

ĐỀ CƯƠNG TỐT NGHIỆP

HỆ THỐNG HỖ TRỢ MÔN HỌC ĐỒ ÁN

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành: **CÔNG NGHỆ PHẦN MỀM**

Giảng viên hướng dẫn : ThS. Nguyễn Huy Cường

Sinh viên thực hiện : Nguyễn Tri Bảo Thắng

MSSV: 2180601452 Lớp: 21DTHA1

1.1 Lý do chọn đề tài

Quản lý môn học đồ án truyền thống (email, excel, giấy tờ) thiếu hiệu quả, gây khó khăn trong đồng bộ dữ liệu, theo dõi tiến độ, và đánh giá minh bạch. Sinh viên mất phương hướng, giảng viên tốn thời gian quản lý thủ công, dữ liệu dễ thất lạc. Hệ thống “Quản lý các môn học đồ án” kết hợp mã nguồn mở Sakai và các công nghệ hiện đại (Spring Boot, Kafka, Docker) nhằm tự động hóa quy trình, tăng tính minh bạch, tối ưu hiệu suất, và tận dụng Sakai để giảm chi phí phát triển.

1.2 Mục tiêu đề tài

Xây dựng hệ thống quản lý đồ án hiệu quả, tích hợp Sakai, hỗ trợ tương tác giảng viên - sinh viên.

Tạo giao diện trực quan, quản lý đề tài, tiến độ, đánh giá minh bạch.

Tự động hóa thông báo, lưu trữ dữ liệu an toàn, phân quyền bảo mật.

Nâng cao chất lượng đào tạo, tối ưu quy trình qua công nghệ hiện đại.

1.3 Tổng quan/Cơ sở lý luận

Tổng quan: Phương pháp quản lý truyền thống thiếu hệ thống. Các LMS như Moodle chưa tối ưu cho đồ án. Sakai cung cấp nền tảng mã nguồn mở linh hoạt, kết hợp công nghệ hiện đại để giải quyết vấn đề.

Cơ sở lý luận: Dựa trên nghiên cứu LMS (Sakai), quản lý dự án phần mềm, và công nghệ: Spring Boot (RESTful API), MySQL và Redis (lưu trữ), Kafka (thông báo real-time), Docker + Kubernetes (triển khai cloud-native).

1.4 Nội dung và phương pháp nghiên cứu

1.4.1 Đối tượng nghiên cứu:

- + Giảng viên: Quản lý đề tài, tiến độ, đánh giá.
- + Sinh viên: Thực hiện đồ án, nộp bài, nhận phản hồi.
- + Nhà quản lý: Giám sát, tổng hợp dữ liệu.
- + Sakai: Tích hợp quản lý tài khoản, tài liệu.

1.4.2 Phạm vi nghiên cứu:

- + Nội dung: Quản lý toàn diện quy trình đồ án.
- + Đối tượng: Khoa Công nghệ Thông tin, mở rộng tương lai.

+ Công nghệ: Spring Boot, MySQL, Redis, Bootstrap, Sakai, Docker, Kubernetes.

+ Thời gian: 02/2025 - 06/2025.

1.4.3 Nội dung nghiên cứu:

- + Phân tích yêu cầu từ Sakai và thực tế.
- + Thiết kế hệ thống: Kiến trúc MVC, giao diện, cơ sở dữ liệu.
- + Phát triển: Backend (Spring Boot), frontend (Bootstrap), tích hợp Sakai.
- + Triển khai: Docker + Kubernetes.

1.4.4 Phương pháp nghiên cứu:

- + Khảo sát: Bảng hỏi, phỏng vấn.
- + Thiết kế: Sơ đồ UML (Use Case, ERD).
- + Phát triển: Chia module, kiểm thử từng giai đoạn.
- + Kiểm thử: Unit Test, Integration Test.

1.5 Dự kiến kết quả

Danh sách chức năng chính và công cụ sử dụng:

1.5.1 Quản lý toàn diện quy trình đồ án

- + Đăng ký đề tài qua biểu mẫu trực tuyến: Sinh viên điền thông tin (tiêu đề, mô tả, lĩnh vực công nghệ) qua giao diện web, giảng viên duyệt hoặc từ chối.
- + Theo dõi tiến độ: Hiện thị danh sách nhiệm vụ được giao cho từng sinh viên/nhóm, kết hợp biểu đồ trực quan (cột, đường) để theo dõi phần trăm hoàn thành.
- + Nộp bài: Sinh viên tải tệp (PDF, ZIP) qua giao diện, thêm ghi chú phiên bản (v1, v2) để phân biệt các lần nộp.
- + Đánh giá trực tuyến: Giảng viên nhập điểm số chi tiết theo tiêu chí (mã nguồn, báo cáo) và thêm nhận xét, lưu trữ minh bạch.
- + Công cụ: Spring Boot (API: POST /topics, GET /progress), MySQL (lưu trữ dữ liệu), Chart.js (biểu đồ), Sakai (quản lý tài khoản sinh viên).

1.5.2 Giao diện trực quan

- + Thiết kế responsive: Giao diện tự động thích nghi với mọi thiết bị (máy tính, điện thoại, máy tính bảng) để sử dụng linh hoạt.
- + Tooltip hướng dẫn: Cung cấp hướng dẫn chi tiết cho từng chức năng (đăng ký đề tài, nộp bài) khi người dùng di chuột qua.

+ Dễ sử dụng: Font chữ lớn, nút bấm rõ ràng, bố cục thân thiện với người dùng không chuyên về công nghệ.

+ Công cụ: Bootstrap (CSS responsive, tooltip, form), Chart.js (trực quan hóa dữ liệu).

1.5.3 Hỗ trợ giảng viên quản lý đề tài, tiến độ, chấm điểm

+ Quản lý đề tài: Giảng viên tạo danh sách đề tài, duyệt đề xuất từ sinh viên, phân công cho nhóm/cá nhân qua giao diện trực quan.

+ Theo dõi tiến độ: Xem biểu đồ cột/đường thể hiện nhiệm vụ hoàn thành, tỷ lệ tiến độ của từng sinh viên/nhóm.

+ Chấm điểm minh bạch: Nhập điểm theo tiêu chí cụ thể (mã nguồn 40%, báo cáo 60%), lưu lịch sử đánh giá để tra cứu.

+ Công cụ: Spring Boot (API: PUT /topics/assign, POST /evaluations), Chart.js (biểu đồ), MySQL (lưu trữ dữ liệu).

1.5.4 Thông báo tự động

+ Gửi email và thông báo web: Thông báo các sự kiện quan trọng (duyet đề tài, nộp bài thành công) qua email và giao diện hệ thống.

+ Nhắc nhở cá nhân hóa: Gửi thông báo dựa trên tiến độ từng sinh viên, ví dụ: “Còn 24 giờ để nộp báo cáo” qua email/web.

+ Công cụ: Apache Kafka (xử lý thông báo real-time), Spring Boot (JavaMailSender), Redis (cache thông báo để tối ưu hiệu suất).

1.5.5 Lưu trữ dữ liệu

+ Tập trung tài liệu: Quản lý báo cáo, mã nguồn trong cơ sở dữ liệu tập trung, hỗ trợ tải lên/tải xuống dễ dàng.

+ Tìm kiếm nhanh: Tìm kiếm bằng từ khóa (tên đề tài, mã sinh viên) với tốc độ cao.

+ Lọc dữ liệu: Lọc tài liệu theo kỳ học (học kỳ 1, 2) hoặc nhóm sinh viên qua bộ lọc trên giao diện.

+ Công cụ: MySQL (lưu trữ chính), Vector DB (tìm kiếm nhanh), Redis (cache truy vấn để tăng tốc độ).

1.5.6 Đánh giá minh bạch

- + Công khai tiêu chí đánh giá: Hiển thị tiêu chí (mã nguồn, báo cáo) trước khi chấm điểm để sinh viên nắm rõ.
- + Lưu lịch sử: Ghi lại các lần chỉnh sửa bài nộp và điểm số để sinh viên, giảng viên tra cứu khi cần.
- + Công cụ: Spring Boot (API: GET /evaluations/history), MySQL (lưu trữ lịch sử).

1.5.7 Phân quyền và bảo mật

- + Phân quyền theo vai trò: Admin (toàn quyền), Giảng viên (quản lý nhóm), Sinh viên (xem thông tin cá nhân).
- + Mã hóa JWT: Đảm bảo xác thực an toàn cho người dùng khi đăng nhập, sử dụng Refresh Token để gia hạn phiên.
- + Giới hạn đăng nhập sai: Khóa tài khoản tạm thời sau 5 lần nhập sai mật khẩu để ngăn tấn công brute-force.
- + Công cụ: Spring Boot (Spring Security, JWT), Redis (lưu Refresh Token).

1.5.8 Báo cáo và thống kê

- + Tạo báo cáo tiến độ: Hiển thị trạng thái (đang làm, hoàn thành) của từng sinh viên/nhóm qua giao diện.
- + Tổng hợp điểm số: Thống kê chi tiết điểm theo nhóm/sinh viên, hỗ trợ phân tích kết quả.
- + Xuất file: Xuất báo cáo dưới dạng Excel/PDF để lưu trữ hoặc nộp lên khoa.
- + Công cụ: Spring Boot (Apache POI cho Excel, iText cho PDF), Chart.js (biểu đồ), MySQL (truy vấn dữ liệu).

1.5.9 Admin

- + Quản lý tài khoản: Thêm, sửa, xóa thông tin người dùng (sinh viên, giảng viên) qua biểu mẫu và danh sách trực quan.
- + Phân quyền chi tiết: Gán vai trò (Admin, Giảng viên, Sinh viên) qua giao diện kéo thả hoặc checkbox, tùy chỉnh quyền truy cập cụ thể (xem, chỉnh sửa).
- + Cấu hình hệ thống: Tùy chỉnh logo, màu sắc giao diện, thông báo toàn cục qua biểu mẫu quản trị.

- + Theo dõi log hệ thống: Xem bảng nhật ký hiển thị hành động (đăng nhập, nộp bài), thời gian, người thực hiện để giám sát hoạt động.
- + Thành lập hội đồng chấm điểm: Tạo danh sách giảng viên, thời gian, địa điểm bảo vệ qua giao diện quản lý.
- + Quy định thành phần điểm: Thiết lập công thức tính điểm (trung bình, tỷ lệ %) cho từng môn học hoặc kỳ học.
- + Xuất biểu mẫu liên quan: Xuất biên bản chấm điểm, danh sách bảo vệ dưới dạng PDF/Excel để nộp khoa hoặc lưu trữ.
- + Quản lý danh sách môn học: Tạo, sửa, xóa thông tin môn đồ án (tên, kỳ học) qua giao diện.
- + Thiết lập thời gian kỳ học: Đặt ngày bắt đầu, kết thúc kỳ học, thông báo toàn hệ thống khi thay đổi.
- + Kiểm soát quyền truy cập API: Tạo, quản lý token cho ứng dụng bên ngoài, giới hạn quyền đọc/ghi qua giao diện.
- + Quản lý tài nguyên hệ thống: Theo dõi CPU, RAM, phân bổ tài nguyên server để đảm bảo hoạt động ổn định.
- + Tạo tài khoản hàng loạt: Nhập danh sách từ file Excel, tự động gửi email thông báo tài khoản cho người dùng.
- + Khóa/mở khóa hệ thống: Khóa hệ thống khi bảo trì, hiển thị thông báo toàn cục trong thời gian tạm dừng.
- + Quản lý sao lưu dữ liệu định kỳ: Thiết lập lịch tự động sao lưu, lưu file nén để khôi phục khi cần.
- + Tạo thông báo khẩn cấp: Gửi tin nhắn tức thời đến toàn bộ người dùng qua email hoặc giao diện web.
- + Phân tích hiệu suất hệ thống: Hiển thị số liệu (người dùng hoạt động, thời gian phản hồi) và biểu đồ trực quan.
- + Quản lý giảng viên dự bị: Tạo danh sách thay thế, kích hoạt khi cần cho các môn học hoặc nhóm.
- + Công cụ: Spring Boot (API: POST /users, GET /logs), MySQL (lưu trữ), Docker (sao lưu), Kubernetes (quản lý tài nguyên).

1.5.10 Giảng viên

- + Quản lý đề tài: Tạo danh sách đề tài, duyệt đề xuất từ sinh viên, phân công cho nhóm/cá nhân qua giao diện.
- + Giao nhiệm vụ: Giao nhiệm vụ hàng tuần (mô tả, deadline) cho nhóm/sinh viên, theo dõi trạng thái hoàn thành.
- + Nhận xét: Xem tài liệu nộp, gửi phản hồi chi tiết qua giao diện để sinh viên chỉnh sửa.
- + Chấm điểm: Nhập điểm trực tuyến theo tiêu chí cụ thể, xuất báo cáo đánh giá dưới dạng PDF/Excel.
- + Phân công nhóm: Tùy chọn số lượng thành viên, tự chọn hoặc cố định qua danh sách sinh viên.
- + Giao đề tài cho nhóm: Gán đề tài tự chọn từ sinh viên hoặc chọn sẵn từ danh sách qua giao diện.
- + Xem báo cáo tiến độ: Xem tiến độ nhóm/thành viên qua biểu đồ (cột, đường) và danh sách nhiệm vụ hoàn thành.
- + Tạo lịch họp nhóm: Lên lịch họp trực tuyến (Google Meet) hoặc trực tiếp, gửi thông báo qua email/web.
- + Quản lý tài liệu giảng viên: Lưu trữ, chia sẻ tài liệu hướng dẫn (PDF, Word) với nhóm qua giao diện.
- + Kiểm tra tính hợp lệ bài nộp: Xác nhận định dạng (PDF, ZIP), nội dung trước khi chấm điểm trực tuyến.
- + Tạo câu hỏi kiểm tra: Tạo câu hỏi trước bảo vệ, gửi qua giao diện để sinh viên chuẩn bị.
- + Xem lịch sử trao đổi: Xem tin nhắn, nhận xét với sinh viên để tra cứu hoặc theo dõi tiến trình.
- + Gửi thông báo cá nhân: Gửi thông báo đến từng sinh viên qua email hoặc giao diện với nội dung cụ thể.
- + Quản lý phản hồi nhóm: Tập hợp, gửi phản hồi tập thể, theo dõi trạng thái xử lý qua giao diện.
- + Tạo báo cáo đánh giá cá nhân: Tổng hợp điểm số, nỗ lực của từng sinh viên qua giao diện.

+ Phân tích mức độ tham gia: Xem thời gian làm việc, số nhiệm vụ hoàn thành, biểu đồ trực quan.

+ Tạo tài liệu mẫu: Tạo tài liệu mẫu (báo cáo, mã nguồn) để sinh viên tải về tham khảo qua giao diện.

+ Công cụ: Spring Boot (API: POST /tasks, GET /progress), Google Meet (họp), GitHub (mã nguồn), Chart.js (biểu đồ).

1.5.11 Sinh viên

+ Đề xuất đề tài: Điền biểu mẫu trực tuyến (tiêu đề, mô tả) hoặc chọn từ danh sách đề tài sẵn có.

+ Nộp báo cáo: Tải báo cáo/mã nguồn (PDF, ZIP) qua giao diện, thêm ghi chú phiên bản (v1, v2).

+ Xem phản hồi: Xem nhận xét từ giảng viên qua giao diện, tra cứu lịch sử đề cải thiện bài nộp.

+ Tra cứu điểm: Xem điểm chi tiết theo tiêu chí, nhận xét, và lịch sử chỉnh sửa qua giao diện.

+ Tham gia nhóm: Đăng ký nhóm qua giao diện, xem thông tin thành viên nhóm hiện tại.

+ Kiểm tra tiến độ cá nhân: Xem danh sách nhiệm vụ được giao, trạng thái hoàn thành qua giao diện.

+ Gửi câu hỏi thảo luận: Đặt câu hỏi trên diễn đàn trực tuyến để hỏi đáp với nhóm/giảng viên.

+ Xem lịch họp nhóm: Xem thời gian, địa điểm họp qua giao diện, nhận nhắc nhở qua email/web.

+ Tải tài liệu mẫu: Tải báo cáo, mã nguồn mẫu từ giảng viên qua giao diện để tham khảo.

+ Theo dõi thời gian cá nhân: Ghi nhận giờ làm việc mỗi phiên qua nút bắt đầu/kết thúc trên giao diện.

+ Gửi yêu cầu gia hạn: Gửi biểu mẫu gia hạn nộp bài (lý do), chờ giảng viên phê duyệt.

+ Xem tài nguyên nhóm: Xem danh sách tài liệu, link được chia sẻ trong nhóm qua giao diện.

+ Kiểm tra mã nguồn: Chạy thử mã nguồn (Java, C#) trực tuyến trước khi nộp để kiểm tra lỗi.

+ Xem lịch sử chỉnh sửa cá nhân: Xem các phiên bản bài nộp đã gửi để tra cứu qua giao diện.

+ Công cụ: Spring Boot (API: POST /submissions), GitHub (commit mã nguồn), Sakai (diễn đàn thảo luận).

1.5.12 AI hỗ trợ

+ AI nhắc lịch và theo dõi tiến độ: Theo dõi tiến độ cá nhân qua dữ liệu nhiệm vụ, gửi nhắc nhở thông minh (email/web) khi gần hạn, dự đoán rủi ro trễ hạn, gợi ý kế hoạch làm việc.

+ AI dự đoán đề tài phù hợp: Phân tích sở trường (điểm số, sở thích), gợi ý đề tài phù hợp, dự đoán xu hướng công nghệ, đề xuất giảng viên hướng dẫn dựa trên chuyên môn.

+ Công cụ: Spring Boot (logic AI), Apache Kafka (thông báo real-time), MySQL (dữ liệu sinh viên).

1.5.13 Trao đổi, thảo luận

+ Đăng câu hỏi công nghệ trên diễn đàn trực tuyến, trả lời/bình luận, vote up/down để đánh giá, chọn câu trả lời tốt nhất, tích điểm danh tiếng, tìm kiếm bằng từ khóa/thẻ.

+ Công cụ: Sakai (diễn đàn tích hợp), Spring Boot (API: POST /forums).

1.5.14 Nhật ký hoạt động nhóm

+ Ghi lại timeline (nộp bài, nhận phản hồi) qua giao diện, hiển thị chi tiết cho nhóm và giảng viên, xuất file PDF để báo cáo/lưu trữ.

+ Công cụ: Spring Boot (API: GET /group-logs), MySQL (lưu trữ).

1.5.15 Gợi ý từ khóa đề tài

+ Tự động đề xuất từ khóa (AI, Java) khi tìm kiếm/đăng ký đề tài, hỗ trợ tìm kiếm nhanh bằng từ khóa gợi ý.

+ Công cụ: Spring Boot (logic gợi ý), Vector DB (tìm kiếm hiệu quả).

1.5.16 Đề xuất lịch trình cá nhân

+ Lập lịch làm việc dựa trên thời gian rảnh, phân tích tiến độ để ưu tiên nhiệm vụ, gửi nhắc nhở qua email/web, hiển thị lịch cho giảng viên.

- + Công cụ: Spring Boot (API: POST /schedules), Apache Kafka (thông báo).

1.5.17 Quản lý danh sách đề tài bắt buộc

- + Quản lý danh sách đề tài bắt buộc (hashtag #Java, #SQL), sinh viên chọn qua giao diện, theo dõi số lượng đăng ký để tránh vượt quá.

- + Công cụ: Spring Boot (API: GET /mandatory-topics), MySQL (lưu trữ).

1.5.18 Quản lý tài liệu tham khảo

- + Tải lên tệp (PDF, link) với thẻ phân loại, chia sẻ trong nhóm qua giao diện, giảng viên đề xuất tài liệu để sinh viên sử dụng.

- + Công cụ: Spring Boot (API: POST /references), MySQL (lưu trữ).

1.5.19 Cảnh báo trùng lặp đề tài

- + Kiểm tra trùng lặp khi đăng ký, hiển thị cảnh báo nếu trùng đề tài cũ, hỗ trợ tra cứu danh sách đề tài đã đăng ký.

- + Công cụ: Spring Boot (API: POST /topics/check), Vector DB (tìm kiếm).

1.5.20 Tích hợp bảng đánh giá nhóm

- + Sinh viên đánh giá thành viên (thang điểm 1-10), tổng hợp điểm trung bình, hiển thị biểu đồ so sánh đóng góp, xuất báo cáo PDF.

- + Công cụ: Spring Boot (API: POST /group-ratings), Chart.js (biểu đồ).

1.5.21 Kiểm tra tiến độ tự động

- + So sánh deadline với thời gian nộp bài, báo cáo trạng thái (hoàn thành/chậm trễ), gửi thông báo cho sinh viên và giảng viên.

- + Công cụ: Spring Boot (API: GET /progress-check), Apache Kafka (thông báo).

1.5.22 Công cụ lập kế hoạch chi tiết

- + Chia nhỏ nhiệm vụ với deadline cụ thể, đánh dấu hoàn thành qua nút/checklist, hiển thị tiến độ bằng phần trăm.

- + Công cụ: Spring Boot (API: POST /plans), MySQL (lưu trữ).

1.5.23 Gợi ý công cụ lập trình

- + Gợi ý công cụ (IntelliJ, Postman) dựa trên hashtag/nội dung đề tài, cung cấp link tải, giảng viên thêm gợi ý tùy chỉnh.

- + Công cụ: Spring Boot (API: GET /tools-suggestion).

1.5.24 Theo dõi thời gian làm việc

+ Ghi nhận giờ bắt đầu/kết thúc qua giao diện, tổng hợp thời gian theo tuần/kỳ, báo cáo giảng viên để đánh giá tham gia.

+ Công cụ: Spring Boot (API: POST /time-tracking), MySQL (lưu trữ).

1.5.25 Gửi yêu cầu hỗ trợ

+ Sinh viên gửi yêu cầu (nội dung, mức độ ưu tiên) qua biểu mẫu, giảng viên phản hồi trên giao diện, lưu lịch sử yêu cầu.

+ Công cụ: Spring Boot (API: POST /support-requests).

1.5.26 Kiểm tra tài liệu nộp bài

+ Kiểm tra định dạng (PDF), số trang tối thiểu, nội dung bắt buộc, thông báo lỗi nếu không đạt, báo giảng viên khi đạt yêu cầu.

+ Công cụ: Spring Boot (API: POST /submission-check).

1.5.27 Quản lý lịch bảo vệ

+ Lên lịch bảo vệ (ngày, giờ, phòng), kiểm tra xung đột, gửi thông báo qua email/web/Google Meet.

+ Công cụ: Spring Boot (API: POST /defense-schedule), Google Meet (họp).

1.5.28 Hỗ trợ lập trình trực tuyến

+ Viết/chạy mã (Java, C#) qua giao diện, gợi ý cú pháp, giảng viên xem mã để hỗ trợ/đánh giá.

+ Công cụ: Spring Boot (API: POST /code-run), GitHub (mã nguồn).

1.5.29 Quản lý câu hỏi thường gặp

+ Thêm FAQ (câu hỏi, trả lời) qua giao diện, tìm kiếm bằng từ khóa, sinh viên gửi câu hỏi mới để bổ sung.

+ Công cụ: Spring Boot (API: POST /faqs), MySQL (lưu trữ).

1.5.30 Quản lý phiên bản tài liệu

+ Lưu trữ các phiên bản (v1, v2) khi nộp, hiển thị lịch sử thay đổi, hỗ trợ khôi phục phiên bản cũ.

+ Công cụ: Spring Boot (API: GET /document-versions), MySQL (lưu trữ).

1.5.31 Đánh giá năng lực cá nhân

+ Sinh viên tự đánh giá kỹ năng qua biểu mẫu, tổng hợp từ bài nộp và đánh giá nhóm, cung cấp tư vấn nghề nghiệp.

+ Công cụ: Spring Boot (API: POST /self-assessment), MySQL (lưu trữ).

1.5.32 Nhắc nhở sinh viên không hoạt động

+ Theo dõi thời gian không hoạt động (>7 ngày), gửi cảnh báo qua email/giao diện, báo cáo giảng viên.

+ Công cụ: Spring Boot (API: GET /inactive-users), Apache Kafka (thông báo).

1.5.33 Phân tích hiệu suất nhóm

+ Phân tích tiến độ, điểm số qua giao diện, so sánh giữa các nhóm bằng biểu đồ (cột, tròn).

+ Công cụ: Spring Boot (API: GET /group-performance), Chart.js (biểu đồ).

1.5.34 Kiểm tra tài liệu yêu cầu

+ Kiểm tra mục bắt buộc (mục tiêu, phạm vi), báo lỗi thiếu sót, gửi thông tin đạt yêu cầu đến giảng viên.

+ Công cụ: Spring Boot (API: POST /requirement-check).

1.5.35 Quản lý phản hồi giảng viên

+ Phân loại phản hồi (mã nguồn, tài liệu), sinh viên đánh dấu “Đã xử lý”, thông báo giảng viên khi hoàn tất.

+ Công cụ: Spring Boot (API: POST /feedback-management).

1.5.36 Tạo báo cáo tự động

+ Tổng hợp dữ liệu tiến độ, điểm số, xuất báo cáo PDF/Word theo mẫu, hỗ trợ chỉnh sửa trước khi nộp.

+ Công cụ: Spring Boot (Apache POI, iText), MySQL (truy vấn).

1.5.37 Quản lý tài nguyên nhóm

+ Tải lên tệp/link với thẻ phân loại, chia sẻ trong nhóm, giảng viên xem để đánh giá quá trình làm việc.

+ Công cụ: Spring Boot (API: POST /group-resources), MySQL (lưu trữ).

1.5.38 Thông báo nhóm qua tin nhắn

+ Gửi tin nhắn thông báo đến toàn nhóm qua giao diện/email để phối hợp công việc nhanh chóng.

+ Công cụ: Spring Boot (API: POST /group-messages), Apache Kafka (thông báo).

1.5.39 Kiểm tra mã nguồn tự động

+ Kiểm tra mã nguồn trước khi nộp, báo cáo lỗi cú pháp/logic cơ bản để sinh viên chỉnh sửa.

+ Công cụ: Spring Boot (API: POST /code-check), GitHub (mã nguồn).

1.5.40 Đánh giá bài thuyết trình

+ Đánh giá thuyết trình trong buổi bảo vệ theo tiêu chí (nội dung, trình bày) qua biểu mẫu trực tuyến.

+ Công cụ: Spring Boot (API: POST /presentation-evaluation).

1.5.41 Quản lý tài liệu nhóm trực tuyến

+ Chỉnh sửa tài liệu nhóm theo thời gian thực qua giao diện để tăng hiệu quả làm việc nhóm.

+ Công cụ: Spring Boot (API: POST /group-documents), MySQL (lưu trữ).

1.5.42 Gợi ý tài liệu tham khảo

+ Gợi ý tài liệu liên quan đến đề tài dựa trên nội dung/từ khóa qua giao diện để hỗ trợ nghiên cứu.

+ Công cụ: Spring Boot (API: GET /reference-suggestions).

1.5.43 Quản lý lịch sử chỉnh sửa đề tài

+ Lưu trữ các thay đổi đề tài qua giao diện để tra cứu/khôi phục phiên bản cũ.

+ Công cụ: Spring Boot (API: GET /topic-history), MySQL (lưu trữ).

1.5.44 Quản lý danh sách tài liệu nhóm

+ Phân loại tài liệu nhóm (báo cáo, mã nguồn) qua giao diện để dễ tra cứu/sử dụng.

+ Công cụ: Spring Boot (API: POST /group-document-list).

1.5.45 Thông báo sự kiện quan trọng

+ Gửi thông báo (hạn nộp bài, lịch bảo vệ) qua email/giao diện để sinh viên nắm bắt kịp thời.

+ Công cụ: Spring Boot (API: POST /event-notifications), Apache Kafka (thông báo).

1.5.46 Quản lý danh sách sinh viên chờ duyệt

+ Quản lý sinh viên chờ duyệt tham gia nhóm/đề tài qua giao diện với tùy chọn duyệt/từ chối.

+ Công cụ: Spring Boot (API: POST /pending-students).

1.5.47 Phân tích dữ liệu đồ án

+ Phân tích tiến độ, điểm số toàn hệ thống qua giao diện với biểu đồ và báo cáo để đánh giá hiệu quả.

+ Công cụ: Spring Boot (API: GET /project-analytics), Chart.js (biểu đồ).

1.5.48 Gửi khảo sát phản hồi

+ Gửi khảo sát qua giao diện để thu thập ý kiến sinh viên về hệ thống/quá trình thực hiện đồ án.

+ Công cụ: Spring Boot (API: POST /surveys).

1.5.49 Quản lý lịch sử truy cập hệ thống

+ Theo dõi đăng nhập, đăng xuất qua giao diện để đảm bảo an ninh hệ thống.

+ Công cụ: Spring Boot (API: GET /access-logs), MySQL (lưu trữ).

1.5.50 Hỗ trợ phân loại tài liệu

+ Phân loại tài liệu nộp tự động (báo cáo, mã nguồn) qua công cụ tích hợp trong giao diện.

+ Công cụ: Spring Boot (API: POST /document-classification).

1.5.51 Hỗ trợ lập trình nhóm

+ Cho phép chỉnh sửa mã nguồn cùng lúc qua giao diện trực tuyến để tăng hiệu quả hợp tác.

+ Công cụ: Spring Boot (API: POST /group-coding), GitHub (mã nguồn).

1.5.52 Hỗ trợ lập kế hoạch bảo vệ

+ Gọi ý lịch tối ưu dựa trên tài nguyên (phòng, giảng viên) và nhóm sinh viên qua giao diện.

+ Công cụ: Spring Boot (API: POST /defense-planning).

1.5.53 Hỗ trợ kiểm tra tài liệu nhóm

+ Sinh viên góp ý tài liệu trước khi nộp qua giao diện với tính năng nhận xét, đánh dấu chỉnh sửa.

+ Công cụ: Spring Boot (API: POST /group-document-review).

1.5.54 Hệ thống hỗ trợ học tập cá nhân

+ Gọi ý tài liệu, video dựa trên điểm yếu (điểm số, kỹ năng), cung cấp tư vấn học tập qua giao diện.

+ Công cụ: Spring Boot (API: GET /learning-support).

1.5.55 Công cụ quản lý danh sách câu hỏi bảo vệ

+ Tạo, lưu trữ câu hỏi bảo vệ qua giao diện, hỗ trợ giảng viên tái sử dụng, sinh viên trả lời trước buổi bảo vệ.

+ Công cụ: Spring Boot (API: POST /defense-questions).

1.5.56 Hỗ trợ phân tích phản hồi giảng viên

+ Tổng hợp nhận xét qua giao diện, nhận diện vấn đề phổ biến với biểu đồ, gợi ý cải thiện hướng dẫn.

+ Công cụ: Spring Boot (API: GET /feedback-analysis), Chart.js (biểu đồ).

1.5.57 Công cụ hỗ trợ lập kế hoạch tài nguyên nhóm

+ Quản lý tài nguyên (phần mềm, tài liệu) qua giao diện, phân bổ thời gian/tài liệu cho từng thành viên.

+ Công cụ: Spring Boot (API: POST /group-resource-planning).

1.5.58 Kết quả tổng thể

+ Hệ thống chuyên nghiệp, giảm áp lực quản lý, nâng cao chất lượng đào tạo, triển khai trên cloud (Docker, Kubernetes), tích hợp Sakai (quản lý tài khoản, tài liệu).

1.6 Tài liệu tham khảo

Spring Boot Documentation (2022), <https://spring.io/projects/spring-boot>.

MySQL 8.0 Reference Manual (2023), <https://dev.mysql.com/doc/refman/8.0/en/>.

Bootstrap v4 Documentation (2023), <https://getbootstrap.com/docs/4.0/>.

Sakai Project (2024), <https://www.sakailms.org/>.

Apache Kafka Documentation, <https://kafka.apache.org/>.

Docker Documentation, <https://docs.docker.com/>.

Kubernetes Documentation, <https://kubernetes.io/>.

GitHub API, <https://docs.github.com/en/rest>.

1.7 Tiến độ thực hiện

Ngày/Tháng/Năm	Tháng 3	Tháng 4	Tháng 5	Tháng 6
Dự kiến nội dung thực hiện				
Nghiên cứu và thu thập yêu cầu				

ĐỀ CƯƠNG ĐỒ ÁN TỐT NGHIỆP
Trang 15

Thiết kế hệ thống				
Phát triển Backend và Cơ sở dữ liệu				
Phát triển Frontend và Giao diện người dùng				
Kiểm thử và Hoàn thiện hệ thống				
Triển khai thử nghiệm và tối ưu hóa				
Viết báo cáo và bảo vệ đồ án				

Tp.HCM, ngày 15 tháng 03 năm 2025
SINH VIÊN THỰC HIỆN
(Ký và ghi rõ họ tên)

Nguyễn Tri Bảo Thắng

Nguyễn Tri Bảo Thắng

XÁC NHẬN CỦA BỘ MÔN

GIẢNG VIÊN HƯỚNG DẪN
(Ký và ghi rõ họ tên)

Nguyễn Thị Cường

Nguyễn Thị Cường