

E-commerce API – Design Report

Project Overview

This project implements an E-commerce API using FastAPI, SQLAlchemy, and MySQL to manage Products and Orders efficiently. The API provides endpoints for creating, updating, retrieving, and deleting products and orders while ensuring data validation and security.

Design Decisions

Database Choice: MySQL

- Why MySQL?
 - Reliable and scalable relational database management system.
 - Supports transactions, indexing, and efficient query handling.
 - Easily integrates with FastAPI and SQLAlchemy.
- ORM Selection: SQLAlchemy
 - Combines SQLAlchemy and Pydantic for database interaction and data validation.
 - Provides an easy-to-use declarative syntax.
 - Ensures type safety and data integrity.

API Structure

- **Separation of Concerns:**
 - main.py → Handles API requests and routes.
 - models.py → Defines database schemas and validation models.
 - db_config.py → Manages database connection.
 - create_db.py → Handles database table creation.
 - **Pydantic Validation:**
 - Ensures only valid data is accepted for API requests.
 - Prevents SQL injection and data corruption.
-

Implementation Overview

Data Models (models.py)

- **Product:**
 - id (Primary Key)
 - name (String)

- description (String)
- price (Float)
- **Order:**
 - id (Primary Key)
 - customer_id (Integer)
 - product_ids (String - stores multiple product IDs as comma-separated values)
 - status (Default: "pending")

API Features (main.py)

- **Product Management:** Create, Read, Update, Delete products.
- **Order Management:** Create, Read, Update, Delete orders.
- **Update Restrictions:**
 - **Products** → Only name & description can be updated.
 - **Orders** → Only status can be updated.

Database Management (db_config.py, create_db.py)

- Uses SQLAlchemy to define schemas and create tables in MySQL.
- Auto-generates tables based on schema definitions.

API Documentation & User Guide

API Endpoints Overview

Products API

Method	Endpoint	Description
POST	/products/	Create a new product
GET	/products/{id}	Get product by ID
PUT	/products/{id}	Update name & description only
DELETE	/products/{id}	Delete a product

Orders API

Method	Endpoint	Description
POST	/orders/	Create a new order
GET	/orders/{id}	Get order by ID

Method	Endpoint	Description
PUT	/orders/{id}/status	Update order status
DELETE	/orders/{id}	Delete an order

Interactive API Documentation

- FastAPI provides built-in API documentation:
 - **Swagger UI:** <http://127.0.0.1:8000/docs>
 - **Redoc UI:** <http://127.0.0.1:8000/redoc>
-

Conclusion

This FastAPI E-commerce API provides a scalable, secure, and efficient solution for managing products and orders. With SQLAlchemy & MySQL, it ensures data integrity and validation, making it a solid foundation for further enhancements.