**Phát triển phần mềm mã nguồn mở**

Yêu cầu về công nghệ:

1. Thời gian triển khai: 4 tuần (từ tuần 13 - 16)
2. Số lượng service : 4
3. Các ngôn ngữ lập trình backend: ASP.NET Core, Python
4. Cơ sở dữ liệu: SQL Server
5. Nền tảng mở: Docker

**1 Yêu cầu:**

* Tạo 1 website bán sách. Cho phép đăng nhập đăng ký tài khoản khách hàng để mua sản phẩm. Mỗi sản phẩm được phân loại theo thể loại để người mua dễ dàng tìm kiếm. Một sản phẩm cho phép có nhiều thể loại. Hỗ trợ chức năng giảm giá. Có chức năng quản lý số lượng sản phẩm, thông báo nhắc nhở khi số lượng gần hết hoặc đã hết. Hỗ trợ giỏ hàng và thanh toán. Cho phép khách hàng đánh giá sản phẩm. Hỗ trợ các chức năng tìm kiếm, lọc giá sản phẩm.

**2 Chức năng hệ thống:**

* **Cửa hàng theo mô hình B2C**: Website cần có 2 phần giao diện người dùng gồm: Admin và Khách hàng. 2 Giao diện tách bạch nhau nhưng có đường dẫn liên kết với nhau.
* **Tài khoản:** Yêu cầu bảo mật về thông tin tài khoản, mật khẩu, token, ... Cho phép được khóa tài khoản khách hàng. Có khu vực đăng nhập, đăng ký tài khoản khách hàng
* **Sản phẩm**: Mỗi sản phẩm có thể chọn nhiều thể loại. Hỗ trợ chức năng giảm giá.
* **Đơn đặt hàng:** Yêu cầu tích hợp giỏ hàng. Yêu cầu đăng nhập trước khi thanh toán. Cho phép 2 hình thức thanh toán COD hoặc paypal. Khách hàng được phép hủy khi đơn hàng chưa được giao và hoàn tồn sản phẩm. Cho phép khách hàng xem lịch sử mua hàng trên hệ thống.
* **Giao hàng:** Chỉ giao hàng trong khu vực nội thành HCM => Miễn phí vận chuyển
* **Tồn kho, Nhập kho, xuất kho:** Mỗi khi sản phẩm hết hàng trong kho, thì phải nhập thêm số sản phẩm qua phiếu nhập.Hoàn tồn sau mỗi giao dịch nhập hàng bị hủy. Thông báo mỗi khi số lượng sản phẩm gần hết hoặc đã hết
* **Hỗ trợ tìm kiếm sản phẩm, tìm kiếm năng cao: giá, bán chạy,..**
* **Có khu vực đánh giá sản phẩm**
* **Thống kê báo cáo doanh số bán hàng, số lượng sản phẩm bán, tồn kho, ...**

**3 Đối tượng người dùng**

Gồm 2 đối tượng: Admin và Khách hàng

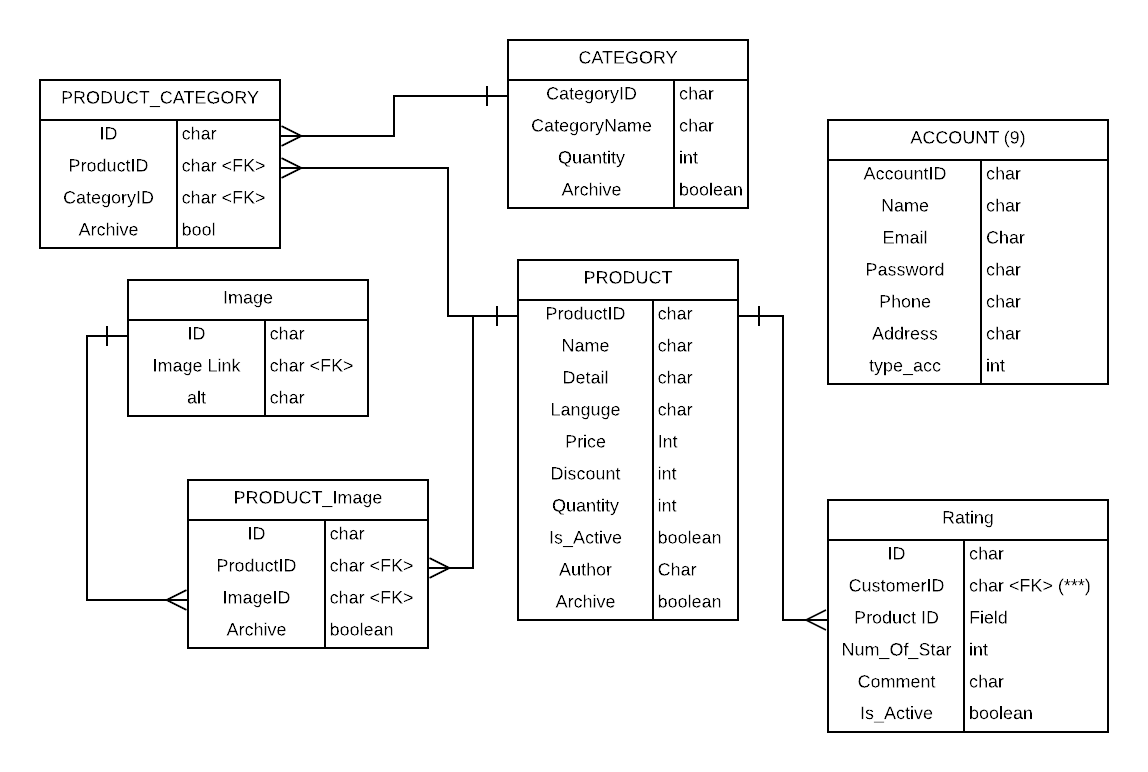
**4 Phân chia Service**

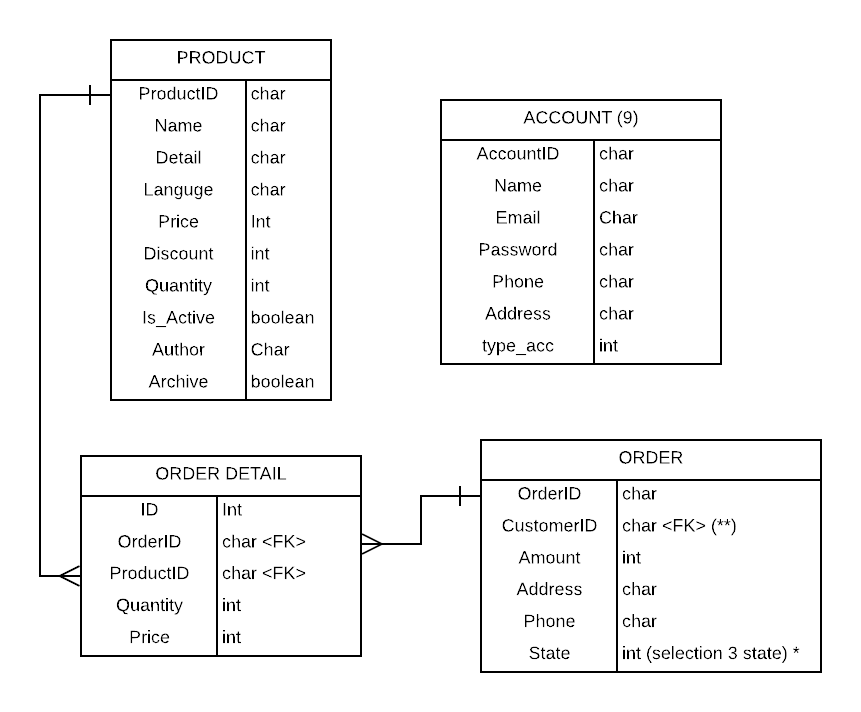
Phân chia theo chức năng, hệ thống gồm có 3 Service: Product Service, Order Service, Stock Service. Mỗi service hoạt động độc lập, không phụ thuộc với nhau

Ngoài ra còn 1 Service dùng để render dữ liệu cho giao diện; cập nhật hoặc khôi phục dữ liệu. nhằm đảm bảo tính đúng đắn của dữ liệu: BookStore Service.

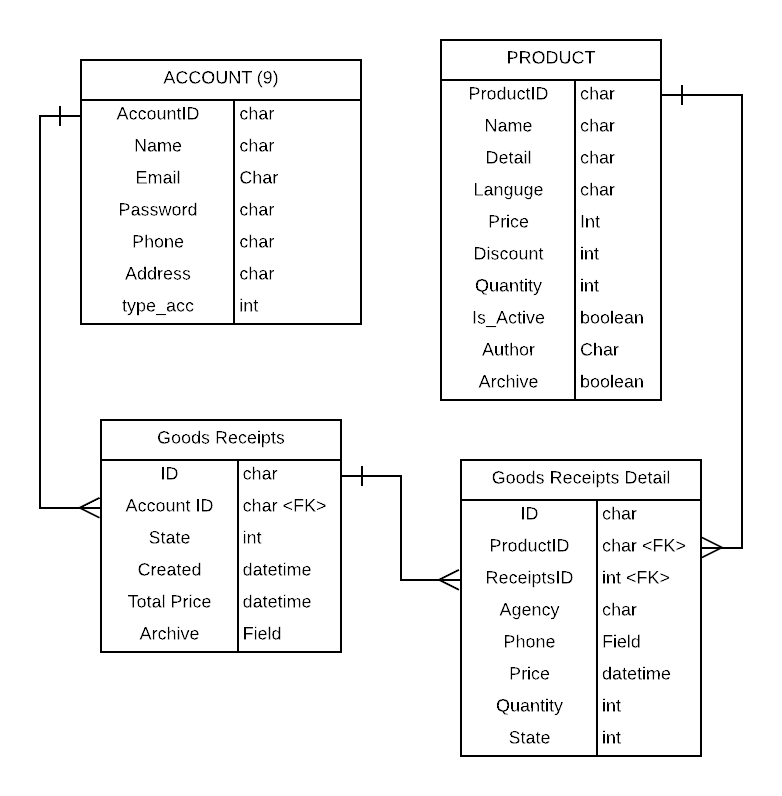
**5 Thiết kế Cơ sở dữ liệu**

+ Product Service:

+ Order Service:



+ Stock service:



**6 Giao diện người dùng**

Giao diện người dùng gồm: giao diện bán hàng (Website) và giao diện quản lý (Admin)

Giao diện Bán hàng (Website):

* Trang chủ: Hiển thị các sản phẩm nổi bật, bán chạy trong hệ thống
* Trang thể loại: Hiển thị danh sách sản phẩm thuộc cùng 1 thể loại. Sử dụng phân trang.
* Trang chi tiết sản phẩm: Hiển thị thông tin sản phẩm, xuất ra danh sách sản phẩm liên quan. Có khu vực đánh giá sản phẩm
* Trang tìm kiếm: Hiển thị danh sách sản phẩm phù hợp với từ khóa hoặc mức giá. sử dụng phân trang
* Trang giỏ hàng: Hiển thị danh sách sản phẩm đã chọn mua.
* Trang thanh toán: Hiện thị form thông tin khách hàng cùng số tiền cần phải trả cho giỏ hàng.
* Trang thông tin cá nhân: Hiện thị thông tin cá nhân của khách hàng. Hiển thị form đổi mật khẩu tài khoản
* Trang Lịch sử mua hàng: Hiển thị các đơn hàng đã mua của khách hàng

Giao diện quản lý (Admin)

* trang chủ: Hiển thị số liệu bán hàng, doanh số bán hàng theo thời gian.
* Các trang thêm sửa khóa tài khoản
* Các trang thêm sửa ngừng bán sản phẩm
* Các trang thêm sửa khóa thể loại
* Các trang thêm sửa đơn hàng
* Trang xem thông tin Đánh giá
* Trang xem thông tin kho hàng
* Các trang thêm sửa xóa phiếu nhập

**7 Phân tích chức năng từng Service**

Ba Service (Product, Order, Stock) đều được cài đặt chung 1 thư viện tạo Token. với thời hạn sử dụng 1 ngày. Ba thư viện này đều dùng chung 1 mã bí mật để giả mã. Mỗi khi tài khoản đăng nhập vào hệ thống. với email và password đã tạo. Hệ thống sẽ tạo chuỗi token cho user đăng nhập. và được lưu trữ trong session hoặc localStorage (browser)

Service Product: Sử dụng (ASP.NET Core + Entity Core)

* Quản lý sản phẩm
* Quản lý thể loại
* Quản lý đánh giá
* Xác thực, phân quyền
* Thống kê số lượng sản phẩm, trạng thái sản phẩm,...

Service Order: Sử dụng (ASP.NET Core + Entity Core)

* Quản lý đơn hàng
* Cập nhập tình trạng đơn hàng
* Xác thực, phân quyền
* Thống kê báo cáo doanh số, số lượng đơn hàng thành công, thất bại,...

Service Stock: Sử dụng (ASP.NET Core + Entity Core)

* Kiểm tra số lượng tồn kho của từng sản phẩm
* Nhập thêm số lượng sản phẩm.
* Xác thực, phân quyền
* Thống kê số lượng tồn theo thời gian

Service BookStore: Sử dụng (Django Framework)

* Render dữ liệu cho giao diện người dùng cuối.
* Thực hiện các cập nhật (update / rollback) khi thực hiện các hành động liên quan tới 2 service trở lên. Nhằm đảm bảo tính đúng đắn của dữ liệu
* Xác thực phân quyền

**8 Triển khai docker:**

Hệ thống gồm 5 container:

* Container chạy Product Service
* Container chạy Order Service
* Container chạy Stock Service
* Container chạy BookStore
* Container SQL Server

**9 Phân công công việc**

Mận: Thiết kế giao diện website và admin.

Hằng: Xây dựng Service Product

Hậu : Xây dựng Service Order

Hiển : Xây dựng Service Stock

Trưởng : Xây dựng Service BookStore

Testing & Deploy Docker: Cả nhóm

10 Tổng kết