Strategic Web Based Management Game

Group 12
Per Eriksson
Per Strand
Simon Ragnar
Ingemar Markström
Max Walter

5.5 Detailed Description

Game Round

createNewMap

Parameters: int numberOfPlayers
Return value: none
Description: Creates the map based on how many players that will play.
Database: stores the created maps size
Pre-condition: none
Validity checks: numberOfPlayers needs to be positive
Post-condition: the map size is calculated and added to the database
Calls: none
Called by: the system

createMapObjects

Parameters: int numberOfPlayers
Return value: none
Description: Creates the ships, resource squares and wormhole and stores them in the database.
Database: stores the created map objects
Pre-condition: the map is created
Validity checks: numberOfPlayers needs to be positive
Post-condition: all types of map objects is created and added to the database
Calls: createMapObject()
Called by: the system

placeMapObjects

Parameters: none
Return value: none
Description: Places all map objects on the map.
Database: stores map-coordinates for all map objects
Pre-condition: Map objects created.
Validity checks: the map objects coordinates should not be initialized
Post-condition: All map objects are given coordinates on the map, and are therefore placed on the map.
Calls: none
Called by: the system

endGameRound

Parameters: int playerID
Return value: none
Description: the function notifies the system that there is a winner to the current game round. The function then gives awards to the players based on their success
Database: stores all relevant results of the game round
Pre-condition: the player with the corresponding playerID is located on the wormhole
Validity checks: none
Post-condition: The game round ends. All players get notified.
Calls: the system
Called by: the system

addNewPlayer

Parameters: int playerID
Return value: none
Description: the function gives the player a default ship and places it on the map
Database: stores a new ship
Pre-condition: none
Validity checks: none
Post-condition: the player isn't already active in the current game round
Calls: none
Called by: the web-page

Highscore

createEscapePointList

Parameters: String[] playerID
Return value: none
Description: Creates a list ordered after the players Escape Points
Database: stores the escape points list in the database
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: calculateEscapePoints()
Called by: the system

createCloseToCenterList

Parameters: String[] playerID
Return value: none
Description: Creates a list ordered after how close the players are to the wormhole
Database: stores the close to center list in the database
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: calculateCloseToCenter()
Called by: the system

Alliance

getName

Parameters: int playerID
Return value: String name
Description: returns the name of the alliance.
Database: returns the name of the alliance that the given player is associated with
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: none
Called by: the web-page

getPlayers

Parameters: String allianceName

Return value: String[] players
Description: returns all players in the alliance.
Database: returns all players within an alliance
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: none
Called by: the web-page

addPlayer

Parameters: int playerID, String allianceName
Return value: none
Description: Adds a player to the alliance.
Database: adds a player to an alliance
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: none
Called by: the web-page

removePlayer

Parameters: int playerID
Return value: none
Description: Removes player from alliance.
Database: sets the player's associated alliance name to nothing
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: none
Called by: the web-page

Map

drawMapObjects

Parameters: int x, int y
Return value: none
Description: draws the map on the screen based on the player's set of coordinates
Database: extracts the map objects that are located within the visual range of the player's ship
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: getMapObjects()
Called by: the web-page

checkStackability

Parameters: int x, int y
Return value: boolean isStackable
Description: the funtion validates if a square is stackable
Database: returns the boolean isStackable for all objects located on the given set of coordinates

Pre-condition: none
Validity checks: none
Post-condition: none
Calls: none
Called by: move()

getMapObjects

Parameters: int x, int y, int xlength, int ylength
Return value: String[] objects, float[] coordinates
Description: return all map objects in an area.
Database: returns all map objects in an area.
Pre-condition: none
Validity checks: none
Post-condition: none
Calls: none
Called by: drawMapObjects()

addIncomingMissiles

Parameters: int playerIDattacker, int playerIDvictim, int amountOfMissiles, int impactTime
Return value: none
Description: adds the missiles to the list containing all incoming missiles. The function also notifies the attacker and the victim about the attack.
Database: none
Pre-condition: none
Validity checks: playerIDvictim must be a valid playerID and the impact time cannot be negative
Post-condition: none
Calls: the system, the web-page
Called by: missileAttack()

addIncomingShell

Parameters: int playerIDattacker, int playerIDvictim, int amountOfShells, int impactTime
Return value: none
Description: adds the shells to the list containing all incoming shells. The function also notifies the attacker and the victim about the attack.
Database: none
Pre-condition: none
Validity checks: playerIDvictim must be a valid playerID and the impact time cannot be negative
Post-condition: none
Calls: the system, the web-page
Called by: cannonAttack()

Research

toogleEngineResearch

Parameters: int numberOfResearch
Return value: none
Description: Togles the specified research object on or off
Database: Togles the specified research object on or off
Pre-condition: none
Validity checks: numberOfResearch is a valid identifier of the specified research object
Post-condition: none
Calls: none

Called by: web-page

toogleCannonResearch

Parameters: int numberOfResearch
Return value: none
Description: Togles the specified research object on or off
Database: Togles the specified research object on or off
Pre-condition: none
Validity checks: numberOfResearch is a valid identifier of the specified research object
Post-condition: none
Calls: none
Called by: web-page

toogleModuleResearch

Parameters: int numberOfResearch
Return value: none
Description: Togles the specified research object on or off
Database: Togles the specified research object on or off
Pre-condition: none
Validity checks: numberOfResearch is a valid identifier of the specified research object
Post-condition: none
Calls: none
Called by: web-page

Ship

getMainResource

Parameters: none
Return value: int amountOfMainResource
Description: the function returns the amount of main resources that are stored in the ship.
Database: returns the amount of main resources that the ship has stored.
Pre-condition: none
Validity checks: the amount of main resources cannot be negative
Post-condition: none
Calls: none
Called by: the web-page, buildMissiles(), buildShells(), buildModule(), gatherResource(), moveShip(), upgradeModule(), toggle[module]Research(), teleportShip()

getSecondaryResource

Parameters: none
Return value: int amountOfSecondaryResource
Description: the function returns the amount of secondary resources that are stored in the storage module of the ship
Database: returns the amount of secondary resources that the ship has stored.
Pre-condition: none
Validity checks: the amount of secondary resources cannot be negative
Post-condition: none
Calls: none
Called by: the web-page, buildMissiles(), buildShells(), buildModule() (not all modules requires the secondary resource), toggle[module]Research(), upgradeModule(), teleportShip()

getConditionStatus

Parameters: none
Return value: int conditionStatus
Description: returns the condition status of the ship
Database: returns the condition status of the ship
Pre-condition: none
Validity checks: the condition status cannot be negative
Post-condition: none
Calls: none
Called by: the web-page, interceptionWithMissile(), interceptionWithShell()

changeMainResource

Parameters: int newAmountOfMainResource
Return value: none
Description: the function updates the amount of main resources that the ship has stored in its storage module. The function must make sure that the new amount of main resources doesn't exceed the amount of resources that can be stored in the main resource storage in the ship.
Database: the function sets a new amount of main resources for the ship
Pre-condition: none
Validity checks: the new amount of main resources doesn't exceed the maximum amount of main resources that can be stored.
Post-condition: none
Calls: none
Called by: buildMissile(), buildShell(), buildAModule(), gatherResource(), moveShip(), produceMainResource(), teleportShip()

changeSecondaryResource

Parameters: int newAmountOfSecondaryResource
Return value: none
Description: the function updates the amount of secondary resources that the ship has stored in its storage module. The function must make sure that the new amount of secondary resources doesn't exceed the amount of resources that can be stored in the secondary resource storage in the ship.
Database: sets a new amount of secondary resources for the ship
Pre-condition: none
Validity checks: the new amount of secondary resources doesn't exceed the maximum amount of secondary resources that can be stored.
Post-condition: none
Calls: none
Called by: buildMissile(), buildShell(), buildAModule() (doesn't apply to all modules), gatherResource (), teleportShip()

getCoordinates

Parameters: none
Return value: float[] setOfCoordinates
Description: the function returns the map-coordinates for a ship.
Database: returns the coordinates for a ship
Pre-condition: the ship must be located on the map
Validity checks: the coordinates of the ship cannot be outside of the map boundary
Post-condition: none
Calls: none
Called by: the web-page, drawMapObject(), fireMissile(), fireShell(), teleportShip(), moveShip()

setCoordinates

Parameters: float[] setOfCoordinates
Return value: none
Description: the function stores a new set of map-coordinates for a ship. The function must make sure that the new set of coordinates are within the map boundary
Database: overwrites the old set of coordinates for the ship.
Pre-condition: none
Validity checks: the set of coordinates must be within the map boundary
Post-condition: changes the ship's coordinates
Calls: none
Called by: createMapObjects(), moveShip(), teleport()

Resource square

mineResource

Parameters: int extractResources
Return value: nones
Description: the function gets an amount of resources that are to be extracted from the square. The function will then subtract this amount from the resource square
Database: sets a new amount of resources available at the resource square
Pre-condition: none
Validity checks: the amount of resources that are to be extracted cannot exceed the amount of resources that are available
Post-condition: none
Calls: none
Called by: gatherResources()

getAmountOfResources

Parameters: none
Return value: int amountOfResources
Description: the function returns the amount of resources available at the resource square
Database: returns the amount of resources that are available
Pre-condition: none
Validity checks: the amount of resources cannot be negative
Post-condition: none
Calls: none
Called by: the web-page, gatherResources()

Wormhole

notifyWin

Parameters: none
Return value: boolean win
Description: the function runs constantly throughout the game round and checks whether or not a ship is located at the wormhole coordinates
Database: returns the amount of ships that are located on the wormhole coordinates
Pre-condition: none
Validity checks: there cannot be a negative amount of ship's located on the wormhole and neither can there be two ships located on the wormhole simultaneously.
Post-condition: none
Calls: none

Called by: The master game loop

Player

CalculateEscapePoints

Parameters: none
Return value: int amountOfEscapePoints
Description: the function calculates the amount of escape points that the user has based on his/her modules on the player's ship and his/her level of research on each module
Database: returns information regarding the user's modules and research within different fields
Pre-condition: none
Validity checks: the amount of escape points cannot be negative
Post-condition: none
Calls: none
Called by: the web-page, createEscapePointList()

getOverallClosestPosition

Parameters: none
Return value: int overallClosestPosition
Description: the function returns the overall closest distance from the wormhole for the ship
Database: returns the overall closest distance from the wormhole for the ship
Pre-condition: none
Validity checks: the overall closest distance from the wormhole cannot be negative
Post-condition: none
Calls: none
Called by: the web-page

getHighestEscapePointEver

Parameters: none
Return value: int highestEscapePointEver
Description: the function returns the player's highest escape point from all game rounds
Database: returns the player's highest escape point from all game rounds
Pre-condition: none
Validity checks: the escape point cannot be negative
Post-condition: none
Calls: none
Called by: the web-page

getAlliance

Parameters: int playerID
Return value: String alliance
Description: the function returns the player's alliance.
Database: returns the alliance associated with the playerID
Pre-condition: the alliance exist
Validity checks: the alliance must have a name
Post-condition: none
Calls: none
Called by: the web-page

createNewAlliance

Parameters: String nameOfAlliance

Return value: boolean allianceCreated
Description: the function validates if the desired alliance name is already taken. If not, the function creates a new alliance.
Database: returns all alliances with the same name as the desired one. Also stores a new alliance if the name wasn't taken.
Pre-condition: none
Validity checks: the String nameOfAlliance cannot be taken by an existing aliance
Post-condition: none
Calls: none
Called by: the web-page

Text Message

getMessages

Parameters: int playerID
Return value: String[] messages
Description: the funtion returns all received text messages for the player
Database: returns all received messages associated with the playerID
Pre-condition: none
Validity checks: the amount of messages cannot be negative
Post-condition: none
Calls: none
Called by: the web-page

sendMessage

Parameters: int playerIDfrom, int playerIDto, String message
Return value: none
Description: the function sends a text message to playerIDto and adds the message to playerIDfrom's sent messages list
Database: adds the message as a sent message to the playerIDfrom's sent message list and to the playerIDto's received messages list.
Pre-condition: playerIDto must be a valid playerID
Validity checks: none
Post-condition: none
Calls: none
Called by: the web-page

removeMessage

Parameters: int messageNumber
Return value: none
Description: removes the specified message from server
Database: deletes the specified message from database
Pre-condition: the givven messageNumber exist.
Validity checks: none
Post-condition:
Calls: none
Called by: web-page

Module

moveShip

Parameters: array of floats, x and y coordinates
Return value: none
Description: Moves the ship to the designated location
Database: Moves the ship to the designated location
Pre-condition: The ship exist
Validity checks: The coordinates givven are valid
Post-condition: none
Calls: none
Called by: the web-page

cannonAttack

Parameters: int shipID
Return value: none
Description: Makes damage to an other specified ship
Database: Writes damage to the specified ship
Pre-condition: none
Validity checks: valid shipID and valid range
Post-condition: none
Calls: addIncomingShell()
Called by: web-page

missilAttack

Parameters: int shipID
Return value: none
Description: Makes damage to an other specified ship
Database: Writes damage to the specified ship
Pre-condition: none
Validity checks: enough missiles in slots to attack.
Post-condition: none
Calls: addIncomingMissile()
Called by: web-page

buildShells

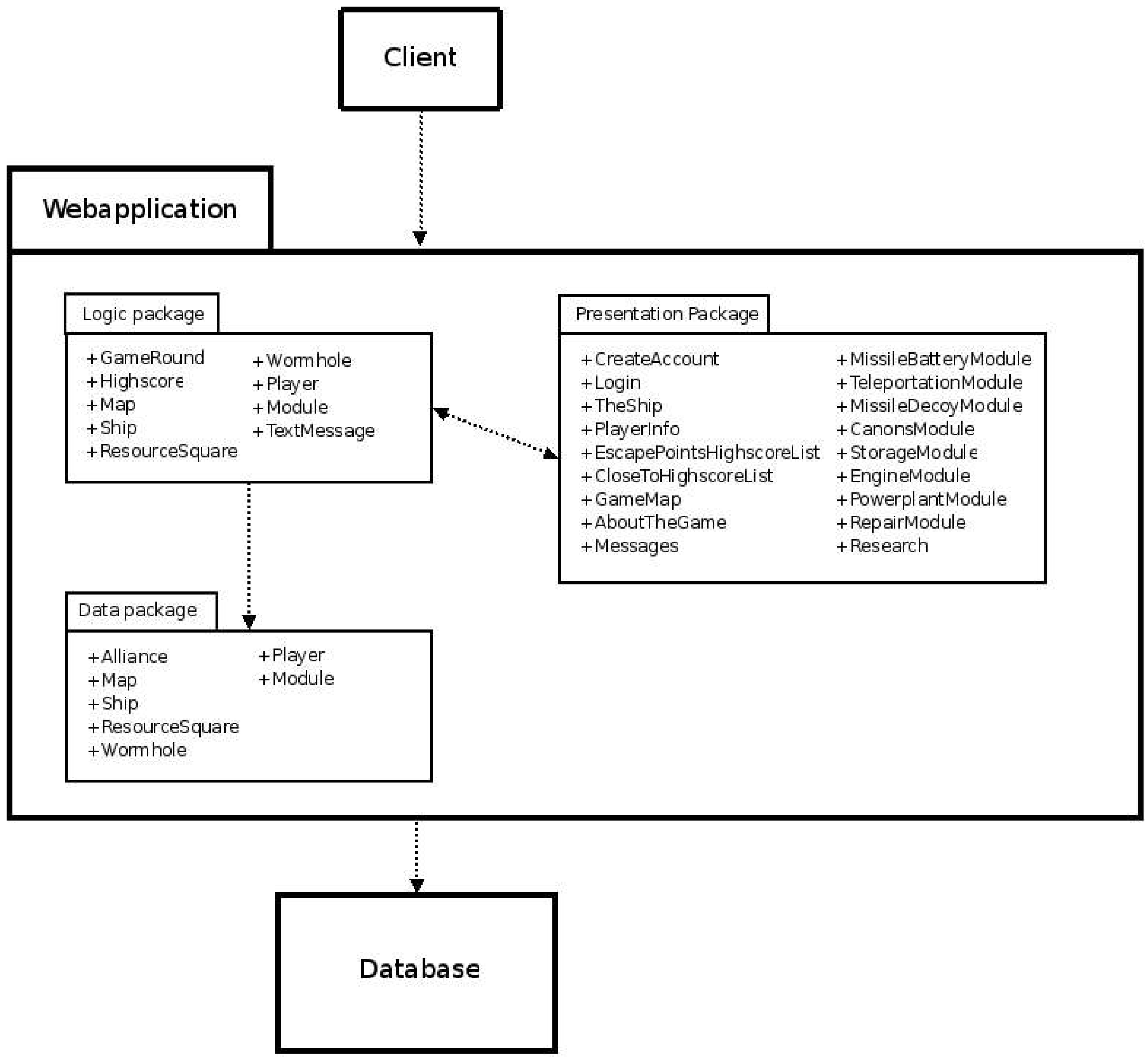
Parameters: int ammount
Return value: none
Description: Increase the ammount of shells.
Database: Writes change to ammount of shells.
Pre-condition: prepayed shell cost, enough storage space, available cannon module
Validity checks: none
Post-condition: none
Calls: none
Called by: web-page

buildMissile

Parameters: int ammount
Return value: none
Description: Increase the ammount of missiles.
Database: Writes change to ammount of shells.

Pre-condition: prepayed missile cost, enough storage space, available missile module
Validity checks: none
Post-condition: none
Calls: none
Called by: web-page

5.6 Package Diagram


Client
Webapplication
Logic package
+GameRound
+Highscore
+Map
+Ship
+ResourceSquare
+Wormhole
+Player
+Module
+TextMessage
Presentation Package
+CreateAccount
+Login
+TheShip
+PlayerInfo
+EscapePointsHighscoreList
+CloseToHighscoreList
+GameMap
+AboutTheGame
+Messages
+MissileBatteryModule
+TeleportationModule
+MissileDecoyModule
+CanonsModule
+StorageModule
+EngineModule
+PowerplantModule
+RepairModule
+Research
Data package
+Alliance
+Map
+Ship
+ResourceSquare
+Wormhole
+Player
+Module
Database