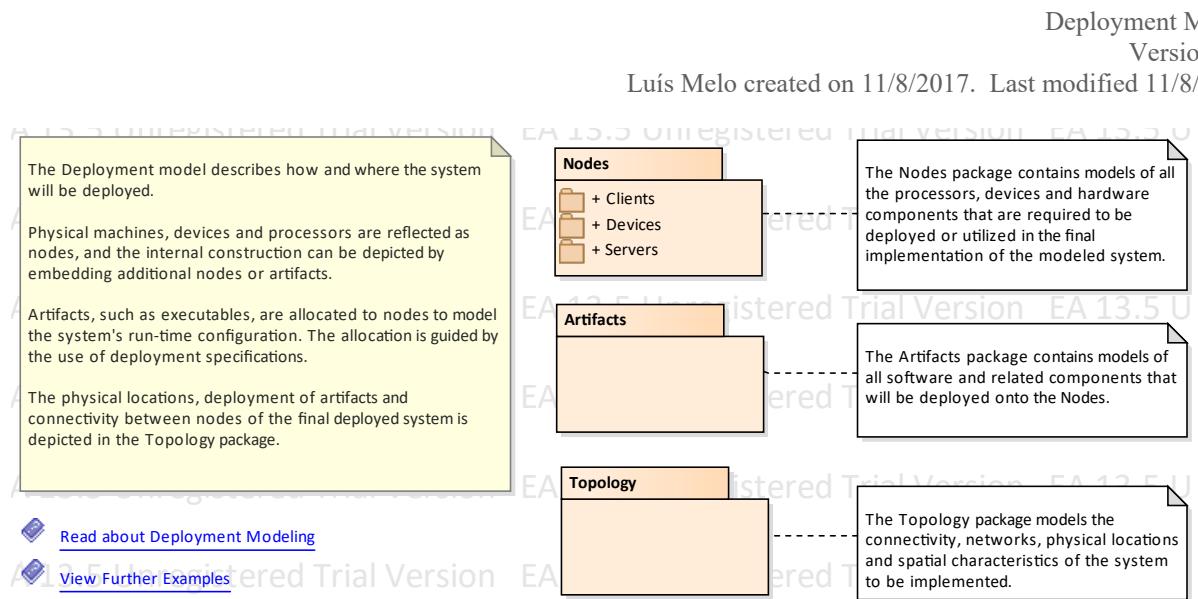


Figure 25: Deployment Model

1.4.4 Note

Note in package 'Deployment Model'

The Topology package models the connectivity, networks, physical locations and spatial characteristics of the system to be implemented.

Note
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017
 Extends

1.4.5 Note

Note in package 'Deployment Model'

The Nodes package contains models of all the processors, devices and hardware components that are required to be deployed or utilized in the final implementation of the modeled system.

Note
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017
 Extends

1.4.6 Note

Note in package 'Deployment Model'

The Artifacts package contains models of all software and related components that will be deployed onto the Nodes.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.4.7 Note

Note in package 'Deployment Model'

The Deployment model describes how and where the system will be deployed.

Physical machines, devices and processors are reflected as nodes, and the internal construction can be depicted by embedding additional nodes or artifacts.

Artifacts, such as executables, are allocated to nodes to model the system's run-time configuration. The allocation is guided by the use of deployment specifications.

The physical locations, deployment of artifacts and connectivity between nodes of the final deployed system is depicted in the Topology package.

Note
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Extends

1.4.8 \$help://deployment_model_pattern.htm

Text in package 'Deployment Model'

\$help://deployment_model_pattern.htm
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Alias Read about Deployment Modeling
Extends

1.4.9 \$help://deploymentdiagram.htm

Text in package 'Deployment Model'

\$help://deploymentdiagram.htm
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017
Alias View Further Examples
Extends

1.4.10 Nodes

Package in package 'Deployment Model'

Nodes
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.11 Artifacts

Package in package 'Deployment Model'

Artifacts
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.12 Topology

Package in package 'Deployment Model'

Topology
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.13 Nodes

Package in package 'Deployment Model'

Nodes
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.13.1 Nodes diagram

Deployment diagram in package 'Nodes'

Nodes
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 26: Nodes

1.4.13.2 Clients

Package in package 'Nodes'

Clients
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.13.2.1 Clients diagram

Deployment diagram in package 'Clients'

Clients
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 27: Clients

1.4.13.3 Devices

Package in package 'Nodes'

Devices
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.13.3.1 Devices diagram

Deployment diagram in package 'Devices'

Devices
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 28: Devices

1.4.13.4 Servers

Package in package 'Nodes'

Servers
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.13.4.1 Servers diagram

Deployment diagram in package 'Servers'

Servers
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 29: Servers

1.4.14 Artifacts

Package in package 'Deployment Model'

Artifacts
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.14.1 Artifacts diagram

Deployment diagram in package 'Artifacts'

Artifacts
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 30: Artifacts

1.4.15 Topology

Package in package 'Deployment Model'

Topology
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/8/2017

1.4.15.1 Network diagram

Deployment diagram in package 'Topology'

Network
Version 1.0
Luís Melo created on 11/8/2017. Last modified 11/8/2017

EA 13.5 Unregistered

EA 13.5 Unregistered

Figure 31: Network

1.4.16 Node1

Node in package 'Deployment Model'

Node1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.4.17 Component1

Component in package 'Deployment Model'

Component1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.4.18 Google Server

Component in package 'Deployment Model'

Google Server
Version 1.0 Phase 1.0 Proposed
Me created on 11/9/2017. Last modified 11/10/2017

1.4.19 Client Machine (Computer)

Device in package 'Deployment Model'

Client Machine (Computer)
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017

ELEMENTS OWNED BY Client Machine (Computer)

- Browser : Component

ASSOCIATIONS

- Association (direction: Unspecified) <<HTTP 2.0>>

Source: Public (Device) Client Machine (Computer)
 Cardinality: [*]

Target: Public (Device) Deployment Machine
 Cardinality: [1]

1.4.19.1 Browser

Component owned by 'Client Machine (Computer)', in package 'Deployment Model'

Browser
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017

CONNECTORS

- Usage Source -> Destination
- From: Browser : Component, Public
 To: Angular SPA : Component, Public

1.4.20 Client Machine (Mobile Phone)

Device in package 'Deployment Model'

Client Machine (Mobile Phone)
 Version 1.0 Phase 1.0 Proposed
 Me created on 11/8/2017. Last modified 11/10/2017

ELEMENTS OWNED BY Client Machine (Mobile Phone)

- Browser : Component

ASSOCIATIONS

- Association (direction: Unspecified) <<HTTP 2.0>>

Source: Public (Device) Client Machine (Mobile Phone)
 Cardinality: [*]

Target: Public (Device) Deployment Machine
 Cardinality: [1]

1.4.20.1 Browser

Component owned by 'Client Machine (Mobile Phone)', in package 'Deployment Model'

Browser
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

CONNECTORS

-  **Usage** Source -> Destination
- From: Browser : Component, Public
- To: Angular SPA : Component, Public

1.4.21 Deployment Machine

Device in package 'Deployment Model'

Deployment Machine
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

ELEMENTS OWNED BY Deployment Machine

-  Apache Tomcat : ExecutionEnvironment
-  Java : ExecutionEnvironment
-  MySQL : ExecutionEnvironment

ASSOCIATIONS

- | | |
|---|--|
|  Association (direction: Unspecified) <<HTTP 2.0>> | |
| Source: Public (Device) Client Machine (Mobile Phone) | Target: Public (Device) Deployment Machine |
| Cardinality: [*] | Cardinality: [1] |
|
 | |
|  Association (direction: Unspecified) <<HTTP 2.0>> | |
| Source: Public (Device) Client Machine (Computer) | Target: Public (Device) Deployment Machine |
| Cardinality: [*] | Cardinality: [1] |

1.4.21.1 Apache Tomcat

ExecutionEnvironment owned by 'Deployment Machine', in package 'Deployment Model'

Apache Tomcat
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

ELEMENTS OWNED BY Apache Tomcat

-  Angular SPA : Component

ELEMENTS OWNED BY Apache Tomcat

 Java Spring RESTful Web API : Component

1.4.21.1.1 Angular SPA

Component owned by 'Apache Tomcat', in package 'Deployment Model'

Angular SPA

Version 1.0 Phase 1.0 Proposed

Me created on 11/8/2017. Last modified 11/10/2017

CONNECTORS

 **Dependency** Source -> Destination

From: Angular SPA : Component, Public
To: Facebook Authentication API : Device, Public

 **Dependency** Source -> Destination

From: Angular SPA : Component, Public
To: Google Authentication API : Device, Public

 **Dependency** Source -> Destination

From: Angular SPA : Component, Public
To: Twitter Authentication API : Device, Public

 **Dependency** Source -> Destination

From: Angular SPA : Component, Public
To: Google Maps Server : Device, Public

 **Usage** Source -> Destination

From: Browser : Component, Public
To: Angular SPA : Component, Public

 **Usage** Source -> Destination

From: Browser : Component, Public
To: Angular SPA : Component, Public

ASSOCIATIONS

 Association (direction: Unspecified) <<HTTP 2.0>>

Source: Public (Component) Angular SPA
Cardinality: [1]

Target: Public (Component) Java Spring RESTful Web API

1.4.21.1.2 Java Spring RESTful Web API

Component owned by 'Apache Tomcat', in package 'Deployment Model'

Java Spring RESTful Web API

Version 1.0 Phase 1.0 Proposed

Me created on 11/8/2017. Last modified 11/10/2017

ASSOCIATIONS	
 Association (direction: Unspecified) <<JDBC>>	Source: Public (Component) Java Spring RESTful Web API Target: Public (Component) Database
 Association (direction: Unspecified) <<HTTP 2.0>>	Source: Public (Component) Angular SPA Cardinality: [1] Target: Public (Component) Java Spring RESTful Web API

1.4.21.2 Java

ExecutionEnvironment owned by 'Deployment Machine', in package 'Deployment Model'

Java
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.4.21.3 MySQL

ExecutionEnvironment owned by 'Deployment Machine', in package 'Deployment Model'

MySQL
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

ELEMENTS OWNED BY MySQL	
 Database : Component	

1.4.21.3.1 Database

Component owned by 'MySQL', in package 'Deployment Model'

Database
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

ASSOCIATIONS	
 Association (direction: Unspecified) <<JDBC>>	Source: Public (Component) Java Spring RESTful Web API Target: Public (Component) Database

1.4.22 Device1

Device in package 'Deployment Model'

Device1
Version 1.0 Phase 1.0 Proposed
Me created on 11/8/2017. Last modified 11/10/2017

1.4.23 Device2

Device in package 'Deployment Model'

Device2
Version 1.0 Phase 1.0 Proposed
Me created on 11/9/2017. Last modified 11/10/2017

1.4.24 Facebook Authentication API

Device in package 'Deployment Model'

Facebook Authentication API
Version 1.0 Phase 1.0 Proposed
Me created on 11/9/2017. Last modified 11/10/2017

CONNECTORS

 **Dependency** Source -> Destination
From: Angular SPA : Component, Public
To: Facebook Authentication API : Device, Public

1.4.25 Google Authentication API

Device in package 'Deployment Model'

Google Authentication API
Version 1.0 Phase 1.0 Proposed
Me created on 11/9/2017. Last modified 11/10/2017

CONNECTORS

 **Dependency** Source -> Destination
From: Angular SPA : Component, Public
To: Google Authentication API : Device, Public

1.4.26 Google Maps Server

Device in package 'Deployment Model'

Google Maps Server
Version 1.0 Phase 1.0 Proposed
Me created on 11/9/2017. Last modified 11/10/2017

CONNECTORS

 **Dependency** Source -> Destination
From: Angular SPA : Component, Public
To: Google Maps Server : Device, Public

1.4.27 Twitter Authentication API

Device in package 'Deployment Model'

Twitter Authentication API
Version 1.0 Phase 1.0 Proposed
Me created on 11/9/2017. Last modified 11/10/2017

CONNECTORS

 **Dependency** Source -> Destination
From: Angular SPA : Component, Public
To: Twitter Authentication API : Device, Public