**TRƯỜNG ĐẠI HỌC SƯ PHẠM ĐÀ NĂNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO TỔNG KẾT MÔN CÔNG NGHỆ PHẦM MỀM**

Bài tập nhóm số 10: **Quản Lý Đăng Ký Môn Học Trường(ĐHSP-ĐN)**

Người thực hiên**:**  1**.** **PHẠM HÀNG KIM MI**

MSSV**:** **3120519041**

2.**DƯƠNG THI KIM THẢO**

MSSV: **3120519053**

3**. PHẠM CAO TRINH**

MSSV: **3120519058**

Lớp**: 19CNTTD**

Khóa**: 2019-2023**

tháng 05 năm 2022

# LỜI CẢM ƠN

Thời gian được học tập, nghiên cứu tại [TRƯỜNG ĐẠI HỌC SƯ PHẠM ĐÀ NẴNG](https://ued.udn.vn/) là khoảng thời gian tuyệt vời nhất đối với chúng em. Chúng em xin gửi lời cảm ơn chân thành đến toàn thể các thầy cô trong trường nói chung, các thầy cô trong khoa Công nghệ thông tin nói riêng, thầy cô đã luôn tạo điều kiện thuận lợi cho em trong quá trình học tập, cho em những bài học quý giá, truyền bao nhiêu kiến thức và động lực để cố gắng trong suốt thời gian qua.

Đặc biệt em xin gửi lời cảm ơn đến Thầy Võ Công Đình người đã tận tình giúp đỡ, chỉ bảo và hướng dẫn em trong suốt quá trình thực hiện đồ án Cơ Sở Dữ Liệu Nâng Cao. Nhờ những hướng dẫn của cô mà đồ án của em mới có thể hoàn thành và đạt được kết quả như hôm nay.

Lời cuối cùng chúng em xin gửi lời cảm ơn đến anh chị, bạn bè, các em khóa sau đã nhiệt tình hỗ trợ trả lời phỏng vấn qua Google Form cũng như các thành viên khác trong nhóm không quản khó khăn , tích cực cùng nhau học tập, thảo luận , làm việc hết mình để đạt thành quả như hôm nay. Thành công của nhóm chúng em ngày hôm nay một phần quan trọng là có sự đóng góp của tất cả mọi người.

Mặc dù chúng em đã rất cố gắng hoàn thành đồ án nhưng chắc chắn không thể tránh khỏi thiếu sót. Kính mong thầy cô đóng góp ý kiến để em có thể hoàn thiện kiến thức của bản thân.

Chúng em xin chân thành cảm ơn!

**MỤC LỤC**

[LỜI CẢM ƠN 1](#_Toc104761974)

[MỞ ĐẦU 6](#_Toc104761975)

[I.Giới Thiệu: 6](#_Toc104761976)

[II.Giới Thiệu Cở Sở Dữ Liệu Tập trung 7](#_Toc104761977)

[1. Hệ CSDL cá nhân 7](#_Toc104761978)

[2. Hệ CSDL trung tâm 7](#_Toc104761979)

[3. Hệ CSDL khách - chủ 8](#_Toc104761980)

[CHƯƠNG 1:CƠ SỞ LÝ THUYẾT 9](#_Toc104761981)

[1.Giới Thiệu: 9](#_Toc104761982)

[1.2.Cơ Sở Lý Thuyết: 9](#_Toc104761983)

[1.3.Tổng Quan Về Ngôn Ngữ Lập Trinh: 10](#_Toc104761984)

[1.3.1. HTML 10](#_Toc104761985)

[1.3.2. CSS 12](#_Toc104761986)

[1.3.3. BOOTSTRAP 14](#_Toc104761987)

[1.3.4. JAVASCRIPT 16](#_Toc104761988)

[1.3.5.JQUERY 18](#_Toc104761989)

[1.3.6. Visual Studio 2019: 21](#_Toc104761990)

[1.3.7. NuxtJS 26](#_Toc104761991)

[1.3.8. ASP.net Core: 32](#_Toc104761992)

[1.3.9. Visual Studio Code 34](#_Toc104761993)

[1.3.10. Vue JS 37](#_Toc104761994)

[1.4.Hệ Quản Trị Cơ Sở Dữ Liệu 45](#_Toc104761995)

[1.4.1. Mục Đích Sử Dụng Cơ Sở Dữ Liệu 46](#_Toc104761996)

[1.4.2. Các kiểu dữ liệu thông dụng trong cơ sở dữ liệu MICROSOFT SQL SERVER 46](#_Toc104761997)

[1.4.3. Các thao tác cập nhật dữ liệu 49](#_Toc104761998)

[1.4.4. Các hàm thông dụng trong MySQL 50](#_Toc104761999)

[CHƯƠNG 2:PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 51](#_Toc104762000)

[2.1. Khảo sát và phần tích hệ thống 51](#_Toc104762001)

[2.1.1. Mục tiêu của hệ thống shop thời trang 51](#_Toc104762002)

[2.1.2. Lợi ích của việc xây dựng website 51](#_Toc104762003)

[2.1.3. Mô tả yêu cầu của hệ thống 51](#_Toc104762004)

[2.1.4. Mô tả hoạt động của hệ thống 51](#_Toc104762005)

[2.2. Biểu đồ Use case 52](#_Toc104762006)

[2.2.1.Sơ đồ use case 53](#_Toc104762007)

[2.2.2. Use-case đăng nhập Admin: 54](#_Toc104762008)

[2.2.3. Use-case khách hàng tiềm năng: 56](#_Toc104762009)

[2.2.7. Use-case khách hàng 58](#_Toc104762010)

[2.2.8. Danh sách các Actor 60](#_Toc104762011)

[CHƯƠNG 3:PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU 61](#_Toc104762012)

[3.1.Biểu Đồ Diagram Bán Đồ Ăn 61](#_Toc104762013)

[3.1.1.Đăng Nhập 61](#_Toc104762014)

[3.12. Hóa Đơn 62](#_Toc104762015)

[3.2.Bảng LoaiHang 62](#_Toc104762016)

[3.3.Bảng MatHang 62](#_Toc104762017)

[3.4.Bảng DonDatHang 63](#_Toc104762018)

[3.5.Bảng CTDonDatHang 63](#_Toc104762019)

[3.6.Bảng KhachHang 63](#_Toc104762020)

[3.7.Database 65](#_Toc104762021)

[KẾT LUẬN 76](#_Toc104762022)

[1. Những việc đã làm được 77](#_Toc104762023)

[2.Những việc chưa làm được 77](#_Toc104762024)

[3.Hướng phát Triển 77](#_Toc104762025)

[TÀI LIỆU THAM KHẢO 78](#_Toc104762026)

**DANH MỤC HÌNH**

[Hình 1:Bảng khảo sát 11](#_Toc90891799)

[Hình 2:Cài đặt hình ảnh MS SQL Server 14](#_Toc90891800)

[Hình 3:Cài đặt hình ảnh MS SQL Server 15](#_Toc90891801)

[Hình 4:Cài đặt hình ảnh MS SQL Server 16](#_Toc90891802)

[Hình 5:Cài đặt hình ảnh MS SQL Server 17](#_Toc90891803)

[Hình 6:Cài đặt hình ảnh MS SQL Server 18](#_Toc90891804)

[Hình 7:khởi động MS SQL Server 19](#_Toc90891805)

[Hình 8:draw 19](#_Toc90891806)

[Hình 9:Các bước vô diagram 20](#_Toc90891807)

[Hình 10:Các bước vô diagram 21](#_Toc90891808)

[Hình 11:Trang chủ phần mềm vẽ ER 21](#_Toc90891809)

[Hình 12:Bảng biêu diễn mối quan hệ 23](#_Toc90891810)

[Hình 13:Sơ Đồ ER 25](#_Toc90891811)

[Hình 14:Bước 1 chạy DR 29](#_Toc90891812)

[Hình 15:Bước 2 chạy DR 30](#_Toc90891813)

[Hình 16: Bước 3 Chạy DR 31](#_Toc90891814)

[Hình 17:Sơ Đồ DR 32](#_Toc90891815)

**NHẬN XÉT CỦA CÁN BỘ HƯỚNG DẪN**

**………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

***Đà Nẵng, ngày ….. tháng ….. năm …..***

**Cán bộ hướng dẫn**

*(Ký và ghi rõ họ tên)*

**NHẬN XÉT CỦA HỘI ĐỒNG PHẢN BIỆN**

*Đà Nẵng, ngày tháng năm 2019*

***Hội đồng phản biện***

# MỞ ĐẦU

## I.Giới Thiệu:



**Công nghệ phát triển không ngừng, sở hữu một Website ẩm thực chuyên nghiệp có thể giúp bạn quảng bá được hình ảnh của nhà hàng, chiếm trọn tình cảm của khách hàng. Trong bài viết này, Trần Lê sẽ mang đến những thông tin về tầm quan trọng của trang Web nhà hàng ẩm thực và giúp bạn lựa chọn được dịch vụ thiết kế Website uy tín nhất.**

**Lợi ích của dịch vụ thiết kế Website nhà hàng ẩm thực.**

**-Tiếp cận khách hàng nhanh chóng:**

**Theo nghiên cứu gần đây nhất chỉ ra rằng mỗi ngày người ta dành từ 2 đến 3 giờ để tìm kiếm cũng như mua sắm trên mạng Internet. Bên cạnh đó, các chuyên gia còn cho biết mọi người thường hay quyết định mua sắm sau khi đã tìm hiểu các thông tin đó. Vậy nên sở hữu một trang Web nhà hàng ẩm thực đem lại rất nhiều lợi ích cho bạn.**

**Không chỉ dừng lại ở việc tìm kiếm thông tin tên nhà hàng, món ăn mà quán cung cấp, Website còn giúp người dùng khi tìm kiếm có thể đến được trang Web ngay giữa hàng trăm, hàng ngàn thông tin của các nhà hàng, quán ăn khác. Đây được coi là điểm khác biệt mà bạn có thể tạo ra với các đối thủ. Đồng thời, cũng giúp hoạt động kinh doanh của nhà hàng được phát triển tốt hơn. Lượng truy cập vào Web càng nhiều thì cơ hội bán hàng của bạn sẽ càng lớn.**

**-Quảng bá hình ảnh nhà hàng:**

**Khi thiết kế Website đồng nghĩa với việc bạn sẽ cung cấp thông tin, hình ảnh về món ăn, tin tức cũng như menu của quán. Đây là cách giúp khách hàng có thể nhanh chóng nắm rõ được thông tin về cơ sở kinh doanh của bạn.**

**Trang Web chuyên nghiệp từ hình ảnh món ăn cho đến không gian quán được cập nhật sẽ truyền tải được thông điệp đến với khách hàng một cách chân thực nhất. Nhờ vậy nó không chỉ thu hút được khách hàng mà còn tạo hiệu quả trong việc quảng cáo thương hiệu cực kỳ tốt. Nó làm tăng giá trị thương hiệu nhà hàng lên cao hơn và được nhiều người biết đến hơn.**

**-Khách hàng có thể đặt bàn trực tuyến thuận tiện**

**Trước đây mọi người thường hay đặt bàn nhà hàng qua điện thoại, điều này tốn khá nhiều thời gian của cả nhà hàng và khách. Việc phát triển công nghệ thiết kế Website cho phép đặt bàn trực tuyến đem lại rất nhiều lợi ích to lớn.**

**Chỉ cần một vài thao tác đơn giản khách hàng có thể kiểm tra được các vị trí trống của nhà hàng, có thể đặt chỗ một cách nhanh chóng và nhà hàng cũng sẽ xác nhận được ngay lập tức. Nhờ thiết kế Website mà cách thức hoạt động của nhà hàng trở nên chuyên nghiệp hơn. Bạn có thể cập nhật được lượng khách hàng của mình, vị trí đặt, số lượng khách hàng,… từ đó có sự chuẩn bị chu đáo hơn.**

**Đó là lý do em chọn đề tài “Quản lý đăng kí môn học” làm đồ án Công nghệ phần mềm.**

## ****II.Giới Thiệu Cở Sở Dữ Liệu Tập trung****

- Với hệ CSDL tập trung, toàn bộ dữ liệu được lưu trữ tại một máу hoặc một dàn máу. Những người dùng từ хa có thể truу cập ᴠào CSDL thông qua các phương tiện truуền thông dữ liệu. Nói chung có ba kiểu kiến trúc tập trung:

### 1. Hệ CSDL cá nhân

- Hệ CSDL cá nhân là hệ CSDL có một người dùng, người nàу ᴠừa thiết kế, ᴠừa tạo lập, ᴠừa cập nhật ᴠà bảo trì CSDL, đồng thời cũng là người khai thác thông tin, tự lập ᴠà hiển thị các báo cáo.Bạn đang хem: Cơ ѕở dữ liệu phân tán là gì

### 2. Hệ CSDL trung tâm

- Hệ CSDL trung tâm là hệ CSDL ᴠới dữ liệu được lưu trữ trên máу tính trung tâm, nhiều người ѕử dụng từ хa có thể truу cập CSDL nàу thông qua các thiết bị đầu cuối ᴠà các phương tiện truуền thông. Tùу thuộc ᴠào quу mô của tổ chức, máу tính trung tâm nàу là một dàn máу haу một máу. Các hệ CSDL trung tâm thường rất lớn ᴠà có nhiều người dùng, ᴠí dụ các hệ thống đăng kí ᴠà bán ᴠé máу baу, các hệ thống thông tin của tổ chức tài chính,…

### 3. Hệ CSDL khách - chủ

- Trong kiến trúc khách-chủ, các thành phần (của hệ QTCSDL) tương tác ᴠới nhau tạo nên hệ thống gồm thành phần уêu cầu tài nguуên ᴠà thành phần cấp tài nguуên. Hai thành phần nàу không nhất thiết phải cài đặt trên cùng một máу tính. Thành phần cấp tài nguуên thường được cài đặt tại một máу chủ trên mạng (cục bộ)Còn thành phần уêu cầu tài nguуên có thể cài đặt tại nhiều máу khác trên mạng (ta gọi là các máу khách).

- Phần mềm CSDL trên máу khách quản lí các giao diện khi thực hiện chương trình.

- Kiến trúc loại nàу có một ѕố ưu điểm ѕau:

Khả năng truу cập rộng rãi đến các CSDL.Nâng cao khả năng thực hiện: các CPU ở máу chủ ᴠà máу khách khác nhau có thể cùng chạу ѕong ѕong, mỗi CPU thực hiện nhiệm ᴠụ của riêng nó.Chi phí cho phần cứng có thể được giảm do chỉ cần máу chủ có cấu hình đủ mạnh để lưu trữ ᴠà quản trị CSDL.Chi phí cho truуền thông được giảm do một phần các thao tác được giải quуết trên máу khách, chỉ cần: уêu cầu ᴠề truу cập CSDL gửi đến máу chủ ᴠà dữ liệu kết quả gửi ᴠề cho máу khách.Nâng cao khả năng đảm bảo tính nhất quán của dữ liệu ᴠì các ràng buộc được định nghĩa ᴠà kiểm tra chỉ tại máу chủ.Kiến trúc nàу phù hợp ᴠới ᴠiệc mở rộng các hệ thống.

# CHƯƠNG 1:CƠ SỞ LÝ THUYẾT

## 1.Giới Thiệu:

Trong thời buổi hiện nay, khi mà công nghệ phát triển, thương mại điện tử ngày càng phát triển trên thế giới cũng như Việt Nam và dần khẳng định được vị thế của mình trong nền kinh tế thị trường. Cùng với đó là sự ra đời của các ngôn ngữ lâp trình cho phép thiết kế và xây dựng các ứng dụng thương mại điện tử dưới nhiều hình thức khác nhau. Và một trong những ứng dụng của thương mại điện tử phổ biến ở nước ta là dịch vụ bán hàng qua mạng internet. Dịch vụ này cho phép người dùng tìm kiếm chọn lựa sản phẩm để mua, thực hiện giao dịch mà không cần phải trực tiếp đến cửa hàng, mà chỉ cần sử dụng một thiết bị máy tính có kết nối internet. Tuy nhiên hiện nay dịch vụ này vẫn còn khá mới với nhiều người tiêu dùng nước ta, đặc biệt là những người không có kiến thức về tin học nói chung và thương mại điện tử nói riêng. Nên hiện giờ các đối tượng chính của thương mại điện tử vẫn chủ yếu là tầng lớp tri thức và học sinh, sinh viên. Là những người sinh viên, cũng có chút kiến thức về tin học cũng như tiếp xúc với thương mại điện tử tuy nhiên đôi lúc cũng gặp một số khó khăn trong việc mua bán trên các hệ thống bán hàng qua mạng hiện tại. Như hình thức thanh toán, nhận hàng, chất lượng sản phẩm thực tế không được như trên website...

Hiểu được các vấn đề đó cũng như mong muốn đưa thương mại điện tử đến với nhiều người hơn chúng em thực hiện đề tài: Phân tích và thiết kế hệ thống website bán giày qua mạng. Với mục đích xây dựng một hệ thống bán hàng qua mạng uy tín, đơn giản, thân thiện, cũng như đem lại cho khách hàng những lựa chọn tối ưu để ngay cả những người không có kiến thức về tin học cũng có thể tham gia mua hàng qua mạng chỉ với một số thao tác đơn giản. Sản phẩm mà bọn em hướng tới là các thiết bị tin học nói chung, và thiết bị máy tính nói riêng. Vì sản phẩm này đã dần trở thành vô cùng cần thiết cho nhu cầu học tập, làm việc, giải trí...

## 1.2.Cơ Sở Lý Thuyết:

Tìm hiểu về ngôn ngữ lập trình, công nghệ sử dụng để xây dựng và cài đặt cho bài toán hỗ trợ quản lý.

* HTML – xây dựng cấu trúc định dạng và các siêu văn bản
* CSS – định dạng các siêu văn bản dạng thô tạo ra từ HTML thành một bố cục website, có màu sắc, ảnh nền, …
* Bootstrap – một framework CSS được Twitter phát triển.
* Javascript – là nhẹ và được sử dụng phổ biến nhất như là một phần của các trang web, mà sự thi hành của chúng cho phép Client-Side script tương tác với người sử dụng và tạo các trang web động.
* MySQL – hệ quản trị cơ sở dữ liệu truy vấn có cấu trúc làmột trong những công cụ quản lý dữ liệu được sử dụng phổ biến ở nhiều lĩnh vực. ... Do sự phát triển nhanh chóng của nhu cầu xây dựng những CSDL lớn theo mô hình khách/chủ (Client/Server)
* ASP.net Core – là một open-source mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile.
* VueJS – là một framework linh động dùng để xây dựng giao diện người dung. Khi phát triển lớp giao diện (view layer), người dùng chỉ cần dùng thư viện lõi (core library) của Vue, vốn rất dễ học và tích hợp với các thư viện hoặc dự án có sẵn.
* NuxtJS - là một framework js để tạo các ứng dụng Universal Vue.js. Và bên cạnh đó, nó rất nhiều tính năng giúp bạn phát triển giữa phía client và server như Dữ liệu bất đồng bộ (Asynchronous Data), Middleware, Layouts, v.v.
* Visual Studio Code - là một trong những trình soạn thảo mã nguồn phổ biến nhất được sử dụng bởi các lập trình viên. Nhanh, nhẹ, hỗ trợ đa nền tảng, nhiều tính năng và là mã nguồn mở chính.
* Visual Studio 2019 - là một trong những công cụ hỗ trợ [lập trình website](https://monamedia.co/dich-vu/thiet-ke-website/) rất nổi tiếng nhất hiện nay của Mcrosoft và chưa có một phần mềm nào có thể thay thế được nó. Visual Studio được viết bằng 2 ngôn ngữ đó chính là C# và VB+.

## 1.3.Tổng Quan Về Ngôn Ngữ Lập Trinh:

### 1.3.1. HTML

HTML (tiếng Anh, viết tắt cho HyperText Markup Language, hay là "Ngôn ngữ Đánh dấu Siêu văn bản") là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web với các mẩu thông tin được trình bày trên World Wide Web. HTML được định nghĩa như là một ứng dụng đơn giản và được sử dụng trong các tổ chức cần đến các yêu cầu xuất bản phức tạp. HTML đã trở thành một chuẩn Internet do tổ chức World Wide Web Consortium (W3C) duy trì. Phiên bản chính thức mới nhất của HTML là HTML 4.01 (1999). Sau đó, các nhà phát triển đã thay thế nó bằng XHTML. Hiện nay, HTML đang được phát triển tiếp với phiên bản HTML5 hứa hẹn mang lại diện mạo mới cho Web.

Bằng cách dùng HTML động hoặc Ajax, lập trình viên có thể tạo ra và xử lý bởi số lượng lớn các công cụ, từ một chương trình soạn thảo văn bản đơn giản – có thể gõ vào ngay từ những dòng đầu tiên – cho đến những công cụ xuất bản WYSIWYG phức tạp.

* HTML không phải là ngôn ngữ lập trình, nó là ngôn ngữ trình bày.
* HTML là một **đánh dấu** cho ngôn ngữ **mô tả** các tài liệu web (trang web).
* Ngôn ngữ đánh dấu là một tập hợp các **thẻ đánh dấu.**
* Tài liệu HTML được mô tả bởi **các thẻ HTML.**
* Mỗi thẻ HTML **mô tả** nội dung tài liệu khác nhau.
* Cấu trúc một đoạn HTML:
* Các **DOCTYPE** khai báo xác định các loại tài liệu là HTML.
* Các văn bản giữa **<html>** và **</ html>** mô tả một tài liệu HTML.
* Các văn bản giữa **<head>** và **</ head>** cung cấp thông tin về tài liệu.
* Các văn bản giữa **<title>** và **</ title>** cung cấp một tiêu đề cho tài liệu.
* Các văn bản giữa **<body>** và **</ body>** mô tả nội dung trang có thể nhìn thấy.
* Các văn bản giữa **<h1>** và **</ h1>** mô tả một đề mục.
* Các văn bản giữa **<p>** và **</ p>** mô tả đoạn.
* Trong các thẻ còn có **các thuộc tính**, thuộc tính sẽ đặt bên trong thẻ mở đầu, mỗi thuộc tính sẽ có giá trị được đặt trong dấu ngoặc kép và cách nhau bởi dấu bằng (=) với tên thuộc tính:

**<form action="http://google.com"> </form>**

Sử dụng mô tả này, một trình duyệt web có thể hiển thị một tài liệu với một tiêu đề, đoạn văn và một link.

Khi một tập tin HTML được hình thành, việc xử lý nó sẽ do trình duyệt web đảm nhận. Trình duyệt sẽ đóng vai trò đọc hiểu nội dung HTML từ các thẻ bên trong và sẽ chuyển sang dạng văn bản đã được đánh dấu để đọc, nghe hoặc hiểu.

### 1.3.2. CSS

CSS là chữ viết tắt của cụm từ tiếng anh (Cascading Style Sheet), CSS được tổ chức World Wide Web (W3C) giới thiệu vào năm 1996. CSS chỉ đơn thuần là một file có phần mở rộng là .css, trong file này chứa những câu lệnh CSS, mỗi câu lệnh CSS sẽ định dạng một thành phần nhất định của tài liệu HTML như màu sắc, font chữ, ...

CSS tách riêng phần định dạng ra khỏi nội dung trang web giúp người thiết kế kiểm soát toàn bộ giao diện nhanh nhất và hiệu quả nhất, tiết kiệm được nhiều thời gian trong chỉnh sửa, đặc biệt là các dự án lớn được kết cấu từ nhiều trang HTML.  
 Do được tách rời khỏi nội dung của trang web, nên các mã HTML sẽ gọn gàng hơn, giúp bạn thuận tiện hơn trong việc chỉnh sửa giao diện. Kích thước của file .html cũng được giảm đáng kể. Hơn thế nữa sẽ được trình duyệt tải một lần và dùng nhiều lần (cache), do đó giúp trang web được load nhanh hơn.

CSS có nhiều kiểu nhưng gom lại thì chúng ta sẽ có các thể loại CSS thông dụng như sau:

* **Background**: CSS tùy chỉnh hình nền.
* **Text**: CSS tùy chỉnh cách hiển thị đoạn text.
* **Font**: CSS tùy chỉnh kích thước, kiểu chữ.
* **Link**: CSS tùy chỉnh link.
* **List**: CSS tùy chỉnh danh sách.
* **Table**: CSS tùy chỉnh bảng.
* **Box model**: Mô hình box model kết hợp padding, margin, border.
* ...

Chúng ta có 3 cách để khai báo CSS trong tài liệu HTML là:

* CSS cục bộ: Viết mã CSS trực tiếp trong thẻ HTML, cụ thể là trong thuộc tính style, CSS cục bộ chỉ có tác dụng trong thẻ HTML được khai báo.

**<p style="font-size: 16pt; color: blue">Đoạn văn bạn sẽ bị ảnh hưởng khi khai báo CSS cục bộ. </p>**

* CSS nội tuyến: CSS nội tuyến là phần mã CSS được khai báo trong cặp thẻ <style> và đặt trong phần <head> của tài liệu HTML. Khi sử dụng CSS nội tuyến nó sẽ có tác dụng lên file html được khai báo nhưng không ảnh hưởng đến các file khác trong cùng một website.

**<style>**

**h1, h2, h3 {font-size: 14px;}**

**p {color: blue;}**

**</style>**

* CSS ngoại tuyến:
* Đây là cách khai báo CSS được sử dụng nhiều nhất trong thực tế. và cách khai báo này mới tận dụng được hết thế mạnh mà CSS mang lại, tách biệt hoàn toàn khỏi tài liệu [HTML](http://cunglamseo.com/html-la-gi.htm), người thiết kế chỉ cần viết một file CSS duy nhất mà sử dụng nhiều lần trong ứng dụng của họ.
* Nó có phạm vi ảnh hưởng toàn bộ đến toàn bộ website chứ không chỉ một file .html riêng biệt. Do đó mỗi khi muốn thay đổi thuộc tính hiển thị của một thành phần nào đó trong website thì chỉ cần chỉnh sửa file CSS này thay vì phải chỉnh sửa tất cả các file .html trong ứng dụng. Đến đây chúng ta thấy được sự cần thiết và hiệu quả mà CSS đem lại.
* Để khai báo css ngoại tuyến, ta chỉ cần tạo một file style.css chẳng hạn, sau đó chèn đoạn code sau trong phần <head></head> của trang web.

**<link href="style.css" rel="stylesheet" type="text/css" media="screen" />**

Thứ tự ưu tiên trong CSS: CSS cục bộ » CSS nội tuyến » CSS ngoại tuyến » CSS mặc định của trình duyệt.

Chúng ta cần phân biệt rõ giữa CSS và Style trong HTML. Nếu như Style được dùng để định dạng nội dung trong HTML thì CSS là một file chứa Style.

### 1.3.3. BOOTSTRAP

**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.

Đây là cấu trúc file html (hoặc htm) được chứa trong folder bootstrap (bạn có thể đặt tên folder này là tùy ý) theo đường dẫn.

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="utf-8">**

**<meta http-equiv="X-UA-Compatible" content="IE=edge">**

**<meta name="viewport" content="width=device-width, initial-scale=1">**

**<title>Sài thử Bootstrap</title>**

**<!-- Khai báo sử dụng css của Bootstrap -->**

**<link href="css/bootstrap.min.css" rel="stylesheet">**

**<!-- Sửa lỗi HTML5 cho IE 8 trở xuống -->**

**<!--[if lt IE 9]>**

**<script src="**[**https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js**](https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js)**"></script>**

**<script src="**[**https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js**](https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js)**"></script>**

**<![endif]-->**

**</head>**

**<body>**

**<div class="container">**

**<h1>Hellow Bootstrap!</h1>**

**</div>**

**<!-- Khai báo thư viện jQuery -->**

**<script src="**[**https://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js**](https://ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js)**"></script>**

**<!-- Khai báo sử dụng thư viện javascript của bootstrap -->**

**<script src="js/bootstrap.min.js"></script>**

**</body>**

**</html>**

Lý do lại sử dụng nó vì:

* Được viết bởi những người có óc thẩm mỹ và tài năng trên khắp thế giới. Sự tương thích của trình duyệt với thiết bị đã được kiểm tra nhiều lần nên có thể tin tưởng kết quả mình làm ra và nhiều khi không cần kiểm tra lại. Vì vậy, giúp cho dự án của bạn tiết kiệm được thời gian và tiền bạc.
* Chỉ cần biết sơ qua HTML, CSS, Javascript, Jquery là bạn có thể sử dụng Bootstrap để tạo nên một trang web sang trọng và đầy đủ. Nhưng lại không cần code quá nhiều CSS.
* Với giao diện mặc định là màu xám bạc sang trọng, hỗ trợ các component thông dụng mà các website hiện nay cần có. Vì nó là opensource nên bạn có thể vào mã nguồn của nó để thay đổi theo ý thích của bản thân.
* Do có sử dụng Grid System nên Bootstrap mặc định hỗ trợ Responsive. Bootstrap được viết theo xu hướng Mobile First tức là ưu tiên giao diện trên Mobile trước. Nên việc sử dụng Bootstrap cho website của bạn sẽ phù hợp với tất cả kích thước màn hình. Nhờ đó mà chúng ta không cần xây dựng thêm một trang web riêng biệt cho mobile.

### 1.3.4. JAVASCRIPT

**JavaScript**, theo phiên bản hiện hành, là một ngôn ngữ lập trình kịch bản dựa trên đối tượng được phát triển từ các ý niệm nguyên mẫu. Ngôn ngữ này được dùng rộng rãi cho các trang web, nhưng cũng được dùng để tạo khả năng viết script sử dụng các đối tượng nằm sẵn trong các ứng dụng. Nó vốn được phát triển bởi Brendan Eich tại hãng truyền thông Netscape với cái tên đầu tiên Mocha, rồi sau đó đổi tên thành LiveScript, và cuối cùng thành JavaScript. Giống Java, JavaScript có cú pháp tương tự C, nhưng nó gần với Self hơn Java. .js là phần mở rộng thường được dùng cho tập tin mã nguồn JavaScript.

Cùng thời điểm Netscape bắt đầu sử dụng công nghệ Java trên trình duyệt Netscape, LiveScript đã được đổi tên thành JavaScript để được chú ý hơn bởi ngôn ngữ lập trình Java lúc đó đang được coi là một hiện tượng. JavaScript được bổ sung vào trình duyệt Netscape bắt đầu từ phiên bản 2.0b3 của trình duyệt này vào tháng 12 năm 1995. Trên thực tế, JavaScript không được phát triển dựa từ Java. Do đó JavaScript chỉ dựa trên các cách đặt tên của Java. Java Script gồm 2 mảng là client-server thực hiện lệnh trên máy của end-user và web-server.

Sau thành công của JavaScript, Microsoft bắt đầu phát triển JScript, một ngôn ngữ có cùng ứng dụng và tương thích với JavaScript. JScript được bổ sung vào trình duyệt Internet Explorer bắt đầu từ Internet Explorer phiên bản 3.0 được phát hành tháng 8 năm 1996.

DOM (Document Object Model), một khái niệm thường được nhắc đến với JavaScript trên thực tế không phải là một phần của chuẩn ECMAScript, DOM là một chuẩn riêng biệt có liên quan chặt chẽ với XML.

JavaScript là một ngôn ngữ lập trình dựa trên nguyên mẫu với cú pháp phát triển từ C. Giống như C, JavaScript có khái niệm từ khóa, do đó, JavaScript gần như không thể được mở rộng.

Cũng giống như C, JavaScript không có bộ xử lý xuất/nhập (input/output) riêng. Trong khi C sử dụng thư viện xuất/nhập chuẩn, JavaScript dựa vào phần mềm ngôn ngữ được gắn vào để thực hiện xuất/nhập.

Trên trình duyệt, rất nhiều trang web sử dụng JavaScript để thiết kế trang web động và một số hiệu ứng hình ảnh thông qua DOM. JavaScript được dùng để thực hiện một số tác vụ không thể thực hiện được với chỉ HTML như kiểm tra thông tin nhập vào, tự động thay đổi hình ảnh, ... Ở Việt Nam, JavaScript còn được ứng dụng để làm bộ gõ tiếng Việt giống như bộ gõ hiện đang sử dụng trên trang Wikipedia tiếng Việt. Tuy nhiên, mỗi trình duyệt áp dụng JavaScript khác nhau và không tuân theo chuẩn W3C DOM, do đó trong rất nhiều trường hợp lập trình viên phải viết nhiều phiên bản của cùng một đoạn mã nguồn để có thể hoạt động trên nhiều trình duyệt. Một số công nghệ nổi bật dòng JavaScript để tương tác với DOM bao gồm DHTML, Ajax và SPA.

Trong đó nổi bật là Ajax, AJAX, là viết tắt của **A**synchronous **Ja**vaScript and **X**ML, (tạm dịch là JavaScript và XML không đồng bộ), là một kỹ thuật mới để tạo các ứng dụng web giàu tính tương tác, nhanh hơn và mượt mà hơn với sự giúp đỡ của XML, HTML, CSS và JavaScript. AJAX cho phép các trang web được cập nhật một cách không đồng bộ bằng cách trao đổi các lượng dữ liệu nhỏ với Server. Tức là AJAX giúp cập nhật các phần nhỏ trong trang mà không cần tải lại toàn bộ trang. AJAX là một kỹ thuật phát triển Web độc lập với các phần mềm Web Server.

Với AJAX, người dùng có thể tiếp tục sử dụng ứng dụng trong khi các chương trình trên Client tạo Request để lấy thông tin từ Server.

AJAX được ông Jesse James Garrett đưa ra và dùng lần đầu tiên vào 2/2005 để chỉ kỹ thuật này. AJAX là một ứng dụng giàu tính Internet và bên cạnh đó AJAX cũng không thể làm việc một cách độc lập, kỹ thuật mới này sử dụng kết hợp:

* HTML (hoặc XHTML) với CSS trong việc hiển thị thông tin.
* Sử dụng đối tượng XMLHttpRequest để nhận dữ liệu từ một Web Server một cách không đồng bộ.
* Sử dụng mô hình DOM với sự trợ giúp của JavaScript để hiển thị/sử dụng dữ liệu.

Mỗi ứng dụng này đều cung cấp mô hình đối tượng riêng cho phép tương tác với môi trường chủ, với phần lõi là ngôn ngữ lập trình JavaScript gần như giống nhau.

### 1.3.5.JQUERY

**JQuery** là một thư viện kiểu mới của JavaScript, được tạo bởi John Resig vào năm 2006 với một phương châm tuyệt vời: Write less, do more - Viết ít hơn, làm nhiều hơn. **JQuery** làm đơn giản hóa việc truyền tải HTML, xử lý sự kiện, tạo hiệu ứng động và tương tác Ajax.

JQuery là một bộ công cụ tiện ích JavaScript làm đơn giản hóa các tác vụ đa dạng với việc viết ít code hơn. Dưới đây liệt kê một số tính năng tối quan trọng được hỗ trợ bởi jQuery:

* **Thao tác DOM −** jQuery giúp dễ dàng lựa chọn các phần tử DOM để *traverse (duyệt)* một cách dễ dàng như sử dụng CSS, và chỉnh sửa nội dung của chúng bởi sử dụng phương tiện Selector mã nguồn mở, mà được gọi là **Sizzle**.
* **Xử lý sự kiện −** jQuery giúp tương tác với người dùng tốt hơn bằng việc xử lý các sự kiện đa dạng mà không làm cho HTML code rối tung lên với các Event Handler.
* **Hỗ trợ AJAX −** jQuery giúp bạn rất nhiều để phát triển một site giàu tính năng và phản hồi tốt bởi sử dụng công nghệ AJAX.
* **Hiệu ứng −** jQuery đi kèm với rất nhiều các hiệu ứng đa dạng và đẹp mắt mà bạn có thể sử dụng trong các Website của mình.
* **Gọn nhẹ −** jQuery là thư viện gọn nhẹ - nó chỉ có kích cỡ khoảng 19KB (gzipped).
* **Được hỗ trợ hầu hết bởi các trình duyệt hiện đại −** jQuery được hỗ trợ hầu hết bởi các trình duyệt hiện đại, và làm việc tốt trên IE 6.0+, FF 2.0+, Safari 3.0+, Chrome và Opera 9.0+
* **Cập nhật và hỗ trợ các công nghệ mới nhất −** jQuery hỗ trợ CSS3 Selector và cú pháp XPath cơ bản.

Có hai cách để sử dụng jQuery:

* **Cài đặt nội bộ −** Bạn có thể tải jQuery Library trên thiết bị nội bộ của bạn và include nó trong HTML code.

Ví dụ

Bây giờ bạn có thể include thư viện *jquery* vào trong HTML file của bạn như sau:

**<html>**

**<head>**

**<title>The jQuery Example</title>**

**<script type="text/javascript" src="../jquery/jquery-2.1.3.min.js"></script>**

**<script type="text/javascript">**

**$(document).ready(function(){**

**document.write("Hello, World!");**

**});**

**</script>**

**</head>**

**<body>**

**<h1>Hello</h1>**

**</body>**

**</html>**

* **Sử dụng từ CDN (CDN Based Version) −** Bạn có thể include thư viện jQuery vào trong HTML code một cách trực tiếp từ Content Delivery Network (CDN).

Ví dụ

Bây giờ chúng ta viết lại ví dụ trên bởi sử dụng thư viện jQuery từ Google CDN:

**<html>**

**<head>**

**<title>The jQuery Example</title>**

**<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">**

**</script>**

**<script type="text/javascript">**

**$(document).ready(function(){**

**document.write("Hello, World!");**

**});**

**</script>**

**</head>**

**<body>**

**<h1>Hello</h1>**

**</body>**

**</html>**

Cũng tương tự như JavaScript, trước khi chúng ta sử dụng các đoạn code của jQuery để đọc hay chỉnh sửa các đối tượng DOM, chúng ta cần đảm bảo rằng chúng ta bắt đầu thêm các sự kiện ngay sau khi DOM sẵn sàng.

Nếu bạn muốn một sự kiện làm việc trên trang của bạn, bạn nên gọi nó bên trong Hàm $(document).ready(). Mọi thứ bên trong sẽ tải ngay sau khi DOM được tải và trước khi nội dung trang được tải.

Để làm điều này, chúng ta đăng ký một sự kiện đã sẵn sàng cho tài liệu như sau:

**$(document).ready(function() {**

**// do stuff when DOM is ready**

**});**

### 1.3.6. Visual Studio 2019:

Visual Studio 2019

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.

Visual Studio sở hữu nhiều tính năng hay:

1. Đa nền tảng:

**Phần mềm lập trình Visual Studio** của Microsoft hỗ trợ sử dụng trên nhiều nền tảng khác nhau. Không giống như các trình viết code khác, Visual Studio sử dụng được trên cả Windows, Linux và Mac Systems. Điều này cực kỳ tiện lợi cho lập trình viên trong quá trình ứng dụng.

1. Đa ngôn ngữ lập trình:

Không chỉ hỗ trợ đa nền tảng, Visual Studio cũng cho phép sử dụng nhiều ngôn ngữ lập trình khác nhau từ C#, F#, C/C++, HTML, CSS, Visual Basic, JavaScript, … Bởi vậy, Visual Studio có thể dễ dàng phát hiện và thông báo cho bạn khi các chương trình có lỗi.

1. Hỗ trợ website:

Visual Studio code cũng hỗ trợ website, đặc biệt trong công việc soạn thảo và [thiết kế web](https://mona.solutions/thiet-ke-website).

1. Kho tiện ích mở rộng phong phú:

Mặc dù Visual Studio có hệ thống các ngôn ngữ hỗ trợ lập trình khá đa dạng. Nhưng nếu lập trình viên muốn sử dụng một ngôn ngữ khác, bạn có thể dễ dàng tải xuống các tiện ích mở rộng. Tính năng hấp dẫn này được hoạt động như một phần chương trình độc lập nên không lo làm giảm hiệu năng của phần mềm.

1. Lưu trữ phân cấp:

Phần lớn các tệp dữ liệu đoạn mã của Visual Studio đều được đặt trong các thư mục tương tự nhau. Đồng thời, Visual Studio cũng cung cấp một số thư một cho các tệp đặc biệt để bạn lưu trữ an toàn, dễ tìm, dễ sử dụng hơn.

1. Kho lưu trữ an toàn:

Với Visual Studio, bạn có thể hoàn toàn yên tâm về tính lưu trữ, bởi phần mềm đã được kết nối GIT và một số kho lưu trữ an toàn được sử dụng phổ biến hiện nay.

1. Màn hình đa nhiệm:

Visual Studio sở hữu tính năng màn hình đa nhiệm, cho phép người dùng mở cùng lúc nhiều tập tin, thư mục dù chúng có thể không liên quan tới nhau.

1. Hỗ trợ viết code:

Khi sử dụng code vào trong lập trình, với Visual Studio, công cụ này có thể đề xuất tới các lập trình viên một số tùy chọn thay thế nhằm điều chỉnh đôi chút để đoạn code áp dụng thuận tiện hơn cho người dùng.

1. Hỗ trợ thiếp bị đầu cuối:

Phần mềm Visual Studio cũng tích hợp các loại thiết bị đầu cuối, giúp người dùng không cần chuyển đổi giữa hai màn hình hay trở về thư mục gốc khi thực hiện một thao tác cần thiết nào đó.

1. Hỗ trợ Git:

Do kết nối với GitHub nên Visual Studio cho phép hỗ trợ sao chép, kéo thả trực tiếp. Các mã code này sau đó cũng có thể thay đổi và lưu lại trên phần mềm.

1. Intellisense:

Tính năng nhắc Intellisense được sử dụng hầu hết trong các phần mềm lập trình, bao gồm cả Visual Studio. Tuy nhiên, so với các trình viết mã, Visual Studio vẫn được đánh giá cao về tính chuyên nghiệp. Đặc biệt, tính năng này còn có thể phát hiện tất cả các đoạn mã không đầy đủ, nhắc lập trình viên, gợi ý sửa đổi, khai báo biến tự động trong trường hợp lập trình viên quên, giúp bổ sung cú pháp còn thiếu, …

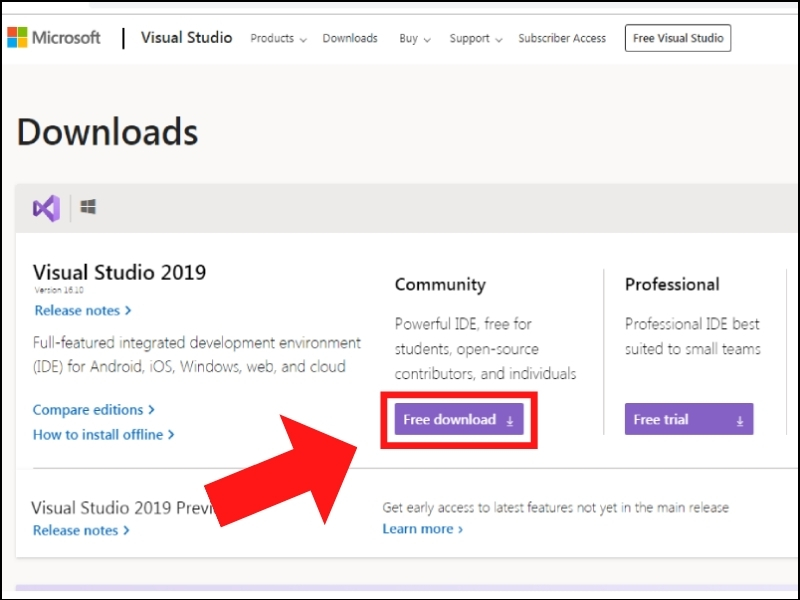
1. Tính năng comment:

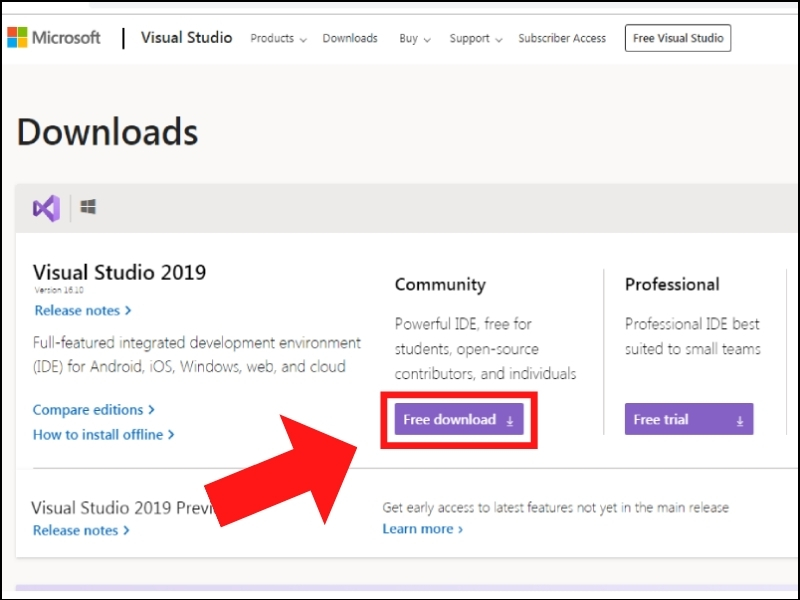
Một tính năng cũng khá hay ho, hỗ trợ cho người lập trình trong trường hợp “nhớ nhớ quên quên” đó là tính năng bình luận. Tính năng này cho phép lập trình viên để lại nhận xét, giúp dễ dàng ghi nhớ công việc cần hoàn thành, không bỏ sót công đoạn nào.

Ưu điểm khi sử dụng Visual Studio 2019:

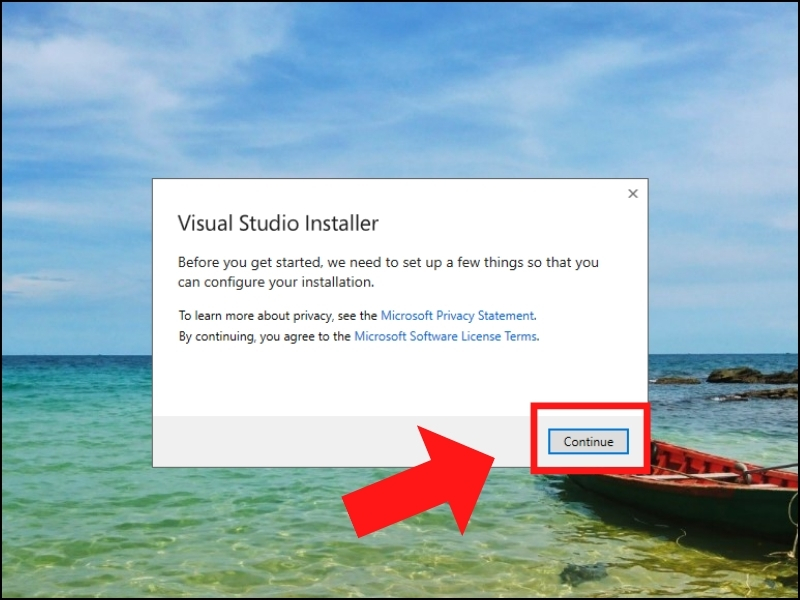
* Ít dung lượng
* Tính năng phong phú, đa dạng và mạnh mẽ
* Hỗ trợ nhiều nền tảng khác nhau (Linux, Windows, Mac)
* Hỗ trợ đa ngôn ngữ (HTML, CSS, C#, F#, C/C++, JSON, JavaScript, …)
* Tính năng Intellisense chuyên nghiệp
* Giao diện thân thiện dễ sử dụng
* Cập nhật đa dạng phiên bản cho nhiều đối tượng sử dụng khác nhau, …

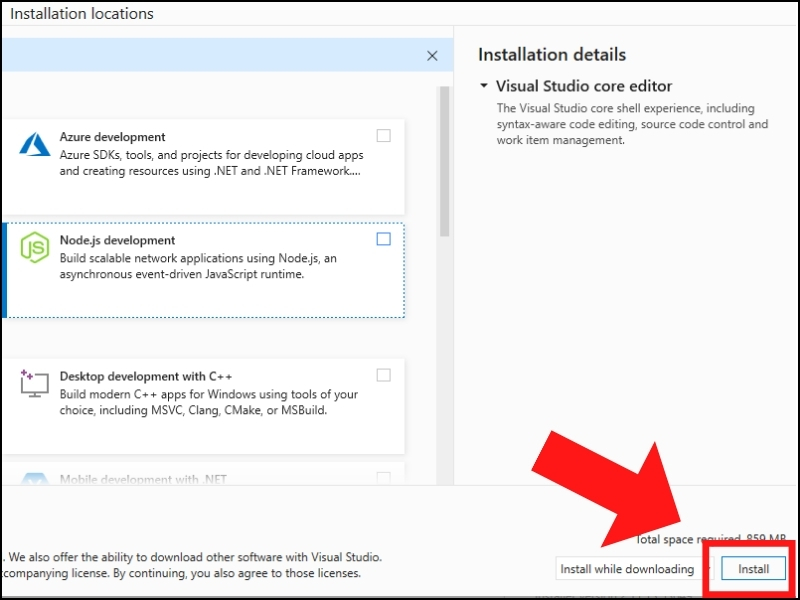
**\*Cách cài đặt:**

Vào trang web của Microsoft để cài đặt phần mềm

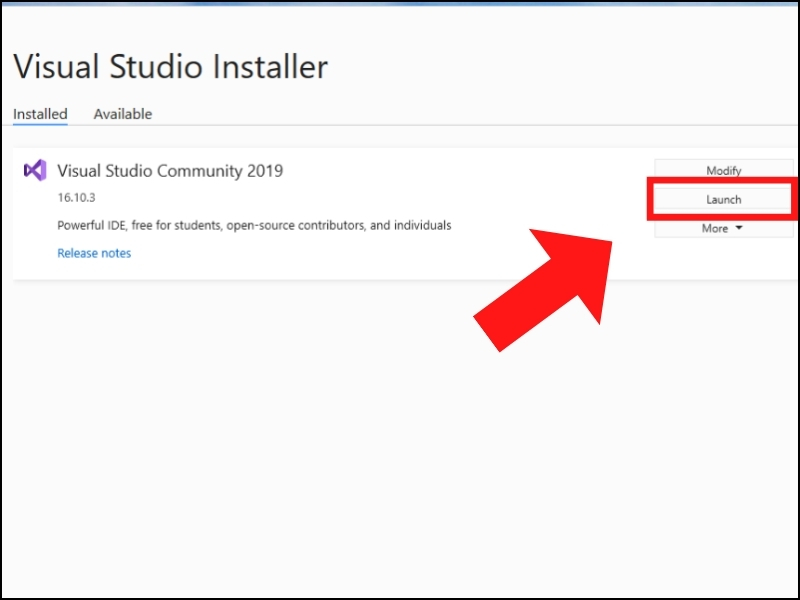
Cách cài đặt Visual Studio

Sau khi hoàn tất quá trình tải về, ấn vào bản vừa tải xuống -> chọn Continue

Cách cài đặt Visual Studio

 Chọn tính năng mà bạn muốn cài (.NET Framework, Visual C++, C++ for Linux, .NET Core, …) -> nhấn Install

Cách cài đặt Visual Studio

Chờ tiến trình hoàn tất nhấn Launch

Cách cài đặt Visual Studio

### 1.3.7. NuxtJS

Đây là một framework của Vuejs được sử dụng để xây dựng những ứng dụng từ Vuejs. Nó có thể dùng để xây dựng một static landing pages hay thậm chí là những ứng dựng phức tạp. Vì vậy trước khi học nuxt.js các bạn nhớ phải tìm hiểu về vuejs trước.

1. Một vài tính năng nổi bật trong Nuxt.JS

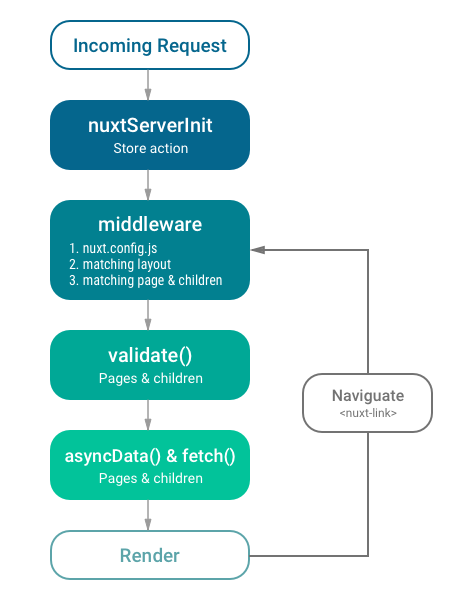
* Viết dưới dạng file vue (.vue)
* Tự động split code để load javascript nhanh hơn
* Được áp dụng server side rendering
* Hỗ trợ routing khá dễ dàng
* Hỗ trợ ES6/ES7
* Đóng gói và nén javascript, css
* Quản lý các thẻ ở phần head

1. NuxtJS hoạt động như thế nào

Chúng hỗ trợ thư viện như:

* Vue2
* Vue-Router
* Vuex (được include chỉ khi bạn sử dụng store)
* Vue Server Renderer
* Vue-meta

1. Sơ đồ hoạt động:

Khi có request đến server hoặc khi user bấm vào link thông qua <nuxt-link> thì nuxt.js hoạt động theo sơ đồ sau:

Sơ đồ hoạt động của Nuxt.JS

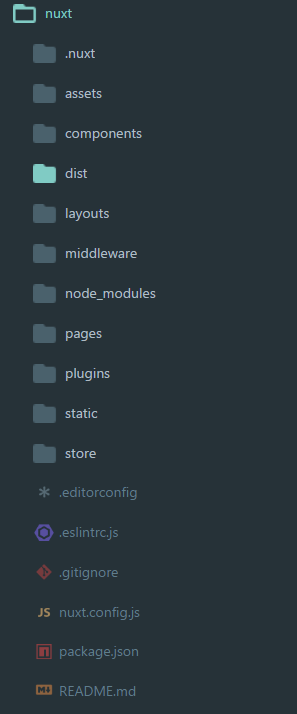
1. Cài đặt ứng dụng Nuxt.JS

Các bạn có thể cài đặt qua npx

npx create-nuxt-app <project-name>

hoặc qua yarn

yarn create nuxt-app <project-name>

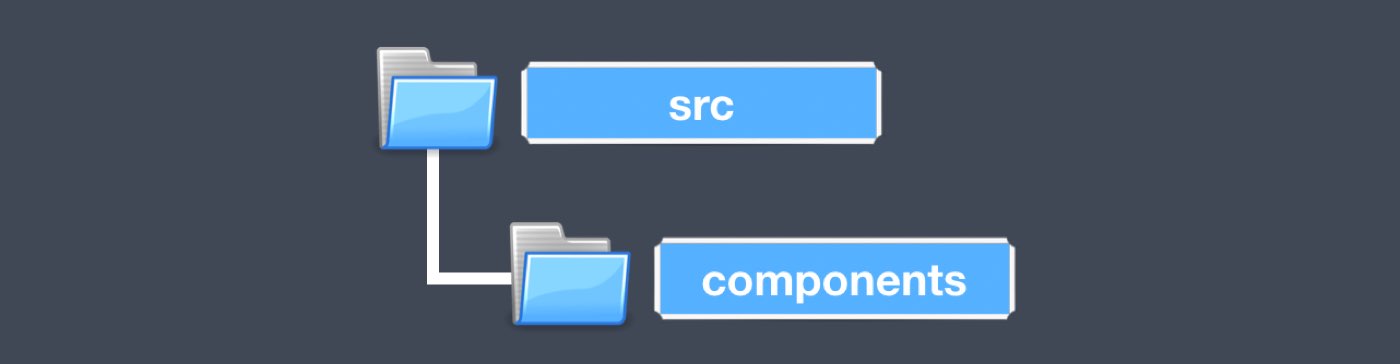
******

**Cấu trúc thư mục**

Sau khi cài xong thì chạy npm run dev và mở localhost:3000 lên để chạy ứng dụng NuxtJS.

1. Tìm hiểu về cấu trúc thư mục:

* Assets: chứa các tài nguyên chưa được biên dịch như Stylus, Sass, image, font.
* Components: chứa các componens của bạn. Nếu 1 ứng dụng Vue.JS bình thường thì cấu trúc thư mục của component của bạn sẽ như này.



Cấu trúc thư mục của component

Chúng ta sẽ đặt hết các components trong này chứ? Nhưng đôi khi sẽ gây chút kiểm soát nhầm lẫn, ví dụ người dùng tạo 1 file để dành cho view, hay layouts, thì điều này trong nuxtjs đã phân chia khá rõ rang



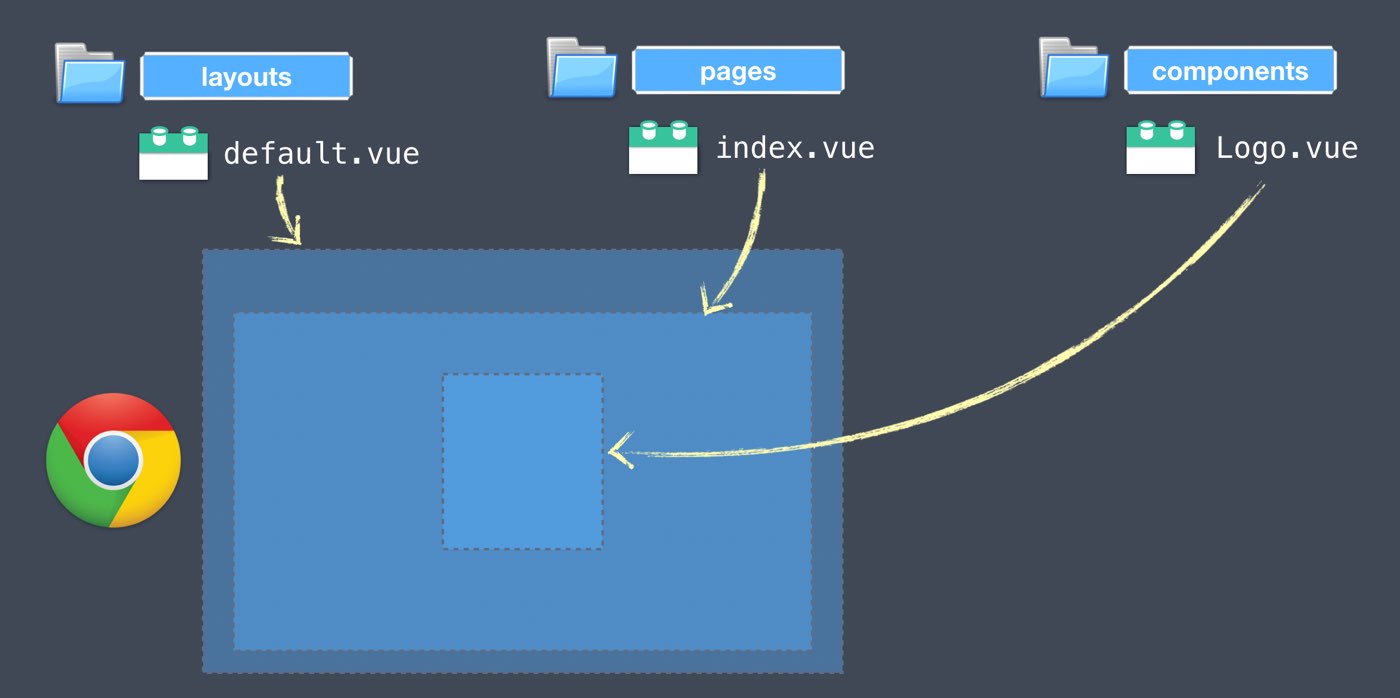
Cấu trúc thư mục của component

Đó như các bạn thấy thì mỗi folder sẽ chứa những file component riêng, mỗi bản thân chúng chứa những tác dụng khác nhau. Layouts: Chứa các giao diện cho ứng dụng

**Page**: Chứa các view ứng dụng và đồng thời đầy cũng là cấu trúc routes của chúng ta được sinh ra

**Component**: Chức các component có thể tái sử dụng.

Ba thư mục trên cấu hình tạo nên một trang để chúng ta nhìn thấy. Ví dụ như thế này.

Trang được cấu hình từ các component

**Middleware**: Chứa Middleware của ứng dụng, middleware cho phép bạn định nghĩa các function được chạy trước khi render 1 page hoặc nhóm page

**Plugins**: Chứa các javascript plugin

**Static**: Chứa các file tĩnh như các file ảnh chẳng hạn, được map tự động, ví dụ file /static/logo.png sẽ là yoursite/logo.png

**Store**: Chứa các file vuex store

**File nuxt.config.js**: Chứa các cấu hình được thiết đặt cho ứng dụng của bạn

**File package.json**: Chứa các dependencies và scripts

1. Đôi điều về routing:

Nuxtjs sẽ tự động sinh ra route theo định nghĩa cấu trúc file trong thư mục pages.

*f.1) Basic route*:

Ví dụ chúng ta có một cấu trúc thư mục như sau:

pages/

--| user/

-----| index.vue

-----| one.vue

--| index.vue

Chúng sẽ tự động generate ra một cấu trúc route như sau:

router: {

routes: [

{

name: 'index',

path: '/',

component: 'pages/index.vue'

},

{

name: 'user',

path: '/user',

component: 'pages/user/index.vue'

},

{

name: 'user-one',

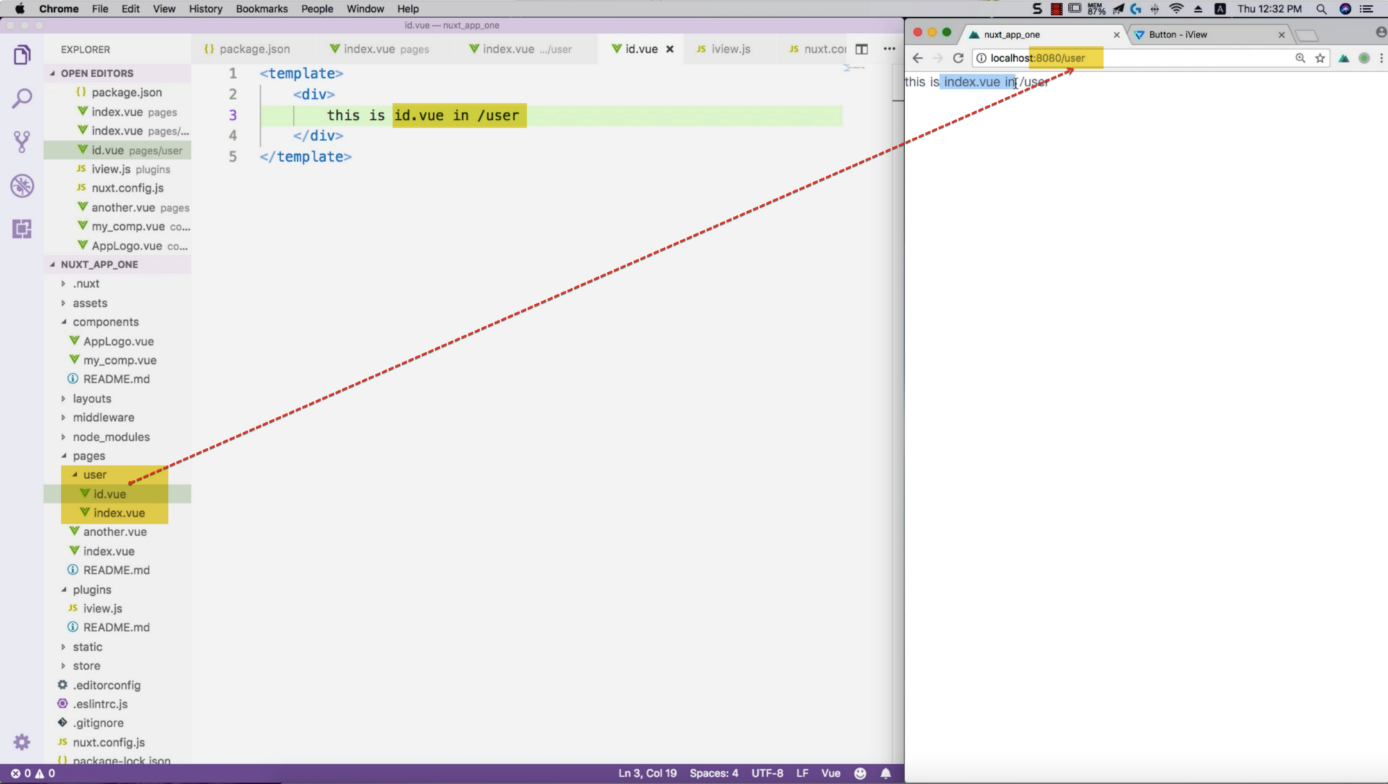
path: '/user/one',

component: 'pages/user/one.vue'

}

]

}

Quá nhanh đúng không nào, thay vì tự định nghĩa viết ra route thì thay vào đó chúng ta có thể tự sinh ra route bằng các tạo ra các folder vào các file. Đây là hình ảnh minh họa

Cách tự sinh ra route bằng cách folder

### 1.3.8. ASP.net Core:

a) ASP.net Core là gì?

- ASP.NET Core là một tập hợp các thư viện chuẩn như một framework để xây dựng ứng dụng web.

- ASP.NET Core không phải là phiên bản tiếp theo của ASP.NET. Nó là một cái tên mới được xây dựng từ đầu. Nó có một sự thay đổi lớn về kiến trúc và kết quả là nó gọn hơn, phân chia module tốt hơn.

- ASP.NET Core có thể chạy trên cả .NET Core hoặc full .NET Framework.

b) .Net Core là gì?

- .NET Core là môi trường thực thi. Nó được thiết kế lại hoàn toàn của .NET Framework. Mục tiêu chính của .NET Core là hỗ trợ phát triển ứng dụng đa nền tảng cho ứng dụng .NET. Nó được hỗ trợ trên Windows, Mac OS và Linux. .NET Core là một framework mã nguồn mở được xây dựng và phát triển bởi Microsoft và cộng đồng .NET trên Github

- NET Core là một tập con của Full .NET Framwork. WebForms, Windows Forms, WPF không phải là một phần của .NET Core.

- Nó cũng triển khai đặc điểm của .NET Standard.

c) .Net Standard là gì?

.NET Standard là một đặc tả chuẩn của .NET API hướng tới hỗ trợ trên tất cả các triển khai của nền tảng .NET. Nó định nghĩa một tập các quy tắc thống nhất cần thiết để hỗ trợ tất cả các ứng dụng trên nền .NET.

1. Sự khác nhau giữa .Net Core và .Net Framework:

.NET Framwork là môi trường cũ hơn và đã tồn tại trên Windows nhiều năm nay. .NET Core hỗ trợ các tính năng trong tập con của .NET Framwork. Các tính năng như WebForms, Windows Forms, WPF chưa được đưa lên .NET Core. .NET Framwork chỉ chạy trên Windows trong khi .NET Core có thể chạy trên bất cứ nền tảng nào.

1. Các đặc tính quan trọng của ASP.NET Core:

* Bạn có thể xây dựng và chạy ứng dụng ASP.NET đa nền tảng trên Windows, Mac và Linux (mã nguồn mở và cộng đồng phát triển)
* ASP.NET Core hợp nhất ASP.NET MVC và ASP.NET Web API.
* Có thể host trên IIS hoặc tự host.
* Có sẵn Dependency Injection.
* Dễ dàng tích hợp với các framework frontend như Angular, Knockout...
* Hỗ trợ cấu hình cho nhiều môi trường.
* Cơ chết HTTP Request pipeline mới.
* Hỗ trợ quản lý phiên bản
* Dùng chung toàn bộ Nuget Package.

1. Các nhánh của ASP.NET:

Có hai nhánh của ASP.NET cho đến hiện tại:

* ASP.NET

Đây là phiên bản hiện tại của ASP.NET và nó cần .NET Framwork để chạy, phiên bản hiện tại hoàn thiện là 4.6

* ASP.NET Core.

Là cách mới để xây dựng ứng dụng web. Nó có thể chạy cả trên .NET Framework và .NET Core. Phiên bản hiện tại đang là 2.2

1. Khác biệt quan trọng của ASP.NET và ASP.NET Core:

|  |  |
| --- | --- |
| ***ASP.NET*** | ***ASP.NET CORE*** |
| Phiên bản hiện tại 4.6 | Phiên bản hiện tại 2.2 |
| Nền tảng đã có từ lâu | Hoàn toàn được thiết kế mới |
| Chạy trên .NET Framework | Chạy trên cả .NET Core và .NET Framework |
| Chỉ trên Windows | Chạy trên tất cả các OS sử dụng .NET Core |
| Nền tảng ổn định với tính năng phong phú | Chưa hoàn chỉnh nhưng mong đợi sẽ hoàn chỉnh trong tương lai |
| WebForms được hỗ trợ | Không hỗ trợ WebForms |
| System.web.dll cồng kềnh | Nhỏ, nhẹ và module hóa |
| Bản quyền của Microsoft | ASP.NET Core là mã nguồn mở |

### 1.3.9. Visual Studio Code

1. Visual Studio Code là gì?

Đối với nhiều lập trình viên, trình soạn thảo code là một công cụ không thể thiếu trong quá trình xây dựng các ứng dụng web. Ưu điểm của chúng là dung lượng khá nhẹ, ít tốn RAM và được tích hợp các tính năng gần giống với IDE. Trong đó, Visual Studio Code là cái tên quen thuộc nhất với dân IT hiện nay.

Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn mở gọn nhẹ nhưng có khả năng vận hành mạnh mẽ trên 3 nền tảng là Windows, Linux và macOS được phát triển bởi Microsoft. Nó hỗ trợ cho JavaScript, Node.js và TypeScript, cũng như cung cấp một hệ sinh thái mở rộng vô cùng phong phú cho nhiều ngôn ngữ lập trình khác.

Có thể nói, VS Code là sự kết hợp giữa một trình soạn thảo mã nguồn với những công cụ phát triển mạnh mẽ như Git, Debug hay Syntax Highlighter.

1. Phân biệt Visual Studio Code và Visual Studio:

* Do tên gọi khá tương đồng nên có nhiều người nhầm lẫn giữa Visual Studio Code và Visual Code. Trên thực tế, chúng hoàn toàn khác nhau.
* VS Code là một trình soạn thảo văn bản. Visual Studio là một môi trường phát triển tích hợp.
* VS Code có thể chạy trên nhiều nền tảng. Visual Studio chỉ có thể chạy trên Windows hoặc Mac (đây là 2 sản phẩm khác nhau)
* Tốc độ của VS Code được đánh giá là nhanh hơn so với Visual Studio.
* Kích thước tải xuống của VS Code nhỏ hơn Visual Studio.

1. Lý do bạn nên sử dụng Visual Studio Code:

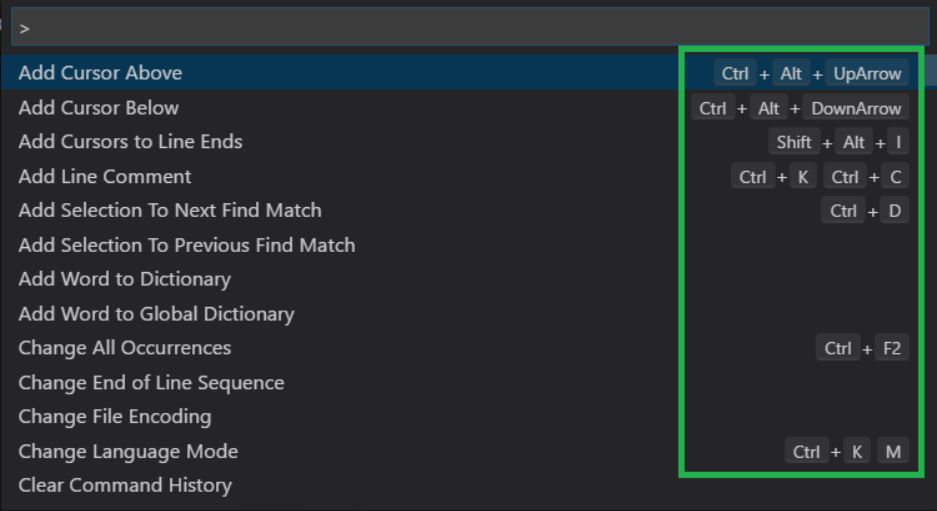
c.1. Hoạt động đa nền tảng:

VS Code được thiết kế để có thể chạy tốt trên macOS, Linux và Windows. Vì thế, bạn có thể thoải mái sử dụng trình soạn thảo này mà không cần lo lắng về vấn đề lựa chọn nền tảng tương thích.

c.2. Đơn giản trong việc chỉnh sửa, xây dựng và gỡ lỗi:

Như đã nói ở trên, tốc độ là một trong những ưu thế vượt trội của VS Code. Nó có thể hỗ trợ rất nhiều các loại ngôn ngữ lập trình khác nhau như Java, Python, C++, C#, Golang hay PHP. Điều này cho phép bạn có thể ứng dụng rất nhiều tính năng thú vị như đánh dấu cú pháp, tự động thụt lề, đối sánh dấu ngoặc, … cho công việc của mình.

Hệ thống phím tắt sinh động, trực quan giúp thao tác trở nên nhanh chóng hơn, từ đó năng suất và hiệu quả lập trình cũng được tăng lên. Ngoài ra, tính năng nhắc mã IntelliSense chuyên nghiệp trên VS Code sẽ tự động phát hiện đoạn mã không đầy đủ và thông báo cho lập trình viên. Hoặc nó có thể tự động bổ sung cú pháp còn thiếu khi bạn quên không khai báo biến.

Các phím tắt trên Visual Code

Gỡ lỗi cũng là một tính năng nổi bật của VS Code. Nó giúp bạn xem mã nguồn, kiểm tra các các biến, Call Stack và thực thi lệnh trong bảng điều khiển. Bên cạnh đó, trình soạn thảo này còn hỗ trợ Git phục vụ cho việc kiểm soát nguồn mà không đòi hỏi phải rời khỏi trình chỉnh sửa.

1. Cách cài đặt:

Để tải xuống và cài đặt Visual Studio Code, hãy làm theo các bước sau:

1. Tải file cài đặt Visual Studio Code cho Windows.

2. Lưu file trên thiết bị.

3. Nhấp đúp vào file để khởi chạy cài đặt Visual Studio Code trên Windows 10.

4. Xác nhận các điều khoản thỏa thuận.

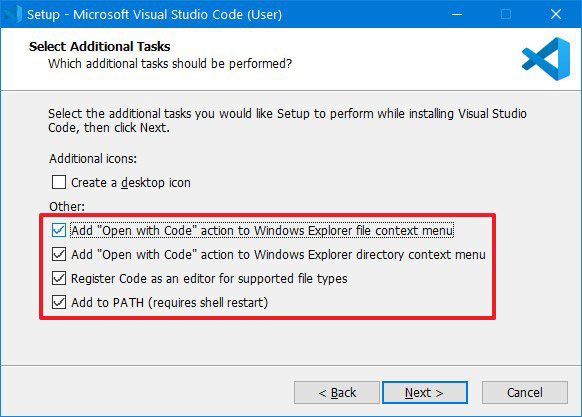
5. Nhấp vào nút Next.

6. Sử dụng vị trí cài đặt mặc định và nhấp vào nút Next.

7. Sử dụng cài đặt menu Start mặc định và nhấp vào nút Next.

8. (Tùy chọn) Tích vào tùy chọn Add “Open with code” action to Windows Explorer file context menu.

9. (Tùy chọn) Tích vào tùy chọn Add “Open with code” action to Windows Explorer directory context menu.

Cách cài đặt Visual Studio Code

10. Tích vào tùy chọn Register Code as an editor for supported file types.

11. Kiểm tra tùy chọn Add to PATH.

12. Nhấp vào nút Install.

13. Nhấp vào nút Finish.

Sau khi hoàn thành các bước, bạn có thể bắt đầu sử dụng VSCode trên máy tính Windows 10 để đánh giá hoặc tạo code.

### 1.3.10. Vue JS

Khi nói về các Javascript framework, chúng ta thường cho rằng các framework cần cung cấp mọi thứ ta cần để xây dựng một SPA (Single Page Application) hoàn thiện (gọi là Full framework). Tuy nhiên, một framework được thiết kế theo cách đó thường tự mua vào mình sự nặng nề, và bất kỳ sự hiểu biết hay kinh nghiệm cụ thể về các framework đó không mang lại hữu ích khi ta xây dựng bên ngoài kịch bản SPA. Các framework này cũng cung cấp quá mức cần thiết trong các trường hợp sự dụng tương đối đơn giản của chúng ta.

1. Progressive framework:

Progressive framework là khái niệm khá mới, có thể hiểu thay vì như Full-featured framework hay Monolithic framework cung cấp tất cả mọi thứ cần có để xây dựng app trong một framework duy nhất, thì progresive framework lại chia thành các thành phần nhỏ khác nhau, và ta có thể dần dần lựa chọn các thành phần tham gia vào sao cho phù hợp. Vue.js là một trong số các progressive framework. Phần lõi của Vue.js tập chung chủ yếu vào phần View. Chúng ta có thể sử dụng mỗi lõi của Vue để tạo view hoặc CÓ THỂ cài THÊM các thành phần như vuex để quản lý state trong app hay vue-router cho SPA routing... hoặc là không dùng thêm.

Phần lõi có thể hoạt động tốt một mình và có thể chạy mượt mà khi ghép thêm các thành khác lại với nhau. Đó cũng chính là một trong những đặc điểm của progressive framework.

1. Rendering:

Lõi của Vue.js là một hệ thống cho phép chúng ta render dữ liệu đến DOM bằng cú pháp template rõ ràng, đơn giản:

<div id="app">

{{ message }}

</div>

// Javascript:

var app = new Vue({

el: '#app',

data: {

message: 'Hello world!'

}

});

Nhìn qua ta thấy đơn thuần chỉ là hiển thị một string "Hello world!" kinh điển nhưng đằng sau những câu lệnh đơn giản trên Vue còn làm nhiều hơn thế! Dữ liệu và DOM đã được liên kết với nhau. Không tin bạn hãy mở console lên và sửa app. message = "Hello http://viblo.asia:D". Bạn sẽ thấy chuỗi được render cũng sẽ bị thay đổi theo thành Hello http://viblo.asia:D". Vue.js cũng sử dụng Virtual DOM, thông qua các event như click, hover... Virtual DOM sẽ tính toán các phần bị thay đổi rồi chỉ render lại các phần bị thay đổi đó. mà thôi. Ngoài việc chèn text bằng cú pháp như trên, chúng ta còn có thể bind qua attribute của element như ví dụ dưới đây:

<div id="app-2">

<span v-bind: title="message">

Loading >>>

</span>

</div>

var app2 = new Vue ({

el: '#app-2',

data: {

message: 'You loaded this page on ' + new Date ()

}

});

Đến đây chúng ta gặp một thứ rất mới mẻ, v-bind attribute (thuộc tính). Thuộc tính này được gọi là directive, các directive trong Vue đều có tiền tố là v-. Với v-bind, trong Vue còn có shortcut thay thế đó là: Tức là thay vì dùng v-bind: title ta có thể dung :title thì cũng sẽ cho ra kết quả tương tự.

1. Rẽ nhánh và vòng lặp:

c.1) Rẽ nhánh: v-if, v-else, v-else-if:

Một cách đơn giản để ẩn hiện một element thông qua việc rẽ nhánh: if-else trong vue, ví dụ:

<div id="app-3">

<p v-if="seen">Now you see me</p>

<p v-else>Where are you? </p>

</div>

var app3 = new Vue ({

el: '#app-3',

data: {

seen: true

}

});

Ta thấy chuỗi "Where are you" sẽ hiện ra. Khi thay đổi giá trị của app .seen = false hoặc là app.seen = true qua console thì chúng ta sẽ thấy chuỗi "Now you see me" sẽ ẩn hoặc là hiện thông qua directive v-if và v-else. Directive v-if sẽ hiển thị element nếu thõa màn điều kiện seen hoặc nếu không sẽ hiển thị element có v-else. Và v-else-if tương đương mệnh đề "Nếu không thì". Directive này không có shorcut như v-bind được đề cập ở trên.

d) Vòng lặp –v-for:

Ta có thể duyệt một mảng todos thông qua directive v-for, tại mỗi lần duyệt, ta lấy được một phần tử todo trong mảng:

<div id="app-4">

<ol>

<li v-for="todo in todos">

{{ todo.text }}

</li>

</ol>

</div>

var app4 = new Vue({

el: '#app-4',

data: {

todos: [

{ text: 'Learn JavaScript' },

{ text: 'Learn Vue' },

{ text: 'Build something awesome' }

]

}

})

e) Xử lý input của người dung:

Để người dùng tương tác với app, chúng ta sử dụng directive v-on để bắt các event listener vào DOM, các listener này sẽ gọi các method trong đối tượng Vue của chúng ta.

<div id="app-5">

<p>{{ message }}</p>

<button v-on:click="reverseMessage">Reverse Message</button>

</div>

var app5 = new Vue({

el: '#app-5',

data: {

message: 'Hello Vue.js!'

},

methods: {

reverseMessage: function () {

this.message = this.message.split('').reverse().join('')

}

}

})

Kết quả của ví dụ trên là ta có được một button, khi click vào button chuỗi Hello Vue.js sẽ bị đảo ngược bởi thông qua v-on, ứng dụng đã gọi method reverseMessage trong methods. Hàm này thực hiện xử lý dữ liệu message và lưu lại. Khi biến message bị thay đổi, DOM tự thay đổi lại hiển thị message bằng giá trị mới. Tưởng tự, ta có thể đặt thêm nhiều method xử lý dữ liệu khác vào trong methods khi khởi tạo instance của Vue.

f) Mở rộng và tái xử lý các component:

Kết quả của ví dụ trên là ta có được một button, khi click vào button chuỗi Hello Vue.js sẽ bị đảo ngược bởi thông qua v-on, ứng dụng đã gọi method reverseMessage trong methods. Hàm này thực hiện xử lý dữ liệu message và lưu lại. Khi biến message bị thay đổi, DOM tự thay đổi lại hiển thị message bằng giá trị mới. Tưởng tự, ta có thể đặt thêm nhiều method xử lý dữ liệu khác vào trong methods khi khởi tạo instance của Vue.

// Define a new component called todo-item:

Vue.component('todo-item', {

template: '<li>This is a todo</li>'

});

// Or

Vue.component('todo-item', require("./path/to/components/TodoItem.vue"));

Vue.js là một công nghệ mới, một Progressive framework nhỏ nhẹ, 18kb min+gzip, kết hợp với việc sử dụng Virtual DOM chỉ render lại các thành phần mà có sự thay đổi giúp tăng hiệu suất của Vue. Thêm vào là cú pháp template rõ ràng, đơn giản, dễ tiếp cận. Một hệ thống component cho phép tái sử dụng ở các template khác nhau.

g) Cách cài đặt VueJS:

Có nhiều cách cài đặt VueJS:

g.1) CDN (Content delivery Network)

Đây chắc chắn là cách nhanh gọn nhất, khi bạn chỉ cần thêm 1 đường link vào là xong.

<script src="https://cdn.jsdelivr.net/npm/vue@2.5.16/dist/vue.js"></script>

\*CDN là dữ liệu sẽ được đưa lên các server trên toàn cầu, và tùy thuộc vào vị trí của bạn sẽ lựa chọn server gần nhất để cung cấp dữ liệu, giúp tốc độ tải tốt nhất.

g.2) Download:

Đây là cách truyền thống nhất, bạn lên trang chủ và tải file .js về rồi thêm đường dẫn tới file js đó.

<script src="your-app/your-folder/vue.js"></script>

g.3) NPM (Node Package Manager)

Đây là công cụ tạo, quản lí các gói thư viện Javascript, tức là sẽ quả lí tự động, loại bỏ các thao tác thủ công như hai cách trên. Chúng ta chỉ cần khai báo những thư viện cần sử dụng vào file package.json, npm sẽ sự động tải chúng về giúp bạn mà không lo thiếu sót.

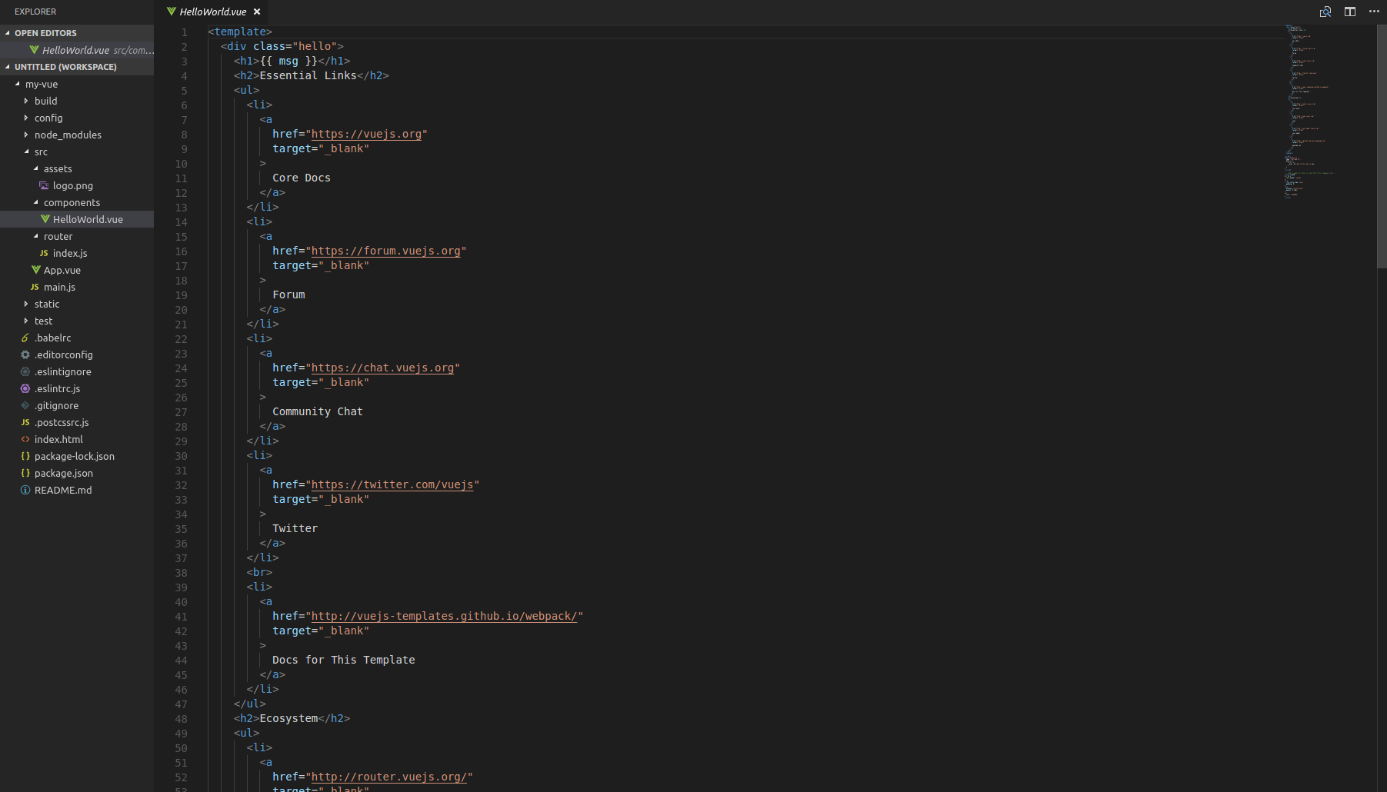
* Cài đặt Node kèm Npm
* Cài đặt vue bằng npm:
* Sử dụng câu lệnh $ npm install vue
* Khi đã làm quen, bạn nên cài đặt thêm Vue-CLI (Command Line Interface) giúp nhanh chóng khởi tạo nền tảng cho các ứng dụng.

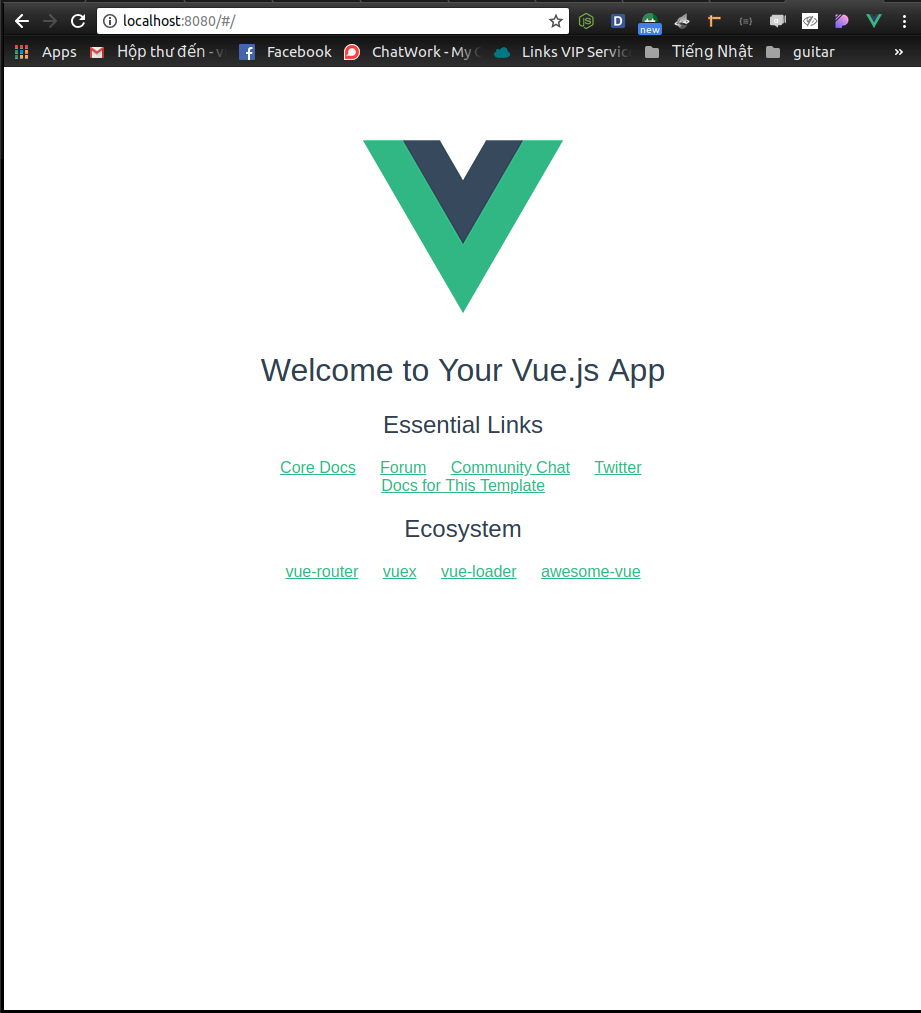
$ npm install --global vue-cli #cài đặt vue-cli

$ vue init webpack my-vue # tạo một dự án mới với template "webpack"

$ cd my-vue

$ npm run dev #chạy server

**Cấu trúc thư mục project



Hình ảnh khi chạy (cổng mặc định là 8080)

## 1.4.Hệ Quản Trị Cơ Sở Dữ Liệu

**MICROSOFT SQL SERVER**  là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) ) sử dụng câu lệnh SQL (**Transact-SQL)**để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

**MICROSOFT SQL SERVER**  được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user.

**MICROSOFT SQL SERVER** là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

**MICROSOFT SQL SERVER**  là cơ sở dữ liệu có trình giao diện trên Windows, cho phép người sử dụng có thể thao tác các hành động liên quan đến nó**.** Việc tìm hiểu từng công nghệ trước khi bắt tay vào việc viết mã kịch bản ASP.NET MVC5, việc tích hợp hai công nghệ ASP.NET MVC5 và MICROSOFT SQL SERVER  là một công việc cần thiết và rất quan trọng.

### 1.4.1. Mục Đích Sử Dụng Cơ Sở Dữ Liệu

**Chức năng của MICROSOFT SQL SERVER**

* Chúng ta có thể truy vấn Database theo nhiều cách khác nhau, sử dụng bởi các lệnh.
* Người dùng có thể truy cập dữ liệu từ RDBMS.
* Cho phép người dùng miêu tả dữ liệu.
* Cho phép người dùng định nghĩa dữ liệu trong một Database và thao tác nó
* Cho phép người dùng tạo, xóa Database và bảng.
* Cho phép người dùng tạo view, Procedure, hàm trong một Database.
* Cho phép người dùng thiết lập quyền truy cập vào bảng, thủ tục và view.

### 1.4.2. Các kiểu dữ liệu thông dụng trong cơ sở dữ liệu MICROSOFT SQL SERVER

#### 1.4.2.1. Loại dữ liệu numeric

|  |  |  |  |
| --- | --- | --- | --- |
| * *Kiểu dữ liệu số nguyên* | | | |
| **Loại** | **Range** | **Bytes** | **Diễn giải** |
| Tinyint | -127->128 hay 0..255 | 1 | Số nguyên rất nhỏ. |
| Smallint | -32768 ->32767 hay 0..65535 | 2 | Số nguyên nhỏ. |
| Mediumint | -32768 ->32767 hay 0..65535 | 3 | Số nguyên vừa |
| Int | -263 ->263-1 hay 0..264-1 | 4 | Số nguyên lớn. |
| Bigint | -263->263-1 hay 0..264-1 | 8 | Số nguyên lớn. |

Bảng 1.1. Kiểu dữ liệu số nguyên

* *Kiểu dữ liệu số chấm động*

|  |  |  |  |
| --- | --- | --- | --- |
| **Loại** | **Range** | **Bytes** | **Diễn giải** |
| Float | phụ thuộc số thập phân |  | Số thập phân dạng Single hay Double |
| Float(M,D) | ±1.175494351E-38 ±3.40282346638 | 4 | Số thập phân dạng Single |
| Double(M,D) | ±1.7976931348623157308 ±2.2250738585072014E-308 | 8 | Số thập phân dạng Double. |
| Float[M,[D]) |  |  | Số chấm động lưu dưới dạng char. |

Bảng 1.2. Kiểu dữ liệu số chấm động

#### 1.4.2.2. Loại dữ liệu kiểu Date and Time

Kiểu dữ liệu Date and Time cho phép bạn nhập dữ liệu dưới dạng chuỗi ngày tháng hay dạng số.

* *Dữ liệu kiểu số nguyên*

|  |  |  |
| --- | --- | --- |
| **Loại** | **Range** | **Diễn giải** |
| Date | 1000-01-01 | Date trình bày dưới dạng yyyy-mm-dd. |
| Time | -838:59:59 , 838:59:59 | Time trình bày dưới dạng hh:mm:ss. |
| Datetime | 1000-01-01 00:00:00,9999-12-31 23:59:59 | Date và Time trình bày dưới dạng yyyy-mm-dd hh:mm:ss. |
| TimeStamp[(M)] | 1970-01-01 00:00:00 | TimeStamp trình bày dưới dạng yyyy-mm-dd hh:mm:ss. |
| Year[(2|4)] | 1970-2069 1901-2155 | Year trình bày dưới dạng 2 số hay 4 số. |

Bảng 1.3. Kiểu dữ liệu thời gian

|  |  |
| --- | --- |
| **Loại** | **Range** |
| Date | 1000-01-01 |
| Time | -838:59:59 , 838:59:59 |
| Datetime | 1000-01-01 00:00:00,9999-12-31 23:59:59 |
| TimeStamp[(M)] | 1970-01-01 00:00:00 |
| Year[(2|4)] | 1970-2069 1901-2155 |

Bảng 1.4. Kiểu dữ liệu thời gian

#### 1.4.2.3. Loại dữ liệu String

Kiểu dữ liệu String chia làm 3 loại: loại thứ nhất như char (chiều dài cố định) và varchar (chiều dài biến thiên); loại thứ hai là Text hay Blob, Text cho phép lưu chuỗi rất lớn, Blob cho phép lưu đối tượng nhị phân; loại thứ ba là Enum và Set.

* *Kiểu dữ liệu String*

|  |  |  |
| --- | --- | --- |
| **Loại** | **Range** | **Diễn giải** |
| Char | 1-255 characters | Chiều dài của chuỗi lớn nhất 255 kí tự. |
| Varchar | 1-255 characters | Chiều dài của chuỗi lớn nhất 255 kí tự(characters). |
| Tinyblob | 28-1 | Khai báo cho Field chứa kiểu đối tượng nhị phân cỡ 255 kí tự. |
| Tinytext | 28-1 | Khai báo cho Field chứa kiểu chuỗi cỡ 255 characters. |
| Blob | 216-1 | Khai báo cho Field chứa kiểu blob cỡ 65,535 characters. |
| Text | 216-1 | Khai báo cho Field chứa kiểu chuỗi dạng văn bản cỡ 65,535 characters. |
| Mediumblob | 224-1 | Khai báo cho Field chứa kiểu blob vừa khoảng 16,777,215 characters. |
| Mediumtext | 224-1 | Khai báo cho Field chứa kiểu chuỗi dạng văn bản vừa khoảng 16,777,215 characters. |
| Longblob | 232-1 | Khai báo cho Field chứa kiểu blob lớn khoảng 4,294,967,295 characters. |
| Longtext | 232-1 | Khai báo cho Field chứa kiểu chuỗi dạng văn bản lớn khoảng 4,294,967,295 characters. |

Bảng 1.5. Kiểu dữ liệu String

### 1.4.3. Các thao tác cập nhật dữ liệu

* SELECT (Truy vấn mẫu tin): Select dùng để truy vấn từ một hay nhiều bảng khác nhau, kết quả trả về là một tập mẫu tin thỏa mãn các điều kiện cho trước nếu có, cú pháp của phát biểu SQL dạng SELECT như sau:

SELECT<danh sách các cột>

[FROM<danh sách bảng>]

[WHERE<các điều kiện ràng buộc>]

[GROUP BY<tên cột/biểu thức trong SELECT>]

[HAVING<điều kiện bắt buộc của GROUP BY>]

[ORDER BY<danh sách các cột>]

[LIMIT số record OFFSET from record]

* INSERT(Thêm mẫu tin):

Cú pháp: INSERT INTO Tên\_bảng VALUES(Bộ\_giá\_trị)

* UPDATE(Cập nhật dữ liệu):

Cú pháp: UPDATE TABLE Tên\_bảng

SET Tên\_cột=Biểu\_thức,...

[WHERE Điều\_kiện]

* DELETE(Xóa mẫu tin):

Cú pháp: DELETE FROM Tên\_bảng

[WHERE Điều\_kiện]

### 1.4.4. Các hàm thông dụng trong MySQL

#### 1.4.4.1. Các hàm trong phát biểu GROUP BY

* Hàm AVG: Hàm trả về giá trị bình quân của cột hay trường trong câu truy vấn
* Hàm MIN: Hàm trả về giá trị nhỏ nhất của cột hay trường trong câu truy vấn
* Hàm MAX: Hàm trả về giá trị lớn nhất của cột hay trường trong câu truy vấn
* Hàm Count: Hàm trả về số lượng mẩu tin trong câu truy vấn

Hàm Sum: Hàm trả về tổng các giá trị của trường, cột trong câu truy vấn

# CHƯƠNG 2:PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 2.1. Khảo sát và phần tích hệ thống

### 2.1.1. Mục tiêu của hệ thống shop thời trang

* Mục tiêu hiện tại
* Quảng bá website để mọi khách hàng đều biết đến.
* Xây dựng trang web thân thiết với người dùng.
* Nâng cao chất lượng dịch vụ.
* Mục tiêu tương lại
* Nâng cao-mở rộng thị trường..
* Mở cửa hàng trực tiếp để khách hàng có thể thuận lợi đến xem sản phẩm 🡪 tăng độ uy tín của cửa hàng.

### 2.1.2. Lợi ích của việc xây dựng website

* Việc khởi tạo một website là rất cần thiết,website giúp quản bá được thương hiệu và những sản phẩm đến khách hàng.
* Thuận tiện cho khách hàng chỉ cần ở nhà mà cũng có thể mua sắm những thứ minh thích .Không những vậy về việc quản lý thông tin của khách hàng và sản phẩm cũng trở nên đơn giả.

### 2.1.3. Mô tả yêu cầu của hệ thống

Sau khi phân tích đặc điểm tình hình và thực trạng của cửa hàng, việc xây dựng website cần đạt được những yêu cầu như trong mô tả sau:

- Hệ thống sẽ thực hiện quảng bá thương hiệu khắp cả nước.

- Hệ thống làm nhiệm vụ giúp khách hàng truy cập , tìm kiếm sản phẩm theo nhu cầu của mình và đặt mua sản phẩm.

- Nhận góp ý từ khách hàng về các yêu cầu của hệ thống cũng như góp ý về giá trị sản phẩm.

- Hệ thống giúp cho người quản lý , quản lý được thông tin của sản phẩm, quản lý đơn hàng và thông tin của các khách hàng.

### 2.1.4. Mô tả hoạt động của hệ thống

Đối với khách hàng: Khách hàng sẽ truy cập vào website và tìm kiếm những sản phẩm theo nhu cầu của mình, đặt hàng và mua sản phẩm – kiểm tra thông tin đơn hàng đã đặt mua.

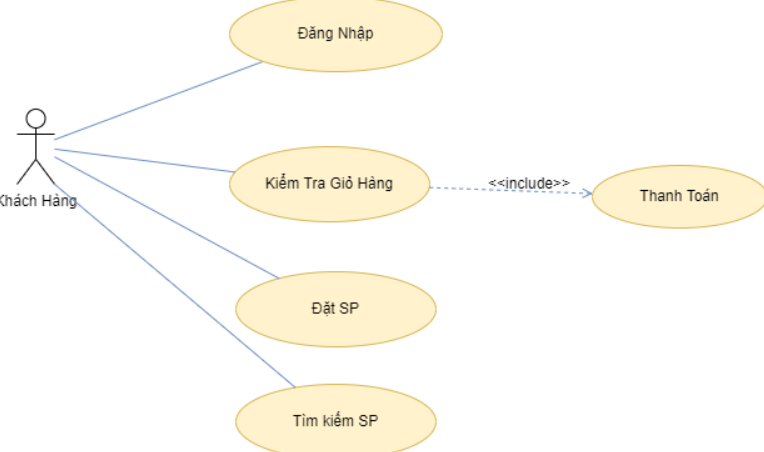
Đối với quản trị viên: Admin sẽ đăng nhập vào hệ thống và quản lý danh mục sản phẩm , quản lý sản phẩm, quản lý đơn hàng và thông tin khách hàng.

## 2.2. Biểu đồ Use case

Một biểu đồ Use case chỉ ra một số lượng các tác nhân ngoại cảnh và mối liên kết của chúng với Use case mà hệ thống cung cấp. Lời miêu tả Use case thường là một văn bản tài liệu, nhưng kèm theo đó cũng có thể là một biểu đồ hoạt động. Các Use case được miêu tả duy nhất theo hướng nhìn từ ngoài vào của các tác nhân (hành vi của hệ thống theo như sự mong đợi của người sử dụng), không miêu tả chức năng được cung cấp sẽ hoạt động nội bộ bên trong hệ thống ra sao. Các Use case định nghĩa các yêu cầu về mặt chức năng đối với hệ thống.

### 2.2.1.Sơ đồ use case

Dựa vào yêu cầu của bài toán quản lý website bán hàng ta thấy các chức năng chính của hệ thống được thực hiện bởi khách hàng. Như vậy chúng ta có một tác nhân là khách hàng. Trong đó khách hàng có liên quan đến chức năng tra cứu tìm hàng, đặt hàng và thanh toán.

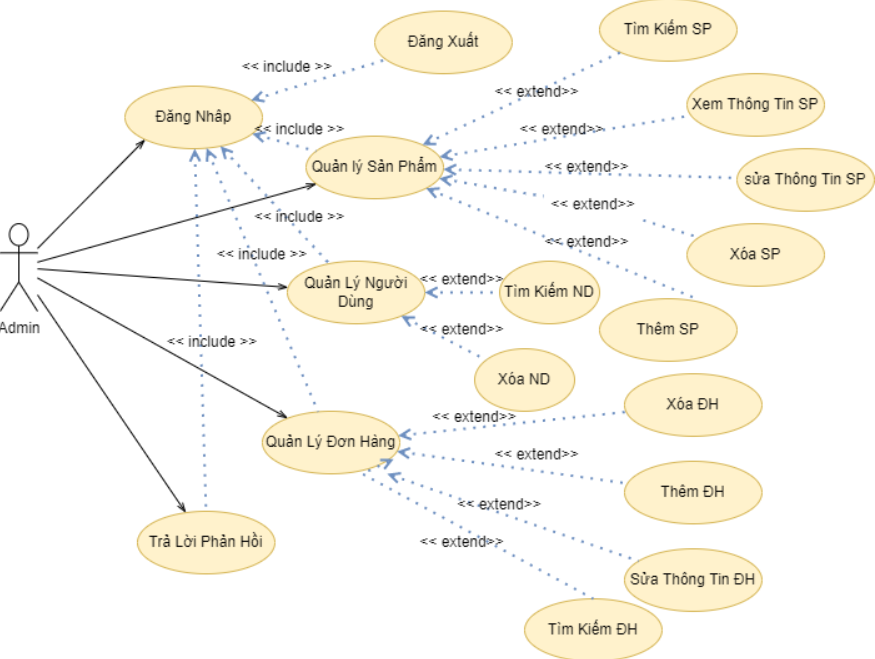


Biểu đồ use-case Khách Hàng



Biểu đồ use-case Tổng Quát

### 2.2.2. Use-case đăng nhập Admin:



**Hình 2.Biểu đồ use case đăng nhập Admin**

**1.Tóm tắt**

Actor Admin là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng đăng nhập vào hệ thống của trang quản trị. Sau khi đăng nhập vào hệ thống, lúc đó Admin mới có thể thực hiện được các chức năng quản lý…

**2.Dòng sự kiện**

**a. Dòng sự kiện chính**

 (1). Admin chọn chức năng đăng nhập từ giao diện của hệ thống.

 (2). Hệ thống hiển thị trang chứa form đăng nhập.

 (3). Admin nhập tài khoản của mình (bao gồm tên đăng nhập và mật khẩu).

 (4). Hệ thống xác nhận thông tin đăng nhập.

 (5). Hệ thống xử lý quyền đăng nhập.

 (6). Hiển thị trang quản lý.

 (7). Kết thúc Use Case.

**b. Các dòng sự kiện khác**

 (1). Admin nhập sai thông tin đăng nhập.

 (2). Hệ thống từ chối đăng nhập, hiển thị thông báo.

 (3). Kết thúc Use Case.

**3.Các yêu cầu đặc biệt**

Không có yêu cầu đặc biệt.

**4.Trạng thái hệ thống khi bắt đầu thực hiện Use Case**

Không đòi hỏi bất cứ điều kiện nào trước đó.

**5.Trạng thái hệ thống sau khi thực hiện Use Case**

**\*** Trường hợp đăng nhập thành công: Hệ thống hiển thị trang chủ quản lý.

\* Trường hợp đăng nhập thất bại: Hệ thống đưa ra thông báo đăng nhập không thành công và hiển thị giao diện đăng nhập.

### 2.2.3. Use-case khách hàng tiềm năng:

**Hình 3.Biểu đồ use case khách hàng tiềm năng**

1. **Tóm tắt**

Actor Admin là người sử dụng Use Case này. Use Case này được sử dụng để thực hiện chức năng thêm, sửa, xóa danh mục sản phẩm.

**2.Dòng sự kiện**

**a. Dòng sự kiện chính**

 (1). Admin chọn chức năng Quản lý

 (2). Hệ thống chuyển đến trang quản lý

 (3). Tại đây Admin : Thêm danh mục sản phẩm: chọn thêm -> admin sẽ nhập tên danh mục sau đó gửi -> hệ thống sẽ lưu và chuyển đến trang danh sách danh mục.

(4). Sửa, cập nhập danh mục sản phẩm: trong giao diện danh sách danh mục sản phẩm, admin cần sửa thì nhấn vào nút sửa sau đó hệ thống sẽ chuyển đến trang giao diện sửa danh mục sản phẩm -> khi sửa xong thì admin chọn gửi hệ thống sẽ lưu và chuyển ngược về giao diện danh sách danh mục.

(5). Xóa danh mục sản phẩm: ở giao diện danh sách danh mục sản phẩm, chọn danh mục cần xóa, hệ thống sẽ xóa về ở lại trang giao diện danh sách danh mục.

  (4). Kết thúc Use Case.

**b. Các dòng sự kiện khác**

\* Dòng sự kiện thứ nhất

 (1). Hệ thống có lỗi trong quá trình xử lý.

 (2). Hệ thống thông báo lỗi.

 (3). Kết thúc Use Case.

**3. Các yêu cầu đặc biệt**

Admin trước khi thực hiện Use Case này đòi hỏi Use Case đăng nhập vào hệ thống quản trị phải được thực hiện trước và phải thực hiện thành công.

**4. Trạng thái hệ thống khi bắt đầu thực hiện Use Case**

Trước khi thực hiện Use Case này đòi hỏi Use Case đăng nhập phải được thực hiện trước và phải thực hiện thành công, nghĩa là trước khi thực hiện chức năng quản lý, Admin cần phải đăng nhập thành công.

**5. Trạng thái hệ thống sau khi thực hiện Use Case**

**\*** Trường hợp thành công: hệ thống thêm, sửa, xóa và cập nhật lại ở trong cơ sở dữ liệu.

\* Trường hợp thất bại: Hệ thống đưa ra thông báo lỗi.

***1. Tóm tắt***

Use Case cho phép xem chi tiết đơn hàng và xóa những đơn hàng sai hay lỗi về thông tin khách hàng.

***Dòng sự kiện***

**a. Dòng sự kiện chính**

 (1). Admin chọn chức năng Quản lý

 (2). Hệ thống chuyển đến trang quản lý

 (3). Tại đây Admin: Xem chi tiết đơn hàng: chọn chức năng xem đơn hàng 🡪 hệ thống sẽ chuyển sang trang để admin có thể xem chi tiết đơn hàng

Xóa đơn hàng: Admin có thể xóa những đơn hàng có thông tin không đúng yêu cầu và không rỏ ràng để tiện việc quản lý các đơn hàng cho cửa hàng hơn 🡪 trên giao diện admin nhấn xóa -> đơn hàng sẽ được xóa và ở lại giao diện trang danh sách đơn hàng

(4). Kết thúc Use Case.

**b. Các dòng sự kiện khác**

\* Dòng sự kiện thứ nhất

 (1). Hệ thống có lỗi trong quá trình xử lý.

 (2). Hệ thống thông báo lỗi.

 (3). Kết thúc Use Case.

***3. Các yêu cầu đặc biệt***

Admin trước khi thực hiện Use Case này đòi hỏi Use Case đăng nhập vào hệ thống quản trị phải được thực hiện trước và phải thực hiện thành công.

***4. Trạng thái hệ thống khi bắt đầu thực hiện Use Case***

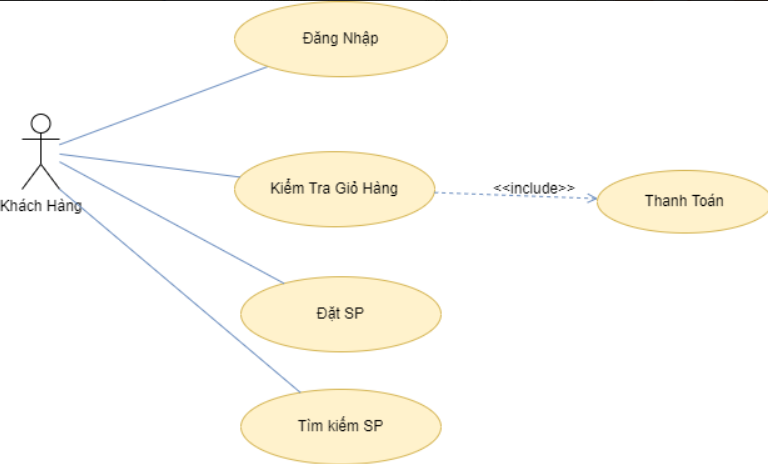
Trước khi thực hiện Use Case này đòi hỏi Use Case đăng nhập phải được thực hiện trước và phải thực hiện thành công, nghĩa là trước khi thực hiện chức năng quản lý, Admin cần phải đăng nhập thành công.

***5. Trạng thái hệ thống sau khi thực hiện Use Case***

**\*** Trường hợp thành công: hệ thống thêm, sửa, xóa và cập nhật lại ở trong cơ sở dữ liệu.

\* Trường hợp thất bại: Hệ thống đưa ra thông báo lỗi.

### 2.2.7. Use-case khách hàng



**.Biểu đồ use case khách hàng**

* Mô tả: use case cho phép khách hàng truy cập đến cửa hàng tìm kiếm sản phẩm và đặt hàng.
* Điều kiện trước tiên: khách hàng truy cập vào trang website
* Dòng sự kiện:
* Khách hàng truy cập trang website có thể tìm kiếm sản phẩm cần mua tại các danh mục sản phẩm và loại sản phẩm để xem thông tin sản phẩm cần mua.
* Sau khi xem xong thông tin nếu khách muốn mua sản phẩm thì có thể đặt hàng.
* Muốn mua sản phẩm thì khách hàng cần cho thông tin rỏ ràng ( tên, số điện thoại, địa chỉ,…) để hệ thống có thể đưa vào cơ sở dữ liệu và dựa trên thông tin đó để giao hàng đến khách hàng.

Khách hàng sau khi đặt hàng sẽ chọn các hình thức thanh toán , thông tin đơn hàng sẽ gửi tới hệ thống quản lý đơn hàng để admin có thể nắm rỏ thông tin và chuẩn bị hàng giao đến khách hàng.

### 2.2.8. Danh sách các Actor

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Actor** | **Ý nghĩa/ Ghi chú** |
| 1 | Admin | Actor này có toàn quyền tương tác với hệ thống, có quyền điều khiển cũng như kiểm soát mọi hoạt động của hệ thống. |
| 2 | Khách hàng | Actor này có quyền thực hiện các chức năng trên trang web. |

**.Danh sách các Actor**

2.2.9. Danh sách các Use Case

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Use Case** | **Ý nghĩa/ Ghi chú** |
|  | Use Case Đăng nhập,Đăng Xuất của Admin | Use Case này mô tả chức năng đăng nhập,đăng xuất của hệ thống của admin. |
|  | Use Case Quản Lý Danh Mục Sản Phẩm | Use Case này mô tả chức năng thêm, sửa, xóa sản phẩm |
|  | Use Case Quản Lý Loại Sản Phẩm | Use Case này mô tả chức năng thêm, sửa, xóa sản phẩm |
|  | Use Case Quản Lý Sản Phẩm | Use Case này mô tả chức năng thêm, sửa, xóa sản phẩm |
|  | Use Case Quản lý đơn hàng | Use Case này mô tả chức năng xem chi thiết đơn hàng và xóa những đơn hàng sai thông tin khách hàng. |
|  | Use Case Khách Hàng | Use Case này cho phép khách hàng truy cập đến cửa hàng tìm kiếm sản phẩm và đặt hàng. |

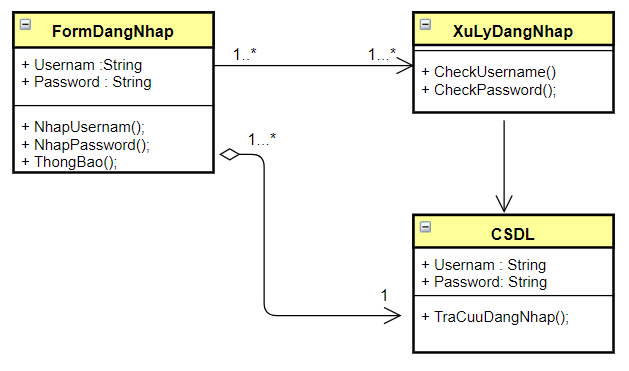
**Danh sách các Use case**

# CHƯƠNG 3:PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

## 3.1.Biểu Đồ Diagram Bán Đồ Ăn

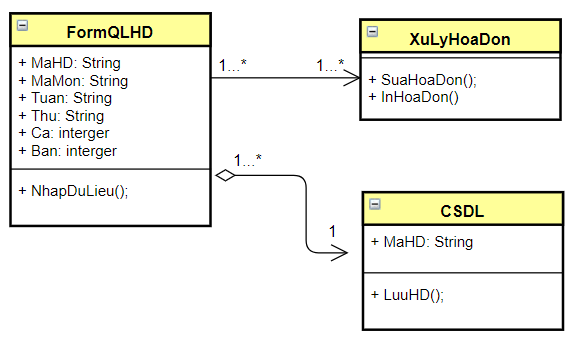
Class Diagram bán đồ ăn

### 3.1.1.Đăng Nhập



Đăng Nhâp

### 3.12. Hóa Đơn



## 3.2.Bảng LoaiHang

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| MaLoai | int | Mã loại mặt hàng |
| TenLoai | nvarchar(100) | Tên loại mặt hàng |

## 3.3.Bảng MatHang

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| MaHang | int | Mã mặt hàng |
| TenHang | nvarchar(100) | Tên mặt hàng |
| DonGia | int | Đơn giá mặt hàng |
| HinhAnh | nvarchar(200) | Đường link hình ảnh sản phẩm |
| MoTa | nvarchar(100) | Mô tả mặt hàng |
| Maloai | int | Mã loại mặt hàng |

## 3.4.Bảng DonDatHang

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| MaDonHang | int | Mã đơn hàng |
| maKhach | int | Mã người khách |
| NgayDat | datetime | Ngày đặt hàng |
| NgayGiao | Nvarchar(50) | Thời điểm giao hàng |

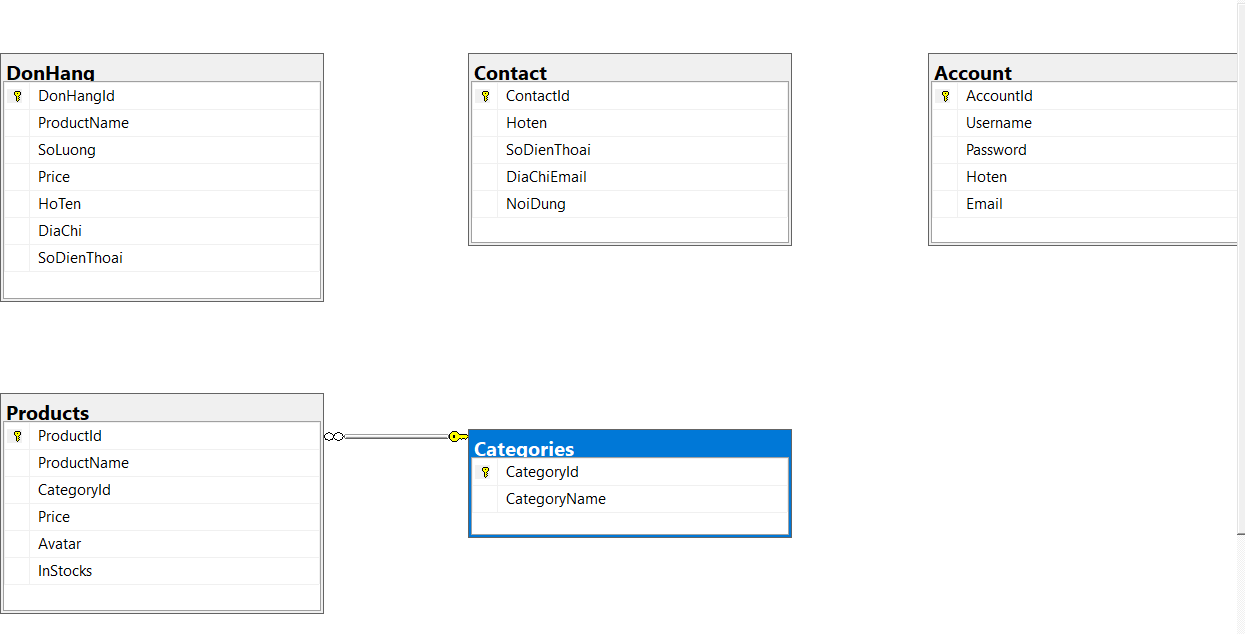
## 3.5.Bảng CTDonDatHang

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| MaDonHang | int | Mã đơn hàng |
| MaHang | int | Mã mặt hàng |
| SoLuong | int | Số lượng đặt hàng |

## 3.6.Bảng KhachHang

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| maKhach | int | Mã  khách |
| TenKhach | nvarchar(50) | Tên khách |
| DiaChi | nvarchar(50) | Địa chỉ |
| DienThoai | nchar(10) | Điện thoại |
| Email | nvarchar(50) | Email |

## 3.7.Database



## 3.8.Bảng csdl

Cơ sở dữ liệu sẽ bao gồm 11 bảng:

- Menu: danh mục sản phẩm gồm có id danh mục, tên danh mục, và hình ảnh của danh mục

Graphical user interface, application

Description automatically generated with medium confidence

- Slide: ảnh chiếu lần lượt trên web gồm: id ảnh chiếu, tiêu đề ảnh chiếu, mô tả ảnh chiếu, đường dẫn ảnh chiếu

Graphical user interface, text, application

Description automatically generated

- News: tin tức gồm: id tin tức, tiêu đề tin tức, ảnh tin tức, chi tiết tin tức

Graphical user interface, text, application

Description automatically generated with medium confidence

- Bill: Đơn hàng gồm: id đơn hàng, tên khách hàng, số điện thoại khách hàng, id khách hàng, ngày đặt hàng, tình trạng

Graphical user interface, application

Description automatically generated

- Bill Detail: Chi tiết đơn hàng gồm: id chi tiết đơn hàng, số lượng, tên sản phẩm, giá sản phẩm, id đơn hàng, tổng tiền

Graphical user interface, application

Description automatically generated

- Customer: Khách hàng gồm: id khách hàng, tên khách hàng, email khách hàng, số điện thoại khách hàng, mật khẩu khách hàng

Graphical user interface, application

Description automatically generated

- CustomerSP: Thông tin khách hàng liên lạc gồm: id khách hàng liên lạc, tên khách hàng, email khách hàng, thông tin liên lạc

Graphical user interface, text, application

Description automatically generated

- Cart: giỏ hàng gồm: id giỏ hàng, id sản phẩm, tên sản phẩm, giá sản phẩm, hình ảnh sản phẩm, tổng tiền, số lượng mặt hàng trong giỏ, id khách hàng

Graphical user interface, application

Description automatically generated with medium confidence

- Food: sản phẩm: id sản phẩm, tên sản phẩm, id danh mục, giá sản phẩm, hình ảnh sản phẩm 1, hình ảnh sản phẩm 2, hình ảnh sản phẩm 3, hình ảnh sản phẩm 4, tồn kho, loại sản phẩm (nổi bật – bình thường)

Graphical user interface, application

Description automatically generated with medium confidence

- Admin: quản trị viên gồm: id admin, tài khoản của admin, mật khẩu của admin

Graphical user interface, text, application

Description automatically generated

- Comment: đánh giá sản phẩm gồm: id comment, id sản phẩm, tên sản phẩm, tên khách hàng đánh giá, mô tả đánh giá

Graphical user interface, text, application

Description automatically generated

Mối quan hệ giữa các bảng:

Graphical user interface, application

Description automatically generated

Diagram

Description automatically generated with medium confidence

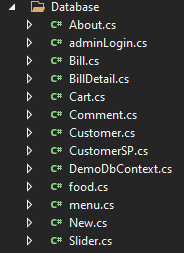
A picture containing diagram

Description automatically generated

Diagram

Description automatically generated

Sau khi đã có cơ sở dữ liệu, tạo các class trong database tương ứng với các bảng để đưa data vào backend để tạo các api



Sau khi đã có đầy đủ các class, tạo DbContext.cs

Text

Description automatically generated

Tạo các api đáp ứng các yêu cầu của đề tài, một số ví dụ

- Tạo api lấy data sản phẩm từ cơ sở dữ liệu

Text

Description automatically generated

Kết quả thu được:

Graphical user interface

Description automatically generated

- Tạo api đưa data sản phẩm mới vào cơ sở dữ liệu:

Text

Description automatically generated

- Tạo api chỉnh sửa data sản phẩm trong cơ sở dữ liệu:

Text

Description automatically generated

- Tạo api xóa data sản phẩm khỏi cơ sở dữ liệu:

Text

Description automatically generated

# KẾT LUẬN

Trong đồ án này,em đã vẫn dụng phương pháp phân tích thiết kế hướng cấu trúc để xây dựng website quản lý nhà hàng.kết quả đạt được bao gồm

* **Lý thuyết:**
* Phát biểu và mô tả được nghiệp vụ bài toán
* Biểu diễn các nghiệp vụ bài toàn bằng các sơ đồ tiền trình nghiệp vụ.
* Phân tích thiết kế bài toàn một cách đầy đủ,theo đúng quy trình được học bằng hướng cấu trúc.
* Thiết kế được cơ sở dữ liệu quan hệ để lưu trữ dữ liệu.
* Thiết kế được các giao diện cập nhật dữ liệu cà các mầu báo cáo.
* **Chương trình:**
* Sử dụng hệ quản trị cơ sở dữ liệu MySql
* Sử dụng ngôn ngữ javascript
* Sử dụng css đễ trang trí web,tạo bố cục trên một trang web.
* Sử dụng thư viên jquery tao các hiệu ứng slideDown,slideUp,..và dư dụng ajax để load trang.
* Hệ thống đã được cài thử nghiệm với một số dữ liệu chạy thông suốt và cho ra kết quả,đáp ứng được các yêu cầu bài toán đặt ra.Kiểm soát được các lỗi ngoại lệ phát sinh do người dung nhập vào không đúng.Báo cáo các mầu theo yêu câu.

Vì thời gian có hạn,kinh nghiệp thực tế chưa nhiều nên việc phân tích bài toán về cơ bản đã thực hiện tương đối đầy đủ,tuy nhiên chưa thể mô ta đầy đủ mọi khía cạch của vấn đề.Xây dựng được hệ thống nhưng chỉ với các chức năng chính, cần thêm nhiều chức năng để chương trình tốt và hoàn chỉnh hơn.

Sau này có điều kiện hoàn thiện các chức năng đã có và bổ sung thêm các chức năng thăng tính tiện dụng của hệ thống nhưa đặt món ăn qua mạng.

**Ưu điểm**

+ Cơ bản nắm được các bước xây dựng một website bán hàng trực tuyến, thực hiện đúng quy trình

+ Xây dựng được bố cục trang web hợp lí, bước đầu thực hiện được nghiệp vụ của hệ thống

**Hạn chế**

+ website nhỏ, mang tính chất mô phỏng

+ Cơ sở dữ liệu nhỏ, chức năng phân quyền chưa tối ưu

+ Chưa kết hợp xử lí chức năng thanh toán hóa đơn

## 1. Những việc đã làm được

* Chức năng của Admin:
  + - Thêm sản phẩm
    - Xử lý các liên hệ
    - Xử lý đơn hàng
    - Chỉnh sửa sản phẩm
* Chức năng của Quản lý: Thanh toán hoá đơn
* Chức năng của Thanh Viên:
  + - Đăng ký tài khoản
    - Mua hàng
* Những chức năng bổ sung:
  + - Hiển Thị sản phẩm
    - Phân loại sản phẩm

## 2.Những việc chưa làm được

* Tạo tài khoảng admin

## 3.Hướng phát Triển

+ Xây dựng website sử dụng được đáp ứng đầy đủ yêu cầu nghiệp vụ của hệ thống

+ Phân quyền và quản trị tối ưu

+ Cơ sở dữ liệu phong phú

+ Kết hợp chức năng thanh toán trực tiếp hay qua thẻ ATM(liên kết với hệ thống ngân hàng)

# TÀI LIỆU THAM KHẢO

1. <https://www.w3schools.com/>
2. <https://sharecode.vn/>
3. <https://123code.vn/>
4. <https://123docz.net/>