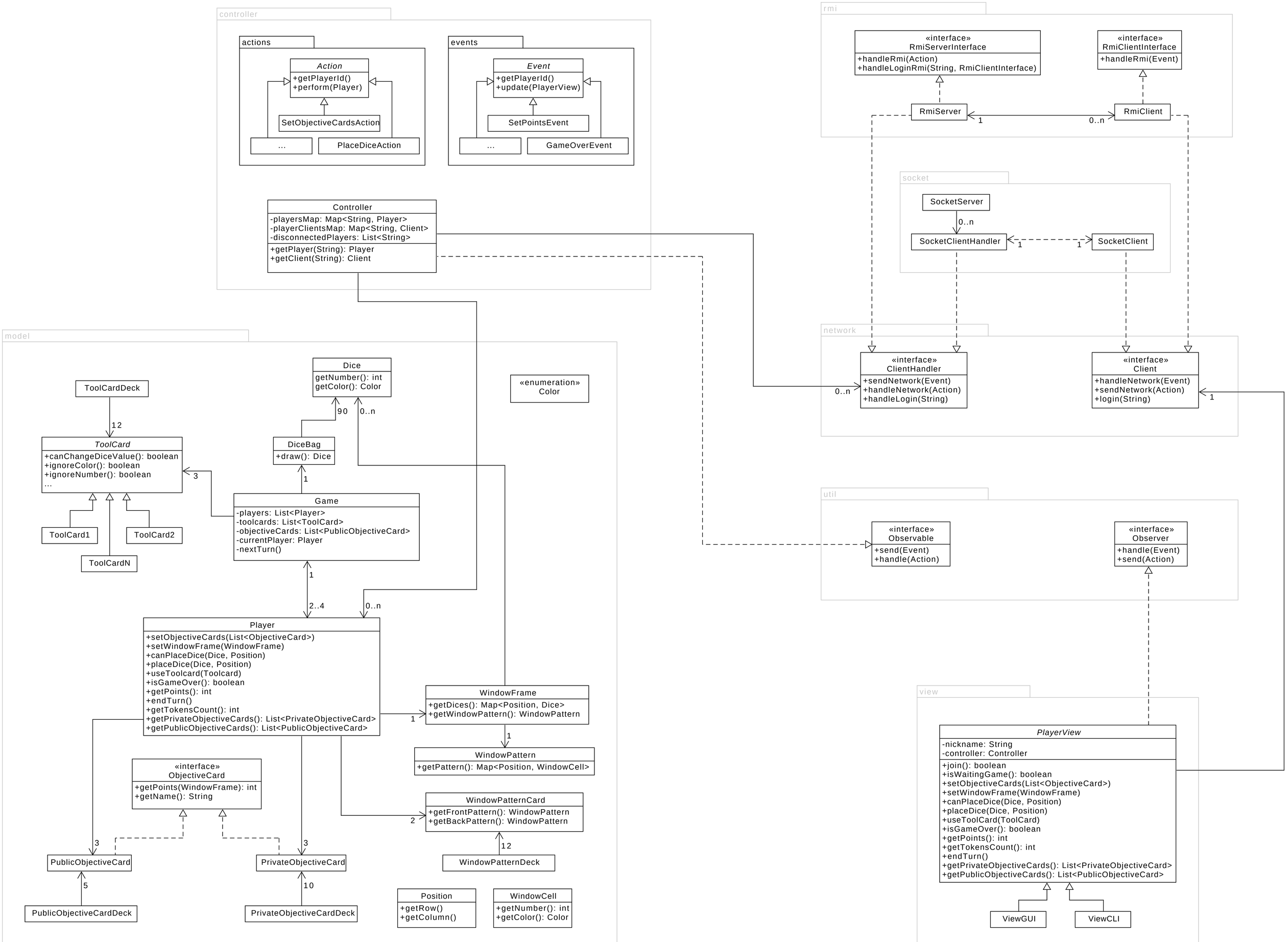
controller
actions
Action
+getPlayerId()
+perform(Player)
SetObjectiveCardsAction
...
PlaceDiceAction
events
Event
+getPlayerId()
+update(PlayerView)
SetPointsEvent
...
GameOverEvent
Controller
-playersMap: Map<String, Player>
-playerClientsMap: Map<String, Client>
-disconnectedPlayers: List<String>
+getPlayer(String): Player
+getClient(String): Client
rmi
«interface»
RmiServerInterface
+handleRmi(Action)
+handleLoginRmi(String, RmiClientInterface)
«interface»
RmiClientInterface
+handleRmi(Event)
RmiServer
RmiClient
1
0..n
socket
SocketServer
0..n
SocketClientHandler
1
1
SocketClient
network
«interface»
ClientHandler
+sendNetwork(Event)
+handleNetwork(Action)
+handleLogin(String)
0..n
«interface»
Client
+handleNetwork(Event)
+sendNetwork(Action)
+login(String)
1
model
ToolCardDeck
12
ToolCard
+canChangeDiceValue(): boolean
+ignoreColor(): boolean
+ignoreNumber(): boolean
...
3
ToolCard1
ToolCard2
ToolCardN
Dice
getNumber(): int
getColor(): Color
90
0..n
DiceBag
+draw(): Dice
1
«enumeration»
Color
Game
-players: List<Player>
-toolcards: List<ToolCard>
-objectiveCards: List<PublicObjectiveCard>
-currentPlayer: Player
-nextTurn()
1
2..4
0..n
Player
+setObjectiveCards(List<ObjectiveCard>)
+setWindowFrame(WindowFrame)
+canPlaceDice(Dice, Position)
+placeDice(Dice, Position)
+useToolcard(Toolcard)
+isGameOver(): boolean
+getPoints(): int
+endTurn()
+getTokensCount(): int
+getPrivateObjectiveCards(): List<PrivateObjectiveCard>
+getPublicObjectiveCards(): List<PublicObjectiveCard>
1
WindowFrame
+getDices(): Map<Position, Dice>
+getWindowPattern(): WindowPattern
1
WindowPattern
+getPattern(): Map<Position, WindowCell>
«interface»
ObjectiveCard
+getPoints(WindowFrame): int
+getName(): String
WindowPatternCard
+getFrontPattern(): WindowPattern
+getBackPattern(): WindowPattern
2
3
3
PublicObjectiveCard
PrivateObjectiveCard
12
WindowPatternDeck
5
10
PublicObjectiveCardDeck
PrivateObjectiveCardDeck
Position
+getRow()
+getColumn()
WindowCell
+getNumber(): int
+getColor(): Color
util
«interface»
Observable
+send(Event)
+handle(Action)
«interface»
Observer
+handle(Event)
+send(Action)
view
PlayerView
-nickname: String
-controller: Controller
+join(): boolean
+isWaitingGame(): boolean
+setObjectiveCards(List<ObjectiveCard>)
+setWindowFrame(WindowFrame)
+canPlaceDice(Dice, Position)
+placeDice(Dice, Position)
+useToolCard(ToolCard)
+isGameOver(): boolean
+getPoints(): int
+getTokensCount(): int
+endTurn()
+getPrivateObjectiveCards(): List<PrivateObjectiveCard>
+getPublicObjectiveCards(): List<PublicObjectiveCard>
ViewGUI
ViewCLI