# Client Environment Model Diagram

**<<enumeration>>**
Request_name

-finalStatus
- pause
- pauseOver
- backup

- readRule()
- getCharacterInfo()
- goBack()
- signIn(idp: Player ID, psw: Password)
- changeSettings()
- modifySettings(s: Settings; v: Value)
- cancelModification()
- sendChat (m: message)
- createGame()
- newLobbyInfoAck(n: name, mx: Maximum Player, mn: Minimum Player)
- chooseGame()
- chosenGameAck(idg: Game ID)
- startGame()
- loadGame()
- chosenSavedGameAck(idg: Game ID)
- openMenu()
- closeMenu()
- callAPause()
- pauseIsOver()
- saveGame()
- specifyBackupLocation()
- quitGame()
- leavingGameAck()

- notLeavingGameAck
- closeProgram()
- closeApplicationAck()
- notCloseApplicationAck()
-backToLobbyAck()
- startPhaseOne(g:Game)
- positionCharacter(c:Position)
- doAction(a:Phase1Action)
- chooseCard(c:Player)
- useNormalWhiskey(w:Player)
- useOldWhiskey(w:Player)
- pickThreeActionCard(p:Player)
- Time-triggered: timer()
- chosenMove(p: Position)
- chosenRideMove(p: Position)
- chosenRobberyLoot(l: Loot)
- chosenBanditFire(b: Bandit)
- chosenCharacterPunch(b: Bandit)
- chosenMarshalMove(p: Position)
- chosenHostage(h: Hostage)
- randomHostage(h: Hostage)
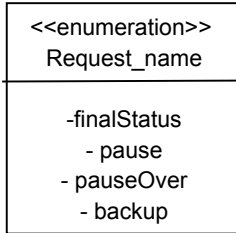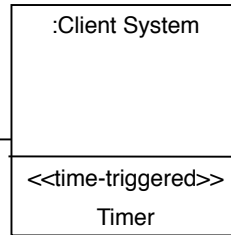
- request (re: request_name)
- checkMessage(m: message)
- playerLeavingServer(idg: Game ID)
- changeSettings(s: Settings, v: Value)
- cancelModificationRequest()
- canPlayerLeaveGame()
- initializeNewGame()
- nextPlayerAck()
- nextTurnAck()
- nextRoundAck()

- positionCharacterRequest(r:Actionequest)
- chooseCardRequest(r:ActionRequest)
- normalWhiskeyRequest(r:ActionRequest)
- oldWhiskeyRequest(r:ActionRequest)
- ActionCardRequest(r:ActionRequest)
- validateChosenMove(p: Position)
- validateChosenRideMove(p: Position)
- validateChosenRobberyLoot(l: Loot)
- validateChosenBanditFire(b: Bandit)
- validateChosenCharacterPunch(pc: PunchableCharacter)
- validateChosenMarshalMove(p: Position)
- validateChosenHostage(h: Hostage)

:Client System

<<time-triggered>>
Timer

:Server

:GUI
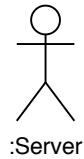
- sendPlayersStatus()
- messageBroadcasted()
- messagesNotBroadcasted()
- requestReceived()
- gamePaused()
- gameResumed()
- gameCantBePaused
- gameBackupReady()
- gameBackupNotPossible()
- settingsChangedAck()
- modificationCanceledAck()
- playerCanLeaveGameAck()
- playerCanNotLeaveGameAck()
- leaveGameServerAck()
- initializeNewGameAck()
- nextPlayer(p: Player)
- nextTurn(t: TurnType)
- nextRound(Integer: currentRound)

- characterCabin(c: Position)
-stateChange(p:Phase1Action)
- positionCharacterInvalid(s: String)
- normalWhiskeyUsed(m:String)
- normalWhiskeyInvalid(s: String)
- playerChoosedCard(m: String)
- oldWhiskeyUsed(m: String)
- oldWhiskeyInvalid(s: String)
- ActionCardDistributed(m:String)
- promptMoveOptions(ps: Set{Position})
- promptRideOptions(ps: Set{Position})
- promptRobberyOptions(l: Set{Loot})
- promptFireOptions(bs: Set{Bandit})
- promptPunchOptions(bs: Set{Bandit})
- promptMarshalMoveOptions(ps: Set{Position})
- promptHostageOptions(hs: Set{Hostage})
- refreshGameState(pa: Phase2Action)
- invalidRequest_e(m: String)

- displayRules()
- goToPreviousWindow()
- displayCharacterInfo()
- signInSucceded()
- incorrectCredentials()
- displaySettings()
- displayChat(m: message)
- displayGameToChoose()
- needMorePlayers()
- chooseName()
- chooseMinPlayer()
- chooseMaxPlayer()
- startColtExpress()
- displaySavedGames()
- displayMenu()
- gamePaused()
- pauseIsOver()
- gameSuccessfullySaved()
- leavingGameConfirmation()
- closeApplicationConfirmation()

- requestAction()
- randomAction()
- positionYourCharacter()
- positionCharacterFail(s:String)
- currentRoundCard(s:Game)
- currentTurn(t:Game)
- distributeCard(c:Player)
- currentPosition(c:Position)
- moveCard(c:Player)
- returnThreeCard(c: Player)
- chooseCardFail(s:String)
- normalWhiskeyFail(s:String)
- oldWhiskeyFail(s:String)
- pickThreeCardFail(s:String)
- displayMoveOptions(ps: Set{Position})
- displayRideOptions(ps: Set{Position})
- displayRobberyOptions(l: Set{Loot})
- displayFireOptions(bs: Set{Bandit})
- displayPunchOptions(bs: Set{Bandit})

- networkConnectionFailure()          - displayMarshalMoveOptions(ps:
- displayResult()                                      Set{Position})
- goBackToLobby()          - displayHostageOptions(hs: Set{Hostage})
- displayNextPlayer()          - refreshGameState(pa: Phase2Action)
- displayNextTurn()
- displayNextRound()

- chosenGameAvailable()
- signInSuccessful()
- incorrectCredentials()                              - isGameAvailable()
- signInFailed()                  - signInRequest (idp: Player ID, psw: Password)
- gameEnterredAck()                        - enterGame(idg: Game ID)
- notEnoughPlayersAck()                        - isNumberOfPlayerEnough()
- EnoughPlayersAck()                      - playerLeavingLobby(idp: Player ID)
- leaveGameLobbyAck()                       - arePlayerConnected(idp: Player ID)
- playersConnectedAck()          - createNewLobby(n: name, mx: Maximum Player,
- playersNotAllConnectedAck()                          mn: Minimum Player)
- newLobbyCreatedAck()

:Lobby Service