ITECH3108 Assignment 2 – Report

Paper link -Link sharing website

Student Name:Vikas kundal
Student_ID:30413540

System Architecture

This project was developed to create a full-stack web application named Paper Link. This application was intended for users who wanted to share, rate, and manage links that pertained to paper. This application was built in a client-server architecture where the application was divided into the frontend, backend, and database parts.

For the frontend part of this application, HTML, CSS, and JavaScript were used. This part of the application was used for user interaction and dynamic content updates. For the backend part of this application, the Deno runtime environment and Oak framework were used. This part of the application was used for RESTful API implementation. For the database part of this application, a PostgreSQL relational database was used. This application used the Fetch API for the frontend and backend interaction, where JSON was used for data exchange. JSON Web Tokens were used for authentication.

API and Database Design

The backend has implemented RESTful API endpoints for user authentication, creating links, rating them, hiding them, and retrieving favorites. HTTP methods have been correctly utilized for each action as per RESTful API design.

The database schema has four tables: members, links, ratings, and hidden_links. Foreign key constraints have been implemented for maintaining data integrity. Passwords for users are stored securely via bcrypt hashing. JWT has been implemented for authenticating sensitive operations.

Feature Implemented

All of the functional and non-functional requirements mentioned in the assignment have been implemented.

- Users are able to sign up and log in properly
- Users are able to add new links after they are logged in
- All users are able to see links on the front page
- Links are able to be rated positively and negatively
- The Paper Points system works by rewarding/penalizing users according to ratings
- Users are able to see their favourite links that they rated positively
- Users are able to hide links and see them on a different page
- Sorting by most recent and highest rated links is supported
- All of these actions are performed without reloading the page

The application has a consistent visual design and a clear and consistent interface.

Database Content Acknowledgement

All the data used in the database, including users, links, and ratings, was created specifically for this assignment. This data is original and used only for demonstration and testing purposes.

Help received and reference

Throughout the course of developing this particular project, multiple publicly accessible learning resources have been used as a reference point for developing an understanding of, as well as implementing, the concepts and ideas that have gone into creating this particular project. Some of these publicly accessible learning resources include MDN Web Docs, which have been used as a reference point for understanding and implementing concepts related to HTML, CSS, JavaScript, and the Fetch API. Stack Overflow has also been used as a reference point for troubleshooting any particular issues that arose throughout the course

of developing and implementing the concepts and ideas that have gone into creating this particular project. In addition, other publicly accessible web-related resources have been used as a reference point for developing an understanding of concepts related to RESTful API design, structuring of projects, and backend database integration.

Conclusion

This assignment shows how a secure RESTful full-stack web application can be built using the Deno framework and PostgreSQL database. This application meets all requirements of this assignment and follows all software engineering principles.

My video demonstration link: