**D. Y. Patil Technical Campus,**

**Faculty of Engineering & Faculty of Management, Talsande**

**Department of Computer Science & Engineering**

**Course Name: Domain Specific Mini Projects**

**Class: TY Division: A Batch:T3**

**Team Members:**

|  |  |
| --- | --- |
| **Roll No.** | **Name of Student** |
| 3057 | Vishal Vitthal Lohar |
| 3059 | Abhishek Sambhaji Patil |
| 3060 | Purushottam Santosh Patil |
| 3067 | Prasad Sanjay Shinde |
| 3071 | Suhas Shivanand Uttare |
| 3073 | Omkar Sanjay Yadav |

**Project Guide Name: B. S. Jadhav**

**SRS Report**

**Project Title: Shopping Cart**

**Introduction:**

On-Line Shopping System” is a web-based project which is made for remote-shopping or shopping through Internet. As the technology is being advanced the way of life is changing accordance. Now a day’s we can place the order for anything from our home. There is no need to go the shop of the things we want. The order can be placed online through Internet. The payment, the confirmation of purchasing; we can do everything we want. Now we can think that how the days have been changed with time. People had to stand in rows to wait the terms to buy a particular thing from a popular shop. But what is happening now a day’s; we can extremely surprise that those things can be available on the door-step in few hours

**Definitions**

A shopping cart in a mini project is a virtual container that holds selected products/items chosen by the user for purchase during their online shopping experience.

**Acronyms, and Abbreviations**:

UI: User Interface

UX: User Experience

HTML: Hypertext Markup Language

CSS: Cascading Style Sheets

JS: JavaScript

CRUD: Create, Read, Update, Delete (operations in database management)

HTTP: Hypertext Transfer Protocol

**Overall description:**

**Product perspective**

1. User Interface (UI): The shopping cart interface should be intuitive and user-friendly. It should allow users to easily view, add, remove, and update items in their cart. Clear visual indicators, such as item thumbnails, quantities, and prices, should be provided.
2. Mobile Responsiveness: With the increasing use of mobile devices for online shopping, the shopping cart interface should be responsive and optimized for various screen sizes. This ensures a seamless experience for users across different devices.
3. Product Information: The shopping cart should display relevant information about each item, such as product name, description, price, and quantity. Users should be able to access detailed product information without leaving the cart interface.
4. Promotional Features: The shopping cart can incorporate promotional features, such as displaying related products, offering discounts or coupons based on cart contents, and highlighting special offers or promotions.
5. Cross-Selling and Upselling: The shopping cart can suggest additional products or accessories based on

the items in the cart, encouraging users to add more items to their purchase or upgrade their selections.

**Product functions**

1. Add Product to Cart: Users should be able to easily add products to their shopping cart from product listings or detail pages. This function typically involves clicking an "Add to Cart" button or icon associated with each product.
2. View Cart Contents: Users should have the ability to view the contents of their shopping cart at any time during their shopping session. This function allows users to see the products they have selected, along with details such as product names, quantities, prices, and subtotal amounts.
3. Update Product Quantity: Users should be able to adjust the quantity of each product in their shopping cart. This function allows users to increase or decrease the quantity of a product they want to purchase before proceeding to checkout.
4. Remove Product from Cart: Users should have the option to remove products from their shopping cart if they no longer wish to purchase them. This function enables users to delete individual items from their cart or clear the entire cart.
5. Calculate Cart Total: The shopping cart should automatically calculate the total cost of all items in the cart, including any applicable taxes, shipping fees, or discounts. This function provides users with a clear understanding of the total amount they will be charged before proceeding to checkout.
6. Save Cart for Later: Users may want to save their shopping cart for future reference or continue shopping at a later time. This function allows users to save their cart contents and retrieve them later when they return to the website.
7. Proceed to Checkout: Once users have finished adding items to their shopping cart, they should have the option to proceed to the checkout process. This function initiates the checkout flow, where users provide shipping, billing, and payment information to complete their purchase.

**Assumptions and dependencies**

**Assumptions**:

Users:

Users have basic computer and internet literacy.

Users can browse and select products.

Products:

Products have unique identifiers, descriptions, and prices.

Inventory levels are accurate (for physical goods).

System:

The shopping cart is part of a larger e-commerce system or a point-of-sale system.

The system has a secure connection for handling customer information and transactions.

**Dependencies:**

Software:

The shopping cart functionality relies on a server-side application and potentially a client-side application (web browser).

The system may depend on a database to store product information, user data, and shopping cart contents.

External Services:

For online shopping, the system might depend on payment processing services to handle secure transactions.

Inventory management systems (for physical goods) might need to be integrated for real-time stock updates.

**Specific requirements:**

**Hardware Requirements**

**Server:**

* Processor: Minimum Intel Xeon dual-core or equivalent (consider future scalability)
* RAM: Minimum 16 GB RAM for smooth operation (more if handling a large user base)
* Storage:
* Minimum 250 GB SSD for the operating system and application files.
* Additional storage (HDD/SSD) as needed for patient data based on volume and regulations.
* Network: Gigabit Ethernet connection for efficient data transfer
* Operating System:
* Linux distribution like Ubuntu/CentOS is a popular choice for web development due to its stability and open-source nature.
* Windows Server 2022 can also be used if preferred.

**Workstations:**

* Processor: Minimum Intel Core i3/i5 or equivalent for handling hospital management tasks
* RAM: Minimum 8 GB RAM (16 GB recommended for multitasking)
* Storage: 250 GB SSD/NV Me for faster application loading and data access
* Network: Fast Ethernet connection for stable network access
* Operating System: Windows 10 or equivalent for user familiarity (other options like Linux are also possible)

**Software Requirements**

* Web Server: Apache or Nginx are popular open-source web servers for running the HTML, CSS, and JavaScript files of your application.

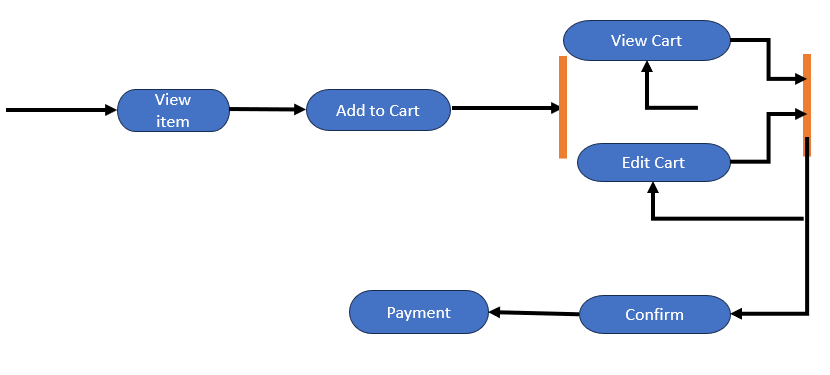
**Front-end Technologies:**

* HTML: for structuring the web pages of your application.
* CSS: for styling the web pages and defining the user interface.
* JavaScript: for adding interactivity and dynamic behavior to the web pages. You might also consider a JavaScript framework like React or Angular for complex user interfaces.

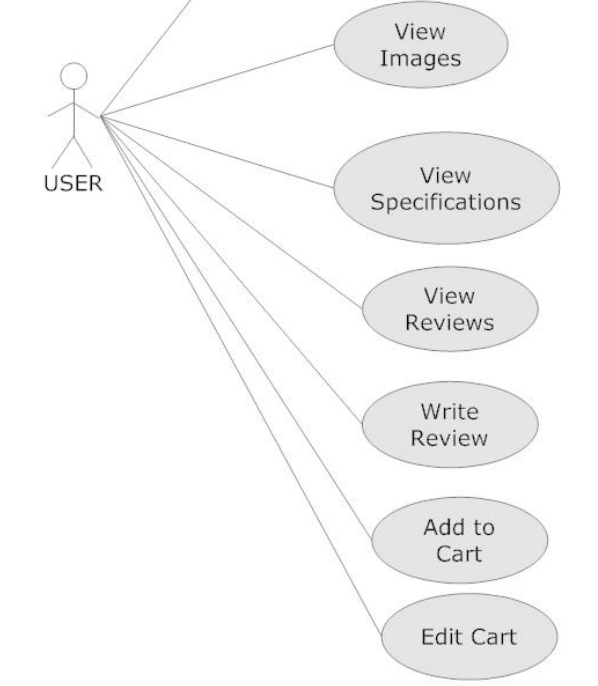
**Back-end Development Language:**

JavaScript (Node.js) can be used for server-side

**Data Flow Diagram**



**Use Case :-**



**Date:**

**Place:**

**Team Leader:**

|  |  |  |
| --- | --- | --- |
| **Roll No.** | **Name of Student** | **Signature** |
| 3059 | Abhishek Sambhaji Patil |  |

**Team Members:**

|  |  |  |
| --- | --- | --- |
| **Roll No.** | **Name of Student** | **Signature** |
| 3057 | Vishal Vitthal Lohar |  |
| 3060 | Purushottam Santosh Patil |  |
| 3067 | Prasad Sanjay Shinde |  |
| 3071 | Suhas Shivanand Uttare |  |
| 3073 | Omkar Sanjay Yadav |  |

**Guide Sign. HOD Sign.**