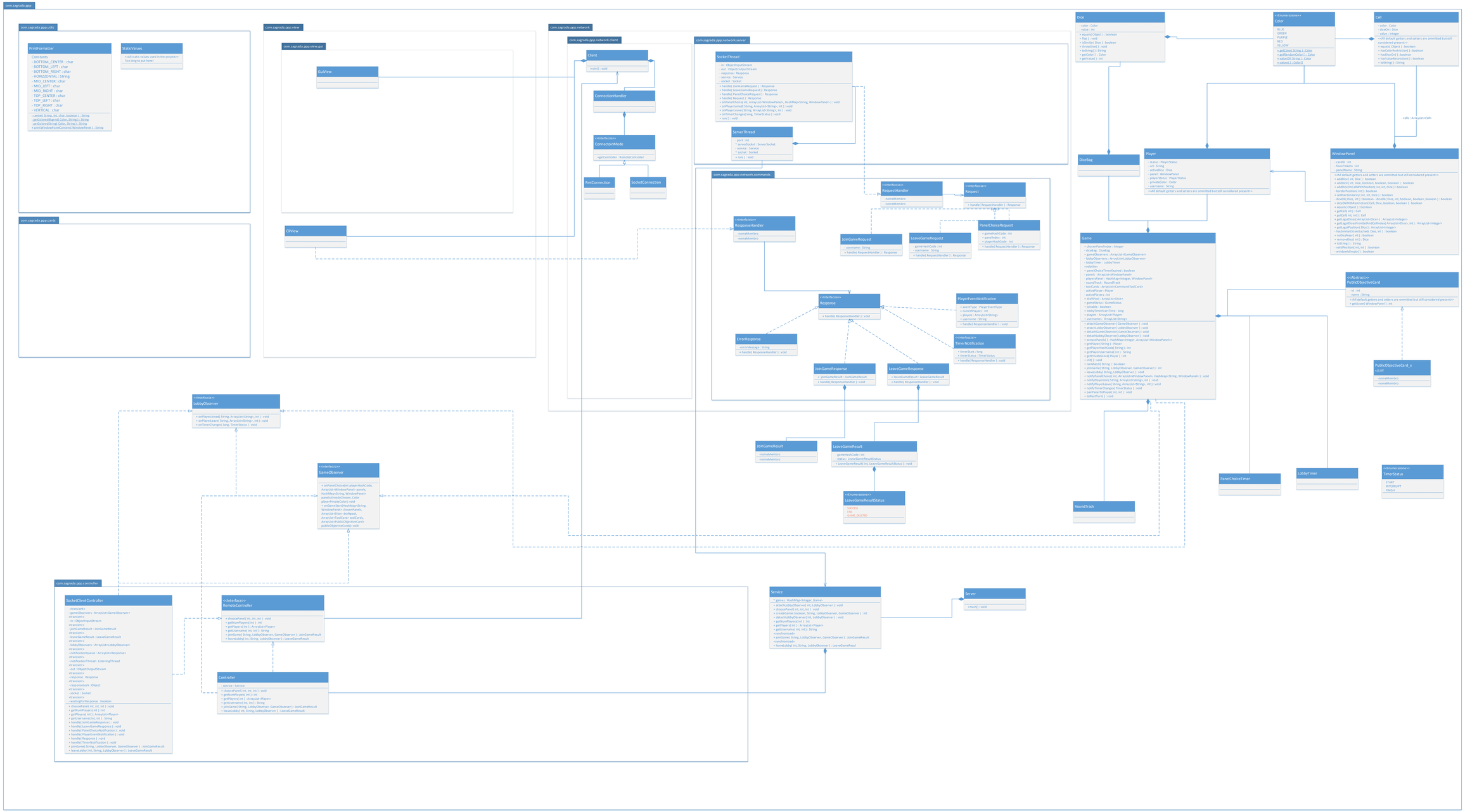
com.sagrada.app
com.sagrada.app.utils
PrintFormatter
Constants
- BOTTOM_CENTER : char
- BOTTOM_LEFT : char
- BOTTOM_RIGHT : char
- HORIZONTAL : String
- MID_CENTER : char
- MID_LEFT : char
- MID_RIGHT : char
- TOP_CENTER : char
- TOP_LEFT : char
- TOP_RIGHT : char
- VERTICAL : char
- center(String, int, char, boolean) : String
- getColoredBlank(Color, String) : String
- getColoredString(Color, String) : String
StaticValues
<<All static values used in the project>>
- Too long to put here!
com.sagrada.app.view
com.sagrada.app.view.gui
GuiView
CSView
com.sagrada.app.network
com.sagrada.app.network.client
Client
- main() : void
ConnectionHandler
<<Interfaccia>>
ConnectionMode
+getController : RemoteController
RmiConnection
SocketConnection
com.sagrada.app.network.server
SocketThread
- in : ObjectInputStream
- out : ObjectOutputStream
- response : Response
- server : Server
- socket : Socket
+ handle(JoinGameRequest) : Response
+ handle(LeaveGameRequest) : Response
+ handle(PanelChoiceRequest) : Response
+ handle(Request) : Response
+ onPanelChoice(int, ArrayList<WindowPanel>, HashMap<String, WindowPanel>) : void
+ onPlayerJoined(String, ArrayList<String>, int) : void
+ onPlayerLeaved(String, ArrayList<String>, int) : void
+ onTimerChanged(long, TimerStatus) : void
+ run() : void
ServerThread
- port : int
- serverSocket : ServerSocket
- server : Server
- socket : Socket
+ run() : void
com.sagrada.app.network.commands
<<Interfaccia>>
RequestsHandler
-nomeMembro
-nomeMembro
<<Interfaccia>>
Request
+ handle(RequestsHandler) : Response
<<Interfaccia>>
ResponseHandler
-nomeMembro
-nomeMembro
JoinGameRequest
-username : String
+ handle(RequestHandler) : Response
LeaveGameRequest
- gameHashCode : int
- username : String
+ handle(RequestHandler) : Response
PanelChoiceRequest
+ gameHashCode : int
+ panelIndex : int
+ playerHashCode : int
+ handle(RequestHandler) : Response
<<Interfaccia>>
Response
+ handle(ResponseHandler) : void
PlayerEventNotification
+ eventType : PlayerEventType
+ players : ArrayList<String>
+ username : String
+ handle(ResponseHandler) : void
ErrorResponse
-errorMessage : String
+ handle(ResponseHandler) : void
<<Interfaccia>>
TimerNotification
+ timerStart : long
+ timerStatus : TimerStatus
+ handle(ResponseHandler) : void
JoinGameResponse
+ joinGameResult : JoinGameResult
+ handle(ResponseHandler) : void
LeaveGameResponse
+ leaveGameResult : LeaveGameResult
+ handle(ResponseHandler) : void
JoinGameResult
-nomeMembro
-nomeMembro
LeaveGameResult
- gameHashCode : int
- status : LeaveGameResultStatus
+ LeaveGameResult(int, LeaveGameResultStatus) : void
<<Enumerazione>>
LeaveGameResultStatus
SUCCESS
FAIL
GAME_DELETED
Dice
- color : Color
- value : int
+ equals(Object) : boolean
+ flip() : void
+ isSimilar(Dice) : boolean
+ throwDice() : void
+ getValue() : int
+ getColor() : Color
+ setValue( ) : int
<<Enumerazione>>
Color
BLUE
GREEN
PURPLE
RED
YELLOW
+ getColor(String) : Color
+ getRandomColor( ) : Color
+ value(CS String) : Color
+ value( ) : Color[ ]
Cell
- color : Color
- diceOn : Dice
- value : Integer
<<All default getters and setters are ommitted but still considered present>>
+ equals(Object) : boolean
+ isEmpty( ) : boolean
+ hasDiceOn( ) : boolean
+ hasValueRestriction( ) : boolean
+ toString( ) : String
- cells : ArrayList<Cell>
DiceBag
Player
- Status : PlayerStatus
- id : int
- activeDice : Dice
- panel : WindowPanel
- playerStatus : PlayerStatus
- privateColor : Color
- username : String
<<All default getters and setters are ommitted but still considered present>>
WindowPanel
- CardID : int
- gameID : int
- gameName : String
<<All default getters and setters are ommitted but still considered present>>
+ addDiceI(int, Dice) : boolean
+ addDice(int, Dice, boolean, boolean, boolean) : boolean
+ addDiceOnCellWithPosition(int, int, Dice) : boolean
- borderPosition(int) : boolean
- certifyPlaceable(int, int, Dice) : boolean
- diceOk(Dice, int) : boolean - diceOk(Dice, int, boolean, boolean, boolean) : boolean
+ diceOnWithRestriction(Cell, Dice, boolean) : boolean
+ equals(Object) : boolean
+ getCell(int) : Cell
+ getCell(int, int) : Cell
+ getLegalPositions(Dice) : ArrayList<Integer>
+ getLegalPositionsHaveNoCellNeed(ArrayList<Dice>, int) : ArrayList<Integer>
+ getLegalPositions(Dice) : ArrayList<Integer>
- hasSimilarDiceAttached(Dice, int) : boolean
+ noDiceMoss( ) : boolean
+ removeDice(int) : Dice
+ toString( ) : String
- validPosition(int, int) : boolean
- windowIsEmpty( ) : boolean
Game
+ chosenPanelIndex : Integer
- diceBag : DiceBag
- gameObservers : ArrayList<GameObserver>
- lobbyObservers : ArrayList<LobbyObserver>
- lobbyTimer : LobbyTimer
+ isStatic+
+ panelChoiceTimerExpired : boolean
- panels : ArrayList<WindowPanel>
- playersPanel : HashMap<Integer, WindowPanel>
- roundTrack : RoundTrack
- toolCards : ArrayList<CommandToolCard>
- activePlayer : Player
- activePlayers : int
- availableDices : ArrayList<Dice>
+ gameStatus : GameStatus
+ joinable : boolean
- lobbyTimerStartTime : long
+ players : ArrayList<Player>
+ usernames : ArrayList<String>
+ attachGameObserver(GameObserver) : void
+ attachLobbyObserver(LobbyObserver) : void
+ detachGameObserver(GameObserver) : void
+ detachLobbyObserver(LobbyObserver) : void
+ extractPanels( ) : HashMap<Integer, ArrayList<WindowPanel>>
+ getPlayer(String) : Player
+ getPlayerHashCode(String) : int
+ getPlayerUsernameFrom( ) : String
+ getPrivateColorof(Player) : int
+ init( ) : void
+ isInMatch(String) : boolean
+ joinGame(String, LobbyObserver, GameObserver) : int
+ leaveLobby(String, LobbyObserver) : void
+ notifyPanelChoice(int, ArrayList<WindowPanel>, HashMap<String, WindowPanel>) : void
+ notifyPlayerJoin(String, ArrayList<String>, int) : void
+ notifyPlayerLeave(String, ArrayList<String>, int) : void
+ notifyTimerChanged(TimerStatus) : void
+ parPanelToPlayer(int, int) : void
+ toNextTurn( ) : void
<<Abstract>>
PublicObjectiveCard
- id : int
- name : String
<<All default getters and setters are ommitted but still considered present>>
+ getScore(WindowPanel) : int
PublicObjectiveCard_x
<<Enum>>
-nomeMembro
-nomeMembro
PanelChoiceTimer
LobbyTimer
<<Enumerazione>>
TimerStatus
START
INTERRUPT
FINISH
RoundTrack
<<Interfaccia>>
LobbyObserver
+ onPlayerJoined(String, ArrayList<String>, int) : void
+ onPlayerLeaved(String, ArrayList<String>, int) : void
+ onTimerChanged(long, TimerStatus) : void
<<Interfaccia>>
GameObserver
+ onPanelChoice(int playerHashCode, ArrayList<WindowPanel> panels, HashMap<String, WindowPanel> panelsAlreadyChosen, Color playerPrivateColor): void
+ onGameStart(HashMap<String, WindowPanel> chosenPanels, ArrayList<Dice> diceSpool, ArrayList<ToolCard> toolCards, ArrayList<PublicObjectiveCard> publicObjectiveCards) :void
com.sagrada.app.controller
SocketClientController
<transient>
- gameObservers : ArrayList<GameObserver>
<transient>
- in : ObjectInputStream
<transient>
- joinGameResult : JoinGameResult
<transient>
- leaveGameResult : LeaveGameResult
<transient>
- lobbyObservers : ArrayList<LobbyObserver>
<transient>
- notificationQueue : ArrayList<Response>
<transient>
- notificationThread : ListeningThread
<transient>
- out : ObjectOutputStream
<transient>
- response : Response
<transient>
- waitingAck : Object
<transient>
- socket : Socket
<transient>
- waitingForResponse : boolean
+ choosePane(int, int, int) : void
+ getNumPlayers(int) : int
+ getPlayers(int) : ArrayList<Player>
+ getUsername(int, int) : String
+ handle(JoinGameResponse) : void
+ handle(LeaveGameResponse) : void
+ handle(PanelChoiceNotification) : void
+ handle(PlayerEventNotification) : void
+ handle(Response) : void
+ handle(TimerNotification) : void
+ joinGame(String, LobbyObserver, GameObserver) : JoinGameResult
+ leaveLobby(int, String, LobbyObserver) : LeaveGameResult
<<Interfaccia>>
RemoteController
+ choosePane(int, int, int) : void
+ getNumPlayers(int) : int
+ getPlayers(int) : ArrayList<Player>
+ getUsername(int, int) : String
+ joinGame(String, LobbyObserver, GameObserver) : JoinGameResult
+ leaveLobby(int, String, LobbyObserver) : LeaveGameResult
Controller
- service : Service
+ choosePane(int, int, int) : void
+ getNumPlayers(int) : int
+ getPlayers(int) : ArrayList<Player>
+ getUsername(int, int) : String
+ joinGame(String, LobbyObserver, GameObserver) : JoinGameResult
+ leaveLobby(int, String, LobbyObserver) : LeaveGameResult
Service
- games : HashMap<Integer, Game>
+ attachLobbyObserver(int, LobbyObserver) : void
+ choosePane(int, int, int) : void
+ createGame(boolean, String, LobbyObserver, GameObserver) : int
+ detachLobbyObserver(int, LobbyObserver) : void
+ getNumPlayers(int) : int
+ getPlayers(int) : ArrayList<Player>
+ getUsername(int, int) : String
<synchronized>
+ joinGame(String, LobbyObserver, GameObserver) : JoinGameResult
<synchronized>
+ leaveLobby(int, String, LobbyObserver) : LeaveGameResul
Server
+ main() : void