SecondStateVisitorServerImplementation

- server: Server
- message: Message

+ visit(FinPowerChoice): void
+ visit(PLANETS_LandingChoice): void
+ visit(AddAndRearrangeStocks): void
+ visit(EndGame): void
+ visit(ClaimRewardChoice): void
+ visit(GiveUpCrewChoice): void
+ visit(DockingChoice): void
+ visit(MotorPowerChoice): void
+ visit(CardDrawing): void
+ visit(InitializationState): void
+ visit(ShipConstructionState): void
+ visit(ManageProjectile): void
+ visit(TubaFaxesAndFixingShips): void
+ visit(WaitingForPlayers): void

StateHandlerVisitorServerImplementation

- server: Server
- message: Message

+ visit(PLANETS_LandingChoice): void
+ visit(InitializationState): void
+ visit(CardDrawing): void
+ visit(EndGame): void
+ visit(MotorPowerChoice): void
+ visit(WaitingForPlayers): void
+ visit(AddAndRearrangeStocks): void
+ visit(FinPowerChoice): void
+ visit(GiveUpCrewChoice): void
+ visit(DockingChoice): void
+ visit(ManageProjectile): void
+ visit(ClaimRewardChoice): void
+ visit(TubaFaxesAndFixingShips): void
+ visit(ShipConstructionState): void

<<Interface>>
RemoteClientRMI

+ cardActivationRequestMessage(CardActivationRequestMessage): void
+ emailDeckRequestMessage(SmallDeckRequestMessage): void
+ updateServerShipMessage(UpdateServerShipMessage): void
+ getRemoteReference(): ServerRMI
+ waitingPartnersMessage(WaitingPartnersMessage): void
+ stockoTubaMessage(StockoTubaMessage): void
+ endGameMessage(EndGameMessage): void
+ playersNumRequestMessage(PlayersNumRequestMessage): void
+ freeSmallDeckRequestMessage(AvailableSmallDeckRequestMessage): void
+ beginMessage(BeginMessage): void
+ pickRevealedComponentRequestMessage(PickRevealedComponentRequestMe
+ claimRewardChoiceMessage(ClaimRewardChoiceMessage): void
+ discardStockMessage(DiscardStockMessage): void
+ playersPositionMessage(PlayersPositionsMessage): void
+ returnComponentRequestMessage(ReturnComponentRequestMessage): void
+ timeExpiredMessage(TimeExpiredMessage): void
+ resilienceResponseMessage(ResilienceResponseMessage): void
+ logRequestMessage(LogRequestMessage): void
+ drawComponentRequestMessage(DrawComponentRequestMessage): void
+ cardDrawnMessage(CardDrawnMessage): void
+ planetLandRequestMessage(PlanetLandRequestMessage): void
+ resilienceRequestMessage(ResilienceRequestMessage): void
+ logResponseMessage(LogResponseMessage): void
+ premadeShipComponentResponse(PremadeShipComponentResponse): void
+ projectileTrajectoryMessage(ProjectileTrajectoryMessage): void
+ resumeGameResumeGameMessage(): void
+ premadeShipComponentRequest(PremadeShipComponentRequest): void
+ playerNumResponseMessage(PlayersNumResponseMessage): void
+ emailDeckResponseMessage(SmallDeckResponseMessage): void
+ startTimerRequestMessage(StartTimerRequestMessage): void
+ winnerMessage(WinnerMessage): void
+ drawComponentResponseMessage(DrawComponentResponseMessage): void
+ dynamicFinePowerMessage(DynamicFinePowerMessage): void
+ returnSmallDeckMessage(ReturnSmallDeckMessage): void
+ dynamicMotorPowerMessage(DynamicMotorPowerMessage): void
+ updateClientShipMessage(UpdateClientShipMessage): void
+ gameStateMessage(GameStateMessage): void
+ timerStartedMessage(TimerStartedMessage): void
+ freePlanetsResponseMessage(FreePlanetsResponseMessage): void
+ availableSmallDeckMessage(AvailableSmallDeckMessage): void
+ genericMessage(GenericMessage): void
+ drawSpecificComponentRequestMessage(DrawSpecificComponentRequestMess
+ positionsAndTurnsMessage(PositionsAndTurnsMessage): void
+ turnMessage(TurnMessage): void
+ drawCardRequestMessage(DrawCardRequestMessage): void
+ placeShipRequestMessage(PlaceShipRequestMessage): void
+ creditsEarnedMessage(CreditsEarnedMessage): void
+ revealedComponentMessage(RevealedComponentMessage): void

<<Interface>>
ServerRMI

+ projectileTrajectoryMessage(ProjectileTrajectoryMessage): void
+ pickRevealedComponentRequestMessage(PickRevealedComponentRequestMe
+ returnSmallDeckMessage(ReturnSmallDeckMessage): void
+ turnMessage(TurnMessage): void
+ emailDeckRequestMessage(SmallDeckRequestMessage): void
+ positionsAndTurnsMessage(PositionsAndTurnsMessage): void
+ logRequestMessage(LogRequestMessage): void
+ playersNumResponseMessage(PlayersNumResponseMessage): void
+ startTimerRequestMessage(StartTimerRequestMessage): void
+ addClient(RemoteClientRMI, String): void
+ creditsEarnedMessage(CreditsEarnedMessage): void
+ winnerMessage(WinnerMessage): void
+ playersNumRequestMessage(PlayersNumRequestMessage): void
+ revealedComponentMessage(RevealedComponentMessage): void
+ resilienceResponseMessage(ResilienceResponseMessage): void
+ drawComponentResponseMessage(DrawComponentResponseMessage): void
+ updateClientShipMessage(UpdateClientShipMessage): void
+ dynamicFinePowerMessage(DynamicFinePowerMessage): void
+ returnComponentRequestMessage(ReturnComponentRequestMessage): void
+ beginMessage(BeginMessage): void
+ discardStockMessage(DiscardStockMessage): void
+ updateServerShipMessage(UpdateServerShipMessage): void
+ placeShipRequestMessage(PlaceShipRequestMessage): void
+ claimRewardChoiceMessage(ClaimRewardChoiceMessage): void
+ drawSpecificComponentRequestMessage(DrawSpecificComponentRequestMess
+ endGameMessage(EndGameMessage): void
+ drawCardRequestMessage(DrawCardRequestMessage): void
+ logResponseMessage(LogResponseMessage): void
+ stockoTubaMessage(StockoTubaMessage): void
+ planetLandRequestMessage(PlanetLandRequestMessage): void
+ resumeGameResumeGameMessage(): void
+ emailDeckResponseMessage(SmallDeckResponseMessage): void
+ timeExpiredMessage(TimeExpiredMessage): void
+ timerStartedMessage(TimerStartedMessage): void
+ drawComponentRequestMessage(DrawComponentRequestMessage): void
+ availableSmallDeckMessage(AvailableSmallDeckMessage): void
+ cardActivationRequestMessage(CardActivationRequestMessage): void
+ cardDrawnMessage(CardDrawnMessage): void
+ premadeShipComponentRequest(PremadeShipComponentRequest): void
+ genericMessage(GenericMessage): void
+ waitingPartnersMessage(WaitingPartnersMessage): void
+ freePlanetsResponseMessage(FreePlanetsResponseMessage): void
+ gameStateMessage(GameStateMessage): void
+ dynamicMotorPowerMessage(DynamicMotorPowerMessage): void
+ freeSmallDeckRequestMessage(AvailableSmallDeckRequestMessage): void
+ premadeShipComponentResponse(PremadeShipComponentResponse): void
+ resilienceRequestMessage(ResilienceRequestMessage): void

RMIServerToClient

- client: RemoteClientRMI

+ visit(DrawComponentResponseMessage): void
+ visit(PlayersPositionsMessage): void
+ visit(EndGameMessage): void
+ visit(PremadeShipComponentRequest): void
+ visit(ResilienceRequestMessage): void
+ visit(PlayersNumRequestMessage): void
+ visit(RevealedComponentMessage): void
+ visit(DrawSpecificComponentRequestMessage): void
+ visit(ResumeGameMessage): void
+ visit(ClaimRewardChoiceMessage): void
+ visit(CardActivationRequestMessage): void
+ visit(AvailableSmallDeckRequestMessage): void
+ visit(TimeExpiredMessage): void
+ visit(LogRequestMessage): void
+ visit(CardDrawnMessage): void
+ visit(FreePlanetsResponseMessage): void
+ visit(StockoTubaMessage): void
+ visit(DrawCardRequestMessage): void
+ visit(WinnerMessage): void
+ visit(PickRevealedComponentRequestMessage): void
+ visit(ProjectileTrajectoryMessage): void
+ visit(DiscardStockMessage): void
+ visit(ResilienceResponseMessage): void
+ visit(CreditsEarnedMessage): void
+ visit(BeginMessage): void
+ visit(PremadeShipComponentResponse): void
+ visit(ReturnSmallDeckMessage): void
+ visit(StartTimerRequestMessage): void
+ visit(PositionsAndTurnsMessage): void
+ visit(TurnMessage): void
+ visit(PlaceShipRequestMessage): void
+ visit(UpdateClientShipMessage): void
+ visit(DynamicMotorPowerMessage): void
+ visit(LogResponseMessage): void
+ visit(EmailDeckRequestMessage): void
+ visit(GenericMessage): void
+ visit(DrawComponentRequestMessage): void
+ visit(PlanetLandRequestMessage): void
+ visit(TimerStartedMessage): void
+ visit(UpdateServerShipMessage): void
+ visit(PlayersNumResponseMessage): void
+ visit(ReturnComponentRequestMessage): void
+ visit(SmallDeckResponseMessage): void
+ visit(DynamicFinePowerMessage): void
+ visit(WaitingPartnersMessage): void
+ visit(AvailableSmallDeckMessage): void
+ visit(GameStateMessage): void

RMIMessageSender

- client: RemoteClientRMI

+ visit(UpdateServerShipMessage): void
+ visit(PositionsAndTurnsMessage): void
+ visit(DynamicMotorPowerMessage): void
+ visit(CardDrawnMessage): void
+ visit(GenericMessage): void
+ visit(FreePlanetsResponseMessage): void
+ visit(ReturnSmallDeckMessage): void
+ visit(TimerStartedMessage): void
+ visit(SmallDeckRequestMessage): void
+ visit(StockoTubaMessage): void
+ visit(DiscardStockMessage): void
+ visit(LogRequestMessage): void
+ visit(ProjectileTrajectoryMessage): void
+ visit(PlayersNumResponseMessage): void
+ visit(PlanetLandRequestMessage): void
+ visit(WinnerMessage): void
+ visit(UpdateClientShipMessage): void
+ visit(ResilienceResponseMessage): void
+ visit(AvailableSmallDeckMessage): void
+ visit(AvailableSmallDeckRequestMessage): void
+ visit(PremadeShipComponentRequest): void
+ visit(LogResponseMessage): void
+ visit(PlaceShipRequestMessage): void
+ visit(BeginMessage): void
+ visit(TimerExpiredMessage): void
+ visit(DynamicFinePowerMessage): void
+ visit(PremadeShipComponentResponse): void
+ visit(CardActivationRequestMessage): void
+ visit(GameStateMessage): void
+ visit(EndGameMessage): void
+ visit(DrawComponentRequestMessage): void
+ visit(RevealedComponentMessage): void
+ visit(TurnMessage): void
+ visit(DrawCardRequestMessage): void
+ visit(PlayersNumRequestMessage): void
+ visit(WaitingPartnersMessage): void
+ visit(ReturnComponentRequestMessage): void
+ visit(ResilienceRequestMessage): void
+ visit(ResumeGameMessage): void
+ visit(PlayersPositionsMessage): void
+ visit(SmallDeckResponseMessage): void
+ visit(StartTimerRequestMessage): void
+ visit(ClaimRewardChoiceMessage): void
+ visit(PickRevealedComponentRequestMessage): void
+ visit(CreditsEarnedMessage): void
+ visit(DrawSpecificComponentRequestMessage): void

ServerImpl

- server: Server
- port: int

+ planetLandRequestMessage(PlanetLandRequestMessage): void
+ dynamicFinePowerMessage(DynamicFinePowerMessage): void
+ revealedComponentMessage(RevealedComponentMessage): void
+ discardStockMessage(DiscardStockMessage): void
+ drawComponentRequestMessage(DrawComponentRequestMessage): void
+ waitingPartnersMessage(WaitingPartnersMessage): void
+ emailDeckRequestMessage(SmallDeckRequestMessage): void
+ logResponseMessage(LogResponseMessage): void
+ premadeShipComponentRequest(PremadeShipComponentRequest): void
+ getServer(): Server
+ startTimerRequestMessage(StartTimerRequestMessage): void
+ addClient(RemoteClientRMI, String): void
+ creditsEarnedMessage(CreditsEarnedMessage): void
+ timerStartedMessage(TimerStartedMessage): void
+ updateServerShipMessage(UpdateServerShipMessage): void
+ cardActivationRequestMessage(CardActivationRequestMessage): void
+ drawComponentResponseMessage(DrawComponentResponseMessage): void
+ freeSmallDeckRequestMessage(AvailableSmallDeckRequestMessage): void
+ availableSmallDeckMessage(AvailableSmallDeckMessage): void
+ playersPositionMessage(PlayersPositionsMessage): void
+ drawSpecificComponentRequestMessage(DrawSpecificComponentRequestMess
+ beginMessage(BeginMessage): void
+ turnMessage(TurnMessage): void
+ drawCardRequestMessage(DrawCardRequestMessage): void
+ resumeGameResumeGameMessage(): void
+ returnComponentRequestMessage(ReturnComponentRequestMessage): void
+ endGameMessage(EndGameMessage): void
+ gameStateMessage(GameStateMessage): void
+ emailDeckResponseMessage(SmallDeckResponseMessage): void
+ playersNumRequestMessage(PlayersNumRequestMessage): void
+ timerExpiredMessage(TimeExpiredMessage): void
+ claimRewardChoiceMessage(ClaimRewardChoiceMessage): void
+ start(): void
+ cardDrawnMessage(CardDrawnMessage): void
+ genericMessage(GenericMessage): void
+ resilienceResponseMessage(ResilienceResponseMessage): void
+ positionsAndTurnsMessage(PositionsAndTurnsMessage): void
+ freePlanetsResponseMessage(FreePlanetsResponseMessage): void
+ projectileTrajectoryMessage(ProjectileTrajectoryMessage): void
+ updateClientShipMessage(UpdateClientShipMessage): void
+ stockoTubaMessage(StockoTubaMessage): void
+ premadeShipComponentResponse(PremadeShipComponentResponse): void
+ dynamicMotorPowerMessage(DynamicMotorPowerMessage): void
+ resilienceRequestMessage(ResilienceRequestMessage): void
+ placeShipRequestMessage(PlaceShipRequestMessage): void
+ playersNumResponseMessage(PlayersNumResponseMessage): void
+ logRequestMessage(LogRequestMessage): void
+ pickRevealedComponentRequestMessage(PickRevealedComponentRequestMes
+ returnSmallDeckMessage(ReturnSmallDeckMessage): void
+ winnerMessage(WinnerMessage): void

ClientRMI

- IP: String
- serverRMI: ServerRMI
- clientinterface: ClientInterface
- port: int

+ claimRewardChoiceMessage(ClaimRewardChoiceMessage): void
+ creditsEarnedMessage(CreditsEarnedMessage): void
+ drawComponentRequestMessage(DrawComponentRequestMessage): void
+ sendMessage(Message): void
+ timerStartedMessage(TimerStartedMessage): void
+ availableSmallDeckMessage(AvailableSmallDeckMessage): void
+ premadeShipComponentResponse(PremadeShipComponentResponse): void
+ updateClientShipMessage(UpdateClientShipMessage): void
+ logRequestMessage(LogRequestMessage): void
+ updateServerShipMessage(UpdateServerShipMessage): void
+ emailDeckRequestMessage(SmallDeckRequestMessage): void
+ genericMessage(GenericMessage): void
+ pickRevealedComponentRequestMessage(PickRevealedComponentRequestMes
+ waitingPartnersMessage(WaitingPartnersMessage): void
+ freeSmallDeckRequestMessage(AvailableSmallDeckRequestMessage): void
+ playersNumResponseMessage(PlayersNumResponseMessage): void
+ playersPositionMessage(PlayersPositionsMessage): void
+ cardActivationRequestMessage(CardActivationRequestMessage): void
+ dynamicFinePowerMessage(DynamicFinePowerMessage): void
+ positionsAndTurnsMessage(PositionsAndTurnsMessage): void
+ startTimerRequestMessage(StartTimerRequestMessage): void
+ turnMessage(TurnMessage): void
+ discardStockMessage(DiscardStockMessage): void
+ timeExpiredMessage(TimeExpiredMessage): void
+ smallDeckResponseMessage(SmallDeckResponseMessage): void
+ disconnect(): void
+ getRemoteReference(): ServerRMI
+ playersNumRequestMessage(PlayersNumRequestMessage): void
+ planetLandRequestMessage(PlanetLandRequestMessage): void
+ resumeGameResumeGameMessage(): void
+ gameStateMessage(GameStateMessage): void
+ returnSmallDeckMessage(ReturnSmallDeckMessage): void
+ beginMessage(BeginMessage): void
+ winnerMessage(WinnerMessage): void
+ endGameMessage(EndGameMessage): void
+ revealedComponentMessage(RevealedComponentMessage): void
+ drawSpecificComponentRequestMessage(DrawSpecificComponentRequestMess
+ projectileTrajectoryMessage(ProjectileTrajectoryMessage): void
+ premadeShipComponentRequest(PremadeShipComponentRequest): void
+ resilienceResponseMessage(ResilienceResponseMessage): void
+ drawCardRequestMessage(DrawCardRequestMessage): void
+ drawComponentResponseMessage(DrawComponentResponseMessage): void
+ logRequestMessage(LogRequestMessage): void
+ cardDrawnMessage(CardDrawnMessage): void
+ returnComponentRequestMessage(ReturnComponentRequestMessage): void
+ dynamicMotorPowerMessage(DynamicMotorPowerMessage): void
+ freePlanetsResponseMessage(FreePlanetsResponseMessage): void
+ placeShipRequestMessage(PlaceShipRequestMessage): void
+ stockoTubaMessage(StockoTubaMessage): void
+ resilienceRequestMessage(ResilienceRequestMessage): void

ServerAdapterRMI

+ handleMessage(Message): void

ClientHandlerRmi

- client: RemoteClientRMI
- connected: boolean
- server: ServerImpl

+ isConnected(): boolean
+ disconnect(): void
+ send(Message): void