

# Active learning tools

CS39440 Major Project Report

*Author:* Morgan Powell (mop14@aber.ac.uk)

*Supervisor:* Dr David Hunter (dah56@aber.ac.uk)

9<sup>th</sup> May 2020

Version 1.0 (Release)

This report is submitted as partial fulfilment of a BSc degree in  
Computer Science (Includes Foundation Year) (G40F)

Department of Computer Science  
Aberystwyth University  
Aberystwyth  
Ceredigion  
SY23 3DB  
Wales, UK

## **Declaration of originality**

I confirm that:

- This submission is my own work, except where clearly indicated.
- I understand that there are severe penalties for Unacceptable Academic Practice, which can lead to loss of marks or even the withholding of a degree.
- I have read the regulations on Unacceptable Academic Practice from the University's Academic Registry (AR) and the relevant sections of the current Student Handbook of the Department of Computer Science.
- In submitting this work, I understand and agree to abide by the University's regulations governing these issues.

Name Morgan Powell

Date 09/05/2020

## **Consent to share this work**

By including my name below, I hereby agree to this project's report and technical work being made available to other students and academic staff of the Aberystwyth Computer Science Department.

Name Morgan Powell

Date 09/05/2020

## **Abstract**

This report describes in detail the process that was used throughout this project. The purpose of this project was to build a system that enhances engagement in the classroom. This would help students pay more attention during lessons, potentially increasing their overall grade as they would retain more information.

The system created is quiz-based, which is a simple but effective way to increase engagement in the classroom. It would allow a teacher to host a quiz during lessons, while students use their own devices (or ones provided by the establishment) to connect to the quiz and answer the questions. The system is web-based as it provides the best accessibility, working on many devices.

Background research for this area is plentiful, allowing for a vast insight. This research would often encourage active learning to greatly improve students' learning, by increasing engagement and motivation in the classroom.

## Contents

<b>FIGURES .....</b>	<b>7</b>
<b>TABLES .....</b>	<b>7</b>
<b>1. BACKGROUND, ANALYSIS &amp; PROCESS .....</b>	<b>8</b>
<b>1.1. Introduction .....</b>	<b>8</b>
<b>1.2. Background .....</b>	<b>8</b>
1.2.1. Problem Overview .....	8
1.2.2. Education .....	8
1.2.3. Motivation .....	8
1.2.4. Gamification .....	9
1.2.5. Similar tools .....	10
<b>1.3. Analysis .....</b>	<b>10</b>
1.3.1. Data Storage .....	11
1.3.2. Security .....	11
1.3.3. Objectives .....	11
<b>1.4. Process .....</b>	<b>11</b>
1.4.1. Methodology .....	11
<b>2. DESIGN .....</b>	<b>14</b>
<b>2.1. Overall Design .....</b>	<b>14</b>
<b>2.2. Programming Languages .....</b>	<b>14</b>
<b>2.3. Libraries .....</b>	<b>14</b>
<b>3. TESTING .....</b>	<b>16</b>
<b>3.1. Manual Testing .....</b>	<b>16</b>
<b>3.2. User Testing .....</b>	<b>16</b>
<b>4. IMPLEMENTATION .....</b>	<b>17</b>
<b>4.1. General Approach .....</b>	<b>17</b>
<b>4.2. Iteration 1 - Foundations .....</b>	<b>17</b>
4.2.1. Use Case Diagram .....	18
4.2.2. Navigation Flow Diagram .....	19
4.2.3. Activity Diagram .....	20
4.2.4. Database .....	20
4.2.5. Entity Relationship Diagram .....	21
4.2.6. Feature List .....	21
<b>4.3. Iteration 2 – Answer Types .....</b>	<b>23</b>
4.3.1. Libraries .....	23
4.3.2. Answer Types .....	23

4.3.3.	Tests .....	25
<b>4.4.</b>	<b>Iteration 3 - Create an account.....</b>	<b>26</b>
4.4.1.	Design .....	26
4.4.2.	Implementation .....	26
4.4.3.	Tests .....	26
<b>4.5.</b>	<b>Iteration 4- Login .....</b>	<b>27</b>
4.5.1.	Design .....	27
4.5.2.	Implementation .....	27
4.5.3.	Tests .....	27
<b>4.6.</b>	<b>Iteration 5 - Creating a Quiz.....</b>	<b>27</b>
4.6.1.	Design .....	27
4.6.2.	Implementation .....	28
4.6.3.	Tests .....	28
<b>4.7.</b>	<b>Iteration 6 - View quizzes.....</b>	<b>29</b>
4.7.1.	Design .....	29
4.7.2.	Implementation .....	29
4.7.3.	Tests .....	29
<b>4.8.</b>	<b>Iteration 7 - Delete a quiz.....</b>	<b>30</b>
4.8.1.	Design .....	30
4.8.2.	Implementation .....	30
4.8.3.	Tests .....	30
<b>4.9.</b>	<b>Iteration 8 - Host a quiz .....</b>	<b>30</b>
4.9.1.	Design .....	30
4.9.2.	Implementation .....	31
4.9.3.	Tests .....	32
<b>4.10.</b>	<b>Iteration 9 - Connecting to a quiz.....</b>	<b>32</b>
4.10.1.	Design .....	32
4.10.2.	Implementation .....	32
4.10.3.	Tests .....	35
<b>4.11.</b>	<b>Iteration 10 - Point system .....</b>	<b>36</b>
4.11.1.	Design .....	36
4.11.2.	Implementation .....	36
4.11.3.	Tests .....	36
<b>4.12.</b>	<b>Iteration 11 - Leaderboard .....</b>	<b>37</b>
<b>5.</b>	<b>CRITICAL EVALUATION.....</b>	<b>38</b>
<b>5.1.</b>	<b>Methodology .....</b>	<b>38</b>
<b>5.2.</b>	<b>Technologies .....</b>	<b>38</b>
<b>5.3.</b>	<b>Requirements .....</b>	<b>38</b>
<b>5.4.</b>	<b>Design .....</b>	<b>39</b>
<b>5.5.</b>	<b>Implementation .....</b>	<b>39</b>
<b>5.6.</b>	<b>Testing .....</b>	<b>39</b>

<b>5.7. Future Work.....</b>	<b>39</b>
<b>5.8. Starting again .....</b>	<b>39</b>
<b>5.9. Conclusion.....</b>	<b>40</b>
<b>6. ANNOTATED BIBLIOGRAPHY .....</b>	<b>41</b>
<b>7. APPENDICES.....</b>	<b>46</b>
<b>A. Third-Party Code and Libraries .....</b>	<b>46</b>
<b>B. Ethics Submission .....</b>	<b>47</b>
<b>C. Code Samples .....</b>	<b>49</b>
<b>Answer Types .....</b>	<b>49</b>
Answer Type 1 .....	50
Answer Type 2 .....	53
Answer Type 3 .....	57
<b>Login Check.....</b>	<b>60</b>
<b>Creating a Quiz.....</b>	<b>61</b>
<b>Hosting a Quiz .....</b>	<b>63</b>
<b>Interacting with Quiz.....</b>	<b>65</b>

## Figures

Figure 1 Use Case Diagram .....	18
Figure 2 Navigation Flow Diagram .....	19
Figure 3 Activity Diagram .....	20
Figure 4 Entity Relationship Diagram .....	21
Figure 5 Communication Diagram - Quiz Interaction .....	34

## Tables

Table 1 Feature List .....	22
Table 2 Iteration 2 - Test Table .....	25
Table 3 Iteration 3 - Test Table .....	26
Table 4 Iteration 4 - Test Table .....	27
Table 5 Iteration 5 - Test Table .....	28
Table 6 Iteration 6 - Test Table .....	29
Table 7 Iteration 7 - Test Table .....	30
Table 8 Iteration 8 - Test Table .....	32
Table 9 Iteration 9 - Test Table .....	35
Table 10 Iteration 10 - Test Table.....	36

# 1. Background, Analysis & Process

## 1.1. Introduction

This project involves creating a quiz-based system, which will help students pay more attention in lesson by increasing their engagement. Hopefully, this enhancement will help improve student's grades by helping them learn and remember the teaching material.

## 1.2. Background

The idea behind this project is about active learning and how it can help enhance student's learning by motivating them to participate more. Here is the research about the subject and all its relevant components.

### 1.2.1. Problem Overview

One of the problems teachers face today is students not paying attention during lessons. This problem severely influences learning and can lead to a reduction in grades. This generation of students is the first to have grown up using technology, which has become a major part of their daily lives [1]. One popular technology is computers and its use of video games. Research shows that young people spend up to 10,000 hours playing video games by the time they are 21 [2]. While this demonstrates that young people can concentrate for long periods, research suggests the activity would need to be voluntary [2]. Whereas in the classroom, students are forced to learn from a set curriculum thus potentially harming their level of concentration.

### 1.2.2. Education

Education in the current day uses methods of teaching which focus on passive learning. This centers on remembering and trying to understand the materials which are taught via lecturing. A method like this can be an ineffective way for students to learn, as according to the learning pyramid model people only remember around 5% of the information they receive while listening to a lecture [3]. On the other hand, active learning allows for higher thinking, encouraging skills like applying, analysing and evaluating in order to have a deeper understanding and remembering more information [3]. Thus, making active learning a better teaching method as students are more likely to learn the material due to an increased engagement during each lesson.

People on average can only pay attention for around 10 – 20 minutes [4], this makes lecturing an ineffective way of teaching as less than half of the material would get taught. Technology can grasp a person's attention for longer periods, this may be because technology is highly stimulating, visually appealing and is not associated with learning but more of a daily norm. Additionally, this generation has grown up with technology and are used to having instant access to information. From a young age, they are taught using the action and reward method used in games. This is vastly different from the lessons taught in classrooms, where students find lessons boring as it does not compare to games in terms of holding their attention [5].

### 1.2.3. Motivation

Motivation plays a key role when attempting to learn. If a person is not motivated to learn then they will not pay attention. Motivation is defined as the 'psychological



processes that are responsible for initiating and continuing goal-directed behaviours' [6]. Motivation can be split into two different types, intrinsic and extrinsic. Intrinsic motivation can be defined as the "wish or tendency to execute an action for its own sake" [7], meaning that the person must want to perform the task as it fits with what they believe in [8]. In contrast, extrinsic motivation is when a person is motivated to do something due to the potential consequences, which can be both positive and negative. This type of motivation focuses on external factors, that are not in our control but motivate people to participate in order to receive the desired reward or to avoid a potential punishment [9].

### ***Motivation in education***

Those that are internally motivated (have intrinsic motivation) take part in classroom activities as they find the experience enjoyable [10]. They tend to be highly motivated as they enjoy learning without expecting to receive any benefits from doing so [11]. Research shows that intrinsic motivation correlates with academic achievement [12], indicating that if students are motivated to learn on their own terms then they will engage in lessons, which are shown in their grades.

On the other hand, those with extrinsic motivation can also be motivated in the classroom although they are motivated by different things. These students are influenced by external factors such as praise, grades and achievements [13]. Extrinsic motivation is especially helpful at the initial stage of learning where students have little interest in a subject. Though this type of motivation is short-lived, therefore it is important for teachers to be able to encourage the transformation to intrinsic motivation, so students are not purely relying on rewards for motivation to learn and instead are doing it because they are interested in the material.

It can be concluded that both types of motivations are useful for education, with extrinsic motivation aiding with the initial engagement and intrinsic motivation helping maintain engagement and interest in the topic [14].

### ***Motivation in computer games***

Unlike the real world, which tends to be boring, computer games can be designed to aid learning where 'action is triggered by rewards, fun, and competition' [5]. This rewarding nature of the games (receiving rewards for completing goals or tasks) externally motivates players. Though many players engage in computer games because they enjoy the activity, showing they are internally motivated. The players are so motivated to achieve the goals and receive the rewards that they become completely immersed and only the game's goal seems to matter to them [15]. Computer games are very good at holding the attention of the player, with some spending hours at a time playing [2]. Often flow occurs, which is when a player is 'completely absorbed or engaged in an activity' [16]. This state of flow increases learning as those playing are focused on learning to achieve the goal of the game [15].

One element of computer games that increases learning is competition. Friendly competition, due to the social aspect of games, increases learning through boosting motivation [17]. Competition is a good way of engaging players as it gives them another goal to win. Those that are intrinsically motivated find the activity fun and those who are extrinsically motivated participate as they may receive a reward or wanting to place near the top of the leader board. However, the use of a leader board that displays all the player's scores can be demoralising [18].

### **1.2.4. Gamification**

Gamification is the application of 'game elements to a non-game context' [19]. Other researchers expand on this to include that gamification aims to use the motivation and

competitive nature from computer games to aid with learning to help engage people and make activities more enjoyable [5].

### ***Gamification in education***

The inclusion of gamification in education has promising results, with students eager to learn through computer games [5]. The competitive aspect of gamification would encourage engagement and increases the attention paid, especially if they are expected to have retained information from the activity. There are many benefits to education by including gamification, in particular, the supportive and fun environment that games create seems to suit the generation that has grown up with technology [20]. Other benefits include students feeling more relaxed and focusing less on failure as games can be repeated and they can see visible progress of their learning [21]. Games may also help students discover that they have an internal motivation to learn since games can be manipulated to encourage the 'intrinsic motivation drive' [5]. Gamification can help hold student's attention through the inclusion of games and activities [22]. This active method of learning shall help students retain more information as they are applying their knowledge to a game and increase their participation in the class [23].

#### **1.2.5. Similar tools**

There are similar tools available online that give a fun way of hosting quizzes. These are many aimed at those who want to play for fun and not necessarily for learning, although some are being marketed towards use in the classroom. They can be available for free or potentially requiring a subscription fee. One tool that is very popular worldwide via the internet is Kahoot! [24]. This system is quiz-based, with a points system that allows players to gain more points by answering questions quickly and correctly. There are many features available such as custom or anonymised names. Overall, this system is fun, colourful and engaging. Another similar tool that is available is Quizizz [25], though Kahoot! has been found to be a more effective tool [26].

### **1.3. Analysis**

This system would allow for interaction between students and lecturers. A web-based approach would be a great starting point for this kind of system. This would allow for easier interaction between host (lecturers) and users (students), along with the benefit of students using their own device. Meaning no specialist hardware or software would be needed. Whereas developing this system into a specific application would require students to download software for it to work which also add the problem of device compatibility, different devices and operating systems may have restrictions on what can be installed or multiple versions of the application may need to be created to accommodate a plethora of devices. This would also mean that the establishment may need to provide devices to students, which can be costly and disruptive during the session, potentially wasting a lot of time too.

A web-based approach would provide cross-compatibility as most web browsers follow a standard which requires them to support many different features and elements used within web development languages. Meaning, following those standards a webpage should work on numerous browsers with minimal issues. Most people in the modern-day carry an internet-connected device, mainly smartphones, tablets or laptops, which will cut down on the number of devices an establishment may need to supply for this activity. Also, cross-compatibility would accommodate for most students.

### **1.3.1. Data Storage**

Being web-based, there is a multitude of ways to store data. One example considered is the file format JSON. This format allows data to be stored in a compact, readable way with a very small file size, perfect for transferring data over the web. There would be two easy ways to store these files, either within a file store which essentially stores individual files on the server. Alternatively, a document storage system could be used in which you can store documents that can be easily accessed, searched and updated.

Alternatively, a database could be used. All data required would be stored in a singular place but split up into different tables. A database setup would make it easy to store data whilst avoiding unnecessary duplicates. Specific data can be found easily with well-prepared queries and are especially great in situations where not all data is needed to be retrieved. Being familiar with utilising a web-based application that has a database backend gives this approach an advantage over other methods.

### **1.3.2. Security**

Security is a significant issue when creating a web-based application. There are many ways a user can maliciously attack a website. The two main security issues include cross-site scripting and SQL injections. Cross-site scripting involves injecting JavaScript into the page via inputs field available on the page. The input field's data gets processed and if it is not sanitised correctly, code can be injected into the system and executed.

SQL injections could be devastating for the users who have an account with the web application. The attack involves injecting SQL code into an input field which gets processed and then executed. This could potentially compromise the user's data within a database or even delete all data within.

To minimise the risk of malicious code entering the system, all input fields must be sanitised and checked to ensure that the user's data is secure.

### **1.3.3. Objectives**

The objective of this project is to build a system that teachers can use and incorporate into their lessons to help engage students and aid their learning. The format of the system will be based on quizzes. Where teachers can base questions on recent teaching material and challenge students to achieve as many points as possible by answering questions correctly. Hopefully, this system will be fun, challenging and visually appealing to students and achieve the goal of grasping their attention for a longer period and should positively influence their overall grade.

## **1.4. Process**

### **1.4.1. Methodology**

Various methodologies were researched before the commencement of this project. Mainly these are agile approaches such as XP (extreme programming), Scrum, FDD (Feature Driven Development) and Kanban. A plan-driven approach, for example, the waterfall model was also considered. Each has their own advantages and disadvantages, whilst most revolving around a team of developers meaning some adaptations would need to be made. A plan-driven approach bodes most familiarity and would make for a great choice. Whereas an agile approach would be more difficult due to a lack of experience however it also provides the opportunity to gain an

understanding of the development process which would be beneficial when starting work within the computing industry.

For this project, various practices from each of the agile methodology would apply to this project. But many would have to be adapted or ignored, mainly because there is not a development team but merely a single developer.

Hence, after researching each methodology it was decided that a mix of practices from the methodologies FDD and XP would be followed. Not all practices from both would apply, so combining them would make for a clear process which can be followed.

Feature Driven Development's focus is to build a system by developing individual features that will work together in the final system. Features are worked on during iterations that range from a few days up to two weeks. During each iteration, a feature is designed, implemented and tested. This method of building the system by small parts ensures that features have a dedicated timeslot in which there is minimal risk of anything going wrong [27].

### ***Feature-Driven Development***

FDD is made up of five steps that guide the development process. These practices will help guide the project. These include firstly, developing an overall model, this process involves creating an understanding between team members by creating a model of the system. Multiple models would be proposed by team members before a final model is decided on. [27]

The second step involves building a feature list which is essentially the requirements for a system. This provides a clear list of things that would need to be considered. The third step comprises of the concept: planning by feature. This step fixates on organisation, more specifically deciding on the order in which features will be developed, which helps guide each iteration of the project. Step four, designing by feature. This step makes use of teams which are assigned a feature by a chief programmer. Each team has an expert who will oversee analysing and designing. The final step, step 5 is about building by feature. Once a design is made, work can begin on the feature. When the code is complete and tested then it can be implemented into the system [27].

Steps one, two and three will be followed during the initial iterations. This time would be used to follow each step precisely and truly understand the features that would be needed in the system [27]. Diagrams will be created to show the overall architecture of the system. A feature list will also be created to determine what features are required to make the system work, each feature will have a number, name, description and an acceptance test.

Steps 4 and 5 will have to be adapted slightly as there are not multiple teams available for each feature. To counteract this, step 4 will have the design created by the sole developer leading on to step 5 where the feature is built, tested and implemented into the system. Everything will be done by a single developer, making this process somewhat less efficient. But this is to be expected with all agile methodologies [27].

### ***Extreme Programming***

XP is the other methodology whose practices will be followed [28]. Naturally, some practices will not apply as they require more than one developer. These are the practices that will help shape the project. Merciless Refactoring is a practice that involves improving code without losing functionality.

A simple design is the idea of getting what you need and nothing more. Keeping the design simple also cuts down on the development time and cost. This will be a key practice when deciding on the features to be implemented into the system.

Sustainable development is all about having a steady workflow which will help avoid burnout for the developers. Do follow this practice, proper planning of iterations would need to be completed and have a set number of hours each workday where the feature will be worked on and avoid having to work overtime where possible.

The main reasoning to favouring FDD practices over XP is the way requirements are presented and how iterations are feature-focused along with smaller sections of code to develop. FDD features a clearly defined list of features of which would need to be designed, built, implemented and tested. This approach benefits a developer due to its clarity and preciseness. Whereas XP features stories, which are a very useful way to describe desired features for a system. But requires a team and a customer who would help create these stories along with extra details in terms of the type of user accompanied by the features business value.

## 2. Design

### 2.1. Overall Design

The overall design of this systems aims to be exciting, fun and visually appealing. This would help draw student's attention and create a fun environment for them to interact with. Similar tools in which inspiration can be drawn from are found in section 1.2.5. Further, features need to be easy to use, to not deter the student's motivation while learning to use the system.

### 2.2. Programming Languages

HTML, CSS and JavaScript are the building blocks of every web page, granted there are tools available to help create the pages. HTML will provide the essentials for building webpages while CSS provides style to each of the elements within each page. JavaScript allows for extra functionality when needed, whilst also providing a way to communicate with a data source and update the current webpage with little disruption. Having more experience in these languages will prove to be beneficial for future employers.

PHP is a server-side language that allows data to be processed on a server and the response being sent and possibly displayed on the webpage. With this language, communicating with a database can be done quite easily. Data can be inserted, updated, deleted or simply searched for.

Continuing with the theme of using a database, SQL would be needed. This language is used within databases and is the basis of how programmers communicate with it. This could be creating tables, inserting, updating, selecting and removing data that is no longer needed.

### 2.3. Libraries

As this project aims to add a fun and interactive element to the quizzes, simply using pure JavaScript simply will not suffice. The reasoning being that the language does not have many features that aid with animations, especially as they become more complicated. Creating a game like scenario can be done too but like animations, this would be easier and better to do with the use of a library.

There are many libraries available for animations, which is a slight issue as they offer very similar features. Some research is needed to determine which library would be most suitable for this project. One main feature to consider is if the library uses the HTML canvas. Making use of the canvas element is extremely important as most users who will connect to the quiz will be using a mobile device. This is because there is a vast amount of support for the canvas element on mobile devices / browsers.

Upon researching various libraries, this one was the one that stood out the most. Mainly due to its object orientated design and various event methods that allow animations to be created very easily. Its name is Paper.js [29]. There are many tutorials available which make it easy to learn, also the documentation is clear and informative. This library will be used to create animations which in turn will allow fun and interesting interactions with quizzes. These libraries were researched during an introductory iteration, further details can be found there.

Most of the design is produced during each of the iterations in accordance with the features, along with overall designs being completed during the initial iterations.

## 3. Testing

### 3.1. Manual Testing

For this project, manual testing was executed to ensure that at the end of each iteration the feature being implemented works as expected and that it meets the expectations of the acceptance test. A variety of different tests were done to not only ensure the acceptance tests were completed as mentioned previously but to ensure that there were no unexpected outcomes which could potentially cause an error in the system. A test table will be created for each iteration.

### 3.2. User Testing

Unfortunately, user testing could not be completed, but there is an idea as to how it would have been done. This system is intended to accommodate a class of various amounts of people. So, a group of users would be gathered in a classroom / lecture hall and join a quiz hosted on the main computer in the room. This will test how the system performs under the stress of multiple devices connected simultaneously. Upon quiz completion, the users would be given a survey on what they think of the animations, overall opinion on the system and suggestions for improvement.

The survey would gather data to help shape the final steps in development and improve the system. The data would include questions about each animation type, the points and leaderboard system, the display of each question and general ease of use along with other comments or suggestions. This would then be used, and a consensus will be found for each question, which would indicate what areas of the system work well and what areas could be improved.

Once the data has been analysed, any issues arisen can be worked on and improved in time for the completion of development. User testing would have helped build on the system and make it better for future users. Also, user testing could then be used in the future to see how well new features perform as they are implemented.

Naturally, these tests would have been completed nearer the end of development when many features had been implemented. This was unable to be completed due to the coronavirus pandemic, which stopped any opportunity of planning and completing user testing.



## **4. Implementation**

### **4.1. General Approach**

Following the iterative of the chosen methodologies, each feature will be worked on during a specific iteration. With some initial iterations being used for research, spike work and general preparation to better prepare for building each feature. For example, creating template webpages that would be used as a basis for features and researching JavaScript libraries that could be used for animations. Once this first iteration is complete, work can begin on each of the features.

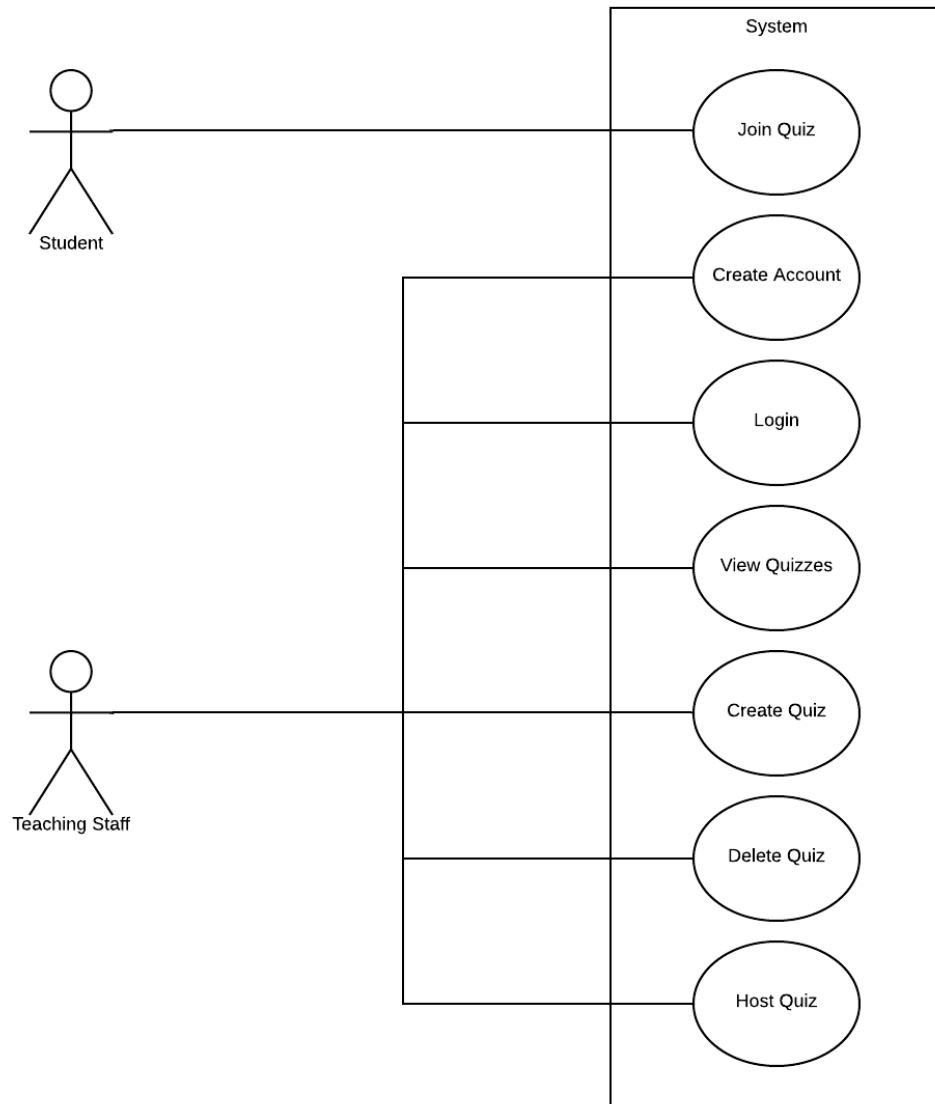
When implementing each feature, it is imperative to focus on user experience, particularly for students. The general design of this system expresses the use of exciting and fun interactions with the system. This would then entice students to participate leading to further engagement. This is kept in mind throughout development.

### **4.2. Iteration 1 - Foundations**

For this iteration, some basic foundations for the project will need to be created. Using multiple diagrams and the list of features, a design for the webpages can be created. This will include all the pages required for each specific feature, plus extra general pages that will provide extra information for the users. Each page would be created as a template along with minimal information which would be replaced when building the feature that resides on that specific page.

### 4.2.1. Use Case Diagram

This use case diagram shows clearly how each type of user will interact with the system and what specific features they would have access to. As you can see, the accessible features for each type of user are vastly different, but it also makes sense when understanding the project and what each user would need while using this system.



*Figure 1 Use Case Diagram*

#### 4.2.2. Navigation Flow Diagram

This flow diagram indicated the webpages in which the user would be able to access. The pages outside of the “Logged in” container is available to all users who are not logged in. On the other hand, if a user has logged in, then the options inside the “Logged in” container become available. As you can see, the “Host Quiz” and “Join Quiz” are connected. This is intended to show that users who are not logged in can interact with a quiz that is hosted by a user that is logged in. This diagram was created by analysing the use case diagram (Figure 1) and depicting what webpages would be needed for this system.

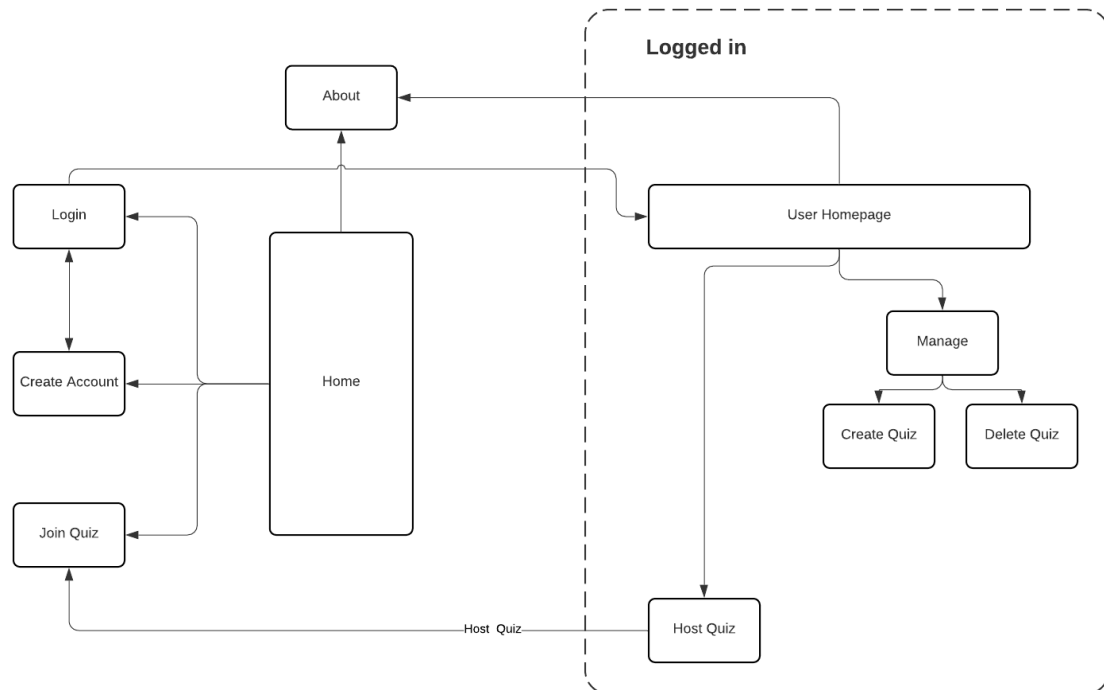


Figure 2 Navigation Flow Diagram

### 4.2.3. Activity Diagram

This activity diagram illustrates how the user's interaction with the quiz will work. It shows each step that occurs while a user is connected to a quiz. As shown, the system works as a loop until the quiz completes, at which the points total is displayed.

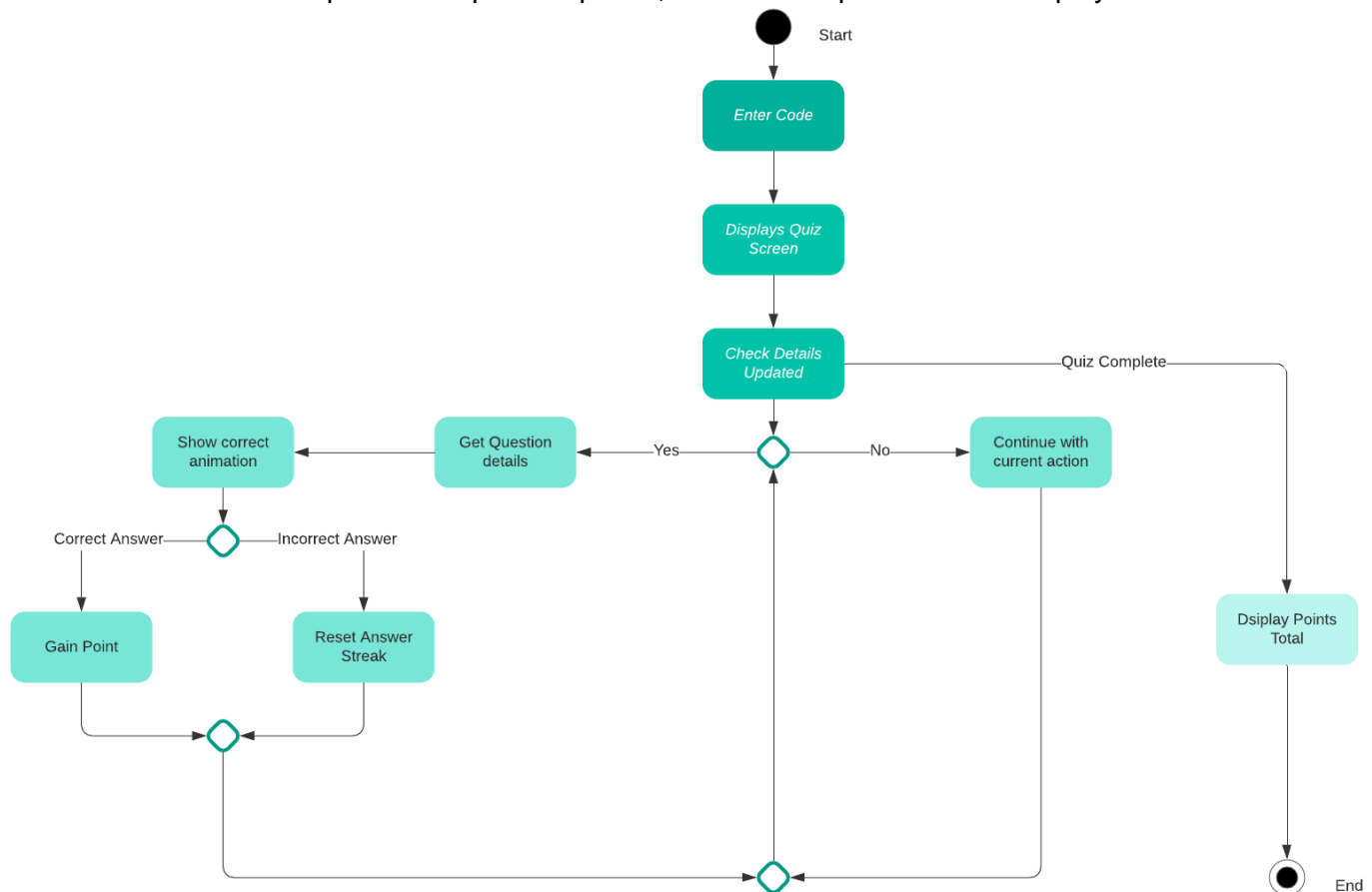


Figure 3 Activity Diagram

### 4.2.4. Database

A database is essential for this project as there is data to be stored for both the user and their quizzes. User's details were kept to a minimum as not many details are required for this system especially considering the size and potential user base of the system further this decision is partly due to security, if there were to be a breach then no personal details will be compromised. Although if such a system were to be used across the country, then further details would be needed such as an email address which would be used for account verification. Quiz details would include an ID for identification, number of questions, question number, question content and answer type for each question. Answers would need a question number to correspond to, the number / letter of the answer for example 1, 2, 3 or A, B, C and of course the answer content along with if the answer is the correct one for that specific question.

#### 4.2.5. Entity Relationship Diagram

This entity relationship diagram shows how the database will store data, whilst indicating the relationship between each table via the crow's foot notation. Many of the databases relationships being one to many.

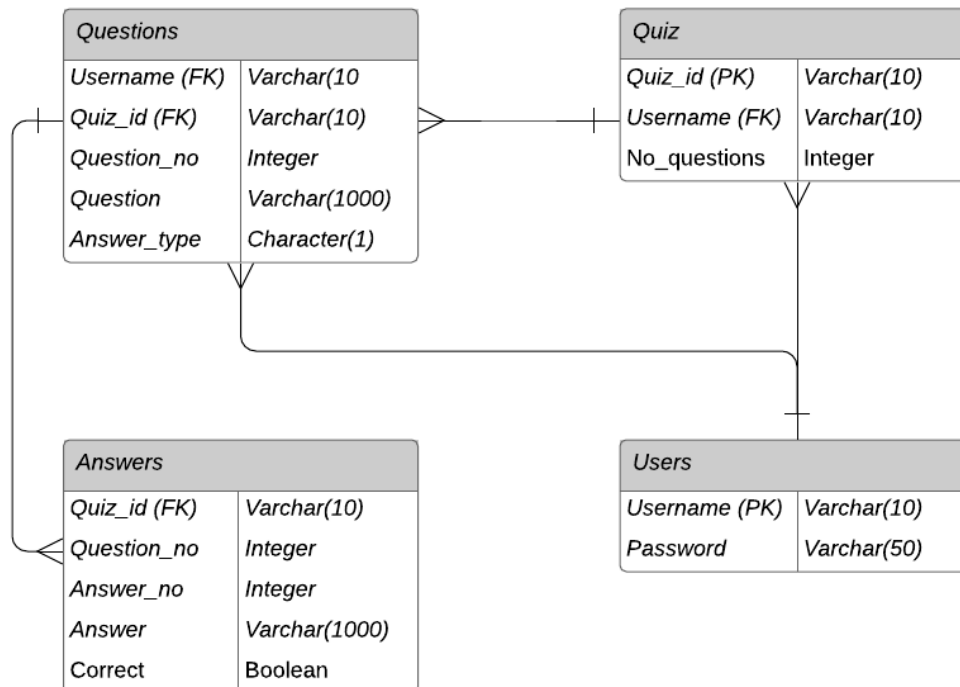


Figure 4 Entity Relationship Diagram

Using the diagrams created, basic webpages were created to match the navigational diagram (Figure 2). The database is set up and features all the tables seen in the entity relationship diagram (Figure 3) with all primary and foreign keys set correctly to ensure that data is stored and retrieved correctly.

#### 4.2.6. Feature List

To get an idea of what web pages this system and follow Feature-Driven Development's practices a feature list is created. It includes the simplest version of each feature to show clearly what is expected.

Table 1 Feature List

Feature Number	Feature Name	Feature Description	Acceptance Test
1	User Login	A user should be able to create an account and login.	A user should be able to create an account and login. Upon creation, usernames should be unique, and duplicates should not be created.
2	Create a Quiz	A user should be able to create a new quiz, which will be saved once created.	A user should be able to create a new quiz, enter questions, answers and other relevant details.
3	View Quizzes	A user should be able to see all created quizzes currently available to the user.	Created quizzes should be showed with an option to view in more detail.
4	Delete a Quiz	A user should be able to delete a quiz. The quiz should no longer show when viewing quizzes.	A quiz can be viewed, then upon clicking a delete button the quiz's details should be removed from the database and will no longer show when viewing available quizzes.
5	Host Quizzes	A user should be able to host a quiz, displaying each question and answer options.	A user would be able to host a quiz that would display each question and relevant answers. Next question would be shown after clicking a button.
6	Connect to quiz	Users should be able to connect to the quiz with the webpage updating correctly with the hosts machine	A user would be able to join a quiz and questions would update in real-time with the host.
7	Quiz Interaction	Users should be able to interact with the quiz with answer correctness displaying correctly.	Users should be able to select an answer, which would then display correct or incorrect depending on the question.
8	Animated Answer Types	Each question will have multiple answer types to choose from which add variety and fun to each quiz.	There should be distinct answer animations that can be viewed and interacted with by joining users. Two animations are a minimum to pass.
9	Points System	Users will gain points for correct answers during the quiz.	The number of points a user has will increase with each correct answer. The total amount of points will be displayed at the end of the quiz.
10	Leaderboard	Users will be able to see a leaderboard of the top three users displayed on the host's machine. The usernames would be anonymous of course.	After each question, a leaderboard displayed on the host machine will be updated to show the top three student's scores.

### 4.3. Iteration 2 – Answer Types

#### 4.3.1. Libraries

Libraries were discussed earlier and are a key component for animation creation for this project. There are many libraries available for JavaScript especially in terms of animations, so various considerations had to be made. A library to use in such a narrow time scale would be easy to learn and straightforward to use but must have great functionality for animating objects along with interacting with them. One key element which would add a great amount of device compatibility is canvas support, meaning that the library should use the HTML canvas element for its animations. With these principles in mind, many libraries were researched but inevitably had to be narrowed down to one.

Each library has its own positives and would be a great tool for this project. After carefully observing and testing each library, Paper.js was chosen [29]. This is due to its simplicity, object orientated design and event functions, specifically on frame events which trigger on each frame.

#### 4.3.2. Answer Types

Quizzes usually come with questions that have two to four answers, keeping this in mind three answer types were creatively thought of to be used. Two can be used for questions that boast four answers, while one that accommodates true or false style questions. The code for each answer type can be found in Appendix C (Answer Types).

##### ***Answer type 1***

The first answer revolves around the idea of thinking quickly. The user's screen is split into four separate sections, one for each quarter that corresponds to an individual answer. Each section shrinks at a steady pace, they keep shrinking until there is nothing left. Meaning the users must look at the question, figure out the correct answer and select it before the answer disappears entirely. This answer type rushes the user, but there is enough time to figure out the answer before time runs out.

To create this answer type, a simple approach is taken. Firstly, four rectangles of equal size (a quarter size of the canvas element) were created. These are separate objects and have their own on click event. Next, these rectangles need to shrink, to do this the built-in on-frame event is combined with a function that takes each rectangle's size and applies a scale function to them that decreases their size down very slightly. Because the on-frame event triggers every frame, then the rectangles will continually shrink until another event occurs, therefore the decrease in size is very small. Each rectangle's on click event will call a correct or incorrect function depending on the correctness of that choice. These functions remove all objects on the canvas and replaces them with a correct or incorrect message.

The tests performed on this answer type have the test IDs: AT1A, AT1B, AT1C and AT1D. The test table can be found in the tests section of this iteration.

##### ***Answer type 2***

The second answer type involves splitting the four answers into circles. These then move around the screen following a specific pattern. The original intent was to give the objects a float like visual on the screen, but this may seem not challenging enough to users, so constantly moving objects seemed like a good idea. Each object follows a predictable path, the movement is smooth but quick.

To create the second answer type, four circle objects were created, they are all relative to the canvas, so that they are not too big or small no matter the device's screen size. Next the on-frame event is used to move each circle. The movement path of each circle is determined by the position of the adjacent circle. The adjacent circle is clockwise to the original circle meaning the flow of movement is also clockwise. The on click event acts the same as it does in answer type 1.

The tests performed on this answer type have the test IDs: AT2A, AT2B, AT2C and AT2D. The test table can be found in the tests section of this iteration.

### ***Answer type 3***

Answer type number three is a simple case of true or false, meaning only two objects were needed. These objects are two squares, one being option A the other option B. These objects do not fill the whole screen, just a segment half the size the device's screen. Both objects move simultaneously in opposite directions, for example, one moves towards the top of the device whereas the other moves towards the bottom. Then both reversing directions as they reach the top of the device. Both objects stay on their side of the screen.

This answer type has two rectangles of equal size that were placed adjacent to each other in the middle of the canvas. These objects also rely on the on-frame event to move. Answer A starts by moving towards the top of the screen, once the top is reached, it moves towards the bottom of the screen and back to the top once the bottom is reached. Answer B moves in a very similar manner, except instead moving towards the bottom of the device's screen at the start of the animation. This makes both objects always move in opposite directions. Like the first and second answer types, each object can be clicked and will display a correct or incorrect message accordingly.

The tests performed on this answer type have the test IDs: AT3A and AT3B. The test table can be found in the tests section of this iteration.

Demo pages have also been created to show off each of the answer types so that user's wishing to host a quiz can see exactly which ones are available and how they function.



### 4.3.3. Tests

Some basic manual tests were done to check that the demos show correctly what happens when an answer is selected. For each answer type, option A will be correct and should be the only one that displays a "correct" screen, meaning every other option will display an "incorrect" screen. Each test ID is an abbreviated form of "Answer Type (answer type number) Option (A, B, C or D)".

*Table 2 Iteration 2 - Test Table*

Test ID	Description	Pass / Fail	Comments
AT1A	Selecting option A should display a "correct" screen.	Pass	When option A is chosen, the screen shows a "correct" message.
AT1B	Selecting option B should display a "incorrect" screen.	Pass	When option B is chosen, the screen shows a "incorrect" message.
AT1C	Selecting option C should display a "incorrect" screen.	Pass	When option C is chosen, the screen shows a "incorrect" message.
AT1D	Selecting option D should display a "incorrect" screen.	Pass	When option D is chosen, the screen shows a "incorrect" message.
AT2A	Selecting option A should display a "correct" screen.	Pass	When option A is chosen, the screen shows a "correct" message.
AT2B	Selecting option B should display a "incorrect" screen.	Pass	When option B is chosen, the screen shows a "incorrect" message.
AT2C	Selecting option C should display a "incorrect" screen.	Pass	When option C is chosen, the screen shows a "incorrect" message.
AT2D	Selecting option D should display a "incorrect" screen.	Pass	When option D is chosen, the screen shows an "incorrect" message.
AT3A	Selecting option A should display a "correct" screen.	Pass	When option A is chosen, the screen shows a "correct" message.
AT3B	Selecting option B should display a "correct" screen.	Pass	When option B is chosen, the screen shows a "incorrect" message.

## 4.4. Iteration 3 - Create an account

### 4.4.1. Design

The user will navigate to the appropriate web page where they will be greeted by a form. This form simply includes username and password fields. Once details are entered, the submit button is clicked and the data is sent via post to the account creation page. This page will check the username against the IDs stored in the database. If an ID exists, then the new account will not be created, whereas if there is no record of the ID, a new account can be created. The data to be stored is taken from the post request, added to an insert statement that will be communicated to the database. The data will also be escaped, and the password encrypted for security. Once this is complete, the account will have been stored in the database and the user is informed that their account is ready to be used.

### 4.4.2. Implementation

Both username and password are sent via the post, this is for security reasons. The data is received and processed on the completion page in which the PHP script will connect to the database using PDO class that is built into PHP and requests all available data matching the posted username. This is to check if any data is retrieved then the username already exists. If this happens then the account will not be created. If no data is retrieved then the username is available, a simple SQL insert is performed using the posted data and the account is successfully created. During the execution of the query, encryption is applied to the password, this conceals the password from the system entirely so that only the user knows what their password is. A message will appear informing the user that they may now login using those details.

### 4.4.3. Tests

Manual tests were done on the account creation feature. These will mainly test the checks that are being completed. These checks are encouraging but there were some issues. The IDs are abbreviated as "Create Account Test (test number)".

*Table 3 Iteration 3 - Test Table*

Test ID	Description	Input	Pass / Fail	Comments
CAT1	Entering a username and password and clicking submit should allow for account creation.	Username: Test Password: Test	Fail	SQL error. It appears there is an issue with the insert statement.
CAT2	Entering a username and password and clicking submit should now allow for account creation.	Username: Test Password: Test	Pass	The account was created successfully.
CAT3	Entering a username and password that already exists then clicking submit should not allow for account creation.	Username: Test Password: Test	Pass	The account was not created, and an error message is displayed.

## 4.5. Iteration 4- Login

### 4.5.1. Design

The login page will provide a link to the account creation page, along with a form that the user will use to login. The form will have two input fields, one for username and the other being password. This data will be posted to a script for processing. The username and password are then checked against existing data where both username and password must match, the password is encrypted of course to match what is stored. If there is a match the user will have a logged-in status. Otherwise they will not be logged in.

### 4.5.2. Implementation

The login functionality is implemented similarly to creating an account. The user enters their username and password into the correct form fields. That data is then sent via the post method to a PHP script which takes both the username and password adding them to a query. This query requests data from the server that matches both the username and password. If either is incorrect then no data will be retrieved, prompting the web page to deny the user a successful login. However, if data is retrieved, the user is given a logged-in status by setting their username within the session variable. The user will now have access to the user homepage where they can access the various quiz related features that the system offers. Code snippets for checking username and password can be found in Appendix C (Login Check).

### 4.5.3. Tests

Manual tests were done on the login feature. These will mainly test the checks that are being completed. The IDs are abbreviated as "Login Account Test (test number)".

*Table 4 Iteration 4 - Test Table*

Test ID	Description	Input	Pass / Fail	Comments
LAT1	Entering a correct username and password and clicking submit should allow the user to login.	Username: Test Password: Test	Pass	Successfully directed to the user's homepage.
LAT2	Entering an incorrect username and password and clicking submit should not allow the user to login.	Username: Test1 Password: Test1	Pass	Unsuccessful login attempt with an error displayed.

## 4.6. Iteration 5 - Creating a Quiz

### 4.6.1. Design

Quiz creation is a fundamental part of this system, a quiz needs to be created before it can be hosted. This process should be as simple as possible. Starting out the user navigates to the quiz creation page. There they will find a form requiring a quiz ID and the number of questions for this quiz. The data will be sent via post, leading to the ID and number of questions being stored in the database providing the ID does not already exist. Thus, leading to the main part of quiz creation. A page showing input fields for all question details including the question itself, answer type for that question,

the answers and which answer is correct. Submitting this form will send the data to be processed and stored. The user will see a “successful” message and can proceed with the next question. The question number will be determined by how many questions are in the quiz. For example, if a quiz has three questions, then the first question the user inputs details for will be give the number “1” in the database. This is updated incrementally until the number of questions is reached. Once all questions have been submitted and stored the user is presented with a “completion” message. The quiz should now be able to be viewed and hosted.

#### 4.6.2. Implementation

Upon starting the creation process, the user enters a preferred ID for the quiz along with the number of questions that will be in the quiz. The quiz ID is checked against data within the database as each ID needs to be unique and therefore duplicate IDs cannot be used. The next step involves entering each question’s details with each one being inserted into the database upon completion. Question number will be sequential to easily store and match each question to its answers. The user will not have any interaction with this. Once a quiz has been created successfully, it can be viewed, deleted and hosted. Code snippets can be found in Appendix C (Create a Quiz).

#### 4.6.3. Tests

These tests check that no duplicates quizzes can be created and that all fields must be entered before moving on to the next question.

*Table 5 Iteration 5 - Test Table*

Test ID	Description	Input	Pass / Fail	Comments
CQT1	Entering a quiz ID and number of questions should allow for quiz creation to begin.	Quiz ID: Quiz No. Of Questions: 2	Pass	Quiz creation was initiated, and the question form was presented.
CQT2	Entering an existing quiz ID and any number of questions should not allow for quiz creation to begin.	Quiz ID: Quiz No. Of Questions: 2	Pass	Quiz creation was not initiated, and an error message appears.
CQT3	Entering all questions details should progress the quiz creation with the question being stored.	All question details including answers. All fields entered as “Test”	Pass	The question was stored, and the creation process was continued.
CQT4	Entering no details into the form’s fields should prompt the user and not continue with quiz creation.	None.	Pass	All fields are required, and the quiz creation process does not continue.

## 4.7. Iteration 6 - View quizzes

### 4.7.1. Design

While logged in, the user must be able to see what quizzes they have available. To do this, the user navigates to the manage page which shows all available quizzes. This feature would be a simpler one to implement as all it requires is the username of the logged-in user. Once navigated to the manage quizzes page, all created quizzes should be available to view. This initial screen will show the quiz ID and number of questions for each quiz. Selecting a quiz for further viewing will direct the user to a page in which all quiz details are shown. Every question and its answers will match correctly. This can be done by searching the database for all quiz IDs linking to the user's username, then using the quiz ID too search for the quizzes questions.

### 4.7.2. Implementation

To show these quizzes, a simple SQL query is executed which requests all quizzes that match the user's ID. As a reminder, the user's ID is stored within the session variable. Each quiz that is retrieved is stored within its own div element that includes basic details featuring the quiz's ID and the number of questions. A button is included which directs the user to a page displaying more information regarding the quiz's questions and answers. The quiz ID will be sent as part of a form (which the button is linked to). The ID is then used to form part of a query which requests all the quiz's details that match the ID. Each question is identifiable along with its answers and other details. An option for deleting the quiz is also available here.

### 4.7.3. Tests

Tests were used for this feature, one that checks that there are no quizzes available, one for one quiz available and one for multiple quizzes available along with more quiz details. The IDs were abbreviated forms of View Quiz Test (test number).

*Table 6 Iteration 6 - Test Table*

Test ID	Description	Pass / Fail	Comments
VQT1	Navigating to the manage page should display no quizzes as none should be created.	Pass	No quizzes are shown.
VQT2	Navigating to the manage page should display one quiz that was created for this test.	Pass	One quiz is displayed
VQT3	The quiz is available for further details to be viewed. Selecting it should display all question's details.	Pass	All question's details are displayed
VQT4	Navigating to the manage page should display two quizzes that were created for this test.	Pass	Both quizzes are displayed

## 4.8. Iteration 7 - Delete a quiz

### 4.8.1. Design

While viewing a quiz's details, an option for deletion is made available. Upon clicking this button, the quizzes details will be sent to a page (along with the user) which searches the database for that quizzes details and deletes it. This should remove all data regarding the quiz, and it should no longer appear on the manage or hosting page.

### 4.8.2. Implementation

A click of the delete button will send the quiz ID to a PHP script which will take this ID and incorporate it into an SQL delete statement along with the user's ID. This SQL statement searches for data in multiple tables that matches both quiz ID and username's and removes them from the database. A confirmation message is then shown stating that the quiz had been successfully removed. The quiz will no longer show up when managing quizzes, and a new one can be created with the same ID.

### 4.8.3. Tests

Tests were done to ensure that quizzes get removed from the database. Tests are abbreviated forms of Delete Quiz Test (test number).

*Table 7 Iteration 7 - Test Table*

Test ID	Description	Pass / Fail	Comments
DQT1	Navigating to the manage page and selecting a quiz should show further details. When clicking the delete button, the quiz should be removed.	Fail	SQL error, there appears to be an issue with the delete statement.
DQT2	Navigating to the manage page and selecting a quiz should show further details. When clicking the delete button, the quiz should be removed.	Pass	The quiz is deleted from the database and no longer shows on the manage page

## 4.9. Iteration 8 - Host a quiz

### 4.9.1. Design

Hosting a quiz is an extremely important feature in the system. The goal is to have a page that displays each question and relevant answers. The user views the available quizzes and selects the one they wish to host. The user is then prompted, asking are they sure they want to host this quiz before allowing them to host. Once hosting, the user is presented with a screen that shows each question in the quiz along with its answers. These questions are displayed individually, with the option for the user to show the next question. With each question change, the connected user's devices will update too.

After each question change, data would need to be sent to the connected user's devices. This will be done by saving the necessary details to a temporary file on the server which can be retrieved by devices that know the file name (this is where the code is used).

#### **4.9.2. Implementation**

Once the user navigates to the hosting page and select a quiz to host, they are directed to the quiz hosting page and the quiz will be live. Some of the quiz's details are sent via post for the questions to be displayed. These details form part of two SQL statements which selects data from multiple tables matching these details. Initially, there is a waiting screen where the host can wait for users to connect before showing the first question. A code is also randomly generated and displayed at the bottom of the screen; this is shown throughout the quiz. This will be important when communicating with connected users.

The current question number is stored as a session variable along with the quiz's number of questions which was retrieved from the previous page. The current question is set to zero and increments by one with every question displayed. Clicking start displays the first question on the screen. The question's details are searched via a query that uses the session's stored variables and displays the question along with its answers. In the background, answer's correctness and answer type is sent to a file which stores the data, ready to send to the connected users in order to alert their devices as to which answer type to display along with which answer is correct and which ones are not. Code snippets for hosting can be found in Appendix C (Hosting a Quiz).

### 4.9.3. Tests

Tests were completed to check that details are saving correctly to the file and each question is displayed correctly. Test IDs are abbreviated to Host Quiz Test (test number).

*Table 8 Iteration 8 - Test Table*

Test ID	Description	Pass / Fail	Comments
HQT1	Selecting a quiz for hosting should display a start screen with a code.	Pass	The start screen is shown correctly
HQT2	Clicking start while hosting a quiz should display question details.	Fail	SQL error, there is an issue with the complex SQL statement
HQT3	Clicking start while hosting a quiz should display question details.	Pass	Question details are displayed correctly.
HQT4	A file should be created when displaying a question, the file should have a file name relating to the code and contain answer details.	Pass	The file is created, and answer details are stored correctly.
HQT5	The file created should be deleted when the quiz is finished.	Pass	The file is removed and is no longer available.

## 4.10. Iteration 9 - Connecting to a quiz

### 4.10.1. Design

Connecting to a quiz involves typing in a code that matches that of the host. This will be the link between the host and connected users. The code is the name of the file which question details are retrieved from. Upon entering the code and confirming it, the user is directed to a waiting page that updates when the quiz starts. Questions are displayed and answers are shown using the correct answer type.

### 4.10.2. Implementation

#### *Updating the page*

Updating the users' page for each question change is done using AJAX. Two specific functions were used. One which checks the hash of the file and updates a hidden element within the page. Once this function is complete, a secondary one will retrieve the question details providing the file has been updated i.e. the hash code has changed. The hash is checked at one-second intervals. Code snippets for quiz interaction can be found in Appendix C (Quiz Interaction).



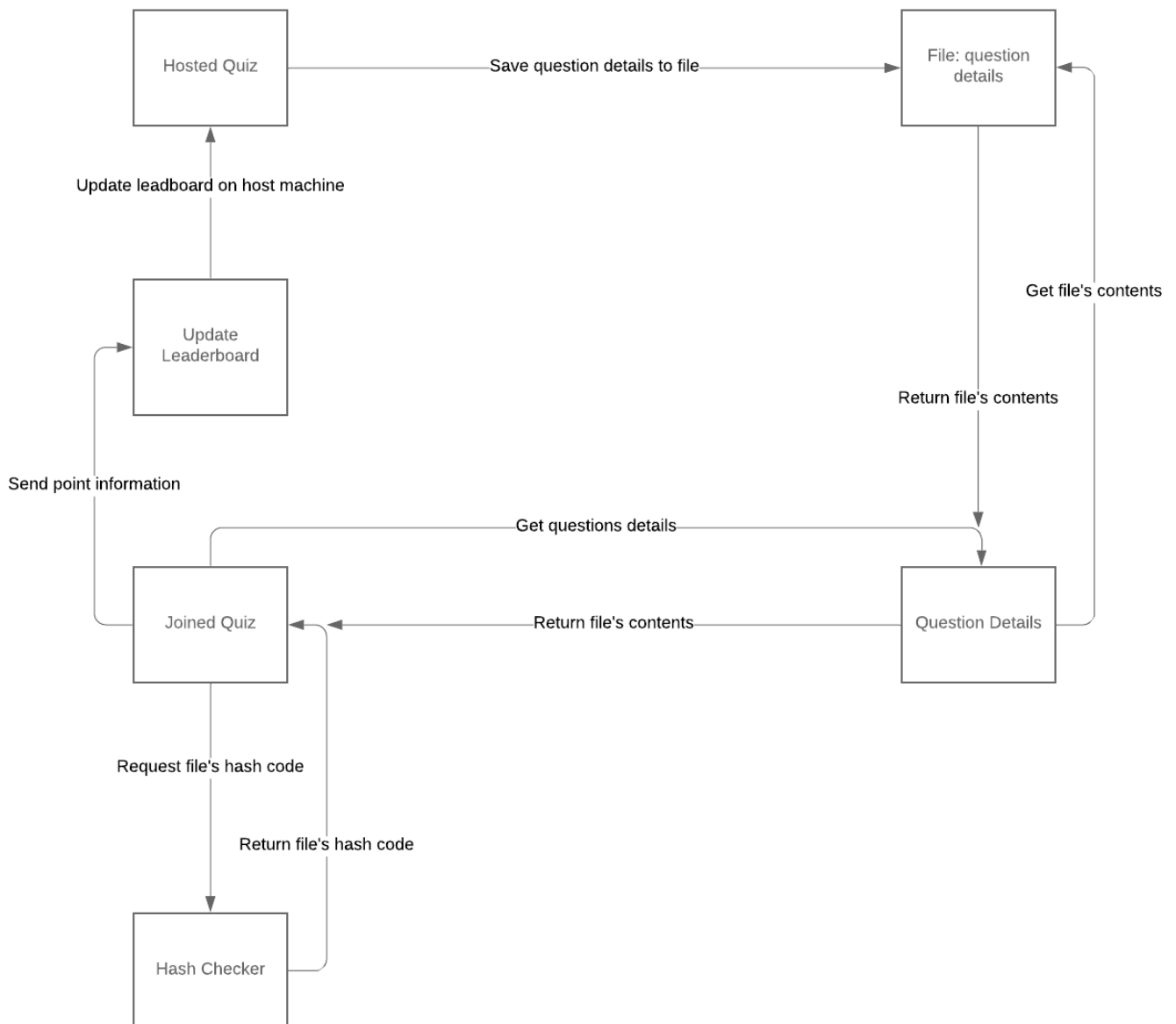
***Correct / Incorrect answer selection***

This feature uses the animations created in an earlier iteration. Each object in the animation corresponds to an answer. The correctness of each answer is determined by the details provided by the file created by the host. This file is retrieved via an AJAX function mentioned earlier. Tapping or clicking a correct answer will display a congratulatory screen and will award a point whilst also increasing the user's answer streak. Points and answer streaks are stored as variables. Each set to zero upon connecting, then increase with each correct answer. Answer streaks would award extra points if a streak greater than two is reached. Whereas incorrect answers would display wrong answer screen that would also reset the user's answer streak.

### **Communication Diagram Showing Quiz Interaction**

This diagram shows the basics of how the quiz interaction will be handled and how data will be passed between the host and other users. This is a simple plan, once again following the XP practice. This also makes it simpler and therefore easier to create this system.

The plan is to store the hosted quizzes' question's details in a file. Which when updated, the joined user's webpage will retrieve the file's details and update the page. Upon completion of each question, each user's webpage will send their scores and usernames to a script that will in turn form and update the leaderboard on the host's machine.



*Figure 5 Communication Diagram - Quiz Interaction*

### 4.10.3. Tests

Tests were performed to check that a user's device can connect correctly and update with the host. The test IDs are abbreviated forms of Connect Quiz Test (test number).

*Table 9 Iteration 9 - Test Table*

Test ID	Description	Pass / Fail	Comments
CQT1	Entering the host's session code should direct the user to a waiting screen if a quiz is yet to start.	Pass	The waiting screen is shown correctly.
CQT2	Hash hidden element should be updated when the first question is shown.	Pass	Hash gets updated correctly.
CQT3	Answer's hidden field should get updated with file's details.	Pass	Answer's hidden fields are updated correctly.
CQT4	Answers should be displayed and animated depending on answer type stored in a hidden field.	Pass	The correct answer type is shown, and animations are showing.
CQT5	Selecting the correct option should result in a "correct" screen being displayed.	Pass	Correct screen is shown after the correct answer is selected.
CQT6	Hash and answer hidden fields should be updated with host displaying next question.	Pass	All hidden fields are updated correctly.
CQT7	Correct answer type should be displayed, and animations are shown.	Fail	Nothing displayed, a slight error in code leading to a null pointer.
CQT8	Correct answer type should be displayed, and animations are shown after the second question is displayed.	Fail	Incorrect answer type displayed, but correct animation being shown. Animation being updated before answer type.
CQT9	Correct answer type should be displayed, and animations are shown after second question displayed.	Pass	Correct answer type and animation are displayed.
CQT10	Selecting the incorrect option should result in "incorrect" screen being displayed.	Pass	The correct screen is shown after an incorrect answer is selected.
CQT11	After a quiz has been completed, a finished screen should be displayed.	Pass	The finished screen displays informing the user that the quiz has ended.

## 4.11. Iteration 10 - Point system

### 4.11.1. Design

The points system will act as a reward for each user and shows them how well they performed in the quiz. At the start of the quiz each user's points total is set to zero and they can gain points by answering questions correctly. One point is gained per correct answer, with an extra point awarded if the user is on an answer streak. An answer streak is obtained by answering consecutive questions correctly, with each correct answer increasing the streak by one. If the streak is greater than one, then an extra point is awarded per correct answer. This streak is reset to zero if a question is answered incorrectly. The answer streak is used to motivate the users to answer questions correctly and more consistently.

### 4.11.2. Implementation

As mentioned, Points and answer streaks were set to zero initially and increment when a correct answer is chosen. This is simply just adding one to both variable on the background of displaying a congratulatory screen. Although the answer streak is set to zero if an incorrect answer is chosen. The users point total is displayed at the end of the quiz.

### 4.11.3. Tests

Tests were completed to check that points are being increased along with answer streaks as each correct answer is selected. As well as answer streaks being set to zero on incorrect answer selections. The test IDs are abbreviated forms of the Points System Test (test number).

*Table 10 Iteration 10 - Test Table*

Test ID	Description	Pass / Fail	Comments
PST1	Selecting the correct answer while connected to a quiz should increase points value by one. (Single question quiz was created for this test).	Pass	The points total is shown as 1 at the end of the quiz.
PST2	Selecting 2 correct answers in a row should add an extra point as the answer streak is above one. (Two question quiz was created for this test).	Pass	The points total is shown as 3, 2 from correct answers and an extra point from the answer streak.
PST3	Answer streak set to zero if the wrong answer is selected. (Two question quiz, first answered correctly, second incorrectly).	Pass	The points total is shown as 1, the extra point was not awarded for the answer streak which was reset to zero.

#### **4.12. Iteration 11 - Leaderboard**

A leaderboard is implemented to entice the user to do well and have their name on show on the host's machine, clear for everyone to see. Although these names will be anonymised so that only individual users know what their pseudonym is. This is to not make users feel demotivated if their name is not on the leaderboard [18], but also not allow a form of un-sportsmanship comments towards those who excel within each quiz. Additionally, only the top three scores shall be displayed on the leaderboard. This shall ensure players are still motivated to participate but also reduce the fear of failing as only the top three scores shall be displayed for the rest of the class to see.

The idea / design of the functionality behind the leaderboard is to accumulate each user's current points total and store them in a file (like saving quiz details). This file is then sorted to show which users have gained the most points, with the top three being retrieved by the host machine. These usernames and points totals are then displayed on the host's machine for all to see.

At least this is what the feature's functionality would be. Unfortunately, this feature was unable to be implemented due to current circumstances within the world. But given a little more time, this feature would have completed the project.

## 5. Critical Evaluation

### 5.1. Methodology

The methodology followed during this project combined practices from two agile methodologies. It took the main practices from Feature-Driven Development and the relevant practices from Extreme Programming. Some of these would not apply or would need to be altered. FDD was mainly chosen as the process is very appealing, along with a defined list of features compared to stories which can have a lot of extra information. Clear, concise information is preferred. Overall, this process worked well, with issues coming from personal errors rather than a problem with the practices themselves. Following the practices allows for efficient and safe programming, meaning a minimal risk of bugs. The iteration process is very useful, it provides a clear indication of what sections of the system will be worked on and how much time should be spent building and implementing.

The list of features shown in Table 1 was arranged in the order that they will be worked, with iterations ranging from a few days up to two weeks. Some perceived shorter iterations needed to be extended due to bugs and software issues, but overall, most features could be completed within the time frame given.

### 5.2. Technologies

The choice of the development environment used is Visual Studio Code [30]. This environment can be used on various operating systems, which allows features to be worked on regardless of computers available. There are many extensions available, especially helpful for web applications where multiple languages are used.

These languages include HTML, CSS, JavaScript, AJAX, PHP and SQL. These languages are all familiar due to previous experience from past modules. Although, the use of JavaScript libraries was unfamiliar. Libraries have been used before in other languages, just not in JavaScript. It was a new experience, that felt like learning a new language as some libraries' programming styles differ from the original language. Although, through building, testing and reading documentation, figuring out how to use the library proved to not be that difficult. However, there were some moments where functions would work unexpectedly and cause confusion in the animation.

### 5.3. Requirements

The objective of this project was to create a system that allowed teachers to create a quiz that students could join and interact with. This would help increase engagement in the classroom leading to an increase in grades achieved. This objective was met, although there are some improvements that could be made. Based on the research completed in section 1.1 this system should vastly increase engagement and learning.

The specific requirements of this project are shown in the form of a feature list. These are broken down into their simplest form and as they are concise, and it is easy to keep track of which features have been implemented. As discussed in iteration 11, the leaderboard feature was not implemented. This feature would have been a nice addition to the system; however, the system still functions well without it. This did not get implemented due to the special circumstance in the world today with the coronavirus pandemic, but also issues that occurred with various bugs leading to

iterations being extended slightly. This left very little time to implement the leaderboard effectively.

## **5.4. Design**

The design of the system works well, but many other technologies could have been used which could have improved the system and may have helped during development. Most features were designed and built without much further research, though some features would have benefitted from a different approach or at least more research.

The development environment chosen was very helpful and provided support for all Programming languages used. Although the debugging feature was unavailable, so it was done using the browsers built-in debugger. The choice of using an SQL database over JSON files and a document storage database was a good one, mainly because specific data can be requested when needed rather than requesting a whole file, then decoding and sorting the data until the required items are found. On the other hand, the JSON file approach would have been interesting.

## **5.5. Implementation**

The implementation of each feature works even though some had unexpected bugs which were frustrating at times. Each one was implemented well and abides by the original design without many issues or changes. Apart from one feature which was not implemented however the overall functionality of the system remains unaffected. The quiz creation feature proved to be quite challenging as there are multiple steps depending on how many questions there are. But in general, most features were implemented without much issue.

## **5.6. Testing**

Testing was completed at the end of each feature's iteration; these were manual tests. Each test checked that the feature works as intended with test results being stored in a test table. Some tests failed but most passed which was relieving. User testing was going to be performed but that was disregarded due to the current pandemic situation. This kind of testing would have been very beneficial had it been performed.

## **5.7. Future Work**

If there were more time or the project could be continued there are many features that could be implemented. Starting with the leaderboard. Extra answer types with options to choose multiple answers would improve the experience greatly. Extra options for hosting users would also be added, such as graphs that shows details for each question, and transition screens between questions.

## **5.8. Starting again**

Given the chance to start again, the process would have been different. Time management would be greatly emphasized, this was an issue which leads to some features not being completed as quickly as they could, leading to extra hours being worked. Other options for transferring data would have been researched and attempted to find a better solution. The chosen library would have included graph / chart functionality, and this would also be reflected in the requirements. One last

change would be researching the background of this project. As seen, a vast amount of research was completed before building the system, this may have influenced the amount of time available to implement all features. More time would be prioritized for building rather than researching.

## **5.9. Conclusion**

Overall, this project went well with some obvious issues that will be avoided in the future. One issue, revolving time management has already been discussed as something that would be focused on heavily if the project were to be completed again. Another issue that continually came up is the development environments somewhat lack support for PHP. There are some extensions that provide support, but these are basic while more advanced support would require extra installs. This meant that the main way to debug any errors were manually or using a browser's debugger, which often displayed a general error code. These issues would be considered in the future.

A lot of experience was gained involving agile methodologies, the iteration process is a very effective way of working on a project and is one that will be used in the future. Further experience was gained for various web development languages which will be beneficial when searching for future employment.



## 6. Annotated Bibliography

- [1] M. Prensky, "Digital natives, digital immigrants," *On the horizon*, vol. 9, no. 5, 2001.

An excellent summary of the difference between digital natives and digital immigrants that have arisen due to the increased use of technology among the youngest generation. The paper talks about how the lack of technological knowledge from teachers can influence the learning of their students.

- [2] M. Prensky, "Digital game-based learning," *Computers in Entertainment*, vol. 1, no. 1, p. 21, 2003.

The paper talks about how in the past computer games were typically played by teenage boys and were associated with aggressive behaviours. Nowadays children of all ages spend a significant time gaming. It introduces the idea that computer games could be utilised to help with children's learning to help them engage more in classroom activities.

- [3] A. Sousa, *How the brain learns: a classroom teacher's guide*, Thousand Oaks: Corwin Press, 2001.

This book covers the concept of how students learn. The chapter on memory, retention and learning was of interest as it mentioned the implications of teaching on how students learn, along with describing the learning pyramid. The book explained that there are different levels of learning and the more that you do with the information you receive, the more you retain from it. This is useful to support the idea of active learning.

- [4] J. Glifkins, "What Is 'Active Learning' and Why Is It Important?," *E-International Relations*, 8 October 2015. [Online]. Available: <https://www.e-ir.info/2015/10/08/what-is-active-learning-and-why-is-it-important/>. [Accessed 28 April 2020].

This blog post provides an introduction into active learning, Specifically, focusing on the importance of adding it to lessons to aid with children's learning since children have a short attention span so are unable to focus for the full duration of lessons.

- [5] E. Zarzycka-Piskorz, "Kahoot it or not? Can games be motivating in learning grammar?," *Teaching English with Technology*, vol. 16, no. 3, pp. 17-36, 2016.

This paper uses the tool Kahoot! to address if games have a positive influence on students' motivation to learn using grammar tests. It talks about how the real world differs to games and how this can be addressed using gamification. It also goes on to discuss how the concept of gamification has a major influence on how students engage in lessons due to the change in their motivations to want to learn.

- [6] M. Sailer, J. Hense, J. Mandl and M. Klevers, "Psychological perspectives on motivation through gamification," *Interaction Design and Architecture Journal*, no. 19, pp. 28-37, 2014.

This journal article discusses how games keep people playing them. Games are great at holding the players attention for long periods of time by offering them rewards and new levels. The specific game elements that are included to hold a player's attention also increase their motivation to want to play the game. This causes them to become invested in the game, thus spending many hours playing at a time.

- [7] E. Deci and R. Ryan, *Intrinsic motivation and self-determination in human behavior*, New York: Plenum Press, 1985.

This book focuses on motivation that comes from within. It details the self-determination theory to explain intrinsic motivation. The focus of the theory is that people are intrinsically motivated due to three psychology needs, which are competence (ability to do the task that will help them achieve their goals), autonomy (how much control they have) and relatedness (want to be attached).

- [8] Leadership-central.com, "Types of Motivation," [Online]. Available: <https://www.leadership-central.com/types-of-motivation.html>. [Accessed 28 April 2020].

This webpage defines both intrinsic motivation and extrinsic motivation and explains the difference between each motivation. It also includes some examples of situations where each are present.

- [9] R. Benabou and J. Tirole, "Intrinsic and extrinsic motivation," *The review of economic studies*, vol. 70, no. 3, pp. 488-520, 2003.

The paper looks at the impact of a range of different factors on motivation, including rewards, punishments and empowerment. It says that some people are motivated by receiving a reward or avoiding the threat of a punishment, though those that were intrinsically motivated performed worse as their motivation has decreased. The paper concludes that rewards and punishment are only effective for short periods of time.

- [10] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary Educational Psychology*, vol. 25, p. 54-67, 2000.

This paper offers an updated perspective from their previous paper published in 1985 on motivation and the self-determination theory. The paper discusses the self-determination theory throughout history, expanding on each type of motivation based on more current research and theories.

- [11] E. A. Skinner and M. J. Belmont, "Motivation in the classroom: Reciprocal effects of teacher behavior and student engagement across the school year," *Journal of educational psychology*, vol. 85, no. 4, pp. 571-581, 1993.

While this paper is dated, it shows research conducted across an entire school year. It investigates the importance of the teacher in a student's motivation. The paper shows how disengagement in lessons combined with a negative response from teachers can cause students' motivation to significantly decrease.

- [12] D. Pérez-López and M. Contero, "Delivering Educational Multimedia Contents through an Augmented Reality Application: A Case Study on Its Impact on Knowledge Acquisition and Retention," *Turkish Online Journal of Educational Technology*, vol. 12, no. 4, 2013.

This paper used augmented reality to aid in students' learning. Their results conclude that augmented reality was an efficient tool for helping students retain more information. They also demonstrate the impact of a student's motivation for learning when using augmented reality in comparison to traditional teaching methods.

- [13] H. Tohidi and M. M. Jabbari, "The effects of motivation in education," *Procedia-Social and Behavioral Sciences*, vol. 31, pp. 820-824, 2012.

This paper provides a good summary of previous research on intrinsic and extrinsic motivation. The paper also discusses how motivation can be controlled, focusing on how motivation can be influenced in an education setting.

- [14] T. Li and R. Lynch, "Relationship between motivation for learning and academic achievement among basic and advanced level students studying Chinese as a foreign language in years 3 to 6 at Ascot International School in Bangkok, Thailand," *Scholar: Human Sciences*, vol. 8, no. 1, 2016.

This study focused on motivational goal orientation. The study found a correlation between intrinsic motivation and academic grades, but not extrinsic motivation, when specifically looking into motivational goal orientation. The paper also discusses the levels of intrinsic motivation and extrinsic motivation comparing students in normal classes with more advanced classes

- [15] J. Webster, L. K. Trevino and L. Ryan, "The dimensionality and correlates of flow in human-computer interactions," *Computers in human behavior*, vol. 9, no. 4, pp. 411-426, 1993.

This paper reviews their previous paper on how computer games encourage playfulness in people. It then goes on to explain how the concept of flow influences this idea of playfulness when playing games as players become obsessed with the goals within the game.

- [16] M. Csikszentmihalyi, *Flow: The psychology of optimal experience*, vol. 41, New York: Harper Perennial, 1991.

This book, while dated, introduces the concept of flow. It starts with the history, looking into how people find happiness and the optimal experiences that people aim for. The book then goes on to talk about how when attention is focused onto a goal, it can influence the optimal experience of a person.

- [17] B. C. DiMenichi and E. Tricomi, "The power of competition: Effects of social motivation on attention, sustained physical effort, and learning," *Frontiers in psychology*, vol. 6, no. 1282, 2015.

This study conducted an experiment to investigate the influence of competition on learning and attention. The results showed that reaction times improved when they were competing against another person.

- [18] İ. Ü. Yapıcı and F. Karakoyun, "Gamification in biology teaching: A sample of kahoot application," *Turkish Online Journal of Qualitative Inquiry*, vol. 8, no. 4, pp. 396-414, 2017.

This paper shows how the online quiz system Kahoot! can benefit education by looking at its influence on biology lessons. The paper concluded that due to the use of Kahoot! that students were more motivated and willing to participate. Though it does point out that students receiving a bad score found it demoralising to have their score displayed to the entire class.

- [19] S. Deterding, M. Sicart, L. Nacke, K. O'Hara and D. Dixon, "Gamification: using game-design elements in non-gaming contexts," *CHI'11 extended abstracts on human factors in computing systems*, pp. 2425-2428, 2011.

This paper provides an exhaustive look into gamification. While it remains a new term, the paper investigates the history of how gamification came about and discusses its potential based on the current research

- [20] A. Bosworth, "Keas: Developing a Successful Game-Based Employee Wellness Program," *Games for Health Journal*, vol. 1, no. 3, pp. 189-191, 2012.

This paper aims to help people improve their health, reducing the number of sick days they take and therefore reducing the costs to companies. It states that motivation is a key factor in achieving this. The paper found that by using computer games people were more motivated to improve their health since computer games made the experience more fun.

- [21] Learning Theories, "Gamification In Education," 26 January 2016. [Online]. Available: <https://www.learning-theories.com/gamification-in-education.html>. [Accessed 28 April 2020].

This website provides a clear discussion into the benefits of gamification in education. It starts by introducing gamification and what specific game elements are included. The website further goes on to discuss how these game elements could be utilised to benefit students learning.

- [22] K. Kiili, "Digital game-based learning: Towards an experiential gaming model," *The Internet and higher education*, vol. 8, no. 1, pp. 13-24, 2005.

This paper addresses that while previous literature has introduced game aspects into education, that no model has been developed to aid the process. The paper combines the attention holding nature from flow theory with experiential learning theory to create a model that can be used to aid with students learning using games.

- [23] H. Bicen and S. Kocakoyun, "Perceptions of students for gamification approach: Kahoot as a case study," *International Journal of Emerging Technologies in Learning*, vol. 13, no. 2, pp. 72-93, 2018.

This paper uses Kahoot! as a case study to investigate how students perceive using games to learn. It found students were more likely to participate when these games were included in the learning experience.

- [24] Kahoot!, "Kahoot!," [Online]. Available: [kahoot.com](https://kahoot.com). [Accessed 10 February 2020].

This is the online quiz system Kahoot!. Popular around the world, it provides a way of hosting quizzes that multiple users can join. This system has also been the centre of attention for various papers discussed.

- [25] Quizizz, [Online]. Available: <https://quizizz.com/>. [Accessed 29 April 2020].

Quizizz is an educational quiz-based system that aids to help with teach for schools and places of work. This system was also discussed in one of the papers researched, being compared to Kahoot!.

- [26] D. O. Göksün and G. Gürsoy, "Comparing success and engagement in gamified learning experiences via Kahoot and Quizizz," *Computers & Education*, vol. 135, pp. 15-29, 2019.

The study explores the effectiveness of gamification in education by comparing two tools that were designed to improve learning. Results showed that Kahoot! increased engagement and academic performance. Quizizz performed lower than a control group.

- [27] L. Karam, "FDD; Its processes and comparison to other agile methodologies," Apiumhub, 18 May 2017. [Online]. Available: <https://apiumhub.com/tech-blog-barcelona/feature-driven-development/>. [Accessed 21 February 2020].

This article discusses the agile methodology Feature-Driven Development. While explaining each step and general usefulness, the article also compares this methodology to others such as XP and Scrum.

- [28] N. Taylor, "Introducing extreme programming (XP)," 2020. [Online]. Available: [https://blackboard.aber.ac.uk/webapps/blackboard/content/listContent.jsp?course\\_id=\\_23768\\_1&content\\_id=\\_1502815\\_1](https://blackboard.aber.ac.uk/webapps/blackboard/content/listContent.jsp?course_id=_23768_1&content_id=_1502815_1). [Accessed 20 February 2020].

This presentation introduces the agile methodology XP and discusses the process it follows while detailing key practices.

- [29] Paper.js, "Paper.js," [Online]. Available: <http://paperjs.org/>. [Accessed 5 March 2020].

Paper.js is a JavaScript library that provides various tools for vector graphics, animations and interactions.

- [30] Visual Studio Code, "Code editing. Redefined," Visual Studio Code, [Online]. Available: <https://code.visualstudio.com/>. [Accessed 24 April 2020].

Visual Studio Code is a development environment that provides support for many programming languages.

- [31] A. I. Wang, M. Zhu and R. Saetre, "The effects of digitizing and gamifying quizzing in classrooms," in *10th European Conference on Game Based Learning*, Paisley, Scotland, 2016.

The conference paper compares quiz-based games with a response system that does not incorporate game to investigate if gamification aids in the classroom. It concluded that the use of game-based quizzes increased motivation, enjoyment and engagement in lessons although found it did not increase learning.

- [32] J. McGonigal, *Reality is broken: Why games make us better and how they can change the world*, New York: The Penguin Press, 2011.

This book addresses the key difference between games and reality; games can be manipulated. It is this factor that suggests that games can help in a variety of sectors since they can be adapted to function. For example, it talks about improving the happiness of workers in different environments.

- [33] Z. Dornyei, *Motivational Strategies in the Language Classroom*, Cambridge: Cambridge University Press, 2001.

This book provided an insight into strategies that are currently used in schools to help students maintain motivation while learning. It talks about motivation in terms of both generating motivation and maintaining motivation.

- [34] L. C. Wood and T. Reiners, "Gamification," in *Encyclopaedia of Information Science and Technology*, Hershey, Information Science, 2015, pp. 3039-3047

This chapter concentrates on gamification. The key elements of gamification are each addressed in detail including the main components, game mechanics and game design.

- [35] D. Baranovskiy, "Raphaël," [Online]. Available: <https://dmitrybaranovskiy.github.io/raphael/>. [Accessed 20 February 2020].

A JavaScript library that uses vector graphics to create objects and manipulate them. This library was looked at as apart of animation research.

- [36] Fabric.js, "Fabric.js," [Online]. Available: <http://fabricjs.com/>. [Accessed 20 February 2020].

A JavaScript library that uses vector graphics to create object, animate and apply various effects to them. This library was looked at as apart of animation research.

## 7. Appendices

### A. Third-Party Code and Libraries

Here is the JavaScript library I used in this project.

#### **Paper JS** [29]

This is a JavaScript Library that I used for creating animations and add interactivity to each quiz. It can be found via this link - <http://paperjs.org/>

Here is the License:

Paper.js is distributed under the permissive MIT License:

Copyright (c) 2011, Juerg Lehni & Jonathan Puckey  
<http://lehni.org/> & <http://jonathanpuckey.com/>  
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## B. Ethics Submission

**AU Status**

Undergraduate or PG Taught

**Your aber.ac.uk email address**

mop14@aber.ac.uk

**Full Name**

Morgan Powell

**Please enter the name of the person responsible for reviewing your assessment.**

Reyer Zwiggelaar

**Please enter the aber.ac.uk email address of the person responsible for reviewing your assessment**

rrz@aber.ac.uk

**Supervisor or Institute Director of Research Department**

cs

**Module code (Only enter if you have been asked to do so)**

CS39440

**Proposed Study Title**

Active Learning Tools

**Proposed Start Date**

27 January 2020

**Proposed Completion Date**

01 June 2020

**Are you conducting a quantitative or qualitative research project?**

Mixed Methods

**Does your research require external ethical approval under the Health Research Authority?**

No

**Does your research involve animals?**

No

**Are you completing this form for your own research?**

Yes

**Does your research involve human participants?**

Yes

**Institute**

IMPACS

**Please provide a brief summary of your project (150 word max)**

I am creating a quiz-based system which aims to help test student's knowledge of the provided learning material, whilst giving an element of fun and interactivity to each lecture.

**I can confirm that the study does not involve vulnerable participants including participants under the age of 18, those with learning/communication or associated difficulties or those that are otherwise unable to provide informed consent?**

Yes

**I can confirm that the participants will not be asked to take part in the study without their consent or knowledge at the time and participants will be fully informed of the purpose of the research (including what data will be gathered and how it shall be used during and after the study). Participants will also be given time to consider whether they wish to take part in the study and be given the right to withdraw at any given time.**

Yes

**I can confirm that there is no risk that the nature of the research topic might lead to disclosures from the participant concerning their own involvement in illegal activities or other activities that represent a risk to themselves or others (e.g. sexual activity, drug use or professional misconduct).**

Yes

**I can confirm that the study will not induce stress, anxiety, lead to humiliation or cause harm or any other negative consequences beyond the risks encountered in the participant's day-to-day lives.**

Yes

**Please include any further relevant information for this section here:**

**Where appropriate, do you have consent for the publication, reproduction or use of any unpublished material?**

Not applicable

**Will appropriate measures be put in place for the secure and confidential storage of data?**

Yes

**Does the research pose more than minimal and predictable risk to the researcher?**

Not applicable

**Will you be travelling, as a foreign national, in to any areas that the UK Foreign and Commonwealth Office advise against travel to?**

No

**Please include any further relevant information for this section here:**

**If you are to be working alone with vulnerable people or children, you may need a DBS (CRB) check. Tick to confirm that you will ensure you comply with this requirement should you identify that you require one.**

Yes

**Declaration: Please tick to confirm that you have completed this form to the best of your knowledge and that you will inform your department should the proposal significantly change.**

Yes

**Please include any further relevant information for this section here:**



## C. Code Samples

### Answer Types

Shared Functions:

Correct Answer:

```
function correctAnswer(){
  project.activeLayer.removeChildren();
  result = new Path.Rectangle({
    point: [0, 0],
    size: [canvasWidth, canvasHeight],
    strokeColor: 'black'
  });

  result.fillColor = 'green';
  Rtext = new PointText(new Point(result.position, result.position));
  Rtext.justification = 'center';
  Rtext.fillColor = 'black';
  Rtext.content = 'Correct \n You gained a Point!';
  Rtext.fontSize = canvasWidth*0.05;
}
```

Incorrect Answer:

```
function incorrectAnswer(){
  project.activeLayer.removeChildren();
  result = new Path.Rectangle({
    point: [0, 0],
    size: [canvasWidth, canvasHeight],
    strokeColor: 'black'
  });

  result.fillColor = 'red';
  Rtext = new PointText(new Point(result.position, result.position));
  Rtext.justification = 'center';
  Rtext.fillColor = 'black';
  Rtext.content = 'Incorrect';
  Rtext.fontSize = canvasWidth*0.1;
}
```

## Answer Type 1

This snippet simply calls the function to create the objects to be animated. Then in the “onFrame” function a scale function is called (this occurs every frame) that animates each object.

```
one();  
function onFrame(event){  
  scaleOne();  
}
```

Creating each answer object:

```
function one(){  
  A = new Path.Rectangle({  
    point: [0, 0],  
    size: [canvasWidth * 0.50, canvasHeight * 0.50],  
    strokeColor: 'black'  
  });  
  
  A.fillColor = 'purple';  
  Atext = new PointText(new Point(A.position, A.position));  
  Atext.justification = 'center';  
  Atext.fillColor = 'black';  
  Atext.content = 'A';  
  Atext.fontSize = '3em';  
  
  B = new Path.Rectangle({  
    point: [A.position.x + canvasWidth * 0.25, A.position.y - canvasHeight * 0.25],  
    size: [canvasWidth * 0.50, canvasHeight * 0.50],  
    strokeColor: 'black'  
  });  
  
  B.fillColor = 'red';  
  Btext = new PointText(new Point(B.position, B.position));  
  Btext.justification = 'center';  
  Btext.fillColor = 'black';  
  Btext.content = 'B';  
  Btext.fontSize = '3em'
```

```
C = new Path.Rectangle({
  point: [A.position.x - canvasWidth * 0.25, A.position.y + canvasHeight * 0.25],
  size: [canvasWidth * 0.50, canvasHeight * 0.50],
  strokeColor: 'black'
});

C.fillColor = 'blue';
Ctext = new PointText(new Point(C.position, C.position));
Ctext.justification = 'center';
Ctext.fillColor = 'black';
Ctext.content = 'C';
Ctext.fontSize = '3em'

D = new Path.Rectangle({
  point: [A.position.x + canvasWidth * 0.25, A.position.y + canvasHeight * 0.25],
  size: [canvasWidth * 0.50, canvasHeight * 0.50],
  strokeColor: 'black'
});

D.fillColor = 'orange';
Dtext = new PointText(new Point(D.position, D.position));
Dtext.justification = 'center';
Dtext.fillColor = 'black';
Dtext.content = 'D';
Dtext.fontSize = '3em'

A.onMouseDown = function(event) {
  correctAnswer();
}
Atext.onMouseDown = function(event) {
  correctAnswer();
}

B.onMouseDown = function(event) {
  incorrectAnswer();
}
Btext.onMouseDown = function(event) {
  // incorrectAnswer();
}
C.onMouseDown = function(event) {
  incorrectAnswer();
}
Ctext.onMouseDown = function(event) {
  incorrectAnswer();
}
```

```
D.onMouseDown = function(event) {  
  incorrectAnswer();  
}  
Dtext.onMouseDown = function(event) {  
  incorrectAnswer();  
}  
}
```

Animating each answer object:

```
function scaleOne(){  
  A.scale(0.9990);  
  Atext.scale(0.9990);  
  Atext.position = new Point((A.position, A.position));  
  B.scale(0.9990);  
  Btext.scale(0.9990);  
  Btext.position = new Point((B.position, B.position));  
  C.scale(0.9990);  
  Ctext.scale(0.9990);  
  Ctext.position = new Point((C.position, C.position));  
  D.scale(0.9990);  
  Dtext.scale(0.9990);  
  Dtext.position = new Point((D.position, D.position));  
}
```

## Answer Type 2

Like answer type 1, the object creation function is called. Then the “onFrame” function calls the animation function during each frame.

```
two();
function onFrame(event){
  | moveTwo();
}
```

Creating each object in the animation:

```
function two(){
  A = new Shape.Circle(new Point(canvasWidth * 0.25, canvasHeight * 0.25), canvasWidth * 0.03 + canvasHeight * 0.03);
  A.strokeColor = 'black';
  A.fillColor = "red";
  Atext = new PointText(new Point(A.position, A.position));
  Atext.justification = 'center';
  Atext.fillColor = 'black';
  Atext.content = 'A';
  Atext.fontSize = '2em';

  B = new Shape.Circle(new Point(canvasWidth * 0.75, canvasHeight * 0.25), canvasWidth * 0.03 + canvasHeight * 0.03);
  B.strokeColor = 'black';
  B.fillColor = "orange";
  Btext = new PointText(new Point(B.position, B.position));
  Btext.justification = 'center';
  Btext.fillColor = 'black';
  Btext.content = 'B';
  Btext.fontSize = '2em';

  C = new Shape.Circle(new Point(canvasWidth * 0.75, canvasHeight * 0.75), canvasWidth * 0.03 + canvasHeight * 0.03);
  C.strokeColor = 'black';
  C.fillColor = "blue";
  Ctext = new PointText(new Point(C.position, C.position));
  Ctext.justification = 'center';
  Ctext.fillColor = 'black';
  Ctext.content = 'C';
  Ctext.fontSize = '2em';

  D = new Shape.Circle(new Point(canvasWidth * 0.25, canvasHeight * 0.75), canvasWidth * 0.03 + canvasHeight * 0.03);
  D.strokeColor = 'black';
  D.fillColor = "green";
  Dtext = new PointText(new Point(D.position, D.position));
  Dtext.justification = 'center';
  Dtext.fillColor = 'black';
  Dtext.content = 'D';
  Dtext.fontSize = '2em';

  A.onClick = function(event) {
    | correctAnswer();
  }
  Atext.onClick = function(event) {
    | correctAnswer();
  }

  B.onClick = function(event) {
    | incorrectAnswer();
  }
}
```

```
Btext.onClick = function(event) {  
  | incorrectAnswer();  
}  
C.onMouseDown = function(event) {  
  | incorrectAnswer();  
}  
Ctext.onMouseDown = function(event) {  
  | incorrectAnswer();  
}  
D.onMouseDown = function(event) {  
  | incorrectAnswer();  
}  
Dtext.onMouseDown = function(event) {  
  | incorrectAnswer();  
}  
}
```

Locations for each point that each object moves to:

```
var locations = [new Point(canvasWidth * 0.25, canvasHeight * 0.25), new Point(canvasWidth * 0.75, canvasHeight * 0.25),  
  | new Point(canvasWidth * 0.75, canvasHeight * 0.75), new Point(canvasWidth * 0.25, canvasHeight * 0.75)];  
var h = 0;  
var j = 1;  
var k = 2;  
var l = 3;  
var destinationA = locations[h];  
var destinationB = locations[j];  
var destinationC = locations[k];  
var destinationD = locations[l];
```

Function for animating each object:

```
function moveTwo(){  
  //This tutorial helped with moving the circles for this animation  
  //http://paperjs.org/tutorials/animation/creating-animations/  
  
  var vector = destinationA - A.position;  
  A.position += vector / 60;  
  
  if (vector.length < 2) {  
    destinationA = locations[h];  
    h++;  
    if(h > 3){  
      h = 0;  
    }  
  }  
  Atext.position = A.position;  
  
  var vector = destinationB - B.position;  
  B.position += vector / 60;  
  
  if (vector.length < 2) {  
    destinationB = locations[j];  
    j++;  
    if(j > 3){  
      j = 0;  
    }  
  }  
  Btext.position = B.position;  
  
  var vector = destinationC - C.position;  
  C.position += vector / 60;  
  
  if (vector.length < 2) {  
    destinationC = locations[k];  
    k++;  
    if(k > 3){  
      k = 0;  
    }  
  }  
  Ctext.position = C.position;
```

```
var vector = destinationD - D.position;
D.position += vector / 60;

if (vector.length < 2) {
  destinationD = locations[1];
  l++;
  if(l > 3){
    l = 0;
  }
}
Dtext.position = D.position;
}
```



**Answer Type 3**

Also, like Answer Type 1, function is called to create the objects, then the “onFrame” function calls a function to animate the objects.

```
three();  
function onFrame(event){  
|   moveThree();  
}
```

Creating both objects:

```
function three(){  
  A = new Path.Rectangle({  
    point: [0, canvasHeight * 0.25],  
    size: [canvasWidth * 0.50, canvasHeight* 0.45],  
    strokeColor: 'black'  
  });  
  
  A.fillColor = 'purple';  
  Atext = new PointText(new Point(A.position, A.position));  
  Atext.justification = 'center';  
  Atext.fillColor = 'black';  
  Atext.content = 'A';  
  Atext.fontSize = '3em';  
  
  B = new Path.Rectangle({  
    point: [canvasWidth * 0.50, canvasHeight * 0.25],  
    size: [canvasWidth * 0.50, canvasHeight* 0.45],  
    strokeColor: 'black'  
  });  
  
  B.fillColor = 'red';  
  Btext = new PointText(new Point(B.position, B.position));  
  Btext.justification = 'center';  
  Btext.fillColor = 'black';  
  Btext.content = 'B';  
  Btext.fontSize = '3em'  
  
  A.onMouseDown = function(event) {  
    | correctAnswer();  
  }  
  Atext.onMouseDown = function(event) {  
    | correctAnswer();  
  }  
  
  B.onMouseDown = function(event) {  
    | incorrectAnswer();  
  }  
  Btext.onMouseDown = function(event) {  
    | incorrectAnswer();  
  }  
}
```

Variable to help direct each object during animation:

```
var top = (canvasHeight * 0.20);  
var bottom = (canvasHeight * 0.80);  
var up = true;  
var nextA = top;  
var nextB = bottom;
```

Function to animate each object:

```
function moveThree(){  
  
    var vector = nextA - A.position.y;  
    A.position.y += vector / 60;  
  
    if (A.position.y <= canvasHeight * 0.21 || A.position.y >= canvasHeight * 0.79 ) {  
        if (nextA === top){  
            nextA = bottom;  
        }else{  
            nextA = top;  
        }  
    }  
    Atext.position = A.position;  
  
    var vector = nextB - B.position.y;  
    B.position.y += vector / 60;  
  
    if (B.position.y <= canvasHeight * 0.21 || B.position.y >= canvasHeight * 0.79 ) {  
        if (nextB === top){  
            nextB = bottom;  
        }else{  
            nextB = top;  
        }  
    }  
    Btext.position = B.position;  
}
```

## Login Check

This snippet contains the code that is used to check the username and password of a user logging in. “result(\$res)” is a function that decides what the pages displays depending on the result of the check. A similar function is used for creating an account where instead of both username and password being used, just the username is used.

```
$conn = get_db();

$stmt = $conn ->
|   prepare("select username, password from users where username = :username and password = :password");
//Both fields have been escaped just in case, the password has been hashed for security.
$stmt -> bindParam(':username', pg_escape_string($_POST['username']));
$stmt -> bindParam(':password', sha1(pg_escape_string($_POST['password'])));
//sha1(
$stmt -> execute();
|   $count = $stmt -> rowCount();
/*If the account exists and the given credentials are correct, the player's username is assigned
to a Session variable called "username". The result variable is also set, and used when calling the "result" function.
*/
if($count == 1){
|   $_SESSION["username"] = pg_escape_string($_POST['username']);
|   $res = true;
}else{
|   $res = false;
}
if(isset($_SESSION['username'])){
|   $res = true;
}
result($res);
```

## Creating a Quiz

This form is displayed when creating questions:

```
function questions() {
echo"<form id=\"quizCreate\" class=\"col-12 col-s-12\" action=\"quizInsert.php\" method=\"POST\">";
echo"
<div class=\"col-3 col-s-12\">
<label for=\"question\">Question:</label><br />
<input type=\"text\" id=\"question\" name=\"question\" required><br />
<p>Answer Type:</p>
<input type=\"radio\" id=\"A\" name=\"answer\" value=\"A\" onchange=\"show()\" checked>
<label for=\"A\">A</label><br />
<input type=\"radio\" id=\"B\" name=\"answer\" value=\"B\" onchange=\"show()\">
<label for=\"B\">B</label><br />
<input type=\"radio\" id=\"C\" name=\"answer\" value=\"C\" onchange=\"hide()\">
<label for=\"C\">C</label><br />
</div>

<div class=\"col-3 col-s-12\">
<label for=\"answer1\">Answer 1:</label><br />
<input type=\"text\" id=\"answer1\" name=\"answer1\" required><br />

<label for=\"answer2\">Answer 2:</label><br />
<input type=\"text\" id=\"answer2\" name=\"answer2\" required>
</div>
<div class=\"col-3 col-s-12\">

<label class=\"c\" for=\"answer3\">Answer 3:</label><br />
<input class=\"c\" type=\"text\" id=\"answer3\" name=\"answer3\"><br />

<label class=\"c\" for=\"answer4\">Answer 4:</label><br />
<input class=\"c\" type=\"text\" id=\"answer4\" name=\"answer4\"><br /><br />
</div>
<div class=\"col-3 col-s-12\">
<p>Correct Answer:</p>
<input type=\"radio\" id=\"1\" name=\"correct\" value=\"1\" checked>
<label for=\"1\">1</label><br />

<input type=\"radio\" id=\"2\" name=\"correct\" value=\"2\">
<label for=\"2\">2</label><br />

<input class=\"c\" type=\"radio\" id=\"3\" name=\"correct\" value=\"3\">
<label class=\"c\" for=\"3\">3</label><br />

<input class=\"c\" type=\"radio\" id=\"4\" name=\"correct\" value=\"4\">
<label class=\"c\" for=\"4\">4</label><br /><br />
</div>
<div class=\"col-12 col-s-12\">
<button type=\"submit\">Submit Question</button>
</div>
</form>";
}
```

Simple if statement to determine if the quiz creation is just starting or already in progress:

```
if(isset($_SESSION["currentQ"])){  
  
    questions();  
  
} else {  
  
    $_SESSION['quiz_id'] = pg_escape_string($_POST['quiz_id']);  
    $_SESSION["currentQ"] = 1;  
  
    $_SESSION["no_quiz"] = $_POST['no_quiz'];  
    questions();  
  
}
```

## Hosting a Quiz

Checking if code exists, setting the code then checking if that file already exists.  
Changing the code if the file exists.

```
if(!isset($_SESSION['code'])){
    echo"<div class=\"col-12 col-s-12\" id=\"quiz\">
    <h3>Ready to start?</h3>
    <p>Click the \"Start\" button below to display the next question</p>
    </div>";
    $code = rand(1000, 9999);
    if(file_exists($code . ".txt")){
        $code = rand(1000, 9999);
    }
    $_SESSION['code'] = $code;
    $_SESSION['currentQ'] = (int)1;
```

Saving answer details to a file:

```
$stmt2 = $conn ->
    prepare("select * from answers where quiz_id = :quiz_id and question_no = :question_no");
    $stmt2 -> bindParam(':quiz_id', pg_escape_string($_SESSION['quiz_id']));
    $stmt2 -> bindParam(':question_no', intval(pg_escape_string($_SESSION['currentQ'])));
    $stmt2 -> execute();
}
catch(Exception $e) {
    echo 'Exception -> ';
    var_dump($e->getMessage());
}

while ($row = $stmt2->fetch()) {
    echo"<p>Answer ". letter($row['answer_no']). ": ".$row['answer'] . "</p><br /><br />";
    $txt .= "<input type=\"hidden\" id=\"answer".$row['answer_no']."\" name=\"answer\".
    $row['answer_no']." value=\"\" . correct_chk($row['correct']).\">";
}
```

//This link helped with writing to file  
[https://www.w3schools.com/PHP/php\\_file\\_create.asp](https://www.w3schools.com/PHP/php_file_create.asp)

```
fwrite($file, $txt);
fclose($file);
$current = (int)$_SESSION['currentQ'];
$current = $current+1;
$_SESSION['currentQ'] = $current;
```

Display current question's details:

```
$stmt1 = $conn ->
    prepare("select * from questions where username = :username and quiz_id = :quiz_id and question_no = :question_no");
    $stmt1 -> bindParam(':username', pg_escape_string($_SESSION['username']));
    $stmt1 -> bindParam(':quiz_id', pg_escape_string($_SESSION['quiz_id']));
    $stmt1 -> bindParam(':question_no', intval(pg_escape_string($_POST['currentQ'])));
    $stmt1 -> execute();
}

catch(Exception $e) {
    echo 'Exception -> ';
    var_dump($e->getMessage());
}

$row = $stmt1->fetch();
echo"<p> Question No: ";
echo $row['question_no']."</p><br /><br />";
echo"<p> Question: ";
echo $row['question']."</p><br /><br />";
echo"<br /><br />";
$txt .= "<input type=\"hidden\" id=\"answer_type\" name=\"answer_type\" value=\"\" . $row['answer_type'].">";
```



## Interacting with Quiz

Checking hash of file and updating answers with its details. Functions written using AJAX:

```
var code = document.getElementById('code').value;
setInterval("getHash()", 1000);
function getHash() {
    //W3Schools helped as usual
    //https://www.w3schools.com/xml/ajax_intro.asp
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("hash").value =
                this.responseText;
        }
    };
    xhttp.open("POST", "hash_chk.php", true);
    xhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
    xhttp.send("code="+code);
    getAnswers();
}
```

“onFrame” function that controls the animation and transitions / interactions for connected users:

```
function onFrame(event){
    if(hash != document.getElementById("hash").value){
        hash = document.getElementById("hash").value;
        answer = false;
        achk = null;
        A = null;
    } else{
        if(document.getElementById("hash").value ==
            "57df429b2afffa63756a085af7bf295a3c66c8c90b0e35b6b3564254c8128473"
            || hash == "57df429b2afffa63756a085af7bf295a3c66c8c90b0e35b6b3564254c8128473"
            || document.getElementById("hash").value == "start"){
            waiting();
        }else if(document.getElementById("hash").value == "done"){
            done();
        }else if(hash == document.getElementById("hash").value && aCheck() && answer == false){
            if(A != null){
                movement();
            }else{
                create();
            }
        }
    }
}
```