**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**---------------------------------------**

BÁO CÁO TL, BTL, ĐA/DA THUỘC HỌC PHẦN: THỰC TẬP CƠ SỞ NGÀNH

**THIẾT KẾ TRANG WEB VỀ TOUR DU LỊCH**

|  |  |
| --- | --- |
| **GVHD:** | **Ths Nguyễn Xuân Hoàng** |
| **Sinh viên:** | **Nguyễn Thùy Trang** |
|  | **Nguyễn Thị Hồng Ngọc** |
|  | **Đỗ Thị Lan** |
|  | **Nguyễn Văn Tuân** |
|  | **Đỗ Mạnh Dũng** |
| **Lớp – Khóa:** | **20232IT6055002 – K17** |

**Hà Nội – Năm 2024**

|  |
| --- |
|  |
| **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **KHOA CÔNG NGHỆ THÔNG TIN**  **---------------------------------------** |
|  |
| BÁO CÁO TL, BTL, ĐA/DA THUỘC HỌC PHẦN: THỰC TẬP CƠ SỞ NGÀNH |
|  |
| **THIẾT KẾ TRANG WEB VỀ TOUR DU LỊCH** |
|  |
| |  |  | | --- | --- | | **GVHD:** | **Ths Nguyễn Xuân Hoàng** | | **Sinh viên:** | **Nguyễn Thùy Trang** | |  | **Nguyễn Thị Hồng Ngọc** | |  | **Đỗ Thị Lan** | |  | **Nguyễn Văn Tuân** | |  | **Đỗ Mạnh Dũng** | | **Lớp – Khóa:** | **20232IT6055002 – K17** | |
|  |
| **Hà Nội – Năm 2024** |

*Ghi chú : TL, BTL, ĐA/DA (Tiểu luận, Bài tập lớn, Đồ án/Dự án)*

**MỤC LỤC**

**DANH MỤC HÌNH ẢNH**

**MỞ ĐẦU**

Công nghệ thông tin trong giai đoạn hiện nay đang phát triển mạnh mẽ ở mọi lĩnh vực hoạt động khắp nơi trên thế giới. Điều xảy ra trong vòng vài năm qua ở Việt Nam là sự đầu tư lớn và công nghệ. Tin học đã và đang là một trong những vấn đề không thể thiết đối với bất kỳ doanh nghiệp nào. Đặc biệt tin học ngày càng giữ vai trò quan trọng trong vấn đề quản lý tại các tổ chức nhất là tại các lĩnh vực thu thập thông tin và kinh doanh.

Là những sinh viên ngành Hệ thống thông tin của trường Đại học Công nghiệp Hà Nội, chúng em xin chân thành cảm ơn nhà trường đã đưa ra chương trình giảng dạy phù hợp và bổ ích cho chúng em. Đặc biệt, chúng em xin gửi lời cảm ơn sâu sắc đến giảng viên bộ môn *“Thực tập cơ sở ngành”* - Thầy Nguyễn Xuân Hoàng đã giúp đỡ chúng em trong thời gian qua. Sau 3 kỳ học tập và rèn luyện tại nhà trường, chúng em đã được sử dụng những kiến thức đã được học tập, tìm hiểu, trau dồi để thực hiện bài tập lớn môn *“Thực tập cơ sở ngành”.*

Với sự phát triển và phổ biến rộng rãi của công nghệ là một trong những điểm sáng của nền kinh tế những năm gần đây. Việc xây dựng các websites giới thiệu về các lĩnh vực trong đời sống là cần thiết, bởi vì nhu cầu du lịch, mua sắm,... của chúng ta ngày càng đa dạng hơn. Với những thao tác đơn giản trên máy có nối mạng Internet bạn sẽ có thể tìm hiểu bất kỳ thông tin gì mình cần mà không phải mất nhiều thời gian.

Để góp phần đẩy mạnh sự phát triển của công nghệ cũng như giới thiệu về những địa điểm du lịch – một trong những ngành chiếm ưu thế hiện nay. Nhóm chúng em đã quyết định thực hiện đề tài “Thiết kế trang web về tour du lịch”. Tuy nhiên, do vốn kiến thức còn nhiều hạn chế, mặc dù chúng em đã cố gắng hết sức nhưng chắc chắn bài báo cáo khó có thể tránh khỏi những thiếu sót, mong thầy xem xét và góp ý để bài báo cáo của chúng em được hoàn thiện hơn!

          Chúng em xin chân thành cảm ơn!

# **CHƯƠNG 1: TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU**

## **Giới thiệu về ngôn ngữ HTML, CSS, Javascript**

### **Giới thiệu về ngôn ngữ HTML (Hypertext Markup Language)**

#### Giới thiệu

* HTML là từ viết tắt của Hypertext Markup Language, là sự kết hợp của Hypertext và Markup, hay còn được gọi là ngôn ngữ siêu văn bản. HTML có chức năng giúp người dùng xây dựng và cấu trúc các phần trong trang web hoặc ứng dụng, thường được sử dụng trong phân chia các đoạn văn, heading, link, blockquotes, …

#### Lịch sử của ngôn ngữ HTML

* + Cha đẻ của HTML là Tim Berners – Lee – nhà vật lý học, là người nghĩ ra ý tưởng dựa trên hệ thống hypertext trên nền internet.
  + Năm 1991, xuất bản phiên bản đầu tiên của HTML gồm 18 tag HTML.
  + Năm 1998, HTML phiên bản 4.01 ra đời.
  + Năm 2000, các phiên bản HTML được thay thế bằng XHTML.
  + Năm 2014, HTML được nâng cấp lên HTML5 với sự cải tiến rõ rệt.

#### Dấu hiệu nhận biết, cấu trúc thẻ, cấu trúc tài liệu HTML

* Dấu hiệu nhận biết HTML documents:
  + Files có kết thúc đuôi là .html hoặc htm và có thể xem bằng cách sử dụng bất kỳ trình duyệt web nào, chẳng hạn như Safari, Google chrome, Microsoft edge, …
  + Các trình duyệt sẽ đọc được files HTML và xuất bản nội dung lên internet, tại đây người dùng hoàn toàn dễ dàng đọc được nó.
* Cấu trúc thẻ HTML
  + Phần tử HTML hay còn được biết đến với tên Tag hay Entity hay thẻ. HTML là ngôn ngữ đánh dấu (markup-language) do đó có thể hiểu một phần tử HTML chính là một đoạn văn bản được đánh dấu để thể hiện theo một cách nào đó.
  + Một phần tử HTML luôn có thể được nhận ra bởi nó được bao quanh bởi cặp dấu < và >. Ví dụ: <body> hay <b>
  + Thẻ HTML được chia ra làm 02 loại thẻ: thẻ chứa dữ liệu và thẻ rỗng
* Thẻ chứa dữ liệu
  + Mỗi phần tử của HTML luôn bao gồm một cặp thẻ đi song song, một thẻ mở và một thẻ đóng, hai thẻ này còn được gọi là thẻ chứa vì chức năng của thẻ sẽ tác động vào các thành phần nó chứa bên trong thẻ đóng và thẻ mở.
  + Cấu trúc thẻ mở có dạng: **<tên\_thẻ>**
  + Còn thẻ đóng cũng tương tự nhưng có thêm dấu gạch chéo: **</tên\_thẻ>**. *Ví dụ:* <p> đây là nội dung của thẻ "p"</p>.
  + Trong thẻ HTML có thể có thêm các thuộc tính, thuộc tính của một phần tử có thể được biểu diễn ngay trong thẻ mở của phần tử.
  + Cú pháp thuộc tính của thẻ mở:

**<tên\_thẻthuộc\_tính\_1="giá\_trị\_1"thuộc\_tinh2="giá\_trị\_2">**

* + Trong thẻ mở của ngôn ngữ HTML có thể có nhiều thuộc tính với các thuộc tính sẽ thể hiện thêm các chức năng cụ thể của thẻ như thuộc tính class, id, color, ... *Ví dụ:* <p id="tagline"></p>.
* Thẻ rỗng
  + Thẻ rỗng là thẻ mà trong đó chỉ tồn tại thẻ mở và không có thẻ đóng, đối với thẻ này thì chức năng của thẻ sẽ thực hiện ngay tại vị trí được đặt thẻ. Khi đó để đảm bảo tính hợp lệ của HTML ta cần đặt thêm một dấu gạch chéo ở cuối thẻ mở theo cấu trúc: **<tên\_thẻ các\_thuộc\_tính />** *Ví dụ:* <img src="test.jpg" />
* Cấu trúc tài liệu HTML

**<html>**

**<head>**

**<title>** tiêu đề trang **</title>**

//Các thẻ thiết lập cấu trúc

**</head>**

**<body>**

…………..

//Các thẻ HTML xây dựng nội dung trang Web

  …………..

**</body>**

**</html>**

* Một tài liệu HTML gồm 2 phần riêng biệt:
  + Phần đầu **<head>….</head>** : Chứa các thông tin về tài liệu và không được hiển thị trên màn hình.
  + Phần thân **<body>….</body>**: Chứa nội dung của trang web được hiển thị trên màn hình.

#### Ưu, nhược điểm của ngôn ngữ HTML

* Ưu điểm
  + Có nguồn tài nguyên khổng lồ, hỗ trợ một cộng đồng người dùng lớn. Bên cạnh đó, cộng đồng HTML ngày càng phát triển trên thế giới.
  + Mã nguồn của HTML là mã nguồn mở, do đó người dùng có thể sử dụng hoàn toàn miễn phí.
  + HTML được sử dụng và được sử dụng trên nhiều trình duyệt được nhiều người dùng ưa chuộng hiện nay như Internet Explorer, Chrome, FireFox, Cốc cốc, …
  + Học và tìm hiểu HTML đơn giản nên người học dễ dàng nắm được kiến thức và vận dụng trong xây dựng trang web căn bạn.
  + HTML được thực hiện dễ dàng bởi HTML được tích hợp nhiều ngôn ngữ khác nhau như PHP, Java, NodeJs, Ruby,…Điều này sẽ giúp tạo thành một website hoàn chỉnh với nhiều tính năng.
* Nhược điểm:
  + Nhược điểm lớn nhất của HTML đó chính là chỉ có thể web tĩnh, web tĩnh có thể hiểu là những trang web chỉ hiện thông tin mà không có sự tương tác cho người dùng. Do đó, khi xây dựng tính năng động hoặc xây dựng hệ thống website có sự tương tác với người dùng, lập trình viên cần phải dùng thêm JavaScript hoặc ngôn ngữ backend của bên thứ ba.
  + HTML thường chỉ có thể thực thi những thứ logic và cấu trúc nhất định, HTML không có khả năng tạo sự khác biệt và mới mẻ.
  + Một số trình duyệt vẫn còn chậm trong viết hỗ trợ các phiên bản mới của HTML, đặc biệt là HTML5.

### **Giới thiệu về ngôn ngữ CSS (Cascading Style Sheet)**

#### Giới thiệu

* CSS là viết tắt của cụm từ "Cascading Style Sheet", nó là một ngôn ngữ quy định cách trình bày của các thẻ HTML trên trang web. Là ngôn ngữ đang được sử dụng rất nhiều trong lập trình web, có thể nói CSS ra đời đã tạo nên một cuộc cách mạng ở các website trên toàn thế giới.
* Có thể thấy CSS có thể quản lý toàn bộ các thành phần (giao diện) của trang web thông qua 1 file CSS, điều khiển chính xác hơn cách trình bày Layout. Quan trọng nhất là tính tái sử dụng để áp dụng cho nhiều thành phần giống nhau. Sử dụng được các kỹ thuật tiên tiến, phức tạp mà HTML không làm được.
* Để học CSS cần có kiến thức về HTML, nó không thật sự cần thiết nếu bạn chỉ dùng CSS để trình bày cho một trang HTML có sẵn, nhưng bạn vẫn cần biết ý nghĩa một số thẻ HTML, nó sẽ có ích khi bạn viết CSS. Tuy nhiên, nếu bạn muốn tự thiết kế, trình bày một trang web của riêng mình thì tùy theo quy mô trang web, bạn cần phải học thêm cả HTML, XHTML, Javascript và một số ngôn ngữ lập trình web khác.
* Để viết mã CSS có thể sử dụng một trình soạn thảo đơn giản như Notepad, Wordpad trong Windows hay Pico trong Linux, Simple Text trong Mac, … tuy nhiên được sử dụng phổ biến nhất hiện nay vẫn là công cụ Macro Dreamweaver. Ngoài ra, để có thể hiển thị tốt nhất cách thể hiện của một trang HTML kết hợp với CSS nên sử dụng trình duyệt phiên bản mới nhất.

#### Lịch sử ngôn ngữ CSS

* CSS được đề xuất lần đầu tiên vào ngày 10/10/1994 bởi Håkon Wium Lie. Kể từ đó, các phiên bản của CSS dần được hình thành qua nhiều giai đoạn. Tính từ lúc xuất hiện đến nay, CSS đã có nhiều phiên bản khác nhau. Các phiên bản mới sẽ giúp vá các lỗi của phiên bản cũ và mang đến nhiều cải tiến hơn.
* Với phiên bản đầu tiên hay còn được gọi là CSS cấp 1 có những đặc điểm cụ thể như: thuộc tính font chữ, màu văn bản, hình nền, các thuộc tính văn bản, căn lề, định vị cho các yếu tố, nhận dạng duy nhất và phân loại chung các nhóm thuộc tính.
* Phiên bản CSS cấp 2 được W3C phát triển vào tháng 5 năm 1998. Với những cải tiến từ phiên bản CSS cấp 1 và mang đến những cải tiến mới như định vị tuyệt đối, tương đối và cố định các yếu tố chỉ mục z. Khái niệm về các loại phương tiện, hỗ trợ cho các biểu định kiểu âm thanh và văn bản hai chiều. Xuất hiện các kiểu font chữ mới để định dạng văn bản.
* Ngoài ra, sau phiên bản cấp 2 còn có một sự nâng cấp khác là CSS 2.1 được đề xuất vào tháng 4 năm 2011. Nhằm mục đích sửa lỗi và loại bỏ những tính năng kém hoặc không tương thích cho người dùng.

#### Cú pháp, vị trí đặt CSS

* Cú pháp CSS:

Tên\_thẻ {Thuộc\_tính:giá\_trị ;}

 Hoặc

 .Tên\_class {Thuộc\_tính:giá\_trị ;}

 Hoặc

#Tên\_ID {Thuộc\_tính:giá\_trị ;}

Trong đó:

* + Tên thẻ: Là tên một thẻ HTML mà người lập trình muốn CSS tương tác;
  + Tên class: Tên lớp bất kỳ do người lập trình định nghĩa được sử dụng nhiều lần trên trang HTML;
  + Tên ID: Tên của một phần tử ID bất kỳ chỉ được sử dụng 1 lần duy nhất trên trang HTML.
  + Thuộc tính: Chính là các thuộc tính quy định cách trình bày. Ví dụ: background-color, font-family, color, padding, margin,…

Mỗi thuộc tính CSS phải được gán một giá trị. Nếu có nhiều hơn một thuộc tính cho một thẻ thì chúng ta phải dùng một dấu ; (chấm phẩy) để phân cách các thuộc tính. Tất cả các thuộc tính trong một selector sẽ được đặt trong một cặp ngoặc nhọn sau thẻ.

* Vị trí đặt CSS:
  + Cách 1: Nội tuyến (kiểu thuộc tính) đây là 1 phương pháp nguyên thủy nhất để nhúng CSS vào 1 trang HTML. Bằng cách nhúng vào từng thẻ HTML muốn áp dụng và dĩ nhiên trong trường hợp này chúng ta sẽ không cần thẻ trong cú pháp.

 Lưu ý: Nếu bạn muốn áp dụng nhiều thuộc tính cho nhiều thẻ HTML khác nhau thì không nên dùng cách này.

* + Cách 2: Bên trong (thẻ style) Thật ra nếu nhìn kỹ chúng ta cũng nhận ra đây chỉ là một phương cách thay thế cách thứ nhất bằng cách rút tất cả các thuộc tính CSS vào trong thẻ style (để tiện cho công tác bảo trì, sửa chữa).

Lưu ý: Thẻ style nên đặt trong thẻ head.

Đối với những trình duyệt cũ, không thể nhận ra thẻ <style>. Theo mặc định, thì khi một trình duyệt không nhận ra một thẻ thì nó sẽ hiện ra phần nội dung chứa trong thẻ.

* + Cách 3: Bên ngoài (liên kết với một file CSS bên ngoài): Tương tự như cách 2, thay vì đặt tất cả mã CSS trong thẻ style chúng ta sẽ đưa chúng vào trong 1 file CSS (có đuôi là .css) bên ngoài là liên kết nó vào trong web bằng thuộc tính href trong thẻ link. Đây là cách làm tối ưu nhất, được khuyến cáo, nó đặc biệt hữu ích cho việc đồng bộ hay bảo trì một website lớn sử dụng cùng một kiểu.

#### Ưu, nhược điểm của CSS

* Ưu điểm:
  + Giải quyết một vấn đề lớn

+ Trước khi có CSS, các thẻ như phông chữ, màu sắc, kiểu nền, cách sắp xếp phần tử, đường viền và kích thước phải được lặp lại trên mọi trang web. Đây là một quá trình rất dài tốn thời gian và công sức. Ví dụ: Nếu bạn đang phát triển một trang web lớn nơi phông chữ và thông tin màu được thêm vào mỗi trang, nó sẽ trở thành một quá trình dài và tốn kém. CSS đã được tạo ra để giải quyết vấn đề này. Đó là một khuyến cáo của W3.

+ Nhờ CSS mà source code của trang Web sẽ được tổ chức gọn gàng hơn, trật tự hơn. Nội dung trang web sẽ được tách bạch hơn trong việc định dạng hiển thị. Từ đó, quá trình cập nhập nội dung sẽ dễ dàng hơn và có thể hạn chế tối thiểu làm rối cho mã HTML.

* + Tiết kiệm được thời gian:

+ Khi sử dụng CSS, bạn có thể chỉ cần phải viết lần đầu, sau đó thì hoàn toàn có thể sử dụng CSS này cho các trang HTML tiếp theo. Ngoài ra, bạn cũng có thể xác định được một kiểu cho từng thành phần HTML. Hơn nữa còn có thể áp dụng cho nhiều trang web khác khi cần thiết.

* + Khả năng tải trang nhanh

+ CSS có đặc trưng đó chính là sử dụng ít mã hơn, chính vì vậy nó giúp cho thời gian tải trang trở nên nhanh hơn. Nhờ vậy mà người dùng khi truy cập vào [website](https://vi.wikipedia.org/wiki/Website) của bạn có thể tiết kiệm được thời gian một cách đáng kể, từ đó tạo cảm giác hài lòng hơn.

+ Với quy tắc CSS của một thẻ cụ thể bạn hoàn toàn có thể áp dụng cho mọi lần xuất hiện tiếp theo của thẻ đó. Bên cạnh đó cũng không cần tiến hành thay đổi thuộc tính của thẻ HTML.

* + Khả năng tương thích tốt

+ CSS được đánh giá cao nhờ có những thuộc tính rộng hơn khi so sánh với HTML. Bên cạnh đó thì khả năng tương thích với nhiều thiết bị cũng là một ưu điểm của CSS.

+ Bạn có thể cùng sử dụng một tài liệu HTML song với nhiều phiên bản website khác nhau được trình bày ở từng [thiết bị di động](https://vi.wikipedia.org/wiki/Thi%E1%BA%BFt_b%E1%BB%8B_di_%C4%91%E1%BB%99ng). Và chúng đều được hỗ trợ tốt, có độ tương thích cao.

* + Cung cấp thêm các thuộc tính

+ Định nghĩa kiểu CSS được lưu trong các tệp CSS bên ngoài vì vậy có thể thay đổi toàn bộ trang web bằng cách thay đổi chỉ một tệp. Sử dụng CSS sẽ giúp bạn không cần thực hiện lặp lại các mô tả cho từng thành phần. Từ đó, bạn có thể tiết kiệm được tối đa thời gian làm việc với nó, làm code ngắn lại giúp kiểm soát dễ dàng hơn các lỗi không đáng có.

+ CSS tạo ra nhiều style khác nhau nên có thể được áp dụng với nhiều trang web, từ đó giảm tránh việc lặp lại các định dạng của các trang web giống nhau

* Nhược điểm:
  + Hoạt động khác biệt với từng trình duyệt

+ Những thay đổi ban đầu của CSS được thực hiện trên một website hết sức dễ dàng. Tuy nhiên đòi hỏi ta cần phải xác nhận được tính tương thích khi CSS hiển thị hiệu ứng thay đổi tương tự cho từng [trình duyệt](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web).

* + Tương đối khó khăn với người mới

+ Ngày nay thì sự phát triển của ngôn ngữ lập trình vô cùng đa dạng và phức tạp, rất dễ gây ra những khó khăn cho người mới. Hơn nữa với nhiều cấp độ khác nhau, thì việc tìm hiểu và nắm bắt để sử dụng được CSS ngày càng trở nên khó khăn hơn.

* + Có khả năng gặp rủi ro

+ Việc truy cập vào CSS khá dễ dàng vì nó là hệ thống dựa trên văn bản mở. Điều này có thể làm cho định dạng của toàn bộ web bị chịu tổn thương, hoặc những tai nạn có thể xảy ra với tệp. Khi đó sẽ có yêu cầu truy cập để đọc hoặc ghi dữ liệu để có thể ghi đè lên các thay đổi.

### **Giới thiệu về ngôn ngữ Javascript**

#### Giới thiệu

* JavaScript là một ngôn ngữ lập trình website, được tích hợp và nhúng trong HTML giúp website sống động hơn. JavaScript cho phép kiểm soát các hành vi của trang web tốt hơn so với khi chỉ sử dụng mỗi HTML. Vậy ứng dụng thực tiễn của JavaScript là gì? Các slideshow, pop-up quảng cáo và tính năng autocomplete của Google là những ví dụ dễ thấy nhất cho bạn, chúng đều được viết bằng JavaScript

JavaScript là ngôn ngữ lập trình được hỗ trợ hầu như trên tất cả các trình duyệt như Firefox, Chrome,… thậm chí các trình duyệt trên thiết bị di động.

* JavaScript có thể được ứng dụng trong rất nhiều lĩnh vực:
  + Lập trình website.
  + Xây dựng ứng dụng cho website máy chủ.
  + Ứng dụng di động, app, trò chơi.

Khi tải một trang web, trình duyệt phân tích cú pháp HTML và tạo ra một loại dữ liệu gọi là DOM từ nội dung. DOM thể hiện chế độ xem trực tiếp của trang web với mã JavaScript. Đoạn mã này thực hiện cập nhật cho DOM và được trình bày ngay lập tức cho người dùng.

Trình duyệt cũng ghi nhận các sự kiện giao diện người dùng như: di chuyển chuột, nhấp chuột, v.v. Sau đó, tùy theo phản hồi của người dùng, đoạn mã sẽ thực hiện công việc được lập trình tương ứng. Sử dụng tất cả các tiện ích này, bạn có thể xây dựng các ứng dụng nhỏ để phục vụ cho một vài mục đích được cho phép.

#### Lợi ích của JavaScript

* Là một ngôn ngữ lập trình phổ biến sử dụng trên 92% nền tảng website hiện nay, JavaScript đã thể hiện vai trò quan trọng với lĩnh vực này. Tầm quan trọng của nó thể hiện qua các thao tác, công dụng như:
  + Triển khai tập lệnh phía máy khách: Nhờ Javascript, các lập trình viên có thể dễ dàng viết tập lệnh phía máy khách, tích hợp các tập lệnh một cách liền mạch vào HTML, cho phép website tương tác, trả lời người dùng ngay lập tức và tạo ra giao diện hiển thị phong phú hơn.
  + Viết mã phía máy chủ: Các lập trình viên có thể viết mã phía máy chủ bằng JavaScript.
  + Đơn giản hóa phát triển ứng dụng web phức tạp: Javascript cho phép các nhà phát triển đơn giản hóa thành phần của ứng dụng, qua đó đơn giản hóa việc phát triển các ứng dụng web phức tạp.
  + Thiết kế web responsive: JavaScript cho phép thiết kế web responsive – tối ưu trên cả máy tính và thiết bị di động chỉ với một bộ mã.
  + Google AMP: Để tham gia vào dự án Tăng tốc trang di động (AMP) của Google, các lập trình viên phải sử dụng ngôn ngữ lập trình JavaScript.
  + Nhiều bộ chuyển đổi: Mặc dù thiếu một số tính năng phức tạp được cung cấp bởi các ngôn ngữ lập trình hiện đại như Java và C#, JavaScript vẫn có thể dễ dàng mở rộng bằng cách sử dụng các bộ chuyển đổi như CoffeeScript, TypeScript, DukeScript và Vaadin.

#### Các thư viện, các IDE và các hoạt động của JavaScript

* Các thư viện Javascript phổ biến:

Hiện nay có rất nhiều thư viện và khung được viết từ Javascript như:

* + jQuery: Chuyên về hiệu ứng.
  + js: Chuyên xây dựng ứng dụng thời gian thực.
  + AngularJS: Chuyên xây dựng ứng dụng trang cá nhân.
  + ReactJS: Chuyên viết ứng dụng mobile.
  + Một số thư viện khác như ExtJS, Sencha Touch,….
* Các IDE JavaScript tốt nhất để phát triển ứng dụng JavaScript:
  + Visual Studio.
  + Visual Studio Code (VS Code).
  + WebStorm.
  + Atom.
  + NetBeans
* Cách hoạt động của JavaScript:
  + JavaScript thường được nhúng trực tiếp vào một trang web hoặc được tham chiếu qua file .js riêng.
  + Nó là ngôn ngữ lập trình từ phía khách hàng, tức là script được tải về máy của người dùng đang truy cập và được xử lý tại đó, thay vì xử lý trên máy chủ rồi mới đưa kết quả tới khách hàng.

#### Ưu, nhược điểm của ngôn ngữ JavaScript

* Ưu điểm:
  + JavaScript là ngôn ngữ lập trình dễ học.
  + Lỗi của JavaScript dễ phát hiện hơn và vì vậy dễ sửa hơn.
  + JavaScript hoạt động trên nhiều trình duyệt, nền tảng.
  + JavaScript giúp website tương tác tốt hơn với khách truy cập.
  + JavaScript nhanh hơn và nhẹ hơn các ngôn ngữ lập trình khác.
* Nhược điểm:
  + Dễ bị khai thác.
  + Có thể được dùng để thực thi mã độc trên máy tính của người dùng.
  + Có thể bị triển khai khác nhau tùy từng thiết bị dẫn đến việc không đồng nhất.

## **Giới thiệu về phần mềm Visual Studio Code và Cơ sở dữ liệu**

### **Giới thiệu về phần mềm Visual Studio Code**

#### Giới thiệu:

* Visual studio code là một trình soạn thảo mã nguồn được phát triển bởi Microsoft dành cho Windows, Linux và macOS.
* Nó hỗ trợ chức năng debug, đi kèm với Git, có chức năng nổi bật cú pháp, tự hoàn thành mã thông minh và cải tiến mã nguồn. Nó cũng cho phép tùy chỉnh, do đó người dùng có thể thay đổi theme, phím tắt và các tùy chọn khác. Nó miễn phí và là phần mềm mã nguồn mở theo giấy phép MIT.
* Visual studio code được dựa trên Electron, một nền tảng được sử dụng để triển khai các ứng dụng Node.js máy tính cá nhân chạy trên động cơ bố trí Blink.
* Visual studio code là một trình biên tập mã. Nó hỗ trợ nhiều ngôn ngữ và chức năng tùy vào ngôn ngữ sử dụng như trong bảng sau:

|  |  |
| --- | --- |
| Chức năng | Ngôn ngữ |
| Syntax highlighting | Batch, C++, Java, Objective-C, Perl, Python, R, Ruby, SQL, Visual Basic, XML,... |
| Snippets | Groovy, Markdown, Nim, PHP, Swift |
| Tự động hoàn thành mã thông minh | CSS, HTML, JavaScript, JSON, TypeScript, Less, Sass |
| Cải tiến mã nguồn | C#, TypeScript |
| Debugging | * JavaScript và TypeScript cho Node.js * C# và F# cho Mono trên Linux và macOS * C và C++ trên Windows, Linux và macOS * Python |

* Nhiều chức năng của Visual studio code không hiển thị ra trong các menu tùy chọn hay giao diện người dùng.

#### Ưu, nhược điểm của Visual Studio Code

* Ưu điểm:
  + Hỗ trợ đa nền tảng: Windows, Linux, macOS,...
  + Đa dạng ngôn ngữ lập trình.
  + Ngôn ngữ, giao diện tối giản, thân thiện giúp các lập trình viên dễ dàng định hình.
  + Các tiện ích mở rộng rất đa dạng và phong phú.
  + Tích hợp các tính năng quan trọng như tính năng bảo mật (Git), khả năng tăng tốc xử lý vòng lặp (Debug).
  + Đơn giản hóa việc tìm quản lý hết tất cả các Code có trên hệ thống.
* Nhược điểm:
  + Mất nhiều thời gian để mở và không có extension nào.
  + Việc chạy các ngôn ngữ khác cũng có thể bị chậm. Đây chủ yếu là sự cố với Raspberry Pi. Tuy nhiên điều này không phải là vấn đề lớn với Python, vì nó là một ngôn ngữ thông dịch.

### **Giới thiệu về Cơ sở dữ liệu**

#### Giới thiệu:

Cơ sở dữ liệu (Database) là một tập hợp các dữ liệu có tổ chức liên quan đến nhau, thường được lưu trữ và truy cập điện tử từ hệ thống máy tính.

* Có hai loại cơ sở dữ liệu là cơ sở dữ liệu quan hệ và cơ sở dữ liệu phi quan hệ:
  + Cơ sở dữ liệu quan hệ:

+ Đối với cơ sở dữ liệu quan hệ, dữ liệu được cấu thành các bảng, mỗi bảng gồm có các hàng và các cột, mỗi dữ liệu được ghi thêm vào phải đảm bảo cấu trúc đúng như đã quy định của bảng đó. Dữ liệu sẽ được ghi thông qua ngôn ngữ truy vấn SQL (Structured Query Language). Các cơ sở dữ liệu quan hệ có thể kể đến như MS SQL Server, MySQL, Oracle,...

* + Cơ sở dữ liệu phi quan hệ:

+ Cơ sở dữ liệu phi quan hệ (thường được gọi là NoSQL database)là cơ sở dữ liệu không cần quy định trước cấu trúc, mỗi dữ liệu có thể có các trường thông tin khác nhau (lưu trữ cấu trúc khá giống với json). Cơ sở dữ liệu phi quan hệ thì không dùng ngôn ngữ truy vấn. Một số cơ sở dữ liệu phi quan hệ có thể kể đến như MongoDB, CouchDB…

+ Để quản lý các cơ sở dữ liệu này, ta sử dụng các hệ quản trị cơ sở dữ liệu hay còn gọi là DBMS (Database Management System). Các hệ quản trị này cung cấp cho chúng ta các tiện ích để tương tác với cơ sở dữ liệu và hầu hết thời gian là chúng ta làm việc với DBMS để quản lý cơ sở dữ liệu.

#### Ưu, nhược điểm của Cơ sở dữ liệu

* Ưu điểm:

Việc sử dụng cơ sở dữ liệu sẽ giải quyết các vấn đề được đặt ra: Trùng lặp dữ liệu, dữ liệu không đồng  bộ, không an toàn, lọc và tìm kiếm thông tin không dễ dàng

* + Việc sử dụng cơ sở dữ liệu sẽ giúp dễ dàng truy cập database và lấy thông tin khi cần, điều này giúp tránh được việc trùng lặp dữ liệu, lãng phí tài nguyên.
  + Khi đã dùng chung cơ sở dữ liệu, thông tin được cập nhật trên cơ sở dữ liệu thì người dùng lấy dữ liệu, dữ liệu sẽ được đảm bảo đồng bộ hoàn toàn.
  + Thông tin được quản lý ở một nơi duy nhất, đảm bảo việc bảo mật dữ liệu, tránh rò rỉ thông tin. Ngoài ra có thể phân quyền cho người dùng, ai có thể truy cập dữ liệu.
  + Việc tìm kiếm thông tin cũng vô cùng dễ dàng khi dữ liệu đã được quản lsy bởi DBMS, chúng có thể tìm kiếm, gom nhóm các thông tin lại với nhau bằng cách dùng câu truy vấn.
* Nhược điểm:

Việc sử dụng cơ sở dữ liệu cũng có một số nhược điểm sau:

* + Việc sử dụng chung cơ sở dữ liệu khi xảy ra sự cố, toàn bộ dữ liệu sẽ bị mất hoàn toàn. Tuy nhiên nhược điểm này có thể dễ dàng khắc phục bằng cách sao lưu dữ liệu thường xuyên.
  + Việc dùng chung cơ sở dữ liệu khi có quá nhiều truy vấn sẽ gây nghẽn, sập hệ thống. Tuy nhiên, trong thực tế, lượng truy vấn đến cơ sở dữ liệu đã được ước tính và đầu tư cấu hình máy chủ cho phù hợp.

# **CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

## **Khảo sát hệ thống**

### **Khảo sát sơ bộ**

* *Admin:*
  + Quản lý hoạt động của trang web, quản lý một số thông tin cơ bản của trang web.
  + Quản lý thông tin chi tiết của trang web, có thể them, sửa, xóa các thông tin trên trang web.
  + Quản lý thông tin, tài khoản khách hàng tương ứng có thể tìm kiếm, thêm, sửa, xóa tài khoản khách hàng.
* *Người dùng:*
  + Có thể truy cập vào trang web, xem thông tin trang web, tìm kiếm một số địa điểm muốn tìm hiểu, tạo tài khoản, đăng nhập.

### **Tài liệu đặc tả người dùng**

#### Khảo sát chi tiết

* Các hoạt động của hệ thống:
  + Quản lý thông tin sản phẩm
  + Quản lý thông tin người dùng
  + Đăng nhập
  + Tìm kiếm thông tin địa điểm
  + ...

#### Yêu cầu chức năng

* Bên phía Khách hàng:
  + Đăng ký tài khoản.
  + Đăng nhập.
  + Tìm kiếm.
  + Danh mục địa điểm.
  + ...
* Bên phía Admin:
  + Đăng nhập:
  + Quản lý tài khoản khách hàng
  + Quản lý thông tin trang web

#### Yêu cầu phi chức năng

* + Sử dụng ngôn ngữ Tiếng Việt, không chứa những từ vùng miền khó hiểu, phải phù hợp với mọi đối tượng khách hang
  + Đối với ngôn ngữ lập trình, sử dụng html và css để tạo giao điện ngoài ra có thể sử dụng thêm javascript để xử lý những thao tác hệ thống…
  + Trang web hỗ trợ cứu hộ hay trang quản trị hoạt động 24/24h.
  + Giao diện(UI) dễ nhìn có thiện cảm với người dung.
  + Các vấn đề thao tác của người dùng cần đơn giản dễ thao tác tránh cảm giác gây ức chế khi người dùng thao tác.
  + Ổn định, xử lý nhanh, tốc độ tải trang hay thực hiện công việc không được quá lâu, hiện các thông báo khi kết thúc một thao tác công việc nào đó.
  + Trang web có thể tương tích với các trình duyệt khác nhau các thiết bị truy cập khác nhau.
  + An toàn, bảo mật: Đảm bảo an toàn thông tin người dùng. Các thao tác nghiệp vụ của người quản trị chỉ thực hiện được khi có quyền.

## **Phân tích hệ thống**

### *Biểu đồ use case*

### *Mô tả chi tiết use case*

### *Biểu đồ trình tự*

### *Mô hình hóa dữ liệu của hệ thống*

### *Thiết kế cơ sở dữ liệu*

### *Thiết kế giao diện*

# **CHƯƠNG 3: CÀI ĐẶT CHƯƠNG TRÌNH**

# **CHƯƠNG 4: KẾT QUẢ**