**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**~~~~~~\*~~~~~~**

A picture containing text, sign

Description automatically generated

**BÁO CÁO CUỐI KỲ**

**QUẢN LÝ LỚP HỌC**

**HỌC PHẦN: 2111COMP106401**

**CÔNG NGHỆ NET**

**Thành Phố Hồ Chí Minh – ngày 16 tháng 12 năm 2021**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**~~~~~~\*~~~~~~**

**Đề tài:**

**QUẢN LÝ LỚP HỌC**

**HỌC PHẦN: 2111COMP106001**

**CÔNG NGHỆ NET**

**Người hướng dẫn:** Thầy Lương Trần Hy Hiến

**Nhóm thực hiện:** Aktsuki

**Sinh viên thực hiện:**

Huỳnh Mẫn Đạt - 4501104047

Nguyễn Danh Trung - 4501104260

Nguyễn Thị Niệm – 4501104168

Phan Thị Thu Hà – 4501104064

**Thành Phố Hồ Chí Minh – ngày 16 tháng 12 năm 2021**

#### Phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| **Họ và tên** | **MSSV** | **Công việc** | **Mức độ hoàn thành** |
| Huỳnh Mẫn Đạt | 4501104047 |  |  |
| Nguyễn Danh Trung | 4501104260 |  |  |
| Nguyễn Thị Niệm | 4501104168 |  |  |
| Phan Thị Thu Hà | 4501104064 |  |  |

#### Mục lục

[Phân công công việc 2](#_Toc91546082)

[Mục lục 3](#_Toc91546083)

[Lời nói đầu 5](#_Toc91546084)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 6](#_Toc91546085)

[1. Giới Thiệu Đề Tài 6](#_Toc91546086)

[2. Mục Đích Nghiên Cứu 6](#_Toc91546087)

[3. Mục Tiêu Đề Tài 7](#_Toc91546088)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 8](#_Toc91546089)

[1. Tổng quan về ASP.NET 8](#_Toc91546090)

[2. Mô hình lập trình web MVC của ASP.NET 8](#_Toc91546091)

[2.1 Khái niệm mô hình MVC 8](#_Toc91546092)

[2.2 Lựa chọn và áp dụng mô hình trong xây dựng ứng dụng 11](#_Toc91546093)

[2.2.1 Lợi ích của web ứng dụng dựa trên MVC mô hình. 11](#_Toc91546094)

[2.2.2 Lợi ích của ứng dụng được xây dựng trên Web Forms nền tảng 11](#_Toc91546095)

[2.3 Các tính năng của nền tảng ASP.NET MVC 11](#_Toc91546096)

[2.4 Giao tiếp trong kiến trúc ASP.NET MVC 12](#_Toc91546097)

[3. Mô hình lập trình web MVC của ASP.NET5 14](#_Toc91546098)

[3.1 Tính năng của ASP.Net MVC 5 15](#_Toc91546099)

[4. Hoạt động của các thành phần chính trong ASP.NET MVC 16](#_Toc91546100)

[5. Ưu điểm và khuyết điểm của ASP.NET 17](#_Toc91546101)

[6. Entity Framework 19](#_Toc91546102)

[6.1 Entity framework là gì? 19](#_Toc91546103)

[6.2 Tại sao lại sử dụng Entity Framework 19](#_Toc91546104)

[6.3 Cấu trúc Entity Framework 20](#_Toc91546105)

[6.4 Các tình huống dùng Entity Framework 21](#_Toc91546106)

[7. Bootstrap 23](#_Toc91546107)

[7.1 Bootstrap là gì? 23](#_Toc91546108)

[7.2.1 file chính của Bootstrap: 24](#_Toc91546109)

[7.2 Tại sao phải dùng bootstrap 25](#_Toc91546110)

[8. Jquery 27](#_Toc91546111)

[8.1 Jquery là gì 27](#_Toc91546112)

[8.2 Jquery giúp gì cho người lập trình 28](#_Toc91546113)

[8.3 Tại sao nên dùng Jquery 29](#_Toc91546114)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 30](#_Toc91546115)

[1. Phân tích và thiết kế hệ thống 30](#_Toc91546116)

[1.1 Yêu cầu cơ bản của hệ thống 30](#_Toc91546117)

[1.2 Mô tả chức năng ứng dụng 30](#_Toc91546118)

[2. Phân tích hệ thống 30](#_Toc91546119)

[2.1 Sơ đồ cơ sở dữ liệu 30](#_Toc91546120)

[2.2 Lược đồ phân cấp chức năng 30](#_Toc91546121)

[3. Công nghệ áp dụng 30](#_Toc91546122)

[CHƯƠNG 4: GIAO DIỆN DEMO 31](#_Toc91546123)

[KẾT LUẬN 32](#_Toc91546124)

[TÀI LIỆU THAM KHẢO 33](#_Toc91546125)

#### Lời nói đầu

Đầu tiên, chúng em xin được bày tỏ lòng biết ơn chân thành đến thầy Lương Tần Hy Hiến. Thầy là người trực tiếp hướng dẫn, tận tình chỉ bảo và tạo điều kiện thuận lợi cho chúng em thực hiện nghiên cứu đề tài. Nhờ đó, chúng em có kiến thức nền và định hướng đúng đắn trong quá trình nghiên cứu. Tiếp đến, em xin được giới thiệu qua về đề tài.

Cùng với sự phát triển nhanh chóng của Công nghệ thông tin con người đã nghĩ ra rất nhiều các ứng dụng phục vụ cho nhu cầu và lợi ích con người. Trong đó xây dựng phần mềm phục vụ công việc quản lý là một trong những vấn đề mà các nhà lập tình viên luôn quan tâm hàng đầu. Phần mền Quản lý lớp học cũng là một trong những ứng dụng mà con người quan tâm nhiều nhất. Nó phục vụ cho công tác nghiệp vụ của trường học làm giảm thiểu tối đa những vất vả trong công việc quản lý học sinh

Với mong muốn tìm hiểu một phần nhỏ trong kho tàng phát triển của công nghệ thông tin, nhóm em đã đi vào tìm hiểu về cách thức tổ chức và hoạt động của lớp học để từ đó xây dựng lên phần mềm “Quản lý lớp học” nhằm phục vụ lợi ích của lớp , trường học.

Trong quá trình thực hiện đề tài gặp khó khăn do dịch Covid 19 bùng phát, nhóm em đã cố gắng tham khảo và chọn lọc khá kỹ về nội dung nhưng không tránh khỏi những thiếu sót. Ứng dụng phần mềm có thể chưa hoàn chỉnh, nhưng một phần nào đã thể hiện được sự cố gắng, hiểu biết của nhóm trong những bước đầu hoà nhập với các môi trường khác nhau và thế giới công nghệ mới.

Thành phố Hồ Chí Minh, ngày 16 tháng 12 năm 2021

# TỔNG QUAN VỀ ĐỀ TÀI

## Giới Thiệu Đề Tài

Trong những năm gân đây, ứng dụng Công nghệ thông tin về quản lý Web phát triển mạnh mẽ, giúp cho các trường học dễ dàng quản lý học sinh hơn nhờ vào các trang web quản lý lớp học. Nên nhóm chúng em quyết định chọn đề tài xây dựng trang website quản lý lớp học. Với mong muốn giúp cho thầy cô kiểm soát tốt hơn các thông tin cá nhân, điểm của học sinh. Ngoài ra website còn rất thân thiện, giao diện đẹp, bố cục hợp lý giúp cho thầy cô thuận tiện hơn trong việc quản lý. Đây là một điều tuyệt vời nhất đối với giáo viên.

Một trong các nhà phát triển phần mềm hàng đầu thế giới về lĩnh vực này là Microsoft, họ đã cho ra đời mô hình ASP.NET MVC5 hỗ trợ người phát triển các ứng dụng web trên nền .NET Framework trong đó có tích hợp công nghệ LINQ với Entity Framework hỗ trợ làm việc với các loại Cơ sở dữ liệu.

Do đó, nhóm chúng em chọn đề tài “Quản lý lớp học” để thực hiện đồ án lập trình web đáp ứng theo chương trình đào tạo của Trường, đồng thời mong muốn góp phần đẩy đưa ứng dụng Công nghệ thông tin vào Website để cho việc Quản lý lớp học được hoàn thiện và thân thiện nhất, giúp cho người dung hài long khi vào website Quản lý lớp học của nhóm em.

## Mục Đích Nghiên Cứu

1. Học được nhiều kiến thức mới
2. Tìm hiểu kĩ thuật lập trình trên ASP.NET MVC
3. Tìm hiểu về ngôn ngữ và kiến thức tương tác CSDL (LINQ to SQL và Entity Framework)
4. Thiết kế trang web với giao diện đẹp, bố cục rõ ràng, thân thiện với người dùng
5. Nắm được các kỹ năng và phương pháp lập trình hướng đối tượng trong .Net
6. Tìm hiểu về các công nghệ hỗ trợ như: HTML, CSS, Jquery, Web Sevice , bookstrap, charts, calendar, …
7. Cách điều khiển dữ liệu (Controller).

## Mục Tiêu Đề Tài

Vì là một website quản lý lớp học, nên điều nhóm chúng em chứ trọng đó là tính thân thiện với người dùng, giao diện đẹp mắt, cách trình bày, bố trí các thông tin một cách hợp lý nhất. Người dùng sẽ cảm thấy thoải mái khi sử dụng một trang web đẹp và dễ truy cập, sử dụng.

Người dùng (User)

Người quản lý (Admin)

# CƠ SỞ LÝ THUYẾT

## Tổng quan về ASP.NET

Đầu năm 2002, Microsoft giới thiệu một kỹ thuật lập trình khá mới mẻ với tên gọi ban đầu là ASP+, tên chính thưcsau này là một ASP.Net. Với ASP.Net, không những không cần đòi hỏi bạn phải biết các nhãn HTML, thiết kế Web, mà nó còn hỗ trợ mạnh trong lập trình hướng đối tượng, trong quá trình xây dựng và phát triển ứng dụng. Mạng lưới là kỹ thuật lập trình và phát triển ứng dụng web phía Máy chủ dựa trên nền tảng Microsoft, mạng lưới Framework. Hầu hết, những người mới đến với lập trình web đều bắt đầu tìm hiểu kỹ thuật ở phía Client (Client-side) như: HTML, Java Script , CSS, …

Khi Web browser yêu cầu một trang web (trang web sử dụng kĩ thuật client-side), Web server tìm trang web mà Client yêu cầu, sau đó gởi về cho Client. Client nhận kết quả trả về từ server và hiển thị ra màn hình. ASP.Net sử dụng kỹ thuật ở phía lập trình server thì hoàn toàn khác, mã lệnh phía server (ví dụ: mã lệnh trong ASP) sẽ được biên dịch và thi hành tại Web Server. Sau khi được Server đọc, biên dịch và thi hành, kết quả tự động được chuyển sang HTML/JavaScript/CSS và trả về cho Client. Tất cả các xử lý lệnh ASP.Net đều được thực hiện tại Server và do đó, gọi là kỹ thuật lập trình ở phía Server. ASP.NET được Microsoft phát triển qua nhiều phiên bản từ ASP.Net 1.0, 1.1, 2.0 và gần đây nhất là phiên bản ASP.NET 5.

## Mô hình lập trình web MVC của ASP.NET

Hiện nay đa số framework về web (PHP, Java…) điều sử dụng mô hình MVC như một mẫu thiết kế chuẩn (Design Pattern). Tùy theo đặc điểm của ứng dụng mà bạn chọn cấu trúc cho trang web của mình theo ASP.NET Web Forms hay theo kiến trúc MVC, đôi khi ta có thể kết hợp cả hai. Nó giúp cho tổ chưucs ứng dụng (phân bố source code ứng dụng) thành 3 phần khác nhau Model, View và Cotroller.

### Khái niệm mô hình MVC

MVC là viết tắt của 3 từ M - Model, V - View, C - Controller. Mô hình MVC là một kiến trúc phần mềm hay mô hình thiết kế phần mềm giúp chúng ta tách ứng dụng của chúng ta thành những phần độc lập có những nhiệm vụ riêng biệt. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác

* Model (tầng business – business layer)

Là các thành phần chứa tất cả nghiệp vụ logic, phương thức xử lý, truy xuất CSDL, đối tượng mô tả dữ liệu như class, hàm xử lý … Model được giao nhiệm vụ cung cấp dữ liệu cho CSDL và lưu trữ dữ liệu vào các kho chứa dữ liệu. Tất cả nghiệp vụ logic được thực thi bởi Model. Dữ liệu vào từ người dung sẽ thông qua View để kiểm tra ở Model trước khi lưu vào CSDL. Việc truy xuất, xác nhận và lưu trữ dữ liệu là một phần của Model.

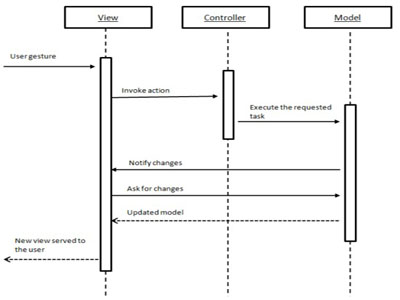
* View (tầng hiển thị – display layer):

View hiển thị các thông tin cho người dùng của ứng dụng và được giao nhiệm vụ cho việc nhận các dữ liệu vào từ người dung, gởi đi các yêu cầu người dung đến bộ điều khiển Controller, 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 (điều khiển đầu vào – input control):

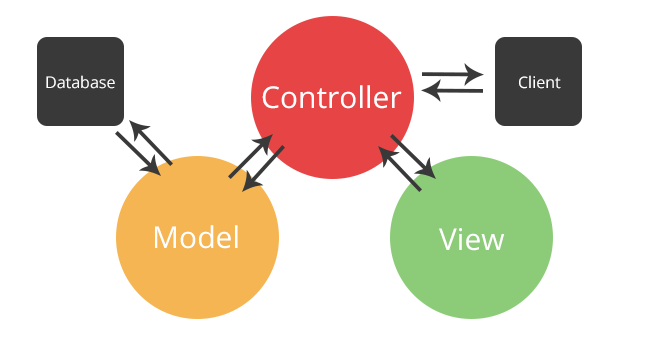
Controller là tầng trung gian giữa Model và View. Controller được giao nhiệm vụ nhận các yêu cầu từ người dùng (phía máy khách). Một yêu cầu được nhận từ máy khách được thể hiện bởi một chức năng logic thích hợp từ thành phần Model và sau đó sinh ra các kết quả cho người dùng và được thành phần View hieenr thị. ActionServlet, Action, ActionForm, struts-config.xml là các thành phần của Controller.

Một ứng dụng web được phát triển theo mô hình MVC, có nguyên lý hoạt động xử lý yêu cầu của người dùng như sau:



Đây là một cách đơn giản để mô tả lại luồng sự kiện được xử lý trong MVC:

* User tương tác với View, bằng cách click vào button, user gửi yêu cầu đi.
* Controller nhận và điều hướng chúng đến đúng phương thức xử lý ở Model.
* Model nhận thông tin và thực thi các yêu cầu.
* Khi Model hoàn tất việc xử lý, View sẽ nhận kết quả từ Model và hiển thị lại cho người dùng.



### Lựa chọn và áp dụng mô hình trong xây dựng ứng dụng

Bạn cần phải xem xét kỹ càng việc áp dụng ASP.NET MVC mô hình hoặc ASP.NET Web Forms mô hình khi xây dựng một ứng dụng. Mô hình MVC là một hướng khác phục vụ cho những mục đích cụ thể nó không thay thế cho Web Forms.

Trước khi quyết định, chúng tôi sẽ phân tích tích cực và điểm khuyết của từng mô hình.

##### Lợi ích của web ứng dụng dựa trên MVC mô hình.

* ASP.NET MVC nền tảng mang lại những lợi ích sau:
* Thích hợp cho các ứng dụng lớn có tạp chất phức tạp bằng cách chia ứng dụng thành mô hình, khung nhìn, bộ điều khiển ba thành phần
* Loại bỏ dạng xem trạng thái hoặc dạng dựa trên máy chủ. Điều này tốt cho sẻ khiến lập trình viên tốn nhiều thời gian hơn nhưng sẻ quản lý tốt ứng dụng của mình.
* Nó sử dụng Front Controller mẫu, mẫu này giúp quản lý các yêu cầu (request) chỉ thông qua một Controller do đó công việc định tuyến.
* Hổ trợ tốt cho việc kiểm tra từng phần.
* Nó hỗ trợ tốt cho các ứng dụng có nhiều thành viên thiết lập và thiết kế mà vẫn quản lý được các tính năng của ứng dụng

##### Lợi ích của ứng dụng được xây dựng trên Web Forms nền tảng

* Dễ dàng cho bạn đi từ nền tảng lập trình Winform với hàng trăm điều khiển hỗ trợ kéo thả và hỗ trợ các máy chủ sự kiện.
* This model is use view state or server-based form, nhờ đó sẽ giúp bạn thực hiện công việc quản lý trạng thái các trang web một cách dễ dàng.
* Phù hợp với các thiết bị nhỏ và thiết kế và lập trình viên là một, những người muốn tận dụng các thành phần giúp xây dựng ứng dụng một cách nhanh chóng.
* Tóm tắt lại khi áp dụng Web Forms.

### **Các tính năng của nền tảng ASP.NET MVC**

* Thừa hưởng tính toán minh bạch và ràng buộc cấu trúc của MVC mô hình
* MVC là nền tảng nền tảng mở rộng (có thể mở rộng) & nhúng (có thể cắm được). Các thành phần của ASP.NET MVC được thiết kế để chúng tôi có thể thay thế một cách dễ dàng hoặc tùy chỉnh dễ dàng.
* ASP.NET MVC có URL cấu trúc tốt cho phép bạn xây dựng các ứng dụng có URL địa chỉ tích cực và dễ tìm kiếm. URL địa chỉ không cần phải có phần mở rộng của tệp tên.
* Hỗ trợ sử dụng các thẻ của các trang ASP.NET(.aspx) Usercontrol (.ascx) và trang master page (.marter).
* Support khá đầy đủ các tính năng của ASP.NET như data caching, seession và profile…
* ASP.NET MVC 3 bổ sung một công cụ xem mới là Razor View Engine cho phép thiết lập các chế độ xem một cách nhanh chóng, dễ dàng.
* ASP.NET MVC 4 hỗ trợ tốt hơn cho ASP.NET Web API bổ sung di động.
* ASP.NET MVC 5 hỗ trợ chứng thực thông qua các API khác (facebook, google, …), Bootstrap được bổ sung vào giao diện thiết kế hổ trợ.

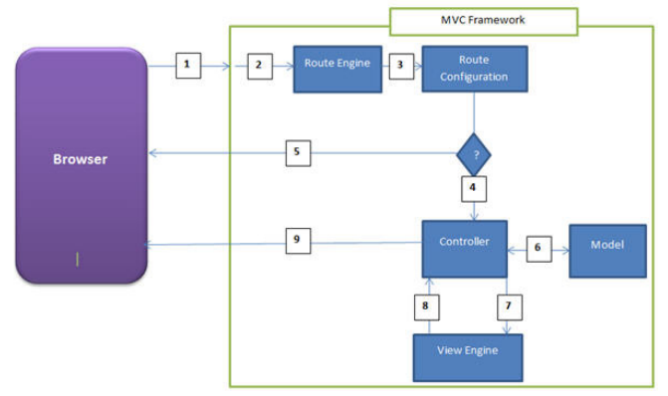
### **Giao tiếp trong kiến trúc ASP.NET MVC**

Các thành phần cơ bản của ứng dụng Asp.Net MVC bao gồm:

* MVC Framework
* Route engine
* Route configuration
* Control
* Model
* View engine
* View

Các thành phần này giao tiếp với nhau nhằm xử lý các yêu cầu của ứng dụng Asp.Net MVC. Quá trình xử lý yêu cầu thường liên quan tới một chuỗi các xử lý, mỗi xử lý sẽ được một component trong Asp.net đảm nhiệm.

1. Trình duyệt gửi yêu cầu của ứng dụng Asp.Net MVC
2. MVC Engine chuyển yêu cầu tới cho Routing engine (Bộ điều hướng)
3. Routing engine kiểm tra cấu hình điều hướng (route configuration) của ứng dụng nhằm chuyển đến các controller phù hợp với yêu cầu.
4. Khi Controller được tìm thấy, control này sẽ được thực thi
5. Nếu Controller không được tìm thấy, bộ điều hướng sẽ chỉ ra rằng controller không được tìm thấy và MVC Engine sẽ thông báo lỗi cho trình duyệt.
6. Controller giao tiếp với model, tầng Model đại diện cho các Entity bên trong hệ thống, Controller làm việc với tầng model để lấy thông tin cần cho người dùng.
7. Controller yêu cầu View engine hiển thị thông tin dựa trên dữ liệu của tầng model
8. View engine trả kết quả cho controller
9. Controller gửi kết quả cho trình duyệt thông qua giao thức Http.

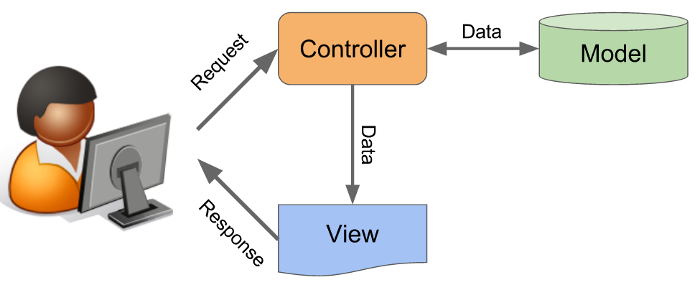


*Hình số: Giao tiếp trong kiến trúc Asp.Net MVC*

## Mô hình lập trình web MVC của ASP.NET5

Hệ thống được phát triển dựa trên nền tảng **ASP.Net MVC 5**. ASP.Net MVC là một framework web được phát triển bởi [**Microsoft**](https://gocnhinso.com/redirect-to/?url=https%3A%2F%2Fdotnet.microsoft.com%2F), thực thi mô hình MVC. MVC là tên một mẫu phát triển ứng dụng, phương pháp này chia nhỏ một ứng dụng thành ba thành phần để cài đặt, mỗi thành phần đóng một vai trò khác nhau và ảnh hưởng lẫn nhau, đó là Model, Views, và Controller. ASP.Net MVC đang nổi lên là phương pháp phát triển web mạnh nhất và phổ biển nhất trên nền .Net hiện nay.

**Mô hình MVC**



**Model:**Các đối tượng Models là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng model lấy và lưu trạng thái của model trong CSDL.

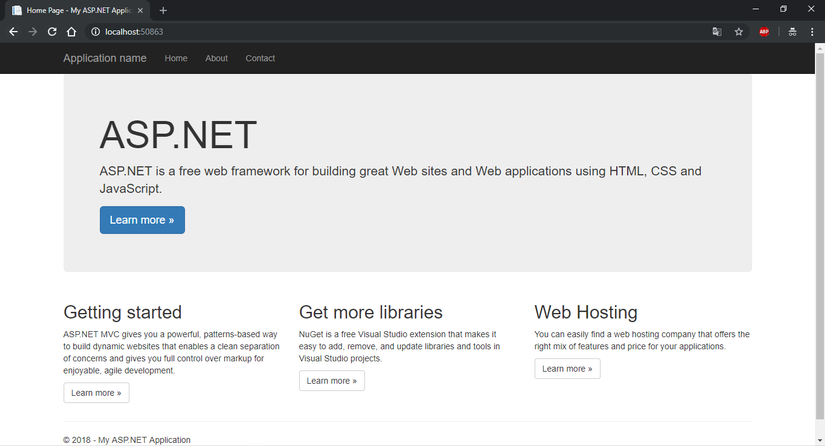
**Views:**Views là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, view được tạo dựa vào thông tin dữ liệu model.

**Controllers:**Controller là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng.

**Luồng xử lý dữ liệu trong MVC:**Khi có một yêu cầu từ phía Client gửi đến Server, bộ phận Controller có nhiệm vụ tiếp nhận yêu cầu, xử lý dữ liệu đầu vào và quyết định luồng đi tiếp theo (trả về kết quả hay tương tác với Cơ sở dữ liệu để lấy dữ liệu). Nếu cần tương tác với cơ sở dữ liệu, Controller sẽ gọi tới Model để lấy dữ liệu đầu ra. Model tương tác với Cơ sở dữ liệu để truy xuất dữ liệu phù hợp với yêu cầu của phía Controller, sau đó Model trả dữ liệu về cho Controller xử lý. Controller sẽ gọi đến View phù hợp với yêu cầu kèm theo dữ liệu cho View. View có trách nhiệm hiển thị kết quả, dữ liệu đầu ra phù hợp với yêu cầu của người dùng trên giao diện Web.

### Tính năng của ASP.Net MVC 5

* Tách bạch các tác vụ của ứng dụng (logic nhập liệu, business logic, và logic giao diện), dễ dàng kiểm thử và mặc định áp dụng hướng phát triển Test Driven Development (TDD). Tất cả các tính năng chính của mô hình MVC được cài đặt dựa trên interface và được kiểm thử bằng cách sử dụng các đối tượng mocks, mock object là các đối tượng mô phỏng các tính năng của những đối tượng thực sự trong ứng dụng. Bạn có thể kiểm thử unit-test cho ứng dụng mà không cần chạy controller trong tiến trình ASP.NET, và điều đó giúp unit test được áp dụng nhanh chóng và tiện dụng. Bạn có thể sử dụng bất kỳ nền tảng unit-testing nào tương thích với nền tảng .NET.

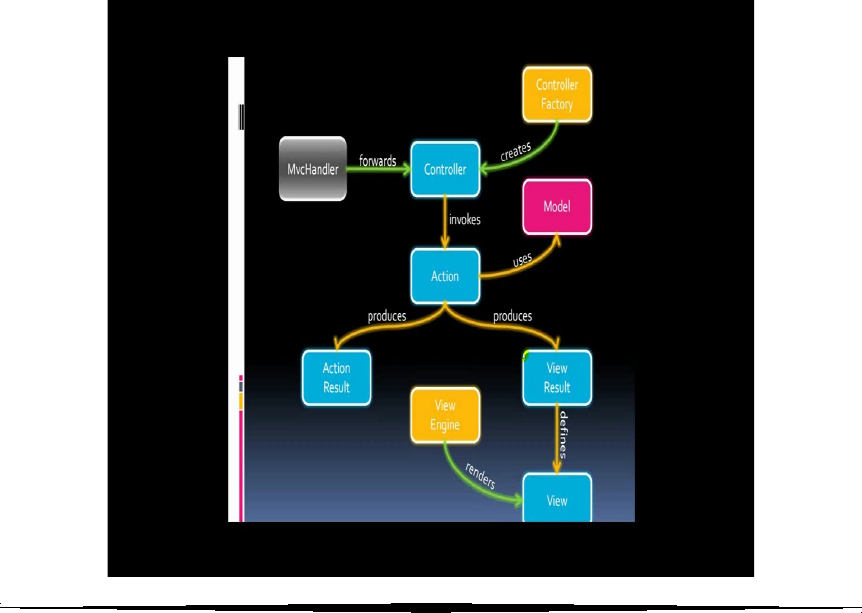


* MVC là một nền tảng khả mở rộng (extensible) & khả nhúng (pluggable). Các thành phần của **ASP.NET MVC 5** được thiết kể để chúng có thể được thay thế một cách dễ dàng hoặc dễ dàng tùy chỉnh. Bạn có thể nhúng thêm view engine, cơ chế định tuyến cho URL, cách kết xuất tham số của action-method và các thành phần khác. ASP.NET MVC cũng hỗ trợ việc sử dụng Dependency Injection (DI) và Inversion of Control (IoC). DI cho phép bạn gắn các đối tượng vào một lớp cho lớp đó sử dụng thay vì buộc lớp đó phải tự mình khởi tạo các đối tượng. IoC quy định rằng, nếu một đối tượng yêu cầu một đối tượng khác, đối tượng đầu sẽ lấy đối tượng thứ hai từ một nguồn bên ngoài, ví dụ như từ tập tin cấu hình. Và nhờ vậy, việc sử dụng DI và IoC sẽ giúp kiểm thử dễ dàng hơn.
* ASP.NET MVC có thành phần ánh xạ URL mạnh mẽ cho phép bạn xây dựng những ứng dụng có các địa chỉ URL xúc tích và dễ tìm kiếm. Các địa chỉ URL không cần phải có phần mở rộng của tên tập tin và được thiết kế để hỗ trợ các mẫu định dạng tên phù hợp với việc tối ưu hóa tìm kiếm (URL) và phù hợp với lập địa chỉ theo kiểu Representational State Transfer (REST)…
* Hỗ trợ các tính năng có sẵn của ASP.NET như cơ chế xác thực người dùng, quản lý thành viên, quyền, output caching và data caching, seession và profile, quản lý tình trạng ứng dụng, hệ thống cấu hình…

**ASP.NET MVC** **5**sử dụng view engine Razor View Engine cho phép thiết lập các view nhanh chóng, dễ dàng và tốn ít công sức hơn so với việc sử dụng Web Forms view engine.

## Hoạt động của các thành phần chính trong ASP.NET MVC

* Khi một request phát sinh từ một web browser đi đến IIS Web Server thì request đó cuối cùng được đưa đến 1 MVC Handler.
* MVC Handler có nhiệm vụ chọn ra đúng 1 Controller để xử lý request đó. Controller được tạo ra bởi 1 thành phần được gọi là Comtroller Factory
* Sau khi được tạo ra, Controller sẽ xác định request này được xử lý bởi
* Action Method cụ thể nào và sau đó sẽ thực thi Action Method đó. Action Method có thể tương tác với những Model Class để có thể truy xuất dữ liệu hoặc thực thi một số business logic.
* Sau khi hoàn tất xử lý Action method đó sẽ trả về 1 Action Result. ASP.NET MVC cung cấp nhiều Action Result, trong đó có 1 Action Result đặc biệt là View Result. Action Result này có nhiệm vụ làm việc 1 View nhất định để tạo ra mã HTML để trả về cho web browser và người dùng có thể nhận được kết quả xử lý.
* View Engine là thành phần thực thi 1 View, đi kèm với ASP.NET MVC là Webform View Engine tức là chúng ta có thể viết View bằng ASPX.



*Hình: Mô tả hoạt động MVC trong ASP.NET*

## Ưu điểm và khuyết điểm của ASP.NET

**Ưu điểm:**

* Không phải ngẫu nhiên mà ASP.NET được nhiều người dùng ưa chuộng và đang ngàng càng trở nên phổ biến hơn. Việc sở hữu một loạt ưu điểm nổi trội mang đầy tính thuyết phục, đặc biệt là độ bảo mật cao, đã giúp ASP.NET nhận được nhiều điểm cộng hơn từ người dùng.
* \*ASP.NET là ngôn ngữ lập trình có bộ thư việc phải nói là cực kỳ đa dạng và phong phú, được cung cấp bởi Net Framewrork nên có nhiều ưu điểm rất nổi bật. Chưa kể là ngôn ngữ lập trình này còn được hỗ trợ rất tốt về XML và khả năng truy cập cơ sở dữ liệu qua ADO.net.
* Ưu điểm nổi trội tiếp theo của ngôn ngữ lập trình này là có khả năng hoạt động tốt trên nhiều ứng dụng để đảm bảo hiệu suất tốt nhất, cao nhất cho website. Khi sử dụng ASP.NET, chúng ta có thể tách bạch phần code và giao diện thành 2 phần riêng biệt hoàn toàn để tạo ra những nét thú vị nhất cho web. Ngoài ra thì cũng nhờ vậy mà việc quản lý và bảo trì web trong quá trình vận hành, sử dụng đã trở nên dễ dàng và thuận tiện hơn rất nhiều.
* Các website được lập trình bởi ASP.NET thường hoạt động ổn định hơn, mượt mà hơn, có tốc độ tải trang tốt hơn so với một số ngôn ngữ khác, tạo sự thoải mái và cảm giác tin tưởng cao hơn nơi người dùng, từ đó giảm được tỉ lệ thoát trang và tăng thêm cơ hội khách hàng trở lại với website vào những lần sau. Tốc độ load của ASP.NET thậm chí còn được giới chuyên gia đánh giá cao hơn cả PHP và JS.
* Khả năng tùy biến của ASP.NET  là khá cao, giúp website dễ dàng tương thích với các kích thước màn hình, thiết bị truy cập khác nhau.
* ASP.NET  cũng cho phéo tạp ra các mã HTML một cách tự động cho các server để tạo điều kiện cho website hoạt động tốt trên bất kỳ trình duyệt nào.
* Độ bảo mật của ASP.Net rất cao do được kế thừa từ Java. Hiện nay, đây là ngôn ngữ lập trình được các chuyên gia trong ngành đánh giá là có khả năng chống tấn công mạng cao nhất.
* ASP.NET  có độ truy xuất dữ liệu cực nhanh, hỗ trợ lưu trữ dung lượng lớn  
  \*Hỗ trợ SEO tốt.
* Cho phép tùy biến, sửa đổi và mở rộng một cách dễ dàng khi có nhu cầu.

**Nhược điểm:**

* Ngoài những ưu điểm nổi trội mang đầy tính thuyết phục như đã kể trên thì hiện tại, ASP.NET cũng có một số nhược điểm tương đối lớn, chưa thể khắc phục được như: không hỗ trợ cho các thiết bị sử dụng hệ điều hành Linux. Bên cạnh đó, do có khá nhiều ưu điểm hấp dẫn như vậy nên ASP.NET có phí sử dụng khá cao, không phù hợp sử dụng cho các doanh nghiệp nhỏ hay các cá nhân.
* Tốn thời gian trung chuyển dữ liệu của thành phần

## Entity Framework

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

### Entity framework là gì?

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

**Note:** ORM là một công cụ tự động tạo ra các **Objects** trong ứng dụng từ **tables,views, store procedures** của một **Relational Database**.

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

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

### Tại sao lại sử dụng Entity Framework

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

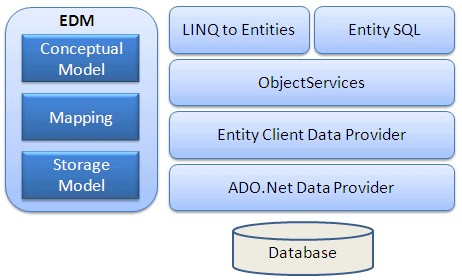
* Nếu bạn thực hiện truy vấn dữ liệu. Bạn có thể sử dụng Linq to Entitiesđể thao tác với objects được sinh ra từ Entity Framework, nghĩa là bạn không phải viết code sql.
* Việc update các classes,commands dễ dàng mỗi khi cơ sở dữ liệu có sự thay đổi, điều này giúp bạn tiết kiệm thời gian đáng kể.
* Entity Frameworksẽ tự động tạo ra các classes, commands tương ứng cho việc select, insert, update,delete dữ liệu từ cơ sở dữ liệu quan hệ.
* Entity Framework tự động tạo ra các classes cho việc truy suất cơ sở dữ liệu giúp lập trình viên giảm được thời gian viết code thao tác với database. Hỗ trợ bạn không phải mất quá nhiều thời gian cho việc viết code để thao tác với database.

Những tính năng đáng chú ý nhất của **Entity framework**:

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

### Cấu trúc Entity Framework

Cấu trúc được mô tả đơn giản qua hình sau:



*cấu trúc entity framework*

EDM (Entity Data Model):

EDM bao gồm 3 phần:

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

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

* Entity SQL:

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

* Object Service:

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

* Entity Client Data Provider:

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

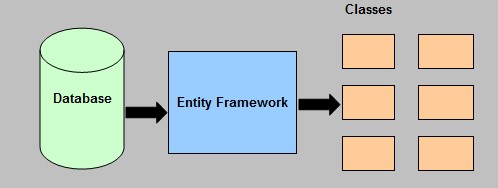
* ADO.Net Data Provider:

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

### Các tình huống dùng Entity Framework

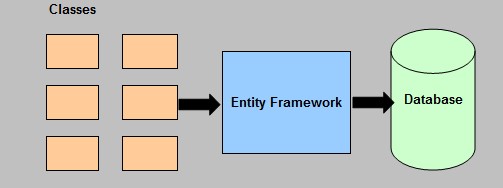
Entity framework là rất hữu ích trong 3 tình huống sau:

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



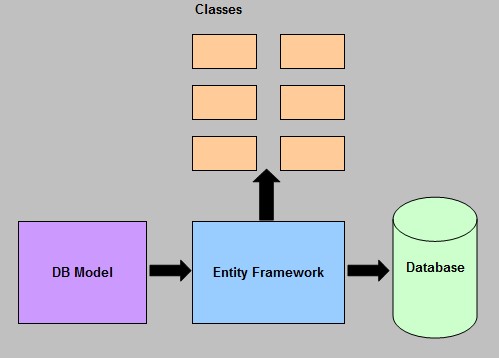
*Database First entity framework*

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



*Code First entity framework*

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



*Model First entity framework*

## Bootstrap

**Bootstrap** cho phép quá trình thiết kế website diễn ra nhanh chóng và dễ dàng hơn dựa trên những thành tố cơ bản sẵn có như typography, forms, buttons, tables, grids, navigation, image carousels…

### Bootstrap là gì?

Bootstrap là một Framework CSS được Twitter phát triển. Nó là một tập hợp các bộ chọn, thuộc tính và giá trị có sẵn để giúp web designer tránh việc lặp đi lặp lại trong quá trình tạo ra các Class CSS và những đoạn mã HTML giống nhau trong dự án web của mình. Ngoài CSS ra, thì bootstrap còn hỗ trợ các function tiện ích viết dựa trên Jquery.

***Bootstrap*** là một bộ sưu tập miễn phí của các ***mã nguồn mở*** và công cụ dùng để tạo ra một mẫu webiste hoàn chỉnh. Các **designer** có thể sáng tạo nhiều sản phẩm mới mẻ nhưng vẫn tiết kiệm thời gian khi làm việc với **framework** này trong quá trình ***thiết kế giao diện website***.

##### ****file chính của Bootstrap:****

Dưới đây là 3 File chính giúp quản lý các chức năng của Website và giao diện người dùng:

* **Bootstrap.CSS**

Nó là một **Framework CSS** có chức năng quản lý và sắp xếp bố cục của các trang Web. HTML có nhiệm vụ quản lý cấu trúc và nội dung thì CSS sẽ xử lý bố cục của Website. Vì vậy mà hai cấu trúc này phải tồn tại cùng nhau để thực hiện được hành động cụ thể.

Giờ bạn sẽ không phải tốn thời gian để chỉnh sửa thủ công chỉ để thay đổi một thiết kế nhỏ. Thay vào đó, bạn có thể dùng CSS để tạo giao diện thống nhất trên nhiều Website mà không bị giới hạn.

Hàm CSS không chỉ giới hạn ở kiểu văn bản mà còn dùng định dạng như bảng, bố cục hình. Bạn cần mất một ít thời gian để CSS ghi nhớ hết tất cả các khai báo và bộ chọn.

* **Bootstrap.JS**

File JavaScript sẽ chịu trách nhiệm cho việc tương tác của các trang Web

Đây là phần cốt lõi vì chứa các **File JavaScript**, nó chịu trách nhiệm cho việc tương tác của Website. Để tiết kiệm thời gian viết cú pháp JavaScript mà nhiều nhà phát triển sẽ sử dụng jQuery. Đây là thư viện JavaScript mã nguồn mở, đa nền tảng giúp bạn thêm nhiều chức năng vào trang Web.

Theo kinh nghiệm của tôi, jQuery sẽ thực hiện một số chức năng như:

* Thực hiện các yêu cầu của Ajax như: loại trừ dữ liệu đến từ một vị trí khác một cách linh hoạt.
* Giúp tạo các tiện ích bằng bộ sưu tập Plugin JavaScript.
* jQuery sẽ tạo hình động tùy chỉnh bằng các thuộc tính của CSS.
* Thêm một số tính năng động cho nội dung các trang Web của bạn.

Tuy **Bootstrap** với các thuộc tính Element HTML và CSS có thể hoạt động tốt, nhưng vẫn cần jQuery để tạo thiết kế Responsive. Nếu thiếu đi phần này, bạn chỉ có thể dùng các phần tĩnh của CSS mà thôi.

* **Glyphicons**

Trong giao diện trang Web, phần không thể thiếu chính là Icons. Chúng thường được liên kết với các dữ liệu nhất định và các hành động trong giao diện người dùng.

**Bootstrap** dùng Glyphicons để đáp ứng nhu cầu trên và nó đã mở khóa bộ Halflings Glyphicons để bạn sử dụng miễn phí. Tại bản miễn phí, tuy chỉ có giao diện chuẩn nhưng phù hợp với các chức năng thiết yếu.

Nếu như bạn muốn có các Icon phong cách hơn thì có thể mua bộ bộ Icon Premium khác nhau. Chắc chắn các Icon tại Glyphicons sẽ giúp cho trang Web của bạn trông đẹp và nổi bật hơn.

### Tại sao phải dùng bootstrap

Giữa muôn vàn ứng dụng thiết kế website hiện nay, **Bootstrap** vẫn có khả năng cạnh tranh cao là nhờ những đặc điểm nổi bật sau:

* Dễ dàng thao tác
* Tùy chỉnh dễ dàng
* Chất lượng sản phẩm đầu ra hoàn hảo
* Độ tương thích cao
* **Dễ dàng thao tác**

**Cơ chế hoạt động của Bootstrap** là dựa trên xu hướng **mã nguồn mở HTML, CSS và Javascript**. Người dùng cần trang bị kiến thức cơ bản 3 mã này mới có thể **sử dụng Bootstrap** hiệu quả. Bên cạnh đó, các mã nguồn này cũng có thể dễ dàng thay đổi và chỉnh sửa tùy ý.

* **Tùy chỉnh dễ dàng**

**Bootstrap** được tạo ra từ các mã nguồn mở cho phép designer linh hoạt hơn. Giờ đây có thể lựa chọn những thuộc tính, phần tử phù hợp với dự án họ đang theo đuổi. ***CDN Boostrap*** còn giúp bạn tiết kiệm dung lượng vì không cần tải mã nguồn về máy.

* **Chất lượng sản phẩm đầu ra hoàn hảo**

**Bootstrap** là sáng tạo của các lập trình viên giỏi trên khắp thế giới. **Bootstrap** đã được nghiên cứu và thử nghiệm trên các thiết bị. Được kiểm tra nhiều lần trước khi đưa vào sử dụng. Do đó, khi chọn **Bootstrap**, bạn có thể tin rằng mình sẽ tạo nên những sản phẩm với chất lượng tốt nhất.

* **Độ tương thích cao**

Điểm cộng lớn nhất của Bootstrap là khả năng tương thích với mọi trình duyệt và nền tảng. Đây là một điều cực kì quan trọng và cần thiết trong trải nghiệm người dùng. Sử dụng **Grid System** cùng với hai bộ tiền xử lý Less và Sass, **Bootstrap** mặc định hỗ trợ **Responsive** và ưu tiên cho các giao diện trên thiết bị di động hơn. Bootstrap có khả năng tự động điều chỉnh kích thước trang website theo khung browser. Mục đích để phù hợp với màn hình của máy tính để bàn, tablet hay laptop.

## Jquery

### Jquery là gì

Jquery là thư viện được viết từ JavaScript, Jquery giúp xây dựng các chức năng bằng Javascript dễ dàng, nhanh và giàu tính năng hơn.

**jQuery** được tích hợp nhiều module khác nhau. Từ module hiệu ứng cho đến module truy vấn selector. **jQuery** được sử dụng đến 99% trên tổng số website trên thế giới. Vậy các module chính của **jQuery** là gì?

Các module phổ biến của jQuery bao gồm:

* **Ajax**– xử lý Ajax
* **Atributes**– Xử lý các thuộc tính của đối tượng HTML
* **Effect**– xử lý hiệu ứng
* **Event**– xử lý sự kiện
* **Form**– xử lý sự kiện liên quan tới form
* **DOM**– xử lý Data Object Model
* **Selector**– xử lý luồng lách giữa các đối tượng HTML

*jQuery là một thư viện JavaScript, giúp đơn giản hóa việc lập trình JavaScript*

**jQuery** không phải là một ngôn ngữ lập trình riêng biệt mà hoạt động liên kết với JavaScript. Với jQuery, bạn có thể làm được nhiều việc hơn mà lại tốn ít công sức hơn. jQuery cung cấp các API giúp việc duyệt tài liệu HTML, hoạt ảnh, xử lý sự kiện và thao tác AJAX đơn giản hơn. jQuery hoạt động tốt trên nhiều loại trình duyệt khác nhau. Một trong những đối thủ nặng ký của jQuery đó là JS Framework.

### Jquery giúp gì cho người lập trình

* ***Gọn nhẹ***: **jQuery** là một thư viện khá gọn nhẹ – có kích cỡ khoảng 19KB (gzipped).
* ***Tương thích đa nền tảng***: Nó tự động sửa lỗi và chạy được trên mọi trình duyệt phổ biến như Chrome, Firefox, Safari, MS Edge, IE, Android và iOS.
* ***Dễ dàng tạo Ajax****:* Nhờ thư viện **jQuery**, code được viết bởi Ajax có thể dễ dàng tương tác với ***server*** và cập nhật nội dung tự động mà không cần phải tải lại trang.
* ***Xử lý nhanh nhạy thao tác DOM***: **jQuery** giúp lựa chọn các phần tử DOM để traverse (duyệt) một cách dễ dàng, và chỉnh sửa nội dung của chúng bằng cách sử dụng Selector mã nguồn mở, mà còn được gọi là Sizzle.
* ***Đơn giản hóa việc tạo hiệu ứng***: Giống với code snippet có hiệu ứng animation, nó phủ các dòng code và bạn chỉ việc thêm biến/nội dung vào thôi.
* ***Hỗ trợ tốt phương thức sự kiện HTML***: Xử lý sự kiện − **jQuery** xử lý các sự kiện đa dạng mà không làm cho HTML code trở nên lộn xộn với các Event Handler.
* Truy cập các phần tử trong nội dung trang web.
* Jquery cho phép bạn chọn bất cứ thành phần nào của tài liệu để “vọc” một cách dễ dàng như sử dụng CSS.
* Thay đổi hình thức giao diện của trang web
* Jquery cũng có thể thay đổi class hoặc những định dạng CSS đã được áp dụng lên bất cứ thành phần nào của HTML ngay cả khi trang web đó đã được trình duyệt load thành công,
* Thay đổi nội dung trang web.

### Tại sao nên dùng Jquery

Có rất nhiều lý do để mình khuyên các bạn nên dùng jquery thay vì dùng javascript phải nói đến đầu tiên đó chính là số plugin của jquery khá là khổng lồ giúp bạn tiết kiệm được kha khá thời gian viết các ứng dụng liên quan đến giao diện, các hiệu ứng.

…

Hơn nữa, jquery đơn giản hoá cách viết Javascript và tăng tốc độ xử lý các sự kiện trên trang web, giúp tiết kiệm thời gian và công sức rất nhiều so với việc ngồi viết javascript theo cách thông thường. Đúng như khẩu hiệu của nó Viết ít hơn, làm nhiều hơn, nếu không dùng Jquery thì một chuyên gia cũng phải viết vài trang giấy mới làm được nút Back to Top với hiệu ứng trượt, trong khi đó dân nghiệp dư chỉ cần vài dòng code ngắn ngủi. Jquery là đã có thể “Múa rìu qua mắt thợ”. Bên cạnh đó, việc sử dụng Jquery giúp giải quyết tốt vấn đề xung đột trình duyệt web.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Phân tích và thiết kế hệ thống

### Yêu cầu cơ bản của hệ thống

* Yêu cầu lưu trữ thông tin
* Yêu cầu thêm, sửa, xoá, cập nhật thông tin về học sinh và điểm của học sinh
* Yêu cầu tìm kiếm, xuất kết quả về thông tin học sinh theo:
* Tên học sinh
* Điểm
* Yêu cầu xuất bảng điểm học sinh
* Yêu cầu xuất tổng kết môn học, tổng kết hàng tháng của lớp

### Mô tả chức năng ứng dụng

## Phân tích hệ thống

### Sơ đồ cơ sở dữ liệu

### Lược đồ phân cấp chức năng

## Công nghệ áp dụng

Ckeditor:

Bootstrap:

Charts:

Datapicker:

Jquery:

# GIAO DIỆN DEMO

#### KẾT LUẬN

Trong thời gian vừa qua với sự nỗ lực của cả nhóm và dưới sự hướng dẫn tận tình đến từ thầy Lương Trần Hy Hiến thì nhóm em đã hoàn thành xong luận án với đề tài “Quản lý lớp học” với các mục tiêu đã đặt ra trước đó. Luận án có sử dụng thông tin trích dẫn từ nhiều nguồn tham khảo khác nhau và các thông tin trích dẫn được ghi rõ nguồn gốc. Dưới đây là một số kết quả mà luận án đã đạt được.

* Nghiên cứu tổng quan về môn Công nghệ Net và một số công nghệ tích hợp: Phân tích, làm rõ các công nghệ để dựa trên cơ sở đó, đề xuất một số hướng nghiên cứu về các chức năng trong đề tài “Quản lý lớp học”.
* Nắm bắt được các khái niệm cơ bản về các công nghệ, chỉ ra những ưu, nhược điểm và các tính năng của chúng, phân tích hiệu năng của từng công nghệ đối với các chức năng. Biết được các công cụ có thể điều khiển, sử dụng hoạt động của trang Web một cách chính xác.
* Thông qua các bài học, tài liệu đã tham khảo được, nhóm em đã mô tả các chức năng cần thiết, ứng dụng các công nghệ vào hệ thống “Quản lý lớp học”, ... Thiết kế cài đặt cho hệ thống bao gồm: Giao diện trang Web, các chức năng xử lý chính, …

Nhóm em xin được bày tỏ lời cảm ơn chân thành tới thầy Lương Trần Hy Hiến đã hướng dẫn và truyền đạt những kinh nghiệm, bài học quý báu cho nhóm em trong thời gian học và làm luận án.

Luận án của nhóm em còn những hạn chế về năng lực và những thiếu sót trong quá trình nghiên cứu. Em xin lắng nghe và tiếp thu những ý kiến của giáo viên phản biện để hoàn thiện, tiếp tục bổ sung kiến thức.

Em xin chân thành cảm ơn!

#### TÀI LIỆU THAM KHẢO