

Designing initial Quiz Website

Harris Spratt

¹ Edinburgh Napier University, Merchiston Campus, Edinburgh, EH10 5DT, Scotland
40530696@live.napier.ac.uk
10/10/2024

1 Introduction

1.1 Background Description

I aim to design a fully functioning quiz website that offers different quiz categories alongside additional features such as a leaderboard. This quiz site will be for entertainment purposes and will aim to appeal to trivia enthusiasts. The site will contain a Navbar across the top of the page. This will act as the main navigation tool and will feature on every page not just the home page. Each word in my navbar will be a category of quiz eg Geography. There will also be a leaderboard page and link for it on the navbar. The colour scheme across the website will be light blue and white. Each quiz page will have the same layout. At this stage I am looking to one large square in the middle of the page with the question written across the top of the box and then 4 multiple choice ovals below laid out in a 2x2 grid. Ideally the user will click an answer and then there will be two arrows at the bottom of the square where the user can move to the next question or back to the previous one. If I am unable to implement this where the questions are dynamically changing on the one page, then I will opt for a basic quiz template where each question will have a rectangle box and each one will be laid out on the page one after the other. Each quiz will also have a timer running where the user has a specific set time to answer the questions. This allows another value column in the database needed for the leaderboard feature. The leaderboard could then be sorted by number of correct answers or could also be sorted by quickest time to complete a quiz. Finally a user sign in page will be available in the top right corner of the navbar. This is complex however as I will need to consider the implications and potential dangers with storing usernames and passwords. At this stage only quizzes pre defined by me will be available on the website.

1.2 Background Research

Kahoot

. When researching through some of the most visited quiz websites one of my main criticisms would be the amount of content on their home pages. See figure 1 for an example of Kahoot's homepage. Here the home page is very clustered and this is something I aim to avoid for my home page. There is no consistent colour scheme either on Kahoot with the home page including yellow, white, purple, blue, green and red. I will only look to incorporate light blue, white and black text for my web pages. This should provide some consistency and make navigation and decision making across my website much easier. A feature I liked on Kahoot was when hovering over an option on their navigation bar a dropdown menu appears. I could implement this for my website eg for the word geography on my navbar, Countries, flags and cities could appear on a dropdown menu and thus allow the user to pick the type of geography quiz they wish to do.

Sporacle

Sporacle gives the user the option to sign in with a simple sign in button. This may be useful for my website as for the leaderboard to keep track of who achieved what then the entries will have to be associated with a username. In terms of their homepage, Sporacle makes more use of whitespace than Kahoot which I think is easier on the eye. Since their quizzes are created by users there is no consistent template for the layout of each quiz. So, I cannot take inspiration from a quiz design. Alongside the sign in button there is a blank portrait icon which allows user to sign up if they do not have an account yet. Potentially when signing in I could attempt to make this icon change to the first letter of their username. See figure 2 for a screenshot of the home page with a purple highlighted box of the sign in and account icon in the top right corner.

Britannica

See figure 3 for the quiz layout for this website. This is an exemplary layout of what I will look to replicate for my website. The timer resets every question completed but for the purpose of my scoreboard feature I will keep the timer running the whole time. Also, when the user clicks an answer it gives instant feedback as to whether the user's answer was

correct. I am undecided yet whether to total up the answers at the end of the quiz or to replicate what Britannica does.

Background research summary

Through researching and using these three websites I was able to envision the colour scheme and complete layout of my website. The business of Kahoot home page allowed me to design a simple blue and white colour scheme home page with little content. Sporacle made me incorporate a user sign in option and Britannica provided me a template for my actual quiz pages.

2 Features and Pages

2.1 Navigation Bar

The navigation bar at the top of my pages will be the primary way to navigate through my website jumping from quiz to quiz or from quiz to leaderboard. The navbar will be light blue and the sign in option and icon will be part of the navbar, positioned top right. The dropdown menu when a user hovers over a word on the navbar will allow the user to jump straight to the specific quiz within that word's category.

2.2 Quiz pages

Each quiz page will have the same layout for answering questions. This makes it fair to compare different quizzes for speed for the leaderboard but also make the user want to play more quizzes due them knowing the layout and way to answer. The 2x2 grid with multiple choice answers gives the user a chance to recognise an answer and not leave them demotivated by leaving questions blank.

2.3 Leaderboard

A leaderboard provides me with a chance to incorporate a database into my website. It also makes the completion of quizzes more worthwhile for the user and encourage them to frequently visit the website to compare scores and speed of completion.

2.4 Timer

A timer provides another metric to measure on the leaderboard and again boots engagement with the user as something to directly play against. The timer will most likely be set at the same time limit across all quizzes.

2.5 User accounts and sign in option

An account username will be the name entry to the leaderboard. An sign in and account feature will also add another layer of sophistication to my website with another potential database to store these details. A consideration of privacy and safety will also have to be in effect as passwords are being stores.

2.6 Colour Scheme

A light blue and white colour scheme will make my content readable and bright. For my text within the quiz, I may use black text to make the quiz look more professional.

3 Site Organisation

3.1 Navigation Tree

See Figure 4.

3.2 Navigation Flow

The navbar will be the only to navigate through my website across different quiz pages and the leaderboard page. The words geography, history and sports will not be a link on the navbar but only a word where which when the user hovers over it a dropdown menu will appear. The dropdown menu will give the user the option of three quiz pages for each category. Sign in and leaderboard will be a direct link to their pages via their word on the navbar. The navbar will be visible on every page across the website allowing the user to navigate back and forth. This will provide a consistent intuitive way to navigate throughout.

4 Wireframes

4.1 Home Page

See figure 5 for an initial sketch of the home page. The home page of my website is just a landing place and will not have much content. There will be a quiz related image in the middle with some text underneath welcoming the user to the page. When implementing the nav bar the text will have to be the right shade of blue because looking at my initial design the colour of text for the navbar may be too light to read efficiently.

4.2 Quiz template

See Figure 6 for an example of the quiz layout for capital cities quiz page. The navbar will stay visible throughout all questions and the user will be able to navigate back and forth using the two arrows at the bottom corner of the page between questions. Therefore my website has to be able to dynamically change the questions. Also when a user clicks an answer the box will become grey so that the user knows the website has recorded their answer. I will try out black and light blue text to see which looks better and which is easier to read. The numbers top right of the quiz box will indicate what question number the user is on so that is something again which will have to dynamically change. The text top left of the box is the title of the quiz. I can then replicate this template for all my quiz questions. One anomaly I can think of however is that for a flag quiz instead of having text in the centre of the box as a question I will need an image of a flag. Therefore I will need to adjust my CSS to achieve this by potentially adding in a class/ID.

5 Project plan

See figure 7 for a table of dates and tasks to be completed. Initially I will set up my working environment through PyCharm. After having difficulties setting up my virtual machine, PyCharm will be my main working

environment for my code and deployment. I will also link a GitHub repository to this so that I am able to just keep pushing my code to it and then at end up of project I can zip up code using GitHub to submit. Then I will move on to implement my homepage which should be straightforward enough. I will look at my project last year from web technologies to refresh myself on the navbar layout and CSS as I used a similar one there. Designing and coding my quiz template will be next in line along with researching my actual content of the quizzes. I will do one basic quiz to start with for each category which will then let me test my dropdown navbar works correctly and that the quiz totals up answers correctly. Previously I have used JavaScript to achieve a quiz but I am more comfortable using python so I will research how to implement a quiz using python. If I achieve these steps I should then have a base project that I can fall back on and submit if I fail to add my additional leaderboard and sign in functionality. Theoretically I then have to try to implement the sign in option. As this will be used as the name for the entry to my leaderboard database and if I achieved the leaderboard database without a name column then it would be no use. One challenge I will face with a sign in functionality is the storage of passwords. I will discuss this in the next section. The sign in link from the navbar will take the user to a simple sign in page. Design of the page will not be too important just two simple text boxes to enter a username and a password. Once the sign in option is achieved, I will implement a leaderboard page and database. The visible page will just be a simple black text table. I am currently studying advanced database systems alongside this module so I already have a good understanding of SQL and databases which will be beneficial. Finally I will begin writing my second report and also I will record my screencast. I will record this using OBS Studios and during screencast will show all functionalities of my website. At this current stage I am unsure how I will host my website due to trouble with the VMs but I will research through different options including Heroku's free tier. Therefore with this plan this gives me at the very least a deployable quiz website using python flask which should meet the minimum project requirements. Any implementation of the sign in and leaderboard will allow me to showcase database skills within a web environment and will bolster my website massively. I see this project as a good progression of my web skills from my second year website which was an information site with a tiny quiz at the end.

6 Additional Features

6.1 Sign in database

Skipping ahead to the end of the workbook I believe that I will use SQLite to implement my database parts of my website. This will allow me to implement a relational database. I will have three columns in my database. A primary key called ID which will just autoincrement for each entry and then also a username and password column. Security is vital for this database so the database will only store the hashed version of the password and not the password itself. I will use py-bcrypt to hash my password and then when the user enters their password the hash of the password input is generated and compared to the hash value stored in the database. Hashing the password practically makes it unrecognisable in the database or to hackers via transforming the password into a fixed length string via bcrypt's password hashing algorithm. This should prevent brute force attacks or if not prevent slow the hacker down massively. This is because it makes each guess take longer due to the length of the hash string. However hashing is still vulnerable to rainbow table attacks. This is an attack where the hacker has the corresponding hash values of the most common passwords and compares these to find a match in the sites stored user passwords. At the minute if two user's had the same password then their hashes would be identical. However this is where another useful feature of py-bcrypt is utilised which is the ability to salt the hashed passwords. I will use the gensalt() function to randomly generate a salt value. This is then added to the password before hashing and when the password is hashed, the salt is embedded in the hash output. When verifying a password bcrypt then removes the salt.

6.2 Leaderboard database

This will be more simpler to implement due there being no need to encrypt any passwords. However this database will need to receive data dynamically rather than inputted data. It will need to receive the completed quiz time and also the username associated with that account. Therefor there will be three columns: username – string, time – date/time and marks – integer. The username as previously mentioned is tied to

the user's account and will potentially have some foreign key relationship to the sign in database. The time column will be in the format of minutes and seconds. Finally the integer marks column will just be a single number which reflect the total number of correct answers out of 20. In the future I could change this to a scoring system by awarding more points to quicker answers.

Appendices

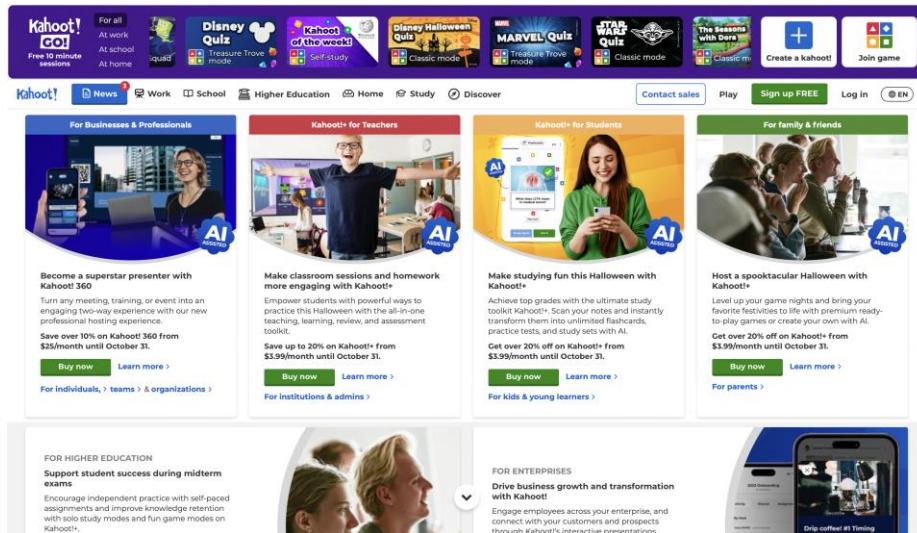


Fig. 1. - Kahoot.com homepage

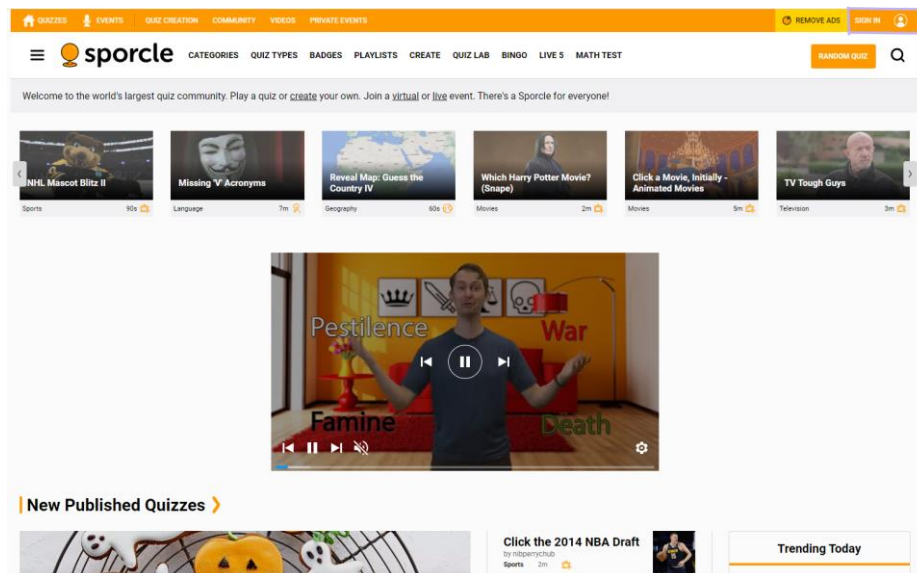


Fig. 2. – Sporcle home page with purple highlighted box in corner



Fig. 3. – Britannica quiz layout

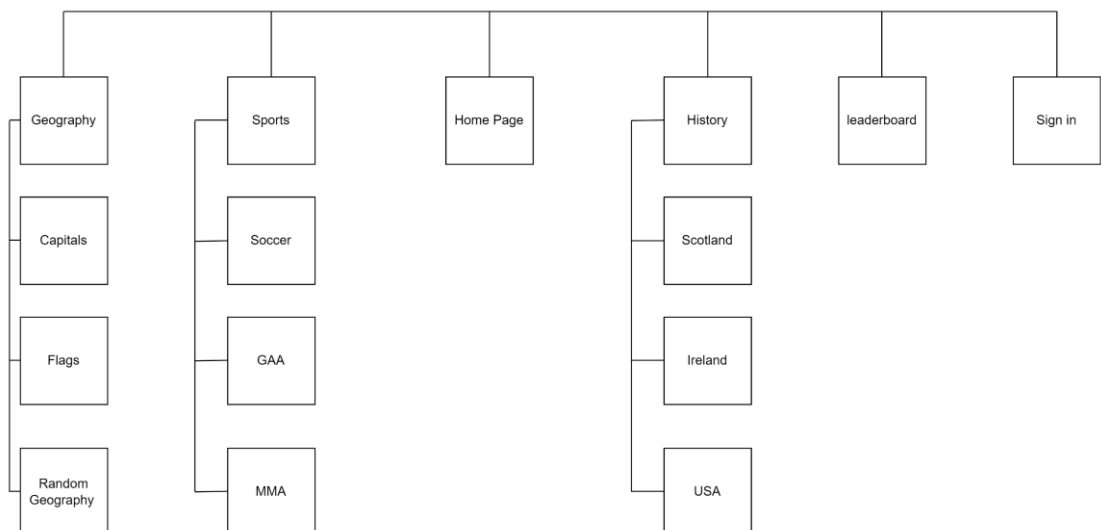
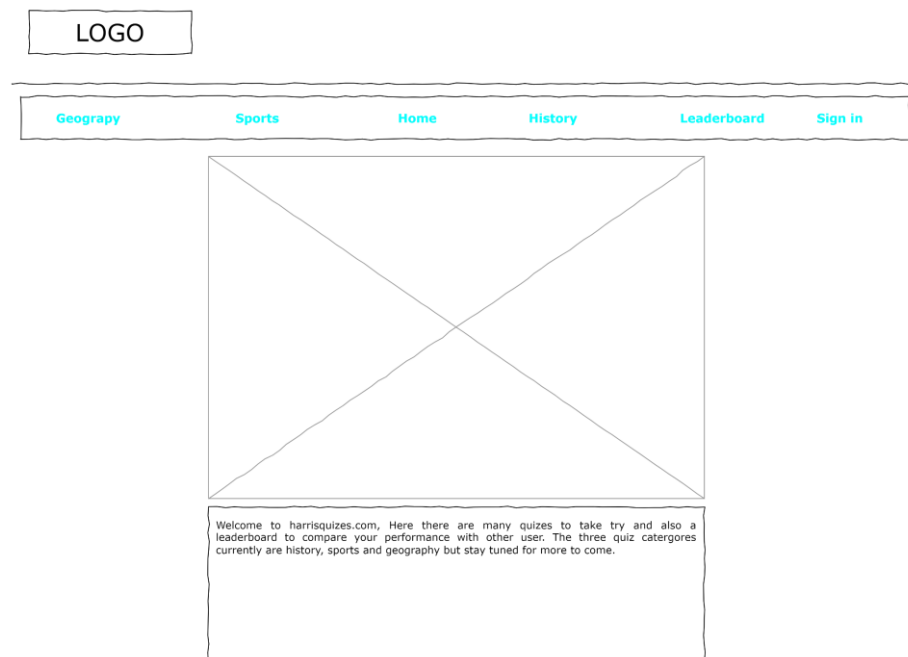


Fig. 4. Navigation tree**Fig. 5.** Initial sketch of home page

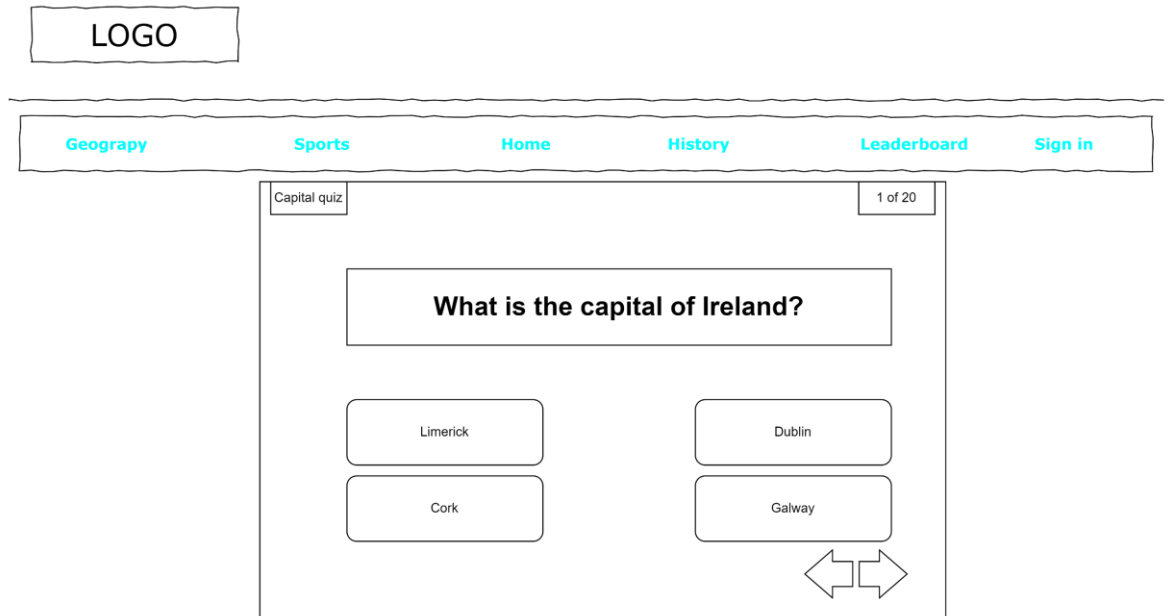


Fig. 6. Initial sketch of quiz layout

Week starting	Task to be completed/stage
14 th October	Set up fully functioning working environment
21 st October	Look to start homepage and in particular navbar CSS
28 th October	Complete home page html and CSS for navbar. Also have all quiz questions ready to implement
4 th November	Complete quiz template CSS and html and insert quiz questions

11 th November	Research through how to achieve a sign option. This is needed before leaderboard as need name to enter
18 th November	Continue to try to implement sign in option and then move onto leaderboard page.
25 th November	Look to achieve leaderboard database and display and begin writing second report
2 nd December	Complete second report and record screencast

Fig. 7. - Project plan with dates