**Designing a Software Solution for Battleship**

Pudota, Arpitha

La Roche University

ISTG6010-2025: OBJECT ORIENTED SYSTEMS

Instructor: Dustin Updyke,MS

March 26, 2025

**Mock Interview Report**

Interview Overview

In order to start the process of gathering requirements in order to design a software version of the board game Battleship, I had a mock interview with Alex, who is a long-time enthusiast and has practiced the game many times. The goal of the interview was to understand the gameplay, common problems in translating the physical game into a software application, and to identify enhancements that are needed of the translated application.

Interview Transcript

Thank you for granting me this interview. How did you feel about Battleship?

Alex: Sure! It’s been since I was a kid when I first started to play Battleship. I have played the board game and digital versions. What I like about the game is how strategic it is, especially trying to predict where the other player’s ships are going to be.

Essential rules and objectives of Battleship Q1

B1: Battleship is a two player game. Each player hides a 10x10 grid with five ships positioned differently on the grid, each of different sizes, which is concealed. The purpose is to call the coordinates of all the ships desired to be sunk. When all occupied spaces of a ship are hit, the ship is sunk. With it being sink all five of the opponent’s ships, the first player to do so wins.

Q2: What’s the main actions need to be fulfilled during the gameplay?

A2: Specific actions include placing ships, taking turns firing at coordinate numbers, marking hits and misses, receiving word on sunk ships, and monitoring the direction of and your own and opponent’s ships. Each turn has the caller specify a coordinate and is informed whether or not it hit the target.

Q3: What mistakes are made in the physical version is there anything that we should be addressing in the software.

In physical game it’s easy to place ships off grid or overlapping by accident. It should be restrained from allowing invalid placemens. The second problem can be players accidentally firing the same coordinate twice. Disabling used cells should prevent that.

Q4: How important is visual and audio feedback in a digital version?

A4: Very important. For example, with visual feedback such as color crossed hits and misses, and animations or sound effects, players remain engaged. It should be short but sufficient.

Q5: What additional features can be added to the modern game to enhance the overall gameplay experience?

A few ideas for solo play with A5 AI opponent, online multiplayer, monitoring of player statistics and tracked stats, thematic boards etc. These features would serve to make the game more interesting and easy to grasp.

**User Stories: Core Gameplay**

User Story 1: Ship Placement

While playing, I wish I am able to put in my five ships on a grid so that I can be ready for the game before it starts.

Acceptance Criteria:

The grid was 10x10, and the numbers were clearly labelled.

There are vertical and horizontal ship placement.

Grid is not allowed for ships to overlap or ships can’t extend beyond the grid.

Placement is possible before rotatable ships are in place.

The user can also clear all placements using a reset button.

The layout is locked by a confirm button and the game starts.

User Story 2: Firing at the Opponent’s Grid

I want as a player to pick a coordinate off my opponent’s grid to shoot at and try to hit their ships.

Acceptance Criteria:

Any uncalled coordinate can be selected by players.

A firing is preceded by a confirmation prompt.

After every shot, the system will return “hit” or “miss”.

When a hit occurs, a red indicator indicates; when a hit does not take place it is shown as a white indicator.

Visual marks are placed onto fired coordinates and cannot be selected again.

User Story 3: Turn Management

In the game, I am a player and I want the game to arrange the alternate turns automatically in order to allow the game to be structured and fair.

Acceptance Criteria:

The display clearly shows who is finished.

It is allowed only for the active player.

Once a move is complete the turn then naturally changes automatically.

It shows recent moves with the outcome in a log or status area.

**User Stories: Tracking, Feedback, and Game Conclusion**

User Story 4: Tracking Fired Shots

From a player perspective, a setting I want is for me to remember which coordinates I’ve fired at so that I don't refire moves.

Acceptance Criteria:

All tried shots are shown on a personal tracking grid.

Missed and hits are integrated, and red and white are used for marking them.

The coordinate is shown when hovering over any cell.

After every move, the grid gets updated automatically.

User Story 5: Notifications for Sunk Ships

When sinking one of my opponent’s ships, as a player, I want to be notified to keep track of my progress.

Acceptance Criteria:

If one does well while in the process of sinking the Cruiser, the system responds with the message “You sunk the Cruiser!”

This gives a visual update through which ships have been sunk.

A sunk ship plays a unique sound effects or animation.

User Story 6: Declaring the Winner

Once all ships have been sunk, I as a player want the game to end and declare the winner.

Acceptance Criteria:

Remaining ships are checked by system continuously.

The game ends when a player has no remaining ships.

It shows winner’s name on a victory message.

At the end both players’ full boards are revealed.

The players can also restart or exit the game using a menu.