



🎯 Khóa học Thực chiến

# Triển khai dự án với Spring Boot trong 6 Sprint

Mô hình đào tạo mentor – Học qua làm dự án thật

Từ thiết kế ERD, xây dựng API RESTful chuẩn, áp dụng Spring Data JPA, bảo mật với Spring Security + JWT đến triển khai module hoàn chỉnh



# Tại vì sao nên chọn khóa học này?

Được thiết kế dành cho sinh viên, lập trình viên backend có đủ kỹ năng



## Lộ trình toàn diện

6 Sprint được thiết kế khoa học từ cơ bản đến nâng cao, giúp bạn nắm vững Spring Boot từ A-Z một cách có hệ thống và thực tế.



## Dự án thực tế

Xây dựng hệ thống "Quản lý học viên lớp học" hoàn chỉnh với ERD, API RESTful, JWT authentication và quality gate chuẩn công nghiệp.



## Chất lượng cao

Definition of Done rõ ràng, SonarQube quality gate, test coverage  $\geq 60\%$ , và code review 2 chiều đảm bảo chất lượng.

## Kỹ năng đạt được:

- ❖ Thiết kế ERD & Database Migration
- ❖ Spring Boot 3.x & RESTful API
- ❖ Spring Data JPA & MapStruct
- ❖ Spring Security 6 & JWT
- ❖ JUnit5, Mockito & SonarQube

# 6 SPRINT trở thành một BACKEND DEVELOPER

Mục tiêu: Tự thiết kế ERD, xây API RESTful chuẩn, áp dụng Spring Data JPA, bảo mật JWT

## Sprint 1

- Khởi động & nền tảng Web
- Tổng quan Web, CRUD RESTful API, Mô hình MVC

## Sprint 2

- Từ JDBC đến ORM & JPA
- JDBC, ORM, Spring Data JPA cơ bản

## Sprint 3

- JPA nâng cao & DTO
- Quan hệ nhiều bảng, DTO & MapStruct

## Sprint 4

- Chất lượng & ổn định
- Validate, Exception, Logger, JUnit, SonarQube

## Sprint 5

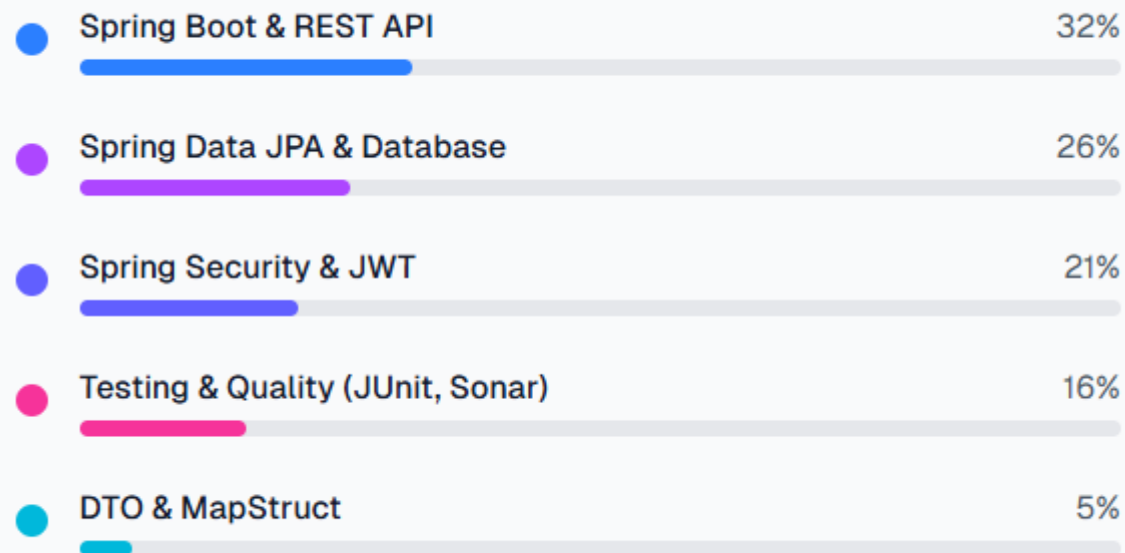
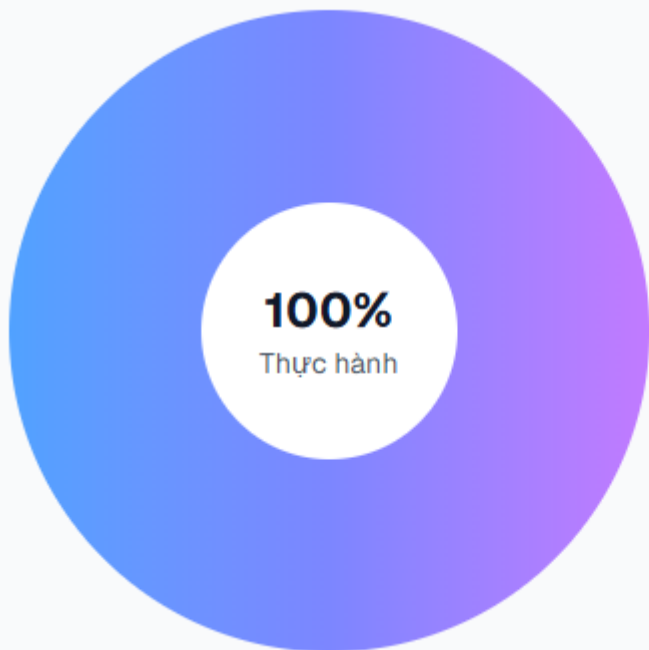
- Bảo mật thực chiến
- Spring Security, JWT, Refresh Token

## Sprint 6

- Case Study & hoàn thiện
- Hardening, tính năng nâng cao, triển khai

# KỸ NĂNG NHẬN ĐƯỢC SAU KHÓA HỌC

Phân bố kỹ năng theo từng Sprint và mức độ thành thạo



# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 1 Khởi động & nền tảng Web

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

### Buổi 1

#### Tổng quan Web & Project Setup

- ✓ Kiến trúc web: Client–Server, HTTP lifecycle, REST stateless.
- ✓ HTTP Protocol: methods, status codes, headers, body.
- ✓ Giới thiệu Spring Boot 3.x.Maven/Gradle: quản lý dependencies.
- ✓ Cấu trúc dự án chuẩn (Controller, Service, Repository, Model).
- ✓ Kết nối PostgreSQL (Docker/local).

#### Bài tập

- ❖ Tạo dự án Spring Boot 3.x và endpoint /health.
- ❖ Kết nối PostgreSQL và xác nhận kết nối thành công.

#### User Story

Là Admin, tôi muốn kiểm tra trạng thái API qua /health để biết hệ thống đang chạy.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 1 Khởi động & nền tảng Web

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

### Buổi 2

#### RESTful API Design

- ✓ REST API best practices: naming, versioning, status codes.
- ✓ JSON response format chuẩn hóa.
- ✓ Tích hợp Swagger/OpenAPI.
- ✓ Sử dụng Postman để test API.

#### Bài tập

- ❖ Thiết kế API `/api/v1/students` trả danh sách rỗng.
- ❖ Test API trên Postman + Swagger UI.

#### User Story

Là Admin, tôi muốn có API `/students` trả danh sách học viên để hiển thị trên hệ thống.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 1 Khởi động & nền tảng Web

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

Buổi 3

### Mô hình MVC

- ✓ MVC pattern: Controller, Service, Repository.
- ✓ Dependency Injection (Spring IoC).
- ✓ Tạo module "Student" CRUD (in-memory DB).
- ✓ Unit test cơ bản cho Controller.

### Bài tập

- ❖ Tạo CRUD cho Student (in-memory).
- ❖ Viết Unit test cho GET /students.

### User Story

Là Admin, tôi muốn tạo/sửa/xóa/xem học viên để quản lý dữ liệu học viên.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 1

### Khởi động & nền tảng Web

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên



#### Điều kiện Done:

- Dự án Spring Boot khởi tạo (Java 17+, Spring Boot 3.x).
- Maven/Gradle configuration hoàn chỉnh.
- PostgreSQL kết nối OK.
- API /health và /students chạy OK, trả JSON chuẩn.
- Swagger UI hiển thị API.



# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 2 Từ JDBC đến ORM & JPA cơ bản

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

### Buổi 4

#### JDBC Fundamentals

- ✓ JDBC API, DriverManager, Connection.
- ✓ PreparedStatement, ResultSet handling.
- ✓ Nhược điểm JDBC thuần.

#### Bài tập

- ❖ Kết nối PostgreSQL thật qua JDBC.
- ❖ Viết truy vấn SELECT học viên.

#### User Story

Là Developer, tôi muốn kết nối DB và truy vấn dữ liệu học viên bằng JDBC.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 2 Từ JDBC đến ORM & JPA cơ bản

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

### Buổi 5

#### ORM & Hibernate Intro

- ✓ ORM concepts, benefits.
- ✓ Entity mapping cơ bản.
- ✓ Hibernate session & transaction.
- ✓ Chuyển từ JDBC sang ORM.

#### Bài tập

- ❖ Tạo entity Student và lưu xuống DB bằng Hibernate.

#### User Story

Là Developer, tôi muốn map bảng DB thành entity Java để làm việc dễ dàng hơn.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 2 Từ JDBC đến ORM & JPA cơ bản

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

### Buổi 6

#### Spring Data JPA + Migration

- ✓ JPA Repository interface, CRUD methods.
- ✓ Query methods cơ bản.
- ✓ Flyway/Liquibase migration.

#### Bài tập

- ❖ Tạo bảng Student, Classroom, Teacher bằng migration.
- ❖ CRUD dữ liệu qua Spring Data JPA..

#### User Story

Là Admin, tôi muốn quản lý lớp học và giảng viên để tổ chức khoá học.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 2 Từ JDBC đến ORM & JPA cơ bản

Nắm kiến trúc web, luồng request/response, REST, MVC; dựng khung dự án Spring Boot với CRUD học viên

### Điều kiện DONE:

- DB thật kết nối OK (PostgreSQL/MySQL).
- Migration chạy thành công (Flyway/Liquibase).
- Entities Student, Classroom, Teacher tạo OK.
- CRUD hoạt động với DB thật.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 3 JPA nâng cao, quan hệ nhiều bảng & DTO

Mapping quan hệ N–N, 1–N, truy vấn tùy biến; tách DTO với MapStruct

Buổi 7

### Spring Data JPA + Migration

- ✓ Custom query methods.
- ✓ @Query annotation.
- ✓ Specification pattern.
- ✓ Pagination & sorting nâng cao..

### Bài tập

❖ Viết query tìm học viên theo tên/email.

### User Story

Là Admin, tôi muốn tìm kiếm học viên theo tên/email để lọc danh sách.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 3 JPA nâng cao, quan hệ nhiều bảng & DTO

Mapping quan hệ N–N, 1–N, truy vấn tùy biến; tách DTO với MapStruct

Buổi 8

### Entity Relationships & Performance

- ✓ One-to-Many, Many-to-Many mapping.
- ✓ Join operations, cascade.
- ✓ Lazy vs Eager loading.
- ✓ N+1 problem & cách xử lý.

### Bài tập

❖ Tạo quan hệ Student–Classroom, Student–Course.

### User Story

Là Giảng viên, tôi muốn xem danh sách học viên trong lớp tôi dạy.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 3 JPA nâng cao, quan hệ nhiều bảng & DTO

Mapping quan hệ N–N, 1–N, truy vấn tùy biến; tách DTO với MapStruct

Buổi 9

### DTO & MapStruct

- ✓ Lợi ích DTO pattern.
- ✓ MapStruct configuration.
- ✓ Entity  $\leftrightarrow$  DTO mapping.
- ✓ Response optimization.

### Bài tập

- ❖ Tách entity Student thành StudentDTO cho API trả về.

### User Story

Là Developer, tôi muốn API trả DTO thay vì entity để bảo mật dữ liệu.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 3 JPA nâng cao, quan hệ nhiều bảng & DTO

Mapping quan hệ N–N, 1–N, truy vấn tùy biến; tách DTO với MapStruct

### Điều kiện DONE:

- Bảng Enrollment (Student–Classroom) tạo OK.
- Thêm bảng Course, Score.
- Quan hệ N–N, 1–N hoạt động.
- API trả DTO, không lộ entity.



# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

Sprint 4

## Chất lượng & ổn định

Đầu vào chuẩn, lỗi có cấu trúc, log đúng chỗ; kiểm thử & quality gate

Buổi 10

### Validation & Exception Handling

- ✓ Bean Validation (JSR 380).
- ✓ Custom validator.
- ✓ Global Exception Handler.
- ✓ RFC7807 error format.

### Bài tập

❖ Validate email, tên khi tạo học viên.

### User Story

Là Admin, tôi muốn hệ thống báo lỗi rõ ràng khi nhập dữ liệu sai.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

Sprint 4

## Chất lượng & ổn định

Đầu vào chuẩn, lỗi có cấu trúc, log đúng chỗ; kiểm thử & quality gate

Buổi 11

### Logging & Testing

- ✓ Logback configuration.
- ✓ AOP Logging (Controller & Service).
- ✓ JUnit5 & Mockito basics.
- ✓ Unit test best practices.

### Bài tập

❖ Viết test cho StudentService.

### User Story

Là Developer, tôi muốn log request/response và test code để đảm bảo chất lượng.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

Sprint 4

## Chất lượng & ổn định

Đầu vào chuẩn, lỗi có cấu trúc, log đúng chỗ; kiểm thử & quality gate

Buổi 12

### Code Quality & Security Intro

- ✓ SonarLint & SonarQube
- ✓ integration.Coverage
- ✓ reporting.Authentication vs Authorization concepts.
- ✓ Session vs Token, JWT introduction.

### Bài tập

- ❖ Cấu hình SonarQube và chạy phân tích code.

### User Story

Là Leader, tôi muốn hệ thống được kiểm tra chất lượng code tự động.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

Sprint 4

## Chất lượng & ổn định

Đầu vào chuẩn, lỗi có cấu trúc, log đúng chỗ; kiểm thử & quality gate



### Điều kiện DONE:

- Bean Validation hoạt động.
- Global Exception Handler trả lỗi chuẩn.
- Coverage  $\geq 60\%$ , duplication  $\leq 3\%$ .
- Maintainability đạt mức A.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

**Sprint 5**

## Bảo mật thực chiến

Spring Security 6; JWT access/refresh; role-based authorization

**Buổi 13**

### Spring Security Setup

- ✓ Spring Security 6 config.
- ✓ UserDetails, PasswordEncoder.
- ✓ Authentication flow.
- ✓ Role-based access control.

### Bài tập

❖ Tạo role ADMIN, TEACHER, STUDENT.

### User Story

Là Admin, tôi muốn phân quyền truy cập cho từng nhóm người dùng.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

## Sprint 5 Bảo mật thực chiến

Spring Security 6; JWT access/refresh; role-based authorization

Buổi 14

### JWT Access/Refresh

- ✓ JWT token generation.
- ✓ Access vs Refresh token.
- ✓ Token validation.
- ✓ Role-based authorization.

### Bài tập

- ❖ API đăng nhập trả access+refresh token.

### User Story

Là Người dùng, tôi muốn đăng nhập và nhận token để gọi API bảo mật.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

**Sprint 5**

## Bảo mật thực chiến

Spring Security 6; JWT access/refresh; role-based authorization

**Buổi 15**

### Refresh Token Rotation & Logout

- ✓ Refresh token strategy.
- ✓ Token rotation.
- ✓ Secure logout.
- ✓ Token revocation.

### Bài tập

❖ Logout vô hiệu hóa refresh token.

### User Story

Là Người dùng, tôi muốn logout để token không thể sử dụng nữa.

# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

Sprint 5

## Bảo mật thực chiến

Spring Security 6; JWT access/refresh; role-based authorization



### Điều kiện DONE:

- Entities: UserAccount, Role, RefreshToken tạo OK.
- Flow: /auth/login → access+refresh hoạt động.
- Phân quyền ADMIN, TEACHER, STUDENT.
- Logout vô hiệu hóa refresh token.



# Chương trình học chi tiết

6 Sprint - 18 buổi học - Từ cơ bản đến module hoàn chỉnh

Sprint 6

## Case Study & hoàn thiện

Tự triển khai một hệ thống hoàn chỉnh

# Dự án mẫu "Quản lý học viên lớp học"

ERD và API design cho hệ thống hoàn chỉnh



## Database Schema (ERD)

### Student

id, code, fullName, email, status

### Teacher

id, fullName, email

### Classroom

id, code, name, semester, teacher\_id

### Course

id, code, name, credit

### Enrollment

id, student\_id, classroom\_id, enrolledAt, status

### Score

id, enrollment\_id, type, value, gradedAt

### UserAccount

id, username, passwordHash, student\_id?, teacher\_id?

### Role

id, name + user\_roles(user\_id, role\_id)

### Quan hệ chính:

- Student–Enrollment–Classroom (N–N)
- Teacher–Classroom (1–N)
- Enrollment–Score (1–N)

# Dự án mẫu "Quản lý học viên lớp học"

ERD và API design cho hệ thống hoàn chỉnh

## <> API Endpoints

### Authentication:

POST /auth/login

POST /auth/refresh

POST /auth/logout

### Core Resources:

CRUD /students, /teachers

CRUD /classrooms, /enrollments

### Advanced Features:

GET /classrooms/1/students

POST /classrooms/1/enroll

GET /POST/enrollments/1/scores

### API Features:

- Tất cả trả DTO, có phân trang & sort
- Mã lỗi chuẩn HTTP
- OpenAPI/Swagger documentation

# Definition of Done (DoD)

Tiêu chuẩn chất lượng cho mỗi Sprint

## ✓ Technical Requirements

- ✓ Pass test & linter
- ✓ AC (Acceptance Criteria) đạt
- ✓ Swagger cập nhật
- ✓ Log/exception chuẩn

## ✓ Quality & Review

- ✓ Review 2 chiều (peer+mentor)
- ✓ Sonar không bug/vuln blocker
- ✓ Coverage  $\geq 60\%$
- ✓ Demo functional

### Đánh giá & tốt nghiệp:

**40%**

Chức năng + AC đạt

**20%**

Code quality (Sonar)

**20%**

Test &amp; CI local

**20%**

Demo + tài liệu