

Smart-Cart

SmartCart is an online shopping platform concept designed to deliver a smooth, intuitive shopping experience for everyday users while also providing store managers with the tools needed to oversee products and customer orders. The application aims to strike a balance between ease of use and essential eCommerce functionality, with the flexibility for teams to enhance it further with additional features.

Core Capabilities (Required)

The following capabilities form the foundation of SmartCart and must be implemented to ensure a fully usable and testable experience.

User Access and Role-based Experience

The platform will support two distinct types of users. Regular customers will be able to browse and shop for products, while store administrators will have access to tools for managing product listings and order fulfillment. Each user type will have access to specific areas of the application appropriate to their needs, ensuring a focused and secure experience for both.

Product Discovery and Browsing

Customers visiting the platform should be greeted with a clean and well-organized display of available products. Each product entry will feature basic details, such as the name, pricing information, and an image to help shoppers make quick decisions. To help customers navigate through the selection more easily, there should be simple ways to locate items — whether by typing in keywords to search or by using filters or sorting options to arrange the product list based on factors like price or category.

Shopping Cart Management

Once customers find items they wish to purchase, they will be able to add these to a personal shopping cart. The cart acts as a temporary collection space where users can review selected items before placing an order. Customers should be able to view all items added, adjust quantities, remove items, and see a running total of their current selections.

Checkout Process

After finalizing their product selections, customers will proceed to a checkout area designed to summarize their order and collect essential information needed to complete the transaction. This process includes an order summary, a form for personal and delivery

information, and an automatic calculation of delivery costs based on location details. A confirmation message or screen will acknowledge successful order submission.

Store and Order Management for Administrators

To keep the store running smoothly, administrators will have access to a management interface where they can add, update, and control the availability of products. They will track stock levels and incoming customer orders, updating statuses to indicate when an order has been completed and processed.

Optional Enhancements

Teams are encouraged to introduce additional improvements to enhance the shopping experience. While these are not required, selecting one or two of the following ideas could help set your implementation apart:

- Offer customers the ability to save favorite products for future purchase consideration.
- Provide a setting for switching between different display themes, such as a light or dark background.
- Display product or stock updates in real-time without requiring users to refresh the page.
- Suggest related products or recommendations based on what customers have viewed or added to their cart.
- Allow users to search for products by speaking into their device's microphone.
- Include fun visual effects or feedback, such as celebratory animations when a customer places an order.
- Enable customers to install the application directly on their device as a web-based app.
- Improve the platform's accessibility so it can be easily used by people relying on assistive technologies.
- Offer language selection options so users can interact with the platform in their preferred language.
- Keep shopping cart contents saved between sessions, even if the user leaves the site and returns later.
- Introduce simple testing mechanisms to confirm that key interactions and workflows function as expected.

Expected Outcomes

- A working system in a git repo.
- A small demo video showcasing all features.
- 5-minute presentation on how you incorporated [go-vibe](#) in your development process and the learnings from it.
- Documentation – Architecture, Developer guide, Readme, Prompt-logs etc.