

```
networking
              GameStartView
   hostServer : ServerSocket
   hostSocket : Socket
   mainServer: Server
   mainClient : Client
   scene : Scene
  theView1 : View
  theView2 : View
  p1 : Player
 p2 : Player
   theModel: Model
  theCtrl : Controller
  initGame : Thread
   + start( primaryStage : Stage )
```

```
Server
hostServer : ServerSocket
hostSocket: Socket
input : ObjectInputStream
output : ObjectOutputStream
 getHostServer() : ServerSocket
 initServer()
 initController() : Controller
 getHostSocket(): Socket
```

```
Client
clientSocket : Socket
input : ObjectInputStream
- output : ObjectOutputStream
initController() : Controller
+ Client( serverIp : String )
sendToServer()
getClientSocket() : Socket
+ getInputStream() : ObjectInputStream
 getOutputStream(): ObjectOutputStream
```

```
player_AI
      - myShips : ArrayList<Ship>
      - initIndex : int
      - SHIP_CONFIG : int[]
      - TYPES : ShipType[]
      - TOTAL SHIPS : int
      - myBoard : Board
      - targetBoard : Board
      - shipRemain : int
      - mvView : View
      - shipsAdded : boolean
      - carrierAdded : boolean
      - battleshipAdded : boolean
      - cruiserAdded : boolean

    destroverAdded : boolean

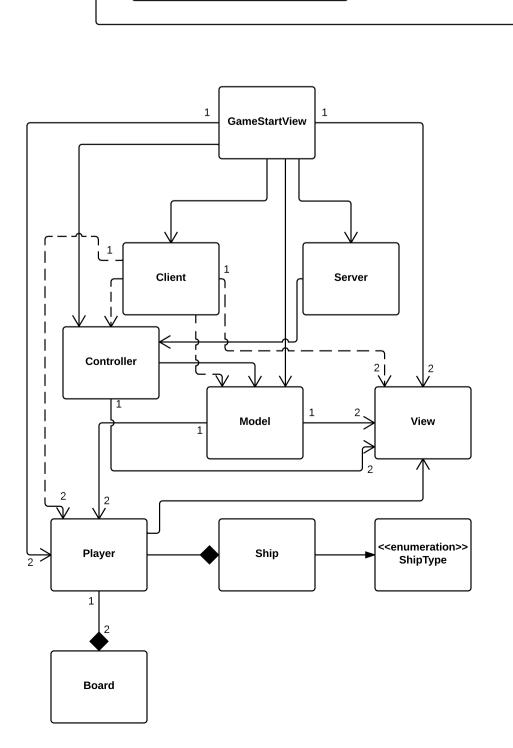
      - submarineAdded : boolean
     + Player( view : View )
      - initShips()
      + initShipLocations( loc : int[] )
      + attack( loc : int[], enemy : Player ) : boolean
      + defend( loc : int[] )
      - updateShips( loc : int[] ) : boolean
      - updateSelf( loc : int[] )
      + didILose(): boolean
     + shipsAddedTrue()
      + addedCarrier()
      + addedBattleship()
     + addedSubmarine()
      + addedDestroyer()
      + addedCruiser()
      + getMyView() : View
      + getMyBoard() : Board
      + getShipsAdded() : boolean
      + getShipRemain() : int
```

```
shipType : ShipType
orientation : Orientation
- isSunk : boolean
                                                                                                <<enumeration>>
- shipHits : int[]
                                                                                                    ShipType
- shipLoc : ArrayList<int[]>
numHits : int
                                                                                     ⊦ size : int
+ Ship( shipLoc : ArrayList<int[]>, shipType : ShipType )
                                                                                     + color : String
+ Ship( type : ShipType )
+ updateShip( hitTarget : int[] )
+ checkIfSunk( ) : boolean
                                                                                     + ShipType( size : int, color : String )
+ getShipType(): ShipType
                                                                                     + getSize( ) : int
+ getShipLoc( ) : ArrayList<int[]>
                                                                                     getColor(): String
getOrientation(): Orientation
+ setOrientation( orientation : Orientation )
+ rotateCW( )
+ rotateCCW()
+ setLocation( startLoc : int[] )
+ isIsSunk(): boolean
+ getShipHits( ) : int[]
```

```
Controller
 theModel : Model
 theView1 : View
                                                                                              Model
 theView2: View
                                                    - carrier : SimpleBooleanProperty
                                                    - battleship : SimpleBooleanProperty
 + Controller( theModel : Model )
                                                    - cruiser : SimpleBooleanProperty
 + handle( event : ActionEvent )
                                                    - submarine : SimpleBooleanProperty
 + getTheModel() : Model
                                                    - destroyer : SimpleBooleanProperty
                                                    - addedCarrier : SimpleBooleanProperty
                                                    - addedBattleship : SimpleBooleanProperty

    addedCruiser : SimpleBooleanProperty

                                                    - addedSubmarine : SimpleBooleanProperty
                                                    - addedDestroyer : SimpleBooleanProperty
+ board : Rectangle[][]
 enemyBoard : Rectangle[[[]
                                                    p1View : View
                                                    p2View : View
 + myBoard : GridPane
                                                    - p1 : Player
 opponentBoard : GridPane
                                                    - p2 : Player
 root : BorderPane
                                                    - p1Target : int[]
 grids : GridPane
 p1InitDone : boolean
                                                    - p2Target : int[]
 - p2InitDone : boolean
                                                    - p1InitDone : boolean
                                                    - p2InitDone : boolean
 rightPane : FlowPane
 - carrierBtn : ToggleButton
                                                    + Model( p1View : View, p2View : View, p1 : Player, p2 : Player )
 - battleshipBtn : ToggleButton
 - cruiserBtn : ToggleButton
                                                    + getTargetLocation( player : int ) : int[]
 - submarineBtn : ToggleButton
                                                    + timerStart( p : Player )
 - destroyerBtn : ToggleButton
                                                    + timerRestart( p : Player )
shipGroup : ToggleGroup
                                                    + enableAttack( p : Player )
                                                    + disableAttack( p : Player )
 - bottomPane : FlowPane
 rotateCWBtn : Button
                                                    + disableShipSelection( p : Player )
 rotateCCWBtn : Button
                                                    + finishInitShip( p : Player )
 - shipHorizontal : RadioButton
                                                    + displayVictoryMsg( p : Player )
 shipVertical : RadioButton
                                                    + displayDefeatMsg( p : Player )
orientationGroup : ToggleGroup
                                                    + attackShip( player : int )
 - attackBtn : Button
                                                    + startGame()
 - shipSelectionDone : Button
                                                    + showGUI( p : Player )
 timer : AnimationTimer
                                                    + disableGUIControl( p : Player )
- i : Integer
                                                    + buildShipMy( ship : Ship, player : Player )
 timerText : Text
                                                    + updateShip( view : View, player : int )
 timeline : TImeline
                                                    + readyToStart() : boolean
 onFinished : EventHandler
                                                    + getCarrier() : SimpleBooleanProperty
 keyFrame : KeyFrame
                                                    + getBattleship() : SimpleBooleanProperty
 duration : Duration
                                                    + getCruiser() : SimpleBooleanProperty
                                                    + getSubmarine() : SimpleBooleanProperty
+ View()
                                                    + getDestroyer() : SimpleBooleanProperty
+ createShips()
                                                    + getP1() : Player
 + createEnemyBoard()
                                                    + getP2() : Player
 + createBoard()
                                                    + getP1View() : View
 + showWinOrLoss( winOrLoss: String)
                                                    + getP2View() : View
 + showShipSelectionError()
                                                    + isP1InitDone() : boolean
 + readyToStart() : boolean
                                                    + isP2InitDone() : boolean
 + getRoot() : BorderPane
                                                    + getAddedCarrier() : SimpleBooleanProperty
+ getCarrierBtn() : ToggleButton
                                                    + getAddedBattleship() : SimpleBooleanProperty
  getBattleshipBtn() : ToggleButton
                                                    + getAddedCruiser() : SimpleBooleanProperty
 + getCruiserBtn() : ToggleButton
                                                    + getAddedSubmarine() : SimpleBooleanProperty
 + getSubmarineBtn() : ToggleButton
                                                     getAddedDestroyer() : SimpleBooleanProperty
 + getDestroyerBtn(): ToggleButton
                                                    + setP1InitDone( p1InitDone : boolean )
 + getBoard() : Rectangle∏∏
                                                    + setP2InitDone ( p2InitDone : boolean )
 + getEnemyBoard() : Rectangle[][]
                                                    + setAddedCarrier( addedCarrier : SimpleBooleanProperty )
 getMyBoard() : GridPane
                                                    + setAddedBattleship( addedBattleship : SimpleBooleanProperty )
 + getAttackBtn() : Button
                                                     - setAddedCruiser( addedCruiser : SimpleBooleanProperty )
 + getShipSelectionDone() : Button
                                                    + setAddedSubmarine( addedSubmarine : SimpleBooleanProperty )
 + getShipHorizontal() : RadioButton
                                                    + setAddedDestroyer( addedDestroyer : SimpleBooleanProperty )
 + getShipVertical() : RadioButton
 + getTimer() : AnimationTimer
 + getTimeline() : Timeline
 + isP1InitDone() : boolean
 + setP1InitDone( p1InieDone : boolean )
 + isP2InitDone() : boolean
 + setP2InitDone( p2InitDone : boolean )
```



board Board - board : int[][] - size : int + Board(size : int) + updateBoard(loc : int[]) + toString() : String + getBoard() : int[][]

Controller

+ checkOverlap(shipSize : int, col : int, row : int, horizontal : int) : boolean

GUI_AI

- theModel : Model

- theView1 : View

- theView2 : View

- Board2 : int[][]

- location : int[]

+ startGame()

+ clearBoard()

+ placeShips()

+ checkWin()

+ buildShip()

+ setEnemyBoard()

+ click()

+ Controller(theModel: Model)

+ handle(event : ActionEvent)

+ board : Rectangle[][]

+ myBoard : GridPane

- rightPane : FlowPane

- carrierBtn : ToggleButton

- cruiserBtn : ToggleButton

shipGroup: ToggleGroup

- bottomPane : FlowPane

- rotateCCWBtn : Button

- rotateCWBtn : Button

- attackBtn : Button

- startGame : Button

: Integer

+ View()

+ createShips()

+ createBoard()

+ clearBoards()

+ createEnemyBoard()

+ showShipSelectionError()

+ getCarrierBtn() : ToggleButton

+ getBattleshipBtn() : ToggleButton + getCruiserBtn() : ToggleButton + getSubmarineBtn() : ToggleButton + getDestroyerBtn() : ToggleButton + getBoard() : Rectangle[][]

+ getEnemyBoard() : Rectangle[][]

getShipSelectionDone(): Button

+ getShipVertical() : RadioButton

+ getTimer() : AnimationTimer

+ getTimeline() : Timeline

getStartGame(): Button

+ getShipHorizontal() : RadioButton

+ getMyBoard() : GridPane

+ getAttackBtn() : Button

+ getRoot(): BorderPane

+ showWinOrLoss (winOrLoss : String)

timerText : Text

- timeline : TImeline

- duration : Duration

- clearBoards : Button

timer : AnimationTimer

- battleshipBtn : ToggleButton

- submarineBtn : ToggleButton

- destroyerBtn : ToggleButton

- shipHorizontal : RadioButton

orientationGroup : ToggleGroup

- shipVertical : RadioButton

shipSelectionDone: Button

- onFinished : EventHandler

keyFrame : KeyFrame

- root : BorderPane

grids : GridPane

- enemyBoard : Rectangle[][]

- opponentBoard : GridPane

Model - carrier : SimpleBooleanProperty battleship : SimpleBooleanProperty - cruiser : SimpleBooleanProperty - submarine : SimpleBooleanProperty destroyer : SimpleBooleanProperty - addedCarrier : SimpleBooleanProperty - addedBattleship : SimpleBooleanProperty - addedCruiser : SimpleBooleanProperty - addedSubmarine : SimpleBooleanProperty addedDestroyer : SimpleBooleanProperty - p1View : View - p2View : View - p1 : Player - p2 : Player p1Target : int[] - p2Target : int[] + Model(p1View : View, p2View : View, p1 : Player, p2 : Player) + getTargetLocation(player : int) : int[] + timerStart(p : Player)

+ timerRestart(p : Player) + enableAttack(p : Player) + disableAttack(p : Player) + disableShipSelection(p : Player) + finishInitShip(p : Player) + displayVictoryMsg(p : Player) + displayDefeatMsg(p : Player) + displayTargetStatus() + attackShip(player : int) + startGame() + showGUI(p : Player) - disableGUIControl(p : Player) + buildShipMy(ship : Ship, player : Player) + updateShip(view : View, player : int) + getCarrier() : SimpleBooleanProperty + getBattleship() : SimpleBooleanProperty - getCruiser() : SimpleBooleanProperty + getSubmarine() : SimpleBooleanProperty getDestroyer() : SimpleBooleanProperty getP1() : Player + getP2() : Player + getP1View() : View + getAddedCarrier() : SimpleBooleanProperty getAddedBattleship() : SimpleBooleanProperty + getAddedCruiser() : SimpleBooleanProperty + getAddedSubmarine() : SimpleBooleanProperty getAddedDestroyer(): SimpleBooleanProperty + setAddedCarrier(addedCarrier : SimpleBooleanProperty) + setAddedBattleship(addedBattleship : SimpleBooleanProperty) + setAddedCruiser(addedCruiser : SimpleBooleanProperty) + setAddedSubmarine(addedSubmarine : SimpleBooleanProperty) - setAddedDestroyer(addedDestroyer : SimpleBooleanProperty)