

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

NGHIÊN CỨU VÀ XÂY DỰNG HỆ
THỐNG QUẢN LÝ HỌC TẬP THÔNG
MINH (SMART LMS) DỰA TRÊN
KIẾN TRÚC RAG VÀ MÔ HÌNH
NGÔN NGỮ LỚN

Giảng viên hướng dẫn: TS. Nguyễn Văn B

Sinh viên thực hiện: Nguyễn Quốc Tùnggggg

Mã số sinh viên: 2251120259

Lớp: CN22E

TP. HỒ CHÍ MINH, NĂM 2025

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi dưới sự hướng dẫn của giảng viên hướng dẫn. Các kết quả nêu trong luận văn là trung thực và chưa từng được công bố trong bất kỳ công trình nào khác. Mọi sự giúp đỡ cho việc thực hiện luận văn này đã được cảm ơn và các thông tin trích dẫn trong luận văn đã được chỉ rõ nguồn gốc.

LỜI MỞ ĐẦU

Sự bùng nổ của Trí tuệ nhân tạo tạo sinh (Generative AI) đã mở ra những cơ hội chưa từng có trong việc cá nhân hóa giáo dục. Tuy nhiên, các hệ thống Quản lý học tập (LMS) hiện nay tại Việt Nam chủ yếu vẫn đóng vai trò là kho lưu trữ tài liệu tĩnh, thiếu đi tính tương tác thông minh và khả năng hỗ trợ học sinh học tập chủ động.

Dự án này được thực hiện với mong muốn xây dựng một hệ sinh thái học tập "Smart LMS" dành cho cấp trung học. Điểm đột phá của hệ thống là việc ứng dụng kỹ thuật Retrieval-Augmented Generation (RAG) để tạo ra một Trợ lý ảo (AI Tutor) có khả năng hiểu sâu sắc nội dung bài giảng của giáo viên, từ đó phản hồi chính xác thắc mắc của học sinh. Bên cạnh đó, hệ thống tích hợp các công cụ giám sát thi cử dựa trên AI nhằm đảm bảo tính liêm chính trong môi trường học tập trực tuyến.

MỤC LỤC

Lời cam đoan	i
Lời mở đầu	ii
1 MỞ ĐẦU	1
1.1 Lý do chọn đề tài	1
1.2 Mục tiêu nghiên cứu	1
1.3 Đối tượng và phạm vi nghiên cứu	1
2 CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ	2
2.1 Mô hình ngôn ngữ lớn (LLM) và Gemini API	2
2.2 Kiến trúc RAG (Retrieval-Augmented Generation)	2
2.3 Kiến trúc Web hiện đại: Next.js App Router	2
3 PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	3
3.1 Phân tích yêu cầu chức năng	3
3.1.1 Vai trò Giáo viên	3
3.1.2 Vai trò Học sinh	3
3.2 Thiết kế cơ sở dữ liệu	3
4 HIỆN THỰC HÓA HỆ THỐNG	4
4.1 Xây dựng trợ lý ảo AI Tutor	4
4.2 Hệ thống đánh giá và giám sát thi cử	4
4.2.1 Cơ chế tự động chấm bài	4
4.2.2 Phát hiện hành vi đáng ngờ	4
4.3 Giao diện người dùng (UI/UX)	4
5 ĐÁNH GIÁ VÀ THỬ NGHIỆM	5
5.1 Đánh giá độ chính xác của AI Tutor	5
5.2 Đánh giá hiệu năng hệ thống	5
6 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	6
6.1 Kết quả đạt được	6
6.2 Hướng phát triển tương lai	6
Tài liệu tham khảo	7

DANH SÁCH HÌNH VẼ

DANH SÁCH BẢNG

CHƯƠNG 1

MỞ ĐẦU

1.1 Lý do chọn đề tài

Trong kỷ nguyên giáo dục 4.0, vai trò của giáo viên đang dịch chuyển từ người truyền thụ kiến thức sang người điều phối học tập. Tuy nhiên, khối lượng công việc hành chính và việc theo sát từng học sinh trong lớp học trực tuyến là một thách thức lớn. Hệ thống LMS thông minh cần phải giải quyết được bài toán giảm tải cho giáo viên và tăng tính tự học cho học sinh.

1.2 Mục tiêu nghiên cứu

- Thiết kế kiến trúc hệ thống LMS đa vai trò (Admin, Teacher, Student, Parent).
- Hiện thực hóa module AI Tutor hỗ trợ học tập dựa trên cơ sở dữ liệu vector.
- Xây dựng quy trình tự động hóa đánh giá năng lực học sinh qua AI.
- Đảm bảo an ninh và tính liêm chính học thuật qua hệ thống chống gian lận.

1.3 Đối tượng và phạm vi nghiên cứu

- **Đối tượng:** Quy trình quản lý và hỗ trợ học tập tại các trường trung học.
- **Phạm vi:** Xây dựng ứng dụng Web dựa trên Next.js và tích hợp Google Gemini API.

CHƯƠNG 2

CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

2.1 Mô hình ngôn ngữ lớn (LLM) và Gemini API

Trình bày về khả năng của mô hình Gemini 1.5 Flash trong việc xử lý ngữ cảnh dài (long context window), điều này rất quan trọng cho việc phân tích tài liệu bài giảng.

2.2 Kiến trúc RAG (Retrieval-Augmented Generation)

Đây là nền tảng kỹ thuật của AI Tutor. Quy trình bao gồm:

1. **Văn bản hóa:** Chuyển đổi PDF/DOCX bài giảng thành văn bản thô.
2. **Chia nhỏ (Chunking):** Sử dụng các thuật toán chia đoạn để giữ nguyên ngữ nghĩa.
3. **Nhúng Vector (Embedding):** Chuyển đoạn văn thành các vector không gian.
4. **Truy xuất (Retrieval):** Tìm kiếm đoạn văn có độ tương đồng cao nhất với câu hỏi của học sinh.

2.3 Kiến trúc Web hiện đại: Next.js App Router

Phân tích lợi ích của Server Components và API Routes trong việc bảo mật các Key của AI và tối ưu hiệu suất Rendering.

CHƯƠNG 3

PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1 Phân tích yêu cầu chức năng

3.1.1 *Vai trò Giáo viên*

Quản lý khóa học, tạo bài tập thông minh (Quiz/Essay), chấm bài hỗ trợ bởi AI và giám sát thi cử.

3.1.2 *Vai trò Học sinh*

Tương tác với AI Tutor, làm bài tập, nhận phản hồi tức thì và theo dõi lộ trình học tập.

3.2 Thiết kế cơ sở dữ liệu

Mô tả các thực thể chính dựa trên Schema Prisma:

- User: Quản lý thông tin và phân quyền người dùng.
- LessonEmbedding: Lưu trữ các đoạn văn bản đã được vector hóa của bài học.
- ExamEvent: Ghi lại các hành vi bất thường trong quá trình thi.

CHƯƠNG 4

HIỆN THỰC HÓA HỆ THỐNG

4.1 Xây dựng trợ lý ảo AI Tutor

Mô tả chi tiết quá trình hiện thực hóa tệp src/lib/rag/indexLessonEmbeddings.ts. Cách sử dụng pgvector trong PostgreSQL để lưu trữ và truy vấn vector.

4.2 Hệ thống đánh giá và giám sát thi cử

4.2.1 Cơ chế tự động chấm bài

Sử dụng Prompt Engineering để hướng dẫn Gemini chấm bài tự luận dựa trên Rubric của giáo viên.

4.2.2 Phát hiện hành vi đáng ngờ

Phân tích logic trong suspicious-behavior.ts: Theo dõi chuyển đổi Tab, phím tắt cảm và mất kết nối.

4.3 Giao diện người dùng (UI/UX)

Sử dụng Tailwind CSS và các thành phần từ Shadcn UI để tạo ra một giao diện tối giản, tập trung vào trải nghiệm học tập (Learning-centric design).

CHƯƠNG 5

ĐÁNH GIÁ VÀ THỬ NGHIỆM

5.1 Đánh giá độ chính xác của AI Tutor

Thử nghiệm với các bộ câu hỏi mẫu và so sánh câu trả lời của AI với nội dung bài giảng thực tế.

5.2 Đánh giá hiệu năng hệ thống

Phân tích thời gian phản hồi khi truy xuất Vector và khả năng chịu tải của API Routes.

CHƯƠNG 6

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1 Kết quả đạt được

Hệ thống đã giải quyết được vấn đề cá nhân hóa hỗ trợ học tập và giúp giáo viên tự động hóa các công việc lặp đi lặp lại.

6.2 Hướng phát triển tương lai

Tích hợp nhận diện khuôn mặt để xác thực danh tính học sinh khi thi và mở rộng ứng dụng trên nền tảng di động.

TÀI LIỆU THAM KHẢO

- [1] Vercel, "Next.js 14 Documentation," 2024. [Online]. Available: <https://nextjs.org/docs>.
- [2] Google DeepMind, "Gemini: A Family of Highly Capable Multimodal Models," 2023.
- [3] Lewis, P., et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," NeurIPS, 2020.