

WebbiSkools Ltd - Quiz Manager

Design Document

I have been commissioned by WebbiSkools to design, build and test a database driven web application which will act as a quiz manager, containing quizzes and their associated answers.

The brief states the application should allow all users the ability to view and answer the questions in the available quizzes. In addition to this, users with higher privileges will have the ability to view the questions as well as the answers on the quiz. Finally, the users with the highest permissions will be able to do all the above, but also edit the questions and answers for a quiz.

With the above brief in mind, I came up with appropriate designs which meet all the requirements outlined by the client.

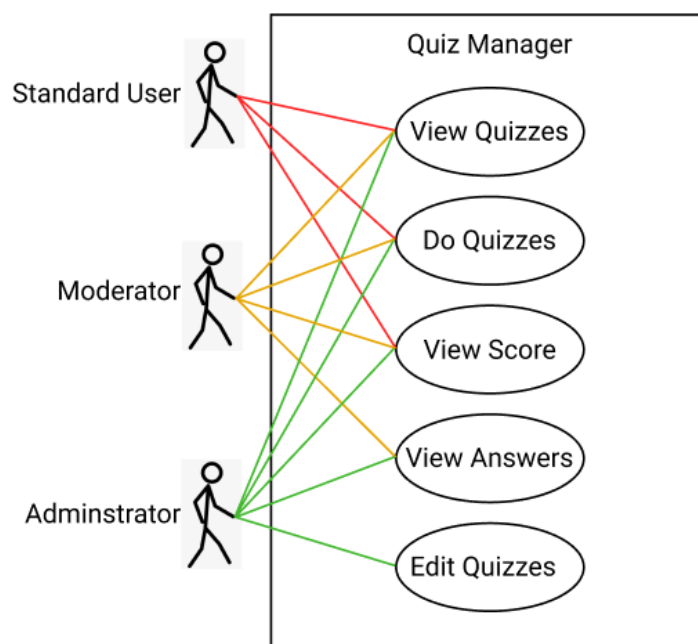
With my understanding of the requirements, I decided to split the three user types into:

Edit – These are the users who will be able to access all functions of the quiz system. This includes editing questions/answers, seeing answers to questions and viewing/partaking in any one of the many quizzes.

View – These are the users who will be able to see answers to questions and view/partake in any one of the many quizzes

Restricted – These are the users who will only be able to view/partake in any one of the many quizzes.

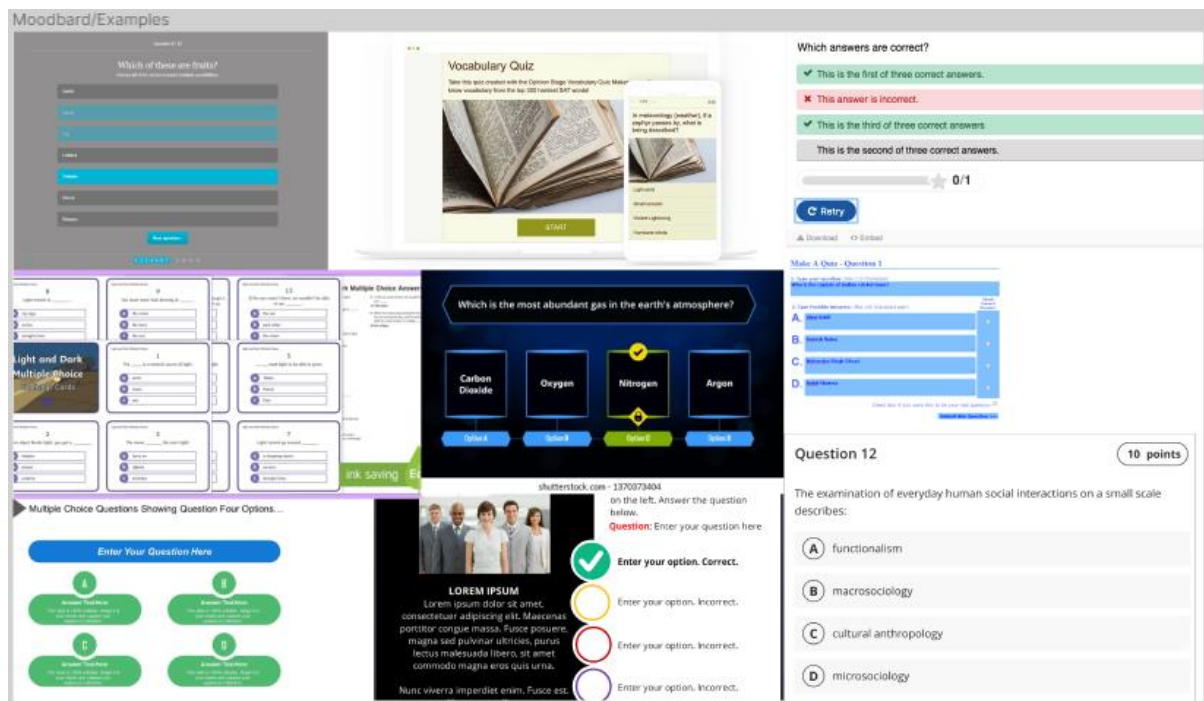
To help me visualise the system, I created a Use Case diagram. This allowed me to easily identify the requirements of each user type and visualise their interactions with the system.



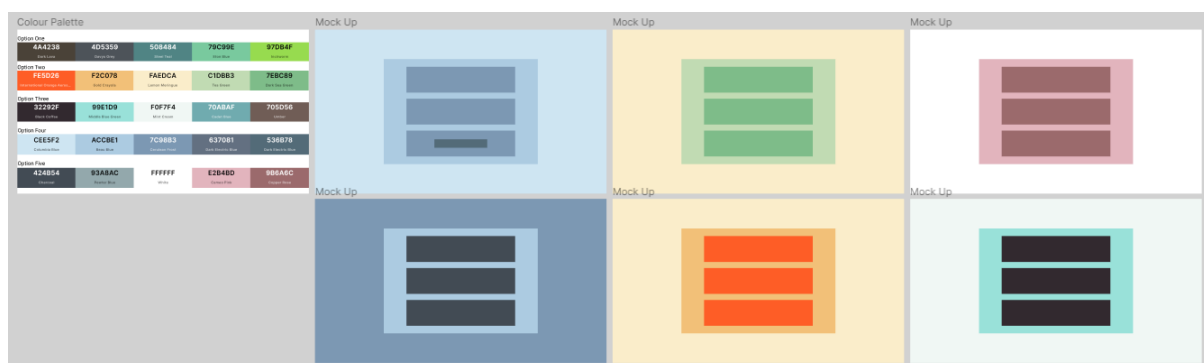
As you can clearly see, the Edit user is able to perform all functions and use all features, whereas the Restricted user is limited to a small subset.

With the information about the functionality of the site understood and the requirements now defined, I was able to begin drafting up some designs for the website. The following designs are my interpretation of the project description given. The requirements are clear, yet somewhat non-specific, which gives me a lot of freedom to create the designs.

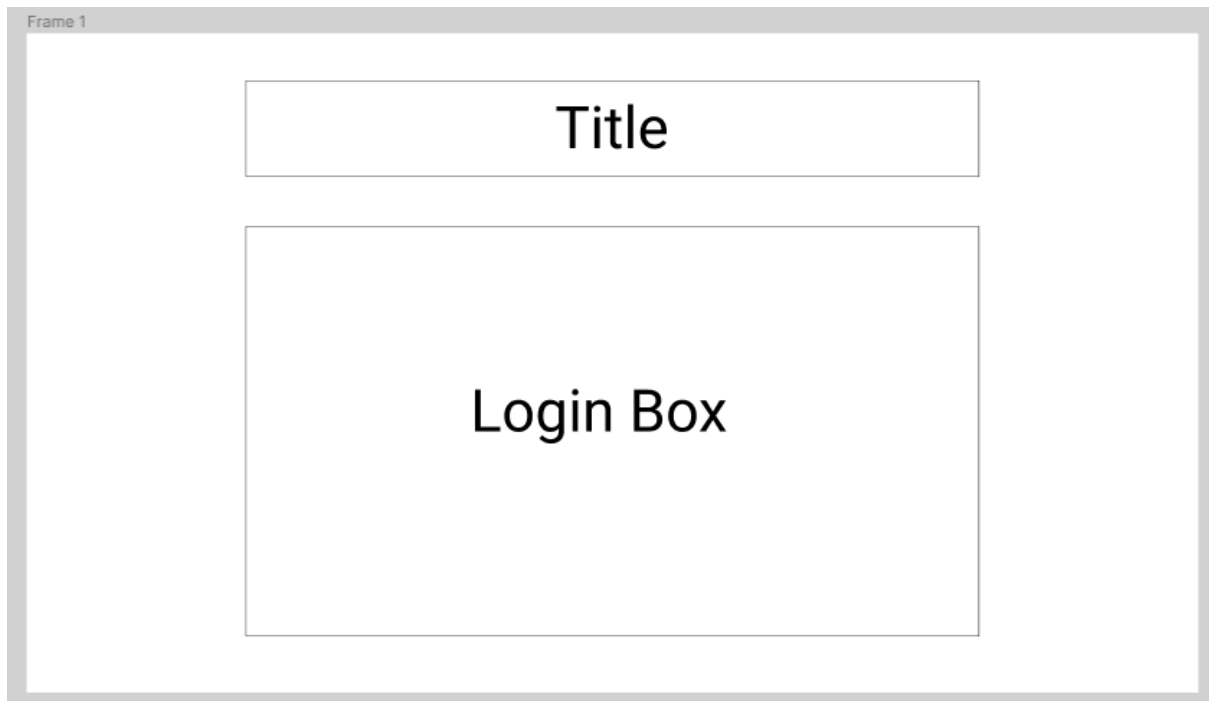
To do so, I first looked towards similar existing systems for ideas and creative inspiration. I took note of some of my favourite designs and collated them into a mood board. The purpose of this was to simply have quick access to reference material, should I be unsure about certain aspects, or about the best way to lay out the information in the most efficient way for the user.



I then decided to design and pick several colour palettes. The reasoning behind doing this early in the process was to allow me to better visualise how the site would look and create the layout with these colours in mind.

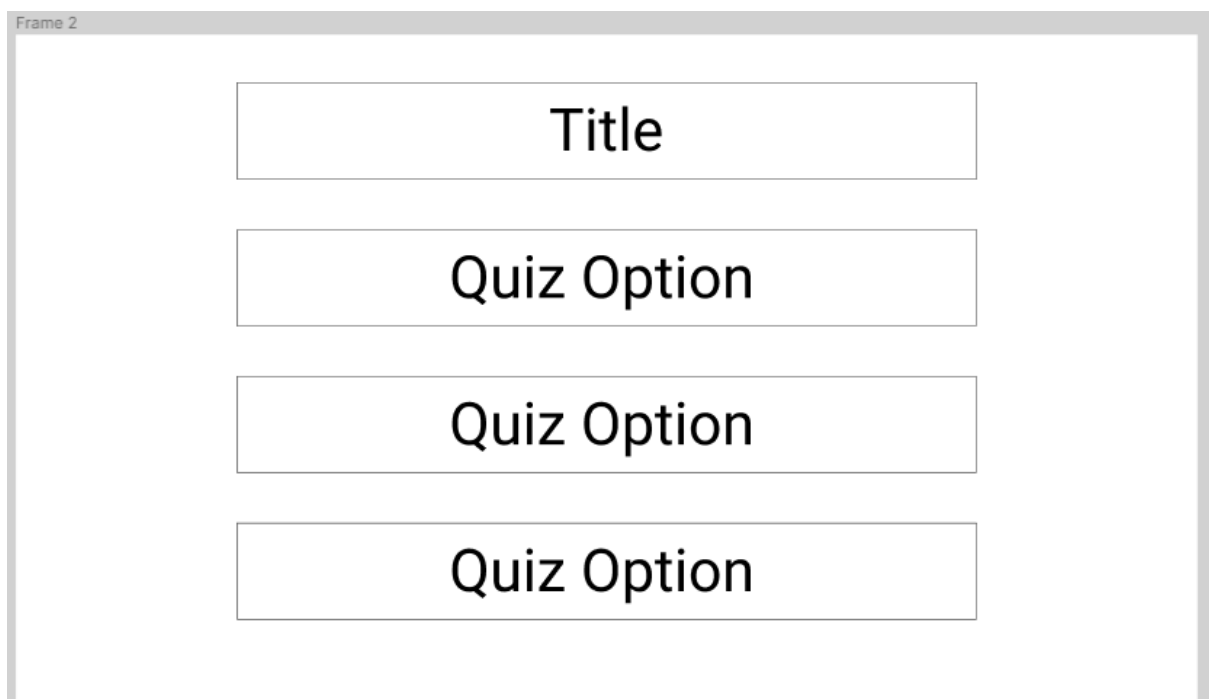


I then began the design phase and created low fidelity wireframe designs of the core components and functionality of the site.



The first page the user would interact with when loading the website would be the homepage. This is where they would be required to login with the credentials they have been provided. This page allows the separation between the different user access and would determine what they see on the following pages of the site after logging in.

There are two components on the page, the title, and the login box. The login box contains further components such as sub-headings, input fields and a login button. The actual styling for these components are done in the higher fidelity prototypes I create.



Once logged in, the next page the user will be presented with is the quiz selection screen. On this screen there will be a list of quizzes laid out and each quiz will contain information

regarding the quiz, such as the name or category. In addition to this, the page will have a title.

Frame 3

Question
Answer Option
Answer Option
Answer Option
Answer Option

The following page is used to represent each individual question within a quiz. The question page will have the question at the top and each of the answer options underneath.

After completing the low fidelity designs, the next step in the process is to create high fidelity prototypes. These are similar; however, they are much more detailed and contain information about how each of the components within the application will work.





Homepage/Login Screen

Quiz Manager

Login
Username
Password
Login

As you can see, the login screen is now much more detailed and has been designed to closely represent the final product upon release. There is a field where the user can enter their username/password. In addition to this, there is the login button.





Select a Quiz

Quiz Title 1	Easy	5 Questions	
Quiz Title 2	Easy	5 Questions	
Quiz Title 3	Hard	15 Questions	
Quiz Title 4	Medium	10 Questions	

The above shows the quiz selection screen for a restricted user. From here they will only be able to see information about the quiz and have the ability to select one of the quizzes from the list.

Select a Quiz

View Answers





Quiz Title 1	Easy	5 Questions	
Quiz Title 2	Easy	5 Questions	
Quiz Title 3	Hard	15 Questions	
Quiz Title 4	Medium	10 Questions	

This screen will have additional functionality if a View user accesses it. You can see there is now a tab to view the answers to a question. This will appear when one of the quiz boxes is hovered over with the mouse.

Select a Quiz

View Answers

Edit Quiz

Quiz Title 1	Easy	5 Questions	
Quiz Title 2	Easy	5 Questions	
Quiz Title 3	Hard	15 Questions	
Quiz Title 4	Medium	10 Questions	

The final variation on this screen is apparent when an Edit user logs into the system. Here they will see the option to view answers like before, but they will now have the ability to edit the quiz via a similar button tab that appears on the mouse hovering over a quiz option.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Auctor augue mauris augue neque gravida. Tempor orci dapibus ultrices in iaculis. Donec et odio pellentesque diam volutpat?

A) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

B) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

C) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

D) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

[Back](#)

1/15

[Next](#)

The above shows the question answering page with higher detail. You can see how the questions and answers will look on the page. Finally, I have included the current question number out of the total number of questions on the page.

Admin User - Option to Edit Questions and Answers

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Auctor augue mauris augue neque gravida. Tempor orci dapibus ultrices in iaculis. Donec et odio pellentesque diam volutpat?

A) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

B) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

C) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

D) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Back

1/15

Next

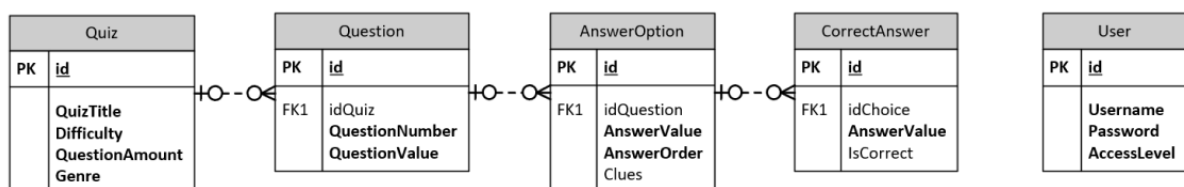
The final variation on this screen is for the Edit user view. Here you are able to see each question and answer has an edit icon present. Once clicking the icon, the user will be able to change the values stored in the question/answer and the quiz will be updated accordingly.

After the designs of the website were finished and I was happy that it met all the criteria, the next step in the process was to model the data.

When planning I decided I would use SQL for the database.

I came up with three different database schemas and these can be seen in the Entity Relationship diagrams I created.

Option 1



This schema shows five different tables and breaks up the data very specifically, so each aspect of the quiz system had its own corresponding table. Each table contains an ID field as the primary key, this is used to uniquely identify each item in the table.

The first table is 'Quiz'. This table solely contains information about the quizzes themselves, such as the title, difficulty of the quiz, number of questions and genre of the quiz.

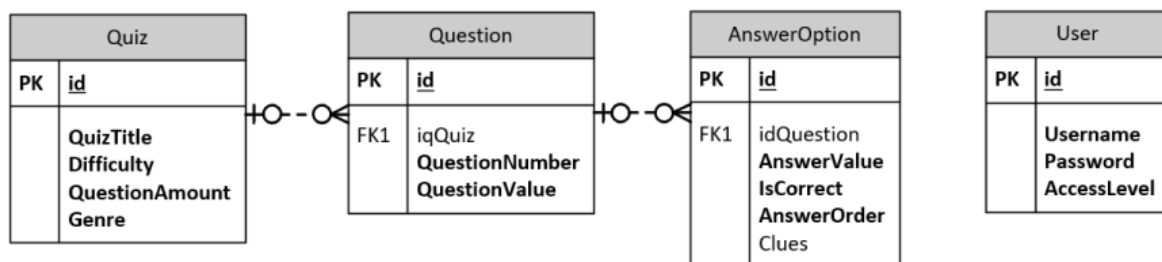
The next table is 'Question' and contains only basic information about each question, such as the 'idQuiz' field. This is the foreign key and is used to create a relationship with the 'Quiz' table. This table also includes the current 'QuestionNumber' field and 'QuestionValue' field. The question value field is the string containing the actual question text.

There is the 'AnswerOption' table. This too contains a foreign key of 'idQuestion' and is used to link to the 'Question' table. In addition, there is the 'AnswerValue'. This is the text containing the question. Finally, there are the 'AnswerOrder' and 'Clues' fields. The 'AnswerOrder' field is a value from 1 to 4 and indicates which order the answers are displayed in.

The fourth table is 'CorrectAnswer'. This table is used to indicate the correct answer for each question and contains a foreign key, 'idChoice' relating to the 'AnswerOption' table. Also, there is the 'AnswerValue' field and a Boolean 'isCorrect' field to flag the correct answer.

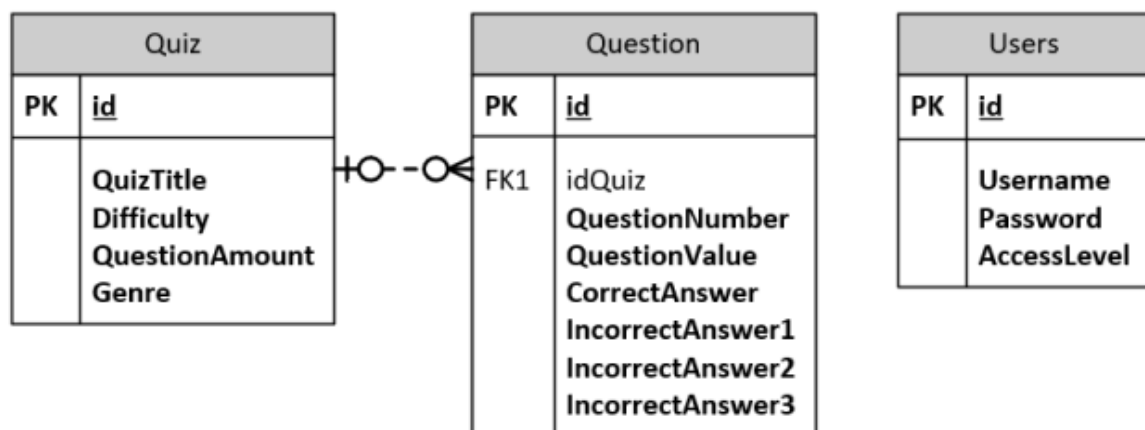
Finally, there is the 'User' table. This will contain information about the user, such as the username, password and access the account has.

Option 2



The next design consists of one less table. Here you can see that I have combined the AnswerOption and CorrectAnswer into one table.

Option 3



The final option is much less normalised than the first two and comprises of half the number of tables as the first. This is the schema I decided to go with and the reason for this was because it reduced the complexity of implementing the system. It is fit for purpose and fully adequate for the intended use and would save time in development having to manage the relationships between all the different tables.

After careful consideration and multiple iterations of the database design, I then decided that it would be simpler and much more fit for purpose to use MongoDB instead of SQL. The reasoning behind my decision to switch is because of the fact that the data does not rely on

any relationship constraints and therefore would be much simpler to implement using a JSON based database rather than a SQL one.

```
{
  "_id": NUMBER,
  "title": STRING,
  "questions":
  [
    {
      "question": STRING,
      "answers":
      [
        STRING,
        STRING,
        STRING
      ]
    }
  ]
}
```

The above is the schema for my data when using MongoDB. It is much simpler and contains only the relevant information to accomplish all the requirements of the brief.

After deciding I was happy with the user interface and database designs it was then time to begin the coding and implantation stage of the project. This will be discussed in the 'Synoptic Project – Implementation Document' report.