

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY  
UNIVERSITY OF TECHNOLOGY  
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



## CÔNG NGHỆ PHẦN MỀM (CO3001)

---

Bài tập lớn

# RESTAURANT POS 2.0

---

GVHD: Lê Đình Thuận  
Nhóm sinh viên thực hiện: Nguyễn Quang Anh - 1912594  
Cù Đỗ Thanh Nhân - 1914424  
Nguyễn Hữu Phúc - 1914698  
Hoàng Minh Tiến - 1915470  
Lâm Thiện Toàn - 1915540  
Ngô Đức Trí - 1915656  
Nguyễn Phúc Vinh - 1915940

HO CHI MINH CITY, OCTOBER 2021



## PHÂN CHIA CÔNG VIỆC

Mã số sinh viên	Họ và tên	Phân công
1912594	Nguyễn Quang Anh	1. Hiện thực trang chủ của nhà hàng, thanh điều hướng AppBar. 2. Hiện thực tính năng hiển thị các món ăn được mua nhiều nhất tại cửa hàng. 3. Thiết kế giao diện cho trang chủ
1914424	Cù Đỗ Thanh Nhân	1. Hiện thực giao diện thông tin chi tiết đồ ăn 2. Hiện thực giao diện thêm đồ ăn vào giỏ hàng 3. Hiện thực chức năng xem thông tin chi tiết đồ ăn 4. Hiện thực chức năng thêm đồ ăn vào giỏ hàng 5. Hiện thực chức năng chỉnh sửa đồ ăn trong giỏ hàng 6. Hiện thực chức năng thanh toán online qua PayPal
1914698	Nguyễn Hữu Phúc	1. Hiện thực giao diện hiển thị món trên trang chủ. 2. Hiện thực chức năng tính phí ship và ước tính thời gian giao hàng dựa vào khoảng cách từ nhà hàng đến nhà của khách (sử dụng MapBox API).
1915470	Hoàng Minh Tiến	1. Hiện thực giao diện và chức năng thanh toán. 2. Hiện thực giao diện Login và chức năng xác thực đăng nhập. 3. Hiện thực giao diện Quản lý đơn hàng, hiển thị danh sách đơn hàng và các chức năng xử lý đơn hàng.
1915540	Lâm Thiện Toàn	1. Thiết kế giao diện cho trang khám phá món ăn và phân trang. 2. Thực hiện tính năng tìm kiếm món ăn theo tên. 3. Thực hiện tính năng phân loại món ăn theo mục. 4. Thực hiện tính năng sắp xếp món ăn theo giá tiền. 5. Thực hiện tính năng phân trang cho từng mục món ăn. 6. Thêm records cho mô hình cơ sở dữ liệu.
1915656	Ngô Đức Trí	1. Hiện thực chức năng Xử lý đơn hàng 2. Hiện thực chức năng Chi tiết đơn hàng 3. Thiết kế giao diện Quản lý đơn hàng, hiển thị danh sách đơn hàng và chi tiết đơn hàng 4. Thiết kế giao diện Chỉnh sửa giỏ hàng 5. Tạo cơ sở dữ liệu trên mongoDB Cloud
1915940	Nguyễn Phúc Vinh	Hiện thực chức năng cho giỏ hàng Hiện thực các chức năng cho Nhân viên: thêm/xoá/chỉnh sửa thông tin đồ ăn Thiết kế cơ sở dữ liệu Thiết kế kiến trúc hệ thống theo mô hình phân lớp



## Mục lục

<b>1</b>	<b>Kiến trúc cho hệ thống</b>	<b>3</b>
1.1	Sơ lược về lý thuyết kiến trúc phân lớp . . . . .	3
1.2	Kiến trúc lớp đề xuất . . . . .	4
1.3	Sơ lược về các Implementation diagram . . . . .	5
1.4	Mô hình cho toàn hệ thống . . . . .	6
1.5	Component diagram . . . . .	7
1.5.1	Ở phía Frontend . . . . .	7
1.5.2	Ở phía Backend . . . . .	8

# 1 Kiến trúc cho hệ thống

## 1.1 Sơ lược về lý thuyết kiến trúc phân lớp

Trong một hệ thống phân lớp, một lớp có các đặc điểm sau:

- Phụ thuộc vào các lớp bên dưới
- Không can thiệp và không phụ thuộc vào các lớp trên

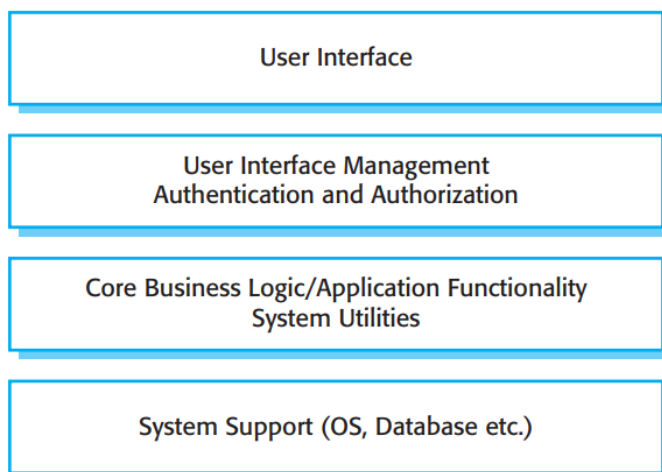
Các lớp được thiết kế theo hai kiểu:

- Một lớp chỉ có thể hiểu và sử dụng lớp ngay dưới nó
- Một lớp có thể sử dụng các layer song song (cross-cutting) và tất cả các lớp bên dưới. Kiểu thiết kế này có tên gọi là Anti-pattern: Lasagna Architecture, được sử dụng nhiều hơn kiểu thứ nhất, tránh trường hợp sử dụng lớp trung gian chỉ để truyền message dẫn đến giảm hiệu suất hệ thống.

Kiến trúc phân lớp gồm ba lớp chính (hình 1):

- User Interface: Giao diện người dùng, nơi thao tác với ứng dụng
- Core Business Logic/Application Functionality/System Utilities: Nơi chứa các module xử lý logic nghiệp vụ của hệ thống, chuyên tiếp nhận các request từ client, xử lý và lưu trữ dữ liệu thông qua Data source layer
- System Support (OS, Database etc.)

Vai trò của lớp Authentication & Authorization tùy vào ứng dụng nên không bắt buộc phải có.

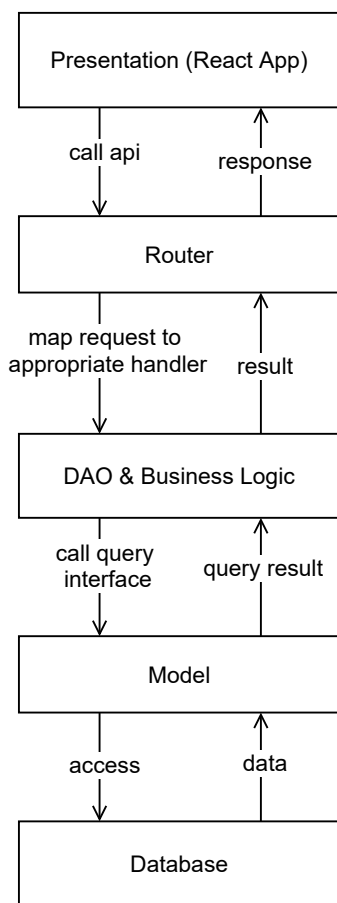


Hình 1: Kiến trúc phân lớp

## 1.2 Kiến trúc lớp đề xuất

Nhóm lựa chọn mô hình kiến trúc phân lớp (layered architecture) cho hệ thống nhà hàng POS v2.0. Kiến trúc gồm 5 lớp:

- Lớp Presentation: đây là lớp sẽ trả về giao diện cho người dùng cũng như xử lý các thao tác của người dùng như điền form, nhấn nút,... Ở mức độ implementation, lớp này sẽ chứa các file trong React App.
- Lớp Router: đây là lớp định tuyến có nhiệm vụ xử lý các http request gọi từ lớp presentation. Cụ thể, dựa vào từng request cụ thể mà lớp này sẽ map với từng handlers được cung cấp bởi lớp DAO và Business Logic ở phía dưới.
- Lớp DAO và Business Logic: đây là lớp truy cập dữ liệu và xử lý các nghiệp vụ như xác thực (authentication) hoặc tiền xử lý các dữ liệu trước khi trả về kết quả. Lớp này sẽ hiểu rõ các schema trong Database để có thể truy xuất hợp lý dựa vào các interfaces được cung cấp bởi lớp Model phía dưới
- Lớp Model: đây là lớp sẽ cung cấp các interface truy xuất dữ liệu trong Database cho lớp DAO và Business Logic. Các interface điển hình có thể kể đến như find, findById,...
- Lớp Database: Đây là nơi lưu trữ dữ liệu của toàn bộ hệ thống.



Hình 2: Kiến trúc lớp đề xuất

### 1.3 Sơ lược về các Implementation diagram

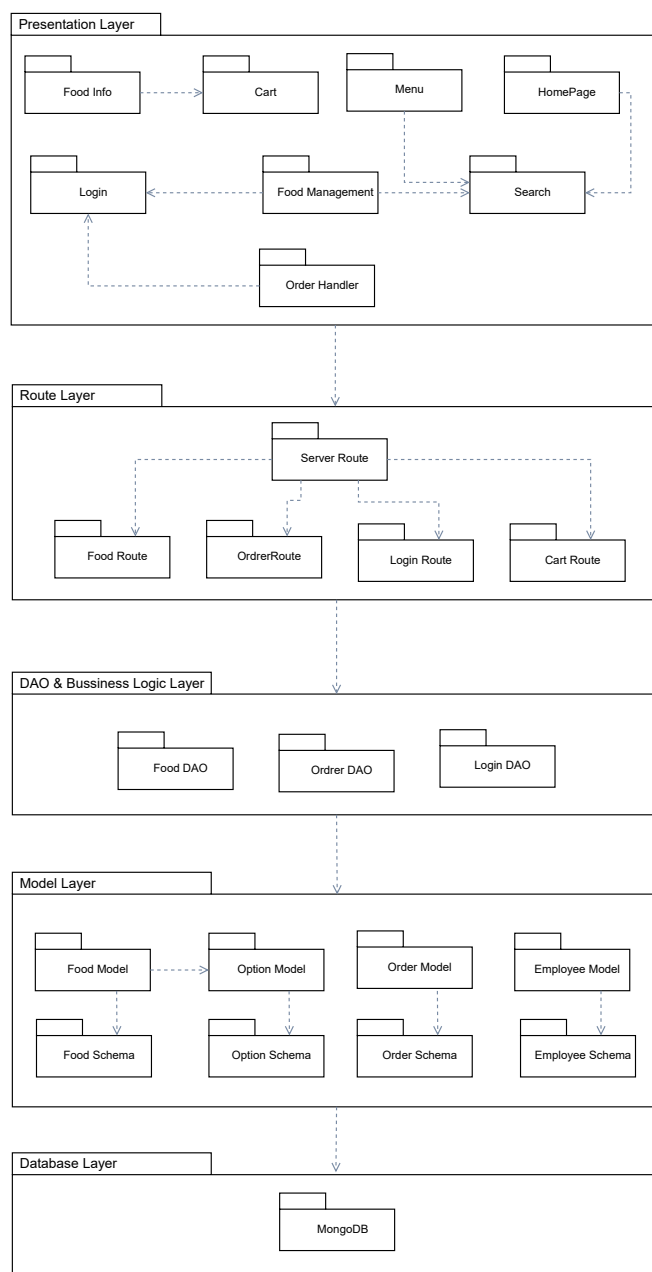
Implementation diagram bao gồm ba loại mô hình khác nhau có thể kể đến như:

- Deployment diagram là một sơ đồ cấu trúc thể hiện kiến trúc của hệ thống như việc triển khai (phân phối) các tạo tác phần mềm đến các mục tiêu triển khai.
- Package diagram là sơ đồ cấu trúc UML thể hiện cấu trúc của hệ thống được thiết kế ở cấp độ các gói. Các phần tử sau thường được vẽ trong một sơ đồ gói: package, packageable element, dependency, element import, package import, package merge.
- Component diagram hiển thị các thành phần, các giao diện được cung cấp và bắt buộc, các cổng và mối quan hệ giữa chúng. Loại sơ đồ này được sử dụng trong Phát triển dựa trên thành phần (CBD) để mô tả các hệ thống có Kiến trúc hướng dịch vụ (SOA).

Mỗi loại mô hình đều có những đặc điểm và ý nghĩa riêng. Để cho đơn giản, nhóm đã chọn package diagram để mô tả kiến trúc chung cho toàn hệ thống và 2 component diagrams để thể hiện các components chính ở phía backend và frontend.

## 1.4 Mô hình cho toàn hệ thống

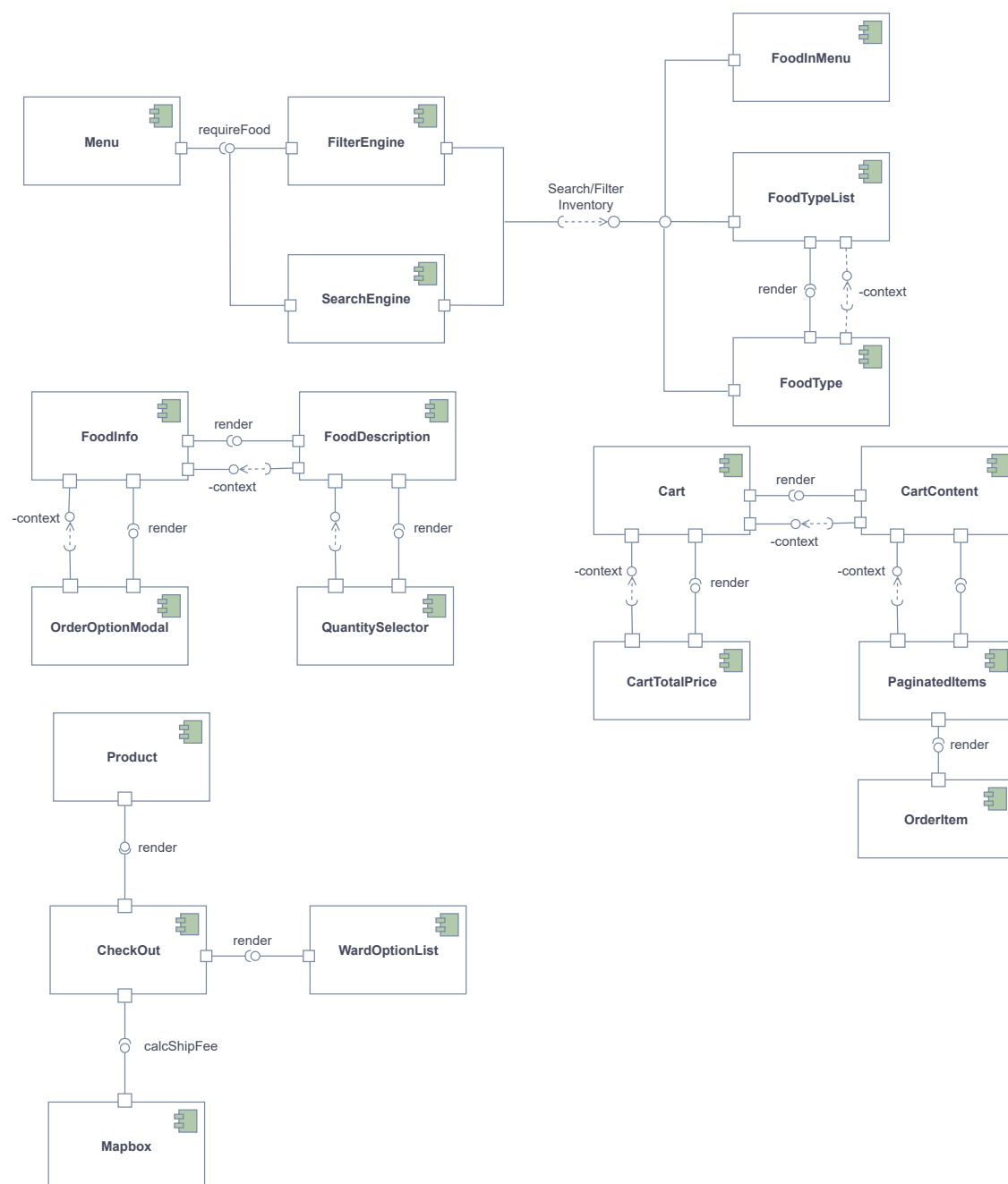
Dựa vào kiến trúc lớp mà nhóm đã đề xuất ở trên, nhóm sẽ mô tả rõ hơn những package lớn nằm trong từng lớp thông qua package diagram ở hình 3.



Hình 3: Package Diagram cho toàn hệ thống

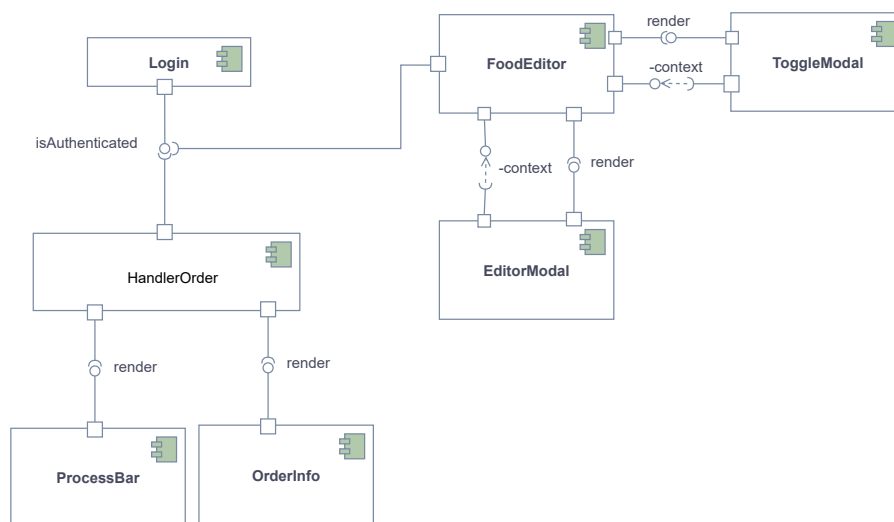
## 1.5 Component diagram

### 1.5.1 Ở phía Frontend



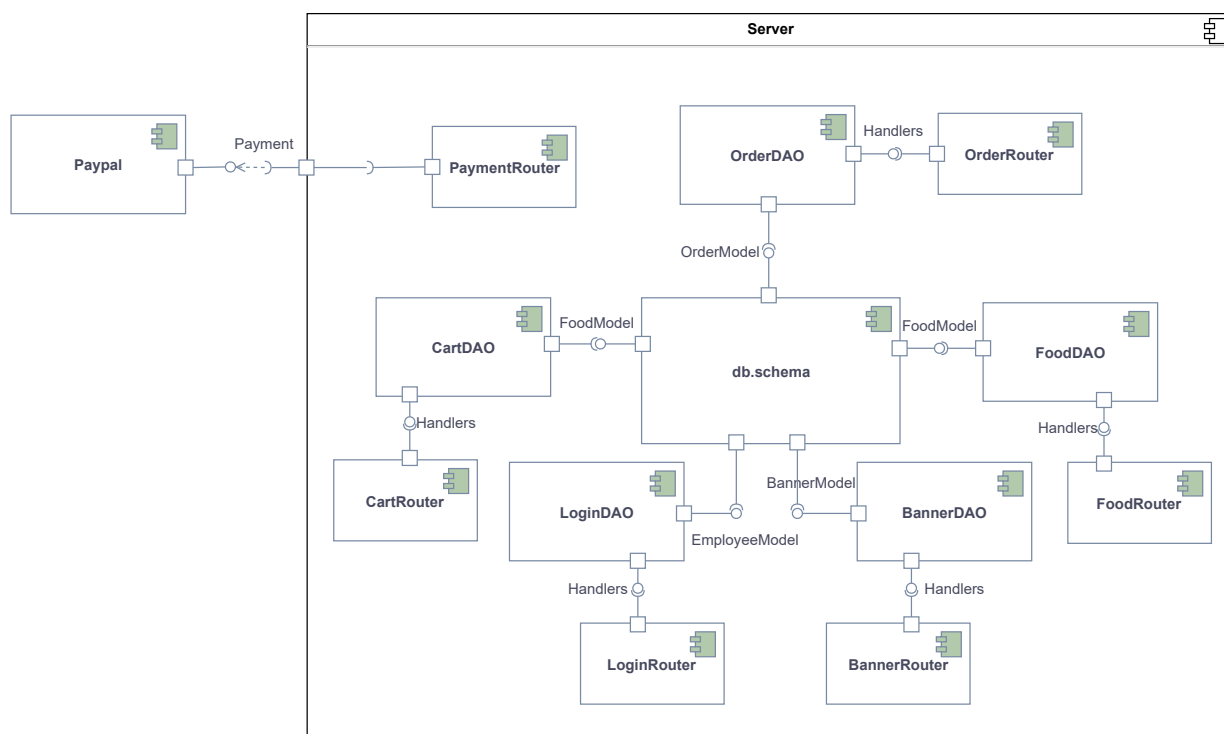
Hình 4: Component diagram Frontend (phần 1)





Hình 5: Component diagram Frontend (phần 2)

### 1.5.2 Ở phía Backend



Hình 6: Component diagram Backend