“Artilane”

PROJECT REQUIREMENTS

TEAM MEMBERS

Prashant Kumar Mishra (2100291530040)

Abhinav Kumar (2100291530002)

Arpit Gupta (2100291530011)

BACHELOR OF TECHNOLOGYIN

COMPUTER SCIENCE AND ENGINEERING(AI/AIML)

To create an e-commerce website for artisans using React for the frontend, Django for the backend, and MongoDB as the database, we would need the following requirements:

Frontend (React):

1. Node.js: Install Node.js to use npm (Node Package Manager) for managing dependencies and running scripts.

2. React: Set up a React application using Create React App or a similar boilerplate.

3. React Router: Implement React Router for handling navigation and routing within the application.

4. State Management: Utilize a state management library like Redux or React Context API to manage application state efficiently.

5. UI Framework: Choose a UI framework/library like Material-UI, Ant Design, or Bootstrap for building responsive and visually appealing components.

6. HTTP Requests: Use a library like Axios or Fetch API to handle HTTP requests to the backend API.

7. Authentication and Authorization: Implement user authentication and authorization features using libraries like Firebase, Auth0, or JWT (JSON Web Tokens).

Backend (Django):

1. Python and Django: Install Python and Django to set up the backend environment.

2. Django REST framework: Use Django REST framework to build a powerful and flexible API for the e-commerce functionalities.

3. Models and Database: Define Django models for artisans, products, orders, and other necessary entities. Configure the MongoDB database connection using Django MongoDB Engine or similar libraries.

4. Views and Serializers: Create views to handle API requests and serializers to convert data to and from JSON format.

5. Authentication and Authorization: Implement authentication and authorization mechanisms using Django's built-in authentication system, JWT, or other authentication libraries.

6. Payment Gateway Integration: Integrate a payment gateway such as Stripe or PayPal to facilitate secure online transactions.

Database (MongoDB):

1. Install MongoDB: Set up and install MongoDB on your server or use a cloud-based MongoDB service like MongoDB Atlas.

2. Database Design: Design the database schema to store information about artisans, products, orders, and other related data.

3. Database Connectivity: Use a MongoDB driver or an Object-Document Mapper (ODM) like Mongoose for connecting the backend Django application to the MongoDB database.

Other Requirements:

1. Hosting: Choose a hosting service provider like AWS, Heroku, or DigitalOcean to deploy your frontend and backend applications.

2. Domain and SSL: Register a domain name and obtain an SSL certificate to secure the website and provide a professional online presence.

3. SEO and Analytics: Implement search engine optimization (SEO) techniques and integrate analytics tools like Google Analytics to track website traffic and user behavior.

4. Testing and Deployment: Write unit tests and perform end-to-end testing to ensure the functionality and reliability of the website. Use CI/CD (Continuous Integration/Continuous Deployment) tools like Jenkins, CircleCI, or GitLab CI/CD for automating the testing and deployment process.

5. Scalability and Performance: Optimize the website for scalability and performance, considering factors like caching, code optimization, and efficient database queries.