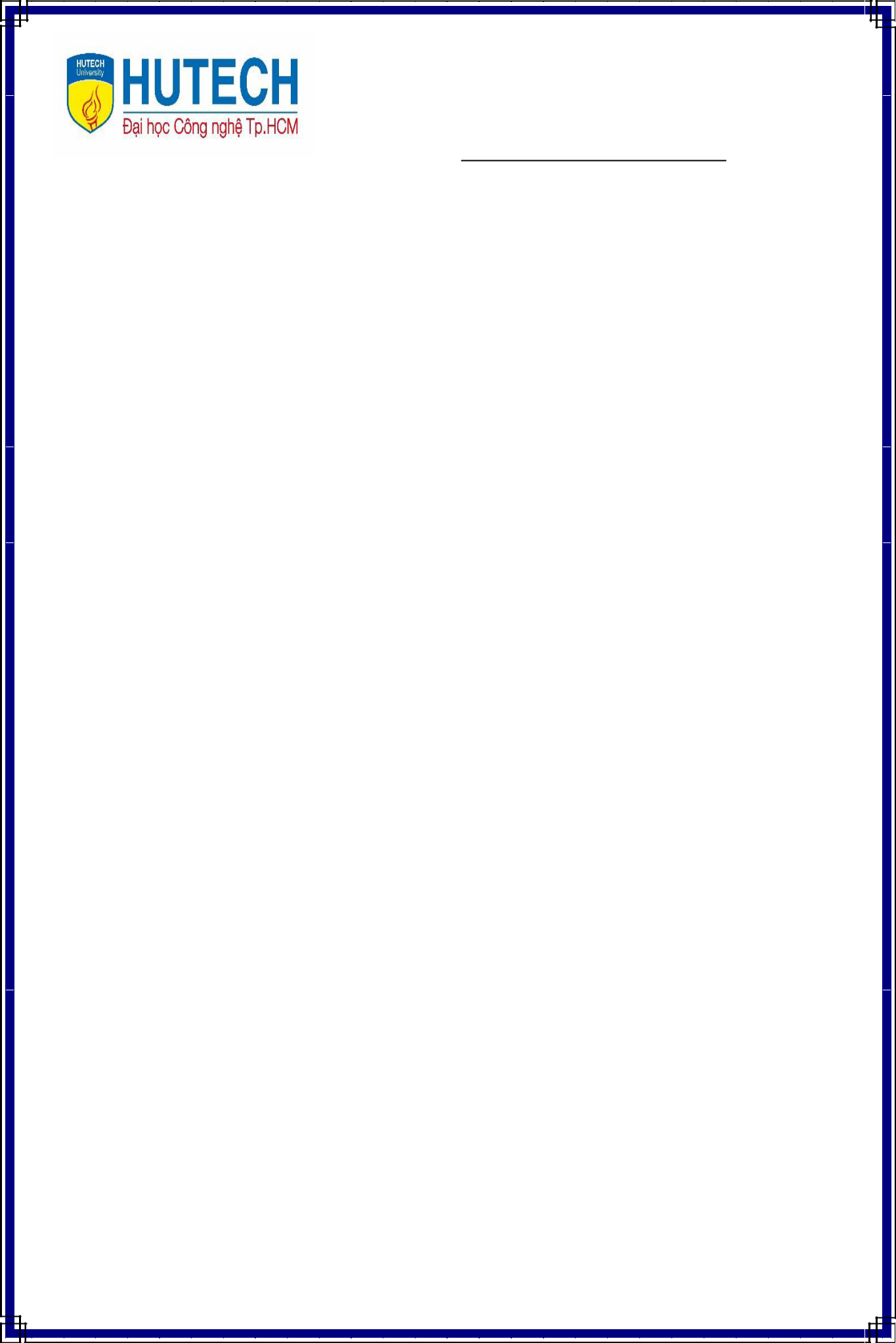
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP.HCM



**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN HỌC PHẦN**

**CÔNG NGHỆ PHẦN MỀM**

**PHẦN MỀM QUẢN LÝ THƯ VIỆN**

Chuyên ngành: **CÔNG NGHỆ THÔNG TIN**

Giảng viên hướng dẫn : **ThS. Dương Thành Phết**

Sinh viên thực hiện:

|  |  |  |
| --- | --- | --- |
| **MSSV:** | **Họ và tên:** | **Lớp:** |
| **2080600938** | **Nguyễn Hữu Bằng** | **20DTHE1** |
| **1811061022** | **Lê Quốc Thái** | **19DTHD4** |

*TP. Hồ Chí Minh, tháng 08/ 2022*

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP.HCM



**KHOA CÔNG NGHỆ THÔNG TIN**

**PHIẾU PHÂN CÔNG NHIỆM VỤ**

**THỰC HIỆN ĐỒ ÁN HỌC PHẦN: CÔNG NGHỆ PHẦN MỀM**

1. **Tên đề tài:** Phần mềm quản lý thư viện.
2. **Giảng viên hướng dẫn:** ThS. Dương Thành Phết**.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **MSSV/Họ tên** | **Nội dung công việc thực hiện** | | **Tự đánh giá** |
| **Tỉ lệ và điễm** |
|  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | |  |
| **1** | 2080600938 |  |  |  |
| Nguyễn Hữu Bằng |  |
|  |  |
|  |  | |
|  |  |  |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  | |  |
|  |  |  |  |  |
|  |  |  | |  |
|  | 1811061022 |  | |  |
| **2** |  |  |
| Lê Quốc Thái |  |  |  |
|  |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  | |  |
|  |  |  |  |  |

**LỜI CẢM ƠN**

Dù cho học kỳ hè ngắn ngủi 4 tuần tuy nhiên qua những bài giảng và những tài liệu được chia sẽ cực kỳ hay và hữu ích từ thầy. Nhờ những lời góp ý của thầy cùng với những kiến thức thực chiến được thầy chia sẽ qua đó có thể hiểu được những mong muốn mà những nhà tuyển dụng và những định hướng tương lai cần thực hiện.

Mặt dù trong quá trình thực hiện đồ án này, có nhiều khó khăn trong quá trình làm việc nhóm và về mặt số lượng thành viên so với các nhóm khác, tuy nhiên với sự động viên và giúp đở của thầy thì với 4 tuần gấp rút cuối cùng chúng em đã có thể hoàn thành đồ án hoàn chỉnh để kết thúc học phần này.

Nhóm em xin gửi lời cảm ơn chân thành và biết ơn sâu sắc đến thầy người đã tận tình chỉ dẫn và đưa ra những lời khuyên giúp em giải quyết được các vấn đề gặp phải trong quá trình nghiên cứu và hoàn thành đề tài một cách tốt nhất. Người đã dạy dỗ và truyền đạt những kiến thức quý báu cho em trong suốt học kỳ vừa qua. Trong thời gian tham dự lớp học, thầy đã giúp em tích luỹ thêm nhiều kiến thức hay và bổ ích, rất cần thiết cho quá trình học tập mà em chưa biết đến. Kính chúc thầy luôn dồi dào sức khỏe để tiếp tục dìu dắt và nâng đở bọn em. Nhóm em xin chân thành cảm ơn!

**Kính gửi**

**ThS. Dương Thành Phết**

**MỤC LỤC**

[**CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI**](#page11)[**1**](#page11)

[1.1. ĐẶT VẤN ĐỀ](#page11) [1](#page11)

[1.1.1. Phân tích yêu cầu](#page11) [1](#page11)

[1.2. MỤC TIÊU XÂY DỤNG ĐỀ TÀI](#page11) [1](#page11)

[1.2.1. Mục tiêu](#page12) [2](#page12)

[1.2.2. Yêu cầu chức năng](#page12) [2](#page12)

[1.2.3. Yêu cầu phi chức năng](#page14) [4](#page14)

[**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**](#page15)[**5**](#page15)

[2.1. GIỚI THIỆU REACT JS](#page15) [5](#page15)

[2.1.1. React là gì?](#page15) [5](#page15)

[2.1.2. Giới thiệu Virtual DOM](#page15) [5](#page15)

[2.1.3. Giới thiệu về JSX](#page16) [6](#page16)

[2.1.4. Giới thiệu về Components](#page16) [6](#page16)

[2.1.5. Props và State:](#page16) [6](#page16)

[2.2. GIỚI THIỆU NODE JS](#page17) [7](#page17)

[2.2.1. Node.js hổ trợ ES6](#page17) [7](#page17)

[2.2.2. Node.js giúp chúng ta chạy code JavaScript trên Server](#page17) [7](#page17)

[2.2.3. Node.js phù hợp với những loại app nào?](#page18) [8](#page18)

[2.2.4. Thế mạnh của Node.js?](#page18) [8](#page18)

[2.3. GIỚI THIỆU MONGODB](#page18) [8](#page18)

[2.3.1. 1.MongoDB là gì?](#page18) [8](#page18)

[2.3.2. Khái niệm NoSQL là gì?](#page19) [9](#page19)

[2.3.3. Tìm hiểu hệ cơ sở dữ liệu MongoDB có tính năng gì?](#page19) [9](#page19)

[2.3.4. Ưu điểm của MongoDB](#page19) [9](#page19)

[2.3.5. Nhược điểm của MongoDB](#page20) [10](#page20)

[2.4. GIỚI THIỆU VISUAL STUDIO CODE](#page20) [10](#page20)

[2.4.1. Visual Studio Code là gì?](#page21) [11](#page21)

[2.4.2. Một số tính năng của Visual Studio code](#page21) [11](#page21)

[2.4.3. Tại sao nên sử dụng Visual Studio Code?](#page21) [11](#page21)

[2.4.4. Ai là đối tượng nên sử dụng Visual Studio Code?](#page22) [12](#page22)

[2.5. GIỚI THIỆU GITHUB](#page22) [12](#page22)

[2.5.1. Github là gì?](#page22) [12](#page22)

[2.5.2. Tính năng của Github](#page23) [13](#page23)

[2.5.3. Một vài khái niệm của Git bạn cần nắm](#page23) [13](#page23)

[2.5.4. Lợi ích của Github đối với lập trình viên](#page24) [14](#page24)

[2.6. PHÂN TÍCH THIẾT KẾ](#page24) [14](#page24)

[2.6.1. Sơ đồ DFD](#page24) [14](#page24)

[2.6.2. Sơ đồ BFD](#page25) [15](#page25)

[2.6.3. Sơ đồ ngữ cảnh (Context Diagram)](#page25) [15](#page25)

[2.6.4. Sơ đồ ERD](#page26) [16](#page26)

[2.6.5. Sơ đồ Class Diagram](#page26) [16](#page26)

[2.6.6. Sơ đồ Use Case](#page27) [17](#page27)

[2.7. CƠ SỞ DỮ LIỆU HOÀN THIỆN](#page31) [21](#page31)

[**CHƯƠNG 3. HIỆN THỰC SẢN PHẨM**](#page33)[**23**](#page33)

[3.1. GIAO DIỆN HỆ THỐNG](#page33) [23](#page33)

[3.1.1. Giao diện trang chủ](#page33) [23](#page33)

[3.1.2. Giao diện giới thiệu](#page33) [23](#page33)

[3.1.3. Giao diện liên hệ](#page34) [24](#page34)

[3.1.4. Giao diện chính](#page34) [24](#page34)

[3.1.5. Giao diện đăng nhập](#page35) [25](#page35)

[3.1.6. Giao diện đăng ký tài khoản](#page36) [26](#page36)

[3.1.7. Giao diện quên mật khẩu](#page36) [26](#page36)

[3.1.8. Giao diện tin tức](#page37) [27](#page37)

[3.1.9. Giao diện hở trợ](#page37) [27](#page37)

[3.2. CÁC CHỨC NĂNG](#page38) [28](#page38)

[3.2.1. Đăng ký – kích hoạt (xác nhận email) và đăng nhập.](#page38) [28](#page38)

[3.2.2. Quên mật khẩu](#page38) [28](#page38)

[3.2.3. Quản lý lĩnh vực](#page39) [29](#page39)

[3.2.4. Quản lý tài liệu](#page39) [29](#page39)

[3.2.5. Quản lý yêu cầu xác nhận thành viên](#page41) [31](#page41)

[3.2.6. Quản lý bài giảng.](#page41) [31](#page41)

[3.2.7. Quản lý tài liệu yêu thích](#page42) [32](#page42)

[3.2.8. Quản lý tài khoản người dùng.](#page42) [32](#page42)

[3.2.9. Quản lý thông tin tài khoản cá nhân.](#page43) [33](#page43)

[3.2.10. Nâng cấp tài khoản](#page43) [33](#page43)

[3.2.11. Bình luận](#page44) [34](#page44)

[**CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**](#page45)[**35**](#page45)

[4.1. KẾT QUẢ ĐẠT ĐƯỢC](#page45) [35](#page45)

[4.2. ĐÁNH GIÁ WEBSITE](#page46) [36](#page46)

[4.2.1. Ưu điểm](#page46) [36](#page46)

[4.2.2. Nhược điểm](#page46) [36](#page46)

[4.3. HƯỚNG PHÁT TRIỂN VÀ MỞ RỘNG](#page46) [36](#page46)

[**Tài Liệu Tham Khảo**](#page48)[**38**](#page48)

**MỤC LỤC HÌNH ẢNH**

[Hình 2.1 Sơ đồ DFD](#page24) [14](#page24)

[Hình 2.2 Sơ đồ BFD](#page25) [15](#page25)

[Hình 2.3 Sơ đồ ngữ cảnh](#page25) [15](#page25)

[Hình 2.4 Sơ đồ ERD](#page26) [16](#page26)

[Hình 2.5 Sơ đồ Class Diagram](#page26) [16](#page26)

[Hình 2.6 Sơ đồ Use Case](#page27) [17](#page27)

[Hình 2.7 Phân rã chức năng Quản Lý Tài Khoản](#page27) [17](#page27)

[Hình 2.8 Phân rã chức năng Quản Lý Bài Giảng](#page28) [18](#page28)

[Hình 2.9 1.1.1.1. Phân rã chức năng Quản Lý Tin Tức](#page28) [18](#page28)

[Hình 2.10 Phân rã chức năng Quản Lý Tài Liệu](#page29) [19](#page29)

[Hình 2.11 1.1.1.1. Phân rã chức năng Quản Lý Lĩnh Vực](#page29) [19](#page29)

[Hình 2.12 1.1.1.1. Phân rã chức năng Quản Lý Yêu Cầu Nâng Cấp](#page30) [20](#page30)

[Hình 2.13 1.1.1.1. Phân rã chức năng Quản Lý Bình Luận](#page30) [20](#page30)

[Hình 3.1 Giao diện trang chủ.](#page33) [23](#page33)

[Hình 3.2 Giao diện giới thiệu trang web.](#page33) [23](#page33)

[Hình 3.3 Giao diện liên hệ nhà phát triển](#page34) [24](#page34)

[Hình 3.4 Giao diện chính sau khi đăng nhập (Super-admin / Admin).](#page34) [24](#page34)

[Hình 3.5 Giao diện chính sau khi đăng nhập (Thành viên / Khách).](#page35) [25](#page35)

[Hình 3.6 Giao diện đăng nhập.](#page35) [25](#page35)

[Hình 3.7 Giao diện đăng ký tài khoản.](#page36) [26](#page36)

[Hình 3.8 Giao diện quên mật khẩu.(reset password).](#page36) [26](#page36)

[Hình 3.9 Giao diện tin tức.](#page37) [27](#page37)

[Hình 3.10 Giao diện hổ trợ.](#page37) [27](#page37)

[Hình 3.11 Chức năng đăng ký tài khoản.](#page38) [28](#page38)

[Hình 3.12 Chức năng quên mật khẩu.](#page39) [29](#page39)

[Hình 3.13 Chức năng thêm – xoá – sửa lĩnh vực.](#page39) [29](#page39)

[Hình 3.14 Chức năng hiển thị tài liệu theo dạng lưới.](#page40) [30](#page40)

[Hình 3.15 Chức năng hiển thị tài liệu theo dạng danh sách và thêm – xoá – sửa. . 30](#page40)

[Hình 3.16 Chức năng xác nhận thành viên.](#page41) [31](#page41)

[Hình 3.17 Chức năng thêm – xoá – sửa bài giảng.](#page41) [31](#page41)

[Hình 3.18 Chức năng tìm kiếm và tải danh sách tài liệu yêu thích.](#page42) [32](#page42)

[Hình 3.19 Chức năng tìm kiếm, thêm – xoá – sửa tài khoản người dùng.](#page42) [32](#page42)

[Hình 3.20 Chức năng xem và chỉnh sửa thông tin tài khoản cá nhân.](#page43) [33](#page43)

[Hình 3.21 Chức năng yêu cầu nâng cấp tài khoản.](#page43) [33](#page43)

[Hình 3.22 Chức năng bình luận](#page44) [34](#page44)

**CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI**

**1.1. ĐẶT VẤN ĐỀ**

* + 1. **Phân tích yêu cầu**

Ngày nay, khi xu thế toàn cầu hoá, hội nhập quốc tế đang diễn ra mạnh mẽ ở hầu hết các quốc gia trên thế giới. Khi sống trong thời đại mà nền khoa học công nghệ hiện đại nói chung đang phát triển vượt bậc, thì khối ngành công nghệ thông tin nói riêng cũng đang có những thành tựu vĩ đại trong hầu hết các lĩnh vực. Tin học đã giúp cho các nhà quản lý điều hành công việc một cách khoa học, nhẹ nhàng hơn nhưng lại mang đến hiệu quả chính xác cao hơn khi chưa đưa vào máy tính.

Quản lý thư viện trường đại học cũng là một trong những ví dụ điển hình cho vấn đề quản lý. Nếu không được công nghệ hoá lẫn tin học hoá thì việc quản lý sẽ vất vả hơn rất nhiều lần với khối lượng tài liệu khổng lồ và đa dạng các thể loại, cùng với khối lượng công việc khổng lồ khi phải phục vụ cho hàng ngàn sinh viên. Hệ thống quản lý trước đây đa phần đều là kiểm kê thủ công, thông qua hàng loạt sổ sách rời rạc, phức tạp thế nên người quản lý sẽ gặp nhiều khó khăn trong việc cập nhật kịp thời các thông tin tài liệu, kiểm kê cũng như bị mất hoặc thất thoát trong việc mượn, trả tài liệu và vận chuyển. Do đó các thông tin quản lý phục vụ kinh doanh không tránh khỏi sự dư thừa hoặc không đầy đủ, thêm vào đó phương pháp quản lý thủ công lại rất tốn kém về mặt thời gian lẫn công sức và đòi hỏi phải có nguồn nhân lực cao.

Thư viện với chức năng là cơ quan văn hoá, giáo dục trong và ngoài nhà trường có vai trò quan trọng trong việc nâng cao dân trí, đào tạo nhân lực, bồi dưỡng nhân tài và là nơi để mỗi con người có thể học hỏi, khám phá những điều mới mẻ. Xuất phát từ nhu cầu thực tế là cần có một hệ thống trợ giúp quản lý việc xuất - nhập và các thao tác trong các khâu quản lý của thư viện giúp cho sinh viên có cơ hội tiếp xúc với các thông tin một cách nhanh chống thông qua sự phát phát triển của khoa học - kỹ thuật, đặc biệt với sự xuất hiện của điện thoại di động, laptop, máy tính bảng... Vì thế chúng em đã quyết định thực hiện đề tài “Quản lý thư viện sách”, ngoài việc giảm bớt áp lực thời gian công sức cho người quản lý mà còn đảm bảo được yêu cầu “nhanh chóng - hiệu quả - chính xác - tiện lợi”. Với phần mềm quản lý thư viện bằng C# là một phần mềm được thiết kế để quản lý sách với đầu đủ chức năng về số lượng đầu sách và chức năng quản lý giữa người đọc và người mượn, trả sách để có cái nhìn tổng quan, cụ thể nhất về thư viện.

**1.2. MỤC TIÊU XÂY DỤNG ĐỀ TÀI**

* *Đối tượng ứng dụng đề tài:*

Các học sinh, sinh viên và thủ thư của các trường học cấp 1, 2, 3, các trường đại học, các thư viện còn phải dùng hồ sơ và tài liệu giấy cho việc quản lý

* *Mục tiêu nghiên cứu:*
* Tìm hiểu và học thêm được về git và cách sử dụng github
* Học và tìm hiểu về ngôn ngữ lập trình C#
* Tìm hiểu về các công nghệ hỗ trợ đề tài như: SQL sever, Devexpress, Visual Studio, StarUML, Case studio 2
* Tìm hiểu về winform, cách tổ chức code và những toolbox
* Hiểu hơn về quy trình kết nối giữa ứng dụng và database
* Nâng cao kiến thức về SQL sever
* Vận dụng cách kiến thức trên xây dựng phần mềm thân thiện và chất lượng

**1.2.1. Mục tiêu đề tài**

* **Đối với sinh viên**: Sinh viên là người có nhu cầu tìm kiếm tài liệu học tập để bổ sung kiến thức. Thông qua phần mềm, sinh viên sẽ dể dàng tìm kiếm và tra cứu tài liệu trên ứng dụng mà không cần đến thư viện, giúp dể dàng tìm kiếm và đăng ký mượn sách mong muốn mà không cần nhiều thủ tục rờm rà.
* **Đối với thư viện**: Nhân viên quản lý và thủ thư có thể dể dàng quản lý và chỉnh sửa các thông tin về sách, độc giả dể dàng. Giúp tiết kiệm thời gian và dể dàng tra cứu thông tin

Về phía phần mềm còn phải có tính dễ nâng cấp, bảo trì, sửa chữa khi cần bổ sung, cập nhật những tính năng khi thư viện có thêm những quy định mới. Cùng với sự bảo mật thông tin của độc giả trong cơ sở dữ liệu

**1.2.2. Yêu cầu chức năng**

*1.2.2.1. Chức năng nghiệp vụ*

* **Lưu trữ:** sách, thể loại, nhà xuất bản, phiếu mượn sách của độc giả, thông tin độc giả và thông tin nhân viên cùng với tài khoản của nhân viên
* **Tra cứu:** tìm kiếm sách, tìm kiếm nhân viêm, tìm kiếm độc giả dể dàng theo nhiều trường dữ liệu

**1.2.3. Yêu cầu phi chức năng**

* **Tính tiến hoá:**
  + Cho phép thay đổi về các quy định, nội dung của của thư viện trong khi sử dụng hệ thống.
* **Tính tiện dụng:**
  + Có thể tra cứu được thông tin của sách, nhân viên, độc giả nhanh chóng, dể dàng
  + Thống kê được tình hình sách trong thư viện, sách được mượn và tất cả sách chỉ bằng cú click chuột
* **Tính hiệu quả**
  + - Giao diện thân thiện, dể dàng sử dụng.

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

**2.1. Giới thiệu về C# và .NET Framework**



**2.1.1. C# là gì?**

[C#](https://quantrimang.com/hoc/csharp) là một ngôn ngữ lập trình đơn giản, hiện đại, mục đích tổng quát, hướng đối tượng được phát triển bởi Microsoft và được phê chuẩn bởi European Computer Manufacturers Association (ECMA) và International Standards Organization (ISO) và được phát triển bởi Anders Hejlsberg và team của ông trong khi phát triển .[Net Framework](https://quantrimang.com/cong-nghe/microsoft-net-framework-la-gi-va-tai-sao-no-duoc-cai-dat-tren-pc-154550).

C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java, với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), . . . trở nên rất dễ dàng với những lập trình viên.

**2.1.2. Các đặc trưng của C#**

### C# là ngôn ngữ đơn giản

### Như bạn đã biết, ngôn ngữ C# rất đơn giản vì nó dựa trên C++ và Java. Nếu bạn đã quen thuộc với C và C++, hoặc thậm chí là Java, bạn sẽ thấy rằng C# rất giống nhau về hình thức, cú pháp, biểu thức, toán tử và các tính năng khác được lấy trực tiếp từ ngôn ngữ C và C +, nhưng ngôn ngữ này đã được cải thiện làm cho nó dễ dàng hơn. Một số cải tiến là loại bỏ tính dài dòng hoặc bổ sung cú pháp đã thay đổi.

### C# là ngôn ngữ hiện đại

Một vài khái niệm khá mới mẻ khá mơ hồ với các bạn vừa mới học lập trình, như xử lý ngoại lệ, những kiểu dữ liệu mở rộng, bảo mật mã nguồn..v..v… Đây là những đặc tính được cho là của một ngôn ngữ hiện đại cần có. Và C# chứa tất cả các đặt tính ta vừa nêu trên.

### C# là một ngôn ngữ lập trình thuần hướng đối tượng

***Lập trình hướng đối tượng****(****Object-Oriented Programming – OOP****)* là một kỹ thuật lập trình hỗ trợ hướng đối tượng, trở thành một chuẩn lập trình của các ngôn ngữ lập trình tiên tiến hiện nay. Điểm mạnh của lập trình hướng đối tượng là tái sử dụng lại code, cung cấp một cấu trúc các module của chương trình một cách rõ ràng, che dấu được dữ liệu bên trong, giúp lập trình viên đơn giản hóa độ phức tạp của bài toán.

Ngôn ngữ C# là ngôn ngữ hướng đối tượng tiên tiến nhất hiện nay, nó kế thừa những điểm mạnh của ngôn ngữ lập trình hướng đối tượng C++ và Java, đồng thời nó cũng loại bỏ đi những sự phức tạp của lập trình hướng đối tượng của hai ngôn ngữ này, ví dụ như nó loại bỏ tính đa thừa kế trong C++, hoặc nó cho phép viết chồng các toán tử mà Java không có

### C# là một ngôn ngữ ít từ khóa

C# là ngôn ngữ sử dụng giới hạn những từ khóa. Phần lớn các từ khóa được sử dụng để mô tả thông tin. Chúng ta có thể nghĩ rằng một ngôn ngữ có nhiều từ khóa thì sẽ mạnh hơn. Điều này không phải sự thật, ít nhất là trong trường hợp ngôn ngữ C#, chúng ta có thể tìm thấy rằng ngôn ngữ này có thể được sử dụng để làm bất cứ nhiệm vụ nào.

*Dưới đây là một số ưu điểm nổi bật của C# so với ngôn ngữ lập trình khác:*

* C# có cấu trúc khá gần gũi với các ngôn ngữ lập trình truyền thống, nên cũng khá dể dàng tiếp cận và học nhanh với C#.
* C# có thể biên dịch trên nhiều nền tảng máy tính khác nhau.
* C# được xây dựng trên nền tảng của C++ và Java nên nó được thừa hưởng những ưu điểm của ngôn ngữ đó.
* C# là một phần của .NET Framework nên được sự chống lưng khá lớn đến từ bộ phận này.
* C# có IDE Visual Studio cùng nhiều plug-in vô cùng mạnh mẽ.

## 2.1.3. Sơ lượt về .NET Framework

* NET Framework được Microsoft đưa ra chính thức từ năm 2002. .NET Framework chỉ hoạt động trên Windows. Những nền tảng ứng dụng như WPF, Winforms, ASP.NET(1-4) hoạt động dựa trên .NET Framework.
* Cho đến năm 2013, Microsoft định hướng đi đa nền tảng và phát triển .NET core. .NET core hiện được sử dụng trong các ứng dụng Universal Windows platform và ASP.NET Core. Từ đây, C# có thể được sử dụng để phát triển các loại ứng dụng đa nền tảng trên các hệ điều hành khác nhau (Windows, Linux, MacOS,…)
* Framework được triển khai và thực thi trong phần mềm với tên CLR (Common Language Runtime). Phần mềm thực thi này là một máy ảo cung cấp gồm an ninh phần mềm (security), quản lý bộ nhớ (memory management), và các xử lý lỗi ngoại lệ (exception handling)
* .NET Framework gồm các tập thư viện lập trình lớn và những thư viện này hỗ trợ việc xây dựng các chương trình phần mềm. CLR và bộ thư viện là 2 thành phần chính của .NET Framework

**2.2. Giới thiệu về Winform**



## **2.2.1 Winform là gì?**

Winform là thuật ngữ mô tả một ứng dụng được viết dùng .NET FrameWorrk và có giao diện người dùng Windows Forms. Mỗi màn hình windows cung cấp một giao diện giúp người dùng giao tiếp với ứng dụng. Giao diện này được gọi là giao diện đồ họa (GUI) của ứng dụng.

Là các ứng dụng windows chạy trên máy tính – mã lệnh thực thi ngay trên máy tính: Microsoft, Word, Excel, Access, Calculator, yahoo, Mail…  là các ứng dụng được tạo ra từ **Windows Forms**.

## **2.2.2 Ưu điểm của Winform**

Đa phần lập trình viên C#. NET nào cũng từng học/sử dụng Winform. Bởi vì: Giao diện kéo thả dễ sử dụng; Gắn các event cho các button chỉ cần double click, lại hỗ trợ quá trời event như click, hover,…; Việc viết code cũng vô cùng trực quan: từ việc lấy text từ TextBox cho tới show dữ liệu bằng MessageBox, hoặc dùng Grid để kết nối SQL. **WinForm rất dễ học và dễ dạy nhưng cũng không có nghĩa là có thể dể dàng hiểu sâu về nó.**

* Tốc độ xử lý dữ liệu nhanh chóng
* Đảm bảo an toàn, bảo mật thông tin
* Có thể chạy trên các phiên bản Windows khác nhau.
* Thao tác trên nhiều giao diện

## 2.2.3 Nhược điểm của Winform

* Phần mềm chạy trên nền tảng Windows. Điều này có nghĩa là bất kỳ ai sử dụng phần mềm đều phải sử dụng máy tính có cài đặt phần mềm. Vì vậy, bạn sẽ cần phải mang theo máy tính riêng để thực hiện công việc của mình.
* Đồ họa trên winform không cao nên giao diện phần mềm sẽ thiếu tính trực quan, hơi khó thao tác, không thân thiện với người dùng.

**2.3. Tổng quan về SQL Server**



**2.3.1. Giới thiệu SQL Server**

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) ) sử dụng câu lệnh SQL (Transact-SQL)để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server…

### 2.3.2. **Các thành phần cơ bản trong SQL Server**

Các thành cơ bản trong SQL Server gồm có: Reporting Services, Database Engine, Integration Services, Notification Services, Full Text Search Service,…

**+ *Database Engine***: Đây là một engine có khả năng chứa dữ liệu ở các quy mô dưới dạng support và table.

***+ Integration Services:*** là tập hợp các đối tượng lập trình và các công cụ đồ họa cho việc sao chép, di chuyển và chuyển đổi dữ liệu. Khi bạn muốn định dạng dữ liệu trước khi lưu vào database. Chắc chắn *Integration Services* sẽ giúp bạn giải quyết được công việc này dễ dàng.

**+ *Analysis Services***: Công cụ này giúp bạn trong việc phân tích dữ liệu một cách hiệu quả và dễ dàng bằng cách dùng kỹ thuật khai thác dữ liệu – datamining và khái niệm hình khối nhiều chiều – multi dimendion cubes.

***+ Notification Services***: Dịch vụ thông báo này là nền tảng cho sự phát triển và triển khai các ứng dụng soạn và gửi thông báo. Ngoài ra, dịch vụ này còn có chức năng gửi thông báo theo dịch thời đến hàng ngàn người dăng ký sử dụng trên nhiều loại thiết bị khác nhau.

***+ Reporting Services***: là một công cụ tạo, quản lý và triển khai báo cáo bao gồm: server và client. Ngoài ra, nó còn là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

***+ Full Text Search Service***: là một thành phần đặc biệt trong việc truy vấn và đánh chỉ mục dữ liệu văn bản không cấu trúc được lưu trữ trong các cơ sở dữ liệu SQL Server.

***+ Service Broker***: là một môi trường lập trình cho việc tạo ra các ứng dụng trong việc nhảy qua các Instance.

**2.4. GIỚI THIỆU VISUAL STUDIO 2022**



**2.4.1. Visual Studio 2022 là gì?**

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio được phát triển bởi Microsoft, là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Công cụ này được tạo lên và thuộc quyền sở hữu của ông lớn công nghệ Microsoft. Năm 1997, phần mềm lập trình nay có tên mã Project Boston. Nhưng sau đó, Microsoft đã kết hợp các công cụ phát triển, đóng gói thành sản phẩm duy nhất.

Visual Studio 2022 hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio 2022 cũng cho phép người dùng thay đổi theme và các tùy chọn khác.

**2.4.2. Một số tính năng của Visual Studio 2022**

* Đa nền tảng: Không giống như các trình viết code khác, Visual Studio sử dụng được trên cả Windows, Linux và Mac Systems.
* Đa ngôn ngữ lập trình: Visual Studio cũng cho phép sử dụng nhiều ngôn ngữ lập trình khác nhau từ C#, F#, C/C++, HTML, CSS, [Visual Basic](https://vi.wikipedia.org/wiki/Visual_Basic#:~:text=Visual%20Basic%20(vi%E1%BA%BFt%20t%E1%BA%AFt%20VB,h%E1%BB%A3p%20(IDE)%20k%E1%BA%BFt%20b%C3%B3.), JavaScript
* Kho lưu trữ an toàn: Với Visual Studio, bạn có thể hoàn toàn yên tâm về tính lưu trữ, bởi phần mềm đã được kết nối GIT và một số kho lưu trữ an toàn được sử dụng phổ biến hiện nay.
* Tính năng comment: Tính năng này cho phép lập trình viên để lại nhận xét, giúp dễ dàng ghi nhớ công việc cần hoàn thành, không bỏ sót công đoạn nào.
* Hỗ trợ website: Visual Studio code cũng hỗ trợ website, đặc biệt trong công việc soạn thảo và [thiết kế web](https://mona.solutions/thiet-ke-website).

**2.5. GIỚI THIỆU CÔNG CỤ LÀM VIỆC GITHUB**



**2.5.1. Github là gì?**

GitHub là một dịch vụ nổi tiếng cung cấp kho lưu trữ mã nguồn Git cho các dự án phần mềm. Github có đầy đủ những tính năng của Git, ngoài ra nó còn bổ sung những tính năng về social để các developer tương tác với nhau.

* Là công cụ giúp quản lý source code tổ chức theo dạng dữ liệu phân tán.
* Giúp đồng bộ source code của team lên 1 server.
* Hỗ trợ các thao tác kiểm tra source code trong quá trình làm việc

Github cung cấp các tính năng social networking như feeds, followers, và network graph để các developer học hỏi kinh nghiệm của nhau thông qua lịch sử commit.

**2.5.2. Tính năng của Github**

GitHub được coi là một mạng xã hội dành cho lập trình viên lớn nhất và dễ dùng nhất với các tính năng cốt lõi như:

* Wiki, issue, thống kê, đổi tên project, project được đặt vào namespace là user.
* Hợp tác cùng lập trình
* Quản lý tiến độ của dự án bằng watch project
* GitHub Al Autocomplete
* Follow user: theo dõi hoạt động của người khác.

**2.5.3. Lợi ích của Github đối với lập trình viên**

* Quản lý source code dễ dàng
* Tracking sự thay đổi qua các version
* Markdown
* Github giúp chứng tỏ bạn là ai
* Github giúp cải thiện kỹ năng code, thậm chí là tracking bug
* Github là một kho tài nguyên tuyệt vời
* Github Action
* Github Package Registry
* Mở rộng mối quan hệ

**2.6. MÔ TẢ YÊU CẦU**

Một thư viện cần xây dựng một cơ sở dữ liệu để quản lý sách và việc cho mượn sách. Hệ thống thư viện sẽ nhập sách từ các nhà cung cấp, nhà cung cấp sẽ bao gồm thông tin: mã nhà cung cấp, tên nhà cung cấp. Mỗi nhà cung cấp sẽ cung cấp nhiều sách thuộc nhiều thể loại khác nhau và một quyển sách chỉ thuộc 1 thể loại duy nhất. Thông tin về thể loại sách bao gồm mã loại và tên loại. Mỗi lần nhập sách từ nhà cung cấp cần ghi nhận lại thông tin nhập, phiếu nhập sách gồm các thông tin sau: mã phiếu nhập, các loại sách nhập vào, số lượng sách đã được nhập, đơn giá, nhập từ nhà cung cấp, Các sách được nhập về sẽ có thông tin bao gồm: mã sách, tên sách, mã loại, ngày xuất bản, thuộc nhà xuất bản nào, thuộc mã tác giả nào, có số lượng là bao nhiêu, mã kệ sách. Những thông tin về nhà xuất bản bao gồm: mã nhà xuất bản, tên nhà xuất bản, địa chỉ, số điện thoại.

Thư viện có rất nhiều kệ sách trưng bày các thể loại sách, tuy nhiên mỗi thể loại chỉ được trưng bày ở một kệ sách nhất định nào đó. Thông tin về kệ sách bao gồm: mã kệ và vị trí. Các độc giả sẽ đến thư viện để mượn sách cần lập thẻ thư viện hoặc xuất trình thẻ thư viện hiện có, mỗi độc giả chi có duy nhất 1 thẻ sử dụng. Khi muốn mượn sách, thủ thư sẽ tiến hành tra cứu sách mà độc giả muốn mượn trong hệ thống. Nếu hợp lệ, thủ thư sẽ tiến hành thiết lập phiếu mượn-trả cho độc giả bao gồm: mã phiếu mượn, ngày mượn sách, ngày trả sách và tình trạng sách. Mỗi độc giả sẽ được mượn nhiều sách theo số lượng quy định của thư viện. Khi trả sách, độc giả cần xuất trình phiếu mượn-trả, thủ thư sẽ tiến hành kiểm tra phiếu mượn- trả , trường hợp độc giả quên trả sách đúng hẹn hoặc làm hư hỏng đến sách thì sẽ phải đền bù và nhân viên sẽ phải lập phiếu phạt dành cho độc giả. Thông tin về phiếu phạt cần ghi rỏ: mã phiếu phạt,  mức phí phạt.

Mỗi Nhân viên sẽ được admin cung cấp tài khoản cá nhân và chỉ sử dụng được một số chức năng mà admin quy định. Đối với những độc giả mới cần xuất trình thẻ thư viện để được nhân viên tạo tài khoản để sử dụng.. Tất cả tài khoản và mật khẩu phải có độ dài từ 6-24 ký tự bao gồm chử và số, tên đăng nhập là duy nhất.

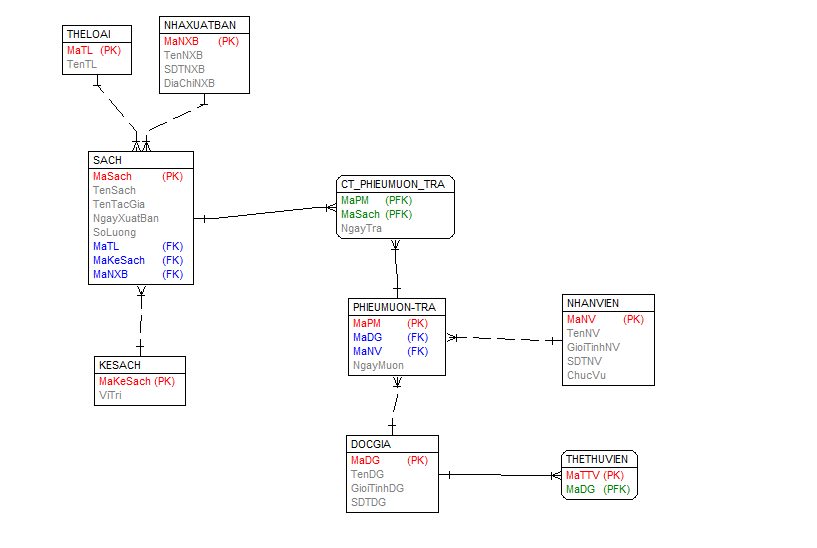
Admin sẽ có chức năng thống kê sách được mượn, số lượng sách có trong thư viện. Mỗi tháng, dựa theo báo cáo thống kê, thủ thư sẽ tiến hành nhập thêm số lượng sách từ nhà cung cấp sao cho phù hợp với nhu cầu của độc giả.

**2.7. DANH SÁCH USE CASE**

|  |  |  |
| --- | --- | --- |
| STT | Use Case | Diễn giải |
| 1 | Đăng nhập | Cho phép admin hoặc người dùng đăng nhập vào hệ thống |
| 2 | Đăng xuất | Trở về màn hình đăng nhập |
| 3 | Thêm thông tin sách | Cho phép thêm thông tin sách |
| 4 | Sửa thông tin sách | Cho phép chỉnh sửa thông tin sách |
| 5 | Xóa thông tin sách | Cho phép quản lý xóa thông tin sách |
| 3 | Thêm thông tin thể loại | Cho phép thêm thông tin thể loại sách |
| 4 | Sửa thông tin thể loại | Cho phép chỉnh sửa thông tin thể loại sách |
| 5 | Xóa thông tin thể loại | Cho phép quản lý xóa thông tin thể loại sách |
| 6 | Thêm độc giả | Cho phép thêm thông tin một độc giả |
| 7 | Sửa thông tin độc giả | Cho phép sửa thông tin độc giả |
| 8 | Xóa thông tin độc giả | Cho phép xóa thông tin một độc giả |
| 9 | Thêm nhân viên | Cho phép thêm một nhân viên |
| 10 | Sửa nhân viên | Cho phép sửa thông tin nhân viên |
| 11 | Xóa nhân viên | Cho phép xóa thông tin nhân viên |
| 12 | Tạo phiếu mượn-trả | Cho phép tạo mới 1 phiếu mượn-trả |
| 13 | Sửa phiếu mượn-trả | Cho phép sửa thông tin phiếu mượn trả |
| 14 | Xóa phiếu mượn-trả | Cho phép xóa phiếu mượn-trả |
| 19 | Tra cứu sách | Cho phép tìm kiếm thông tin, vị trí của sách theo nhiều trường dữ liệu(theo mã sách, tên sách, thể loại, tác giả) |
| 20 | Tìm kiếm độc giả | Cho phép tìm kiếm thông tinđộc giả theo nhiều trường dữ liệu(theo mã độc giả, tên độc giả, số điện thoại) |
| 21 | Báo cáo-Thống kê | Thống kê lại tất cả sách của thư viện, số lượng sách đã được mượn, số lượng sách còn lại trong thư viên |

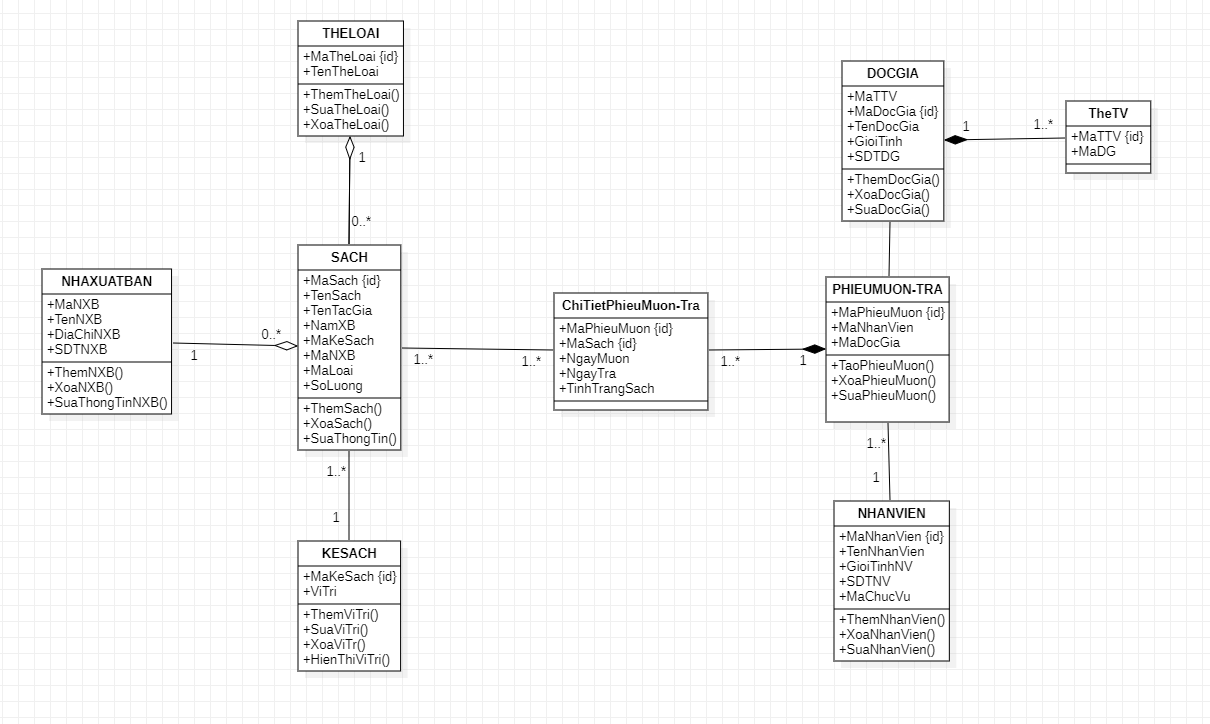
**2.8. PHÂN TÍCH THIẾT KẾ**

**2.8.1. Sơ đồ thực thể kết hợp ( ERD – Entity Relationship Diagram)**



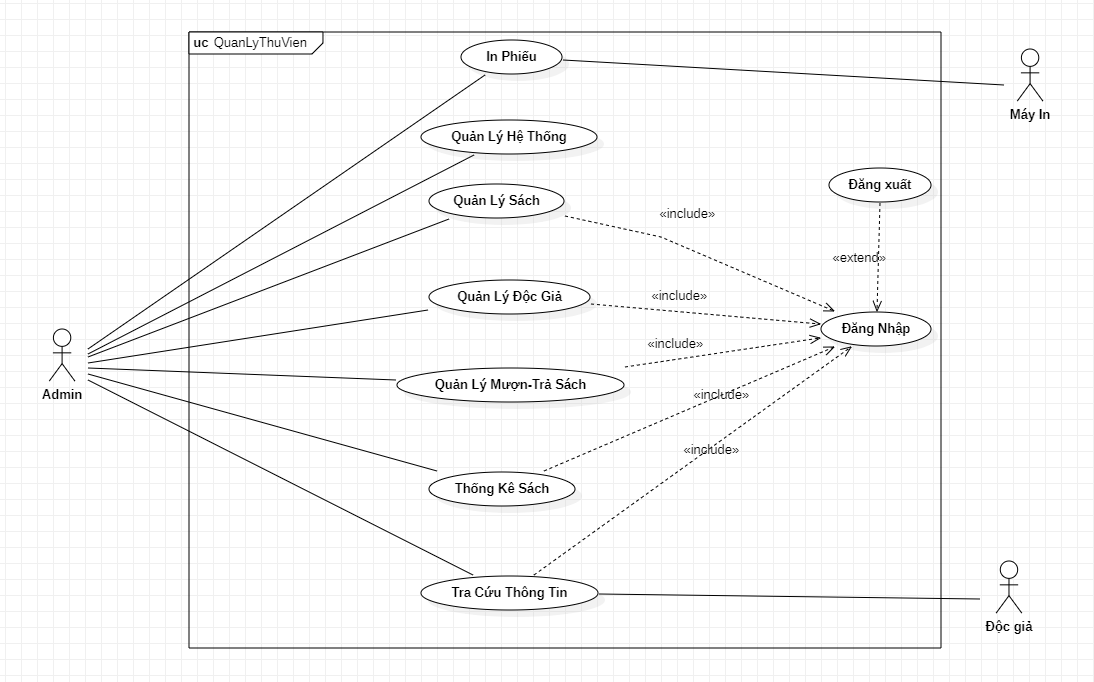
*Hình 2.1 Entity Relationship Diagram*

**2.8.2. Sơ đồ lớp ( Class Diagram )**

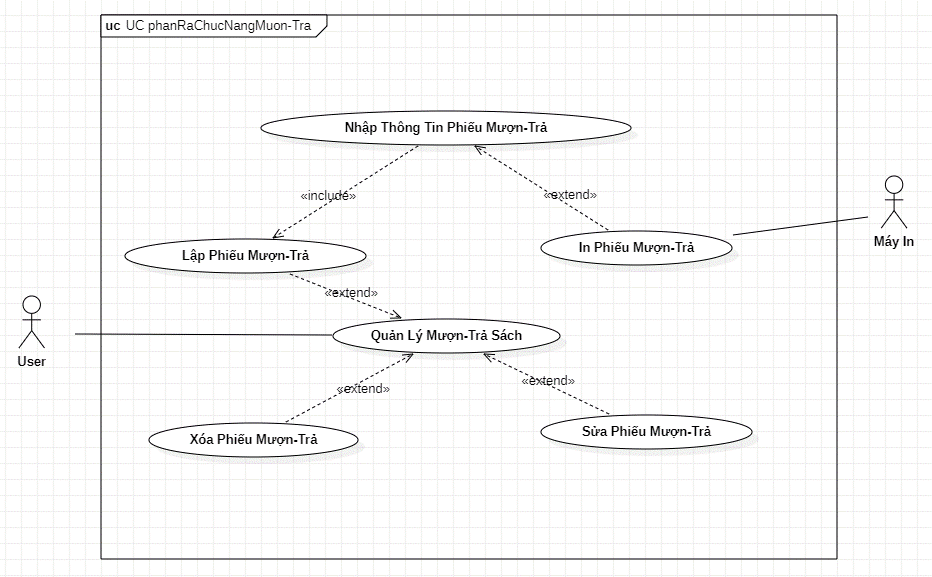


*Hình 2.2 Class Diagram*

**2.8.3. Sơ đồ Use Case tổng quát**

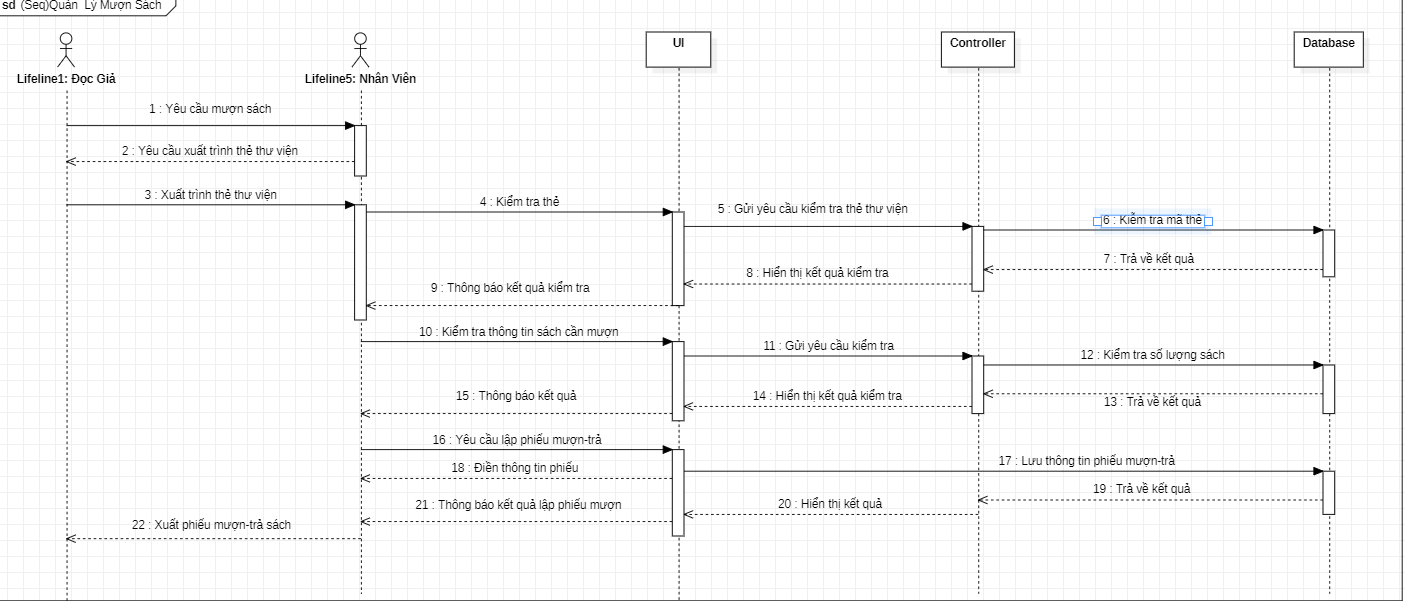


*Hình 2.6 Sơ đồ Use Case*

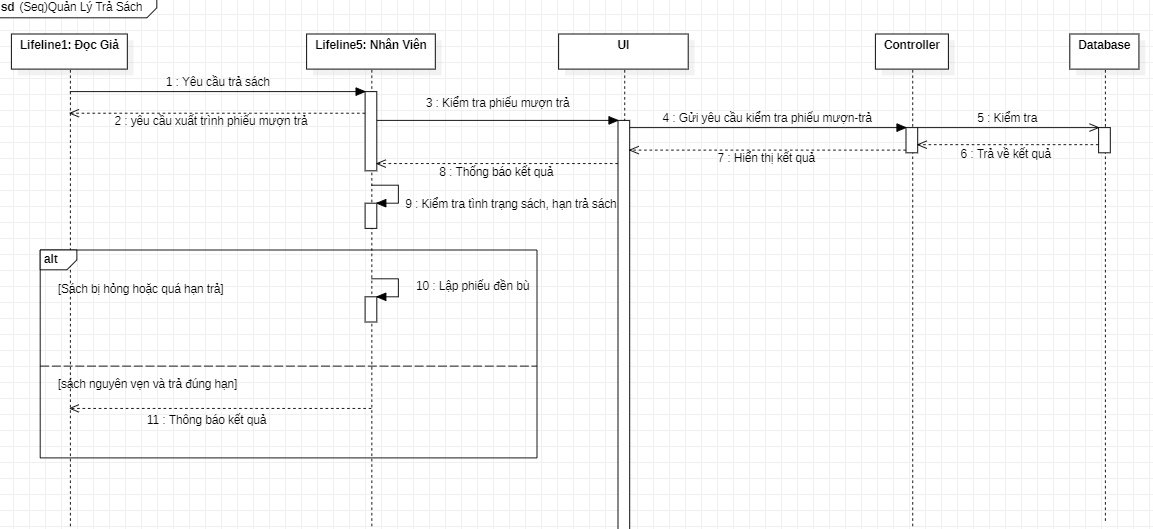


*Hình 2.7 Phân rã chức năng Mượn-Trả*

**2.8.4 Biểu đồ tuần tự (Sequence Diagram)**

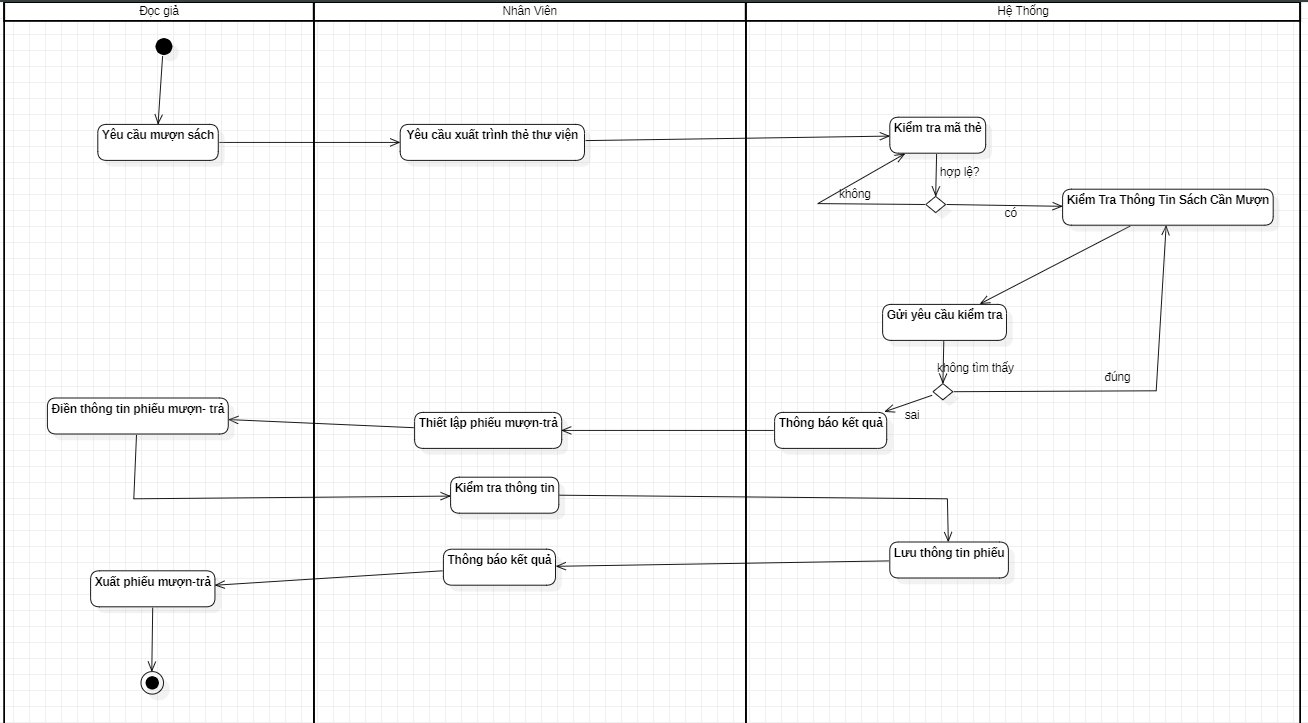


*Hình 5 :* Sequence Diagram chức năng mượn sách

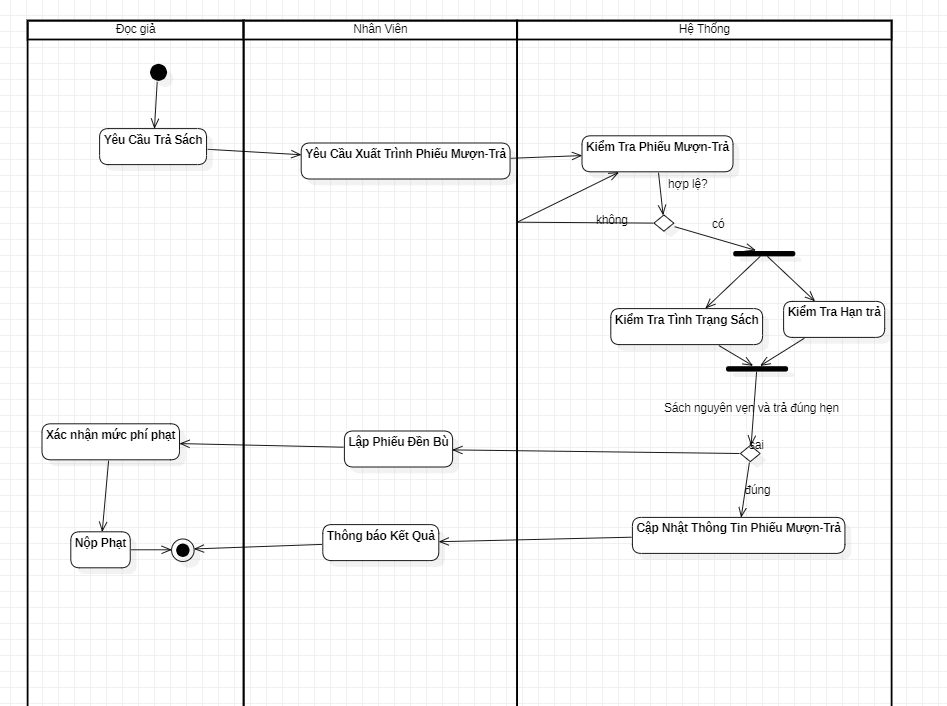


*Hình 6 :* Sequence Diagram chức năng trả sách

* + 1. **Biểu đồ hoạt động(Activity Diagram)**



*Hình 7:* Activity Diagram chức năng mượn sách



*Hình 8:* Activity Diagram chức năng trả sách

**2.9. CƠ SỞ DỮ LIỆU HOÀN THIỆN**

* + 1. **Tạo lập các bảng**: **In đậm *là khóa chính****, In nghiêng là khóa ngoại*

Bảng **DOCGIA**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaDG** | Char | 4 | Mã độc giả |
| TenDG | Nvarchar | 35 | Họ và tên độc giả |
| NgaySinhDG | Date |  | Ngày sinh độc giả |
| GioiTinh | Nvarchar | 3 | Giới tính của độc giả(nam/nữ) |
| SDTDG | varchar | 15 | Số điện thoại độc giả |

Bảng **support**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **Mssv** | Char | 10 | Mã số sinh viên |
| hoTen | Nvarchar | 35 | Họ tên |
| gioiTinh | Nvarchar | 3 | Giới tính |
| SDT | Varchar | 15 | Số điện thoại |
| Email | Varchar | 50 | Email độc giả |

Bảng **TheTV**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **maTTV** | Char | 5 | Mã thẻ thư viện |
| *MaDG* | Char | 4 | Mã độc giả |

Bảng **KESACH**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaKeSach** | Char | 6 | Mã kệ sách |
| ViTri | Nvarchar | 10 | Vị trí kệ |

Bảng **THELOAI**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaTL** | Char | 4 | Mã thể loại |
| TenTL | Nvarchar | 35 | Tên thể loại |

Bảng **NHAXUATBAN**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaNXB** | Char | 5 | Mã nhà xuất bảng |
| TenNXB | Nvarchar | 50 | Tên nhà xuất bảng |
| SDTNXB | varchar | 15 | Số điện thoại nhà xuất bảng |

Bảng **PHIEUMUON**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaPM** | Char | 4 | Mã phiếu mượn |
| *MaNV* | Char | 4 | Mã nhân viên lập phiếu |
| *MaDG* | Char | 4 | Mã độc giả mượn sách |

Bảng **NHANVIEN**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaNV** | Char | 4 | Mã nhân viên |
| TenNV | Nvarchar | 35 | Tên nhân viên |
| NgaySinhNV | Date |  | Ngày sinh nhân viên |
| SDT | Varchar | 15 | Số điện thoại nhân viên |
| GioiTinh | Nvarchar | 3 | Giới Tính(nam/nữ) |
| ChucVu | Nvarchar | 10 | Chức vụ của nhân viên |

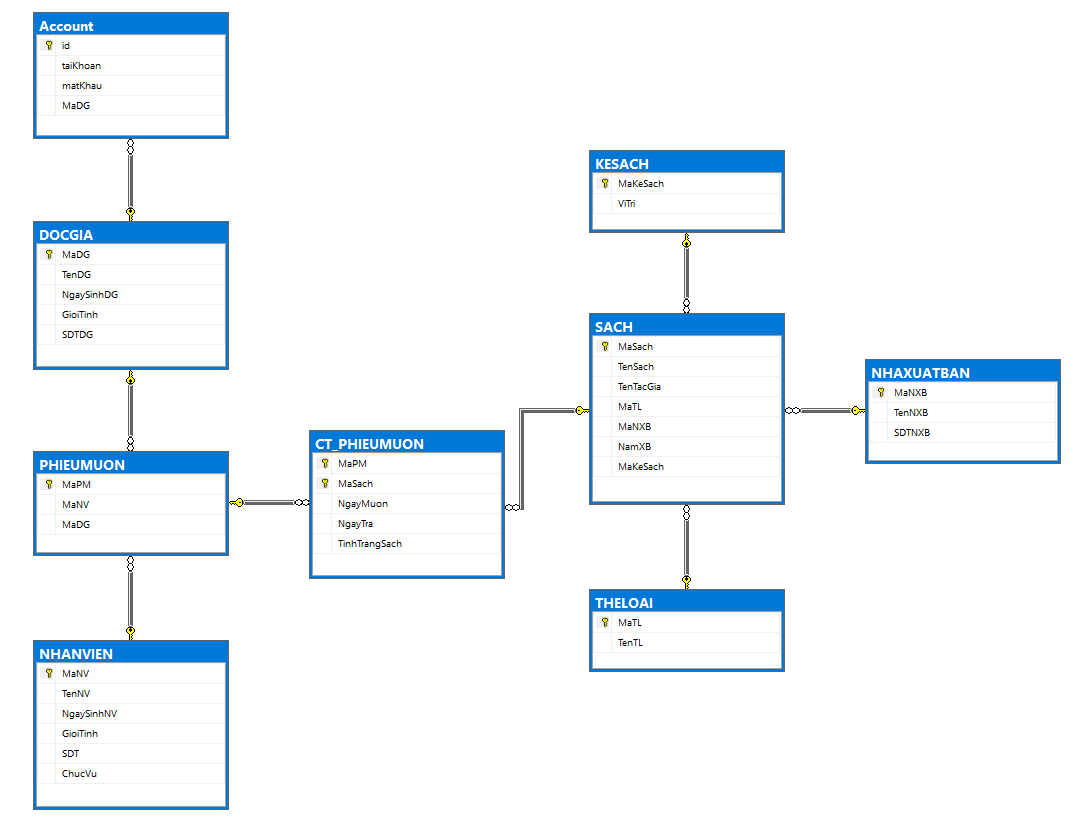
Bảng **CT\_PHIEUMUON**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaPM** | Char | 4 | Mã phiếu mượn |
| *MaSach* | Char | 4 | Mã sách được mượn |
| NgayMuon | Date |  | Ngày mượn sách |
| NgayTra | Date |  | Ngày trả sách |
| TinhTrangSach | Nvarchar | 10 | Tình trạng sách |

Bảng **SACH**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thuộc Tính | Kiểu Dữ Liệu | Kích Thước | Ghi Chú |
| **MaSach** | Char | 4 | Mã sách |
| TenSach | Nvarchar | 100 | Tên sách |
| TenTacGia | Nvarchar | 50 | Tên tác giả |
| NamXB | Int |  | Năm xuất bảng |
| *MaKeSach* | Char | 3 | Mã kệ sách |
| *MaNXB* | Char | 5 | Mã Nhà xuất bảng |
| *MaTL* | Char | 4 | Mã thể loại |

* + 1. **Thiết kế database diagram sau khi lập bảng trong MS SQL Server**

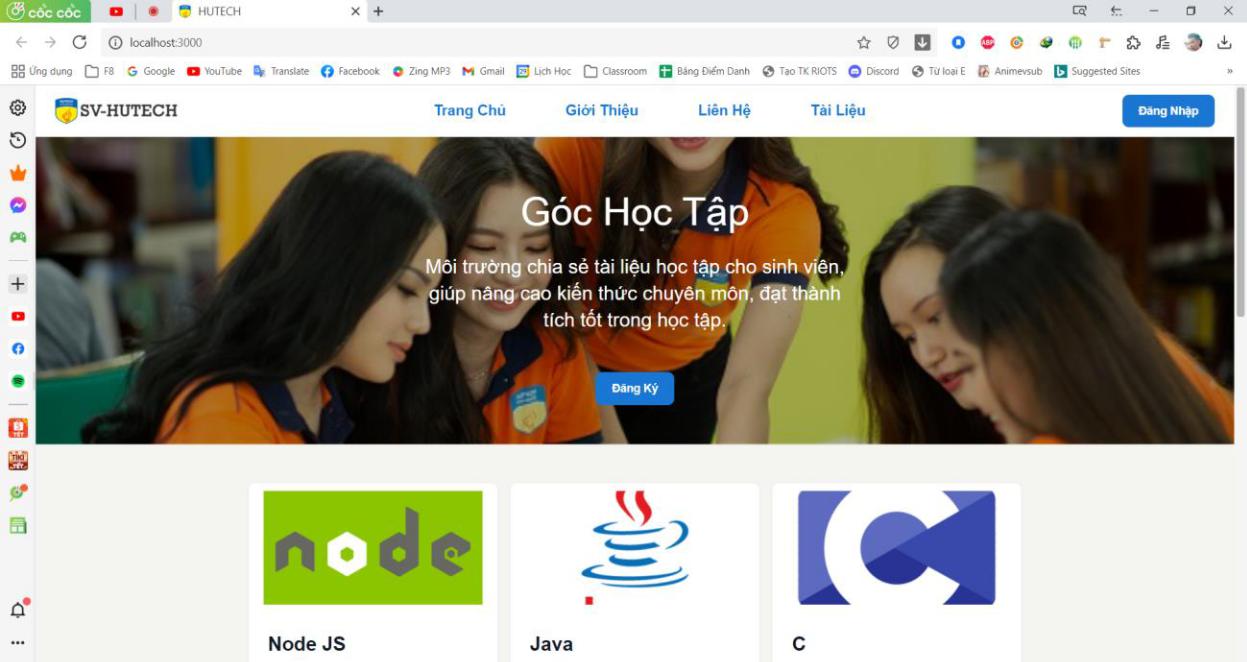


*Hình 8: Database Diagram*

**CHƯƠNG 3. GIẢI QUYẾT VẤN ĐỀ**

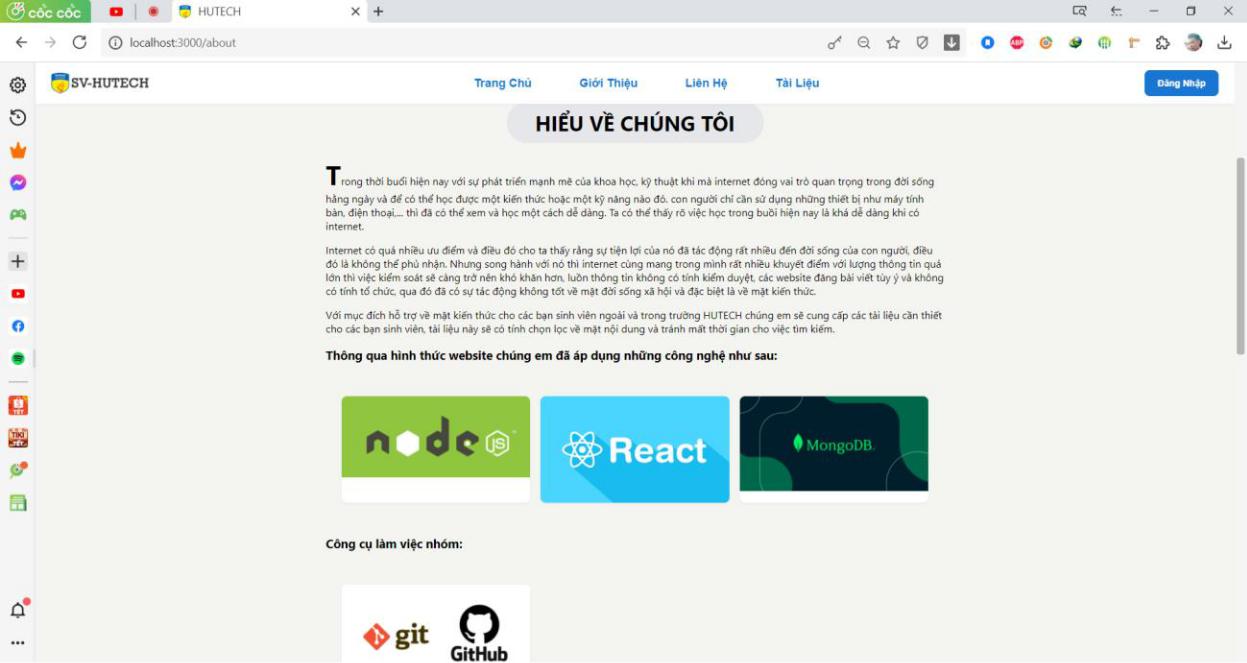
**3.1. GIAO DIỆN HỆ THỐNG**

**3.1.1. Giao diện trang chủ**



*Hình 3.1 Giao diện trang chủ.*

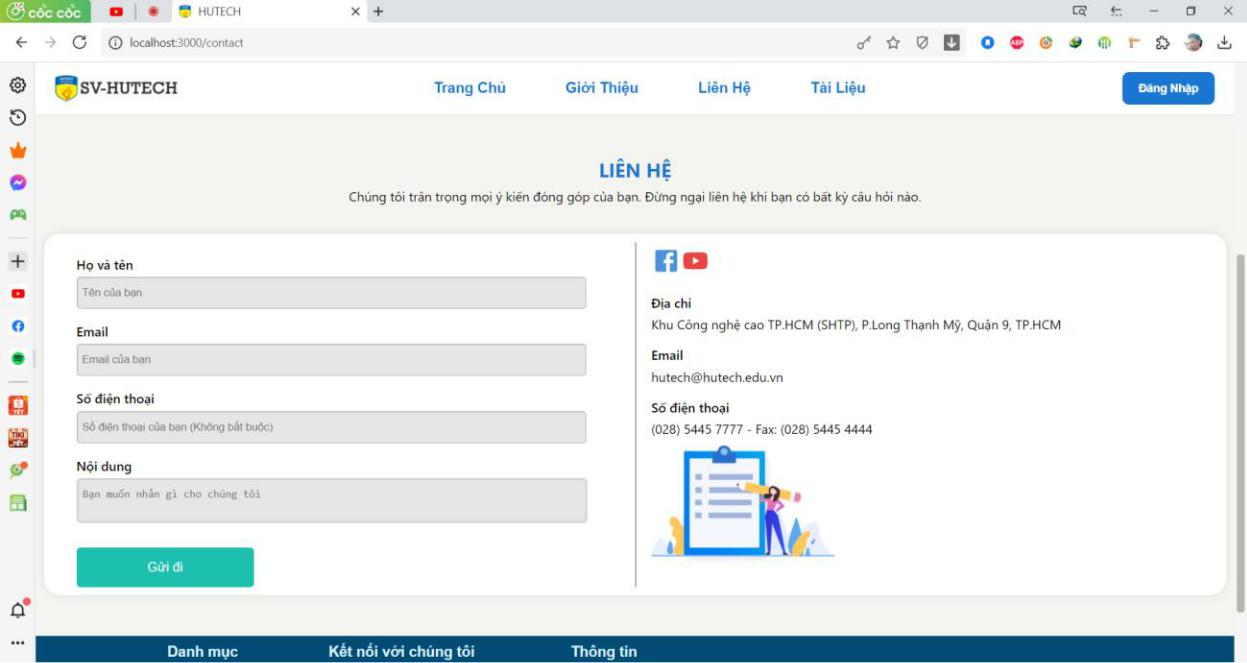
**3.1.2. Giao diện giới thiệu**



*Hình 3.2 Giao diện giới thiệu trang web.*

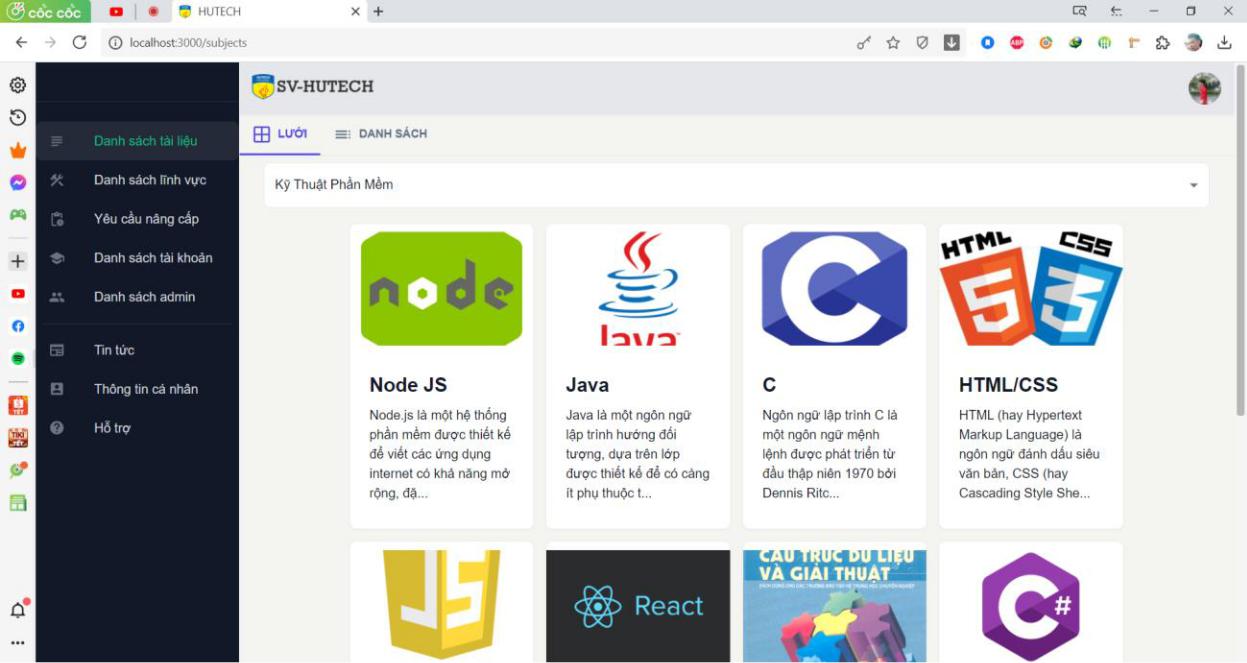
23

**3.1.3. Giao diện liên hệ**



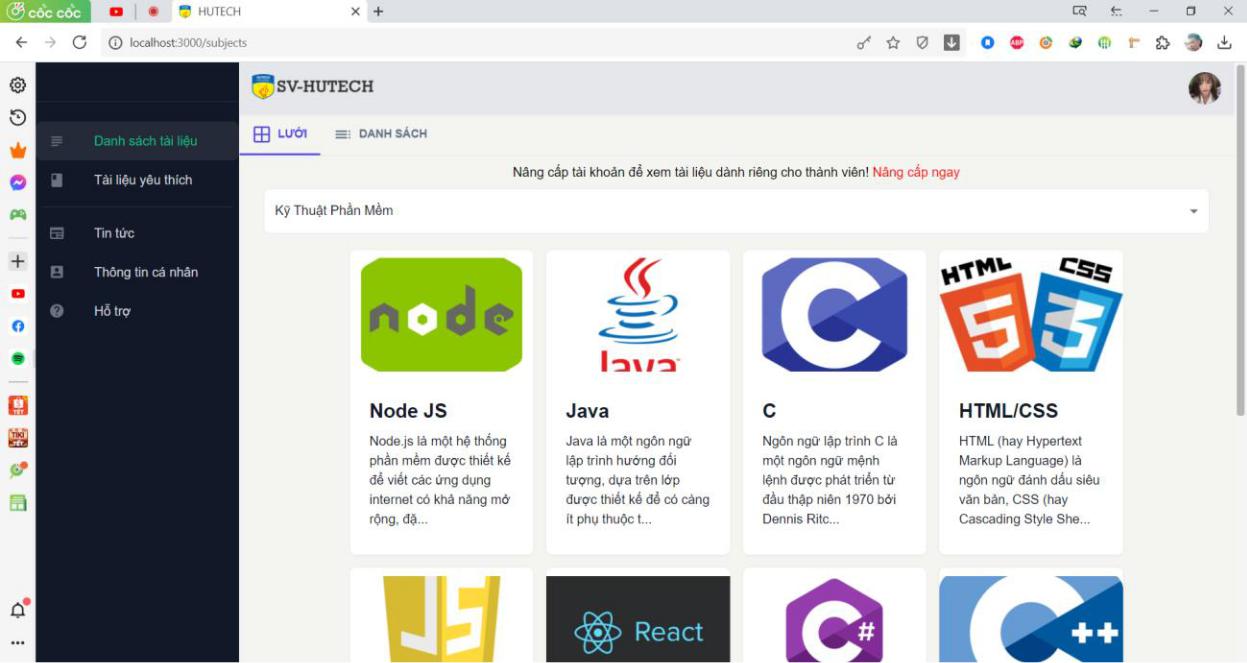
*Hình 3.3 Giao diện liên hệ nhà phát triển.*

**3.1.4. Giao diện chính**



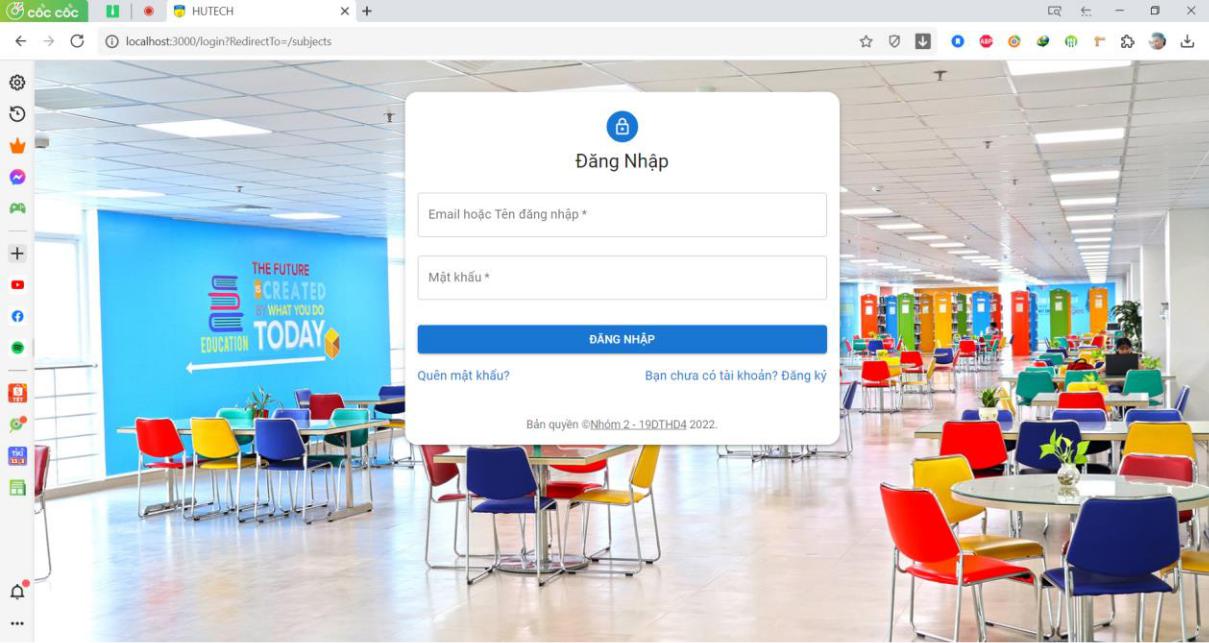
*Hình 3.4 Giao diện chính sau khi đăng nhập (Super-admin / Admin).*

24



*Hình 3.5 Giao diện chính sau khi đăng nhập (Thành viên / Khách).*

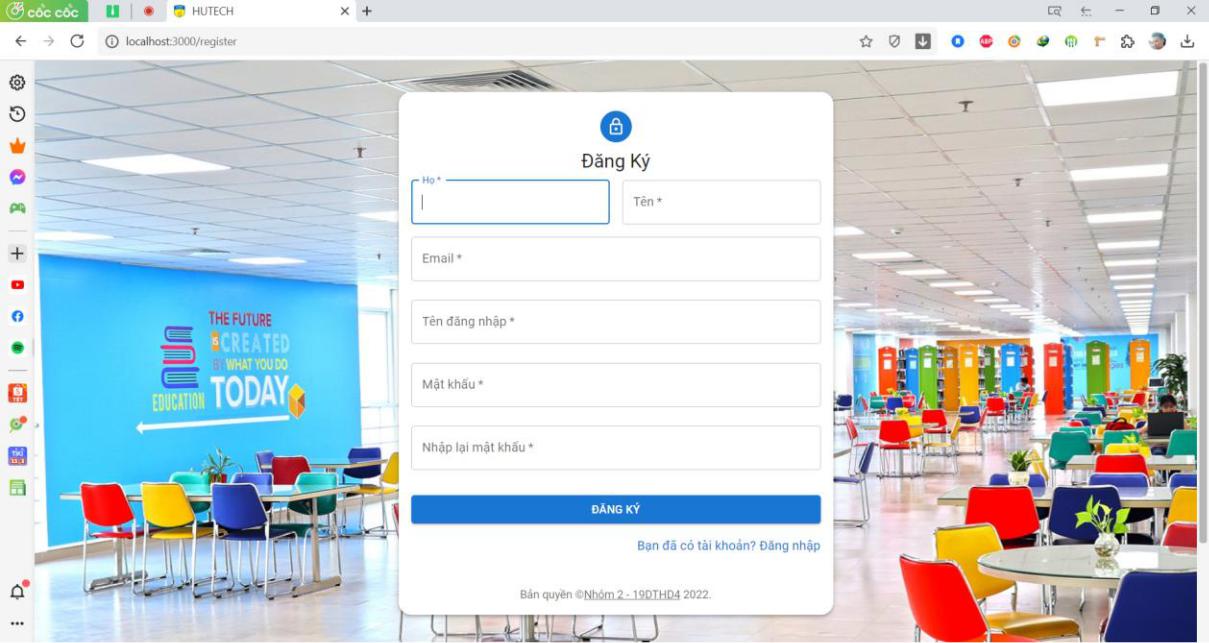
**3.1.5. Giao diện đăng nhập**



*Hình 3.6 Giao diện đăng nhập.*

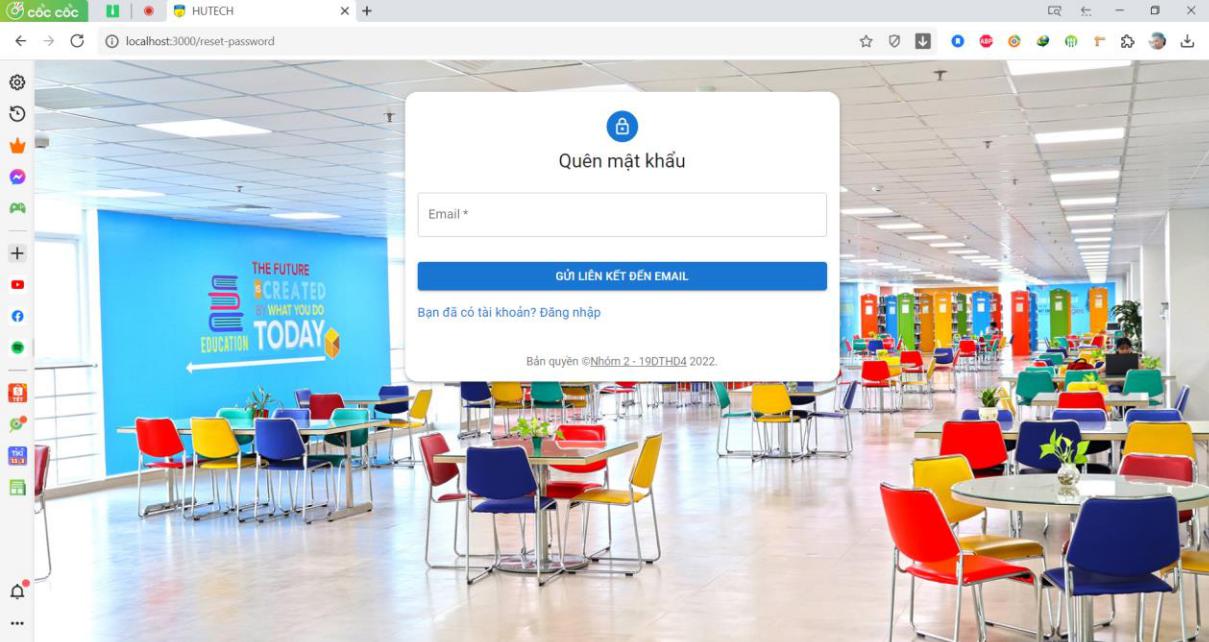
25

**3.1.6. Giao diện đăng ký tài khoản**



*Hình 3.7 Giao diện đăng ký tài khoản.*

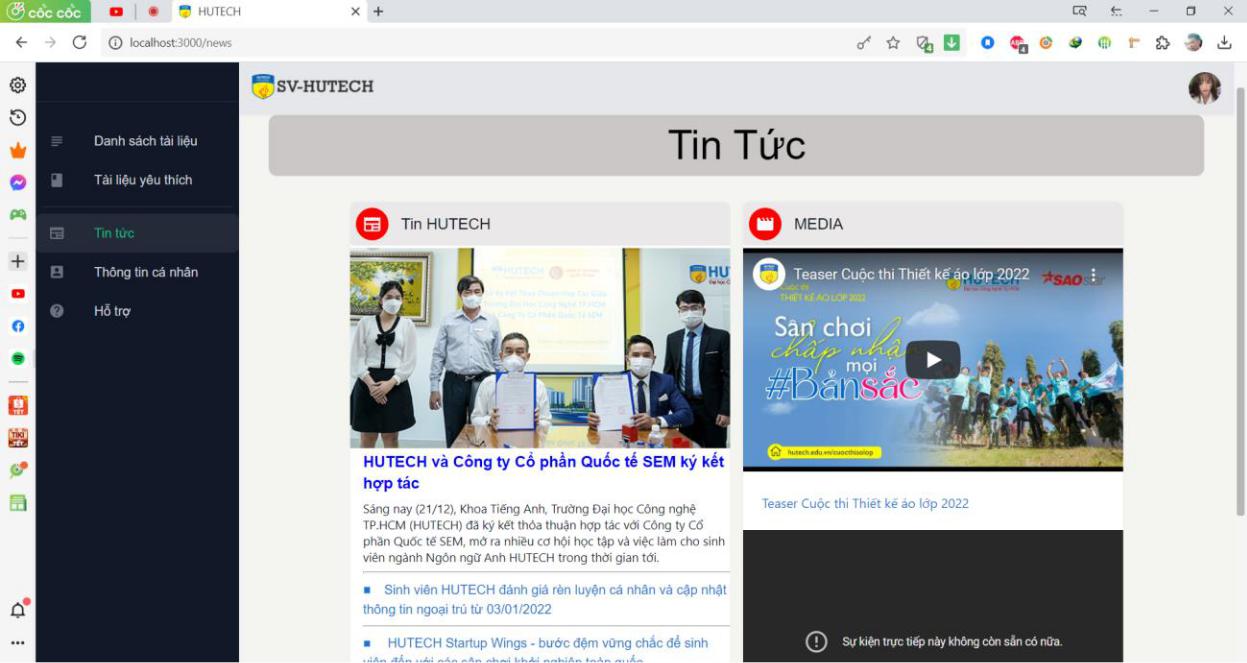
**3.1.7. Giao diện quên mật khẩu**



*Hình 3.8 Giao diện quên mật khẩu.(reset password).*

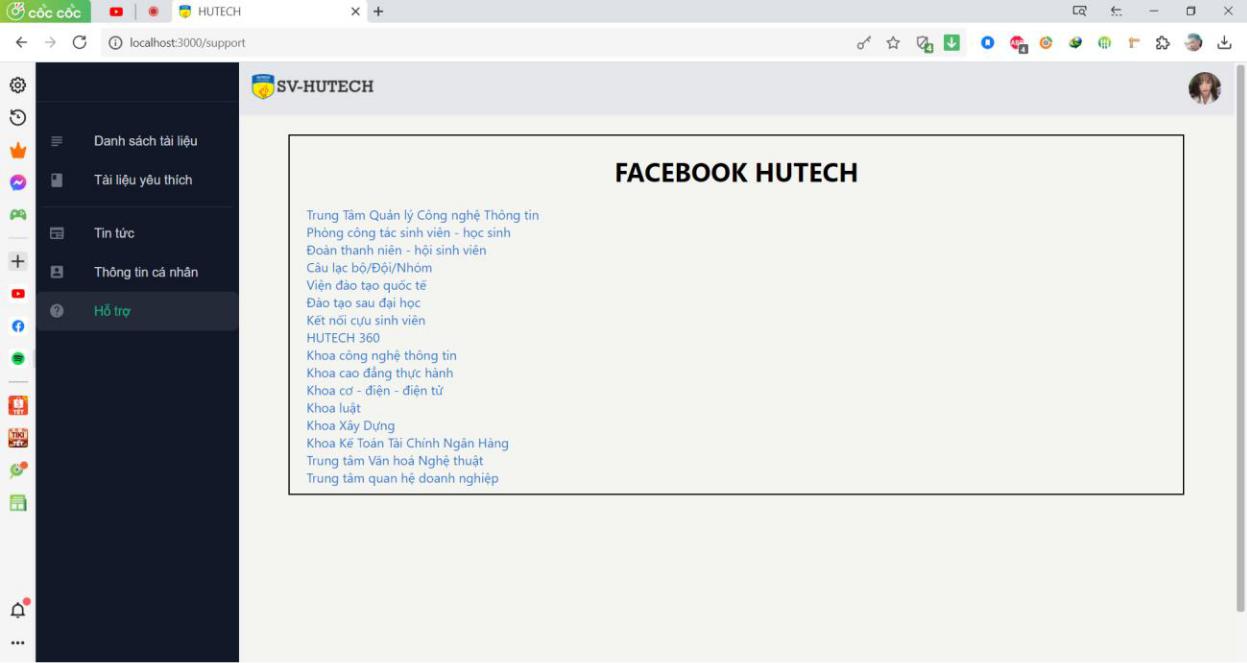
26

**3.1.8. Giao diện tin tức**



**3.1.9. Giao diện hở trợ.**

*Hình 3.9 Giao diện tin tức.*

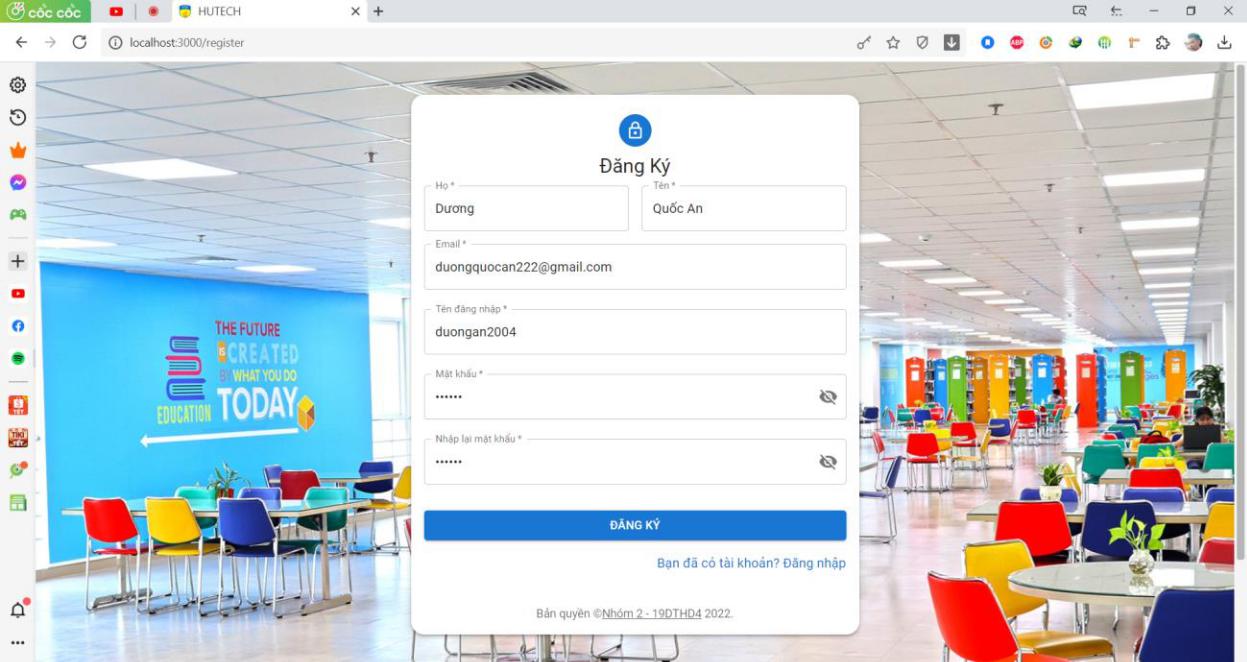


*Hình 3.10 Giao diện hổ trợ.*

27

**3.2. CÁC CHỨC NĂNG**

**3.2.1. Đăng ký – kích hoạt (xác nhận email) và đăng nhập.**



*Hình 3.11 Chức năng đăng ký tài khoản.*

Chức năng này cho phép người dùng đăng ký tài khoản.

Quá trình đăng ký bao gồm 3 bước:

* Bước 1: Điền đầy đủ thông tin đăng ký được yêu cầu.
* Bước 2: Nhấn nút “**ĐĂNG KÝ**”
  + Nếu người dùng nhập thiếu dữ liệu hoặc email, tên người dùng đã tồn tại hoặc nhập lại mặt khẩu không trung khớp thì hệ thống sẽ hiển thị thông tương ứng cho người dùng.
  + Nếu đã thoả tất cả các điều kiện hệ thống sẽ gửi mã xác nhận đến email đã đăng ký và thông báo yêu cầu người dùng kiểm tra email để kích hoạt tài khoản. (Mã kích hoạt chỉ có hiệu lực trong vòng 10 phút.)
* Bước 3: Kích hoạt tài khoản và đăng nhập.

**3.2.2. Quên mật khẩu**

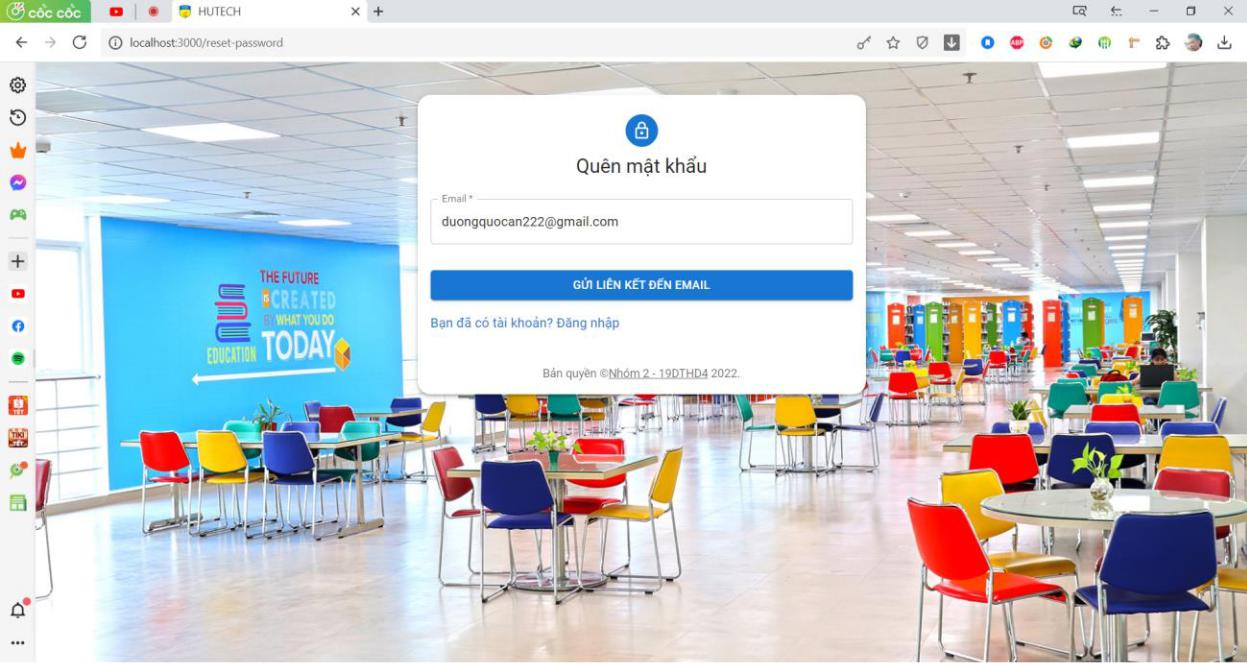
Chức năng này cho phép người dùng đặt lại mật khẩu trong trường hợp người dùng

quên mật khẩu.

Bao gồm 3 bước:

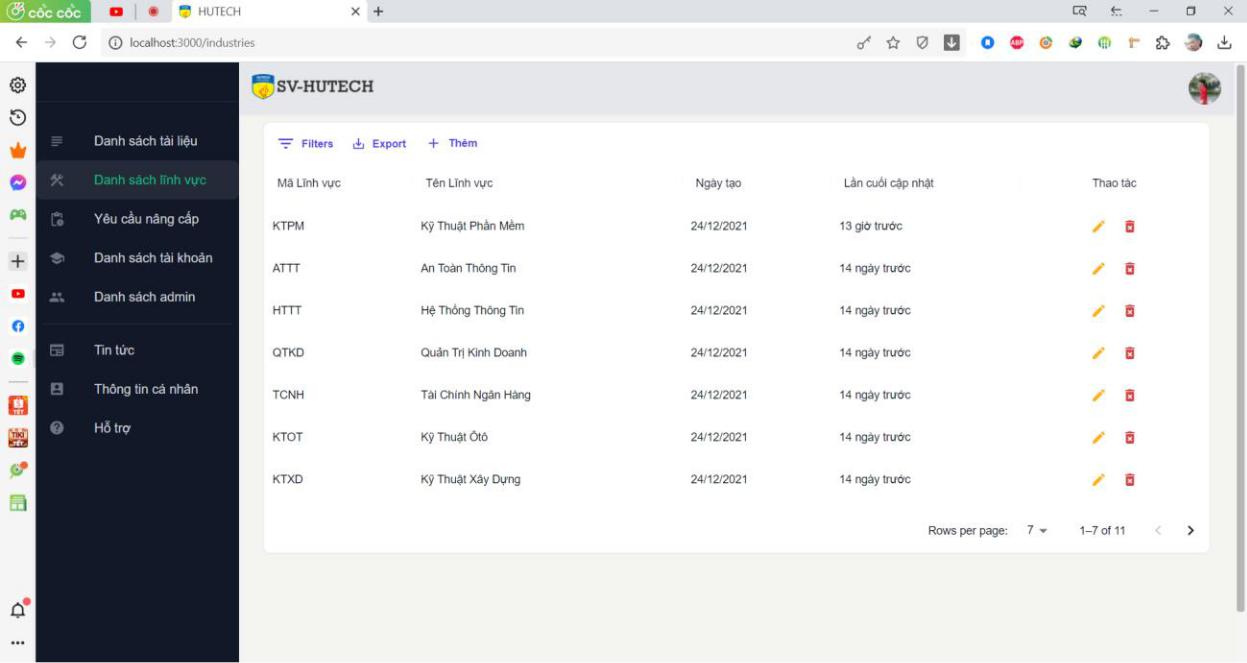
* Bước 1: Nhập email tài khoản muốn khôi phục mật khẩu.
* Bước 2: Xác nhận email và được chuyển hướng sang trang đặt lại mật khẩu.
* Bước 3: Nhập mật khẩu mới và xác nhận mật khẩu mới.

28



*Hình 3.12 Chức năng quên mật khẩu.*

**3.2.3. Quản lý lĩnh vực**



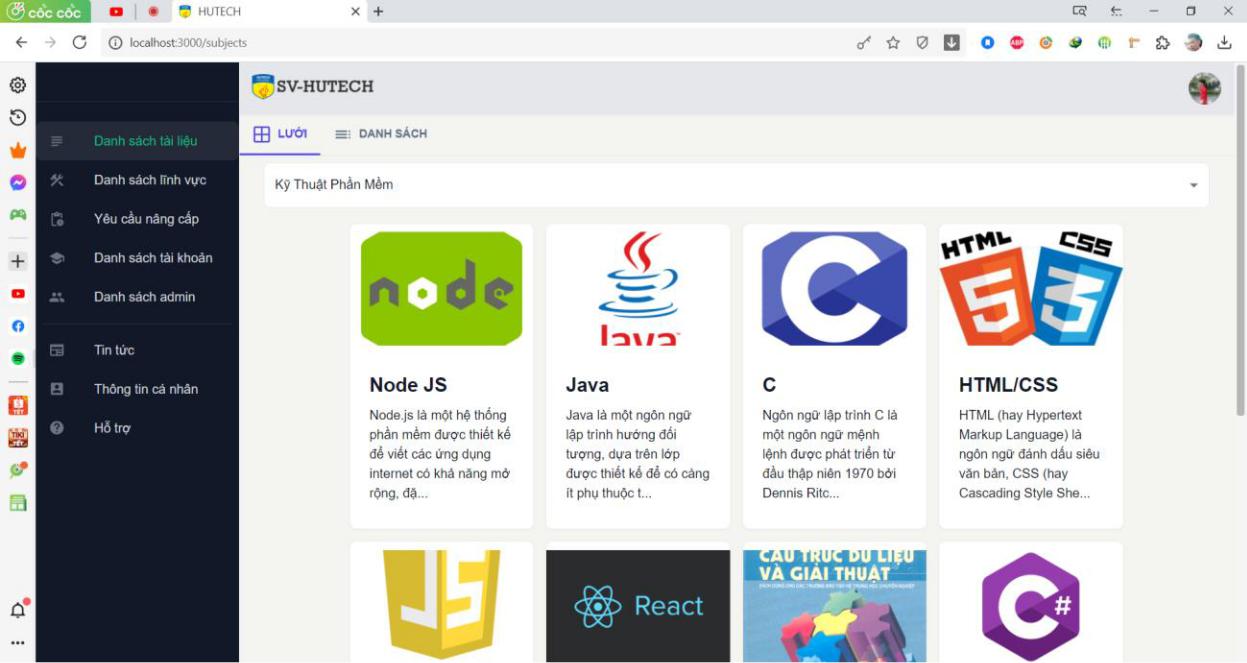
*Hình 3.13 Chức năng thêm – xoá – sửa lĩnh vực.*

Khi đăng nhập bằng tài khoản Super-admin / admin và truy cập chức năng này sẽ cho phép tìm kiếm, thêm – xoá - sửa các lĩnh vực hiện có trên hệ thống

**3.2.4. Quản lý tài liệu**

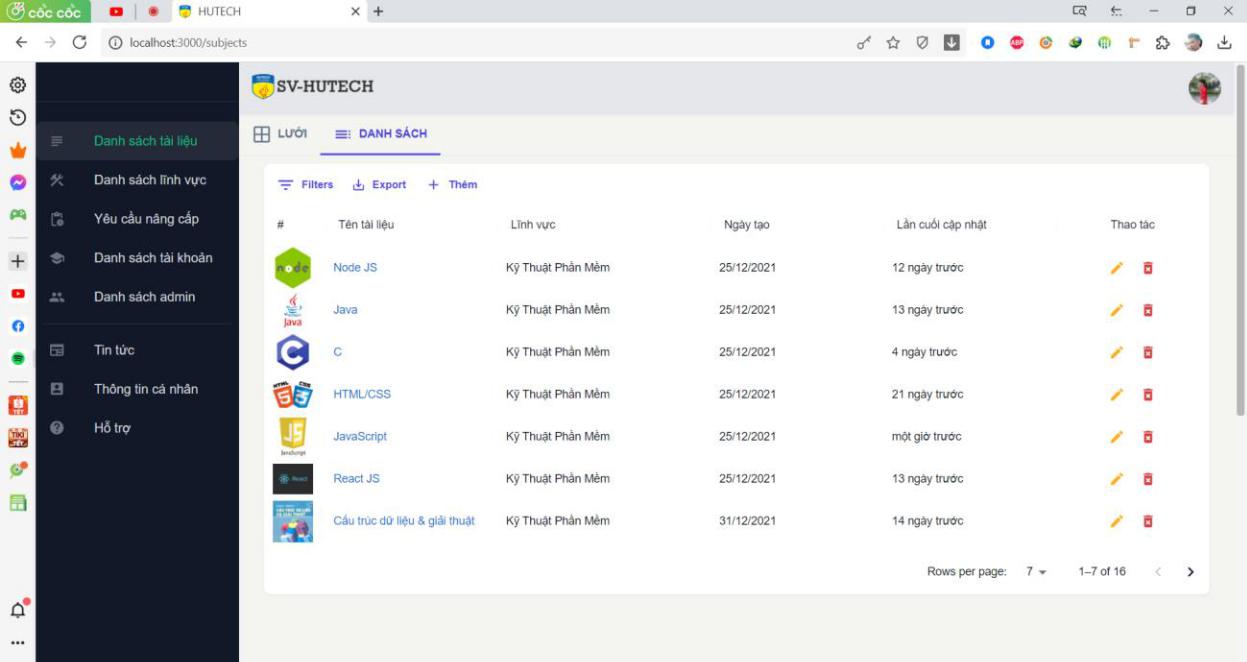
+ Hiển thị theo dạng lưới

29



*Hình 3.14 Chức năng hiển thị tài liệu theo dạng lưới.*

* Hiển thị theo dạng danh sách

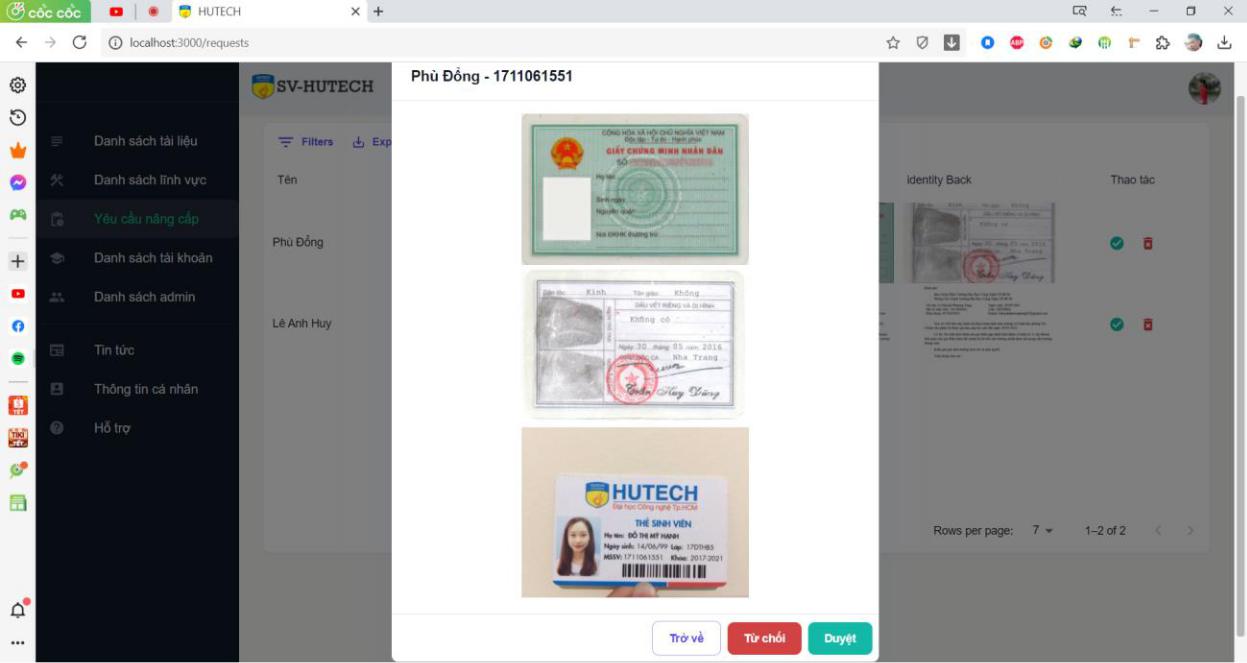


*Hình 3.15 Chức năng hiển thị tài liệu theo dạng danh sách và thêm – xoá – sửa.* Có thể xem các tài liệu theo 2 dạng: lưới và danh sách

* Lưới: có thể xem tài liệu theo lĩnh vực
* Danh sách: có thể xem tài liệu của tất cả lĩnh vực Tại đây người dùng sẽ được phân quyền sử dụng cụ thể
* Khách: chỉ được xem tài liệu công khai
* Thành viên: được xem tắt các các tài liệu hiện có trên hệ thống
* Super admin / admin: có thể tìm kiếm, thêm – xoá – sửa các tài liệu.

30

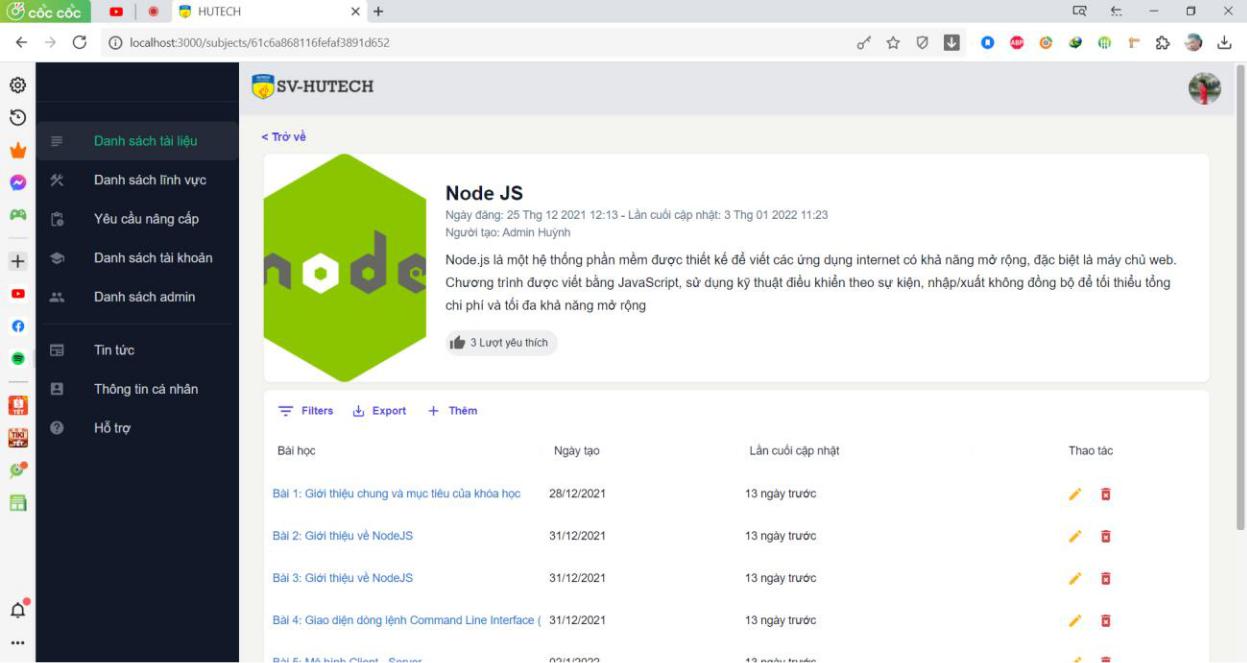
**3.2.5. Quản lý yêu cầu xác nhận thành viên.**



*Hình 3.16 Chức năng xác nhận thành viên.*

Chức năng này cho phép Super-admin / admin có thể duyệt các yêu cầu để trở thành tài khoản thành viên

**3.2.6. Quản lý bài giảng.**



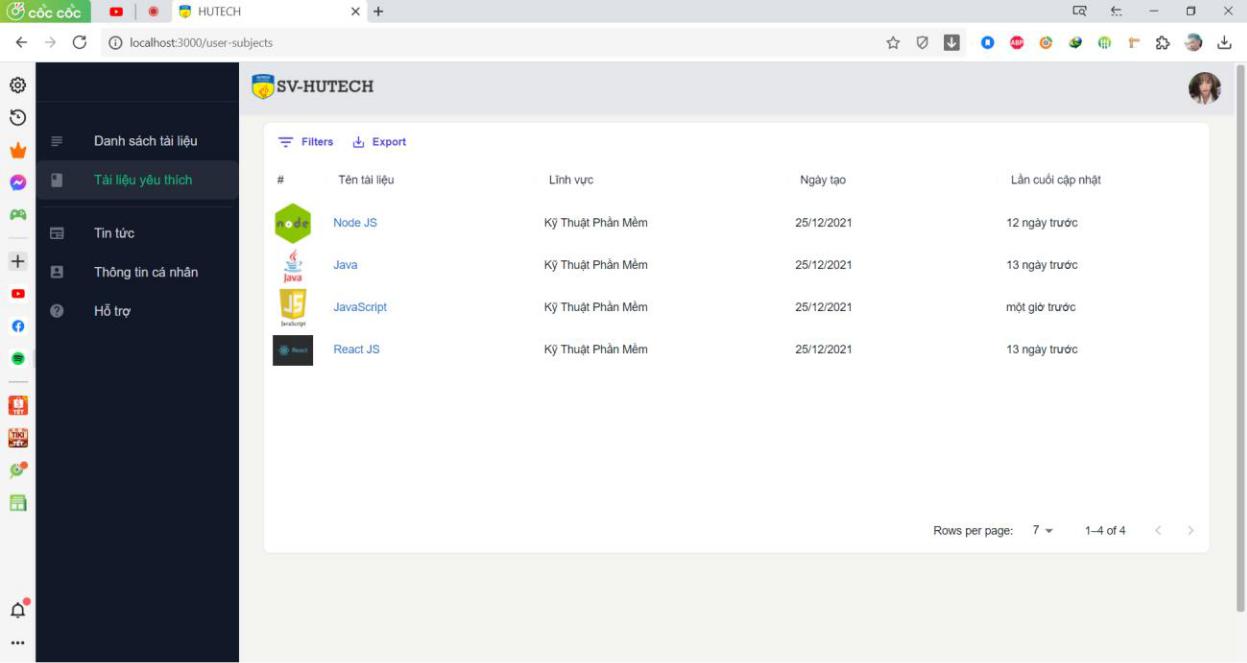
*Hình 3.17 Chức năng thêm – xoá – sửa bài giảng.*

Tại đây sẽ có bao gồm tất cả các bài giảng của tài liệu đó. Người dùng có thể xen chi tiết và tải các bài giảng đó.

Riêng Super-admin / admin có thêm chức năng chỉnh sửa bài giảng đó.

31

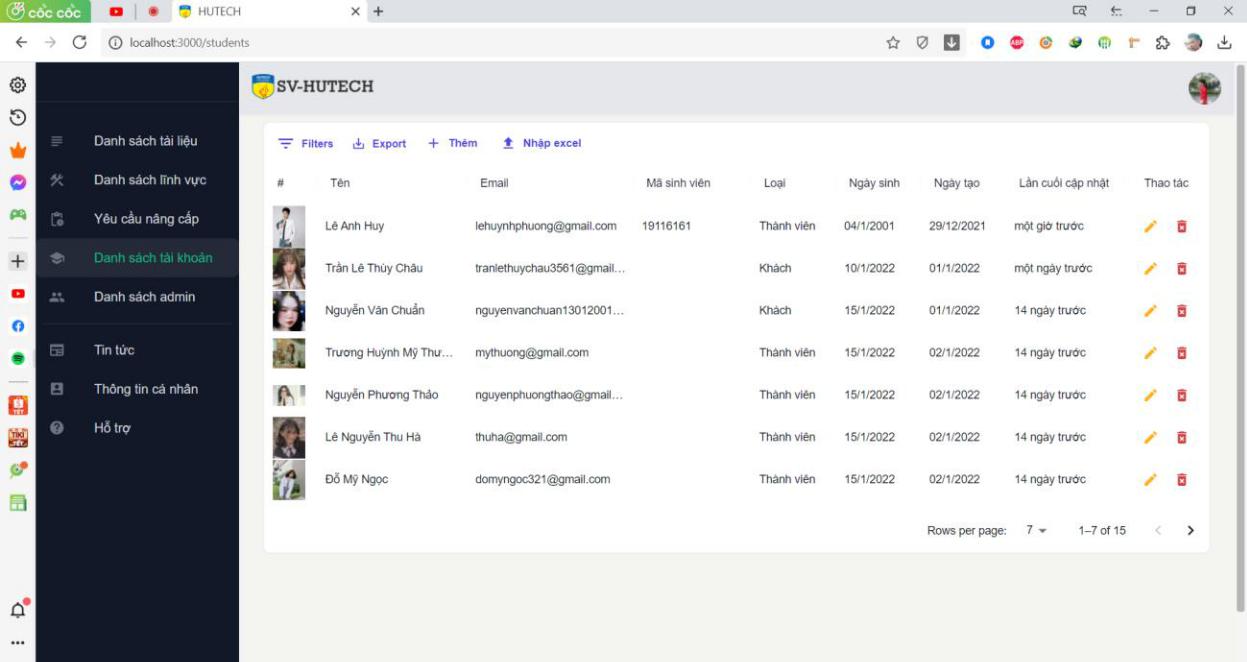
**3.2.7. Quản lý tài liệu yêu thích**



*Hình 3.18 Chức năng tìm kiếm và tải danh sách tài liệu yêu thích.*

Chức năng này cho phép người dùng tài khoản thành viên / khách có thể xem và quản lý tài liệu mình đã yêu thích trước đó.

**3.2.8. Quản lý tài khoản người dùng.**

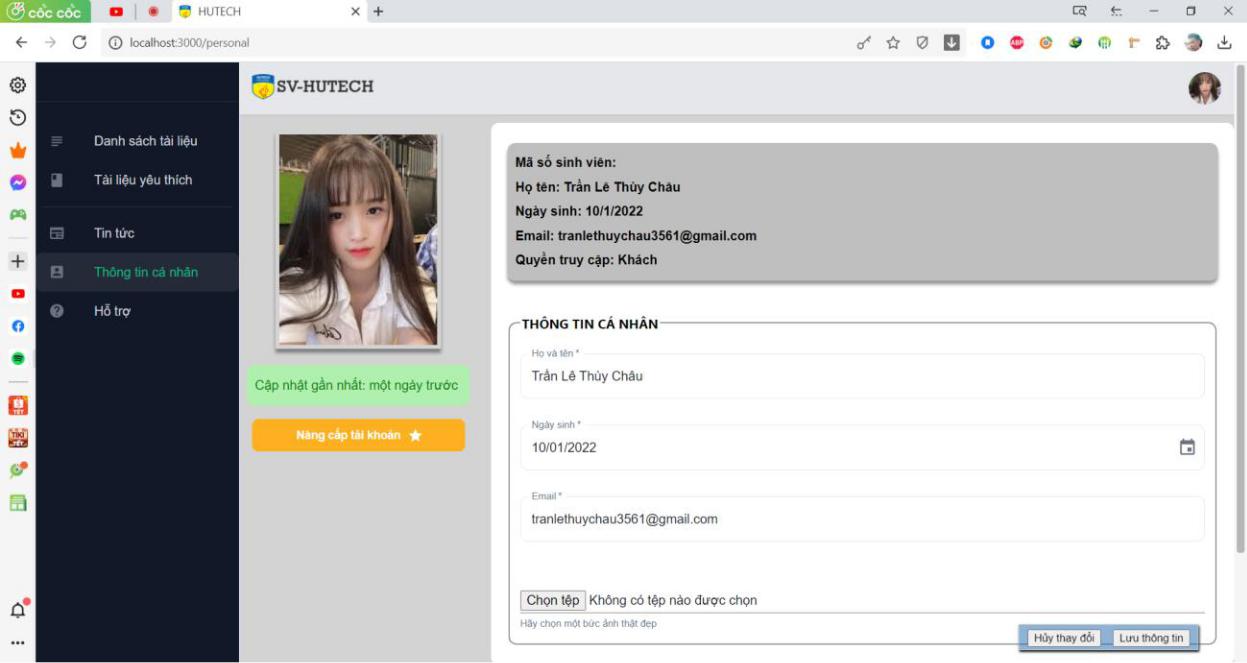


*Hình 3.19 Chức năng tìm kiếm, thêm – xoá – sửa tài khoản người dùng.*

Chức năng này cho phép **Super-admin / admin** tìm kiếm, tải xuống danh sách người dùng và chỉnh sửa tài khoản người dùng sao cho phù hợp với môi trường, nếu phát hiển hành vi quấy phá hoặc vi phạm quy tắc cộng đồng chung sẽ bị xoá tài khoản.

32

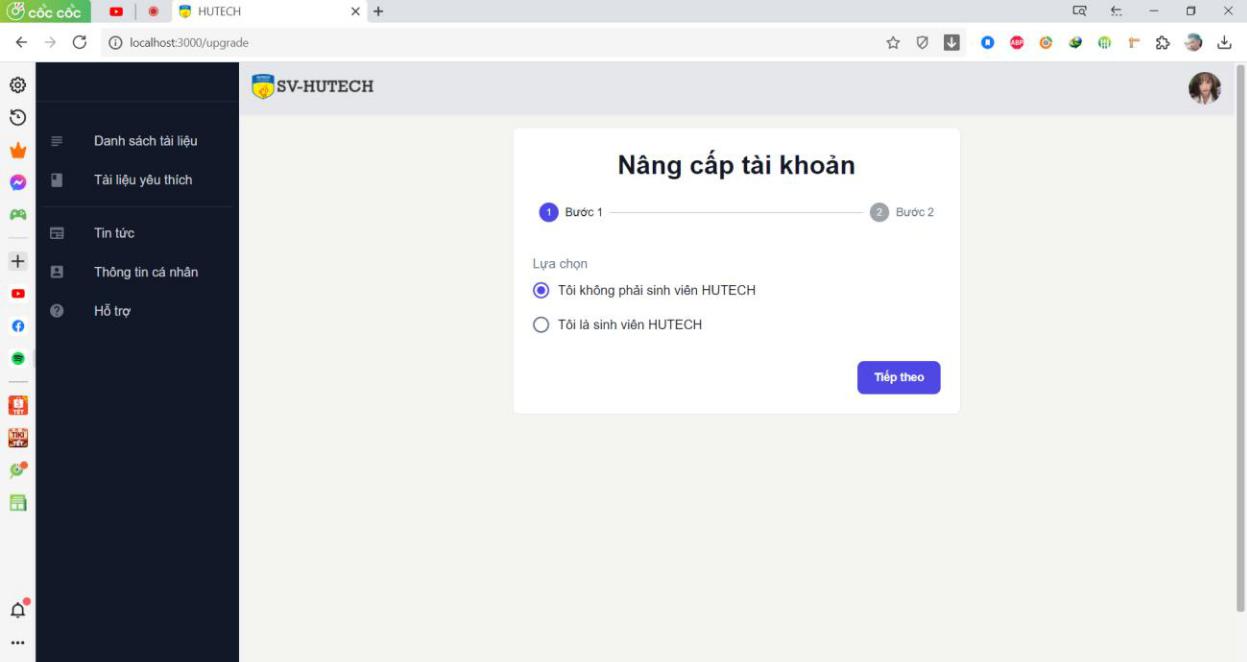
**3.2.9. Quản lý thông tin tài khoản cá nhân.**



*Hình 3.20 Chức năng xem và chỉnh sửa thông tin tài khoản cá nhân.*

Chức năng này cho phép người dung tự chỉnh sửa thông tin tài khoản cá nhân của mình và hiển thị thêm chức năng nâng cấp tài khoản nếu tài khoản đó chưa nâng cấp.

**3.2.10. Nâng cấp tài khoản**



*Hình 3.21 Chức năng yêu cầu nâng cấp tài khoản.*

Chức năng này cho phép các tài khoản người dùng chưa phải là thành viên.

Có 2 lựa chọn để yêu cầu nâng cấp:

33

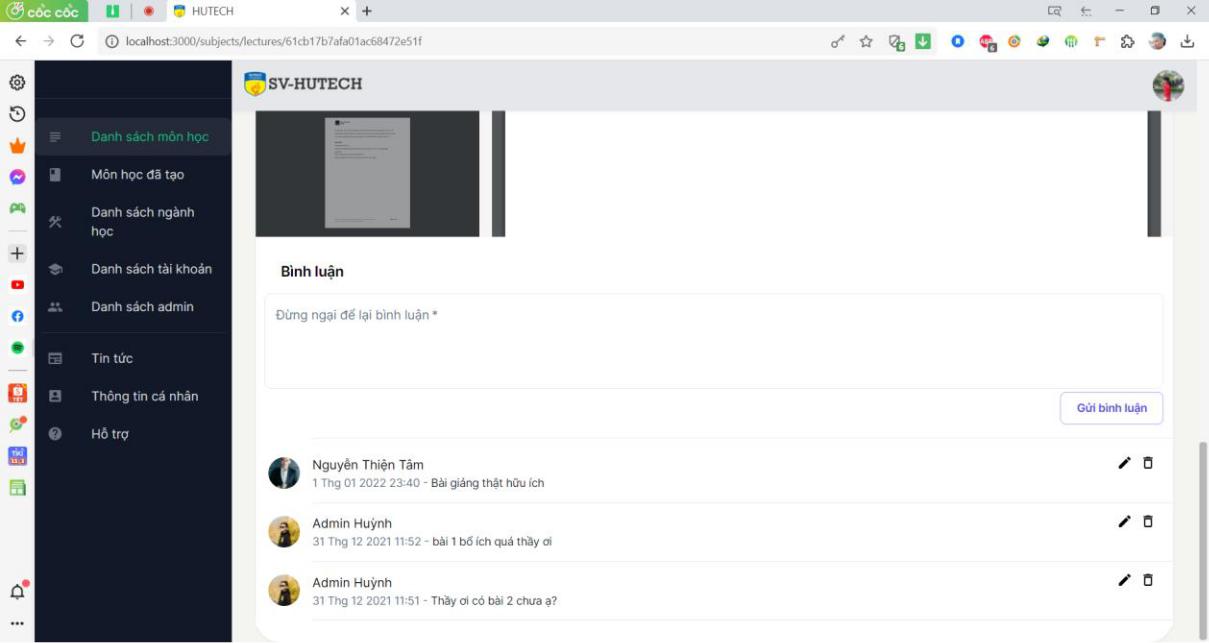
* Nếu không phải là sinh viên, người dùng bắt buộc phải bỏ phí để nâng cấp tài khoản.
* Nếu là sinh viên, người dùng cần cung cấp thông tin cá nhân và thẻ sinh viên để chứng minh.

Sau khi tài khoản được nâng cấp sẽ được quyền xem và tải các loại tài liệu riêng tư của hệ thống và có thể bình luận được tại bất kỳ các bài giảng nào.

**3.2.11. Bình luận**

**Quyền hạn**

* **Khách**: chỉ xem bình luận.
* **Thành viên**: xem và gửi bình luận.
* **Super-admin / admin**: xem - xoá - sửa binh luận và để lại các câu trả lời.



*Hình 3.22 Chức năng bình luận*

Chức năng này cho phép các tài khoản thành viên để lại bình luận, ý kiến cá nhân. Nếu trong trường hợp bình luận vi phạm quy tắc cộng đồng, mang tính xúc phạm, bôi nhọ danh dự nguòi khác,… quản lý sẽ xoá bỏ bình luận đó và xoá đặc quyền của tài khoản thành viên

**CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

**4.1. KẾT QUẢ ĐẠT ĐƯỢC**

**Giao diện q4.1.1. Những chức năng đã xây dựng hoàn thành n tr**

**ị hệ thống**

|  |
| --- |
| Giao diện hệ thống |
| Xây dựng phần mềm theo ứng dụng winform |
| Trang chủ |
| Hệ thống |
| Danh mục |
| Mượn – Trả sách |
| Tìm kiếm |
| Báo Cáo |
| Trợ giúp |

**4.1.2. Những chức năng chưa hoàn thành**

|  |
| --- |
| Các giao diện hệ thống |
| Nhân viên lập phiếu nhập sách |
| Nhân viên kiểm tra sách và lập phiếu phạt |
| Phiếu report |
| Giao diện và các chức năng sử dụng của độc giả |
| Phân quyền chức năng cho các nhân viên |
| Tạo tài khoản và đổi mật khẩu có xác nhận email |
| Chức năng quên mật khẩu |

**4.2. ĐÁNH GIÁ**

**4.2.1. Ưu điểm**

* Việc học tập và trao dồi kiến thức sẽ thuận tiện, nhanh chóng.
* Thiết kế thân thiện dể dùng giúp người dùng dể dàng sử dụng ngay lần đầu.
* Thư viện có thể kiểm soát chặt chẻ lượng sách của thư viện
* Độc giả không cần đăng ký thủ tục mượn trên giấy rườm rà
* Kiểm soát tài liệu 1 cách hệ thống

**4.2.2. Nhược điểm**

* Chưa có độ bảo mật về thông tin
* Chưa hoàn thành giao diện cho độc giả và phân quyền cho các nhân viên
* Chức năng tìm kiếm chưa có gợi ý
* Chưa kết nối LAN

**4.3. HƯỚNG PHÁT TRIỂN VÀ MỞ RỘNG**

|  |
| --- |
| Mục tiêu |
| Mã hóa dữ liệu người dùng v |
| Nâng giao diện đẹp mắt, thân thiện hơn |
| Mượn tài liệu online download sách không cần đến thư viện |
| Gán QR code cho từng mã sách để dể dàng tìm kiếm |
| Đăng nhập thư viện thông qua liên kết tài khoản sinh viên |
| Chat trực tuyến với đội ngũ hỗ trợ |

**Tài Liệu Tham Khảo**

[Thư viện Đại học Công nghệ Tp.HCM (hutech.edu.vn)](http://lib.hutech.edu.vn/)

[Lập trình phần mềm Quản Lý Quán Cafe với C# Winform - YouTube](https://www.youtube.com/playlist?list=PL33lvabfss1xnPhBJHjM0A8TEBBcGCTsf)

[Web cá nhân của thầy Dương Thành Phết | Thayphet.net](http://thayphet.net/)

[Tài liệu 2021-2022 - Google Drive](https://drive.google.com/drive/u/0/folders/1tra6nQsriZTmPJ5CBvsCfc_KNimOtmNo)

[C# là gì? Tìm hiểu về ngôn ngữ lập trình C# - CodeGym](https://codegym.vn/blog/2020/06/29/c-la-gi-tim-hieu-ve-ngon-ngu-lap-trinh-c/)

[GitHub là gì? Những lợi ích GitHub mang lại cho lập trình viên | TopDev](https://topdev.vn/blog/github-la-gi/?utm_source=google&utm_medium=cpc&utm_campaign=topdev&utm_term=github%20l%C3%A0%20g%C3%AC&utm_content=blog&gclid=Cj0KCQjwrs2XBhDjARIsAHVymmQ8D3chhCjSrJz3SIRcsDECx8r4z14R2Nt_TzHsAfca7eFJRysnmBIaAk-4EALw_wcB)

[Visual Studio là gì? Giới thiệu các tính năng phần mềm Visual Studio (visualcpp.net)](http://visualcpp.net/visual-studio-la-gi-gioi-thieu-phan-mem-visual-studio/)

[Tổng quan về SQL Server - Ngôn ngữ cần thiết cho mọi lập trình viên (sqladvice.com)](http://sqladvice.com/tong-quan-ve-sql-server/)

[Winform là gì? Được ứng dụng như thế nào? Có nên học Winform không? (plctech.com.vn)](https://plctech.com.vn/winform-la-gi/)

Tài liệu Phân tích thiết kế hệ thống thông tin – Đại học Công Nghệ TP.HCM

Tài liệu Công nghệ phần mềm – Đại học Công Nghệ TP.HCM

Tài liệu Cơ sở dữ liệu – Đại học Công Nghệ TP.HCM