**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**VIỆN KỸ THUẬT – CÔNG NGHỆ**



**BÁO CÁO NGHIÊN CỨU KHOA HỌC**

Đề Tài

**XÂY DỰNG WEBSITE**

**HỔ TRỢ HỌC TIẾNG NHẬT**

**GVHD: ths. NGUYỄN HỮU VĨNH**

**SVTH: Nguyễn Anh Hào**

**Trần Tiến Đạt**

**MSSV: 1824801030064**

**1824801030161**

**LỚP: D18PM01 & D18PM03**

**Tháng 3/202**

**LỜI CẢM ƠN**

Trong suốt thời gian học tập và rèn luyện tại Trường Đại học Thủ Dầu Một cho đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý Thầy Cô và bạn bè. Với lòng biết ơn sâu sắc  và chân thành nhất, em xin gửi đến quý Thầy Cô ở Viện Kỹ Thuật Công Nghệ – Trường Đại học Thủ Dầu Một đã cùng với tri thức và tâm huyết của mình để truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường.

Và đặc biệt, trong học kỳ này, Viện đã tổ chức cho chúng em được làm báo cáo nghiên cứu khoa học mà theo em là rất hữu ích đối với sinh viên ngành Kỹ Thuật Phần Mềm cũng như tất cả các sinh viên thuộc các chuyên ngành Khoa Học Kĩ Thuật khác. Đó là đề tài “Xây dựng website hổ trợ học tiếng Nhật”. Em xin chân thành cảm ơn thầy Vĩnh đã tận tâm hướng dẫn chúng em qua từng buổi học trên lớp cũng như những buổi nói chuyện, thảo luận về lĩnh vực sáng tạo trong nghiên cứu khoa học.

Em cũng xin bày tỏ lòng biết ơn đến ban lãnh đạo của Trường Đại học Thủ Dầu Một và các Khoa Phòng ban chức năng đã trực tiếp và gián tiếp giúp đỡ em trong suốt quá trình học tập và nghiên cứu đề tài này.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên, bài báo cáo này không thể tránh được những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các quý thầy cô để em có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn công tác thực tế sau này.

**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

Bình Dương, ngày...... tháng......năm 2020

**Giáo viên hướng dẫn**

**(Ký tên và ghi rõ họ tên)**

**TS. Nguyễn Hữu Vĩnh**

**NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN**

Bình Dương, ngày...... tháng......năm 2020

**Giáo viên hướng dẫn**

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 4](#_Toc65495683)

[**CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN** 5](#_Toc65495684)

[**1. Tên Đề Tài. 5**](#_Toc65495685)

[**2. Mục Tiêu của Đề Tài. 5**](#_Toc65495686)

[**3. Ý Nghĩa Của Đề Tài. 5**](#_Toc65495687)

[***4.* Đối tượng, phạm vi nghiên cứu, cách tiếp cận và phương pháp nghiên cứu: 5**](#_Toc65495688)

[***a. Đối tượng:* 5**](#_Toc65495689)

[***b. Phạm vi nghiên cứu:* 5**](#_Toc65495690)

[***c. Cách tiếp cận* 5**](#_Toc65495691)

[***d. Phương pháp nghiên cứu:* 5**](#_Toc65495692)

[**5. Một Số Trang Chính Của Website. 5**](#_Toc65495693)

[**CHƯƠNG 2: GIỚI THIỆU CÔNG NGHỆ 6**](#_Toc65495694)

[**1. Giới Thiệu Về HTML. 6**](#_Toc65495695)

[**1.1 HTML là gì? 6**](#_Toc65495696)

[**1.2 Công dụng của HTML. 6**](#_Toc65495697)

[**1.3 Định dạng của HTML. 6**](#_Toc65495698)

[**1.4 Cấu trúc của trang HTML 7**](#_Toc65495699)

[**2. Giới thiệu về Visual Studio Code 7**](#_Toc65495700)

[**3. Giới thiệu về Adobe Dreamweaver 9**](#_Toc65495701)

[**4. Visual Studio 10**](#_Toc65495702)

[**5. Ngôn ngữ CSS. 13**](#_Toc65495703)

[**6. Bootstrap 14**](#_Toc65495704)

[**7. Ngôn ngữ Javascript. 16**](#_Toc65495705)

[**8. ASP.Net 21**](#_Toc65495706)

[**9. ADO.Net 22**](#_Toc65495707)

[**CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 25**](#_Toc65495708)

[**1. Giới thiệu về phần mềm 25**](#_Toc65495709)

[**2. Tác nhân 25**](#_Toc65495710)

[**3. Use Case 25**](#_Toc65495711)

[***4.* *Biểu đồ hoạt độ (Activity Diagram )* 26**](#_Toc65495712)

[***a.* *Đăng nhập* 26**](#_Toc65495713)

[***b.* *Đăng xuất* 27**](#_Toc65495714)

[***c.* *Đổi mật khẩu* 28**](#_Toc65495715)

[***5.* *Biểu đồ tuần tự (Seqence Diagram)* 29**](#_Toc65495716)

[***a.* *Đăng nhập* 29**](#_Toc65495717)

[***b.* *Đăng xuất* 30**](#_Toc65495718)

[***c.* *Đăng kí* 31**](#_Toc65495719)

[***d.* *Tìm kiếm* 32**](#_Toc65495720)

[***e.* *Nhận thông báo* 33**](#_Toc65495721)

[***6.* *Thiết kế lớp* 34**](#_Toc65495722)

[**CHƯƠNG 4: GIAO DIỆN CHƯƠNG TRÌNH 35**](#_Toc65495723)

[**1. Cấu trúc Giao diện 35**](#_Toc65495724)

[**2. Đặc tả Giao diện 36**](#_Toc65495725)

[**2.1 Phần giao diện người dùng 36**](#_Toc65495726)

[**2.2 Phần giao diện quản trị 38**](#_Toc65495732)

[**CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 41**](#_Toc65495733)

[**1. Kết quả đạt được: 41**](#_Toc65495734)

[**2. Việc chưa làm được: 41**](#_Toc65495735)

[**3. Hướng phát triển của đề tài 41**](#_Toc65495736)

[**TÀI LIỆU THAM KHẢO 42**](#_Toc65495737)

# **LỜI NÓI ĐẦU**

Trong thời đại toàn cầu hóa, nhiều công ty Nhật có xu hướng đầu tư nước ngoài và với khả năng kiểm soát dịch bệnh một cách triệt để của Việt Nam thì nước ta là lựa chọn hoàn hảo cho khoản đầu tư như vậy.

Để có thể làm việc một cách tối ưu với người Nhật thì tiếng Anh không phải lựa chọn tối ưu nhất. Hơn thế nữa, Nhật Bản là một nước có ngành công nghiệp phát triển nhanh chóng nên các công nghệ của Nhật vẫn là sự lựa chọn tốt nhất cho các quốc gia khác. Tuy nhiên, ở Nhật có rất nhiều công nghệ hay nhưng họ chỉ áp dụng vào nọi bộ nên các máy móc, các trang thiết bị công nghệ luôn chỉ có tiếng Nhật là ngôn ngữ mạc định. Mà để tiếp thu cũng như triển khai, sử dụng các trang thiết bị hiện đai đó, nguồng nhân lực phải biết tiếng Nhật để dể dàng tiếp thu và sử dụng chúng.

Hay xu hướng hiện nay chính là hội nhập hóa quốc tế, có rất nhiều người Việt Nam muốn làm việc và định cư ở Nhật. Và chính vì để sống được một các dể dàng ở nước ngoài đó không phải là chuyện dể nhưng nếu biết được ngôn ngữ nước họ đó sẽ là ngọn lữa sôi sáng cong đường cho họ.

Tóm lại, tiếng Nhật là một thứ rất quan trọng đối với tất cả mọi người và việc học vô cùng cấp thiết.

Tuy nhiên không phải ai cũng có thời gian để học. Hiện nay, con người chúng ta sống rất nhanh và bận rộn, ta chỉ có những thời gian rảnh khi đi trên xe điện như văn hóa Xe Điện của người Nhật, họ tranh thủ sẽ học tập vào thời gian đó. Vậy ta sẽ phải có giải pháp như thế nào?

Xây dựng một Hệ Thống Phần Mềm Hổ Trợ Học Tiếng Nhật để giúp người bận rộn hoặc yêu thích tiếng Nhật có thể học tiếng Nhật bất cứ nơi đâu bất cứ lúc nào với những bài test hằng ngày và đặc biệt là luyện thi JLPT.

Đề tài tiểu luận gồm các phần được phân chương như sau:

**Chương 1: Tổng quan về đề tài**

**Chương 2: Giới thiệu công nghệ**

**Chương 3: Phân tích thiết kế hệ thống**

**Chương 4: Giao diện website**

**Chương 5: Kết luận và hướng phát triển**

Link tài nguyên: https://github.com/aokidai/baiduthinckh

# **CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN**

1. Tên Đề Tài.

XÂY DỰNG WEBSITE HỔ TRỢ HỌC TIẾNG NHẬT

1. Mục Tiêu của Đề Tài.

Xây dựng một trang web giúp người bận rộn hoặc yêu thích tiếng Nhật có thể học tiếng Nhật bất cứ nơi đâu bất cứ lúc nào với những bài test hằng ngày và đặc biệt là luyện thi JLPT.

1. Ý Nghĩa Của Đề Tài.

* Giúp mọi người có cơ hội học Tiếng Nhật
* Hổ trợ luyện thi JLPT

1. **Đối tượng, phạm vi nghiên cứu, cách tiếp cận và phương pháp nghiên cứu:**

### ***a. Đối tượng:***

Những người mong muốn học tiếng Nhật.

### ***b. Phạm vi nghiên cứu:***

Công tác thu thập, tổng hợp ý kiến từ các giảng viên và các sinh viên.

### ***c. Cách tiếp cận***

Tìm hiểu thực trạng về mặt ưu điểm và khuyết điểm của cách điểm danh thủ công.

Tìm hiểu các ứng dụng có chức năng tương tự, từ đó phân tích các ưu, khuyết điểm làm cơ sở để thực hiện đề tài.

Tìm hiểu các ngôn ngữ lập trình và kỹ thuật xây dựng ứng dụng di động đáp ứng các yêu cầu đề ra.

### ***d. Phương pháp nghiên cứu:***

* Phương pháp nghiên cứu lý thuyết.
* Tìm hiểu công nghệ ASP.Net.
* Lý thuyết phân tích thiết kế hệ thống.
* Phương pháp thực nghiệm.
* Thiết kế và lập trình web app.
* Xây dựng cơ sở dử liệu

1. Một Số Trang Chính Của Website.

* Trang chủ
* Trang quản trị
* Các khóa học

# **CHƯƠNG 2: GIỚI THIỆU CÔNG NGHỆ**

1. Giới Thiệu Về HTML.
   1. **HTML là gì?**

HTML (Hypertext Markup Language-Ngôn ngữ đánh dấu siêu văn bản) là ngôn ngữ đánh dấu chuẩn dùng soạn thảo các tài liệu World Wide Web, chỉ rõ một trang Web được hiển thị như thế nào trong trình duyệt.

Ngôn ngữ HTML dùng các tag hoặc các đoạn mã lệnh để chỉ cho các trình duyệt (Web browsers) cách hiển thị các thành phần của trang như text và graphics.

HTML là ngôn ngữ xác định cấu trúc của thông tin.

HTML sử dụng một loạt các thẻ và thuộc tính và được dùng để hiện thị văn bản và các thông tin khác, cung cấp siêu liên kết tới các tài liệu khác.

* 1. **Công dụng của HTML.**

Thiết kế được nội dung và hình thức của trang web.

Xuất bản được các tài liệu trực tuyến hay truy xuất các thông tin trực tuyến bằng cách dùng các liên kết được chèn vào trang web.

Tạo ra các biểu mẫu trực tuyến nhằm thu thập các thông tin người dùng, quản lý giao dịch….

Thêm vào đối tượng các hình ảnh video, âm thanh…

* 1. **Định dạng của HTML.**

Là một tập tin có phần mở rộng là .htm hoặc .html

Là một tập tin văn bản thuần túy, có chứa các mã đặc biệt gọi là Tag(thẻ), thường được đặt xung quanh một khối văn bản nào đó.

Có thể viết trên nhiều trình soạn thảo: Microsoft Fontpage, notepad…và được sử dụng nhiều nhất đó là Adobe Dreamweaver.

* 1. **Cấu trúc của trang HTML**



1. Giới thiệu về Visual Studio Code

Visual Studio Code là một trình biên tập mã được phát triển bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft) dành cho [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows), [Linux](https://vi.wikipedia.org/wiki/Linux" \o "Linux) và [MacOS](https://vi.wikipedia.org/wiki/MacOS" \o "MacOS). Nó hỗ trợ chức năng debug, đi kèm với [Git](https://vi.wikipedia.org/wiki/Git_(ph%E1%BA%A7n_m%E1%BB%81m)), có syntax highlighting, tự hoàn thành mã thông minh, [snippets](https://vi.wikipedia.org/w/index.php?title=Snippets&action=edit&redlink=1" \o "Snippets (trang chưa được viết)), và [cải tiến mã nguồn](https://vi.wikipedia.org/wiki/C%E1%BA%A3i_ti%E1%BA%BFn_m%C3%A3_ngu%E1%BB%93n" \o "Cải tiến mã nguồn). Nó cũng cho phép tùy chỉnh, do đó, người dùng có thể thay đổi theme, phím tắt, và cá tùy chọn khác. Nó miễn phí và là [phần mềm mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_t%E1%BB%B1_do_ngu%E1%BB%93n_m%E1%BB%9F), mặc dù gói tải xuống chính thì là [có giấy phép](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_s%E1%BB%9F_h%E1%BB%AFu_%C4%91%E1%BB%99c_quy%E1%BB%81n" \o "Phần mềm sở hữu độc quyền).

Visual Studio Code được dựa trên [Electron](https://vi.wikipedia.org/wiki/Electron), một nền tảng được sử dụng để triển khai các ứng dụng [Node.js](https://vi.wikipedia.org/wiki/Node.js) máy tính cá nhân chạy trên động cơ bố trí Blink. Mặc dù nó sử dụng nền tảng Electron nhưng phần mềm này không phải là một bản khác của [Atom](https://vi.wikipedia.org/w/index.php?title=Atom&action=edit&redlink=1), nó thực ra được dựa trên trình biên tập của [Visual Studio Online](https://vi.wikipedia.org/wiki/Microsoft_Visual_Studio) (tên mã là "Monaco").

Trong cuộc khảo sát vào năm 2018 trên Stack Overflow, Visual Studio Code được xếp hạng là trình biên tập mã phổ biến nhất, với 34.9% của 75398 người trả lời tuyên bố sử dụng nó.

Visual Studio Code được công bố, ra mắt phiên bản thử nghiệm vào ngày 29 tháng 4 năm 2015 bởi Microsoft tại hội nghị [Build](https://vi.wikipedia.org/wiki/Build_(h%E1%BB%99i_ngh%E1%BB%8B_nh%C3%A0_ph%C3%A1t_tri%E1%BB%83n)) 2015.

Vào ngày 18 tháng 11 năm 2015, Visual Studio Code được phát hành dưới giấy phép [MIT](https://vi.wikipedia.org/wiki/Gi%E1%BA%A5y_ph%C3%A9p_MIT" \o "Giấy phép MIT) và mã nguồn của nó được đưa lên [GitHub](https://vi.wikipedia.org/wiki/GitHub" \o "GitHub). Chức năng Extesion (tiện ích mở rộng) cũng được công bố.

Vào ngày 14 tháng 4 năm 2016 Visual Studio Code hoàn tất thử nghiệm và được [phát hành chính thức](https://vi.wikipedia.org/wiki/V%C3%B2ng_%C4%91%E1%BB%9Di_ph%C3%A1t_h%C3%A0nh_ph%E1%BA%A7n_m%E1%BB%81m)

Visual Studio Code là một trình biên tập mã. Nó hỗ trợ nhiều ngôn ngữ và chức năng tùy vào ngôn ngữ sử dụng theo như trong bảng sau. Nhiều chức năng của Visual Studio Code không hiển thị ra trong các menu tùy chọn hay giao diện người dùng. Thay vào đó, chúng được gọi thông qua khung nhập lệnh hoặc qua một tập tin [.json](https://vi.wikipedia.org/wiki/JSON) (ví dụ như tập tin tùy chỉnh của người dùng). Khung nhập lệnh là một giao diện theo dòng lệnh. Tuy nhiên, nó biến mất khi người dùng nhấp bất cứ nơi nào khác, hoặc nhấn tổ hợp phím để tương tác với một cái gì đó ở bên ngoài đó. Tương tự như vậy với những dòng lệnh tốn nhiều thời gian để xử lý. Khi thực hiện những điều trên thì quá trình xử lý dòng lệnh đó sẽ bị hủy.

|  |  |
| --- | --- |
| Các chức năng phụ thuộc vào ngôn ngữ | |
| Chức năng | Ngôn ngữ |
| Syntax highlighting | [Batch](https://vi.wikipedia.org/wiki/Batch_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), [C++](https://vi.wikipedia.org/wiki/C%2B%2B), Clojure, CoffeeScript, [DockerFile](https://vi.wikipedia.org/wiki/Docker_(ph%E1%BA%A7n_m%E1%BB%81m)), Elixir, F#, [Go](https://vi.wikipedia.org/wiki/Go_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), Pug template language, [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), HandleBars, Ini, [Lua](https://vi.wikipedia.org/wiki/Lua" \o "Lua), Makefile, Objective-C, [Perl](https://vi.wikipedia.org/wiki/Perl" \o "Perl), PowerShell, [Python](https://vi.wikipedia.org/wiki/Python_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), [R](https://vi.wikipedia.org/wiki/R_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), Razor, [Ruby](https://vi.wikipedia.org/wiki/Ruby_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)" \o "Ruby (ngôn ngữ lập trình)), Rust, [SQL](https://vi.wikipedia.org/wiki/SQL" \o "SQL), [Visual Basic](https://vi.wikipedia.org/wiki/Visual_Basic), [XML](https://vi.wikipedia.org/wiki/XML) |
| Snippets | [Groovy](https://vi.wikipedia.org/wiki/Groovy_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)), [Markdown](https://vi.wikipedia.org/wiki/Markdown), Nim, [PHP](https://vi.wikipedia.org/wiki/PHP" \o "PHP), [Swift](https://vi.wikipedia.org/wiki/Swift_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) |
| Tự động hoàn thành mã thông minh | [CSS](https://vi.wikipedia.org/wiki/CSS), [HTML](https://vi.wikipedia.org/wiki/HTML), [JavaScript](https://vi.wikipedia.org/wiki/JavaScript), [JSON](https://vi.wikipedia.org/wiki/JSON), Less, Sass, TypeScript |
| [Cải tiến mã nguồn](https://vi.wikipedia.org/wiki/C%E1%BA%A3i_ti%E1%BA%BFn_m%C3%A3_ngu%E1%BB%93n) | [C#](https://vi.wikipedia.org/wiki/C_th%C4%83ng), [TypeScript](https://vi.wikipedia.org/wiki/Typescript) |
| Debugging | [JavaScript](https://vi.wikipedia.org/wiki/JavaScript) và [TypeScript](https://vi.wikipedia.org/wiki/Typescript) cho [Node.js](https://vi.wikipedia.org/wiki/Node.js)  [C#](https://vi.wikipedia.org/wiki/C_th%C4%83ng) và F# cho [Mono](https://vi.wikipedia.org/wiki/Mono_(ph%E1%BA%A7n_m%E1%BB%81m)) trên [Linux](https://vi.wikipedia.org/wiki/Linux) và [macOS](https://vi.wikipedia.org/wiki/MacOS)  [C](https://vi.wikipedia.org/wiki/C_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [C++](https://vi.wikipedia.org/wiki/C%2B%2B) trên [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux) và [macOS](https://vi.wikipedia.org/wiki/MacOS)  [Python](https://vi.wikipedia.org/wiki/Python_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) |

Visual Studio Code có thể được mở rộng qua [plugin](https://vi.wikipedia.org/wiki/Plugin). Điều này giúp bổ sung thêm chức năng cho trình biên tập và hỗ trợ thêm ngôn ngữ. Một tính năng đáng chú ý là khả năng tạo phần mở rộng để phân tích mã, như là các linter và công cụ phân tích, sử dụng [Language Server Protocol](https://vi.wikipedia.org/w/index.php?title=Language_Server_Protocol&action=edit&redlink=1).

1. Giới thiệu về Adobe Dreamweaver

Macromedia Dreamweaver 8 là trình biên soạn HTML chuyên nghiệp dùng để thiết kế, viết mã và phát triển website cùng các trang web và các ứng dụng web. Cho dù bạn có thích thú với công việc viết mã HTML thủ công hoặc bạn thích làm việc trong môi trường biên soạn trực quan, Dreamweaver cung cấp cho bạn những công cụ hữu ích để nâng cao kinh nghiệm thiết kế web của bạn.

Các tính năng biên soạn trực quan trong Dreamweaver cho phép bạn tạo nhanh các trang web mà không cần các dòng mã. Bạn có thể xem tất cả các thành phần trong website của bạn và kéo chúng trực tiếp từ một panel dễ sử dụng vào 1 văn bản. Bạn có thể nâng cao sản phẩm của bạn bằng cách tạo và sửa các ảnh trong Macromedia Fireworks hoặc trong ứng dụng ảnh khác, rồi sau đó chèn trực tiếp vào Dreamweaver. Dreamweaver cũng cung cấp những công cụ giúp đơn giản hóa việc chèn Flash vào trang web.

Dreamweaver cũng cho phép bạn Thiết kế các ứng dụng web động dựa theo dữ liệu sử dụng công nghệ máy chủ như CFML, ASP.NET, ASP, JSP, và PHP. Nếu sở thích của bạn là làm việc với dữ liệu XML, Dreamweaver cung cấp những công cụ cho phép bạn dễ dàng tạo các trang XSLT, chèn file XML và hiển thị dữ liệu XML trên trang web của bạn.

Dreamweaver có thể tùy biến hoàn toàn. Bạn có thể tạo cho riêng mình những đối tượng và yêu cầu, chỉnh sửa shortcut bàn phím và thậm chí viết mã JavaScript để mở rộng những khả năng của Dreamweaver với những hành vi mới, những chuyên gia giám định Property mới và những báo cáo site mới.

1. Visual Studio

Microsoft Visual Studio là một [môi trường phát triển tích hợp](https://vi.wikipedia.org/wiki/M%C3%B4i_tr%C6%B0%E1%BB%9Dng_ph%C3%A1t_tri%E1%BB%83n_t%C3%ADch_h%E1%BB%A3p) (IDE) từ [Microsoft](https://vi.wikipedia.org/wiki/Microsoft). Nó được sử dụng để phát triển [chương trình máy tính](https://vi.wikipedia.org/wiki/Ch%C6%B0%C6%A1ng_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh) cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như [Windows API](https://vi.wikipedia.org/wiki/Windows_API" \o "Windows API), [Windows Forms](https://vi.wikipedia.org/w/index.php?title=Windows_Forms&action=edit&redlink=1), [Windows Presentation Foundation](https://vi.wikipedia.org/wiki/Windows_Presentation_Foundation), [Windows Store](https://vi.wikipedia.org/wiki/Windows_Store) và [Microsoft Silverlight](https://vi.wikipedia.org/wiki/Microsoft_Silverlight). Nó có thể sản xuất cả hai [ngôn ngữ máy](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_m%C3%A1y" \o "Ngôn ngữ máy) và [mã số quản lý](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_s%E1%BB%91_qu%E1%BA%A3n_l%C3%BD&action=edit&redlink=1).

Visual Studio bao gồm một [trình soạn thảo mã](https://vi.wikipedia.org/w/index.php?title=Tr%C3%ACnh_so%E1%BA%A1n_th%E1%BA%A3o_m%C3%A3&action=edit&redlink=1) hỗ trợ [IntelliSense](https://vi.wikipedia.org/w/index.php?title=IntelliSense&action=edit&redlink=1) cũng như [cải tiến mã nguồn](https://vi.wikipedia.org/wiki/C%E1%BA%A3i_ti%E1%BA%BFn_m%C3%A3_ngu%E1%BB%93n" \o "Cải tiến mã nguồn). Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng [giao diện ứng dụng](https://vi.wikipedia.org/w/index.php?title=Giao_di%E1%BB%87n_%E1%BB%A9ng_d%E1%BB%A5ng&action=edit&redlink=1), [thiết kế web](https://vi.wikipedia.org/wiki/Thi%E1%BA%BFt_k%E1%BA%BF_web), thiết kế [lớp](https://vi.wikipedia.org/wiki/L%E1%BB%9Bp_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)) và thiết kế [giản đồ cơ sở dữ liệu](https://vi.wikipedia.org/w/index.php?title=Gi%E1%BA%A3n_%C4%91%E1%BB%93_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u&action=edit&redlink=1). Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các [hệ thống quản lý phiên bản](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_qu%E1%BA%A3n_l%C3%BD_phi%C3%AAn_b%E1%BA%A3n" \o "Hệ thống quản lý phiên bản) (như [Subversion](https://vi.wikipedia.org/wiki/Subversion)) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các [miền ngôn ngữ cụ thể](https://vi.wikipedia.org/w/index.php?title=Mi%E1%BB%81n_ng%C3%B4n_ng%E1%BB%AF_c%E1%BB%A5_th%E1%BB%83&action=edit&redlink=1" \o "Miền ngôn ngữ cụ thể (trang chưa được viết)) hoặc bộ công cụ dành cho các khía cạnh khác trong [quy trình phát triển phần mềm](https://vi.wikipedia.org/wiki/Quy_tr%C3%ACnh_ph%C3%A1t_tri%E1%BB%83n_ph%E1%BA%A7n_m%E1%BB%81m" \o "Quy trình phát triển phần mềm).

Visual Studio hỗ trợ nhiều [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có [C](https://vi.wikipedia.org/wiki/C_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)" \o "C (ngôn ngữ lập trình)),[[4]](https://vi.wikipedia.org/wiki/Microsoft_Visual_Studio#cite_note-4) [C++](https://vi.wikipedia.org/wiki/C%2B%2B) và [C++/CLI](https://vi.wikipedia.org/w/index.php?title=C%2B%2B/CLI&action=edit&redlink=1) (thông qua [Visual C++](https://vi.wikipedia.org/wiki/Visual_C%2B%2B" \o "Visual C++)), [VB.NET](https://vi.wikipedia.org/wiki/VB.NET) (thông qua [Visual Basic.NET](https://vi.wikipedia.org/wiki/Visual_Basic.NET" \o "Visual Basic.NET)), [C#](https://vi.wikipedia.org/wiki/C_th%C4%83ng) (thông qua [Visual C#](https://vi.wikipedia.org/w/index.php?title=Visual_C&action=edit&redlink=1" \o "Visual C (trang chưa được viết))) và [F#](https://vi.wikipedia.org/w/index.php?title=F_th%C4%83ng&action=edit&redlink=1" \o "F thăng (trang chưa được viết)) (như của [Visual Studio 2010](https://vi.wikipedia.org/w/index.php?title=Visual_Studio_2010&action=edit&redlink=1)). Hỗ trợ cho các ngôn ngữ khác như [J++](https://vi.wikipedia.org/w/index.php?title=J%2B%2B&action=edit&redlink=1" \o "J++ (trang chưa được viết))/[J#](https://vi.wikipedia.org/w/index.php?title=J_th%C4%83ng&action=edit&redlink=1), [Python](https://vi.wikipedia.org/wiki/Python) và [Ruby](https://vi.wikipedia.org/wiki/Ruby) thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ [XML](https://vi.wikipedia.org/wiki/XML)/[XSLT](https://vi.wikipedia.org/wiki/XSLT), [HTML](https://vi.wikipedia.org/wiki/HTML)/[XHTML](https://vi.wikipedia.org/wiki/XHTML), [JavaScript](https://vi.wikipedia.org/wiki/JavaScript) và [CSS](https://vi.wikipedia.org/wiki/CSS).

Microsoft cung cấp phiên bản "Express" (đối với phiên bản Visual Studio 2013 trở về trước) và "Community" (đối với bản Visual Studio 2015 trở về sau) là phiên bản miễn phí của Visual Studio.

Visual Studio không hỗ trợ cho bất kỳ ngôn ngữ lập trình nào về giải pháp hoặc công cụ thực chất, thay vào đó nó cho phép cắm chức năng được mã hóa như là một VSPackage. Khi cài đặt, các chức năng có sẵn như là một dịch vụ. IDE cung cấp ba dịch vụ: SVsSolution cung cấp khả năng liệt kê các dự án và các giải pháp; SVsUIShell cung cấp cửa sổ và giao diện người dùng và SVsShell. Ngoài ra, IDE cũng có trách nhiệm điều phối và cho phép truyền thông giữa các dịch vụ. Tất cả các biên tập viên, nhà thiết kế, các loại dự án và các công cụ khác được thực hiện theo VSPackages. Visual Studio sử dụng [COM](https://vi.wikipedia.org/w/index.php?title=Component_Object_Model&action=edit&redlink=1) để truy cập VSPackages. Visual Studio [SDK](https://vi.wikipedia.org/wiki/B%E1%BB%99_ph%C3%A1t_tri%E1%BB%83n_ph%E1%BA%A7n_m%E1%BB%81m" \o "Bộ phát triển phần mềm) cũng bao gồm Managed Package Framework (MPF) là một tập hợp quản lý bao bọc quanh các COM-interfaces cho phép các gói được viết bằng bất kỳ ngôn ngữ nào. Tuy nhiên, MPF không cung cấp tất cả các chức năng bộc lộ trong Visual Studio COM-interfaces. Các dịch vụ có thể được tiêu thụ để tạo ra các gói khác, để thêm chức năng cho Visual Studio IDE.

Hỗ trợ cho các ngôn ngữ lập trình được thêm vào bằng cách sử dụng một VSPackage đặc biệt được gọi là một dịch vụ ngôn ngữ. Một dịch vụ ngôn ngữ định nghĩa giao tiếp khác nhau mà việc thực hiện VSPackage có thể thực hiện để hỗ trợ thêm cho các chức năng khác nhau. Các chức năng có thể được thêm vào theo cách này bao gồm cú pháp màu, hoàn thành báo cáo kết quả, kết hợp đôi, công cụ chú giải tham số thông tin, danh sách thành viên và đánh dấu lỗi trên nền biên dịch. Nếu giao diện được thực hiện, các tính năng sẽ có sẵn ngôn ngữ. Dịch vụ ngôn ngữ sẽ được thực hiện trên cơ sở mỗi ngôn ngữ. Việc triển khai có thể tái sử dụng mã từ phân tích cú pháp hoặc trình biên dịch cho ngôn ngữ. Dịch vụ ngôn ngữ có thể được triển khai hoặc trong mã nguồn gốc hoặc mã số quản lý. Đối với mã nguồn gốc, thì cả COM-interfaces gốc hoặc Babel Framework (một phần của Visual Studio SDK) đều có thể được sử dụng. Đối với mã số quản lý thì các MPF sẽ bao hàm các dịch vu quản lý văn bản.

Visual Studio không bao gồm bất kỳ [Hệ thống quản lý phiên bản](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_qu%E1%BA%A3n_l%C3%BD_phi%C3%AAn_b%E1%BA%A3n) hỗ trợ kiểm soát mã nguồn nhưng nó xác định hai cách thay thế cho các hệ thống kiểm soát mã nguồn để tích hợp với IDE. Một VSPackage kiểm soát mã nguồn có thể cung cấp giao diện người dùng tùy chỉnh của riêng mình. Ngược lại, một plugin kiểm soát mã nguồn bằng cách sử dụng MSSCCI (Microsoft Source Code Control Interface) cung cấp một tập các chức năng được sử dụng để thực hiện chức năng kiểm soát mã nguồn khác nhau, với một giao diện người dùng Visual Studio tiêu chuẩn. MSSCCI lần đầu tiên được sử dụng để tích hợp [Visual SourceSafe](https://vi.wikipedia.org/w/index.php?title=Visual_SourceSafe&action=edit&redlink=1) với Visual Studio 6.0 nhưng sau đó được mở ra thông qua Visual Studio SDK. Visual Studio.NET 2002 dùng MSSCCI 1.1, và Visual Studio.NET 2003 dùng MSSCCI 1.2. Visual Studio 2005, 2008 và 2010 dùng MSSCCI 1.3.

Visual Studio hỗ trợ chạy nhiều cá thể của môi trường (tất cả đều có VSPackages riêng của mình). Những trường hợp sử dụng các registry hives khác nhau để lưu trữ trạng thái cấu hình và được phân biệt bởi AppID (Application ID). Các trường hợp được đưa ra bởi một AppId-specific.exe cụ thể mà lựa chọn AppID, thiết lập các hive gốc và khởi chạy IDE. VSPackages đăng ký một AppID được tích hợp với VSPackages khác cho AppID đó. Các phiên bản sản phẩm khác nhau của Visual Studio được tạo ra bằng cách sử dụng AppIds khác nhau. Các sản phẩm phiên bản [Visual Studio Express](https://vi.wikipedia.org/w/index.php?title=Visual_Studio_Express&action=edit&redlink=1) được cài đặt với AppIds riêng nhưng với các sản phẩm Standard, Professional và Team Suite chia sẻ cùng AppID. Do đó, người ta có thể cài đặt các phiên bản Express song song với các phiên bản khác, không giống như các phiên bản khác cập nhật các cài đặt tương tự. Phiên bản Professional bao gồm các VSPackages khổng lồ trong phiên bản Standard và Team. Hệ thống AppID được thừa hưởng bởi Visual Studio Shell trong Visual Studio 2008.

1. Ngôn ngữ CSS.

CSS (Cascading Style Sheets) là một ngôn ngữ quy định cách trình bày cho các tài liệu viết bằng HTML, XHTML, XML, SVG, hay UML,…

CSS cung cấp nhiều thuộc tính trình bày dành cho các đối tượng với sự sáng tạo trong việc kết hợp các thuộc tính giúp mang lại hiệu quả cao.

CSS đã được hỗ trợ bởi tất cả các trình duyệt và hiển thị “như nhau” trên mọi hệ điều hành.

CSS đưa ra phương thức áp dụng từ một file CSS ở ngoài. Có hiệu quả đồng bộ khi tạo một website có hàng trăm trang hay khi muốn thay đổi một thuộc tính trình bày nào đó.

CSS được cập nhật liên tục mang lại các trình bày phức tạp và tinh vi hơn.

CSS là viết tắt của Cascading Style Sheets.

CSS mô tả cách các phần tử HTML hiển thị trên màn hình và các phương tiện khác.

CSS rất hữu ích và tiện lợi. Nó có thể kiểm soát tất cả các trang trên một website.

Các stylesheet ngoài được lưu trữ dưới dạng các tập tin .CSS.

CSS được dùng để định nghĩa kiểu cách cho các trang trên website của bạn, gồm cả thiết kế, dàn trang và các cách hiển thị khác nhau trên nhiều thiết bị với kích thước màn hình khác nhau.

**CSS giúp giải quyết vấn đề lớn của HTML**

HTML không có phần tử để để định dạng cho trang web. HTML chỉ được dùng để tạo nội dung cho trang. Khi các phần tử như <font> và thuộc tính màu sắc được thêm vào HTML 3.2, cơn ác mộng của các nhà phát triển web bắt đầu. Việc phát triển một website lớn mà thêm thông tin font hay màu sắc vào từng trang đòi hỏi rất nhiều thời gian.

Để giải quyết vấn đề này, World Wide Web Consortium (W3C) đã tạo ra CSS, giúp loại bỏ việc định dạng kiểu cách khỏi trang HTML.Các định nghĩa liên quan đến kiểu cách được đưa vào tập tin .css và nhờ vào tập tin stylesheet ngoài, bạn có thể thay đổi toàn bộ website chỉ bằng một tập tin duy nhất.

1. Bootstrap

**Bootstrap** là một nền tảng (framework) miễn phí, mã nguồn mở, dựa trên **HTML, CSS & Javascript**, nó được tạo ra để xây dựng các giao diện Website tương thích với tất cả các thiết bị có kích thước màn hình khác nhau.

**Bootstrap** bao gồm những cái cơ bản có sẵn như: **typography, forms, buttons, tables, navigation, modals, image carousels** và nhiều thứ khác. Nó cũng có nhiều **Component, Javascript** hỗ trợ cho việc thiết kế **Reponsive** của bạn dễ dàng, thuận tiện và nhanh chóng hơn.



Hiện nay **Bootstrap** là một trong những framework được sử dụng nhiều nhất trên thế giới để tạo ra các **Responsive Website**. **Bootstrap** đã tạo ra một tiêu chuẩn riêng, và rất được các lập trình viên ưu chuộng. Về cơ bản **Bootstrap** có 3 ưu điểm:

**Dễ sử dụng:** Vì **Bootstrap** được xây dựng trên **HTML, CSS & Javascript**.

**Responsive**: **Bootstrap** đã xây dựng sẵn các **"Responsive Css"** tương thích với các thiết bị khác nhau, vì vậy bạn chỉ cần học cách sử dụng chúng. Tính năng này giúp tiết kiệm rất nhiều thời gian cho các người dùng khi tạo ra các **Website** thân thiện.

**Tương thích với các trình duyệt**: Nó tương thích với tất cả các trình duyệt (**Chrome, Firefox, Internet Explorer, Safari, Opera**). Tuy nhiên, với **IE**, **Bootstrap 4** chỉ hỗ trợ từ **IE10** trở lên.

Lịch sử của Bootstrap:

**Bootstrap**, ban đầu có tên là **Twitter Blueprint**, được phát triển bởi **Mark Otto** và **Jacob Thornton** tại **Twitter** như một khuôn khổ để khuyến khích sự nhất quán trên các công cụ nội bộ. Trước khi **Bootstrap**, các thư viện khác nhau được sử dụng để phát triển giao diện, dẫn đến sự thiếu nhất quán và gánh nặng bảo trì cao.

Sau một vài tháng phát triển bởi một nhóm nhỏ, nhiều nhà phát triển tại **Twitter** đã bắt đầu đóng góp cho dự án như một phần của Tuần lễ **Hack**, một tuần theo phong cách **hackathon** dành cho nhóm phát triển **Twitter**. Nó được đổi tên từ **Twitter Blueprint** thành **Bootstrap**, và được phát hành như một dự án nguồn mở vào ngày 19 tháng 8 năm 2011. Nó tiếp tục được duy trì bởi **Mark Otto, Jacob Thornton** và một nhóm nhỏ các nhà phát triển cốt lõi, cũng như một cộng đồng lớn người dùng.

Bootstrap Versions:

|  |  |  |
| --- | --- | --- |
| **Phiên bản** | **Ngày phát hành** | **Nội dung chính** |
| 1.0 | 19/08/2011 | Phiên bản đầu tiên của **Bootstrap**, chưa hô trợ **Mobile**. |
| 2.0 | 31/01/2012 | Bổ xung hệ thống **Grid-Layout** 12 cột. Thêm một số thành phần (component) mới. Và thay đổi một vài thành phần sẵn có. Vẫn chưa hỗ trợ **Mobile**. |
| 3.0 | 19/08/2013 | Các thành phần được thiết kế lại theo phong cách thiết kế phẳng (flat design). Và lần đầu tiên hỗ trợ các thiết bị **Mobile**. |
| 4.0 | 19/01/2018 | **Bootstrap 4** gần như viết lại hoàn toàn từ **Bootstrap 3**, và được đánh giá là dễ sử dụng hơn rất nhiều so với phiên bản trước. |

1. Ngôn ngữ Javascript.

Với HTML chỉ biểu diễn thông tin chứ chưa phải là các trang Web động có khả năng đáp ứng các sự kiện từ phía người dùng.

Hãng Netscape đã đưa ra ngôn ngữ script có tên là LiveScript để thực hiện chức năng này. Sau đó đổi tên thành JavaScript để tận dụng tính đại chúng của ngôn ngữ lập trình Java.

JavaScript là ngôn ngữ dưới dạng script có thể gắn với các file HTML. Được trình duyệt diễn dịch, trình duyệt đọc JavaScript dưới dạng mã nguồn.

JavaScript là ngôn ngữ dựa trên đối tượng, nghĩa là bao gồm nhiều kiểu đối tượng, ví dụ đối tượng Math với tất cả các chức năng toán học. Nhưng JavaScript không là ngôn ngữ hướng đối tượng như C++/Java.

JavaScript là một ngôn ngữ kịch bản (scripting language) được dùng để tạo các script ở máy client (client-side script) và máy server (server-side script). Các script ở máy client được thực thi tại trình duyệt, các script ở máy server được thực hiện trên server. Chương này sẽ giới thiệu cho chúng ta về ngôn ngữ Javascript, và cách chèn một script vào trong tài liệu HTML.

HTML *lúc đầu* được phát triển như là một định dạng của tài liệu có thể chuyển dữ liệu trên Internet Tuy nhiên, không lâu sau đó, trọng tâm của HTML nặng tính hàn lâm và khoa học dần chuyển hướng sang người dùng thường nhật vì ngày nay người dùng xem Internet như là một nguồn thông tin và giải trí. Các trang Web ngày càng mang tính sáng tạo và đẹp mắt hơn nhằm thu hút nhiều người dùng hơn. Nhưng thực chất kiểu dáng và nội dung bên trong vẫn không thay đổi. Và người dùng hầu như không thể điều khiển trên trang Web mỗi khi nó được hiển thị.

Javascript được phát triển như là một giải pháp cho vấn đề nêu trên. Javascript là một ngôn ngữ kịch bản được Sun Microsystems và Netscape phát triển. Nó được dùng để tạo các trang Web động và tương tác trên Internet. Đối với những người phát triển HTML, Javascript rất hữu ích trong việc xây dựng các hệ thống HTML có thể tương tác với người dùng.

JavaScript là gì?

Sun Microsystems đã viết ra một ngôn ngữ phức tạp và mạnh mẽ mà chúng ta đã biết đó là ngôn ngữ Java. Mặc dù Java có tính khả dụng cao nhưng nó lại phù hợp nhất đối với các nhà lập trình có kinh nghiệm và cho các công việc phức tạp hơn. Netscape Communications nhận thấy nhu cầu cần một ngôn ngữ thiết kế web có khả năng tương tác vớI ngườI sử dụng hay vớI các Java Applet, dễ sử dụng ngay cả vớI những ngườI lập trình ít kinh nghiệm.

LiveScript là một ngôn ngữ mớI chỉ ở dạng phác thảo, tuy nhiên nó hấp dẫn ngườI sử dụng vì hứa hẹn sẽ đáp ứng tốt những yêu cầu trên. LiveScript được thiết kế theo tinh thần của nhiều ngôn ngữ script đơn giản nhưng nó lại có tính khả dụng cao được thiết kế đặc biệt để xây dựng các trang Web (chẳng hạn như HTML và các form tương tác). Để giúp ‘bán chạy’ ngôn ngữ mới này,*Netscape hợp tác với Sun cho ra đời ngôn ngữ Javascript. Trên thực tế, Microsoft là người tiên phong triển khai phiên bản của Javascript (còn có tên là Jscript), nhưng họ không sử dụng các đặc tả chính thức của Javascript.*

Mục tiêu của JavaScript là nhằm cung cấp cho các nhà phát triển Web một số khả năng và quyền điều khiển chức năng cho trang Web. Mã Javascript có khả năng nhúng trong tài liệu HTML để điều khiển nội dung của trang Web và kiểm tra sự hợp lệ của dữ liệu mà người dùng nhập vào. Khi một trang hiển thị trong trình duyệt, các câu lệnh được trình duyệt thông dịch và thực thi.

* JavaScript

JavaScript là một ngôn ngữ kịch bản dựa trên đối tượng nhằm phát triển các ứng dụng Internet chạy trên phía client và phía server.

Các ứng dụng client được chạy trong một trình duyệt như Netscape Navigator hoặc Internet Explorer, và các ứng dụng server chạy trên một Web server như Microsoft’s Internet Information Server hoặc Netscape Enterprise Server.

Hiệu ứng và quy tắc JavaScript

JavaScript là một ngôn ngữ lập trình được nhúng được trong các trang HTML. JavaScript nâng cao tính động và khả năng tương tác cho web-site bằng cách sử dụng các hiệu ứng của nó như thực hiện các phép tính, kiểm tra form, viết các trò chơi, bổ sung các hiệu ứng đặc biệt, tuỳ biến các chọn lựa đồ hoạ, tạo ra các mật khẩu bảo mật và hơn thế nữa.

Chúng ta có thể sử dụng JavaScript để:

**Tương tác với người dùng**.Chúng ta có thể viết mã để đáp lạI các sự kiện. Các sự này sẽ có thể phát sinh bởi người dùng - - nhấp chuột hay được phát sinh từ hệ thống - - định lại kích thước của trang và v.v.

**Thay đổi nội dung động**. Mã JavaScript có thể dùng để thay đổi nội dung và vị trí các phần tử một cách động trên một trang nhằm đáp lại sự tương tác với người dùng.

**Kiểm tra tính hợp lệ dữ liệu**. Chúng ta có thể viết mã nhằm kiểm tra tính hợp lệ của dữ liệu do người dùng nhập vào trước khi nó được gửi lên Web server để xử lý.

Giống như các ngôn ngữ khác, JavaScript cũng tuân thủ một số quy tắc ngữ pháp căn bản. Việc nắm vững các quy tắc ngữ pháp này có thể giúp ta đọc được script và tự viết các script không bị lỗi.

Một số trong các luật này bao gồm:

**Dùng Caps.**JavaScript phân biệt chữ hoa chữ thường

**Dùng Pairs.**Trong JavaScript, luôn luôn có cặp ký hiệu mở và đóng. Lỗi sẽ xuất hiện khi bỏ sót hoặc đặt sai một trong hai ký hiệu này.

**Dùng Spaces (các ký tự trắng).**Như HTML, JavaScript thường bỏ qua ký tự trắng. Trong JavaScript, ta có thể thêm vào các ký tự trắng hoặc các tab giúp cho ta dễ dàng đọc hay sửa các file script.

**Dùng Chú thích (Comments).**Các chú thích giúp ta ghi chú về chức năng của đoạn script, thờI gian và ngườI tạo ra đoạn script.

Mặc dù cả client-side JavaScript và server-side JavaScript đều dựa trên một ngôn ngữ nền tảng như nhau, nhưng mỗI loạI còn có thêm những tính năng chuyên biệt phù hợp với môi trường mà nó chạy. Nghĩa là, client-side JavaScript bao gồm các đối tượng được định nghĩa sẵn chỉ có thể sử dụng trên trình duyêt, Server-side JavaScript bao gồm các đối tượng và các hàm được định nghĩa sẵn chỉ có thể sử dùng trong các ứng dụng phía server (server-side applications)

Các công cụ JavaScript và IDE, và môi trường thực thi

Các công cụ sinh mã JavaScript và IDE giúp tạo ra mã JavaScript rất hữu hiệu. . Các công cụ này còn giúp ta nhanh chóng phát triển website của mình.

Một vài công cụ JavaScript và IDE được đề cập dưới đây:

**Dialog Box.**Công cụ này tự động tạo mã để sinh ra các hộp thoại tuỳ biến trên các trình duyệt khác nhau (*alert, confirm, prompt, v.v.*) mang lại kiểu dáng chuyên nghiệp cho website.

**Pop-up Menu builder.**Chỉ cần đưa vào các lựa chọn, công cụ này sẽ tự động tạo ra các pop-up menu trên các trình duyệt khác nhau.

**Remotes.**Tự động sinh mã để mở ra một cửa sổ popup.

Như chúng ta đã biết, JavaScript có thể được chạy trên máy khách (client) và máy chủ (server). Bên phía máy khách, trình duyệt sẽ thực thi mã lệnh javascript trong trang web khi mở nó. Bên phía máy chủ, mã javascript sẽ được thực thi tại máy chủ và do máy chủ thực hiện.

**Client-side Java Script**

Khi máy client yêu cầu một trang HTML, server sẽ kiểm tra xem trang đó có chứa script hay không. Nếu nó chứa các client-side script, server sẽ chuyển toàn bộ tài liệu bao gồm mã lệnh JavaScript và nội dung HTML của nó cho trình duyệt. Khi trình duyệt nhận được tài liệu đó, nó thực thi các mã lệnh HTML và JavaScript mà không cần bất kì sự tương tác nào với server.

**JavaScript trên Web Server**

Chúng ta có thể nhúng các lệnh JavaScript chạy trên server (server-side script) vào trong tài liệu HTML. Quá trình tạo ra các ứng dụng server-side là một quá trình gồm hai giai đọan.

Các trang HTML có chứa các câu lệnh JavaScript của cả client-side và server-side đều được tạo ra cùng với các file JavaScript. Tất cả các file này sẽ được biên dịch thành dạng mã thực thi được là bytecode.

Khi trình duyệt yêu cầu trang HTML, run-time engine sẽ thực thi mã lệnh server-side JavaScript rồi trả trang HTML về cho trình duyệt.

Một số công dụng của script server-side bao gồm:

Kết nối vào các cơ sở dữ liệu

Chia sẻ thông tin cho những người dùng của một ứng dụng

Truy cập vào hệ thống file trên server

Nhúng JavaScript vào trong trang Web

Chúng ta có thẻ chèn các lệnh JavaScritp vào trong một tài liệu HTML theo những cách sau đây:

Nhúng các câu lệnh trực tiếp vào trong tài liệu bằng cách sử dụng thẻ <SCRIPT>

Liên kết file nguồn JavaScript với tài liệu HTML

Đặt các biểu thức JavaScript làm giá trị cho thuộc tính của thẻ HTML.

Dùng như trình xử lý sự kiện trong các thẻ HTML.

Chúng ta có xem chi tiết một số ví dụ sau đây:

**Dùng thẻ SCRIPT**

Mã JavaScript cụ thể được nhúng vào trong tài liệu HTML bằng thẻ SCRIPT. Chúng ta có thể nhúng nhiều script vào trong cùng một tài liệu, mỗi script nằm trong một thẻ SCRIPT. Khi trình duyệt gặp phải một thẻ <SCRIPT> nào đó, nó sẽ đọc từng dòng một cho đến khi gặp thẻ đóng </SCRIPT>. Tiếp đến nó sẽ kiểm tra lỗi trong các câu lệnh JavaScript. Nếu gặp phải lỗi, nó sẽ cho hiển thị lỗi đó trong chuỗi các hộp cảnh báo (alert boxes) lên màn hình. Nếu không có lỗi, các câu lệnh sẽ được biên dịch sao cho máy tính có thể hiểu được lệnh đó.

1. ASP.Net

ASP.NET là một nền tảng ứng dụng web (web application framework) được phát triển và cung cấp bởi [Microsoft](https://vi.wikipedia.org/wiki/Microsoft), cho phép những người lập trình tạo ra những trang web động, những [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web) và những [dịch vụ web](https://vi.wikipedia.org/wiki/D%E1%BB%8Bch_v%E1%BB%A5_web). Lần đầu tiên được đưa ra thị trường vào tháng 1 năm 2002 cùng với phiên bản 1.0 của.NET framework, là công nghệ nối tiếp của Microsoft's Active Server Pages(ASP). ASP.NET được biên dịch dưới dạng [Common Language Runtime](https://vi.wikipedia.org/w/index.php?title=Common_Language_Runtime&action=edit&redlink=1) (CLR), cho phép những người lập trình viết mã ASP.NET với bất kỳ ngôn ngữ nào được hỗ trợ bởi.NET language.

Sau khi phát hành phiên bản [Internet Information Service](https://vi.wikipedia.org/w/index.php?title=Internet_Information_Service&action=edit&redlink=1) 4.0 vào năm 1997, hãng Microsoft bắt đầu nghiên cứu một mô hình ứng dụng web để giải quyết những bất tiện của ASP, đặc biệt là việc tách riêng biệt phần thể hiện và phần nội dung cũng như cách viết mã rõ ràng hơn. [Mark Anders](https://vi.wikipedia.org/w/index.php?title=Mark_Anders&action=edit&redlink=1" \o "Mark Anders (trang chưa được viết)), quản lý của nhóm IIS và [Scott Guthrie](https://vi.wikipedia.org/w/index.php?title=Scott_Guthrie&action=edit&redlink=1" \o "Scott Guthrie (trang chưa được viết)), gia nhập Microsoft vào năm 1997 sau khi tốt nghiệp [Đại học Duke](https://vi.wikipedia.org/wiki/%C4%90%E1%BA%A1i_h%E1%BB%8Dc_Duke), được giao nhiệm vụ định hình mô hình cần phát triển. Những thiết kế ban đầu được thực hiện trong vòng 2 tháng bởi Anders và Guthrie, Guthrie đã viết mã prototype đầu tiên trong khoảng thời gian nghỉ lễ [Giáng sinh](https://vi.wikipedia.org/wiki/L%E1%BB%85_Gi%C3%A1ng_Sinh) năm 1997.

Những trang ASP.NET, được biết đến như những web form, là khối chính trong phát triển ứng dụng. Những web form được chứa trong những file có phần mở rộng ASPX; những nhà phát triển có thể đặt nội dung tĩnh hoặc động vào trang aspx dưới dạng server-side Web Control và User Control. Ngoài ra, có thể viết mã bằng cách chèn <% -- mã cần viết -- %> vào trang web giống như những công nghệ phát triển web khác [PHP](https://vi.wikipedia.org/wiki/PHP" \o "PHP), [JSP](https://vi.wikipedia.org/wiki/JSP) và [ASP](https://vi.wikipedia.org/wiki/Active_Server_Pages), nhưng những công nghệ nào không hỗ trợ [data binding](https://vi.wikipedia.org/w/index.php?title=Data_binding&action=edit&redlink=1) khi nó phát sinh nội dung trang web.

1. ADO.Net

ADO.NET là một tập các lớp nằm trong bộ thư viện lớp cơ sở của .NET Framework, cho phép các ứng dụng windows (như C#, VB.NET) hay ứng dụng web (như ASP.NET) thao tác dễ dàng với các nguồn dữ liệu.

Mục tiêu chính của ADO.NET là:

Cung cấp các lớp để thao tác CSDL trong cả hai môi trường là phi kết nối (Disconected data) và kết nối (Connected data).

Tích hợp chặt chẽ với XML (Extensible Markup Language)

Tương tác với nhiều nguồn dữ liệu thông qua mô tả dữ liệu chung.

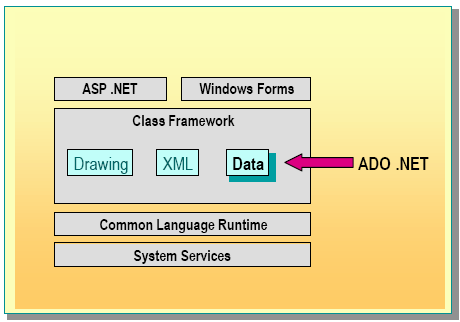
Tối ưu truy cập nguồn dữ liệu (OLE DB & SQL server).

Làm việc trên môi trường Internet.

Các lớp của ADO.NET được đặt trong Namespace là System.Data/ System.Data.oledb

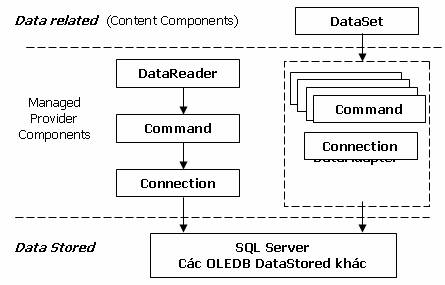
ADO.NET bao gồm 2 Provider (2 bộ thư viện thường dùng) để thao tác với các CSDL là: OLE DB Provider (nằm trong System.Data.OLEDB) dùng để truy xuất đến bất kỳ CSDL nào có hỗ trợ OLEDB; SQL Provider (nằm trong System.Data.SQLClient) chuyên dùng để truy xuất đến CSDL SQL Server (Không qua OLE DB nên nhanh hơn). Hiện nay, các hãng thứ ba còn cung cấp các Provider khác như: MySQL, Oracle… provider để cho phép ứng dụng .NET truy xuất đến các cơ sở dữ liệu không phải của Microsoft khác.

Vị trí của ADO.NET trong kiến trúc của .NET Framework



Vị trí của ADO.NET trong kiến trúc của .net Framework

Từ kiến trúc ta thấy rằng: ADO.NET là một thành phần nội tại (Instrict) của .NET framework, do vậy nó có thể được sử dụng trong tất cả các ngôn ngữ hỗ trợ .NET như C#, VB.NET… mà không có sự khác biệt nào (Tức là các chức năng cũng như cách sử dụng hoàn toàn giống nhau).



Hình biểu diễn kiến trúc ADO.NET

 Kiến trúc ADO.NET có thể chia làm 2 phần chính:

Managed Provider Component: bao gồm các đối tượng như DataAdapter, DataReader,… giữ nhiệm vụ làm việc trực tiếp với dữ liệu như database, file,…

Content Component: bao gồm các đối tượng như DataSet, DataTable,… đại diện cho dữ liệu thực sự cần làm việc. DataReader là đối tượng mới, giúp truy cập dữ liệu nhanh chóng nhưng forward-only và read-only giống như ADO RecordSet sử dụng Server cursor, OpenFowardOnly và LockReadOnly.

DataSet cũng là một đối tượng mới, không chỉ là dữ liệu, DataSet có thể coi là một bản sao gọn nhẹ của CSDL trong bộ nhớ với nhiều bảng và các mối quan hệ. DataAdapter là đối tượng kết nối giữa DataSet và CSDL, nó bao gồm 2 đối tượng Connection và Command để cung cấp dữ liệu cho DataSet cũng như cập nhật dữ liệu từ DataSet xuống CSDL.

 Trước khi đi vào học cụ thể các đối tượng của ADO.NET chúng ta cùng xem qua một ví dụ HelloWorld với ADO.NET qua đó bạn sẽ thấy được công việc cần thực hiện khi thao tác với database(ở ví dụ này mình dùng với SQLExpress).

# **CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

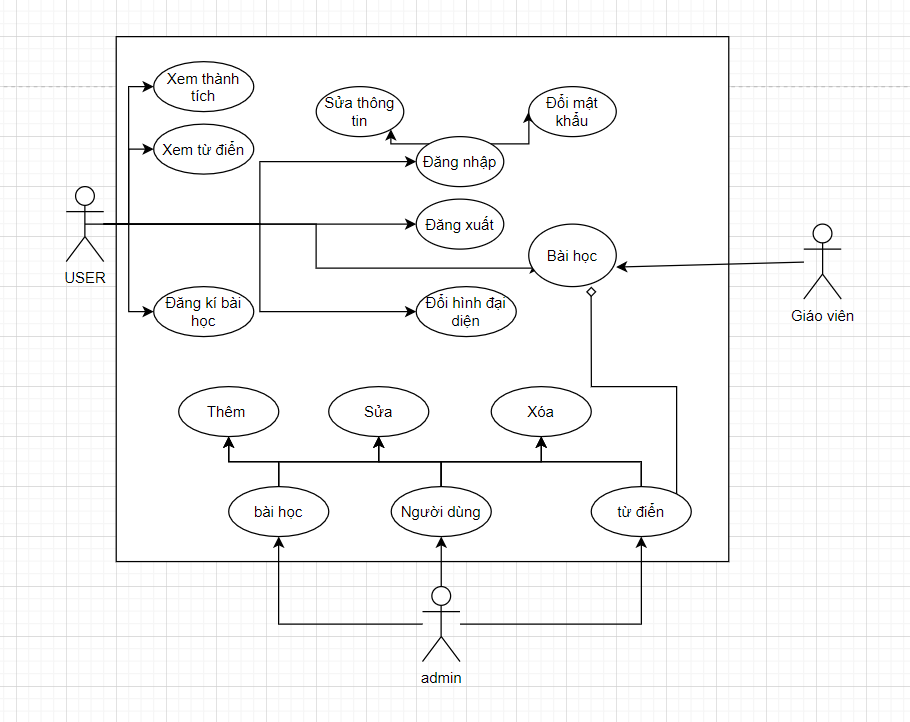
1. **Giới thiệu về phần mềm**

* Tên phần mềm: Hệ thống Phần mềm hổ trợ học tiếng Nhật
* Các chức năng: Học tâp, Xem kết quả, Tra từ điển

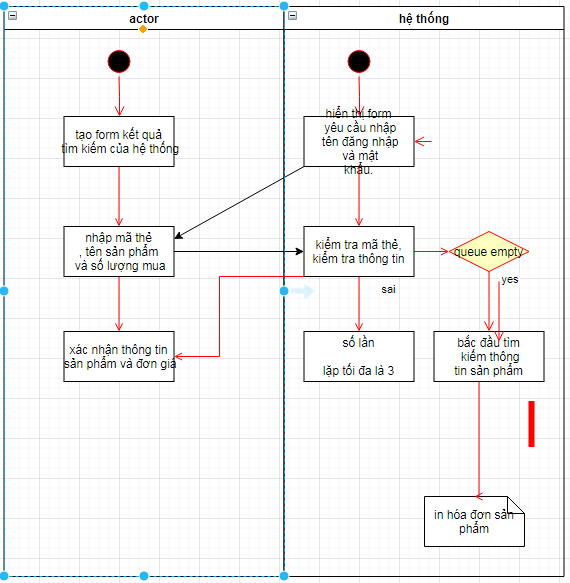
1. **Tác nhân**

* Admin
* Giáo viên
* Người dùng

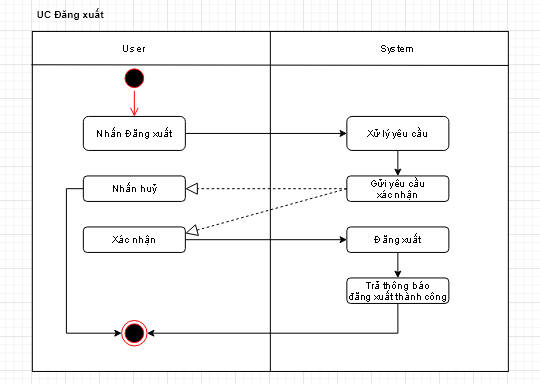
1. **Use Case**



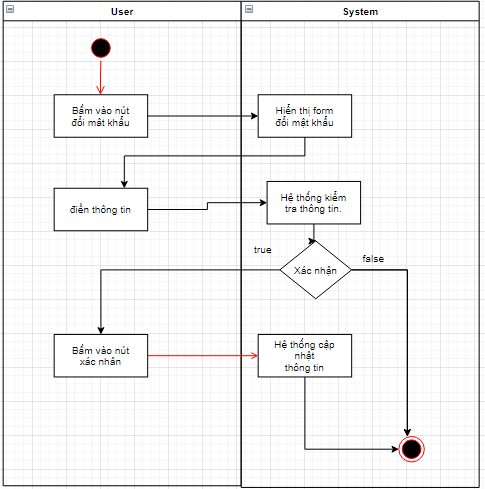
1. ***Biểu đồ hoạt độ (Activity Diagram )*** 
   1. ***Đăng nhập***



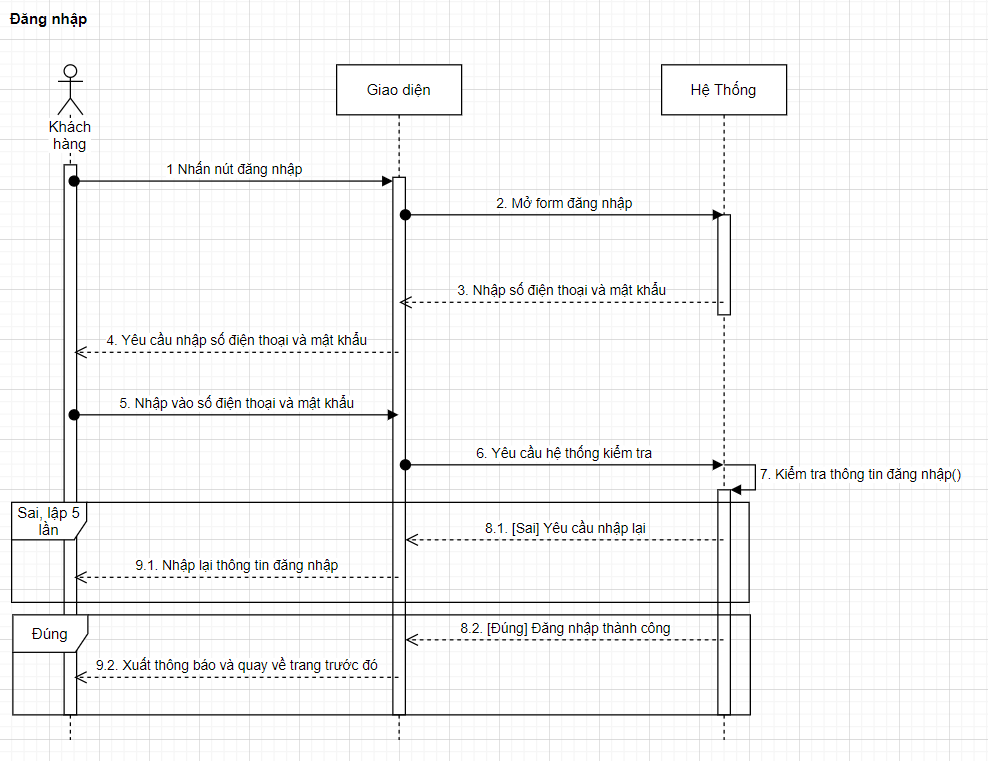
* 1. ***Đăng xuất***



* 1. ***Đổi mật khẩu***



1. ***Biểu đồ tuần tự (Seqence Diagram)*** 
   1. ***Đăng nhập***

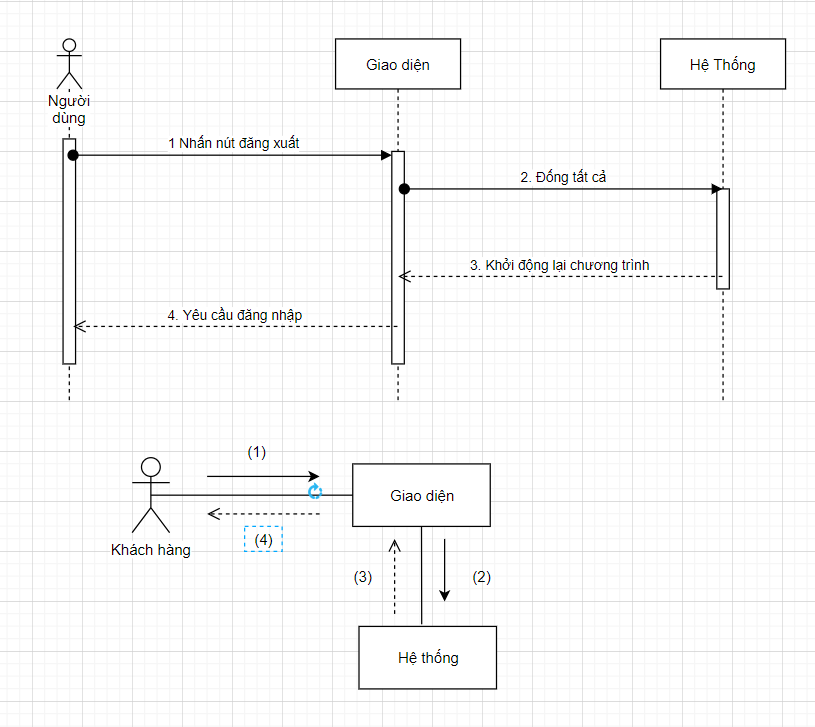


/Nhân viên



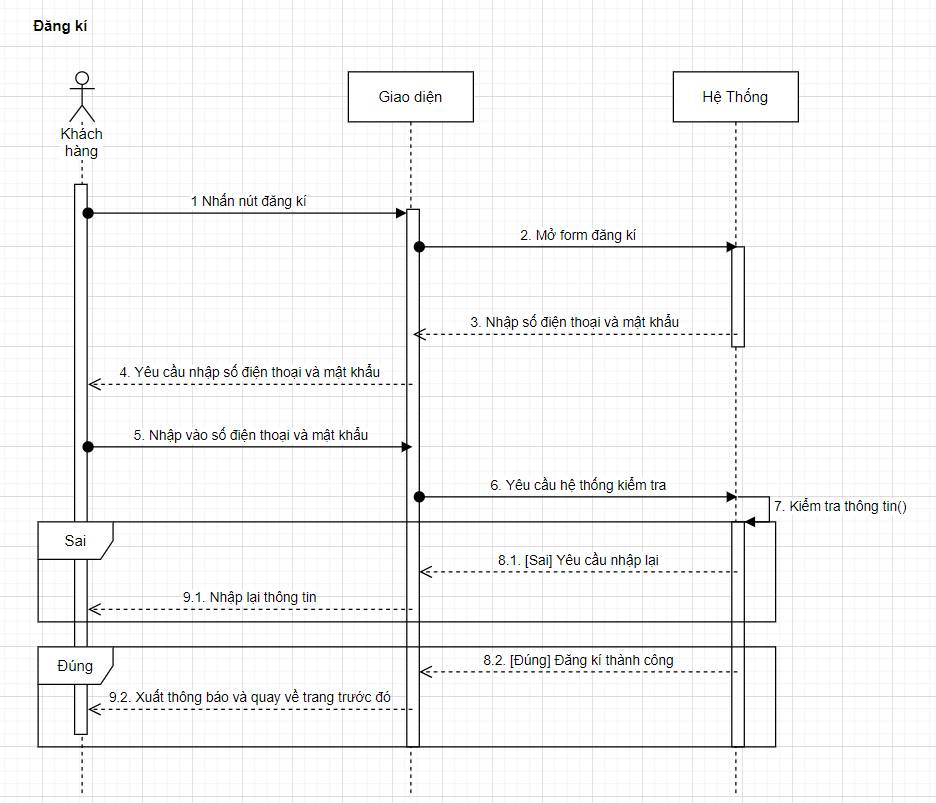
/Nhân viên

* 1. ***Đăng xuất***



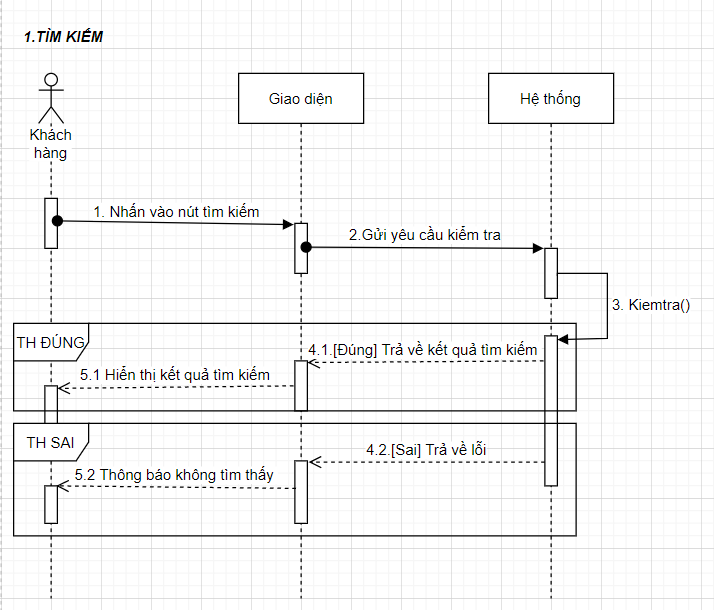
/Nhân viên

* 1. ***Đăng kí***

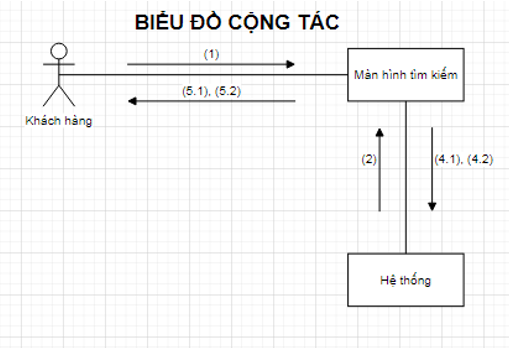




* 1. ***Tìm kiếm***

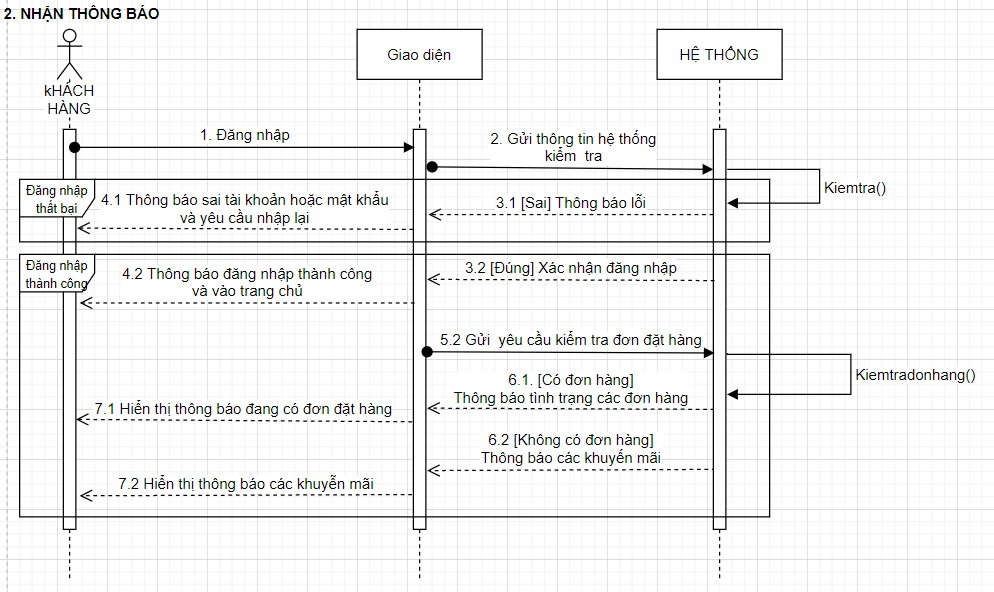


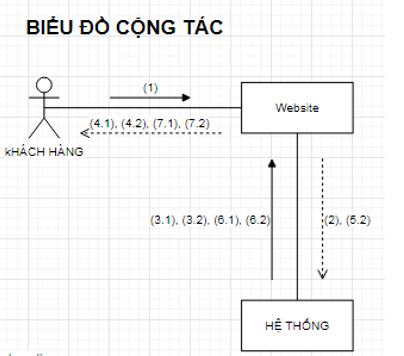
/Nhân viên



/Nhân viên

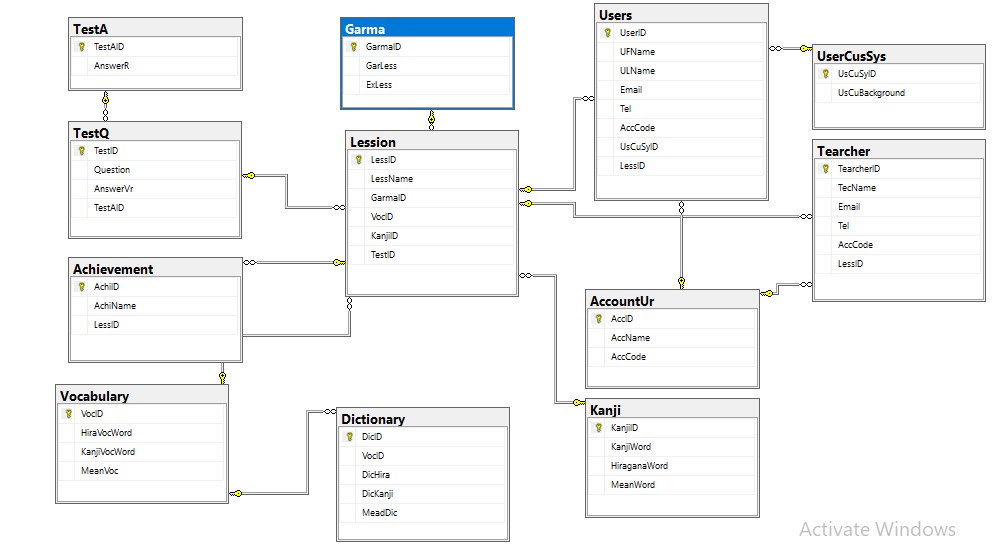
* 1. ***Nhận thông báo***





1. ***Thiết kế lớp***

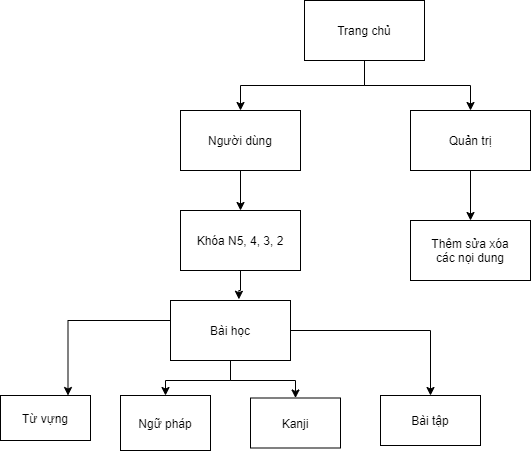
Biểu đồ lớp



# **CHƯƠNG 4: GIAO DIỆN CHƯƠNG TRÌNH**

1. **Cấu trúc Giao diện**

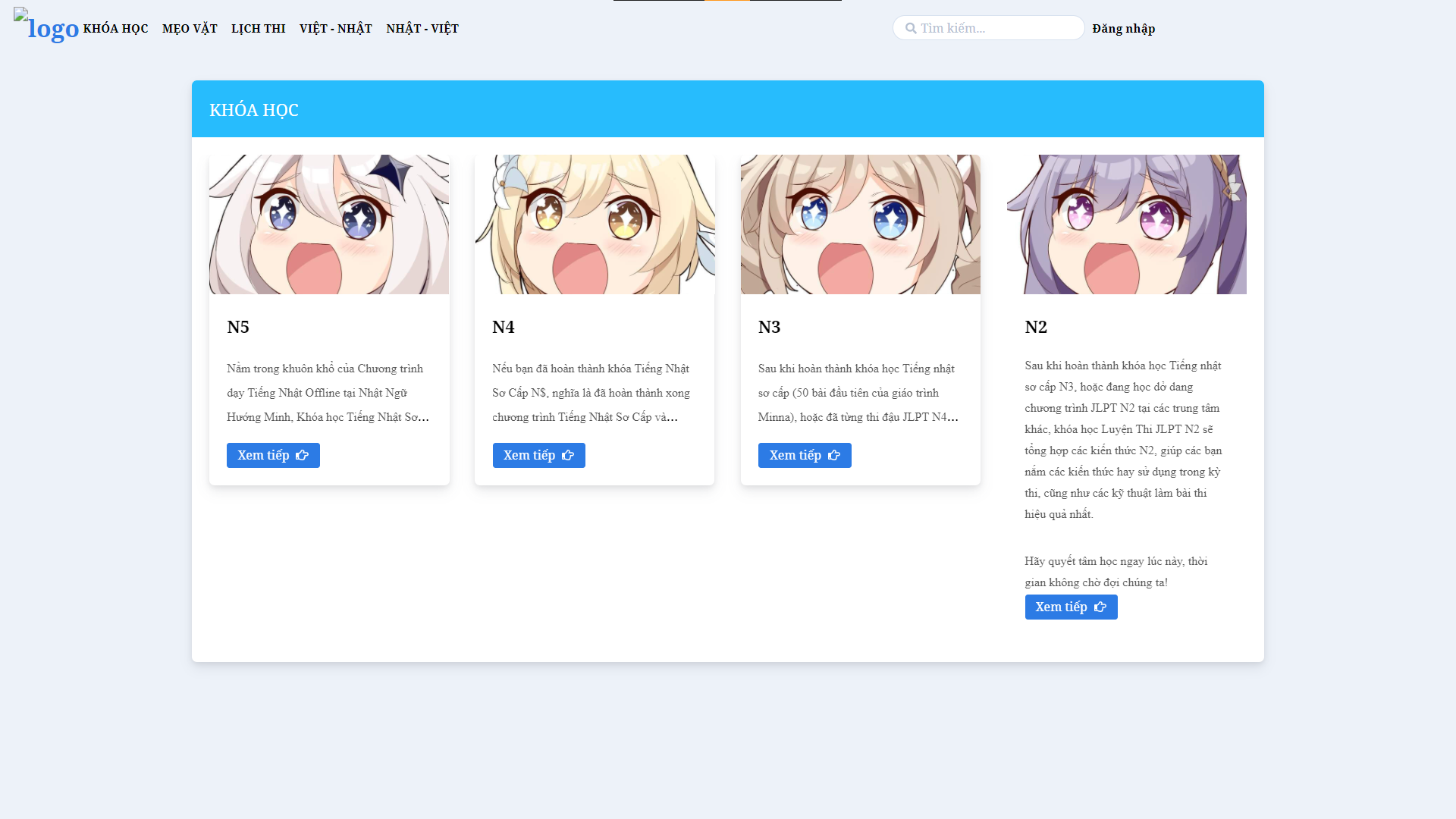
Giao diện được phân thành các nhánh như sau

****

* Tất cả các trang con của hệ thống website đều được kết nối xuống cơ sở dử liệu tại Server.

1. **Đặc tả Giao diện**
   1. **Phần giao diện người dùng**
2. **Trang chủ**

* Mục đích: Là giao diện tổng quan của Website
* Tại đây, người xem có thể đến được các trang bài học

****

1. **Các khóa học, các bài học (Các khóa học có cấu trúc gióng nhau)**

* Website hiện tại có 4 khóa học là N5, 4, 3, 2. Tuy nhiên, hiện tại chỉ có N5 là có dử liệu mẫu, các trang N4, 3, 2 không được thêm dử liệu nhưng chúng có cùng một dạng thiết kế.
* Các bài học trong khóa học được phân chia ra theo: Từ vựng, ngữ pháp, kanji. Riêng phần bài tập được gợp chung lại của các bài học.

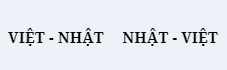


* Các phân loại được đưa sang bên cạnh trái của website:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Các bài học về Từ vựng | Ngữ pháp | Kanji | Riêng bài tập là tổng hợp nên được tách riêng ra |

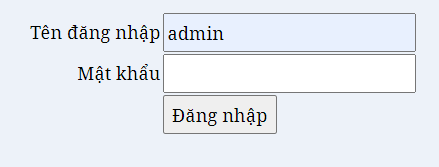
1. **Từ điển**

* Từ điển được phân thành hai loại là tra cứu từ tiếng Việt sang tiếng Nhật và từ tiếng Nhật sang tiếng Việt
* Dữ liệu của từ điển chính là từ vựng của các bài học thuộc các khóa học được đưa vào và trích xuất ra.

****

|  |  |
| --- | --- |
| **Việt – Nhật** | **Nhật – Việt** |
|  |  |

* 1. **Phần giao diện quản trị**
* Đây là nơi giúp thêm, sửa và xóa các dữ liệu bài học như từ vựng, ngữ pháp, kanji và bài tập
* Chỉ có quyền admin được phép đăng nhập để thêm, sửa và xóa các nội dung.

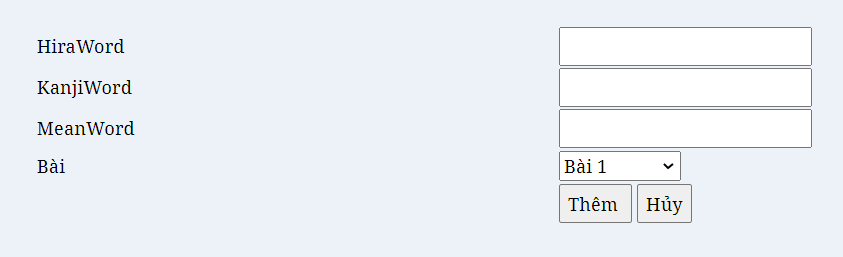
****

* Giao diện thêm, sửa và xóa các nội dung gióng nhau

****

1. **Thêm**

* Sau khi nhấn vào nút *thêm bài,* giao diện thêm sẽ xuất hiện.
* Đây là nơi giúp quản trị viên có thể thêm các từ vựng, ngữ pháp, bài học, kanji kể cả bài kiểm tra vào.



* Tùy theo là bài học hay từ vựng hay kanji mà sẽ có các trang thêm bài tương ứng.

1. **Sửa**

* Khi nhấp vào biểu tượng , hệ thống sẽ cho phép sửa nội dung lại
* Giao diện sửa sẽ sửa trực tiếp trên bản lưu nội dung



* Sau khi nhấp vào biểu tượng sửa hệ thống cho ta sửa chữ lại và sẽ xuất hiện 2 biểu tượng lưu  và hủy . Nếu nhấn lưu hệ thống sẽ cập nhật giá trị mới còn khi nhấn hũy hệ thống sẽ quây về giá trị ban đầu.

1. **Xóa**

* Khi nhấp vào biểu tượng xóa , hệ thống sẽ cho phép xóa trường dữ liệu đã được chọn tương ứng, sau đó làm tươi lại dử liệu.

# **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

1. **Kết quả đạt được:**

* Thiết kế và xây dựng thành công website Hổ trợ học tiếng Nhật, đáp ứng được yêu cầu của người dùng.
* Tìm hiểu và nắm cơ bản được những công cụ để Thiết kế website như: Visual studio code, Adobe Dreamweaver CS6 và các ngôn ngữ lập trình Website như: HTML, Javascript, CSS.
* Lập trình và sữ dụng thành thạo các công cụ trên SQL Server, ASP.Net và ADO.Net
* Giao diện website được thiết kế đơn giản, thân thiện và dễ sử dụng

1. **Việc chưa làm được:**

* Việc biểu diễn các thông tin trên website chưa được linh hoạt.
* Phân tích thiết kế chưa được hoàn chỉnh.
* Nhiều chức năng còn thiếu và chưa sử dụng được.
* Các chữ, các khung còn bị chạy, chưa đạt tính thẩm mĩ cao

1. **Hướng phát triển của đề tài**

Vì thời gian không cho phép và kiến thức về thiết kế cũng như lập trình nên website còn hạn chế cũng như các vấn đề chưa tìm hiểu rõ và kỹ hơn về đề tài này. Vì thế đề tài cũng còn có nhiều thiếu sót, mong hội đồng giúp đỡ chúng em nhiều hơn và em sẽ tiếp tục tìm hiểu sâu hơn về công nghệ này, nhằm mục đích xây dựng hệ thống webside hoàng chỉnh:

* Giao diện đẹp, thân thiện với người dùng, có nhiều chức năng và tương tác tốt hơn với người dùng.
* Thiết kế, xây dựng và lập trình bổ sung các tính năng còn thiếu hay chưa thật sự hoàn chỉnh trong quá trình sử dụng.

# **TÀI LIỆU THAM KHẢO**

* Tài liệu bài giảng môn “Lập trình web” của thầy Nguyễn Hữu Vĩnh.
* Website Hiepsiit: <http://www.hiepsiit.com>
* Website W3schools: <https://www.w3schools.com/>
* Website tham khảo:
* <http://www.hiepsiit.com>
* https://dungmori.com/
* Các nguồn tài nguyên được upload tại github: https://github.com/aokidai/baiduthinckh