Database Documentation - Clothing E-Commerce

1. Overview

The project uses **MongoDB** as its primary data store, managed through the **Mongoose ODM**. It follows a document-based schema design optimized for e-commerce workflows.

2. Technology Stack

- **Database:** MongoDB

- **ODM:** Mongoose 8.21.0

- **Management:** Docker-based MongoDB instance with a `db-manager` for backups and scheduling.

3. Data Models (Schemas)

User Model

- `name` (String, Required)

- `email` (String, Required, Unique)

- `password` (String, Required)

- `timestamps` (CreatedAt, UpdatedAt)

Product Model

- `name` (String, Required)

- `description` (String)

- `price` (Number, Required)

- `imageUrl` (String)

- `category` (String, Enum: Men, Women, Kids, etc.)

- `sizes` (Array of Strings, Enum: S, M, L, XL, etc.)

- `isActive` (Boolean, Default: True)

Order Model

- user (ObjectId, Ref: 'User')

`items`: Array of:

- `product` (ObjectId, Ref: 'Product')

- `size` (String)

- `quantity` (Number)

- `price` (Number)

- `totalPrice` (Number)

- `orderDate` (Date)

Analytics Models

- **UserAnalytics:** Tracks user registration, login frequency, and spending habits.

- **ProductAnalytics:** Tracks views and purchase frequency for individual products.

4. Relationships

- **One-to-Many:** A User can have multiple Orders.

- **Many-to-Many (via Orders):** Products appear in many Orders, and Orders contain many Products.

- **Embedded Documents:** Cart items and Order items are embedded within their respective parent documents for performance.