## 

# Computer Games Development CW208

# GDD and Project Report

# Year IV

## Dylan Curran

## C00217112

## 03/05/2020

## 

[**Computer Games Development CW208**](#_53ewi7wpg3ja) **1**

[**GDD and Project Report**](#_sbowdgo0u95y) **1**

[**Year IV**](#_53ewi7wpg3ja) **1**

[Dylan Curran](#_873rjmb5lxsc) 1

[C00217112](#_h0kkxbvl5rcy) 1

[03/05/2020](#_mz2vd86hpa2l) 1

[Project Abstract](#_u3n3li3uxf83) 2

[Project Introduction](#_b3b7xa7lm9z6) 3

[Project Description](#_c13cb05ze1ma) 3

[Overview of Game](#_s793ejvw56tu) 3

[Philosophy](#_frqus388ao2z) 3

[What is this game?](#_stmzaepy8hsh) 3

[Theme and Setting](#_v0mak2hcyx53) 3

[Why I made the game](#_8bhafwt7n42j) 3

[The Desired Player Experience](#_s3n6m0878y78) 4

[The Camera View](#_merj8i923yit) 4

[Gameplay](#_t5n63466uvq1) 4

[The First Minute](#_fyttxwq0tsv6) 4

[Game Progression](#_nxemppd0twgm) 4

[Victory Conditions](#_94shuja9fe2) 4

[Features](#_f5fin2syysf8) 4

[House Component](#_9aiwyschh30n) 4

[AI Component](#_1gqcflngl6i1) 5

[Player Component](#_wi9497w5fy3) 5

[NavMesh Component](#_i7vf9aigowls) 5

[Easy Level Design](#_o5s4xa8s1c9z) 5

[Collectible Component](#_rke9orlhiq6s) 5

[Game Engine](#_phf2jn8v0tis) 5

[Collision Detection](#_3j2tx1jsmnru) 6

[Project Milestones](#_4vpwlskmp62w) 6

[Milestone 1 December 18th](#_izcjv573pxok) 6

[Milestone 2: March 15th](#_vaokzqqz99z7) 6

[Milestone 3: April 20th](#_b1cio58q9npo) 6

[Milestone 4: April 27th](#_p4k3bthqhu6d) 6

[Project Review and Conclusions](#_kcx8uycf9ahq) 6

[What went right?](#_nsnubegz75gw) 6

[What went wrong?](#_m1u36dt9htgf) 7

[What is missing?](#_faqxrfrletce) 7

[What would I do differently?](#_bvio482j0lmp) 7

[Advice for people doing this in this future](#_1u7x5bw3dsex) 7

[Was it the right technology choice?](#_p8w3f7mfowbk) 7

## 

## Project Abstract

For this project I chose to look at AI Learning. The reason I chose this is because I felt that simulating an AI to make it look realistic would be a very difficult job, henceforth why I chose this. I specifically chose Q Learning as my Ai learning type due to the fact my AI would be searching for something and Q Learning gives positive and negative reinforcement for actions so I felt it was perfect for this project.

## Project Introduction

Ye Olde Escape is a 3D stealth strategy game where you (the player) must escape the house locked by the murderer hunting you in the house. Your aim is to gather collectibles before the AI hunts you down. This game could have potential to have many more levels created over time with an adaptive AI that will hunt the player. Any audience can play this game but it would be more suited to those who love stealth games.

## Project Description

For my technical learning achievements I learned how to successfully use a navmesh and adapt it to any environment. I also learned how to make a camera follow a player despite how the player and camera would not rotate in sync.

As for personal achievements I don’t really have any I’d mention off of the top of my head.

## Overview of Game

### Philosophy

What this game is trying to do is trying to make a simple game more difficult over time due to an adapting AI that I have never done before, thus expanding my own understanding for the future.

### What is this game?

This game is a 3D stealth game where the player must gather collectibles and avoid the AI hunting the player.

### Theme and Setting

The theme for this project is that it has a minimalist design as the objective is simple, just like the setting.

### Why I made the game

I made this game due to the fact I felt it would be the most relevant to AI learning as it would have to chase the player.

### The Desired Player Experience

The player should enjoy the game but the emphasis for this game will be on the implementation of the AI rather than the gameplay itself.

### The Camera View

The camera will follow the player as the view will be in first person from the player perspective for the game.

What is differentiating this from other games?

This game goes for a serious stealth game but with a minimalist design, almost like thomas was alone0 met a stealth game.

## Gameplay

### The First Minute

The player will spawn in a room and will have to figure out the basic movements of the player and be able to pick up collectibles.

### Game Progression

Over the course of playing the game the AI will begin to learn the patterns of the player and begin to optimize their decisions to make it harder for the player to win based on positive and negative feedback due to player movement.

### Victory Conditions

The win conditions for this game is being able to collect all the collectibles for as many levels as possible without being caught by the AI.

## Features

### House Component

The mechanics of the house component is that the house is used as a container for the player and the AI for the game to be carried out in. (medieval version no longer required due to quality change)

### AI Component

The Mechanics for the AI in this game will be that the AI over time will learn the habits of the player and adapt their playing strategy to make it more difficult for the player to win the game consecutively.

### Player Component

The player for this game will be keyboard controlled with the WASD keys for movement and rotation (WS to move AD to rotate left and right respectively).

### NavMesh Component

The navmesh component is designed to allow agents (objects dependent on the navmesh to move around) by detecting the area the agent objects can move around based on the obstacles the agent could face as they are navigating each level.

### Easy Level Design

There will be a prefab folder made in Unity that will contain instances of walls, floors, collectibles and new instances of Player and AI models.

### Collectible Component

The collectible component is used in the game as a win condition for the player, the player must collect a certain amount of the collectibles to pass the level, only the player can collide with and collect the collectibles.

## Game Engine

Unity is a game engine that makes design far simpler, it works with c~ code and is able to keep track of the position, rotation, and scale of objects, other things such as collisions and other things can be dropped onto the objects due to Unity’s simple design.

### Collision Detection

Unity has an OnCollisionEnter and OnCollisionExit function to deal with collisions, there is also a physics editor that allows specific objects not to collide with each other.

## Project Milestones

### Milestone 1 December 18th

The first level was designed with a player and a very basic Ai that could move around the house but not well.

### Milestone 2: March 15th

This milestone was having a better working level rather than the older level that was not very efficient.

### Milestone 3: April 20th

This milestone was having an updated player movement that would rotate and move in first person with the camera.

### Milestone 4: April 27th

This milestone is where I made the first tutorial level with obstacles for the player so the player would understand the basic movement system for the game.

## Project Review and Conclusions

### What went right?

I was happy with the ease of the level design as I could make another level in a couple of minutes, I also thought the player movement went well as the first person movement was more difficult than anticipated and I was able to get it working in the end. Lastly I was happy that I was able to effectively learn how to use and adapt a navmesh per level as I has unsure about the navmesh and how to work it all so it was a learning experience for me.

### What went wrong?

First and foremost the AI did not go near as well as planned due to an error I made with an array and as a result my AI didn’t actually work, nor was I able to test it. My initial level that I made never had collisions on it for some reason so I had to scrap the level due to the problems associated with that. I also had trouble pushing large files at the start due to me needing a large asset file for the level that I no longer used but that set me back a bit.

### What is missing?

What is missing from this project is a working AI due to the array problem. The completion of a level is missing due to the fact I only have a debug log saying you completed the level. I would add doors to the level too so they are missing too.

### What would I do differently?

If I was to do anything differently I would definitely work on the project earlier and adhere to a more scheduled layout rather than have lackadaisical layouts for time, if I kept up the motivation I could have had a much better project.

### Advice for people doing this in this future

Start with the AI first going to a stationary location, then create more locations and have the AI wander, after that, create a player that can move and have that AI try to find the player, after this maybe try creating some levels and see how the AI adapts and finally adhere to a good schedule or you will fall behind.

### Was it the right technology choice?

Yes, I felt Unity was the right choice due to it’s simple layout and nice UI, thus allowing me to spend more time on the important things I needed to spend my time on (if I did so). If doing this again I would still Have Unity as my technology choice without doubt.