### User Story: Online Shop Functionality

As an E-commerce Developer, I need to build the necessary functionalities to allow users to browse, select, and purchase electric bikes easily.

#### Tasks:

1. Build Product Listings Page
   * Create the "Shop" tab on the homepage.
   * Design and implement the product listing layout to show images, names, prices, and ratings of available electric bikes.
   * Integrate the products' data from the database to be dynamically displayed on the page.
2. Create Product Details Page
   * Develop the product details page template to display expanded information, specifications, and customer reviews.
   * Implement the "Add to Cart" button functionality.
   * Ensure navigation from product listing to product details page is seamless.
3. Implement Shopping Cart Functionality
   * Build the shopping cart page to display selected items, their quantities, prices, and the total amount.
   * Add features to allow users to update item quantities and remove items from the cart.
   * Ensure the cart is updated in real-time as users make changes.
4. Develop Checkout Process
   * Create a multi-step checkout process capturing shipping information, payment method selection, and order confirmation.
   * Implement form validations to ensure all necessary fields are correctly filled.
   * Integrate the checkout flow with backend services to process the order and update inventory.
5. Enable Customer Reviews and Ratings
   * Implement a review section on the product details page to display customer reviews and ratings.
   * Provide a review submission form that is accessible to users who have purchased the product.
   * Add validation to ensure only verified purchasers can submit reviews.

### User Story: Payment Methods Integration

As an E-commerce Developer, I need to securely integrate Stripe to allow users to pay for their orders.

#### Tasks:

1. Integrate Stripe Payment Gateway
   * Set up the Stripe API on the server-side to process payments.
   * Implement client-side scripts to securely capture and tokenize payment information.
   * Ensure secure transmission of payment data between the client and Stripe servers.
2. Support Multiple Currencies
   * Configure Stripe to support multiple currencies.
   * Update checkout process to allow users to select their preferred currency.
   * Ensure the total amount is displayed in the selected currency and is consistent with the selected items' prices.
3. Implement Transaction History
   * Create a feature where users can view their transaction history.
   * Store transaction details in the database after each successful payment.
   * Generate and send email receipts to users upon successful payment.

### User Story: Inventory Management

As an E-commerce Developer, I need to establish a real-time inventory management system and low stock notifications for admins.

#### Tasks:

1. Build Real-time Inventory Dashboard
   * Create an inventory dashboard for admin users to monitor stock levels.
   * Implement live updates using WebSockets or polling mechanisms to reflect real-time stock levels.
2. Set Up Stock Level Notifications
   * Develop a threshold-based notification system to alert admins when stock levels are low.
   * Implement backend logic to trigger notifications via email or in-app alerts when predefined stock levels are reached.
3. Integrate Supplier Systems
   * Identify supplier systems and APIs for inventory reordering.
   * Develop integration features to automate reorder processes from the supplier systems.
   * Ensure secure and reliable communication between the application and supplier systems.

### User Story: User Accounts and Profile Creation

As an E-commerce Developer, I need to implement user account functionalities to help users track their orders and manage preferences.

#### Tasks:

1. Set Up User Registration and Login
   * Create registration and login forms for users to create accounts and access their profiles.
   * Implement authentication mechanisms to secure user access.
   * Ensure password encryption and secure handling of user credentials.
2. Develop Profile Management
   * Create a user profile page that allows users to update personal information, view order history, and manage preferences.
   * Integrate the profile page with backend services to persist and retrieve user data.
3. Implement Order History
   * Develop a feature to display a list of previous orders within the user profile.
   * Retrieve order history data from the database and display it in a user-friendly format.
   * Ensure the order history page is secure and accessible only to authenticated users.
4. Build Wishlist and Preferences
   * Create a mechanism for users to add items to a wishlist for future consideration.
   * Develop backend support to store and manage wishlist data.
   * Integrate wishlist functionalities into the user profile for easy access.
5. Implement Password Recovery
   * Develop a "Forgot Password" feature to help users recover their accounts.
   * Implement email-based password reset process with secure token generation and validation.
   * Ensure the password reset process is secure and prevents unauthorized access.

### User Story: Customer Support

As an E-commerce Developer, I need to implement customer support features to assist users with inquiries and issues.

#### Tasks:

1. Create Contact Form
   * Develop a "Contact Us" page with a form to capture user inquiries.
   * Implement backend support to handle form submissions and route them to the support team.
2. Implement Live Chat Support
   * Integrate a live chat solution to provide real-time assistance to users.
   * Configure the live chat plugin or develop custom chat functionality.
   * Ensure the live chat system is reliable and accessible during support hours.
3. Develop FAQ Section
   * Create an FAQ page with a list of common questions and their answers.
   * Structure the FAQ content to be easily searchable and user-friendly.
   * Implement a backend system to manage and update FAQ content as needed.
4. Set Up Email Support System
   * Implement a ticketing system for users to submit support requests via email.
   * Develop backend support to handle incoming support tickets and track their resolution status.
   * Automate email notifications to inform users about the status of their support requests.

### User Stories Relevant for UX/UI Designer

#### User Story: Design and Usability

As a user, I want the web application to be modern, responsive, and user-friendly.

Acceptance Criteria:

1. Modern Design:
   * Given I am using the website,
   * When I browse different pages,
   * Then the design should reflect current trends with a clean and appealing visual style.
2. Responsive Design:
   * Given I access the website on different devices,
   * When I navigate through the application,
   * Then the layout should adjust seamlessly to desktop, tablet, and mobile screens.
3. User-Friendly Navigation:
   * Given I am a first-time visitor,
   * When I interact with the website,
   * Then navigation should be intuitive and straightforward, allowing me to find the information I need without difficulty.

### Tasks for User Story: Design and Usability

1. Modern Design
   * Task 1: Conduct competitor analysis to identify current design trends in e-commerce websites.
   * Task 2: Create a mood board and style guide reflecting the design goals.
   * Task 3: Design a high-fidelity mockup for the homepage.
   * Task 4: Design high-fidelity mockups for key pages (Shop, Product Details, Cart, Checkout).
   * Task 5: Validate mockups with stakeholders and gather feedback.
2. Responsive Design
   * Task 1: List common screen resolutions for desktop, tablet, and mobile platforms.
   * Task 2: Develop wireframes for different screen sizes.
   * Task 3: Use responsive design tools like Bootstrap or CSS Grid for prototype development.
   * Task 4: Conduct usability tests across various devices and screen resolutions.
   * Task 5: Refine the designs based on test feedback for consistency and usability.
3. User-Friendly Navigation
   * Task 1: Create a sitemap and navigation flowchart for the application.
   * Task 2: Design intuitive navigation menus and dropdowns.
   * Task 3: Develop wireframes focused on ease of navigation.
   * Task 4: Conduct usability testing sessions to observe user interactions with the navigation.
   * Task 5: Iterate on the navigation design based on usability testing insights.

#### User Story: Online Shop Functionality (Complementary to Development Work)

As a user, I want to browse, select, and purchase electric bikes easily.

Acceptance Criteria:

1. Product Listings:
   * Given I am on the homepage,
   * When I click on the "Shop" tab,
   * Then I should see a list of available electric bikes with images, names, prices, and ratings.
2. Product Details and Specifications:
   * Given I am on the product listing page,
   * When I click on a product,
   * Then I should be taken to a product details page containing expanded information, specifications, customer reviews, and an "Add to Cart" button.
3. Shopping Cart:
   * Given I have selected products,
   * When I click on the shopping cart icon,
   * Then I should see a summary of the items in my cart, their quantities, prices, and a total amount,
   * And I should be able to update quantities or remove items.

### Tasks for User Story: Online Shop Functionality

1. Product Listings
   * Task 1: Design UI components for product thumbnails (images, price, ratings).
   * Task 2: Create a grid layout that adapts to various screen sizes.
   * Task 3: Design filters and sorting options for product listings.
   * Task 4: Create an intuitive "Shop" tab layout for easy navigation.
   * Task 5: Conduct A/B testing on different listing designs to determine the most effective layout.
2. Product Details and Specifications
   * Task 1: Design the structure of the product details page with emphasis on readability and aesthetics.
   * Task 2: Incorporate interactive elements for viewing product images (e.g., image zoom, 360-degree views).
   * Task 3: Design UI components for customer reviews and ratings.
   * Task 4: Design "Add to Cart" button with clear visual cues for user interaction.
   * Task 5: Create a layout focusing on presenting comprehensive product specifications effectively.
3. Shopping Cart
   * Task 1: Design the shopping cart page layout with item summaries, quantity adjusters, and total cost.
   * Task 2: Design and incorporate buttons for updating item quantities and removing items from the cart.
   * Task 3: Design visual feedback elements for actions within the shopping cart (e.g., item added, quantity updated).
   * Task 4: Ensure cart designs are cohesive and match with the branding and overall site design.
   * Task 5: Conduct usability testing specifically for the shopping cart to ensure smooth user experience.

As a Product Manager, my role involves overseeing the project to ensure it aligns with business goals, validating target audience details, confirming timelines and budget, identifying potential third-party services, ensuring secure payment methods integration, complying with data protection regulations, and continuously monitoring the project's progress. Based on the user stories provided, the following relevant tasks are determined:

### User Story: Payment Methods Integration

Tasks:

1. Requirement Gathering for Payment Integration:
   * Conduct a meeting with stakeholders to understand detailed requirements for the payment integration with Stripe.
   * Document the entire payment process flow, including error handling and user notifications.
2. Secure Payment Gateway Integration Planning:
   * Identify security measures and best practices for integrating Stripe as a payment method.
   * Coordinate with the E-commerce Developer to ensure secure implementation of the payment gateway.
3. Multiple Currency Support:
   * Research and define the currencies that need to be supported.
   * Ensure that the exchange rates are updated regularly and correctly reflected at checkout.
4. Transaction History and Receipts:
   * Define the structure of transaction history and receipt details.
   * Work with developers to ensure that transaction details are recorded appropriately and are accessible via the user profile.
   * Coordinate with UX/UI Designer to create a user-friendly design for displaying transaction history.
5. Compliance and Security Checks:
   * Verify that all payment processes comply with PCI-DSS standards.
   * Ensure that the payment information is encrypted and securely stored.

### User Story: Inventory Management

Tasks:

1. Requirement Analysis for Real-time Inventory Tracking:
   * Meet with key stakeholders to understand the inventory management needs.
   * Document the specifications for real-time inventory tracking and stock updates.
2. System Selection and Integration:
   * Evaluate and select suitable inventory management systems that can be integrated with the web application.
   * Oversee the integration process, ensuring continuous real-time synchronization of inventory data.
3. Stock Level Notifications Setup:
   * Define the threshold levels for low stock notifications.
   * Coordinate with the development team to implement automated notifications when stock levels drop below the threshold.
4. Supplier System Integration:
   * Identify and document API requirements for integrating with supplier systems.
   * Oversee the implementation of the integration to enable seamless reordering of stock from suppliers.
5. Testing and Validation:
   * Develop a testing plan to ensure all inventory management features function correctly.
   * Monitor the system post-deployment to identify and resolve any issues quickly.

### User Story: Compliance and Security

Tasks:

1. Compliance Assessment:
   * Conduct a thorough review of GDPR, CCPA, and other relevant data protection regulations.
   * Create a compliance checklist and ensure all user data handling processes adhere to these regulations.
2. Data Security Framework:
   * Define a robust data security framework that includes encryption protocols, secure storage, and access controls.
   * Coordinate with developers to implement these security measures within the application.
3. Data Handling Policies:
   * Develop and document clear data handling policies that outline how user data will be collected, processed, and stored.
   * Ensure these policies are reviewed and approved by legal experts.
4. Audit and Monitoring:
   * Plan regular audits to verify ongoing compliance with data protection laws.
   * Implement monitoring tools to detect and respond to potential data breaches.
5. User Consent and Rights Management:
   * Ensure the system can handle user consent and GDPR rights such as the right to access, rectification, and deletion of personal data.
   * Coordinate with the UX/UI Designer to create transparent and user-friendly consent forms and data management interfaces.