|  |
| --- |
| Application Design Document  Application: Electronic Battle Ship Game |

Name: Manisai Bejjanki

Author: Manisai

18 October 2021

Version: 1.1

Confidential

**Table of Contents**

[1 Project Summary 4](#_Toc44627483)

[2 Project Description 5](#_Toc44627484)

3 DESign patterns USED 9

[4 Challenges encountered 12](#_Toc44627496)

[5 improvements 13](#_Toc44627496)

[6 Test Application 14](#_Toc44627483)

# Project Summary

Battleship is a guessing game for two players. It is played on four grids. Two grids (one for each player) are used to mark each players' fleets of ships (including battleships). The locations of the fleet (these first two grids) are concealed from the other player so that they do not know the locations of the opponent's ships. Players alternate turns by 'firing torpedoes' at other player's ships. The objective of the game is to destroy the opposing player's entire fleet. In our game, 'firing a torpedo' will be allowing the player to take a guess at where on the grid their opponent may have placed a ship.

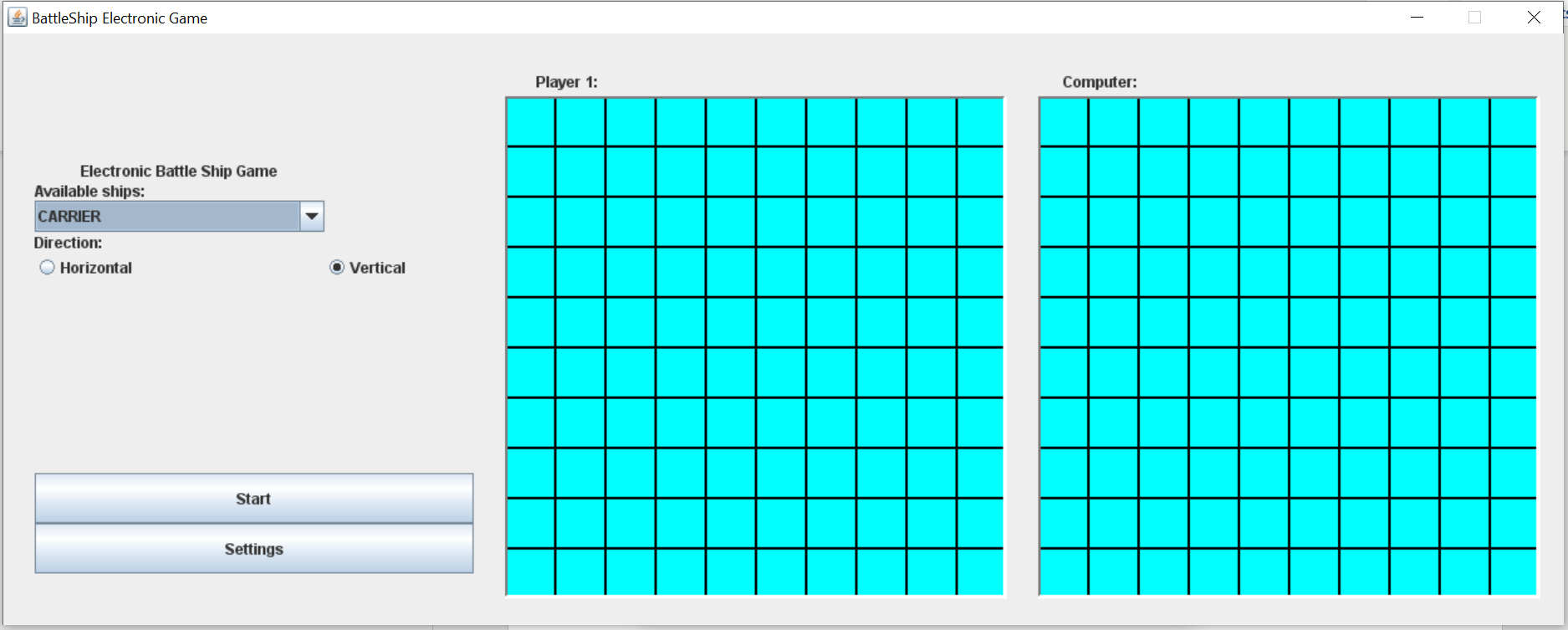
# Project Description

Battleship is a guessing game for two players. It is played on four grids. Two grids (one for each player) are used to mark each players' fleets of ships (including battleships). The locations of the fleet (these first two grids) are concealed from the other player so that they do not know the locations of the opponent's ships. Players alternate turns by 'firing torpedoes' at other player's ships. The objective of the game is to destroy the opposing player's entire fleet. In our game, 'firing a torpedo' will be allowing the player to take a guess at where on the grid their opponent may have placed a ship.

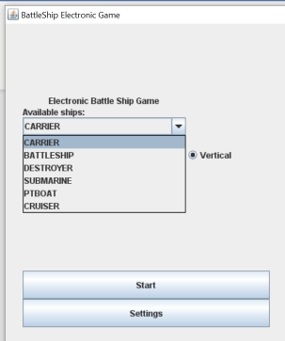
This project is built on the JDK 1.8 version and uses java inbuilt awt module for interface.

**Please find the project flow for the battleship Game.**



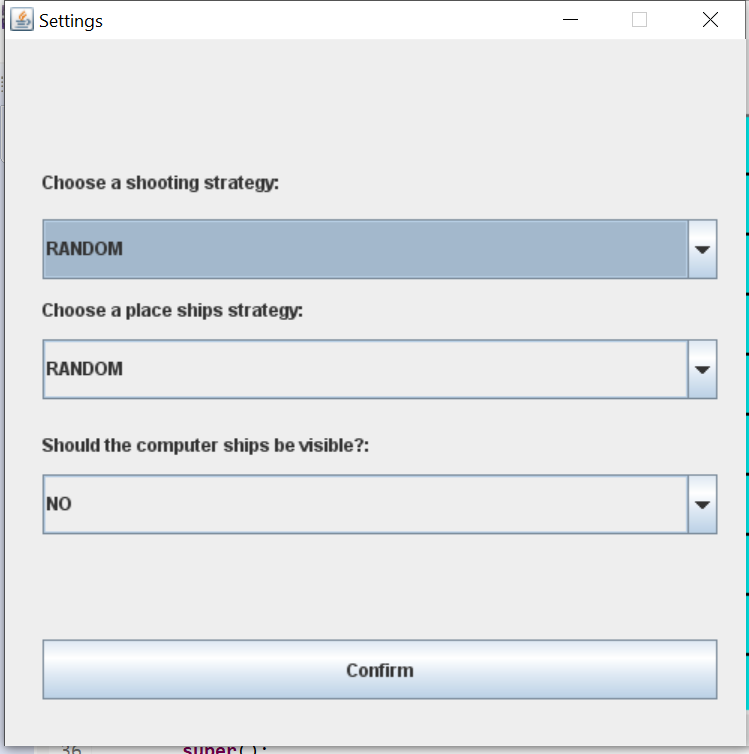


Player has to select the Available ships from the dropdown list

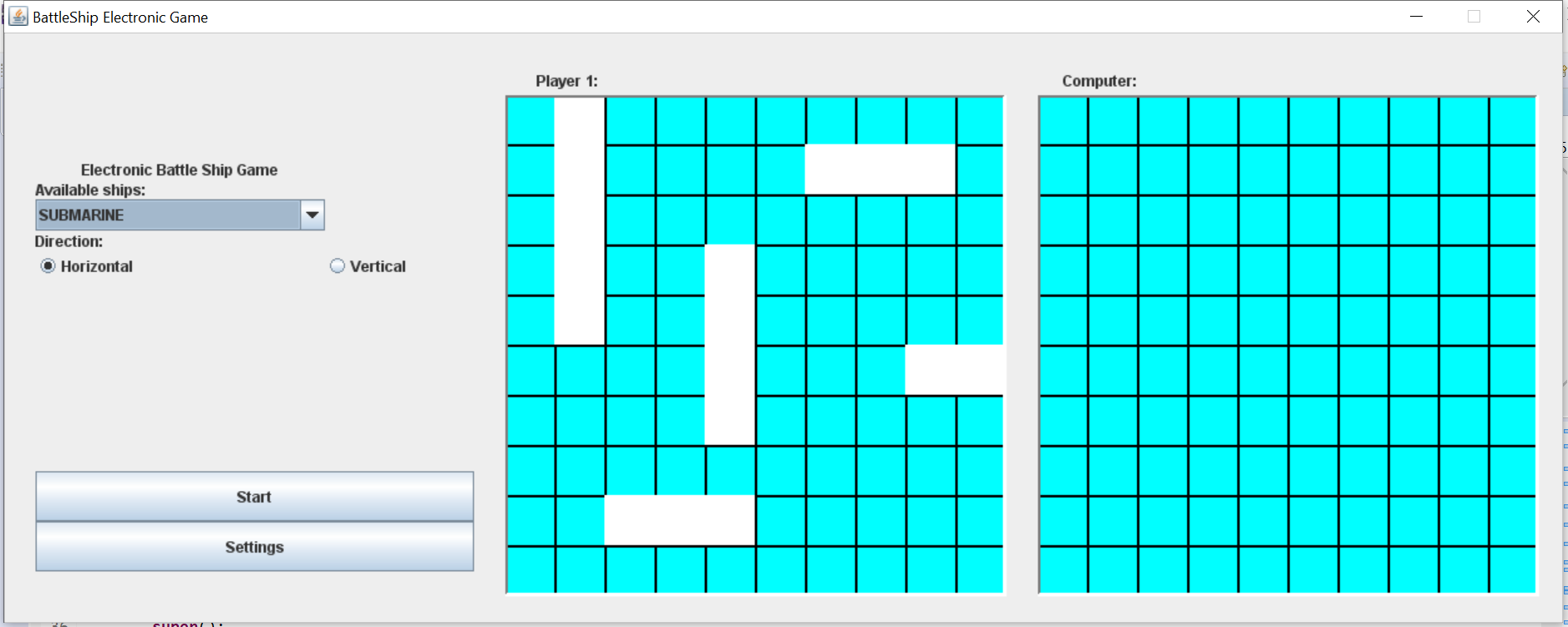


A player can choose the direction of the placement of the ship.

Additionally, a player can choose the advance settings to choose the strategy for placement and hit ship respectively.



A player has to place his ships in his grid like below

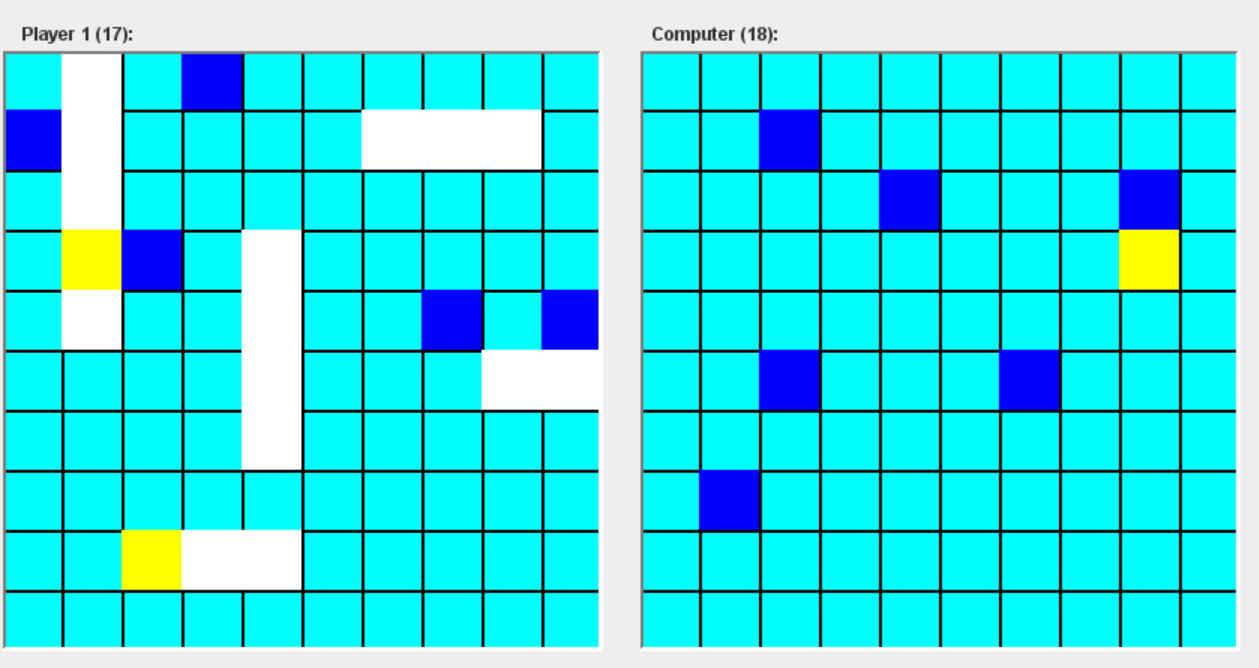


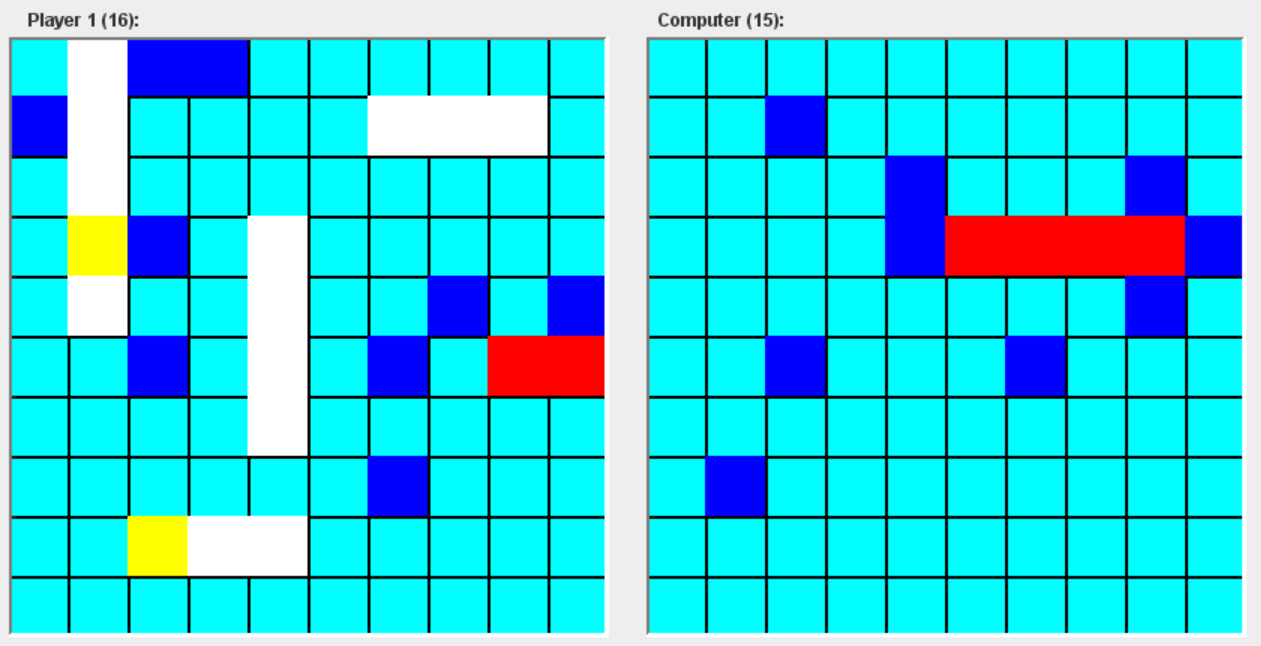
A white cells represents the placements of the ship.

On Clicking of the **Start** button. A game will start and computer will automatically arrange its own ships based on the selected strategy.

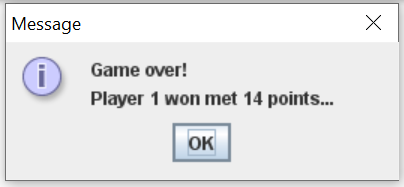
A player has to click on the Computer grid to start the game,

* Blue cell represents the miss hit
* Yellow cell represents the partial hit
* Red cell represents the complete ship destruction.





Once after all the 5 ships got destroyed. Application will show the below message.

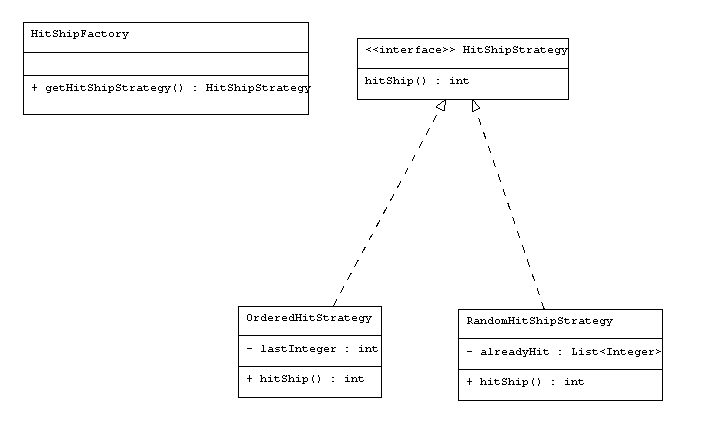


# Design Patterns used

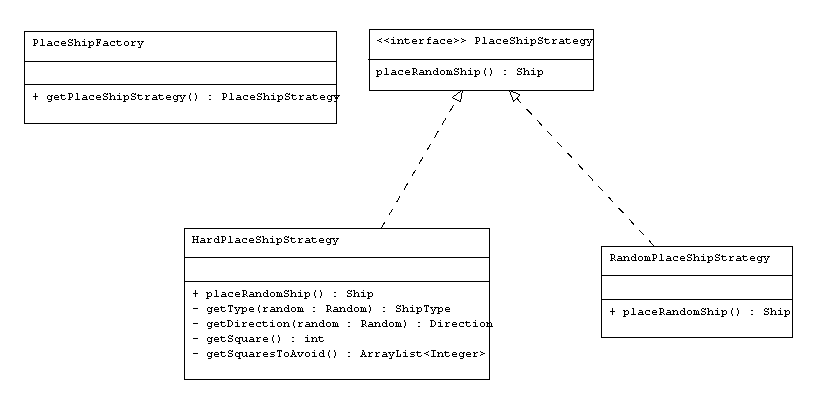
## Strategy Pattern

There are two strategies used in the battleship game i.e., Hit and place ship strategies which are used by Computer player to generate automatic placement of ships in grid and automatic hit of the opposite player grid.

**HitShipStrategy**

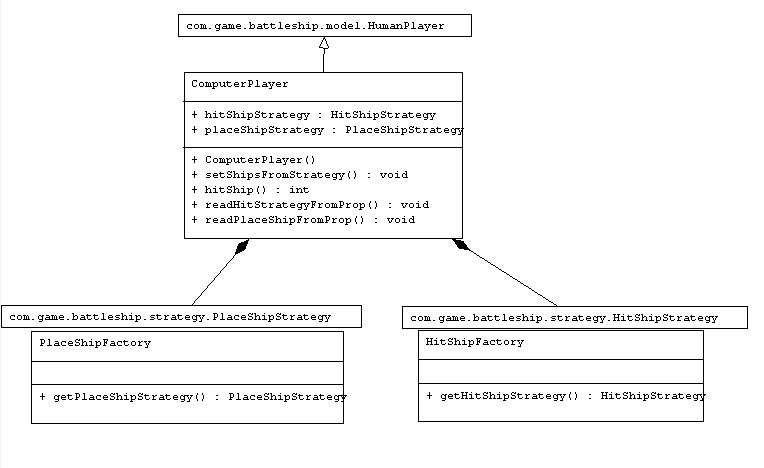


**PlaceShipStrategy**



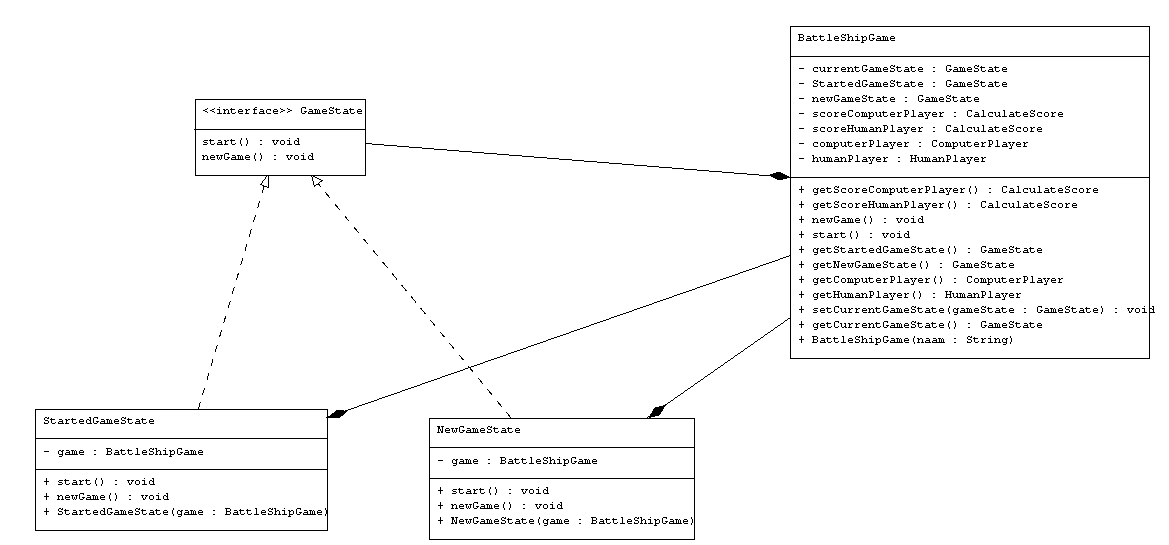
## Factory Pattern

There are two factory patterns used to invoke the hit and place strategies at run time. Computer player uses these factory patterns to get the respective strategies at run time.



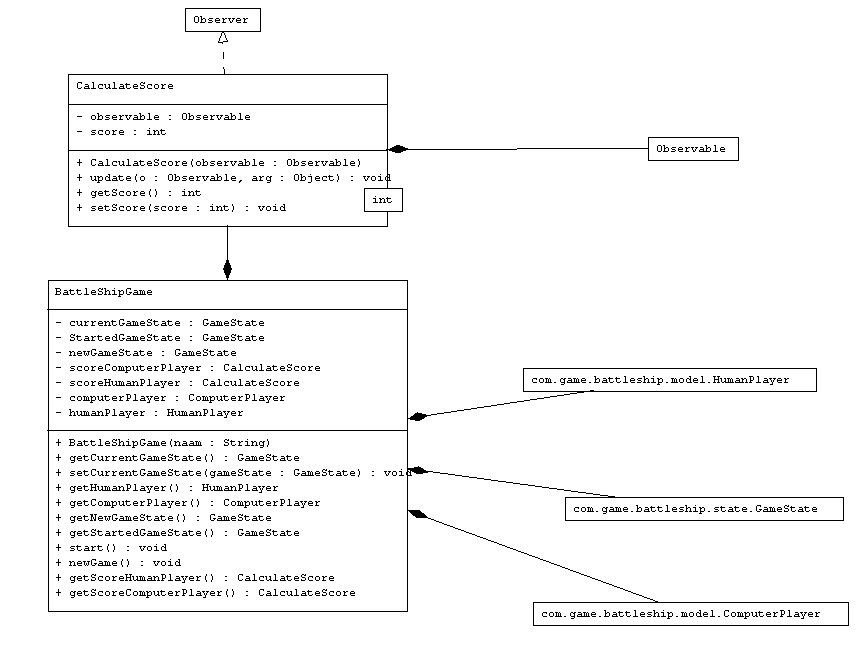
## State Pattern

State pattern is used in this project to get the current state of the project either current game or new game



## Observer Pattern

Observer pattern is used to notify computer and player when each hit on others grid.



# Challenges faced

The Battleship game is played on 10x10 Grid.

While calculating the placement of the ship , need to validate the placement whether its out of bound or overplaced on top of other ships.

Added check on every ship add to check ship is in bounded grid and should not overlap on the other ships.

For Horizontal and vertical placements of the ship . proper calculations has to be done to place the ships without any error.

# Improvements

* The Battleship game is developed on 10x10 Grid. This can be changed to MxN Grid.
* More strategies can be added for ship placement and hit placement
* A new singleton design pattern can be added to load StrategyProperties.properties on application start.
* Number and types of ships are hardcoded in source code . this can be extracted to properties file for future changes to achieve configurability.

# Test the Application

Battleship Application can be run in Two ways.

1. After importing the Application into eclipse , click on BatteleShipApplication and Run as Java Application.
2. Since the Battleship application uses java inbuilt aws interface , the application can start from the command line using **‘java -jar battleship-0.0.1-SNAPSHOT.jar’** (make sure jdk 1.8 installed on your system)