

Design and Development Synopsis: E-Commerce Flask Web Application

Xin Feng

Render deployment URL: <https://web-programming-1ws4.onrender.com>

This project, crafted for the CS551P web programming course, serves as a foundational e-commerce Flask web interface. It is designed to display essential e-commerce functionalities, such as showcasing an index page, listing products, presenting product details, and managing customer data. Additionally, the application integrates a CSV import feature, enhancing data entry efficiency and supporting batch updates.

Design Considerations

The Flask micro-framework was selected for its simplicity and rapid development capabilities. Its lightweight structure is ideal for educational projects where the focus is on learning and application of core concepts rather than full-scale e-commerce operations. The application's design follows Flask's conventional setup, with a clear division between the front-end templates and the back-end logic, facilitating ease of understanding and maintenance.

Functionality Overview

- Index Page: Serves as the launch point for user interaction.
- Product Listing: Displays a catalog of products with an emphasis on readability and accessibility.
- Product Details: Offers a detailed view for each product, providing more in-depth information.
- Customer Management: Allows for the visualization and handling of customer data within the platform.
- Pagination: Implemented to enhance user experience when navigating through extensive lists of products or customers.
- CSV Import: Streamlines the process of data integration, permitting the import of product and customer information in bulk.

Development Approach

Development was structured around best practices, such as using Git for version control, which facilitated a collaborative and iterative approach. The project's history, as documented in `'git-log.txt'`, reflects a development process focused on incremental growth and feature integration. To maintain an isolated development environment, a virtual environment was utilized, as indicated by the ``.venv`` directory, ensuring consistent dependencies via `'requirements.txt'`.