**UỶ BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

A blue circle with text

AI-generated content may be incorrect.

BÁO CÁO BÀI TẬP HỌC PHẦN

CÁC CÔNG NGHỆ LẬP TRÌNH HIỆN ĐẠI

Sinh viên thực hiện:

Đoàn Minh Khôi – DCT121C3 – 3121411106

Trần Minh Chiến – DCT121C3 -3121411028

Nguyễn Quang Hiếu – DCT121C3- 31214111..

Giảng viên phụ trách: **TS. Đỗ Như Tài**

**TP. Hồ Chí Minh, tháng 03 năm 2025**

Contents

[TÀI LIỆU THIẾT KẾ HỆ THỐNG QUẢN LÝ NHÀ SÁCH 2](#_Toc196000176)

[1. Giới thiệu chung . 2](#_Toc196000177)

[1.1. Mục đích của tài liệu. 2](#_Toc196000178)

[1.2. Phạm vi hệ thống. 2](#_Toc196000179)

[1.3. Đối tượng sử dụng tài liệu. 3](#_Toc196000180)

[1.4. User Stories cho hệ thống quản lý nhà sách. 3](#_Toc196000181)

[1.5. Các công nghệ dự kiến. 3](#_Toc196000182)

[1.6. Kiến trúc chi tiết cho hệ thống quản lý nhà sách. 6](#_Toc196000183)

[1.7. Cấu trúc dự án và thư mục dựu kiến. 7](#_Toc196000184)

[1.8. Kiến trúc chi tiết mới nếu tích hợp Resilience, Observability và Monitoring. 8](#_Toc196000185)

[2. Phân tích thiết kế hệ thống. 10](#_Toc196000186)

[2.1. Yêu cầu chức năng. 10](#_Toc196000187)

[**2.2.** Focus Use-case. 10](#_Toc196000188)

# TÀI LIỆU THIẾT KẾ HỆ THỐNG QUẢN LÝ NHÀ SÁCH

## Giới thiệu chung .

### Mục đích của tài liệu.

Tài liệu này mô tả toàn bộ hệ thống quản lý nhà sách từ góc độ yêu cầu, kiến trúc, thiết kế hệ thống cho đến công nghệ triển khai. Mục tiêu là xây dựng một nền tảng quản lý hiện đại, hỗ trợ việc kinh doanh sách cả ở mức người dùng cuối (khách hàng) và người quản trị hệ thống, đồng thời đảm bảo khả năng mở rộng, bảo mật và hiệu suất cao. Tài liệu cũng đóng vai trò làm cơ sở cho việc phát triển phần mềm, kiểm thử, triển khai và vận hành.

### Phạm vi hệ thống.

Hệ thống quản lý nhà sách hướng tới việc số hóa toàn bộ quy trình mua bán và quản lý sách, bao gồm:

* Cho phép người dùng tìm kiếm, xem chi tiết sách, đặt hàng và thanh toán trực tuyến.
* Hỗ trợ quản lý kho sách, đơn hàng, khách hàng, đánh giá, khuyến mãi và báo cáo doanh thu.
* Quản trị viên có thể theo dõi, cập nhật thông tin và thực hiện các tác vụ vận hành một cách hiệu quả.

Hệ thống được xây dựng ban đầu theo mô hình monolithic để tăng tốc phát triển ban đầu, sau đó chuyển dần sang kiến trúc microservices nhằm hỗ trợ mở rộng linh hoạt theo nhu cầu thực tế.

### Đối tượng sử dụng tài liệu.

* Nhóm phát triển phần mềm: để hiểu rõ yêu cầu, kiến trúc và thiết kế hệ thống.
* Nhóm kiểm thử: để xây dựng các ca kiểm thử phù hợp với chức năng của hệ thống.
* Nhóm triển khai và vận hành: để nắm bắt kiến trúc, công nghệ và quy trình triển khai.
* Các bên liên quan khác như quản lý dự án, khách hàng hoặc người dùng kỹ thuật.

### User Stories cho hệ thống quản lý nhà sách.

**Khách hàng:**

* Tôi muốn tìm kiếm sách theo tên, thể loại, tác giả để dễ dàng tìm được sách mình quan tâm.
* Tôi muốn xem thông tin chi tiết về sách (giá cả, tình trạng còn hàng, nội dung tóm tắt).
* Tôi muốn thêm sách vào giỏ hàng và thanh toán.
* Tôi muốn theo dõi trạng thái đơn hàng của mình.
* Tôi muốn đánh giá và viết nhận xét về sách đã mua.

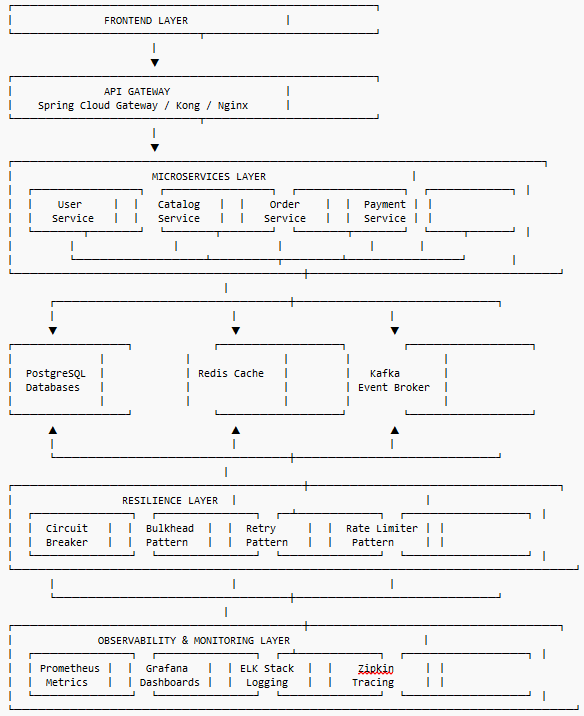
**Quản lý:**

* Tôi muốn thêm/sửa/xóa thông tin sách trong kho.
* Tôi muốn quản lý đơn hàng và cập nhật trạng thái giao hàng.
* Tôi muốn xem báo cáo doanh thu theo ngày/tháng/năm.
* Tôi muốn quản lý thông tin khách hàng.
* Tôi muốn theo dõi hàng tồn kho và nhận cảnh báo khi sắp hết hàng.

### Các công nghệ dự kiến.

* + 1. Kiến trúc Microservice.
* Service Discovery & Configuration:
  + Spring Cloud Netflix Eureka hoặc Kubernetes native service discovery.
  + Spring Cloud Config Server hoặc Kubernetes ConfigMaps & Secrets.
* API Gateway:
  + Spring Cloud Gateway - Tích hợp tốt với hệ sinh thái Spring.
  + Thay thế: Kong hoặc Nginx Ingress Controller (cho Kubernetes).
* Circuit Breaker & Resilience:
  + Resilience4j (thay thế cho Hystrix đã không còn được phát triển).
    1. Giao tiếp giữa các Service.
* Đồng bộ:
  + Spring Cloud OpenFeign - Client HTTP khai báo cho gọi API giữa các services.
  + Spring WebFlux - Cho các API reactive.
* Bất đồng bộ (Event Driven):
  + Apache Kafka - Cho các luồng dữ liệu lớn và phức tạp.
  + RabbitMQ - Cho messaging thông thường, đơn giản hơn Kafka.
    1. Containerization & Orchestration.
* Docker:
  + Docker Compose cho môi trường development.
  + Dockerfile tối ưu với multi-stage builds.
* Kubernetes:
  + Helm - Quản lý Kubernetes packages.
  + Kustomize - Quản lý YAML manifests.
  + Ingress Controller - NGINX hoặc Traefik.
    1. CI/CD & DevOps.
* CI/CD Pipeline:
  + GitHub Actions - Tích hợp tốt với GitHub repositories.
  + Jenkins hoặc GitLab CI/CD - Cho các pipeline phức tạp hơn.
* Container Registry:
  + Docker Hub hoặc GitHub Container Registry (ghcr.io).
  + Harbor cho private registry trong doanh nghiệp.
    1. Monitoring & Logging.
* Monitoring:
  + Prometheus - Thu thập metrics.
  + Grafana – Visualization.
  + Spring Boot Actuator - Endpoints cho health checks và metrics.
* Logging:
  + ELK Stack (Elasticsearch, Logstash, Kibana).
  + Fluentd - Thu thập logs từ Kubernetes pods.
  + Loki + Grafana - Giải pháp logging nhẹ hơn.
    1. Database & Caching.
* Database Access:
  + Spring Data JPA - ORM cho PostgreSQL.
  + Flyway hoặc Liquibase - Database migrations.
* Caching:
  + Redis - In-memory cache và session storage.
  + Spring Cache - Abstraction layer cho caching.
    1. Bảo mật.
* Authentication & Authorization:
  + Spring Security + OAuth2 / JWT.
  + Keycloak - Identity và Access Management.
* API Security:
  + Spring Security configs.
  + API Key  cho xác thực API giữa các services.

### Kiến trúc chi tiết cho hệ thống quản lý nhà sách.



### Cấu trúc dự án và thư mục dựu kiến.

A close-up of a form

AI-generated content may be incorrect.

### Kiến trúc chi tiết mới nếu tích hợp Resilience, Observability và Monitoring.

A diagram of a computer

AI-generated content may be incorrect.

## Phân tích thiết kế hệ thống.

### Yêu cầu chức năng.

* + 1. Quản lý sách.
* Thêm, sửa, xóa, tìm kiếm thông tin sách.
* Phân loại sách theo thể loại, tác giả.
* Quản lý số lượng tồn kho.
* Hiển thị thông tin chi tiết sách.
  + 1. Quản lý khách hàng.
* Đăng ký, đăng nhập tài khoản.
* Quản lý thông tin cá nhân.
* Xem lịch sử mua hàng.
* Đánh giá và bình luận sách.
  + 1. Quản lý đơn hàng.
* Tạo đơn hàng mới.
* Theo dõi trạng thái đơn hàng.
* Hủy đơn hàng.
* Xử lý thanh toán.
* Quản lý vận chuyển.
  + 1. Quản lý bán hàng.
* Thống kê doanh thu.
* Quản lý khuyến mãi, giảm giá.
* Quản lý đánh giá, phản hồi.
* Báo cáo bán hàng.

### Focus Use-case.