Software Requirement Specification Document for Battleship Console Application

Team: Defenders

Members:
1. Sahat al Ferdous Fahim
2. Md Limon Apu
3. Md Saidur Rahman
4. Moshiour Rahman Prince

Contents:

1. Introduction

This document is created with the intend to illustrate the software requirement for a game. This section gives a scope description and overview of everything included in this SRS document.

1.1 Purpose

The purpose of this documentation is to record all the requirements in order to build a successful game. This will explain detailed features of the game. The number of gameplays, game type, instruction for playing the game can be learned from this document. An idea of the total interface is described in this document.
This game is a semester project for the 2nd semester student of Electronic Engineering from Hochschule Hamm Lippstadt. This document is intended for responsible project tutor and professor from the department of Electronic Engineering.

1.2 Scope

This game is called 'Battleship'. It is played in a computers Terminal. Battleship is a two-person game. Each player has the same amount of space to arrange their ships. Battleship is played in a grid system space. At the beginning of the game players will get same numbers of ships. They can choose to arrange their ships manually or random. Players can shoot multiple time as long as they do not miss a target. Winner is decided when one of the players lose all of his/her ships. It could be played either against computer or human.
Furthermore, this game is built for entertainment purpose. It is best played between two humans. This game could be a good leisure activity since most people played this game non-digitally during their childhood.

1.3 Definitions, acronyms, and abbreviations

Player	Someone who interacts with the programme.
AI	Artificial Intelligence.
C file	Source code file.
.txt	Extention for a text file.
2D	Two Dimensional
UC	Use Case
FR	Functional requirement
IR	Interface Requirements
PER	Physical Environment Requirements
UFR	User factor Requirements

1.4 References

1. IEEE software requirement specification standard 830-1998.

1.5 Overview

The rest of the document is built on two parts. The second part is containing overall description of the game. It gives an overall functionality of the game. How the plyers will interact with game is included here with use case diagram and sequence diagram. It also gives information on what type of knowledge a user should have in order to enjoy ‘Battleship’. A brief description of the use case diagram will also be found in this section.

Third and final part of the document represent specific requirements for the game. Most importantly it describes functional requirement for the game. Many other types of requirement technique are used in this section for better understanding to the audience.

2. Overall description:

2.1. Product perspective: The application is independent and totally self-contained.

2.2. Product Function:

The functionality of the application is graphically presented using Use Case Diagram and Sequence Diagram. Description of the respective diagrams are followed by them.

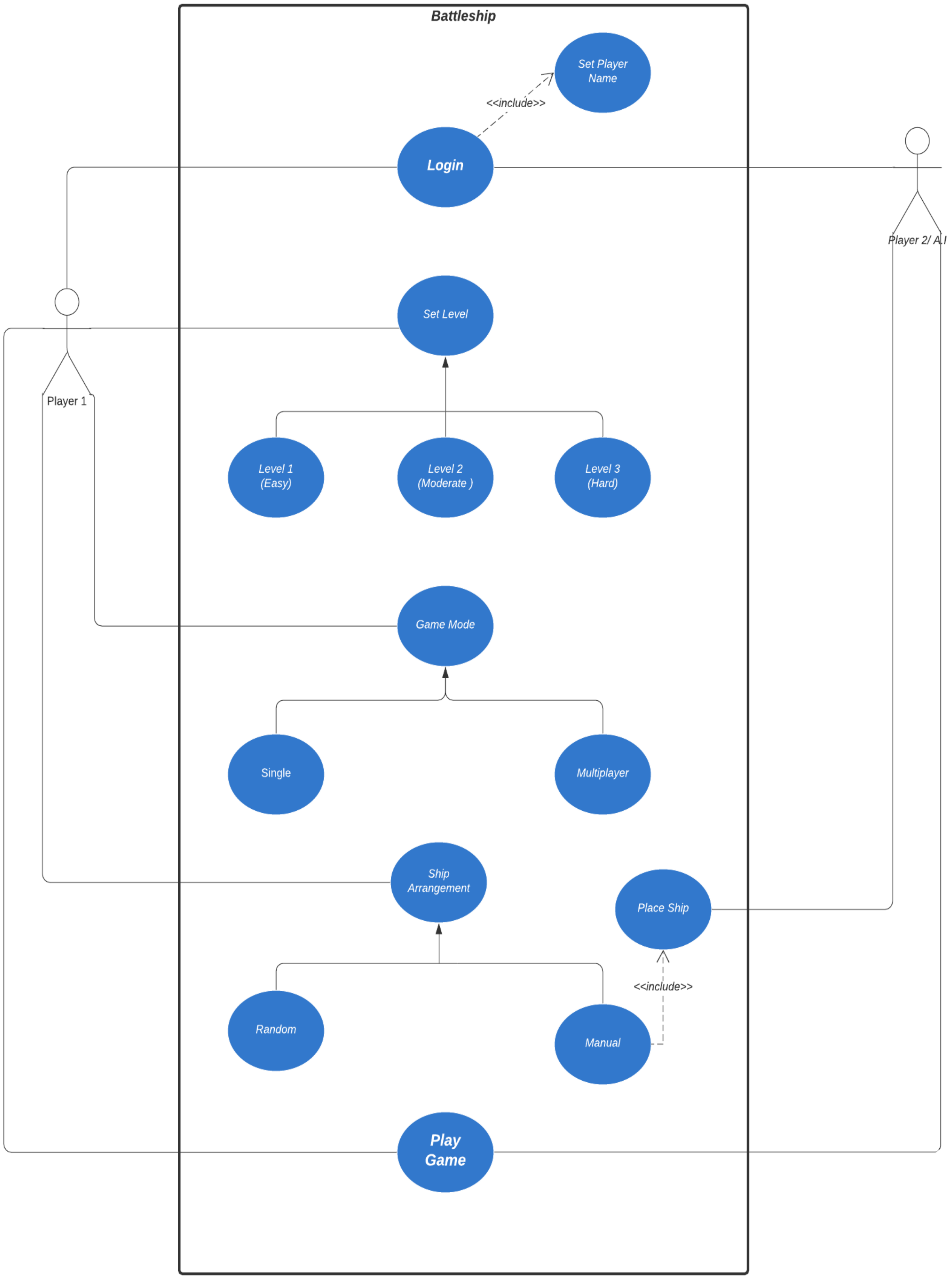

Diagram: I. Battleship Use Case Diagram

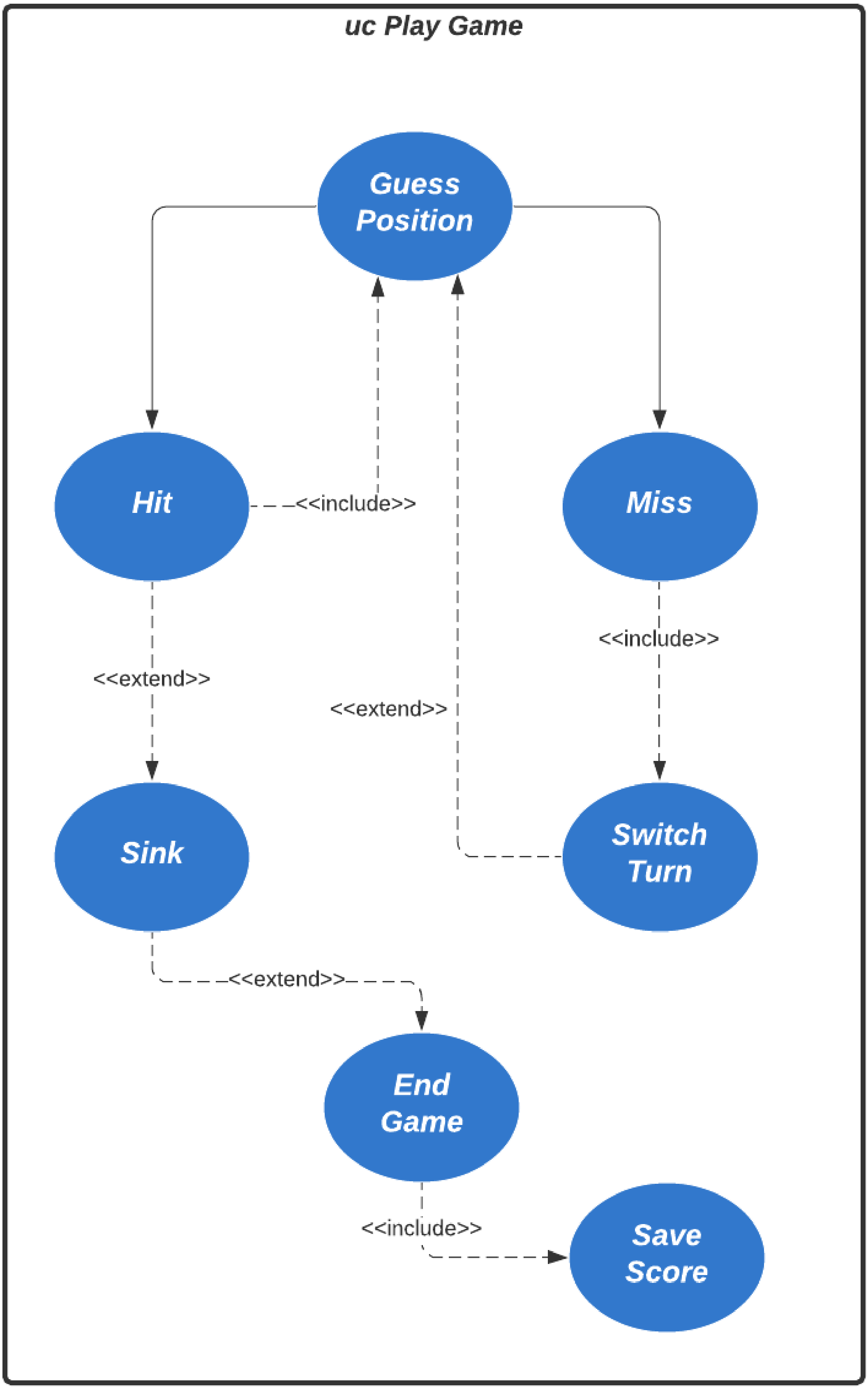

Diagram: II. use case Play Game

Use Case Description:

UC 10: Log In

Player shall input their name to save their score against it. If the game is played against computer, default opponent name is set as “Computer”. If the game is played against another human player, the name of the second player is also inserted.

UC 20: Set Level

Player Selects difficulty level from out of three levels.

UC 30: Game Mode

Player selects game mode. They can select single player mode to play against computer (game AI) or Multiplayer mode to play against another human player (Player 2).

UC 40: Ship arrangement

Player can select ship arrangement mode. They can select random mode in which ships will be randomly placed by the program. Otherwise, they can select manual mode in which they will be able to manually place each ship on the game board. If the game is played against another human player (Player 2), they also get the chance to place their ship manually. If the game is played against computer, the game AI automatically places opponent ships.

UC 50: Play Game

This is the actual play of the game. The use case “Play Game” is further explained in the second use case diagram. A player enters the guess location of opponent’s ship. If the guess is right a hit is declared and the player gets another chance to guess. If a guess position does not contain a ship a miss is declared, and another player gets the turn to guess. Whoever gets to hit all the ships of the opponent first becomes the winner. If a winner is a human payer (Player 1 or Player 2) their total guess count(score) is automatically stored against their name on an external text file.

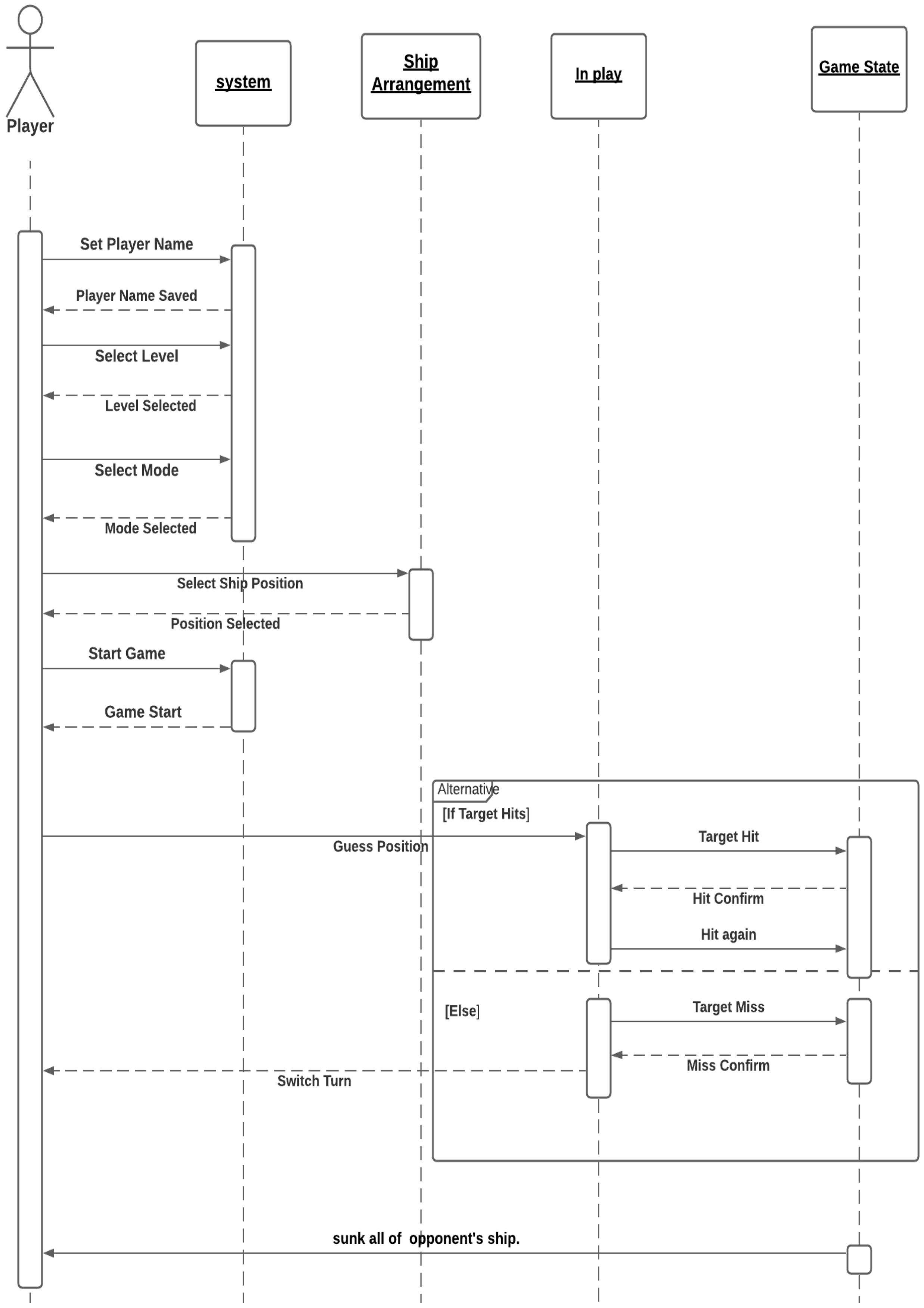

Diagram: III. Battleship Sequence Diagram

2.3. User Characteristics: User shall have knowledge about running console application. Moreover, user shall be able to read and understand game instructions.

2.4. Constrains: The application shall be developed in C programming language.

2.5. Assumption: User has C compiler installed in computer.

3. Specific Requirements

3.1 Functional requirement:

FR10: The application shall be able to play a fully functional version of the classic board game "Battleship".

FR 20: The application shall be able to host a game between two human players.

FR 30: The application shall be able to host a game between a human player and program A.I

FR 40: The game shall have 3 difficulty level.

FR 50: The game shall allow players to manually place their ships, or the game shall be able to randomly place the ships.

FR 60: The application shall be able to store player name, guess count in a .txt file in order to save game score.

3.2 Interface Requirements:

IR 10: The application shall have a 2D coordinate interface which represents the classic game board of "Battleship".

3.3 Physical Environment Requirements:

PER 10: The application shall run on command prompt or on the console of an integrated development environment.

PER 20: The application shall run on varying computer speeds.

3.4 User factor Requirements:

UFR 10: User shall have knowledge about how to run a C program file.

UFR 20: User shall be able to read and understand the provided instruction manual.