Technical Requirements Document

HunarBazaar E-Commerce Marketplace

1.0 Project Workflow Description:

HunarBazaar connects artisans and buyers through a responsive marketplace. Users browse products via a Responsive frontend, while product data, customer profiles, and orders are managed in the backend using **Sanity CMS**. Integrated **third-party APIs** ensure secure payments and real-time shipment tracking, creating a seamless user experience.

2.0 Frontend Requirements

2.1 User Interface:

- User-friendly navigation with intuitive interactions.
- Culturally relevant and minimalist design tailored to the target audience.

2.2 Responsive Design:

- Fully functional on mobile devices, tablets, and desktops, ensuring cross-platform usability.

2.3 Essential Pages:

1. **Home Page**: Display featured products, a prominent search bar, and artisan highlights
2. **Product Listing Page:** Enable product browsing with categories, filters, and sorting options.
3. **Product Details Page:** Provide detailed product information, reviews, and an "Add to Cart" feature.
4. **Cart Page:** List selected products with options to edit quantity or remove items.
5. **Checkout Page:** Capture shipping information, payment details, and order summary.
6. **Order Confirmation Page:** Display confirmation, order details, and estimated delivery timeline.

3.0 Backend Requirements (Sanity CMS)

3.1 Data Management:

- **Product Data:**
  - Store product name, description, price, stock levels, categories, and images.
- **Customer Details:**
  - Maintain profiles with names, contact information, addresses, and order history.
- **Order Records:**
  - Record order details such as product IDs, customer IDs, total cost, payment status, and shipment details.

3.2 Sanity Schema Design:

- **Schemas:**
  - **Product Schema**: Fields for name, price, categories, stock, and images.
  - **Customer Schema:** Fields for contact details and purchase history.
  - **Order Schema**: Fields for order ID, customer ID, products, total cost, and delivery status
- Designed to align with business scalability and maintain consistency.

Countine on nextpage

4.0 Third-Party API Integration

4.1 Shipment Tracking:

- Enable real-time shipment tracking using APIs like ShipEngine.
- Provide accurate delivery status and estimated arrival time on the platform.

4.2 Payment Gateway:

- Ensure secure and reliable payment processing with gateways like Stripe, JazzCash, or EasyPaisa.
- Support multiple payment options for user convenience.

5.0 Testing and QA:

5.1 Unit and Integration Testing:

- Test all components for functionality, compatibility, and performance**.**

5.2 End-to-End Testing:

- Simulate user interactions across the entire workflow.

5.3 Security Testing:

- Ensure all sensitive user data is encrypted and protected.

6.0 Deployment:

6.1 Deployment

Host the platform using Vercel or an equivalent scalable cloud service**.**

Author: Khadija
Date: 17-January 2025