# Contents

[1.0 Contents 1](#_Toc497831718)

[2.0 Introduction 3](#_Toc497831719)

[2.1 Glossary of Terms 3](#_Toc497831720)

[2.2 Background 3](#_Toc497831721)

[2.2.1 Accessibility in Gaming 3](#_Toc497831722)

[2.2.2 The Problem 3](#_Toc497831723)

[2.2.3 Development Solution 4](#_Toc497831724)

[2.2.4 Justification of Participants 4](#_Toc497831725)

[2.3 Project Outline and Research Question 4](#_Toc497831726)

[2.3.1 Risk Assessment 4](#_Toc497831727)

[2.3.2 Ethics Considerations (as approved by David Moffat) 5](#_Toc497831728)

[2.3.3 Project Development Lifespan 6](#_Toc497831729)

[2.3.4 Research Question 6](#_Toc497831730)

[3.0 Literature and Technology Review 7](#_Toc497831731)

[3.1 Development process 7](#_Toc497831732)

[3.1.1 Waterfall 7](#_Toc497831733)

[3.1.2 Agile 7](#_Toc497831734)

[3.2 Development engine 8](#_Toc497831735)

[3.2.1 Unity 3D 8](#_Toc497831736)

[3.2.2 Unreal Engine 4 9](#_Toc497831737)

[3.3 Development Platform 10](#_Toc497831738)

[3.3.1 Virtual Reality 10](#_Toc497831739)

[3.3.2 Computer 11](#_Toc497831740)

[3.3.3 Mobile 12](#_Toc497831741)

[3.4 Blindness in gaming 13](#_Toc497831742)

[4.0 Methods 13](#_Toc497831743)

[4.1 Dev methods 13](#_Toc497831744)

[4.2 Experiment methods 13](#_Toc497831745)

[4.3 Evaluation Methods 13](#_Toc497831746)

[5.0 Remaining tasks 13](#_Toc497831747)

[5.1 Finish Development and debugging 13](#_Toc497831748)

[5.2 Experiment carry out 13](#_Toc497831749)

[5.3 Final report 13](#_Toc497831750)

[5.4 Presentation 13](#_Toc497831751)

[6.0 Appendices 13](#_Toc497831752)

# 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 coordination 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.

## 2.2 Background

### 2.2.1 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 industry worth 99.6 billion dollars in 2016 according to 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 are predicted to increase past 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 with Audiogames.net showing an available 606 audio games at time of writing (Audiogames.net, n.d.).

While there are 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 upon 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 games genre. 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. Which would put the audio development of a game to the forefront and allow visuals to take somewhat of a backseat by comparison, which up to this point has not occurred.

### 2.2.2 The Problem

It is this is inherent lack of accessibility that is the problem within the games industry. This not only lacks the inclusivity which the industry is often known for but also leaves a large market of potential players untapped. This market, treated by many developers as empty or even non-existent considering 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).

This lack of inclusivity is largely a problem for any who wish to become part of the multimillion dollar competitive gaming industry. This is exemplified by the fact that currently in the industry there are no competitive games aimed at non-sighted individuals. As such, solving this problem and the methods to do so will be the primary focus of the technology and literature review sections of this report.

### 2.2.3 Development Solution

The proposed solution to this problem is the design and development of a game which would allow a non-sighted person to play at equal level with a sighted opponent. The primary solution is to use common audio game development techniques to develop this game and to design the game with accessibility at its core. To do this, research has been conducted on a variety of topics such as; audio game development techniques, possible game development technologies (namely engines and platforms), development processes and the common uses of gaming technologies in the treatment and lifestyles of those who suffer from a lack of sight.

### 2.2.4 Justification of Participants

This research and development project will primarily focus on those with complete blindness under the assumption that if those with no light perception are able to comfortably play the game then those with limited light perception will also be capable of this. This also serves to reduce the variety of required test subjects as well making test subjects easier to find and simplify the testing process. If possible, second stage tests will be conducted to include those with more limited degrees of visual impairment. This will serve to be able to widen the user base by confirming that those with less severe visual impairment are able to play the game. If necessary, though undesirable, test subjects will consist of sighted individuals who are made to wear blindfolds. This is simply a method -of managing the worst-case scenario in the instance that no subjects who suffer from no light perception can be found.

## 2.3 Project Outline and Research Question

### 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. |

### 2.3.2 Ethics Considerations (as approved by David Moffat)

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.

### 2.3.3 Project Development Lifespan

The initial gannt chart detailing the projected development of this project as intended at the time of the project proposal can be found in Appendix A. The new gannt char including the addition of realistic timescales as the project currently stands is found in Appendix B. As shown, the development of the project is being given the most time in order to ensure that the programme itself is completed in time, this stage will also incorporate the testing phase which will stretch into the second semester. After completion of testing to a satisfactory standard, the actual experimental phase of the project will take place. During the experimentation phase participants will be recruited to play the game. Each play session will be timed, and scores recorded using analytics software. These shall all be evaluated to determine the validity of the hypothesis presented in the evaluation stage

### 2.3.4 Research Question

“Can a competitive video game be developed which gives no advantage to users who are fully sighted compared to those whose sight is impaired?”

# 3.0 Literature and Technology Review

## 3.1 Development process

For the development of this project traditional and modern development processes must be considered. For the purposes of this report the traditional waterfall model will be compared with the more model agile methodology.

### 3.1.1 Waterfall

The waterfall development model is a predictive sequential development lifecycle which is split into specific phases. These phases are, in order; Requirement analysis, System design, Implementation, Testing, Deployment, Maintenance. Within this system each phase is only begun upon completion of the previous phase.

The sequential nature of this model means that it is simple to understand and implement, being task oriented in a humanistic view. This model is most useful when the product requirements are explicitly defined and rigidly implemented, this means there are no vague or nonspecific requirements as they are ironed out before any sort of design or development takes place and allows projects to have clear milestones. In this model when less specific details are provided certain phases take longer and become more difficult to implement causing a backlog of the whole process however. This staged approach leads to a lack of any working software until halfway through the life cycle of the project (TutorialsPoint, 2017) which, in the case of a task backlog can put a project far behind schedule. The knock-on effect this can have to the project impedes the testing and evaluation phases in the case that the requirements are not fully specified. This can often be the case in a game as they regularly need to be altered based upon the data retrieved from the evaluation stage of alpha and beta tests. To that effect regularly games employ this model with additional testing phases (alpha and beta) to allow changes to take place after initial development and returning to previous stages when it becomes apparently necessary.

### 3.1.2 Agile

Agile is an adaptive development methodology which takes an iterative cyclical approach to software development.

Each iteration involves aspects of all members of the development team going together through the following stages in order; planning, requirements analysis, designing, building, testing. Each of these iterations increment upon the features of the previous with the final iteration (ideally the release build) holding all required features (TutorialsPoint, 2017). Regular project sprints are done to determine progress of the project and identify any issues as they arise. This is a common methodology for games as it emphasizes regular progress meetings to allow regular bug detection. This can mean that development time is extended however with the testing phase in the cyclical nature of these iterations taking longer than initially hoped. Often considered a good development strategy for video games, this outputs a semi functional product very quickly to allow as proof of concept for client review or pitch, creating a vertical or horizontal slice after each iteration. This model takes an adaptive approach to software development compared to more traditional methods (such as the previously mentioned waterfall method). This development method can unfortunately fall down when complex features that cannot be broken down are introduced. In the case of a feature such as this the development time slows as a bottleneck occurs in whichever team is primarily responsible for that feature.

Despite this, the Agile development methodology is the best suited to the game development project proposed. Applying a version of Agile development methodology tailored for a single person team with self-imposed sprints and limitations for bug-fixing will hopefully prove to be an effective development practice.

## 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.

### 3.2.1 Unity 3D

Unity 3D is a development engine designed on 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 also a particularly portable development environment allowing porting to; 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 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 use of sight identify the location of a 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 time recording of in game analytics such as play session duration and the level at which the player quit. These analytics work through an events system integrated into unity examples off 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 developing games such as; Unreal Tournament, the Gears of War series, Bulletstorm and Paragon. All developed in the unreal engine, these games show intuitive use of physics, lighting and mechanical techniques that 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 off the feature list available (Games, 2017) there is no in-built analytics system at 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 in built analytics means that several analytics 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 of the programs ability to reach a target demographic. The ease of development 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 this 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 equipped to the headset. This then comes to 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 quick software setup via the Oculus Software download. This is 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 setup, 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 or 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.

For the reasons stated above, 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 more appropriate device. The use of the Vive’s body tracking system should allow greater ease for 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 meaning that if a peripheral is developed with drivers specifically for Windows or MacOS then there will often be either a Linux equivalent available for download or one of the Windows or MacOS emulators can be used to properly run the required software for the device to function. In the case of this development project the only require peripheral in the case of 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 ease of 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 for documentation for Unity 3D which supports all three operating systems means that each ope0rating system can be developed for with ease in this instance however to develop for MacOS a Mac computer is require 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 for development when using Unity 3D and each operating system is equally available for development, market penetration becomes the only metric to bias the decision of which to use. As such Windows become 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. According to the Newzoo 2017 games market report 42% of all game revenue being available 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 criteria to determine the most appropriate system for the development of this project.

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 touch screen 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 touch screen. 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 target would be feasible than that 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 for either mobile operating system are 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.

## 3.4 Blindness in gaming

Gaming as an industry has been a largely inclusive one, with specialised peripherals being created for those who require additional help. From eye trackers for those with locked in syndrome to trackballs for those with arthritis, the computer gaming industry allows a great deal of people to be embraced within this art form. Blindness is, as previously mentioned, being steadily included within the various facets of gaming, however the industry is slow to adopt certain technologies and techniques into the mainstream. This report will review and discuss some of these and the current state of the industry as a whole.

### 3.4.1 Blindness in casual Gaming

Blindness and visual impairment is quickly becoming a larger part of the casual gaming industry. In many instances binaural recording is used to create a fully immersive audio only world (Brennan, 2014). Games such as “Blind Legend” and “A Trial in St. Petersburg” attempt to fully involve the player through the use of an audio only environment. The development of this style of game has become easier with the creation of Google’s recently developed immersive audio toolkit, “Resonance Audio” which gives developers the opportunity to control where and how the audio within the game occurs (Fingas, 2017). This is designed initially for VR however could be used to more effectively develop non-visual 3D games.

Several technologies have been developed with the aim of

### 3.4.2 Blindness in competitive gaming

Using VR for blindness

# 4.0 Methods

## 4.1 Dev methods

## 4.2 Experiment methods

## 4.3 Evaluation Methods

# 5.0 Remaining tasks

## 5.1 Finish Development and debugging

## 5.2 Experiment carry out

## 5.3 Final report

## 5.4 Presentation

# 6.0 Appendices

AppendixA: ****

*Figure 1: Gantt chart Generated by Creately Gantt Chart Software (2017)*

References

Newzoo. (2016). *The Global Games Market 2016 | Per Region & Segment | Newzoo*. [online] Available at: https://newzoo.com/insights/articles/global-games-market-reaches-99-6-billion-2016-mobile-generating-37/ [Accessed 13 Oct. 2017].

Transversal, A. (2014). *How many people in the UK have sight loss? - RNIB - supporting blind and partially sighted people*. [online] Help.rnib.org.uk. Available at: https://help.rnib.org.uk/help/newly-diagnosed-registration/registering-sight-loss/statistics [Accessed 15 Oct. 2017].

Specsavers (2017). *The State of the Nation Eye Health 2017: A Year in Review*. [online] Royal National Institute of Blind People. Available at: https://www.specsavers-spectrum.com/wp-content/uploads/2017/09/RNIB-Specsavers-State-of-the-Nation-Report-2017.pdf [Accessed 15 Oct. 2017].

Audiogames.net. (n.d.). *AudioGames, your resource for audiogames, games for the blind, games for the visually impaired!*. [online] Available at: https://www.audiogames.net/list-games/listgames.php?mode=full [Accessed 15 Oct. 2017].

World Health Organisation (2010). *Global Data on Visual Impairments 2010*. [online] World Health Organisation, p.Page 5. Available at: http://www.who.int/blindness/GLOBALDATAFINALforweb.pdf [Accessed 15 Oct. 2017].

Grubb, J. (2014). *Gaming advocacy group: The average gamer is 31, and most play on a console*. [online] VentureBeat. Available at: https://venturebeat.com/2014/04/29/gaming-advocacy-group-the-average-gamer-is-31-and-most-play-on-a-console/ [Accessed 15 Oct. 2017].

TutorialsPoint (2017). *SDLC Waterfall Model*. [online] www.tutorialspoint.com. Available at: https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm [Accessed 25 Oct. 2017].

TutorialsPoint (2017). *SDLC Agile Model*. [online] www.tutorialspoint.com. Available at: https://www.tutorialspoint.com/sdlc/sdlc\_agile\_model.htm [Accessed 27 Oct. 2017].

Rongala, A. (2017). *Benefits of C / C++ over Other Programming Languages*. [online] Invensis Blog. Available at: https://www.invensis.net/blog/it/benefits-of-c-c-plus-plus-over-other-programming-languages/ [Accessed 31 Oct. 2017].

Technologies, U. (2017). *Unity - Manual: Physics*. [online] Docs.unity3d.com. Available at: https://docs.unity3d.com/Manual/PhysicsSection.html [Accessed 31 Oct. 2017].

Games, E. (2017). *Programming Guide*. [online] Docs.unrealengine.com. Available at: https://docs.unrealengine.com/latest/INT/Programming/index.html [Accessed 31 Oct. 2017].

Games, E. (2017). *Unreal Engine Features*. [online] Unrealengine.com. Available at: https://www.unrealengine.com/en-US/features [Accessed 3 Nov. 2017].

Hesketh, D. and Campbell, L. (2017). *Bright Void*. Glasgow: Whiteboard Games.

Ergürel, D. (2017). *The latest virtual reality headset sales numbers we know so far*. [online] Haptical. Available at: https://haptic.al/latest-virtual-reality-headset-sales-so-far-9553e42f60b5 [Accessed 4 Nov. 2017].

HTC Corporation (2017). *VIVE™ United Kingdom | Buy Vive Hardware*. [online] Vive.com. Available at: https://www.vive.com/uk/product/ [Accessed 4 Nov. 2017].

HTC Corporation (2016). *http://www.htc.com/managed-assets/shared/desktop/vive/Vive\_PRE\_User\_Guide.pdf*. [ebook] New Taipei City: HTC Corporation, pp.20-25. Available at: http://www.htc.com/managed-assets/shared/desktop/vive/Vive\_PRE\_User\_Guide.pdf [Accessed 4 Nov. 2017].

Stuff.com (2016). *Oculus Rift vs HTC Vive*. [online] Stuff. Available at: https://www.stuff.tv/my/features/oculus-rift-vs-htc-vive/setup-and-requirements [Accessed 5 Nov. 2017].

Merabet, L., Connors, E., Halko, M. and Sánchez, J. (2012). Teaching the Blind to Find Their Way by Playing Video Games. *PLoS ONE*, [online] 7(9), p.e44958. Available at: http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0044958 [Accessed 9 Apr. 2017].

Technologies, U. (2017). *Unity - Google Daydream*. [online] Unity. Available at: https://unity3d.com/partners/google/daydream [Accessed 5 Nov. 2017].

Valve Corporation (2017). *Steam Hardware & Software Survey*. [online] Store.steampowered.com. Available at: http://store.steampowered.com/hwsurvey?platform=combined [Accessed 7 Nov. 2017].

Lyer, K. (2017). *The Best Operating Systems For PC Gaming*. [online] TechWorm. Available at: https://www.techworm.net/2017/02/best-operating-systems-pc-gaming.html [Accessed 7 Nov. 2017].

Microsoft Corporation (2017). *Your request appears to be from an automated process*. [online] Microsoft.com. Available at: https://www.microsoft.com/en-gb/store/d/windows-10-home/d76qx4bznwk4/1NT3 [Accessed 7 Nov. 2017].

McDonald, E. (2017). *The Global Games Market 2017 | Per Region & Segment | Newzoo*. [online] Newzoo. Available at: https://newzoo.com/insights/articles/the-global-games-market-will-reach-108-9-billion-in-2017-with-mobile-taking-42/ [Accessed 7 Nov. 2017].

Gartner (2016). *Gartner Says Five of Top 10 Worldwide Mobile Phone Vendors Increased Sales in Second Quarter of 2016*. [online] Gartner.com. Available at: https://www.gartner.com/newsroom/id/3415117 [Accessed 7 Nov. 2017].

Allied Business Intelligence (2011). *97% of All Smartphones Will Have Touchscreens by 2016*. [online] Abiresearch.com. Available at: https://www.abiresearch.com/press/97-of-all-smartphones-will-have-touchscreens-by-20/ [Accessed 7 Nov. 2017].

Brennan, C. (2014). *'Video-less' 3D game made for blind players*. [online] BBC News. Available at: http://www.bbc.co.uk/news/technology-28757186 [Accessed 8 Nov. 2017].

Fingas, J. (2017). *Google built a spatial audio kit for games and VR*. [online] Engadget. Available at: https://www.engadget.com/2017/11/06/google-resonance-audio/ [Accessed 8 Nov. 2017].