# CHƯƠNG 1. GIỚI THIỆU

## Lý do chọn đề tài

Trong bối cảnh công nghệ số phát triển mạnh mẽ và sự chuyển đổi số trong giáo dục ngày càng trở nên cấp thiết, việc học tập trực tuyến qua nền tảng web đã trở thành xu hướng tất yếu, đặc biệt sau những biến động toàn cầu như đại dịch COVID-19. Đến năm 2025, với sự phổ biến của internet và các thiết bị thông minh, eLearning không chỉ là giải pháp thay thế mà đã trở thành phương thức học tập chính thức, được ứng dụng rộng rãi trong các trường học, doanh nghiệp và cả đối với cá nhân. Tuy nhiên, nhiều hệ thống eLearning hiện tại vẫn còn những hạn chế đáng kể, từ việc thiếu tính tương tác và cá nhân hóa trong việc đáp ứng nhu cầu riêng biệt của học viên, đến những khó khăn trong quản lý nội dung, đánh giá tiến độ hay tương tác hiệu quả với lớp học quy mô lớn. Thêm vào đó, trải nghiệm người dùng trên một số hệ thống vẫn chưa được tối ưu, đặc biệt với những người chưa quen thuộc với công nghệ, khiến giao diện và tính năng trở nên phức tạp.

Chính vì những thách thức này, việc phát triển một hệ thống hỗ trợ học tập eLearning trên nền tảng web trở nên cần thiết nhằm đáp ứng nhu cầu học tập linh hoạt ở mọi lúc, mọi nơi, phục vụ cả học viên tự do tham gia các khóa học thị trường và sinh viên trong các lớp học thuật. Đồng thời, nó hỗ trợ giảng viên trong việc quản lý nội dung học tập, bài kiểm tra, lớp học thuật và theo dõi tiến độ học viên một cách trực quan, dễ dàng. Việc tận dụng các công nghệ hiện đại như React và NestJS giúp xây dựng giao diện thân thiện, hiệu suất cao, trong khi tích hợp các dịch vụ như MySQL, Nodemailer và Web API đảm bảo tính năng đa dạng và vận hành mượt mà. Hơn thế nữa, đề tài này góp phần thúc đẩy quá trình chuyển đổi số trong giáo dục, phù hợp với xu hướng năm 2025 khi các tổ chức giáo dục và doanh nghiệp ngày càng chú trọng đến đào tạo trực tuyến.

Đề tài không chỉ mang ý nghĩa thực tiễn mà còn có giá trị học thuật cao, cho phép khám phá và áp dụng các công nghệ trí tuệ nhân tạo, quản lý dữ liệu và phát triển phần mềm vào một hệ thống thực tế, đồng thời đóng góp một giải pháp hiệu quả, bền vững cho lĩnh vực giáo dục trực tuyến tại Việt Nam và trên toàn cầu.

## 1.2 Đặt vấn đề

Trong bối cảnh công nghệ số phát triển mạnh mẽ, nhu cầu học tập trực tuyến ngày càng gia tăng, đặc biệt trong lĩnh vực giáo dục chính quy và học tập suốt đời. Các nền tảng eLearning không chỉ cung cấp kiến thức học thuật mà còn hỗ trợ phát triển kỹ năng mềm, kỹ năng chuyên môn, và học tập linh hoạt cho nhiều đối tượng, từ học sinh, sinh viên đến người đi làm. Tuy nhiên, việc quản lý và triển khai một hệ thống eLearning hiệu quả đang đối mặt với nhiều thách thức, bao gồm sự thiếu tích hợp giữa các mô hình học tập (học thuật và thương mại), khó khăn trong đảm bảo chất lượng khóa học, và hạn chế trong việc cá nhân hóa trải nghiệm học tập.

Các phương pháp quản lý học tập truyền thống, như sử dụng giấy tờ hoặc các phần mềm đơn lẻ, thường không đáp ứng được yêu cầu về tính linh hoạt, khả năng mở rộng, và hiệu quả trong việc quản lý khóa học, học viên, và giảng viên. Điều này dẫn đến sai sót trong theo dõi tiến độ học tập, thiếu thống nhất trong đánh giá chất lượng, và giảm sự hài lòng của người học. Đặc biệt, sự kết hợp giữa mô hình Learning Management System (LMS) – phục vụ giáo dục chính quy, và Market Course – phục vụ học tập tự do, đòi hỏi một hệ thống tích hợp hiện đại, có khả năng đáp ứng đồng thời cả hai nhu cầu.

Chính vì vậy, việc nghiên cứu và phát triển một hệ thống hỗ trợ học tập eLearning trên nền tảng web theo mô hình Hybrid là một nhu cầu cấp thiết. Hệ thống này không chỉ giúp quản lý hiệu quả các khóa học, bài kiểm tra, và tương tác giữa học viên và giảng viên, mà còn hỗ trợ thương mại hóa giáo dục, mang lại sự linh hoạt và chất lượng cao cho người học. Với mục tiêu ứng dụng công nghệ thông tin để nâng cao hiệu quả quản lý và học tập, tôi quyết định chọn đề tài “Hệ thống hỗ trợ học tập eLearning trên nền tảng web” làm đề tài cho khóa luận tốt nghiệp đại học ngành Công nghệ thông tin tại trường Đại học Nam Cần Thơ. Đề tài nhằm xây dựng một hệ thống tích hợp, đáp ứng nhu cầu giáo dục hiện đại và góp phần thúc đẩy chuyển đổi số trong lĩnh vực giáo dục.

## 1.3 Giải quyết vấn đề

Trong thời đại công nghệ 4.0, nhu cầu về các hệ thống học tập trực tuyến (eLearning) ngày càng tăng, đặc biệt với sự kết hợp giữa giáo dục chính quy và học tập tự do. Tuy nhiên, các phương pháp quản lý học tập truyền thống, như sử dụng giấy tờ hoặc phần mềm đơn lẻ, đã bộc lộ nhiều hạn chế, bao gồm thiếu tích hợp, khó khăn trong theo dõi tiến độ học tập, và không đáp ứng được tính linh hoạt của người học.

Sự gia tăng nhu cầu học tập trực tuyến đã thúc đẩy nhiều nhà cung cấp phần mềm phát triển các nền tảng eLearning. Tuy nhiên, không phải hệ thống nào cũng đáp ứng được yêu cầu của cả mô hình Learning Management System (LMS) – phục vụ giáo dục chính quy, và Market Course – hỗ trợ học tập thương mại hóa. Nhiều nền tảng chỉ cung cấp các chức năng cơ bản, thiếu các tính năng chuyên biệt như quản lý bài kiểm tra, tương tác học viên-giảng viên, hoặc tích hợp thanh toán học phí, gây khó khăn cho các tổ chức giáo dục và học viên trong việc lựa chọn giải pháp phù hợp.

Một hệ thống eLearning hiệu quả theo mô hình Hybrid cần tích hợp các chức năng chuyên biệt để đáp ứng nhu cầu đa dạng của người học và giảng viên. Cụ thể, hệ thống cần có khả năng:

* Quản lý thông tin học viên và tiến độ học tập: Lưu trữ hồ sơ học viên, điểm số, và tiến độ khóa học, hỗ trợ tra cứu nhanh chóng để cá nhân hóa trải nghiệm học tập trong LMS và Market Course.
* Tổ chức khóa học và bài kiểm tra: Cho phép giảng viên tạo khóa học, phân loại nội dung theo danh mục, và thiết kế bài kiểm tra định kỳ, đảm bảo chất lượng học tập và cấp chứng chỉ cho học viên.
* Tăng cường hiệu quả quản lý và tương tác: Rút ngắn quy trình quản lý khóa học, cung cấp diễn đàn thảo luận, nhắn tin nội bộ, và thông báo tự động để tăng cường giao tiếp giữa học viên và giảng viên.
* Quản lý tài chính minh bạch: Hỗ trợ thanh toán học phí trực tuyến, thống kê doanh thu từ Market Course, đảm bảo tính minh bạch và thuận tiện cho học viên tự do.
* Tích hợp công nghệ hiện đại: Sử dụng AI để đề xuất khóa học, chatbot hỗ trợ học viên, và hệ thống thông báo qua email để cập nhật thông tin kịp thời về lịch học hoặc hạn nộp bài.

Những lợi ích này cho thấy tầm quan trọng của việc phát triển một hệ thống hỗ trợ học tập eLearning trên nền tảng web theo mô hình Hybrid. Hệ thống không chỉ nâng cao hiệu quả quản lý học tập mà còn đáp ứng nhu cầu ngày càng cao của học viên, giảng viên, và các tổ chức giáo dục, góp phần thúc đẩy chuyển đổi số trong lĩnh vực giáo dục.

## 1.4 Những đóng góp chính của đề tài

Trong quá trình thực hiện đề tài **"Hệ thống hỗ trợ học tập eLearning trên nền tảng web"**, những đóng góp chính bao gồm:

**Xây dựng hệ thống eLearning tích hợp**:

* + Phát triển một hệ thống theo mô hình Hybrid, đáp ứng đồng thời nhu cầu giáo dục chính quy (LMS) và học tập tự do (Market Course).
  + Hỗ trợ quản lý học viên/sinh viên, khóa học, bài kiểm tra, tài chính, và tương tác một cách hiệu quả.

**Tăng cường hiệu quả quản lý và học tập**:

* + Giúp giảng viên dễ dàng tạo và quản lý khóa học, theo dõi tiến độ học viên/sinh viên, và cung cấp phản hồi qua bài kiểm tra.
  + Cho phép học viên/sinh viên truy cập nội dung học tập, theo dõi tiến độ, và hoàn thành khóa học để nhận chứng chỉ.

**Tích hợp các tính năng hiện đại**:

* + Cung cấp chức năng gửi mail thông báo tự động về lịch học, kết quả kiểm tra, hoặc cập nhật khóa học.
  + Tích hợp thanh toán trực tuyến qua các cổng thanh toán an toàn, hỗ trợ học viên tự do mua khóa học dễ dàng.

**Giao diện và trải nghiệm người dùng**:

* + Xây dựng giao diện web thân thiện, trực quan với ReactJS, đảm bảo dễ sử dụng cho học viên/sinh viên, giảng viên, và quản trị viên.
  + Đảm bảo hệ thống hoạt động ổn định, bảo mật, và có khả năng mở rộng với NestJS và MSSQL.

## 1.5 Bố cục luận văn

Bố cục quyển báo cáo luận văn gồm có 6 chương chính như sau:

Chương 1: Giới thiệu

* + Đặt vấn đề
  + Lịch sử giải quyết vấn đề
  + Mục tiêu nghiên cứu
  + Phạm vi và đối tượng nghiên cứu
  + Nội dung nghiên cứu
  + Những đóng góp chính của đề tài

Chương 2: Cơ sở lý thuyết

* + Hệ quản trị cơ sở dữ liệu MSSQL
  + TypeScript
  + ReactJS
  + NestJS
  + MateriaUI
  + TypeORM
  + JSON Web Token (JWT)
  + Nodemailer
  + Web API
  + Retrieval-Augmented Generation (RAG)

Chương 3: Phân tích hệ thống

* + Mô tả hệ thống
  + Mô hình Retrieval-Augmented Generation (RAG)
  + Mô hình phân cấp chức năng BFD
  + Mô hình ngữ cảnh hệ thống
  + Biểu đồ use-case tác nhân
  + Sơ đồ tuần tự thức năng

Chương 4: Thiết kế dữ liệu

* + Xác định thực thể
  + Sơ đồ ERD
  + Mô hình vật lý PDM
  + Sơ đồ chuẩn hóa dữ liệu
  + Phát hiện các ràng buộc dữ liệu

Chương 5: Kết quả đề tài

Chương 6: Tổng kết và đánh giá

# CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

## 2.1 Mục tiêu nghiên cứu

### 2.1.1 Mục tiêu chung

* Nghiên cứu công tác quản lý hệ thống học tập trực tuyến theo mô hình Hybrid (kết hợp LMS và Market Course).
* Thiết kế và phát triển một **hệ thống hỗ trợ học tập eLearning trên nền tảng web** nhằm tối ưu hóa quá trình quản lý khóa học, làm bài kiểm tra, điểm số, học viên/sinh viên, giảng viên.
* Hệ thống hướng đến việc nâng cao hiệu quả học tập, đảm bảo chất lượng khóa học, và hỗ trợ cả giáo dục chính quy lẫn học tập tự do.

### 2.1.2 Mục tiêu cụ thể

Để đạt được mục tiêu chung, nghiên cứu cần thực hiện các mục tiêu cụ thể sau:

**- Xây dựng cơ sở dữ liệu hệ thống eLearning**:

* + Thiết kế cơ sở dữ liệu để lưu trữ thông tin về học viên/sinh viên, giảng viên, khóa học, bài kiểm tra, và tài liệu học tập.
  + Đảm bảo tính bảo mật và an toàn dữ liệu, bảo vệ thông tin cá nhân và kết quả học tập của học viên/sinh viên.

**- Phát triển hệ thống quản lý học viên/sinh viên**:

* + Cung cấp chức năng đăng ký, đăng nhập, cập nhật hồ sơ, và tra cứu thông tin học viên/sinh viên một cách nhanh chóng.
  + Theo dõi tiến độ học tập, điểm số, và chứng nhận hoàn thành khóa học của học viên/sinh viên.

**- Quản lý khóa học và giảng viên**:

* + Tổ chức khóa học theo danh mục, phân công giảng viên, và quản lý nội dung học tập (video, PDF, bài tập, trắc nghiệm, v.v).
  + Hỗ trợ giảng viên tạo bài kiểm tra, theo dõi tiến độ học viên/sinh viên, và gửi thông báo.

**- Tích hợp AI**

* + Sử dụng AI chatbot để hỗ trợ người dùng trong hệ thống, nhằm tăng trãi nghiệm người dùng, chatbot luôn phản hồi các câu hỏi từ người dùng nhanh chống và chính xác
  + Đưa ra gợi ý các khóa học dựa theo sở thích hoặc nhu cầu học tập của bản thân thông qua nhắn tin trực tiếp với chatbot, chatbot sẽ gợi ý phù hợp với người dùng

**- Quản lý tài chính và thanh toán**:

* + Xây dựng quy trình thanh toán học phí trực tuyến, quản lý doanh thu từ Market Course, và lập báo cáo tài chính chi tiết.
  + Đảm bảo tính minh bạch và thuận tiện trong các giao dịch tài chính cho học viên/sinh viên.

**- Phát triển giao diện người dùng thân thiện**:

* + Thiết kế giao diện web trực quan, dễ sử dụng cho học viên/sinh viên, giảng viên, và quản trị viên.
  + Tích hợp các tính năng như bảng điều khiển cá nhân, diễn đàn thảo luận, và nhắn tin nội bộ.

**- Kiểm thử và triển khai hệ thống**:

* + Thực hiện kiểm thử toàn diện để đảm bảo hệ thống hoạt động ổn định, đáp ứng các yêu cầu của mô hình Hybrid.
  + Triển khai hệ thống trên nền tảng web với khả năng mở rộng.

**- Đánh giá và cải tiến hệ thống**:

* + Thu thập phản hồi từ học viên/sinh viên, giảng viên, và quản trị viên để đánh giá hiệu quả của hệ thống.
  + Đề xuất và thực hiện các cải tiến nhằm nâng cao trải nghiệm người dùng và chất lượng học tập.

## 2.2 Phạm vị và đối tương nghiên cứu

### 2.2.1 Phạm vi đối tượng:

* + - * Học viên/sinh viên:
      * Sinh viên tham gia các khóa học chính quy trong mô hình LMS (Learning Management System), thuộc các lớp học do giảng viên quản lý.
      * Học viên tự do tham gia các khóa học thương mại trên nền tảng Market Course, học theo tiến độ cá nhân.
      * Giảng viên: Những người phụ trách tạo và quản lý khóa học, bài kiểm tra, và theo dõi tiến độ học tập của học viên/sinh viên trong cả LMS và Market Course.
      * Quản trị viên: Nhân sự quản lý hệ thống, chịu trách nhiệm duy trì hoạt động, quản lý tài khoản người dùng, và giám sát doanh thu từ Market Course.

### 2.2.2 Phạm vi chức năng:

Hệ thống eLearning trên nền tảng web sẽ bao gồm các chức năng chính sau:

* **Quản lý học viên/sinh viên**: Đăng ký, đăng nhập, cập nhật hồ sơ, theo dõi tiến độ học tập, điểm số, và chứng nhận hoàn thành khóa học.
* **Quản lý khóa học**: Tạo, tổ chức, và phân loại khóa học theo danh mục; quản lý nội dung học tập (video, PDF, bài tập, trắc nghiệm, v.v).
* **Quản lý giảng viên**: Lưu trữ thông tin giảng viên, hỗ trợ tạo khóa học, theo dõi tiến độ học viên/sinh viên, và gửi thông báo.
* **Quản lý bài kiểm tra và đánh giá**: Tạo bài kiểm tra trắc nghiệm/tự luận, chấm điểm tự động, lưu trữ điểm, và cung cấp phản hồi.
* **Quản lý tài chính**: Xử lý thanh toán học phí, thống kê doanh thu từ Market Course, và quản lý hóa đơn.
* **Tích hợp thanh toán**: Hỗ trợ thanh toán học phí qua các cổng thanh toán trực tuyến (thẻ tín dụng, ví điện tử, chuyển khoản ngân hàng) với giao dịch an toàn và minh bạch.
* **Gửi mail thông báo**: Tự động gửi email thông báo về lịch học, hạn nộp bài, kết quả kiểm tra, hoặc cập nhật khóa học đến học viên/sinh viên và giảng viên.
* **Tương tác và cộng đồng**: Tích hợp diễn đàn thảo luận, nhắn tin nội bộ, và hệ thống thông báo qua email hoặc ứng dụng.
* **Báo cáo và thống kê**: Cung cấp báo cáo về tiến độ học tập, hiệu suất giảng dạy, doanh thu.
* **Tích hợp AI**: Sử dụng AI để chatbot hỗ trợ người dùng hệ thống

### 2.2.3 Phạm vi thời gian:

Nghiên cứu và phát triển hệ thống trong khoảng thời gian **03 tháng**, bao gồm các giai đoạn:

* + - * **Phân tích yêu cầu**: Xác định các chức năng cần thiết cho mô hình Hybrid.
      * **Thiết kế**: Xây dựng kiến trúc hệ thống và cơ sở dữ liệu.
      * **Phát triển**: Viết mã nguồn và tích hợp các chức năng.
      * **Kiểm thử**: Kiểm tra toàn diện để đảm bảo hệ thống ổn định.
      * **Triển khai và đánh giá**: Đưa hệ thống vào sử dụng và thu thập phản hồi để cải tiến.

### 2.2.4 Phạm vi kỹ thuật:

* Sử dụng các công nghệ hiện đại:
  + **Frontend**: ReactJS để xây dựng giao diện người dùng động, trực quan, và thân thiện.
  + **Backend**: NestJS để phát triển API mạnh mẽ, có khả năng mở rộng và quản lý logic nghiệp vụ.
  + **Cơ sở dữ liệu**: MSSQL để lưu trữ và quản lý thông tin học viên/sinh viên, khóa học, bài kiểm tra, và tài chính.
* Đảm bảo:
  + **Tính bảo mật**: Áp dụng các biện pháp mã hóa và xác thực để bảo vệ dữ liệu người dùng.
  + **Hiệu năng**: Tối ưu hóa truy vấn cơ sở dữ liệu và xử lý yêu cầu để phục vụ số lượng lớn học viên/sinh viên.
  + **Khả năng mở rộng**: Thiết kế kiến trúc linh hoạt để hỗ trợ thêm người dùng và chức năng trong tương lai.

## 2.3 Nội dung nghiên cứu

Nghiên cứu tập trung vào việc thiết kế và phát triển hệ thống eLearning trên nền tảng web theo mô hình Hybrid, kết hợp Learning Management System (LMS) và Market Course. Đề tài được mô tả thông qua các sơ đồ như ERD (Entity-Relationship Diagram) và DFD (Data Flow Diagram) các cấp để minh họa kiến trúc và luồng hoạt động của hệ thống.

**Về lý thuyết**

* Phân tích và thiết kế hệ thống:
* Tìm hiểu và xây dựng sơ đồ ERD để mô hình hóa cơ sở dữ liệu, xác định các thực thể như học viên/sinh viên, giảng viên, quản trị viên, khóa học, bài kiểm tra, bài trắc nghiệm, v.v.
* Thiết kế DFD các cấp để mô tả luồng dữ liệu và quy trình xử lý trong hệ thống, bao gồm quản lý khóa học, thanh toán, gửi thông báo, và truy xuất thông tin bằng RAG.
* Nghiên cứu công nghệ:
* **Tìm hiểu các công nghệ lập trình web:** ReactJS cho giao diện người dùng, NestJS cho backend, và MSSQL cho hệ quản trị cơ sở dữ liệu.
* **ReactJS**: Sử dụng các thư viện như Axios (gửi yêu cầu HTTP), React Router (quản lý định tuyến).
* **NestJS**: Tích hợp TypeORM cho truy vấn MSSQL, JWT (JSON Web Token) và Passport cho xác thực người dùng.
* **MSSQL**: Thiết kế cơ sở dữ liệu hiệu quả với các bảng liên quan đến học viên/sinh viên, khóa học, và giao dịch.
* **RAG (Retrieval-Augmented Generation)**: Nghiên cứu kỹ thuật AI kết hợp truy xuất thông tin và tạo văn bản, sử dụng các công cụ như vector database (Pinecone, Weaviate) và mô hình ngôn ngữ lớn (LLM) để hỗ trợ chatbot thông minh, đề xuất khóa học, và trả lời câu hỏi của học viên/sinh viên.

**Về thực tế**

Hệ thống được phát triển dựa trên quy trình hoạt động thực tế của các nền tảng eLearning:

* Tìm hiểu quy trình quản lý khóa học, bài kiểm tra, và tiến độ học tập trong môi trường LMS và Market Course.
* Phân tích nhu cầu của học viên/sinh viên, giảng viên, và quản trị viên để đảm bảo hệ thống đáp ứng đầy đủ các yêu cầu.
* Nghiên cứu cách tích hợp các chức năng như gửi mail thông báo (qua email) và thanh toán trực tuyến (qua cổng thanh toán zalopay), và chatbot thông minh sử dụng RAG để hỗ trợ học viên/sinh viên.

## 2.4 Hệ quản trị cơ sở dữ liệu my sql

**Định nghĩa**

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) mã nguồn mở, được sử dụng rộng rãi trong các ứng dụng web và doanh nghiệp. MySQL hỗ trợ xử lý dữ liệu hiệu quả, quản lý các giao dịch, và cung cấp khả năng truy vấn nhanh chóng thông qua ngôn ngữ SQL (Structured Query Language). Với tính linh hoạt và hiệu suất cao, MySQL là lựa chọn phù hợp cho hệ thống eLearning, giúp lưu trữ và quản lý thông tin về học viên/sinh viên, khóa học, bài kiểm tra, và giao dịch tài chính.

**Tính năng quản lý**

MySQL Workbench: Công cụ quản trị trực quan giúp thiết kế cơ sở dữ liệu, tạo sơ đồ ERD, và quản lý dữ liệu hiệu quả, hỗ trợ phát triển hệ thống eLearning.

Tích hợp với các công cụ phát triển: MySQL dễ dàng tích hợp với NestJS (thông qua TypeORM hoặc Sequelize) và ReactJS, tạo điều kiện thuận lợi cho việc xây dựng API và giao diện người dùng.

## 2.5 Ngôn ngữ lập trình typescript

**Định nghĩa**

TypeScript là một ngôn ngữ lập trình mã nguồn mở, được phát triển bởi Microsoft vào năm 2012. TypeScript là một siêu tập (superset) của JavaScript, bổ sung hệ thống kiểu tĩnh (static typing) và các tính năng lập trình hướng đối tượng nâng cao. TypeScript được biên dịch thành JavaScript để chạy trên mọi trình duyệt hoặc môi trường Node.js, khiến nó trở thành lựa chọn lý tưởng cho việc phát triển các ứng dụng web hiện đại như hệ thống eLearning trên nền tảng web.

Trong hệ thống eLearning, TypeScript được sử dụng ở cả frontend (với ReactJS) và backend (với NestJS) để xây dựng giao diện người dùng động và API mạnh mẽ, hỗ trợ quản lý học viên/sinh viên, khóa học, bài kiểm tra, gửi mail thông báo, và tích hợp thanh toán.

**Đặc trưng của ngôn ngữ lập trình TypeScript**

**TypeScript là ngôn ngữ đơn giản và dễ học**

TypeScript kế thừa cú pháp của JavaScript, giúp các lập trình viên quen thuộc với JavaScript dễ dàng làm việc với TypeScript.

Hệ thống kiểu tĩnh (static typing) giúp phát hiện lỗi sớm trong quá trình phát triển, giảm thiểu sự phức tạp so với các ngôn ngữ như C++ hoặc Java.

**TypeScript là ngôn ngữ hiện đại**

Hỗ trợ các tính năng hiện đại như xử lý bất đồng bộ (async/await), module, và arrow function, phù hợp với việc phát triển ứng dụng web hiệu suất cao.

Tích hợp các công cụ như IntelliSense trong các IDE (VS Code), cải thiện trải nghiệm lập trình và tăng năng suất.

**TypeScript là ngôn ngữ hướng đối tượng**

TypeScript hỗ trợ đầy đủ các tính chất của lập trình hướng đối tượng (OOP): tính trừu tượng (abstraction), đóng gói (encapsulation), đa hình (polymorphism), và kế thừa (inheritance)

Các tính năng như interface, class, và generic giúp xây dựng mã nguồn có cấu trúc rõ ràng, dễ bảo trì trong hệ thống eLearning.

**TypeScript có hệ thống kiểu tĩnh mạnh mẽ**

TypeScript cho phép định nghĩa kiểu dữ liệu (types) cho biến, tham số, và giá trị trả về, giúp phát hiện lỗi cú pháp và logic trước khi chạy chương trình.

Hỗ trợ các kiểu phức tạp như union types, intersection types, và type inference, tăng tính an toàn và độ tin cậy của mã nguồn.

**TypeScript tích hợp tốt với các framework**

Trong hệ thống eLearning, TypeScript được sử dụng với ReactJS để xây dựng giao diện người dùng (ví dụ: bảng điều khiển học viên/sinh viên, form thanh toán) và với NestJS để phát triển API (ví dụ: quản lý khóa học, gửi mail thông báo).

Tích hợp với các thư viện như TypeORM (cho MySQL), Nodemailer (gửi mail), và Stripe SDK (thanh toán), giúp phát triển hệ thống nhanh chóng và hiệu quả.

**Vai trò trong hệ thống eLearning**

TypeScript đóng vai trò quan trọng trong việc phát triển hệ thống eLearning theo mô hình Hybrid:

Frontend (ReactJS): Xây dựng giao diện người dùng thân thiện, cho phép học viên/sinh viên đăng ký khóa học, theo dõi tiến độ học tập, thực hiện thanh toán, và nhận thông báo.

Backend (NestJS): Phát triển API RESTful để quản lý học viên/sinh viên, khóa học, bài kiểm tra, giao dịch tài chính, và gửi mail thông báo tự động.

Bảo trì và mở rộng: Hệ thống kiểu tĩnh và tính năng OOP của TypeScript giúp mã nguồn dễ đọc, dễ bảo trì, và dễ mở rộng khi thêm chức năng mới (ví dụ: tích hợp AI hoặc báo cáo nâng cao).

Hiệu suất và bảo mật: TypeScript đảm bảo mã nguồn đáng tin cậy, giảm lỗi runtime, và hỗ trợ các cơ chế xác thực (JWT) để bảo vệ dữ liệu học viên/sinh viên.

## 2.6 ReactJS

**Định nghĩa**

ReactJS là một thư viện JavaScript mã nguồn mở, phát triển bởi Facebook, dùng để xây dựng giao diện người dùng (UI) động, tương tác, và hiệu suất cao. ReactJS sử dụng mô hình component-based, cho phép tái sử dụng và quản lý giao diện một cách hiệu quả.

**Lịch sử phát triển**

* **2011**: Facebook phát triển ReactJS để giải quyết vấn đề hiệu suất giao diện trên nền tảng của họ.
* **2013**: ReactJS được công bố mã nguồn mở tại JSConf US.
* **Hiện nay**: React 18 (2025) hỗ trợ các tính năng như Concurrent Rendering, Server Components, và cải thiện hiệu suất.

**Đặc trưng**

* **Component-based**: Chia **Virtual DOM**: Tăng tốc độ render bằng cách chỉ cập nhật các thay đổi, cải thiện trải nghiệm người dùng.
* **Tích hợp TypeScript**: Hỗ trợ định kiểu dữ liệu, giảm lỗi khi phát triển giao diện.
* **Thư viện phong phú**: Kết hợp với Material UI, Axios (gửi HTTP request), và React Router (quản lý định tuyến).
* **Cộng đồng lớn**: Có nhiều plugin và tài liệu hỗ trợ, phù hợp vớiq hệ thống eLearning.

**Ưu điểm**

* Giao diện nhanh, đáp ứng, và dễ tái sử dụng nhờ mô hình component.
* Tích hợp tốt với Material UI và TypeScript, tạo giao diện hiện đại và an toàn.
* Hỗ trợ SEO qua Server-Side Rendering (SSR) hoặc Static Site Generation (SSG).
* Cộng đồng lớn, dễ tìm tài liệu và thư viện bổ sung.

**Hạn chế**

* Yêu cầu học thêm các thư viện bổ sung (như Redux, React Router) để xây dựng ứng dụng phức tạp.
* Có thể phức tạp với các dự án lớn nếu không tổ chức mã nguồn tốt.
* Cập nhật phiên bản mới đôi khi gây ra lỗi tương thích.

**Vai trò trong hệ thống eLearning**

* Xây dựng giao diện thân thiện cho học viên/sinh viên để đăng ký khóa học, xem nội dung học tập, thực hiện thanh toán, và nhận thông báo qua email.
* Hiển thị danh mục khóa học, bảng điều khiển tiến độ, và báo cáo thống kê một cách trực quan với Material UI.
* Tích hợp với API NestJS qua Axios để gửi yêu cầu (đăng ký khóa học, nộp bài kiểm tra) và nhận phản hồi (kết quả, thông báo).

## NestJS

**Định nghĩa**

NestJS là một framework Node.js mã nguồn mở, sử dụng TypeScript để xây dựng các ứng dụng backend mạnh mẽ, có khả năng mở rộng. NestJS kết hợp các khái niệm từ lập trình hướng đối tượng, lập trình chức năng, và lập trình phản ứng (reactive programming).

**Lịch sử phát triển**

* **2017**: Kamil Myśliwiec tạo ra NestJS, lấy cảm hứng từ Angular và Spring Boot.
* **Hiện nay**: NestJS 10.x (2025) hỗ trợ các tính năng như WebSocket, gRPC, và tích hợp sâu với TypeORM, JWT, và các thư viện khác.

**Đặc trưng**

* **Kiến trúc mô-đun**: Chia ứng dụng thành các mô-đun (quản lý khóa học, thanh toán, thông báo), dễ bảo trì và mở rộng.
* **Dependency Injection (DI)**: Quản lý phụ thuộc hiệu quả, tích hợp các thư viện như TypeORM, Nodemailer, Stripe SDK.
* **RESTful API**: Hỗ trợ xây dựng API chuẩn REST với các phương thức HTTP (GET, POST, PUT, DELETE).
* **Tích hợp TypeScript**: Tận dụng kiểu tĩnh và OOP để tăng độ tin cậy mã nguồn.

**Ưu điểm**

* Kiến trúc rõ ràng, dễ mở rộng, phù hợp với hệ thống eLearning phức tạp.
* Tích hợp tốt với MySQL qua TypeORM, Nodemailer (gửi mail), và Stripe SDK (thanh toán).
* Hỗ trợ kiểm thử (unit test, e2e test) để đảm bảo chất lượng mã nguồn.
* Cộng đồng phát triển nhanh, tài liệu chi tiết.

**Hạn chế**

* Có thể phức tạp với người mới do kiến trúc mô-đun và DI.
* Yêu cầu cấu hình thêm để tối ưu hiệu suất trong môi trường sản xuất.
* Một số thư viện tích hợp cần định nghĩa kiểu TypeScript bổ sung.

**Vai trò trong hệ thống eLearning**

* Phát triển API RESTful để quản lý học viên/sinh viên, khóa học, bài kiểm tra, giao dịch tài chính, và gửi mail thông báo.
* Tích hợp với MySQL qua TypeORM để xử lý dữ liệu và với Nodemailer/Stripe SDK để hỗ trợ gửi mail và thanh toán.
* Đảm bảo backend ổn định, bảo mật (qua JWT), và xử lý nhiều yêu cầu đồng thời từ học viên/sinh viên.

## 2.8 Material UI

**Định nghĩa**

Material UI (MUI) là một thư viện giao diện người dùng mã nguồn mở, dựa trên nguyên tắc thiết kế Material Design của Google. MUI cung cấp các component giao diện sẵn có, giúp xây dựng giao diện web đẹp, đồng nhất, và đáp ứng cho hệ thống eLearning.

**Lịch sử phát triển**

* **2014**: Material UI được tạo ra để áp dụng Material Design cho các ứng dụng React.
* **Hiện nay**: MUI 5.x (2025) hỗ trợ các tính năng như theme tùy chỉnh, component hiện đại, và tích hợp tốt với TypeScript.

**Đặc trưng**

* **Component phong phú**: Cung cấp button, form, table, dialog, modal, và nhiều component khác, dễ tùy chỉnh.
* **Thiết kế đáp ứng**: Đảm bảo giao diện hoạt động tốt trên desktop, mobile, và tablet.
* **Tích hợp với ReactJS**: Hỗ trợ TypeScript và React hooks để xây dựng giao diện an toàn.
* **Tùy chỉnh theme**: Cho phép thay đổi màu sắc, font chữ, và style để phù hợp với thương hiệu.

**Ưu điểm**

* Giao diện đẹp, đồng nhất, và chuyên nghiệp nhờ Material Design.
* Dễ sử dụng, tích hợp tốt với ReactJS và TypeScript.
* Cộng đồng lớn, nhiều tài liệu và ví dụ mẫu.
* Hỗ trợ tối ưu SEO và accessibility (a11y).

**Hạn chế**

* Có thể làm tăng kích thước bundle nếu sử dụng quá nhiều component.
* Tùy chỉnh sâu có thể phức tạp nếu cần thay đổi style mặc định.
* Một số component phức tạp cần cấu hình thêm để tối ưu hiệu suất.

**Vai trò trong hệ thống eLearning**

* Tạo giao diện trực quan cho học viên/sinh viên khi đăng ký khóa học, xem nội dung học tập, thực hiện thanh toán, hoặc nhận thông báo.
* Hỗ trợ hiển thị danh mục khóa học, bảng điều khiển tiến độ, và báo cáo thống kê một cách thẩm mỹ.
* Cải thiện trải nghiệm người dùng với các component như form thanh toán, modal thông báo, và bảng điểm thi.

## 2.9 TypeORM

**Định nghĩa**

TypeORM là một thư viện ORM (Object-Relational Mapping) mã nguồn mở, hỗ trợ ánh xạ dữ liệu giữa cơ sở dữ liệu quan hệ (như MySQL) và các đối tượng trong ứng dụng. TypeORM được viết bằng TypeScript, tích hợp tốt với NestJS.

**Lịch sử phát triển**

* **2016**: TypeORM được tạo ra bởi Alexander Putilov và cộng đồng mã nguồn mở.
* **Hiện nay**: TypeORM 0.3.x (2025) hỗ trợ MySQL, PostgreSQL, SQLite, và các tính năng như migrations, query builder.

**Đặc trưng**

* **Ánh xạ đối tượng**: Chuyển đổi dữ liệu MySQL thành các entity (học viên/sinh viên, khóa học) để dễ thao tác.
* **Truy vấn dễ dàng**: Cung cấp cú pháp đơn giản cho các thao tác CRUD và truy vấn phức tạp.
* **Hỗ trợ migrations**: Tự động tạo và cập nhật cấu trúc cơ sở dữ liệu khi thay đổi entity.
* **Tích hợp với NestJS**: Dễ dàng cấu hình để quản lý dữ liệu trong hệ thống eLearning.

**Ưu điểm**

* Giảm thời gian phát triển nhờ ánh xạ tự động giữa đối tượng và bảng cơ sở dữ liệu.
* Hỗ trợ nhiều cơ sở dữ liệu, linh hoạt cho MySQL trong hệ thống này.
* Tích hợp tốt với TypeScript và NestJS, tăng độ an toàn mã nguồn.
* Hỗ trợ query builder và repository pattern để xử lý truy vấn phức tạp.

**Hạn chế**

* Có thể chậm với các truy vấn rất phức tạp nếu không tối ưu hóa.
* Một số tính năng nâng cao (như caching) cần cấu hình bổ sung.
* Tài liệu đôi khi thiếu chi tiết cho các trường hợp đặc biệt.

**Vai trò trong hệ thống eLearning**

* Quản lý dữ liệu MySQL cho các thực thể như học viên/sinh viên, khóa học, bài kiểm tra, giao dịch thanh toán, và thông báo.
* Hỗ trợ truy vấn dữ liệu để tạo báo cáo (tiến độ học tập, doanh thu Market Course).
* Đảm bảo tính toàn vẹn dữ liệu khi xử lý giao dịch tài chính và lưu trữ lịch sử thông báo.

## 2.10 Json Web Token (JWT)

**Định nghĩa**

JSON Web Token (JWT) là một tiêu chuẩn mở để tạo token xác thực, được sử dụng để xác minh danh tính người dùng và bảo vệ các API. JWT gồm ba phần: Header, Payload, và Signature, được mã hóa dưới dạng chuỗi Base64.

**Lịch sử phát triển**

* **2010**: JWT được giới thiệu như một phần của tiêu chuẩn RFC 7519.
* **Hiện nay**: JWT được sử dụng rộng rãi trong các hệ thống web, đặc biệt với các framework như NestJS.

**Đặc trưng**

* **Xác thực an toàn**: JWT xác minh danh tính học viên/sinh viên, giảng viên, và quản trị viên.
* **Không trạng thái**: Token lưu trữ ở client, giảm tải cho server.
* **Tích hợp với NestJS**: Sử dụng module Passport và JWT để triển khai xác thực.
* **Hỗ trợ phân quyền**: Lưu trữ thông tin vai trò (role) trong Payload để giới hạn chức năng.

**Ưu điểm**

* Bảo mật cao, khó giả mạo nhờ chữ ký số (Signature).
* Không yêu cầu lưu trữ session trên server, phù hợp với hệ thống phân tán.
* Dễ tích hợp với NestJS và các thư viện xác thực như Passport.
* Hỗ trợ xác thực đa nền tảng (web, mobile).

**Hạn chế**

* Token bị đánh cắp có thể gây rủi ro nếu không có cơ chế thu hồi.
* Kích thước token lớn hơn so với session truyền thống, ảnh hưởng đến hiệu suất nếu Payload chứa nhiều dữ liệu.
* Quản lý refresh token phức tạp trong các hệ thống lớn.

**Vai trò trong hệ thống eLearning**

* Xác thực người dùng khi đăng nhập, đăng ký khóa học, hoặc thực hiện thanh toán.
* Bảo vệ các API (quản lý khóa học, gửi mail thông báo, thanh toán) khỏi truy cập trái phép.
* Hỗ trợ phân quyền để giới hạn chức năng theo vai trò (học viên/sinh viên chỉ xem khóa học, giảng viên tạo bài kiểm tra).

## 2.11 NodeMailer

**Định nghĩa**

Nodemailer là một thư viện Node.js mã nguồn mở, dùng để gửi email từ ứng dụng. Nodemailer hỗ trợ gửi email qua các dịch vụ SMTP (Gmail, SendGrid, Outlook) với cấu hình đơn giản và hiệu quả.

**Lịch sử phát triển**

* **2010**: Andris Reinman tạo ra Nodemailer để đơn giản hóa việc gửi email trong Node.js.
* **Hiện nay**: Nodemailer 6.x (2025) hỗ trợ các tính năng như HTML email, tệp đính kèm, và tích hợp với NestJS.

**Đặc trưng**

* **Dễ sử dụng**: Cung cấp API đơn giản để gửi email với văn bản, HTML, hoặc tệp đính kèm.
* **Hỗ trợ nhiều nhà cung cấp**: Tích hợp với Gmail, Outlook, SendGrid, và SMTP tùy chỉnh.
* **Tích hợp với NestJS**: Dễ dàng cấu hình trong NestJS để gửi mail thông báo tự động.
* **Bảo mật**: Hỗ trợ SSL/TLS để gửi email an toàn.

**Ưu điểm**

* Cấu hình đơn giản, dễ tích hợp với các dịch vụ email phổ biến.
* Hỗ trợ gửi email hàng loạt và tùy chỉnh nội dung (HTML, template).
* Miễn phí, mã nguồn mở, và có cộng đồng hỗ trợ lớn.
* Tích hợp tốt với MySQL để lưu trữ lịch sử email.

**Hạn chế**

* Phụ thuộc vào dịch vụ SMTP, có thể gặp hạn chế với các nhà cung cấp miễn phí (như Gmail giới hạn số lượng email).
* Yêu cầu cấu hình bảo mật (OAuth2, API key) cho các dịch vụ lớn.
* Xử lý email hàng loạt có thể chậm nếu không tối ưu hóa.

**Vai trò trong hệ thống eLearning**

* Gửi mail thông báo tự động đến học viên/sinh viên về lịch học, hạn nộp bài, kết quả kiểm tra, hoặc cập nhật khóa học.
* Thông báo cho giảng viên về tiến độ học viên/sinh viên hoặc các sự kiện quan trọng.
* Lưu trữ lịch sử email trong MySQL (qua TypeORM) để theo dõi giao tiếp.

## 2.12 Web API

**Định nghĩa**

Web API là một giao diện lập trình ứng dụng (API) sử dụng giao thức HTTP để cung cấp dữ liệu hoặc chức năng giữa client (frontend) và server (backend). Trong hệ thống eLearning, Web API được xây dựng với NestJS theo chuẩn RESTful.

**Lịch sử phát triển**

* **2000s**: RESTful API trở nên phổ biến với sự phát triển của web 2.0.
* **Hiện nay**: Web API là tiêu chuẩn trong các ứng dụng web, đặc biệt với các framework như NestJS.

**Đặc trưng**

* **RESTful**: Sử dụng các phương thức HTTP (GET, POST, PUT, DELETE) để thực hiện thao tác CRUD.
* **Tích hợp với frontend**: Kết nối với ReactJS để gửi/nhận dữ liệu (danh sách khóa học, kết quả bài kiểm tra).
* **Bảo mật**: Kết hợp với JWT để xác thực và bảo vệ các endpoint API.
* **Hiệu suất cao**: NestJS tối ưu hóa xử lý yêu cầu, đảm bảo phản hồi nhanh.

**Ưu điểm**

* Dễ thiết kế, triển khai, và kiểm thử với các công cụ như Postman hoặc Swagger.
* Tích hợp tốt với ReactJS và MySQL, hỗ trợ tất cả chức năng của hệ thống.
* Có khả năng mở rộng để thêm endpoint mới (ví dụ: API cho AI đề xuất khóa học).
* Hỗ trợ chuẩn REST, dễ dàng tích hợp với các ứng dụng bên thứ ba.

**Hạn chế**

* Yêu cầu bảo mật chặt chẽ để tránh các cuộc tấn công như SQL injection hoặc XSS.
* Quản lý API phức tạp trong các hệ thống lớn với nhiều endpoint.
* Hiệu suất có thể giảm nếu không tối ưu hóa truy vấn cơ sở dữ liệu.

**Vai trò trong hệ thống eLearning**

* Cung cấp các endpoint API để quản lý học viên/sinh viên, khóa học, bài kiểm tra, giao dịch thanh toán, và gửi mail thông báo.
* Hỗ trợ tích hợp thanh toán qua cổng thanh toán (Stripe) và gửi email qua Nodemailer.
* Đảm bảo giao tiếp mượt mà giữa frontend (ReactJS) và backend (NestJS), cải thiện trải nghiệm người dùng.

## 2.13 Retrieval-Augmented Generation (RAG)

**Định nghĩa**

Retrieval-Augmented Generation (RAG) là một kỹ thuật trí tuệ nhân tạo (AI) kết hợp **truy xuất thông tin** (Retrieval) và **tạo văn bản** (Generation) để cung cấp câu trả lời chính xác, ngữ cảnh hóa dựa trên kho dữ liệu bên ngoài. Trong hệ thống eLearning, RAG sử dụng **Qdrant**, một cơ sở dữ liệu vector mã nguồn mở được viết bằng Rust, để lưu trữ và tìm kiếm các vector nhúng (vector embeddings) của tài liệu học tập, từ đó hỗ trợ chatbot thông minh, đề xuất khóa học, và trả lời câu hỏi của học viên/sinh viên. Qdrant cung cấp khả năng tìm kiếm tương tự vector (vector similarity search) hiệu quả, giúp RAG truy xuất nhanh thông tin liên quan từ cơ sở tri thức.

**Lịch sử phát triển**

* **2020**: RAG được giới thiệu bởi Facebook AI Research trong bài báo "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks". Qdrant bắt đầu phát triển như một cơ sở dữ liệu vector mã nguồn mở.
* **2021-2023**: Qdrant được tích hợp vào các ứng dụng RAG, hỗ trợ các framework như LangChain, LlamaIndex, và mô hình ngôn ngữ lớn (LLM) như GPT, LLaMA.
* **Hiện nay (2025)**: Qdrant 1.14 cung cấp các tính năng như Score Boosting Reranker, quantization, và hỗ trợ đa vector, tối ưu cho RAG trong các ứng dụng giáo dục. RAG với Qdrant được áp dụng rộng rãi trong chatbot, tìm kiếm ngữ nghĩa, và đề xuất nội dung.

**Đặc trưng**

* **Truy xuất thông tin với Qdrant**: Qdrant lưu trữ tài liệu dưới dạng vector nhúng, sử dụng các thuật toán tìm kiếm tương tự (như cosine similarity) để truy xuất tài liệu liên quan đến câu hỏi của học viên/sinh viên.
* **Tạo văn bản ngữ cảnh hóa**: Mô hình ngôn ngữ lớn (LLM) như GPT hoặc LLaMA sử dụng thông tin truy xuất từ Qdrant để tạo câu trả lời tự nhiên, giảm hiện tượng "hallucination" (tạo thông tin sai lệch).
* **Hiệu suất cao**: Qdrant đạt độ trễ thấp (3ms cho 1 triệu vector nhúng Open AI) và khả năng xử lý hàng tỷ vector, phù hợp với hệ thống eLearning có lượng truy vấn lớn.
* **Tích hợp với hệ thống**: RAG với Qdrant có thể được triển khai trong NestJS (backend) để xử lý truy xuất và ReactJS (frontend) để hiển thị câu trả lời qua giao diện Material UI.
* **Hỗ trợ đa dạng dữ liệu**: Qdrant lưu trữ vector nhúng từ văn bản, hình ảnh, hoặc âm thanh, cho phép RAG xử lý tài liệu học tập đa phương tiện.

**Ưu điểm**

* **Độ chính xác cao**: Qdrant truy xuất thông tin từ kho dữ liệu thực tế, giúp RAG cung cấp câu trả lời đáng tin cậy hơn so với LLM thuần túy.
* **Khả năng mở rộng**: Qdrant hỗ trợ sharding, multitenancy, và triển khai trên Docker hoặc Kubernetes, phù hợp với hệ thống eLearning quy mô lớn.
* **Tiết kiệm chi phí**: So với việc sử dụng LLM với cửa sổ ngữ cảnh lớn, RAG với Qdrant giảm chi phí tính toán (ước tính rẻ hơn 100 triệu lần cho mỗi truy vấn).
* **Dễ tích hợp**: Qdrant tương thích với các thư viện như LangChain, LlamaIndex, và FastAPI, cho phép tích hợp với NestJS và MySQL qua TypeORM.
* **Cập nhật linh hoạt**: Chỉ cần cập nhật kho dữ liệu trong Qdrant mà không cần huấn luyện lại LLM, phù hợp với nội dung khóa học thay đổi thường xuyên.

**Hạn chế**

* **Phụ thuộc vào kho dữ liệu**: Chất lượng câu trả lời phụ thuộc vào tài liệu trong Qdrant; nếu tài liệu thiếu hoặc không chính xác, kết quả có thể không đáp ứng yêu cầu.
* **Phức tạp triển khai**: Yêu cầu cấu hình Qdrant (Docker, vector nhúng) và tích hợp với LLM, có thể phức tạp với hệ thống nhỏ.
* **Chi phí ban đầu**: Quá trình tạo vector nhúng (indexing) đòi hỏi tài nguyên tính toán lớn, mặc dù chi phí vận hành sau đó thấp.
* **Xử lý truy vấn phức tạp**: RAG có thể gặp khó khăn với các câu hỏi không rõ ràng hoặc không có thông tin trong kho dữ liệu.

**Vai trò trong hệ thống eLearning**

RAG với Qdrant nâng cao tính thông minh và tương tác của hệ thống eLearning theo mô hình Hybrid:

* **Chatbot hỗ trợ học viên/sinh viên**: Qdrant truy xuất tài liệu khóa học từ MySQL (qua TypeORM) hoặc tài liệu PDF, sau đó LLM tạo câu trả lời chi tiết cho các câu hỏi như "Cách lập trình REST API với NestJS".
* **Đề xuất khóa học**: RAG phân tích lịch sử học tập (lưu trong MySQL) và sử dụng Qdrant để truy xuất khóa học tương tự, đề xuất các khóa học phù hợp từ Market Course, ví dụ: "Lập trình JavaScript" cho học viên hoàn thành "HTML/CSS".
* **Hỗ trợ giảng viên**: Qdrant truy xuất câu hỏi mẫu hoặc tài liệu để giảng viên tạo bài kiểm tra hoặc nội dung khóa học mới.
* **Tích hợp với các chức năng**:
  + **Gửi mail thông báo**: RAG tạo nội dung email (ví dụ: câu trả lời hoặc tài liệu khóa học) để gửi qua Nodemailer.
  + **Tích hợp thanh toán**: Qdrant cung cấp thông tin khóa học chi tiết (như mô tả, đánh giá) để hỗ trợ học viên quyết định mua khóa học qua ZaloPay.

**Tăng trải nghiệm người dùng**: Chatbot sử dụng RAG với Qdrant cung cấp câu trả lời nhanh, chính xác, hiển thị qua giao diện ReactJS/Material UI, cải thiện tương tác trong cả LMS và Market Course.