**Furniture E-commerce Application**

**Overview**

This project is a single-page e-commerce application for a furniture store, featuring essential functionalities such as user authentication, product listing, and cart management. It is designed to provide a seamless user experience with a clean and professional interface.

**Features**

1. **User Authentication**:
   * Secure Signup and Login functionality with form validation.
   * JWT-based authentication for secure access.
2. **Product Listing**:
   * Displays products dynamically fetched from the database.
   * Responsive grid layout with hover animations for product cards.
3. **Cart Management**:
   * Add and remove products from the cart.
   * Dynamically updates the cart count in real-time.

**Tech Stack**

* **Frontend**: React.js, Material-UI
* **Backend**: Node.js, Express.js
* **Database**: PostgreSQL
* **Authentication**: JWT (JSON Web Tokens)

**Prerequisites**

* [Node.js](https://nodejs.org/) (version 14+)
* [PostgreSQL](https://www.postgresql.org/download/) (version 12+)

**Setup Instructions**

**1. Clone the Repository**

bash

Copy code

git clone https://github.com/nethragovinda/furniture-ecommerce.git

cd furniture-ecommerce

**2. Frontend Setup**

1. Navigate to the frontend folder:

bash

Copy code

cd frontend

1. Install dependencies:

bash

Copy code

npm install

1. Start the development server:

bash

Copy code

npm start

1. The frontend will run at http://localhost:3000.

**3. Backend Setup**

1. Navigate to the backend folder:

bash

Copy code

cd backend

1. Install dependencies:

bash

Copy code

npm install

1. Create a .env file in the backend folder with the following variables:

env

Copy code

DATABASE\_URL=postgresql://postgresql:test@localhost:5432/ecommerce

JWT\_SECRET=your\_secret\_key

PORT=5000

1. Set up the database:
   * Open your PostgreSQL shell and create the database:

sql

Copy code

CREATE DATABASE ecommerce;

* + Run the provided SQL file (backend/database/schema.sql) to create tables:

bash

Copy code

psql -U postgresql -d ecommerce -f backend/database/schema.sql

* + Optionally, seed the database with initial data:

bash

Copy code

psql -U postgresql -d ecommerce -f backend/database/seed.sql

1. Start the backend server:

bash

Copy code

npm run dev

1. The backend will run at http://localhost:5000.

**4. Testing the Application**

1. Open your browser and navigate to:
   * **Frontend**: http://localhost:3000
   * **Backend API (Optional)**: Test endpoints using [Postman](https://www.postman.com/).
2. Use the app to:
   * Register a new user.
   * Log in with your credentials.
   * Browse the product listing.
   * Add/remove items from the cart.

**Project Structure**

**Frontend**

* **/src/components**: Reusable UI components.
* **/src/pages**: Full-page components like Login, Signup, Products, and Cart.
* **/src/api**: API integration for backend communication.

**Backend**

* **/routes**: API route definitions.
* **/controllers**: Route logic and business logic.
* **/db**: Database configuration and queries.
* **/middleware**: Middleware for authentication and validation.

**Environment Variables**

| **Variable** | **Description** |
| --- | --- |
| DATABASE\_URL | PostgreSQL connection string |
| JWT\_SECRET | Secret key for JWT token generation |
| PORT | Port on which the backend server will run |

**Troubleshooting**

1. **Frontend Issues**:
   * If the frontend doesn’t load, ensure the backend server is running.
   * Check for any errors in the browser console.
2. **Database Issues**:
   * Verify that PostgreSQL is running.
   * Check the database connection string in the .env file.

**Contact**

For any questions or additional details, please feel free to contact me:

* **Name**: Nethra Govinda
* **Email**: nethragovinda@gmail.com