



Lab-week-2 - java www

Lập trình phân tán với công nghệ Java (Trường Đại học Công nghiệp Thành phố Hồ Chí Minh)



Scan to open on Studeersnel

LAB WEEK 2

Back-end

Một công ty cần lập một **API** cho trang web bán hàng của họ. Thông tin cần lưu trữ bao gồm:

- employee (**emp_id**, full_name, dob, email, phone, address, status)
- product (**product_id**, name, description, unit, manufacturer_name, status)
- customer (**cust_id**, cust_name, email, phone, address)
- product_image (product_id, **image_id**, path, alternative)
- orders (**order_id**, order_date, emp_id, cust_id)
- order_detail (**order_id**, **product_id**, quantity, **price**, note)
- product_price (**product_id**, **price_date_time**, price, note)

Diễn giải:

- Một product có nhiều image, một image thuộc về một product. Status chỉ trạng thái kinh doanh của sản phẩm: 1- đang kinh doanh, 0 - tạm ngưng, -1 - không kinh doanh nữa.
- Employee có status: 1- đang làm việc, 0 - tạm nghỉ, -1 – nghỉ việc.
- Một order có nhiều order_detail, một order_detail thuộc về một order.
- Một order thuộc về một employee, một employee có nhiều order.
- Một customer có nhiều order, một order chỉ thuộc một customer.
- Một product_detail có một giá (price) được lưu trong product_price. Một giá được xác định bằng product_id và price_date_time. Tại thời điểm lập order, **giá được lấy với price_date_time gần nhất.**

Lập **REST API** cho các thao tác cần thiết của yêu cầu này (dùng JakartaEE):

- Các thao tác CRUD cho các đối tượng, lập order. (dùng JPA).
- Thống kê order theo ngày, theo khoảng thời gian.
- Thống kê order theo nhân viên bán hàng trong 1 khoảng thời gian.
- ...

Font-end

- Tạo các trang web cho việc hiển thị sản phẩm, chọn vào giỏ hàng và thanh toán (giả lập việc thanh toán qua thẻ nếu có thể)
- Vẽ biểu đồ giá theo thời gian
- *** và các chức năng khác tự nghĩ ra.

HƯỚNG DẪN

Kiến trúc ứng dụng

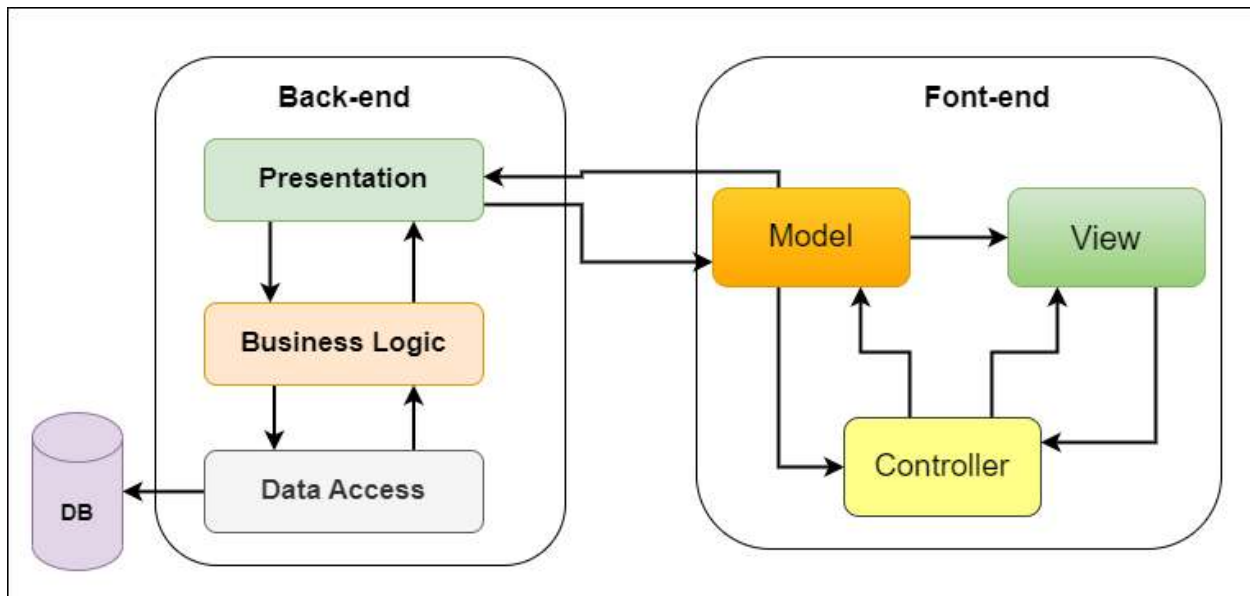


Figure 1: Application architecture

Back-end

Thiết kế theo kiến trúc 3-tiers. Trong đó:

- Tầng Data Access nhận nhiệm vụ truy xuất cơ sở dữ liệu.
- Tầng Business Logic là nơi xử lý nghiệp vụ của ứng dụng.
- Tầng Presentation đảm bảo làm giao diện cho phần back-end.

Cấu trúc project

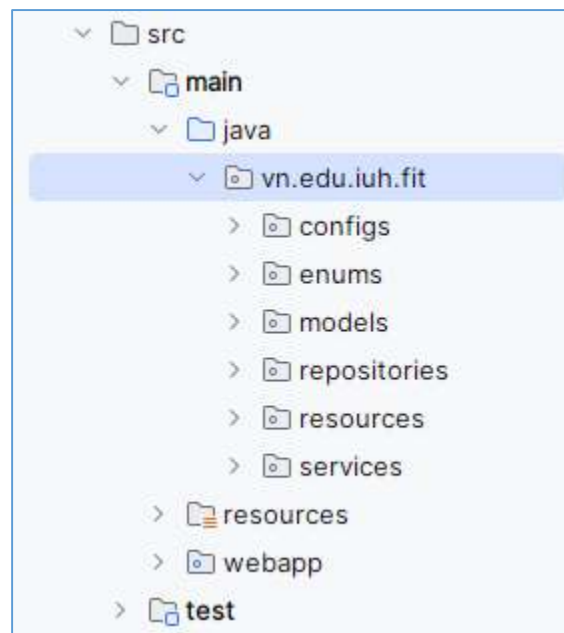


Figure 2: Project structure

Các packages được ánh xạ thành các tầng của kiến trúc như sau:

- Gói **repositories** chứa các lớp của tầng Data Access.
- Gói **services** chứa các lớp của tầng Business Logic.
- Gói **resources** chứa các lớp của tầng Presentation (sử dụng REST API).

Các gói phụ trợ khác gồm có:

- Gói **models** là nơi chứa các entities. Các lớp này nhận nhiệm vụ ánh xạ các bảng trong cơ sở dữ liệu đồng thời cũng là các đối tượng mang dữ liệu trao đổi giữa các tầng.
- Gói **configs** chứa các lớp dùng cho việc cấu hình ứng dụng.
- Gói **enums** chứa các tập hằng số cho các đối tượng. Trong ví dụ này, đó là hai tập hằng số chỉ các trạng thái của nhân viên (employee) và sản phẩm (product).

Entities design

Thiết kế các entities ánh xạ các bảng dữ liệu và thiết lập các mối quan hệ

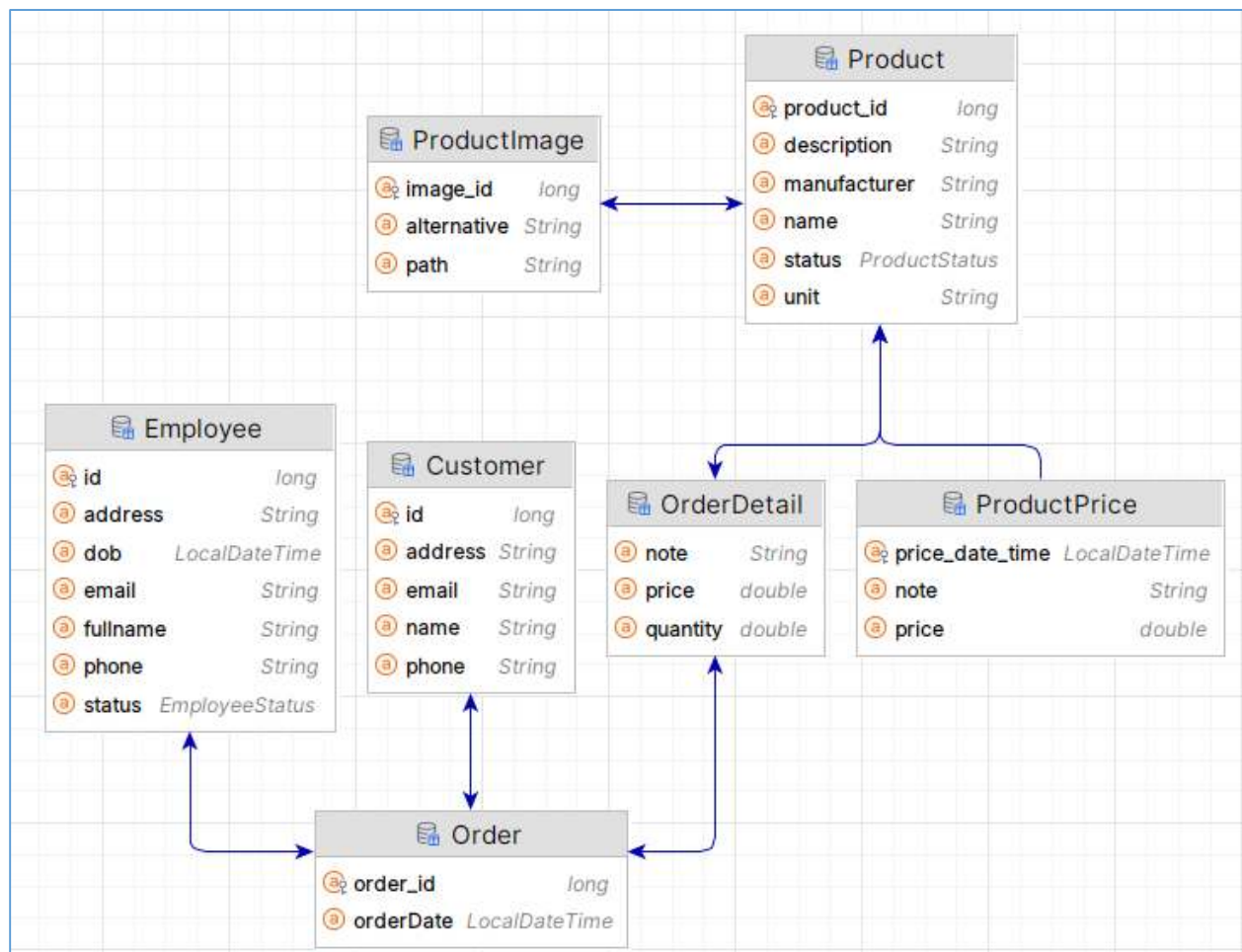


Figure 3: Persistence Unit view of entites

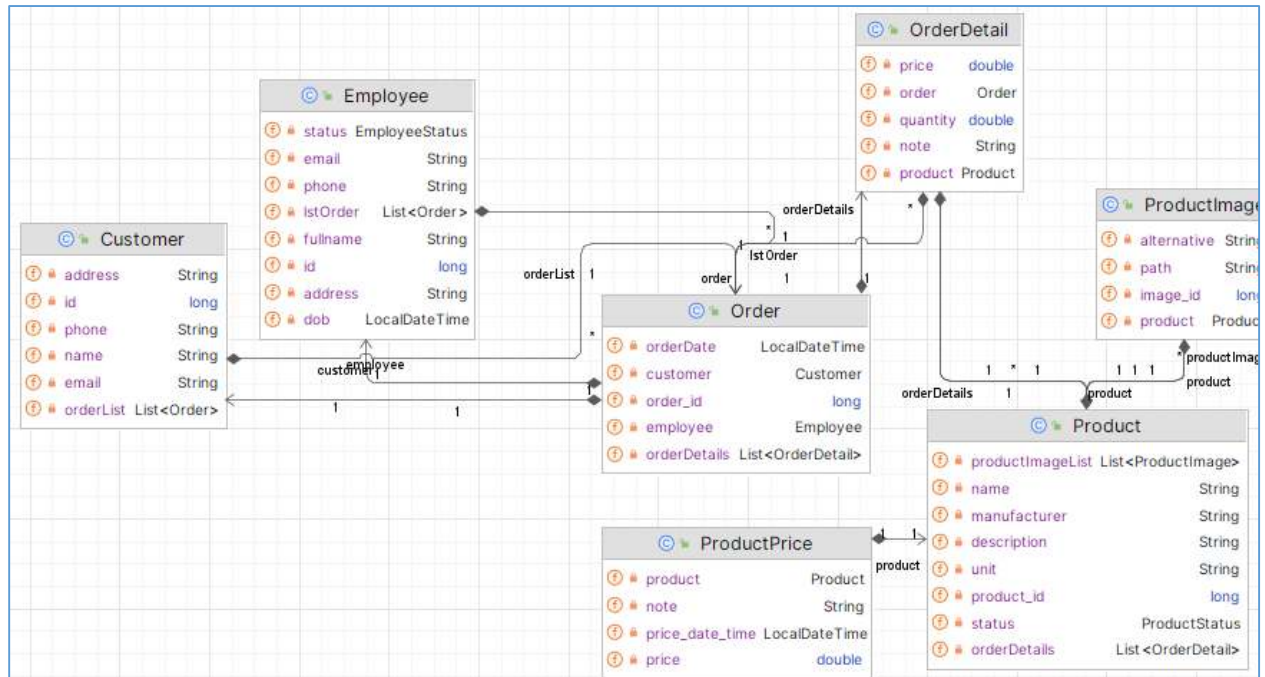


Figure 4: Class diagram of entities

Sau khi thiết kế và thực thi, database sẽ có Entity Relationship diagram như hình sau:

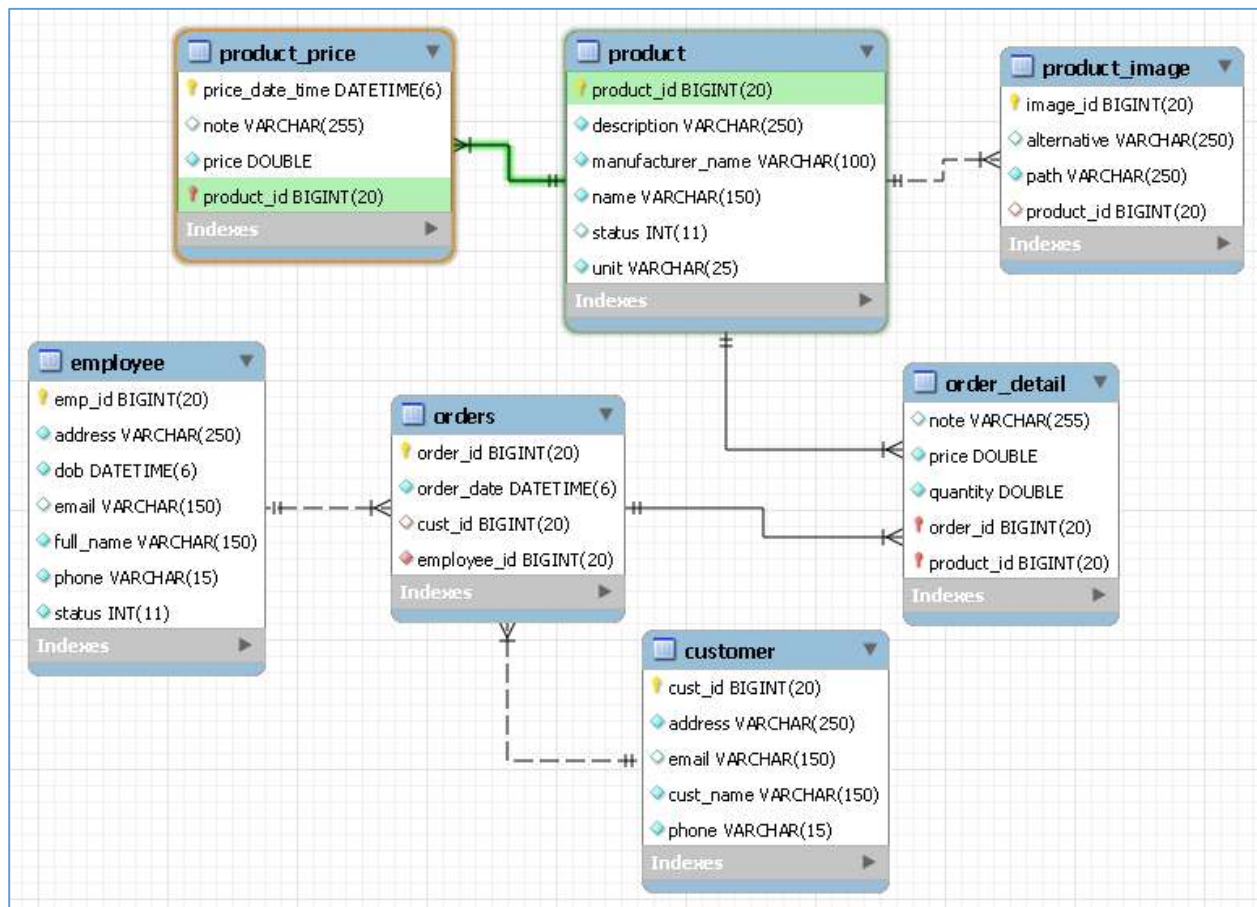


Figure 5: Entity Relationship Diagram