**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: THIẾT KẾ, LẬP TRÌNH FRONT-END**

**ĐỀ TÀI: THIẾT KẾ WEBSITE KINH DOANH ĐIỆN THOẠI**

**Giáo viên hướng dẫn: Kiều Tuấn dũng**

**Sinh viên thực hiện:**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã sinh viên** | **Họ và tên** | **Lớp** |
| 1 | 1871020044 | Đào quang việt Anh | 18-05 |

**Hà Nội, năm 2024**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: THIẾT KẾ, LẬP TRÌNH FRONT-END**

**ĐỀ TÀI: THIẾT KẾ WEBSITE KINH DOANH ĐIỆN THOẠI**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Mã Sinh Viên** | **Họ và Tên** | **Ngày Sinh** | **Điểm** | |
| **Bằng Số** | **Bằng Chữ** |
| **1** | **1871020044** | **Đào Quang Việt Anh** | **02/09/2006** |  |  |

### 

### **CÁN BỘ CHẤM THI**

**Hà Nội, năm 2024**

**LỜI NÓI ĐẦU**

Trong thời đại công nghệ thông tin phát triển như vũ bão hiện nay, thương mại điện tử đang dần trở thành xu thế tất yếu trong hoạt động kinh doanh và tiếp thị sản phẩm, dịch vụ của doanh nghiệp. Việc xây dựng và phát triển một website chuyên nghiệp không chỉ giúp các doanh nghiệp quảng bá thương hiệu một cách rộng rãi mà còn hỗ trợ gia tăng doanh số bán hàng, tiếp cận khách hàng tiềm năng một cách nhanh chóng, tiện lợi và hiệu quả. Trong bối cảnh đó, ngành kinh doanh điện thoại - một trong những lĩnh vực cạnh tranh khốc liệt nhất hiện nay - cũng không nằm ngoài xu thế số hóa và chuyển đổi mô hình kinh doanh truyền thống sang trực tuyến.

Website kinh doanh điện thoại đóng vai trò là cầu nối giữa doanh nghiệp và khách hàng, là nơi cung cấp thông tin đầy đủ, minh bạch về các sản phẩm, dịch vụ cũng như hỗ trợ người dùng thực hiện các giao dịch một cách dễ dàng. Một website bán điện thoại hiệu quả không chỉ cần có nội dung phong phú, hình ảnh bắt mắt mà còn đòi hỏi phải có giao diện thân thiện với người dùng, tốc độ tải trang nhanh, dễ dàng tương tác và thích ứng tốt trên nhiều thiết bị khác nhau. Điều này đặt ra yêu cầu cao đối với quá trình thiết kế và lập trình Front-end, phần trực tiếp tương tác với người dùng và quyết định trải nghiệm tổng thể trên website.

Front-end là phần không thể thiếu trong bất kỳ một hệ thống website thương mại điện tử nào, đóng vai trò chính trong việc xây dựng giao diện và định hình phong cách nhận diện thương hiệu của doanh nghiệp trên môi trường số. Thiết kế và lập trình Front-end không chỉ dừng lại ở việc tạo ra các trang web có giao diện đẹp mắt, mà còn bao gồm việc đảm bảo tính năng động, trực quan, dễ sử dụng và thân thiện với người dùng cuối. Đặc biệt, trong lĩnh vực kinh doanh điện thoại - nơi mà khách hàng thường xuyên truy cập qua các thiết bị di động - việc tối ưu hóa giao diện và trải nghiệm người dùng trên nhiều kích thước màn hình càng trở nên quan trọng.

Với mong muốn tìm hiểu sâu hơn về vai trò, quy trình và các công nghệ hiện đại trong thiết kế lập trình Front-end, em đã chọn đề tài **"Thiết kế website kinh doanh điện thoại - phần thiết kế lập trình Front-end"** làm đề tài nghiên cứu và thực hành. Trong khuôn khổ đề tài này, em tập trung vào việc xây dựng một website kinh doanh điện thoại với giao diện hiện đại, phù hợp với xu hướng thiết kế web mới nhất hiện nay, đồng thời ứng dụng các công nghệ Front-end phổ biến như HTML5, CSS3, JavaScript và các thư viện hỗ trợ như Bootstrap, jQuery cũng như kết hợp với các công nghệ mới như Flexbox, Grid layout và kỹ thuật Responsive Web Design để tối ưu hóa trải nghiệm người dùng.

Quá trình thiết kế Front-end cho website không chỉ là công việc lập trình thuần túy mà còn đòi hỏi sự sáng tạo, khả năng nắm bắt tâm lý người tiêu dùng và xu hướng thiết kế giao diện hiện đại. Đặc biệt, việc đảm bảo tính nhất quán về màu sắc, bố cục, font chữ cũng như khả năng tương tác mượt mà giữa các thành phần trên trang web sẽ góp phần nâng cao sự chuyên nghiệp và uy tín cho thương hiệu doanh nghiệp trong mắt khách hàng. Bên cạnh đó, xu hướng thiết kế hiện đại còn hướng đến việc tạo ra các hiệu ứng chuyển động (animation) tinh tế, các thao tác chuyển đổi mượt mà (transition) nhằm thu hút sự chú ý và tạo điểm nhấn cho người dùng khi truy cập website.

Đề tài cũng đồng thời đề cập đến việc nghiên cứu và áp dụng các kỹ thuật tối ưu hiệu suất Front-end như giảm thiểu dung lượng tài nguyên (hình ảnh, CSS, JavaScript), tối ưu tốc độ tải trang và cải thiện điểm số đánh giá hiệu năng từ các công cụ như Google PageSpeed Insights, Lighthouse,... nhằm đảm bảo website vận hành trơn tru, mượt mà và đạt chuẩn SEO tốt nhất. Mặt khác, em cũng chú trọng đến việc thiết kế các thành phần giao diện như menu điều hướng, banner, trang sản phẩm, trang giỏ hàng, trang thanh toán,... sao cho đảm bảo tính thẩm mỹ, dễ sử dụng và đồng bộ với phong cách thiết kế tổng thể của website.

Thông qua quá trình thực hiện đề tài, em đã có cơ hội củng cố và mở rộng kiến thức, kỹ năng về thiết kế giao diện người dùng, lập trình Front-end, đồng thời nâng cao khả năng làm việc độc lập, tư duy logic và giải quyết vấn đề thực tiễn. Em nhận thức rằng thiết kế Front-end là một lĩnh vực có vai trò quan trọng trong việc xây dựng các hệ thống web hoàn chỉnh, góp phần không nhỏ vào thành công của các mô hình kinh doanh trực tuyến nói chung và kinh doanh điện thoại nói riêng.

em xin chân thành cảm ơn sự hướng dẫn tận tình của quý thầy cô trong suốt quá trình học tập và nghiên cứu. Đồng thời, em cũng xin gửi lời cảm ơn đến các anh chị, bạn bè đã hỗ trợ và đóng góp ý kiến giúp em hoàn thành đề tài này. Mặc dù đã cố gắng hết sức nhưng do giới hạn về mặt thời gian và kiến thức thực tiễn, đề tài chắc chắn vẫn còn tồn tại những thiếu sót nhất định. Em kính mong nhận được sự góp ý, nhận xét từ quý thầy cô và các bạn để em có thể hoàn thiện tốt hơn trong tương lai.

**MỤC LỤC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mục** | **Nội dung** | **Trang** | | |
| **CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN THIẾT KẾ WEBSITE ĐIỆN THOẠI** | | | | |
| **1.1** | **Giới thiệu về đề tài** |  | | |
| **1.2** | **Mục tiêu đề tài** |  | | |
| **1.3** | **Phạm vi nghiên cứu** |  | | |
| **1.4** | **Đối tượng nghien cứu và học tập** |  | | |
| **1.5** | **Công nghệ áp dụng** |  | | |
| **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** | | | | |
| **2.1** | **Tổng quan về JavaScript** |  | | |
| *2.1.1* | *JavaScript hoạt động như thế nào?* |  | | |
| *2.1.1* | *Khung JavaScript* |  | | |
| *2.1.3* | *Thư viện* |  | | |
| *2.1.4* | *JavaScript mang lại những lợi ích gì* |  | | |
| *2.1.5* | *So sánh giữa HTML, CSS và JavaScript* |  | | |
| **2.2** | **Giới thiệu về jQuery** |  | | |
| *2.2.1* | *Khái niệm tổng quan và sự ra đời của jQuery* |  | | |
| *2.2.2* | *Các tính năng và lợi ích nổi bật của jQuery* |  | | |
| *2.2.3* | *Ưu nhược điểm, ứng dụng thực tế và xu hướng của jQuery* |  | | |
| **2.3** | **Bootstrap** |  | | |
| *2.3.1* | *Lịch sử* |  | | |
| *2.3.2* | *Đặc trưng* |  | | |
| *2.3.3* | *Nội dung* |  | | |
| **CHƯƠNG 3. THIẾT KẾ WEBSITE** | | | | |
| **3.1** | **Ý tưởng thiết kế** |  | | |
| **3.2** | **Sơ đồ website** |  | | |
| **3.3** | **Giao diện chính các trang chính** |  | | |
| *3.3.1* | *Trang chủ* |  | | |
| *3.3.2* | *Trang chi tiết sản phẩm* |  | | |
| *3.3.3* | *Trang đặt hàng* |  | | |
| *3.3.4* | *Trang đăng nhập* |  | | |
| **3.4** | **Code của một số trang chính** |  | | |
| **Kết luận** | | |  |
| **Tài liệu** | | |  |

**CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ NHÀ HÀNG**

**1.1 Giới thiệu đề tài**

Trong bối cảnh nền kinh tế số ngày càng phát triển và thói quen mua sắm của người tiêu dùng dần dịch chuyển sang môi trường trực tuyến, việc xây dựng và vận hành các website thương mại điện tử đóng vai trò vô cùng quan trọng trong việc tạo dựng thương hiệu và thúc đẩy doanh số cho các doanh nghiệp, đặc biệt là trong lĩnh vực kinh doanh các sản phẩm công nghệ cao như điện thoại di động. Với sự bùng nổ của các thiết bị di động thông minh, các nhà bán lẻ và doanh nghiệp trong ngành điện thoại không chỉ cần có mặt trên thị trường truyền thống mà còn phải đẩy mạnh hoạt động trên không gian mạng, nơi mà khách hàng có thể dễ dàng tìm kiếm thông tin sản phẩm, tham khảo giá cả, so sánh các mẫu mã và thực hiện các giao dịch mua bán nhanh chóng, tiện lợi ngay trên website.

Đề tài **“Thiết kế website kinh doanh điện thoại”** được lựa chọn thực hiện nhằm xây dựng một nền tảng giao diện người dùng (Front-end) hiện đại, mang tính thực tiễn cao, giúp mô phỏng một hệ thống bán hàng trực tuyến chuyên nghiệp, từ đó tạo ra một môi trường học tập ứng dụng thực tế cho sinh viên khi tiếp cận các công nghệ lập trình giao diện web hiện đại. Website sẽ cung cấp đầy đủ các chức năng cơ bản và nâng cao mà một website thương mại điện tử cần có như: hiển thị danh mục sản phẩm, trang chi tiết sản phẩm, giỏ hàng, thanh toán, liên hệ, đăng ký/đăng nhập tài khoản khách hàng, tích hợp công cụ tìm kiếm và lọc sản phẩm nâng cao, các slider quảng cáo sinh động, các hiệu ứng tương tác mượt mà giúp nâng cao trải nghiệm người dùng.

Đặc biệt, đề tài chú trọng vào việc thiết kế giao diện website theo xu hướng responsive, nghĩa là giao diện sẽ tương thích tốt trên nhiều loại thiết bị khác nhau, từ màn hình lớn của máy tính để bàn cho đến màn hình nhỏ của điện thoại thông minh hay máy tính bảng, đảm bảo rằng khách hàng có thể thao tác và trải nghiệm trọn vẹn website ở bất kỳ đâu, bất kỳ thời điểm nào. Ngoài ra, website cũng hướng đến việc tối ưu hóa khả năng tương tác giữa người dùng và giao diện thông qua việc áp dụng các công nghệ hiện đại như CSS3 animation, thư viện jQuery, Bootstrap cũng như các kỹ thuật lập trình JavaScript thuần để xử lý các sự kiện như hover, click, validate form, hiển thị popup, thông báo và nhiều yếu tố động khác.

Việc thực hiện đề tài này không chỉ giúp người học củng cố và nâng cao kiến thức về lập trình Front-end mà còn tạo điều kiện cho sinh viên rèn luyện tư duy thiết kế giao diện người dùng (UI), tối ưu trải nghiệm người dùng (UX) cũng như tìm hiểu quy trình xây dựng một website thương mại điện tử chuyên nghiệp theo đúng quy chuẩn thực tế hiện nay. Đây sẽ là hành trang quan trọng, tạo nền tảng vững chắc cho sinh viên khi bước vào lĩnh vực thiết kế và phát triển web chuyên nghiệp, sẵn sàng đáp ứng các yêu cầu khắt khe của thị trường lao động trong thời đại công nghệ số hiện đại và không ngừng đổi mới.

**2.2. Mục tiêu của đề tài**

Xây dựng giao diện website hiện đại và chuyên nghiệp: Tạo ra một giao diện website bán điện thoại có thiết kế hiện đại, mang tính thẩm mỹ cao, phù hợp với xu hướng thiết kế web hiện nay. Giao diện cần được thiết kế nhất quán, bố cục rõ ràng, màu sắc hài hòa, phù hợp với lĩnh vực kinh doanh điện thoại và thể hiện rõ nét đặc trưng thương hiệu.

Tối ưu trải nghiệm người dùng (UX/UI): Thiết kế giao diện thân thiện, dễ sử dụng, đảm bảo các thao tác mua sắm, tìm kiếm sản phẩm, xem chi tiết sản phẩm, thêm vào giỏ hàng, thanh toán… được thực hiện nhanh chóng, thuận tiện. Đảm bảo người dùng cảm thấy dễ chịu, thoải mái khi thao tác trên website, từ đó nâng cao mức độ hài lòng và tăng tỷ lệ chuyển đổi.

Đáp ứng khả năng tương thích đa nền tảng (Responsive Design): Thiết kế website có khả năng hiển thị tốt trên mọi thiết bị từ máy tính để bàn (PC), laptop, máy tính bảng (tablet) đến điện thoại thông minh (smartphone), đảm bảo giao diện linh hoạt, tự động điều chỉnh phù hợp với độ phân giải của từng thiết bị.

Xây dựng và lập trình các trang chức năng cơ bản:

* Thiết kế các trang chính bao gồm: Trang chủ, trang danh mục sản phẩm, trang chi tiết sản phẩm, trang giỏ hàng, trang thanh toán, trang liên hệ, trang giới thiệu.
* Phát triển giao diện cho các tính năng hỗ trợ như tìm kiếm sản phẩm, lọc sản phẩm theo danh mục, thêm sản phẩm vào giỏ hàng, quản lý giỏ hàng, đặt hàng và hiển thị các thông báo cho người dùng.

Ứng dụng và thực hành các công nghệ Front-end: Sử dụng các công nghệ và ngôn ngữ phổ biến trong thiết kế web như HTML5, CSS3, JavaScript kết hợp với các thư viện hỗ trợ như Bootstrap 5, jQuery, thư viện Font Awesome,… nhằm rèn luyện và nâng cao kỹ năng lập trình Front-end, đồng thời giúp giao diện website đạt tiêu chuẩn chuyên nghiệp.

Đảm bảo website thân thiện với SEO: Thiết kế giao diện chuẩn SEO, tối ưu hóa tốc độ tải trang, cấu trúc HTML hợp lý để hỗ trợ việc website có thể dễ dàng đạt thứ hạng cao trên các công cụ tìm kiếm như Google, Bing,... từ đó nâng cao khả năng tiếp cận khách hàng tiềm năng.

Tạo nền tảng để phát triển mở rộng trong tương lai: Giao diện được thiết kế với cấu trúc và bố cục khoa học, dễ bảo trì, dễ dàng mở rộng thêm các tính năng nâng cao như quản lý tài khoản khách hàng, hệ thống đánh giá sản phẩm, hệ thống thanh toán trực tuyến hay tích hợp các cổng thanh toán như PayPal, VNPay, Momo,... trong các giai đoạn phát triển sau.

Rèn luyện tư duy thiết kế và kỹ năng lập trình thực tế: Thông qua đề tài, sinh viên được thực hành quy trình thiết kế giao diện web thực tế, từ phân tích yêu cầu, phác thảo mockup, thiết kế UX/UI đến lập trình Front-end hoàn chỉnh. Đây là cơ hội để áp dụng lý thuyết vào thực hành, chuẩn bị hành trang kiến thức và kỹ năng vững vàng phục vụ cho công việc sau khi ra trường.

**1.3. Phạm vi nghiên cứu**

Đề tài "Thiết kế website kinh doanh điện thoại – phần thiết kế lập trình Front-end" được thực hiện trong phạm vi nghiên cứu cụ thể như sau:

Phạm vi nội dung: Đề tài tập trung vào thiết kế và xây dựng giao diện người dùng (Front-end) cho một website thương mại điện tử chuyên về kinh doanh điện thoại di động và các phụ kiện liên quan. Trong đó, trọng tâm là phát triển phần giao diện và bố cục cho các trang chính của website, đảm bảo tính thẩm mỹ và tính tiện dụng cho người dùng. Nội dung nghiên cứu không bao gồm phần lập trình Back-end như xử lý cơ sở dữ liệu, quản lý máy chủ, hay tích hợp các cổng thanh toán thực tế.

Các trang giao diện được nghiên cứu và thiết kế bao gồm:

* Trang chủ: Giới thiệu tổng quan về cửa hàng, các sản phẩm nổi bật, chương trình khuyến mãi, tin tức mới nhất.
* Trang danh mục sản phẩm: Hiển thị các nhóm sản phẩm điện thoại và phụ kiện theo thương hiệu, phân loại theo tiêu chí giá, cấu hình hoặc nhu cầu sử dụng.
* Trang chi tiết sản phẩm: Hiển thị đầy đủ thông tin sản phẩm như hình ảnh, mô tả chi tiết, cấu hình kỹ thuật, giá bán, các tùy chọn màu sắc, đánh giá sản phẩm,...
* Trang giỏ hàng: Hiển thị danh sách sản phẩm đã chọn, cho phép cập nhật số lượng, xóa sản phẩm và hiển thị tổng giá trị đơn hàng.
* Trang thanh toán (checkout): Giao diện nhập thông tin người mua, lựa chọn phương thức thanh toán, địa chỉ giao hàng và xác nhận đơn hàng.
* Trang liên hệ và trang giới thiệu: Giúp khách hàng dễ dàng nắm bắt thông tin liên hệ, chính sách mua hàng và giới thiệu về doanh nghiệp.

Phạm vi công nghệ: Nghiên cứu và sử dụng các công nghệ, ngôn ngữ lập trình phổ biến hiện nay trong thiết kế Front-end bao gồm:

* HTML5: Đảm bảo cấu trúc nội dung website rõ ràng và chuẩn SEO.
* CSS3: Thiết kế giao diện đẹp mắt, sử dụng các hiệu ứng chuyển động (animation), hiệu ứng hover cho các thành phần giao diện.
* JavaScript và jQuery: Xử lý các thao tác tương tác trên giao diện như menu dropdown, slider, popup thông báo, giỏ hàng động,...
* Bootstrap 5: Hỗ trợ thiết kế responsive, tối ưu hiển thị trên nhiều loại thiết bị khác nhau (PC, tablet, smartphone).
* Font Awesome & Google Fonts: Tăng tính thẩm mỹ cho giao diện bằng hệ thống icon đa dạng và font chữ hiện đại.

Phạm vi thiết kế UX/UI:

* Đề tài giới hạn trong việc thiết kế giao diện với các yếu tố cơ bản về trải nghiệm người dùng (UX) như bố cục rõ ràng, dễ sử dụng, giảm số bước thao tác trong quy trình mua hàng.
* Tập trung vào việc xây dựng giao diện trực quan, thân thiện và nhất quán, phù hợp với đối tượng khách hàng mục tiêu là người tiêu dùng phổ thông có nhu cầu mua điện thoại trực tuyến.

Phạm vi người dùng:  
Website phục vụ cho hai nhóm người dùng chính

* Khách hàng: Người tiêu dùng có nhu cầu tìm hiểu, so sánh và mua các sản phẩm điện thoại trực tuyến.
* Nhân viên và quản trị viên website: Tham khảo giao diện để nhập sản phẩm, quảng bá thông tin, hỗ trợ bán hàng và chăm sóc khách hàng thông qua các chức năng cơ bản của giao diện người dùng.

Phạm vi thời gian: Đề tài được thực hiện trong khoảng thời gian giới hạn theo kế hoạch học tập, do đó chỉ tập trung vào hoàn thiện phần giao diện Front-end và chưa đi sâu vào các tính năng Back-end như quản lý dữ liệu, xử lý thanh toán thực tế hay phát triển hệ thống quản trị nội bộ.

**1.4.Đối tượng nghiên cứu và học tập**

Sinh viên ngành công nghệ thông tin, lập trình web: Đây là nhóm đối tượng sử dụng website với mục đích học tập, nghiên cứu quy trình thiết kế và lập trình giao diện Front-end thực tế cho các dự án thương mại điện tử. Thông qua đề tài, sinh viên có thể tham khảo bố cục, cách bố trí thành phần, áp dụng các công nghệ Front-end phổ biến vào dự án thực tiễn.

Giảng viên, giáo viên hướng dẫn: Nhóm đối tượng này sẽ theo dõi sản phẩm như một phần đánh giá quá trình học tập, thực hành và khả năng ứng dụng kiến thức lập trình Front-end vào thực tế của sinh viên.

**1.5. Công nghệ áp dụng**

Để xây dựng giao diện website kinh doanh điện thoại chuyên nghiệp và hiện đại, đề tài lựa chọn và áp dụng một số công nghệ Front-end phổ biến, đáp ứng tốt các yêu cầu về thiết kế, tính thẩm mỹ, khả năng tương tác và tính thân thiện với người dùng. Cụ thể như sau:

HTML5 (HyperText Markup Language 5):

* Đây là ngôn ngữ đánh dấu siêu văn bản được sử dụng để xây dựng cấu trúc nội dung cho website. HTML5 cung cấp các thẻ ngữ nghĩa mới như <header>, <nav>, <section>, <article>, <footer>, giúp tổ chức nội dung website một cách rõ ràng, dễ hiểu và chuẩn SEO.
* HTML5 hỗ trợ đa phương tiện (audio, video) và các biểu mẫu (form) nâng cao, tạo điều kiện thuận lợi cho việc xây dựng các form liên hệ, form đặt hàng, và các thành phần tương tác khác trên website.

CSS3 (Cascading Style Sheets 3):

* CSS3 giúp định dạng và trang trí các thành phần HTML, từ màu sắc, font chữ, bố cục đến các hiệu ứng động (animation) và hiệu ứng chuyển đổi (transition).
* Website sẽ sử dụng CSS3 để thiết kế giao diện hiện đại, tối ưu trải nghiệm người dùng như thiết kế responsive, menu dropdown, hiệu ứng hover khi rê chuột vào các nút bấm, hình ảnh sản phẩm và các thành phần khác.
* Áp dụng Flexbox và Grid Layout giúp giao diện được căn chỉnh chính xác, bố cục linh hoạt và dễ dàng tùy chỉnh.

JavaScript:

* JavaScript là ngôn ngữ lập trình kịch bản phía trình duyệt, giúp website có thể xử lý các hành vi tương tác của người dùng như: thêm sản phẩm vào giỏ hàng, cập nhật số lượng sản phẩm, xử lý các sự kiện nhấp chuột (onclick), đóng/mở popup, xác thực form trước khi gửi dữ liệu.
* Ngoài ra, JavaScript còn hỗ trợ hiển thị các thông báo động (alert, toast), điều hướng mượt mà giữa các trang và nhiều tính năng tăng cường trải nghiệm người dùng khác.

jQuery:

* Là một thư viện JavaScript phổ biến, giúp đơn giản hóa quá trình xử lý DOM, thao tác với các phần tử HTML, thực hiện hiệu ứng động và xử lý các sự kiện tương tác.
* jQuery giúp tăng tốc quá trình phát triển giao diện bằng cách rút ngắn các đoạn mã JavaScript phức tạp, đồng thời dễ dàng tương thích với các trình duyệt web khác nhau.

Bootstrap 5:

* Bootstrap là framework CSS nổi tiếng, cung cấp sẵn hệ thống lưới (grid system), các thành phần giao diện như nút bấm, form, bảng, thanh điều hướng, carousel, modal,... theo chuẩn responsive.
* Sử dụng Bootstrap giúp website hiển thị tốt trên nhiều kích thước màn hình từ PC, laptop, tablet cho đến smartphone, đồng thời tiết kiệm thời gian và công sức khi xây dựng bố cục giao diện.
* Bootstrap 5 cũng hỗ trợ nhiều tiện ích CSS và JavaScript giúp website trở nên sinh động và dễ sử dụng hơn.

Font Awesome & Google Fonts:

* Font Awesome: Thư viện biểu tượng (icon) vector đa dạng, dễ dàng tích hợp vào website nhằm tăng tính thẩm mỹ cho giao diện, giúp người dùng dễ nhận diện chức năng của các nút bấm và thành phần giao diện như giỏ hàng, tìm kiếm, liên hệ, mạng xã hội...
* Google Fonts: Bộ sưu tập font chữ đa dạng, miễn phí, giúp giao diện website trở nên hiện đại và đồng bộ phong cách thiết kế, phù hợp với nhận diện thương hiệu.

Công cụ hỗ trợ thiết kế:

* Figma/Adobe XD: Công cụ thiết kế UX/UI giúp phác thảo bố cục (wireframe) và thiết kế mẫu giao diện (mockup) trước khi tiến hành lập trình thực tế.
* Visual Studio Code: Trình soạn thảo mã nguồn chính sử dụng để lập trình, tích hợp nhiều tiện ích hỗ trợ lập trình Front-end như Emmet, Live Server, Prettier,…

Chuẩn SEO & Web Accessibility:

* Áp dụng các tiêu chuẩn thiết kế giao diện web thân thiện với công cụ tìm kiếm (SEO) như sử dụng thẻ heading đúng chuẩn (H1, H2, H3...), tối ưu tốc độ tải trang, cấu trúc URL thân thiện và thẻ meta hợp lý.
* Đảm bảo website dễ truy cập cho người dùng khuyết tật bằng cách tuân thủ các tiêu chuẩn Web Accessibility như WAI-ARIA, gán đầy đủ thuộc tính alt cho hình ảnh, màu sắc tương phản tốt, dễ đọc.

**1.6. Ý nghĩa**

thiết kế và xây dựng một giao diện website kinh doanh điện thoại di động nhằm mang tính thẩm mỹ cao, hiện đại, chuẩn responsive, giúp tối ưu hóa trải nghiệm người dùng trên nhiều nền tảng thiết bị khác nhau, đồng thời hỗ trợ doanh nghiệp nâng cao hình ảnh thương hiệu, mở rộng thị trường, gia tăng doanh số bán hàng trực tuyến, góp phần số hóa hoạt động kinh doanh truyền thống và đáp ứng xu hướng thương mại điện tử ngày càng phát triển mạnh mẽ trong thời đại công nghệ số hiện nay.

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

Lập trình Front-end là quá trình xây dựng giao diện người dùng (User Interface - UI) của các ứng dụng hoặc website, đảm bảo hiển thị các thông tin một cách trực quan và cho phép người dùng tương tác trực tiếp trên trình duyệt. Lập trình Front-end đóng vai trò quan trọng trong việc thiết kế, trình bày và tổ chức nội dung của website, giúp người dùng có trải nghiệm mượt mà, dễ sử dụng và thân thiện. Cơ sở lý thuyết của lập trình Front-end bao gồm các khái niệm, ngôn ngữ và công nghệ nền tảng

**2.1. Tổng quan về JavaScript**

JavaScript là ngôn ngữ lập trình được nhà phát triển sử dụng để tạo trang web tương tác. Từ làm mới bảng tin trên trang mạng xã hội đến hiển thị hình ảnh động và bản đồ tương tác, các chức năng của JavaScript có thể cải thiện trải nghiệm người dùng của trang web. Là ngôn ngữ kịch bản phía máy khách, JavaScript là một trong những công nghệ cốt lõi của World Wide Web. Ví dụ: khi duyệt internet, bất cứ khi nào bạn thấy quảng cáo quay vòng dạng hình ảnh, menu thả xuống nhấp để hiển thị hoặc màu sắc phần tử thay đổi động trên trang web cũng chính là lúc bạn thấy các hiệu ứng của JavaScript.

Trước đây, các trang web có dạng tĩnh, tương tự như các trang trong một cuốn sách. Một trang tĩnh chủ yếu hiển thị thông tin theo một bố cục cố định và không làm được mọi thứ mà chúng ta mong đợi như ở một trang web hiện đại. JavaScript dần được biết đến như một công nghệ phía trình duyệt để làm cho các ứng dụng web linh hoạt hơn. Sử dụng JavaScript, các trình duyệt có thể phản hồi tương tác của người dùng và thay đổi bố cục của nội dung trên trang web.

Khi ngôn ngữ này phát triển hoàn thiện, các nhà phát triển JavaScript đã thiết lập các thư viện, khung và cách thức lập trình cũng như bắt đầu sử dụng ngôn ngữ này bên ngoài trình duyệt web. Ngày nay, bạn có thể sử dụng JavaScript để thực hiện hoạt động phát triển cả ở phía máy khách và máy chủ

*2.1.1. JavaScript hoạt động như thế nào?*

Tất cả các ngôn ngữ lập trình đều hoạt động bằng cách dịch cú pháp giống tiếng Anh thành mã máy, hệ điều hành sau đó sẽ chạy mã này. JavaScript được phân loại khái quát là một ngôn ngữ viết tập lệnh hoặc một ngôn ngữ diễn giải. Mã JavaScript được diễn giải, tức là được công cụ JavaScript dịch trực tiếp sang mã ngôn ngữ máy cơ bản. Với các ngôn ngữ lập trình khác, trình biên dịch sẽ biên dịch toàn bộ mã thành mã máy trong một bước riêng. Như vậy, tất cả các ngôn ngữ viết tập lệnh đều là ngôn ngữ lập trình, nhưng không phải tất cả các ngôn ngữ lập trình đều là ngôn ngữ viết tập lệnh.

Công cụ JavaScrip :Công cụ JavaScript là một chương trình máy tính chạy mã JavaScript. Các công cụ JavaScript đầu tiên chỉ là bộ diễn giải, nhưng tất cả các công cụ hiện đại đều sử dụng biên dịch tại chỗ hoặc thời gian chạy để cải thiện hiệu suất.

JavaScript phía máy khách:JavaScript phía máy khách đề cập đến cách thức JavaScript hoạt động trong trình duyệt của bạn. Trong trường hợp này, công cụ JavaScript nằm trong mã trình duyệt. Tất cả các trình duyệt web phổ biến đều có các công cụ JavaScript tích hợp riêng.

Các nhà phát triển ứng dụng web viết mã JavaScript với các hàm khác nhau được liên kết với nhiều loại sự kiện, chẳng hạn như nhấp chuột hoặc khi di chuột đến. Các hàm này thực hiện các thay đổi đối với HTML và CSS.

Sau đây là tổng quan về cách thức hoạt động của JavaScript phía máy khách:

* Trình duyệt tải một trang web khi bạn truy cập trang đó.
* Trong khi tải, trình duyệt chuyển đổi trang và tất cả các phần tử của trang, chẳng hạn như các nút, nhãn và hộp thả xuống, thành một cấu trúc dữ liệu được gọi là Mô hình đối tượng tài liệu (DOM).
* Công cụ JavaScript của trình duyệt chuyển đổi mã JavaScript thành mã bytecode. Mã này đóng vai trò là trung gian giữa cú pháp JavaScript và máy.
* Các sự kiện khác nhau, chẳng hạn như nhấp vào nút, sẽ kích hoạt việc thực thi khối mã JavaScript được liên kết. Sau đó, công cụ này sẽ diễn giải bytecode và thực hiện các thay đổi đối với DOM.
* Trình duyệt hiển thị DOM mới.

JavaScript phía máy chủ:JavaScript phía máy chủ đề cập đến việc sử dụng ngôn ngữ viết mã trong logic máy chủ back-end. Trong trường hợp này, công cụ JavaScript nằm trực tiếp trên máy chủ. Hàm JavaScript phía máy chủ có thể truy cập cơ sở dữ liệu, thực hiện các phép toán logic khác nhau và phản hồi nhiều loại sự kiện do hệ điều hành của máy chủ kích hoạt. Ưu điểm chính của viết tập lệnh phía máy chủ là bạn có khả năng tùy chỉnh linh hoạt phản hồi của trang web dựa trên yêu cầu và quyền truy cập của mình cũng như theo yêu cầu thông tin từ trang web.

So sánh giữa phía máy khách và phía máy chủ:Từ linh hoạt đúng với cả JavaScript phía máy khách và phía máy chủ. Hành vi linh hoạt là khả năng cập nhật phần hiển thị của trang web để tạo nội dung mới theo yêu cầu. Sự khác biệt giữa JavaScript phía máy khách và phía máy chủ nằm ở cách chúng tạo ra nội dung mới. Mã phía máy chủ tạo nội dung mới một cách linh hoạt bằng cách sử dụng logic ứng dụng và sửa đổi dữ liệu từ cơ sở dữ liệu. Mặt khác, JavaScript phía máy khách tạo nội dung mới một cách linh hoạt bên trong trình duyệt thông qua sử dụng logic giao diện người dùng và sửa đổi nội dung trang web đã có trên máy khách. Ý nghĩa hơi khác biệt ở hai ngữ cảnh nhưng có sự liên quan đến nhau và cả hai cách tiếp cận này đều cùng nhau nâng cao trải nghiệm người dùng.Ngoài việc triển khai các tính năng linh hoạt, một điểm khác biệt nữa giữa hai cách sử dụng JavaScript là ở tài nguyên mà mã có thể truy cập. Về phía máy khách, trình duyệt kiểm soát môi trường thời gian chạy của JavaScript. Mã chỉ có thể truy cập những tài nguyên mà trình duyệt cho phép truy cập. Ví dụ: mã không thể ghi nội dung vào ổ cứng của bạn trừ khi bạn nhấp vào nút tải xuống. Mặt khác, các hàm phía máy chủ có thể truy cập tất cả các tài nguyên của máy chủ khi cần thiết.

*2.1.2. Khung JavaScript*

Một khung JavaScript (tiếng Anh: JavaScript Framework) là một bộ sưu tập các thư viện JavaScript cùng với một cấu trúc và bộ công cụ được thiết kế sẵn nhằm hỗ trợ các lập trình viên phát triển các ứng dụng web hoặc ứng dụng di động nhanh chóng, hiệu quả và dễ bảo trì hơn. Thay vì phải viết lại tất cả các đoạn mã lặp đi lặp lại từ đầu, lập trình viên có thể sử dụng các thành phần được đóng gói sẵn trong framework để xây dựng các chức năng phổ biến như xử lý DOM (Document Object Model), gọi API, quản lý trạng thái, điều hướng (routing), xử lý sự kiện và nhiều hơn nữa.

Đặc điểm của khung JavaScript:

* Cấu trúc rõ ràng: Framework cung cấp sẵn kiến trúc tổng thể và quy tắc lập trình (convention) cho dự án, giúp lập trình viên dễ dàng tổ chức code và tuân theo những tiêu chuẩn nhất định.
* Tái sử dụng mã nguồn: Các thành phần (components), module, hoặc class có thể được tái sử dụng trong nhiều phần khác nhau của ứng dụng hoặc trong các dự án khác.
* Tự động hóa quy trình phát triển: Framework thường tích hợp sẵn các công cụ build, đóng gói, kiểm thử, giúp rút ngắn thời gian phát triển và giảm bớt các thao tác thủ công.
* Quản lý vòng đời UI: Hầu hết các framework hiện đại hỗ trợ quản lý vòng đời của các thành phần giao diện người dùng (UI Component Lifecycle), bao gồm tạo, cập nhật và hủy bỏ các thành phần.
* Khả năng mở rộng tốt: Framework cho phép dễ dàng mở rộng tính năng thông qua các thư viện bổ sung hoặc plugin bên ngoài.
* Cộng đồng hỗ trợ mạnh mẽ: Các framework phổ biến thường đi kèm với cộng đồng lớn, hệ sinh thái plugin phong phú và tài liệu đầy đủ.

Phân biệt Framework và Library:

* Framework: Áp dụng quy tắc “Inversion of Control” (đảo ngược quyền kiểm soát), nghĩa là framework sẽ điều khiển luồng thực thi của chương trình. Khi sử dụng framework, lập trình viên phải tuân thủ cấu trúc và cách thức hoạt động mà framework định sẵn. Ví dụ như Angular hay Vue.
* Library: Ngược lại, khi sử dụng thư viện, lập trình viên là người chủ động kiểm soát luồng thực thi và chỉ gọi đến các hàm hoặc module của thư viện khi cần thiết, ví dụ như thư viện jQuery hay Lodash.

Các loại Framework phổ biến:

* Framework phát triển giao diện người dùng (UI Frameworks):
* React (mặc dù React là thư viện nhưng thường được sử dụng như một phần của framework khi kết hợp với các công cụ khác như Redux, React Router).
* Angular (một framework hoàn chỉnh, hỗ trợ xây dựng SPA – Single Page Application).
* Vue.js (nhẹ, dễ học, hỗ trợ MVVM).

b. Framework phát triển backend (Server-side Frameworks):

* Next.js (dựa trên React, hỗ trợ SSR - Server Side Rendering).
* Express.js (framework phổ biến nhất của Node.js).
* NestJS (framework hiện đại hỗ trợ TypeScript cho ứng dụng backend).

c. Full-stack Frameworks (hỗ trợ cả frontend và backend):

* Meteor.js (một framework full-stack cho ứng dụng realtime).
* Sails.js (xây dựng trên Express, phù hợp với các ứng dụng RESTful API).
* Nuxt.js (dựa trên Vue, hỗ trợ SSR và static site generation).

Ưu điểm của việc sử dụng khung JavaScript:

* Tiết kiệm thời gian phát triển: Framework cung cấp các module và công cụ dựng sẵn, giúp rút ngắn thời gian viết code.
* Tính bảo trì cao: Code được viết theo cấu trúc chuẩn, dễ dàng cho các lập trình viên khác hiểu và tiếp tục phát triển.
* Hiệu suất tốt: Các framework nổi tiếng thường được tối ưu hóa để mang lại hiệu suất cao cho các ứng dụng.
* Bảo mật tốt hơn: Một số framework đi kèm với các tính năng bảo mật sẵn có như chống XSS (Cross-site scripting) hay CSRF (Cross-site request forgery).
* Hỗ trợ cộng đồng: Với lượng người dùng đông đảo, việc tìm kiếm tài liệu, hướng dẫn, hoặc sự trợ giúp trở nên dễ dàng.

Nhược điểm:

* Đường cong học tập: Một số framework có cấu trúc phức tạp, đòi hỏi thời gian học lâu (ví dụ như Angular).
* Dư thừa tính năng: Đôi khi framework có quá nhiều tính năng mà dự án không sử dụng hết, làm tăng kích thước ứng dụng.
* Phụ thuộc vào framework: Khi dự án phụ thuộc quá nhiều vào framework, việc chuyển đổi sang một công nghệ khác có thể tốn nhiều công sức.

Ví dụ thực tế khi sử dụng khung JavaScript: Angular: Thường được sử dụng để phát triển các hệ thống quản trị (Admin Dashboard), các ứng dụng quản lý nội bộ trong doanh nghiệp.

*2.1.3 Thư viện*

Thư viện JavaScript là tập hợp các đoạn mã viết sẵn mà các nhà phát triển web có thể tái sử dụng để thực hiện các hàm JavaScript chuẩn. Mã thư viện JavaScript được gắn vào phần còn lại của mã dự án khi cần thiết. Nếu bạn coi mã ứng dụng JavaScript như một ngôi nhà thì các thư viện JavaScript sẽ giống như đồ nội thất làm sẵn mà các nhà phát triển có thể sử dụng để cải thiện chức năng của ngôi nhà.

Sau đây là một số cách sử dụng thư viện JavaScript phổ biến:

* Trực quan hóa dữ liệu:

Trực quan hóa dữ liệu có ý nghĩa rất quan trọng để người dùng xem số liệu thống kê, chẳng hạn như trong bảng điều khiển quản trị, bảng thông tin và chỉ số hiệu suất.

Các thư viện như Chart.js, ApexCharts và Algolia Places có các hàm tích hợp mà bạn có thể sử dụng để tạo các ứng dụng web hiển thị dữ liệu ở dạng biểu đồ và bản đồ.

* Điều chỉnh DOM:Bạn có thể sử dụng các thư viện như jQuery và Umbrella JS để dễ dàng phát triển web vì những thư viện này cung cấp mã cho các chức năng tiêu chuẩn của trang web như hoạt ảnh menu, thư viện hình ảnh, nút, cửa sổ bật lên làm tối nền, v.v.
* Biểu mẫu: Tất cả các công việc phát triển web đều sử dụng các biểu mẫu để khách truy cập trang web liên hệ với ai đó, đặt mua sản phẩm và đăng ký tham gia các sự kiện. Một số thư viện JavaScript, chẳng hạn như wForms, LiveValidation, Validanguage và qForms, đơn giản hóa các chức năng biểu mẫu, bao gồm xác thực biểu mẫu, bố cục, điều kiện và chuyển đổi.
* Các hàm toán học và văn bản: Nhiều ứng dụng web phải giải các phương trình toán học và xử lý ngày, giờ cũng như văn bản. Thay vì gửi tất cả các yêu cầu đó đến máy chủ, việc xử lý một số yêu cầu ở phía máy khách sẽ hiệu quả hơn. Các nhà phát triển web thực hiện việc này bằng cách sử dụng các thư viện JavaScript như Date.js, Sylvester và JavaScript URL Library

*2.1.4 JavaScript mang lại những lợi ích gì*

* Tính phổ biến và dễ học

Phổ biến rộng rãi toàn cầu: JavaScript là một trong ba công nghệ cốt lõi của phát triển web, cùng với HTML và CSS, và là ngôn ngữ được sử dụng rộng rãi nhất trong phát triển frontend. Theo nhiều khảo sát (ví dụ như Stack Overflow Developer Survey), JavaScript luôn đứng đầu danh sách ngôn ngữ lập trình phổ biến nhất thế giới trong nhiều năm liền.

Cộng đồng lớn và tài nguyên dồi dào: Việc có một cộng đồng khổng lồ trên toàn cầu giúp lập trình viên dễ dàng tìm kiếm tài liệu, khóa học, ví dụ thực tiễn, giải pháp cho các vấn đề gặp phải hoặc nhận được sự hỗ trợ nhanh chóng trên các diễn đàn như Stack Overflow, Reddit, GitHub.

Dễ tiếp cận: Với cú pháp thân thiện và dễ hiểu, JavaScript là một trong những ngôn ngữ lập trình dễ học nhất cho người mới bắt đầu. Nó không yêu cầu môi trường phức tạp để chạy, chỉ cần một trình duyệt web là có thể bắt đầu thực hành.

* Khả năng chạy đa nền tảng (Cross-platform)

Chạy trên trình duyệt: JavaScript là ngôn ngữ duy nhất có thể chạy trực tiếp trong tất cả các trình duyệt hiện đại (Chrome, Firefox, Safari, Edge, v.v.) mà không cần cài đặt thêm phần mềm hay plugin.

Chạy trên server: Với sự xuất hiện của Node.js, JavaScript không còn chỉ là ngôn ngữ dành riêng cho trình duyệt nữa mà còn có thể chạy trên server, giúp xây dựng các ứng dụng backend mạnh mẽ.

Ứng dụng di động và desktop: Thông qua các framework như React Native (cho mobile) hay Electron (cho desktop), JavaScript còn có thể phát triển các ứng dụng di động đa nền tảng (iOS/Android) và ứng dụng máy tính để bàn (Windows, macOS, Linux).

* Khả năng xây dựng ứng dụng full-stack

Frontend + Backend: Nhờ sự kết hợp giữa JavaScript trên trình duyệt (với các framework như React, Vue, Angular) và trên server (Node.js + Express), lập trình viên có thể xây dựng cả frontend lẫn backend chỉ với một ngôn ngữ duy nhất.

Tiết kiệm chi phí nhân lực: Do chỉ cần tập trung vào một ngôn ngữ, các công ty có thể tuyển dụng các lập trình viên full-stack JavaScript để đảm nhận toàn bộ quá trình phát triển sản phẩm, từ giao diện người dùng đến xử lý server.

* Giao diện người dùng tương tác và linh hoạt

Tạo trải nghiệm người dùng tốt: JavaScript cho phép tạo ra các hiệu ứng động, các thành phần giao diện phản hồi theo thời gian thực (real-time interaction) như kéo thả (drag & drop), popup, modal, animation, slide, menu dropdown, carousel,... làm cho giao diện web trở nên sinh động và chuyên nghiệp hơn.

SPA (Single Page Application): Các framework như Angular, React, Vue cho phép phát triển các ứng dụng đơn trang với hiệu suất cao, nơi người dùng có thể tương tác mà không cần tải lại trang, giúp tối ưu hóa trải nghiệm người dùng.

* Tốc độ thực thi cao và hiệu năng tốt

Tối ưu hóa bởi trình duyệt: Các trình duyệt hiện đại như Google Chrome, Firefox, Edge đều tích hợp các engine JavaScript mạnh mẽ (ví dụ như V8 của Google) giúp mã JavaScript được thực thi với tốc độ rất nhanh, gần như đạt hiệu suất tối ưu cho các tác vụ trên giao diện người dùng.

Không yêu cầu biên dịch: JavaScript là ngôn ngữ interpreted language (ngôn ngữ thông dịch), nghĩa là mã nguồn có thể được chạy ngay lập tức mà không cần phải biên dịch (compile) như các ngôn ngữ như C/C++, Java.

* Hệ sinh thái phong phú và phát triển mạnh mẽ

Hàng triệu thư viện và gói npm: Kho thư viện npm (Node Package Manager) là hệ sinh thái lớn nhất thế giới cho JavaScript với hàng triệu thư viện và module giúp tăng tốc quá trình phát triển và tích hợp nhiều tính năng phức tạp như xử lý thời gian thực (socket.io), phân tích dữ liệu (D3.js), xác thực người dùng (jsonwebtoken), hoặc AI/ML (Tensorflow.js).

Hỗ trợ đa dạng framework: JavaScript cung cấp vô số framework nổi bật phục vụ cho nhiều mục đích khác nhau như:

React.js: Xây dựng UI linh hoạt và hiệu suất cao.

Vue.js: Nhẹ, dễ học, phù hợp cho startup và các ứng dụng quy mô nhỏ đến trung bình.Angular: Framework toàn diện cho ứng dụng quy mô lớn.

Express.js: Xây dựng backend API nhanh chóng.

Next.js, Nuxt.js: Hỗ trợ server-side rendering và SEO.

* Khả năng xử lý thời gian thực

Realtime Communication: Với sự hỗ trợ của WebSocket và thư viện như Socket.io, JavaScript cho phép phát triển các ứng dụng thời gian thực như chat, trò chơi trực tuyến (online game), các hệ thống giám sát hoặc cập nhật dữ liệu live (ví dụ bảng giá chứng khoán, dashboard quản lý).

Event-driven programming: JavaScript được thiết kế theo mô hình lập trình hướng sự kiện, giúp dễ dàng quản lý và xử lý các tương tác từ người dùng như click chuột, di chuyển chuột, nhập dữ liệu từ bàn phím, cuộn trang,…

* Tính mở rộng và khả năng tích hợp mạnh mẽ

Dễ dàng tích hợp với các API: JavaScript rất linh hoạt khi tích hợp với các RESTful API, GraphQL hoặc Web API từ các bên thứ ba như Google Maps, Facebook API, Stripe, PayPal, giúp ứng dụng mở rộng thêm nhiều tính năng như thanh toán, định vị, mạng xã hội, AI/ML,...

Microservices và Cloud-ready: Khi dùng với Node.js, JavaScript hỗ trợ tốt cho việc xây dựng các kiến trúc microservices, dễ dàng triển khai lên các nền tảng cloud như AWS, Azure, Google Cloud.

* Khả năng tương thích cao

Tương thích trình duyệt: JavaScript hoạt động tốt trên mọi trình duyệt hiện đại, giúp giảm thiểu rủi ro về lỗi không tương thích hoặc giới hạn môi trường chạy.

Responsive và adaptive: Kết hợp với CSS và các thư viện như Bootstrap, TailwindCSS, JavaScript giúp xây dựng các giao diện web responsive, tương thích trên mọi kích thước màn hình từ điện thoại đến máy tính bàn.

* Tính bảo mật và hỗ trợ kiểm thử

Hỗ trợ kiểm thử tự động: JavaScript có hệ sinh thái testing phong phú với các công cụ như Jest, Mocha, Jasmine, Cypress giúp lập trình viên dễ dàng viết unit test, integration test và end-to-end test, tăng tính ổn định và an toàn cho sản phẩm.

Cập nhật bảo mật liên tục: Nhờ cộng đồng lớn và sự hỗ trợ từ các tập đoàn lớn như Google, Meta, Microsoft, các lỗ hổng bảo mật thường được phát hiện và cập nhật nhanh chóng.

* Phát triển công nghệ mới

Trí tuệ nhân tạo và Machine Learning: Với các thư viện như TensorFlow.js, lập trình viên có thể xây dựng và huấn luyện các mô hình AI ngay trong trình duyệt mà không cần đến các ngôn ngữ phức tạp như Python.

IoT (Internet of Things): JavaScript, thông qua Johnny-Five và Node.js, có thể được sử dụng để lập trình và điều khiển các thiết bị IoT, robot, board mạch như Arduino, Raspberry Pi.

* Mang lại lợi ích kinh tế cho doanh nghiệp

Tối ưu hóa chi phí phát triển: Nhờ khả năng full-stack và cộng đồng lớn, doanh nghiệp có thể rút ngắn thời gian dự án và giảm chi phí thuê lập trình viên cho từng phần riêng biệt (frontend/backend).

Tăng tốc đưa sản phẩm ra thị trường: Khả năng phát triển nhanh chóng các ứng dụng MVP (Minimum Viable Product) nhờ vào hệ sinh thái phong phú giúp doanh nghiệp nhanh chóng đưa sản phẩm ra thị trường để thử nghiệm.

Ngôn ngữ lập trình sử dụng các biến làm phần giữ chỗ cho các giá trị dữ liệu thực tế. Ví dụ: trong một khối mã, nhà phát triển có thể viết x = 5 và y = x + 1. Khi mã chạy, máy tính sẽ tự động thay đổi x và y tương ứng thành 5 và 6 để thực hiện các hàm đối với chúng. Dữ liệu có thể là nhiều loại khác nhau, chẳng hạn như một chuỗi văn bản, số hoặc ngày. Đó là lý do hầu hết các ngôn ngữ lập trình đều cho phép bạn xác định kiểu biến. Khi đã xác định, kiểu biến sẽ không thay đổi; bạn không thể lưu trữ số trong các biến chuỗi.

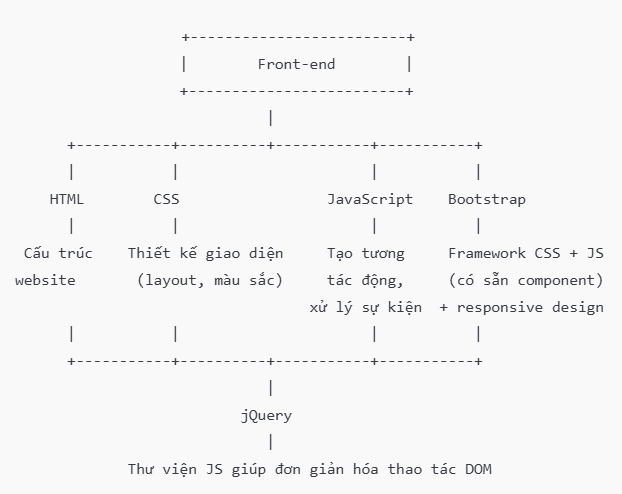
Ví dụ: nếu bạn cho chương trình biết rằng x và y là số và sau đó thực hiện phép toán x + y, máy tính sẽ biết dự kiến x và y là hai số và rồi cộng chúng. Mặt khác, nếu bạn xác định x và y là chuỗi, toán tử + sẽ nối hai chuỗi lại với nhau để tạo ra một cụm từ dài hơn.

Ngôn ngữ định kiểu yếu:JavaScript là một ngôn ngữ định kiểu yếu, có nghĩa là không cho phép người lập trình xác định kiểu biến. Một biến có thể lưu trữ bất kỳ kiểu dữ liệu nào trong thời gian chạy và các phép toán sẽ giả định kiểu của biến. Kết quả cũng có thể bị ép chuyển thành một kiểu dữ liệu khác. Ví dụ: một phép toán có thể trả về kết quả là chuỗi "5" thay vì số 5. Điều này có thể dẫn đến những sai lầm vô tình khi viết mã và lỗi trong mã do có lỗi về kiểu loại.

*2.1.5. So sánh giữa HTML, CSS và JavaScript*

Cả ba ngôn ngữ cùng nhau tạo ra trải nghiệm người dùng tích cực trên bất kỳ trang web nào. Trong khi HTML và CSS chủ yếu có thể điều chỉnh nội dung tĩnh, chúng có thể tích hợp với mã JavaScript phía máy khách để cập nhật nội dung một cách linh hoạt.

Ví dụ: khối mã tập lệnh trên trang HTML có thể chứa JavaScript bên trong. Sau đó, trình duyệt có thể xử lý cả HTML và mã JavaScript nội bộ khi trang HTML tải trong trình duyệt.



Hình 1.1: Sơ đồ tổng quan về mối quan hệ giữa HTML - CSS - JavaScript - jQuery – **Bootstrap**

**2.2.giới thiệu về jquery**

*2.2.1 Khái niệm tổng quan và sự ra đời của jQuery*

jQuery là một thư viện JavaScript mã nguồn mở phổ biến và mạnh mẽ, được phát hành lần đầu tiên vào tháng 1 năm 2006 bởi lập trình viên người Mỹ John Resig tại hội nghị BarCamp NYC. Ngay từ khi ra đời, jQuery đã tạo ra cuộc cách mạng lớn trong cộng đồng lập trình viên front-end nhờ khả năng giúp đơn giản hóa đáng kể việc lập trình JavaScript, đặc biệt là đối với các tác vụ thao tác với DOM (Document Object Model), quản lý sự kiện, tạo hiệu ứng động và thực hiện các yêu cầu AJAX.

Khẩu hiệu chính thức của jQuery là "Write less, do more" (Viết ít, làm nhiều), hàm ý rằng bạn có thể đạt được những kết quả phức tạp chỉ với một lượng mã ngắn gọn và dễ hiểu hơn nhiều so với khi sử dụng JavaScript thuần (vanilla JS). Đây cũng chính là một trong những điểm hấp dẫn lớn nhất khiến jQuery nhanh chóng trở thành một phần không thể thiếu của hàng triệu website trên toàn thế giới, đặc biệt trong giai đoạn từ 2008 đến 2015.

Vào đầu những năm 2000, lập trình web chủ yếu dựa vào JavaScript thuần, nhưng việc tương tác với DOM, tạo hiệu ứng động hoặc xử lý sự kiện thường gặp phải vấn đề khác biệt trình duyệt (cross-browser issues) và cú pháp JavaScript phức tạp, khó bảo trì. Trình duyệt Internet Explorer, Firefox, Safari, Opera hay Chrome đều có cách triển khai DOM khác nhau khiến cho việc phát triển web trở nên khó khăn, tốn thời gian và tiềm ẩn nhiều lỗi.

John Resig và nhóm cộng tác đã xây dựng jQuery với mục tiêu:

* Trừu tượng hóa các tác vụ phổ biến của JavaScript thành những hàm, phương thức dễ dùng và dễ hiểu hơn.
* Tự động xử lý các vấn đề về tương thích trình duyệt cho lập trình viên.
* Tăng tốc quá trình phát triển giao diện web, giúp developer tập trung vào logic và trải nghiệm người dùng thay vì phải lo xử lý từng trình duyệt riêng lẻ.

jQuery là thư viện JavaScript client-side (chạy trên trình duyệt) và chủ yếu được sử dụng trong:

* Phát triển các trang web có giao diện động và trực quan.
* Xây dựng các tương tác người dùng phức tạp (UI/UX).
* Thực hiện các thao tác gửi – nhận dữ liệu không đồng bộ với máy chủ (AJAX).
* Cung cấp hiệu ứng và hoạt họa cho các thành phần HTML.
* Giúp tối ưu hóa quá trình phát triển, rút ngắn thời gian và giảm thiểu lỗi.

Đến nay, mặc dù các framework và thư viện mới như React, Vue, Angular đã dần thay thế jQuery trong nhiều dự án phức tạp hoặc hiện đại, nhưng***jQuery vẫn rất phổ biến*** *trong nhiều dự án truyền thống, CMS như WordPress, Magento và hàng loạt các website tĩnh hoặc website quy mô vừa và nhỏ.*

*2.2.2. Các tính năng và lợi ích nổi bật của jQuery*

Truy vấn và thao tác DOM :DOM là mô hình dạng cây đại diện cho cấu trúc HTML của một trang web. Khi lập trình với JavaScript thuần, việc tìm kiếm, chỉnh sửa, thêm/xóa các phần tử DOM thường yêu cầu nhiều dòng lệnh dài và phức tạp. jQuery giúp đơn giản hóa điều này bằng cách:

* Sử dụng **cú pháp ngắn gọn và trực quan** dựa trên CSS selector.
* Cho phép lập trình viên truy xuất, thay đổi nội dung, thuộc tính, hoặc thêm mới các phần tử DOM chỉ với **1 hoặc 2 dòng lệnh**.

Vd

*A screen shot of a computer code

AI-generated content may be incorrect.*

Hình 2.1 :vd về DOM

**Xử lý sự kiện dễ dàng và chuẩn hóa:**Với jQuery, lập trình viên có thể **lắng nghe và xử lý các sự kiện người dùng** như click, double click, hover, keypress, focus, blur,... với cú pháp cực kỳ ngắn gọn, dễ hiểu, đồng thời đảm bảo tính tương thích trình duyệt:

A group of words on a white background

AI-generated content may be incorrect.

Hình 2.2: vd về jquery

**Hỗ trợ hiệu ứng động và animation:**jQuery cung cấp sẵn rất nhiều **hiệu ứng animation** giúp giao diện web sống động hơn mà không cần viết nhiều CSS hoặc JavaScript phức tạp:

* **Hiệu ứng ẩn/hiện:** fadeIn(), fadeOut(), slideUp(), slideDown(), toggle(),...
* **Hiệu ứng chuyển động:** animate() cho phép tạo animation tùy chỉnh các thuộc tính CSS như width, height, opacity,...

Khả năng mở rộng với plugin:jQuery có một kho plugin khổng lồ do cộng đồng phát triển, giúp thêm các tính năng sẵn có vào website mà không cần tự viết từ đầu. Ví dụ:

* jQuery UI: thư viện giao diện người dùng với các widget như datepicker, tabs, accordion.
* Slick slider, Owl Carousel: tạo các slider, carousel đẹp mắt.
* Magnific Popup: tạo popup lightbox cho ảnh và video.
* jQuery Validation: hỗ trợ kiểm tra form (validate) đơn giản.

Các plugin đều dễ tích hợp và sử dụng chỉ với vài dòng mã.

*2.2.3. Ưu nhược điểm, ứng dụng thực tế và xu hướng của jQuery*

Ưu điểm nổi bật của jQuery

* Tiết kiệm thời gian phát triển: jQuery giúp lập trình viên giảm thiểu đáng kể số lượng dòng mã cần viết.
* Dễ học, dễ sử dụng: Phù hợp với cả người mới bắt đầu và lập trình viên lâu năm nhờ cú pháp trực quan.
* Tương thích đa trình duyệt: Hạn chế tối đa lỗi khi chạy trên các trình duyệt khác nhau, đặc biệt là Internet Explorer - trình duyệt từng gây đau đầu cho nhiều lập trình viên.
* Cộng đồng mạnh: jQuery có hàng triệu lập trình viên sử dụng, hàng ngàn plugin và tài liệu hướng dẫn phong phú.
* Tích hợp dễ dàng với HTML/CSS và CMS: Được sử dụng phổ biến trong WordPress, Magento, Joomla,... và dễ kết hợp với Bootstrap.

Nhược điểm của jQuery

* Hiệu suất chưa tối ưu cho dự án lớn: Với các ứng dụng SPA (Single Page Application) hoặc dự án web phức tạp, jQuery không tối ưu bằng các framework hiện đại như React, Vue hoặc Angular.
* Tệp thư viện nặng hơn JavaScript thuần: jQuery minified có kích thước khoảng 80-90KB, điều này có thể ảnh hưởng đến tốc độ tải trang nếu sử dụng không hợp lý.
* Xu hướng giảm dần trong dự án mới: Ngày nay, JavaScript ES6+ và các framework mới cung cấp nhiều tính năng tương tự jQuery nhưng tối ưu và hiện đại hơn.

Ứng dụng thực tế của jQuery

* Dùng trong website tĩnh hoặc các website nhỏ và trung bình.
* Xây dựng nhanh các tính năng như form validation, slider, popup, menu dropdown, tự động load dữ liệu với AJAX.
* Phổ biến trong các CMS như WordPress, các template website hoặc giao diện thương mại điện tử.
* Dùng cho các dự án cần tối ưu thời gian và chi phí phát triển.

Xu hướng và vai trò hiện nay

* Trong các dự án cũ hoặc bảo trì: jQuery vẫn giữ vai trò quan trọng khi nhiều website truyền thống hoặc CMS vẫn dựa trên nó.
* Trong các dự án nhỏ/ít phức tạp: jQuery vẫn là lựa chọn tối ưu để xây dựng nhanh và tiết kiệm chi phí.
* Trong các dự án mới: jQuery đang dần được thay thế bởi JavaScript thuần (ES6+), hoặc các framework hiện đại có khả năng quản lý state và build SPA hiệu quả hơn.

**2.3.bootstrap**

*2.3.1.lịch sử bootstrap*

Bootstrap, ban đầu có tên là Twitter Blueprint, được Mark Otto và Jacob Thornton tại [Twitter](https://en.wikipedia.org/wiki/Twitter) phát triển như một khuôn khổ để khuyến khích tính nhất quán giữa các công cụ nội bộ. Trước Bootstrap, nhiều thư viện khác nhau đã được sử dụng để phát triển giao diện, dẫn đến sự không nhất quán và gánh nặng bảo trì cao. Theo Otto:

Một nhóm nhỏ các nhà phát triển và tôi đã cùng nhau thiết kế và xây dựng một công cụ nội bộ mới và thấy cơ hội để làm nhiều hơn thế nữa. Thông qua quá trình đó, chúng tôi thấy mình đang xây dựng một thứ gì đó quan trọng hơn nhiều so với một công cụ nội bộ khác. Vài tháng sau, chúng tôi đã có phiên bản đầu tiên của Bootstrap như một cách để ghi lại và chia sẻ các mẫu thiết kế và tài sản chung trong công ty. [[ 7 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-bootstrap_a_list_apart-7)

Sau một vài tháng phát triển bởi một nhóm nhỏ, nhiều nhà phát triển tại Twitter bắt đầu đóng góp cho dự án như một phần của Tuần lễ Hack, một tuần theo phong cách [hackathon](https://en.wikipedia.org/wiki/Hackathon) dành cho nhóm phát triển Twitter. Nó được đổi tên từ Twitter Blueprint thành Twitter Bootstrap và được phát hành dưới dạng một dự án nguồn mở vào ngày 19 tháng 8 năm 2011. [[ 8 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-v1-release-8) Nó vẫn tiếp tục được duy trì bởi Otto, Thornton, một nhóm nhỏ các nhà phát triển cốt lõi và một cộng đồng lớn những người đóng góp. [[ 9 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-about-9)

Vào ngày 31 tháng 1 năm 2012, Bootstrap 2 đã được phát hành, bổ sung hỗ trợ tích hợp cho Glyphicons, một số thành phần mới cũng như thay đổi nhiều thành phần hiện có. Phiên bản này hỗ trợ , nghĩa là bố cục của các trang web điều chỉnh động, có tính đến các đặc điểm của thiết bị được sử dụng (cho dù là máy tính để bàn, máy tính bảng, điện thoại di động). Ngay trước khi phát hành Bootstrap 2.1.2, Otto và Thornton đã rời Twitter, nhưng cam kết tiếp tục làm việc trên Bootstrap như một dự án độc lập.  Vào ngày 19 tháng 8 năm 2013, Bootstrap 3 đã được phát hành. Nó thiết kế lại các thành phần để sử dụng [thiết kế phẳng](https://en.wikipedia.org/wiki/Flat_design)và phương pháp tiếp cận [ưu tiên thiết bị di động](https://en.wikipedia.org/wiki/Responsive_web_design#Mobile_first,_unobtrusive_JavaScript,_and_progressive_enhancement) . [[ 12 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-12) Bootstrap 3 có hệ thống plugin mới với các sự kiện có không [gian tên](https://en.wikipedia.org/wiki/Namespace) . Bootstrap 3 đã ngừng hỗ trợ Internet Explorer 7 và Firefox 3.6, nhưng có một [polyfill](https://en.wikipedia.org/wiki/Polyfill_(programming)) tùy chọn cho các trình duyệt này. [[ 13 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-v3-release-13) Bootstrap 3 cũng là phiên bản đầu tiên được phát hành theo tổ chức twbs trên GitHub thay vì Twitter. [[ 14 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-14)

Otto đã công bố Bootstrap 4 vào ngày 29 tháng 10 năm 2014. [[ 15 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-v3.3.0-release-15) Phiên bản alpha đầu tiên của Bootstrap 4 được phát hành vào ngày 19 tháng 8 năm 2015. [[ 16 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-v4.0.0-alpha-release-16) Phiên bản beta đầu tiên được phát hành vào ngày 10 tháng 8 năm 2017. [[ 17 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-v4.0.0-beta-release-17) Otto đã tạm dừng công việc trên Bootstrap 3 vào ngày 6 tháng 9 năm 2016 để giải phóng thời gian làm việc trên Bootstrap 4. Bootstrap 4 đã được hoàn thiện vào ngày 18 tháng 1 năm 2018. [[ 18 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-18)

Những thay đổi đáng kể bao gồm:

* Viết lại toàn bộ mã
* Thay thế [Less](https://en.wikipedia.org/wiki/Less_(stylesheet_language)) bằng [Sass](https://en.wikipedia.org/wiki/Sass_(stylesheet_language))
* Thêm vào đó Reboot, một tập hợp các thay đổi CSS cụ thể của từng phần tử trong một tệp duy nhất, dựa trênNormalize
* Bỏ hỗ trợ cho [IE8](https://en.wikipedia.org/wiki/Internet_Explorer_8) , [IE9](https://en.wikipedia.org/wiki/Internet_Explorer_9) và [iOS 6](https://en.wikipedia.org/wiki/IOS_6)
* Hỗ trợ [CSS Flexible Box](https://en.wikipedia.org/wiki/CSS_Flexible_Box_Layout)
* Thêm tùy chọn tùy chỉnh điều hướng
* Thêm tiện ích khoảng cách và kích thước phản hồi
* Chuyển đổi từ đơn vị [pixel](https://en.wikipedia.org/wiki/Pixel) trong CSS sang [đơn vị ems gốc](https://en.wikipedia.org/wiki/Root_em)
* Tăng kích thước phông chữ toàn cầu từ 14px lên 16px để tăng khả năng đọc
* Bỏ các thành phần panel, thumbnail, pager, vàwell
* Bỏ phông Glyphiconschữ biểu tượng
* Số lượng lớn [[*định lượng*](https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Dates_and_numbers)] các lớp tiện ích
* Cải thiện kiểu dáng biểu mẫu, nút, menu thả xuống, đối tượng phương tiện và lớp hình ảnh

Bootstrap 4 hỗ trợ các phiên bản mới nhất của [Google Chrome](https://en.wikipedia.org/wiki/Google_Chrome) , [Firefox](https://en.wikipedia.org/wiki/Firefox) , [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer) , [Opera](https://en.wikipedia.org/wiki/Opera_(web_browser)) và [Safari](https://en.wikipedia.org/wiki/Safari_(web_browser)) (trừ Windows). Ngoài ra, nó còn hỗ trợ trở lại [IE10 và](https://en.wikipedia.org/wiki/Internet_Explorer_10)[Firefox](https://en.wikipedia.org/wiki/Firefox) Extended Support Release (ESR) mới nhất

Bootstrap 5 chính thức được phát hành vào ngày 5 tháng 5 năm 2021. [[ 20 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-20)[[ 21 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-21)

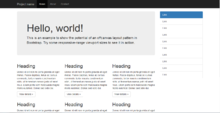
Những thay đổi chính bao gồm

* Thành phần menu offcanvas mới
* Loại bỏ sự phụ thuộc vào [jQuery](https://en.wikipedia.org/wiki/JQuery) để ủng hộ JavaScript thuần túy
* Viết lại lưới để hỗ trợ các rãnh phản hồi và các cột được đặt bên ngoài các hàng
* Di chuyển tài liệu từ [Jekyll](https://en.wikipedia.org/wiki/Jekyll_(software)) sang [Hugo](https://en.wikipedia.org/wiki/Hugo_(software))
* Bỏ hỗ trợ cho [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer)[[ 23 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-23)
* Di chuyển cơ sở hạ tầng thử nghiệm từ [QUnit](https://en.wikipedia.org/wiki/QUnit) sang [Jasmine](https://en.wikipedia.org/wiki/Jasmine_(JavaScript_testing_framework))
* Thêm bộ biểu tượng SVG tùy chỉnh [[ 24 ]](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)#cite_note-24)
* Thêm thuộc tính tùy chỉnh CSS
* API được cải thiện
* Hệ thống lưới nâng cao
* Cải thiện việc tùy chỉnh tài liệu
* Biểu mẫu cập nhật
* Hỗ trợ RTL
* Được xây dựng trong hỗ trợ chế độ tố

*2.3.2.Đặc trưng của bootstrap*

*Bootstrap là một thư viện HTML, CSS và JS tập trung vào việc đơn giản hóa quá trình phát triển các trang web thông tin (khác với*[*các ứng dụng web*](https://en.wikipedia.org/wiki/Web_application)*). Mục đích chính của việc thêm nó vào một dự án web là áp dụng các lựa chọn về màu sắc, kích thước, phông chữ và bố cục của Bootstrap cho dự án đó. Do đó, yếu tố chính là liệu các nhà phát triển phụ trách có thấy những lựa chọn đó phù hợp với sở thích của họ hay không. Sau khi được thêm vào một dự án, Bootstrap cung cấp các định nghĩa kiểu cơ bản cho tất cả*[*các phần tử HTML*](https://en.wikipedia.org/wiki/HTML_element)*. Kết quả là giao diện thống nhất cho văn xuôi, bảng và các phần tử biểu mẫu trên*[*các trình duyệt web*](https://en.wikipedia.org/wiki/Web_browser)*. Ngoài ra, các nhà phát triển có thể tận dụng các lớp CSS được xác định trong Bootstrap để tùy chỉnh thêm giao diện của nội dung của họ. Ví dụ: Bootstrap đã cung cấp cho các bảng màu sáng và tối, tiêu đề trang,*[*trích dẫn*](https://en.wikipedia.org/wiki/Pull_quote)*nổi bật hơn và văn bản có tô sáng.*

*Bootstrap cũng đi kèm với một số thành phần JavaScript không yêu cầu các thư viện khác như*[*jQuery*](https://en.wikipedia.org/wiki/JQuery)*. Chúng cung cấp các thành phần giao diện người dùng bổ sung như*[*hộp thoại*](https://en.wikipedia.org/wiki/Dialog_box)*,*[*chú giải công cụ*](https://en.wikipedia.org/wiki/Tooltip)*, thanh tiến trình, menu thả xuống điều hướng và băng chuyền. Mỗi thành phần Bootstrap bao gồm một cấu trúc HTML, khai báo CSS và trong một số trường hợp là mã JavaScript đi kèm. Chúng cũng mở rộng chức năng của một số thành phần giao diện hiện có, bao gồm ví dụ như hàm tự động hoàn thành cho các trường nhập liệu.*

*[](https://en.wikipedia.org/wiki/File:Twitter_Bootstrap_Under_Firefox_32.png)*

*Hinh3.1:Ví dụ về trang web sử dụng khung Bootstrap được hiển thị trong*[*Firefox*](https://en.wikipedia.org/wiki/Firefox)

*Các thành phần nổi bật nhất của Bootstrap là các thành phần bố cục của nó, vì chúng ảnh hưởng đến toàn bộ trang web. Thành phần bố cục cơ bản được gọi là "Container", vì mọi thành phần khác trong trang đều được đặt trong đó. Các nhà phát triển có thể chọn giữa một container có chiều rộng cố định và một container có chiều rộng linh hoạt. Trong khi container sau luôn lấp đầy chiều rộng bằng trang web, container trước sử dụng một trong năm chiều rộng cố định được xác định trước, tùy thuộc vào kích thước của màn hình hiển thị trang:*

* *Nhỏ hơn 576 pixel*
* *576–768 điểm ảnh*
* *768–992 điểm ảnh*
* *992–1200 điểm ảnh*
* *1200–1400 điểm ảnh*
* *Lớn hơn 1400 pixel*

*Khi vùng chứa đã được đặt đúng vị trí, các thành phần bố cục Bootstrap khác sẽ triển khai bố cục CSS Flexbox bằng cách xác định các hàng và cột.*

*Phiên bản Bootstrap được biên dịch trước có sẵn dưới dạng một tệp CSS và ba tệp JavaScript có thể dễ dàng thêm vào bất kỳ dự án nào. Tuy nhiên, dạng thô của Bootstrap cho phép các nhà phát triển triển khai tùy chỉnh và tối ưu hóa kích thước hơn nữa. Dạng thô này là mô-đun, nghĩa là nhà phát triển có thể xóa các thành phần không cần thiết, áp dụng một chủ đề và sửa đổi các tệp*[*Sass*](https://en.wikipedia.org/wiki/Sass_(stylesheet_language))*chưa biên dịch .*

*2.3.3.Nội dung*

*Bootstrap là một framework front-end mã nguồn mở toàn diện bao gồm hệ thống lưới 12 cột chuẩn responsive linh hoạt cho phép tùy chỉnh bố cục theo từng kích thước màn hình (extra small, small, medium, large, extra large, extra extra large), tích hợp sẵn hàng loạt thành phần giao diện người dùng (UI components) như button, button group, forms, form controls, input groups, custom forms, navs, navbars, dropdowns, modals, cards, accordions, tabs, pills, breadcrumb, pagination, alerts, badges, progress bars, spinners, tooltips, popovers, carousel, collapse, scrollspy, offcanvas, list groups, toast, figures, media objects, jumbotron, cùng với các tiện ích CSS như utilities về margins, paddings, display, colors, typography, sizing, positioning, borders, shadows, flexbox, grid utilities, text utilities, visibility utilities, responsive embed, và thư viện JavaScript đi kèm giúp xử lý các hành vi động như modal, dropdown, collapse, toast, tooltip, popover, carousel, scrollspy, affix, hỗ trợ sẵn theme và icon mở rộng thông qua Bootstrap Icons, tuân thủ nguyên tắc mobile-first và thiết kế chuẩn responsive, đảm bảo tính tương thích đa trình duyệt (cross-browser compatibility), thân thiện với người dùng, đồng thời tối ưu hóa quy trình phát triển giao diện web hiện đại, nhanh chóng, dễ bảo trì và dễ mở rộng cho các dự án quy mô từ nhỏ đến lớn.*

**Chương 3. THIẾT KẾ WEBSITE**

**3.1. Ý tưởng thiết kế**

* Giao diện hiện đại, rõ ràng với bố cục chuẩn thương mại điện tử gồm: Header, Banner, Sản phẩm nổi bật, Footer.
* Header: Logo + menu điều hướng (Trang chủ, Giới thiệu, Liên hệ, Giỏ hàng).
* Banner: Hình ảnh lớn tạo điểm nhấn, thu hút người dùng.
* Sản phẩm nổi bật: Hiển thị danh sách sản phẩm theo dạng lưới với ảnh, tên, giá và nút "Chi tiết" cho từng sản phẩm.
* Footer: Logo, liên kết chính sách, mạng xã hội (Facebook, Instagram, Zalo).
* Tông màu: Chủ đạo đỏ hoặc đậm màu, kết hợp trắng/xám để làm nổi bật sản phẩm.
* Phong cách: Gọn gàng, dễ sử dụng, hỗ trợ responsive trên mọi thiết bị.
* Chức năng chính: Điều hướng nhanh, xem chi tiết sản phẩm, liên kết mạng xã hội.

**3.2. Sơ đồ website**

**A screenshot of a computer program

AI-generated content may be incorrect.**

Hifnh3.1: sơ đồ website

Trang chủ là trung tâm, dẫn tới các trang chi tiết sản phẩm riêng lẻ.

Menu điều hướng liên kết đến các trang thông tin khác (Giới thiệu, Liên hệ, Giỏ hàng).

Mỗi sản phẩm nổi bật đều có liên kết riêng sang trang chi tiết sản phẩm.

Footer cung cấp liên kết tới chính sách và các mạng xã hội.

**3.3. Giao diện chính các trang chính**

*3.3.1. Trang chủ*

**A screenshot of a advertisement

AI-generated content may be incorrect.**

Hình 3.2:Dao diện trang chủ

*3.3.2. Trang chi tiết sản phẩm*

*A screen shot of a phone

AI-generated content may be incorrect.*

Hình 3.3:Trang chi tiết sản phẩm

*3.3.3. Trang đặt hàng*

*3.3.4. Trang đăng nhập*

*A screenshot of a computer

AI-generated content may be incorrect.*

Hình 3.4: Trang đăng nhập

**3.4. Code của một số trang chính**

<!DOCTYPE html>

<html lang="vi">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Vanh Mobile</title>

    <link rel="stylesheet" href="c.css">

</head>

<body>

    <header>

        <div class="logo">

            <h1>Vanh Mobile</h1>

        </div>

        <nav>

            <ul>

                <li><a href="#">Trang chủ</a></li>

                <li><a href="gt.html">Giới thiệu</a></li>

                <li><a href="lh.html">Liên hệ</a></li>

                <li><a href="gh.html">Giỏ hàng (0)</a></li>

            </ul>

        </nav>

    </header>

    <div>

        <img src="anh7.png" alt="" height="50%" width="100%">

    </div>

    <section class="products">

        <h2>Sản phẩm nổi bật</h2>

        <div class="product-list">

            <div class="product-item">

                <img src="./may1.png" alt="Điện thoại 1" height="250px " width=" 150px">

                <h3>iPhone 13</h3>

                <p>Giá: 16,000,000 VNĐ  <br><br></p>

                <a href="chitietip13.html" class="add-to-cart" >Chi tiết</a>

            </div>

            <div class="product-item">

                <img src="./anh8.png" alt="Điện thoại 2" height="250px " width=" 150px"9o,>

                <h3>iphone 14</h3>

                <p>Giá: 18,000,000 VNĐ<br><br></p>

                <a href="chitietip14.html" class="add-to-cart" >Chi tiết</a>

            </div>

            <div class="product-item">

                <img src="./anh9.png" alt="Điện thoại 2" height="250px " width=" 150px"9o,>

                <h3>iphone 15</h3>

                <p>Giá: 20,000,000 VNĐ<br><br></p>

                <a href="chitietip15.html" class="add-to-cart" >Chi tiết</a>

            </div>

            <div class="product-item">

                <img src="./anh4.png" alt="Điện thoại 2" height="250px " width=" 150px"9o,>

                <h3>iphone 16</h3>

                <p>Giá: 22,000,000 VNĐ<br><br></p>

                <a href="chitietip16.html" class="add-to-cart" >Chi tiết</a>

            </div>

            <div class="product-item">

                <img src="./anh6.png" alt="Điện thoại 3"  height="250px " width=" 160px">

                <h3>iphone 16Promax</h3>

                <p>Giá: 32,000,000 VNĐ<br><br></p>

                <a href="chitietip16pr.html" class="add-to-cart" >Chi tiết</a>

            </div>

            <div class="product-item">

                <img src="./anh3.png" alt="Điện thoại N"  height="250px " width=" 160px">

                <h3> Galaxy S25UlLTRA</h3>

                <p>Giá: 32,000,000 VNĐ<br><br></p>

                <a href="chitiets25.html" class="add-to-cart" >Chi tiết</a>

            </div>

        </div>

    </section>

    <footer>

        <div class="footer-content">

            <div class="footer-logo">

                <h3>Vanh Mobile</h3>

            </div>

            <div class="footer-links">

                <ul>

                    <li><a href="#">Điều khoản sử dụng</a></li>

                    <li><a href="#">Chính sách bảo mật</a></li>

                    <li><a href="#">Hỗ trợ khách hàng</a></li>

                </ul>

            </div>

            <div class="footer-social">

                <a href="https://www.facebook.com/viet.anh.183298?locale=vi\_VN">Facebook</a>

                <a href="https://www.instagram.com/Viet%20anh/?fbclid=IwY2xjawIbjRZleHRuA2FlbQIxMAABHWwCkJqL7k\_n27MZ5Efnl5BybXifb27gGAvBGdDBxkfTyvBK5lSlRm-2uw\_aem\_Qs6O3RXWFqtzYya6-sWqdg#">Instagram</a>

                <a href="bip">Zalo</a>

            </div>

        </div>

    </footer>

    <script src="c.js"></script>

</body>

</html>

**KẾT LUẬN**

Trong bối cảnh nền kinh tế số đang phát triển mạnh mẽ và thương mại điện tử ngày càng đóng vai trò quan trọng trong hành vi tiêu dùng hiện đại, việc thiết kế và phát triển một website kinh doanh điện thoại chuyên nghiệp không chỉ giúp các doanh nghiệp nâng cao hình ảnh thương hiệu mà còn tạo ra một kênh bán hàng hiệu quả, dễ tiếp cận và tiện lợi cho khách hàng. Qua quá trình nghiên cứu và thực hiện đề tài “Thiết kế website kinh doanh điện thoại” trong môn học Thiết kế lập trình Front-End, nhóm chúng tôi đã nắm vững và vận dụng một cách hiệu quả các kiến thức nền tảng và chuyên sâu liên quan đến HTML, CSS, JavaScript, jQuery và Bootstrap để xây dựng một giao diện website thân thiện, hiện đại và phù hợp với xu hướng công nghệ hiện nay.

Website được xây dựng đảm bảo tính thẩm mỹ cao, giao diện được thiết kế rõ ràng, mạch lạc, dễ sử dụng, đáp ứng tốt các yêu cầu cơ bản về trải nghiệm người dùng (UX) và giao diện người dùng (UI). Bên cạnh đó, việc tích hợp các chức năng động như hiệu ứng khi tương tác với nút bấm, bộ lọc sản phẩm, slideshow hình ảnh, form đăng ký – đăng nhập, giỏ hàng tạm thời cùng các tính năng hỗ trợ tìm kiếm sản phẩm đã giúp cho website mang tính ứng dụng thực tế cao và sẵn sàng phục vụ nhu cầu của các doanh nghiệp kinh doanh điện thoại.

Ngoài ra, website cũng được tối ưu hóa khả năng hiển thị trên nhiều loại thiết bị (responsive) nhờ việc áp dụng framework Bootstrap, giúp khách hàng có thể dễ dàng truy cập và sử dụng dịch vụ từ máy tính, máy tính bảng cho đến điện thoại di động. Qua đó, doanh nghiệp có thể tiếp cận được nhiều đối tượng khách hàng hơn, đồng thời tạo điều kiện thuận lợi cho việc quảng bá sản phẩm và gia tăng doanh số bán hàng. trong quá trình thực hiện đề tài, nhóm cũng nhận thấy rằng để một website hoàn chỉnh và chuyên nghiệp hơn, ngoài phần giao diện Front-End thì cần phối hợp chặt chẽ với lập trình Back-End và cơ sở dữ liệu để quản lý thông tin sản phẩm, đơn hàng, người dùng và các nghiệp vụ liên quan khác. Do giới hạn về thời gian và phạm vi môn học nên đề tài chỉ tập trung vào phần thiết kế giao diện và các chức năng cơ bản ở phía người dùng.

Kết thúc đề tài này tôi đã tích lũy thêm được nhiều kinh nghiệm quý báu về quy trình thiết kế giao diện website, cách vận dụng linh hoạt các công nghệ Front-End hiện đại và hiểu rõ hơn về tầm quan trọng của trải nghiệm người dùng trong thiết kế web. Những kiến thức và kỹ năng đạt được sẽ là nền tảng vững chắc để nhóm tiếp tục phát triển các dự án web hoàn chỉnh hơn trong tương lai.

**TÀI LIỆU THAM KHẢO**

[1] Responsive Web Design with HTML5 and CSS, Fourth Edition, 2022.

[2] Learning Web Design - HTML5, Shanmugapriyaa S, Self-Published, 2022.

[3] Web Development & Design Foundations with HTML5, 10th Edition, 2022.