Mad 1 Project Report May Term -2023

Author: -

Amit Kumar

Roll no: - 21f3002445

Email: - 21f3002445@ds.study.iitm.ac.in

About me:

Hey, currently I am student of B.S. in Data Science from IIT Madras. I am interested in data science field. Through this course I have also knowledge in website development using Flask and jinja 2 framework. I thrive on engaging in competitions and technical hackathons, as they allow me to showcase my skills and learn from other like-minded individuals.

Description:

In this project, my role is developing a robust web application for a Grocery Store using Flask and integrating database functionalities with SQL Alchemy. I am responsible for creating user and admin authentication systems, implementing CRUD operations for categories and products, ensuring data integrity, designing responsive front-end interfaces and style through using CSS.

Technologies used:

1. Flask: flexible framework for building Web applications.
2. Flask -SQL Alchemy: these technologies use for making simplifies database interaction provide efficient data handling and ensure the data integrity.
3. Flask -login: this technology helps in for user authentication, authorization, and session management.it is also help in find out the current user.
4. Html-CSS: for structuring and designing frontend of our website.
5. Matplotlib: this technology help in generating the graph for visual understanding.

DB Schema Design:

Architecture and Features

Root Directory: The root directory contains the main application file app.py, an instance folder for the database file, and other project-related files like myenv, requirements.txt, and the static folder for static assets.

Groce Directory: This directory serves as a package and contains the core functionality of the application.

__init__.py: Marks the directory as a Python package and initialization of all package .

models.py: Defines the SQLAlchemy models for database tables like User, Category, Product, manager and add cart.

routes.py: Defines the different routes and their corresponding view functions.

static/: groce static assets such as images.

templates/: Contains HTML templates for different views, such as admin_dashboard.html, admin_login.html, base.html, and others.

The project follows a modular structure where each component has its dedicated file or directory, promoting code organization and maintainability. The groce directory acts as the core of the application, containing models, routes, and static assets. The templates directory holds the HTML templates used for rendering different pages.

Features:

1. Home Page: Displays all products and categories.
2. User Registration: Allows new users to register with a registration form.
3. User Login: Users can log in to their accounts.
4. User Logout: Logged-in users can log out.
5. Add to Cart: Users can add products to their cart.

6. Buy : Users can buy the product
7. Cart Details: Displays and updates the user's cart.
8. Profile : Displays and edits user profile details.
9. Order Confirmation: Displays order confirmation message.
10. Admin Login: Special login for admin accounts.
11. Admin Dashboard: Admin-only view with product, and category details.
12. Manage Categories: Admins can add, update, and delete categories.
13. Manage Products: Admins can add, update, and delete products.
14. Admin can see the current product product data .

Video link:
https://drive.google.com/file/d/1EQ91buICpjFNT9n9jBgqZFfS71VF8pNo/view?usp=sharing