Battle Ship Game

Faculty of Computer Science and Electrical Engineering, Fachhochschule Kiel – University of Applied Sciences

Ahsan Yusuf, Manuel Bois, Talha Bin Nusrat, Vikash Pirthini

Introduction

Battleship is a very well-known battle game, played all over the world with different forms and with different rules. The project is to develop a web application where user can play Battleship game. This Web site also supports multiple player session also. The page that we design mainly consist of four main side options,

•**HOME:** Main webpage with login option.

•**PLAY INSTRUCTION**: give instruction to user about how to play game.

•**GAME:** page where field is located.

•**YOUR MESSAGE:** extra page for communication. The Web site support multiple user game playing with sign-up and sign-in.

Game Play/ Instruction

This is primarily a guessing game, where each player trying to figure out the location of another's ships usually of enemy and shoot them down (main process). The first thing the player must do in this game is arrange all ships in to the board which is 10X10 in size. There are total of 3 ships with 2 square, 4 square and 5 square in length. They can be placed horizontally and vertically. Each player will also being possession of a second grid of the same size. Where the first grid is their own ships, the second one is a mirror of their opponent battlefield. The first player to shoot down all three of the other's ships wins the game.

Technology Used

• Html, CSS, JavaScript (Web Designing)

• Socket.io (Websocket)

• MongoDB (Database)

• Phaseri.io (Game Graphics)

Program Structure

Figure 1. shows over all dynamics of Battle ship game. On one hand we have multiple user or multiple players waiting to get authenticated or waiting for joining with the game server through Websockets session. On the other hand game is being initiated. Session of players are saved in MongoDB and also sent back to HTTP/Websocket Server. After game is initiated with the session ID, different connected players can start their game. Where as in figure 2 Server side game components communication is shown :
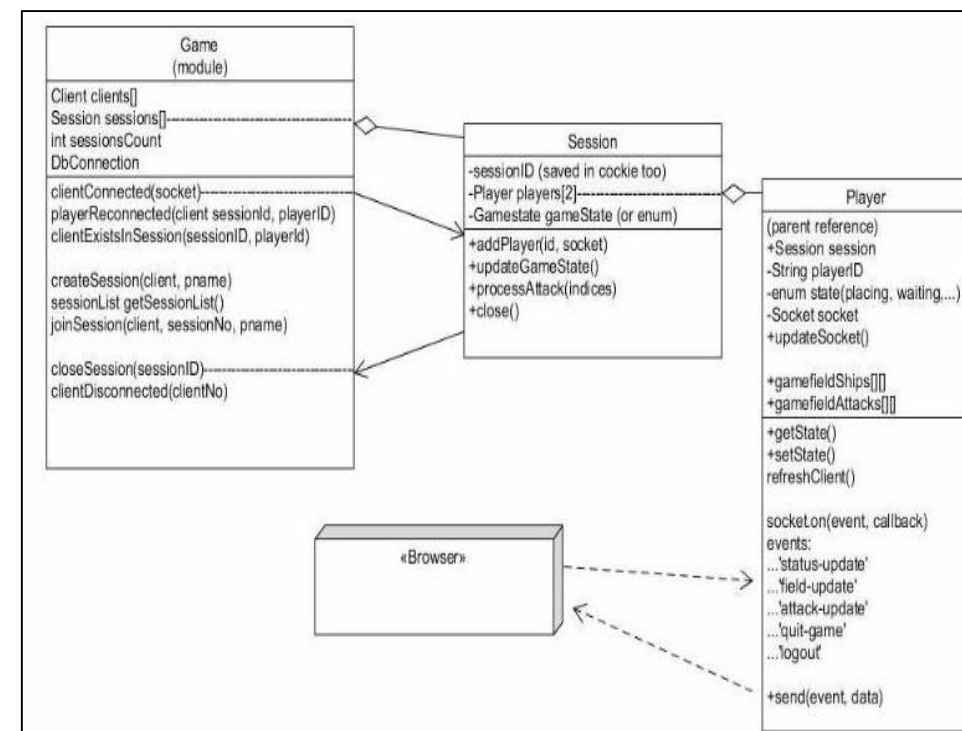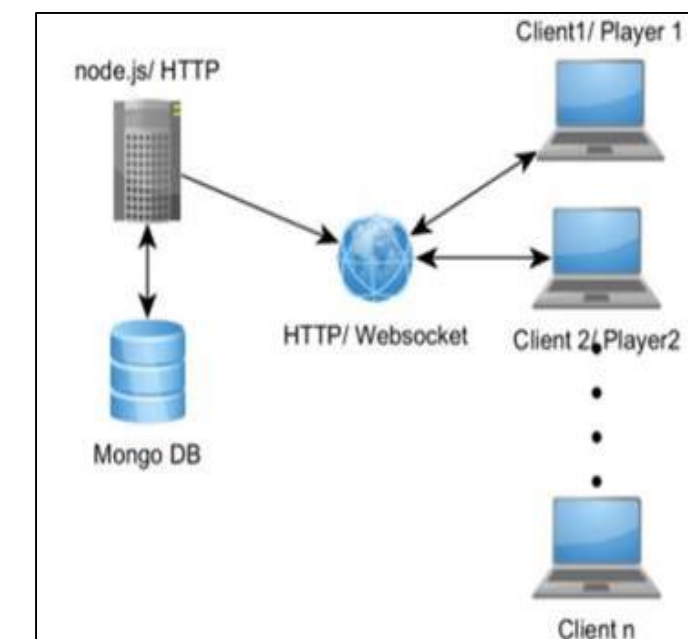

Fig. 1 Basic Structure

Fig. 2 Serverside Game Components

Figure 3 is our main page where user/player interact lively with game. Player can choose different tabs. If Player wants to read the instruction of how to paly then it can easily be done. In the Game tab player can initiate the 'game'. While User/player can send messages from 'your message' tab (future goal). The page gives you immense drama and a taste of real time battle with your enemies.
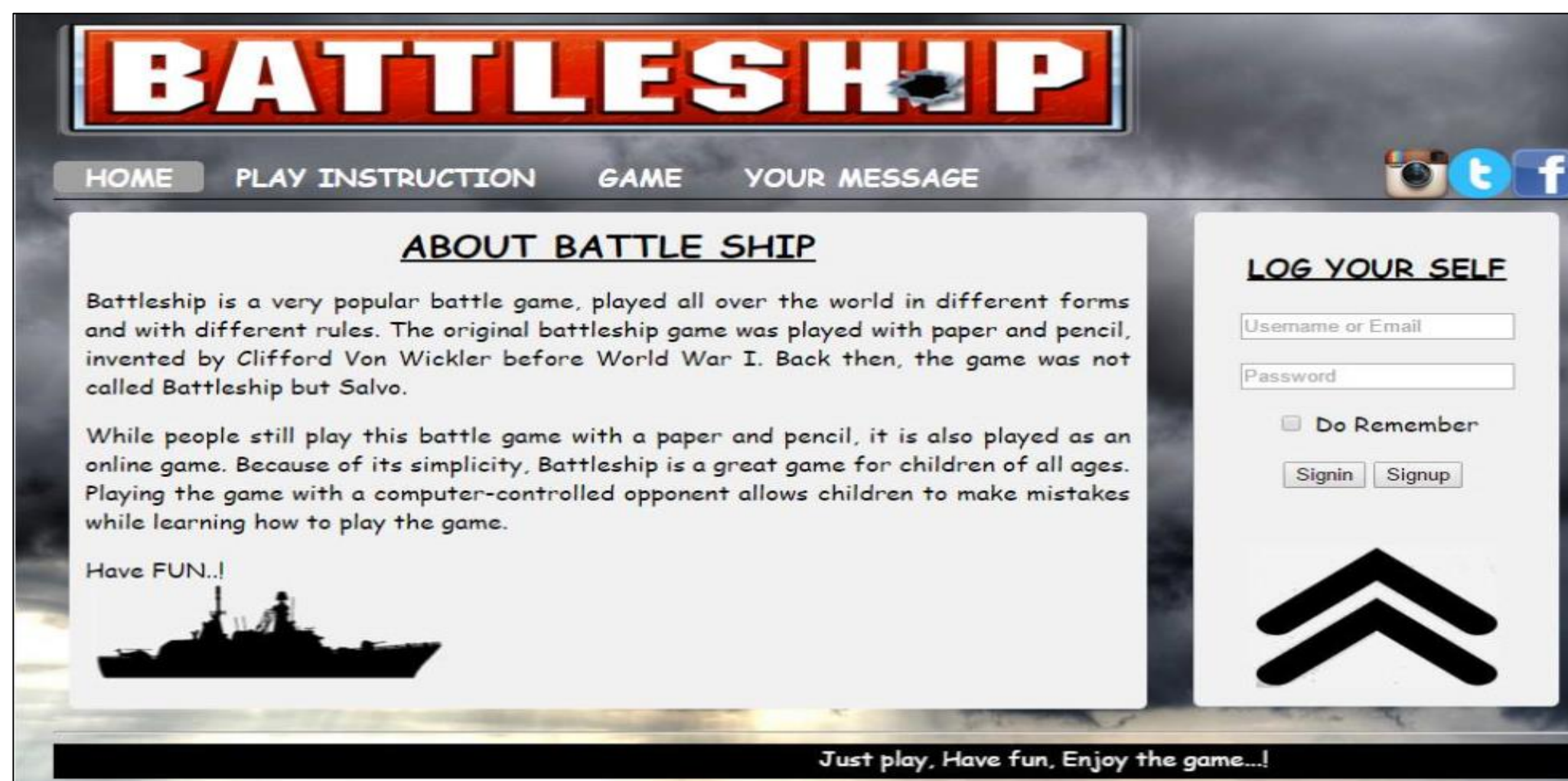

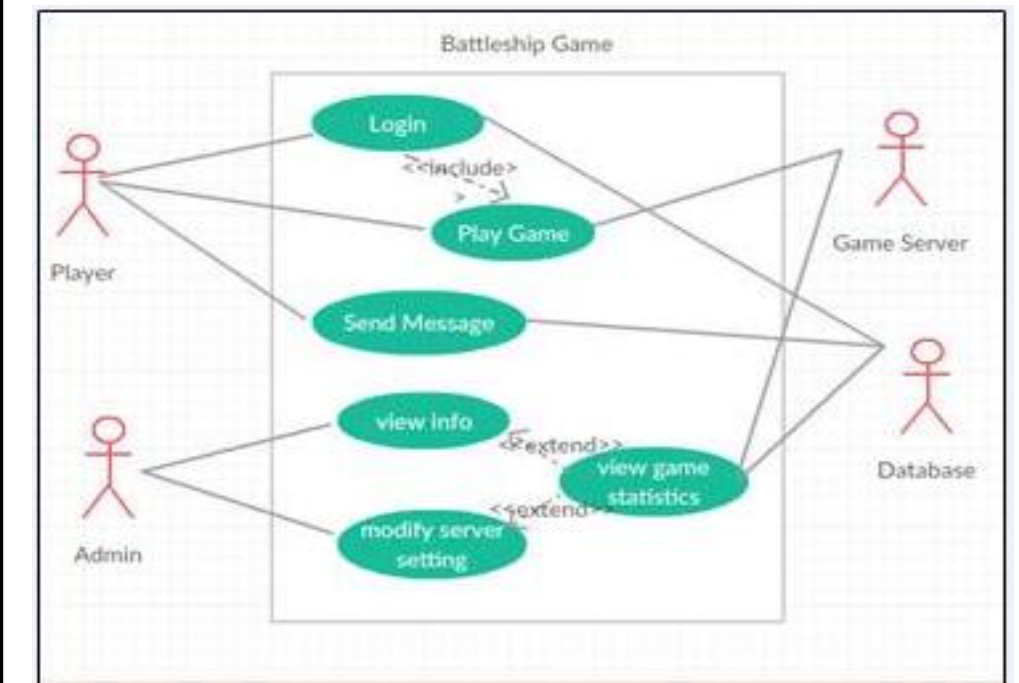
Fig. 3 Main Web Page

Use Case

Fig. 4 UML Diagram

• Player can log-in, play game and send message via using our web page

• Database has all player information along with the session of game

• Admin can view all information about the game statistics and modify service setting

• Websocket communication between server and client

Conclusion and Future Work

➢ Battle ship is a well known game, played all over the world with different form and different rules. Here we are with an Online Website where multiple user can play Battle ship game in virtual board

➢ More user related data stored in DB

➢ Merging of Phaser.io for better game UI

➢ User messaging feature

➢ Social Media merger

➢ Game request can be sent from webpage to different social interacting sites