

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN



Đinh Ngọc Định

XÂY DỰNG NỀN TẢNG QUẢN LÝ VĂN BẢN
CHO TRƯỜNG ĐẠI HỌC

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Chuyên ngành: Hệ thống thông tin

Hà Nội 2021

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN

Đinh Ngọc Định

XÂY DỰNG NỀN TẢNG QUẢN LÝ VĂN BẢN
CHO TRƯỜNG ĐẠI HỌC

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Chuyên ngành: Hệ thống thông tin

Cán bộ hướng dẫn: ThS. Đỗ Bảo Sơn

MỤC LỤC

MỤC LỤC.....	i
TÓM TẮT ĐỒ ÁN.....	iii
LỜI CAM ĐOAN.....	iv
LỜI CẢM ƠN	v
DANH MỤC HÌNH ẢNH	vi
DANH MỤC BẢNG BIỂU	viii
LỜI MỞ ĐẦU	ix
CHƯƠNG 1: GIỚI THIỆU CHUNG.....	1
1.1. Lý do chọn đề tài	1
1.2. Mục tiêu của đề tài	2
1.3. Giới hạn phạm vi của đề tài.....	2
1.4. Kết quả dự kiến đạt được.....	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ SỬ DỤNG	3
2.1. Cơ sở lý thuyết.....	3
2.1.1. Tìm hiểu về Front End	3
2.1.2. Tìm hiểu về Back End	5
2.1.3. Tìm hiểu về hệ quản trị cơ sở dữ liệu MSSQL Server	16
2.1.4. Tìm hiểu về mô hình hóa UML	20
2.2. Công cụ sử dụng	21
2.2.1. MSSQL Server.....	21
2.2.2. Visual Studio.....	23
2.2.3. StarUML	27
2.2.4. Visual Studio Code	27
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	31
3.1. Khảo sát hệ thống và đánh giá hiện trạng	31

3.1.1. Giới thiệu về hệ thống	31
3.1.2. Nội dung khảo sát nghiệp vụ quản lý văn bản trường đại học	36
3.1.3. Tóm tắt nghiệp vụ quản lý văn bản	50
3.1.4. Đánh giá hiện trạng hệ thống.....	53
3.1.5. Phát biểu bài toán.....	53
3.2. Phân tích và đặc tả yêu cầu.....	53
3.2.1. Xây dựng biểu đồ Usecase	54
3.2.2. Xây dựng biểu đồ lớp	61
3.2.3. Xây dựng biểu đồ trình tự.....	62
3.2.4. Xây dựng biểu đồ hoạt động.....	73
3.2.5. Xây dựng biểu đồ trạng thái	73
3.3. Thiết kế về cơ sở dữ liệu	74
3.4. Thiết kế các module chương trình chính	74
3.5. Thiết kế giao diện chương trình	74
CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH	74
4.1. Thu thập dữ liệu, xây dựng cơ sở dữ liệu vật lý.....	74
4.2. Viết code cho các module chương trình chính.....	74
4.3. Kiểm thử và đánh giá chương trình.....	74
4.4. Kết luận và hướng phát triển	74
TÀI LIỆU THAM KHẢO.....	75
NHẬN XÉT ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN	76

TÓM TẮT ĐỒ ÁN

LỜI CAM ĐOAN

Em xin cam đoan đề án này là công trình nghiên cứu của riêng em, dưới sự hướng dẫn của ThS. Đỗ Bảo Sơn – giảng viên Trường Đại học Công nghệ Giao thông vận tải. Các nội dung nghiên cứu, kết quả trong đề tài là trung thực, không sao chép bất kỳ tài liệu nào và chưa công bố nội dung này ở đâu. Các nguồn trích dẫn có chú thích rõ ràng, có tính kế thừa, phát triển từ một số tài liệu đã được liệt kê ở mục Tài Liệu Tham Khảo.

Em xin chân thành chịu trách nhiệm về lời cam đoan của mình.

Hà Nội, ngày 9 tháng 5 năm 2021

Sinh viên thực hiện

Đinh Ngọc Định

LỜI CẢM ƠN

Có thể nói đồ án như sự thể hiện kết quả của cả một quá trình học tập tại đại học. Lời cảm ơn đầu tiên em xin gửi tới Ban giám hiệu nhà trường cùng các thầy cô trong khoa Công nghệ thông tin vì đã cung cấp kiến thức, đào tạo tận tình và tạo điều kiện tốt nhất cho em trong suốt quá trình học tập và phát triển tại Trường Đại học Công nghệ GTVT.

Để hoàn thành được đồ án này, em đã cần rất nhiều sự giúp đỡ từ giảng viên hướng dẫn của em - Thầy Đỗ Bảo Sơn, em xin gửi Thầy sự tri ân và lời cảm ơn chân thành nhất, cảm ơn Thầy đã luôn tận tình hướng dẫn, khuyến khích, chỉ bảo và tạo điều kiện tốt nhất cho em từ khi bắt đầu cho tới khi hoàn thành đồ án của mình.

Trong quá trình hoàn thành đồ án của mình còn có nhiều sai sót, em mong nhận được những lời nhận xét, góp ý, chỉ bảo từ các thầy cô để hoàn thiện đồ án một cách tốt nhất.

Em xin chân thành cảm ơn!

Hà Nội, ngày 9 tháng 5 năm 2021

Sinh viên thực hiện

Đinh Ngọc Định

DANH MỤC HÌNH ẢNH

Hình 2.1: N-Tier Diagram.....	8
Hình 2.2: Mô hình MVC.....	9
Hình 2.3: Sự phát triển của ASP.....	10
Hình 2.4: Lịch sử phát triển của ASP.NET MVC.....	11
Hình 2.5: Cơ chế hoạt động của ASP.NET MVC.....	12
Hình 2.6: Cấu trúc của Entity Framework.....	14
Hình 2.7: Kiến trúc của SQL Server.....	18
Hình 2.8: Giao diện khởi động của SSMS 2014.....	22
Hình 2.9: Giao diện làm việc chính của SSML 2014.....	23
Hình 2.10: Giao diện khởi động của Visual Studio 2019.....	26
Hình 2.11: Giao diện của Star UML.....	27
Hình 2.12: Giao diện của Visual Studio Code.....	30
Hình 3.1: Trường đại học Công nghệ Giao thông vận tải – Cơ sở Hà Nội.....	31
Hình 3.2: Logo trường đại học Công nghệ Giao thông vận tải.....	33
Hình 3.3: Sơ đồ tổ chức trường đại học Công nghệ Giao thông vận tải.....	34
Hình 3.4: Biểu đồ Usecase tổng quát.....	55
Hình 3.5: Biểu đồ Usecase quản lý tài khoản.....	55
Hình 3.6: Biểu đồ Usecase quản lý phòng ban.....	56
Hình 3.7: Biểu đồ Usecase quản lý loại văn bản.....	57
Hình 3.8: Biểu đồ Usecase quản lý văn bản đến.....	58
Hình 3.9: Biểu đồ Usecase quản lý văn bản đi.....	59
Hình 3.10: Biểu đồ Usecase quản lý cá nhân.....	60
Hình 3.11: Biểu đồ Usecase thống kê.....	61
Hình 3.12: Biểu đồ lớp.....	62
Hình 3.13: Biểu đồ trình tự đăng nhập.....	63
Hình 3.14: Biểu đồ trình tự thêm tài khoản.....	63

Hình 3.15: Biểu đồ trình tự cập nhật tài khoản.	64
Hình 3.16: Biểu đồ trình tự thêm phòng ban.	64
Hình 3.17: Biểu đồ trình tự cập nhật phòng ban.	65
Hình 3.18: Biểu đồ trình tự xóa phòng ban.	65
Hình 3.19: Biểu đồ trình tự thêm loại văn bản.	66
Hình 3.20: Biểu đồ trình tự cập nhật loại văn bản.	66
Hình 3.21: Biểu đồ trình tự xóa loại văn bản.	67
Hình 3.22: Biểu đồ trình tự thêm văn bản đến.	67
Hình 3.23: Biểu đồ trình tự cập nhật văn bản đến.	68
Hình 3.24: Biểu đồ trình tự xóa văn bản đến.	68
Hình 3.25: Biểu đồ trình tự duyệt văn bản đến.	69
Hình 2.26: Biểu đồ trình tự ban hành văn bản đến.	69
Hình 3.27: Biểu đồ trình tự thêm văn bản đi.	70
Hình 3.28: Biểu đồ trình tự cập nhật văn bản đi.	70
Hình 3.29: Biểu đồ trình tự xóa văn bản đi.	71
Hình 3.30: Biểu đồ trình tự ký duyệt văn bản đi.	71
Hình 3.31: Biểu đồ trình tự thay đổi thông tin cá nhân.	72
Hình 3.32: Biểu đồ trình tự đổi mật khẩu.	72
Hình 3.33: Biểu đồ trạng thái tài khoản.	73
Hình 3.34: Biểu đồ trạng thái văn bản đến.	73
Hình 3.35: Biểu đồ trạng thái văn bản đi.	73

DANH MỤC BẢNG BIỂU

Bảng 3.1: Tóm tắt quy trình quản lý văn bản đến.....	51
Bảng 3.2: Tóm tắt quy trình quản lý văn bản đi.	52
Bảng 3.3: Kịch bản Usecase quản lý tài khoản.....	56
Bảng 3.4: Kịch bản Usecase quản lý phòng ban.....	56
Bảng 3.5: Kịch bản Usecase quản lý loại văn bản	57
Bảng 3.6: Kịch bản Usecase quản lý văn bản đến	58
Bảng 3.7: Kịch bản Usecase quản lý văn bản đi.....	59
Bảng 3.8: Kịch bản Usecase quản lý cá nhân	60
Bảng 3.9: Kịch bản Usecase thống kê.....	61

LỜI MỞ ĐẦU

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Lý do chọn đề tài

Công nghệ thông tin là tập hợp các phương pháp khoa học và các công cụ kỹ thuật hiện đại, chủ yếu là kỹ thuật máy tính và viễn thông nhằm tổ chức khai thác và sử dụng có hiệu quả các nguồn tài nguyên thông tin phong phú, tiềm năng trong mọi lĩnh vực hoạt động của con người và xã hội.

Ngày nay, công nghệ thông tin phát triển nhanh chóng và ứng dụng vào tất cả các lĩnh vực, có thể nói công nghệ thông tin trở thành thước đo để đánh giá sự phát triển của xã hội hiện đại – nơi mà con người đang từ bỏ cách làm việc thủ công, tiến đến tin học hóa trong tất cả các lĩnh vực, để công việc thực hiện có hiệu quả hơn, tiết kiệm thời gian và nhân lực hơn.

Do đó, ứng dụng của công nghệ thông tin vào việc quản lý dường như không còn xa lạ với các doanh nghiệp. Lợi ích mà các hệ thống quản lý đem lại khiến ta không thể không thừa nhận tính hiệu quả của nó.

Trong một vài năm gần đây, cùng với sự phát triển của khoa học, công nghệ thì việc sử dụng phần mềm quản lý văn bản, lưu trữ thông tin thay cho các cách làm truyền thống được ứng dụng rất nhiều trong các cơ quan, doanh nghiệp. Thay vì lưu văn bản bằng phương pháp thủ công truyền thống như lưu trữ trong tủ, hòm... thì tất cả dữ liệu đều được lưu trữ trên hệ thống phần mềm quản lý.

Nếu như với các phương pháp cũ việc tra cứu, tìm kiếm và lưu trữ tài liệu thường tốn rất nhiều thời gian. Đồng thời cần một không gian rộng để chứa đựng tài liệu, với các văn bản cũ thường phải tiêu hủy sau một khoảng thời gian nhất định để tránh sự quá tải. Nhằm khắc phục tình trạng trên, phần mềm quản lý văn bản sẽ giúp doanh nghiệp số hóa được dữ liệu, mật mã hóa được thông tin.

Bên cạnh đó chúng cũng giúp đơn giản hóa quá trình lưu trữ dữ liệu. Khi cần tìm kiếm, người dùng chỉ cần thực hiện các thao tác đơn giản, hệ thống sẽ trả ra thông tin tìm kiếm trong khoảng thời gian rất ngắn và có độ chính xác tuyệt đối.

Việc áp dụng phần mềm vào việc quản lý tập tin văn bản, giúp doanh nghiệp thiết lập được một quy trình quản lý chặt chẽ. Ngoài ra, phần mềm còn có thể lưu trữ được dung lượng lớn các thông tin, kết nối quản lý qua mạng nội bộ, giúp người dùng dễ dàng thao tác tìm kiếm từ xa. Mà không tốn công sức và thời gian.

Một ưu điểm không thể không đề cập đến khi sử dụng hệ thống lưu trữ công nghệ, đó chính là tính bảo mật tuyệt đối, tránh thất thoát và mất mát thông tin.

Sử dụng phần mềm chuyên dụng để quản lý văn bản được xem là phương pháp khoa học và chuyên nghiệp nhất. Chính vì những lý do trên nên em quyết định chọn đề tài “Xây dựng nền tảng quản lý văn bản cho nhà trường” làm đề án tốt nghiệp của mình.

1.2. Mục tiêu của đề tài

- Nhằm bắt được quy trình quản lý văn bản của trường đại học.
- Áp dụng công nghệ thông tin để xây dựng, thiết kế một nền tảng quản lý văn bản cho trường đại học với các chức năng cần thiết.
- Triển khai và cài đặt chương trình trong thực tế.
- Kiểm chứng những kiến thức đã được học trên giảng đường Đại học, đồng thời trang bị những kiến thức mới, là hành trang chuẩn bị bước vào cuộc sống.

1.3. Giới hạn phạm vi của đề tài

Trong phạm vi đề tài này, em sẽ nghiên cứu các vấn đề:

- Nghiên cứu về công tác quản lý văn bản của trường đại học, đi sâu vào nghiên cứu và phân tích một hệ thống thông tin để xây dựng ứng dụng.

1.4. Kết quả dự kiến đạt được

Hệ thống thông tin quản lý văn bản của trường đại học khi hoàn thành dự kiến đạt được kết quả sau:

- Quản lý văn bản đến
- Quản lý văn bản đi
- Quản lý phòng ban, đơn vị
- Quản lý loại văn bản
- Trình ký văn bản
- Quản lý thông tin người dùng

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ SỬ DỤNG

2.1. Cơ sở lý thuyết

2.1.1. Tìm hiểu về Front End

Phần front-end của một trang web là phần tương tác với người dùng. Tất cả mọi thứ bạn nhìn thấy khi điều hướng trên Internet, từ các font chữ, màu sắc cho tới các menu xổ xuống và các thanh trượt, là một sự kết hợp của HTML, CSS, và JavaScript được điều khiển bởi trình duyệt máy tính của bạn.

2.1.1.1. HTML

a. Lịch sử

Thuở sơ khai của internet mọi người thường chia sẻ tài liệu văn bản cho nhau thông qua một giao thức nào đó. Người nhận sẽ tải tệp tin về máy tính và đọc tệp tin bằng chương trình tương ứng.

Đến khi World Wide Web ra đời việc chia sẻ, xem, chỉnh sửa các tài liệu văn bản có thể thực hiện trực tuyến thông qua một phần mềm có kết nối internet gọi là trình duyệt.

Các tài liệu văn bản này sử dụng ngôn ngữ đánh dấu HTML kết hợp với các đoạn văn bản (text) để xây dựng cấu trúc tài liệu, cách thức các text đó thể hiện trên trình duyệt. Và chúng được gọi là các siêu văn bản.

Cùng với sự phát triển nhanh chóng của World Wide Web, HTML dần trở thành một ngôn ngữ vô cùng hữu ích đối với các nhà phát triển browser. Nhiều trình duyệt được xây dựng và phát triển HTML theo hướng độc lập nhau. Điều này tạo nên nhiều rắc rối khi một website lại thể hiện khác nhau ở các browser khác nhau.

Dần nhận thấy sự phát triển tràn lan của ngôn ngữ HTML và những rắc rối mà nó mang lại, W3C đã đi đến thống nhất và phê duyệt bộ ngôn ngữ HTML chuẩn.

b. Giới thiệu

HTML là viết tắt của HyperText Markup Language (ngôn ngữ đánh dấu siêu văn bản) dùng mô tả cấu trúc của các trang Web và tạo ra các loại tài liệu có thể xem được trong trình duyệt.

HTML là một tiêu chuẩn quốc tế có các thông số kỹ thuật được duy trì bởi World Wide Web Consortium.

2.1.1.2. CSS

a. Lịch sử

Khi World Wide Web phát triển với quy mô lớn với các trang web hoành tráng, dòng code HTML dần trở nên hỗn độn phức tạp, yêu cầu về trình bày, sắp xếp hiển thị của mọi người đối với siêu văn bản tăng lên (ví dụ hiển thị siêu văn bản như một mặt báo chẵn hạn). Vì lẽ đó, W3C đã phê duyệt và tung ra một ngôn ngữ mới bổ sung gọi là CSS (Cascading Style Sheets).

CSS trở thành một công cụ đắc lực dọn dẹp phần code HTML một cách gọn gàng, có cấu trúc hơn và định dạng HTML để tạo ra những layout đa dạng, đẹp hơn.

b. Giới thiệu

CSS (viết tắt của Cascading Style Sheets) là một ngôn ngữ định dạng được sử dụng để mô tả trình bày các trang Web, bao gồm màu sắc, cách bố trí và phông chữ. Nó cho phép hiển thị nội dung tương thích trên các loại thiết bị có kích thước màn hình khác nhau, chẳng hạn như màn hình lớn, màn hình nhỏ, hoặc máy in.

CSS là độc lập với HTML và có thể được sử dụng với bất kỳ ngôn ngữ đánh dấu nào xây dựng dựa trên XML. CSS tuân theo chuẩn chung do W3C quy định.

2.1.1.3. JAVASCRIPT

a. Lịch sử

HTML, CSS chỉ là ngôn ngữ đánh dấu và định dạng khá cứng nhắc, không có khả năng lập trình. Vì thế Javascript - một ngôn ngữ "ban đầu" được thiết kế khá sơ sài chỉ chạy trên trình duyệt web ra đời để tương tác với HTML, CSS tạo nên sự ảo diệu và mềm dẻo cho website.

b. Giới thiệu

JS (viết tắt của Javascript) là một nền tảng (cross-platform), ngôn ngữ kịch bản hướng đối tượng (object-oriented). Nó là một ngôn ngữ nhỏ và nhẹ. Chạy trong môi trường máy chủ lưu trữ (ví dụ: trình duyệt web), JavaScript có thể được kết nối với các đối tượng của môi trường để cung cấp kiểm soát chương trình đối với chúng.

JavaScript cho phép bạn thực hiện những điều phức tạp trên các trang web như bản đồ tương tác...

Hiện nay Javascript có thể sử dụng để lập trình cả phía client lẫn server. [1]

2.1.2. Tìm hiểu về Back End

Thế nhưng điều gì giúp phần front-end của một trang web có thể hoạt động được? Tất cả dữ liệu sẽ được lưu trữ ở đâu? Đó là phần việc của back end. Phần back end của một trang web bao gồm một máy chủ, một ứng dụng, và một cơ sở dữ liệu. Một lập trình viên back-end xây dựng và duy trì công nghệ mà sức mạnh của những thành phần đó, cho phép phân giao diện người dùng của trang web có thể tồn tại được.

Để khiến cho máy chủ, ứng dụng, và cơ sở dữ liệu có thể giao tiếp được với nhau, các lập trình viên back-end sử dụng các ngôn ngữ server-side như PHP, Ruby, Python, Java, và .Net để xây dựng một ứng dụng, và các công cụ như MySQL, Oracle, và SQL Server để tìm kiếm, lưu trữ, hoặc thay đổi dữ liệu và phục vụ trở lại tới người dùng trong phần front-end.

2.1.2.1. Ngôn ngữ C# (hay CSharp)

a. C# là gì ?

C# (C Sharp, đọc là "xi-sáp") là một ngôn ngữ lập trình hướng đối tượng đa năng vô cùng mạnh mẽ được phát triển bởi Microsoft, C# là phần khởi đầu cho kế hoạch .NET của họ. Tên của ngôn ngữ bao gồm ký tự thăng theo Microsoft nhưng theo ECMA là C#, chỉ bao gồm dấu số thường. Microsoft phát triển C# dựa trên C++ và Java. C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, Visual Basic, Delphi và Java.

C# được thiết kế chủ yếu bởi Anders Hejlsberg kiến trúc sư phần mềm nổi tiếng với các sản phẩm Turbo Pascal, Delphi, J++, WFC. Phiên bản gần đây nhất là 9.0, được phát hành vào năm 2020 cùng với Visual Studio 2019 phiên bản 16.8.

b. Mục tiêu của việc phát triển C#

- Ngôn ngữ được dự định là một ngôn ngữ lập trình đơn giản, hiện đại, hướng đến nhiều mục đích sử dụng, và là một ngôn ngữ lập trình hướng đối tượng.
- Ngôn ngữ và việc triển khai đáp ứng các nguyên tắc của ngành kỹ thuật phần mềm như kiểm tra chặt chẽ kiểu dữ liệu, kiểm tra giới hạn mảng, phát hiện các trường hợp sử dụng các biến chưa có dữ liệu, và tự động thu gom

rác. Tính mạnh mẽ, sự bền bỉ, và năng suất của việc lập trình là rất quan trọng đối với ngôn ngữ này.

- Ngôn ngữ sẽ được sử dụng để phát triển các thành phần của phần mềm theo hướng thích hợp cho việc triển khai trong các môi trường phân tán.
- Khả năng di chuyển (portability) là rất quan trọng, đặc biệt là đối với những lập trình viên đã quen với C và C++.
- Ngôn ngữ sẽ được thiết kế để phù hợp với việc viết các ứng dụng cho cả hai hệ thống: hosted và nhúng, từ các phần mềm quy mô lớn, đến các phần mềm chỉ có các chức năng đơn giản.
- Mặc dù các ứng dụng C# có tính kinh tế đối với các yêu cầu về bộ nhớ và chế độ xử lý, ngôn ngữ này không cạnh tranh trực tiếp về hiệu năng và kích thước đối với ngôn ngữ C hoặc assembly.

c. C# được sử dụng để làm gì?

- Phát triển web backend (ASP.NET MVC, ASP.NET core, Web API, Graph API, gPRC, Blazor server)
- Phát triển web front end (Blazor WebAssembly)
- Phát triển desktop app (Winform, WPF, UWP, WinUI, Mono, Uno, MAUI, Blazor desktop...)
- Phát triển game (Unity, Monogame, Godot, Stride, CryEngine..)
- Phát triển mobile app, IOS native, Android native (Xamarin, MAUI)
- Phát triển đám mây (Azure, AWS, Google Cloud...)
- Phát triển thực tế ảo (VR) và thực tế tăng cường (AR) (HoloLens, CryEngine, Unity, Oculus quest..)
- Học máy và trí tuệ nhân tạo (ML.Net, TensorFlow, csSharp..)
- Blockchain (NEO, Stratis)
- Microservices and containers
- Internet of thing (IoT, 5G)

d. Đặc trưng của ngôn ngữ C#

- C# là ngôn ngữ đơn giản: C# loại bỏ một vài sự phức tạp và rối rắm của những ngôn ngữ như Java và c++, bao gồm việc loại bỏ những macro,

những template, đa kế thừa, và lớp cơ sở ảo (virtual base class). Ngôn ngữ C# đơn giản vì nó dựa trên nền tảng C và C++. Nếu chúng ta thân thiện với C và C++ hoặc thậm chí là Java, chúng ta sẽ thấy C# khá giống về diện mạo, cú pháp, biểu thức, toán tử và những chức năng khác được lấy trực tiếp từ ngôn ngữ C và C++, nhưng nó đã được cải tiến để làm cho ngôn ngữ đơn giản hơn.

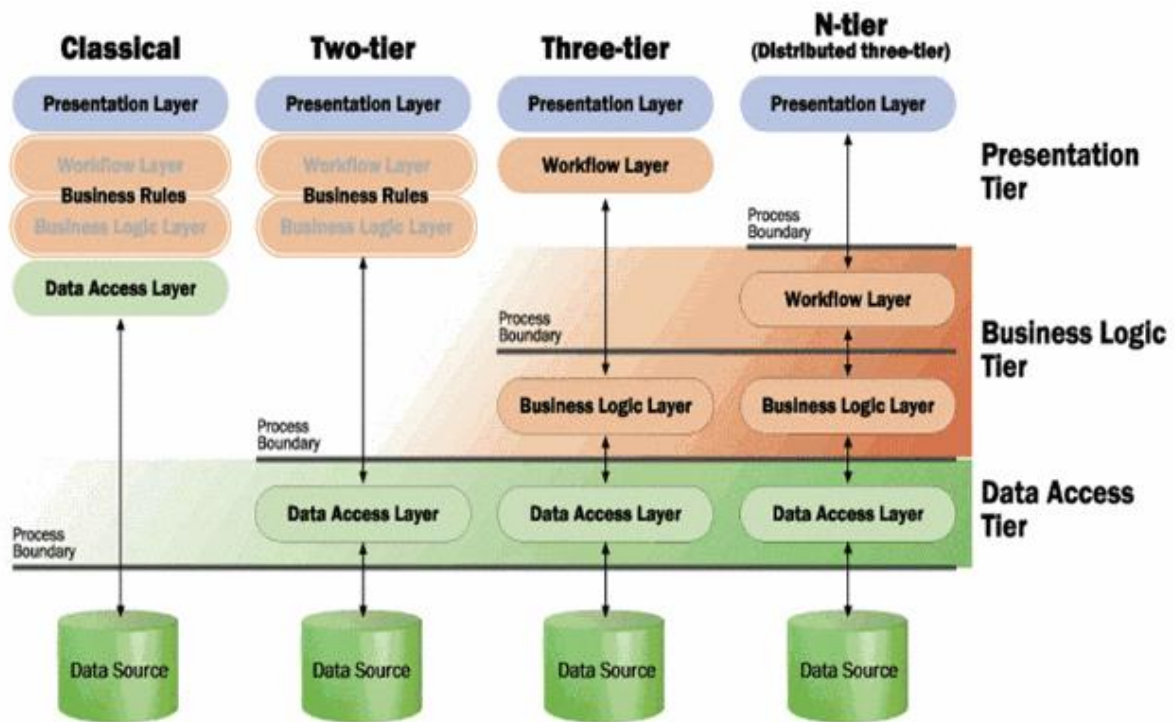
- C# là ngôn ngữ hiện đại: Điều gì làm cho một ngôn ngữ hiện đại? Những đặc tính như là xử lý ngoại lệ, thu gom bộ nhớ tự động, những kiểu dữ liệu mở rộng, và bảo mật mã nguồn là những đặc tính được mong đợi trong một ngôn ngữ hiện đại. C# chứa tất cả những đặc tính trên. Nếu là người mới học lập trình có thể chúng ta sẽ cảm thấy những đặc tính trên phức tạp và khó hiểu. Tuy nhiên, cũng đừng lo lắng chúng ta sẽ dần dần được tìm hiểu những đặc tính qua các nội dung khoá học này.
- C# là ngôn ngữ lập trình thuần hướng đối tượng: Lập trình hướng đối tượng (OOP: Object-oriented programming) là một phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (abstraction), tính đóng gói (encapsulation), tính đa hình (polymorphism) và tính kế thừa (inheritance). C# hỗ trợ cho chúng ta tất cả những đặc tính trên.
- 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. [2]

2.1.2.2. Tổng quan về công nghệ ASP.NET MVC

a. Tổng quan về mô hình MVC

Trong lập trình web chúng ta chia ra 3 tầng như sau:

1. Presentation Layer (tầng 3): tầng này giúp hiển thị giao diện
2. Business Logic Layer (tầng 2): tầng này chứa các hàm thực thi, giúp xử lý sự kiện xảy ra trên tầng Presentation Layer.
3. Data Access Layer (tầng 1): tầng này chứa hàm kết nối với các hệ quản trị cơ sở dữ liệu như SQL Server, Mysql, Oracle ...

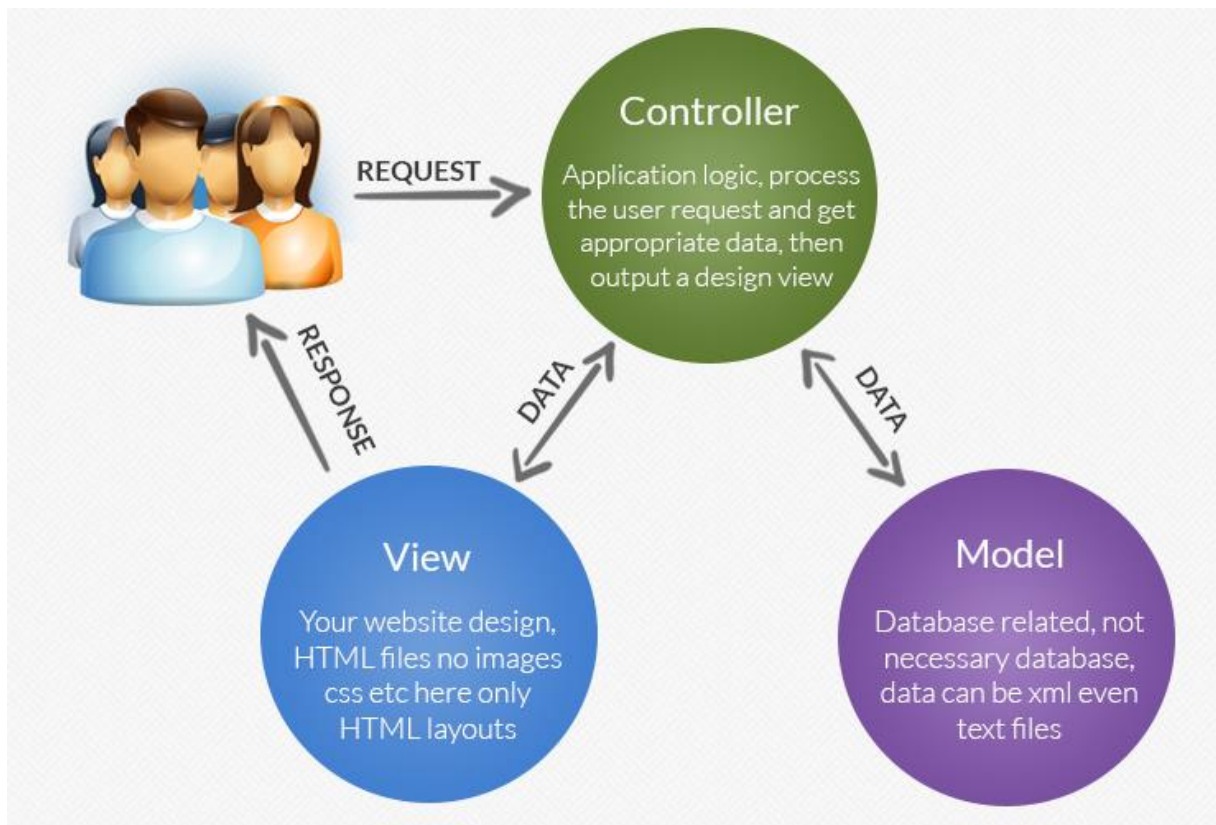


Hình 2.1: N-Tier Diagram.

Theo như hình mô tả, cấu trúc cổ điển là 3 tầng này trộn vào nhau hay còn gọi là cấu trúc One – Tier. Khi làm việc với cấu trúc One – Tier chúng ta gặp khó khăn trong việc chỉnh sửa code hay chỉnh sửa giao diện vì các tầng này bị trộn vào nhau.

Tiếp đó, cấu trúc Two – Tier ra đời, tuy nhiên vẫn chưa giải quyết vấn đề của cấu trúc One – Tier gặp phải đó là việc chỉnh sửa code hay giao diện gặp khó khăn vì tầng 2 và tầng 3 vẫn chưa tách biệt khỏi nhau.

Vì vậy cấu trúc Three – Tier đã ra đời và giải quyết tất cả các vấn đề mà One – Tier, Two – Tier gặp phải. Lúc này 3 tầng đã tách biệt nhau ra do đó việc chỉnh sửa code hay giao diện sẽ trở nên dễ dàng hơn giúp cho ứng dụng tối ưu hóa nhất. Sau khi hiểu và nhớ lại cấu trúc của một mô hình lập trình web cơ bản thì chúng ta sẽ đi qua tìm hiểu về mô hình MVC.



Hình 2.2: Mô hình MVC.

MVC (Model – View - Controller) là một design pattern đã tồn tại rất lâu trong ngành công nghệ phần mềm. Một ứng dụng viết theo mô hình MVC sẽ bao gồm 3 thành phần tách biệt nhau đó là Model, View, Controller. Giống như trong cấu trúc Three – Tier, mô hình MVC giúp tách biệt 3 tầng trong mô hình lập trình web, vì vậy giúp tối ưu ứng dụng, dễ dàng thêm mới và chỉnh sửa code hoặc giao diện

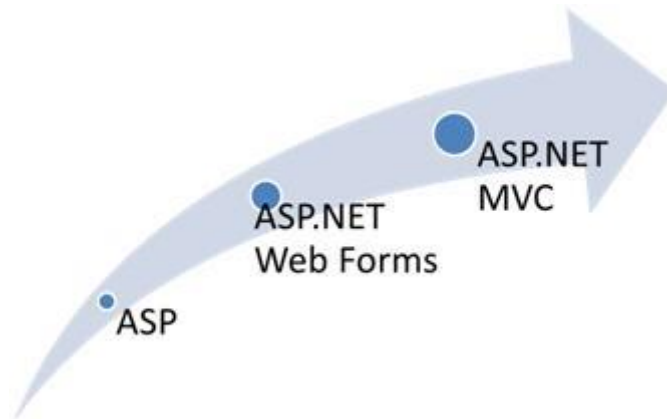
- Model: ở phần trước mình đã nhắc lại cho các bạn về 3 tầng trong mô hình
 - Three – Tier thì trong đó gồm có 2 tầng Data Access Layer và tầng Business Logic Layer. Hai tầng này là hai tầng tương đương với tầng model trong mô hình MVC.
- View: là tầng giao diện, hiển thị dữ liệu được truy xuất từ tầng model. Tầng này tương đương với tầng Presentation Layer trong cấu trúc Three – Tier.
- Controller: đây là tầng giúp kết nối giữa tầng model và tầng view trong mô hình MVC, có nghĩa là nếu phía client yêu cầu hiển thị dữ liệu thì

controller gọi giữ liệu từ model và trả về cho view vì view tương tác trực tiếp với client.

Ví dụ: User yêu cầu hiển thị thông tin cá nhân của user

1. User gửi một yêu cầu tới controller.
2. Controller nhận yêu cầu, xử lý yêu cầu, nếu yêu cầu cần truy xuất dữ liệu thì controller sẽ gửi yêu đó xuống tầng model để truy xuất dữ liệu.
3. Tầng model sẽ lấy dữ liệu từ database sau đó truyền dữ liệu qua tầng view thông qua tầng controller để tầng view hiển thị dữ liệu cho User.
4. User sẽ thấy thông tin hiển thị ở giao diện và cụ thể ở đây là tầng view.

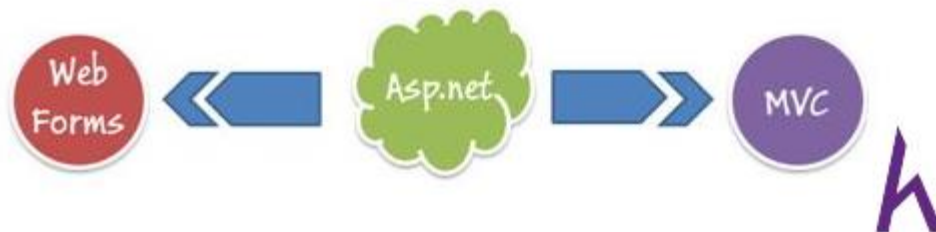
b. Tổng quan về mô hình lập trình web ASP.NET MVC



Hình 2.3: Sự phát triển của ASP.

Hình mô tả phía trên cho các bạn thấy trước khi ASP.Net MVC ra đời thì ASP.Net webform là 1 framework lập trình web khá quen thuộc đối với những tín đồ của ASP.Net. Tuy nhiên, khi ASP.Net MVC xuất hiện thì dường như ASP.Net webform bị lãng quên. Để hiểu được lý do tạo sao lại như vậy thì chúng ta sẽ đi từng bước tìm hiểu nhé!

ASP.Net là gì?



ASP.NET là 1 framework lập trình web được cung cấp bởi Microsoft và nó sử dụng ngôn ngữ C# làm ngôn ngữ phát triển. ASP.NET bao gồm 2 model đó là Web Forms và MVC.

Nhược điểm khi lập trình với ASP.Net web forms:

Trong nội dung bài ngày hôm nay, mình tập trung giới thiệu ASP.Net MVC nên mình sẽ không đi vào cụ thể về Web Forms mà mình chỉ đưa ra 1 vài nhược điểm của Web Forms.

- Web Forms không có sự tách biệt rõ ràng 3 tầng trong mô hình lập trình web.
- Trong Web Forms có sử dụng 1 đối tượng View State mà đối tượng này làm cho trang web bị tăng kích thước vì vậy làm giảm hiệu năng của ứng dụng.
- Giao diện của Web Forms được thiết kế dựa trên những toolbox có sẵn mà web forms cung cấp mà hiện nay đa phần designer sử dụng html, css để thiết kế website vì nó gây khó khăn trong việc thiết kế web hiện nay, việc thiết kế trở nên mất linh hoạt.

Với những khó khăn như vậy thì Microsoft đã cho ra đời ASP.Net MVC.

Cơ chế hoạt động và những ưu điểm trong lập trình web với ASP.Net MVC

ASP.Net MVC là 1 framework lập trình web mới của microsoft, công nghệ này ứng dụng mô hình MVC vào trong ASP.Net, còn mô hình MVC là như thế nào thì mình đã nói rõ ở phần phía trên rồi nhé

Stable Release	.NET Framework	Included in Visual Studio IDE	Year
ASP.NET MVC 1.0	.Net Framework 3.5	Visual Studio 2008, 2008 SP1	13-Mar-2009
ASP.NET MVC 2	.Net Framework 3.5, 4.0	Visual Studio 2008, 2010	10-Mar-2010
ASP.NET MVC 3	.Net Framework 4.0	Visual Studio 2010	13-Jan-2011
ASP.NET MVC 4	.Net Framework 4.0, 4.5	Visual Studio 2010 SP1, 2012	15-Aug-2012
ASP.NET MVC 5	.Net Framework 4.5, 4.5.1	Visual Studio 2013, Community	17-Oct-2013

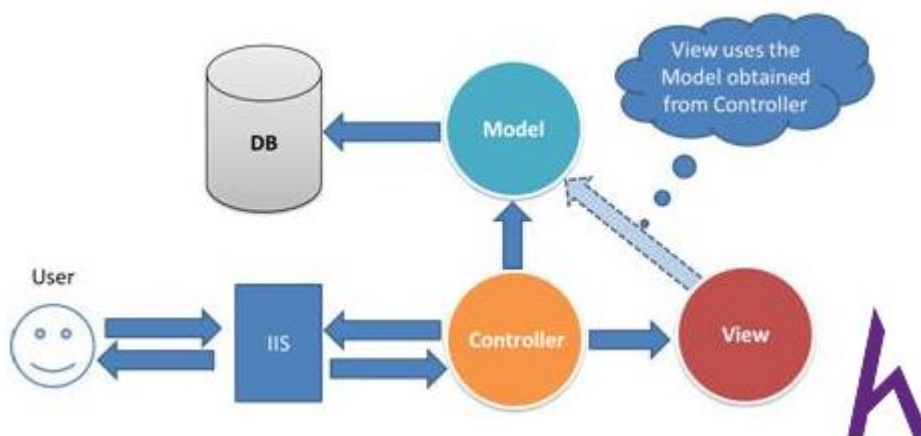
Hình 2.4: Lịch sử phát triển của ASP.NET MVC.

Qua hình mô tả phía trên cho các bạn thấy tuy ASP.Net MVC ra đời năm 2009 nhưng mà tính tới năm 2013 thì nó đã được update lên tới phiên bản ASP.Net MVC 5, đây cũng là 1 trong những ưu điểm của ông lớn Microsoft cập nhật công nghệ liên tục.

Và mới đây thì ASP.Net cũng đã cho ra đời 1 công nghệ mới đó là ASP.Net core 1.0

Với sự cập nhật thường xuyên như vậy đã giúp cho ASP.Net MVC ngày nay phát triển mạnh mẽ không thua kém gì các công nghệ lập trình web sử dụng Java và Php.

Cơ chế hoạt động của ASP.Net MVC



Hình 2.5: Cơ chế hoạt động của ASP.NET MVC.

Dựa vào hình mô phỏng phía trên này thì mình sẽ khái quát sơ qua cơ chế hoạt động của ASP.Net MVC

1. User gửi 1 yêu cầu tới server bằng cách truyền vào 1 URL trong browser
2. Yêu cầu đó được gửi tới controller đầu tiên, controller sẽ xử lý yêu cầu, nếu yêu cầu cần truy xuất dữ liệu thì controller sẽ chuyển qua tầng model
3. Tại tầng model, dữ liệu được truy xuất từ database và sau đó truyền qua view thông qua controller
4. Controller sẽ giúp dữ liệu được chuyển từ model qua view
5. View là tầng cuối cùng giao tiếp với User, mọi dữ liệu sẽ được hiển thị cho User thông qua tầng View

Những ưu điểm mà ASP.Net MVC mang lại cho chúng ta:

1. Do sử dụng mô hình MVC nên trong ASP.Net MVC đã tách biệt được các tầng trong mô hình lập trình web vì vậy giúp tối ưu ứng dụng và dễ dàng trong việc viết code, giao diện

2. Giao diện trong ASP.Net MVC sử dụng công nghệ thiết kế web HTML, CSS nên việc thiết kế giao diện trở nên dễ dàng và giúp cho designer linh hoạt trong việc thiết kế
3. ASP.Net MVC không sử dụng view state vì vậy trang web không bị tăng kích thước do đó hiệu năng hoạt động không bị giảm

ASP.Net MVC đã khắc phục được các nhược điểm của web forms vì vậy web forms hiện nay không còn được dùng phổ biến nữa. [3]

2.1.2.3. Tổng quan về Entity Framework

Entity Framework là gì?

Entity Framework ra đời nhằm hỗ trợ sự tương tác giữa các ứng dụng trên nền tảng .NET với các cơ sở dữ liệu quan hệ. Hay, Entity Framework chính là công cụ giúp ánh xạ giữa các đối tượng trong ứng dụng, phần mềm của bạn với các bảng của một cơ sở dữ liệu quan hệ.

Hay một định nghĩa khác về Entity Framework:

Entity Framework là một khung ORM(Object Relational Mapper) mã nguồn mở cho các ứng dụng .NET được Microsoft hỗ trợ. là 1 phần của .NET Framework. là một Open source ORM Framework.

Entity Framework giúp các nhà phát triển Web tương tác với dữ liệu quan hệ theo phương pháp hướng đối tượng với ít mã hơn so với các ứng dụng truyền thống. Lợi ích lớn nhất của nó là giúp lập trình viên giảm thiểu việc lập trình mã nguồn để thực hiện truy cập và tương tác với cơ sở dữ liệu.

Hiện nay, Entity framework là 1 framework mạnh để phát triển ứng dụng Web với sự hỗ trợ đông đảo của cộng đồng.

Tại sao lại sử dụng Entity Framework?

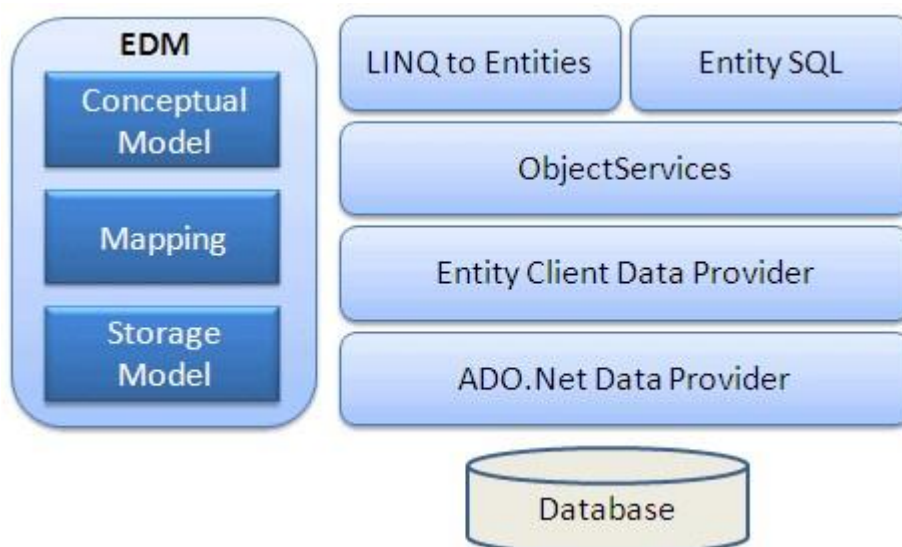
Sử dụng Entity Framework sẽ đem lại những lợi ích sau:

- Nếu bạn thực hiện truy vấn dữ liệu. Bạn có thể sử dụng Linq to Entities để thao tác với objects được sinh ra từ Entity Framework, nghĩa là bạn không phải viết code sql.
- Việc update các classes, commands dễ dàng mỗi khi cơ sở dữ liệu có sự thay đổi, điều này giúp bạn tiết kiệm thời gian đáng kể.

- Entity Framework sẽ tự động tạo ra các classes, commands tương ứng cho việc select, insert, update, delete dữ liệu từ cơ sở dữ liệu quan hệ.
- Entity Framework tự động tạo ra các classes cho việc truy suất cơ sở dữ liệu giúp lập trình viên giảm được thời gian viết code thao tác với database. Hỗ trợ bạn không phải mất quá nhiều thời gian cho việc viết code để thao tác với database.

Cấu trúc của Entity Framework

Cấu trúc được mô tả đơn giản qua hình sau:



Hình 2.6: Cấu trúc của Entity Framework.

EDM (Entity Data Model): bao gồm 3 phần :

- Conceptual Model: Chứa các model class và các quan hệ của nó. Nó độc lập với thiết kế bảng CSDL của bạn.
- Mapping: gồm có thông tin về cách Conceptual model kết nối với Storage model.
- Storage Model: Đây là database design model bao gồm các bảng, views, stored procedures, và những quan hệ và các khóa.

LINQ to Entities:

Là ngôn ngữ truy vấn sử dụng để viết các truy vấn tới object model. Trả về các thực thể được định nghĩa bên trong Conceptual model.

Entity SQL:

Đây là một ngôn ngữ truy vấn khác, nó giống LINQ to Entities.

Object Service:

Object service sẽ có trách nhiệm trong việc cụ thể hóa quá trình chuyển đổi dữ liệu trả về từ một entity client data provider tới một entity object structure.

Entity Client Data Provider:

Nhiệm vụ chính của nó là chuyển đổi L2E hoặc những truy vấn Entity SQL vào một truy vấn SQL. Nó được hiểu bởi CSDL cơ bản. Giao tiếp với ADO.Net data provider lần lượt gửi và nhận dữ liệu từ CSDL.

ADO.Net Data Provider:

Thực hiện giao tiếp với CSDL bằng sử dụng chuẩn ADO.Net.

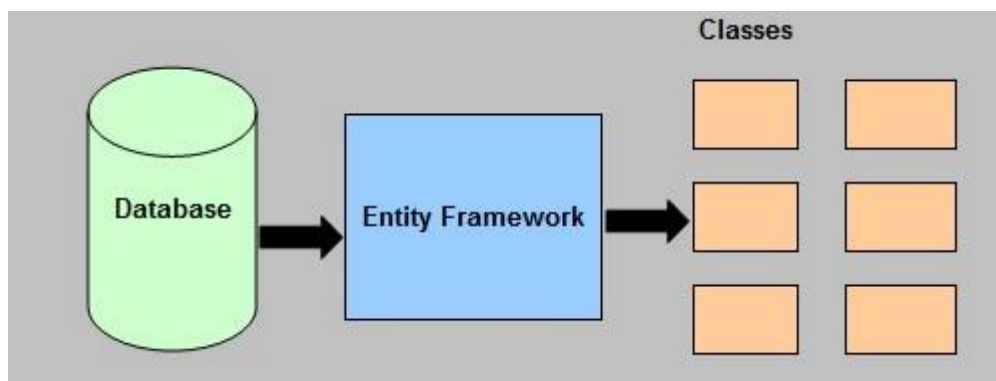
Những tính năng của Entity Framework

- Entity framework hỗ trợ stored procedure.
- Nó làm việc với bất kỳ cơ sở dữ liệu quan hệ nào có Entity Framework hợp lệ.
- Entity Framework tạo truy vấn SQL từ LINQ to Entities.
- Nó được phát triển như một sản phẩm mã nguồn mở.
- Entity framework là một sản phẩm của Microsoft.
- EF cho phép tạo những câu lệnh thêm, xóa, cập nhật.
- Theo dõi những thay đổi của các đối tượng trong bộ nhớ.

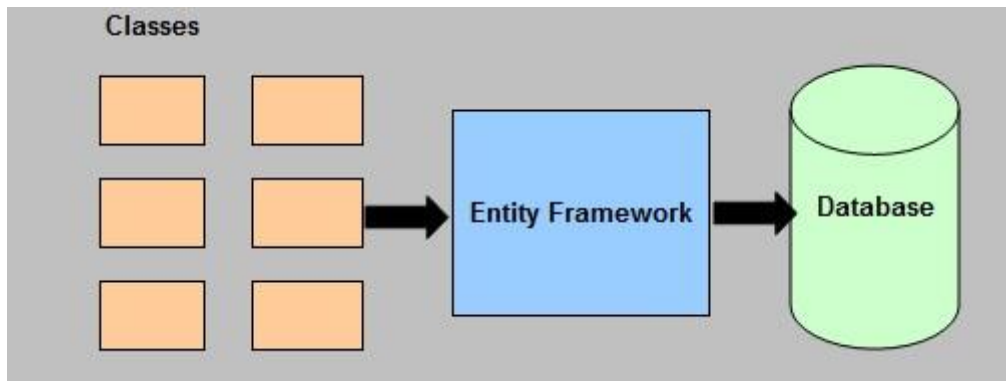
Các tình huống dùng Entity Framework

Entity framework là rất hữu ích trong 3 tình huống sau:

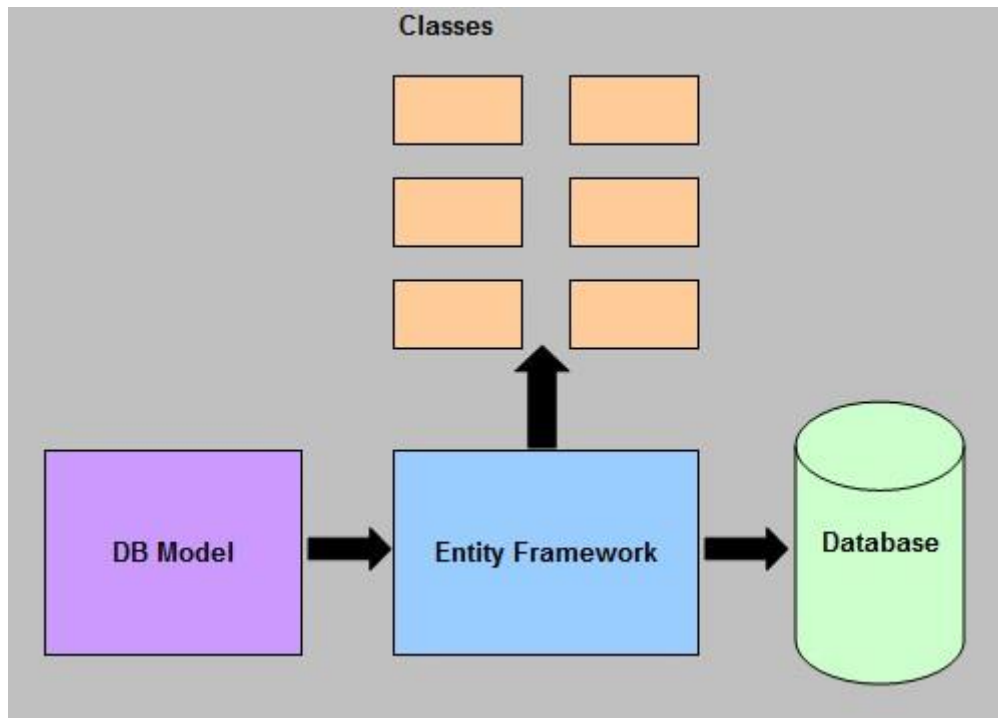
Database First: Đầu tiên là nếu bạn đã có một CSDL hoặc muốn thiết kế CSDL trước khi làm các phần khác của ứng dụng.



Code First: Thứ hai là bạn muốn tập trung vào các domain class rồi mới tạo CSDL từ các domain class đó.



Model First: Thứ ba là bạn muốn thiết kế schema của CSDL trên visual designer rồi mới tạo CSDL và các class. [4]



2.1.3. Tìm hiểu về hệ quản trị cơ sở dữ liệu MSSQL Server

2.1.3.1. SQL là gì?

SQL là ngôn ngữ phi thủ tục, không yêu cầu cách thức truy cập cơ sở dữ liệu như thế nào. Tất cả các thông báo của SQL rất dễ dàng sử dụng và ít mắc lỗi.

SQL cung cấp các tập lệnh phong phú cho các công việc hỏi đáp dữ liệu như:

- Chèn, xóa và cập nhật các hàng trong 1 quan hệ
- Tập, thêm, xóa và sửa đổi các đối tượng trong của cơ sở dữ liệu.

- Điều khiển việc truy cập tới cơ sở dữ liệu và các đối tượng của cơ sở dữ liệu để đảm bảo tính bảo mật, tính nhất quán và sự ràng buộc của cơ sở dữ liệu.

Đối tượng của SQL Server là các bảng dữ liệu với các cột và các hàng. Cột được gọi là trường dữ liệu và hàng là bản ghi của bảng. Cột dữ liệu và kiểu dữ liệu xác định tạo nên cấu trúc của bảng. Khi bảng được tổ chức thành một hệ thống cho một mục đích sử dụng cụ thể vào công việc nào đó sẽ trở thành một cơ sở dữ liệu.

2.1.3.2. Microsoft SQL Server là gì?

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 phát triển và tiếp thị bởi Microsoft.

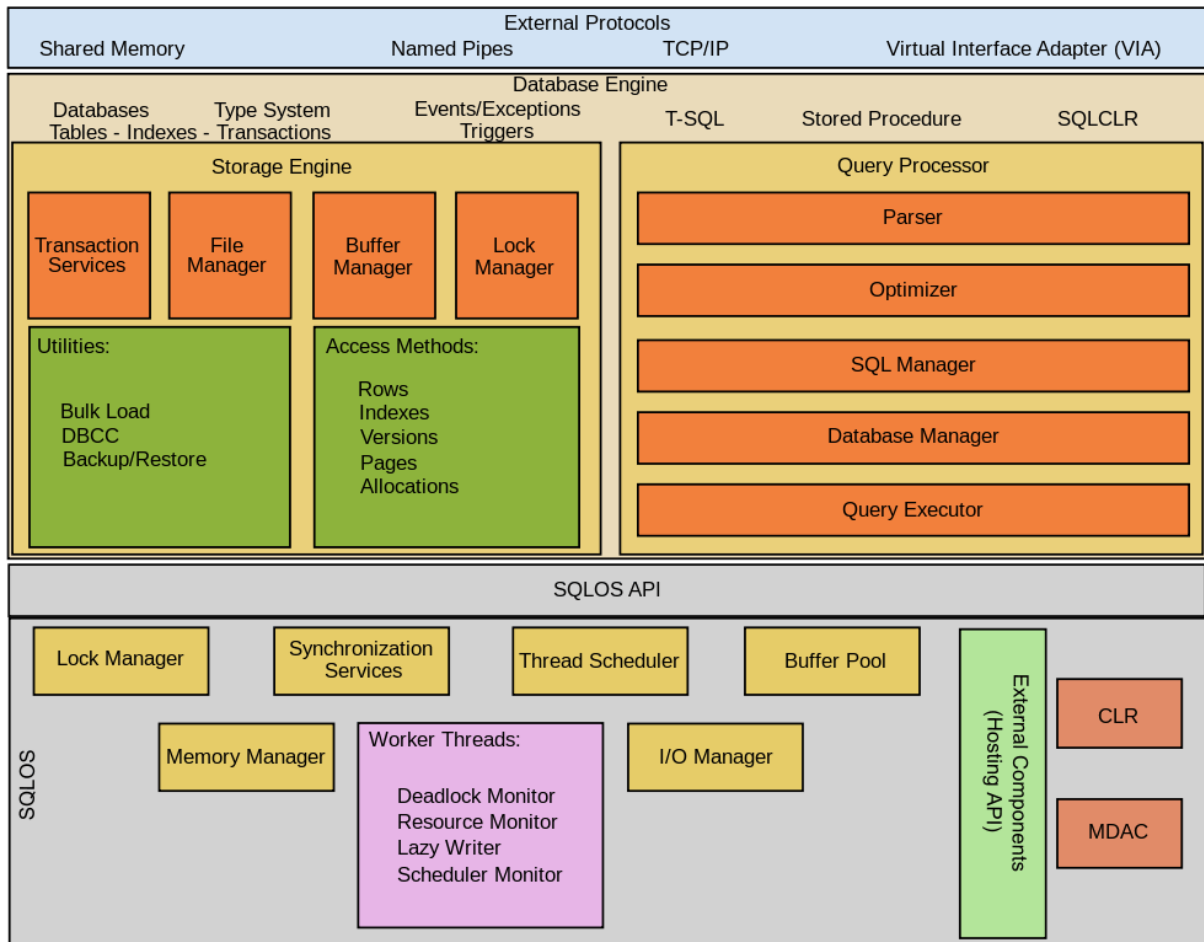
(Nói dễ hiểu là – Tương tự như phần mềm RDBMS khác, SQL Server được xây dựng dựa trên SQL, một ngôn ngữ lập trình tiêu chuẩn để tương tác với các cơ sở dữ liệu quan hệ. Máy chủ SQL được liên kết với Transact-SQL hoặc T-SQL, triển khai SQL Microsoft Microsoft bổ sung một tập hợp các cấu trúc lập trình độc quyền).

SQL Server hoạt động độc quyền trên môi trường Windows trong hơn 20 năm. Năm 2016, Microsoft đã cung cấp phiên bản trên Linux. SQL Server 2017 ra mắt vào tháng 10 năm 2016 chạy trên cả Windows và Linux, SQL Server 2019 sẽ ra mắt trong năm 2019.

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....

Kiến trúc của SQL Server

Sơ đồ sau minh họa kiến trúc của SQL Server:



Hình 2.7: Kiến trúc của SQL Server.

SQL Server bao gồm hai thành phần chính:

- Database Engine
- SQLOS

SQL Server Database Engine, công cụ này kiểm soát việc lưu trữ, xử lý và bảo mật dữ liệu. Thành phần này bao gồm một công cụ quan hệ có chức năng xử lý các lệnh và truy vấn, một công cụ lưu trữ quản lý các tệp, bảng, trang, chỉ mục, bộ đệm dữ liệu và giao dịch cơ sở dữ liệu. Các nhiệm vụ, trigger, trình xem và các đối tượng dữ liệu lưu trữ khác cũng được Database Engine khởi tạo và xử lý.

Lớp phía dưới Database Engine là Hệ điều hành SQL Server – viết tắt SQLOS. Hệ điều hành xử lý các chức năng ở cấp độ thấp hơn như quản lý bộ nhớ và I/O, lên lịch nhiệm vụ và khóa dữ liệu để tránh các xung đột xảy ra khi update. Một lớp giao diện mạng nằm trên lớp Database Engine và sử dụng một giao thức gọi là *Tabular Data Stream* của Microsoft để các yêu cầu và phản hồi tương tác với máy chủ cơ sở dữ liệu thuận tiện hơn. Ở cấp độ user, SQL Server DBAs và

developers viết các câu lệnh T-SQL để xây dựng và sửa đổi cấu trúc cơ sở dữ liệu, thao tác, thiết lập các bảo vệ, sao lưu cơ sở dữ liệu, cùng với nhiều nhiệm vụ khác.

Công cụ và dịch vụ của SQL Server

Microsoft cung cấp cả công cụ và dịch vụ quản lý dữ liệu và kinh doanh (BI) cùng với SQL Server.

Tính năng doanh nghiệp chuyên biệt

Microsoft hiện nay đã có thêm các tính năng quản lý dữ liệu đa dạng, tính năng dành cho doanh nghiệp (business intelligence – BI), và các công cụ phân tích SQL Server. Bên cạnh các dịch vụ Machine Learning mới được tích hợp lần đầu tiên trong phiên bản SQL Server 2016, các dịch vụ phân tích dữ liệu bao gồm SQL Server Analysis Services, công cụ phân tích xử lý dữ liệu sử dụng trong BI, các ứng dụng trực quan hóa dữ liệu và các dịch vụ SQL Server Reporting, hỗ trợ tạo và phân phối các báo cáo BI.

Tính năng quản lý

Về phương diện quản trị, Microsoft SQL Server gồm có các dịch vụ tích hợp SQL Server, dịch vụ SQL Server Data Quality và dịch vụ SQL Server Master. Hai bộ công cụ dành riêng cho DBAs và developer: SQL Server Data Tools, sử dụng trong việc phát triển cơ sở dữ liệu và SQL Server Management Studio, nhằm mục đích triển khai, giám sát và quản lý cơ sở dữ liệu.

Các ấn bản SQL Server

- Enterprise : chứa tất cả các đặc điểm nổi bật của SQL Server, bao gồm nhân bộ máy cơ sở dữ liệu và các dịch vụ đi kèm cùng với các công cụ cho tạo và quản lý phân cụm SQL Server. Nó có thể quản lý các CSDL lớn tới 524 petabytes và đánh địa chỉ 12 terabytes bộ nhớ và hỗ trợ tới 640 bộ vi xử lý(các core của cpu)
- Standard : Rất thích hợp cho các công ty vừa và nhỏ vì giá thành rẻ hơn nhiều so với Enterprise Edition, nhưng lại bị giới hạn một số chức năng cao cấp (advanced features) khác, edition này có thể chạy tốt trên hệ thống lên đến 4 CPU và 2 GB RAM.
- Developer : Có đầy đủ các tính năng của Enterprise Edition nhưng được chế tạo đặc biệt như giới hạn số lượng người kết nối vào Server cùng một lúc.... Đây là phiên bản sử dụng cho phát triển và kiểm tra ứng dụng. Phiên

bản này phù hợp cho các cá nhân, tổ chức xây dựng và kiểm tra ứng dụng. MIỄN PHÍ

- Workgroup: ấn bản SQL Server Workgroup bao gồm chức năng lõi cơ sở dữ liệu nhưng không có các dịch vụ đi kèm. Chú ý phiên bản này không còn tồn tại ở SQL Server 2012.
- Express : SQL Server Express dễ sử dụng và quản trị cơ sở dữ liệu đơn giản. Được tích hợp với Microsoft Visual Studio, nên dễ dàng để phát triển các ứng dụng dữ liệu, an toàn trong lưu trữ, và nhanh chóng triển khai. SQL Server Express là phiên bản miễn phí, không giới hạn về số cơ ở dữ liệu hoặc người sử dụng, nhưng nó chỉ dùng cho 1 bộ vi xử lý với 1 GB bộ nhớ và 10 GB file cơ sở dữ liệu. SQL Server Express là lựa chọn tốt cho những người dùng chỉ cần một phiên bản SQL Server 2005 nhỏ gọn, dùng trên máy chủ có cấu hình thấp, những nhà phát triển ứng dụng không chuyên hay những người yêu thích xây dựng các ứng dụng nhỏ. MIỄN PHÍ.

Mục đích sử dụng SQL Server

- Tạo cơ sở dữ liệu.
- Duy trì cơ sở dữ liệu.
- Phân tích dữ liệu bằng SSAS – SQL Server Analysis Services.
- Tạo báo cáo bằng SSRS – SQL Server Reporting Services.
- Thực hiện quá trình ETL (Extract-Transform-Load) bằng SSIS – SQL Server Integration Services. [5]

2.1.4. Tìm hiểu về mô hình hóa UML

Ngôn ngữ mô hình hóa thống nhất (tiếng Anh: Unified Modeling Language, viết tắt thành UML) là một ngôn ngữ mô hình gồm các ký hiệu đồ họa mà các phương pháp hướng đối tượng sử dụng để thiết kế các hệ thống thông tin một cách nhanh chóng.

Cách xây dựng các mô hình trong UML phù hợp mô tả các hệ thống thông tin cả về cấu trúc cũng như hoạt động. Cách tiếp cận theo mô hình của UML giúp ích rất nhiều cho những người thiết kế và thực hiện hệ thống thông tin cũng như những người sử dụng nó; tạo nên một cái nhìn bao quát và đầy đủ về hệ thống thông tin dự định xây dựng. Cách nhìn bao quát này giúp nắm bắt trọn vẹn các yêu cầu của người dùng; phục vụ từ giai đoạn phân tích đến việc thiết kế, thẩm

định và kiểm tra sản phẩm ứng dụng công nghệ thông tin. Các mô hình hướng đối tượng được lập cũng là cơ sở cho việc ứng dụng các chương trình tự động sinh mã trong các ngôn ngữ lập trình hướng đối tượng, chẳng hạn như ngôn ngữ C++, Java,... Phương pháp mô hình này rất hữu dụng trong lập trình hướng đối tượng. Các mô hình được sử dụng bao gồm Mô hình đối tượng (mô hình tĩnh) và Mô hình động.

UML sử dụng một hệ thống ký hiệu thống nhất biểu diễn các Phần tử mô hình (model elements). Tập hợp các phần tử mô hình tạo thành các Sơ đồ UML (UML diagrams). Có các loại sơ đồ UML chủ yếu sau:

- Sơ đồ lớp (Class Diagram)
- Sơ đồ đối tượng (Object Diagram)
- Sơ đồ tình huống sử dụng (Use Cases Diagram)
- Sơ đồ trình tự (Sequence Diagram)
- Sơ đồ cộng tác (Collaboration Diagram hay là Composite Structure Diagram)
- Sơ đồ trạng thái (State Machine Diagram)
- Sơ đồ thành phần (Component Diagram)
- Sơ đồ hoạt động (Activity Diagram)
- Sơ đồ triển khai (Deployment Diagram)
- Sơ đồ gói (Package Diagram)
- Sơ đồ liên lạc (Communication Diagram)
- Sơ đồ tương tác (Interaction Overview Diagram - UML 2.0)
- Sơ đồ phối hợp thời gian (Timing Diagram - UML 2.0) [6]

2.2. Công cụ sử dụng

2.2.1. MSSQL Server

2.2.1.1. Giới thiệu

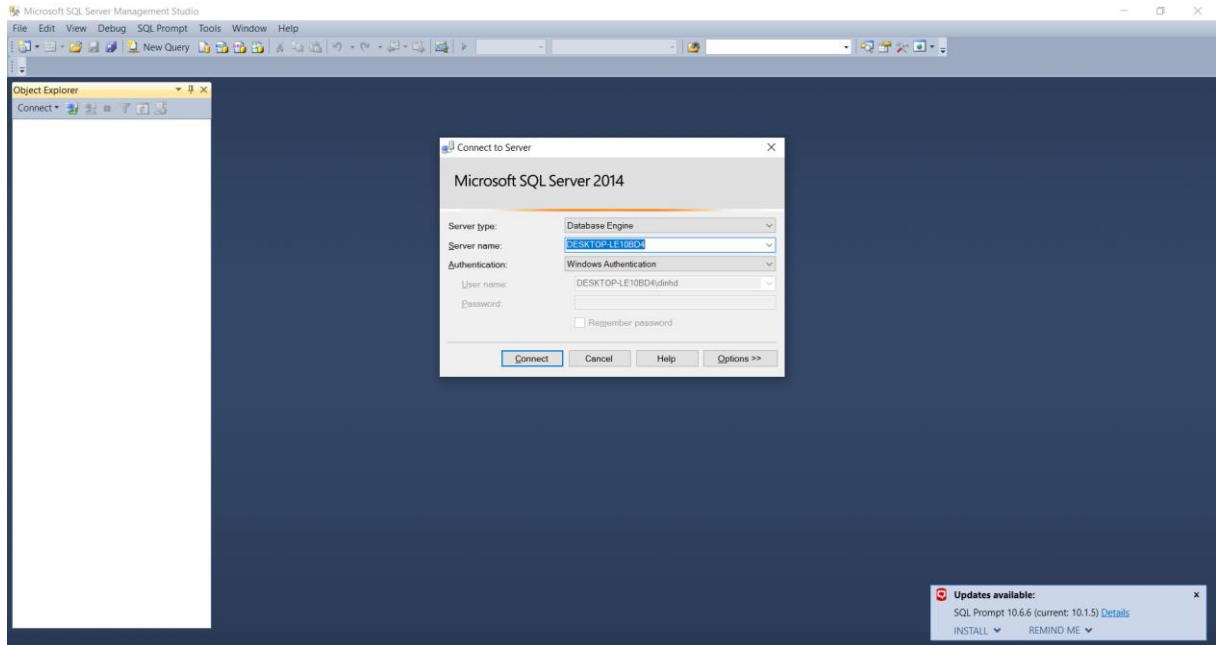
MSSQL Server có 2 phiên bản:

- Express (bản miễn phí)
- Developer (bản trả phí)

2.2.1.2. Cài đặt

Download MSSQL Server tại: <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>.

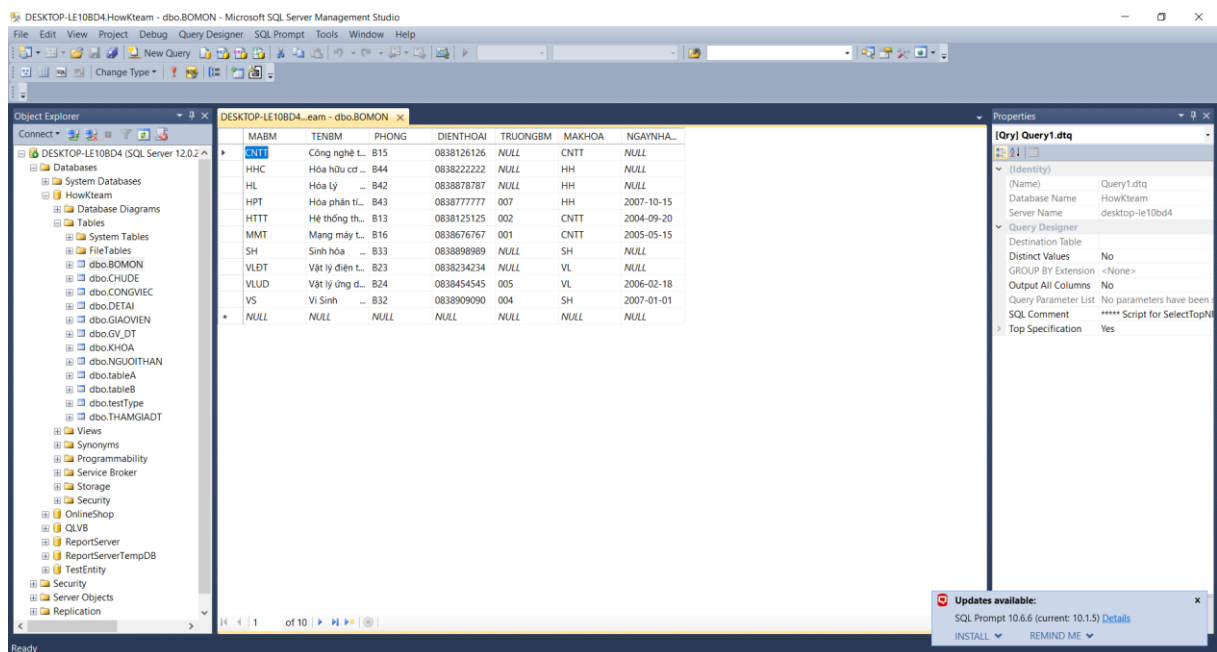
2.2.1.3. Một số hình ảnh của MSSQL 2014



Hình 2.8: Giao diện khởi động của SSMS 2014.

SSMS (SQL Server Management Studio) là một ứng dụng phần mềm thiết kế bởi Microsoft, ra mắt lần đầu năm 2005. Ứng dụng này cho phép lập trình viên cấu hình, quản lý và quản trị bộ máy cơ sở dữ liệu (database engine) SQL Server. SSMS phổ biến và được sử dụng rộng rãi trong cộng đồng lập trình viên và quản trị viên cơ sở dữ liệu bởi những ưu điểm

- Miễn phí
- Trải nghiệm người dùng tốt
- Nhiều lựa chọn add-in
- Dễ cài đặt



Hình 2.9: Giao diện làm việc chính của SSML 2014.

SSMS cho phép lập trình viên tạo và triển khai các truy vấn T-SQL.

Khi nhấp vào nút New Query trên thanh công cụ, trình chỉnh sửa truy vấn sẽ xuất hiện. Tại đây, các lập trình viên sẽ nhập các truy vấn cần thiết, nhấn nút Execute để chạy những truy vấn này. Kết quả truy vấn sẽ được hiển thị bên dưới bảng điều khiển Results.

Một số tính năng của thanh công cụ trong SQL Editor

- Available Database: Trong drop-down menu này, bạn có thể lựa chọn truy vấn được triển khai trên cơ sở dữ liệu nào.
- Execute: Bắt đầu chạy một cửa sổ truy vấn đang hoạt động gần đây.
- Cancel Executing Query: Chấm dứt chạy truy vấn.
- Parse: Phê chuẩn cú pháp của lệnh truy vấn, tuy nhiên không kiểm tra phần cơ sở dữ liệu.
- Changing query result destination: Chọn nơi hiển thị kết quả truy vấn.
- Comment/ Uncomment out the Selected Lines: bình luận hoặc bỏ bình luận về một hoặc một vài dòng code được chọn.

2.2.2. Visual Studio

2.2.2.1. Giới thiệu

Visual studio là một trong những công cụ hỗ trợ lập trình website rất nổi tiếng nhất hiện nay của Microsoft và chưa có một phần mềm nào có thể thay thế được nó. Visual Studio được viết bằng 2 ngôn ngữ đó chính là C# và VB+. Đây là 2 ngôn ngữ lập trình giúp người dùng có thể lập trình được hệ thống một cách dễ dàng và nhanh chóng nhất thông qua Visual Studio.

Visual Studio là một phần mềm lập trình hệ thống được sản xuất trực tiếp từ Microsoft. Từ khi ra đời đến nay, Visual Studio đã có rất nhiều các phiên bản sử dụng khác nhau. Điều đó, giúp cho người dùng có thể lựa chọn được phiên bản tương thích với dòng máy của mình cũng như cấu hình sử dụng phù hợp nhất.

Bên cạnh đó, Visual Studio còn cho phép người dùng có thể tự chọn lựa giao diện chính cho máy của mình tùy thuộc vào nhu cầu sử dụng.

Một số tính năng của phần mềm Visual Studio

- **Biên tập mã**

Giống như bất kỳ một IDE khác, Visual Studio gồm có một trình soạn thảo mã hỗ trợ tô sáng cú pháp và hoàn thiện mã bằng các sử dụng IntelliSense không chỉ cho các hàm, biến và các phương pháp mà còn sử dụng cho các cấu trúc ngôn ngữ như: Truy vấn hoặc vòng điều khiển.

Bên cạnh đó, các trình biên tập mã Visual Studio cũng hỗ trợ cài đặt dấu trang trong mã để có thể điều hướng một cách nhanh chóng và dễ dàng. Hỗ trợ các điều hướng như: Thu hẹp các khối mã lệnh, tìm kiếm gia tăng,...

Visual Studio còn có tính năng biên dịch nền tức là khi mã đang được viết thì phần mềm này sẽ biên dịch nó trong nền để nhằm cung cấp thông tin phản hồi về cú pháp cũng như biên dịch lỗi và được đánh dấu bằng các gạch gợn sóng màu đỏ.

Phiên bản Visual Studio 2015 đang được sử dụng phổ biến nhất hiện nay.

- **Trình gỡ lỗi**

Visual Studio có một trình gỡ lỗi có tính năng vừa lập trình gỡ lỗi cấp máy và gỡ lỗi cấp mã nguồn. Tính năng này hoạt động với cả hai mã quản lý giống như ngôn ngữ máy và có thể sử dụng để gỡ lỗi các ứng dụng được viết bằng các ngôn ngữ được hỗ trợ bởi Visual Studio.

- **Thiết kế**

Windows Forms Designer

Được sử dụng với mục đích xây dựng GUI sử dụng Windows Forms, được bố trí dùng để xây dựng các nút điều khiển bên trong hoặc cũng có thể khóa chúng vào bên cạnh mẫu. Điều khiển trình bày dữ liệu có thể được liên kết với các nguồn dữ liệu như: Cơ sở dữ liệu hoặc truy vấn.

WPF Designer

Tính năng này cũng giống như Windows Forms Designer có công dụng hỗ trợ kéo và thả ảnh dụ. Sử dụng tương tác giữa người và máy tính nhằm mục tiêu vào Windows Presentation Foundation.

Web designer/development

Visual Studio cũng có một trình soạn thảo và thiết kế website cho phép các trang web được thiết kế theo tính năng kéo và thả đối tượng. Mục đích là để hỗ trợ người dùng tạo trang web dễ dàng hơn, những yêu cầu đơn giản như thiết kế web du lịch hay các trang giới thiệu của công ty có thể sử dụng tính năng này vì nó vẫn đảm bảo cho bạn sở hữu được một website hoàn chỉnh.

Vì sao nên sử dụng Visual Studio?

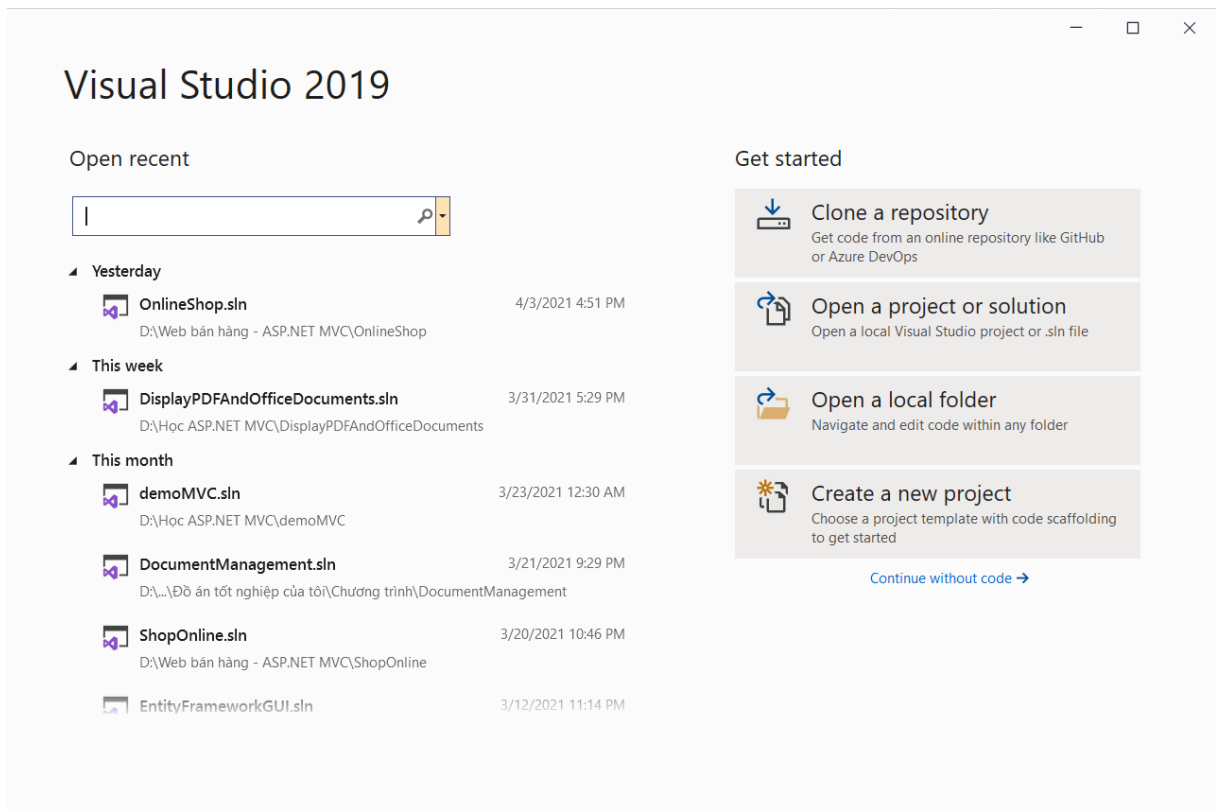
Để trả lời cho câu hỏi này, các bạn có thể tìm hiểu những điểm mạnh của Visual Studio:

- Visual Studio hỗ trợ lập trình trên nhiều ngôn ngữ như: C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript.
- Là một công cụ hỗ trợ việc Debug một cách dễ dàng và mạnh mẽ như: Break Point, xem giá trị của biến trong quá trình chạy, hỗ trợ debug từng câu lệnh.
- Giao diện Visual Studio rất dễ sử dụng đối với người mới bắt đầu lập trình.
- Visual Studio hỗ trợ phát triển các ứng dụng: desktop MFC, Windows Form, Universal App, ứng dụng mobile Windows Phone 8/8.1, Windows 10, ...
- Visual Studio hỗ trợ xây dựng ứng dụng một cách chuyên nghiệp bằng các công cụ kéo thả.
- Visual Studio được đông đảo lập trình viên trên thế giới sử dụng. [7]

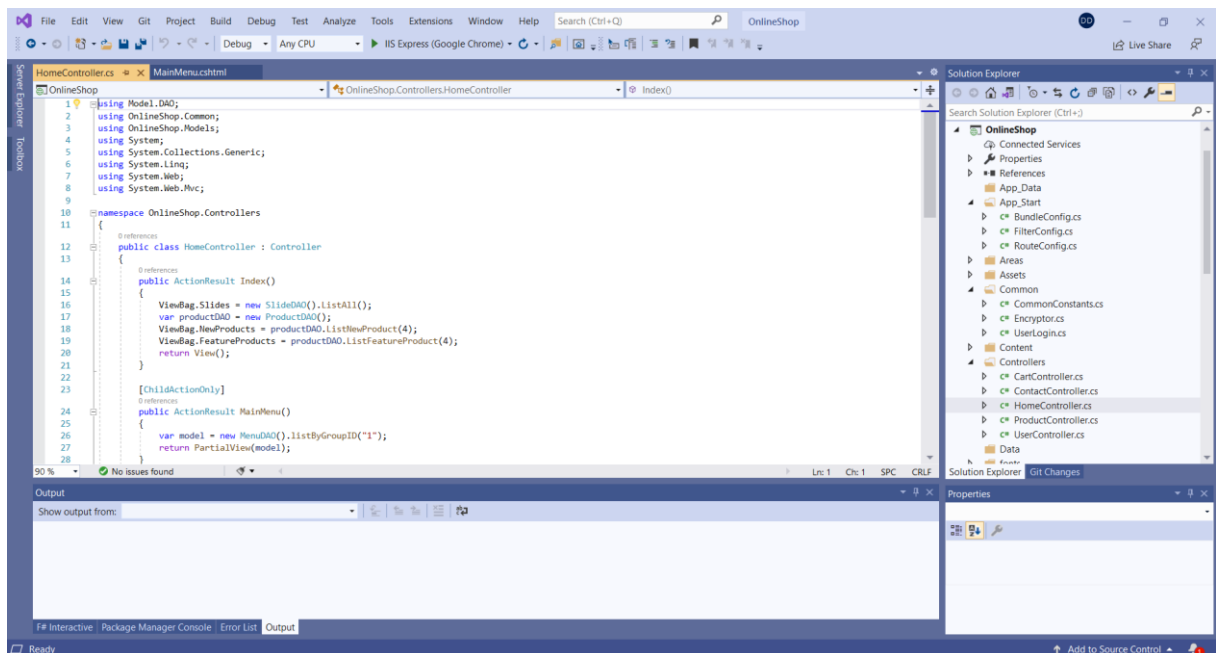
2.2.2.2. Cài đặt

Download Visual Studio 2019 tại: <https://visualstudio.microsoft.com/downloads/>

2.2.2.3. Một số hình ảnh của Visual Studio 2019



Hình 2.10: Giao diện khởi động của Visual Studio 2019.



2.2.3. StarUML

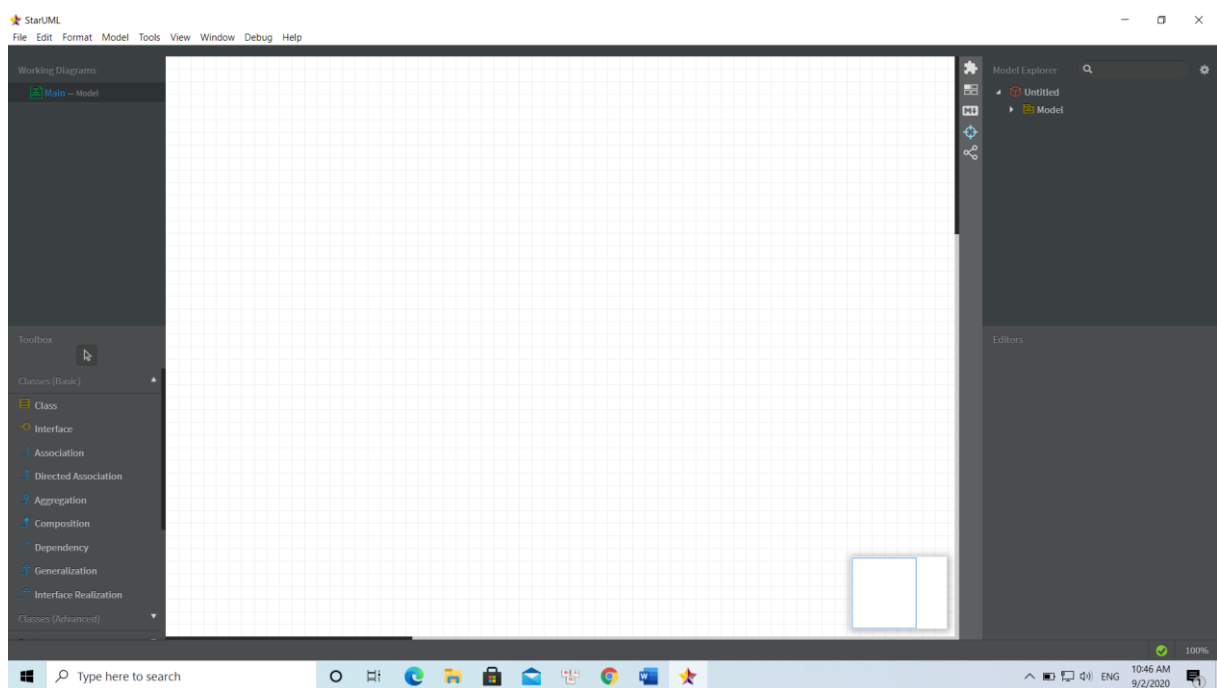
2.2.3.1. Giới thiệu

StarUML là một mô hình nền tảng, là phần mềm hỗ trợ UML (Unified Modeling Language). Nó hỗ trợ các phương pháp tiếp cận MDA (Model Driven Architecture) bằng cách hỗ trợ các khái niệm hồ sơ UML. Tức là StarUML hỗ trợ phân tích và thiết kế hệ thống một điều mà bất cứ dự án nào đều cần có. Ngoài ra dùng StarUML sẽ đảm bảo tối đa hóa năng suất và chất lượng của các dự án phần mềm của bạn. Vì nó cho phép mô hình hóa nên sẽ không phụ thuộc vào người code, ngôn ngữ code hay nền tảng sử dụng. Do mô hình hóa nên rất dễ đọc và dễ hiểu. Một điều mình thích nhất ở StarUML là khả năng tự sinh code từ những mô hình class. [8]

2.2.3.2. Cài đặt

Download StarUML tại: <http://staruml.io/>

2.2.3.3. Một số hình ảnh của StarUML



Hình 2.11: Giao diện của Star UML.

2.2.4. Visual Studio Code

2.2.4.1. Giới thiệu

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor.

Visual Studio Code 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 Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.

Một số tính năng của Visual Studio Code

- Hỗ trợ nhiều ngôn ngữ lập trình

Visual Studio Code hỗ trợ nhiều ngôn ngữ lập trình như C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, ... Vì vậy, nó dễ dàng phát hiện và đưa ra thông báo nếu chương trình có lỗi.

- Hỗ trợ đa nền tảng

Các trình viết code thông thường chỉ được sử dụng hoặc cho Windows hoặc Linux hoặc Mac Systems. Nhưng Visual Studio Code có thể hoạt động tốt trên cả ba nền tảng trên.

- Cung cấp kho tiện ích mở rộng

Trong trường hợp lập trình viên muốn sử dụng một ngôn ngữ lập trình không nằm trong số các ngôn ngữ Visual Studio hỗ trợ, họ có thể tải xuống tiện ích mở rộng. Điều này vẫn sẽ không làm giảm hiệu năng của phần mềm, bởi vì phần mở rộng này hoạt động như một chương trình độc lập.

- Kho lưu trữ an toàn

Đi kèm với sự phát triển của lập trình là nhu cầu về lưu trữ an toàn. Với Visual Studio Code, người dùng có thể hoàn toàn yên tâm vì nó dễ dàng kết nối với Git hoặc bất kỳ kho lưu trữ hiện có nào.

- Hỗ trợ web

Visual Studio Code hỗ trợ nhiều ứng dụng web. Ngoài ra, nó cũng có một trình soạn thảo và thiết kế website.

- Lưu trữ dữ liệu dạng phân cấp

Phần lớn tệp lưu trữ đoạn mã đều được đặt trong các thư mục tương tự nhau. Ngoài ra, Visual Studio Code còn cung cấp các thư mục cho một số tệp đặc biệt quan trọng.

- Hỗ trợ viết Code

Một số đoạn code có thể thay đổi chút ít để thuận tiện cho người dùng. Visual Studio Code sẽ đề xuất cho lập trình viên các tùy chọn thay thế nếu có.

- Hỗ trợ thiết bị đầu cuối

Visual Studio Code có tích hợp thiết bị đầu cuối, giúp người dùng khỏi phải chuyển đổi giữa hai màn hình hoặc trở về thư mục gốc khi thực hiện các thao tác.

- Màn hình đa nhiệm

Người dùng Visual Studio Code có thể mở cùng lúc nhiều tệp tin và thư mục – mặc dù chúng không hề liên quan với nhau.

- Intellisense

Hầu hết các trình viết mã đều có tính năng nhắc mã Intellisense, nhưng ít chương trình nào chuyên nghiệp bằng Visual Studio Code. Nó có thể phát hiện nếu bất kỳ đoạn mã nào không đầy đủ. Thậm chí, khi lập trình viên quên không khai báo biến, Intellisense sẽ tự động giúp họ bổ sung các cú pháp còn thiếu.

- Hỗ trợ Git

Visual Studio Code hỗ trợ kéo hoặc sao chép mã trực tiếp từ GitHub. Mã này sau đó có thể được thay đổi và lưu lại trên phần mềm.

- Bình luận

Việc để lại nhận xét giúp người dùng dễ dàng nhớ công việc cần hoàn thành.

Tại sao nên sử dụng Visual Studio Code?

Lý do bạn nên sử dụng Visual Studio Code là vì nó có rất nhiều ưu điểm vượt trội so với bất kỳ IDE nào khác:

- Hỗ trợ đa nền tảng: Windows, Linux, Mac
- Hỗ trợ đa ngôn ngữ: C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, JSON
- Ít dung lượng
- Tính năng mạnh mẽ
- Intellisense chuyên nghiệp
- Giao diện thân thiện

Và trên hết, Visual Studio Code là phần mềm miễn phí, được đông đảo lập trình viên trên thế giới sử dụng.

Ai là đối tượng nên sử dụng Visual Studio Code?

Với việc cải tiến và áp dụng các công nghệ mới, Visual Studio Code đã chứng minh cho các lập trình viên thấy hiệu quả của mình. Ngoài các Developers, thì Testers, Data Administrators và tất cả những ai quan tâm đến lập trình đều có thể sử dụng Visual Studio.

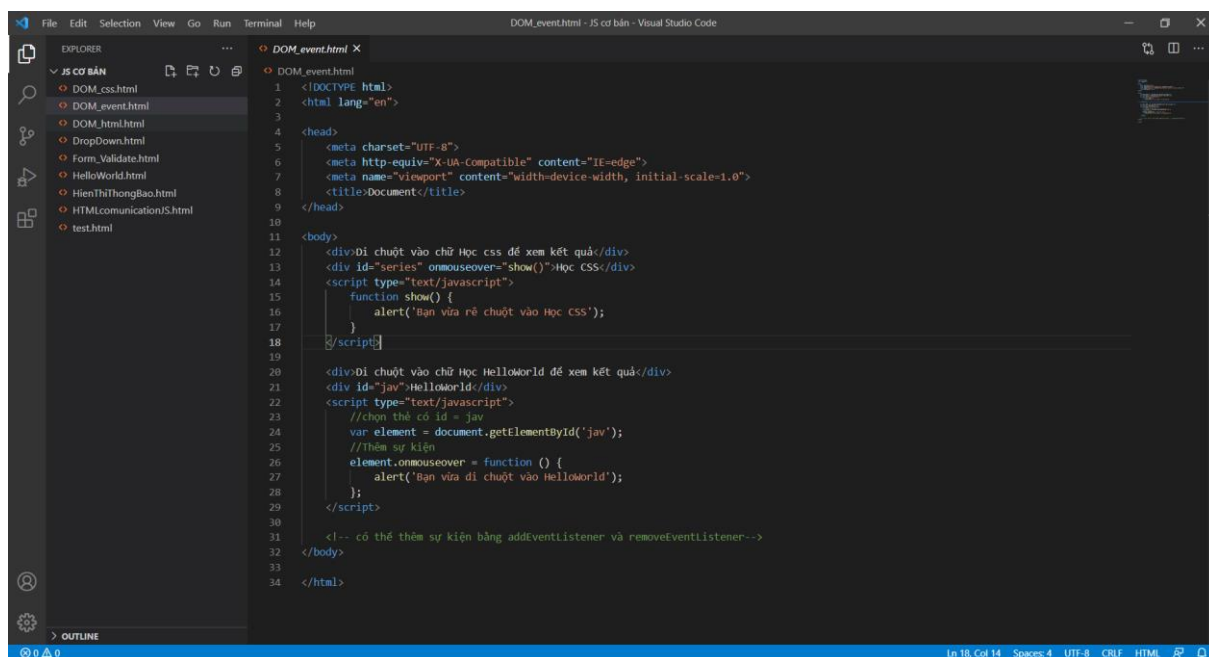
Kết luận

Visual Studio Code là một trong những trình biên tập mã phổ biến nhất hiện nay, và đang ngày càng chứng tỏ ưu thế vượt trội của mình so với những phần mềm khác. Tuy bản miễn phí không có nhiều tính năng nâng cao nhưng Visual Studio Code có thể đáp ứng được hầu hết nhu cầu cơ bản của lập trình viên. [9]

2.2.4.2. Cài đặt

Download Visual Studio Code tại: <https://code.visualstudio.com/>

2.2.4.3. Một số hình ảnh của Visual Studio Code



Hình 2.12: Giao diện của Visual Studio Code.

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1. Khảo sát hệ thống và đánh giá hiện trạng

3.1.1. Giới thiệu về hệ thống

3.1.1.1. Giới thiệu về trường đại học Công nghệ Giao thông vận tải

Trường Đại học Công nghệ Giao thông Vận tải (tên tiếng Anh: University Of Transport Technology, tên viết tắt: UTT) là trường Đại học công lập được nâng cấp năm 2011 từ Trường Cao đẳng giao thông vận tải – trực thuộc Bộ Giao thông Vận tải. Tiền thân là trường Cao đẳng Công chính, được thành lập ngày 15/11/1945. Trường đào tạo đa ngành, đa lĩnh vực về công nghệ kỹ thuật giao thông, công nghệ kỹ thuật cơ khí, ô tô, kinh tế, vận tải, logistics, Công nghệ thông tin, điện tử viễn thông, môi trường... theo định hướng ứng dụng phục vụ ngành GTVT và các ngành kinh tế quốc dân. Năm 2016, Trường được Thủ tướng Chính phủ quy hoạch phát triển thành trường đại học trọng điểm Quốc gia; năm 2017 trường là 1 trong 15 trường đại học đầu tiên của Việt Nam được công nhận đạt tiêu chuẩn chất lượng giáo dục đại học.



Hình 3.1: Trường đại học Công nghệ Giao thông vận tải – Cơ sở Hà Nội.

- *Cơ sở vật chất*

Tổng diện tích đất 21.3ha

- Cơ sở đào tạo Hà Nội: 2,3 ha. (Số 54 phố Triều Khúc, phường Thanh Xuân Nam, quận Thanh Xuân)
- Cơ sở đào tạo Vĩnh Phúc: 13 ha. (Số 278 đường Lam Sơn, phường Đồng Tâm, thành phố Vĩnh Yên, tỉnh Vĩnh Phúc)
- Cơ sở đào tạo Thái Nguyên: 6 ha. (Đường Phú Thái, phường Tân Thịnh, thành phố Thái Nguyên, tỉnh Thái Nguyên)

Trường có 220 phòng học lý thuyết, 109 phòng thí nghiệm và 02 xưởng thực hành công nghệ công trình, cơ khí.

- Ký túc xá đảm bảo cho 2.000 sinh viên ở nội trú.
- Thư viện 4.000 m² có trên 10.000 đầu sách.
- Nhà ăn có hơn 500 chỗ ngồi.
- Có đầy đủ hệ thống sân bãi, cơ sở vật chất phục vụ sinh viên tham gia các hoạt động ngoại khóa.

- *Đội ngũ cán bộ giảng viên*

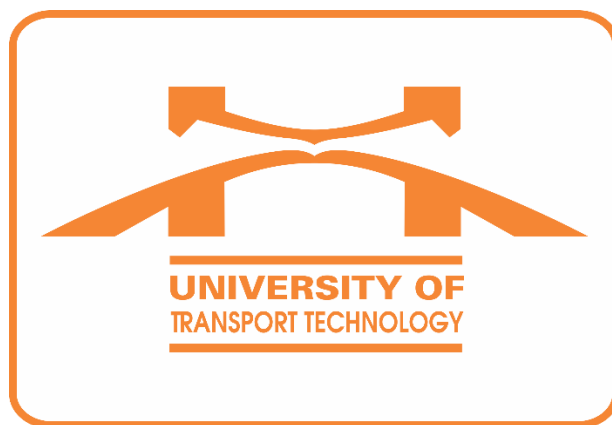
Tổng số cán bộ – giảng viên – công nhân viên: 700 người, trong đó có 486 giảng viên

- Giáo sư, phó giáo sư: 12 người
- Tiến sĩ: 110 người
- Thạc sĩ và nghiên cứu sinh: 360 người

- *Sứ mạng và tầm nhìn*

Sứ mạng

“Trường Đại học Công nghệ Giao thông vận tải có sứ mạng đào tạo và cung cấp nguồn nhân lực chất lượng cao theo hướng ứng dụng, nghiên cứu khoa học và chuyển giao công nghệ phục vụ sự nghiệp phát triển của ngành GTVT và của đất nước, phù hợp với xu thế phát triển quốc tế, hội nhập với nền giáo dục đại học tiên tiến của khu vực và trên thế giới”.



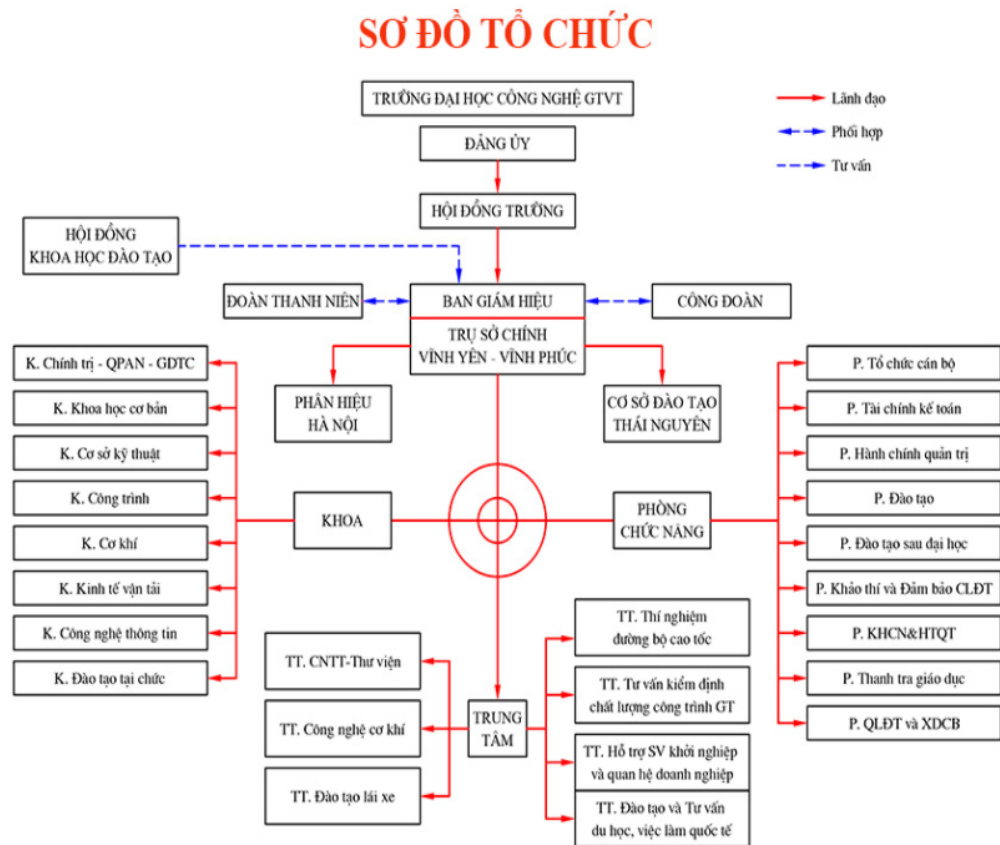
Hình 3.2: Logo trường đại học Công nghệ Giao thông vận tải.

Tầm nhìn

Tầm nhìn đến năm 2030, Trường Đại học Công nghệ Giao thông vận tải đạt được những mục tiêu sau:

- Xây dựng Trường trở thành trung tâm đào tạo nguồn nhân lực chất lượng cao hàng đầu trong lĩnh vực giao thông vận tải theo hướng ứng dụng, ... Phấn đấu đến năm 2030, trở thành trường trọng điểm Quốc gia theo hướng ứng dụng, có một số ngành đạt trình độ ngang bằng với các trường đại học có uy tín trong nước và khu vực đào tạo về lĩnh vực giao thông vận tải.
- Xây dựng Trường trở thành trung tâm nghiên cứu khoa học, ứng dụng, chuyển giao công nghệ mới trong lĩnh vực giao thông vận tải, ...Phấn đấu đến năm 2030, có 07 lĩnh vực có tầm ảnh hưởng ở Việt Nam và khu vực, gồm: Công nghệ Vật liệu - Mặt đường; Đường sắt - Metro; Kết cấu công trình; Cơ khí động lực; Logistics và vận tải đa phương thức; Ứng phó với biến đổi khí hậu và tác động môi trường GTVT; Quy hoạch, quản lý giao thông và giao thông thông minh (ITS).
- Xây dựng Trường trở thành trung tâm hợp tác quốc tế trong đào tạo và nghiên cứu khoa học, ứng dụng và chuyển giao công nghệ mới trong lĩnh vực GTVT, phát triển mô hình trường liên kết đào tạo với các trường đại học của Pháp theo định hướng ứng dụng; xây dựng mô hình hợp tác đào tạo theo đơn đặt hàng của doanh nghiệp để cung cấp cho các doanh nghiệp, nhất là các doanh nghiệp nước ngoài đang đầu tư tại Việt Nam.

- *Cơ cấu tổ chức*



Hình 3.3: Sơ đồ tổ chức trường đại học Công nghệ Giao thông vận tải.

3.1.1.2. Giới thiệu về phòng Hành chính – Quản trị của trường đại học Công nghệ Giao thông vận tải

Địa chỉ: Phòng 101, 102, Nhà H1- Trường Đại học Công nghệ GTVT- Số 54 Triều Khúc - Thanh Xuân- Hà Nội

Điện thoại: 04.35527493

Website: <http://utt.edu.vn/hanhchinhquantri>.

Email: phonghcqt@utt.edu.vn.

- *Tổ chức nhân sự*

- Trưởng phòng: Ths. Lê Xuân Ngọc
- Phó trưởng phòng: Ths. Hoàng Đình Thi
- Phó trưởng phòng: KS. Lê Văn Dậu
- Nhân viên: 63 người

- *Lịch sử*

Phòng Hành chính – Quản trị được hiệu trưởng trường đại học Công nghệ GTVT ra quyết định thành lập, số 1444/QĐ – ĐHCNGTVT ngày 23/08/2011.

- *Chức năng*

- Giúp hiệu trưởng quản trị, tổ chức thực hiện công tác hành chính, văn thư, lưu trữ theo quy định của pháp luật.
- Quản lý về cơ sở vật chất.
- Thực hiện công tác bảo vệ, giữ gìn trật tự trị an, phòng chống cháy nổ, phòng chống thiên tai, vệ sinh môi trường và cảnh quan sư phạm.

- *Nhiệm vụ*

- *Công tác hành chính:*

- Giúp Hiệu trưởng tổ chức và thực hiện quản lý hành chính các mặt hoạt động của trường, quản lý con dấu theo quy định của pháp luật
- Tiếp nhận, xử lý, lưu trữ các loại văn bản đi và đến, sao y các văn bản theo đúng quy định của Pháp luật
- Tổng hợp các thông tin hành chính từ các cơ sở đào tạo, các đơn vị; lập kế hoạch công tác hằng tháng, quý, năm; ghi biên bản và ra văn bản thông báo kết luận của các cuộc họp do Ban giám hiệu chủ trì
- Tổ chức đón tiếp khách, chuẩn bị các phòng họp phục vụ hội nghị cấp trường
- Đánh máy, sao y văn bản, tài liệu; quản lý hộp thư điện tử, thông tin liên lạc phục vụ làm việc, giảng dạy, học tập của Nhà trường
- Cấp các loại giấy tờ về hành chính cho cán bộ, viên chức của Trường đi công tác
- Kiểm tra thể thức và trình tự phát hành các văn bản hành chính theo quy định của Pháp luật
- Hướng dẫn việc thực hiện công tác hành chính, văn thư lưu trữ cho các cơ sở đào tạo, các đơn vị thuộc Trường

○ *Công tác quản trị*

- Quản lý toàn bộ cơ sở hạ tầng bao gồm: Đất đai, hệ thống giao thông, hệ thống điện, nước, vườn hoa cây cảnh, đảm bảo vệ sinh, cảnh quan môi trường; quản lý trụ sở làm việc, nhà xưởng, phòng thí nghiệm, các lớp học của Trường
- Chủ trì xây dựng định mức vật tư, quy định mua sắm; các văn bản liên quan đến quản lý tài sản, mua sắm phục vụ giảng dạy, làm việc
- Đề xuất mua sắm, bảo trì, bảo dưỡng trang thiết bị phục vụ làm việc

○ *Công tác bảo vệ*

- Tổ chức thực hiện công tác giữ gìn trật tự, đảm bảo an ninh, an toàn về con người và cơ sở vật chất tài sản trong phạm vi toàn trường
- Phối kết hợp với cơ quan công an, các đơn vị trong và ngoài trường để phòng ngừa, xử lý kịp thời các vụ việc liên quan đến an ninh chính trị, trật tự an toàn khu vực
- Tổ chức thực hiện công tác phòng chống cháy nổ, phòng chống thiên tai theo quy định của Pháp luật và kế hoạch công tác của nhà Trường

○ *Các nhiệm vụ khác theo chức năng, quyền hạn của phòng*

- Soạn thảo các văn bản quản lý, liên quan đến lĩnh vực công tác được giao trình Hiệu trưởng ký, ban hành
- Hàng tháng báo cáo kết quả triển khai nhiệm vụ công tác tại hội nghị giao ban hoặc bằng văn bản trình Hiệu trưởng
- Thực hiện các nhiệm vụ khác do Hiệu trưởng giao

3.1.2. Nội dung khảo sát nghiệp vụ quản lý văn bản trường đại học

Nội dung phần khảo sát này em có tham khảo theo quyết định số 3074/QĐ – ĐHCNGTVT ngày 16 tháng 09 năm 2016 về trình tự, thủ tục soạn thảo văn bản, ban hành văn bản, quản lý văn bản và quản lý con dấu trong Trường Đại học Công nghệ Giao thông vận tải.

3.1.2.1. Giải thích từ ngữ

Văn bản: là thông tin thành văn được truyền đạt bằng ngôn ngữ hoặc ký hiệu, hình thành trong hoạt động của các cơ quan, tổ chức và được trình bày dùng thể thức, kỹ thuật theo quy định.

Văn bản chuyên ngành: là văn bản hình thành trong quá trình thực hiện hoạt động chuyên môn, nghiệp vụ của một ngành, lĩnh vực do người đứng đầu cơ quan quản lý ngành, lĩnh vực quy định.

Văn bản hành chính: là văn bản hình thành trong quá trình chỉ đạo, điều hành, giải quyết công việc của cơ quan, tổ chức.

Văn bản điện tử: là văn bản dưới dạng thông điệp dữ liệu được tạo lập hoặc được số hóa từ văn bản giấy và trình bày đúng thể thức, kỹ thuật, định dạng theo quy định.

Bản thảo văn bản: là bản được viết hoặc đánh máy hình thành trong quá trình soạn thảo văn bản.

Bản gốc văn bản: là bản hoàn chỉnh về nội dung, thể thức văn bản được nhà trường ban hành và có chữ ký trực tiếp của người có thẩm quyền.

Bản chính văn bản: là bản hoàn chỉnh về nội dung, thể thức văn bản và được nhà trường ban hành.

Bản sao y: là bản sao đầy đủ, chính xác nội dung của bản gốc hoặc bản chính văn bản, được trình bày theo thể thức và kỹ thuật quy định.

Bản sao lục: là bản sao đầy đủ, chính xác nội dung của bản sao y, được trình bày theo thể thức và kỹ thuật quy định.

Văn bản đi: là tất cả các loại văn bản do cơ quan, tổ chức ban hành.

Văn bản đến: là tất cả các loại văn bản do cơ quan, tổ chức nhận được từ cơ quan, tổ chức, cá nhân khác gửi đến.

Văn thư trường: là bộ phận thực hiện một số nhiệm vụ công tác văn thư của trường.

3.1.2.2. Ký tắt dự thảo văn bản

a) Trách nhiệm ký tắt dự thảo văn bản

Văn bản do Hiệu trưởng, Phó Hiệu trưởng ký thì Trưởng đơn vị soạn thảo phải ký tắt vào dự thảo văn bản.

Văn bản do Trưởng đơn vị trực thuộc Trường ký thừa lệnh Hiệu trưởng thì cán bộ soạn thảo ký tắt.

Trường hợp văn bản do Trưởng đơn vị trực tiếp soạn thảo và ký thì Trưởng đơn vị không phải ký tắt vào dự thảo văn bản.

b) Vị trí ký tắt dự thảo văn bản

Chữ ký tắt của đơn vị, cá nhân soạn thảo văn bản được ký vào cuối nội dung văn bản (sau dấu ./.). Kích cỡ chữ ký tắt không vượt quá 0,5 cm².

Chữ ký tắt được đăng ký tại Văn thư Trường (sau đây gọi tắt là Văn thư).

3.1.2.3. *Ký ban hành văn bản*

a) Thẩm quyền ban hành văn bản

Trường Đại học Công nghệ Giao thông vận tải ban hành văn bản quản lý, chỉ đạo hoạt động của Trường; văn bản quy định, hướng dẫn và các văn bản khác theo chức năng, nhiệm vụ do Bộ Giao thông vận tải quy định.

b) Thẩm quyền ký văn bản

Hiệu trưởng ký các văn bản theo chức năng, nhiệm vụ quy định tại Điều 35 Quy chế Tổ chức và hoạt động của Trường Đại học Công nghệ Giao thông vận tải ban hành kèm theo Quyết định số 1702/QĐ-BGTVT ngày 09/5/2014 của Bộ trưởng Bộ Giao thông vận tải.

Trong trường hợp đặc biệt, Hiệu trưởng có thể ủy quyền cho Phó Hiệu trưởng, Trưởng đơn vị trực thuộc Trường ký một số văn bản thuộc thẩm quyền của mình. Người được ủy quyền không được ủy quyền lại.

Phó Hiệu trưởng ký các văn bản theo chức năng, nhiệm vụ quy định tại khoản 2, Điều 36 Quy chế tổ chức và hoạt động của Trường Đại học Công nghệ Giao thông vận tải.

Trưởng các đơn vị trực thuộc Trường ký các văn bản theo chức năng, nhiệm vụ, quyền hạn được giao. Trưởng đơn vị trực thuộc Trường phân công cho cấp phó của mình ký một số văn bản thuộc chức năng, nhiệm vụ, quyền hạn được giao.

Văn bản do Trưởng đơn vị trực thuộc Trường ký hoặc văn bản do Trưởng đơn vị trực thuộc Trường phân công cho cấp phó của mình ký, khi phát hành phải gửi Ban Giám hiệu để báo cáo. Văn bản do Phó Hiệu trưởng ký phải gửi Hiệu trưởng để báo cáo.

Trưởng phòng Hành chính – Quản trị ký thừa lệnh Hiệu trưởng sao các văn bản, ký văn bản đính chính, giấy mời họp và một số văn bản khác theo yêu cầu.

c) Trình ký văn bản

Văn bản được trình ký phải đảm bảo về thủ tục ban hành, thẩm quyền, thể thức và kỹ thuật trình bày.

Trình tự, thủ tục trình ký

- Văn bản do các đơn vị soạn thảo phải được gửi qua Văn thư, Văn thư có trách nhiệm chuyển văn bản tới người có thẩm quyền ký văn bản.
- Hồ sơ trình ký bao gồm: Văn bản trình ký và các văn bản liên quan; cặp trình ký có ghi rõ tên của đơn vị soạn thảo; tờ trình kèm theo.

Sau khi Hiệu trưởng, Phó Hiệu trưởng ký, Phòng Hành chính – Quản trị làm thủ tục phát hành văn bản và trả hồ sơ cho đơn vị chủ trì soạn thảo văn bản lưu.

Trường hợp Hiệu trưởng, Phó Hiệu trưởng chưa thống nhất nội dung văn bản, phòng Hành chính – Quản trị trả hồ sơ cho đơn vị chủ trì soạn thảo văn bản hoàn chỉnh theo chỉ đạo của Hiệu trưởng, Phó Hiệu trưởng. Đơn vị trình lại Hiệu trưởng, Phó Hiệu trưởng văn bản hoàn chỉnh theo chỉ đạo của Hiệu trưởng, Phó Hiệu trưởng, kèm theo hồ sơ trình ký lần trước có bút phê của Hiệu trưởng, Phó Hiệu trưởng.

d) Ký tắt phát hành văn bản

Sau khi người có thẩm quyền ký văn bản, việc ký tắt phát hành văn bản được thực hiện như sau:

- Thẩm quyền ký tắt phát hành văn bản: Trưởng phòng Hành chính – Quản trị ký tắt văn bản do Trường Đại học Công nghệ Giao thông vận tải ban hành.
- Vị trí và kích cỡ chữ ký tắt: Chữ ký tắt phát hành văn bản của Trưởng phòng Hành chính – Quản trị được ký sau dấu chấm (.) tại dòng cuối của phần “*Nơi nhận*”. Kích cỡ chữ ký tắt không vượt quá 0,3 cm².

e) Bản sao văn bản

Các hình thức bản sao gồm: sao y bản chính, trích sao và sao lục.

Thể thức sao và kỹ thuật trình bày bản sao được thực hiện theo quy định tại Điều 16 và Điều 17 Thông tư số 01/2011/TT-BNV.

Việc sao y bản chính, trích sao và sao lục văn bản do Trưởng phòng Hành chính – Quản trị quyết định.

Bản sao y bản chính, bản trích sao và bản sao lục được thực hiện theo đúng quy định tại Điều này có giá trị pháp lý như bản chính.

Bản sao chụp (photocopy cả dấu và chữ ký của văn bản chính) không thực hiện theo quy định tại khoản 2 Điều này thì chỉ có giá trị thông tin, tham khảo.

Không được sao, chụp, chuyển phát ra ngoài Trường những ý kiến ghi bên lề văn bản. Trường hợp ý kiến của Hiệu trưởng, Phó Hiệu trưởng ghi bên lề văn bản cần thiết cho việc trao đổi công tác phải được thể chế hóa bằng văn bản hành chính.

3.1.2.4. Nguyên tắc chung của việc quản lý văn bản

Tất cả văn bản đi, văn bản đến Trường Đại học Công nghệ Giao thông vận tải phải được tập trung tại Văn thư để làm thủ tục tiếp nhận, đăng ký, trừ những loại văn bản được đăng ký riêng theo quy định của pháp luật. Những văn bản đến không được đăng ký tại Văn thư, cá nhân không có trách nhiệm giải quyết.

Văn bản đi, văn bản đến thuộc ngày nào phải được đăng ký, phát hành hoặc chuyển giao trong ngày, chậm nhất là trong ngày làm việc tiếp theo. Văn bản đến có đóng dấu chỉ các mức độ khẩn “Hỏa tốc” (kể cả hỏa tốc hẹn giờ), “Thượng khẩn” và “Khẩn” (sau đây gọi chung là văn bản khẩn) phải được đăng ký, trình và chuyển giao ngay sau khi nhận được. Văn bản khẩn đi phải được hoàn tất thủ tục phát hành và chuyển phát ngay sau khi văn bản được ký.

Văn bản, tài liệu có nội dung mang bí mật Nhà nước (sau đây gọi tắt là văn bản mật) được đăng ký, quản lý theo quy định của pháp luật hiện hành về bảo vệ bí mật nhà nước và quy định của Trường Đại học Công nghệ Giao thông vận tải.

3.1.2.5. Quy trình quản lý văn bản đến

Tất cả văn bản đến phải được quản lý theo trình tự sau:

- Tiếp nhận, đăng ký văn bản đến.
- Trình, chuyển giao văn bản đến.
- Giải quyết và theo dõi, đôn đốc việc giải quyết văn bản đến.

a) Tiếp nhận, đăng ký văn bản đến

Tiếp nhận văn bản

Khi tiếp nhận văn bản đến từ mọi nguồn, Văn thư phải kiểm tra số lượng, tình trạng bì thư, dấu niêm phong (nếu có), kiểm tra, đối chiếu với nơi gửi trước khi nhận và ký nhận

Trường hợp phát hiện thiếu bì, tình trạng bì không còn nguyên vẹn hoặc văn bản chuyển đến muộn hơn thời gian ghi trên bì (đối với bì có đóng dấu “Hỏa tốc” hện giờ), Văn thư phải báo cáo ngay với Trưởng phòng Hành chính – Quản trị, trong trường hợp cần thiết phải lập biên bản cụ thể.

Văn bản khẩn đến ngoài giờ làm việc, ngày lễ, ngày nghỉ, thì người tiếp nhận có trách nhiệm ký nhận và báo cáo ngay với Trưởng phòng Hành chính – Quản trị, để giải quyết.

Đối với văn bản khẩn được chuyển phát qua máy Fax hoặc qua mạng, Văn thư phải kiểm tra số lượng văn bản, số lượng trang của mỗi văn bản, nếu phát hiện có sai sót, phải kịp thời thông báo cho nơi gửi hoặc báo cáo người được giao trách nhiệm xem xét, giải quyết.

Phân loại sơ bộ, bóc bì văn bản đến

Sau khi tiếp nhận, các bì văn bản đến được phân loại và xử lý như sau:

- Loại bóc bì: các bì văn bản gửi đến
- Loại không bóc bì: các bì có ghi dấu chỉ mức độ mật hoặc gửi đích danh cá nhân và các tổ chức đảng, đoàn thể trong Trường, Văn thư chuyển tiếp cho nơi nhận. Những bì gửi đích danh cá nhân, nếu là văn bản liên quan đến công việc chung của Trường thì cá nhân nhận văn bản có trách nhiệm chuyển cho Văn thư để đăng ký.
- Việc bóc bì văn bản mật được thực hiện theo quy định của pháp luật về Bảo vệ bí mật nhà nước và quy định cụ thể của Trường Đại học Công nghệ Giao thông vận tải.

Việc bóc bì văn bản phải đảm bảo các yêu cầu:

- Bì có đóng dấu các mức độ khẩn phải được bóc trước để giải quyết kịp thời.
- Không gây hư hại đối với văn bản, không bỏ sót văn bản trong bì
- Đối chiếu số, ký hiệu ghi ngoài bì với số, ký hiệu của văn bản trong bì (nếu có); nếu văn bản đến có kèm theo phiếu gửi thì phải đối chiếu văn bản trong bì với phiếu gửi; khi nhận phiếu gửi phải ký xác nhận,

đóng dấu vào phiếu gửi và gửi trả lại cho nơi gửi văn bản; trường hợp phát hiện có sai sót, cần thông báo cho nơi gửi biết để giải quyết.

- Đối với đơn, thư khiếu nại, tố cáo, văn bản có thời hạn giải quyết hoặc những văn bản đến mà ngày nhận cách quá xa ngày tháng của văn bản thì cần giữ lại bì và đính kèm với văn bản làm bằng chứng.

Đóng dấu “Công văn đến”

Tất cả văn bản đến thuộc diện đăng ký tại Văn thư phải được đóng dấu “CÔNG VĂN ĐẾN”; ghi số và ngày đến (kể cả giờ đến trong những trường hợp cần thiết). Đối với văn bản khẩn được chuyển qua máy fax cần sao chụp lại trước khi đóng dấu “CÔNG VĂN ĐẾN”; đối với văn bản đến được chuyển phát qua mạng trong trường hợp cần thiết phải in ra giấy và làm thủ tục đóng dấu “CÔNG VĂN ĐẾN”.

Những văn bản đến không thuộc diện đăng ký tại Văn thư (văn bản gửi tổ chức đảng, đoàn thể, cá nhân) thì vào sổ theo dõi số lượng và giao nhận với nơi nhận (không phải đóng dấu “CÔNG VĂN ĐẾN”).

Văn bản mật đến được đăng ký theo thủ tục quy định tại Mục 3 Chương III Quy định này.

Văn thư làm thủ tục gửi trả lại nơi gửi những văn bản không có chữ ký, không có dấu; ký và đóng dấu không đúng thẩm quyền; không ghi số, không ghi ngày, tháng, năm hoặc dấu đen (trừ bản Fax), bản rách, mờ không đọc được...

Dấu “CÔNG VĂN ĐẾN” được đóng rõ ràng, ngay ngắn vào khoảng giấy trống, dưới số, ký hiệu (đối với những văn bản có tên loại), dưới trích yếu nội dung (đối với công văn) hoặc vào khoảng giấy trống phía dưới ngày, tháng, năm ban hành văn bản.

Đăng ký văn bản đến

Lập 02 Sổ đăng ký văn bản đến trong đó 01 Sổ đăng ký văn bản đến dùng để đăng ký tất cả các loại văn bản (trừ văn bản mật) và 01 Sổ đăng ký văn bản mật đến.

b) Trình, chuyển giao văn bản đến

Phân phối văn bản và trình văn bản đến

Sau khi hoàn tất thủ tục tiếp nhận và đăng ký, tất cả văn bản đến được trình như sau:

- Văn thư kịp thời trình Hiệu Trưởng, Hiệu trưởng đưa ra ý kiến chỉ đạo giải quyết, văn bản đến được chuyển trở lại Văn thư.
- Đối với văn bản có nội dung liên quan đến nhiều đơn vị hoặc cá nhân thì cần ghi rõ đơn vị hoặc cá nhân chủ trì, đơn vị hoặc cá nhân phối hợp và thời hạn giải quyết (nếu cần).

Chuyển giao văn bản đến

Căn cứ vào ý kiến phân phối và ý kiến chỉ đạo giải quyết văn bản đến, Văn thư chuyển giao ngay cho đơn vị trực thuộc Trường thông qua Trưởng các đơn vị để giải quyết. Việc chuyển giao văn bản phải đảm bảo kịp thời, chính xác, đúng đối tượng, chặt chẽ, giữ gìn bí mật nội dung văn bản và được người nhận ký nhận. Trưởng đơn vị cho ý kiến chỉ đạo giải quyết và chuyển cho cá nhân trực tiếp theo dõi, giải quyết.

Khi nhận được bản chính của bản Fax hoặc văn bản chuyển qua mạng, Văn thư phải đóng dấu “CÔNG VĂN ĐẾN”, đăng ký, lấy số đến, ngày đến theo bản fax, bản chuyển phát qua mạng và chuyển cho đơn vị, cá nhân đã được giao giải quyết.

c) Giải quyết theo dõi và đôn đốc việc giải quyết văn bản đến

Giải quyết văn bản đến

Khi nhận được văn bản đến, các đơn vị, cá nhân có trách nhiệm giải quyết kịp thời theo thời hạn sau:

- Đối với văn bản khẩn, có thời hạn thì giải quyết theo thời hạn ghi trên văn bản theo quy định của pháp luật hoặc theo ý kiến chỉ đạo của Ban Giám hiệu.
- Đối với văn bản không có yêu cầu về thời hạn: thời hạn giải quyết không quá 05 ngày làm việc đối với văn bản có nội dung đơn giản; không quá 15 ngày làm việc đối với văn bản có nội dung phức tạp, liên quan đến nhiều đơn vị.
- Nếu xét thấy không thể hoàn thành thời hạn giải quyết văn bản theo quy định, đơn vị, cá nhân có trách nhiệm báo cáo Ban Giám hiệu xin ý kiến kéo dài thời hạn giải quyết.

Theo dõi, đôn đốc việc giải quyết văn bản đến

Trưởng đơn vị được giao giải quyết chịu trách nhiệm chính trong việc theo dõi, đôn đốc về thời hạn giải quyết.

Đối với văn bản đến có đóng dấu “Tài liệu thu hồi”, Văn thư có trách nhiệm theo dõi, thu hồi hoặc gửi trả lại nơi gửi theo đúng thời hạn quy định.

Trưởng phòng Hành chính – Quản trị có trách nhiệm báo cáo Hiệu trưởng về việc theo dõi thời hạn giải quyết văn bản, thừa lệnh Hiệu trưởng nhắc nhở các đơn vị chậm tiến độ trong khi giải quyết văn bản.

3.1.2.6. Quy trình quản lý văn bản đi

Tất cả văn bản đi phải được quản lý theo trình tự sau:

- Kiểm tra thể thức, kỹ thuật trình bày văn bản; ghi số và ngày tháng năm của văn bản.
- Đăng ký văn bản đi.
- Nhân bản, đóng dấu Trường và dấu mức độ mật, khẩn.
- Làm thủ tục phát hành, chuyển và theo dõi việc chuyển phát văn bản đi.
- Lưu văn bản đi.

a) Kiểm tra thể thức, kỹ thuật trình bày văn bản; ghi số và ngày tháng năm của văn bản

Kiểm tra lại thể thức và kỹ thuật trình bày văn bản: Văn thư chỉ tiếp nhận và phát hành những văn bản đã được đánh máy, không sửa chữa, tẩy xóa và có chữ ký của người có trách nhiệm; trước khi phát hành văn bản, Văn thư kiểm tra lại thể thức, kỹ thuật trình bày văn bản; nếu phát hiện sai sót thì báo cáo người có trách nhiệm xem xét, giải quyết.

Ghi số và ngày tháng năm ban hành văn bản: Việc đánh số văn bản hành chính thực hiện theo quy định tại điểm a khoản 1 Điều 8; việc ghi ngày, tháng, năm của văn bản được thực hiện theo quy định tại điểm b khoản 1 Điều 9 của Thông tư số 01/2011/TT-BNV.

Văn bản mật đi được đánh số và đăng ký riêng.

b) Đăng ký văn bản đi

Căn cứ tổng số và số lượng mỗi loại văn bản đi hàng năm, lập sổ đăng ký văn bản đi cho phù hợp.

Sổ văn bản mật đi được đánh số và theo dõi riêng.

c) Nhân bản, đóng dấu Trường và dấu mức độ mật, khẩn

Nhân bản

Số lượng văn bản cần nhân bản để phát hành được xác định trên cơ sở số lượng tại nơi nhận văn bản; nếu gửi đến nhiều nơi mà trong văn bản không liệt kê đủ danh sách thì đơn vị soạn thảo phải có phụ lục nơi nhận kèm theo để lưu ở Văn thư.

Nơi nhận phải được xác định cụ thể trong văn bản trên nguyên tắc văn bản chỉ gửi đến cơ quan, tổ chức, đơn vị có chức năng, thẩm quyền giải quyết, tổ chức thực hiện, phối hợp thực hiện, báo cáo, giám sát, kiểm tra liên quan đến nội dung văn bản; không gửi vượt cấp, không gửi nhiều bản cho một đối tượng, không gửi đến các đối tượng khác chỉ để biết, tham khảo.

Giữ gìn bí mật nội dung văn bản nhân bản theo đúng thời gian quy định.

Việc nhân bản văn bản mật phải có ý kiến của Hiệu trưởng và được thực hiện theo quy định hiện hành về Bảo vệ bí mật nhà nước.

Đóng dấu

Việc đóng dấu Trường được thực hiện theo quy định tại Điều 31 Quy định này.

Việc đóng dấu các mức độ mật, dấu “Tài liệu thu hồi” thực hiện theo quy định tại điểm a, khoản 2, Điều 15 Thông tư số 01/2011/TT-BNV.

Việc đóng dấu các mức độ khẩn được thực hiện theo quy định tại điểm b, khoản 2, Điều 15 Thông tư số 01/2011/TT-BNV.

d) Làm thủ tục phát hành, chuyển và theo dõi việc chuyển phát văn bản đi

Làm thủ tục chuyển phát văn bản đi

Văn bản đi được chuyển phát thông qua các hình thức: qua hệ thống bưu điện, qua hệ thống thư điện tử.

Văn bản đi phải được chuyển phát trong ngày. Trường hợp nhận cuối giờ làm việc của ngày hôm trước hoặc văn bản nhiều trang, phát hành với số lượng lớn thì phát hành chậm nhất trong ngày làm việc tiếp theo.

Đối với các văn bản khẩn có thể gửi fax, sau đó gửi ngay bản chính văn bản.

Văn bản đi chuyển giao trực tiếp cho các đơn vị, cá nhân ngoài Trường phải được đăng ký vào Sổ chuyển giao văn bản đi, người nhận ký nhận vào sổ (nếu cần); trường hợp nhận trực tiếp tại Văn thư, người nhận phải có chứng minh thư nhân dân, xác nhận hoặc giấy giới thiệu của cơ quan, đơn vị nhận văn bản; đối với văn bản mật phải có ý kiến của người có thẩm quyền.

Văn bản đi được chuyển phát qua bưu điện phải được đăng ký vào Sổ gửi văn bản đi và nhân viên bưu điện phải ký nhận vào sổ.

Chuyển phát văn bản mật thực hiện theo quy định của pháp luật hiện hành về Bảo vệ bí mật nhà nước.

Theo dõi chuyển phát văn bản đi

Lập phiếu gửi để theo dõi việc chuyển phát văn bản đi theo yêu cầu của người ký văn bản (nếu có). Việc xác định những văn bản đi cần lập phiếu gửi do đơn vị hoặc cá nhân soạn thảo văn bản đề xuất, trình người ký văn bản quyết định.

Đối với những văn bản đi có đóng dấu “Tài liệu thu hồi”, phải theo dõi, thu hồi đúng thời hạn; khi nhận lại, phải kiểm tra, đối chiếu để bảo đảm văn bản không bị thiếu hoặc thất lạc.

Đối với bì văn bản gửi đi mà Bưu điện trả lại phải báo cáo người có thẩm quyền, kịp thời chuyển cho đơn vị hoặc cá nhân soạn thảo văn bản đó; đồng thời cập nhật vào sổ gửi văn bản đi để theo dõi.

Trường hợp phát hiện văn bản bị thất lạc phải kịp thời báo cáo người có thẩm quyền để xử lý.

e) Lưu văn bản đi

Mỗi văn bản đi phải lưu 2 bản: bản gốc lưu tại Văn thư và 01 bản chính lưu trong hồ sơ công việc của đơn vị soạn thảo. Ngoài ra, đơn vị soạn thảo phải lưu một bản có chữ ký của người có thẩm quyền nhưng không có dấu.

Trường hợp văn bản đi được dịch ra tiếng nước ngoài hoặc bằng tiếng dân tộc thiểu số, ngoài bản lưu bằng tiếng Việt phải lưu kèm theo bản dịch chính xác nội dung bằng tiếng nước ngoài hoặc bằng tiếng dân tộc thiểu số.

Bản gốc lưu tại Văn thư phải được đóng dấu và sắp xếp theo thứ tự đăng ký.

Việc lưu giữ, bảo quản và sử dụng bản lưu văn bản đi có đóng dấu chỉ các mức độ mật được thực hiện theo quy định hiện hành về bảo vệ bí mật nhà nước.

3.1.2.7. *Quản lý và sử dụng con dấu*

a) Quản lý con dấu

Trưởng phòng Hành chính – Quản trị chịu trách nhiệm trước Hiệu trưởng trong việc quản lý, sử dụng con dấu của Nhà Trường.

Con dấu được giao cho Văn thư quản lý và sử dụng. Người được giao sử dụng và bảo quản con dấu chịu trách nhiệm trước Hiệu trưởng và trước pháp luật. Việc quản lý con dấu được thực hiện như sau:

- Con dấu phải được bảo quản tại phòng làm việc; không giao con dấu cho người khác, không đưa con dấu ra khỏi Trường khi chưa được phép bằng văn bản của Hiệu trưởng. Con dấu phải được bảo quản tuyệt đối an toàn trong giờ cũng như ngoài giờ làm việc.
- Khi đóng dấu xong, con dấu được cất vào tủ, khi ra ngoài phòng làm việc phải khóa tủ dấu; chỉ người được giao giữ dấu mới được giữ chìa khóa.

Con dấu đang sử dụng bị biến dạng, mòn, hỏng, hoặc có thay đổi tổ chức, đổi tên thì phải thực hiện thủ tục đăng ký lại mẫu con dấu, nộp lại con dấu và giấy chứng nhận đăng ký mẫu con dấu đã được cấp trước đó cho cơ quan đăng ký.

Trường hợp con dấu bị mất, trong thời hạn 02 ngày Trưởng phòng Hành chính – Quản trị báo ngay cho cơ quan công an nơi xảy ra mất con dấu và cơ quan công an đã cấp giấy chứng nhận đăng ký mẫu dấu; lập biên bản và báo cáo Hiệu trưởng, đồng thời phải thông báo hủy con dấu bị mất.

Giấy chứng nhận đăng ký mẫu con dấu bị mất phải thực hiện thủ tục cấp lại giấy chứng nhận đăng ký mẫu con dấu. Trường hợp bị hỏng phải thực hiện thủ tục đổi lại giấy chứng nhận đăng ký mẫu con dấu và nộp lại giấy chứng nhận đăng ký mẫu con dấu đã được cấp trước đó cho cơ quan đăng ký mẫu con dấu.

Khi có quyết định thu hồi con dấu, nộp lại con dấu cho cơ quan công an cấp giấy chứng nhận đăng ký mẫu dấu và làm thủ tục xin khắc con dấu mới.

b) Sử dụng con dấu

Người được giao sử dụng, bảo quản con dấu phải tự tay đóng dấu vào các văn bản của Trường.

- Chỉ đóng dấu vào các văn bản khi văn bản đã có chữ ký của người có thẩm quyền.

- Không được đóng dấu trong các trường hợp sau: Đóng dấu vào giấy không có nội dung, đóng dấu trước khi ký, đóng dấu sẵn trên giấy trắng hoặc đóng dấu lên các văn bản có chữ ký của người không có thẩm quyền.

Dấu đóng phải rõ ràng, ngay ngắn, đúng chiều và dùng mực dấu màu đỏ tươi theo quy định.

Đóng dấu trên văn bản chính thức.

- Văn thư chỉ đóng dấu khi văn bản đúng thể thức, kỹ thuật trình bày và chữ ký của người có thẩm quyền ký văn bản.
- Khi đóng dấu lên chữ ký phải trùm lên khoảng 1/3 chữ ký về phía bên trái.
- Dấu giáp lai được đóng vào khoảng giữa mép phải của văn bản hoặc phụ lục văn bản, trùm lên một phần các tờ giấy, mỗi dấu đóng tối đa 05 trang văn bản.
- Khi đóng dấu những văn bản, tài liệu như hợp đồng, biên bản, giấy chứng nhận ... Văn thư phải lập sổ theo dõi riêng và lưu 01 bản.

Đóng dấu vào phụ lục kèm theo văn bản chính do người ký văn bản quyết định và dấu được đóng lên trang đầu, trùm lên một phần tên Trường, đơn vị hoặc tên của phụ lục.

Trường hợp đóng dấu nhầm không được đóng chồng lên dấu cũ mà phải hủy văn bản đó và làm lại văn bản mới.

3.1.2.8. *Quản lý tài liệu mật*

a) Gửi tài liệu mật đi

Vào sổ: Tài liệu mật trước khi gửi đi các nơi nhất thiết phải vào sổ "Tài liệu mật" riêng để theo dõi. Sổ "Tài liệu mật đi" phải ghi đầy đủ các cột mục: Số thứ tự (đồng thời là số tài liệu gửi đi) ngày tháng năm, nơi nhận, trích yếu nội dung, độ mật, độ khẩn, người ký, ghi chú (ghi văn bản kèm theo nếu có).

Trường hợp tài liệu "Tuyệt mật", thì người chuẩn bị văn bản vẫn phải lấy sổ đi và đăng ký theo đúng các cột mục trong sổ, riêng cột trích yếu nội dung bỏ trống (ghi sau nếu người có thẩm quyền đồng ý) cho vào bì dán kín và làm các yêu cầu bảo mật trước khi đưa đến văn thư để làm bì gửi đi.

Lập phiếu gửi: Tài liệu mật gửi đi phải kèm theo phiếu bỏ chung với tài liệu. Phiếu gửi ghi rõ nơi gửi, số phiếu, nơi nhận, số, ký hiệu từng tài liệu gửi đi, đóng dấu độ mật, độ khẩn theo độ mật, độ khẩn của tài liệu vào góc phải phía trên của tờ phiếu. Nơi nhận khi nhận được tài liệu mật phải hoàn lại ngay phiếu gửi cho nơi gửi.

Làm bì: Tài liệu mật gửi đi không gửi chung trong một bì với tài liệu thường. Giấy làm bì phải là loại giấy dai, khó bóc, không thấm nước, không nhìn thấu qua được; gấp bì theo mỗi chéo; hồ dán phải dính, khó bóc.

Tài liệu độ "Mật" ngoài bì đóng dấu ký hiệu chữ "C" in hoa nét đậm (không đóng dấu "Mật")

Tài liệu "Tuyệt mật", "Tối mật" gửi bằng hai bì:

- Bì trong: Ghi rõ số, ký hiệu của tài liệu, tên người nhận, đóng dấu mật theo độ mật của tài liệu. Nếu là tài liệu "Tuyệt mật" gửi đích danh cho người có trách nhiệm giải quyết, thì đóng dấu "chỉ người có tên mới được bóc phong bì".
- Bì ngoài: Ghi như gửi tài liệu thường, đóng dấu ký hiệu độ mật chữ "A" in hoa nét đậm là "Tuyệt mật", chữ "B" in hoa nét đậm là "Tối mật" (không đóng dấu "Tuyệt mật", "Tối mật").

Niêm phong: Tài liệu "Tuyệt mật", "Tối mật" gửi đi, bì trong sau khi dán bằng hồ, phải dán bằng keo đề lên các mép dán hồ và niêm phong bằng chỉ hoặc si hoặc giấy thật mỏng khó bóc, niêm phong lên giao điểm các mối chéo phía sau của bì, dấu niêm phong ở các góc giấy niêm, một nửa trên giấy niêm, một nửa trên giấy bì. Mực niêm phong dùng loại mực màu đỏ tươi.

b) Nhân tài liệu mật đến

Mọi tài liệu mật bất cứ từ nguồn nào gửi đến đều phải qua Văn thư vào sổ "Tài liệu mật đến" riêng để theo dõi, và chuyển đến người có trách nhiệm giải quyết.

Những bì thuộc loại mật hoặc ghi "*chỉ người có tên mới được bóc bì*", Văn thư đóng dấu đến vào bì, đăng ký vào "Sổ đăng ký văn bản mật đến" và chuyển giao cho người có thẩm quyền hoặc người có tên trên bì giải quyết. Người nhận tùy tính chất, mức độ quan trọng của văn bản chuyển Văn thư để đăng ký văn bản đến hoặc lưu lại đơn vị và lập sổ theo dõi riêng.

Đối với điện mật, Văn thư trình người có thẩm quyền ngay sau khi tiếp nhận; sau khi có ý kiến giải quyết, Văn thư chuyển cho đơn vị liên quan xử lý. Khi xử lý xong, đơn vị được giao chủ trì phải chuyển cho Văn thư để làm thủ tục hoàn trả theo chế độ bảo mật của nhà nước quy định.

Tài liệu mật gửi đến không đủ thủ tục theo quy định, một mặt chuyển ngay đến người có trách nhiệm giải quyết, một mặt nhanh chóng tìm cách hỏi lại và rút kinh nghiệm với nơi gửi.

c) Thu hồi tài liệu mật

Văn thư phải theo dõi, thu hồi hoặc trả lại nơi gửi đúng kỳ hạn những tài liệu mật có đóng dấu thu hồi. Khi nhận cũng như khi trả phải kiểm tra, đối chiếu và xoá sổ.

d) Giao nhận tài liệu mật

Mọi trường hợp giao nhận tài liệu mật giữa các khâu (người dự thảo, đánh máy, in, văn thư, giao thông viên, người có trách nhiệm giải quyết, người cất giữ bảo quản...) đều phải vào sổ có ký nhận giữa hai bên giao nhận, giao nhận trực tiếp, tại phòng làm việc theo quy định của Hiệu trưởng.

Hiệu trưởng quy định việc lưu giữ tài liệu mật.

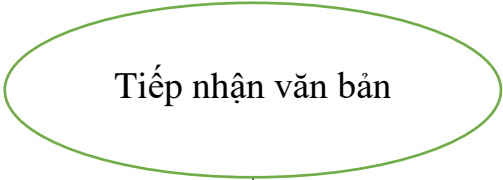
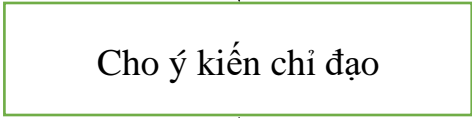
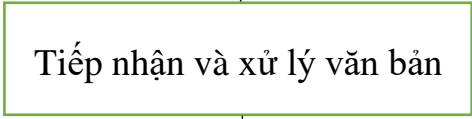
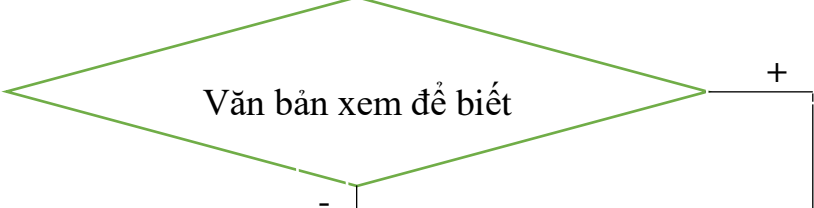
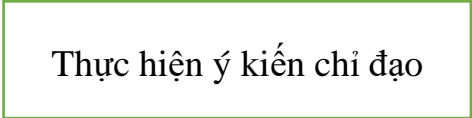
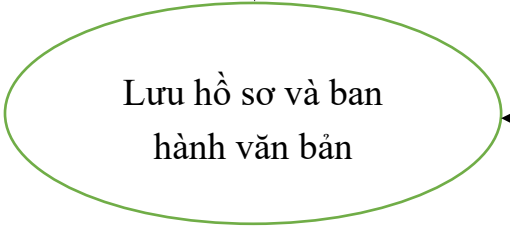
e) Vận chuyển tài liệu mật

Mọi trường hợp vận chuyển tài liệu mật phải có phương tiện mang giữ tốt (hòm sắt, cặp có khoá chắc chắn) không buộc sau xe đạp, mô tô, không được giao cho người không có trách nhiệm giữ hộ, không được để ở bất cứ nơi nào mà không có người có trách nhiệm coi giữ cẩn thận

3.1.3. Tóm tắt nghiệp vụ quản lý văn bản

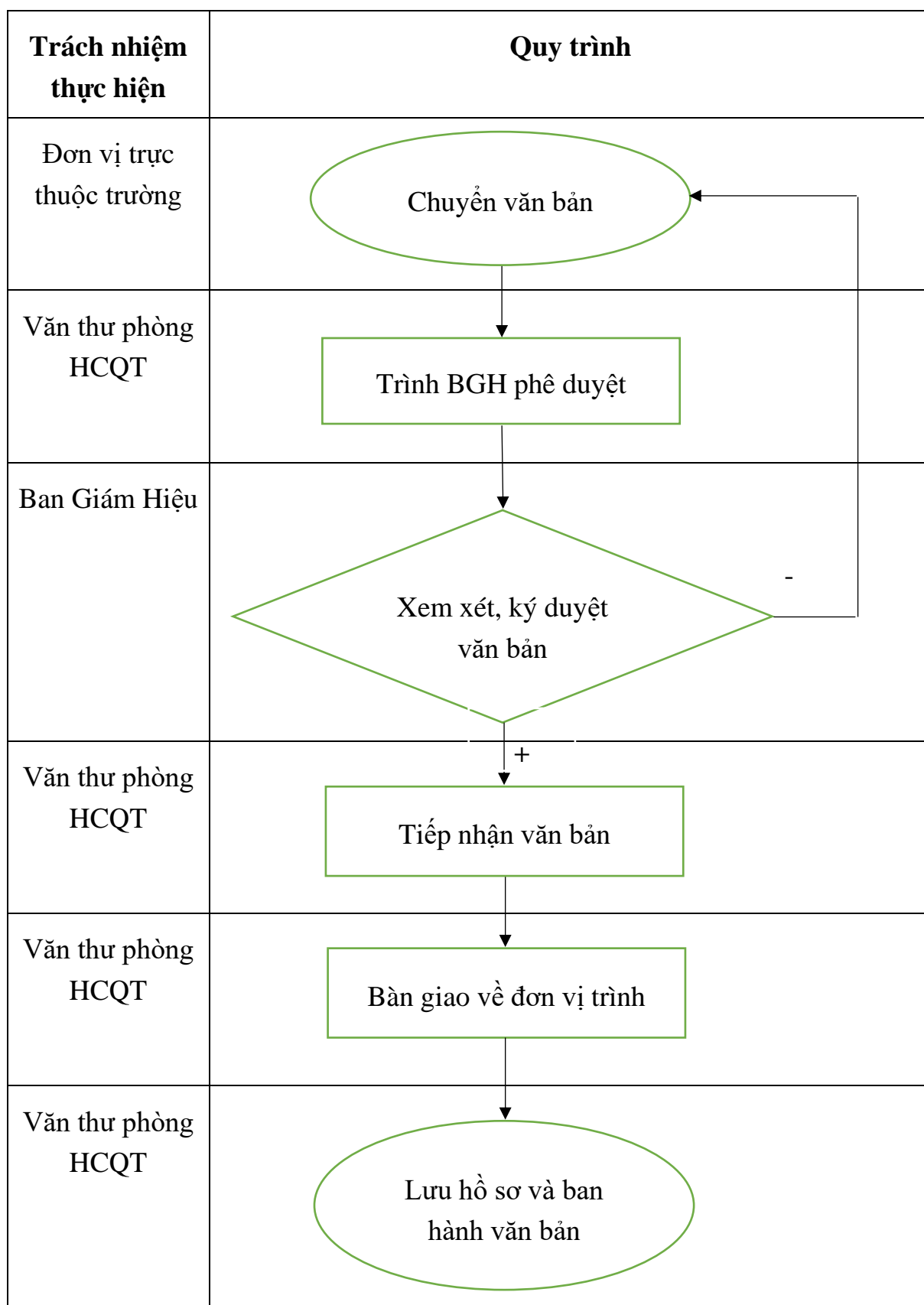
3.1.3.1. Nghiệp vụ quản lý văn bản đến

Bảng 3.1: Tóm tắt quy trình quản lý văn bản đến.

Trách nhiệm thực hiện	Quy trình
Văn thư phòng HCQT	
Ban Giám Hiệu	
Văn thư phòng HCQT	
Văn thư phòng HCQT	
Đơn vị tiếp nhận	
Văn thư phòng HCQT	

3.1.3.2. Nghiệp vụ quản lý văn bản đi

Bảng 3.2: Tóm tắt quy trình quản lý văn bản đi.



3.1.4. Đánh giá hiện trạng hệ thống

Quản lý văn bản là một việc quan trọng nhưng hiện tại việc quản lý văn bản của trường đại học Công nghệ Giao thông vận tải chưa thực sự được ưu tiên, Thực tế các thầy cô vẫn quản lý văn bản bằng phương pháp thủ công truyền thống như lưu trữ trong tủ, hòm...

Việc quản lý văn bản bằng phương pháp này đem lại hiệu quả quản lý không cao, gây tiêu tốn thời gian, chi phí và nguồn lực. Dữ liệu trên giấy thường được thanh lọc sau một khoảng thời gian nhất định vì vậy việc mất mát dữ liệu điều khó tránh khỏi.

3.1.5. Phát biểu bài toán

Việc xây dựng một hệ thống quản lý văn bản là cần thiết, tuy nhiên hệ thống phải được thiết kế sao cho đáp ứng được các yêu cầu sau:

a) Dựa trên quy định về quản lý văn bản của trường đại học CNGTVT

- Quy định về ký ban hành văn bản.
- Quy định về quản lý văn bản.

b) Yêu cầu về dữ liệu

- Dựa theo các quy định của trường trong việc quản lý văn bản.
- Dữ liệu phải là dữ liệu mới nhất và có thể thay đổi theo nghiệp vụ, cũng như nhu cầu của người sử dụng.
- Đảm bảo dữ liệu luôn đầy đủ, chính xác.

c) Yêu cầu phi chức năng

- Để phù hợp với môi trường giáo dục, giao diện của hệ thống nên chọn những gam màu cơ bản, trầm, đơn giản.
- Tối giản các trường dữ liệu.
- Tối đa các lưu ý, hướng dẫn để việc sử dụng hệ thống, với mong muốn hệ thống dễ hiểu, thân thiện và dễ sử dụng.

3.2. Phân tích và đặc tả yêu cầu

Sau khi khảo sát và xác định được các yêu cầu của hệ thống, có thể phân tích để đưa ra các quy trình nghiệp vụ chung của hệ thống bao gồm các phần:

Quyền Văn Thư:

- Quản lý tài khoản: Phân quyền, thêm, sửa, thay đổi trạng thái tài khoản.
- Quản lý phòng ban: Thêm, sửa, xóa phòng ban.
- Quản lý loại văn bản: Thêm, sửa, xóa loại văn bản.
- Quản lý văn bản đến: Thêm, sửa, xóa, ban hành văn bản đến.
- Quản lý văn bản đi: Thêm, sửa, xóa văn bản đi.
- Đọc, tìm kiếm văn bản.

Quyền Ban Giám Hiệu

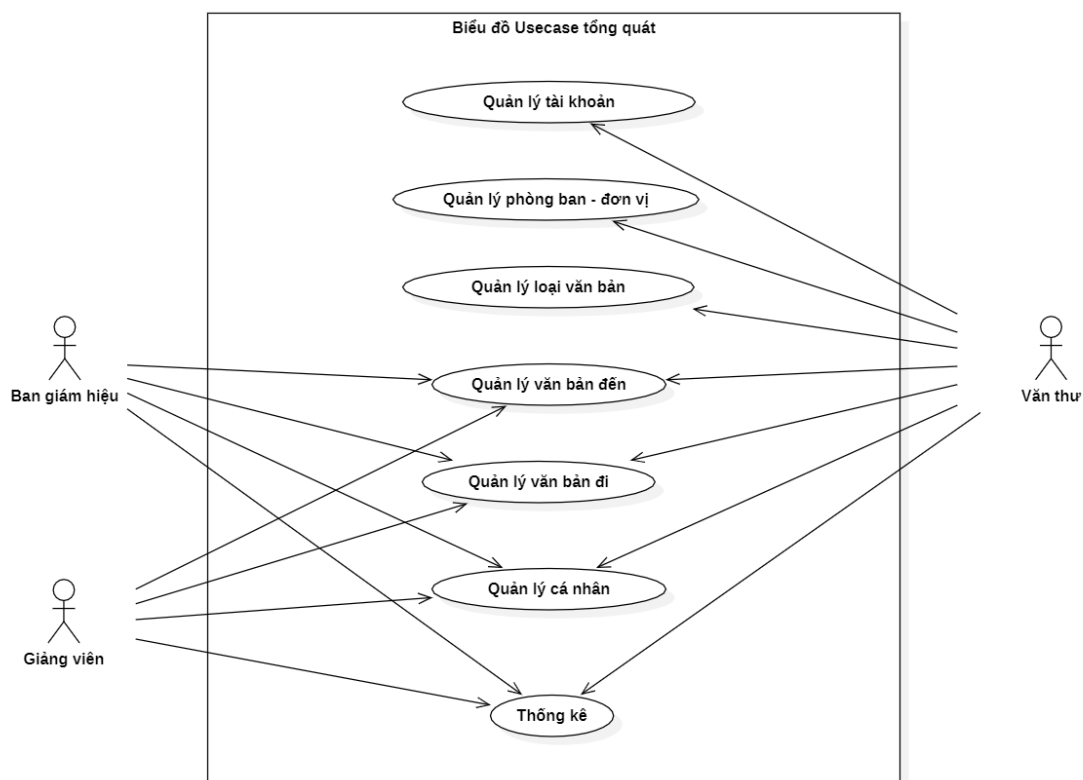
- Phê duyệt văn bản đến.
- Phê duyệt văn bản đi.
- Đọc, tìm kiếm văn bản văn bản.
- Thay đổi thông tin cá nhân.

Quyền Giảng Viên

- Đọc, tìm kiếm văn bản đã ban hành.
- Thay đổi thông tin cá nhân.

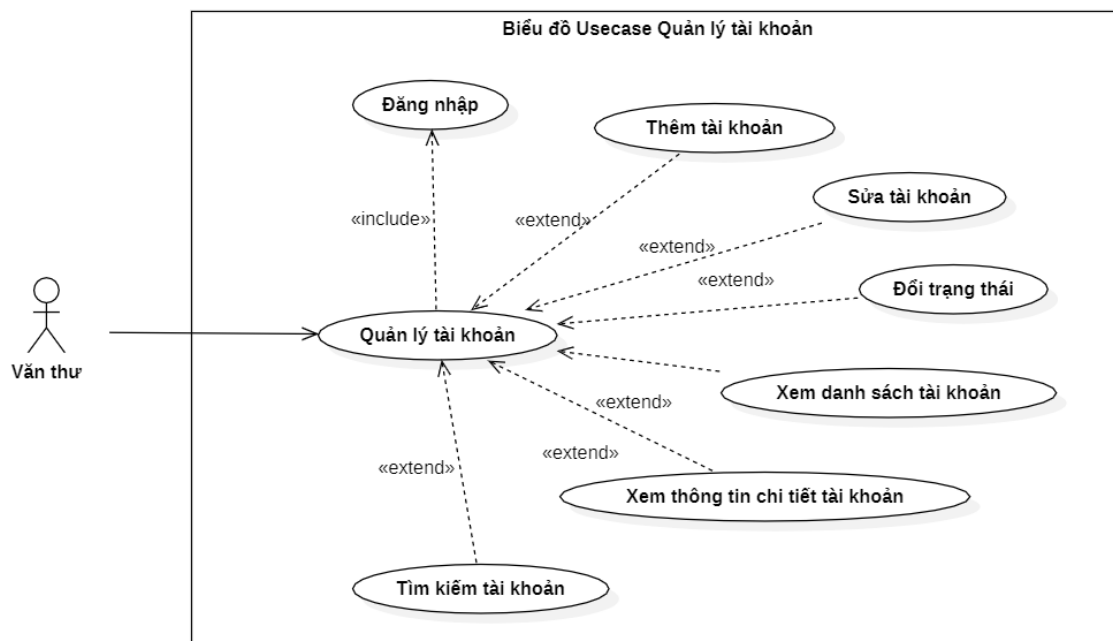
3.2.1. Xây dựng biểu đồ Usecase

3.1.1.1. Biểu đồ Usecase tổng quát



Hình 3.4: Biểu đồ Usecase tổng quát.

3.1.1.2. Biểu đồ Usecase quản lý tài khoản

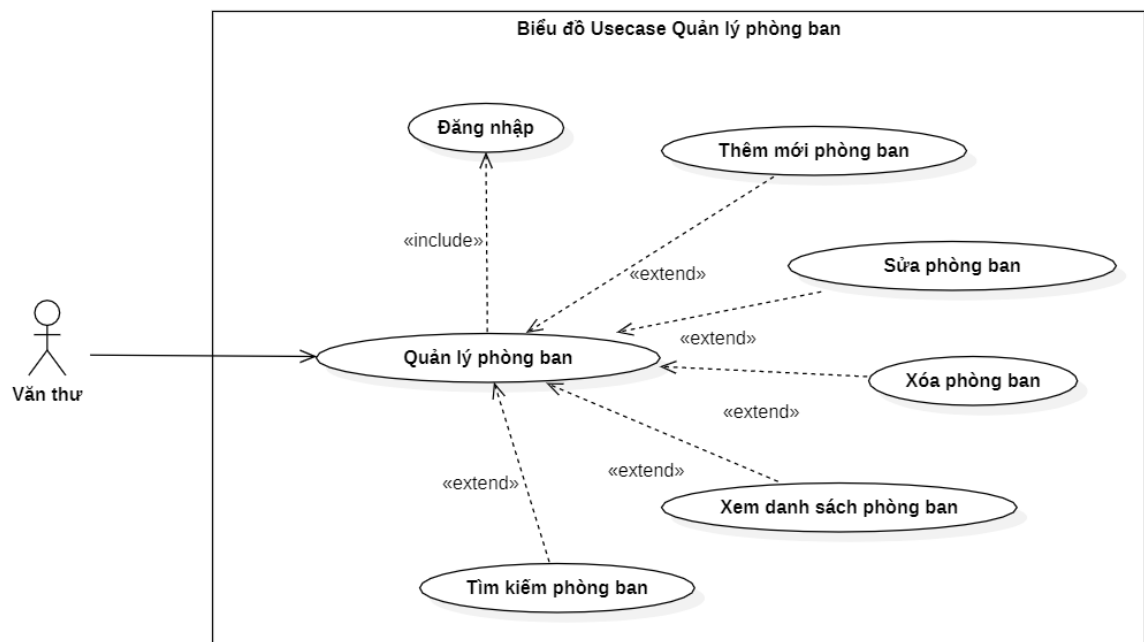


Hình 3.5: Biểu đồ Usecase quản lý tài khoản.

Bảng 3.3: Kịch bản Usecase quản lý tài khoản

Tên Use Case	
Tên Actor	
Mô tả	
Tiền điều kiện	
Hậu điều kiện	
Luồng sự kiện chính	
Luồng sự kiện phụ	

3.1.1.3. Biểu đồ Usecase quản lý phòng ban



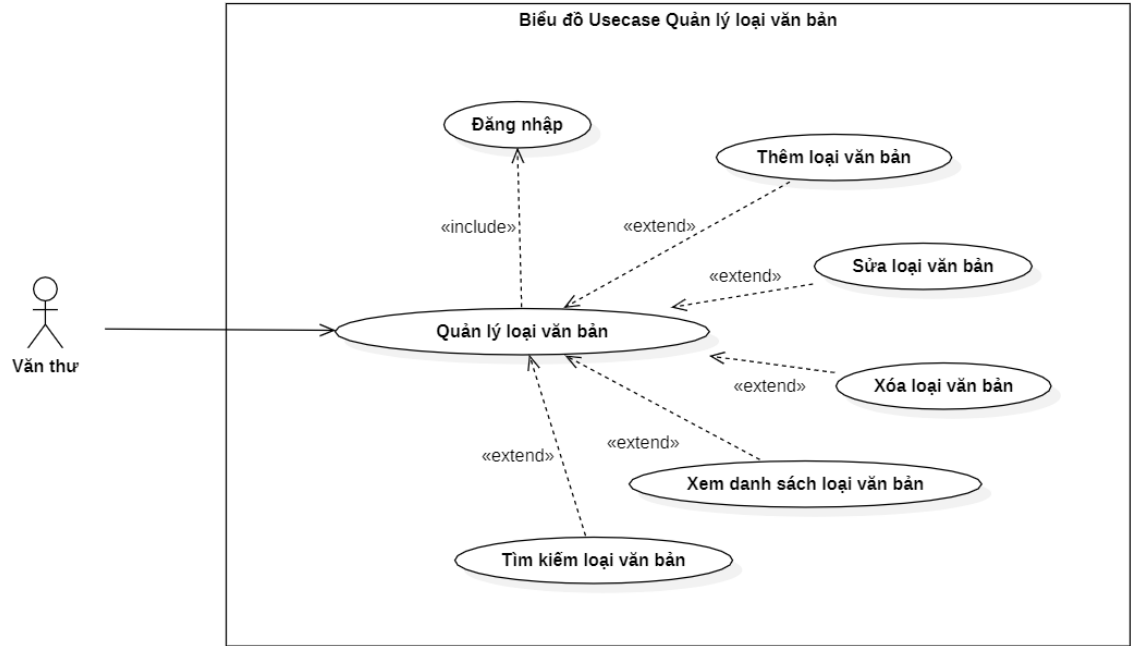
Hình 3.6: Biểu đồ Usecase quản lý phòng ban.

Bảng 3.4: Kịch bản Usecase quản lý phòng ban

Tên Use Case	
Tên Actor	

Mô tả	
Tiền điều kiện	
Hậu điều kiện	
Luồng sự kiện chính	
Luồng sự kiện phụ	

3.1.1.4. Biểu đồ Usecase quản lý loại văn bản



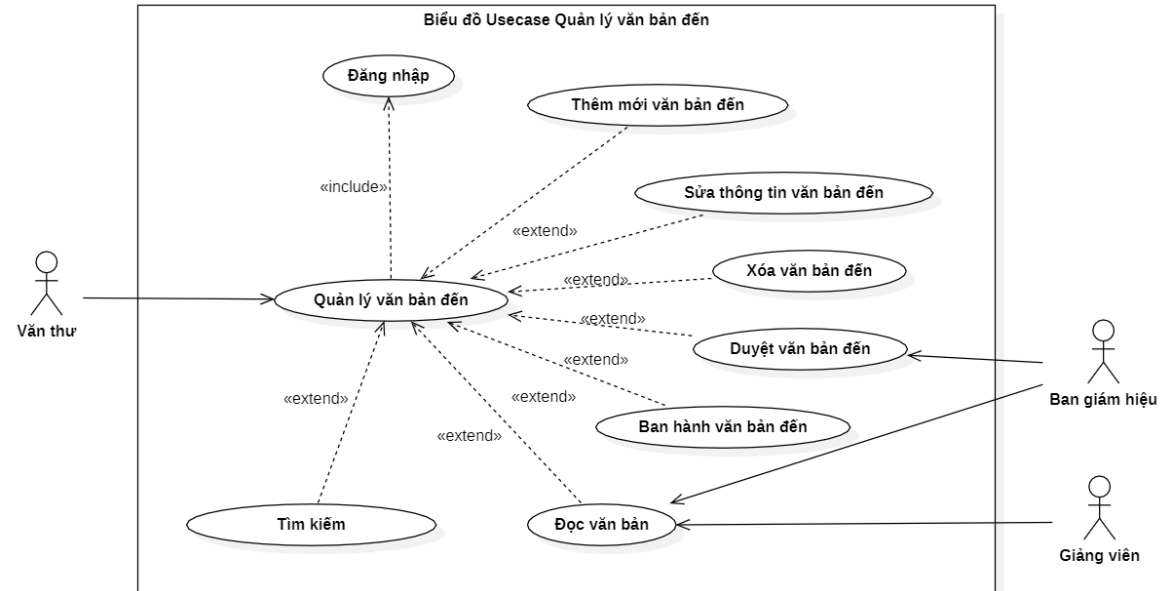
Hình 3.7: Biểu đồ Usecase quản lý loại văn bản.

Bảng 3.5: Kịch bản Usecase quản lý loại văn bản

Tên Use Case	
Tên Actor	
Mô tả	
Tiền điều kiện	
Hậu điều kiện	

Luồng sự kiện chính	
Luồng sự kiện phụ	

3.1.1.5. Biểu đồ Usecase quản lý văn bản đến

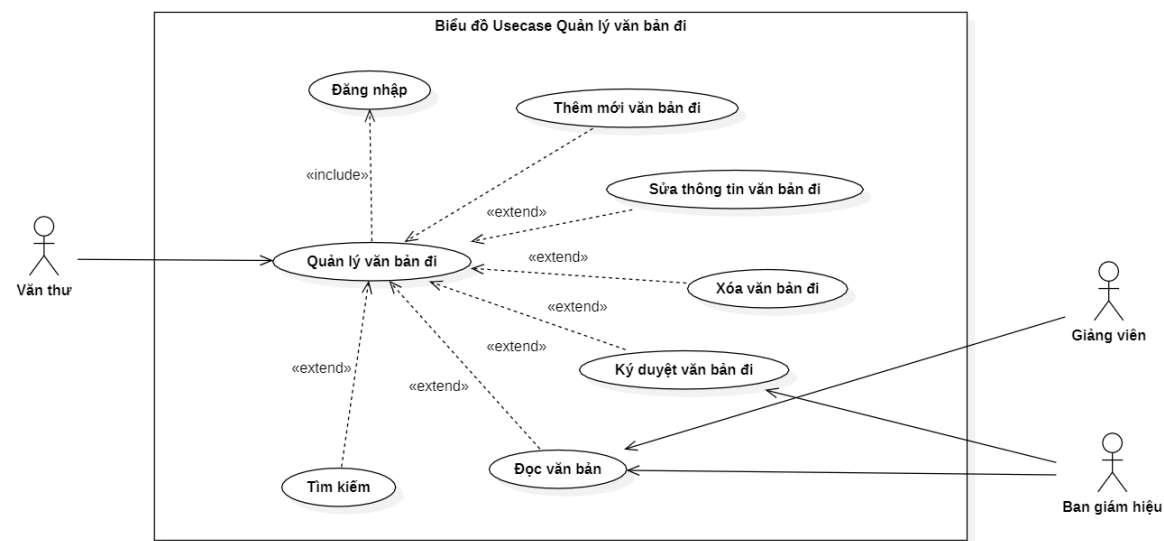


Hình 3.8: Biểu đồ Usecase quản lý văn bản đến.

Bảng 3.6: Kịch bản Usecase quản lý văn bản đến

Tên Use Case	
Tên Actor	
Mô tả	
Tiền điều kiện	
Hậu điều kiện	
Luồng sự kiện chính	
Luồng sự kiện phụ	

3.1.1.6. Biểu đồ Usecase quản lý văn bản đi

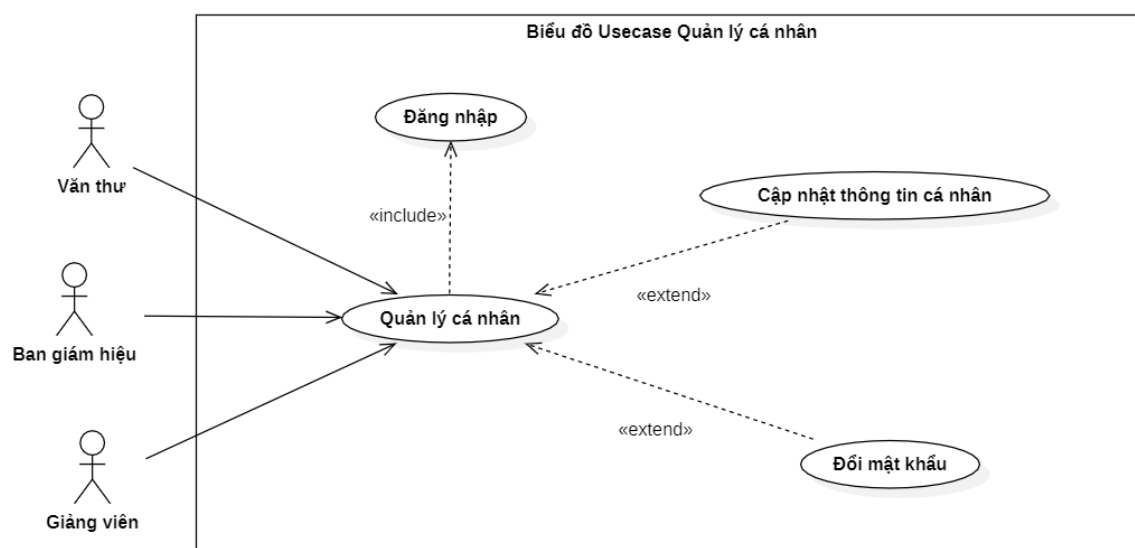


Hình 3.9: Biểu đồ Usecase quản lý văn bản đi.

Bảng 3.7: Kịch bản Usecase quản lý văn bản đi

Tên Use Case	
Tên Actor	
Mô tả	
Tiền điều kiện	
Hậu điều kiện	
Luồng sự kiện chính	
Luồng sự kiện phụ	

3.1.1.7. Biểu đồ Usecase quản lý cá nhân

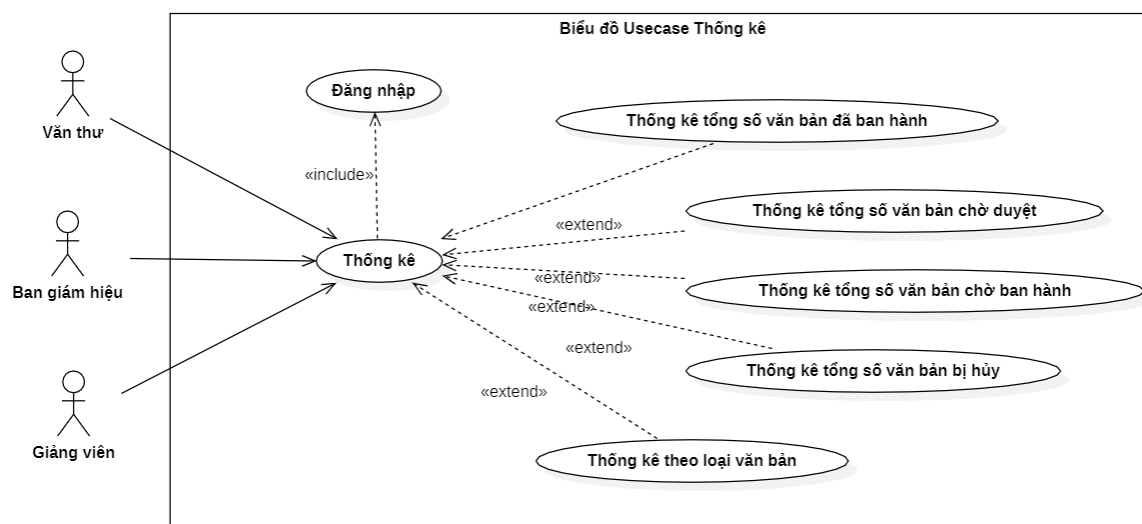


Hình 3.10: Biểu đồ Usecase quản lý cá nhân.

Bảng 3.8: Kịch bản Usecase quản lý cá nhân

Tên Use Case	
Tên Actor	
Mô tả	
Tiền điều kiện	
Hậu điều kiện	
Luồng sự kiện chính	
Luồng sự kiện phụ	

3.1.1.8. Biểu đồ Usecase thống kê

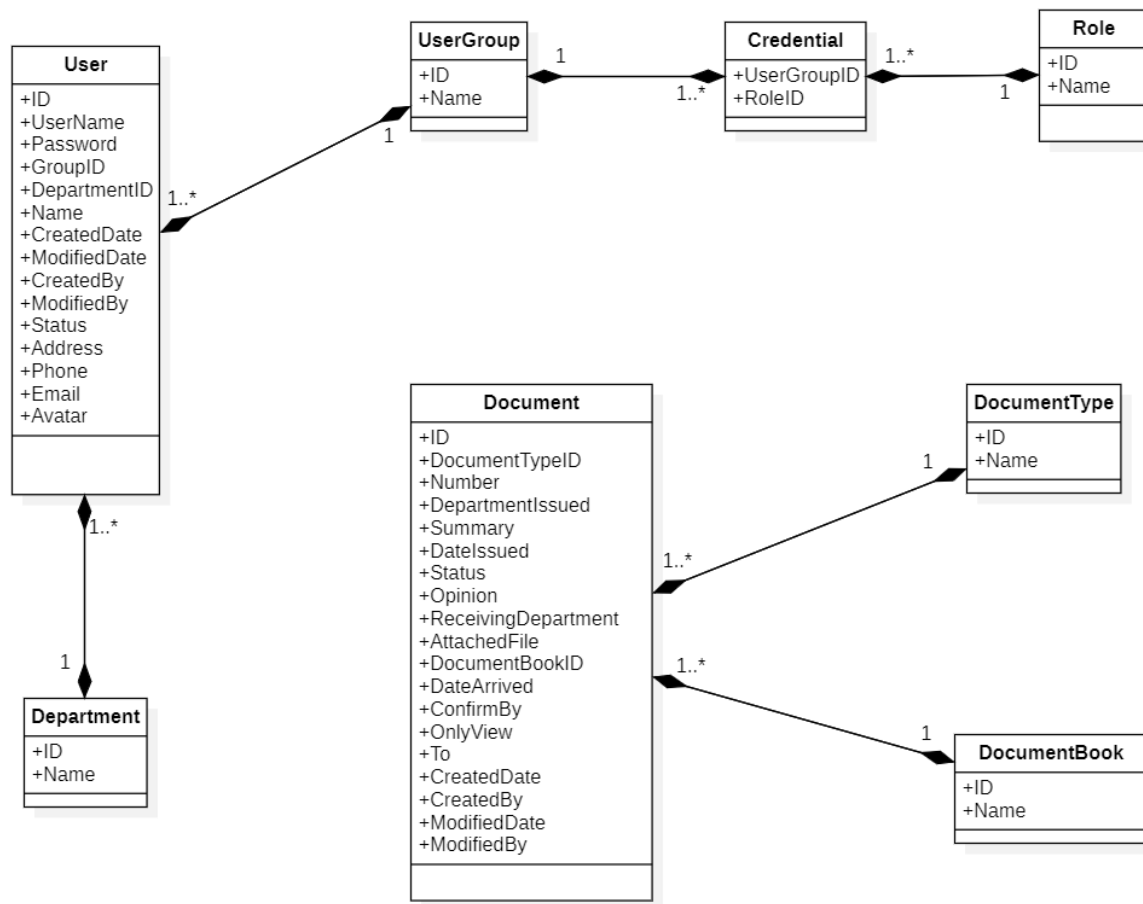


Hình 3.11: Biểu đồ Usecase thống kê.

Bảng 3.9: Kịch bản Usecase thống kê

Tên Use Case	
Tên Actor	
Mô tả	
Tiền điều kiện	
Hậu điều kiện	
Luồng sự kiện chính	
Luồng sự kiện phụ	

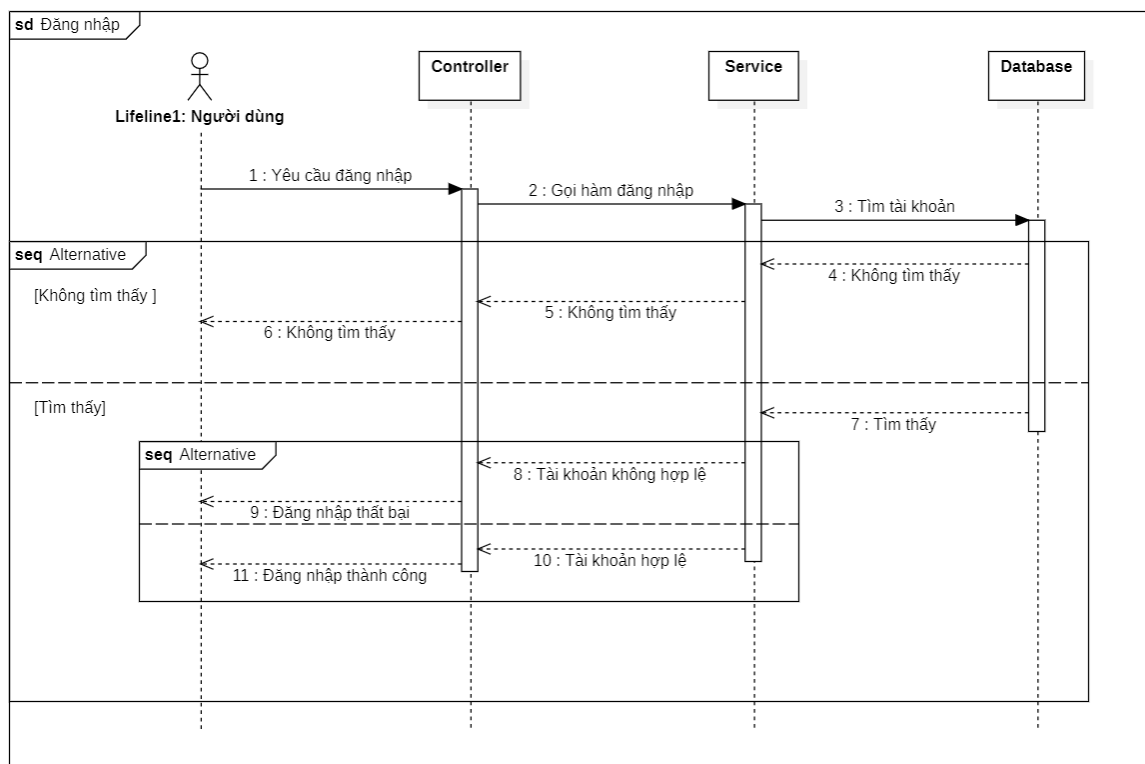
3.2.2. Xây dựng biểu đồ lớp



Hình 3.12: Biểu đồ lớp.

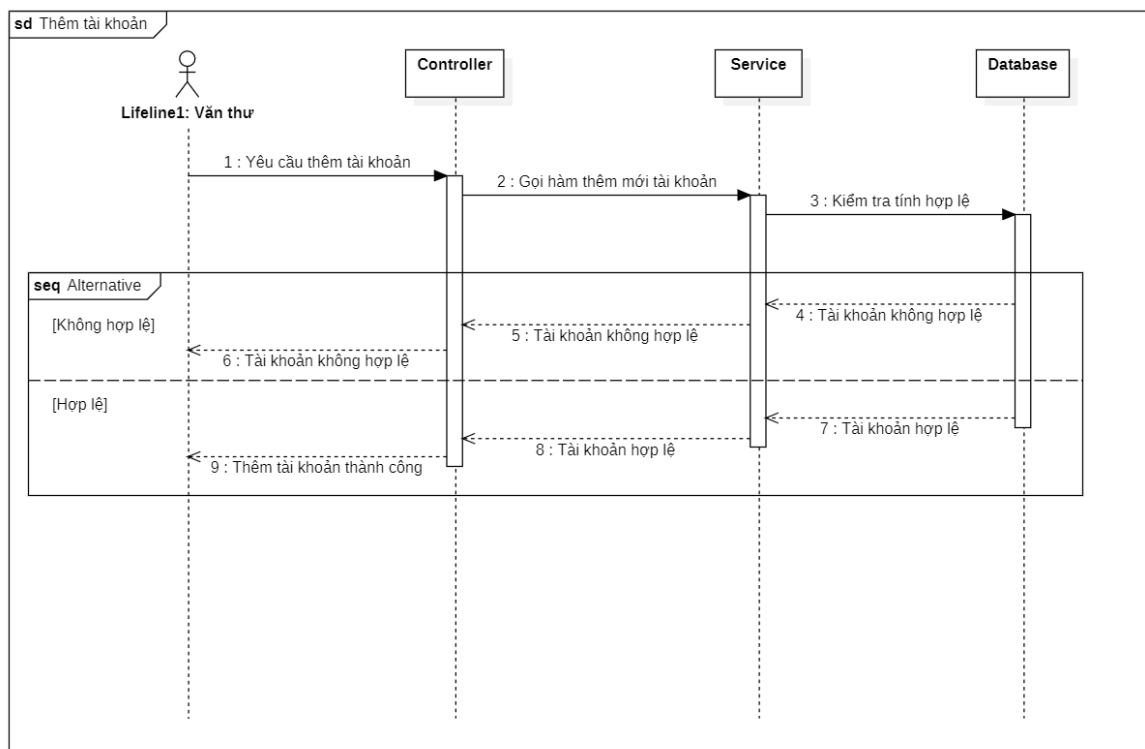
3.2.3. Xây dựng biểu đồ trình tự

3.2.1.1. Biểu đồ trình tự đăng nhập



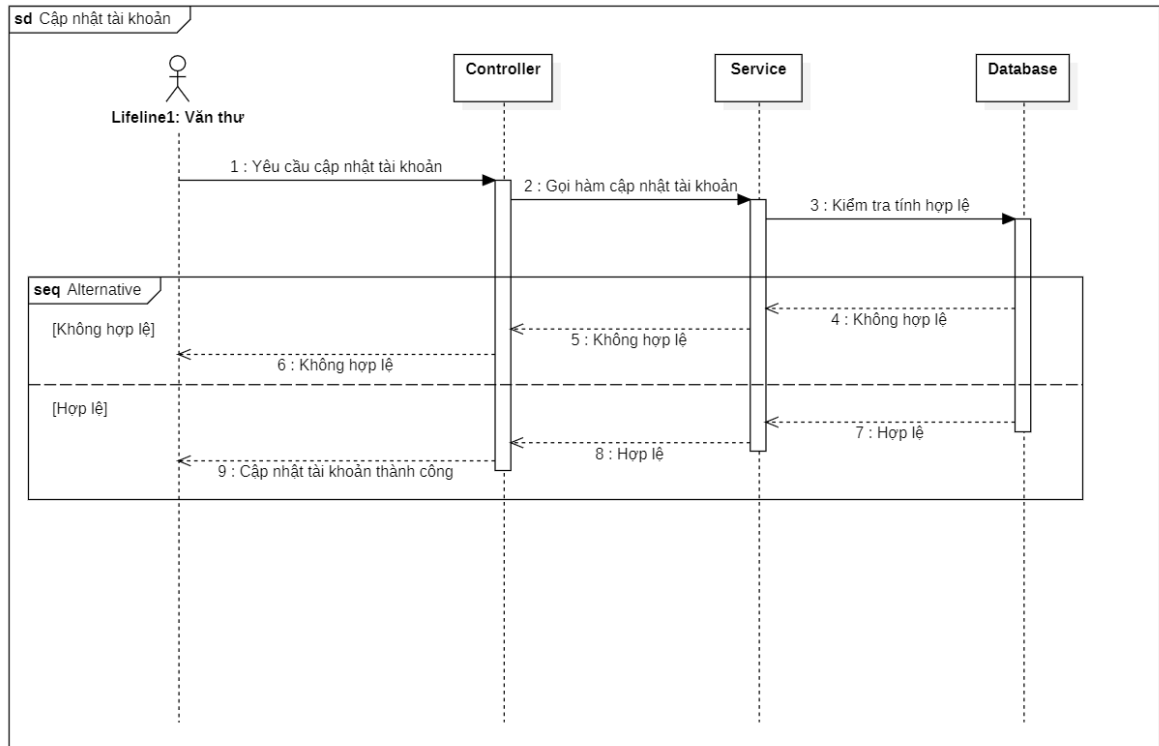
Hình 3.13: Biểu đồ trình tự đăng nhập.

3.2.1.2. Biểu đồ trình tự thêm tài khoản



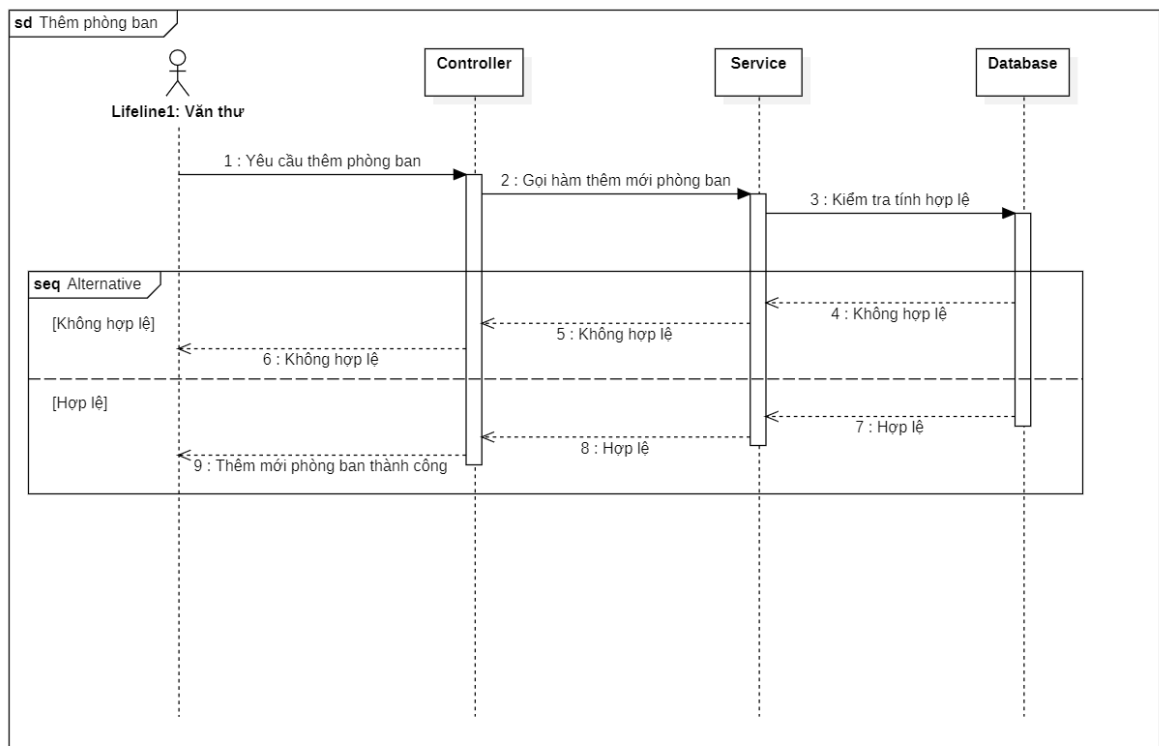
Hình 3.14: Biểu đồ trình tự thêm tài khoản.

3.2.1.3. Biểu đồ trình tự cập nhật tài khoản



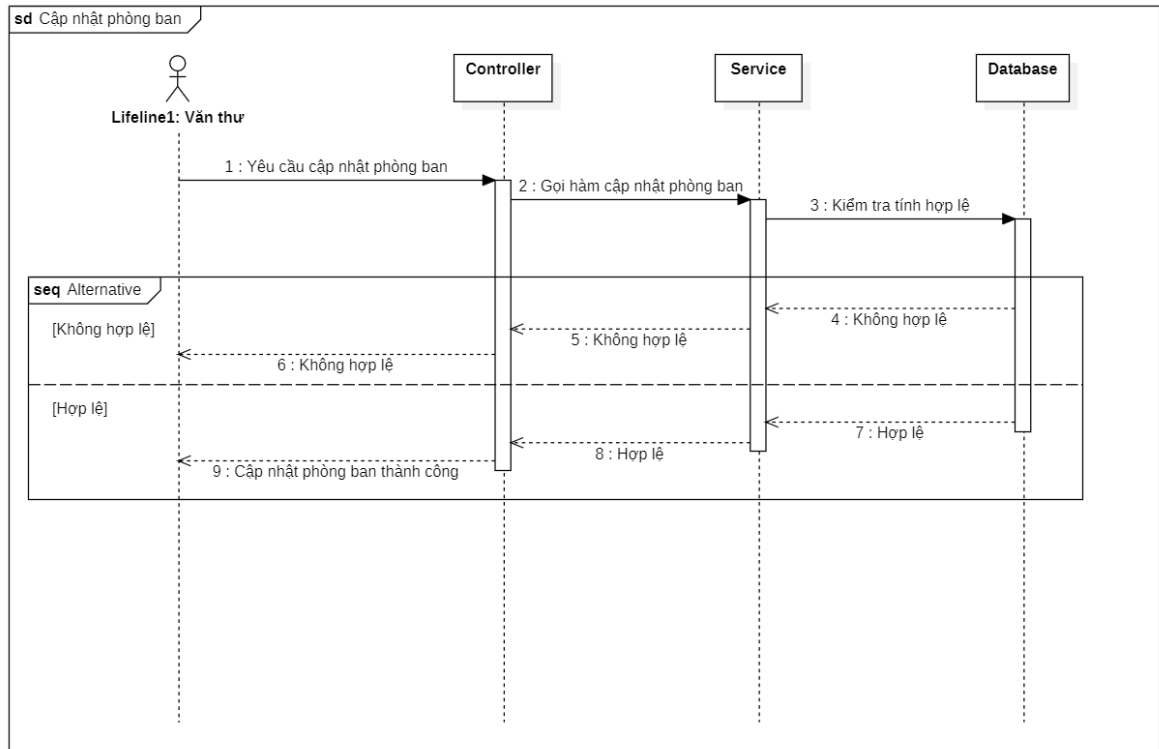
Hình 3.15: Biểu đồ trình tự cập nhật tài khoản.

3.2.1.4. Biểu đồ trình tự thêm phòng ban



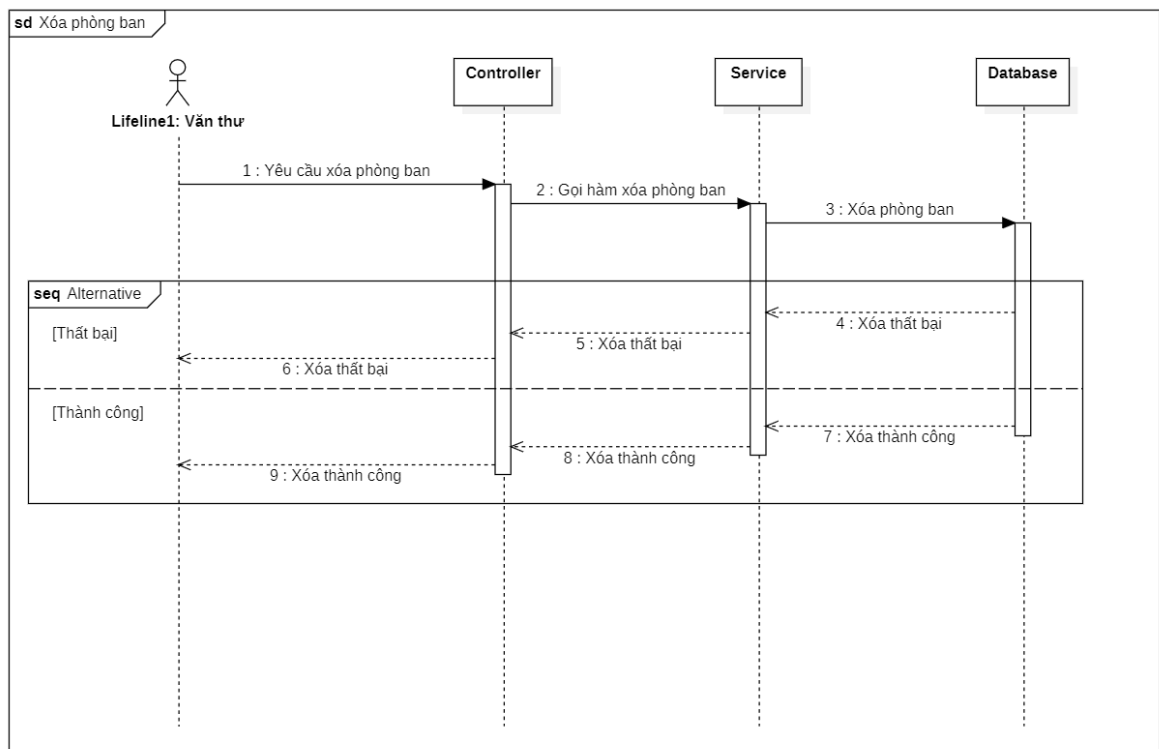
Hình 3.16: Biểu đồ trình tự thêm phòng ban.

3.2.1.5. Biểu đồ trình tự cập nhật phòng ban



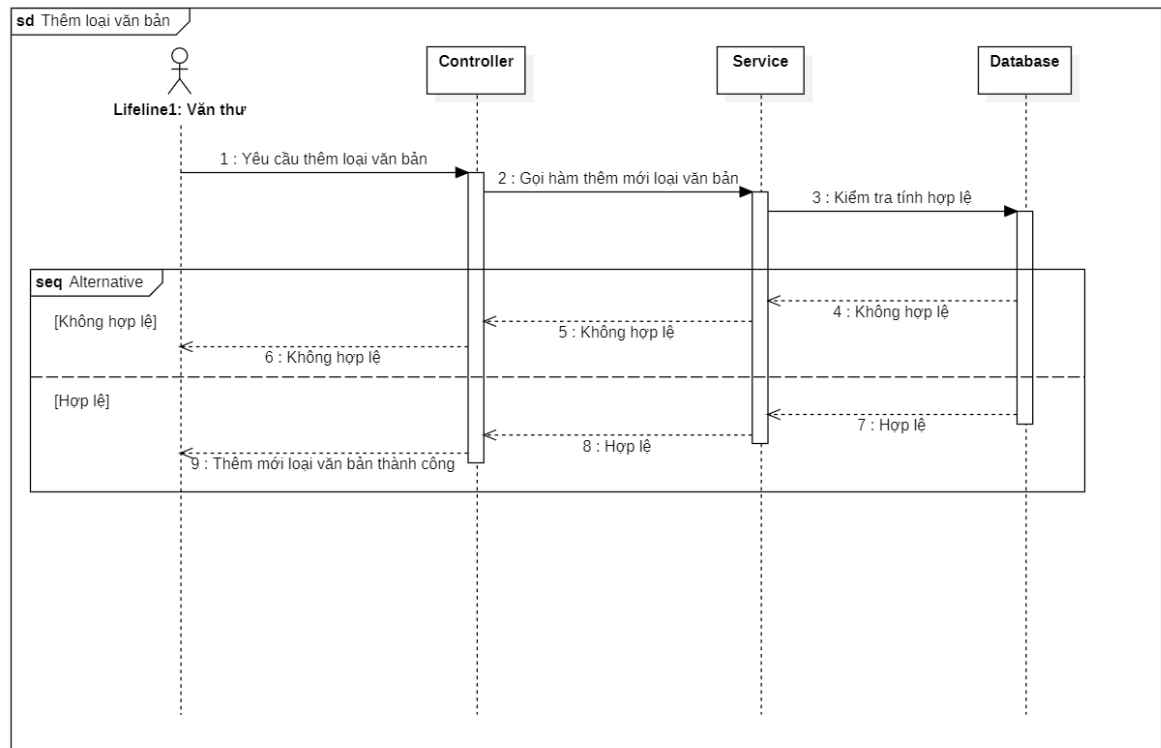
Hình 3.17: Biểu đồ trình tự cập nhật phòng ban.

3.2.1.6. Biểu đồ trình tự xóa phòng ban



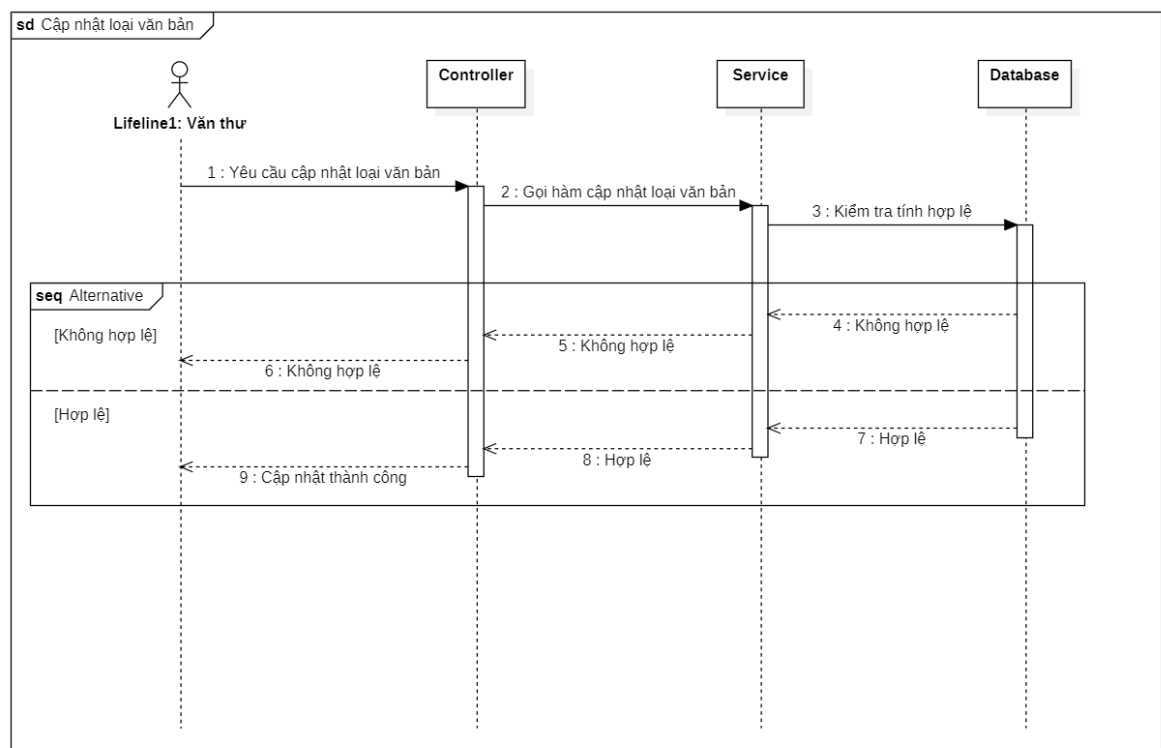
Hình 3.18: Biểu đồ trình tự xóa phòng ban.

3.2.1.7. Biểu đồ trình tự thêm loại văn bản



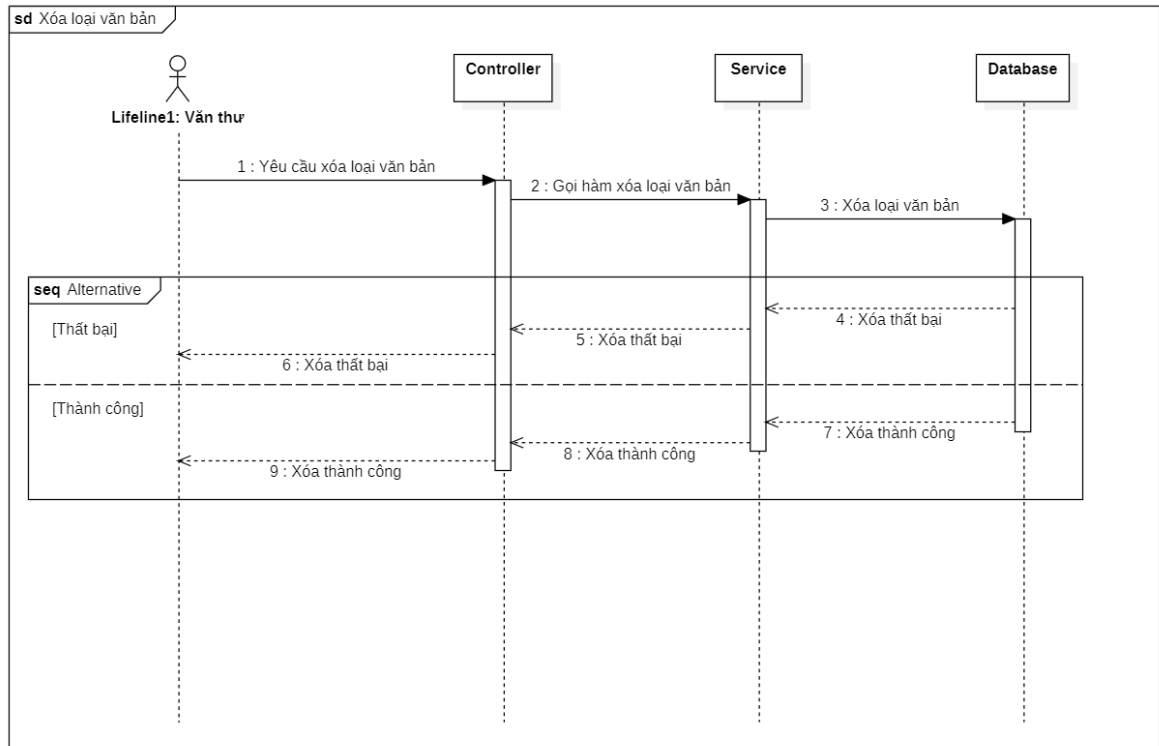
Hình 3.19: Biểu đồ trình tự thêm loại văn bản.

3.2.1.8. Biểu đồ trình tự cập nhật loại văn bản



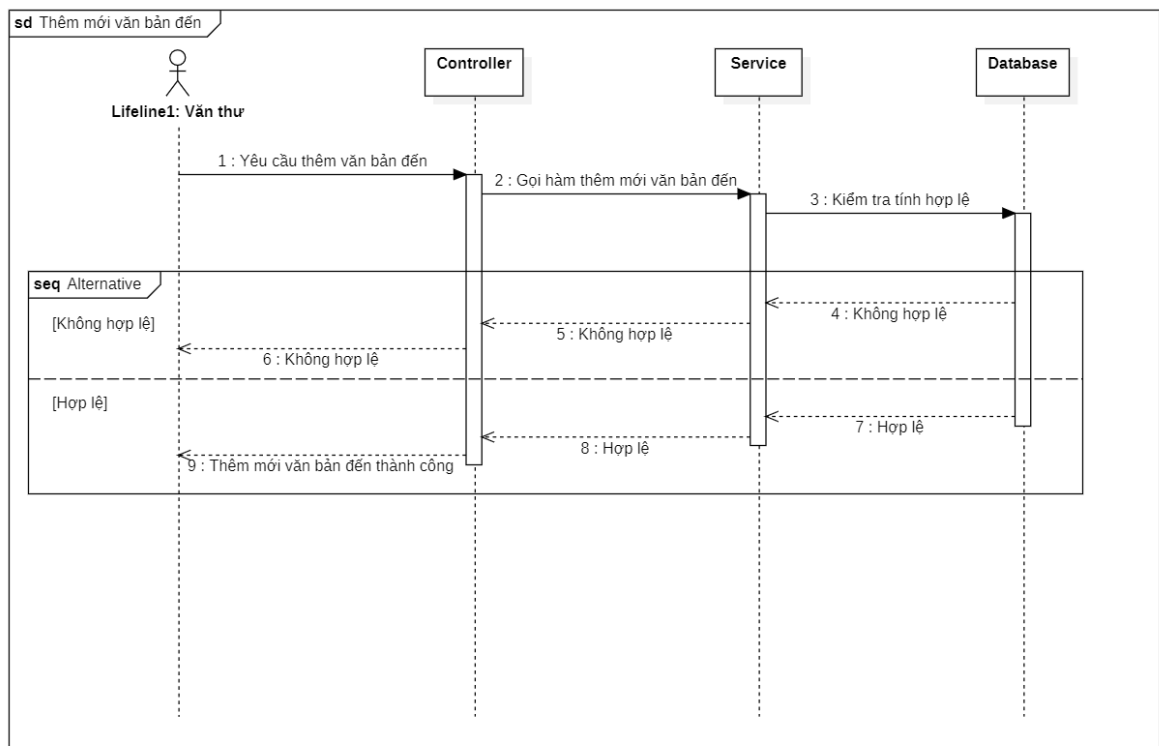
Hình 3.20: Biểu đồ trình tự cập nhật loại văn bản.

3.2.1.9. Biểu đồ trình tự xóa loại văn bản



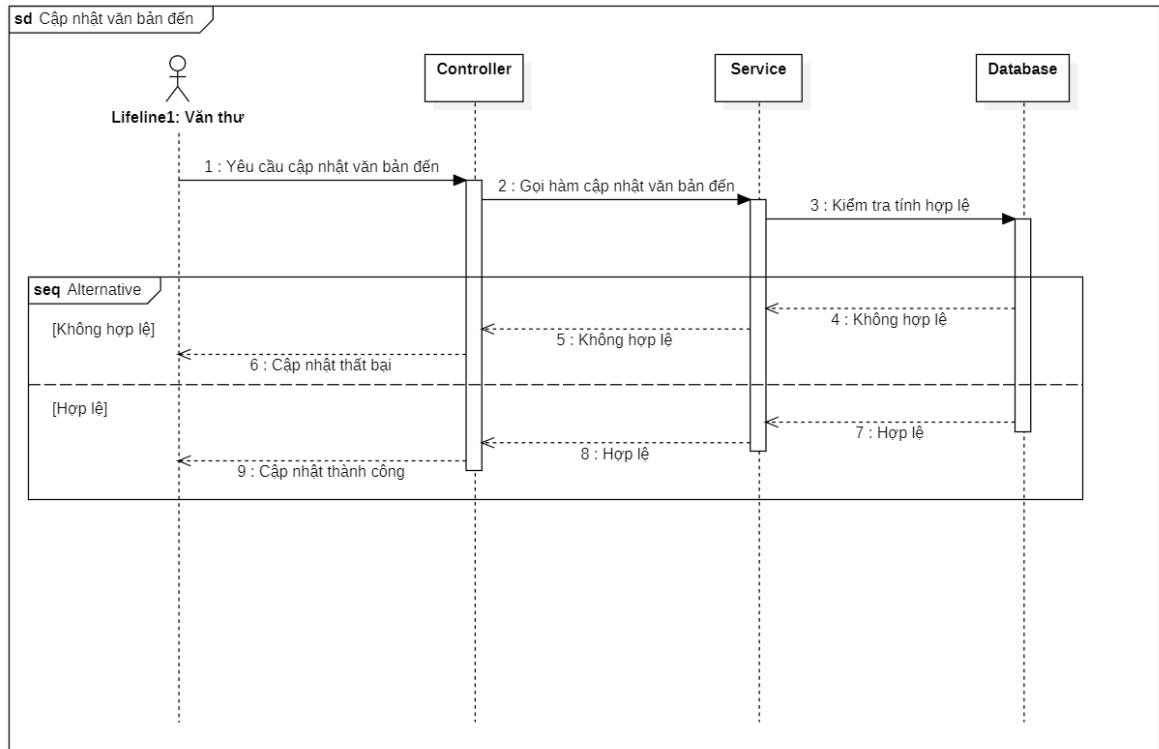
Hình 3.21: Biểu đồ trình tự xóa loại văn bản.

3.2.1.10. Biểu đồ trình tự thêm văn bản đến



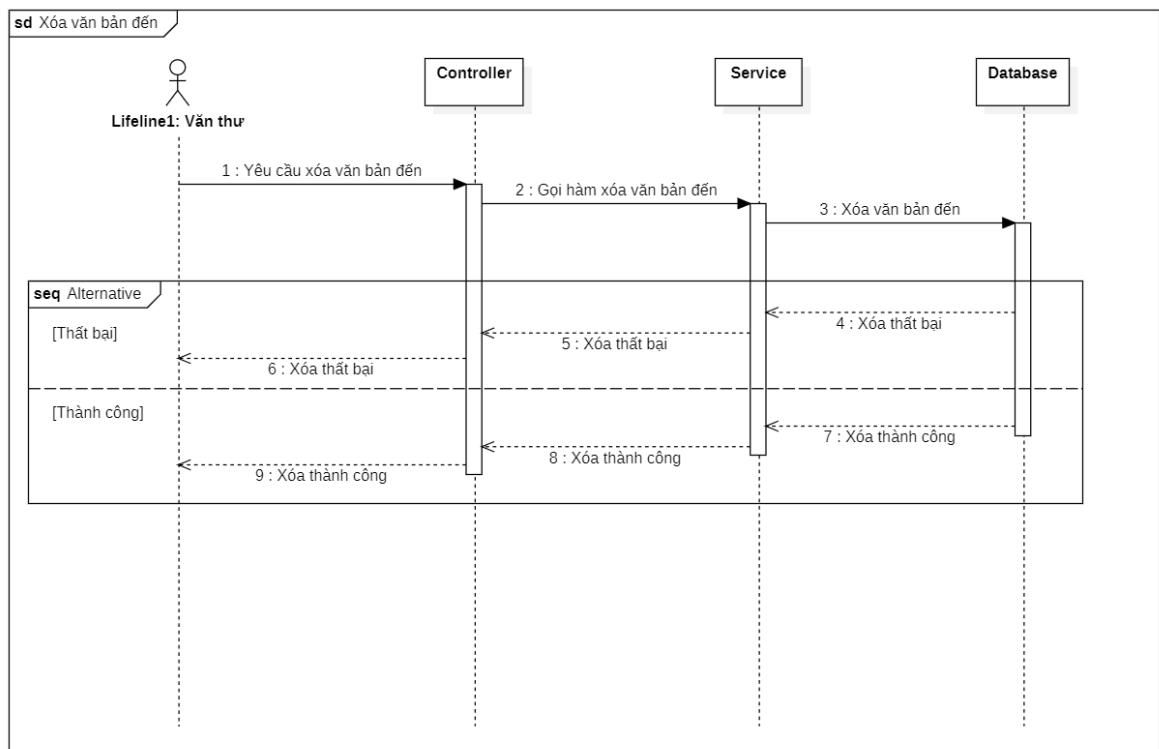
Hình 3.22: Biểu đồ trình tự thêm văn bản đến.

3.2.1.11. Biểu đồ trình tự cập nhật văn bản đến



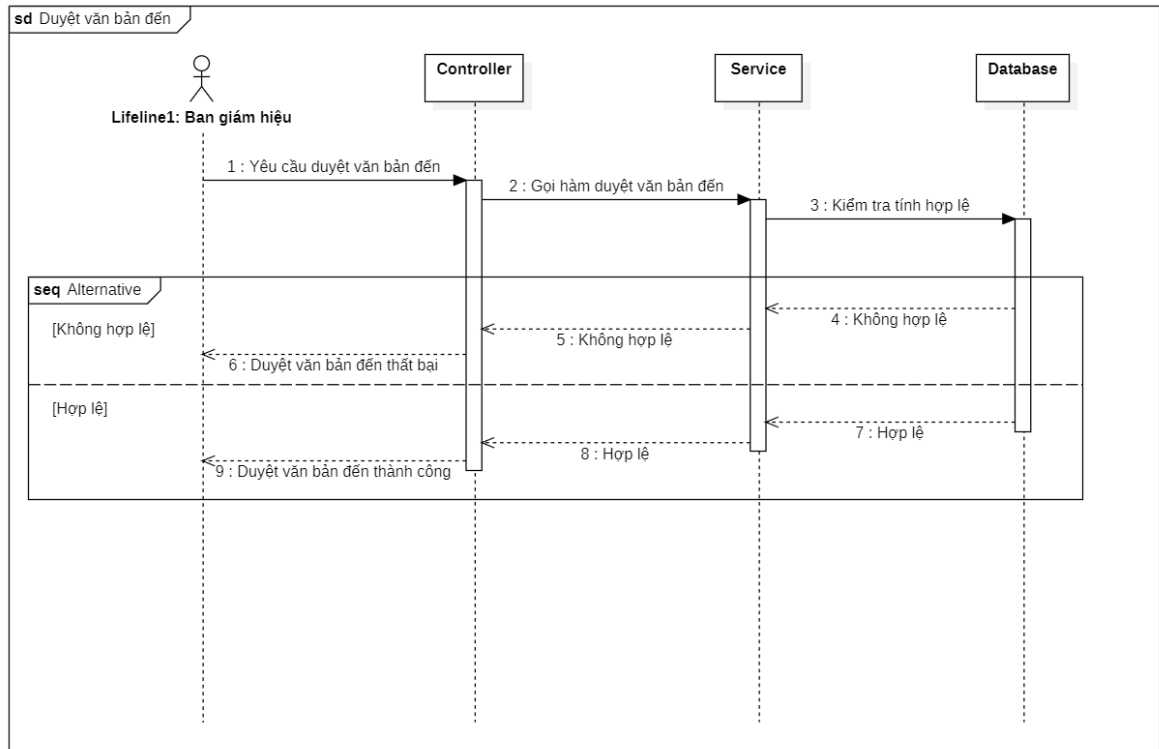
Hình 3.23: Biểu đồ trình tự cập nhật văn bản đến.

3.2.1.12. Biểu đồ trình tự xóa văn bản đến



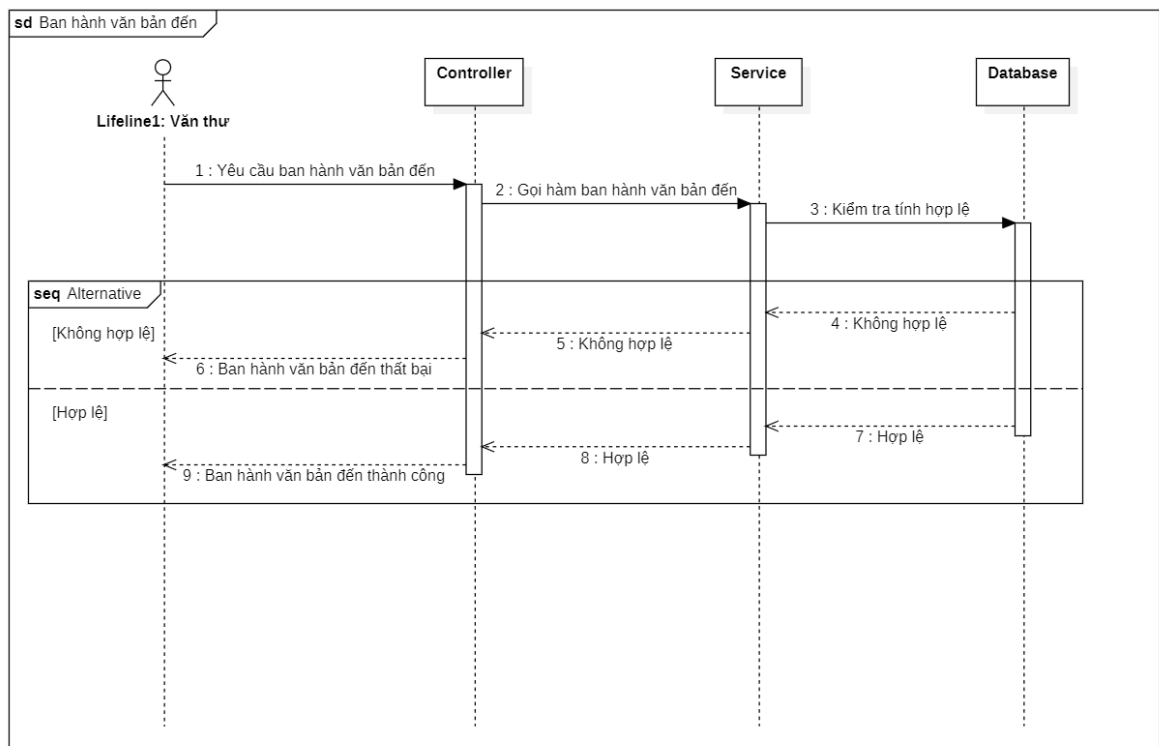
Hình 3.24: Biểu đồ trình tự xóa văn bản đến.

3.2.1.13. Biểu đồ trình tự duyệt văn bản đến



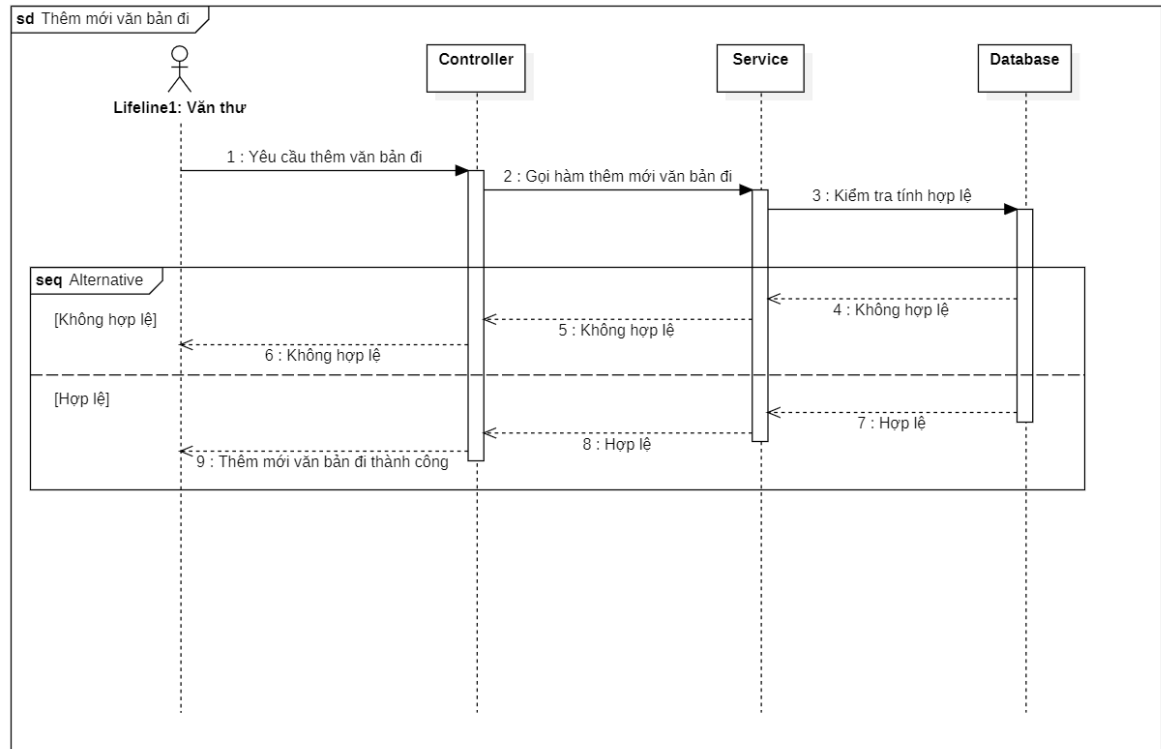
Hình 3.25: Biểu đồ trình tự duyệt văn bản đến.

3.2.1.14. Biểu đồ trình tự ban hành văn bản đến



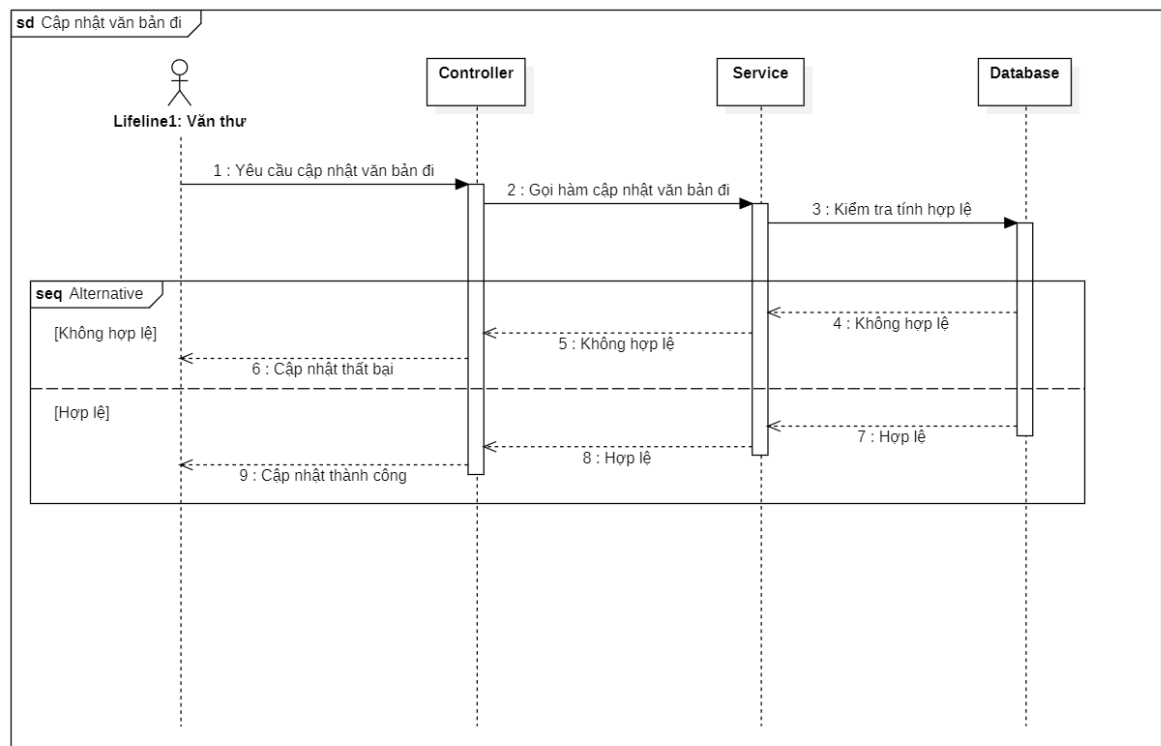
Hình 2.26: Biểu đồ trình tự ban hành văn bản đến.

3.2.1.15. Biểu đồ trình tự thêm văn bản đi



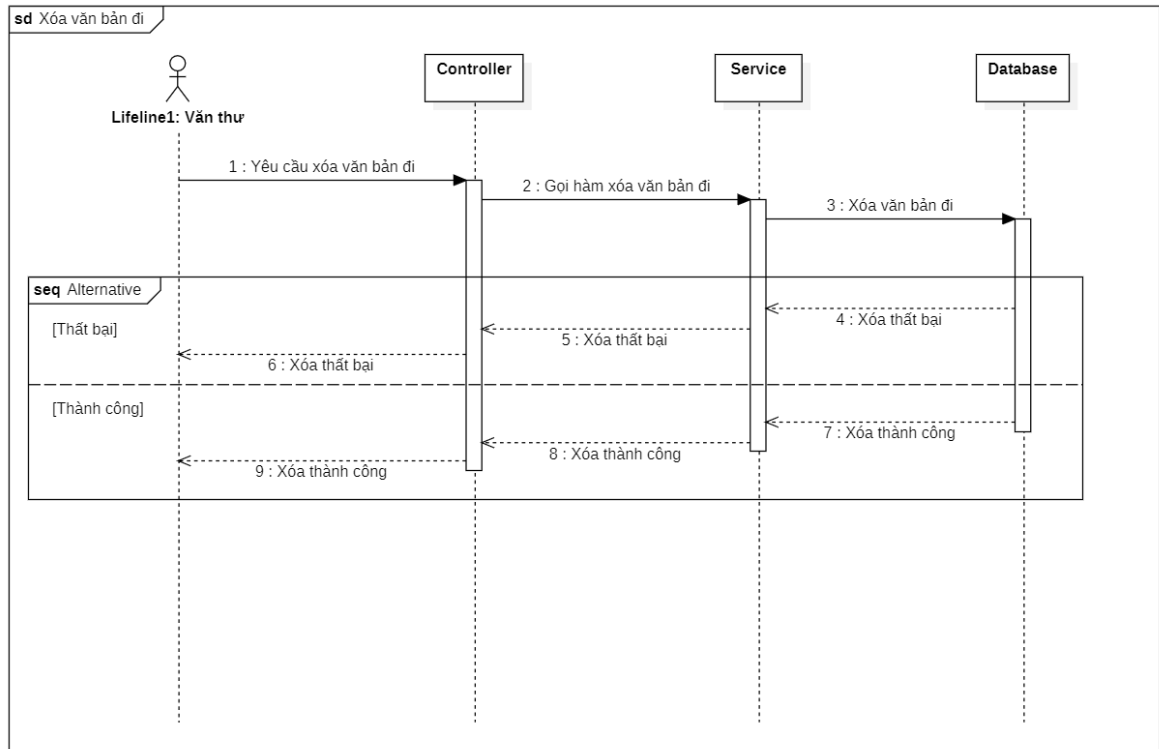
Hình 3.27: Biểu đồ trình tự thêm văn bản đi.

3.2.1.16. Biểu đồ trình tự cập nhật văn bản đi



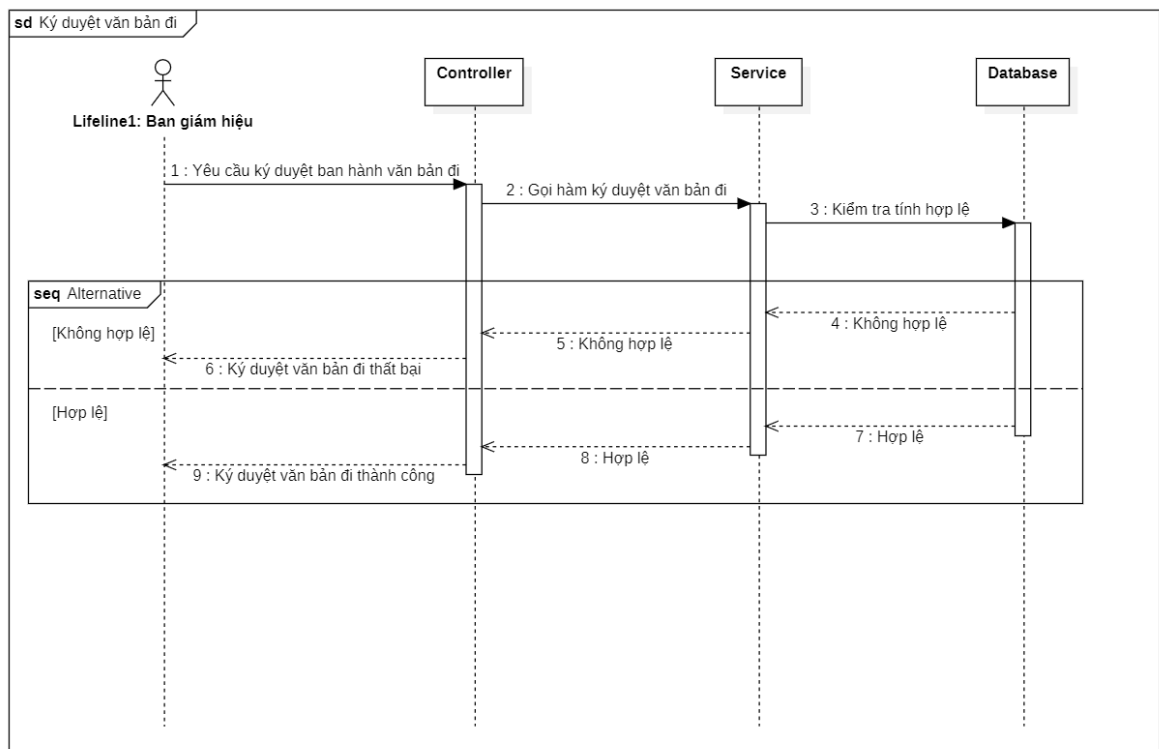
Hình 3.28: Biểu đồ trình tự cập nhật văn bản đi.

3.2.1.17. Biểu đồ trình tự xóa văn bản đi



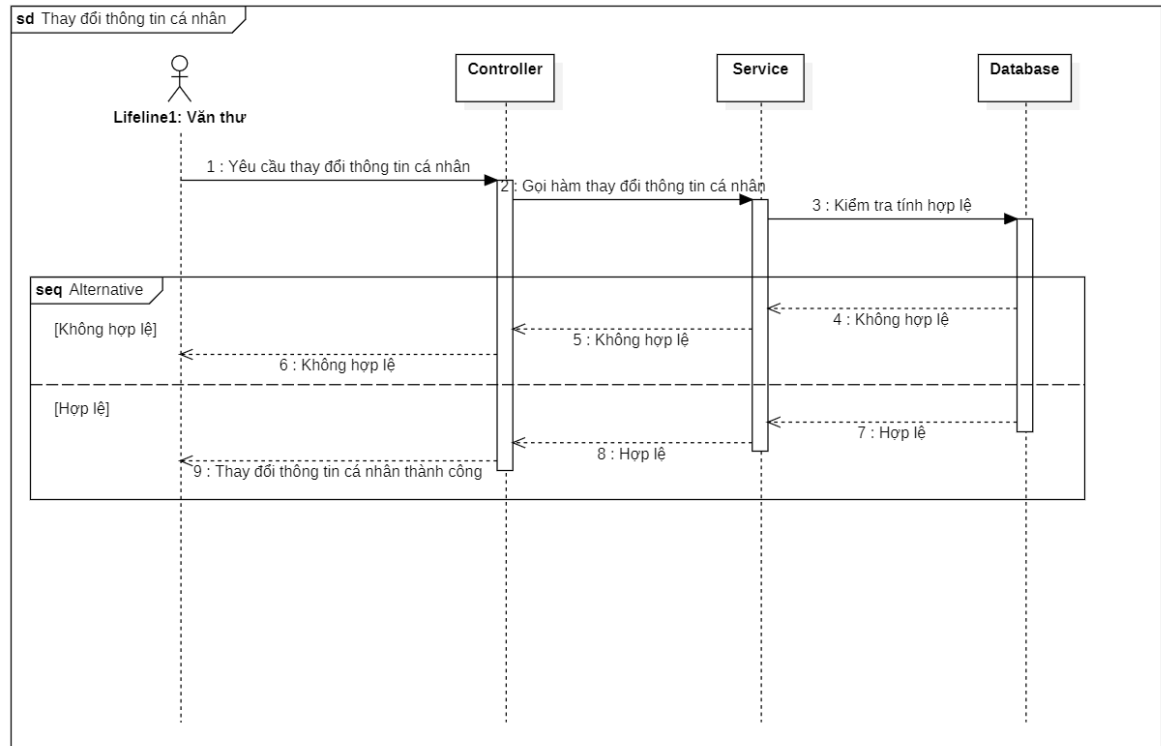
Hình 3.29: Biểu đồ trình tự xóa văn bản đi.

3.2.1.18. Biểu đồ trình tự ký duyệt văn bản đi



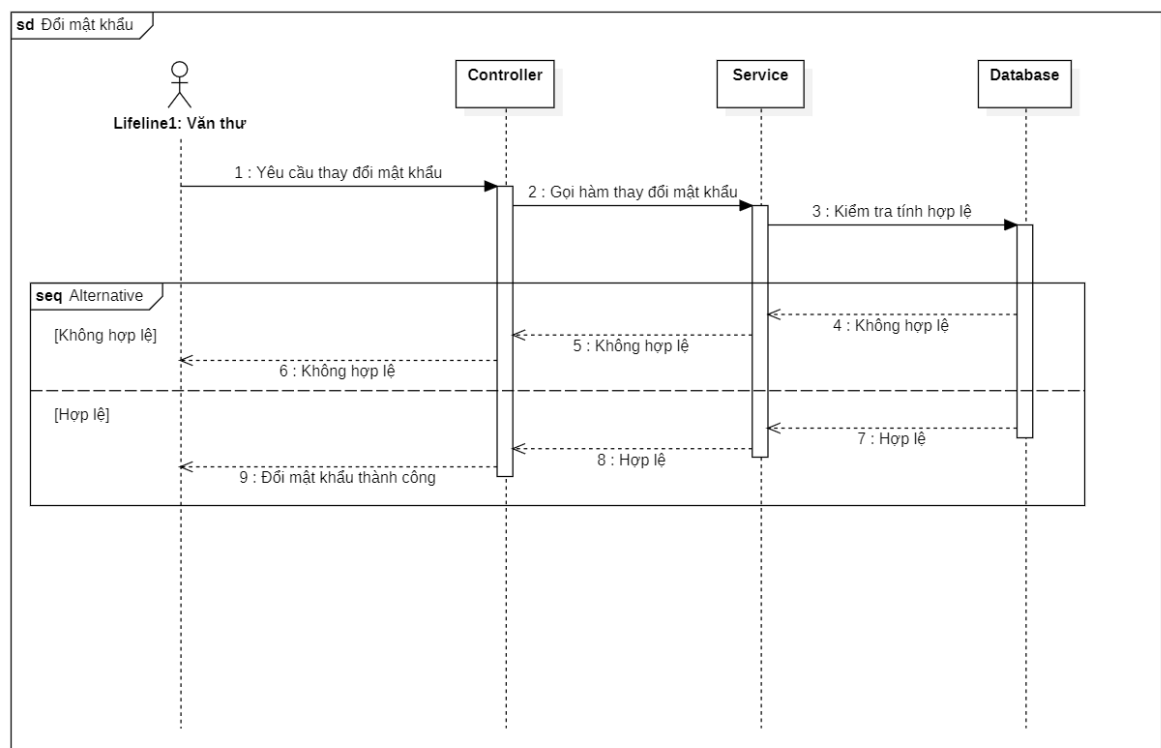
Hình 3.30: Biểu đồ trình tự ký duyệt văn bản đi.

3.2.1.19. Biểu đồ trình tự thay đổi thông tin cá nhân



Hình 3.31: Biểu đồ trình tự thay đổi thông tin cá nhân.

3.2.1.20. Biểu đồ trình tự đổi mật khẩu

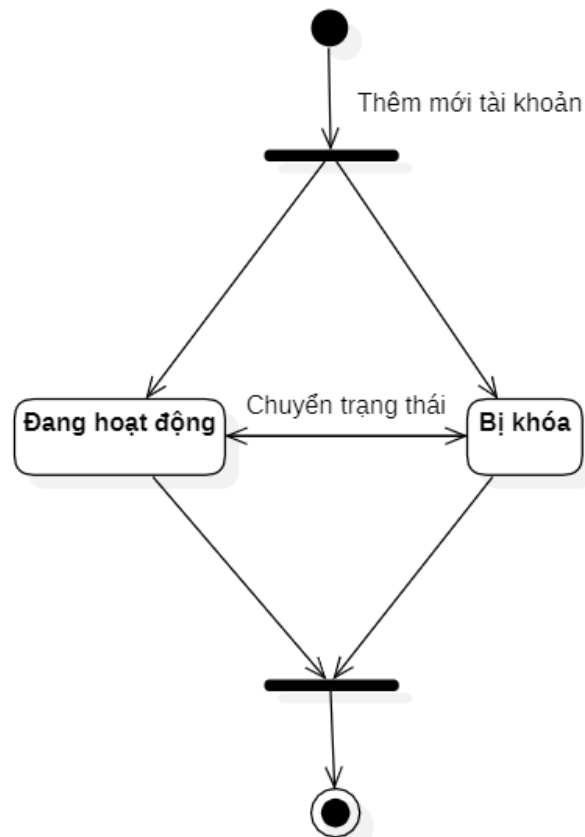


Hình 3.32: Biểu đồ trình tự đổi mật khẩu.

3.2.4. Xây dựng biểu đồ hoạt động

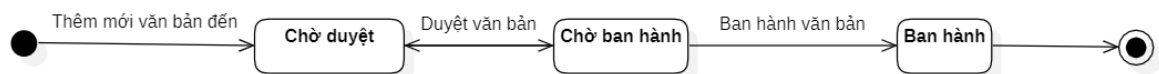
3.2.5. Xây dựng biểu đồ trạng thái

3.2.5.1. Biểu đồ trạng thái tài khoản



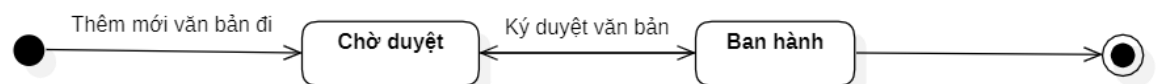
Hình 3.33: Biểu đồ trạng thái tài khoản.

3.2.5.2. Biểu đồ trạng thái văn bản đến



Hình 3.34: Biểu đồ trạng thái văn bản đến.

3.2.5.3. Biểu đồ trạng thái văn bản đi



Hình 3.35: Biểu đồ trạng thái văn bản đi.

3.3. Thiết kế về cơ sở dữ liệu

3.4. Thiết kế các module chương trình chính

3.5. Thiết kế giao diện chương trình

CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH

4.1. Thu thập dữ liệu, xây dựng cơ sở dữ liệu vật lý.

4.2. Viết code cho các module chương trình chính.

4.3. Kiểm thử và đánh giá chương trình.

4.4. Kết luận và hướng phát triển

TÀI LIỆU THAM KHẢO

- [1] Giới thiệu về HTML, CSS, JS: <https://viblo.asia/p/gioi-thieu-html-css-js-WAyK81RnZxX>
- [2] Giới thiệu về ngôn ngữ C#: [https://vi.wikipedia.org/wiki/C_Sharp_\(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh\)](https://vi.wikipedia.org/wiki/C_Sharp_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh))
- [3] Tổng quan về ASP.NET MVC: <https://www.howkteam.vn/course/lap-trinh-website-voi-aspnet-mvc-co-ban/gioi-thieu-tong-quan-cong-nghe-web-aspnet-mvc-123>
- [4] Tổng quan về Entity Framework: <https://chiasekinang.com/entity-framework-la-gi/>
- [5] Tổng quan về MSSQL Server: <https://help.pacisoft.com/knowledgebase/sql-server-la-gi-sql-server-dung-lam-gi-co-may-phien-ban/>
- [6] Khái niệm về UML: https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_m%C3%B4_h%C3%ACnh_h%C3%B3a_th%E1%BB%91ng_nh%E1%BA%A5t
- [7] Tổng quan về Visual Studio: <http://bugnetproject.com/visual-studio-la-gi/>
- [8] Khái niệm về Star UML: <https://viblo.asia/p/staruml-cai-dat-va-trai-nghiem-4P856dn1ZY3>
- [9] Khái niệm về Visual Studio Code: <https://cuongquach.com/visual-studio-code-la-gi.html>

NHẬN XÉT ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

cho sinh viên bảo vệ trước hội đồng chấm đồ án tốt nghiệp.

Hà Nội, ngày tháng năm 2021

Giảng viên hướng dẫn

CHỖ
CÁI
KÝ