**Project Report**

41891 - Professional Practice in IT

Create a 3D Video Game



<https://github.com/GallagherStephen/3dProjectGame>

G00314689

Year 3

Software Development

Stephen Gallagher

G00354442

Year 3

Software Development

Dylan Creaven

Table of Contents

[Introduction 3](#_Toc38466182)

[System Requirements 3](#_Toc38466183)

[Technologies Used 3](#_Toc38466184)

[Design Methodology 3](#_Toc38466185)

[Limitations. 4](#_Toc38466186)

[Research 4](#_Toc38466187)

[In Game 5](#_Toc38466188)

[Storyboard of game 9](#_Toc38466189)

[Game Requirements 9](#_Toc38466190)

[Possible tests 10](#_Toc38466191)

[Game Design Process Pitfalls 10](#_Toc38466192)

[References 11](#_Toc38466193)

[Recommendations for Future Development 11](#_Toc38466194)

[Conclusion 11](#_Toc38466195)

[Learning Outcomes for Stephen Gallagher 12](#_Toc38466196)

[Areas Learned on: 12](#_Toc38466197)

[Areas I would like to focus on next time: 15](#_Toc38466198)

[Learning Outcomes for Dylan Creaven 16](#_Toc38466199)

[Areas Learned on: 16](#_Toc38466200)

[Areas I would like to focus on next time: 17](#_Toc38466201)

# Introduction

We set out in this project to challenge ourselves to create something using software that we had never used before. We had some experience using the 2D version of Unity, a development platform, but the 3D version of it was completely different in terms of tools, how to use it, and to put together a whole game when compared to the 2D version. It was going to involve a steep learning curve and teamwork to create a working 3D game. We prepared a schedule of the weeks ahead to meet and gave ourselves targets to hit. We took some setbacks in our schedule, but we were able to deliver what we believe is a good game before the deadline.

# System Requirements

The system requirements for this game are very basic. The game is does not exhaust the computer’s hardware. So, any PC with say a minimum of any i3 processor, 2GB Ram, a few 100MB(?) of space, Windows 7/8/10. It is a basic demo game, with a single level and will run on virtually every PC.

# Technologies Used

We primarily used Unity 3D development platform for creation and development of the game. We used this because we were already familiar with the UI and layout of Unity from our 2D modules, just not familiar with any of the 3D controls, tools or functions, but we felt that a small familiarity was better to start with than no familiarity with a brand new platform such as CryEngine3, Indie Game Maker, GDevelop or Unreal Engine. Unity gave us the ability to just create the game our way, with freedom of the creative design, even the ability to import our own created 3D characters if we wished while Unity handled the compiling into the different platforms e.g. PC, PS4, iOS, Xbox One, Android. For any created characters, we used Blender, an open-source computer graphics software toolset. Blender gives you everything you need to sculpt and mould your own characters. A good feature of Blender is that Unity supports Blender files, so it was easy to import and use our creations immediately in Unity in our game.

# Design Methodology

The methodology employed by us was a **scrum** methodology. We thought scrum would be optimal for us, in a student environment, where our time is already grouped by weeks/days so using a sprint-orientated methodology was ideal for us. We chose scrum for some adaptability and flexibility in terms of our schedule, which can be hectic as students, and so we would be forced to work closely on the code together as a single unit. Our mentor also contributed to the scrum methodology by not telling what to do/ what not to do which feeds into the lack of “leader” in a scrum workplace. This also helped boost our morale while in development as we had creative freedom and thus were invested into the game more and identified with the project.

# Limitations.

There wasn’t much limitations in the way of bugs/errors/Unity problems as we understood most of the basic concepts of our design from our 2D module and we used a lot of Unity documentation, YouTube tutorials and advice from seasoned programmers to help us in development of the game, resulting in few programming limitations. There was a lot of *time* limitations though, with not only doing our class work and study during the weeks of development but with the projects we had for our other classes. We got through this by sticking to our schedule and project plan strictly and communicating regularly as we could with our mentor and with each other so that we didn’t fall too far behind in our development.

# Research

***What is a 3D Game?***

3D gaming is interactive computer entertainment that is graphically presented in the three dimensions of height, width and depth; the addition of depth to 2D gaming enabled the exploration of virtual worlds with more realistic representation.

Example image of a 3D game below:



What are examples of 3D games?

**3D games:**

**1) Call of duty modern warfare**

**2)Battlefield v**

**3)Fortnight**

Pictures of the games in list above as an insight to a **3D** game:

1) 2) 3)

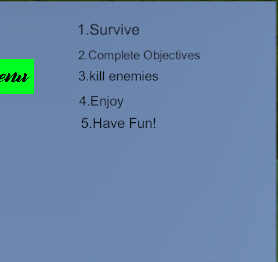


# In Game

In the Main Menu, there is a Start button, to begin the day, and a Rules button to show the player’s what to do and how to play the game.



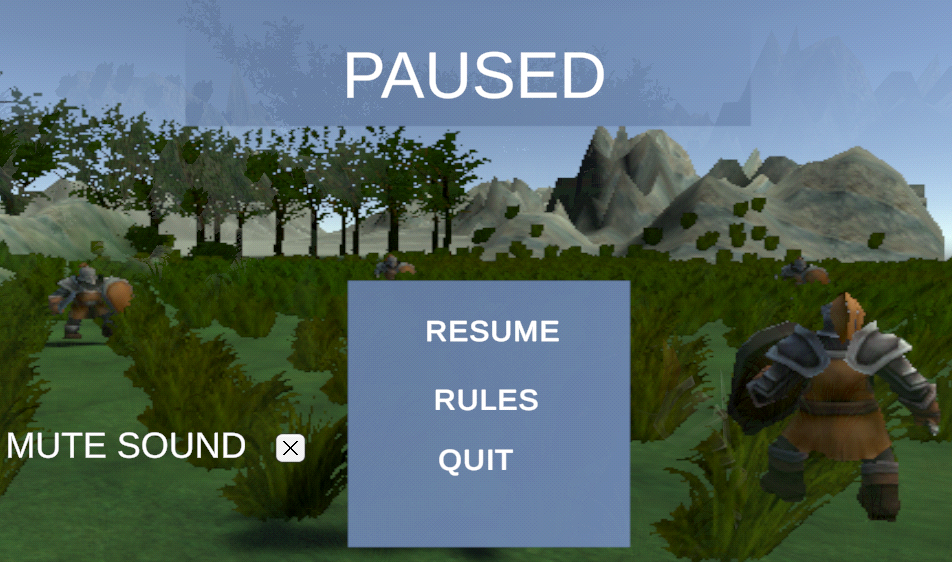
Below is the Rules scene/menu which gives a brief description on how to play/ understand the game and its objectives.



We chose the objective of the game to be very basic, the purpose of the player is simply to reach a flag at the end of the map and to not get killed by any of the enemies along the way. Once they reach the flag the game will end prompting the user to the end game scene.



There is an Options Button on the screen for the Player to bring up a pause menu and mute the sound or quit to main menu or to even view the rules of the game again



We took inspiration from classic shooters such as NightFire, GoldenEye, Halo 3 for its Capture the Flag scenario. We also took inspiration from ‘From Honor’ and the Elder Scrolls IV: Skyrim as the basis for our Vikings/ Scandinavian/Medieval atmosphere and environment.



*Figure 1 For Honor - Viking character*

*Figure 2 Halo 3 - Capture the Flag*

**Control Mechanism**

For my game we will have my game controlled using the **arrows** to look **(left and right)** if wantedor **WASD** keys to move (in all directions) while having the **spacebar** to Attack and using **escape** to pause the game

As displayed below in the image and if researched, using either of these two types of control mechanisms can be quite effective and nice to use in games.

Images below:





**The Game**

**Chosen type of game**

The type of game we Have chosen is a melee medieval 3D game. We Have Chosen to go with this type of game as it would be the type of game we would preferably go for if choosing a game to play. We feel that there are many shooting games out now and are very common but hopefully we can make a game that is unique and that changes the experience for the user like no other game.

**Overview of game**

The game will consist of a medieval character that can attack his enemies coming at him using his sword. Attacking to stay alive, trying to keep the players health to maximum for long as possible while trying to get to the objective, that being make it to the final objective (the flag at the castle) and complete the game.

**Goal of game**

To keep going killing the enemies coming at you till you get to the final objective. The final objective being reach the castle's flag.

# Storyboard of game

The story of the game will be a heartfelt survival where the medieval player is trying to stay alive and not be taken down by these enemies that are trying to cause harm to the player. The tense/stress of keeping this player alive for you to reach the final objective will attract an addictive experience for the user making it hopefully a well-liked game for all.

# Game Requirements

Some game requirements listed below:

* Create a main character
* Character being able to Attack
* Character being able to kill enemies if player hits them
* enemy health
* player movement using WASD
* Main menu
* Pause menu
* Game over screen when dead
* Enemies inflicting damage to player

# Possible tests

Possible tests:

* moving to and from each Scene (menu – > gameplay - > pause menu – > end)
* Player being able to attack enemies
* control mechanism working (WASD or arrows and left click)
* Displaying end game scene when dead (game over)
* menu buttons work
* Additional features work correctly
* enemy being able to damage the player

# Game Design Process Pitfalls

Looking into the designing of a game for this project we came across an interesting website which had listed the top 10 game design process pitfalls and hopefully after our research I don’t land in one of these pitfalls or one of my own.

we found reading this that it gave us a broader outlook on the importance of the game design process and things that may be overlooked.

The website listed out the following and discussed them in detail:

1. Not Structuring Time for Game Playing

2. Placing Too Much Importance on Paper Designs

3. Peer Review Not Taken Seriously

4. Decision-Maker Picked for His Producer Skills

5. Not Taking Advantage of Placeholders

6. Allowing the Story to Control the Game Design

7. Not Giving Designers Enough Tools

8. Entering Production Without Something Fun

9. Not Keeping Design Documentation Up to Date

10. Not Making Outside Play tests Part of The Process

Website: <https://www.gamasutra.com/view/feature/132408/10_game_design_process_pitfalls.php>

# References

3D Game tutorial videos saved in personal playlist which found helpful throughout project

<https://www.youtube.com/playlist?list=PLm46bxzBw9LtspneRzpph4EVfLq5H7QN1>

Design Pitfalls

<https://www.gamasutra.com/view/feature/132408/10_game_design_process_pitfalls.php>

# Recommendations for Future Development

If we were to do this project again with the knowledge we now have, we would try to research more about the animations of the characters as animations like the attacks and idle movements, took a while for us to complete. If we had done more research, we are sure we could have found better, more comprehensive documentation and tutorials on how to design our own animations for the characters. We would also try a more detailed schedule/ project plan because, even though we stuck to it very well, we believe there is room for improvement for our schedule to ensure a high-quality game/product at the end of the development deadline.

# Conclusion

The type of game we have chosen is down to our own personal preferences and we believe it will be a good game to play for all with an addictive aspect making people want to keep playing.

We believe additional features may be added in time or during development. We also believe more features / requirements will lead to additional time needed and hopefully getting the basics up and working will already lead to a good game. Then to focus on additional features to improve what already would be a good game.

We believe looking at the game from now during the design phase we will have fun creating this game and some stressful moments if things don’t work out, but we think with the right amount of time spent at this game it should be achievable. Hopefully we can get the game up to a good standard and be able to add extra features to make the game even better for the user.

We hope finishing this project that it exceeds our expectations and if not, we will have learned a lesson going forward.

# Learning Outcomes for Stephen Gallagher

## Areas Learned on:

Movement to Player-

learned in this area where you can get the player to transition between different movements using a Boolean scenario. This where you can make a connection and apply a 1 or 0 condition. This which I found to be quite easy to understand and effective when it came to learn this area.

Terrain size -

In this area I didn't learn it the easy way as I had made my terrain far too big and after all my work when trying to reduce the terrain size I could only do it so much without taking away all my work on designing the games look.aka the buildings and mountains. This where I designed my houses and buildings locations around the terrain I had built. As when I went to reduce the terrain size, I soon became aware that it shrinks the mountains and everything making my game nearly unrecognisable. As all I wanted to do was make the base of the terrain smaller not the contents linked aka the mountains.

Terrain Shape -

What I learned with terrain shape is that it is quite difficult to shape a terrain to look realistic without practice as getting to know each brush that causes the terrain to go up and down at different shapes will come down to knowledge and practice. As when trying to build a mountain for instance picking a brush that is 100% pointy is not what you want. Then when trying to use another brush to flatten the point it can be tricky to not ruin your goal/shape intended. Practice and Time to know the different brushes and effects will build on your possible outcomes in this area I feel.

Painting Terrain -

Painting terrain can be tricky I found. As for instance the mountains in our 3d game I found not to have come out as good as I would have liked as when I did the terrain the terrain I used didn’t seem to look as realistic as I would have liked as well as having too much patterns. That when you look closely it’s like a duplicated image unlike what I did for the town. In the Town I put a terrain of cobble stones which if you look closely worked perfectly!.The cobbles linked perfectly looking quite realistic and satisfying to the eye when playing the game, just like the grass in other areas .The grass being all one colour made it easy to use unlike the mountains. I feel the trick with painting terrain is that the design of the terrain paint is vital. The more patterns and size image you use plays a big role.

#### Assets -

I found with assets that yes, they are easy to come across but when using the assets found on the asset store that you can easily download more than you need. Causing your game to jump in size, where this will be a vital problem if stuck for space. Choosing the right assets and not downloading too much can be better as 1. Doesn’t make game too big and 2. makes your game hierarchy more manageable with less items. Especially unnecessary items. If going forward I would look into selecting the vital items I would need to carry over to my project as I came across a problem that when I downloaded an asset from the asset store that there would be 20 items but when I drag 5 items into a specific folder I want to keep and then delete the rest. The items I wanted to delete, deletes perfectly but so does the items I have moved which I wanted to keep. This which I would investigate in more depth. As you can see in the project, I have two folder assets which I downloaded for the project to populate the game with. Items to make it look like a town/village. Here where there are some unused assets in folders but couldn't seem to find a way around it without deleted used items within the game.

Sounds -

I found with sounds that you must choose them wisely to fit in with the theme of your game. I came across the game sounds from YouTube. The In-game sound I found by looking for natural day time sounds. This where there was an hour clip of natural sounds with birds etc. I thought this fitted in perfectly as it was an outdoor game. The menu Sounds I also found on YouTube. This being an old medieval English song. The sword sound I came across online which was only a few seconds long. I edited the player script so that when you press down the attack button the sound of a sword will be played. This which was pleasing to hear when playing. The problem I faced here was that when my clips being the hour long in game natural sound was that it slowed down the loading on the game as it had to load up the very long file when launching into the game scene. This which I had to shorten down to about a minute which worked perfectly. Having these audio clips down to about a minute seemed to work flawlessly because I also had turned on the loop function on the side which unity makes possible for sound audio.

Menus -

When creating the menus I used my past knowledge of moving scenes using buttons from another module when creating a 2d game in unity .This which helped me hugely as I was able to make up the different scenes, main menu scene ,settings scene, end game scene, rules scene and move between them easily. This which I created a script that when a specific button is pressed it will load the scene I specified. This where I dragged the scene into the "level" section serialized field.

Objective text to screen -

I found that using triggers can be more effective in just moving a player. That when I used a trigger in this instance, I had text displayed to the screen for the user. This which allowed the user to see the objective on the bottom left hand side to the screen when passed through which I located at the start of the game which the user could not avoid. This trigger that I made could not be re-entered to redisplay.

#### Trigger to spawn enemy at location -

I used this technique as I found it brought a different element to the game as in areas of the game. for example, just before entering the town it looks like there is no enemies in your path but without the player knowing there is when the player goes through a trigger spawning an enemy at a specific location I had specified to be at the entrance. This which gives not only a shock/surprise to the user but also a look into the different aspects that the creator aka me went through to create a game that doesn't just have the same type of elements throughout.

Trigger to move scene -

I used a trigger to move scene as It was another area in which I found effective in surprising the user. This which I could have incorporated into entering houses. This where I could have made the user be able to enter a different scene when thinking they are entering a house. Making the game have a more real-life experience. Unfortunately, I didn't get to put this to its potential, but I did put it to use when entering the grounds of the castle. Here where when entering the grounds of the castle it prompts the player to a different scene locating the player within the grounds of the castle.

enemy AI that he follows you when you get near -

I learned that giving the enemy a basic AI script that you can get the enemy to follow the location of the player when you get near. Adding to the realism of the game. Making it more enjoyable as the player tries to not get into conflict with the enemy.

Objective -

for the Objective I found that reaching a destination would make the games possibilities much larger. Being able to have an open world game with no limitations. This where I could learn in any area, I come across that I might find useful for my game. I used my learning of switching scenes to my advantage I this area as I was able to use an object when touched prompt the user to a different scene. This making the object I choose to be the end game objective.

Populating objects into sections -

I found for the game that populating the game objects into other objects like folders that it was a great way in organising the hierarchy table of the game. I did this for each section of the game as I found it made items easier to locate. for instance, I had one for town, windmill, home and castle. Breaking the hierarchy into sections. Within theses sections I then had houses, fences and other objects within these.

## Areas I would like to focus on next time:

#### enemies/civilians being able to move on their own -

I would have liked to of had a look into making the enemies move on their own in a random pattern making it not just when approached that the enemy moves.

inventory system -

I would have liked to of looked into being able to give the player more capabilities ,that when they reach a specific item they would benefit and it would be stored .This which I did not focus on as I was trying to get a general game running.

graphics more -

I had a general look into graphics. That being that I could tell you that you can change the way the lighting of a game is done as it loads etc. But I didn't get to investigate this in detail. I have changed one or two in the game but This Which I am unaware of its true potential effects to the running on our game.

player & enemy design/outfits -

I would next time when creating a game look into the design of player & enemy outfits as I feel I didn't get to achieve in this game. As I would have liked for there to be civilians and enemies dressed differently. As for this game everyone is the same character which brings down the games potential unfortunately.

Terrain -

I would like to spend more time on the terrain next time as knowing the style brushes when making the terrain smooth etc can make some very nice landscapes etc for the user to play upon. You want the user to feel as if they are in the game amazed with the viewings they see before them.

general enemy civilian sounds when approached -

I would add movement sounds to the player and the enemies along with civilians too. This which can be seen in many games to date which plays a big role in games. The different sounds which can distinguish between a civilian or enemy player.

objectives -

next time look into games for more ideas on harder/unique game objectives as I find that the one I chose using a trigger was too simple ,yet it is still quite effective I would have liked to have set out a longer real life quest for the user.

# Learning Outcomes for Dylan Creaven

## Areas Learned on:

#### ***Pause Menu –***

I learned that using a canvas as a Pause Menu instead of a separate scene allows the player to go in-between the game and the pause menu with no loss of progress. I also learned how to active certain assets in the scene in a script to allow the canvas to disappear and appear with the press of a button (e.g. canvasUI.SetActive(true) will cause the canvas to be visible in-game)

Sound Toggle –

I learned how easy it was to set up a sound/music toggle box inside the pause menu. I then added our Music Player object to the toggle object and used the Music Player’s Audio Source method; bool mute. This way the player could toggle mute the sound as they so wished.

Enemy Movement -

I learned how to use assign the player as the ‘target’ and have the enemies move towards the ‘target’ position whenever the player was a certain distance away from the enemy.

Enemy Attacking -

I learned how to calculate the distance between the enemy and the player and have the enemy attack if within a certain distance. I learned how to use the animation bool my partner had set up so that if enemy is attacking the bool is set to 1 and the attacking animation starts and then when the animation ends, the bool is reset back to 0 again. I also learned how to attach an attack cooldown time to the enemy so that they could only attack a certain amount of times a second. This stopped the enemy from ‘attacking’ multiple times a second so that all the player’s health was depleted quickly.

Downloaded Assets management –

I learned that when downloading pre-made assets off the Unity Asset Store, you are usually downloading way more than what is needed. E.g we used a Nature Pack to add trees and bushes into the game but when we downloaded the Asset Pack, we realised that there was dozens of different types of trees and bushes and hedges etc. that we just didn’t need in our game. We had to prioritize and choose which flora looked the best in-game and fitted the setting of the game best. We choose 2/3 trees and bushes and then deleted the rest of them to avoid cluttering up the Unity hierarchy.

Character Stats -

I learned how to use assign game Objects (i.e. player and enemies) ‘Stats’ such as total health, the amount of damage they can deal out to other game objects and their armour which can lessen the amount of damage they take from being attacked. I learned how to clamp the damage to a certain value (i.e cannot go below 0) because the damage taken by an object is calculated by the damage taken minus the value of the armour of the object but if the value of the armour is *greater*  than the value of the damage then the damage becomes a minus number which means the health of the object (i.e. health = health - damage) is not going *down* when hit, it is actually going *up*. Clamping the damage to not go below 0 prevents this.

## Areas I would like to focus on next time:

#### ***Collisions between Enemy and Player -***

I would like to be able to learn how to detect a collision between the enemy’s sword and the player and vice-versa the player’s sword and the enemy. Right now, if an Enemy/Player attacks, they hit the other object if they are simply within a certain distance of the object and not if the sword hits the object. This means that the Player for example could hit the enemy even if the enemy is behind them.

Volume Slider -

Originally, I wanted to have a volume slider in the pause menu, but I was unable to get it working correctly, it either had the volume at full volume or just muted so I needed to change the slider into a toggle button. I would like to have more time with the slider to try and perfect it.

Rules in Pause Menu -

I would like to have tried to have the rules be another canvas within the pause menu that would have been activated when the Rules button was pressed because the Rules button right now brings the player to a separate scene, which causes the player to lose their progress made in-game.

Enemy design/skins -

I would have liked to have more variety in the enemy’s appearance as we just used the same assets as used for the player. Next time, I would like to have multiple different types of enemies with different attacks and difficulty and have them all look distinct to one another.

Different Armour -

Next time I would add a way to pick up and drop armour and have different types of armour give the player different stat boosts e.g. Armour goes up by 5 but damage dealt and movement speed goes down by 2 because the armour is heavy. Something to give the game variety and make the player think about what they want to equip going into a battle with enemies.