fsd-demo-FRS

December 30, 2024

0.1 Functional Requirements Specification (FRS) for the full-stack demo Flask and SQLite application (fsd-demo):

---

1 Functional Requirements Specification (FRS)

1.1 1. Introduction

1.1.1 1.1 Purpose

The purpose of this document is to outline the functional requirements for a simple full-stack application based on Flask and SQLite. This application will manage user data, including creating, reading, updating, and deleting user information.

1.1.2 1.2 Scope

This document details the functionalities required for the application, which will be used to manage a list of users. The application will provide a web interface for performing CRUD operations on user data.

1.2 2. Functional Requirements

1.2.1 2.1 User Management

2.1.1 Create User

- **Description**: The system shall allow the admin to add a new user to the database.
- **Inputs**: User name, email, and age.
- **Process**: The admin enters the user details into a form and submits it. The system validates the input and inserts the new user into the database.
- **Outputs**: A confirmation message indicating the user has been added successfully, and the updated list of users.

2.1.2 Read Users

- **Description**: The system shall display a list of all users stored in the database.
- **Inputs**: None.
- **Process**: The system retrieves all user records from the database and displays them on the home page.
- **Outputs**: A list of users with their details (ID, name, email, age).

2.1.3 Update User

- **Description**: The system shall allow the admin to update the details of an existing user.
- **Inputs**: User ID, new name, new email, and new age.
- **Process**: The admin selects a user to update, enters the new details, and submits the form. The system validates the input and updates the user record in the database.
- **Outputs**: A confirmation message indicating the user has been updated successfully, and the updated list of users.

2.1.4 Delete User

- **Description**: The system shall allow the admin to delete a user from the database.
- **Inputs**: User ID.
- **Process**: The admin selects a user to delete and confirms the action. The system deletes the user record from the database.
- **Outputs**: A confirmation message indicating the user has been deleted successfully, and the updated list of users.

1.3 3. Non-Functional Requirements

1.3.1 3.1 Performance Requirements

- The system should respond to user actions within 2 seconds.

1.3.2 3.2 Security Requirements

- The system should validate all inputs to prevent SQL injection attacks.

1.3.3 3.3 Usability Requirements

- The user interface should be intuitive and easy to navigate.

1.4 4. External Interface Requirements

1.4.1 4.1 User Interfaces

- The home page shall display a list of users.
- The update page shall provide a form to update user information.
- The delete functionality shall be accessible via a button next to each user.

1.4.2 4.2 Hardware Interfaces

- The application shall run on standard server hardware.

1.4.3 4.3 Software Interfaces

- The application shall interact with the SQLite database using SQL queries.

1.4.4 4.4 Communications Interfaces

- The application shall use HTTP/HTTPS for communication between the client and server.
