Glow up Beauty - E-Commerce Marketplace

Welcome to the **MarkeetPlace E-Shop** repository! This project is a dynamic and fully functional e-commerce platform designed for purchasing beauty products, built with modern web practices and a focus on user experience and scalability.

🚀 Features

- **User Management:**
  - Secure sign-up and login system.
  - Personal user dashboards for account management.
- **Product Management:**
  - Dynamic product listings with real-time updates.
  - Categorization and search functionality for products.
- **Cart & Checkout:**
  - Add-to-cart functionality with smooth checkout process.
  - Secure payments using Stripe.
- **Wishlist:**
  - Users can save their favorite products for later review or purchase.
- **Order & Shipment Tracking:**
  - Automated stock updates post-purchase.
  - Shipment tracking for delivered products.

🛠 Tech Stack

- **Frontend:**
  - [Next.js](#)
  - [Tailwind CSS](#)
- **Backend:**
  - [Sanity CMS](#)
  - REST APIs
- **Database:**

o   Sanity CMS for content management.
- **Payments:**
    - o   [Stripe](#) for secure payment processing.

📁 Folder Structure

```
├── public/            # Static assets
├── src/
│   ├── components/    # Reusable UI components
│   ├── pages/         # Frontend pages (Next.js routing)
│   ├── styles/        # Global and Tailwind CSS files
│   └── utils/         # Utility functions and API integrations
└── sanity/            # Sanity CMS configuration
```

⚙️ Setup

1. **Clone the Repository:**
2. `git clone https://github.com/your-username/glow-by-beauty.git`
   `cd glow-by-beauty`

3. **Install Dependencies:**

   `npm install`

4. **Set Up Environment Variables:**
   Create a `.env` file in the root directory and add the following keys:
5. `NEXT_PUBLIC_SANITY_PROJECT_ID=your_sanity_project_id`
6. `NEXT_PUBLIC_SANITY_DATASET=your_dataset`
   `STRIPE_SECRET_KEY=your_stripe_key`

7. **Run the Development Server:**

   `npm run dev`

8. **Access the Application:**
   Open your browser and navigate to `http://localhost:3000`.

🗂 Data Schemas

Product Schema

```
{
  name: 'product',
  fields: [
    { name: 'title', type: 'string' },
    { name: 'price', type: 'number' },
   { name: 'category', type: 'reference', to: [{ type: 'category' }] },
  ]
}
```

User Schema

```
{
  name: 'user',
  fields: [
    { name: 'name', type: 'string' },
    { name: 'email', type: 'string' },
    { name: 'password', type: 'string' },
    { name: 'orders', type: 'array', of: [{ type: 'reference', to: [{ type:
'order' }] }] },
  ]
}
```

Order Schema

```
{
  name: 'order',
  fields: [
    { name: 'user', type: 'reference', to: [{ type: 'user' }] },
    { name: 'items', type: 'array', of: [{ type: 'reference', to: [{ type:
'product' }] }] },
    { name: 'status', type: 'string' },
    { name: 'shipmentDetails', type: 'object', fields: [
      { name: 'address', type: 'string' },
      { name: 'tracking', type: 'string' },
    ]},
  ]
}
```

📖 How It Works

1. Users sign up or log in to access their dashboard.
2. Products are dynamically fetched and displayed from Sanity CMS.
3. Users can browse, add products to their cart, and proceed to checkout.
4. Stripe handles payments securely.
5. Users can track their order and view shipment details.

🤝 Contribution

Contributions are welcome! Fork the repository, create a branch, and submit a pull request with your improvements.

📄 License

This project is licensed under the MIT License.

❄️ Acknowledgments

Special thanks to Next.js, Tailwind CSS, Sanity CMS, and Stripe for the tools and services that made this project possible.