

C4 Diagram Level 1 - System Context Diagram

1. Tổng quan – System Context Diagram (Quickship Platform)

Mục tiêu:

System Context Diagram là cấp độ tổng quan nhất trong C4 Model, mô tả **Quickship Platform** trong toàn bộ bối cảnh hệ sinh thái giao vận.

Biểu đồ này tập trung vào:

- Actors (Người dùng):** Ai đang tương tác với hệ thống?
- External Systems (Hệ thống bên ngoài):** Quickship tích hợp với dịch vụ nào?
- Relationships (Quan hệ):** Luồng dữ liệu và giao tiếp giữa chúng.

Ý nghĩa:

Giúp đội ngũ phát triển, vận hành và quản lý hiểu rõ **phạm vi và vai trò của Quickship** trong tổng thể chuỗi dịch vụ giao hàng Viettel Post.

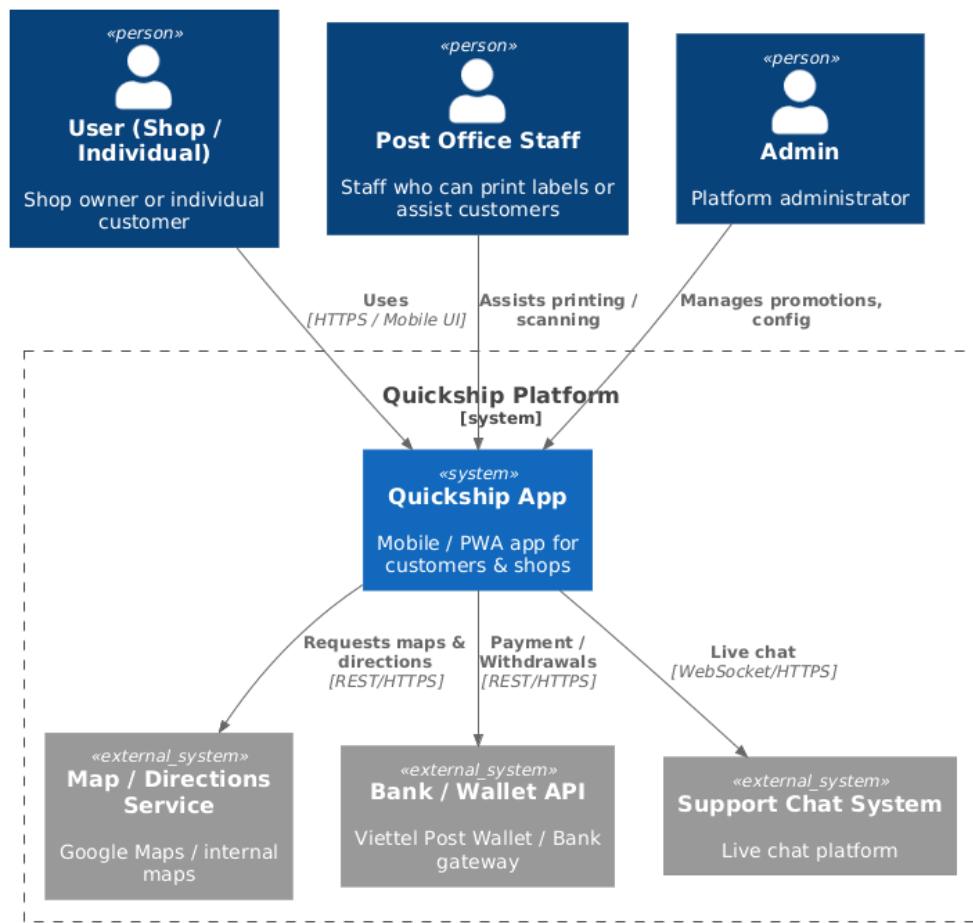
2. Actors (Người dùng)

Actor	Vai trò	Tương tác với hệ thống
Sender (Shop / Individual)	Người gửi hàng, chủ shop online	Tạo đơn, in mã vận đơn, thanh toán, theo dõi trạng thái đơn, rút tiền COD
Receiver (Customer)	Người nhận hàng	Theo dõi đơn, xác nhận nhận hàng, phản hồi
Post Office Staff	Nhân viên bưu cục Viettel Post	Quét mã, in tem, xác nhận trạng thái đơn
Admin	Quản trị hệ thống	Quản lý người dùng, khuyến mãi, cấu hình hệ thống, theo dõi báo cáo hoạt động

3. External Systems (Hệ thống tích hợp bên ngoài)

External System	Mô tả	Protocol / API
Viettel Post Wallet / Bank API	Xử lý thanh toán, nạp tiền, rút tiền COD	HTTPS / REST API
Google Maps API	Định vị, tìm bưu cục gần nhất, hiển thị đường đi	HTTPS / REST API
Firebase Cloud Messaging (FCM)	Gửi thông báo đơn hàng, khuyến mãi, cảnh báo	HTTPS / REST API
SMTP Email Server	Gửi email xác nhận đơn, biên lai thanh toán, báo cáo	SMTP

4. Mermaid Diagram



5. Tóm tắt quan hệ

5.1. Actors → System

Actor	Action	Description
Sender (Shop / Individual)	Create & Manage Orders	Tạo đơn, in mã vận đơn, theo dõi, rút tiền COD
Receiver (Customer)	Track Orders	Tra cứu đơn, xác nhận giao hàng
Post Office Staff	Process & Verify	Quét mã, xác nhận trạng thái, hỗ trợ người gửi
Admin	Manage & Monitor	Quản trị hệ thống, thống kê, khuyến mãi, cấu hình

5.2. System → External System

Target	Protocol	Purpose
Quickship Wallet / Bank API	HTTPS / REST	Thanh toán COD, rút tiền về ví hoặc tài khoản ngân hàng
Google Maps API	HTTPS / REST	Tìm kiếm, định vị bưu cục gần nhất, hiển thị đường đi
Firebase Cloud Messaging (FCM)	HTTPS / REST	Gửi thông báo đơn hàng, khuyến mãi, trạng thái giao
SMTP Email Server	SMTP	Gửi email xác nhận, thông báo giao dịch

6. Tóm tắt

Mục tiêu chính	Mô tả
Phạm vi hệ thống	Quickship là nền tảng hỗ trợ gửi hàng nhanh, quản lý đơn, rút tiền COD, tìm bưu cục, và theo dõi giao hàng.

Mục tiêu chính	Mô tả
Actors chính	Người gửi, người nhận, nhân viên bưu cục, quản trị viên.
Tích hợp ngoài	Ví Viettel Post, bản đồ Google Maps, Firebase, SMTP Email.
Kết quả	Cung cấp bức tranh tổng thể về hệ thống Quickship và cách nó kết nối với môi trường xung quanh.

C4 Diagram Level 2 – Container Diagram

1. Tổng quan

- Ứng dụng người dùng (Web + Mobile)
- API Server (Backend monolith)
- Database chung
- Các dịch vụ bên ngoài (Viettel Post Wallet, Google Maps API, Firebase, SMTP)

Toàn bộ logic nghiệp vụ (Orders, COD, Location, Promotions, Auth, Notification) được tổ chức dưới dạng **module nội bộ** trong cùng một codebase.

2. Danh sách Containers

2.1. Frontend Application

Container	Công nghệ	Mô tả	Port
Web App (PWA)	React.js, TypeScript, TailwindCSS	Ứng dụng web/PWA cho người gửi, người nhận, nhân viên bưu cục	5173
Mobile App	React Native / Expo	Ứng dụng di động tích hợp API chung, hỗ trợ quét mã, thông báo realtime	-

2.2. Backend Monolith API

Container	Công nghệ	Mô tả	Port
Quickship API Server	Node.js, Express / NestJS, TypeORM	Ứng dụng backend monolith – xử lý toàn bộ nghiệp vụ: Auth, Orders, COD, Location, Promotion, Notification	3000

Các module chính trong Monolith:

Module	Chức năng chính
Auth Module	Đăng nhập, đăng ký, phân quyền, quản lý JWT
Order Module	Tạo và tra cứu đơn hàng, in mã vận đơn
COD Module	Theo dõi và rút tiền COD, quản lý lịch sử giao dịch
Location Module	Tìm kiếm bưu cục, định vị, hiển thị bản đồ
Promotion Module	Áp dụng khuyến mãi, quản lý mã giảm giá
Notification Module	Gửi thông báo qua FCM, email, hiển thị lịch sử gửi

2.3. Database

Container	Công nghệ	Mô tả
Quickship DB	PostgreSQL / MySQL	CSDL duy nhất, chứa tất cả bảng: users, orders, cod_transactions, promotions, offices, notifications
Redis (tùy chọn)	Redis Cache	Lưu phiên đăng nhập, token, cache vị trí tạm thời

2.4. External Systems

Container	Công nghệ	Mô tả
Viettel Post Wallet / Bank API	REST API	Xử lý thanh toán COD, rút tiền, truy vấn ví
Google Maps API	REST API	Định vị bưu cục, tính khoảng cách, hiển thị bản đồ
Firebase Cloud Messaging (FCM)	REST API	Gửi push notification đến người dùng
SMTP Email Server	SMTP	Gửi email xác nhận đơn, rút tiền, thông báo khuyến mãi

3. Giao tiếp giữa các Containers

3.1. Protocols

Giao thức Sử dụng cho

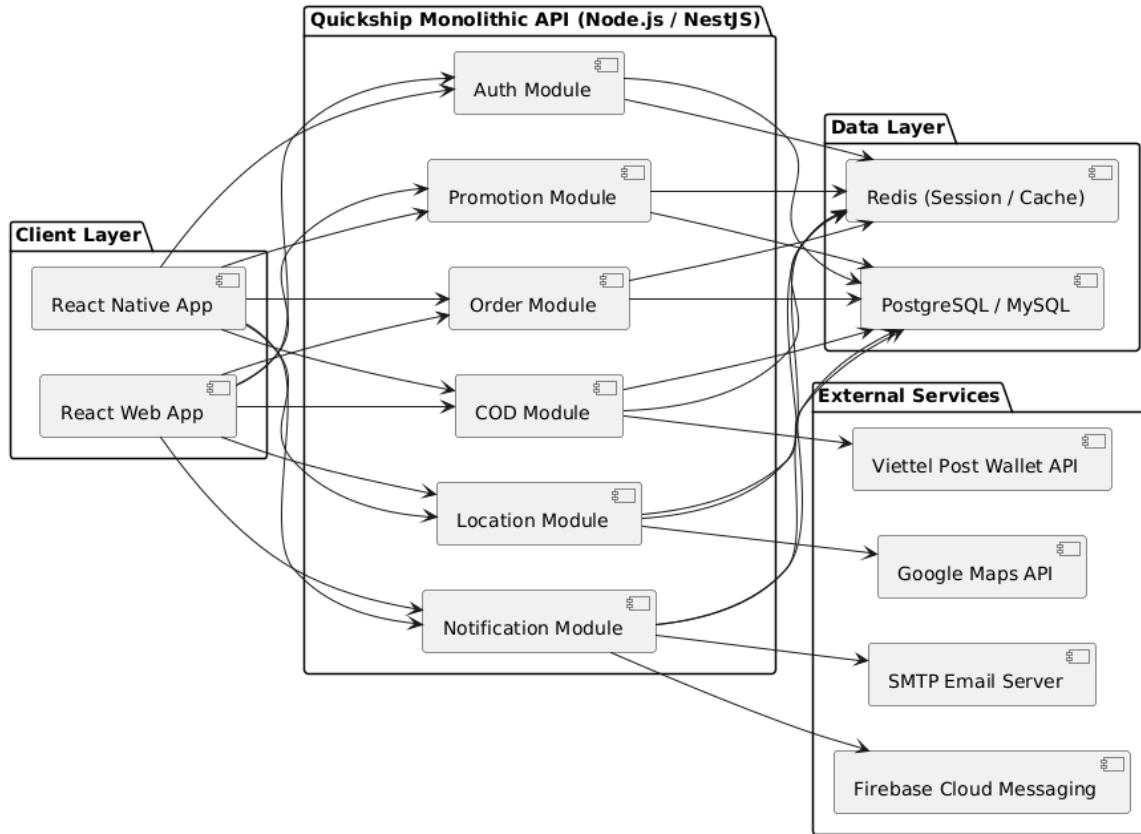
HTTPS/REST Web/Mobile App ↔ Quickship API

TCP API ↔ Database

HTTPS/REST API ↔ External Services

SMTP API ↔ Email Server

3.2. Sơ đồ giao tiếp



4. Communication Flow

Source	Target	Protocol	Mục đích
Web/Mobile App	Quickship API	HTTPS/REST	Gửi request người dùng
Quickship API	Database	TCP	Lưu và truy vấn dữ liệu
Quickship API	Wallet API	HTTPS/REST	Rút tiền, kiểm tra ví
Quickship API	Google Maps API	HTTPS/REST	Tra cứu bưu cục gần nhất
Quickship API	FCM	HTTPS/REST	Gửi thông báo realtime
Quickship API	SMTP	SMTP	Gửi email xác nhận

C4 Diagram Level 3 – Component Diagram

Tổng quan cấu trúc

Layer	Vai trò	Thành phần tiêu biểu
Presentation Layer	Tiếp nhận request, trả response	Controllers (NestJS), DTOs
Business Layer	Xử lý logic nghiệp vụ, validate, gọi APIs ngoài	Services (OrderService, CODService, LocationService, PromotionService, NotificationService)
Data Access Layer	Truy cập dữ liệu trong DB	Repositories (TypeORM), Entities
External Integration Layer	Kết nối dịch vụ bên ngoài	API Clients (WalletAPI, MapsAPI, FCM, SMTP)
Database Layer	Lưu trữ dữ liệu	PostgreSQL

Các Components chính

Presentation Layer

Component	Mô tả
OrderController	API /orders: tạo, xem, huỷ đơn hàng
CODController	API /cod: truy vấn số dư, rút tiền

Component	Mô tả
LocationController	API /locations: tìm bưu cục, định vị
PromotionController	API /promotions: mã giảm giá, ưu đãi
NotificationController	API /notifications: gửi thông báo
AuthController	API /auth: đăng nhập, đăng ký

Business Layer

Component	Vai trò chính
OrderService	Quản lý đơn hàng (create, confirm, update, cancel)
CODService	Xử lý tiền thu hộ, tính phí COD, rút tiền về ví
PromotionService	Xác thực mã khuyến mãi, tính giảm giá
NotificationService	Gửi thông báo qua FCM hoặc Email
LocationService	Tra cứu bưu cục gần nhất, định vị vị trí
AuthService	Xác thực JWT, quản lý người dùng
QuickOrderService	Tạo đơn nhanh từ text/voice input (ASR)

Data Layer

Component	Mô tả
OrderRepository	CRUD bảng orders
CODRepository	CRUD bảng cod_transactions
UserRepository	CRUD bảng users
PromotionRepository	CRUD bảng promotions
NotificationRepository	CRUD bảng notifications
LocationRepository	CRUD bảng post_offices
 BaseEntity	Class cha kế thừa của tất cả entities
TypeORM	ORM kết nối PostgreSQL

External Integration Layer

Component	API	Chức năng
WalletAPI	Viettel Post Wallet	Rút / chuyển tiền COD
MapsAPI	Google Maps API	Tính khoảng cách, tìm bưu cục
FCMClient	Firebase Cloud Messaging	Gửi push notification
SMTPClient	Email server	Gửi email xác nhận đơn
ASRClient (tùy chọn)	Speech API	Nhập đơn hàng bằng giọng nói

Database Layer (PostgreSQL)

Bảng	Mô tả
users	Thông tin người dùng
orders	Đơn hàng
cod_transactions	Lịch sử thu hộ COD
promotions	Mã khuyến mãi
notifications	Lịch sử thông báo
post_offices	Danh sách bưu cục
quick_orders	Đơn hàng tạo nhanh
sessions	Phiên đăng nhập

3 Luồng xử lý ví dụ: Tạo đơn hàng

(Presentation → Business → Data → API → DB)

1. OrderController nhận POST /orders

2. Gọi OrderService.createOrder()

3. OrderService gọi:

- PromotionService.validateCode()
- CODService.calculateCOD()
- OrderRepository.save()

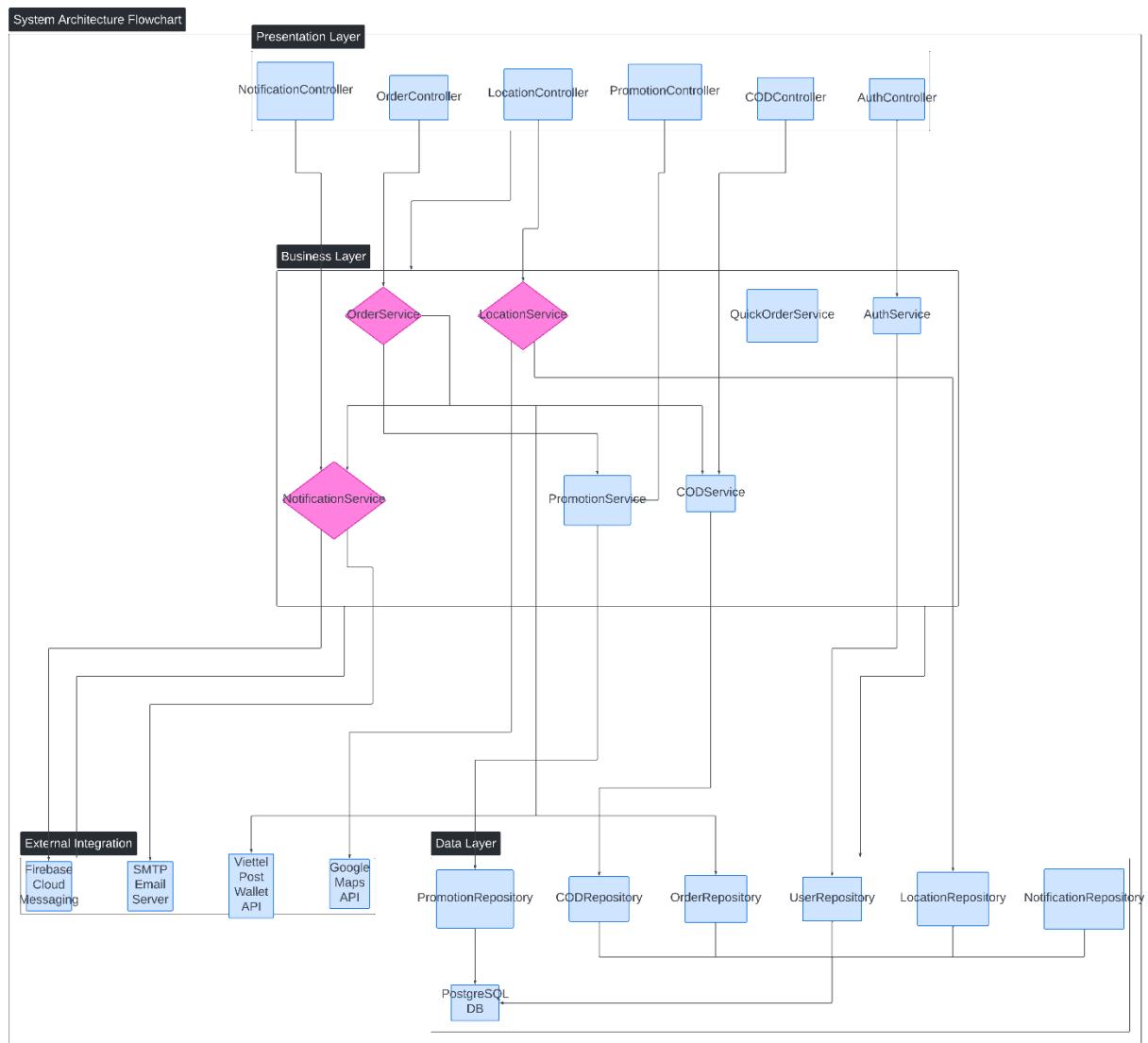
- **NotificationService.sendCreated()**

4. OrderRepository ghi dữ liệu vào PostgreSQL

5. NotificationService gọi FCMClient gửi thông báo

6. Trả về response 201 Created

4 Mermaid Diagram – Component View



5 Tóm tắt mối quan hệ

Nguồn	Đích	Mục đích
OrderController → OrderService	Gọi logic tạo đơn hàng	
OrderService → PromotionService	Kiểm tra khuyến mãi	
OrderService → CODService	Tính tiền thu hộ	
OrderService → NotificationService	Gửi thông báo đơn mới	
NotificationService → FCMClient	Gửi push	
LocationService → MapsAPI	Lấy vị trí	
CODService → WalletAPI	Rút tiền COD	
Repositories → DB	CRUD dữ liệu	

6 Lợi ích của mô hình Layered Monolith

- Dễ mở rộng thành Microservice trong tương lai
- Dễ test từng layer (Unit test Service, Integration test Repository)
- Dễ bảo trì: phân tách rõ Presentation – Logic – Data
- Tương thích hoàn toàn với NestJS Modular Architecture

7 Tóm lược

Level	Mục tiêu	Trình bày
Level 1	System Context	Toàn cảnh Quickship + Users + External APIs
Level 2	Containers	Web App, API Server (Monolith), DB
Level 3	Components	Controllers, Services, Repositories, APIs, DB
Level 4	Code (Class / Function)	Chi tiết logic, hàm, flow nội bộ

C4 Diagram Level 4 – Code Quickship

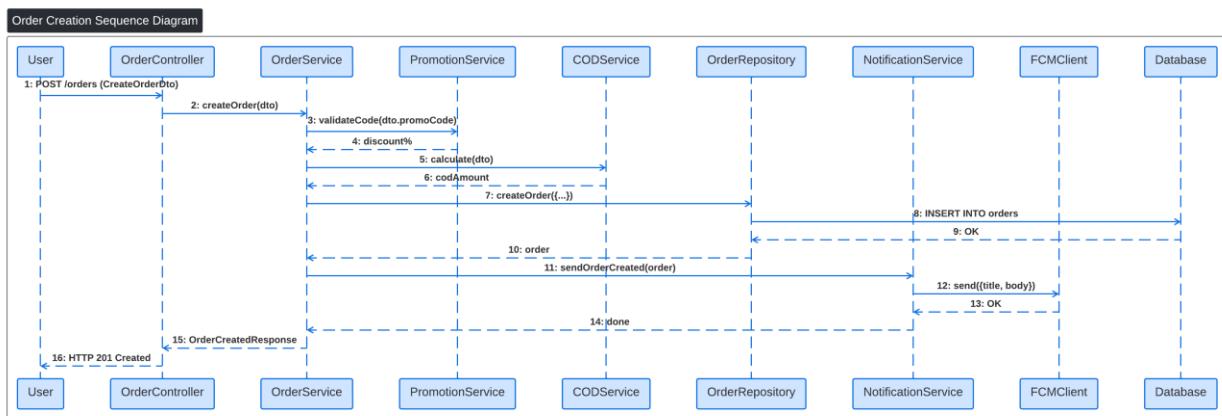
🎯 Mục tiêu

Hiển thị mối quan hệ giữa các hàm trong từng layer và cách dữ liệu di chuyển từ request người dùng → xử lý logic → ghi DB → gọi API ngoài → phản hồi client.

1 Module: Order (Tạo đơn hàng)

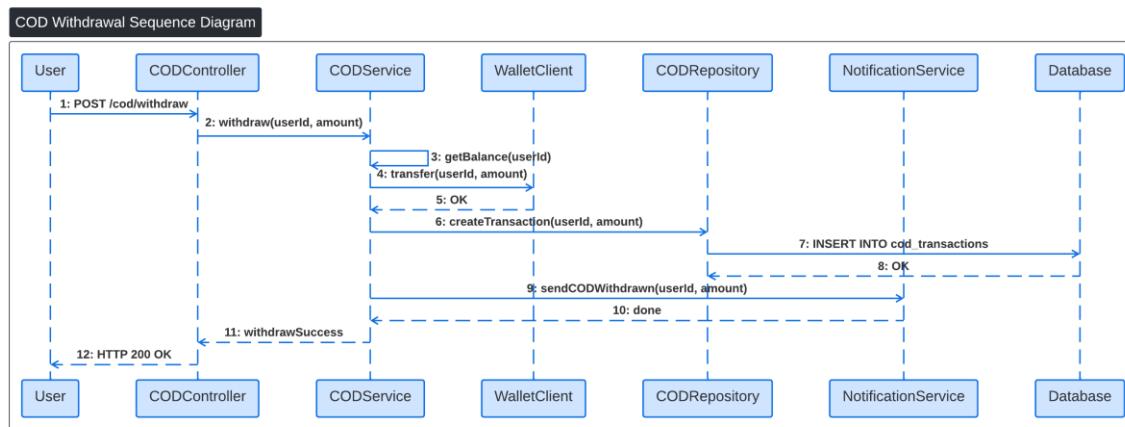
Bước	Thành phần	Hàm	Chức năng
1	OrderController	createOrder(dto)	Nhận request POST /orders, gọi service
2	OrderService	validate(dto)	Kiểm tra dữ liệu đầu vào
3	PromotionService	validateCode(code)	Xác thực mã giảm giá
4	CODService	calculate(dto)	Tính tiền COD thu hộ
5	OrderRepository	createOrder(data)	Lưu đơn hàng mới vào DB

Bước Thành phần	Hàm	Chức năng
6	NotificationService sendOrderCreated(order)	Gửi thông báo
7	FCMClient send(msg)	Gửi push notification
8	OrderController return Response	Trả JSON kết quả cho client



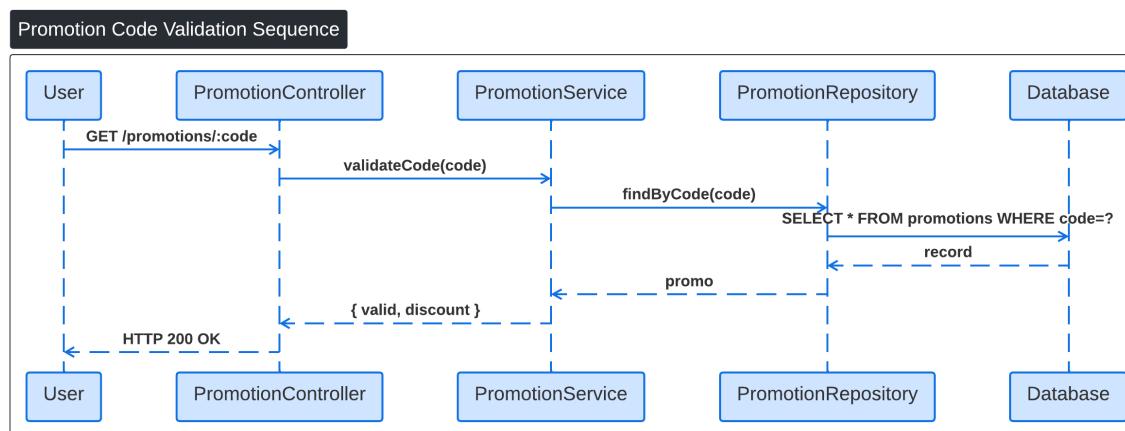
2 Module: COD (Quản lý tiền thu hộ)

Bước Thành phần	Hàm	Chức năng
1	CODController withdraw()	Nhận yêu cầu rút tiền
2	CODService getBalance(userId)	Lấy số dư khả dụng
3	CODService withdraw(userId, amount)	Thực hiện rút tiền
4	WalletClient transfer(userId, amount)	Gọi Viettel Wallet API
5	CODRepository createTransaction()	Lưu lịch sử rút tiền
6	NotificationService sendCODWithdrawn()	Gửi thông báo
7	OrderController return Response	Trả phản hồi thành công



3 Module: Promotion (Xác thực khuyến mãi)

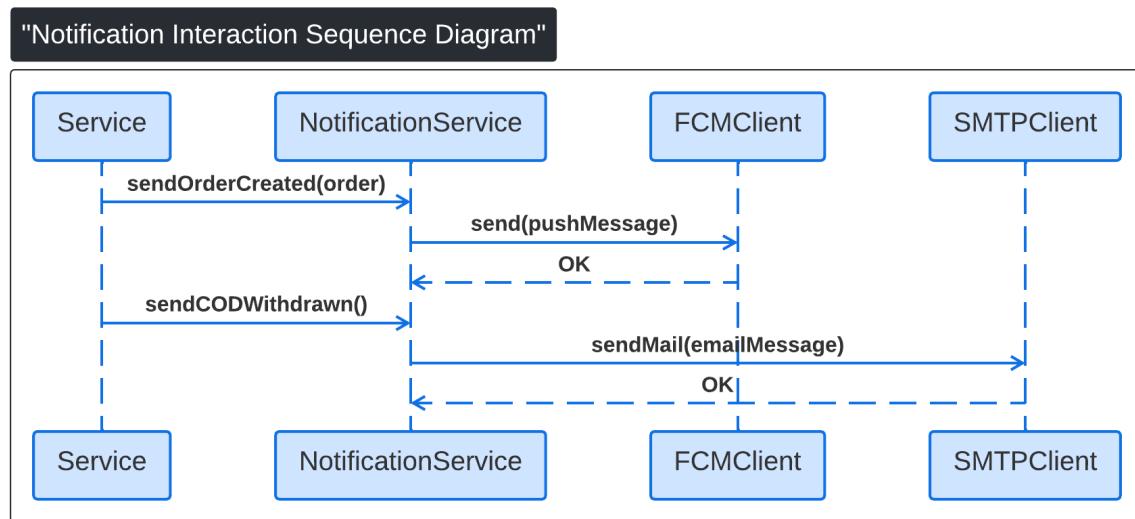
Thành phần	Hàm	Vai trò
PromotionController	checkCode(code)	Nhận request xác minh mã
PromotionService	validateCode(code)	Kiểm tra hiệu lực
PromotionRepository	findByCode(code)	Tìm trong DB
Database		Lưu bảng promotions



4 Module: Notification (Thông báo)

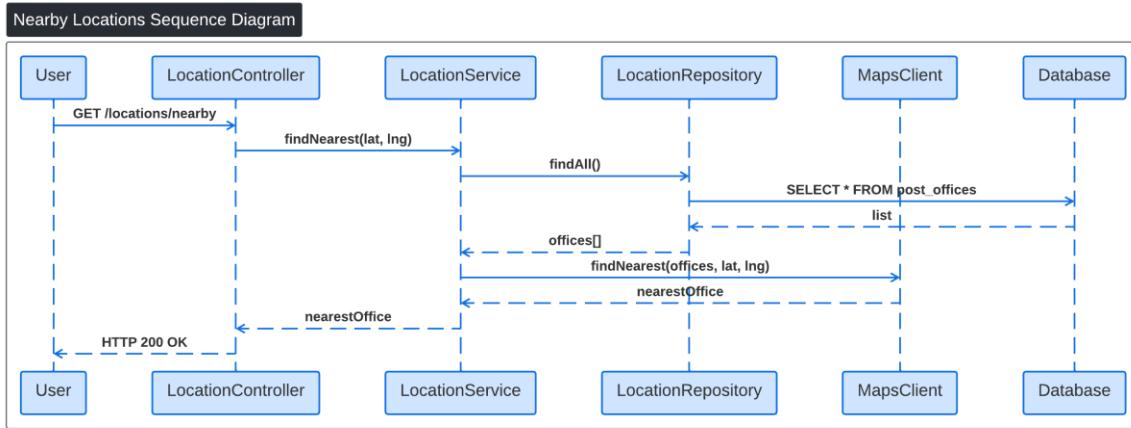
Thành phần	Hàm	Vai trò
NotificationService	sendOrderCreated()	Gửi thông báo đơn hàng mới

Thành phần	Hàm	Vai trò
	NotificationService sendCODWithdrawn()	Gửi thông báo rút tiền
FCMClient	send()	Gửi push
SMTPClient	sendMail()	Gửi email



5 Module: Location (Tìm bưu cục gần nhất)

Thành phần	Hàm	Vai trò
LocationController	findNearby()	Nhận yêu cầu người dùng
LocationService	findNearest(lat, lng)	Xử lý logic tìm vị trí
LocationRepository	findAll()	Lấy danh sách bưu cục
MapsClient	findNearest()	Gọi API Google Maps
Database		Bảng post_offices



6 Module: Auth (Đăng nhập / Xác thực người dùng)

Thành phần	Hàm	Vai trò
AuthController	login()	Nhận yêu cầu đăng nhập
AuthService	validateUser()	Kiểm tra thông tin đăng nhập
UserRepository	findByEmail()	Truy xuất user
JWTUtil	signToken()	Sinh JWT token
Database		Bảng users

