

Innovation Documentation

Innovations in Data Centric Web Application

Name: Tomás Pettit

Student ID: G00419414

Table of Contents

1. Introduction
2. Features
3. Innovation
4. Testing and Validation
5. Conclusion
6. Appendices

1. Introduction

The web application of ExpressJS has been designed to query and update a MySQL database and a MongoDB database. It can create and update using MySQL, and delete databases using MongoDB.

2. Features

- **Home Page (.html):** Consists of 3 links:
 - One to the Students page
 - One to the Grades page
 - One to the Lecturers (MongoDB) page
- **Student Page (.ejs):** Displays details of all Students, includes an Update link for each Student, features an Add Student link and links back to the Home page.
- **Grades Page (.ejs):** Displays each Student's name, the name of each Module studied by a student, and the grade received in each Module, if a student is not studying any Module, only the student's name is shown and links back to the Home page.
- **Lecturers Page (.ejs):** Displays details of all Lecturers (from MongoDB), includes a Delete link for each Lecturer and links back to the Main page.

3. Innovation

1. The home page includes three navigation links to the Students, Grades, and Lecturers pages.
2. The Student page, created with .ejs, features:
 - A table with borders.
 - A link to add students.
 - A link to update students.
 - For adding students:
 - Input fields for Student ID, name, and age.
 - Navigates back to the student page after submission.
 - Validation checks:
 - Student ID must be 4 characters and unique.
 - Name must be a minimum of 2 characters.
 - Age must be 18 or older.
 - For updating students:
 - Similar input validation as for adding students.
3. The Grades page uses .ejs and SQL to join tables, displaying:
 - Student names.
 - Module names.
 - Grades.
 - SQL query: `SELECT s.name as Student, m.name as Module, g.grade as Grade FROM student s LEFT JOIN grade g ON s.sid=g.sid LEFT JOIN module m ON g.mid=m.mid ORDER BY s.name ASC, g.grade ASC;`
4. The Lecturers page uses .ejs to:
 - Display lecturer details.
 - Provide a link to delete data.
 - Error handling and data deletion using `await deleteLecturer(req.params.lid)` and `await mongoDao.deleteLecturer(req.params.lid)` respectively.

4. Testing and Validation

1. **Student Page:** The table has been completely finished and as for the rest of the links. For the add student link that I clicked, I got four errors validations complete & creating a student has been completed, and then the update student link, I got four errors validations complete & updating a student has been completed. The Student Page has been validated for adding and updating students tested and completed.
2. **Grade Page:** The table has been completely finished and as for the rest of the links. And to check that I placed them in order of student name and grades. Table and links of the Grade Page has been tested and validated.
3. **Lecture Page:** The table has been completely finished and as for the rest of the links. I clicked on the delete link in one of those data, it worked, the data has been deleted from the table. And for the error message, since I clicked the delete link, it has given me an error message to make sure that one of those data cannot be deleted from the table. Data deletion and error handling of the Lecturer has been tested and validated.

5. Conclusion

1. What is the best way to get an error message on the lecturer page?
2. Can I create data again on the lecturer page after deletion?
3. Can I delete data on the student page after adding a student?
4. Should the tables for students, grades, and lecturers use borders for simplicity?

6. Appendices

Data Tables

Data tables are essential tools for organizing and displaying data in a structured and accessible manner. In your web application, they can be used to present information clearly and efficiently.

Using EJS (Embedded JavaScript)

EJS (Embedded JavaScript) is a simple templating language that lets you generate HTML markup with plain JavaScript. It is often used in web applications to produce dynamic web pages.