

DISTRIBUTED BOOKING HEALTHCARE USING MICROSERVICES

Tiểu luận môn: Hệ thống phân tán.



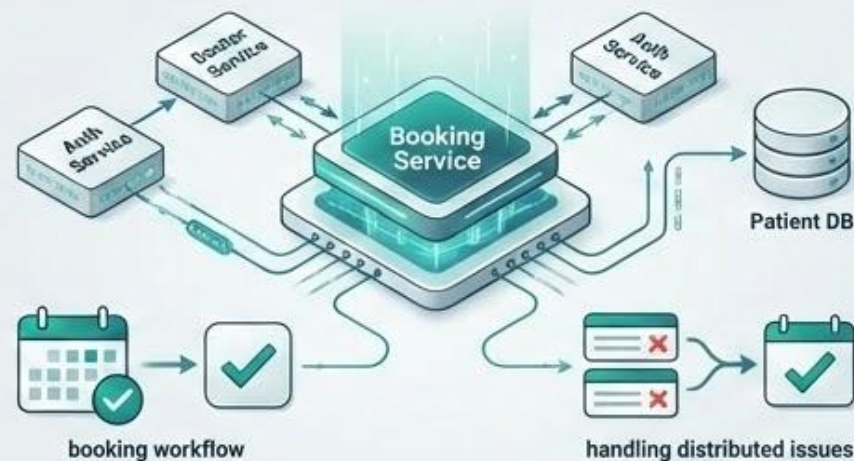
NHÓM 14: Bùi Thị Phương Thảo • Phan Thị Nguyệt • Dương Thị Thu Phương
Giảng viên hướng dẫn: TS. Kim Ngọc Bách • PTIT 2025

LÝ DO CHỌN ĐỀ TÀI & MỤC TIÊU



Lý Do Chọn Đề Tài

Thách thức quá tải tại các bệnh viện và quy trình đặt lịch thủ công kém hiệu quả yêu cầu một giải pháp phân tán, kết nối đa thực thể: bệnh nhân, bác sĩ, cơ sở y tế.



Mục Tiêu Đề Tài

Phân tích kiến trúc Microservices, làm rõ các luồng nghiệp vụ đặt lịch, và xử lý các vấn đề phân tán điển hình như trùng lịch và đồng bộ dữ liệu thực tế.

CÁC NHÓM NGƯỜI DÙNG HỆ THỐNG



Bệnh Nhân

Tìm kiếm bác sĩ, chuyên khoa. Đặt lịch khám và quản lý hồ sơ sức khỏe cá nhân.



Bác Sĩ

Thiết lập lịch trình làm việc, quản lý danh sách bệnh nhân và gửi kết quả khám bệnh.



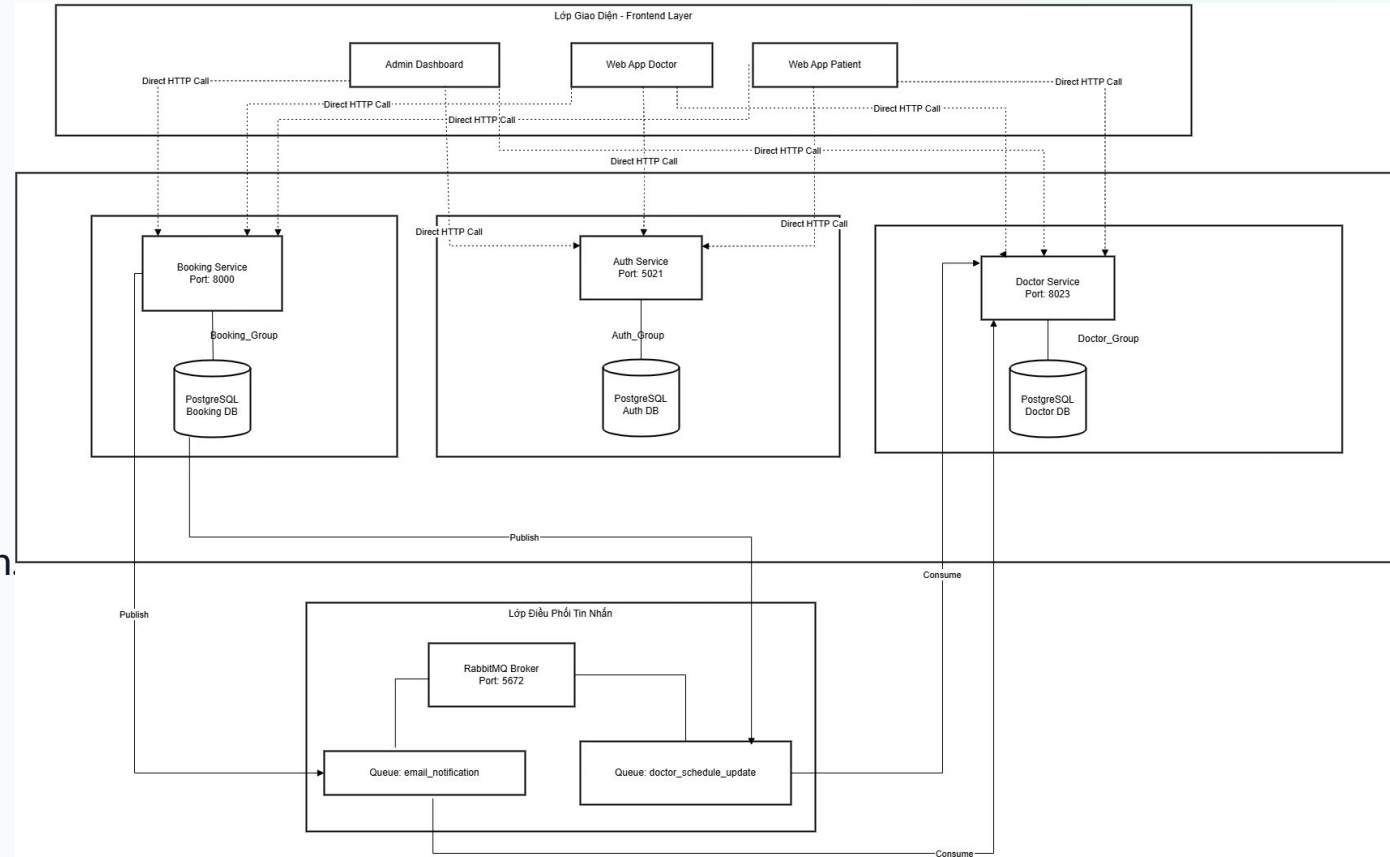
Quản Trị Viên

Quản lý danh mục chuyên khoa, phòng khám, bài viết cẩm nang và điều phối hệ thống.

MÔ HÌNH KIẾN TRÚC TỔNG QUÁT

Hệ thống áp dụng kiến trúc **Microservice** với các đặc điểm phân tán:

- Triển khai độc lập cho từng dịch vụ nghiệp vụ.
- Giao tiếp linh hoạt qua REST API và Message Queue (RabbitMQ).
- Cơ chế **Eventual Consistency** đảm bảo tính nhất quán.
- Khả năng chịu lỗi (Tolerant to Failure) vượt trội.



CÁC MICROSERVICES CHÍNH



Auth Service

Xác thực người dùng (JWT), phân quyền vai trò Doctor/Admin/Staff và bảo mật hệ thống.



Doctor Service

Quản lý hồ sơ bác sĩ, thông tin cơ sở y tế và danh mục chuyên khoa đa dạng.



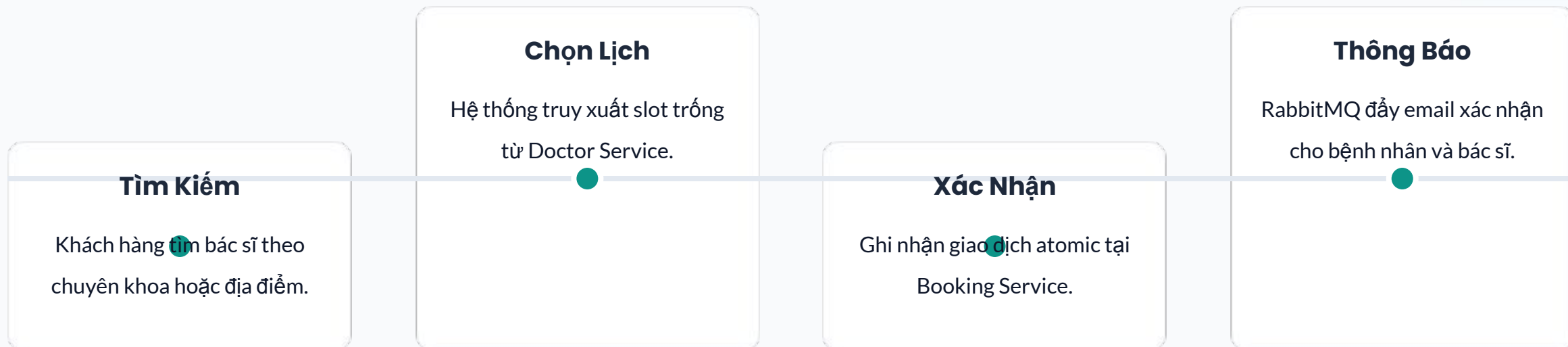
Booking Service

Xử lý logic đặt lịch, kiểm tra tính khả dụng của slot (Atomic booking) và điều phối lịch hẹn.

THIẾT KẾ API ENDPOINTS TIÊU BIỂU

Dịch vụ	Method	Endpoint	Mô tả chức năng
Auth	POST	/api/v1/auth/login	Đăng nhập và nhận JWT Token
Doctor	GET	/api/v1/doctors/{id}	Xem chi tiết hồ sơ bác sĩ chuyên khoa
Doctor	GET	/api/v1/doctors/{id}/schedules	Lấy lịch khám còn trống thực tế
Booking	POST	/api/v1/bookings	Khởi tạo đặt lịch và gửi thông báo

LUỒNG NGHIỆP VỤ ĐẶT LỊCH KHÁM

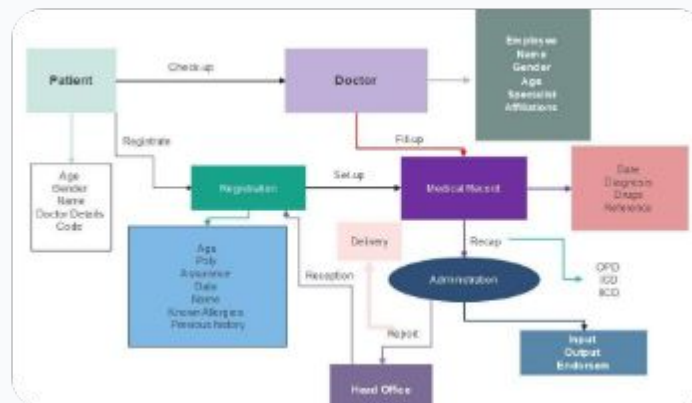


DATABASE PER SERVICE PATTERN



Auth DB

Lưu trữ tài khoản, mật khẩu băm và quyền hạn.



Doctor DB

Dữ liệu bác sĩ, phòng khám và lịch trình chi tiết.



Booking DB

Lịch sử lịch hẹn và trạng thái đặt chỗ thời gian thực.

QUẢN LÝ LỖI & TÍNH NHẤT QUÁN

Circuit Breaker

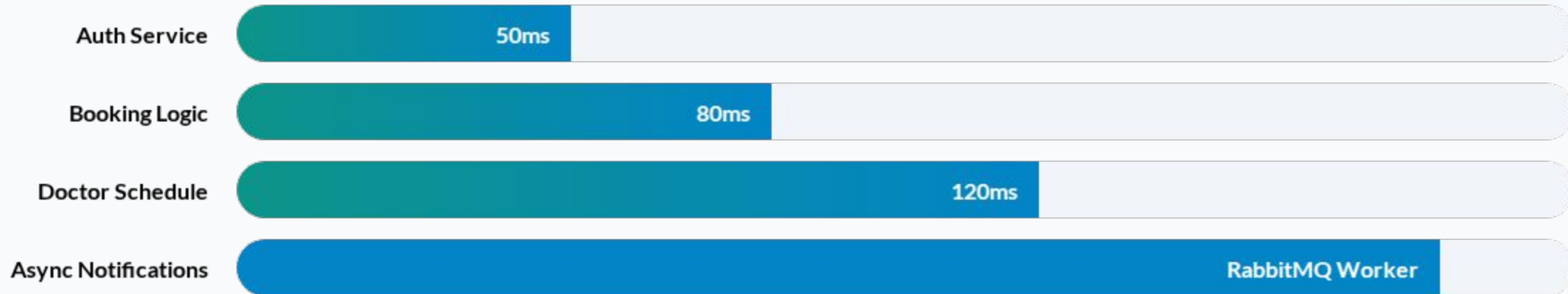
Ngắt kết nối tạm thời đến dịch vụ gặp sự cố (ví dụ: Notification) để đảm bảo luồng đặt lịch chính không bị gián đoạn.

Saga Pattern

Quản lý các giao dịch phức tạp đi qua nhiều dịch vụ, đảm bảo tính nhất quán dữ liệu thông qua cơ chế bù trừ (compensation).

Hệ thống đảm bảo tính "Tolerant to failure" trong mọi tình huống vận hành.

MÔ PHỎNG HIỆU NĂNG PHÂN TÁN



Xử lý bất đồng bộ giúp giảm tải đáng kể cho hệ thống chính và cải thiện trải nghiệm người dùng.

TRIỂN KHAI & VẬN HÀNH

Frontend Layer: ReactJS, Redux, SCSS, Bootstrap.

Service Layer: Node.js, Express Framework.

Message Queue: RabbitMQ 3.x (Port 5672/15672).

Containerization: Dockerized services kết nối mạng nội bộ an toàn.

Scale: Dễ dàng scale-out từng service theo nhu cầu thực tế.

Questions?

Trân trọng cảm ơn Thầy và các bạn đã chú ý lắng nghe!

Hệ Thống Booking Healthcare Phân Tán • Microservices Architecture • 2025