|  |  |
| --- | --- |
| **XÂY DỰNG HỆ THỐNG QUẢN LÝ CHẤM CÔNG CHO NHÂN VIÊN** | Học Viện Kĩ Thuật Quân Sự **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Khóa Luận**  **TỐT NGHIỆP ĐẠI HỌC**  **NGÀNH: CÔNG NGHỆ DỮ LIỆU**  **XÂY DỰNG HỆ THỐNG QUẢN LÝ CHẤM CÔNG CHO NHÂN VIÊN**  Sinh viên thực hiện: LƯƠNG VĂN CHUNG ĐỨC  NGUYỄN THẾ AN  Lớp CNDL 14  Giảng viên hướng dẫn: NGUYỄN QUỐC KHÁNH  ***Hà Nội, 12/2019*** |
| **Hà Nội**  **12/2019** |

|  |
| --- |
| TRƯỜNG HỌC VIỆN KĨ THUẬT QUÂN SỰ  **KHOA CÔNG NGHỆ THÔNG TIN \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  Khóa Luận  **TỐT NGHIỆP ĐẠI HỌC**  ***Đề tài:***  **XÂY DỰNG HỆ THỐNG QUẢN LÝ CHẤM CÔNG CHO NHÂN VIÊN**  Sinh viên thực hiện: LƯƠNG VĂN CHUNG ĐỨC  NGUYỄN THẾ AN  Lớp CNDL 14  Giảng viên hướng dẫn: NGUYỄN QUỐC KHÁNH  Cán bộ phản biện:  ***Hà Nội, 12/2019*** |

# **LỜI NÓI ĐẦU**

Những năm gần đây, Việc chấm công bằng vân tay đang được nhiều các công ty sử dụng để quản lý thời gian làm việc và tính lương cho nhân viên nhưng hiện tại thường thì các công ty chưa có phần mềm để quản lý chấm công mà chỉ lấy dữ liệu chấm công từ máy chấm công theo cách thủ công chính vì vậy chúng em xây dựng hệ thống quản lý thông tin chấm công của từng nhân viên trong công ty

Hệ thống sẽ được xây dựng trên nền Web để nhân viên có thể dễ dàng check thông tin của mình ở bất cứ nơi đầu có mạng internet, có rất nhiều cách hay công nghệ để xây dựng một website. Một trong những nền tảng rất được ưa chuộng hiện nay là Java Spring Boot.

Dựa vào những nền tảng trên, đề tài “**Hệ thống quản lý chấm công cho nhân viên**” của em sẽ được xây dựng bằng ngôn ngữ Java trên nền tảng Java Spring Boot.

Trong thời gian hoàn thành đồ án tương đối ngắn, nhưng với sự hướng dẫn của Nguyễn Quốc Khánh, em đã hoàn thành khóa luận trong thời gian cho phép. Em xin chân thành cảm ơn, em cảm ơn thầy đã luôn ủng hộ và hướng dẫn em, cho em những ý kiến, những lời khuyển để hoàn thành tốt đồ án. Bên cạnh đó em cũng cảm ơn cá thầy (cô) trong khoa đã truyền đạt những kiến thức nền tảng cho em để em hoàn thành được đồ án.

Trong quá trình làm đề tài, em vẫn còn những khúc mắc và thiếu sót trong đề tài. Em mong thầy có thể góp ý cho em có thể hoàn thiện đề tài hơn, để đưa sản phẩm của em ra với thực tiễn.

*Em xin trân thành cảm ơn!*

*Hà Nội, ngày 1 tháng 12 năm 2019*

*Sinh viên*

***Lương Văn Chung Đức***

***Nguyễn Thế An***

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 3](#_Toc6438117)

[**MỤC LỤC** 4](#_Toc6438118)

[**DANH MỤC HÌNH ẢNH SỬ DỤNG TRONG ĐỒ ÁN** 6](#_Toc6438119)

[**DANH MỤC BẢNG BIỂU SỬ DỤNG TRONG ĐỒ ÁN** 7](#_Toc6438120)

[**CÁC TỪ VIẾT TẮT SỬ DỤNG TRONG ĐỒ ÁN** 8](#_Toc6438121)

[**MỞ ĐẦU** 9](#_Toc6438122)

[**CHƯƠNG 1: KHẢO SÁT HỆ THỐNG VÀ TÌM HIỂU CÁC NỀN TẢNG CÔNG NGHỆ MỚI HIỆN NAY** 10](#_Toc6438123)

[**1.1.** **Khảo sát hệ thống** 10](#_Toc6438124)

[**1.1.1.** **Giới thiệu đề tài** 10](#_Toc6438125)

[**1.1.2. Lý do chọn đề tài** 10](#_Toc6438126)

[**1.1.3. Khảo sát yêu cầu người dùng** 10](#_Toc6438127)

[**1.2.** **Tìm hiểu các nền tảng công nghệ** 10](#_Toc6438128)

[**1.2.1.** **Nền tảng Java Spring Boot** 10](#_Toc6438129)

[**1.2.2.** **Xác thực người dùng với Json Web Token** 12](#_Toc6438130)

[**1.2.3.** **Bootstrap** 16](#_Toc6438131)

[**1.2.4.** **Angular** 16](#_Toc6438132)

[**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG** 19](#_Toc6438133)

[**1.1.** **Các tác nhân của hệ thống** 19](#_Toc6438134)

[**1.2.** **Các ca sử dụng của hệ thống** 19](#_Toc6438135)

[**1.3.** **Biểu đồ ca sử dụng** 20](#_Toc6438136)

[**1.4.** **Đặc tả ca sử dụng** 20](#_Toc6438137)

[**1.4.1.** **Đăng ký** 20](#_Toc6438138)

[**1.4.2.** **Đăng nhập** 21](#_Toc6438139)

[**1.4.3.** **Quản lý câu hỏi** 22](#_Toc6438140)

[**1.4.4.** **Quản lý đề thi** 23](#_Toc6438141)

[**1.4.5.** **Quản lý người dùng** 25](#_Toc6438142)

[**1.4.6.** **Quản lý môn** 26](#_Toc6438143)

[**1.4.7.** **Quản lý tin tức** 28](#_Toc6438144)

[**1.4.8.** **Quản lý slider bar** 29](#_Toc6438145)

[**1.5.** **Biểu đồ trình tự ca sử dụng** 31](#_Toc6438146)

[**1.5.1.** **Biểu đồ trình tự cho ca sử dụng “Đăng ký”** 31](#_Toc6438147)

[**1.5.2.** **Biểu đồ trình tự cho ca sử dụng “Đăng nhập”** 32](#_Toc6438148)

[**1.5.3.** **Biểu đồ trình tự cho ca sử dụng “Quản lý câu hỏi”** 33](#_Toc6438149)

[**1.5.4.** **Biểu đồ trình tự cho ca sử dụng “Quản lý đề thi”** 34](#_Toc6438150)

[**1.5.5.** **Biểu đồ trình tự cho ca sử dụng “Quản lý người dùng”** 35](#_Toc6438151)

[**1.5.6.** **Biểu đồ trình tự cho ca sử dụng “Quản lý môn”** 36](#_Toc6438152)

[**1.5.7.** **Biểu đồ trình tự cho ca sử dụng “Quản lý tin tức”** 37](#_Toc6438153)

[**1.5.8.** **Biểu đồ trình tự cho ca sử dụng “Quản lý slider bar”** 38](#_Toc6438154)

[**1.6.** **Biểu đồ lớp chi tiết** 39](#_Toc6438155)

[**1.7.** **Thiết kế cơ sở dữ liệu** 40](#_Toc6438156)

[**1.7.1.** **Chi tiết bảng cơ sở dữ liệu** 40](#_Toc6438157)

[**1.7.2.** **Mô hình dữ liệu liên kết vật lý** 52](#_Toc6438158)

[**CHƯƠNG 3: CÀI ĐẶT** 53](#_Toc6438159)

[**1.1.** **Cấu trúc Project** 53](#_Toc6438160)

[**1.2.** **Một số giao diện website sau khi cài đặt chương trình** 55](#_Toc6438161)

[**1.2.1.** **Giao diện trang user** 55](#_Toc6438162)

[**1.2.2.** **Giao diện trang admin** 57](#_Toc6438163)

[**KẾT LUẬN CHUNG** 64](#_Toc6438164)

[**TÀI LIỆU THAM KHẢO** 65](#_Toc6438165)

# **DANH MỤC HÌNH ẢNH SỬ DỤNG TRONG ĐỒ ÁN**

[Hình 1: Mô tả Spring Boot 10](#_Toc6438056)

[Hình 2: Nguyên tắc hoạt động của JWT [3] 14](#_Toc6438057)

[Hình 3. Biểu đồ usecase hệ thống của Website thi trắc nghiệm trực tuyến 19](#_Toc6438058)

[Hình 4: Biều đồ trình tự ca sử dụng đăng ký 30](#_Toc6438059)

[Hình 5: Biều đồ trình tự ca sử dụng đăng nhập 31](#_Toc6438060)

[Hình 6: Biểu đồ trình tự ca sử dụng quản lý câu hỏi 32](#_Toc6438061)

[Hình 7: Biểu đồ trình tự ca sử dụng quản lý đề thi 33](#_Toc6438062)

[Hình 8: Biểu đồ trình tự ca sử dụng quản lý người dùng 34](#_Toc6438063)

[Hình 9: Biểu đồ trình tự ca sử dụng quản lý môn 35](#_Toc6438064)

[Hình 10: Biểu đồ trình tự ca sử dụng quản lý tin tức 36](#_Toc6438065)

[Hình 11: Biểu đồ trình tự ca sử dụng quản lý slider bar 37](#_Toc6438066)

[Hình 12. Biểu đồ lớp chi tiết 38](#_Toc6438067)

[Hình 13. Biểu đồ mô hình dữ liệu liên kết vật lý 51](#_Toc6438068)

[Hình 14. Cấu trúc project backend 52](#_Toc6438069)

[Hình 15. Cấu trúc project frontend 53](#_Toc6438070)

[Hình 16: Giao diện trang chủ trang khách hàng 54](#_Toc6438071)

[Hình 17: Giao diện trang chủ trang đăng ký tài khoản 54](#_Toc6438072)

[Hình 18: Giao diện trang chủ trang đăng nhập 55](#_Toc6438073)

[Hình 19: Giao diện trang thi 55](#_Toc6438074)

[Hình 20: Giao diện trang tổng quan 56](#_Toc6438075)

[Hình 21: Giao diện trang quản lý người dùng 56](#_Toc6438076)

[Hình 22: Giao diện trang quản lý chapter 57](#_Toc6438077)

[Hình 23: Giao diện trang quản lý domain 57](#_Toc6438078)

[Hình 24: Giao diện trang quản lý exam 58](#_Toc6438079)

[Hình 25: Giao diện trang quản lý menu 58](#_Toc6438080)

[Hình 26: Giao diện trang quản lý permission 59](#_Toc6438081)

[Hình 27: Giao diện quản lý tin tức 59](#_Toc6438082)

[Hình 28: Giao diện quản lý câu hỏi 60](#_Toc6438083)

[Hình 29: Giao diện quản lý role 60](#_Toc6438084)

[Hình 30: Giao diện quản lý role menu 61](#_Toc6438085)

[Hình 31: Giao diện quản lý role permission 61](#_Toc6438086)

[Hình 32: Giao diện quản lý slide bar 62](#_Toc6438087)

[Hình 33: Giao diện quản lý môn học 62](#_Toc6438088)

# **DANH MỤC BẢNG BIỂU SỬ DỤNG TRONG ĐỒ ÁN**

[Bảng 1: Chi tiết bảng cơ sở dữ liệu Answer\_option 40](#_Toc6438089)

[Bảng 2: Chi tiết bảng cơ sở dữ liệu Chapter\_exam 40](#_Toc6438090)

[Bảng 3: Chi tiết bảng cơ sở dữ liệu Chapters 41](#_Toc6438091)

[Bảng 4: Chi tiết bảng cơ sở dữ liệu Customer 41](#_Toc6438092)

[Bảng 5: Chi tiết bảng cơ sở dữ liệu Domain\_exam 42](#_Toc6438093)

[Bảng 6: Chi tiết bảng cơ sở dữ liệu Domains 42](#_Toc6438094)

[Bảng 7: Chi tiết bảng cơ sở dữ liệu Exam\_answer 42](#_Toc6438095)

[Bảng 8: Chi tiết bảng cơ sở dữ liệu Exam\_group 43](#_Toc6438096)

[Bảng 9: Chi tiết bảng cơ sở dữ liệu Exam\_questions 43](#_Toc6438097)

[Bảng 10: Chi tiết bảng cơ sở dữ liệu Exam\_result 44](#_Toc6438098)

[Bảng 11: Chi tiết bảng cơ sở dữ liệu Exam\_setting 44](#_Toc6438099)

[Bảng 12: Chi tiết bảng cơ sở dữ liệu Exam\_user 45](#_Toc6438100)

[Bảng 13: Chi tiết bảng cơ sở dữ liệu Exams 45](#_Toc6438101)

[Bảng 14: Chi tiết bảng cơ sở dữ liệu Group 46](#_Toc6438102)

[Bảng 15: Chi tiết bảng cơ sở dữ liệu Menu 46](#_Toc6438103)

[Bảng 16: Chi tiết bảng cơ sở dữ liệu News 47](#_Toc6438104)

[Bảng 17: Chi tiết bảng cơ sở dữ liệu News\_tags 47](#_Toc6438105)

[Bảng 18: Chi tiết bảng cơ sở dữ liệu Permission 48](#_Toc6438106)

[Bảng 19: Chi tiết bảng cơ sở dữ liệu Questions 48](#_Toc6438107)

[Bảng 20: Chi tiết bảng cơ sở dữ liệu Role 49](#_Toc6438108)

[Bảng 21: Chi tiết bảng cơ sở dữ liệu Role\_menu 49](#_Toc6438109)

[Bảng 22: Chi tiết bảng cơ sở dữ liệu Role\_permission 49](#_Toc6438110)

[Bảng 23: Chi tiết bảng cơ sở dữ liệu Slide\_bar 49](#_Toc6438111)

[Bảng 24: Chi tiết bảng cơ sở dữ liệu Subject 50](#_Toc6438112)

[Bảng 25: Chi tiết bảng cơ sở dữ liệu Tags 50](#_Toc6438113)

[Bảng 26: Chi tiết bảng cơ sở dữ liệu User\_group 50](#_Toc6438114)

[Bảng 27: Chi tiết bảng cơ sở dữ liệu User 51](#_Toc6438115)

[Bảng 28: Chi tiết bảng cơ sở dữ liệu User\_role 51](#_Toc6438116)

# **CÁC TỪ VIẾT TẮT SỬ DỤNG TRONG ĐỒ ÁN**

|  |  |  |
| --- | --- | --- |
| **STT** | **Ký hiệu** | **Ý nghĩa** |
| 1 | API | **Application Programming Interface** (giao diện lập trình ứng dụng) |
| 2 | JSON | JavaScript Object Notation |
| 3 | JWT | Json Web Token |
| 4 | … | … |

# **MỞ ĐẦU**

Với mục đích xây dựng website thi trắc nghiệm trực tuyến thì trước hết đồ án cần thực hiện phân tích, thiết kế, xây dựng ứng dụng web thi trắc nghiệm, cho phép hiển thị, tìm kiếm, quản trị, thống kê,… các thông tin về câu hỏi, bài thi, đề thi, người dùng một cách khoa học, nhanh chóng và hiệu quả. Hệ thống website được phát tiền cần đáp ứng được các yêu cầu như sau:

* + - Phân quyền truy cập;
    - Quản trị các thông tin nông sản: thêm, sửa, xóa, tìm kiếm, hiển thị danh sách, phân trang...;
    - Quản trị các thông tin câu hỏi, đề thi, bài thi: thêm, sửa, xóa, hiển thị, tìm kiếm,…;
    - Giao diện thân thiện người dùng;

Để đạt được những mục đích và mục yêu cầu đặt ra như trên đồ án sẽ trình bày những vẫn đề trong ba chương như sau:

**Chương 1: Khảo sát hệ thống và tìm hiểu các nền tảng công nghệ mới hiện nay**

Chương 1 sẽ đi giới thiệu tổng quan về đề tài, lý do chọn đề tài, khảo sát yêu cầu người dùng và tìm hiểu về các nền tảng công nghệ mới hiện nay như Java Spring Boot, Angular 7, Json Web Token và Bootstrap 4.

**Chương 2: Phân tích và thiết kế hệ thống**

Chương 2 sẽ đi đặc tả về hệ thống website thi trắc nghiệm trực tuyến, thiết kế cơ sở dữ liệu.

**Chương 3: Cài đặt và kết quả đạt được**

Chương 3 sẽ trực tiếp đi xây dựng hệ thống website thi trắc nghiệm trực tuyến với những kiến thức tìm hiểu về công nghệ ở chương 1, và triển khai xây dựng hệ thống từ những phân tích và thiết kế ở chương 2 (bao gồm biểu đồ ca sử dụng, đặc tả ca sử dụng, biểu đồ trình tự ca sử dụng, biểu đồ lớp chi tiết, …).

# **CHƯƠNG 1: KHẢO SÁT HỆ THỐNG VÀ TÌM HIỂU CÁC NỀN TẢNG CÔNG NGHỆ MỚI HIỆN NAY**

## **Khảo sát hệ thống**

### **Giới thiệu đề tài**

Việc sử dụng hình thức thi trắc nghiệm trên giấy còn gặp nhiều khó khan như khó khăn trong việc quản lý, chấm thi, lãng phí nguồn nhân lực vì vậy em xây dựng website thi trắc nghiệm trực tuyến này để phục vụ cho việc giảm bớt việc lãng phí nguồn nhân lực, khó khăn trong vấn đề quản lý.

### **1.1.2. Lý do chọn đề tài**

Như chúng ta cũng thấy hiện nay việc thi trắc nghiệm trên giấy gặp rất nhiều khó khăn và phải sự nhiều nhân lực.

So với thi trắc nghiệm trên giấy thì thi trắc nghiệm trực tuyến chi phí thấp hơn, hiệu quả đạt cao hơn. Hơn thế nữa sẽ giảm bớt nguồn nhân lực.

Biết được những nhu cầu đó **Website thi trắc nghiệm trực tuyến** được xây dựng trên nền tảng **Java Spring Boot** nhằm đáp ứng cho người dùng và thông qua hệ thống Website này họ có thể thi trắc nghiệm trực tuyến.

Trước thực tế đó em đã chọn đề tài: “**Xây dựng Website thi trắc nghiệm trực tuyến**” cho đồ án tốt nghiệp của mình.

### **1.1.3. Khảo sát yêu cầu người dùng**

Website cho phép tài khoản có quyền cao nhất có thể thêm sửa xóa các loại tài khoản khác. Sau đó có thể thêm hoặc xóa quyền sử dụng của các tài khoản khác.

## **Tìm hiểu các nền tảng công nghệ**

### **Nền tảng Java Spring Boot**

1. **Giới thiệu**

Spring Boot là một module của Spring Framework, cung cấp tính năng RAD (Rapid Application Development) – Phát triển ứng dụng nhanh.

Spring Boot được dùng để tạo các ứng dụng độc lập dựa trên Spring.

Spring Boot không yêu cầu cấu hình XML.

Nó là một chuẩn cho cấu hình thiết kế phần mềm, tang cao năng suất cho developer.

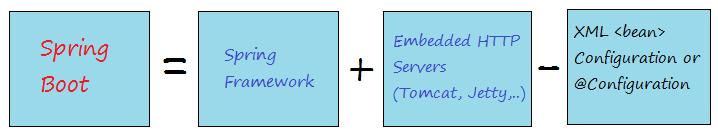
1. **Spring Boot là gì?**

Spring Boot là một bước tiếp theo của Spring, để làm cho Spring dễ dàng hơn trong việc thiết lập và phát triển ứng dụng. Với Spring Boot các cấu hình Spring được giảm thiểu tối đa. Spring Boot hỗ trợ các bộ chứa nhúng (embedded containers) điều này cho phép các ứng dụng web có thể chạy độc lập mà không cần phải triển khai lên các Web Server.

Dưới đây là các lợi ích của Spring Boot:

* + - Nó rất dễ dàng để phát triển các ứng dụng dựa trên Spring với Java hoặc Groovy.
    - Nó giảm thiểu thời gian phát triển và tang năng xuất.
    - Nó tránh việc phải viết nhiều mã nguyên mẫu (boilerplate Code), Annotations và các cấu hình XML.
    - Nó dễ dàng để bạn tương tác các ứng dụng Spring Boot với các hệ sinh thái của Spring như Spring JDB, Spring ORM, Spring Data, Spring Security ect.
    - Nó đi theo cách tiếp cận “Nguyên tắc cấu hình mặc định” để giảm thiểu thời gian và nỗ lực cần thiết để phát triển ứng dụng.
    - Nó cung cấp các Server nhúng (Embedded HTTP servers) như là Tomcat, Jetty… để phát triển ứng dụng.
    - Nó cung cấp công cụ CLI (Command Line Interfacce) để phát triển và test các ứng dụng Spring Boot (Java hoặc Groovy) từ các dòng lệnh (command prompt) rất dễ dàng và nhanh chóng.
    - Nó cung cấp rất nhiều các plugin để phát triển và test các ứng dụng Spring Boot nhanh chóng sử dụng các công cụ Build như Maven và Gradle.
    - Nó cung cấp nhiều plugin để làm việc với các cơ sở dữ liệu nhúng (embedded database) và các cơ sở dữ liệu lưu trữ trên bộ nhớ (in-memory database) một cách dễ dàng.

Spring Boot có thể giải thích một cách đơn giản bởi hình minh họa dưới đây:



Hình 1: Mô tả Spring Boot

1. **Restfull API**

REST (REpresenttational State Transfer) lần đầu tiên được giới thiệu vào năm 2000 trong luận văn tiến sĩ của Roy Thomas Fielding (đồng sáng lập giao thức HTTP). Trong luận văn ông giới thiệu khá chi tiết về các ràng buộc, quy ước cũng như cách thức thực hiện với hệ thống để có được một hệ thống REST. Hiểu một cách đơn giản, REST là một hệ thống các ràng buộc (constraints), chỉ cần đảm bảo những điều đó hệ thống của bạn có thể được gọi là RESTfull.

REST định nghĩa các quy tắc kiến trúc để bạn thiết kế Web services, chú trọng vào tài nguyên hệ thống, bao gồm các trạng thái tài nguyên được định dạng như thế nào và được truyền tải qua HTTP, và được viết bởi nhiều ngôn ngữ khác nhau.

REST có kiến trúc đơn giản, định rõ các ràng buộc nhằm tạo ra ứng dụng Web service đạt được những tính chất mong muốn về hiệu suất, khả năng mở rộng, khả năng điều chỉnh.

REST hướng tới việc xây dựng ứng dụng Web service có khả năng làm việc tốt nhất trên môi trường WWW.

Dữ liệu và các tính năng được coi như tài nguyên và được truy suất thông qua các URI (Uniform Resource Identifier)

REST sử dụng 4 phương thức chính của HTTP là POST, GET, PUT và DELETE để thực hiện các hành động CRUD đối với các tài nguyên

**Restfull API** là một tiêu chuẩn dùng trong việc thiết kế các thiết kế API. Các web service thường được áp dụng các cấu trúc REST vào để xây dựng.

Các cấu trúc cơ bản của cấu trúc REST:

* + - Sử dụng các phương thức HTTP một cách rõ rang.
    - Phi trạng thái.
    - Hiển thị cấu trúc thư mục như các URLs.
    - Truyền tải JavaScript Object Notation (JSON), XML hoặc cả hai.

### **Xác thực người dùng với Json Web Token**

1. **Json web token (jwt) là gì?**

JSON Web Mã (JWT) là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA.

Một ví dụ về JWT Token:

* eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJleHAiOjEzODY4OTkxMzEsImlzcyI6ImppcmE6MTU0ODk1OTUiLCJxc2giOiI4MDYzZmY0Y2ExZTQxZGY3YmM5MGM4YWI2ZDBmNjIwN2Q0OTFjZjZkYWQ3YzY2ZWE3OTdiNDYxNGI3MTkyMmU5IiwiaWF0IjoxMzg2ODk4OTUxfQ.uKqU9dTB6gKwG6jQCuXYAiMNdfNRw98Hw\_IWuA5MaMo

Cấu trúc JWT:

<base64-encoded header>.<base64-encoded payload>.<base64-encoded signature>

1. **Chi tiết về thành phần của JWT.**

* **Header**

Header bao gồm hai phần chính: loại token (mặc định là JWT - Thông tin này cho biết đây là một Token JWT) và thuật toán đã dùng để mã hóa (HMAC SHA256 - HS256 hoặc RSA).



* Payload

Payload chứa các claims. Claims là một các biểu thức về một thực thể (chẳng hạn user) và một số metadata phụ trợ. Có 3 loại claims thường gặp trong Payload: reserved, public và private claims.

Reserved claims: Đây là một số metadata được định nghĩa trước, trong đó một số metadata là bắt buộc, số còn lại nên tuân theo để JWT hợp lệ và đầy đủ thông tin: iss (issuer), iat (issued-at time) exp (expiration time), sub (subject), aud (audience), jti (Unique Identifier cho JWT, can be used to prevent the JWT from being replayed. This is helpful for a one time use token.) ... Ví dụ:

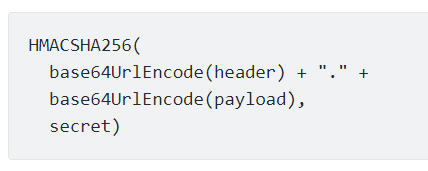


**Public Claims** - Claims được cộng đồng công nhận và sử dụng rộng rãi.

**Private Claims** - Claims tự định nghĩa (không được trùng với Reserved Claims và Public Claims), được tạo ra để chia sẻ thông tin giữa 2 parties đã thỏa thuận và thống nhất trước đó.

* **Signature**

Chữ ký Signature trong JWT là một chuỗi được mã hóa bởi header, payload cùng với một chuỗi bí mật theo nguyên tắc sau:



Do bản thân Signature đã bao gồm cả header và payload nên Signature có thể dùng để kiểm tra tính toàn vẹn của dữ liệu khi truyền tải.

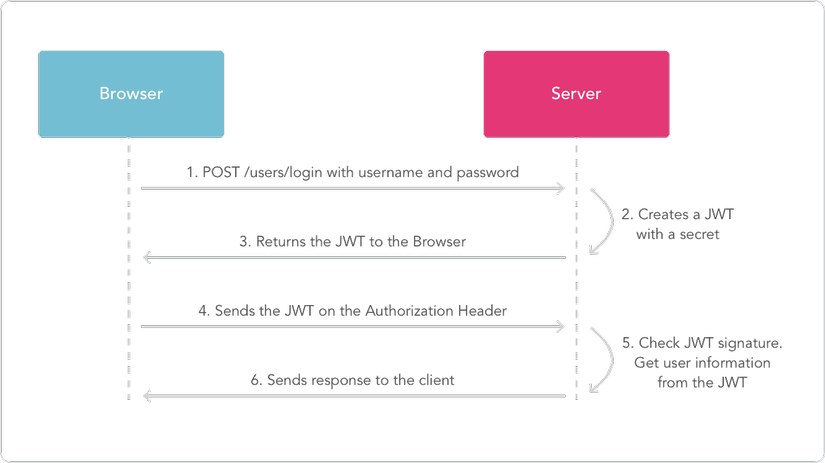
1. **Khi nào nên sử dụng JWT?**

Một trong những tình huống ứng dụng JWT thường gặp, đó là:

Authentication: Tình huống thường gặp nhất, khi user logged in, mỗi request tiếp đó đều kèm theo chuỗi token JWT, cho phép người dùng có thể truy cập đường dẫn, dịch vụ và tài nguyên được phép ứng với token đó. Single Sign On cũng là một chức năng có sử dụng JWT một cách rộng rãi, bởi vì chuỗi JWT có kích thước đủ nhỏ để đính kèm trong request và sử dụng ở nhiều hệ thống thuộc các domain khác nhau.

Information Exchange: JSON Web Token cũng là một cách hữu hiệu và bảo mật để trao đổi thông tin giữa nhiều ứng dụng, bởi vì JWT phải được ký (bằng cặp public / private key), bạn có thể chắc rằng người gửi chính là người mà họ nói rằng họ là (nói tóm tắt hơn là không hoặc khó để mạo danh bằng JWT), ngoài ra, chữ ký cũng được tính toán dựa trên nội dung của header và nội dung payload, nhờ đó, bạn có thể xác thực được nội dung là nguyên bản, chưa được chỉnh sửa hoặc can thiệp. Tuy nhiên, một lưu ý hết sức quan trọng là do cấu trúc của JWT đơn giản nên JWT có thể dễ dàng bị decode, do vậy, bạn không nên dùng JWT để transfer các thông tin nhạy cảm.

1. **Nguyên tắc hoạt động của JWT.**



Hình 2: Nguyên tắc hoạt động của JWT [3]

### **Bootstrap**

1. **Bootstrap là gì?**

Bootstrap là một framework cho phép thiết kế website reponsive nhanh hơn và dễ dàng hơn.  
 Bootstrap là bao gồm các HTML templates, CSS templates và Javascript tao ra những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm các plugin Javascript trong nó. Giúp cho việc thiết kế reponsive của bạn dễ dàng hơn và nhanh chóng hơn.  
Lịch sử Bootstrap  
Bootstrap là dược phát triển bởi Mark Otto và Jacob Thornton tại Twitter. Nó được xuất bản như là một mã nguồn mở vào tháng 8 năm 2011 trên GitHub. Tính ra đến thời điểm mình viết bài viết này nó cũng đã phát triển được 3 năm rồi. Bản bootstrap mới nhất bây giờ là bootstrap 3.

1. **Tại sao bạn nên sử dụng Bootstrap?**

Những điểm thuận lợi khi bản sử dụng bootstrap:

Rất dễ để sử dụng: Nó đơn giản vì nó được base trên HTML, CSS và Javascript chỉ cẩn có kiến thức cơ bản về 3 cái đó là có thể sử dụng bootstrap tốt.  
 Tính năng Responsive: Bootstrap’s xây dựng sẵn reponsive css trên các thiết bị phones, tablets, và desktops  
Mobile: Trong Bootstrap 3 mobile-first styles là một phần của core framework  
 Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, and Opera) nhưng lưu ý vì em IE vẫn rất hãm với IE phiên bản cũ vì thế việc IE9 hay IE8 đổ xuống không support là chuyện bình thường. Các bạn có dùng nên luy ý điểm này theo như mình dùng làm dự án thì nó support tuyệt vời trên IE10 đổ lên nhé, thấp nhất có lẽ là IE9 thôi còn IE8 thì không nên xài nhé.

### **Angular**

1. **Angular là gì?**

Angular là một framework Javascript do Google phát triển, dùng để xây dựng và phát triển các ứng dụng web (web app) hoặc cũng có thể xây dựng các native app cho mobile hoặc desktop.

1. **Tại sao chọn Angular?**

Angular giúp nâng cao năng suất của các lập trình viên:

* Việc phát triển **Web** đã có bước thay đổi đáng kể trong vài năm qua. Với phiên bản **ECMAScript (ES) 2015** - chúng ta quen thuộc với cái tên ES6, với những **class** hay **arrow function**. **Angular 2+** ứng dụng những tính năng mới này giúp việc **code** với **Angular** trở nên rõ ràng và dễ học hơn rất nhiều.
* Thêm vào đó, với việc ứng dụng **Typescript** - một ngôn ngữ - hay là một bản **nâng cấp** đáng giá của **Javascript**, **Angular** kết hợp với **Typescript**, chúng ta có một công cụ tuyệt vời giúp xử lý các vấn đề hạn chế của **JS** như kiểm tra kiểu dữ liệu, **refactor code** an toàn hơn,… từ đó cũng hỗ trợ tốt hơn cho việc **Debug** cũng như giúp các **Dev** thực sự hiểu rõ mã nguồn của họ hơn.

Cấu trúc phát triển rõ ràng:

* Điều quan trọng của một **Frameworks** đối với lập trình viên đó là cấu trúc phát triển ứng dụng của nó, và **Angular** mang đến một kiến trúc rất rõ ràng, dựa trên ba yếu tố chính: **class**, các **dependency**được thêm vào và mô hình MVVM (model-view-view/model).
* **Angular** sử dụng **class** trong ES6 với một loạt các thuộc tính để xây dựng toàn bộ các cấu trúc chủ chốt, giả sử bạn muốn tạo một **Angular component** - Tạo một **class** và thêm vào các thuộc tính cần thiết. Hay bạn muốn tạo một **Angular module** - Hãy tạo một **class**và thêm vào đó các thuộc tính cần thiết. Về cơ bản sẽ là như vậy, **Angular** cung cấp một cấu trúc rõ ràng để xây dựng từng tính năng cho ứng dụng của bạn.
* Các **dependency** mạnh mẽ được sử dụng trong ứng dụng khi cần thiết, và khi cần tích hợp bất kì **dependency** nào, như **HTTP** hay **Router**, chúng ta chỉ cần thêm nó vào bên trong **constructor** của **class**.

Hỗ trợ đầy đủ tính năng điều hướng:

* Đa số các ứng dụng **Web** không chỉ có 1 **view** hay một **page** duy nhất, mà sẽ cung cấp nhiều **view** khác nhau tương ứng với với các chức năng chính. Ví dụ như một trang **web** với các trang giới thiệu, trang nội dung, trang chi tiết, trang đăng nhập, đăng ký,... Chúng ta sẽ cần hiển thị đúng **view** vào đúng thời điểm. Đó là mục đích của điều hướng (**routing**). Và **Angular** cung cấp đầy đủ tính năng cho việc này, chúng ta định nghĩa các đường dẫn (**route**) cho mỗi **page view** của ứng dụng. Và chúng ta sẽ kích hoạt **route** dựa trên tương tác của người dùng (**user**).
* Chúng ta có thể truyền thêm dữ liệu vào các **route**, giúp **view** hiển thị nội dung một cách **dynamic**, có thể bảo vệ **route** để người dùng chỉ có thể truy cập sau khi đã đăng nhập hoặc có quyền truy cập, có thể ngăn chặn việc người dùng ngay lập tức rời một trang khi các thao tác còn dang dở cho đến khi họ thực sự xác nhận việc rời đi hoặc lưu lại tiến trình sử dụng,...**Angular**đồng thời cũng hỗ trợ **child-route** cho việc điều hướng bên trong một **route**. Việc điều hướng giữa các **view** bên trong ứng dụng **Angular** thực sự rất linh hoạt và mạnh mẽ.

Angular giúp giảm tối đa kích thước và tang tối đa hiệu suất của ứng dụng:

* Kích thước và hiệu năng có mối liên quan mật thiết khi chúng ta làm việc trên nền tảng **Web**. Một **component** nhỏ hơn sẽ giúp nâng cao hiệu suất khởi động - giảm cả thời gian **download** cũng như thời gian cũng như thời gian **compile** trên trình duyệt. Giảm kích thước **component** và giúp tăng hiệu suất là một ưu điểm cũng như mục tiêu mà **Angular** mong muốn mang đến cho các lập trình viên.
* Giảm kích thước ứng dụng có thể thực hiện bằng nhiều cách. Đầu tiên chúng ta có thể giảm tối đa kích thước của từng **component** tới mức tối thiểu có thể. Tiếp theo các **component** sẽ được sắp xếp bên trongtrong **Angular Module** bằng 1 cách đề cho các nhóm logic có liên quan đến nhau sẽ được **download** cùng với nhau. Và bước thứ ba, **lazy loading** bên trong các **route** sẽ chỉ **downloaad** những **module** cần thiết cho việc hiển thị nội dung cần thiết tới người dùng, và sẽ không bao giờ **download** những nội dung không cần thiết.
* Chúng ta có một trình biên dịch tên là **AOT**, trình biên dịch này sẽ chạy một lần trong thời gian **build** ứng dụng. Trình duyệt sau đó sẽ **download** phiên bản chưa được biên dịch của ứng dụng và **render**ứng dụng tới người dùng ngay lập tức mà không cần biên dịch nó lần đầu trong trình duyệt. Thêm nữa là sẽ koong cần **download** trình biên dịch **Angular**, giúp làm giảm đáng kể kích thước (**size**) của ứng dụng cần tải về.

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

## **Các tác nhân của hệ thống**

Hệ thống Quản lý chấm công cho nhân viên có 2 tác nhân chính trong hệ thống :

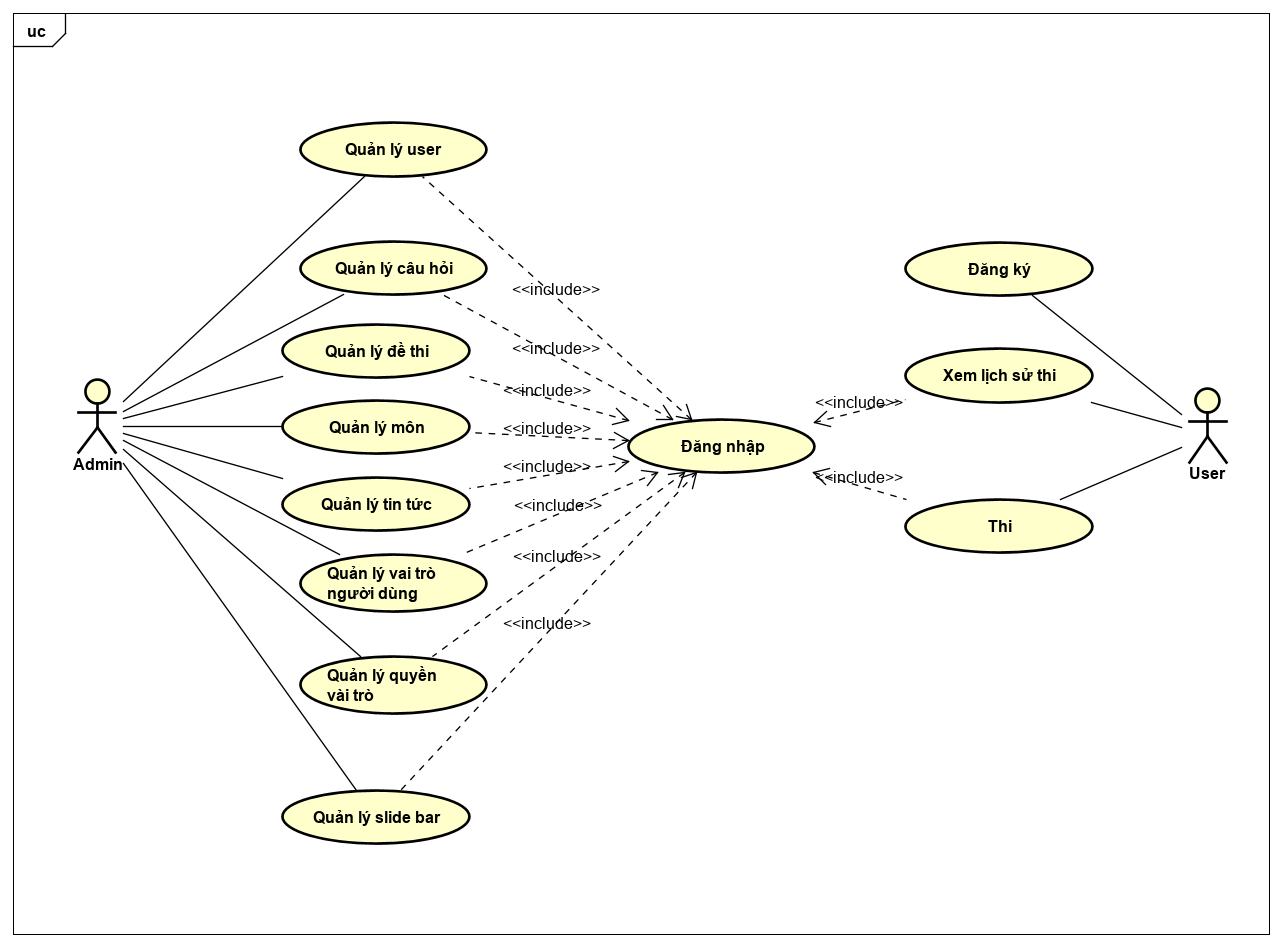
* Admin
* Member

## **Các ca sử dụng của hệ thống**

Hệ thống có các ca sử dụng ứng với mỗi tác nhân chính như sau:

* Người dùng
* Đăng ký
* Đăng nhập
* Xem lịch sử thi
* Thi
* Quản trị viên
* Đăng nhập
* Quản lý user
* Quản lý câu hỏi
* Quản lý tin tức
* Quẩn lý đề thi
* Quản lý môn
* Quản lý vai trò người dùng.
* Quản lý quyền vai trò
* Quản lý slide bar.

## **Biểu đồ ca sử dụng**



Hình 3. Biểu đồ usecase hệ thống của Website thi trắc nghiệm trực tuyến

## **Đặc tả ca sử dụng**

### **Đăng ký**

1. Mô tả tóm tắt

* Tên ca sử dụng: Đăng ký
* Mục đích: Mô tả công việc người dùng làm thế nào có thể đăng để có tài khoản trên website.
* Tác nhân: User

1. Tiền điều kiện: Không có
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng này bắt đầu khi tác nhân nhấn chọn đăng ký.
* Sau đó hệ thống sẽ yêu cầu user điền đầy đủ thông tin của mình vào mẫu gồm có: họ tên, email, số điện thoại, mật khẩu.
* Sau khi điền xong thông tin, user nhấn đăng ký
* Hệ thống sẽ kiểm tra thông tin điền đã hợp lệ hay chưa nếu thông tin hợp lệ thì thực hiện luồng A1. Nếu không hợp lệ thì thực hiện luồng A2.
* Hệ thống sẽ ghi lại tài khoản của user.
* Các luồng rẽ nhánh:
* Luồng A1: Nhập thông tin hợp lệ.
* Hệ thống kiểm tra tài khoản đã tồn tại chưa nếu có hiển thị thông báo lỗi
* User nhập lại thông tin, nếu tài khoản chưa tồn tại thì thông báo đăng ký tài khoản thành công
* Luồng A2: Nhập thông tin không hợp lệ.
* Hệ thống hiển thị thông báo lỗi.
* User có thể chọn hoặc là nhập lại thông tin sao cho hợp lệ hoặc là hủy bỏ đăng ký, khi đó ca sử dụng sẽ kết thúc.

1. Hậu điều kiện: Không có

### **Đăng nhập**

1. Mô tả tóm tắt

* Tên ca sử dụng: Đăng nhập
* Mục đích: Mô tả cách thức User đăng nhập vào hệ thống.
* Tác nhân: Admin, User.

1. Tiền điều kiện: Không có
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi User hoặc Admin chọn đăng nhập vào hệ thống.
* Hệ thống sẽ hiển thị giao diện để User hoặc Admin điền thông tin: tài khoản, mật khẩu.
* User hoặc Admin sẽ điền tài khoản được cấp và mật khẩu của mình vào.
* User hoặc Admin chọn nút đăng nhập. Nếu tài khoản đã được kích hoạt thì hệ thống thông báo thông báo đăng nhập thành công. Nếu lỗi sẽ thực hiện luồng A1.
* Luồng rẽ nhánh:
* Luồng A1: Đăng nhập không thành công:
* Hệ thống thông báo tài khoản hoặc mật khẩu nhập sai.
* User hoặc Admin nhập lại tài khoản hoặc mật khẩu nếu muốn tiếp tục đăng nhập.
* User hoặc Admin không muốn tiếp tục đăng nhập và chọn thoát.
* Ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu đăng nhập thành công, User hoặc Admin sẽ đăng nhập được vào hệ thống.

### **Quản lý câu hỏi**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý câu hỏi
* Mục đích: Ca sử dụng này cho phép Admin quản lý được thông tin câu hỏi trong hệ thống. Admin có thể thêm, sửa, xóa thông tin câu hỏi từ hệ thống
* Tác nhân: Admin

1. Tiền điều kiện: Đăng nhập thành công vào hệ thống
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi Admin muốn thêm sửa xóa thông tin câu hỏi từ hệ thống
* Hệ thống sẽ hiển thị các lựa chọn
* Hệ thống yêu cầu Admin chọn công việc mà họ muốn thực hiện.
* Nếu Admin chọn “Thêm câu hỏi”, luồng sự kiện con thêm câu hỏi sẽ được thực hiện.
* Nếu Admin chọn “Sửa thông tin câu hỏi”, luồng sự kiện con sửa thông tin câu hỏi sẽ được thực hiện.
* Nếu Admin chọn “Xóa câu hỏi”, luồng sự kiện con xóa câu hỏi được thực hiện.
* Thêm một câu hỏi:
* Hệ thống yêu cầu Admin nhập thông tin câu hỏi:
* Tiêu đề
* Nội dung
* Thời gian
* Môn thi
* Chương
* Kĩ năng
* Các đáp án
* Admin nhập thông tin được yêu cầu
* Hệ thống tạo ra Id tự động và gán cho câu hỏi.
* Một câu hỏi mới được thêm vào hệ thống.
* Hệ thống cung cấp cho Admin Id mới của câu hỏi.
* Sửa một câu hỏi:
* Hệ thống yêu cầu Admin nhập Id của câu hỏi.
* Admin nhập vào Id của câu hỏi.
* Hệ thống kiểm tra Id của câu hỏi vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về câu hỏi có Id vừa nhập.
* Admin sửa những thông tin cần thiết về câu hỏi và xác nhận việc sửa đổi.
* Hệ thống cập nhật những thay đổi vừa được thực hiện.
* Xóa một câu hỏi:
* Hệ thống yêu cầu Admin nhập Id của câu hỏi.
* Admin nhập vào Id câu hỏi.
* Hệ thống kiểm tra Id của câu hỏi vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về câu hỏi có Id vừa nhập.
* Hệ thống nhắc Admin xác nhận việc xóa câu hỏi.
* Admin xác nhận việc xóa.
* Câu hỏi đó sẽ bị xóa khỏi hệ thống.
* Luồng rẽ nhánh:
* Luồng A1: nhập Id câu hỏi không tồn tại
* Hệ thống thông báo lỗi
* Hệ thống yêu cầu Admin nhập vào Id câu hỏi khác hoặc hủy bỏ thao tác xóa hoặc sửa.
* Nếu quản trị viên nhập Id câu hỏi khác thì quay lại luồng xóa hoặc sửa.
* Nếu quản trị hệ thống chọn hủy, ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu ca sử dụng này được thực hiện thành công, thông tin về câu hỏi được thêm vào, sửa đổi hay xóa khỏi hệ thống. Trong các trường hợp khác hệ thống ở trong trạng thái chưa đổi.

### **Quản lý đề thi**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý đề thi
* Mục đích: Ca sử dụng này cho phép Admin quản lý được thông tin đề thi trong hệ thống. Admin có thể thêm, sửa, xóa thông tin đề thi từ hệ thống
* Tác nhân: Admin

1. Tiền điều kiện: Đăng nhập thành công vào hệ thống
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi Admin muốn thêm sửa xóa thông tin đề thi từ hệ thống
* Hệ thống sẽ hiển thị các lựa chọn
* Hệ thống yêu cầu Admin chọn công việc mà họ muốn thực hiện.
* Nếu Admin chọn “Thêm đề thi”, luồng sự kiện con thêm đề thi sẽ được thực hiện.
* Nếu Admin chọn “Sửa thông tin đề thi”, luồng sự kiện con sửa thông tin đề thi sẽ được thực hiện.
* Nếu Admin chọn “Xóa đề thi”, luồng sự kiện con xóa đề thi được thực hiện.
* Thêm một đề thi:
* Hệ thống yêu cầu Admin nhập thông tin đề thi:
* Tiêu đề
* Tên bài thi
* Môn thi
* Tài khoản phải làm
* Thời gian
* Số câu hỏi
* Số lần thi
* Tỉ lệ qua
* Ngày bắt đầu
* Ngày kết thúc
* Mô tả
* Admin nhập thông tin được yêu cầu
* Hệ thống tạo ra Id tự động và gán cho đề thi.
* Một câu hỏi mới được thêm vào hệ thống.
* Hệ thống cung cấp cho Admin Id mới của đề thi.
* Sửa một đề thi:
* Hệ thống yêu cầu Admin nhập Id của đề thi.
* Admin nhập vào Id của đề thi.
* Hệ thống kiểm tra Id của đề thi vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về đề thi có Id vừa nhập.
* Admin sửa những thông tin cần thiết về đề thi và xác nhận việc sửa đổi.
* Hệ thống cập nhật những thay đổi vừa được thực hiện.
* Xóa một đề thi:
* Hệ thống yêu cầu Admin nhập Id của đề thi.
* Admin nhập vào Id đề thi.
* Hệ thống kiểm tra Id của đề thi vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về đề thi có Id vừa nhập.
* Hệ thống nhắc Admin xác nhận việc xóa đề thi.
* Admin xác nhận việc xóa.
* Đề thi đó sẽ bị xóa khỏi hệ thống.
* Luồng rẽ nhánh:
* Luồng A1: nhập Id đề thi không tồn tại
* Hệ thống thông báo lỗi
* Hệ thống yêu cầu Admin nhập vào Id đề thi khác hoặc hủy bỏ thao tác xóa hoặc sửa.
* Nếu quản trị viên nhập Id đề thi khác thì quay lại luồng xóa hoặc sửa.
* Nếu quản trị hệ thống chọn hủy, ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu ca sử dụng này được thực hiện thành công, thông tin về đề thi được thêm vào, sửa đổi hay xóa khỏi hệ thống. Trong các trường hợp khác hệ thống ở trong trạng thái chưa đổi.

### **Quản lý người dùng**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý người dùng
* Mục đích: Ca sử dụng này cho phép Admin quản lý được thông tin người dùng trong hệ thống. Admin có thể thêm, sửa, xóa thông tin đề thi từ hệ thống
* Tác nhân: Admin

1. Tiền điều kiện: Đăng nhập thành công vào hệ thống
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi Admin muốn thêm sửa xóa thông tin người dùng từ hệ thống
* Hệ thống sẽ hiển thị các lựa chọn
* Hệ thống yêu cầu Admin chọn công việc mà họ muốn thực hiện.
* Nếu Admin chọn “Thêm người dùng”, luồng sự kiện con thêm người dùng sẽ được thực hiện.
* Nếu Admin chọn “Sửa thông tin người dùng”, luồng sự kiện con sửa thông tin người dùng sẽ được thực hiện.
* Nếu Admin chọn “Xóa người dùng”, luồng sự kiện con xóa người dùng được thực hiện.
* Thêm một người dùng:
* Hệ thống yêu cầu Admin nhập thông tin người dùng:
* Fullname
* Email
* Password
* Address
* Phone
* Birthday
* Admin nhập thông tin được yêu cầu
* Hệ thống tạo ra Id tự động và gán cho người dùng.
* Một người dùng mới được thêm vào hệ thống.
* Hệ thống cung cấp cho Admin Id mới của người dùng.
* Sửa một người dùng:
* Hệ thống yêu cầu Admin nhập Id của người dùng.
* Admin nhập vào Id của người dùng.
* Hệ thống kiểm tra Id của người dùng vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về người dùng có Id vừa nhập.
* Admin sửa những thông tin cần thiết về người dùng và xác nhận việc sửa đổi.
* Hệ thống cập nhật những thay đổi vừa được thực hiện.
* Xóa một người dùng:
* Hệ thống yêu cầu Admin nhập Id của người dùng.
* Admin nhập vào Id người dùng.
* Hệ thống kiểm tra Id của người dùng vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về người dùng có Id vừa nhập.
* Hệ thống nhắc Admin xác nhận việc xóa người dùng.
* Admin xác nhận việc xóa.
* Người dùng đó sẽ bị xóa khỏi hệ thống.
* Luồng rẽ nhánh:
* Luồng A1: nhập Id người dùng không tồn tại
* Hệ thống thông báo lỗi
* Hệ thống yêu cầu Admin nhập vào Id đề thi khác hoặc hủy bỏ thao tác xóa hoặc sửa.
* Nếu quản trị viên nhập Id người dùng khác thì quay lại luồng xóa hoặc sửa.
* Nếu quản trị hệ thống chọn hủy, ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu ca sử dụng này được thực hiện thành công, thông tin về người dùng được thêm vào, sửa đổi hay xóa khỏi hệ thống. Trong các trường hợp khác hệ thống ở trong trạng thái chưa đổi.

### **Quản lý môn**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý môn
* Mục đích: Ca sử dụng này cho phép Admin quản lý được thông tin môn trong hệ thống. Admin có thể thêm, sửa, xóa thông tin môn từ hệ thống
* Tác nhân: Admin

1. Tiền điều kiện: Đăng nhập thành công vào hệ thống
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi Admin muốn thêm sửa xóa thông tin môn từ hệ thống
* Hệ thống sẽ hiển thị các lựa chọn
* Hệ thống yêu cầu Admin chọn công việc mà họ muốn thực hiện.
* Nếu Admin chọn “Thêm môn”, luồng sự kiện con thêm môn sẽ được thực hiện.
* Nếu Admin chọn “Sửa thông tin môn”, luồng sự kiện con sửa thông tin môn sẽ được thực hiện.
* Nếu Admin chọn “Xóa môn”, luồng sự kiện con xóa môn được thực hiện.
* Thêm một môn:
* Hệ thống yêu cầu Admin nhập thông tin môn:
* Tên môn
* Admin nhập thông tin được yêu cầu
* Hệ thống tạo ra Id tự động và gán cho môn.
* Một môn mới được thêm vào hệ thống.
* Hệ thống cung cấp cho Admin Id mới của môn.
* Sửa một môn:
* Hệ thống yêu cầu Admin nhập Id của môn.
* Admin nhập vào Id của môn.
* Hệ thống kiểm tra Id của môn vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về môn có Id vừa nhập.
* Admin sửa những thông tin cần thiết về môn và xác nhận việc sửa đổi.
* Hệ thống cập nhật những thay đổi vừa được thực hiện.
* Xóa một môn:
* Hệ thống yêu cầu Admin nhập Id của môn.
* Admin nhập vào Id môn.
* Hệ thống kiểm tra Id của môn vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về môn có Id vừa nhập.
* Hệ thống nhắc Admin xác nhận việc xóa môn.
* Admin xác nhận việc xóa.
* Môn đó sẽ bị xóa khỏi hệ thống.
* Luồng rẽ nhánh:
* Luồng A1: nhập Id môn không tồn tại
* Hệ thống thông báo lỗi
* Hệ thống yêu cầu Admin nhập vào Id đề thi khác hoặc hủy bỏ thao tác xóa hoặc sửa.
* Nếu quản trị viên nhập Id môn khác thì quay lại luồng xóa hoặc sửa.
* Nếu quản trị hệ thống chọn hủy, ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu ca sử dụng này được thực hiện thành công, thông tin về môn được thêm vào, sửa đổi hay xóa khỏi hệ thống. Trong các trường hợp khác hệ thống ở trong trạng thái chưa đổi.

### **Quản lý tin tức**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý tin tức
* Mục đích: Ca sử dụng này cho phép Admin quản lý được thông tin tin tức trong hệ thống. Admin có thể thêm, sửa, xóa thông tin tin tức từ hệ thống
* Tác nhân: Admin

1. Tiền điều kiện: Đăng nhập thành công vào hệ thống
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi Admin muốn thêm sửa xóa thông tin tin tức từ hệ thống
* Hệ thống sẽ hiển thị các lựa chọn
* Hệ thống yêu cầu Admin chọn công việc mà họ muốn thực hiện.
* Nếu Admin chọn “Thêm tin tức”, luồng sự kiện con thêm tin tức sẽ được thực hiện.
* Nếu Admin chọn “Sửa thông tin tin tức”, luồng sự kiện con sửa thông tin tin tức sẽ được thực hiện.
* Nếu Admin chọn “Xóa tin tức”, luồng sự kiện con xóa tin tức được thực hiện.
* Thêm một tin tức:
* Hệ thống yêu cầu Admin nhập thông tin tin tức:
* Tiêu đề
* Hình ảnh
* Tóm tắt
* Tag
* Nội dung
* Admin nhập thông tin được yêu cầu
* Hệ thống tạo ra Id tự động và gán cho tin tức.
* Một tin tức mới được thêm vào hệ thống.
* Hệ thống cung cấp cho Admin Id mới của tin tức.
* Sửa một tin tức:
* Hệ thống yêu cầu Admin nhập Id của tin tức.
* Admin nhập vào Id của tin tức.
* Hệ thống kiểm tra Id của tin tức vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về tin tức có Id vừa nhập.
* Admin sửa những thông tin cần thiết về tin tức và xác nhận việc sửa đổi.
* Hệ thống cập nhật những thay đổi vừa được thực hiện.
* Xóa một tin tức:
* Hệ thống yêu cầu Admin nhập Id của tin tức.
* Admin nhập vào Id tin tức.
* Hệ thống kiểm tra Id của tin tức vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về tin tức có Id vừa nhập.
* Hệ thống nhắc Admin xác nhận việc xóa tin tức.
* Admin xác nhận việc xóa.
* Tin tức đó sẽ bị xóa khỏi hệ thống.
* Luồng rẽ nhánh:
* Luồng A1: nhập Id tin tức không tồn tại
* Hệ thống thông báo lỗi
* Hệ thống yêu cầu Admin nhập vào Id tin tức khác hoặc hủy bỏ thao tác xóa hoặc sửa.
* Nếu quản trị viên nhập Id tin tức khác thì quay lại luồng xóa hoặc sửa.
* Nếu quản trị hệ thống chọn hủy, ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu ca sử dụng này được thực hiện thành công, thông tin về tin tức được thêm vào, sửa đổi hay xóa khỏi hệ thống. Trong các trường hợp khác hệ thống ở trong trạng thái chưa đổi.

### **Quản lý slider bar**

1. Mô tả tóm tắt

* Tên ca sử dụng: Quản lý slider bar
* Mục đích: Ca sử dụng này cho phép Admin quản lý được thông tin slider bar trong hệ thống. Admin có thể thêm, sửa, xóa thông tin slider bar từ hệ thống
* Tác nhân: Admin

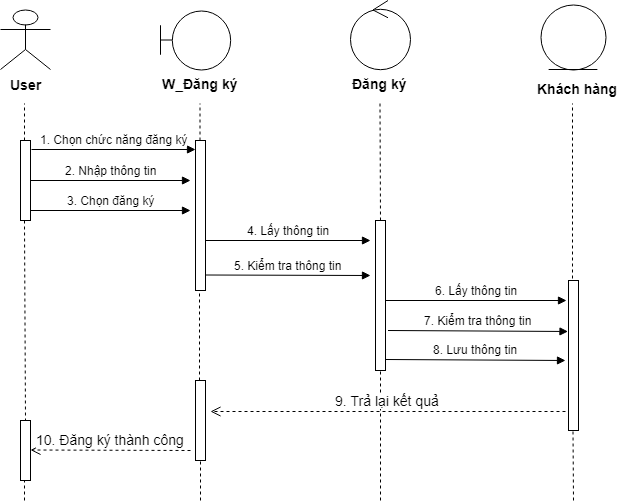
1. Tiền điều kiện: Đăng nhập thành công vào hệ thống
2. Các luồng sự kiện

* Luồng sự kiện chính: Ca sử dụng bắt đầu khi Admin muốn thêm sửa xóa thông tin slider bar từ hệ thống
* Hệ thống sẽ hiển thị các lựa chọn
* Hệ thống yêu cầu Admin chọn công việc mà họ muốn thực hiện.
* Nếu Admin chọn “Thêm slider bar”, luồng sự kiện con thêm slider bar sẽ được thực hiện.
* Nếu Admin chọn “Sửa thông tin slider bar”, luồng sự kiện con sửa thông tin slider bar sẽ được thực hiện.
* Nếu Admin chọn “Xóa slider bar”, luồng sự kiện con xóa slider bar được thực hiện.
* Thêm một slider bar:
* Hệ thống yêu cầu Admin nhập thông tin slider bar:
* Tiêu đề
* Ảnh nền
* Khẩu hiệu
* Admin nhập thông tin được yêu cầu
* Hệ thống tạo ra Id tự động và gán cho slider bar.
* Một slider bar mới được thêm vào hệ thống.
* Hệ thống cung cấp cho Admin Id mới của slider bar.
* Sửa một slider bar:
* Hệ thống yêu cầu Admin nhập Id của slider bar.
* Admin nhập vào Id của slider bar.
* Hệ thống kiểm tra Id của slider bar vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về slider bar có Id vừa nhập.
* Admin sửa những thông tin cần thiết về slider bar và xác nhận việc sửa đổi.
* Hệ thống cập nhật những thay đổi vừa được thực hiện.
* Xóa một slider bar:
* Hệ thống yêu cầu Admin nhập Id của slider bar.
* Admin nhập vào Id slider bar.
* Hệ thống kiểm tra Id của slider bar vừa nhập có tồn tại hay không. Nếu không tồn tại thì thực hiện luồng A1.
* Hệ thống hiển thị thông tin về slider bar có Id vừa nhập.
* Hệ thống nhắc Admin xác nhận việc xóa slider bar.
* Admin xác nhận việc xóa.
* Slider bar đó sẽ bị xóa khỏi hệ thống.
* Luồng rẽ nhánh:
* Luồng A1: nhập Id slider bar không tồn tại
* Hệ thống thông báo lỗi
* Hệ thống yêu cầu Admin nhập vào Id slider bar khác hoặc hủy bỏ thao tác xóa hoặc sửa.
* Nếu quản trị viên nhập Id slider bar khác thì quay lại luồng xóa hoặc sửa.
* Nếu quản trị hệ thống chọn hủy, ca sử dụng kết thúc.

1. Hậu điều kiện: Nếu ca sử dụng này được thực hiện thành công, thông tin về slider bar được thêm vào, sửa đổi hay xóa khỏi hệ thống. Trong các trường hợp khác hệ thống ở trong trạng thái chưa đổi.

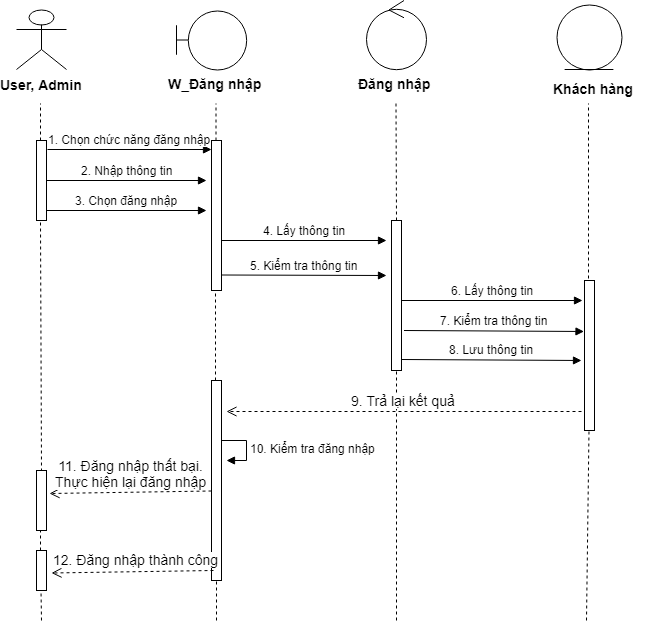
## **Biểu đồ trình tự ca sử dụng**

### **Biểu đồ trình tự cho ca sử dụng “Đăng ký”**



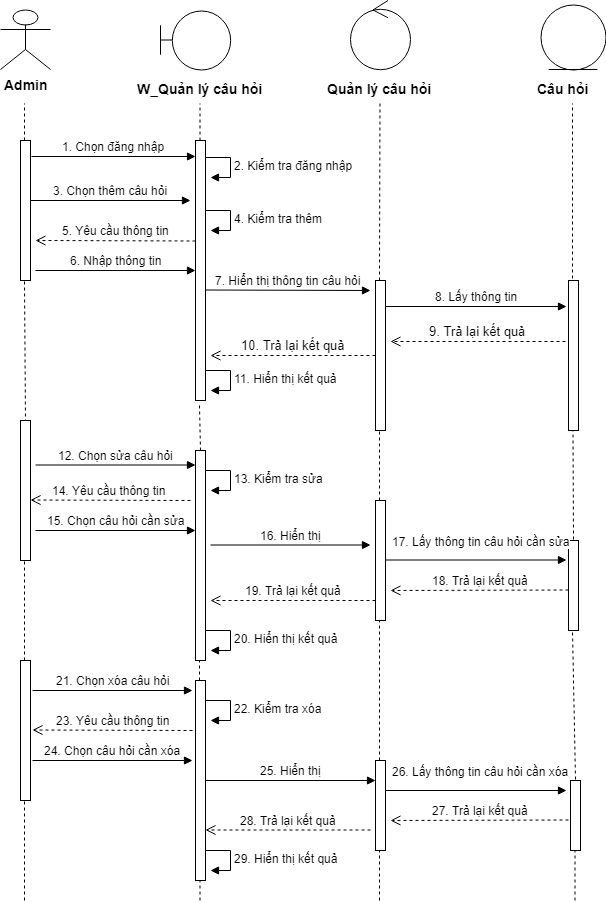
Hình 4: Biều đồ trình tự ca sử dụng đăng ký

### **Biểu đồ trình tự cho ca sử dụng “Đăng nhập”**



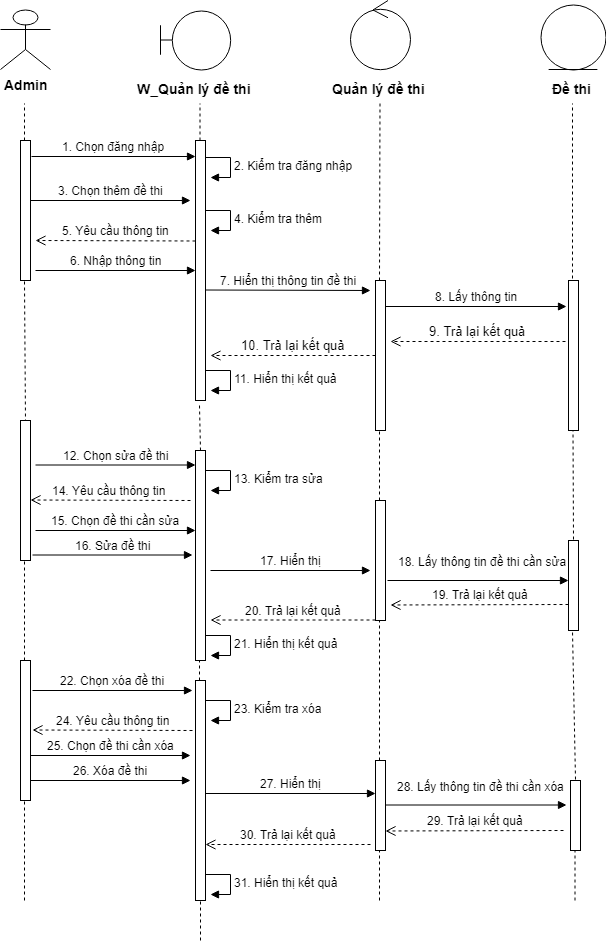
Hình 5: Biều đồ trình tự ca sử dụng đăng nhập

### **Biểu đồ trình tự cho ca sử dụng “Quản lý câu hỏi”**



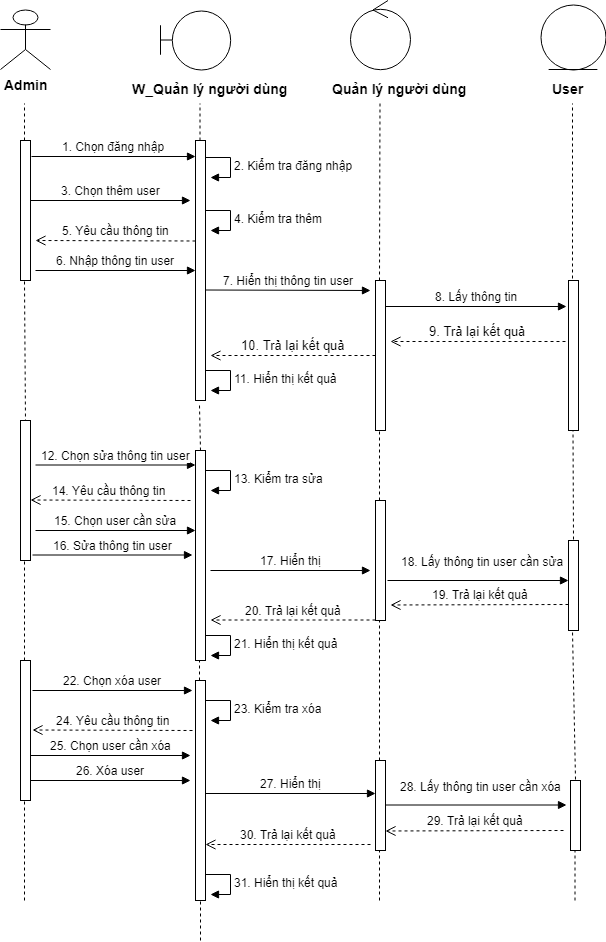
Hình 6: Biểu đồ trình tự ca sử dụng quản lý câu hỏi

### **Biểu đồ trình tự cho ca sử dụng “Quản lý đề thi”**



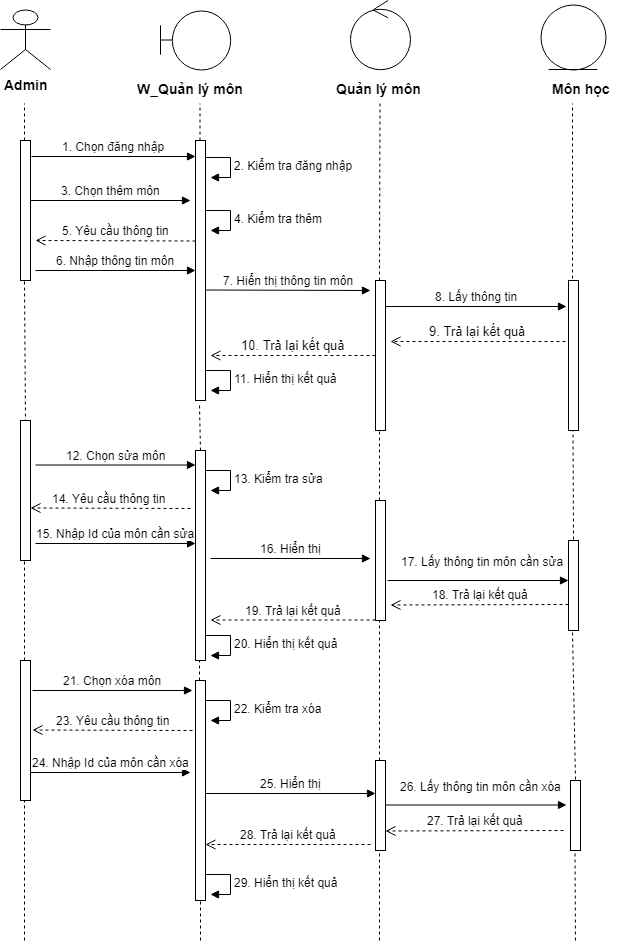
Hình 7: Biểu đồ trình tự ca sử dụng quản lý đề thi

### **Biểu đồ trình tự cho ca sử dụng “Quản lý người dùng”**



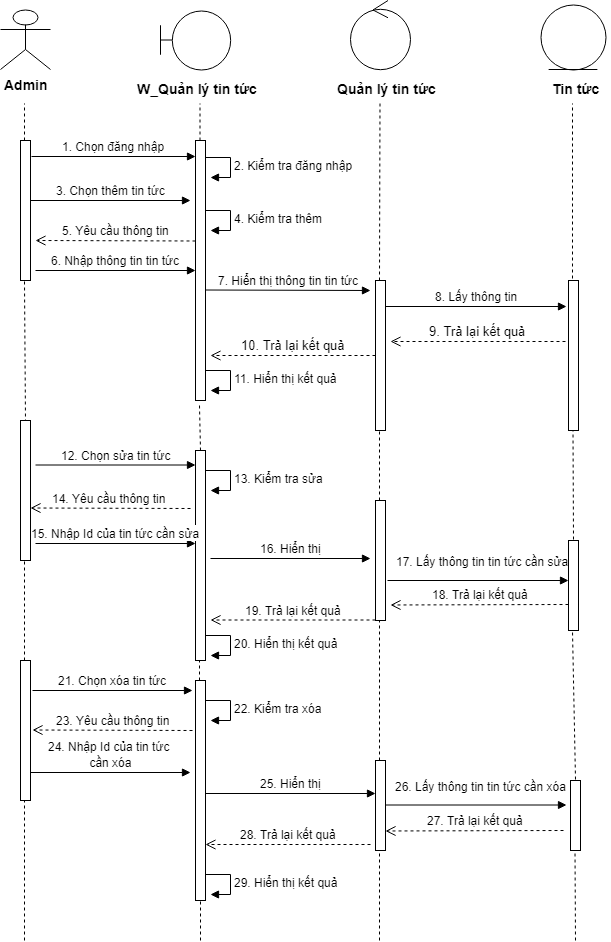
Hình 8: Biểu đồ trình tự ca sử dụng quản lý người dùng

### **Biểu đồ trình tự cho ca sử dụng “Quản lý môn”**



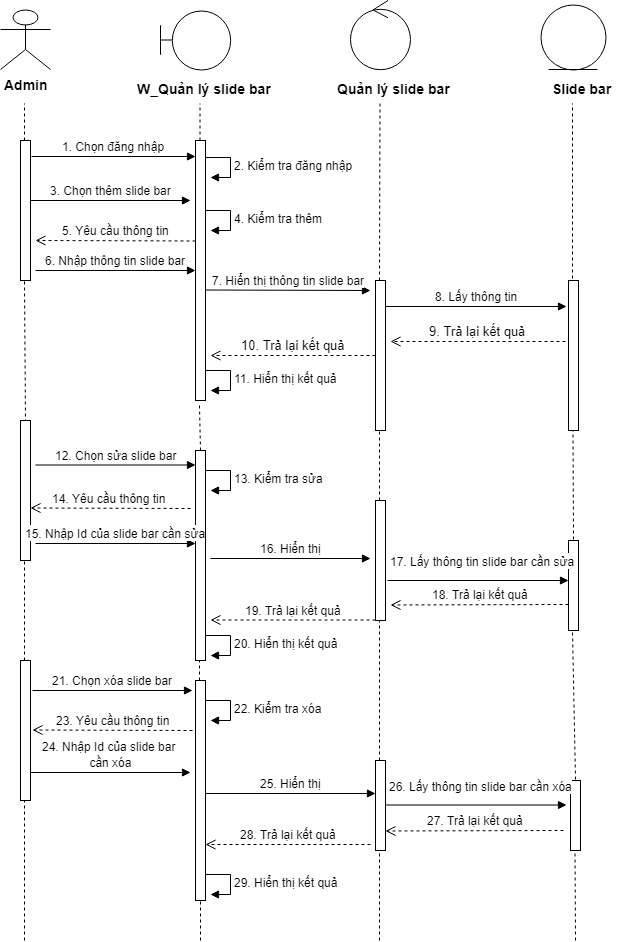
Hình 9: Biểu đồ trình tự ca sử dụng quản lý môn

### **Biểu đồ trình tự cho ca sử dụng “Quản lý tin tức”**



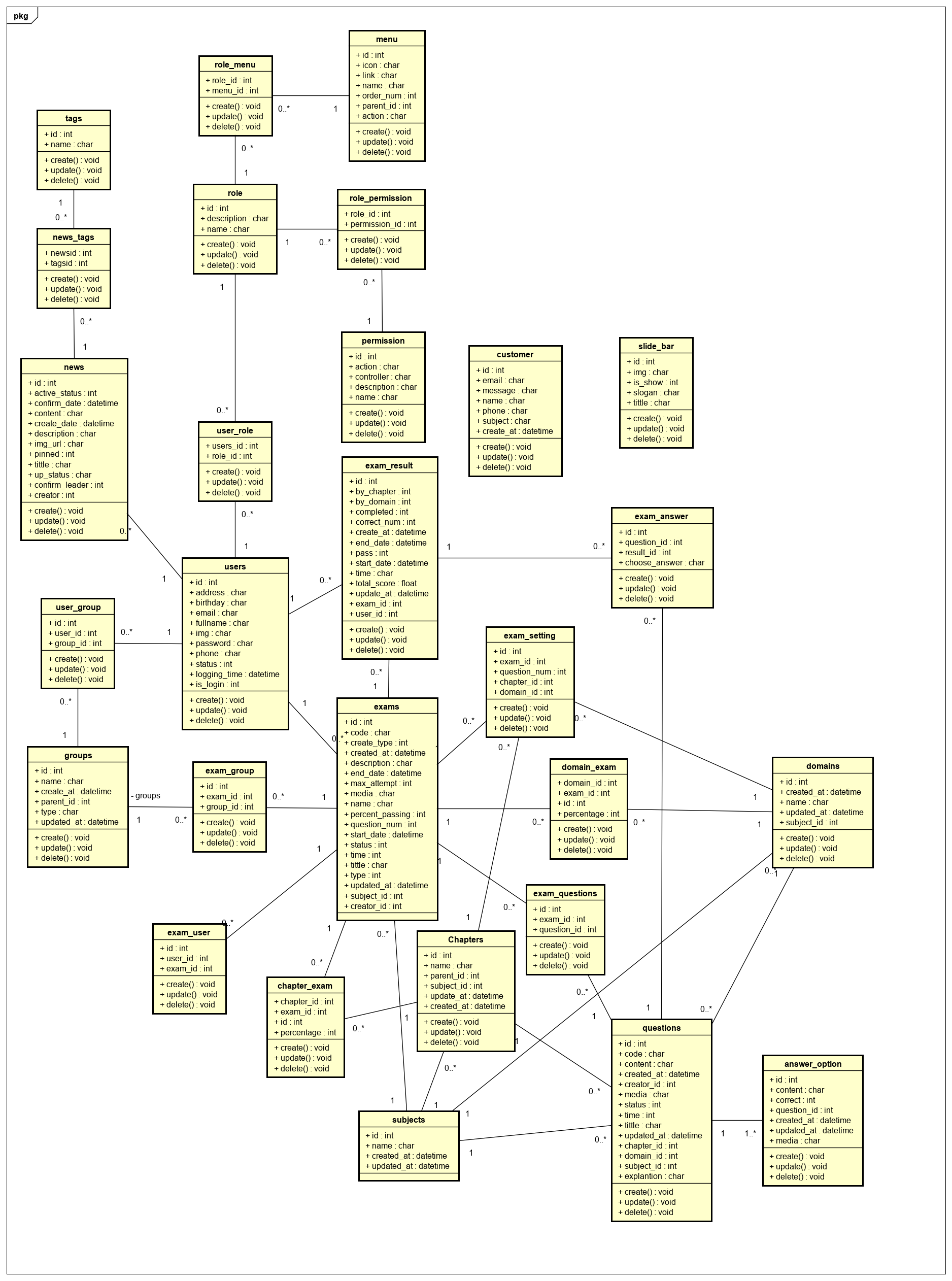
Hình 10: Biểu đồ trình tự ca sử dụng quản lý tin tức

### **Biểu đồ trình tự cho ca sử dụng “Quản lý slider bar”**



Hình 11: Biểu đồ trình tự ca sử dụng quản lý slider bar

## **Biểu đồ lớp chi tiết**



Hình 12. Biểu đồ lớp chi tiết

## **Thiết kế cơ sở dữ liệu**

### **Chi tiết bảng cơ sở dữ liệu**

* **Answer\_option**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | int | Not Null | PK | Id câu trả lời |
| 2 | content | Nvachar(MAX) | Not Null |  | Nội dung câu trả lời |
| 3 | correct | int | Not null |  | Đúng/Sai |
| 4 | media | Varchar(256) | Not null |  | Media |
| 5 | question\_id | int | Not null | FK | Id câu hỏi |
| 6 | created\_at | Datetime | Not null |  | Ngày tạo |
| 7 | update\_at | Datetime | Not null |  | Ngày sửa |

Bảng 1: Chi tiết bảng cơ sở dữ liệu Answer\_option

* **Chapter\_exam**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | Chapter\_id | Int | Not null | FK | Id chapter |
| 3 | Exam\_id | int | Not Null | FK | Id exam |
| 5 | Percentage | Int | Not nul |  | Tỷ lệ phần trăm |

Bảng 2: Chi tiết bảng cơ sở dữ liệu Chapter\_exam

* **Chapters**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | int | Not Null | PK | Id chapter |
| 2 | name | Varchar(50) | Not null |  | Tên chapter |
| 3 | Parent\_id | int | Not Null |  | Id parent |
| 4 | Subject\_id | int | Not null | FK | Id môn học |
| 5 | Created\_at | Datetime | Not nul |  | Ngày tạo |
| 6 | Updated\_at | Datetime | Not null |  | Ngày sửa |

Bảng 3: Chi tiết bảng cơ sở dữ liệu Chapters

* **Customer**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id liên hệ |
| 2 | Name | Varchar(50) | Not null |  | Tên người liên hệ |
| 3 | Email | Varchar(50) | Not Null |  | Email người liên hệ |
| 4 | Message | Varchar(255) | Not null |  | Lời nhắn |
| 5 | Phone | Varchar(50) | Not nul |  | Số điện thoại |
| 6 | Subject | Varchar(50) | Not null |  | Tiêu đề |
| 7 | Created\_at | Datetime | Not null |  | Ngày tạo |

Bảng 4: Chi tiết bảng cơ sở dữ liệu Customer

* **Domain\_exam**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | Percentage | Int | Not null |  | Tỷ lệ phần tram |
| 3 | Domain\_id | Int | Not Null | FK | Domain id |
| 4 | Exam\_id | Int | Not null | FK | Exam id |

Bảng 5: Chi tiết bảng cơ sở dữ liệu Domain\_exam

* **Domains**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id domain |
| 2 | Name | Varchar(50) | Not null |  | Tên domain |
| 3 | Subject\_id | Int | Not null | FK | Id môn học |
| 4 | Created\_at | Datetime | Not nul |  | Ngày tạo |
| 5 | Updated\_at | Datetime | Not null |  | Ngày sửa |

Bảng 6: Chi tiết bảng cơ sở dữ liệu Domains

* **Exam\_answer**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | Choose\_answer | Varchar(20) | Not null |  | Câu trả lời chọn |
| 3 | Question\_id | Int | Not Null | FK | Id câu hỏi |
| 4 | Result\_id | Int | Not null | FK | Id kết quả |

Bảng 7: Chi tiết bảng cơ sở dữ liệu Exam\_answer

* **Exam\_group**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Int |
| 2 | Exam\_id | Int | Not null | FK | Id exam |
| 3 | Group\_id | Int | Not Null | FK | Id group |

Bảng 8: Chi tiết bảng cơ sở dữ liệu Exam\_group

* **Exam\_questions**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | Question\_id | Int | Not null | FK | Id question |
| 3 | Exam\_id | Int | Not Null | FK | Id exam |

Bảng 9: Chi tiết bảng cơ sở dữ liệu Exam\_questions

* **Exam\_result**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | By\_chapter | Int | Not Null |  | Theo chương |
| 3 | By\_domain | Int | Not null |  | Theo domain |
| 4 | Completed | Int | Not null |  | Tình trạng hoàn thành |
| 5 | Correct\_num | Int | Not null |  | Số câu đúng |
| 6 | Created\_at | Datetime | Not null |  | Ngày tạo |
| 7 | End\_date | Datetime | Not null |  | Ngày kết thúc |
| 8 | Pass | Int | Not null |  | Pass/Fail |
| 9 | Start\_date | Datetime | Not null |  | Ngày bắt đầu |
| 10 | Time | Varchar(10) | Not null |  | Thời gian làm bài |
| 11 | Total\_score | Float | Not null |  | Tổng điểm |
| 12 | Updated\_at | Datetime | Not null |  | Ngày sửa |
| 13 | Exam\_id | Int | Not null | FK | Id exam |
| 14 | User\_id | Int | Not null | FK | Id user |

Bảng 10: Chi tiết bảng cơ sở dữ liệu Exam\_result

* **Exam\_setting**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | Question\_num | Int | Not Null |  | Số câu hỏi |
| 3 | Chapter\_id | Int | Not null | FK | Id chapter |
| 4 | Domain\_id | Int | Not null | FK | Id domain |
| 5 | Exam\_id | Int | Not null | FK | Id exam |

Bảng 11: Chi tiết bảng cơ sở dữ liệu Exam\_setting

* **Exam\_user**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id |
| 2 | Exam\_id | Int | Not Null | FK | Id exam |
| 3 | User\_id | Int | Not null | FK | Id user |

Bảng 12: Chi tiết bảng cơ sở dữ liệu Exam\_user

* **Exams**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id exam |
| 2 | Code | Varchar(20) | Not Null |  | Mã exam |
| 3 | Create\_type | Int | Not null |  | Kiểu tạo exam |
| 4 | Description | Varchar(255) | Not null |  | Mô tả |
| 5 | End\_date | Datetime | Not null |  | Ngày kết thúc |
| 6 | Max\_attempt | Int | Not null |  | Số lần thi |
| 7 | Media | Varchar(256) | Not null |  | Media |
| 8 | Name | Varchar(50) | Not null |  | Tên exam |
| 9 | Percent\_passing | Int | Not null |  | Phần trăm pass |
| 10 | Question\_num | Int | Not null |  | Số câu hỏi |
| 11 | Start\_date | Datetime | Not null |  | Ngày bắt đầu |
| 12 | Status | Int | Not null |  | Trạng thái |
| 13 | Time | Int | Not null |  | Thời gian |
| 14 | Tittle | Varchar(256)) | Not null |  | Tiêu đề |
| 15 | Type | int | Not null |  | Loại exam |
| 16 | Updated\_at | Datetime | Not null |  | Ngày sửa |
| 17 | Subject\_id | Int | Not null | FK | Id subject |
| 18 | Creator\_id | Int | Not null | FK | Int người tạo |

Bảng 13: Chi tiết bảng cơ sở dữ liệu Exams

* **Group**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id group |
| 2 | Name | Varchar(50) | Not Null |  | Tên group |
| 3 | Type | Varchar(20) | Not null |  | Loại group |
| 4 | Parent\_id | Int | Not null |  | Id parent |
| 5 | Created\_at | Datetime | Not null |  | Ngày tạo |
| 6 | Updated\_at | Datetime | Not null |  | Ngày sửa |

Bảng 14: Chi tiết bảng cơ sở dữ liệu Group

* **Menu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Int menu |
| 2 | Name | Varchar(50) | Not Null |  | Tên menu |
| 3 | Icon | Varchar(50) | Not null |  | Icon |
| 4 | Link | Varchar(256) | Not null |  | Link |
| 5 | Order\_num | Int | Not null |  | Vị trí |
| 6 | Parent\_id | Int | Not null |  | Id gốc |
| 7 | Action | Varchar(30) | Not null |  | action |

Bảng 15: Chi tiết bảng cơ sở dữ liệu Menu

* **News**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id news |
| 2 | Active\_status | Int | Not null |  | Trạng thái kích hoạt |
| 3 | Confirm\_date | Datetime | Not Null |  | Ngày xác nhận |
| 4 | Content | Varchar(12000) | Not nul |  | Nội dung |
| 5 | Create\_date | Datetime | Not null |  | Ngày tạo |
| 6 | The\_description | Varchar(5000) | Not null |  | Mô tả |
| 7 | Img\_url | Varchar(255) | Not null |  | url hình ảnh |
| 8 | Pinned | Int | Not null |  | Trạng thái pinned |
| 9 | Title | Varchar(1000) | Not null |  | Tiêu đề |
| 10 | Up\_status | Varchar(255) | Not null |  | Trạng thái approve |
| 11 | Confirm\_leader | Int | Not null | FK | Leader confirm |
| 12 | Creator | Int | Not null | FK | Id Người tạo |

Bảng 16: Chi tiết bảng cơ sở dữ liệu News

* **News\_tags**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Newsid | Int | Not Null | FK | Id news |
| 2 | Tagsid | Int | Not null | FK | Id tag |

Bảng 17: Chi tiết bảng cơ sở dữ liệu News\_tags

* **Permission**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id quyền |
| 2 | Action | Varchar(30) | Not null |  | Action |
| 3 | Controller | Varchar(30) | Not Null |  | Controller |
| 4 | Description | Varchar(50) | Not null |  | Mô tả |
| 5 | Name | Varchar(20) | Not null |  | Tên quyền |

Bảng 18: Chi tiết bảng cơ sở dữ liệu Permission

* **Questions**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id câu hỏi |
| 2 | Code | Varchar(20) | Not null |  | Mã câu hỏi |
| 3 | Content | Varchar(256) | Not Null |  | Nội dung |
| 4 | Created\_at | Datetime | Not Null |  | Ngày tạo |
| 5 | Creator\_id | Int | Not null |  | Id người tạo |
| 6 | Media | Varchar(256) | Not Null |  | Media |
| 7 | Status | Int | Not null |  | Trạng thái |
| 8 | Time | Int | Not null |  | Thời gian |
| 9 | Title | Varchar(256) | Not null |  | Tiêu đề |
| 10 | Updated\_at | Datetime |  |  | Ngày sửa |
| 11 | Chapter\_id | Int | Not null | FK | Id chapter |
| 12 | Domain\_id | Int | Not null | FK | Id domain |
| 13 | Subject\_id | Int | Not null | FK | Id subject |
| 14 | Explantion | Varchar(500) |  |  | Giải thích |

Bảng 19: Chi tiết bảng cơ sở dữ liệu Questions

* **Role**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id role |
| 2 | Description | Varchar(50) | Not null |  | Mô tả |
| 3 | Name | Varchar(20) | Not Null |  | Tên role |

Bảng 20: Chi tiết bảng cơ sở dữ liệu Role

* **Role\_menu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Role\_id | Int | Not Null | PK | Id role |
| 2 | Menu\_id | Int | Not null | PK | Id menu |

Bảng 21: Chi tiết bảng cơ sở dữ liệu Role\_menu

* **Role\_permission**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Role\_id | Int | Not Null | PK | Id role |
| 2 | Permission\_id | Int | Not null | PK | Id permission |

Bảng 22: Chi tiết bảng cơ sở dữ liệu Role\_permission

* **Slide\_bar**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id slide |
| 2 | Img | Varchar(225) | Not null |  | Hình ảnh |
| 3 | Is\_show | Int | Not Null |  | Trạng thái |
| 4 | Slogan | Varchar(260) | Not null |  | Slogan |
| 5 | Title | Varchar(60) | Not null |  | Tiêu đề |

Bảng 23: Chi tiết bảng cơ sở dữ liệu Slide\_bar

* **Subject**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id subject |
| 2 | Created\_at | Datetime | Not null |  | Ngày tạo |
| 3 | Name | Varchar(50) | Not Null |  | Tên subject |
| 4 | Updated\_at | Datetime |  |  | Ngày sửa |

Bảng 24: Chi tiết bảng cơ sở dữ liệu Subject

* **Tags**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id tag |
| 2 | Name | Varchar(255) | Not Null |  | Tên tag |

Bảng 25: Chi tiết bảng cơ sở dữ liệu Tags

* **User\_group**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id group |
| 2 | User\_id | Int | Not null | FK | Id user |
| 3 | Group\_id | Int | Not Null | FK | Id group |

Bảng 26: Chi tiết bảng cơ sở dữ liệu User\_group

* **User**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | Id | Int | Not Null | PK | Id user |
| 2 | Address | Varchar(100) | Not null |  | Địa chỉ |
| 3 | Birthday | Varchar(10) | Not Null |  | Ngày sinh |
| 4 | Email | Varchar(50) | Not null |  | Email |
| 5 | Fullname | Varchar(50) | Not null |  | Họ và tên |
| 6 | Img | Varchar(100) |  |  | Hình ảnh |
| 7 | Password | Varchar(100) | Not null |  | Mật khẩu |
| 8 | Phone | Varchar(50) | Not null |  | Số điện thoại |
| 9 | Status | Int | Not null |  | Trạng thái |
| 10 | Logging\_time | Datetime |  |  | Ngày đăng nhập |
| 11 | Is\_login | Int | Not null |  | Is login |

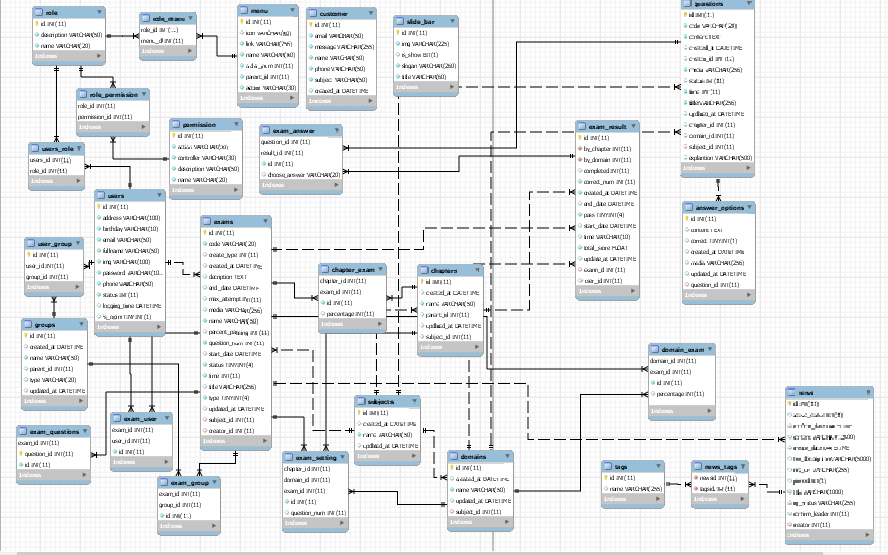
Bảng 27: Chi tiết bảng cơ sở dữ liệu User

* **User\_role**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | PK/FK | Mô tả |
| 1 | User\_id | Int | Not Null | FK | Id user |
| 2 | Role\_id | Int | Not null | FK | Id role |

Bảng 28: Chi tiết bảng cơ sở dữ liệu User\_role

### **Mô hình dữ liệu liên kết vật lý**

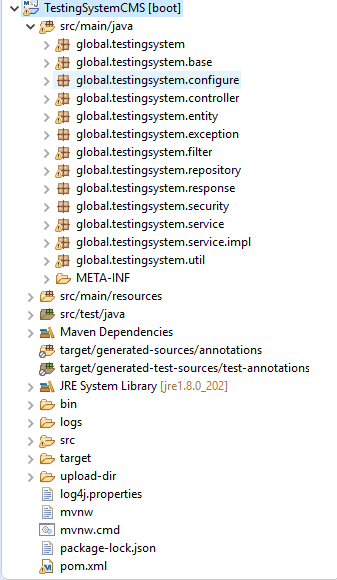


Hình 13. Biểu đồ mô hình dữ liệu liên kết vật lý

**CHƯƠNG 3: CÀI ĐẶT**

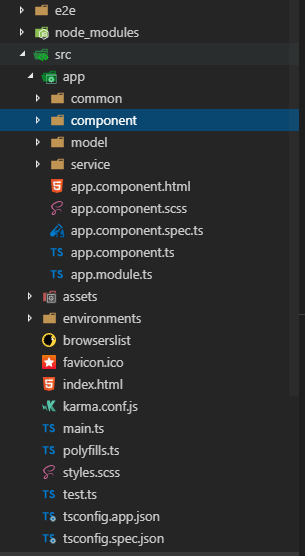
## **Cấu trúc Project**

* + 1. **Backend**

**

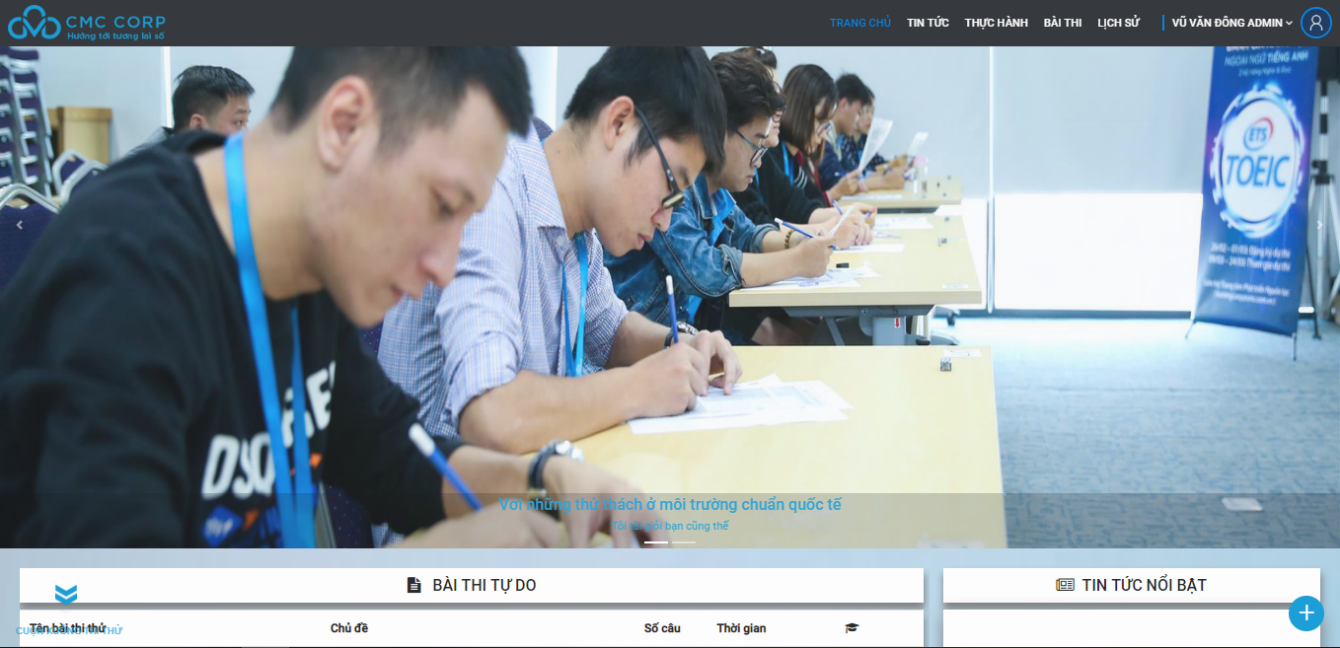
Hình 14. Cấu trúc project backend

* + 1. **Frontend**

****

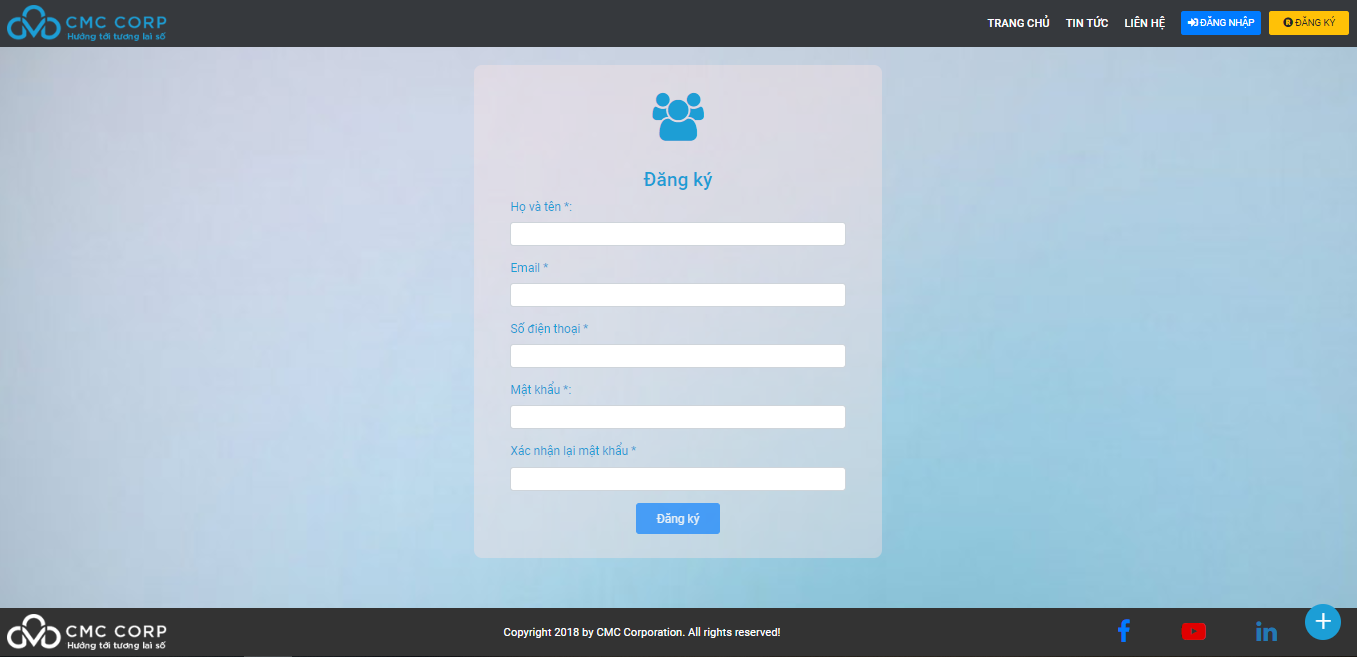
Hình 15. Cấu trúc project frontend

* 1. **Một số giao diện website sau khi cài đặt chương trình**
     1. **Giao diện trang user**
  2. **Trang chủ**

****

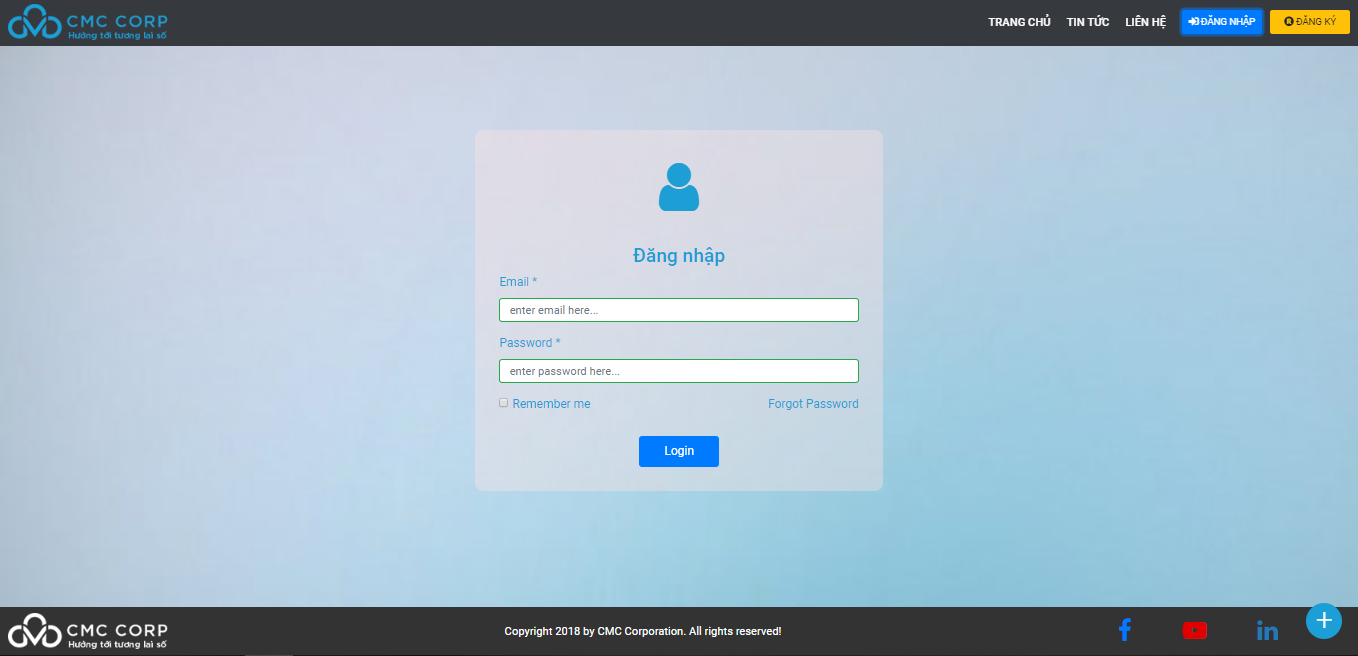
Hình 16: Giao diện trang chủ trang khách hàng

* 1. **Trang đăng ký tài khoản**

****

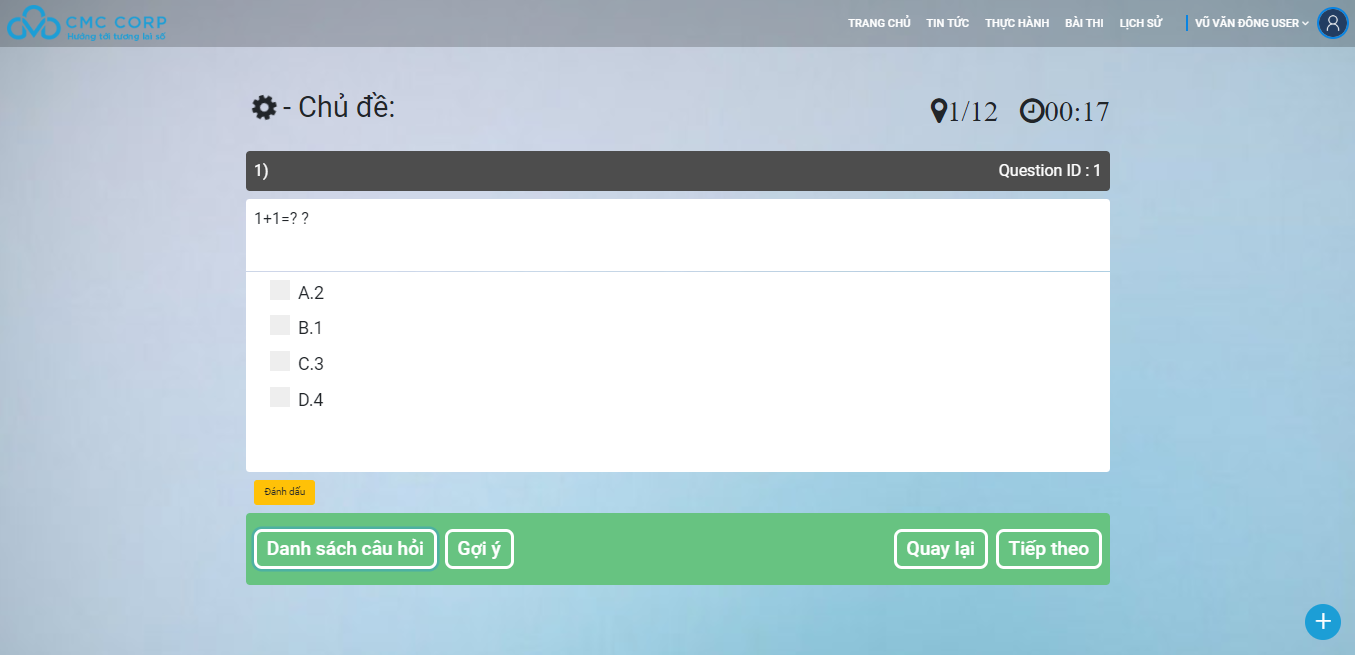
Hình 17: Giao diện trang chủ trang đăng ký tài khoản

* 1. **Trang đăng nhập**

****

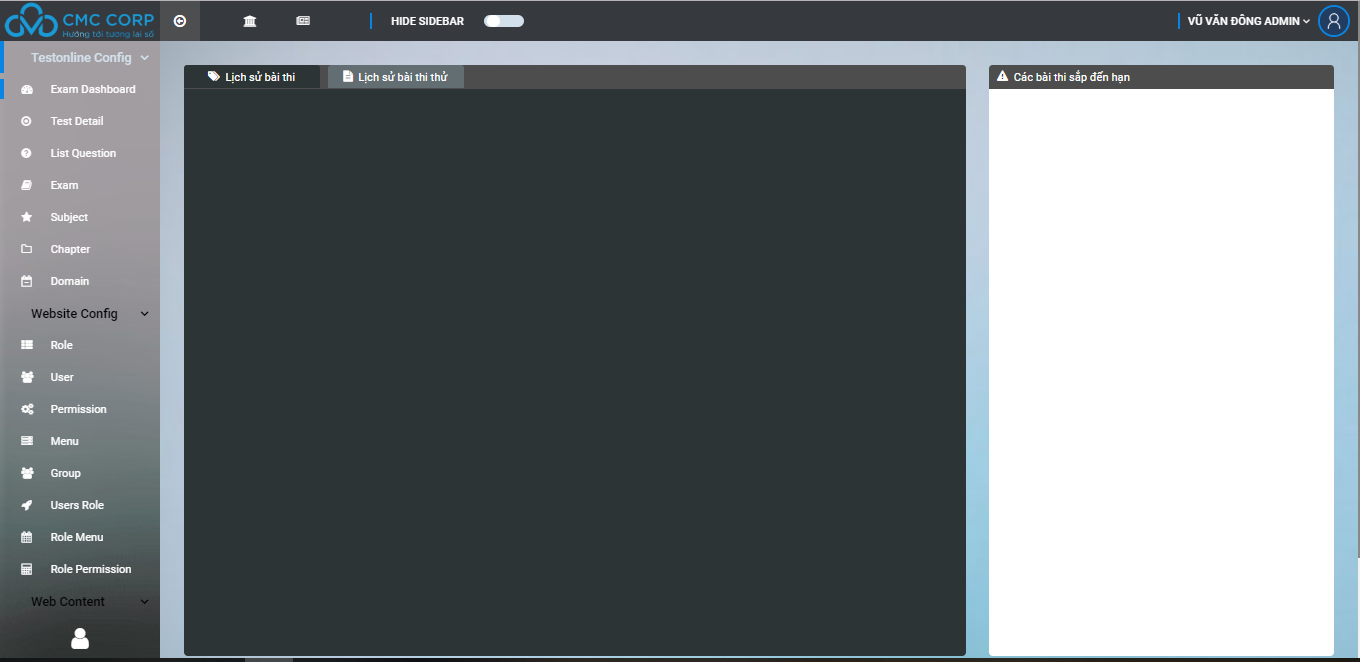
Hình 18: Giao diện trang chủ trang đăng nhập

* 1. **Trang thi**

****

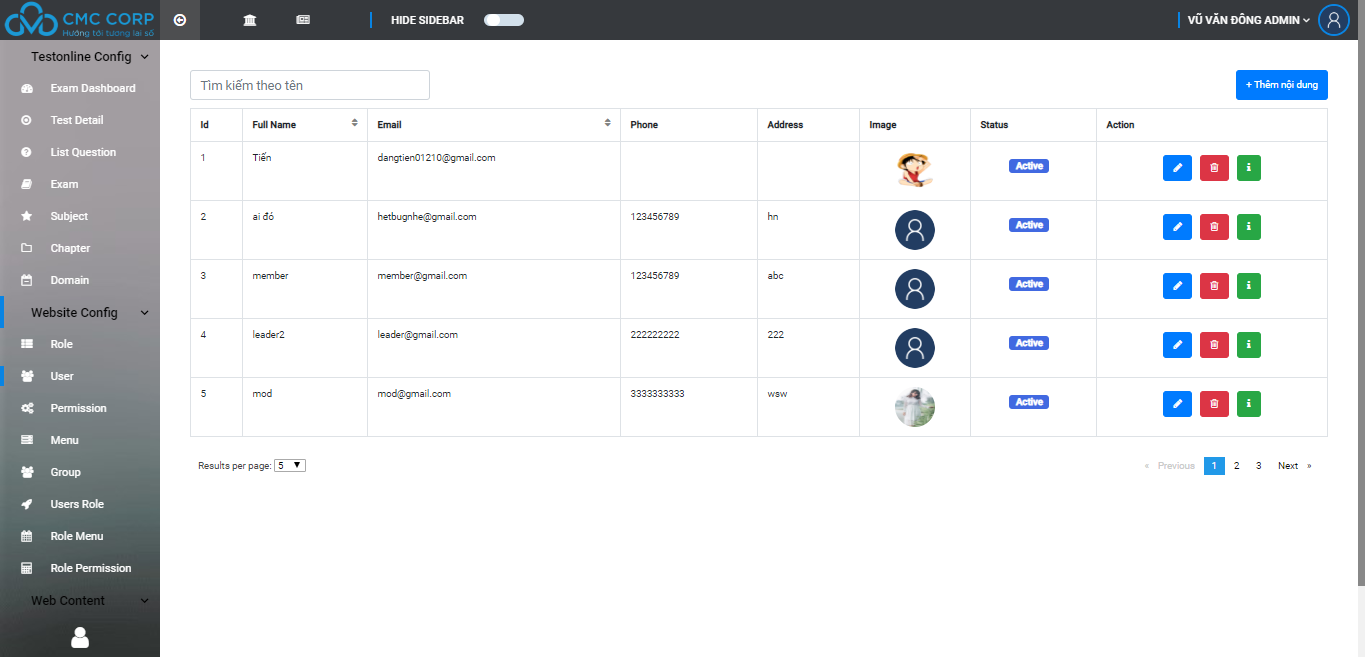
Hình 19: Giao diện trang thi

* + 1. **Giao diện trang admin**
       1. **Trang chủ**

****

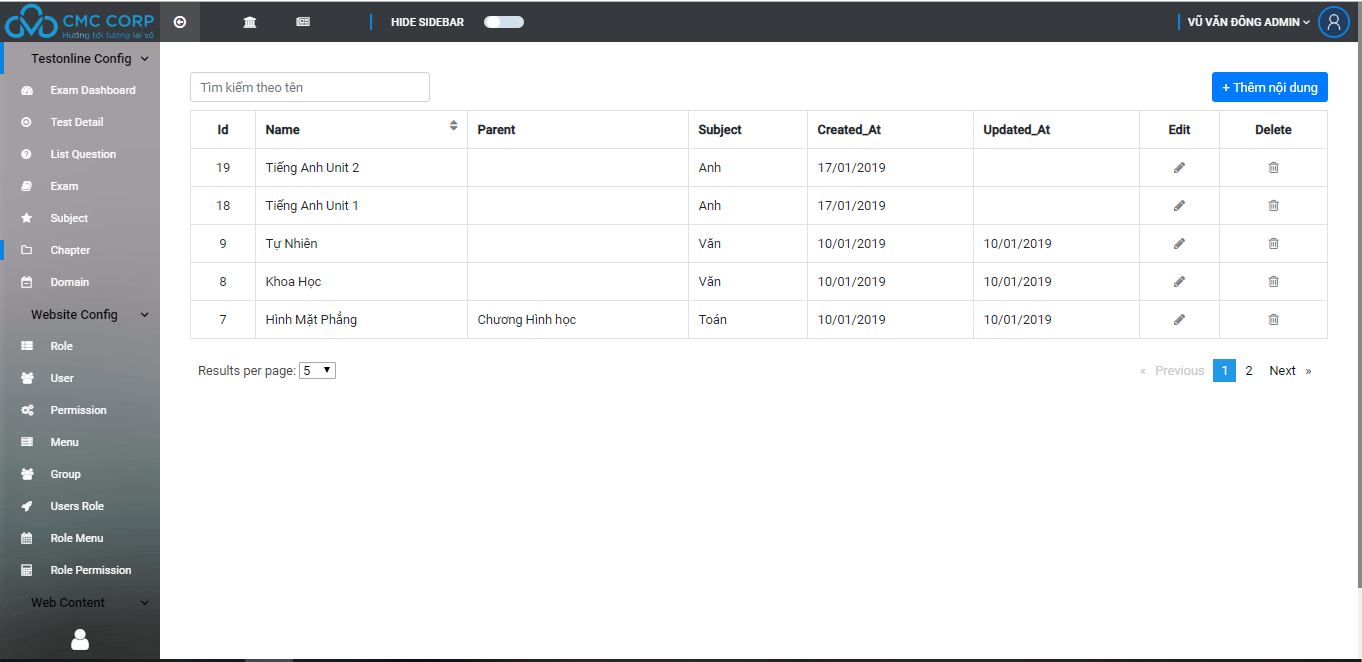
Hình 20: Giao diện trang tổng quan

* + - 1. **Trang quản lý người dùng**

****

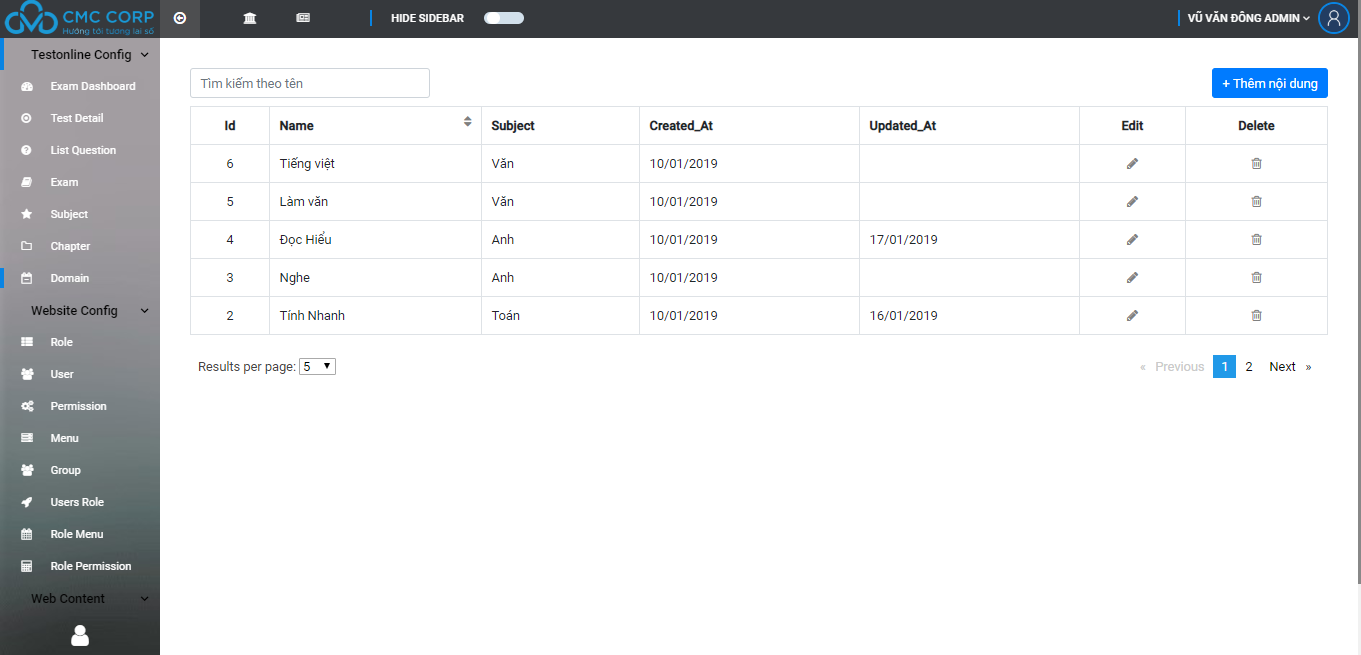
Hình 21: Giao diện trang quản lý người dùng

* + - 1. **Trang phân quản lý chapter**

****

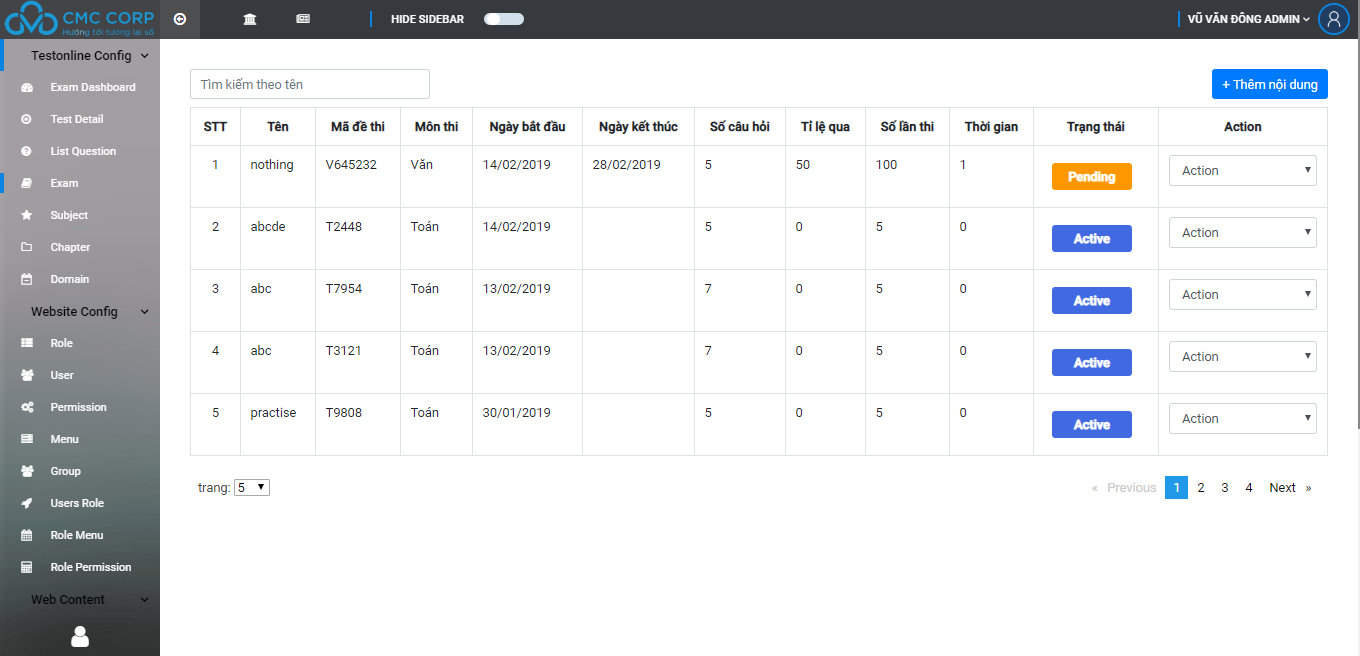
Hình 22: Giao diện trang quản lý chapter

* + - 1. **Trang quản lý domain**

****

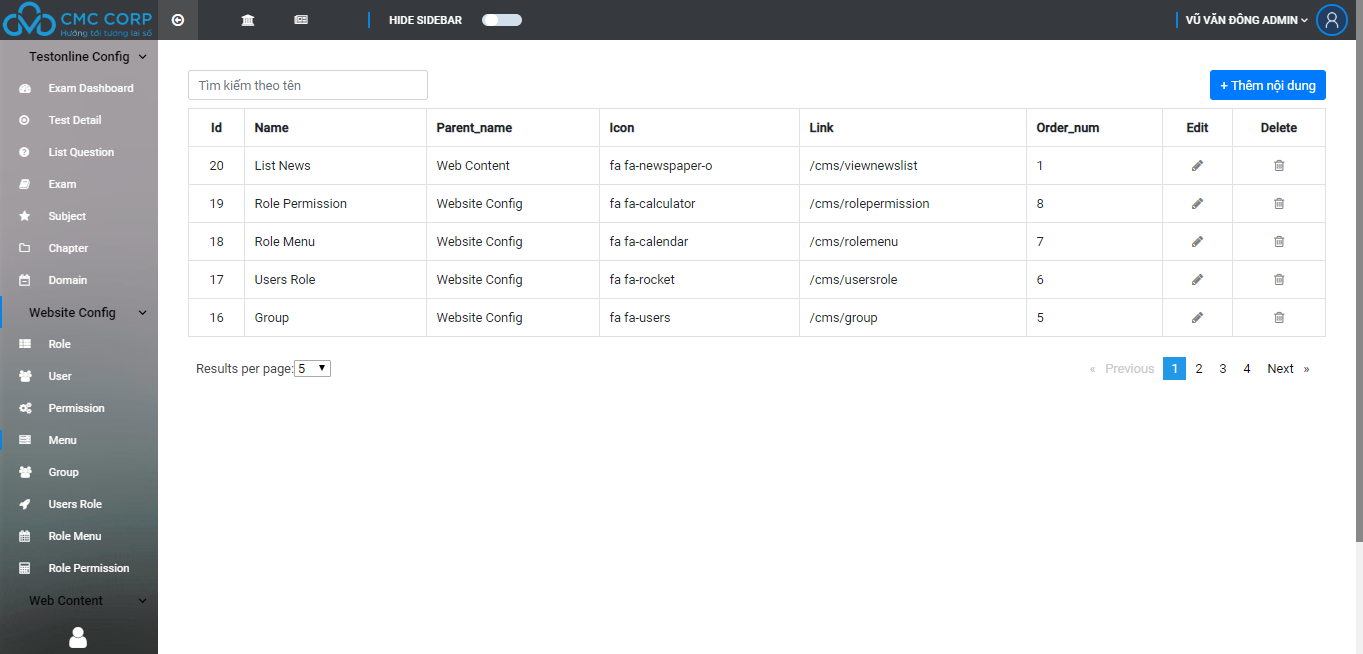
Hình 23: Giao diện trang quản lý domain

* + - 1. **Trang quản lý exam**

****

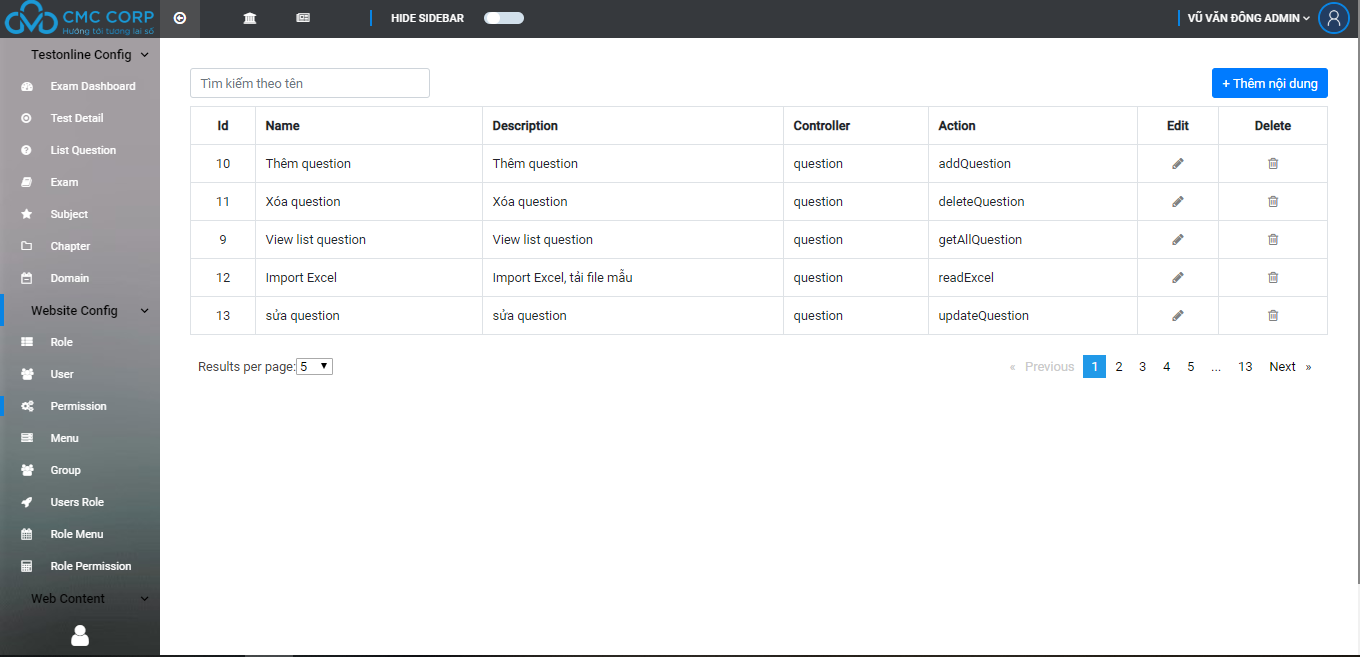
Hình 24: Giao diện trang quản lý exam

* + - 1. **Trang quản lý menu**

****

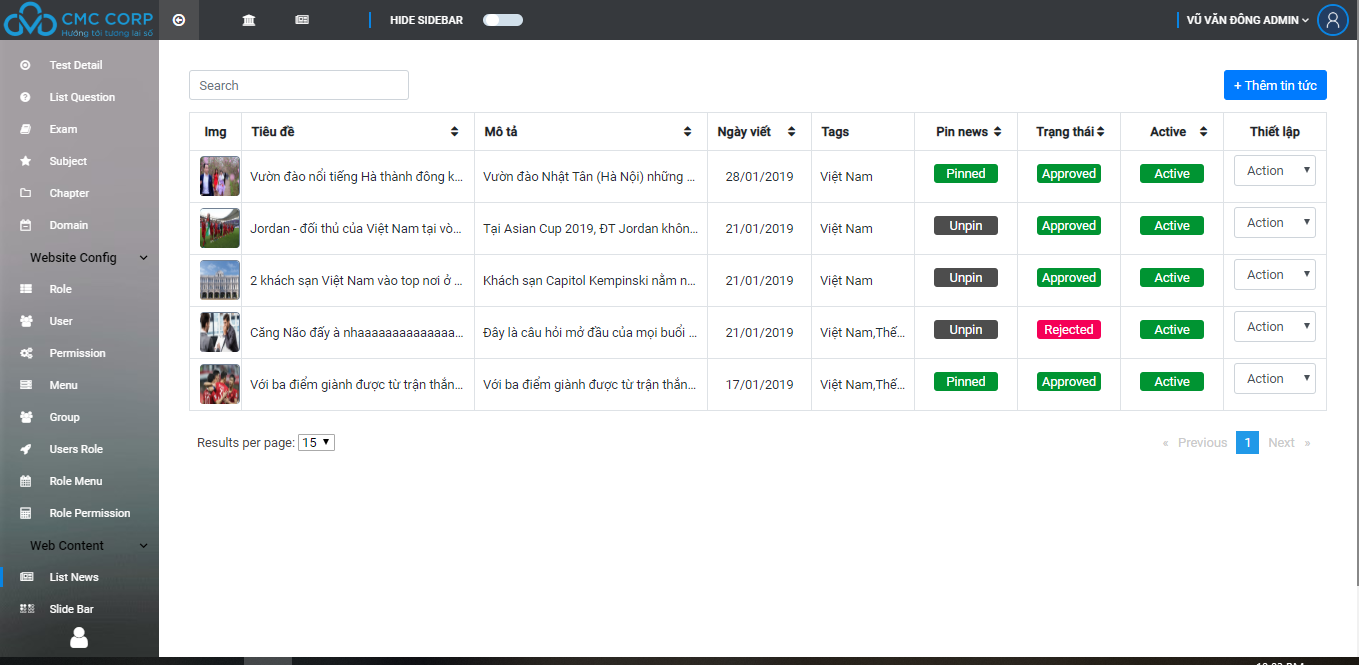
Hình 25: Giao diện trang quản lý menu

* + - 1. **Trang Quản lý permission**

****

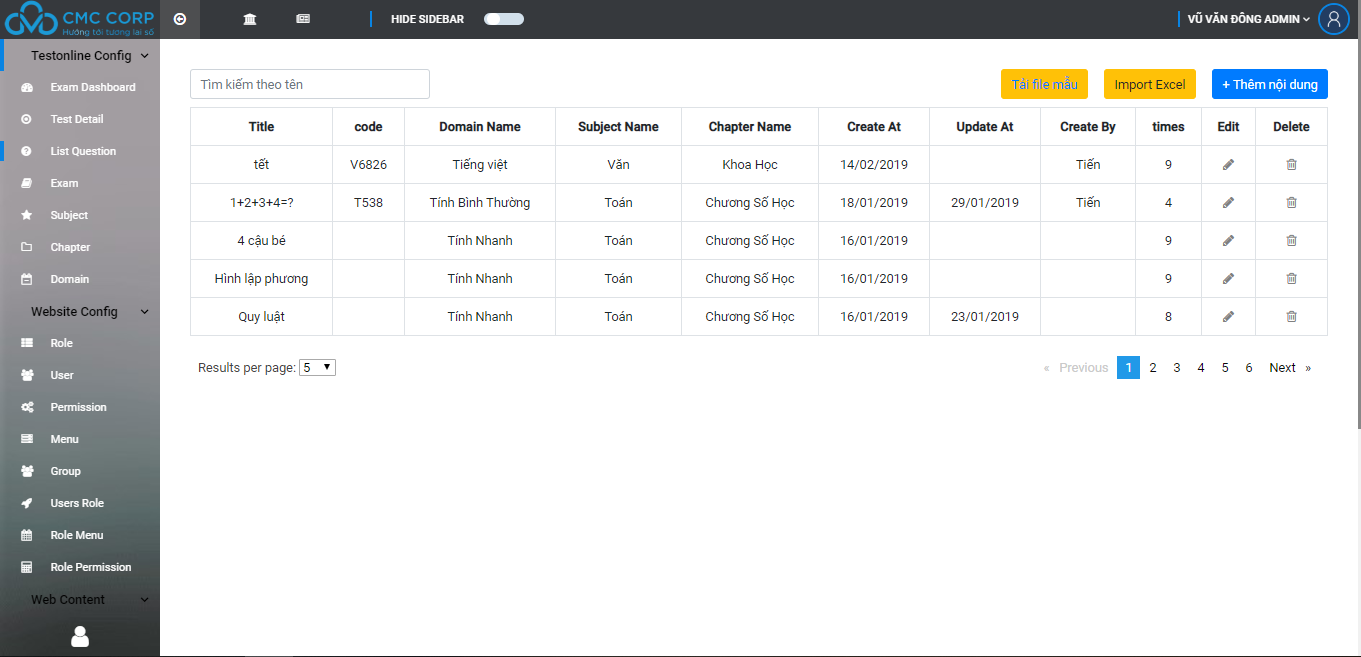
Hình 26: Giao diện trang quản lý permission

* + - 1. **Trang quản lý tin tức**

****

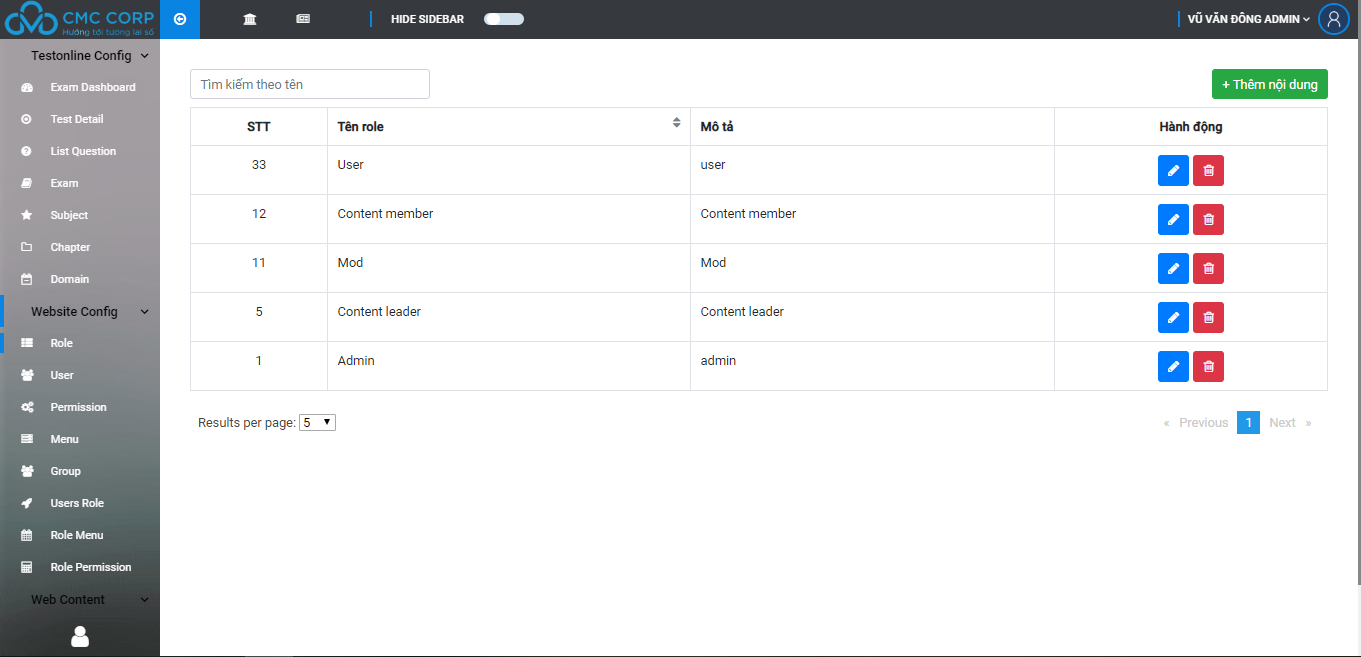
Hình 27: Giao diện quản lý tin tức

* + - 1. **Trang quản lý câu hỏi**

****

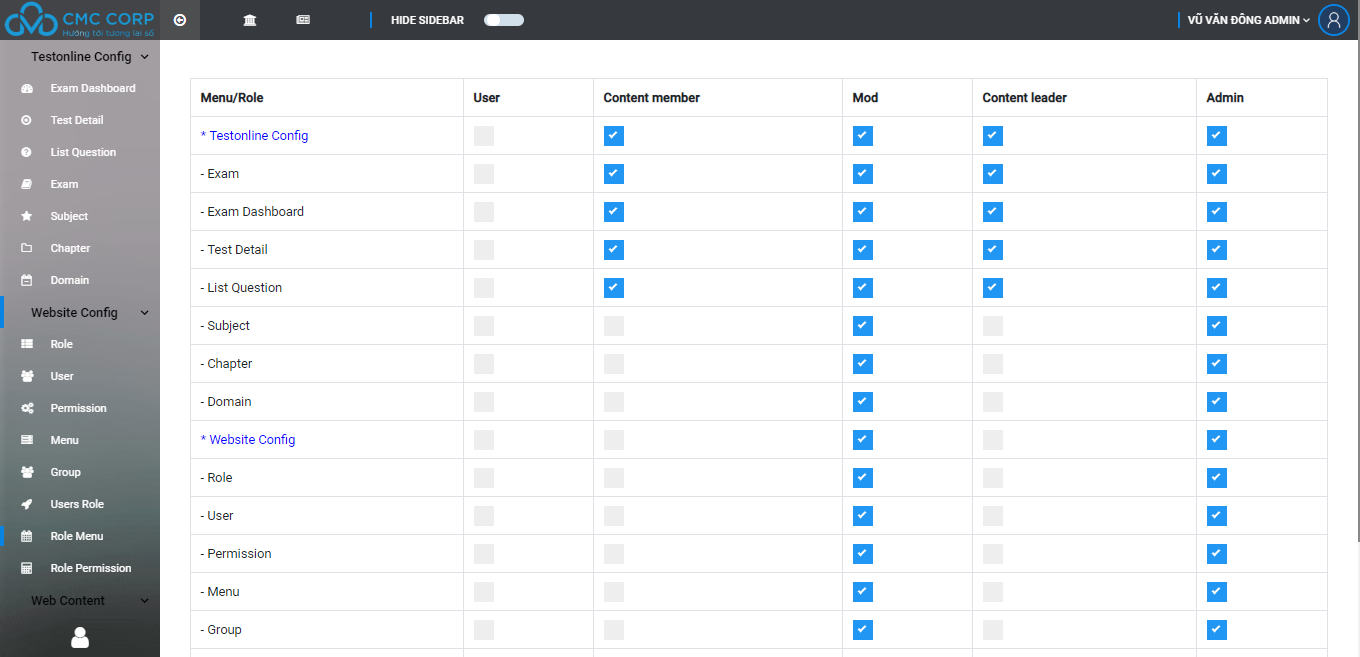
Hình 28: Giao diện quản lý câu hỏi

* + - 1. **Trang quản lý role**

****

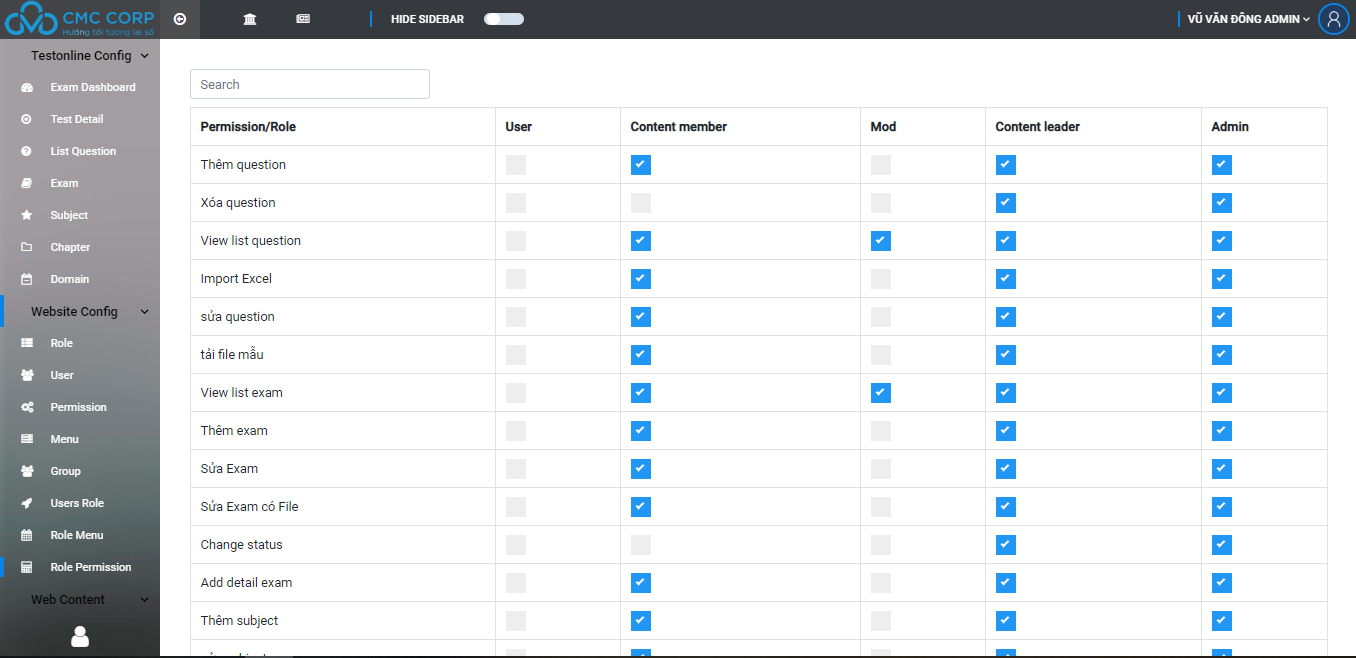
Hình 29: Giao diện quản lý role

* + - 1. **Trang quản lý role menu**

****

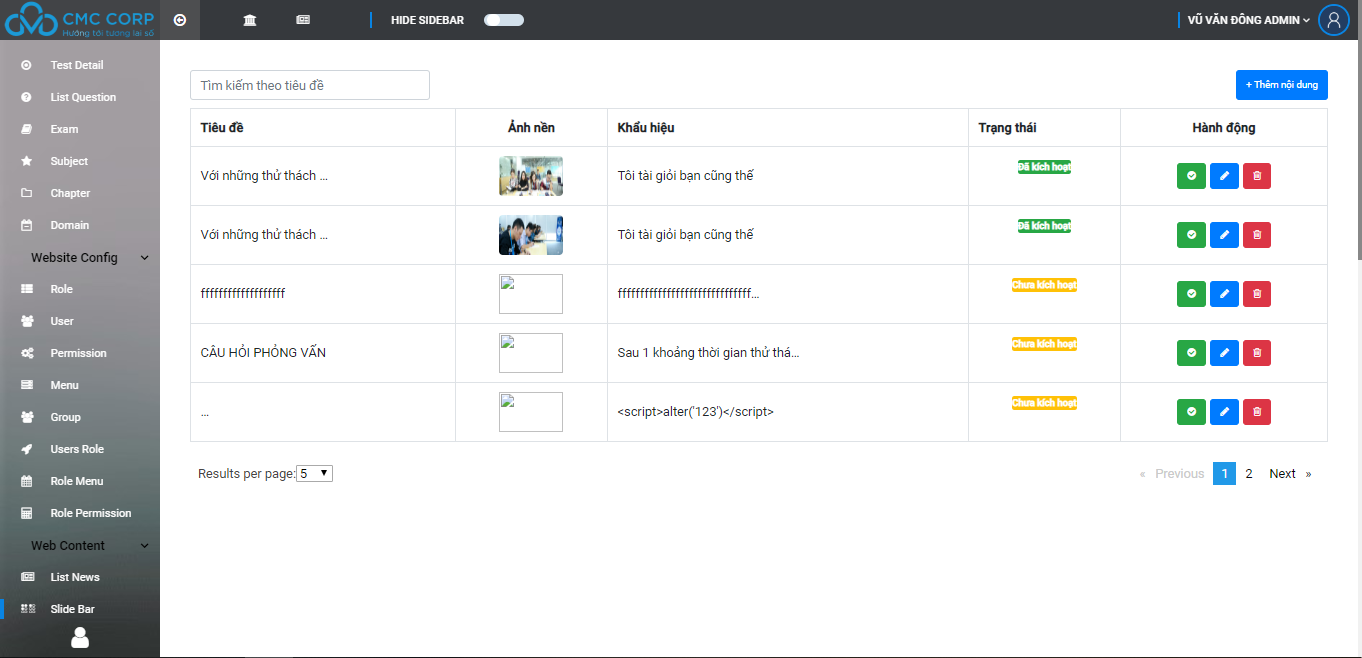
Hình 30: Giao diện quản lý role menu

* + - 1. **Trang quản lý role permission**

****

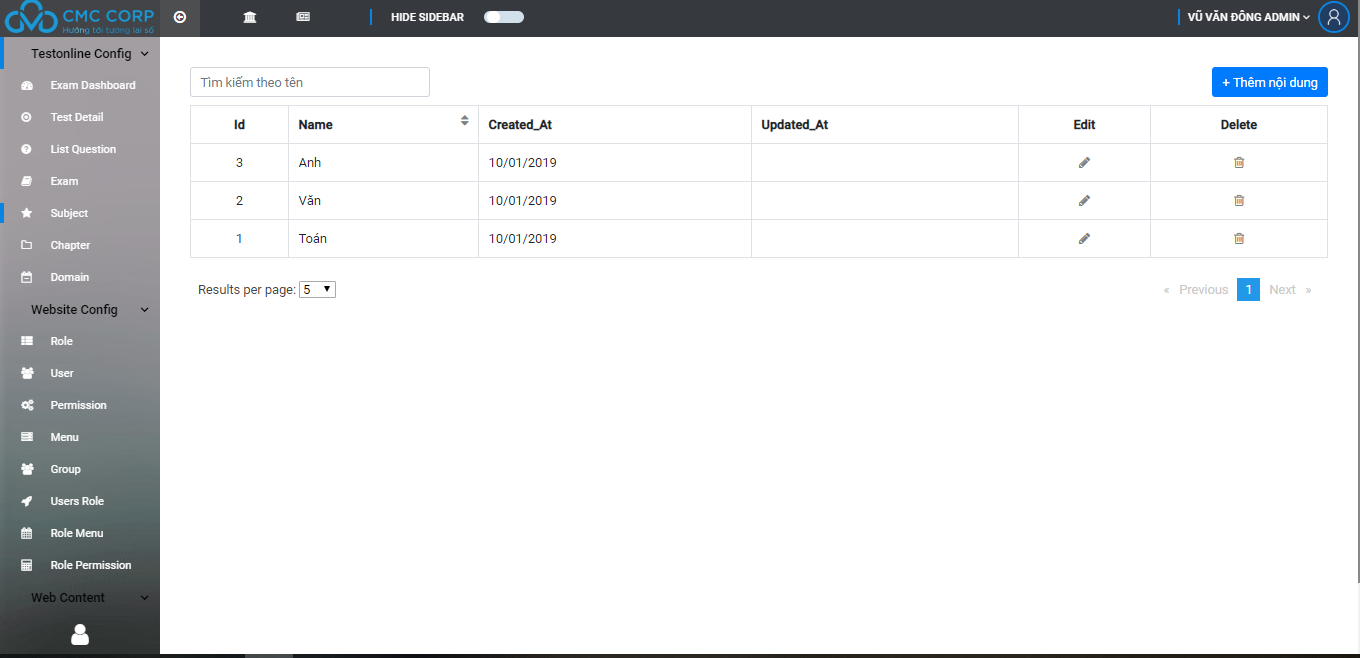
Hình 31: Giao diện quản lý role permission

* + - 1. **Trang quản lý slide bar**

****

Hình 32: Giao diện quản lý slide bar

* + - 1. **Trang quản lý môn học**

****

Hình 33: Giao diện quản lý môn học

# **KẾT LUẬN CHUNG**

Trong sự phát triển của công nghệ hiện nay, việc phát triển các ứng dụng web với nhiều ngôn ngữ, công nghệ hỗ trợ cho việc phát triển các ứng dụng web như Java web, PHP, NodeJs, ASP.net framework, ASP.net Core, … thì Java Spring Boot là một lựa chọn cho các công ty về công nghệ cũng như các lập trình viên web với những ưu điểm nổi bật như mã nguồn mở, cộng đồng hỗ trợ rộng lớn, ….

Sau quá trình thực hiện đồ án dưới sự hướng dẫn của ThS. Đoàn Văn Trung, em đã hoàn thành được một số công việc, hạn chế và định hướng cho những việc em chưa hoàn thành như sau:

**Kết quả đạt được:**

* Tìm hiểu và đưa ra được các khái niệm, nguyên lý hoạt động của các nền tảng công nghệ mới hiện nay như Java Spring Boot, Angular, JWT, Boostrap, …
* Phân tích và thiết kế hệ thống cho website thi trắc nghiệm trực tuyến.
* Xây dựng được website thi trắc nghiệm trực tuyến với một số chức năng như thi, quản lý người dùng, đề thi, câu hỏi, …

**Hạn chế:**

* Website vẫn chưa đáp ứng đủ một số yêu cầu như giao diện thân thiện và lượng truy cập người dùng lớn, …
* Một số chức năng đã xây dựng được nhưng vẫn còn hạn chế về mặt xử lý logic

**Hướng phát triển:**

* Xây dựng thêm các chức năng với đầy đủ tiện ích hỗ trợ người dùng.
* Xây dựng tính năng học trực tuyến
* …

# **TÀI LIỆU THAM KHẢO**

[1]. **Hướng dẫn lập trình Spring Boot cho người mới bắt đầu**

<https://o7planning.org/vi/11267/huong-dan-lap-trinh-spring-boot-cho-nguoi-moi-bat-dau>

[2]. <https://jwt.io/>

[3]. <https://viblo.asia/p/tim-hieu-ve-json-web-token-jwt-7rVRqp73v4bP>

[4]. [https://angular.io](https://angular.io/)

[5]. Một số tài liệu tham khảo khác.