# Tuần 05 – Kế hoạch thực hiện (VN‑Law RAG)

## Đã thực hiện (đến cuối tuần 04)

* Dựng và chạy ổn định docker‑compose stack: mysql, redis, chroma, rag-service, law-service (đã kiểm tra cổng và readiness).
* Sửa lỗi Flyway với MySQL 8.0 bằng cách bổ sung org.flywaydb:flyway-mysql; migrations V1–V3 chạy thành công, tạo bảng + FULLTEXT + seed mẫu.
* Bổ sung và chuẩn hóa RAG service:
  + Thêm endpoint POST /gen và POST /admin/reindex (stub) ; chuẩn hóa JSON keys theo RagClient (content, law\_code, node\_path, node\_id).
  + Giữ /qa, thêm lọc theo effective\_at, cơ chế lazy init model/Chroma và endpoint /ready để kiểm tra sẵn sàng.
* Ổn định law-service:
  + Sửa lỗi 500 ở TOC bằng fetch‑join parent (findByLaw\_IdOrderBySortKeyAscWithParent) và sử dụng trong LawController.
  + Xác nhận các endpoint chính hoạt động: liệt kê/tìm kiếm luật, fulltext nodes, QA (text/plain và JSON), readiness Actuator.
* Thêm README tổng hướng dẫn chạy và test (ưu tiên PowerShell/Windows), bao gồm phần embed vào Chroma và khắc phục sự cố.
* Smoke test đã chạy: GET /api/laws, GET /api/laws/search, GET /api/nodes/search/fulltext, kiểm tra /actuator/health/readiness và rag-service /ready.

## 1) Mục tiêu tuần 05

* RAG end‑to‑end chạy ổn trên docker‑compose: hỏi đáp có ngữ cảnh + citation.
* Chuẩn hóa pipeline embed (DB → Chroma) phù hợp schema hiện tại, có reindex thủ công.
* Ổn định API phía law-service: QA/Gen/TOC, xử lý lỗi rõ ràng.
* Tối thiểu hóa rủi ro triển khai: log, readiness, tài liệu chạy thử.

## 2) Kết quả mong đợi (Acceptance Criteria)

* Gọi POST /api/qa trả về answer + context[] gồm content, lawCode, nodePath, nodeId hợp lệ.
* Gọi POST /api/qa/gen trả answer + citations[] + usedNodes[] (ít nhất 3 citation đúng khi có dữ liệu).
* Chạy embed trên container RAG thành công, dữ liệu được ghi vào Chroma, truy vấn có ngữ cảnh.
* TOC /api/laws/{id}/toc trả về cây đầy đủ, không lỗi 500.
* README hướng dẫn chạy/test mới cập nhật, ai cũng chạy được trong 10–15 phút.

## 3) Hạng mục công việc

1. Dữ liệu & Embedding (RAG)

* Sửa embed\_laws.py khớp schema Flyway: dùng laws.code (thay law\_code), node\_path lấy từ law\_nodes.path, đảm bảo trả node\_id, effective\_start, effective\_end.
* Kiểm tra/loại bỏ bản ghi rỗng; batch upsert (64 items/batch như hiện tại).
* Chạy trong container: cài pymysql, thực thi script, xác nhận số chunk nạp vào Chroma.
* Chuẩn bị lệnh reindex thủ công: docker compose exec rag-service python /app/embed\_laws.py.

1. RAG service (Python)

* Hoàn thiện /gen: chấp nhận k, max\_tokens, temperature, gom citations từ context, chuẩn hóa khóa JSON theo RagClient.
* Cải thiện log/error: trả mã lỗi + body rõ ràng khi chưa sẵn sàng hoặc thiếu question.
* Đảm bảo filter theo effective\_at đã hoạt động (đã có), test với ngày khác nhau.

1. Law service (Spring Boot)

* Đã fix TOC (fetch parent) – rà soát lại N+1, hiệu năng.
* Kiểm thử controller: QAController (text/plain + JSON), NodeSearchService (fulltext) ở mức tích hợp tối thiểu.
* Bật admin key qua cấu hình và test /api/admin/reindex (proxy đến RAG stub).

1. Tài liệu & Công cụ

* Cập nhật README.md (đã thêm) + bổ sung mục reindex, embed.
* Cập nhật Postman collection: thêm QA Gen, Admin Reindex, ví dụ effective\_at.

## 4) Lịch trình gợi ý (5 ngày)

* Ngày 1 (Mon): Sửa embed\_laws.py, thử embed trên môi trường docker, xác thực dữ liệu trong Chroma.
* Ngày 2 (Tue): Hoàn thiện /gen + chuẩn JSON, kiểm thử QA/Gen với dữ liệu đã embed.
* Ngày 3 (Wed): Viết/rá soát test tích hợp tối thiểu cho QAController, NodeSearch; kiểm tra TOC với dữ liệu mẫu.
* Ngày 4 (Thu): Ổn định log/readiness, dọn lỗi biên; cập nhật Postman/README chi tiết.
* Ngày 5 (Fri): Smoke test end‑to‑end, ghi nhận issue tồn, chốt demo nội bộ.

## 5) Rủi ro & phương án

* Model tải chậm/lỗi mạng: cho phép lazy init + endpoint /ready để kiểm tra, hướng dẫn preload.
* Lệch schema DB: chuẩn hóa embed\_laws.py, thêm ghi chú trong README.
* Chroma trống → câu trả lời nghèo ngữ cảnh: nhấn mạnh bước embed và cung cấp script/lệnh nhanh.

## 6) Definition of Done (DoD)

* Compose lên: mysql, redis, chroma, rag-service, law-service đều sẵn sàng.
* /api/qa và /api/qa/gen hoạt động với dữ liệu đã embed, trả về citation hợp lệ.
* Tài liệu chạy thử hoàn chỉnh; Postman có sẵn request mẫu; tối thiểu 1 video/gif minh hoạ (tuỳ chọn).