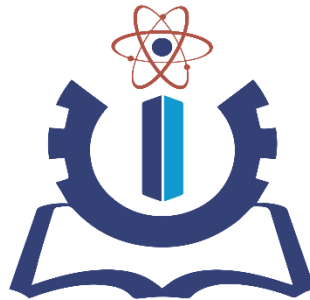


TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP VIỆT TRÌ
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO THỰC TẬP TỐT NGHIỆP

Đề tài

**NGHIÊN CỨU XÂY DỰNG WEBSITE HỎI ĐÁP TRỰC TUYẾN
SỬ DỤNG CÔNG NGHỆ ASP.NET MÔ HÌNH MVC**

Giảng viên hướng dẫn : *ThS. Đào Minh Sang*

Sinh viên thực hiện: *Phạm Ngọc Nam*

Lớp: *TT1Đ19*

Mã số sinh viên: *19A3403010027*

Phú Thọ, Ngày....tháng....năm...

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

Ngày Tháng Năm

MUC LUC

I. PHẦN MỞ ĐẦU	5
1. Lý do chọn đề tài.....	5
2. Mục tiêu và nhiệm vụ nghiên cứu	6
3. Đối tượng và khách thể nghiên cứu.....	6
4. Giới hạn đề tài.....	6
5. Phương pháp nghiên cứu.	7
II. PHẦN NỘI DUNG	8
CHƯƠNG 1. CƠ SỞ LÝ THUYẾT.....	8
1.1 Tìm hiểu về ngôn ngữ lập trình.	8
MVC ra đời khi nào?.....	11
Kiến trúc MVC.....	11
View.....	12
Controller	12
Model	12
Sự tương tác giữa các thành phần	12
Ưu điểm của MVC:	13
Nhược điểm của việc sử dụng MVC.....	13
Kết luận	13
CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG WEBSITE.....	14
2.1 Thiết kế cơ sở dữ liệu	14
2.2. Thiết kế hệ thống.	21
CHƯƠNG 3: THIẾT KẾ,CÀI ĐẶT WEBSITE	27
3.1 Cài đặt các công cụ hỗ trợ.	27
KẾT LUẬN	29
Kết quả đạt được:.....	29
Những gì chưa đạt được:	29
Hướng phát triển:	29
Các loại tài liệu tham khảo:	29

LỜI CẢM ƠN

Em xin chân thành cảm ơn thầy giáo: Đào Minh Sang là giáo viên hướng dẫn trong đề tài mà em thực hiện, em rất trân trọng sự giúp đỡ nhiệt tình của thầy để em có thể hoàn thành đề tài này.

Em cũng xin cảm ơn khoa Công Nghệ Thông Tin đã tạo điều kiện tốt nhất để em có được những tiết học tuyệt vời.

Em cũng xin cảm ơn trường Đại Học Công Nghiệp Việt Trì là ngôi trường tuyệt đẹp và tốt nhất mà chúng em từng học. Xin cảm ơn trường vì những đóng góp đã giúp em có được kiến thức không chỉ về mặt học vấn mà còn những kiến thức về xã hội để giúp em có thể vững bước trên con đường của chính mình.

Sinh viên thực hiện

I. PHẦN MỞ ĐẦU

1. Lý do chọn đề tài

Nếu ai đã từng sử dụng Internet chắc sẽ không ít lần nghe hoặc sử dụng dịch vụ nhóm thảo luận trên Internet (NewsGroup), đây là dịch vụ khá phổ biến hiện nay. Khi ta gặp một vấn đề thắc mắc, hoặc đang tìm kiếm một thông tin gì đó, thì NewsGroup sẽ là người trợ giúp đắc lực cho chúng ta. Dịch vụ này cho phép người dùng gửi lên mạng Internet những câu hỏi hoặc những thắc mắc của mình, sau đó nếu một người nào đó, ở đâu đó trên thế giới cũng tham gia vào nhóm thảo luận đó nếu họ đọc được được mẫu tin của chúng ta và biết về thông tin về vấn đề đó họ sẽ trả lời cho ta và ngược lại ta cũng có thể trở thành một người trả lời. Thật là thú vị khi được mọi người trên toàn cầu giúp ta giải quyết một vấn đề.

Ngày nay các NewsGroup thường được tích hợp luôn trên web nên tạo điều kiện thuận lợi hơn cho người sử dụng và người ta gọi đó là các Diễn đàn trên web (web forum).

Trên mạng Internet hiện đang có rất nhiều diễn đàn nói về nhiều chủ đề khác nhau. Ví dụ trang www.stackoverflow.com đây là nơi mà lập trình viên trên khắp thế giới có thể đặt câu hỏi về bugs hoặc những khó khăn mà họ chưa giải quyết được. Trong một môi trường bao gồm số lượng lớn lập trình viên trên khắp thế giới, những vấn đề mà bạn đang thắc mắc có thể được giải đáp một cách nhanh chóng. Chính vì vậy, Stack Overflow được xem là một nguồn tài nguyên vô cùng quý báu và cần thiết mà các lập trình viên có thể tham khảo, học hỏi.

Tác dụng lớn nhất của Diễn đàn trên web là mọi người khi tham gia vào mạng có thể đặt ra một vấn đề gì đó theo một chủ đề nhất định để có thể nhận được câu trả lời của nhiều người trên toàn thế giới.

Để có thể thấy được lợi ích về tính quảng bá và khối lượng thông tin, kiến thức đồ sộ trên diễn đàn ta có thể so sánh dịch vụ này với dịch vụ E-mail như sau: Nếu như nói thông tin trao đổi qua mail chỉ là quan hệ 1-1, tức là chỉ có một người gửi và chỉ có một người, hoặc một nhóm người nhận thư. Trong khi đó Diễn đàn trên web lại có quan hệ 1-n, tức là một người hỏi có nhiều người trả lời. Như vậy lợi ích mà Diễn đàn trên web đem lại cho người sử dụng là không nhỏ.

Ngày nay Diễn đàn trên Web được ứng dụng rất nhiều nơi, ví dụ trong một trường đại học, diễn đàn sẽ giúp đỡ các sinh viên trao đổi thảo luận với nhau qua mạng, hay trong mạng nội bộ của một cơ quan...

Xuất phát từ lợi ích mà diễn đàn đem lại, với mong muốn tạo ra một ứng dụng trên mạng, giúp mọi người trao đổi thông tin với nhau nên em chọn đề tài: **“NGHIÊN CỨU XÂY DỰNG WEBSITE HỎI ĐÁP TRỰC TUYẾN SỬ DỤNG CÔNG NGHỆ ASP.NET MÔ HÌNH MVC”**.

2. Mục tiêu và nhiệm vụ nghiên cứu

2.1 Mục tiêu

Mục tiêu của website diễn đàn đặt ra là :

- Người dùng có thể đưa ra những câu hỏi, ý kiến của mình.
- Người dùng có thể đưa ra câu trả lời cho câu hỏi, ý kiến của người khác.
- Tạo ra một môi trường an toàn, lành mạnh, bảo mật dành cho người dùng tìm kiếm thông tin.
- Tạo ra nơi để mọi người có thể giao lưu, trò chuyện, trao đổi thông tin qua mạng Internet

2.2 Nhiệm vụ nghiên cứu

- Nghiên cứu cơ sở lý thuyết và các ngôn ngữ lập trình C#
- Nghiên cứu công cụ xây dựng trang web và ứng dụng web ASP.NET và mô hình MVC,
- Xây dựng được trang chủ hiển thị được các chủ đề thảo luận mới nhất, nổi bật nhất,...
- Xây dựng chức năng trả lời, bình chọn, chỉnh sửa, chặn, báo cáo.
- Xây dựng chức năng yêu thích, theo dõi chủ đề, cuộc thảo luận.
- Xây dựng chức năng tìm kiếm chủ đề, cuộc thảo luận.
- Tìm hiểu các công cụ xây dựng hệ thống như: Visual Studio Code 2022, SQL 2019, Visual Studio 2022

3. Đối tượng và khách thể nghiên cứu.

3.1. Đối tượng nghiên cứu.

Website diễn đàn trao đổi tin tức trực tuyến.

3.2. Khách thể nghiên cứu.

Website diễn đàn bằng các ngôn ngữ lập trình ASP.NET, Mô hình MVC

4. Giới hạn đề tài.

Trang web có chức năng sau:

- Chức năng trả lời, bình chọn, chỉnh sửa, chặn, báo cáo.
- Chức năng yêu thích, theo dõi chủ đề, cuộc thảo luận.
- Chức năng tìm kiếm chủ đề, cuộc thảo luận.
- Chức năng tin nhắn riêng.

5. Phương pháp nghiên cứu.

- Tìm hiểu sách báo.
- Tìm hiểu ở mạng xã hội.
- Tìm hiểu, học hỏi từ các diễn đàn đã có trên mạng Internet.

II. PHẦN NỘI DUNG

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT.

1.1 Tìm hiểu về ngôn ngữ lập trình.

1.1.1 Giới thiệu về ASP.NET

Ta phải công nhận một điều là .NET Framework và các ứng dụng của nó đã và đang tạo một cuộc cách mạng kỹ thuật trong công nghệ tin học (Information Technology), thay đổi tận gốc rễ các kiểu mẫu lập trình hay phát triển và triển khai mạng trên thế giới và do đó tạo một vận hội mới đáp ứng mọi yêu cầu khẩn thiết cho các ngành nghề kỹ thuật và thương mại hiện nay cũng như vạch một hướng đi vững chắc và lâu dài cho tương lai tin học. ASP.NET chính là một trong những ứng dụng quan trọng nhất để phát triển và triển khai mạng một cách dễ dàng chưa từng thấy... từ xưa đến nay. Hãy lắng nghe chính Microsoft nói về ASP.NET như thế nào:

“ASP.NET is a revolutionary programming framework that enables the rapid development of powerful web applications and services. Part of the Microsoft .NET Platform, it provides the easiest and most scalable way to develop, deploy and run distributed web applications that can target any browser or any application.”

Chúng ta cùng tìm hiểu những đặc tính của ASP.NET:

- Easy Programming Model:

ASP.NET giúp chúng ta phát triển và triển khai các ứng dụng về mạng trong một thời gian kỷ lục vì nó cung cấp cho ta một kiểu mẫu lập trình dễ dàng và gọn gàng nhất. Ngoài ra, các trang ASP.NET còn làm việc với mọi browsers hiện nay như Internet Explorer (IE), Netscape, Opera, AOL...mà không cần đổi tới đổi lui các nguồn mã vật vờ như trước.

- Flexible Language Options:

Không như ASP kiểu cổ điển chỉ giới hạn với VBScripts and JScripts, ASP.NET yểm trợ trên 25 ngôn ngữ lập trình (dĩ nhiên ngoài các ngôn ngữ mới thiết lập đã cài sẵn yểm trợ .NET framework như là VB.NET, C# và JScript.NET còn có MC++.NET, Smalltalk.NET, COBOL.NET, Eiffel.NET, Perl.NET, Component Pascal.NET, Mercury.NET, Oberon.NET, Python.NET...).

- Great Tool Support:

Mặc dù ta có thể chỉ dùng đến Notepad để triển khai các trang ASP.NET nhưng Visual Studio.NET giúp năng suất triển khai mạng thêm phần hiệu quả vì ta có thể quan sát các kế hoạch của ta dễ dàng hơn khi phác họa các thành phần của ASP.NET bằng hình ảnh với ASP.NET Web Forms hay Services theo phương pháp “drag-drop-doubleclick” quen thuộc của nền Windows. Hơn nữa, còn yểm trợ ta trong việc phát

hiện và loại bỏ các lỗi sai một cách rất thuận lợi trong khi phát triển các ứng dụng về mạng (support for debugging and deploying ASP.NET Web applications).

- Rich Class Framework:

Nhờ nền tảng vững vàng và tài nguyên phong phú .NET Framework với hơn 5000 classes bao gồm đủ thứ như XML, data access, file upload, regular expressions, transactions, message queuing, SMTP mail, ... nên việc thiết kế các đặc tính trong một ứng dụng trở nên nhẹ nhàng và thoải mái hơn xưa rất nhiều.

- Compile execution:

ASP.NET không những chạy nhanh hơn ASP cổ điển gấp 5 lần mà còn có thể duy trì kiểu mẫu cập nhật gọi là kiểu mẫu “just hit save”, nghĩa là ASP.NET tự động dò tìm mọi sự thay đổi và compile files khi cần thiết cũng như lưu trữ kết quả compile đó để cung ứng dịch vụ cho những yêu cầu tiếp theo sau, nhờ vậy ứng dụng của bạn luôn luôn được cập nhật hóa và chạy với tốc độ nhanh.

- Rich output caching:

ASP.NET có khả năng lưu trữ một kết quả chung trong bộ phận memory của trang để gửi giải đáp cho cùng một yêu cầu từ nhiều khách hàng khác nhau và nhờ đó không những tiết kiệm được sự lặp đi lặp lại công tác thi hành của một trang web mà còn gia tăng hiệu suất một cách ngoạn mục do giới hạn tối đa việc chất vấn các cơ sở dữ liệu (eliminating the need to query the datab ase on every request) rất tốn nhiều thời gian.

- .NET Outperforms J2EE:

Trong việc đối đầu nhau về hiệu suất (performance) và scalability với cùng một ứng dụng phát triển giữa Sun's Java Pet Store J2EE và ASP.NET thì ASP.NET không những nhanh vượt trội hơn J2EE đến 28 lần (khoảng 2700%), nguoonfmaxlaij ít hơn nhiều (khoảng 1/4 nguồn mã của J2EE) mà còn dùng bộ xử lý (processor) chỉ khoảng 1/6 lần so với việc sử dụng processor của J2EE.

- Memory Leak, DeadLock và Crash protection:

ASP.NET cũng có khả năng dò tìm và phục hồi (detects and recovers) những trở ngại nghiêm trọng như deadlocks hay bộ nhớ bị rỉ để đảm bảo ứng dụng của bạn luôn luôn sẵn sàng khi dùng mà không làm cản trở việc cung ứng dịch vụ cần thiết thường lệ.

- Simple application deployment:

ASP.NET đơn giản hóa việc triển khai ứng dụng mạng, do đó biến việc triển khai toàn bộ ứng dụng trở nên dễ dàng và thuận tiện hơn hẳn trước kia vì bây giờ ta chỉ cần sao (với XCOPY) và lưu trữ ở server chứ không cần phải chạy chương trình “regsrv32” để đăng ký bất cứ thành phần nào cả, và thêm nữa khi cần lưu trữ những yếu tố phụ cần thiết cho việc thiết lập hay bố trí các ứng dụng, ta chỉ cần lưu trữ nó vào trong một hồ sơ dưới dạng XML là đủ.

- Dynamic update of running application:

ASP.NET cho phép ta tự động cập nhật hóa các thành phần đã compiled (compiled components) mà không cần phải khởi động lại các Web server.

- Easy Migration Path:

Ta không cần phải du nhập những ứng dụng được phát triển và triển khai bằng ASP cổ điển hiện có vào ASP.NET vì ASP.NET có thể chạy song song với ASP ở cùng một Internet Information Server (IIS) trong nền windows 2000 hay windowx XP. Các ứng dụng cũ vẫn được chạy hết sức thoải mái với ASP.DLL trong khi ASP.NET sẽ xử lý các ứng dụng mới. Ngoài ra, ASP.NET còn cho phép bạn dùng lại những thành phần thương mại hiện nay kiểu COM cổ điển trong các ứng dụng của nó.

- XML Web Services:

Dịch vụ về mạng với XML cũng cho phép bạn truyền đạt (communicate) và chia sẻ các dữ kiện xuyên qua mạng Internet dễ dàng hơn các SOAP client mà không hề phân biệt đối xử các hệ điều hành hay ngôn ngữ lập trình khác nhau (regardless of OS or programming language). Nhờ đó, ta không cần phải học thêm hay đào sâu các kiến thức về Networking, XML hay SOAP, ...

- Mobile Web Service Support:

Thêm nữa, ASP.NET Mobile Controls còn giúp ta phát triển và triển khai mạng nhắm vào thị trường những cell phone hay PDA với gần hơn 80 Mobile Web Services được cung cấp trong .NET framework. Bạn chỉ cần lập trình cho ứng dụng của bạn rồi phó mặc cho Mobile Controls đó tự động phát sinh ra những nguồn mã WAP/WML, HTML hay iMode thích hợp với từng loại thiết bị riêng biệt.

1.1.2 ASP.NET là gì :

Trước hết, họ tên của ASP.NET là Active Server Pages .NET (.NET đây là .NET framework). Nói đơn giản và ngắn gọn thì ASP.NET là một công nghệ có tính cách mạng dùng để phát triển các ứng dụng về mạng hiện nay cũng như trong tương lai (ASP.NET is a revolutionary technology for developing web applications). Lưu ý ở chỗ ASP.NET là một phương pháp tổ chức hay khung tổ chức (framework) để thiết lập

các ứng dụng hết sức hùng mạnh cho mạng dựa trên CLR (Common Language Runtime) chứ không phải là một ngôn ngữ lập trình. Ngôn ngữ lập trình được dùng để diễn đạt ASP.NET là VB.NET (Visual Basic .NET) và VB.NET chỉ là một trong 25 ngôn ngữ .NET hiện nay được dùng để phát triển các trang ASP.NET mà thôi.

Tuy mang họ tên gần giống như ASP cổ điển nhưng ASP.NET không phải là ASP. Ta sơ lược ở đây một vài khác biệt giữa ASP.NET và ASP để ta có một khái niệm tổng quát.

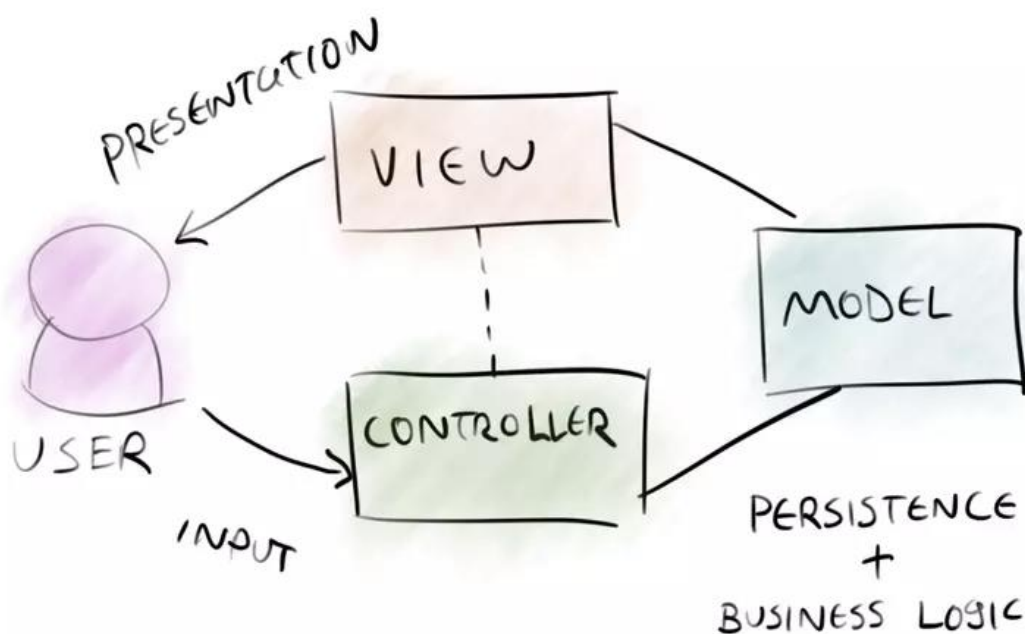
1.1.3 Mô hình MVC là gì :

Mô hình Model-View-Controller (MVC) là một mẫu kiến trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller. Do đó viết tắt MVC. Mỗi thành phần kiến trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng. MVC tách lớp logic nghiệp vụ và lớp hiển thị ra riêng biệt. Ngày nay, kiến trúc MVC đã trở nên phổ biến để thiết kế các ứng dụng web cũng như ứng dụng di động.

MVC ra đời khi nào?

- Kiến trúc MVC được thảo luận lần đầu vào năm 1979 bởi Trygve Reenskaug.
- Mô hình MVC được giới thiệu lần đầu tiên vào năm 1987 bằng ngôn ngữ lập trình Smalltalk.
- MVC lần đầu tiên được chấp nhận như một khái niệm chung, trong một bài báo năm 1988.
- Trong thời gian gần đây, MVC pattern được sử dụng rộng rãi trong các ứng dụng web hiện đại.

Kiến trúc MVC



MVC quan trọng bao gồm:

- Model: Nó bao gồm tất cả dữ liệu và logic liên quan của nó.
- View: Trình bày dữ liệu cho người dùng hoặc xử lý tương tác của người dùng.
- Controller: Là phần quan trọng nhất trong mô hình, nó liên kết phần Model và View.

View

- View là một phần của ứng dụng đại diện cho việc trình bày dữ liệu.
- View được tạo bởi các dữ liệu mà chúng ta lấy từ dữ liệu trong model. Một view yêu cầu model cung cấp đầy đủ dữ liệu để nó hiển thị đầu ra cho người dùng.
- View chính là nơi chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh... nó đảm nhiệm nhiệm vụ hiển thị dữ liệu và giúp người dùng tương tác với hệ thống.

Controller

- Controller là một phần của ứng dụng xử lý tương tác của người dùng. Bộ điều khiển diễn giải đầu vào chuột và bàn phím từ người dùng, thông báo cho model và view để thay đổi khi thích hợp.
- Controller là nơi tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/ function xử lý nhiều nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng nhờ lớp View.
- Controller gửi các lệnh đến model để làm thay đổi trạng thái của nó (Ví dụ: ta thêm mới 1 user hoặc cập nhật tên 1 user). Controller cũng gửi các lệnh đến view liên quan của nó để thay đổi cách hiển thị của view (Ví dụ: xem thông tin 1 user).

Model

- Thành phần model lưu trữ dữ liệu và logic liên quan của nó. Bao gồm các class function xử lý các tác vụ như truy vấn, thêm, sửa hoặc xóa dữ liệu. Ví dụ, một đối tượng Controller sẽ lấy thông tin khách hàng từ cơ sở dữ liệu. Nó thao tác dữ liệu và gửi trở lại cơ sở dữ liệu hoặc sử dụng nó để hiển thị dữ liệu.

Sự tương tác giữa các thành phần

- Controller tương tác với qua lại với View.
- Controller tương tác qua lại với Model.
- Model và View không có sự tương tác với nhau trực tiếp mà nó tương tác với nhau thông qua Controller.

Ví dụ cho sự tương tác: Khi người dùng ấn đăng nhập từ view thì request sẽ được gửi từ trình duyệt đến controller, controller sẽ gọi đến model xử lý logic và trả lại kết quả đó cho user thông qua view .

Ưu điểm của MVC:

- Bảo trì code dễ dàng, dễ dàng mở rộng và phát triển.
- Hỗ trợ dễ dàng hơn cho khách hàng mới.
- Việc phát triển các thành phần khác nhau có thể được thực hiện song song.
- Nó giúp bạn tránh sự phức tạp bằng cách chia ứng dụng thành ba đơn vị Model, View và Controller.
- Cung cấp hỗ trợ tốt nhất cho phát triển theo hướng thử nghiệm.
- Nó hoạt động tốt cho các ứng dụng Web được hỗ trợ bởi các nhóm lớn các nhà thiết kế và phát triển web.
- Cung cấp khả năng phân tách rõ ràng các mối quan tâm.
- Thân thiện với Công cụ Tìm kiếm (SEO).
- Tất cả các đối tượng được phân loại và đối tượng độc lập với nhau để bạn có thể kiểm tra chúng một cách riêng biệt.

Nhược điểm của việc sử dụng MVC

- Khó đọc, thay đổi, kiểm tra và sử dụng lại mô hình này.
- Không có hỗ trợ xác thực chính thức.
- Tăng độ phức tạp và tính kém hiệu quả của dữ liệu.
- Khó khăn khi sử dụng MVC với giao diện người dùng hiện đại.
- Cần có nhiều người lập trình để tiến hành lập trình song song.
- Cần có kiến thức về nhiều công nghệ.
- Bảo trì nhiều code trong Controller.

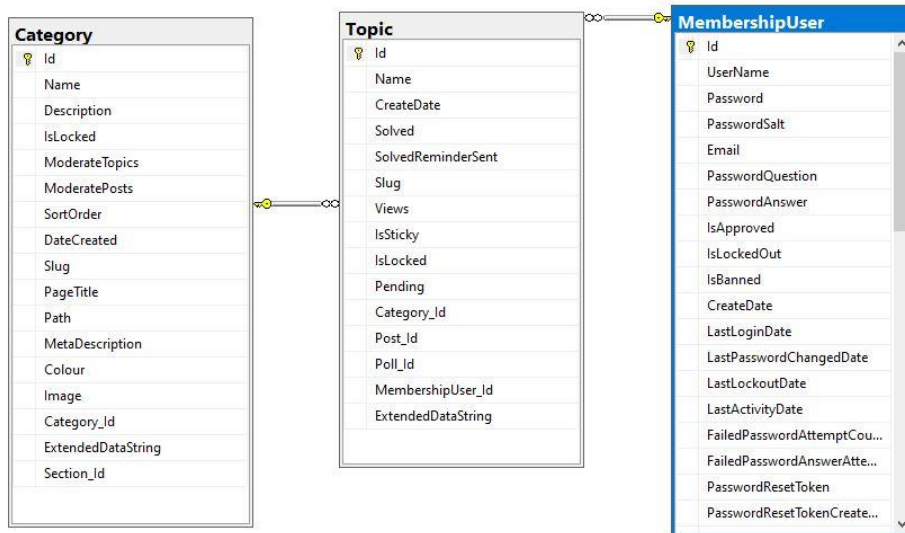
Kết luận

- MVC là một mẫu kiến trúc phân tách ứng dụng thành 1) Model, 2) View và 3) Controller.
- Model: Nó bao gồm tất cả dữ liệu và logic liên quan của nó.
- View: Trình bày dữ liệu cho người dùng hoặc xử lý tương tác của người dùng.
- Controller: Là nơi kết nối giữa các thành phần Model và View.
- Kiến trúc MVC được thảo luận lần đầu vào năm 1979 bởi Trygve Reenskaug.
- MVC là một mô hình dễ dàng mở rộng và bảo trì code.
- Một số Framework MVC phổ biến là Rails, Zend Framework, CodeIgniter, Laravel, Fuel PHP, v.v.

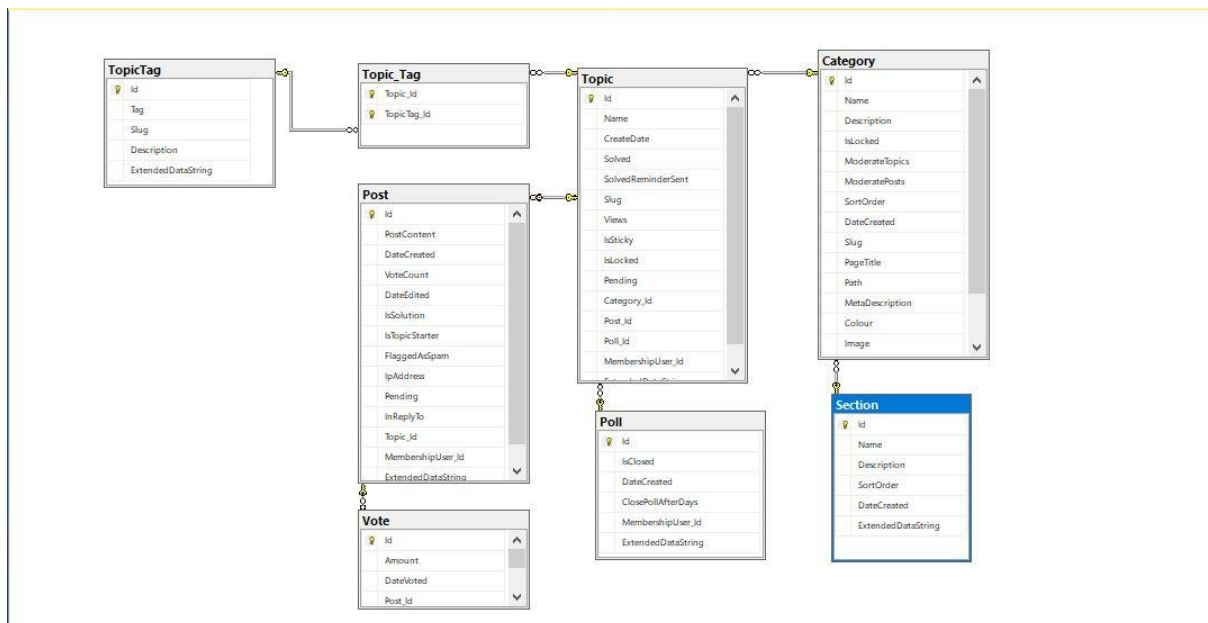
CHƯƠNG 2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG WEBSITE

2.1 Thiết kế cơ sở dữ liệu

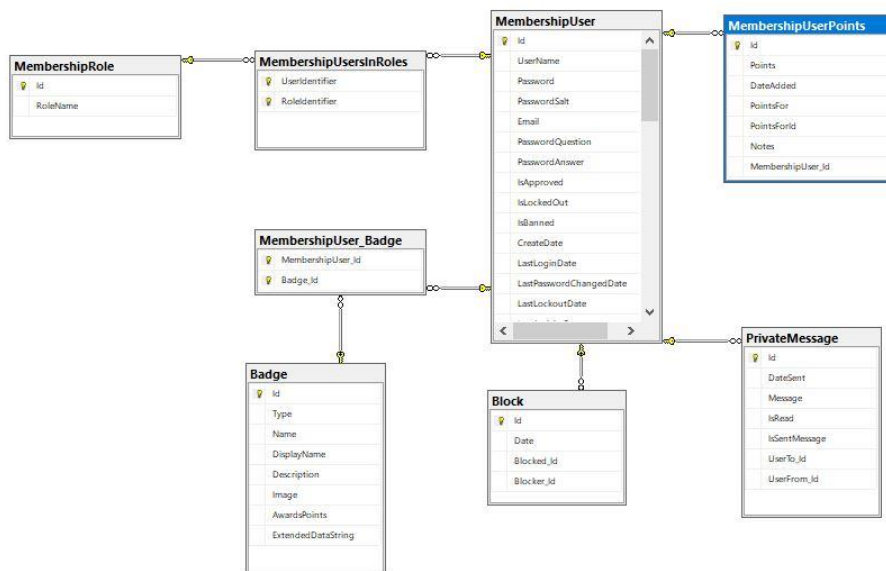
2.1.1 Biểu đồ lớp



Mối quan hệ chung của hệ thống: Người Dùng - Chủ đề - Danh Mục



Mối quan hệ của Chủ đề



Mối quan hệ của User

2.1.2 Tác nhân hệ thống

Tác Nhân	Hành Động
ADMIN	<ul style="list-style-type: none"> • Đăng nhập hệ thống bằng tài khoản Admin • Quản lý danh mục • Quản lý tài khoản người dùng • Phân quyền cho người dùng
MEMBER	<ul style="list-style-type: none"> • Đăng nhập bằng tài khoản người dùng • Xem, trả lời bài đăng • Đánh giá, bình chọn bài đăng • Đăng bài viết, chỉnh sửa bài viết đã đăng

2.1.3 Phân tích cơ sở dữ liệu :

- Bảng [Category] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã chủ đề	Khóa chính
2	Name	nvarchar(450)	Tên	
3	CreateDate	datetime	Ngày tạo	
4	Solved	bit	Đã được giải đáp	
6	Slug	nvarchar(450)	Url	
7	Views	int	Lượt xem	
8	IsLocked	bit	Khóa	
9	Category_Id	uniqueidentifier	Mã danh mục	
10	Post_Id	uniqueidentifier	Mã bài đăng	
11	Poll_Id	uniqueidentifier	Mã bình chọn	
12	MembershipUser_Id	uniqueidentifier	Mã User	

- Bảng [Category] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã danh mục	Khóa chính
2	Name	nvarchar(450)	Tên	
3	Description	nvarchar(MAX)	Chú thích	
4	DateCreated	datetime	Ngày tạo	

- Bảng [Section] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã lĩnh vực	Khóa chính
2	Name	nvarchar(450)	Tên lĩnh vực	
3	Description	nvarchar(MAX)	Chú thích	
4	DateCreated	datetime	Ngày tạo	

- Bảng [Poll] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã bình chọn	Khóa chính
2	IsClosed	bit	Đã đóng	
3	DateCreated	datetime	Ngày tạo	
4	ClosePollAfterDays	int	Đóng bình chọn sau n ngày	
5	MembershipUser_Id	uniqueidentifier	Mã User	

- Bảng [TopicTag] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã thẻ	Khóa chính
2	Tag	nvarchar(100)	Tên thẻ	
3	Slug	nvarchar(100)	Slug	
4	Description	nvarchar(MAX)	Chú thích	

- Bảng [Post] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã bài đăng	Khóa chính
2	PostContent	nvarchar(MAX)	Nội dung	
3	DateCreated	datetime	Ngày tạo	
4	VoteCount	int	Số lượng vote	
5	DateEdited	datetime	Ngày chỉnh sửa	
6	IsSolution	bit	Là giải pháp	
7	IsTopicStarter	bit	Là của người đăng chủ đề	
8	InReplyTo	uniqueidentifier	Là câu trả lời của câu trả lời khác	
9	Topic_Id	uniqueidentifier	Mã chủ đề	
10	MembershipUser_Id	uniqueidentifier	Mã User	

- Bảng [Vote] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã vote	Khóa chính
2	Amount	int	Số lượng vote	
3	DateVoted	datetime	Ngày vote	

4	Post_Id	uniqueidentifier	Mã bài đăng	
5	MembershipUser_Id	uniqueidentifier	Mã User	
6	VotedByMembershipUser_Id	uniqueidentifier	Mã User đã vote	

- Bảng [MembershipUser] :

STT	Tên trường	Kiểu dữ liệu	Diễn giải	Chú thích
1	id	uniqueidentifier	Mã User	Khóa chính
2	UserName	nvarchar(150)	Tên người dùng	
3	Password	nvarchar(128)	Mật khẩu	
4	Email	nvarchar(128)	Email	
5	IsBanned	bit	Bị cấm	
6	CreateDate	datetime	Ngày tạo	
7	LastLoginDate	datetime	Ngày đăng nhập gần nhất	
8	LastActivityDate	datetime	Ngày hoạt động gần nhất	

- Bảng [MembershipUserPoint]

STT	Tên trường	Kiểu giữ liệu	Diễn giải	Chú Thích
1	id	uniqueidentifier	id	Khóa chính
2	Points	int	Điểm	
3	DateAdded	datetime	Ngày thêm	
4	Notes	nvarchar(400)	Ghi chú	
5	MembershipUser_Id	uniqueidentifier	Mã user	

- Bảng [PrivateMessage]

STT	Tên trường	Kiểu giữ liệu	Diễn giải	Chú Thích
1	id	uniqueidentifier	Id	Khóa chính
2	DateSent	datetime	Ngày gửi	
3	Message	nvarchar (MAX)	Nội dung	

4	IsRead	bit	Đã đọc	
5	IsSendMessage	bit	Đã gửi	
6	UserTo_Id	uniqueidentifier	Id người nhận	
7	UserFrom_Id	uniqueidentifier	Id người gửi	

- Bảng [Block]

STT	Tên trường	Kiểu giữ liệu	Diễn giải	Chú Thích
1	id	uniqueidentifier	Id	Khóa chính
2	Date	datetime	Ngày	
3	Blocked_Id	uniqueidentifier	Id người bị chặn	
4	Blocker_Id	uniqueidentifier	Id người chặn	

- Bảng [Badge]

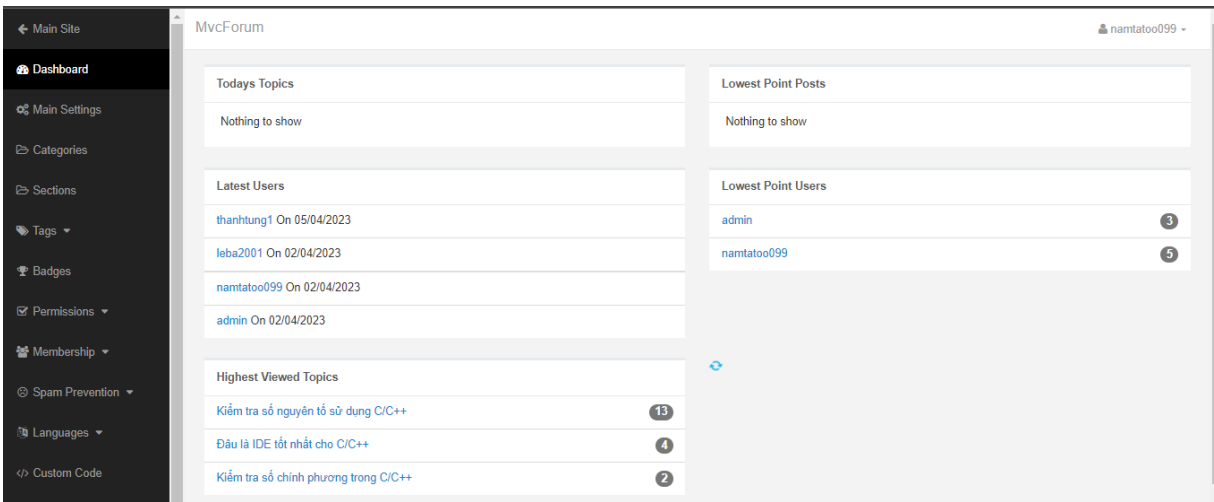
STT	Tên trường	Kiểu giữ liệu	Diễn giải	Chú Thích
1	id	uniqueidentifier	Mã huy hiệu	Khóa chính
2	Type	nvarchar(50)	Loại huy hiệu	
3	Name	nvarchar(50)	Tên huy hiệu	
4	DisplayName	nvarchar(50)	Tên hiển thị	
5	Description	nvarchar(MAX)	Chú thích	
6	Image	nvarchar(50)	Ảnh	
7	AwardsPoints	int	Số điểm	

- Bảng [MembershipRole]

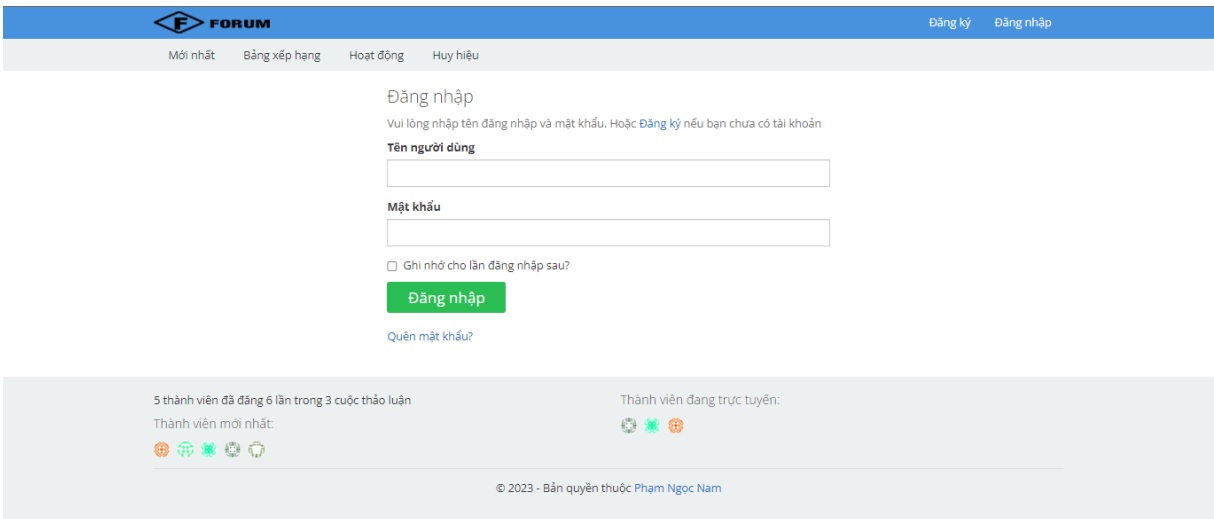
STT	Tên trường	Kiểu giữ liệu	Diễn giải	Chú Thích
1	id	uniqueidentifier	Mã phân quyền	
2	RoleName	nvarchar(256)	Tên phân quyền	

2.2. Thiết kế hệ thống.

2.2.1. Trang quản trị.



Hình 1. Giao diện trang quản trị ADMIN



Hình 2. Màn hình đăng nhập

FORUM

Đăng ký

Đăng nhập

Mới nhất

Bảng xếp hạng

Hoạt động

Hủy hiệu

Đăng ký

Tên người dùng

Mật khẩu

Xác nhận mật khẩu

Địa chỉ email (Ảnh được sử dụng làm ảnh đại diện của bạn)

Đăng ký

5 thành viên đã đăng 6 lần trong 3 cuộc thảo luận

Thành viên mới nhất:

Thành viên đang trực tuyến:

© 2023 - Bản quyền thuộc Phạm Ngọc Nam

Hình 3. Màn hình đăng ký

FORUM

Cài đặt

Mới nhất

Đang theo dõi

Bài đăng

Yêu thích

Bảng xếp hạng

Hoạt động

Hủy hiệu

Danh mục bạn đang theo dõi

Bạn không theo dõi danh mục nào

Chủ đề bạn đang theo dõi

Danh mục: Kiến thức lập trình

Kiểm tra số chính phương trong C/C++

Bỏ theo dõi

Đăng bởi namtatoo099 18 phút trước

Danh mục: Kiến thức lập trình

Kiểm tra số nguyên tố sử dụng C/C++

Bỏ theo dõi

Đăng bởi namtatoo099 23 giờ trước

1

12

Cuộc thảo luận mới

THAY ĐỔI NGÔN NGỮ

Vietnamese (Vietnam)

DANH MỤC

Tất cả danh mục

Công cụ lập trình

Kiến thức lập trình

TÌM KIẾM

Nội dung tìm kiếm...

Tiếp tục

THỂ PHỔ BIẾN

HOCĐÁP: 2

C++: 1

BẢNG XẾP HẠNG TUẦN NÀY

namtatoo099

5

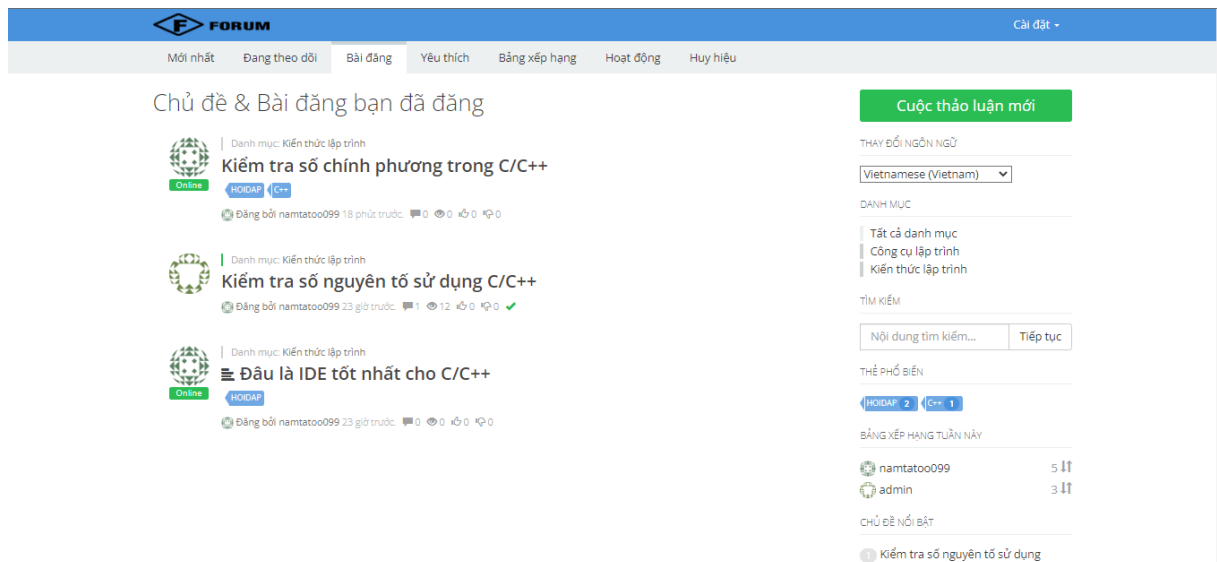
admin

3

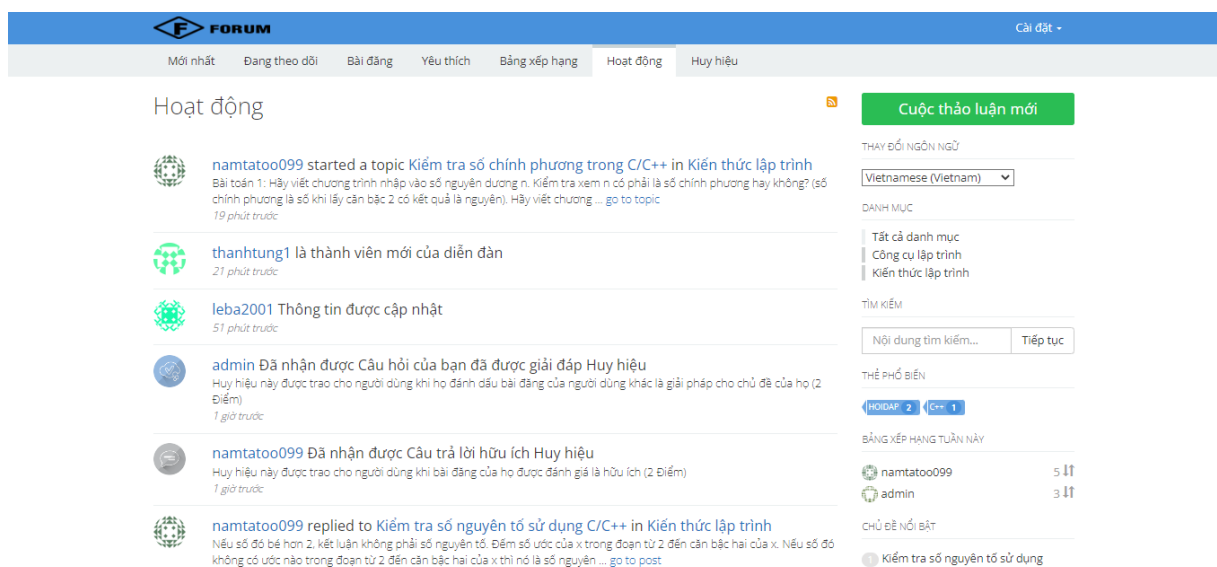
CHỦ ĐỀ NỔI BẬT

Kiểm tra số nguyên tố sử dụng

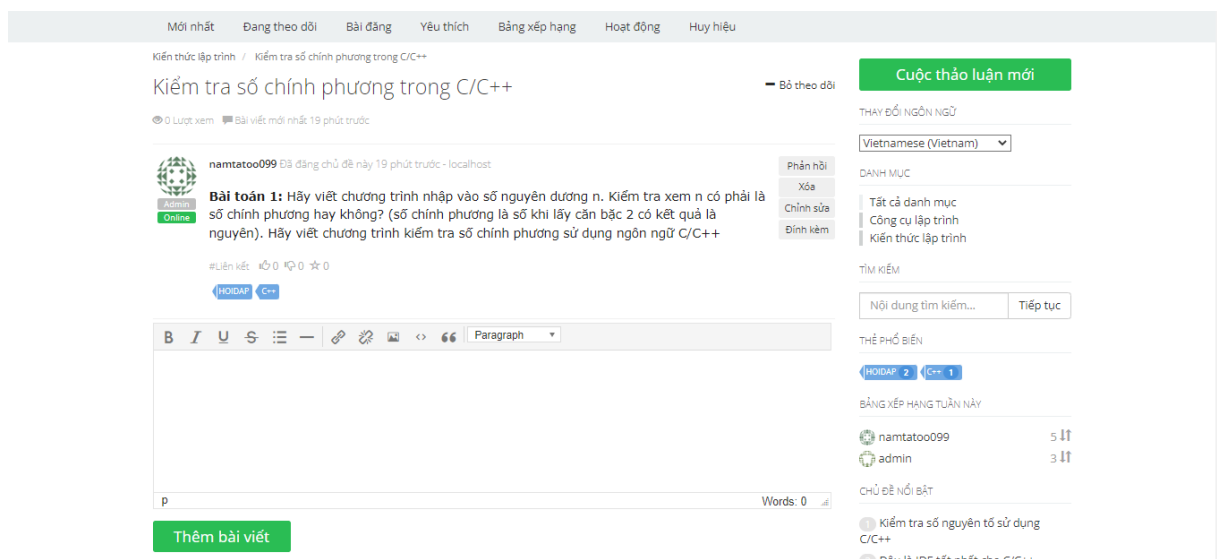
Hình 4. Giao đang theo dõi



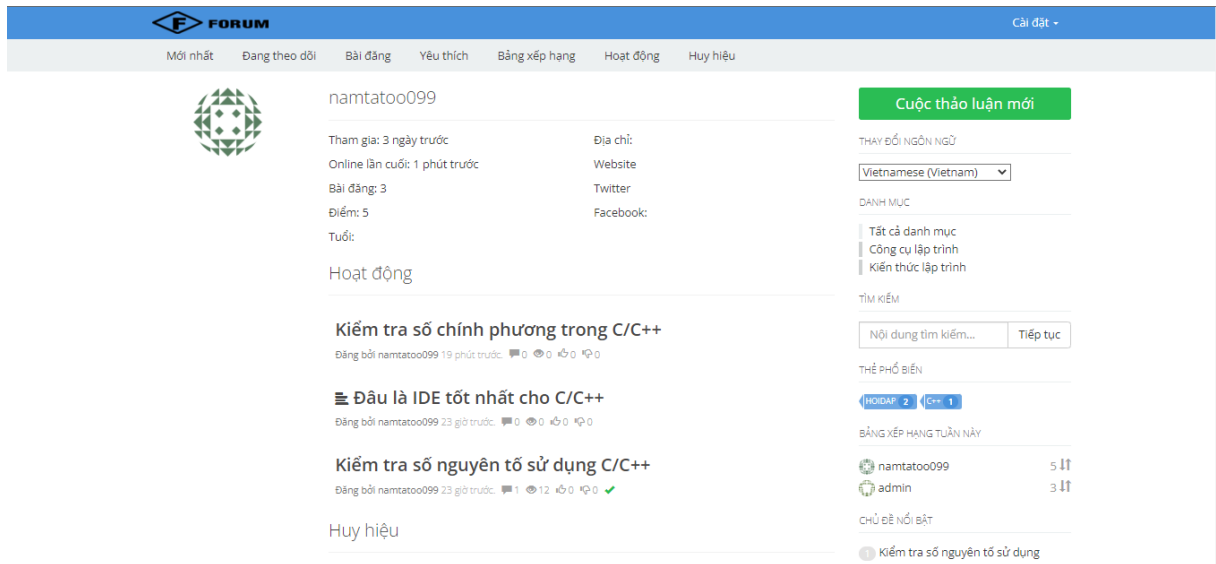
Hình 5. Giao diện bài đăng đã đăng



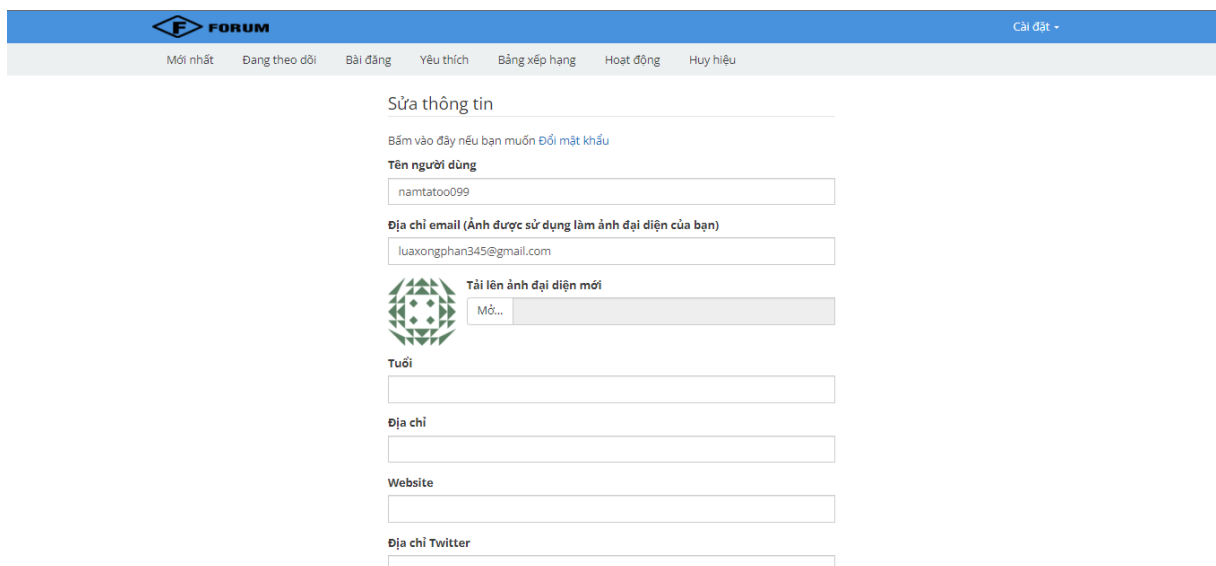
Hình 6. Giao diện hoạt động



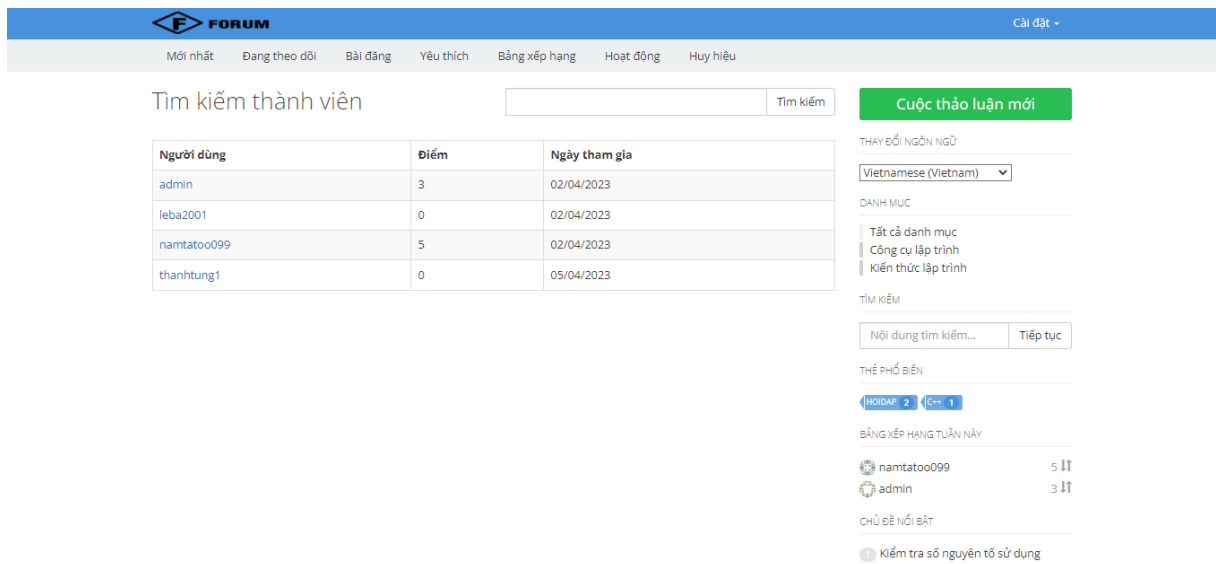
Hình 7. Giao diện chi tiết chủ đề và trả lời



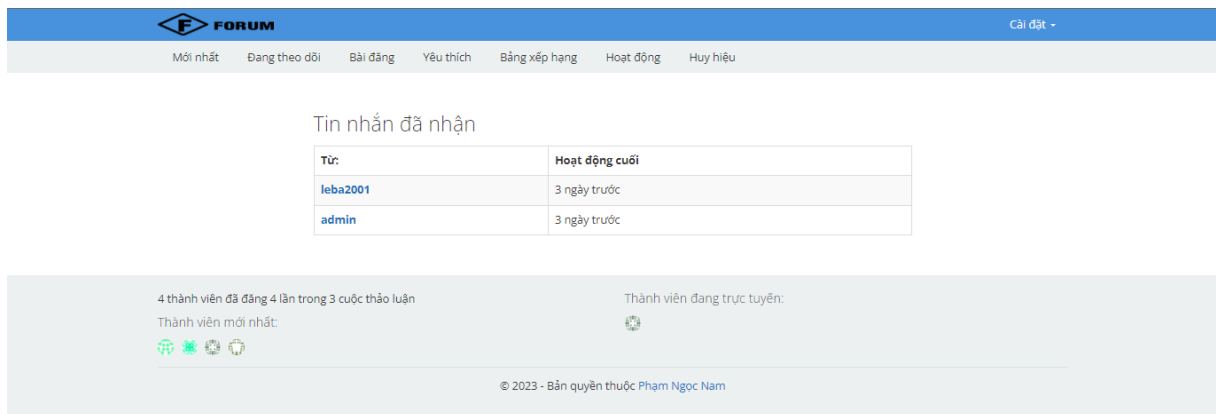
Hình 8. Giao diện tổng quan người dùng



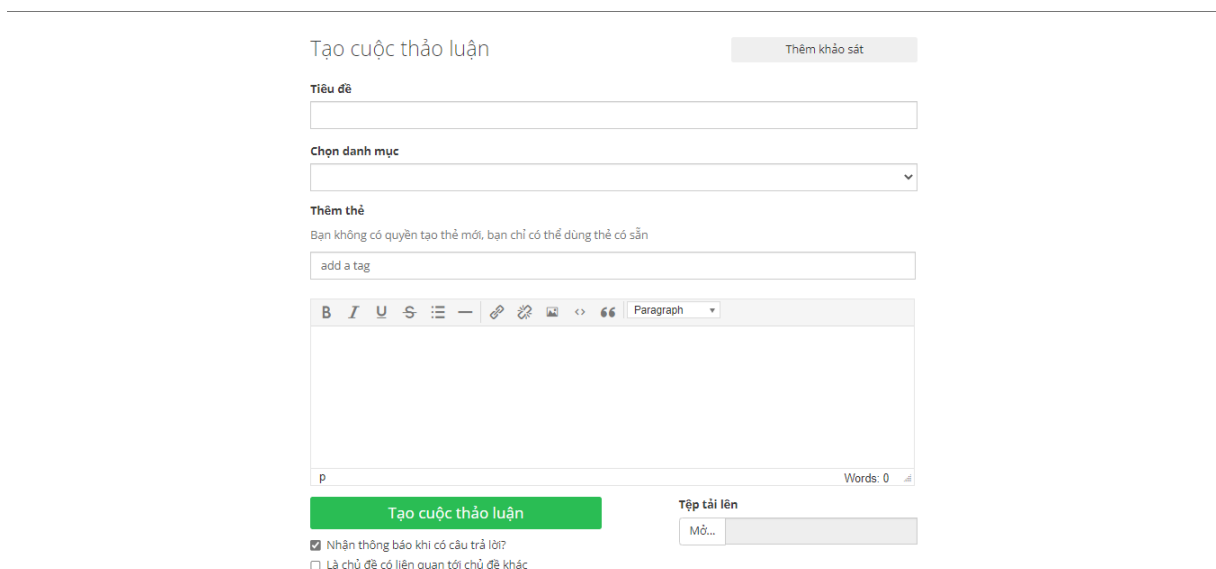
Hình 9. Giao diện thông tin người dùng



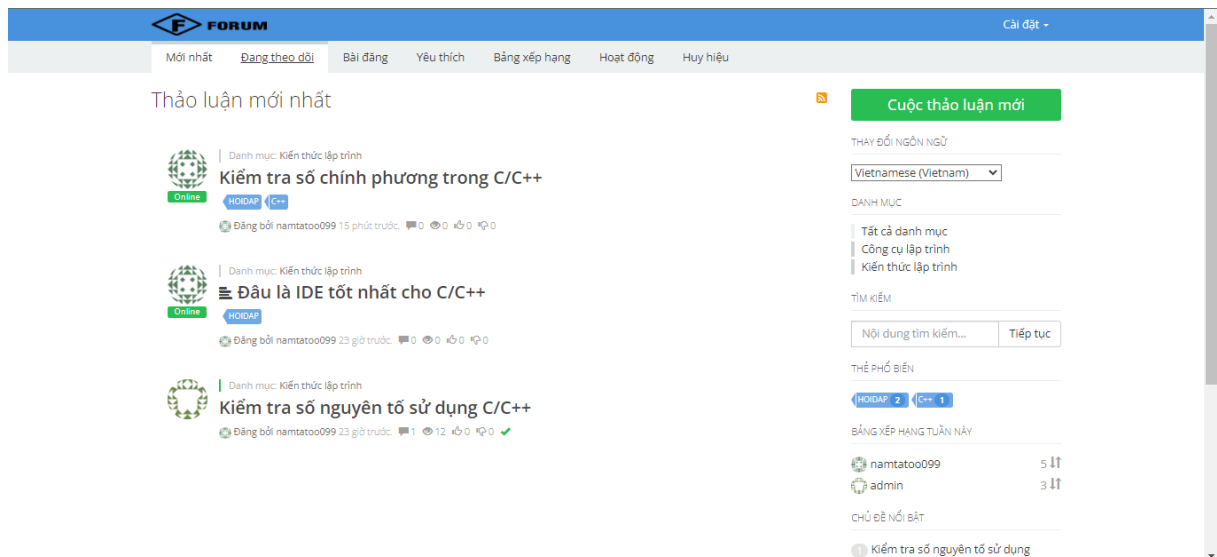
Hình 10. Giao diện tìm kiếm thành viên



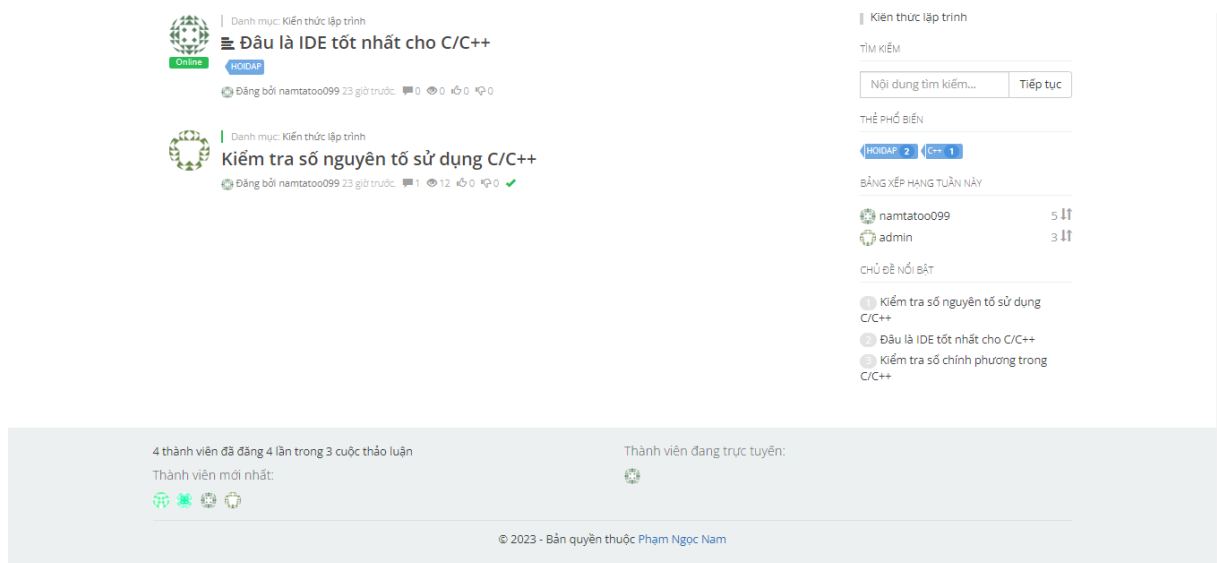
Hình 11. Giao diện tin nhắn đã nhận



Hình 12. Giao diện thêm mới chủ đề



Hình 13. Phần header trang chủ



Hình 17b. Phần footer trang chủ

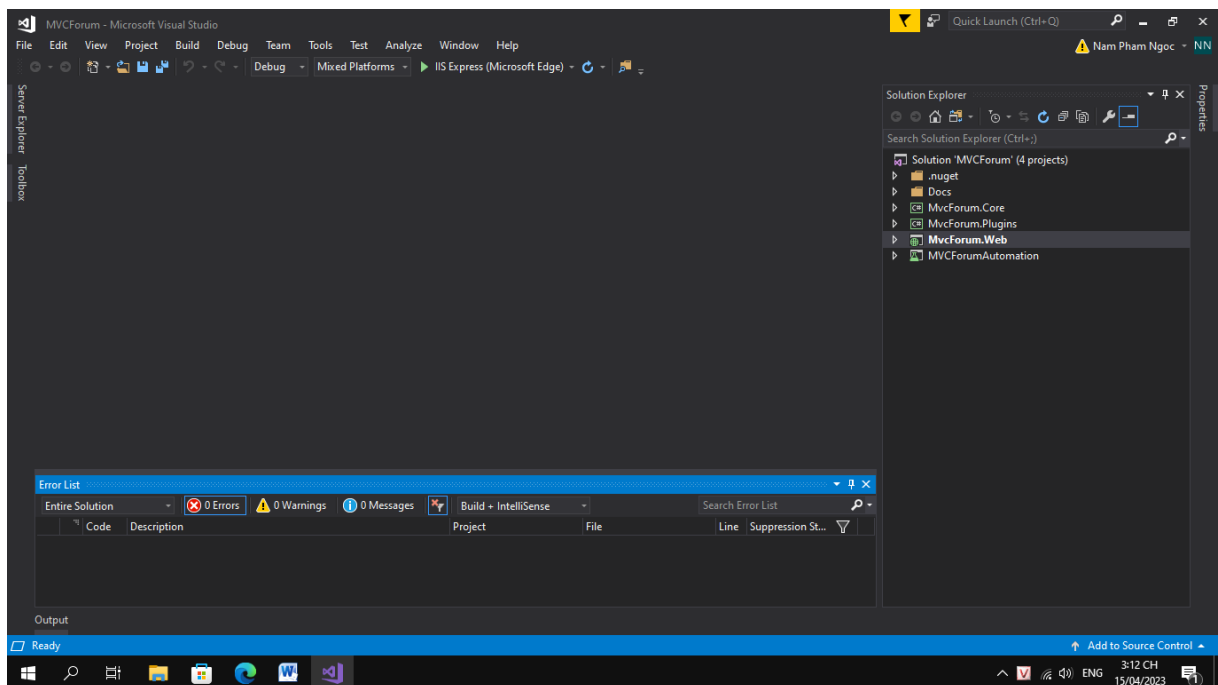
CHƯƠNG 3: THIẾT KẾ, CÀI ĐẶT WEBSITE

3.1 Cài đặt các công cụ hỗ trợ.

3.1.1 Visual Studio.

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

Visual Studio là hệ thống tập hợp tất cả những gì liên quan tới phát triển ứng dụng, bao gồm trình chỉnh sửa mã, trình thiết kế, gỡ lỗi. Tức là, bạn có thể viết code, sửa lỗi, chỉnh sửa thiết kế ứng dụng dễ dàng chỉ với 1 phần mềm Visual Studio mà thôi. Không dừng lại ở đó, người dùng còn có thể thiết kế giao diện, trải nghiệm trong Visual Studio như khi phát triển ứng dụng Xamarin, UWP bằng XAML hay Blend vậy.



3.1.2 SQL SERVER

SQL server hay còn được gọi là Microsoft SQL Server, nó từ viết tắt của MS SQL Server. Đây chính là một loại phần mềm đã được phát triển bởi Microsoft và nó được sử dụng để có thể dễ dàng lưu trữ cho những dữ liệu dựa theo tiêu chuẩn RDBMS.

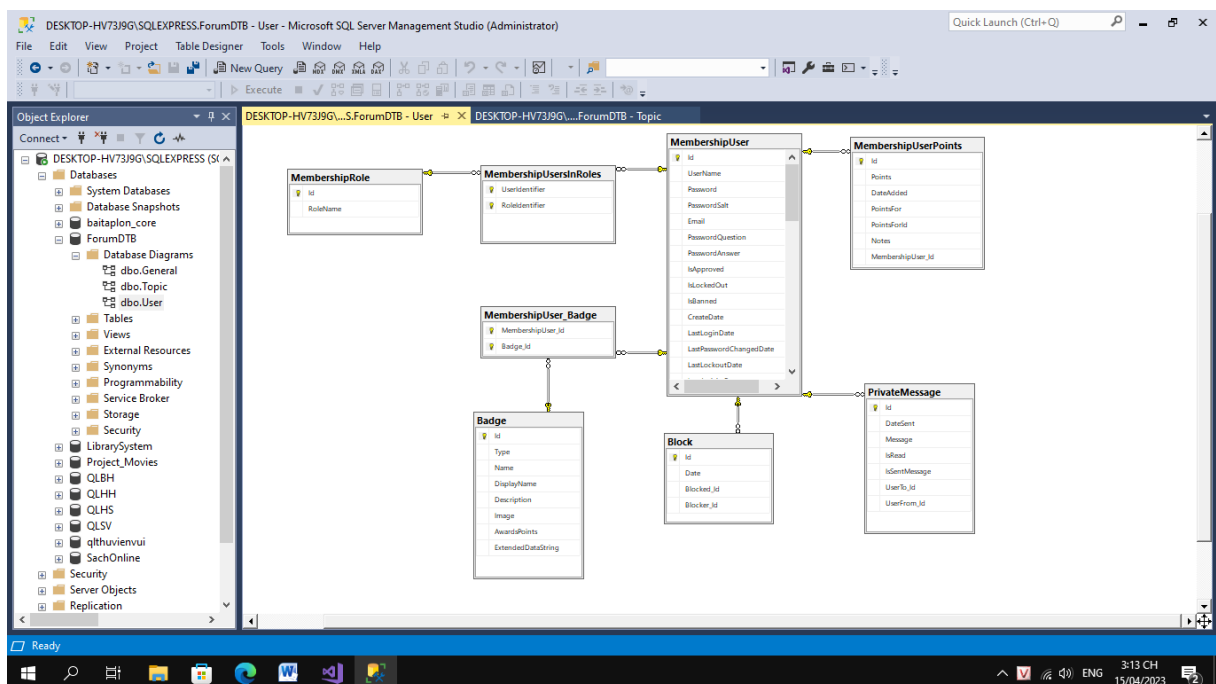
Từ đó, người ta sẽ lưu trữ dữ liệu dựa vào tiêu chuẩn RDBMS và nó cũng là một trong những hệ quản trị cơ sở dữ liệu dạng quan hệ đối tượng.

SQL Server có khả năng cung cấp đầy đủ các công cụ cho việc quản lý từ giao diện GUI đến sử dụng ngôn ngữ cho việc truy vấn SQL. Điểm mạnh của SQL điểm mạnh của nó là có nhiều nền tảng được kết hợp cùng như: ASP.NET, C# để xây dựng Winform cũng chính nó có khả năng hoạt động độc lập. Tuy nhiên, SQL Server thường đi kèm với việc thực hiện riêng các ngôn ngữ SQL, T-SQL,...

Cụ thể như sau:

- T-SQL là một trong những loại ngôn ngữ thuộc quyền sở hữu của Microsoft và được gọi với cái tên Transact-SQL. Nó thường cung cấp thêm rất nhiều cho các khả năng khai báo biến, thủ tục lưu trữ và xử lý ngoại lệ,...
- SQL Server Management Studio là một loại công cụ giao diện chính cho máy chủ cơ sở của chính dữ liệu SQL, thông thường thì nó hỗ trợ cho cả môi trường 64 bit và 32 bit.

SQL Server 2019 Bản 2019 cho phép người được sử dụng để tham gia vào các thùng chứa SQL Server, HDFS và Spark cùng nhau bằng nhiều tính năng. Ngoài ra, nó còn giới thiệu cho việc xây dựng chỉ mục cột, xây dựng lại cũng như che giấu đi dữ liệu tĩnh. Từ đó, phục hồi dữ liệu tăng tốc mới và thực hiện, hoàn tác các giai đoạn làm lại số thứ tự nhật ký trang.



KẾT LUẬN

Kết quả đạt được:

- Củng cố thêm các kiến thức về phân tích thiết kế hệ thống, HTML, CSS, JS, Bootstrap, C#.
- Xây dựng được Website diễn đàn trao đổi thông tin.
- Website cung cấp cho người dùng một trang giao diện dễ sử dụng và dễ quản lý được các mục.
- Website chạy được trên tất cả trên các trình duyệt web như IE, Google Chrome, Mozilla Firefox.
- Qua việc tự tìm hiểu và xây dựng website, cùng với sự hướng dẫn chỉ bảo của thầy cô giáo hướng dẫn. Em đã biết thêm được nhiều kiến thức mới hơn như cách quản lý 1 website, cách triển khai dự án web. Hiểu được quy trình nghiệp vụ khi làm một website.

Những gì chưa đạt được:

- Một số chức năng vẫn còn chưa hoàn thiện và chưa đúng với mong muốn của đã đặt ra. Cách thiết kế giao diện, màu sắc vẫn chưa được chuyên nghiệp. Một số chỗ vẫn còn chưa đúng và dư thừa ...
- Do lần đầu làm nên một số chức năng của hệ thống vẫn còn lỗi và một số khác vẫn chưa thực hiện được.

Hướng phát triển:

- Khắc phục những gì chưa đạt được như: cải thiện giao diện, tối ưu lại hệ thống...
- Tìm hiểu thêm các công nghệ mới hiện nay để áp dụng vào hệ thống.
- Triển khai phần mềm trên domain

Các loại tài liệu tham khảo:

- <https://www.w3schools.com/>
- <https://stackoverflow.com/>