**SOFTWARE REQUIREMENT SPECIFICATION FOR**

**E-COMMERCE APPLICATION**

**Product Description**

1. E-commerce platform is built with a focus on green software development, minimizing energy consumption and promoting environmental sustainability.
2. Customers enjoy an interface for easy browsing and purchasing, while sellers benefit from robust inventory management tools.
3. Developed in Python, platform prioritizes eco-friendly practices, contributing positively to sustainable IT initiatives.

**Requirement Analysis:**

1. **User Authentication and Authorization**

**Description**: Users should be able to register, login, and logout securely.

* **Functional Requirements:**
* Registration: Users can create an account with email/password or social media authentication.
* Login: Users should be able to log in securely using their credentials.
* Logout: Users can securely log out of their accounts.
* **Non-functional Requirements:**
* Security: Implement secure authentication mechanisms like bcrypt for password hashing.
* Privacy: Ensure user data privacy and compliance with relevant regulations

1. **Product Browsing and Searching**

**Description**: Users should be able to browse and search for products easily.

* **Functional Requirements:**
  + Product Categories: Products should be organized into categories for easy navigation.
  + Search: Implement a search functionality allowing users to find products by name, category, or attributes.
  + Filtering: Allow users to filter search results based on various criteria (e.g., price, brand).
* **Non-functional Requirements:**
  + Performance: Ensure fast and responsive browsing and searching, minimizing server-side processing time.
  + Accessibility: Design the browsing and search interfaces to be accessible to users with disabilities.

1. **Shopping Cart Management**

**Description**: Users should be able to add/remove products to/from their shopping carts.

* **Functional Requirements:**
  + Add to Cart: Users should be able to add products to their carts from product pages.
  + Remove from Cart: Users should be able to remove products from their carts.
  + Quantity Adjustment: Allow users to change the quantity of products in their carts.
* **Non-functional Requirements:**
  + Persistence: Ensure that shopping cart contents persist across user sessions.
  + Scalability: Design the shopping cart system to handle a large number of concurrent users.

1. **Checkout Process**

**Description**: Streamlined process for users to complete purchases securely.

* **Functional Requirements:** 
  + Payment option: Multiple payment options (credit card, PayPal, etc.).
  + Address : Shipping address and delivery options.
  + Order Summary : Order summary before finalizing purchase.
* **Non-Functional Requirements:**
  + - Security: Secure transmission of payment information using encryption.
    - Esurance : Confirmation email with order details sent to the user.

1. **Order Management**

**Description**: Administrators should be able to manage orders efficiently.

* **Functional Requirements:**
  + Track Order : View and filter orders by status (pending, shipped, delivered).
  + Status : Update order status (e.g., mark as shipped).
  + Billing : Generate invoices and packing slips.
* **Non-Functional Requirements**:
  + Notifications : Notification system for new orders and status updates.
  + Data Management: Data archival and backup mechanisms to prevent data loss.

1. **Product Listing and Management**

**Description**: Sellers should be able to list and manage their products effectively.

* + **Functional Requirements:**
    - Product Upload: Allow sellers to upload product listings with details such as title, description, images, price, and inventory quantity.
    - Bulk Upload: Support bulk uploading of products through CSV or Excel files for efficient inventory management.
    - Product Variation Management: Enable sellers to create product variations (e.g., size, colour) and manage inventory for each variant.
  + Non-functional Requirement:
* Scalability: scalable to accommodate a growing number of sellers and products over time.
* Auditability: system should maintain a detailed audit trail of product listing and management

1. **Payment Processing**

**Description**: Secure processing of payments for orders.

* + Functional Requirements:
  + Payment gateways: Integration with payment gateways (Stripe, PayPal, etc.).
  + Error Handling :Handling of payment failures and retries.
  + Refund : Refund processing for cancelled orders.
  + Non-Functional Requirements:
    - Security: Compliance with PCI DSS standards for payment security.
    - Reliability: Redundant payment gateway integration for reliability.

1. **Sustainability into Software design and development**

**Description:** Incorporate sustainability principles into software design and development.

* + **Functional Requirements:**
    - Energy-efficient algorithms and data structures.
    - Minimization of server resource usage during idle periods.
    - Carbon footprint tracking and reporting.
  + **Non-Functional Requirements**:
    - Adoption of green hosting solutions (renewable energy sources, energy-efficient data center).
    - Regular environmental impact assessments and improvement plans.

**Functional Specification:**

1. **User Authentication and Authorization**

* **Registration**:
  + Users can create an account using email/password or social media authentication.
  + Mandatory fields: email, password, username (optional), and agreement to terms of service.
* **Login**:
  + Users can securely log in using their registered email and password.
* **Logout**:
  + Users can securely log out of their accounts.

**2. Product Browsing and Searching**

* **Product Categories**:
  + Products are categorized into various categories (e.g., electronics, clothing, home decor).
  + Each category displays a list of relevant products.
* **Search**:
  + Users can search for products by name, category, brand, or attributes using a search bar.
  + Search results are displayed in real-time with relevant product suggestions.
* **Filtering**:
  + Users can filter search results based on criteria such as price range, brand, size, colour, and customer ratings.

**3. Shopping Cart Management**

* **Add to Cart**:
  + Users can add products to their shopping cart from product detail pages or search results.
* **View Cart**:
  + Users can view the contents of their shopping cart, including product details, quantity, and total price.
* **Update Cart**:
  + Users can update the quantity of products in their cart or remove products entirely.
* **Save for Later**:
  + Users can move items from their cart to a "Save for Later" list for future purchase consideration.

**4. Checkout Process**

* **Cart Review**:
  + Users review the items in their cart, adjust quantities, and remove items if necessary.
* **Shipping Address**:
  + Users provide a shipping address for delivery, with the option to save multiple addresses for future use.
* **Payment Options**:
  + Users select a preferred payment method (credit/debit card, PayPal, etc.) and enter payment details securely.
* **Order Confirmation**:
  + Users receive a confirmation page with order details, including an order number and estimated delivery date.
* **Order History**:
  + Completed orders are stored in the user's order history for future reference.

**5. User Profile Management**

* **Edit Profile**:
  + Users can edit their profile information, including name, email, password, and contact details.
* **Address Management**:
  + Users can add, edit, or delete shipping addresses for convenient checkout.
* **Payment Method Management**:
  + Users can add, edit, or delete payment methods, including credit/debit cards and PayPal accounts.

**6. Ratings and Reviews**

* **Product Ratings**:
  + Users can rate products on a scale (e.g., 1 to 5 stars) and leave written reviews.
  + Average rating and total number of reviews are displayed on product detail pages.
* **Review Moderation**:
  + Reviews are moderated to ensure they meet community guidelines and are not spam or abusive.
* **User Feedback**:
  + Users can provide feedback on their shopping experience, product quality, and delivery satisfaction.

**7. Product Listing and Management**

* **Product Upload:**
  + Sellers can upload product listings with details such as title, description, images, price, and inventory quantity.
* **Product Variation Management:**
  + Enable sellers to create product variations (e.g., size, color) and manage inventory for each variant.

**8. Order Management (for Admin)**

* **Order Processing**:
  + Admins can view and process incoming orders, update order status (e.g., pending, processing, shipped).
* **Inventory Management**:
  + Admins can manage product inventory, update stock levels, and add new products to the catalog.
* **Customer Support**:
  + Admins can communicate with customers regarding order inquiries, returns, and refunds.

**External Interface Specification**

* **User Interface**

***UI for User***

1. **Homepage**:
   * Shows featured products, promotions, and navigation.
   * Includes a search bar and category navigation.
2. **Product Pages**:
   * Displays detailed product info, variants, and an "Add to Cart" button.
   * Shows related products for cross-selling.
3. **Shopping Cart**:
   * Summarizes items in the cart with quantity adjustment.
   * Offers a "Proceed to Checkout" button.
4. **Checkout Process**:
   * Guides users through shipping, payment, and order review.
   * Allows guest checkout or account creation.
5. **User Account Dashboard**:
   * Provides order history and status.
   * Offers address management and account settings.

***UI for Seller***

**1. Seller Dashboard:**

* Quick access to tools for managing products, orders, and customers.
* Graphs showing sales trends, top products, and order status.

1. **Product Management:**

* Interface to add, update, and manage products and inventory.
* Tools for categorizing, pricing, and defining product variations.

***UI for Admin***

**1. Dashboard:**

* Centralized hub for sales, revenue, and website traffic metrics.
* Quick access to tools for managing the e-commerce platform.

1. **Order Management:**

* Dashboard for incoming orders with status indicators.
* Tools for order processing, invoicing, and shipping label printing.
* Ability to track order status, manage returns, and handle inquiries.

1. **User Management:**

* Interface for managing user accounts (customers, sellers, admins).
* Tools for adding users, updating profiles, and resetting passwords.
* Ability to monitor user activity and enforce security measures.
* **Communication Interfaces**

1. HTTPS Encryption:
   * Secure communication between clients and servers using HTTPS protocol.
   * SSL/TLS certificates for data encryption and authentication.
2. RESTful APIs:
   * Expose RESTful APIs for client-server communication.
   * Define endpoints for user authentication, product management, and order processing.

* **Hardware Interfaces**
  + 1. Device Compatibility:
* Support for desktop computers, laptops, tablets, and smartphones.
* Responsive design for different screen sizes and resolutions.
  + 1. Peripheral Devices:
* Compatibility with standard input devices (keyboard, mouse) and output devices (display).
  + 1. Internet Connectivity
* Need internet connectivity medium as for modem , LAN-WAN, Ethernet Cross Cable
* **Database**

1.Relational Database management system such as MySQL

2.Data access layer

**Technical Specification**

**Architecture**

* The application shall follow a three-tier architecture, comprising presentation, application logic, and data storage layers.

**Database Design**

* The system shall use a relational database for storing user data, product information, and order details.

**Programming Language**

* The application shall be primarily developed using Python.

**Security Measures**

* Data transmission shall be encrypted using HTTPS to ensure secure communication between clients and the server.
* Passwords shall be securely hashed using bcrypt before storage to protect user credentials.

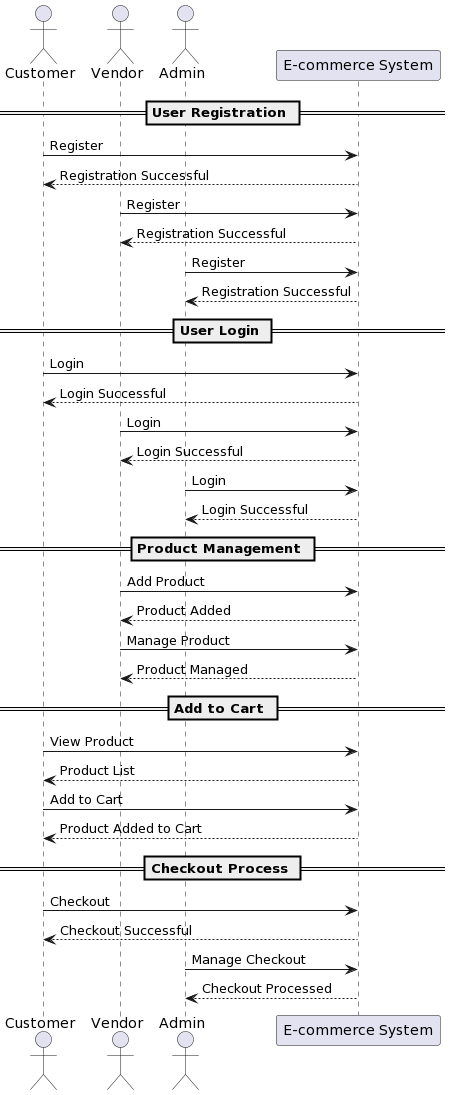
**Memory Specifications**

* The application server shall have a minimum of 4 GB of RAM to handle concurrent user requests, database operations, and caching.
* The database server shall have a minimum of 2 GB of RAM for efficient query processing and caching of frequently accessed data.

**OS Specification**

* The application shall be compatible with Linux-based operating systems such as Ubuntu Server, CentOS, Debian, and Fedora for both development and deployment.

**UML Diagrams**

* **Sequence Diagram**