Quiz Manager Submission:

Project Evaluation

Prepared by: Linna Trieu, Associate Software Engineer at Paddle

Qualification: Level 4 Software Development Apprenticeship with Makers Academy

Prepared for: BCS, Digital Industries Apprenticeship

Date: September 2020

PROJECT EVALUATION

Overview

This document will assess how the implementation of the Quiz Manager project evaluated against the user requirements. For the purpose of structure, I will do this against the product features/slices, as this will ensure I evaluate against the user requirements.

Fulfilment of product requirements

- A user can login and log out - met ✓
- A user of any kind can view the list of all quizzes - met ✓
- A user of any kind can select a quiz and view the questions - met ✓
- A user of any kind can view answers to questions - met ✓
- A user of any kind can view answers to questions - met ✓
- A restricted user can only see quiz questions - met ✓
- An edit user can add a quiz - met ✓
  - If I had further time, the UX of this feature needs to be improved. At this point a user can submit a new quiz with a list of new questions at the same time. Instead a user has to add a quiz. To navigate back to the quiz list and select the same quiz, to then add a question.Unfortunately I didn't get around to connecting the two creation user flows.
- An edit user can delete a quiz - met ✓
  - Would like to change the hard-deletion of resources into soft-deletion. This would mean we could keep a log of the deletion date, and keep the record in the database. At the moment if a user deletes, all history is removed from the application and database (no audit trail). If a user were to accidentally delete a quiz, there would be no way for us to retrieve the quiz back.
- An edit user can add a question - met ✓

    - Unfortunately I ran out of time to improve this feature, to enable an end user to add multiple-questions at a time.
    - At the moment a user has to add a question one at a time, which could be tedious if a user wanted to add multiple. The current implementation is set up for a user who wants to add 1 or 2 questions to an existing quiz, which is a limitation.
- An edit user can update a question - met ✓
    - I think the User Experience on this feature turned out better than I had anticipated. I wanted to make sure that a user would see the current record of information, before they updated it; and it came out quite well in the update question form because I didn't want the user to toggle between pages, I made sure the current records were displayed to the user in the same form they could use to edit the text.
- An edit user can delete a question - met ✓
    - Same comment as above with deleting a question. Introducing soft deletion of records would provide a better audit trail for administrators of the application.
- An edit user can add an answer - met ✓
- An edit user can update an answer - met ✓
- An edit user can delete an answer - met ✓
    - For the add, update and delete an answer features: I believe my implementation of this functionality was the most flawed. At the begging of planning, I didn't ascertain the user requirements sufficiently or make clear enough assumptions on this. As a result the answers were tied closed with its corresponding question. In the short term, this made life easier in development (regarding one less table in the database, etc). However it meant it was harder to manipulate the answers, as they were not a standalone model but rather a property of the question model. However it would be better to have created the Answers as a standalone model with its own database table.
    - This flaw manifested itself in a couple ways for the end user- a user can't delete an answer on it's own. The only workaround is that a user can 'edit the question/answer' and then choose to just remove the text in the text box. However there is no functionality to delete an answer individually.

- As well as this the indexation of the answers was flawed. At the moment if a user wanted to only provide 3 multiple choice options, all 5 would still be listed (with the last two options with an empty text box).
- This is something I would definitely re-structure and do differently if I could continue developing the Quiz Manager.

Future Product Improvements & Further Development

In addition to those mentioned above, here are additional product improvements for future development:

- It would have been good to add a search field on the All Quizzes homepage.
- Users stay logged in indefinitely. From a security perspective, it would be good to log out users after a certain period of inactivity.
- I hoped to have added images to the website, such as icons for quiz, user or actions (e.g. delete). This could improve the UI. However I ran out of time. But this would have been relatively straightforward to implement using the bootstrap framework.
- I wrote Javascript in the form of jQuery in the blade template pages under a script tag. For separation of concerns, I should extract this into a separate script file.
- The application structure was built to allow for further product features expansion, such as:
    - Extra data displayed against quizzes instead of just the title e.g. total number of questions, difficulty, or category.
    - In the new/edit question form, I would add another HTML form control element, e.g. radio button, which would allow a user to indicate the correct answer; without having to manually type the answer key (bad UX currently).