|  |  |  |
| --- | --- | --- |
| **Phase** | **Task** | **Description** |
| **Backend Development** | Install Node.js and npm |  |
|  | Create a project directory and initialize it as a Node.js project | npm init |
|  | Install the necessary dependencies for chosen backend framework |  |
|  | Define the database schema for application | user, product, order |
|  | Create a database connection and implement data access methods for CRUD operations |  |
|  | Implement email notification using a Nodemailer library | Nodemailer or SendGrid |
|  | Implement payment processing using a payment gateway | PayPal or Stripe |
|  |  |  |
|  |  |  |
| **Frontend Development** | Create a React project for the frontend application |  |
|  | Define the React components for the user interface | Signup, login, product list, product detail, product review, cart, checkout, edit product, add product, profile |
|  | Use React Router to handle routing between different pages |  |
|  | State management |  |
|  | Create API calls to the backend to fetch data and perform actions |  |
|  |  |  |
|  |  |  |
| **Integration** | Deploy the backend application to a server |  |
| **Integration** | Configure the frontend application to point to the deployed backend URL |  |
|  |  |  |
| **Testing and Deployment** | Test after every function implementation |  |

**# Specs**

**## Admin**

**- Predefined in system**

**- CRUD products**

**- Process orders**

**- Receive notification by email if products are not available**

**- Receive notification by email if products are not available when a product is reviewed**

**## Customers**

**- Signup**

**- View all products**

**- Search specific products**

**- View the details of a product**

**- Add products to cart**

**- checkout cart. If the cart is checking out successfully, it should be cleared out**

**- Review products**

**- Check status of orders**

**- Check previous orders**

**- Update profile**

**# System Design**

**## Front End: React**

**\* Customers:**

**- Signup**

**- Login**

**- ProductList: Show name/price of products**

**- Product: name/price of a product. Button: Review, Add To Card**

**- ProductDetails: More information about the products. Button: Review, Add To Card**

**- Review: comment, rating**

**- Cart: See all added products, be able to update the quantity**

**- Checkout**

**- Logout**

**\* Admin**

**- Login**

**- ProductList, AddProduct, UpdateProduct**

**- Delete product**

**## BE**

**\* CRUD Users**

**\* CRUD products**

**\* Create/Read/Update orders**

**## Database**

**```JavaScript**

**const user = {**

**\_id: ObjectId**

**username: string,**

**email: String,**

**password: string**

**role: string, //admin or customer**

**deleted: boolean**

**};**

**const product = {**

**\_id: ObjectId,**

**name: String,**

**description: String,**

**price: Number,**

**quantity: Number,**

**images: [], //should be url**

**category: String,**

**averageRatings: number,**

**reviews: [**

**{**

**userId: ObjectId,**

**text: String,**

**rating: number**

**}**

**],**

**deleted: boolean**

**};**

**const order = {**

**\_id: ObjectId,**

**userId: ObjectId,**

**items: [**

**{**

**productId: ObjectId,**

**quantity: Number,**

**price: Number**

**}**

**],**

**totalAmount: Number,**

**orderDateTime: string,**

**payment: {**

**method: string, //cash, debit, credit**

**amount: number**

**},**

**status: string, //ordered, processing, delivered, canceled**

};

```

- Check how to implement payment with Paypal

- Email notification with nodejs using nodemailer.