5/9/2017

Pawns of War

Project Report

Kevin Roberts X13115952
NATIONAL COLLEGE OF IRELAND

Declaration Cover Sheet for Project Submission

SECTION 1 *Student to complete*

Name:
Student ID:
Supervisor:

SECTION 2 Confirmation of Authorship
The acceptance of your work is subject to your signature on the following declaration:
I confirm that I have read the College statement on plagiarism (summarised overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature:_______________________________________ Date:__________

NB. If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the College's Disciplinary Committee. Should the Committee be satisfied that plagiarism has occurred this is likely to lead to your failing the module and possibly to your being suspended or expelled from college.

Complete the sections above and attach it to the front of one of the copies of your assignment,

Table of Contents

Executive Summary

The purpose of this document is to outline the development of a game that has been written, developed, and designed by one person. This document will show how far the game has come along with the features that proved too difficult to get done in the time frame and the future development that will rectify them features.

Pawns of war uses Unity engine, this engine is responsible for multiple awards in 2016 for titles such as, Firewatch (awarded best 3D visuals), Job Simulator (awarded best VR game), and Lara Croft Go (awarded best mobile game). Unity offers a user-friendly interface with outstanding graphics and performance.

Pawns of War was built from the ground up to offer the best user experience possible. However, this is not without its challenges as this game does not follow the main stream approach. There are limited resources available for a game like this.

In addition to the Unity engine, Maya Autodesk was heavily involved in the making of this game. Maya was used to create the characters that are used in place of chess pieces. The textures that were added to the character helmets in Maya was easily imported into unity from the Maya .fbx file, there was no need to make any additions when opening them up in the Unity screen, apart from scaling the characters to fit the chessboard.

The end goal of this game is to provide an immersive user experience, that can be played by gamers around the world. Additionally, I wanted to create a game that will continue to be developed in the future.

In Pawns of War you do battle with your friends to see who is the ultimate strategy master.

1 Introduction

In this document, I am going to be discussing the development of my project and the steps I took in creating it. The appendices provided will include the Project Proposal and my monthly diaries that began in September and ended in April.

1.1 Initial Game Concept

The basic concept of this game is strategy and tactics, what will make the difference when playing this game against others is how far into the game you can calculate the other persons moves. Essentially this game will feature the main mechanics of chess but what makes this game unique is that you will be battling each other as part of a clan of warriors. These battles will take place on the different arenas, they will be depicted as the territory for the individual clans.

1.1.1 Further Game Concept

When you face off against your opponent you will not only be battling them in a strategy and tactics based game but you will need to fight for the square that you are trying to capture. Players will enter a 2.5D fighting game were each character will have their own unique abilities, in here players will fight for the square and whoever is victorious will win the square and attain the other players piece.

1.2 Background

When you are facing such a rise in video game culture and more and more video games are getting created every year, I feel that big companies are just regurgitating the same type of game over and over every year. This project came about through wanting to bring a classic into the modern day, it got me thinking just how classic I wanted to go and that brought me to chess. One of the oldest games that two people can play, brought into the modern day of 3d video game and animation.

As a lover of warriors from all walks of life throughout history I got thinking why not put them up against each other in the ultimate strategy game. With all this in mind my fondness for the cinematics in fighting games like Mortal

Kombat, I wanted to combine it all and bring to light what I feel could be an ever-expanding video game.

1.3 Full Game Concept Breakdown

Overview of Genre Setting:

- The main genre of this game is *strategy*
- The view of this game will be top down to get a full view of all the characters on the board.
- The secondary genre that will make up this game is: (historical, fighting game)

Environment (World) Setting:

- War-torn historical setting
- Years 793/1066 to 1185/1868
- The battle ground of the clans
- Traditional weapons of the warriors are used.

Story Characters:

- Vikings: Farmers and raider that have set out to find new land in which to pillage and set up new colonies around the world. They kill anyone that opposes them and make waste of villages they come across.
- Samurai: Having conquered Asia, the samurai have set out to find new land in which to conquer, in a quest to be the most powerful force on earth.

1.4 Aims

The main aim of this project is to create a video game that will appeal to all kinds of people young and old. I feel that the 3D and animation aspect of the game will be the key for attracting younger people to it, however, the fact that it will be a chess game at its core will mean that the older generation will find it a game that will be easy to pick up, easy to understand and it will have a shorter learning curve for them if they feel it is a game they would be interested in.

As the core of this video game is based around the concepts of chess, I am hoping to create a new and exciting why in which a user can play chess, plus with the added functionality, I am aiming to create a video game that will appeal to users of all ages.

The Functional Aims are as Follows:

- Create a turn based strategy game
- Create a tutorial within the start menu
- Create the functionality that allows players to move pieces in a specific way
- Allow for the functionality to have touch input for smart devices and mouse input for personal computers
- Have a full save function
- Have a full load function

The Style Objectives are as Follows:

- Create custom assets using Maya Autodesk. These custom assets include the following:
    - Vikings:
        - Pawn
        - Knight
        - Bishop
        - Rook
        - Queen
        - King
    - Samurai:
        - Pawn
        - Knight
        - Bishop
        - Rook
        - Queen
        - King
    - Chessboard
    - Weapons

2 Technologies

Brief description of the technologies used in the project. Do not copy & paste descriptions from websites here, but describe what it is and how it contributes to your project.

For this project, I will be using multiple technologies, the main piece of software that I will be using to create the game is Unity. For the creation of the characters and I will be using Maya 2016, after the characters have been created I will be adding them to Blender for them to be animated before being added to the game within Unity.

Unity is a cross-platform game engine that I will be using within this project. It will be the main piece of software that I will be using in the creation of this project. In unity, I will be creating scripts that will set up the functionality of the game and push the assets onto the screen when that game is running, it will also be used to run the animations when they are necessary. Unity will also be used to make this project a cross-platform video game.

Maya 2016 is a piece of software that is mainly used in the animation and creation of characters and models, it is one of the leading animation software's in the world. I will be using Maya in the creation of models of which I will be using in the game as my characters.

Blender is also a piece of software that is used in the creation and animation of models and characters. I will be using this software to create rigs on my characters in order to be able to animate them I will also be using it in the animation of my models. The reason for me using two software's that are designed for the one this is because, I am very comfortable in the creation of models in Maya and I feel that Blender is more user friendly when it comes to animating the character.

To allow users to download the game I will be using Amazon Web Services to create a bucket in which the game will be stored. This will be made public to be deployed on a website so that users can download the game for free.

2.1 Hardware

- Computer Mouse (Standard wired and wireless)
  - The controls for this game will be mainly controlled using the computer mouse.

- Smart Devices (Phones and Tablets)
  - If the player opts to use download the game for their smart device they will control it using the touch screen.

2.2 Software

The following is the list of software that was used in developing this game:

- Unity Engine
- Maya Autodesk
- Adobe Photoshop
- Blender

2.2.2 How the software is used

The bulk of this project was done in the Unity engine, here is where all the coding and development took place. Maya Autodesk was used for all the modelling of the characters, their weapons, and the chessboard. In photoshop I created textures and visuals. Blender was used for the animation of the characters.

2.2.2.1 Unity Engine

What this software was used for:

- Level design
- Mechanics and layout of the game
- Character movement
- In-game systems

2.2.2.2 Maya Autodesk

What this software was used for:

- All the character Modelling: Viking and Samurai pieces
- Weapons Modelling: The weapons that each character is holding in the game
- Modelling the chessboard: the arena that the characters are standing on
- Rigging of all the characters in the game

2.2.2.3 Adobe Photoshop

What this software was used for:

- Character textures
- Environment textures
- Start screen design
- Poster design
- Logo design

2.2.2.4 Blender

What this software was used for:

- Animation of the characters: I used blender over Maya for the animation as I was having difficulty with Maya when animation the characters

3 System Requirements

3.1 Functional Requirements

I. The application should start when the user launches the game.

II. The game must begin when the user selects “Play Game”.

III. The characters must move to the appropriate squares.

IV. Users progress must be saved and loaded at the main menu.

V. The game must allow for the user to exit the game at any time.

3.2 Requirements 1: Start Application

3.2.1 Description & Priority

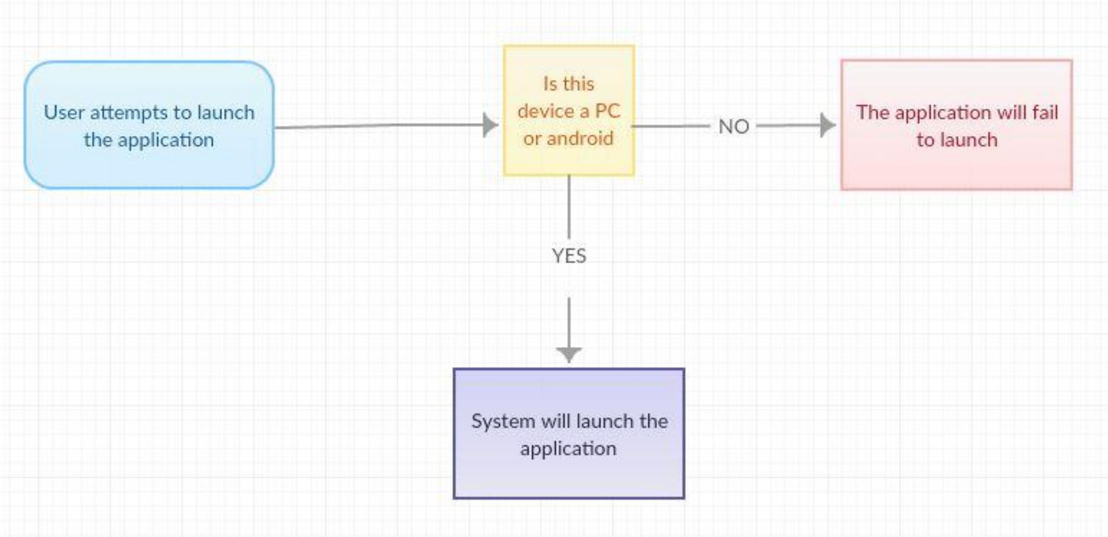
This is the most important requirement of the game, without it the game will not be playable by the users. The game will not load if the hardware cannot launch the application or if the player is unaware how to launch the application.

3.2.2 Use Case

Scope:

This use case it to indicate how a user would go about launching the application.

Use Case Diagram:


User attempts to launch the application
Is this device a PC or android
NO
The application will fail to launch
YES
System will launch the application

Precondition:

The application was downloaded and installed onto an android device using a .sdk file, or the .exe file was downloaded onto a personal computer.

Activation:

This use case will initiate when the player wants to launch the game.

Main Flow:

Actor Action	System Response
I. The user will click on the game icon from their android device or their personal computer.	II. The application will launch into the main menu of the game.

Alternate Flow:

Actor Action	System Response
I. The user will click on the game icon from their android device or their personal computer.	II. The application does not launch as the game may not have been designed for their chosen operating system.

Termination:

This flow will be terminated when the system has booted up.

Post condition:

The user will then be brought to the main menu of the game where it will enter a “wait state” until the user makes another action.

3.3 Requirements 2: Play Game

3.3.1 Description & Priority

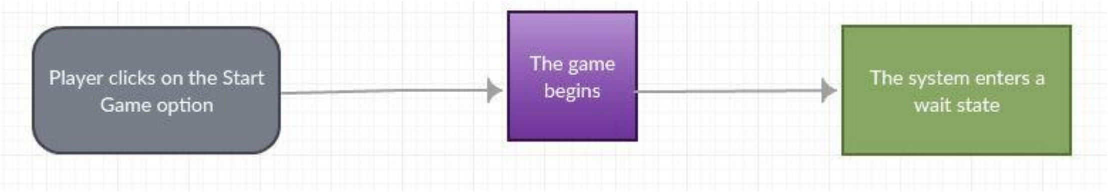
The second most important requirement is the play game. When the application has finally booted up the player must be able to choose the “Play Game” option for the game to begin, and for the player to start playing.

3.3.2 Use Case

Scope:

The scope of this use case is to indicate how the player can start playing the game, when the game is booted.

Use Case Diagram:


Player clicks on the Start Game option
The game begins
The system enters a wait state

Precondition:

The system is running and in a wait state.

Activation:

This use case will begin when the player starts up the application.

Main Flow:

Actor Action	**System Response**
I. Player clicks on the “Play Game” option	II. The game session will be launched

Alternate Flow:

Actor Action	**System Response**
I. Player does not click on the “Play Game” option	II. The game session will not launch and will remain in a wait state.

Termination:

This flow will terminate when the player starts the game or decides to load a previously saved game.

Post condition:

The game will begin

3.4 Requirements 3: Character Movement

3.4.1 Description & Priority

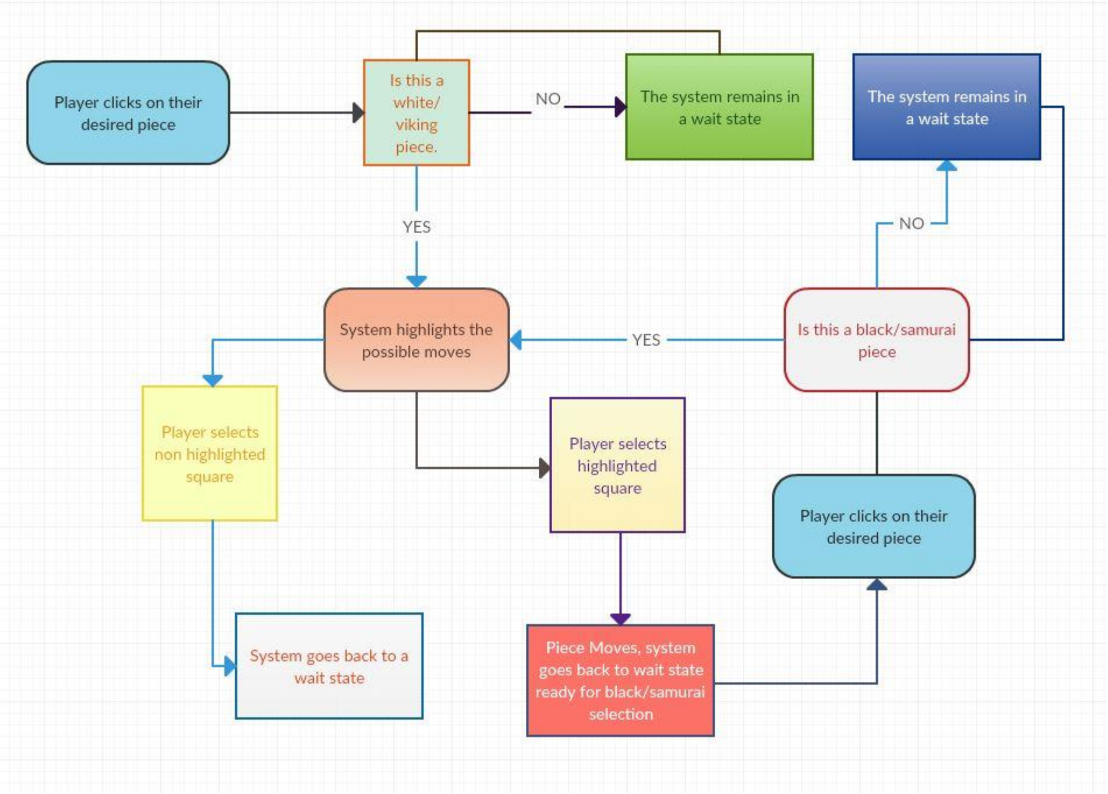
The character movement is third on the priority list. When the player gets to this point they will have launched the game and selected the “Play Game” option, at this point the system will be in a wait state, until the player makes their first move.

3.4.2 Use Case

Scope:

The scope of this use case is to allow the player to move their pieces around the board.

Use Case Diagram:



Player clicks on their desired piece
Is this a white/ viking piece.
NO
The system remains in a wait state
The system remains in a wait state
YES
NO
System highlights the possible moves
YES
Is this a black/samurai piece
Player selects non highlighted square
Player selects highlighted square
Player clicks on their desired piece
System goes back to a wait state
Piece Moves, system goes back to wait state ready for black/samurai selection

Precondition:

There is a new game started or the player have loaded up a saved game and it is running, only then can the player move the pieces.

Activation:

This is activated when the player clicks on a piece in order to move it.

Main Flow:

Actor Action	**System Response**
I. Player clicks on the piece they wish to move.	II. The system highlights the squares they can move to and waits on the decision.

Alternate Flow:

Actor Action	**System Response**
III. Player clicks on the piece they wish to move.	III. The system does not highlight any square as the move with that piece is not possible at that time.

Termination:

This flow terminates after the player has selected the square they wish to move their piece to.

Post condition:

The system will enter a wait state.

3.5 Requirements 4: Load Game

3.5.1 Description & Priority

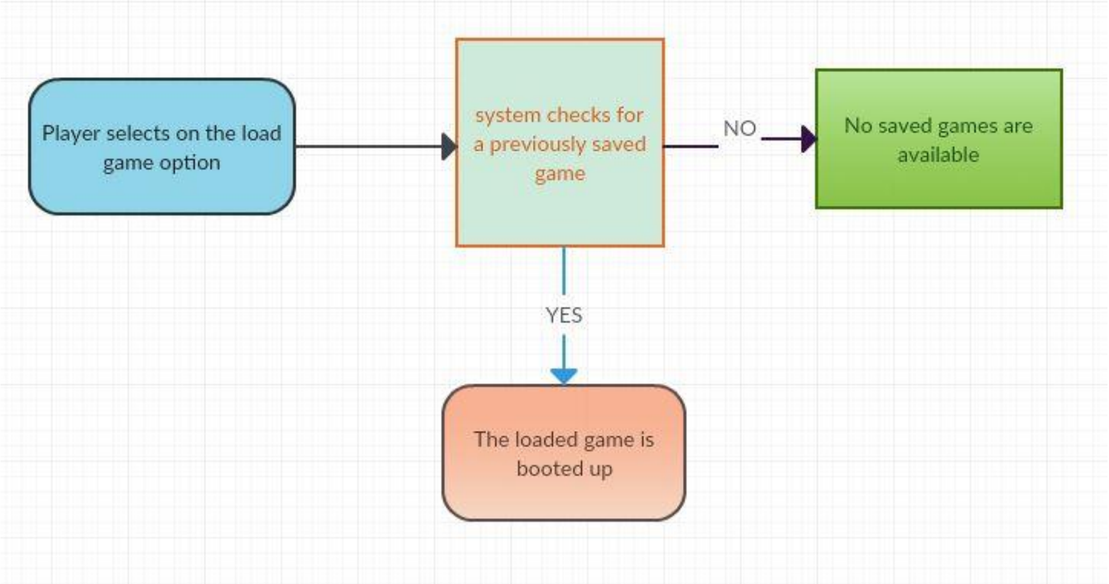
The next in the priority list is the Load Game option. This feature will allow the player to launch their previously saved game. This means the player will still have access to the game at a later stage after it has been shut down.

3.5.2 Use Case

Scope:

The scope of this use case it to indicate how the player can load a game that was previously saved.

Use Case Diagram:


Player selects on the load game option
system checks for a previously saved game
NO
No saved games are available
YES
The loaded game is booted up

Precondition:

The system has been launched and it in a wait state.

Activation:

This use case will begin when the application has been launched by the player.

Main Flow:

Actor Action	**System Response**
I. Player clicks the “Load Game” option.	II. The system boots up the saved game.

Alternate Flow:

Actor Action	**System Response**
I. Player clicks the “Load Game” option.	II. The system does not have any previously saved games.

Termination:

This flow terminates after the player has selected the “Load Game” option and the previously saved game is booted up or the player selects the “Play Game” option and a new game is started.

Post condition:

The game loads the saved game and system enters a waiting state.

3.6 Requirements 5: Exit Game

3.6.1 Description & Priority

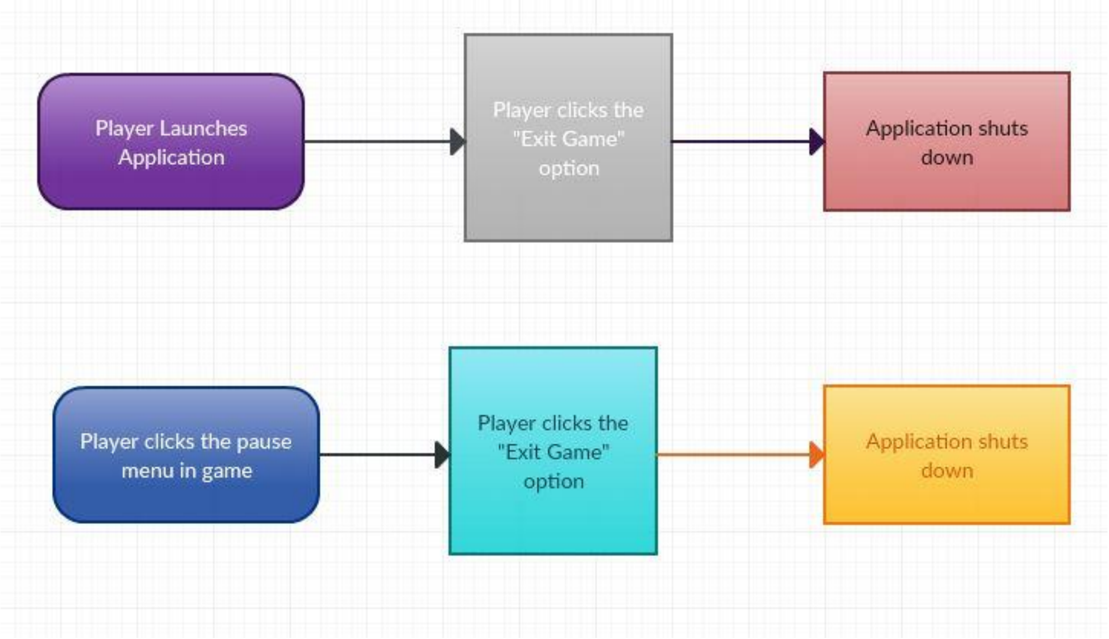
The requirement that is at the bottom of the priority list is the “Exit Game” option.

3.6.2 Use Case

Scope:

The scope of this it to allow the player to exit the game at any time, if the player has just launched the game there will be a “Exit Game” option or if the player is in the middle of the game the start menu will provide an “Exit Game” option also.

Use Case Diagram:


Player Launches Application
Player clicks the "Exit Game" option
Application shuts down
Player clicks the pause menu in game
Player clicks the "Exit Game" option
Application shuts down

Precondition:

The player has launched the game and is in a wait state at the main menu or the player has started into a new game or loaded a saved game and selects the pause menu.

Activation:

This use case starts when the player has launched the game or booted up a game.

Main Flow:

Actor Action	**System Response**
I. Player clicks the “Exit Game” option.	II. The system shuts down

Alternate Flow:

Actor Action	**System Response**
I. Player clicks the pause menu in game.	II. The system loads the pause menu.
III. The player selects the “Exit Game” option.	IV. he system shuts down.

Termination:

This flow has ended when the player clicks the “Exit Game” option. The game shuts down and all functions are stopped.

Post condition:

The game has been successfully stopped and ready for relaunch.

3.2 User Requirements

Having carried out a survey before developing this game I have come up with the following expectations that users will have for a game such as this. They are as follows:

Expectations:

- Clear user interface: It seems that gamers now are more keep on an interface that will give them all the information that they need without being overwhelmed by information.
- Intuitive controls: People want to control their characters with controls that makes sense to the game.
- Ability to save data: With more and more gamers using devices like smart phones and tablets to game, it is important for them to be able to save their progress as these devices tend to be used on the go.
- Strong visuals and audio: These are what keeps the user engaged in the game, without them and the user may not feel much of a desire to play the game.
- Players involvement and problem solving: Games in which players do not have to think about their moves in the game have fallen by the waist side, it has become important for gamers to have to think about moves that they make in the game as it could lead to them loosing.

3.3 Non-Functional Requirements

3.3.1 Usability requirements

Understandability

- Interface should be easy to understand
- Buttons and links should be well defined and easy to find
- The rules of the game should be well written and easy to find

Learnability

- Help should be provided to help achieve in game tasks
- The gameplay should be easy to learn
- Additional help should be provided within the pause menu

Operability

- Interface actions should be consistent
- Error message should be displayed when/if game crashes
- Quick save button enabled while in game
- Smooth transitions between menu screens

Attractiveness

- Menu layout should appeal to destined audience
- Gameplay should be smooth
- Characters animation should run at a high frame rate

3.3.2 Performance/Response time requirements

The performance requirement is decided upon by whether the machine running the Unity game meets the Unity Engines system requirements.

Desktop:

- Operating system:
  - Windows XP SP2+
  - Mac OS X 10.8+
  - Ubuntu 12.04+
  - Steam OS+
- Graphics Card:
  - DX9 (shader model 3.0) or DX11 with feature level 9.3 capabilities.

- CPU: SSE2 instruction set support.

Other Devices:

- iOS player requires iOS 7.0 or higher (dropping 6.0).
- Android: OS 4.1 or later; ARMv7 (Cortex) CPU with NEON support or Atom CPU; OpenGL ES 2.0 or later.
- WebGL: Any recent desktop version of Firefox, Chrome, Edge or Safari.
- Windows Phone: 8.1 or later.
- Windows Store Apps: 8.1 or later.

3.3.3 Availability requirement

To make this game available to the masses I will be developing a website, and will use Amazon Web Services to create a link to the game download that will be implemented into the websites home page.

Deploy a website

- This will be a site that is designed and maintained by myself, there will be a digital download made available through this website for users to download and install.
- This will give me full control over the distribution of the game and would depend on an active internet / social media campaign to spread the word.
- I will also be in full control of the pricing of the game.

3.3.4 Security requirement

There are very little security requirements for this game as it will come down to the client sides privileges. If the user has the access to download content from the internet then they will be able to download the game with ease.

3.3.5 Reliability requirements

The reliability of the game is pretty straight forward, the games .exe and .apk needs to launch on the devices in which it has been designed.

3.3.6 Maintainability requirement

The maintenance of the game will be supported by social media and through the website, here will be where users can get in touch about and bugs or issues with the game. From there I will be able to upload patches for the game that will help with any of the bugs and issues the user may come across.

3.3.7 Extendibility requirement

When the game gets its final release on as many platforms as possible, there will be additional DLC (downloadable content) that will be made available to users though the website. This requirement will be vitel for the longevity of the game, user feedback will dictate the type of DLC that will be necessary to keep users interested in the game.

3.3.8 Reusability requirement

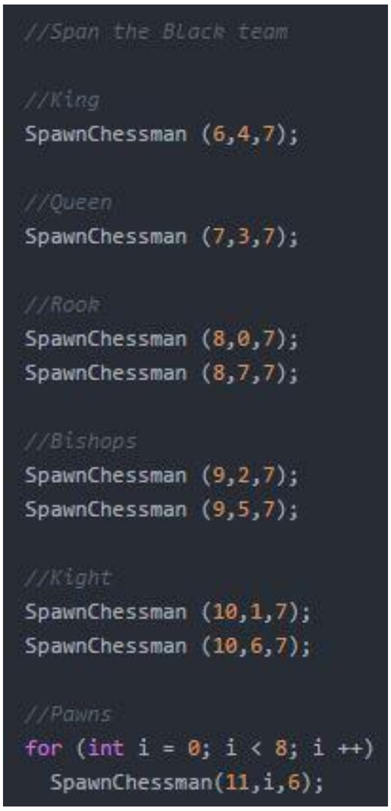
The rigs that I have created for the models and the animation that I have incorporated into them will be critical to the further development of the game and the DLC that will be made available.

4 Implementation

For implementing this project, the code that I have used is C sharp, and the software that it is written in is MonoDevelop. Some of the pieces of interesting code so far is as follows:

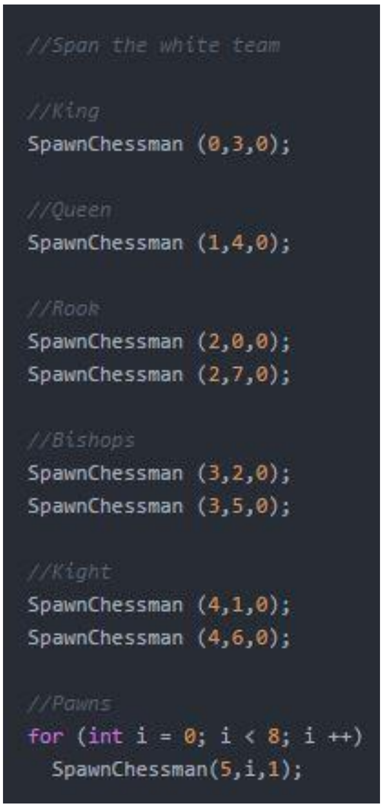
4.1 Chess Pieces

The following piece of code allows for all the white pieces within the project to be push out to the gameplay screen and have their set coordinates for the starting position. The code that pushes the black pieces to the board is very much similar to this, however, the coordinate points are different. The for loop is used on the pawns so that you are not setting individual coordinate for each pawn.



```
//Span the Black team

//King
SpawnChessman (6,4,7);

//Queen
SpawnChessman (7,3,7);

//Rook
SpawnChessman (8,0,7);
SpawnChessman (8,7,7);

//Bishops
SpawnChessman (9,2,7);
SpawnChessman (9,5,7);

//Kight
SpawnChessman (10,1,7);
SpawnChessman (10,6,7);

//Pawns
for (int i = 0; i < 8; i ++)
  SpawnChessman(11,i,6);
```

Figure 1: Spawn White Pawns

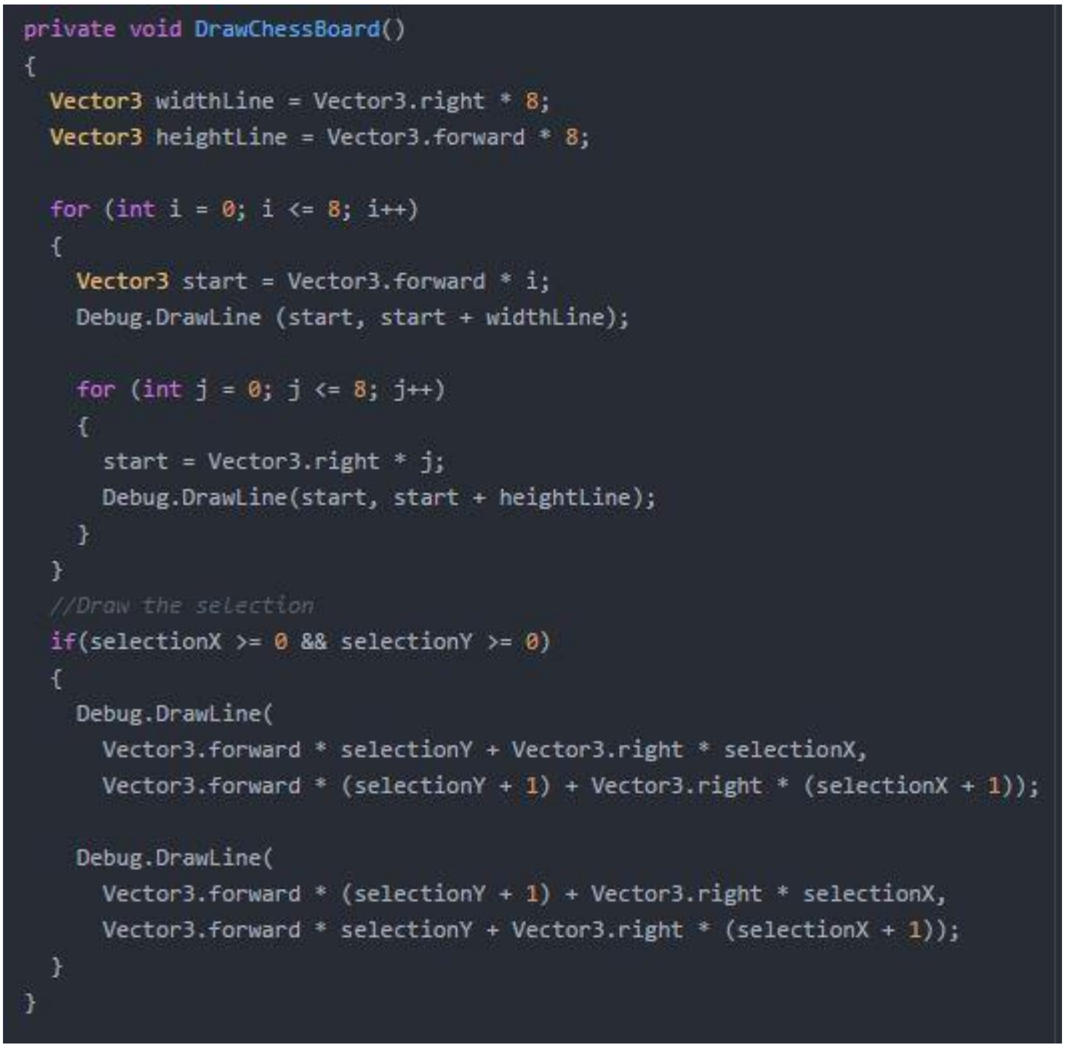


```
//Span the white team

//King
SpawnChessman (0,3,0);

//Queen
SpawnChessman (1,4,0);

//Rook
SpawnChessman (2,0,0);
SpawnChessman (2,7,0);

//Bishops
SpawnChessman (3,2,0);
SpawnChessman (3,5,0);

//Kight
SpawnChessman (4,1,0);
SpawnChessman (4,6,0);

//Pawns
for (int i = 0; i < 8; i ++)
  SpawnChessman(5,i,1);
```

Figure 2: Spawn Black Pawns

4.2 Chessboard

In this piece of code the chessboard is created and is the foundation of which this game is going to be based off. The intersecting lines that make up the chessboard are being created here, they are spanning 9 lines in the x axis and 9 lines in the y axis.



```
private void DrawChessBoard()
{
  Vector3 widthLine = Vector3.right * 8;
  Vector3 heightLine = Vector3.forward * 8;

  for (int i = 0; i <= 8; i++)
  {
    Vector3 start = Vector3.forward * i;
    Debug.DrawLine (start, start + widthLine);

    for (int j = 0; j <= 8; j++)
    {
      start = Vector3.right * j;
      Debug.DrawLine(start, start + heightLine);
    }
  }
  //Draw the selection
  if(selectionX >= 0 && selectionY >= 0)
  {
    Debug.DrawLine(
      Vector3.forward * selectionY + Vector3.right * selectionX,
      Vector3.forward * (selectionY + 1) + Vector3.right * (selectionX + 1));

    Debug.DrawLine(
      Vector3.forward * (selectionY + 1) + Vector3.right * selectionX,
      Vector3.forward * selectionY + Vector3.right * (selectionX + 1));
  }
}
```

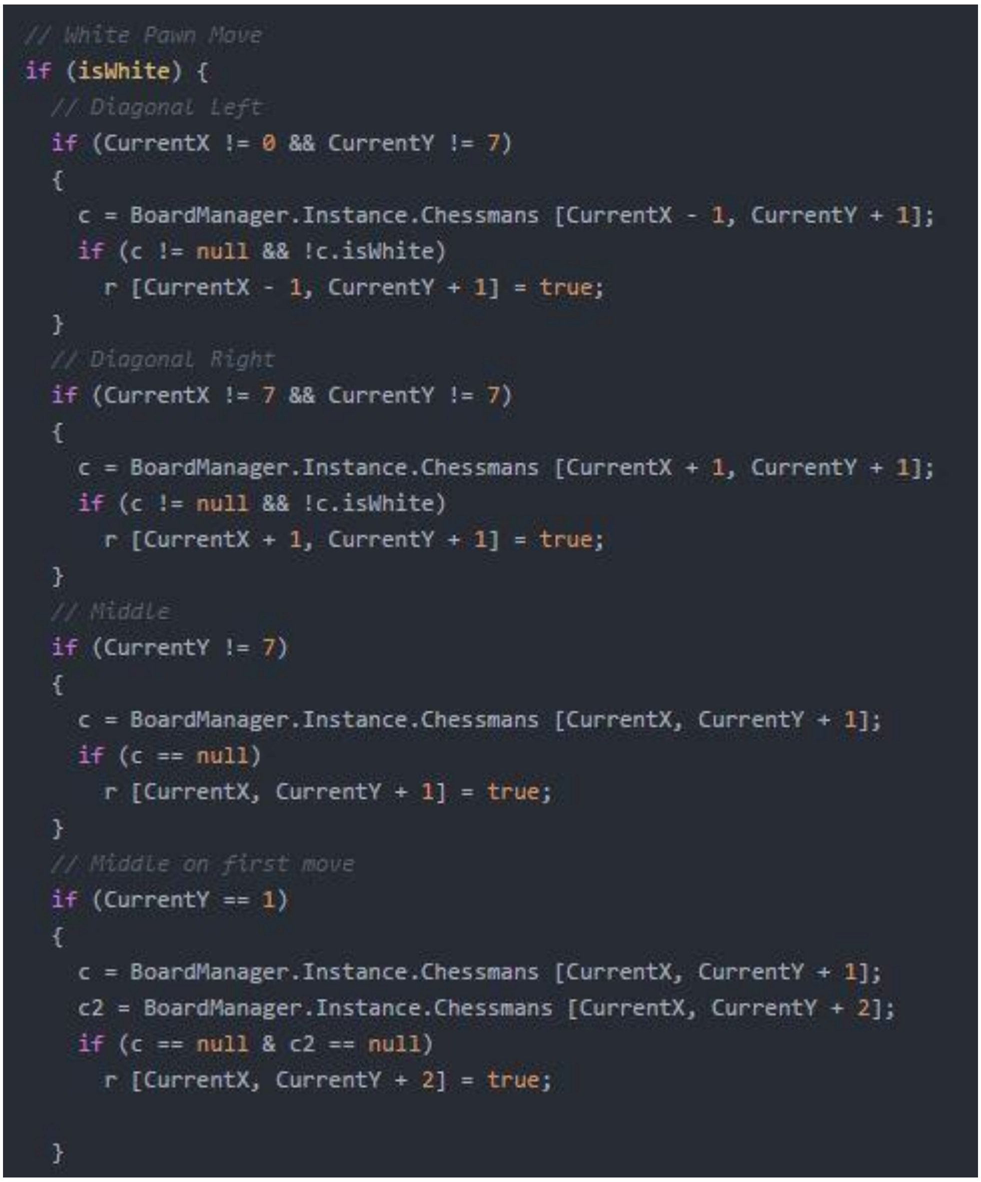
Figure 3: Draw Chessboard Code

4.3 Pawn Movement

“Diagonal left” this starts by checking to see if the piece is on the first tile or if it is at the end of the board. It then sets the axis so that it can move -1 in the x and +1 in the y. finally it checks to see if the piece in that position is not a white piece. If it is not a white piece it will take that tile.

“Middle” this will check if the piece has reached the end of the board, if no it will allow the white piece to move +1 in the y axis.

“Middle on first move” here we are checking if the piece is sitting on the number 1 tile on the y axis. If there is nothing on the tile +1 and +2 in the y axis it will let this piece move 1 or 2 tiles.



```
// White Pawn Move
if (isWhite) {
  // Diagonal Left
  if (CurrentX != 0 && CurrentY != 7)
  {
    c = BoardManager.Instance.Chessmans [CurrentX - 1, CurrentY + 1];
    if (c != null && !c.isWhite)
      r [CurrentX - 1, CurrentY + 1] = true;
  }
  // Diagonal Right
  if (CurrentX != 7 && CurrentY != 7)
  {
    c = BoardManager.Instance.Chessmans [CurrentX + 1, CurrentY + 1];
    if (c != null && !c.isWhite)
      r [CurrentX + 1, CurrentY + 1] = true;
  }
  // Middle
  if (CurrentY != 7)
  {
    c = BoardManager.Instance.Chessmans [CurrentX, CurrentY + 1];
    if (c == null)
      r [CurrentX, CurrentY + 1] = true;
  }
  // Middle on first move
  if (CurrentY == 1)
  {
    c = BoardManager.Instance.Chessmans [CurrentX, CurrentY + 1];
    c2 = BoardManager.Instance.Chessmans [CurrentX, CurrentY + 2];
    if (c == null & c2 == null)
      r [CurrentX, CurrentY + 2] = true;

  }
```

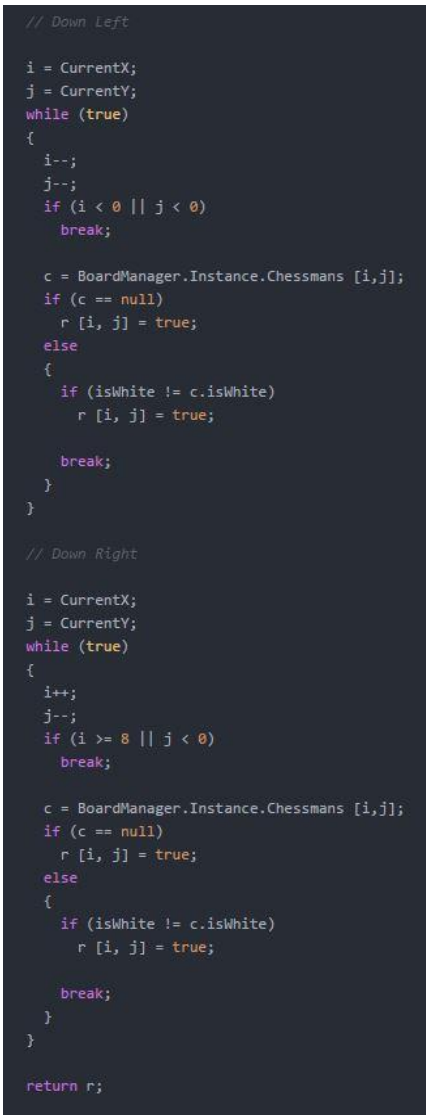
Figure 4: White Pawn Movement

4.4 Bishop Movement

Like the Rook the Bishop moves in four direction until it reaches another piece of the end of the board.

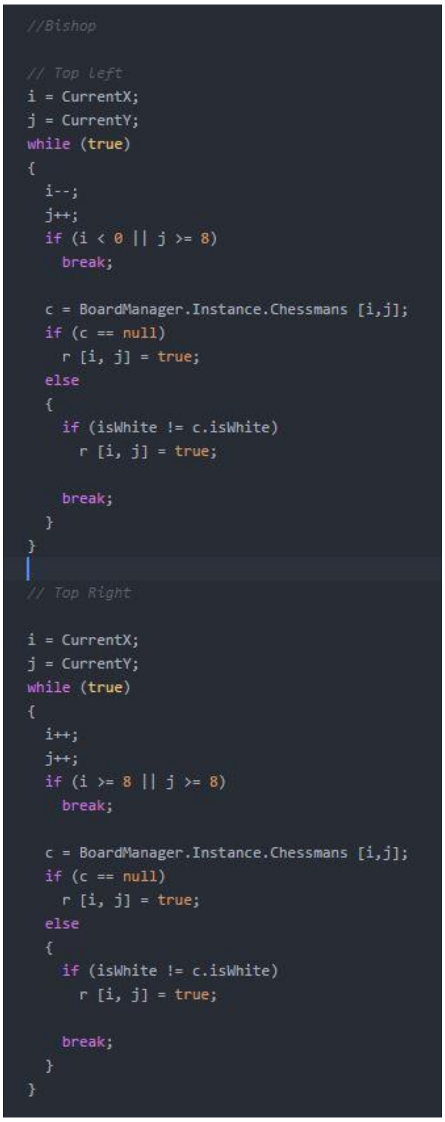
“Top Left” because we are moving the piece in the top left direction I needed to set the x axis to – and the y axis to ++, this will carry the piece left in the x direction and up in the y direction.

I then ran a loop to check if the piece was still on the board, by checking if the x axis is less than 0 or the y axis is greater or equal to 8. If this is the case we break, the next lines of code are checking to see if that direction is clear or if there is a not white piece that can be taken. This process if then repeated for the other direction this piece can be moved.



```
// Down Left

i = CurrentX;
j = CurrentY;
while (true)
{
  i--;
  j--;
  if (i < 0 || j < 0)
    break;

  c = BoardManager.Instance.Chessmans [i,j];
  if (c == null)
    r [i, j] = true;
  else
  {
    if (isWhite != c.isWhite)
      r [i, j] = true;

    break;
  }
}

// Down Right

i = CurrentX;
j = CurrentY;
while (true)
{
  i++;
  j--;
  if (i >= 8 || j < 0)
    break;

  c = BoardManager.Instance.Chessmans [i,j];
  if (c == null)
    r [i, j] = true;
  else
  {
    if (isWhite != c.isWhite)
      r [i, j] = true;

    break;
  }
}

return r;
```

Figure 5: Bishop Movement Code 2



```
//Bishop

// Top Left
i = CurrentX;
j = CurrentY;
while (true)
{
  i--;
  j++;
  if (i < 0 || j >= 8)
    break;

  c = BoardManager.Instance.Chessmans [i,j];
  if (c == null)
    r [i, j] = true;
  else
  {
    if (isWhite != c.isWhite)
      r [i, j] = true;

    break;
  }
}

// Top Right

i = CurrentX;
j = CurrentY;
while (true)
{
  i++;
  j++;
  if (i >= 8 || j >= 8)
    break;

  c = BoardManager.Instance.Chessmans [i,j];
  if (c == null)
    r [i, j] = true;
  else
  {
    if (isWhite != c.isWhite)
      r [i, j] = true;

    break;
  }
}
```

Figure 6: Bishop Movement Code

4.5 Knight Movement

“Up Left” here we are moving the piece down 1 in the x axis and up 2 in the y axis.

“Up Right” her we are moving the pieces up 1 in the x axis and up 2 in the y axis.

“Right Up” her we are moving the pieces up 2 in the x axis and up 1 in the y axis.

“Right Down” her we are moving the pieces up 2 in the x axis and down 1 in the y axis.

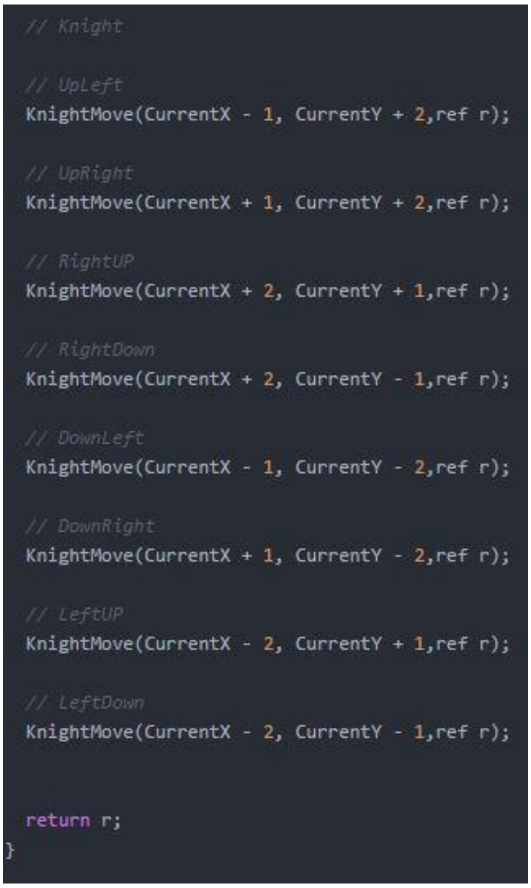
“Down Left” her we are moving the pieces down 1 in the x axis and down 2 in the y axis.

“Down Right” her we are moving the pieces up 1 in the x axis and down 2 in the y axis.

“Left Up” her we are moving the pieces down 2 in the x axis and up 1 in the y axis.

“Left Down” her we are moving the pieces down 2 in the x axis and down 1 in the y axis.

The direction that this code sent the piece in can be seen as an L shape. The piece is always moving 2 on one axis and 1 on another.



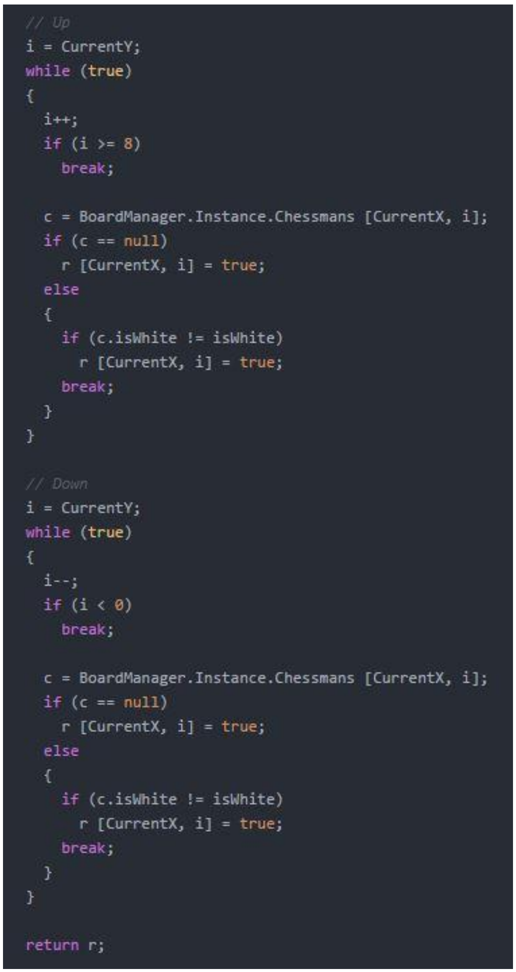
```
  // Knight

  // UpLeft
  KnightMove(CurrentX - 1, CurrentY + 2,ref r);

  // UpRight
  KnightMove(CurrentX + 1, CurrentY + 2,ref r);

  // RightUP
  KnightMove(CurrentX + 2, CurrentY + 1,ref r);

  // RightDown
  KnightMove(CurrentX + 2, CurrentY - 1,ref r);

  // DownLeft
  KnightMove(CurrentX - 1, CurrentY - 2,ref r);

  // DownRight
  KnightMove(CurrentX + 1, CurrentY - 2,ref r);

  // LeftUP
  KnightMove(CurrentX - 2, CurrentY + 1,ref r);

  // LeftDown
  KnightMove(CurrentX - 2, CurrentY - 1,ref r);


  return r;
}
```

Figure 7: Knight Movement Code

4.6 Rook Movement

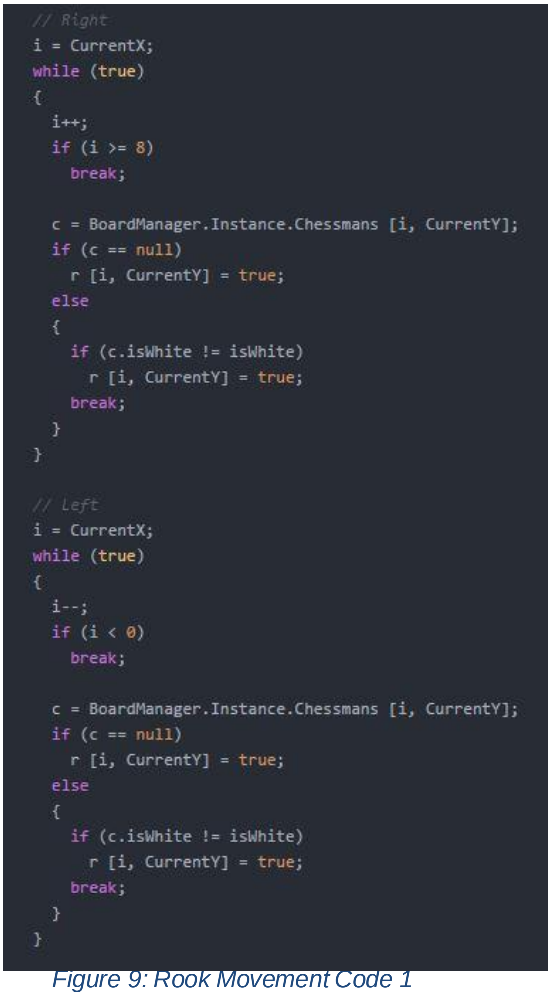
“Right” in this I am checking to see if the x axis is greater than or equal to 8, if it is it means that we went too far on the board so we break the while and exit the loop. If the direction is less than 8 then this piece will be allowed travel in the right direction so long as there is not a piece in the way, with that we are checking for a not white piece that can be taken.

This is repeated for each of the direction, we are just moving on a different axis in each one.



```
// Up
i = CurrentY;
while (true)
{
  i++;
  if (i >= 8)
    break;

  c = BoardManager.Instance.Chessmans [CurrentX, i];
  if (c == null)
    r [CurrentX, i] = true;
  else
  {
    if (c.isWhite != isWhite)
      r [CurrentX, i] = true;
    break;
  }
}

// Down
i = CurrentY;
while (true)
{
  i--;
  if (i < 0)
    break;

  c = BoardManager.Instance.Chessmans [CurrentX, i];
  if (c == null)
    r [CurrentX, i] = true;
  else
  {
    if (c.isWhite != isWhite)
      r [CurrentX, i] = true;
    break;
  }
}

return r;
```

Figure 8: Rook Movement Code 2

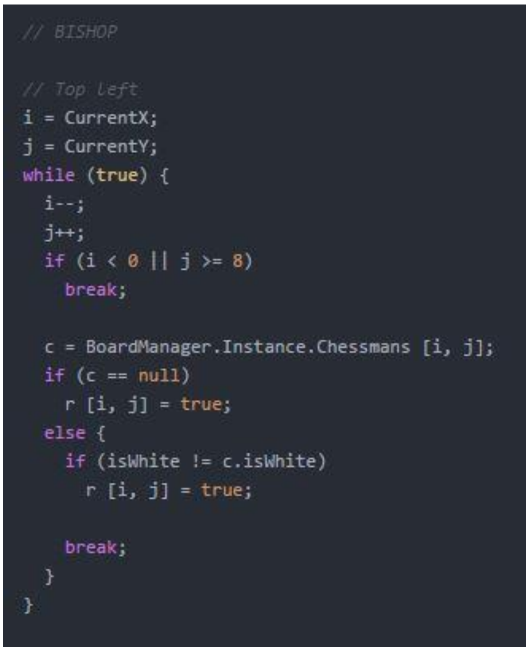


```
// Right
i = CurrentX;
while (true)
{
  i++;
  if (i >= 8)
    break;

  c = BoardManager.Instance.Chessmans [i, CurrentY];
  if (c == null)
    r [i, CurrentY] = true;
  else
  {
    if (c.isWhite != isWhite)
      r [i, CurrentY] = true;
    break;
  }
}

// Left
i = CurrentX;
while (true)
{
  i--;
  if (i < 0)
    break;

  c = BoardManager.Instance.Chessmans [i, CurrentY];
  if (c == null)
    r [i, CurrentY] = true;
  else
  {
    if (c.isWhite != isWhite)
      r [i, CurrentY] = true;
    break;
  }
}
```

Figure 9: Rook Movement Code 1

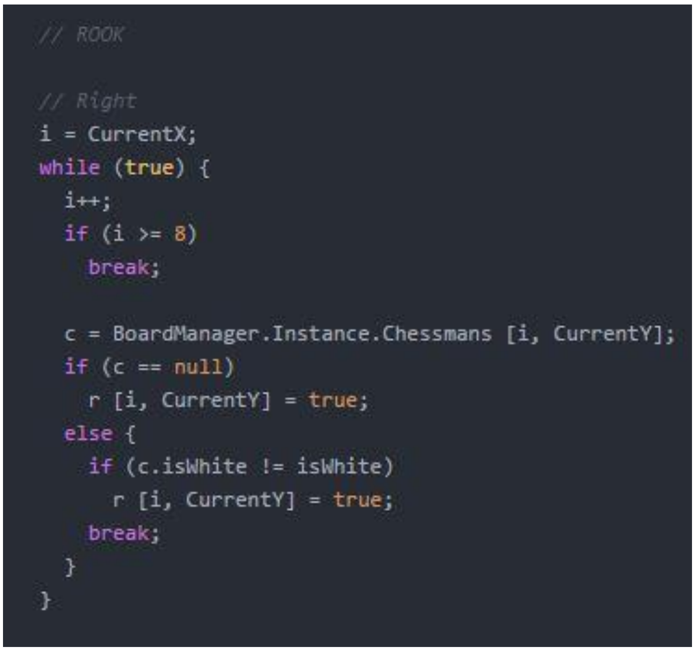
4.7 Queen Movement

Once the Rook and the Bishop pieces were moving the way they are meant to be, the Queen was easy to do. The Queen can move in many direction on the board so, it was simply a matter of implementing the code from the Bishop and the code from the Rook in to the Queen and from there the Queens movement was functioning perfectly.



```
// BISHOP

// Top Left
i = CurrentX;
j = CurrentY;
while (true) {
  i--;
  j++;
  if (i < 0 || j >= 8)
    break;

  c = BoardManager.Instance.Chessmans [i, j];
  if (c == null)
    r [i, j] = true;
  else {
    if (isWhite != c.isWhite)
      r [i, j] = true;

    break;
  }
}
```

Figure 10: Queen Movement Code 2



```
// ROOK

// Right
i = CurrentX;
while (true) {
  i++;
  if (i >= 8)
    break;

  c = BoardManager.Instance.Chessmans [i, CurrentY];
  if (c == null)
    r [i, CurrentY] = true;
  else {
    if (c.isWhite != isWhite)
      r [i, CurrentY] = true;
    break;
  }
}
```

Figure 11: Queen Movement Code 1

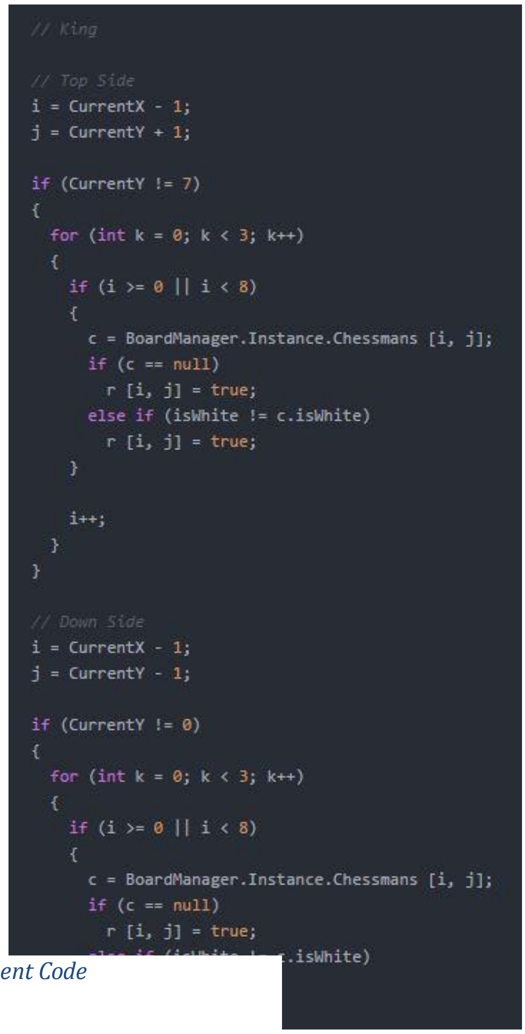
4.8 King Movement

The Kings movement is relatively straight forward, in order for the King to move in the direction it is designed for, it must be able to move 1 tile in each direction.

“Top Side” here the x axis has been set to – 1 and the y axis has been set to + 1. Similar to the other pieces we are first checking to see if the current position is equal to 7, if it is them we are on the last tile on the board and will not be able to go any further in that direction. A loop is then run to check if the movement of the King is within the boundaries of the board.

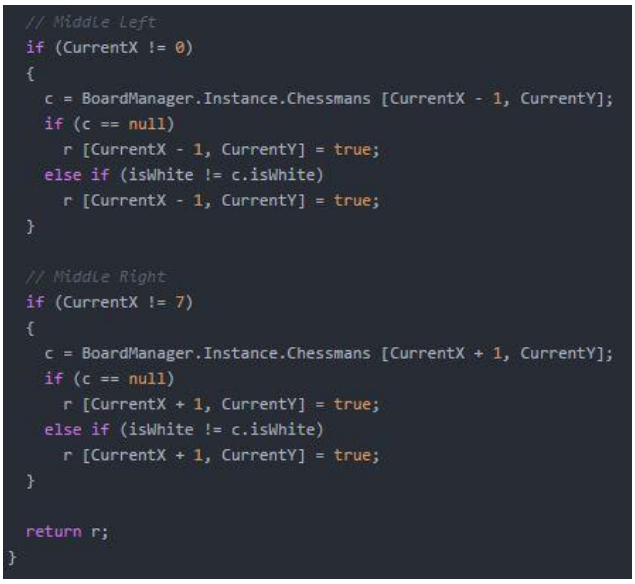
If the King is within the boundaries of the board and there nothing blocking its path it can move, however, if there is a not white piece it can take that tile.

This process is repeated for each of the directions of the Kings movement.



```
// King

// Top Side
i = CurrentX - 1;
j = CurrentY + 1;

if (CurrentY != 7)
{
  for (int k = 0; k < 3; k++)
  {
    if (i >= 0 || i < 8)
    {
      c = BoardManager.Instance.Chessmans [i, j];
      if (c == null)
        r [i, j] = true;
      else if (isWhite != c.isWhite)
        r [i, j] = true;
    }

    i++;
  }
}

// Down Side
i = CurrentX - 1;
j = CurrentY - 1;

if (CurrentY != 0)
{
  for (int k = 0; k < 3; k++)
  {
    if (i >= 0 || i < 8)
    {
      c = BoardManager.Instance.Chessmans [i, j];
      if (c == null)
        r [i, j] = true;
      else if (isWhite != c.isWhite)
```

Figure 13: King Movement Code

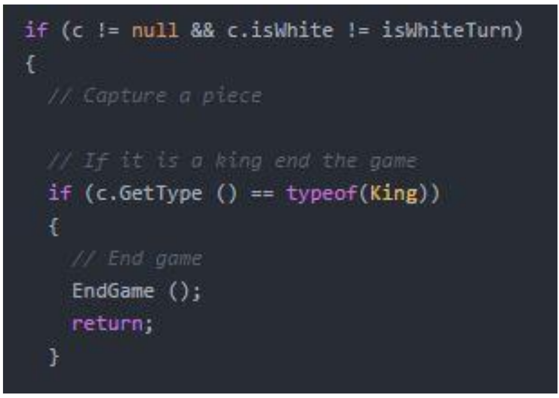


```
  // Middle Left
  if (CurrentX != 0)
  {
    c = BoardManager.Instance.Chessmans [CurrentX - 1, CurrentY];
    if (c == null)
      r [CurrentX - 1, CurrentY] = true;
    else if (isWhite != c.isWhite)
      r [CurrentX - 1, CurrentY] = true;
  }

  // Middle Right
  if (CurrentX != 7)
  {
    c = BoardManager.Instance.Chessmans [CurrentX + 1, CurrentY];
    if (c == null)
      r [CurrentX + 1, CurrentY] = true;
    else if (isWhite != c.isWhite)
      r [CurrentX + 1, CurrentY] = true;
  }

  return r;
}
```

Figure 12: King Movement Code 2

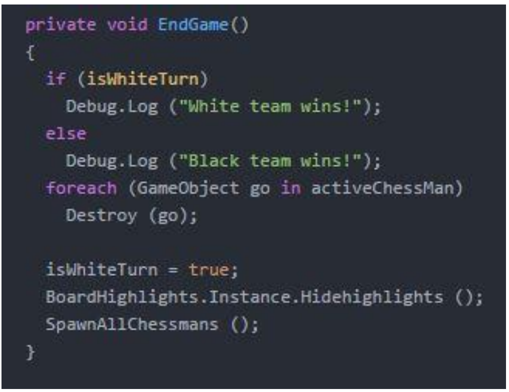
4.10 End Game

“isWhiteTurn” here the code is checking to see if it is the white turn when the king is taken, if it is the message printed will be “White team wins!”. Else if this is not the case then it will print “Black team wins!”. After this the “foreach” wipes the chess board and sets all the pieces back to their original position.



```
if (c != null && c.isWhite != isWhiteTurn)
{
  // Capture a piece

  // If it is a king end the game
  if (c.GetType () == typeof(King))
  {
    // End game
    EndGame ();
    return;
  }
```

Figure 14: End Game Code 1



```
private void EndGame()
{
  if (isWhiteTurn)
    Debug.Log ("White team wins!");
  else
    Debug.Log ("Black team wins!");
  foreach (GameObject go in activeChessMan)
    Destroy (go);

  isWhiteTurn = true;
  BoardHighlights.Instance.Hidehighlights ();
  SpawnAllChessmans ();
}
```

Figure 15: End Game Code 2

5 Testing

With testing being one of the crucial things you can do when developing a game, it was important for me to do it to the best of my abilities. Testing will reveal information to me that I may not have notice, also it could help in making the game an overall better experience for the users of the application. No matter what the feedback from users it will provide me with either the necessary information to make changes to the application or to make additions to the application.

Methods of testing that was used in Pawn of War:

- Customer Testing
- Usability Testing
- Unit & Integration Testing

Each of these tests are further explained on the heading below.

5.1 Usability Testing

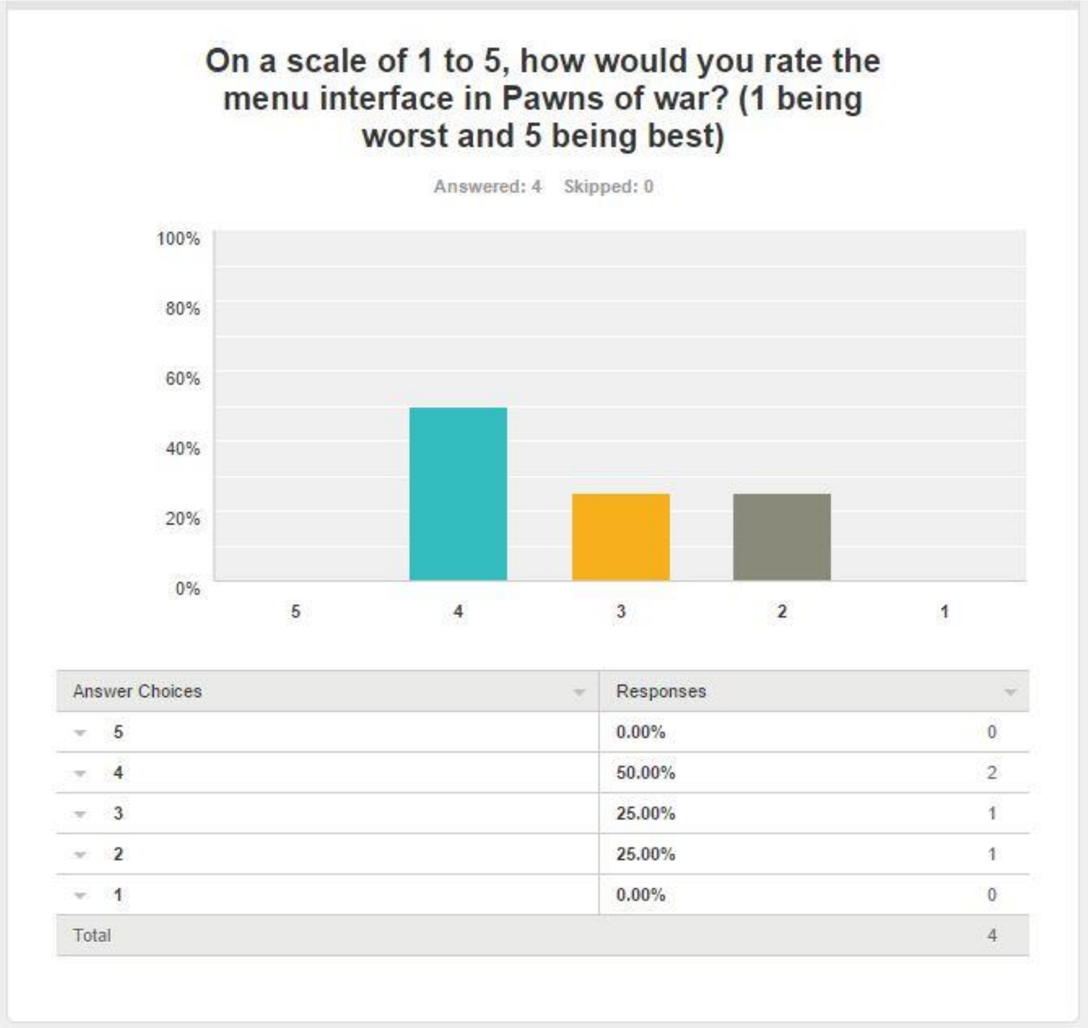
The usability study that was designed for Pawns of War was done so to help identify any interface and control problems that may have been lingering in the gameplay. This testing will help make the game more user friendly and weave out any issues in the controls.

To test the usability of this application I used an online survey to assist me in finding any interface or control issues.

Usability Survey:

All the questions in this survey are either based on the design of the application or the features and functionality of the application. Of the 10 people that surveys had been sent out, only 4 got back to me in time. The following is the answers to the surveys along with the additional comments I received via email related to specific questions.

***Question One**: On a scale of 1 to 5, how would you rate the menu interface in Pawns of war? (1 being worst and 5 being best)*



On a scale of 1 to 5, how would you rate the menu interface in Pawns of war? (1 being worst and 5 being best)

Answered: 4 Skipped: 0

Answer Choices	Responses	
5	0.00%	0
4	50.00%	2
3	25.00%	1
2	25.00%	1
1	0.00%	0
Total		4

Figure 16: Usability Survey Question 1

From this question two people emailed me with further information on this the following is sections taken directly from their message:

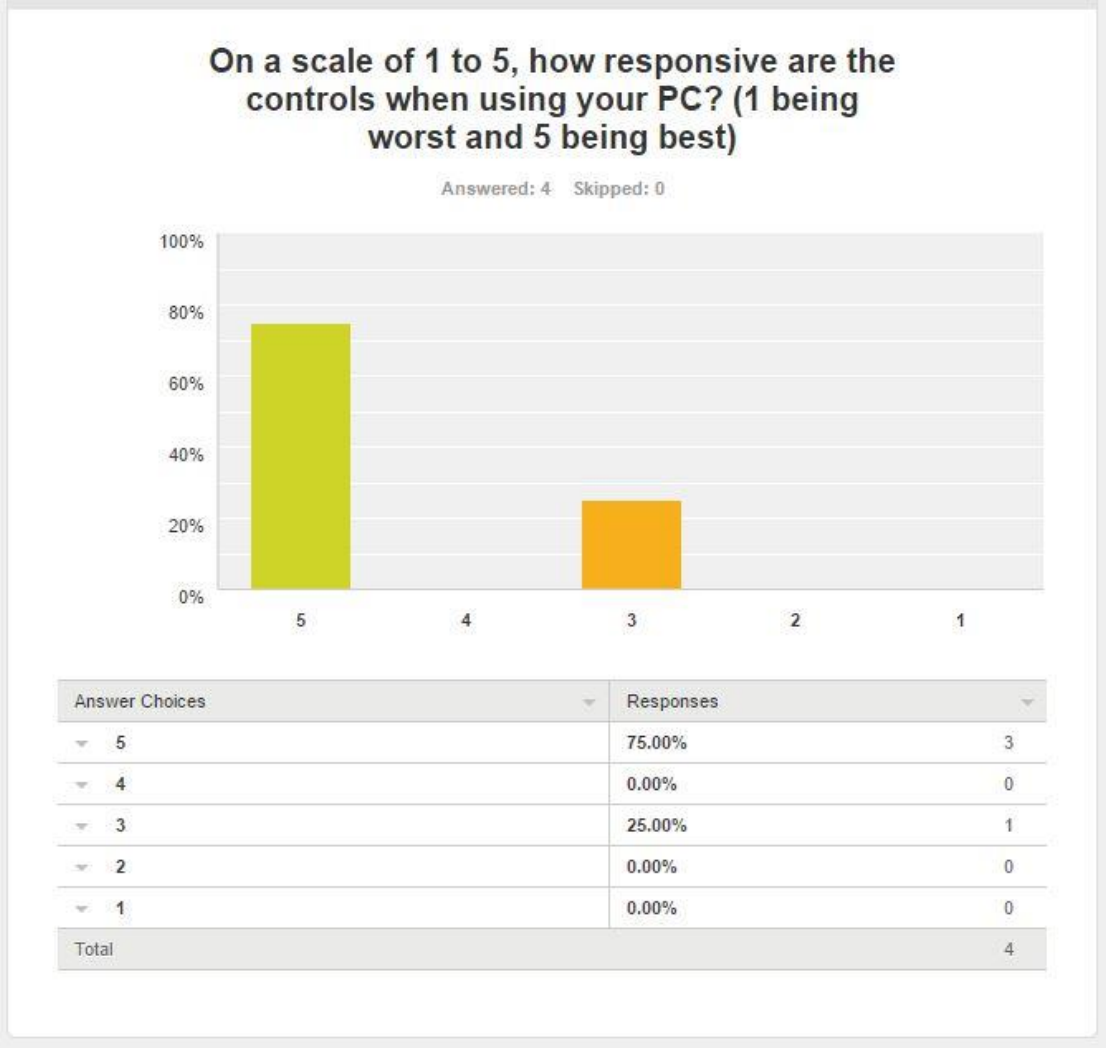
- Person one: *"I did like the interface but it just felt a little bare, like it was missing something"*
- Person two: *"Pretty cool, the background is awesome"*

Analysis & Solution:

This question showed mixed reviews, which was something that I was expecting. The interface of the menu needs working on and this was to the best way to fully cement that.

I will draw up other variations of this and see what the response is on them.

Question Two: *On a scale of 1 to 5, how responsive are the controls when using your PC? (1 being worst and 5 being best)*



On a scale of 1 to 5, how responsive are the controls when using your PC? (1 being worst and 5 being best)

Answered: 4 Skipped: 0

Answer Choices	Responses	
5	75.00%	3
4	0.00%	0
3	25.00%	1
2	0.00%	0
1	0.00%	0
Total		4

Figure 17: Usability Survey Question 2

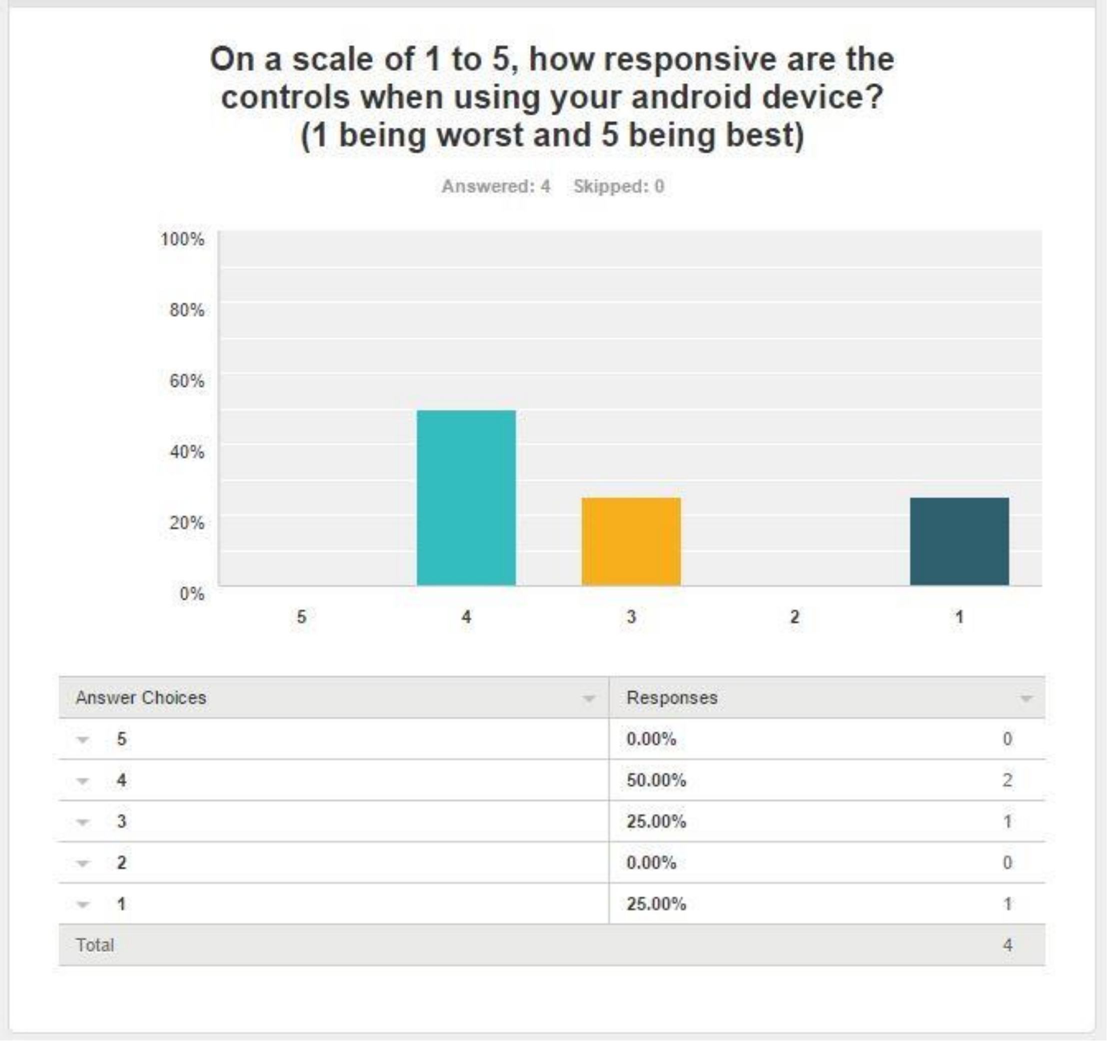
There were no additional comment for this question.

Analysis & solution:

I expected this to get a good response, considering the game simply relies on one click to move the pieces.

To better this I will just have to make something that I a little more interactive so you are not just clicking a mouse.

Question Three: *On a scale of 1 to 5, how responsive are the controls when using your android device? (1 being worst and 5 being best)*



On a scale of 1 to 5, how responsive are the controls when using your android device? (1 being worst and 5 being best)

Answered: 4 Skipped: 0

Answer Choices	Responses	
5	0.00%	0
4	50.00%	2
3	25.00%	1
2	0.00%	0
1	25.00%	1
Total		4

Figure 18: Usability Survey Question 3

There were one additional response to this question:

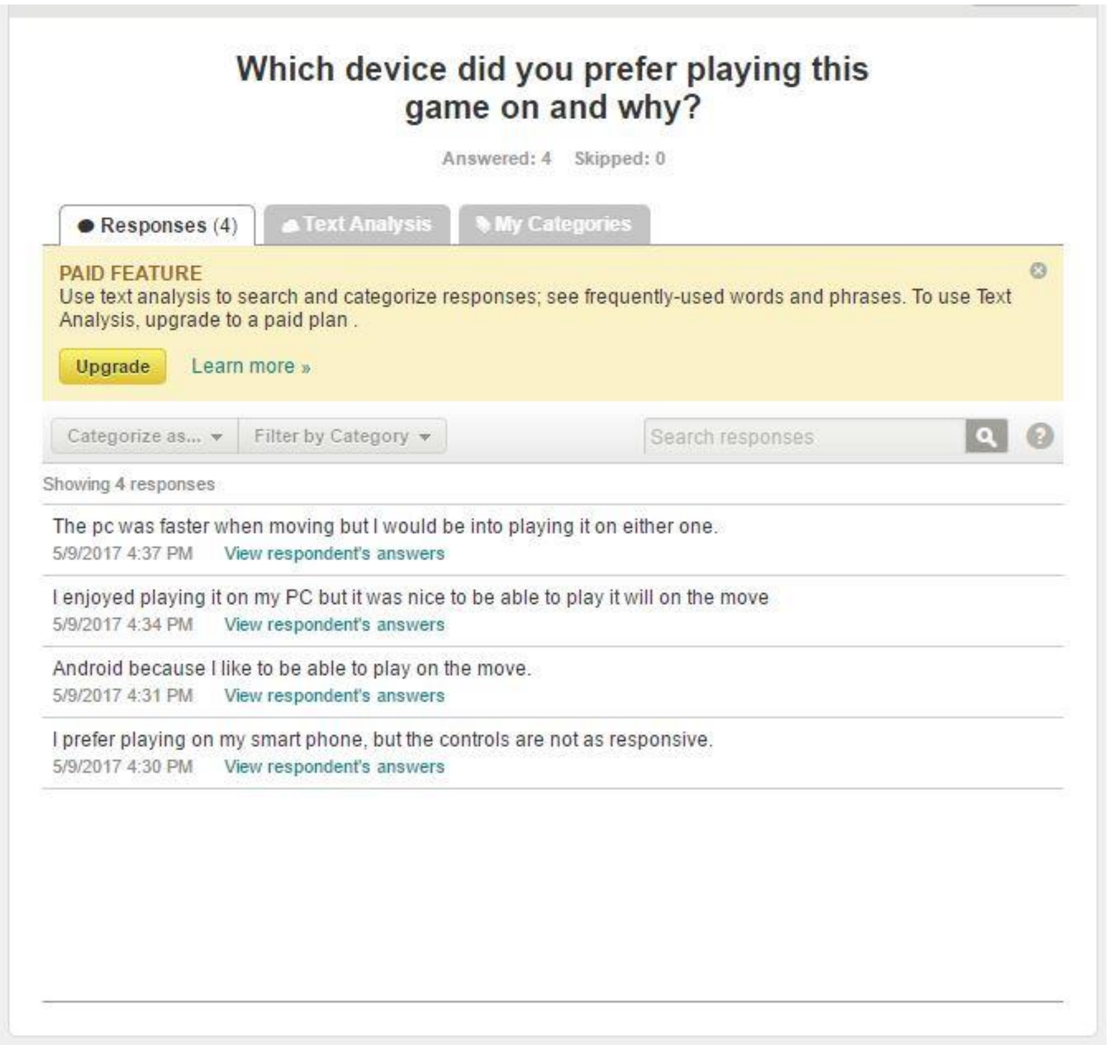
- "*Whenever I pressed the square it was selecting a different one and there was a lot of lag after pressing the square*"

Analysis & solution:

I may have made the tiles a little big small for using on an android phone, others must have encountered lag also, I will have to go back into the game and see how this can be resolved.

For the tiles I think if I made the camera closer to the board for android it will take away any issues with pressing the wrong thing by mistake.

Question Four: *Which device did you prefer playing this game on and why?*



Which device did you prefer playing this game on and why?

Answered: 4 Skipped: 0

PAID FEATURE
Use text analysis to search and categorize responses; see frequently-used words and phrases. To use Text Analysis, upgrade to a paid plan .

Showing 4 responses

The pc was faster when moving but I would be into playing it on either one.
5/9/2017 4:37 PM View respondent's answers

I enjoyed playing it on my PC but it was nice to be able to play it will on the move
5/9/2017 4:34 PM View respondent's answers

Android because I like to be able to play on the move.
5/9/2017 4:31 PM View respondent's answers

I prefer playing on my smart phone, but the controls are not as responsive.
5/9/2017 4:30 PM View respondent's answers

Figure 19: Usability Survey Question 4

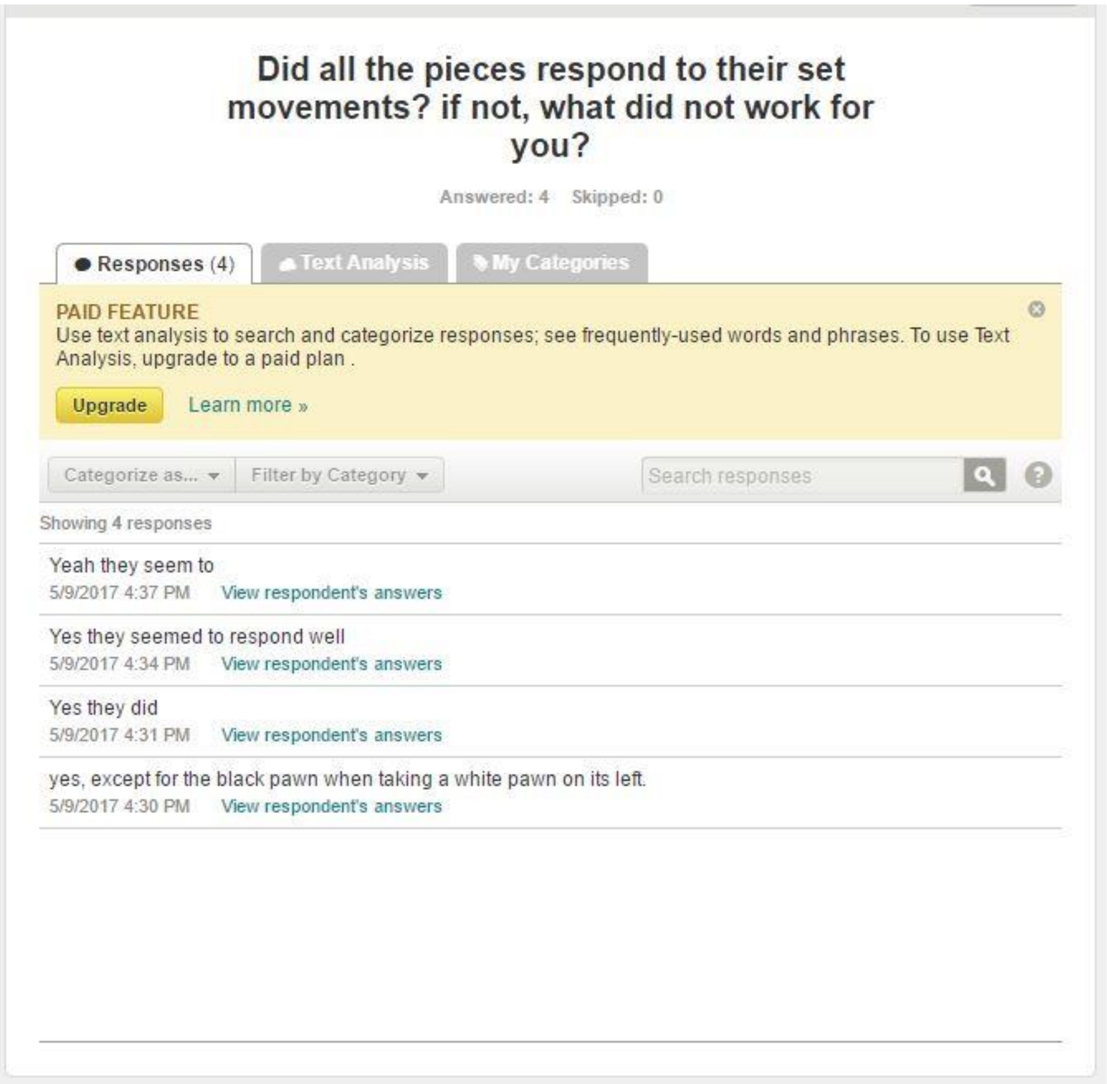
There were no additional comment on this question.

Analysis & solution:

This question seemed to get a 50/50 response. I think for people that are on the go, it makes sense that playing this game on the go is more optimal for them.

I think the best way to improve this is to improve the control responsiveness for the android.

Question Five: *Did all the pieces respond to their set movements? if not, what did not work for you?*



Did all the pieces respond to their set movements? if not, what did not work for you?

Answered: 4 Skipped: 0

PAID FEATURE
Use text analysis to search and categorize responses; see frequently-used words and phrases. To use Text Analysis, upgrade to a paid plan .

Showing 4 responses

Yeah they seem to
5/9/2017 4:37 PM View respondent's answers

Yes they seemed to respond well
5/9/2017 4:34 PM View respondent's answers

Yes they did
5/9/2017 4:31 PM View respondent's answers

yes, except for the black pawn when taking a white pawn on its left.
5/9/2017 4:30 PM View respondent's answers

Figure 20: Usability Survey Question 5

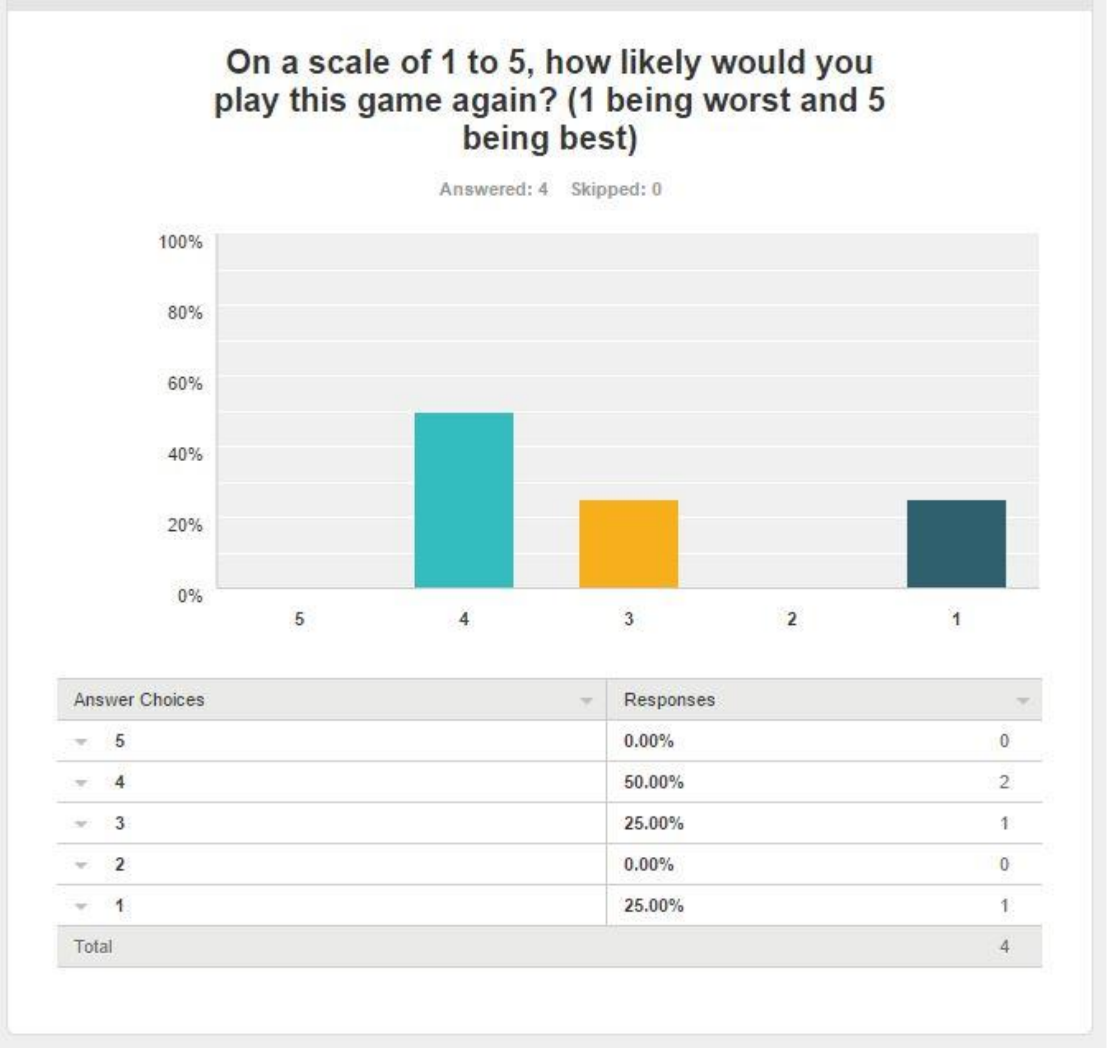
There were no additional comments on this question.

Analysis & solution:

The responses to this question seemed to be straight forward, as expected. The pieces move the way in which they are intended, however, someone did spot a problem that I had not.

The black piece was not taking white pawns that were to the left of it, this was a simple fix in the code, I just needed to add a +1 in the x axis on the pawns movement.

Question Six: *On a scale of 1 to 5, how likely would you play this game again? (1 being worst and 5 being best)*



On a scale of 1 to 5, how likely would you play this game again? (1 being worst and 5 being best)

Answered: 4 Skipped: 0

Answer Choices	Responses	
5	0.00%	0
4	50.00%	2
3	25.00%	1
2	0.00%	0
1	25.00%	1
Total		4

Figure 21: Usability Survey Question 6

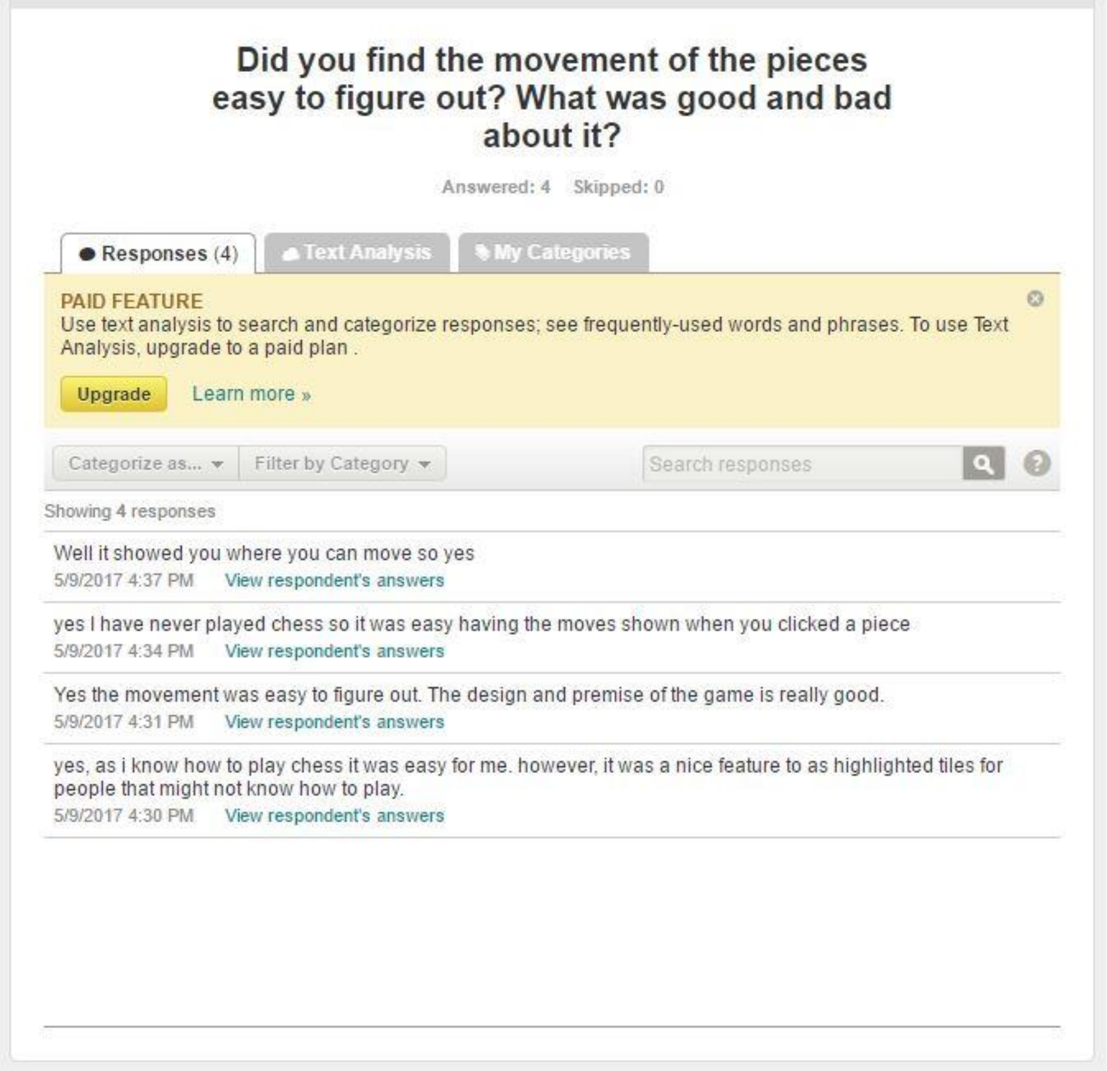
There were no addition comments on this question.

Analysis & solution:

From this question, I gather that the game did not offer enough to the users, they seem as if they had of wanted more from the game. That or they do just not like strategy games

I feel like with extra time to work on the project I will be able to create something that will attract a higher majority of people.

Question Seven: *Did you find the movement of the pieces easy to figure out? What was good and bad about it?*



Did you find the movement of the pieces easy to figure out? What was good and bad about it?

Answered: 4 Skipped: 0

PAID FEATURE
Use text analysis to search and categorize responses; see frequently-used words and phrases. To use Text Analysis, upgrade to a paid plan .

Showing 4 responses

Well it showed you where you can move so yes
5/9/2017 4:37 PM View respondent's answers

yes I have never played chess so it was easy having the moves shown when you clicked a piece
5/9/2017 4:34 PM View respondent's answers

Yes the movement was easy to figure out. The design and premise of the game is really good.
5/9/2017 4:31 PM View respondent's answers

yes, as i know how to play chess it was easy for me. however, it was a nice feature to as highlighted tiles for people that might not know how to play.
5/9/2017 4:30 PM View respondent's answers

Figure 22: Usability Survey Question 7

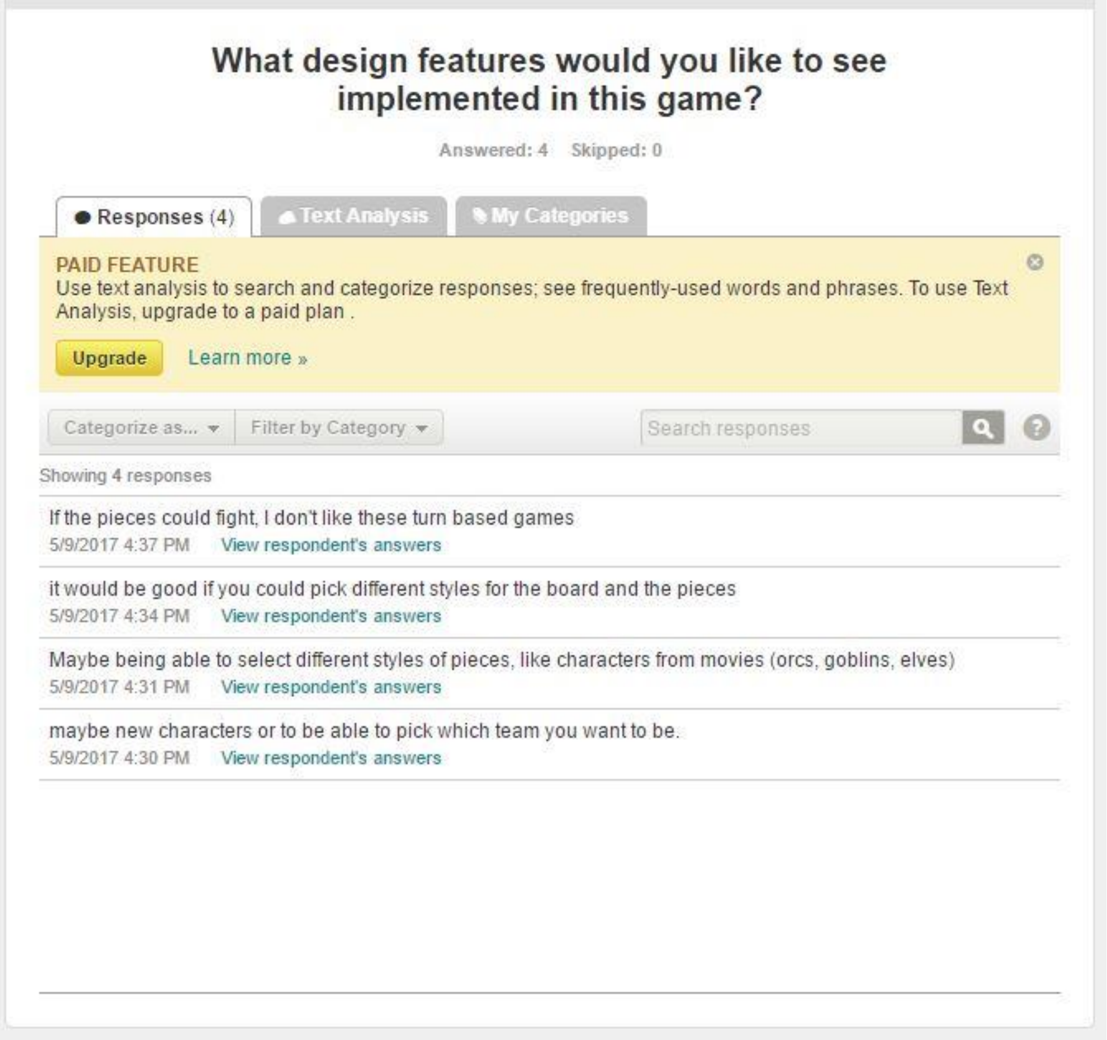
There were no additional comments with this question.

Analysis & solution:

I am glad that I added the highlighted tiles in this game, it would have been naïve of me assume that everyone knows how to play chess.

I would like to make the highlights for this game a little bit more less obvious, and to give people an option on whether they would like them on or not as they might be frustrating for some people.

Question Eight: *What design features would you like to see implemented in this game?*



What design features would you like to see implemented in this game?

Answered: 4 Skipped: 0

PAID FEATURE
Use text analysis to search and categorize responses; see frequently-used words and phrases. To use Text Analysis, upgrade to a paid plan .

Showing 4 responses

If the pieces could fight, I don't like these turn based games
5/9/2017 4:37 PM View respondent's answers

it would be good if you could pick different styles for the board and the pieces
5/9/2017 4:34 PM View respondent's answers

Maybe being able to select different styles of pieces, like characters from movies (orcs, goblins, elves)
5/9/2017 4:31 PM View respondent's answers

maybe new characters or to be able to pick which team you want to be.
5/9/2017 4:30 PM View respondent's answers

Figure 23: Usability Survey Question 8

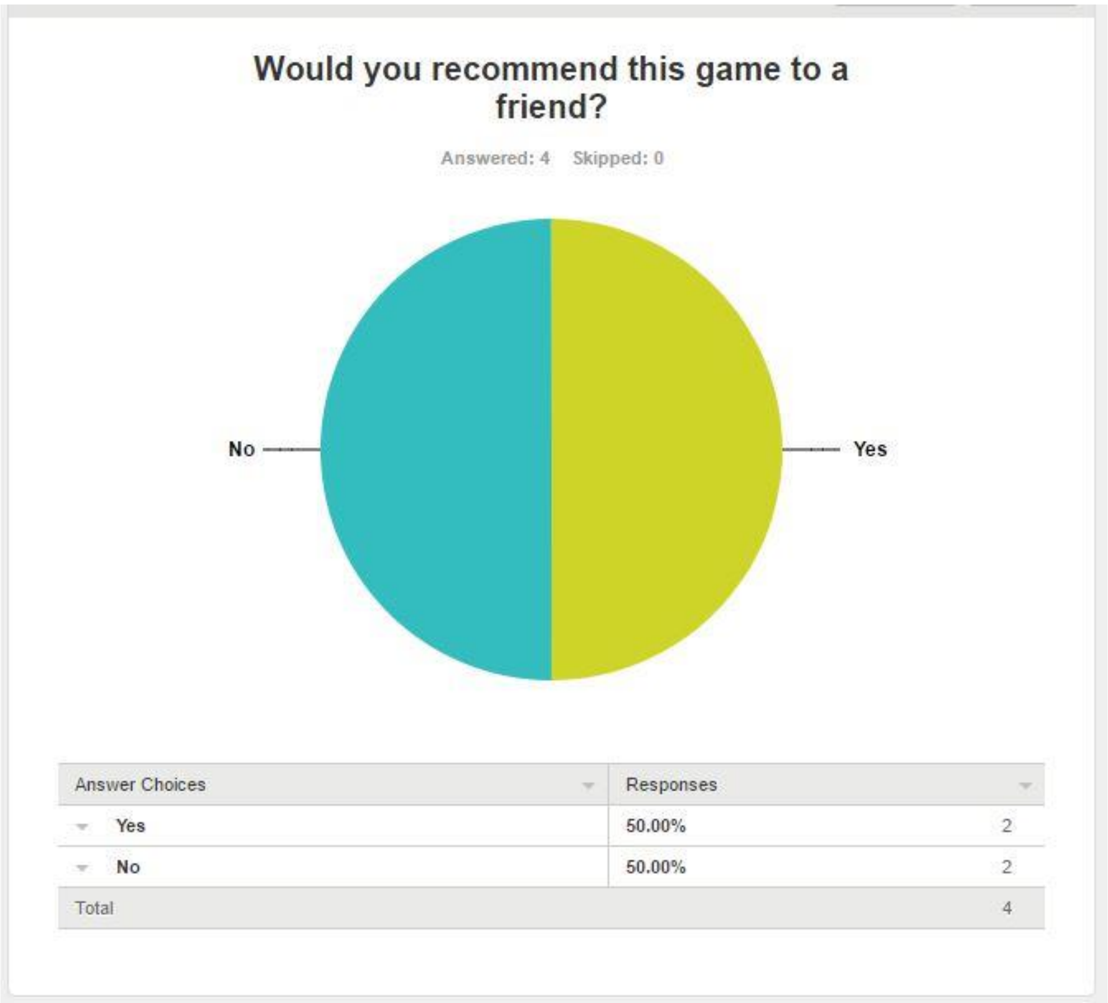
There were no additional comments on this question.

Analysis & solution:

There seems to be a recurring theme in these answers, and that seems to be variety. Something that everyone craves, I found this myself when developing the game, however, there was just not enough time to implement everything that I wanted to implement.

Over the course of this games development I will be creating additional DLC (downloadable content) that users can install on their device to have a wide variety of options to pick from.

Question Nine: *Would you recommend this game to a friend?*



Would you recommend this game to a friend?

Answered: 4 Skipped: 0

Answer Choices	Responses	
Yes	50.00%	2
No	50.00%	2
Total		4

Figure 24: Usability Survey Question 9

There were two comments that I got with this question.

- Person One: *“I think that there needs to be more to this game, I enjoyed it but there needs to be something added that has not been added to a game like this”*
- Person Two: *“Great game, everything worked perfect but it got old. There needs to be something that makes it different”*

Analysis & solution:

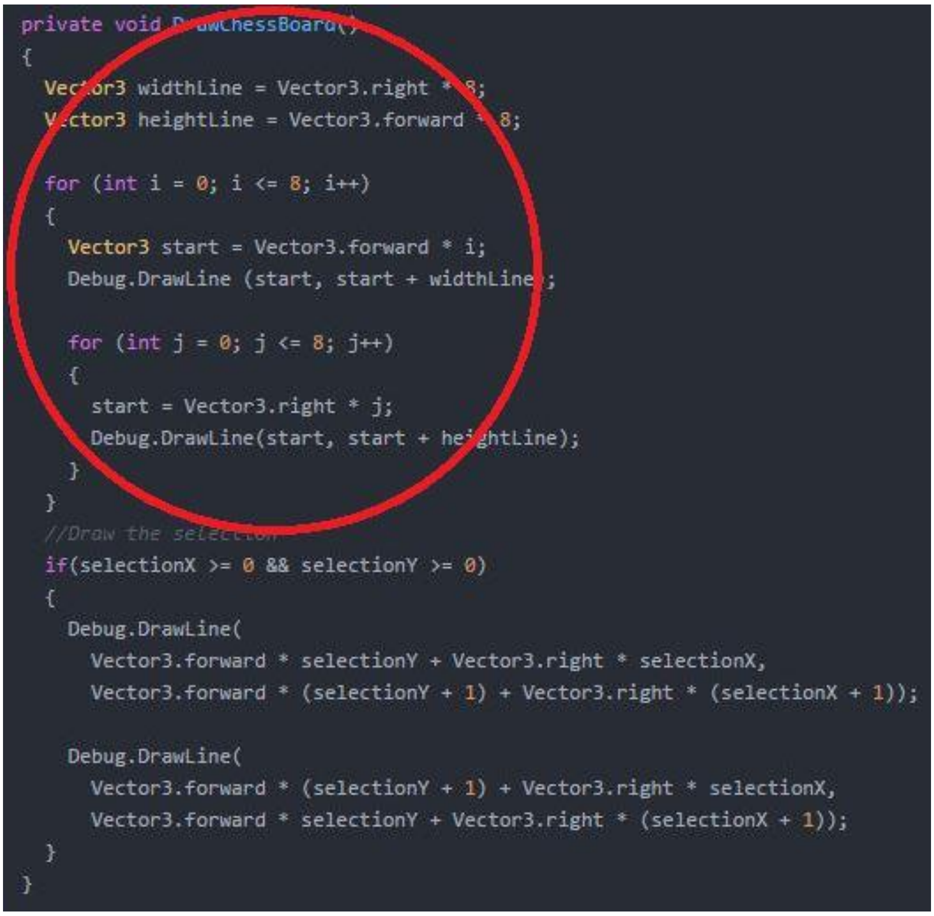
I completely understand why the game got this response, I will be creating more features for the game in the future and hopefully it will get a better response the next time round.

5.2 Unit & Integration Testing

For my project, there were not much on offer as far as unity and integration testing in unity, which is understandable as the only game object that remains in the screen is the chessboard object.

DrawChessBoard Test:

Everything in my project is created and brought to the game screen using code. The chessboard was laid out using code which is indicated below:



```
private void DrawChessBoard()
{
  Vector3 widthLine = Vector3.right * 8;
  Vector3 heightLine = Vector3.forward * 8;

  for (int i = 0; i <= 8; i++)
  {
    Vector3 start = Vector3.forward * i;
    Debug.DrawLine (start, start + widthLine);

    for (int j = 0; j <= 8; j++)
    {
      start = Vector3.right * j;
      Debug.DrawLine(start, start + heightLine);
    }
  }
  //Draw the selection
  if(selectionX >= 0 && selectionY >= 0)
  {
    Debug.DrawLine(
      Vector3.forward * selectionY + Vector3.right * selectionX,
      Vector3.forward * (selectionY + 1) + Vector3.right * (selectionX + 1));

    Debug.DrawLine(
      Vector3.forward * (selectionY + 1) + Vector3.right * selectionX,
      Vector3.forward * selectionY + Vector3.right * (selectionX + 1));
  }
}
```

Figure 25: Draw Chessboard Test

The section that is circled red is where the tiles for the chessboard is being created. Tests were run on this by running the game and if the lines appeared then this worked if not there was something wrong with the code.

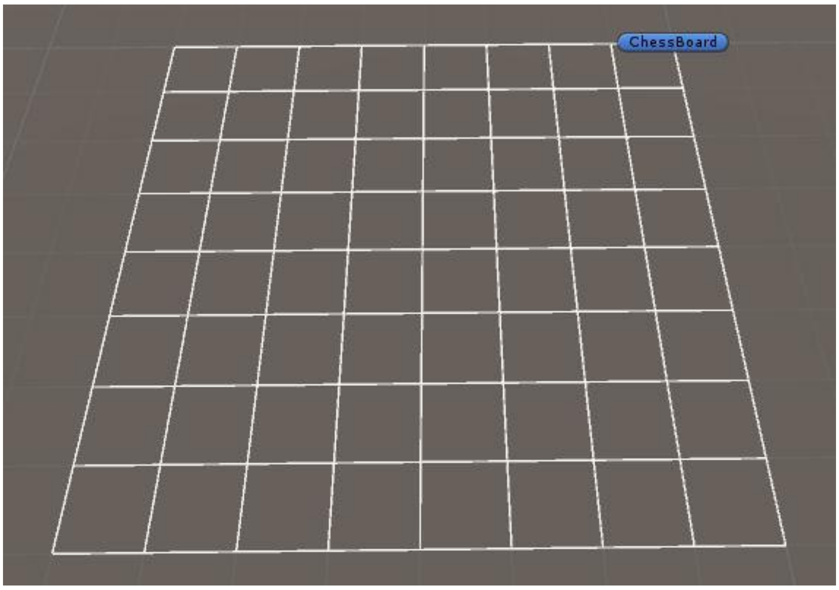
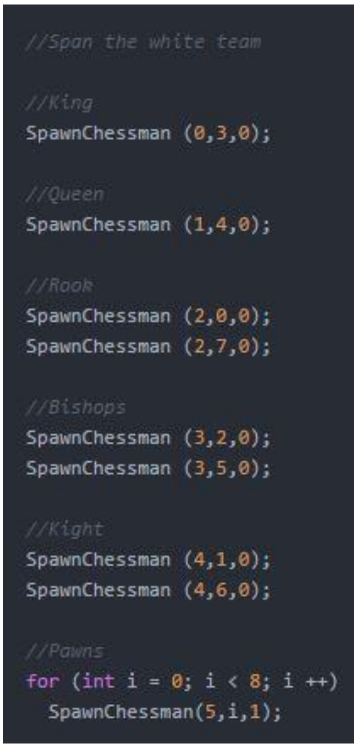


Figure 26: Working Chessboard Code

Luckily it worked otherwise the pieces that spawn would not spawn in their allocated coordinates.

Spawn Chess Pieces Test:

To test the chess pieces spawning the DrawChessBoard had to be working correctly, otherwise the coordinates that the pieces were given will not correspond to the same location as the board. This code is seen below:



```
//Span the white team

//King
SpawnChessman (0,3,0);

//Queen
SpawnChessman (1,4,0);

//Rook
SpawnChessman (2,0,0);
SpawnChessman (2,7,0);

//Bishops
SpawnChessman (3,2,0);
SpawnChessman (3,5,0);

//Kight
SpawnChessman (4,1,0);
SpawnChessman (4,6,0);

//Pawns
for (int i = 0; i < 8; i ++)
  SpawnChessman(5,i,1);
```

Figure 27: Spawn Chess Pieces

This is where the chess pieces are given their coordinate to the board, they are perfectly lined up the way a chessboard should be. To test this, I had to run the game and check that they all fell into the same squares that the coordinates indicated.

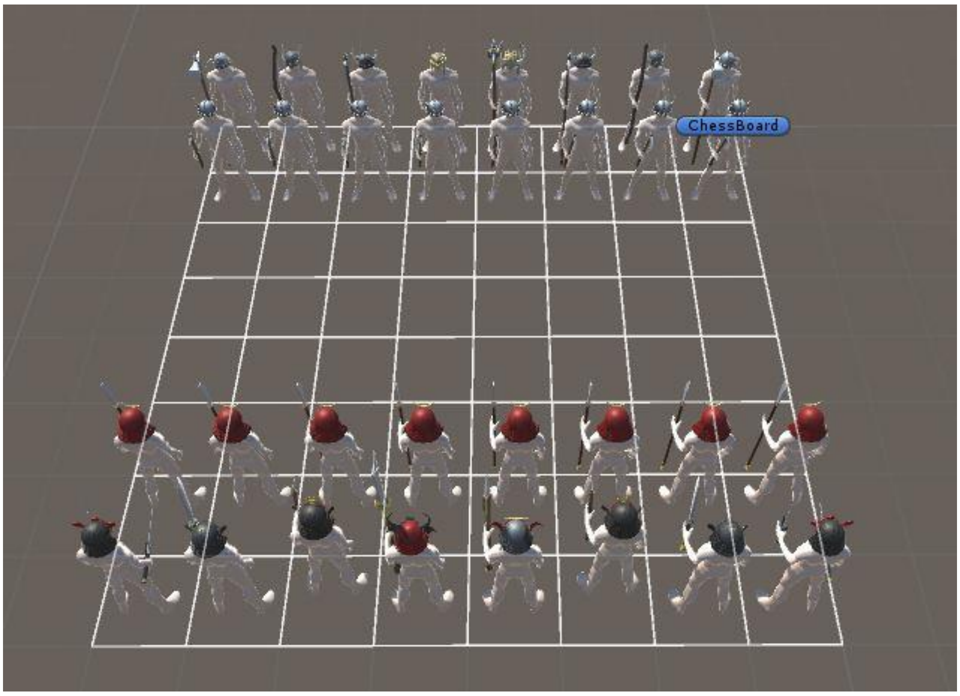
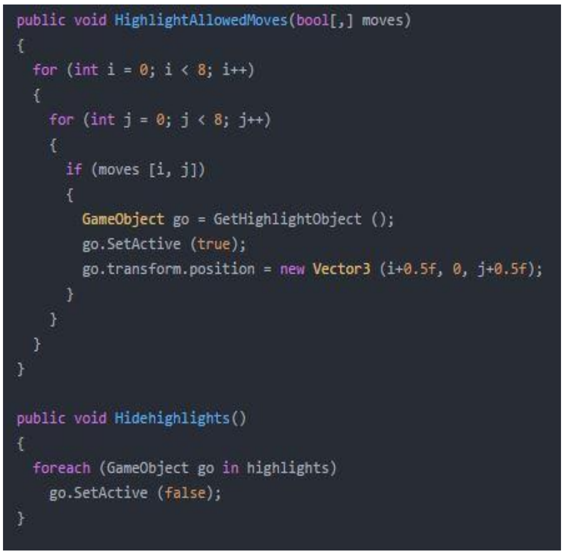


Figure 28: Chess Piece Spawn Test

This test worked perfectly, if not then the game would be unplayable.

Movement & Highlight Test:

To make sure that the pieces were moving in the way that they are intended I created a HighlightAllowedMoves section in the code, this works on a true or false basis and will return the movement of each piece on the board. The code for this is shown here:



```
public void HighlightAllowedMoves(bool[,] moves)
{
  for (int i = 0; i < 8; i++)
  {
    for (int j = 0; j < 8; j++)
    {
      if (moves [i, j])
      {
        GameObject go = GetHighlightObject ();
        go.SetActive (true);
        go.transform.position = new Vector3 (i+0.5f, 0, j+0.5f);
      }
    }
  }
}

public void Hidehighlights()
{
  foreach (GameObject go in highlights)
    go.SetActive (false);
}
```

Figure 29: Highlight Allowed Moves Code

Having set the movement for each piece on the board (code for this in the implementation section) the only way to test if the pieces are truly moving in the way they are intended was to create the highlight feature. The images below indicate how this highlight feature works:

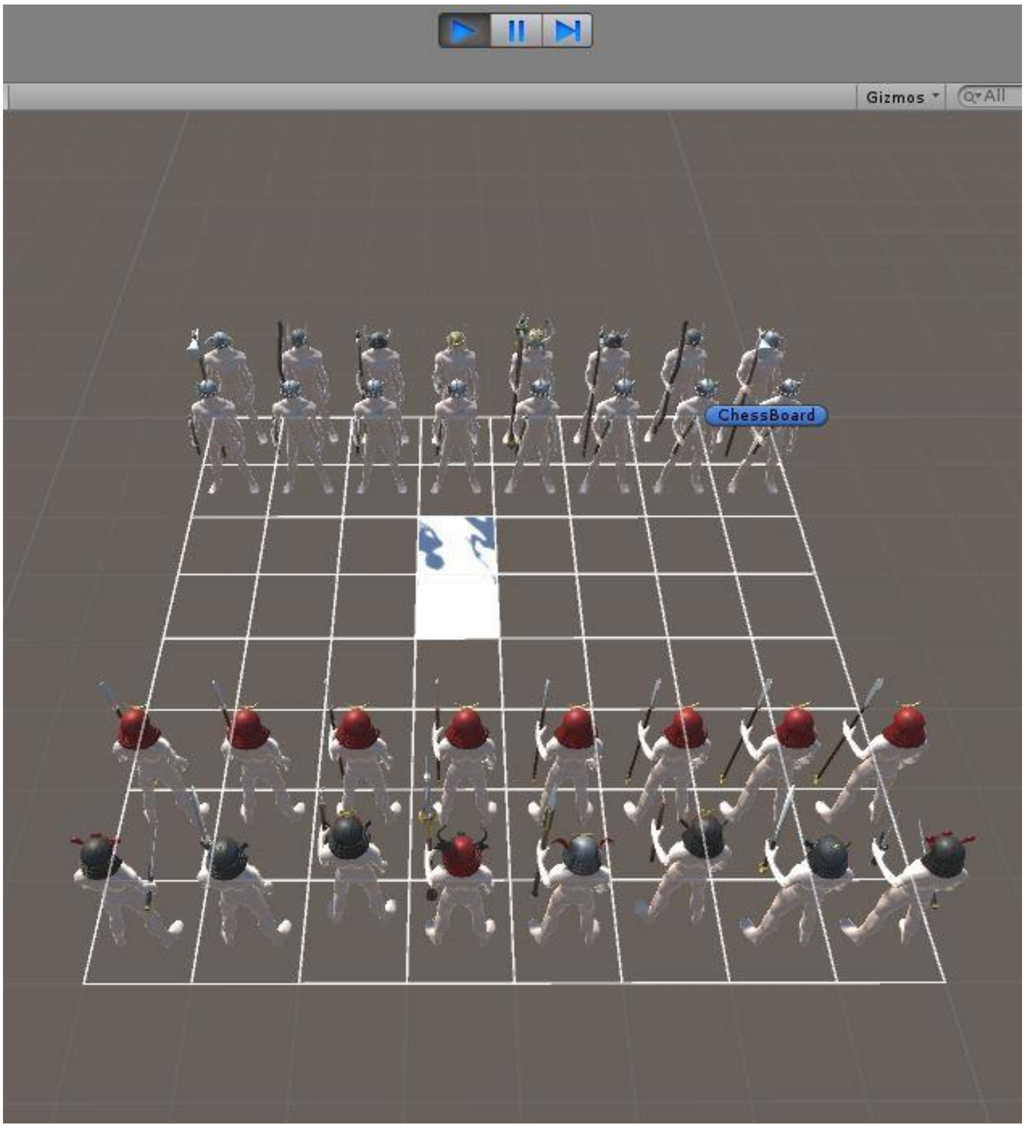


Figure 30:Highlight Allowed Moves Test

When the running the application, the highlighted area indicates the tiles that the white pawn can move on its first movement. It can move one to two spaces on its first movement provided there is nothing obstructing it. This test proved successful

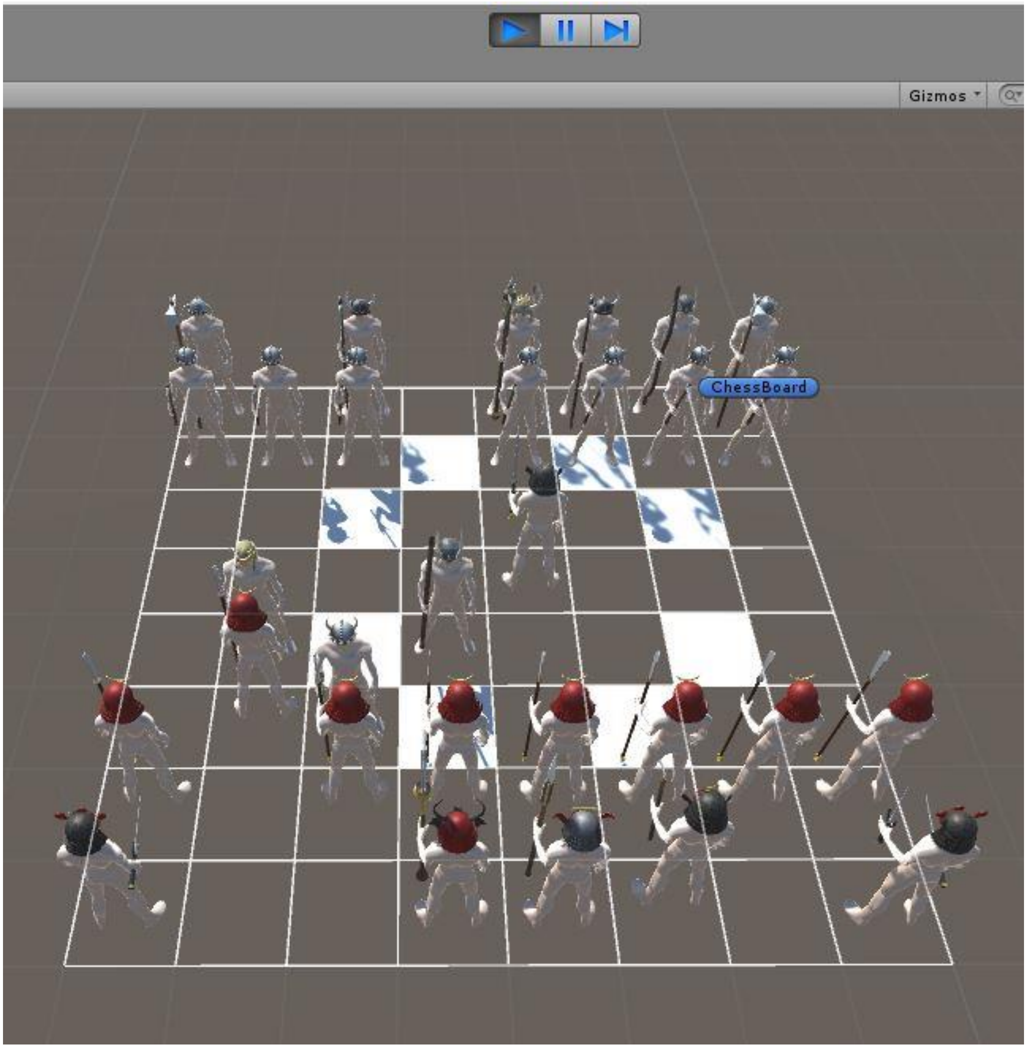


Figure 31: Highlight Allowed Moves Test 2

In this image, the game is running and for demonstration purposes I have moved the pieces in such a way to indicate how the Knight moves. Here the black Knight is selected and (as seen in the Knight code in the implementation section) moves in an L shape around the board. This test proved successful.

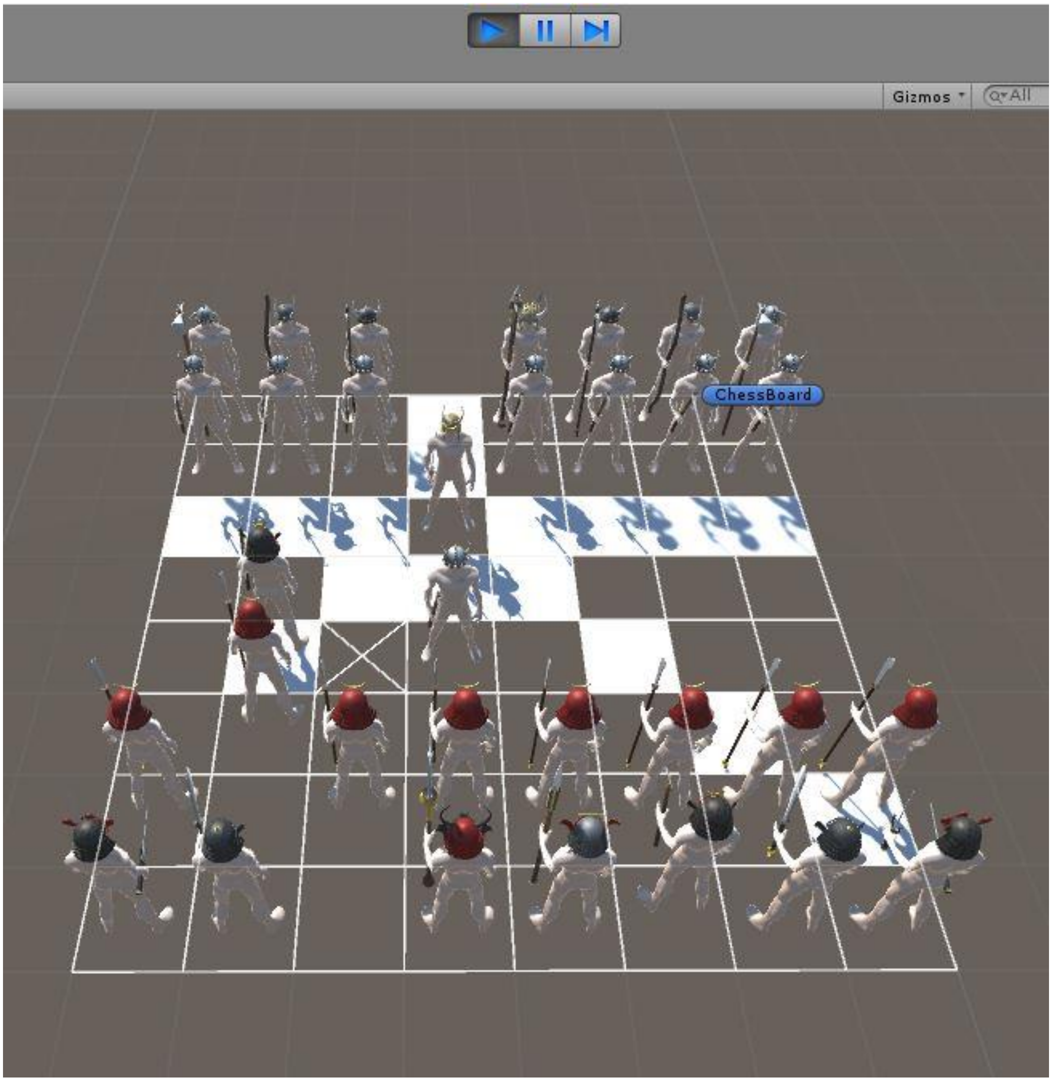


Figure 32: Highlight Allowed Moves Test 3

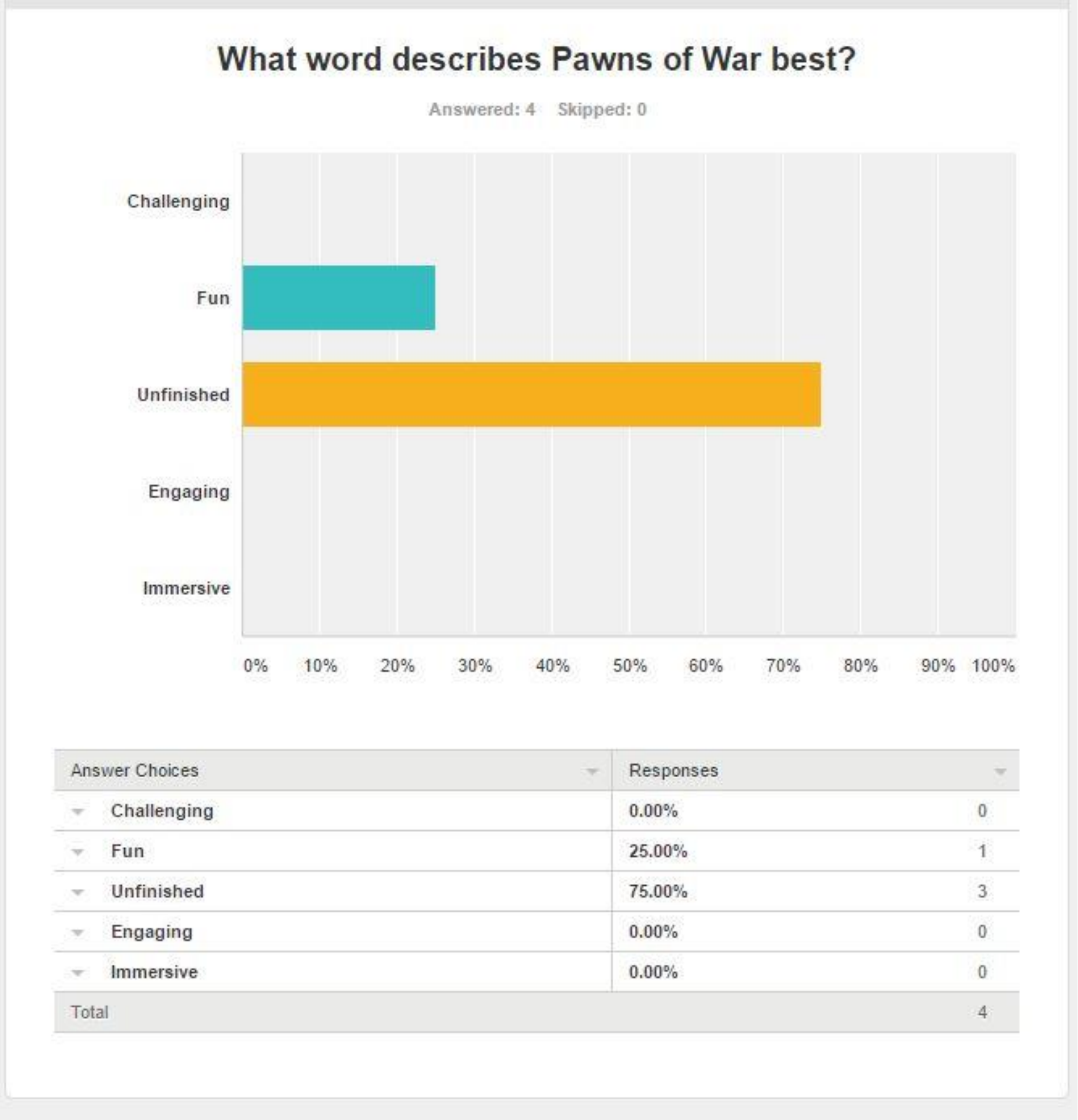
The final test on the chess piece movement is on the white Queen, here you can see the highlighted tiles that this Queen can move to and were it is being cut off due to obstruction. The Queen can move in the same directions and the Rook and the Bishop as shown here. This test was carried out by running the game and navigating the pieces around in such a way that I could test the Queen.

Customer Testing

The final test for this project was a customer survey, like any product the best was to get information on the best direction to go with a product is through the customers that you are targeting. The following survey was carried out on possible customers aged 18 – 26. Of the 10 surveys that I sent out I got 4 back, the following are the questions that were asked and the finding that I got back.

Customer User Survey:

Question One: *What word describes Pawns of War best?*



What word describes Pawns of War best?

Answered: 4 Skipped: 0

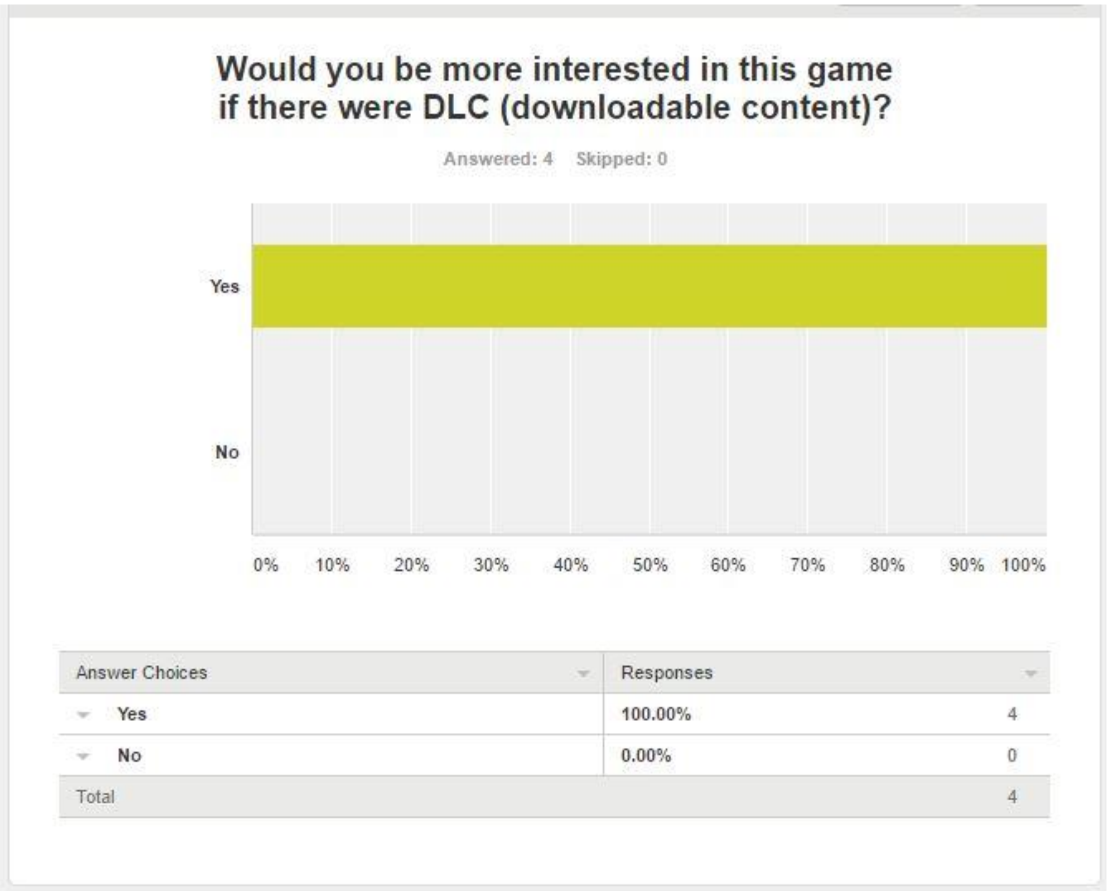
Answer Choices	Responses	
Challenging	0.00%	0
Fun	25.00%	1
Unfinished	75.00%	3
Engaging	0.00%	0
Immersive	0.00%	0
Total		4

Figure 33: Customer Survey Question 1

Analysis & Solution:

It is safe to assume that the game does not come across as being finished, this can be seen as a benefit because I can get solid feedback from users to make improvements to the game for the future evolution of Pawns of War. The features that I want to implement into this project will hopefully lead to a successful turnaround in customer feedback.

Question Two: *Would you be more interested in this game if there were DLC (downloadable content)?*



Would you be more interested in this game if there were DLC (downloadable content)?

Answered: 4 Skipped: 0

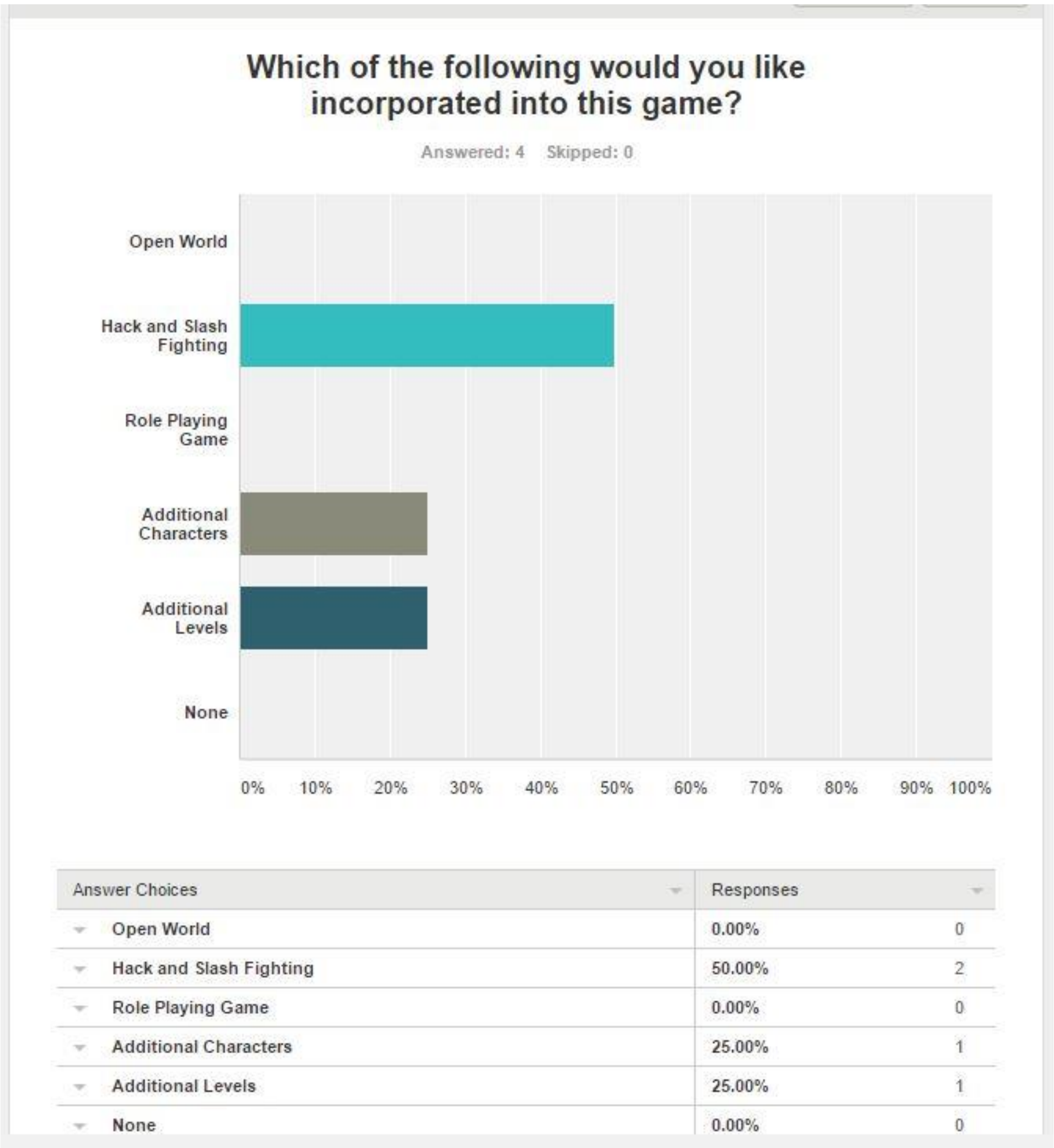
Answer Choices	Responses	
Yes	100.00%	4
No	0.00%	0
Total		4

Figure 34: Customer Survey Question 2

Analysis & Solution:

From the feedback on this question it further acknowledgement that with some added features and new DLC (downloadable content) it is very possible that the game will be more successfully received.

Question Three: *Which of the following would you like incorporated into this game?*



Which of the following would you like incorporated into this game?

Answered: 4 Skipped: 0

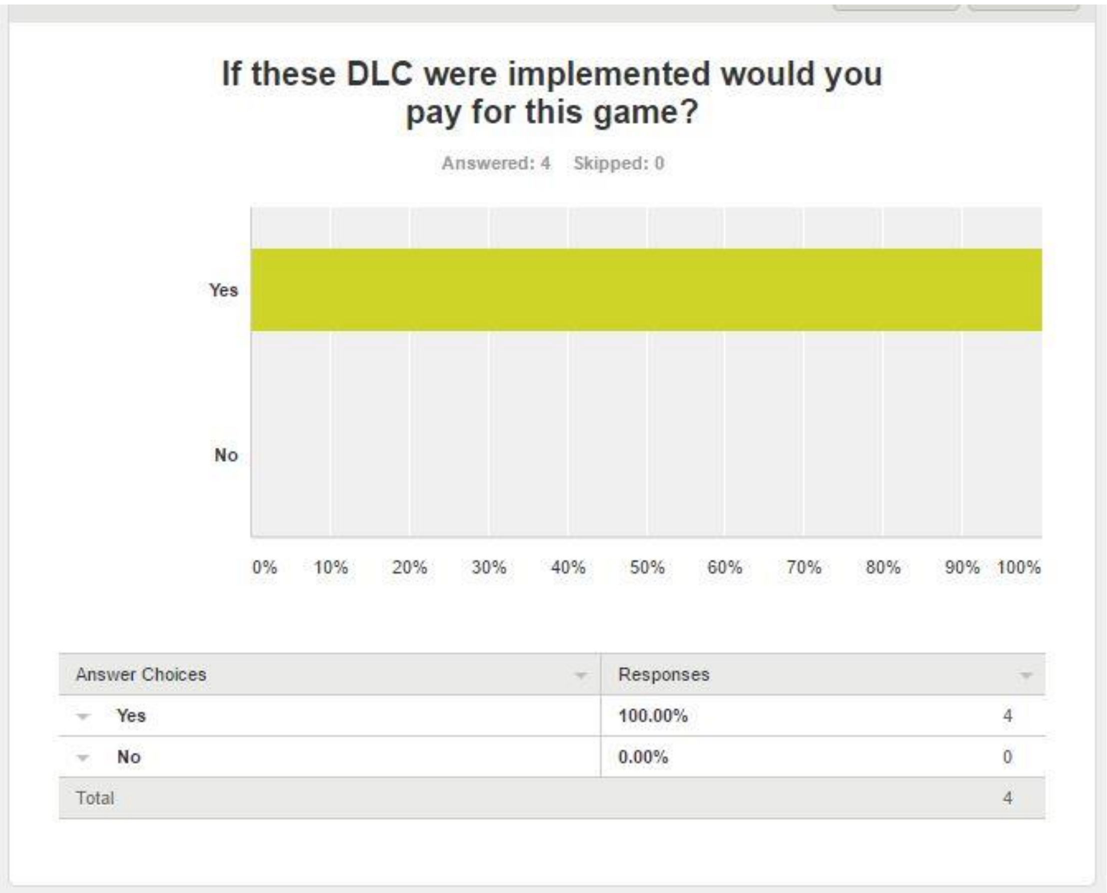
Answer Choices	Responses	
Open World	0.00%	0
Hack and Slash Fighting	50.00%	2
Role Playing Game	0.00%	0
Additional Characters	25.00%	1
Additional Levels	25.00%	1
None	0.00%	0

Figure 35: Customer Survey Question 3

Analysis & Solution:

This information from the survey will be the key component in creating additional content for this game title. The feedback indicates exactly where the game can be improved upon. With this information I know where to start making the improvements.

Question Four: *If these DLC were implemented would you pay for this game?*



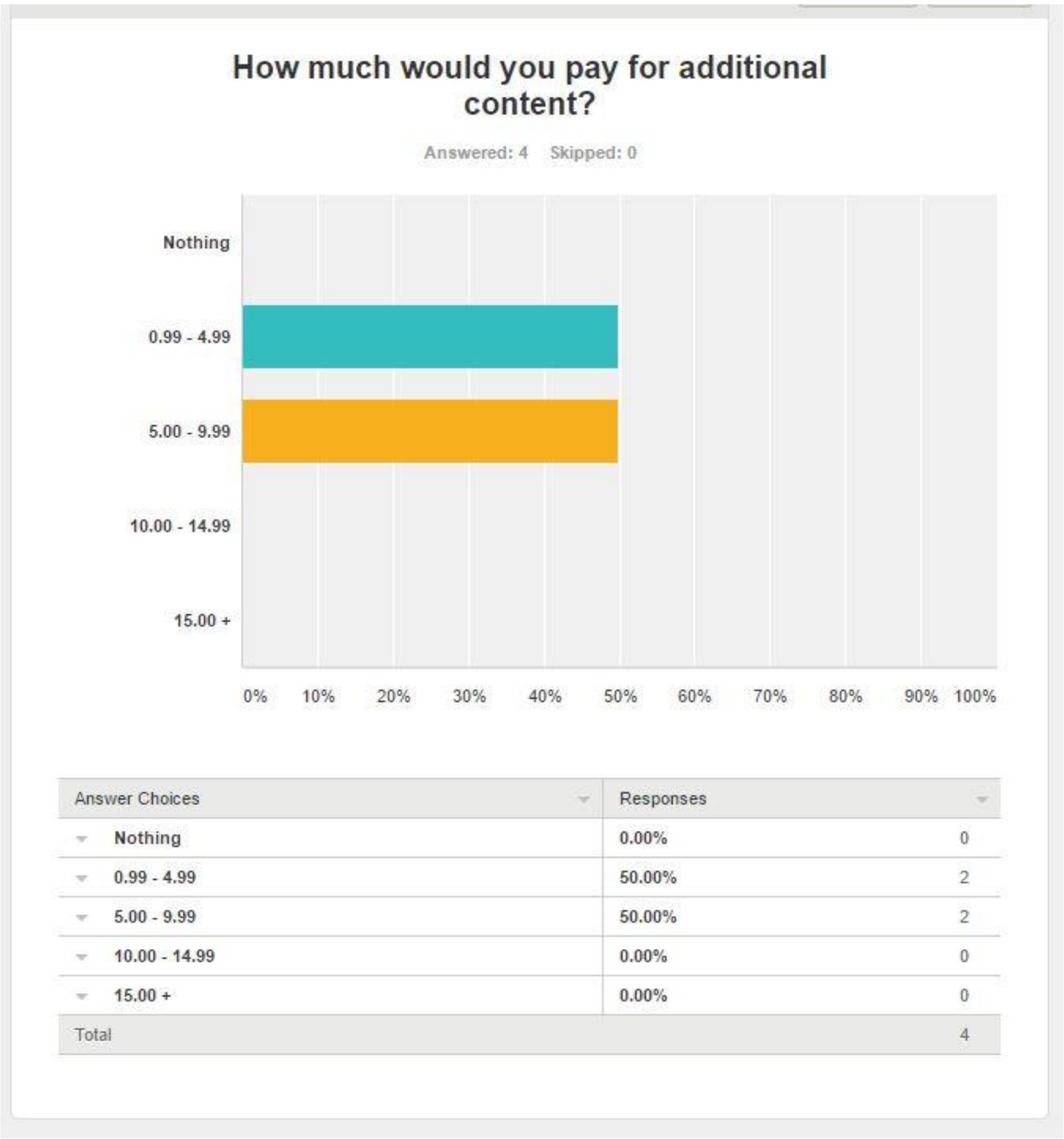
Answer Choices	Responses	
Yes	100.00%	4
No	0.00%	0
Total		4

Figure 36: Customer Survey Question 4

Analysis & Solution:

In addition to the other information we can see clearly here that there is a demand for the game to feature new content. This new content will be implemented in the form of DLC that the player can access through the games website or social media.

Question Five: *How much would you pay for additional content?*



How much would you pay for additional content?

Answered: 4 Skipped: 0

Answer Choices	Responses	
Nothing	0.00%	0
0.99 - 4.99	50.00%	2
5.00 - 9.99	50.00%	2
10.00 - 14.99	0.00%	0
15.00 +	0.00%	0
Total		4

Figure 37: Customer Survey Question 5

Analysis & Solution:

This information gives me a good starting point at which the game will be worth when the DLC for it has finally been released. From this I can get a good indication on how much gamers will be willing to spend on the game, from here additions can be made to help increase the gamers value on this title.

6 Graphical User Interface (GUI)

The following images are of the game as it is today, the game will be in a constraint state of development for the foreseeable future. The graphical user interface is something that I feel has an incredibly amount of importance when it comes to appealing to a wide audience. The following represents by game as best as possible.

6.1 Main Menu

This section of the project needed to be as appealing as possible, it also needs to be a good representation of what the game is all about without having to spell it out for the user. I left it simple, you have a background to indicate the game and three option, “Play Game”, “Load Game”, and “Exit” all self-explanatory for the user to get an easy grasp on what each button does.

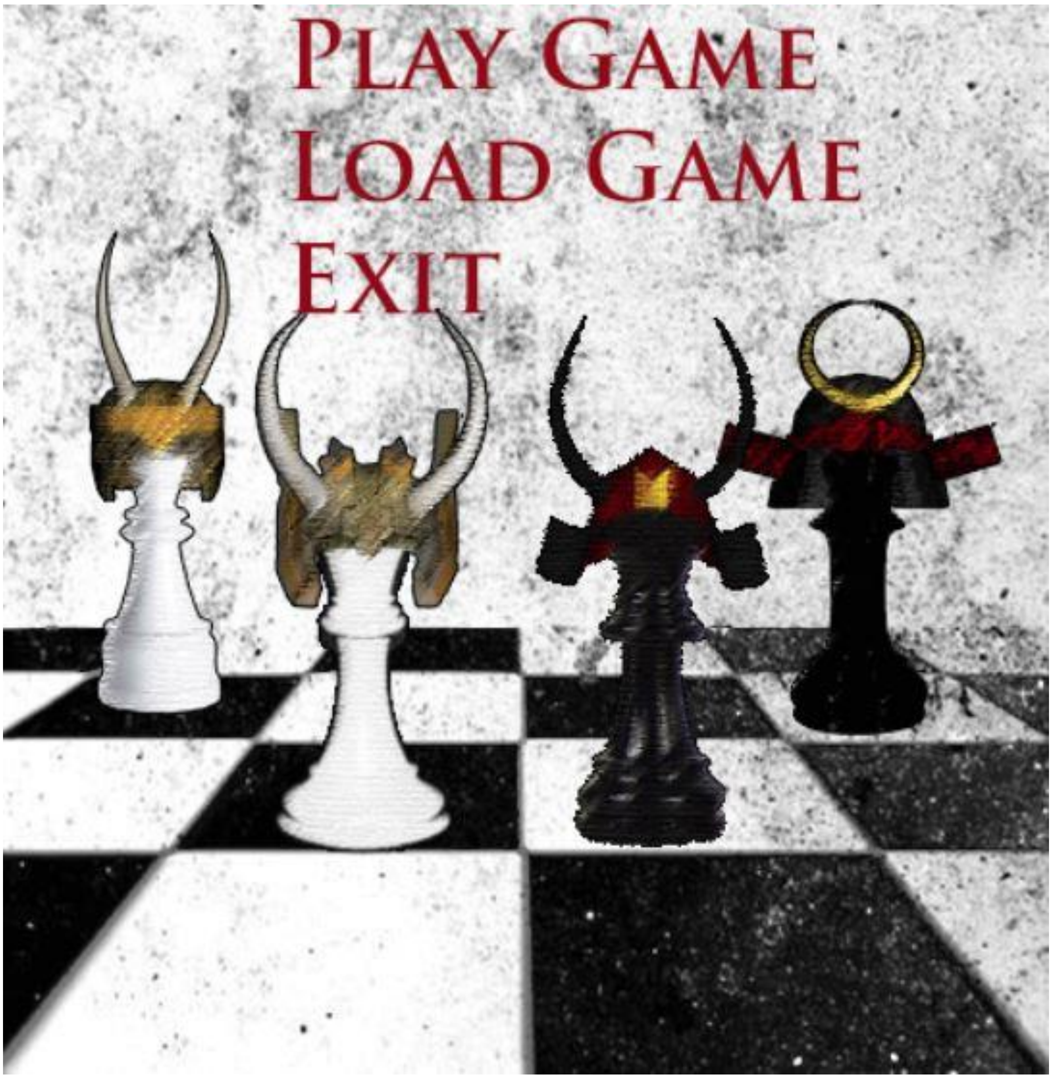


Figure 38: Game Main Menu

6.2 Gameplay

There are a number of camera angles in this section to show off the game from different points of view. The game will only have on angle when users are playing the game, however, I felt I necessary to so the game from different perspectives in the GUI as I would like to incorporate a multiple angle option in the game with the further development of the project.

The first image shows the gameplay from behind the Samurai / Black pieces, and looking at the Viking / White pieces.

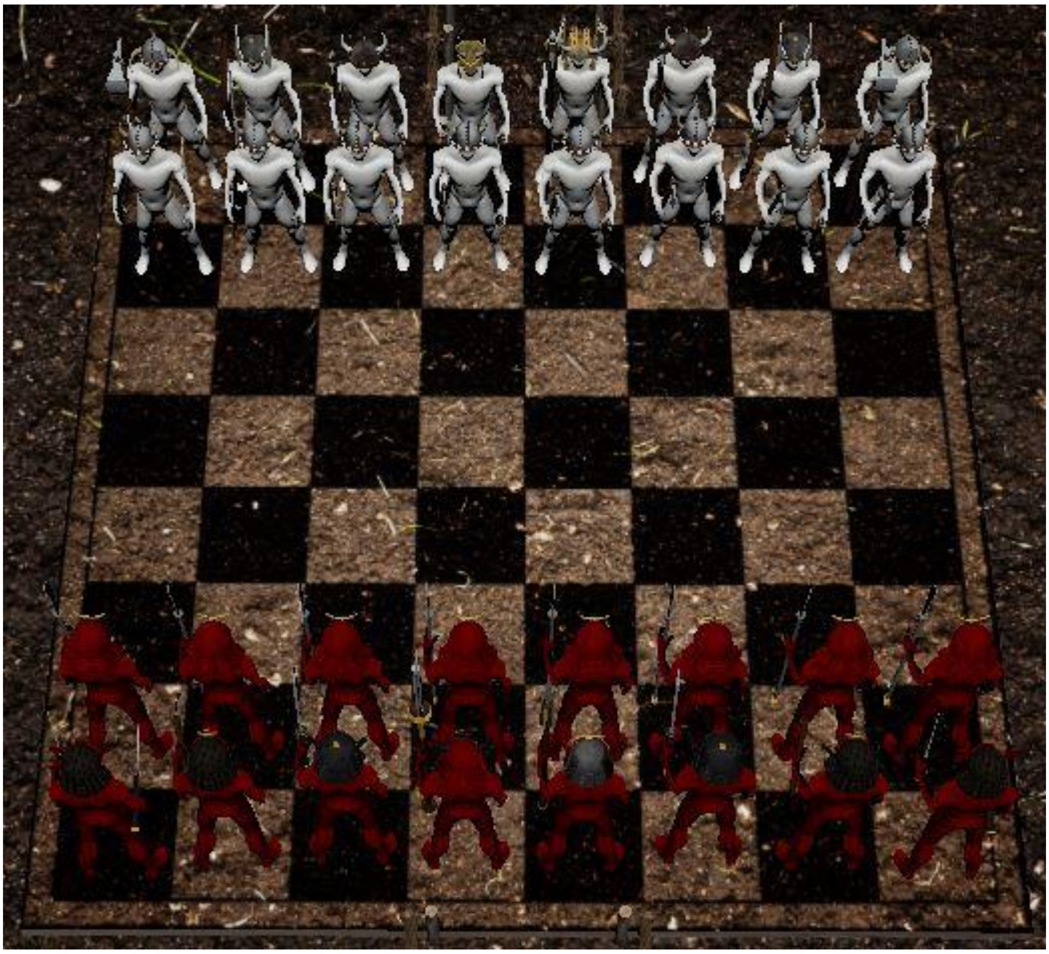


Figure 39: Gameplay Behind Black Pieces

In this image, we see the gameplay from a top down perspective, the view that I may be sticking with for now as it offers the best visibility to both teams. The image shows that the pieces have made there moves and are both lining up for attacks on both King.

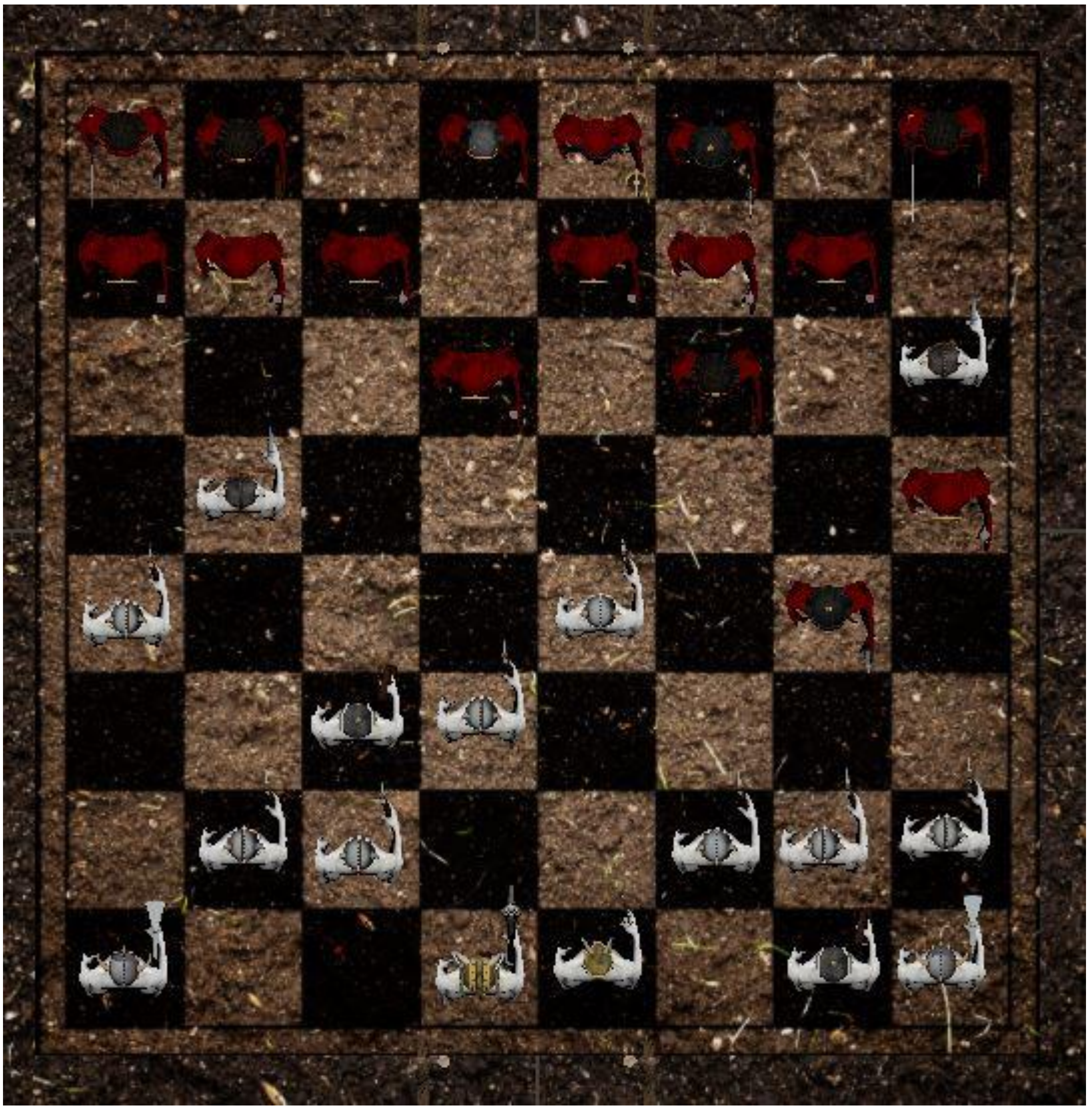


Figure 40: Gameplay Mid-Game

Like the image before this image indicates how the game looks at its starting position with the camera in a top down position, giving both side an even view of the board.

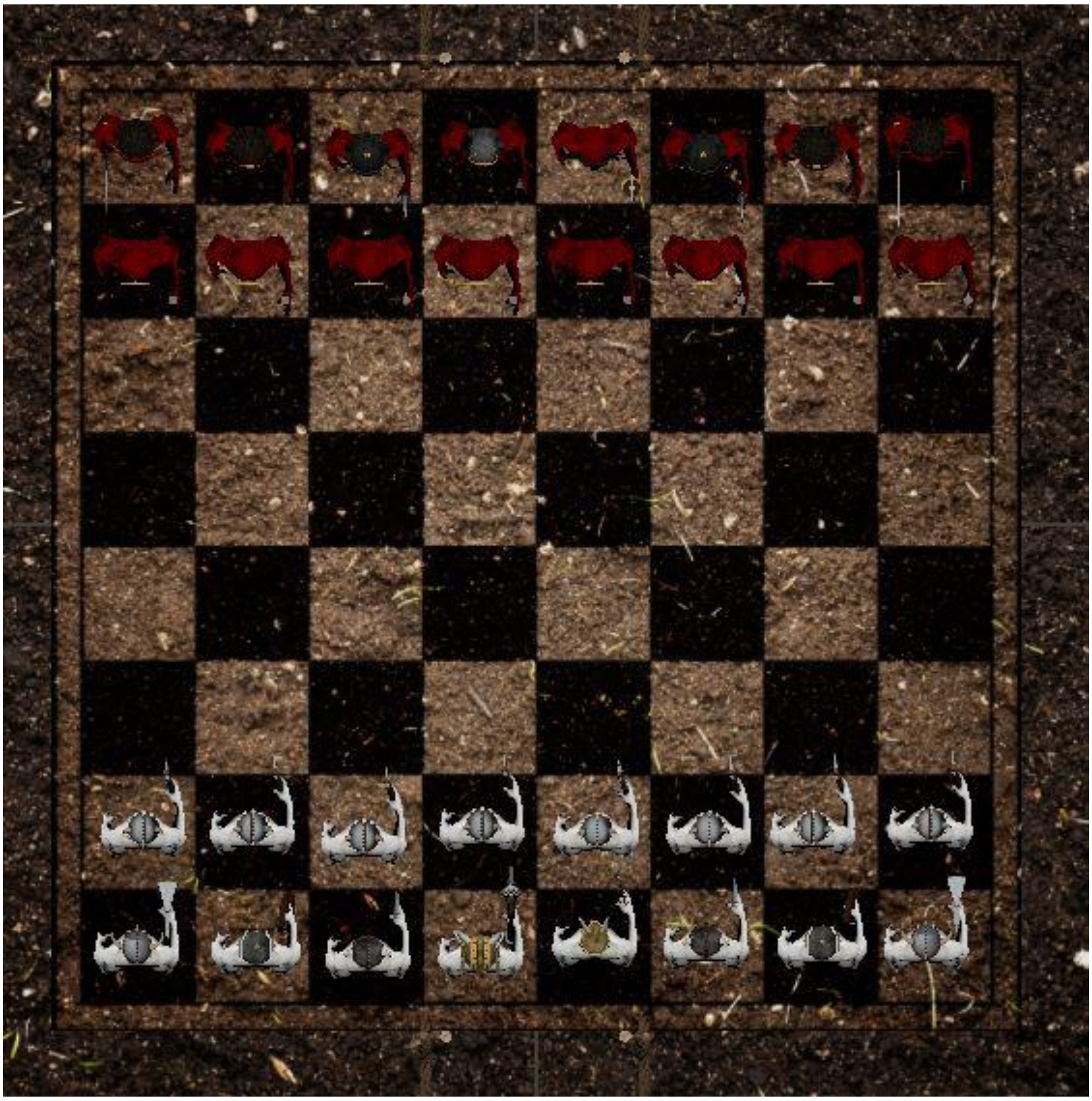


Figure 41: Gameplay Top Down View

Another possible beneficial view for both teams could be from the side down angle, this will provide the players with a solid view of the chess board that is even for both teams. Players may come into difficulty with this view if they are playing on an android device, so, I feel that the top down view is still the better one to go with for the time being.

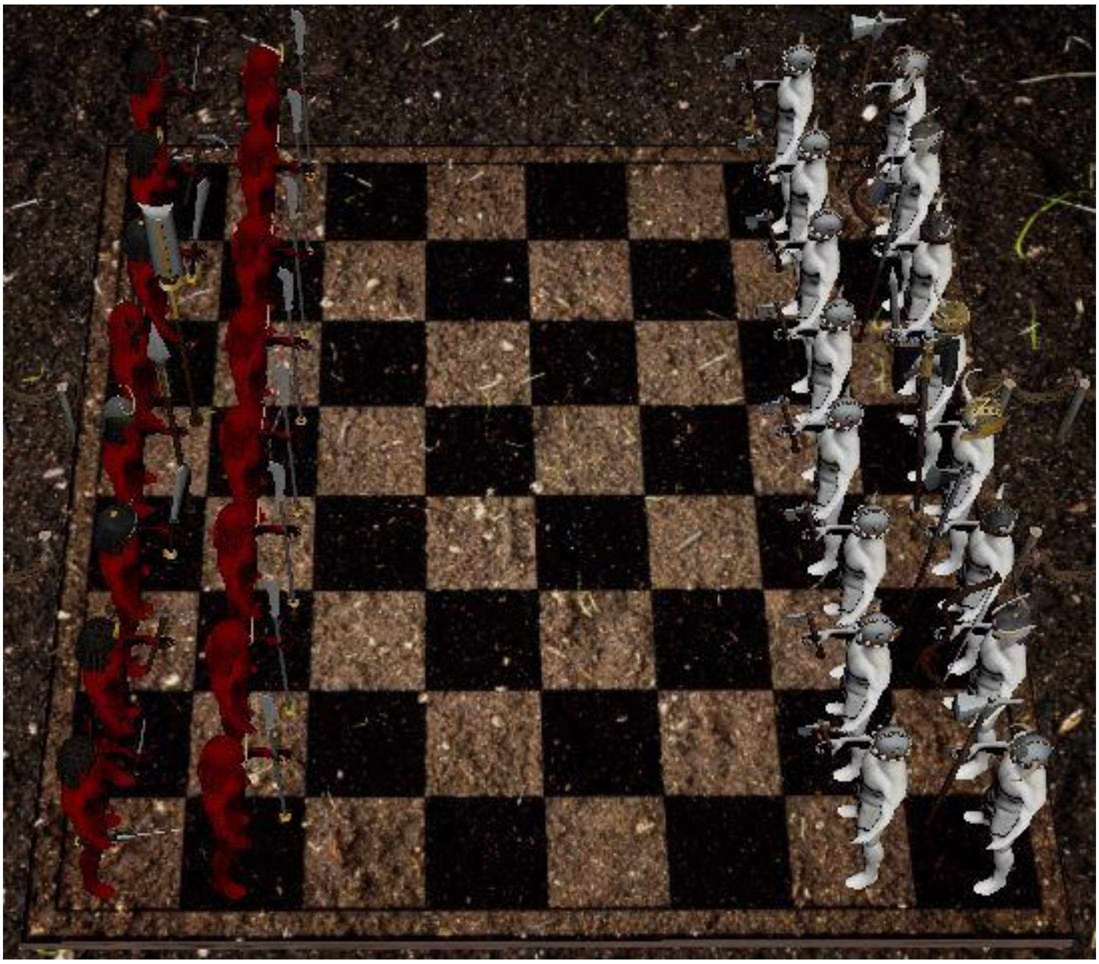


Figure 42: Gameplay Side Down View

6.3 Character & Environmental

The characters and environment for this game is important for setting the mood of the game, without it the player may as well be playing with the same boring pieces. For this project, I elected to create Samurais to take the place of the black pieces, and Vikings to take the place of the white pieces. The main reason for creating in this way was simply because I felt after creating the helmets for the Samurai and Viking, I felt that they were suited to take on them roles. I made the conscious decision to only create the helmets and weapons for these as modelling there full armour would have took me far too long and I would not have gotten them all finished in time. Below are the King for the Samurai in red and the King for the Vikings in white. They hold the most intimidating weapons of all the pieces as there are the most important.

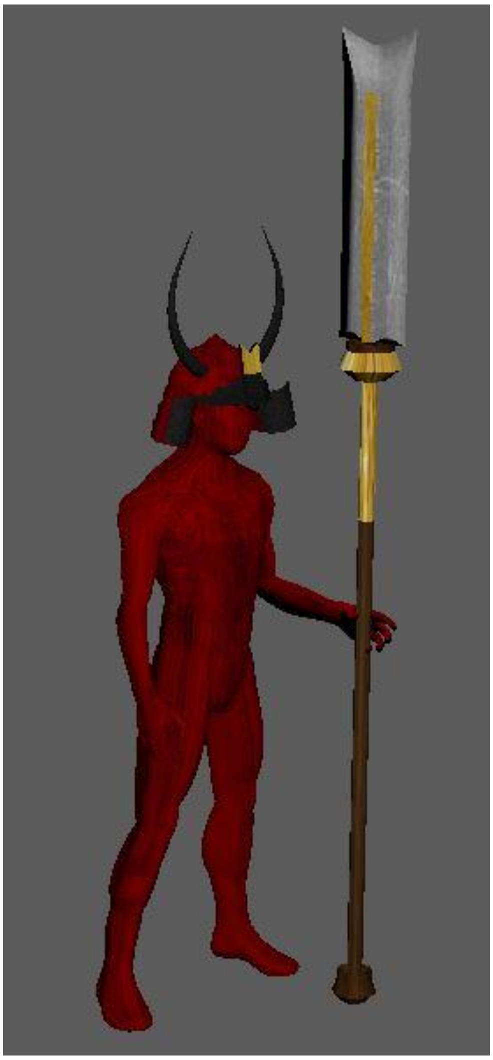


Figure 43: Samurai King Piece

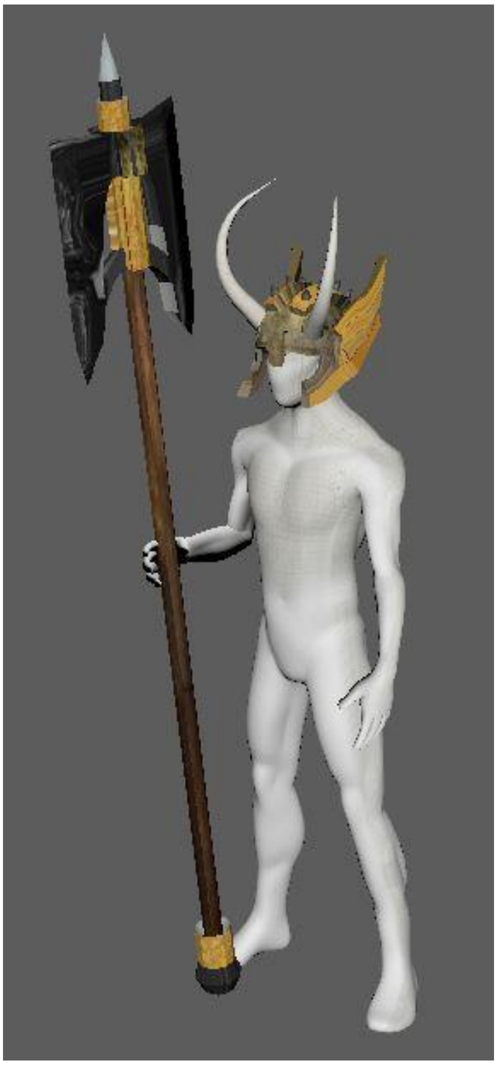


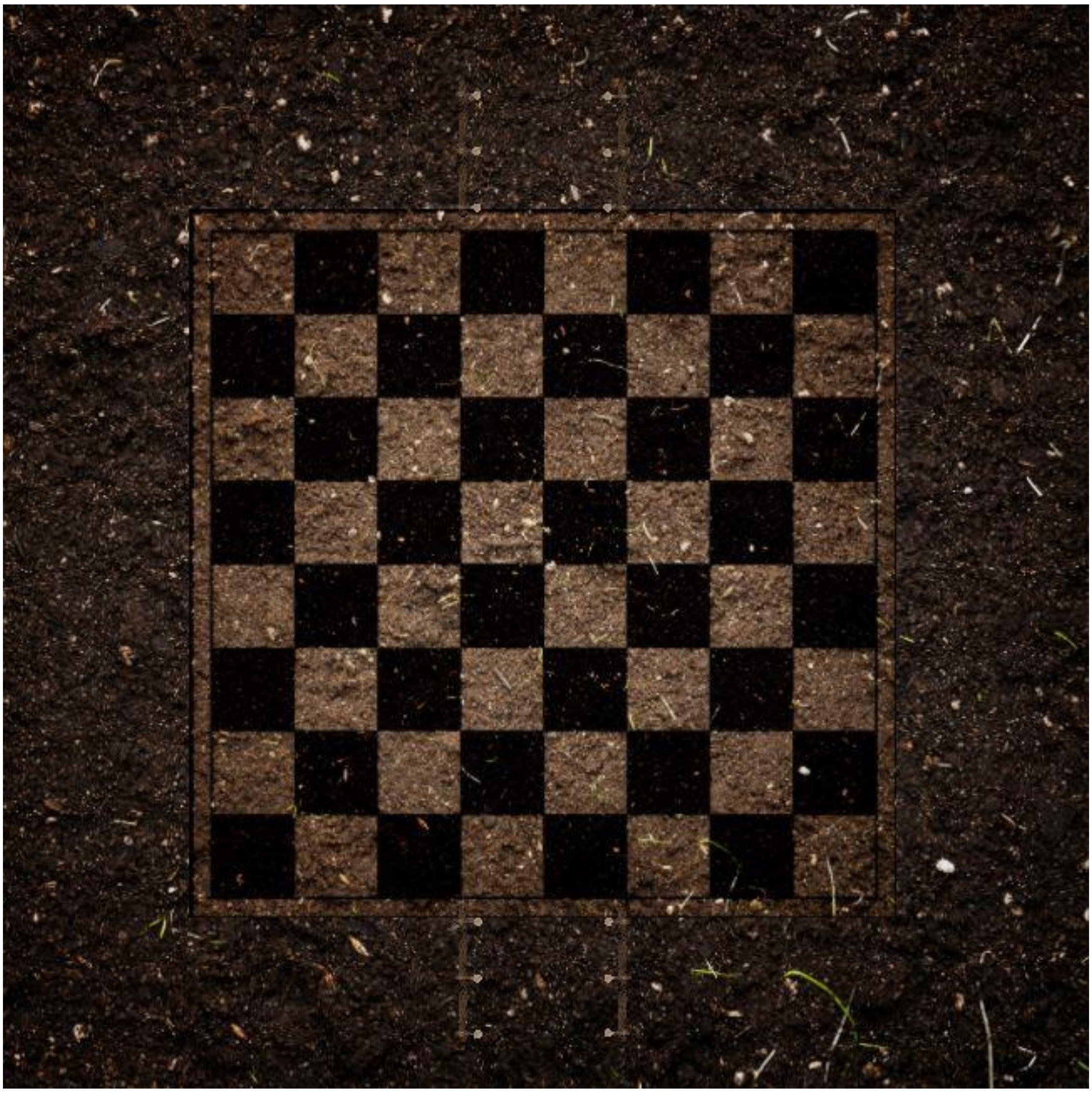
Figure 44: Viking King Piece

These Models below that I created in Maya are the Models that will act as Rooks in the game, the Rook goes along the board at a diagonal and can move as many places as long as there is not another piece in their way. Like the Kings I only created helmets and weapons for them as I felt that it would be substantial for this project.

Figure 45: Samurai Rook

Figure 46: Viking Rook

For the environment of the game, I wanted to make it gritty, after all it is a Viking vs Samurai based game. Also for the future development of the game I would like to make it an open world game in which you travel around the land as the leader of a clan looking for other clans to do battle with for territory, so, the dirt ground environment suited that perfectly.



7 Conclusions

In conclusion, I feel that this project proved to be more difficult that I had originally anticipated. I believe if I did not have other modules to attend, projects to do, and exams to study for throughout the process of working on this project I would have been more successful in the completion of the application. This however, is something that I can take away from this experience as well as the future development of an application I feel can be quite successful.

One of the key pieces of information that came with this report was the direction that this project can be lead, and further developed. The testing of this application was the key ingredient in finding out this information, the surveys provided the landscape in which to build additional content that will keep the application in constant development for the foreseeable future.

A project like this will need a strong team working on it, working as part of a team throughout college to working on this project on my own has shown me the value in having a team that can work well together to produce high quality content.

In saying that I am proud of how far this project has come and all the I have accomplished in creating this application. I am excited to continue this and make more content that will aid in the success of this video game.

7.1 Further Development & Resources

With more resources, where could the results of this project lead to?

Resources need to make this project a leading video game in the market place:

- A team to work on game development with a team leader
- A team to work on animation with a team leader
- A team to work on cinematics with a team leader
- A team to work on character design with a team leader
- A team to manage servers

If it was possible to acquire all these resources with the required hardware I feel that this project could lead to being one of the top strategy video games played worldwide.

Some of the additions that this would lead to are as follows:

- Single player, you play as the king of the clan in an open world game where you must travel the land to find other clans to do battle with over territory. Having to do various objectives along the way that involve your entire clan
- Create your own clan (design each member of your clan and give it a name)
- Online multiplayer, use real money to enter competitions with the chance of winning a cash prize.
- Fight for the square (instead of traditional chess, when a player wants to take out an opponent's piece they will enter a 2.5D fighting game in which they will have to fight over that square, after which the game will revert back to the board for the next move)
- Multiple warrior clans to select from.

- Build off of the chess style game play and have an optional checkers gameplay.
- Extensions that could be purchased. Such as:
- The animal extension (animal clan's players can use to play)
- The weapons pack (get all new weapons for your clan)
    - The projectile pack (weapons you can use to shout your enemy)

8 References

"Unity - Game Engine". *Unity*. N.p., 2017. Web. 10 May 2017.

Foundation, Blender. "Blender.Org". *blender.org*. N.p., 2017. Web. 10 May 2017.

"Maya | Computer Animation & Modeling Software | Autodesk". *Autodesk.com*. N.p., 2017. Web. 10 May 2017.

"Monodevelop | Monodevelop". *Monodevelop.com*. N.p., 2017. Web. 10 May 2017.

"N3K EN". *YouTube*. N.p., 2017. Web. 10 May 2017.

"Pinterest". *Pinterest*. N.p., 2017. Web. 10 May 2017.

"Buy Adobe Photoshop CC | Download Photo Editing Software Free Trial". *Adobe.com*. N.p., 2017. Web. 10 May 2017.

"Samurai Helmet". *Pinterest*. N.p., 2017. Web. 10 May 2017.

9 Appendix

9.1 Reflective Monthly Journal

The following journals were a great help to me in keeping track of my monthly projections. They were carried out from the months of September 2016 to March 2017.

9.1.1 September 2016 Journal

So, the time has come to really make a decision on what my final year project will be. I had been hoping it would come to me throughout the summer but nothing.

While sitting in the first class of forth year I was thinking “Feck” I really need to start forming some sort of ideas for my project, when nothing was coming to me I started to panic a little.

When I got home that evening I was sitting at my desk and wondering what am I going to do with this project. I started playing a few different types of games to see if the idea would magically just come to me, when nothing came to mind I start looking up the most popular mobile games now days.

I started to get some structure to the type of project/game I would like to make, I knew that I would like it to be mobile and pc compatible and, that I would like it to be a top down survival game.

Thinking back to the project that I worked on in 3rd year I knew I would like it to be something similar. The game we created in 3rd year was a top down zombie survival game, it worked with turn based mechanics. I worked on all the design of the game, it was done in pixel art so I knew I would be okay creating something with similar graphics.

While sitting at my desk looking over unity projects I needed a break so I went up to check on and feed the cows, I brought the dog up with me for a walk because he was nagging me to get out. When I got back to my desk I started to think maybe I am approaching this the wrong way, I realised instead of creating a game from thin air, why I am not thinking of my own surroundings and how they can be incorporated into a game?

Maybe I could make some sort of farming survival game, with electric fences and slurry pits as sort of obstacles and the animals start to get aggressive at certain times of day.

It all started to become clearer to me after that, the ideas were starting flow and I knew this was the one. Now I just had to bring it all together and form a more cohesive plan on what exactly will be the games mechanics and how to make the game that bit different to all other survival games.

I knew I wanted to work in unity and that I want it to be pixel graphics and that I would like to make it compatible with mobile and pc. Now I just need to work on my pitch and how to structure my ideas.

9.1.2 October 2016 Journal

After giving my pitch, I was left having very little confidence in my idea for the game I wanted to make. The lectures asked good questions and gave good feedback on my idea, and it could have been salvageable had I worked on implementing them into it, however while I was trying to figure out ways to make the game more appealable to others I had an idea for another game.

The idea for this other game was to make a 3D battle chess game, I know of 3 similar battle chess games that have been made but nothing like the one I was thinking about. At this point I was still unsure as to whether the idea I had for my original game was going ahead, so I was pondering what game I wanted to push for. I knew that I could make the first game more appealing to other although I feel it would have affected my overall result in that it was not necessarily an original idea. On the other hand the 3D battle chess game I knew would score a higher mark as it is unlike anything that is out there today and at the same time it would be very relatable to kids that play all the latest 3D games.

When I found out the Anu is my sponsor, I brought the idea of both of the games to her, she confirmed what I was initially thinking. So, from that I was going to go with the 3D battle chess game, I knew that it was going to be a more challenging game to make than the other one but it is worth the risk because I am aiming for top marks for this project.

When we had our meeting with Anu, she asked was I still going to go with the 3D chess game, when I told her that I was she seemed glad and I further explained the premise of the game to her and the other in her group. When I got home that night I start working on character designs, essentially, I am going to have to create/design 32 models and animate them in various ways so that they are unlike each other. I am very much looking forward to creating these, modelling them, and animating them I am just a little worried about the coding of this game. It is somewhat comforting to be creating a chess game within ai, hopefully the same principals of that can be implemented into this project, although it will still be a lot of work creating it.

I began filling up my sketch pad with life drawings, trying to get myself back used to drawing the human form. I was glad that since doing animation in college before this course, my drawing ability has not left me. During this reading week, I plan to start figuring out how the cinematics would work within the game, and how I could implement them.

9.1.3 November 2016 Journal

With all the projects that have been thrown at us I have found it very hard to find time to actually get working on the prototype for the midpoint presentation. Throughout the start of November, I began working on the very basic functionality of the 3d chess game, when I was looking into ways of doing it I was being charged 12 euro for the 3D chess pieces and the chess board. There was no point in buying them because I only needed them to be place holders for the characters that I will later be creating. After creating basic shapes to use a place holders for the chess game I start running into problems with setting out the chess board grid and calling the place holders to the game.

I decided to move on to creating a checkers game. I got thinking maybe it would be a good idea to have an additional game within my project, and the most obvious choice would be a 3D checkers game. I was thinking I could create 3d character similar to the way I will be creating the characters for the chess game. When I start coding the checkers game it went a lot smoother than the chess code, however after I had finished coding up the checkers game with the place holders there was a few little glitches. The game was playable but it will need a few touch ups.

From this I moved on to working on the models and creating some sketches to use as characters within the game. I was doing this whenever I had time in between all the other projects that I had to have done. To be honest at this point of the project I would have loved to be at least half way through the creation of my video game however, all the other modules have been taking up my time. In reflecting on the other modules, I would have benefited a lot more from having modules that will be a direct benefit to the outcome of my final project, why are we picking a specialisation if we are still having to do all the same kinds of modules that we have done before. Giving the circumstances I feel that all of this is having a direct effect on my final 4th year project.

Not a lot got done after the few sketches I drew up for characters, I was trying to animate the character that I had models however, I was having issues with it and was not able to put the time in to be able to correct it.

9.1.4 December 2016 Journal

During the end of November I had been very busy with projects that were due by the end of December. To keep on track of my main project I drew up some sketches on how I would like to create the characters.

I started to think that it was going to be a massive challenge for me to actually create all the characters for the game, the time in which to do this was getting more and more pressing. To make use of my breaks in between working on the project and attending classes I would research images that would help with the design of the game. I know that there is no way I was going to have enough time to create the game to its full potential but at least I would have an idea on the direction that I would like the game to go in.

With the technical report, due by the 11th I start working on that from the start of December. I very much enjoyed working on it and gave me a good idea on how I want to develop this game, however it also made me realise the limitations that I am going to be under when finalising the game in May.

In my mid-point presentation, I was presenting to Anu and Dominic, I was a little nervous because I felt that the game was at such an early stage in development that they would be questioning the final result. The information that I got back of them was very informative and helpful. They put my mind at easy when Dominic recognised that it was a big undertaking and told me to keep the work simple not to delve in to deep with the sort time left for it to be completed.

By the end of December, I had 80% of the pieces working however, I had very little of the design and I knew that the design and animation was going to be the biggest undertaking, and it was all still to come along with another semester of new modules.

9.2.5 January 2017 Journal

The end of December proved to be very stressful, working in group projects with certain people lead to a very panicked few week of work that almost ended in missed deadlines. All this took a toll on my main project and cause it further delays, I was sick of working in groups, mainly because it always leads to me having to rely on people the proved to be unreliable. I felt with it being our fourth year and having worked in multiple groups every semester that it was an unnecessary inconvenience to be forced to work in groups again during our final year.

Getting any work done on my project was very difficult during this month as the exams were scheduled in such an inconvenient way that lead me to feel like I wanted to just quit the course. I have never felt so stressed in all my life, and with the group work from December still looming over my head it was even more difficult to get through the exams.

After the exams had finished there was little to no time before college was starting again. I was unable to get any work done on my project as I needed a few days to get my head straight before the next semester started back again. The new semester started and with it more project and group work was dished out, I felt as though this semester would prove to be more manageable than the last as the three new modules are something that I will very much enjoy learning more about.

Still I have not been able to further my progress on the project and am hoping to get a bit more of a handle on this as this semester continues.

9.2.6 February 2017 Journal

The new semester seemed get off to a good start, the projects and assignments were very reasonable and are not as demanding as last semester. However, the project was very slow in coming around. I was having great difficulty with the animation of the character that I created. This character is a simple base model that all the other characters will be based from. With the project, due in three months I did not have enough time in which to create 12 character and multiple levels for the characters to battle on.

I set up rigs for the character in blender hoping that I could transfer it into Maya easier that way, thought the rigs where easy to set up the animation was very time consuming and ended up leading to nowhere. I then tried to set the rigging up in Maya and animate It from there to see if this would prove to be easier, but as it turned out it was not, this lead to more time waisted and nearly the whole month of February being a complete waste of time.

I had a long think about what I was going to do about this project. it would have been so much easier to create a game that worked off the tutorials from unity (which is what I am sure many people have done), but I knew that this was going to be challenging and I just had to do what I could in the time that was remaining.

I decided that I was going to leave out the animation, if I had time I would do it at the end if I had time remaining. I needed to concentrate on the project at hand and see just how much I could get done.

9.2.7 March 2017 Journal

After deciding that I was going to just concentrate on the game itself and leave out the bells and whistles, the next step was to finish the movement of the chess pieces so that the game actually done what it was supposed to do.

The only pieces left to have working was to Rook the Queen and the King. These pieces seemed to be easier to get working and went a lot smoother than the others. Once I had the Rook working all I needed to do for the queen was add the Bishops movement and the Bishops movement.

The King proved to be handy in its own right. All it needed to do was to move +1 and -1 throughout the Y and the X axis.

Having the pieces working was a very positive thing however it was going to be difficult to incorporate the new pieces that I will have to finish creating. If not I will have to do the game all over again using these pieces. I am just hoping that I will be able to incorporate these pieces into the game.

Now I just needed to create that characters for the game so that I can start trying to add them to the game. I was not going to be able to get all the pieces completely finished so I am just going to create the helmets of Vikings that will play as the white pieces and Samurais that will be played as the black side.

Over the next two months I will be working on getting these finalised and finishing off other projects as well as studying for the last two exams. It is going to be a very hectic time and I am just hoping that I get enough done to attain a high enough grade to continue on and do a masters.