1 Normalisation and Relational Schema

I have attached diagrams for the normal forms and the schema in dbcw.pdf.

2 Implementation

The implementation of the database flows nicely from the schema, for the most part. Exceptions are:

- Booleans: due to my inexperience with PHP, I found it faster to work with integers (0 and 1) instead of boolean values.

- Foreign key quiz_questions.quiz_id: this is the only FK that doesn't use CONSTRAINT. Instead it uses DELETE, which makes more sense – whenever a quiz is deleted, its questions will be deleted as well. A similar trigger would have been useful for quiz attempts, but I did not have the time to do this.

- Primary key quiz_deletions.id: this would fare very well as a foreign key, but in my implementation the row is deleted from the original table. Given more time, I could have added a quizzes.deleted boolean field that remembers what a deleted quiz used to be.

- Trigger key quiz_delete_log.quiz_id: the database doesn't know who deleted the row, so it provisionally assigns the author of the quiz as its deleter. I programmatically fix this in the website.

Full table in db_dump.sql and on phpMyAdmin database n98211tb (the latter I do not know whether you can access).

3 The application

The website can be found already deployed at **this address**. You need to be connected to the university VPN to access it. All resources were tested using phpMyAdmin and the university dashboard, so if something does not work locally, it will work at the link above.
Initially, the navbar will contain an option to register and an option to log in. If you do not wish to register a new account, the following are already provided (passwords are hashed):

- N98211TB, password "pass" (staff account)

- st1234, password "password" (student account)

- su1234, password "pass" (staff account)

- uid8, password "pass" (staff account)

When logged in, the main page will display previous quiz attempts. Students will also be able to start a new attempt under the Take Quiz tab. Staff will additionally see options to Create Quiz and Modify Quiz.

If any technical issues arise, log out and then log back in.

3.1   Take quiz

Students can only take student-visible quizzes. Staff can take any student-visible quiz, any self-designed quiz, and any quiz not designed by them that can be modified by them. Choosing a quiz from the drop-down menu will prompt a series of quiz questions to be answered by the letter assigned to the correct answer. At the end, a percentage score is displayed, but this is stored as a value between 0 and 1 internally.

If two quizzes share the same name, it will be confusing to distinguish them in the drop-down menu; however, the website keeps track of which quiz was chosen regardless of name and they are displayed in order of creation.

Quiz attempts can be paused and resumed by leaving and returning to the Take Quiz page. They can also be canceled before the last question.

3.2   Create quiz

Staff can create new quizzes by designating a title, number of questions, duration to complete, and specifying whether students can take the quiz and other staff can modify it. Questions are prompted one by one, in a multiple-choice format. Each question must have at least one answer and at most 4, out of which only one is correct.

3.3   Modify quiz

Staff can change quizzes that they have created or that other staff have allowed to be edited by everyone. After choosing a quiz from the drop-down, an initial menu allows to change the title, estimated duration, visibility, and ability for other staff to modify. If a non-author disables the modification option, then their changes to the quiz will be granted, but subsequently only the author of the quiz will be able to modify it or re-enable public modification.

Subsequently, each question will appear on a separate menu, with fields that can be modified. Starting with the last question, new questions can be appended to the quiz. Each old question can be deleted, which will fill the details of the next question. Deleting the last question or a new question will automatically submit the form.

Modifications can be paused and resumed by leaving and returning to the Modify Quiz page. They can also be canceled at any time.

4 Stored procedures and triggers

4.1 Stored procedure: st_names_scores_fail

```
SELECT users.name AS 'Name',
       quiz_attempts.score * 100 AS 'Score percentage'
FROM quiz_attempts
INNER JOIN users
ON quiz_attempts.uid = users.uid
WHERE quiz_attempts.score < 0.4
```

4.2 Trigger: quizzes.quiz_delete_log

```
INSERT INTO
quiz_deletions (quiz_id, uid, date_deleted)
VALUES (OLD.id, OLD.author_uid, NOW())
```
This trigger registers the author as the deleter. The database cannot know who on the website prompted the action. The PHP code accounts for this.