ONLINE SHOPPING SYSTEM

PROBLEM STATEMENT:

The traditional brick-and-mortar retail industry has been facing challenges due to the increasing popularity of online shopping. With the rapid advancement of technology and the widespread availability of the internet, customers are increasingly turning to online shopping to fulfill their purchasing needs. This has resulted in a significant shift in consumer behavior, with many people now preferring the convenience of shopping from the comfort of their homes, rather than having to physically visit a store.

However, there are still several challenges associated with online shopping that need to be addressed. These include issues such as the lack of personalized attention and the inability to touch and feel the products before making a purchase decision. Additionally, there are concerns around the security of online transactions, as well as the reliability and speed of delivery

To overcome these challenges, there is a need for a robust and user-friendly online shopping system that provides customers with a seamless and secure shopping experience. The system should be designed to enable customers to easily browse through products, make payments securely, track their orders, and receive timely delivery of their purchases. The system should also provide personalized recommendations based on customer preferences and purchase history, as well as offer a range of payment options to cater to the diverse needs of customers

Software Requirement Specification(SRS)

1 Introduction:

1.1 **Purpose of this Document:**The aim of this document is to specify the software requirements for Online shopping system.

1.2 **Scope of this document** –An online shopping system scope includes user registration, product catalog, search, shopping cart, checkout, payment gateway, order tracking, reviews, account management, customer support, and admin panel features. The system must ensure a seamless and secure shopping experience for users while keeping their information private..

1.3 **Overview** – Is a platform for customers to browse, purchase, and track products from a variety of sellers. The website would have a user-friendly interface with features such as a product catalog, search functionality, shopping cart, and secure payment gateway integration. Customers would be able to create accounts and manage their information, and the system would offer order tracking and customer support channels.The benefits of the system are convenience, a wide selection of products, secure transactions, and responsive customer support. It would also emphasize the target audience, which would likely be anyone looking to purchase products online.

2 General description:

An online shopping system is a web-based application that enables users to browse, search, select, and purchase products or services over the internet. The system typically includes a catalog of products or services, a shopping cart, a checkout process, and a payment gateway. Users can create an account, manage their profile, track their order status, and provide feedback on their shopping experience. The system may also include features such as product recommendations, ratings and reviews, promotions and discounts, and customer support. The goal of the online shopping system is to provide a convenient, secure, and enjoyable shopping experience for users, while also increasing the sales and revenue of the online business.

3 Functional Requirements:

Functional Requirements for Online Shopping System:

1. User Registration and Authentication:

Allow users to create new accounts and authenticate existing users.Validate user credentials and ensure secure access to the system.

2. Product Catalog and Search:

Provide a comprehensive catalog of products with categories, descriptions, and images. Implement a search functionality to enable users to find products based on keywords, categories, or filters.

3. Product Details:

Display detailed information about each product, including price, specifications, availability, and customer reviews.Enable users to compare products based on features, prices, and ratings.

4. Shopping Cart Management:

Allow users to add products to their shopping cart for later purchase. Support adding, removing, and updating quantities of items in the shopping cart.Calculate and display the total price of items in the cart, including taxes and shipping charges.

5. Secure Payment Processing:

Provide a secure payment gateway to facilitate online transactions. Support various payment methods, such as credit/debit cards, digital wallets, or cash on delivery. Ensure the confidentiality of users' financial information.

6. Order Placement and Tracking:

Enable users to place orders and provide necessary shipping and billing information.Generate order confirmation and provide users with a unique order ID.Allow users to track the status of their orders, including shipping updates and estimated delivery dates.

7. User Reviews and Ratings:

Allow users to rate and provide reviews for products they have purchased.Display average ratings and user reviews to help other users make informed buying decisions.

8. Wishlist and Saved Items:

Enable users to create wishlists and save products for future reference or purchase.Allow users to share wishlists with others or save them for later.

9. Order History and Account Management:

Maintain a record of users' order history, including order details, payment information, and delivery status.Allow users to view and manage their account information, including personal details and preferences.

10. Return and Refund Management:

Implement a return and refund process for users to initiate returns or request refunds. Define return policies, including timelines, eligibility criteria, and refund methods.

11. **Customer Support and Communication:**

Provide channels for users to contact customer support for inquiries, complaints, or assistance. Enable the system to send automated email notifications for order confirmations, shipping updates, and other relevant information.

4 Interface Requirements:

1. **Graphical User Interface (GUI):** The GUI should be designed to be visually appealing, with a consistent layout and color scheme. It should also be easy to navigate, with clear and concise labels and buttons. The GUI should be compatible with different devices and screen sizes, including desktops, laptops, tablets, and smartphones.

2**. Product Catalog Interface:** The product catalog interface should allow customers to easily browse and search for products based on various criteria, such as product categories, brands, prices, and ratings. The interface should also provide detailed product information, including product descriptions, images, specifications, and customer reviews.

3. **Shopping Cart Interface:** The shopping cart interface should allow customers to add and remove products from their cart, view the total price, and apply any discounts or coupons. The interface should also provide a checkout button that leads customers to the payment and shipping details page.

4. **Payment and Shipping Interface:** The payment and shipping interface should allow customers to choose their preferred payment and shipping methods, enter their payment and shipping details, and view the estimated delivery time and shipping cost. The interface should also provide a confirmation page that summarizes the order details and allows customers to confirm or modify their order.

5. **Account Interface:** The account interface should allow customers to create and manage their accounts, view their order history, track their shipments, and update their personal and payment information. The interface should also provide a password reset option and account security features, such as two-factor authentication.

6. **Customer Support Interface:** The customer support interface should allow customers to contact the customer support team via various channels, such as email, phone, chat, or social media.

5 Performance Requirements:

1. **Response Time:** The system should have a fast response time, with pages loading within a few seconds. The response time should be consistent, regardless of the number of concurrent users or the complexity of the queries.

2. Scalability: The system should be scalable, meaning that it should be able to handle an increasing number of users and transactions without affecting the response time or the system's performance. The system should also be able to handle peak loads during holidays or sales events.

3. Reliability: The system should be reliable, meaning that it should be available 24/7 and have a high uptime percentage. The system should also have backup and recovery mechanisms to ensure data integrity and prevent data loss in case of system failures.

4. Security: The system should be secure, meaning that it should protect user data and transactions from unauthorized access, theft, or fraud. The system should also comply with industry-standard security protocols, such as Secure Sockets Layer (SSL) encryption, Payment Card Industry Data Security Standard (PCI-DSS), and General Data Protection Regulation (GDPR).

5. Capacity: The system should have sufficient capacity to handle large volumes of data, such as product catalogs, user accounts, and order histories. The system should also have a capacity planning mechanism to estimate the required resources and optimize their utilization.

6. Availability: The system should have high availability, meaning that it should be accessible to users from different geographic locations and time zones. The system should also have a failover mechanism to redirect traffic to backup servers or data centers in case of system outages.

6 Design Constraints:

1. Hardware Constraints: The system design should consider the hardware constraints, such as the server capacity, storage space, and processing power. The system should also be compatible with the hardware infrastructure of the hosting provider or the client's environment.

2. Software Constraints: The system design should consider the software constraints, such as the operating system, database management system, web server, and programming languages. The system should also be compatible with the software architecture and protocols of the client's ecosystem.

3. Network Constraints: The system design should consider the network constraints, such as the bandwidth, latency, and reliability of the internet connection. The system should also be able to handle different types of network protocols and security mechanisms, such as firewalls, virtual private networks (VPNs), and intrusion detection systems (IDS).

4. User Constraints: The system design should consider the user constraints, such as the user's technical skills, language preferences, and accessibility needs. The system

should also comply with the user experience (UX) standards and guidelines, such as the Web Content Accessibility Guidelines (WCAG) and the Human Interface Guidelines (HIG).

5. **Legal Constraints:** The system design should consider the legal constraints, such as the data protection laws, copyright laws, and consumer protection laws. The system should also comply with the industry standards and regulations, such as the Payment Card Industry Data Security Standard (PCI-DSS) and the General Data Protection Regulation (GDPR)

7 Non-Functional Attributes:

Usability: The system should be user-friendly, easy to learn, and intuitive, with a well-designed user interface that enables users to perform tasks quickly and efficiently.

Reliability: The system should be reliable, with a low error rate and minimal downtime, ensuring that users can access and use the system at all times.

Security: The system should be secure, protecting user data and transactions from unauthorized access or breaches, and complying with regulatory requirements.

Performance: The system should be fast and responsive, with minimal latency and delay, enabling users to perform tasks quickly and efficiently.

Scalability: The system should be scalable, able to handle a large number of users and transactions, without compromising performance or reliability.

Maintainability: The system should be easy to maintain, with well-documented code, clear error messages, and easy-to-use debugging tools, enabling developers to identify and fix issues quickly.

8 Preliminary Schedule and Budget:

Requirements gathering……. 2 weeks

Systemdesign……5weeks
Implementation…….. 4 weeks

Unit testing…… 2 weeks

Final testing…….. 3 weeks

Budget: The budget for whole project is : Rs.30,000