



International Islamic University Chittagong

Final Project Lab Report

Course Title: CSE3532 – Tools and Technology for
Internet Programming

Instructor: Bibi Sara Karimullah

Student Name: Tasnim Nahar Raisa

Student ID: C231526

1. Introduction

This report details the planning, development, and execution of my final semester project: a full-stack educational website titled “Learn with Raisa.” It simulates a real-life digital learning platform, allowing users to sign up, browse academic resources, enroll in courses, and purchase e-books.

Throughout the semester, I gradually built up my understanding of web development through coursework, small exercises, and collaborative assignments. This project was an opportunity to put all those skills into one integrated application. Initially intended to be separate frontend and backend submissions, the scope was unified into a complete standalone project by instructor approval.

2. Objective of the Project

The main goal was to create a fully functional, visually appealing educational platform where:

- Users can register, log in, and maintain a session.
- Courses and eBooks are displayed interactively.
- Students can enroll in courses via button-triggered backend logic.
- E-books are protected by a payment/promo code access system.
- All user actions are securely logged and processed.

This project bridges theory and practice—applying database design, authentication logic, server-side routing, and client-side layout styling.

3. Technologies Used

Frontend:

- HTML5 & CSS3 (with embedded inline styles for dynamic elements)
- Swiper.js for carousels
- Google Fonts + custom font integration (Steph font)

Backend:

- Node.js for runtime
- Express.js framework
- SQLite3 database engine
- Express-session for user session handling
- Body-parser for form data handling

Environment:

- Visual Studio Code for coding
- Terminal for running Node scripts and managing dependencies
- Google Chrome for browser testing

4. Features Implemented

4.1 User Authentication System

- Signup form with validations for email, password confirmation, and error messaging.
- Login form with validation, session generation, and conditional redirects.
- Session persistence using cookies (via express-session).
- Logout functionality that clears session and redirects back to the homepage.

4.2 Homepage Courses and E-Books

- Interactive sections with swiper.js sliders.
- Course cards include images, descriptions, and “View & Enroll” buttons.
- E-book cards include previews and “Buy & Read” buttons.
- Designed to simulate user flow through visual CTA buttons.

4.3 File & Resource Access

- Courses trigger a future-enroll feature, tied to the user session.
- E-books are hosted securely; promo code or login session is required for access.

4.4 Aesthetic & UX

- Clean, minimal interface using pastel tones and custom typography.
- Fonts used: Poppins and Steph (custom .ttf files)
- Responsive layout via flexible containers and adaptive sizing

5. Backend & Database Structure

5.1 SQLite3 Database Design

- users.db stores all user credentials.
- users table: id, email, password
- (Planned) purchases.db to track e-book and course transactions

5.2 Routing and Static File Serving

- All HTML, CSS, fonts, and images are placed in the root directory (no public/ folder used).
- `express.static(__dirname)` allows direct access to frontend resources.
- `server.js` handles routing, login/post requests, and conditional access logic.

5.3 setup.js

- Initializes the database with necessary tables
- Automatically runs once before server launch

6. Folder & File Structure

```
Raisa_teaches_5DF_raisa/  
├─ index_C231526.html  
├─ login.html  
├─ signup.html
```

```
├─ styles_C231526.css
├─ fonts/
├─ images/
├─ ebooks/
|   ├─ olevel_maths_d.pdf
|   ├─ olevel_chemistry.pdf
|   └─ alevel_physics.pdf
├─ server.js
├─ setup.js
├─ package.json
└─ users.db
```

7. Challenges Faced

- Session Management: Implementing persistent login across multiple page requests required learning how express-session works behind the scenes.
- Static Routing Issues: When moving files outside the public/ directory, I had to carefully reconfigure Express to correctly serve static resources.
- Validation & UX Feedback: Writing JavaScript logic to catch user errors (e.g., mismatched passwords) and reflecting that cleanly in the frontend.
- Visual Integration: Matching the new backend-driven pages to the visual standard of the HTML-only homepage was difficult but necessary.
- Time Management: Building the database, server routes, and secure access flows within tight constraints required careful planning.

8. Learning Outcomes

- Developed confidence in building a real-world full-stack application.
- Learned backend database handling with SQLite and Express.
- Experienced working with user sessions and access control.
- Strengthened frontend design practices using carousels and media responsiveness.
- Improved project debugging and version control skills.

9. Conclusion

This project marks a significant milestone in my development journey. I've successfully integrated all aspects of a full-stack application, from form validation and session-based login to file protection and custom design. It represents my growth over the semester, combining foundational skills with advanced execution under pressure.

Although a few backend modules like course enrollment and eBook payment are still being completed, the structure and flow of the website are strong and ready for extension. This project will serve as a stepping stone toward building more advanced web applications.

[End of Report]