Dissertation

**Development of a Mobile Game utilising Stimuli to Improve Human Reaction Time**

A Final Year Project

Jake Walder

BSc Games Programming

# Abstract

The human reaction is fundamental to any everyday activity, and the speed of this reaction can often change the severity of the consequence. This project will design and implement three small games within a mobile application developed using Unity’s (2018) game engine, and will aim to improve the player’s reaction time. The games will utilise visual, auditory and tactile stimuli to induce a reaction from the user, which will then be recorded. The games will be tested by users over a set period of time, and the data will be recorded to evaluate how well the games can improve the human response time.

# Contents

[Abstract 2](#_Toc10229585)

[Contents 3](#_Toc10229586)

[Introduction & Objectives 5](#_Toc10229587)

[Introduction 5](#_Toc10229588)

[Aims 7](#_Toc10229589)

[Objectives 7](#_Toc10229590)

[Hypothesis 7](#_Toc10229591)

[Background Theory & Design 8](#_Toc10229592)

[Background Theory 8](#_Toc10229593)

[Design Process Overview 9](#_Toc10229594)

[Application Design 10](#_Toc10229595)

[Shooting Game Design 10](#_Toc10229596)

[Driving Game Design 11](#_Toc10229597)

[Dodging Game Design 13](#_Toc10229598)

[Development Methodology 14](#_Toc10229599)

[Development Overview 14](#_Toc10229600)

[Shooting Game Development 15](#_Toc10229601)

[Driving Game Development 16](#_Toc10229602)

[Dodging Game Development 17](#_Toc10229603)

[Testing Methodology 19](#_Toc10229604)

[Testing Overview 19](#_Toc10229605)

[Controlling Variables 19](#_Toc10229606)

[Expected Findings 20](#_Toc10229607)

[Testing Results – 2000 Words 21](#_Toc10229608)

[Reaction Time Results 21](#_Toc10229609)

[Age Influence on Results 24](#_Toc10229610)

[Results Discussion 24](#_Toc10229611)

[Reaction Time Discussion 24](#_Toc10229612)

[Handedness Evaluation 24](#_Toc10229613)

[Conclusion 24](#_Toc10229614)

[Project Analysis 24](#_Toc10229615)

[Application Evaluation 24](#_Toc10229616)

[Conclusion 24](#_Toc10229617)

[References 25](#_Toc10229618)

[Appendices 26](#_Toc10229619)

* Official front cover (see appendix)
* Disclaimer
* Acknowledgements
* Abstract
* Contents page(s)
* Chapter 1 Introduction including aims and objectives – 1000 Words
* Chapter 2 Background theory and Design process – 2500 Words
* Chapter 3 Details of the development process and test/evaluation methods. – 1500 Words
* Chapter 4 Testing/Evaluation methods – 2000 Words
* Chapter 5 Results of test and evaluation of the software and/or hardware product. – 1500 Words
* Chapter 6 Discussion of results and conclusions including critical reflection. – 1500 Words
* References cited and listed using BU Harvard format.
* Appendices - the appendices should include relevant supporting documentation and the approved ethics checklist.

# Introduction & Objectives

## Introduction

A human response can alter the severity of any everyday situation, so it is important to keep the time of these responses as low as possible. Whether driving or playing a sport, the outcome of every reaction will be defined by both its speed and accuracy.

It has been said that playing action video games *‘significantly reduces reaction times’* (Dye, Green and Bavelier, 2009). In order to prove this, participants played action games such as Call of Duty 2 (2005) and Unreal Tournament (1999), and their response times were tested after a period of use. These discoveries led to the motivation behind this project which is to create a series of three small games. These will require the player to react quickly, improving the speed of their responses over time. The games will be simple versions of popular action game types and features (i.e. driving and shooting) that demand fast and accurate responses. They will induce the reactions by displaying a stimulus on screen for the player to interact with, and then the reaction from the player will be recorded (from between the stimulus being displayed and the player’s reaction) by the game and saved to the device for later use. Action games are the optimal game type for improving response times due to their nature of fast paced gameplay, and requisite for quick decisions and actions by the player. Once the player has finished a game session, all of the response time readings will be output and then compared to any previous plays. This should show an increase in response time from the first play session of the game when compared to the last, which will then provide proof of video games’ ability to improve response times and accuracy.

There are various types of stimuli that cause reactions within games. Technology has the capability of using visual, auditory and tactile stimuli to produce a reaction from the user. Mobile phones are a great example of a device that can utilise all three (See Fig. 1 for an example of a mobile game), for instance notifications on a mobile phone can be reported in the form of visual information, auditory sounds and tactile vibrations to the user. For the sake of this project, only visual stimuli will be used to test the reaction times, as a study showed that the response to *‘tactile stimuli was significantly shorter’* (Ng and Chan, 2012), followed by auditory and then visual stimuli respectively. As visual stimuli are the slowest of the three to generate a response, this allows for the improvement of the reactions to be greater and clearer to analyse. Mobile phones will be the perfect platform to test these responses as they have an interactive touch screen, allowing for immediate response from the user that can be used within a video game scenario. This project will also test the concept of how decision making within a video game scenario can influence the speed of a response, as the three games will require an increasing amount of decision making to play.

Figure 1: A simple mobile phone game (Greenbot, 2015)

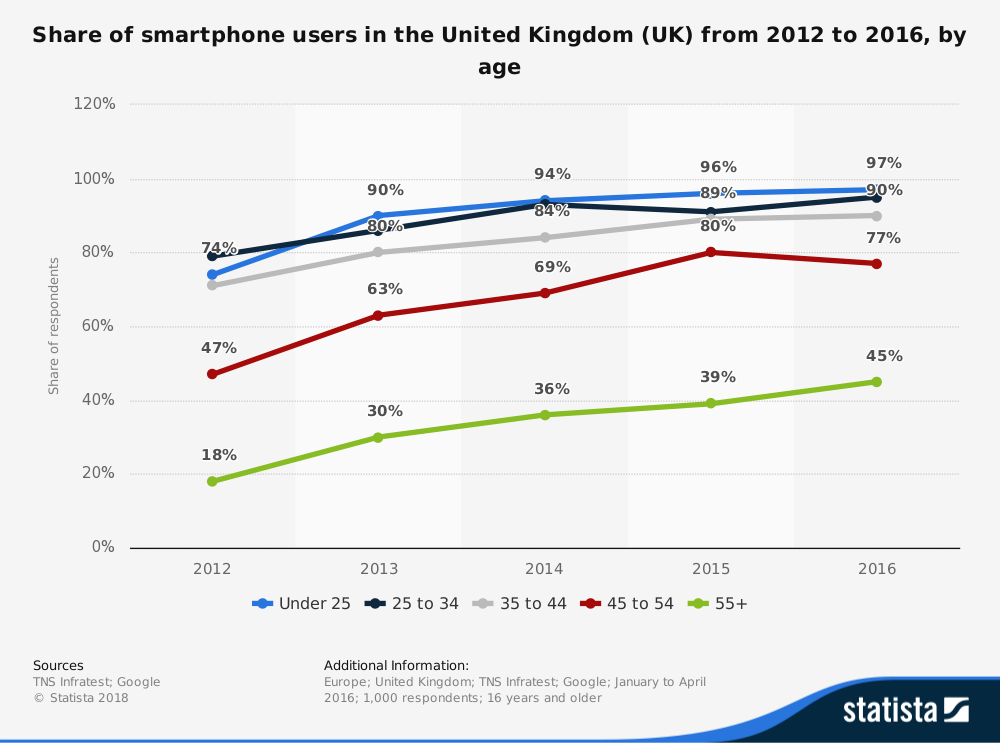
Not only are mobile phones a great platform for these simple games, but they are also ideal for making these games very accessible to the public. There is a huge increase in the number of smartphones per capita; the population of smartphone users was at a rate of 75% of males and 73% of females in the UK in 2016, which means that the population of users increased by 21% and 25% respectively since 2012. (Statista, 2018). Fig. 2 below also shows the main age demographics of smartphone users and the growth of each group.

Figure 2: The percentage of smartphone users in the UK (Statista, 2018)

These games will be simple and accessible, yet retaining a challenging and entertaining factor, making the improvement of one’s reaction time enjoyable. This means that this project will enable both gamers and non-gamers to improve their response time for any reason they deem necessary, whether it’s to improve their ability in sports, driving, or video games.

## Aims

* The primary aim is to create a mobile application for Android using Unity (2018), consisting of three games that test response time.
* The secondary aim is to test multiple users’ response times over a set period, and compare the results to show any improvement of their response time.

## Objectives

* Create a clear and easy to navigate User Interface (UI) to store the 3 games that will be made.
* Create the three games which are as follows:
  + A shooting game where random targets will appear on screen that the player must shoot.
  + A driving game where a random course will be created, and the player must choose the correct direction when prompted.
  + A dodging game where asteroids will appear on screen that the user must dodge before being hit.
* These three games have their own objective intentions:
  + The shooting game must test only the speed of the player’s reactions to visual stimuli.
  + The driving game must increase the difficulty of the game slightly by forcing the player to make a decision as well as react, but this decision must be simple and sometimes pre-emptive.
  + The dodging game must increase this difficulty by forcing the player to make a decision that is totally on their own merit, and must never be pre-emptive.
* Test a group of users’ reaction times over a set period of play-time to assess the base state, and then determine any improvement of reaction time through extended participation, by comparing visual stimuli to reactions caused with the addition of auditory and tactile stimuli.

## Hypothesis

The expected outcome of this project is to show that action video games have a positive influence on the reaction time of the user, as well as improving their decision-making skills, and that these influences can be demonstrated and defined during the playing of these games.

# Background Theory & Design

## Background Theory

As outlined in the introduction, there is a justification to investigate the current knowledge of video games’ effects on the human response time. The project will aim to discover the existing understanding of these effects and develop it by monitoring how including decision making into the video games can affect the response time. In this section, the literature survey that has been undertaken will be discussed and it will review the details of the critical findings. This literature review describes the effects of visual stimuli and the inclusion of decision making, as well as video games’ importance in these discoveries, and how these two factors are implemented together.

A video game requires reactive inputs from the user in order to play. In the case of most modern video game platforms, three different types of stimuli are used to initiate these reactions: visual, auditory and tactile. For testing purposes, the time between the stimulus occurring and the user reacting can be recorded. It was discovered by Ng et al. (2012) that each stimulus generates different speeds of reaction, the fastest reaction being caused by tactile stimuli, followed by auditory and visual respectively. However, for a video game, visual stimuli are the only essential stimuli and will therefore be the only stimuli tested in this project. Video game players have already been proven to have faster reactions than people who don’t by Castel et al. (2005). Orosy-Fildes et al. (1989) also observed that reaction times dropped after the usage of video games, along with no decrease in reaction times in users that did not participate in any video games. Combining these findings along with the knowledge that 74% of the UK population owned a smartphone in 2016 in a growing trend (Statista, 2019), alludes to the credibility of using mobile phones for testing the hypothesis. This credibility is extended by the fact that mobile phones have the ability to allow input directly onto the screen displaying the stimuli, therefore allowing the user to directly interact with said stimuli.

Dye et al. (2009) said playing action video games ‘requires rapid processing of sensory information’, requiring the player to make quick and accurate decisions, faster than any typical daily activity. Playing First Person Shooter (FPS) video games, a type of action video games, have also been ‘associated with increased cognitive flexibility’ by Colzato et al. (2012), thus corroborating the use of action games as the most effective video game type to improve human response time. Accuracy is also important alongside the speed of a reaction, and must be considered during testing by increasing the importance of decision making throughout the three games. Greenfield (1984) discussed the importance of video games ability to improve ‘sensorimotor skills such as eye-hand coordination’, which Green et al. (2006) went on to confirm, along with ‘decreased reaction times’ and ‘augmented manual dexterity’. Boot et al. (2011) also claims that ‘video game training enhances cognitive performance on tasks’ which further describes the importance of the role that video games play on these reactions and imply the significance of measuring accuracy during the testing procedure, as it will most likely be improved over the course of the testing period. The quality and speed of the decisions are easily compared by using the total time playing the game, implies how many correct decisions the user made, with the speed of the reactions. It is also crucial to have repetitive testing during the testing period, as Greenfield et al. (1994) found that video games experience ‘produced a significant decrease in response time’, especially when repeated over a period of time. This confirms the decision to have the user play the games three separate times over a set period, allowing for improvement to be shown over a set time.

With this information in tow, the importance of video games in relation to player’s response times can now be understood fully. The importance of the types of stimuli is shown and from that, the choice of using only visual stimuli is validated, as well as the process of repetitive testing in order to understand the gradual improvement video games can have. Both response time and accuracy must also be considered when measuring during the testing procedure, in order to generate an accurate representation of the response quality from the user. From these findings, the games can be developed correctly and used to test the response times of the users, which can then be evaluated effectively.

## Design Process Overview

Once the literary review had taken place, the design and development of the application could begin. The goal for this project is to consider any previous work/games, and then refine and improve upon them during the planning process. The design will begin with a simple overview of the application outline and then detail the three games. The games were designed while bearing in mind how the response times were to be calculated and stored. The reaction times will vary due to the amount of decision making required by each game. Due to these differentiating factors, the response times were not be able to be compared to one another, and left the user with three separate response time averages when the games were played, which were considered when it came to the testing. The games’ visual design is simple to accommodate for mobile phones’ smaller screens, relative to the size of a computer monitor. The UI for the games is also clear and basic to reduce as much confusion for the user (which was minimised by an application run-through that will be explained before the testing).

The application will be bespoke designed to a Samsung Galaxy S9+ (2018), as this is the device that is used during the testing. The assets that will be used for the game will come from a simple asset modelling website called Kenney (2018). These assets are clear and simple, enhancing the basic design that the application will consist of.

## Application Design

The design of the application aims to be simple and clear to navigate, and to only display the necessary information to the user. Any information gained during the testing for evaluation purposes will be saved to a hidden file that will be used after the testing is complete. The saving of the data is achieved by assigning a unique identifier to each user, which is then used when playing the games, and can be chosen on the title screen menu. The testing was supervised throughout to assure the correct identifier was used to store the data. This means that the only information displayed is the identifier and the countdown throughout the games.

Each game utilises the Kenney (2018) assets pack to keep a consistent design. This asset pack includes a multitude of assets for use in 2D games, which are perfect for these simple mobile games. These assets are simple in design to clearly indicate the purpose within the games, and when used alongside the clear user interface, interaction within the application and games is intuitive.

## Shooting Game Design

The first game that was developed was the shooting game. The concept of this game is very simple: a series of targets appear on screen in a random location, and the player must tap on them as soon as they appear. Up to three targets can appear at once but in all situations only one can be tapped. The reason behind the inclusion of multiple targets is to add the ability to analyse the preferences of tap location and comparing it to the handedness of the user. While doing this it is possible to then also compare the speed of the reactions for said locations, and draw a conclusion on whether handedness can play a part in the speed of a reaction.

The locations of the targets are randomly selected, but there will be a set number of predetermined locations that must be used be used. The game area is split into 18 locations in the form of a standard grid. When the targets spawn, each location must be used at least once throughout the game, in order to evenly spread the locations and to control that variable. This prevents the targets from occurring differently between users, creating an even scenario for each user. There is also a countdown timer that counts for two minutes, and when it reaches zero, the game ends.

When it comes to the response calculation functionality, the stimuli must be clearly defined in order to calculate the response times from the exact moment of the stimuli appearing, and the user responding. In the case of the shooting game, the targets appearing are the stimuli, so therefore the time must be calculated between then and the input from the user.

Figure 3 shows a mock-up of how the assets given could be used, however the game that was designed for this project uses a simpler user interface and design. The figure also shows how the simple design and bright contrasting colours allows for clear and easy indication for stimuli. The clear indication is essential for the game, as the stimuli must be obvious to the point that it cannot be considered as a mitigating variable during testing, and that it assures the functionality of a stimulus.

Figure 3: Example of the shooting game assets (Kenney 2018)

## Driving Game Design

The next game that was created was the driving game. The concept for this game was for the player to follow along a track, and when prompted to tap either side of the screen to change the direction that they are travelling. There are 3 different direction types that can occur, which are left, right, and split. When left or right appear, the player must tap on the corresponding side of the screen to change to that direction. The next direction will be the opposite of the previous in order to bring the player back to the middle of the road. When the next direction is split, the player can choose to go either left or right, but then the same concept applies where they must re-join the middle of the road.

The driving game is intended to be a slight increase of difficulty over the previous shooting game, by adding the chance of a game ending prematurely if the wrong direction is chosen. This means that the player must make a decision before reacting. The results of this can then be correlated between how quickly someone reacts and how long they last within the game before making a mistake, which is translated by recording how long the player lasts while playing the game. The direction is not always random, as the direction change required from the player will be known for every other direction change that occurs, due to the course always returning to the centre. This keeps the game from being too difficult, as well as allowing for the controlling of values, and not having extreme cases causing anomalies during testing.

The response calculation for the driving game is defined from between when the next direction that occurs is displayed on screen, to when the player taps on either side of the screen to choose their direction. The direction is given in the form of text, in bold, in the middle of the screen. Instead of this indicating a direct action for the game, these stimuli indicate to the user that a decision needs to be made, which is how the difficulty is increased, and how the objective to test the user’s decision-making skills is implemented.

Figure 4: Example of the driving game assets (Kenney 2018)

Figure 4 shows a sample of a potential design using the assets chosen, but in the same fashion as the shooting game, a simpler design choice was made in order to declutter the screen, and make sure the stimuli was as clear as possible. This is why a clear and bold font in a contrasting colour to the background works, to clearly indicate when the stimulus occurs.

## Dodging Game Design

The final game that was designed was the dodging game. This game was designed to be the most difficult of the three, and therefore had to include the most amount of player decision interaction. The aim of the game is to dodge the incoming asteroids, but this can only be done in one move with a single tap. The directions of the asteroids appear on screen, and the user must use this information to make a decision and to move the ship as quickly as possible. The player can move the ship to any position they choose, but can only do it once. The benefit of having the free movement of the user in the game means that the position of the taps is completely free choice, allowing for the clearest demonstration of handedness and its influence on tap position. Between one and three asteroids can appear at once, so when there are more, the moves that are possible for the player to make are decreased. As the asteroids get closer, the number of possible moves also decrease, meaning that the response time of the user is critical.

The stimuli for the dodging game are the arrows that indicate the direction of the asteroids path. This means that when the arrows are shown, the player knows the direction and origin of the asteroids, and can therefore make a decision on where to move in order to dodge the asteroids. Using the reaction of the player, the reaction time is calculated from between the arrows appearing and the user reacting.

Figure 5: Example of the dodging game assets (Kenney 2018)

Figure 5 illustrates the assets of the dodging game used in context, yet to follow the design of the previous two games, the final game has a simpler design. The indicators for the direction of the asteroids appear as arrows around the ship, and they point towards the ship from the direction of the asteroid. The arrows were chosen to be white to contrast the dark background, making them distinct.

With the increased difficulty in the game, it is assumed that the time that players will last while playing the game should be smaller than the previous. However, with the added pressure of difficulty, the desire is to induce a more urgent response from the user, which would exemplify the effect that more intense action games have on the user. This increase of difficulty across the games is expected to show a decrease in reaction times across the games.

# Development Methodology

## Development Overview

To develop the application, Unity (2019), a vast and powerful game engine was used in conjunction with C# as the coding language, and Android (2019) was the target platform for the application, however the application also runs on Windows (2019). Since the design choices had been made, including the use of the assets from Kenney, the development could start implementing the games straight away. The title screen was made first, as this is where the user can access the three games. The identifier for each user is also set on this title screen. To access games, there are 3 buttons that the user can press to enter the game scenes. Since the application was designed to have very little amount of input required from the user, the simple UI chosen required only a small amount of interact ability. The only other button visible on the title screen is the identifier selector. When pressed, this button shows a dropdown list where the supervisor for testing can select the right identifier, and also a button that allows for the supervisor to create a new identifier file for user data to be assigned and saved to. Each game also uses the same system to record the reaction time of the player, which is engrained into each game differently due to their varying nature. When the stimuli are shown to the player, the game starts to record the time that it takes for the player to make an input. When this time is recorded, the exact location of the input is also recorded. Then, at the end of the game, the best, worst, and average reaction time is calculated and stored along with the total time played for that session. The data is stored into a JSON file that is associated only to that user by using the current identifier that was assigned before starting the session. This data was then analysed and evaluated after the testing was complete.

## Shooting Game Development

The first game that was developed was the shooting game, as this had the simplest concept, but it was also the most important game, because this was the only game where the user could not fail. This assures that a full sample of data was collected during every test. As every test would span the whole time, each playthrough needed to be similar to the last, yet still random. This meant that the inclusion of controlled random target locations and target multiples was crucial. In order to do this, every variable that was randomised, had to be defined beforehand and shuffled when the game began. Consequently, every playthrough will be somewhat similar to the last, but still completely random, so that patterns could not be detected. With this implemented, a timer could be set up that can countdown until the end of the game. This timer compensates for the time in between spawning phases of targets, as well as an exaggerated user reaction time. This means that every user will get a chance to hit every available target, but if their reaction time is considerably too slow, the game will cut short before all of the targets could be hit, but this is very unlikely and is just in place in order to control testing time.

One of the random but controlled variables are the locations. These locations are stored as a struct, with a reference to the game object, a Boolean that defines whether the location has been used, and the location’s position. There is then a list of every possible location within the game, which is defined as the game starts. Then as the game goes through, targets are spawned in one of these positions randomly, but when a target is hit by the user, that location is then set to used and cannot be used throughout the rest of the game. There are three separate lists of locations for each type of spawn: single, double, and triple spawns. This means that if a single target is hit in one location, a double target can still spawn in that location. The locations are only set as used for the same type of spawn, not restricting the use of the locations between spawn types. Furthermore, if a double or triple spawn occurs, it is set so that at least one target must appear on either side of the screen, which is used to embellish the testing for handedness, and to further control the randomness of the locations.

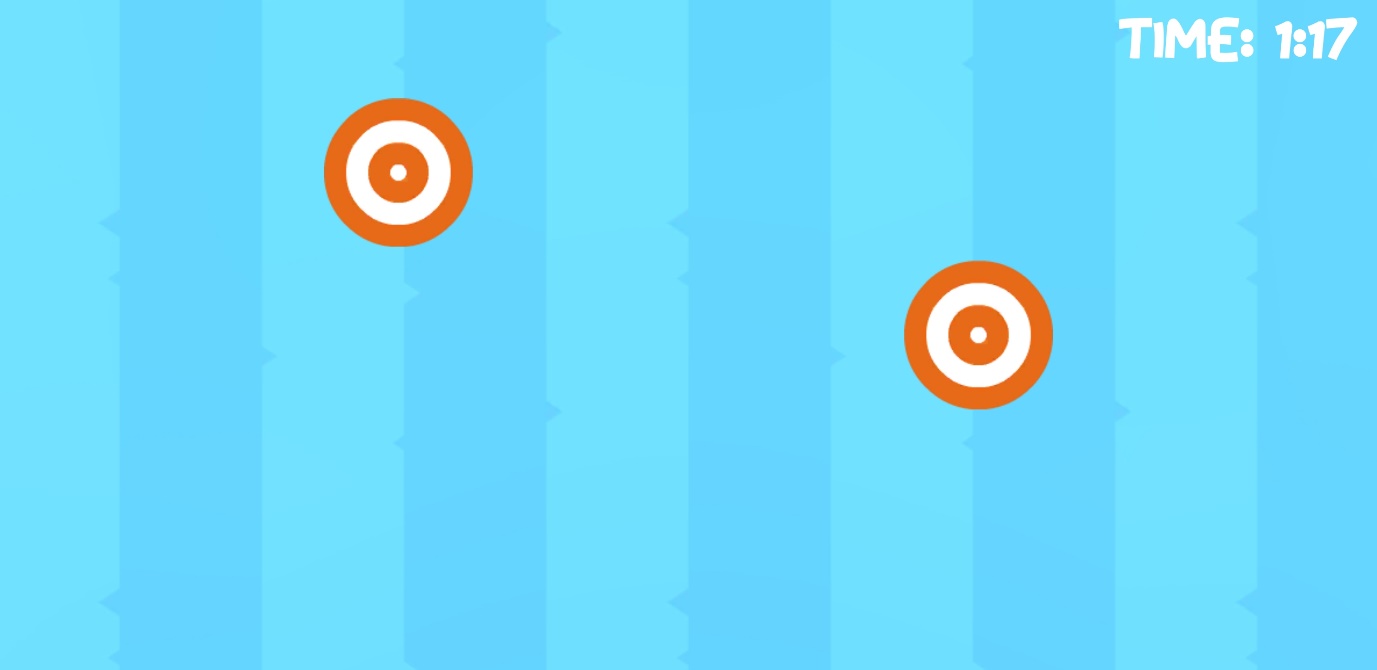
Figure 6 shows an example of the shooting game in session. The UI is clear to indicate how much time is left in the game, as well as the contrast of colours to increase the visibility of the targets themselves. It should also be noted that the device vibrates when a target has been to hit, to reaffirm the user of a valid response being accepted.

Figure 6: An example of a shooting game session

## Driving Game Development

The driving game was the next game that was developed. The core design of the game uses sprite tiles as the road, which were formed into larger sections of road that correlated to 4 different states of the game: middle, left, right, and split. The middle is the default state of the game, where the road always returns to in between road sections. Left and right are the sections that the player must move to when prompted by the stimuli, and the split section is where the player can choose either side to move to, but must return back to the middle when the section ends. The car moves only if an input has been made, and when the car hits the trigger section of the road. If a movement has not been made by this trigger point, the game will end, as well as if the wrong movement has been made.

As the games are all of the same vein, the road directions and the lengths of the road sections are all predetermined. However as these were simpler to define, they were stored using predefined arrays. These are then shuffled when the game starts. There is a fixed number of each sections, as well as a fixed number of lengths, which are shuffled at random to give each session a random order, but with a fixed overall outcome, therefore controlling all variables and denying the ability to notice patterns.

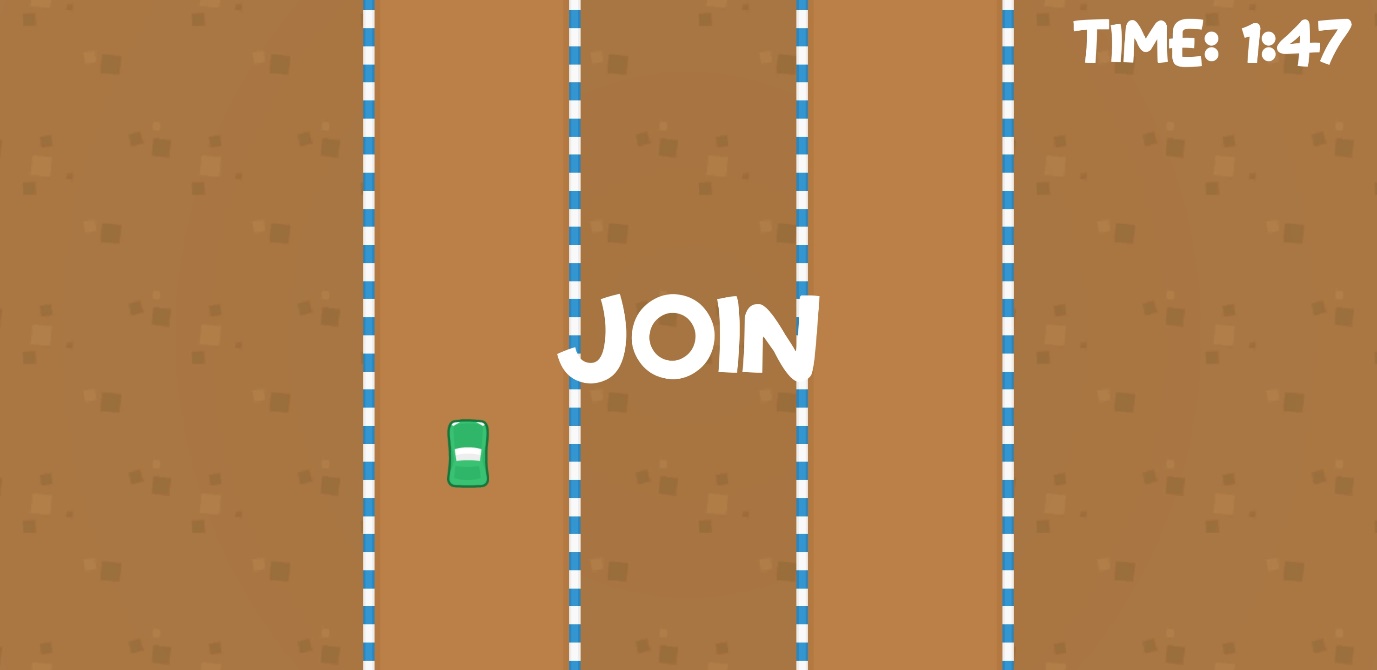
The nature of the stimuli for this game works differently to the shooting game, as the stimuli tells the user information, rather than giving a target to hit. This means that the response time has to be calculated by only recording the time between a direction being shown, and a valid input being made to the game. This is done by only allowing the user to interact with the game at a set time between the direction being displayed and the first input being made. The fact that the user has to process the information before reacting also augments the difficulty, as they must make a decision before reacting. If they make the wrong decision, the game will end and the testing for that session will be cut short. This loss of time can be stored however and can be compared to other users and subsequent test sessions that the user has undertaken.

Figure 7: An example of the driving game in session

Figure 7 shows an example of the driving game being played, where the player is currently in the split position, and is being prompted to move back to the middle. It is at this point that the player would be required to tap the right-hand side of the screen as soon as that prompt was shown. The stimulus is in an easily legible font in a contrasting colour, and the device will vibrate when the user has responded to the stimulus.

## Dodging Game Development

The final game that was developed was the dodging game. The aim of this game is to dodge any incoming asteroids using only one move. The player starts in the middle of the playing area and arrows appear around the player’s ship as soon as the asteroids spawn. These arrows are activated depending on the origin of the asteroids, and indicate the path of which the asteroids are travelling. These arrows are used as the stimuli for this game, and in a similar sense to the driving game, they provide information for the user to interpret and react to. However, to increase the difficulty further from the driving game, the stimuli is not an instruction and is only information labelling the direction of the asteroids, therefore requiring more tasking engagement from the user. Once the player has made their decision, they can tap on the screen once to move to that specific position. This complete freedom of movement also increases the responsibility of the player by forcing them to make their own decision, increasing the difficulty.

The controlled variable in this game is the number of asteroids that spawn at once. There can be between one and three asteroids that appear, but the number of these is predetermined in an array, which is shuffled as the game begins. As soon as the asteroids spawn, the correlating arrows will be shown to indicate the paths of the asteroids; these are the stimuli for this game. The arrows are always shown in a set perimeter around the ship, as the asteroids use the ships position to calculate where to spawn. The response time is recorded from between when the arrows are shown and when the user makes an input. If the player fails to make a movement that avoids the asteroids, the game will end upon collision, otherwise the game will carry out until the timer runs out.

Figure 8: An example of the dodging game in session

Figure 8 shows an example of the dodging game being played, and shows an instance where the player is shown the paths of the asteroids and must make a movement in order to dodge them. The arrows are a contrasting colour to the background in order to clearly indicate the stimuli, and the device will vibrate when the player taps to move the ship.

# Testing Methodology

## Testing Overview

The testing was carried out upon completion of the application. Firstly, the participants will provide two small pieces of information in order to compare the results after testing; the information given will be their age, and their dominant hand. The user will also be given a unique identifier that is assigned by the testing supervisor. The participants will play each game two times consecutively, and then this procedure will be carried out three times throughout one week, with three days in between tests (e.g. Monday, Thursday, and Sunday). In order to allow the users to get accustomed to the games, a brief will be explained before playing each game that details everything they need to know about how the games function. They will also be able to play each game once on a build that does not record data, so that when they play the games for testing purposes, their results are only recorded when they have this prior knowledge and understanding. It is important to not only explain how the games function, but to also play the games to get a feel for the controls and experience the game first-hand. This prevents the user from experiencing any unexpected scenarios, and clearly illustrates what is detailed in the brief.

## Controlling Variables

The games were developed in order to control all of the necessary variables, in order to control the results and make them valid. It is important to control variables ‘to clearly identify the relationship between an independent variable and a dependent variable’ (Carpi and Egger, 2009); the independent variable in this experiment being the different users testing the games, and the dependent variable being the reaction time from the user. The nature of the controlled variables in this application revolves around keeping each game session uniform to the previous, but also stopping the user from noticing any patterns in the gameplay by randomising the games’ stimuli. This therefor controls the variables by limiting the variations in stimuli production, yet randomising the variations to prevent the user from making any pre-emptive reactions. This process changes the variables from random to controlled by having only a limited number of scenarios that are possible, and that are not dissimilar to one another. Doing this also validates the experiment further when using these discrete random variables, as it imitates the reality of video games, which are rarely ever controlled to one hundred percent. With all of this under consideration, it allows for the data to be accurate in order to show any improvement over the course of the testing. These results are then compared to evaluate the application’s ability to improve the user’s reaction time.

The shooting game controlled the number of targets spawning (one, two, or three), and how the locations of the targets are spread out in order to not have a target spawn in the same place twice. It also prevented from targets spawning on only one side if multiple targets spawn, allowing for a proper evaluation of how handedness can affect reaction times. With these variables controlled, the dependent variable can be accurately measured, as each session should provide scenarios where the targets appear in similar but not identical situations.

To control the driving game, the number of direction changes and lengths of each section were controlled. This assured every session included an even number of directions (left, right, and middle), as well as having an even amount of lengths for each section. This presents every session with random combinations of sections, but they are evenly distributed to keep the randomisation discrete.

Finally, the dodging game controlled the number of asteroids that could appear in the game at once (one, two, or three). When the number of asteroids increases, the difficulty increases, so in order to keep the game at a consistent difficulty for each user, this was controlled. Keeping the difficulty consistent assures the fact that each user will experience a similar session, not only from one to user to the next, but from between testing sessions.

## Expected Findings

After the testing is complete, there are predictions for the outcome of the results. An overall improvement of reaction time across the users should be found, assuming that the research that was carried out during this project is accurate. An overall increased length of playing time in the driving and dodging games should also be found, as the user should become more and more competent after a series of play-throughs. The results will also allow for a couple of discussions using the data provided from each user. It will be possible to see how age influences reaction time, and whether or not there is a larger room for improvement if their response time starts slower than average. It will also be possible to see how a user’s handedness preference can influence the positions of the taps within the shooting game. When all the testing is done and the data is stored, it will then be possible to make a judgement on how action games and their concepts can have an effect on the user’s response time, even when boiled down to a simple demonstration.

# Testing Results – 2000 Words

## Reaction Time Results

Now that the testing is complete, the data that has been recorded can be analysed. The data has been retrieved from the JSON files where it was stored on the device and has been organised into tables for each testing session. The tables display the best, worst and average response times from each user for each game, as well as the total time that each session lasted. The data is shown in seconds, to three decimal places to show a clear but precise amount of detail, however the data was stored at 15 decimal places to assure optimum accuracy.

Figure 9: Test 1 data

Figure 9 shows the data collected from the first test. This data is used to set a precedent for the rest of the results. This table shows how the core data was stored and how the data can be compared to the other tests. It also displays the data associated with each user’s identifier, and also displays their age to get an understanding of the user that is associated.

Figure 10 shows the data that was recorded from the second test. Here we can start to compare the results to see if there are any improvements. Immediately, it can be seen in the shooting and dodging games that there has been an overall decrease in response times when averaged out across the users. It can also be seen that the time spent playing both the driving and and dodging game has increased, meaning that the users’ reactions are improving in order to prevent them from failing these games early. The time spent playing the shooting game has also decreased, further confirming the decrease in overall reaction times.

Figure 11: Test 3 data

Figure 10: Test 2 data

Figure 11 shows the final test’s data. This test also shows a further improvement on reaction time across users. What can clearly be seen in the last test is that 75% of the users managed to complete the driving and dodging game, which is a large improvement over the 45% that finished these games in the second test, which was also an improvement on the completion rate of 25% in the first test.

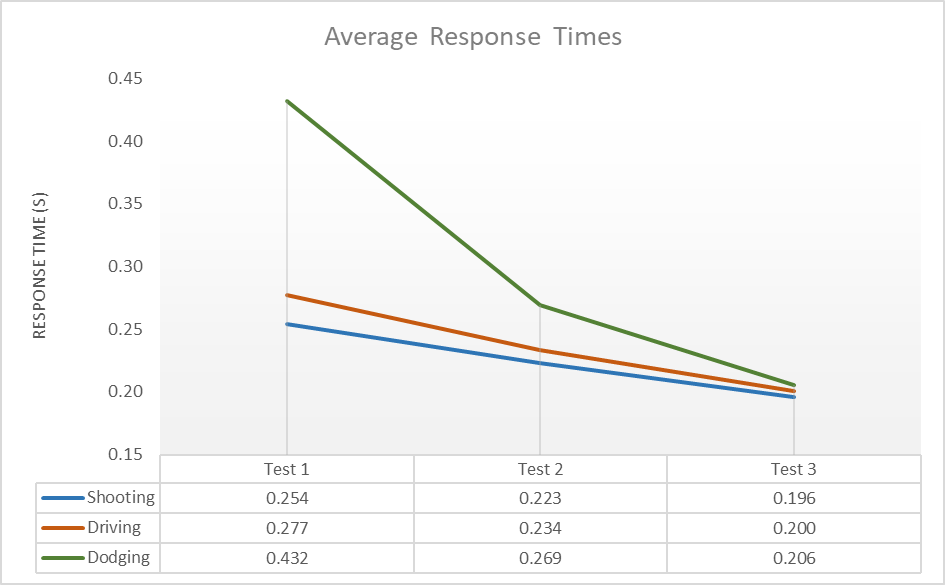
Figure 12 breaks down the core information gathered from the three testing phases. In all three games, there is a clear reduction of average response times over the three testing sessions. There is also a drastic increase in session time in both the driving and dodging games, which shows a correlation between these two factors. This is reaffirmed by the fact that the session time of the shooting game decreases as the response time decreases, which means that the reaction times must be faster throughout the whole game session in order to finish it earlier than the previous tests. The fastest response also decreases over the testing phases; however, the slowest response does not show any clear correlation throughout the tests.

Figure 13: Response time graph

Figure 12: Test data averages comparison

Figure 13 clearly illustrates how the response time over the testing phases decreases in all three games, with the largest decrease being shown in the dodging game. It can also be seen that the response time for all three games seems to reach a remarkably similar point, with only one hundredth of a second between the three games’ average response times.

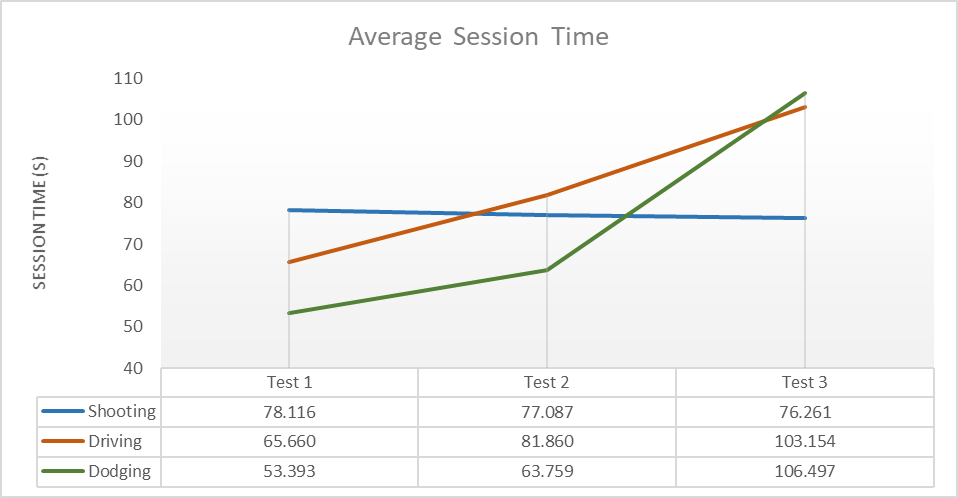
Figure 14 displays the relationship between the length of reaction times between the games. The shooting game travels in the opposite direction as the intention of the shooting game’s timer is to have the opposite effect to the driving and dodging games.

Figure 14: Average session time graph

## Age Influence on Results

## Handedness Influence on Tap Position

Figure 00: Right handedness graph

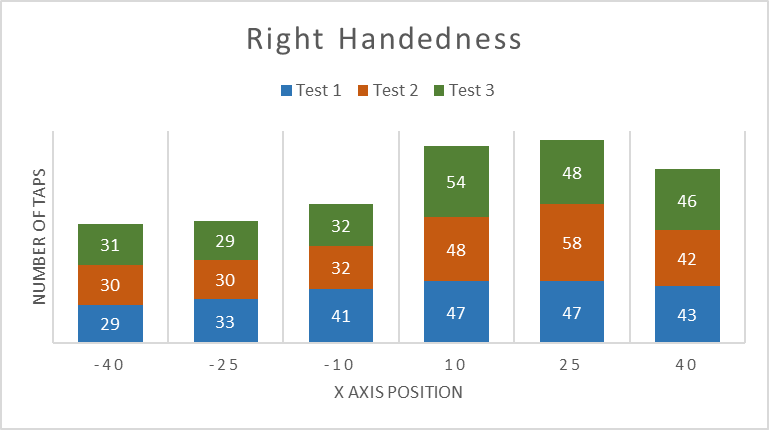
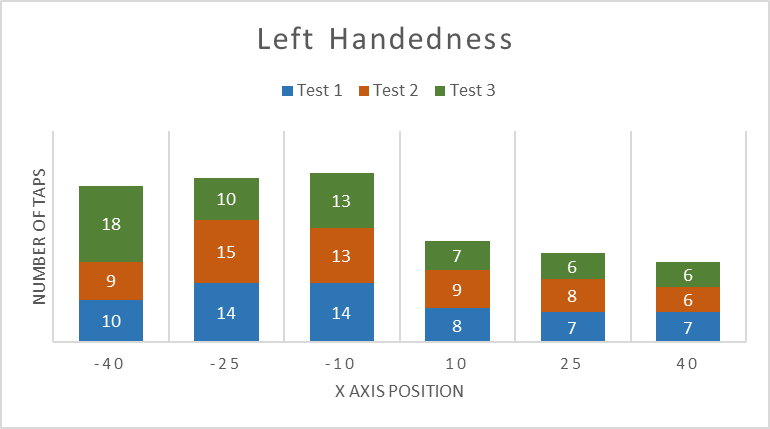


Figure 00

Figure 00: Left handedness graph

Figure 00

# Results Discussion

## Reaction Time Evaluation

## Age Influence Evaluation

## Handedness Evaluation

# Conclusion

## Application Evaluation

## Project Analysis

# References

Dye, M., Green, C. and Bavelier, D. 2009. Increasing speed of processing with action video games. *Current directions in psychological science*, 18(6), pp. 321-326.

Whitham, R. 2015. *10 multiplayer android games where you share the same phone [online].* Greenbot. Available from: <https://www.greenbot.com/article/2876235/10-android-games-you-and-a-friend-can-play-on-the-same-phone.html#slide1> [Accessed 1 November 2018].

Ng, A. and Chan, A. 2012. Finger Response Times to Visual, Auditory and Tactile Modality Stimuli. *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 2.

Statista, 2018. *Share of smartphone users in the United Kingdom (UK) from 2012 to 2016, by gender [online].* Google. Available from: <https://www.statista.com/statistics/732591/united-kingdom-smartphone-users-by-gender/> [Accessed 1 November 2018]

Statista, 2018. ***Share of smartphone users in the United Kingdom (UK) from 2012 to 2016, by age [online].* Google. Available from:** <https://www.statista.com/statistics/732535/united-kingdom-smartphone-users-by-age/> **[Accessed 1 November 2018]**

Kenney, 2018. *Kenney Bundle [online].* Itch Corp. Available from: <https://itch.io/s/6789/kenney-bundle> [Accessed 1 November 2018]

Ng, A. and Chan, A. 2012. Finger response times to visual, auditory and tactile modality stimuli. *Proceedings of the international multiconference of engineers and computer scientists*, 2.

Castel, A. D., Pratt, J., & Drummond, E., 2005. The effects of action video game experience on the time course of inhibition of return and the efficiency of visual search. *Acta psychologica*, 119(2), 217–230.

Orosy-Fildes, C., & Allan, R. W., 1989. Psychology of computer use: XII. videogame play: human reaction time to visual stimuli. *Perceptual and motor skills,* 69(1), 243–247.

Statista, 2018. ***Share of smartphone users in the United Kingdom (UK) from 2012 to 2016, by age [online].* Google. Available from:** https://www.statista.com/statistics/732535/united-kingdom-smartphone-users-by-age/ **[Accessed 1 November 2018]**

Colzato, L. S., van den Wildenberg, W. P. M., Zmigrod, S., & Hommel, B., 2012. Action video gaming and cognitive control: playing first person shooter games is associated with improvement in working memory but not action inhibition. *Psychological research*, 77(2), 234–239.

Greenfield, P. M., 1984. *Mind and media: the effects of television, video games and computers.* Cambridge: Harvard University Press.

Green, C. S., Bavelier, D., 2004. The cognitive neuroscience of video games. *Digital media: transformations in human communication.*

Greenfield, P. M., DeWinstanley, P., Kilpatrick, H., & Kaye, D., 1994. Action video games and informal education: effects on strategies for dividing visual attention. *Journal of applied developmental psychology*, 15(1), 105–123.

References to Format

<https://www.frontiersin.org/articles/10.3389/fpsyg.2011.00226/full>

<https://www.sciencedirect.com/science/article/pii/S0960982212001303>

<https://www.samsung.com/uk/smartphones/galaxy-s9/>

<https://www.visionlearning.com/en/library/Process-of-Science/49/Controlling-Variables/163>

# Appendices