**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

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

**TÊN ĐỀ TÀI: XÂY DỰNG WEBSITE PROFILE CÔNG VIỆC BẰNG JAVASCRIPT, JQUERY VÀ BOTSTRAP**

**Giảng viên hướng dẫn: ThS. Lê Văn Phong**

**Sinh viên thực hiện:**  **Nguyễn Tiến Lực**

**Hà Nội, 2024**

**LỜI NÓI ĐẦU**

Chào thầy/cô và các bạn!

Trong thời đại công nghệ số phát triển như hiện nay, việc xây dựng một website cá nhân hoặc cá nhân hóa về công việc không chỉ là một nhu cầu mà còn là một cách để mỗi cá nhân có thể giới thiệu bản thân và kỹ năng của mình đến với mọi người một cách trực quan và hiệu quả. Đây cũng là cơ hội để thể hiện sự sáng tạo và khả năng kỹ năng lập trình của mình.

Đề tài "Xây dựng website profile công việc bằng JavaScript, jQuery và Bootstrap" mà em đã chọn là một sự kết hợp hài hòa giữa những công nghệ phổ biến và mạnh mẽ nhất hiện nay trong lĩnh vực phát triển web. JavaScript và jQuery là hai thư viện lập trình phổ biến giúp cho việc xử lý sự kiện và tương tác trên website trở nên linh hoạt và dễ dàng hơn. Bootstrap, với hệ thống Grid và các thành phần UI được chuẩn hóa, giúp tăng tốc quá trình phát triển và đảm bảo tính responsive của trang web.

Trong đề tài này, em sẽ hướng đến việc thiết kế và phát triển một website profile cá nhân hoặc portfolio chuyên nghiệp, từ việc cấu trúc layout, thiết kế giao diện đến việc tối ưu hóa trải nghiệm người dùng và hiệu suất của trang web. Bằng cách này, em mong muốn có thể trình bày một cách rõ ràng và hệ thống về quy trình làm việc của mình, cũng như kỹ năng sử dụng các công nghệ front-end để tạo ra sản phẩm cuối cùng đáp ứng được các yêu cầu của người dùng và nhu cầu của thị trường hiện nay.

Đề tài này không chỉ là một bài tập lớn mà còn là một cơ hội để em rèn luyện và phát triển kỹ năng thực tế trong lĩnh vực thiết kế và lập trình front-end. Em hy vọng rằng báo cáo này sẽ mang lại giá trị và có ích cho thầy/cô trong việc đánh giá và cũng như cho các bạn sinh viên quan tâm đến lĩnh vực này.

Em xin chân thành cảm ơn sự quan tâm và hỗ trợ của thầy/cô trong suốt quá trình thực hiện đề tài này.

Trân trọng!

MỤC LỤC

[**CHƯƠNG I: GIỚI THIỆU VỀ THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM** 8](#_Toc169733013)

[**1.1.** **Các khái niệm cơ bản** 8](#_Toc169733014)

[**1.2.** **Phân biệt Web tĩnh, Web động** 8](#_Toc169733015)

[1.3. **Một số thuật ngữ** (Hosting, Tên miền-Domain Name-Web Server Name, HomePage, 8](#_Toc169733016)

[**1.4.** **Một số công cụ dùng thiết kế Web** 9](#_Toc169733017)

[**Kết luận chương 1** 10](#_Toc169733018)

[**CHƯƠNG II: CƠ SỞ LÝ THUYẾT XÂY DỰNG WEBSITE PROFILE CÔNG VIỆC** 11](#_Toc169733019)

[**2.1. Ngôn ngữ JavaScript** 11](#_Toc169733020)

[***2.1.1. Giới thiệu về JavaScript*** 11](#_Toc169733021)

[***2.1.2. Cách dùng JavaScript*** 11](#_Toc169733022)

[**2.2. jQuery** 11](#_Toc169733023)

[***2.2.1. Giới thiệu về jQuery*** 11](#_Toc169733024)

[***2.2.2. Cách dùng jQuery*** 11](#_Toc169733025)

[**2.3. Botstrap** 12](#_Toc169733026)

[***2.3.1. Giới thiệu về Botstrap*** 12](#_Toc169733027)

[***2.3.2. Các dùng Botstrap*** 12](#_Toc169733028)

[Kết luận chương 2 12](#_Toc169733029)

[**3.1. Ý tưởng xây dựng Website** 13](#_Toc169733030)

[**3.2. Xây dựng bố cục trang Web** 13](#_Toc169733031)

[**Thiết kế và lập trình Website** 14](#_Toc169733032)

[+ Phần HTML: Chức năng của Header: 15](#_Toc169733033)

[+ Phần jQuery: Chức năng chính của plugin counterUp: 16](#_Toc169733034)

[**Kết quả xây dựng trang Web** 19](#_Toc169733035)

[Phần Preloader 19](#_Toc169733036)

[Header cho thiết bị di động (mobile header) 20](#_Toc169733037)

[Header cho desktop 21](#_Toc169733038)

[Các hàm tiện ích (s và l) 44](#_Toc169733039)

[Các hằng số và biến 45](#_Toc169733040)

[Các thành phần Bootstrap 46](#_Toc169733041)

[Sự kiện khi tài liệu được tải 48](#_Toc169733042)

[Khởi tạo Bộ kiểm tra form 49](#_Toc169733043)

[Xử lý gửi form 49](#_Toc169733044)

[Sự kiện DOMContentLoaded 50](#_Toc169733045)

[Xử lý khi nhấn nút gửi (submit) 51](#_Toc169733046)

[HTML tương ứng 52](#_Toc169733047)

[Giải thích: 52](#_Toc169733048)

[Kết luận chương 3 65](#_Toc169733049)

[Ưu điểm của Profile Công việc 66](#_Toc169733050)

[Nhược điểm của Profile Công việc 66](#_Toc169733051)

[Hướng phát triển của Profile Công việc 66](#_Toc169733052)

[Hướng dẫn xếp tài liệu tham khảo 70](#_Toc169733053)

**MỤC LỤC HÌNH ẢNH**

**BẢNG CÁC TỪ VIẾT TẮT**

**(Nếu có)**

|  |  |
| --- | --- |
| **TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| HTML | Hyper Text Markup Language |
|  |  |

# **CHƯƠNG I: GIỚI THIỆU VỀ THIẾT KẾ WEB VÀ TRIỂN KHAI HỆ THỐNG PHẦN MỀM**

## **Các khái niệm cơ bản**

**Thiết kế Web**

Thiết kế web là quá trình tạo ra các trang web, bao gồm việc lập kế hoạch, thiết kế giao diện và tổ chức nội dung. Mục tiêu của thiết kế web là tạo ra một trang web thân thiện với người dùng, dễ sử dụng và thu hút người truy cập.

**Hệ thống phần mềm**

Hệ thống phần mềm là tập hợp các chương trình máy tính, tài liệu và dữ liệu liên quan, được thiết kế để thực hiện một số chức năng cụ thể. Việc triển khai hệ thống phần mềm bao gồm các bước từ thiết kế, phát triển, kiểm thử đến bảo trì hệ thống.

## **Phân biệt Web tĩnh, Web động**

**Web tĩnh**

Web tĩnh là các trang web mà nội dung của nó không thay đổi hoặc chỉ thay đổi khi được chỉnh sửa bởi nhà phát triển. Các trang web tĩnh thường được tạo ra bằng HTML và CSS, không có sự tương tác với cơ sở dữ liệu hoặc xử lý dữ liệu động.

**Web động**

Web động là các trang web có nội dung thay đổi linh hoạt dựa trên các tương tác của người dùng hoặc các sự kiện bên ngoài. Web động sử dụng các ngôn ngữ lập trình phía máy chủ (như PHP, ASP.NET) và cơ sở dữ liệu để tạo ra các trang web tùy biến và có khả năng tương tác cao.

## **Một số thuật ngữ** (Hosting, Tên miền-Domain Name-Web Server Name, HomePage,

Web Site, WebServer, URL-Uniform Resource Location, Browser, …)

**Hosting**

Hosting là dịch vụ lưu trữ dữ liệu của trang web trên một máy chủ, giúp trang web có thể truy cập được qua internet. Nhà cung cấp dịch vụ hosting cung cấp không gian lưu trữ, băng thông và các dịch vụ hỗ trợ khác.

**Tên miền (Domain Name)**

Tên miền là địa chỉ duy nhất trên internet giúp người dùng dễ dàng truy cập vào trang web của bạn. Ví dụ, "example.com" là một tên miền.

**Web Server**

Web server là máy chủ chứa các tệp tin của trang web và phục vụ chúng cho người dùng thông qua giao thức HTTP hoặc HTTPS. Các web server phổ biến bao gồm Apache, Nginx và Microsoft IIS.

**HomePage**

HomePage là trang chính của một trang web, thường là điểm khởi đầu và chứa các liên kết đến các phần khác của trang web.

**Web Site**

Web Site là tập hợp các trang web con có liên quan, được tổ chức dưới một tên miền duy nhất và phục vụ một mục đích chung.

**URL (Uniform Resource Locator)**

URL là địa chỉ duy nhất của một tài nguyên trên internet, bao gồm tên miền và đường dẫn đến tài nguyên đó.

**Browser**

Browser (trình duyệt web) là phần mềm cho phép người dùng truy cập và xem nội dung trên các trang web. Các trình duyệt phổ biến bao gồm Google Chrome, Mozilla Firefox, Microsoft Edge và Safari.

* 1. **Một số công cụ dùng thiết kế Web (Visual Studio code, Adobe Dreamweaver, …)**

**Visual Studio Code**

Visual Studio Code là một trình soạn thảo mã nguồn mở và miễn phí, hỗ trợ nhiều ngôn ngữ lập trình và có nhiều tiện ích mở rộng giúp việc phát triển web trở nên dễ dàng và hiệu quả.

**Adobe Dreamweaver**

Adobe Dreamweaver là một công cụ phát triển web chuyên nghiệp, cung cấp giao diện đồ họa để thiết kế và chỉnh sửa các trang web. Dreamweaver hỗ trợ cả web tĩnh và web động, với nhiều tính năng mạnh mẽ giúp tăng tốc độ phát triển web.

**Các công cụ khác**

Ngoài Visual Studio Code và Adobe Dreamweaver, còn nhiều công cụ khác hỗ trợ thiết kế và phát triển web như Sublime Text, Atom, Notepad++, và các IDE chuyên dụng như WebStorm, PHPStorm.

## **Kết luận chương 1**

Chương 1 đã giới thiệu tổng quan về các khái niệm cơ bản trong thiết kế web và triển khai hệ thống phần mềm. Chúng ta đã phân biệt rõ ràng giữa web tĩnh và web động, nắm bắt được một số thuật ngữ quan trọng và tìm hiểu về các công cụ hỗ trợ thiết kế web. Kiến thức này là nền tảng quan trọng để chúng ta tiếp tục các chương tiếp theo, nơi sẽ đi sâu vào việc xây dựng một trang web đọc truyện sử dụng JavaScript, jQuery và Bootstrap.

**CHƯƠNG II: CƠ SỞ LÝ THUYẾT XÂY DỰNG WEBSITE PROFILE CÔNG VIỆC**

**2.1. Ngôn ngữ JavaScript**

***2.1.1. Giới thiệu về JavaScript***

JavaScript là một ngôn ngữ lập trình phổ biến, được sử dụng rộng rãi để tạo ra các trang web tương tác và động. Được Brendan Eich phát triển vào năm 1995, JavaScript ban đầu được thiết kế để chạy trên trình duyệt Netscape Navigator, và hiện nay được hỗ trợ bởi tất cả các trình duyệt hiện đại.

JavaScript có thể được sử dụng để:

Thao tác với nội dung HTML và CSS để thay đổi giao diện người dùng.

Xử lý các sự kiện người dùng như nhấp chuột, nhập liệu, và di chuyển chuột.

Giao tiếp với máy chủ web thông qua các yêu cầu HTTP và xử lý dữ liệu phản hồi.

Lưu trữ và quản lý dữ liệu tạm thời trong trình duyệt bằng cách sử dụng Local Storage hoặc Session Storage.

***2.1.2. Cách dùng JavaScript***

JavaScript có thể được nhúng trực tiếp vào các trang HTML hoặc tách biệt trong các tệp tin riêng lẻ với phần mở rộng .js. Để sử dụng JavaScript trong HTML, ta có thể sử dụng thẻ <script>:

**2.2. jQuery**

***2.2.1. Giới thiệu về jQuery***

jQuery là một thư viện JavaScript nhanh và gọn nhẹ, được tạo ra bởi John Resig vào năm 2006. jQuery được thiết kế để đơn giản hóa việc xử lý các sự kiện, thao tác DOM, tạo hiệu ứng động, và giao tiếp với máy chủ.

Những ưu điểm của jQuery:

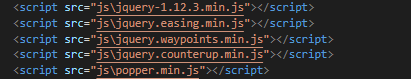
Cú pháp ngắn gọn và dễ hiểu.

Tương thích tốt với nhiều trình duyệt khác nhau.

Cung cấp nhiều tiện ích sẵn có, giúp tăng tốc độ phát triển ứng dụng web.

***2.2.2. Cách dùng jQuery***

Để sử dụng jQuery, trước hết cần thêm thư viện jQuery vào trang web của bạn. Có thể thêm trực tiếp từ CDN hoặc tải về và lưu trữ trên máy chủ của bạn. Ví dụ, thêm từ CDN:



**2.3. Botstrap**

***2.3.1. Giới thiệu về Botstrap***

Bootstrap là một framework CSS phổ biến, được phát triển bởi Mark Otto và Jacob Thornton tại Twitter vào năm 2011. Bootstrap cung cấp nhiều công cụ và thành phần giao diện sẵn có, giúp việc thiết kế trang web trở nên nhanh chóng và dễ dàng hơn.

Những tính năng nổi bật của Bootstrap:

Hệ thống lưới (grid system) giúp bố trí giao diện dễ dàng.

Các thành phần UI như nút bấm, thanh điều hướng, biểu mẫu, v.v.

Tích hợp sẵn các tiện ích JavaScript cho các tính năng như carousel, modal, tooltip.

***2.3.2. Các dùng Botstrap***

Để sử dụng Bootstrap, cần thêm các tệp CSS và JavaScript của Bootstrap vào trang web. Có thể thêm từ CDN hoặc tải về và lưu trữ trên máy chủ của bạn. Ví dụ, thêm từ CDN:

(còn )

## Kết luận chương 2

Chương 2 đã giới thiệu các cơ sở lý thuyết quan trọng về JavaScript, jQuery và Bootstrap, những công cụ chính sẽ được sử dụng trong quá trình xây dựng trang web đọc truyện. Chúng ta đã tìm hiểu về cách sử dụng JavaScript để tạo tính năng tương tác, sử dụng jQuery để đơn giản hóa các thao tác DOM và sự kiện, và sử dụng Bootstrap để thiết kế giao diện đẹp mắt và thân thiện với người dùng. Kiến thức này sẽ là nền tảng để chúng ta tiếp tục xây dựng và phát triển trang web đọc truyện trong các chương tiếp theo.

**CHƯƠNG III: XÂY DỰNG TRANG WEB PROFILE CÔNG VIỆC**

## **3.1. Ý tưởng xây dựng Website**

Ý tưởng của trang web profile công việc là tạo ra một nền tảng trực tuyến giúp người dùng dễ dàng tìm kiếm và đọc các truyện yêu thích. Website sẽ cung cấp các chức năng chính như:

**Trang chủ:** Hiển thị thông tin của profile

**Công Việc :** Hiển thị thông tin các công việc

**Xem công việc :** Hiển thị các website demo

**Chức năng:** tăng tốc độ mượt mà không bị delay

**Chức năng đăng ký và đăng nhập:** Cho phép người dùng tạo tài khoản và đăng nhập để theo dõi truyện yêu thích.

## **3.2. Xây dựng bố cục trang Web**

Phải có hình vẽ bố cục Trang chủ của Web và phân tích các thành phần trong bố cục (Header, Menu, nav, Main, Footer)

#### Header

**Mobile Header (<header class="mobile-header-1">):**

Bao gồm biểu tượng menu và logo.

Biểu tượng menu có thể được kích hoạt để mở rộng menu điều hướng.

**Desktop Header (<header class="desktop-header-1">):**

Chứa logo và menu chính.

Menu chính (<nav>) bao gồm các liên kết đến các phần khác nhau của trang web (Home, About, Services, Experience, Works, Blog, Contact).

Phần footer của header (<div class="footer">) chứa thông tin bản quyền.

#### Main Content (<main class="content">)

**Section Home (<section id="home">):**

Phần đầu tiên người dùng nhìn thấy khi truy cập vào trang.

Bao gồm hình ảnh avatar, tên của người dùng và mô tả ngắn về bản thân.

Các biểu tượng mạng xã hội cho phép người dùng kết nối với tác giả.

Nút "Thuê tôi" để kêu gọi hành động từ người dùng.

Có hiệu ứng parallax với các lớp SVG và scroll down button để hướng dẫn người dùng đi xuống phần About.

**Section About (<section id="about">):**

Giới thiệu chi tiết về người dùng.

Bao gồm hình ảnh avatar khác và mô tả chi tiết về kinh nghiệm và kỹ năng của người dùng.

Có các nút download CV cho người dùng có thể tải xuống.

Danh sách các kỹ năng và tiến độ của từng kỹ năng được biểu thị bằng thanh tiến độ.

#### Footer

**Footer:** Nằm dưới cùng của trang web.

Chứa thông tin bản quyền và tên của người dùng.

Liên kết đến các trang mạng xã hội khác của người dùng.

#### Phân tích:

**Header:** Đảm bảo sự nhất quán giữa giao diện người dùng trên nền tảng di động và máy tính để bàn. Cung cấp menu dễ sử dụng và liên kết logo trang web với trang chủ.

**Main Content:** Sử dụng parallax và hiệu ứng scroll để tăng tính tương tác và thẩm mỹ. Các phần home và about được thiết kế rõ ràng, giúp người dùng dễ dàng hiểu về người dùng và các dịch vụ mà họ cung cấp.

**Footer:** Cung cấp thông tin bổ sung và cơ hội kết nối với người dùng thông qua các nền tảng mạng xã hội.

Đây là một bố cục trang web dễ hiểu và dễ sử dụng, mang đến trải nghiệm người dùng tích cực và thú vị.

## **Thiết kế và lập trình Website**

Trình bày từng phần nhỏ theo gợi ý sau:

Thiết kế Header:



### + Phần HTML: Chức năng của Header:

**Hiển thị Logo:**

**Mục đích:** Logo của trang web thường được đặt ở góc trên cùng của header để người dùng có thể nhận biết và quay lại trang chủ dễ dàng.

**Đặc điểm:** Logo có thể là một hình ảnh hoặc biểu tượng văn bản, khi nhấp vào sẽ dẫn người dùng về trang chủ.

**Menu Điều Hướng (Navigation Menu):**

**Mục đích:** Các liên kết điều hướng chính giúp người dùng điều hướng qua các trang khác nhau của trang web.

**Đặc điểm:** Trên thiết bị di động, menu có thể được ẩn và chỉ hiển thị khi người dùng nhấp vào biểu tượng menu (thanh hamburger) ở góc trên cùng của màn hình.

**Menu Icon (Biểu tượng Menu):**

**Mục đích:** Dùng để mở và đóng menu trên thiết bị di động.

**Đặc điểm:** Thường là biểu tượng thanh hamburger (ba đường ngang) hoặc biểu tượng mũi tên xuống, khi nhấp vào sẽ mở menu điều hướng.

**Responsive Design (Thiết kế đáp ứng):**

**Mục đích:** Header cần được thiết kế để phù hợp với mọi kích thước màn hình từ thiết bị di động đến máy tính để bàn.

**Đặc điểm:** Trên thiết bị di động, header có thể có bố cục và hiển thị khác so với trên máy tính để bàn để tối ưu trải nghiệm người dùng.

**Hiệu ứng và Tương tác (jQuery và Bootstrap):**

**Mục đích:** Sử dụng jQuery và Bootstrap để thêm hiệu ứng và tương tác như hiển thị menu mở rộng, chỉnh sửa hình dạng và kích thước của các thành phần.

**Đặc điểm:** jQuery được sử dụng để bật/tắt menu trên thiết bị di động và Bootstrap giúp cấu trúc và căn chỉnh các thành phần.

**Accessibility (Tiếp cận dễ dàng):**

**Mục đích:** Header cần phải dễ dàng tiếp cận và sử dụng cho tất cả người dùng, bao gồm cả những người sử dụng trình đọc màn hình hoặc thiết bị hỗ trợ.

**Đặc điểm:** Các thành phần của header cần có thẻ HTML phù hợp và hỗ trợ các yếu tố trợ giúp như mô tả hình ảnh để đảm bảo trải nghiệm tốt cho người dùng có khuyết tật.

Tóm lại, header không chỉ là phần đầu trang web để hiển thị logo và menu điều hướng, mà còn là một phần quan trọng trong việc cung cấp trải nghiệm người dùng tốt và dễ tiếp cận. Thiết kế header cần phải được chú ý đến cả về mặt thiết kế và chức năng để đảm bảo tính thẩm mỹ và tính tương tác hiệu quả.

### + Phần jQuery: Chức năng chính của plugin counterUp:

**Định nghĩa các thiết lập mặc định:**

Plugin cho phép người dùng định nghĩa các thiết lập thông qua đối số options.

Các thiết lập mặc định bao gồm:

time: Thời gian hiệu ứng, mặc định là 400ms.

delay: Độ trễ giữa các bước số học, mặc định là 10ms.

offset: Độ lệch bắt đầu khi kích hoạt hiệu ứng, mặc định là 100% của viewport.

beginAt: Số bắt đầu hiển thị hiệu ứng, mặc định là 0.

formatter: Hàm định dạng số học (nếu có).

context: Ngữ cảnh, mặc định là cửa sổ (window).

callback: Hàm callback khi hoàn thành hiệu ứng.

**Xử lý mỗi phần tử mà plugin được áp dụng:**

Đối với mỗi phần tử được chọn, plugin xử lý các thuộc tính dữ liệu (data-counterup-\*) để lấy các giá trị đặc biệt hoặc sử dụng các thiết lập mặc định.

**Hàm counterUpper:**

Là hàm chính thực hiện hiệu ứng số học.

Tách số cần đếm thành các phần từ nhỏ hơn dựa trên thời gian và độ trễ.

Xử lý số liệu có dấu phẩy (thường xuyên sử dụng trong số lớn).

Xử lý thời gian định dạng giờ (hh:mm

).

**Hàm f:**

Là hàm callback được gọi trong setTimeout để thay đổi nội dung của phần tử đang xử lý với các giá trị số liệu đã tính toán.

Hàm này sẽ lặp lại cho đến khi tất cả các giá trị đã được hiển thị.

**Kết nối với waypoint:**

Plugin kết hợp với Waypoint để xác định khi nào phần tử vào vùng nhìn thấy (viewport).

Khi phần tử vào vùng nhìn thấy với độ lệch (offset), plugin sẽ kích hoạt hàm counterUpper và sau đó ngừng theo dõi (destroy waypoint).

**+ Botstrap:**

**Navbar (Thanh điều hướng):**

Sử dụng Navbar để hiển thị logo hoặc tên của bạn và các liên kết như Home, About, Portfolio, Contact.

Các dropdown menu để nhóm các liên kết liên quan.

**Carousel (Slideshow):**

Sử dụng Carousel để hiển thị các hình ảnh nổi bật về bạn, dự án hay sản phẩm của bạn.

Thêm các nút điều hướng trước, sau và chỉ số để người dùng có thể chuyển đổi giữa các hình ảnh.

**Cards (Thẻ):**

Dùng Cards để hiển thị thông tin về bản thân, các dự án hoặc kỹ năng chính của bạn.

Mỗi Card có thể chứa một hình ảnh, tiêu đề và mô tả ngắn.

**Button (Nút):**

Sử dụng các Button để điều hướng người dùng đến các phần quan trọng như Portfolio, Contact, hoặc để kêu gọi hành động như Đăng ký hoặc Đăng nhập.

**Scrollspy (Theo dõi cuộn trang):**

Để theo dõi vị trí của người dùng trên trang và tự động làm nổi bật các liên kết trong thanh điều hướng tương ứng với vị trí đang xem.

**Modal (Cửa sổ pop-up):**

Dùng Modal để hiển thị các thông tin chi tiết như hình ảnh lớn, mô tả chi tiết dự án hoặc form liên hệ mà người dùng có thể mở và đóng lại.

**Forms (Biểu mẫu):**

Sử dụng Forms để người dùng có thể liên hệ với bạn hoặc đăng ký tài khoản.

Thêm các trường input và button để người dùng có thể nhập thông tin và gửi đi.

**Footer (Chân trang):**

Sử dụng Footer để đặt các thông tin liên hệ như địa chỉ email, số điện thoại, địa chỉ văn phòng (nếu có), và liên kết đến mạng xã hội của bạn.

## **Kết quả xây dựng trang Web**

Giao diện của trang Web (Hình ảnh phải ghi chú và phân tích kết quả của giao diện)



### Phần Preloader

<div id="preloader">

<div class="outer">

<div class="infinityChrome">

<div></div>

<div></div>

<div></div>

</div>

<div class="infinity">

<div>

<span></span>

</div>

<div>

<span></span>

</div>

<div>

<span></span>

</div>

</div>

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" class="goo-outer">

<defs>

<filter id="goo">

<feGaussianBlur in="SourceGraphic" stdDeviation="6" result="blur"></feGaussianBlur>

<feColorMatrix in="blur" values="1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 18 -7" result="goo"></feColorMatrix>

<feBlend in="SourceGraphic" in2="goo"></feBlend>

</filter>

</defs>

</svg>

</div>

</div>

<div id="preloader">: Đây là phần tử chứa toàn bộ preloader, sẽ hiển thị khi trang web đang tải.

.outer: Lớp này bao bọc preloader và đảm bảo các phần tử bên trong được căn giữa.

.infinityChrome và .infinity: Đây là hai phần tử dùng để tạo hiệu ứng vô hạn (infinity effect), có thể làm việc với CSS hoặc JavaScript để tạo chuyển động hoặc hiệu ứng sinh động.

<svg>: Đây là một phần SVG dùng để tạo hiệu ứng mờ (blur effect) cho preloader.

### Header cho thiết bị di động (mobile header)

html

Sao chép mã

<header class="mobile-header-1">

<div class="container">

<div class="menu-icon d-inline-flex mr-4">

<button>

<span></span>

</button>

</div>

<div class="site-logo">

<a href="index.html">

<img src="images\logo.svg" alt="Nguyễn Tiến Lực">

</a>

</div>

</div>

</header>

.mobile-header-1: Đây là phần header dành cho thiết bị di động, sử dụng container để căn giữa nội dung.

.menu-icon: Đây là phần tử chứa nút menu, có thể sử dụng để mở rộng menu khi người dùng nhấp vào.

.site-logo: Phần tử này chứa logo của trang web, là liên kết trở về trang chủ khi nhấp vào.

### Header cho desktop

html

Sao chép mã

<header class="desktop-header-1 d-flex align-items-start flex-column">

<div class="site-logo">

<a href="index.html">

<img src="images\logo.svg" alt="">

</a>

</div>

</header>

.desktop-header-1: Đây là phần header dành cho thiết bị desktop, sử dụng flexbox để sắp xếp các phần tử theo chiều dọc.

.site-logo: Phần tử này chứa logo của trang web, là liên kết trở về trang chủ khi nhấp vào.

**Section Element**:

<section id="home" class="home d-flex align-items-center">: Đây là phần tử <section> có id là "home" và các lớp CSS là "home d-flex align-items-center". Các lớp này sẽ hỗ trợ việc căn chỉnh phần tử bên trong.

**Container**:

<div class="container">: Đây là một phần tử <div> dùng để giới hạn chiều rộng và căn giữa các phần tử bên trong.

**Intro Section**:

<div class="intro">: Phần này chứa thông tin chính như hình ảnh avatar, tên và mô tả ngắn về bản thân.

**Avatar Image**:

<img src="images\avatar-1.svg" alt="Bolby" class="mb-4">: Đây là hình ảnh avatar của người dùng với đường dẫn images\avatar-1.svg và alt là "Bolby". Lớp CSS mb-4 sẽ thêm khoảng cách dưới cho phần tử.

**Thông tin và mô tả**:

<h1 class="mb-2 mt-0">Nguyễn Tiến Lực</h1>: Tiêu đề chính hiển thị tên người dùng.

<span>I'm a <span class="text-rotating">Nhà thiết kế UI/UX, nhà phát triển Front-End, người yêu thích nhiếp ảnh</span></span>: Mô tả ngắn về bản thân người dùng, với một phần tử con <span> chứa lớp text-rotating để tạo hiệu ứng chữ chuyển đổi.

**Social Icons**:

<ul class="social-icons light list-inline mb-0 mt-4">: Danh sách các biểu tượng mạng xã hội.

Mỗi biểu tượng mạng xã hội được hiển thị bằng một phần tử <li> trong danh sách, mỗi phần tử có một liên kết <a> với biểu tượng từ thư viện Font Awesome (<i>).

**Buttons**:

<div class="mt-4">: Vùng chứa nút hoặc các liên kết.

<a href="https://nguyenluc11.github.io/nguyenluc11/?fbclid=IwAR2hR-BDXSIE\_s5PRj954l5WhrBoryZc1TmJZhps6hW86SdZs\_fA3JSU7GA" class="btn btn-default">Thuê tôi</a>: Nút có liên kết đến một trang web khác, với lớp btn btn-default để thiết lập kiểu dáng nút.

**Scroll Down Mouse Wheel**:

<div class="scroll-down">: Vùng chứa phần tử để dẫn dắt người dùng cuộn xuống.

<a href="#about" class="mouse-wrapper">: Liên kết dẫn đến phần tử có id là "about".

Có một biểu tượng chuột để chỉ ra hướng cuộn xuống.

**Parallax Layers**:

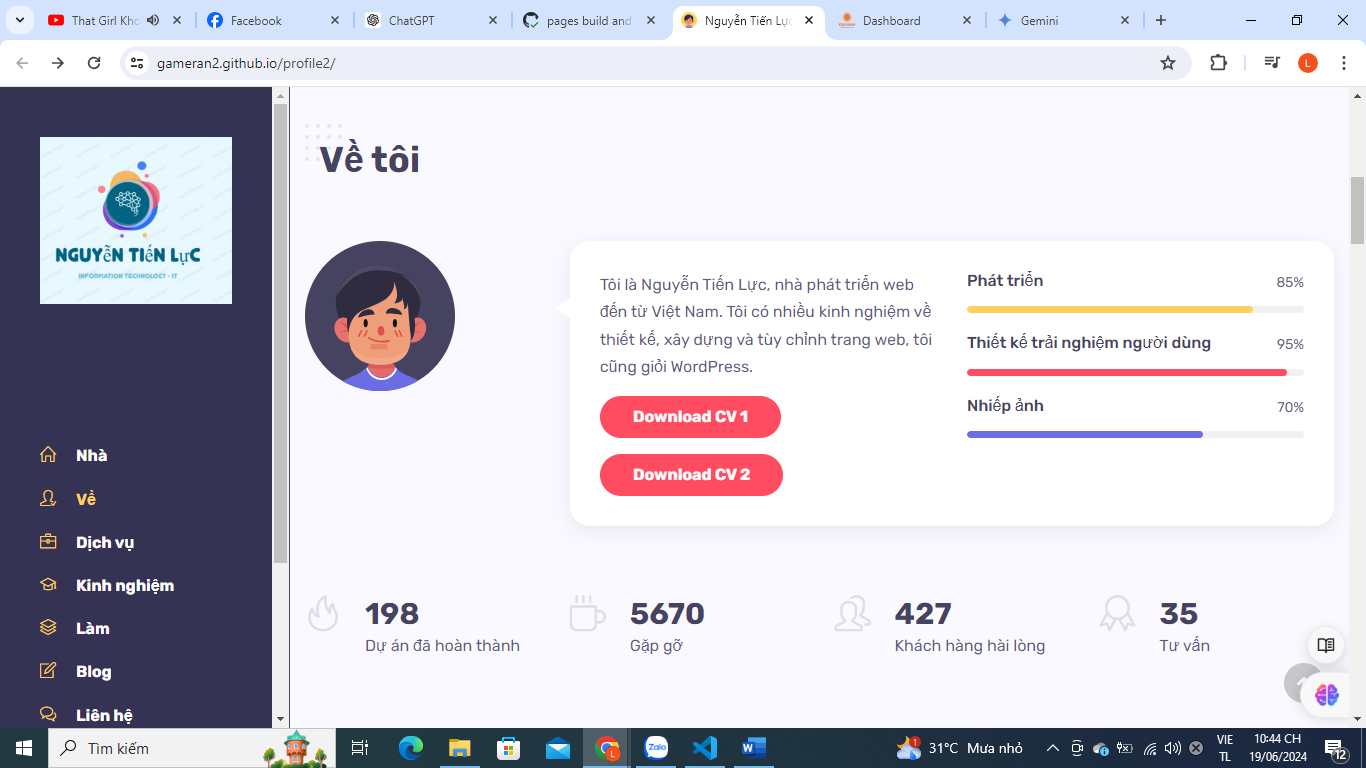
<div class="parallax" data-relative-input="true">: Vùng chứa các lớp SVG để tạo hiệu ứng Parallax.

Mỗi <svg> là một lớp với thuộc tính data-depth để điều chỉnh độ sâu của hiệu ứng Parallax.

Các <svg> này chứa các <path> và <rect> với các thuộc tính fill và stroke để tạo ra các hình ảnh và hiệu ứng khác nhau, ví dụ như dải màu, các hình vuông, hình tròn, ...

Mỗi <svg> có một lớp CSS nhất định (p1, p2, ..., p11) để kiểm soát hiệu ứng Parallax của từng lớp.

Đây là một trang web đầy màu sắc và có hiệu ứng, được thiết kế để làm nổi bật hồ sơ cá nhân của người dùng.



**Section Element**:

<section id="about">: Phần tử <section> có id là "about", dùng để phân biệt và định vị phần nội dung này trên trang web.

**Container**:

<div class="container">: Phần tử <div> với lớp CSS "container", giới hạn chiều rộng và giữa các phần tử bên trong.

**Section Title**:

<h2 class="section-title wow fadeInUp">Về tôi</h2>: Tiêu đề của phần "Về tôi", có hiệu ứng hiển thị fadeInUp từ thư viện CSS animation wow.

**Spacer**:

<div class="spacer" data-height="60"></div>: Khoảng trắng để tạo không gian giữa các phần tử, có chiều cao là 60px.

**Row**:

<div class="row">: Dòng chứa các cột trong Bootstrap Grid System để xếp lần lượt các phần tử.

**Avatar Image**:

<img src="images\avatar-2.svg" alt="Bolby">: Hình ảnh avatar của người dùng, đường dẫn images\avatar-2.svg.

**Col-md-3 và Col-md-9**:

Dòng này chia thành 2 cột trong Bootstrap:

**Col-md-3**: Cột chiếm 3/12 (25%) chiều rộng trên màn hình lớn (md), chứa hình ảnh avatar và một khoảng trắng dành cho màn hình nhỏ (d-md-none d-lg-none).

**Col-md-9**: Cột chiếm 9/12 (75%) chiều rộng trên màn hình lớn (md), có các hiệu ứng hình tam giác (triangle-left-md triangle-top-sm).

**Rounded, bg-white, shadow-dark, padding-30**:

Các lớp CSS đi kèm để làm tròn góc (rounded), nền trắng (bg-white), đổ bóng (shadow-dark), và thêm padding 30px (padding-30).

**Row và Col-md-6 trong Col-md-9**:

Cột này chia thành 2 cột trong Bootstrap Grid System:

**Col-md-6**: Cột chiếm 6/12 (50%) chiều rộng trên màn hình lớn (md), chứa văn bản giới thiệu về người dùng và nút tải CV.

**Paragraph (<p>)**: Mô tả về bản thân người dùng.

**Buttons (<a>)**: Liên kết tải CV với lớp btn btn-default.

**Col-md-6**: Cột chiếm 6/12 (50%) chiều rộng trên màn hình lớn (md), chứa các phần tử kỹ năng.

**Skill Item và Progress Bar**:

Mỗi kỹ năng được chia thành một phần tử skill-item.

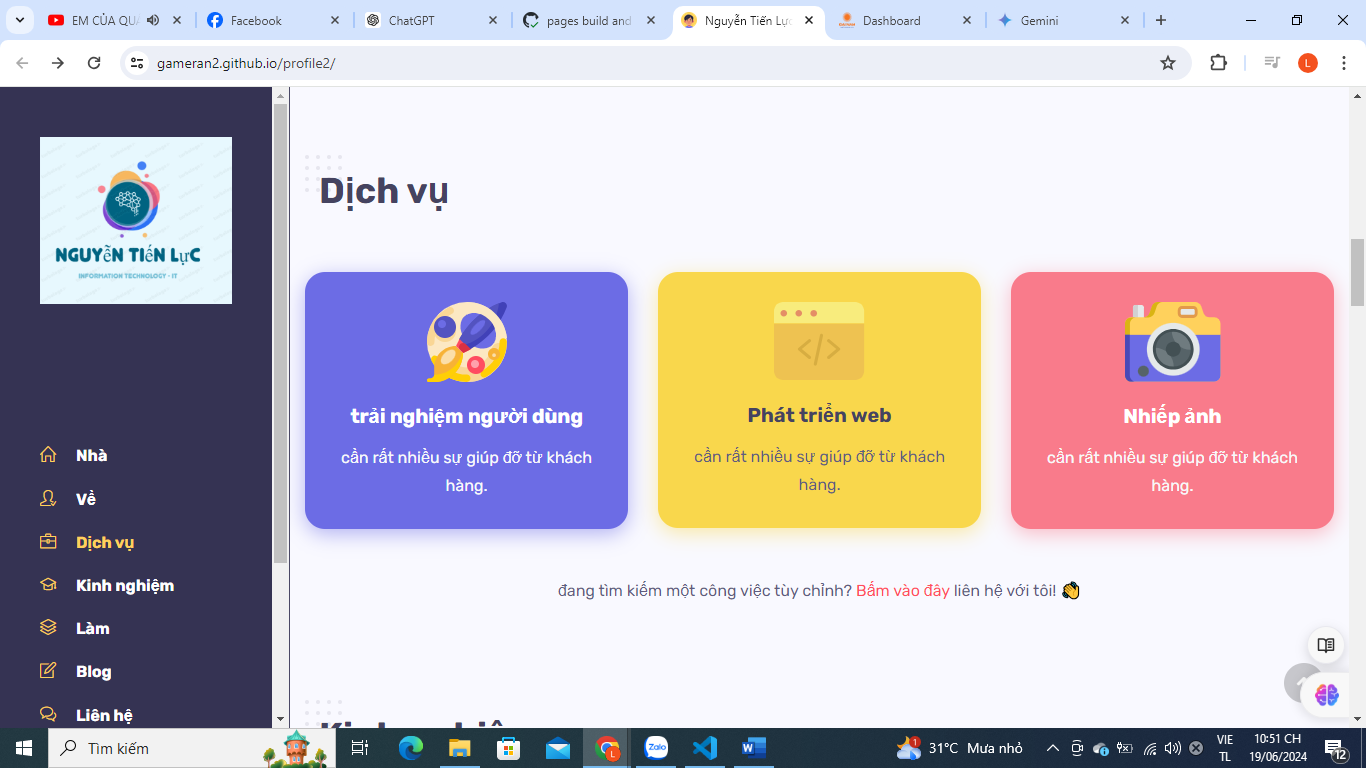
**Skill Info**: Chứa tiêu đề kỹ năng và phần trăm hiện tại.

**Progress Bar**: Thanh tiến độ để hiển thị phần trăm kỹ năng.

**Fact Item**:Các phần tử thông tin số liệu như số dự án hoàn thành, số lần gặp gỡ, số khách hàng hài lòng, và số lần tư vấn.

**Icons**: Mỗi phần tử fact-item có một biểu tượng (icon icon-fire, icon icon-cup, icon icon-people, icon icon-badge) và chi tiết số liệu.

**Spacer**: <div class="spacer" data-height="70"></div>: Khoảng trắng cuối cùng của phần tử, có chiều cao là 70px.



**Section Element**:

<section id="services">: Phần tử <section> có id là "services", dùng để phân biệt và định vị phần nội dung này trên trang web.

**Container**:

<div class="container">: Phần tử <div> với lớp CSS "container", giới hạn chiều rộng và giữa các phần tử bên trong.

**Section Title**:

<h2 class="section-title wow fadeInUp">Dịch vụ</h2>: Tiêu đề của phần "Dịch vụ", có hiệu ứng hiển thị fadeInUp từ thư viện CSS animation wow.

**Spacer**:

<div class="spacer" data-height="60"></div>: Khoảng trắng để tạo không gian giữa các phần tử, có chiều cao là 60px.

**Row và Col-md-4**:

Dòng này chia thành 3 cột trong Bootstrap Grid System:

**Col-md-4**: Mỗi cột chiếm 4/12 (33.33%) chiều rộng trên màn hình lớn (md).

Mỗi cột chứa một hộp dịch vụ (service-box).

**Service Box**:<div class="service-box rounded data-background padding-30 text-center text-light shadow-blue" data-color="#6C6CE5">: Hộp dịch vụ với các lớp CSS:

rounded: Làm tròn góc.

data-background: Sử dụng màu nền từ thuộc tính data-color.

padding-30: Padding 30px.

text-center: Căn giữa nội dung.

text-light: Màu chữ sáng.

shadow-blue: Đổ bóng màu xanh lam.

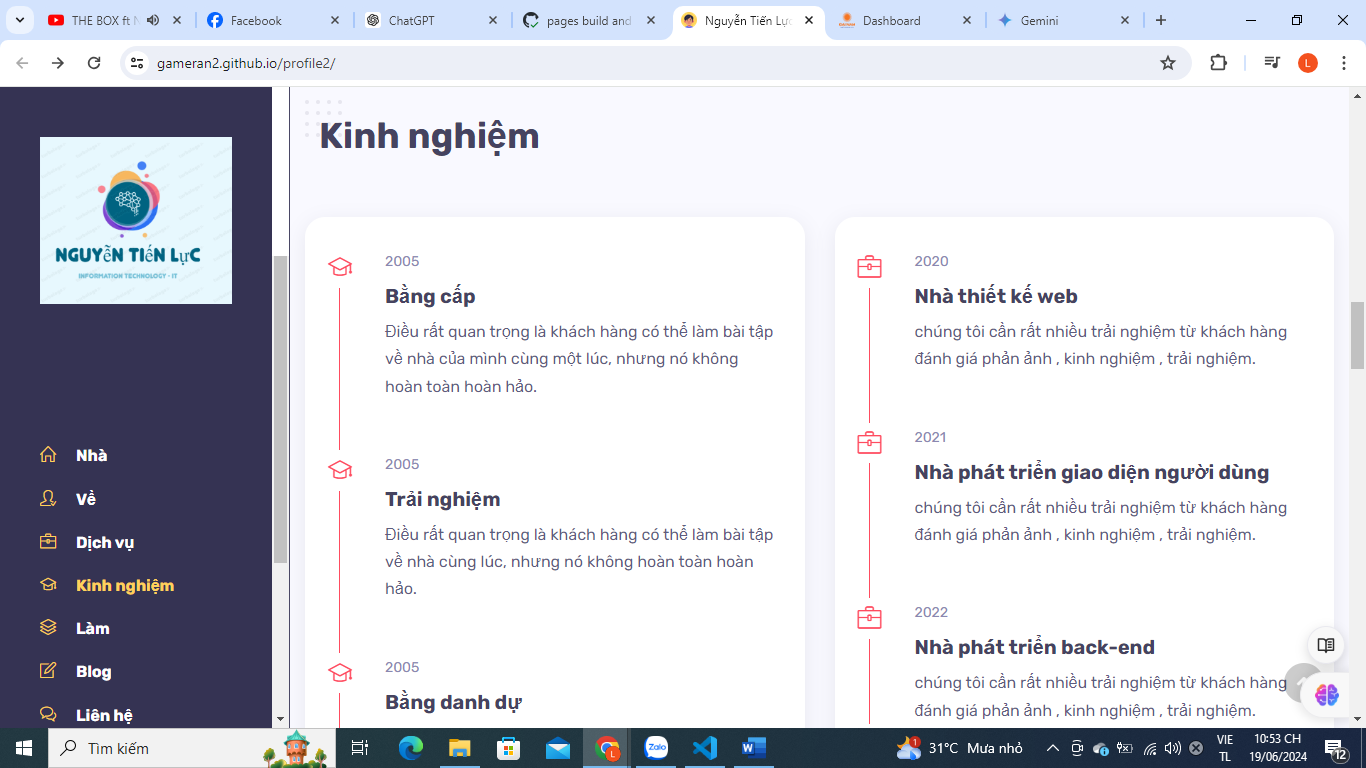
**Hình ảnh (<img>)**: Biểu tượng dịch vụ.

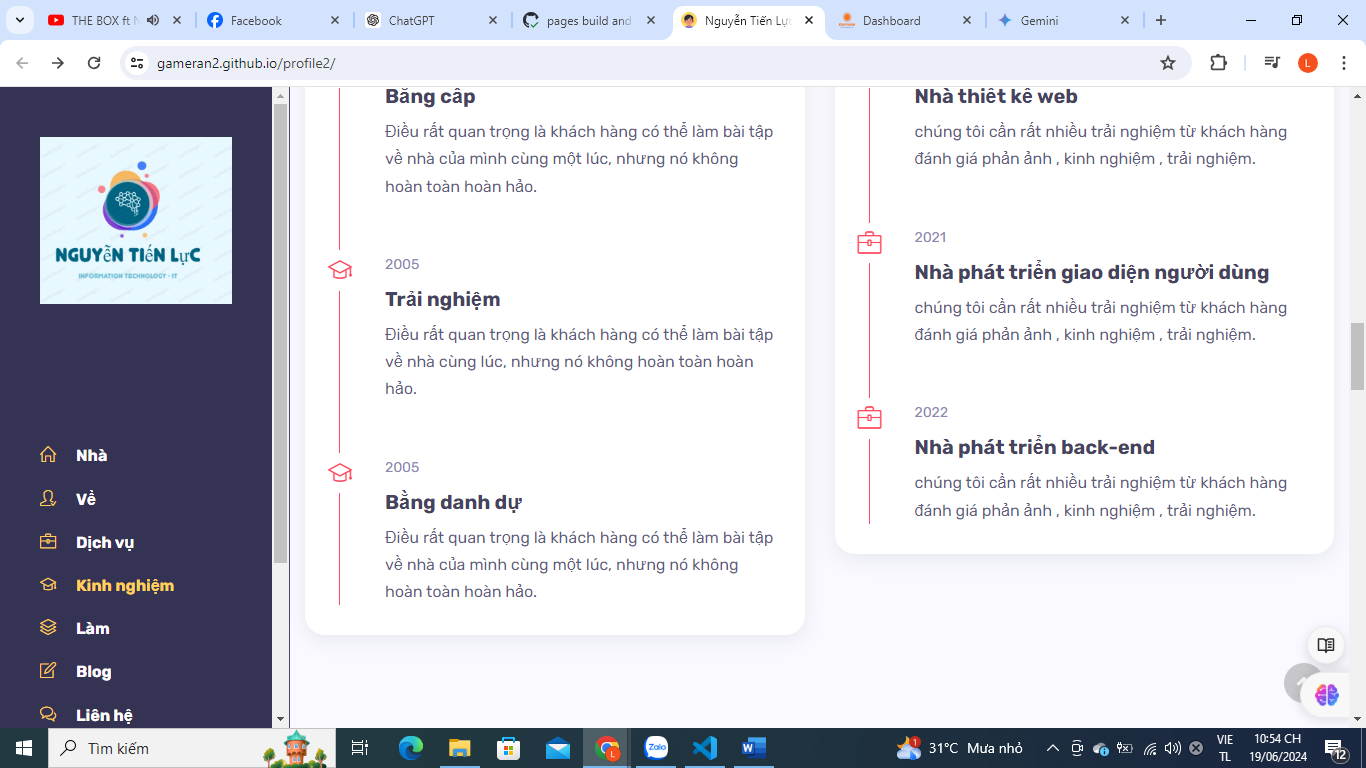
**Tiêu đề (<h3>)** và **Nội dung (<p>)**: Mô tả về dịch vụ cung cấp.

**Link liên hệ**:

<div class="mt-5 text-center">: Dòng văn bản ở cuối phần "Dịch vụ".

<p class="mb-0">đang tìm kiếm một công việc tùy chỉnh? <a href="#contact">Bấm vào đây</a> liên hệ với tôi! 👋</p>: Đoạn văn bản có chứa liên kết "Bấm vào đây" để chuyển đến phần liên hệ (#contact).





**Section Element**:<section id="experience">: Phần tử <section> có id là "experience", dùng để định vị phần nội dung này trên trang web.

**Container**:<div class="container">: Phần tử <div> với lớp CSS "container", giới hạn chiều rộng và căn giữa các phần tử bên trong.

**Section Title**:<h2 class="section-title wow fadeInUp">Kinh nghiệm</h2>: Tiêu đề của phần "Kinh nghiệm", có hiệu ứng hiển thị fadeInUp từ thư viện CSS animation wow.

**Spacer**:<div class="spacer" data-height="60"></div>: Khoảng trắng để tạo không gian giữa các phần tử, có chiều cao là 60px.

**Row và Col-md-6**:Dòng này chia thành 2 cột trong Bootstrap Grid System:

**Col-md-6**: Mỗi cột chiếm 6/12 (50%) chiều rộng trên màn hình lớn (md).

**Timeline**:

**Timeline Wrapper**:<div class="timeline edu bg-white rounded shadow-dark padding-30 overflow-hidden"> và <div class="timeline exp bg-white rounded shadow-dark padding-30 overflow-hidden">: Hai hộp đựng timeline, mỗi hộp có các lớp CSS để tạo nền, đổ bóng và padding.

**Timeline Item**:Mỗi <div class="timeline-container wow fadeInUp"> là một mốc thời gian trong timeline.

**Data-wow-delay**: Thuộc tính để xác định thời gian trễ hiệu ứng wow.

**Content of Timeline Item**:

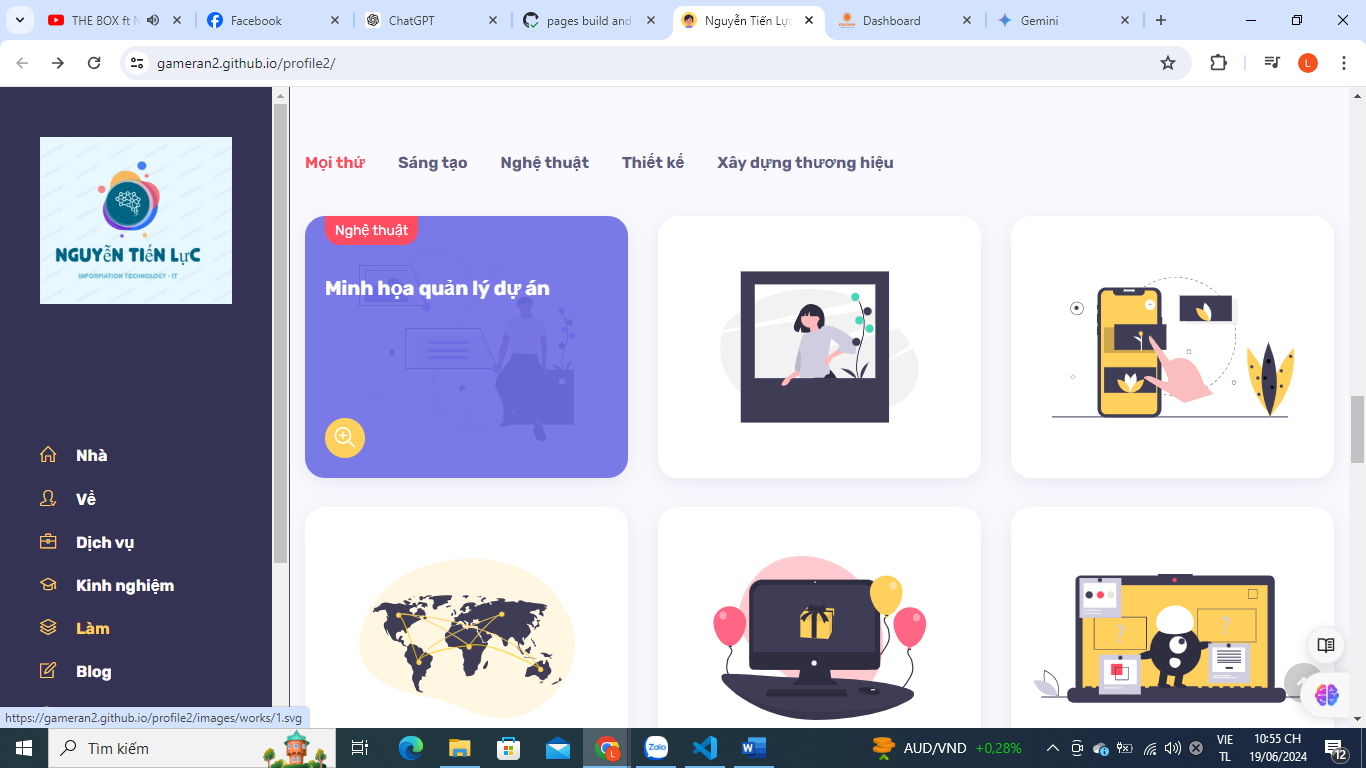
**Thời gian (<span class="time">)**: Năm hoặc thời điểm quan trọng.

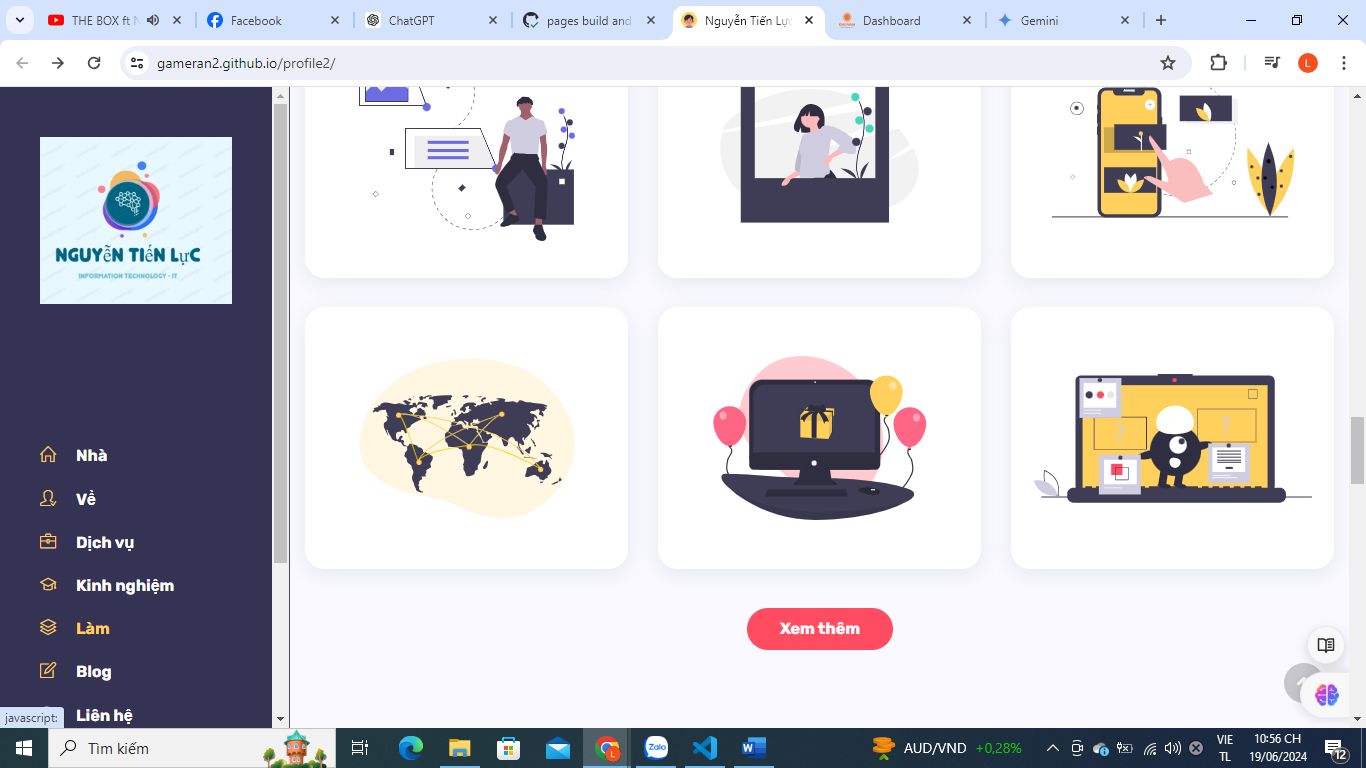
**Tiêu đề (<h3 class="title">)**: Mô tả sự kiện quan trọng.

**Nội dung (<p>)**: Mô tả chi tiết về sự kiện.

**Main Line (<span class="line"></span>)**: Đường chính của timeline.

**Responsive Spacer**:<div class="spacer d-md-none d-lg-none" data-height="30"></div>: Khoảng trắng chỉ hiển thị trên các thiết bị nhỏ hơn md.





**Section Element**:<section id="works">: Phần tử <section> có id là "works", dùng để định vị phần nội dung này trên trang web.

**Container**:<div class="container">: Phần tử <div> với lớp CSS "container", giới hạn chiều rộng và căn giữa các phần tử bên trong.

**Section Title**:<h2 class="section-title wow fadeInUp">Những công việc mới đây</h2>: Tiêu đề của phần "Những công việc mới đây", có hiệu ứng hiển thị fadeInUp từ thư viện CSS animation wow.

**Spacer**:<div class="spacer" data-height="60"></div>: Khoảng trắng để tạo không gian giữa các phần tử, có chiều cao là 60px.

**Portfolio Filter (Desktop)**:

<ul class="portfolio-filter list-inline wow fadeInUp">: Danh sách lọc các dự án, sử dụng <li> để liệt kê từng danh mục.

Mỗi <li> có thuộc tính data-filter để xác định danh mục tương ứng.

**Portfolio Filter (Mobile)**:

<select class="portfolio-filter-mobile">: Dropdown chọn danh mục dự án trên thiết bị di động, với các <option> tương ứng với danh mục.

**Portfolio Wrapper**:

<div class="row portfolio-wrapper">: Dòng chứa các mục dự án, sử dụng lưới (grid) Bootstrap.

**Portfolio Item**:

<div class="col-md-4 col-sm-6 grid-item art">, <div class="col-md-4 col-sm-6 grid-item creative design">, vv.: Mỗi mục dự án được đặt trong một cột Bootstrap có kích thước phù hợp.

class="grid-item": Chỉ định các phần tử có thể lọc và sắp xếp.

class="art", class="creative", class="design", vv.: Các lớp này xác định danh mục của từng dự án để có thể lọc.

**Portfolio Item Content**:

Mỗi mục dự án (<div class="portfolio-item rounded shadow-dark">) bao gồm:

**Details (<div class="details">)**: Hiển thị thông tin chi tiết dự án.

**Thumbnail (<div class="thumb">)**: Hình ảnh đại diện cho dự án.

**Link hoặc nội dung chi tiết dự án (<a>)**: Dẫn đến trang chi tiết hoặc popup với thông tin bổ sung về dự án.

**Popup Content**:

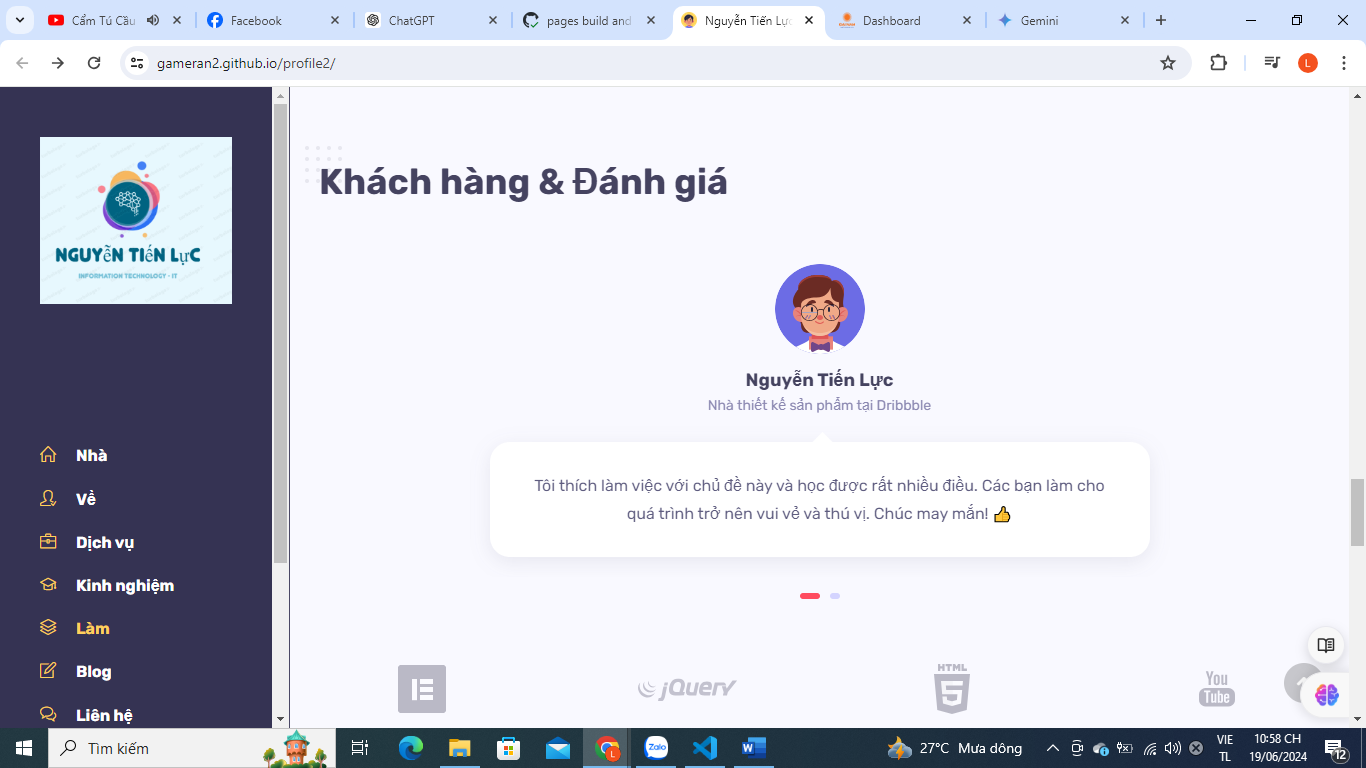
<div id="small-dialog" class="white-popup zoom-anim-dialog mfp-hide">: Popup hiển thị khi người dùng nhấp vào một dự án có chứa class="work-content" hoặc tương tự.

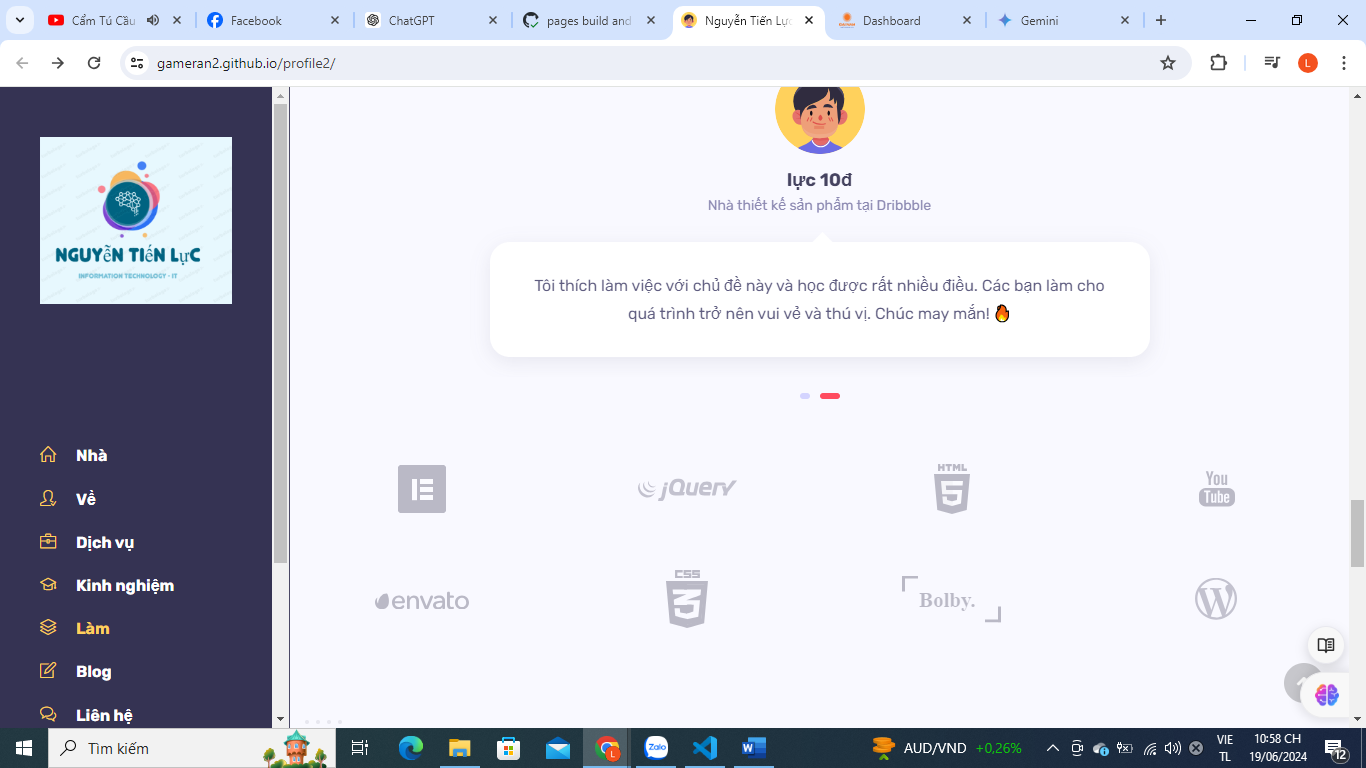
<div id="gallery-1" class="gallery mfp-hide">: Popup gallery cho các dự án hình ảnh, có thể chứa nhiều ảnh.

**Load More Button**:

<div class="load-more text-center mt-4">: Nút "Xem thêm" để tải thêm dự án.

<ul class="portfolio-pagination list-inline d-none">: Phân trang dự án (hiện tại là ẩn vì chỉ có một trang).





**Section Element**:

<section id="testimonials">: Phần tử <section> có id là "testimonials", dùng để định vị phần nội dung này trên trang web.

**Container**:

<div class="container">: Phần tử <div> với lớp CSS "container", giới hạn chiều rộng và căn giữa các phần tử bên trong.

**Section Title**:

<h2 class="section-title wow fadeInUp">Khách hàng & Đánh giá</h2>: Tiêu đề của phần "Khách hàng & Đánh giá", có hiệu ứng hiển thị fadeInUp từ thư viện CSS animation wow.

**Spacer**:

<div class="spacer" data-height="60"></div>: Khoảng trắng để tạo không gian giữa các phần tử, có chiều cao là 60px.

**Testimonials Wrapper**:

<div class="testimonials-wrapper">: Dòng chứa các mục đánh giá từ khách hàng.

**Testimonial Item**:

<div class="testimonial-item text-center mx-auto">: Mỗi mục đánh giá từ khách hàng.

**Avatar (<img>)**: Hình ảnh đại diện của khách hàng.

**Tên khách hàng (<h4>)**: Tên của khách hàng.

**Chức vụ (<span class="subtitle">)**: Chức vụ của khách hàng hoặc thông tin khác.

**Nội dung đánh giá (<div class="bg-white padding-30 shadow-dark rounded triangle-top position-relative mt-4">)**: Phần nội dung đánh giá từ khách hàng.

**Client Items**:

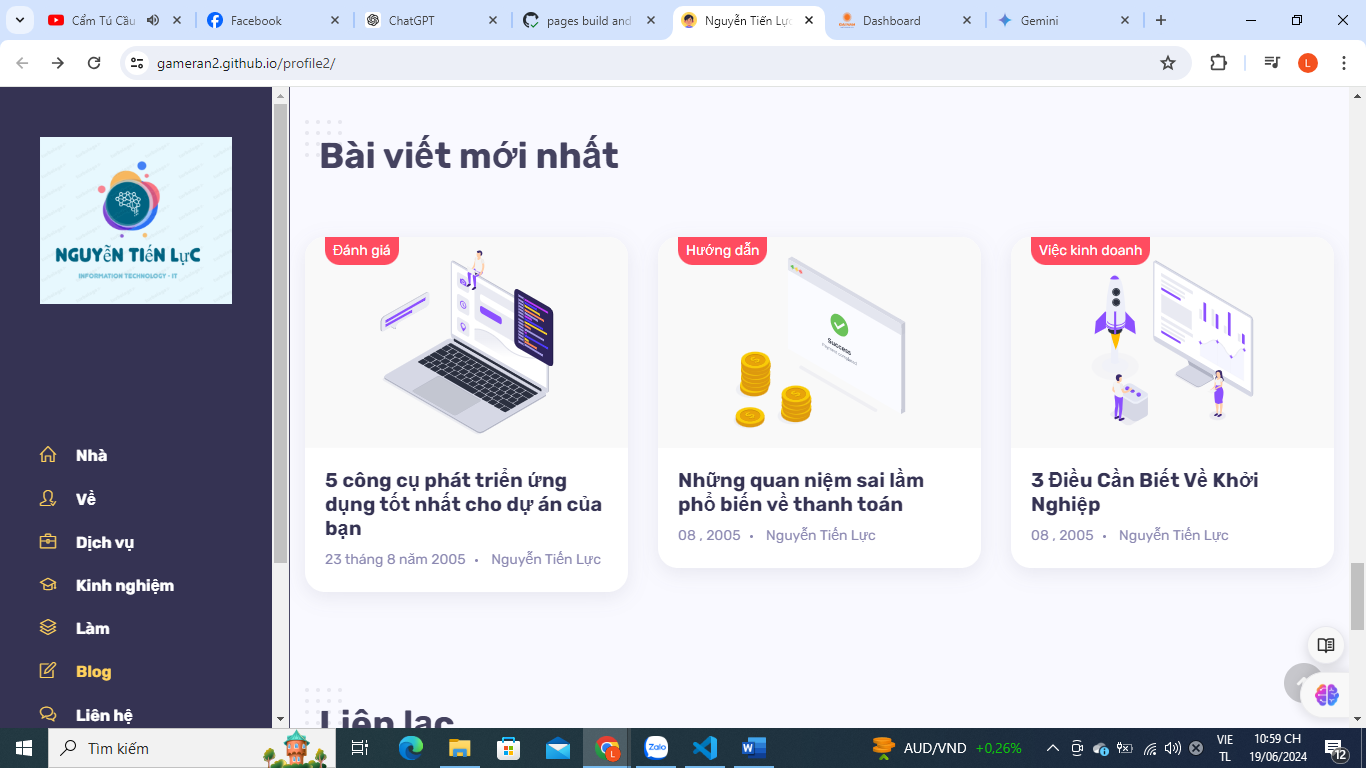
<div class="row">: Dòng chứa các logo đại diện cho khách hàng đã từng hợp tác hoặc được đề cập.

Mỗi <div class="col-md-3 col-6"> chứa một logo khách hàng trong lưới (grid) Bootstrap.

**Client Item**:

<div class="client-item">: Mỗi mục đại diện cho một khách hàng.

**Logo khách hàng (<img>)**: Hình ảnh đại diện của khách hàng.



**Section Element**:

<section id="blog">: Phần tử <section> có id là "blog", dùng để định vị phần nội dung này trên trang web.

**Container**:

<div class="container">: Phần tử <div> với lớp CSS "container", giới hạn chiều rộng và căn giữa các phần tử bên trong.

**Section Title**:

<h2 class="section-title wow fadeInUp">Bài viết mới nhất</h2>: Tiêu đề của phần "Bài viết mới nhất", có hiệu ứng hiển thị fadeInUp từ thư viện CSS animation wow.

**Spacer**:

<div class="spacer" data-height="60"></div>: Khoảng trắng để tạo không gian giữa các phần tử, có chiều cao là 60px.

**Blog Wrapper**:

<div class="row blog-wrapper">: Dòng chứa các mục bài viết blog.

**Blog Item**:

<div class="col-md-4">: Mỗi mục bài viết chiếm 1/3 (4/12) phần của cột (Bootstrap grid system).

<div class="blog-item rounded bg-white shadow-dark wow fadeIn">: Mỗi mục bài viết blog có lớp CSS để làm tròn viền, nền trắng với bóng đổ và hiệu ứng fadeIn.

**Thumb (<div class="thumb">)**: Phần thumbnail của bài viết.

<span class="category">: Thẻ chứa loại bài viết (ví dụ: Đánh giá, Hướng dẫn, Việc kinh doanh).

<img src="images\blog\1.svg" alt="blog-title">: Hình ảnh đại diện của bài viết.

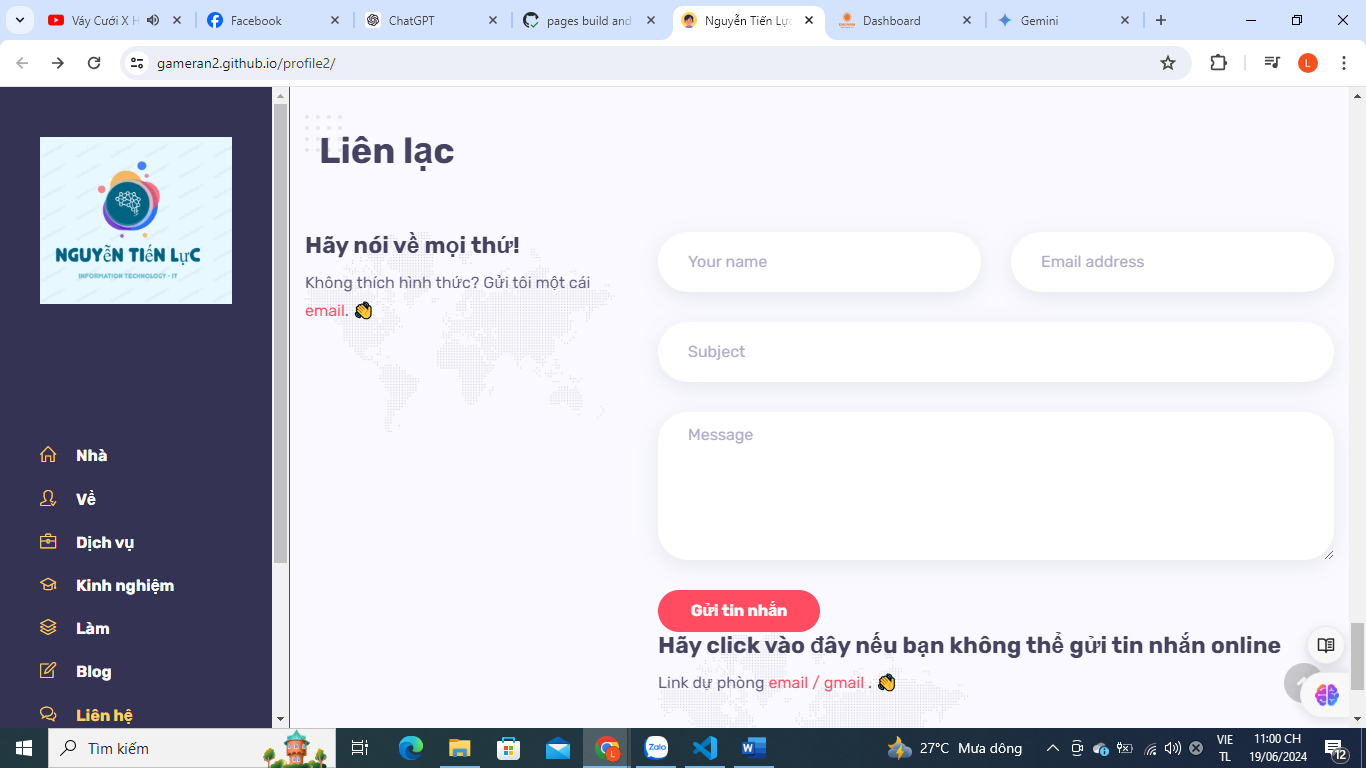
**Details (<div class="details">)**: Phần chi tiết của bài viết.

<h4 class="my-0 title"><a href="#">5 công cụ phát triển ứng dụng tốt nhất cho dự án của bạn</a></h4>: Tiêu đề bài viết và liên kết đến bài viết.

<ul class="list-inline meta mb-0 mt-2">: Danh sách ngang để hiển thị thông tin meta của bài viết.

<li class="list-inline-item">23 tháng 8 năm 2005</li>: Ngày đăng bài viết.

<li class="list-inline-item">Nguyễn Tiến Lực</li>: Tác giả của bài viết.



**Phần Liên lạc (<section id="contact">)**:

<section id="contact">

<div class="container">

<!-- Tiêu đề phần -->

<h2 class="section-title wow fadeInUp">Liên lạc</h2>

<!-- Khoảng trống để tạo khoảng cách giữa tiêu đề và nội dung -->

<div class="spacer" data-height="60"></div>

<div class="row">

<div class="col-md-4">

<!-- Thông tin liên hệ -->

<div class="contact-info">

<h3 class="wow fadeInUp">Hãy nói về mọi thứ!</h3>

<p class="wow fadeInUp">Không thích hình thức? Gửi tôi một cái <a href="mailto:lucdubai2380@gmail.com">email</a>. 👋</p>

</div>

</div>

<div class="col-md-8">

<!-- Biểu mẫu liên hệ -->

<form id="contact-form" class="contact-form mt-6" method="post" action="form/contact.php">

<div class="messages"></div>

<div class="row">

<div class="column col-md-6">

<!-- Ô nhập tên -->

<div class="form-group">

<input type="text" class="form-control" name="InputName" id="InputName" placeholder="Your name" required="required" data-error="Name is required.">

<div class="help-block with-errors"></div>

</div>

</div>

<div class="column col-md-6">

<!-- Ô nhập email -->

<div class="form-group">

<input type="email" class="form-control" id="InputEmail" name="InputEmail" placeholder="Email address" required="required" data-error="Email is required.">

<div class="help-block with-errors"></div>

</div>

</div>

<div class="column col-md-12">

<!-- Ô nhập chủ đề -->

<div class="form-group">

<input type="text" class="form-control" id="InputSubject" name="InputSubject" placeholder="Subject" required="required" data-error="Subject is required.">

<div class="help-block with-errors"></div>

</div>

</div>

<div class="column col-md-12">

<!-- Ô nhập nội dung tin nhắn -->

<div class="form-group">

<textarea name="InputMessage" id="InputMessage" class="form-control" rows="5" placeholder="Message" required="required" data-error="Message is required."></textarea>

<div class="help-block with-errors"></div>

</div>

</div>

</div>

<!-- Nút gửi tin nhắn -->

<button type="submit" name="submit" id="submit" value="Submit" class="btn btn-default">Gửi tin nhắn</button>

<!-- Thông tin liên hệ dự phòng -->

<div class="contact-info">

<h3 class="wow fadeInUp">Hãy click vào đây nếu bạn không thể gửi tin nhắn online</h3>

<p class="wow fadeInUp">Link dự phòng <a href="mailto:lucdubai2380@gmail.com"> email / gmail </a>. 👋</p>

</div>

</form>

</div>

</div>

</div>

</section>

**.container**: Tạo một khu vực giới hạn chiều rộng nội dung.

**<h2 class="section-title wow fadeInUp">Liên lạc</h2>**: Tiêu đề của phần liên lạc.

**.spacer**: Tạo một khoảng trống giữa tiêu đề và nội dung bên dưới.

**.row và .col-md-\***: Sử dụng lưới Bootstrap để chia cột, với thông tin liên hệ chiếm 4 cột và biểu mẫu liên hệ chiếm 8 cột trong lớp lưới động (col-md-\*).

**Thông tin liên hệ (<div class="contact-info">)**:

Hiển thị thông tin về liên hệ và hướng dẫn người dùng.

Liên kết đến địa chỉ email lucdubai2380@gmail.com cho phép người dùng gửi email khi họ không thích sử dụng biểu mẫu.

**Biểu mẫu liên hệ (<form id="contact-form">)**:

**<form>**: Biểu mẫu được sử dụng để người dùng nhập thông tin và gửi tin nhắn.

**<input> và <textarea>**: Các phần nhập dữ liệu cho tên, email, chủ đề, và nội dung tin nhắn.

**<button>**: Nút "Gửi tin nhắn" để người dùng có thể gửi thông tin đã nhập.

**<div class="messages"></div>**: Khu vực để hiển thị các thông báo và phản hồi từ máy chủ khi gửi biểu mẫu.

**Thông tin liên hệ dự phòng**:

Nếu người dùng không thể gửi tin nhắn trực tuyến, họ có thể sử dụng link email dự phòng để liên hệ.

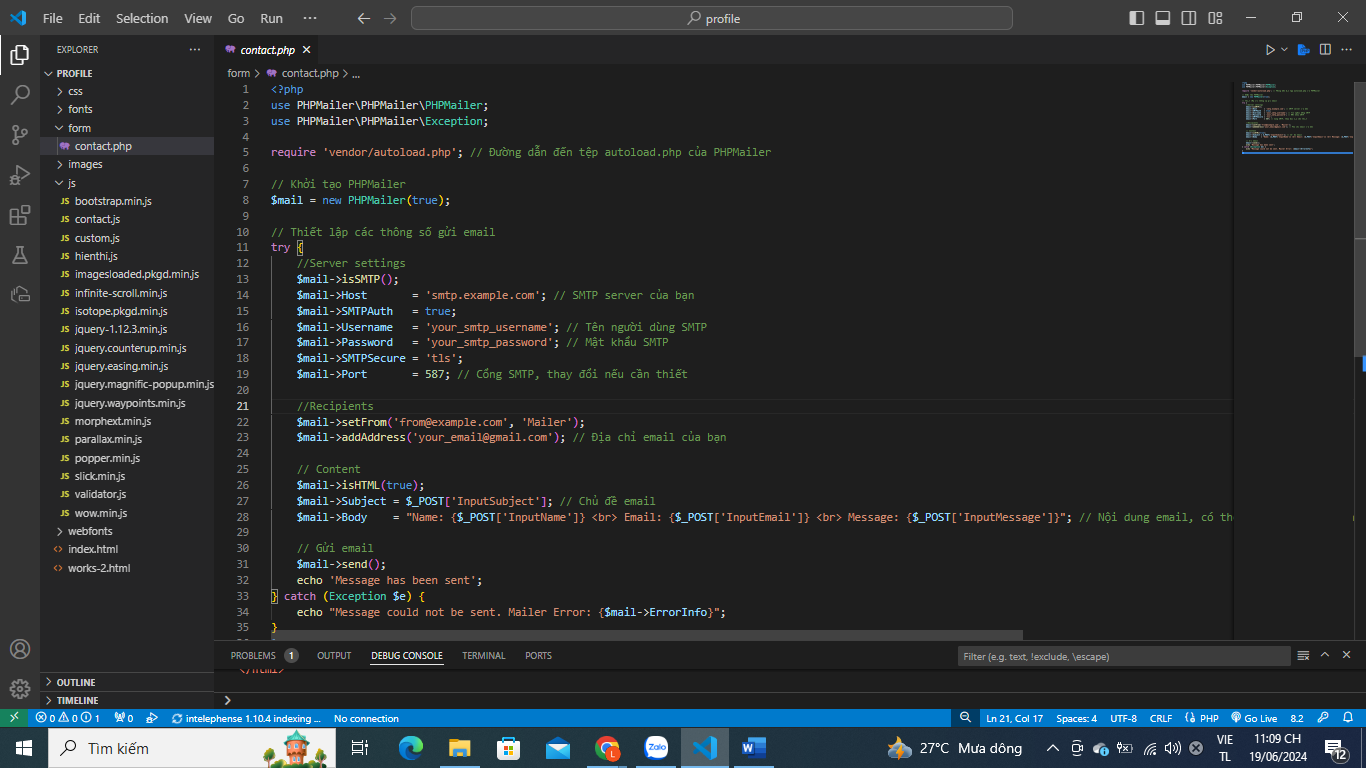
**Khoảng trống (<div class="spacer" data-height="96"></div>)**:

Tạo một khoảng trống để giữa các phần trên trang web.

**Kết thúc của trang web**:

Kết thúc <body> và <html> tags.

Liên kết các tệp JavaScript để cung cấp các tính năng tương tác như hiệu ứng scroll, xử lý biểu mẫu, và các thư viện hỗ trợ khác.



**Khởi tạo và thiết lập PHPMailer**:

<?php

use PHPMailer\PHPMailer\PHPMailer;

use PHPMailer\PHPMailer\Exception;

require 'vendor/autoload.php'; // Đường dẫn đến tệp autoload.php của PHPMailer

// Khởi tạo PHPMailer

$mail = new PHPMailer(true);

Đoạn mã đầu tiên sử dụng use để import các lớp từ thư viện PHPMailer.

require 'vendor/autoload.php'; là lệnh để tải các lớp PHPMailer từ tệp autoload.php. Điều này giúp tự động tải các lớp cần thiết mà không cần phải include hay require từng tệp riêng lẻ.

**Thiết lập thông số gửi email**:

php

Sao chép mã

try {

// Thiết lập các thông số gửi email

$mail->isSMTP();

$mail->Host = 'smtp.example.com'; // SMTP server của bạn

$mail->SMTPAuth = true;

$mail->Username = 'your\_smtp\_username'; // Tên người dùng SMTP

$mail->Password = 'your\_smtp\_password'; // Mật khẩu SMTP

$mail->SMTPSecure = 'tls';

$mail->Port = 587; // Cổng SMTP, thay đổi nếu cần thiết

isSMTP() khai báo rằng bạn đang sử dụng SMTP để gửi email.

Host, SMTPAuth, Username, Password, SMTPSecure, Port là các thông số cấu hình cho máy chủ SMTP của bạn. Bạn cần thay đổi các giá trị này để phù hợp với máy chủ SMTP mà bạn đang sử dụng.

**Thiết lập người gửi và người nhận**:

php

Sao chép mã

//Recipients

$mail->setFrom('from@example.com', 'Mailer');

$mail->addAddress('your\_email@gmail.com'); // Địa chỉ email của bạn

setFrom() để đặt địa chỉ email người gửi và tên người gửi.

addAddress() để thêm địa chỉ email người nhận.

**Nội dung và gửi email**:

php

Sao chép mã

// Content

$mail->isHTML(true);

$mail->Subject = $\_POST['InputSubject']; // Chủ đề email, lấy từ dữ liệu POST của biểu mẫu

$mail->Body = "Name: {$\_POST['InputName']} <br> Email: {$\_POST['InputEmail']} <br> Message: {$\_POST['InputMessage']}"; // Nội dung email, có thể được tạo dựa trên dữ liệu từ biểu mẫu

// Gửi email

$mail->send();

echo 'Message has been sent';

isHTML(true) cho biết nội dung email có dạng HTML.

$\_POST['InputSubject'], $\_POST['InputName'], $\_POST['InputEmail'], $\_POST['InputMessage'] là dữ liệu được gửi từ biểu mẫu HTML, sử dụng để tạo tiêu đề và nội dung email.

send() là phương thức để gửi email. Nếu thành công, in ra thông báo "Message has been sent".

**Xử lý lỗi**:

php

Sao chép mã

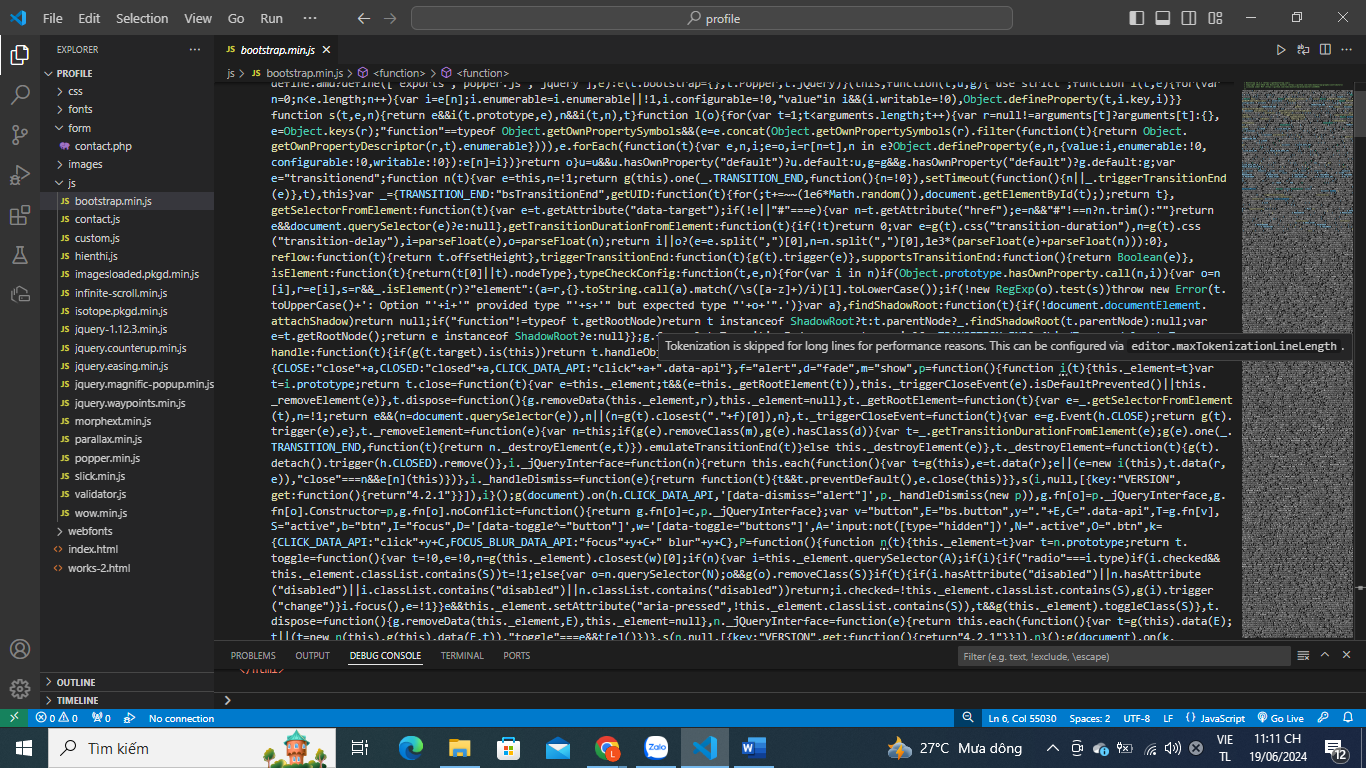
} catch (Exception $e) {

echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";

}

Đoạn mã trong khối try được bao bọc trong try-catch để bắt lỗi nếu có bất kỳ vấn đề gì xảy ra trong quá trình gửi email.

Nếu có lỗi, thông báo lỗi được hiển thị ra màn hình.



### ****Các hàm tiện ích (****s ****và**** l****)****

#### Hàm s(t, e, n)

javascript

Sao chép mã

function s(t, e, n) {

return e && i(t.prototype, e), n && i(t, n), t

}

**Mục đích**: Đây là một hàm tiện ích dùng để mở rộng các đối tượng. Hàm này kết hợp các thuộc tính từ e (các thuộc tính nguyên mẫu) và n (các thuộc tính tĩnh) vào đối tượng t.

#### Hàm l(o)

javascript

function l(o) {

for (var t = 1; t < arguments.length; t++) {

var r = null != arguments[t] ? arguments[t] : {};

var e = Object.keys(r);

if (typeof Object.getOwnPropertySymbols === 'function') {

e = e.concat(Object.getOwnPropertySymbols(r).filter(function(t) {

return Object.getOwnPropertyDescriptor(r, t).enumerable;

}));

}

e.forEach(function(t) {

var e, n, i;

e = o;

i = r[n = t];

if (n in e) {

Object.defineProperty(e, n, { value: i, enumerable: !0, configurable: !0, writable: !0 });

} else {

e[n] = i;

}

});

}

return o;

}

**Mục đích**: Hàm này được sử dụng để sao chép các thuộc tính từ các đối số vào đối tượng o. Nó cho phép bạn sao chép các thuộc tính enumerable và non-enumerable, bao gồm cả các symbol properties nếu có.

### ****Các hằng số và biến****

#### Biến và hằng số

u và g: Đây là biến đại diện cho thư viện jQuery (hoặc một thư viện tương tự). Nếu u và g tồn tại và có thuộc tính default, thì sẽ sử dụng giá trị mặc định của chúng.

e = "transitionend": Hằng số này định nghĩa tên sự kiện kết thúc transition CSS, được sử dụng để xác định khi transition hoàn thành.

#### Đối tượng \_

javascript

Sao chép mã

var \_ = {

TRANSITION\_END: "bsTransitionEnd",

getUID: function(t) { /\* ... \*/ },

getSelectorFromElement: function(t) { /\* ... \*/ },

getTransitionDurationFromElement: function(t) { /\* ... \*/ },

reflow: function(t) { /\* ... \*/ },

triggerTransitionEnd: function(t) { /\* ... \*/ },

supportsTransitionEnd: function() { /\* ... \*/ },

isElement: function(t) { /\* ... \*/ },

typeCheckConfig: function(t, e, n) { /\* ... \*/ },

findShadowRoot: function(t) { /\* ... \*/ }

};

**Mục đích**: Đây là một đối tượng chứa các phương thức hỗ trợ và các hằng số liên quan đến transition và DOM manipulation.

### ****Các thành phần Bootstrap****

#### Alert Component

javascript

Sao chép mã

var o = "alert",

r = "bs.alert",

a = "." + r,

c = g.fn[o],

h = {

CLOSE: "close" + a,

CLOSED: "closed" + a,

CLICK\_DATA\_API: "click" + a + ".data-api"

};

**Mục đích**: Định nghĩa các hằng số và phương thức cho thành phần Alert trong Bootstrap. Bao gồm các sự kiện như close, closed và click.

#### Button Component

javascript

Sao chép mã

var v = "button",

E = "bs.button",

y = "." + E,

C = ".data-api",

T = g.fn[v],

S = "active",

b = "btn",

I = "focus",

D = '[data-toggle^="button"]',

w = '[data-toggle="buttons"]',

A = 'input:not([type="hidden"])',

N = ".active",

= ".btn",

k = {

CLICK\_DATA\_API: "click" + y + C,

FOCUS\_BLUR\_DATA\_API: "focus" + y + C + " blur" + y + C

};

**Mục đích**: Định nghĩa các hằng số và phương thức cho thành phần Button trong Bootstrap. Bao gồm các sự kiện như click, focus, blur và các lựa chọn khác liên quan đến button.

#### Carousel Component

javascript

Sao chép mã

var L = "carousel",

j = "bs.carousel",

H = "." + j,

R = ".data-api",

U = g.fn[L],

W = {

interval: 5000,

keyboard: !0,

slide: !1,

pause: "hover",

wrap: !0,

touch: !0

},

x = {

interval: "(number|boolean)",

keyboard: "boolean",

slide: "(boolean|string)",

pause: "(string|boolean)",

wrap: "boolean",

touch: "boolean"

},

F = "next",

q = "prev",

M = "left",

K = "right",

Q = {

SLIDE: "slide" + H,

SLID: "slid" + H,

KEYDOWN: "keydown" + H,

MOUSEENTER: "mouseenter" + H,

MOUSELEAVE: "mouseleave" + H,

TOUCHSTART: "touchstart" + H,

TOUCHMOVE: "touchmove" + H,

TOUCHEND: "touchend" + H,

POINTERDOWN: "pointerdown" + H,

POINTERUP: "pointerup" + H,

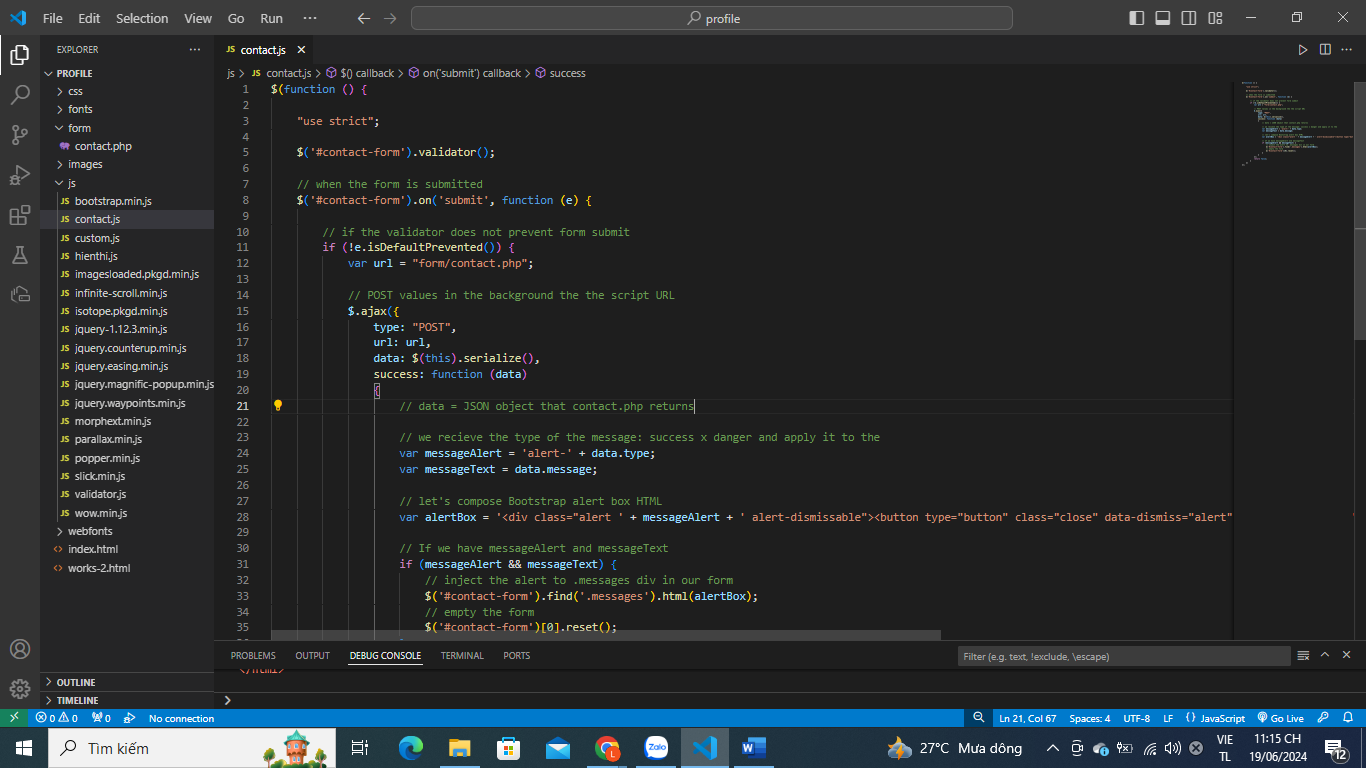
DRAG\_START: "dragstart" + H,

LOAD\_DATA\_API: "load" + H + R,

CLICK\_DATA\_API: "click" + H + R

};

**Mục đích**: Định nghĩa các hằng số và phương thức cho thành phần Carousel trong Bootstrap. Đây bao gồm các sự kiện như slide, slid, keydown, mouseenter, mouseleave, touchstart, touchmove, touchend, pointerdown, pointerup, dragstart, load, click.



### Sự kiện khi tài liệu được tải

javascript

Sao chép mã

$(function () {

"use strict";

// Đoạn mã trong này được thực thi khi DOM đã tải hoàn toàn

});

$(function() { ... });: Đây là một cách viết tắt cho $(document).ready(function() { ... });, đảm bảo rằng đoạn mã trong đó chạy sau khi DOM đã tải xong.

"use strict";: Lệnh này bật chế độ nghiêm ngặt trong JavaScript, giúp phát hiện lỗi lập trình phổ biến và các hành động "không an toàn".

### Khởi tạo Bộ kiểm tra form

javascript

Sao chép mã

$('#contact-form').validator();

$('#contact-form').validator();: Dòng này khởi tạo bộ kiểm tra form sử dụng plugin Bootstrap Validator trên form có id contact-form. Plugin Bootstrap Validator cải thiện tính năng kiểm tra form trong các form dựa trên Bootstrap.

### Xử lý gửi form

javascript

Sao chép mã

$('#contact-form').on('submit', function (e) {

// Ngăn chặn gửi form mặc định của trình duyệt

e.preventDefault();

// Kiểm tra tính hợp lệ của form sử dụng Bootstrap Validator

var isValid = $(this).validator('validate').has('.has-error').length === 0;

// Nếu form hợp lệ

if (isValid) {

var url = "form/contact.php";

// Thực hiện yêu cầu AJAX POST

$.ajax({

type: "POST",

url: url,

data: $(this).serialize(), // Chuyển đổi dữ liệu form thành chuỗi query

success: function (data) {

// Khi thành công, xử lý dữ liệu phản hồi

var messageAlert = 'alert-' + data.type;

var messageText = data.message;

// Tạo HTML của hộp cảnh báo Bootstrap

var alertBox = '<div class="alert ' + messageAlert + ' alert-dismissable"><button type="button" class="close" data-dismiss="alert" aria-hidden="true">&times;</button>' + messageText + '</div>';

// Chèn hộp cảnh báo vào .messages trong form

$('#contact-form').find('.messages').html(alertBox);

// Xóa trắng form

$('#contact-form')[0].reset();

}

});

}

});

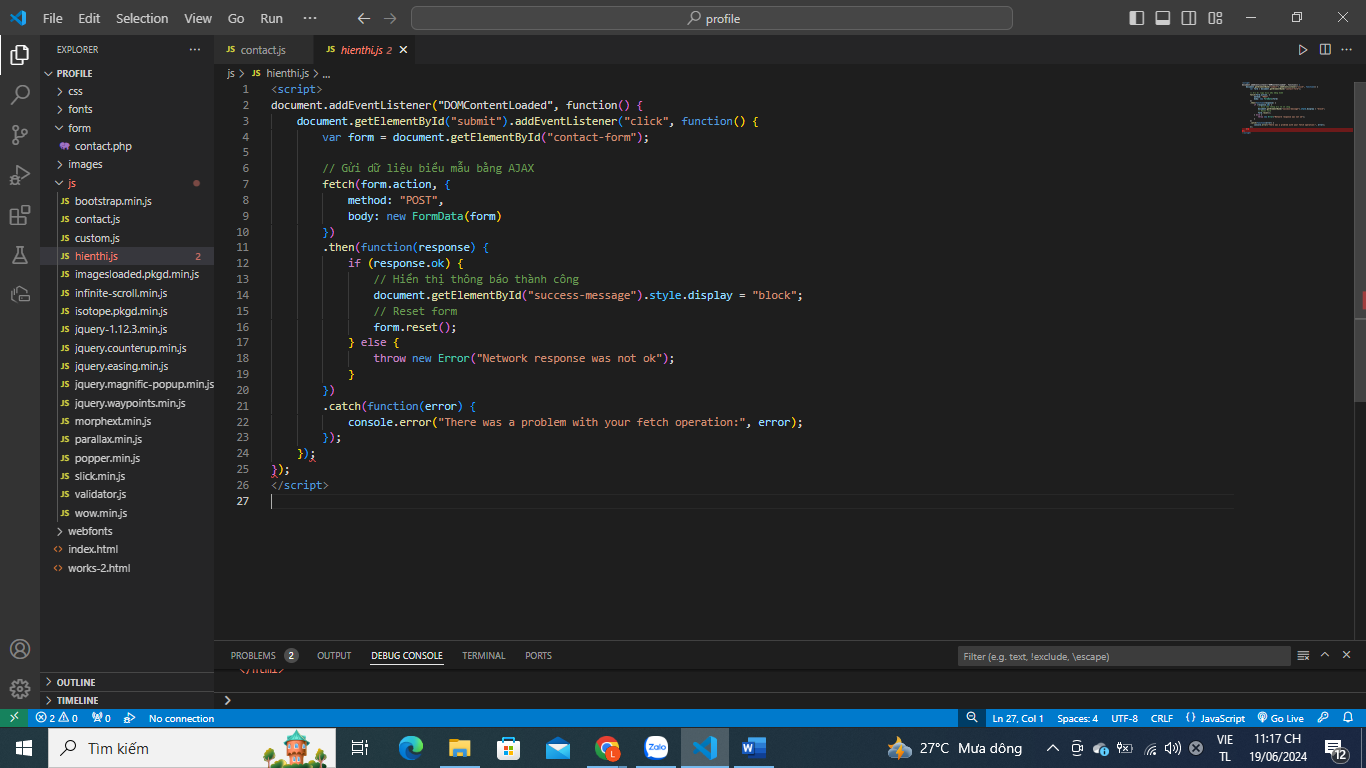
Đoạn mã này thực hiện các bước sau khi người dùng nhấn nút gửi form:

Ngăn chặn gửi form mặc định của trình duyệt (e.preventDefault();).

Kiểm tra tính hợp lệ của form bằng Bootstrap Validator ($(this).validator('validate').has('.has-error').length === 0).

Nếu form hợp lệ, thực hiện một yêu cầu AJAX POST đến url (form/contact.php) để gửi dữ liệu form.

Trong hàm success, xử lý dữ liệu phản hồi từ contact.php. Nó lấy data.type và data.message để tạo ra một hộp cảnh báo Bootstrap (alertBox), sau đó chèn nó vào .messages trong form và làm trống form ($('#contact-form')[0].reset();).



### Sự kiện DOMContentLoaded

javascript

Sao chép mã

document.addEventListener("DOMContentLoaded", function() {

// Đoạn mã trong này được thực thi khi DOM đã tải xong

});

document.addEventListener("DOMContentLoaded", function() { ... });: Đoạn mã này chờ đến khi toàn bộ DOM đã được tải xong (bao gồm cả CSS và ảnh nếu có) trước khi thực thi mã bên trong.

### Xử lý khi nhấn nút gửi (submit)

javascript

Sao chép mã

document.getElementById("submit").addEventListener("click", function() {

var form = document.getElementById("contact-form");

// Gửi dữ liệu biểu mẫu bằng AJAX

fetch(form.action, {

method: "POST",

body: new FormData(form)

})

.then(function(response) {

if (response.ok) {

// Hiển thị thông báo thành công

document.getElementById("success-message").style.display = "block";

// Reset form

form.reset();

} else {

throw new Error("Network response was not ok");

}

})

.catch(function(error) {

console.error("There was a problem with your fetch operation:", error);

});

});

document.getElementById("submit").addEventListener("click", function() { ... });: Đoạn mã này gán một sự kiện nhấn cho nút có id là submit. Khi nút này được nhấn, hàm được định nghĩa bên trong sẽ được thực thi.

var form = document.getElementById("contact-form");: Lấy đối tượng biểu mẫu từ DOM có id là contact-form.

fetch(form.action, { method: "POST", body: new FormData(form) }): Sử dụng fetch để gửi yêu cầu POST đến URL được chỉ định trong thuộc tính action của biểu mẫu (form.action). Dữ liệu gửi đi là một đối tượng FormData được tạo từ biểu mẫu (new FormData(form)).

.then(function(response) { ... }): Xử lý phản hồi từ máy chủ sau khi gửi yêu cầu AJAX. Nếu phản hồi thành công (response.ok là true), thực hiện các hành động sau:

Hiển thị thông báo thành công (có id là success-message) bằng cách đặt style.display thành "block".

Đặt lại biểu mẫu bằng cách gọi form.reset().

Nếu không thành công, ném ra một lỗi để bắt bởi .catch().

.catch(function(error) { ... }): Xử lý các lỗi xảy ra trong quá trình gửi yêu cầu AJAX. Lỗi được hiển thị trong console bằng console.error.

### HTML tương ứng

Để đoạn mã JavaScript này hoạt động, bạn cần có các phần tử HTML tương ứng như sau:

html

Sao chép mã

<form id="contact-form" action="form/contact.php" method="POST">

<!-- Các trường nhập liệu của biểu mẫu -->

<input type="text" name="name" placeholder="Your Name" required>

<input type="email" name="email" placeholder="Your Email" required>

<!-- Nút gửi biểu mẫu -->

<button type="submit" id="submit">Submit</button>

</form>

<!-- Thông báo thành công (ban đầu ẩn đi) -->

<div id="success-message" style="display: none;">

Your message has been sent successfully.

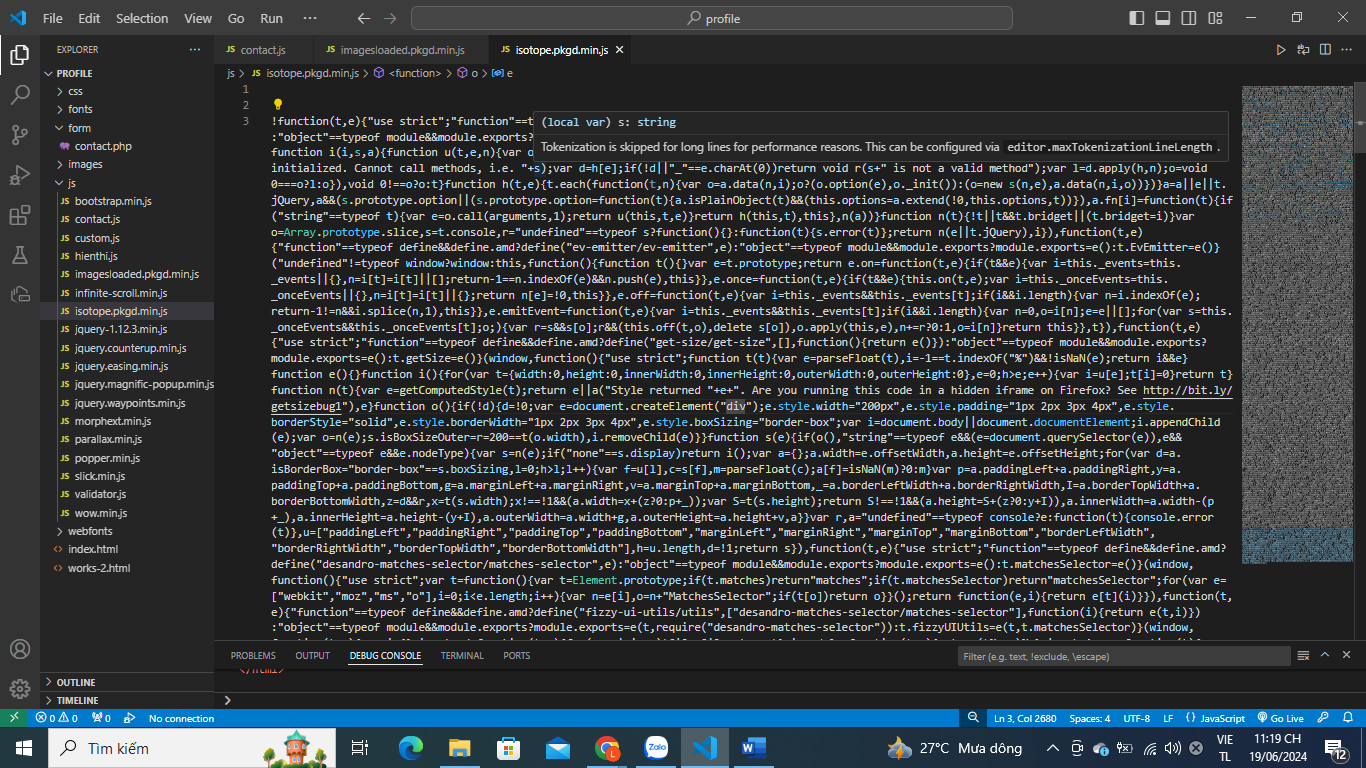
</div>

### Giải thích:

<form id="contact-form" action="form/contact.php" method="POST">: Định nghĩa một biểu mẫu có id là contact-form và sử dụng phương thức POST để gửi dữ liệu đến form/contact.php.

<button type="submit" id="submit">Submit</button>: Nút gửi biểu mẫu có id là submit, dùng để kích hoạt sự kiện gửi form trong JavaScript.

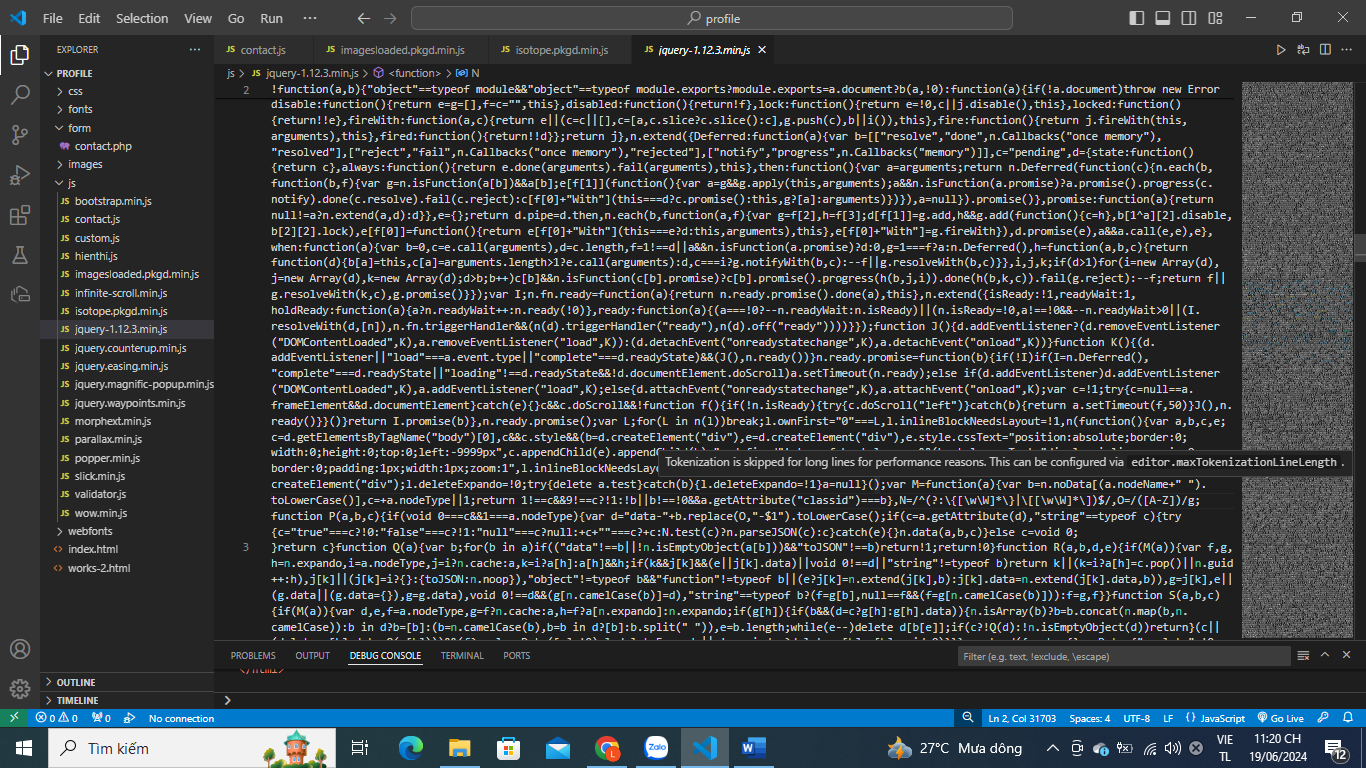
<div id="success-message" style="display: none;">: Thẻ div để hiển thị thông báo thành công khi form được gửi thành công. Ban đầu được ẩn đi (style="display: none;")

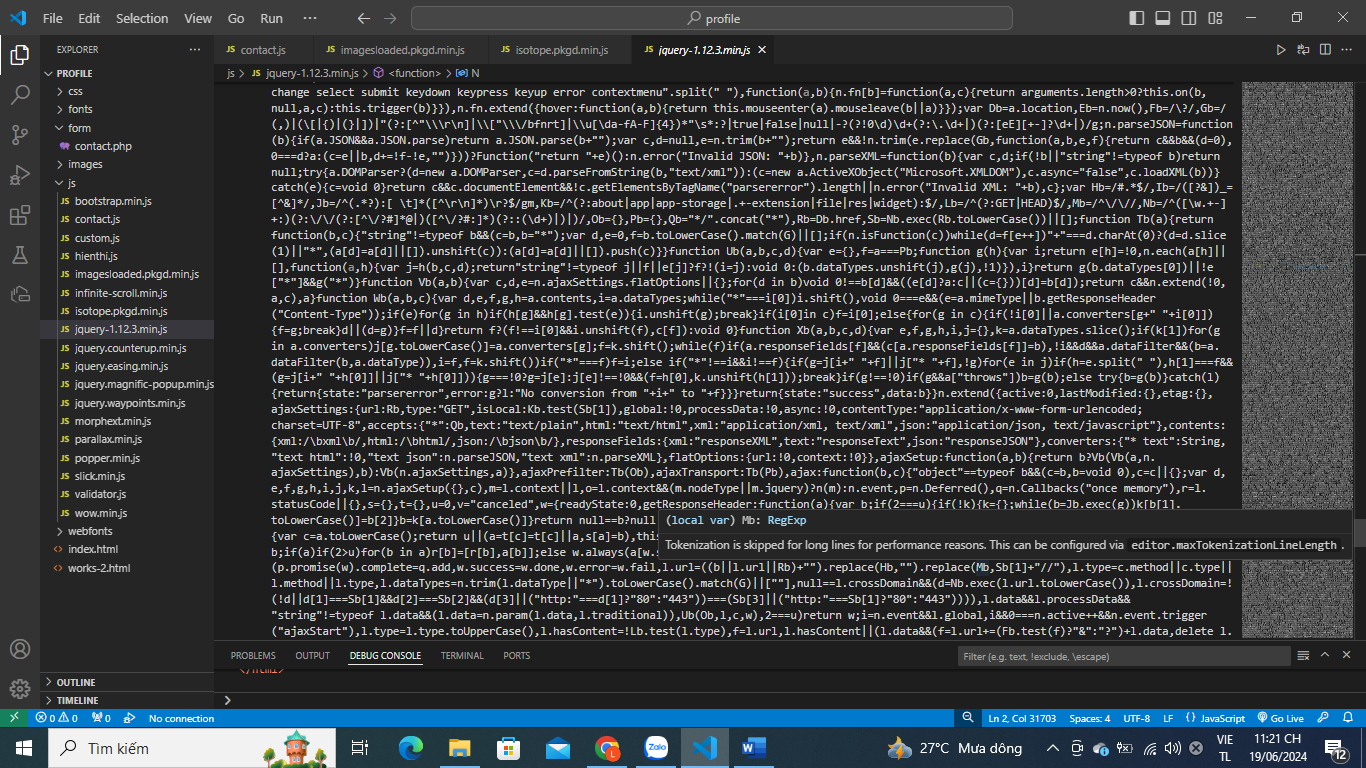
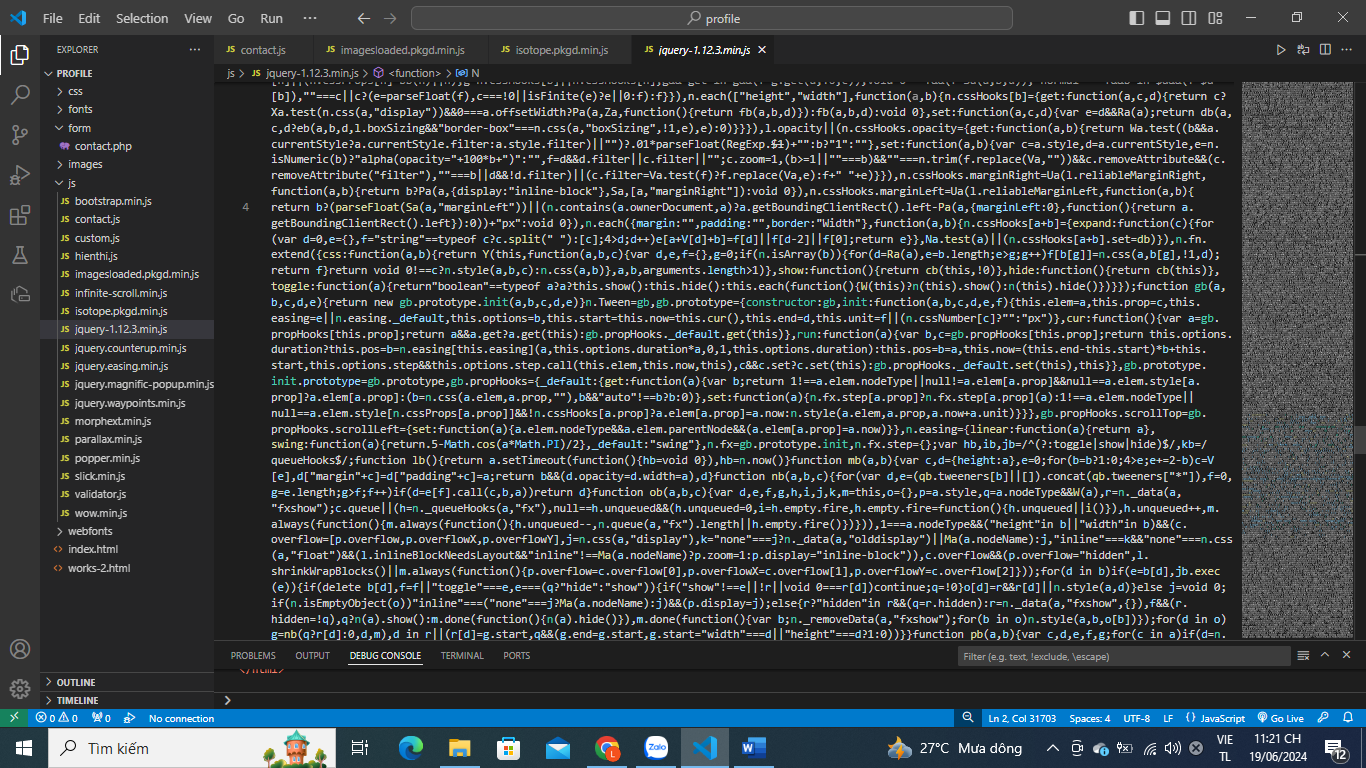
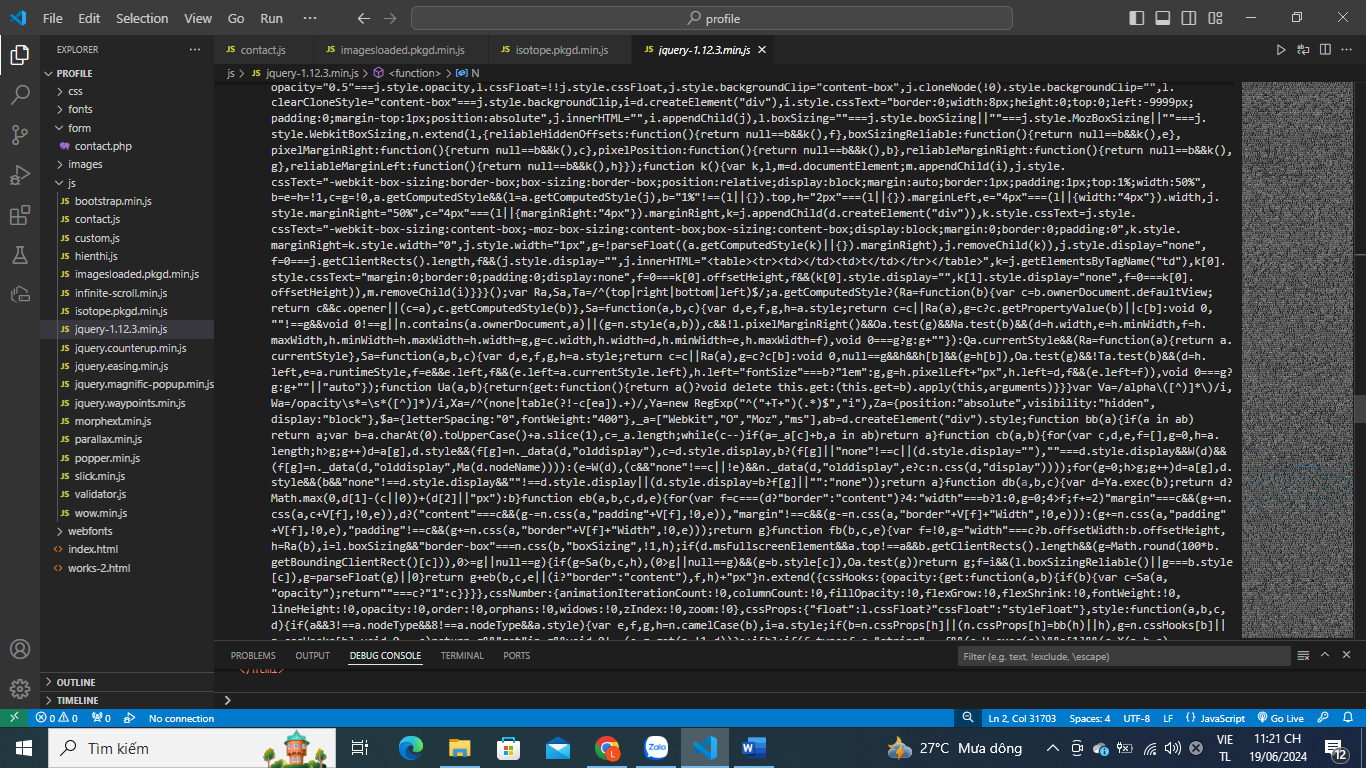
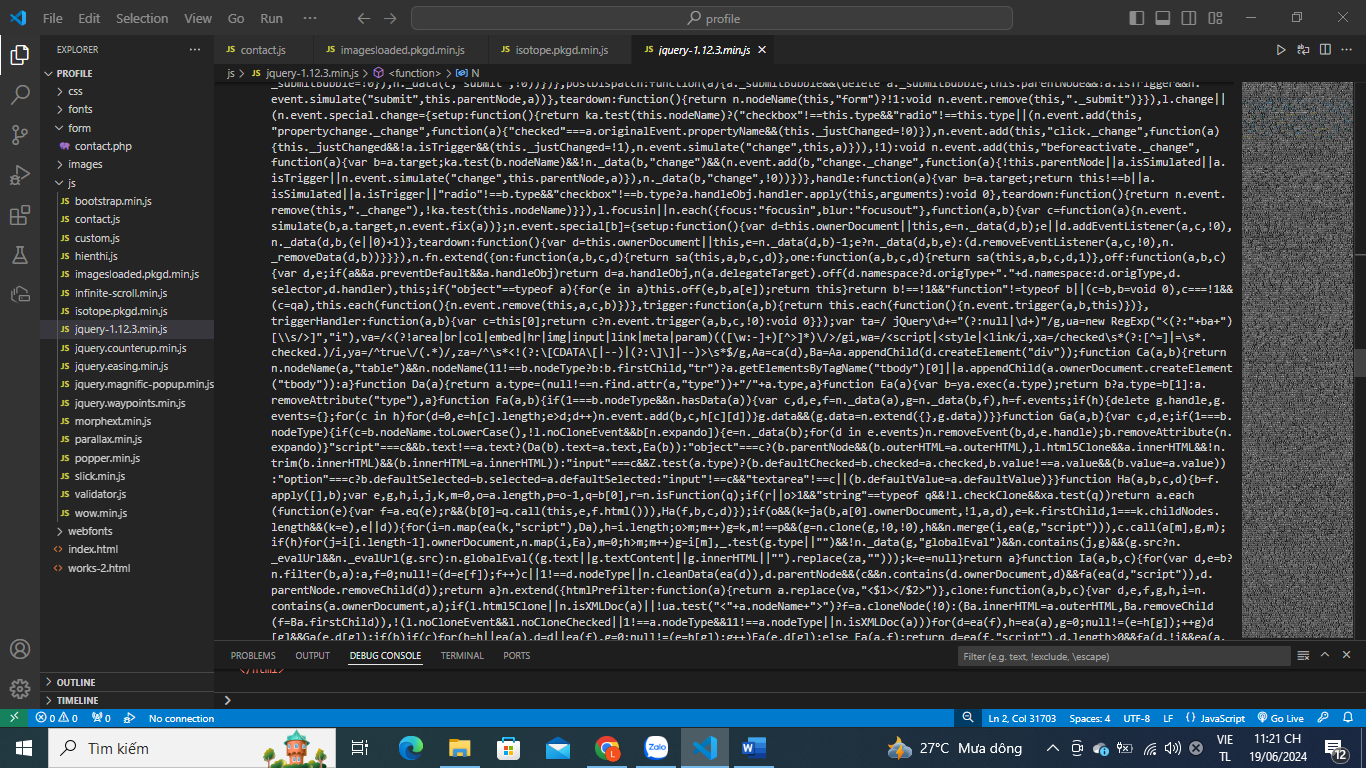
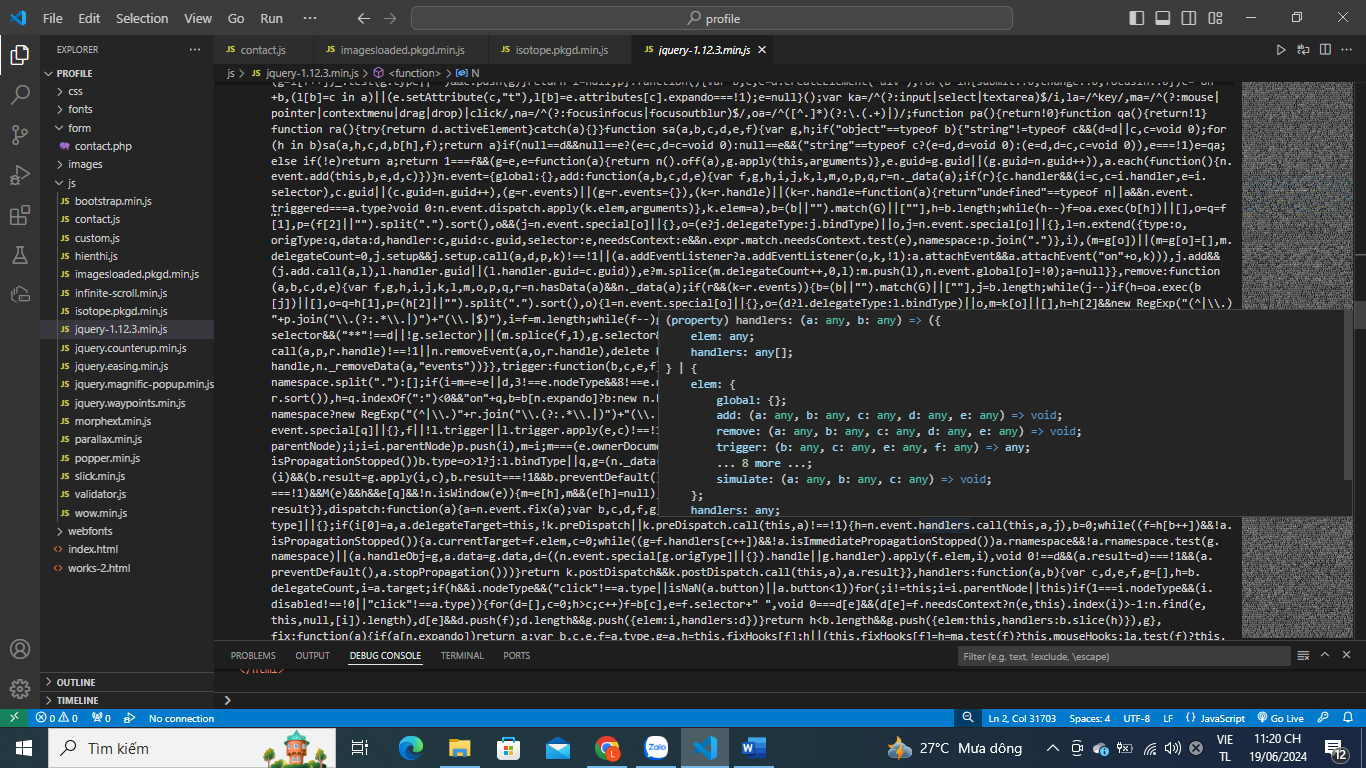
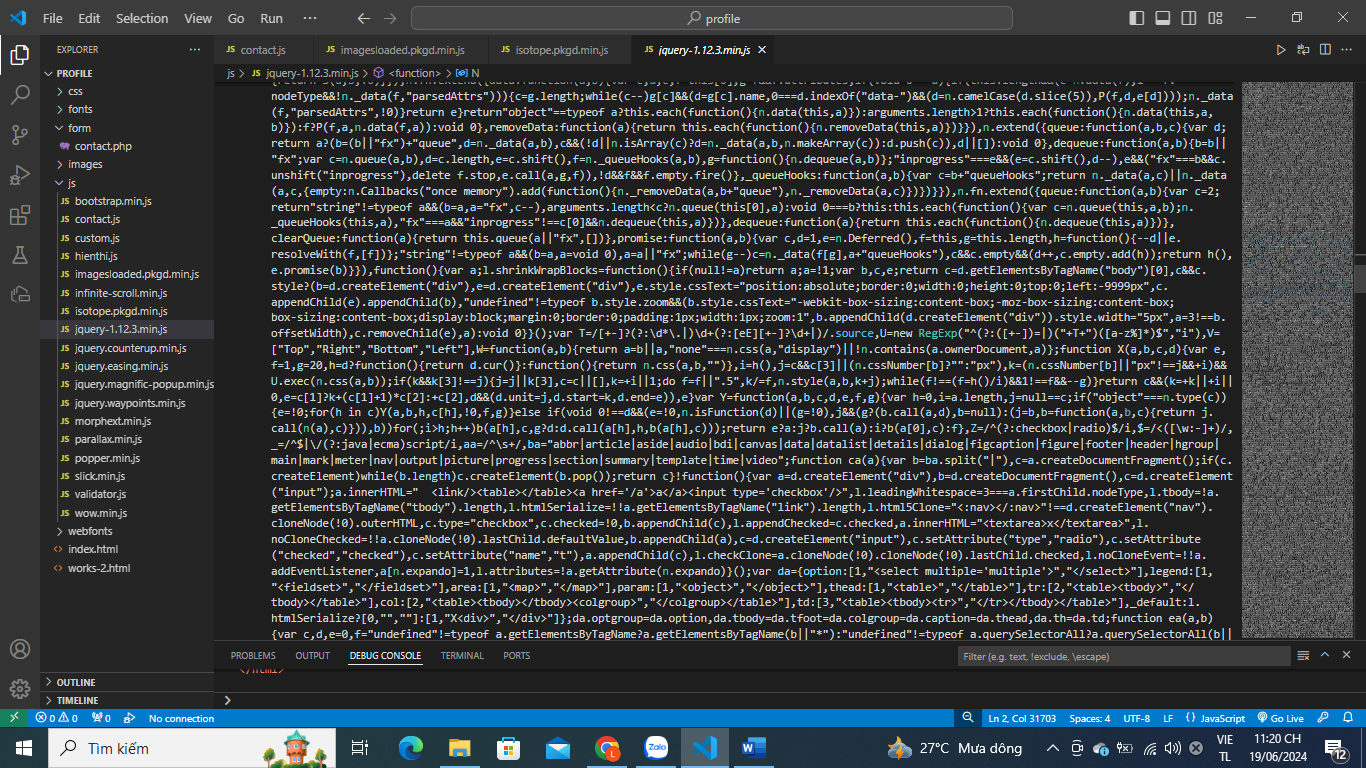


**Mục đích của s.compatOptions**: Đối tượng này chứa các tùy chọn tương thích (compatibility options) dùng để cấu hình và điều khiển hành vi của framework. Các tùy chọn này có thể được sử dụng để xác định các thiết lập quan trọng liên quan đến layout, hiệu ứng chuyển đổi, và các tính năng khác của framework.

**Thuộc tính isHideRevealTransitionEnd**: Thuộc tính này có thể được sử dụng để xác định tùy chọn liên quan đến sự kết thúc của hiệu ứng chuyển đổi khi các phần tử được ẩn hoặc hiện. Trong một framework quản lý layout và hiệu ứng như Outlayer, việc định nghĩa một thuộc tính như này giúp quản lý các sự kiện kết thúc chuyển đổi một cách hiệu quả, cung cấp tính linh hoạt trong việc điều chỉnh hành vi và giao diện người dùng.

**Mục đích của đoạn mã**: Mã này cung cấp một khung làm việc để xây dựng và quản lý các phần tử trong giao diện người dùng, đồng thời hỗ trợ các hiệu ứng chuyển đổi một cách mượt mà và dễ dàng. Việc định nghĩa isHideRevealTransitionEnd nhấn mạnh vào tính linh hoạt và tiện ích của framework trong việc xử lý các hiệu ứng chuyển đổi.





**Change Select Submit Keydown Keypress Keyup Error Contextmenu**:

Đoạn này gắn các sự kiện nhất định vào phần tử jQuery. Sự kiện này sẽ được gắn vào các phần tử mỗi khi bạn gọi phương thức .fn[b].

Các sự kiện bao gồm: "change", "select", "submit", "keydown", "keypress", "keyup", "error", "contextmenu".

javascript

Sao chép mã

"change select submit keydown keypress keyup error contextmenu".split(" ")

**Extend**:

Đoạn này mở rộng các phương thức của đối tượng jQuery (n.fn).

Phương thức .fn[b] cho phép bạn gắn các hàm xử lý cho sự kiện được đặt tên b.

Nếu có đối số đầu vào, nó sẽ gắn sự kiện với phần tử hiện tại; nếu không có đối số, nó sẽ kích hoạt sự kiện đó trên các phần tử hiện tại.

javascript

Sao chép mã

"change select submit keydown keypress keyup error contextmenu".split(" ").forEach(function(a, b) {

n.fn[a] = function(a, c) {

return arguments.length > 0 ? this.on(a, null, a, c) : this.trigger(a);

}

})

**Hover**:

Phương thức .hover() của jQuery là viết tắt của .mouseenter() và .mouseleave().

Nó cho phép gắn các hàm xử lý khi con trỏ chuột đi vào và ra khỏi phần tử.

javascript

Sao chép mã

n.fn.extend({

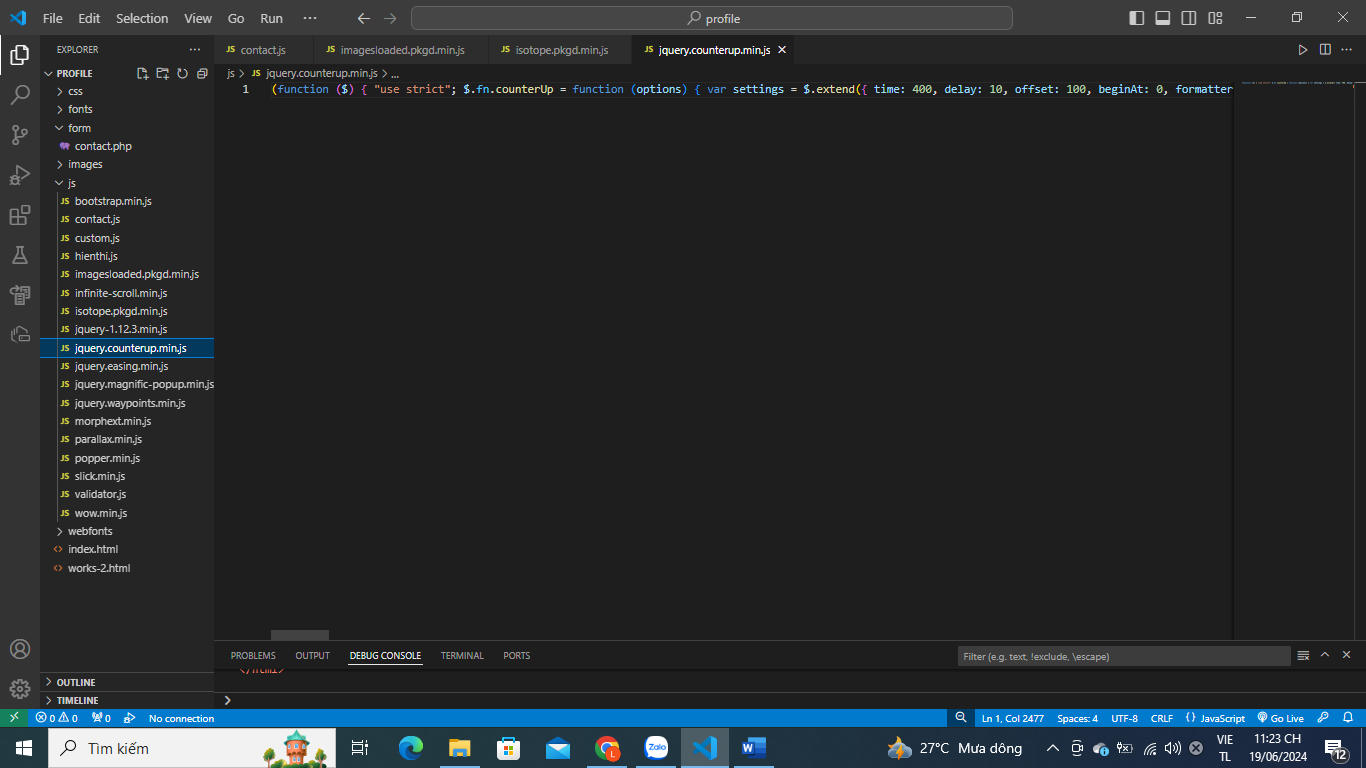
hover: function(a, b) {

return this.mouseenter(a).mouseleave(b || a);

}

});

Đây là những đoạn mã cơ bản của thư viện jQuery để mở rộng và gắn các sự kiện và phương thức vào các phần tử DOM một cách dễ dàng. Chúng giúp cho việc xử lý và tương tác với DOM trở nên đơn giản và hiệu quả hơn.



**Khởi tạo Plugin jQuery**:

javascript

Sao chép mã

(function($) {

"use strict";

$.fn.counterUp = function(options) {

var settings = $.extend({

time: 400,

delay: 10,

offset: 100,

beginAt: 0,

formatter: false,

context: "window",

callback: function() {}

}, options), s;

return this.each(function() {

// Code xử lý số đếm sẽ được thêm ở đây

});

};

})(jQuery);

Plugin được định nghĩa bằng cách mở rộng phương thức counterUp của đối tượng jQuery ($.fn.counterUp).

Các tùy chọn mặc định được định nghĩa trong đối tượng settings. Các tùy chọn này bao gồm thời gian (time), độ trễ (delay), lệch (offset), giá trị ban đầu (beginAt), hàm định dạng (formatter), ngữ cảnh (context), và hàm callback (callback).

**Hàm counterUpper**:

Hàm này được gọi cho mỗi phần tử trong jQuery collection (this.each()). Nó được sử dụng để tính toán các giá trị số đếm và áp dụng hiệu ứng số đếm lên phần tử đó.

**Biến counter**: Lưu trữ các tùy chọn cụ thể cho từng phần tử.

**Tính toán số đếm**: Sử dụng các biến như divisions, num, isComma, decimalPlaces, và isTime để xử lý số liệu và chuẩn bị cho hiệu ứng số đếm.

**Xử lý số thời gian (isTime)**: Nếu số là dạng thời gian (hh:mm

), thì nó sẽ được chuyển đổi và định dạng lại.

**Định dạng số có dấu phẩy (isComma)**: Nếu số có định dạng có dấu phẩy (ví dụ: 1,000,000), sẽ được xử lý để hiển thị đúng.

**Hàm counterUpper được gọi thông qua waypoint**:

Plugin sử dụng Waypoints để gọi hàm counterUpper khi phần tử hiển thị trong view. Waypoints là thư viện JavaScript cho phép bạn chạy mã JavaScript khi một phần tử đạt tới một điểm nhất định trong tài liệu.

**Hàm formatter**:

Nếu được cung cấp, hàm formatter sẽ được gọi để định dạng lại giá trị số trước khi hiển thị.

**Kết quả cuối cùng**:

Sau khi tính toán, số cuối cùng sẽ được hiển thị lên phần tử và hàm callback sẽ được gọi khi hoàn thành.

Chức năng của trang Web (Hình ảnh phải ghi chú và phân tích kết quả của chức năng trang web)

**Chức năng chính:**

Hiển thị thông tin cá nhân và hồ sơ chuyên môn của người sử dụng.

Cho phép người dùng đăng nhập và quản lý hồ sơ của họ.

Cung cấp các công cụ và tính năng để tạo, chỉnh sửa và cập nhật hồ sơ.

**Các thành phần chính của trang web:**

**Trang chủ:**

Mô tả tổng quan về dịch vụ và lợi ích của việc sử dụng trang web.

Lựa chọn đăng nhập hoặc đăng ký nếu người dùng chưa có tài khoản.

**Đăng nhập và Đăng ký:**

Form đăng nhập và form đăng ký để người dùng có thể truy cập vào hồ sơ cá nhân của mình.

**Hồ sơ cá nhân:**Thông tin cơ bản: tên, địa chỉ, thông tin liên hệ.

Hồ sơ chuyên môn: kinh nghiệm làm việc, bằng cấp, kỹ năng chuyên môn.

Dự án đã làm và thành tựu nghề nghiệp.

**Công cụ quản lý hồ sơ:**

Chỉnh sửa thông tin cá nhân.

Tải lên hồ sơ mới.

Quản lý danh sách công việc đã làm hoặc đang làm.

**Tính năng bảo mật:**

Xác thực hai bước hoặc mật khẩu mạnh để bảo vệ thông tin cá nhân của người dùng.

Quản lý quyền truy cập và phân quyền cho người dùng.

**Phân tích kết quả của chức năng trang web:**

**Thuận lợi:**

**Tiện ích trong việc quản lý hồ sơ**: Người dùng có thể dễ dàng tạo, cập nhật và quản lý hồ sơ chuyên nghiệp của mình.

**Tăng tính tương tác**: Các công cụ như đăng nhập, đăng ký và quản lý hồ sơ giúp tăng tính tương tác và thuận tiện cho người dùng.

**Bảo mật thông tin**: Các tính năng bảo mật như xác thực hai bước giúp bảo vệ thông tin cá nhân của người dùng.

**Thách thức:**

**Đáp ứng yêu cầu đa dạng của người dùng**: Cần đảm bảo rằng hệ thống có thể xử lý nhiều loại hồ sơ và yêu cầu từ người dùng khác nhau.

**Bảo mật và quản lý dữ liệu**: Cần có chiến lược rõ ràng để đảm bảo an toàn thông tin và tuân thủ các quy định bảo mật.

**Cải tiến tiềm năng:**

**Tích hợp công nghệ mới**: Cập nhật thường xuyên để tích hợp các công nghệ mới giúp cải thiện trải nghiệm người dùng và hiệu quả làm việc.

**Phát triển tính năng mới**: Phản hồi từ người dùng để cải thiện và mở rộng tính năng của trang web.

**Trang chủ(index)**



**Link index : https://gameran2.github.io/profile2/**

**Giao diện trang web:**

**Header (<header>)**:

**Logo**: Được đặt bên trái và có thể là liên kết về trang chủ.

**Menu Navigation (<nav>)**: Chứa danh sách các liên kết đến các trang khác của trang web.

**Body Content (<section class="body-content">)**:

**Container (<div class="container">)**: Bao gồm nội dung chính của trang web.

**Tiêu đề và nội dung**: Đây là ví dụ về một phần nội dung cơ bản.

**CSS**:

Đảm bảo reset các giá trị margin và padding mặc định.

Thiết lập các style cơ bản cho header, logo, menu và nội dung chính.

Sử dụng lưới (container) để căn chỉnh và giới hạn chiều rộng của nội dung.

## Kết luận chương 3

Trang profile này không chỉ giúp bạn tổng kết lại các thành tựu và kỹ năng của mình mà còn giúp người xem có cái nhìn tổng quan và sâu sắc hơn về bạn. Đồng thời, đảm bảo rằng các trang này được thiết kế sáng tạo và chuyên nghiệp để tăng tính thẩm mỹ và tạo ấn tượng tốt đối với mọi người.

**KẾT LUẬN**

***(Trình bày thành 3 đoạn văn nêu Ưu điểm, nhược điểm và hướng phát triển chủ đề)***

### Ưu điểm của Profile Công việc

Profile công việc là một công cụ quan trọng giúp bạn tự giới thiệu và gây ấn tượng với nhà tuyển dụng. Việc tạo một profile công việc có thể mang lại nhiều lợi ích như sau:

1. **Tự tin và Chuyên nghiệp:** Một profile công việc tốt giúp bạn tự tin khi gửi CV và đơn xin việc, thể hiện sự chuyên nghiệp và sự chuẩn bị kỹ lưỡng của bạn đối với vị trí mong muốn.
2. **Tập trung vào Kỹ năng và Kinh nghiệm:** Nó cho phép bạn tập trung vào những kỹ năng, kinh nghiệm và thành tựu quan trọng nhất mà bạn đã đạt được trong sự nghiệp của mình, giúp nhà tuyển dụng dễ dàng nhận thấy giá trị mà bạn có thể mang đến.
3. **Tiếp cận mở rộng:** Profile công việc có thể giúp bạn tiếp cận với nhiều cơ hội việc làm hơn. Nó có thể được chia sẻ trực tiếp với nhà tuyển dụng hoặc được đưa lên các nền tảng mạng xã hội chuyên nghiệp như LinkedIn để mở rộng mạng lưới liên hệ.

### Nhược điểm của Profile Công việc

Tuy nhiên, cũng có những nhược điểm cần lưu ý khi tạo profile công việc:

1. **Độ phổ biến:** Một số profile công việc có thể không thu hút sự chú ý nếu không được cập nhật thường xuyên hoặc không phù hợp với tiêu chí tuyển dụng hiện tại.
2. **Khả năng tái sử dụng:** Một số nền tảng yêu cầu bạn phải tạo profile công việc theo đúng các mẫu có sẵn, không cho phép tùy chỉnh nhiều hoặc không thể tái sử dụng cho các mục đích khác nếu không có sự phù hợp.
3. **Cạnh tranh:** Trên các nền tảng phổ biến như LinkedIn, sự cạnh tranh để thu hút sự chú ý của nhà tuyển dụng là rất cao, đòi hỏi bạn phải có một profile công việc nổi bật hơn để được lựa chọn.

### Hướng phát triển của Profile Công việc

Để tối ưu hóa profile công việc và đạt được hiệu quả cao, bạn có thể cân nhắc các hướng phát triển sau:

1. **Cập nhật thường xuyên:** Đảm bảo rằng profile công việc của bạn luôn được cập nhật với các kỹ năng mới, dự án thành công và các chứng chỉ mới nhất mà bạn đã đạt được.
2. **Tùy chỉnh và tối ưu hóa:** Tối ưu hóa profile công việc để phù hợp với các vị trí tuyển dụng cụ thể. Hãy sử dụng các từ khóa quan trọng mà nhà tuyển dụng có thể tìm kiếm để dễ dàng được tìm thấy.
3. **Xây dựng mạng lưới:** Sử dụng profile công việc để mở rộng mạng lưới liên hệ chuyên nghiệp. Tham gia vào các nhóm và diễn đàn có liên quan để tăng khả năng tiếp cận với cơ hội việc làm mới.

Thông qua việc xây dựng và phát triển một profile công việc hiệu quả, bạn có thể tăng cơ hội thu hút sự quan tâm của nhà tuyển dụng và tiến xa hơn trong sự nghiệp của mình.

**DANH MỤC TÀI LIỆU THAM KHẢO**

1. ThS. Lê Văn Phong (2024), *Bài giảng Thiết kế, lập trình Front-end*, Trường Đại học Đại Nam.
2. Phạm Quốc Hùng (2017), *Đề cương bài giảng Mạng máy tính*, Đại học SPKT Hưng Yên.
3. James F. Kurose and Keith W. Ross (2013), *Computer Networking: A top-down approach sixth Edition*, Pearson Education.

**QUY ĐỊNH TRÌNH BÀY TRONG BÁO CÁO CÀI TẬP LỚN**

- Bài tập lớn được in trên một mặt giấy trắng khổ A4 (210 x 297mm), dày lớn hơn 30 trang, nhỏ hơn 100 trang, không kể hình vẽ, bảng biểu, đồ thị và danh mục tài liệu tham khảo.

- Phần nội dung trình bày trong bài tập lớn sử dụng Font chữ **Times New Roman** cỡ **13**, hệ soạn thảo Microsoft Word; mật độ chữ bình thường, không được nén hoặc kéo dãn khoảng cách giữa các chữ; dãn dòng đặt ở chế độ **1,5 lines**; lề trên **2,0 cm**; lề dưới **2,0 cm**; lề trái **2,5 cm**, lề phải **2,0 cm**. Số trang được đánh ở giữa, phía dưới trang giấy.

- Cách ghi trích dẫn tài liệu tham khảo: Cuối đoạn trích dẫn đánh số thứ tự tài liệu tham khảo (ví dụ: [1]: tham khảo tài liệu số 1; [3,4,8]: tham khảo 3 tài liệu số 3, 4, 8).

- Tuyệt đối không được tẩy, xoá, sửa chữa trong bài tập lớn.

- Quy cách trình bày nội dung

|  |  |  |  |
| --- | --- | --- | --- |
| **Đề mục** | **Cỡ chữ** | **Định dạng** | **Canh lề trang** |
| Tên chương | 14 | In hoa, đậm | Giữa |
| Tên tiểu mục mức 1 | 13 | Chữ thường, đậm | Trái |
| Tên tiểu mục mức 2 | 13 | Chữ thường, đậm, nghiêng | Trái |
| Tên tiểu mục mức 3 | 13 | Đánh chỉ mục bằng chữ cái thường  a), ...  b), ..... | Trái |
| Nội dung | 13 | Chữ thường (Normal) | Đều hai bên |
| Nội dung bảng (table) | 12 | Normal | Giữa ô |
| Tên bảng | 12 | Chữ thường, nghiêng | Giữa, trên bảng |
| Tên hình | 12 | Chữ thường, nghiêng | Giữa, dưới hình |
| Tài liệu tham khảo | 12 | APA style | Chú thích bên dưới |

***Cách đánh dấu câu****:*

Các dấu: : , . ; ) } ] ! ? ” được gõ ngay sau ký tự cuối cùng (không khoảng cách), và gõ 1 phím cách (space) sau chúng. Sau các dấu: “{ ( [ không gõ dấu cách.

***\* Cách đánh số các tiểu đề mục nhiều nhất là 3 mức và không lùi sang phải***

Hướng dẫn xếp tài liệu tham khảo

1. Tài liệu tham khảo xếp theo thứ tự trích dẫn trong bài tập lớn.
2. Tài liệu tham khảo là sách, luận án, báo cáo phải ghi đầy đủ các thông tin sau:

* Tên các tác giả hoặc cơ quan ban hành (không có dấu ngăn cách)
* (năm xuất bản), (đặt trong ngoặc đơn, dấu phẩy sau ngoặc đơn)
* Tên sách, luận án hoặc báo cáo, (in nghiêng, dấu phẩy cuối tên)
* Nhà xuất bản, (dấu phẩy cuối tên nhà xuất bản)
* Nơi xuất bản, (dấu chấm kết thúc tài liệu tham khảo).

Tài liệu tham khảo là bài báo trong tạp chí, bài trong một cuốn sách… ghi đầy đủ các thông tin sau:

* Tên các tác giả (không có dấu ngăn cách);
* (Năm công bố), (đặt trong ngoặc đơn, dấu phẩy sau ngoặc đơn)
* “Tên bài báo”, (đặt trong ngoặc kép, không in nghiên, dấu phẩy cuối tên)
* Tên tạp chí hoặc tên sách, (in nghiên, dấu phẩy cuối tên)
* Tập (không có dấu ngăn cách)
* (Sổ), (đặt trong ngoặc đơn, dấu phẩy sau ngoặc đơn)
* Các số trang, (gạch ngang giữa hai chữ số, dấu chấm kết thúc)

Cần chú ý những chi tiết về trình bày nêu trên. Nếu tài liệu dài hơn một dòng thì nên trình bày sau cho từ dòng thứ hai lùi vào so với dòng thứ nhất 1 cm để phần tài liệu tham khảo được rõ ràng và dễ theo dõi.

Ví dụ:

1. Nguyễn Hồng Sơn (2007), *Giáo trình hệ thống Mạng máy tính CCNA* (Semester 1), NXB Lao động xã hội.
2. Phạm Quốc Hùng (2017), *Đề cương bài giảng Mạng máy tính*, Đại học SPKT Hưng Yên.
3. James F. Kurose and Keith W. Ross (2013), *Computer Networking: A top-down approach sixth Edition*, Pearson Education.