**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**THÁI NGUYÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

----------- o0o -----------

|  |  |
| --- | --- |
|  | Ảnh 3 x 4 |

**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Ngành kỹ thuật phần mềm**

***Đề tài*:**

**XÂY DỰNG WEBSITE HỖ TRỢ QUẢN LÝ VÀ THEO DÕI QUÁ TRÌNH HỌC TẬP CỦA HỌC VIÊN TẠI TRUNG TÂM NGOẠI NGỮ VIETKIDS**

**Sinh viên thực hiện: NGUYỄN QUANG LINH**

**Lớp: KTPM – K18A HỆ CHÍNH QUY**

**Giáo viên hướng dẫn: ThS. NGUYỄN THỊ DUNG**

**THÁI NGUYÊN - 2024**

**LỜI CẢM ƠN**

Trong thời gian làm đồ án tốt nghiệp, em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy cô, gia đình và bạn bè.

Em xin gửi lời cảm ơn chân thành đến cô Nguyễn Thị Dung - giảng viên Bộ môn CNPM - Trường CNTT và Truyền thông Thái Nguyên người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm khoá luận.

Em cũng xin chân thành cảm ơn các thầy cô giáo trong trường nói chung, các thầy cô trong khoa CNTT nói riêng đã dạy dỗ cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ em trong suốt quá trình học tập.

Cuối cùng, em xin chân thành cảm ơn gia đình và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên em trong suốt quá trình học tập và hoàn thành khoá luận tốt nghiệp.

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

**CAM ĐOAN**

Tôi xin cam đoan đề tài “**Xây dựng website hỗ trợ quản lý và theo dõi quá trình học tập của học viên tại trung tâm ngoại ngữ Vietkids**” trong đồ án tốt nghiệp của tôi được tiến hành một cách công khai và minh bạch, dựa trên sự cố gắng và nỗ lực của bản thân cũng như sự giúp đỡ, hướng dẫn tận tình của đơn vị chủ quản.

Các số liệu nghiên cứu nêu trong đồ án đảm bảo tính trung thực, không sao chép hay sử dụng kết quả của bất kỳ công trình nào đã được công bố trước đây. Nếu phát hiện có sự sao chép, bản thân tôi xin chịu hoàn toàn trách nhiệm và kỷ luật từ phía nhà trường.

**MỤC LỤC**

[LỜI MỞ ĐẦU 6](#_Toc165473077)

[CHƯƠNG 1. 7](#_Toc165473078)

[1.1. Tìm hiểu về các ngôn ngữ lập trình 7](#_Toc165473079)

[1.1.1. Ngôn ngữ HTML 7](#_Toc165473080)

[1.1.2. Ngôn ngữ CSS 9](#_Toc165473081)

[1.1.3. Ngôn ngữ Javascript 11](#_Toc165473082)

[1.1.4. Ngôn ngữ PHP 12](#_Toc165473083)

[1.2. Laravel framework 14](#_Toc165473084)

[1.3. Vuejs Framework 18](#_Toc165473085)

[1.4. Hệ quản trị cơ sở dữ liệu (MySQL) 19](#_Toc165473086)

[CHƯƠNG 2. 22](#_Toc165473087)

[2.1. Đặt vấn đề 22](#_Toc165473088)

[2.2. Phân tích yêu cầu 23](#_Toc165473089)

[2.2.1. Danh sách actor tác động vào hệ thống 23](#_Toc165473090)

[2.2.2. Mục đích 23](#_Toc165473091)

[2.3. Xác định yêu cầu phi chức năng của hệ thống 23](#_Toc165473092)

[2.4. Biểu đồ phân tích thiết kế hệ thống 24](#_Toc165473093)

[2.4.1. Biểu đồ usecase 24](#_Toc165473094)

[2.4.2. Đặc tả use case 28](#_Toc165473095)

[2.4.3 Biểu đồ trình tự 36](#_Toc165473096)

[2.4.4. Biểu đồ hoạt động 41](#_Toc165473097)

[*2.4.5. Biểu đồ lớp* 47](#_Toc165473098)

[CHƯƠNG 3. 49](#_Toc165473099)

[DEMO SẢN PHẨM WEBSITE 49](#_Toc165473100)

[3.1. Giao diện 49](#_Toc165473101)

[3.1.1 Giao diện người dùng 49](#_Toc165473102)

[3.1.2. Giao diện nội dung bài học 49](#_Toc165473103)

[3.1.3. Giao diện làm bài kiểm tra 50](#_Toc165473104)

[3.2. Giao diện quản trị 50](#_Toc165473105)

[3.2.1. Giao diện đăng nhập 50](#_Toc165473106)

[3.2.2. Giao diện đăng ký 51](#_Toc165473107)

[3.2.3. Giao diện trang quản trị 51](#_Toc165473108)

[3.2.4. Quản lý cấp độ bài học 52](#_Toc165473109)

[3.2.5 Giao diện quản lý câu hỏi 53](#_Toc165473110)

[3.2.6. Giao diện trang quản lý người dùng 53](#_Toc165473111)

[3.2.7. Giao diện trang quản lý vai trò 54](#_Toc165473112)

[KẾT LUẬN 56](#_Toc165473113)

[TÀI LIỆU THAM KHẢO 57](#_Toc165473114)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 58](#_Toc165473115)

**LỜI MỞ ĐẦU**

Ngày nay công nghệ thông tin đã và đang đóng vai trò quan trọng trong xã hội. Nó giúp con người làm việc với mức độ chính xác cao, quản lý và tổ chức công việc đạt hiệu quả, cũng như thông tin được phổ biến và cập nhật nhanh chóng, chính xác. Do đó việc ứng dụng công nghệ thông tin vào các phòng ban công ty, xí nghiệp, trường học là nhu cầu tất yếu.

Thấy được lợi ích của ứng dụng công nghệ thông tin, trung tâm ngoại ngữ Vietkids đang từng bước tin học hóa vào công tác quản lý. Bởi lẽ công việc quản lý này đòi hỏi phải xử lý một số lượng lớn thông tin với mức độ chính xác cao. Trong khi số lượng học viên ngày càng gia tăng nên công việc quản lý dễ dẫn đến sai sót, tốn thời gian là không thể tránh khỏi. Vì thế sự ra đời của chương trình quản lý học viên mang lại những lợi ích:

* Tiết kiệm chi phí, thời gian, tránh sai sót cho người quản lý
* Công tác lưu trữ được an toàn và tiện ích

Xuất phát từ những lợi ích trên, em đã chọn đề tài tốt nghiệp “ Xây dựng website hỗ trợ quản lý và theo dõi quá trình học tập của học viên tại trung tâm ngoại ngữ vietkids”. Với đề tài này, em mong muốn áp dụng những kiến thức được học trong trường cùng với việc tìm hiểu nghiên cứu ngôn ngữ và môi trường lập trình để xây dựng hệ thống đạt được hiệu quả hơn.

# CHƯƠNG 1.

# TỔNG QUAN VỀ ĐỀ TÀI

## 1.1. Tìm hiểu về các ngôn ngữ lập trình

### 1.1.1. Ngôn ngữ HTML

* **Định nghĩa**
* HTML là viết tắt của HyperText Markup Language, là một ngôn ngữ đánh dấu được sử dụng để tạo cấu trúc và hiển thị nội dung trên các trang web. Điều này bao gồm các phần tử như tiêu đề, đoạn văn, liên kết, hình ảnh và các thành phần tương tác khác.
* HTML không chỉ định cách nội dung được tổ chức mà còn mô tả cách nội dung đó sẽ được hiển thị và tương tác trên trình duyệt web. Bằng cách sử dụng các thẻ và thuộc tính, HTML cho phép nhà phát triển tạo ra các trang web có cấu trúc rõ ràng và dễ dàng để duyệt qua và tương tác.
* HTML mô tả cấu trúc của trang Web bằngp cách đánh dấu
* Các phần tử HTML là các khối để xây dựng các trang Web
* Các phần tử HTML được biểu thị bởi các thẻ tag
* Trình duyệt không hiển thị các thẻ HTML, nhưng dùng chúng để thông dịch nội dung của trang web
* **Cấu trúc của một tập tin HTML**

Một tập tin HTML thông thường có cấu trúc cơ bản như sau:

* **Khai báo DOCTYPE**: Đây là phần đầu tiên của tập tin HTML và nó cho biết phiên bản HTML mà trang web đang sử dụng. Ví dụ:

<!DOCTYPE html>

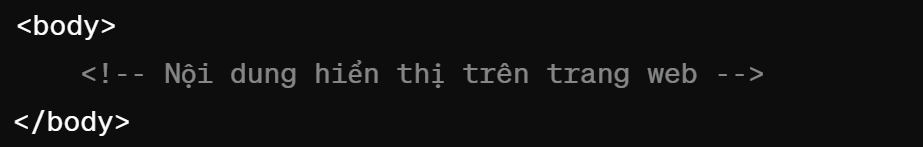
* **Thẻ <html>**: Đây là thẻ bao bọc toàn bộ nội dung của trang web và xác định ngôn ngữ của trang. Ví dụ:



* **Thẻ <head>**: Thẻ này chứa các thông tin không hiển thị trực tiếp trên trình duyệt như tiêu đề của trang, mô tả, từ khóa, và các liên kết tới các tập tin CSS và JavaScript. Ví dụ:



* **Thẻ <body>**: Phần này chứa toàn bộ nội dung mà người dùng sẽ thấy trên trình duyệt. Ví dụ:



* **Các phần tử HTML**: Bên trong thẻ **<body>**, đặt các phần tử như tiêu đề, đoạn văn, hình ảnh, liên kết, biểu đồ, bảng và các phần tử khác để tạo nên nội dung của trang web.
* **Kết thúc tập tin HTML**: Cuối cùng, bạn kết thúc tập tin HTML bằng cách đóng thẻ **<html>**.

Cấu trúc này giúp tập tin HTML trở nên có tổ chức, dễ đọc và dễ bảo trì.

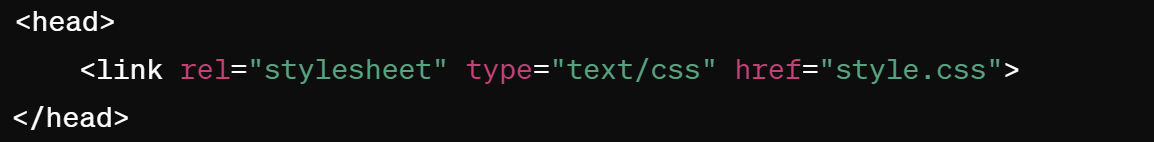
* **Ưu điểm của HTML**
* **Tính linh hoạt**: HTML cho phép tạo ra các trang web đa dạng với các phần tử như văn bản, hình ảnh, video, biểu đồ, biểu mẫu và nhiều loại phương tiện khác, giúp biểu diễn ý tưởng một cách sáng tạo và đa chiều.
* **Dễ học và sử dụng**: HTML có cú pháp đơn giản và dễ hiểu, làm cho việc học và sử dụng nó trở nên dễ dàng đối với người mới bắt đầu và cả những người có kinh nghiệm.
* **Tương thích rộng rãi**: HTML được hỗ trợ trên hầu hết các trình duyệt web và thiết bị di động hiện đại, đảm bảo rằng trang web có thể được truy cập một cách nhất quán trên nhiều nền tảng.
* **Tính tương tác**: HTML kết hợp với CSS và JavaScript để tạo ra các trang web có tính tương tác cao, giúp tăng cường trải nghiệm người dùng và tạo ra giao diện đáng chú ý và thu hút.
* **Tính mở rộng và tuỳ biến**: HTML cung cấp một cơ sở mạnh mẽ cho việc phát triển và mở rộng trang web với các tính năng và chức năng tùy chỉnh, cho phép tùy chỉnh trang web theo nhu cầu cụ thể của dự án.
* **Nhược điểm của HTML**
* **Phụ thuộc vào CSS và JavaScript**: HTML mặc định không có khả năng tự mình tạo ra giao diện hoặc tương tác phức tạp mà cần phải kết hợp với CSS và JavaScript. Điều này có thể tạo ra một mức độ phức tạp trong việc phát triển và bảo trì trang web.
* **Khả năng mở rộng hạn chế**: HTML có thể gặp hạn chế trong việc xử lý các yêu cầu phức tạp và tùy chỉnh cao hơn mà có thể đòi hỏi sự hỗ trợ từ các ngôn ngữ lập trình khác như PHP hoặc Python.
* **Không có khả năng tạo ra ứng dụng động**: HTML không thể tạo ra ứng dụng web động mà không cần phải tải lại trang. Đối với các yêu cầu phức tạp và tương tác, cần phải sử dụng JavaScript hoặc các framework JavaScript như React hoặc Angular.
* **Khó khăn trong việc quản lý trạng thái**: HTML không có khả năng quản lý trạng thái ứng dụng, điều này có thể gây ra khó khăn trong việc duy trì trạng thái của ứng dụng và tương tác giữa các thành phần trên trang web.
* **Bảo mật có thể yếu**: Mặc dù HTML không phải là nguyên nhân chính gây ra các lỗ hổng bảo mật, nhưng việc thiếu hiểu biết hoặc triển khai không đúng cách có thể dẫn đến các vấn đề bảo mật trên trang web.

### 1.1.2. Ngôn ngữ CSS

* ***Định nghĩa***
* CSS là viết tắt của Cascading Style Sheets, là một ngôn ngữ được sử dụng để định dạng và trang trí nội dung của trang web. Thay vì chỉ tập trung vào cấu trúc và nội dung như HTML, CSS tập trung vào cách mà các phần tử HTML sẽ được hiển thị trên trình duyệt.
* Bằng cách sử dụng các quy tắc và thuộc tính, CSS cho phép bạn tùy chỉnh màu sắc, font chữ, khoảng cách, độ rộng, chiều cao và nhiều thuộc tính khác của các phần tử HTML. Điều này giúp tạo ra giao diện trực quan và hấp dẫn cho người dùng.
* CSS cũng hỗ trợ tính năng kế thừa, cho phép bạn áp dụng các kiểu được định nghĩa cho một phần tử cha để áp dụng cho các phần tử con, giúp tối ưu hóa mã nguồn và quản lý kiểu dáng trang web một cách hiệu quả.
* **Cách sử dụng**
* Thêm CSS vào trang web: Có thể thêm CSS vào trang web bằng cách sử dụng các phần tử <style> hoặc bằng cách liên kết tới một tệp CSS bên ngoài từ trang HTML.
* Sử dụng phần tử **<style>**:



* Liên kết tới một tệp CSS bên ngoài:



* Viết CSS code: để tùy chỉnh giao diện và kiểu dáng của các phần tử trên trang web. CSS bao gồm các quy tắc, gọi là các quy tắc CSS, mà mỗi quy tắc bao gồm một bộ chọn (selector) và một tập hợp các thuộc tính và giá trị.

Ví dụ về một quy tắc CSS:



* **Xác định các bộ chọn (selectors)**: Bạn cần xác định các phần tử HTML mà bạn muốn áp dụng CSS cho chúng bằng cách sử dụng các bộ chọn (selectors). Các bộ chọn có thể là tên thẻ, class hoặc ID của phần tử.
* Bộ chọn tên thẻ:



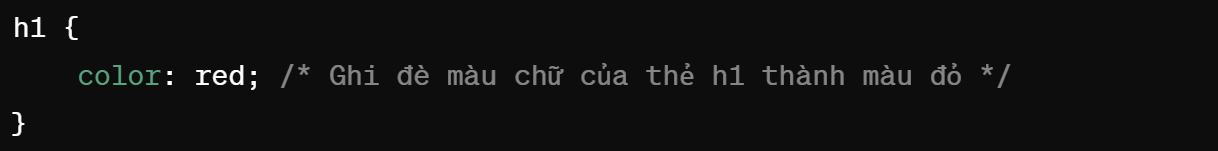
* Bộ chọn class:



* Bộ chọn ID:



* **Ghi đè các thuộc tính CSS**: Bạn có thể ghi đè các thuộc tính CSS của các phần tử bằng cách sử dụng các bộ chọn chính xác và quy tắc ưu tiên của CSS. Ví dụ:



* **Kiểm tra và chỉnh sửa**: Sau khi viết CSS, bạn cần kiểm tra trang web của mình trên nhiều trình duyệt và thiết bị để đảm bảo rằng giao diện hiển thị đúng như mong đợi.

### 1.1.3. Ngôn ngữ Javascript

- JavaScript là một ngôn ngữ lập trình phía client đa mục đích, là trái tim của web hiện đại. Nó không chỉ là một công cụ mạnh mẽ để thêm tính năng động và tương tác cho trang web, mà còn là một nền tảng linh hoạt cho việc xây dựng các ứng dụng web phức tạp và đa dạng.

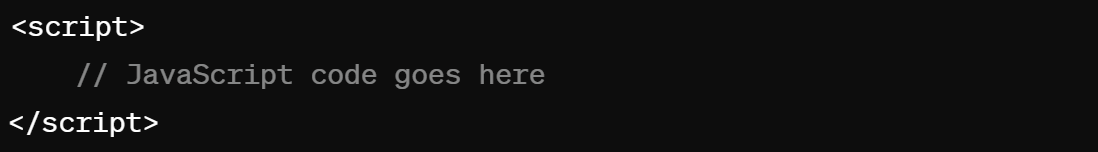
- Điểm mạnh của JavaScript nằm ở khả năng tương tác trực tiếp với các phần tử HTML và CSS của trang web, giúp tạo ra các trải nghiệm người dùng tương tác và phong phú. Tính linh hoạt của nó cho phép tạo ra các hiệu ứng động, biểu đồ tương tác, và các ứng dụng web đầy tính năng mà không gặp hạn chế lớn.

- JavaScript cũng được sử dụng rộng rãi trong việc phát triển các ứng dụng di động và desktop thông qua các framework như React Native và Electron. Điều này mở ra cánh cửa cho việc xây dựng ứng dụng đa nền tảng và đa kênh một cách dễ dàng.

- Với JavaScript, có thể biến các ý tưởng trừu tượng thành hiện thực trên web. Khả năng của nó không chỉ làm cho trang web trở nên đẹp mắt và độc đáo, mà còn mở ra nhiều cơ hội mới cho sự phát triển và sáng tạo trong ngành công nghiệp công nghệ thông tin.

* **Cách sử dụng**
* **Thêm JavaScript vào trang web**: Bạn có thể thêm mã JavaScript trực tiếp vào trang web hoặc liên kết tới một tệp JavaScript bên ngoài.

*Sử dụng phần tử* ***<script>*** *để thêm mã JavaScript trực tiếp vào trang web*

**

*Liên kết tới một tệp JavaScript bên ngoài*



* **Viết mã JavaScript**: Bạn viết mã JavaScript để thực hiện các chức năng và tương tác trên trang web. JavaScript cung cấp các tính năng như biến, hàm, điều kiện, vòng lặp, và xử lý sự kiện để tạo ra các tính năng động và tương tác trên trang.

Ví dụ về một hàm JavaScript đơn giản:



* **Gọi các hàm JavaScript**: Bạn có thể gọi các hàm JavaScript từ các sự kiện trên trang web hoặc từ mã JavaScript khác.

Ví dụ:



* **Kiểm tra và sửa lỗi**: Sau khi viết mã JavaScript, bạn cần kiểm tra và sửa lỗi để đảm bảo rằng chúng hoạt động đúng như mong đợi trên trình duyệt.
* **Tối ưu hóa và tái sử dụng**: Tối ưu hóa mã JavaScript của bạn bằng cách sử dụng biến và hàm một cách có tổ chức, và tái sử dụng mã khi cần thiết để giảm thiểu mã lặp.

### 1.1.4. Ngôn ngữ PHP

* ***Định nghĩa***
* PHP là một ngôn ngữ lập trình phía máy chủ được sử dụng rộng rãi trong việc phát triển các trang web động và ứng dụng web. Với PHP, bạn có thể tạo ra các trang web tương tác và động bằng cách tương tác với cơ sở dữ liệu, xử lý dữ liệu biểu mẫu, tạo và quản lý phiên làm việc, và thực hiện nhiều chức năng máy chủ khác.
* Điểm mạnh của PHP là tính linh hoạt và dễ sử dụng. Nó tích hợp dễ dàng với HTML và có thể nhúng trực tiếp vào mã HTML, cho phép bạn tạo ra các trang web động một cách hiệu quả. PHP cũng hỗ trợ nhiều cơ chế lập trình như hàm, điều kiện, vòng lặp, và các thư viện mạnh mẽ để xử lý dữ liệu và tương tác với cơ sở dữ liệu.
* Với PHP, bạn có thể xây dựng các ứng dụng web đa dạng, từ các trang web cá nhân đơn giản đến các ứng dụng thương mại điện tử phức tạp. Nó cung cấp một cách tiếp cận mạnh mẽ để phát triển các dự án web và là một phần không thể thiếu trong ngành công nghiệp phát triển web ngày nay.
* ***Ưu điểm ngôn ngữ lập trình PHP***
* Đầu tiên, tính linh hoạt của nó là điều rất đáng chú ý. PHP có thể được sử dụng trên hầu hết các máy chủ web và tương thích tốt với hầu hết các hệ thống quản lý cơ sở dữ liệu, bao gồm MySQL, PostgreSQL và SQLite. Điều này giúp cho việc triển khai ứng dụng trở nên dễ dàng hơn.
* Một ưu điểm khác của PHP là tính dễ học và sử dụng. Với cú pháp giống C và Java, PHP dễ tiếp cận đối với người mới học lập trình và cũng thân thiện với những nhà phát triển có kinh nghiệm. Hơn nữa, PHP là một ngôn ngữ lập trình mã nguồn mở, điều này có nghĩa là bạn có thể tìm thấy nhiều tài liệu, tài nguyên và cộng đồng hỗ trợ trực tuyến để hỗ trợ trong quá trình phát triển.
* PHP cũng cung cấp khả năng tích hợp tốt với HTML, cho phép nhúng mã PHP trực tiếp vào mã HTML. Điều này giúp cho việc phát triển các trang web động trở nên dễ dàng và hiệu quả hơn. Bạn có thể tạo ra các trang web động một cách linh hoạt và mạnh mẽ, tương tác với người dùng và cơ sở dữ liệu một cách dễ dàng.
* ***Nhược điểm ngôn ngữ lập trình PHP***
* **Bảo mật**: Một trong những nhược điểm lớn của PHP là vấn đề về bảo mật. Nếu không được lập trình và cấu hình đúng cách, ứng dụng PHP có thể dễ bị tấn công bởi các lỗ hổng bảo mật như SQL injection, Cross-Site Scripting (XSS), và Cross-Site Request Forgery (CSRF).
* **Quản lý mã nguồn**: PHP thường không có cấu trúc mã nguồn tổ chức, đặc biệt là trong các dự án lớn. Điều này có thể dẫn đến mã nguồn rối rắm, khó bảo trì và mở cửa cho lỗi và lỗ hổng bảo mật.
* **Tốc độ xử lý**: Mặc dù PHP đã được cải thiện rất nhiều qua các phiên bản mới, nhưng nó vẫn thường chậm hơn so với một số ngôn ngữ lập trình khác như Node.js hoặc Go. Điều này đặc biệt quan trọng trong các ứng dụng đòi hỏi xử lý lượng truy cập lớn.
* **Phụ thuộc vào máy chủ**: PHP là một ngôn ngữ phía máy chủ, có nghĩa là nó phụ thuộc vào máy chủ web như Apache hoặc Nginx để chạy. Điều này có thể tạo ra sự phụ thuộc và hạn chế trong việc triển khai và mở rộng ứng dụng.
* ***Phần mềm lập trình PHP***

Netbeans, PHP Designer, Visual studio code, PHP Storm, Sublime Text…

## 1.2. Laravel framework

* **Laravel là gì ?**
* Laravel được phát triển bởi Taylor Otwell với phiên bản cho ra mắt đầu tiên là vào tháng 6 năm 2011. Đây là một framework có mã nguồn mở miễn phí cùng với cú pháp mạch lạc, rõ ràng. Laravel ra đời nhằm mục đích hỗ trợ cho các ứng dụng web, lấy cốt lõi dựa theo mô hình MVC (Model – View – Controller).
* **Ưu điểm khi dùng Laravel**

Laravel sở hữu rất nhiều mặt ưu điểm để thu hút các nhà lập trình chắc chắn không thể bỏ qua như:

* Nguồn tài nguyên tài liệu sẵn có vô cùng đa dạng và phong phú, thân thiện với các lập trình viên
* Tốc độ xử lý dữ liệu nhanh chóng: Laravel hỗ trợ tối đã trong việc thiết lập trang web hay các dự án lớn trong vòng thời gian ngắn hạn.
* Di chuyển Database dễ dàng : Laravel cho phép bạn duy trì cấu trúc cơ sở dữ liệu mà không cần phải tạo lại. Thay vì sử dụng SQL, bạn có thể dùng PHP để kiểm soát dữ liệu. Bên cạnh đó, bạ cũng có thể khôi phục được những thay đổi gần nhất của Database.
* Dễ sử dụng: Cho dù kiến thức PHP của bạn có hạn hẹp đi chăng nữa thì bạn vẫn có thể phát triển website một cách nhanh chóng.j
* Tính năng bảo mật cao: Laravel sử dụng PDO nhằm để chống lại sự tấn công SQL Injection, cộng với field token ẩn để chống lại tấn công kiểu CSRF. Điều này sẽ giúp cho người dùng có thể hoàn toàn tập trung vào việc phát triển sản phẩm.
* **Nhược điểm khi dùng Laravel**

Ngoài những ưu điểm nổi trội thì Framework này vẫn còn nhiều nhược điểm chưa được hoàn thiện cần phải tối ưu thêm. Có thể kể đến như sau:

* Thiếu sự liên kết giữa các phiên bản làm cho ứng dụng dễ bị phá vỡ hoặc gián đoạn khi cập nhật code
* Dung lượng nặng dẫn đến quá trình tải trang trở nên chậm chạp
* Không sở hữu tính năng thanh toán
* **Cấu trúc**
* **Thư mục App**: Thư mục này chứa toàn bộ mã nguồn của ứng dụng, bao gồm các model, controller, middleware và các thành phần khác của ứng dụng.
* **Thư mục Config**: Thư mục này chứa các file cấu hình của ứng dụng, cho phép bạn cấu hình các cài đặt như cơ sở dữ liệu, lưu trữ, và các cài đặt khác của ứng dụng.
* **Thư mục Database**: Thư mục này chứa các file migration và seeders để quản lý cơ sở dữ liệu của ứng dụng.
* **Thư mục Public**: Thư mục này chứa tất cả các file tĩnh như hình ảnh, file CSS và JavaScript, được trình duyệt truy cập trực tiếp.
* **Thư mục Resources**: Thư mục này chứa các file nguồn của ứng dụng như các template Blade, file CSS và JavaScript, và các file ngôn ngữ.
* **Thư mục Routes**: Thư mục này chứa các file route của ứng dụng, định nghĩa các URL và xử lý các request của người dùng.
* **Thư mục Storage**: Thư mục này chứa các file tạm thời, file log và các file cache của ứng dụng.
* **Thư mục Tests**: Thư mục này chứa các file kiểm thử đơn vị và kiểm thử tích hợp của ứng dụng.
* **Router trong Laravel**
* Router cho phép bạn định nghĩa các tuyến đường (routes) cho ứng dụng của mình. Các tuyến đường này liên kết một URL cụ thể với một hành động hoặc một tác vụ trong ứng dụng.
* HTTP Verbs: Router cho phép bạn xác định các loại yêu cầu HTTP mà tuyến đường đó sẽ xử lý, như GET, POST, PUT, DELETE, vv. Sử dụng các phương thức như get(), post(), put(), delete() để định nghĩa các tuyến đường tương ứng với các phương thức HTTP.
* Named Routes: Laravel cho phép bạn đặt tên cho các tuyến đường, điều này giúp dễ dàng tham chiếu đến chúng từ các phần khác của ứng dụng mà không cần phải sử dụng URL cụ thể.
* Route Parameters: Router cũng hỗ trợ các tham số đường dẫn (route parameters), cho phép bạn trích xuất các giá trị từ URL để sử dụng trong xử lý yêu cầu.
* Route Groups: Laravel cho phép bạn nhóm các tuyến đường vào các nhóm, điều này giúp quản lý các tuyến đường có liên quan và áp dụng các middleware hoặc các điều kiện chung cho tất cả các tuyến đường trong nhóm.
* Middleware: Router cũng cho phép bạn gán middleware cho các tuyến đường, cho phép bạn thực hiện các xử lý trước hoặc sau khi yêu cầu được xử lý.
* Route Caching: Laravel cung cấp tính năng cache các tuyến đường để cải thiện hiệu suất ứng dụng. Việc này giúp giảm thời gian cần thiết để xác định xem yêu cầu sẽ được xử lý bởi route nào
* **Controller trong Laravel**
* Controller trong Laravel là một phần của mô hình MVC (Model-View-Controller) được sử dụng để quản lý logic xử lý của ứng dụng. Controller là nơi mà bạn định nghĩa các phương thức để xử lý các yêu cầu HTTP từ người dùng và tương tác với dữ liệu từ các mô hình (models) và trả về các kết quả cho người dùng thông qua các view.
* **Định nghĩa Controller**: Bạn có thể tạo các Controller mới bằng cách sử dụng artisan, một công cụ dòng lệnh đi kèm với Laravel. Ví dụ, để tạo một Controller mới, bạn có thể chạy lệnh sau:

php artisan make:controller ExampleController

Điều này sẽ tạo ra một file ExampleController.php trong thư mục app/Http/Controllers.

* **Phương thức Controller**: Trong Controller, bạn định nghĩa các phương thức để xử lý các yêu cầu HTTP. Mỗi phương thức trong Controller thường tương ứng với một hành động cụ thể, như hiển thị một trang, xử lý dữ liệu biểu mẫu, hoặc thực hiện các thao tác CRUD (Create, Read, Update, Delete).
* **Dependency Injection**: Laravel hỗ trợ dependency injection, cho phép bạn tiện lợi truy cập vào các service hoặc các đối tượng khác thông qua constructor của Controller.
* **Middleware**: Bạn có thể áp dụng middleware cho các phương thức của Controller để thực hiện các xử lý trước hoặc sau khi phương thức được gọi. Middleware giúp kiểm soát quyền truy cập, xác thực, ghi log, vv.
* **Truy cập dữ liệu**: Trong Controller, bạn có thể truy cập dữ liệu từ các mô hình (models) thông qua các phương thức của chúng. Điều này cho phép bạn thực hiện các thao tác như lấy dữ liệu từ cơ sở dữ liệu, thêm mới, cập nhật hoặc xóa dữ liệu.
* **Trả về Response**: Controller có nhiệm vụ trả về các phản hồi cho người dùng thông qua các phương thức của Laravel như view() để trả về các view, redirect() để chuyển hướng, hoặc response() để tạo ra các phản hồi tùy chỉnh.
* **Model trong Laravel**

Model trong Laravel là một phần của mô hình MVC (Model-View-Controller) và đại diện cho dữ liệu của ứng dụng cũng như quản lý các thao tác liên quan đến dữ liệu. Model thường tương tác với cơ sở dữ liệu để thực hiện các thao tác như truy xuất, thêm mới, cập nhật và xóa dữ liệu.

* **Tương tác với Cơ sở dữ liệu**: Trong Model, bạn định nghĩa các thuộc tính và phương thức để tương tác với cơ sở dữ liệu. Laravel cung cấp cho bạn các phương thức và tính năng mạnh mẽ như Eloquent ORM để thực hiện các thao tác này một cách dễ dàng và linh hoạt.
* **Eloquent ORM**: Eloquent là một ORM (Object-Relational Mapping) được tích hợp sẵn trong Laravel. Nó cho phép bạn định nghĩa các mô hình và quan hệ giữa chúng bằng cách sử dụng các cú pháp đơn giản và tự nhiên, giúp tương tác với cơ sở dữ liệu trở nên dễ dàng hơn.
* **Validation**: Model trong Laravel cũng cho phép bạn định nghĩa các quy tắc xác thực (validation rules) để kiểm tra dữ liệu trước khi lưu vào cơ sở dữ liệu. Điều này giúp đảm bảo tính toàn vẹn của dữ liệu và ngăn chặn các dữ liệu không hợp lệ.
* **Relationships (Quan hệ):** Model trong Laravel cho phép bạn định nghĩa các quan hệ giữa các Model, như mối quan hệ một-nhiều, nhiều-nhiều, hoặc một-một. Điều này giúp bạn tạo ra các truy vấn phức tạp và linh hoạt để truy vấn dữ liệu.
* **Events và Observers**: Laravel cung cấp các sự kiện (events) và các trình quan sát (observers) cho Model, giúp bạn thực hiện các hành động như ghi log, gửi email, vv., dựa trên các hoạt động của Model.

## 1.3. Vuejs Framework

* **Vuejs là gì ?**
  + Vue.js là một framework JavaScript mã nguồn mở được sử dụng rộng rãi trong phát triển ứng dụng web đơn trang (Single Page Applications - SPAs) và các ứng dụng web tương tác. Được thiết kế với mục tiêu đơn giản, linh hoạt và dễ dàng tiếp cận, Vue.js cho phép các nhà phát triển xây dựng các giao diện người dùng động một cách hiệu quả và linh hoạt.
  + Điểm mạnh của Vue.js nằm ở việc nó kết hợp các tính năng mạnh mẽ của các framework khác như React và Angular, nhưng với một cấu trúc nhỏ gọn và dễ hiểu hơn. Vue.js sử dụng mô hình MVVM (Model-View-ViewModel) để quản lý dữ liệu và hiển thị, giúp cho việc tổ chức và duy trì mã nguồn trở nên dễ dàng hơn.
  + Ngoài ra, Vue.js cung cấp một loạt các tính năng như các directive, computed properties, watchers và lifecycle hooks để giúp bạn xây dựng các ứng dụng linh hoạt và mạnh mẽ. Đồng thời, Vue.js cũng có một cộng đồng lớn và mạnh mẽ, với nhiều tài liệu, ví dụ và plugin được phát triển sẵn để hỗ trợ việc phát triển ứng dụng của bạn.
* **Ưu điểm khi dùng Vuejs**
* **Cú pháp dễ hiểu**: Vue.js có cú pháp rõ ràng và dễ hiểu, giúp cho mã nguồn trở nên sáng sủa và dễ đọc. Điều này giúp giảm thiểu sự phức tạp trong việc hiểu mã nguồn và tăng tính rõ ràng, tránh được sự lặp lại không cần thiết.
* **Tổ chức mã nguồn**: Vue.js thúc đẩy việc tổ chức mã nguồn một cách logic và có hệ thống, nhờ vào các khái niệm như components, mixins và directives. Điều này giúp tránh được mã nguồn rối rắm và việc lặp lại không cần thiết.
* **Tính tái sử dụng**: Vue.js cho phép tái sử dụng các component và mixins, giúp giảm thiểu việc viết mã lặp lại. Điều này làm cho mã nguồn trở nên sáng sủa hơn và dễ bảo trì hơn.
* **Quản lý trạng thái dễ dàng**: Với Vuex - thư viện quản lý trạng thái của Vue.js, việc quản lý trạng thái của ứng dụng trở nên dễ dàng hơn. Điều này giúp tránh được việc lạm dụng các biến toàn cục và làm cho mã nguồn trở nên dễ bảo trì hơn.
* **Cộng đồng lớn và hỗ trợ mạnh mẽ**: Vue.js có một cộng đồng lớn và mạnh mẽ, với nhiều tài liệu, ví dụ và plugin được phát triển sẵn. Điều này giúp giảm thiểu thời gian tìm kiếm giải pháp cho các vấn đề cụ thể và giúp tránh được việc tự "phát minh lại bánh xe".
* **Nhược điểm khi dùng Vuejs**
* **Khả năng mở rộng**: Vue.js không có cấu trúc hoàn chỉnh và mạnh mẽ như Angular, điều này có thể gây khó khăn trong việc mở rộng ứng dụng khi chúng trở nên lớn và phức tạp.
* **Những vấn đề về hiệu suất**: Trong một số trường hợp, Vue.js có thể gặp phải những vấn đề về hiệu suất, đặc biệt là khi xử lý các ứng dụng lớn hoặc có nhiều dữ liệu.
* **Khó khăn trong việc tìm kiếm giải pháp khi gặp vấn đề**: Do Vue.js không có một hệ thống hỗ trợ quy mô lớn như Angular, việc tìm kiếm giải pháp khi gặp vấn đề có thể trở nên khó khăn hơn.
* **Thời gian học và chuyển đổi**: Dù Vue.js có cú pháp đơn giản và dễ hiểu hơn so với Angular, nhưng việc học và chuyển đổi từ các framework khác có thể tốn thời gian và công sức.
* **Phiên bản và hỗ trợ**: Mặc dù Vue.js đang phát triển mạnh mẽ, nhưng không thể tránh khỏi những thay đổi và sự thất vọng khi một số plugin hoặc thư viện không tương thích với phiên bản mới nhất.

## 1.4. Hệ quản trị cơ sở dữ liệu (MySQL)

* **MySQL là gì?**
* MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở phổ biến được sử dụng rộng rãi trong phát triển ứng dụng web và các hệ thống thông tin. Được phát triển bởi Oracle Corporation, MySQL cung cấp một cơ sở dữ liệu ổn định, đáng tin cậy và linh hoạt cho việc lưu trữ và quản lý dữ liệu.
* Điểm mạnh của MySQL là tính đa nhiệm và khả năng mở rộng. Nó có thể xử lý đồng thời hàng ngàn kết nối và triệu bản ghi, đảm bảo hiệu suất cao cho các ứng dụng web có lượng truy cập lớn. Ngoài ra, MySQL hỗ trợ nhiều tính năng như các loại dữ liệu phong phú, các chức năng truy vấn mạnh mẽ, và các tiện ích quản lý dữ liệu đa dạng.
* MySQL cũng được đánh giá cao về tính bảo mật, với khả năng kiểm soát truy cập cơ sở dữ liệu, mã hóa dữ liệu, và cơ chế phòng chống tấn công. Điều này giúp bảo vệ thông tin quan trọng của người dùng và tổ chức trước các mối đe dọa bảo mật.
* **Ưu điểm**
* **Tính ổn định và đáng tin cậy**: MySQL đã được sử dụng và phát triển từ nhiều năm nay, nên đã được kiểm chứng là một hệ thống ổn định và đáng tin cậy trong việc quản lý dữ liệu.
* **Hiệu suất tốt**: MySQL được tối ưu hóa để đáp ứng các yêu cầu về hiệu suất và xử lý dữ liệu một cách hiệu quả. Nó có khả năng xử lý các tải công việc lớn mà không gây ra sự gián đoạn.
* **Dễ sử dụng**: MySQL có một giao diện dễ sử dụng và cung cấp các công cụ quản lý dữ liệu đơn giản, giúp cho việc quản lý và tương tác với cơ sở dữ liệu trở nên dễ dàng hơn.
* **Hỗ trợ đa nền tảng**: MySQL có thể chạy trên nhiều hệ điều hành khác nhau, bao gồm Linux, Windows và macOS, giúp cho việc triển khai trên các môi trường khác nhau trở nên thuận tiện.
* **Tính linh hoạt và mở rộng**: MySQL cho phép bạn tạo ra các cơ sở dữ liệu linh hoạt và mở rộng theo nhu cầu của ứng dụng. Bạn có thể dễ dàng thêm mới các bảng, chỉnh sửa cấu trúc hoặc tối ưu hóa hiệu suất mà không gặp phải quá nhiều rắc rối.
* **Nhược điểm**
* **Hiệu suất và mở rộng**: Trong một số trường hợp, MySQL có thể gặp phải vấn đề về hiệu suất khi xử lý các tập dữ liệu lớn và có nhiều truy vấn cùng lúc. Điều này có thể gây ra sự giảm hiệu suất và tăng thời gian phản hồi của ứng dụng.
* **Quản lý khóa và transaction**: MySQL có thể gặp phải vấn đề về quản lý khóa và transaction trong các môi trường có nhiều truy cập cùng lúc. Điều này có thể dẫn đến các vấn đề như blocking và deadlocks, ảnh hưởng đến tính nhất quán và hiệu suất của hệ thống.
* **Cộng đồng và hỗ trợ**: Mặc dù MySQL có một cộng đồng lớn và phong phú, nhưng đôi khi việc tìm kiếm giải pháp cho các vấn đề cụ thể hoặc các hỗ trợ kỹ thuật có thể gặp khó khăn, đặc biệt là đối với các phiên bản mới và các tính năng tiên tiến.
* **Bảo mật và quản lý quyền truy cập**: MySQL cung cấp các tính năng bảo mật nhưng việc cấu hình và quản lý quyền truy cập có thể phức tạp và dễ gây ra lỗi nếu không được thực hiện đúng cách.
* **Tính dễ bảo trì và khả năng sao lưu**: Mặc dù MySQL có các công cụ hỗ trợ sao lưu và phục hồi dữ liệu, nhưng việc quản lý và bảo trì cơ sở dữ liệu có thể đòi hỏi kỹ năng và thời gian đáng kể.

# CHƯƠNG 2.

# KHẢO SÁT VÀ PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 2.1. Đặt vấn đề

Trong khi công nghệ thông tin đang phát triển mạnh mẽ, nó cũng đang góp phần vào sự thay đổi của cách chúng ta học và giảng dạy. Lĩnh vực giáo dục không ngoại lệ, các công cụ và phần mềm giúp học tập đang ngày càng phát triển và trở nên phổ biến hơn.

Việc thiết kế một website hỗ trợ học tiếng Anh là một việc quan trọng vì nó cung cấp cho người học một công cụ tiên tiến và dễ sử dụng để học tiếng Anh mọi lúc mọi nơi. Ngoài ra, website cũng cung cấp cho người giảng dạy một công cụ hỗ trợ giảng dạy tiên tiến và dễ sử dụng, giúp họ tăng hiệu quả giảng dạy và cải thiện trải nghiệm học tập của người học.

Tất cả trong tất cả, việc thiết kế một website hỗ trợ học tiếng Anh là một bước quan trọng để giúp cải thiện chất lượng giáo dục và giúp người học và người giảng dạy tận dụng đầy đủ các tiện ích của công nghệ thông tin.

Yêu cầu chức năng của website:

* Chức năng cho tác nhân Admin
* Quản lý thông tin người dùng: Hiển thị, Thêm , sửa , xóa, tìm kiếm thông tin người dùng.
* Quản lý thông tin vai trò, quyền hệ thống: Hiển thị, Thêm, sửa, xóa, tìm kiếm thông tin vai trò, quyền hệ thống.
* Chức năng đăng nhập.
* Xem lịch sử làm bài kiểm tra.
* Chức năng cho tác nhân giáo viên
* Quản lý thông tin cấp độ bài học: Hiển thị, thêm, sửa, xóa thông tin.
* Quản lý thông tin bài học: Hiển thị, thêm, sửa, xóa thông tin.
* Quản lý thông tin đề kiểm tra: Hiển thị, Thêm , sửa , xóa thông tin đề kiểm tra.
* Quản lý lịch sử kiểm tra của người dùng hệ thống.
* Đăng nhập.
* Đăng ký.
* Chức năng cho tác nhân người học
* Chức năng làm bài kiểm tra.
* Chức năng học.
* Xem lịch sử làm bài kiểm tra.
* Đăng nhập.
* Đăng ký.

## 2.2. Phân tích yêu cầu

### 2.2.1. Danh sách actor tác động vào hệ thống

Có 3 đối tượng sử dụng cơ bản là Admin, Teacher, Learner.

* Giáo viên: Thông qua Website giáo viên có thể quản lý các cấp độ bài học, quản lý các bài học, quản lý bài kiểm tra, quản lý người học ...
* Người học: Người học có thể là học sinh, sinh viên, người có nhu cầu học tiếng anh, thông qua website có thể học, làm các bài kiểm tra, xem lại lịch sử làm bài kiểm tra.
* Người quản lý: Thông qua website có thể quản lý người dùng, quản lý vai trò người dùng.

### 2.2.2. Mục đích

* Đáp ứng nhu cầu giảng dạy của giáo viên và việc học tiếng anh của học sinh, sinh viên người có nhu cầu học tiếng anh.
* Thúc đẩy việc giảng dạy và học online.
* Rút ngắn khoảng cách giữa người dạy và người học.
* Việc quản lý bài học, nội dung học, kiểm tra trở nên dễ dàng.

## 2.3. Xác định yêu cầu phi chức năng của hệ thống

* Giao diện đơn giản, từ ngữ không khó hiểu cho người sử dụng, các thẻ chức năng và nút bấm phải khác màu nền và màu chữ.
* Tương thích với mọi trình duyệt web.
* Tốc độ phản hồi sau mỗi thao tác không quá 20s.

## 2.4. Biểu đồ phân tích thiết kế hệ thống

### 2.4.1. Biểu đồ usecase

* ***Biểu đồ usecase tổng quát***

A diagram of a company

Description automatically generated with medium confidence

*Hình 2.1: Biểu đồ use case tổng quát*

* ***Biểu đồ phân rã chức năng***
* Phân rã cho chức năng “Quản lý thông tin đề kiểm tra”

A diagram of a diagram

Description automatically generated

*Hình 2.2: Phân rã Use Case quản lý thông tin đề kiểm tra*

* Phân rã cho chức năng “Quản lý thông tin cấp độ bài học”

A diagram of a diagram

Description automatically generated with medium confidence

*Hình 2.3: Phân rã Use Case thông tin cấp độ bài học*

* Phân rã cho chức năng “Quản lý thông tin bài học”

A diagram of a diagram

Description automatically generated with medium confidence

*Hình 2.4: Phân rã Use Case quản lý thông tin bài học*

* Phân rã cho chức năng “Quản lý thông tin người dùng”

A diagram of a diagram

Description automatically generated

*Hình 2.5: Phân rã Use Case quản lý thông tin người dùng*

* Phân rã cho chức năng “Quản lý thông tin vai trò và quyền”

A diagram of a diagram

Description automatically generated

*Hình 2.6: Phân rã Use Case quản lý thông tin vai trò và quyền*

* Phân rã cho chức năng “Làm bài kiểm tra”

A diagram of a diagram

Description automatically generated

*Hình 2.7: Phân rã Use Case làm bài kiểm tra*

### 2.4.2. Đặc tả use case

* Kịch bản cho Use Case “Đăng nhập”

|  |  |
| --- | --- |
| Tên use case | Đăng nhập |
| Tác nhân chính | Người dùng, admin |
| Tiền điều kiện | Có tài khoản để đăng nhập |
| Thất bại | Email hoặc Password không chính xác |
| Thành công | Đăng nhập thành công vào trang quản lý |
| Chuỗi sự kiện chính:   1. User truy cập trang đăng nhập. 2. Hệ thống hiển thị giao diện đăng nhập. 3. User nhập thông tin tài khoản và nhấn đăng nhập. 4. Hệ thống lấy thông tin tài khoản. 5. Hệ thống truy vấn CSDL. 6. Hệ thống kiểm tra thông tin tài khoản. 7. Hệ thống kiểm tra thông tin tài khoản người dùng có phải Admin hay không ?   Hệ thống chuyển tiếp đến trang quản lý | |
| Ngoại lệ:   * 1. Thông tin tài khoản và mật khẩu không chính xác   2. Hệ thống thông báo Email hoặc Password không chính xác.   Hệ thống chuyển tiếp đến trang chủ. | |

*Bảng 2.1: Kịch bản cho Use Case “Đăng nhập”*

* Kịch bản cho UseCase “Đăng ký”

|  |  |
| --- | --- |
| Tên use case | Đăng ký |
| Tác nhân chính | Teacher, Learner(Gọi tắt là User) |
| Tiền điều kiện |  |
| Thất bại | Thông báo thất bại, Nhập lại thông tin |
| Thành công | Lưu thông tin vào cơ sở dữ liệu |
| Chuỗi sự kiện chính:   1. Người dùng truy cập trang đăng ký. 2. Hệ thống hiển thị giao diện đăng ký. 3. Người dùng nhập thông tin đăng ký. 4. Hệ thống lấy thông tin đăng ký. 5. Hệ thống kiểm tra thông tin.   Hệ thống thông báo đăng ký thành công. | |
| Ngoại lệ:  6.1 Hệ thống thông báo đăng ký không thành công. | |

*Bảng 2.2: Kịch bản cho UseCase “Đăng ký”*

* + Kịch bản cho UseCase “ Thêm thông tin cấp độ bài học”

|  |  |
| --- | --- |
| Tên use case | Thêm thông tin cấp độ bài học |
| Tác nhân chính | Teacher |
| Tiền điều kiện | Teacher phải đăng nhập vào hệ thống |
| Thất bại | Thông báo thất bại, nhập lại. |
| Thành công | Thêm mới vào CSDL |
| Chuỗi sự kiện chính:   1. Teacher truy cập chức năng quản lý cấp độ bài học trên trang chủ quản trị. 2. Hệ thống hiện danh sách cấp độ đang có hiện tại. 3. Teacher bấm vào nút Thêm (Biểu tượng dấu +). 4. Hệ thống hiện form nhập thông tin. 5. Teacher nhập thông tin mới và nhấn xác nhận. 6. Hệ thống kiểm tra thông tin nhập. 7. Hệ thống lưu thông tin vào cơ sở dữ liệu. 8. Hệ thống thông báo thêm cấp độ bài học thành công.   Hệ thống ẩn form nhập thông tin. | |

*Bảng 2.3:Kịch bản cho UseCase “Thêm thông tin cấp độ bài học”*

* Kịch bản cho Use Case “Sửa thông tin cấp độ bài học”

|  |  |
| --- | --- |
| Tên use case | Sửa thông tin cấp độ bài học |
| Tác nhân chính | Teacher |
| Tiền điều kiện | Teacher phải đăng nhập vào hệ thống |
| Thất bại | Thông báo thất bại, nhập lại. |
| Thành công | Cập nhật thông tin vào trong cơ sở dữ liệu |
| Chuỗi sự kiện chính:   1. Teacher truy cập chức năng quản lý cấp độ bài học trên trang chủ quản trị. 2. Hệ thống hiện danh sách cấp độ đang có hiện tại. 3. Teacher chọn biểu tượng sửa ở cấp độ cần sửa. 4. Hệ thống hiện form thông tin. 5. Teacher nhập thông tin mới cần chỉnh sửa và nhấn xác nhận. 6. Hệ thống kiểm tra thông tin nhập. 7. Hệ thống lưu thông tin vào cơ sở dữ liệu.   Hệ thống thông báo sửa cấp độ bài học thành công. | |

B*ảng 2.4: Kịch bản cho Use Case “Sửa thông tin cấp độ bài học*”

* Kịch bản cho UseCase “Xóa cấp độ bài học, bài học, ...”

|  |  |
| --- | --- |
| Tên use case | Xóa cấp độ bài học, bài học, ... |
| Tác nhân chính | Teacher |
| Tiền điều kiện | Teacher phải đăng nhập vào hệ thống |
| Thất bại | Thông báo xóa thất bại |
| Thành công | Xóa khỏi cơ sở dữ liệu |
| Chuỗi sự kiện:   1. Teacher truy cập chức năng quản lý trên trang chủ quản lý. Hệ thống hiện danh sách đang có hiện tại. 2. Teacher bấm vào biểu tượng xóa tương ứng với danh mục cần xóa. Hệ thống hiện ra popup thông báo xác nhận xóa. 3. Teacher ấn OK. 4. Hệ thống đóng popup 5. Hệ thống xóa khỏi cơ sở dữ liệu. 6. Hệ thống thông báo xóa thành công. | |
| Ngoại lệ:  5.1 Teacher ấn Cancel.  5.1.2 Hệ thống đóng popup xác nhận xóa. | |

*Bảng 2.5: Kịch bản cho UseCase “*Xóa cấp độ bài học, bài học, ...”

* Kịch bản cho UseCase “Thêm thông tin người dùng, thêm vai trò”

|  |  |
| --- | --- |
| Tên use case | Thêm thông tin người dùng, thêm vai trò |
| Tác nhân chính | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu | Trở lại trang danh sách |
| Thành công | Thêm vào cơ sở dữ liệu |
| Chuỗi sự kiện chính:   1. Admin truy cập vào chức năng quản lý trên trang chủ quản trị. 2. Hệ thống hiện danh sách đang có hiện tại. 3. Admin bấm vào nút thêm mới thông tin 4. Hệ thống chuyển đến trang nhập thông tin. 5. Admin nhập thông tin mới và nhấn xác nhận. 6. Hệ thống kiểm tra thông tin. 7. Hệ thống nhập thông tin vào cơ sở dữ liệu. 8. Hệ thống thông báo thêm thông tin thành công. 9. Hệ thống quay trở lại trang danh sách | |

*Bảng 2.6:Kịch bản cho UseCase “Thêm thông tin người dùng, thêm vai trò”*

* Kịch bản cho UseCase “Sửa thông tin người dùng, sửa vai trò”

|  |  |
| --- | --- |
| Tên use case | Sửa thông tin người dùng, sửa vai trò |
| Tác nhân chính | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Đảm bảo tối thiểu | Trở lại trang danh sách |
| Thành công | Thêm vào cơ sở dữ liệu |
| Chuỗi sự kiện chính:   1. Admin truy cập vào chức năng quản lý trên trang chủ quản trị. 2. Hệ thống hiện danh sách đang có hiện tại. 3. Admin bấm vào nút biểu tượng sửa tại thông tin cần sửa. 4. Hệ thống chuyển đến trang nhập thông tin cần sửa. 5. Admin nhập thông tin mới và nhấn xác nhận. 6. Hệ thống cập nhật thông tin vào cơ sở dữ liệu. 7. Hệ thống thông báo cập nhật thông tin thành công. | |

*Bảng 2.7:Kịch bản cho Use Case “Sửa thông tin người dùng, thêm vai trò”*

* Kịch bản cho UseCase “Tìm kiếm thông tin người dùng, thông tin vai trò”

|  |  |
| --- | --- |
| Tên use case | Tìm kiếm thông tin người dùng, thông tin vai trò |
| Tác nhân chính | Admin |
| Tiền điều kiện | Admin phải đăng nhập vào hệ thống |
| Thất bại | Không có dữ liệu |
| Thành công | Hiển thị danh sách thông tin |
| Chuỗi sự kiện chính:   1. Admin truy cập vào chức năng quản lý trên trang quản trị. 2. Hệ thống hiện danh sách đang có hiện tại. 3. Admin bấm vào ô input tìm kiếm và nhập thông tin tìm kiếm. 4. Hệ thống kiểm tra thông tin Admin vừa nhập đối chiếu với dữ liệu đang có.   Nếu có dữ liệu hệ thống hiển thị danh sách thông tin gần giống với dữ liệu Teacher nhập | |
| Ngoại lệ:  5.1 Nếu không có dữ liệu hệ thống hiển thị không tìm thấy dữ liệu nhập | |

*Bảng 2.8: Kịch bản cho UseCase “Tìm kiếm”*

* Kịch bản cho Use Case “Học”

|  |  |
| --- | --- |
| Tên use case | Học |
| Tác nhân chính | Learner |
| Tiền điều kiện | Learner đã đăng nhập, learner đang ở giao diện trang chủ. |
| Đảm bảo tối thiểu | Hiển thị nội dung bài học |
| Thành công | Hiển thị kết quả bài học |
| Chuỗi sự kiện chính:   1. Learner chọn trình độ và bấm Start. 2. Hệ thống hiển thị form danh sách bài học thuộc cấp độ. 3. Learner chọn bài học muốn học. | |
| Ngoại lệ:   * 1. Không có bài học nào thuộc cấp độ.   Hệ thống hiển thị “Không có bài học nào”. | |

*Bảng 2.9: Kịch bản cho UseCase “Học”*

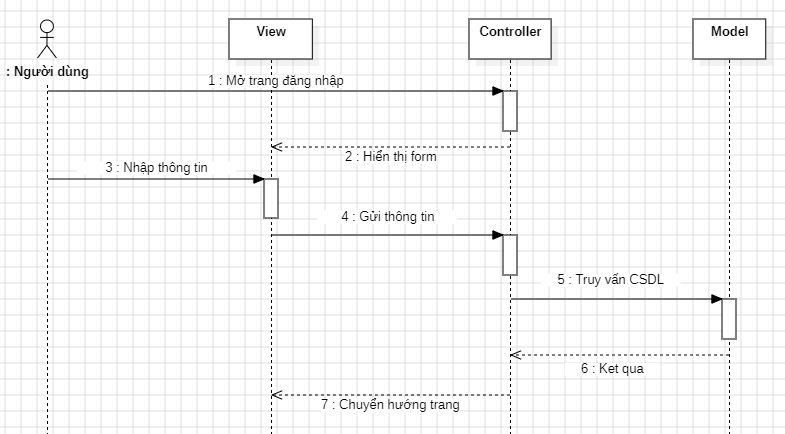
* Kịch bản cho UseCase “Làm bài kiểm tra”

|  |  |
| --- | --- |
| Tên use case | Làm bài kiểm tra |
| Tác nhân chính | Learner |
| Tiền điều kiện | User đã đăng nhập, learner đang ở giao diện trang chủ. |
| Đảm bảo tối thiểu | Hiển thị nội dung bài kiểm tra |
| Thành công | Lưu dữ liệu kết quả bài kiểm tra vào cơ sở dữ liệu |
| Chuỗi sự kiện chính:   1. Learner bấm vào Exam. 2. Hệ thống chuyển sang trang hiển thị danh sách cấp độ bài học. 3. Learner chọn biểu tượng loại kiểm tra tương ứng. 4. Hệ thống lấy thông tin và hiển thị nội dung bài kiểm tra. 5. Learner thực hiện trả lời các câu hỏi có trong nội dung bài kiểm tra 6. Learner nhấn nộp bài. 7. Hệ thống hiển thị kết quả bài kiểm tra và lưu kết quả bài kiểm. | |
| Ngoại lệ:   1. Learner không nhấn nộp bài.   Sau từng kĩ năng hết thời gian, hệ thống tự động nộp bài | |

*Bảng 2.10: Kịch bản cho UseCase “Làm bài kiểm tra”*

### Biểu đồ trình tự

* *Đăng nhập*



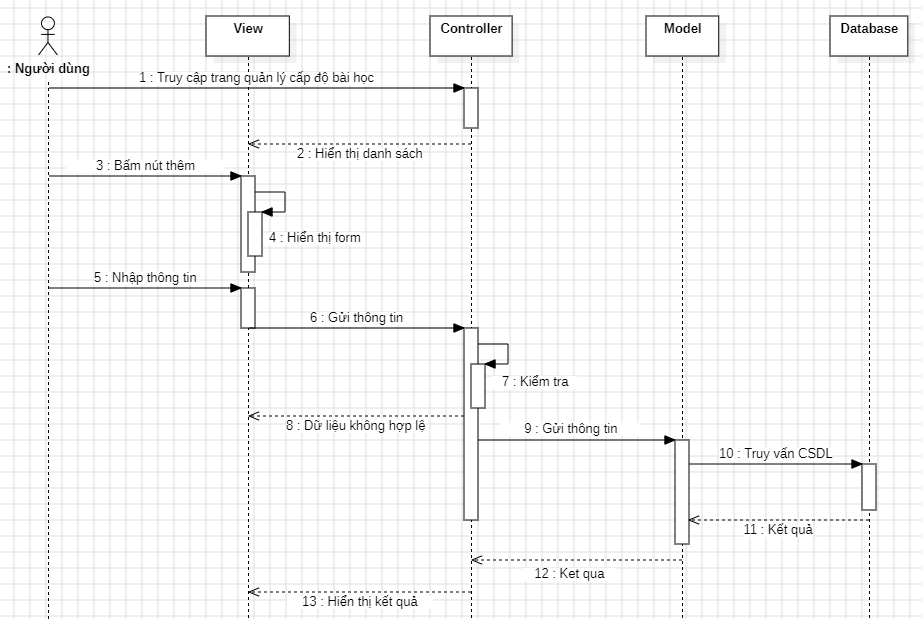
*Hình 2.8: Biểu đồ trình tự cho chức năng “đăng nhập”*

* *Đăng ký tài khoản*



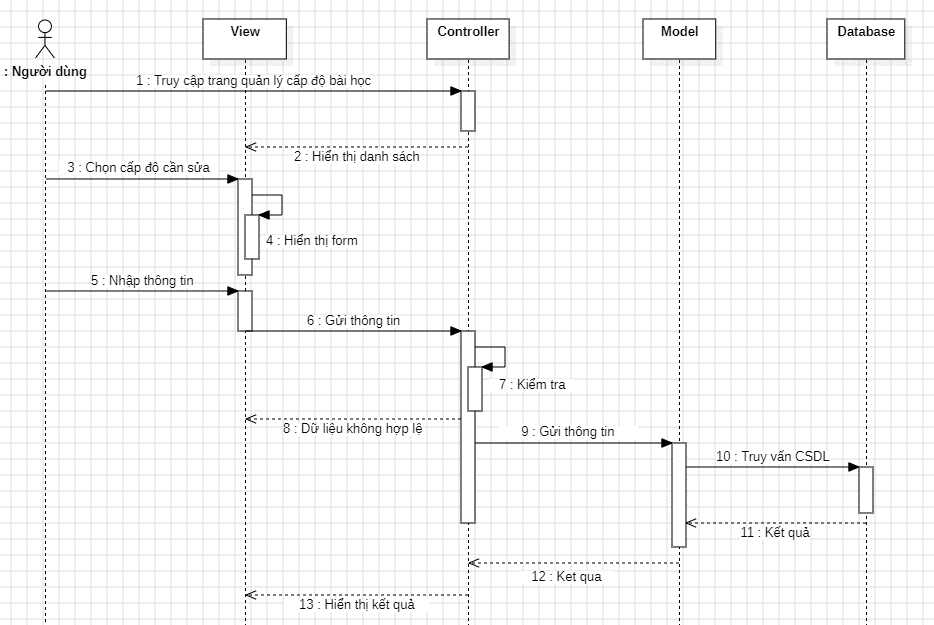
*Hình 2.9: Biểu đồ trình tự cho chức năng “đăng ký tài khoản”*

* *Thêm thông tin cấp độ bài học*



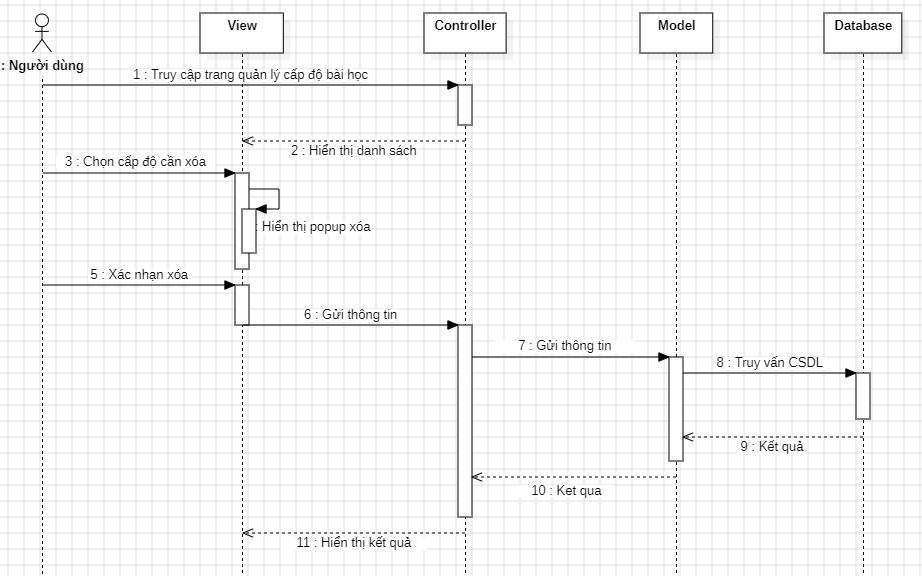
*Hình 2.10: Biểu đồ trình tự cho chức năng “Thêm thông tin cấp độ bài học”*

* *Sửa thông tin cấp độ bài học*



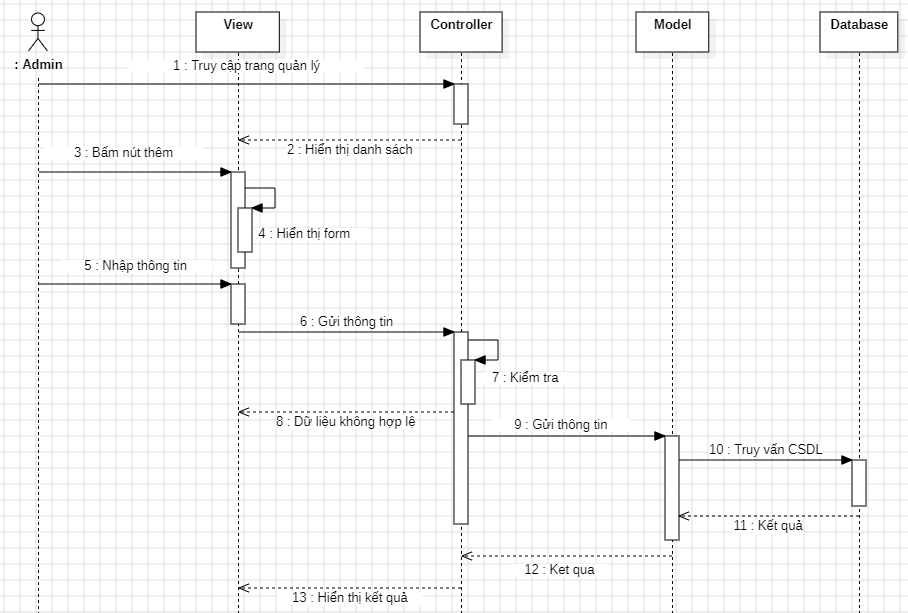
*Hình 2.11: Biểu đồ trình tự cho chức năng “Sửa thông tin cấp độ bài học*

* *Xóa cấp độ bài học, bài học, ...*



*Hình 2.12: Biểu đồ trình tự cho chức năng “Xóa cấp độ bài học, bài học, ...”*

* *Thêm thông tin người dùng, thêm vai trò*

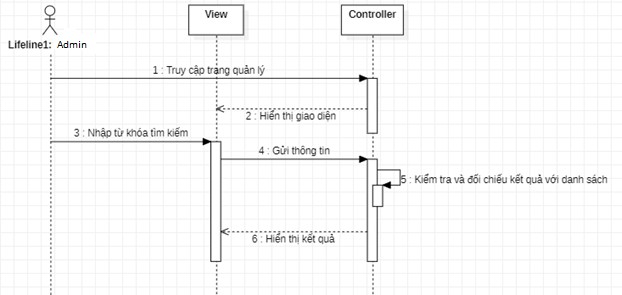


*Hình 2.13: Biểu đồ trình tự cho chức năng “Thêm thông tin người dùng, thêm vai trò”*

* *Sửa thông tin người dùng, sửa vai trò*

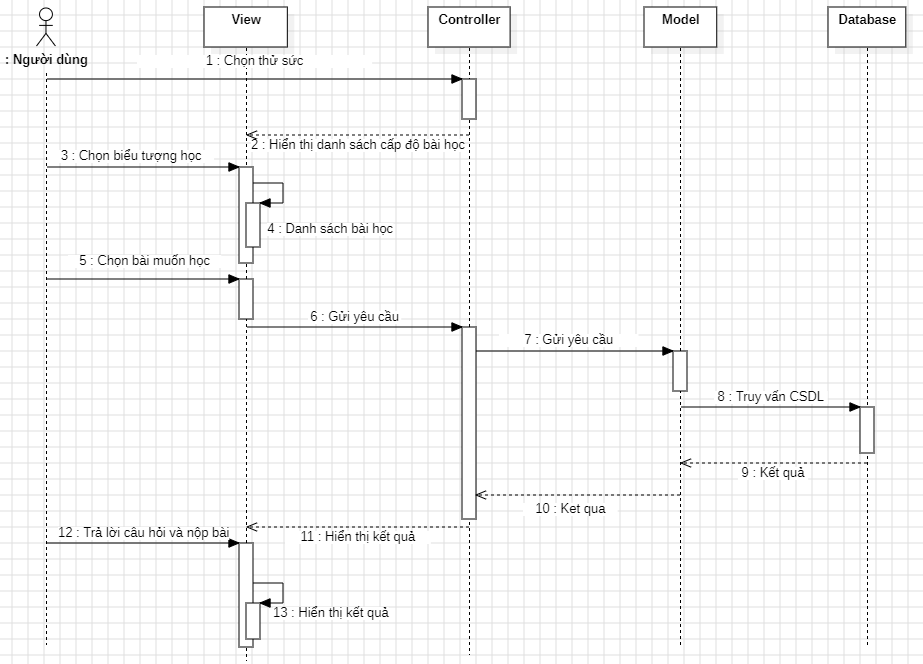


*Hình 2.14: Biểu đồ trình tự cho chức năng “Sửa thông tin người dùng, sửa vai trò”*

* *Tìm kiếm thông tin người dùng, thông tin vai trò*

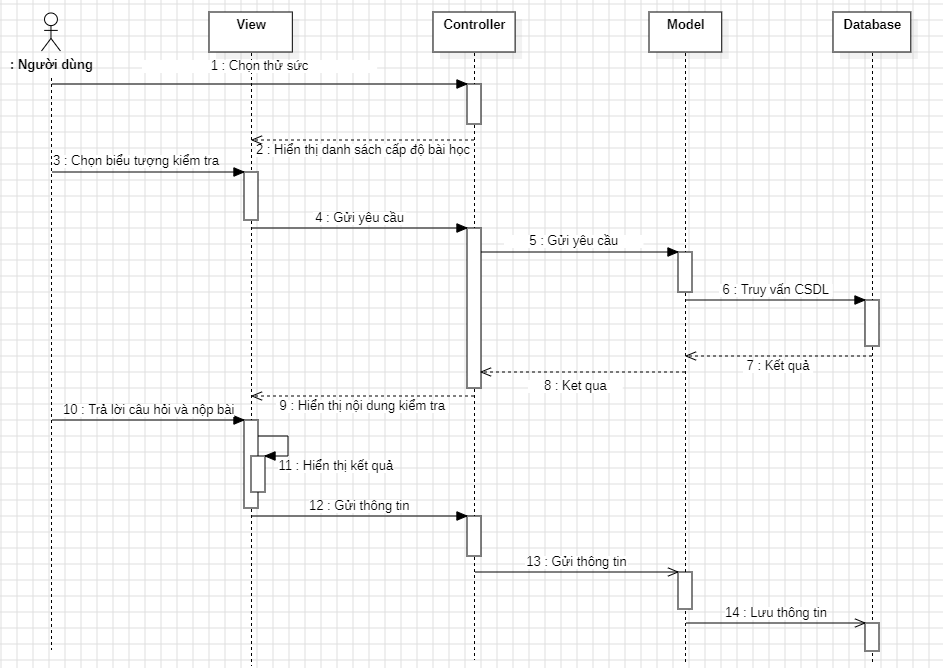
*Hình 2.15: Biểu đồ trình tự cho chức năng “Tìm kiếm thông tin người dùng, thông tin vai trò”*

* *Học*



*Hình 2.16: Biểu đồ trình tự cho chức năng “Học”*

* *Làm bài kiểm tra*



*Hình 2.17: Biểu đồ trình tự cho chức năng “Làm bài kiểm tra”*

### 2.4.4. Biểu đồ hoạt động

* + - * *Đăng nhập*

A diagram of a system

Description automatically generated

*Hình 2.18: Biểu hoạt động cho chức năng “đăng nhập”*

* *Đăng ký*

A diagram of a system

Description automatically generated

*Hình 2.19: Biểu đồ hoạt động cho chức năng “đăng ký”*

* + - * *Thêm thông tin cấp độ bài học*

A diagram of a system

Description automatically generated

*Hình 2.20: Biểu đồ hoạt động cho chức năng “Thêm thông tin cấp độ bài học”*

* + - * *Sửa thông tin cấp độ bài học*

A diagram of a system

Description automatically generated

*Hình 2.21: Biểu đồ hoạt động cho chức năng “Sửa thông tin cấp độ bài học”*

* + - * *Xóa cấp độ bài học, bài học, .*..

A diagram of a system

Description automatically generated

*Hình 2.22: Biểu đồ hoạt động cho chức năng “Xóa cấp độ bài học, bài học, ..*

* + - * *Thêm thông tin người dùng, thêm vai trò*

A diagram of a system

Description automatically generated

*Hình 2.24: Biểu đồ hoạt động cho chức năng “Thêm thông tin người dùng, thêm vai trò”*

* + - * *Sửa thông tin người dùng, sửa vai trò*

A diagram of a system

Description automatically generated

*Hình 2.25: Biểu đồ hoạt động cho chức năng “Sửa thông tin người dùng, sửa vai trò”*

* + - * *Tìm kiếm thông tin người dùng, thông tin vai trò*

A screenshot of a computer

Description automatically generated*Hình 2.26: Biểu đồ hoạt động cho chức năng “Tìm kiếm thông tin người dùng, thông tin vai trò”*

* + - * A screenshot of a diagram

        Description automatically generated*Học*

*Hình 2.27: Biểu đồ hoạt động cho chức năng “Học”*

* + - * A diagram of a diagram

        Description automatically generated with medium confidence*Làm bài kiểm tra*

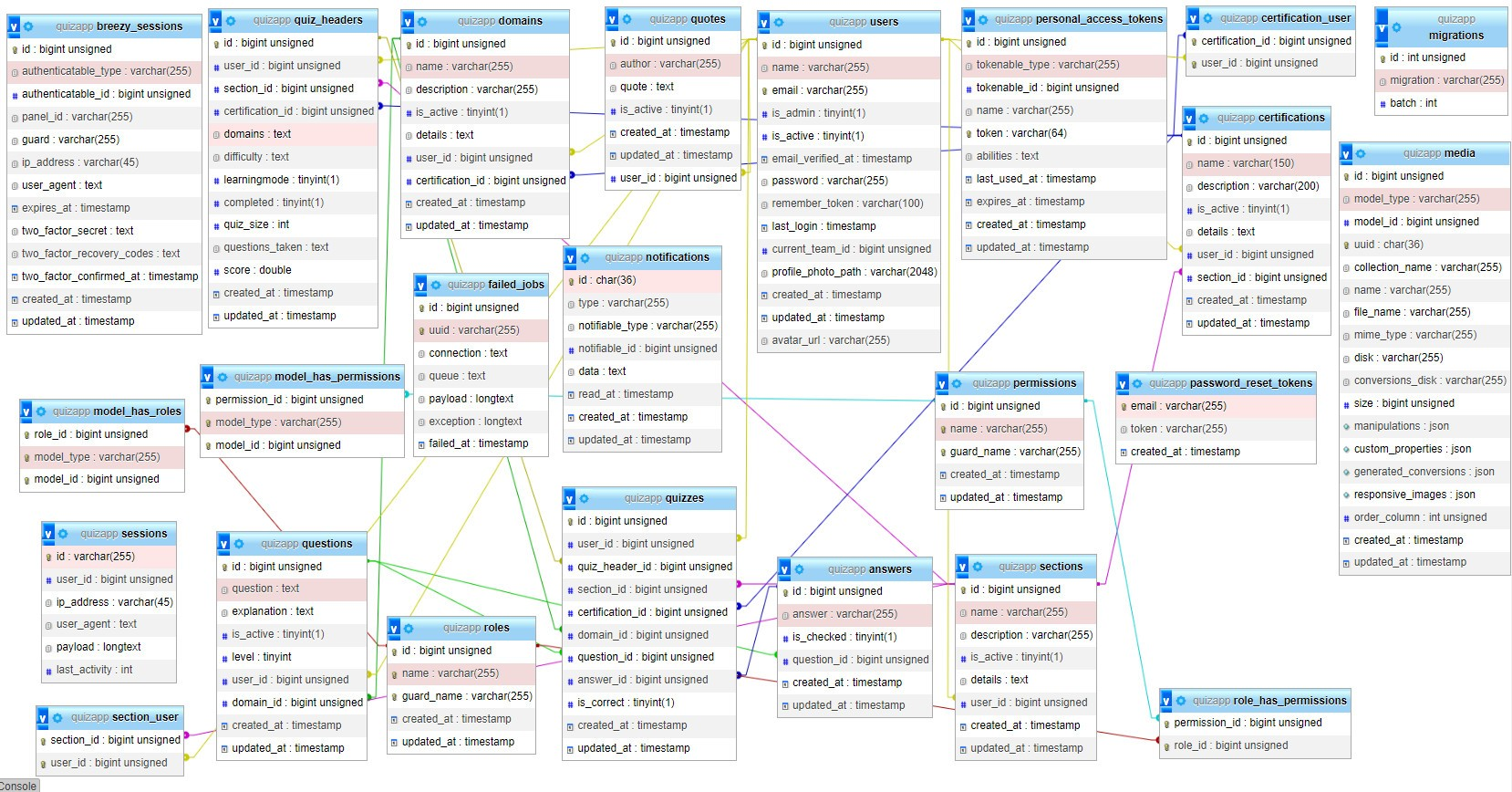
*Hình 2.28: Biểu đồ hoạt động cho chức năng “Làm bài kiểm tra”*

***2.4.5. Biểu đồ lớp***



*Hình 2.29: Biểu đồ lớp*

### 2.4.6. Thiết kế cơ sở dữ liệu

******

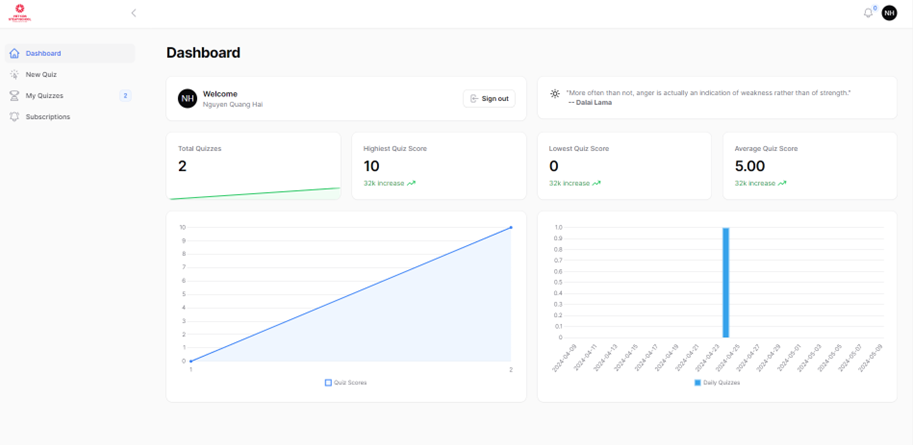
*Hình 2.30: Thiết kế cơ sở dữ liệu*

# CHƯƠNG 3.

# DEMO SẢN PHẨM WEBSITE

## Giao diện

### Giao diện người dùng



*Hình 3.1: Giao diện người dùng*

### Giao diện nội dung bài học

A screenshot of a computer

Description automatically generated

*Hình 3.1: Giao diện nội dung bài học*

### 3.1.3. Giao diện làm bài kiểm tra

A screenshot of a chat

Description automatically generated

*Hình 3.2: Giao diện làm bài kiểm tra.*

## Giao diện quản trị

### Giao diện đăng nhập

A screenshot of a login form

Description automatically generated

*Hình 3.3: Giao diện đăng nhập*

### Giao diện đăng ký

A screenshot of a login form

Description automatically generated

*Hình 3.4: Giao diện đăng ký*

### Giao diện trang quản trị

A screenshot of a computer

Description automatically generated

*Hình 3.5: Giao diện trang chủ quản trị*

### Quản lý cấp độ bài học

A screenshot of a computer

Description automatically generated

*Hình 3.6: Giao diện trang quản lý cấp độ bài học*

A screenshot of a computer

Description automatically generated

*Hình 3.7: Giao diện thêm, sửa cấp độ bài học*

### 3.2.5 Giao diện quản lý câu hỏi

A screenshot of a computer

Description automatically generated

*Hình 3.8: Giao diện quản lý câu hỏi*

* + 1. **Giao diện trang quản lý người dùng**

A screenshot of a computer

Description automatically generated

*Hình 3.9: Giao diện trang quản lý người dùng*

A screenshot of a computer

Description automatically generated

*Hình 3.10: Giao diện trang thêm, sửa thông tin người dùng*

### Giao diện trang quản lý vai trò

A white screen with many squares

Description automatically generated with medium confidence

*Hình 3.11: Giao diện trang quản lý vai trò*

A screenshot of a computer

Description automatically generated

*Hình 3.12: Giao diện trang thêm vai trò*

A screenshot of a computer

Description automatically generated

*Hình 3.13: Giao diện trang sửa vai trò*

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Dưới sự giúp đỡ tận tình của giáo viên hướng dẫn là ThS. Nguyễn Thị Dung, sự hỗ trợ nhiệt tình của các thầy/ cô giáo bộ môn, em đã hoàn thành bài báo cáo đồ án tốt nghiệp với đề tài “**Xây dựng website hỗ trợ quản lý và theo dõi quá trình học tập của học viên tại trung tâm ngoại ngữ Vietkids**”. Trong thời gian thực hiện đề tài, em đã đạt được một số kết quả như sau:

1. Hệ thống lại một số kiến thức liên quan đến phần mềm và hiểu sâu hơn về các ngôn ngữ.
2. Tìm hiểu được đặc điểm, cách sử dụng của frameword laravel và vuejs và ứng dụng vào xây dựng trang web hỗ trợ quản lý và theo dõi quá trình học tập của học viên tại trung tâm ngoại ngữ Vietkids.
3. Củng cố kiến thức và kỹ năng làm việc framework laravel và vuejs.

**Tuy nhiên đề tài thực tập của em vẫn còn nhiều điểm hạn chế:**

1. Chưa tìm hiểu và vận dụng được hết các tính năng của framework laravel và vuejs vào trong trang.
2. Việc thiết kế và tối ưu hóa code còn chưa được tốt.
3. Code chưa tối ưu

**Hướng phát triển:**

Tiếp tục nghiên cứu sâu hơn về framework laravel và vuejs, cố gắng áp dụng các tính năng của framework laravel và vuejs vào xây dựng website, hệ thống khác nhằm tích lũy kinh nghiệm cho công việc trong tương lai.

# TÀI LIỆU THAM KHẢO

1. Trần Đình Quế, Nguyễn Mạnh Sơn (2007). *Phân tích và thiết kế hệ thống thông tin*, Nhà xuất bản Hà Nội, Hà Nội.
2. Tô Oai Hùng (2018). *Lập trình web*, NXB Thông tin và truyền thông.
3. Joel Murach, Ray Harris (2013). *Lập trình cơ bản PHP và MySql*, NXB Khoa học và kỹ thuật
4. [https://www.w3school.com,](http://www.w3school.com/)
5. https://bizflycloud.vn/tin-tuc/html-la-gi-cac-kien-thuc-co-ban-nhat-ve-html-phai- biet-20181222101253656.htm
6. https://itnavi.com.vn/blog/php-la-gi/
7. [https://](https://code24h.com/gioi-thieu-ve-codeigniter-d33394.htm)https://laravel.com/
8. https://https://vuejs.org/

# NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………..…

………………………………………………………………………………..……

……………………………………………………………………………..………

…………………………………………………………………………..…………

……………………………………………………………………..………………

……………………………………………………………………..………………

……………………………………………………………..………………………

……………………………………………………………..………………………

……………………………………………………………..………………………

…………………………………………………………..…………………………

*Thái Nguyên, tháng 05 năm 2024*

GVHD