

ITSC 4155 Project Design Document

1. Project Overview

This project is a full-stack web application designed to implement a basic login system using React for the frontend, Node.js with Express for the backend, and MySQL as the database. The application enables users to sign in with their email and password, storing credentials in a secure database.

2. Team Roles

- **Scrum Master:** Shelton Latham
 - **Product Owner:** Robert Broughton
 - **Developers:** Peyton Hopkins, DeCristo Franceschini, Ryan Jones, Maisha Alam Meem
-

3. Technologies Used

- **Frontend:** React
 - **Backend:** Node.js, Express
 - **Database:** MySQL
 - **Other Libraries:** CORS, MySQL (npm package)
 - **Version Control:** Git
-

4. System Architecture

Client (React) <--> Server (Express/Node.js) <--> MySQL Database

The frontend sends requests (e.g., login) to the backend via HTTP. The backend validates user credentials by querying the MySQL database and returns appropriate responses to the frontend.

5. Database Design

Database Name: `ITSC4155`

Table: login

Field	Type
email	VARCHAR(255)
password	VARCHAR(255)

6. Frontend Design

The React frontend includes input forms for email and password. It communicates with the backend using HTTP requests (likely through Axios or Fetch). The UI resides in the `react-app/src` directory.

7. Backend Design

The backend is powered by Node.js and Express. It defines API endpoints to interact with the MySQL database. The main backend logic is located in `database.js` which connects to the database and handles query logic for login functionality.

8. Setup & Deployment Instructions

Database Setup:

1. Install MySQL
2. Run the following SQL commands:

```
CREATE DATABASE ITSC4155;  
USE ITSC4155;  
CREATE TABLE login (email VARCHAR(255), password VARCHAR(255));
```

Backend Setup:

1. Run: `npm install mysql cors express`
2. Navigate to `react-app/src`
3. Run: `node database.js`

Frontend Setup:

1. Install Node.js
2. Clone the repository
3. Run:

```
npm install
npm install react-app
cd react-app
npm start
```

9. Future Enhancements

- Add user registration functionality
- Implement password hashing for security
- Use environment variables for DB credentials
- Add session handling or JWT authentication
- Improve frontend UI with form validation and error handling