Developer Documentation

1. How to Obtain the Source Code

To clone and set up the project locally, follow these steps:

1. Clone the repository:

> *git clone https://github.com/CS319-24-FA/S2-T5-guidex.git*

---

2. The Layout of Your Directory Structure

Backend

- **app/**
    - **routers/** - Contains API route handlers.
    - **utils/** - Utility functions and reusable backend code.
    - database.py - Database connection setup.
    - main.py - Entry point to start the backend server.
    - models.py - Defines database models or schemas.
    - .dockerignore - Excludes unnecessary files from the Docker image.
    - .env - Stores backend environment variables.
    - .gitignore - Ignores specific backend files in version control.
    - Dockerfile - Docker setup for containerizing the backend.
    - requirements.txt - Backend dependencies list.

Frontend

- **public/**
    - **assets/** - Static assets like images.
    - **json/** - JSON files for application data.
- **src/**
    - **components/** - Reusable React components.
    - **layouts/** - Layout structures for pages.
    - **pages/** - Individual pages for the application.
    - **utils/** - Utility functions for frontend logic.
    - App.tsx - Main React app component.
    - main.tsx - Entry point for the React frontend.
    - index.css - Global CSS styles.
    - vite-env.d.ts - Vite environment configurations.
- .dockerignore - Excludes unnecessary frontend files in Docker builds.
- .gitignore - Ignores specific frontend files in version control.
- Dockerfile - Docker setup for the frontend.
- eslint.config.js - ESLint configuration for linting.
- index.html - Main HTML template for the React app.

- package.json - Node.js metadata, dependencies, and scripts.
- postcss.config.js - PostCSS configuration for styling.
- tailwind.config.js - Tailwind CSS configuration.
- tsconfig.app.json - TypeScript configuration.

---

3. How to Build the Software

Backend (Python)

1. Ensure all dependencies are installed:

   > pip install -r requirements.txt

2. Run the server:

   > python main.py

3. The backend server will start at http://localhost:8000 (default).

Frontend (React + TypeScript)

1. Install dependencies:

   > npm install

2. Run the development server:

   > npm run dev

3. Access the app at http://localhost:5173 (default Vite port).

Docker Build

For containerized deployment, you can build and run the backend and frontend using Docker.

   > docker compose up

Notes for Developers

- Both backend and frontend can be run both together and independently.
- Use **.env** files to configure environment variables.