Grocery Store Report

Author :

- Name : Akshay Bhalchandra Sonaje
- Roll No : 21f1006436
- Email **:** 21f1006436@ds.study.iitm.ac.in
- About me : I am pursuing a Bachelor of Engineering from SPPU university and am currently in my final year. Along with it I am also pursuing a BS in Data Science and Applications from IIT Madras and am in the diploma level.

Description :

The Grocery Store web application is a multi-user platform that facilitates the buying and selling of grocery products. It aims to provide a convenient and efficient shopping experience for users while enabling store managers to list and manage their products.

Technologies used:

- **Python:** Develop the controllers and server as the host programming language for the app.
- **Flask**: A lightweight and flexible Python web framework for building the backend of the application, handling routing, and request handling.
- **SQLAlchemy**: An ORM library for Python that simplifies database interactions by allowing developers to work with databases using Python objects.
- **SQLite**: A self-contained and serverless database management system used as the application's database due to its simplicity.
- **HTML, CSS :** : Core web technologies for structuring pages, handling styling, and adding interactivity to the user interface.
- **Bootstrap**: A front-end framework providing pre-designed components for responsive web design and a visually appealing user interface.
- **Flask-Migrate**: An extension for Flask that simplifies database migrations and schema changes.

Database Schema:

User: Fields: id (primary key), username, address, mobile, password,first_order

Relationships: One-to-one with Cart (user.cart)

Cart: Fields: id (primary key), user_id (foreign key)

Relationships: One-to-many with Product (cart.cart_products),One-to-many with User (cart.user),One-to-many with Product (cart.cart_products)

Product: Fields: id (primary key), name, price, stock, section_id (foreign key), manager_id (foreign key), quantity, description, frequency

Relationships: Many-to-one with Section (product.section), Many-to-one with Manager (product.manager)

Back-Reference: One-to-many with Cart (product.carts)

Section: Fields: id (primary key), name

Relationships: One-to-many with Product (section.product)

Manager: Fields: id (primary key), managername, store, password

Relationships: One-to-many with Product (manager.products)

Architecture and Features :

The application is built with the standard MVC architecture, where SQLite serves as the Model for data storage, Python with Flask handles the Controller for managing logic and request handling, and HTML/CSS/Bootstrap constitute the View for a visually appealing user interface.

User Registration and Authentication: Allows users to create accounts and log in, enabling personalised shopping experiences and order tracking.

Product Browsing and Searching: Users can browse grocery products categorised into different sections, also can view and buy frequently bought products simplifying product discovery and selection.

Checkout : Users can place the order for products which are in the cart and get 20% discount on first order.

Manager Product Listing: Store managers can list grocery products for selling, specifying prices, stock availability, and descriptions.

Section Management: Enables managers to organise products into various sections. A manager can Add, remove and edit a section.

Products Management: Managers can add, edit and remove grocery products to their store inventory, specifying product names, prices, stock availability, and descriptions.

Video Link : click here