GitHub Link: <https://github.com/dallonjarman/CS-440-Inventory-App>  
  
YouTube Link: <https://youtu.be/W8K-nC6-fGw>

**System Description**  
This system is an online web shop that provides customers with a platform to buy products, register and log in, and order products with ease. It is a one-stop shop for viewing various products, providing a hassle-free and secure shopping experience as well as providing customers with the ability to track their orders with ease.

**1.3 Technology Stack Justification**  
We built the app in Flask, a minimal and yet agile Python web framework that is suitable for rapid prototyping and development. Flask's ease of use and extensive ecosystem enabled easy inclusion of SQLAlchemy for database operations and Jinja for templates. SQLite as a database was chosen because it is simple, especially at development time, and switching to a heavier database later on is straightforward if necessary. This combination of technology ensures the application to be secure, scalable, and maintainable.

**1.4 Architectural Decisions and Rationale**  
The app is organized in a modular way decoupling concerns: initialization and configuration are made in the init.py file, database models are kept in their respective modules, and routes are defined in one routes file. Templates and static assets are kept in their own directories to promote a clear separation between front-end presentation and back-end logic. This approach enhances readability, makes debugging easier, and allows for the simple addition of new features or modules in the future.

**UML Diagrams:**

A screenshot of a diagram

AI-generated content may be incorrect.

Figure 1: Class Diagram for the Online Shop System

A diagram of a customer service

AI-generated content may be incorrect.

Figure 2: Sequence Diagram for the Login Process

A diagram of a software system

AI-generated content may be incorrect.

Figure 3: Activity Diagram for the Order Placement Process

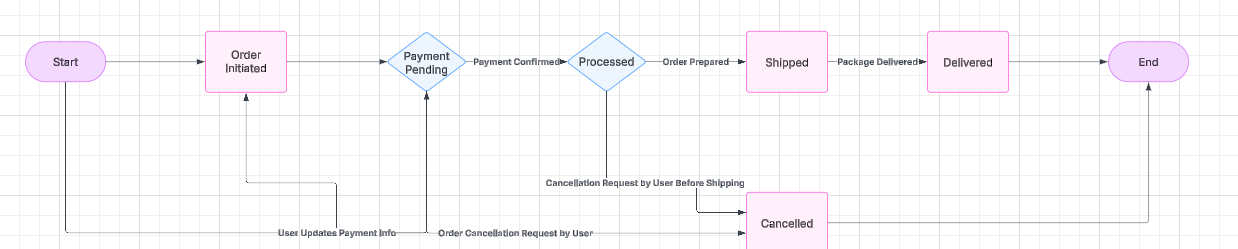


Figure 4: Class Diagram for Order States

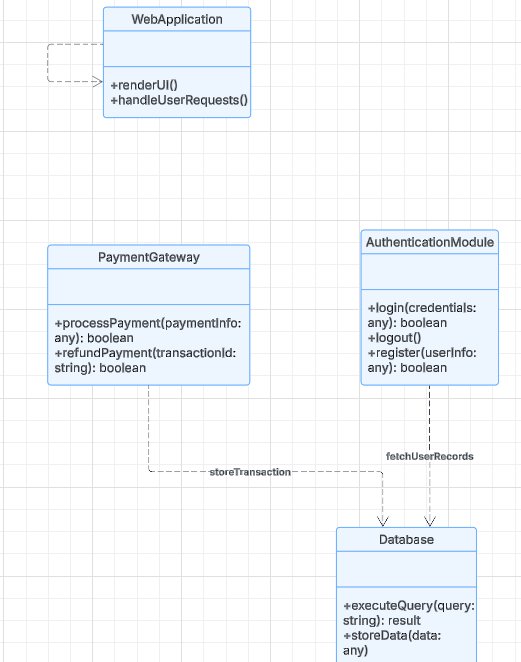


Figure 5: Sequence Diagram of the System Architecture



Figure 6: Activity Diagram for the Application Environment

**Use Case Diagram**

**A diagram of a system

AI-generated content may be incorrect.**

Figure 7: Register User

A diagram of a program

AI-generated content may be incorrect.

Figure 8: Logging in

A diagram of a product

AI-generated content may be incorrect.

Figure 9: View Product

A diagram of a system

AI-generated content may be incorrect.

Figure 10: Place Order

A diagram of a system

AI-generated content may be incorrect.

Figure 11: View Order

A diagram of a product

AI-generated content may be incorrect.

Figure 12: Manage Product

4.1

When asked about a hardcoded secret issue, the AI gave me the following.

AI Response:

* **Hardcoding secrets (like SECRET\_KEY in app/\_\_init\_\_.py)** means anyone who gains access to the codebase can see and misuse them.
* This is a serious security flaw, as exposed keys can be used for attacks like **session hijacking, CSRF attacks, and unauthorized database access**.

eess

We agree with the AI on this issue. We did address this issue by removing the config,py file and instead changing the secret key to a .env.

4.2

On a scale of 0-10 for the diagram based AI review, we would say we improved about a 2. The AI did not give us much information or provide us with much help that we felt was important for improving our diagrams. We did not completely agree with the AI and felt it was hallucinating when giving us comments on out diagrams.

On a scale of 0-10 for improving our code. We would say the AI helped us about an 8. It gave us much needed insight on what we were missing and what we could do to better secure our code. However, we felt the comments from the AI were outside the scope of this project and it was a lot more security based rather than bug based or feature based.

On a scale of 0 to 10 for the AI review, we would give it around a three. You don’t get the human-to-human feedback that would really enhance our project. Instead, the AI is giving us feedback from a prompt and lacks the skills an instructor would provide. It was okay in what it told us but lacked the human part and felt repetitive as if the AI was trying to meet a word quota.

On a scale of 0 to 10 for how much we learned with the AI review, we would give it a 5. It is great insight for what we could work on in terms of security but in terms of UI or features from the AI, it felt lack luster. It felt repetitive and again was missing a human touch from a professor that could really tell us what we needed to work on.

On a scale of 0 to 10 for how much we learned about the capacities and limitations of AI based on this assignment, we would give it a 6. AI is very capable for a real world environment but sitting inside of a learning environment where a student would require a professor to further explain their thinking, the AI is missing a lot of those critical thinking skill. However, when it comes to security issues and ensuring the app is ready for product, it does a great job. Overall, AI is great for real world uses and lacks the needed skills in a learning environment.