view

gui

Extends

**Observable**

.

+ addObserver(o: Observable)
+ notifyObservers()

1

Extends

Extends

**GameControllerAbstract**
**{abstract}**

~ skipTurnButton : Button
~ deleteMoveButton : Button
~ turnLabel : Label
~ pane : AnchorPane
~ background: ImageView
~ publicObjectiveCard1 : ImageView
~ publicObjectiveCard2: ImageView
~ windowPattern1 : ImageView
~ dicePatternGridPane1 : ImageVIew
~ roundTrackGridPane : GridPane
~ roundTrack : ImageView
~ draftPool : GridPane
~ dicePatternImage00 : ImageView
~ dicePatternImage01 : ImageView
~ dicePatternImage02 : ImageView
~ dicePatternImage03 : ImageView
~ dicePatternImage04 : ImageView
~ dicePatternImage10 : ImageView
~ dicePatternImage11 : ImageView
~ dicePatternImage12 : ImageView
~ dicePatternImage13 : ImageView
~ dicePatternImage14 : ImageView
~ dicePatternImage20 : ImageView
~ dicePatternImage21 : ImageView
~ dicePatternImage22 : ImageView
~ dicePatternImage23 : ImageView
~ dicePatternImage24 : ImageView
~ dicePatternImage30 : ImageView
~ dicePatternImage31 : ImageView
~ dicePatternImage32 : ImageView
~ dicePatternImage33 : ImageView
~ dicePatternImage34 : ImageView
~ player1NameLabel : Label
~ turnProgressBar : ProgressBar
~ turnTimeline : Timeline
~ imageViewsSetup : boolean
~ diceChosenFromDraftPool : ImageView
- diceChosenFromDicePattern : ImageView
~ toolCardSelected : ImageView
~ mvEvents : Map<Integer, Runnable>
~ diceIndexMap : Map<Integer, Runnable>
~ dicePositionFromDicePatternMap<Integer, Runnable>
~ dicePatternRowPosition : int
~ dicePatternColumnPosition : int
- diceMoved : boolean
~ idToolCardSelected : int
- listDicePattern : List<ImageView>
- roundIndex : int
- step : int
- isToolCardSelected : boolean
~ toolCardImageList : List <ImageView>
- choiceTapWheel : int
- choiceGrozingPliers : int
- choiceFluxRemover : int
- diceIndexRoundTrack : int
~ diceIndexDraftPool : int
- yourTurnText : String
~ waitText : String

~ abstractInitialize()
~ updateProgressBar()

1

**NicknameChoiceController**

- confirmButton : Button
- exitButton : Button
- textFiedl : TextField
- nickAlreadyTaken1: Label
- nickAlreadyTaken2 : Label
- nickAlreadyTaken3: Label
- firstTime : boolean

- exitFromButton()
+ nickNameEntered()
- showError( errorEvent : ErrorEvent)

**ConnectionChoiceController**

- rmiButton : Button
- socketButton : Button
- ipServer : TextField
- serverNotFound1 : Label
- serverNotFound2 : Label
- ipAddressString : String

- rmiButtonAction()
- socketButtonAction()
- getIpServer()

**WaitingRoomController**

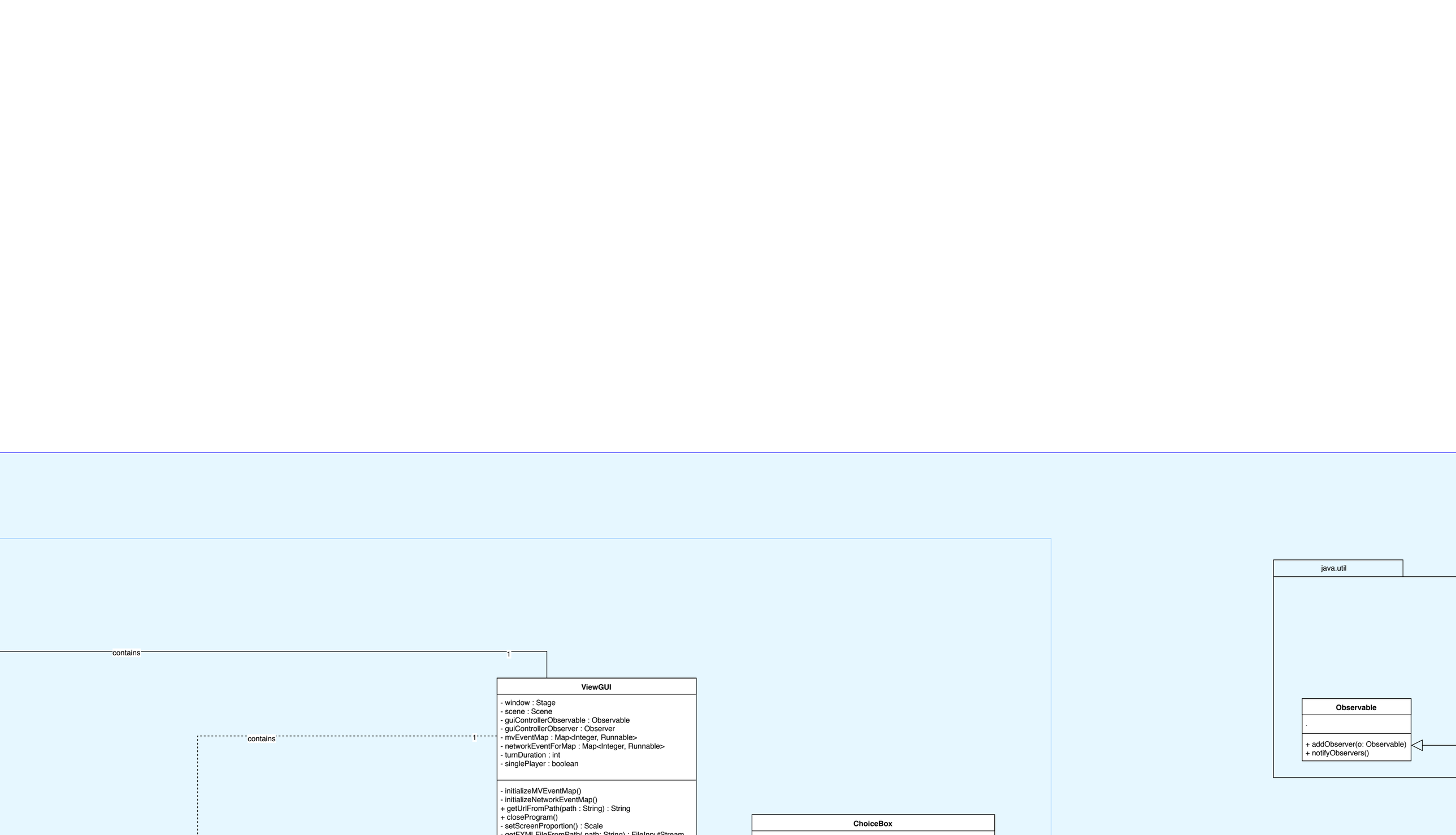
- header : ImageView
- leftColumn1 : ImageView
- leftColumn2 : ImageView
- rightColumn1 : ImageView
- rightColumn2 : ImageView
- notificationTextArea : TextArea
- label1 : Label
- label2 : Label
- label3 : Label
- difficultySinglePlayer : ChoiceBox<Integer>
- singlePlayerButton : Button

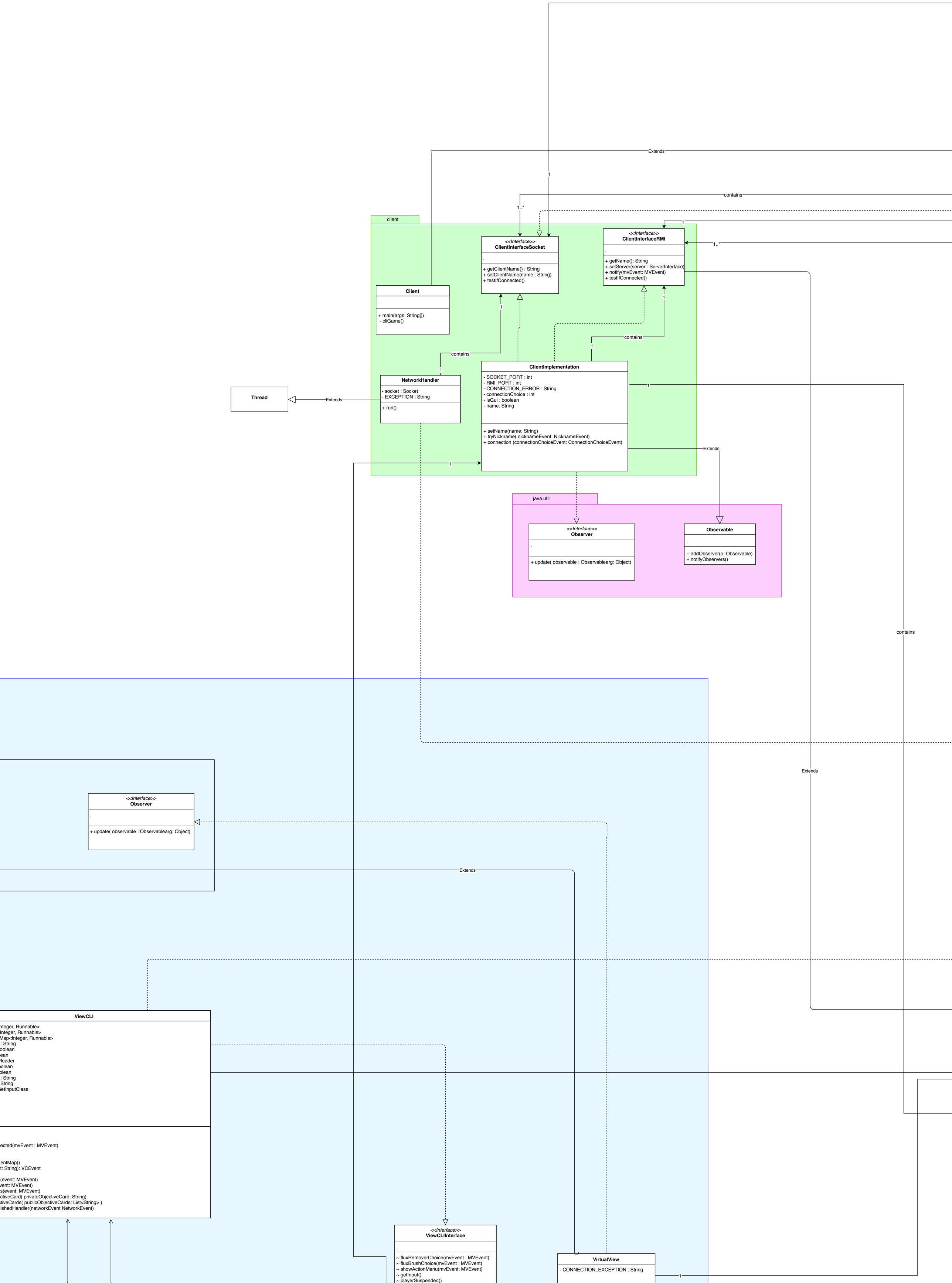
- singlePlayerButtonAction()
- setClientLabel(clientConnectedStringList : List<String>)
- deactivateSinglePlayerOption()
+ initialize()

**WindowPatternChoiceControlle**

- windowPattern1 : ImageView
- windowPattern2 : ImageView
- windowPattern3 : ImageView
- windowPattern4 : ImageView
- confirmButton : Button
- cancelButton : Button
- privateObjectiveCard1 : ImageView
- privateObjectiveCard2 : ImageView
- waitYourTurn1 : Label
- waitYourTurn2 : Label
- background : ImageView
- chosen : int

+ initialize()
- windowPatternSelected()
- setBorderGlow() : DropShadow


contains
1
contains
1
1
<<Interface>>
Observer
.
+ update( observable : Observablearg: Object)
ViewGUI
- window : Stage
- scene : Scene
- guiControllerObservable : Observable
- guiControllerObserver : Observer
- mvEventMap : Map<Integer, Runnable>
- networkEventForMap : Map<Integer, Runnable>
- turnDuration : int
- singlePlayer : boolean
- initializeMVEventMap()
- initializeNetworkEventMap()
+ getUrlFromPath(path : String) : String
+ closeProgram()
- setScreenProportion() : Scale
- getFXMLFileFromPath( path: String) : FileInputStream
+ getGuiControllerObservable() : Observable
+ startGUI(primaryStage : Stage)
- setNicknameChoiceScene()
- setConnectionChoiceScene()
- setLobbyScene()
- setWindowPatternChoiceScene()
- setGameScene()
- setEndScreen()
- connectionEstablishedHandler ( networkEvent : NetworkE
- newObserverEventHandler(networkEvent : NetworkEven
ChoiceBox
- choice : boolean
+ confirmChoice( title : String, message : String) : boolean
java.util
Observable
.
+ addObserver(o: Observable)
+ notifyObservers()
Implements
Implements
GameController
- toolCard1 : ImageView
- toolCard2 : ImageView
- toolCard3 : ImageView
- privateObjectiveCard :ImageView
- publicObjectiveCard3 : ImageView
- windowPattern4 : ImageView
- windowPattern3 : ImageView
- windowPattern2 : ImageView
- dicePatternGridPane4 : GridPane
- dicePatternGridPane3 : GridPane
- dicePatternGridPane2 : GridPane
- toolCard1FavorTokensLabel : Label
- toolCard2FavorTokensLabel : Label
- toolCard3FavorTokensLabel : Label
- player2Name : Label
- player3Name : Label
- player4Name : Label
- favorTokensToolCard1ImageView : ImageView
1
SinglePlayerGameController
- privateObjectiveCard1 : ImageView
- privateObjectiveCard1 : ImageView
- toolCard1 : ImageView
- toolCard2 : ImageView
- toolCard3 : ImageView
- toolCard4 : ImageView
- toolCard5 : ImageView
- idToolCard1 : int
- idToolCard2 : int
- idToolCard3 : int
- idToolCard4 : int
- idToolCard5 : int
- dicePaid : boolean
- toolCardsUsed : List <ImageView>
- diceChosenToPay : ImageView
+ initialize()
- setToolCardEventHandler( size : int)
EndScreenController
- exitButton : Button
- playAgainButton : Button
- winnerLabel : Label
- firstPlaceLabel : Label
- secondPlaceLabel : Label
- thirdPlaceLabel : Label
- forthPlaceLabel : Label
- firstPlacePoints : Label
- secondPlacePoints : Label
- thirdPlacePoints : Label
- forthPlacePoints : Label
- scene : AnchorPane
- scoreTrack : ImageView
- yellowScoreMarker : ImageView
- yellowScoreMarker2 : ImageView
- redScoreMarker : ImageView
- redScoreMarker2 : ImageView
- blueScoreMarker . ImageView
1
1
1
1
+ askConnection()
- askName()
- showClientConn
- createMVMap()
- createVCMap()
- createNetworkEv
- createEvent(inpu
+ getInputStarter()
- showDicePattern
- showDraftPool(e
- showFavorToken
- showPrivateObje
- showPublicObjec
- connectionEstabl
- vcEvents: Map<I
- mvEvents: Map<
- networkEvents :
- eventParameters
- toolCardUsed : b
- diceMoved : bool
- reader: Buffered
- firstTimeNick : bo
- singlePlayer : boo
- INVALID_MOVE
- IOE_THROWN :
- getInputClass : G


Extends
1
contains
1..*
client
1
<<Interface>>
ClientInterfaceSocket
.
+ getClientName() : String
+ setClientName(name : String)
+ testIfConnected()
<<Interface>>
ClientInterfaceRMI
.
+ getName(): String
+ setServer(server : ServerInterface)
+ notify(mvEvent: MVEvent)
+ testIfConnected()
1..*
Client
.
+ main(args: String[])
- cliGame()
1
1
contains
contains
1
1
NetworkHandler
- socket : Socket
- EXCEPTION : String
+ run()
ClientImplementation
- SOCKET_PORT : int
- RMI_PORT : int
- CONNECTION_ERROR : String
- connectionChoice : int
- isGui : boolean
- name: String
+ setName(name: String)
+ tryNickname( nicknameEvent: NicknameEvent)
+ connection (connectionChoiceEvent: ConnectionChoiceEvent)
1
Thread
Extends
1
Extends
java.util
<<Interface>>
Observer
.
+ update( observable : Observablearg: Object)
Observable
.
+ addObserver(o: Observable)
+ notifyObservers()
contains
Extends
<<Interface>>
Observer
.
+ update( observable : Observablearg: Object)
Extends
ViewCLI
nteger, Runnable>
Integer, Runnable>
Map<Integer, Runnable>
: String
oolean
ean
Reader
olean
olean
: String
String
GetInputClass
ected(mvEvent : MVEvent)
entMap()
t: String): VCEvent
(event: MVEvent)
vent: MVEvent)
s(event: MVEvent)
ctiveCard( privateObjectiveCard: String)
ctiveCards( publicObjectiveCards: List<String> )
lishedHandler(networkEvent NetworkEvent)
<<Interface>>
ViewCLIInterface
.
~ fluxRemoverChoice(mvEvent : MVEvent)
~ fluxBrushChoice(mvEvent : MVEvent)
~ showActionMenu(mvEvent: MVEvent)
~ getInput()
~ playerSuspended()
VirtualView
- CONNECTION_EXCEPTION : String
1

**Application**

+ field: Type

+ start ( primaryStage : Stage )

contains

«interface»
**java.rmi.Remote**

Extends

java.rmi.RemoteException

throws

**Thread**

Extends

**Server**

*<<Interface>>*
**ServerInterface**

+ addClient(client: ClientInterfaceRMI)
+ send(event: MVEvent)
+ sendTo(event: MVEvent, clientInterface ClientInterface)
+ notify(VCEvent: vcEvent)

contains

1

1

**ClientGatherer**

- serverSocket : ServerSOcket

+ run()

**VirtualClient**

- clientSocket : Socket
- clientName : String
- stream : ObjectOutputStream

+ run()

1

contains

1

java.rmi.Server.UnicastRemoteObject

Extends

**ServerImplementation**

- CONNECTION_LOST : String
- ABANDONED_GAME : String
- lock : Object
- timer : Timer
- gameStarted : boolean
- choosingWindowPatterns : boolean
- setupDuration : int
- turnDuration : int

+ getClients(): List<ClientInterface>
+ getOldClients(): List<ClientInterface>
+ addSocketClient(client : ClientInterfaceSocket, name : String)
- createGame()
- createPlayerList(clients: List<ClientInterface>): List<Player>
+ sendTo(mvEvent: MVEvent, clientInterface: ClientInteface)
+ send(mvEvent: MVEvent)
+ notify(vcEvent: VCEvent)
- waitingRoomRefresh ()

1

contains

1

**Server**

.

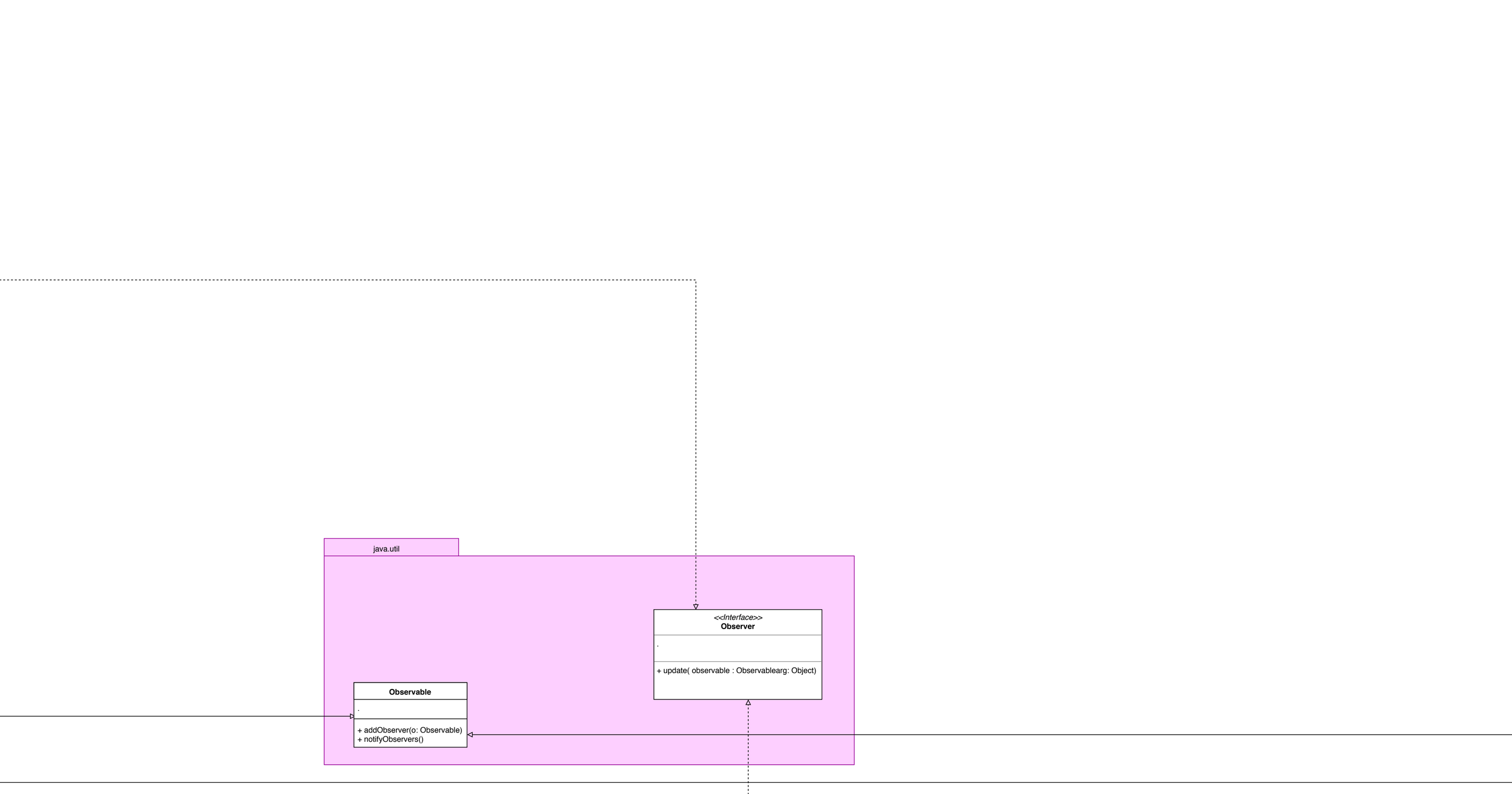
+ main(args: String[])
+ addClient(newClient : Socket)
~ getServerImplementation : ServerImplementation

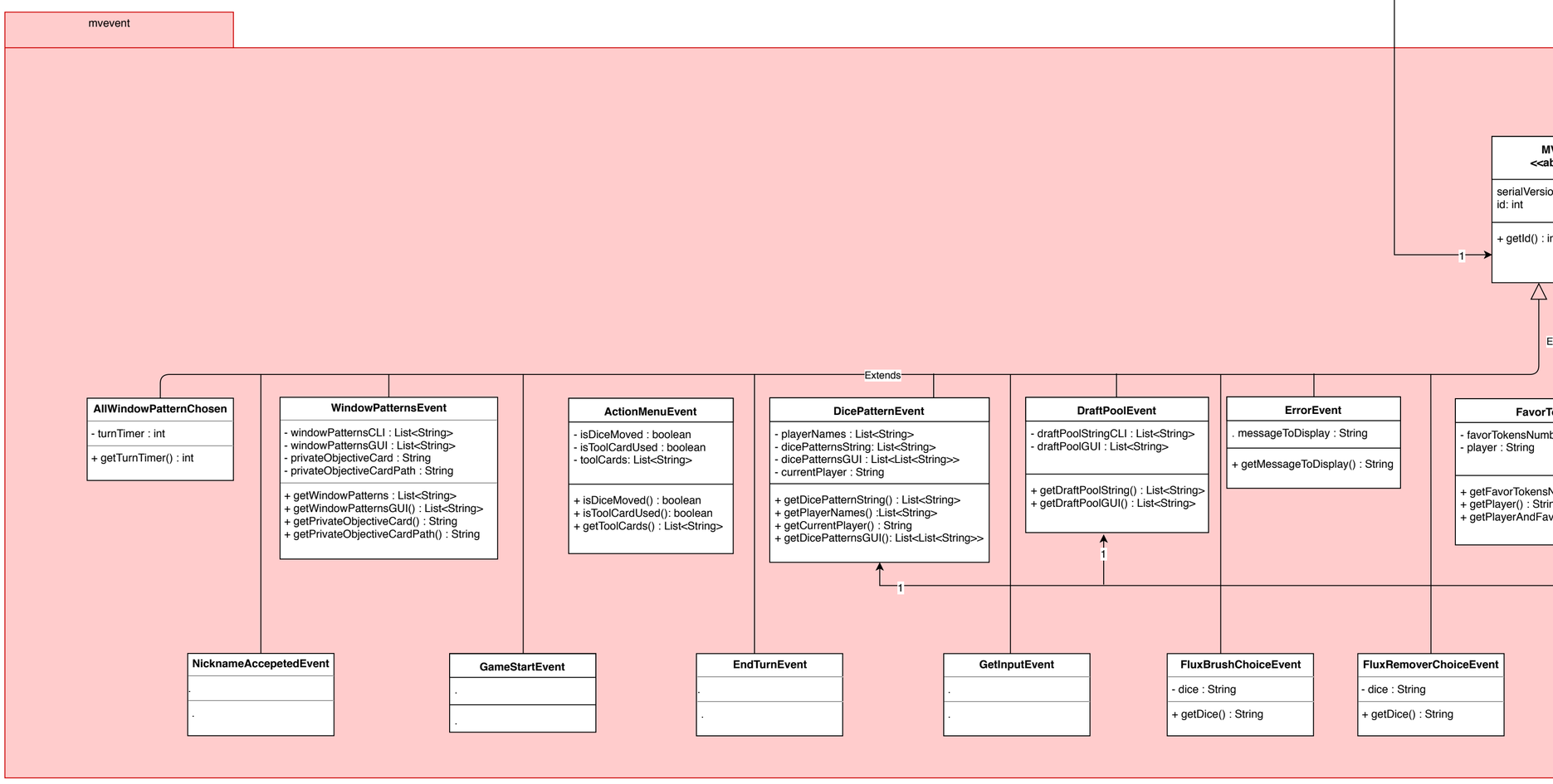
Extends

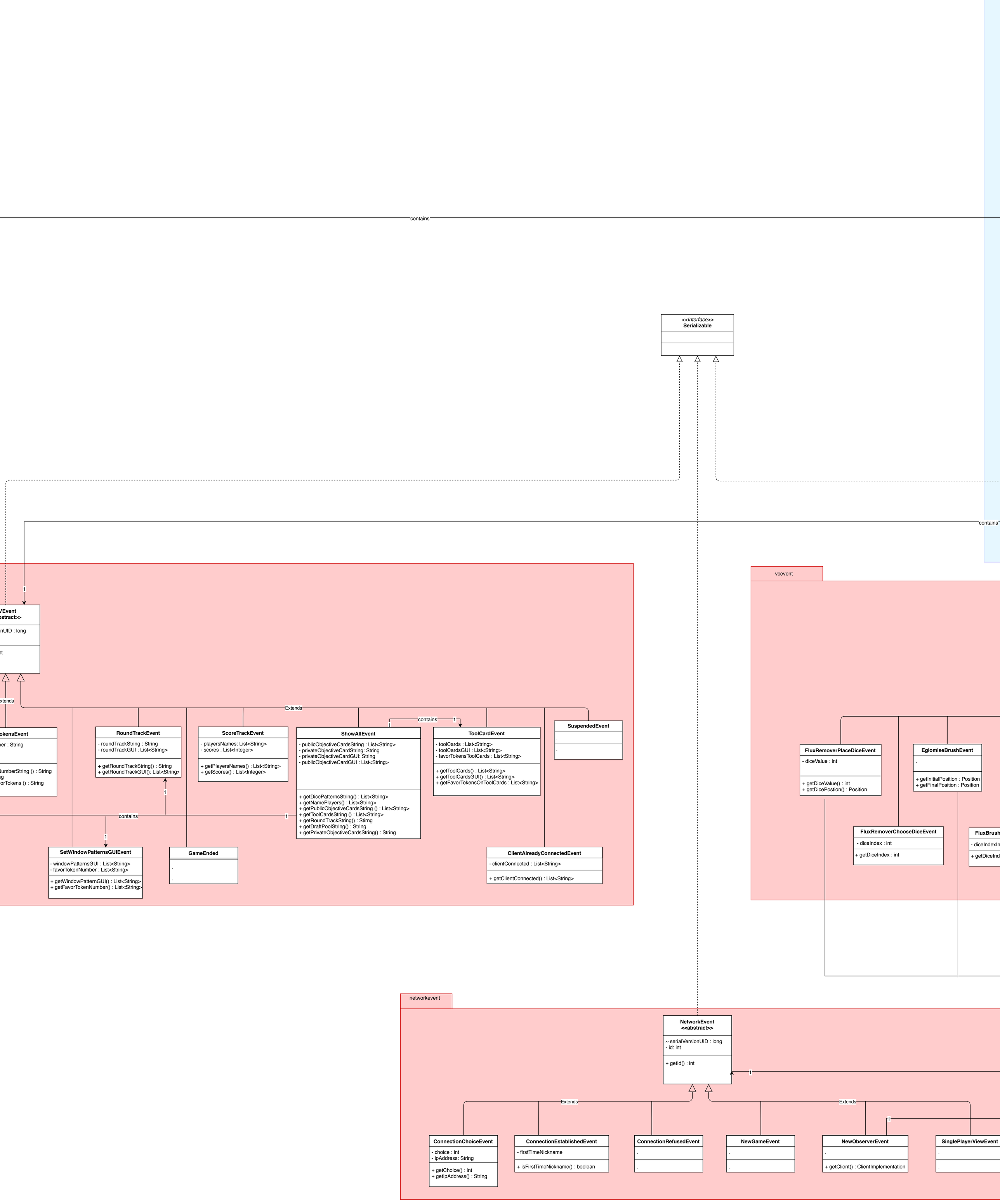
1

1

contains


java.util
<<Interface>>
Observer
.
+ update( observable : Observablearg: Object)
Observable
.
+ addObserver(o: Observable)
+ notifyObservers()


mvevent
Extends
AllWindowPatternChosen
- turnTimer : int
+ getTurnTimer() : int
WindowPatternsEvent
- windowPatternsCLI : List<String>
- windowPatternsGUI : List<String>
- privateObjectiveCard : String
- privateObjectiveCardPath : String
+ getWindowPatterns : List<String>
+ getWindowPatternsGUI() : List<String>
+ getPrivateObjectiveCard() : String
+ getPrivateObjectiveCardPath() : String
ActionMenuEvent
- isDiceMoved : boolean
- isToolCardUsed : boolean
- toolCards: List<String>
+ isDiceMoved() : boolean
+ isToolCardUsed(): boolean
+ getToolCards() : List<String>
DicePatternEvent
- playerNames : List<String>
- dicePatternsString: List<String>
- dicePatternsGUI : List<List<String>>
- currentPlayer : String
+ getDicePatternString() : List<String>
+ getPlayerNames() :List<String>
+ getCurrentPlayer() : String
+ getDicePatternsGUI(): List<List<String>>
DraftPoolEvent
- draftPoolStringCLI : List<String>
- draftPoolGUI : List<String>
+ getDraftPoolString() : List<String>
+ getDraftPoolGUI() : List<String>
ErrorEvent
. messageToDisplay : String
+ getMessageToDisplay() : String
FavorT
- favorTokensNumb
- player : String
+ getFavorTokensN
+ getPlayer() : Strin
+ getPlayerAndFav
1
1
1
NicknameAccepetedEvent
.
.
GameStartEvent
.
.
EndTurnEvent
.
.
GetInputEvent
.
.
FluxBrushChoiceEvent
- dice : String
+ getDice() : String
FluxRemoverChoiceEvent
- dice : String
+ getDice() : String


<<Interface>>
Serializable
contains
contains
1
vcevent
1
xtends
Extends
contains
1
1
1
contains
1
okensEvent
er : String
NumberString () : String
g
orTokens () : String
RoundTrackEvent
- roundTrackString : String
- roundTrackGUI : List<String>
+ getRoundTrackString() : String
+ getRoundTrackGUI(): List<String>
ScoreTrackEvent
- playersNames: List<String>
- scores : List<Integer>
+ getPlayersNames() : List<String>
+ getScores() : List<Integer>
ShowAllEvent
- publicObjectiveCardsString : List<String>
- privateObjectiveCardString: String
- privateObjectiveCardGUI: String
- publicObjectiveCardGUI : List<String>
+ getDicePatternsString() : List<String>
+ getNamePlayers() : List<String>
+ getPublicObjectiveCardsString () : List<String>
+ getToolCardsString () : List<String>
+ getRoundTrackString() : Stirng
+ getDraftPoolString() : String
+ getPrivateObjectiveCardsString() : String
ToolCardEvent
- toolCards : List<String>
- toolCardsGUI : List<String>
- favorTokensToolCards : List<String>
+ getToolCards() : List<String>
+ getToolCardsGUI() : List<String>
+ getFavorTokensOnToolCards : List<String>
SuspendedEvent
.
.
SetWindowPatternsGUIEvent
- windowPatternsGUI : List<String>
- favorTokenNumber : List<String>
+ getWindowPatternGUI() : List<String>
+ getFavorTokenNumber() : List<String>
GameEnded
.
.
ClientAlreadyConnectedEvent
- clientConnected : List<String>
+ getClientConnected() : List<String>
FluxRemoverPlaceDiceEvent
- diceValue : int
+ getDiceValue() : int
+ getDicePostion() : Position
EglomiseBrushEvent
.
+ getInitialPosition : Position
+ getFinalPosition : Position
FluxRemoverChooseDiceEvent
- diceIndex : int
+ getDiceIndex : int
FluxBrush
- diceIndexIn
+ getDiceInd
networkevent
NetworkEvent
<<abstract>>
~ serialVersionUID : long
- id: int
+ getId() : int
1
Extends
Extends
1
ConnectionChoiceEvent
- choice : int
- ipAddress: String
+ getChoice() : int
+ getIpAddress() : String
ConnectionEstablishedEvent
- firstTimeNickname
+ isFirstTimeNickname() : boolean
ConnectionRefusedEvent
.
.
NewGameEvent
.
.
NewObserverEvent
.
+ getClient() : ClientImplementation
SinglePlayerViewEvent
.
.

- updateProgressBar()
- initializeDicePatternEventHandler(gridPane : GridPane)
- handleDraftPool()
~ disableDraftPool()
~ enableDraftPool()
- setDraftPoolOpacity(opacity : Double)
~ handleDraftPoolAndToolCards()
~ skipTurn()
~ addImageToImageView(path : String, imageView : ImageView, height : int, width : int)
~ zoomIn(imageView : ImageView)
~ zoomOut(imageView : ImageView)
~ zoomInPOC(imageView : ImageView)
~ zoomOutPOC(imageView : ImageView)
- getNodeByRowColumnIndex(row : int, column : int, gridPane : GridPane, dimension : int) : Object
~ initializeGridPaneImagesView(gridPane : GridPane, row:int, column: int , height :int, width :int )
~ setBorderGlow():DropShadow
- addImagesViewToGridPane(gridPane,:GridPane, imagesView : List<ImageView> )
~ disableToolCards()
~ enableToolCards()
- handleToolCardsEffect()
~ disableGridPane(gridPane : GridPane , imageView: ImageView )
- setDisableGridPane(gridPane : GridPane)
~ enableGridPane(gridPane : GridPane , imageView : ImageView )
- setEnableGridPane(gridPane : GridPane )
- grozingPliersWindow()
- fluxRemoverWindow()
- tapWheelWindow()
- createMVMapAbstract()
- showAll(event : MVEvent )
- handleEndTurnEvent()
- showEndTurn()
~ playerSuspended()
~ createAssociationWithOurGridPane(mvEvent1 : MVEvent , mvEvent2: MVEvent)
~ updatePrivateObjectiveCards(paths: List<String>)
~ showErrorAbstract(errorEvent : ErrorEvent)
~ showError(mvEvent : MVEvent)
~ updateDicePatterns(mvEvent : MVEvent )
~ updateFavorTokens(mvEvent : MVEvent )
~ updateToolCards(mvEvent : MVEvent)
- handleActionMenu(event : MVEvent)
~ updatePublicObjectiveCards(publicObjectiveCards : List<String>)
- updateRoundTrack(event : MVEvent)
~ handleToolCards(idToolCard : int)
~ updateDicePattern ( list : List<String>, gridPane : GridPane, height : int, width : int)
~  updateDraftPool(event :MVEvent)
- cleanDraftPool()
~ getDicePositionFromRoundTrack(imageView : ImageView )
~  createDiceIndexMap()
~ putDiceOnDicePattern()
~ fluxRemoverChooseDice()
~ grozinPliers()
~ fluxBrushChooseDice()
~ grindingStone()
~ getDiceIndexFromDraftPool(imageView : ImageView )
- initializeRoundTrackEventHandler(gridPane:GridPane )
-initializeDraftPoolEventHandler(gridPane:GridPane )
- getDicePositionFromDicePattern(imageView: ImageView )
~ placeDiceMove(row : int , column : int)
~ eglomiseBrush(row :int , column : int )
- eglomiseBrushSecondStep(row :int, column:int)
~ copperFoilBurnisher(row :int , column :int)
- copperFoilBurnisherSecondStep(row : int , column : int)
~ lathekin(row : int, column : int)
- lathekinSecondStep(row : int , column: int )
- lathekinForthStep(row : int , column: int )
~ fluxBrushPlaceDice(row : int , column : int)
~ runnerPliers(row :int , column : int )
~ corkBackedStraighedge(row :int , column : int )
~ fluxRemoverPlaceDice(row :int , column: int )
~ tapWheel(row:int , column:int )
- tapWheelSecondStep(row :int , column:int )
- tapWheelForthStep(row:int , column:int )
- firstStep(row :int , column:int )
~ createDicePositionFromDicePatternMap()

- deleteBorderGlow()
- windowPattern1Chosen ()
- windowPattern2Chosen()
- windowPattern3Chosen()
- windowPattern4Chosen()
- confirmButtonPressed()
- cancelButtonPressed()
+ setWindowPatterns(windowPatternsEvent : WindowPatte

Extends

Extends

1

1

contains

1..*

**VCEvent**
**<<abstract>>**

serialVersionUID: long
- id : int

+ getId() : int

1

1

Extends

Extends

Extends

**FluxBrushPlaceDiceEvent**

.

+ getFinalPostion() : Postion

**CopperFoilBurnisherEvent**

.

+ getInitialPosition() : Position
+ getFinalPosition() : Position

**CorkBakedStraightEdgeEvent**

- indexDraftPool : int

+ getIndexDraftPool() : int
+ getFinalPosition() : Position

**LensCutterEvent**

- roundIndex : int
- diceIndexRoundTrack : int
- diceIndexDraftPool : int

+ getRoundIndex() : int
+ getDiceIndexRoundTrack() : int
+ getDiceIndexDraftPool() : int

**PlaceDiceEvent**

- diceIndexDraftPool : int

+ getDiceIndexDraftPool() : int
+ getPositionToPlace() : Position

**RunnerPliersEvent**

- diceIndex : int

+ getDiceIndex(): int
+ getPosition() : Position

**SkipTurnEvent**

.

.

**TapWheelEvent**

- roundIndex : int
- diceIndex : int

+ getRoundIndex() : int
+ getDiceIndex() : int
+ getFirstDicePosition() : Position
+ getSecondDicePosition() : Position
+ getNewFirstDicePosition () : Position
+ getNewSecondDicePosition () : Position

**GrindingStoneEvent**

- dicePosition : int

+ getDicePostion() : int

**ChooseDiceEvent**

DraftPool : int

exInDraftPool() : int

**NicknameEvent**

- nickname : String
- firstTime : boolean

+ getNickname() : String
+ isFirstTime() : boolean

**DiceChosenSinglePlayer**

- dicePosition : int

+ getDicePosition() : int

**TestEvent**

.

.

**SinglePlayerEvent**

- difficulty : int

+ getDifficulty(): int

**Uns**

- name

+ getNa

contains

ernsEvent)

- favorTokensToolCard2ImageView : ImageView
- favorTokensToolCard3ImageView : ImageView
- favorTokensPlayer1ImageView : ImageView
- favorTokensPlayer2ImageView : ImageView
- favorTokensPlayer3ImageView : ImageView
- favorTokensPlayer4ImageView : ImageView
- favorTokensPlayer1Label : Label
- favorTokensPlayer2Label : Label
- favorTokensPlayer3Label : Label
- favorTokensPlayer4Label : Label
- idToolCard1 : int
- idToolCard2 : int
- idToolCard3 : int
- favorTokensPath : String

---

+ initalize()
- useToolCard1()
- useToolCard2()
- useToolCard3()
+ update( o: Observable, event : Object)
- getCurrentPlayerIndex(players : List<String> ,  currentPlayer : String ) : int
- getRemainingPositions(players : List<String> , index : int ) : List<Integer>
- getCurrentPlayerOurGridPane(currentPlayer : String ) : OurGridPane
- updateLabel(label : Label, text : String)

- payDice()
~ getDiceIndexFromDraftPool(imageView: ImageView)
- useToolCard1()
- useToolCard2()
- useToolCard3()
- useToolCard4()
- useToolCard5()
- useToolCardGeneric(toolCard: ImageView, idToolCard: int)
~ enableToolCards(event : MVEvent)
~ enableDraftPool()
~ updateDraftPool(event : MVEvent)
~ createDiceIndexMap()
~ createDicePositionFromDicePatternMap()
~ getDicePositionFromRoundTrack(imageView: ImageView)

- blueScoreMarker2 : ImageView
- greenScoreMarker : ImageView
- greenScoreMarker2 : ImageView

---

- exitButtonAction()
- playAgainButtonAction()
- setImageOnScene(marker : ImageView, score : int)
- setScoreMarketUnit(marker : ImageView, score : int)
- setRankingLabels(marker : ImageView, score : int)
- setRankingLabels(scoreTrackEvent : ScoreTrackEvent)
- placeScoreMarkerOnScene(score : List<Integer>)
- scoreTrackEventHandler(mvEvent : MVEvent)
- winnerEventHandler(mvEvent : MVEvent)
+ initialize()

**GetInputClass**

.

+ run()

**SuspendedPlayer**

.

+ run()

**SelectWind**

.

+ run()

Extends

contains

3

**OurGridPane**

- gridPane : GridPane
- playerName : String
- windowPatternPath : String
- playerNameLabel: Label
- favorTokens : int
- favorTokensLabel : Label

---

+ OurGridPane( gridPane : GridPane, playerName : String, windowPatternPath: String,  playerNameLabel: Label, favorTokens : int, favorTokensLabel: Label) : OurGridPane
+ OurGridPane( gridPane: GridPane, playerName : String, windowPatternPath : String, playerNameLabel : Label) : OurGridPane
+ getGridPane() : GridPane
+ getPlayerName() : String
- setPlayerNameLabel( name : String)
+ setFavorTokensLabel( favorTokens : int)

contains

**GrozingPliersEvent**

- diceIndexDraftPool : int
- choice : int

---

+ getDiceIndexDraftPool(): int
+ getChoice() : int

**LathekinEvent**

.

---

+ getInitialPosition1() : Position
+ getFinalPosition1() : Position
+ getInitialPosition2() : Position
+ getFinalPosition2() : Position

**GlazingHammerEvent**

.

---

.

**WindowPatternChoiceEvent**

- choice : int
- name : String

---

+ getChoice(): int
+ getName() : String

**suspendedEvent**

: String

---

ame() : String

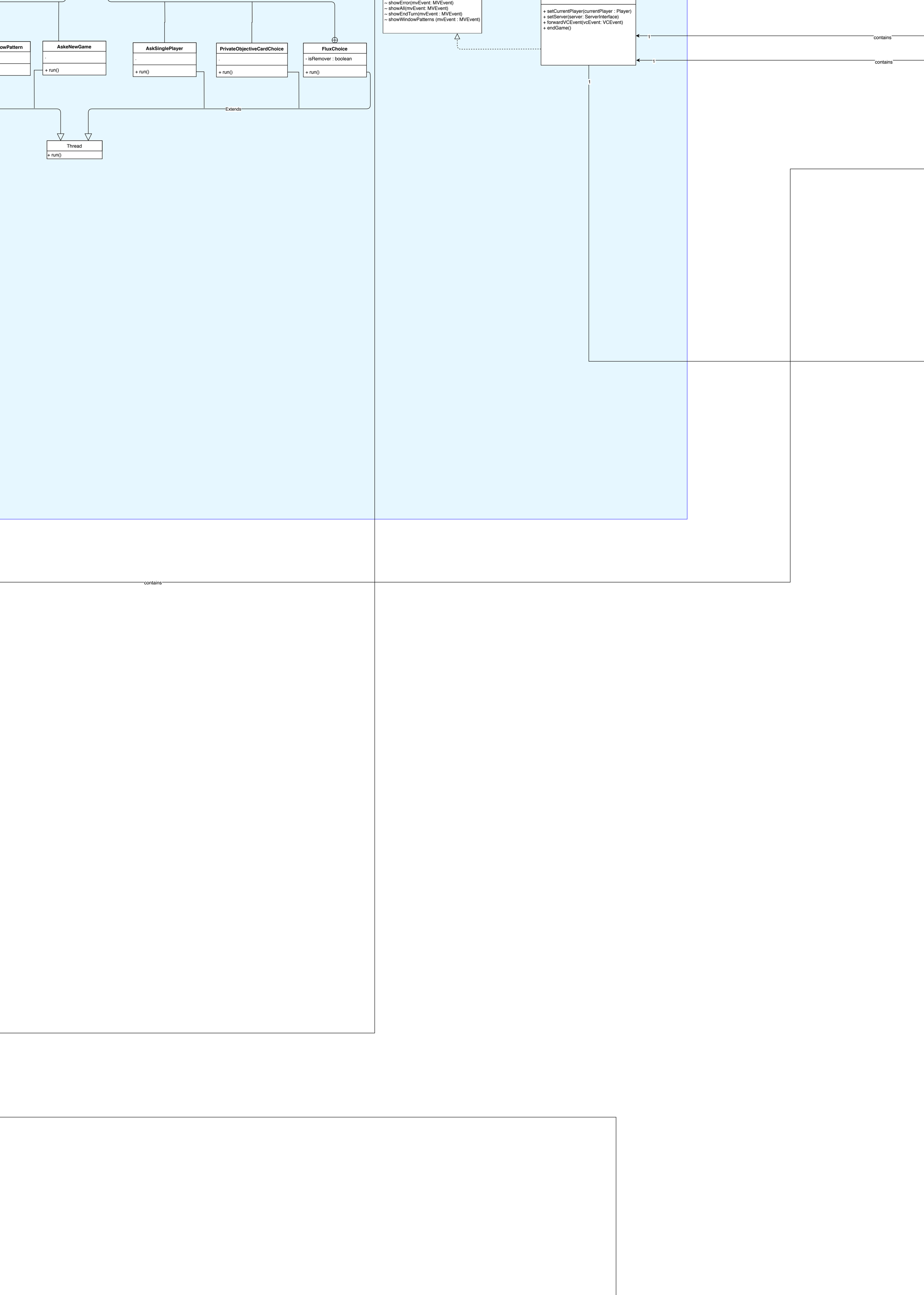
**PrivateObjectiveCardChosen**

- indexOfChosenCard

---

+ getIndexOfChosenCard() : int

contains


owPattern
AskeNewGame
.
+ run()
AskSinglePlayer
.
+ run()
PrivateObjectiveCardChoice
.
+ run()
FluxChoice
- isRemover : boolean
+ run()
Extends
Thread
+ run()
~ showError(mvEvent: MVEvent)
~ showAll(mvEvent: MVEvent)
~ showEndTurn(mvEvent : MVEvent)
~ showWindowPatterns (mvEvent : MVEvent)
+ setCurrentPlayer(currentPlayer : Player)
+ setServer(server: ServerInterface)
+ forwardVCEvent(vcEvent: VCEvent)
+ endGame()
1
contains
1
contains
1
contains

- addClientHelper(player : Player)
- handleSocketLostClient(client : ClientInterfaceSocket)
- handlerRMILostClient(client : ClientInterfaceRMI)
- sendTo(mvEvent : MVEvent, clientInterface : ClientInterfaceRMI)
- sendTo(mvEvent : MVEvent, clientInterface : ClientInterfaceSocket)
- testConnection()
+ removeClient(client : ClientInterfaceSocket)

**WindowPatternCountDown**

.

+ run()

1

contains

contains

contains

model

**GameSetupSi**

- instance : GameSetupSingleton

- GameSetupSingleton (players: ArrayList <Player
privateObjectiveCardDeck
publicObjectiveCardDeck
windowPatternCardDeck:
- choosePlayersOrder ()
+ addPlayers(players : List<Players>)
+ addPlayers(players: List<Player>
+ assignWindowPattern()
- assingPrivateObjective()
+ createNewGame() : GameSingleton
+ instance () : GameSetupSingleton
+ instanceToNull()
- getPrivateObjectiveCardList(numberToExtract : i
- getPublicObjectiveCardList(numberToExtract : int
-getWindowPatternCardList(numberToExtract : int)
+ getPlayers() : List<Player>
- getToolCardList(numberToExtract : int) : List<Too

- cardDeck: I

- mix()
- distribute ()
+ mixAndDis

controller

**Controller**

- eventsHandler : Map<Integer, Runnable>
- nonValidInput: String
- event : VCEvent
- turnDuration : long
- turnEnded : boolean
- turnTimer : TurnTimer
- turnLock : Object
- lock : Object
- singlePlayer : boolean
- diceIndexSinglePlayer : int
- EMPTY_POSITION :String
- RESTRICTION_VIOLATED: String
- INSERTION_DENIED : String

+ createMap()
+ getDiceForFlux () : Dice
+ setDiceForFlux()
- getDiceFromDraftPool ( dieIndex: int) : Dice
- getDiceFromRoundTrack ( round : int, index : int) : Dice
- getDiceFromDicePattern ( position : Position) : Dice
- setWindowPatternPlayer()
- canGameStart()
- handleFavorTokensNumber ( toolCard : ToolCard)
- searchToolCard(id : int) : ToolCard
- grozingPliersSubOne(diceChosen : Dice)
- grozingPliersAddOne(diceChosen : Dice)
- grozingPliers ()
- eglomiseBrushValidRestriction(initialPosition : Position, finalPosition : Position)
- eglomiseBrush()
- copperFoilBurnisherValidRestriction(initialPosition : Position, finalPosition : Position)
- copperFoilBurnisher ()
- lathekinValidRestriction (initialPosition1 : Position, finalPosition1 : Position, initialPosition2 : Position, finalPosition2 : Position)
- lathekin()
- swapDice ( round: int, indexOfRoundTrack : int , indexOfDraftPool : int)
- lensCutterHelper(roundIndex : int)
- lensCutter ()
- fluxBrushPlaceDiceHelper(finalPosition : Position, diceChosen : Dice) : boolean
- fluxBrushPlaceDice()
- fluxBrushChooseDice()
- glazingHammer()
- runnerPliersHelper(finalPosition : Position, diceChosen : Dice)
- runnerPliers()
- corkBakedStringedge ()
- grindingStone()
- fluxRemoverPlaceDice()
- fluxRemoverChoiceDice ()
- tapWheelOneDiceMoved(initialPosition1 : Position, finalPosition1 : Position)
- tapWheelTwoDiceMove(initialPosition1 : Position, finalPosition1 : Position, initialPosition2 : Position, finalPosition2 : Position)
- tapWheel()
- unsuspendedPlayer()
- performAction( event : VCEvent)
- placedDiceFromDraftPoolToDicePattern()
- skipTurn()

contains

contains

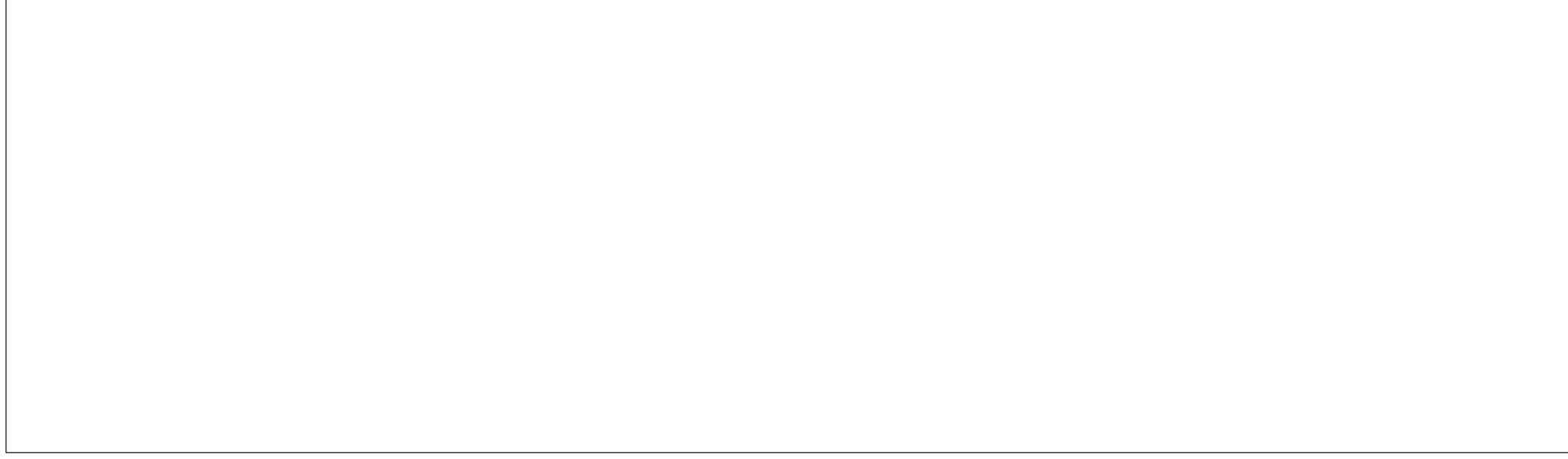
contains

contains

contains

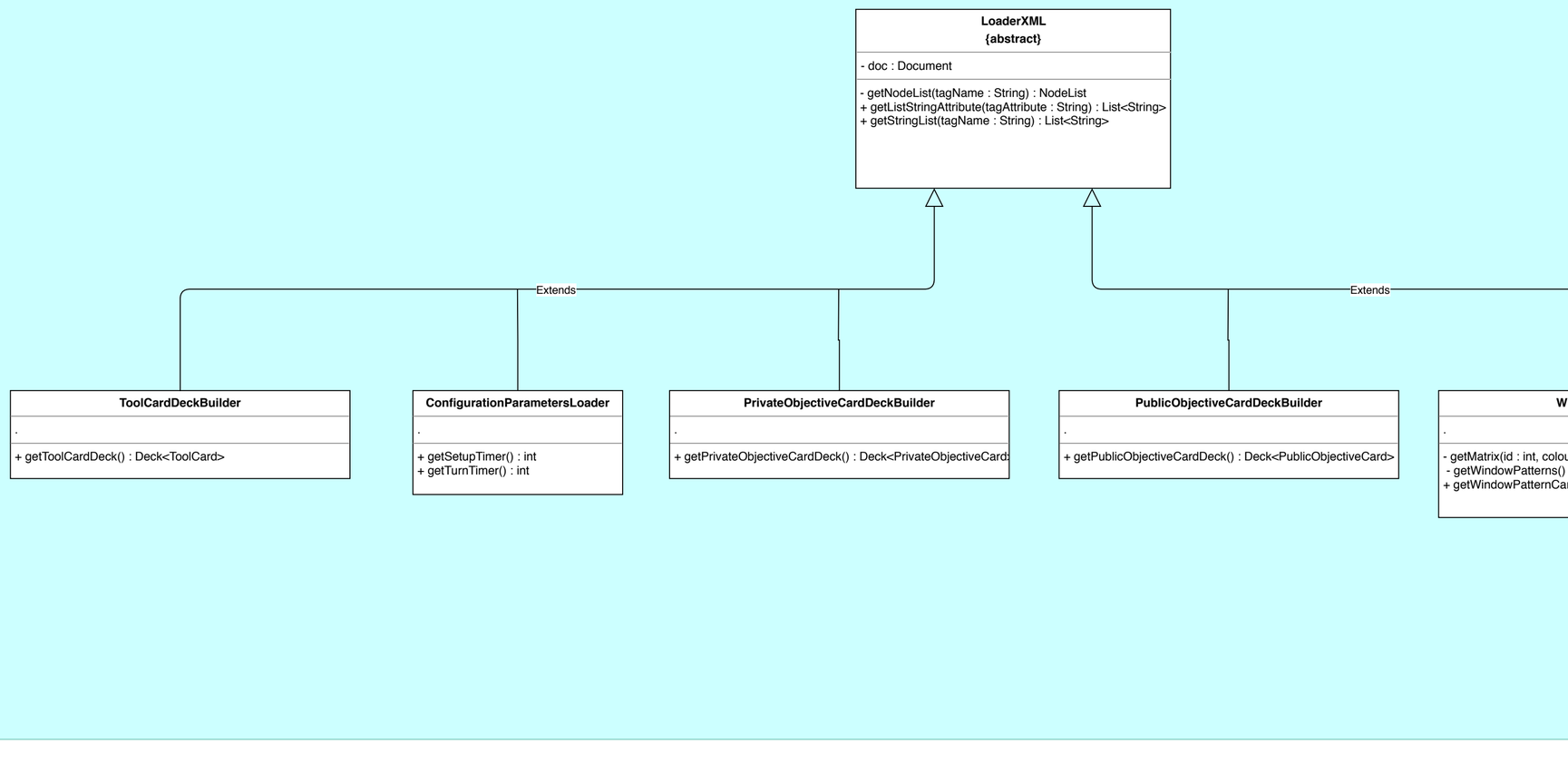
Dice): boolean

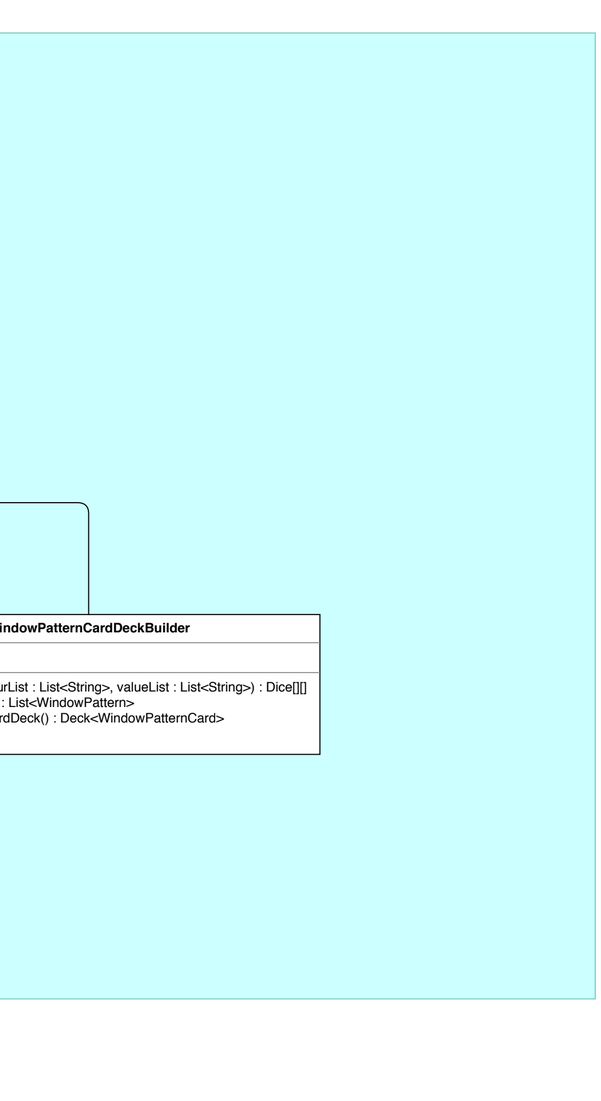
contains

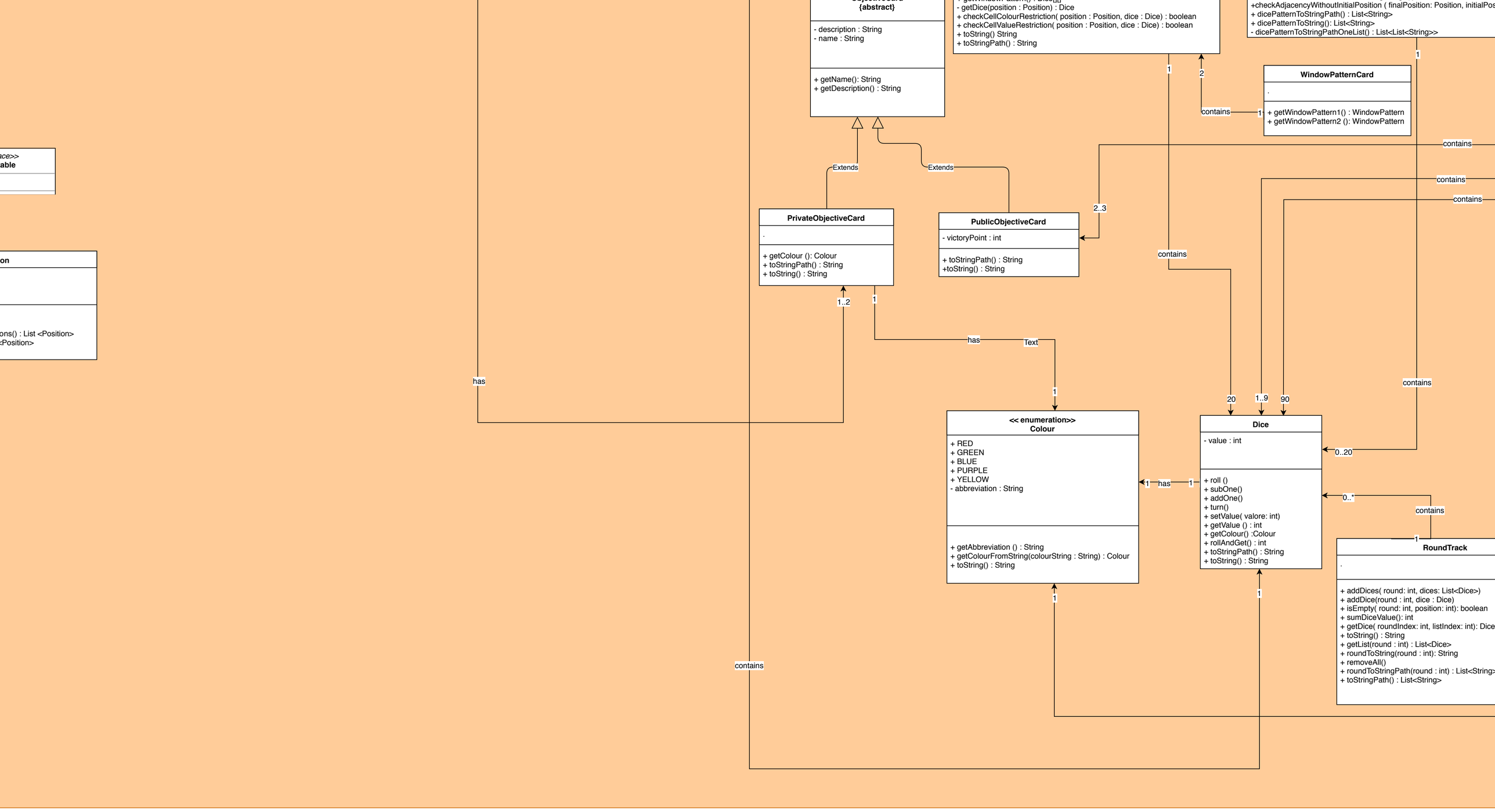


contains

utilities


LoaderXML
{abstract}
- doc : Document
- getNodeList(tagName : String) : NodeList
+ getListStringAttribute(tagAttribute : String) : List<String>
+ getStringList(tagName : String) : List<String>
Extends
Extends
ToolCardDeckBuilder
.
+ getToolCardDeck() : Deck<ToolCard>
ConfigurationParametersLoader
.
+ getSetupTimer() : int
+ getTurnTimer() : int
PrivateObjectiveCardDeckBuilder
.
+ getPrivateObjectiveCardDeck() : Deck<PrivateObjectiveCard>
PublicObjectiveCardDeckBuilder
.
+ getPublicObjectiveCardDeck() : Deck<PublicObjectiveCard>
W
.
- getMatrix(id : int, colou
- getWindowPatterns()
+ getWindowPatternCa


indowPatternCardDeckBuilder
rList : List<String>, valueList : List<String>) : Dice[][]
: List<WindowPattern>
rdDeck() : Deck<WindowPatternCard>
5
Positi
- x: int
- y : int
- serialVersionUID : long
+ getX () : int
+ getY () : int
+ getOrthogonalAdjacentPositi
+ getAdjacentPositions(): List <
Serializ


ace>>
able
on
ons() : List <Position>
<Position>
- description : String
- name : String
+ getName(): String
+ getDescription() : String
- getDice(position : Position) : Dice
+ checkCellColourRestriction( position : Position, dice : Dice) : boolean
+ checkCellValueRestriction( position : Position, dice : Dice) : boolean
+ toString() String
+ toStringPath() : String
+checkAdjacencyWithoutInitialPosition ( finalPosition: Position, initialPo
+ dicePatternToStringPath() : List<String>
+ dicePatternToString(): List<String>
- dicePatternToStringPathOneList() : List<List<String>>
WindowPatternCard
.
+ getWindowPattern1() : WindowPattern
+ getWindowPattern2 (): WindowPattern
contains
Extends
Extends
PrivateObjectiveCard
.
+ getColour (): Colour
+ toStringPath() : String
+ toString() : String
PublicObjectiveCard
- victoryPoint : int
+ toStringPath() : String
+toString() : String
has
Text
has
<< enumeration>>
Colour
+ RED
+ GREEN
+ BLUE
+ PURPLE
+ YELLOW
- abbreviation : String
+ getAbbreviation () : String
+ getColourFromString(colourString : String) : Colour
+ toString() : String
Dice
- value : int
+ roll ()
+ subOne()
+ addOne()
+ turn()
+ setValue( valore: int)
+ getValue () : int
+ getColour() :Colour
+ rollAndGet() : int
+ toStringPath() : String
+ toString() : String
RoundTrack
.
+ addDices( round: int, dices: List<Dice>)
+ addDice(round : int, dice : Dice)
+ isEmpty( round: int, position: int): boolean
+ sumDiceValue(): int
+ getDice( roundIndex: int, listIndex: int): Dice
+ toString() : String
+ getList(round : int) : List<Dice>
+ roundToString(round : int): String
+ removeAll()
+ roundToStringPath(round : int) : List<String>
+ toStringPath() : List<String>
contains
contains
contains
contains
contains
contains
contains
1
2
2..3
1..2
1
1
20
1..9
90
0..20
0..*
1
1
1
1
1
1
1
1

ition : Position) : boolean

1 1 1 1 1..*

**GameSingleton**

- playersNumber : int
- diceNumberToExtract: int
- round: int
- instance : GameSingleton
- lap : int

- gameSingleton(players: ArrayList <Player>, publicObjectiveCards: ArrayList<PublicObjectiveCard>, toolCards : ArrayList <ToolCard>, scoreTrack: ScoreTrack, roundTrack: RoundTrack )
+ increaseRound()
+ extractAndRoll ()
+ fromDraftPoolToRoundTrack ()
+ selectWinner(): Player
+ instance() : GameSingleton
- createDiceBag() : List<Dice>
+ instanceToNull()
+ getToolCards(): List<ToolCard>
+ getPublicObjectiveCards(): List<PublicObjectiveCard>
+ getCurrentPlayer() : Player
+ getDraftPool(): List<Dice>
+ getRoundTrack(): RoundTrack
+ getPlayersNumber() : int
+getRound(): int
+ getPlayers(): List<Player>
+ getDiceBag(): List<Dice>
+ setRound(round: int)
+ extractAndRollOneDiceWithoutReturning()
+ extractAndRollOneDice(): Dice
+ draftPoolToString() : List<String>
+ draftPoolToStringPath(): List<String>
+dicePatternsToString():List <String>
+toolCardsToString():List <String>
+toolCardsToStringAbbreviated():List <String>
+toolCardsToStringPath():List <String>
+ playersToString():List <String>
+publicObjectiveCardsToStringPath():List <String>
+dicePatternsToStringPath():List<List<String>>
+ windowPatternsToStringPath():List <String>
+getFavorTokensOnToolCards():List <String>
+getFavorTokensNumberPlayers():List <String>
+publicObjectiveCardsToString():List <String>
+ myNotify(event: MVEvent)
+ getInstance(): GameSingleton
+ rollEveryDice()
+ createScoreTrack(scores: List<Integer>)
+ lastPlayer()+ setCurrentPlayer(currentPlayer: Player)
+ setLap( lap: int)
+ getLap(): int

1 contains 3

**ToolCard**

- description: String
- name: String
- favorPoint: int
- id : int

+ getFavorPoint(): int
+ getColour () : Colour
+ getDescription() : String
+ getName () : String
+ increaseFavorPoint(n: int)
+ toString(): String
+ getId() : int
+ toStringAbbreviated() : String
+ toStringPath() : String

1 1 contains 1

has