

### THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH HỌC KỲ I, NĂM HỌC 2024-2025

XÂY DỰNG WEBSITE QUẢNG BÁ TUYỂN SINH

*Giảng viên hướng dẫn:*



**ThS. Nguyễn Hoàng Duy Thiện**

*Sinh viên thực hiện:* Họ tên: **Lê Minh Nhựt** MSSV: **110121234** Lớp: **DA21TTC**



### THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH HỌC KỲ I, NĂM HỌC 2024-2025

XÂY DỰNG WEBSITE QUẢNG BÁ TUYỂN SINH

*Giảng viên hướng dẫn:*



**ThS. Nguyễn Hoàng Duy Thiện**

*Sinh viên thực hiện:* Họ tên: **Lê Minh Nhựt** MSSV: **110121234** Lớp: **DA21TTC**

### NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

*Trà Vinh, ngày ..... tháng ..... năm .....*

### Giáo viên hướng dẫn

*(Ký tên và ghi rõ họ tên)*

### NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

.............................................................................................................................

# LỜI CẢM ƠN

Trong quá trình thực hiện đề tài “Xây dựng website quảng bá tuyển sinh,” em đã nhận được sự hỗ trợ, hướng dẫn và động viên nhiệt tình từ nhiều cá nhân và tổ chức. Đây chính là nguồn động lực lớn giúp em hoàn thành tốt bài nghiên cứu của mình.

Trước tiên, em xin gửi lời cảm ơn chân thành và sâu sắc đến thầy Nguyễn Hoàng Duy Thiện, người đã tận tình hướng dẫn, định hướng và đưa ra những ý kiến quý giá giúp em hoàn thiện đề tài này.

Em cũng xin cảm ơn ban lãnh đạo đã tạo điều kiện thuận lợi, cung cấp các tài liệu và cơ sở vật chất cần thiết để em có thể thực hiện nghiên cứu một cách suôn sẻ.

Bên cạnh đó, em xin bày tỏ lòng biết ơn sâu sắc đến gia đình, bạn bè và những người thân yêu, những người luôn bên cạnh, động viên tinh thần và hỗ trợ em trong suốt quá trình thực hiện đề tài.

Cuối cùng, em xin cảm ơn tất cả những ai đã trực tiếp hoặc gián tiếp đóng góp vào sự thành công của bài nghiên cứu này. Dù còn nhiều điểm cần hoàn thiện, em hy vọng đề tài sẽ mang lại giá trị thực tiễn và đóng góp tích cực cho hoạt động tuyển sinh của nhà trường.

Em xin chân thành cảm ơn!

# MỤC LỤC

[LỜI CẢM ƠN 3](#_TOC_250083)

[DANH MỤC HÌNH ẢNH 9](#_TOC_250082)

[DANH MỤC BẢNG BIỂU 10](#_TOC_250081)

[TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH 11](#_TOC_250080)

[MỞ ĐẦU 13](#_TOC_250079)

[CHƯƠNG 1: TỔNG QUAN 15](#_TOC_250078)

* 1. [Thực trạng và bối cảnh nghiên cứu 15](#_TOC_250077)
  2. [Vấn đề cần giải quyết 15](#_TOC_250076)
  3. [Hướng tiếp cận và cách giải quyết 16](#_TOC_250075)
  4. [Ý nghĩa và đóng góp của đề tài 17](#_TOC_250074)

[CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT 18](#_TOC_250073)

* 1. [Ngôn ngữ HTML (HyperText Markup Language) 18](#_TOC_250072)
     1. [Khái niệm 18](#_TOC_250071)
     2. [Lịch sử phát triển 18](#_TOC_250070)
     3. [Điểm nhấn trong HTML5 19](#_TOC_250069)
     4. [Vai trò của HTML trong phát triển web 19](#_TOC_250068)

[2.1.5. Cấu trúc cơ bản của HTML 20](#_TOC_250067)

* + 1. [Các thành phần chính của HTML 21](#_TOC_250066)
    2. [Các loại thẻ phổ biến 22](#_TOC_250065)
    3. [Ưu điểm và hạn chế của HTML 22](#_TOC_250064)
    4. [Kết luận 23](#_TOC_250063)
  1. [Ngôn ngữ CSS (Cascading Style Sheets) 23](#_TOC_250062)
     1. [Khái niệm 23](#_TOC_250061)
     2. [Lịch sử phát triển 23](#_TOC_250060)
     3. [Vai trò của CSS trong phát triển web 24](#_TOC_250059)
     4. [Cấu trúc cơ bản của CSS 26](#_TOC_250058)
     5. [Các thành phần cơ bản trong CSS 27](#_TOC_250057)
     6. [Các mô-đun quan trọng trong CSS 28](#_TOC_250056)
     7. [Ưu điểm và hạn chế của CSS 29](#_TOC_250055)
     8. [Kết luận 29](#_TOC_250054)
  2. [Ngôn ngữ PHP (Hypertext Preprocessor) 30](#_TOC_250053)
     1. [Khái niệm 30](#_TOC_250052)
     2. [Lịch sử phát triển 30](#_TOC_250051)
     3. [Vai trò của PHP trong phát triển web 31](#_TOC_250050)
     4. [Cấu trúc cơ bản của PHP 33](#_TOC_250049)
     5. [Các tính năng đặc biệt của PHP 34](#_TOC_250048)
     6. [Ưu điểm và hạn chế của PHP 34](#_TOC_250047)
  3. [JavaScript (JS) 35](#_TOC_250046)
     1. [Khái niệm 35](#_TOC_250045)
     2. [Lịch sử phát triển 35](#_TOC_250044)
     3. [Vai trò của JavaScript trong phát triển web 36](#_TOC_250043)
     4. [Cấu trúc cơ bản của JavaScript 37](#_TOC_250042)
     5. [Ưu điểm và hạn chế của JavaScript 38](#_TOC_250041)
     6. [Kết luận 38](#_TOC_250040)
  4. [Ngôn ngữ SQL (Structured Query Language) 39](#_TOC_250039)
     1. [Khái niệm 39](#_TOC_250038)
     2. [Lịch sử phát triển 39](#_TOC_250037)
     3. [Vai trò của SQL trong quản lý cơ sở dữ liệu 39](#_TOC_250036)
     4. [Cấu trúc cơ bản của SQL 41](#_TOC_250035)
     5. [Ưu điểm và hạn chế của SQL 42](#_TOC_250034)
     6. [Ví dụ minh họa 42](#_TOC_250033)
     7. [Kết luận 43](#_TOC_250032)

[CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU 44](#_TOC_250031)

* 1. [Mô tả bài toán 44](#_TOC_250030)
  2. [Đặc tả nhu cầu 45](#_TOC_250029)
     1. [Phân tích yêu cầu chức năng 45](#_TOC_250028)
     2. [Phân tích yêu cầu phi chức năng 45](#_TOC_250027)
  3. [Phân tích và thiết kế hệ thống 45](#_TOC_250026)
     1. [Kiến trúc hệ thống 45](#_TOC_250025)
     2. [Mô hình nghiệp vụ (Use Case Diagram) 47](#_TOC_250024)
     3. [Lược đồ thực thể quan hệ (ERD - Entity Relationship Diagram) 49](#_TOC_250023)
     4. [Lược đồ tuần tự (Sequence Diagram) 49](#_TOC_250022)
  4. [Thiết kế dữ liệu 53](#_TOC_250021)
     1. [Bảng admins 53](#_TOC_250020)
     2. [Bảng news 53](#_TOC_250019)
     3. [Bảng programs 54](#_TOC_250018)
     4. [Bảng admission\_info 54](#_TOC_250017)
     5. [Bảng contact\_messages 55](#_TOC_250016)
  5. [Cài đặt và triển khai 55](#_TOC_250015)
     1. [Công nghệ sử dụng 55](#_TOC_250014)
     2. [Quy trình cài đặt 55](#_TOC_250013)

[3.5 Kết luận 59](#_TOC_250012)

[CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU 60](#_TOC_250011)

* 1. [Giới thiệu 60](#_TOC_250010)
  2. [Hiệu năng hệ thống 60](#_TOC_250009)
  3. [Các giao diện chức năng 61](#_TOC_250008)
     1. [Giao diện người dùng 61](#_TOC_250007)
     2. Giao diện Admin 64
  4. [Đánh giá chung 68](#_TOC_250006)
     1. [Thành công của hệ thống 68](#_TOC_250005)
     2. [Hạn chế và hướng cải thiện 68](#_TOC_250004)
  5. [Kết luận 69](#_TOC_250003)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 70](#_TOC_250002)

* 1. [Kết luận 70](#_TOC_250001)
  2. [Hướng phát triển 70](#_TOC_250000)

# DANH MỤC HÌNH ẢNH

*Hình 3.1. Sơ đồ Use Case* 48

*Hình 3.2 Mô hình ERD* 49

*Hình 3.3 Luồng xử lý tin nhắn tư vấn* 49

*Hình 3.4 Luồng xử lý quản lý tin tức của Admin* 50

*Hình 3.5 Luồng xử lý Quản lý ngành học của Admin* 50

*Hình 3.6 Luồng xử lý quản lý thông tin tuyển sinh* 51

*Hình 3.7 Luồng xử lý xem thông tin tuyển sinh* 51

*Hình 3.8 Luồng xử lý xem thông tin ngành học* 52

*Hình 3.9 Luồng xử lý xem tin tức* 52

*Hình 3.10 Luồng xử lý gửi tin nhắn liên hệ* 52

*Hình 3.11 Thiết kế giao diện* 56

*Hình 3.12 Tạo máy ảo trên VMware Workstation* 57

*Hình 3.13 Thiết lập máy ảo* 58

*Hình 3.14 Sao chép mã nguồn ứng dụng vào máy ảo bằng FileZilla* 58

*Hình 3.15 Nhập cơ sở dữ liệu* 59

*Hình 4.1 Giao diện trang chủ* 61

*Hình 4.2 Giao diện trang chủ* 61

*Hình 4.3 Giao diện ngành học* 62

*Hình 4.4 Giao diện xem ngành học chi tiết* 62

*Hình 4.5 Giao diện đăng ký tư vấn tuyển sinh* 63

*Hình 4.6 Giao diện tin tức* 63

*Hình 4.7 Giao diện xem tin tức chi tiết* 64

*Hình 4.8 Giao diện admin chính* 64

*Hình 4.9 Giao diện thêm tin tức* 65

*Hình 4.10 Giao diện sửa tin tức* 65

*Hình 4.11 Giao diện thêm ngành học* 66

*Hình 4.12 Giao diện sửa ngành học* 66

*Hình 4.13 Giao diện quản lý thông tin tuyển sinh* 67

*Hình 4.14 Giao diện quản lý tin nhắn liên hệ tư vấn tuyển sinh* 67

*Hình 4.15 Giao diện quản lý tài khoản Admin* 68

# DANH MỤC BẢNG BIỂU

*Bảng 3.1 Bảng admins* 53

*Bảng 3.2 Bảng news* 53

*Bảng 3.3 Bảng programs* 54

*Bảng 3.4 Bảng admission\_info* 54

*Bảng 3.5 Bảng contact\_messages* 55

# TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH

### Tóm tắt vấn đề nghiên cứu

Với sự phát triển của công nghệ và nhu cầu thông tin nhanh chóng, website tuyển sinh của các trường đại học ngày càng trở thành công cụ quan trọng giúp thu hút sinh viên và quảng bá hình ảnh. Tuy nhiên, nhiều trường đại học hiện nay vẫn chưa có website tuyển sinh hiệu quả và thân thiện với người dùng. Website cần cung cấp thông tin đầy đủ về chương trình đào tạo, quy trình tuyển sinh, học bổng và các thông tin hữu ích khác một cách dễ dàng và nhanh chóng. Do đó, vấn đề đặt ra là làm sao xây dựng một website tuyển sinh hiện đại, trực quan và tối ưu cho người dùng, đáp ứng tốt nhu cầu tra cứu của thí sinh và phụ huynh.

### Các hướng tiếp cận

* + Khảo sát và phân tích yêu cầu người dùng, nghiên cứu nhu cầu của thí sinh và phụ huynh khi tìm kiếm thông tin tuyển sinh, cũng như tham khảo các website tuyển sinh của các trường đại học trong và ngoài nước để xác định các yếu tố quan trọng cần có.
  + Phân tích công nghệ phù hợp, tìm hiểu và lựa chọn các công nghệ xây dựng website hiện đại như HTML5, CSS3, JavaScript, và các framework (ReactJS, Angular, hoặc VueJS) để tạo giao diện thân thiện, dễ sử dụng và tốc độ tải nhanh.
  + Thiết kế một giao diện hiện đại, đơn giản nhưng thu hút, đảm bảo người dùng dễ dàng tìm kiếm và tra cứu thông tin cần thiết.
  + Xây dựng tính năng tương tác như chat trực tiếp, tư vấn online, biểu mẫu đăng ký để tăng cường khả năng tương tác và hỗ trợ thí sinh nhanh chóng.

### Cách giải quyết vấn đề

* + Website được thiết kế tối ưu cho các thiết bị di động, máy tính bảng và máy tính để bàn, giúp người dùng dễ dàng truy cập trên mọi thiết bị.
  + Cập nhật nội dung thường xuyên, xây dựng hệ thống quản trị nội dung (CMS) để nhân viên có thể dễ dàng cập nhật thông tin tuyển sinh, ngành học, học phí, học bổng mà không cần kiến thức lập trình.
  + Tối ưu hóa tốc độ tải trang, bảo mật dữ liệu người dùng và duy trì tính ổn định của hệ thống.

### Một số kết quả đạt được

* + Website có giao diện dễ sử dụng, thân thiện với người dùng, tối ưu cho các thiết bị khác nhau, giúp thí sinh và phụ huynh dễ dàng tra cứu thông tin.
  + Nâng cao mức độ tương tác với các công cụ hỗ trợ tư vấn trực tuyến, biểu mẫu đăng ký và chức năng chat trực tiếp, website tăng cường sự kết nối giữa nhà trường và thí sinh.
  + Hệ thống CMS cho phép nhà trường chủ động cập nhật thông tin mà không cần sự can thiệp của kỹ thuật viên, giúp tiết kiệm thời gian và đảm bảo thông tin luôn được cập nhật.

Nhờ vậy, website tuyển sinh sẽ hỗ trợ hiệu quả quá trình quảng bá của nhà trường, đáp ứng nhu cầu thông tin của thí sinh, đồng thời nâng cao hình ảnh chuyên nghiệp và uy tín của trường trong mắt công chúng.

# MỞ ĐẦU

### Lý do chọn đề tài

Hiện nay, với sự phát triển mạnh mẽ của công nghệ thông tin, internet đã trở thành phương tiện không thể thiếu trong cuộc sống. Các trường đại học cũng cần tận dụng internet để quảng bá hình ảnh, cung cấp thông tin và thu hút sinh viên tiềm năng. Việc xây dựng một website quảng bá tuyển sinh không chỉ giúp nâng cao uy tín, tăng cường khả năng cạnh tranh của trường mà còn giúp thí sinh và phụ huynh dễ dàng tiếp cận thông tin chính xác, đầy đủ. Vì vậy, đề tài "Xây dựng website quảng bá tuyển sinh cho trường đại học" được chọn nhằm đáp ứng nhu cầu thực tế này.

### Mục đích nghiên cứu

Mục đích của đề tài là xây dựng một website hiện đại, thân thiện, dễ sử dụng và đầy đủ thông tin, giúp quảng bá hình ảnh và thông tin tuyển sinh của trường đại học. Website sẽ cung cấp các thông tin quan trọng về ngành học, chương trình đào tạo, quy trình tuyển sinh, học phí, hỗ trợ tài chính, các hoạt động ngoại khóa và cơ hội nghề nghiệp sau khi tốt nghiệp. Qua đó, website sẽ giúp trường đại học thu hút được nhiều sinh viên tiềm năng và hỗ trợ thí sinh trong quá trình tìm hiểu, đăng ký tuyển sinh.

### Đối tượng nghiên cứu

* + Đối tượng nghiên cứu của đề tài bao gồm:
  + Các thí sinh và phụ huynh có nhu cầu tìm hiểu về tuyển sinh.
  + Các website tuyển sinh của các trường đại học khác nhằm tham khảo và học hỏi.
  + Các công cụ, nền tảng và công nghệ xây dựng website hiện đại, bao gồm ngôn ngữ lập trình web, hệ quản trị nội dung và các công nghệ tối ưu hóa trải nghiệm người dùng.

### Phạm vi nghiên cứu

Phạm vi nghiên cứu của đề tài tập trung vào việc xây dựng website quảng bá tuyển sinh cho trường đại học với các tính năng chính như cung cấp thông tin tuyển

sinh, giao diện thân thiện, dễ sử dụng và tích hợp các công cụ hỗ trợ tương tác giữa nhà trường và thí sinh. Đề tài cũng tập trung vào việc tìm hiểu và áp dụng các công nghệ phát triển website phù hợp, đáp ứng yêu cầu về tốc độ, bảo mật và tính tiện ích cho người dùng.

# CHƯƠNG 1: TỔNG QUAN

### Thực trạng và bối cảnh nghiên cứu

Trong thời đại chuyển đổi số, internet đã trở thành công cụ quan trọng trong mọi lĩnh vực, bao gồm giáo dục. Đối với các trường đại học, website tuyển sinh đóng vai trò cốt lõi trong việc quảng bá thương hiệu, cung cấp thông tin và thu hút sinh viên tiềm năng. Tuy nhiên, thực tế cho thấy nhiều trường vẫn chưa khai thác hiệu quả kênh thông tin này.

* + - Hạn chế của các website tuyển sinh hiện tại**:**

+ Giao diện lỗi thời, thiếu trực quan và chưa tối ưu trên thiết bị di động.

+ Nội dung không đầy đủ hoặc không được cập nhật thường xuyên.

+ Thiếu các công cụ hỗ trợ tương tác và tư vấn trực tuyến cho thí sinh.

* Yêu cầu ngày càng cao từ người dùng**:**

+ Thí sinh và phụ huynh mong muốn truy cập thông tin nhanh chóng, đầy đủ và chính xác.

+ Website cần dễ sử dụng, thân thiện với nhiều đối tượng và tích hợp các tính năng hỗ trợ hiện đại.

* Áp lực cạnh tranh giữa các trường đại học**:**

+ Ngày càng nhiều trường đầu tư mạnh mẽ vào truyền thông số để quảng bá hình ảnh.

+ Một website tuyển sinh chuyên nghiệp, hiện đại là yếu tố quan trọng để tạo ấn tượng tốt và thu hút sinh viên.

### Vấn đề cần giải quyết

Đề tài sẽ tập trung vào việc khắc phục các hạn chế hiện tại và cung cấp giải pháp toàn diện nhằm nâng cao hiệu quả hoạt động của website tuyển sinh như sau:

* + - Cải thiện giao diện và trải nghiệm người dùng**:**

+ Thiết kế giao diện hiện đại, thân thiện, dễ sử dụng và trực quan.

+ Đảm bảo website hoạt động tốt trên mọi thiết bị (máy tính, điện thoại, máy tính bảng).

* + - Cung cấp thông tin đầy đủ, chính xác:

+ Xây dựng cấu trúc thông tin hợp lý, dễ tra cứu.

+ Tích hợp hệ thống quản trị nội dung (CMS) giúp nhà trường dễ dàng cập nhật thông tin.

* + - Tăng cường khả năng tương tác và hỗ trợ thí sinh**:**

+ Tích hợp các công cụ như tư vấn trực tuyến, chat trực tiếp, và biểu mẫu đăng ký.

+ Xây dựng chatbot hoặc hệ thống trả lời tự động để hỗ trợ thí sinh nhanh chóng.

* + - Đảm bảo website hoạt động ổn định, đặc biệt trong các mùa tuyển sinh cao điểm.

### Hướng tiếp cận và cách giải quyết

* + - Khảo sát và phân tích yêu cầu người dùng**:**

+ Nghiên cứu nhu cầu của thí sinh và phụ huynh khi truy cập website tuyển sinh.

+ Tham khảo các website tuyển sinh tiêu biểu trong và ngoài nước để học hỏi kinh nghiệm.

* + - Sử dụng các công nghệ hiện đại như HTML5, CSS3, JavaScript, ...
    - Thiết kế giao diện và tối ưu hóa trải nghiệm người dùng (UI/UX)**:**

+ Sử dụng nguyên tắc thiết kế trực quan, đảm bảo mọi đối tượng người dùng đều dễ dàng truy cập và sử dụng.

+ Tích hợp giao diện đáp ứng (responsive) để website hiển thị tốt trên mọi loại thiết bị.

* + - Phát triển các tính năng tương tác và hỗ trợ trực tuyến**:**

+ Tích hợp các công cụ tư vấn trực tiếp, biểu mẫu đăng ký và chatbot hỗ trợ tự động.

+ Xây dựng hệ thống thông báo và phản hồi nhanh qua email hoặc tin nhắn.

* + - Kiểm thử và cải tiến liên tục**:**

+ Tiến hành kiểm thử website để đảm bảo hoạt động ổn định, bảo mật và hiệu suất cao.

+ Thu thập phản hồi từ người dùng để cải tiến và hoàn thiện website.

### Ý nghĩa và đóng góp của đề tài

Việc giải quyết các vấn đề trên không chỉ mang lại giá trị thực tiễn cho trường đại học mà còn đóng góp vào việc nâng cao hiệu quả truyền thông số trong lĩnh vực giáo dục:

* + - Hỗ trợ thí sinh và phụ huynh, cung cấp thông tin nhanh chóng, chính xác, giúp họ dễ dàng đưa ra quyết định phù hợp.
    - Tăng cường quảng bá hình ảnh trường giúp trường đại học khẳng định uy tín và thu hút sinh viên tiềm năng.
    - Ứng dụng công nghệ hiện đại góp phần thúc đẩy quá trình chuyển đổi số trong giáo dục, đáp ứng xu hướng phát triển của thời đại.

Với cách tiếp cận toàn diện và giải pháp cụ thể, đề tài hứa hẹn mang lại một website tuyển sinh chất lượng, hỗ trợ hiệu quả cho nhà trường trong việc quảng bá và tuyển sinh.

# CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

### Ngôn ngữ HTML (HyperText Markup Language)

### Khái niệm

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu siêu văn bản được thiết kế để tạo cấu trúc và nội dung cho các tài liệu trên World Wide Web. Đây là một ngôn ngữ không thể thiếu trong quá trình phát triển web, đóng vai trò như nền tảng xây dựng các trang web tĩnh và hiện đại.

HTML được gọi là "ngôn ngữ đánh dấu" vì nó sử dụng các thẻ (tags) để định nghĩa và tổ chức dữ liệu, cho phép trình duyệt hiểu và hiển thị nội dung theo một cách cụ thể. Mặc dù HTML không phải là ngôn ngữ lập trình, nhưng nó là bước đầu tiên và cốt lõi để tạo ra bất kỳ trang web nào.

HTML hoạt động dựa trên nguyên tắc sử dụng các thẻ để bao bọc nội dung. Mỗi thẻ có một vai trò riêng, ví dụ như thẻ <p> dùng để viết đoạn văn, <h1> dùng để hiển thị tiêu đề chính, hoặc <a> để tạo liên kết. Thông qua việc sử dụng HTML, người lập trình có thể tổ chức nội dung trang web một cách rõ ràng, dễ quản lý và dễ truy cập.

### Lịch sử phát triển

HTML được phát minh vào năm 1991 bởi Tim Berners-Lee, cha đẻ của World Wide Web. Ban đầu, HTML được tạo ra để phục vụ nhu cầu tổ chức tài liệu khoa học và chia sẻ thông tin qua Internet. Qua nhiều năm, HTML đã phát triển mạnh mẽ và trở thành tiêu chuẩn toàn cầu, được sử dụng trong mọi lĩnh vực liên quan đến web.

Các phiên bản của HTML:

* HTML 1.0 (1993): Phiên bản đầu tiên, tập trung vào các yếu tố cơ bản như đoạn văn, tiêu đề và liên kết.
* HTML 2.0 (1995): Phiên bản cải tiến, bổ sung thêm các tính năng biểu mẫu (forms) và bảng biểu (tables).
* HTML 3.2 (1997): Hỗ trợ các yếu tố đa phương tiện cơ bản và khả năng sử dụng JavaScript.
* HTML 4.01 (1999): Phiên bản ổn định, hỗ trợ CSS để định dạng giao diện và cải thiện khả năng tương tác người dùng.
* HTML5 (2014): Phiên bản hiện đại nhất, hỗ trợ các thẻ mới như <video>,

<audio>, <canvas>, cùng với các API mạnh mẽ để tích hợp đa phương tiện và công nghệ web hiện đại.

### Điểm nhấn trong HTML5

HTML5 được coi là một cuộc cách mạng trong phát triển web với các tính năng nổi bật:

* Hỗ trợ đa phương tiện: Các thẻ <video> và <audio> cho phép nhúng và phát nội dung đa phương tiện mà không cần phần mềm bên thứ ba như Flash.
* Thẻ cấu trúc mới: Các thẻ như <header>, <footer>, <article>, <section>

giúp tổ chức nội dung một cách logic và rõ ràng hơn.

* Tương thích thiết bị di động: Hỗ trợ tốt hơn cho các thiết bị như điện thoại thông minh và máy tính bảng thông qua các thuộc tính như viewport.
* API mạnh mẽ: HTML5 tích hợp các API như Canvas (vẽ đồ họa), Geolocation (định vị), và Web Storage (lưu trữ dữ liệu cục bộ).3. Vai trò của HTML trong phát triển web.

### Vai trò của HTML trong phát triển web

HTML là yếu tố nền tảng để xây dựng mọi website. Không có HTML, trình duyệt sẽ không thể hiển thị nội dung và người dùng không thể truy cập thông tin trên Internet. Một số vai trò chính của HTML bao gồm:

* + - 1. *Tạo cấu trúc nội dung*

HTML giúp tổ chức nội dung trang web thành các phần rõ ràng, từ tiêu đề, đoạn văn, hình ảnh đến bảng biểu và biểu mẫu. Điều này không chỉ giúp trình duyệt hiển thị nội dung một cách hợp lý mà còn cải thiện trải nghiệm người dùng.

* + - 1. *Hiển thị nội dung trên trình duyệt*

HTML đóng vai trò như một "người dịch", giúp trình duyệt hiểu cách hiển thị nội dung. Một đoạn mã HTML có thể xác định rằng một phần văn bản là tiêu đề, một phần khác là danh sách, hoặc một hình ảnh cần được hiển thị ở vị trí cụ thể.

Ví dụ:

<h1>Chào mừng đến với website của chúng tôi</h1>

<p>Đây là một đoạn văn mô tả nội dung.</p>

<img src="hinh-anh.jpg" alt="Hình minh họa">

* + - 1. *Tích hợp đa phương tiện*

HTML cho phép nhúng hình ảnh, video và âm thanh vào trang web một cách dễ dàng, tạo nên trải nghiệm sống động và hấp dẫn cho người dùng. Các thẻ như

<img>, <video>, và <audio> được sử dụng phổ biến trong các trang web hiện đại.

Ví dụ:

<video controls>

<source src="video.mp4" type="video/mp4"> Trình duyệt của bạn không hỗ trợ video.

</video>

* + - 1. *Kết hợp với các công nghệ khác*

HTML thường được sử dụng cùng với CSS (Cascading Style Sheets) để định dạng giao diện và JavaScript để thêm tính năng động. Sự kết hợp này tạo ra các trang web không chỉ đẹp mắt mà còn tương tác cao.

### 2.1.5. Cấu trúc cơ bản của HTML

Một tài liệu HTML chuẩn bắt đầu với khai báo <!DOCTYPE html> để chỉ định phiên bản HTML, theo sau là các thẻ <html>, <head> và <body>.

Ví dụ:

<!DOCTYPE html>

<html lang="vi">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Tiêu đề trang</title>

</head>

<body>

<header>

<h1>Chào mừng bạn</h1>

</header>

<main>

<p>Nội dung chính của trang web.</p>

</main>

<footer>

<p>Bản quyền © 2024</p>

</footer>

</body>

</html>

### Các thành phần chính của HTML

* + - 1. *Thẻ (Tags)*

Thẻ là yếu tố cơ bản nhất trong HTML, được sử dụng để định nghĩa các phần tử trên trang web. Thẻ bao gồm:

* Thẻ mở <h1>Đánh dấu điểm bắt đầu của một phần tử.
* Thẻ đóng </h1>: Đánh dấu điểm kết thúc.
* Thẻ tự đóng: Không cần thẻ đóng riêng biệt.

<img src="hinh-anh.jpg" alt="Hình minh họa">

* + - 1. *Thuộc tính (Attributes)*

Thuộc tính cung cấp thông tin bổ sung cho thẻ, thường đi kèm với một giá trị. Ví dụ:

<a href="https://example.com">Truy cập</a>

* + - 1. *Nội dung (Content)*

Nội dung là phần thông tin nằm giữa thẻ mở và thẻ đóng, được hiển thị cho người dùng.

### Các loại thẻ phổ biến

* + - 1. *Thẻ tiêu đề (Heading):*

Dùng để hiển thị tiêu đề, từ <h1> đến <h6>.

<h1>Tiêu đề chính</h1>

<h2>Tiêu đề phụ</h2>

* + - 1. *Thẻ đoạn văn (Paragraph)****:***

Hiển thị đoạn văn bản.

<p>Đây là một đoạn văn bản.</p>

* + - 1. *Thẻ liên kết (Link):*

Tạo liên kết đến trang khác hoặc tài nguyên khác.

<a href="https://example.com">Truy cập</a>

* + - 1. *Thẻ hình ảnh (Image)****:***

Nhúng hình ảnh vào trang web.

<img src="hinh-anh.jpg" alt="Hình ảnh minh họa">

### Ưu điểm và hạn chế của HTML

* + - 1. *Ưu điểm*
* Dễ học**:** Ngôn ngữ đơn giản, dễ tiếp cận.
* Tương thích rộng rãi**:** Hoạt động trên mọi trình duyệt và thiết bị.
* Mã nguồn mở**:** Không tốn chi phí.
  + - 1. *Hạn chế*
* Không có khả năng xử lý logic**:** Không thể thực hiện các tác vụ như tính toán hoặc điều kiện.
* Phụ thuộc vào CSS và JavaScript**:** Để tạo giao diện đẹp và tính năng động.

### Kết luận

HTML là nền tảng xây dựng mọi website, từ những trang web tĩnh cơ bản đến các ứng dụng web phức tạp. Việc hiểu rõ HTML không chỉ giúp tạo ra các trang web hiệu quả mà còn là bước đệm để tiếp cận các công nghệ web tiên tiến khác.

### Ngôn ngữ CSS (Cascading Style Sheets)

### Khái niệm

CSS (Cascading Style Sheets) là một ngôn ngữ kiểu dáng (style sheet language) được sử dụng để mô tả cách thức các phần tử trong tài liệu HTML được hiển thị. CSS chủ yếu được sử dụng để kiểm soát cách bố trí và định dạng các phần tử trên trang web như màu sắc, font chữ, kích thước, khoảng cách, và nhiều yếu tố khác, nhằm tạo ra giao diện người dùng đẹp mắt và dễ sử dụng.

CSS tách biệt phần nội dung (HTML) với phần trình bày (style), giúp việc quản lý, bảo trì và tái sử dụng mã nguồn trở nên dễ dàng hơn. Với CSS, nhà phát triển có thể điều chỉnh giao diện của trang web mà không cần phải thay đổi phần nội dung HTML. Điều này giúp việc phát triển trang web trở nên hiệu quả, linh hoạt và dễ dàng duy trì trong suốt vòng đời của nó.

CSS sử dụng các quy tắc để thay đổi cách các phần tử HTML được hiển thị, trong đó mỗi quy tắc bao gồm bộ chọn (selector) và một hoặc nhiều thuộc tính (property) cùng với giá trị tương ứng. Các quy tắc này có thể được áp dụng nội bộ (trong trang HTML), hoặc liên kết từ một tệp CSS ngoài (external stylesheet).

### Lịch sử phát triển

CSS được Håkon Wium Lie phát minh vào năm 1996, khi ông làm việc tại CERN (Cơ quan Nghiên cứu Hạt nhân Châu Âu) với Tim Berners-Lee, người sáng lập World Wide Web. Trước khi CSS ra đời, tất cả các thuộc tính định dạng và trình bày đều được nhúng trực tiếp trong các thẻ HTML, điều này khiến mã nguồn trở nên rất khó quản lý, đặc biệt là đối với các trang web lớn. CSS được phát triển để tách biệt phần cấu trúc (HTML) với phần trình bày (CSS), giúp cải thiện khả năng bảo trì và tính linh hoạt của các trang web.

CSS đã trải qua nhiều phiên bản và cải tiến để đáp ứng nhu cầu phát triển web hiện đại, từ CSS1, CSS2 đến CSS3, với CSS3 hiện nay hỗ trợ một loạt các tính năng tiên tiến như hiệu ứng động (animations), bố cục linh hoạt (flexbox), lưới (grid), và nhiều tính năng khác.

Các phiên bản chính của CSS:

* CSS1 (1996): Phiên bản đầu tiên của CSS, hỗ trợ các thuộc tính cơ bản như màu sắc, kích thước font, và căn chỉnh văn bản.
* CSS2 (1998): Phiên bản CSS mở rộng, bao gồm các tính năng như định vị các phần tử trên trang, hỗ trợ phương tiện (media types), và các kỹ thuật trình bày nâng cao như bảng (table formatting).
* CSS2.1 (2011): Một phiên bản cập nhật và sửa lỗi từ CSS2, CSS2.1 là phiên bản ổn định và được sử dụng rộng rãi trong nhiều năm trước khi CSS3 ra đời.
* CSS3 (2012): Phiên bản lớn và quan trọng nhất của CSS, chia thành nhiều mô-đun nhỏ, mỗi mô-đun tập trung vào một tính năng cụ thể. CSS3 mang đến những cải tiến vượt bậc trong thiết kế giao diện web như hiệu ứng chuyển động (animations), bố cục lưới (grid layout), và các yếu tố tương tác động (hover, active states).

### Vai trò của CSS trong phát triển web

CSS đóng vai trò rất quan trọng trong việc tạo ra trải nghiệm người dùng trên web. Nó không chỉ giúp làm đẹp giao diện mà còn cung cấp những công cụ mạnh mẽ giúp các nhà phát triển điều khiển cách mà nội dung được trình bày trên các thiết bị khác nhau. Một số vai trò quan trọng của CSS bao gồm:

* + - 1. *Tạo giao diện đẹp mắt*

CSS cho phép nhà phát triển trang trí và tạo ra giao diện trang web hấp dẫn. Bằng cách điều chỉnh màu sắc, font chữ, kích thước, khoảng cách và các yếu tố giao diện khác, CSS giúp trang web trở nên bắt mắt hơn và thu hút người dùng.

Ví dụ:

h1 {

color: #333; font-size: 36px;

text-align: center;

font-family: 'Arial', sans-serif;

}

* + - 1. *Bố cục trang web*

CSS cung cấp các kỹ thuật bố trí mạnh mẽ, giúp các phần tử HTML được sắp xếp hợp lý trên trang. Các kỹ thuật như flexbox, grid, float, và positioning cho phép nhà phát triển tạo ra các bố cục linh hoạt và đẹp mắt, từ các trang web đơn giản đến các ứng dụng phức tạp.

Ví dụ sử dụng flexbox:

.container {

display: flex;

justify-content: space-between; align-items: center;

}

* + - 1. *Tạo hiệu ứng động và tương tác*

CSS cho phép thêm các hiệu ứng động và tương tác vào các phần tử trên trang. Các hiệu ứng như chuyển động (animations), thay đổi màu sắc khi di chuột (hover effects), và chuyển đổi (transitions) giúp trang web trở nên sống động và thú vị hơn.

Ví dụ về hiệu ứng hover:

button:hover {

background-color: #4CAF50; color: white;

transition: background-color 0.3s ease;

}

* + - 1. *Thiết kế responsive*

Một trong những tính năng mạnh mẽ của CSS là khả năng tạo ra giao diện responsive—tức là giao diện tự động thay đổi và điều chỉnh sao cho phù hợp với các kích thước màn hình khác nhau. Với media queries, CSS có thể áp dụng các quy

tắc khác nhau tùy thuộc vào kích thước và độ phân giải của màn hình, giúp trang web hoạt động tốt trên các thiết bị từ máy tính để bàn đến điện thoại di động.

Ví dụ về media query:

@media (max-width: 768px) {

.menu {

display: none;

}

}

### Cấu trúc cơ bản của CSS

CSS có thể được áp dụng vào tài liệu HTML theo ba cách chủ yếu:

* + - 1. *CSS nội dòng (Inline CSS)*

CSS có thể được thêm trực tiếp vào một phần tử HTML thông qua thuộc tính style. Đây là cách dễ dàng nhất nhưng không linh hoạt khi áp dụng cho các phần tử khác nhau.

Ví dụ:

<p style="color: red; font-size: 20px;">Đoạn văn có màu

đỏ và kích thước chữ lớn.</p>

* + - 1. *CSS nội bộ (Internal CSS)*

CSS có thể được khai báo trong phần <head> của tài liệu HTML, trong thẻ

<style>. Đây là phương pháp hữu ích khi áp dụng các quy tắc cho trang HTML cụ thể mà không cần phải tạo tệp CSS riêng biệt.

Ví dụ:

<head>

<style>

body {

background-color: #f0f0f0;

}

h1 {

font-family: 'Arial', sans-serif;

}

</style>

</head>

* + - 1. *CSS ngoại vi (External CSS)*

Phương pháp phổ biến và được khuyến khích khi phát triển web là sử dụng CSS ngoại vi, tức là tạo một tệp CSS riêng biệt và liên kết tệp đó với trang HTML. Điều này giúp giữ mã nguồn gọn gàng, dễ bảo trì, và có thể tái sử dụng trên nhiều trang khác nhau.

Ví dụ:

<link rel="stylesheet" href="styles.css">

### Các thành phần cơ bản trong CSS

* + - 1. *Bộ chọn (Selectors)*

Bộ chọn là phần tử trong CSS dùng để chỉ định các phần tử HTML mà các quy tắc CSS sẽ áp dụng. Một số loại bộ chọn phổ biến là:

* Bộ chọn phần tử: Lựa chọn tất cả các phần tử của một loại nhất định, như

<h1>, <p>, … Ví dụ:

h1 {

color: blue;

}

* Bộ chọn lớp (class): Lựa chọn tất cả các phần tử có một lớp CSS cụ thể.

Ví dụ:

.highlight {

background-color: yellow;

}

* Bộ chọn ID: Lựa chọn các phần tử có thuộc tính id cụ thể.

Ví dụ:

#main-header {

font-size: 24px;

}

* Bộ chọn thuộc tính: Lựa chọn các phần tử có thuộc tính nhất định.

Ví dụ:

a[href^="https"] { color: green;

}

* + - 1. *Thuộc tính và giá trị (Properties and Values)*

Mỗi quy tắc CSS bao gồm một hoặc nhiều thuộc tính và giá trị. Thuộc tính xác định loại thay đổi bạn muốn áp dụng (ví dụ: màu sắc, kích thước font chữ), và giá trị xác định cách thay đổi đó sẽ xảy ra.

Ví dụ:

h2 {

color: red; /\* Thuộc tính: color, giá trị: red \*/ font-size: 28px; /\* Thuộc tính: font-size, giá trị:

28px \*/

}

* + - 1. *Nhóm quy tắc (Rulesets)*

Một nhóm quy tắc CSS gồm bộ chọn và các thuộc tính với giá trị tương ứng. Ví dụ:

p {

color: black; font-size: 16px; line-height: 1.5;

}

### Các mô-đun quan trọng trong CSS

CSS3 chia thành nhiều mô-đun, mỗi mô-đun tập trung vào một tính năng cụ thể. Một số mô-đun quan trọng bao gồm:

* CSS Grid: Hệ thống bố trí lưới mạnh mẽ, giúp tạo ra các giao diện phức tạp với cấu trúc lưới.
* Flexbox: Một mô-đun bố trí giúp căn chỉnh và phân phối không gian giữa các phần tử trong một container theo chiều ngang hoặc chiều dọc.
* CSS Animations: Tạo ra các hiệu ứng chuyển động mượt mà cho các phần tử.
* Media Queries: Cải thiện khả năng thiết kế responsive, giúp trang web tự động thay đổi giao diện phù hợp với các loại thiết bị khác nhau.

### Ưu điểm và hạn chế của CSS

* + - 1. *Ưu điểm*
* Tách biệt trình bày và nội dung: Điều này giúp mã nguồn dễ dàng bảo trì và quản lý.
* Định dạng dễ dàng và linh hoạt: CSS cung cấp nhiều công cụ để tạo ra các giao diện đẹp mắt và phù hợp với nhiều thiết bị khác nhau.
* Tối ưu hóa hiệu suất: Tệp CSS có thể được tái sử dụng trên nhiều trang, giúp giảm thiểu việc lặp lại mã nguồn và tối ưu hóa hiệu suất của trang web.
* Tương thích với trình duyệt: CSS hỗ trợ hầu hết các trình duyệt web hiện đại và có thể dễ dàng tương thích với các phiên bản trước.
  + - 1. *Hạn chế*
* Khó quản lý đối với dự án lớn: Khi số lượng quy tắc CSS tăng lên, việc quản lý mã có thể trở nên phức tạp.
* Không hỗ trợ tính năng động: CSS chỉ giúp định dạng giao diện, nhưng không hỗ trợ các tính năng xử lý logic hay tương tác phức tạp. Để làm điều này, JavaScript cần được sử dụng kết hợp với CSS.

### Kết luận

CSS là một công cụ mạnh mẽ và không thể thiếu trong phát triển web hiện đại. Bằng cách tách biệt phần nội dung và phần trình bày, CSS giúp các nhà phát triển dễ dàng quản lý và tạo ra giao diện đẹp mắt, linh hoạt, và dễ duy trì. Việc hiểu và thành thạo CSS sẽ giúp bạn xây dựng các trang web và ứng dụng web chuyên nghiệp, hiệu quả và đáp ứng tốt nhu cầu của người dùng.

### Ngôn ngữ PHP (Hypertext Preprocessor)

### Khái niệm

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình mã nguồn mở, được thiết kế chủ yếu để phát triển các ứng dụng web động. PHP có thể nhúng trực tiếp trong mã HTML, giúp tạo ra các trang web động và tương tác với cơ sở dữ liệu, mang lại khả năng tạo ra các ứng dụng web phong phú và mạnh mẽ. Với khả năng xử lý trên máy chủ (server-side), PHP chủ yếu được sử dụng để xây dựng các ứng dụng web, hệ thống quản lý nội dung (CMS), và các nền tảng thương mại điện tử.

Với việc hỗ trợ mạnh mẽ các thao tác với cơ sở dữ liệu, đặc biệt là MySQL, PHP cho phép các nhà phát triển dễ dàng xây dựng các hệ thống quản lý dữ liệu, cho phép người dùng tạo, sửa đổi và xóa dữ liệu qua giao diện web. Hơn nữa, PHP còn có khả năng xử lý các tác vụ như xác thực người dùng, quản lý phiên làm việc (sessions), gửi email, và xây dựng các API.

PHP là một ngôn ngữ lập trình phổ biến và được sử dụng rộng rãi trên thế giới nhờ vào tính linh hoạt, mã nguồn mở và cộng đồng hỗ trợ lớn. Nó là ngôn ngữ được sử dụng cho nhiều ứng dụng web nổi tiếng như WordPress, Facebook, và Wikipedia.

### Lịch sử phát triển

PHP được phát triển lần đầu tiên vào năm 1993 bởi Rasmus Lerdorf dưới cái tên “Personal Home Page Tools” (công cụ trang chủ cá nhân). Ban đầu, PHP chỉ là một tập hợp các script giúp Lerdorf theo dõi truy cập vào trang web cá nhân của mình. Tuy nhiên, sau đó, PHP đã được phát triển thêm và trở thành một ngôn ngữ lập trình mạnh mẽ, hỗ trợ nhiều tính năng như tương tác với cơ sở dữ liệu, gửi email và xử lý các biểu mẫu.

Phiên bản PHP đầu tiên được phát hành vào năm 1995, và từ đó đến nay, PHP đã liên tục được cải tiến, với các phiên bản đáng chú ý như PHP 4, PHP 5 và PHP 7. Các phiên bản mới của PHP tiếp tục tối ưu hóa hiệu suất, cải thiện tính bảo mật và hỗ trợ các tính năng lập trình hiện đại.

Các phiên bản chính của PHP:

* PHP 3 (1998): Đây là phiên bản đầu tiên của PHP được phát hành rộng rãi và bắt đầu được sử dụng để phát triển web động. Nó hỗ trợ các tính năng cơ bản như xử lý biểu mẫu và kết nối cơ sở dữ liệu.
* PHP 4 (2000): Phiên bản này ra đời với các cải tiến lớn về hiệu suất và khả năng hỗ trợ các ứng dụng web phức tạp. PHP 4 cũng giới thiệu cơ chế xử lý đối tượng (object-oriented programming - OOP), mặc dù các tính năng OOP vẫn còn hạn chế.
* PHP 5 (2004): Đây là một bước nhảy vọt lớn với việc cải thiện hoàn toàn khả năng lập trình hướng đối tượng (OOP). PHP 5 hỗ trợ các tính năng như lớp và đối tượng, kế thừa, và giao diện (interface), giúp việc phát triển các ứng dụng web lớn trở nên dễ dàng hơn.
* PHP 7 (2015): Phiên bản này mang lại hiệu suất vượt trội nhờ vào việc tối ưu hóa mã nguồn và cải thiện khả năng xử lý song song. PHP 7 giúp các ứng dụng web chạy nhanh hơn và tiêu thụ ít tài nguyên hệ thống hơn.
* PHP 8 (2020): Phiên bản hiện tại của PHP, với nhiều cải tiến về tính năng và hiệu suất, bao gồm các cải tiến về khả năng hỗ trợ loại dữ liệu, các tính năng mới như JIT (Just In Time compilation) giúp cải thiện tốc độ thực thi mã, và tính bảo mật mạnh mẽ hơn.

### Vai trò của PHP trong phát triển web

PHP đóng vai trò quan trọng trong việc xây dựng các ứng dụng web động, đặc biệt là khi kết hợp với HTML, CSS, và JavaScript. PHP không chỉ giúp tạo ra các trang web động mà còn hỗ trợ xử lý các tác vụ phức tạp như kết nối cơ sở dữ liệu, xác thực người dùng, và xây dựng các tính năng tương tác.

* + - 1. *Tạo ứng dụng web động*

PHP cho phép tạo ra các trang web động, nghĩa là nội dung của trang có thể thay đổi dựa trên sự tương tác của người dùng hoặc dữ liệu từ cơ sở dữ liệu. Khi người dùng gửi thông tin qua biểu mẫu, PHP có thể xử lý dữ liệu đó và trả về kết quả ngay lập tức mà không cần phải tải lại trang web.

Ví dụ:

<?php

$name = $\_POST['name'];

echo "Chào mừng bạn, " . $name;

?>

* + - 1. *Tương tác với cơ sở dữ liệu*

PHP được biết đến với khả năng kết nối và thao tác với cơ sở dữ liệu rất mạnh mẽ, đặc biệt là với MySQL. PHP cung cấp các hàm và thư viện để thực hiện các thao tác cơ bản như tạo, đọc, cập nhật và xóa (CRUD) dữ liệu từ cơ sở dữ liệu.

Ví dụ kết nối PHP với MySQL:

<?php

$conn = new mysqli("localhost", "username", "password", "database");

if ($conn->connect\_error) {

die("Kết nối cơ sở dữ liệu thất bại: " . $conn-

>connect\_error);

}

echo "Kết nối thành công!";

?>

* + - 1. *Xác thực và quản lý người dùng*

PHP thường được sử dụng để xây dựng các hệ thống đăng ký và đăng nhập người dùng. Các tính năng như xác thực mật khẩu, tạo phiên làm việc (session), và mã hóa dữ liệu đều có thể thực hiện dễ dàng với PHP. Điều này giúp bảo vệ thông tin người dùng và đảm bảo các thao tác truy cập dữ liệu diễn ra an toàn.

Ví dụ về xác thực người dùng:

<?php

session\_start();

if ($\_POST['username'] == 'admin' &&

$\_POST['password'] == '1234') {

$\_SESSION['username'] = 'admin'; echo "Đăng nhập thành công!";

} else {

echo "Tên đăng nhập hoặc mật khẩu không đúng!";

}

?>

* + - 1. *Gửi email và thông báo*

Một tính năng mạnh mẽ khác của PHP là khả năng gửi email tự động. PHP cung cấp các hàm để gửi email, giúp các ứng dụng web có thể gửi thông báo, xác nhận đăng ký, hoặc gửi thông tin qua email cho người dùng.

Ví dụ gửi email trong PHP:

<?php

mail("[someone@example.com](mailto:someone@example.com)", "Chào bạn!", "Chúc bạn một ngày tốt lành!");

?>

### Cấu trúc cơ bản của PHP

PHP có thể được nhúng trực tiếp vào trong mã HTML hoặc có thể được sử dụng trong các tệp PHP riêng biệt. Đoạn mã PHP thường được đặt trong cặp thẻ

<?php ... ?>.

* + - 1. *PHP trong HTML*

PHP có thể được nhúng trực tiếp vào trong HTML để xử lý các tác vụ động khi trang được tải.

Ví dụ:

<!DOCTYPE html>

<html>

<head>

<title>Trang chủ</title>

</head>

<body>

<h1>Chào bạn, <?php echo $name; ?></h1>

</body>

</html>

* + - 1. *Tệp PHP riêng biệt*

PHP có thể được lưu trong các tệp có phần mở rộng .php. Điều này cho phép các tệp PHP thực thi mã phía máy chủ và trả về HTML cho trình duyệt.

Ví dụ tệp hello.php:

<?php

echo "Xin chào, đây là một trang PHP!";

?>

### Các tính năng đặc biệt của PHP

* Lập trình hướng đối tượng (OOP): PHP hỗ trợ lập trình hướng đối tượng, cho phép nhà phát triển tạo các lớp và đối tượng, giúp mã nguồn trở nên dễ bảo trì và mở rộng hơn.
* Quản lý lỗi và ngoại lệ: PHP cung cấp cơ chế xử lý lỗi và ngoại lệ giúp xử lý các tình huống bất ngờ trong quá trình thực thi mã.
* Thư viện phong phú: PHP có rất nhiều thư viện và framework hỗ trợ phát triển web nhanh chóng và hiệu quả, như Laravel, Symfony, CodeIgniter, giúp giảm bớt khối lượng công việc và tăng tốc quá trình phát triển.

### Ưu điểm và hạn chế của PHP

* 1. *Ưu điểm*
* Mã nguồn mở: PHP là ngôn ngữ miễn phí và có cộng đồng người dùng lớn, giúp dễ dàng tìm kiếm tài liệu và hỗ trợ.
* Tính linh hoạt: PHP có thể tích hợp dễ dàng với nhiều loại cơ sở dữ liệu khác nhau và có khả năng hoạt động trên hầu hết các hệ điều hành và máy chủ web.
* Hiệu suất cao: PHP có thể xử lý hàng triệu yêu cầu mỗi ngày, điều này giúp xây dựng các ứng dụng web quy mô lớn.
* Cộng đồng hỗ trợ lớn: Với sự phát triển mạnh mẽ của cộng đồng, PHP luôn có rất nhiều tài liệu, diễn đàn, và công cụ hỗ trợ phát triển ứng dụng.
  1. *Hạn chế*
* Khả năng mở rộng: Đối với các ứng dụng web lớn và phức tạp, PHP có thể gặp khó khăn trong việc xử lý các tác vụ đòi hỏi khả năng mở rộng mạnh mẽ.
* Tính bảo mật: Mặc dù PHP cung cấp nhiều công cụ bảo mật, nhưng nếu không được cấu hình và sử dụng đúng cách, ứng dụng PHP có thể gặp phải các vấn đề.

### JavaScript (JS)

### Khái niệm

JavaScript là một ngôn ngữ lập trình động, được thiết kế để giúp các trang web trở nên tương tác và hấp dẫn hơn. Được phát triển vào năm 1995 bởi Brendan Eich tại Netscape, JavaScript đã nhanh chóng trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới. JavaScript chủ yếu chạy trên trình duyệt web của người dùng (client-side), nhưng với sự ra đời của Node.js, nó cũng được sử dụng trên máy chủ (server-side).

JavaScript được sử dụng để thêm các tính năng động vào trang web như xử lý sự kiện (event), tạo các hiệu ứng động, xác thực dữ liệu từ người dùng, thao tác với DOM (Document Object Model), và giao tiếp với máy chủ thông qua AJAX. Nó là một phần không thể thiếu trong bộ ba công nghệ phát triển web: HTML (cấu trúc), CSS (giao diện), và JavaScript (tương tác).

### Lịch sử phát triển

JavaScript ban đầu được tạo ra trong vòng 10 ngày dưới tên gọi “Mocha” và sau đó đổi thành “LiveScript” trước khi được đặt tên chính thức là JavaScript. Mặc dù tên gọi có vẻ liên quan đến Java, nhưng thực tế hai ngôn ngữ này hoàn toàn khác nhau.

Trong những năm đầu, JavaScript chủ yếu được sử dụng để thêm các tính năng cơ bản vào trang web, như tạo popup hoặc xử lý form. Tuy nhiên, với sự phát triển của công nghệ trình duyệt và sự ra đời của các framework và thư viện như jQuery, AngularJS, React, và Vue.js, JavaScript đã phát triển thành một ngôn ngữ lập trình mạnh mẽ, có thể xây dựng các ứng dụng web và mobile phức tạp.

Các mốc phát triển quan trọng

* 1995: Brendan Eich phát triển JavaScript tại Netscape.
* 1996: Microsoft phát hành phiên bản của riêng mình, gọi là JScript, để cạnh tranh với JavaScript.
* 1997: JavaScript được chuẩn hóa bởi tổ chức ECMA dưới tên gọi ECMAScript (ES).
* 2009: Node.js ra đời, mở ra khả năng sử dụng JavaScript trên máy chủ.
* 2015: ECMAScript 6 (ES6) được phát hành, mang lại nhiều cải tiến lớn như arrow functions, promises, template literals và classes.

### Vai trò của JavaScript trong phát triển web

JavaScript đóng vai trò cốt lõi trong việc xây dựng các trang web hiện đại. Với sự hỗ trợ của JavaScript, các trang web không chỉ hiển thị nội dung tĩnh mà còn cung cấp trải nghiệm tương tác, mượt mà, và linh hoạt.

* + - 1. *Tương tác với người dùng*

JavaScript cho phép thực hiện các thao tác động dựa trên hành vi của người dùng, chẳng hạn như nhấn chuột, di chuột, hoặc nhập dữ liệu. Ví dụ, khi người dùng nhấp vào một nút, JavaScript có thể thực hiện một hành động như hiển thị thông báo hoặc tải thêm nội dung.

Ví dụ:

document.getElementById("myButton").addEventListener("click", function() {

alert("Bạn đã nhấn nút!");

});

* + - 1. *Thao tác với DOM*

JavaScript có thể truy cập và thay đổi nội dung của trang web thông qua Document Object Model (DOM). Điều này cho phép cập nhật nội dung trang web mà không cần tải lại toàn bộ trang.

Ví dụ:

document.getElementById("myText").innerHTML = "Chào bạn!";

* + - 1. *Giao tiếp với máy chủ (AJAX)*

AJAX (Asynchronous JavaScript and XML) là một kỹ thuật cho phép trang web giao tiếp với máy chủ mà không cần tải lại trang. Điều này được sử dụng phổ biến trong các ứng dụng như tìm kiếm tự động, trò chuyện trực tiếp, và cập nhật dữ liệu thời gian thực.

Ví dụ:

fetch("https://api.example.com/data")

.then(response => response.json())

.then(data => console.log(data));

* + - 1. *Xây dựng ứng dụng web và mobile*

Với sự hỗ trợ của các thư viện và framework như React, Angular, và Vue.js, JavaScript được sử dụng để xây dựng các ứng dụng web đơn trang (SPA - Single Page Application) và thậm chí cả ứng dụng di động.

### Cấu trúc cơ bản của JavaScript

JavaScript là một ngôn ngữ lập trình dễ học, với cú pháp đơn giản và linh hoạt. Dưới đây là một số thành phần cơ bản:

* + - 1. *Biến (Variables)*

JavaScript sử dụng các từ khóa var, let, và const để khai báo biến. Ví dụ:

let name = "Nguyen Van A";

const PI = 3.14;

* + - 1. *Hàm (Functions)*

JavaScript hỗ trợ khai báo hàm để thực hiện các tác vụ cụ thể. Ví dụ:

function greet(name) {

return `Chào bạn, ${name}!`;

}

* + - 1. *Vòng lặp (Loops)*

JavaScript hỗ trợ các loại vòng lặp như for, while, và do...while. Ví dụ:

for (let i = 0; i < 5; i++) { console.log(i);

}

### Ưu điểm và hạn chế của JavaScript

* + - 1. *Ưu điểm*
* Chạy trên nhiều nền tảng: JavaScript hoạt động trên hầu hết các trình duyệt hiện đại mà không cần cài đặt thêm phần mềm.
* Tương tác mạnh mẽ: JavaScript giúp trang web trở nên sống động và phản hồi nhanh chóng.
* Dễ học: Với cú pháp đơn giản, JavaScript là ngôn ngữ lý tưởng cho người mới bắt đầu.
* Hệ sinh thái phong phú: Cộng đồng JavaScript đã phát triển rất nhiều thư viện và framework hỗ trợ.
  + - 1. *Hạn chế*
* Hiệu suất: Vì chạy trên trình duyệt, JavaScript có thể bị hạn chế bởi tài nguyên phần cứng của người dùng.
* Bảo mật: JavaScript chạy trên client-side, điều này khiến nó dễ bị tấn công nếu không được bảo vệ đúng cách.
* Tính không đồng nhất: Các trình duyệt cũ có thể không hỗ trợ đầy đủ các tính năng mới của JavaScript.

### Kết luận

JavaScript là một ngôn ngữ lập trình mạnh mẽ và không thể thiếu trong phát triển web hiện đại. Từ việc thêm các tính năng động vào trang web đến xây dựng các ứng dụng web và mobile phức tạp, JavaScript đã chứng tỏ tầm quan trọng của mình trong hệ sinh thái công nghệ ngày nay. Với sự phát triển không ngừng của các công cụ và framework, JavaScript tiếp tục là một lựa chọn hàng đầu cho các nhà phát triển web và ứng dụng.

### Ngôn ngữ SQL (Structured Query Language)

### Khái niệm

SQL (Structured Query Language) là ngôn ngữ chuẩn được sử dụng để quản lý và thao tác cơ sở dữ liệu quan hệ (Relational Database). SQL cho phép người dùng thực hiện các thao tác như truy vấn dữ liệu, cập nhật, xóa, thêm mới dữ liệu, cũng như quản lý và kiểm soát quyền truy cập.

Một cơ sở dữ liệu có thể được xem như một bộ sưu tập các bảng (tables) lưu trữ dữ liệu có cấu trúc. SQL chính là công cụ giúp người dùng giao tiếp với cơ sở dữ liệu để truy xuất và xử lý thông tin một cách dễ dàng.

Ví dụ: Truy vấn đơn giản để lấy danh sách khách hàng từ bảng customers:

SELECT \*

FROM customers;

## Lịch sử phát triển

SQL được phát triển vào những năm 1970 tại IBM bởi Donald D. Chamberlin và Raymond F. Boyce. Ban đầu, nó được gọi là SEQUEL (Structured English Query Language), nhưng sau đó tên này được rút gọn thành SQL.

SQL trở thành tiêu chuẩn quốc tế đầu tiên cho cơ sở dữ liệu quan hệ vào năm 1986 (ANSI) và năm 1987 (ISO). Ngày nay, SQL được sử dụng rộng rãi trong các hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL, Oracle Database, Microsoft SQL Server, và SQLite.

## Vai trò của SQL trong quản lý cơ sở dữ liệu

SQL đóng vai trò cốt lõi trong việc quản lý và khai thác dữ liệu. Một số vai trò chính của SQL bao gồm:

* + - 1. *Truy vấn dữ liệu*

SQL cho phép tìm kiếm và lấy dữ liệu từ các bảng trong cơ sở dữ liệu một cách nhanh chóng và hiệu quả.

Ví dụ: Lấy tên và tuổi của khách hàng:

SELECT name, age FROM customers;

* + - 1. *Thêm, sửa, xóa dữ liệu*

Ngôn ngữ SQL cung cấp các lệnh để thêm mới, cập nhật, và xóa dữ liệu trong cơ sở dữ liệu.

Ví dụ:

– Thêm dữ liệu:

INSERT INTO customers (name, age) VALUES ('Nguyen Van A', 25);

– Sửa dữ liệu:

UPDATE customers SET age = 26

WHERE name = 'Nguyen Van A';

* Xóa dữ liệu:

DELETE FROM customers

WHERE name = 'Nguyen Van A';

* + - 1. *Quản lý cấu trúc cơ sở dữ liệu*

SQL cho phép tạo, chỉnh sửa, và xóa bảng hoặc các đối tượng khác trong cơ sở dữ liệu.

Ví dụ:

* Tạo bảng:

CREATE TABLE customers ( id INT PRIMARY KEY, name VARCHAR(50),

age INT

);

* Thêm cột vào bảng:

ALTER TABLE customers ADD email VARCHAR(100);

* Xóa bảng:

DROP TABLE customers;

* + - 1. *Kiểm soát truy cập và bảo mật*

SQL cung cấp các lệnh để kiểm soát quyền truy cập của người dùng nhằm bảo vệ dữ liệu.

Ví dụ: Cấp quyền truy cập cho người dùng:

GRANT SELECT, INSERT

ON customers TO user1;

## Cấu trúc cơ bản của SQL

SQL bao gồm nhiều lệnh được chia thành các nhóm chính:

* DDL (Data Definition Language): Dùng để định nghĩa cấu trúc cơ sở dữ liệu.

+ CREATE: Tạo đối tượng như bảng, chỉ mục.

+ ALTER: Thay đổi cấu trúc bảng.

+ DROP: Xóa đối tượng.

* DML (Data Manipulation Language): Dùng để thao tác dữ liệu.

+ SELECT: Truy vấn dữ liệu.

+ INSERT: Thêm dữ liệu.

+ UPDATE: Cập nhật dữ liệu.

+ DELETE: Xóa dữ liệu.

* DCL (Data Control Language): Quản lý quyền truy cập.

+ GRANT: Cấp quyền.

+ REVOKE: Thu hồi quyền.

* TCL (Transaction Control Language): Quản lý giao dịch.

+ COMMIT: Xác nhận giao dịch.

+ ROLLBACK: Hoàn tác giao dịch.

## Ưu điểm và hạn chế của SQL

* + - 1. *Ưu điểm*
* Dễ học và sử dụng, cú pháp ngắn gọn, dễ hiểu.
* SQL là tiêu chuẩn quốc tế, hỗ trợ bởi hầu hết các hệ quản trị cơ sở dữ liệu.
* Khả năng xử lý dữ liệu lớn, thao tác trên hàng triệu bản ghi hiệu quả.
* Dễ dàng tích hợp với các ngôn ngữ lập trình như Python, Java, PHP.

*2.5.5.1 Hạn chế*

* SQL không thay thế được các ngôn ngữ lập trình.
* Một số câu lệnh SQL không tương thích hoàn toàn giữa các hệ quản trị.

## Ví dụ minh họa

Ví dụ: Quản lý thông tin sinh viên trong bảng students.

* Tạo bảng:

CREATE TABLE students ( id INT PRIMARY KEY, name VARCHAR(100),

age INT,

grade VARCHAR(10)

);

* Thêm dữ liệu:

INSERT INTO students (id, name, age, grade) VALUES (1, 'Nguyen Van A', 20, 'A');

* Truy vấn dữ liệu:

SELECT name, grade FROM students

WHERE age > 18;

* Cập nhật dữ liệu:

UPDATE students SET grade = 'B'

WHERE name = 'Nguyen Van A';

* Xóa dữ liệu:

DELETE FROM students

WHERE id = 1;

## Kết luận

SQL là ngôn ngữ không thể thiếu trong lĩnh vực quản lý và khai thác cơ sở dữ liệu. Với khả năng truy vấn mạnh mẽ, dễ học và tính chuẩn hóa cao, SQL đã trở thành công cụ quan trọng cho các nhà phát triển phần mềm, nhà quản trị cơ sở dữ liệu, và các nhà phân tích dữ liệu. Việc nắm vững SQL không chỉ mở ra cơ hội trong phát triển ứng dụng mà còn đặt nền tảng cho các công nghệ dữ liệu tiên tiến như Big Data và AI.

# CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

### Mô tả bài toán

Trong bối cảnh tuyển sinh ngày càng cạnh tranh, việc xây dựng một hệ thống quản lý tuyển sinh trực tuyến hiệu quả là vô cùng cần thiết. Để đáp ứng nhu cầu thông tin ngày càng tăng của thí sinh và phụ huynh, hệ thống quản lý tuyển sinh trực tuyến cần được xây dựng một cách bài bản với các chức năng như sau:

– Quản lý tin tức:

+ Thêm bài viết tin tức mới (tiêu đề, nội dung, hình ảnh).

+ Sửa đổi bài viết tin tức đã tồn tại.

+ Xóa bài viết tin tức.

– Quản lý ngành học:

+ Thêm ngành học mới (tên ngành, mã ngành, mô tả, yêu cầu đầu vào).

+ Sửa đổi thông tin ngành học.

+ Xóa ngành học.

* Quản lý thông tin tuyển sinh:

+ Cập nhật thông tin tuyển sinh chung (chỉ tiêu, phương thức xét tuyển, thời gian nộp hồ sơ).

+ Đăng tải các văn bản, thông báo liên quan đến tuyển sinh.

* Quản lý tài khoản quản trị:

+ Quản lý thông tin tài khoản quản trị.

+ Thay đổi mật khẩu tài khoản quản trị.

+ Xem danh sách tài khoản quản trị.

+ Thêm mới tài khoản quản trị.

+ Xóa tài khoản quản trị.

### Đặc tả nhu cầu

### Phân tích yêu cầu chức năng

Hệ thống website phải đáp ứng các chức năng chính sau để đảm bảo hoạt động hiệu quả:

* Quản trị viên có thể nhập liệu, chỉnh sửa, và xóa các thông tin liên quan đến tuyển sinh như ngành học, điều kiện xét tuyển, thời gian và hình thức thi tuyển.
* Thí sinh có thể đăng ký tư vấn, đọc tin tức và xem ngành học
* Hệ thống phân quyền người dùng, bao gồm quản trị viên và thí sinh, đảm bảo an toàn và dễ dàng quản lý.
* Cung cấp kênh liên hệ để thí sinh có thể gửi thắc mắc, phản hồi qua email hoặc biểu mẫu trực tuyến.

### Phân tích yêu cầu phi chức năng

Bên cạnh các chức năng chính, hệ thống cần đáp ứng các yêu cầu phi chức năng sau:

* Giao diện trực quan, dễ sử dụng, tối ưu hóa trên cả máy tính và thiết bị di động.
* Dễ dàng nâng cấp hoặc tích hợp thêm các tính năng mới trong tương lai, chẳng hạn như thanh toán trực tuyến hoặc tích hợp mạng xã hội.

### Phân tích và thiết kế hệ thống

### Kiến trúc hệ thống

* + - 1. *Kiến trúc 3 lớp (3-tier architecture):*

– Lớp trình bày (Presentation Layer):

+ Đây là giao diện người dùng (UI) mà người dùng tương tác.

+ Trong trường hợp này, nó có thể là một trang web hoặc ứng dụng di động cho phép người dùng xem thông tin tuyển sinh, tin tức, gửi tin nhắn liên hệ.

+ Admin cũng có thể sử dụng giao diện này để quản lý dữ liệu.

– Lớp ứng dụng (Application Layer/Business Logic Layer):

+ Đây là lớp chứa logic xử lý nghiệp vụ của ứng dụng.

+ Trong trường hợp này, nó bao gồm các controller và model xử lý các yêu cầu từ lớp trình bày và tương tác với lớp dữ liệu.

+ Các controller nhận yêu cầu từ lớp trình bày gọi các model để xử lý dữ liệu và trả kết quả về lớp trình bày.

+ Các model tương tác với lớp dữ liệu để truy xuất và cập nhật dữ liệu.

* Lớp dữ liệu (Data Layer):

+ Đây là lớp lưu trữ dữ liệu của ứng dụng.

+ Trong trường hợp này, nó là cơ sở dữ liệu MySQL admission\_db với các bảng: admins, news, programs, admission\_info, contact\_messages.

* + - 1. *Luồng hoạt động:*
* Người dùng tương tác với lớp trình bày.
* Lớp trình bày gửi yêu cầu đến lớp ứng dụng (controller).
* Controller gọi model để xử lý yêu cầu.
* Model tương tác với lớp dữ liệu (database) để truy xuất hoặc cập nhật dữ liệu.
* Lớp dữ liệu trả kết quả về model.
* Model trả kết quả về controller.
* Controller trả kết quả về lớp trình bày.
* Lớp trình bày hiển thị kết quả cho người dùng.
  + - 1. *Các thành phần chính:*
* Cơ sở dữ liệu (MySQL): Lưu trữ dữ liệu của hệ thống.
* Controller: Xử lý các yêu cầu từ lớp trình bày và điều phối hoạt động của các model.
* Model: Tương tác với cơ sở dữ liệu để truy xuất và cập nhật dữ liệu.
* Giao diện người dùng (UI): Cho phép người dùng và admin tương tác với hệ thống.

*3.3.1.4. Các chức năng chính:*

* Quản lý người dùng (admins): Đăng nhập, đăng xuất, quản lý tài khoản admin.
* Quản lý tin tức (news): Thêm, sửa, xóa, xem tin tức.
* Quản lý ngành học (programs): Thêm, sửa, xóa, xem danh sách ngành học.
* Quản lý thông tin tuyển sinh (admission\_info): Cập nhật thông tin tuyển sinh.
* Quản lý tin nhắn liên hệ (contact\_messages): Xem và trả lời tin nhắn từ người dùng.

### Mô hình nghiệp vụ (Use Case Diagram)

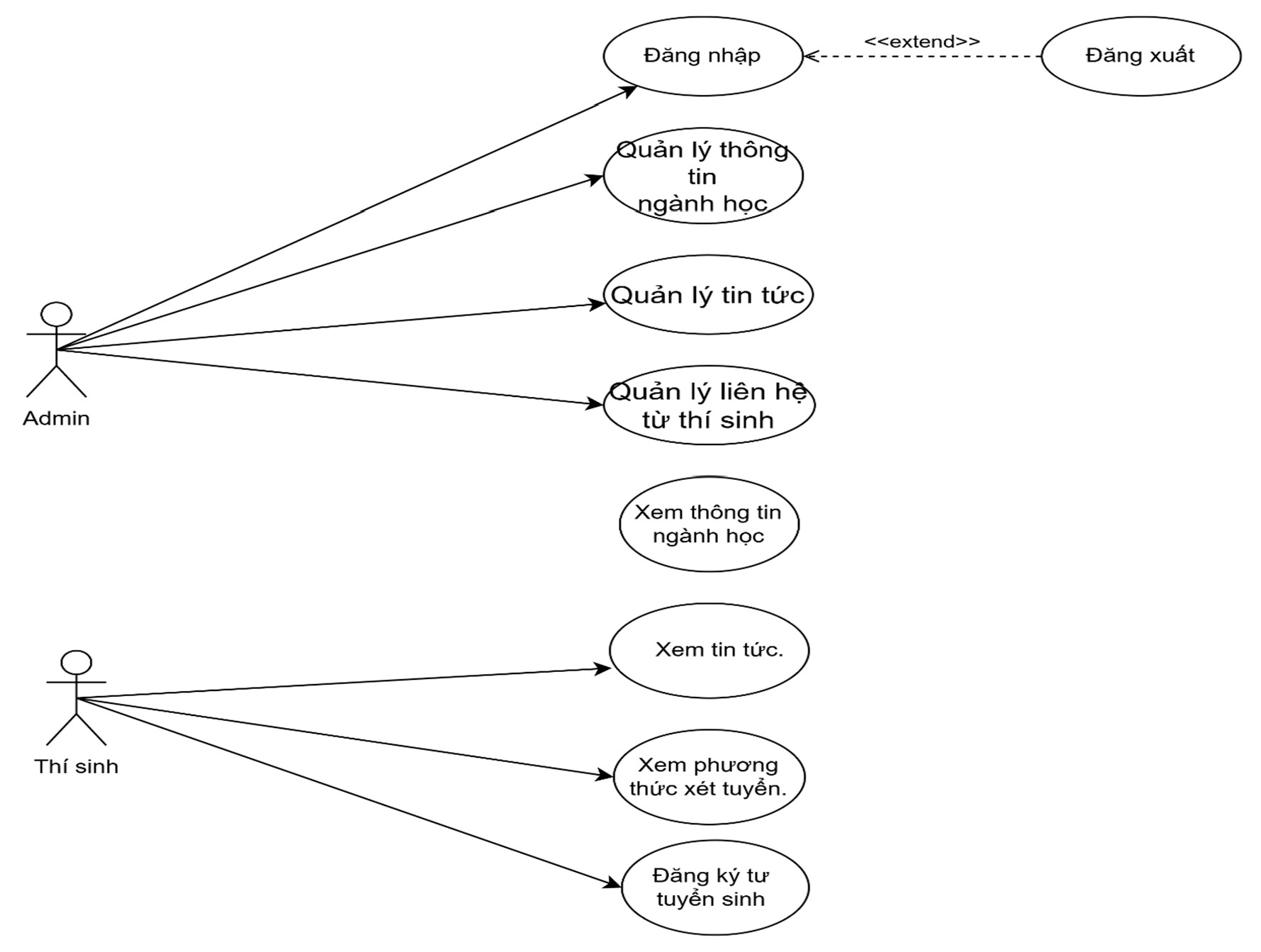
Hệ thống được phân tích dưới dạng các **Use Case** chính, trong đó các tác nhân bao gồm:

#### \* Thí sinh:

* Xem thông tin ngành học.
* Xem tin tức.
* Xem phương thức xét tuyển.
* Đăng ký tư vấn tuyển sinh.

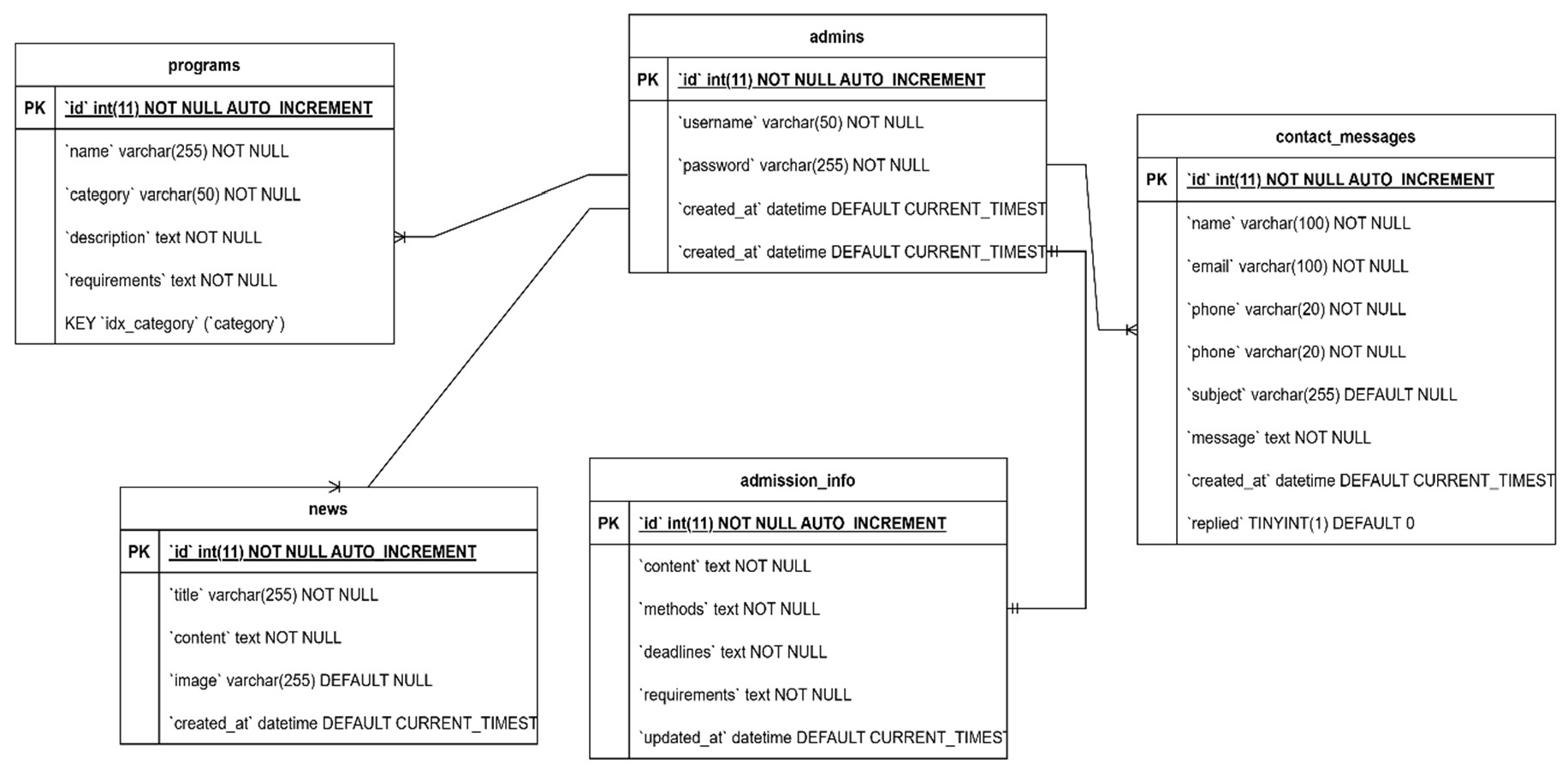
#### \* Quản trị viên:

* Đăng nhập, đăng xuất.
* Quản lý thông tin ngành học, chương trình đào tạo.
* Quản lý tin tức.
* Quản lý liên hệ từ thí sinh.



*Hình 3.1. Sơ đồ Use Case*

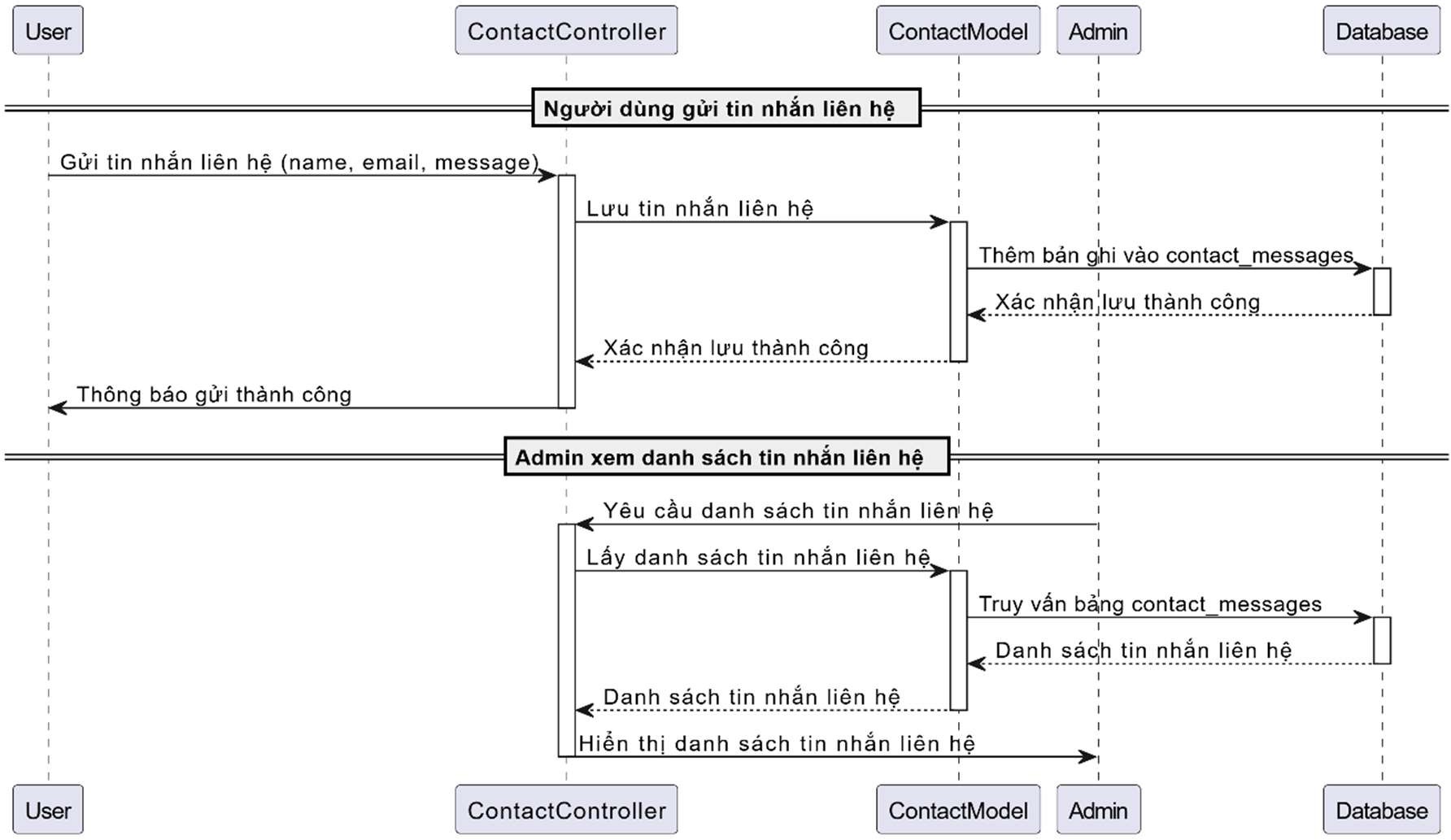
### Lược đồ thực thể quan hệ (ERD - Entity Relationship Diagram)

****

*Hình 3.2 Mô hình ERD*

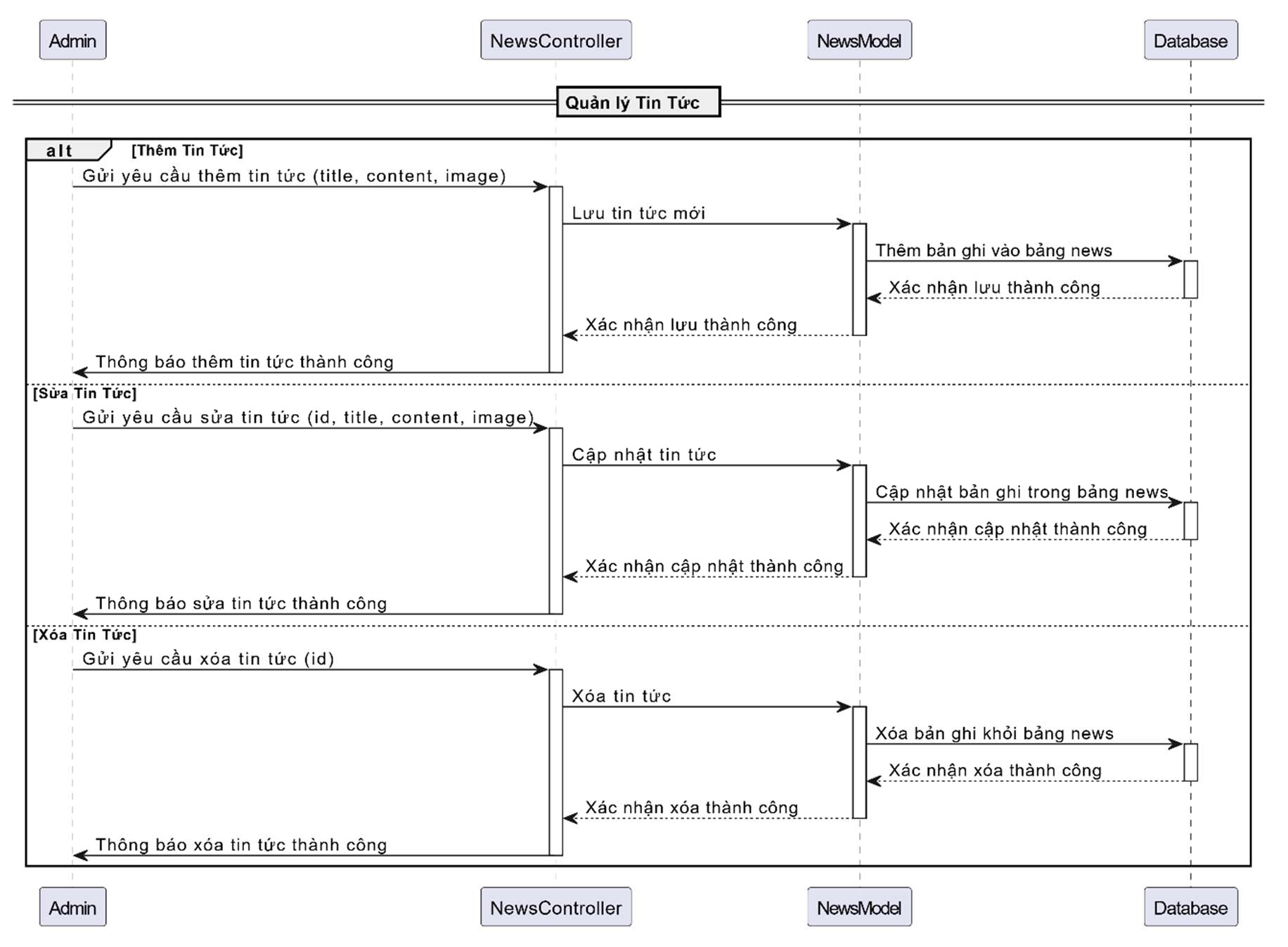
### Lược đồ tuần tự (Sequence Diagram)

* + - 1. *Gửi tin nhắn liên hệ và quản lý tin nhắn:*

**

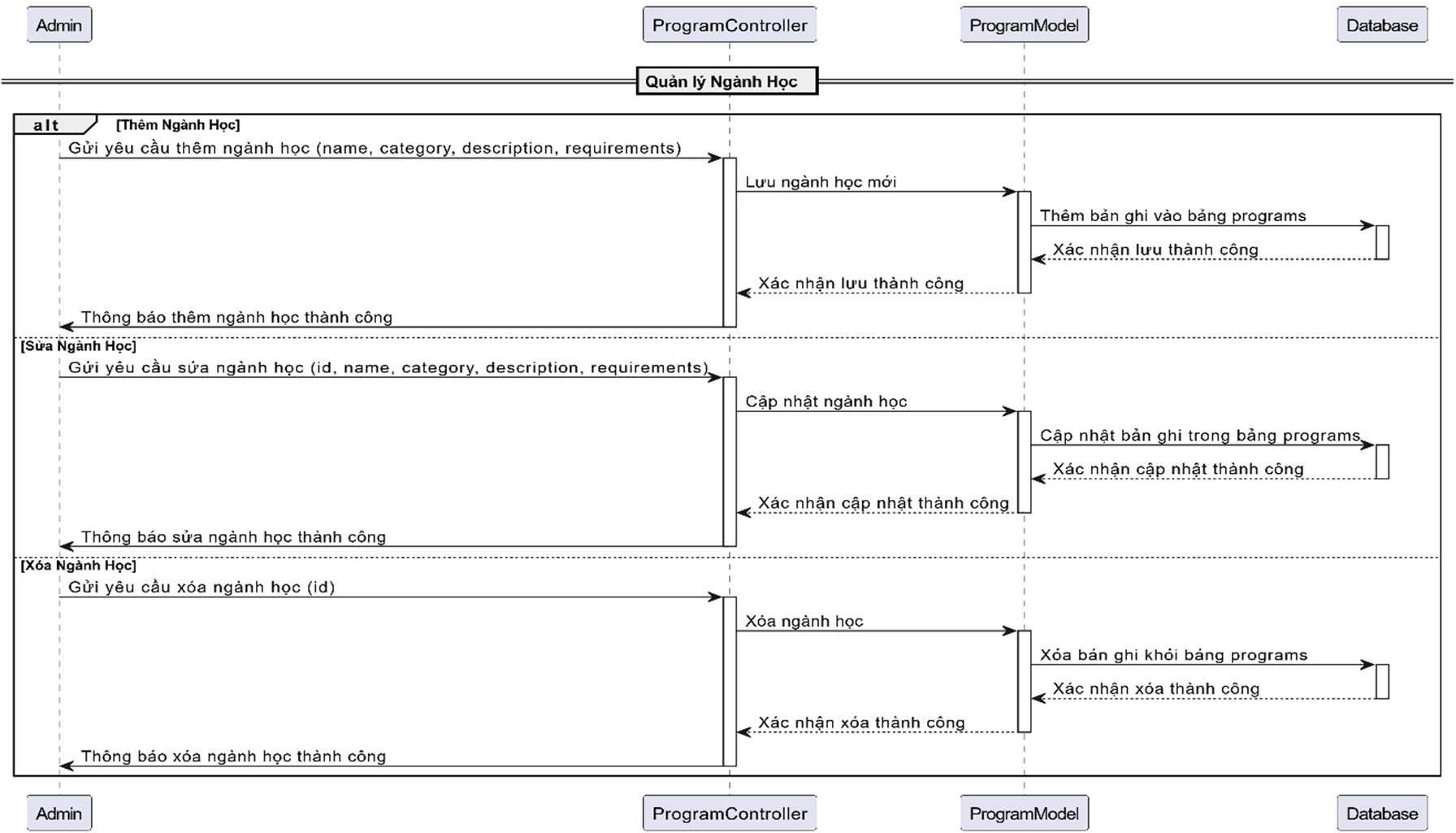
*Hình 3.3 Luồng xử lý tin nhắn tư vấn*

* + - 1. *Quản lý tin tức*

**

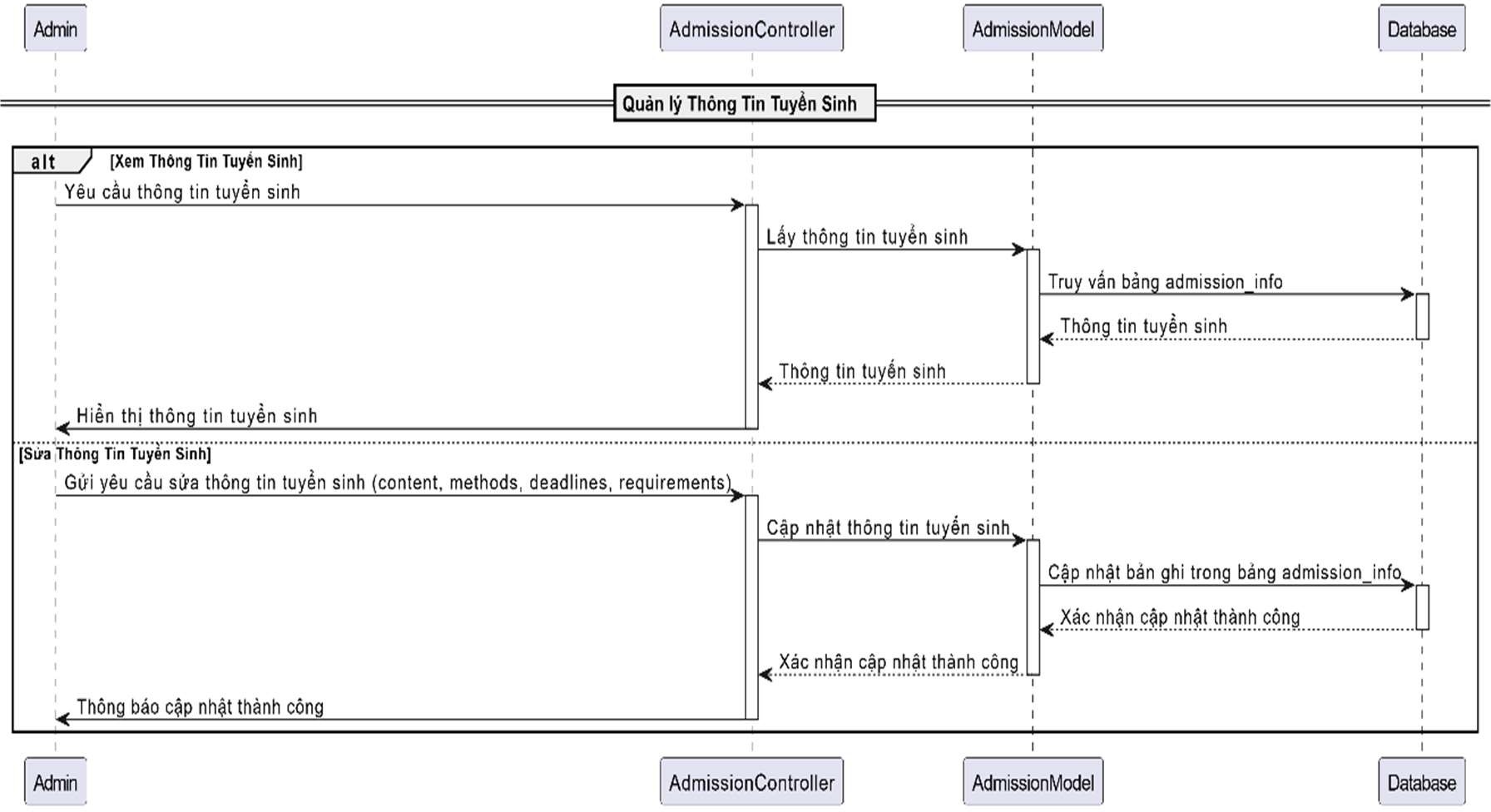
*Hình 3.4 Luồng xử lý quản lý tin tức của Admin*

* + - 1. *Quản lý ngành học*

**

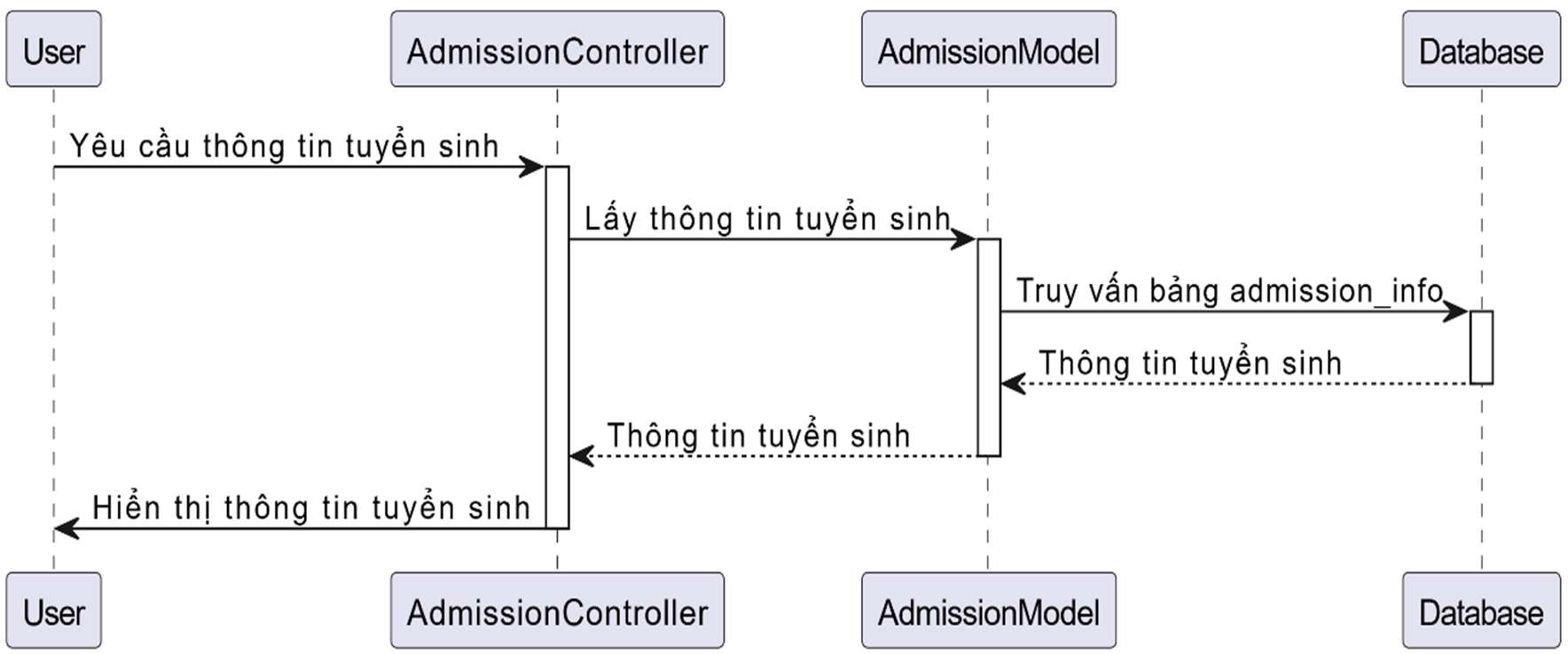
*Hình 3.5 Luồng xử lý Quản lý ngành học của Admin*

* + - 1. *Quản lý thông tin tuyển sinh:*

**

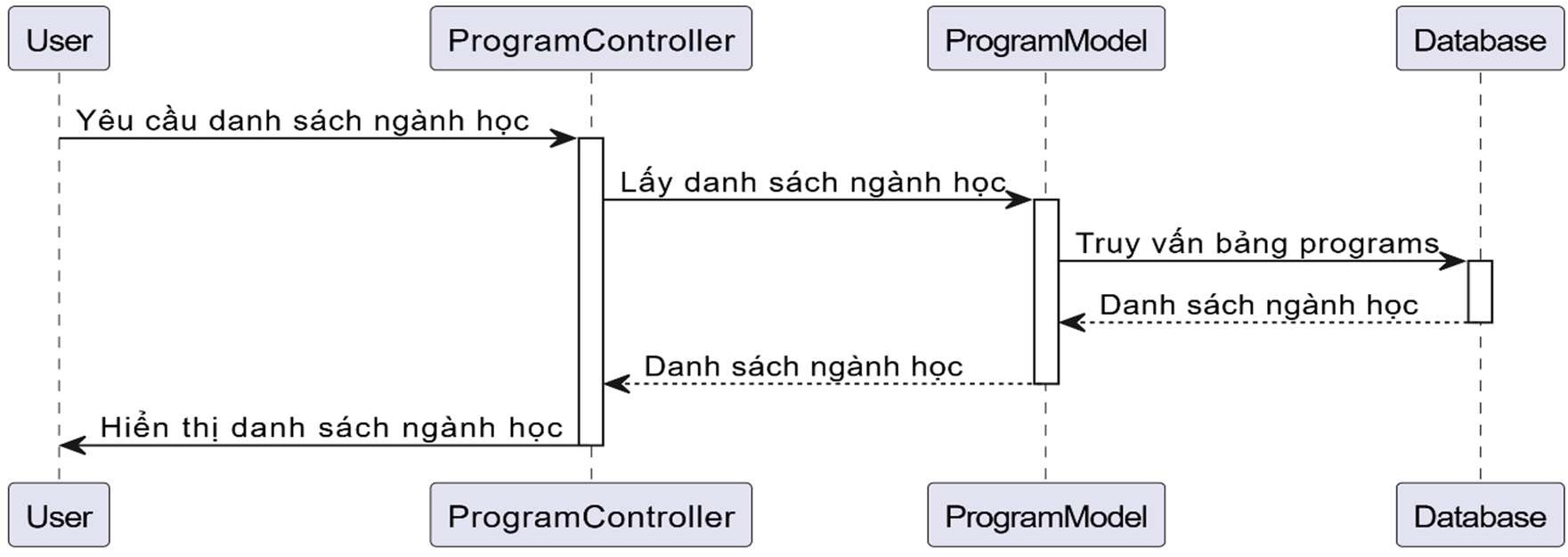
*Hình 3.6 Luồng xử lý quản lý thông tin tuyển sinh*

* + - 1. *Thí sinh xem thông tin tuyển sinh*

**

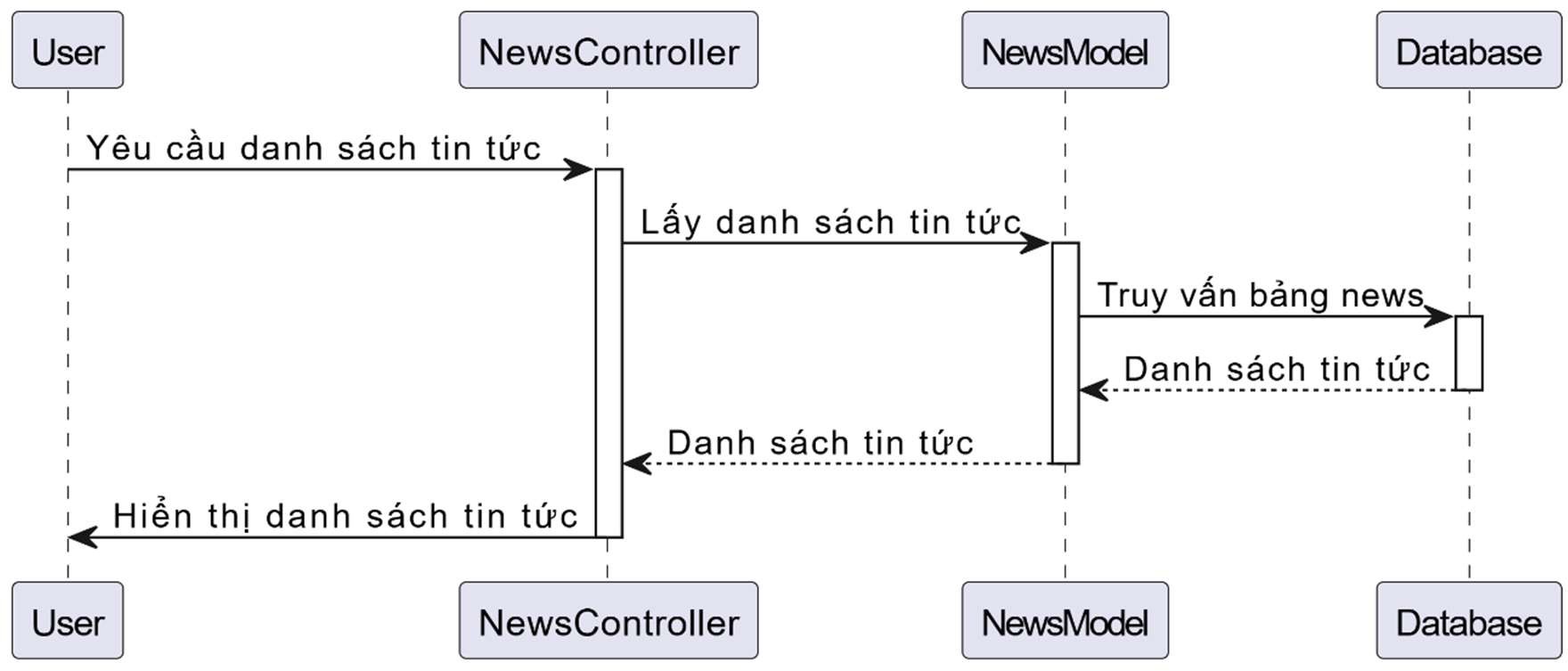
*Hình 3.7 Luồng xử lý xem thông tin tuyển sinh*

* + - 1. *Thí sinh xem thông tin ngành học*

**

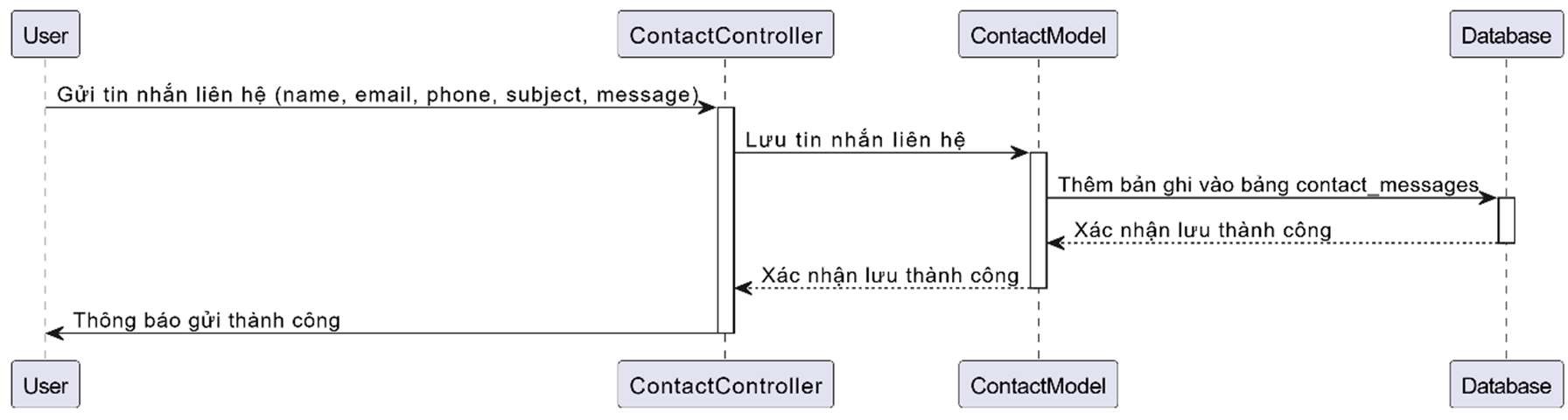
*Hình 3.8 Luồng xử lý xem thông tin ngành học*

* + - 1. *Thí sinh xem tin tức*

**

*Hình 3.9 Luồng xử lý xem tin tức*

* + - 1. *Thí sinh gửi tin nhắn liên hệ*

**

*Hình 3.10 Luồng xử lý gửi tin nhắn liên hệ*

### Thiết kế dữ liệu:

### Bảng admins

*Bảng 3.1 Bảng admins*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cột** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| id | INT(11) | Mã định danh duy nhất cho mỗi quản trị viên | Khóa chính, tự tăng |
| username | VARCHAR(50) | Tên đăng nhập của quản trị viên | Not null, duy nhất |
| password | VARCHAR(255) | Mật khẩu của quản trị viên (đã được băm) | Not null |
| created\_at | DATETIME | Thời gian tạo tài khoản | Mặc định CURRENT\_TIMESTAMP |

### Bảng news

*Bảng 3.2 Bảng news*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cột** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| id | INT(11) | Mã định danh duy nhất cho mỗi bài viết | Khóa chính, tự tăng |
| title | VARCHAR(255) | Tiêu đề bài viết | Not null |
| content | TEXT | Nội dung bài viết | Not null |
| image | VARCHAR(255) | Đường dẫn đến hình ảnh minh họa |  |
| created\_at | DATETIME | Thời gian đăng bài viết | Mặc định CURRENT\_TIMESTAMP |

### Bảng programs

*Bảng 3.3 Bảng programs*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cột** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| id | INT(11) | Mã định danh duy nhất cho mỗi chương trình | Khóa chính, tự tăng |
| name | VARCHAR(255) | Tên chương trình | Not null |
| category | VARCHAR(50) | Danh mục chương trình (ví dụ: đại học, thạc sĩ) | Not null |
| description | TEXT | Mô tả chương trình | Not null |
| requirements | TEXT | Yêu cầu đầu vào | Not null |
| idx\_category | INDEX | Index cho cột category | Index |

### Bảng admission\_info

*Bảng 3.4 Bảng admission\_info*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cột** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| id | INT(11) | Mã định danh duy nhất | Khóa chính, tự tăng |
| content | TEXT | Nội dung thông tin  tuyển sinh | Not null |
| methods | TEXT | Phương thức tuyển sinh | Not null |
| deadlines | TEXT | Thời hạn nộp hồ sơ | Not null |
| requirements | TEXT | Yêu cầu hồ sơ | Not null |
| updated\_at | DATETIME | Thời gian cập  nhật thông tin | Mặc định CURRENT\_TIMESTAMP ON  UPDATE CURRENT\_TIMESTAMP |

### 3.5.5 Bảng contact\_messages

*Bảng 3.5 Bảng contact\_messages*

|  |  |  |  |
| --- | --- | --- | --- |
| **Cột** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| id | INT(11) | Mã định danh duy nhất cho mỗi tin nhắn | Khóa chính, tự tăng |
| name | VARCHAR(100) | Tên người gửi | Not null |
| email | VARCHAR(100) | Email người gửi | Not null |
| phone | VARCHAR(20) | Số điện thoại người gửi | Not null |
| subject | VARCHAR(255) | Tiêu đề tin nhắn |  |
| message | TEXT | Nội dung tin nhắn | Not null |
| created\_at | DATETIME | Thời gian gửi tin nhắn | Mặc định CURRENT\_TIMESTAMP |
| replied | TINYINT(1) | Trạng thái đã trả lời (0: chưa, 1: đã) | Mặc định 0 |

### Cài đặt và triển khai

### Công nghệ sử dụng

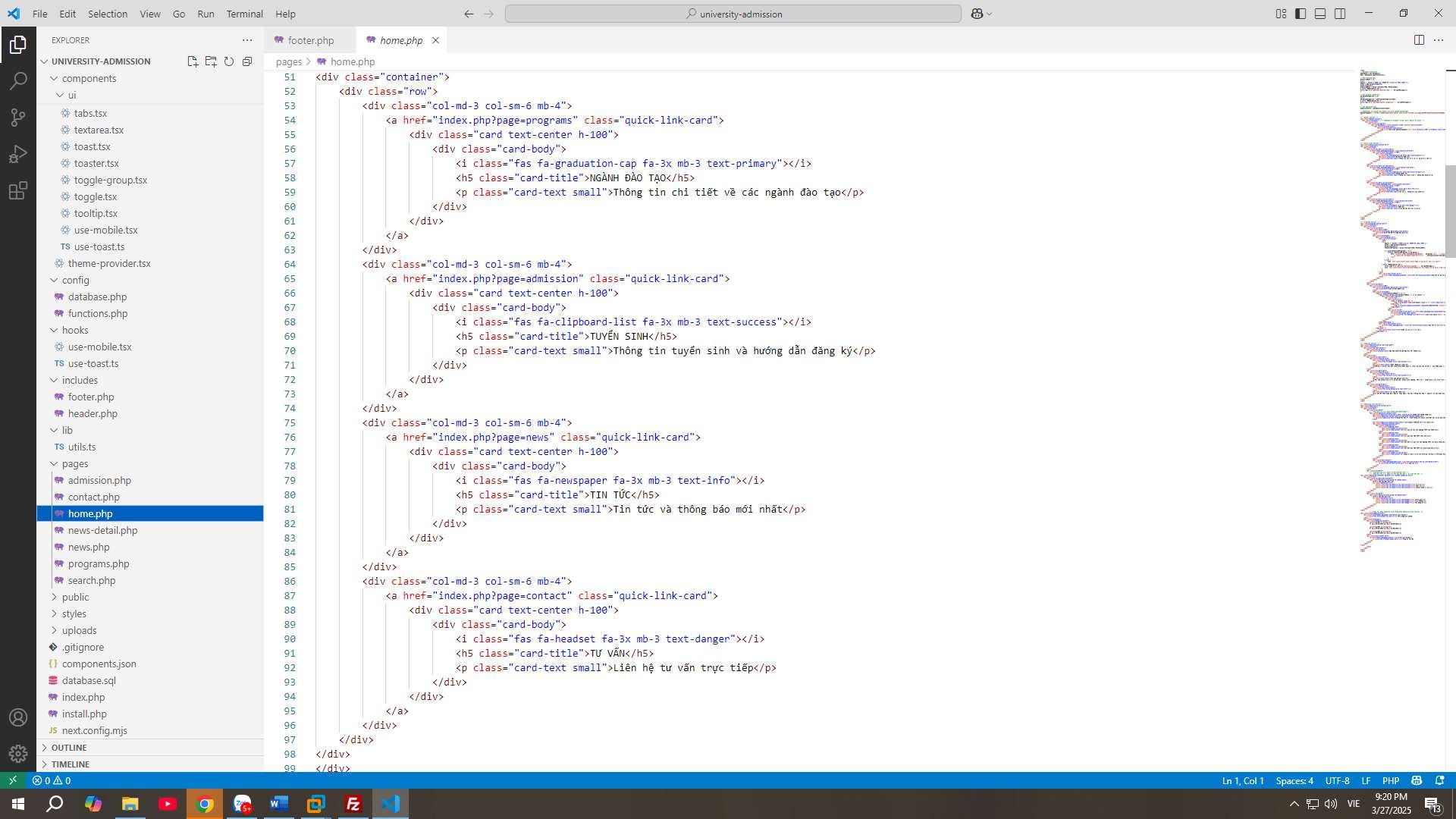
* Ngôn ngữ lập trình**:** PHP, JavaScript.
* Cơ sở dữ liệu**:** MySQL để lưu trữ thông tin.
* Công cụ thiết kế giao diện**:** HTML5, CSS3.
* Máy chủ web**:** Apache.

### Quy trình cài đặt

* + - 1. *Cài đặt môi trường phát triển:*

Cài đặt XAMPP để chạy PHP và MySQL trên máy tính cá nhân.

* + - 1. *Phát triển giao diện người dùng:*
* Thiết kế giao diện trang chủ, trang thông tin ngành học, tin tức, ...

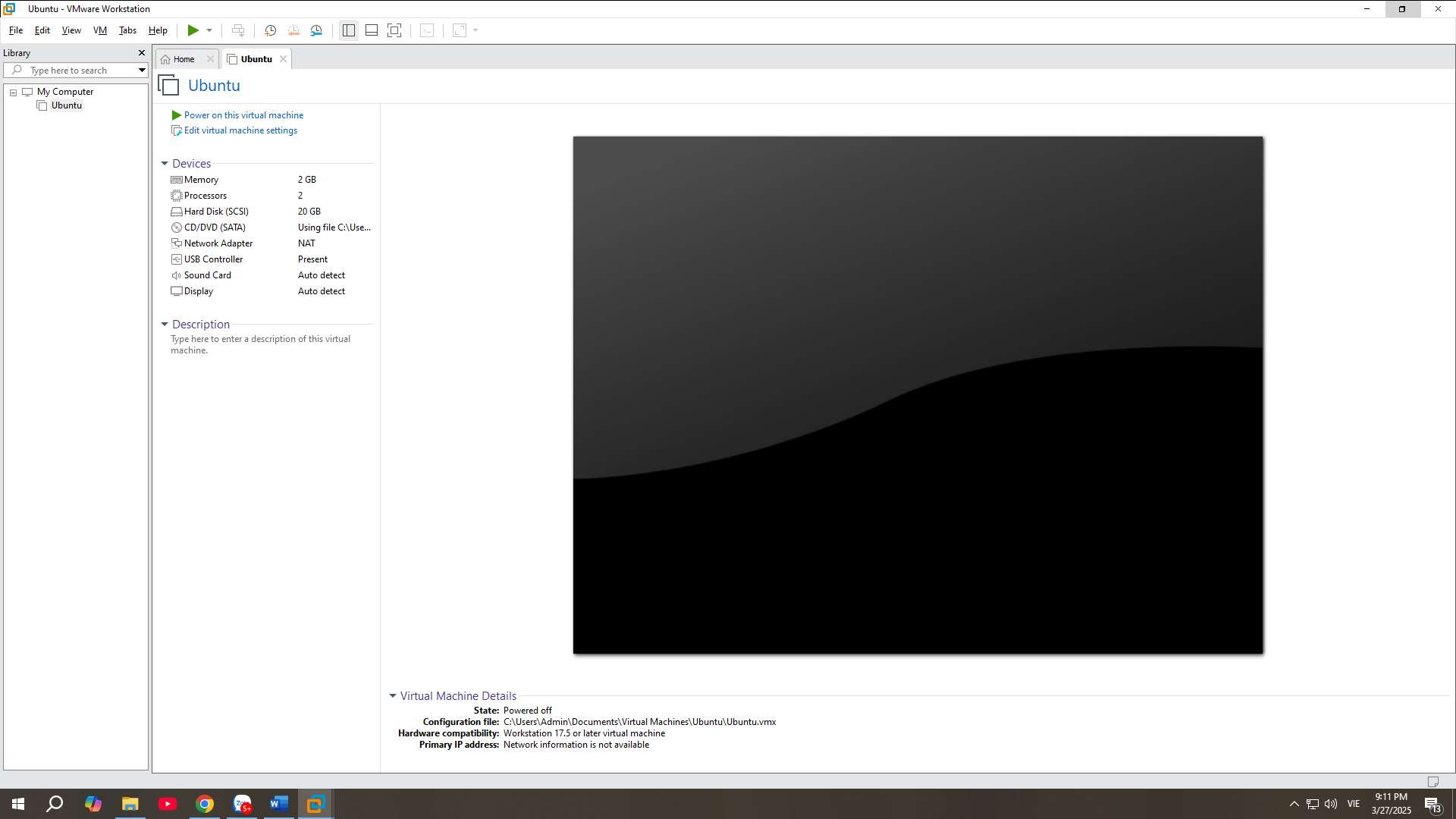


*Hình 3.11 Thiết kế giao diện*

* Sử dụng CSS và JavaScript để tối ưu hóa trải nghiệm người dùng.
  + - 1. *Tích hợp backend:*

Xây dựng các module xử lý logic như kiểm tra dữ liệu đăng ký, nhập xuất, ...

* + - 1. *Kiểm tra và sửa lỗi:*
* Thực hiện kiểm tra chức năng toàn diện, bao gồm đăng ký, tìm kiếm, và quản lý thông tin.
* Khắc phục lỗi và tối ưu hóa hiệu suất.
  + - 1. *Cài đặt và thiết lập máy chủ:*
* Chọn phần mềm ảo hóa: VMware Workstation



*Hình 3.12 Tạo máy ảo trên VMware Workstation*

* Tải xuống và cài đặt Ubuntu Server:

+ Tải file ISO Ubuntu Server từ trang chủ Ubuntu.

+ Tạo máy ảo mới và cài đặt Ubuntu Server từ file ISO.

* Cấu hình mạng: Đảm bảo máy ảo có thể kết nối internet.
* Cập nhật hệ thống:

sudo apt update && sudo apt upgrade -y

* Cài đặt các thành phần cần thiết:

+ Máy chủ web (Apache):

sudo apt install apache2 -y

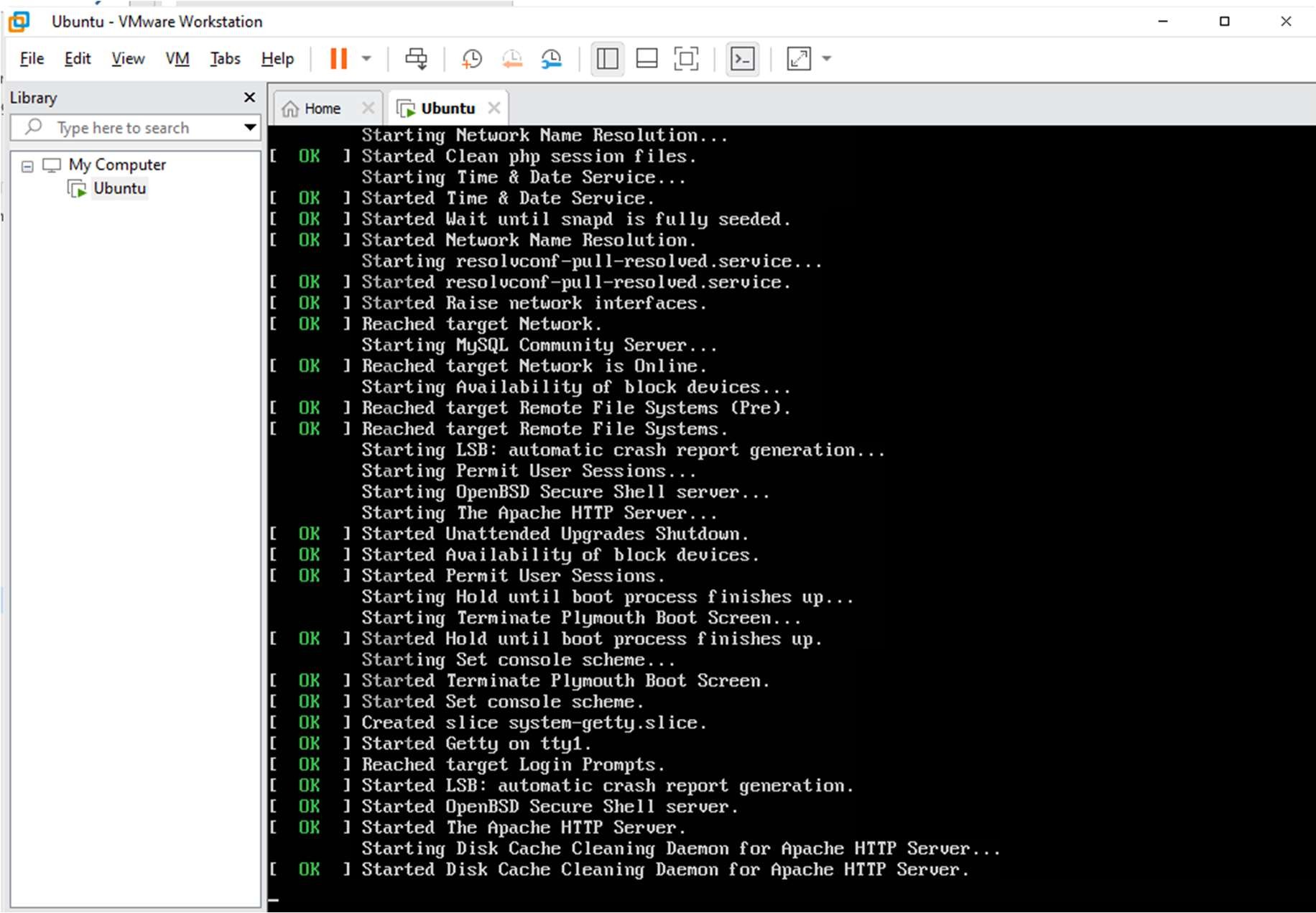
+ Hệ quản trị cơ sở dữ liệu (MySQL/MariaDB):

sudo apt install mysql-server -y

+ Ngôn ngữ lập trình PHP:

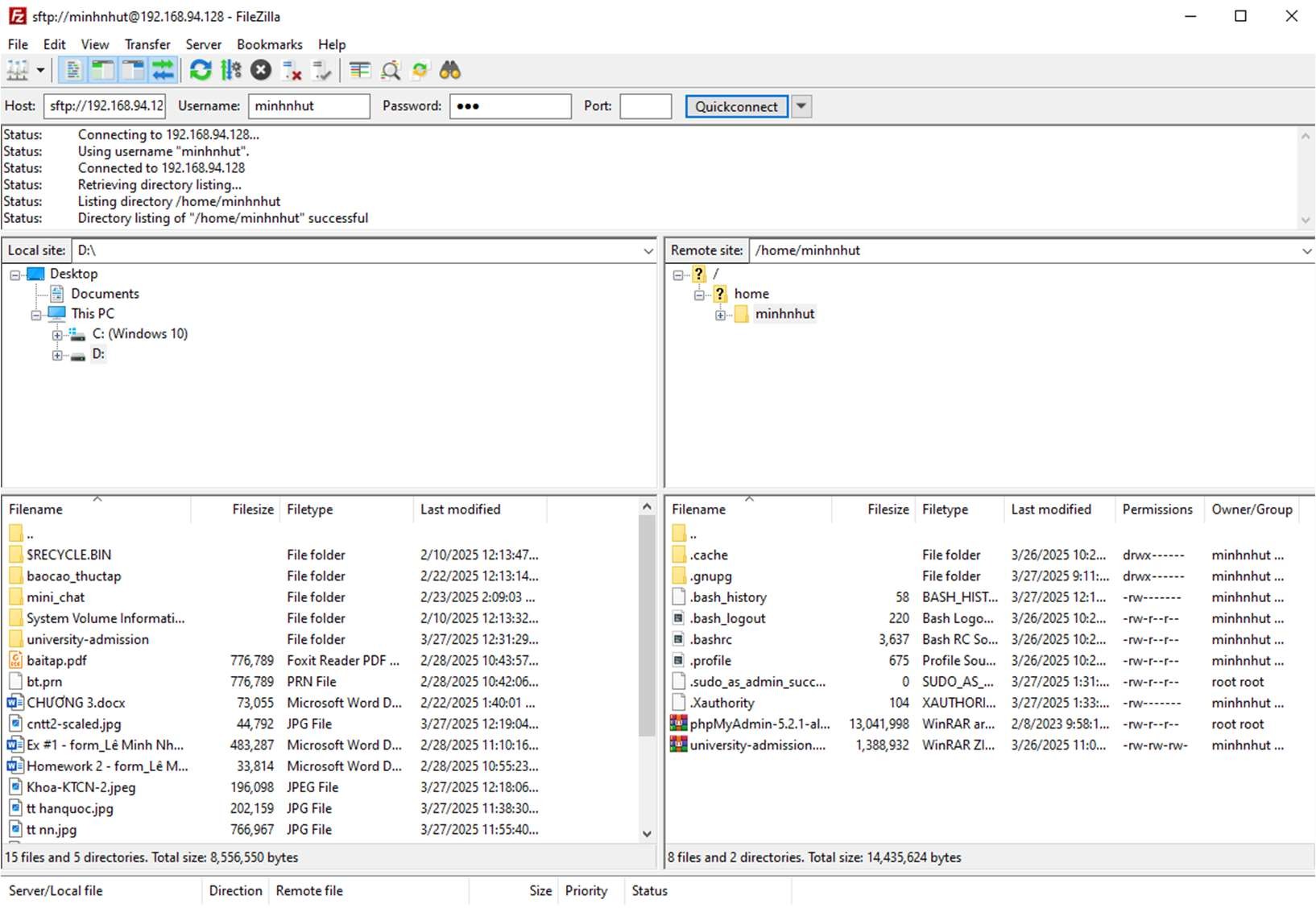
sudo apt install php libapache2-mod-php php-mysql -y

* Các thư viện và phần mềm bổ sung.
* Cài đặt các thư viện PHP cần thiết.



*Hình 3.13 Thiết lập máy ảo*

* + - 1. *Triển khai:*
* Sao chép mã nguồn ứng dụng vào máy ảo bằng FileZilla

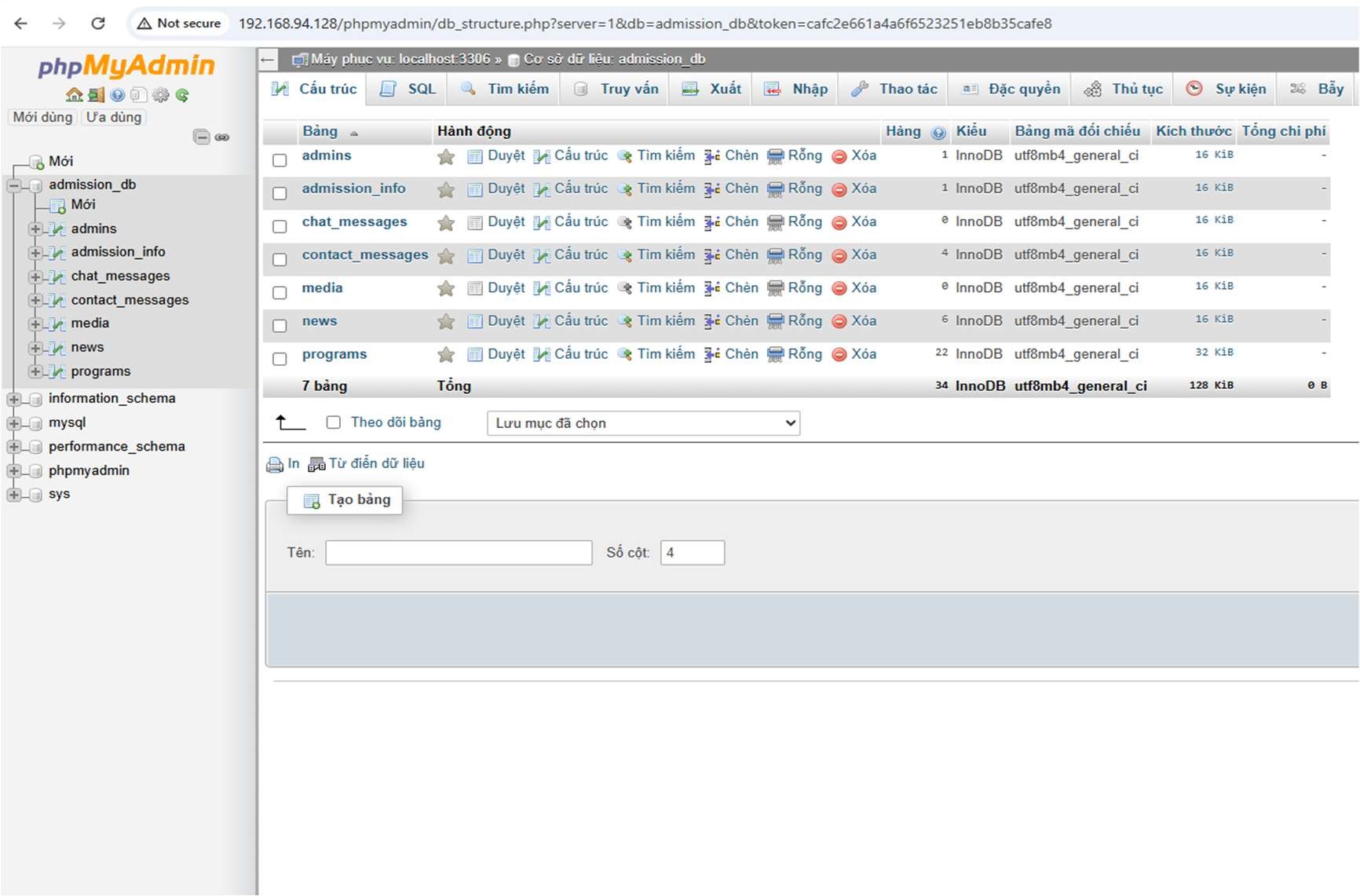


*Hình 3.14 Sao chép mã nguồn ứng dụng vào máy ảo bằng FileZilla*

* Cấu hình cơ sở dữ liệu:

+ Tạo cơ sở dữ liệu và người dùng cho ứng dụng.

+ Nhập dữ liệu (nếu cần).



*Hình 3.15 Nhập cơ sở dữ liệu*

* Cấu hình ứng dụng: Cập nhật các thông tin kết nối cơ sở dữ liệu và các cấu hình khác.

### 3.5 Kết luận

Chương này đã trình bày toàn bộ quá trình hiện thực hóa nghiên cứu, từ phân tích nhu cầu đến thiết kế và triển khai hệ thống. Dự án được xây dựng dựa trên các công nghệ hiện đại và quy trình phát triển phần mềm chuyên nghiệp, đảm bảo hệ thống website hoạt động hiệu quả, đáp ứng tốt các yêu cầu chức năng và phi chức năng.

Trong các chương tiếp theo, các kết quả kiểm thử và đánh giá hiệu quả của hệ thống sẽ được trình bày chi tiết.

# CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

### Giới thiệu

Chương này trình bày các kết quả đạt được sau quá trình thực hiện đồ án "Xây dựng website quảng bá và tuyển sinh". Các kết quả được đánh giá theo các tiêu chí chính: hiệu năng hệ thống, tính thân thiện với người dùng và sự hoàn chỉnh của các giao diện chức năng. Ngoài ra, chương này cũng mô tả cụ thể các giao diện chính của website và các tính năng đã được triển khai thành công.

### Hiệu năng hệ thống

* Trang chủ tải trong khoảng 1-2 giây với kết nối internet trung bình.
* Các trang ngành học, đăng ký tư vấn tuyển sinh tải trong khoảng 1-2 giây.
* Xử lý lỗi**:** Các lỗi không mong muốn (như nhập sai dữ liệu, kết nối thất bại) đều được xử lý và thông báo rõ ràng đến người dùng.
* Bố cục rõ ràng**:** Trang chủ trình bày các mục chính như thông tin tuyển sinh, ngành học và các chức năng quan trọng một cách dễ dàng tiếp cận.
* Ngôn ngữ và biểu tượng dễ hiểu**:** Nội dung được trình bày bằng ngôn ngữ tự nhiên, biểu tượng trực quan giúp người dùng dễ dàng thao tác mà không cần hướng dẫn phức tạp.

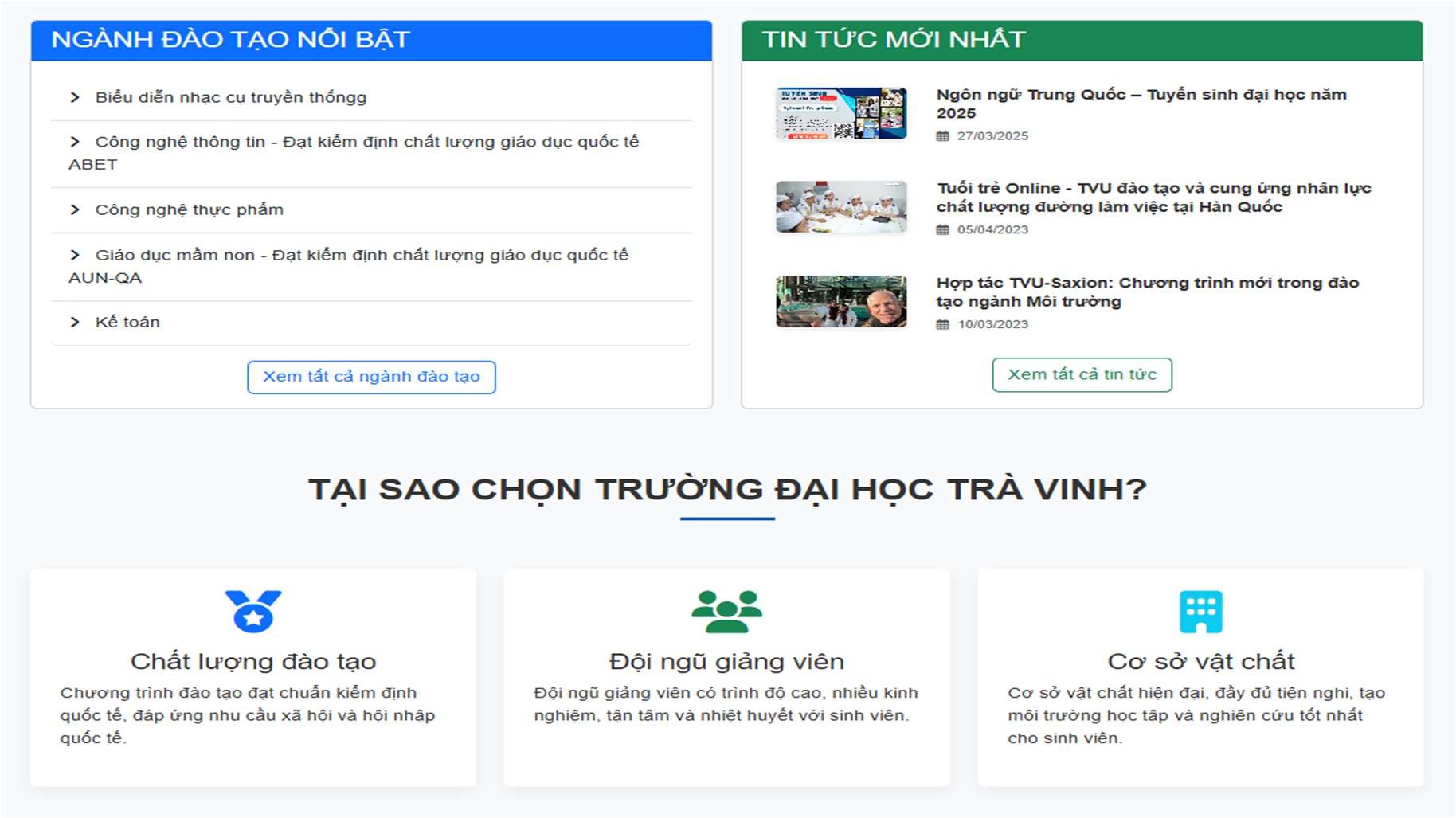
### Các giao diện chức năng

### Giao diện người dùng

* + - 1. *Giao diện trang chủ*

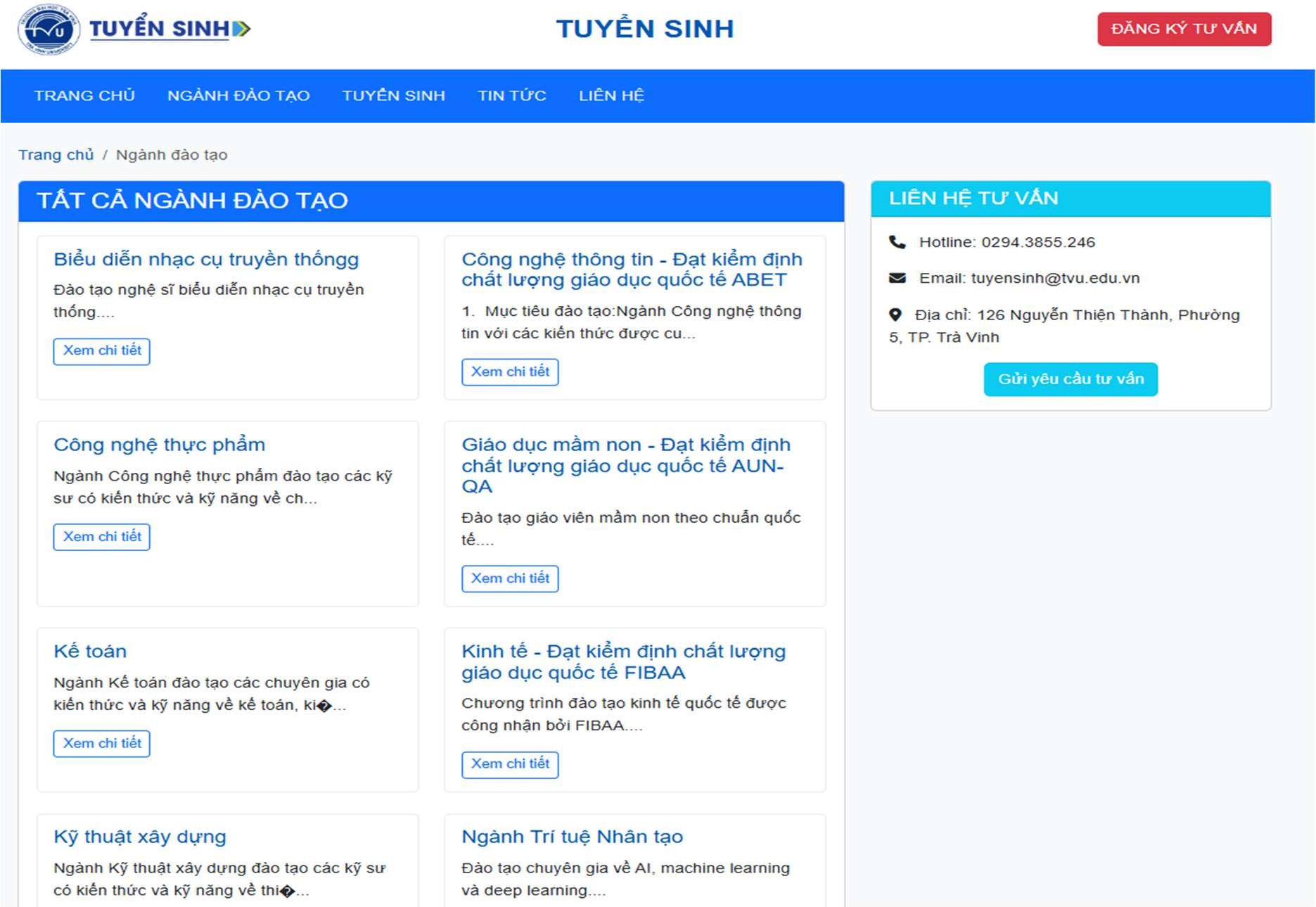
**

*Hình 4.1 Giao diện trang chủ*

**

*Hình 4.2 Giao diện trang chủ*

* + - 1. *Giao diện ngành học*

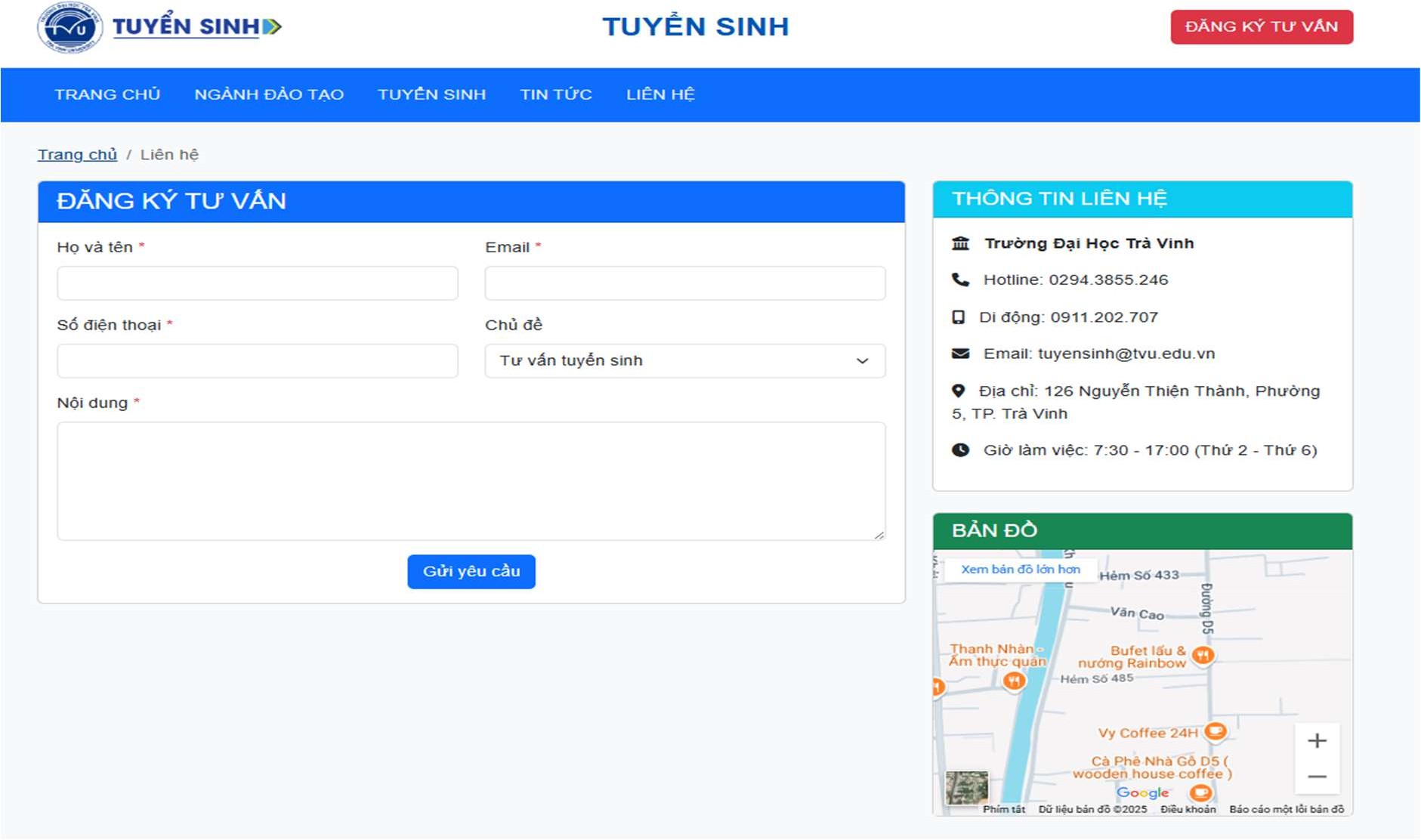
**

*Hình 4.3 Giao diện ngành học*

**

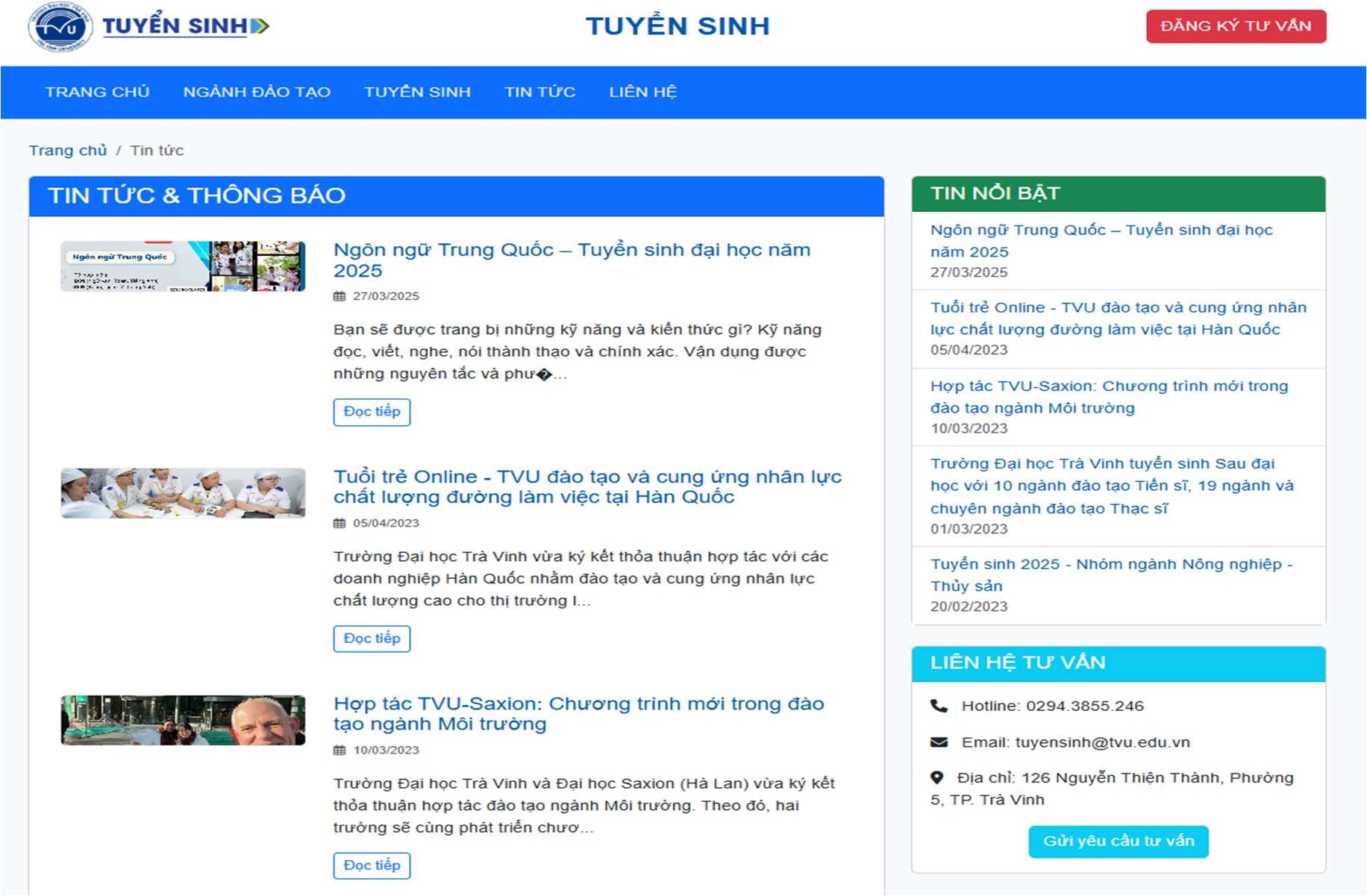
*Hình 4.4 Giao diện xem ngành học chi tiết*

* + - 1. *Giao diện đăng ký tư vấn tuyển sinh*

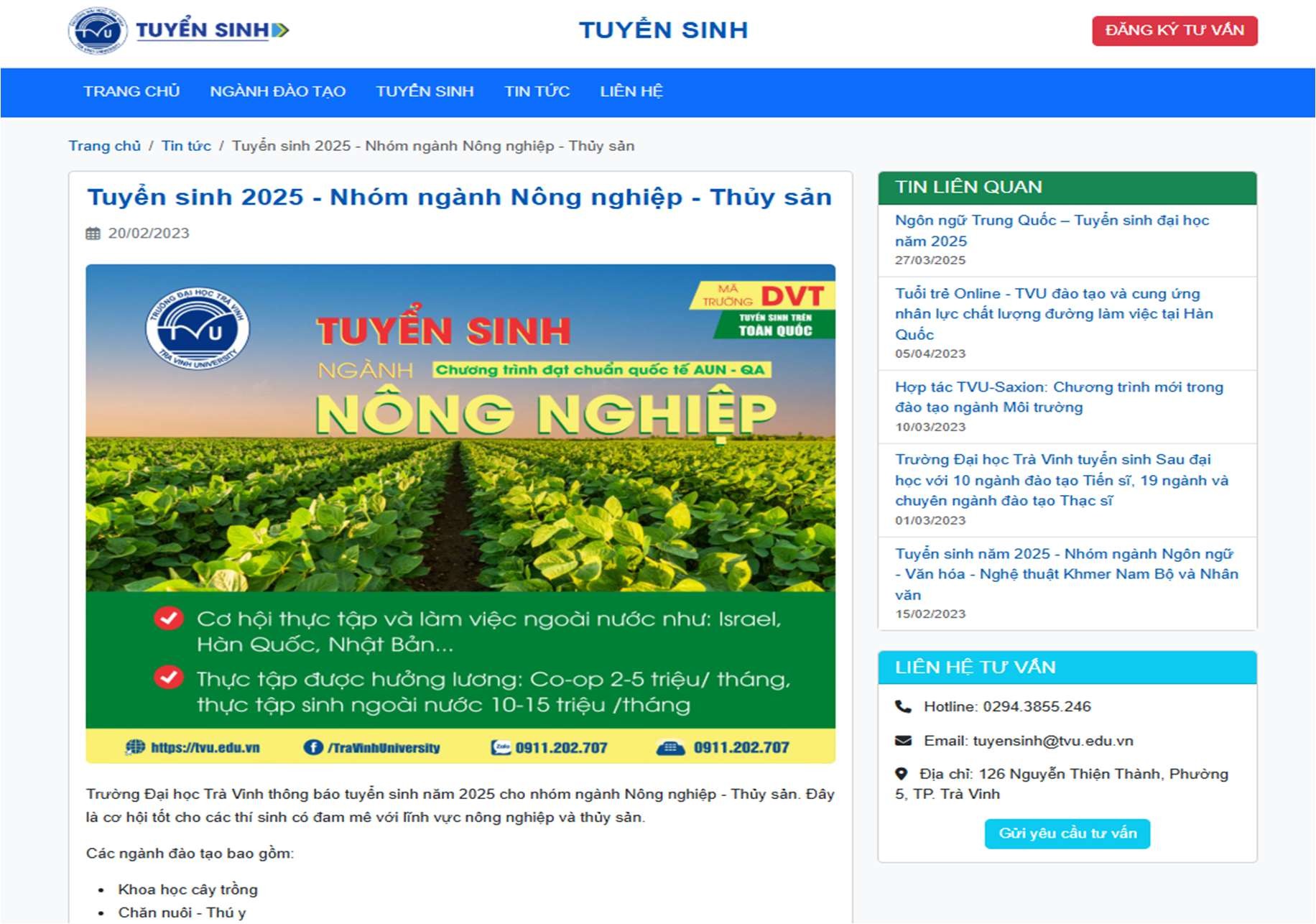
**

*Hình 4.5 Giao diện đăng ký tư vấn tuyển sinh*

* + - 1. *Giao diện tin tức*

**

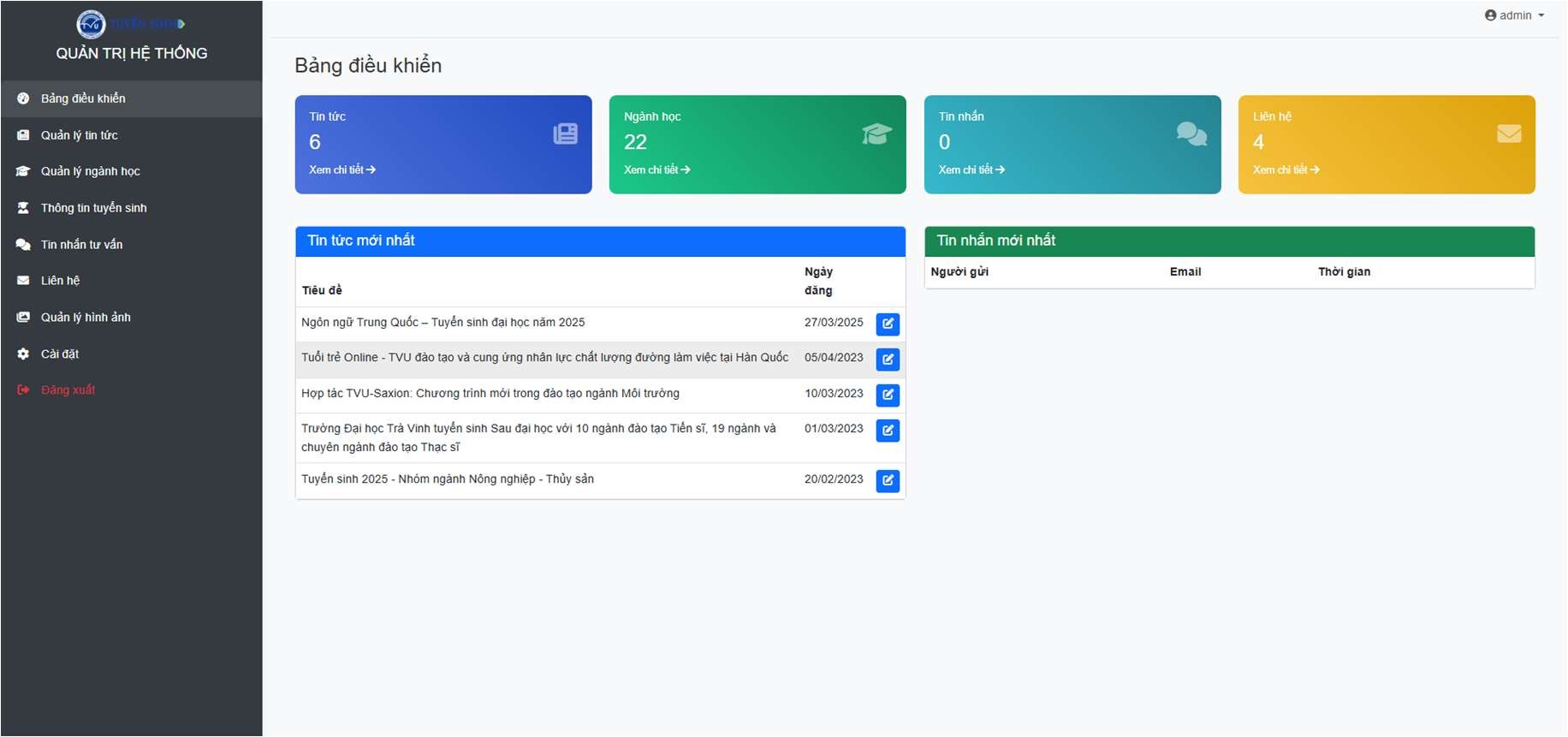
*Hình 4.6 Giao diện tin tức*



*Hình 4.7 Giao diện xem tin tức chi tiết*

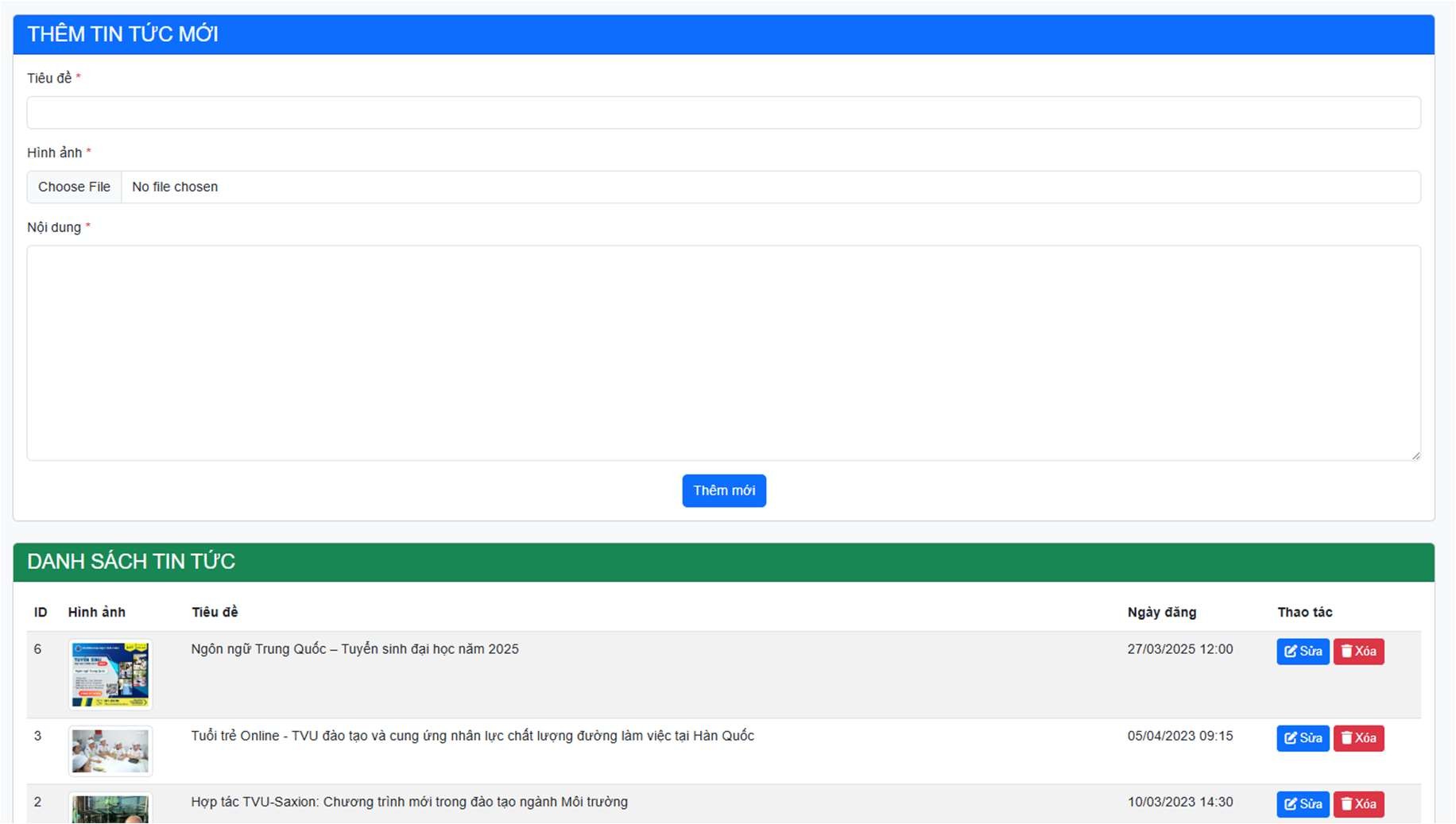
### Giao diện Admin

* + - 1. *Giao diện chính*

**

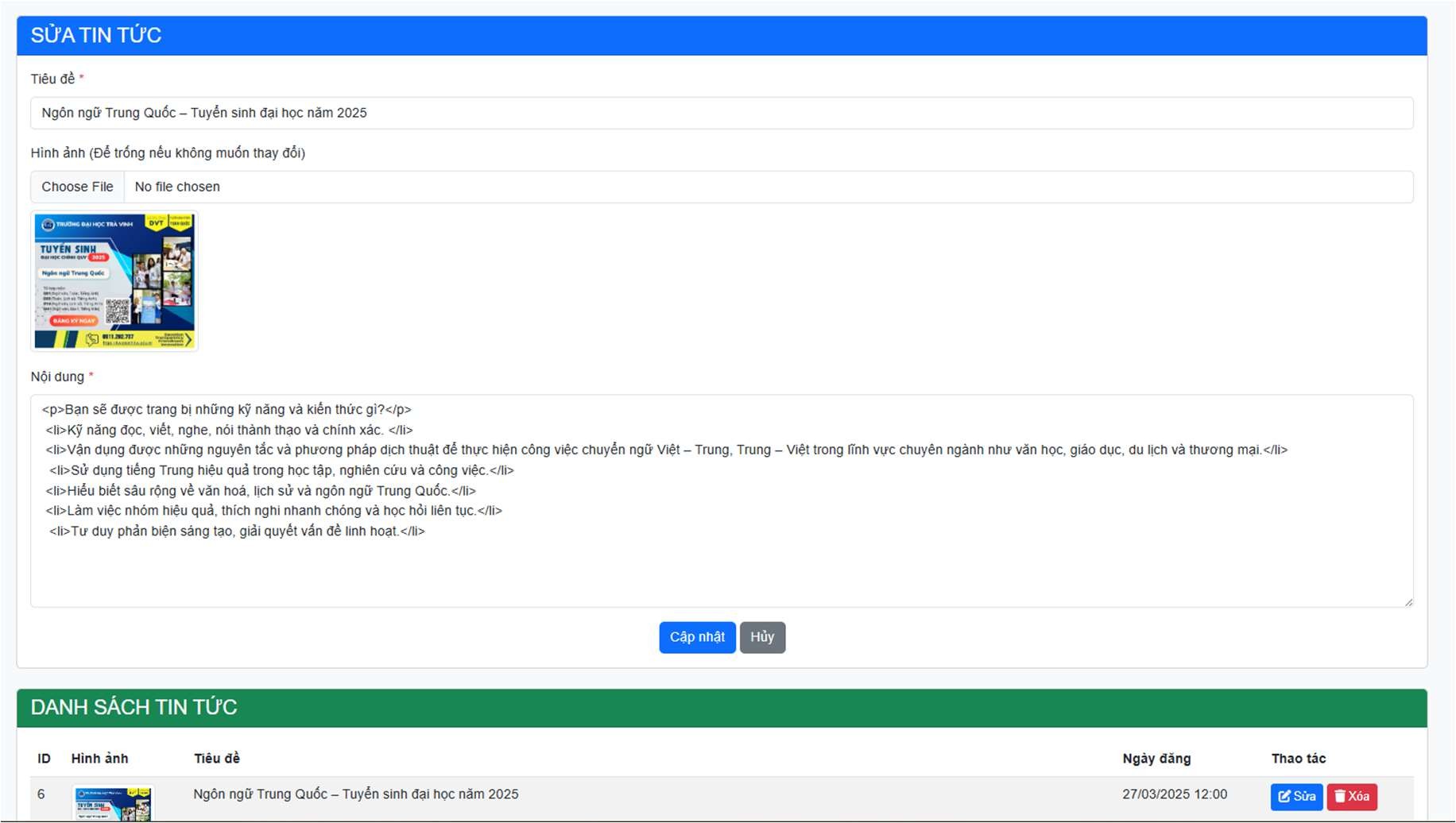
*Hình 4.8 Giao diện admin chính*

* + - 1. *Giao diện quản lý tin tức*
* *Thêm tin tức:*

**

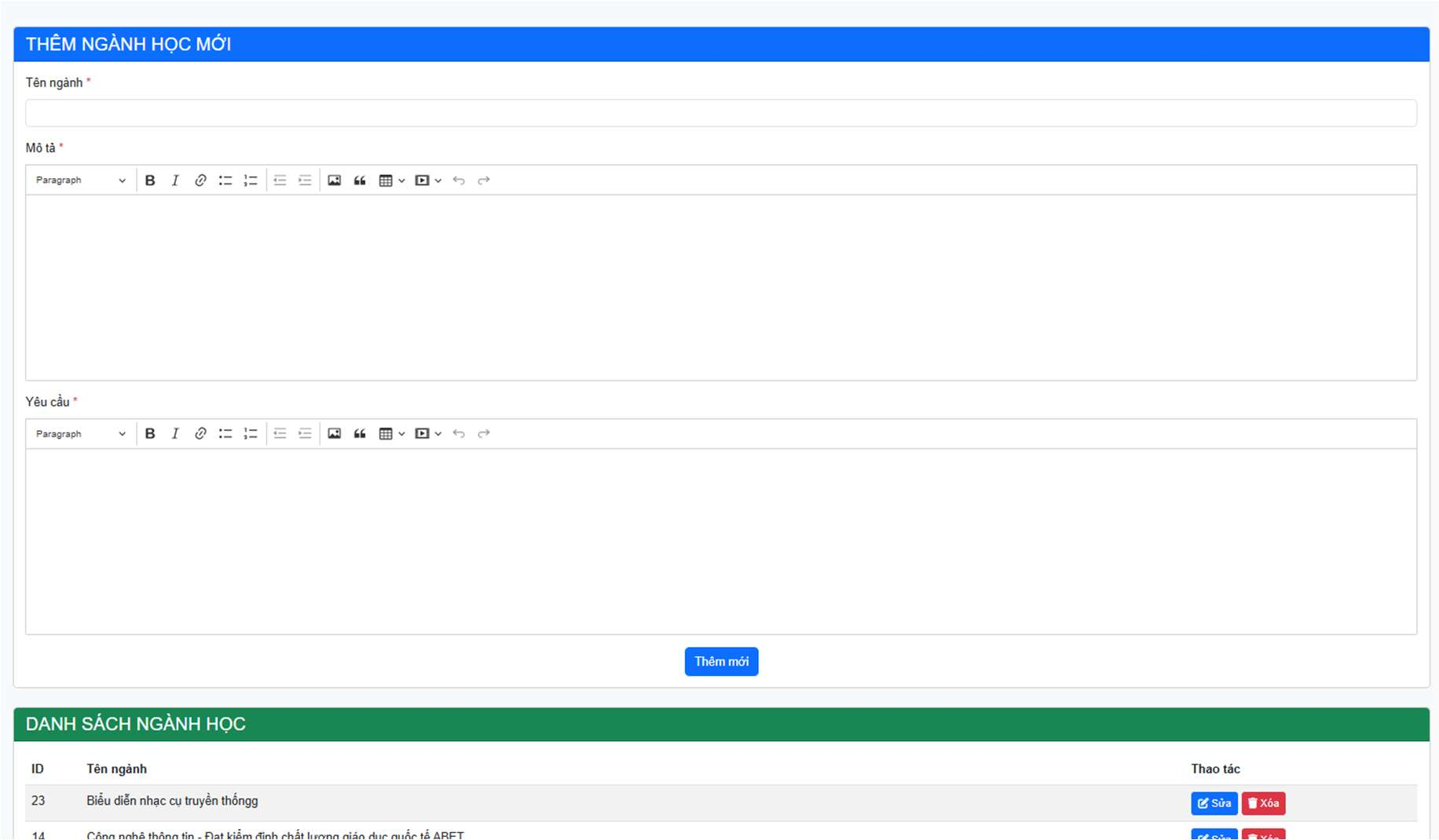
*Hình 4.9 Giao diện thêm tin tức*

* *Sửa tin tức:*

**

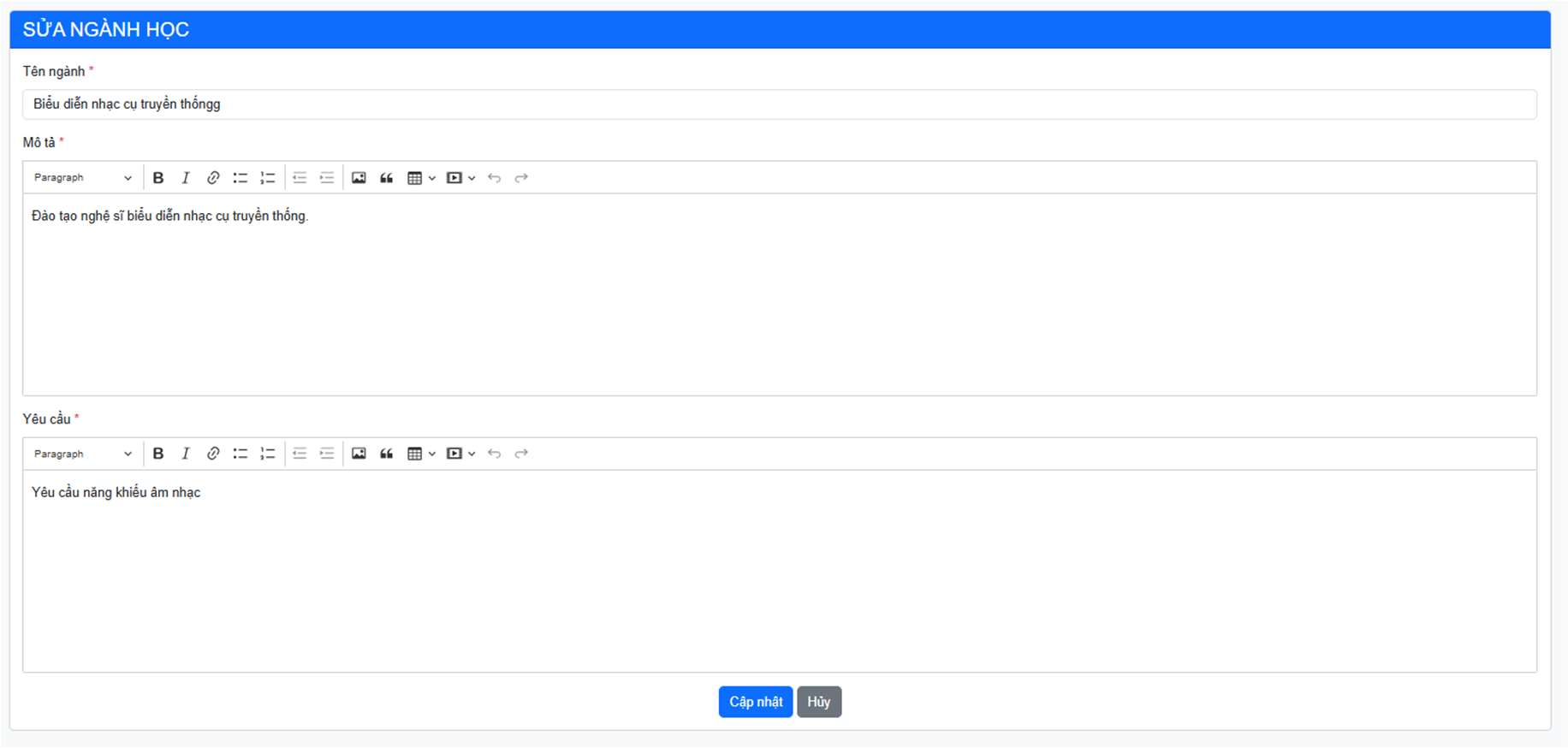
*Hình 4.10 Giao diện sửa tin tức*

* + - 1. *Giao diện quản lý ngành học*
* *Thêm ngành học:*



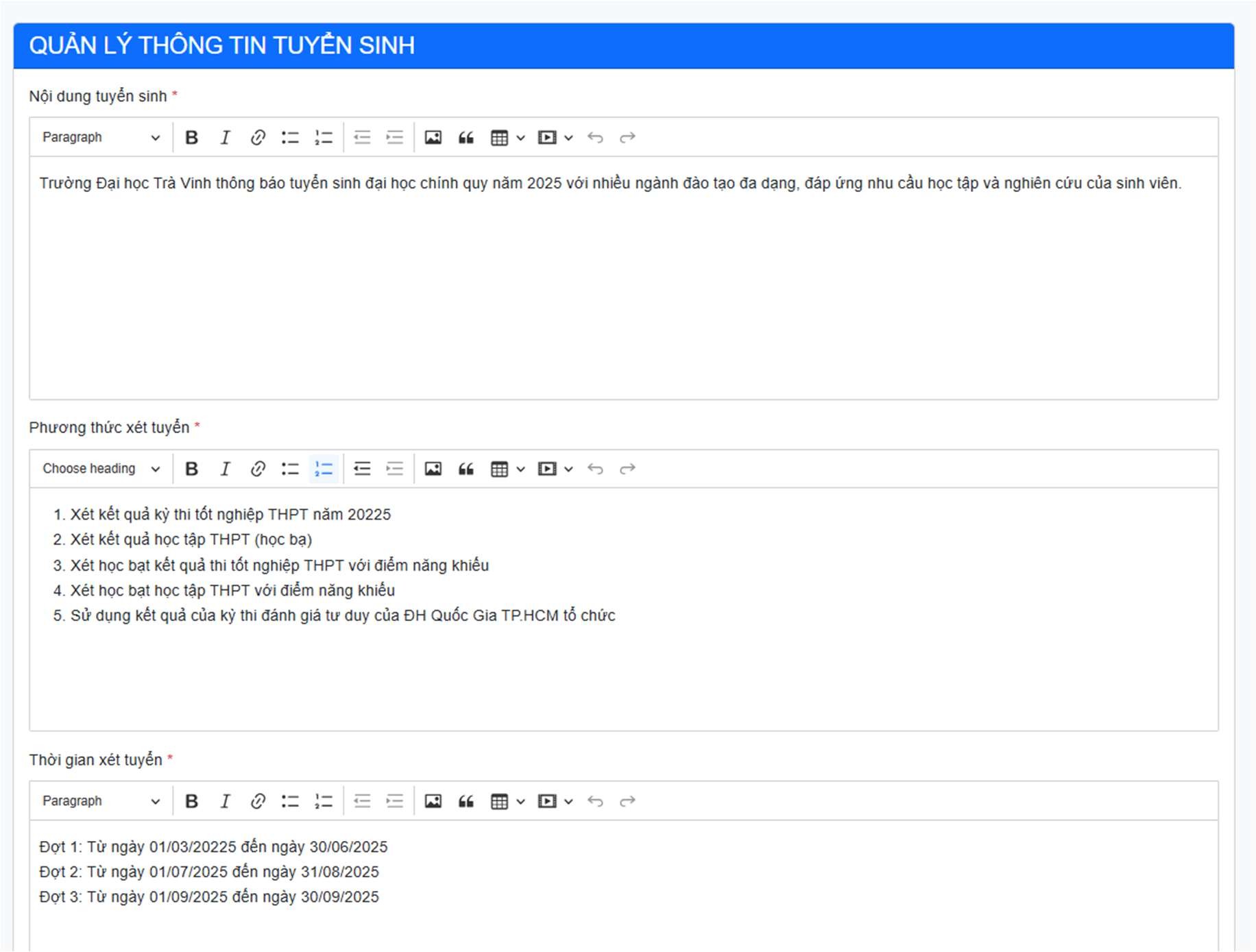
*Hình 4.11 Giao diện thêm ngành học*

* *Sửa ngành học:*

**

*Hình 4.12 Giao diện sửa ngành học*

* + - 1. *Giao diện quản lý thông tin tuyển sinh*

**

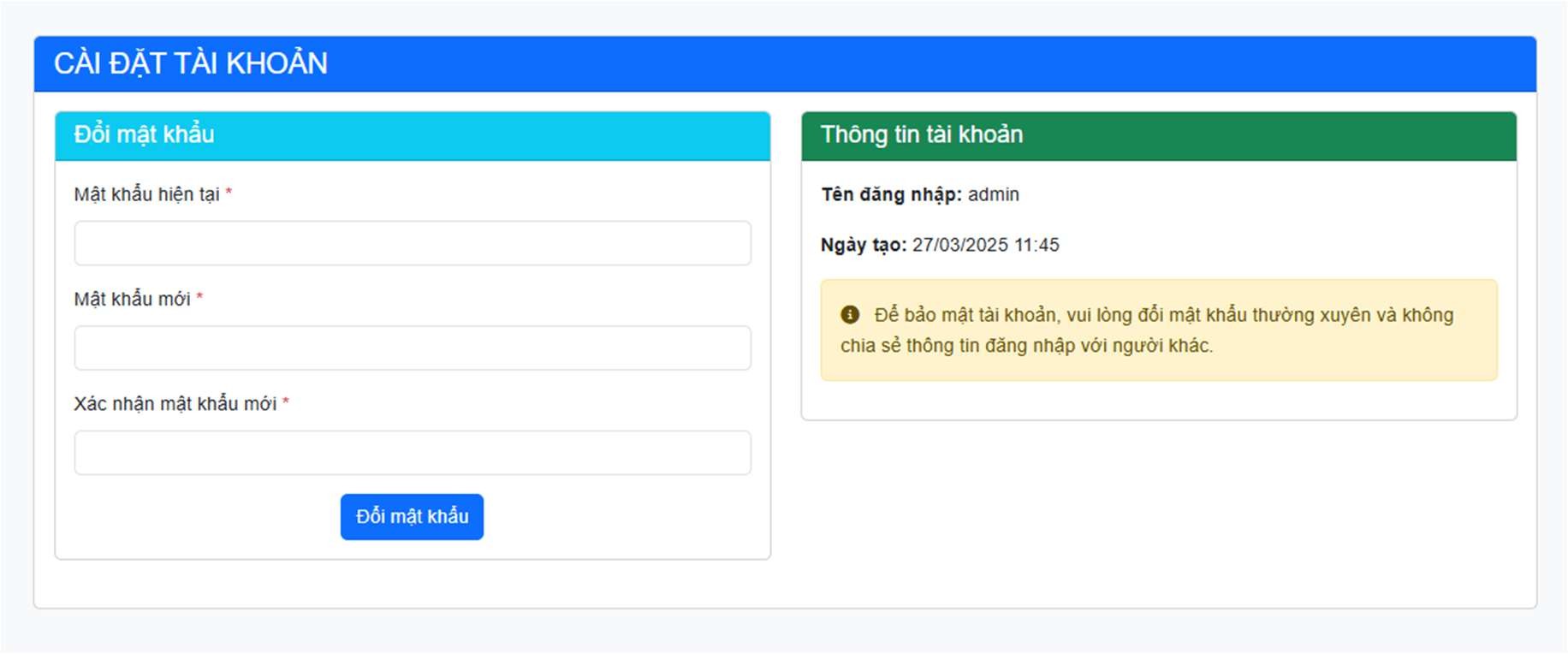
*Hình 4.13 Giao diện quản lý thông tin tuyển sinh*

* + - 1. *Giao diện quản lý tin nhắn liên hệ tư vấn tuyển sinh*

**

*Hình 4.14 Giao diện quản lý tin nhắn liên hệ tư vấn tuyển sinh*

* + - 1. *Giao diện quản lý tài khoản Admin*

**

*Hình 4.15 Giao diện quản lý tài khoản Admin*

### Đánh giá chung

### Thành công của hệ thống

Dự án đã đạt được các mục tiêu đề ra ban đầu, bao gồm:

* + - * Xây dựng website với giao diện hiện đại, dễ sử dụng.
      * Hệ thống đăng ký trực tuyến hoạt động ổn định, hỗ trợ quản lý hiệu quả.
      * Bảo mật thông tin người dùng và dữ liệu hệ thống được đảm bảo.

### Hạn chế và hướng cải thiện

* + - * Hạn chế**:**

+ Hiện tại hệ thống chỉ hỗ trợ ngôn ngữ tiếng Việt, chưa tích hợp đa ngôn ngữ.

+ Chưa có chức năng thanh toán trực tuyến cho các khoản phí tuyển sinh.

* + - * Hướng cải thiện**:**

+ Tích hợp thêm tính năng đa ngôn ngữ để mở rộng đối tượng sử dụng.

+ Phát triển module thanh toán trực tuyến để tăng tính tiện lợi cho người dùng.

### Kết luận

Kết quả nghiên cứu và thực hiện dự án đã chứng minh tính khả thi và hiệu quả của việc ứng dụng công nghệ vào quy trình tuyển sinh trực tuyến. Website không chỉ đáp ứng được nhu cầu quảng bá thông tin mà còn tạo ra một kênh tương tác trực tiếp, tiện lợi giữa nhà trường và thí sinh.

Trong tương lai, việc cải thiện và mở rộng các chức năng sẽ giúp hệ thống ngày càng hoàn thiện, đóng góp tích cực vào việc nâng cao chất lượng dịch vụ giáo dục và tuyển sinh.

# CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### Kết luận

Sau quá trình nghiên cứu và thực hiện, dự án "Xây dựng website quảng bá và tuyển sinh trường đại học" đã đạt được các kết quả sau:

### Kết quả đạt được:

* Xây dựng thành công hệ thống website tuyển sinh trực tuyến với giao diện thân thiện, hiện đại và dễ sử dụng.
* Cung cấp các chức năng chính như quản lý thông tin tuyển sinh, đăng ký trực tuyến và tìm kiếm ngành học, đáp ứng tốt các nhu cầu của thí sinh và quản trị viên.
* Đảm bảo hiệu năng cao, xử lý đồng thời nhiều yêu cầu và duy trì tính ổn định trong môi trường sử dụng thực tế.
* Bảo mật dữ liệu người dùng và thông tin hệ thống được tối ưu, giảm thiểu các rủi ro xâm nhập.

### Đề xuất mới:

* Xây dựng cơ chế phân tích dữ liệu đăng ký để hỗ trợ nhà trường trong việc đưa ra các chiến lược tuyển sinh phù hợp.
* Phát triển các tính năng hỗ trợ thí sinh như công cụ tư vấn chọn ngành học hoặc các gợi ý định hướng nghề nghiệp.

### Hướng phát triển

* Tích hợp đa ngôn ngữ**:** Phát triển hệ thống hỗ trợ nhiều ngôn ngữ để phục vụ thí sinh quốc tế, mở rộng phạm vi tuyển sinh.
* Thanh toán trực tuyến: Triển khai chức năng thanh toán học phí hoặc các khoản phí tuyển sinh trực tuyến để tăng tính tiện lợi.
* Ứng dụng di động**:** Phát triển phiên bản ứng dụng di động nhằm tối ưu hóa trải nghiệm người dùng trên nền tảng di động.
* Phân tích dữ liệu nâng cao: Tích hợp các công cụ phân tích dữ liệu để hỗ trợ quản trị viên trong việc theo dõi xu hướng đăng ký và dự báo nhu cầu tuyển sinh.
* Tăng cường bảo mật**:** Nâng cao các biện pháp bảo mật dữ liệu, như xác thực hai yếu tố (2FA), mã hóa cấp cao, và giám sát an ninh mạng.
* Kết nối với mạng xã hội**:** Hỗ trợ đăng nhập qua mạng xã hội và chia sẻ thông tin ngành học trên các nền tảng phổ biến như Facebook, LinkedIn để tăng khả năng tiếp cận.
* Hỗ trợ AI**:** Ứng dụng trí tuệ nhân tạo trong việc tư vấn tuyển sinh, gợi ý ngành học dựa trên sở thích và năng lực của thí sinh.