2020

Manchester
Metropolitan
University

An interactive web-based learning package for teaching HTML

Final year project

Student name: Lewis Frater (17081955)

Project Supervisor: Marie Carroll

No part of this project has been submitted in support of an application for any other degree or qualification at this or any other institute of learning. Apart from those parts of the project containing citations to the work of others, this project is my own unaided work.

Ethos reference number: 13002

Signed:

I Gruter

Figure Contents

Contents

1 Introduction

1.1 Project Description

In an ever more digital age, using websites is part and parcel of many people's daily lives across the developed world. They allow businesses to advertise their services to a wide audience, a person to share their experiences on social media, it's remarkable how technology has advanced in the past 20 years to enable such interconnectivity between different people from across the whole planet.

This report will provide the steps and details of creating an interactive web-based learning package which teaches the user HTML (HyperText Markup Language), the language under the hood of every website created. It will implement ideas from previously researched learning methodologies, as well as providing insight of how the human brain learns new concepts.

1.1.1 Report Outline

This report is organised into eight different sections which will discuss various aspects of the project. A breakdown of the sections is below to provide a clearer outline of how this report is structured.

1. Introduction – This will be the opener to the report which will give the reader a brief overview of the project and the background of it.

2. Literature Review – This is the research section of the project, obtaining information from reliable sources about learning methodologies and theories.

3. Product Design – This is where the base of the report will be, it will show the thought process behind the steps taken in the design aspect of this project.

4. Implementation – This will detail how the project was managed, which tools were used to produce it, including code snippets to help explain how different pieces of functionality work and the file structure.

5. Product Evaluation – The evaluation will talk about different parts of the project which worked well as well as what could be changed and improved. It will also highlight any problems encounter throughout the development process.

6. Conclusion – The conclusion will relate to what was achieved and which objectives were met. It will highlight issues and potential future work.

7. References – This is a list of referenced work and research papers relating to this project.

8. Appendices – Additional documents relating to the project itself.

1.2 Project Background

Tim Berners-Lee invented the web, along with HTML in 1989. In this early stage it was used as a means of organising and pooling information, usually research papers. Instead of research papers being individual files which a computer has to download, Tim Berners-Lee had the idea of linking the text within the file itself, so there were cross-references from each different research paper leading to another, this is how it became known as the web of information, the world wide web. [1]

Hyper Text Markup Language is the bricks and mortar of the world wide web. It organises the structure of web content and uses Hyper Text to link one webpage to another. It is often used alongside Cascading Style Sheets to describe the appearance and JavaScript to provide interactivity and functionality to the webpage.

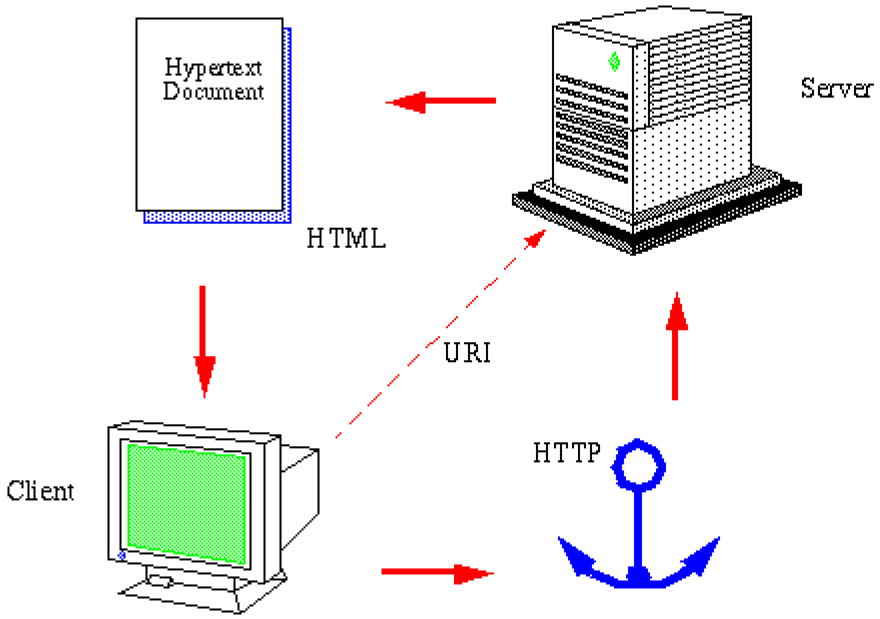


Figure 1: An illustration of the world wide web, adopted from (W3 1994)

As seen in Figure 1, the client makes a request via a unique resource identifier using HTTP to another computer acting as a server. The server contains the HTML document the client is attempting to access through the HyperText on a webpage. This enables the client to jump from one webpage to another, navigating through the nearly endless sources of information within the world wide web.

There are currently over 1.7 billion websites on the world wide web, of which over 200 million are actively maintained. [2] This emphasises how vast the world wide web is.

1.3 Aims and Objectives

1.3.1 Aims

The aim of this project is to teach HTML to beginners (16-18 year olds) in an intuitive manner by creating an interactive website containing information and quizzes.

1.3.2 Objectives

The following objectives must be met to ensure a successful project.

- Research and analyse existing systems and other sources which teach HTML, identifying their strengths and weaknesses.
- Research learning methodologies to assist how to help the user absorb new information.
- Plan website design, content and structure then implement the front-end.
- Plan and design a website back-end, using a database containing tables holding user information and user progress on quizzes.
- Use the website back-end to implement a user login and registration system.

- Reflect on and review the interactive website by asking fellow students to trial it so I can act on the feedback – Rigorous testing.

2 Literature review – Learning how to learn

2.1 Molecular level of learning

‘…the emphasis is on what students need to learn, whereas little emphasis—if any—is placed on training students how they should go about learning the content and what skills will promote efficient studying to support robust learning.*’* – [3]

In the last decade neuroscience researchers have been able to go inside the brain to observe how learning occurs at the molecular level. In the past, neuroscientists have believed that memory formation and learning are acquired by the strengthening and weakening of connections between neurons. [4] The neuroscientists have recently been proved correct in their belief.

In an experiment involving mice, neuroscientists were able to observe the actions of the brain while performing, and learning, a new task. The researchers found that when two neurons frequently interact, a bond is formed that allows them to transmit more efficiently and precisely. This enables the subject to more easily recall complete memories. In contrast, when neurons infrequently interacted, the communication between them was often incoherent, resulting in a faulty memory, or in the worst case, no memory at all. [5]

In evaluation of the research above, it is clearly the case, that initiating neurons to fire to one another, as frequently as possible to retain and learn new topics, is essential in the learning process. This literature review will cover some of the learning strategies that will be used on the project (Creating an interactive website to teach HTML).

2.2 Learning strategies

Strategies of learning refer to the methods that conscious beings use to learn. It is a process of learning about learning, or thinking about thinking, which is metacognition.

“Metacognition refers to one’s knowledge concerning one’s own cognitive processes, or anything related to them, e.g., the learning-relevant properties of information or data. For example, I am engaging in metacognition if I notice that I am having more trouble learning A than B; if it strikes me that I should double check C before accepting it as fact.” - Flavell, J. (1976) [6]

Metacognition is relevant to the study of learning strategies as they are both concerned with the control processes in so far that the individual has awareness of their thinking or learning processes. There are lots of learning theories which highlight the importance of different learning strategies such as:

Double loop learning - Argyris, C. (1976) [7]

Lateral thinking - DeBono, E. (1967) [8]

Conversation theory - Pask, G. (1975) [9]

Conditions of learning – Gagne, R. (1985) [10]

This literature review will look closer at Gagne’s Conditions of learning.

2.2.1 Conditions of learning

Robert Mills Gagne (1916-2002) was an educational psychologist with 198 works in 750 publications in 10 languages and 11,759 library holdings [13]. R. M. Gagné proposed a taxonomy of educational objectives in the UK first edition (1969) of his book ‘The Conditions of Learning’ which was later revised and updated in 1985.

Gagne’s theory stipulates that there are many different levels of learning. Each different level requires different instructions, he identifies fiver major categories of learning: intellectual skills, verbal information, cognitive strategies, attitudes and motor skills. Different conditions are required for each type of learning, for example, if cognitive strategies are to be learned, there must be a chance to practice providing new solutions to overcome an obstacle. To learn an attitude, a persuasive argument or a credible role model must be exposed to the learner.

The theory provides a blueprint of nine instructional events and the cognitive processes corresponding to those events:

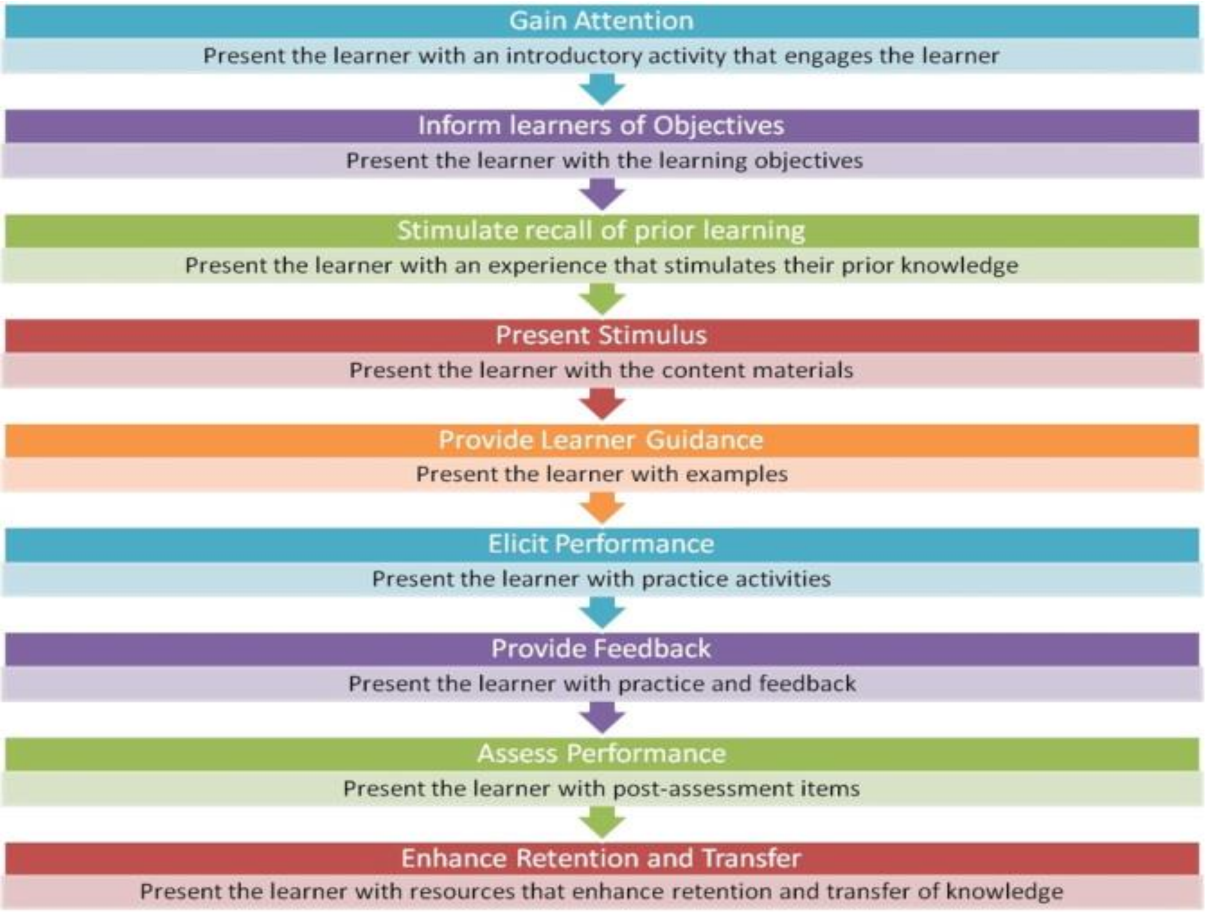

Gain Attention
Present the learner with an introductory activity that engages the learner
Inform learners of Objectives
Present the learner with the learning objectives
Stimulate recall of prior learning
Present the learner with an experience that stimulates their prior knowledge
Present Stimulus
Present the learner with the content materials
Provide Learner Guidance
Present the learner with examples
Elicit Performance
Present the learner with practice activities
Provide Feedback
Present the learner with practice and feedback
Assess Performance
Present the learner with post-assessment items
Enhance Retention and Transfer
Present the learner with resources that enhance retention and transfer of knowledge

Figure 2: Gagne’s nine instructional events (Adopted from instructionaldesign.org)

According to Gagne these events provide the necessary conditions for learning and serve as the foundation for selecting appropriate media to help the learner learn the desired task (Briggs, Gagne & Wager, 1992). [11]

Here is a more in-depth analysis of Gagne's nine instructional events.

1. **Gaining attention (reception)**
   In this event of instruction, to facilitate learning to take place, the teacher must first catch the attention of the student. Curiosity motivates students to learn. An opening statement could be a thought-provoking question, or a multimedia program that begins with exciting animations accompanied by music to stimulate the auditory and visual senses.

2. **Informing learners of the objective (expectancy)**
   A list of learning objectives should be compiled to show the students what is expected of them in a particular lesson. This activates the internal process of expectancy which helps motivate the student to complete a lesson. Usually, these objectives should form the basis for assessment further down the line. Learning objectives are presented in the from "Upon completing this lesson you will be able to…." to describe what skills will be accomplished and how the student will be able to use the knowledge, a demonstration may be appropriate.

3. **Stimulating recall of prior learning (retrieval)**
   Associating prior knowledge with new information can help facilitate the learning process. If there are links to personal experience and knowledge, it is easier for students to store information in their long-term memory. This is backed up by information provided earlier in this literature review "when two neurons frequently interact, a bond is formed that allows them to transmit more efficiently and precisely". This association with prior knowledge helps facilitate this bond between neurons hence the encoding of information into the long-term memory is more successful.

4. **Presenting the stimulus (selective perception)**

This is where new content is presented to the learner. The content should be segmented and organised in a meaningful way. At first it is typically explained rather than demonstrated. To appeal to different learning modalities, it should be a mixture of media, text, figures, graphics or sound. A consistent presentation style should be adhered to while the chunking of information should always try to avoid memory overload.

5. **Providing learning guidance (semantic encoding)**
   To help learners encode information for long-term storage in the memory, a variety of different approaches to demonstrate the same information must be used. These guidance strategies should include the use of examples, case studies or simple analogies to show the relevance of the material.

6. **Eliciting performance (responding)**
   The student is required to apply knowledge in order to practice the new skill or behaviour. This provides an opportunity for the student to confirm their understanding. This can be done by asking the student to explain a concept or answer a question, as previously discussed, repetition increases the likelihood of retention in the long-term memory.

7. **Providing feedback (reinforcement)**
   In this event instruction it is important that as students learn or practice new behaviour, the teacher must provide immediate feedback of their performance. Exercises within lessons should be used for comprehension purposes, not for formal scoring. The guidance and answers helping the student at this stage is usually called formative feedback.

8. **Assessing performance (retrieval)**
   In order to assess a student's performance on whether information has been learnt, the student must undertake a test. The test should be completed without the ability to receive hints or additional coaching. This enables the assessor to see how well a student is learning and retaining the information. A

commonly accepted score for mastery of material is a result of 80% and higher.

9. **Enhancing retention and transfer (generalisation)**
   In order to enhance retention, a student must be provided with additional practice as well as being put in transfer situation. A transfer situation is a means of the student becoming the teacher. This can include the student reviewing training materials for another person to learn, it can greatly increase the student's retention of information in the long term.

Gagne's nine instructional events provide a foundation to structure content to help students learn more efficiently.

2.3 My approach

Different people have varying amounts of success using different learning strategies. What is greatly helpful for one person, may not suitable for another person. However, there are underlying facts that precedes the diverse set of learning theories:

Repetition –

There is much weight to the old cliché "*Practise makes perfect*", learning content and retaining information is paramount. As discussed previously, the more the neurons communicate with each, the more coherent the memory.

Asking questions –

Upon reading content, to enable a person to achieve an understanding, it is necessary to answer questions about the content they have just read. Whether the questions are asked directly to the person by the content, or the person giving themselves questions to answer. It is vital to the learning and understanding of information.

Retrospection –

A person should review their progress by skimming the content and again, ask questions about it and correct any previous errors. This will solidify the knowledge and understanding a person has of a topic.

Taking all these points into account, it is with utmost importance that the content of the website is laid out in a logical and coherent manner. This is the plan that will tackle these targets:

- The information given about a topic will be expressed in straightforward and unambiguous language

- Questions will be engrained in the content section to encourage the user to think about what they are reading more carefully. (Answers to these questions will be within the content).

- A quiz section will assess the user’s understanding. Some questions that were engrained in the content section will be asked here to aid the retainment of information.

In conclusion of this literature review, we have covered the core concepts that underpin Gagne’s Conditions of Learning. These learning principles will be tried to be implemented in the final application to help ensure the target audience is learning via the most optimal techniques in teaching.

3 Product Design

3.1 Design Introduction

The decisions taken in the design process was heavily influenced by the target audience. The project is specifically aimed at 16-18 year olds in college without much experience of coding in HTML. It is paramount that the users can easily navigate through the website without being confused. A simple page layout and a navigation bar with clearly labelled pages had to be at the forefront of the design.

As the project is aimed at helping the users to learn new information, the content of the pages had to be clear, concise and descriptive which takes the form of multimedia such as videos or plain text stating it. It was decided that the content would be split up into five different chapters on the learning page of the website. The material provided had to be tested with some form of user interaction, so that’s why an interactive quiz had to be designed which tested the knowledge of what the user just read or watched at the end of each chapter of information.

It was decided a simple registration system had to be designed to provide extra functionality for the users that really enjoyed the website. Allowing users to sign up and log in to access the interactive HTML environment, the playground page. The playground page would be to allow users to doodle their HTML code on the website directly, and see instant results, this provides extra interactivity for the user to help the learning process.

3.2 Project Management

In order to complete the project in a timely manner, a framework for managing the project had to be implemented. A way to make the project accessible to potential future employees was also of great importance. The project had to have a changelog of any edits made to files to allow for transparency and integrity. This is achieved by using scrum and GitHub.

3.2.1 An overview of scrum

Scrum is one of the most popular agile frameworks in software development. It is an iterative approach, the term for this in scrum is a sprint. In this process, teams use specific roles to make sure they meet the goals of each part of the project. This allows for the project to be highly adaptive throughout the development cycle to meet potentially changing requirements.

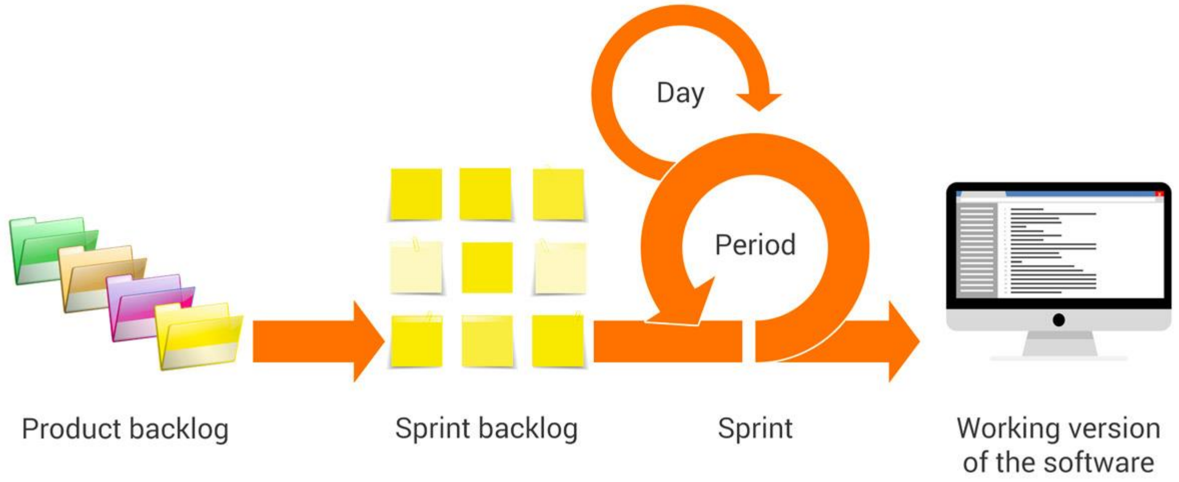


Figure 3: The scrum approach (adopted from https://bright-side-of-life.com/2017/03/03/winning-with-scrum-approach/).

As you can see in figure 3, the sprint is at the heart of the scrum approach. The sprint is what allows this approach to be highly adaptive and flexible.

Within each sprint a functional part of the product is developed and tested, usually until the product owner accepts it. When one sprint is finished, another sprint starts on a new piece of functionality. The functionality becomes a potentially shippable increment of the overall product, as when the product owner determines that enough value exists, it can be released [12].

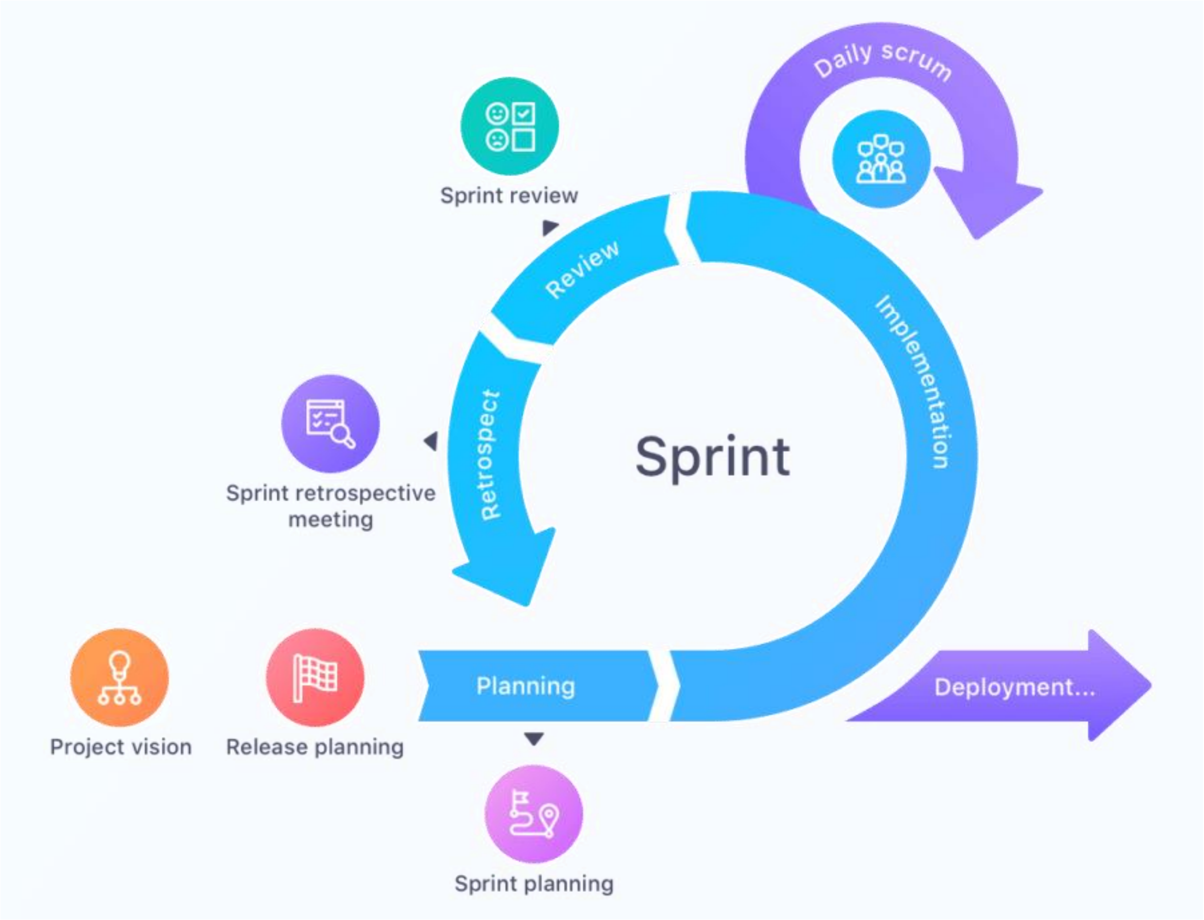


Figure 4: An overview of a sprint (adopted from https://hive.com/blog/scrum-project-management/).

The core principle of a sprint is its cyclical nature. The sprint, as well as the processes within it, planning, implementation, review and retrospection, repeates over and over as seen in figure 4.

There are tenets of inspection and adaptation on a daily basis as part of a scrum project:

- During a sprint, constant inspections are conducted to assess progress toward the sprint goal.
- A scrum review is held to see what was completed in previous days and to coordiate what the next feature will be implemented using the sprint. Ultimately bringing the product closer towards the release goal.
- At the end of sprint, a sprint retrospective review is held to assess performance and plan any necessary adaptations as required by the client.

3.2.2 Tracking changes with GitHub

In order to keep a backup, as well as track all the changes made day by day of the project, GitHub was used.

Developers use GitHub to store their projects. It’s very useful because while providing a back-up for your project, it takes advantage of Git, a version control system. It enables you to store file changes efficiently as well as upholding file integrity.

In my project it was primarily used to track changes and provide a back-up in case something went wrong. I could pull the most recent back-up files and start the new piece of functionality again if it was needed.

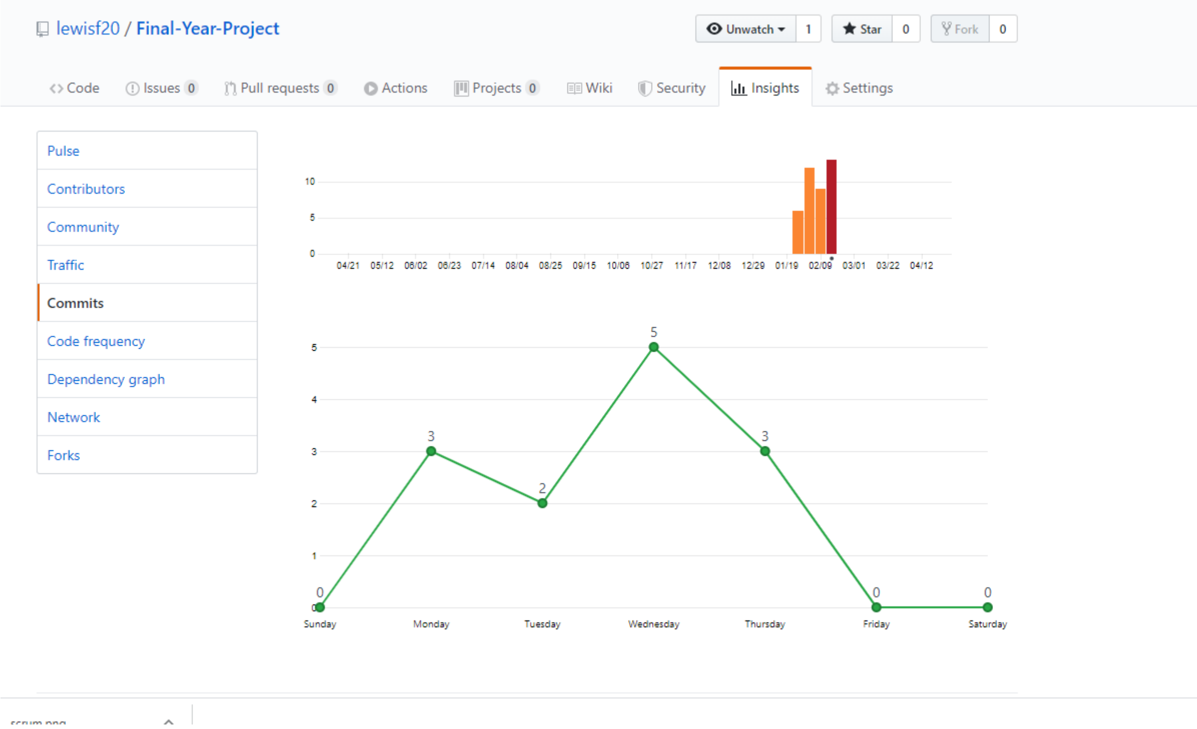
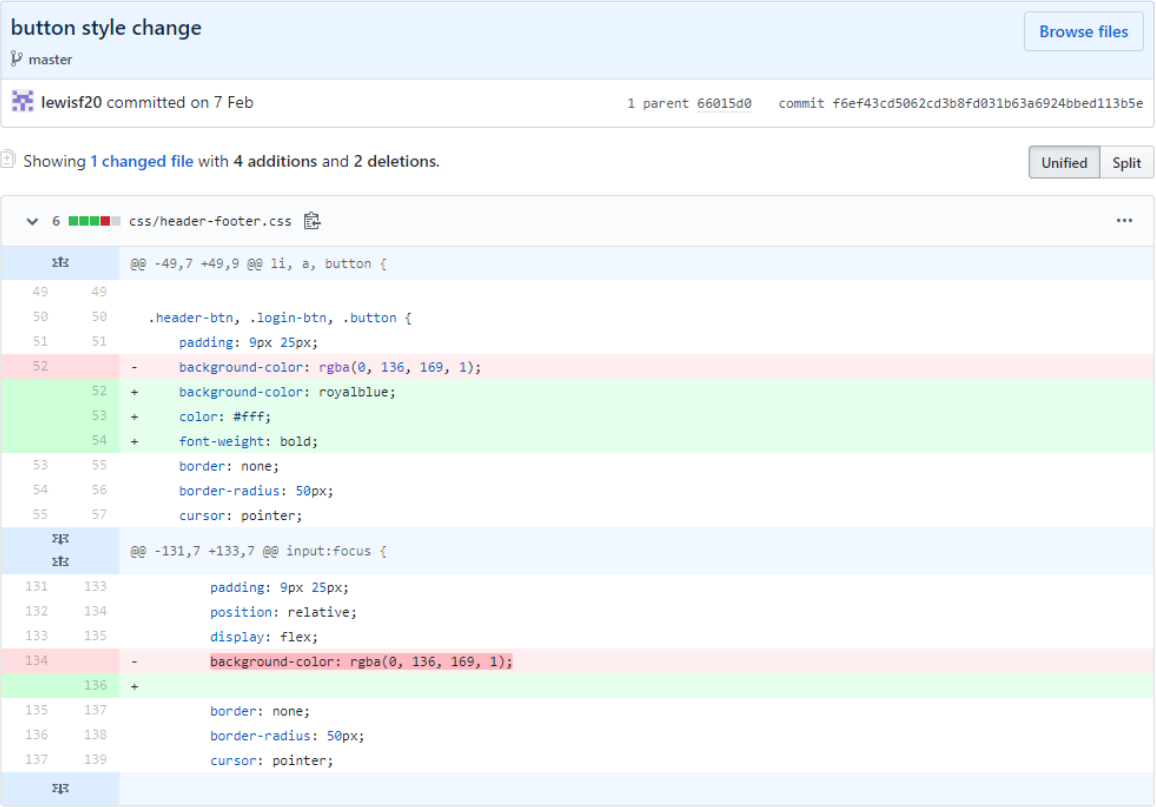


Figure 5: My GitHub commits (publicly available at https://github.com/lewisf20/Final-Year-Project/graphs/commit-activity).

Commits are uploads to the GitHub repository when a piece of content or functionality is completed. GitHub provides a change log highlighting any editing of files as seen in a figure below containing a random example.



button style change

lewisf20 committed on 7 Feb

1 parent 66015d0   commit f6ef43cd5062cd3b8fd031b63a6924bbed113b5e

Showing 1 changed file with 4 additions and 2 deletions.

6 css/header-footer.css

```
@@ -49,7 +49,9 @@ li, a, button {

   .header-btn, .login-btn, .button {
       padding: 9px 25px;
-      background-color: rgba(0, 136, 169, 1);
+      background-color: royalblue;
+      color: #fff;
+      font-weight: bold;
       border: none;
       border-radius: 50px;
       cursor: pointer;
@@ -131,7 +133,7 @@ input:focus {
           padding: 9px 25px;
           position: relative;
           display: flex;
-          background-color: rgba(0, 136, 169, 1);
+
           border: none;
           border-radius: 50px;
           cursor: pointer;
```

Figure 6: GitHub highlighting changes in a file after a commit (available at https://github.com/lewisf20/Final-Year-Project/commit/f6ef43cd5062cd3b8fd031b63a6924bbed113b5e).

In this figure you can see the changes made to that file since the last commit. The parts marked in green indicated additions to the file, while the parts marked in red highlight removals. This provides great insight to the overall project as other people and developers can look at your code and see into your thought processes throughout the creation of the project.

GitHub was a vital aspect in the management of this project. It also provides a way to show potential employers my work in the future. It allows the project to be transparent with all the changes made, it shows the project has integrity which is of utmost importance to potential employers.

3.3 Functionality Design

In this section we will discuss the functionality required on the website to achieve the project objectives. The many different languages used to accomplish this are expanded upon in more detail in the next section.

The website comprises of HTML, PHP, CSS and JavaScript. HTML is used to structure a page while CSS alters the aesthetic look. PHP is used to create user sessions when they log in and connect to the databases required to produce the login/registration system. JavaScript is used to create the interactive quizzes.

As previously mentioned, a registration and log in system had to be created. This involves storing user information in database tables. A user would be allowed to change their password and email on profile page. On the profile page a user would be able to see their own scores on each quiz so another database table holding the scores of each user had to be created. A global leaderboard page would allow a user to compare their scores to every other user that have completed the same quizzes.

3.4 Language choices

Below is a list of the languages used in the development of this project with a short description of why they were chosen.

HTML – The project is a web-based learning package to teach HTML so it is at the forefront of the languages used. There is no avoiding HTML when creating a website. It is also a language that has not been touched in any units in the 3rd year at university, despite this, I am very comfortable coding in HTML.

CSS – Cascading Style Sheets is used to describe the style of an HTML document. This was used to implement the colour schemes and adjust the layout of how the HTML elements should be displayed for the user. Media queries allows the website to become responsive depending on the size of the display the user is on.

JavaScript – This is the scripting language for webpages, it allows functionality to be implemented that provides user interactivity with the website which is ideal for this project. The interactive quiz was made using this language.

PHP – PHP is a widely used server-side language. It allows dynamic web applications to be built as it interacts with databases. This is perfect for creating the registration system and all other pieces of functionality which require interaction with databases.

While I have used this language before in my course of study, I have never felt confident using it. Despite this, I decided it was necessary as well as being a great challenge, so I put in a lot of research to help understand and use this language effectively.

MySQL – MySQL is integrated into PHP; It allows you to query and manipulate databases. Information stored in tables can be created, retrieved, updated, and deleted. This is important as it provides the means to interact with the information stored in the tables to allow users to create accounts, log in and view the leaderboards in this project.

3.5 Layout Design

As mentioned previously, the layout of the website had to be simple and easy to navigate. Below are some wireframe designs that were created using pen and paper.

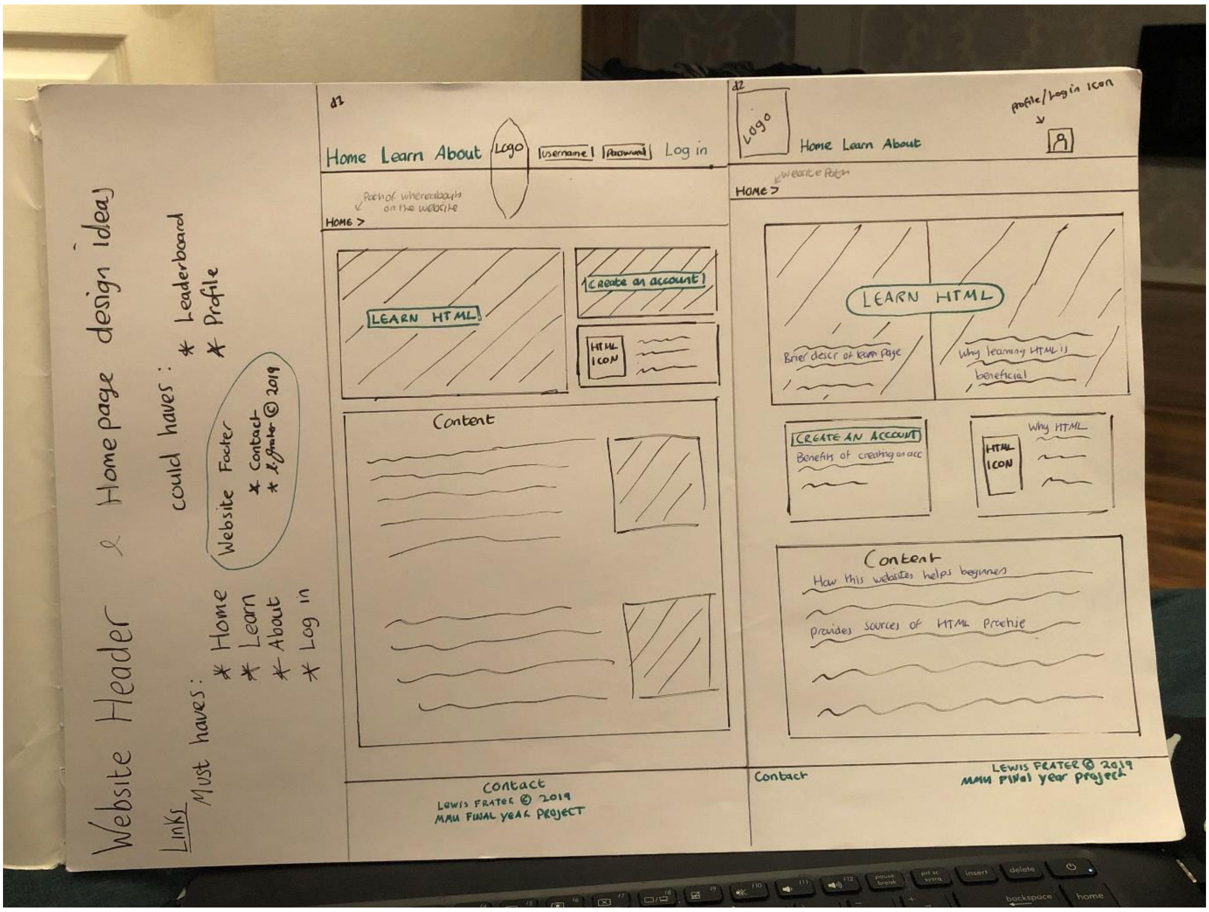


Figure 7: Wireframe design for the homepage of the website.

This is the initial page a user would land on, the homepage. It uses boxes to split different pieces of information into segments, so it is clear for the user, it is later discussed how CSS grid was used to implement this type of layout.

The first ideas for the must haves of navigation are displayed here, Home, Learn, About and Log in along with some ideas that were considered at the time with Profile and Leaderboard (Which was implemented but not specifically on the navigation bar). This page provided the base of how the user interface would look throughout the website, with minor tweaks, but the box format of the page allowed each page to be very flexible with the different content that was required on them.

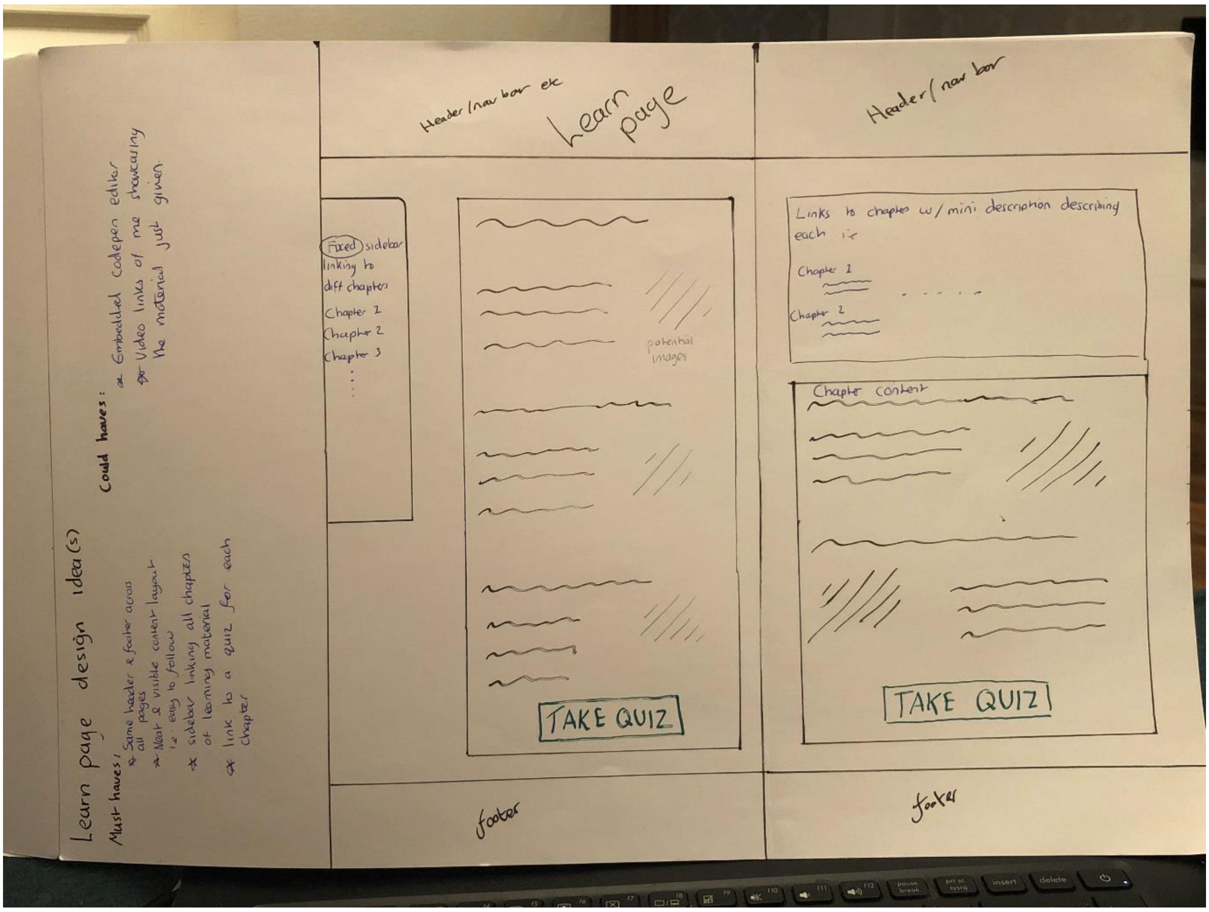

Learn page design idea(s)
Must haves:
* Same header & footer across all pages
* Neat & visible content layout i.e. easy to follow
* sidebar linking all chapters of learning material
* link to a quiz for each chapter
Could haves:
* Embedded codepen editor
* Video links of me showcasing the material just given.
Header/nav bar etc
Learn page
Fixed sidebar linking to diff chapters
Chapter 1
Chapter 2
Chapter 3
potential images
TAKE QUIZ
footer
Header/nav bar
Links to chapters w/ mini description describing each i.e
Chapter 1
Chapter 2
Chapter content
TAKE QUIZ
footer

Figure 8: Wireframe design for the learn page of the website.

This is the page which would contain the multimedia and content for the user to learn from. It was decided an extra navigation bar for the different chapters would be added here to provide simple navigation for the user.

The idea of having the same header and footer across the whole website was decided here to ensure it was familiar to the user across all the pages. A link to a quiz at the end of each chapter so the user can be tested on the information they have just taken in.

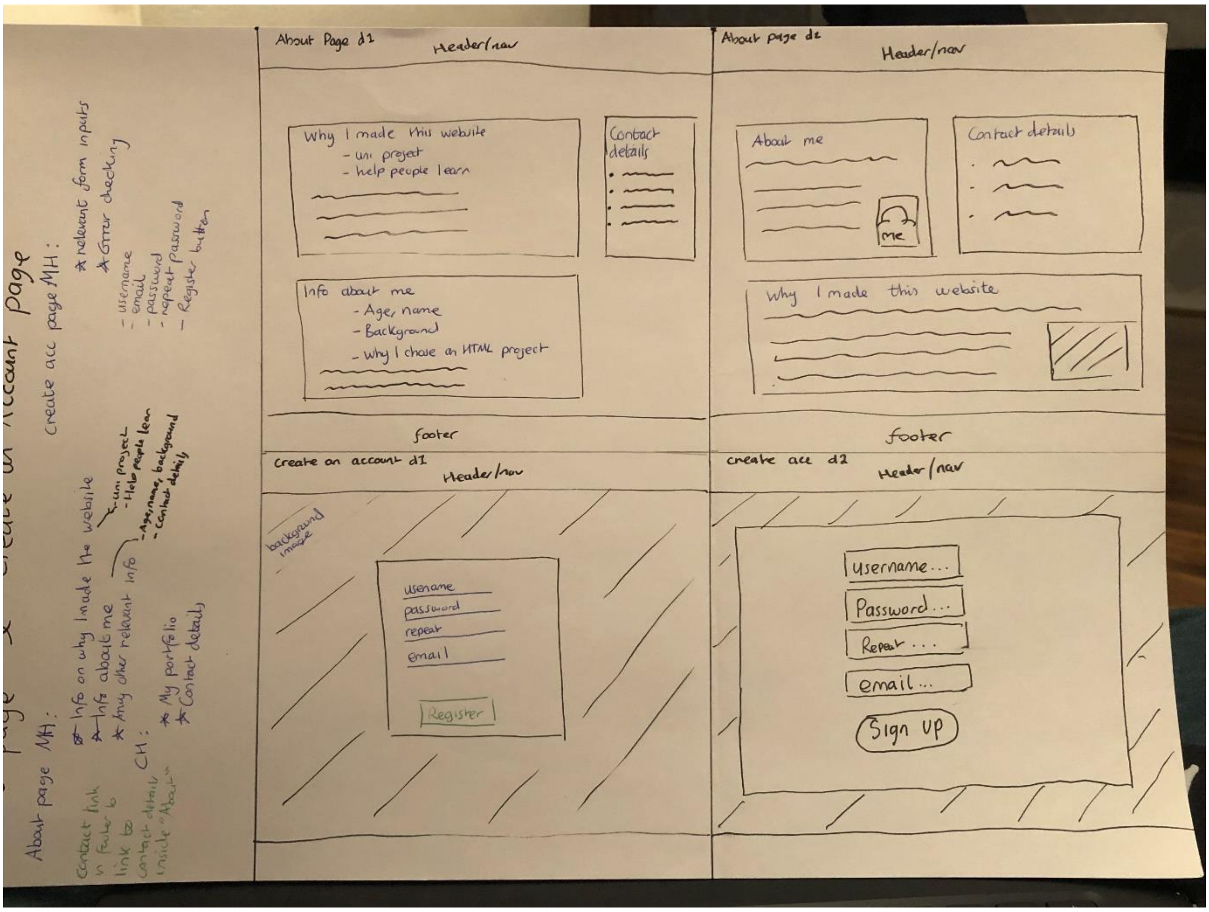


Figure 9: Wireframe designs for the about page and the create and account page.

The about page has the simple box layout previously discussed. It will contain information about the creator, such as contact details, the background and a curriculum vitae. It will also contain information about the website itself.

The registration page has a straightforward layout which contains the relevant form inputs and a button to create an account.

The idea of having a static background image behind the boxes was decided here to help the website look aesthetically pleasing for the users.

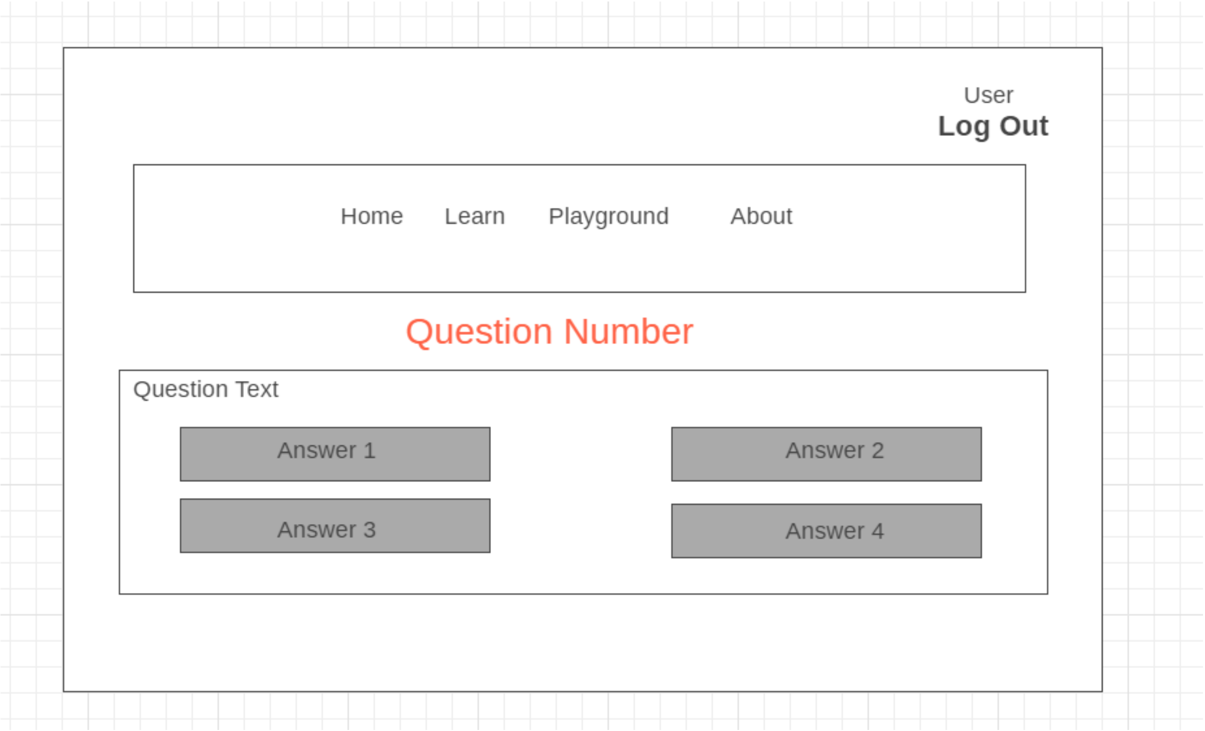


Figure 10: The layout of an interactive quiz (Created using https://wireframe.cc/).

The interactive quiz involves the user reading a question, then answering it by clicking on one of the four provided answers. The layout had to be clear to help the user not be confused.

It was decided to highlight the colour of the background green for every correct answer, conversely, highlighting the background red to indicate a wrong answer. This ensures the user receives convenient visual feedback for each answer given. At the end of the quiz, it would tell the user how many were correct out of the total number of answers.

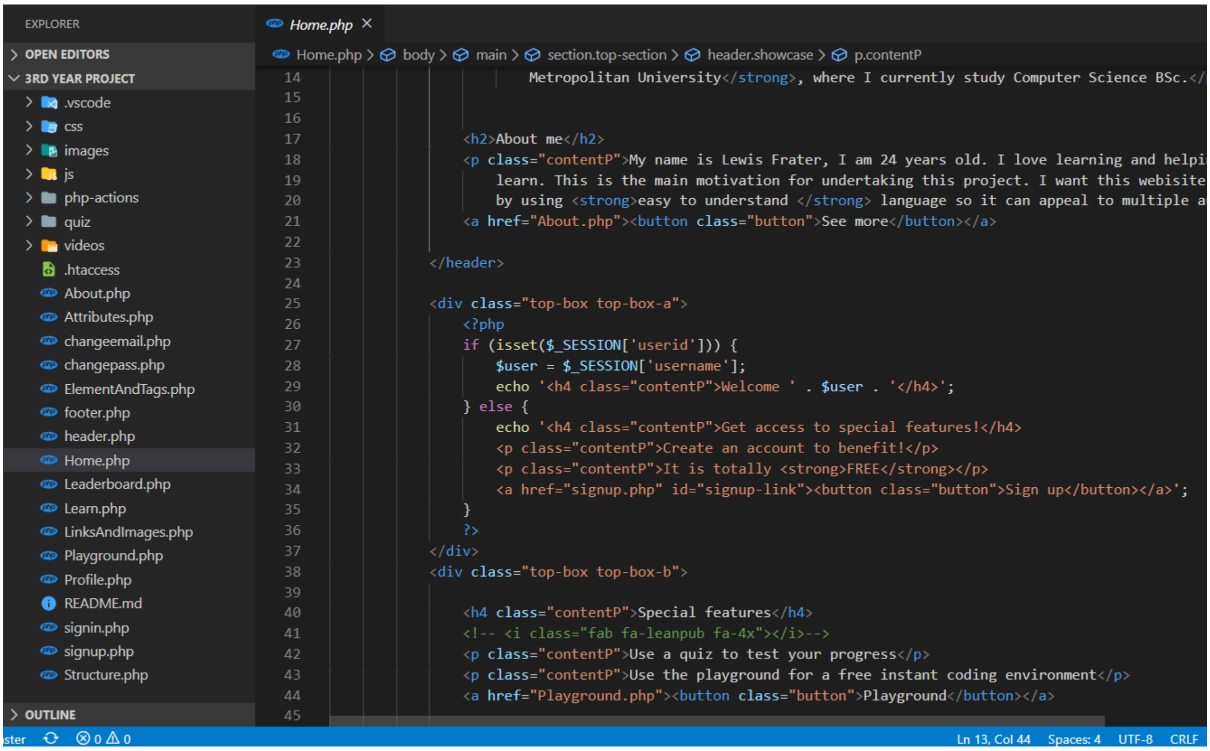
4 Implementation

4.1 Implementation Overview

In creation of this project, three main areas had to be completed. The front-end, the back-end and the JavaScript interactive quiz. The first stage implemented was the front-end, this was done using HTML within PHP files and CSS for the styling.

4.2 File Structure

As this was going to be a sizeable project, a tidy file structure had to be implemented in order to keep the project organised.



```
                        Metropolitan University</strong>, where I currently study Computer Science BSc.</


                <h2>About me</h2>
                <p class="contentP">My name is Lewis Frater, I am 24 years old. I love learning and helpi
                    learn. This is the main motivation for undertaking this project. I want this webisite
                    by using <strong>easy to understand </strong> language so it can appeal to multiple a
                <a href="About.php"><button class="button">See more</button></a>

            </header>

            <div class="top-box top-box-a">
                <?php
                if (isset($_SESSION['userid'])) {
                    $user = $_SESSION['username'];
                    echo '<h4 class="contentP">Welcome ' . $user . '</h4>';
                } else {
                    echo '<h4 class="contentP">Get access to special features!</h4>
                    <p class="contentP">Create an account to benefit!</p>
                    <p class="contentP">It is totally <strong>FREE</strong></p>
                    <a href="signup.php" id="signup-link"><button class="button">Sign up</button></a>';
                }
                ?>
            </div>
            <div class="top-box top-box-b">

                <h4 class="contentP">Special features</h4>
                <!-- <i class="fab fa-leanpub fa-4x"></i>-->
                <p class="contentP">Use a quiz to test your progress</p>
                <p class="contentP">Use the playground for a free instant coding environment</p>
                <a href="Playground.php"><button class="button">Playground</button></a>

```

Figure 11: Screen shot of the file structure in Visual Studio Code and a part of the home page.

In figure 11 above, you can see the file structure to the left. Files are organised into their specified folders. Any files concerning the styling of the website are contained within the CSS folder. Images and videos have their own respective folders. The back-end of the project was stored in the php-actions folder. The root folder contains all the different webpages that can be accessed by a user. Any additional functionality and interactivity were stored in the js folder.

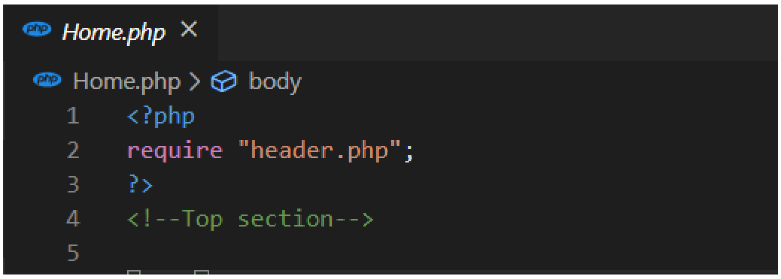
The organised file structure allowed me to keep track of where each part of the project was contained. It allowed for easy changes as I would know where to look for different parts of the project; content, styling, back-end and functionality.

4.3 Front-end implementation

The website had to allow users to signup and login, so the decision was made to use the .php file extension instead of .html. This allowed me to create user sessions in future to indicate a user is active which is discussed in the back-end section.

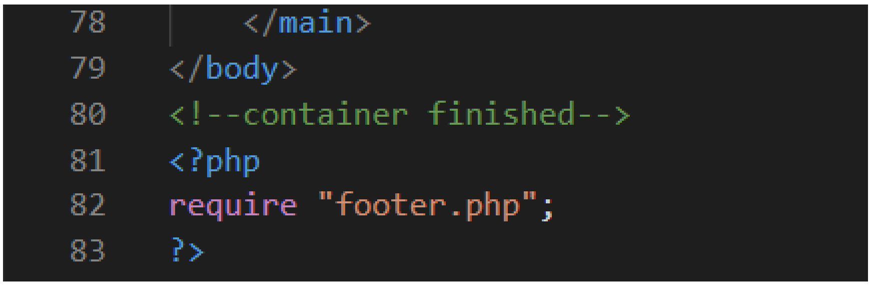
4.3.1 Using PHP in the front-end

PHP also allowed me to create a header and footer which was consistent throughout the website. A simple “require” command at the beginning and end of each of the webpages as shown in the figures below.



```
<?php
require "header.php";
?>
<!--Top section-->

```

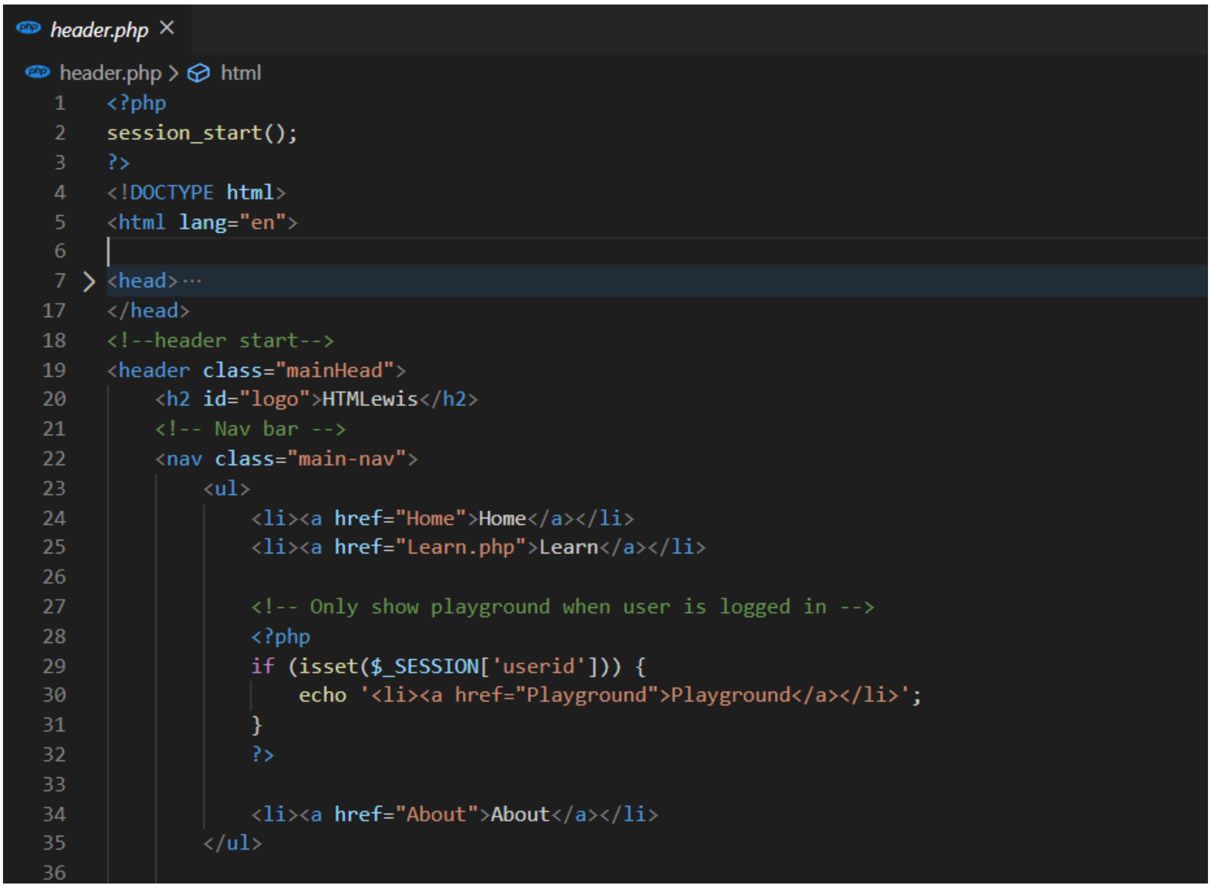
Figure 12: Using the php require command for the header.



```
    </main>
</body>
<!--container finished-->
<?php
require "footer.php";
?>
```

Figure 13: Using the php require command for the footer.

This was consistent over every viewable webpage. Next we will have a closer look of what is inside of the header.php file.



```
header.php ×
header.php > html
1   <?php
2   session_start();
3   ?>
4   <!DOCTYPE html>
5   <html lang="en">
6   
7 > <head>···
17  </head>
18  <!--header start-->
19  <header class="mainHead">
20      <h2 id="logo">HTMLewis</h2>
21      <!-- Nav bar -->
22      <nav class="main-nav">
23          <ul>
24              <li><a href="Home">Home</a></li>
25              <li><a href="Learn.php">Learn</a></li>
26  
27              <!-- Only show playground when user is logged in -->
28              <?php
29              if (isset($_SESSION['userid'])) {
30                  echo '<li><a href="Playground">Playground</a></li>';
31              }
32              ?>
33  
34              <li><a href="About">About</a></li>
35          </ul>
36  
```

Figure 14: The header.php file contents – with the head collapsed (lines 7-17) in order to see the navigation.

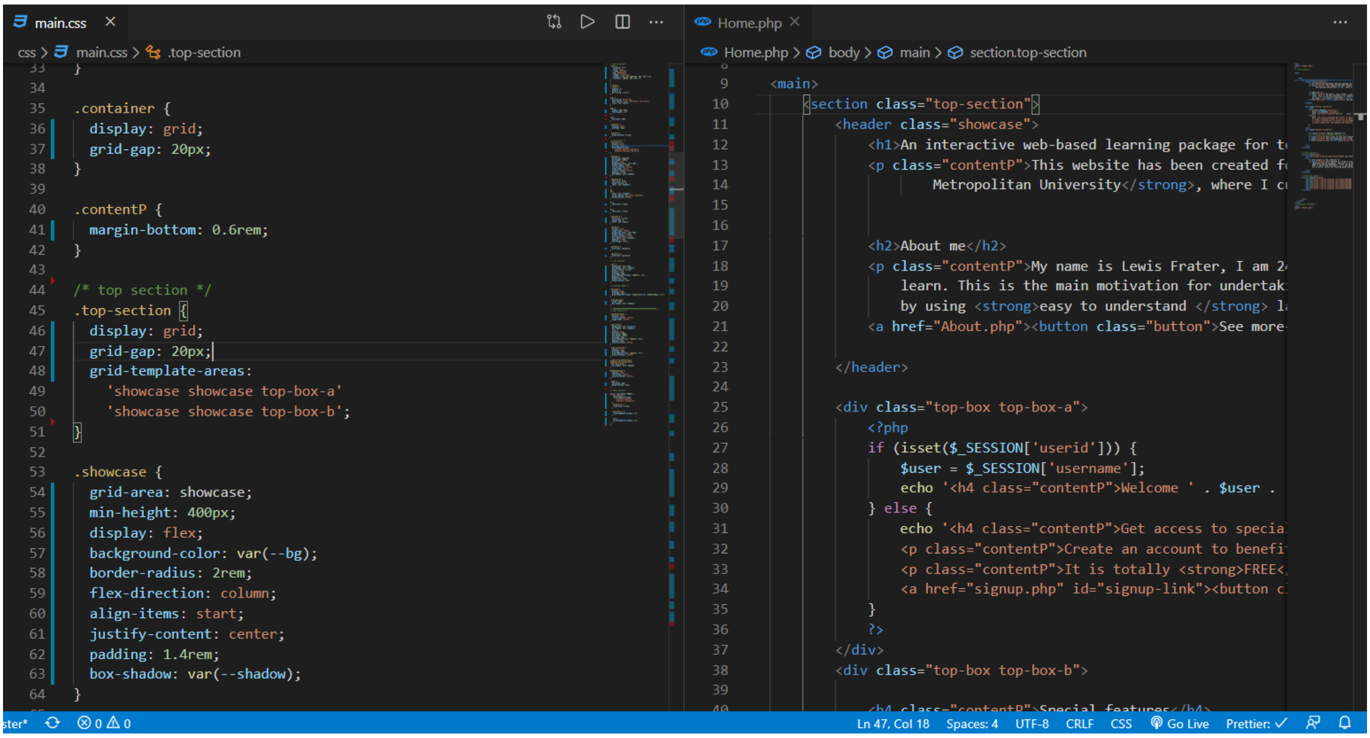
As you can see from figure 14, a session is created at the top of the file. This keeps track of whether there is a user logged in or not, this can be manipulated which will be discussed later in this chapter.

The navigation utilises the power of PHP. I wanted the users to experience the benefit of signing up by being able to use the HTML playground. This was done by an if statement. If a user is logged in, the playground link would be visible on the navigation bar, otherwise, invisible as seen in figure 14 at line 29 to 31.

4.3.2 Using CSS

CSS was used to style the web pages of this project. It enabled me to change colour schemes, adjust margins, use fonts and make the website responsive using media queries as well as much more.

CSS grid allowed for the easy implementation of a box style layout throughout the website. This is shown in the figures below.

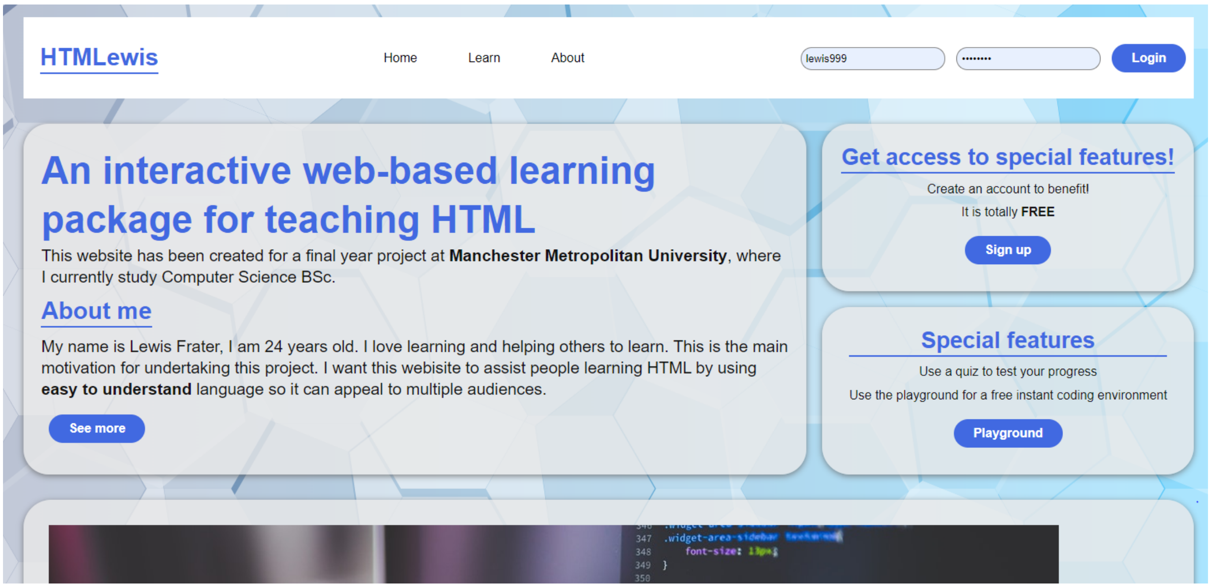


```css
33  }
34
35  .container {
36    display: grid;
37    grid-gap: 20px;
38  }
39
40  .contentP {
41    margin-bottom: 0.6rem;
42  }
43
44  /* top section */
45  .top-section {
46    display: grid;
47    grid-gap: 20px;
48    grid-template-areas:
49      'showcase showcase top-box-a'
50      'showcase showcase top-box-b';
51  }
52
53  .showcase {
54    grid-area: showcase;
55    min-height: 400px;
56    display: flex;
57    background-color: var(--bg);
58    border-radius: 2rem;
59    flex-direction: column;
60    align-items: start;
61    justify-content: center;
62    padding: 1.4rem;
63    box-shadow: var(--shadow);
64  }
```

```php
9       <main>
10          <section class="top-section">
11              <header class="showcase">
12                  <h1>An interactive web-based learning package for t
13                  <p class="contentP">This website has been created f
14                          Metropolitan University</strong>, where I c
15
16
17                  <h2>About me</h2>
18                  <p class="contentP">My name is Lewis Frater, I am 2
19                      learn. This is the main motivation for undertak
20                      by using <strong>easy to understand </strong> l
21                  <a href="About.php"><button class="button">See more
22
23              </header>
24
25              <div class="top-box top-box-a">
26                  <?php
27                  if (isset($_SESSION['userid'])) {
28                      $user = $_SESSION['username'];
29                      echo '<h4 class="contentP">Welcome ' . $user .
30                  } else {
31                      echo '<h4 class="contentP">Get access to specia
32                      <p class="contentP">Create an account to benefi
33                      <p class="contentP">It is totally <strong>FREE<
34                      <a href="signup.php" id="signup-link"><button c
35                  }
36                  ?>
37              </div>
38              <div class="top-box top-box-b">
39
40                  <h4 class="contentP">Special features</h4>
```

Figure 15: Showing the content of main.css and Home.php.

As you can see from figure 15, the top-section class was set to a display of grid. The grid template areas instructed the display to allow the showcase class to take up two thirds of the width as seen on lines 49 and 50. Letting top-box-a and top-box-b to only take up the remaining third. CSS grid is a powerful layout tool which helped style the webpages across this project. A figure of the result is below, laid out in the manner described in the CSS commands above.



HTMLewis

Home Learn About

lewis999 •••••••• Login

An interactive web-based learning package for teaching HTML

This website has been created for a final year project at **Manchester Metropolitan University**, where I currently study Computer Science BSc.

About me

My name is Lewis Frater, I am 24 years old. I love learning and helping others to learn. This is the main motivation for undertaking this project. I want this webiste to assist people learning HTML by using **easy to understand** language so it can appeal to multiple audiences.

See more

Get access to special features!

Create an account to benefit!

It is totally **FREE**

Sign up

Special features

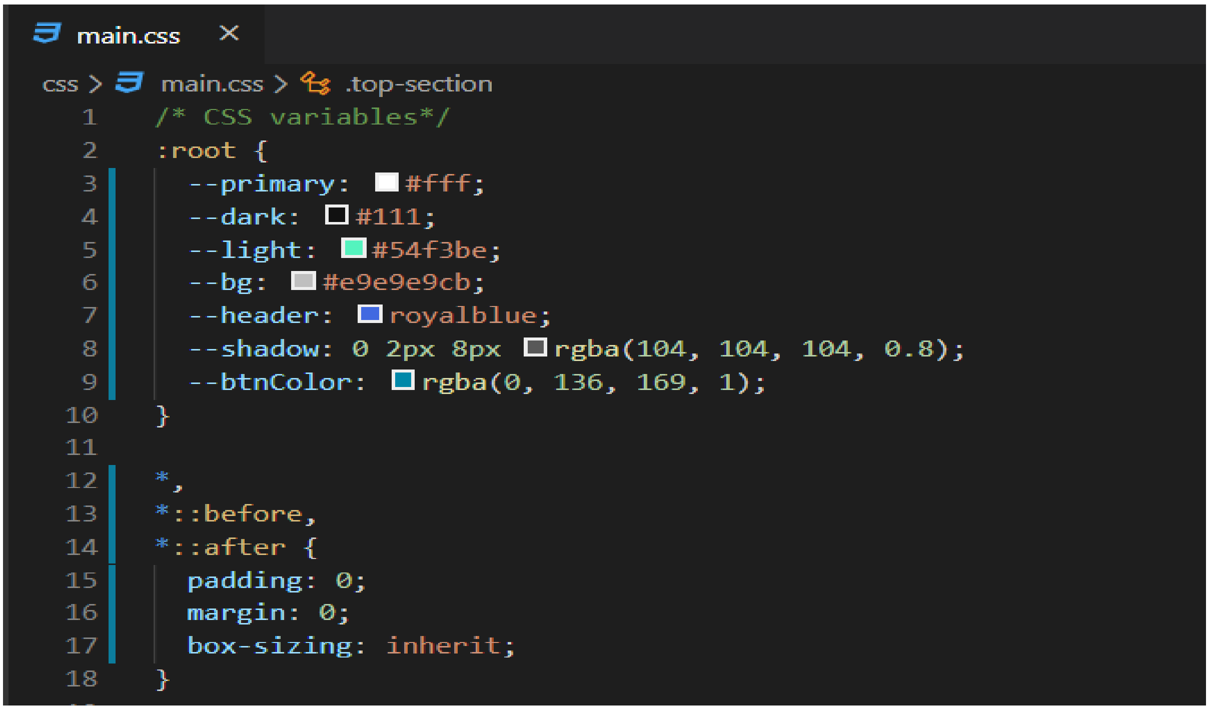
Use a quiz to test your progress

Use the playground for a free instant coding environment

Playground

Figure 16: Showing the layout of the boxes using CSS grid (Showcase box on the left, and the two top-boxes on the right).

In CSS, variables can be used to save frequently used colours. This was implemented in this project. It allowed me to change one variable to change every single instance of where that variable was used throughout the project if I felt a style changes was necessary. In a sense, it allowed to save the colour scheme of every webpage in just a few lines of code.



main.css ✕

css > main.css > .top-section

```css
/* CSS variables*/
:root {
    --primary: #fff;
    --dark: #111;
    --light: #54f3be;
    --bg: #e9e9e9cb;
    --header: royalblue;
    --shadow: 0 2px 8px rgba(104, 104, 104, 0.8);
    --btnColor: rgba(0, 136, 169, 1);
}

*,
*::before,
*::after {
    padding: 0;
    margin: 0;
    box-sizing: inherit;
}
```

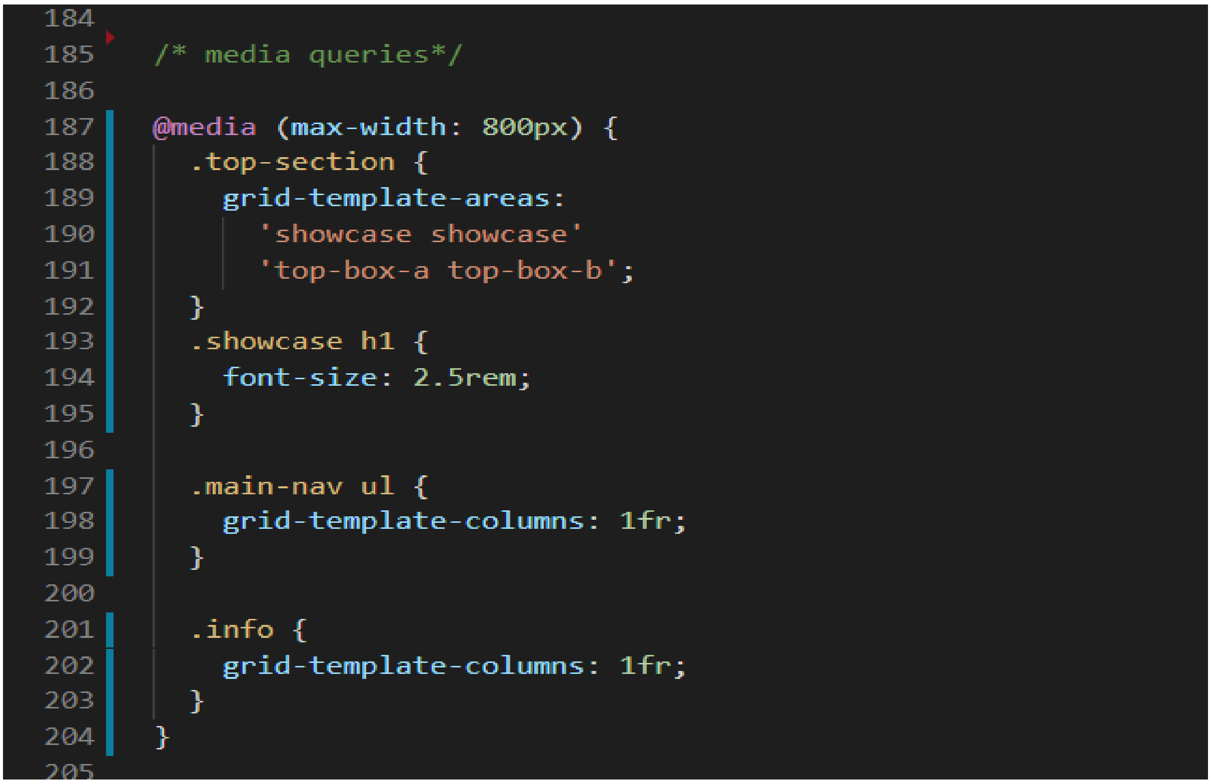
Figure 17: Showing the CSS root variables and the CSS reset below.

In the figure above, you can see the CSS reset from line 12. Some browsers put their own default padding and margins on HTML elements which can cause problems in positioning elements in the future. A CSS reset deletes any automatically implemented padding and margins on the different HTML elements. This helped in the design of the website as I decided to have everything default at 0, changing the padding and margins later in the code where needed to achieve the specified design of the webpages.

4.3.3 Responsive Design

In the present day, webpages are viewed by a variety of devices, all ranging in their display sizes. It is vital to incorporate these smaller screen sizes in the design of this project. To help adjust the styles for smaller screens, CSS media queries were used.

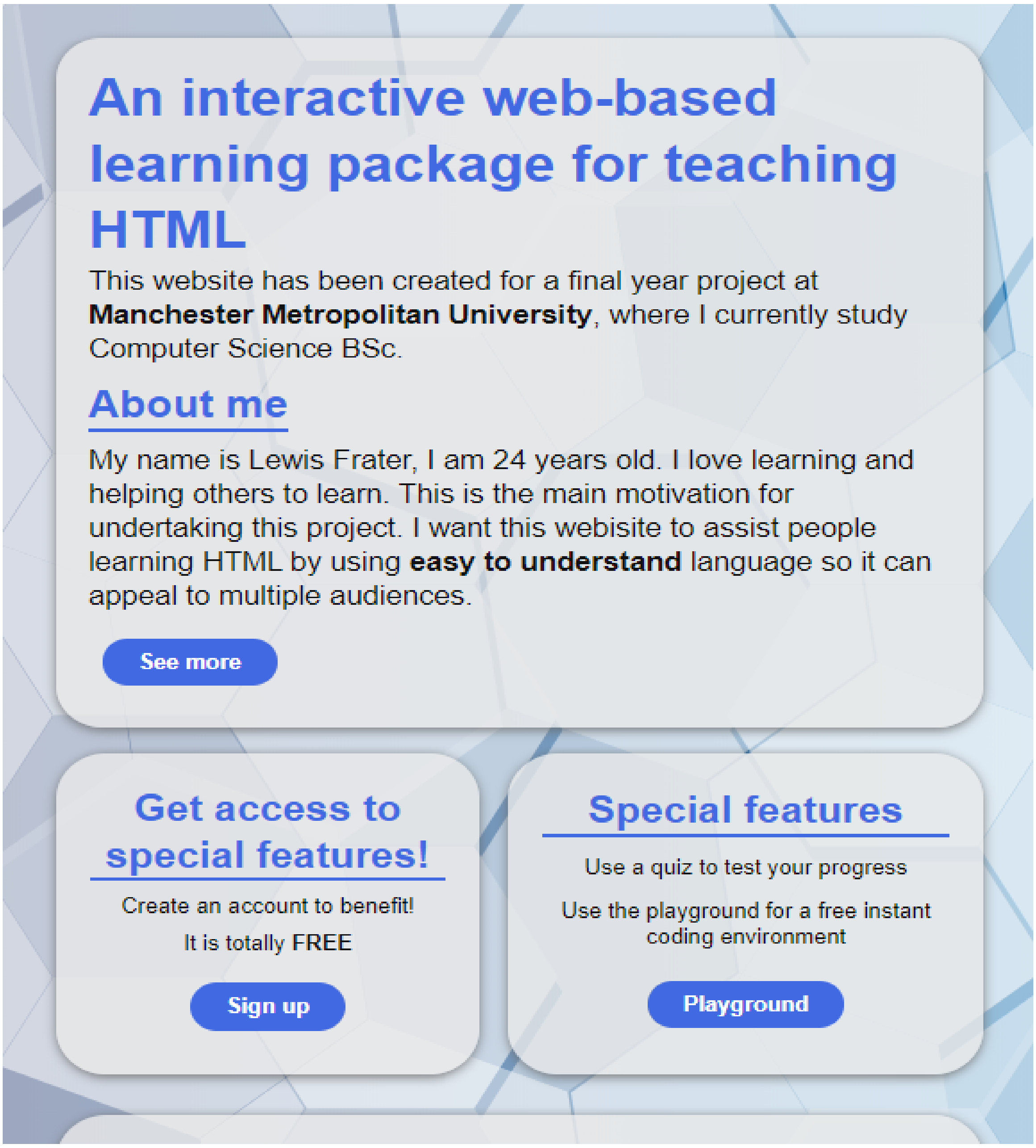
CSS media queries allow you to change the styling options depending on whether a particular condition is true. An example of a condition is if a screen size is smaller than 800 pixels, then execute these styling options. The figure below is an example of a media query I used for the home page.



```css

/* media queries*/

@media (max-width: 800px) {
  .top-section {
    grid-template-areas:
      'showcase showcase'
      'top-box-a top-box-b';
  }
  .showcase h1 {
    font-size: 2.5rem;
  }

  .main-nav ul {
    grid-template-columns: 1fr;
  }

  .info {
    grid-template-columns: 1fr;
  }
}

```

Figure 18: A media query in main.css.

The media query illustrated above implements these styling options if the screen has a max width of 800px. The HTML elements with “showcase” as their class would be above those with “top-box-a” and “top-box-b”. The result of this is shown below in figure 19.



An interactive web-based learning package for teaching HTML

This website has been created for a final year project at **Manchester Metropolitan University**, where I currently study Computer Science BSc.

About me

My name is Lewis Frater, I am 24 years old. I love learning and helping others to learn. This is the main motivation for undertaking this project. I want this webisite to assist people learning HTML by using **easy to understand** language so it can appeal to multiple audiences.

Get access to special features!

Create an account to benefit!

It is totally FREE

Special features

Use a quiz to test your progress

Use the playground for a free instant coding environment

Figure 19: Showing the homepage when width is less than 800 pixels.

As you can see the media query has changed the layout of the homepage compared to the one shown previously for larger devices in figure 16. Media queries are powerful and allow for responsive designs of websites. Making them accessible to the billions of devices with smaller screens across the world.

4.4 Back-end Implementation

In the back-end a registration and log in system had to be implemented. This was done using PHP and SQL queries. Database tables had to be created to hold all the relevant information.

4.4.1 Database handler and tables

In order to save the user information needed in this project, two database tables were created using phpMyAdmin.

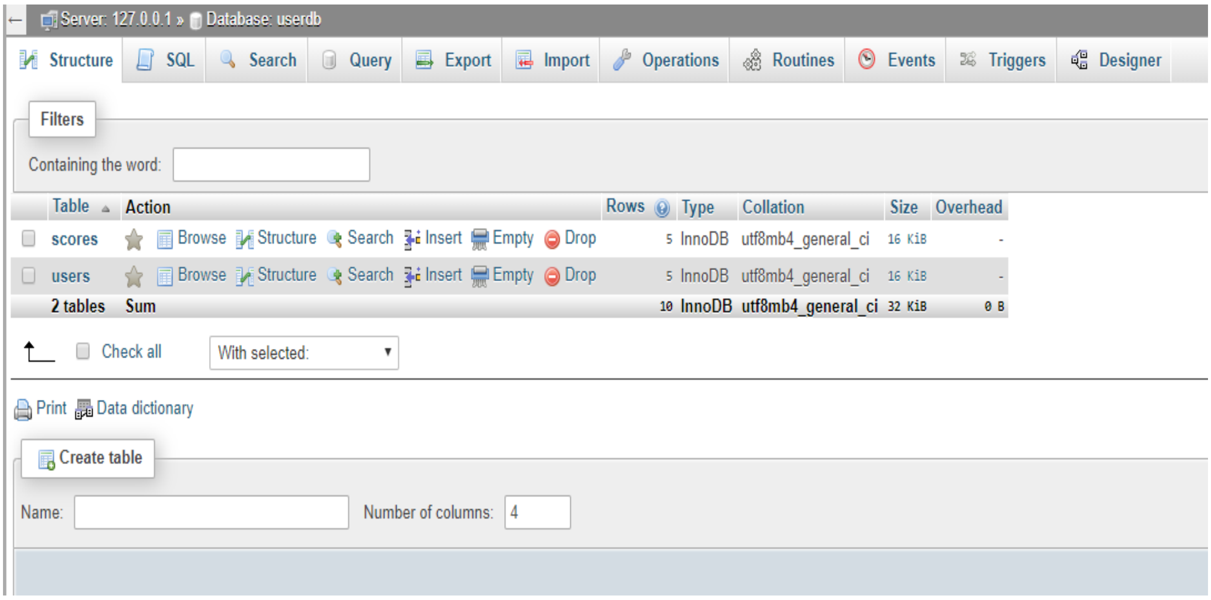
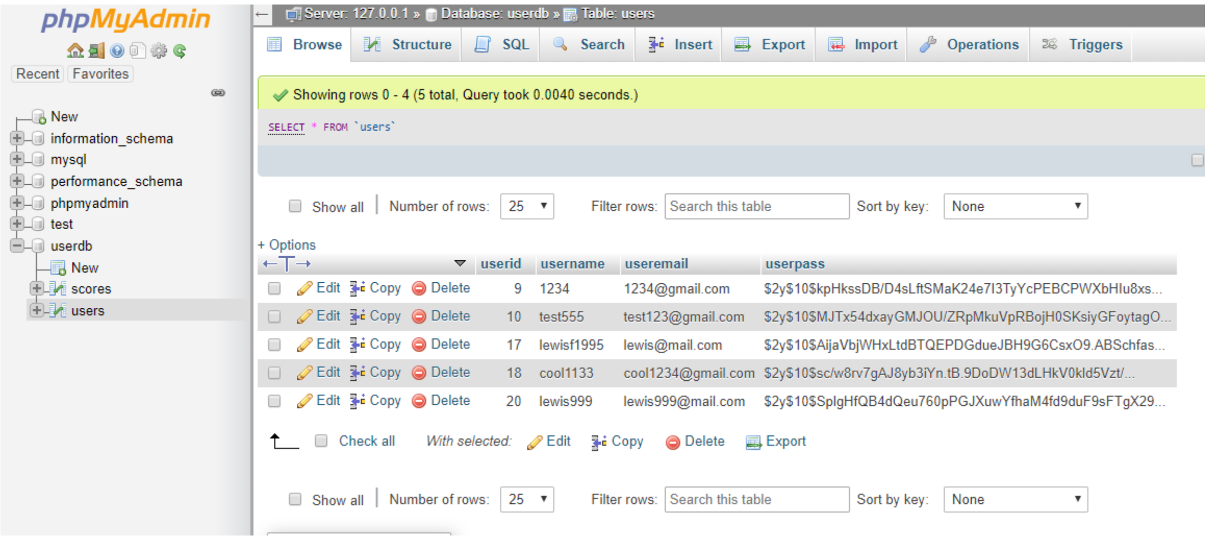


	Table	Action	Rows	Type	Collation	Size	Overhead
□	scores	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16 KiB	-
□	users	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16 KiB	-
	2 tables	Sum	10	InnoDB	utf8mb4_general_ci	32 KiB	0 B

Figure 20: The two tables in the database.

The user table had to contain information such as: User ID, Username, Email and Password (See figure 21 below).

The scores table contained information on the user's performance in each quiz (See figure 22 below).



Server: 127.0.0.1 » Database: userdb » Table: users

✓ Showing rows 0 - 4 (5 total, Query took 0.0040 seconds.)

```
SELECT * FROM `users`
```

□ Show all | Number of rows: 25 ▾ Filter rows: Search this table Sort by key: None ▾

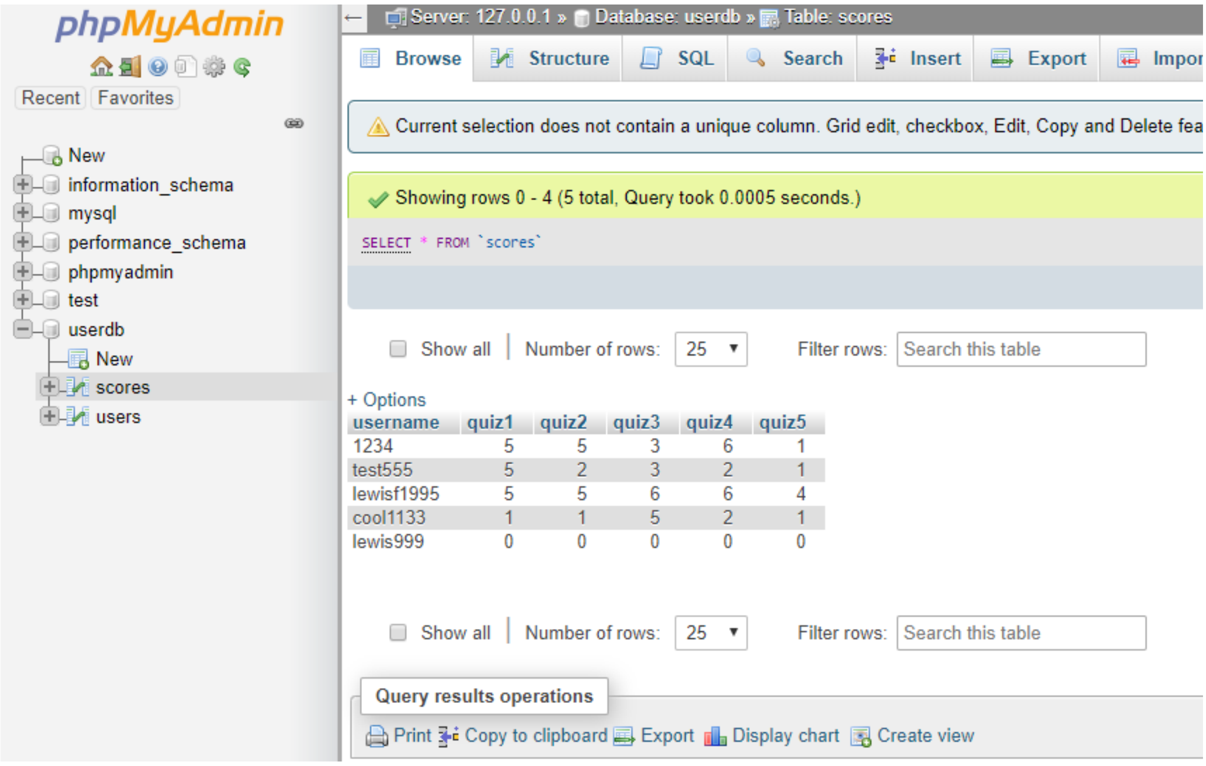
+ Options

	userid	username	useremail	userpass
□ Edit Copy Delete	9	1234	1234@gmail.com	$2y$10$kpHkssDB/D4sLftSMaK24e7I3TyYcPEBCPWXbHIu8xs...
□ Edit Copy Delete	10	test555	test123@gmail.com	$2y$10$MJTx54dxayGMJOU/ZRpMkuVpRBojH0SKsiyGFoytagO...
□ Edit Copy Delete	17	lewisf1995	lewis@mail.com	$2y$10$AijaVbjWHxLtdBTQEPDGdueJBH9G6CsxO9.ABSchfas...
□ Edit Copy Delete	18	cool1133	cool1234@gmail.com	$2y$10$sc/w8rv7gAJ8yb3iYn.tB.9DoDW13dLHkV0kld5Vzt/...
□ Edit Copy Delete	20	lewis999	lewis999@mail.com	$2y$10$SplgHfQB4dQeu760pPGJXuwYfhaM4fd9duF9sFTgX29...

□ Check all *With selected:* Edit Copy Delete Export

□ Show all | Number of rows: 25 ▾ Filter rows: Search this table Sort by key: None ▾

Figure 21: The users table.



Server: 127.0.0.1 » Database: userdb » Table: scores

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete fea

✓ Showing rows 0 - 4 (5 total, Query took 0.0005 seconds.)

```
SELECT * FROM `scores`
```

□ Show all | Number of rows: 25 ▾ Filter rows: Search this table

+ Options

username	quiz1	quiz2	quiz3	quiz4	quiz5
1234	5	5	3	6	1
test555	5	2	3	2	1
lewisf1995	5	5	6	6	4
cool1133	1	1	5	2	1
lewis999	0	0	0	0	0

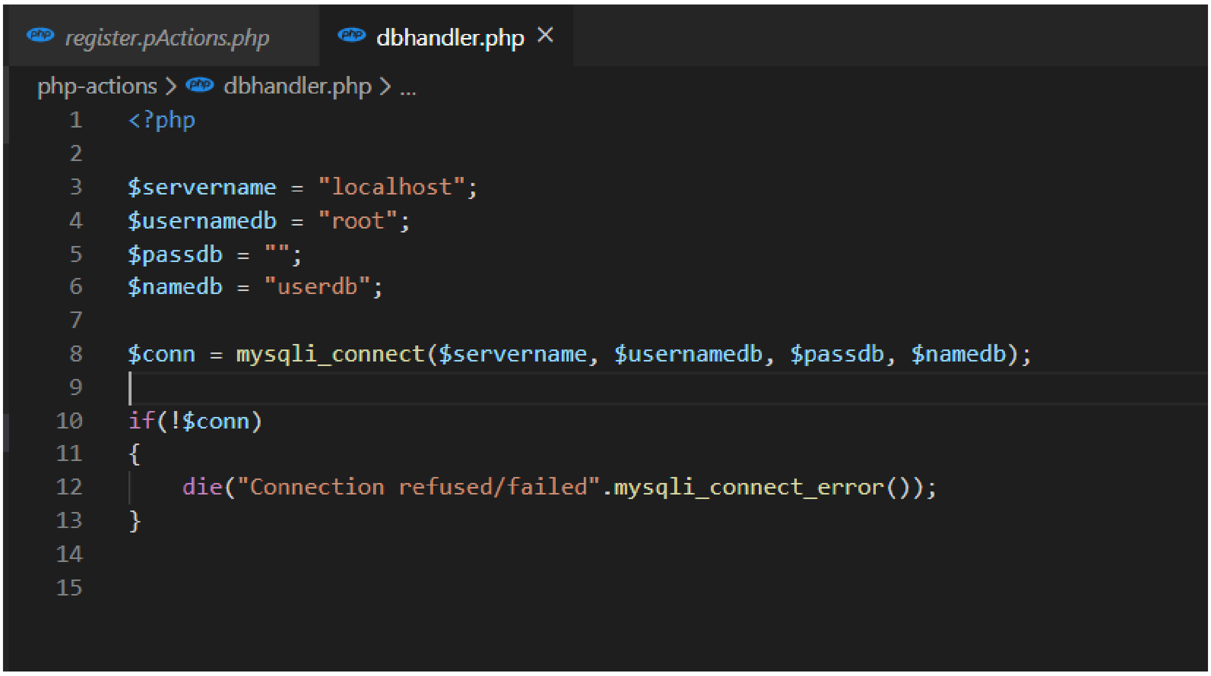
□ Show all | Number of rows: 25 ▾ Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view

Figure 22: The scores table holding each user’s score in each quiz.

The database handler initialises the connection to the database. This allows queries to be executed upon that database.



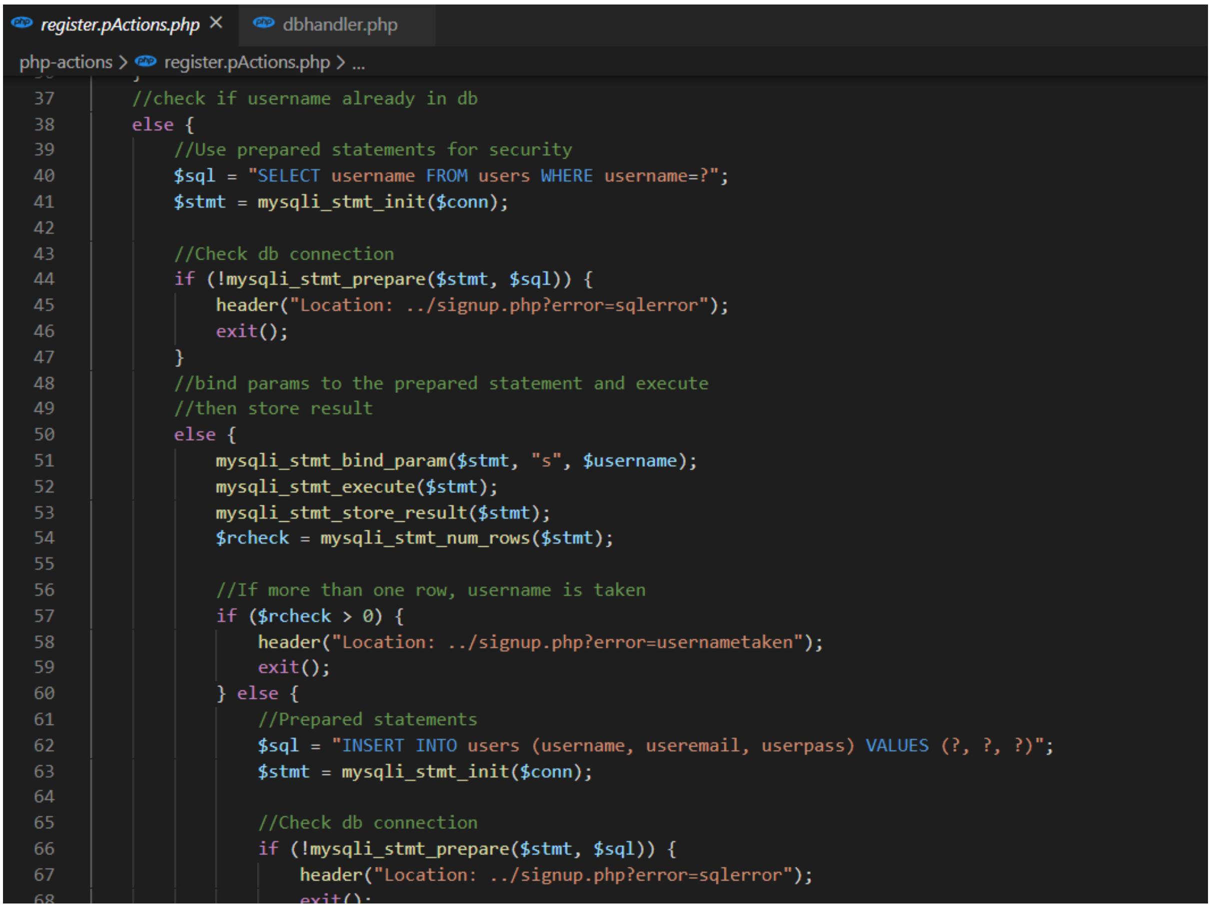
```php
<?php

$servername = "localhost";
$usernamedb = "root";
$passdb = "";
$namedb = "userdb";

$conn = mysqli_connect($servername, $usernamedb, $passdb, $namedb);

if(!$conn)
{
    die("Connection refused/failed".mysqli_connect_error());
}


```

Figure 23: dbhandler.php.

In the figure above, the database handler allows for the $conn variable to be used in other PHP files. It creates the connection to the database so it can be manipulated via SQL queries. This was the foundation every PHP file used to connection to the database on the server.

4.4.2 Registration and log in system

The registration system, along with the log in system followed a similar pattern. Using prepared statements for security, and SQL queries to insert or check data in the database tables.

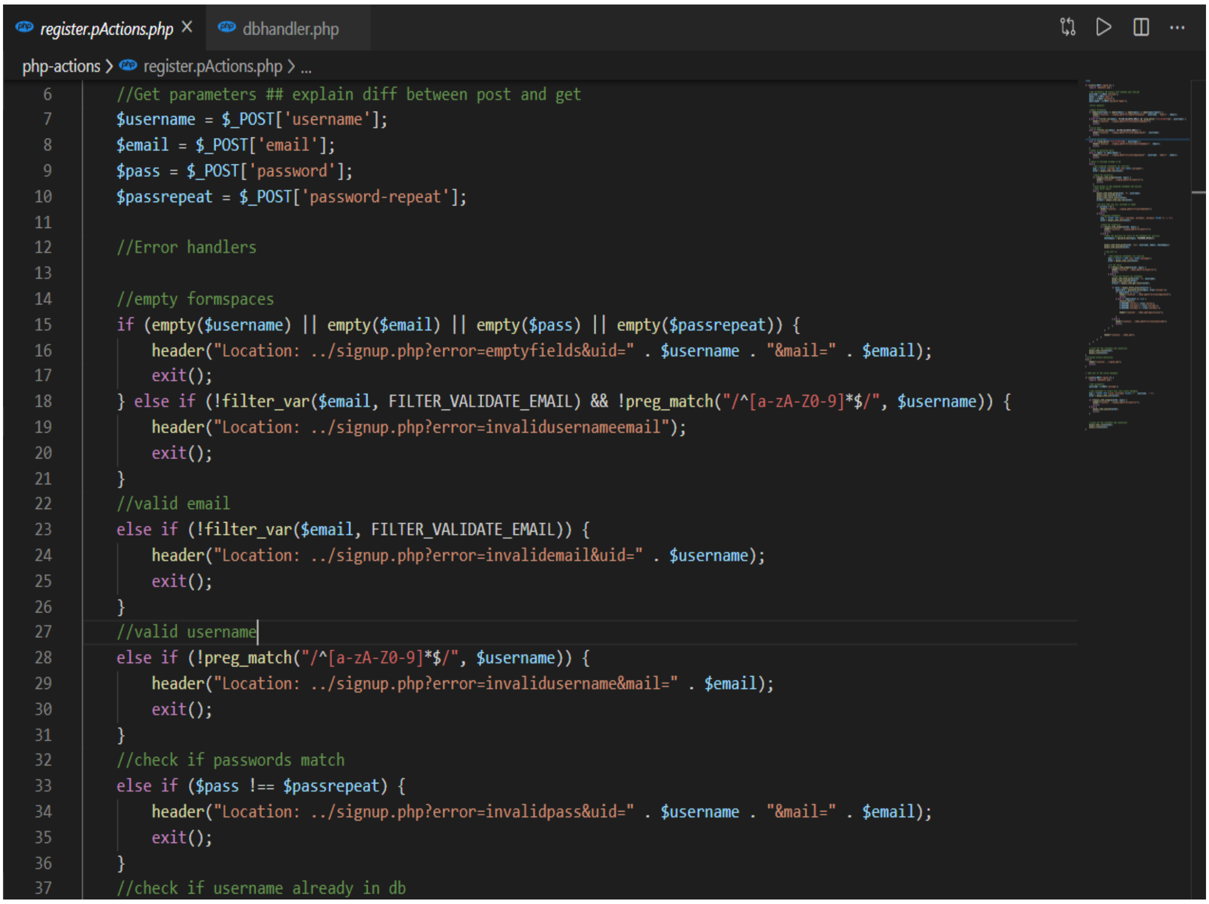


```
37      //check if username already in db
38      else {
39          //Use prepared statements for security
40          $sql = "SELECT username FROM users WHERE username=?";
41          $stmt = mysqli_stmt_init($conn);
42
43          //Check db connection
44          if (!mysqli_stmt_prepare($stmt, $sql)) {
45              header("Location: ../signup.php?error=sqlerror");
46              exit();
47          }
48          //bind params to the prepared statement and execute
49          //then store result
50          else {
51              mysqli_stmt_bind_param($stmt, "s", $username);
52              mysqli_stmt_execute($stmt);
53              mysqli_stmt_store_result($stmt);
54              $rcheck = mysqli_stmt_num_rows($stmt);
55
56              //If more than one row, username is taken
57              if ($rcheck > 0) {
58                  header("Location: ../signup.php?error=usernametaken");
59                  exit();
60              } else {
61                  //Prepared statements
62                  $sql = "INSERT INTO users (username, useremail, userpass) VALUES (?, ?, ?)";
63                  $stmt = mysqli_stmt_init($conn);
64
65                  //Check db connection
66                  if (!mysqli_stmt_prepare($stmt, $sql)) {
67                      header("Location: ../signup.php?error=sqlerror");
68                      exit();
```

Figure 24: Code from the register.pActions.php file.

Before registering, the database table is queried to see if the username is already taken. If one or more rows are returned, then the username is taken as seen by line 57 in the figure above. Otherwise a new set of prepared statements are executed to insert that user and their information in the user table shown on line 62.

Error handlers are used before the information is queried on. This is to ensure the information the user has inputted is correct and in the right format.

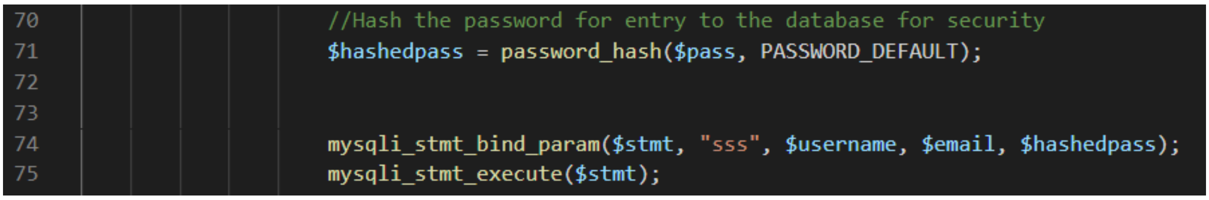


```php
//Get parameters ## explain diff between post and get
$username = $_POST['username'];
$email = $_POST['email'];
$pass = $_POST['password'];
$passrepeat = $_POST['password-repeat'];

//Error handlers

//empty formspaces
if (empty($username) || empty($email) || empty($pass) || empty($passrepeat)) {
    header("Location: ../signup.php?error=emptyfields&uid=" . $username . "&mail=" . $email);
    exit();
} else if (!filter_var($email, FILTER_VALIDATE_EMAIL) && !preg_match("/^[a-zA-Z0-9]*$/", $username)) {
    header("Location: ../signup.php?error=invalidusernameemail");
    exit();
}
//valid email
else if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
    header("Location: ../signup.php?error=invalidemail&uid=" . $username);
    exit();
}
//valid username
else if (!preg_match("/^[a-zA-Z0-9]*$/", $username)) {
    header("Location: ../signup.php?error=invalidusername&mail=" . $email);
    exit();
}
//check if passwords match
else if ($pass !== $passrepeat) {
    header("Location: ../signup.php?error=invalidpass&uid=" . $username . "&mail=" . $email);
    exit();
}
//check if username already in db
```

Figure 25: The error handlers before the information is processed.

As you can see in figure 25, the error handlers include: empty form spaces, valid email, valid username as well as checking if the password and repeat password form inputs are identical. Error handling helps keep a web application robust and avoids any unnecessary flawed data being inserted into the database tables.

The passwords the user chooses must be hashed to provide security in the unlikely event that the database tables were compromised. This is shown in the figure below.

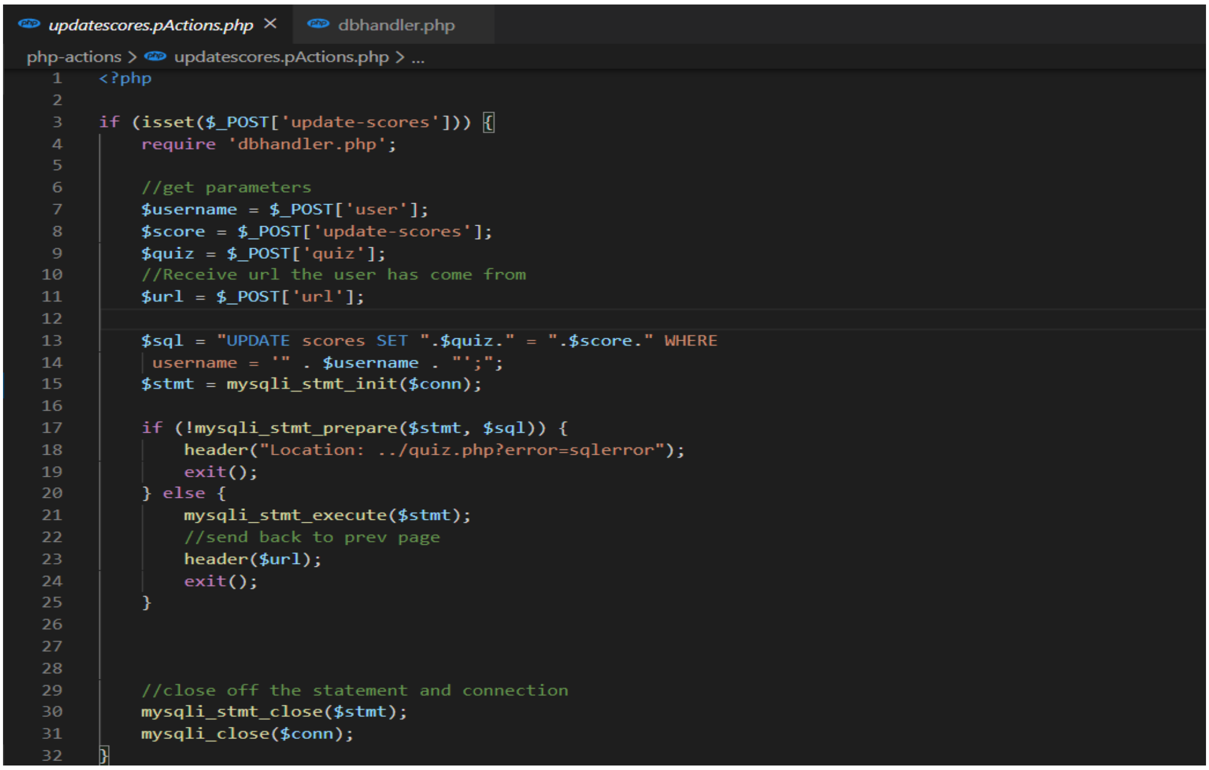


```php
//Hash the password for entry to the database for security
$hashedpass = password_hash($pass, PASSWORD_DEFAULT);


mysqli_stmt_bind_param($stmt, "sss", $username, $email, $hashedpass);
mysqli_stmt_execute($stmt);
```

Figure 26: Code excerpt from register.pActions.php showing the password being hashed using PHP’s default password hasher.

4.4.3 Tracking Quiz Scores

Another feature of the back-end was to track and store the quiz scores of each user. This was done to provide a global leaderboard for the users to compare themselves with each other. This helps give the quizzes a competitive edge which helps a user to improve themselves.


updatescores.pActions.php ×
dbhandler.php
php-actions > updatescores.pActions.php > ...

```
 1   <?php
 2
 3   if (isset($_POST['update-scores'])) {
 4       require 'dbhandler.php';
 5
 6       //get parameters
 7       $username = $_POST['user'];
 8       $score = $_POST['update-scores'];
 9       $quiz = $_POST['quiz'];
10       //Receive url the user has come from
11       $url = $_POST['url'];
12
13       $sql = "UPDATE scores SET ".$quiz." = ".$score." WHERE
14         username = '" . $username . "';";
15       $stmt = mysqli_stmt_init($conn);
16
17       if (!mysqli_stmt_prepare($stmt, $sql)) {
18           header("Location: ../quiz.php?error=sqlerror");
19           exit();
20       } else {
21           mysqli_stmt_execute($stmt);
22           //send back to prev page
23           header($url);
24           exit();
25       }
26
27
28
29       //close off the statement and connection
30       mysqli_stmt_close($stmt);
31       mysqli_close($conn);
32   }
```

Figure 27: The code used to update the scores of each user.

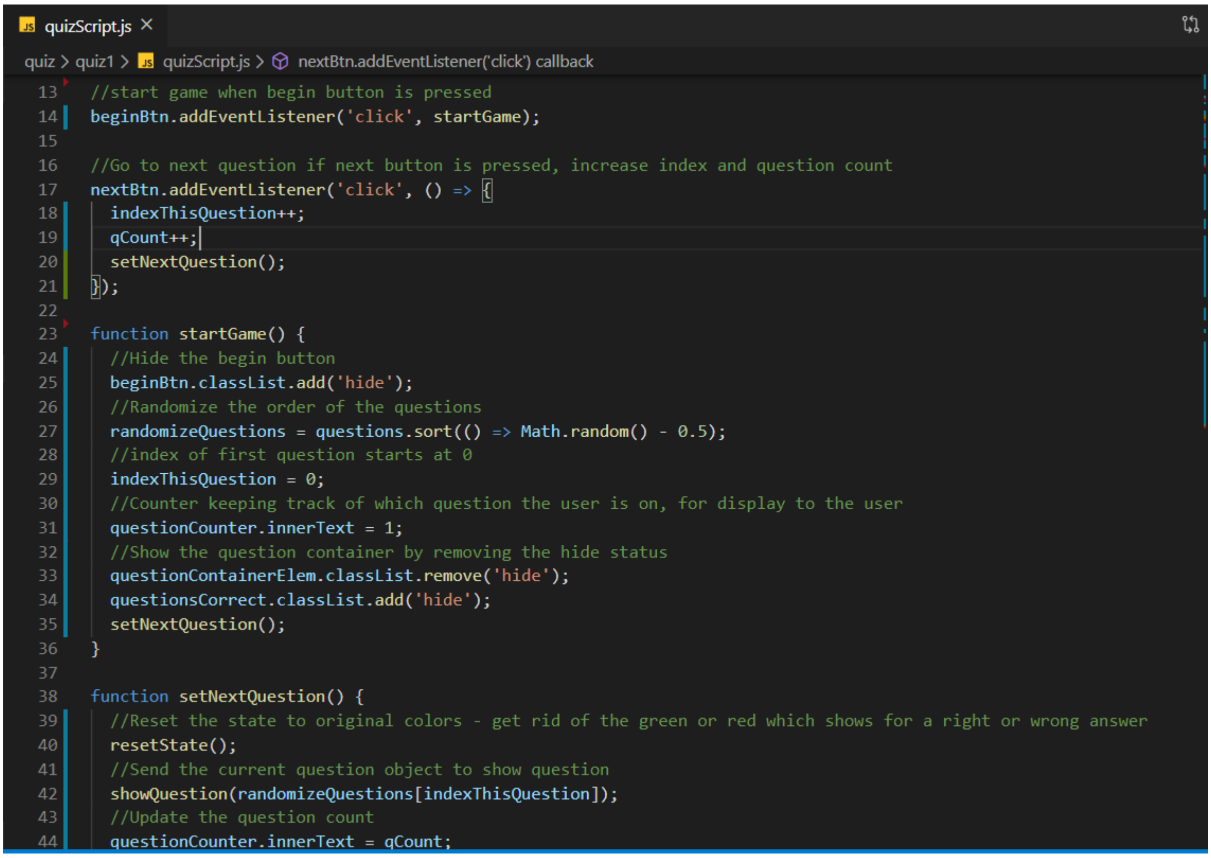
As seen in the figure 27, the parameters are retrieved as well as the URL the user came from. The parameters are used to create the SQL statement which updates the score the user got in that particular quiz (line 13). The statement is executed on line 21 if the statement is prepared correctly (line 17).

If the statement is executed successfully, the user is redirected back to the previous page (line 23).

As said previously, the log in system follows a very similar pattern to the registration system. It won’t be included to avoid repetition.

4.5 Creating the Quiz

A quiz was created to provide the interactivity needed to help the user learn the new information they were absorbing. It provided a way to test the user on the concepts learnt. The engine of the quiz was created using JavaScript.

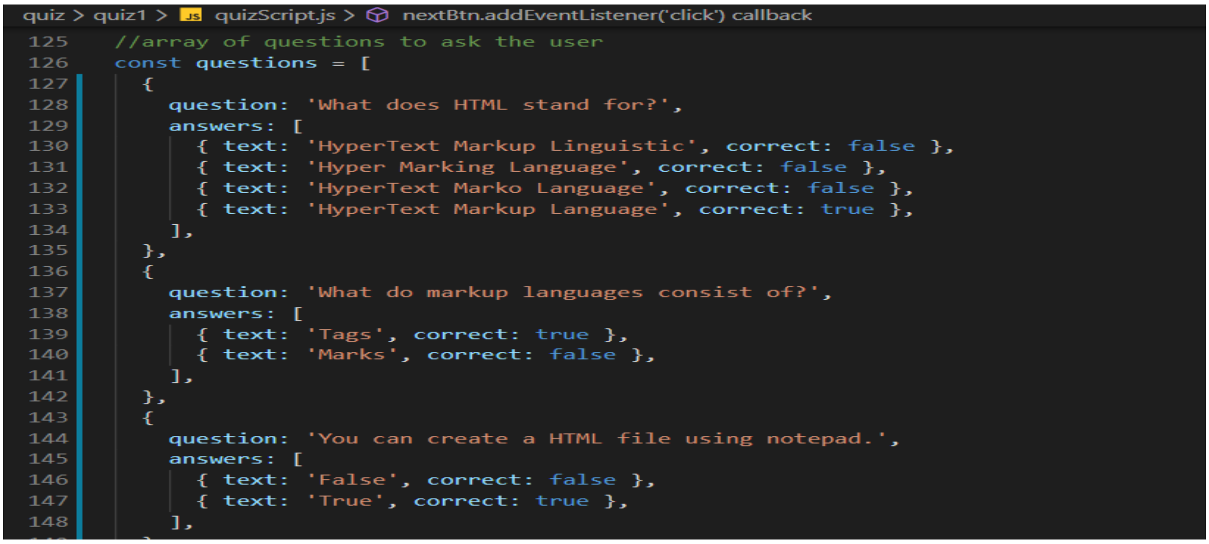

quizScript.js ×
quiz > quiz1 > quizScript.js > nextBtn.addEventListener('click') callback

```
//start game when begin button is pressed
beginBtn.addEventListener('click', startGame);

//Go to next question if next button is pressed, increase index and question count
nextBtn.addEventListener('click', () => {
  indexThisQuestion++;
  qCount++;
  setNextQuestion();
});

function startGame() {
  //Hide the begin button
  beginBtn.classList.add('hide');
  //Randomize the order of the questions
  randomizeQuestions = questions.sort(() => Math.random() - 0.5);
  //index of first question starts at 0
  indexThisQuestion = 0;
  //Counter keeping track of which question the user is on, for display to the user
  questionCounter.innerText = 1;
  //Show the question container by removing the hide status
  questionContainerElem.classList.remove('hide');
  questionsCorrect.classList.add('hide');
  setNextQuestion();
}

function setNextQuestion() {
  //Reset the state to original colors - get rid of the green or red which shows for a right or wrong answer
  resetState();
  //Send the current question object to show question
  showQuestion(randomizeQuestions[indexThisQuestion]);
  //Update the question count
  questionCounter.innerText = qCount;
```

Figure 28: Excerpt from the quizScript.js file showing the startGame and setNextQuestion functions.

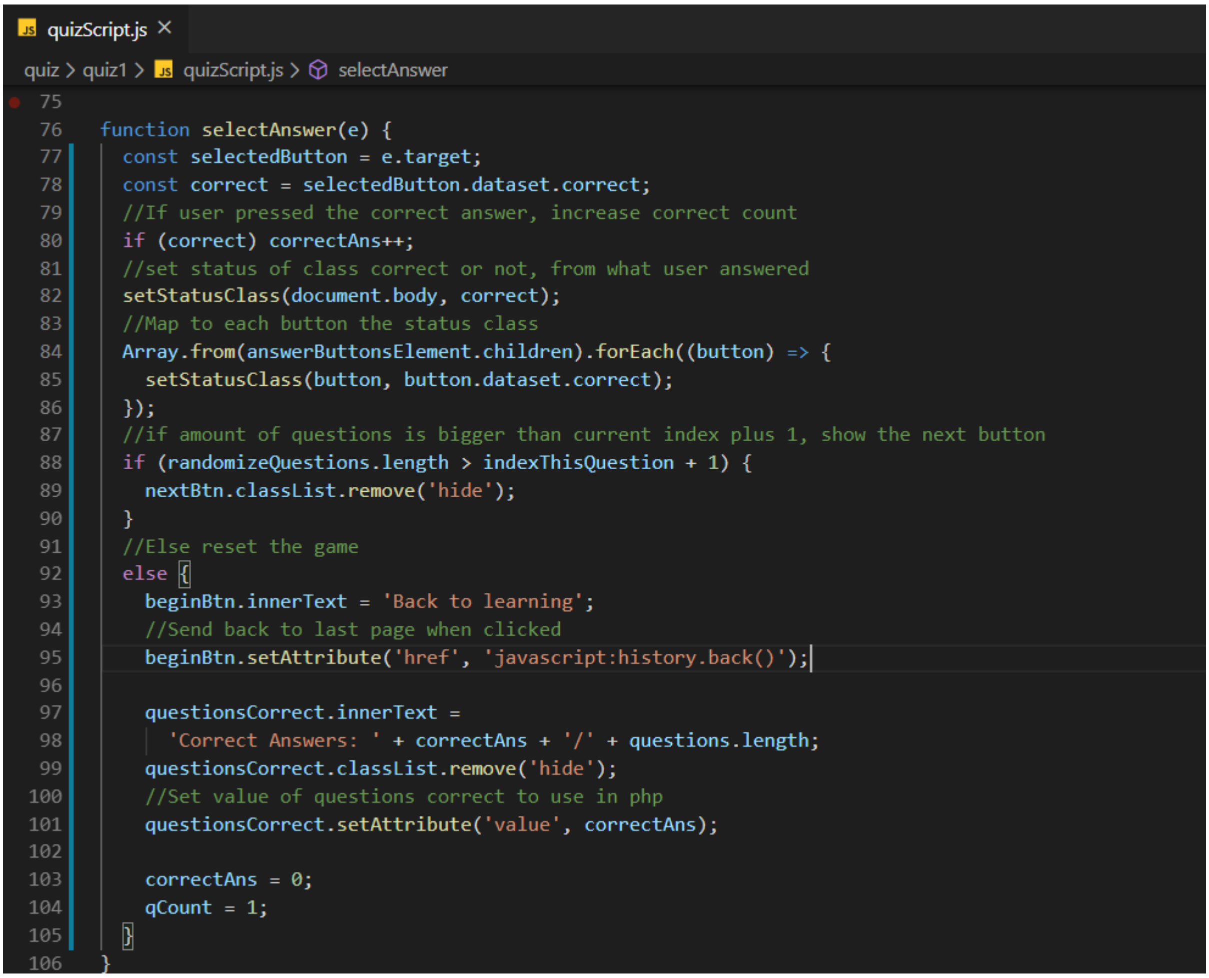
The code starts with two event listeners which check if a button has been pressed on the quiz. When the begin button is pressed the startGame function is executed.

The startGame function initially hides the begin button from the user then randomizes the order of the questions given which are stored in an array shown in figure 29 below. A question counter keeps track of which question a user in on, so it can be displayed above the quiz question. The setNextQuestion function resets the state when the next question is loaded while loading the next question to be asked. This is to remove the red or green feedback the user receives on screen upon the correct/incorrect answering of a question.


quiz > quiz1 > quizScript.js > nextBtn.addEventListener('click') callback

```
//array of questions to ask the user
const questions = [
  {
    question: 'What does HTML stand for?',
    answers: [
      { text: 'HyperText Markup Linguistic', correct: false },
      { text: 'Hyper Marking Language', correct: false },
      { text: 'HyperText Marko Language', correct: false },
      { text: 'HyperText Markup Language', correct: true },
    ],
  },
  {
    question: 'What do markup languages consist of?',
    answers: [
      { text: 'Tags', correct: true },
      { text: 'Marks', correct: false },
    ],
  },
  {
    question: 'You can create a HTML file using notepad.',
    answers: [
      { text: 'False', correct: false },
      { text: 'True', correct: true },
    ],
```

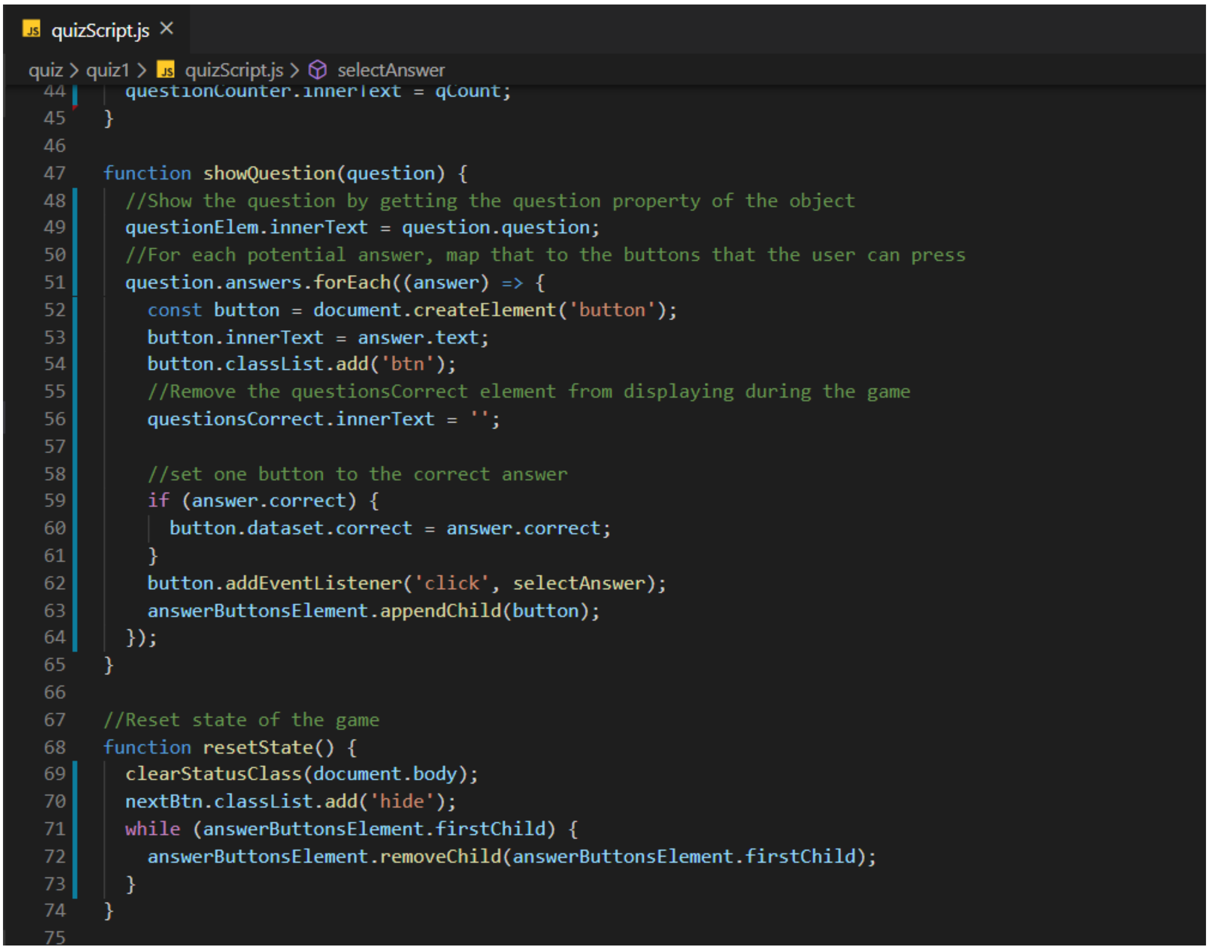
Figure 29: Example of an array of questions in a quiz.



```

function selectAnswer(e) {
  const selectedButton = e.target;
  const correct = selectedButton.dataset.correct;
  //If user pressed the correct answer, increase correct count
  if (correct) correctAns++;
  //set status of class correct or not, from what user answered
  setStatusClass(document.body, correct);
  //Map to each button the status class
  Array.from(answerButtonsElement.children).forEach((button) => {
    setStatusClass(button, button.dataset.correct);
  });
  //if amount of questions is bigger than current index plus 1, show the next button
  if (randomizeQuestions.length > indexThisQuestion + 1) {
    nextBtn.classList.remove('hide');
  }
  //Else reset the game
  else {
    beginBtn.innerText = 'Back to learning';
    //Send back to last page when clicked
    beginBtn.setAttribute('href', 'javascript:history.back()');

    questionsCorrect.innerText =
      'Correct Answers: ' + correctAns + '/' + questions.length;
    questionsCorrect.classList.remove('hide');
    //Set value of questions correct to use in php
    questionsCorrect.setAttribute('value', correctAns);

    correctAns = 0;
    qCount = 1;
  }
}
```

Figure 30: The selectAnswer function in quizScript.js

This function determines whether the answer selected by the user is correct and if there is a following question or not. If a question is answered correctly then the amount of correct answers variable is incremented by one, this is vital as it is used to send the score of the quiz to the database. If there is no next question to be asked, then the state of the game is reset and the button to continue has a path referenced to the user’s previous page.

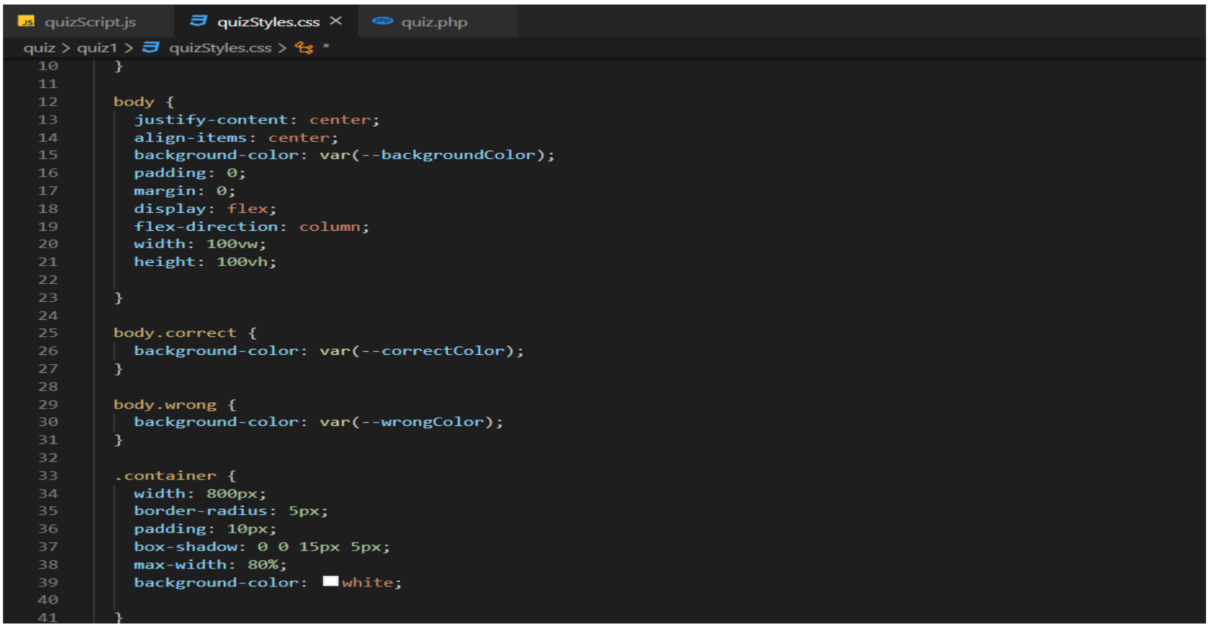


```javascript
    questionCounter.innerText = qCount;
  }

  function showQuestion(question) {
    //Show the question by getting the question property of the object
    questionElem.innerText = question.question;
    //For each potential answer, map that to the buttons that the user can press
    question.answers.forEach((answer) => {
      const button = document.createElement('button');
      button.innerText = answer.text;
      button.classList.add('btn');
      //Remove the questionsCorrect element from displaying during the game
      questionsCorrect.innerText = '';

      //set one button to the correct answer
      if (answer.correct) {
        button.dataset.correct = answer.correct;
      }
      button.addEventListener('click', selectAnswer);
      answerButtonsElement.appendChild(button);
    });
  }

  //Reset state of the game
  function resetState() {
    clearStatusClass(document.body);
    nextBtn.classList.add('hide');
    while (answerButtonsElement.firstChild) {
      answerButtonsElement.removeChild(answerButtonsElement.firstChild);
    }
  }
```

Figure 31: Excerpt from quizScript.js showing the showQuestion function.

The showQuestion function handles what is displayed to the user on screen. It uses the DOM (Document Object Model) of an HTML document to manipulate the data displayed. It allows the questions and answers to be displayed on screen, creating the buttons to be pressed which act as the interactive interface for the user. Once an answer is clicked, an event listener picks this up and runs the selectAnswer function previously discussed.

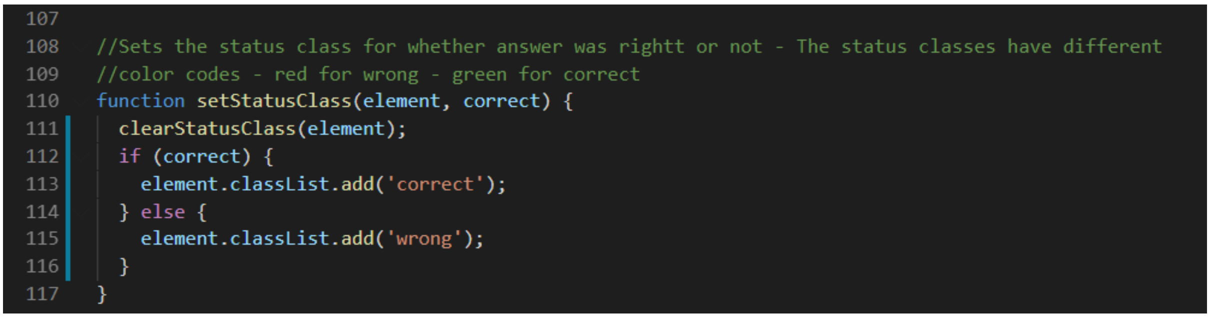
The quiz was styled using CSS. The layout had to be simplistic and provide feedback for correct and wrong answers. This was done by changing the background colour of the quiz as seen below in figure 32 at lines 26 and 30.



```
10      }
11  
12      body {
13          justify-content: center;
14          align-items: center;
15          background-color: var(--backgroundColor);
16          padding: 0;
17          margin: 0;
18          display: flex;
19          flex-direction: column;
20          width: 100vw;
21          height: 100vh;
22  
23      }
24  
25      body.correct {
26          background-color: var(--correctColor);
27      }
28  
29      body.wrong {
30          background-color: var(--wrongColor);
31      }
32  
33      .container {
34          width: 800px;
35          border-radius: 5px;
36          padding: 10px;
37          box-shadow: 0 0 15px 5px;
38          max-width: 80%;
39          background-color: white;
40  
41      }
```

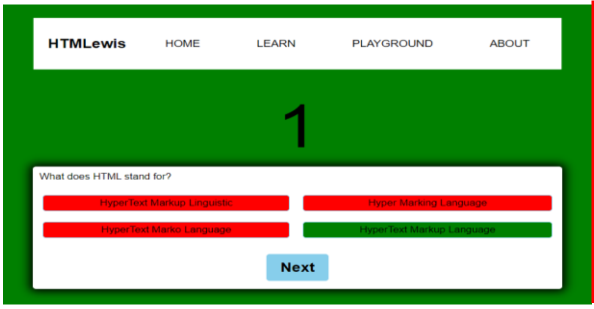
Figure 32: Excerpt from quizStyles.css showing the CSS of the quiz.

The feedback was implemented via the JavaScript script in the quiz itself. Making use of the status class. This is shown in figure 33 below.

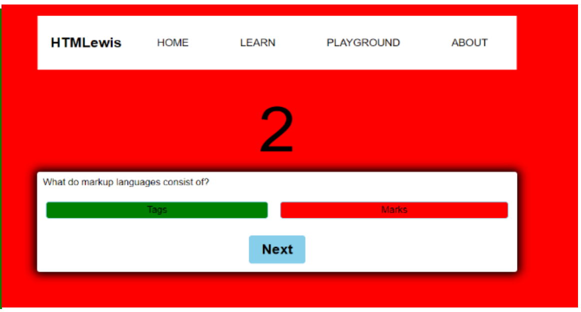


```

//Sets the status class for whether answer was rightt or not - The status classes have different
//color codes - red for wrong - green for correct
function setStatusClass(element, correct) {
  clearStatusClass(element);
  if (correct) {
    element.classList.add('correct');
  } else {
    element.classList.add('wrong');
  }
}
```

Figure 33: Functionality changing the status class for the background.



What does HTML stand for?



What do markup languages consist of?

Figure 34: Correct feedback.

Figure 35: Wrong feedback.

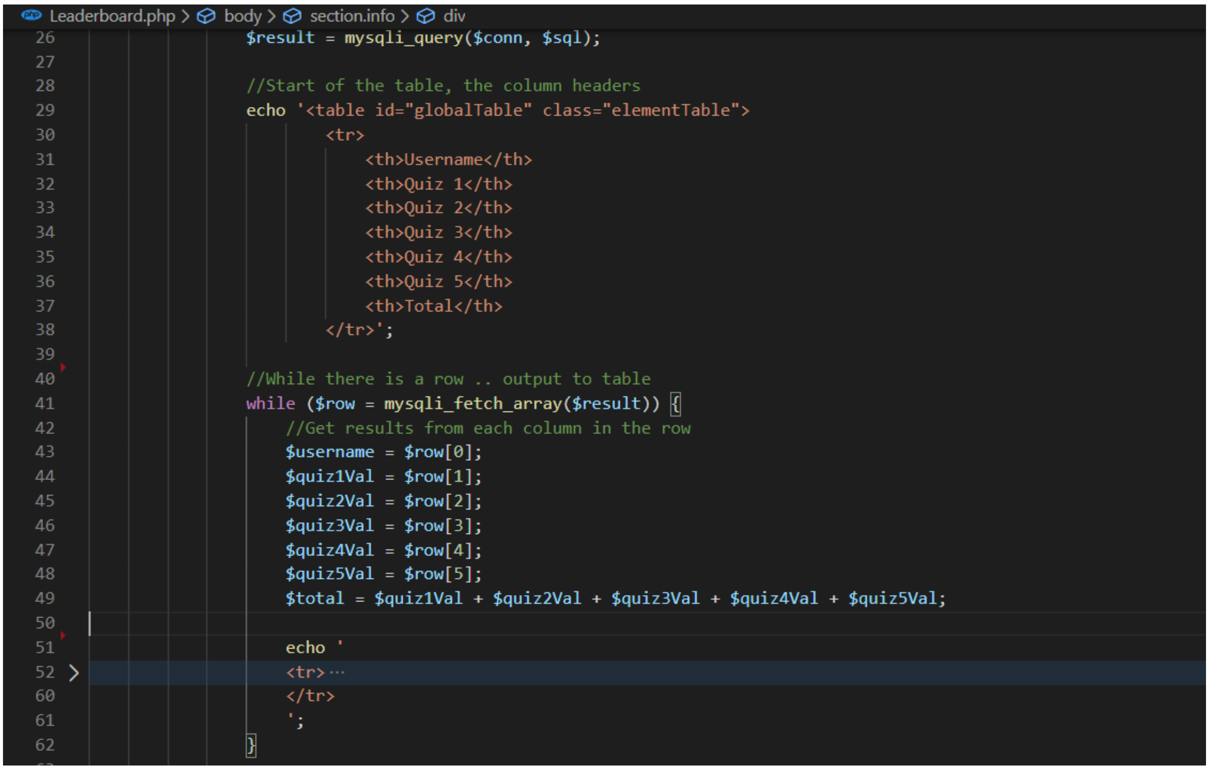
5 Product Evaluation

5.1 Evaluation Overview

This section of the report will describe some of the problems encountered during the development process. Most of the problems stemmed from the lack of knowledge of PHP and SQL beforehand. This meant that many issues were corrected while I was writing the code, which required further research to solve them. Below are some of the main problems I encountered.

5.1.1 The global leaderboard

The global leader board was difficult to implement. It involved retrieving every single quiz score for each different user in the database. Initially the leaderboard wasn't a priority so I focussed on other aspects of the web package. However, upon further research I eventually implemented the global leaderboard, albeit not very efficiently.



```
                $result = mysqli_query($conn, $sql);

                //Start of the table, the column headers
                echo '<table id="globalTable" class="elementTable">
                        <tr>
                            <th>Username</th>
                            <th>Quiz 1</th>
                            <th>Quiz 2</th>
                            <th>Quiz 3</th>
                            <th>Quiz 4</th>
                            <th>Quiz 5</th>
                            <th>Total</th>
                        </tr>';

                //While there is a row .. output to table
                while ($row = mysqli_fetch_array($result)) {
                    //Get results from each column in the row
                    $username = $row[0];
                    $quiz1Val = $row[1];
                    $quiz2Val = $row[2];
                    $quiz3Val = $row[3];
                    $quiz4Val = $row[4];
                    $quiz5Val = $row[5];
                    $total = $quiz1Val + $quiz2Val + $quiz3Val + $quiz4Val + $quiz5Val;

                    echo '
                    <tr>···
                    </tr>
                    ';
                }
```

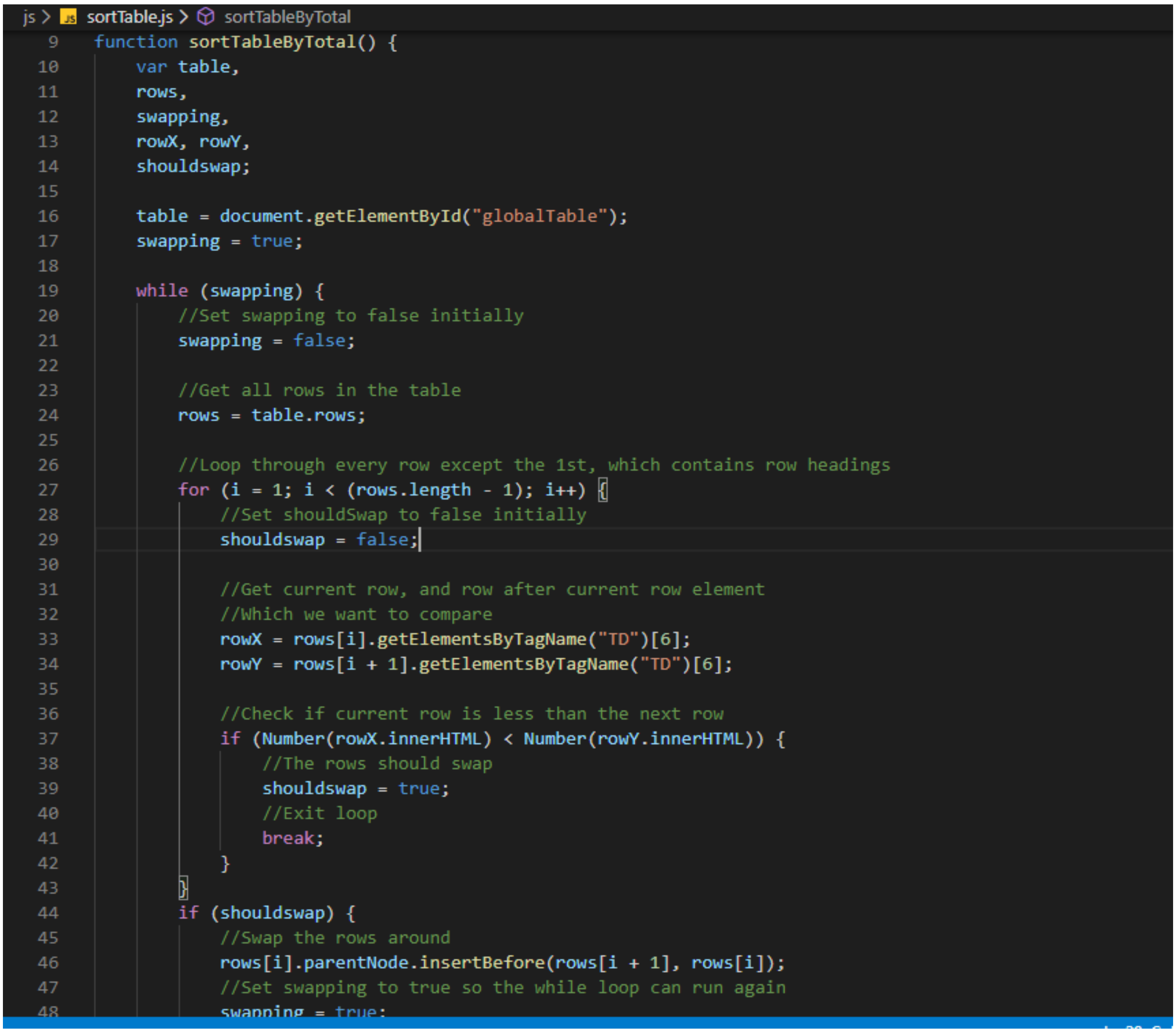
Figure 36: The global leaderboard.

The vital function that I hadn't known about was "mysqli_fetch_array", this was of utmost importance to fetch every row and column needed to output the information in a table. As you can see from line 43 to 49, it doesn’t seem to be very efficient and I’m still sure there is an alternate way to do this. Despite this, it worked and after it had caused me so many problems, I was satisfied with the code.

5.1.2 Sorting the leaderboard in descending order

Another aspect of the global leaderboard which also caused a lot of problems was sorting the table by quiz score total. After much misery with using SQL to sort the table before being output, I could not implement this using SQL.

The only remaining way to manipulate the table to my knowledge was JavaScript. I created a JavaScript file named sortTable.js shown in figure 37 below.



```
function sortTableByTotal() {
    var table,
    rows,
    swapping,
    rowX, rowY,
    shouldswap;

    table = document.getElementById("globalTable");
    swapping = true;

    while (swapping) {
        //Set swapping to false initially
        swapping = false;

        //Get all rows in the table
        rows = table.rows;

        //Loop through every row except the 1st, which contains row headings
        for (i = 1; i < (rows.length - 1); i++) {
            //Set shouldSwap to false initially
            shouldswap = false;

            //Get current row, and row after current row element
            //Which we want to compare
            rowX = rows[i].getElementsByTagName("TD")[6];
            rowY = rows[i + 1].getElementsByTagName("TD")[6];

            //Check if current row is less than the next row
            if (Number(rowX.innerHTML) < Number(rowY.innerHTML)) {
                //The rows should swap
                shouldswap = true;
                //Exit loop
                break;
            }
        }
        if (shouldswap) {
            //Swap the rows around
            rows[i].parentNode.insertBefore(rows[i + 1], rows[i]);
            //Set swapping to true so the while loop can run again
            swapping = true;
```

Figure 37: The sort table by total function used to manipulate the table.

As you can see in figure 37, a form of bubble sort was implemented to solve the problem. It involved comparing the score in one row to the adjacent row. If the first row had a smaller score than the next, then they will swap as indicated by lines 37 to 41 and 44 to 48. This process would continue until all the rows were in the correct place, in descending order.

Although there is a way to do this using SQL, I couldn’t implement it. I believe I am competent using JavaScript, so it was shown to be very useful to solve this problem.

6 Conclusion

6.1 Project Review

The objective of this project was to create an interactive web package which would teach HTML to beginners aged 16 to 18. The web package was created, and the objective was achieved.

6.2 Project Constraints

The constraint of the project was that only HTML was covered and explained. It would have been great to cover more concepts such as CSS and JavaScript, but due to the project objective and time constraints, this was not implemented. The application delivers on its intended purpose but could have been exciting to help other students learn other languages of the web.

Another constraint initially was my lack of knowledge of PHP. This made the project very difficult at first, as I had to undertake a lot of research into PHP in order to use it effectively.

6.3 Next Steps

As previously mentioned, due to time constraints CSS and JavaScript was not included in this project. It is my personal aim to expand this project portfolio and incorporate teaching the other languages of the web within this web package. Not only would this provide great personal experience, it would show I am clearly motivated to improve myself as well as others. Building on this portfolio would be of great help when undertaking interviews to potential employers in the future.

An interesting addition to the project would be sending users monthly emails about the new languages covered and quizzes that have been released. This would keep users up to date if they want to improve their scores on the global leaderboard.

6.4 Project Summary

Overall, this project was immensely satisfying, this is because it gave me the motivation to learn languages such as PHP which I wasn't very proficient in and wasn't initially fond of. The research conducted detailing learning theories was very interesting and gave me the tools to help others learn differently as well as myself. Taking the time to do all this has allowed me to complete a working project which I believe tackles the aims set at the beginning. It's successful in providing students the ability to learn in an interactive way using the features of the web package such as the quizzes and HTML playground.

It has made use of the principles researched in the literature review which takes a variety of learning styles into account. This was done using different multimedia such as videos, diagrams, text and quizzes. The quizzes provide the instant feedback a user requires and the global leaderboard helps to motivate the users to understand the information alluding to HTML. This project could be extended to make use of other languages of the web very easily. The quizzes could be replicated with different questions and the content altered in the required way.

7 References

[1] – Longman, W. (1998) Ragett on HTML 4 – Chapter 2 – a history of HTML

Available at: https://www.w3.org/People/Raggett/book4/ch02.html (Accessed on 06/04/2020)

[2] – Available at: https://www.internetlivestats.com/total-number-of-websites/ (Accessed on 06/04/2020)

[3] - Dunlosky, J. (2013) Strengthening the Student Toolbox: Study Strategies to Boost Learning, American Educator, Vol. 37, No 3

Available at: https://www.aft.org/sites/default/files/periodicals/dunlosky.pdf (Accessed 20/12/2019)

[4], [5] – Ford, J D. (2011) How the brain learns

Available at: https://trainingindustry.com/articles/content-development/how-the-brain-learns/ (Accessed 22/12/2019)

[6] - Flavell, J. (1976). Metacognitive aspects of problem-solving. In L. Resnick (Ed.), The Nature of Intelligence. Hillsdale, NJ: Erlbaum Assoc.

[7] - Argyris, C. (1976). Increasing Leadership Effectiveness. New York: Wiley.

[8] - DeBono, E. (1967). New Think: The Use of Lateral Thinking in the Generation of New Ideas. New York: Basic Books.

[9] - Pask, G. (1975). Conversation, Cognition, and Learning. New York: Elsevier.

[10] – Gagne, R. (1985). The Conditions of Learning (4th Ed.). New York: Holt, Rinehart & Winston.

[11] - Gagne, R., Briggs, L. & Wager, W. (1992). Principles of Instructional Design (4th Ed.). Fort Worth, TX: HBJ College Publishers.

[12] - Layton, Mark C. and Ostermiller, Steven J. (2007, New Jersey). Agile Project Management.

[13] – Gagne's publication facts – Available at http://worldcat.org/identities/lccn-n79018030/ (Accessed 20/04/2020)

Manchester
Metropolitan
University

8 Appendix

8.1 Terms of reference

Learning outcomes

LO1 - Independently plan, manage and successfully complete a project of substantial size in an area that is relevant to their degree programme.

LO2 - Design, implement and interrogate database systems

LO3 - Critically reflect and evaluate existing work and their own work.

LO4 - Demonstrate effective communication, decision making and creative problem-solving skills, and identify appropriate practices within a professional, legal and ethical framework.

Project Background

HTML (Hyper Text Markup Language) is the language used that makes up every webpage. It's incredibly useful to understand how it works and the syntax it consists of. Many people find it hard to learn, or be motivated to learn it, this project aims to tackle this by making it exciting and rewarding for the users.

Target audience

The target audience for this project is mainly 16-18 year olds. The content used and my strategy to motivate this audience will reflect that. By taking an ambitious approach to how young people learn, this project will:

Provide **incentive** by rewarding the user when they complete quizzes

Set goals to aim for?

Unlock something?

Earn XP to level up account?

Make it **competitive** by showing a leaderboard of the quiz score of each user.

Providing **intelligible** content so the information is easy to understand and clear to the young audience.

Provide **motivation** by using motivational quotes throughout the website in between content– “Optimism is the faith that leads to achievement.”

Other sources of information on HTML

There are many websites online which help teach people HTML, sources such as; Mozilla Development Network, W3Schools, learn-html, codecademy and many more. All are very useful where-in people self-teach HTML via the information available on these websites. This project is realistic and doesn’t aim to exceed the content available on these websites.

Expanding on LOs

LO3 (Critically reflect and evaluate existing work and their own work) is the learning outcome which has most relevance for this project for me. This is because the development process that this project will be using is AGILE. The work done will have to be constantly tested, critically reviewed and improved, so I believe this makes perfect sense. I will also be critically analysing other sources of HTML information on the internet, their advantages and disadvantages so I look forward to realising this learning outcome.

LO1 (Independently plan, manage and successfully complete a project of substantial size in an area that is relevant to their degree programme) is also very relevant. Time management is a serious issue, however, by managing time effectively and meeting the objectives I have set, this LO is recognised as essential in the successful completion of this project

Aim

The aim of this project is to teach HTML to beginners (16-18 year olds) in an intuitive manner by creating an interactive website containing information and quizzes.

Objectives

Research and analyse existing systems as well as learning methodologies of other sources for teaching HTML, identifying their strengths and weaknesses

Plan website design, content and structure

Develop website front-end

Plan and develop quiz questions based on website content

Plan website back-end;

Database table holding vital user information (username / password)

Database table holding user progress on the quizzes

Database connectivity

Develop website back-end

Create a login and registration system

Reflect on and review the interactive website by asking fellow students to trial it so I can act on the feedback – Rigorous testing

Problems

Time management: Allocating the right amount of time or running out of time to deliver each objective. Otherwise the project may not be delivered on time (Layton & Ostermiler, 2007)

Log in system: Creating a secure login system which isn't vulnerable to being hacked

Amount of information: The amount of information to have the user read before displaying quiz questions on it

Database: Having the database correctly track the user's progress and scores on each quiz. Also, a problem may arise connecting to the database

Deliverables

The deliverable for this project would be an interactive website containing information and short quizzes on HTML while tracking user progress including how well they score on quizzes.

Project timeline/Deliverables

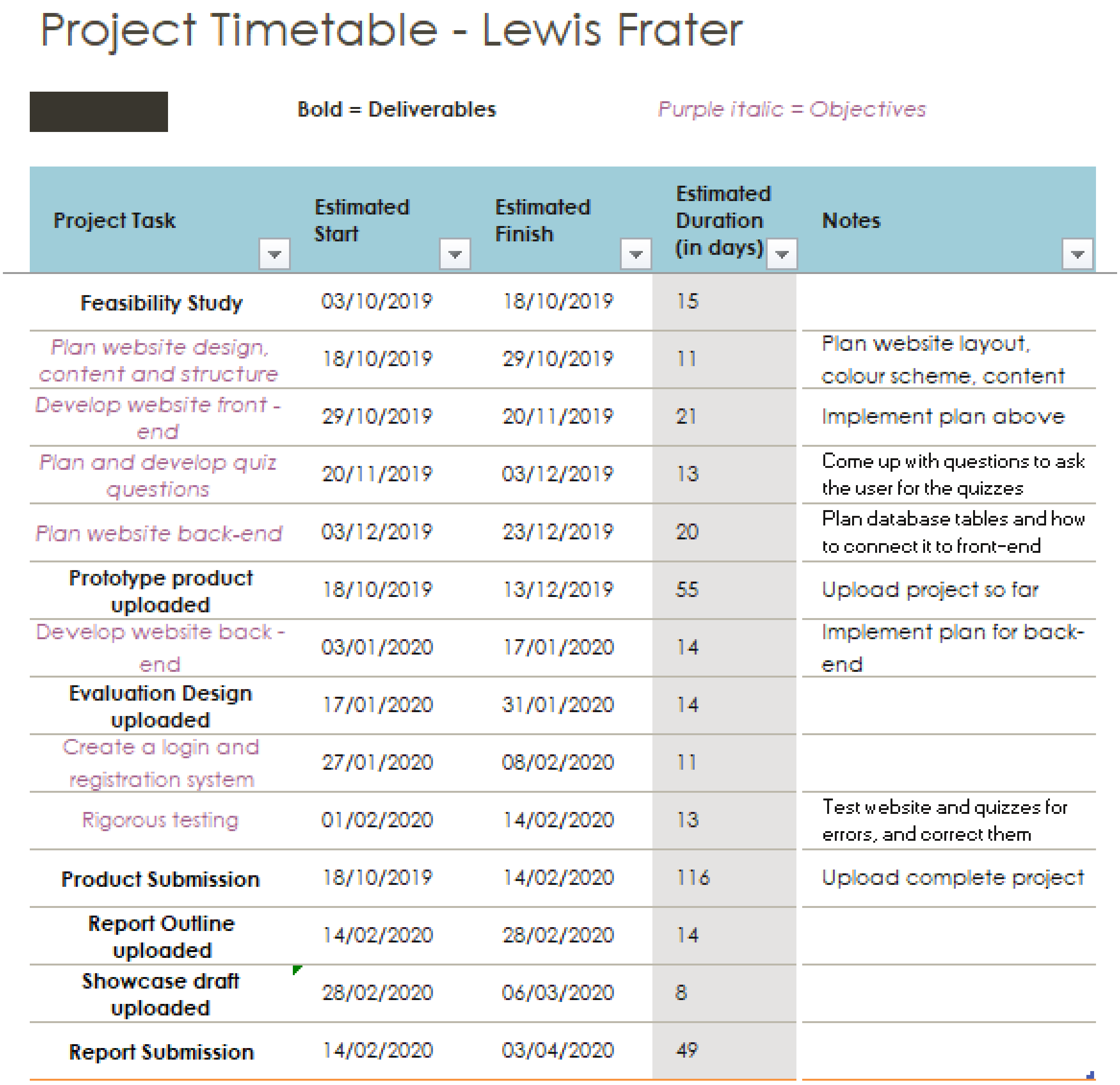
My project timeline is subject to change due to the on-going research I will be undertaking, except the obvious deadlines that need to be met.

The AGILE development cycle will be used for this project, following its methodology which aims to reduce risk and manage time in an efficient manner. Using sprints which is a popular agile framework for organizing work and exposing project progress (Layton & Ostermiller, 2007).

Iteration of a sprint:

Design -> test -> Integrate -> test -> Develop -> test

In the first iterations the highest priority features will be implemented. The later iterations will add the sacrificial, or optional features as well as refining the main features and mainly the aesthetic look of the quizzes/webpage.



Project Timetable - Lewis Frater

Bold = Deliverables *Purple italic = Objectives*

Project Task	Estimated Start	Estimated Finish	Estimated Duration (in days)	Notes
Feasibility Study	03/10/2019	18/10/2019	15	
Plan website design, content and structure	18/10/2019	29/10/2019	11	Plan website layout, colour scheme, content
Develop website front - end	29/10/2019	20/11/2019	21	Implement plan above
Plan and develop quiz questions	20/11/2019	03/12/2019	13	Come up with questions to ask the user for the quizzes
Plan website back-end	03/12/2019	23/12/2019	20	Plan database tables and how to connect it to front-end
Prototype product uploaded	18/10/2019	13/12/2019	55	Upload project so far
Develop website back - end	03/01/2020	17/01/2020	14	Implement plan for back-end
Evaluation Design uploaded	17/01/2020	31/01/2020	14	
Create a login and registration system	27/01/2020	08/02/2020	11	
Rigorous testing	01/02/2020	14/02/2020	13	Test website and quizzes for errors, and correct them
Product Submission	18/10/2019	14/02/2020	116	Upload complete project
Report Outline uploaded	14/02/2020	28/02/2020	14	
Showcase draft uploaded	28/02/2020	06/03/2020	8	
Report Submission	14/02/2020	03/04/2020	49	

Required Resources

Computer with a web browser

Eclipse IDE (Java web apps)

Brackets IDE (For the HTML, CSS)

Text editors (Microsoft word)

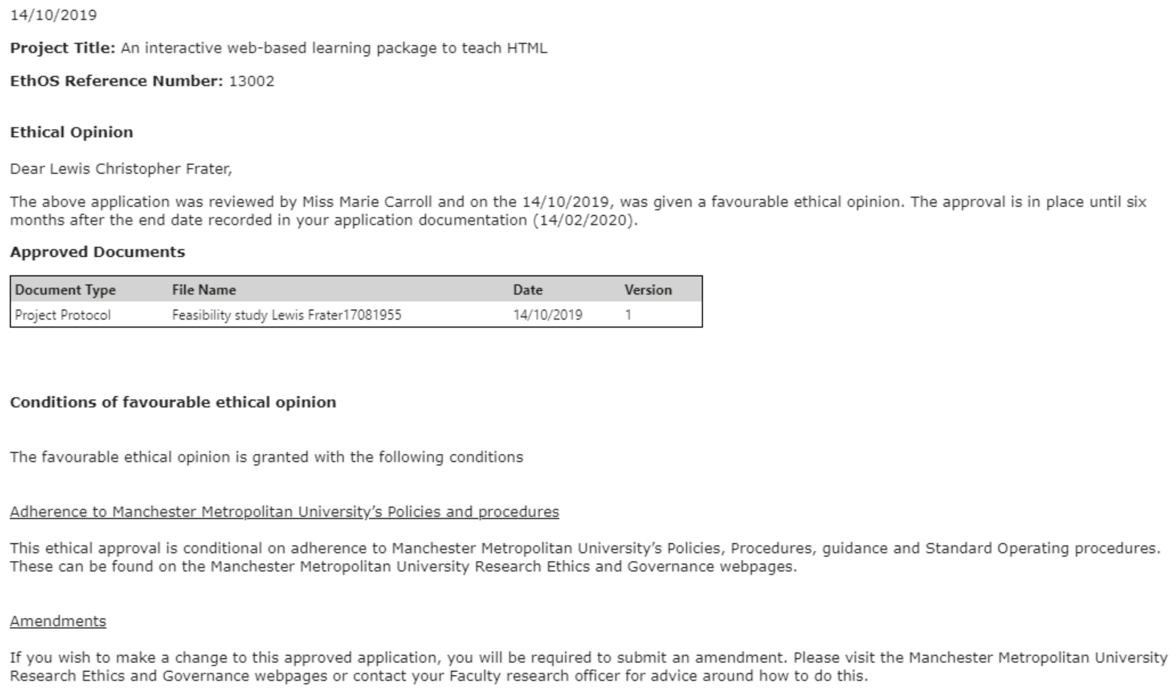
MMU library and Google Scholar (Access to books for research purposes)

References

Layton, Mark C. and Ostermiller, Steven J. (2007, New Jersey). Agile Project Management

Ethics

Proof of my EthoS application being accepted.



14/10/2019

Project Title: An interactive web-based learning package to teach HTML

EthOS Reference Number: 13002

Ethical Opinion

Dear Lewis Christopher Frater,

The above application was reviewed by Miss Marie Carroll and on the 14/10/2019, was given a favourable ethical opinion. The approval is in place until six months after the end date recorded in your application documentation (14/02/2020).

Approved Documents

Document Type	File Name	Date	Version
Project Protocol	Feasibility study Lewis Frater17081955	14/10/2019	1

Conditions of favourable ethical opinion

The favourable ethical opinion is granted with the following conditions

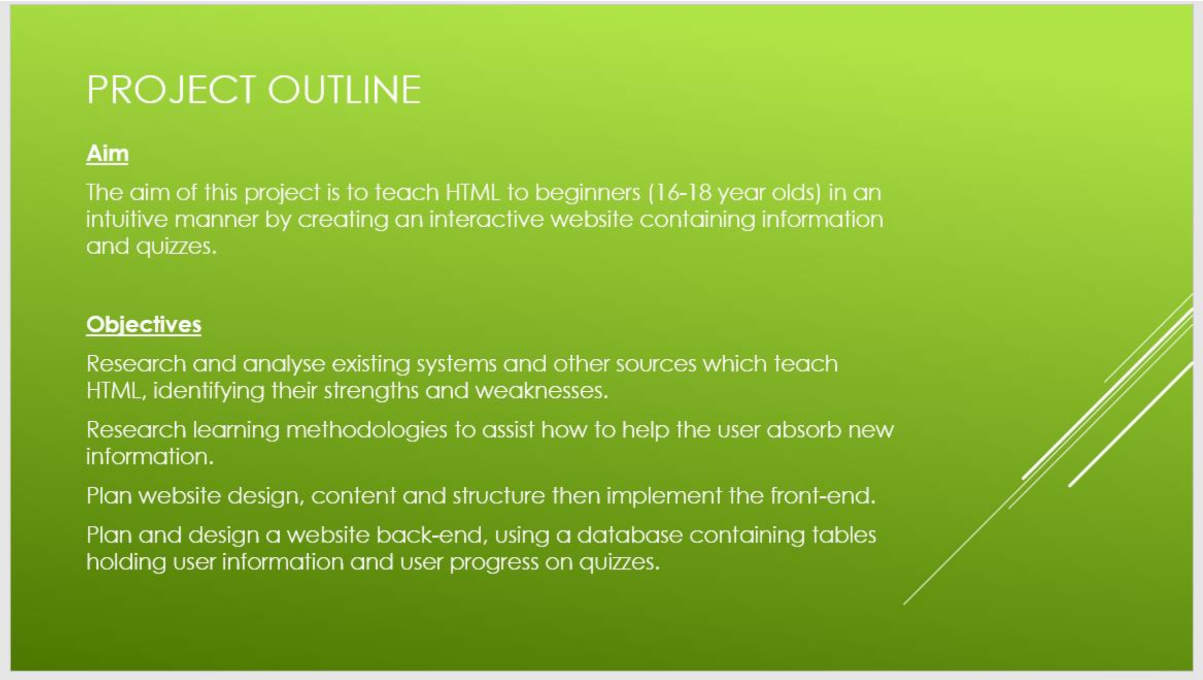
Adherence to Manchester Metropolitan University's Policies and procedures

This ethical approval is conditional on adherence to Manchester Metropolitan University's Policies, Procedures, guidance and Standard Operating procedures. These can be found on the Manchester Metropolitan University Research Ethics and Governance webpages.

Amendments

If you wish to make a change to this approved application, you will be required to submit an amendment. Please visit the Manchester Metropolitan University Research Ethics and Governance webpages or contact your Faculty research officer for advice around how to do this.

8.2 Showcase Presentation Slides



PROJECT OUTLINE

Aim

The aim of this project is to teach HTML to beginners (16-18 year olds) in an intuitive manner by creating an interactive website containing information and quizzes.

Objectives

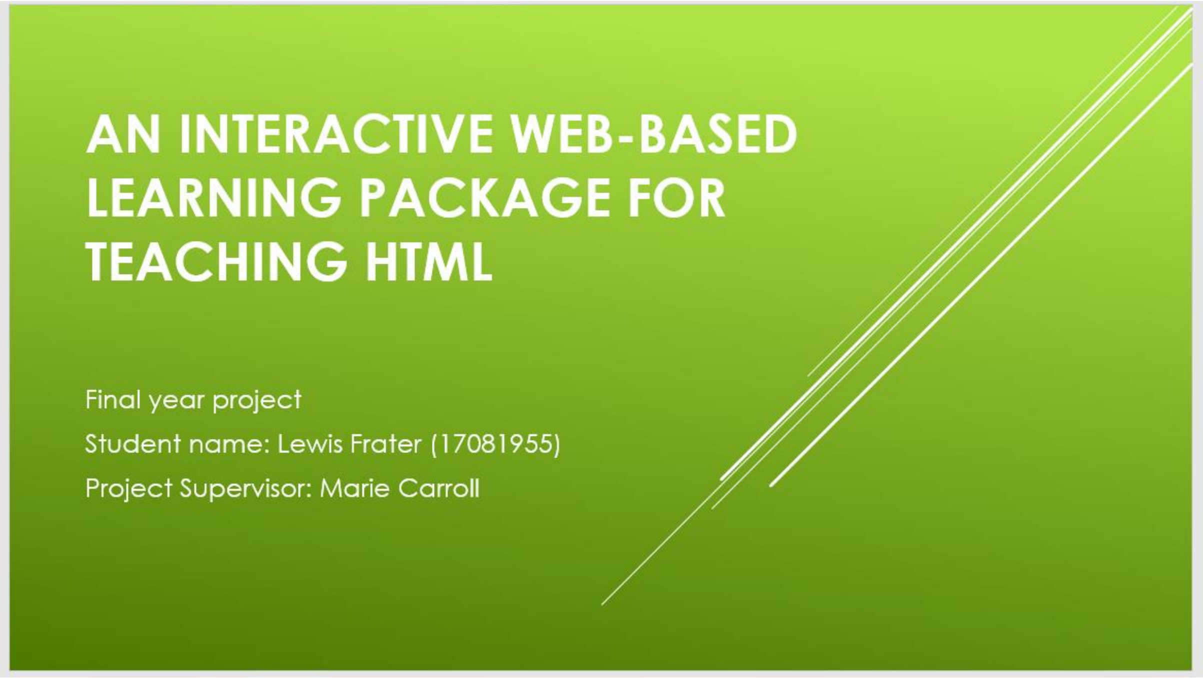
Research and analyse existing systems and other sources which teach HTML, identifying their strengths and weaknesses.

Research learning methodologies to assist how to help the user absorb new information.

Plan website design, content and structure then implement the front-end.

Plan and design a website back-end, using a database containing tables holding user information and user progress on quizzes.

Figure 38: Slide 1



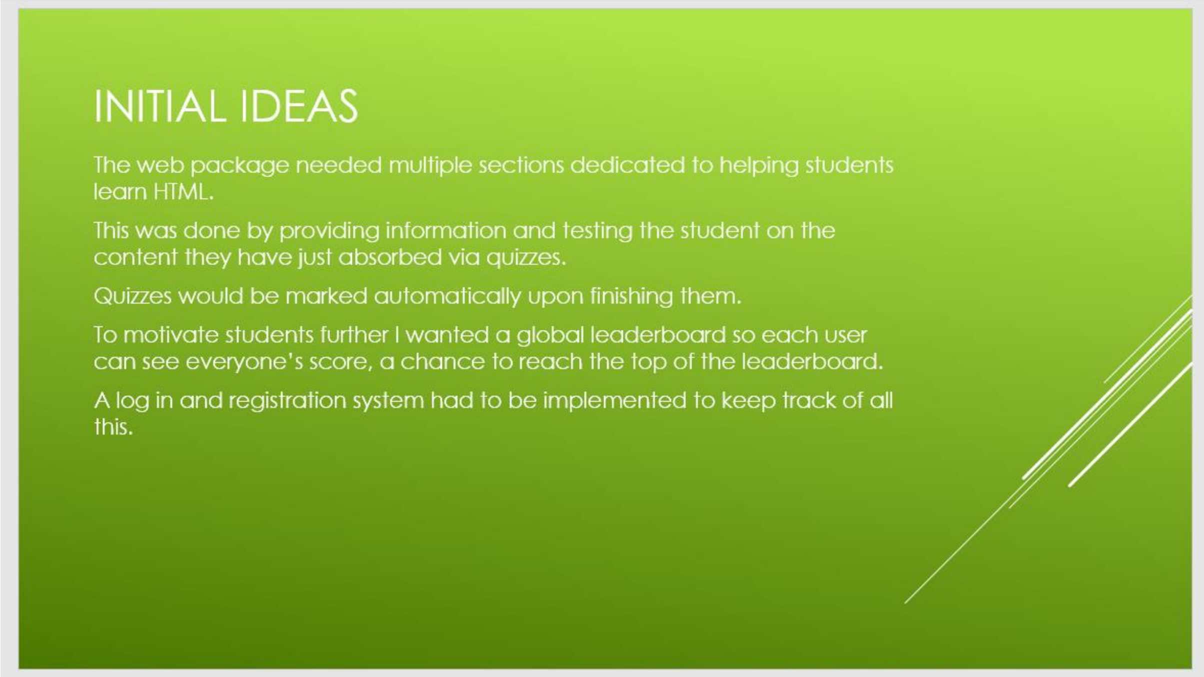
AN INTERACTIVE WEB-BASED LEARNING PACKAGE FOR TEACHING HTML

Final year project

Student name: Lewis Frater (17081955)

Project Supervisor: Marie Carroll

Figure 39: Slide 2



INITIAL IDEAS

The web package needed multiple sections dedicated to helping students learn HTML.

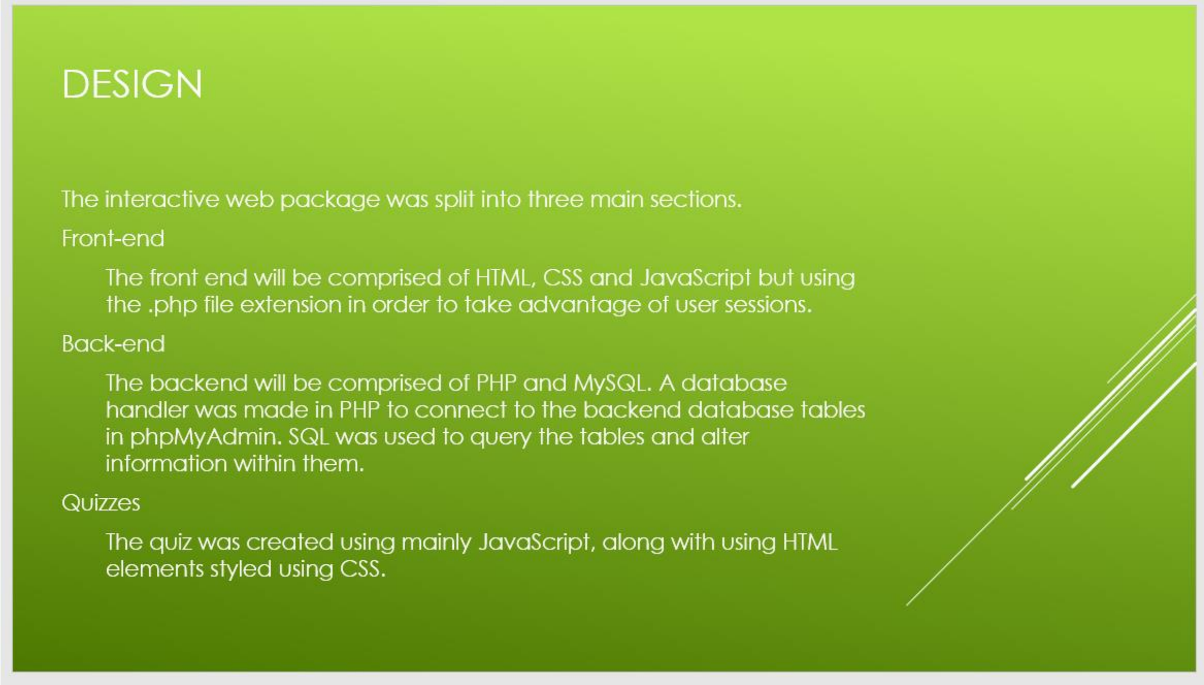
This was done by providing information and testing the student on the content they have just absorbed via quizzes.

Quizzes would be marked automatically upon finishing them.

To motivate students further I wanted a global leaderboard so each user can see everyone's score, a chance to reach the top of the leaderboard.

A log in and registration system had to be implemented to keep track of all this.

Figure 40: Slide 3



DESIGN

The interactive web package was split into three main sections.

Front-end

> The front end will be comprised of HTML, CSS and JavaScript but using the .php file extension in order to take advantage of user sessions.

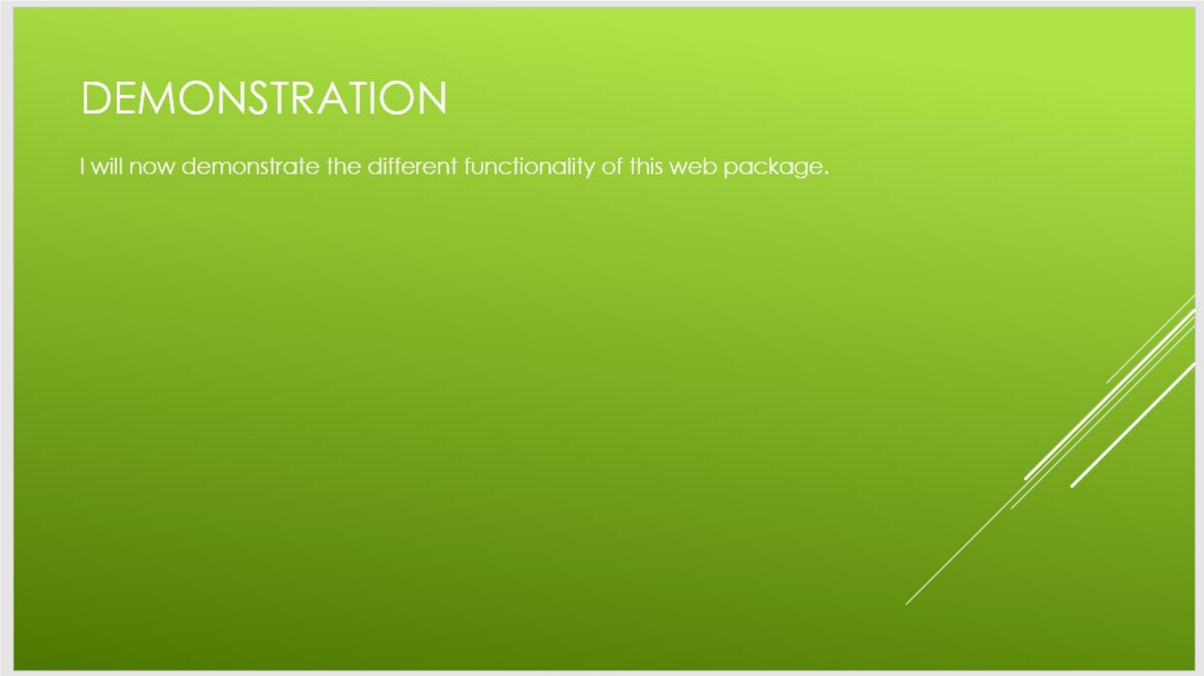
Back-end

> The backend will be comprised of PHP and MySQL. A database handler was made in PHP to connect to the backend database tables in phpMyAdmin. SQL was used to query the tables and alter information within them.

Quizzes

> The quiz was created using mainly JavaScript, along with using HTML elements styled using CSS.

Figure 41: Slide 4



DEMONSTRATION

I will now demonstrate the different functionality of this web package.

Figure 42: Slide 5

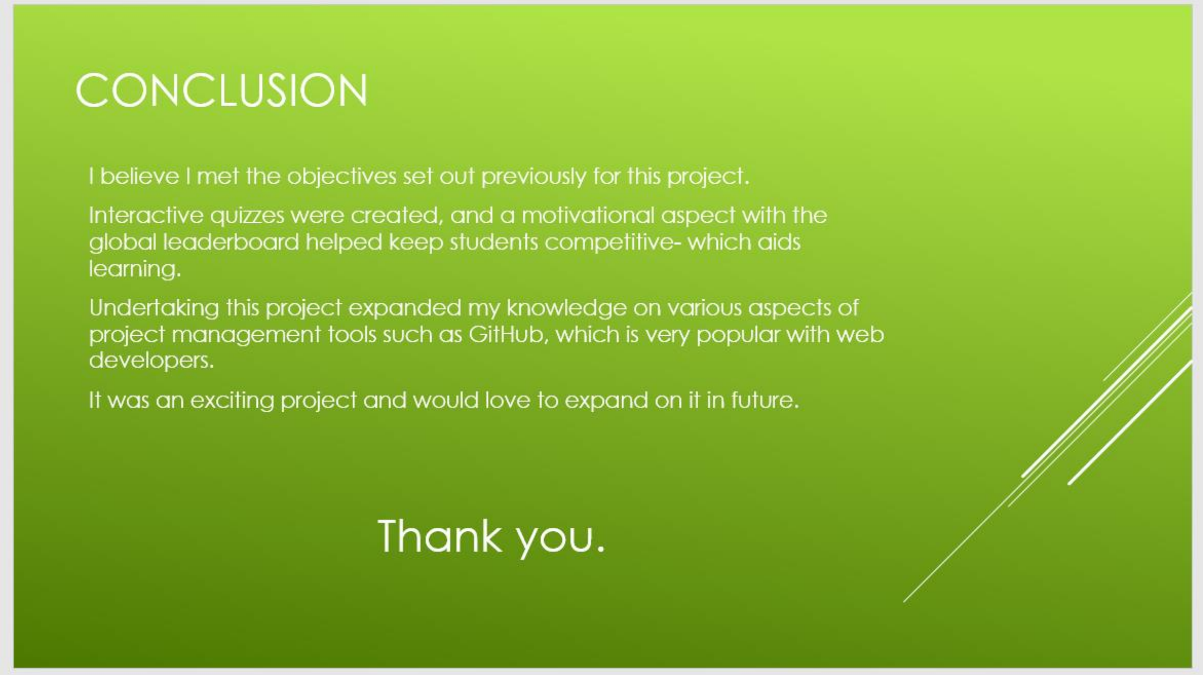
RE-USEABILITY

This web package could be expanded upon in future covering other concepts such as CSS and JavaScript.

These are important languages which provide the foundation of web development.

This would also benefit me by enlarging my portfolio for potential employers in the future.

Figure 43: Slide 6



CONCLUSION

I believe I met the objectives set out previously for this project.

Interactive quizzes were created, and a motivational aspect with the global leaderboard helped keep students competitive- which aids learning.

Undertaking this project expanded my knowledge on various aspects of project management tools such as GitHub, which is very popular with web developers.

It was an exciting project and would love to expand on it in future.

Thank you.

Figure 44: Slide 7