De Montfort University

Annihilation Intelligence (A.I.)

A 3D 1st Person Horror Mobile Phone Game

By Simon Chiu

Computer Games Programming

P14133969

Table of Contents

Research

This is a focus on research for software to use, their pros and cons, and what is viable for this final year project.

To start, the software must be able to develop apps for an Android operating system, so they will need to be able to create an .apk file which can be installed on Android phones. The main software that comes to mind from my knowledge is: Android Studio, Visual Studio, Game Maker Studio and Unity.

Android Studio develops specifically for android phones, tablets and devices, and uses Java to do so. I have some experience in doing this from the Mobile Games module in second year; however, the main concern is using 3D in Android Studio, which was not explored in earlier modules.

Visual Studio would allow me to use C++, which I have learned since the beginning of my course. I am aware of functionality to develop for Android, and even for iOS and Windows phones. However, this has not been done in any modules for Visual Studio, so research would have to take place in how to do so, as well as any functionality for 3D.

Game Maker Studio use Game Maker Language (GML), a scripting language with similarities to Java, which can be used to create games for multiple platforms, and is friendly to novices in game making due to its drag and drop features. To release games in certain platforms require licenses, costing $299.99 for the Android export module. I already own this module, so there would be no further costs to develop with Game Maker Studio. The concern here is I do not have experience in using 3D in Game Maker, which will need to be researched.

Unity is a well-known games development program, using C# for coding. It is most often used for 3D games, and is also a popular choice for less experienced programmers. Personally I have little experience with Unity and C#, meaning I will have to research this programming language, as well as learn to use the Unity interface. However, Unity does allow for 3D games development, and can be exported to a mobile phone platform.

Before choosing which software and language to use, I researched each one to see what pros and cons each have, and which would be viable choices for my project.

Researching into Android Studio, without a game engine, 3D objects can be drawn using OpenGL ES, which uses a GLSurfaceView and draws objects to it. This is similar to the Shader Programming module, which also uses OpenGL in Visual Studio to draw 3D objects. Other methods include using a game engine such as j-PCT-AE (Foerster, n.d.). This is a free 3D solution which can be used in Android Studio or Eclipse, and offers shader support for OpenGL ES 2.0, working on Android 1.5 or higher.

With Visual Studio, you can build C++ apps for Android, iOS and Windows phones by installing the SDKs and tools for cross-platform development. A project can be created as an OpenGL ES 2 application, with a shared C++ code library to build an Android Native Activity app, and iOS app and a Windows app. Although it can use C++, some Java and xml is still necessary for developing an Android app.

Game Maker Studio is primarily for 2D games, but has the functionality for 3D games and effects. All 3D functions in Game Maker start with d3d (Draw 3D) and the beginning of their names, and allows us to start drawing in 3D, either drawing primitives, pre-defined shapes (such as cones or cylinders) and can also load models such as .obj files.

Unity, as a game engine, simplifies the use of 3D significantly, using drag and drop features to place objects within a scene. It also simplifies using materials and textures with the same drag and drop features. By simplifying these parts of game creation, I would be able to focus on programming the gameplay, focusing on movement, controls, etc. In the end, this may simplify the process too much, making this choice less desirable.

Each has their pros and cons for my project, and after reviewing each of them, I have decided to use Android Studio with j-PCT-AE for my project. In terms of developing for Android, I have the most experience with this software. Although I have not programmed 3D in Android Studio, j-PCT-AE will help me to draw 3D objects, and using Android Studio will allow me to use Java, as my other modules do not use Java. This will also allow me to research and use a method of game development which interests me.

Literature Review

This literature review will review and discuss many topics, including:

- Popular horror games, and why they are successful
- Horror as a genre and its popularity in other forms of media
- The market of phone games

These topics are all related to this Final Year project: a 1st Person 3D Horror Mobile Phone game. Horror games are popular for the tension and suspense they bring, not only in terms of story, but also how the player physically feels while playing them. They can be slow, atmospheric games designed to immerse the player and make them scared of what could happen, or can be fast-paced action games that use jump-scares to suddenly scare a player.

Horror is a popular genre for many platforms, from video games and film platforms to audiobooks and literature. With the intent to purposely instil fear into an individual, it can be broken down into many different popular elements, such as physical beings (including zombies, creatures and demons) to emotional and psychological horror. In video games, the Horror genre has been a popular genre for a long time, with many games generating sequels, and with many developers (including large companies and indie developers) trying their hand at creating a new game to scare their customers. Some of the best Horror games include the *Silent Hill* and *Resident Evil* series (members of the Survival Horror genre), to *Dead Space* and even the playable teaser *P.T* (2014) (GamesRadar Staff, 2016).

Survival Horror is a popular subgenre of video games. Most Horror games use Survival Horror to describe the game, but there is debate as to whether the meaning of Survival Horror has diminished, as early games such as *Resident Evil* (1996) marked Survival Horror as surviving with limited options or items and managing what items you have to defend yourself, whereas more recent games like *Dead Space* (2008) are more action-packed, testing an individual's ability to fight back under pressure. Action Horror may be the better subgenre choice, as Butler-Harvey states Survival Horror is used *"very lightly when it comes to labelling games"* (Butler-Hartley, 2013).

Sterling argues along the same lines, asserting *"the traditional survival horror is all but dead"* (Sterling, 2008). He explains how early Horror games used the limitations of the hardware to their advantage, namely *Resident Evil* and *Silent Hill* (1999) (which used fixed camera angles and somewhat awkward controls). Tokuro Fujiwara, one of the creators of *Resident Evil*, commented on creating a Horror game with hardware limitations, as well as potential ethical issues in what is described as *"a pioneer of the horror genre"* (GlitterBerri & Fuunoshin, 2012).

For this project, there are no ethical concerns in terms of creatures or monsters, but there are potential concerns depending on the design of the game. Although not finalised, the game may include NPCs (Non-player characters) who may die depending on the choices and actions of the player. Another potential concern is the fact the player we control can also die, depending on their choices. This opens up a discussion into what is acceptable in a piece of media which is available for the public. Compared to early Horror games and movies, modern Horror media is much more severe. This may be due to advances in technology to create almost life-like animations, or due to changes in the public's perspective from being desensitised to gore or violence.

That is not to say there aren't times where media is censored or changed to fit the public's tastes. In *Resident Evil 4* (2005), there are zombies who carry chainsaws, which can instantly kill the player by cutting their head off, and we see the player's headless body fall limp to the ground. *Resident Evil 5* (2009) changed this by turning the camera to a position which shows the body fall to the ground, but hides the player's head so we cannot see any gore. By dying in certain areas or positions where the camera struggles, we see the head is still attached, and the camera is used to give the effect of having the player's head cut off, without revolting the audience.

But even if many modern Horror games have strayed away from the traditional Survival Horror format, this does not mean they are necessarily bad. Returning to the list of top Horror games stated earlier (GamesRadar Staff, 2016), many of these are more action-based than earlier counterparts, including *Resident Evil 4*, directly stated in Sterling's article as *"effectively killing off the series as we knew it"*.

Horror games have adapted to an ever-evolving market and audience, producing faster, action-packed games which can be relatable to modern Horror or Thriller movies, which have become almost de-sensitised to gore and violence in comparison to older films. Simpler, survival-based Horror games have become increasingly rare in today's market, yet the genre is far from dead, as second on the list of top Horror games is *P.T.* Although cancelled, the demo showed how horrifying a game can be without the action-packed battle mechanics that Action Horror games show.

Regardless of how they can be sorted, they all fall under one category: Horror. But what makes them popular? Although classed as Horror, different games make use of different mechanics to either aid gameplay, or as a method of instilling fear. *Resident Evil* games have always used limited ammo and weaponry as a fear mechanic, making players scared of finding any creatures, not necessarily because they are out to kill you, but because you have limited supplies to defend yourself with.

Five Nights at Freddy's (2014) makes use of *'Horror vacui'* which can be described as *"the fear operable in off-screen space, the monstrous unseen that lies outside the frame and constantly threatens to appear within it."* (Soderman, 2015). As a night guard, you play confined within one room, with limited views of the area where animatronics are or where they are headed. The gameplay has the player constantly scared of animatronics which can suddenly appear, however this horror is not produced by necessarily seeing them, but knowing they exist in an area we cannot see easily (the player has access to security camera feeds, however, the vision is blurred and it is difficult to see some animatronics).

Horror vacui is often perceived in Horror games in smaller methods, such as what may lie behind a door or in another area. A well-known method of this is in the early *Resident Evil* games, where an animation for opening a door is played for every door, thus the player instantly enters the room and must examine the area immediately for any threats. There is fear created by what you can see in-game, but more fear is created by what you cannot see, of what could be hidden. Soderman has explained many examples of *Horror vacui*, including *Five Nights at Freddy's* and many others.

The fact that many Horror games make use of *Horror vacui* is no coincidence. It is a popular, if not the most popular part of any Horror game. If we could always see any monsters within a game, we may still be scared, but we have comfort in being able to see and keep an eye on these monsters. The fear of the unknown is not confined to only video games. Horror films have done this long before video games have. Even now if one were to imagine any Horror film, the characters (and subsequently the audience) are always afraid of what they cannot see.

As most anything related to Horror uses this concept, it is important to consider how this applies to this project. Being a 1st person game, where the player moves in an environment, *Horror vacui* will be in the form of what we cannot see in-game as the player explores. This can be as simple as hiding content in areas of the map, or with the use of shadows or limited views to weaken the player's vision. Although the whole story has not been planned out yet, the premise of this project will have the player trapped within an office building, where an AI which controls most of the building is trying to kill the player. In this case, the player is not escaping from a physical entity as such, but rather by an entity which controls seemingly ordinary objects. This can be seen as another form of *Horror vacui*, where what we are afraid of can be on-screen, yet we as the player are oblivious to it until the AI uses it.

Examples of this are prevalent in the latest of Scott Cawthorn's *Five Nights at Freddy's* series: *Sister Location* (2016). One example in this game is when the player must restart the power while in front of an animatronic. To do this, the player uses a panel and restarts each areas power by holding down a button. While doing this, danger levels show how likely the animatronic will attack and kill the player. When stopping, the player is able to see how close the animatronic is. To an extent this uses both *Horror vacui* to create fear when the player is restarting the power, but also fear of what we can see: the animatronic which has moved closer to us.

But can fear be created from a pure non-*Horror vacui* method? A game which does not use *Horror vacui*, yet still instils a sense of fear is the *Souls* series, specifically *Dark Souls* (2011) or *Bloodbourne* (2015). Not classed as Horror games, they are action-RPGs where a small mistake can have large consequences. Players of these games are not afraid of the monsters per se, as you have adequate weaponry and items to take on every creature you face, however, this game punishes you based on your skill. A small mistake can leave you vulnerable to enemy attacks, many of which can seriously injure or kill your character. Instead of fear created by what we cannot see or what may be chasing us, fear is created by what we can see, and knowing the consequences of being killed. In these series of games, death will cause you to drop souls, a commodity used as both currency, and as a method of strengthening your character.

Although they can be re-earned, to lose your souls is essentially wasting all the time spent fighting, as most enemies respawn upon death. You may have gained the knowledge of what you have fought, and what killed you (so you can change your tactics), but it must all be redone. This is similar to many early video games which did not have save mechanics. The original *Super Mario Bros.* (1985) on the *Nintendo Entertainment System* (NES) (1983) did not have any save mechanics, and the entire game had to be completed in one sitting (without the use of modern emulators). The frustration for many players from the difficulty has opened a new type of emotion which is linked to fear: Anger. Players who play Horror games often attribute a game with anger; hatred at losing or failing a game. Although they may not know this while afraid of creatures during gameplay, they do have the inherent fear of losing or failing the game and being forced to re-experience their current fear to continue the game. This is not necessarily a planned or desired mechanic in a game, yet is an interesting view of fear for a gamer.

Besides the act of inducing fear into an individual through a video game, we want the player to be interested in the game; for the game to have certain mechanics or aspects to intrigue the player. These can be described as agency mechanics, mechanics which produce a particular effect, in this case for the player (Habel & Kooyman, 2014). Habel & Kooyman discuss in great detail the mechanics and gameplay design in survival horror video games, using *Dead Space* as their primary subject of discussion, but also into Horror films and other Horror games. They show how they are both closely linked, stating *"numerous horror films and franchises have been developed into games, and Hollywood has in turn adapted horror games to film"*.

The connection between Horror games and Horror films is prominent, and is in part inspiration for this project: to explore a more atmospheric or relaxed Horror game. There are not any planned action aspects as such for this project, and there aren't methods of "fighting back". Instead the player goes through a linear story which involves puzzle solving to reach the end of the game. Developing for a mobile platform, this project is also to test the pros and cons of mobile development for a traditionally console-platform genre, as well as how well it fairs as a game that players have access to without the need of dedicated gaming consoles.

To understand why this project is being developed for mobile phones, it is important to understand the popularity of phones themselves. For mobile gaming, consoles exist as stand-alone handheld gaming devices made specifically for gaming. Examples include the *Nintendo Dual Screen* (DS) (2004), which follows the success of Nintendo's earlier handheld devices, and even has multiple versions. These redesigns were made to be slimmer, and even added new functionality, such as a camera. The success of this has brought about the *Nintendo 3DS* (2011) and now the *Nintendo Switch*, with an expected release date around March 2017.

Despite the rise of smart-phone games, Nintendo has managed to stay on top of the handheld gaming market through their innovations in handheld gaming (most notably the *Nintendo DS's* dual screens, the *Nintendo 3DS's* use of 3D, and now *Nintendo Switch's* detachable and attachable parts). This innovation has kept them the most prominent figure in portable gaming. However, this has not blind-sighted Nintendo from the rise of smartphone games in recent years.

Before Satoru Iwata, the then CEO of Nintendo, passed away in 2015, he revealed his plans to begin development of Nintendo games on smartphones, a surprise decision for fans due to Nintendo's reluctance to release games on other systems in the past (Peckham, 2015). Part of this interview includes the demand of types of games on different systems. *"On smart devices, the main demand is for very accessible games which smart device users can easily start and easily finish"*. In relation to this project, the game to be created is not designed to be very long or difficult to play, and is designed to be a short game. With this in mind, it would fare better as a smartphone game rather than on dedicated video game consoles or computers, where the player wants to generally play for a long period.

That is not to say it cannot fare well as a computer game. A prime example of this is *Five Nights at Freddy's*, which was originally released as a computer game, but was ported to Android and iOS shortly after. The success of this game (both computer and smartphone ports) has also led to smartphone ports of other games in the series, including *Five Nights at Freddy's: Sister Location*, with an expected smart phone release in December 2016.

Considering the success of handheld gaming devices for gaming, why not create a game for a dedicated handheld gaming console? There are a number of obstacles to overcome to develop games for portable consoles. From a technical standpoint, you need access to software to develop for a portable console, as well as the knowledge of the programming language associated with that software. Specifically for the *Nintendo 3DS*, one would have to fill out a developer application with Nintendo, and wait for further details from Nintendo. One of the tools Nintendo allows is Unity to create *Nintendo 3DS* games, and registering as a developer with Nintendo will give you access to a specialised Unity development kit to create 2D or 3D games (at a price). These games can then be released through the *Nintendo eShop* (2011), an online digital distribution service for Nintendo consoles.

Not only is there a waiting period before beginning development, there is also a cost to begin development. Another obstacle is the popularity of portable gaming consoles. Regardless of Nintendo's original reluctance to begin developing games for smartphones, they have decided to begin developing for smartphones due to their rising popularity. Other video game companies are also pressured by smartphone gaming, such as Sony. Their most notable portable gaming consoles are the *PlayStation Portable* (PSP) (2004) and the *PlayStation Vita* (PS Vita) (2012).

The *PSP* was *"aimed at slightly older gamers and offers more than just games"*, with Wi-Fi capability, music playback and full-length movies (Palmer, 2005). These movies used Sony's Universal Media Disk (UMD) format, and were a popular add-on for the *PSP*. Despite this, concerns for the *PSP's* lasting popularity surfaced before it was even released worldwide. Palmer stated *"sales could be eroded within years if mobile phone operators move into the games console market"*, referring to how smartphones have already taken up other devices, namely PDA devices and digital cameras.

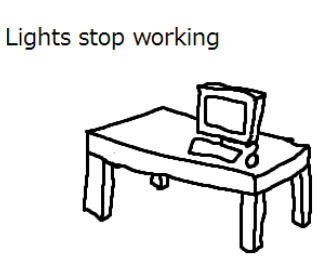
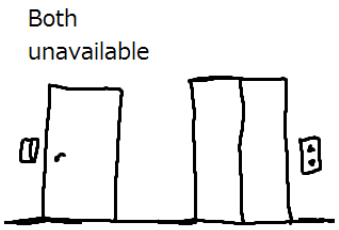
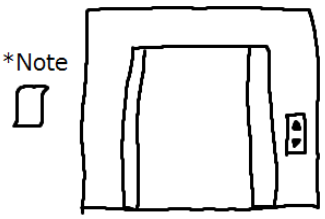
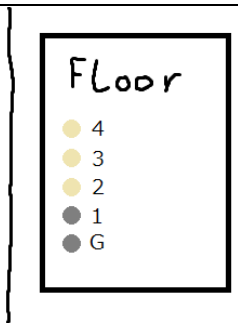
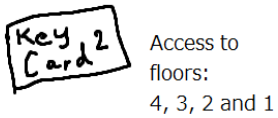
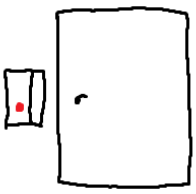
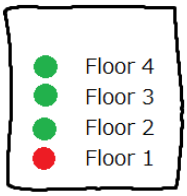
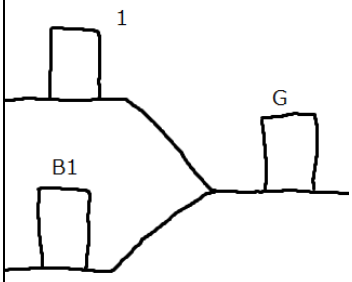
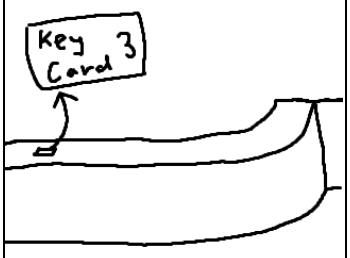
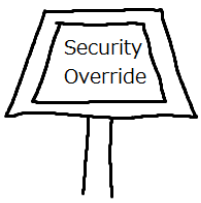
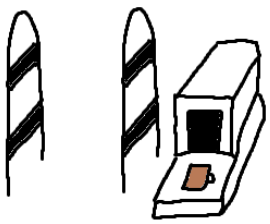
Today's smartphones have in-built, high quality cameras, and contain many apps to function as a PDA, on top of its other functions. Phones have been incorporating existing technology from other devices to create a single device capable of many functions. Even the *PSP's* full-length movie playback feature can now be done on smartphones. To truly put into perspective how powerful modern phones are, they are capable of running *PSP* and *DS* emulators. All of the functionality of last-gen portable consoles are available as apps to download.

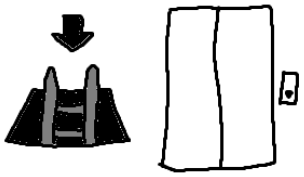
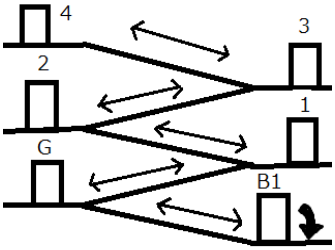
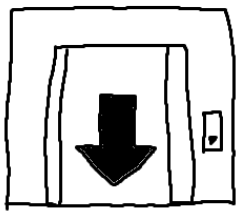
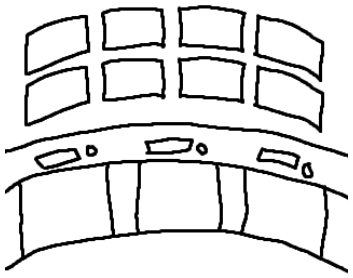
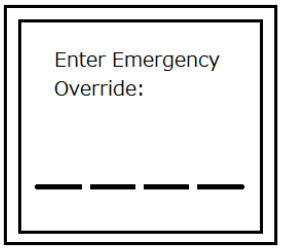
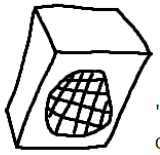
Palmer's article brings a concern for many companies with a smartphone's potential to amalgamate other software, and the availability of this software for people. As well as being a PDA, camera and portable gaming device, smartphones are already contesting with virtual reality, a technology which is still relatively new. A smartphone as a device for virtual reality has made the technology more accessible for the general public. Without smartphones, consumers would have to buy hardware specifically for virtual reality, costing several hundred pounds. Example of these are the *PlayStation VR* (2016) RRP £349.99, and the *HTC Vive* (2016) RRP £759.00.

With the advancements in mobile phone technology, developing software/apps for mobile phones is a popular choice for both indie developers, and for larger companies. It is a cheap platform for consumers to use, and the increase in smartphone availability to the general public makes it popular for developers to distribute their products. Because of this, this project will be developed as a smartphone game, targeting the Android OS, due to the higher number of users and availability of software, as well as the ease of releasing stand-alone games (as an .apk file, which can be distributed as a file or through a digital distribution service).

To conclude, the Horror genre is not simply games which jump at you to produce cheap thrills. True Horror games make use of atmosphere, story, and well-designed gameplay mechanics to truly instil fear into a player. This project will try to recreate the success of Horror games, while also testing new ground, such as creating fear from a non-physical enemy using inanimate objects to kill the player, the use of atmosphere and story as design elements to capture the player's curiosity, and testing the use of smartphones as a console for a Horror game.

Story Board

You wake up at your desk, where the electricity suddenly fails, and you begin exploring the area	The elevator is not functional, and the stairway is locked, requiring your key card at your desk	You get your key card, and enter the stairway, where the 1st and 2nd floor is locked, leaving only the 4th floor	While exploring the 4th floor, the elevator suddenly opens, and you see a note warning the use of the elevator
			
Unable to find anything, you return to the 3rd floor, where the elevator is operational, and you call the elevator for use	The 1st and Ground floor are unavailable, you go to the 2nd floor, where you find a key card to access the 1st floor	On the 1st floor is a second stairway, which is locked due to an electrical error, and the circuit board must be reset	Between the 1st and 2nd floor, you fix the circuitry and unlock the door to the second stairway
			
The second stairway leads to the Ground floor and B1, which is locked. You head to the Ground floor	On the Ground floor at the reception is a key card, and you head back to the staircase to enter B1	B1 is lined with small rooms, one which contains a security override panel, locked by a puzzle	Completing the puzzle, the security system is reset, and beeping can be heard from the scanners above, with an unknown luggage. You take it and hear a sound downstairs
			

You head back down and find a latch which leads to B2. You find another elevator, however the electrics are offline	Looking at the clues to fix the circuitry, you go and collect items and clues from previous floors	After collecting all the necessary items, you return and fix the electrics, and take the elevator far down	You enter an underground facility, and enter the main room , filled with monitors
			
You hear an unfamiliar voice over the speakers, and after speaking to you, the AI begins the self-destruct process	Using the luggage from earlier, you get a USB and get the code for the emergency override by completing puzzles	After completing the puzzles, you enter the override code, and you begin to leave, hearing scientists discuss the AI over the speakers	You leave through the main doors, escaping from the building
			

Story

You play as Cole Johnson, an office worker who works on the third floor, who wakes up after falling asleep while working late, when the electricity suddenly starts flickering. You try to ignore it at first until all the electrics suddenly seem to stop. The backup lights activate, but are weak and dim. At this point, you stand and look around, and start to hear the AI speaking. At first it states there is a power outage, and to carefully leave the building, but static builds and suddenly it starts speaking in a darker tone. Instead of saying to carefully leave the building, it now says you cannot leave this building.

You walk around the current floor, the third floor, mostly made up of office cubicles including your own. You reach the elevator, but cannot operate it, so instead you try to use the stairs. When you reach the stairs, it cannot open as it requires key card access, which you left at your desk. After heading back and getting it, you get to the stairs and try to go down. It goes down two floors, and up one, but both the bottom two are locked. As a worker who works on a certain floor, your key card does not have access to the doors. You go upstairs and into the next floor, where the elevator suddenly opens. There is no AI voice or anything, only the sound effects of the elevator arriving. Nearby is a note on the wall, which reads: "Warning: Take care when using the elevator. It is controlled by the AI which controls when the elevator moves or stops in an emergency. Do not use unless called for."

Avoiding using the elevator which seems dangerous, you head back downstairs, where the elevator now seems to be working. You call the elevator, and it comes down normally, both with sound effects and the AIs voice explaining there is an elevator going down. You enter and use it to go down. The Ground floor is greyed out, as is the First floor. If pressed, AI will say "There is an obstruction on the (Ground or first) floor. Please select another floor". Going to the Second floor, you arrive and head out normally.

On this floor, you explore the area and find a key card which can open the staircase and the door to the First floor as well, and you find there is another staircase which can reach the ground floor. You try to use it, getting an error as there is a problem with the circuitry. Upon entering the circuit board area, there are loose wires, and sparks sometimes flicker from the loose sockets. There is a note stating "WARNING, misuse of electronic circuitry can lead to injury or death. In the case of an emergency, the AI should deactivate all electrics. Electrics can be manually disabled." Heeding the warning, you go and manually deactivate the circuitry, finding a screwdriver and using it at the master circuit box. Inside, all the switches are set to off except for the current floor, which is set to on. You turn this off, which turns off the lights for a moment before dim emergency lights activate. You return to the first circuit board, where there are no longer random sparks or noises from loose circuitry. You plug back in the wires, and can now enter the second set of stairs to reach the Ground floor, as well as B1. Ground floor does not have a lock, whereas B1 is locked.

On the Ground floor, there is the reception and the main doors to exit from. Before reaching the doors, there are barricades used to detect foreign items, such as metal or weapons, and a side computer which processes baggage. Getting close, you hear the AI say “Please remove all metals and weaponry before passing through the scanners. All offenders will be pros- (in dark voice) KILLED (back to normal) –ecuted”.

At the reception is a key card for B1. You head down to an underground area with a tight corridor lined with side rooms. Some are janitorial or electrical closets, but inside one you find the security override panel. You turn it on, and a puzzle appears on the screen. Upon completion, the front door security is reset, and there is a sudden, continuous beeping. You return to the Ground floor, where the alarm is beeping as the scanners have detected something. On the screen which is now on, it shows some luggage inside the scanner which has a metal object in it. Upon taking out the luggage, the alarm stops.

Once you stop the alarm beeping, you hear a switch activate downstairs. Heading back to B1, you find one of the small rooms has a small hatch. It can be opened, showing ladders to B2. After heading down, you enter a small basement area filled with electrics which are all connected and appear to be off. There is an elevator at the end, which is inoperable at the moment. Scattered around the room are small plugs, which can be fitted into the nearby circuit board. They are of different colours, and the open plug holes have numbers on them. There is a note telling the player of one of the positions, however, to find the rest, the note says to go to certain people’s desks on different floors to find the answer. You head to the specified desks, each time finding a note with the next place, a colour linked with a number, and a small diary entry of the person relating to the AI and strange occurrences. As you find these notes, the lights flicker, and the AI talks in a damaged, malfunctioned voice. Upon looking at all the notes and knowing which ones go where, you return and plug them in. When done, the machinery lights up, and the elevator powers on.

Entering the elevator takes you far down. While traveling down, the AI talks, switching between its normal voice, and a dark, broken voice, explaining the warnings of going to the secret underground base, with the deeper voice telling you "you shouldn’t be here". Upon stopping, you enter a well-lit laboratory. A main corridor with side rooms equipped with one-way mirrors show experimental machinery, some even with people inside who appear to be dead. All the rooms on the side are locked, and some have monitors explaining the patient’s vitals, all set as deceased. The final door at the end leads to the main hub, filled with security monitors on all the patients.

Here, the speakers turn on, this time to an unfamiliar voice, a human voice.

“I’m impressed you came down here, honestly I am. Why wouldn’t you simply leave? Especially considering the danger you’re in? A curious one aren’t you. Well, they say curiosity killed the cat.”

The door behind you locks.

“I can tell you want answers. Why the AI has suddenly turned, why there’s an underground base, what sick experiments have been held down here… ehehehe I bet you really wanna know huh.”

The lights and monitors begin to flicker, showing faces on the monitors, sometimes showing skeletal face, and sometimes security feeds of other rooms and floors.

“Well I’ll tell you. The AI for this building was never designed to simply help manage daily tasks in an office. No… it was designed for a much greater purpose. This is only the testing ground for the AI… with you as the victim! Imagine: an AI which can take over a network, gather information on the network, then proceed to kill people with the very network they use! Think of the military applications! Still think this is all preposterous? I’m sure the AI will begin the self-destruct process of this building very soon. Why is there a self-destruct for an office building? Like I said: This building is but a testing ground. And you the victim!”

The AI begins to talk, activating the self-destruct system. With the AI talking, a pillar arises with a keypad and screen. The AI states there is an emergency stop for the self-destruct, by entering the override code into the pillar.

“That would be the normal functionality of the building kicking in again! You’ve been lucky so far, but I doubt you can stop it!”

A side monitor lights up, and you use the unknown box from earlier at the monitor, which contains a USB drive, containing a final puzzle which produces the emergency override code. You complete the puzzle, getting the code to stop the self-destruct, and stopping the buildings destruction. As lights go return to normal and the door opens, you begin to leave. While you do, you hear unknown scientists talking, but not at you.

“Well, despite the success we’ve seen in certain areas, this one’s a failure.”

“Perhaps it’s the environment. An artificially designed office building isn’t a normal scenario.”

“Yes, but everything was in place for the AI, what happened?”

“This AI is near perfect; its success in other scenarios proves so!”

“But not here… maybe it is just the environment, but what if it’s the AI?”

“Preposterous, you designed the AI, what could’ve gone wrong?”

“Well… I didn’t exactly design it myself. I gave it its core functionality, but the rest was designed… by the AI itself”

“Are you saying the AI decided to let this scenario end in a failure?”

“Maybe it’s already got bigger plans. Plans we can’t even begin to dream off”.

You leave the building, escaping the horror of the mysterious scientists and the killer AI.

Game Design Document

Game Overview

Annihilation Intelligence is a 3D 1st Person Horror game, being developed for Android phones. You play as Cole Johnson, an office worker who has become trapped in an office building. The building is normally run by a single AI, which controls most of the machinery in the building. This includes control of the elevators, lights, doors, as well as various other machinery. This AI has suddenly malfunctioned and is now trying to kill you. You must escape from the AI by collecting items and solving puzzles to safely leave the building.

Instead of conventional Horror games where there is a creature or person after you, this game doesn't chase the player as such, but instead tries to kill the player with various machinery in the building. An example is the elevators, which the AI controls. If the player enters the elevators, they will suddenly close and kill the player. To escape, the player must find ways around the AIs programming, for example with the elevators, they can go to a different floor to call for the elevator, and the AI is forced to follow the programming to send an elevator up to the player, which won't suddenly close and kill the player.

The game will feature the use of 3D to create an environment, and will use touch-screen controls to play the game, in the form of virtual joysticks and buttons for the payer to interact with the game. It will feature standard smart phone app software design, such as functionality when the user receives a call or closes the app.

Common questions

What is the game?

Annihilation Intelligence is a Horror mobile phone game, where you play as an office worker in 1st person who must escape from a building where the AI is trying to kill you. You must collect items and solve puzzles to navigate your way safely out of the building.

Why create this game?

This game is a chance for me to create a game in an area which I am both interested in, yet have small experience in, allowing me to learn different areas of game creation, as well as show my capabilities as a programmer.

Where does the game take place?

Set in an office building, there isn't a certain time period as such; however, it would be the 21st century, where the use of an AI to control machinery in a building is tested, but not the norm. The gameplay will only show the player within an office building, and the various floors and rooms of an office building.

What do I control?

Your control Cole Johnson, an office worker who was working late into the night, when the AI suddenly begins to malfunction, and is now set on killing you. Without the skills (or the need) to use weaponry, you must carefully navigate yourself through the building, using items and solving puzzles to evade the AIs attempts to kill you.

What are the goals and objectives of the game?

The objective of the game is to escape the building. You will need to navigate the player safely to find items and solve puzzles which will lead to you being able to escape.

What platform will this game be available on?

This game has been designed as an Android game, meaning it will be available on Android phones. With the final product being an .apk file (which can be run and installed on Android phones) this can be downloaded, or distributed via a digital distribution service, such as the Google Play Store.

Functional Requirements

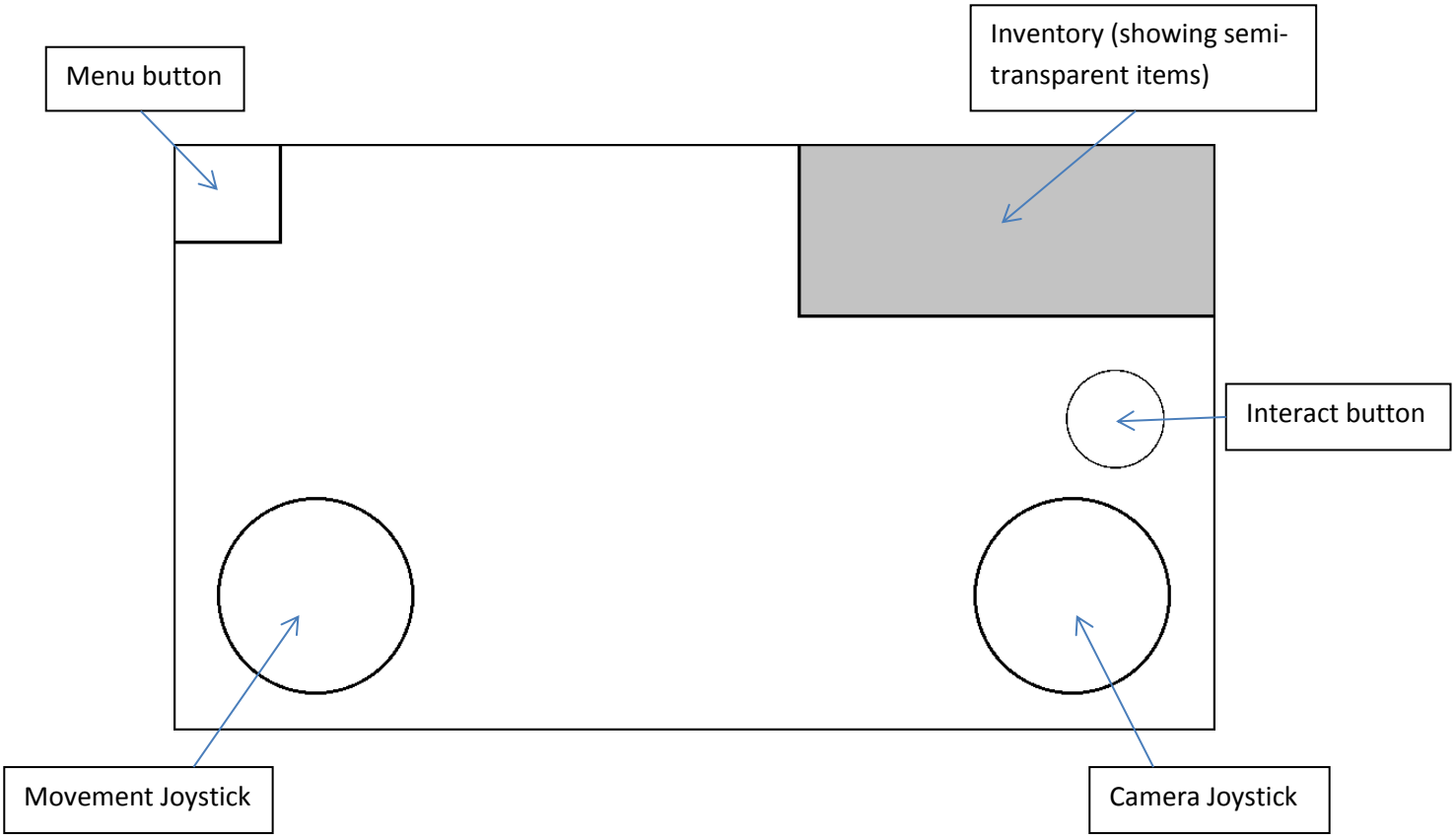
Controls

Movement – Virtual joystick, situated in the bottom left corner. Based on thumb use, it will find the angle your thumb is from the start position, and moves the character in that direction.

Camera – Virtual joystick, situated in the bottom right corner. Similar to above, the angle your thumb is from the start position, and distance, will turn the camera in pitch and yaw.

Menu button – A button in the top left corner, can be used to pause the game, and access the pause menu to return to title or quit. Users can still quit by closing the app manually. Leaving the app during play will bring this screen up.

Interact button – A button on top of the camera joystick, using this allows you to interact with an object you are looking at. This uses the centre of where you are looking, and sees the distance to the nearest object, and if it can be interacted with. This involves using objects, collecting objects



GUI

Main Menu – The main menu will be a simple image or camera of a 3D environment, showing off an area of the game. This could be expanded upon to be fancier, such as a rotating camera of the area, or fading to other parts of the map, but this is extra and can be kept simple. From here there will be a button to start the game. As the game is made to be played through in one sitting, there are no options for loading or continuing an existing game. Options will be selectable, and an Exit button to end the game. The game can still be ended manually.

Options – Another simple screen, which lets the user change certain options or controls for the game. They can reset these to default, or manually set them before returning to the main menu.

Game Mechanics

This project is a 3D 1st person Horror game, which will include the creation of a game engine to develop the game. This involves creating a 3D scene to load multiple objects as the environment for the player to move around in. The objects must be textured or coloured and appropriate lighting will be used on these objects.

The player will control a character in 1st person, therefore an object to represent the character isn't necessary, but rather a moveable camera will be used, which must be able to move in the 3D environment, based on the player's inputs, as well as rotate to look around the player (again, based on the player's inputs).

Part of the 3D engine will also require collisions between the player and the environment. To do this, an invisible bounding box for the player (the camera) can be created to test collisions with the walls, floor and objects.

To interact with certain objects in-game, a check must be made to see if the player is close enough to the object, as well as whether they are facing the object. To check if an object is close enough, a larger bounding box could be used to detect objects (which can be interacted with) within a certain distance; however, there are problems when there are multiple objects near the player. Instead of this, a single ray can be cast (through the centre of the camera) straight forward until it collides with at object. If the collided object is within a certain distance, and is usable by the player, then it can be used or picked up.

The game contains parts where objects may move, such as an elevator door opening. To do this, objects can be translated to give the appearance of animation, such as functions containing a translation over time to simulate moving doors. For this project, there shouldn't be any need for animation files for complex objects.

Characters

Cole Johnson: The main character the player controls, an office worker who works late one night, and is the target of the AI the office building. You play as this character in 1st person.

The AI: Originally an AI which controls the whole office building for day to day activities, something happens which changes the AI, and now it is set on killing the player.

Unknown Scientist: The cause of the AIs sudden change, he, along with a group of other scientists, have been experimenting with the use of AI as a method of gathering information and killing people, weaponising a non-physical entity.

Objects

Floors and Walls – Either a cube box to show the whole room, or could be planes for each side.

Chairs – Simple office chairs around the building. Can also have different types of chairs depending on the area, such as lounge chairs in a reception.

Tables – Mostly office tables, they'll be around the whole building, accompanied by chairs or computers. Could have coffee tables for say the reception.

Computers – Consists of a monitor, keyboard, mouse and computer, could be one whole model with the table.

Doors – To be placed on walls, could either be part of the wall, and when opened will load a new map (like in *Resident Evil*), or placed on a wall where there is a hole, and can be opened to enter the area.

Elevators – The elevator does not need to properly move or be animated, as long as it can open and close, screen shake or vibration can be added to simulate using the elevator.

Stairs – Can be designed as the whole area, stairs to go up and down, with a door on each level.

Keypad – Either for key cards, passwords or both, these will be used for locked doors or elevators.

Lights – Office lights which will be on the ceiling, some smaller lights for certain areas of the map, such as emergency lights in certain rooms.

Decoration – Extras to decorate the map. Includes plants, pictures, etc.

There are other more specific objects which are required, such as items the player will collect and use. The above are simple objects, which can be used to show the core features of the game, namely the game engine, drawing the scene and objects, as well as collision detection with the player.

Technical Design Document

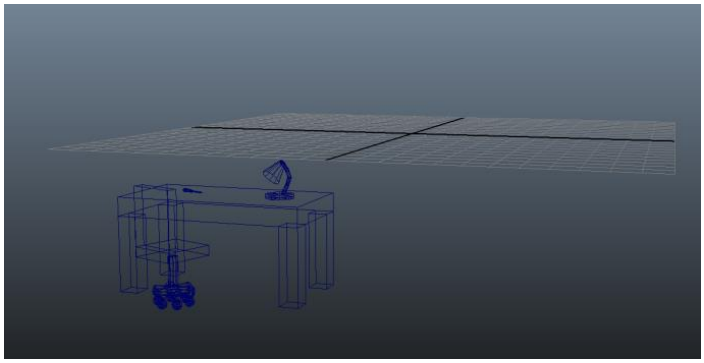
For this project, the software will be developed using Android Studio with j-PCT-AE, developed in Java. The development will consist of creating a 3D game engine for a smartphone platform, and developing the game from it, as well as setting up for mobile phone events, such as incoming calls, the screen turned off, or changing to another app. This will finish with an .apk file, which can be installed on Android phones, and can be distributed.

To create the game, assets will be required both for testing, and for the final version. 3D Models can be either created by me, using Autodesk Maya, or sourced online, using royalty-free/copyright-free models and textures. Music and sound effects can be either created by me, using FL Studio, or again sourced online. Any voices, which may be added to the final version, will be either by me, or by friends and relatives (who will be noted in the credits).

Prototype

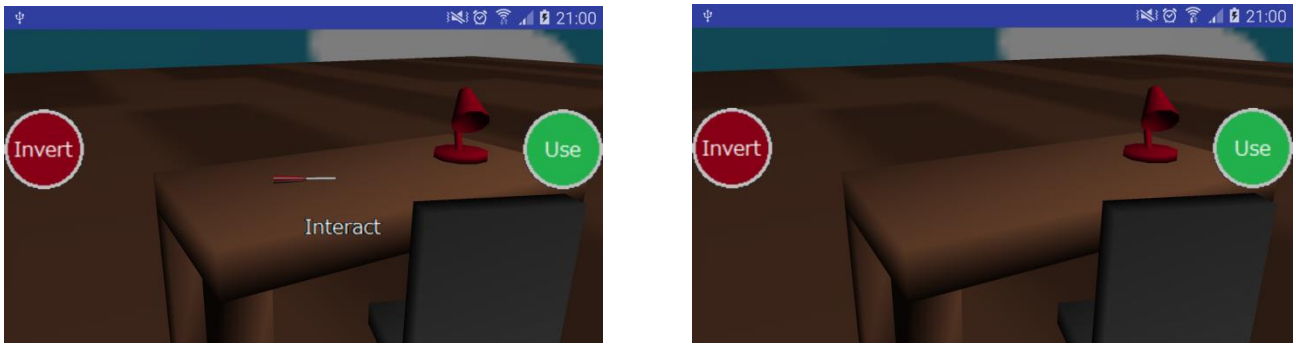
As a proof of concept, the prototype will mostly show the 3D game engine, showing the basics of a 1st person 3D phone game. This includes the creation of a 3D environment (made up of several objects), colouring/texturing, a camera, and the use of translations and rotations to view the 3D environment. To start, I created objects to the screen by loading in .obj models and image textures to render in Android Studio, and a camera to view the objects. For moving the camera, the program takes in inputs from the smartphones touch screen, using the player's thumb/finger to rotate or move the camera. By either using the left or right hand side of the screen, we can move and rotate the camera, using vectors for translations and setting up axis to rotate the camera by. Movement is also made relative to the camera's direction. I loaded a simple skybox and set the camera within it

After testing that objects can be loaded, I created some basic models using Autodesk Maya, with simple textures to test loading multiple objects, as well as loading scene by using the translations and rotations from Maya for each object in my game. Comparing the two screen shots, we can see a scene can be created in Maya, and recreated in my game.



Comparison of a scene in Maya (left) converted and drawn in Android Studio (right). Note the objects are drawn below the origin in Maya. This is to simulate viewing the objects from a standing person's perspective

Buttons were created to test player inputs, logging a message to Android Studio whenever a button is pressed. These can then be used for various parts of the game, such as interacting with certain objects or for setting up options. As part of the game, the player is able to interact with certain objects in order to pick them up or to solve puzzles. Using j-PCT-AE, a ray is cast in front of the camera, and this returns the distance to an object. If the object is one we can interact with, and we are close enough to it, we can interact with it. For the prototype, a log message was set as proof of concept, and as an example of how it can be used in game, a button can be pressed to pick up the object, with a small message appearing if the player can interact with the object. As an example of changing options, a second button was created, which simply inverts the camera axis.



Example of object interaction. The player can interact with the screwdriver (left) picking up the item for later use (right)

So far the prototype shows off many of the core features for the final product. Part of the final submission will include cleaning up the current code, as well as setting up activities for different parts of the game.

Development for the final submission will include:

- Cleaning up the code and setting up classes – As well as making the project more manageable, it will allow easier access for other programmers to understand the code, and will increase maintainability in the long run
- Creating activities for different parts of the project – Including but not limited to: A Splash Screen Activity, a Main Menu Activity, an Options Activity, as well as setting up the activity flow between them
- Music and Sound Effects – Sourced and used throughout the game, and editable in the Options
- Pause Menu – For use while playing the game, and usable when a player receives a call or closes the app briefly
- Device compatibility – Creating code which is relative to a devices screen size and aspect ratio

Test Plan

In terms of testing, the main areas are the mathematics, which consists of the physics of the 3D engine and the controls the user uses in the game, the flow of the game, such as moving from one area to another properly, as well as general tests a smartphone app would need to pass, such as usage of a smartphones other properties while playing the game.

These tests can be split between black box testing and white box testing. Black box testing is a method of testing which looks at the input variables, and tests if the output variables are correct by comparing them to known answers. For example, to calculate the angle our player moves in-game, our inputs would be the movement of a virtual joystick to the left (which we know should move the player left) and seeing if the player moves in the correct direction. If they do, then the test is considered complete, even if we do not know how it does it.

White box testing looks deep in the code to test if a program works properly. Taking the example earlier with player movement, we would check the code and run through, mathematically, what our inputs are producing, and whether the output is correct. We do not necessarily check if the program will give the expected output, but rather we can see that the output should be correct from the code.

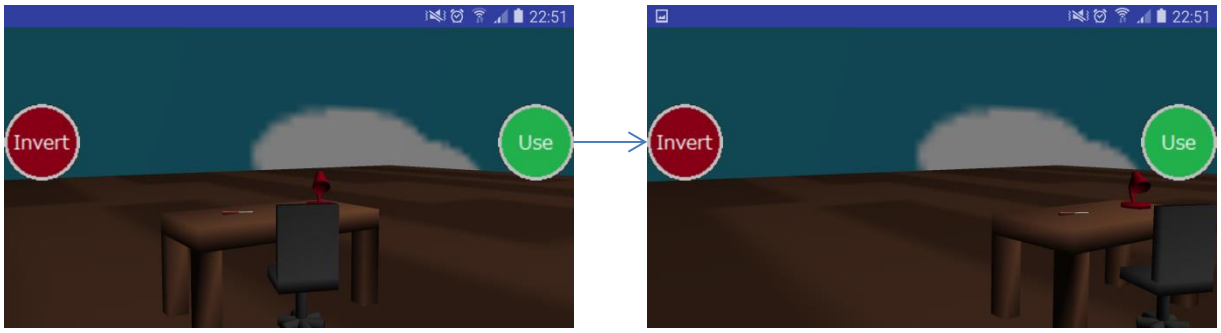
Between the two tests, black box testing is better suited to independent testers, whereas white box testing requires the creator or someone with understanding of the program and language to test them. For this prototype of the project, certain functionality will be required and tested to show proof of concept of the project, showing the basic functionality of a 3D game engine, which can then be expanded upon for the final submission

Test Cases

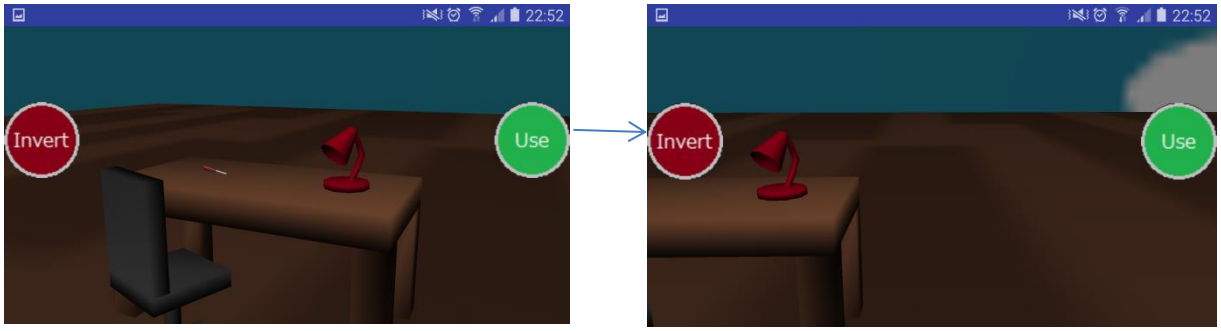
Below are test cases which have been used for the prototype of this project.

Test Case	Type of Test	Inputs	Expected Output	Observed Output	Comments
Character Movement	Black Box Test	Virtual Joystick Left	The camera will translate left to simulate movement	The camera moves left, making other objects in the scene appear to move right	Besides simply translating the camera by a speed, it also takes into account the direction the camera is facing, so it will always move 'left' of where we are facing
Camera Rotation	Black Box Test	Virtual Joystick Right	The camera will rotate right	The camera rotates right, rotating around its own Y-axis	Based on the amount we move the virtual joystick by, it will turn at a proportional speed
Ray to Bounding Box	Black Box Test	N/A	When we move close enough to an object, as well as have the camera face the object, the log should get a message	When we are in range of an object as well as facing it, the log receives a message	To show this in the program, text is creating ("Interact") to show we are in range
Button Press	Black Box Test	Virtual Button pressed	The program should recognise the input, and will show this with a log message	When pressed, the log is sent a message confirming the button has been pressed	To show this in the program, it is used in conjunction with the Ray to Bounding Box test to 'pick up' the screwdriver
Invert X-axis and Y-axis for camera rotation	Black Box Test	Virtual Button pressed, Virtual Joystick Right	As a test to show how we can add options for the player, pressing this button should invert the X-axis and Y-axis for the camera	Instead of rotating to our right with the virtual joystick, the camera will now rotate left instead	By using a simple Boolean which can switch from true to false or vice versa, we can press the button again to re-invert the axis

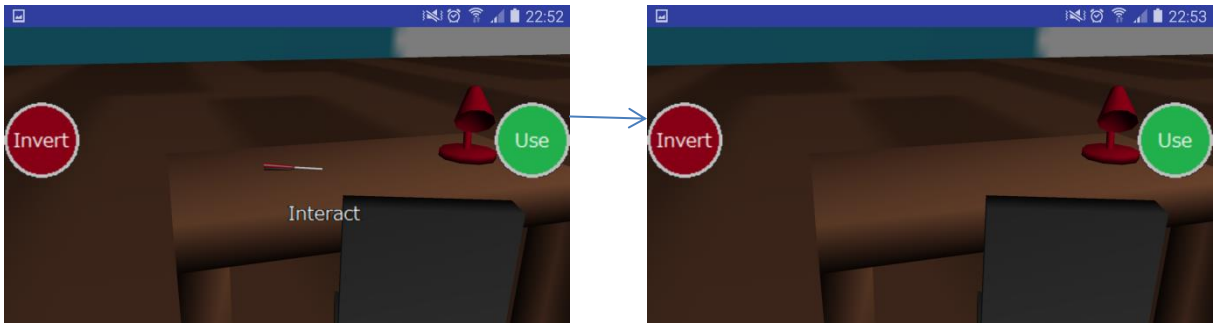
Character Movement



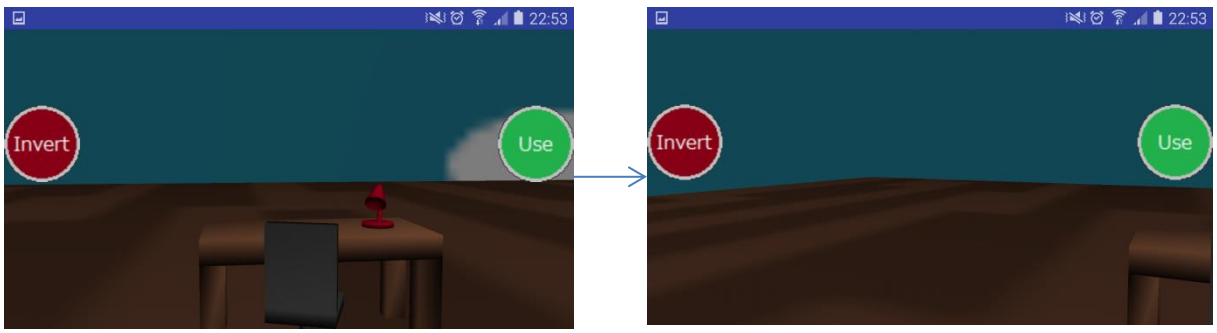
Camera Rotation



Button Press



Invert X-axis and Y-axis for camera rotation



The following are tests which can be used for the final version of the project, as well as tests to run during the development of this project. These do not reflect the final tests for the finished product, and only give an example of the type of tests for the final product. These are subject to change, with additional tests expected as development of this project continues, along with tests of the flow of the game.

Test Case	Type of Test	Expected Output	Comments
Main Menu GUI	Black Box Test	Tapping buttons on the Main Menu will take us to different activities, such as starting the game or to the Options Menu, or exiting the game	The Main Menu will appear after the splash screen, with buttons to go to other activities
Options Menu	Black Box Test	By changing variables in the Options Menu, this should affect parts of the game, such as the volume, the controls, etc.	As well as being able to set different parts of the game, there will also be an options to set default controls
Music and Sound	Black Box Test	At different parts of the game, or in different app activities, music and sound effects will play, and the correct music and sound effects should play, at the volume set in the Options	Changing the music of sound volumes will be relative the smartphones current volume
Different devices/ screen sizes/ aspect ratios	Black Box and White Box Test	On different devices, the program should still run, and the player should be able to use all the functionality regardless of the device	Devices must be at least of the Android version set for the program, and parts of the game and GUI will scale to fit the screen. The White Box part of this will be the code drawing objects/ the user-interface based on the screen size, as opposed to absolute values

Bibliography

Butler-Hartley, J. (2013, May 1). *Survival Horror vs. Action Horror*. Retrieved November 18, 2016, from Zero1Gaming: http://www.zero1gaming.com/editorials/survival-horror-vs-action-horror/

Foerster, H. (n.d.). *j-PCT-AE*. Retrieved October 31, 2016, from http://www.jpct.net/jpct-ae/index.html

GamesRadar Staff. (2016, February 17). *The 20 best horror games of all time*. Retrieved November 17, 2016, from Games Rader: http://www.gamesradar.com/best-horror-games/

GlitterBerri, & Fuunoshin. (2012, July 20). *The Man Who Made Ghosts'n Goblins*. Retrieved November 22, 2016, from GlitterBerri: http://www.glitterberri.com/developer-interviews/tokuro-fujiwara/

Habel, C., & Kooyman, B. (2014). Agency mechanics: gameplay design in survival horror video games. *Digital creativity Vol.25 (Exeter)*, 1-14.

Palmer, M. (2005, August 31). *Mobile phone makers eye booming games market*. Retrieved November 26, 2016, from FT.com: http://search.proquest.com.proxy.library.dmu.ac.uk/docview/229014244?pq-origsite=summon

Peckham, M. (2015, March 18). *Exclusive: Nintendo CEO Reveals Plans for Smartphones*. Retrieved November 26, 2016, from Time.com: http://time.com/3748920/nintendo-mobile-games/

Robbins, M. B. (2015, January 1). Keep the Lights On: Horror Games. *Library Journal*. United States: Media Source.

Soderman, B. (2015). 'Don't Look … Or It Takes You': The Games of Horror Vacui. *Journal of visual culture*, 311-316.

Sterling, J. (2008, December 8). *How survival horror evolved itself into extinction*. Retrieved November 22, 2016, from Destructoid: https://www.destructoid.com/how-survival-horror-evolved-itself-into-extinction-114022.phtml

References

Android Studio

Autodesk Maya

Bloodbourne

Dark Souls

Dead Space

Dead Space (Series)

Five Nights at Freddy's (Series)

Five Nights at Freddy's

Five Nights at Freddy's: Sister Location

FL Studio

Game Maker Studio

HTC Vive

j-PCT-AE

Microsoft Visual Studio

Nintendo 3DS

Nintendo Dual Screen (DS)

Nintendo Entertainment System (NES)

Nintendo eShop

Nintendo Switch

P.T.

PlayStation Portable (PSP)

PlayStation Vita (PS Vita)

PlayStation VR

Resident Evil (Series)

Resident Evil

Resident Evil 4

Resident Evil 5

Silent Hill (Series)

Silent Hill

Souls (Series)

Super Mario Bros.

Unity

Student Name: Simon Chiu

P-Number: P14133969

Programme: Computer Games Programming

Email address: p14133969@my365.dmu.ac.uk

Project Title: 1st Person 3D Horror Mobile Phone game

Project Proposer: Self

Supervisor: Archie Singh, Researcher, De Montfort University, archiedmu@gmail.com

BCS Accreditation:

This contract contains an elucidation of the problem, the objectives of the project and a risk analysis			**Yes** ☐	**No** ☐
The contract states that the project will include an in-depth investigation of the context and literature, and where appropriate, other similar products			**Yes** ☐	**No** ☐
The contract states that the final report will contain a clear description of the stages of the life cycle undertaken			**Yes** ☐	**No** ☐
The contract states that the final report will contain a description of how verification and validation were applied.			**Yes** ☐	**No** ☐
The contract states that the report will contain a description of the use of tools to support the development process			**Yes** ☐	**No** ☐
The contract states that the final report will contain a critical appraisal of the project, indicating the rationale for any design/implementation decisions, lessons learnt during the course of the project, and evaluation (with hindsight) of the project outcome and the process of its production (including a review of the plan and any deviations from it)			**Yes** ☐	**No** ☐
The contract states that there will be a description of any research hypothesis			**Yes** ☐	**No** ☐
The contract states that all research will be fully referenced			**Yes** ☐	**No** ☐
Contract is suitable for BCS Accredited Project	**Yes** ☐	**No** ☐	**Supervisor Signature**	

Introduction:

The project will focus on creating a 3D mobile phone engine to create the game in, and the engines ability to import 3D models and textures for the player to play the game. Certain aspects include touch screen controls, file input and output, 3D rendering and camera control, handling phone functions (such as phone calls or notifications) and music and sound effects.

Project Background:

This particular project arose from my interest in a particular series of games, the “Five Nights at Freddy’s” series by Scott Cawthorn, which is known for not only its gameplay, but also for its story, which interests me the most. Many people are interested in the series for the large story hidden within its games. Another source of inspiration is from my course, and my choice of modules. For this project, I specifically want the horror aspect of the game to arise from an “AI” in the story of the game, with some of the puzzles being how to get around the programmed AI functions.

Aim:

The project will be the creation of a 1st person Horror game, rendered in 3D, and playable on a mobile phone (Android). This will involve the player being able to play from start to finish, using an Android phone with touch screen controls, in a 3D environment, completing puzzles and moving around in-game to reach the end.

Objectives:

- To investigate the use of 3D in mobile phone games
- To design the layout of the game itself from start to finish, including the story
- To design certain aspects of the game, such as puzzles or events in-game
- To design and follow a test plan (for example, unit testing)
- To review and discuss the pros and cons of certain aspects of mobile phone games
- To find or create 3D models, textures, music and sound which is permitted for use

Deliverables:

For Initial Delivery/ Prototype

- Project contract
- Ethics form
- Project Plan
- Literature review
- Story board
- Design Documentation
- Test Plan
- Prototype

For Final Delivery

- Final report
- Appendices
- A .apk file to install the finished game onto Android phones

Resources:

- A computer (with peripherals such as mouse and keyboard)
- Mobile phone app development software
- 3D Models and Textures, or software to create them
- Sounds and Music, or software to create them
- School computers/ labs
- Help from colleagues or lecturers (opinions, testing)

Constraints:

- Time limit
- Deadlines (Interim and Final deadline)
- Software and Hardware limitations (limits of a 3D game on a phone, certain framerate, etc.)

Sources of Information:

- Other mobile phone or horror games (websites, articles, interviews, videos, screenshots and reviews)
- Documentation for software (mobile phone app development software, 3D computer graphics programs, graphics software, music and sound creation software)

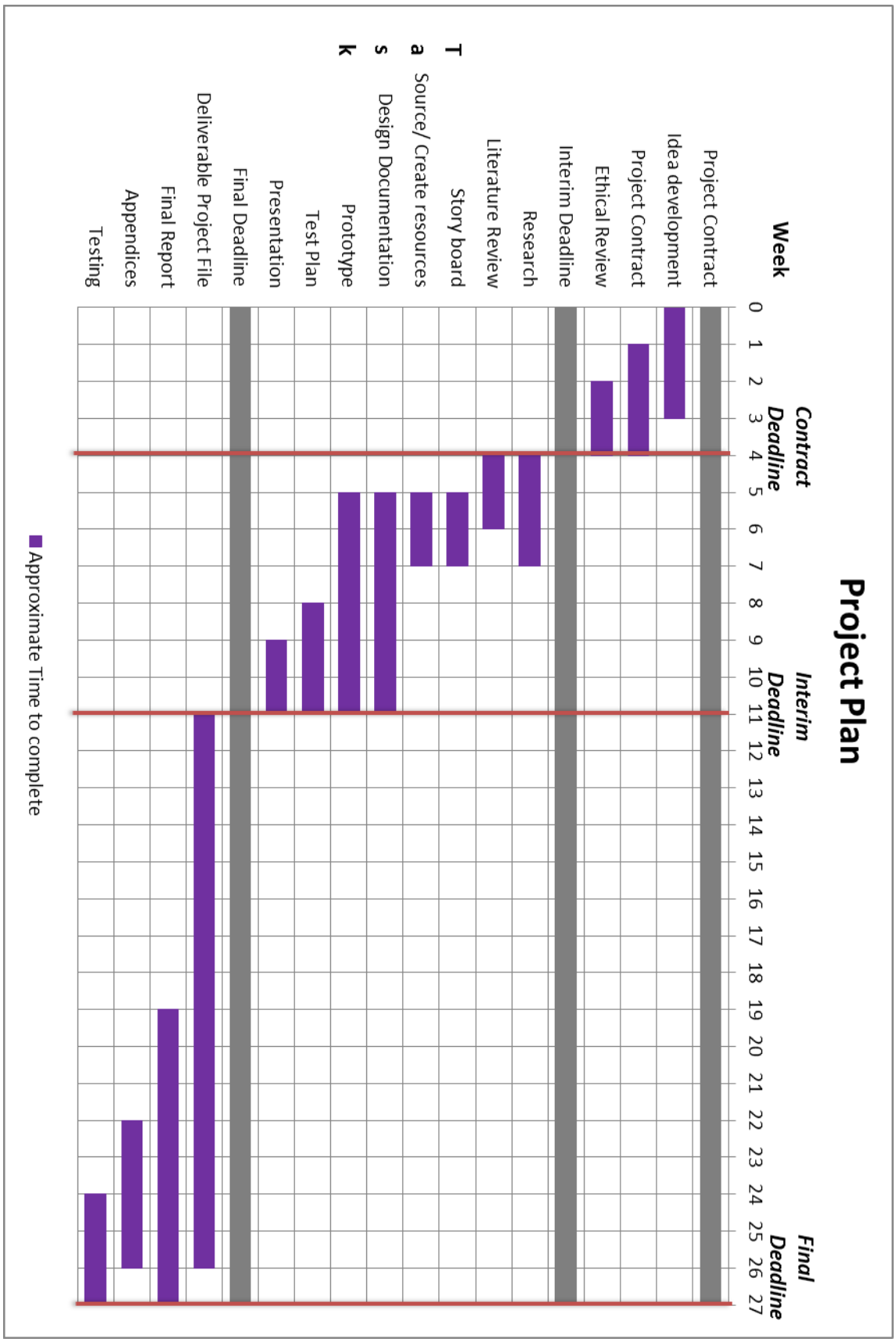
Risk Analysis:

There are some risks which any project could have which would endanger the project, such as illness, certain events, etc. For a development project, there is the risk of software limitations, such as a program not having the functionality I need, or problems with certain operating systems, or school computers not having the software I need.

For risks related to events or illness, a strong project plan should allow leeway for any problems which may arise, although any large events may be unavoidable. In the case of software not having the functionality I want, either alternative software can be found (if early in the project stages) or an alternative functionality used, or have the functionality taken out.

Project Plan:

The project needs to be completed within a certain time limit, from Monday 3rd October (Week 1) to Tuesday 4th April at 2pm (Week 27) with an Interim Deadline on Monday 12th December at 2pm (Week 11). Between these times, certain parts of the project must be done by certain deadlines, with everything completed for the final deadline.


Project Plan
Week
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27
Contract Deadline
Interim Deadline
Final Deadline
Project Contract
Idea development
Project Contract
Ethical Review
Interim Deadline
Research
Literature Review
Story board
Tasks
Source/ Create resources
Design Documentation
Prototype
Test Plan
Presentation
Final Deadline
Deliverable Project File
Final Report
Appendices
Testing
Approximate Time to complete

Idea development: The creation of the basic core idea for this project. In this case, the idea of a 3D 1st Person mobile phone game

Project Contract: This document, which contains the premise of what the project idea is, how it will be executed and what it must include

Ethical Review: A review of any ethical issues which may arise from the development of this project

Research: Research into use of certain software, their pros and cons, and what would be viable for this project

Literature Review: A review of similar products on the market, what was good or bad about them, and how this information can be used for my project

Story board: A story board which shows the scenarios in the game to reach from start to finish

Source/ Create resources: Finding 3D models, textures, music or sound which I can use for my project, or creating them myself

Design Documentation: A documentation of the game explaining controls, gameplay, assets and story in detail

Prototype: A prototype of the game, showing off certain elements, functions and gameplay. The key of this is to show how the final project should be

Test Plan: A plan for testing different aspects of the project, as well as how the player can interact with the project

Presentation: The presentation for the interim deadline. This will be used with the prototype to show how far the project is

Deliverable Project File: Continuing from the Prototype, this is the creation of the final project

Final Report: The core report of the project, including my views of the project as a whole, and of the final product

Appendices: References and acknowledgements to people, websites, articles, etc. which was used for the project

Testing: The actual testing of the project using the premade Test Plan from earlier

Student_____ **Date**_____

Proposer _____ **Date**_____

Supervisor_____ **Date**_____

IMAT3451 FINAL YEAR PROJECT - ETHICAL REVIEW FORM

The University requires all undergraduate final year projects to undergo an ethical review and, where human research ethical issues are identified, to ensure that these issues are addressed.

For the majority of Computing Final Year Projects, the outcome will be either 'No ethical issues' or 'Minor/Major ethical issues which have been addressed'; in these cases approval can be given by the supervisor. In the unlikely event that the outcome is 'Ethical issues that have not been addressed', the completed form will need to be forwarded to the Faculty Research Ethics Committee.

Student Name

Simon Chiu

Programme

Computer Games Programming

Project Title

1st Person 3D Horror Phone Game

Brief description of proposed activity and its objectives:

The project will be the creation of a 1st person Horror game, rendered in 3D, and playable on a mobile phone.

Ethical Issues Identified:
(see overleaf)

None

How these will be addressed:

None

Checklist

Has the project proposal identified any of the following research procedures?

1. Gathering information about human beings through: Interviewing, Surveying, Questionnaires, Observation of human behaviour — Yes / (No)
2. Using archived data in which individuals are identifiable — Yes / (No)
3. Researching into illegal activities, activities at the margins of the law or activities that have a risk of personal injury — Yes / (No)
4. Supporting innovation that might impact on human behaviour e.g. Behavioural Studies — Yes / (No)

If 'Yes' to any of 1-4 above: have you considered the following?

- ☐ Providing participants with full details of the objectives of the research
- ☐ Providing information appropriate for those whose first language is not English
- ☐ Voluntary participation with informed consent
- ☐ Written description of involvement
- ☐ Freedom to withdraw
- ☐ Keeping appropriate records
- ☐ Signed acknowledgement and understanding by participants
- ☐ Consideration of relevant codes of conduct/guidelines

Ethical Review Outcome

- ☐ 1. No ethical issues
- ☐ 2. Minor ethical issues which have been addressed and concerns resolved
- ☐ 3. Major ethical issues which have been addressed and concerns resolved
- ☐ 4. Ethical issues that have not been resolved/addressed

Authorisation

If the outcome is no. 3 or 4 above, this form should be forwarded to the Faculty Research Ethics Committee.

Signature of student _____ Date _____

Signature of supervisor _____ Date _____

Signature of 2nd supervisor _____ Date _____

IMAT3451 FINAL YEAR PROJECT - Global Checklist

The University requires all undergraduate final year projects students to undertake a global review of their project. Here is an International Impact Checklist for you to complete, which can be done in consultation with the project supervisor.

Student Name

Simon Chiu

Programme

Computer Games Programming

Project Title

1st Person 3D Horror Mobile Phone Game

Please indicate which of these possible attributes is addressed by your undertaking of this project.

Possible Global Experience	**Addressed by Project**
Ability to work collaboratively: teams from a range of backgrounds and countries	
Excellent communication skills with a sensitivity to speaking with and listening to non-native English speakers	
An ability to embrace multiple perspectives and challenge thinking in a range of cultural context	X
A capacity to develop new skills and behaviours according to role requirements	
An ability to negotiate and influence clients across the globe from different cultures	
An ability to form professional, global networks	
An openness to/respect of a range of perspectives from around the world	
Multi-cultural learning agility (i.e. able to learn in any culture or environment)	

Brief description of how the ticked attributes have been addressed:

An ability to embrace multiple perspectives and challenge thinking in a range of cultural context

This project challenges Horror games and *Horror vacui* by having a component of the game which can induce fear into the player, however, not necessarily via off-screen space (*Horror vacui*) or by on-screen space, but rather via on-screen objects which may or may not harm the player.

Signature of student _____ Date _____

Signature of supervisor _____ Date _____

IMAT3451 Final Year Project Periodic Progress Report (PPR)

Programme/Course Title: Computer Games Programming

Name: Simon Chiu Assessment Period: Interim

Project Title: 1st Person 3d Horror Phone Game Report Number: 1

Objectives for Period: (refer to previous report)

Review the requirements for the final year project
Answer any questions concerning the final year project

Summary of Progress for Period: (identify evidence of progress)

Developed an idea for a final year project
Discussed project requirements and deadlines

Problem Areas and Suggested Solutions:

Objectives, Deliverables & Plan for Next Period:

Decide on final year project idea
Produce a project contract, including deliverables and project plan
Fill out an ethical review form for the project

Date of Next Review: 24/10/16

Student's Signature: Date: 10/10/16

Supervisor's Signature:

Comments (if any):

IMAT3451 Final Year Project Periodic Progress Report (PPR)

Programme/Course Title: Computer Games Programming

Name: Simon Chiu Assessment Period: Interim

Project Title: 1st Person 3d Horror Phone Game Report Number: 2

Objectives for Period: (refer to previous report)

Review the project contract draft and sign it off
Review the ethical form for any concerns

Summary of Progress for Period: (identify evidence of progress)

Project contract was sufficient, signed off
No ethical concerns, signed off

Problem Areas and Suggested Solutions:

Objectives, Deliverables & Plan for Next Period:

Begin research and literature review of final year project

Date of Next Review: 7/11/16

Student's Signature: Date: 24/10/16

Supervisor's Signature:

Comments (if any):

IMAT3451 Final Year Project Periodic Progress Report (PPR)

Programme/Course Title: Computer Games Programming

Name:Simon Chiu Assessment Period: Interim

Project Title:1st Person 3d Horror Phone Game Report Number: 3

Objectives for Period: (refer to previous report)

Check the progress of the project, namely the research and literature review

Summary of Progress for Period: (identify evidence of progress)

Have started research and literature review, should send copies to supervisor to read through

Problem Areas and Suggested Solutions:

Objectives, Deliverables & Plan for Next Period:

Continue work on research and literature review, and show progress made
Continue work on a prototype

Date of Next Review: 21/11/16

Student's Signature: Date: 7/11/16

Supervisor's Signature:

Comments (if any):

IMAT3451 Final Year Project Periodic Progress Report (PPR)

Programme/Course Title: Computer Games Programming

Name:Simon Chiu Assessment Period: Interim

Project Title:1st Person 3d Horror Phone Game Report Number: 4

Objectives for Period: (refer to previous report)

See progress on project in terms of prototype and literature review

Summary of Progress for Period: (identify evidence of progress)

Have seen how literature review and research write up has progressed so far

Problem Areas and Suggested Solutions:

Objectives, Deliverables & Plan for Next Period:

Show majority of the write up to review areas to improve before the deadline
Continue work on a prototype to show core parts of the project

Date of Next Review: 5/12/16

Student's Signature: Date: 21/11/16

Supervisor's Signature:

Comments (if any):

IMAT3451 Final Year Project Periodic Progress Report (PPR)

Programme/Course Title: Computer Games Programming

Name:Simon Chiu Assessment Period: Interim

Project Title:1st Person 3d Horror Phone Game Report Number: 5

Objectives for Period: (refer to previous report)

See progress of project, with a near complete project which is ready for the submission deadline
Confirm plans for presentation/ Demo of project

Summary of Progress for Period: (identify evidence of progress)

Have seen progress of write up and identified all items to present for interim deadline

Problem Areas and Suggested Solutions:

Objectives, Deliverables & Plan for Next Period:

Show progress over Christmas period and plans for final submission deadline

Date of Next Review: 9/1/17

Student's Signature: Date: 5/12/16

Supervisor's Signature:

Comments (if any):