David Hesketh Honours report

Introduction

## 2.1 Glossary of Terms

Sight Loss – An individual is described as suffering from sight loss when their visual acuity is found to be 6/12 or worse as per the ‘State of the Nation Year in Review’ carried out by Specsavers in cooperation with the Royal National Institute for Blind People (Specsavers, 2017).

Project Sprint – The term used to mean a small but intensive period of work done to a short deadline.

Bugs – An error or flaw in the software system which causes the program to behave in unexpected ways or provide an incorrect result.

Binaural recording – A way of recording audio with 2 microphones arranged in such a way as to create a 3D sound effect for the listener.

Skill Ceiling – The point at which a player has mastered their ability to play a game and as such can improve no further.

## 2.2 The Project

The subject of this report is the design and development of a computer game with the specific aim of being easily accessible and enjoyable to those who suffer from sight loss. The intent behind this is to widen the reach of gaming to those who do suffer from this condition and to shine a light on the community of gamers with this disability.

## 2.3 Background

This project begun as an attempt to design and develop a game following specific constraints. There was the additional aim to show that the community of gamers suffering from sight loss need not be ignored in favour of greater graphical fidelity. To this extent a game has been designed and developed with those who suffer from sight loss as the key target audience. This game can also be played by those with full sight without difficulty.

### 2.3.1 The Problem/Rationale/Reasoning for project

People suffering from sight loss have difficulty playing games with visual effects designed for fully sighted users. Many developers do not take into account the possibility of those without sight playing their game and as such do not include several simple features which make many games much more accessible to those within this community. It is this is inherent lack of accessibility that is the problem within the games industry. This leaves a large market of potential players untapped. This market, treated by many developers as empty or even non-existent, is a wide and diverse one. It is possible that ignorance of developers toward this community is due to the fact that the majority of those suffering from blindness are 50 years of age or older (World Health Organisation, 2010) but the majority of those who play video games are under the age of 36 (Grubb, 2014). Despite this, the community of those who play video games without full access to their sight is a diligent one, taking to message boards and modding communities to help widen many game’s accessibility on their own. The response from many developers regarding this is a positive one, and these outlets can enact real change, as shown when a blind fighting game player raised concerns on the “Killer Instinct” forums (Yin-Poole, 2016). With this in mind, it should not be the responsibility of these gamers to ensure accessibility in the industry.

### 2.3.2 Accessibility in Gaming

The Video Game industry has been a growing economic powerhouse since its creation. From its simple origins in arcades where each play cost 25 cents to the increase of the total industry to a worth of 99.6 billion dollars as of the Newzoo 2016 global games market report (Newzoo, 2016). Despite what is suspected to be continued financial growth, there has been very limited development in the untapped market of accessibility in gaming. With approximately 360,000 people in the UK being registered as blind or partially sighted (1 in 30) as of 2014 according to the Royal National Institute of Blind People (RNIB). The number of those with sight loss is predicted to increase to over 2,250,000 as of 2020 as stated by the RNIB for reasons such as an aging population (Transversal, 2014). This increasing population is however comparatively underserved in the current games market. Audiogames.net shows only an available 606 audio games at the time of writing (Audiogames.net, 2017).

Companies such as Audiogames.net and AppleVis both of which provide gaming accessibility help to those with limited to no sight capabilities. AppleVis provides reviews of iOS games based on how accessible they are. Audiogames.net has compiled a list of games available and accessible to those with visual impairments, primarily in the field of the audio game genre. Being started by two lecturers, Richard van Tol and Sander Huiberts to provide an online information point for audio games. Companies like these are comparatively few and far between whereas companies that do not put accessibility at the forefront of their development model are far greater in number.

Development of audio games as a genre is not inherently more difficult than the development of any other genre of games. It does, however, require a paradigm shift. Many companies put cutting-edge graphical technology as their primary concern allow audio to be an afterthought. The required paradigm shift would put the audio development of a game to the forefront and push visuals to a more secondary consideration by comparison. Which up to this point in the mainstream games industry has not occurred.

### 3.4 Ethical Considerations

Participants who are over the age of 18 are required to take part in this project. Due to the fact that this development and subsequent experimentation will require human participation, an ethics approval form will be required to be filled out, signed by the project supervisor and subsequently sent off to the ethical approval committee. Receiving ethical approval as early as possible will be essential for the ongoing success of this project.

Autonomy – Any participants who wish to leave at any time during the experiment are welcome to do so. All participants will be advised exactly what this research will be used for and what is required from them at time of request for participation. No one will be persuaded or coerced by researchers or outside influences during any meetings. All participants will be notified that they have the option to leave at any time. All who take part will be anonymous and be advised as such.

Confidentiality – All information relating to participants will be kept anonymous and will not be released. Information will exclusively be used for the project and will be destroyed at completion of project.

Special Needs – All special needs will be taken into account for any participants who require any additional support, such as those who have visual impairment requiring a researcher to guide them to the project area. Participants will be made aware that they have the option for this additional support.

Protected Vulnerable Groups – To ensure there is no conflict regarding protected vulnerable groups disclosure no one who is in receipt of any of the following will be asked to take part in this test; Registered Care Services, Community Care Services, Health and Welfare Services. To determine this, participants will be asked in advance and any who receive these services or refuse to answer will not be considered eligible for the test.

# 3.0 Literature Review

## 3.1 Genre

To determine what game should be developed for this project genre must be considered. The aspects of this which will be kept in mind are; ease of development, ease of portability, popularity and level of accessibility. To determine this, reviews will be read and the genre of which appears most frequently with highest ratings in terms of accessibility and developer familiarity with the genre in question. The reviews will be pulled from <http://game-accessibility.com>, <https://www.gamespot.com/reviews/>, <http://blog.aidis.org/>, <http://reviews.sightlesskombat.com/>, as these sites all provide reputable reviews. Several of the sites; game-accessibility.com, sightlesskombat.com and aidis.org all provide accessibility-based reviews with sightlesskombat.com being run by a blind competitive gamer.

The games will be formatted in a chart format to

## 3.2 Development engine

The engine for the development of this project must be carefully considered. Its adaptability portability must be considered prior to development to allow for maximum outreach. Language and peripheral support must be taken into account to ensure development continues with ease without the necessity of attempting to develop a new programming language or learn an entirely new syntax. Experience with the engine and knowledge of its inner workings should be considered to cut down on development time and allow a greater period for experimentation and evaluation of results. The available features of the chosen engine must be considered for the development of the project and, in the instance that a feature is missing, development time and importance level for that feature must be considered. Furthermore, accessibility of documentation is needed for error handling, should any occur. To this effect the popular development engines Unity 3D and Unreal Engine 4 will be reviewed for possible use.

### 3.2.1 Unity 3D

Unity 3D is a development engine designed for portability, accessibility, and ease of use. It supports the languages C# and JavaScript with the ability to write plugins to be read by the engine in C++. The core of unity itself is written in C++ with certain aspects of the editor being written in C#. This allows the engine to be very portable and gives it a shorter compile time.

Of the supported languages C# allows function overloading, JavaScript does not however which would reduce the customisability of the software to a small degree, however the requirement of function overloading is not a complete necessity as instead of using the process of function overloading a simple case of multiple function names can be used (Rongala, 2017). Unity 3D is also a particularly portable development environment allowing development for; virtual reality, mobile (Android and iOS), console, PC, MacOS, Linux, tvOS, WebGL, Samsung TV and PS Vita. This portability can provide a great deal of outreach to a variety of commercial devices, allowing further development into a commercial marketplace in the instance this project proves successful.

Unity 3D offers a standard set of classes for the development of games. These classes allow a simple and easy starting point for the development of game software. This development environment also provides a series of tutorials for all technology used and there is a large community of user made tutorials available online which allow the creation of a variety of games with minimal effort. These tutorials can act as a structure for the creation of software. However, this resource must not be overused to avoid any issues with plagiarism in the development of this game in an academic context.

Unity 3D provides 2 integrated physics engines, one for 2D physics and another for 3D physics. They both make use of rigidbody components (labelled as “Rigidbody 2D” in the case of 2D physics). The primary difference between the two physics engines is largely the addition of the extra dimension which allows the use of the Z axis (Technologies, 2017). This provides more design opportunities for the development of the project with the opportunity to have a 3D or 2D game. This opportunity must also be carefully considered for the development of the project in the instance Unity 3D is the chosen development engine.

In addition to an integrated physics engine Unity 3D offers the ability to provide 3D audio to help allow those without the use of sight identify the location of objects that create sound, reducing volume at distance and increasing as the player character gets closer to the source of the sound.

Unity 3D also provides a simple and easy to use analytics system which will allow recording of in-game analytic metrics such as play session duration and the level at which the player quit. These analytics work through an events system integrated into unity. Examples of this can be found in games such as the Android game Bright Void (Hesketh and Campbell, 2017).

### 3.2.2 Unreal Engine 4

Unreal Engine 4 is the primary game engine for the game development and publishing company, Epic Games. Known for games such as; Unreal Tournament, the Gears of War series, Bulletstorm and Paragon. All developed in the unreal engine, these games show an intuitive use of physics, lighting and mechanical techniques which the unreal engine provides. The engine itself is written in C++ as with Unity 3D allowing it to also have a high degree of portability and reduced compile time.

Unreal supports C++ as a development language as the basis of blueprint classes. The blueprint system is a visual scripting system to allow classes to be created in an innovative manner. This system uses visual blocks of code to show functions and references to variables (Games, 2017). This development method makes it easier for those in non-programming teams to create software for specific features within the game. This, however, is not necessarily a benefit in this project as this is development project is a programmer-specific project. Meaning the programming itself requires less simplification than in the case where designers would be required for the creation of features.

Unreal Engine 4 Also provides a variety of post-processing effects and a large animation toolset. However, due to the nature of the shortened development time and the simplicity of the visuals, these features become largely irrelevant and the learning curve on these features from a starting point of zero experience with them in the engine prohibit their use.

This Engine does, however, offer integrated support for Virtual Reality (VR) and Augmented Reality (AR) technologies. This makes the development of these applications quicker and easier than in most other engines. Since there are multiple virtual reality platforms under consideration for this development project, this integration has the potential to be greatly beneficial to the project’s development cycle in reducing the time required to integrate the VR aspect should that technology be chosen.

While this suite of development tools offers a large variety of high-end features, unfortunately, based on the feature list available (Games, 2017) there is no inbuilt analytics system at the time of writing. There is also a concern that the number of features available from Unreal 4 would be irrelevant and would clutter the development process. The lack of an inbuilt analytics recording system means that several metrics would need to be recorded manually, slowing development and experimentation process. For these reasons, coupled with the lesser amount of experience with the Unreal 4 Engine, the development project will use Unity 3D for the development and evaluation of this project. It is hoped that this will be a quicker and more effective development process as a result of this choice.

## 3.3 Development Platform

The development platform of this program must be carefully considered for a variety of reasons. This project must take into account the accessibility of the chosen device to ensure those suffering from sight loss can make use of the product. The cost and market popularity must be considered as a factor in the game’s ability to reach a target demographic. The ease of development of the project for the chosen platform and availability of documentation and reference materials must also be considered in order to ensure the pace of the development project continues briskly and without issue.

### 3.3.1 Virtual Reality

With the tremendous variety of virtual reality devices on the market, the choice between them can be a daunting task for many consumers, especially for those who do not necessarily benefit from the full effect of them. For that reason, this report will narrow the field of development devices by taking into account only the following virtual reality systems; HTC Vive, Oculus Touch, and the Google Daydream.

The HTC Vive offers the largest scale virtual reality experience with at the highest cost of the reviewed devices with a £599 price point (HTC Corporation, 2017). This device is also the most complicated device in terms of initial setup. The device comes equipped with a pair of base stations which HTC recommend be affixed to a wall. The Headset itself feeds into a “breakout box” via HDMI, USB 3.0 and 3.5mm headphone jack to provide audio to the headphones connected to the headset. Then comes a lengthy setup procedure involving synchronising the base stations and the software setup for either room scale or standing only (HTC Corporation, 2016). Comparatively, the Oculus Touch has a much simpler setup process (Stuff.com, 2016). Two sensors and a headset are plugged into the computer via USB 3.0 cable, and then a quick software setup is done using the Oculus Software download. This is a much quicker and easier process however only offers room scale VR on an experimental basis with standing VR being the more commonly used on this system. This allows the user to more quickly and easily pick up and play with the Oculus Touch compared to the HTC Vive. However, of all the reviewed devices here, the Google Daydream has the simplest setup procedure. Dues to its requirement to be used with the Google Pixel mobile phone as opposed to a desktop setup this device simply requires the user to open the daydream app on their phone and follow the instructions to pair their device to the headset then insert the device into the headset. This makes it the simplest and easiest virtual reality device to set up, as there is no lengthy attachment procedure to attach a computer to the device and sensors to pair. This does, however, come with the downside that the Daydream offers no body tracking whatsoever meaning that design decisions within the project would be compromised.

The HTC Vive is often considered the top end of virtual reality hardware being the best-selling VR device of those reviewed, having sold 420,000 devices as of March 2017 (Ergürel, 2017). This would suggest that the Vive would have the greatest market share of VR devices as the Oculus Touch had only sold 243,000 devices and the Daydream having sold an estimated 260,000 headsets as of the final quarter of 2016 (Ergürel, 2017). This data would imply that to reach the greatest number of users within the general population the Vive should be developed for. This is reinforced in the partially sighted community by academic experiments previously done. Larger scale VR devices are generally the chosen development devices for those working with the blind as it allows easier development of cognitive maps of an area by those who do suffer from blindness (Merabet et al., 2012). Of the reviewed devices technically speaking the Google Daydream offers the largest scale VR as it is not confined to an area by sensors as with the Vive and Oculus. However, this benefit is offset by the lack of any form of body tracking making following the player more difficult within context.

The development cycle for a virtual reality game is largely the same as that of a conventional development project. In the instance of these three devices, the development for an Oculus and HTC Vive are largely the same with the chosen engine (Unity 3D) offering VR support in its latest incarnation. Allowing virtual reality support is as simple as checking the “VR Supported” box within the Unity editor and ensuring that the editor is set to build for PC, MacOS or Linux. In the case of the Google Daydream VR device, development is slightly more complicated despite still being integrated within Unity’s editor (Technologies, 2017) as it involves the integration of the Android SDK and an additional layer of debugging if there are any build errors involved.

Should the development of this project use a virtual reality device, then the chosen device for this purpose shall be the HTC Vive. With its high market share and ease of development within Unity in addition to the increased scale of the available play area, the Vive becomes the most appropriate device. The use of the Vive’s body tracking system should allow greater ease of design and the available peripheral integration should provide suitable complexity for the development portion of the project.

### 3.3.2 Computer

It is no secret that the section of the video gaming industry dedicated to computer games is a popular one, with its advocates fiercely defending the platform. There are three main competitors within the computer market in terms of operating system, those are; Windows, MacOS and Linux based operating systems. In addition to previous review criteria, reviewed computers and technical requirements shall consider requirements for the technologies previously reviewed and operating systems available for development on the Unity 3D engine. Luckily in this instance, Unity3D offers build options for PC, MacOS and Linux provided the developer has appropriate licences.

The accessibility of a desktop computer is second to none when it comes to allowing those with limited access to technology. The incredible variety of peripherals available for the computer user allows almost anyone to play games on a computer. With Apple’s MacOS computers being more geared towards proprietary software, the access to these kinds of peripheral devices are slightly more limited to on this operating system. However, in the instance of a windows operating system device most peripherals are quickly available. The same is true of many Linux based operating systems due to the open source nature of Linux. This means that if a peripheral is developed with drivers specifically for Windows or MacOS then there will often be a Linux equivalent available for download. In the instance that there are not drivers readily available one of the Windows or MacOS emulators such as ‘Wine’ can often used to properly run the required software for the device. In the case of this development project, the only required peripheral in the case of a computer-based game would be; a pair of headphones, a mouse, a keyboard and a monitor. Since the project targets those with visual impairment, a braille keyboard may be necessary. However, this would not be for the development project itself but instead would be for the subject of the experiment to more comfortably use the computer. All named devices are easily available for all reviewed operating systems and as such provides no weighting to the development of the project.

Linux operating systems are, due once again to their open source nature, free to download and use. This means that the cost of a Linux based computer is exclusively on the hardware cost. This, in turn, allows more people to more easily afford a computer. It may seem that this reduction in cost would mean that Linux as an operating system would have the highest market share among gamers. However, this is not necessarily the case as shown by the Steam Hardware and Software Survey (Valve Corporation, 2017). This monthly survey shows that as of October 2017 Linux based software made up only 0.35% of their user base, it is suspected that this is due to the lack of support from developers for Linux (Lyer, 2017). This is found similarly in MacOS based operating systems, again shown in the Steam Hardware and Software Survey (Valve Corporation, 2017) in this instance MacOS barely beats out Linux with only a 1.60% usage rate from steam users, the most popular of which being free after Apple has removed upgrade costs. The clear most popular operating system according to this survey is windows with 98.04% of Steam users using this operating system, for maximum market penetration the project would be developed to run on Windows 7 as 63.60% of Windows users on steam are running this operating system. While this is no longer available for purchase from Microsoft directly the most recent version of windows is available from the Microsoft website at £119.19 (Microsoft Corporation, 2017).

The availability of documentation for the development of the project is largely the same for each operating system due to the choice of the Unity 3D as a development engine. The prevalence of documentation for Unity 3D which supports all three operating systems means that each operating system can be developed for with ease in this instance. However, to develop for MacOS a Mac computer is required along with a licence which has an annual cost of $99 making it less favourable to develop for.

Overall due to ease of development and higher market penetration this project will be developed for Windows. Since all reviewed operating systems are equally easy and available for development when using Unity 3D, market penetration becomes the only metric to bias the decision of which to use. As such Windows becomes the obvious development decision.

### 3.3.3 Mobile

Mobile games have been a rapidly emerging market within the games industry. Quickly becoming one of the most used devices for gaming, 42% of all game revenue has been made due to mobile (McDonald, 2017). For this reason, mobile must be considered for the development of this project. The two most popular mobile operating systems for the development of games are iOS and Android; developed by Apple and Google respectively. These operating systems will be reviewed as per the stated criteria to determine the most appropriate system for the development of this project should mobile be chosen as a development platform.

Both Android and iOS do both offer equal levels of hardware accessibility being operating systems for very similar types of device. Should development for this project take place using a smartphone or tablet device then certain hardware features should be considered. With the knowledge that an estimated 97% of smartphones make use of touchscreen technology (Allied Business Intelligence, 2011) other methods of control must be considered. The development of this project, should it be done on mobile devices, must take into account the fact that those who are unable to see a touch screen may be unable to use a touchscreen. In this instance, the primary method of control would likely be the accelerometer allowing for tilt controls on experimental devices used.

Android has the greatest market penetration of all mobile operating systems as of the second quarter of 2016 according to Gartner (Gartner, 2016). This report states that of all devices currently sold 86.2% have been Android devices. iOS is far from this with a 12.9% market share according to the same report. This shows that should the project be developed with Android in mind a much larger market demographic would be feasible than in the case of iOS.

With the development of this project taking place within the Unity 3D engine, ease of development has become much less of a concern for each device. Since Unity provides support for both reviewed systems, the development of either mobile operating system would be equally simple. However, once again, should the device be developed for the apple based product an annual 99$ licence fee would be required. For this reason, should the project be developed for a mobile device, the Android operating system would be the targeted platform.

Based on all reviewed platforms and devices this project will be developed targeting the **HTC Vive. The reason for this is, as previously shown, virtual reality devices can provide a large amount of assistance to those suffering from sight loss despite the inability to see the environment around them. As the results of Merabet and others have shown the ability of the blind to develop cognitive maps, which can be transferred into a play space, this should allow a much richer play experience for those making use of the developed game.**

**Update of Literature Review & Technology Assessment (approx. 25% of total word count)**

* A review of the experts in the field that relate to your topic
* Evaluate the information
* Where is the evidence in the article to support author(s)’ claims
* What is missing from the article?
* What do other ‘experts’ state in their article
* How do these comments ‘fit’ with your topic?

**Execution approx 25% of word count**

### 2.3.1 Risk Assessment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Risk ID | Risk | Risk Severity | Risk Probability | Contingency Plan | Tools Used | Plan to Subvert Risk |
| 01 | Not enough participants to obtain meaningful data | High | Medium | Offer possible participants gifts such as pizza, fruit, juice etc. in order to encourage participation other students. | Participants, lecturers, email addresses | Contact lecturers and request that they send out emails requesting participants and invite students to participate |
| 02 | Game Not Developed Properly (excessively buggy/broken) | High | Low | Run consistent playtesting sessions prior to the experimental sessions in order to ensure bugs are quickly found and fixed. | Game Engine, Computer, Integrated development environment (IDE) | Risk cannot be fully subverted. Minimised through the full knowledge of the code and research of the techniques to be used coupled with time management. |
| 03 | No meaningful conclusions can be drawn from data gathered | High | Low | No contingency plan necessary as this would simply show invalid hypothesis | Computer with SPSS software installed | Successfully analyse data with SPSS |
| 04 | Participants drop out of experiment after agreeing to do so. | High | Low | Offer possible participants gifts such as pizza, fruit, juice etc. in order to encourage participation from other students. | Participants, lecturers, email addresses | Regularly contract prospective participants with reminders about dates and times scheduled |
| 05 | Game incomplete or unfinished | High | Low | None available as if the game is incompletely testing and further experimentation cannot be done | Game Engine, Computer, Integrated development environment (IDE) | Carefully plan stages and follow development timescale throughout the creation of the game. |

**Does the execution section**

* Clearly **discuss**  original problem?
* Clearly **analyse** the features of the problem?
* Clearly **relate** those features to the literature review’s conclusions
* **Indicate** where they support and or disagree with literature review’s conclusions
* **Explain** your ‘solution’, its approach, design & implementation/instrument
* **Justify** your ‘solution’, its approach, design & implementation/instrument
* **Present** your ‘solution’, its approach, design & implementation/instrument

**Evaluation & Discussion approx 30% of word count**

How can questions below be **demonstrated** in the report?

**Evaluate = attempt to form a judgement about, be specific about the basis for this judgement**

* What does the ‘solution’ its approach, design & implementation/instrument **contribute?** What are **key criticisms** of the solution its approach, design & implementation/instrument
* What are **the advantages/limitations** of the ‘solution’ its approach, design & implementation/instrument
* Do the results **identify** what is more or less important?
* Are the results **compared**  to other knowledge about the project?
* Is there comment on the **consequences** and **implications** of the results?

**Conclusions & Further Work approx 10% of word count**

* **Summarise** the main findings from the lit review, the execution, the evaluation & discussion chapters.
* **Explain** with evidence the final conclusions of your report. What do you want your reader to remember about the report?
* **Discuss** what else could take place to enhance, extend the project.
* Which direction could your project take in the future?

<http://www.rnib.org.uk/nb-online/video-computer-games-people-vision-impairment>

<http://www.rnib.org.uk/rnibconnect/technology/audiogames-and-playing-videogames-without-sight>

<http://game-accessibility.com/documentation/visually-impaired-gamers-where-to-go-what-to-play/>

<http://ifiction.org/>

<http://blog.aidis.org/audio-output-a-review-of-games-for-blindvi-gamers>

<http://www.eurogamer.net/articles/2016-03-29-meet-the-blind-gamer-with-a-killer-instinct>