Contents

[2 Figures 1](#_Toc89523355)

[3 Design and Analysis 1](#_Toc89523356)

[3.1 Analysis 1](#_Toc89523357)

[3.2 Software Requirements 2](#_Toc89523358)

[3.3 Use Case Models 2](#_Toc89523359)

[3.3.1 Player 1 2](#_Toc89523360)

[3.3.2 Player 2 2](#_Toc89523361)

[3.3.3 A.I. 2](#_Toc89523362)

[3.4 Prototype 3](#_Toc89523363)

[3.4.1 Python 3](#_Toc89523364)

[3.4.2 Java 8](#_Toc89523365)

[3.5 Objectives 9](#_Toc89523366)

[3.6 Requirements Specifications 9](#_Toc89523367)

[3.7 Conclusion 13](#_Toc89523368)

# Figures

[Figure 1: Use case diagram of an A.I. Battleship Game 3](#_Toc89444135)

[Figure 2Sample of Java Spring Dependencies 8](#_Toc89444136)

[Figure 3When using Spring dependencies the Battleship project becomes unloadable and will not able to load rest of the files of the error in dependencies. 8](#_Toc89444137)

[Figure 4Battleship game with ship models and two 10\*10 boards. 9](#_Toc89444138)

# Design and Analysis

There are different possibilities in developing any applications projects which also involves implementing a GAN model. Depending on the methodology used for the development it may also involves deciding the different technologies used and specified for the development. This chapter can also help gather solutions for developing a project that can generate artworks as well as implementing a GAN model with generator and discriminator. The four main components will be analysis, design, requirements, and prototype. Each area will have the necessary details to describe the topics, such as potential solutions, other solutions, strengths, weaknesses, physical constraints, software and hardware limitations, and other information.

## Analysis

Prior to the project's development It is important to keep in mind that some elements are necessary for the project's execution. The likelihood of a project's successful development may be increased by examining a wide range of potential technologies, methodologies, and components. The following guidelines are taken into account before designing:

1. **GAN model**: How each player will take a shot from the game board.
2. **Generator:** Deciding of how and where the ships will be placed.
3. **Discriminator:** to provide challenges to the player in a single player match or to itself in an A.I. player match when successfully trained and added to the project after the development of the game.
4. **Internet Connectivity:** depending on the design, access to the network can be supplied by a USB-dongle stick through the 4G network or WIFI technologies.
5. **Available IT Technologies**: At this stage, selecting and researching appropriate technology will help you avoid a number of problems. Depending on the design requirements, IT technologies should be carefully chosen.

## Software Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Version | Purpose |
| Python | Program Language | 3.1 | contains many libraries and it is more simple |
| TensorFlow | FrameWork | 2.7.0 | Useful to implement AI |
| Visual Studio Code | IDE | 1.62 | Supports many languages, git and libraries |
| Github | Repository | None | To store the project |
| Jupyter Notebooks | Web Application | 6.4.5 | use to create and share documents that contain live code, equations, visualizations, and text. |

## Use Case Models

The following diagram in Figure 1 will help describe the stages of a Battleship game. From the diagram there are three actors that are mainly involved game. The table below describes each use case.

### Player 1

The Primarily actor. When Player 1 starts the Battleship game he/she will have different choices of game modes to select whether to player single player, multiplayer or A.I. mode.

### Player 2

The player 2 will be involved when Player 1 selected the multiplayer mode. From this mode is where both players can play the Battleship game and taking turns after each shot taken.

### A.I.

The A.I. is mostly involved in both a single player and A.I. mode. In the Single Player mode A.I. will play against player 1 and provides a challenging or simple match based on the difficulties(easy, medium or hard). But in A.I. mode the A.I. player will simply play against itself instead of the player with no difficulties included in order to further test A.I.’s intelligence in the game.

Diagram, schematic

Description automatically generated

Figure : Use case diagram of an A.I. Battleship Game

|  |  |
| --- | --- |
| Use Case | Use Case Description |
| Multiplayer mode | To enter a mode where two players can play the game. |
| Singleplayer mode | To enter a mode where a player will play against an A.I. with difficulties included between medium, easy or hard. |
| A.I. mode | To enter a mode where an A.I. will play against itself. |
| Ship Placement | To place ships in different positions. |
| Take Shot | Each player must choose the position to shoot in the Battleship game. |
| Take Turn | Players will take turns after each shot taken. |
| Win/Lose | If the players win or lose the game they have an option to repeat the match or close the game. |

## Prototype

### Python

Now to look at a sample code of a battleship game with only multiplayer mode.

## Objectives

The real goals and focus for this project is not only making a multiplayer game of a battleship but to train an AI and implement it in the Battleship with following modes of Singeplayer(which includes difficulties and AI mode(ie bug mode) mainly for AI to play against itself. Another focus is to implement a feature where the player can replay the game again when asked for a rematch as well as having a two board layouts with 10\*10 grid as shown in image below with the following ships:

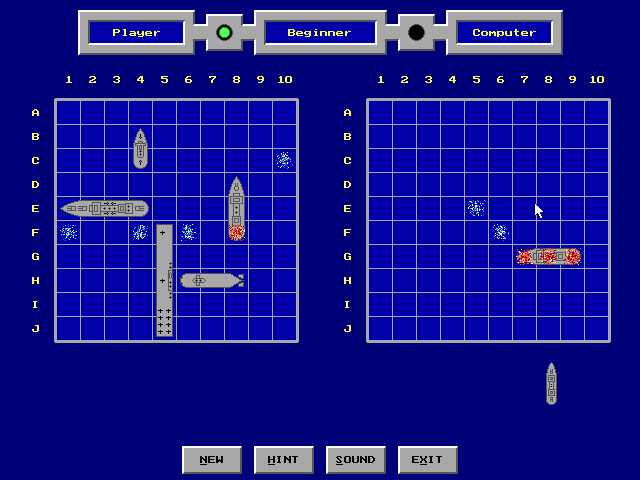


Figure Battleship game with ship models and two 10\*10 boards.

## Requirements Specifications

Providing details of all menu and button functions – indicating for each function:

1. Type(Menu/Button).
2. Location (page location where function is located).
3. Action (describe what happens when item is selected).
4. Player 1 Functions:
   1. Mode Selection:

**Type**: Menu Item

**Location**: Main Landing Page

**Action**: When the application runs a page will be displayed with a message asking “Select Mode” with three buttons which are Singleplayer mode, Multiplayer mode and an AI mode.

* 1. Choose Difficulty:

**Type**: Menu Item

**Location**: When selecting a Singleplayer mode

**Action**: after selecting a Singleplayer mode it will take you to another page where the player is asked to choose difficulties between three options of easy, medium or hard.

* 1. Confirm Difficulty:

**Type**: Button

**Location**: In “Choose Difficulty” page

**Action**: A button that is provided in the “Choose Difficulty” page and it is mainly to confirm the selection of a difficulty chosen. Each difficulty differ from another for example an easy difficulty is to make a game more simple, medium difficulty is for providing standard challenges as for hard difficulty to make the game more challenging.

* 1. Place Your Ships:

**Type**: Menu Item

**Location**: When selecting any mode between Singeplayer and Multiplayer except for Singeplayer mode which will be displayed after Choose Difficulty page

**Action**: When selecting any mode a page will be displayed which contains a 10\*10 board grid layout along with five ships each differ by size. Each player is tasked to place their ships in different positions.

* 1. Confirm Placement:

**Type**: Button

**Location**: “Place Your Ships” page

**Action**: A button provided in “Place Your Ships” page to confirm the placements of ships when pressed.

* 1. Choose Shot Rule:

**Type**: Menu Item

**Location**: After pressing the “Confirm Placement” in the “Place Your Ships” for both Multiplayer and Singleplayer modes.

**Action**: When the Ship Placements is finished a page is displayed asking each player to a “Shot Rule” with the options of Single Shot Rule or 5 or 6 Multi-Shot Rule.

* 1. Confirm Shot Rule:

**Type**: Button

**Location**: “Choose Shot Rule” page

**Action**: A button provided to confirm the Shot Rule chosen for the Battleship game.

* 1. Take Shot:

**Type**: Menu Item

**Location**: After confirming a shot rule in both modes.

**Action**: After deciding on the shot rule the next page will be displayed with 10\*10 board and in that page is where a player chooses which position on the board to click(i.e. strike).

* 1. Finish Turn:

**Type**: Button

**Location**: “Take Shot” page

**Action**: When a player decided on a position to strike a Finish Turn button will be pressed to end the first player’s turn which will be the next player’s turn whether it is Player 2 or a bug player and each turn is taken based on the shots taken for each ship on both sides.

* 1. Win/Lose

**Type**: Menu Item

**Location**: When the game of Battleship is finished with one of the players either won or lost to another player and this is for all modes especially AI player mode.

**Action**: A page will be displayed with different messages that if a player manages to take out every cruises on the other player’s board a win page will be displayed with the following “You win”. However, if a player lost all the ships to another player then a lose page will be displayed with a different message of “You Lose”. What’s common between the win/lose pages is that each page has two buttons which asks the player for “Rematch” or “Finish”.

* 1. Rematch:

**Type**: Button

**Location**: Win/Lose page

**Action**: A rematch button in the win/lose pages for when pressed it will repeat the game process all over again.

* 1. Finish:

**Type**: Button

**Location**: Win/Lose page

**Action**: A Finish button in the win/lose pages for when pressed it will end the game session.

1. Player 2 Functions
   1. Place Your Ships:

**Type**: Menu Item

**Location**: When selecting any mode between Singeplayer and Multiplayer except for Singeplayer mode which will be displayed after Choose Difficulty page

**Action**: When selecting any mode a page will be displayed which contains a 10\*10 board grid layout along with five ships each differ by size. Each player is tasked to place their ships in different positions.

* 1. Confirm Placement:

**Type**: Button

**Location**: “Place Your Ships” page

**Action**: A button provided in “Place Your Ships” page to confirm the placements of ships when pressed.

* 1. Choose Shot Rule:

**Type**: Menu Item

**Location**: After pressing the “Confirm Placement” in the “Place Your Ships” for both Multiplayer and Singleplayer modes.

**Action**: When the Ship Placements is finished a page is displayed asking each player to a “Shot Rule” with the options of Single Shot Rule or 5 or 6 Multi-Shot Rule.

* 1. Confirm Shot Rule:

**Type**: Button

**Location**: “Choose Shot Rule” page

**Action**: A button provided to confirm the Shot Rule chosen for the Battleship game.

* 1. Take Shot:

**Type**: Menu Item

**Location**: After confirming a shot rule in both modes.

**Action**: After deciding on the shot rule the next page will be displayed with 10\*10 board and in that page is where a player chooses which position on the board to click(i.e. strike).

* 1. Finish Turn:

**Type**: Button

**Location**: “Take Shot” page

**Action**: When a player decided on a position to strike a Finish Turn button will be pressed to end the first player’s turn which will be the next player’s turn whether it is Player 2 or a bug player and each turn is taken based on the shots taken for each ship on both sides.

* 1. Win/Lose

**Type**: Menu Item

**Location**: When the game of Battleship is finished with one of the players either won or lost to another player and this is for all modes especially AI player mode.

**Action**: A page will be displayed with different messages that if a player manages to take out every cruises on the other player’s board a win page will be displayed with the following “You win”. However, if a player lost all the ships to another player then a lose page will be displayed with a different message of “You Lose”. What’s common between the win/lose pages is that each page has two buttons which asks the player for “Rematch” or “Finish”.

* 1. Rematch:

**Type**: Button

**Location**: Win/Lose page

**Action**: A rematch button in the win/lose pages for when pressed it will repeat the game process all over again.

* 1. Finish:

**Type**: Button

**Location**: Win/Lose page

**Action**: A Finish button in the win/lose pages for when pressed it will end the game session.

1. Player AI Functions
   1. Win/Lose

**Type**: Menu Item

**Location**: When the game of Battleship is finished with one of the players either won or lost to another player and this is for all modes especially AI player mode.

**Action**: A page will be displayed with different messages that if a player manages to take out every cruises on the other player’s board a win page will be displayed with the following “You win”. However, if a player lost all the ships to another player then a lose page will be displayed with a different message of “You Lose”. What’s common between the win/lose pages is that each page has two buttons which asks the player for “Rematch” or “Finish”.

* 1. Rematch:

**Type**: Button

**Location**: Win/Lose page

**Action**: A rematch button in the win/lose pages for when pressed it will repeat the game process all over again.

* 1. Finish:

**Type**: Button

**Location**: Win/Lose page

**Action**: A Finish button in the win/lose pages for when pressed it will end the game session.

## Conclusion

The stance process for the design and analysis document's requirements, chosen methodology, and necessary tools, such as the IDE, programming languages, and frameworks, are among its most difficult steps. The requirements and tools listed in this document were chosen for the project's development; however, some requirements and tools may change during the project's implementation phase depending on its progress.