**Lời nói đầu**

Ngày nay, chúng ta đang được sống trong kỷ nguyên của tin học nhờ sự vượt bậc, bùng nổ mạnh mẽ của công nghệ thông tin. Công nghệ thông tin không chỉ dừng lại ở mục đích phục vụ cho khoa học kỹ thuật mà còn đi sâu vào các lĩnh vực khác như đời sống, chính trị, kinh tế, xã hội,… nó trở nên thân thiện, gần gũi, mang lại nhiều lợi ích cho con người. Công nghệ thông tin ngày càng khẳng định được tính hữu dụng và sức mạnh trên mọi phương diện, mọi ngành nghề của cuộc sống, nhất là trong thời đại kinh tế thị trường như bây giờ.

Là một sinh viên ngành công nghệ thông tin, việc nắm được các kiến thức về các hệ thống quản trị cơ sở dữ liệu là vô cùng quan trọng. Vì vậy, em đã chọn 2 đề tài là “Xây dựng module lưu 1 bảng dữ liệu vào MongoDB và MySQL. So sánh tốc độ của 2 hệ thống này” và “Xây dựng module lưu 1 bảng dữ liệu vào PostgreSQL và MySQL. So sánh tốc độ của 2 hệ thống này”. Cùng với đó, xây dựng hệ thống thi trắc nhiệm online là 1 bài toán rất hay và thực tế, vì vậy em chọn đề tài thứ 3 là “Xây dựng hệ thống thi trắc nhiệm TOEIC để phục vụ thi thử tiếng anh trong học viện KTQS”

Em xin chân thành cảm ơn các thầy cô giáo khoa công nghệ thông tin trường Học Viện Kỹ Thuật Quân Sự đã hướng dẫn và tận tình giúp đỡ em trong quá trình học tập tại trường và trong thời gian thực tập này.

Với kinh nghiệm non nớt, cùng với kiến thức còn hạn chế, tuy đã cố gắng nỗ lực hết mình song báo cáo đề tài của em vẫn không tránh khỏi những thiếu sót. Em rất mong nhận được những đóng góp của Quý thầy cô và các bạn để rút kinh nghiệm và hoàn thiện hơn.

*Em xin chân thành cảm ơn!*

**Phụ lục**

[**I.** **Các đề tài thực tập** 3](#_Toc12922803)

[**II.** **Nguyên cứu và thu thập tài liệu** 3](#_Toc12922804)

[**III.** **Cơ sở lý thuyết** 4](#_Toc12922805)

[**1.** **MySQL là gì?** 4](#_Toc12922806)

[**2.** **MongoDB là gì?** 5](#_Toc12922807)

[**3.** **PostgreSQL là gì?** 5](#_Toc12922808)

[**4.** **Crawling?** 6](#_Toc12922809)

[**IV.** **Nội dung thực tập** 6](#_Toc12922810)

[**1.** **Đề tài 1:** Xây dựng module lưu 1 bảng dữ liệu vào MongoDB và MySQL. So sánh tốc độ của 2 hệ thống này. 6](#_Toc12922811)

[**2.** **Đề tài 2:** Xây dựng module lưu 1 bảng dữ liệu vào PostgreSQL và MySQL. So sánh tốc độ của 2 hệ thống này. 6](#_Toc12922812)

[**3.** **Đề tài 3:** Xây dựng hệ thống thi trắc nghiệm TOEIC để phục vụ thi thử tiếng Anh trong Học viện KTQS. 6](#_Toc12922813)

[**IV. Những khó khăn trong quá trình làm bài tập** 28](#_Toc12922823)

1. **Các đề tài thực tập**

**Đề tài 1:** Xây dựng module lưu 1 bảng dữ liệu vào MongoDB và MySQL. So sánh tốc độ của 2 hệ thống này.

Yêu cầu: Tạo 1 bảng dữ liệu, insert vào vài triệu bản ghi, đo thời gian thực hiện và so sánh.

**Đề tài 2:** Xây dựng module lưu 1 bảng dữ liệu vào PostgreSQL và MySQL. So sánh tốc độ của 2 hệ thống này.

Yêu cầu: Tạo 1 bảng dữ liệu, insert vào vài triệu bản ghi, đo thời gian thực hiện và so sánh.

**Đề tài 3:** Xây dựng hệ thống thi trắc nghiệm TOEIC để phục vụ thi thử tiếng Anh trong Học viện KTQS.

Yêu cầu: Crawling dữ liệu là các bài thi thử của TOEIC trên mạng để xây dựng ngân hàng đề thi thử ngữ pháp theo mẫu của TOEIC hiện nay.

1. **Nguyên cứu và thu thập tài liệu**

Trong thời gian làm project, em đã tự nghiên cứu, tìm tòi tài liệu về nhiều vấn đề khác nhau:

* Tìm hiểu về .NET Core.
* Tìm hiểu về ASP .NET
* Tìm hiểu về SQL server, MySQL, MongoDB, PostgreSQL.
* Tìm hiểu về html, css, javascript, jquery, json, ajax, boostrap 4, …
* Tìm hiểu về mô hình MVC.
* Deploy server local bằng IIS.
* Các yếu tố chính trong hệ thống thi online:

1. 1. Mục đích

* Xây dựng được hệ thống thi trực tuyến
* Quản lý việc đăng ký và thực hiện thi trực tuyến
* Quản lý kết quả thi
* Quản lý user
* Quản lý việc tạo câu hỏi, tạo bài thi trên hệ thống
  1. Phạm vi
* Đặc điểm phần mềm

Là phần mềm được sử dụng trong việc đăng ký thi trực tuyến, tạo bài thi và quản lý bài thi. Thiết bị sử dụng phải có kết nối internet và cần phải đăng kí tài khoản để thực hiện thi.

* Lĩnh vực ứng dụng

Ứng dụng cho mọi môn học từ các lĩnh vực: Khoa học tự nhiên, khoa học xã hội, công nghệ thông tin….

* Đối tượng của phần mềm
* Tất cả các đối tượng có nhu cầu thi trực tuyến, sử dụng thiết bị kết nối internet
* Đối tượng sử dụng phần mềm gồm có:

+ Admin: những Account có quyền quản lý câu hỏi, quản lý bài thi. Quản lý trainee, trainer và kết quả thi

+ Trainer: có quyền thêm, sửa, xóa câu hỏi, quản lý bài thi, xem kết quả

+ Trainee: Đăng ký account để đăng ký thi, thực hiện thi, theo dõi kết quả bài thi của chính mình

1. **Cơ sở lý thuyết**

## **MySQL là gì?**

MySQL là một hệ thống quản lý cơ sở dữ liệu quan hệ mã nguồn mở (RDBMS) đầy đủ tính năng, ban đầu được xây dựng bởi MySQL AB và hiện đang thuộc sở hữu của Tập đoàn Oracle. Nó Lưu trữ dữ liệu trong các bảng cái mà được nhóm vào cơ sở dữ liệu, sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL) để truy cập dữ liệu và các lệnh như ‘SELECT’, UPDATE’, ‘INSERT’ và ‘DELETE’ để quản lý .Các thông tin liên quan có thể lưu trữ ở bảng khác, nhưng việc sử dụng các hoạt động JOIN cho phép bạn tương quan với nó, thực hiện các truy vấn trên các bảng khác nhau và giảm thiểu khả năng trùng lặp dữ liệu.

MySQL tương thích với gần như tất cả các hệ điều hành, cụ thể là Windows, Linux, Unix, Apple, FreeBSD và nhiều hệ điều hành khác…Nó hỗ trợ các công cụ lưu trữ khác nhau như InnoDB, Federated, MyISAM, Memory, CSV, Archive, Blackhole và Merge.

## **MongoDB là gì?**

MongoDB là một cơ sở dữ liệu hướng tài liệu mã nguồn mở phổ biến được phát triển bởi 10gen, sau này được gọi là MongoDB Inc. Trong trường hợp này, các tài liệu được tạo và lưu trữ trong các tệp BSON, định dạng Binary JSON (JavaScript Object Notation),vì vậy tất cả các loại dữ liệu JS là được hỗ trợ. Trong trường hợp đó, MongoDB thường được áp dụng cho các dự án Node.js. Ngoài ra, JSON cho phép chuyển dữ liệu giữa các máy chủ và ứng dụng web bằng cách sử dụng định dạng có thể đọc được. Nó cũng là một lựa chọn tốt hơn, khi nói đến dung lượng và tốc độ lưu trữ, vì nó mang lại hiệu quả và độ tin cậy cao hơn.

Một trong những lợi ích hàng đầu được MongoDB cung cấp là việc sử dụng các lược đồ động giúp loại bỏ nhu cầu xác định trước cấu trúc, như các trường hoặc các loại giá trị. Mô hình như vậy cho phép biểu diễn mối quan hệ phân cấp, lưu trữ mảng và khả năng thay đổi cấu trúc bản ghi bằng cách thêm hoặc xóa các trường. Giải pháp NoQuery này đi kèm với nhúng, auto-sharding, và on-board replication để có khả năng mở rộng tốt hơn và tính sẵn sàng cao.

## **PostgreSQL là gì?**

PostgreSQL là một hệ thống quản trị cơ sở dữ liệu quan hệ-đối tượng (object-relational database management system) có mục đích chung, hệ thống cơ sở dữ liệu mã nguồn mở tiên tiến nhất hiện nay.

PostgreSQL được phát triển dựa trên POSTGRES 4.2 tại phòng khoa học máy tính Berkeley, Đại học California.

PostgreSQL được thiết kế để chạy trên các nền tảng tương tự UNIX. Tuy nhiên, PostgreSQL sau đó cũng được điều chỉnh linh động để có thể chạy được trên nhiều nền tảng khác nhau như MacOS X, Solaris và Windows.

PostgreSQL là một phần mềm mã nguồn mở miễn phí. Mã nguồn của phần mềm khả dụng theo license của PostgreSQL, một license nguồn mở tự do. Theo đó, bạn sẽ được tự do sử dụng, sửa đổi và phân phối PostgreSQL dưới mọi hình thức.

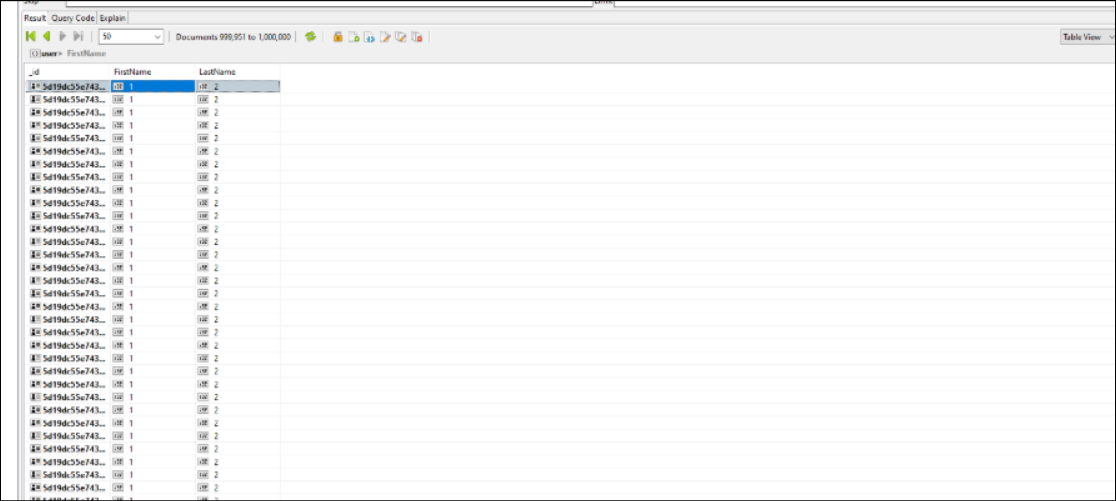
PostgreSQL không yêu cầu quá nhiều công tác bảo trì bởi có tính ổn định cao. Do đó, nếu bạn phát triển các ứng dụng dựa trên PostgreSQL, chi phí sở hữu sẽ thấp hơn so với các hệ thống quản trị dữ liệu khác.

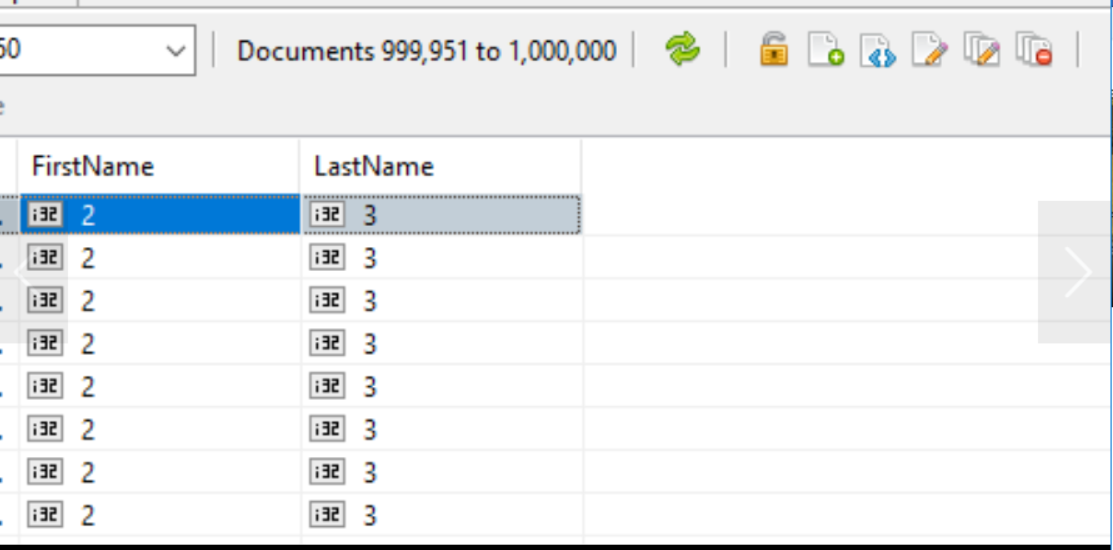
## **Crawling?**

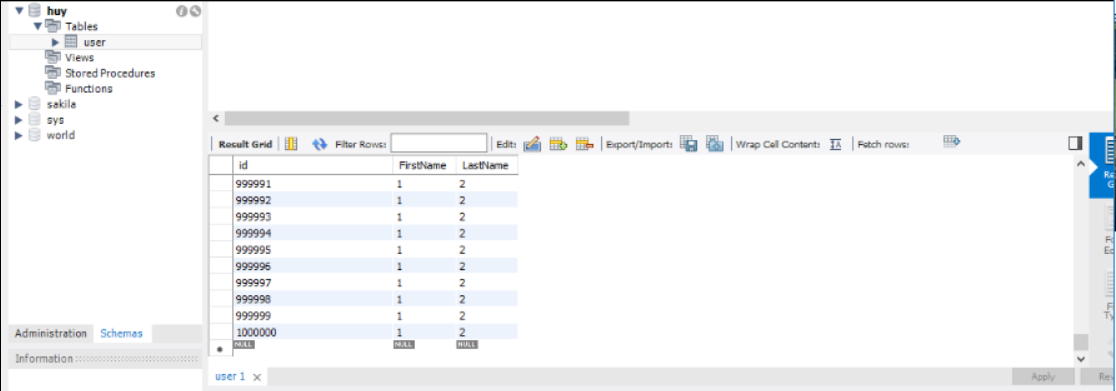
Lấy tin tự động (crawl) hiểu đơn giản là chức năng lấy tất cả những thông tin bạn cần từ các nguồn thông tin sau đó xử lý lưu trữ lại cho mục đích của bạn (làm dữ liệu phân tích, đăng tải lên website, chia sẻ lên data center…).

1. **Nội dung thực tập**

**-**Database sử dụng cho đề tài 1 và 2:

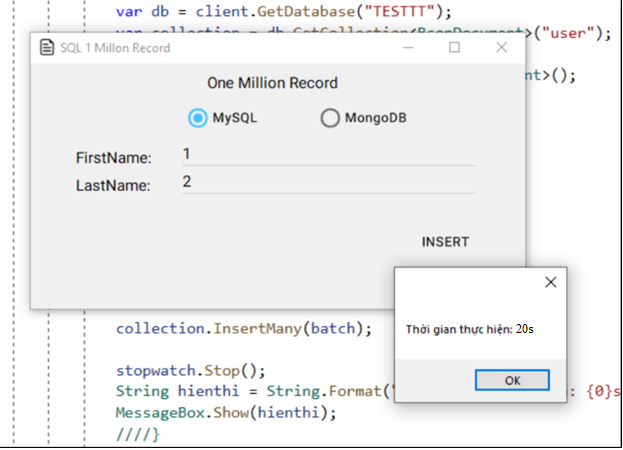




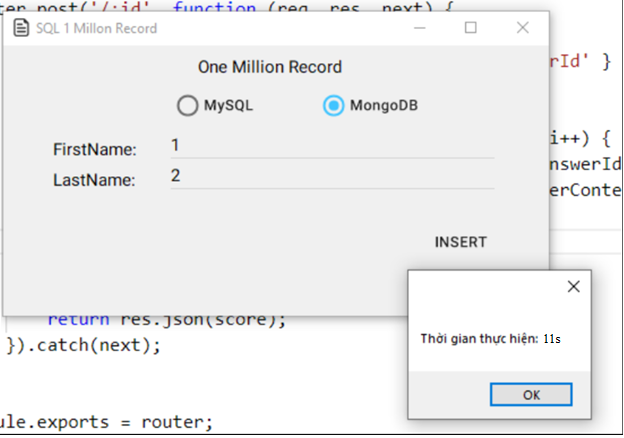


1. **Đề tài 1:** Xây dựng module lưu 1 bảng dữ liệu vào MongoDB và MySQL. So sánh tốc độ của 2 hệ thống này.

Chú ý: Việc đẩy 1 triệu bản ghi vào CSDL MongoDB và MySQL nhanh hay chậm bao nhiêu còn tùy thuộc vào hiệu năng từng máy.



Hệ CSDL MySQL: thời gian 20 giây

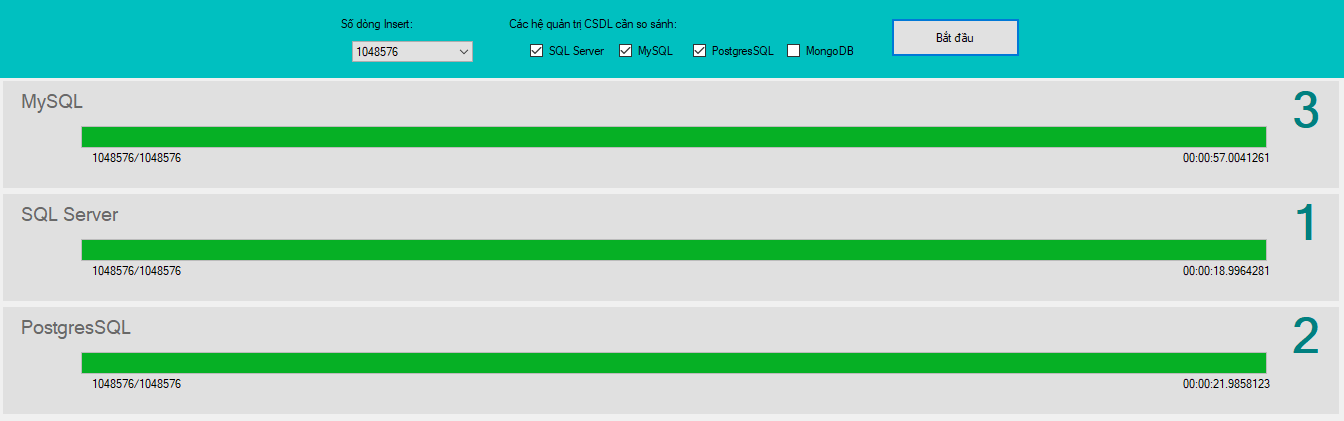


Hệ CSDL MongoDB: thời gian 11 giây

**Kết luận:** MongoDB đẩy 1 triệu bản ghi nhanh hơn MySQL.

1. **Đề tài 2:** Xây dựng module lưu 1 bảng dữ liệu vào PostgreSQL và MySQL. So sánh tốc độ của 2 hệ thống này.

Chú ý: Việc đẩy 1 triệu bản ghi vào CSDL PostgreSQL và MySQL nhanh hay chậm bao nhiêu còn tùy thuộc vào hiệu năng từng máy.



Hệ CSDL MySQL: thời gian 57 giây

Hệ CSDL PostgreSQL: thời gian 21 giây

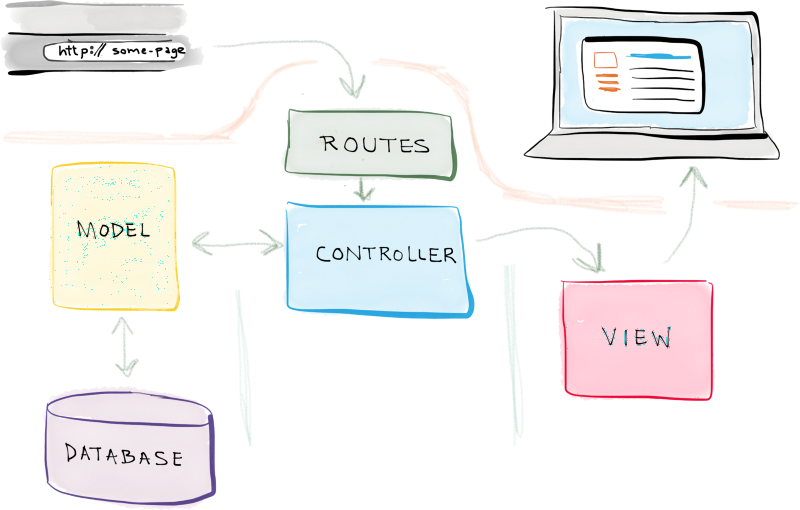
**Kết luận:** PostgreSQL đẩy 1 triệu bản ghi nhanh hơn MySQL.

1. **Đề tài 3:** Xây dựng hệ thống thi trắc nghiệm TOEIC để phục vụ thi thử tiếng Anh trong Học viện KTQS.

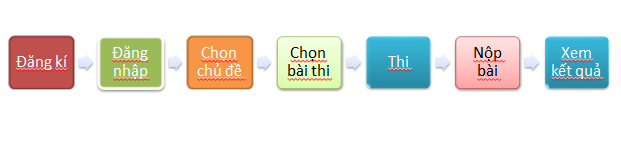
**Giới thiệu chung:**

* Nền tảng ứng dụng: ASP .NET
* Phiên bản Framework: .NET framework 4.6.1
* Mô hình: MVC

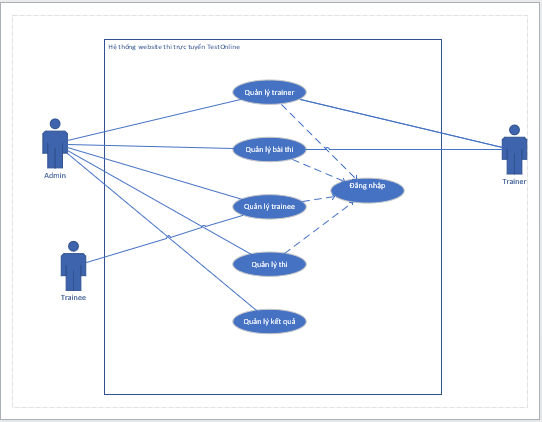
**Mô hình MVC:**



**Nghiệp vụ cơ bản:**



**Mô hình Usecase:**



**Danh sách user và mô tả:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tên Use Case | Mô tả ngắn gọn Use case | Vai trò trong hệ thống | Ghi chú |
| UC\_01 | Admin | Account được phân quyền với vai trò là Admin, được phép truy cập và sử dụng tất cả các chức năng của hệ thống. | Quản lý kiểm soát hoạt động của user trong hệ thống. Quản lý việc tạo câu hỏi, đề thi,bài thi, quản lý kết quả thi của user |  |
| UC\_02 | Trainer | Account được phân quyền với vai trò là Trainer. Được phép truy cập để tạo,sửa,xóa câu hỏi,đề thi, bài thi và được quyền thực hiện bài thi | Quản lý việc tạo đề thi bài thi. Chịu sự quản lý từ Admin |  |
| UC\_03 | Trainee | Đăng ký account để truy cập hệ thống. Thực hiện việc đăng ký bài thi, vào thi và xem kết quả thi | Quản lý thông tin tài khoản của chính người dùng. Thực hiện việc thi |  |

**Chức năng chi tiết của từng nhóm:**

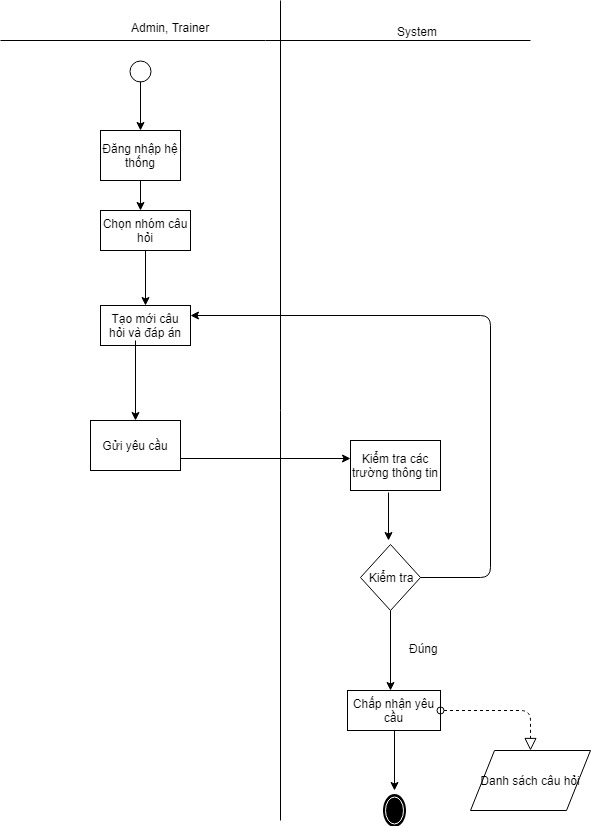
1. Admin
2. Quản lý tài khoản

|  |  |
| --- | --- |
| Use case | UC\_01 |
| Mục đích | Quản lý và kiểm soát hoạt động của các tài khoản user trong hệ thống |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Admin |
| Luồng sự kiện chính | 1. Chọn xem thông tin tài khoản 2. Phân quyền cho user. 3. Theo dõi trạng thái hoạt động của user: chỉ xem 4. Theo dõi lịch sử hoạt động của user: chỉ xem 5. Tìm kiếm tài khoản theo các tiêu chí: email, sđt, ngày đki… |
| Luồng sự kiện phụ | 1. Phân quyền: hệ thống gồm 2 quyền: trainer và trainee. User đăng kí mặc định ở quyền trainee , việc phân quyền do admin chọn 2. Admin có thể tìm kiếm theo textbox tìm kiếm hoặc filter |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

1. Quản lý đề thi

|  |  |
| --- | --- |
| Use case | UC\_01 |
| Mục đích | Quản lý các thông tin liên quan đến đề thi |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Admin |
| Luồng sự kiện chính | Trong phần quản trị bài thi gồm các sự kiện sau:   1. Tạo mới câu hỏi- câu hỏi phải có đáp án 2. Update câu hỏi-đáp án. 3. Xóa câu hỏi. 4. Tạo mới chủ để 5. Update chủ đề 6. Xóa chủ đề 7. Tạo bài thi 8. Sửa bài thi 9. Xóa bài thi 10. Xem lịch sử sửa chữa   Nhấn submit để thực hiện các chức năng đã làm trước đó |
| Luồng sự kiện phụ | 1. Đáp án có thể có 1 hoặc nhiều đáp án. Số lượng đáp án đúng phải nhỏ hơn tổng số đáp án của câu đó 2. Câu hỏi được update khi chưa thi 3. Xóa câu hỏi khi câu đó chưa thuộc đề thi nào. Nếu không muốn cho nó hiển thị có thể chuyển trạng thái disable 4. Chủ đề tạo mới không được trùng với chủ đề trc đó 5. Sửa bài thi: chỉ được sửa số lượng câu hỏi, thời gian… không được sửa nội dung câu hỏi trong đề 6. Bài thi được xóa chỉ khi bài đó chưa từng được thi |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

* Mô hình hoạt động : Tạo mới câu hỏi



1. Chọn bài thi

|  |  |
| --- | --- |
| Use case | UC\_01 |
| Mục đích | Theo dõi tình trạng các bài thi được chọn bởi user |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Admin |
| Luồng sự kiện chính | Vào trang chọn bài thi: sẽ có 2 chức năng tìm kiếm và xem số lượng account đã chọn bài thi. |
| Luồng sự kiện phụ |  |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

* 1. Thực hiện thi

|  |  |
| --- | --- |
| Use case | UC\_01 |
| Mục đích | Theo dõi lịch sử thi |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Admin |
| Luồng sự kiện chính | Admin vào phần thi để xem được tại thời điểm này có những bài thi nào đang được thi. |
| Luồng sự kiện phụ |  |
| Điều kiện sau: |  |

* 1. Quản lý kết quả thi

|  |  |
| --- | --- |
| Use case | UC\_01 |
| Mục đích | Quản lý kết quả thi của các user trong hệ thống |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Admin |
| Luồng sự kiện chính | Vào trang kết quả thi:   1. Hiển thị ra kết quả chi tiết theo các tiêu chí mà Admin chọn. Ví dụ hiển thị ra theo: email, thời gian thi, chủ đề thi,… 2. Thống kê: click thống kê , lựa chọn sơ đồ muốn hiển thị. Thống kê kết quả của user theo chủ đề thi biểu diễn kết quả user theo sơ đồ. |
| Luồng sự kiện phụ |  |
| Điều kiện sau: |  |

* 1. Quản lý nhóm học tập

|  |  |
| --- | --- |
| Use case | UC\_01 |
| Mục đích | Theo dõi tình trạng các bài thi được chọn bởi user |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Trainer |
| Luồng sự kiện chính | Vào trang chọn bài thi: sẽ có 2 chức năng tìm kiếm và xem số lượng account đã chọn bài thi. |
| Luồng sự kiện phụ |  |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

1. Trainer
   1. Quản lý tài khoản

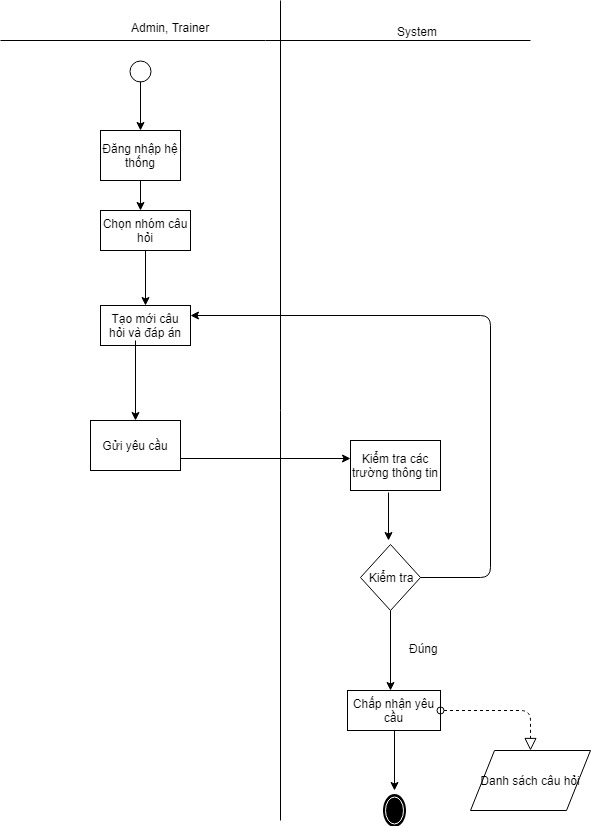
|  |  |
| --- | --- |
| Use case | UC\_02 |
| Mục đích | Quản lý tài khoản của chính mình |
| Điều kiện trước: | * Truy cập vào trang web |
| Luồng sự kiện chính | Vào trang thông tin user cập nhật:   1. Đăng kí tài khoản(mục 1 phần A) 2. Thông tin cá nhân 3. Đổi mật khẩu: click chọn đổi password-> điền pass mới -> xác nhận lại pass-> submit |
| Luồng sự kiện phụ | 1. Điền không khớp pass muốn đổi sẽ báo lỗi |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

* 1. Quản lý đề thi

|  |  |
| --- | --- |
| Use case | UC\_02 |
| Mục đích | Quản lý các thông tin liên quan đến đề thi |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Trainer |
| Luồng sự kiện chính | Trong phần quản trị bài thi gồm các sự kiện sau:   1. Tạo mới câu hỏi- câu hỏi phải có đáp án 2. Update câu hỏi-đáp án. 3. Xóa câu hỏi. 4. Tạo mới chủ để 5. Update chủ đề 6. Xóa chủ đề 7. Tạo bài thi 8. Sửa bài thi 9. Xóa bài thi   Nhấn submit để thực hiện các chức năng đã làm trước đó |
| Luồng sự kiện phụ | 1. Đáp án có thể có 1 hoặc nhiều đáp án. Số lượng đáp án đúng phải nhỏ hơn tổng số đáp án của câu đó 2. Câu hỏi được update khi chưa thi 3. Xóa câu hỏi khi câu đó chưa thuộc đề thi nào. Nếu không muốn cho nó hiển thị có thể chuyển trạng thái disable 4. Chủ đề tạo mới không được trùng với chủ đề trc đó 5. Sửa bài thi: chỉ được sửa số lượng câu hỏi, thời gian… không được sửa nội dung câu hỏi trong đề 6. Bài thi được xóa chỉ khi bài đó chưa từng được thi |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

* Mô hình activity:

Tạo mới câu hỏi:



* 1. Chọn bài thi

|  |  |
| --- | --- |
| Use case | UC\_02 |
| Mục đích | Chọn bài thi để test thử -> điều chỉnh lại nội dung hoặc thời gian cho phù hợp |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Trainer |
| Luồng sự kiện chính | Vào trang Thi thực hiện các bước sau:   1. Chọn loại môn học(bắt buộc, tự chọn, tổng hợp…) 2. Chọn môn học liên quan đến loại môn học đó 3. Chọn bài thi 4. vào thi hoặc hủy thi |
| Luồng sự kiện phụ | 1. Khi click vào button vào thi sẽ hiển thị thông báo có chắc chắn vào thi hay không? Nếu không vào thi thì user vẫn ở lại màn hình cũ 2. Chọn kết quả: được phép chọn 1 hoặc nhiều kết quả. Tuy nhiên kết quả chọn phải nhỏ hơn số kết quả đưa ra của câu. Sử dụng checkbox thì đc chọn nhiều câu trả lời. dùng radiobutton chỉ chọn 1 đáp án 3. Nộp bài có 2 hình thức:   + User tự nộp bài khi chưa hết thời gian: hiển thị thông báo có chắc chắn nộp bài? Hoặc nếu có câu hỏi mà user chưa trả lời sẽ hiển thị thông báo để báo cho user biết điều đó  + Hết giờ hệ thống tự động nộp bài. |
| Điều kiện sau: | Nếu chọn vào thi thì màn hình sẽ vào trang để thi. Còn nếu hủy thi thì sẽ ở lại trang danh sách bài thi đã chọn |

* 1. Thực hiện thi

|  |  |
| --- | --- |
| Use case | UC\_02 |
| Mục đích | Thực hiện thi |
| Điều kiện trước: | * Truy cập vào trang web với tư cách trainer |
| Luồng sự kiện chính | Tại màn hình danh sách các bài thi đã chọn click vào button”vào thi”. Chọn đồng ý để vào thi gồm các bước sau:   1. Xem nội dung bài thi 2. Chọn đáp án 3. Nộp bài 4. Chờ kết quả |
| Luồng sự kiện phụ | 1. Chọn kết quả: được phép chọn 1 hoặc nhiều kết quả. Tuy nhiên kết quả chọn phải nhỏ hơn số kết quả đưa ra của câu. Sử dụng checkbox thì đc chọn nhiều câu trả lời. dùng radiobutton chỉ chọn 1 đáp án 2. Nộp bài có 2 hình thức:   + User tự nộp bài khi chưa hết thời gian: hiển thị thông báo có chắc chắn nộp bài? Hoặc nếu có câu hỏi mà user chưa trả lời sẽ hiển thị thông báo để báo cho user biết điều đó  + Hết giờ hệ thống tự động nộp bài. |
| Điều kiện sau: | Màn hình về trang xem kết quả |

* 1. Xem kết quả

|  |  |
| --- | --- |
| Use case | UC\_02 |
| Mục đích | Xem kết quả của các user đăng kí thi trong hệ thống |
| Điều kiện trước: | * Truy cập vào trang web với tư cách Trainer |
| Luồng sự kiện chính | Vào trang kết quả thi:   1. Xem kết quả thi của chính mình 2. Xem kết quả thi của các user theo các tiêu chí có thể chọn |
| Luồng sự kiện phụ | Kết quả được hiển thị theo thứ tự thời gian từ gần đến xa. |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

1. Trainee
   1. Quản lý tài khoản

|  |  |
| --- | --- |
| Use case | UC\_03 |
| Mục đích | Quản lý tài khoản của chính mình |
| Điều kiện trước: | * Truy cập vào trang web |
| Luồng sự kiện chính | Vào trang thông tin user cập nhật:   1. Đăng kí tài khoản(mục 1 phần A) 2. Thông tin cá nhân 3. Đổi mật khẩu: click chọn đổi password-> điền pass mới -> xác nhận lại pass-> submit |
| Luồng sự kiện phụ | 1. Điền không khớp pass muốn đổi sẽ báo lỗi |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

* 1. Chọn nhóm học tập

|  |  |
| --- | --- |
| Use case | UC\_03 |
| Mục đích | Chọn nhóm để thực hiện việc thi theo chủ đề |
| Điều kiện trước: | * Truy cập vào trang web |
| Luồng sự kiện chính | Trong quá trình đăng kí hệ thống, user đã phải chọn nhóm học tập. User được chọn nhiều nhóm học tập và có quyền được thay đổi nhóm:   1. Vào trang Nhóm học tập 2. Thay đổi việc chọn nhóm 3. Lưu lại |
| Luồng sự kiện phụ | 1. Việc thay đổi nhóm sẽ được thực hiện khi đã đăng nhập hệ thống 2. Được phép chọn nhiều nhóm hoặc bỏ bớt nhóm. Điều kiện: mỗi user bắt buộc phải thuộc tối thiểu là 1 nhóm học tập |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

6.3 Chọn bài thi

|  |  |
| --- | --- |
| Use case | UC\_03 |
| Mục đích | Chọn bài thi để test thử -> điều chỉnh lại nội dung hoặc thời gian cho phù hợp |
| Điều kiện trước: | * Truy cập vào trang web |
| Luồng sự kiện chính | Vào trang Thi thực hiện các bước sau:   1. Chọn loại môn học(bắt buộc, tự chọn, tổng hợp…) 2. Chọn môn học liên quan đến loại môn học đó 3. Vào thi 4. Chọn kết quả 5. Nộp bài |
| Luồng sự kiện phụ | 1. Khi click vào button vào thi sẽ hiển thị thông báo có chắc chắn vào thi hay không? 2. Chọn kết quả: được phép chọn 1 hoặc nhiều kết quả. Tuy nhiên kết quả chọn phải nhỏ hơn số kết quả đưa ra của câu. Sử dụng checkbox thì đc chọn nhiều câu trả lời. dùng radiobutton chỉ chọn 1 đáp án 3. Nộp bài có 2 hình thức:   + User tự nộp bài khi chưa hết thời gian: hiển thị thông báo có chắc chắn nộp bài? Hoặc nếu có câu hỏi mà user chưa trả lời sẽ hiển thị thông báo để báo cho user biết điều đó  + Hết giờ hệ thống tự động nộp bài. |
| Điều kiện sau: | Nộp bài thành công màn hình trả về kết quả bài thi. |

6.4 Thực hiện thi

|  |  |
| --- | --- |
| Use case | UC\_03 |
| Mục đích | Thực hiện thi |
| Điều kiện trước: | * Truy cập vào trang web |
| Luồng sự kiện chính | Tại màn hình danh sách các bài thi đã chọn click vào button”vào thi”. Chọn đồng ý để vào thi gồm các bước sau:   1. Xem nội dung bài thi 2. Chọn đáp án 3. Nộp bài 4. Chờ kết quả |
| Luồng sự kiện phụ | 1. Chọn kết quả: được phép chọn 1 hoặc nhiều kết quả. Tuy nhiên kết quả chọn phải nhỏ hơn số kết quả đưa ra của câu. Sử dụng checkbox thì đc chọn nhiều câu trả lời. dùng radiobutton chỉ chọn 1 đáp án 2. Nộp bài có 2 hình thức:   + User tự nộp bài khi chưa hết thời gian: hiển thị thông báo có chắc chắn nộp bài? Hoặc nếu có câu hỏi mà user chưa trả lời sẽ hiển thị thông báo để báo cho user biết điều đó  + Hết giờ hệ thống tự động nộp bài. |
| Điều kiện sau: | Màn hình về trang xem kết quả |

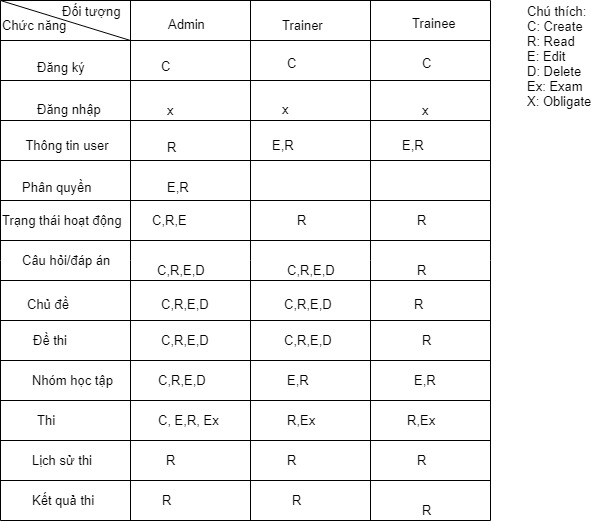
* 1. Xem kết quả

|  |  |
| --- | --- |
| Use case | UC\_03 |
| Mục đích | Xem kết quả của các user đăng kí thi trong hệ thống |
| Điều kiện trước: | * Truy cập vào trang web |
| Luồng sự kiện chính | Vào trang kết quả thi:  Xem kết quả thi của chính mình |
| Luồng sự kiện phụ | Kết quả được hiển thị theo thứ tự thời gian từ gần đến xa. |
| Điều kiện sau: | Thực hiện các hoạt động mong muốn |

1. Các yêu cầu đối với hệ thông

|  |  |  |
| --- | --- | --- |
| Chức năng | Mô tả | Yêu cầu |
| Th Chỉnh sửa, tạo câu hỏi | Câu hỏi gồm các trường bắt buộc:   1. Tiêu đề 2. Nội dung 3. Đáp án 4. Đáp án đúng 5. Thời gian làm | Bắt buộc điền thời gian dự kiến của từng câu. Để khi tạo đề thi có thể tính được tổng thời gian dự kiến cho đề thi đó-> thuận tiện trong việc căn chỉnh thời gian |
| Hiển thị câu hỏi trong bài thi | Có list câu hỏi ở góc bên phải/trái màn hình. Click vào câu hỏi bên list đó có để chuyển câu hỏi. Những câu đã trả lời sẽ hiển thị màu khác so với những câu chưa trả lời | Chỉ hiển thị nội dung của từng câu 1. |
| Next/previous | Trong list câu hỏi để user nhấn chuyển câu. |  |
| Thời gian | Thời gian của đề thi được đặt bởi người tạo. có thể hiển thị thông báo theo từng mốc thời gian còn lại trên màn hình để user biết |  |
| Lịch sử thi , kết quả thi | Xem lịch sử thi và kết quả thi | Các đối tượng chỉ có quyền xem. |
| Đăng nhập 1 tài khoản trên nhiều ứng dụng cùng lúc | Sửa dụng cùng 1 lúc 1 tài khoản trên nhiều trình duyệt khác nhau. ví dụ: coccoc,chrome | Không cho phép |
| Phân trang | Phân trang nếu như dữ liệu hiển thi ra màn hình nhiều. Số lượng dữ liệu trong mỗi trang dev tự căn chỉnh |  |

1. Ma trận phân quyền

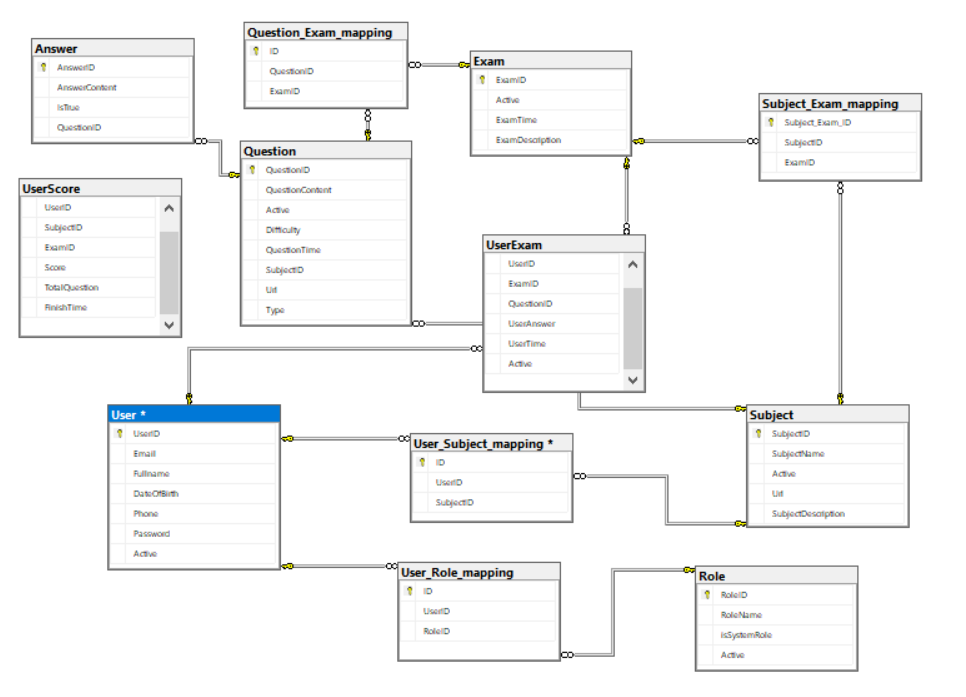


1. Yêu cầu phi chức năng

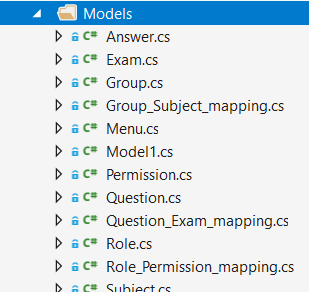
* Độ bảo mật cao
* Sử dụng được nhiều ngôn ngữ
* Giao diện thân thiện với người dùng
* Tốc độ truy cập nhanh
* Thông tin trên website chính xác

**Mô hình MVC trong Project:**

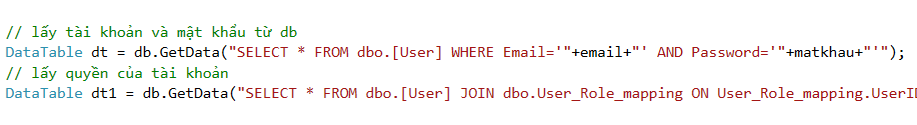
* Database:



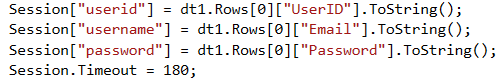
* Model: ADO .NET Code First.



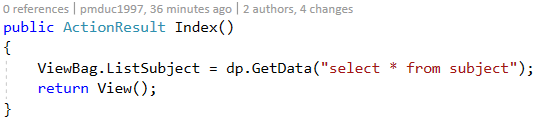
* Controller + Model: Sử dụng câu lệnh truy vấn SQL để lấy dữ liệu hoặc truyền dữ liệu vào Model.



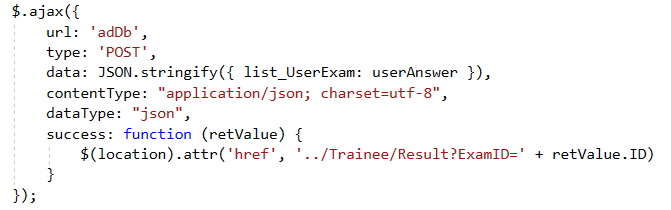
* Controller:
* Tạo session cho người dùng.
* Phân quyền.



* Controller + View: Truyền dữ liệu lên View thông qua ViewBag:

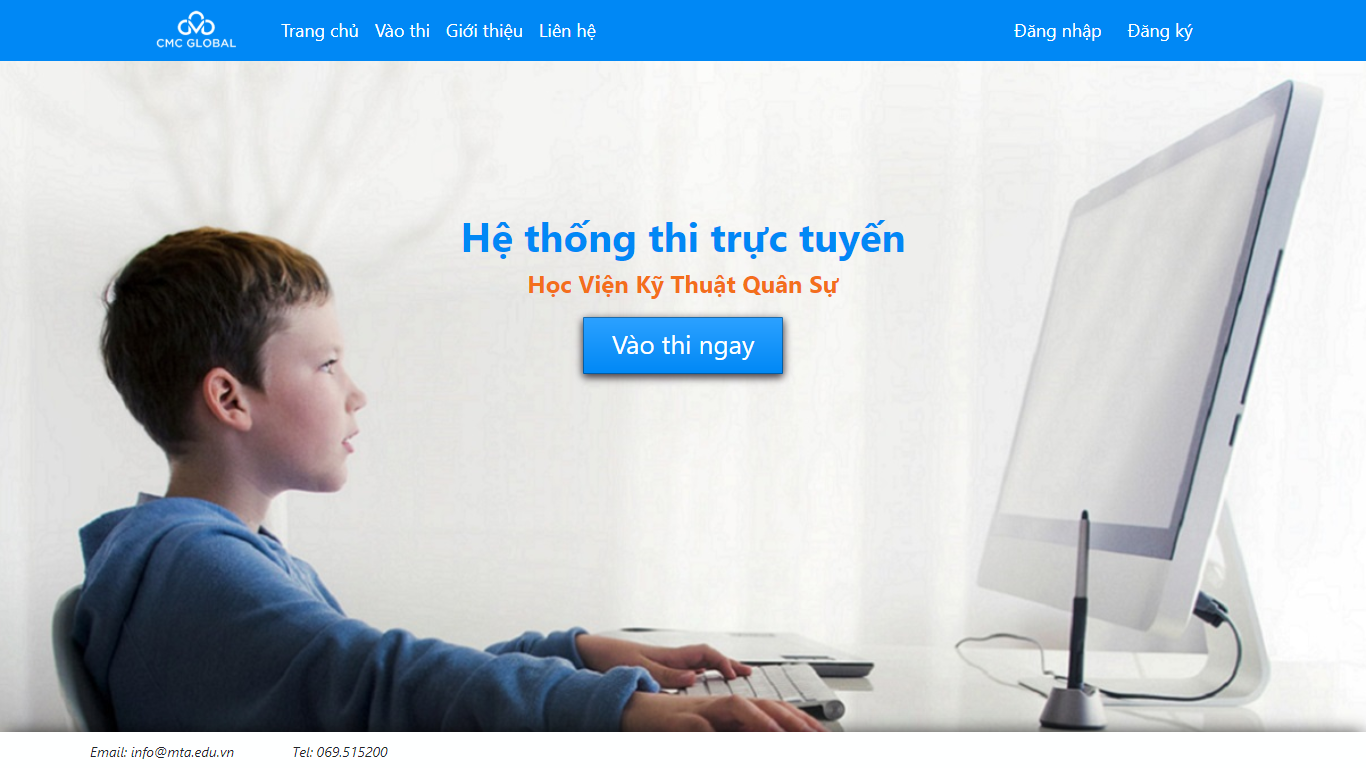


* Controller + View: Nhận dữ liệu từ View thông qua Ajax.

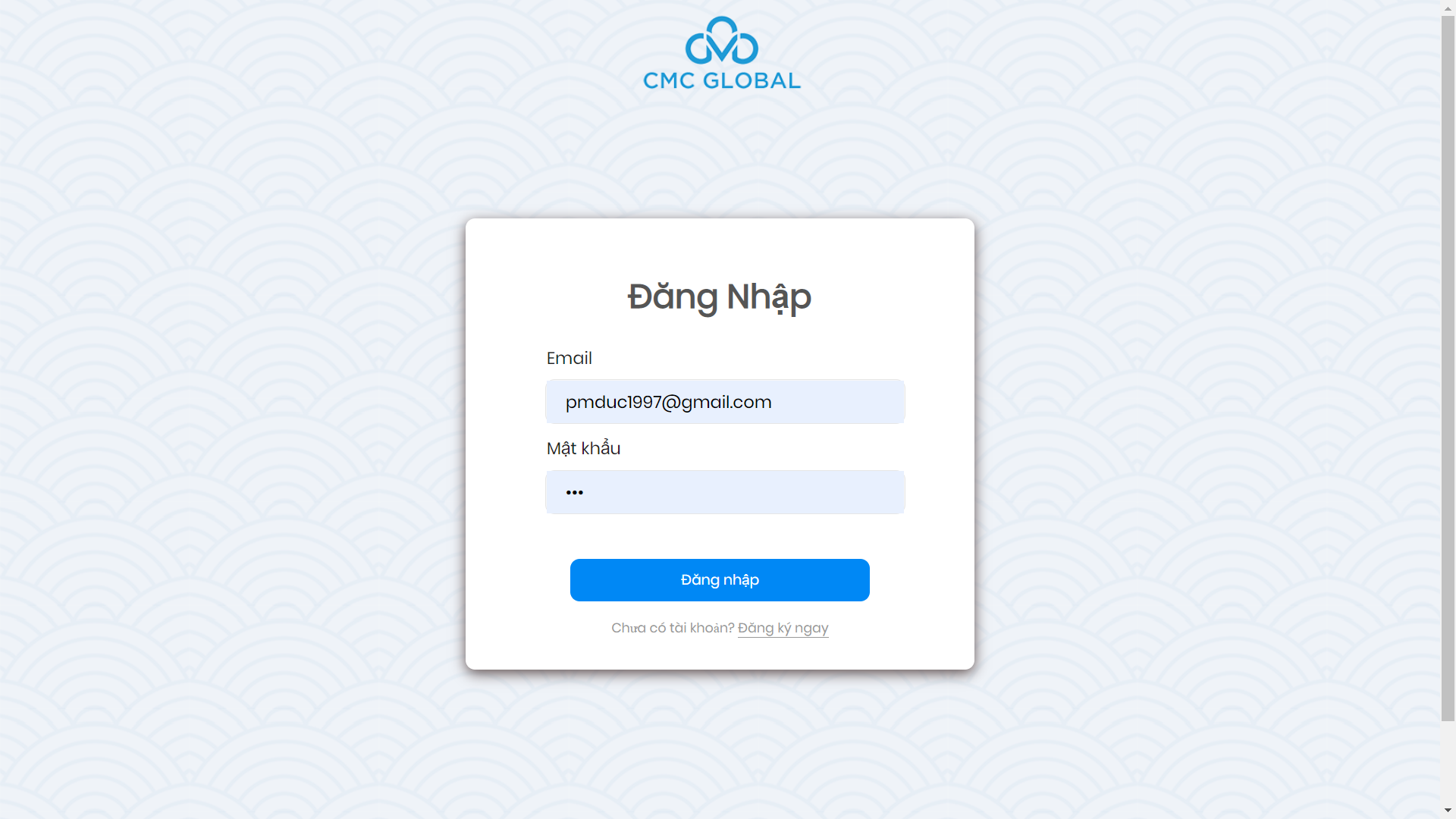


* View:

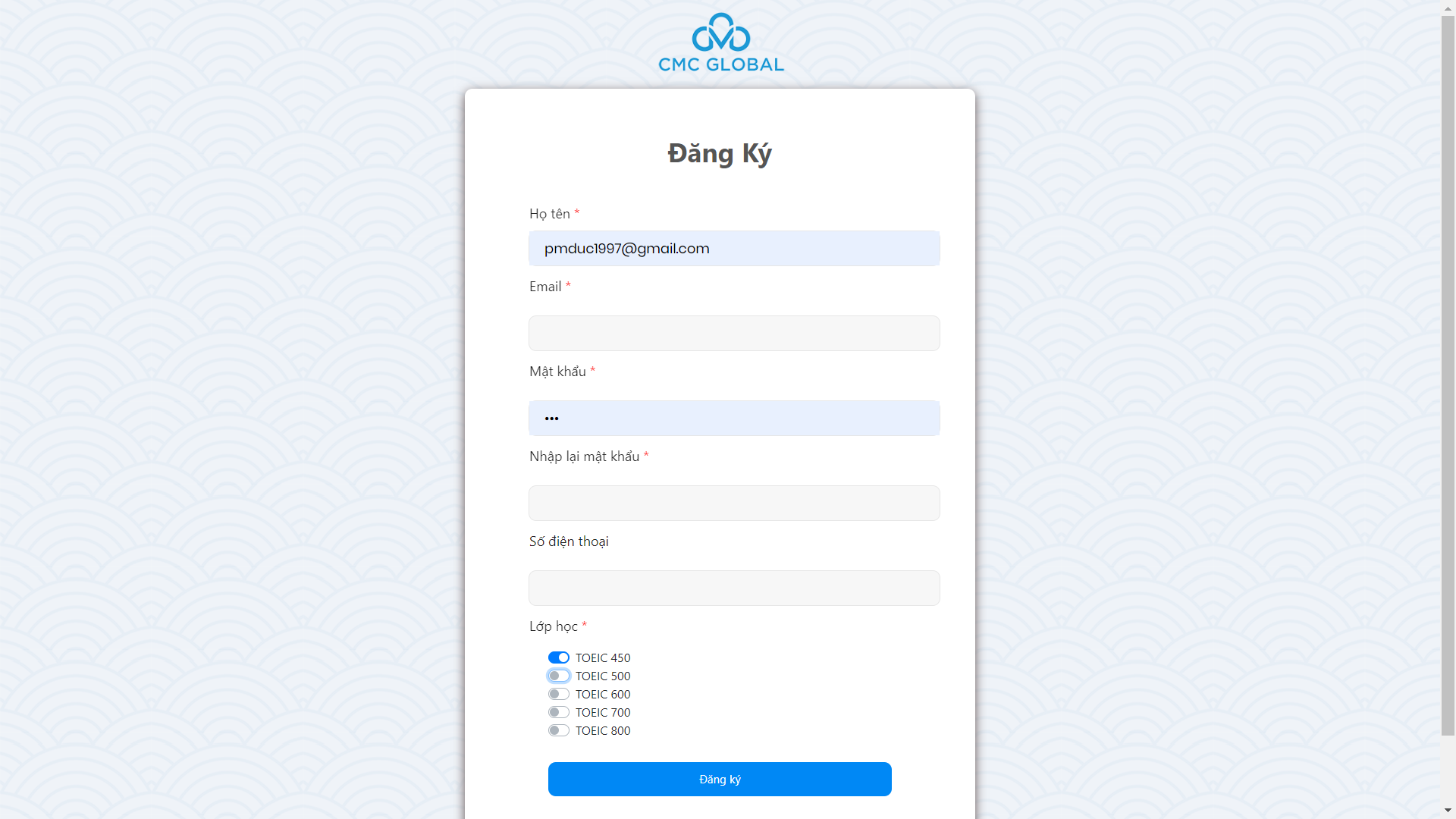
Trang chủ:



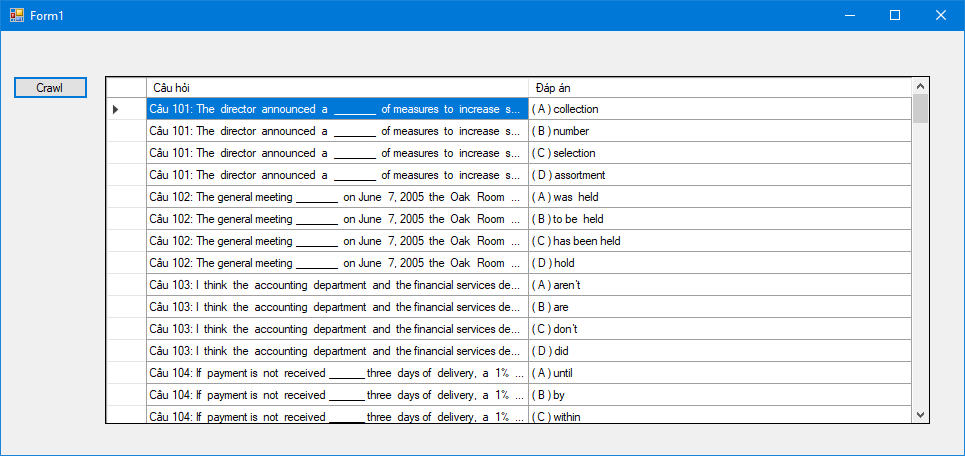
Đăng nhập:



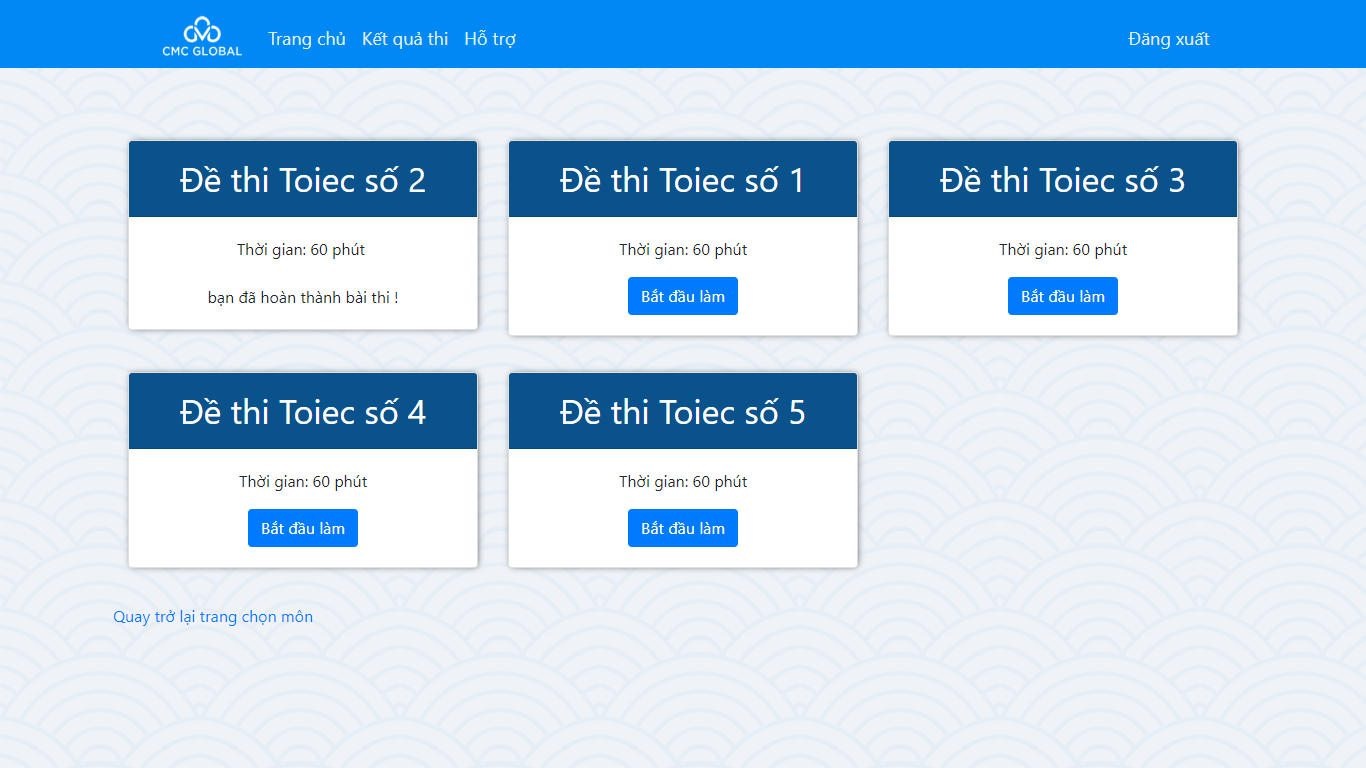
Đăng ký:



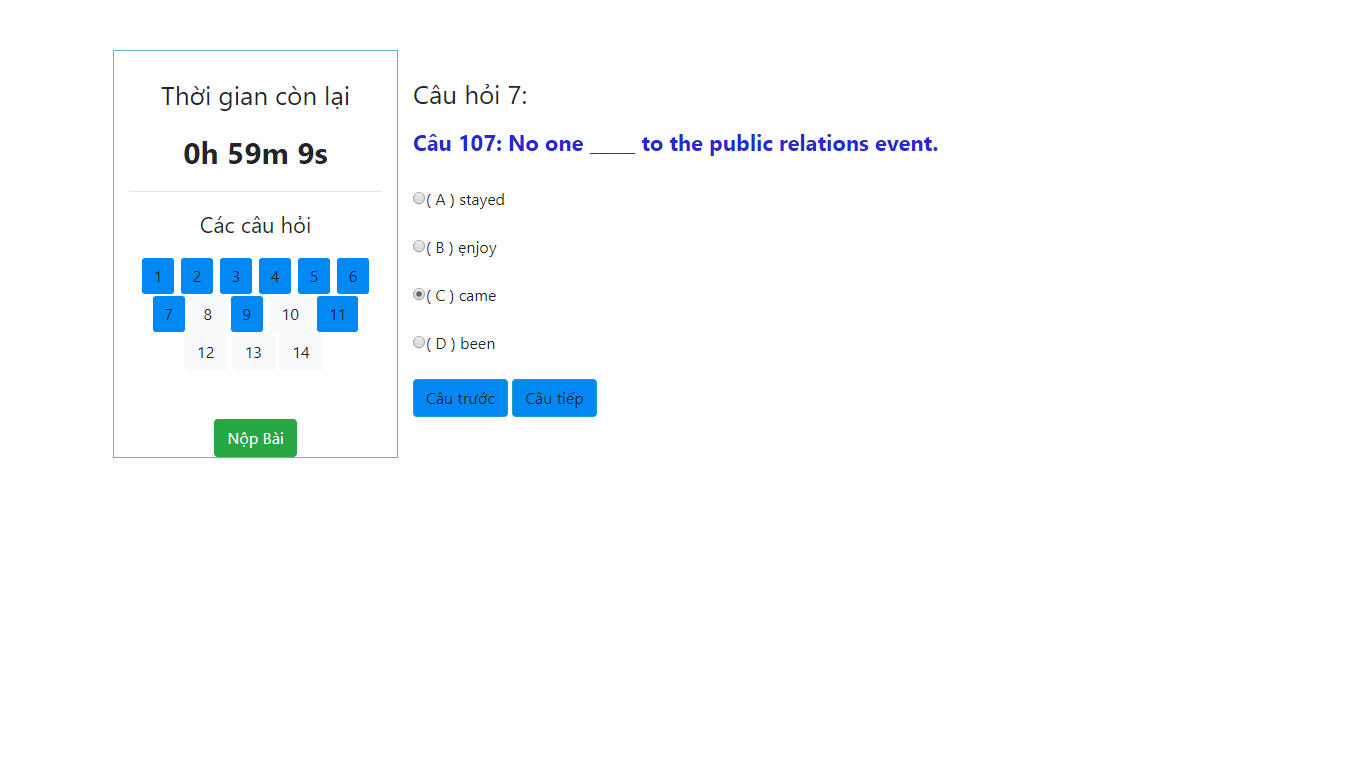
Crawl data: crawl data từ trang web : http://tuyendung.flyhigh.edu.vn/php/reading3.php



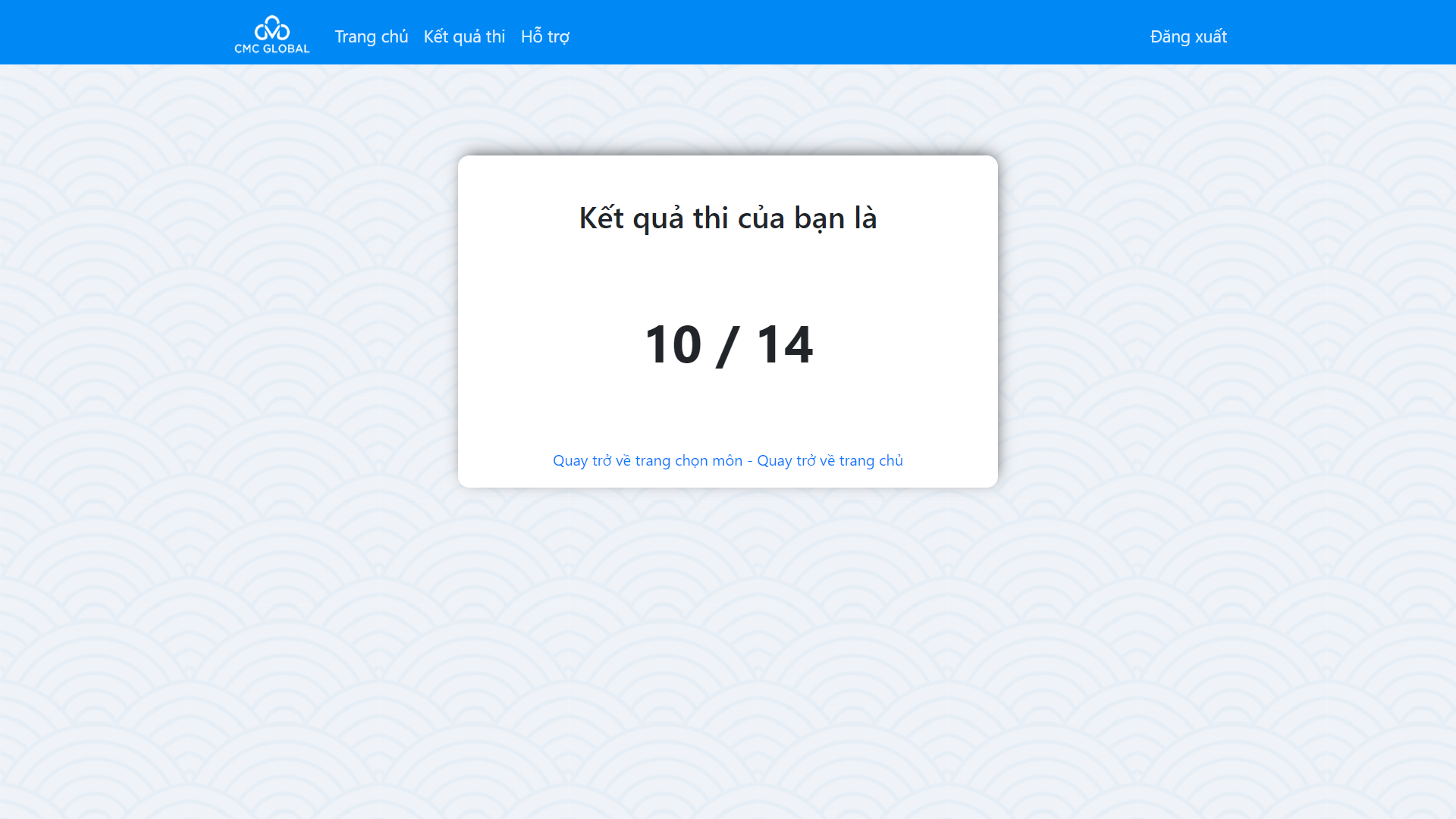
Crawl data xong tự động insert vào database:



Trang làm bài thi:



Kết quả thi:



# **IV. Những khó khăn trong quá trình làm bài tập**

Trong thời gian thực tập và làm bài tập được giao vừa qua, bản thân em đã học được thêm rất nhiều kiến thức mới. Em hiểu được thêm về nhiều hệ quản trị cơ sở dữ liệu khác nhau chứ không chỉ có SQL Server hay MySQL. Em được tự xây dựng một hệ thống thi trắc nghiệm online, đó là một bài toán rất thực tế. Ở trường thì em cảm thấy rất tự tin với kiến thức mình có, nhưng thực sự thì khi bắt đầu làm việc vào dự án thực tế thì em lại cảm thấy kiến thức của mình không đáp ứng được đối với bài toán thực tế. Các kiến thức từ nhà trường như thiết kế giao diện html, css, ASP .NET, mô hình MVC,… đều khá cơ bản, không đủ khi áp dụng đối với bài toán thực tế do có quá nhiều vấn đề khác nhau nảy sinh. Chính vì điều đó, em phải tăng khả năng tự học, tự nghiên cứu của mình, cần phải cung cấp, nâng cao kiến thức của mình hơn nữa để có thể hoàn thành tốt các dự án.

Vậy nên, qua quá trình thực tập, em nhận thấy mình nhận được rất nhiều thứ kiến thức, kinh nghiệm làm việc, tác phong, môi trường…, dù chỉ gặp 1 vài vấn đề khó khăn nhỏ nhưng em đã rất cố gắng có thể vượt qua nó.