**📄 SMF Jewels – Full Stack eCommerce Web App Documentation**

**🔰 Project Overview**

**Title:** SMF Jewels eCommerce Web App  
**Description:** A professional-grade, fully functional eCommerce platform for jewelry products, including both user and admin functionalities, built with modern, job-ready technologies.

**🚀 Tech Stack**

| **Layer** | **Technology** |
| --- | --- |
| Frontend | React.js or Next.js + TailwindCSS |
| Backend | FastAPI (Python) |
| Database | MongoDB (Atlas) |
| Auth | JWT (JSON Web Token) |
| Deployment | Vercel (frontend), Render (backend) |
| Storage | Cloudinary (for product images) |

**🎯 Project Goals**

* Build a production-grade eCommerce website
* Support full product browsing and purchasing flows
* Admin features for product and order management
* Secure authentication using JWT
* Live deployment for portfolio and job showcase

**📁 Folder Structure**

smf-ecommerce/

├── backend/ → FastAPI backend

│ ├── main.py

│ ├── routes/ → API endpoints

│ ├── models/ → MongoDB schemas

│ ├── schemas/ → Pydantic schemas

│ ├── controllers/ → Logic layer

│ ├── core/ → DB config, JWT utils

│ └── .env

│

├── frontend/ → React or Next.js frontend

│ ├── pages/

│ ├── components/

│ ├── styles/

│ ├── utils/

│ └── .env

│

└── README.md

**🔐 Auth System**

* Signup/Login with hashed passwords
* JWT token-based authentication
* Admin and User roles
* Auth-protected routes (frontend + backend)

**📦 Key Features**

**User Features:**

* Register/Login
* Browse products
* View product details
* Add to cart
* Checkout and place order
* View order history

**Admin Features:**

* Add/edit/delete products
* View all users
* View and manage all orders
* Dashboard with metrics (future scope)

**⚙️ Backend API Routes (FastAPI)**

| **Route** | **Method** | **Auth** | **Description** |
| --- | --- | --- | --- |
| /api/auth/signup | POST | Public | Register new user |
| /api/auth/login | POST | Public | Login and get JWT token |
| /api/products/ | GET | Public | Get all products |
| /api/products/{id} | GET | Public | Get product by ID |
| /api/products/ | POST | Admin | Create a new product |
| /api/products/{id} | PUT | Admin | Edit product |
| /api/products/{id} | DELETE | Admin | Delete product |
| /api/cart/ | GET | User | Get user's cart |
| /api/cart/ | POST | User | Add item to cart |
| /api/orders/ | POST | User | Place an order |
| /api/admin/orders | GET | Admin | View all orders |

**🧱 Frontend Pages**

| **Route** | **Description** |
| --- | --- |
| / | Homepage with featured products |
| /products | Product listing with filters |
| /product/:id | Product detail page |
| /cart | Cart view and modify |
| /checkout | Checkout flow |
| /login | Login form |
| /register | Signup form |
| /profile | User profile and orders |
| /admin/dashboard | Admin dashboard |
| /admin/products | Admin product manager |
| /admin/orders | Admin order manager |

**🗃️ Database Models (MongoDB)**

**users**

{

"\_id": "uuid",

"name": "string",

"email": "string",

"password": "hashed",

"role": "user" | "admin",

"created\_at": "timestamp"

}

**products**

{

"\_id": "uuid",

"name": "string",

"price": "number",

"category": "string",

"images": ["url1", "url2"],

"description": "string",

"stock": "number",

"created\_by": "admin\_id"

}

**orders**

{

"\_id": "uuid",

"user\_id": "user\_id",

"items": [{ "product\_id": "id", "qty": 2 }],

"total": "number",

"status": "pending" | "shipped" | "delivered",

"created\_at": "timestamp"

}

**🚀 Deployment Plan**

| **Component** | **Platform** | **Notes** |
| --- | --- | --- |
| Frontend | Vercel | GitHub integration, env vars |
| Backend | Render | Free FastAPI deployment |
| Database | MongoDB | MongoDB Atlas cloud database |
| Storage | Cloudinary | Product image uploads |

**🗓️ Timeline / Milestones**

| **Week** | **Goals** |
| --- | --- |
| 1 | Setup + Python/React refresh + Git |
| 2 | FastAPI backend + MongoDB + CRUD |
| 3 | Frontend UI + Tailwind + API integration |
| 4 | Auth (JWT) + protected routes |
| 5 | Cart + Order system + Admin panel |
| 6 | Deployment + Testing + Polish + LinkedIn post |

**💼 Outcome**

* ✅ Ready for resume and LinkedIn
* ✅ Live deployed full-stack product
* ✅ Shows backend, frontend, auth, and database skills
* ✅ Can be reused for job interviews, freelancing, and clients