**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**THIẾT KẾ WEBSITE HỖ TRỢ GIẢNG DẠY  
CHO GIẢNG VIÊN**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | TS. Nguyễn Đình Dương |
| **Sinh viên:** | Nguyễn Văn Linh |
| **Lớp:** | CNTT3-K59 |
| **Mã sinh viên:** | 181202589 |

**MỤC LỤC**

[CHƯƠNG I. TỔNG QUAN 5](#_Toc122469400)

[I. Đặt vấn đề 5](#_Toc122469401)

[II. Nhiệm vụ của đồ án 5](#_Toc122469402)

[1. Mục đích 5](#_Toc122469403)

[2. Yêu cầu 5](#_Toc122469404)

[3. Môi trường phát triền 6](#_Toc122469405)

[III. Giới thiệu công nghệ 6](#_Toc122469406)

[1. VueJS 6](#_Toc122469407)

[2. DotNet Core 8](#_Toc122469408)

[3. PostgreSQL 11](#_Toc122469409)

[CHƯƠNG II. PHÂN TÍCH 14](#_Toc122469410)

[I. Xác định yêu cầu 14](#_Toc122469411)

[1. Yêu cầu chức năng 14](#_Toc122469412)

[2. Yêu cầu phi chức năng 14](#_Toc122469413)

[II. Đặc tả phần mềm 14](#_Toc122469414)

[1. Phát biểu bài toán 15](#_Toc122469415)

[2. Mô hình Use Case 16](#_Toc122469416)

[CHƯƠNG III. THIẾT KẾ CƠ SỞ DỮ LIỆU 19](#_Toc122469417)

[I. Sơ đồ quan hệ giữa các bảng 19](#_Toc122469418)

[1. Chi tiết các bảng 20](#_Toc122469419)

[CHƯƠNG IV. HIỆN THỰC CHƯƠNG TRÌNH 23](#_Toc122469420)

**LỜI CẢM ƠN**

Trong suốt thời gian học tập, rèn luyện tại trường Đại học Giao Thông Vận Tải, em xin chân thành cảm ơn tới các thầy/cô giáo trong nhà trường vì đã tạo điều kiện thuận lợi nhất cho chúng em trong việc học tập và nghiên cứu.

Em cũng xin gửi lời cảm ơn chân thành tới tất cả các thầy cô giáo trong Khoa CNTT và các thầy cô bộ môn liên quan đã giảng dạy và truyền đạt lại những kiến thức bổ ích cho chúng em trong suốt thời gian học tập tại trường, giúp em có thêm nhiều kiến thức để sẵn sàng có thể bước tiếp trong con đường sau này

Đặc biệt em xin gửi lời chân thành cảm ơn thầy Nguyễn Đình Dương, trong suốt thời gian làm đồ án tốt nghiệp vừa qua, thầy đã giành nhiều thời gian để theo sát và hướng dẫn em hoàn thành đề tài.

Mặc dù đã cố gắng hoàn thành đồ án trong những phạm vi và khả năng cho phép, nhưng cũng không thể tránh khỏi các thiếu sót. Rất mong nhận được sự cảm thông cũng như sự đóng góp, chỉ dẫn của quý Thầy Cô và các bạn.

***Em xin chân thành cảm ơn !***

Hà Nội, ngày… tháng … năm 2022

Sinh viên thực hiện

Nguyễn Văn Linh

**GIỚI THIỆU**

Trong những năm gần đây cùng với sự ra đời và phát triển nhanh chóng của máy tính điện tử, một công cụ lưu trữ và xử lý thông tin đã làm tăng khả năng nghiên cứu và vận hành hệ thống lớn phức tạp. Hệ thống thông tin là một trong những nghành mũi nhọn của công nghệ thông tin đã có những ứng dụng rất quan trọng trong quản lý.

Tuy nhiên hiện nay đối với một hệ thống lớn việc vận dụng ngay các phần mềm chuyên dụng còn là một vấn đề gặp không ít khó khăn. Các hệ thống thông tin tin học hoá chưa đáp ứng được yêu cầu của các nhà quản lý. Để đáp ứng được yêu cầu cấp thiết đó thì đòi hỏi phải có một đội ngũ cán bộ chuyên môn có đủ trình độ để phân tích hệ thống quản lý hệ thống quản lý một cách đầy đủ chi tiết mà không bị thiếu sót hay thừa thông tin. Từ đó thiết kế hệ thống thành những chương trình thuận tiện trong quá trình làm việc như: tìm kiếm, nhập liệu, thống kê,…

Việc ứng dụng tin học trong công tác quản lý đã phát triển mạnh mẽ giúp cho công tác quản lý ngày càng trở nên hiệu quả hơn như nâng cao hiệu quả trong công việc, đưa ra các báo cáo, các số liệu thống kê một cách chính xác kịp thời. Đồng thời nhờ có việc ứng dụng tin học đã tiết kiệm được rất nhiều thời gian, công sức của con người.

Song song với lĩnh vực công nghệ thông tin, trong công cuộc đổi mới đất nước, nghành Giáo dục và Đào tạo cũng là một ngành rất quan trọng trong xu thế hiện nay. Trong các công tác quản lí của các trường phổ thông trung học như việc quản lý lịch giảng dạy cho các giáo viên cũng là một vấn đề được quan tâm vì nó quyết định công tác giảng dạy và học tập của giáo viên, học sinh. Quản lí lịch giảng dạy của các trường phổ thông trung học phải có sự hợp lí và cần thiết sao cho hệ thống quản lí phải thật gọn nhẹ, dễ hiểu đối với người sử dụng.

Trong kỳ làm đồ án tốt nghiệp, được TS. Nguyễn Đình Dương hướng dẫn trong việc xây dựng chương trình “Hỗ trợ giảng dạy cho giảng viên” là một cơ hội để em tìm hiểu sâu về việc quản lý. Do hạn chế về mặt thời gian cũng như kiến thức và kinh nghiệm quản lý nên em không thể tránh khỏi những khiếm khuyết và sai sót. Em rất mong được sự giúp đỡ và ý kiến đóng góp của thầy cô và các bạn để em tiếp tục hoàn thành đề tài này được hoàn thiện hơn.

1. TỔNG QUAN
2. Đặt vấn đề

Trong hầu hết các trường học hiện nay, việc lên lớp của giảng (GV) viên phụ thuộc vào lịch dạy được phòng đào tạo phân lớp và sắp xếp thời khóa biểu. Việc thay đổi lịch dạy hạn chế tối đa trừ các trường hợp bắt buộc. Khi GV thay đổi lịch dạy cần thông báo cho bộ môn, phòng thanh tra, ban quản lý giảng đường để dịch chuyển lịch dạy cho phù hợp, việc này mất khá nhiều thời gian, đồng thời cũng cần thông báo kịp thời cho sinh viên để sinh viên có thể chủ động nắm bắt và sắp xếp thời gian biểu sao cho phù hợp

Hơn nữa, với những GV dạy nhiều lớp trong một học kỳ, việc báo nghỉ và dạy bù cũng gây khó khăn như nhầm lịch, trùng lịch trong một số trường hợp.

Trong rất nhiều các phương thức giao tiếp hiện nay thì giao tiếp thông qua Internet là một trong những phương thức phổ biến và tiện lợi nhất. Việc xây dựng website “Hỗ trợ giảng dạy cho giảng viên” với mục đích nhằm tạo một công cụ hiệu quả giúp cho việc quản lý TKB, báo nghỉ, dạy bù được dễ dàng và nhanh chóng thông qua môi trường Internet

1. Nhiệm vụ của đồ án
2. Mục đích

* Xây dựng website cho phép giảng viên quản lý TKB, báo nghỉ, đăng ký dạy bù, báo cáo số tiết học đã báo nghỉ và dạy bù giúp giảng viên nắm được thông tin về tình hình giảng dạy nhanh chóng, thuận tiện
* Tìm hiểu về kỹ thuật lập trình web với JS, C# và ASP.NET, SQL Server, mô hình ba lớp trong lập trình và cách thức tổ chức mã nguồn trong Dự án.

1. Yêu cầu

Website hỗ trợ giảng dạy phải đạt các tính năng sau:

* Giảng viên có thể xem TKB của mình
* Giảng viên có thể import lịch dạy vào phần mềm từ file excel tránh việc nhập tay gây mất thời gian và công sức
* Giảng viên có thể đăng ký dạy bù/báo nghỉ dạy nhưng phải tuân thủ các quy tắc
* Giảng viên có thể xem danh sách báo cáo những tiết học dạy bù/báo nghỉ
* Giảng viên có thể export báo cáo ra file excel để thuận tiện theo
* Phần mềm tự động gửi thông báo cho tất cả sinh viên khi Giảng viên đăng ký dạy bù/báo nghỉ

1. Môi trường phát triền

Trong quá trình tìm hiểu và xây dựng đồ án, em đã chọn thực hiện website trên nền tảng ứng dụng mã nguồn mở với JS, C# và .NET .Với ưu điểm mã nguồn được tổ chức rất tốt không những giúp em có thể tìm hiểu và phát triển website của mình trên đó mà còn là một trong những tài liệu tuyệt vời giúp cho việc ôn lại những kiến thức trên lớp và thực hành về mô hình lập trình ba lớp với C# và ASP.NET

* Môi trường phát triển: MS Visual Studio 2022, Visual Studio Code
* Ngôn ngữ lập trình: C#, JavaScript
* Hệ quản trị CSDL: Postgresql
* Nền tảng hỗ trợ: .NET 6.0, .NET Framework, VueJs
* Thiết kế đồ họa: Adobe Photoshop CC2022.
* Môi trường thực thi ứng dụng: Windows/IIS với MS Framework

1. Giới thiệu công nghệ
2. VueJS

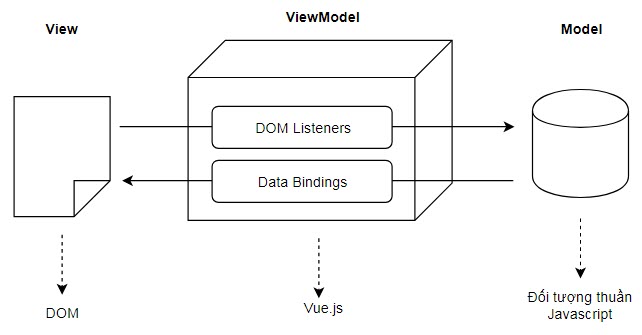
Hình 1. 1 VuejS

1. Vue.js là gì?

Vue (phiên âm /vjuː/, đọc giống như từ view) là một framework Javascript tiên tiến trong xây dựng giao diện người dùng, không giống như các framework khác, Vue được xây dựng từ những dòng code cơ bản nhất nhằm tối ưu tốc độ. Thư viện của Vue chỉ tập trung vào lớp hiển thị, rất đơn giản để tiếp cận và dễ dàng tích hợp vào các hệ thống khác. Vue cũng có khả năng cung cấp các ứng dụng web đơn trang Single Page Application (SPA) cho phép kết hợp với nhiều các công cụ hiện đại như Laravel chẳng hạn.

Vue.js mới chỉ ra mắt năm 2015, nhưng đã sớm khẳng định mình và sớm trở thành người thay thế Angular và React

1. Mô hình MVVM

MVVM là viết tắt của Model-View-ViewModel là một mô hình được áp dụng trong framework Vue.js.

Trong mô hình này, dữ liệu mỗi khi được "gán" vào View hoặc Model sẽ đều được Vue.js tự động gắn cho phần còn lại. Tức là khi dữ liệu thay đổi ở Model nó sẽ tự động được "cập nhật" sang View và khi người dùng thay đổi dữ liệu trên View (ví dụ nhập liệu vào ô địa chỉ email chẳng hạn) thì dữ liệu cũng được tự động cập nhật ngược lại Model. Trong cộng đồng Vue.js thường gọi mô hình này với một thuật ngữ khác là two-way data binding, tạm gọi là gán dữ liệu hai chiều.

1. Kiến thức cần thiết cho Vue.js

Sử dụng Vue.js nói riêng hay phát triển các ứng dụng web nói chung đòi hỏi khá nhiều các kiến thức cơ bản khác nhau, dưới đây là các kiến thức bạn **nên** có khi làm việc với Vue.js nên tìm hiểu: HTML, CSS, JS, JQUERY

1. So sánh Vue.js với các framework cùng dạng

Hiện nay có rất nhiều các framework Javascript mạnh mẽ, kể đến có React (hậu thuẫn bởi Facebook), Angular (hậu thuẫn bởi Google), Ember, Riot, Polymer... tuy nhiên Vue.js đang là sự lựa chọn lý tưởng cho các ứng dụng web ở mức vừa và lớn. Có các lý do như sau:

* Hiệu năng Vue.js là thực sự đáng nể so với các đối thủ khác.
* Vue.js có dung lượng tải cực bé do đã tách một số phần ra khỏi core như Vuex, vue-router... nó giúp cải thiện tốc độ tải của toàn trang.
* Một lý do mà nhiều người tìm đến với Vue.js là tính đơn giản, dễ học, dễ áp dụng đặc biệt cho những người chưa có nhiều kiến thức nền.
* Vue.js luôn học hỏi và tích hợp những gì tốt nhất từ các framework khác, nó cũng giống như framework Laravel nổi tiếng, luôn thay đổi và luôn tốt hơn.

1. DotNet Core

Hình 1. 2 DotNet Core

1. .NET core là gì?

.NET core là phiên bản mới của  .NET framework. Nó là một nền tảng miễn phí và mã nguồn mở được phát triển, duy trì bởi Microsoft.

.NET core là một framework đa nền tảng chạy trên các hệ điều hành: Windows, Mac OS và Linux. Nó được dùng để xây dựng ứng dụng trên các thiết bị khác nhau như mobile, desktop, web, cloud, máy học tập, game,..

Mặc dù nói .NET core là phiên bản từ .NET framework, nhưng thực tế nó được viết lại từ đầu để đem lại những ưu điểm như nhanh, nhẹ và đa nền tảng.

1. Asp.Net Core là gì?

Khi tìm hiểu về .NET core, nhiều người thường thắc mắc **ASP.NET Core là gì**?

 ASP.NET Core cũng là một framework -là phiên bản cải biến mới của .NET Core -được phát triển lại từ đầu với mục tiêu chạy trên nền tảng .NET Core.

Nó được thiết kế để cung cấp khả năng tối ưu tốt hơn cho các ứng dụng dựa trên kết nối cloud (đám mây). Ví dụ như: ứng dụng web, Internet of Things và backend cho điện thoại di động,…

1. Lịch sử ra đời của .NET core?

.NET core được ra đời vào ngày 12/11/2014 với niềm tin rằng nó sẽ là nền tảng của tất cả .NET framework trong tương lai (theo Immo Landwerth).

Các phiên bản sau này được ra đời lần lượt:

* .NET core 1.0 và .NET core 1.1 – năm 2016
* .NET core 2.0 và .NET core 2.1 – năm 2017
* .NET core 3.0 và .NET core 3.1 – năm 2019

Đến tháng 11/2020, Microsoft đã phát hành .NET 5.0 thay thế .NET framework. Chữ “Core” bị xóa khỏi tên và phiên bản 4.0 được bỏ qua để tránh nhầm lẫn với .NET framework

1. Đặc điểm của .NET core

Các đặc điểm nổi bật của .NET core:

* Open-source Framework: .NET Core là nền tảng nguồn mở được duy trì bởi Microsoft và có sẵn trên GitHub theo giấy phép MIT và Apache 2. Nó là một dự án .NET Foundation.
* Cross-platform: Đa nền tảng. .NET Core chạy trên hệ điều hành Windows, macOS và Linux. Có thời gian chạy khác nhau cho mỗi hệ điều hành thực thi mã và tạo ra cùng một output.
* Nhất quán giữa các kiến trúc: Bạn có thể thực thi mã giống nhau trong các kiến trúc tập lệnh khác nhau (bao gồm x64, x68 và RAM)
* Hỗ trợ nhiều ngôn ngữ: Bạn có thể sử dụng ngôn ngữ lập trình C#, F# và Visual Basic để phát triển ứng dụng .NET Core.
* Triển khai linh hoạt: Các ứng dụng .NET Core có thể được triển khai song song (cài đặt toàn bộ người dùng hoặc toàn hệ thống). No cũng có thể triển khai với các Container Docker.
* Khả năng tương thích: .NET Core tương thích với .NET Framework và Mono API (thông qua .NET Standard).
* Hỗ trợ bởi Microsoft: Vì .NET Core được phát triển bởi Microsoft nên tài liệu cũng như các cập nhật, hỗ trợ được update thường xuyên. Từ đó, giúp người sử dụng có thể giải quyết vấn đề nhanh chóng.
* Công cụ CLI: .NET core bao gồm các công cụ CLI (Command – line –interface: Giao tiếp thông qua dòng lệnh) để phát triển và tích hợp liên tục.
* Kiến trúc mô-đun: .NET core hỗ trợ cách tiếp cận kiến trúc mô-đun bằng cách sử dụng các gói NuGet. Có các gói NuGet khác nhau cho các tính năng khác nhau có thể được thêm vào dự án .NET core nếu cần. Nhờ đó bạn sẽ ít tiêu tốn dung lượng bộ nhớ, tăng hiệu suất và dễ bảo trì ứng dụng hơn.

1. Bốn thành phần chính của .NET core.

* CLI Tool: Công cụ CLI(Command – line –interface: Giao tiếp thông qua dòng lệnh) là bộ công cụ để phát triển và triển khai .NET core
* Roslyn: Trình biên dịch ngôn ngữ cho C# và Visual Basic
* CoreFX: Tập hợp các thư viện khung (framework library)
* CoreCLR (Command Language Runtime): Môi trường thực thi của .NET core, CoreCLR sử dụng trình biên dịch trung gian dựa trên JIT. Nhờ đó nó có thể dịch mã IL sang mã máy của những nền tảng mà nó hỗ trợ.

Ngoài ra bạn cũng có thể thêm các thành phần như ASP.NET Core runtime vào .NET core.

1. PostgreSQL
2. PostgreSQL là gì? Tìm hiểu thông tin về cơ sở dữ liệu mã nguồn mở

[PostgreSQL](https://bizflycloud.vn/tin-tuc/postgresql-la-gi-tim-hieu-ve-co-so-du-lieu-ma-nguon-mo-tien-tien-nhat-the-gioi-20180919175924611.htm) là một hệ thống quản trị cơ sở dữ liệu quan hệ-đối tượng (object-relational database management system) có mục đích chung, hệ thống cơ sở dữ liệu mã nguồn mở tiên tiến nhất hiện nay.

PostgreSQL được phát triển dựa trên POSTGRES 4.2 tại phòng khoa học máy tính Berkeley, Đại học California.

PostgreSQL được thiết kế để chạy trên các nền tảng tương tự UNIX. Tuy nhiên, PostgreSQL sau đó cũng được điều chỉnh linh động để có thể chạy được trên nhiều nền tảng khác nhau như Mac OS X, Solaris và Windows.

PostgreSQL là một [phần mềm mã nguồn mở miễn phí](https://bizflycloud.vn/tin-tuc/phan-mem-ma-nguon-mo-la-gi-5-phan-mem-ma-nguon-mo-duoc-ua-chuong-nhat-hien-nay-20201028120531765.htm). Mã nguồn của phần mềm khả dụng theo license của PostgreSQL, một license nguồn mở tự do. Theo đó, bạn sẽ được tự do sử dụng, sửa đổi và phân phối PostgreSQL dưới mọi hình thức.

PostgreSQL không yêu cầu quá nhiều công tác bảo trì bởi có tính ổn định cao. Do đó, nếu bạn phát triển các ứng dụng dựa trên PostgreSQL, chi phí sở hữu sẽ thấp hơn so với các hệ thống quản trị dữ liệu khác.

1. Vì sao sử dụng PostgreSQL?

PostgreSQL sở hữu một hệ tính năng đa dạng giúp hỗ trợ các nhà phát triển xây dựng app, các nhà quản trị bảo vệ toàn vẹn dữ liệu, và tạo ra một môi trường chịu lỗi [fault-tolerant](https://bizflycloud.vn/tin-tuc/fault-tolerance-la-gi-he-thong-chiu-loi-co-tac-dung-gi-20180716163002091.htm) giúp bạn quản lý dữ liệu bất kể tập dữ liệu lớn hay nhỏ. Bên cạnh hệ thống nguồn mở và miễn phí, PostgreSQL cũng có khả năng mở rộng tuyệt vời. Ví dụ, bạn có thể định nghĩa các kiểu dữ liệu riêng của bạn, xây dựng các hàm tùy chỉnh, hay viết mã từ các ngôn ngữ lập trình khác nhau mà không cần biên dịch lại cơ sở dữ liệu!

PostgreSQL tuân theo tiêu chuẩn SQL nhưng không mâu thuẫn với các tính năng truyền thống hay có thể dẫn đến các quyết định kiến trúc gây hại. Nhiều tính năng theo tiêu chuẩn SQL được hỗ trợ, tuy nhiên đôi khi có thể có cú pháp hoặc hàm hơi khác một chút.

1. Tìm hiểu về ưu, nhược điểm của PostgreSQL

Ưu điểm:

* Dễ sử dụng
* Cho phép chạy các trang web, ứng dụng web động với LAMP
* Lưu nhật ký đã viết trước đó để sử dụng như một cơ sở dữ liệu với khả năng sửa lỗi tốt
* Có sẵn mã nguồn miễn phí theo giấy phép nguồn mở, cho phép chỉnh sửa và triển khai tùy theo nhu cầu
* Có khả năng đối tượng hóa dữ liệu
* Được hỗ trợ bởi một cộng đồng đông đảo và nhiều kinh nghiệm
* Giảm thiểu việc bảo trì hệ thống

Nhược điểm:

* Do không thuộc quyền quản lý của bất kỳ tổ chức nào nên gây khó khăn cho người dùng khi muốn tiếp cận với đầy đủ tính năng
* Tập trung vào khả năng tương thích nên khi thực hiện thay đổi để cải thiện tốc độ nên sẽ yêu cầu nhiều công việc hơn so với MySQL
* Một số ứng dụng nguồn mở không hỗ trợ PostgreSQL dù có hỗ trợ MySQL
* Hiệu suất chậm hơn so với MySQL

1. Các yếu tố giúp PostgreSQL nổi bật

PostgreSQL là hệ thống quản lý cơ sở dữ liệu đầu tiên triển khai tính năng kiểm soát đồng thời nhiều phiên bản (MVCC) trước cả Oracle. Tính năng MVCC cũng tương tự với các snapshot riêng biệt trong Oracle.

Là hệ thống quản lý cơ sở dữ liệu quan hệ đối tượng, PostgreSQL cho phép thêm vào các tính năng tùy chỉnh được phát triển bằng các ngôn ngữ chương trình khác nhau như C/C , Java,...

Bên cạnh đó, khi bạn có bất kỳ vấn đề gì cần hỗ trợ, luôn có một cộng đồng nhiệt tình sẵn sàng để trợ giúp. Bạn luôn có thể tìm thấy câu trả lời từ cộng đồng của PostgreSQL về các vấn đề có thể gặp phải khi làm việc với PostgreSQL. Ngoài ra, cũng có rất nhiều công ty cung cấp các dịch vụ hỗ trợ trong trường hợp bạn thực sự cần đến.

1. PHÂN TÍCH
2. Xác định yêu cầu
3. Yêu cầu chức năng

Website được xây dựng dành cho các nhóm người dùng sau

1. Đối với người quản trị/giáo vụ/quản lý đào tạo (Administrators)

* Quản lý, cài đặt toàn bộ hệ thống
* Quản lý, cập nhật thông tin giảng viên, sinh viên, lớp học phần

1. Đối với giảng viên

* Quản lý chi tiết lịch dạy của cá nhân trên phần mềm
* Quản lý việc báo nghỉ/đăng ký dạy bù cho sinh viên
* Thống kê, theo dõi tình hình giảng dạy của bản thân

1. Yêu cầu phi chức năng
2. Yêu cầu hệ thống

* Phân quyền: Hệ thống có 2 quyền chính:
* Quyền quản trị/Giáo vụ/Quản lý đào tạo (Admin)
* Quyền người dùng (Giảng viên)

1. Yêu cầu về chất lượng

* Tính tiện dụng: Website tổ chức theo mô hình giống như trang web về giáo dục, với các chức năng và giao diện đơn giản, dễ sử dụng
* Tính dễ tiếp cận: Website có các tính năng và hình ảnh trực quan, giúp cho người dùng lần đầu sử dụng không cảm thấy bỡ ngỡ
* Tính nhanh nhẹn: Thông tin sẽ được gửi ngay trực tiếp tới người có liên quan khi người dùng thao tác trên website

1. Đặc tả phần mềm
2. Phát biểu bài toán

Trong mô hình đào tạo tín chỉ tại các trường, sau khi Sinh viên đăng ký các môn học của mình thông qua các hệ thống đăng ký học phần tín chỉ (trực tiếp, qua website, …). Căn cứ vào đó để phòng đào tạo bố trí lịch dạy phù hợp cho từng Giảng viên

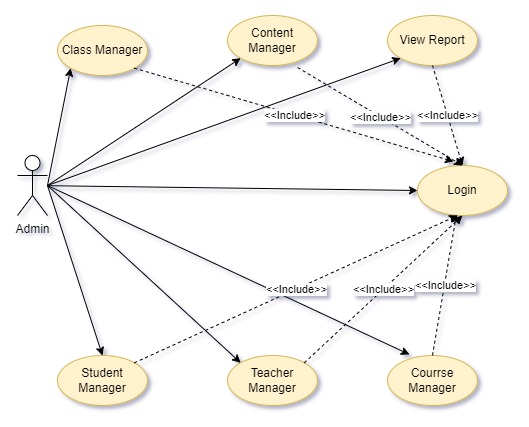
Người quản trị là người có quyền cao nhất, từ cài đặt hệ thống tới việc quản lý giảng viên, sinh viên, lớp học …

Sau khi Giảng viên nhập lịch dạy của mình vào phần mềm, Giảng viên sau đó khi đăng nhập vào hệ thống sẽ có thể xem thông tin lịch dạy cũng như báo cáo tình hình giảng dạy của mình.

Giảng viên đăng ký dạy bù/báo nghỉ trên danh sách lịch dạy của bản thân. Khi hủy lịch dạy sẽ yêu cầu GV nhập: lý do hủy, số tiết hủy. Khi đăng ký dạy bù sẽ yêu cầu GV nhập: ngày dạy bù, thời gian bắt đầu, số tiết dạy. Tuy nhiên khi đăng ký dạy bù thì 1 Sinh viên có thể bị trùng lịch với lịch học của môn học phần khác, phần mềm chỉ cho đăng ký dạy bù nếu số Sinh viên trùng lịch không vượt quá số lượng quy định.

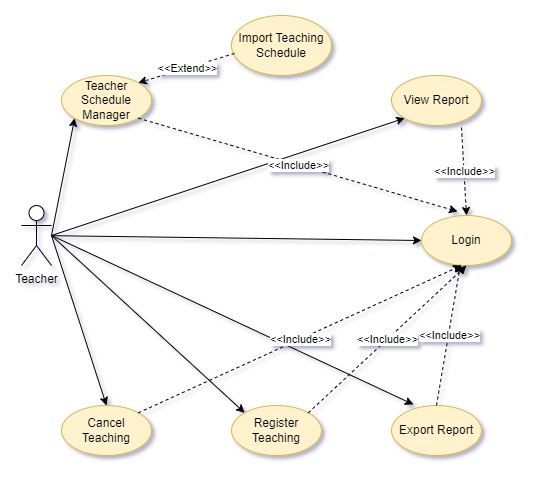
Khi Giảng viên có thao tác dạy bù/báo nghỉ thì thông tin sẽ ngay lập tức được gửi tới Email tất cả của sinh viên trong lớp học phần bảo đảm Sinh viên có thể nhận được thông tin sớm nhất và sắp sếp thời gian biểu hợp lý.

1. Mô hình Use Case
2. Người quản trị



Hình 2. 1 Mô hình Use Case cho người quản trị

1. Giảng viên



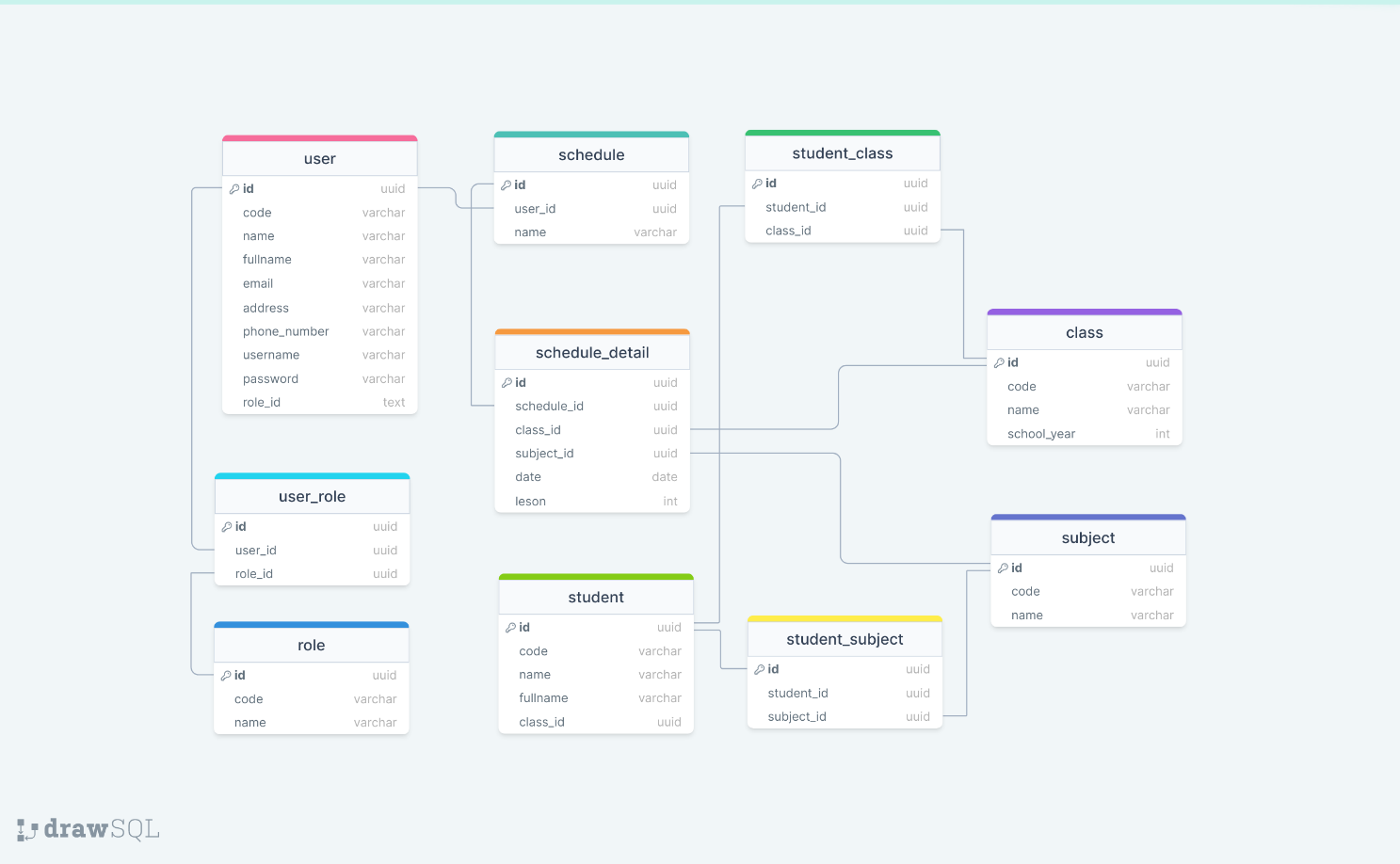
Hình 2. 2 Mô hình Use Case cho Giảng viên

1. Bảng danh sách các nghiệp vụ (Use Case List)

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên nghiệp vụ** | **Diễn giải** |
| 1 | Login | Chức năng đăng nhập vào hệ thống |
| 2 | Content Manager | Chức năng quản lý các thành phần nội dung chỉ dùng cho người có quyền quản trị (Admin) |
| 3 | Course Manager | Quản lý môn học: Cho phép xem, sửa, xóa chi tiết môn học (dùng cho quyền quản trị) |
| 4 | Class Manager | Quản lý lớp học: Cho phép xem, sửa, xóa chi tiết lớp học (dùng cho quyền quản trị) |
| 5 | Teacher Manager | Quản lý giảng viên: Cho phép xem, sửa, xóa chi tiết giảng viên (dùng cho quyền quản trị) |
| 6 | Student Manager | Quản lý sinh viên: Cho phép xem, sửa, xóa chi tiết Sinh viên (dùng cho quyền quản trị) |
| 7 | Teaching Schedule Manager | Xem chi tiết lịch dạy |
| 8 | View Report | Xem báo cáo tình hình giảng dạy |
| 9 | Import Teaching Schedule | Nhập khẩu lịch dạy vào phần mềm từ file Excel |
| 10 | Export Report | Xuất khẩu báo cáo từ phần mềm ra file Excel |
| 11 | Register Teaching | Đăng ký dạy bù |
| 12 | Cancel Teaching | Báo nghỉ |
| 13 | Send Notify | Gửi thông báo cho sinh viên |

Bảng 2. 1 Bảng danh sách nghiệp vụ

1. THIẾT KẾ CƠ SỞ DỮ LIỆU
2. Sơ đồ quan hệ giữa các bảng



Hình 3. 1 Sơ đồ quan hệ giữa các bảng

1. Chi tiết các bảng

Bảng 3. 1 Bảng người dùng (user)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| user | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | Code | Varchar | Mã user | Not null |
| 3 | Name | Varchar | Tên user | Not null |
| 4 | Fullname | Varchar | Tên đầy đủ | Not null |
| 5 | Email | Varchar | Email |  |
| 6 | Address | Varchar | Địa chỉ |  |
| 7 | Phone\_number | Varchar | Số điện thoại |  |
| 8 | Username | Varcahr | Tên tài khoản | Not null |
| 9 | Password | Varchar | Mật khẩu | Not null |
| 10 | Role\_id | Uuid | Id quyền | Khóa ngoại |

Bảng 3. 2: Bảng sinh viên (student)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| student | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | Code | Varchar | Mã sinh viên | Not null |
| 3 | Name | Varchar | Tên sinh viên | Not null |
| 4 | Fullname | Varchar | Tên đầy đủ | Not null |
| 5 | Email | Varchar | Email |  |
| 6 | Address | Varchar | Địa chỉ |  |
| 7 | Phone\_number | Varchar | Số điện thoại |  |
| 8 | Class\_id | Uuid | Id lớp học | Khóa ngoại |

Bảng 3. 3: Bảng lớp học phần (class)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| class | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | Code | Varchar | Mã lớp học | Not null |
| 3 | Name | Varchar | Tên lớp học | Not null |
| 4 | School\_year | Varchar | Niên khóa | Not null |

Bảng 3. 4: Bảng môn học phần (subject)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| subject | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | Code | Varchar | Mã user | Not null |
| 3 | Name | Varchar | Tên user | Not null |

Bảng 3. 5: Bảng lịch dạy (schedule)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| schedule | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | User\_id | Uuid | Id người dùng | Khóa ngoại |
| 3 | Name | Varchar | Tên lịch dạy | Not null |

Bảng 3. 6: Bảng chi tiết lịch dạy (schedule\_detail)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Schedule\_detail | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | Schedule\_id | Uuid | Id lịch dạy | Khóa ngoại |
| 3 | Class\_id | Uuid | Mã lớp học | Khóa ngoại |
| 4 | Subject\_id | Uuid | Mã môn học phần | Khóa ngoại |
| 5 | Start | Date | Thời gian bắt đầu | Not null |
| 6 | Lession | Int | Số tiết dạy | Not null |

Bảng 3. 7: Bảng phân quyền (role)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| role | | | | |
| STT | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Id | Uuid | Id | Khóa chính |
| 2 | Code | Varchar | Mã quyền | Not null |
| 3 | Name | Varchar | Tên quyền | Not null |
| 4 | Description | Text | Mô tả quyền |  |

1. HIỆN THỰC CHƯƠNG TRÌNH

**TÀI LIỆU THAM KHẢO**