

Trường Đại học Công nghiệp Hà Nội

Khoa Công nghệ thông tin

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO THỰC TẬP**

**Đề tài:** Xây dựng website quản lý cửa hàng Áo Cưới Thủy Linh

|  |  |
| --- | --- |
| Giáo viên : | TS. Phạm Văn Hiệp |
| Họ tên : | Nguyễn Văn Quang |
| Mã sinh viên: | 1141260055 |
| Lớp: | HTTT1-K11 |

.

Hà Nội ngày 14 tháng 07 năm 2022

LỜI NÓI ĐẦU

Hiện nay, cửa hàng Thủy Linh hoạt động dựa ???. Em mong rằng với sự ra đời của website này sẽ được ngừơi dùng đón nhận bởi sự tiện lợi mà nó mang lại cho người dùng.

Em xin gửi lời cảm ơn tới thầy Phạm Văn Hiệp đã chỉ bảo và hướng dẫn tận tình cho em trong suốt quá trình thực tập để em có thể hoàn thành đề tài.

**Em xin chân thành cảm ơn!**

# MỤC LỤC

[MỤC LỤC 2](#_Toc109328995)

[CHƯƠNG 1: TỔNG QUAN 9](#_Toc109328996)

[**1.1** **Giới thiệu đề tài** 9](#_Toc109328997)

[**1.1.2** **Lý do chọn đề tài** 10](#_Toc109328998)

[**1.1.3** **Mục tiêu sản phẩm** 11](#_Toc109328999)

[**1.1.4** **Yêu cầu công nghệ** 11](#_Toc109329000)

[**1.1.5** **Hoạt động hệ thống** 11](#_Toc109329001)

[**1.2** **Công nghệ sử dụng** 12](#_Toc109329002)

[**1.2.1** **Giới thiệu về Java** 12](#_Toc109329003)

[**1.2.2** **Giới thiệu về Spring Framework** 16](#_Toc109329004)

[**1.2.3** **Giới thiệu về Spring Boot** 22](#_Toc109329005)

[CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG 30](#_Toc109329006)

[**2.1** **Phân tích biểu đồ usecase** 30](#_Toc109329007)

[**2.1.1** **Xác định các tác nhân** 30](#_Toc109329008)

[**2.1.2** **Xác định usecase** 30](#_Toc109329009)

[**2.1.3** **Biểu đồ usecase** 31](#_Toc109329010)

[**2.2** **Đặc tả usecase** 34](#_Toc109329011)

[**2.2.1** **Usecase xem trang chủ** 34](#_Toc109329012)

[**2.2.2** **Usecase xem khóa học** 35](#_Toc109329013)

[**2.2.3** **Usecase đăng kí khóa học** 36](#_Toc109329014)

[**2.2.4** **Usecase xem danh sách giảng viên** 36](#_Toc109329015)

[**2.2.5** **Usecase xem thông tin liên hệ** 37](#_Toc109329016)

[**2.2.6** **Usecase xem danh sách hồ sơ** 37](#_Toc109329017)

[**2.2.7** **Usecase tìm kiếm hồ sơ** 38](#_Toc109329018)

[**2.2.8** **Usecase xóa hồ sơ** 39](#_Toc109329019)

[**2.2.9** **Usecase xem danh sách khóa học** 40](#_Toc109329020)

[**2.2.10** **Usecase xem tìm kiếm khóa học** 40](#_Toc109329021)

[**2.2.11** **Usecase thêm khóa học** 41](#_Toc109329022)

[**2.2.12** **Usecase sửa khóa học** 42](#_Toc109329023)

[**2.2.13** **Usecase xóa khóa học** 43](#_Toc109329024)

[**2.2.14** **Usecase xem danh sách giảng viên** 44](#_Toc109329025)

[**2.2.15** **Usecase tìm kiếm giảng viên** 44](#_Toc109329026)

[**2.2.16** **Usecase thêm giảng viên** 45](#_Toc109329027)

[**2.2.17** **Usecase xóa giảng viên** 46](#_Toc109329028)

[**2.3** **Biểu đồ tuần tự** 47](#_Toc109329029)

[**2.3.1** **Biểu đồ tuần tự xem khóa học** 47](#_Toc109329030)

[**2.3.2** **Biểu đồ tuần tự đăng kí khóa học** 48](#_Toc109329031)

[**2.3.3** **Biểu đồ tuần tự xem danh sách giảng viên** 49](#_Toc109329032)

[**2.3.4** **Biểu đồ tuần tự đăng nhập** 50](#_Toc109329033)

[**2.3.5** **Biểu đồ tuần tự quản lí hồ sơ** 52](#_Toc109329034)

[**2.3.6** **Biểu đồ tuần tự quản lí bài viết** 53](#_Toc109329035)

[**2.3.7** **Biểu đồ tuần tự quản lí giáo viên** 54](#_Toc109329036)

[**2.3.8** **Biểu đồ tuần tự quản lí người dùng** 55](#_Toc109329037)

[**2.3.9** **Biểu đồ tuần tự quản lí quyền** 56](#_Toc109329038)

[**2.4** **Biểu đồ lớp** 57](#_Toc109329039)

[**2.4.1** **Biểu đồ lớp** 57](#_Toc109329040)

[**2.4.2** **Xác định các lớp** 58](#_Toc109329041)

[CHƯƠNG 3: THIẾT KẾ HỆ THỐNG 67](#_Toc109329042)

[**3.1** **Thiết kế giao diện** 67](#_Toc109329043)

[**3.2.1** **Giao diện trang đăng nhập** 67](#_Toc109329044)

[**3.2.2** **Giao diện trang chủ web** 68](#_Toc109329045)

[**3.2.3** **Giao diện trang đăng kí khóa học web** 68](#_Toc109329046)

[68](#_Toc109329047)

[**3.2.4** **Giao diện danh sách khóa học web** 69](#_Toc109329048)

[69](#_Toc109329049)

[**3.2.5** **Giao diện chi tiết khóa học web** 69](#_Toc109329050)

[**3.2.6** **Giao diện trang chủ quản trị** 70](#_Toc109329051)

[**3.2.7** **Giao diện trang quản lí khóa học** 70](#_Toc109329052)

[**3.2.8** **Giao diện thêm mới khóa học** 71](#_Toc109329053)

[**3.2.9** **Giao diện cập nhật thông tin khóa học** 71](#_Toc109329054)

[**3.2.10** **Giao diện chi tiết khóa học** 72](#_Toc109329055)

[72](#_Toc109329056)

[**3.2.11** **Giao diện quản lí thông tin đăng kí** 72](#_Toc109329057)

[**3.2.12** **Giao diện thông tin đăng kí chờ duyệt** 73](#_Toc109329058)

[73](#_Toc109329059)

[**3.2.13** **Giao diện thông tin đã chấp nhận** 73](#_Toc109329060)

[73](#_Toc109329061)

[**3.2.14** **Giao diện thông tin đăng kí đã hủy** 74](#_Toc109329062)

[74](#_Toc109329063)

[**3.2.15** **Giao diện trang thông tin chi tiết đăng kí** 74](#_Toc109329064)

[**3.2.16** **Giao diện quản lí giảng viên** 75](#_Toc109329065)

[**3.2.17** **Giao diện thêm mới giảng viên** 75](#_Toc109329066)

[**3.2.18** **Giao diện cập nhập thông tin giảng viên** 76](#_Toc109329067)

[**3.2.19** **Giao diện thông tin chi tiết giảng viên** 76](#_Toc109329068)

[76](#_Toc109329069)

[**3.2.20** **Giao diện quản lí người dùng** 77](#_Toc109329070)

[**3.2.21** **Giao diện thêm mới người dùng** 77](#_Toc109329071)

[**3.2.22** **Giao diện trang cập nhập thông tin người dùng** 78](#_Toc109329072)

[78](#_Toc109329073)

[**3.2.23** **Giao diện trang thông tin chi tiết người dùng** 78](#_Toc109329074)

[78](#_Toc109329075)

[KẾT LUẬN 79](#_Toc109329076)

[TÀI LIỆU THAM KHẢO 80](#_Toc109329077)

# CHƯƠNG 1: TỔNG QUAN

1. **Giới thiệu đề tài**
2. **Tên đề tài**

Xây dựng website tuyển sinh cho trung tâm hợp tác quốc tế trường Đại Học Công Nghiệp Hà Nội, ứng dụng Java Springboot, ReactJS.

1. **Lý do chọn đề tài**
2. Nhu cầu thực tế

* Hàng năm, lượng người nước ngoài đến sinh sống và làm việc ở nước ta là rất lớn và có xu hướng ngày càng tăng. Phần lớn họ đều có nhu cầu học tiếng việt và văn hóa Việt Nam bởi vì đó chính là chìa khóa giúp học phá vỡ rào cản ngôn ngữ và văn hóa để giao tiếp và hiểu rõ về con người, văn hóa Việt Nam. Đồng thời giúp họ đạt được thành công trong cuộc sống và công việc. Tuy nhiên, khi phát sinh nhu cầu họ lại vướng phải câu hỏi “Học tiềng việt và văn hóa Việt Nam ở đâu? Chỗ nào chuyên nghiệp, đạt hiệu quả cao?”.

1. Hiện trạng thị trường

* Hiện nay, số lượng người nước ngoài có nhu cầu học tiếng việt và văn hóa Việt Nam đang rất lớn và ngày càng tăng thêm. Điều này cho thấy đây là một thị trường rất tiềm năng đáng để đầu tư, phát triển. Chính vì vậy em quyết định xây dựng một website phục vụ công tác tuyển sinh chương trình tiếng việt và văn hóa Việt Nam.
* Vấn đề đặt ra là muốn website được nhiều bạn trẻ nước ngoài biết đến và sử dụng phổ biến thì website phải thỏa mãn các tiêu chí sau:
* Giao diện thân thiện, sử dụng dễ dàng, hỗ trợ nhiều

loại ngôn ngữ.

* Thông tin đầy đủ, nhanh chóng và cập nhập liên tục.
* Tốc độ xử lí yêu cầu nhanh và chính xác.
* Tập trung phát triển các tính năng cốt lõi.
* Cung cấp thêm các tiện ích mà người dùng mong muốn.
* Với những dẫn chứng trên em đã chọn đề tài *“*Xây dựng website phục vụ công tác tuyển sinh chương trình tiếng việt và văn hóa Việt Nam tại trung tâm hợp tác quốc tế trường Đại Học Công Nghiệp Hà Nội*”* để phù hợp với nhu cầu hiện nay cũng như đáp ứng những tiêu chí kể trên.

1. **Mục tiêu sản phẩm**

Với đề tài *“*Xây dựng website tuyển sinh cho trung tâm hợp

tác quốc tế trường Đại Học Công Nghiệp Hà Nội, ứng dụng

Java Springboot, ReactJS*”* em mong muốn website sẽ đạt một

số mục tiêu sau:

* Cung cấp đầy đủ chức năng, tiện ích cho người dùng.
* Giao diện đẹp và dễ dàng sử dụng.
* Tra cứu thông tin nhanh chóng và dễ dàng.
* Đáp ứng nhu cầu sử dụng website của nhiều người cùng một thời điểm.
* Thông tin cá nhân của các người dùng được bảo mật an toàn.

1. **Yêu cầu công nghệ**

* Hệ thống hoạt động ổn định, không bị treo khi có lượng truy cập lớn.
* Dễ dàng bảo trì khi gặp sự cố và phát triển thêm các tiện ích trong quá trình sử dụng.
* Sử dụng tiện lợi trên cả máy tính và điện thoại thông minh.

1. **Hoạt động hệ thống**

* Sinh viên đăng kí khóa học trên website của trung tâm, thông tin đăng kí được gửi về hệ thống và lưu lại trong danh sách hồ sơ ứng tuyển. Nhân viên trung tâm xem danh sách hồ sơ ứng tuyển vả báo lại với người quản lí hệ thống.Quản lí hệ thống xem xét danh sách hồ sơ ứng tuyển và đưa ra tiêu chí xét tuyển. Tiêu chí xét tuyển dựa trên kết quả học tập ở trung học phổ thông và trình độ ngoại ngữ của ứng viên.
* Những hồ sơ đạt yêu cầu sẽ được giữ lại và chuyển thành hồ sơ thông tin sinh viên của trung tâm.. Những hồ sơ không đạt yêu cầu sẽ bị hệ thống hủy. Nhân viên trung tâm sẽ gửi mail thông báo kết quả tới chủ hồ sơ và xác nhân chủ hồ sơ có nhập học hay không nếu là hồ sơ trúng tuyển.
* Người quản lí hệ thống sẽ tiếp tục xem sét danh sách hồ sơ trúng tuyển và đưa ra tiêu chí cấp học bổng (học bổng toàn phần, bán phần và theo phần trăm). Tiêu chí cấp học bổng dựa trên kết quả tập tại trường trung học phổ thông, trình độ ngoại ngữ và hoàn cảnh gia đình của sinh viên. Kết quả sau khi xét học bổng sẽ được gửi mail tới những sinh viên được cấp học bổng.
* Với những sinh viên đạt yêu cầu xét tuyển, sinh viên trước khi nhập học phải mang giấy tờ xác minh tới trực tiếp văn phòng xác nhân của trung tâm để xác nhận thông tin đã đăng kí online. Nếu xác nhận thành công, sinh được tiếp tục làm thủ tục nhập. Nếu thông tin trên giấy tờ xác nhận không chính xác với thông tin đăng kí online, sinh viên sẽ không được làm thủ tục nhập học, đồng thời sinh viên sẽ bị hủy hồ sơ đăng kí online và hủy học bổng (nếu có)
* Khi nhân dược yêu cầu cấp tài khoản cho người dùng mới , quản trị hệ thống thực hiện tạo mới, phân quyền và cấp tài khoản cho người dùng. Căn cứ vào chức vụ của nhân viên trung tâm để phân quyền tài khoản của nhân viên.

1. **Công nghệ sử dụng**
2. **Giới thiệu về Java**
3. Khái niệm

* Java là một ngôn ngữ lập lập trình, được phát triển bởi **Sun Microsystem** vào năm 1995, là ngôn ngữ kế thừa trực tiếp từ C/C++ và là một ngôn ngữ lập trình hướng đối tượng.
* Vì sao ngôn ngữ này lại được đặt tên là Java?
* Java là tên một hòn đảo ở Indonesia - hòn đảo nổi tiếng với loại coffee Peet và cũng là loại nước uống phổ biến của các kỹ sư Sun. Ban đầu Ngôn ngữ này được đặt tên là “oak” (có nghĩa là cây sồi 1991)*,* nhưng các luật sư của Sun xác định rằng tên đó đã được đăng ký nhãn hiệu nên các nhà phát triển đã phải thay thế bằng một tên mới -  và cũng vì lý do trên mà cái tên Java đã ra đời và trở thành tên gọi chính thức của Ngôn ngữ này - Ngôn ngữ Lập trình Java.

1. Ứng dụng của Java

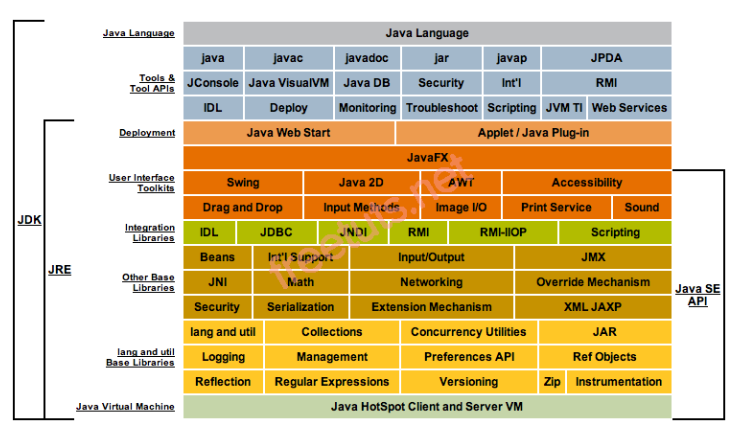
* Ngày nay Java được sử dụng với các mục đích sau:
* Phát triển ứng dụng cho các thiết bị điện tử thông minh, các ứng dụng cho doanh nghiệp với quy mô lớn.
* Tạo các trang web có nội dung động (web applet), nâng cao chức năng của server.
* Phát triển nhiều loại ứng dụng khác nhau: Cơ sở dữ liệu, mạng, Internet, viễn thông, giải trí,...

1. Những đặc điểm cơ bản của Java

* Tiêu chí hàng đầu của Ngôn ngữ Lập trình Java là **"Write Once, Run Anywhere"** (viết 1 lần chạy mọi nơi), nghĩa là Java cho phép chúng ta viết code một lần và thực thi được trên các hệ điều hành khác nhau. Ví dụ, chúng ta viết code trên Hệ điều hành Windows và nó có thể thực thi được trên các Hệ điều hành Linux và Mac OS...
* Với đặc điểm nổi bật đó, Java có những đặc điểm cơ bản như sau:
* Đơn giản và quen thuộc: Vì Java kế thừa trực tiếp từ C/C++ nên nó có những đặc điểm của ngôn ngữ này, Java đơn giản vì mặc dù dựa trên cơ sở C++ nhưng Sun đã cẩn thận lược bỏ các tính năng khó nhất của của C++ để làm cho ngôn ngữ này dễ sử dụng hơn.
* Hướng đối tượng và quen thuộc.
* Mạnh mẽ (thể hiện ở cơ chế tự động thu gom rác – garbage collection)và an toàn.
* Kiến trúc trung lập, độc lập nền tảng và có tính khả chuyển (Portabbility).
* Hiệu suất cao.
* Máy ảo (biên dịch và thông dịch).
* Phân tán.
* Đa nhiệm: Ngôn ngữ Java cho phép xâ dựng trình ứng dụng, trong đó nhiều quá trình có thể xảy ra đồng thời. Tính đa nhiệm cho phép các nhà lập trình có thể biên soạn phần mềm đáp ứng tốt hơn, tương tác tốt hơn và thực hiện theo thời gian thực.

1. Các platform cơ bản của Java

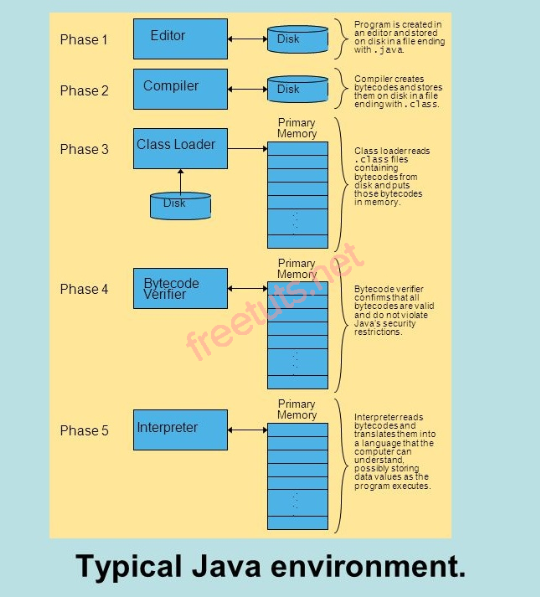
* Java Platform gồm có 3 thành phần chính:
* Java Virtual Machine (JVM): Máy ảo Java.
* Java Application Programming Interface(Java API).
* Java Development Kit (JDK) gồm trình biên dịch, thông dịch, trợ giúp, soạn tài liệu... và các thư viện chuẩn.



Hình 1. Các thành phần platform Java

1. Tiêu chuẩn của một môi trường Java điển hình

* Thông thường, các chương trình Java trải qua 5 giai đoạn chính:
* **Editor**: Lập trình viên viết chương trình và được lưu vào máy tính với định dạng .java.
* **Compiler**: Biên dịch chương trình thành bytecodes (ddingj dạng .class) - nhờ bước trung gian này mà Java được viết 1 lần và chạy trên các hệ điều hành khác nhau.
* **Class Loader**: Đọc file .class chứa mã bytecodes và lưu vào trong bộ nhớ.
* **Bytecode Verifier**: Đảm bảo rằng mã bytecodes là hợp lệ và không vi phạm các vấn đề về bảo mật của Java.
* **Intepreter**: Biên dịch bytecodes thành mã máy để máy tính có thể hiểu được và sau đó thực thi chương trình.



Hình 1. Môi trường Java điển hình

1. **Giới thiệu về Spring Framework**
2. Giới thiệu Spring Framework

* **Spring Framework** là một [bộ khung ứng dụng](https://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_khung_%E1%BB%A9ng_d%E1%BB%A5ng&action=edit&redlink=1) và [bộ chứa](https://en.wikipedia.org/wiki/Servlet_container) [đảo ngược điều khiển](https://en.wikipedia.org/wiki/inversion_of_control) cho [nền tảng Java](https://vi.wikipedia.org/wiki/Java_(c%C3%B4ng_ngh%E1%BB%87)). Chức năng tính của bộ khung này có thể áp dụng cho bất kỳ ứng dụng Java nào, tuy vậy, nhiều bản mở rộng dành cho việc xây dựng ứng dụng nền web cũng được phát triển trên nền tảng [Java EE](https://vi.wikipedia.org/w/index.php?title=Java_EE&action=edit&redlink=1). Tuy bộ khung không ấn định lên một cụ thể nào, Spring Framework đã trở nên phổ biến trong cộng đồng [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) như một phương án song hành, thay thế, hoặc thậm chí bổ sung cho mô hình [Enterprise JavaBeans](https://vi.wikipedia.org/w/index.php?title=Enterprise_JavaBeans&action=edit&redlink=1) (EJB).

1. Lịch sử ra đời của Spring Framework

* Phần đầu tiên của Spring Framework ban đầu nguyên được [Rod Johnson](https://vi.wikipedia.org/w/index.php?title=Rod_Johnson&action=edit&redlink=1) viết vào năm 2000, trong khi ông đang làm một chuyên viên cố vấn cho các khách hàng trong thị trường [kinh tế tài chính](https://vi.wikipedia.org/w/index.php?title=Kinh_t%E1%BA%BF_t%C3%A0i_ch%C3%ADnh&action=edit&redlink=1) tại [Luân Đôn](https://vi.wikipedia.org/wiki/Lu%C3%A2n_%C4%90%C3%B4n). Trong lúc đang viết quyển Expert One-on-one J2EE Design And Development (Programmer to programmer) (Thiết kế và kiến tạo Java EE - tương giao cá nhân một đối một chuyên ngành (lập trình viên với lập trình viên), ông đã mở rộng thêm phần mã nguồn, hòng biểu đạt ý tưởng của mình về mối tương quan của các trình ứng dụng với những phần khác nhau của [J2EE](https://vi.wikipedia.org/wiki/J2EE) và cho rằng người ta có thể đơn giản hóa mối tương quan của chúng, và mối liên hệ ấy có thể trở nên cố định, bất biến hơn là những gì mà các lập trình viên và các công ty vốn quen sử dụng tại thời điểm đó.

1. Tính năng của Spring Framework

* Bao gồm bộ quản lý cài đặt rất có hiệu lực dựa trên JavaBeans, áp dụng lý thuyết [đảo ngược quyền khống chế](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BA%A3o_ng%C6%B0%E1%BB%A3c_quy%E1%BB%81n_kh%E1%BB%91ng_ch%E1%BA%BF&action=edit&redlink=1) (IoC) làm cho việc mắc nối các chương trình ứng dụng trở nên nhanh chóng và dễ dàng hơn. Việc sử dụng (IoC) trong khung hình Spring Framework còn được ám chỉ đến với cái tên [Dependency Injection](https://vi.wikipedia.org/w/index.php?title=Dependency_Injection&action=edit&redlink=1) (DI) (Nghĩa đen: bơm hay tiếp thành phần phụ thuộc, từ bên ngoài vào, lúc cần thiết, thay vì phải tự tạo từ bên trong). Nguyên lý của IoC sử dụng trong Spring Framework là một kỹ thuật nhằm ngoại biên hóa (externalize) việc kiến tạo và quản lý những thành phần phụ thuộc. Lấy ví dụ trường hợp lớp "Foo" phụ thuộc vào một trường hợp cá biệt của lớp "Bar" để thi hình một nhiệm vụ nào đấy. Theo phương pháp cổ truyền, "Foo" phải tự kiến tạo trường hợp cá biệt của lớp "Bar", dùng hành động "new", hoặc phải tiếp nhận trường hợp cá biệt của lớp này từ một lớp chế xuất nào đấy (factory class). Nếu sử dụng tiếp cận của IoC thì một tiến trình ngoại biên sẽ phải cung cấp trường hợp của "Bar" (hoặc một tiểu lớp của nó) cho "Foo" trong thời gian chạy (runtime). Đây chính là nguyên nhân tại sao "DI" là một thành ngữ diễn tả tính năng của Spring Framework rõ hơn "IoC".
* Bao gồm một [mẫu hình chế xuất trừu tượng](https://vi.wikipedia.org/w/index.php?title=M%E1%BA%ABu_h%C3%ACnh_ch%E1%BA%BF_xu%E1%BA%A5t_tr%E1%BB%ABu_t%C6%B0%E1%BB%A3ng&action=edit&redlink=1) (abstract factory pattern) [Enterprise Java Beans](https://vi.wikipedia.org/w/index.php?title=Enterprise_Java_Beans&action=edit&redlink=1) (EJB) nòng cốt, có thể sử dụng trong bất cứ môi trường nào, từ những khung chứa (containers) [applet](https://vi.wikipedia.org/w/index.php?title=Applet&action=edit&redlink=1) cho đến [Java EE](https://vi.wikipedia.org/w/index.php?title=Java_EE&action=edit&redlink=1).
* Một lớp trừu tượng chung quản lý [giao dịch cơ sở dữ liệu](https://vi.wikipedia.org/wiki/Giao_d%E1%BB%8Bch_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) (database transaction), cho phép các bộ phận quản lý giao dịch được bổ trợ theo lối lắp ráp (pluggable), và tạo điều kiện thuận lợi cho việc phân định danh giới các giao dịch (demarcate transactions) mà không cần phải xử lý với những vấn đề ở tầng dưới (low-level issues).
* Bao gồm một số các chiến lược phổ thông đã được cài sẵn bên trong (Built-in generic strategies) dành cho [JTA](https://vi.wikipedia.org/w/index.php?title=Java_Transaction_API&action=edit&redlink=1) và một đơn thể nguồn dữ liệu [JDBC](https://vi.wikipedia.org/w/index.php?title=Java_Database_Connectivity&action=edit&redlink=1) (DataSource). Đối lập với những giao dịch do khung chứa quản lý (Container-Managed Transactions - CMT) đơn thuần trong JTA hoặc [EJB](https://vi.wikipedia.org/w/index.php?title=EJB&action=edit&redlink=1), hỗ trợ giao dịch trong bộ khung Spring không đòi hỏi môi trường [Java EE](https://vi.wikipedia.org/w/index.php?title=Java_EE&action=edit&redlink=1).
* Tầng liên nối với cơ sở dữ liệu (JDBC abstraction layer ) cung cấp một cơ chế tầng cấp đầy ý nghĩa đối với những ngoại lệ (exception hierarchy) (không còn phải dựa vào việc nắn bóp mã nguồn của nhà sản xuất phần mềm như đối với SQLException nữa), đơn giản hóa việc xử lý lỗi, và cũng đồng thời thuyên giảm số lượng dòng lệnh mà lập trình viên cần phải viết. Lập trình viên sẽ không bao giờ phải viết khối 'finally' trong các dòng mã sử dụng JDBC nữa.
* Tích hợp với [Hibernate](https://vi.wikipedia.org/w/index.php?title=Hibernate_(Java)&action=edit&redlink=1), [JDO](https://vi.wikipedia.org/w/index.php?title=Java_Data_Objects&action=edit&redlink=1) và [iBATIS](https://vi.wikipedia.org/w/index.php?title=IBATIS&action=edit&redlink=1) [SQL Maps](https://vi.wikipedia.org/w/index.php?title=SQL_Maps&action=edit&redlink=1): trên phương diện cầm dữ tài nguyên (resource holders), hỗ trợ phần thực thi DAO (Data Access Object - Đối tượng truy cập dữ liệu), và các chiến lược giao dịch (transaction strategies). Đứng hàng đầu trong việc hỗ trợ Hibernate với rất nhiều tính năng IoC tiện dụng, giải quyết nhiều nan đề trong việc tích hợp với Hibernate. Tất cả những tính năng này đều phục tùng cơ chế tầng cấp giao dịch chung cùng tầng cấp ngoại lệ DAO của Spring.
* Khung hình [mô hình-giao diện-điều khiển](https://vi.wikipedia.org/w/index.php?title=M%C3%B4_h%C3%ACnh-giao_di%E1%BB%87n-%C4%91i%E1%BB%81u_khi%E1%BB%83n&action=edit&redlink=1) (Model-view-controller, hay MVC) linh hoạt dành cho [ứng dụng mạng](https://vi.wikipedia.org/w/index.php?title=%E1%BB%A8ng_d%E1%BB%A5ng_m%E1%BA%A1ng&action=edit&redlink=1) (web application), được gây dựng trên tính năng nền tảng của Spring. Các nhà xây dựng phần mềm (lập trình viên) có quyền khống chế cao độ trong khung hình này thông qua những giao diện chiến lược (strategy interfaces), đồng thời điều tiết với những kỹ thuật đa giao diện (multiple view technologies) như [JSP](https://vi.wikipedia.org/wiki/JSP), [FreeMarker](https://vi.wikipedia.org/wiki/FreeMarker), [Velocity](https://vi.wikipedia.org/w/index.php?title=Velocity_(software)&action=edit&redlink=1), [Tiles](https://vi.wikipedia.org/w/index.php?title=Tiles&action=edit&redlink=1), [iText](https://vi.wikipedia.org/w/index.php?title=IText&action=edit&redlink=1), và [POI](https://vi.wikipedia.org/w/index.php?title=Apache_Jakarta_POI&action=edit&redlink=1). Nên nhớ là [Tầng trung gian](https://vi.wikipedia.org/w/index.php?title=3_tier&action=edit&redlink=1) của Spring có thể dễ dàng kết hợp với một tầng mạng sử dụng bất cứ một khung hình MVC mạng nào đó (web MVC framework), như [Struts](https://vi.wikipedia.org/wiki/Struts), [WebWork](https://vi.wikipedia.org/w/index.php?title=WebWork&action=edit&redlink=1), hoặc [Tapestry](https://vi.wikipedia.org/w/index.php?title=Tapestry&action=edit&redlink=1).
* Khung hình [lập trình định dạng](https://vi.wikipedia.org/w/index.php?title=L%E1%BA%ADp_tr%C3%ACnh_%C4%91%E1%BB%8Bnh_d%E1%BA%A1ng&action=edit&redlink=1) (Aspect-oriented programming - hay AOP) nhằm cung cấp các dịch vụ như dịch cụ quản lý giao dịch (transaction management). AOP cung cấp khả năng thực thi lôgic [đan kết](https://vi.wikipedia.org/w/index.php?title=%C4%90an_k%E1%BA%BFt&action=edit&redlink=1) (crosscutting logic) - tức những lôgic tác động đến rất nhiều phần của chương trình ứng dụng - tại cùng một chỗ (nhóm lại) và để cho lôgic ấy áp dụng xuyên suốt chương trình ứng dụng của mình một cách tự động.

1. Các modun của Spring Framework

* Spring Framework được coi như là một tổ hợp của nhiều khung hình nhỏ, hay nói cách khác, nó là tổ hợp của khung hình trong khung hình. Đại bộ phận các khung hình này được thiết kế để hoạt động biệt lập với các khung hình khác, song lại tạo nên nhiều chức năng tốt hơn khi được phối hợp làm việc với nhau. Những khung hình này được phân loại theo các khối nền cơ bản mà những chương trình phần mềm phức tạp điển hình thường sử dụng:
* Khung chứa [đảo ngược khống chế](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BA%A3o_ng%C6%B0%E1%BB%A3c_kh%E1%BB%91ng_ch%E1%BA%BF&action=edit&redlink=1) (Inversion of Control container): Sự cài đặt các thành phần của chương trình ứng dụng và quản lý chu trình của các đối tượng trong Java (lifecycle management of Java objects).
* Khung hình [lập trình định dạng](https://vi.wikipedia.org/w/index.php?title=L%E1%BA%ADp_tr%C3%ACnh_%C4%91%E1%BB%8Bnh_d%E1%BA%A1ng&action=edit&redlink=1) (Aspect-oriented programming framework): Spring AOP module tích hợp chức năng lập trình hướng khía cạnh vào Spring framework thông qua cấu hình của nó. Spring AOP module cung cấp các dịch vụ quản lý giao dịch cho các đối tượng trong bất kỳ ứng dụng nào sử dụng Spring. Với Spring AOP chúng ta có thể tích hợp declarative transaction management vào trong ứng dụng mà không cần dựa vào EJB component.
* Spring AOP module cũng đưa lập trình metadata vào trong Spring. Sử dụng cái này chúng ta có thể thêm annotation vào source code để hướng dẫn Spring nơi và làm thế nào để liên hệ với aspect..
* Khung hình [truy cập dữ liệu](https://vi.wikipedia.org/wiki/Truy_c%E1%BA%ADp_d%E1%BB%AF_li%E1%BB%87u) (Data access framework):Tầng JDBC và DAO đưa ra một cây phân cấp exception để quản lý kết nối đến database, điều khiển exception và thông báo lỗi được ném bởi vendor của database. Tầng exception đơn giản điều khiển lỗi và giảm khối lượng code mà chúng ta cần viết như mở và đóng kết nối. Module này cũng cung cấp các dịch vụ quản lý giao dịch cho các đối tượng trong ứng dụng Spring..
* Khung hình [quản lý giao dịch](https://vi.wikipedia.org/w/index.php?title=Qu%E1%BA%A3n_l%C3%BD_giao_d%E1%BB%8Bch&action=edit&redlink=1) (Transaction processing / Transaction management framework): Sự đồng hòa các API quản lý giao dịch và sự điều hợp quản lý giao dịch đối với các đối tượng Java dùng phương pháp cài đặt.
* Khung hình [mô hình-giao diện-điều khiển](https://vi.wikipedia.org/w/index.php?title=M%C3%B4_h%C3%ACnh-giao_di%E1%BB%87n-%C4%91i%E1%BB%81u_khi%E1%BB%83n&action=edit&redlink=1) (Model-view-controller framework): Khung hình dựa trên nền tảng [HTTP](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol) và [Servlet](https://vi.wikipedia.org/w/index.php?title=Java_Servlet_API&action=edit&redlink=1) cung cấp rất nhiều móc điểm có thể mở rộng và cải biên (extension and customization).
* Khung hình truy cập từ xa (Remote Access framework): Xuất nhập khẩu các đối tượng Java trên mạng lưới truyền thông hỗ trợ các giao thức dựa trên nền [HTTP](https://vi.wikipedia.org/wiki/Hypertext_Transfer_Protocol), sử dụng phong cách [RPC](https://vi.wikipedia.org/w/index.php?title=RPC&action=edit&redlink=1) (Remote procedure call) thông qua cài đặt, như [RMI](https://vi.wikipedia.org/w/index.php?title=RMI&action=edit&redlink=1) (Java remote method invocation - khởi động phương pháp từ xa trong Java), [CORBA](https://vi.wikipedia.org/w/index.php?title=CORBA&action=edit&redlink=1) và [dịch vụ mạng](https://vi.wikipedia.org/wiki/D%E1%BB%8Bch_v%E1%BB%A5_m%E1%BA%A1ng) (web services - [SOAP](https://vi.wikipedia.org/wiki/SOAP)).
* Khung hình [xác thực](https://vi.wikipedia.org/wiki/X%C3%A1c_th%E1%BB%B1c) và [ủy quyền](https://vi.wikipedia.org/w/index.php?title=%E1%BB%A6y_quy%E1%BB%81n&action=edit&redlink=1) (Authentication and authorization framework): Sự điều hợp các quá trình xác thực và ủy quyền thông qua cài đặt hỗ trợ nhiều tiêu chuẩn công nghiệp và những tiêu chuẩn được sử dụng phổ biến, cũng như các giao thức, công cụ và các thói quen thực hành phổ dụng.
* Khung hình quản lý từ xa (Remote Management framework): Sự quản lý và quảng bá các đối tượng Java đối với các bố trí địa phương và bố trí ngoại biên thông qua [JMX](https://vi.wikipedia.org/w/index.php?title=JMX&action=edit&redlink=1) (Java Management Extension - quản lý mở rộng của Java cho phép các phần mềm không trực thuộc Java có thể lắp ráp vào và sử dụng được trong nền tảng Java).
* Khung hình truyền thông điệp (Messaging framework): đăng ký các đối tượng lắng nghe thông điệp dùng để tiếp thu và sử dụng thông điệp từ các [hàng thông điệp](https://vi.wikipedia.org/w/index.php?title=H%C3%A0ng_th%C3%B4ng_%C4%91i%E1%BB%87p&action=edit&redlink=1) (message queue) thông qua [JMS](https://vi.wikipedia.org/w/index.php?title=JMS&action=edit&redlink=1) (Java Message Service - Dịch vụ Thông điệp của Java), cải tiến việc truyền gửi thông điệp dùng tiêu chuẩn trong [API](https://vi.wikipedia.org/wiki/Giao_di%E1%BB%87n_l%E1%BA%ADp_tr%C3%ACnh_%E1%BB%A9ng_d%E1%BB%A5ng) của [JMS](https://vi.wikipedia.org/w/index.php?title=JMS&action=edit&redlink=1).
* Khung hình [kiểm thử phần mềm](https://vi.wikipedia.org/w/index.php?title=Ki%E1%BB%83m_th%E1%BB%AD_(ph%E1%BA%A7n_m%E1%BB%81m)&action=edit&redlink=1) (Software testing framework): Những chủ thể (lớp) hỗ trợ việc viết các kiểm thử đơn vị và các kiểm thử tích hợp (unit tests and integration tests).

1. **Giới thiệu về Spring Boot**
2. Giới thiệu Spring Boot

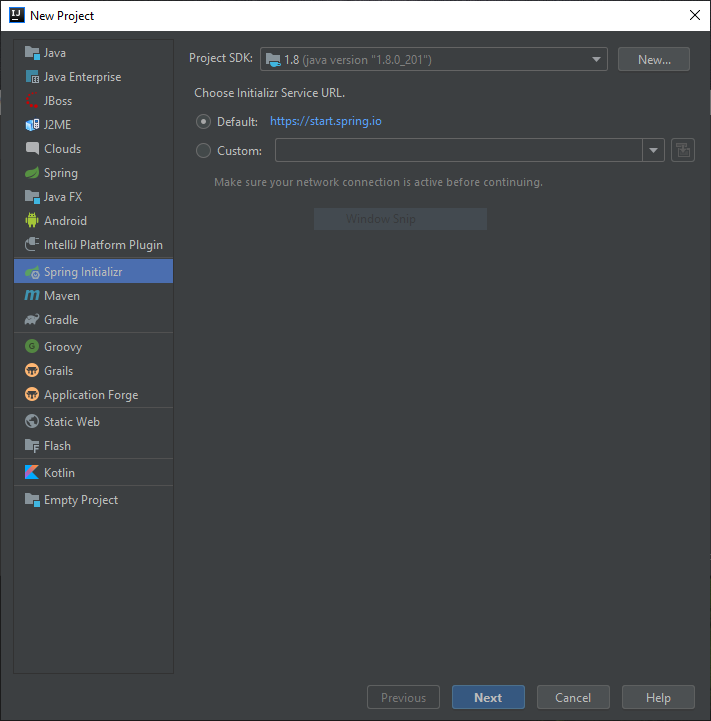
* **Spring Boot** là một dự án phát triển bởi **JAV**(ngôn ngữ java) trong hệ sinh thái Spring framework. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng.
* Có các tính năng của Spring Framework.
* Tạo ứng dụng độc lập, có thể chạy bằng java -jar (cho cả java web)
* Nhúng trực tiếp các ứng dụng server (Tomcat, Jetty…) do đó không cần phải triển khai file WAR
* Cấu hình ít, tự động cậu hình bất kì khi nào có thể (Giảm thời gian viết code, tăng năng suất)
* Không yêu cầu XML config…
* Cung cấp nhiều plugin
* Chuẩn cho Microservices (Cloud support; giảm việc setup, config; các thư viện hỗ trợ…)

1. Các tính năng nổi bật

* Đóng gói ứng dụng Spring dưới dạng một file JAR (stand-alone application). Chúng ta có thể dễ dàng start ứng dụng Spring chỉ với câu lệnh quen thuộc java -jar.
* Tối ưu công đoạn cấu hình cho ứng dụng Spring, không sinh code cấu hình và không yêu cầu phải cấu hình bằng XML.
* Cung cấp một loạt các tính năng phi chức năng phổ biến cho các dự án lớn như nhúng trực tiếp web server như Tomcat, Jetty, ... vào ứng dụng, bảo mật, health check...

1. Tạo một project sử dụng Spring Boot với Intelij Idea

Bước 1: Chọn Spring Initializr và chọn phiên bản JDK của bạn. Sau đó chọn Next.



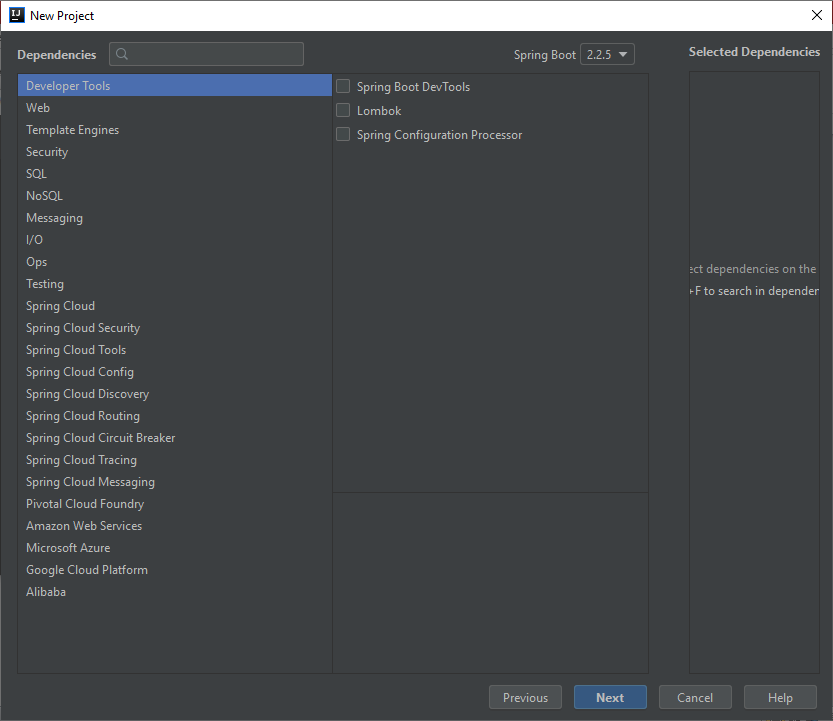
Bước 2: Thiết lập các thông tin cơ bản cho project, gồm các thông tin:

* Group: Tên tổ chức /công ty /cá nhân của dự án.
* Artifact: Tên package của dự án.
* Type: Cấu hình trình quản lí dự án.
* Language: Loại ngôn ngữ sử dụng cho dự án.
* Paking: Sản phẩm sau khi build dự án.
* Java Version: Phiên bản java sử dụng trong dự án.
* Version: Phiên bản dự án là lần thứ mấy.
* Name:Tên của dự án.
* Description:Mo tả dự án.
* Package: gói sẽ chứa các file hoạt động của dự án.

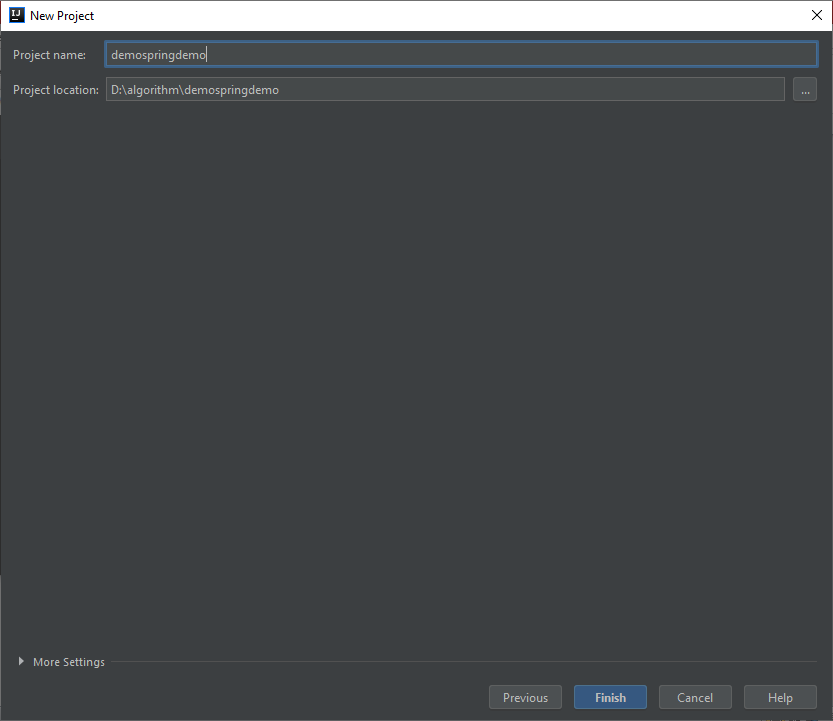
sau đó chọn Next.



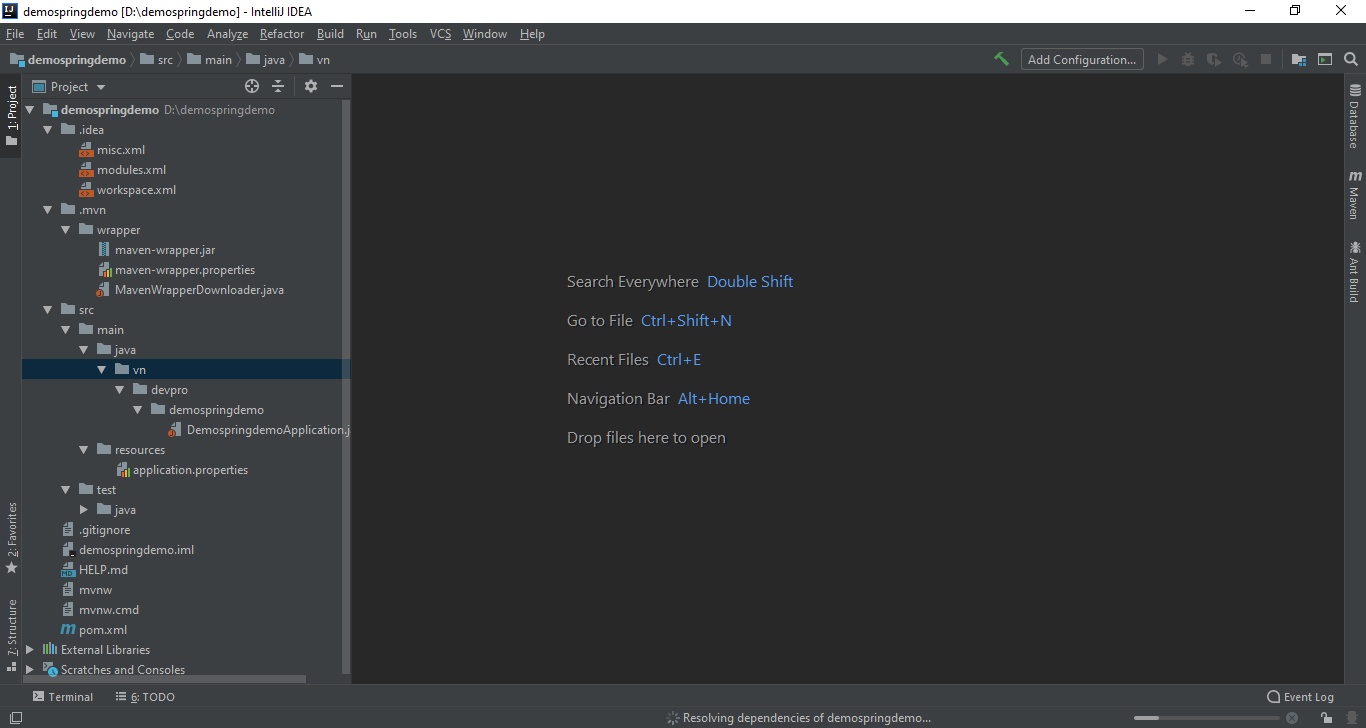
Bước 3: chọn các công cụ mà bạn sẽ sử dụng trong dự án của mình. Nếu không xác định trước thì để mặc định rồi chọn Next.



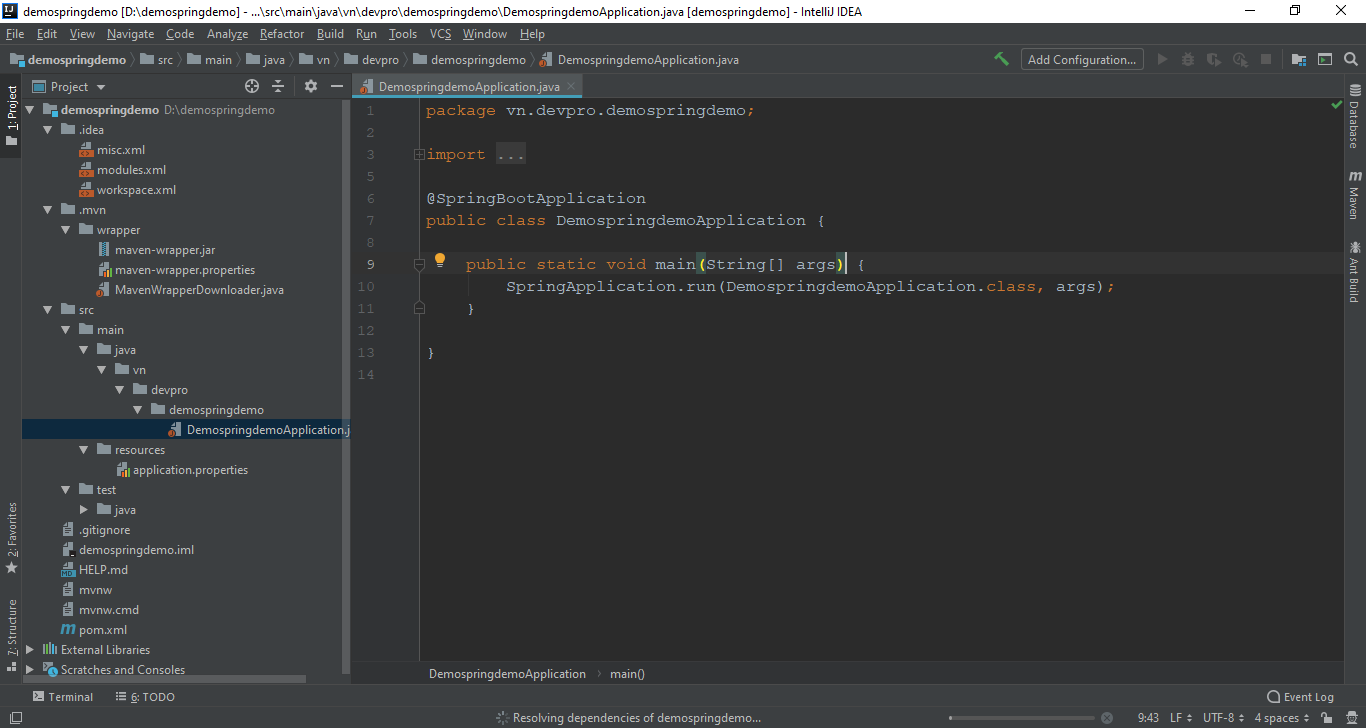
Bước 4:Đặt tên cho project và setup nơi lưu trữ dự án. Sau đó chọn Finish.



Sau khi hoàn tất chúng ta được một project đã được cấu hình sẵn.



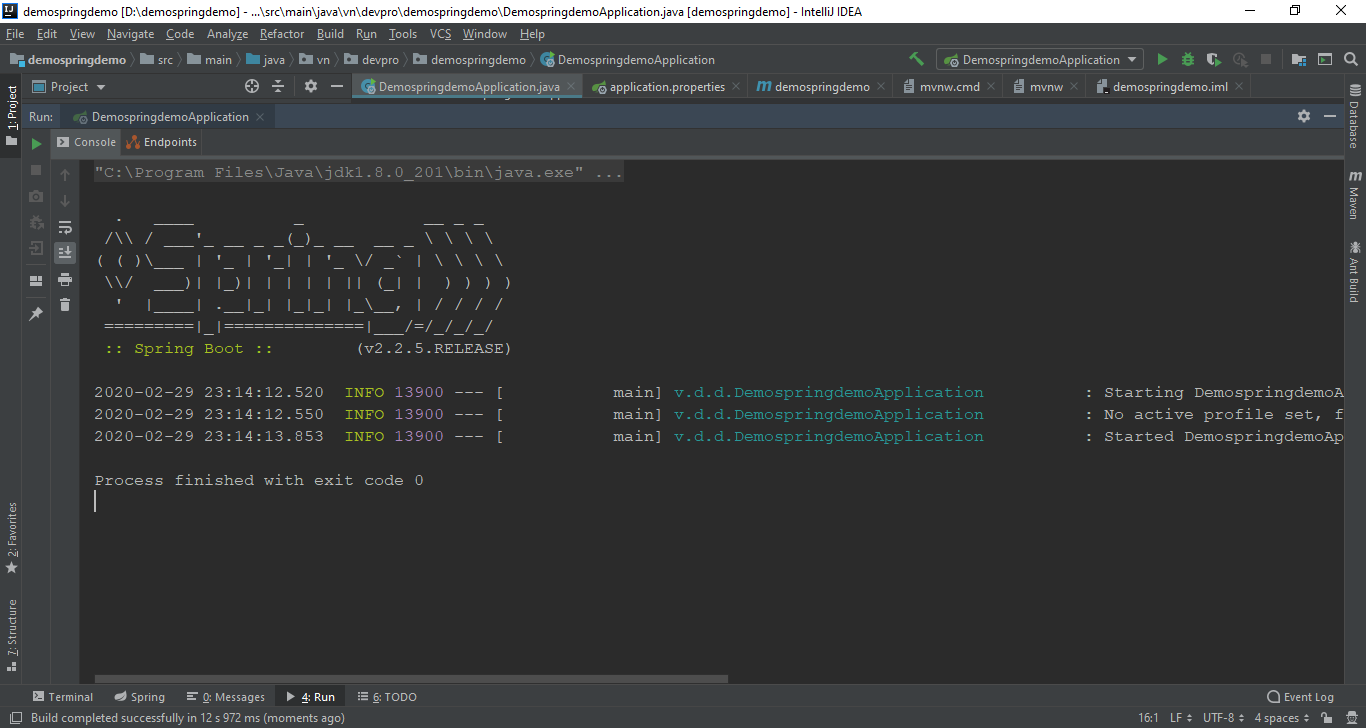
Đây là file chứa hàm main() project:



File pom.xml là nơi khai báo các thư viện mà chúng ta sử dụng.



Màn hình console khi chạy project.



# CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG

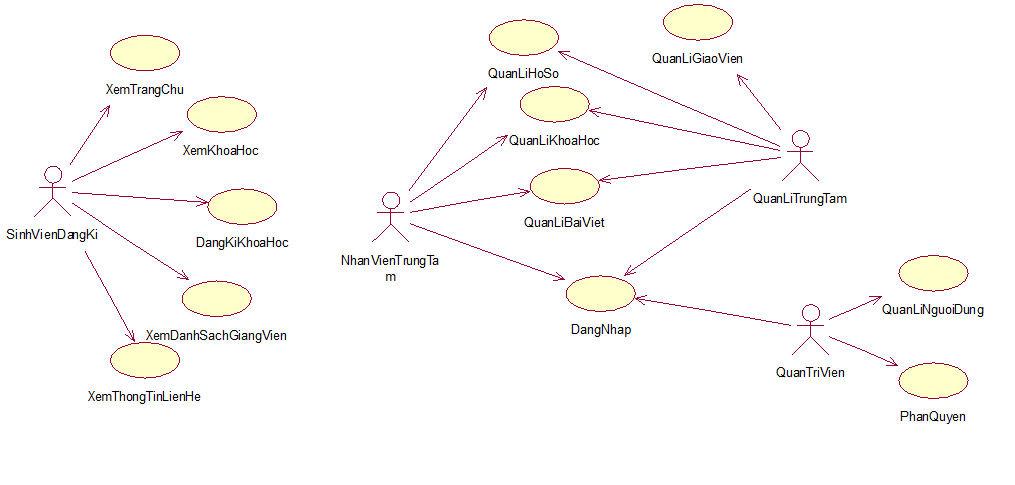
1. **Phân tích biểu đồ usecase**
2. **Xác định các tác nhân**

* Sinh viên đăng kí
* Nhân viên trung tâm
* Quản lí trung tâm
* Quản trị viên

1. **Xác định usecase**

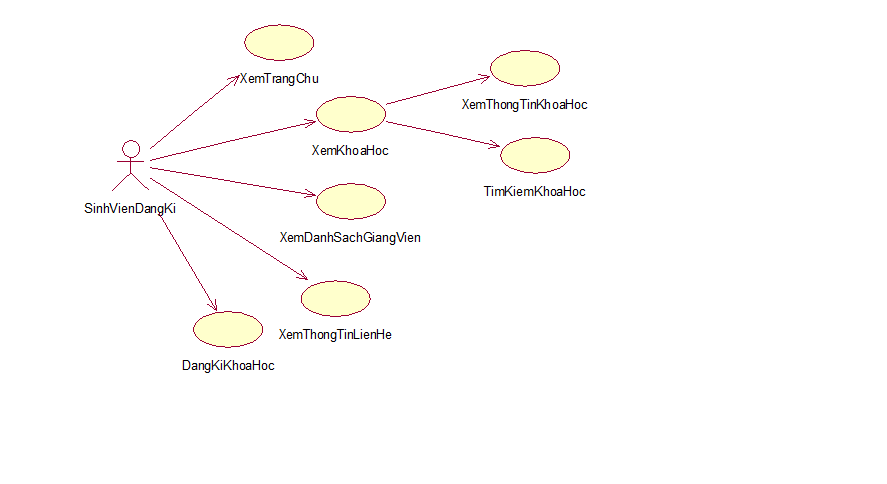
* Sinh viên đăng kí:
  + - * Xem trang chủ.
      * Xem thông tin khóa học.
      * Xem danh sách giảng viên.
      * Xem thông tin liên hệ.
      * Đăng kí khóa học.
* Nhân viên trung tâm:
* Đăng nhập.
* Quản lí hồ sơ: Xem danh sách, tìm kiếm, xóa hồ sơ.
* Quản lí bài viết: Xem danh sách, thêm bài viết, sửa bài viết, đăng bài viết, gỡ bài viết, tìm kiếm bài viết.
* Quản lí khóa học: Xem danh sách, thêm mới khóa học, xóa khóa học, chỉnh sửa thông tin khóa học, tìm kiếm khóa học
* Quản lí trung tâm:
* Đăng nhập
* Quản lí hồ sơ: Xem danh sách, tìm kiếm, xóa hồ sơ.
* Quản lí bài viết: Xem danh sách, hêm bài viết, sửa bài viết, đăng bài viết, gỡ bài viết, tìm kiếm bài viết.
* Quản lí slide: Xem danh sách, thêm slide, xóa slide, tìm kiếm slide.
* Quản lí khóa học: Xem danh sách, thêm mới khóa học, xóa khóa học, chỉnh sửa thông tin khóa học, tìm kiếm khóa học
* Quản lí giáo viên: Xem danh sách, thêm giáo viên, xóa giáo viên,tìm kiếm giáo viên.
* Quản trị hệ thống:
* Đăng nhập.
* Quản lí quyền truy cập trong hệ thống: Xem danh sách, thêm quyền, sửa quyền.
* Quản lí người dùng: Xem danh sách, thêm người dùng, xóa người dùng.

1. **Biểu đồ usecase**
2. Usecase tổng quát



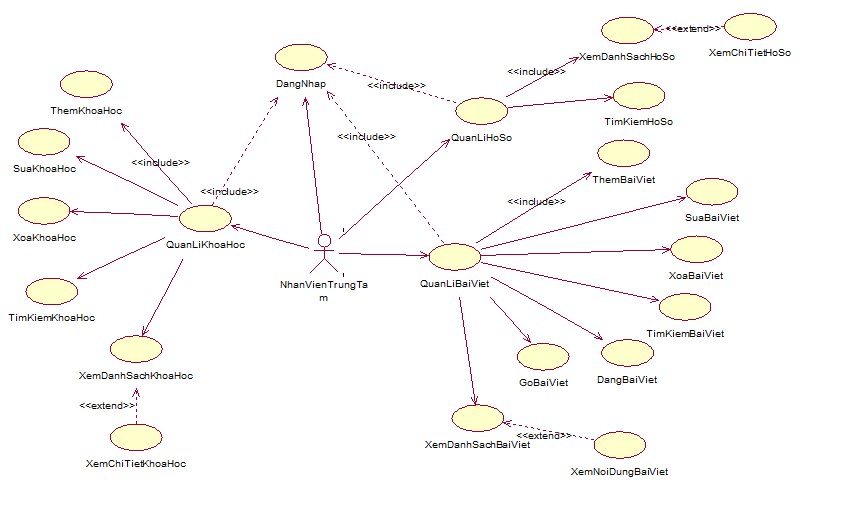
Hình 2.1 Usecase tổng quát

1. Usecase sinh viên đăng kí



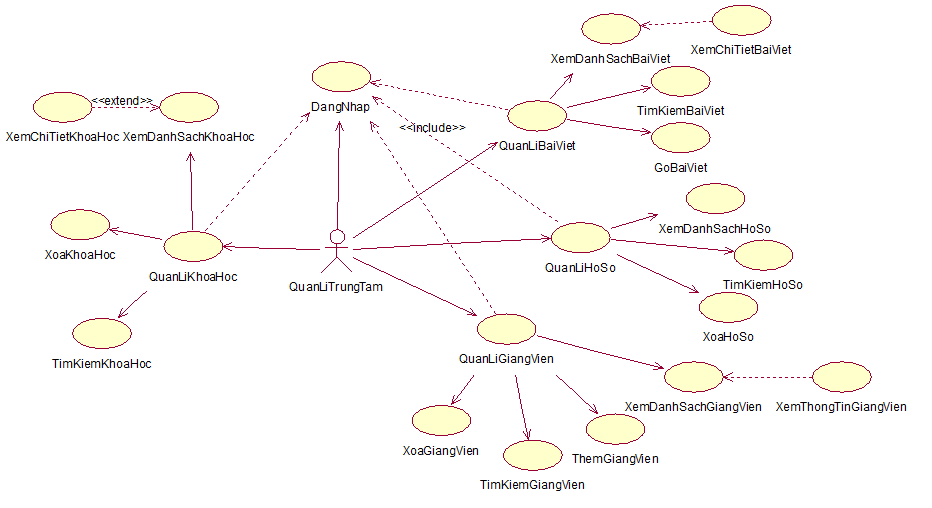
Hình 2.2 Usecase sinh viên đăng kí

1. Usecase nhân viên trung tâm



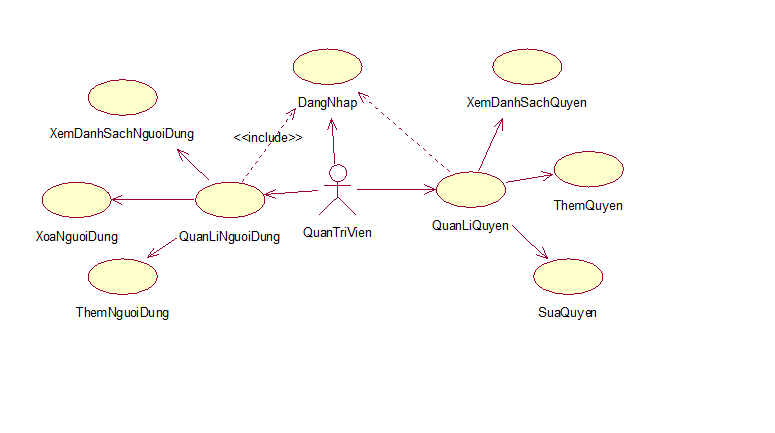
Hình 2.3 Usecase nhân viên trung tâm

1. Usecase quản lí trung tâm



Hình 2.4 Usecase quản lí trung tâm

1. Usecase quản trị viên



Hình 2.5 Usecase quản trị viên

1. **Đặc tả usecase**
2. **Usecase xem trang chủ**
3. Mô tả vắn tắt

* Tên ca sử dụng: Xem trang chủ
* Mục đích: Hiển thị trang chủ cho tác nhân
* Tác nhân: Sinh viên đăng kí

1. Các luồng sự kiện

* Ca sử dụng bắt đầu từ khi sinh viên muốn tìm hiểu thông tin về các khóa học tiếng việt và văn hóa Việt Nam.
* Sinh viên tìm kiếm bằng từ khóa trên google, danh sách các kết quả được trả về, sinh viên click chọn link Website của trung tâm.
* Sinh viên được chuyển đến website trang chủ của trung tâm.

1. Tiền điều kiện
2. Hậu điều kiện

* Sinh viên truy cập được vào trang chủ của trung tâm.

1. **Usecase xem khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xem khóa học.
* Mục đích: Giúp sinh viên xem thông tin các khóa học của trung tâm.
* Tác nhân: Sinh viên đăng kí.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi sinh viên muốn xem danh sách các khóa học.
* Sinh viên click chuột chọn danh mục khóa học.
* Hệ thống lấy danh sách các khóa học hiển thị lên màn hình.
* Sinh viên chọn 1 khóa học và click chuột để xem chi tiết khóa học.
* Hệ thống lấy chi tiết khóa học được yêu cầu hiển thị lên màn hình.
* Nếu sinh viên muốn tìm khóa học khác, sinh viên nhập tên khóa học muốn tìm vào form tìm kiếm, rồi click chuột vào tìm kiếm.
* Hệ thống lấy danh sách khóa học có chứa từ khóa tìm kiếm và hiện thị lên màn hình. Nếu không có kết quả phù hợp thì => luồng A1
* Luồng sự kiện rẽ nhánh:
* A1: không có kết nào được trả về, hệ thống hiển thị trang trắng.

1. Tiền điều kiện

* Sinh viên đang ở trang chủ.

1. Hậu điều kiện

* Hiển thị danh sách các khóa học lên màn hình.

1. **Usecase đăng kí khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Đăng kí khóa học.
* Mục đích: Giúp sinh viên đăng kí khóa học của trung tâm.
* Tác nhân: Sinh viên đăng kí.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi sinh viên muốn đăng kí học tại trung tâm. Sinh viên click chuột chọn đăng kí ngay.
* Hệ thống hiển thị trang đăng kí.
* Sinh viên nhập đầy đủ thông tin bắt buộc rồi chọn đăng kí. Nêu không nhập đầy đủ thông tin bắt buộc thì => luồng A1
* Luồng sự kiện rẽ nhánh:
* A1: Hiển thị lỗi không cho đăng kí, yêu cầu sinh viên nhập đầy đủ thông tin cần thiết.

1. Tiền điều kiện

* Sinh viên đang ở trang chủ.

1. Hậu điều kiện

* Hiển thị thông báo thành công.

1. **Usecase xem danh sách giảng viên**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xem danh sách giang viên.
* Mục đích: Giúp sinh viên có thêm thông tin về các giảng viên của trung tâm.
* Tác nhân: Sinh viên đăng kí.

1. Các luồng sự kiện

* Ca sử dụng bắt đàu khi sinh viên muốn tìm hiểu về đội ngũ giảng viên của trung tâm.
* Sinh viên click chuột chọn danh sách giảng viên.
* Hệ thống lấy danh sách các giảng viên của trung trung tâm và hiển thị danh sách lên màn hình.
* Luồng sự kiện rẽ nhánh: không có.

1. Tiền điều kiện

* Sinh viên đang ở trang chủ.

1. Hậu điều kiện

* Hiển thị danh sách giảng viên.

1. **Usecase xem thông tin liên hệ**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xem danh thông tin liên hệ.
* Mục đích: cung cấp cho sinh viên thông tin liên của trung tâm.
* Tác nhân: Sinh viên đăng kí.

1. Các luồng sự kiện

* Ca sử dụng bắt đàu khi sinh viên muốn tìm hiểu về thông tin liên hệ của trung tâm
* Sinh viên click chuột chọn liên hệ.
* Hệ thống hiển thị thông tin liên hệ của trung tâm.
* Luồng sự kiện rẽ nhánh: Không có.

1. Tiền điều kiện

* Sinh viên đăng kí đang ở trang chủ.

1. Hậu điều kiện

* Hiển thị thông tin liên hệ của trung tâm.

1. **Usecase xem danh sách hồ sơ**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xem danh sách hồ sơ.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm thống kê danh sách sinh viên đăng kí học trong thời gian tới.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn xem hồ sơ đăng kí.
* Click chọn mục quản lí hồ sơ.
* Hệ thống chuyển tới trang quản lí hồ sơ. Nếu không =>luồng A1
* Hệ thống lấy danh sách hồ sơ đăng kí và hiển thị lên màn hình.
* Luồng sự kiện rẽ nhánh:
* A1: Hiển thị lỗi không có quyền truy cập vào danh mục.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang chủ admin.

1. Hậu điều kiện

* Hiển thị trang quản lí hồ sơ.

1. **Usecase tìm kiếm hồ sơ**
2. Mô tả vắn tắt

* Tên ca sử dụng: Tìm kiếm hồ sơ.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm tìm kiếm hồ sơ mong muốn.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn tìm hồ sơ đăng kí.
* Nhân viên trung tâm và quản lí trung tâm nhập thông tin muốn tìm: ngày bắt đầu, ngày kết thúc... Click chọn button tìm kiếm.
* Hệ thống lấy danh sách hồ sơ đăng kí phù hợp và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: Không có hồ sơ nào thỏa mãn yêu cầu tìm kiếm. Hiển thị danh sách rỗng lên màn hình.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí hồ sơ.

1. Hậu điều kiện

* Hiển thị trang danh sách sinh viên đang cần tìm.

1. **Usecase xóa hồ sơ**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xóa hồ sơ.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm xóa hồ sơ mong muốn.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn xóa hồ sơ đăng kí.
* Nhân viên trung tâm và quản lí trung tâm chọn hồ sơ muốn xóa. Click chọn xóa.
* Hệ thống xóa hồ hồ sơ đăng kí, cập nhập lại danh sách hồ sơ và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: Hiện thị thông báo lỗi: không có quyền xóa hồ sơ.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí hồ sơ.

1. Hậu điều kiện

* Hiển thị danh sách sinh viên sau khi đã xóa.

1. **Usecase xem danh sách khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xem danh sách khóa học.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm thống kê danh sách khóa học của trung tâm.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn xem danh sách các khóa học.
* Click chọn mục quản lí khóa học.
* Hệ thống chuyển tới trang quản lí khóa học. Nếu không =>luồng A1
* Hệ thống lấy danh sách khóa học và hiển thị lên màn hình.
* Luồng sự kiện rẽ nhánh:
* A1: hiện thị lỗi không có quyền truy cập vào danh mục.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang chủ admin.

1. Hậu điều kiện

* Hiển thị trang quản lí khóa học.

1. **Usecase xem tìm kiếm khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Tìm kiếm khóa học.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm tìm kiếm khóa học mong muốn.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn tìm khóa học.
* Nhân viên trung tâm và quản lí trung tâm nhập thông tin muốn tìm: ngày bắt đầu, ngày kết thúc... Click chọn tìm kiếm.
* Hệ thống lấy danh sách khóa học phù hợp và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: không có khóa học nào thỏa mãn yêu cầu tìm kiếm. Hiển thị danh sách rỗng lên màn hình.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí khóa học.

1. Hậu điều kiện

* Hiển thị danh sách khóa học đang cần tìm.

1. **Usecase thêm khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Thêm khóa học.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm cập nhập khóa học mới.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn thêm 1 khóa học mới
* Nhân viên trung tâm và quản lí trung tâm chọn thêm mới.
* Hệ thống hiện thị form thêm mới khóa học.
* Người dùng nhập thông tin khóa học mới rồi click chọn chọn thêm.
* Hệ thống thêm mới khóa học vào danh sách khóa học, cập nhập lại danh sách khóa học và hiển thị lên màn hình. Nếu không => A1 or =>A2
* Luồng sự kiện rẽ nhánh:
* A1: Không cho thêm mới, hiện thị thông báo lỗi: khóa học đã tồn tại không thể thêm mới.
* A2: Thông tin khóa học không hợp lệ.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí khóa học.

1. Hậu điều kiện

* Hiển thị thông báo và danh sách khóa học sau khi thêm mới.

1. **Usecase sửa khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Sửa khóa học.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm cập nhập thông tin khóa học.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn sửa thông tin một khóa học.
* Nhân viên trung tâm và quản lí trung tâm chọn khóa học muốn sửa. Click chọn sửa.
* Hệ thống hiện thị form sửa khóa học.
* Người dùng nhập thông tin mới rồi click chọn sửa
* Hệ thống cạp nhập thông tin mới của khóa học, cập nhập lại danh sách khóa học và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: Thông tin khóa học không hợp lệ.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí khóa học.

1. Hậu điều kiện

* Hiển thị thông báo và danh sách khóa học sau khi cập nhập lại thông tin.

1. **Usecase xóa khóa học**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xóa khóa học.
* Mục đích: Giúp nhân viên trung tâm và quản lí trung tâm xóa khóa học mong muốn.
* Tác nhân: Nhân viên trung tâm và quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn xóa khóa học.
* Người dùng chọn khóa học muốn xóa. Click chọn xóa.
* Hệ thống xóa khóa học, cập nhập lại danh sách khóa học và hiển thị lên màn hình. Nếu không => A1 or A2
* Luồng sự kiện rẽ nhánh:
* A1: Hiện thị thông báo lỗi: Không có quyền xóa khóa học.
* A2: Hiện thị thông báo lỗi: Khóa học đang có người đang trong quá trình giảng dạy.

1. Tiền điều kiện

* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí khóa học.

1. Hậu điều kiện

* Hiển thị danh sách khóa học sau khi đã xóa.

1. **Usecase xem danh sách giảng viên**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xem danh sách giảng viên.
* Mục đích: Giúp quản lí trung tâm thống kê danh sách giảng viên của trung tâm.
* Tác nhân: Quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi nhân viên trung tâm và quản lí trung muốn xem danh sách các giảng viên.
* Click chọn mục quản lí giảng viên.
* Hệ thống chuyển tới trang quản lí giảng viên. Nếu không =>luồng A1
* Hệ thống lấy danh sách giảng viên và hiển thị lên màn hình.
* Luồng sự kiện rẽ nhánh:
* A1: Hiển thị lỗi không có quyền truy cập vào danh mục.

1. Tiền điều kiện

* Quản lí trung tâm đang ở trang chủ admin.

1. Hậu điều kiện

* Hiển thị trang quản lí giảng viên.

1. **Usecase tìm kiếm giảng viên**
2. Mô tả vắn tắt

* Tên ca sử dụng: Tìm kiếm giảng viên.
* Mục đích: Giúp quản lí trung tâm tìm kiếm giảng viên mong muốn.
* Tác nhân: Quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi giảng viên quản lí trung muốn tìm khóa học.
* Quản lí trung tâm nhập thông tin muốn tìm: ngày bắt đầu, ngày kết thúc... Click chọn tìm kiếm.
* Hệ thống lấy danh sách giảng viên phù hợp và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: Không có giảng viên nào thỏa mãn yêu cầu tìm kiếm. Hiển thị danh sách rỗng lên màn hình.

1. Tiền điều kiện

* Quản lí trung tâm đang ở trang quản lí giảng viên.

1. Hậu điều kiện

* Hiển thị danh sách giảng viên đang cần tìm.

1. **Usecase thêm giảng viên**
2. Mô tả vắn tắt

* Tên ca sử dụng: Thêm giảng viên.
* Mục đích: Giúp quản lí trung tâm cập nhập giảng viên mới.
* Tác nhân: Quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi quản lí trung muốn thêm 1 giảng viên mới.
* Quản lí trung tâm chọn thêm mới.
* Hệ thống hiện thị form thêm mới giảng viên.
* Quản lí trung tâm nhập thông tin giảng viên mới rồi click chọn chọn thêm.
* Hệ thống thêm mới giảng viên vào danh sách giảng viên, cập nhập lại danh sách giảng viên và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: Hiện thị thông báo lỗi: Thông tin giảng viên giảng viên không hợp lệ, không thể thêm mới.

1. Tiền điều kiện

* Quản lí trung tâm đang ở trang quản lí giảng viên.

1. Hậu điều kiện

* Hiển thị thông báo và danh sách giảng viên sau khi thêm mới.

1. **Usecase xóa giảng viên**
2. Mô tả vắn tắt

* Tên ca sử dụng: Xóa giảng viên.
* Mục đích: Quản lí trung tâm xóa khóa học mong muốn.
* Tác nhân: Quản lí trung tâm.

1. Các luồng sự kiện

* Ca sử dụng bắt đầu khi quản lí trung muốn xóa khóa học.
* Người dùng chọn giảng viên muốn xóa. Click chọn xóa.
* Hệ thống xóa giảng viên, cập nhập lại danh sách giảng viên và hiển thị lên màn hình. Nếu không => A1
* Luồng sự kiện rẽ nhánh:
* A1: Hiện thị thông báo lỗi: không thể xóa giảng viên đang giảng dạy.

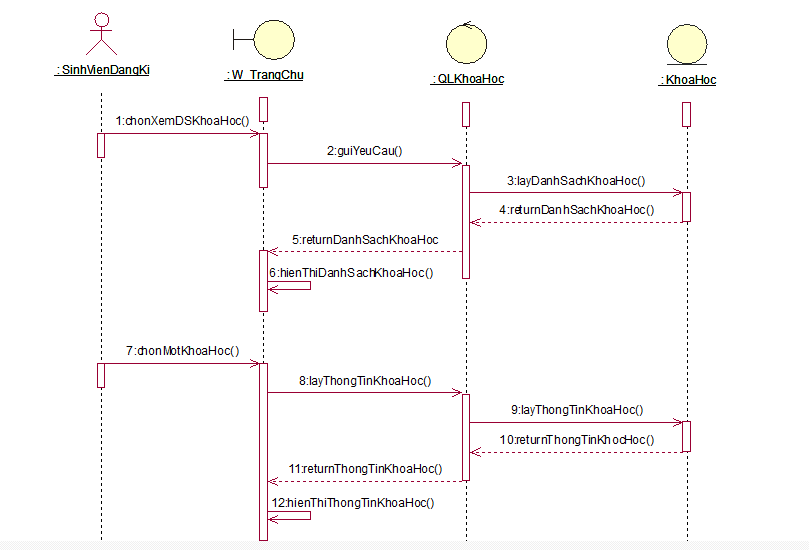
1. Tiền điều kiện

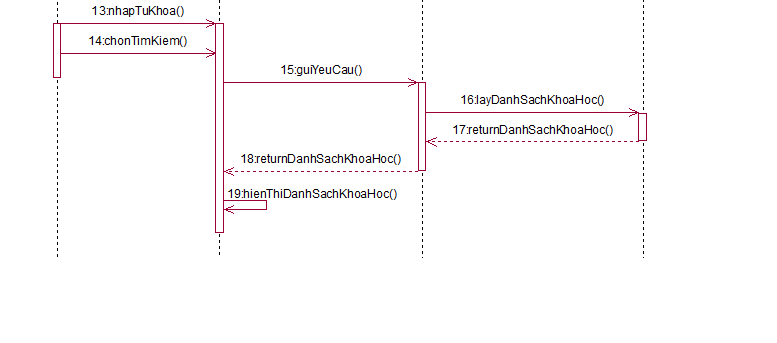
* Nhân viên trung tâm, quản lí trung tâm đang ở trang quản lí giảng viên.

1. Hậu điều kiện

* Hiển thị danh sách giảng viên sau khi đã xóa.

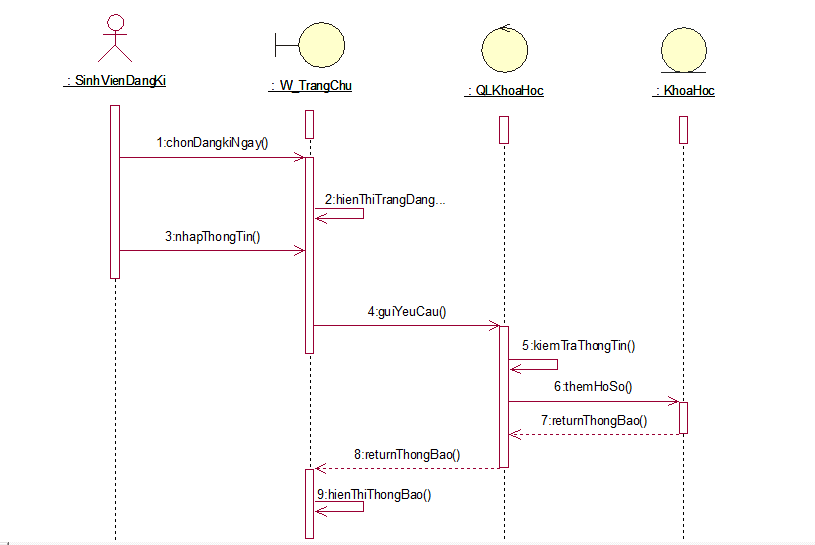
1. **Biểu đồ tuần tự**
2. **Biểu đồ tuần tự xem khóa học**





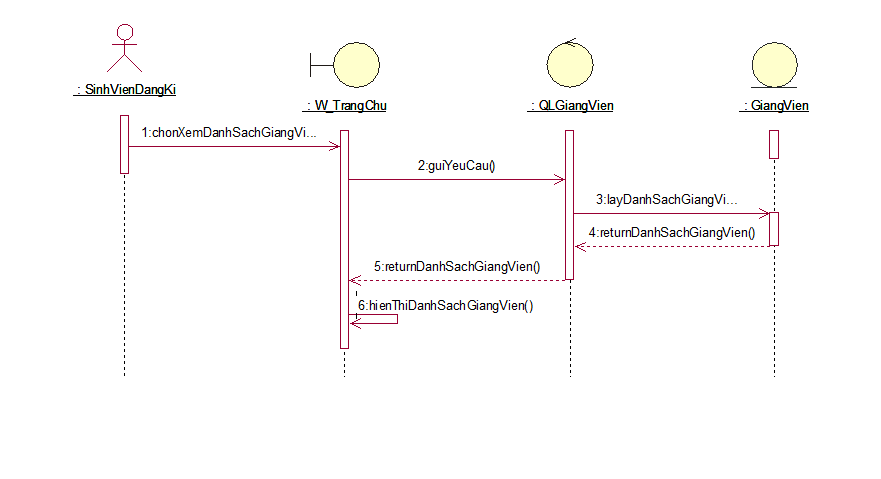
Hình 2.6 Biểu đồ tuần tự xem khóa học

1. **Biểu đồ tuần tự đăng kí khóa học**



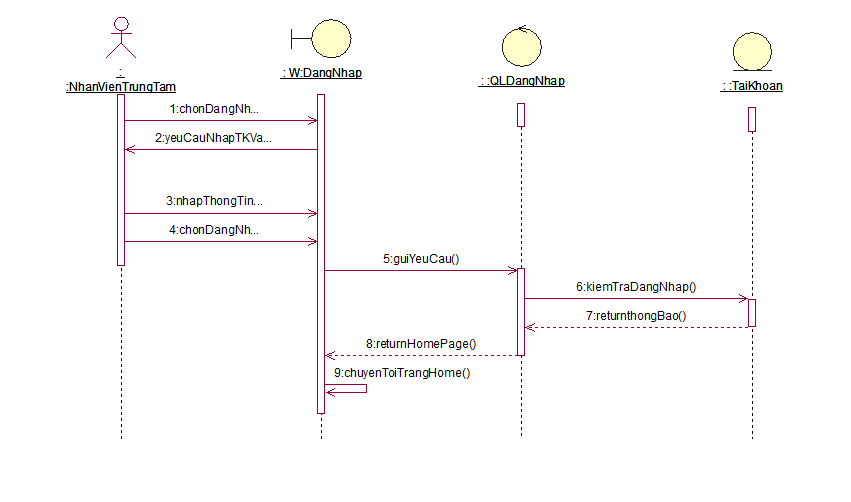
Hình 2.7 Biểu đồ tuần tự đăng kí khóa học

1. **Biểu đồ tuần tự xem danh sách giảng viên**



Hình 2.8 Biểu đồ tuần tự xem danh sách sinh viên

1. **Biểu đồ tuần tự đăng nhập**



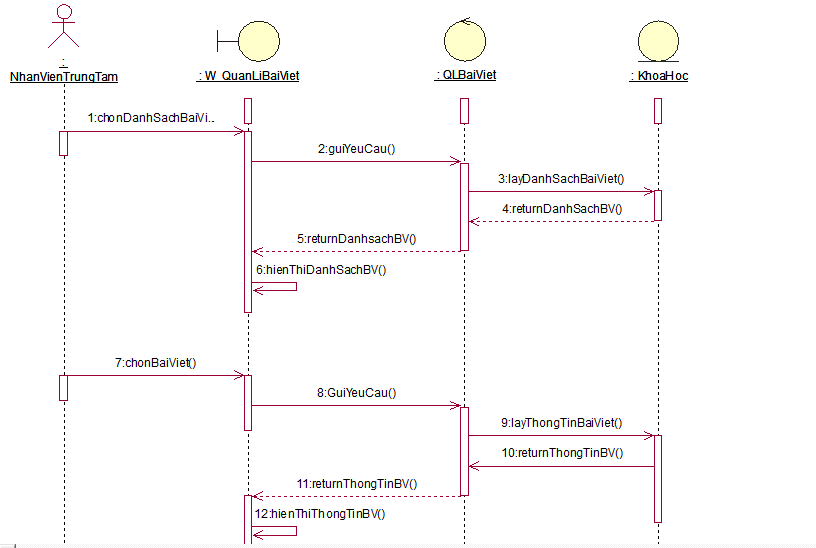
Hình 2.9 Biểu đồ tuần tự đăng nhập

1. **Biểu đồ tuần tự quản lí hồ sơ**



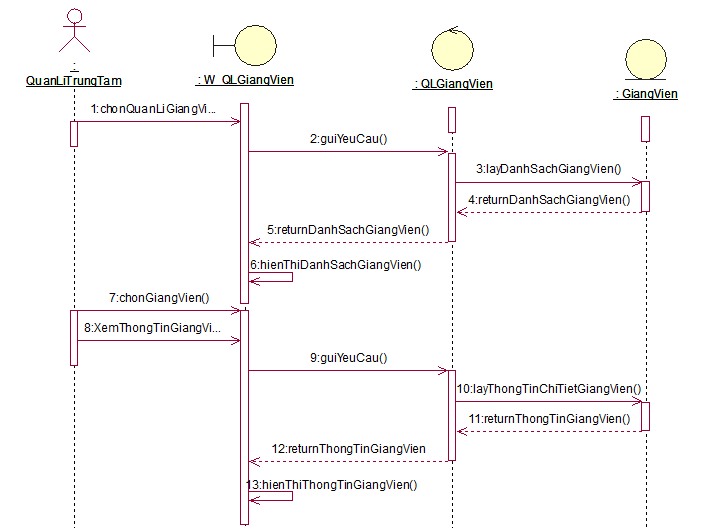
Hình 2.10 Biểu đồ tuần tự quản lí hồ sơ

1. **Biểu đồ tuần tự quản lí bài viết**



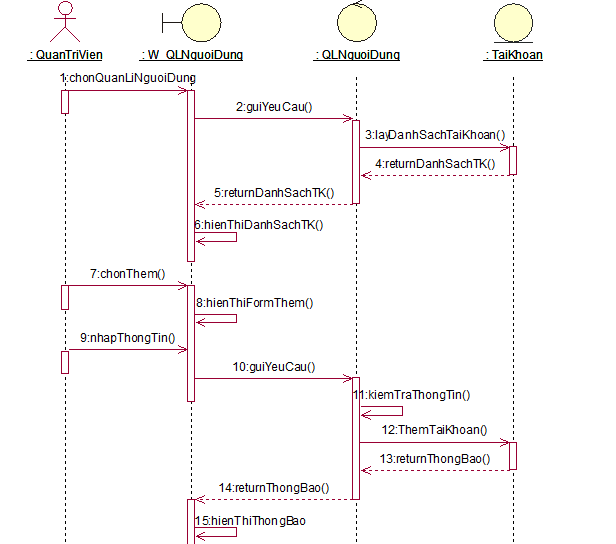
Hình 2.11 Biểu đồ tuần tự quản lí bài viết

1. **Biểu đồ tuần tự quản lí giáo viên**



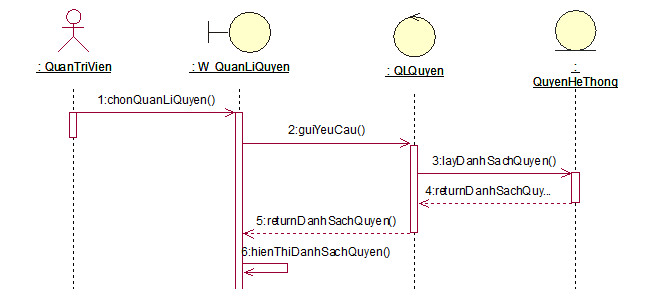
Hình 2.12 Biểu đồ tuần tự quản lí giáo viên

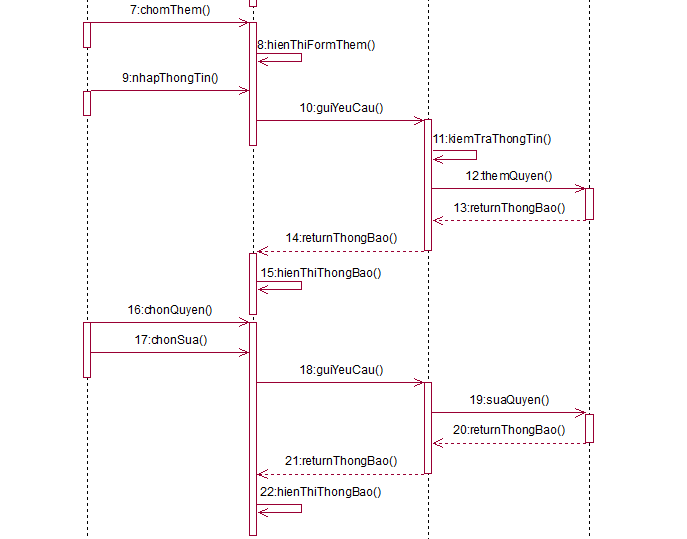
1. **Biểu đồ tuần tự quản lí người dùng**



Hình 2.13 Biểu đồ tuần tự quản lí người dùng

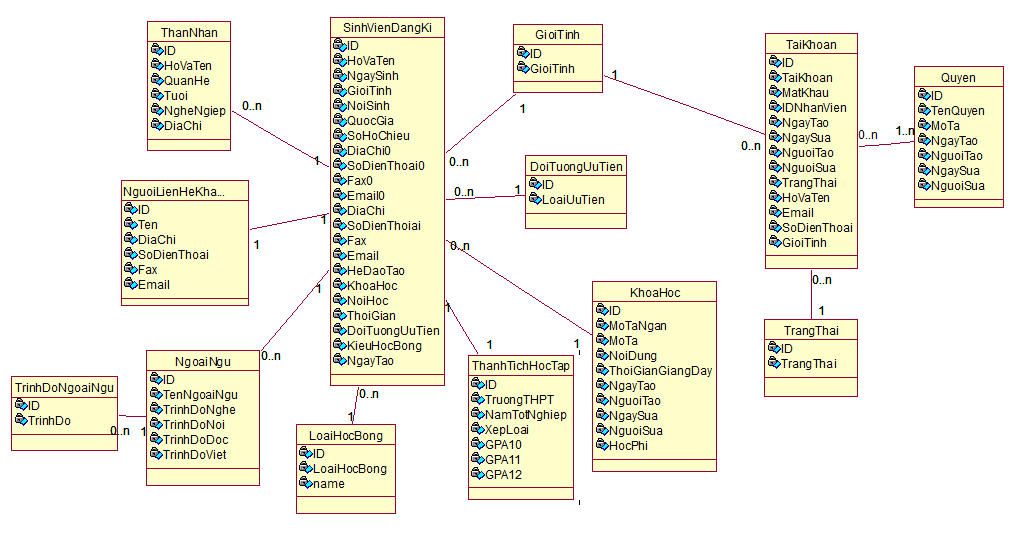
1. **Biểu đồ tuần tự quản lí quyền**





Hình 2.14 Biểu đồ tuần tự quản lí quyền

1. **Biểu đồ lớp**
2. **Biểu đồ lớp**

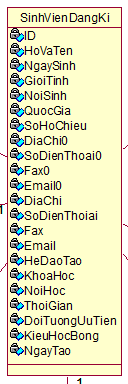


Hình 2.15 Biểu đồ lớp

1. **Xác định các lớp**

* Lớp sinh viên đăng kí lưu thông tin của sinh viên đăng kí khóa học, gồm các thông tin sau

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | HoVaTen | Họ và tên đầy đủ |
| 3 | NgaySinh | Ngày tháng năm sinh |
| 4 | GioiTinh | Giới tính |
| 5 | NoiSinh | Nơi sinh |
| 6 | QuocGia | Quốc gia |
| 7 | SoHoChieu | Số hộ chiếu |
| 8 | DiaChi0 | Địa chỉ trước khi tới Việt Nam |
| 9 | SoDienThoai0 | Số điện thoại trước khi tới Việt Nam |
| 10 | Fax0 | Số fax trước khi tới Việt Nam |
| 11 | Email0 | Địa chỉ email trước khi tới Việt Nam |
| 12 | DiaChi | Địa chỉ ở Việt Nam |
| 13 | SoDienThoai | Số điện thoại ở Việt Nam |
| 14 | Fax | Số fax ở Việt Nam |
| 15 | Email | Địa chỉ email ở Việt Nam |
| 16 | HeDaoTao | Hệ đào tạo |
| 17 | KhoaHoc | Khóa học muốn tham gia |
| 18 | NoiHoc | Chi nhánh muốn học |
| 19 | ThoiGian0 | Thời gian bắt đầu học |
| 20 | DoiTuongUuTien | Đối tượng ưu tiên |
| 21 | KieuHocBong | Kiểu học bổng |
| 22 | NgayTao | Ngày tạo hồ sơ |



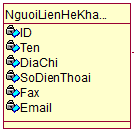
* Lớp thân nhân lưu thông tin người thân của sinh viên đăng kí học, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | HoVaTen | Họ và tên |
| 3 | QuanHe | Mối quan hệ với sinh viên đăng kí học |
| 4 | Tuoi | Tuổi |
| 5 | NgeNghiep | Công việc |
| 6 | DiaChi | Địa chỉ |



* Lớp người liên hệ khẩn cấp lưu thông tin người liên hệ trong tình huống khẩn cấp, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | Ten | Tên |
| 3 | DiaChi | Nơi ở |
| 4 | SoDienThoai | Số điện thoại |
| 5 | Fax | Số fax |
| 6 | Email | Địa chỉ email |



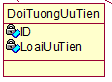
* Lớp giới tính quy ước mã định danh giới tính, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | GioiTinh | Giới tính |



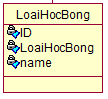
* Lớp đối tượng ưu tiên lưu trữ thông tin về các diện được ưu tiên, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | LoaiUuTien | Tên ngôn ngữ |
| 3 | MoTa | Mô tả về diện ưu tiên, chính sách ưu tiên… |



* Lớp loại học bổng lưu trữ thông tin về các loại học bổng của trung tâm , gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | LoaiHocBong | Tên ngôn ngữ |
| 3 | MoTa | Mô tả học bổng |



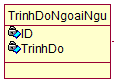
* Lớp ngoại ngữ lưu trữ thông tin trình độ ngoại ngữ của sinh viên, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | TenNgoaiNgu | Tên ngôn ngữ |
| 3 | TrinhDoNghe | Trình độ nghe |
| 4 | TrinhDoNoi | Trình độ nói |
| 5 | TrinhDoDoc | Trình độ đọc |
| 6 | TrinhDoViet | Trình độ viết |



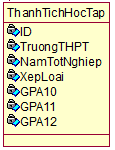
* Lớp trình độ ngoại ngữ quy ước các mức trình độ, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | TrinhDo | Trình Độ |
| 3 | MoTa | Mô tả các mức trình độ ngoại ngữ |



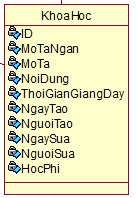
* Lớp thành tích học tập lưu trữ kết quả học tập sinh viên ở THPT, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | TruongTHPT | Tên trường trung học phổ thông |
| 3 | NamTotNghiep | Năm tốt nghiệp |
| 4 | XepLoai | Xếp loại tốt nghiệp |
| 5 | GPA10 | Điểm tổng kết lớp 10 |
| 6 | GPA11 | Điểm tổng kết lớp 11 |
| 7 | GPA11 | Điểm tổng kết lớp 12 |



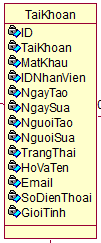
* Lớp khóa học lưu trữ thông các khóa của trung tâm, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | MoTaNgan | Mô tả ngắn |
| 3 | MoTa | Năm tốt nghiệp |
| 4 | NoiDung | Nội dung khoa học |
| 5 | ThoiGianGiangDay | Thời gian giảng dạy |
| 6 | NgayTao | Ngày tạo khóa học trên website |
| 7 | NguoiTao | Người tạo khóa học |
| 8 | NgaySua | Ngày sửa thông tin khóa học |
| 9 | NguoiSua | Người sửa thông tin khóa học |
| 10 | HocPhi | Học phí |



* Lớp tài khoản lưu trữ thông tin của tài khoản của người dùng trong hệ thống, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | TaiKhoan | Tài khoản đăng nhập |
| 3 | MatKhau | Mật khẩu đăng nhập |
| 4 | HoVaTen | HoVaTen |
| 5 | NgayTao | Ngày tao tài khoản |
| 6 | NguoiTao | Người tạo tài khoản |
| 7 | NgaySua | Ngày sửa thông tin tai khoản |
| 8 | NguoiSua | Người sửa thông tin tài khoản |
| 9 | Email | Thư điện tử |
| 10 | TrangThai | Trạng thái hoat động |
| 11 | SoDienThoai | Số điện thoại |
| 12 | GioiTinh | Giới tính |



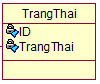
* Lớp quyền lưu trữ danh sách quyền của hệ thống, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | TenQuyen | Tên quyền |
| 3 | MoTa | Mô tả |
| 4 | NgayTao | Ngày tạo quyền |
| 5 | NguoiTao | Người tạo |
| 6 | NgaySua | Ngày sửa |
| 7 | NguoiSua | Người sửa |



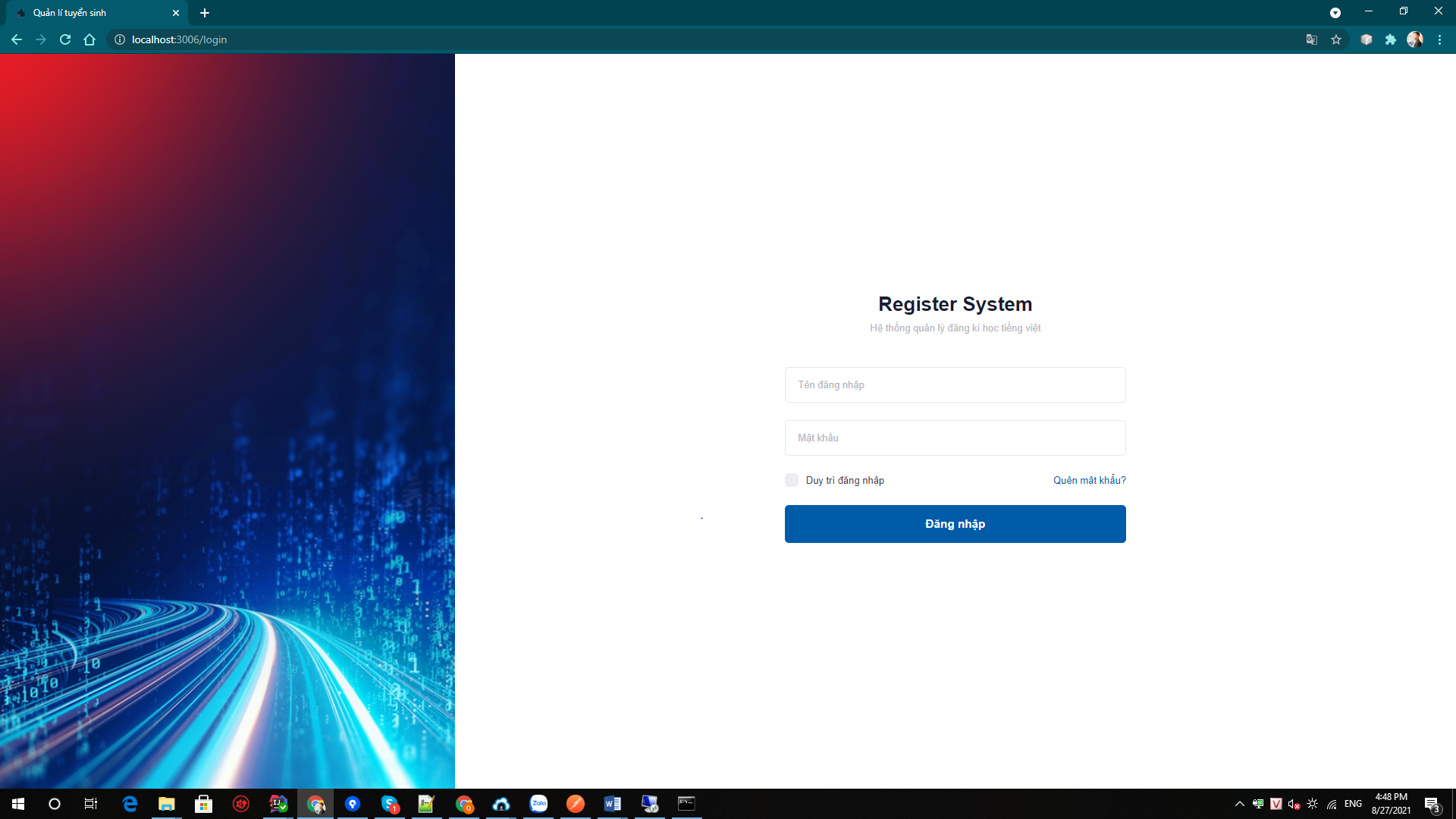
* Lớp trạng thái lưu trữ các kiểu trạng thái hoạt động, gồm những thông tin sau:

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | ID | Mã định danh |
| 2 | TrangThai | Trạng thái hoạt động |
| 3 | MoTa | Mô tả trạng thái |

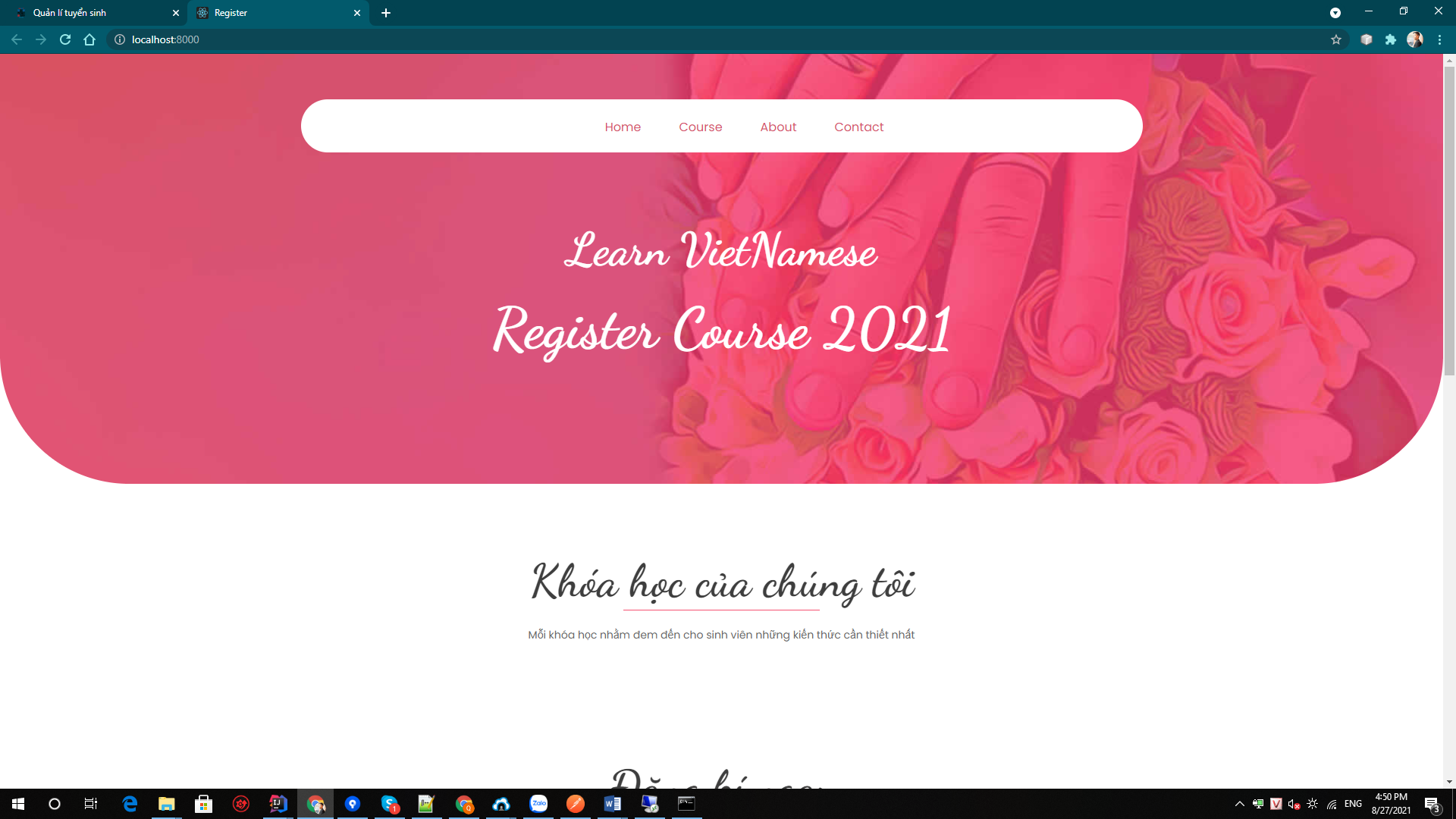


# CHƯƠNG 3: THIẾT KẾ HỆ THỐNG

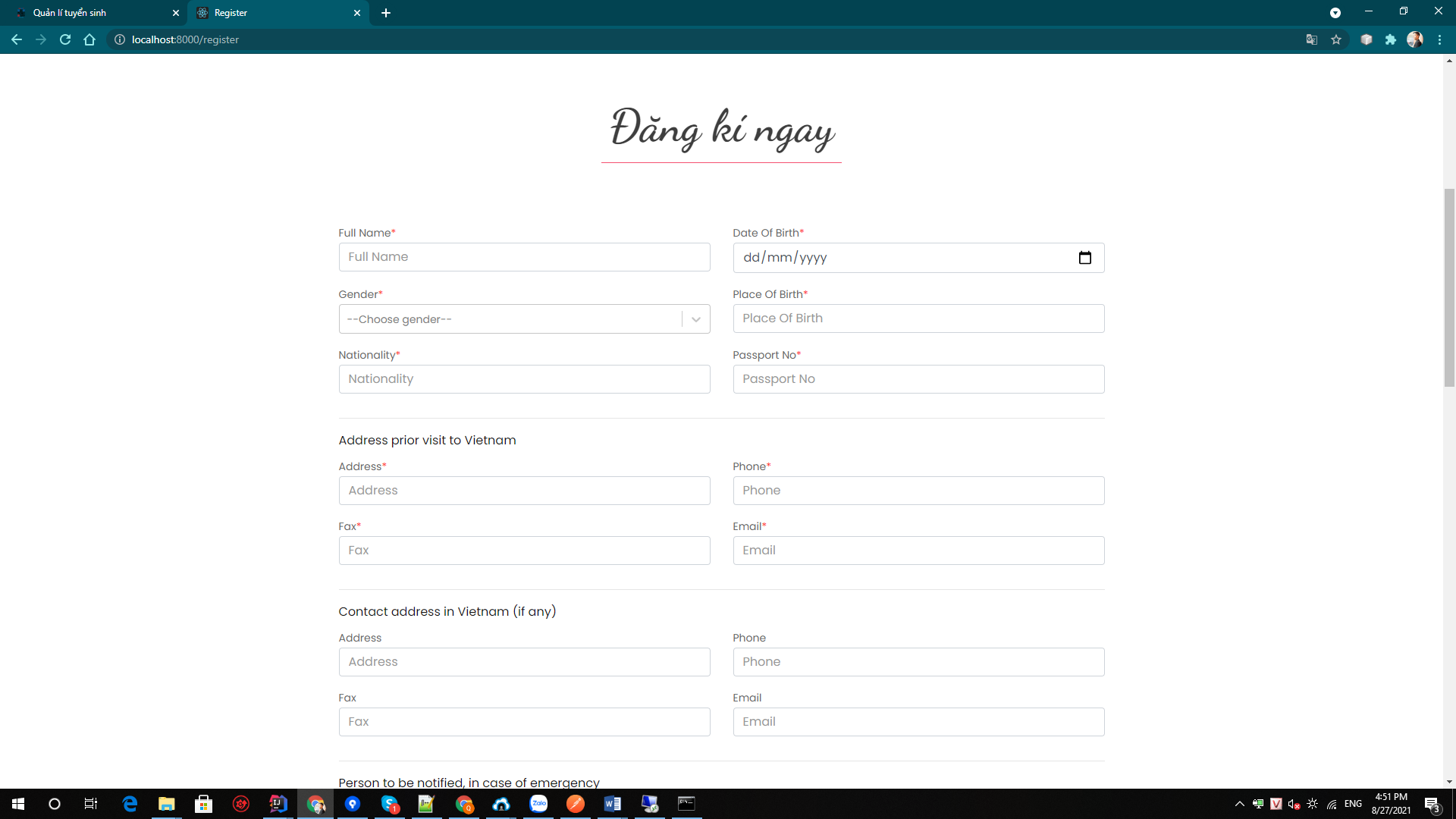
1. **Thiết kế giao diện**
2. **Giao diện trang đăng nhập**



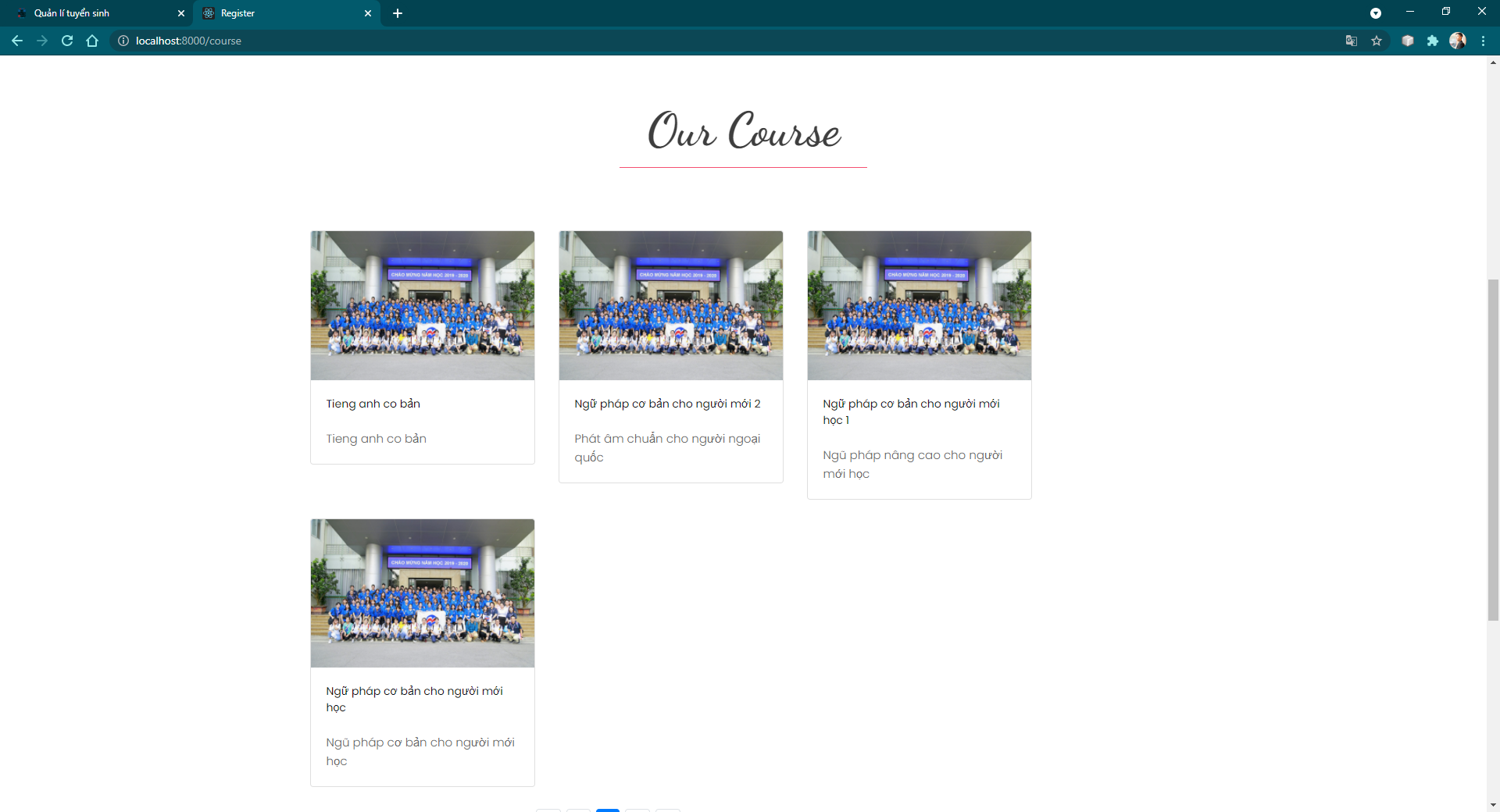
1. **Giao diện trang chủ web**



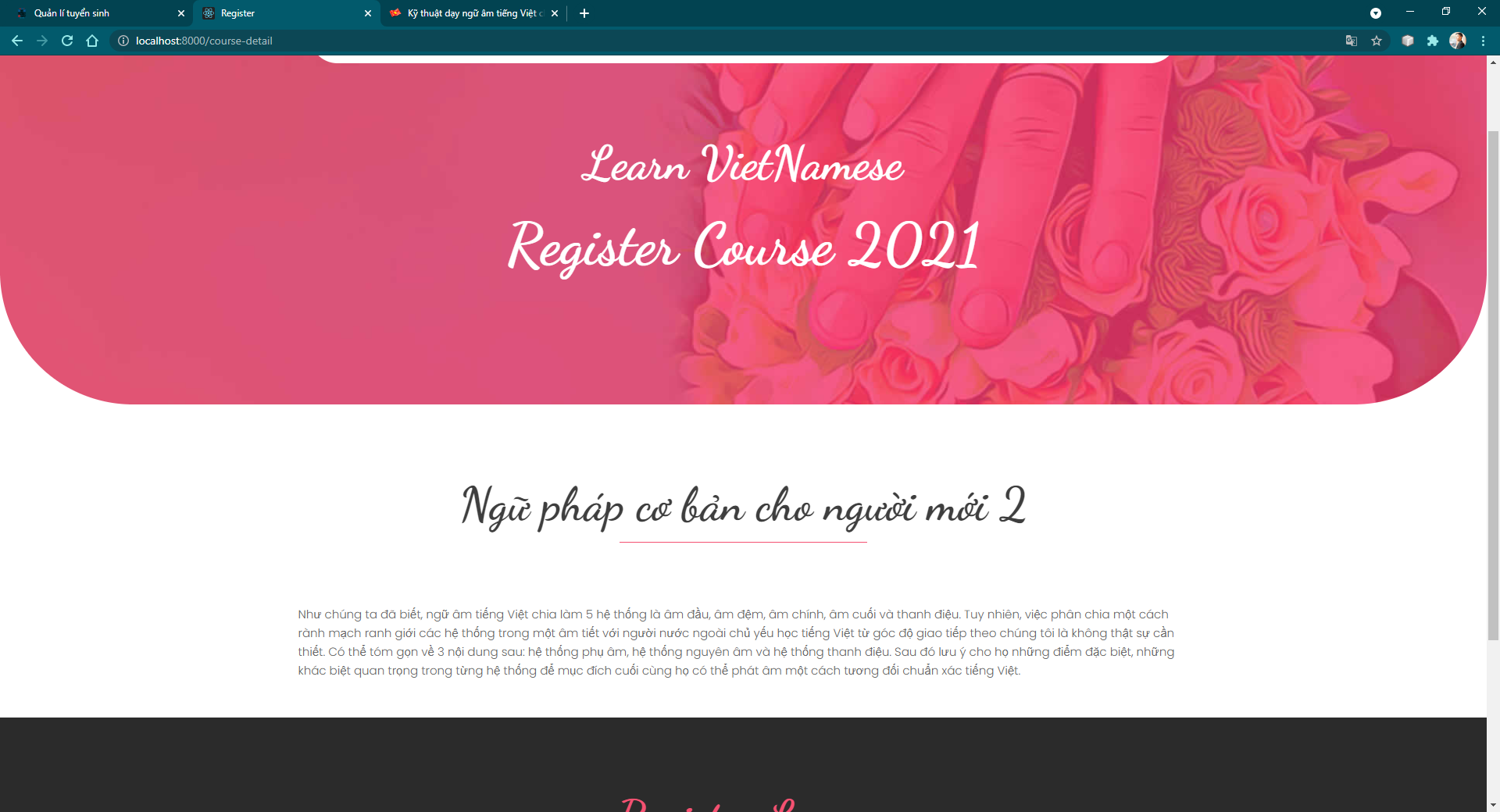
1. **Giao diện trang đăng kí khóa học web**



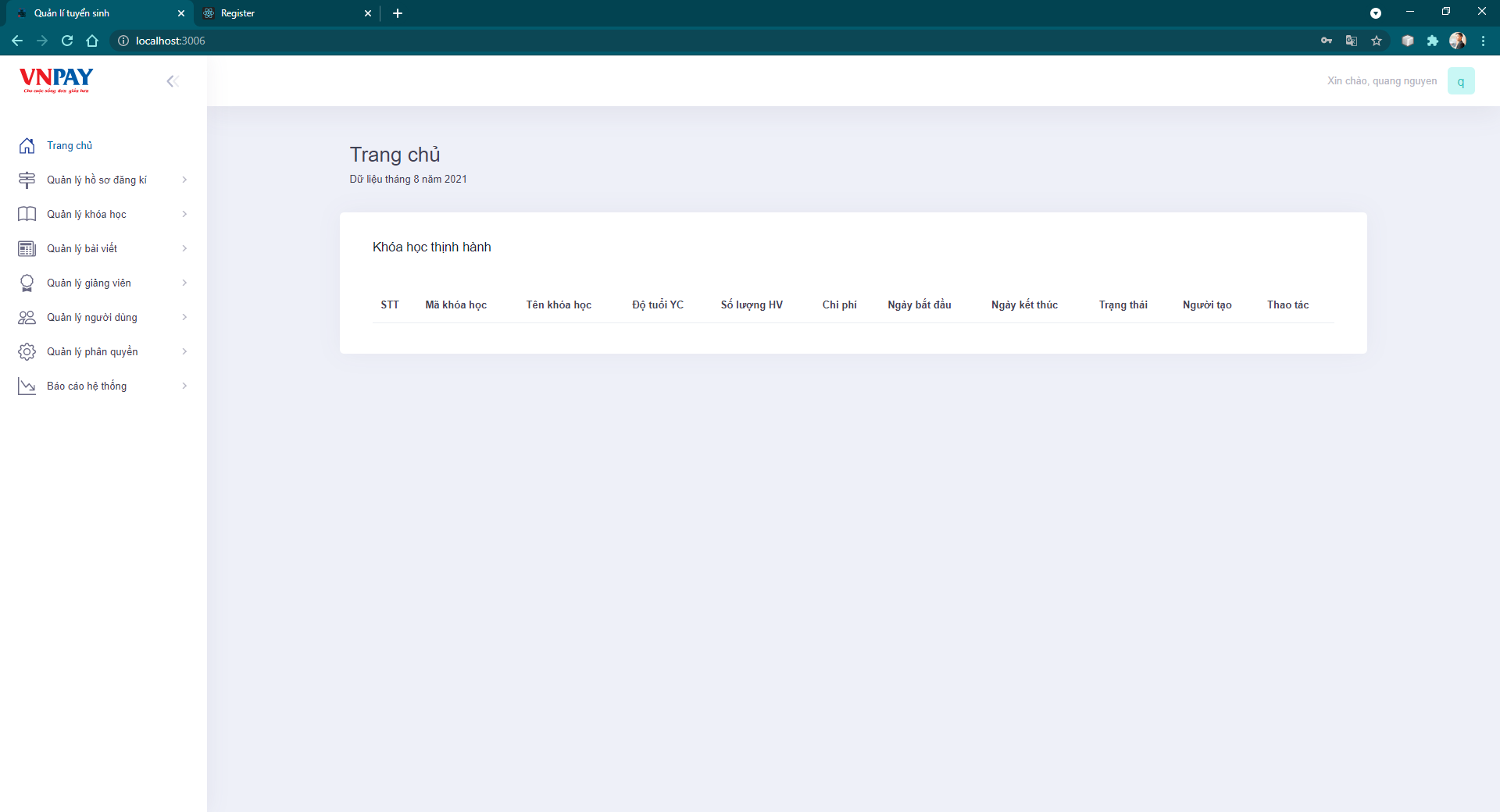
1. **Giao diện danh sách khóa học web**



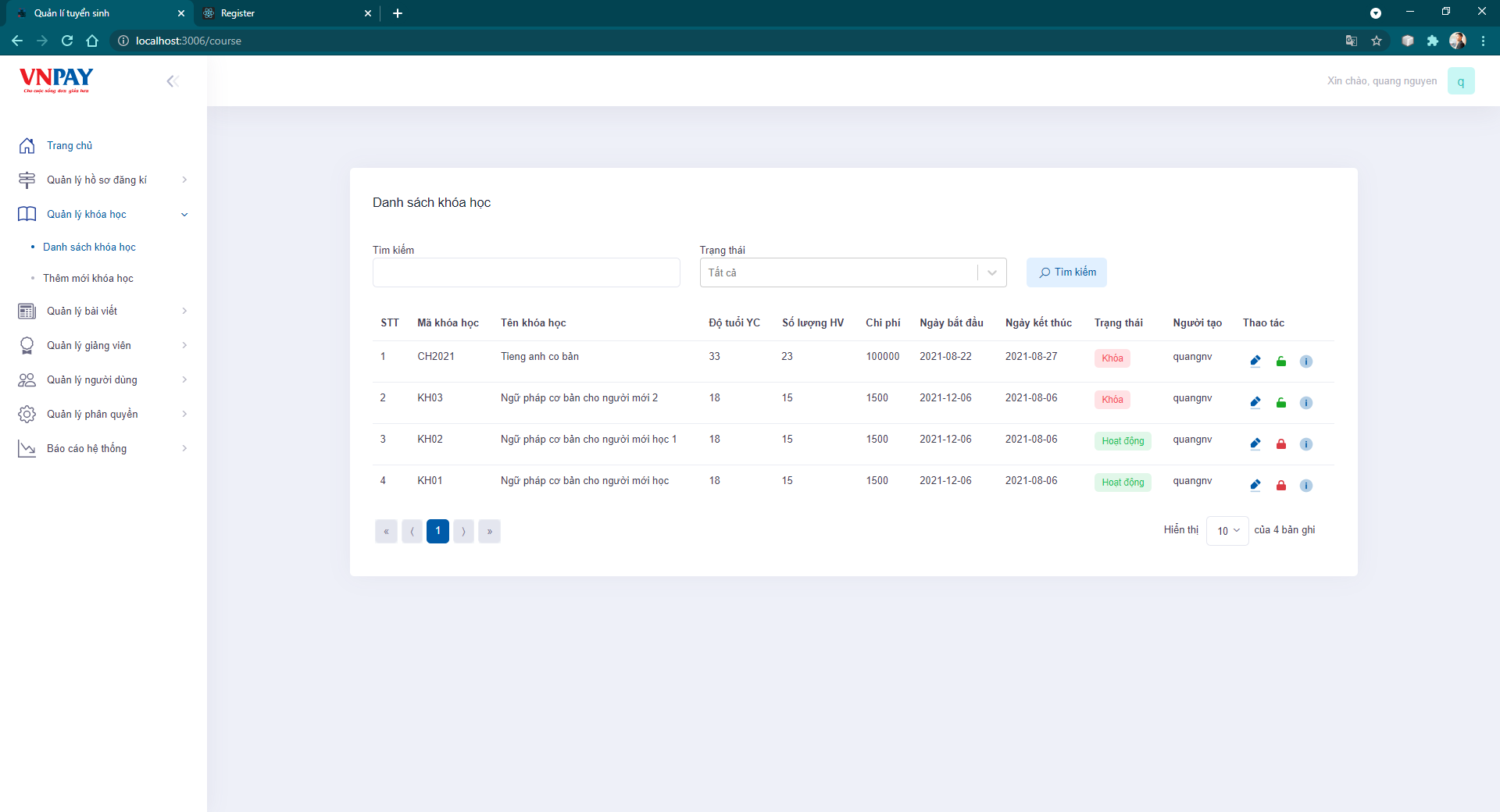
1. **Giao diện chi tiết khóa học web**



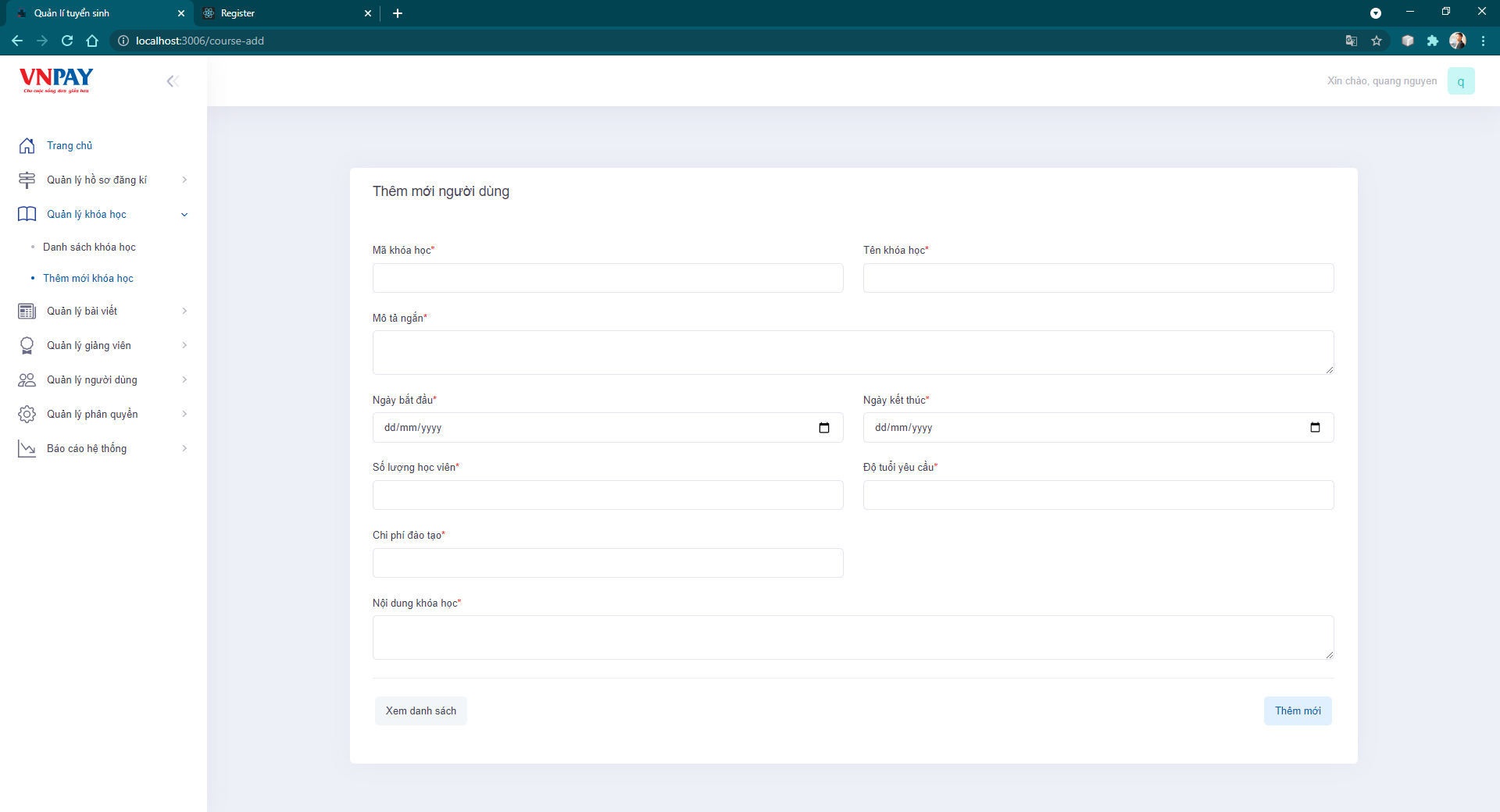
1. **Giao diện trang chủ quản trị**



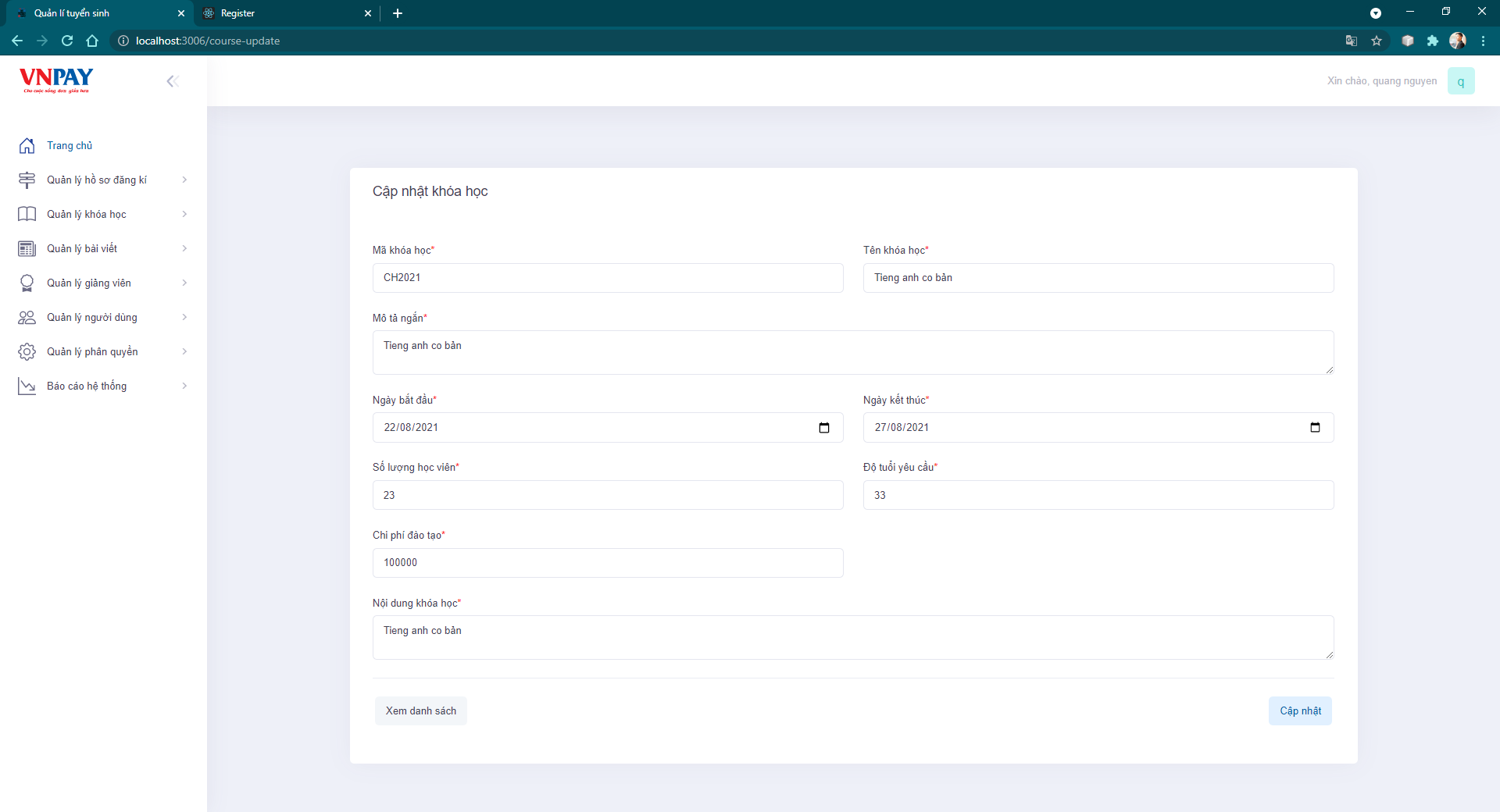
1. **Giao diện trang quản lí khóa học**

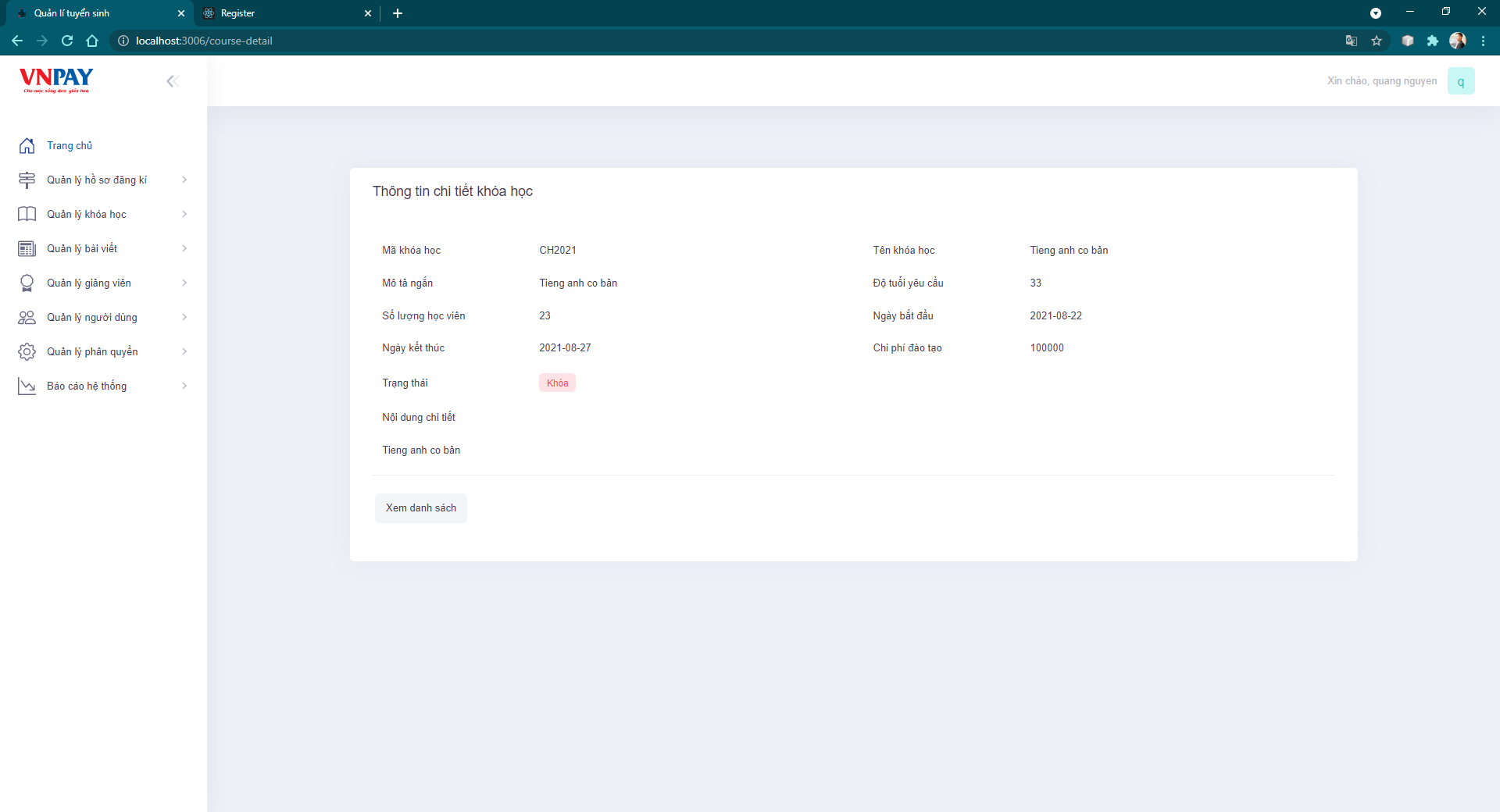


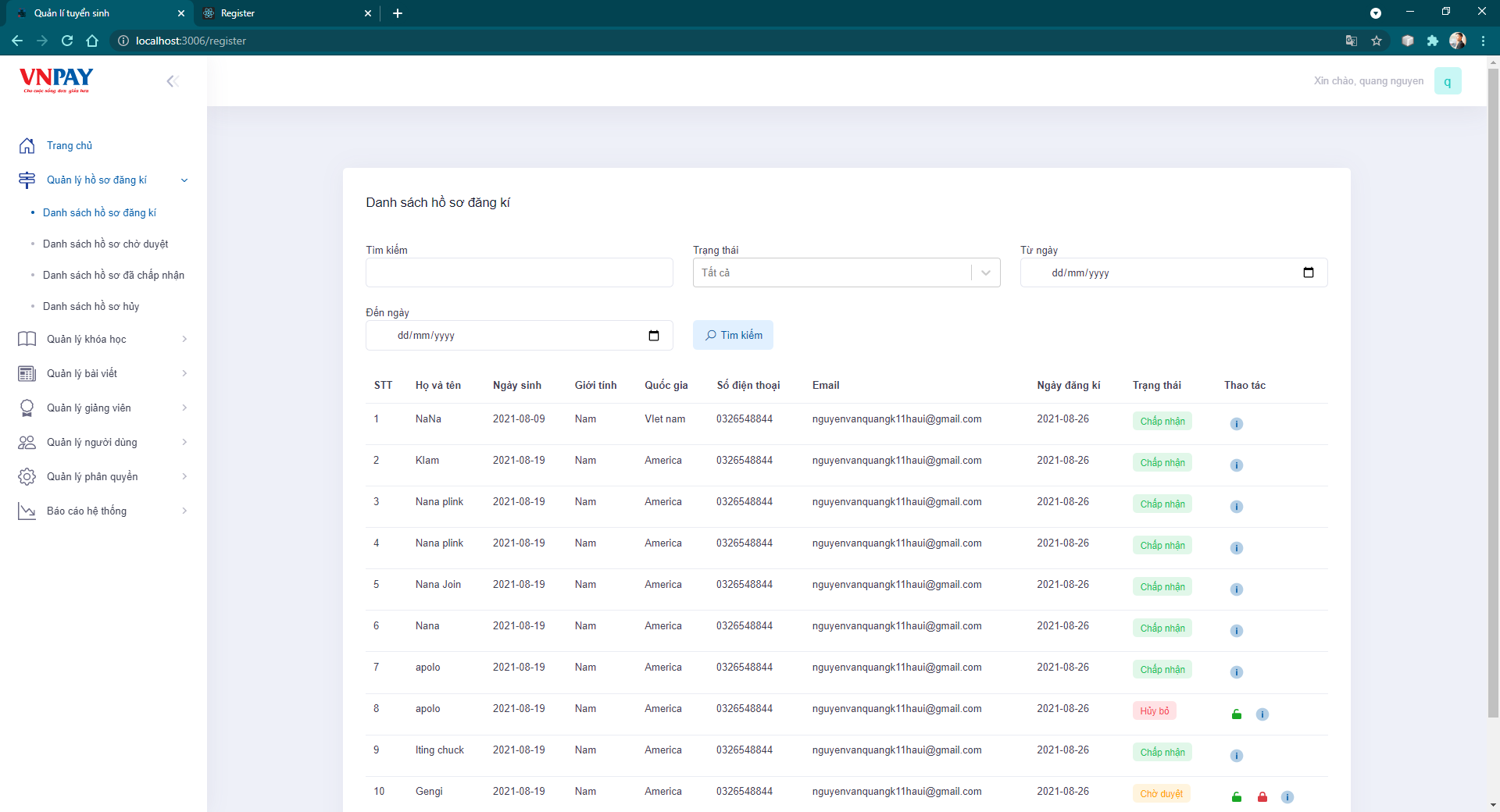
1. **Giao diện thêm mới khóa học**

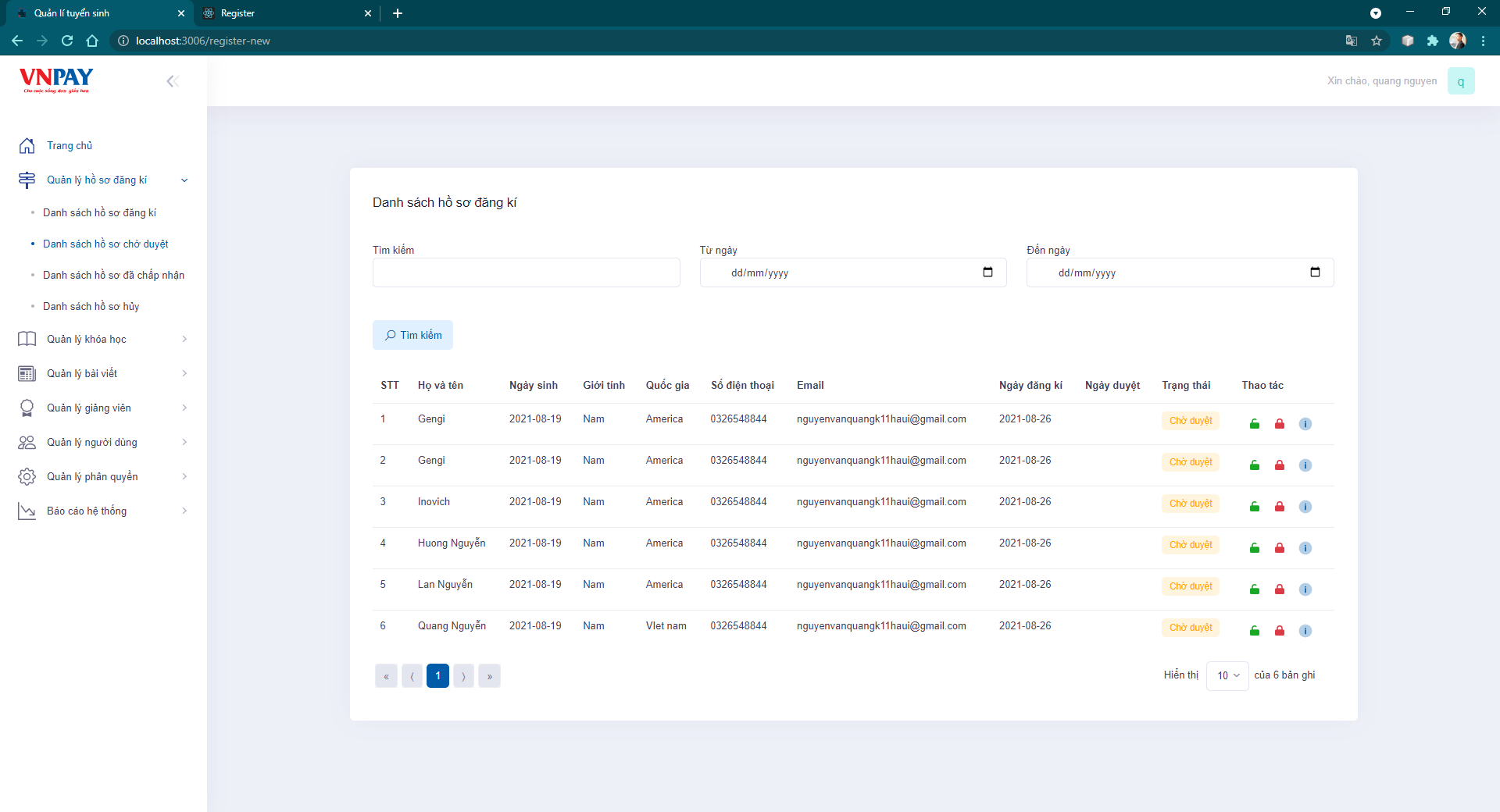
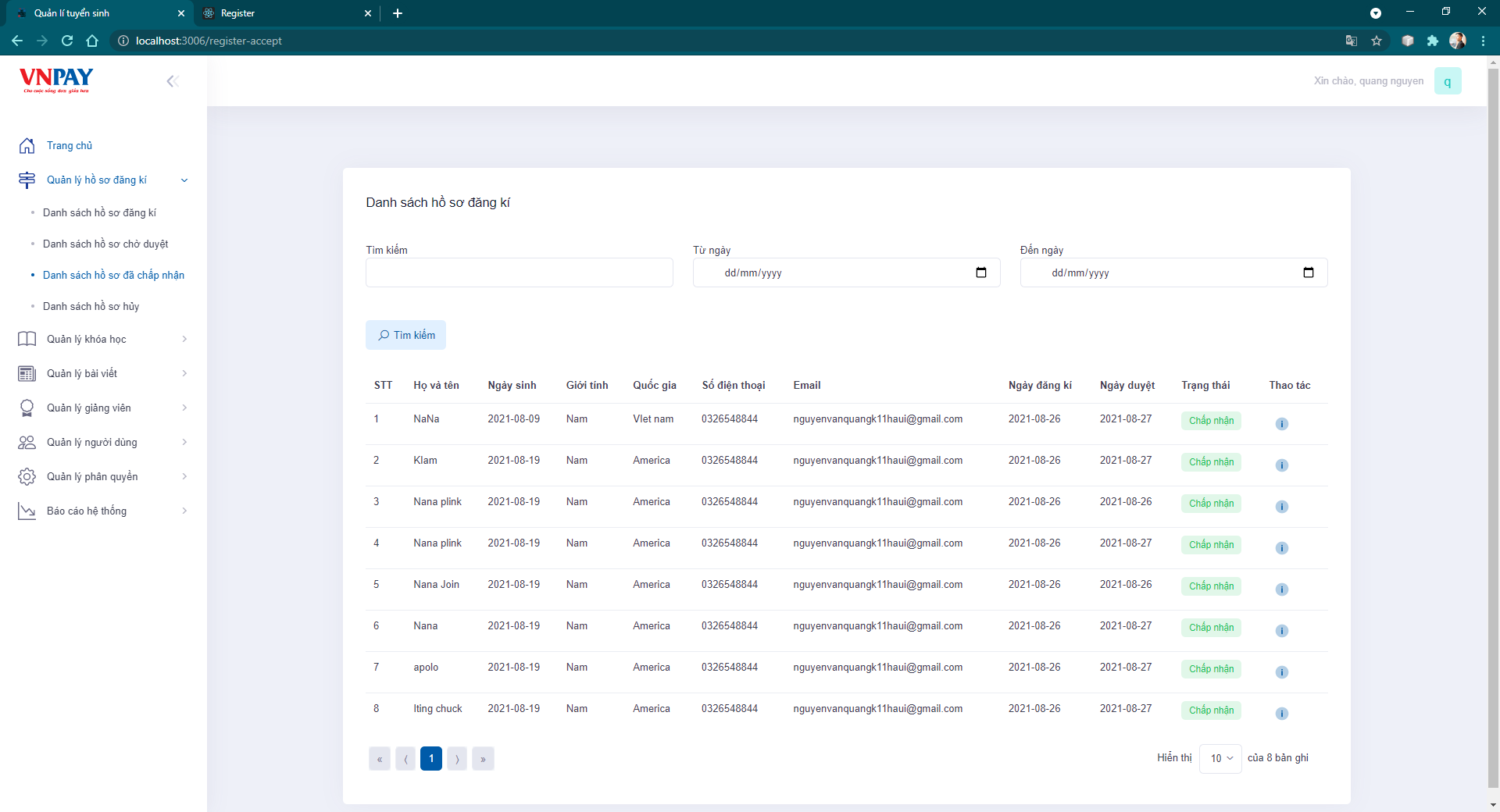
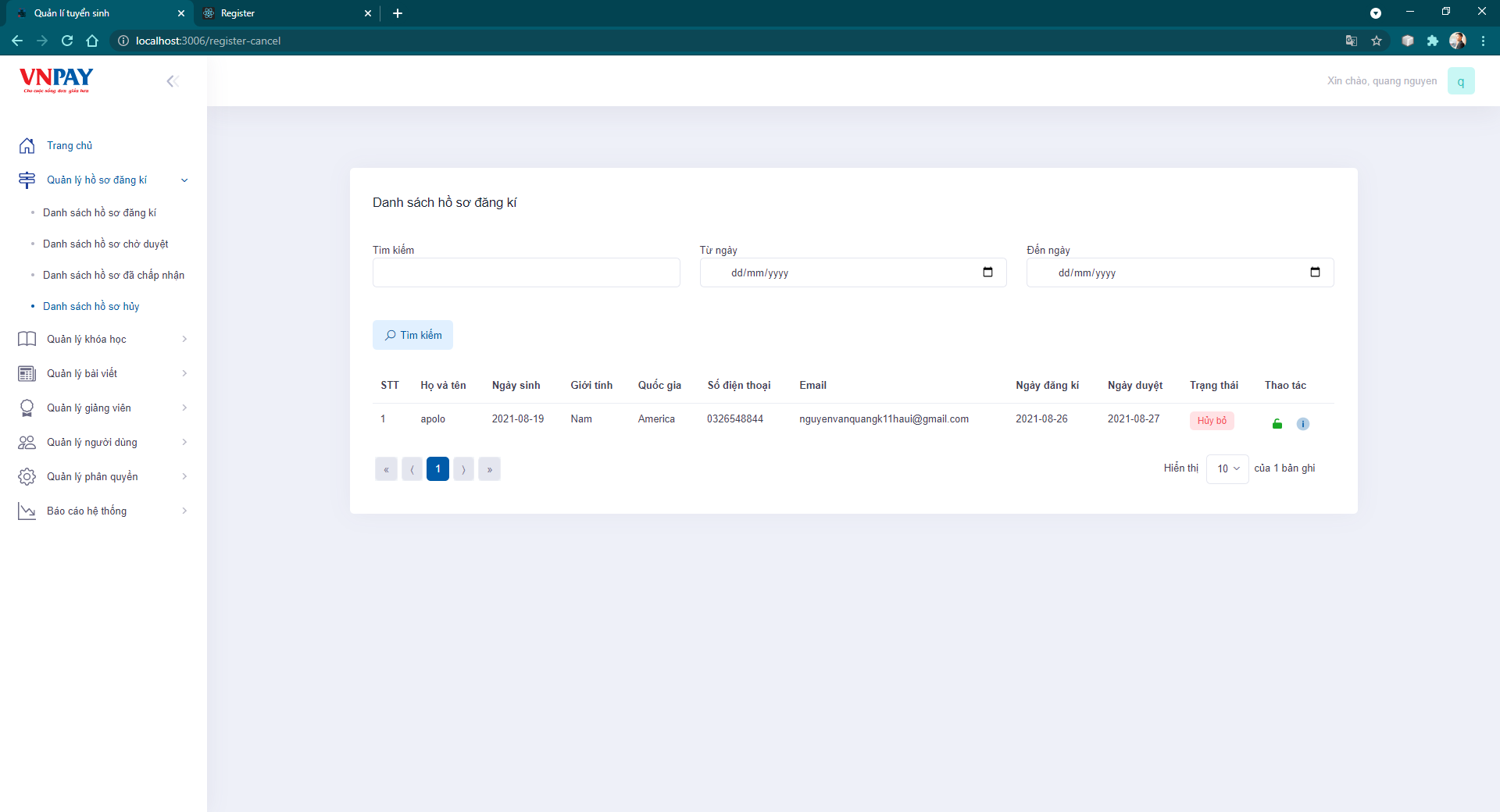


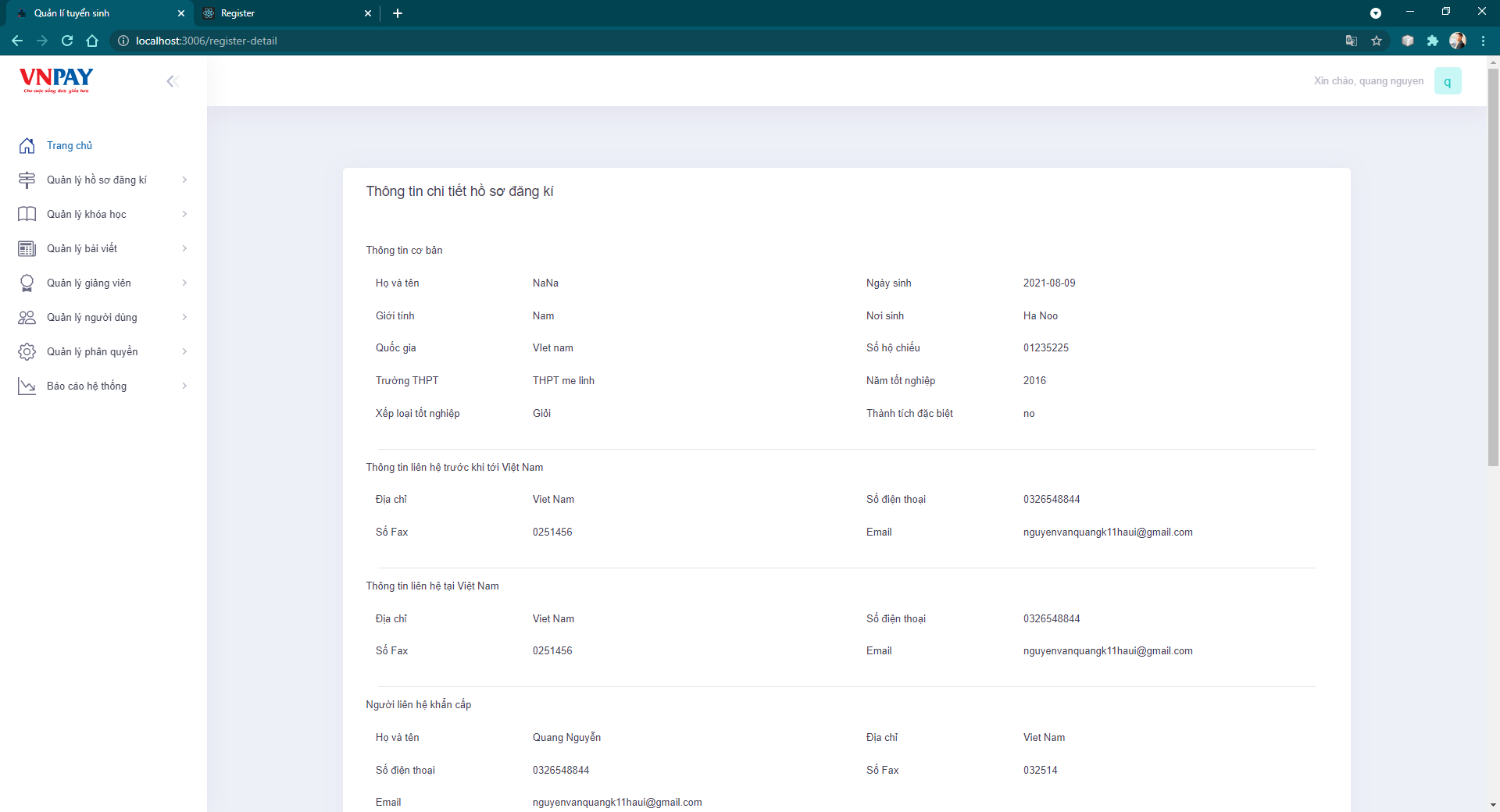
1. **Giao diện cập nhật thông tin khóa học**



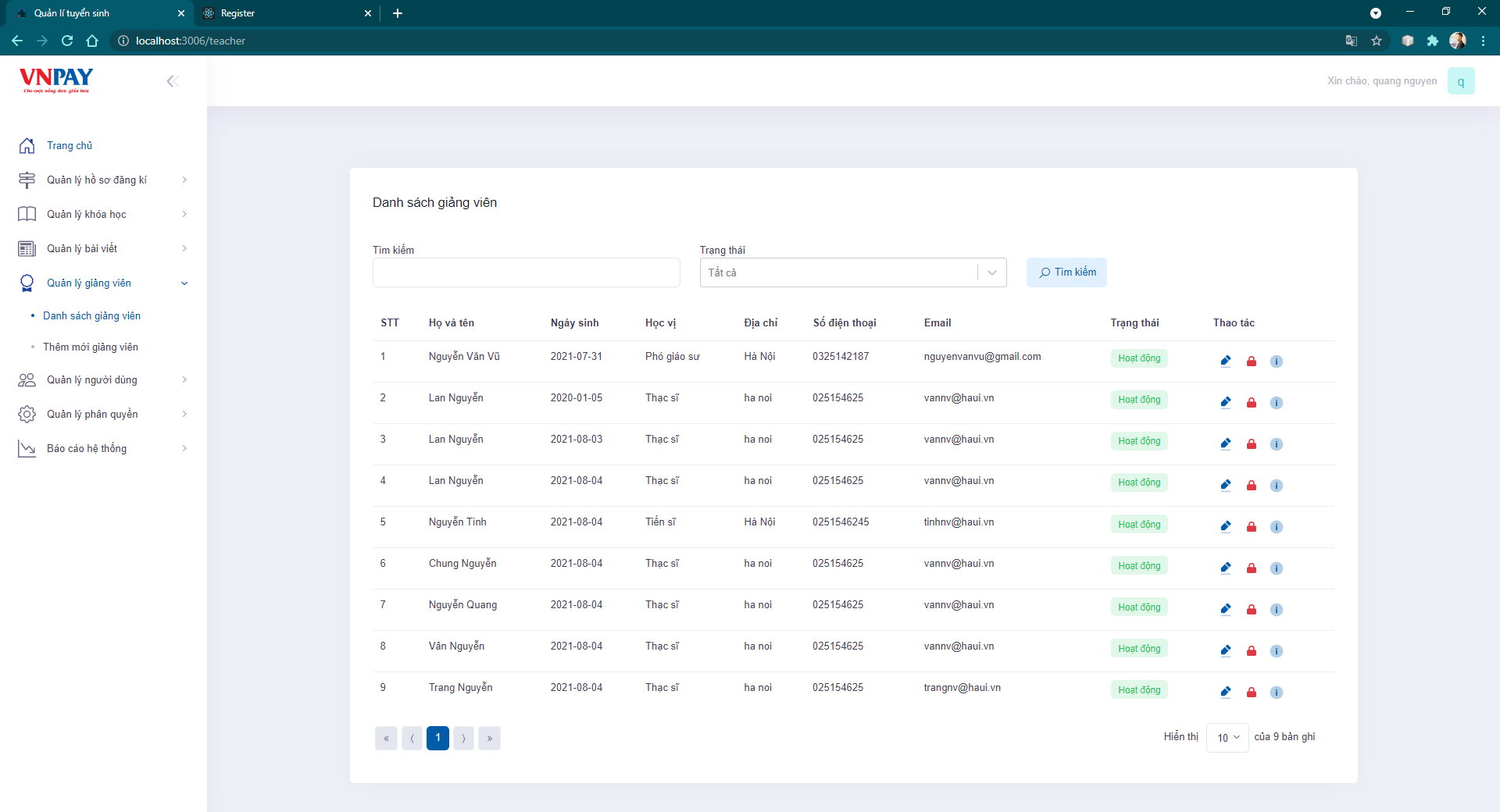
1. **Giao diện chi tiết khóa học**
2. **Giao diện quản lí thông tin đăng kí**



1. **Giao diện thông tin đăng kí chờ duyệt**
2. **Giao diện thông tin đã chấp nhận**
3. **Giao diện thông tin đăng kí đã hủy**
4. **Giao diện trang thông tin chi tiết đăng kí**



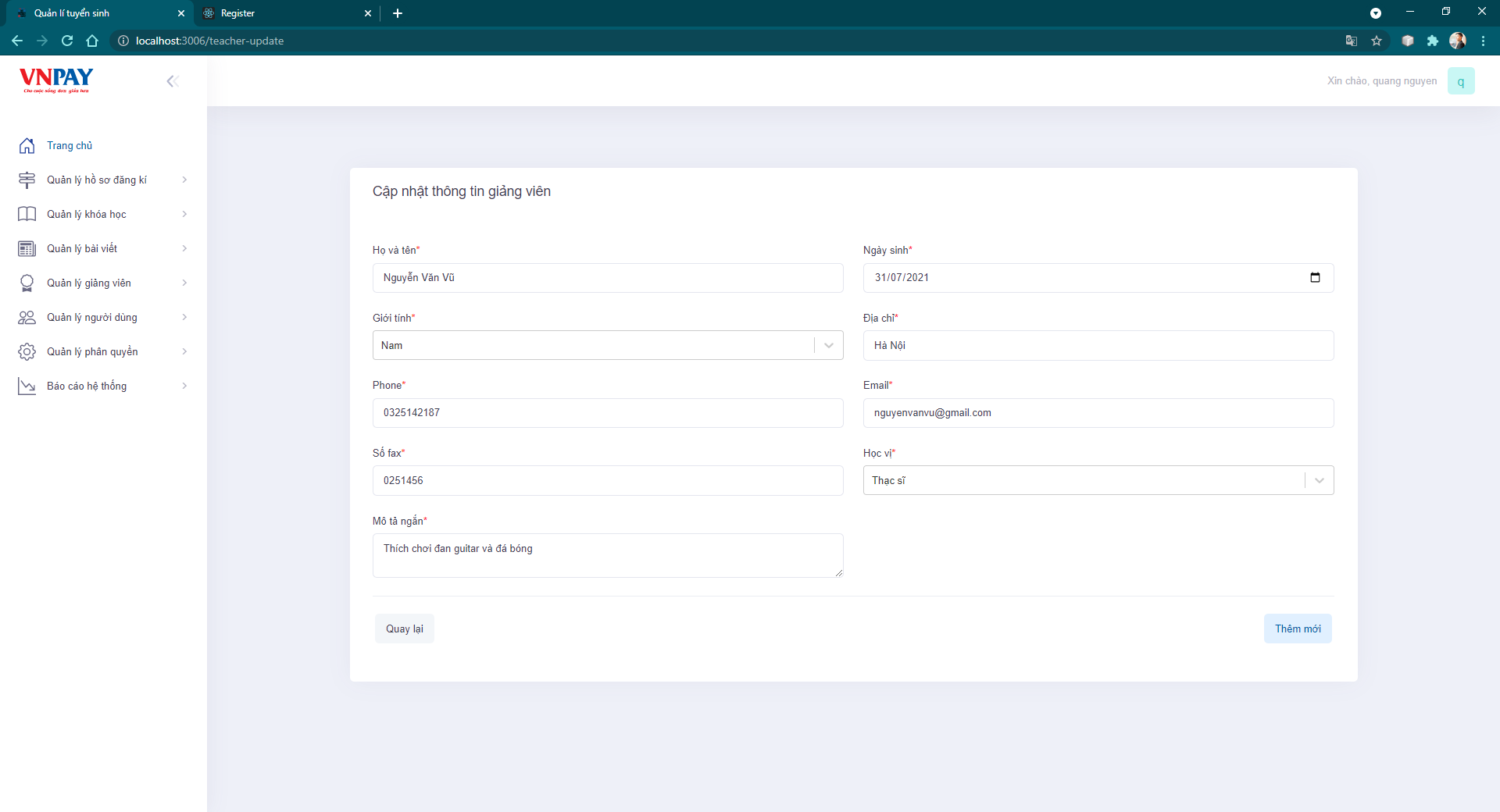
1. **Giao diện quản lí giảng viên**

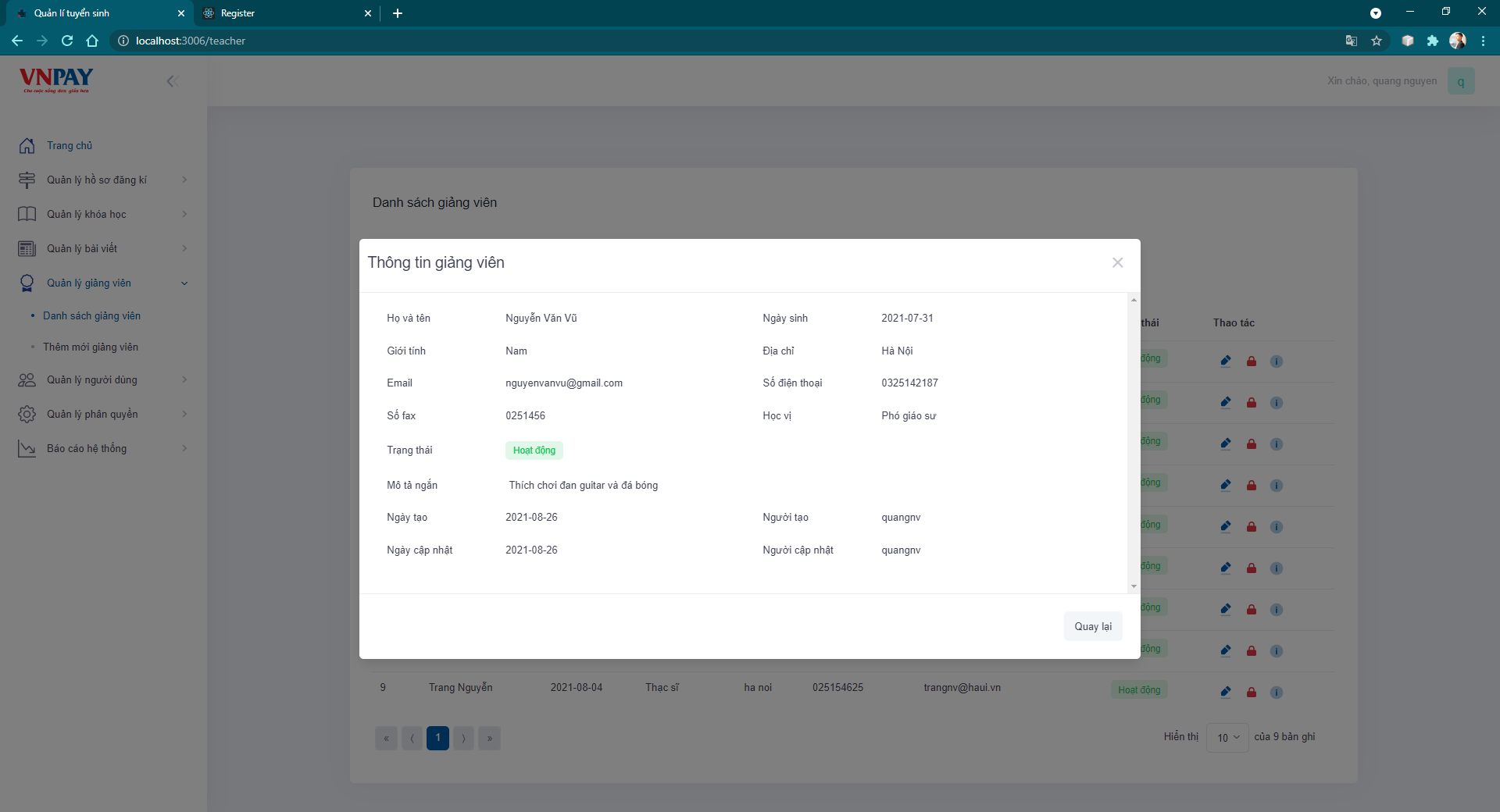


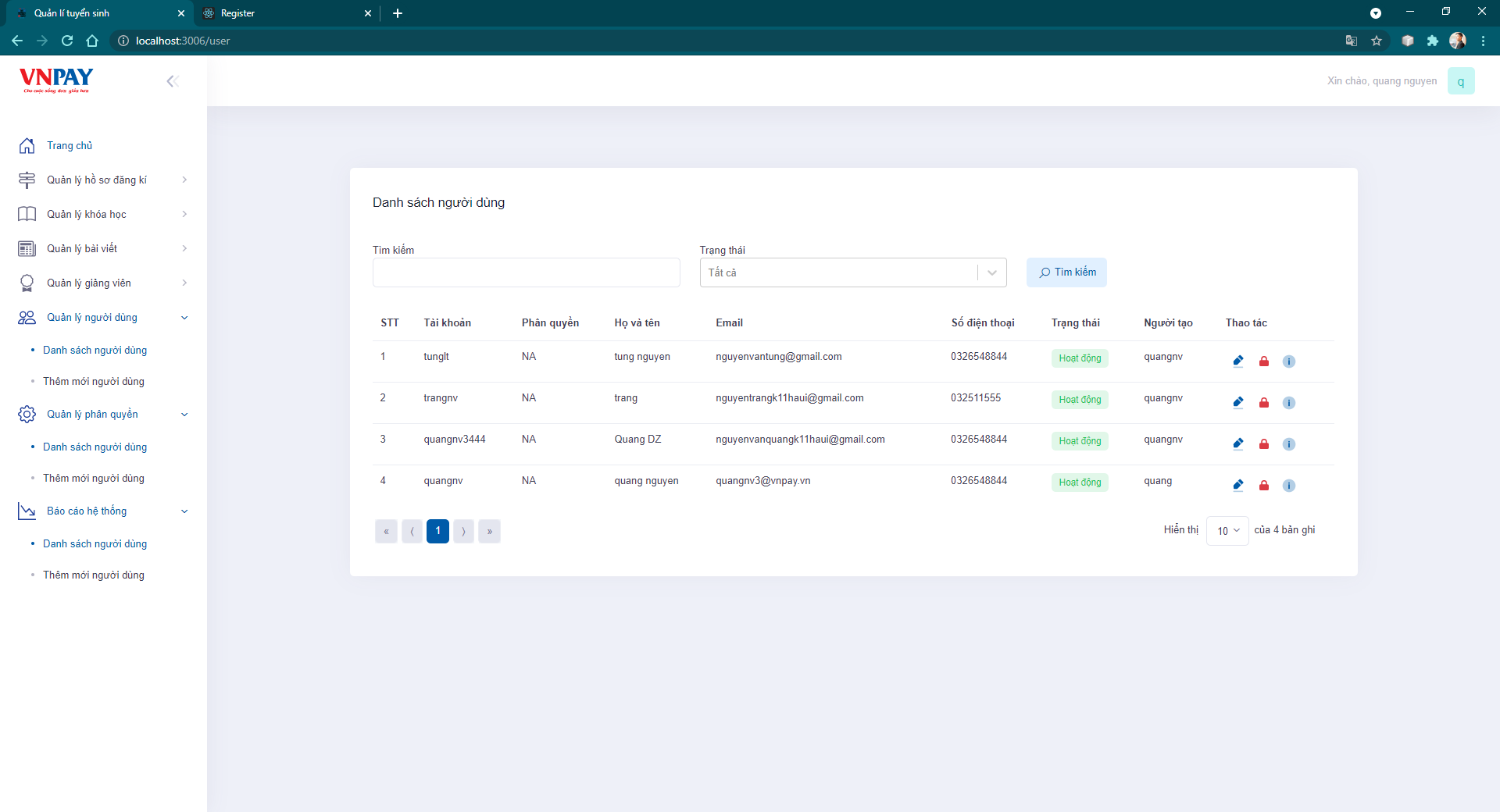
1. **Giao diện thêm mới giảng viên**



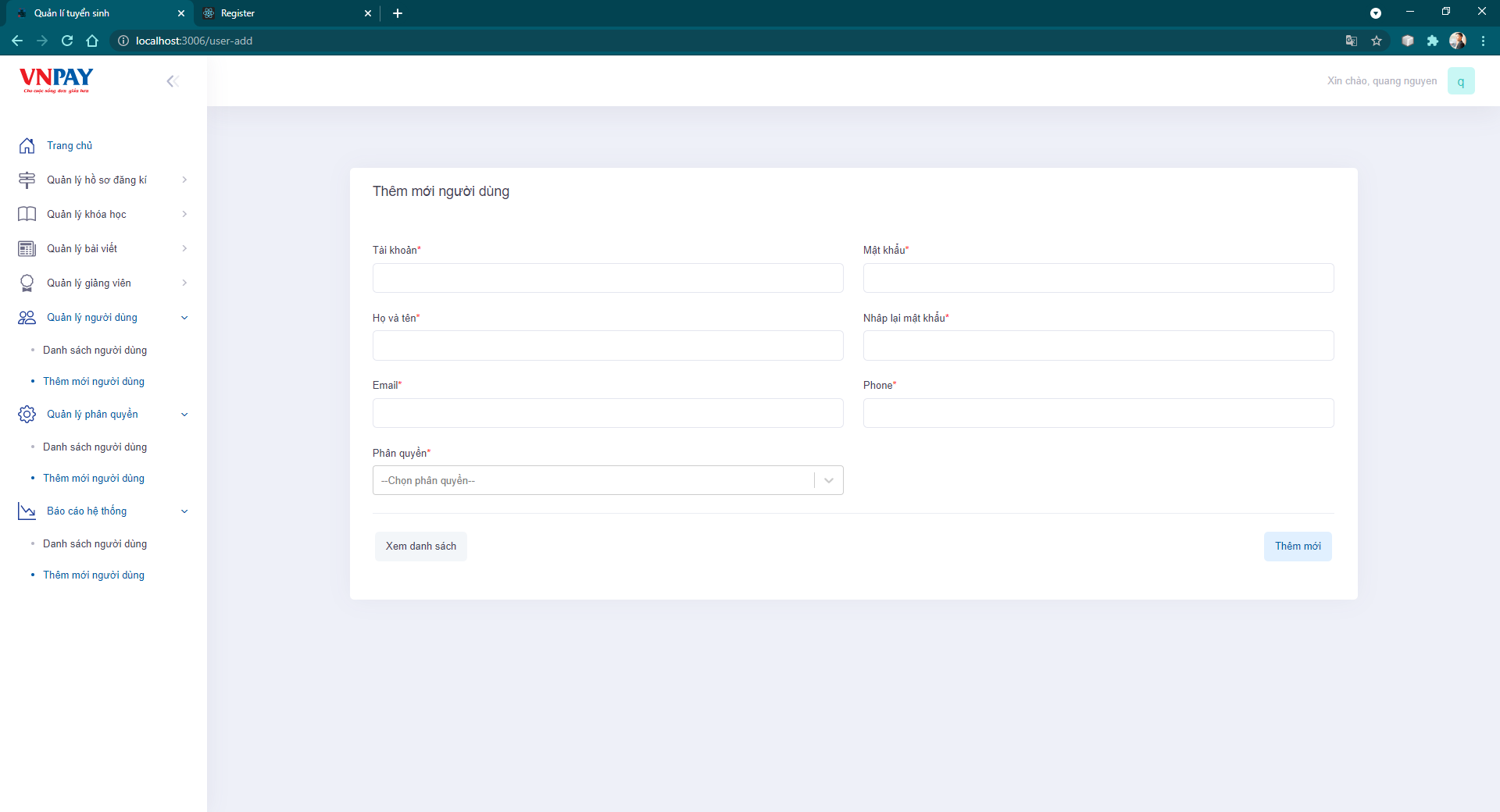
1. **Giao diện cập nhập thông tin giảng viên**

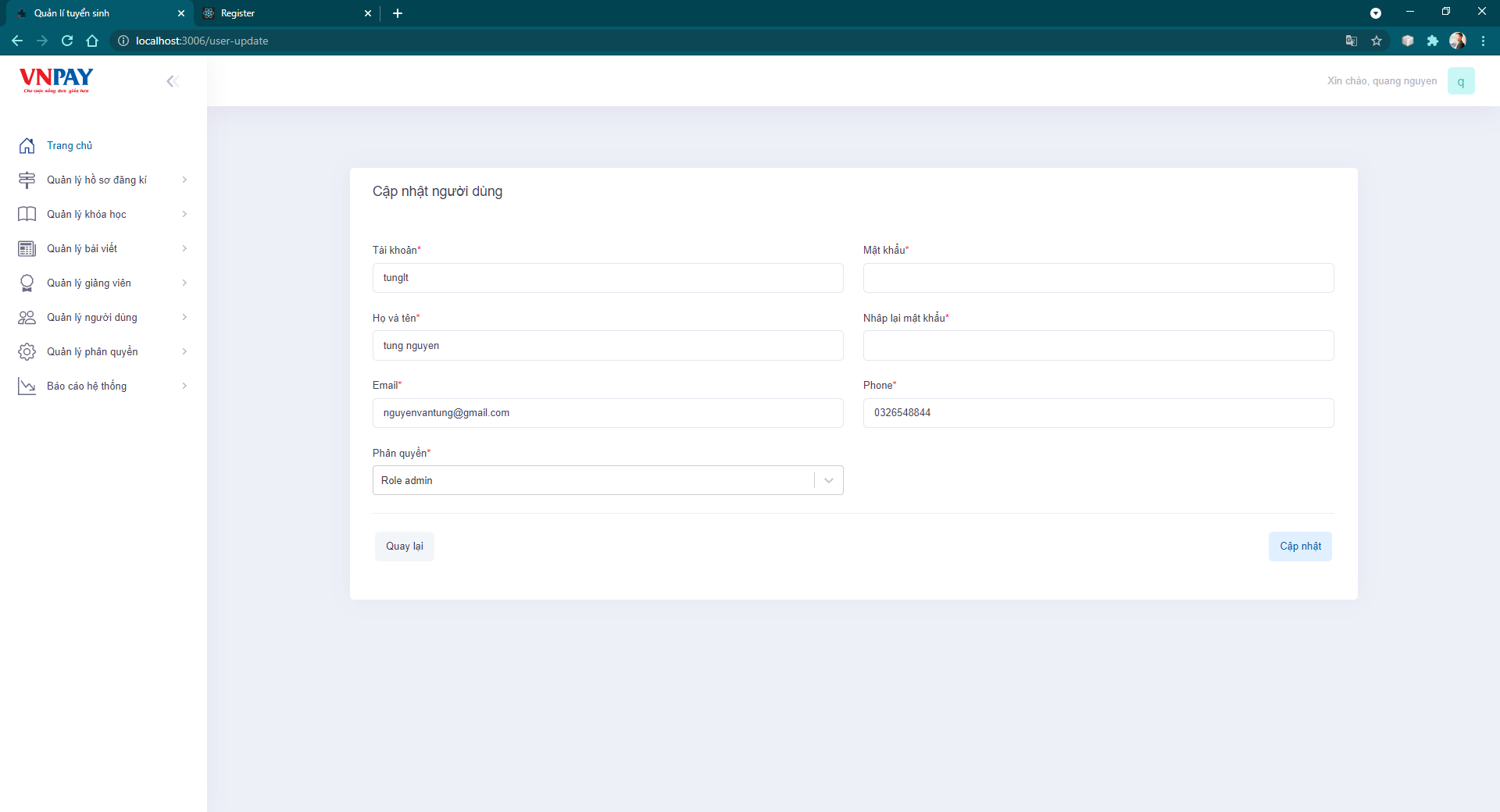
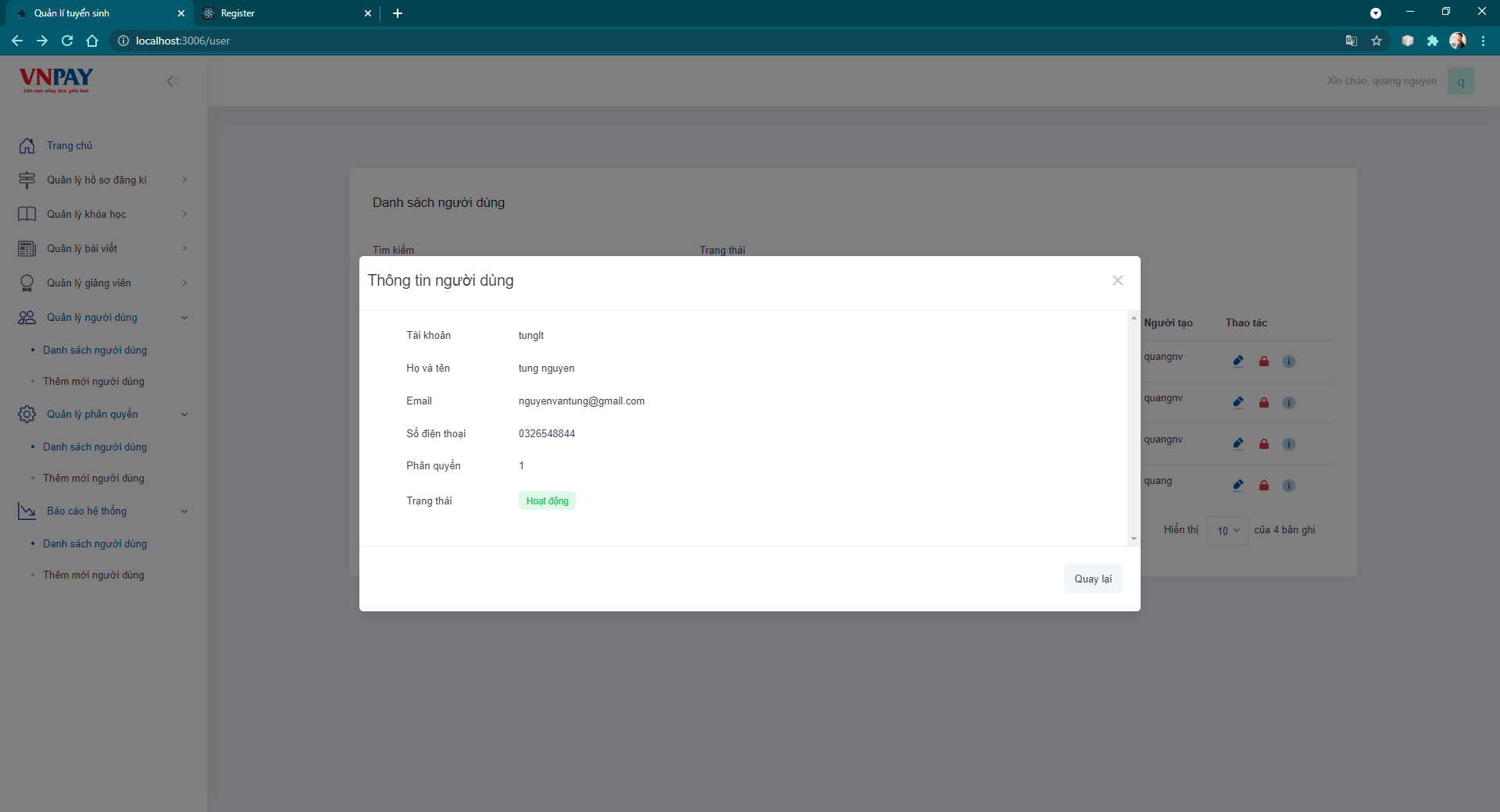


1. **Giao diện thông tin chi tiết giảng viên**
2. **Giao diện quản lí người dùng**



1. **Giao diện thêm mới người dùng**



1. **Giao diện trang cập nhập thông tin người dùng**
2. **Giao diện trang thông tin chi tiết người dùng**

# KẾT LUẬN

Sau một thời gian nghiên cứu và xây dựng đề tài “ Xây dựng website tuyển sinh cho trung tâm hợp tác quốc tế trường Đại Học Công Nghiệp Hà Nội, ứng dụng Java Springboot, ReactJS”, em đã phân tích, thiết kế và hoàn thành về cơ bản theo đúng những yêu cầu về nội dung và thời gian đã định.

Trong quá trình nghiên cứu và thực hiện đề tài vẫn gặp nhiều hạn chế về kinh nghiệm và kiến thức nên đề tài của em không thể tránh khỏi những thiếu xót. Em rất mong nhận được ý kiến đóng góp từ quý thầy cô và các bạn để đề tài được hoàn thiện hơn.

**Kết quả đạt được**

* Về mặt công nghệ, em đã tìm hiểu và nắm bắt quy trình để xây dựng một website phục vụ công tác tuyển sinh, tìm hiểu cách sử dụng intelij idea, Visual Code.
* Hoàn thiện khảo sát, phân tích, thiết kế website “Xây dựng website phục vụ công tác tuyển sinh chương trình tiếng việt và văn hóa Việt Nam tại trung tâm hợp tác quốc tế trường Đại Học Công Nghiệp Hà Nội ”
* Lập trình thành thạo với Framework Spring.

**Hạn chế của đề tài**

* Do thời gian làm có hạn lên em chưa thể hoàn thiện đầy đủ chức năng của chương trình, cũng như xử lý tất cả các ngoại lệ xảy ra trong quá trình xử dụng ứng dụng.
* Chưa kiểm soát được tất cả các lỗi có thể xảy ra trong quá trình sử dụng.

**Hướng phát triển**

* Hoàn thiện giao diện và các chức năng cho sản phẩm (chưa hoàn thành).
* Cải thiện tốc độ truy xuất dữ liệu khi có nhiều người cùng truy cập.
* Em rất mong tiếp tục nhận được sự giúp đỡ và tạo điều kiện của quý thầy cô và nhà trường để em có cơ hội phát triển, hoàn thiện ứng dụng tốt hơn trong thời gian tới.

# TÀI LIỆU THAM KHẢO

<https://freetuts.net/java-la-gi-gioi-thieu-ngon-ngu-java-1023.html>

<https://vi.wikipedia.org/wiki/Spring_Framework>

<http://laptrinhjavaweb.com/trang-chu>