EcoShop - Sustainable Product Marketplace: Phase 1-3 Plan

Phase 1: Initial Setup and Basic Authentication

1. Project Setup

- Initialize a new MERN stack project.
- Set up the folder structure for client and server.
- Install necessary dependencies (React, Redux, Node, Express, MongoDB, etc.).

2. User Authentication

- Implement user registration with secure password hashing.
- Implement user login and JWT-based authentication.
- Create user profiles and authentication middleware.

Schemas:

User Schema:

```
{
  username: { type: String, required: true, unique: true },
  email: { type: String, required: true, unique: true },
  password: { type: String, required: true },
  createdAt: { type: Date, default: Date.now }
}
```

Routes:

- POST /api/auth/register
- POST /api/auth/login

Phase 2: Product Management

1. Product Model and API

- Design the product schema for MongoDB (name, description, price, category, images, sustainability certif

- Create CRUD operations for products (create, read, update, delete).

- Implement routes and controllers for product management.

2. Product Listing Page

- Develop a product listing page to display all products.

- Implement pagination for better performance.

- Display product details including images and sustainability certifications.

Schemas:

Product Schema:

```
{
  name: { type: String, required: true },
  description: { type: String, required: true },
  price: { type: Number, required: true },
  category: { type: String, required: true },
  images: [String],
  sustainabilityCertifications: [String],
  createdAt: { type: Date, default: Date.now }
}
```

Routes:

- POST /api/products

- GET /api/products

- GET /api/products/:id

- PUT /api/products/:id

- DELETE /api/products/:id

Phase 3: Shopping Cart and Checkout

1. Shopping Cart Functionality

- Create a shopping cart model and API.
- Allow users to add, update, and remove products from the cart.
- Store cart data in the database and sync with the user's session.

2. Checkout Process

- Integrate Stripe for secure payment processing.
- Implement a checkout page where users can review their cart and enter payment details.
- Handle order creation and payment confirmation.

Schemas:

Cart Schema:

```
{
  user: { type: mongoose.Schema.Types.ObjectId, ref: 'User', required: true },
  products: [{
    product: { type: mongoose.Schema.Types.ObjectId, ref: 'Product', required: true },
    quantity: { type: Number, required: true, default: 1 }
  }],
  createdAt: { type: Date, default: Date.now }
}
```

Order Schema:

```
{
  user: { type: mongoose.Schema.Types.ObjectId, ref: 'User', required: true },
  products: [{
    product: { type: mongoose.Schema.Types.ObjectId, ref: 'Product', required: true },
    quantity: { type: Number, required: true }
  }],
  amount: { type: Number, required: true },
  paymentStatus: { type: String, required: true },
  createdAt: { type: Date, default: Date.now }
}
```

Routes:

- POST /api/cart
- GET /api/cart
- PUT /api/cart/:id
- DELETE /api/cart/:id

- POST /api/orders
- GET /api/orders
- GET /api/orders/:id