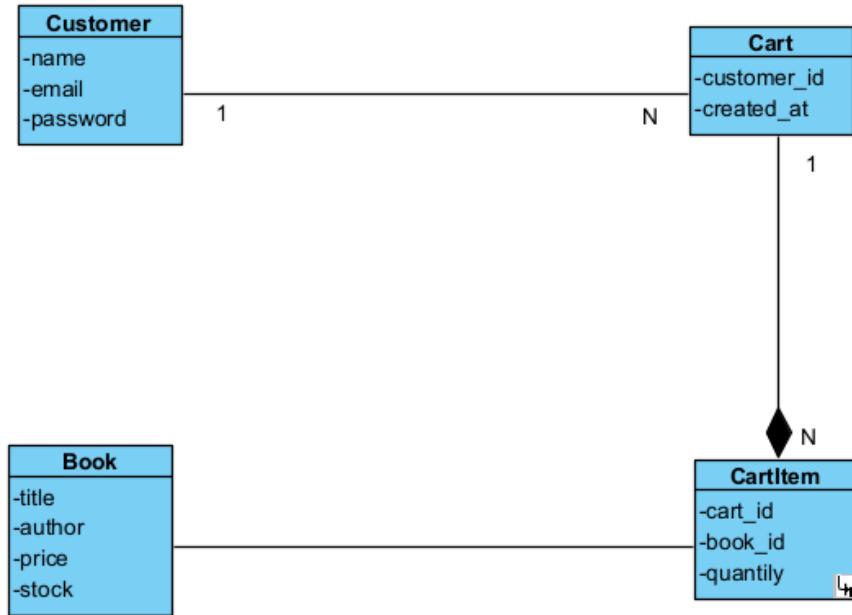


## 5.1. Class Diagram (Biểu đồ lớp)

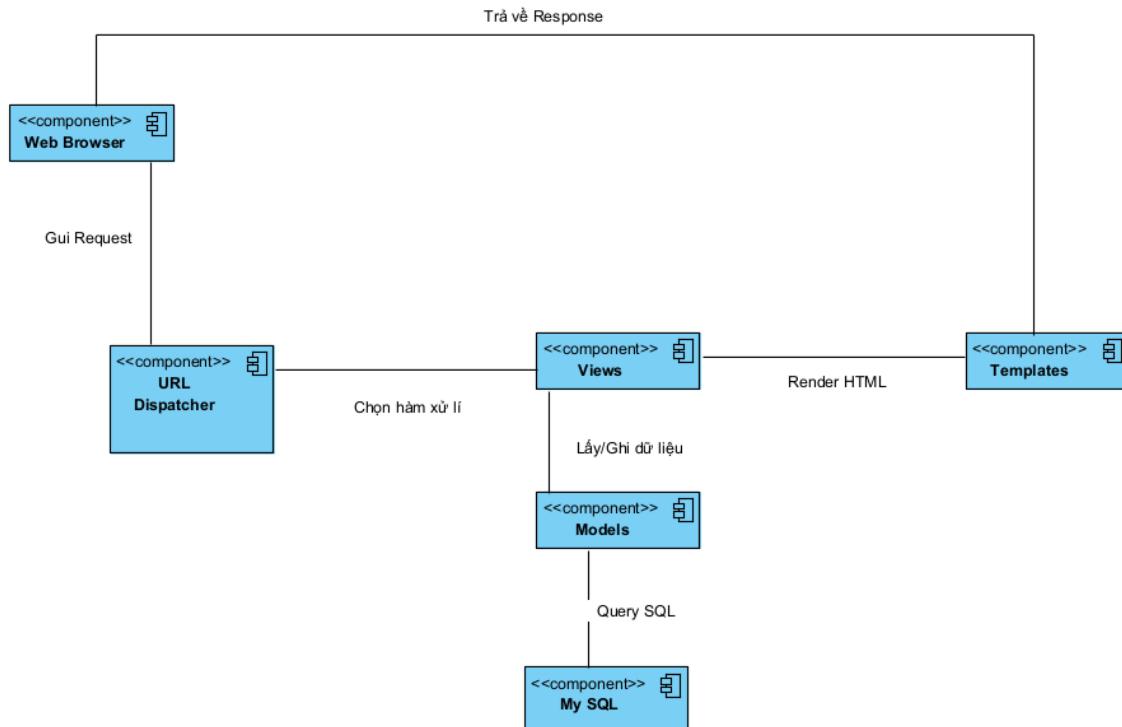


Mô tả kỹ thuật: Biểu đồ thể hiện cấu trúc dữ liệu cốt lõi của hệ thống, bao gồm các thực thể (Entities) và mối quan hệ (Relationships):

- **Customer**: Lưu trữ thông tin định danh người dùng.
- **Book**: Quản lý thông tin sản phẩm (tên, tác giả, giá, tồn kho).
- **Cart**: Đại diện cho phiên mua sắm của khách hàng (Quan hệ 1-1 hoặc 1-n với Customer).
- **CartItem**: Chi tiết sản phẩm trong giỏ hàng (Quan hệ n-1 với Cart và n-1 với Book).

- Mối quan hệ: Sử dụng quan hệ Composition/Aggregation để thể hiện sự phụ thuộc giữa Cart và CartItem.

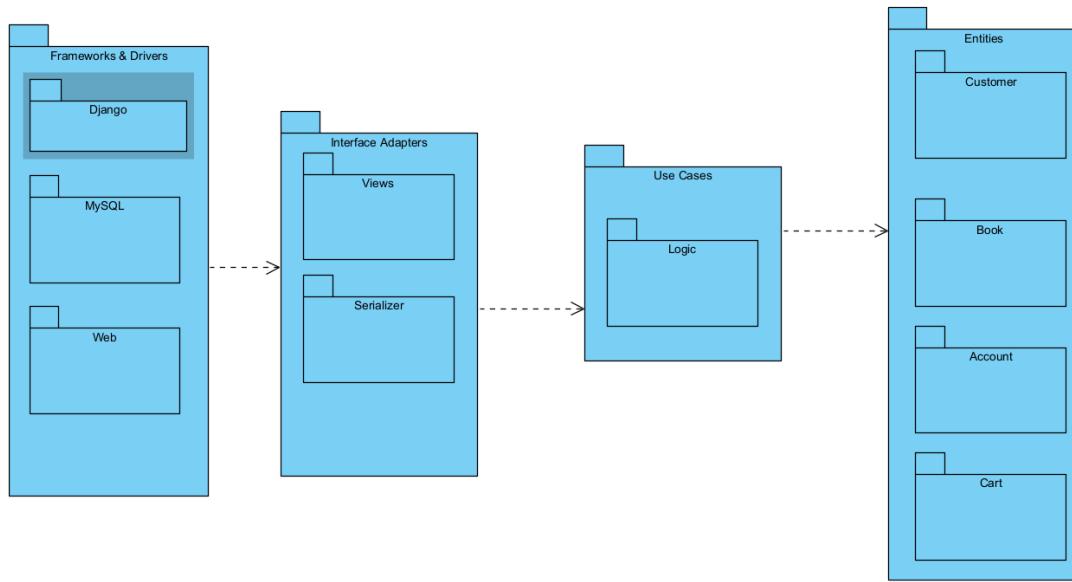
### MVC Layer Diagram (Version A - Monolithic)



Mô tả kiến trúc: Phiên bản Monolithic được xây dựng dựa trên kiến trúc MVT (Model-View-Template) mặc định của Django:

- Model: Tương tác trực tiếp với MySQL Database.
- View: Xử lý logic nghiệp vụ và điều hướng request.
- Template/API: Trả về dữ liệu JSON hoặc giao diện HTML.
- Đặc điểm: Tất cả các module (Auth, Book, Cart) nằm chung trong một codebase duy nhất, giao tiếp qua lời gọi hàm nội bộ (Function Call), dùng chung một Database.

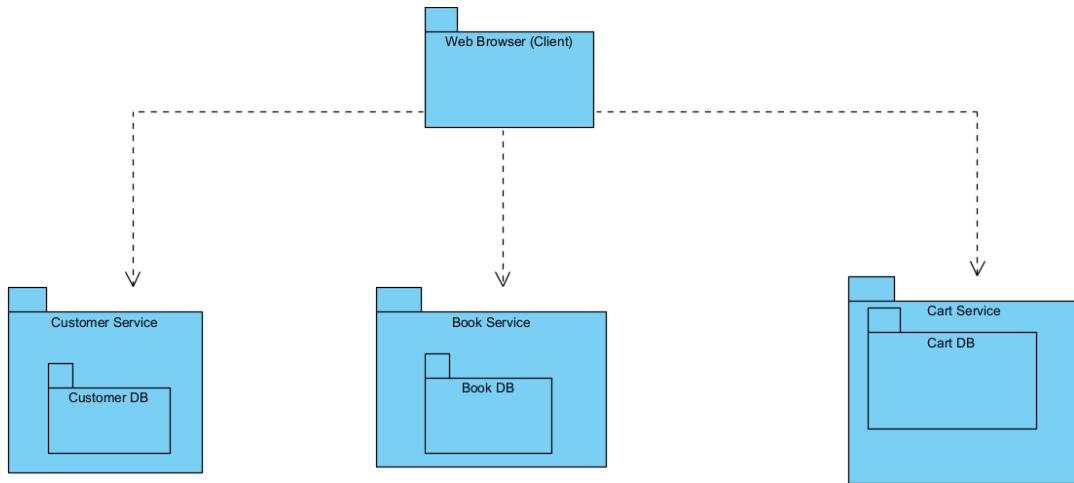
### 5.3. Clean Architecture Diagram (Version B)



Mô tả kiến trúc: Hệ thống được tái cấu trúc theo nguyên lý Dependency Rule (Nguyên lý phụ thuộc), chia thành các tầng đồng tâm:

1. Domain Entities (Core): Chứa các class POJO (Plain Old Java/Python Objects) như Book, Cart. Không phụ thuộc vào bất kỳ framework nào.
  2. Use Cases (Application Business Rules): Chứa logic nghiệp vụ thuần túy (ví dụ: AddToCartUseCase). Tầng này điều phối luồng dữ liệu vào/ra các Entities.
  3. Interface Adapters (Infrastructure): Chứa Repositories để giao tiếp với Database thông qua Django ORM.
  4. Frameworks & Drivers (Outer Layer): Chứa Django Views, Serializers.
- **Lợi ích:** Tách biệt hoàn toàn Business Logic khỏi Framework, giúp code dễ kiểm thử (Unit Test) và bảo trì.

#### 5.4. Microservices Architecture Diagram (Version C)



Mô tả kiến trúc: Hệ thống được phân rã thành 3 dịch vụ độc lập (Distributed System):

1. Auth Service (Port 8001): Chịu trách nhiệm quản lý Identity và Access của người dùng.
2. Book Service (Port 8002): Đóng vai trò là Catalog Service, quản lý kho sách.
3. Cart Service (Port 8003): Quản lý giao dịch giỏ hàng.

Cơ chế giao tiếp (Communication):

- Sử dụng giao tiếp đồng bộ (Synchronous Communication) thông qua RESTful HTTP API.
- Khi người dùng thêm sách vào giỏ, Cart Service gửi HTTP Request sang Book Service để xác thực thông tin sách trước khi lưu vào Database riêng biệt.