

## Task 2

Suppose you are senior engineer on this project, and you are working with three (3) other developers. How would you organize the software development of this project, so that you can all work together collaboratively.

- Note: Key areas to focus is work load assignment using a tool like Jira, Azure Boards, or Trello
- Consider working within an Agile Software Development team and consider code conflicts, pull requests, and other concepts.

Deliverable

- Make a list of Epics (maximum 10 epics), features within those epics, and a few user stories (max 20).

**Answer:**

### Agile Implementation

1. **Sprint Planning:** Assign user stories to team members based on their skills and availability using Jira.
2. **Daily Stand-ups:** Conduct daily stand-up meetings to discuss progress, any blockers, and plans for the day.
3. **Code Collaboration:** Utilize Git for version control. Encourage feature branching and pull requests for code reviews to manage code conflicts effectively.
4. **Sprint Review and Retrospective:** Hold sprint review meetings to demo completed user stories and gather feedback. Conduct retrospectives to discuss improvements and action items.
5. **Continuous Integration and Deployment:** Implement CI/CD pipelines to automate testing, integration, and deployment processes for efficient delivery of features.

## Epics, Features, and User Stories

### Epic Story 1: User Authentication and Management

#### User Story 1: User Registration

**Title:** Allow new users to register for an account

**Acceptance Criteria:**

1. User should be able to access a registration form.
2. User must provide a unique username, a valid email address, and a strong password.
3. Upon successful registration, the user should receive a confirmation email.
4. The user's password should be securely hashed and stored in the database.
5. Registration form should include client-side validation for input fields.

#### User Story 2: User Login

**Title:** Enable existing users to log into their accounts

**Acceptance Criteria:**

1. User should see a login form with fields for username/email and password.
2. User must enter valid credentials to access the system.
3. System should authenticate the user against stored credentials in the database.
4. Upon successful login, user should be redirected to their dashboard or main page.
5. Failed login attempts should provide appropriate error messages.
6. Remember me functionality should be available for convenience.

**User Story 3: Password Recovery**

**Title:** Allow users to recover/reset their passwords

**Acceptance Criteria:**

1. User should have a "Forgot Password?" option on the login screen.
2. User must provide their registered email address for password recovery.
3. System should generate a unique password reset link and send it via email.
4. Password reset link should expire after a specified time period (e.g., 1 hour).
5. User should be able to set a new password after clicking the reset link.

**User Story 4: User Profile Management**

**Title:** Enable users to manage their profile information

**Acceptance Criteria:**

1. User should have access to a profile page where they can view and update their information.
2. User should be able to change their password from the profile page.
3. Profile updates should be validated and saved to the database.
4. Profile page should display user's current information and allow changes to fields like name, email, etc.

**User Story 5: Account Deactivation**

**Title:** Allow users to deactivate their accounts

**Acceptance Criteria:**

1. User should have an option to deactivate their account from the profile settings.
2. Deactivation process should prompt the user for confirmation.
3. Upon deactivation, user's account should be disabled and no longer accessible.
4. Deactivated accounts should retain their data for potential reactivation.

**User Story 6: Admin Dashboard**

**Title:** Provide administrators with a dashboard for user management

**Acceptance Criteria:**

1. Admin should have a separate login and dashboard interface.
2. Dashboard should display a list of registered users with basic details.

3. Admin should be able to search, view, edit, and delete user accounts.
4. Actions performed by admin (edit, delete) should be logged for auditing purposes.

## **User Story 7: User Roles and Permissions**

**Title:** Implement roles and permissions for different user types

**Acceptance Criteria:**

1. System should support multiple user roles (e.g., admin, regular user).
2. Each role should have specific permissions (e.g., create, read, update, delete).
3. Admin should be able to assign roles and permissions to users.
4. Users should only have access to functionalities allowed by their role.

## **User Story 8: Account Lockout Mechanism**

**Title:** Implement security measures against brute force attacks

**Acceptance Criteria:**

1. System should track failed login attempts for each user.
2. After a predefined number of failed attempts (e.g., 5), user's account should be temporarily locked.
3. Locked account should automatically unlock after a specified time period (e.g., 15 minutes).
4. User should receive notification emails for account lockout.

## **User Story 9: Two-Factor Authentication (2FA)**

**Title:** Provide enhanced security with two-factor authentication

**Acceptance Criteria:**

1. User should have an option to enable 2FA from their account settings.
2. System should support at least one 2FA method (e.g., SMS, authenticator app).
3. User must verify their identity with 2FA during login after enabling it.
4. Admin should be able to enforce 2FA for specific user roles or all users.

## **User Story 10: GDPR Compliance**

**Title:** Ensure compliance with GDPR regulations for user data

**Acceptance Criteria:**

1. System should include a privacy policy and terms of service accessible to users.
2. Users must explicitly consent to data collection and processing during registration.
3. Admin should be able to export user data upon request for GDPR compliance.
4. Implement mechanisms for users to request data deletion or account closure.

## **User Story 11: Account Activity Tracking**

**Title:** Track and display user activity within the system

**Acceptance Criteria:**

1. System should log important user actions (e.g., login, logout, profile updates).
2. User should be able to view their recent activity log from the profile page.
3. Admin should have access to a comprehensive audit log of all user activities.
4. Logs should include timestamps, actions performed, and IP addresses for security auditing.

**User Story 12: Multi-Device Support**

**Title:** Enable users to access their accounts from multiple devices

**Acceptance Criteria:**

1. System should securely manage sessions and tokens for user authentication.
2. User should be able to log in and stay logged in across different devices.
3. Implement mechanisms to invalidate sessions or tokens upon user request or security events.
4. Provide options for users to remotely log out from devices they no longer use.

**Epic 2: Product Management****User Story 1: Create Product**

**Title:** Allow administrators to add new products to the system

**Acceptance Criteria:**

1. Admin should have access to a form for adding new products.
2. Form should include fields for product name, description, category, price, and inventory count.
3. System should validate input fields to ensure data integrity.
4. Upon submission, the new product should be saved to the database and made available for sale.

**User Story 2: Update Product Information**

**Title:** Enable administrators to modify existing product details

**Acceptance Criteria:**

1. Admin should be able to access a product management interface listing all products.
2. Admin should have the ability to edit fields such as name, description, price, and inventory count.
3. Changes made to product information should be validated and saved to the database.
4. Updated product details should reflect immediately on the product catalog and listings.

**User Story 3: Delete Product**

**Title:** Allow administrators to remove products from the system

**Acceptance Criteria:**

1. Admin should have a mechanism to select and delete products from the product management interface.
2. System should prompt for confirmation before deleting a product to prevent accidental removal.
3. Deleted products should no longer appear in the product catalog or listings.
4. Ensure that any associated data (e.g., orders, reviews) are appropriately handled or archived.

## **User Story 4: View Product Details**

**Title:** Enable users to view detailed information about a product

**Acceptance Criteria:**

1. User should be able to click on a product listing to view its details.
2. Product details page should display information such as name, description, price, and availability.
3. Include additional features like product images, reviews, and related products for a comprehensive view.
4. Ensure that product details page is accessible and responsive across different devices.

## **User Story 5: Search and Filter Products**

**Title:** Implement search and filter functionality for product browsing

**Acceptance Criteria:**

1. Users should have a search bar prominently displayed to enter keywords.
2. System should provide filtering options (e.g., by category, price range) to refine product listings.
3. Search results and filters should update dynamically without requiring page refresh.
4. Ensure that search and filter functionalities are efficient and responsive for a smooth user experience.

## **User Story 6: Product Categories**

**Title:** Categorize products for better organization and navigation

**Acceptance Criteria:**

1. Admin should be able to create, edit, or delete product categories.
2. Each product should belong to at least one category.
3. Users should be able to browse products by category through a category menu or navigation bar.
4. Ensure that category management and navigation are intuitive and user-friendly.

## **User Story 7: Inventory Management**

**Title:** Track and manage product inventory levels

**Acceptance Criteria:**

1. System should update product inventory count automatically upon purchase or return.
2. Admin should have access to a dashboard or report showing current inventory levels.
3. Implement alerts or notifications for low-stock products to prevent out-of-stock situations.
4. Ensure that inventory management functions smoothly to support sales and logistics.

## **User Story 8: Product Reviews and Ratings**

**Title:** Allow users to review and rate products

**Acceptance Criteria:**

1. Users should have the option to leave a review and rating for products they have purchased.
2. Reviews should include fields for text content, star rating, and optional photos.
3. Display average rating and total number of reviews prominently on the product details page.
4. Admin should have the ability to moderate reviews and ratings if necessary.

## **User Story 9: Product Recommendations**

**Title:** Implement personalized product recommendations

**Acceptance Criteria:**

1. System should analyze user behavior (e.g., purchase history, browsing patterns) to suggest relevant products.
2. Recommend products should be displayed on the homepage, product details page, or during checkout.
3. Provide options for users to view more recommendations or adjust preferences.
4. Ensure that recommendation algorithms are effective and enhance the shopping experience.

## **User Story 10: Product Insights and Analytics**

**Title:** Provide analytics and insights for product performance

**Acceptance Criteria:**

1. Admin should have access to reports showing sales trends, popular products, and inventory turnover.
2. Analytics should include metrics such as revenue generated by product, conversion rates, and customer demographics.
3. Implement visualizations like graphs and charts to present data in a clear and actionable format.
4. Ensure that product analytics contribute to strategic decision-making and business growth.

## **Epic 3: Order Management**

### **User Story 1: Place Order**

**Title:** Allow users to place orders for products

**Acceptance Criteria:**

1. User should be able to add products to their shopping cart from the product listings.
2. Cart should display product details, quantity selection, and total price.
3. User must provide shipping address, contact information, and preferred payment method.
4. Upon submission, system should generate an order confirmation with a unique order ID.
5. Order details should be saved in the database for reference and processing.

## **User Story 2: Order Confirmation**

**Title:** Provide users with confirmation of their placed orders

**Acceptance Criteria:**

1. After successful order submission, user should receive an email confirmation with order details.
2. Order confirmation page should display summary of purchased items, shipping information, and total cost.
3. Include a "Print Order" option on the confirmation page for users to generate a printable receipt.
4. Ensure that order confirmation is displayed immediately after checkout and accessible from user account.

## **User Story 3: Order Tracking**

**Title:** Enable users to track the status of their orders

**Acceptance Criteria:**

1. User should have access to an order tracking page from their account dashboard.
2. Tracking page should display current status (e.g., processing, shipped, delivered) and estimated delivery date.
3. System should provide real-time updates on order status based on logistics information.
4. Ensure that tracking information is clear, accurate, and updated regularly.

## **User Story 4: Cancel Order**

**Title:** Allow users to cancel orders before shipment

**Acceptance Criteria:**

1. User should have an option to cancel an order from their order details page.
2. Cancellation should be possible only if the order status is "pending" or "processing".
3. System should prompt user for confirmation before canceling the order to prevent accidental actions.
4. Cancellation should update the order status and initiate a refund process if applicable.
5. Ensure that canceled orders are reflected immediately in the user's order history and account balance.

## **User Story 5: Refund Processing**

**Title:** Process refunds for canceled or returned orders

**Acceptance Criteria:**

1. Admin should have access to a refund management interface to process refunds.
2. Refund should be initiated automatically upon order cancellation or return approval.
3. System should calculate refund amount based on returned items and any applicable fees.
4. User should receive email notification when refund is processed, with details of refunded amount and timeline.
5. Ensure that refunded amount is credited back to the user's original payment method promptly.

## **User Story 6: Order History**

**Title:** Provide users with a comprehensive order history

**Acceptance Criteria:**

1. User should have access to an order history page displaying all past orders.
2. Each order entry should include details such as order ID, date, status, and total amount.
3. User should be able to view order details by clicking on any entry in the order history.
4. Include options to filter orders by status (e.g., pending, shipped, delivered) or date range.
5. Ensure that order history is paginated for ease of navigation, especially for users with multiple orders.

## **User Story 7: Manage Shipping Options**

**Title:** Allow users to select and manage shipping options

**Acceptance Criteria:**

1. During checkout, user should have options to choose from available shipping methods (e.g., standard, express).
2. System should calculate shipping costs based on selected method, destination, and order weight.
3. Provide estimated delivery dates for each shipping option to inform user's decision.
4. Allow users to update shipping address and preferences from their account settings.
5. Ensure that selected shipping options are accurately reflected in the order summary and confirmation.

## **User Story 8: Bulk Order Processing**

**Title:** Implement functionality for bulk order processing

**Acceptance Criteria:**

1. Admin should have a bulk order management interface to view and process multiple orders simultaneously.
2. Interface should allow filtering and sorting of orders based on status, date, or other criteria.
3. Admin should be able to update order statuses in bulk (e.g., mark orders as shipped).
4. System should handle large volumes of orders efficiently without performance degradation.
5. Ensure that bulk order processing maintains data integrity and sends appropriate notifications to users.

## **User Story 9: Order Notifications**

**Title:** Notify users about important order updates

**Acceptance Criteria:**

1. System should send email notifications to users for order confirmation, status updates, and shipment tracking.
2. Notifications should include relevant details such as order ID, status change, and tracking information.
3. User should have the option to opt in or out of specific notification types (e.g., shipment updates).
4. Ensure that notifications are timely, clear, and enhance user experience without overwhelming users with excessive emails.



## **User Story 10: Order Metrics and Analytics**

**Title:** Provide analytics and insights on order performance

**Acceptance Criteria:**

1. Admin should have access to reports showing key metrics such as total orders, revenue, and average order value.
2. Analytics should include trends analysis (e.g., seasonal variations, popular products) for strategic planning.
3. Implement visualizations like graphs and charts to present data in a meaningful and actionable format.
4. Ensure that order analytics contribute to decision-making processes and business growth strategies.

## **User Story 11: Partial Order Fulfillment**

**Title:** Allow partial fulfillment of orders with multiple items

**Acceptance Criteria:**

1. System should support splitting an order into multiple shipments if some items are out of stock.
2. User should receive notification about items that are shipped separately.
3. Admin should be able to track and manage each partial shipment independently.
4. Ensure that partial order fulfillment does not affect order integrity or cause confusion for users.

## **User Story 12: Order Comments and Communication**

**Title:** Facilitate communication between users and customer support regarding orders

**Acceptance Criteria:**

1. Users should have the option to leave comments or notes regarding specific orders.
2. Admin and support staff should be able to view and respond to order comments from a dedicated interface.
3. Ensure that comments are threaded and organized to maintain context and history.
4. Implement notifications for both users and support staff when new comments are added.
5. Comments should be visible to relevant parties only (e.g., user, support staff handling the order).

## **User Story 13: Order Discounts and Coupons**

**Title:** Implement discounts and coupons for orders

**Acceptance Criteria:**

1. System should allow users to apply discount codes or coupons during checkout.
2. Discount should be applied automatically upon entering a valid code, reducing the order total.
3. Admin should have the ability to create, edit, and deactivate discount codes from a management interface.
4. Ensure that discount codes are validated and applied correctly without impacting other order details.
5. Provide users with feedback on applied discounts (e.g., discount amount, savings).

## **User Story 14: Order Returns and Exchanges**

**Title:** Handle returns and exchanges for purchased items

**Acceptance Criteria:**

1. User should have access to a returns and exchanges process from their order history.
2. System should guide user through steps to initiate a return or exchange, including reason selection and item condition.
3. Admin should receive and manage return requests from a dedicated interface, approving or rejecting based on policy.
4. Ensure that returned items are properly accounted for in inventory and refunds are processed promptly upon approval.
5. Communicate return status and updates to both user and admin throughout the process.

## **User Story 15: Order Audit and Compliance**

**Title:** Maintain order audit trail and ensure compliance with regulations

**Acceptance Criteria:**

1. System should log all significant actions related to order processing (e.g., status changes, refunds).
2. Admin should have access to an audit log showing detailed history of each order transaction.
3. Implement security measures to protect audit logs from unauthorized access or tampering.
4. Ensure that order processing adheres to regulatory requirements (e.g., GDPR, PCI DSS) for data protection and privacy.
5. Periodically review and audit order logs to identify anomalies or discrepancies for corrective action.

## **Epic 4: Payment Integration**

### **User Story 1: Select Payment Method**

**Title:** Allow users to select their preferred payment method during checkout

**Acceptance Criteria:**

1. User should see a list of available payment methods (e.g., credit/debit card, PayPal, Apple Pay).
2. Each payment method should be clearly labeled with an icon and description.
3. System should provide an option to add or manage payment methods from user account settings.
4. Ensure that selected payment method is displayed on the order summary before finalizing the purchase.

## **User Story 2: Process Credit/Debit Card Payments**

**Title:** Implement credit/debit card payment processing for purchases

**Acceptance Criteria:**

1. System should securely collect and validate credit/debit card details (card number, expiry date, CVV).
2. Payment processing should comply with PCI DSS standards for card data security.
3. User should receive real-time feedback on payment success or failure.
4. Ensure that payment processing handles common card types and supports international transactions if applicable.

## **User Story 3: PayPal Integration**

**Title:** Integrate PayPal as a payment option for users

**Acceptance Criteria:**

1. User should have the option to select PayPal as a payment method during checkout.
2. System should redirect user to PayPal's secure payment gateway for authentication and payment confirmation.
3. Upon successful payment, user should be redirected back to the website with payment confirmation.
4. Implement mechanisms to handle PayPal transaction disputes and refunds through the website's admin interface.

## **User Story 4: Apple Pay Integration**

**Title:** Enable Apple Pay as a payment option for users with Apple devices

**Acceptance Criteria:**

1. User accessing the website from an Apple device should see Apple Pay as a payment option.
2. Implement Apple Pay's secure payment sheet for authentication and payment authorization.
3. Ensure that Apple Pay integration adheres to Apple's security and user privacy guidelines.
4. Provide users with a seamless checkout experience using their Apple ID for payment confirmation.

## **User Story 5: Payment Gateway Integration**

**Title:** Integrate a third-party payment gateway for secure transactions

**Acceptance Criteria:**

1. Admin should have access to configure and manage payment gateway settings from the admin dashboard.
2. Implement APIs for communication between the website and selected payment gateway (e.g., Stripe, Braintree).
3. System should handle various payment scenarios (e.g., authorization, capture, refunds) through the gateway.
4. Ensure that payment gateway integration supports multiple currencies and payment methods to cater to diverse user needs.

## **User Story 6: Subscription Payments**

**Title:** Support recurring subscription payments for services or products

**Acceptance Criteria:**

1. User should have the option to subscribe to a service or product with recurring payments (e.g., monthly, annually).
2. System should securely store payment details and initiate automatic recurring charges at specified intervals.
3. Provide users with options to manage subscriptions, including upgrade/downgrade plans and cancel subscriptions.
4. Admin should receive notifications and reports on subscription status and payment history.

## **User Story 7: Payment Confirmation Emails**

**Title:** Send confirmation emails to users after successful payment

**Acceptance Criteria:**

1. User should receive an email confirmation immediately after completing a payment transaction.
2. Confirmation email should include details such as order summary, payment amount, and transaction ID.
3. Ensure that confirmation emails are formatted correctly and include branding elements for a professional look.
4. Provide users with a link to view their order status or print an invoice directly from the email.

## **User Story 8: Handle Failed Payments**

**Title:** Manage and notify users about failed payment transactions

**Acceptance Criteria:**

1. System should detect and flag failed payment attempts (e.g., insufficient funds, card expiration).
2. User should receive an email notification with details about the failed payment and instructions to update payment information.
3. Provide users with a convenient way to update payment details or retry the payment from their account settings.
4. Admin should have access to a dashboard or report listing failed transactions and reasons for failure.

## **User Story 9: Refund Processing**

**Title:** Handle refunds for canceled orders or returned items

**Acceptance Criteria:**

1. Admin should have access to initiate refunds through the website's admin interface.
2. Refund amount should be calculated automatically based on returned items and any applicable fees.
3. System should communicate refund status to users via email, including details of refunded amount and timeline.
4. Ensure that refunded amount is credited back to the user's original payment method promptly and accurately.

## **User Story 10: Payment Security and Compliance**

**Title:** Ensure compliance with payment security standards (e.g., PCI DSS)

**Acceptance Criteria:**

1. Implement secure encryption protocols for handling payment data (e.g., SSL/TLS).
2. System should not store sensitive card information beyond what is required for transaction processing.
3. Regularly conduct security audits and vulnerability assessments to maintain compliance with PCI DSS.
4. Educate and train staff on best practices for handling payment data securely.

## **User Story 11: Payment Analytics and Reporting**

**Title:** Provide analytics and insights on payment transactions

**Acceptance Criteria:**

1. Admin should have access to reports showing key metrics such as total transactions, revenue, and payment method popularity.
2. Analytics should include trends analysis (e.g., peak transaction times, average transaction value) for strategic planning.
3. Implement visualizations like graphs and charts to present payment data in a meaningful and actionable format.
4. Ensure that payment analytics contribute to decision-making processes and business growth strategies.

## **Epic 5: Admin Dashboard**

### **User Story 1: Select Payment Method**

**Title:** Allow users to select their preferred payment method during checkout

**Acceptance Criteria:**

1. User should see a list of available payment methods (e.g., credit/debit card, PayPal, Apple Pay).
2. Each payment method should be clearly labeled with an icon and description.
3. System should provide an option to add or manage payment methods from user account settings.
4. Ensure that selected payment method is displayed on the order summary before finalizing the purchase.

### **User Story 2: Process Credit/Debit Card Payments**

**Title:** Implement credit/debit card payment processing for purchases

**Acceptance Criteria:**

1. System should securely collect and validate credit/debit card details (card number, expiry date, CVV).
2. Payment processing should comply with PCI DSS standards for card data security.

3. User should receive real-time feedback on payment success or failure.
4. Ensure that payment processing handles common card types and supports international transactions if applicable.

### **User Story 3: PayPal Integration**

**Title:** Integrate PayPal as a payment option for users

**Acceptance Criteria:**

1. User should have the option to select PayPal as a payment method during checkout.
2. System should redirect user to PayPal's secure payment gateway for authentication and payment confirmation.
3. Upon successful payment, user should be redirected back to the website with payment confirmation.
4. Implement mechanisms to handle PayPal transaction disputes and refunds through the website's admin interface.

### **User Story 4: Apple Pay Integration**

**Title:** Enable Apple Pay as a payment option for users with Apple devices

**Acceptance Criteria:**

1. User accessing the website from an Apple device should see Apple Pay as a payment option.
2. Implement Apple Pay's secure payment sheet for authentication and payment authorization.
3. Ensure that Apple Pay integration adheres to Apple's security and user privacy guidelines.
4. Provide users with a seamless checkout experience using their Apple ID for payment confirmation.

### **User Story 5: Payment Gateway Integration**

**Title:** Integrate a third-party payment gateway for secure transactions

**Acceptance Criteria:**

1. Admin should have access to configure and manage payment gateway settings from the admin dashboard.
2. Implement APIs for communication between the website and selected payment gateway (e.g., Stripe, Braintree).
3. System should handle various payment scenarios (e.g., authorization, capture, refunds) through the gateway.
4. Ensure that payment gateway integration supports multiple currencies and payment methods to cater to diverse user needs.

### **User Story 6: Subscription Payments**

**Title:** Support recurring subscription payments for services or products

**Acceptance Criteria:**

1. User should have the option to subscribe to a service or product with recurring payments (e.g., monthly, annually).

2. System should securely store payment details and initiate automatic recurring charges at specified intervals.
3. Provide users with options to manage subscriptions, including upgrade/downgrade plans and cancel subscriptions.
4. Admin should receive notifications and reports on subscription status and payment history.

## **User Story 7: Payment Confirmation Emails**

**Title:** Send confirmation emails to users after successful payment

**Acceptance Criteria:**

1. User should receive an email confirmation immediately after completing a payment transaction.
2. Confirmation email should include details such as order summary, payment amount, and transaction ID.
3. Ensure that confirmation emails are formatted correctly and include branding elements for a professional look.
4. Provide users with a link to view their order status or print an invoice directly from the email.

## **User Story 8: Handle Failed Payments**

**Title:** Manage and notify users about failed payment transactions

**Acceptance Criteria:**

1. System should detect and flag failed payment attempts (e.g., insufficient funds, card expiration).
2. User should receive an email notification with details about the failed payment and instructions to update payment information.
3. Provide users with a convenient way to update payment details or retry the payment from their account settings.
4. Admin should have access to a dashboard or report listing failed transactions and reasons for failure.

## **User Story 9: Refund Processing**

**Title:** Handle refunds for canceled orders or returned items

**Acceptance Criteria:**

1. Admin should have access to initiate refunds through the website's admin interface.
2. Refund amount should be calculated automatically based on returned items and any applicable fees.
3. System should communicate refund status to users via email, including details of refunded amount and timeline.
4. Ensure that refunded amount is credited back to the user's original payment method promptly and accurately.

## **User Story 10: Payment Security and Compliance**

**Title:** Ensure compliance with payment security standards (e.g., PCI DSS)

**Acceptance Criteria:**

1. Implement secure encryption protocols for handling payment data (e.g., SSL/TLS).

2. System should not store sensitive card information beyond what is required for transaction processing.
3. Regularly conduct security audits and vulnerability assessments to maintain compliance with PCI DSS.
4. Educate and train staff on best practices for handling payment data securely.

## **User Story 11: Payment Analytics and Reporting**

**Title:** Provide analytics and insights on payment transactions

**Acceptance Criteria:**

1. Admin should have access to reports showing key metrics such as total transactions, revenue, and payment method popularity.
2. Analytics should include trends analysis (e.g., peak transaction times, average transaction value) for strategic planning.
3. Implement visualizations like graphs and charts to present payment data in a meaningful and actionable format.
4. Ensure that payment analytics contribute to decision-making processes and business growth strategies.

## **Epic 6: Product Detail Page**

### **User Story 1: Display Product Information**

**Title:** Display essential product information on the detail page

**Acceptance Criteria:**

1. User should see the product name prominently displayed at the top of the page.
2. Include a detailed product description to provide users with comprehensive information.
3. Display pricing information, including any discounts or promotional offers clearly.
4. Ensure that product attributes such as size, color, and available variants are visible and selectable if applicable.

### **User Story 2: Product Images Gallery**

**Title:** Provide a gallery of product images on the detail page

**Acceptance Criteria:**

1. User should see a carousel or gallery displaying multiple high-resolution images of the product.
2. Allow users to click or swipe through images for a closer look.
3. Implement zoom functionality on images to enable users to view product details in more depth.
4. Ensure that images load quickly and are optimized for various devices and screen sizes.

### **User Story 3: Customer Reviews and Ratings**

**Title:** Display customer reviews and ratings for the product



### **Acceptance Criteria:**

1. Display an average rating for the product based on customer reviews (e.g., out of 5 stars).
2. Show individual customer reviews with text content and star ratings.
3. Provide options for users to filter reviews by rating, relevance, or recency.
4. Implement a form for users to leave reviews and ratings directly from the product detail page.

### **User Story 4: Add to Cart or Wishlist**

**Title:** Enable users to add the product to their cart or wishlist

#### **Acceptance Criteria:**

1. Display an "Add to Cart" button prominently on the product detail page.
2. Upon clicking "Add to Cart," update the cart icon or display a confirmation message.
3. If wishlist functionality is enabled, provide an option to add the product to the user's wishlist.
4. Ensure that users can easily access their cart or wishlist from the product detail page.

### **User Story 5: Related Products**

**Title:** Recommend related or similar products to users

#### **Acceptance Criteria:**

1. Display a section on the product detail page with related or recommended products.
2. Recommendations should be based on product attributes, user behavior, or purchase history.
3. Include options for users to view more related products or refine recommendations.
4. Ensure that related products section enhances cross-selling opportunities without overwhelming the user.

### **User Story 6: Product Availability and Shipping Information**

**Title:** Provide clear information on product availability and shipping details

#### **Acceptance Criteria:**

1. Display current stock availability status (e.g., "In Stock," "Out of Stock," "Pre-order").
2. Include estimated delivery times or shipping options available for the product.
3. Provide information on shipping costs or any applicable fees associated with the product.
4. Ensure that availability and shipping information are updated dynamically based on inventory and user location.

### **User Story 7: Social Sharing and Integration**

**Title:** Implement social sharing and integration features for the product

#### **Acceptance Criteria:**

1. Include social media sharing buttons (e.g., Facebook, Twitter) on the product detail page.
2. Allow users to share the product link with their social networks directly from the page.
3. Implement integration with social platforms to display social proof (e.g., likes, shares, comments).

4. Ensure that social sharing functionality respects user privacy preferences and data protection regulations.

## **User Story 8: Product Variants and Customization**

**Title:** Handle product variants and customization options

**Acceptance Criteria:**

1. Display dropdowns, radio buttons, or swatches for selecting product variants (e.g., size, color).
2. Update product details dynamically based on selected variants without reloading the page.
3. Allow users to customize certain product attributes (e.g., engraving, monogramming) if applicable.
4. Ensure that selected variants or customization options are reflected accurately in the cart and checkout process.

## **User Story 9: Product Comparisons**

**Title:** Enable users to compare the product with similar items

**Acceptance Criteria:**

1. Provide a "Compare" button or checkbox on the product detail page.
2. Allow users to select multiple products for side-by-side comparison.
3. Display a comparison table or visual overlay highlighting differences and similarities.
4. Ensure that comparison functionality is intuitive and supports easy decision-making for users.

## **User Story 10: Accessibility and Responsiveness**

**Title:** Ensure accessibility and responsiveness of the product detail page

**Acceptance Criteria:**

1. Implement accessible design practices to ensure compatibility with screen readers and assistive technologies.
2. Ensure that all interactive elements (e.g., buttons, links) are keyboard navigable.
3. Optimize page loading times and performance for smooth user experience across devices and network conditions.
4. Conduct usability testing to validate accessibility and responsiveness improvements.

## **Epic 7: Checkout Page**

### **User Story 1: Initiate Checkout Process**

**Title:** Initiate the checkout process from the shopping cart

**Acceptance Criteria:**

1. User should see a "Checkout" button on the shopping cart page to proceed to checkout.
2. Clicking on "Checkout" should redirect the user to the checkout page with selected items preloaded.
3. Ensure that the checkout process starts smoothly without errors or interruptions.

## **User Story 2: Display Order Summary**

**Title:** Display a summary of the order on the checkout page

**Acceptance Criteria:**

1. User should see a detailed list of items in the shopping cart with individual prices and quantities.
2. Display subtotal, shipping charges (if applicable), and total order amount prominently.
3. Ensure that any applicable taxes or fees are included in the order summary.
4. Provide a clear breakdown of costs to enhance transparency for the user.

## **User Story 3: Billing and Shipping Information**

**Title:** Collect billing and shipping information from the user

**Acceptance Criteria:**

1. Provide fields for the user to enter billing address details (e.g., name, address, phone number).
2. Implement options for users to choose a different shipping address or select from saved addresses if applicable.
3. Ensure that form fields are validated for completeness and accuracy before proceeding to the next step.
4. Display error messages or prompts for missing or incorrect information to guide users.

## **User Story 4: Select Shipping Method**

**Title:** Allow users to select a preferred shipping method

**Acceptance Criteria:**

1. Display available shipping options (e.g., standard, express) with estimated delivery times.
2. Implement radio buttons or dropdowns for users to select their preferred shipping method.
3. Calculate shipping costs dynamically based on selected method and destination.
4. Ensure that selected shipping method and associated costs are updated in the order summary.

## **User Story 5: Apply Discount Codes or Coupons**

**Title:** Provide users with the option to apply discount codes or coupons

**Acceptance Criteria:**

1. Display a field on the checkout page where users can enter a discount code or coupon.
2. Validate and apply the discount code dynamically to update the order total.
3. Ensure that applied discounts are reflected immediately in the order summary.
4. Provide feedback to users on the discount amount applied and any remaining balance.

## **User Story 6: Payment Method Selection**

**Title:** Allow users to select their preferred payment method

**Acceptance Criteria:**

1. Display available payment methods (e.g., credit/debit card, PayPal, Apple Pay) with icons and descriptions.
2. Implement radio buttons, checkboxes, or dropdowns for users to select their preferred payment method.
3. Ensure that selected payment method is clearly indicated in the checkout process.
4. Provide additional instructions or guidance for each payment method if necessary.

## **User Story 7: Guest Checkout Option**

**Title:** Enable guest checkout for users without an account

**Acceptance Criteria:**

1. Provide an option for users to proceed with checkout without creating an account.
2. Collect essential information (e.g., email, shipping address) for order processing.
3. Allow guest users to complete the checkout process securely and receive order confirmation.
4. Provide an option for guest users to create an account after completing the checkout process.

## **User Story 8: Order Review and Confirmation**

**Title:** Allow users to review and confirm their order details

**Acceptance Criteria:**

1. Display a summary of all entered information including shipping address, selected items, shipping method, and total cost.
2. Provide users with an option to edit or update information before finalizing the order.
3. Include a checkbox for users to agree to terms and conditions (if applicable).
4. Ensure that users can proceed to the payment step only after reviewing and confirming their order details.

## **User Story 9: Order Confirmation Email**

**Title:** Send order confirmation email to users after successful checkout

**Acceptance Criteria:**

1. User should receive an email immediately after completing the checkout process.
2. Confirmation email should include details such as order summary, payment information, and shipping details.
3. Ensure that order confirmation email is formatted correctly and includes branding elements.
4. Provide a link or button in the email for users to track their order status directly.

## **User Story 10: Address Validation**

**Title:** Validate shipping and billing addresses entered by users

### **Acceptance Criteria:**

1. Implement address validation using a third-party service to verify accuracy and completeness.
2. Display suggestions or corrections for users if address validation fails.
3. Ensure that validated addresses are stored correctly for order fulfillment and shipping.
4. Provide feedback to users on successful address validation and any necessary corrections.

## **User Story 11: Order Tracking**

**Title:** Provide users with order tracking information post-purchase

### **Acceptance Criteria:**

1. User should receive a confirmation page with order details immediately after successful checkout.
2. Include a link or button for users to track their order status directly from the confirmation page.
3. Implement order tracking functionality that updates in real-time based on shipping carrier information.
4. Ensure that users can access order tracking information easily from their account dashboard or through email notifications.

## **User Story 12: Save Cart Items**

**Title:** Allow users to save items in the cart for future purchase

### **Acceptance Criteria:**

1. Provide an option for users to save their cart items and return to complete the purchase later.
2. Ensure that saved cart items are accessible from the user's account and remain saved until manually removed.
3. Implement reminders or notifications for users with saved carts to encourage completion of purchase.
4. Allow users to update or remove items from the saved cart as needed before proceeding to checkout.

## **User Story 13: Multi-step Checkout Process**

**Title:** Implement a multi-step checkout process for clarity and ease of use

### **Acceptance Criteria:**

1. Divide the checkout process into distinct steps (e.g., shipping information, payment details, order review).
2. Provide a progress indicator or navigation to guide users through each step of the checkout process.
3. Ensure that users can navigate back and forth between steps to review or update information.
4. Validate and save user input at each step to prevent data loss or errors during checkout.

## **User Story 14: Mobile Optimization**

**Title:** Optimize the checkout page for mobile devices

### **Acceptance Criteria:**

1. Ensure that the checkout process is fully responsive and functional on mobile devices.
2. Adjust layout and design elements to enhance usability and readability on smaller screens.

3. Implement touch-friendly controls and interactions for seamless navigation and input.
4. Conduct usability testing on various mobile devices and browsers to ensure a smooth checkout experience.