SOW Task-1 Documentation – Term Page Clone Application

SOW Task-1 Overview

This project is a full-stack Term Page Clone application built with React for the frontend and Fastify + PostgreSQL for the backend. The repository is separated into two directories: frontend and backend for better maintainability and modularity.

The application supports multi-language terms and dynamically renders content fetched from the backend API based on the selected language. It is fully deployed with the frontend hosted on Vercel, the backend on Render, and the database (PostgreSQL 17) on Render.

Frontend

- React: 19.1.1
- React DOM: 19.1.1
- Vite: 7.1.2
- ESLint: 9.33.0
- Globals: 16.3.0
- dotenv: 17.2.2

Backend

- Fastify: 5.5.0
- Fastify CORS / @fastify/cors: 6.0.3 / 11.1.0
- Sequelize: 6.37.7
- PostgreSQL: 17 (hosted on Render)
- pg / pg-hstore: 8.16.3 / 2.3.4
- Nodemon: 3.1.10
- dotenv: 17.2.2

Environment & Deployment

- Node.js (ES Module support enabled via type: "module")
- Frontend deployed on Vercel   https://term-page-frontend.vercel.app/
- Backend deployed on Render   https://term-page-clone-backend.onrender.com/
- PostgreSQL database hosted on Render
- Frontend build tool: Vite

Note: Due to my free limited account on Render, the backend or database may take longer to respond when idle. Free instances spin down with inactivity, which can delay requests by 50 seconds or more.