

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**



**TIỂU LUẬN HỌC PHẦN:**

**CÔNG NGHỆ .NET**

**TÊN ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG THI TRẮC NGHIỆM**

**NHÓM: 10**

**Thành phố Hồ Chí Minh, tháng 12 năm 2024**

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**



**Nhóm: 10**

**Giảng viên hướng dẫn:**

Nhóm trưởng: Trần Công Minh

Lương Thái Hiền

Thành viên:

1. Lê Đức Trung
2. Vương Cường Thuận

## DANH SÁCH PHÂN CÔNG

STT	MSSV	HỌ TÊN	CÔNG VIỆC ĐƯỢC GIAO	ĐÁNH GIÁ
1	2001222641	Trần Công Minh	Các Form còn lại + Format Word	100%
2	2001225676	Lê Đức Trung	Word(Chương I, II) + Form(Login, ChangePassword, Manage User)	100%
3	2001202264	Vương Cường Thuận	Form Report + Word(Chương II, III)	100%

## Lời cảm ơn

Lời đầu tiên, chúng em xin được gửi lời cảm ơn chân thành nhất đến thầy Lương Thái Hiền. Trong quá trình học tập và tìm hiểu môn **Công nghệ .NET**, chúng em đã nhận được rất nhiều sự quan tâm, giúp đỡ, hướng dẫn tâm huyết và tận tình của thầy. Thầy đã giúp chúng em tích lũy thêm nhiều kiến thức về môn học này để có thể hoàn thành được bài tiểu luận về đề tài: **Xây dựng ứng dụng thi trắc nghiệm**

Trong quá trình làm bài chắc chắn khó tránh khỏi những thiếu sót. Do đó, chúng em kính mong nhận được những lời góp ý của thầy để bài tiểu luận của chúng em ngày càng hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

# MỤC LỤC

<b>GIỚI THIỆU.....</b>	<b>2</b>
1. <b>Tổng quan về WINFROM .....</b>	<b>2</b>
2. <b>Có nên học Winform không? .....</b>	<b>3</b>
3. <b>Kết luận.....</b>	<b>3</b>
<b>CHƯƠNG I. TỔNG QUAN VỀ ĐỀ TÀI.....</b>	<b>5</b>
I. <b>Giới thiệu .....</b>	<b>5</b>
1. <i><b>Định nghĩa vấn đề .....</b></i>	<i><b>5</b></i>
2. <i><b>Phạm vi đề tài .....</b></i>	<i><b>5</b></i>
3. <i><b>Mục tiêu đề tài .....</b></i>	<i><b>6</b></i>
4. <i><b>Sự cần thiết của đề tài.....</b></i>	<i><b>7</b></i>
5. <i><b>Tại sao chọn đề tài này?.....</b></i>	<i><b>7</b></i>
<b>CHƯƠNG II. PHÂN TÍCH ĐỀ TÀI.....</b>	<b>9</b>
I. <b>Phân tích yêu cầu hệ thống .....</b>	<b>9</b>
1. <i><b>Phân tích nghiệp vụ.....</b></i>	<i><b>9</b></i>
2. <i><b>Mô tả nghiệp vụ.....</b></i>	<i><b>9</b></i>
<b>CHƯƠNG III. PHÂN TÍCH THIẾT KẾ.....</b>	<b>11</b>
I. <b>Mô hình thiết kế .....</b>	<b>11</b>
1. <i><b>Mô hình ERD.....</b></i>	<i><b>11</b></i>
2. <i><b>Mô hình SQL .....</b></i>	<i><b>13</b></i>
3. <i><b>Mô hình CDM.....</b></i>	<i><b>15</b></i>
II. <b>Các thủ tục được cài đặt trong SQL .....</b>	<b>15</b>
1. <i><b>Thủ tục của bảng Question.....</b></i>	<i><b>15</b></i>
4. <i><b>Thủ tục bảng Subject.....</b></i>	<i><b>19</b></i>
5. <i><b>Thủ tục của bảng TestHistory.....</b></i>	<i><b>22</b></i>
6. <i><b>Thủ tục của bảng UserAccount .....</b></i>	<i><b>26</b></i>
7. <i><b>Thủ tục của bảng UserRole .....</b></i>	<i><b>31</b></i>
III. <b>Database design và thư viện hỗ trợ.....</b>	<b>32</b>

1.	<i>Database design</i> .....	32
2.	<i>Thư viện, công cụ hỗ trợ</i> .....	32
	<b>CHƯƠNG IV. Thực hiện</b> .....	35
I.	<b>Giao diện thiết kế</b> .....	35
1.	<b>Giao diện form đăng nhập:</b> .....	35
2.	<i>Giao diện form chọn môn thi:</i> .....	36
3.	<i>Giao diện form làm bài:</i> .....	36
4.	<i>Giao diện form kết quả thi:</i> .....	37
5.	<i>Giao diện form quản lý người dùng (Admin):</i> .....	38
6.	<i>Giao diện form quản lý câu hỏi:</i> .....	38
7.	<i>Giao diện form quản lý môn thi:</i> .....	39
8.	<i>Giao diện form báo cáo thống kê theo môn:</i> .....	40
9.	<i>Giao diện form báo cáo thống kê theo thời gian:</i> .....	41
10.	<i>Giao diện form sao lưu dữ liệu và phục hồi:</i> .....	42
	<b>CHƯƠNG V. KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN ...</b>	44
I.	<b>Kết luận</b> .....	44
II.	<b>Định hướng phát triển</b> .....	44
	<b>TÀI LIỆU THAM KHẢO</b> .....	46
	<b>PHỤ LỤC</b> .....	46
	<b>Danh mục hình ảnh</b> .....	46

## Lời mở đầu

Công nghệ thông tin là lĩnh vực được đào tạo để sử dụng máy tính và các phần mềm để xử lý, phân phối, và quản lý thông tin. Đồng thời, nó cũng dùng để trao đổi, lưu trữ và chuyển đổi dữ liệu thông tin dưới nhiều hình thức khác nhau. Với tốc độ phát triển nhanh chóng, ngành Công nghệ thông tin đã đóng góp lớn vào sự tăng trưởng kinh tế và sự tiến bộ của xã hội.

Chúng em, là những sinh viên mới bắt đầu học về ngành Công nghệ thông tin, nhận thức được tầm quan trọng của việc xây dựng một nền tảng kiến thức vững chắc để tiếp tục học tập và phát triển trong ngành này.

Dưới sự hướng dẫn của giảng viên môn "Công Nghệ.NET", chúng em đã được tìm hiểu về lĩnh vực Công nghệ thông tin và nhận thức được vai trò quan trọng của nó trong cuộc sống và sự phát triển của xã hội. Việc này đã giúp chúng em mở rộng kiến thức và nâng cao khả năng ứng dụng của mình trong thực tế.

Một lần nữa, em xin bày tỏ lòng biết ơn sâu sắc đến quý thầy cô và đặc biệt là thầy Lương Thái Hiền, đã dành thời gian và tâm huyết để hỗ trợ chúng em trong quá trình học tập và nghiên cứu. Cảm ơn quý thầy cô đã luôn đồng hành và hỗ trợ chúng em trên con đường học tập và trưởng thành.

# GIỚI THIỆU

## 1. Tổng quan về WINFROM

Winform, viết tắt của Windows Forms, là thuật ngữ mô tả ứng dụng được viết dùng .NET Framework và có giao diện người dùng Windows Forms. Được Microsoft xây dựng, nhưng chính thức chấm dứt hỗ trợ vào năm 2014. Winform vẫn duy trì sức sống ở Việt Nam và được sử dụng rộng rãi trong việc phát triển ứng dụng Window trên máy tính cá nhân. Mặc dù Microsoft đã ngừng phát triển, nhưng Winform vẫn được ưa chuộng tại Việt Nam do sự linh hoạt và tính ổn định của nó. Các ứng dụng như Microsoft Word, Excel, Access, Calculator, Yahoo, và Mail đều sử dụng công nghệ Windows Forms.

### **Ưu điểm của Winfrom:**

- **Giao diện kéo thả và dễ dàng sử dụng:** Việc tạo giao diện thông qua thao tác kéo thả làm cho quá trình trở nên dễ dàng hơn. Sự hỗ trợ đa dạng về các sự kiện như Click, Hover,...giúp tối ưu hóa trải nghiệm lập trình viên
- **Thực quan và dễ học:** Winfrom cung cấp môi trường lập trình lạc quan, giúp người học dễ hiểu và tiếp cận dễ dàng hiểu rõ về chúng hơn. Việc sử dụng component có sẵn và các thư viện như TelerikUI hoặc DevExpress giúp tạo giao diện trở nên hiện đại và lung linh hoa mỹ hơn.
- **Tốc độ xử lý nhanh:** Winfrom cung cấp tốc độ xử lý dữ liệu nhanh chóng, phù hợp cho các ứng dụng quản lý tính tiền, thống kê,...
- **An toàn bảo mật:** Đảm bảo an toàn và bảo mật thông tin trên nền tảng Windows.
- **Phù hợp với nhiều bản Windows:** Winfrom có thể chạy trên nhiều phiên bản Windows khác nhau, tăng sự linh hoạt cho người sử dụng.
- **Thao tác trên nhiều giao diện:** Hỗ trợ thao tác trên nhiều giao diện, tạo nên trải nghiệm người dùng đa dạng.



## **Nhược điểm của Winform:**

- **Giới hạn trên Desktop:** Winform chỉ phù hợp cho các ứng dụng trên Desktop, không linh hoạt cho các ứng dụng di động.

- **Phụ thuộc vào nền tảng Windows:** Người dùng phải sử dụng máy tính đã cài đặt phần mềm, tăng khả năng giới hạn sử dụng.

- **Đồ họa không cao:** Giao diện đồ họa trên Winform có hạn chế, làm giảm tính trực quan và thân thiện với người dùng.

## **2. Có nên học Winform không?**

Ta có thể thấy đây là một công nghệ tuyệt vời, với chức năng kéo thả tạo giao diện vô cùng đơn giản, môi trường viết code trực quan hơn và muôn vàn các chức năng khác. Với những ưu điểm trên có thể thấy Winform rất mạnh mẽ, vẫn tồn tại được vậy đặt ra câu hỏi: “Có nên tập trung vào học Winform không?”

“Bạn có thể tham khảo trải nghiệm Winform, chứ không nên dành quá nhiều thời gian để master vào chúng” là những lời nói, lời khuyên đặc biệt dành cho những người học mới lập trình. Sẽ có một vài lý do để bạn có thể xem xét trước khi quyết định có nên học hay không:

**Microsoft đã khai tử Winform (từ 2014):** Winform đã bị khai tử điều này sẽ dẫn đến việc cho sử dụng và phát triển trở nên khó khăn hơn.

**Yêu cầu tuyển dụng:** Ở thời điểm hiện tại Winform vẫn còn được tuyển dụng nhưng số lượng không được nhiều như trước. Các công ty thường sẽ đặt ưu tiên cho những công nghệ mới hơn, ngôn ngữ hiện đại hơn.

**Bảo trì các dự án cũ:** Nếu môi trường làm việc yêu cầu kiến thức về Winform, có thể bạn sẽ phải bảo trì các dự án cũ thay vì tham gia vào những dự án mới và hấp dẫn. Công việc bảo trì nhàm chán

## **3. Kết luận**

Chung quy lại Winfrom vẫn chưa “hết thời” tại Việt Nam, vẫn còn sống tốt và mạnh mẽ những chỉ là không còn phổ biến như trước kia. Thay vào đó bạn có thể học thêm các công nghệ mới và ngôn ngữ mới để có cơ hội nghề nghiệp tốt hơn. Tuy nhiên bạn vẫn có thể cân nhắc học Winfrom khi thật sự cần thiết.

# CHƯƠNG I. TỔNG QUAN VỀ ĐỀ TÀI

## I. Giới thiệu

Ứng dụng thi trắc nghiệm là một phần mềm ứng dụng thiết thực cho việc thi và đánh giá kết quả học tập của học sinh, sinh viên một cách thiết thực và khách quan nhất. Ngoài ra còn giúp cho các bộ phận công tác quản lý và tổ chức quản lý các kỳ thi thuận tiện và khách quan nhất. Ứng dụng thi trắc nghiệm giúp phương pháp học tập trở nên dễ dàng hơn, tối ưu hơn và hiện đại hóa.

### 1. Định nghĩa vấn đề

Trong bối cảnh giáo dục hiện nay, các tổ chức thực hiện các kỳ thi trắc nghiệm trực tuyến dần trở nên phổ biến hơn. Tuy nhiên, việc xây dựng một ứng dụng thi trắc nghiệm hoàn chỉnh phải đòi hỏi xử lý nhiều vấn đề như: quản lý câu hỏi, thời gian làm bài, tình điểm tự động, và cung cấp giao diện thân thiện với người dùng

Một số hệ thống thi trắc nghiệm hiện tại có thể thiếu tính tùy biến hoặc không đáp ứng được nhu cầu cụ thể của người dùng. Do đó, việc xây dựng một ứng dụng thi trắc nghiệm hiệu quả, dễ sử dụng là rất cần thiết.

### 2. Phạm vi đề tài

Đề tài này tập trung vào việc xây dựng một ứng dụng thi trắc nghiệm đơn giản với các tính năng cơ bản:

- Giao diện thân thiện dễ sử dụng cho người quản lý, người thi.
- Chức năng quản lý câu hỏi, đáp án.
- Tính năng chọn câu trả lời và chấm điểm tự động.
- Quản lý thời gian thi.
- Hiện thị kết quả của bài thi.

- Xem báo cáo dữ liệu các bài thi sau khi người thi hoàn thành bài thi.

Ứng dụng được phát triển dưới dạng phần mềm Winform dựa vào nền tảng .NET. Có thể ứng dụng cho các trường học, công ty chuyên tổ chức kỳ thi trắc nghiệm, kiểm tra định kì.

### **3. Mục tiêu đề tài**

Mục tiêu chính của đề tài là phát triển một ứng dụng thi trắc nghiệm có đầy đủ các tính năng và thân thiện với người dùng, bao gồm:

- Hỗ trợ cho tất cả người dùng:
  - Đăng nhập, đăng xuất hệ thống của ứng dụng.
  - Xem và sửa thông tin cá nhân.
- Hỗ trợ cho hệ thống:
  - Chấm điểm tự động sau khi hoàn thành bài thi.
  - Hiện thị điểm số và thông báo cho người dùng.
  - Cập nhật điểm, thông tin cá nhân, đề thi và câu hỏi vào SQL trên sever hệ thống.
- Hỗ trợ cho ban tổ chức, đào tạo:
  - Quản lí các đợt thi.
  - Quản lí các môn học.
  - Quản lí danh sách các học sinh, sinh viên tham gia thi.
- Hỗ trợ học sinh, sinh viên:
  - Làm bài thi.

- Xem kết quả bài thi.
- Hỗ trợ Admin
  - Xem thông báo.
  - Quản trị toàn bộ hệ thống.
  - Phân quyền vào tạo người dùng mới vào hệ thống.

#### 4. Sự cần thiết của đề tài

**Tối ưu hóa trong việc thi cử:** trong môi trường học tập và làm việc, việc tổ chức các kỳ thi trắc nghiệm bằng giấy tốn thời gian và công sức. Một ứng dụng thi trắc nghiệm số hóa sẽ giúp tự động hóa quá trình này, tiết kiệm thời gian và nâng cao hiệu quả.

**Tính khách quan:** ứng dụng giúp giảm thiểu sự can thiệp của con người trong việc chấm thi, từ đó đảm bảo tính khách quan và minh bạch.

**Tính dễ sử dụng:** nhiều hệ thống thi trắc nghiệm hiện tại có giao diện phức tạp, khó sử dụng cho người dùng không quen thuộc với công nghệ. Ứng dụng này sẽ tập trung vào trải nghiệm người dùng, đảm bảo dễ sử dụng cho mọi đối tượng..

#### 5. Tại sao chọn đề tài này?

**Tiết kiệm thời gian và chi phí tổ chức thi cử:** Việc thi trắc nghiệm truyền thống bằng giấy tốn kém về cả chi phí in ấn và thời gian quản lý. Một ứng dụng thi trắc nghiệm số hóa sẽ giúp tiết kiệm các chi phí này và đồng thời giảm thiểu công sức của giám thị trong việc kiểm soát, phát đề và chấm điểm.

**Quản lý số lượng lớn thí sinh dễ dàng:** Trong các kỳ thi lớn có hàng trăm, thậm chí hàng ngàn thí sinh, việc quản lý và theo dõi kết quả thi là thách thức. Ứng dụng giúp quản lý thí sinh và kết quả thi một cách nhanh chóng và dễ dàng, đặc biệt là trong môi trường giáo dục trực tuyến.

**Độ chính xác và khách quan cao:** Chấm điểm tự động giúp đảm bảo không có sự thiên vị hay sai sót của con người, đồng thời đảm bảo tính công bằng và minh bạch cho tất cả thí sinh.

**Cải thiện trải nghiệm người dùng:** Một ứng dụng thi trắc nghiệm với giao diện thân thiện sẽ giúp thí sinh dễ dàng tập trung vào bài thi, thay vì loay hoay với cách thức thực hiện. Điều này đặc biệt quan trọng trong môi trường học tập và thi cử trực tuyến.

**Khả năng ứng dụng rộng rãi:** Ứng dụng có thể được sử dụng cho nhiều mục đích khác nhau, không chỉ trong giáo dục mà còn trong các bài kiểm tra tuyển dụng, đánh giá năng lực nhân sự trong doanh nghiệp.

**Dễ dàng cập nhật và bảo trì nội dung thi:** Ngân hàng câu hỏi trong ứng dụng có thể được dễ dàng cập nhật và bảo trì mà không phải in lại đề mới như trong thi truyền thống. Điều này giúp tăng tính linh hoạt cho các kỳ thi và bài kiểm tra.

**Phân tích kết quả nhanh chóng:** Ứng dụng có thể tích hợp các tính năng phân tích dữ liệu, giúp nhà quản lý có thể nhanh chóng tổng hợp, đánh giá kết quả và đưa ra các báo cáo chi tiết về hiệu suất của từng thí sinh hoặc nhóm thi.

## CHƯƠNG II. PHÂN TÍCH ĐỀ TÀI

### I. Phân tích yêu cầu hệ thống

#### 1. Phân tích nghiệp vụ

Trong quá trình phân tích, cần khảo sát và nghiên cứu các yêu cầu nghiệp vụ liên quan đến việc xây dựng ứng dụng thi trắc nghiệm. Mục tiêu là hiểu rõ các nhu cầu và mong đợi của người dùng, từ đó xây dựng các tính năng và chức năng của hệ thống đáp ứng đúng yêu cầu thực tiễn.

- **Đối tượng người dùng:** Các giáo viên, giảng viên, học sinh, sinh viên hoặc người đi thi.
- **Quản lý ngân hàng câu hỏi:** Hệ thống cần có chức năng quản lý câu hỏi trắc nghiệm bao gồm việc thêm, sửa, xóa, và cập nhật câu hỏi. Các câu hỏi có thể được phân loại theo môn học.
- **Tạo đề thi tự động:** Ứng dụng cần chức năng cho phép tạo ra các bài thi tự động từ ngân hàng câu hỏi dựa trên các tiêu chí như số lượng câu hỏi.
- **Quản lý thí sinh:** Hệ thống phải lưu trữ thông tin của người thi, bao gồm thông tin cá nhân và lịch sử thi cử của họ.
- **Chấm điểm tự động:** Sau khi hoàn thành bài thi, hệ thống sẽ chấm điểm tự động và đưa ra kết quả dựa trên câu trả lời của thí sinh.
- **Quản lý thời gian thi:** Thiết lập thời gian cho mỗi bài thi, hệ thống phải tự động thu bài khi hết thời gian hoặc thông báo cho người thi về thời gian còn lại.
- **Báo cáo kết quả:** Cần có chức năng xuất báo cáo kết quả thi của từng thí sinh, bao gồm điểm số, thời gian làm bài, và các câu hỏi đã trả lời đúng/sai.

#### 2. Mô tả nghiệp vụ

- Người quản lí(Admin, giáo viên, giảng viên):

**Thêm câu hỏi:** Thêm mới câu hỏi vào ngân hàng câu hỏi, bao gồm nội dung câu hỏi, các lựa chọn đáp án và xác định đáp án đúng.

**Chỉnh sửa câu hỏi:** Có thể cập nhật nội dung câu hỏi hoặc đáp án.

**Xóa câu hỏi:** Xóa các câu hỏi không còn phù hợp hoặc lỗi thời.

**Tạo đề thi:** Chọn câu hỏi từ ngân hàng câu hỏi và tạo đề thi tự động theo các tiêu chí (số lượng câu hỏi, độ khó, môn học).

**Quản lý thí sinh:** Xem danh sách người thi và kết quả thi của từng người.

- Người thi (Học sinh, sinh viên):

**Đăng nhập:** Người thi cần đăng nhập vào hệ thống để thực hiện bài thi.

**Làm bài thi:** Chọn câu trả lời từ các lựa chọn có sẵn cho mỗi câu hỏi, có thể thay đổi câu trả lời trong thời gian làm bài.

**Kết thúc bài thi:** Khi hết thời gian hoặc hoàn thành bài thi, người thi tự nộp bài khi hoàn thành xong hoặc khi hết thời gian hệ thống sẽ tự động nộp bài.

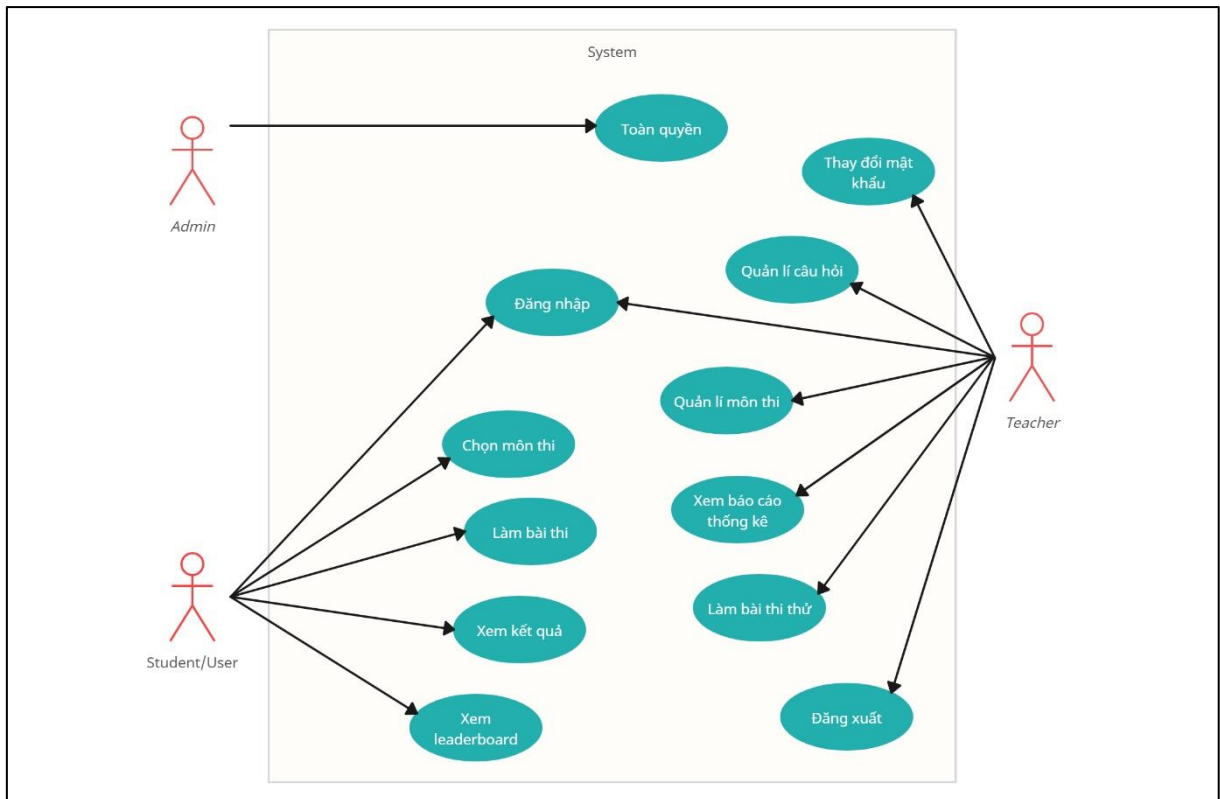
- Hệ thống (Phần mềm):

**Chấm điểm tự động:** So sánh câu trả lời của người thi với đáp án đúng và tính điểm.

**Hiển thị kết quả:** Hiển thị thông báo điểm số cuối cùng và số câu trả lời đúng / sai của người thi ngay sau khi hoàn tất bài thi.

**Lưu trữ kết quả thi:** Lưu trữ kết quả thi vào cơ sở dữ liệu để giáo viên có thể tra cứu, xuất báo cáo.





Hình 1. Use case Admin/Teacher/User

## CHƯƠNG III. PHÂN TÍCH THIẾT KẾ

### I. Mô hình thiết kế

#### 1. Mô hình ERD

Mô hình quan hệ thực thể (**Entity-Relationship Model**) của hệ thống quản lý kỳ thi với các bảng và mối quan hệ như sau:

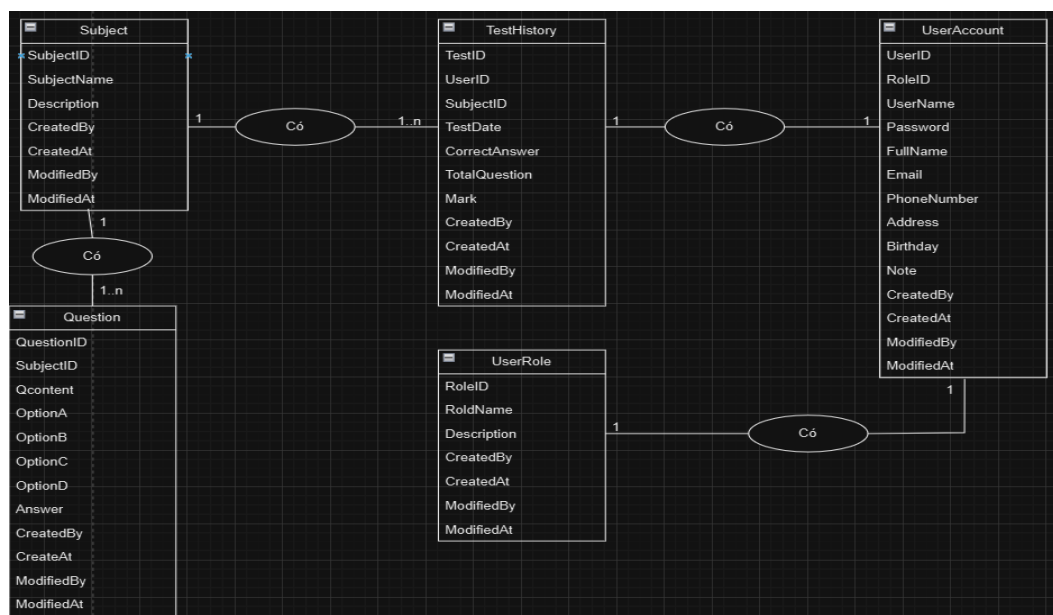
- **Subject:** Lưu trữ thông tin về môn học, bao gồm các thuộc tính như mã môn học (SubjectID), tên môn học (SubjectName), mô tả (Description), và thông tin về người tạo, thời gian tạo và chỉnh sửa. Bảng này có quan hệ "1-nhiều" với bảng **Question** và **TestHistory**.

- **Question:** Chứa thông tin về câu hỏi của từng môn học, bao gồm nội dung câu hỏi (QContent), các tùy chọn trả lời (OptionA, OptionB, OptionC, OptionD), đáp án đúng (Answer) và các thông tin người tạo, thời gian tạo và chỉnh sửa. Mỗi môn học trong **Subject** có thể có nhiều câu hỏi.

- **TestHistory**: Lưu trữ lịch sử làm bài của người dùng, bao gồm mã bài kiểm tra (TestID), mã người dùng (UserID), mã môn học (SubjectID), ngày làm bài (TestDate), số câu trả lời đúng (CorrectAnswer), tổng số câu hỏi (TotalQuestion), điểm số (Mark), và các thông tin người tạo, thời gian tạo và chỉnh sửa. Bảng này có quan hệ "1-nhiều" với bảng **UserAccount** và **Subject**.

- **UserAccount**: Quản lý thông tin tài khoản người dùng, bao gồm mã người dùng (UserID), mã vai trò (RoleID), tên đăng nhập (Username), mật khẩu (Password), tên đầy đủ (Fullname), email, số điện thoại, địa chỉ, ngày sinh và các thông tin khác. Bảng này có quan hệ "1-1" với bảng

- **UserRole** và "1-nhiều" với bảng **TestHistory**. **UserRole**: Xác định vai trò của người dùng trong hệ thống với các thuộc tính như mã vai trò (RoleID), tên vai trò (RoleName), mô tả (Description) và các thông tin người tạo, thời gian tạo và chỉnh sửa.



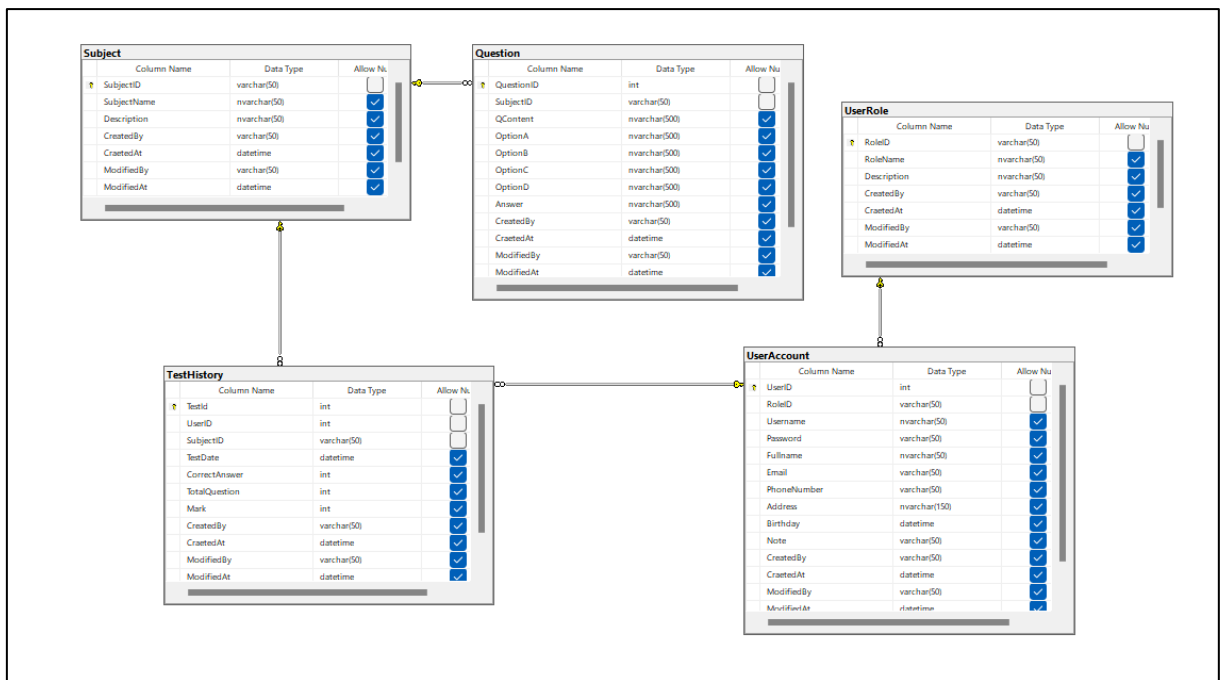
Hình 2. Mô hình ERD

Các quan hệ "Có" thể hiện mối liên kết giữa các thực thể, giúp biểu diễn rõ ràng các liên kết khóa ngoại giữa các bảng, tạo nên tính logic và ràng buộc trong hệ thống quản lý kỳ thi.

## 2. Mô hình SQL

Bảng dữ liệu trên thể hiện một phần cấu trúc của hệ thống quản lý kỳ thi với các bảng sau:

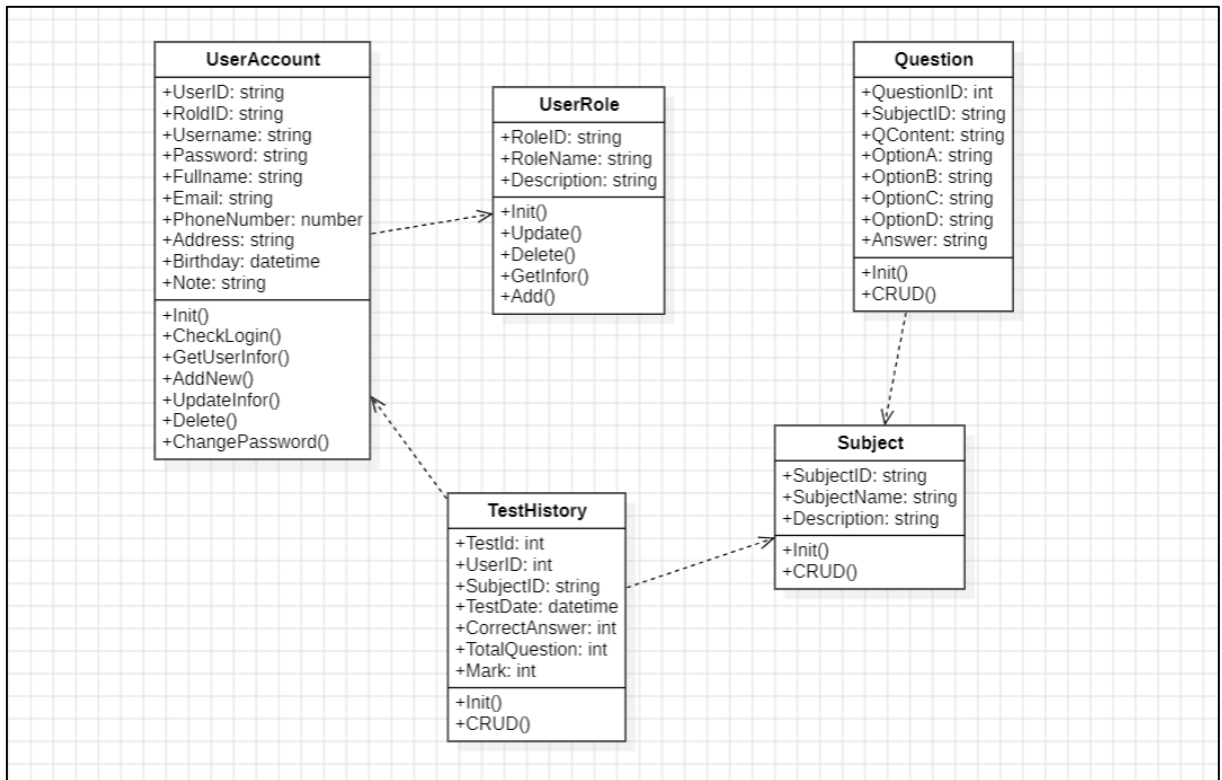
- **Subject:** Lưu thông tin về các môn học, bao gồm mã môn học (SubjectID), tên môn học (SubjectName), mô tả (Description), và các thông tin người tạo, chỉnh sửa cùng thời gian.
- **Question:** Lưu trữ các câu hỏi cho từng môn học, bao gồm nội dung câu hỏi (QContent), các lựa chọn đáp án (OptionA, OptionB, OptionC, OptionD), đáp án đúng (Answer), và các thông tin về người tạo và chỉnh sửa.
- **TestHistory:** Lưu lịch sử các bài kiểm tra của người dùng với thông tin như mã bài kiểm tra (TestID), mã người dùng (UserID), mã môn học (SubjectID), ngày làm bài (TestDate), số câu trả lời đúng (CorrectAnswer), tổng số câu hỏi (TotalQuestion), điểm (Mark), và các thông tin người tạo, chỉnh sửa cùng thời gian.
- **UserAccount:** Quản lý tài khoản người dùng, bao gồm thông tin về mã người dùng (UserID), mã vai trò (RoleID), tên đăng nhập (Username), mật khẩu (Password), tên đầy đủ (Fullname), email, số điện thoại, địa chỉ, ngày sinh và các thông tin khác.
- **UserRole:** Lưu vai trò của người dùng, bao gồm mã vai trò (RoleID), tên vai trò (RoleName), mô tả (Description), và các thông tin người tạo, chỉnh sửa cùng thời gian.



Hình 3. Mô hình SQL

Các bảng này có quan hệ khóa ngoại với nhau, giúp quản lý hiệu quả mối liên hệ giữa các môn học, câu hỏi, người dùng và các vai trò trong hệ thống quản lý kỳ thi.

### 3. Mô hình CDM



Hình 4. Mô hình CDM

## II. Các thủ tục được cài đặt trong SQL

### 1. Thủ tục của bảng Question

#### 3.1. Thủ tục [dbo].[Question\_Delete]

Thủ tục **Question\_Delete** có chức năng xóa một câu hỏi dựa vào mã câu hỏi (**QuestionID**).

```
create procedure [dbo].[UserAccount_Delete]
@UserId int
AS

DELETE FROM [dbo].[UserAccount]
where UserID = @UserId

--
```

Thủ tục ***Question\_Delete*** hỗ trợ xóa câu hỏi khỏi cơ sở dữ liệu một cách an toàn và dễ dàng bằng cách gọi thủ tục, giúp đảm bảo tính tổ chức trong mã lệnh và dễ duy trì hơn.

### 3.2. Thủ tục [dbo].[Question\_GetById]

Thủ tục ***Question\_GetById*** có chức năng truy xuất thông tin của một câu hỏi trong bảng Question dựa vào mã câu hỏi (***QuestionID***).

```
create proc [dbo].[Question_GetById]
@QuestionID int
As
SELECT [QuestionID]
      , [SubjectID], [QContent], [OptionA], [OptionB], [OptionC]
      , [OptionD], [Answer], [CreatedBy], [CreatedAt] , [ModifiedBy]
      , [ModifiedAt]
FROM [dbo].[Question]
where QuestionID = @QuestionID
Go
```

Thủ tục ***Question\_GetById*** giúp lấy chi tiết của một câu hỏi cụ thể từ cơ sở dữ liệu, hỗ trợ việc xem, hiển thị hoặc xử lý thông tin câu hỏi trong các tác vụ quản lý câu hỏi.

### 3.3. Thủ tục [dbo].[Question\_GetQuestionForTest]

Thủ tục ***Question\_GetQuestionForTest*** có chức năng lấy ngẫu nhiên một số lượng câu hỏi nhất định từ một chủ đề cụ thể trong bảng Question

```

CREATE proc [dbo].[Question_GetQuestionForTest]
@NumberOfQuestion int,
@SubjectID varchar(50)
as
Select *, N'Câu ' + CONVERT(varchar(20), ROW_NUMBER() over(order by
QuestionID)) QuestionIndex
From(
SELECT top(@NumberOfQuestion)
        [QuestionID], [SubjectID], [QContent], [OptionA], [OptionB],
        [OptionC]
        , [OptionD], [Answer]
FROM [dbo].[Question]
where SubjectID = @SubjectID
Order By NEWID()
) as TempQuestion
Go

```

Thủ tục ***Question\_GetQuestionForTest*** hữu ích trong việc tạo bài kiểm tra hoặc đề thi với số lượng câu hỏi ngẫu nhiên từ một chủ đề cụ thể. Nó đảm bảo rằng các câu hỏi không bị trùng lặp và được hiển thị một cách ngẫu nhiên cho mỗi lần thực thi.

### 3.4. Thủ tục [dbo].[Question\_Insert]

```

CREATE procedure [dbo].[Question_Insert]
@SubjectID varchar(50)
,@QContent nvarchar(500)
,@OptionA nvarchar(500)
,@OptionB nvarchar(500)
,@OptionC nvarchar(500)
,@OptionD nvarchar(500)
,@Answer nvarchar(500)
,@CreatedBy varchar(50)
as
INSERT INTO [dbo].[Question]
        ([SubjectID], [QContent], [OptionA], [OptionB], [OptionC]
        , [OptionD] , [Answer], [CreatedBy], [CreatedAt], [ModifiedBy],
        [ModifiedAt])
VALUES
        (@SubjectID, @QContent, @OptionA, @OptionB, @OptionC, @OptionD, @Answer, @CreatedBy
        , GETDATE(), @CreatedBy, GETDATE())
GO

```

Thủ tục ***Question\_Insert*** có chức năng thêm một câu hỏi mới vào bảng Question trong cơ sở dữ liệu.

Thủ tục ***Question\_Insert*** giúp thêm câu hỏi mới vào cơ sở dữ liệu một cách nhanh chóng và có tổ chức. Nó tự động ghi lại người tạo và thời gian tạo để dễ dàng theo dõi nguồn gốc và lịch sử chỉnh sửa của câu hỏi trong bảng Question.

### 3.5. Thủ tục [dbo].[Question\_Search]

Thủ tục ***Question\_Search*** có chức năng tìm kiếm các câu hỏi trong bảng Question dựa trên từ khóa được cung cấp

```
Create procedure [dbo].[Question_Search]
@Keyword nvarchar(200)
as
SELECT [QuestionID]
      , [SubjectID], [QContent], [OptionA], [OptionB], [OptionC]
      , [OptionD], [Answer], [CreatedBy], [CreatedAt], [ModifiedBy]
      , [ModifiedAt]
FROM [dbo].[Question]
where QContent like N'%' + @Keyword + '%'
or QuestionID like N'%' + @Keyword + '%'
or SubjectID like N'%' + @Keyword + '%'
or Answer like N'%' + @Keyword + '%'
GO
```

Thủ tục ***Question\_Search*** hỗ trợ tìm kiếm các câu hỏi nhanh chóng theo từ khóa trong cơ sở dữ liệu. Điều này hữu ích khi người dùng cần tra cứu các câu hỏi có liên quan dựa trên nội dung câu hỏi, mã câu hỏi, mã môn học, hoặc đáp án.

### 3.6. Thủ tục [dbo].[Question\_SelectAll]

Thủ tục ***Question\_SelectAll*** có chức năng lấy tất cả các câu hỏi trong bảng Question.

```
Create procedure [dbo].[Question_SelectAll]
as
SELECT [QuestionID]
      , [SubjectID] , [QContent], [OptionA], [OptionB], [OptionC]
      , [OptionD], [Answer], [CreatedBy], [CreatedAt], [ModifiedBy],
      [ModifiedAt]
FROM [dbo].[Question]
GO
```

Thủ tục ***Question\_SelectAll*** giúp lấy toàn bộ danh sách câu hỏi từ cơ sở dữ liệu một cách đơn giản, hữu ích cho các tình huống cần xem hoặc xuất toàn bộ



thông tin về câu hỏi, chẳng hạn khi hiển thị tất cả các câu hỏi trong giao diện quản lý hoặc để sao lưu.

### 3.7. Thủ tục [dbo].[Question\_Update]

Thủ tục *Question\_Update* có chức năng cập nhật thông tin của một câu hỏi trong bảng Question dựa trên *QuestionID*.

```
CREATE procedure [dbo].[Question_Update]
@QuestionID int,
    @SubjectID varchar(50)
    , @QContent nvarchar(500), @OptionA nvarchar(500), @OptionB nvarchar(500)
    , @OptionC nvarchar(500), @OptionD nvarchar(500), @Answer nvarchar(500)
    , @ModifiedBy varchar(50)
as
UPDATE [dbo].[Question]
SET [SubjectID] = @SubjectID
    , [QContent] = @QContent , [OptionA] = @OptionA, [OptionB] = @OptionB
    , [OptionC] = @OptionC, [OptionD] = @OptionD, [Answer] = @Answer
    , [ModifiedBy] = @ModifiedBy, [ModifiedAt] = GETDATE()
WHERE QuestionID = @QuestionID
GO
```

Thủ tục *Question\_Update* giúp sửa đổi thông tin của câu hỏi trong cơ sở dữ liệu khi cần thay đổi nội dung hoặc đáp án của câu hỏi. Nó hỗ trợ cập nhật thông tin một cách nhanh chóng và đảm bảo lịch sử chỉnh sửa được ghi nhận.

## 4. Thủ tục bảng Subject

### 4.1. Thủ tục [dbo].[Subject\_Delete]

Thủ tục *Subject\_Delete* có chức năng xóa một môn học khỏi bảng Subject dựa trên mã môn học (*SubjectID*).

```
create procedure [dbo].[Subject_Delete]
@SubjectID varchar(50)
as
delete from Subject
WHERE [SubjectID] = @SubjectID
GO
```

Thủ tục ***Subject\_Delete*** giúp xóa một môn học khỏi cơ sở dữ liệu, có thể dùng khi người dùng muốn loại bỏ một môn học không còn cần thiết hoặc đã bị xóa trong hệ thống.

## 4.2. Thủ tục [dbo].[Subject\_Insert]

Thủ tục ***Subject\_Insert*** có chức năng thêm một môn học mới vào bảng Subject trong cơ sở dữ liệu.

```
create procedure [dbo].[Subject_Insert]
@SubjectID varchar(50),
@SubjectName nvarchar(50),
@Description nvarchar(50),
@CreatedBy varchar(50),
@ModifiedBy varchar(50)
as
INSERT INTO [dbo].[Subject]
([SubjectID],
[SubjectName], [Description], [CreatedBy],
[ModifiedBy], [ModifiedAt], [CreatedAt])
VALUES
(@SubjectID,@SubjectName,@Description,
@CreatedBy,Getdate(),@ModifiedBy,
Getdate())
GO
```

Thủ tục ***Subject\_Insert*** hỗ trợ thêm một môn học mới vào cơ sở dữ liệu một cách dễ dàng và có tổ chức. Nó tự động ghi lại thông tin về người tạo và thời gian tạo để tiện cho việc theo dõi và quản lý dữ liệu trong bảng Subject.

## 4.3. Thủ tục [dbo].[Subject\_Search]

Thủ tục ***Subject\_Search*** có chức năng tìm kiếm các môn học trong bảng Subject dựa trên từ khóa cung cấp.

```

CREATE procedure [dbo].[Subject_Search]
@Keyword nvarchar(200)
as
SELECT [SubjectID]
      ,[SubjectName], [Description], [CreatedBy]
      ,[CreatedAt], [ModifiedBy], [ModifiedAt]
FROM [dbo].[Subject]
where SubjectName like N'%' + @Keyword + '%'
      or Description like N'%' + @Keyword + '%'
GO

```

Thủ tục ***Subject\_Search*** hỗ trợ người dùng tra cứu các môn học dựa trên tên hoặc mô tả. Nó giúp dễ dàng tìm kiếm và quản lý danh sách môn học khi người dùng cần lọc kết quả theo các từ khóa nhất định.

#### 4.4. Thủ tục [dbo].[Subject\_SelectAll]

Thủ tục ***Subject\_SelectAll*** có chức năng truy xuất tất cả các môn học từ bảng Subject.

```

Create procedure [dbo].[Subject_SelectAll]
As
SELECT [SubjectID]
      ,[SubjectName], [Description], [CreatedBy]
      ,[CreatedAt], [ModifiedBy], [ModifiedAt]
FROM [dbo].[Subject]
GO

```

Thủ tục ***Subject\_SelectAll*** hỗ trợ lấy danh sách đầy đủ các môn học, giúp dễ dàng hiển thị toàn bộ thông tin môn học trong giao diện người dùng hoặc thực hiện các thao tác quản lý môn học.

#### 4.5. Thủ tục [dbo].[Subject\_Update]

Thủ tục ***Subject\_Update*** có chức năng cập nhật thông tin của một môn học trong bảng Subject dựa trên mã môn học (***SubjectID***).

```

create procedure [dbo].[Subject_Update]
@SubjectID varchar(50),
@SubjectName nvarchar(50),
@Description nvarchar(50),
@ModifiedBy varchar(50)
as
UPDATE [dbo].[Subject]
SET [SubjectName] = @SubjectName
, [Description] = @Description
, [ModifiedBy] = @ModifiedBy
, [ModifiedAt] = GETDATE()
WHERE [SubjectID] = @SubjectID
GO

```

Thủ tục ***Subject\_Update*** hỗ trợ cập nhật thông tin môn học, giúp người dùng thay đổi tên, mô tả, và ghi nhận người thực hiện chỉnh sửa cũng như thời gian cập nhật.

## 5. Thủ tục của bảng TestHistory

### 5.1. Thủ tục [dbo].[TestHistory\_Insert]

Thủ tục ***TestHistory\_Insert*** có chức năng ghi lại thông tin lịch sử thi của một người dùng vào bảng ***TestHistory***.

```

create proc [dbo].[TestHistory_Insert]
@UserID int
, @SubjectID varchar(50)
, @TestDate datetime
, @CorrectAnswer int
, @TotalQuestion int
, @Mark int
, @CreatedBy varchar(50)
as
INSERT INTO [dbo].[TestHistory]
([UserID]
, [SubjectID], [TestDate], [CorrectAnswer], [TotalQuestion]
, [Mark], [CreatedBy], [CreatedAt], [ModifiedBy], [ModifiedAt])
VALUES
(@UserID
, @SubjectID, @TestDate, @CorrectAnswer, @TotalQuestion
, @Mark, @CreatedBy, GETDATE(), @CreatedBy, GETDATE())
GO

```

Thủ tục ***TestHistory\_Insert*** hỗ trợ lưu trữ thông tin kết quả thi của người dùng, bao gồm số lượng câu đúng, điểm số, ngày thi, và các thông tin liên quan, giúp quản lý lịch sử thi dễ dàng trong cơ sở dữ liệu.

### 5.2. Thủ tục [dbo].[TestHistory\_ReportBySubject]

Thủ tục ***TestHistory\_ReportBySubject*** có chức năng tạo báo cáo lịch sử thi theo môn học từ bảng TestHistory

```
CREATE proc [dbo].[TestHistory_ReportBySubject]
@SubjectID varchar(50)
as
begin
SELECT [TestId]
      ,[UserID]
      ,(Select Fullname from UserAccount where UserID = T.UserID) Fullname
      ,[SubjectID], [TestDate], [CorrectAnswer], [TotalQuestion]
      ,[Mark], [CreatedBy], [CreatedAt], [ModifiedBy], [ModifiedAt]
FROM [dbo].[TestHistory] T
where SubjectID = @SubjectID
end
GO
```

Thủ tục ***TestHistory\_ReportBySubject*** giúp tạo báo cáo chi tiết lịch sử thi cho một môn học cụ thể, hỗ trợ quản lý và phân tích kết quả thi của các người dùng theo môn học trong cơ sở dữ liệu.

### 5.3. Thủ tục [dbo].[TestHistory\_LeaderBoard]

Thủ tục ***TestHistory\_LeaderBoard*** có chức năng tạo bảng xếp hạng (leaderboard) cho một môn học dựa trên điểm số của người dùng trong các kỳ thi của môn đó.

```

CREATE proc [dbo].[TestHistory_LeaderBoard]
@SubjectID varchar(50)
as
begin
    with CteLeaderBoard
    as
    (
        SELECT [TestId]
            ,U.Username, U.Fullname, S.[SubjectID], S.SubjectName
            , [TestDate], [CorrectAnswer], [TotalQuestion] [Mark]
            , ROW_NUMBER() over(order by Mark Desc) MarkIndex
        FROM [dbo].[TestHistory] T
        INNER JOIN UserAccount U on T.UserID=U.UserID
        inner join Subject S on S.SubjectID = T.SubjectID
        where T.SubjectID = @SubjectID
    )

```

Thủ tục ***TestHistory\_LeaderBoard*** hỗ trợ quản lý kết quả thi một cách hiệu quả bằng cách cung cấp bảng xếp hạng cho một môn học cụ thể, giúp dễ dàng xem xét người dùng nào đạt điểm cao nhất và so sánh thành tích giữa các người dùng.

#### 5.4. Thủ tục [dbo].[TestHistory\_ReportByTime]

Thủ tục ***TestHistory\_ReportByTime*** có chức năng tạo báo cáo lịch sử thi của người dùng trong một khoảng thời gian cụ thể

```

CREATE proc [dbo].[TestHistory_ReportByTime]
@StartDate datetime,
@EndDate datetime
as
begin
SELECT [TestId]
      ,(Select Fullname from UserAccount where UserID = T.UserID) Fullname
      ,(Select SubjectName from Subject where SubjectID = T.SubjectID)
SubjectName
      ,Convert(varchar(10),[TestDate],103) TestDate
      ,[CorrectAnswer], [TotalQuestion], [Mark]
FROM [dbo].[TestHistory] T
where DATEDIFF(day,@StartDate,TestDate)>=0
And DATEDIFF(day,TestDate,@EndDate)>=0
end
GO

```

Thủ tục *TestHistory\_ReportByTime* giúp tạo báo cáo chi tiết các bài thi đã thực hiện trong một khoảng thời gian nhất định, hỗ trợ việc quản lý và theo dõi hoạt động thi cử của người dùng.

### 5.5. Thủ tục [dbo].[TestHistory\_SelectAll]

Thủ tục *TestHistory\_SelectAll* có chức năng truy vấn toàn bộ thông tin từ bảng *TestHistory*.

```

CREATE proc [dbo].[TestHistory_ReportByTime]
@StartDate datetime,
@EndDate datetime
as
begin
SELECT [TestId]
      ,(Select Fullname from UserAccount where UserID = T.UserID) Fullname
      ,(Select SubjectName from Subject where SubjectID = T.SubjectID)
SubjectName
      ,Convert(varchar(10),[TestDate],103) TestDate
      ,[CorrectAnswer], [TotalQuestion], [Mark]
FROM [dbo].[TestHistory] T
where DATEDIFF(day,@StartDate,TestDate)>=0
And DATEDIFF(day,TestDate,@EndDate)>=0
end
GO

```

Thủ tục *TestHistory\_SelectAll* hỗ trợ việc lấy danh sách toàn bộ bài thi từ cơ sở dữ liệu, giúp truy vấn nhanh và hiển thị dữ liệu phục vụ cho việc quản lý và theo dõi lịch sử thi của người dùng.

## 6. Thủ tục của bảng UserAccount

### 6.1. Thủ tục [dbo].[UserAccount\_ChangePassword]

Thủ tục *UserAccount\_ChangePassword* có chức năng thay đổi mật khẩu của người dùng trong bảng UserAccount.

```
Create procedure [dbo].[UserAccount_ChangePassword]
@username varchar(50),
@newPassword varchar(50)
AS
begin
    update UserAccount
    set Password = @newPassword
    where Username = @username
end
GO
```

Thủ tục *UserAccount\_ChangePassword* giúp thay đổi mật khẩu của người dùng một cách nhanh chóng, hỗ trợ việc quản lý bảo mật cho tài khoản của người dùng trong hệ thống.

### 6.2. Thủ tục [dbo].[UserAccount\_CheckExist]

Thủ tục *UserAccount\_CheckExist* có chức năng kiểm tra sự tồn tại của một tài khoản người dùng trong bảng UserAccount.

```
CREATE PROCEDURE [dbo].[UserAccount_CheckExist]
@Username varchar(50),
@Password varchar(50)
As
Begin
    Select UserID
    From UserAccount
    Where Username = @Username And Password = @Password
end
GO
```

Thủ tục *UserAccount\_CheckExist* hỗ trợ việc xác thực tài khoản, giúp xác minh thông tin đăng nhập của người dùng trong hệ thống. Thủ tục này thường được sử dụng trong quá trình đăng nhập để kiểm tra xem tài khoản và mật khẩu có hợp lệ không.

### 6.3. Thủ tục [dbo].[UserAccount\_Delete]



Thủ tục *UserAccount\_Delete* có chức năng xóa một tài khoản người dùng trong bảng UserAccount.

```
create procedure [dbo].[UserAccount_Delete]
@UserId int
AS
DELETE FROM [dbo].[UserAccount]
where UserID = @UserId
GO
```

Thủ tục *UserAccount\_Delete* giúp quản trị viên hoặc hệ thống quản lý có thể xóa tài khoản người dùng không còn hoạt động hoặc không cần thiết, hỗ trợ duy trì cơ sở dữ liệu sạch sẽ và quản lý tài khoản hiệu quả.

#### 6.4. Thủ tục [dbo].[UserAccount\_GetInforUser]

Thủ tục *UserAccount\_GetInforUser* có chức năng lấy thông tin chi tiết của người dùng từ bảng UserAccount dựa trên tên người dùng (Username).

```
create proc [dbo].[UserAccount_GetInforUser]
@Username nvarchar(200)
as
SELECT [UserID]
      ,[RoleID]
      ,[Username], [Password], [Fullname], [Email], [PhoneNumber]
      ,[Address], [Birthday], [Note], [CreatedBy], [CreatedAt]
      ,[ModifiedBy], [ModifiedAt]
FROM [dbo].[UserAccount]
where Username = @Username
GO
```

Thủ tục *UserAccount\_GetInforUser* giúp lấy thông tin chi tiết của một người dùng dựa trên tên đăng nhập của họ, hỗ trợ cho các chức năng quản lý người dùng, như chỉnh sửa thông tin, kiểm tra dữ liệu hoặc hỗ trợ người dùng.

#### 6.5. Thủ tục [dbo].[UserAccount\_Insert]

Thủ tục *UserAccount\_Insert* có chức năng thêm mới một tài khoản người dùng vào bảng UserAccount nếu tên người dùng (Username) chưa tồn tại.

```

CREATE procedure [dbo].[UserAccount_Insert]
@RoleID varchar(50),
  @Username nvarchar(50),
  @Password varchar(50),
  @Fullname nvarchar(50),
  @Email varchar(50),
  @PhoneNumber varchar(50),
  @Address nvarchar(150),
  @Birthday datetime,
  @Note varchar(50),
  @CreatedBy varchar(50),
  @ModifiedBy varchar(50)
AS
Begin
if(not exists (select * from UserAccount where Username = @Username))
INSERT INTO [dbo].[UserAccount]
  ([RoleID],
  [Username], [Password], [Fullname], [Email], [PhoneNumber],
  [Address], [Birthday], [Note], [CreatedBy], [CreatedAt],
  [ModifiedBy], [ModifiedAt])
VALUES
  (@RoleID, @Username, @Password, @Fullname, @Email, @PhoneNumber,
  @Address,
  @Birthday, @Note, @CreatedBy, GETDATE(), @ModifiedBy, GETDATE())
end
GO

```

Thủ tục *UserAccount\_Insert* giúp tạo tài khoản mới một cách an toàn, ngăn chặn việc tạo các tài khoản trùng tên đăng nhập, giúp duy trì tính nhất quán và toàn vẹn dữ liệu trong hệ thống quản lý người dùng.

## 6.6. Thủ tục [dbo].[UserAccount\_Search]

Thủ tục *UserAccount\_Search* có chức năng tìm kiếm các tài khoản người dùng trong bảng UserAccount dựa trên từ khóa và vai trò người dùng.

```

CREATE procedure [dbo].[UserAccount_Search]
@Keyword nvarchar(200),
@RoleID varchar(50)
as
SELECT UserID, RoleID, Username, Password, Fullname, Email, PhoneNumber
    , Address, Birthday, Note, CreatedBy, CreatedAt, ModifiedBy
    , ModifiedAt
FROM dbo.UserAccount
Where Username like '%' + @Keyword + '%'
    or Fullname like N '%' + @Keyword + '%'
    or PhoneNumber like '%' + @Keyword + '%'
    or Email like '%' + @Keyword + '%'
    or Address like N '%' + @Keyword + '%'
    and (@RoleID = 'all' Or RoleID = @RoleID)
GO

```

Thủ tục ***UserAccount\_Search*** giúp hỗ trợ tìm kiếm và lọc thông tin tài khoản người dùng một cách linh hoạt và hiệu quả, phù hợp với nhu cầu tra cứu và quản lý người dùng trong hệ thống.

## 6.7. Thủ tục [dbo].[UserAccount\_SelectAll]

Thủ tục **UserAccount\_SelectAll** có chức năng lấy tất cả các tài khoản người dùng từ bảng UserAccount.

```

create procedure [dbo].[UserAccount_SelectAll]
as
SELECT UserID
    , RoleID, Username, Password, Fullname, Email, PhoneNumber
    , Address, Birthday, Note, CreatedBy, CreatedAt, ModifiedBy
    , ModifiedAt
FROM dbo.UserAccount
GO

```

Thủ tục **UserAccount\_SelectAll** cung cấp cách thức nhanh chóng để lấy toàn bộ thông tin tài khoản người dùng trong hệ thống, hỗ trợ cho việc quản lý và tra cứu thông tin người dùng.

## 6.8. Thủ tục [dbo].[UserAccount\_Update]

Thủ tục **UserAccount\_Update** có chức năng cập nhật thông tin tài khoản người dùng trong bảng UserAccount.

```
Create procedure [dbo].[UserAccount_Update]
    @UserID int,
    @RoleID varchar(50),
    @Username nvarchar(50),
    @Password varchar(50),
    @Fullname nvarchar(50),
    @Email varchar(50),
    @PhoneNumber varchar(50),
    @Address nvarchar(150),
    @Birthday datetime,
    @Note varchar(50),
    @ModifiedBy varchar(50)
AS
Begin
    UPDATE [dbo].[UserAccount]
    SET [RoleID] = @RoleID
      ,[Username] = @Username
      ,[Password] = @Password
      ,[Fullname] = @Fullname
      ,[Email] = @Email
      ,[PhoneNumber] = @PhoneNumber
      ,[Address] = @Address
      ,[Birthday] = @Birthday
      ,[Note] = @Note
      ,[ModifiedBy] = @ModifiedBy
      ,[ModifiedAt] = GETDATE()
    where UserID = @UserID
end
GO
```

Thủ tục **UserAccount\_Update** giúp chỉnh sửa thông tin tài khoản người dùng một cách hiệu quả, bảo đảm rằng hệ thống có thể cập nhật thông tin người dùng khi có thay đổi, đồng thời ghi nhận người thực hiện thay đổi và thời gian thay đổi.

## 6.9. Thủ tục GetPasswordByEmail

Thủ tục này lấy mật khẩu của người dùng dựa trên địa chỉ email được cung cấp. Nếu email tồn tại và liên kết với một tài khoản, thủ tục sẽ trả về mật khẩu của người dùng. Thủ tục này được sử dụng trong quá trình khôi phục mật khẩu để cung cấp mật khẩu cho người dùng khi họ xác nhận mã OTP đúng.

```

CREATE PROCEDURE GetPasswordByEmail
    @Email NVARCHAR(100)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT Password
    FROM UserAccount
    WHERE Email = @Email;
END;
go

```

## 6.10. Thủ tục CheckEmailExists

Thủ tục này kiểm tra xem địa chỉ email người dùng nhập vào có tồn tại trong hệ thống hay không. Nếu email đã có trong cơ sở dữ liệu, thủ tục sẽ trả về giá trị true, ngược lại trả về false. Thủ tục này giúp đảm bảo rằng người dùng chỉ có thể nhận mã OTP nếu email tồn tại trong hệ thống.

```

CREATE PROCEDURE CheckEmailExists
    @Email NVARCHAR(100)
AS
BEGIN
    SELECT CASE WHEN EXISTS (SELECT 1 FROM UserAccount WHERE Email = @Email) THEN 1
    ELSE 0 END AS Exist
END
go

```

## 7. Thủ tục của bảng UserRole

### 7.1. Thủ tục [dbo].[UserRole\_SelectAll]

Thủ tục **UserRole\_SelectAll** có chức năng lấy tất cả các bản ghi từ bảng UserRole trong cơ sở dữ liệu.

```

Create procedure [dbo].[UserRole_SelectAll]
As
SELECT [RoleID]
      ,[RoleName]
      ,[Description]
      ,[CreatedBy]
      ,[CreatedAt]
      ,[ModifiedBy]
      ,[ModifiedAt]
FROM [dbo].[UserRole]
GO

```

Thủ tục này giúp lấy tất cả thông tin về các vai trò người dùng trong hệ thống, từ đó hỗ trợ việc hiển thị hoặc quản lý các vai trò người dùng trong ứng dụng.

### III. Database design và thư viện hỗ trợ

#### 1. Database design

**UserAccount:** UserID, RoleID, Username, Password, Fullname, Email, PhoneNumber, Address, Birthday, Note, CreateBy, CreateAt, ModifiedBy, ModifiedAt.

**UserRole:** RoleID, RoleName, Description, CreateBy, CreateAt, ModifiedBy, ModifiedAt.

**TestHistory:** TestID, UserID, SubjectID, TestDate, CorrectAnswer, TotalQuestion, Mark, CreateBy, CreateAt, ModifiedBy, ModifiedAt.

**Subject:** SubjectID, SubjectName, Description, CreateBy, CreateAt, ModifiedBy, ModifiedAt.

**Question:** QuestionID, SubjectID, Qcontent, QuestionA, QuestionB, QuestionC, QuestionD, Answer, CreateBy, CreateAt, ModifiedBy, ModifiedAt.

#### 2. Thư viện, công cụ hỗ trợ

**DevExpress:**

- **DevExpress** là một trong những bộ công cụ UI hàng đầu dành cho .NET, bao gồm nhiều thư viện và công cụ hỗ trợ phát triển giao diện cho WinForms, WPF, ASP.NET, MVC, Blazor và nhiều nền tảng khác.
- DevExpress cung cấp hàng trăm thành phần UI và các công cụ như GridControl, TreeList, Charts, Scheduler, và nhiều widget khác cho phép người dùng dễ dàng tạo ra các ứng dụng phức tạp và có hiệu suất cao.
- Ngoài các thành phần giao diện, DevExpress còn có các công cụ hỗ trợ phát triển như bộ thiết kế báo cáo (Report Designer) và công cụ biểu đồ mạnh mẽ.
- **Ưu điểm:** Chuyên nghiệp, phù hợp cho các dự án quy mô lớn, đáp ứng được nhiều yêu cầu phức tạp về giao diện và tính năng.



*Hình 4. Công cụ DevExpress*

### **Guna UI:**

- **Guna UI** là một thư viện UI được phát triển dành cho .NET WinForms. Nó cung cấp nhiều thành phần giao diện đẹp mắt như các nút bấm, thanh trượt, biểu đồ và các hiệu ứng khác, giúp làm cho các ứng dụng WinForms trông hiện đại và hấp dẫn hơn.

- Guna hỗ trợ nhiều tính năng và hiệu ứng UI mà WinForms tiêu chuẩn không có, như các hiệu ứng chuyển động, đổ bóng, và các kiểu dáng hiện đại theo xu hướng UI/UX.



*Hình 5. GunaUI Framework*

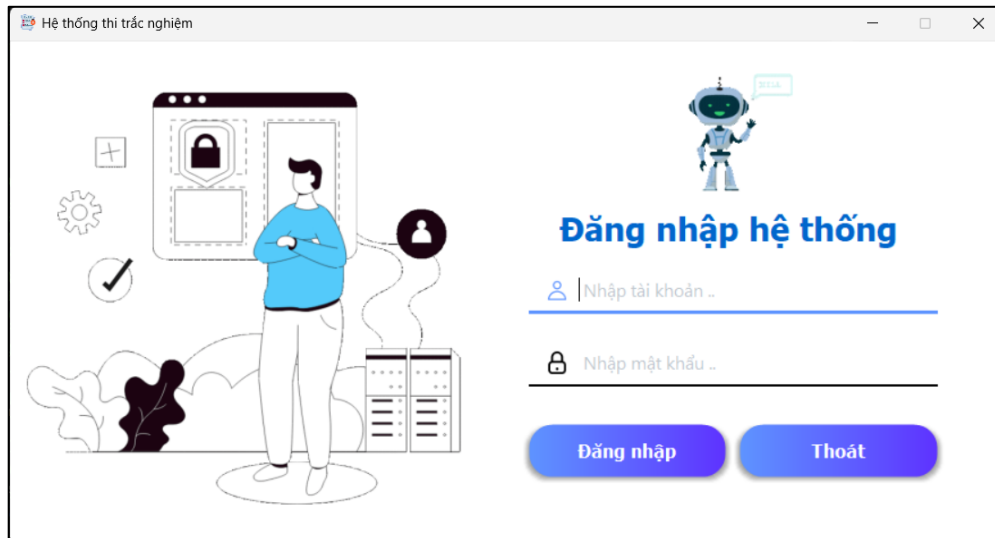
- **Ưu điểm:** Dễ sử dụng, tích hợp nhanh, không yêu cầu nhiều kỹ năng về thiết kế nhưng vẫn có thể tạo ra giao diện đẹp mắt.



## CHƯƠNG IV. THỰC HIỆN

### I. Giao diện thiết kế

#### 1. Giao diện form đăng nhập:



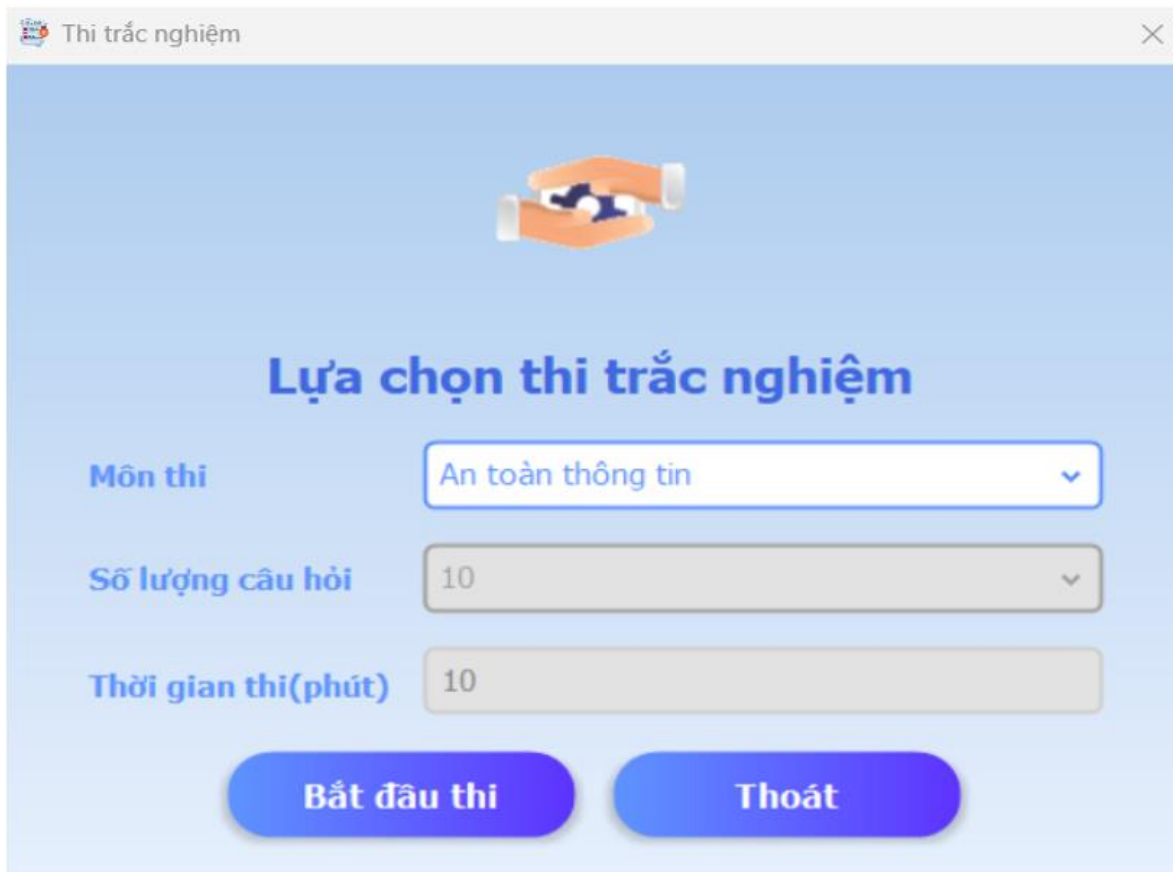
Hình 6. Giao diện đăng nhập

**Mô tả:** Người dùng nhập tên đăng nhập và mật khẩu để truy cập hệ thống và chức năng đóng thoát form.

#### Chức năng:

- Xác thực thông tin người dùng.
- Phân quyền: Giảng viên và sinh viên sẽ có giao diện riêng.
- Hiện thị thông báo khi đăng nhập sai thông tin.
- Đóng thoát form.

## 2. Giao diện form chọn môn thi:



Thi trắc nghiệm

**Lựa chọn thi trắc nghiệm**

Môn thi: An toàn thông tin

Số lượng câu hỏi: 10

Thời gian thi(phút): 10

Bắt đầu thi Thoát

Hình 7. Giao diện form chọn môn thi

**Mô tả:** Người dùng chọn môn học muốn thi từ danh sách.

**Chức năng:**

- Hiện thị danh sách các môn học đang có sẵn.
- Chỉ cho phép bắt đầu thi với môn có sẵn câu hỏi.

## 3. Giao diện form làm bài:

**Mô tả:** Hiện thị bộ câu hỏi của môn thi, thông tin của người thi và thời gian làm bài.

The screenshot shows a web application titled "Làm bài thi trắc nghiệm". On the left, there is a vertical list of questions from "Câu 1" to "Câu 10". The main area displays "Câu 1" with the question: "Giao thức nào không mã hóa dữ liệu khi truyền tải?". Below the question are four radio button options: A. TLS, B. FTP, C. HTTP, and D. SSH. At the top right, there are input fields for "Họ và tên" (Trần Công Minh), "Ngày sinh" (25/04/2004), "Môn thi" (An toàn thông tin), and "Số lượng câu hỏi" (10). A timer shows "09:52" with the text "Thời gian còn lại". At the bottom, there are navigation buttons: "<< Câu đầu", "< Câu trước", "Câu sau >", "Câu cuối >>", and a "Nộp bài" button. A status bar at the bottom right indicates "Số câu đã làm: 0/10".

Hình 8. Giao diện form làm bài

### Chức năng:

- Cho phép chọn đáp án và chuyển câu hỏi.
- Số câu đã làm và thời gian làm bài.
- Nộp bài sau khi hoàn thành bài thi.

## 4. Giao diện form kết quả thi:

The screenshot shows a web application titled "Kết quả thi trắc nghiệm". It features a header with a trophy icon and the text "KẾT QUẢ THI TRẮC NGHIỆM" with five stars. Below the header, there are several data fields: "Họ và tên" (Trần Công Minh), "Ngày sinh" (25/04/2004), "Môn thi" (Cơ sở dữ liệu), "Số câu hỏi" (10), "Số câu đúng" (5), and "Điểm số" (5.00). There is also a "Đánh giá" field with the text "Chúc mừng bạn đã qua môn". A blue "OK" button is centered at the bottom. A link "Bảng xếp hạng" is visible on the left side.

Hình 9. Giao diện form kết quả thi

**Mô tả:** Hiển thị kết quả thi sau khi hoàn thành bài thi.

### Chức năng chính:

- Hiển thị điểm số, thông tin người thi và trạng thái thi (đạt/không đạt).

## 5. Giao diện form quản lý người dùng (Admin):

The screenshot displays the 'Quản lý người dùng' (Manage Users) interface. At the top, there's a navigation bar with icons for 'Hệ thống', 'Danh mục', 'Chức năng', and 'Báo cáo - thống kê'. A sidebar on the left contains icons for 'Thay đổi mật khẩu người dùng', 'Sao lưu dữ liệu', and 'Phục hồi dữ liệu'. The main content area features a 'Thông tin người dùng' (User Information) form with fields for 'Mã người dùng' (User ID), 'Họ và tên' (Full Name), 'Số điện thoại' (Phone Number), 'Chức vụ' (Position), 'Mật khẩu' (Password), 'Ngày sinh' (Date of Birth), 'Tài khoản' (Username), 'Email', and 'Địa chỉ' (Address). Below the form are buttons for 'Thêm' (Add), 'Sửa' (Edit), and 'Xóa' (Delete). A search bar with a 'Tìm kiếm' (Search) button is also present. At the bottom, a table lists the current users.

STT	Họ và tên	Tài khoản	Mã người dùng	Số điện thoại	Email	Địa chỉ	Ngày sinh	Chức vụ
01	Hoàng Kim Anh	kmanh	18	0777999888	ahnzy0407@gmail.com	TP HCM	7/4/2004	Teacher
02	Trần Công Minh	2001222	20	0954876214	tcngminh1604@gmail.com	Hà Nội	4/25/2004	User
03	Admin	Admin	21	0777999496	admin@gmail.com	Hà Nội	3/9/2004	Admin

Hình 10. Giao diện form kết quả thi

**Mô tả:** Chỉ dành cho giảng viên, quản lý thông tin người dùng trong hệ thống.

**Chức năng :**

- Thêm, xóa và chỉnh sửa thông tin sinh viên hoặc giảng viên.
- Tìm kiếm và phân quyền người dùng.

## 6. Giao diện form quản lý câu hỏi:

**Mô tả:** Dành cho giảng viên, quản lý ngân hàng câu hỏi của từng môn

Hệ thống quản lý thi trắc nghiệm

Xin chào, Admin

Quản lý câu hỏi

Thông tin câu hỏi

Mã câu hỏi: 708

Nội dung câu hỏi: Khái niệm nào mô tả việc phân chia CPU giữa các tiến trình?

Môn thi: Hệ điều hành

Đáp án đúng: Multitasking

Lựa chọn A: Multitasking

Lựa chọn B: Multithreading

Lựa chọn C: Scheduling

Lựa chọn D: Virtualization

Thêm Sửa Xóa

Nhập nội dung cần tìm... Tìm kiếm

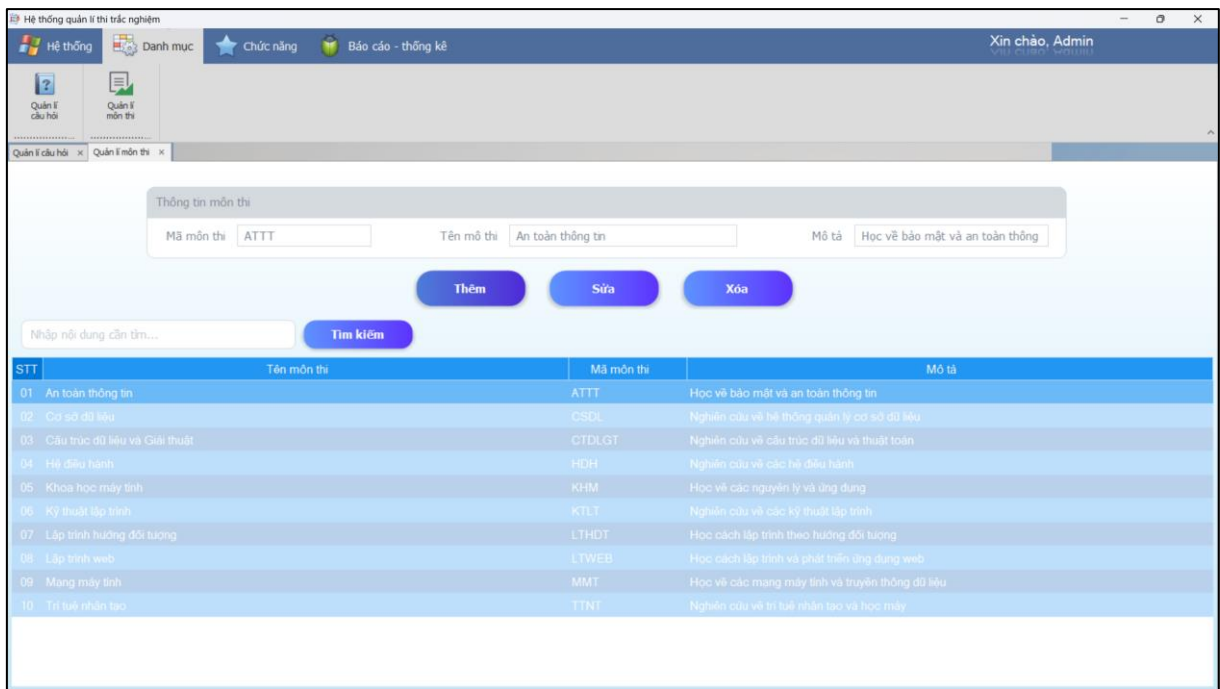
STT	Mã câu hỏi	Môn thi	Nội dung câu hỏi	Lựa chọn A	Lựa chọn B	Lựa chọn C	Lựa chọn D	Đáp án đúng
01	708	HDH	Khái niệm nào mô tả việc phân chia CPU	Multitasking	Multithreading	Scheduling	Virtualization	Multitasking
02	709	HDH	Khái niệm nào mô tả việc chuyển đổi CPU	Context Switching	Multithreading	Scheduling	Virtualization	Context Switching
03	710	HDH	Khái niệm nào mô tả việc quản lý bộ nhớ ảo	Paging	Segmentation	Swapping	Virtualization	Paging
04	711	HDH	Khái niệm nào mô tả việc phân chia bộ nhớ	Paging	Segmentation	Swapping	Virtualization	Paging
05	712	HDH	Khái niệm nào mô tả việc phân chia bộ nhớ	Segmentation	Paging	Swapping	Virtualization	Segmentation
06	713	HDH	Khái niệm nào mô tả việc chuyển đổi giữa	Context Switching	Multithreading	Scheduling	Virtualization	Context Switching
07	714	HDH	Khái niệm nào mô tả việc quản lý bộ nhớ ảo	Virtual Memory	Physical Memory	Cache Memory	RAM	Virtual Memory
08	715	HDH	Khái niệm nào mô tả việc quản lý các tiến trình	Process Management	Memory Management	File Management	Device Management	Process Management

Hình 11. Giao diện quản lý câu hỏi

### Chức năng chính:

- Thêm mới, chỉnh sửa hoặc xóa câu hỏi.
- Gán câu hỏi vào môn học cụ thể.
- Kiểm tra đáp án đúng và đảm bảo không trùng lặp câu hỏi.

## 7. Giao diện form quản lý môn thi:



Hình 12. Giao diện quản lý môn thi

**Mô tả:** Giúp giảng viên quản lý các môn học có trong hệ thống.

**Chức năng:**

- Thêm hoặc xóa môn học.
- Xem số lượng câu hỏi trong từng môn.

## 8. Giao diện form báo cáo thống kê theo môn:

The screenshot shows a web application for HUIT (Ho Chi Minh City University of Industry and Trade). The interface includes a top navigation bar with links like 'Hệ thống', 'Danh mục', 'Chức năng', and 'Bảo cáo - thống kê'. A sidebar on the left contains 'Quản lý câu hỏi', 'Quản lý môn thi', and 'Bảo cáo điểm theo môn thi'. The main content area displays a report titled 'BÁO CÁO ĐIỂM THI THEO MÔN' for the subject 'LTHDT'. The report includes a table with columns: Mã đề thi, Mã sinh viên, Mã sinh viên, Ngày thi, Số câu đúng, Tổng số câu, and Điểm. Below the table, it shows the average score 'Điểm thi trung bình: 0,20' and the date 'TP.HCM, ngày 17 tháng 11 năm 2024'. The report is signed by 'Admin'.

Mã đề thi	Mã sinh viên	Mã sinh viên	Ngày thi	Số câu đúng	Tổng số câu	Điểm
1018	21	Admin	30/09/2024	0	10	0,00
1019	21	Admin	30/09/2024	0	10	0,00
1033	21	Admin	30/09/2024	0	10	0,00
1034	21	Admin	30/09/2024	0	10	0,00
1042	21	Admin	30/09/2024	1	10	1,00

Điểm thi trung bình: 0,20  
TP.HCM, ngày 17 tháng 11 năm 2024  
Người lập báo cáo (Ký, ghi rõ họ tên)  
Admin

Hình 13. Giao diện báo cáo - thống kê theo môn

**Mô tả:** Hiện thị báo cáo kết quả thi của từng môn học.

**Chức năng:**

- Thống kê số lượng sinh viên tham gia và điểm trung bình.
- Xuất báo cáo dưới dạng PDF hoặc Excel (bằng DevExpress).

## 9. Giao diện form báo cáo thống kê theo thời gian:

Mã đề	Tên thí sinh	Môn thi	Ngày thi	Số câu đúng	Tổng số câu	Điểm
5	Admin	An toàn thông tin	26/09/2024	1	10	1.00
9	Admin	Cấu trúc dữ liệu và Giải thuật	28/09/2024	19	45	4.00
1006	Admin	Lập trình web	29/09/2024	5	10	5.00
1007	Admin	An toàn thông tin	30/09/2024	0	10	0.00
1008	Admin	An toàn thông tin	30/09/2024	0	10	0.00
1009	Admin	An toàn thông tin	30/09/2024	1	10	1.00
1010	Admin	An toàn thông tin	30/09/2024	2	10	2.00
1011	Admin	An toàn thông tin	30/09/2024	2	10	2.00
1012	Admin	Kỹ thuật lập trình	30/09/2024	1	20	0.00
1013	Admin	Kỹ thuật lập trình	30/09/2024	1	20	0.00
1014	Admin	Cơ sở dữ liệu	30/09/2024	0	20	0.00
1015	Admin	Cơ sở dữ liệu	30/09/2024	0	20	0.00
1016	Admin	Mạng máy tính	30/09/2024	0	20	0.00
1017	Admin	Mạng máy tính	30/09/2024	0	20	0.00
1018	Admin	Lập trình hướng đối tượng	30/09/2024	0	10	0.00
1019	Admin	Lập trình hướng đối tượng	30/09/2024	0	10	0.00
1020	Admin	Mạng máy tính	30/09/2024	5	10	5.00

Hình 14. Giao diện báo cáo - thống kê theo thời gian

**Mô tả:** Thống kê kết quả thi trong khoảng thời gian cụ thể.

**Chức năng:**

- Lọc và thống kê theo ngày, tháng, năm.
- Hiển thị biểu đồ trực quan (sử dụng **DevExpress**).

## 10. Giao diện form sao lưu dữ liệu và phục hồi:

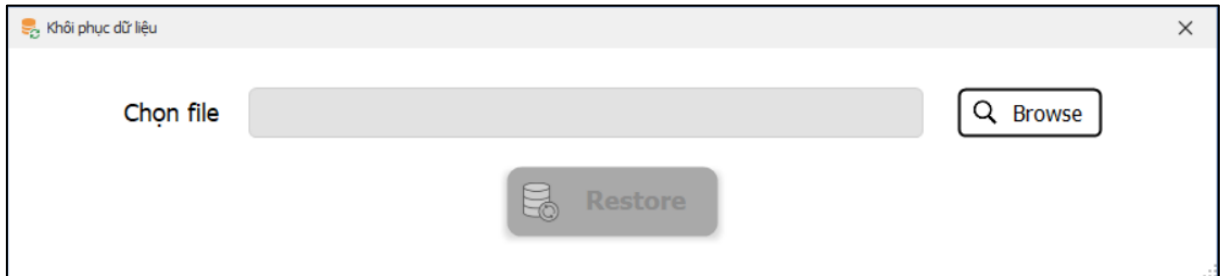
Hình 15. Giao diện sao lưu dữ liệu

**Mô tả:** Cho phép người dùng sao lưu dữ liệu hệ thống.

**Chức năng chính:**



- Chọn vị trí lưu file sao lưu.
- Tạo bản sao lưu toàn bộ dữ liệu để khôi phục khi cần.



*Hình 16. Giao diện phục hồi dữ liệu*

**Mô tả:** Giúp khôi phục dữ liệu từ file sao lưu.

**Chức năng:**

- Chọn file sao lưu để khôi phục dữ liệu.
- Kiểm tra tính toàn vẹn của dữ liệu trước khi phục hồi.

## CHƯƠNG V. KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN

### I. Kết luận

Nói chung, đề án xây dựng ứng dụng thi trắc nghiệm đã hoàn thành như mong đợi của chúng em, cung cấp được các chức năng cơ bản thi, chấm thi dành cho sinh viên và giảng viên. Phần mềm ứng dụng quản lý thi trắc nghiệm không những tạo ra hệ thống ứng dụng tiện lợi cho sinh viên, giảng viên làm bài thi và chấm thi mà còn có những chức năng để theo dõi các báo cáo, thống kê về số điểm của từng môn học. Đây là nền tảng cho sự phát triển của ứng dụng quản lý thi trắc nghiệm và là cơ hội cho các cải tiến mới trong tương lai.

Ngoài ra, qua quá trình phát triển đề án vẫn còn nhiều thiết sót, nhóm chúng em đã có cơ hội trau dồi thêm nhiều kỹ năng quan trọng như phân tích yêu cầu, thiết kế cơ sở dữ liệu, lập trình giao diện, và tối ưu hóa hiệu suất hệ thống. Đây là những kinh nghiệm quý báu giúp chúng em có được cái nhìn sâu sắc hơn về việc phát triển phần mềm ứng dụng thực tế

### II. Định hướng phát triển

Trong tương lai, nhóm chúng em sẽ tiếp tục cải tiến và mở rộng ứng dụng thi trắc nghiệm với những mục tiêu chính sau:

- **Tích hợp các công nghệ tiên tiến:** Trong những phiên bản tiếp theo, nhóm dự định tích hợp thêm trí tuệ nhân tạo (AI) và học máy (Machine Learning) để hỗ trợ việc phân tích dữ liệu điểm thi và gợi ý cải thiện kết quả học tập cho sinh viên.
- **Cải thiện tính năng bảo mật:** Đảm bảo an toàn và bảo mật dữ liệu người dùng, đặc biệt là trong việc lưu trữ thông tin cá nhân và kết quả thi cử. Nhóm sẽ nghiên cứu các phương pháp bảo mật dữ liệu hiện đại để nâng cao độ tin cậy của hệ thống.

- **Hỗ trợ đa nền tảng:** Phát triển thêm các phiên bản ứng dụng cho các nền tảng khác nhau như mobile (iOS, Android), giúp sinh viên và giảng viên dễ dàng truy cập và sử dụng hệ thống trên mọi thiết bị.

- **Mở rộng quy mô:** Định hướng phát triển thành một hệ thống quản lý thi trắc nghiệm toàn diện có thể áp dụng cho nhiều trường học, cơ sở giáo dục và cả những tổ chức ngoài giáo dục có nhu cầu đánh giá và kiểm tra năng lực.

## TÀI LIỆU THAM KHẢO

1. Luanvan.net.vn. (n.d.). *Đề tài xây dựng phần mềm quản lý và thi trắc nghiệm trực tuyến*. Retrieved from <https://luanvan.net.vn/luan-van/de-tai-xay-dung-phan-mem-quan-ly-va-thi-trac-nghiem-truc-tuyen-32316/>
2. Slideshare. (n.d.). *Đề tài xây dựng phần mềm thi đánh giá câu hỏi, đề thi trắc nghiệm*. Retrieved from <https://fr.slideshare.net/slideshow/de-tai-xay-dung-phan-mem-thi-danh-gia-cau-hoi-de-thi-trac-nghiem/238311525>
3. Gunau. (n.d.). Retrieved from <https://gunau.com/>
4. DevExpress. (n.d.). Retrieved from <https://www.devexpress.com/>

## PHỤ LỤC

### Danh mục hình ảnh

Hình 1. Use case Admin/Teacher/User.....	11
Hình 2. Mô hình ERD.....	12
Hình 3. Mô hình SQL.....	14
Hình 4. Mô hình CDM.....	15
Hình 5. GunaUI Framework .....	34
Hình 6. Giao diện đăng nhập .....	35
Hình 7. Giao diện form chọn môn thi.....	36
Hình 8. Giao diện form làm bài.....	37
Hình 9. Giao diện form kết quả thi.....	37
Hình 10. Giao diện form kết quả thi.....	38
Hình 11. Giao diện quản lý câu hỏi.....	39
Hình 12. Giao diện quản lý môn thi.....	40
Hình 13. Giao diện báo cáo - thống kê theo môn.....	41
Hình 14. Giao diện báo cáo - thống kê theo thời gian.....	42
Hình 15. Giao diện sao lưu dữ liệu .....	42
Hình 16. Giao diện phục hồi dữ liệu .....	43

