*LỜI MỞ ĐẦU*

Trong những năm gần đây, khoa học công nghệ Việt Nam đã và đang hội nhập vào dòng chảy của khoa học công nghệ tiên tiến trên thế giới.Công nghệ thông tin ở nước ta phát triển khá nhanh và mạnh và hơn hết ứng dụng của công nghệ thông tin vào cuộc sống cũng đóng một vai trò vô cùng lớn.Như tạo ra rất nhiều phần mềm cũng như website có hiệu quả thực tế cao.Từ các lĩnh vực quản lý cho đến các lĩnh vực quảng cáo,tuyên truyền thì công nghệ thông tin góp phần không hề nhỏ

### **Project Charter**

Project Title : Xây dựng cổng thông tin các thông tin tại Đại Học Công Nghệ (Hutech)

Project Start Date : 01/04/2019 Projected Finish Date : 22/06/2019

Project Manager : Lê Khôi Đệ - 0782657635 - lekhoide@gmail.com

Project Objectives : Thiết kế và xây dựng Website “Cổng thông tin các thông tin tại Đại Học Công Nghệ (Hutech)” trên nền tảng công nghệ ASP.NET MVC 5 cho Đồ Án Cơ Sở hướng dẫn bởi giảng viên Lê Thị Ngọc Thơ trong vòng 12 tuần.

Approach :

* Xác định, phân tích và thiết kế trong 3 tuần đầu tiên. Thực hiện và kiểm thử trong 8 tuần. Vận hành và báo cáo kết quả cho giáo viên hướng dẫn 1 tuần còn lại.
* Mỗi tuần nộp báo cáo cho giáo viên hướng dẫn và nhận yêu cầu cho tuần tiếp theo.

Roles and Responsibilities :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Role | Name | Organization / Position | Contact Information | Mssv / Class |
| Lập kế hoạch cho nhóm  Thiết kế giao diện  Nghiên cứu công nghệ mới | LÊ KHÔI ĐỆ | Project Manager  Designer  Technical Master | 0782657635  lekhoide@gmail.com | 1411010102  15DTH06 |
| Thu thập và phân tích thông tin  Kiểm thử chức năng  Viết code | NGUYỄN BÁ TÂM | Bussiness Analyst  Tester  Programmer | 0333058893  tamnguyen.tgth@gmail.com | 1611061022  16DTHC2 |

Sign-off: (Signatures of all above stakeholders. Can sign by their names in table above.)

LÊ KHÔI ĐỆ

NGUYỄN BÁ TÂM

Comments: (Handwritten or typed comments from above stakeholders, if applicable)

**CHƯƠNG 1. TỔNG QUAN**

* 1. **Giới thiệu dự án**

Với mỗi trường đại học, việc xây dựng một trang thông tin luôn cập nhật và lưu trữ tất cả các thông tin về các hoạt động xoay quanh các lĩnh vực hỗ trợ việc học tập cho sinh viên là điều không thể không có nhất là khi chúng ta đang trong thời kì cách mạng công nghệ 4.0 thì việc luôn cập nhật thông tin là điều thiết yếu đối với mỗi sinh viên đang học tập và công tác tại các trường đại học.

Nhận thấy hầu hết các thông tin liên quan đến việc hỗ trợ học tập cho sinh viên ở các trường đại học nói chung và trường đại học công nghệ nói riêng đều nằm riêng lẻ cả các trang mạng xã hội, các group tập thể và cũng chỉ số ít người có liên quan trong các group hay các trang đó mới cập nhận được thông tin cần thiết còn phần đông nó cũng chưa được công khai hay có một cách nào đó để tất cả các sinh viên trong trường có thể cập nhật được.

Nắm bắt được thực trạng và nhu cầu thiết yếu của hầu hết sinh viên trong trường muốn tích lũy điểm rèn luyện, muốn tham gia tìm hiểu và bám sát các hướng đi của doanh nghiệp, muốn tham gia các sự kiện các hoạt động vui chơi giải trí để giảm áp lực học tập cũng như giao lưu học hỏi..v..v...Sau một thời gian bàn bạc và đưa ra các tình huống các giả thuyết thì nhóm em đã quyết định xây dựng một trang web riêng chuyên sàng lọc và cập nhật các thông tin cần thiết nhất dành cho các bạn sinh viên mang tên “ Cổng thông tin tại trường Đại Học Công Nghệ TP.HCM(Hutech)”.

**1.2 Nhiệm vụ đồ án**

*1.2.1 Lý do hình thành đề tài*

Dự án “Cổng thông tin tại trường Đại Học Công Nghệ TP.HCM(Hutech)” là một dự án được xây dựng lên nhằm hỗ trợ sinh viên thấy được nhiều thông tin hơn, ở đây là các hội thảo, sự kiện, thông tin về các câu lạc bộ, hội nhóm, các cuộc thi được phân chia riêng lẻ ở 7 Trung Tâm, 8 Viện, 11 Phòng Ban, 13 Khoa khác nhau giờ đây sẽ được tổng hợp vào một Website. Nhờ vậy tất cả các sinh viên trong trường sẽ có cơ hội được cập nhật các thông tin không chỉ riêng khoa của mình học mà còn biết được các hoạt động, sự kiện ở các khoa khác trong trường.

*1.2.2 Mục tiêu nghiên cứu ,phạm vi dự án*

Mục tiêu hướng đến của việc xây dựng và hoàn thành dự án của chúng tôi là tất cả cac bạn sinh viên và đội ngũ giảng viên đang học tập và công tác tại trường Đại Học Hutech.

*1.2.3 Ý nghĩa*

Website “Cổng thông tin tại trường Đại Học Công Nghệ TP.HCM(Hutech)” được xây dựng nên sẽ giúp ích rất nhiều cho việc cập nhật các thông tin cho sinh viên đang học tập tại trường và hơn hết website cũng có thể là một giải pháp đăng tải và phát triển các sự kiện các thông tin về trường Đại Học Công Nghệ trên mạng xã hội.

**1.3 Cấu trúc đồ án**

*1.3.1 Xác định (1/04/2019 - 07/04/2019)*

1.3.1.1 Nghiên cứu các Website tương tự

1.3.1.2 Xác định và phân tích các chức năng chính cần có

1.3.1.3 Xác định và phân tích cơ sở dữ liệu

1.3.1.4 Tổng kết và đánh giá sơ bộ

1.3.1.5 Lên kế hoạch chi tiết

- Project Manager lọc ra chức năng chính theo cấp độ

- Lập bảng chi tiết rủi ro và giải pháp đề xuất

- Lên bảng cấu trúc công việc

- Xác định số người tham gia dự án : 2 người

- Xác định thời gian bắt đầu và kết thúc : 1/04/2019 - 22/06/2019

*1.3.2 Phân tích và thiết kế (08/04/2019 - 21/04/2019)*

1.3.2.1 Yêu cầu (08/04 - 14/04)

- Đặc tả yêu cầu chức năng

- Đặc tả yêu cầu giao diện

- Đặc tả yêu cầu hệ thống

1.3.2.2 Dữ liệu (15/04 - 21/04)

- UseCase

- Mô hình UseCase

- Danh sách Actor

- Danh sách UseCase

- Đặc tả chi tiết UseCase

- ERD

- Mô hình thực thể kết hợp ERD

- Danh sách các thực thể

1.2.3.4 BFD

Mô hình phân rã chức năng - BFD

1.2.3.5 DFD

Mô hình luồng dữ liệu - DFD

1.2.3.6 Giao diện (15/04 - 21/04)

- Thiết kế giao diện Demo

- Thiết kế giao diện Đăng ký

- Thiết kế giao diện Đăng nhập

- Thiết kế giao diện Trang chủ

- Thiết kế giao diện Người dùng quản lý tài khoản cá nhân

- Thiết kế giao diện Quản trị quản lý danh sách tài khoản cá nhân người dùng

1.2.3.7 Nhận yêu cầu của giảng viên và cập nhật (Update)

*1.3.3 Thực hiện (22/04/2019 - 2/06/2019)*

1.3.3.1 Cơ sở dữ liệu (22/04 - 28/04)

- Hoàn chỉnh toàn bộ cơ sở dữ liệu

- Nhận yêu cầu giảng viên và cập nhật (Update)

1.3.3.2 Chức năng (29/04 - 2/06)

- Đăng ký bằng Google

- Đăng nhập bằng Google

- Ghi nhớ đăng nhập

- Quên mật khẩu

1.3.3.3 Chức năng người dùng quản lý tài khoản cá nhân (6/05 - 26/05)

- Xem thông tin cá nhân

- Xóa thông tin cá nhân

- Sửa thông tin cá nhân

- Đổi mật khẩu

- Tạo sự kiện

- Chỉnh sửa sự kiện

- Xóa sự kiện

- Lưu sự kiện

- Đăng xuất

1.3.3.4 Chức năng tìm kiếm sự kiện (27/05 - 2/06)

- Tìm kiếm sự kiện theo tên sự kiện

- Tìm kiếm sự kiện theo địa điểm

- Tìm kiếm sự kiện theo ngày

- Tìm kiếm sự kiện theo phòng-ban / khoa/ viện/ trung tâm/ khác / tất cả

1.3.3.5 Quản trị quản lý danh sách tài khoản cá nhân người dùng (27/05 - 2/06)

- Xem danh sách tài khoản cá nhân

- Cấm một tài khoản cá nhân truy cập

- Hủy cấm một tài khoản cá nhân truy cập

- Hoàn chỉnh toàn bộ chức năng

- Nhận yêu cầu của giảng viên và cập nhật (Update)

*1.3.4 Kiểm thử (3/06/2019 - 16/06/2019)*

1.3.4.1 Kiểm thử hệ thống (3/06 - 9/06)

- Kiểm tra các chức năng trong thiết kế

- Hiệu chỉnh và sửa lỗi

1.3.4.2 Kiểm thử chấp nhận (10/06 - 16/06)

- Demo cho giảng viên các chức năng của hệ thống

- Hiệu chỉnh và sửa lỗi theo yêu cầu của giảng viên

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

**2.1 Các khái niệm và cơ chế hoạt động**

*2.1.1. Tổng quan về công nghệ ASP.NET MVC 5*

Nhóm em đã sử dụng những công nghệ công cụ này để áp dụng vào đề tài.

* ASP.NET MVC 5
* SQL Server 2017
* Visual Studio 2017
* Lucidchart
* Entity Framework Code First

2.1.1.1. Môi trường lập trình

Visual Studio là ( IDE - Integrated Development Environment) một công cụ phát triển phần mền do Microsoft phát triển. Visual Studio cũng là một phần mền được sử dụng bởi các lập trình viên để xây dựng nên các sản phẩm phầm mềm.

Phiên bản đầu tiên của Visual Studio được Microsoft phát hành vào năm 1997 với hai phiên bản là Professional và Enterprise. Tính đến thời điểm hiện tại, Visual Studio đã trải qua nhiều thời kì phát triển và đã phát hành những bản Visual Studio như Visual Studio 2005, Visual Studio 2008, Visual Studio 2010, Visual Studio 2012, Visual Studio 2013, Visual Studio 2015, Visual Studio 2017, Visual Studio 2019. Với mỗi phiên bản phát hành có nhiều công nghệ và tính năng mới được tích hợp để hỗ trợ tốt nhất cho lập trình viên.[1]

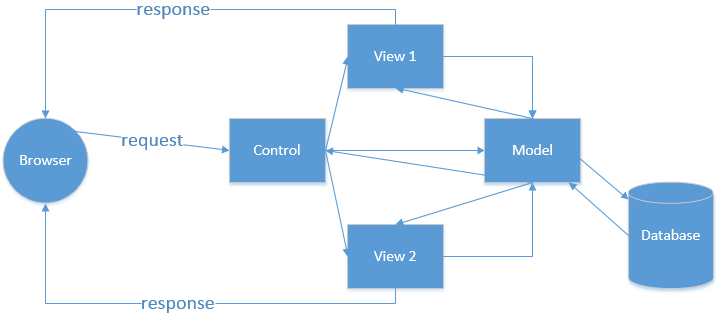
2.1.1.2 Ngôn ngữ lập trình C#

C# được thiết kế bởi kiên trúc sư phần mềm nổi tiếng Anders Hejlsberg. Vào năm 2000, Microsoft muốn phát triển một ngôn ngữ lập trình có thể hoạt đông trong nền tảng của bộ khung .NET và họ đã thành công. Đó chính là ngôn ngữ C#.Điều này cũng lý giải tại sao ngôn ngữ C# chỉ chạy được trên nền tảng Windows và có cài .NET Framework.Tuy nhiên với những ứng dụng của nó thì nhược điểm này không phải là một vấn đề quá lớn.Chỉ sau một thời gian ngắn cho ra mắt, ngôn ngữ này đã trở thành nổi tiếng và hiện nay nó được nằm trong top 10 ngôn ngữ lập trình hàng đầu thế giới.

C# là một ngôn ngữ gần gũi vơi các ngôn ngữ lâp trình thông dụng như C++,Java hay Pascal.Bởi nó được xây dựng trên nền tảng của những ngôn ngữ này.Chính bởi vậy mà nó sẽ được kế thừa những tinh hoa của những ngôn ngữ thông dụng trên.Khi C# ra đời nó đã giải quyết được các khiếm khuyết của C++ như con trỏ, các hiệu ứng phụ,… vì vậy nó dễ tiếp cận và dễ được phát triển hơn . Đây cũng là một lợi thế cho những ai đã học và quen thuộc với C++ hay Java khi mà học ngôn ngữ C#.C# được hỗ trợ bởi một cồng đồng các nhà phát triển lớn.Chính cộng đồng sử dụng này đã khiến cho C# trở thành ngôn ngữ đáng tin cậy và hoạt động hiệu quả hơn.Cộng đồng sử dụng đóng một vai trò rất quan trọng trong một ngôn ngữ lập trình . Nếu cộng đồng của một ngôn ngữ lớn hơn các mục tiêu hướng đến của các công cụ thì ngôn ngữ đó sẽ xuất hiện thường xuyên trên thị trường.Cộng đồng lớn cũng có nghĩa là có rất nhiều trợ giúp thảo luận và có ích cho những người mới chân ướt chân ráo bước vào nghề.

Với sự hỗ trợ của bộ khung .NET, C# đã được sử dụng để xây dựng ứng dụng trên desktop.Nhiều ứng dụng phổ biến bạn có thể tham khảo như Microsoft Office, Skype, Photoshop hay Visual Studio.[2]

#### 2.1.1.3. Mô hình 3 lớp (MVC) trong C#



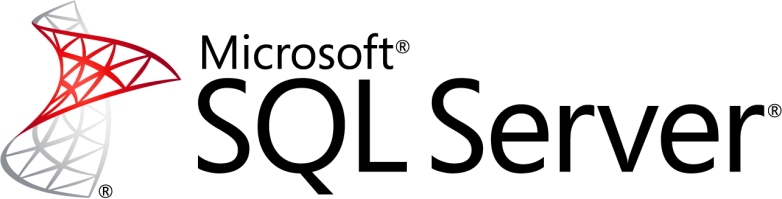
Mô hình MVC là chia tất cả mục của một ứng dụng ra làm ba thành phần (component) khác nhau: Model, View và Controller. Mỗi thành phần của mô hình MVC có một trách nhiệm riêng và không phụ thuộc vào các thành phần khác. Sự thay đổi trong một thành phần sẽ không có hoặc có rất ít ảnh hưởng đến các thành phần khác.

Trách nhiệm của mỗi thành phần:

* Model: Model được giao nhiệm vụ cung cấp dữ liệu cho cơ sở dữ liệu và lưu dữ liệu vào các kho chứa dữ liệu. Tất cả các nghiệp vụ logic được thực thi ở Model. Dữ liệu vào từ người dùng sẽ thông qua View được kiểm tra ở Model trước khi lưu vào cơ sở dữ liệu. Việc truy xuất, xác nhận và lưu dữ liệu là một phần của Model.
* View: View hiển thị các thông tin cho người dùng của ứng dụng và được giao nhiệm vụ nhận các dữ liệu vào từ người dùng, gửi đi các yêu cầu đến bộ điều khiển, sau đó nhận lại các phản hồi từ bộ điều khiển và hiển thị kết quả cho người dùng. Các trang HTML, JSP, các thư viện và các file nguồn là một phần của View.
* Controller: Controller là tầng trung gian giữa Model và View. Controller được giao nhiệm vụ nhận các yêu cầu từ phía máy khách. Một yêu cầu được nhận từ máy khách được thực hiện bởi một chức năng logic thích hợp từ thành phần Model, sau đó sinh ra các kết quả cho người dùng và được thành phần View hiển thị. ActionServlet,Action, ActionForm và struts-config.xml là các phần của Controller.

Với mô hình MVC, khi xây dựng các ứng dụng chúng ta sẽ tránh được rất nhiều những vất vả khi bảo trì, thay đổi. Những thay đổi ở mỗi thành phần thường rất ít khi ảnh hưởng đến các thành phần khác.

### *2.1.2. Tổng quan về hệ quản trị cơ sở dữ liệu SQL Server*



Có rất nhiều hệ quản trị cơ sở dữ liệu mạnh mẽ như: Oracle, My SQL… nhưng chúng em sẽ sử dụng SQL Server để thực hiện đề tài“Xây dựng phần mềm quản lý điểm sinh viên” này.

Hệ quản trị cơ sở dữ liệu Microsoft SQL Server (MSSQL) là một trong những hệ quản trị cơ sở dữ liệu thông dụng nhất hiện nay. Đây là hệ quản trị cơ sở dữ liệu thường được sử dụng trong các hệ thống trung bình, với ưu điểm có các công cụ quản lý mạnh mẽ giúp cho việc quản lý và bảo trì hệ thống dễ dàng, hỗ trợ nhiều phương pháp lưu trữ, phân vùng và đánh chỉ mục phục vụ cho việc tối ưu hóa hiệu năng.

SQL Server luôn được Microsoft cải tiến để nâng cao hiệu năng, tính sẵn sàng của hệ thống, khả năng mở rộng và bảo mật, cung cấp nhiều công cụ cho người phát triển ứng dụng được tích hợp với bộ Visual Studio do Microsoft cung cấp. SQL Server có 4 dịch vụ lớn là Database Engine, Intergration Service, Reporting service và Analysis Services.

* DataBase Engine: được phát triển để thực thi tốt hơn với việc hỗ trợ cả dữ liệu có cấu trúc và dữ liệu phi cấu trúc (XML).
* Khả năng sẵn sàng của hệ thống được nâng cao, hỗ trợ các chức năng: Database mirroring (cơ sở dữ liệu gương), failover clustering , snapshots và khôi phục dữ liệu nhanh.
* Việc quản lý chỉ mục được thực hiện song song với việc hoạt động của hệ thống. Người dùng có thể thêm chỉ mục, xây dựng lại chỉ mục hay xóa một chỉ mục đi trong khi hệ thống vẫn được sử dụng.
* Chức năng phân vùng dữ liệu được hỗ trợ: Người dùng có thể phân vùng các bảng và chỉ mục cũng như quản lý phân vùng dữ liệu một cách dễ dàng. Việc hỗ trợ phân vùng dữ liệu giúp nâng cao hiệu năng hoạt động của hệ thống.
* Dịch vụ đồng bộ hóa dữ liệu được mở rộng với việc hỗ trợ mô hình đồng bộ hóa ngang hàng. Đây là dịch vụ giúp đồng bộ hóa dữ liệu giữa các máy chủ dữ liệu, giúp mở rộng khả năng của hệ thống.
* Dịch vụ tích hợp (Integration Service) thiết kế lại cho phép người dùng tích hợp dữ liệu và phân tích dữ liệu từ nhiều nguồn khác nhau. Hỗ trợ việc quản lý chất lượng dữ liệu và làm sạch dữ liệu, một công việc quan trọng trong tiến trình ETL.
* Dịch vụ phân tích dữ liệu (Analysis Service): cung cấp khung nhìn tích hợp và thống nhất về dữ liệu cho người dùng, hỗ trợ việc phân tích dữ liệu.
* Công cụ khai phá dữ liệu (Data mining) được tích hợp hỗ trợ nhiều thuật toán khai phá dữ liệu, hỗ trợ cho việc phân tích, khai phá dữ liệu và xây dựng các hệ thống hỗ trợ ra quyết định cho người quản lý.
* Dịch vụ xây dựng quản lý báo cáo (Reporting Service) dựa trên nền tảng quản trị doanh nghiệp thông minh và được quản lý qua dịch vụ web. Báo cáo có thể được xây dựng dễ dàng với ngôn ngữ truy vấn MDX. Thông qua các công cụ trên Business Intelligent, người dùng dễ dàng truy cập báo cáo và trích xuất ra nhiều định dạng khác nhau thông qua trình duyệt web.

# *2.1.3 Tổng quan về công nghệ Entity Framework Code First*

ADO.NET Entity Framework (EF) được Microsoft đưa ra từ phiên bản .NET 3.5 tích hợp với Visual Studio 2008. Trọng tâm của EF là Entity Data Model, một mô hình mẫu cho application domain của bạn ánh xạ ngược trở lại vào đồ hình của CSDL của bạn.Mô hình mẫu này mô tả cá lớp chính trong ứng dụng của bạn.EF sử dụng mô hình mẫu này trong truy vấn từ CSDL, tạo các đối tượng từ dữ liệu và sau đó đưa các thay đổi ngược trở lại CSDL.

Phiên bản đầu tiên của EF cho phép lập trình viên tạo mô hình mẫu bằng cách reverse engineering một CSDL đag có vào một tập tin XML. Tập tin XML này sử dụng phần mở rộng EDMX và bạn có thể sử ụng designer để xem và tinh chỉnh mô hình sao cho thích hợp nhất với domain của bạn. Visual Studio 2010 và .NET 4 mang đến phiên bản thứ 2 của EF, được đặt tên là Entity Framework 4 (EF4),để phù hợp vơis phiên bản .NET .Về mặt mô hình hóa, một tính năng mới gọi là Model First được đưa vào.Tại đây, bạn có thể thiết kế một mô hình mẫu của bạn trong Visual desgner và sau đó tạo CSDL dưa trên mô hình này.

Model First cho phép lập trình viên làm việc trên các Project mới mà không phải phụ thuộc vào dữ liệu để tận dụng khả năng của EF.Lập trình viên có thể bắt đầu với việc thập trung vào application domain bằng việc thiết kế mô hình mẫy và để CSDL tự động tạo theo như quy trình.

Dù cho việc thiết kế EDMX theo hướng database - first hay model-first, bước kế tiếp để tạo domain là để cho việc tạo code tự động tạo các lớp dựa trên các thực thể và các mối liên hệ của chúng mà nó tìm thấy trong model. Từ đây,lập trình viên có các strongly typed class đại diện cho các domain object và có thể tiếp tục việc phát triển ứng dụng xoay quanh các class này.

Một thay đổi lớn khác trong phiên bản EF đó là trong .NET 3.5, cách duy nhất EF có thể quản lý cá đối tương trong vùng nhớ(in-memory object) là yêu cầu các lớp phải ké thừa từ EntityObject của EF.EntityObject theo dõi các thay đổi và có khả năng đẩy ngược chúng tại CSDL. Theo tính năng này, .NET 4 đã giới thiệu POCO(Plain Old CLR Object ) hỗ trợ EF theo dõi các thay đổi cho các lớp đơn giản hơn mà không cần EntityObject phải được thực thi.

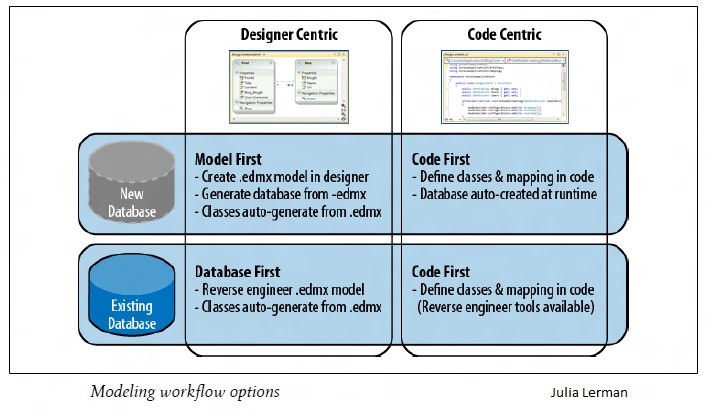
Được xây dựng dựa trên các thành phần được giới thiệu trong EF4, Code First cho phép lập trình viên định nghĩa domain model với code thay vì phải sử dụng một tập tin EDMX.Mặc dù Model First và Database First dử dụng code generation để cung cấp các class cho bạn làm việc, nhiều lập trình viên không muốn làm việc với designer hoặc các class được tạo từ chúng.

Trong Code First, bạn bắt đầu bằng việc định nghĩa domain model của bạn bằng việc sử dụng các POCO class không phụ thuộc vào EF. Code First có thể bao hàm nhiều thông tin về model từ các class của bạn. Bạn có thể cung cấp cấu hình bổ sung để mô tả model hoặc override những gì mà Code First cung cấp.Phần cấu hình này cũng được thực hiện trong code chứ không phải trong tập tin MXL, hay desgner.

EF4 cũng hỗ trợ các POCO class khi làm việc với designer.EF cung cấp một POCO template cho phép tạo ra các POCO class. Những class này sẽ được cập nhật tự động khi bạn thay đổi trên designer.Bạn cũng có thể sử dụng các POCO class của mình, nhưng một khi bạn đã chọn cách này thì phải chấp nhận việc động bộ hóa giữa các class của bạn và tập tin EDMX. Điều này có nghĩa là bất kì việc thay đổi vào phải được thực hiện ở 2 nơi:một là trên designer, một ở trong các class của bạn.Một trong những thuận lợi lớn của Code First đó là các class của bạn trở thành model, điều này có nghĩa rằng bất kì thay đổi nào cho model chỉ cần được thực hiện ở một nới duy nhất-POCO class của bạn.

Code First, Model First và Database First cũng chỉ là các cách để xây dựng một Entity Data Model để có thể được sử dụng với EF để thực hiện việc truy cập dữ liệu.

Microsoft đưa ra các tùy chọn Database First,Model First và Code First như là các Workfolw là vì thực sự mỗi tùy chọn đều bao gồm một tập các bước dù cho bạn tự thực hiện hay tự động.[3]



*2.1.4 Quy trình kiểm thử website*

2.1.4.1 Kiểm thử chức năng website

Trong kiểm thử chức năng chúng ta cần kiểm tra từng thành phần hoạt động có như mong đợi hay không, vì vậy nó còn được gọi là “kiểm thử các thành phần”. Kiểm thử chức năng giúp kiểm tra các chức năng của thành phần ứng dung, về cơ bản là để kiểm tra các chức năng được đề cập trong tài liệu mô tả chức năng cũng như kiểm tra xem ứng dụng phần mềm có đáp ứng được kỳ vọng của người dùng hay không.

2.1.4.2 Kiểm thử liên kết

Kiểm tra tất cả các liên kết hỏng trên website và tất cả các liên kết đang hoạt động chính xác, bạn có thể kiểm tra các [liên kết](https://mona.media/link-la-gi/) khác nhau trên website:

\_ Liên kết nội bộ

\_ Liên kết ngoài

\_ Liên kết mail

\_ Liên kết anchor

2.1.4.3 Kiểm thử web form

đây là phần thiết yếu của bất kỳ kiểm thử website nào, mục đích chính của kiểm thử web form là lấy thông tin từ người sử dụng và lưu trữ vào cơ sở dữ liệu đồng thời tương tác với lượng dữ liệu ấy. Dưới đây là các trường hợp kiểm thử được nhắc tới trong [kiểm thử web form](http://www.twologs.com/en/resources/formtest.asp" \t "https://mona.media/tim-hieu-ve-quy-trinh-kiem-thu-website/_blank):

\_ Điều đầu tiên là kiểm tra tính hợp lệ trên mỗi field của form, dưới đây là hai loại Validation cần được xem xét – “Client side” và “Server side” validations.

\_ Kiểm tra các giá trị mặc định.

\_ Kiểm tra tất cả các field bắt buộc.

\_ Kiểm tra nếu người dùng không nhập vào một field bắt buộc cần hiển thị một thông báo.

\_ Thêm và sửa thông tin bằng cách sử dụng form.

\_ Thứ tự các tab trên web form.

\_ Kiểm tra các giá trị mặc định của field.

\_ Form cần được định dạng tối ưu khả năng đọc.

\_ Kiểm tra số âm.

2.1.4.4 Kiểm thử nội dung website

\_ Không có lỗi chính tả hoặc ngữ pháp lỗi trong nội dung.

\_ Tích hợp Alt trong hình ảnh .

\_ Không có ảnh hỏng.

\_ Xác nhận tính hợp lệ tất cả giao diện người dùng.

\_ Thực hiện theo một số tiêu chuẩn về xây dựng nội dung trên trang web.

\_ Tất cả nội dung phải rõ ràng và dễ hiểu.

\_ Màu tối gây bất bình cho người sử dụng, vì vậy tránh sử dụng theme màu tối.

\_ Kích thước hình ảnh kích nên phù hợp.

\_ Anchor text phải hoạt động bình thường.

2.1.4.5 Kiểm thử sự tương thích

*Độ tương thích của trình duyệt:*Các ứng dụng web được hiển thị khác nhau trên các trình duyệt khác nhau, mục tiêu của kiểm tra khả năng tương thích của trình duyệt là đảm bảo rằng không có lỗi nào xảy ra trên các trình duyệt web khác nhau trong khi hiển thị website. Bạn cần đảm bảo rằng ứng dụng web của bạn đang được hiển thị đúng trên các trình duyệt khác nhau cũng như kiểm tra AJAX, JavaScript và xác thực hoạt động chính xác.

*Độ tương thích hệ điều hành:*Công nghệ mới, sử dụng đồ họa mới hơn, các API khác nhau được sử dụng có thể không hoạt động trên nhiều hệ điều hành, bên cạnh đó các text field, button có thể hiển thị khác nhau trên hệ điều hành khác nhau. Vì vậy, kiểm thử website cần được thực hiện trên các hệ điều hành khác nhau như Windows, MAC, Solaris, Unix, Linux.

2.1.4.6 Kiểm thử hiệu năng website

Kiểm thử website làm việc dưới lượt tải nặng, được phân thành hai phần: kiểm tra tần suất, kiểm tra lượt tải. Bao gồm:

\_ Kiểm tra thời gian phản hồi của website với tốc độ kết nối khác nhau.

\_ Kiểm tra website có xử lý được nhiều yêu cầu người dùng vào cùng một thời điểm.

\_ Kiểm tra website có hoạt động tốt trong thời điểm lượt tải cao.

\_ Kiểm tra dữ liệu đầu vào lớn từ người dùng.

\_ Kiểm tra hành vi của website khi kết nối với cơ sở dữ liệu. Kiểm tra các phương pháp tối ưu hóa như giảm thời gian tải bằng cách bật bộ nhớ cache trên trình duyệt và phía máy chủ, nén gzip…

2.1.4.7 Kiểm thử bảo mật website

Được thực hiện để đảm bảo rằng có bất kỳ rò rỉ thông tin nào về mã hoá dữ liệu hay không. Trong website thương mại điện tử, kiểm thử bảo mật đóng một vai trò rất quan trọng, nếu thông tin an toàn thì kiểm tra xem làm thế nào để lưu trữ các thông tin nhạy cảm như thẻ tín dụng, thanh toán hóa đơn…Các hoạt động kiểm tra sẽ bao gồm:

\_ Kiểm tra truy cập trái phép vào các trang an toàn, nếu người dùng thay đổi từ “https” sang “http” thì thông báo thích hợp sẽ được hiển thị và ngược lại.

\_ Kiểm tra việc truy cập các trang internal, nếu đăng nhập được yêu cầu thì người dùng nên được chuyển hướng đến trang đăng nhập hoặc thông báo thích hợp sẽ được hiển thị.

 \_ Các thông tin liên quan đến giao dịch, thông báo lỗi, cố gắng đăng nhập nên được ghi vào file log.

 \_ Kiểm tra các tệp tin có bị hạn chế tải xuống hay không.

 \_ Kiểm tra các thư mục web hoặc tập tin web có thể truy cập được trừ khi không được cấu hình để tải xuống.

 \_ Kiểm tra CAPTCHA đã được thêm vào và hoạt động bình thường cho đăng nhập để tự động ngăn chặn các đăng nhập hay chưa.

vấn. Ví dụ: nếu bạn đang chỉnh sửa thông tin và trên URL bạn thấy UserID = 123, hãy thử thay đổi các giá trị tham số này và kiểm xem ứng dụng có cung cấp thông tin người dùng khác không, nên từ chối hiển thị cho trường hợp này để ngăn chặn việc xem thông tin người dùng khác.

 \_ Kiểm tra session hết hạn sau thời gian được xác định nếu người dùng không thao tác trên  Kiểm tra việc cố truy cập thông tin bằng cách thay đổi tham số trong chuỗi truy website.

 \_ Kiểm tra user/password không hợp lệ.

Trên đây là các khâu trong quá trình kiểm thử website. Dù là bạn thiết kế website bán hàng, [thiết kế website giới thiệu](https://mona.media/thiet-ke-website-gioi-thieu-doanh-nghiep/), hay bất kỳ loại website nào thì mục tiêu cuối cùng là phục vụ tốt người dùng. Nếu không kiểm tra kỹ lưỡng mà lại xuất bản trang web, app ra cộng đồng thì rõ ràng sẽ mang đến trải nghiệm không tốt cho người dùng. Do đó, khi thiết kế website tại Mona Media, chúng tôi luôn đề cao công việc kiểm thử, mọi thứ đều được kiểm tra, khắc phục kỹ lưỡng trước khi giao cho chủ sở hữu.

2.1.4.8 Kiểm thử Cookie

Cookie là tập tin chứa thông tin hệ thống của người dùng, các tệp này được lưu ở vị trí mong muốn và được sử dụng bởi các trình duyệt. Các session đăng nhập, thông tin được lưu lại trong cookie (như session) và có thể được truy xuất cho các trang web. Người dùng có thể kích hoạt hoặc vô hiệu Cookies trong các tùy chọn trình duyệt, kiểm thử để kiểm tra xem cookie có được lưu trữ trong máy của người dùng ở định dạng mã hóa hay không.

\_ Kiểm tra ứng dụng bằng cách vô hiệu cookies .

\_ Kiểm tra ứng dụng sau khi hỏng các cookies.

\_ Kiểm tra hành vi của ứng dụng sau khi xóa tất cả cookie trên website.

\_ Kiểm tra cookie có hoạt động trên nhiều duyệt khác nhau hay không.

\_ Kiểm tra cookie cho đăng nhập xác thực có hoạt động hay không.

\_ Kiểm tra hành vi của ứng dụng sau khi xoá cookie (session) bằng cách xoá bộ nhớ cache hoặc sau khi cookie hết hạn.

\_ Kiểm tra đăng nhập vào ứng dụng sau khi xóa cookie (session).

2.1.4.9 Kiểm thử cơ sở dữ liệu

Độ tin cậy của dữ liệu là một phần quan trọng trong việc kiểm thử cơ sở dữ liệu. Vì vậy, đối với các ứng dụng web nên được kiểm tra một cách kỹ lưỡng. Các hoạt động kiểm tra bao gồm:

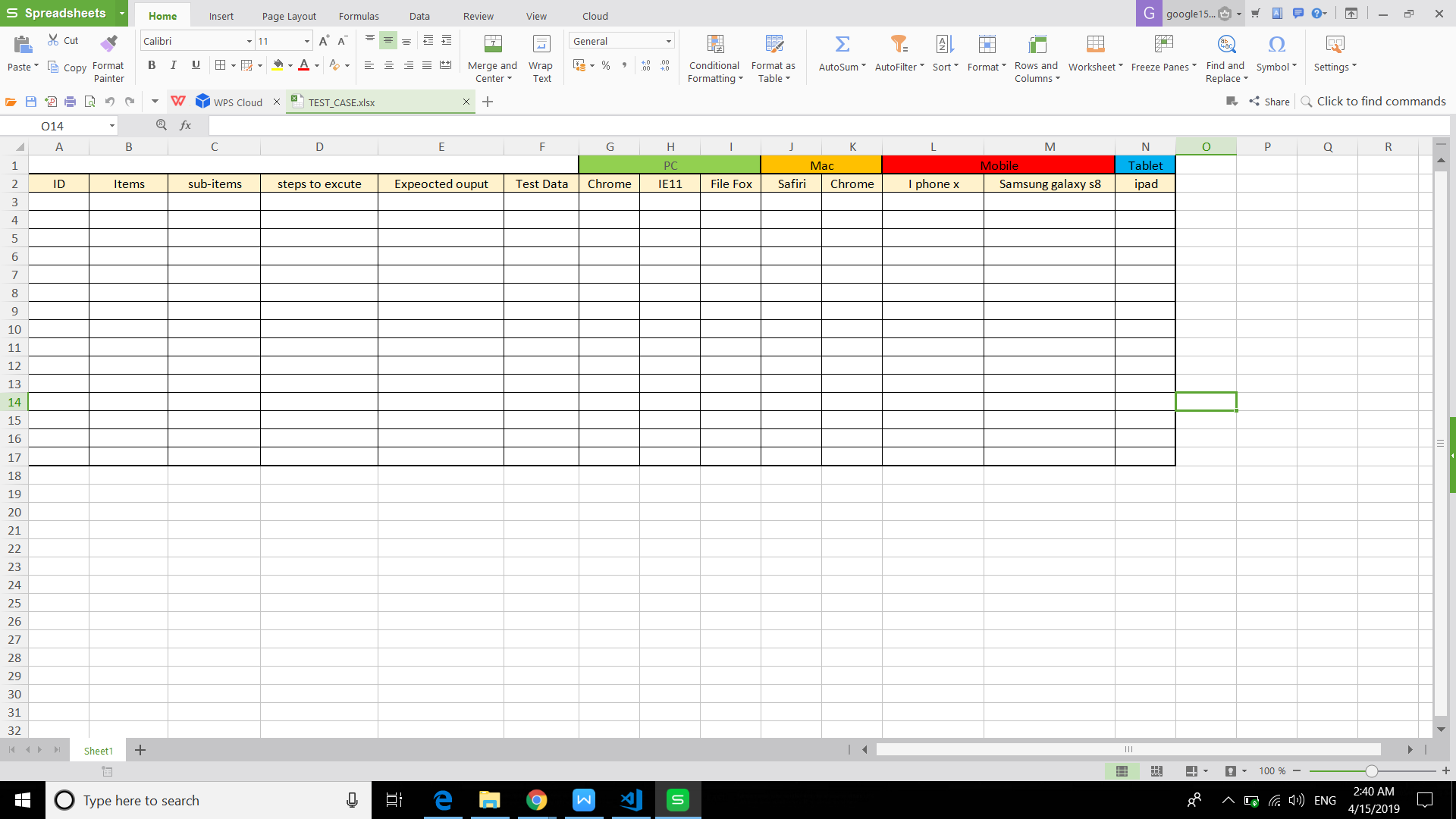
* Kiểm tra nếu các truy vấn được thực hiện mà không xảy ra lỗi.
* Thêm mới, cập nhật hoặc xoá dữ liệu trong cơ sở dữ liệu nên duy trì tính toàn vẹn của dữ liệu.
* Truy vấn dữ liệu không nên mất quá nhiều thời gian.
* Kiểm tra việc load dữ liệu và kết quả nhận được với các câu truy vấn dài.
* Dữ liệu nhận được trên cơ sở dữ liệu và hiển thị trên website có chính xác hay không.

2.1.4.9 Kiểm thử giao diện

Kiểm thử giao diện chủ yếu có ba lĩnh vực cần được kiểm tra: Web Server, Application server và Database server. Đảm bảo rằng tất cả các thông tin liên lạc giữa các server này phải được thực hiện đúng, xác minh kết nối giữa các máy chủ được thiết lập lại hoặc bị mất, kiểm tra xem có bất kỳ xung đột giữa lúc ứng dụng đang hoạt động, trả về bất kỳ lỗi từ web server hoặc database server đến application server sau đó được xử lý và cuối cùng là hiển thị kết quả tới người dùng.

* Web server: kiểm tra xem tất cả các yêu cầu web có đang được chấp nhận và không yêu cầu nào bị từ chối hoặc bị rò rỉ.
* Application server: kiểm tra xem yêu cầu có đang gửi đúng đến server, lỗi có được bắt và hiển thị cho người quản trị.
* Database server: kiểm tra kết quả truy vấn cơ sở dữ liệu.

2.1.4.10 Viết Test case để kiểm thử



*(Hình 2.1.4 Test case kiểm thử)*

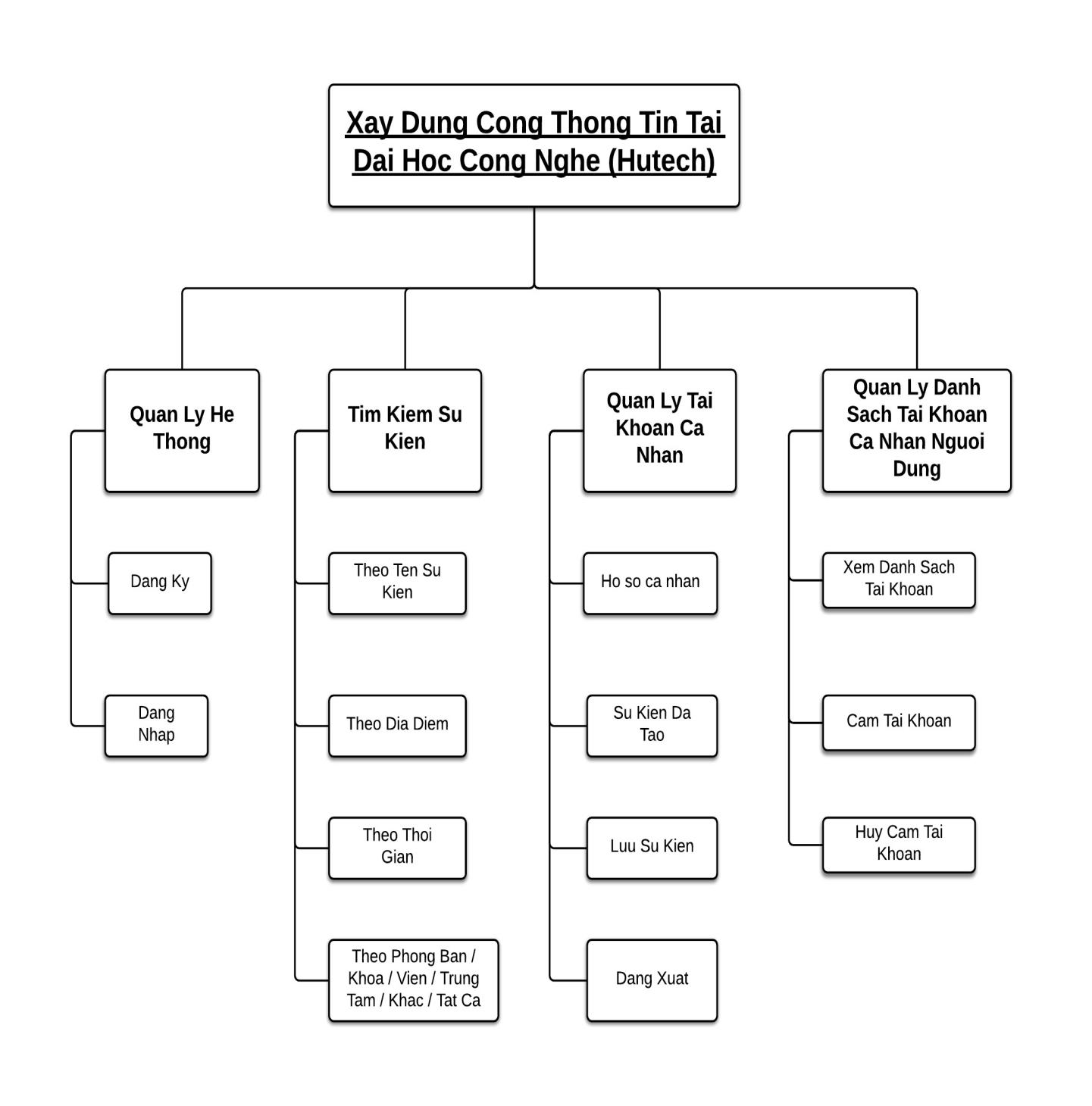
**2.2 Xác định các chức năng chính**

Sau những cuộc thảo luận, chúng em đã nghiên cứu phân tích những Website tương tự và những yêu cầu … đặt ra. Dựa trên những yếu tố đó, nhóm đã đúc kết được những chức năng chính cần có của một Website “Cổng thông tin tại Đại Học Công Nghệ (Hutech)”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **NHÓM USECASE** | **USECASE** | **MÔ TẢ** | **ĐỘ KHÓ** |
| 1 | Đăng nhập | Đăng nhập | Xử lý đăng nhập vào hệ thống | Dễ |
| Đăng nhập bằng Google | Xử lý đăng nhập băng google vào hệ thống | Khó |
| Ghi nhớ đăng nhập | Xử lý ghi nhớ đăng nhập | Dễ |
| Quên mật khẩu | Xử lý quên mật khẩu bằng email cá nhân | Khó |
| 2 | Đăng ký | Đăng ký | Xử lý đăng ký vào hệ thống | Dễ |
| Đăng ký bằng Google | Xử lý đăng ký bằng tài khoản google | Khó |
| 3 | Tìm kiếm | Tìm kiếm theo địa điểm | Xử lý tìm kiếm theo cơ sở của trường | Trung bình |
| Tìm kiếm theo tên sự kiện | Xử lý tìm kiếm theo tên sự kiện | Trung bình |
| Tìm kiếm theo ngày | Thực hiện tìm kiếm sự kiên theo hôm nay / ngày mai / tuần này / tháng. | Trung bình |
| Tìm kiếm theo chủ đề | Thực hiện tìm kiếm sự kiện theo phòng-ban / khoa /viện /trung tâm / khác / tất cả | Trung bình |
| 4 | Người dùng quản lý tài khoản cá nhân | Hồ sơ cá nhân | Xử lý xem xóa sửa thông tin của người dùng | Trung bình |
| Đổi mật khẩu | Xử lý yêu cầu đổi mật khẩu người dùng | Dễ |
| Sự kiện đã tạo | Thực hiện hiển thị sự kiện đã tạo của người dùng | Khó |
| Lưu sự kiện | Xử lý lưu trữ sự kiện của người dùng quan tâm | Khó |
| Đăng xuất | Xử lý đăng xuất tài khoản người dùng ra khỏi hệ thống | Dễ |
| 5 | Sự kiện đã tạo | Tạo sự kiện | Xử lý tạo sự kiện của chính người dùng | Khó |
| Chỉnh sửa sự kiện | Xử lý thực hiện yêu cầu chỉnh sửa sự kiện | Khó |
| Xóa sự kiện | Thực hiện yêu cầu xóa sự kiện | Dễ |
| 6 | Quản trị quản lý danh sách tài khoản cá nhân người dùng | Xem danh sách tài khoản cá nhân | Xử lý yêu cầu xem danh sách tất cả các tài khoản người dùng | Dễ |
| Cấm một tài khoản cá nhân truy câp | Xử lý cấm và thông báo người dùng tài khoản đó truy cập | Khó |
| Hủy cấm một tài khoản cá nhân | Xử lý hủy cấm một tài khoản | Khó |

**2.3 Mô hình giải pháp**

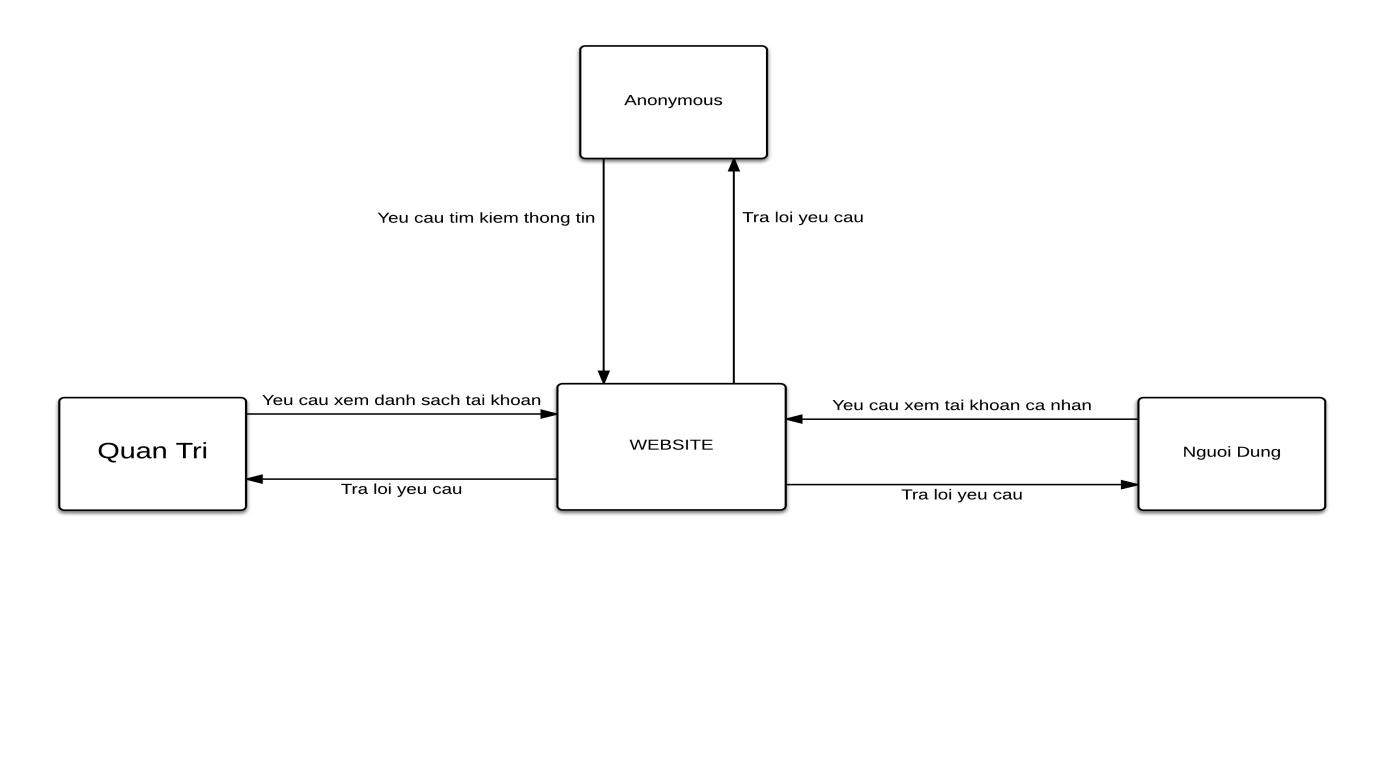
*2.3.1 Mô hình phân rã chức năng BFD*



*(Hình 2.1.1.Mô hình phân rã chức năng BFD)*

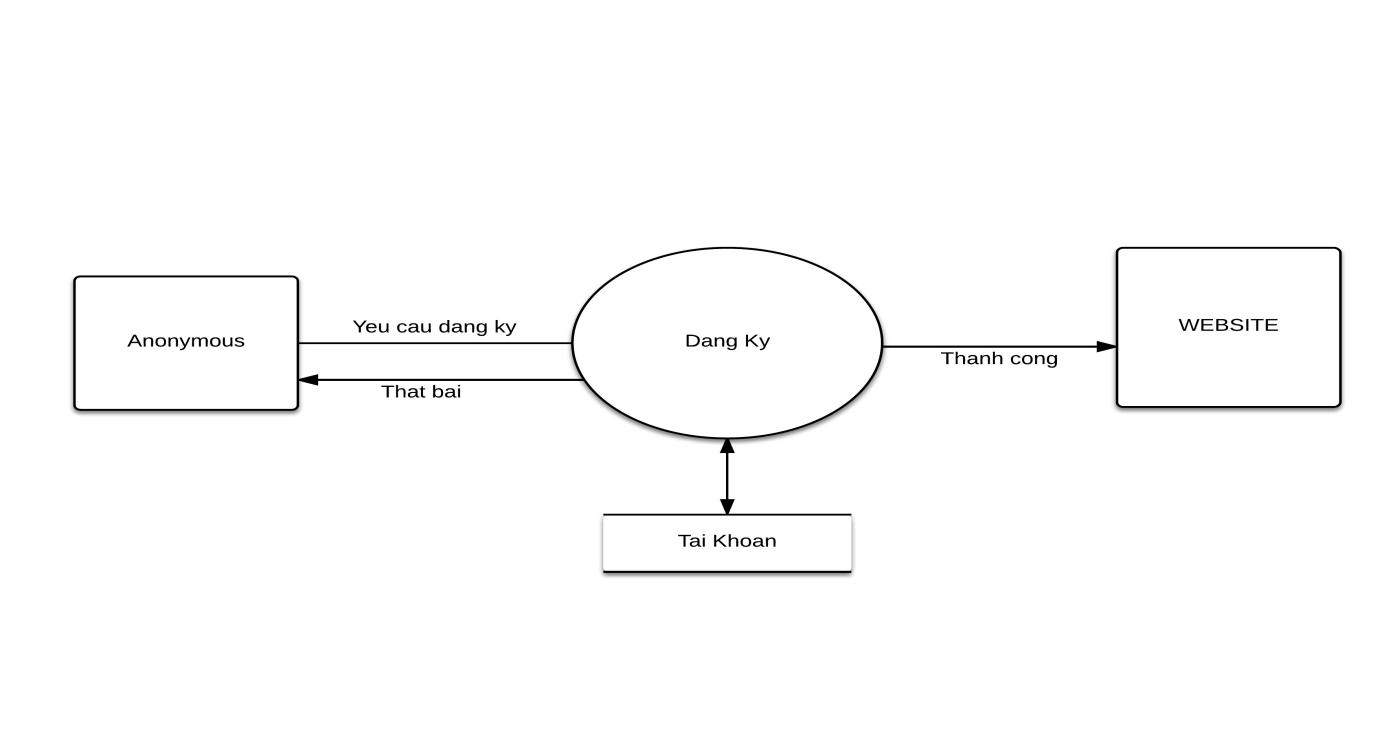
*2.3.2 Mô hình luồng dữ liệu*

2.3.2.1 Sơ đồ mức ngữ cảnh

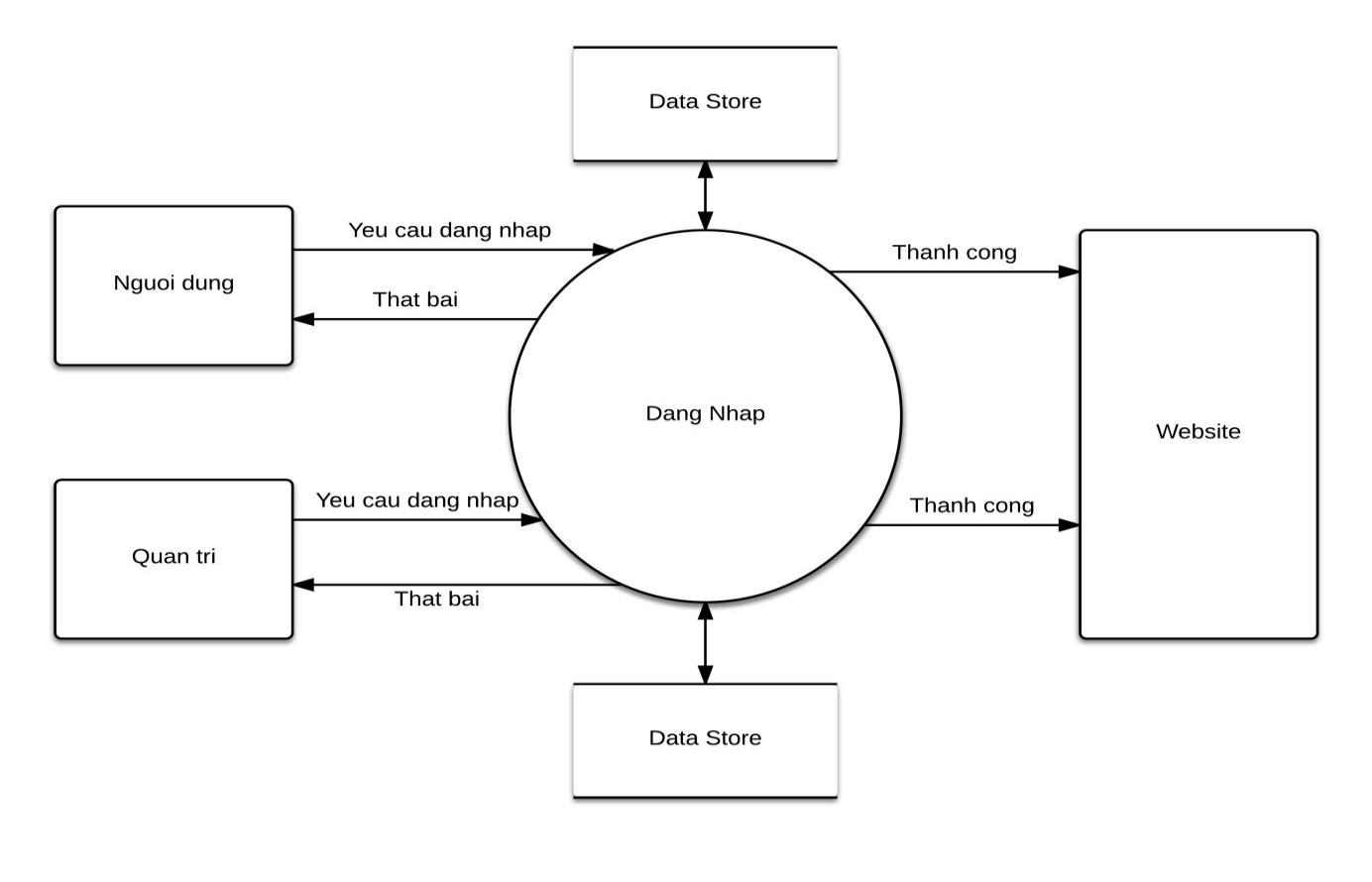


*(Hình 2.2.1 Sơ đồ mức ngữ cảnh)*

2.3.2.2 Đăng ký-Đăng nhập

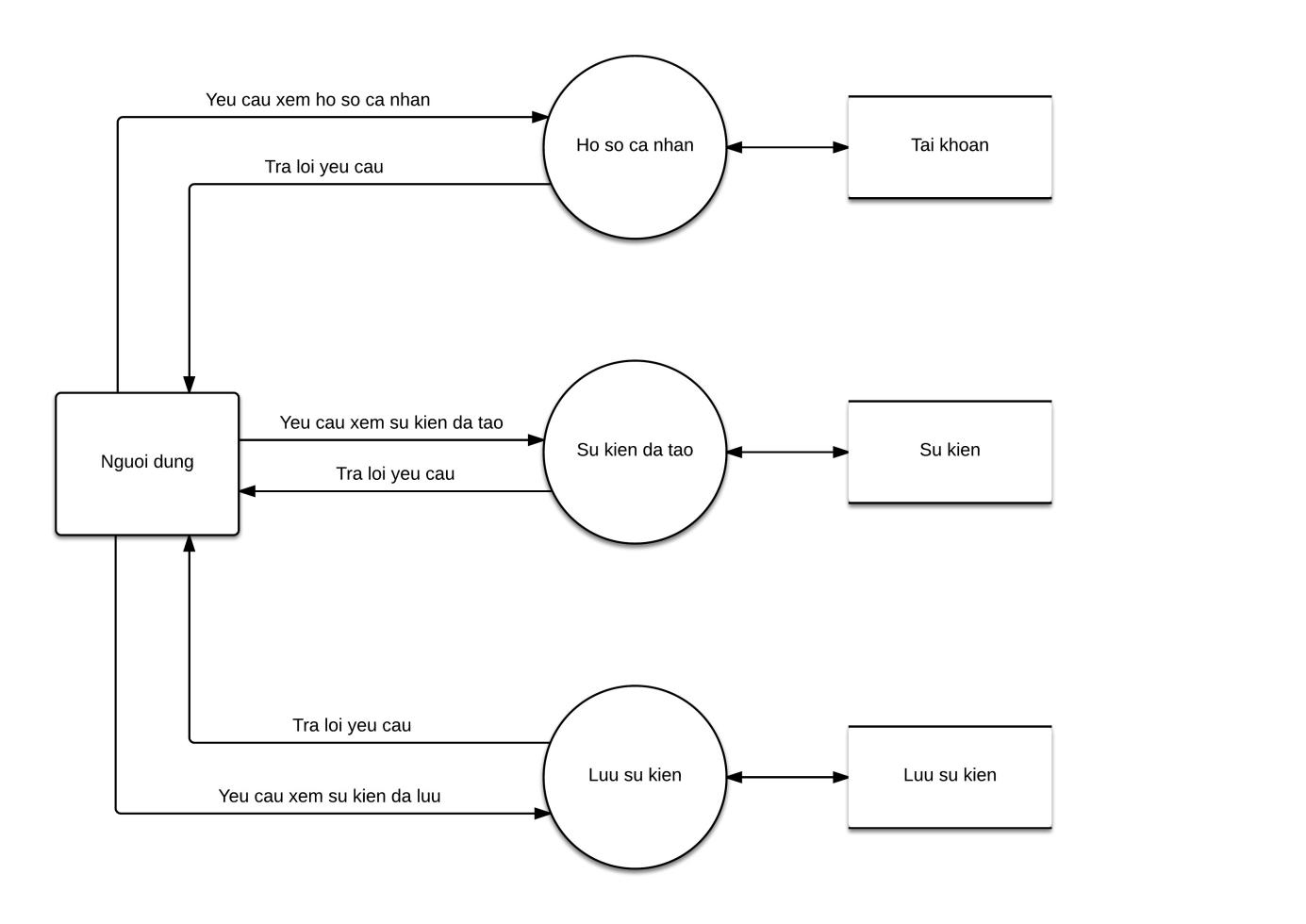


*(Hình 2.2.2.1Mô hình phần đăng ký)*



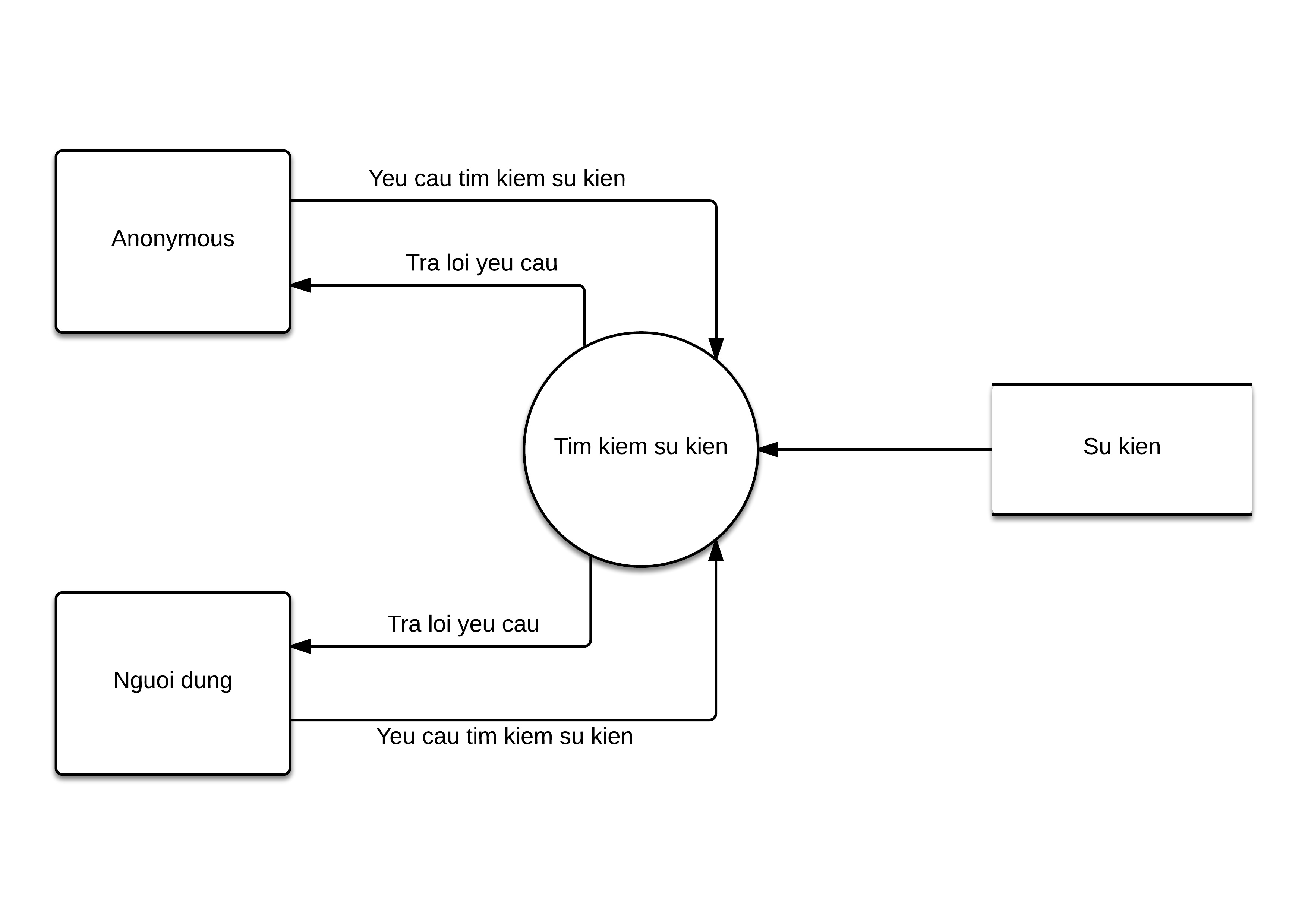
*(Hình 2.2.2.1 Mô hình phần đăng nhập)*

2.3.2.3 Quản lý tài khoản cá nhân



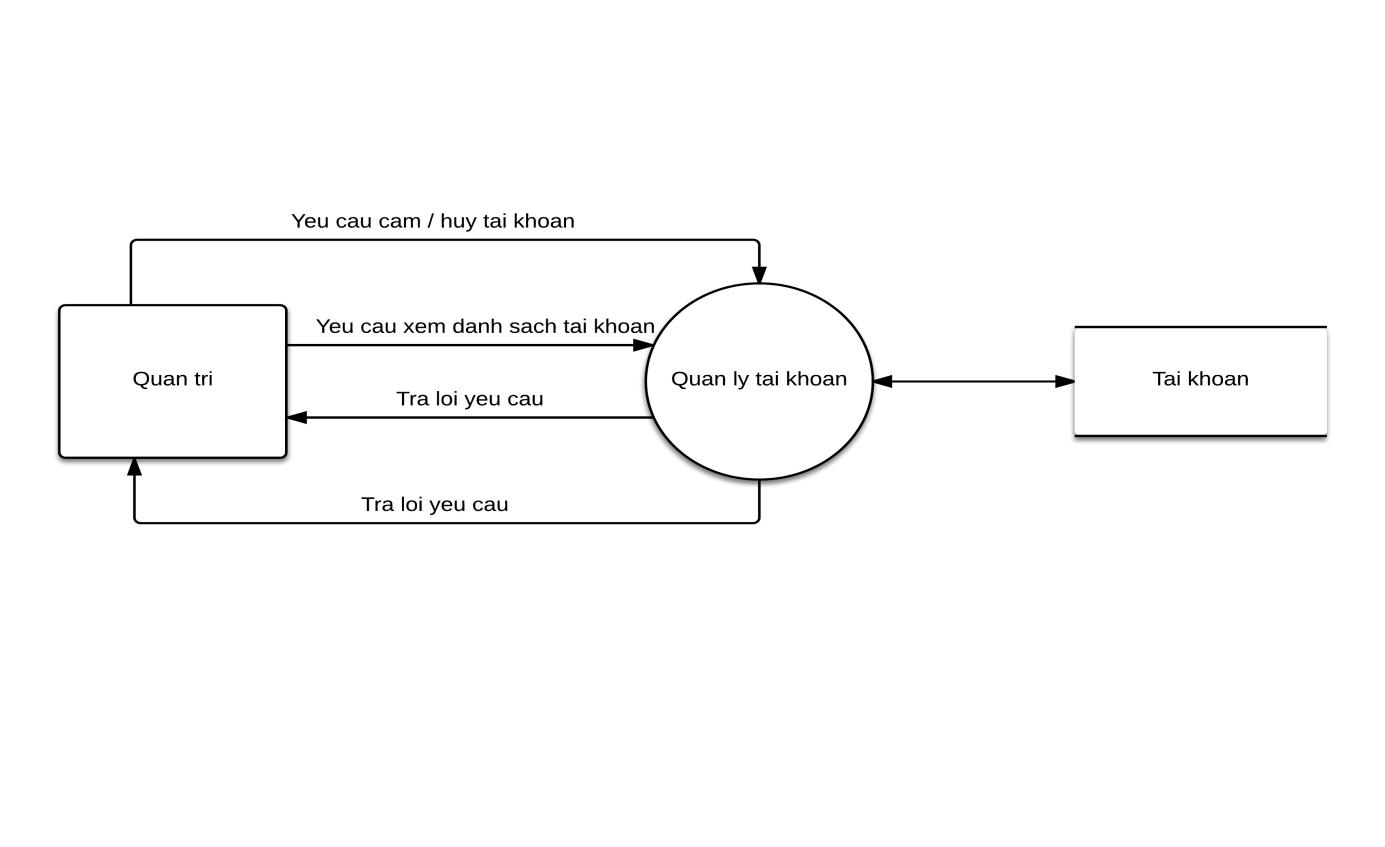
*(Hình 2.2.3 Mô hình quản lý tài khoản cá nhân)*

2.3.2.4 Tìm kiếm sự kiện



*(Hình 2.2.4 Mô hình tìm kiếm sự kiện)*

2.3.2.5 Quản lý danh sách tài khoản cá nhân



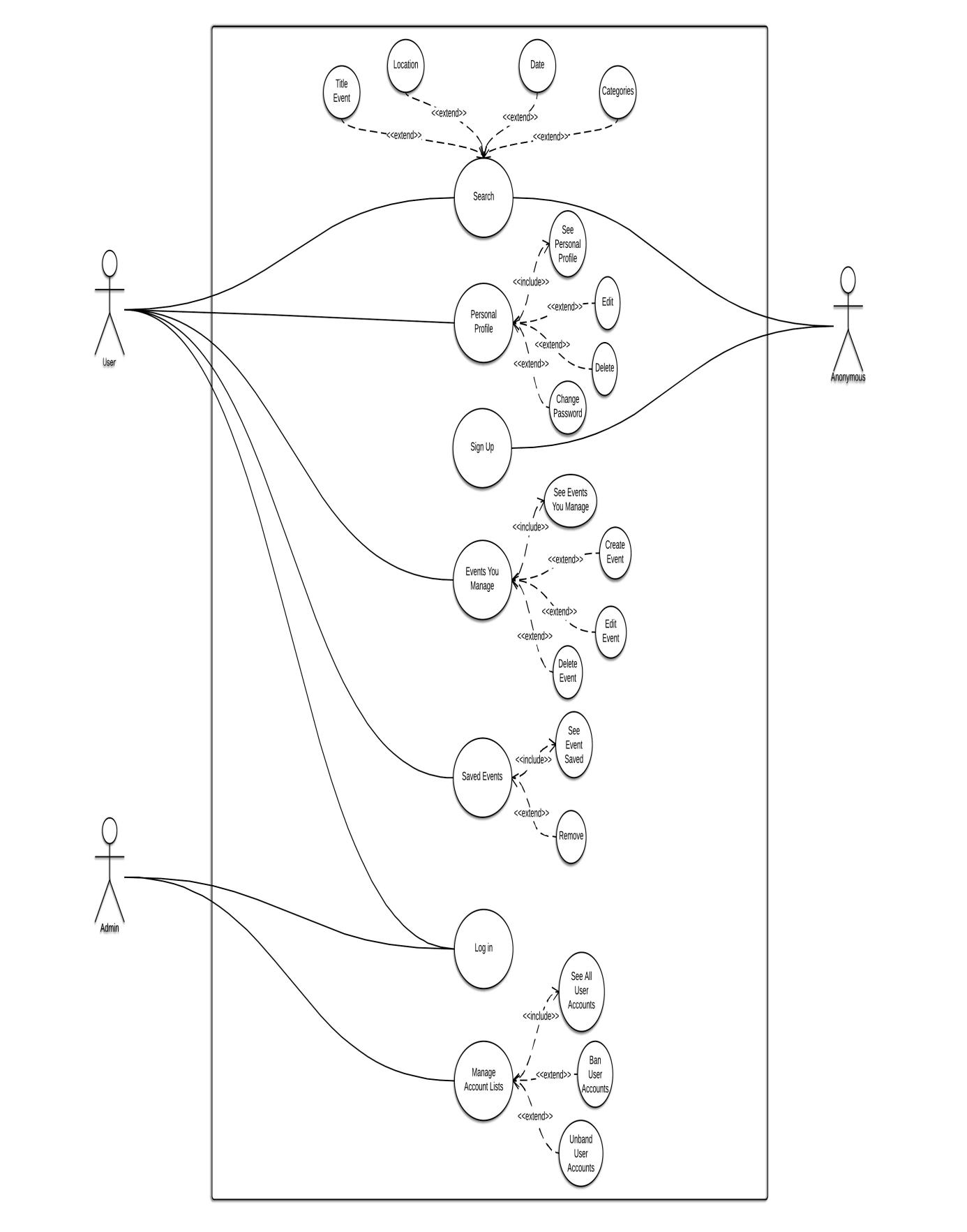
*(Hình 2.2.5 Mô hình quản lý danh sách tài khoản cá nhân)*

*2.3.3 Mô hình dữ liệu quan hệ*



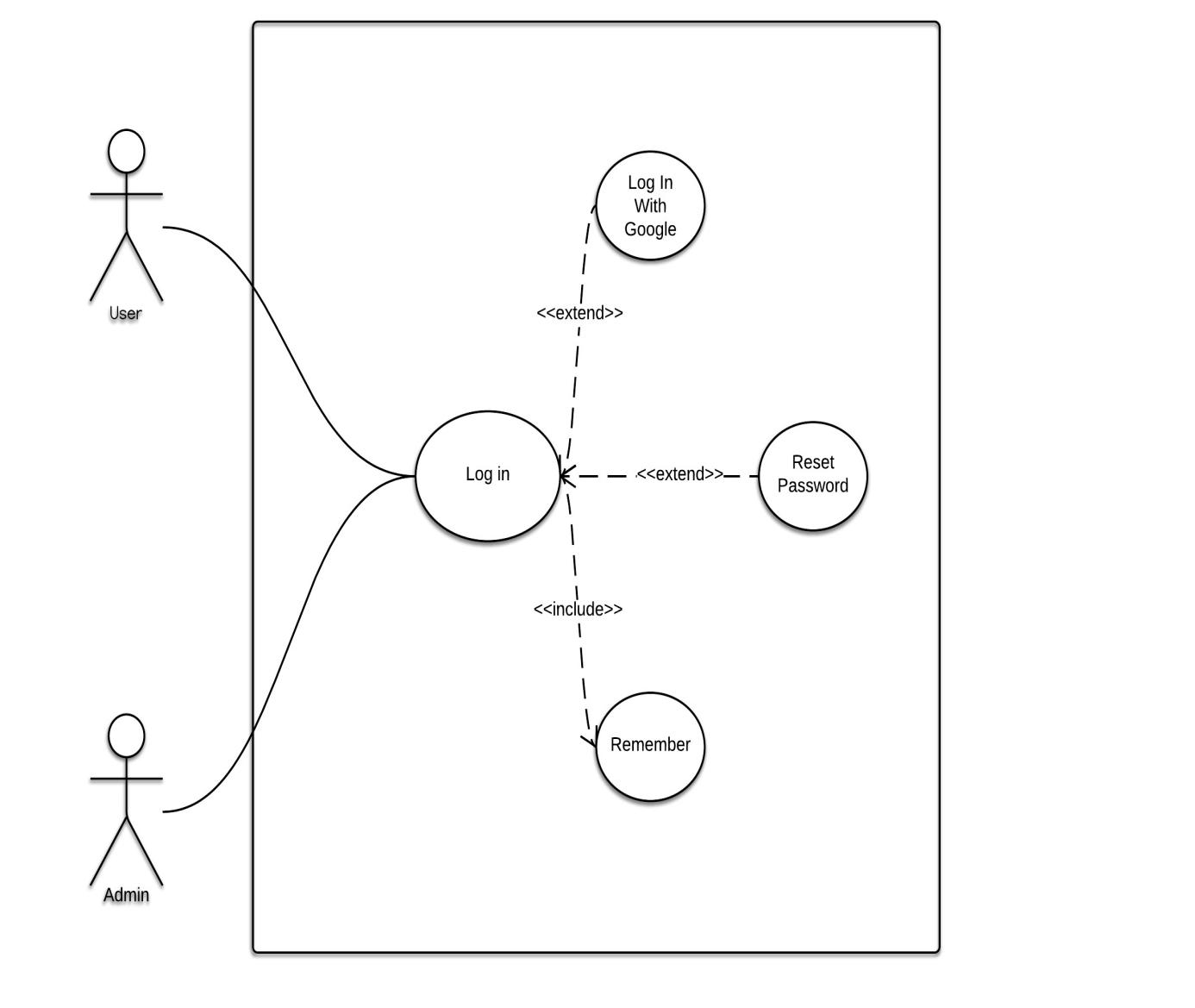
*(Hình 2.2.6 Mô hình dữ liệu quan hệ)*

*2.3.3 Mô hình Use Case*

**

*(Hình 2.2.7 Mô hình Use Case)*

2.3.3.1 Mô hình Use Case đăng nhập



*(Hình 2.2.8 Mô hình Use Case đăng nhập)*

**2.4 Đặc tả yêu cầu**

*2.4.1 Đặc tả yêu cầu hệ thống*

\_ Hệ thống có thể đáp ứng nhiều lượt truy cập cùng lúc

\_ Hệ thống cơ sở dữ liệu có thể lưu trữ một lượng lớn tài khoản người dùng

\_ Tốc độ xử lý nhanh hơn, khả năng truy cập tốt hơn

\_ Sử dụng công nghệ bảo mật tiên tiến

\_ Có sự cập nhật thông tin liên tục,chính xác

*2.4.2 Đặc tả yêu cầu giao diện*

\_ Màu sắc được thiết kế phù hợp với từng loại sự kiện

\_ Thiết kế các thanh công cụ phù hợp và tiện lợi cho người sử dụng

\_ Hình ảnh kích cỡ phù hợp

\_ Font chữ không quá cầu kì dễ nhìn dễ đọc

\_ Sự kiện được thiết kế và sắp xếp tối ưu các nội dung chính cho người truy cập

*2.4.3 Đặc tả yêu cầu chức năng*

2.4.3.1 Đăng ký

Mô tả : Người dùng phải nhập : Họ và tên, Email, Password, Confrim Password. Sau đó người dùng mới tạo được tài khoản.

Quy định : Họ và tên không được để trống, Email phải ghi đúng cấu trúc email (vd: xxx@gmail.com), Password, Confrim Password phải hiện ký tự “\*” khi người dùng nhập để bảo mật.

2.4.3.2 Đăng nhập

Mô tả : Người dùng cần nhập đúng tên đăng nhập và mật khẩu. Sau khi đăng nhập thì người dùng mới có quyền thực hiện chức năng mà nhà quản trị phân quyền.

Quy định : Tên đăng nhập không được để trống và phải giống kiểu cấu trúc email (vd: xxx@gmail.com). Mật khẩu không được để trống và phải hiện ký tự “\*” khi người dùng nhập để bảo mật.

2.4.3.3 Tìm kiếm

Mô tả : Người dùng có thể nhập tên sự kiện, hay chọn thanh danh mục theo Phòng-Ban / Khoa / Viện / Trung tâm / Khác / Tất cả để tìm kiếm sự kiện. Hay chọn danh mục theo Hôm nay / Ngày mai / Tuần này để tìm kiếm sự kiện. Hay chọn danh mục địa điểm cơ sở A / cơ sở B / cơ sở E / cơ sở U để tìm kiếm sự kiện.

Quy định : Có thẻ nhập hoặc chọn thanh danh mục để tìm kiếm không bắt buộc.

2.4.3.4 Tạo sự kiện

Mô tả : Người dùng có thẻ xem, xóa, sửa sự kiện theo ý mình.

Quy định : Chức năng này người dùng phải tạo tài khoản để sử dụng.

2.4.3.5 Hồ sơ cá nhân

Mô tả : Người dùng có thể xem, xóa, sửa thông tin cá nhân của mình.

Quy định : Người dùng phải có tài khoản cá nhân.

2.4.3.6 Đổi mật khẩu

Mô tả : người dùng phải nhập đúng Password cũ, phải nhập Password mới và Confrim Password phải nhập đúng. Sau khi đổi mật khẩu thì tài khoản sẽ quay về Người dùng quản lý tài khoản cá nhân và có thông báo Password đã được thay đổi.

Quy định : Người dùng phải có tài khoản cá nhân. Passoword cũ, Password mới, Confrim Password khi người dùng nhập phải mã hóa kí tự thành “\*”.

2.4.3.7 Lưa sự kiện

Mô tả : Hiện ra danh sách các sự kiện kiện đã lưu, thể hiện chi tiết thông tin sự kiện đó

Quy định : Người dùng phải có tài khoản cá nhân, và phải bấm nút lưu sự kiện để thực hiện chức năng. Và chỉ có thể xem và hủy sự kiện đã lưu.

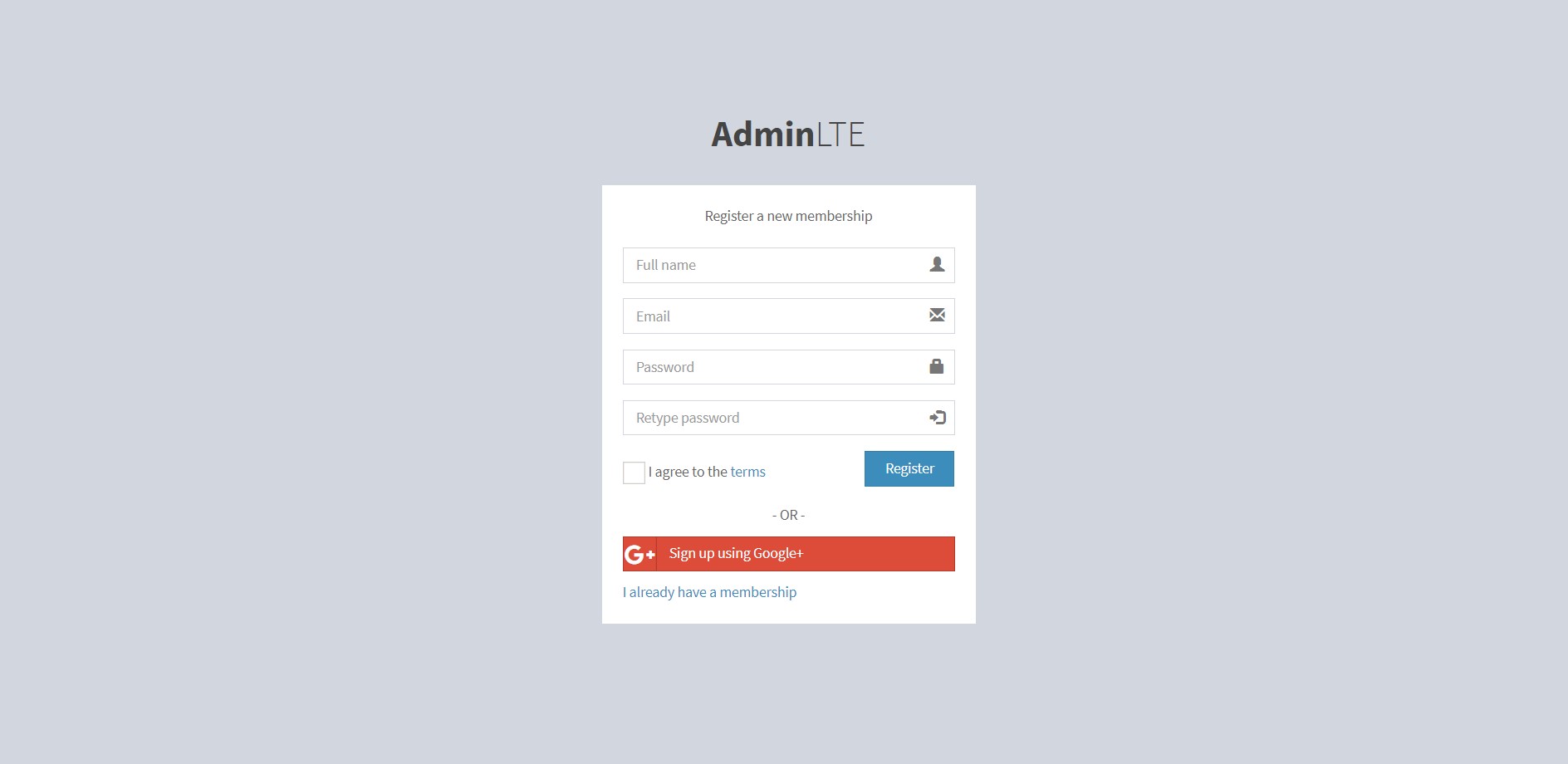
2.4.3.8 Quản trị quản lý danh sách tài khoản cá nhân người dùng

Mô tả : Khi đăng vào tài khoản được phân quyền quản trị. Thì người quản trị có thêm chức năng đã ẩn là xem danh sách tài khoản cá nhân người dùng đã đăng ký. Có quyền cấm một tài khoản khi tài khoản đã đăng nhập và sẽ nhận được thông báo.

Quy định : Chỉ cho phép danh xem danh sách số lượng tài khoản người dùng không được đăng nhập tài khoản đó để thực thi một mục đích khác. Password sẽ được mã hóa thành “\*” để bào mật cho người dùng.

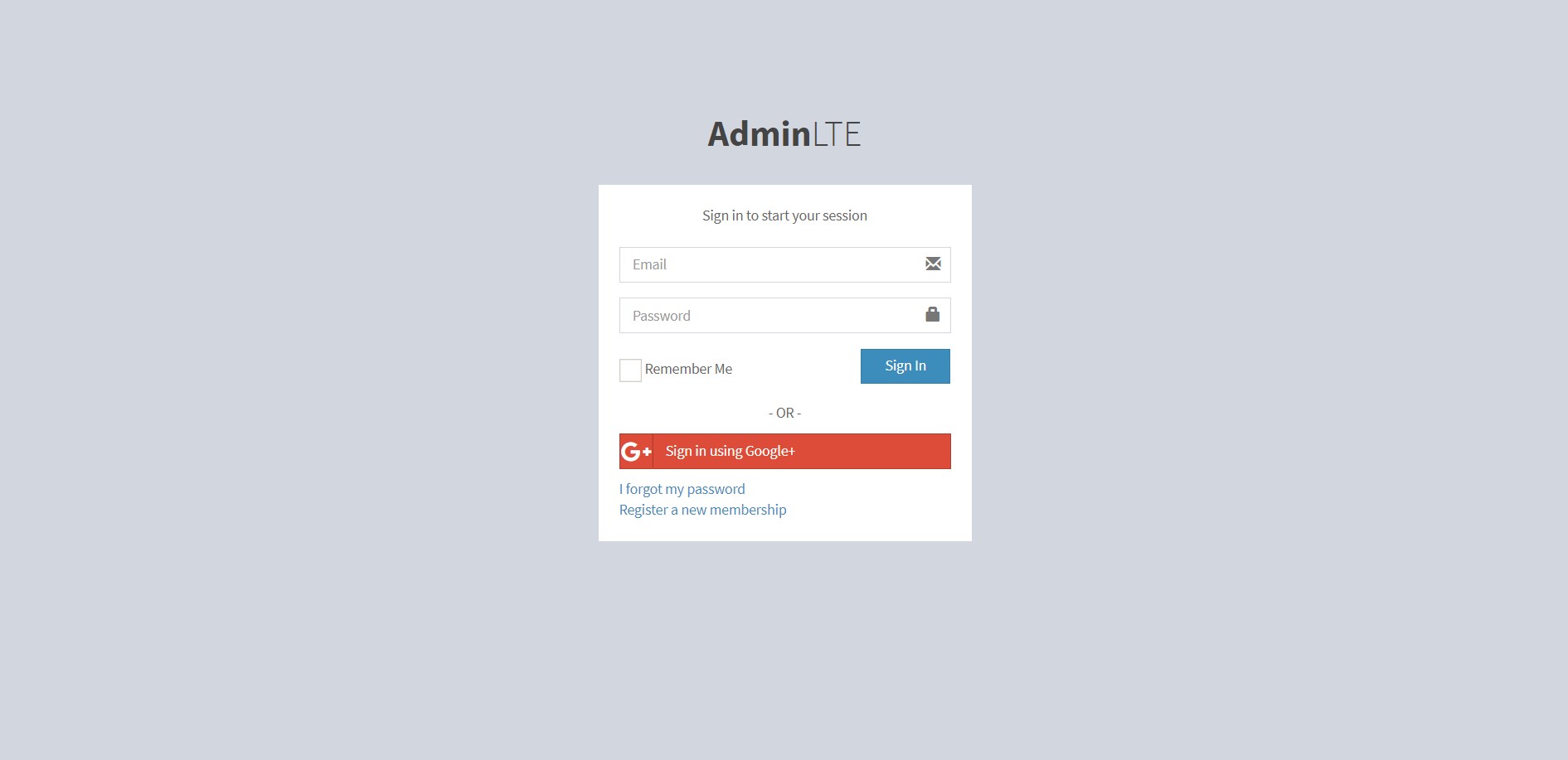
**CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM**

**3.1 Thiết kế giao diện form đăng ký**



*(Hình 3.1. Giao diện form đăng ký demo)*

**3.2 Thiết kế giao diện form đăng nhập**



*(Hình 3.2. Giao diện form đăng nhập demo)*

**CHƯƠNG 4. KẾT LUẬN VÀ KIẾN NGHỊ**

**4.1 Kết quả đạt được**

Xây dụng được một trang webiste nói chung và hoàn thành đề tài nói riêng, kết quả đạt được không chỉ là một công việc xây dựng một trang web mà nó là cả một quá trình cố gắng không biết mỏi mệt trong vòng 12 tuần của nhóm.

Trước một vấn đề về nhu cầu thông tin của sinh viên và sự phát triển mạnh mẽ của công nghệ thông tin thì việc tạo ra một trang web tin tức riêng giành cho các trường đại học là điều cần thiết và mang lại hiệu quả thực tế cao. Qua việc xây dựng là thực hiện đồ án cơ sở cũng giúp chúng em học hỏi thêm được nhiều điều về công việc của lập trình viên và nắm bắt được làm một website là như thế nào từ đó giúp chúng em có được tiền đề để phát triển thêm nhiều website khác.

**4.1 Kết quả chưa đạt được**

**4.2 Đánh giá phần mềm**

*4.2.1 Ưu điểm*

* Tổng hợp được tất cả các thông tin về các hội thảo,các sự kiện,hội thao…v..v..và các hoạt động khác có liên quan đến trường Đại Học Công Nghệ TP.HCM.
* Tốc độ xử lý nhanh,cơ sở dữ liệu đủ lớn để đáp ứng cùng lúc rất đông người truy cập
* Sử dụng công nghệ bảo mật tiên tiến
* Có sự cập nhật thông tin liên tục, chính xác
* Font chữ không quá cầu kì dễ đọc
* Sự xắp xếp các thanh công cụ thân thiện hợp lý

*4.2.2 Nhược điểm*

* + Giao diện vẫn chưa có điểm nổi trội so với nhiều website tin tức khác