Author
Name-Austin Dious
Roll number-21f3001569
Student Email- 21f3001569@ds.study.iitm.ac.in

Description

This project involves developing a web-based grocery store application using Flask and HTML. Users can search for products, view product details with prices and add desired items to their cart. The project includes implementing user authentication, displaying products by category, enabling search functionality, and allowing users to add products to their cart for purchase.

Technologies used

Flask: Web framework used to develop the backend of the application, handle routing, and manage data interactions.
Jinja2: Templating engine integrated with Flask for rendering dynamic content in HTML templates.
SQLAlchemy: Python SQL toolkit and Object-Relational Mapping (ORM) library for database interactions.
Flask-SQLAlchemy: Flask extension that simplifies database integration and management using SQLAlchemy.
HTML/CSS: Frontend technologies for creating the user interface and styling the web pages.
Bootstrap: CSS framework for responsive design and pre-styled UI components.
SQLite: Embedded relational database used for storing product and user data.

DB Schema Design

Users:
Columns: id (Primary Key), username, email, password_hash
Constraints: Unique constraint on username and email
Purpose: Store user account information, including login credentials.

Categories:
Columns: id (Primary Key), name
Purpose: Store different categories of products, such as "Fruits", "Vegetables", "Dairy", etc.

Products:
Columns: id (Primary Key), name, price, quantity, image_url, category_id (Foreign Key)
Constraints: Foreign key reference to Categories table
Purpose: Store details of individual products, including their name, price, quantity in stock, image URL, and category.

Cart:
Columns: id (Primary Key), user_id (Foreign Key)
Constraints: Foreign key reference to Users table
Purpose: Represent a user's shopping cart.

CartItems:
Columns: id (Primary Key), cart_id (Foreign Key), product_id (Foreign Key), quantity
Constraints: Foreign key references to Cart and Products tables
Purpose: Store the items added to a user's shopping cart, along with the quantity of each item.

Orders:
Columns: id (Primary Key), user_id (Foreign Key), order_date, total_price
Constraints: Foreign key reference to Users table
Purpose: Store information about user orders, including the order date and total price.

OrderItems:
Columns: id (Primary Key), order_id (Foreign Key), product_id (Foreign Key), quantity
Constraints: Foreign key references to Orders and Products tables
Purpose: Store the items included in an order, along with the quantity of each item.

Architecture and Features

The grocery shopping web application follows a structured architecture that adheres to the Model-View-Controller (MVC) design pattern. The project is organized into different components, each serving a specific role. The controllers handle the logic and routing of the application. They interact with the models, which represent the data and business logic, including the database interactions. The views consist of HTML templates responsible for rendering the user interface. These templates are populated with dynamic data from the controllers.

The application includes several features that enhance the shopping experience. The user authentication feature allows users to register and log in. The product catalog displays all available products. The search functionality enables users to find products by name. Users can view detailed product information, including an image, name, category, and price. They can add products to their shopping cart and adjust quantities. The cart maintains a summary of selected items.

Video

https://drive.google.com/file/d/1AMS7qtH3xf75vPruxM8IZXReUk4oA4cA/view?usp=drive_link