**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**Vũ Thanh Thiên**

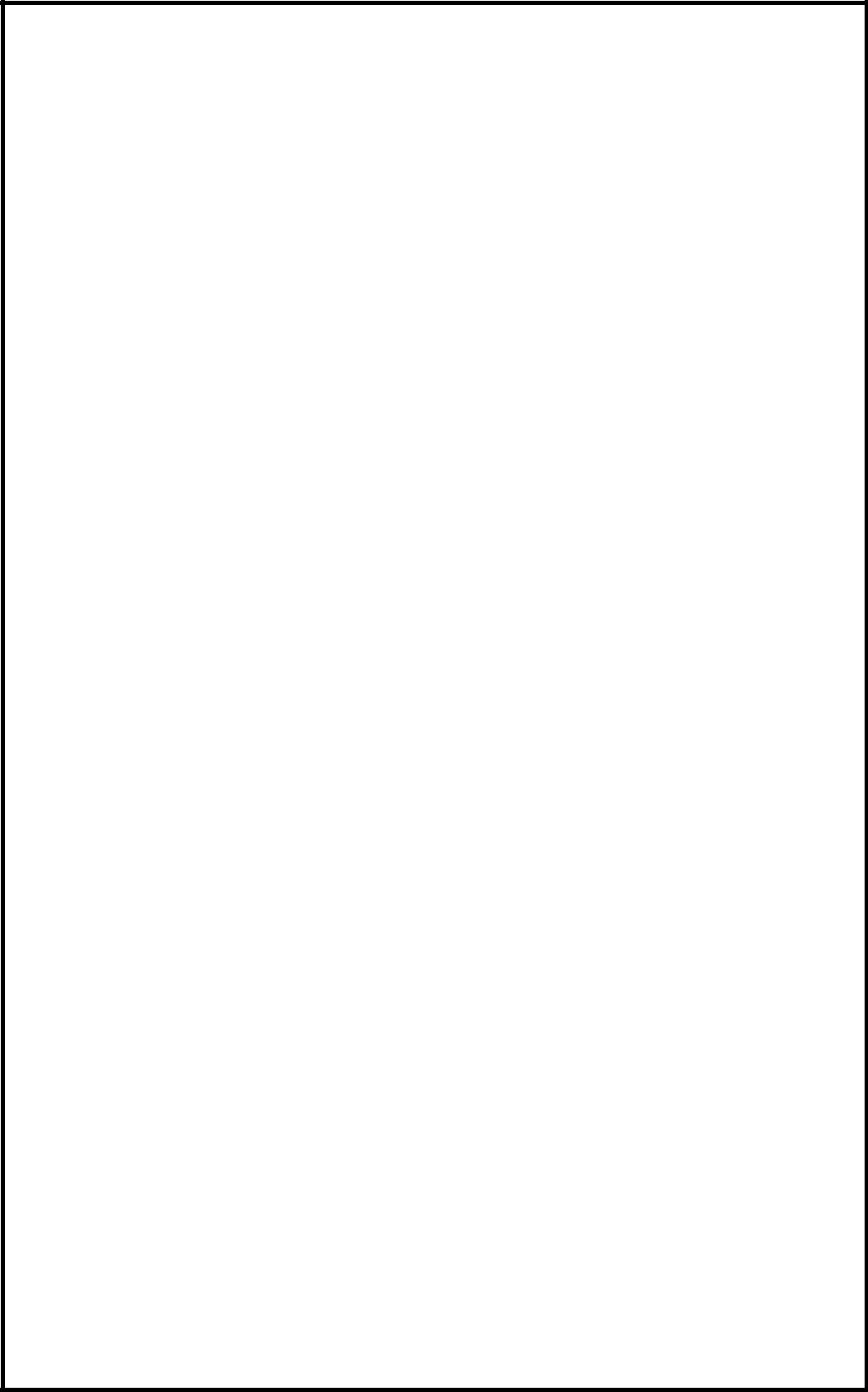
**XÂY DỰNG TRANG WEB QUẢN LÝ ĐỀ TÀI KHOA HỌC CÔNG NGHỆ CHO**

**TRƯỜNG ĐẠI HỌC**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Công nghệ thông tin**

**HÀ NỘI - 2021**



**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**Vũ Thanh Thiên**

**XÂY DỰNG TRANG WEB QUẢN LÝ ĐỀ TÀI KHOA HỌC CÔNG NGHỆ CHO**

**TRƯỜNG ĐẠI HỌC**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Công nghệ thông tin**

**Cán bộ hướng dẫn**:TS.Nguyễn Hoài Sơn

**Cán bộ đồng hướng dẫn**:

**HÀ NỘI - 2021**

# Chương I: Mở đầu

## Đặt vấn đề

Xã hội ngày càng phát triển, cuộc sống của con người cũng ngày càng được nâng cao và với thời đại 4.0 hiện tại thì công nghệ là một trong những lĩnh vực được quan tâm nhất. Công nghệ đã và đang được ứng dụng rất nhiều trong mọi khía cạnh của đời sống con người như: Nông nghiệp, công nghiệp, du lịch, khoa học,… Với tốc độ phát triển hiện tại của khoa học công nghệ thì việc ứng dụng công nghệ vào công việc là hết sức cần thiết, với mục đích nhằm nâng cao hiệu quả và tăng cường sự thuận lợi trong công việc. Thời đại công nghệ số thì việc ứng dụng các phần mềm công nghệ trong việc quản lí và triển khai công việc là hết sức cấp thiết bởi vì nếu như chỉ xử lí công việc một cách thủ công, truyền thống thì hiệu quả đem lại trong công việc sẽ là rất thấp và điều đó có ảnh hưởng rất lớn đến chất lượng công việc cũng như tốc độ phát triển của con người. Ứng dụng công nghệ vào công việc là điều mà con người đã và đang hướng đến bởi vì nó không chỉ giúp con người giảm bớt đi sức lao động mà còn nâng cao hiệu quả công việc, giúp cho con người quản lí tốt hơn cũng như dễ dàng tìm kiếm hơn những vấn đề mà họ quan tâm. Song song với sự phát triển và mở rộng hiểu biết của con người là sự ra đời của hàng loạt những phát minh và nghiên cứu khoa học với những mục đích khác nhau nhưng tựu chung lại cũng chỉ có duy nhất một mục đích là nâng cao chất lượng đời sống con người.

Với sứ mạng của Đại học Công Nghệ là đào tạo nguồn nhân lực chất lượng cao, trình độ cao và bồi dưỡng nhân tài; nghiên cứu phát triển và ứng dụng các lĩnh vực khoa học công nghệ tiên tiến mũi nhọn trên cơ sở phát huy thế mạnh về Công nghệ thông tin và Truyền thông cũng như tiên phong tiếp cận chuẩn mực giáo dục đại học khu vực và thế giới. Việc ứng dụng công nghệ thông tin trong quản trị đại học đóng góp tích cực vào sự phát triển nền kinh tế và xã hội tri thức của đất nước. Cùng với đó, với vị thế là một trong những trường Đại học đi đầu trong lĩnh vực Công nghệ thông tin và Truyền thông thì việc hoàn thiện bộ máy quản lý một cách hoàn hảo là điều cần thiết.Hàng năm nhà trường đều triển khai rất nhiều nhiệm vụ, đề tài nghiên cứu khóa học, vì thế việc hoàn thiện bộ máy quản lý việc Nghiên cứu khoa học cũng là một mục tiêu quan trọng cần được thực hiện.

## Ý nghĩa

Cùng với sự ra đời của hàng loạt đề tài nghiên cứu khoa học thì sự ra đời của “Web quản lí nghiên cứu khoa học” cũng là hết sức cần thiết và nó đem lại rất nhiều những ý nghĩa thiết thực cho hoạt động của những tổ chức nghiên cứu cấp cơ sở nói chung và Đại học Công Nghệ nói riêng.

Lợi ích thứ hai là sự chính xác, chúng ta không cần lo lắng đến việc nhầm lẫn về thủ tục giấy tờ nữa, thay vào đó, sự chính xác về ngày tháng, con người, tiến độ, con số, kinh phí đầu tư,… đều được lưu lại một cách rõ ràng, chi tiết.

Lợi ích thứ ba của đề tài này là sự chuyên nghiệp. Một trường Đại học đi đầu trong lĩnh vực Công nghệ thông tin, mỗi năm có đến hàng trăm đề tài nghiên cứu nhưng việc quản lý vẫn còn thủ công thì đó thực sự là một thiếu sót. Ngoài ra còn nhiều những lợi ích khác mà Website này mang lại.

## Mục tiêu

Mục tiêu mà “Web quản lí nghiên cứu khoa học” hướng tới là trở thành công cụ giúp nhà trường dễ dàng quản lý việc nghiên cứu như con người, kinh phí, tiến độ, đánh giá,...

## Cấu trúc dự kiến

* Chương I: Tổng quan

Nêu các vấn đề tổng quan, bài toán đặt ra và mục tiêu của khóa luận

* Chương II: Phân tích và thiết kế

1. Phân tích nghiệp vụ, chức năng của hệ thống
2. Thiết kế kiến trúc hệ thống
3. Mô hình hóa yêu cầu và biểu đồ ca sử dụng
4. Xây dựng cơ sở dữ liệu

* Chương III: Giới thiệu các công nghệ sử dụng.
* Chương IV: Cài đặt và triển khai hệ thống.
* Chương V: Tổng kết.

# Chương II. Phân tích và thiết kế hệ thống

## Mục tiêu

Mục tiêu hàng đầu của một phần mềm nói chung và phần mềm quản lý nghiên cứu khoa học nói chung là trở thành một phần mềm tốt và thực sự hữu dụng. Đối với Đại học Công Nghệ, phần mềm cần có ý nghĩa khoa học cao, giải quyết các vấn đề mũi nhọn của ngành các vấn đề khoa học lớn có tính đa ngành và liên ngành, có ý nghĩa lý luận và theo hướng ưu tiên của Nhà nước, của ĐHQGHN .

Mục tiêu thứ hai của phần mềm này là sự dễ vận hành. Nhận thức được sự phức tạp của việc “quản lý nghiên cứu khoa học” trong thực tế, phần mềm này được sinh ra để hỗ trợ việc quản lý ấy trở nên thực sự đơn giản. Trọng tâm của phần mềm sẽ đặt vào sự nhanh gọn, dễ hiểu, dễ dùng.

Mục tiêu cuối cùng của phần mềm đó là tính an toàn và độ tin cậy. Vì phần mềm quản lý về những đề tài nghiên cứu khoa học cho nên việc dễ dàng để lộ thông tin là một điều không đáng có. Ngoài ra, việc hạn chế tối đa lỗi lầm cũng là mục tiêu mà mọi phần mềm hướng tới.

## Phân tích hệ thống

### Phân tích chi tiết yêu cầu nghiệp vụ

Các hoạt động nghiệp vụ thông thường của việc quản lý nghiên cứu khoa học sẽ như sau :

* Người dùng muốn đăng ký đề tài nghiên cứu khoa học thì điền form đăng ký, sau đó lưu vào hệ thống.
* Phòng KHCN-HTPT sẽ tổ chức hội đồng xét chọn đề tài, thông báo thời gian, địa điểm đánh giá trước đánh giá và điểm số cũng như quyết định phê duyệt khi đã xét chọn xong.
* Những thông tin về đề tài như tên đề tài, nội dung đề tài, cán bộ nghiên cứu đề tài, kinh phí, thời hạn,… sẽ được lưu lại.
* Khi đề tài đến hạn nghiệm thu thì phòng KHCN-HTPT sẽ lập hội đồng đánh giá, thông báo thời gian, địa điểm, cách thức đánh giá đến những cá nhân liên quan
* Đánh giá cán bộ hoàn thành nhiệm vụ.

### Yêu cầu hệ thống

1. Hệ thống quản lý nghiên cứu khoa học được sinh ra để giải quyết những yêu cầu chức năng sau:

* Giúp cán bộ nghiên cứu khoa học tra cứu đề tài đã thực hiện trước đó.
* Giúp cán bộ nghiên cứu tạo đăng ký nghiên cứu.
* Giúp phòng KHCN-HTPT quản lý đề tài, quản lý văn bản, đơn xin,quyết định,báo cáo,…
* Giúp phòng KHCN-HTPT quản lý trạng thái của những đề tài đang thực hiện.
* Thông tin về các lần chỉnh sửa của người dùng đều được lưu lại một cách chính xác, khách quan.
* Thống kê

1. Yêu cầu phi chức năng:

* Giao diện rõ ràng, mạch lạc. Font chữ đơn giản, màu sắc không quá nhiều.
* Dễ dàng sử dụng, các chức năng cần được mô tả rõ ràng
* An toàn và bảo mật dữ liệu tốt.
* Trang web tương thích với các trình duyệt phổ biến hiện tại.
* Hiệu năng tốt, phản hồi cần nhanh và chính xác

### 2.3 Sơ đồ phân cấp chức năng

1. Xác định các chức năng của hệ thống:

Mô tả chức năng:

* Quản trị hệ thống: admin sử dụng chức năng quản trị hệ thống để quản lý thông tin người dùng và phân quyền sử dụng cho người dùng.
* Quản lý đề tài: Hệ thống cho phép tra cứu, thêm, sửa, xóa tất cả những đề tài với tât cả thông tin về đề tài ấy.
* Quản lý cán bộ nghiên cứu: hệ thống cho phép tra cứu, thêm, sửa, xóa thông tin của những cãn bộ nghiên cứu tương ứng với đề tài.

**Hình 1. Sơ đồ phân cấp chức năng**

## Mô hình hóa nhu cầu với biểu đồ ca sử dụng

* 1. **Các tác nhân của hệ thống :**
* Tác nhân Admin: Người thực hiện các chức năng quản trị hệ thống, có thể thêm mới tài khoản User; thêm, sửa, xóa thông tin đề tài; thêm, sửa, xóa thông tin người nghiên cứu.
* Tác nhân User: Người sử dụng thực hiện các chức năng thêm mới đề tài, gửi yêu cầu cho Admin (sửa, xóa thông tin đề tài mà họ nghiên cứu).

**3.2 Bảng chức năng của hệ thống**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên chức năng | Mô tả | Quyền |
| 1 | Tạo tài khoản User | Khi Admin đăng nhập vào hệ thống thì có thể tạo mới tài khoản cho user | Admin |
| 2 | Đăng nhập | Người dùng sử dụng tài khoản/ mật khẩu được cấp để đăng nhập vào hệ thống | Admin/ User |
| 3 | Xem và tìm kiếm danh sách đề tài | Người dùng có thể xem thông tin của đề tài và tìm kiếm những đề tài đó theo các trường tìm kiếm định sẵn | Admin/ User |
| 4 | Đăng ký đề tài | User sau khi đăng nhập tạo đề tài nghiên cứu và gửi lên hệ thống | User |
| 5 | Sửa thông tin đề tài | Chỉnh sửa thông tin của đề tài | Admin |
| 6 | Xóa thông tin của đề tài | Xóa thông tin về đề tài | Admin |
| 7 | Chỉnh sửa thông tin cán bộ nghiên cứu | Người dùng có thể thay đổi thông tin của cán bộ nghiên cứu | Admin/ User |
| 8 | Xóa thông tin của cán bộ nghiên cứu | Admin có thể xóa thông tin của cán bộ nghiên cứu | Admin |
| 9 | Thống kê | Admin có thể thống kê thông tin của các đề tài đang hoạt động | Admin |

* 1. **Mô tả ca sử dụng :**

3.2.1 Đăng nhập

* Mô tả: người dùng đăng nhập với tài khoản, mật khẩu được cấp sẵn. Sau khi đăng nhập mới có thể làm được một số hành động tiếp theo.
* Tác nhân: Admin/ User.
* Tiền điều kiện: Cần có tài khoản, mật khẩu.
* Hậu điều kiện: Đăng nhập với tài khoản có quyền nào thì sẽ chỉ được thực hiện những hành động được phép với quyền tương ứng.
* Luồng sự kiện:
  + Người dùng vào trang web thì sẽ hiển thị form đăng nhập.
  + Điền đầy đủ thông tin username và password thì chuyền đến trang màn hình chính.
* Luồng thay thế: Không có.
* Yêu cầu đặc biệt: Không có.
* Luồng ngoại lệ: Nhập sai thông tin đăng nhập thì được thông báo nhập sai và yêu cầu nhập lại mới được đăng nhập.

3.2.2 Tạo tài khoản người dùng

* Mô tả: Admin sẽ tạo tài khoản cho người dùng với quyền của người dùng là User, tài khoản người dùng này dùng để đăng nhập, thêm đề tài nghiên cứu,…Vì tài khoản này cần được cấp nên không cho người dùng thông thường tự động đăng ký.
* Tác nhân: Admin.
* Tiền điều kiện: Người dùng cần đăng nhập và có quyền Admin mới được phép thực hiện hành động này.
* Hậu điều kiện: Tài khoản được tạo ra có đầy đủ thông tin đăng nhập và thông tin cơ bản của người dùng đó.
* Luồng sự kiện:
  + Đăng nhập với tài khoản Admin.
  + Chọn vào mục Người dùng -> chọn thêm mới.
  + Điền đầy đủ thông tin cần thiết gồm thông tin tài khoản (username, password, confirm password, email) và thông tin người dùng (Họ tên, mã cán bộ, …).
  + Bấm lưu - > Thành công thì tài khoản sẽ được tạo.
* Luồng thay thế: Khi điền thiếu các trường bắt buộc nhập sẽ yêu cầu nhập lại. Nếu tài khoản đã tạo bị trùng thì cũng phải nhập lại.
* Yêu cầu đặc biệt: Tạo tài khoản với quyền Admin – Vì số lượng tài khoản với quyền Admin rất ít nên sẽ thêm thủ công.

3.2.3 Xem và tìm kiếm danh sách đề tài chung

* Mô tả: Người dùng có thể xem các thông tin về các đề tài đã được đăng ký trên hệ thống. Người dùng có thể tìm kiếm theo bất cứ trường nào hay giá trị nào mà mình muốn. Có thể xuất file excel danh sách các đề tài để tiện xử lý hoặc nhóm các trường, sắp xếp để tiện theo dõi.
* Tác nhân: User/ Admin.
* Tiền điều kiện: Người dùng cần đăng nhập.
* Hậu điều kiện: Không có.
* Luồng sự kiện:
  + Sau khi đăng nhập, chọn mục “Đề tài”, danh sách đề tài sẽ hiển thị ra màn hình
  + Chọn tên đề tài để sắp xếp.
  + Chọn các ô filter ở đầu cột để lọc
  + Nhập vào ô search để tìm kiếm
  + Kéo cột muốn nhóm lên header để nhóm các trường cùng giá trị.
  + Chọn export to excel để xuất ra file excel.
* Luồng thay thế: Không có.
* Yêu cầu ngoại lệ: Không.

3.2.4 Xem danh sách đề tài của bản thân

* Mô tả: Người dùng có thể xem các thông tin về các đề tài mà mình đã đăng ký trên hệ thống. Người dùng có thể tìm kiếm theo bất cứ trường nào hay giá trị nào mà mình muốn. Có thể xuất file excel danh sách các đề tài để tiện xử lý hoặc nhóm các trường, sắp xếp để tiện theo dõi.
* Tác nhân: User/ Admin.
* Tiền điều kiện: Người dùng cần đăng nhập.
* Hậu điều kiện: Không có.
* Luồng sự kiện:
  + Sau khi đăng nhập, chọn mục “Đề tài”, danh sách đề tài sẽ hiển thị ra màn hình
  + Chọn tên đề tài để sắp xếp.
  + Chọn các ô filter ở đầu cột để lọc
  + Nhập vào ô search để tìm kiếm
  + Kéo cột muốn nhóm lên header để nhóm các trường cùng giá trị.
  + Chọn export to excel để xuất ra file excel.
* Luồng thay thế: Không có.

3.2.5 Xem và tìm kiếm danh sách cán bộ

* Mô tả: Người dùng có thể xem các thông tin về các cán bộ nghiên cứu trong hệ thống . Có thể xuất file excel danh sách các đề tài để tiện xử lý hoặc nhóm các trường, sắp xếp để tiện theo dõi.
* Tác nhân: User/ Admin.
* Tiền điều kiện: Người dùng cần đăng nhập.
* Hậu điều kiện: Không có.
* Luồng sự kiện:
  + Sau khi đăng nhập, chọn mục “Tác giả”, danh sách cán bộ nghiên cứu sẽ hiển thị ra màn hình
  + Chọn tên đề tài để sắp xếp.
  + Chọn các ô filter ở đầu cột để lọc
  + Nhập vào ô search để tìm kiếm
  + Kéo cột muốn nhóm lên header để nhóm các trường cùng giá trị.
  + Chọn export to excel để xuất ra file excel.
* Luồng thay thế: Không có.
* Yêu cầu ngoại lệ: Không.

3.2.6 Đăng ký đề tài

* Mô tả: Người dùng đăng ký đề tài mới lên hệ thống
* Tác nhân: User.
* Tiền điều kiện: Đã đăng nhập
* Hậu điều kiện: Dữ liệu được lưu lên database và hiển thị ra màn hình.
* Luồng sự kiện:
  + Người dùng chọn mục đề tài, chọn thêm mới.
  + Điền đầy đủ thông tin của đề tài, sau đó bấm lưu
  + Đề tài được lưu vào database và hiển thị ra màn hình.
* Luồng ngoại lệ: Nhập thiếu các trường bắt buộc thì phải nhập lại mới cho lưu.
* Luồng thay thế: Không có.

3.2.7 Chỉnh sửa thông tin đề tài

* Mô tả: Chỉnh sửa nội dung của đề tài.
* Tác nhân: Admin.
* Tiền điều kiện: Đăng nhập với quyền Admin, có yêu cầu/ quyết định chỉnh sửa đề tài.
* Hậu điều kiện: Không có.
* Luồng sự kiện:
  + Khi đã có quyết định được phép chỉnh sửa thông tin đề tài, Admin chọn đề tài cần sửa, tiến hành sửa thông tin.
  + Các thông tin có thể sửa là: Tên đề tài, mô tả, kinh phí, người nghiên cứu tương ứng với đề tài,…
  + Admin có thể chỉnh sửa trạng thái của đề tài tương ứng với hoạt động nghiên cứu của đề tài ấy (VD: Trạng thái đang đợi xét chọn -> Đang nghiên cứu, Đang nghiên cứu -> Đã nghiệm thu,…)
* Luồng thay thế: Không có.
* Luồng ngoại lệ: Khi sửa thông tin đề tài mà bị trùng mã đề tài thì không cho phép lưu, hiển thị thông báo và yêu cầu sửa lại.

3.2.8 Xóa thông tin đề tài

* Mô tả: Người dùng có thể xóa thông tin về một đề tài nếu có quyết định cho phép xóa đề tài
* Tác nhân : Admin.
* Tiền điều kiện: Đăng nhập với quyền Admin, có quyết định, yêu cầu xóa một đề tài
* Hậu điều kiện: Không thể tìm thấy đề tài trong cơ sở dữ liệu hay tìm kiếm trong hệ thống được nữa.
* Luồng sự kiện:
  + Khi đã có quyết định được phép xóa bỏ đề tài, Admin chọn đề tài cần xóa, bấm xóa.
  + Hiển thị form xác nhận xóa, nếu bấm xóa thì sẽ xóa đề tài khỏi database, nếu bấm hủy thì không thực hiện hành động xóa mà quay về màn hình chính.
* Luồng thay thế: Không có.
* Luồng ngoại lệ: Không có.

3.2.8 Chỉnh sửa thông tin của cán bộ nghiên cứu

* Mô tả: Người dùng có thể chỉnh sửa thông tin của một cán bộ nghiên cứu có trông hệ thống
* Tác nhân: Admin
* Tiền điều kiện: Đăng nhập với quyền Admin, có quyết định, yêu cầu chỉnh sửa đề tài.
* Hậu điều kiện: Thông tin đề tài sau khi được chỉnh sửa phải được lưu vào cơ sở dữ liệu và hiển thị đúng lên màn hình.
* Luồng sự kiện:
  + Admin chọn đề tài cần sửa, chọn sửa
  + Chỉnh sửa các thông tin cần thiết, bấm lưu.
* Luồng thay thế: Không có.
* Luồng ngoại lệ: Nếu cập nhật thông tin thất bại thì hiển thị thông báo, thông tin giữ nguyên.

3.2.9 Xóa thông tin cán bộ nghiên cứu

* Mô tả: Người dùng có thể xóa thông tin về một cán bộ nghiên cứu nếu có quyết định cho phép xóa cán bộ.
* Tác nhân : Admin.
* Tiền điều kiện: Đăng nhập với quyền Admin, có quyết định, yêu cầu xóa một cán bộ.
* Hậu điều kiện: Không thể tìm thấy cán bộ trong cơ sở dữ liệu hay tìm kiếm trong hệ thống được nữa. Tác giả của đề tài người này đã từng nghiên cứu sẽ bị bỏ trống
* Luồng sự kiện:
  + Khi đã có quyết định được phép xóa bỏ thông tin cán bộ, Admin chọn cán bộ cần xóa, bấm xóa.
  + Hiển thị form xác nhận xóa, nếu bấm xóa thì sẽ xóa cán bộ nghiên cứu này khỏi database, nếu bấm hủy thì không thực hiện hành động xóa mà quay về màn hình chính.
* Luồng thay thế: Không có.
* Luồng ngoại lệ: Không có.

3.2.10 Thống kê

* Mô tả: Admin chọn chức năng thống kê, những biểu đồ thống kê sẽ hiện ra màn hình chính.
* Tác nhân: Admin.
* Tiền điều kiện: Đăng nhập với quyền Admin.
* Hậu điều kiện: Không.
* Luồng sự kiện: Người dùng đăng nhập với quyền Admin, sau đó chọn mục “thống kê”. Những bảng biểu được xử lý với dữ liệu động sẽ hiện ra màn hình chính.
* Luồng thay thế: Không có.
* Luồng ngoại lệ: Không có.
* Yêu cầu đặc biệt: Không có.

## 4. Thiết kế cơ sở dữ liệu

Bảng aspnetusers

Bảng này dùng để lưu thông tin tài khoản của người dùng. (Bảng này được tạo bởi package Identity Server 4 nên có những trường dùng để mở rộng thêm ứng dụng mà không sử dụng trong ứng dụng demo )

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Name | Key | Data Type | Not null | Default | Description |
| 1 | Id | Pri | Varchar(255) | V |  | Id tài khoản |
| 2 | NormalizedUserName |  | Varchar(255) |  | Null | Tên đăng nhập hiển thị |
| 3 | Email |  | Varchar(255) |  | Null | Email đăng ký |
| 4 | EmailConfirmed |  | TINYINT(1) | V |  | Xác nhận mật khẩu |
| 5 | PasswordHash |  | LongText |  | Null | Mã hóa mật khẩu |
| 6 | SecurityStamp |  | LongText |  | Null |  |
| 7 | ConcurrencyStamp |  | LongText |  | Null |  |
| 8 | PhoneNumber |  | LongText |  | Null | Số điện thoại |
| 9 | PhoneNumberConfirmed |  | TINYINT(1) | V |  | Xác nhận số điện thoại |
| 10 | TwoFactorEnabled |  | TINYINT(1) | V |  | Xác thực hai lớp |
| 11 | LockoutEnd |  | DateTime |  | Null | Thời hạn khóa tài khoản |
| 12 | LockoutEnabled |  | TINYINT(1) |  | Null | Cho phép khóa tài khoản |
| 13 | AccessFailedCount |  | INT(11) |  | Null | Đếm số lần đăng nhập sai |

2. Bảng aspnetroles

Bảng này để lưu trữ các quyền trong hệ thống

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Name | Key | Data Type | Not null | Default | Description |
| 1 | Id | Pri | Varchar(255) | V |  | Id của role |
| 2 | NormalizedUserName |  | Varchar(256) |  | Null | Tên của role hiển thị |
| 3 | Name |  | Varchar(256) |  | Null | Tên của role lưu trữ |
| 4 | ConcurrencyStamp |  | Longtext |  |  |  |

3. Bảng aspnetuserroles

Bảng này để lưu trữ user với role tương ứng

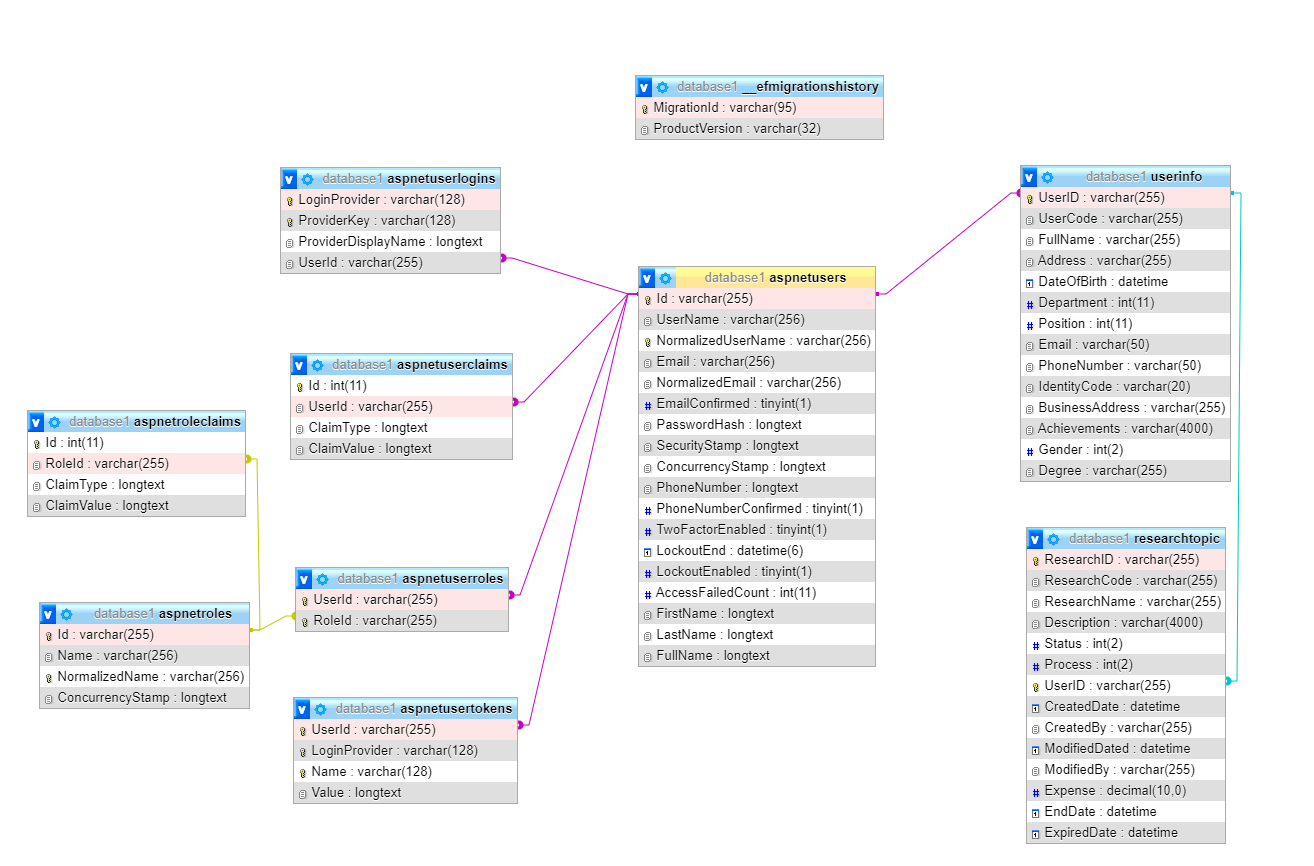
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Name | Key | Data Type | Not null | Default | Description |
| 1 | UserID | Pri | Varchar(255) | V |  | Id của user |
| 2 | RoleID |  | Varchar(255) | V |  | ID của role |

4. Bảng UserInfo

Bảng này để lưu thông tin những cán bộ tham gia nghiên cứu

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STT | Name | Key | Data Type | Not null | Default | Description |
| 1 | UserID | Pri | Varchar(255) | V |  | Id tài khoản |
| 2 | UserCode |  | Varchar(255) | V | Null | Mã cán bộ |
| 3 | FullName |  | Varchar(255) | V | Null | Tên đầy đủ |
| 4 | Address |  | TINYINT(1) |  |  | Địa chỉ |
| 5 | DateOfBirth |  | DateTime |  | Null | Ngày sinh |
| 6 | Department |  | INT(11) |  | Null | Phòng ban |
| 7 | Position |  | INT(11) |  | Null | Vị trí |
| 8 | Email |  | Varchar(50) |  | Null | Email |
| 9 | PhoneNumber |  | Varchar(50) | V |  | Số điện thoại |
| 10 | IdentityCode |  | Varchar(20) | V |  | Số CMND/CCCD |
| 11 | BusinessAddress |  | Varchar(255) |  | Null | Đại chỉ công tác |
| 12 | Achievements |  | Varchar(4000) |  | Null | Thành tích |
| 13 | Gender |  | INT(2) |  | Null | Giới tính |
| 14 | Degree |  | Varchar(255) |  |  | Học vị |

5. Mô hình cơ sở dữ liệu



# Chương III. Giới thiệu các công nghệ sử dụng

## 1. Các công nghệ sử dụng

### 1.1 ASP.NET Core

ASP.NET Core là một web framework mã nguồn mở ( sử dụng giấy phép MIT và Apache 2 ) và được tối ưu hóa cho cloud để phát triển các ứng dụng web chạy trên nhiều nền tảng như Windows, Linux và Mac. Hiện tại, nó bao gồm MVC framework được kết hợp các tính năng của MVC và Web API thành một web framework duy nhất.. Nó tổng hợp các thư viện chuẩn và được phân chia module tốt hơn và gọn hơn so với ASP.NET thường. ASP.NET có thể chạy mã nguồn của bạn với cùng một hành vi trên nhiều kiến ​​trúc hệ thống, bao gồm x64, x86 và ARM. .NET Core tương thích với .NET Framework, Xamarin và Mono, thông qua .NET Standard. Ngoài ra .NET được hỗ trợ bởi Microsoft, cộng đồng người dùng lớn.

Về thành phần .NET Core bao gồm:

* .NET Core Runtime gồm các kiểu dữ liệu nguyên thủy, các kiểu thành phần ứng dụng và tiện ích cơ bản.
* .NET Core SDK và trình biên dịch ngôn ngữ.
* Dotnet command, được sử dụng để khởi chạy các ứng dụng .NET Core và các lệnh CLI. Nó chọn thời gian chạy và lưu trữ thời gian chạy, cung cấp chính sách tải lắp ráp và khởi chạy các ứng dụng và công cụ.

Về ưu điểm:

* Có thể xây dựng và chạy ứng dụng ASP.NET đa nền tảng trên Windows, Mac và Linux.
* Có thể host trên IIS hoặc self-host trong process của riêng bạn.
* Dependency injection được xây dựng sẵn.
* Cơ chết HTTP Request pipeline mới nên HTTP request được tối ưu nhẹ hơn.
* Hỗ trợ quản lý phiên bản.
* Dùng chung toàn bộ Nuget Package.
* Dễ dàng tích hợp với các framework frontend như Angular, Knockout...

### 1.2 Identity Authentication

ASP.NET Core Identity là một hệ thống chứng thực cấp quyền của ASP.NET Core, nó cung cấp cho bạn các tính năng đầy đủ và đa dạng về authentication như: tạo tài khoản, login với user name và password, cập nhật profile. Hoặc cũng có thể sử dụng những provider bên ngoài giống như: Facebook, Google, Twitter, Microsoft account và nhiều cái khác nữa. Bạn có thể cấu hình ASP.NET Core để sử dụng với SQL Server (hoặc có thể dùng các loại CSDL quan hệ khác) nhằm lưu trữ user name, password và dữ liệu profile.

ASP.NET Core Identity là cơ chế dùng chung cho tất cả các ứng dụng Web bao gồm ASP.NET MVC, Web API, WebForm và SignalR. Ta có thể dễ dàng thêm và tùy biến các trường dữ liệu khác vào user, dễ dàng test, dễ dàng quản lý quyền (role). Nó độc lập với web vì sử dụng cơ chế OWIN. Ngoài ra, ASP.NET Core Identity dễ dàng cài đặt từ kho Nuget Packet.

Các bước cài đặt:

* Cài đặt 3 thưu viện
  + Microsoft.AspNet.Identity.EntityFramework
  + Microsoft.AspNet.Identity.Core
  + Microsoft.AspNet.Identity.OWIN
* Tạo mới class User kế thừa từ IdentityUser.
* Tạo mới Role kế thừa từ IdentityRole.
* Kế thừa lớp DbContext từ IdentityDbContext.
* Cấu hình authentication từ file Startup.cs.
* Thực hiện Migration vào database.
* Tạo class quản lý authen.

### 1.3 Dapper/ Entity Framework

Dapper là một Object Relational Mapper, nó là một thư viện xây dựng trên nền ADO.NET data reader. Nói một cách khác Dapper là một thư viện mở rộng của ADO.NET tuy nhiên lại được bổ sung thêm các tính năng của ORM.

Khi sử dụng ADO.NET để để convert datareader thành một object model chúng ta thường sử dụng các delegate để thao tác. Dapper đã xây dựng các method library để tự động thực hiện các việc này cho chúng ta thông qua các lớp metadata.

Cách sử dụng:

* IDbConnection object với tham số là 1 connection String.
* Viết SQL query thực hiện các thao tác Get, Post, Put, Delete.
* Truyền SQL query như là một tham số vào method của dapper, (Execute, Query, ExecuteScala, …).

Entity Framework là một khung ORM mã nguồn mở cho các ứng dụng .NET được Microsoft hỗ trợ. Nó cho phép các nhà phát triển làm việc với dữ liệu bằng cách sử dụng các đối tượng của các lớp cụ thể của miền mà không cần tập trung vào các bảng và cột cơ sở dữ liệu cơ bản nơi dữ liệu này được lưu trữ. Với Entity Framework, các nhà phát triển có thể làm việc ở mức độ trừu tượng cao hơn khi họ xử lý dữ liệu và có thể tạo và duy trì các ứng dụng hướng dữ liệu với ít mã hơn so với các ứng dụng truyền thống. Ở đây, khi chúng ta sử dụng ASP.NET Core Identity thì bắt buộc phải sử dụng Entity Framework cho phần identity này.

### 1.4 MariaDb

MariaDB là hệ quản trị cơ sở dữ liệu miễn phí được phát triển từ hệ quản trị cơ sở dữ liệu mã nguồn mở MySQL. MariaDB được phát triển nhằm thay thế công nghệ cơ sở dữ liệu MySQL, vì thế nó tương thích và cho một hiệu suất cao hơn so với MySQL. Với RDBMS ( viết tắt của Relational Database Management System), MySQL được tích hợp apache, PHP. MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu. Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL. MySQL được phát hành từ thập niên 90s.

Ưu điểm của MariaDb:

* Hoàn toàn miễn phí: người dùng không cần phải bỏ tiền mua bản quyền và vẫn có thể sử dụng đầy đủ những tính năng của phần mềm này
* Khắc phục những hạn chế của MySQL mà vẫn kế thừa đặc điểm của MySQL. MariaDb nhanh hơn 3-5% nhưng có nhiều chức năng hơn. Người dùng có thể chuyển từ MySQL sang MariaDb mà không ảnh hưởng đến hệ thống.
* Bổ sung thêm nhiều Engine hơn: Ngoài các storage engines cơ bản như MyISAM, BLACKHOLE, CSV, MEMORY, ARCHIVE, MERGE; MariaDB còn phát triển thêm các storage engines khác bao gồm: Aria, XtraDB, FederatedX, OQGRAPH, SphinxSE, IBM DB2I, Spider, PBXT,…
* Kết hợp cả SQL và NoSQL
* Hỗ trợ tiếng Việt.

### 1.5 Vue.js

Vue.js là một framework linh động dùng để xây dựng giao diện người dùng (UI). Vue.js giúp xây dựng giao diện từng bước từng bước một cách mạch lạc. Nó cung cấp những thư viện lõi giúp xây dựng một trang Single Page Application một cách dễ dàng.

Trong những framework frontend hiện tại thì vue.js đang có số sao đánh giá cao nhất, mặc dù lượng người sử dụng nhỏ hơn React.js, bởi những ưu điểm sau:

* Dễ dàng build một ứng dụng vuejs mà không cần trải qua quá nhiều bước.
* Tạo cấu trúc project nhanh chóng nên ứng dụng rất nhẹ.
* Tài liệu đa dạng, dễ dàng tra cứu.
* Vue sở hữu một hệ sinh thái vững chắc nên có thể cung cấp một số add-ons rất hữu ích cho việc xây dựng một ứng dụng fontend điển hình nhất. Nó có thể bao gồm: vue-router, vuex, vue-test-utils, vue-dev-tools, vue-cli,…
* Loại bỏ được các tính năng không cần thiết ra khỏi thư viện trong lõi Vue.js và đảm bảo cho framework nhỏ gọn và mềm dẻo hơn.

### 1.6 Công cụ Visual Studio 2019

Visual Studio là công cụ lập trình được Microsoft phát hành hỗ trợ cho việc lập trình web. Công cụ này thực sự đem lại nhiều lợi ích cho người sử dụng:

* Đa nền tảng: Visual Studio sử dụng được trên cả Windows, Linux và Mac Systems.
* Visual Studio cũng cho phép sử dụng nhiều ngôn ngữ lập trình khác nhau từ C#, F#, C/C++, HTML, CSS, Visual Basic, JavaScript,… Bởi vậy, Visual Studio có thể dễ dàng phát hiện và thông báo cho bạn khi các chương trình có lỗi.
* Hỗ trợ tối ưu cho việc thiết kế website.
* Kho tiện ích (Nuget packet) khổng lồ.
* Lưu trữ phân cấp: Phần lớn các tệp dữ liệu đoạn mã của Visual Studio đều được đặt trong các thư mục tương tự nhau. Đồng thời, Visual Studio cũng cung cấp một số thư một cho các tệp đặc biệt để bạn lưu trữ an toàn, dễ tìm, dễ sử dụng hơn.
* Hỗ trợ Git, TFS để quản lý code theo nhóm một cách dễ dàng, an toàn.
* Ngoài ra còn có những tiện ích về mặt sử dụng như màn hình đa nhiệm, hỗ trợ viết code, hỗ trợ thiết bị đầu cuối,…

### 1.7 Công cụ DbForce

DbForce là một công cụ quản lý cho một cơ sở dữ liệu quan hệ. Nó có những ưu điểm sau:

* Xây dựng Database một các dễ dàng. Hỗ trợ thiết kế và vẽ các bảng dữ liệu sử dụng Diagram. Có thể vẽ được các bảng và nhìn tổng quan được các relation giữa các bảng.
* Hỗ trợ Debug SQL: rất cần thiết khi muốn xem rõ lỗi cụ thể trong đoạn mã truy vấn dài hàng trăm dòng.
* Hỗ trợ Compare Data và cấu trúc Database. Rất đắc lực khi bạn làm việc trên nhiều phiên bản của Datatabase. Giúp bạn compare dữ liệu hoặc cấu trúc 2 Database và thực hiện đồng bộ chúng lại.
* Generate Data – sinh dữ liệu test. Việc này cực kỳ cần thiết khi bạn có nhu cầu muốn sinh “một đống” dữ liệu giả và “đẹp như dữ liệu thật”.
* Format SQL – định dạng câu truy vấn đẹp và dễ nhìn. Bạn không phải mất công chỉnh từng câu truy vấn trong chuỗi truy vấn dài ngoằng của mình.
* Auto-suggestions (SQL query) – tự động gợi ý và sinh chuỗi câu truy vấn trong quá trình viết câu truy vấn. Bạn cứ tưởng tượng khi chỉ cần gõ “sel” + phím [TAB] là có ngày chuỗi SELEC \* FROM, chọn bảng rồi focus chuột vào ký tự “\*” chúng ta có chuỗi câu hoàn chỉnh SELECT toàn bộ các cột dữ liệu có trong bảng rồi tùy chỉnh lại theo ý muốn mà không mất công gõ từng cột dữ liệu có trong bảng. Đây là một tính năng thú vị tiết kiệm đến 90% thời gian viết chuỗi truy vấn SELECT.
* Import/ Export dữ liệu, Automation test, và hàng đống các tính năng khác…

## Kiến trúc hệ thống

Mô hình sử dụng cho hệ thống này là mô hình MVC. MVC là từ viết tắt bởi 3 từ Model – View – Controller. Đây là mô hình thiết kế sử dụng trong kỹ thuật phần mềm. Mô hình source code gồm 3 phần, tương ứng mỗi từ. Mỗi từ tương ứng với một hoạt động tách biệt trong một mô hình.

Model (M): Là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng. Bộ phận này là một cầu nối giữa 2 thành phần bên dưới là View và Controller. Model thể hiện dưới hình thức là một cơ sở dữ liệu hoặc có khi chỉ đơn giản là một file XML bình thường. Model thể hiện rõ các thao tác với cơ sở dữ liệu như cho phép xem, truy xuất, xử lý dữ liệu,…

View (V): Đây là phần giao diện dành cho người sử dụng. Nơi mà người dùng có thể lấy được thông tin dữ liệu của MVC thông qua các thao tác truy vấn như tìm kiếm hoặc sử dụng thông qua các website. Thông thường, các ứng dụng web sử dụng MVC View như một phần của hệ thống, nơi các thành phần HTML được tạo ra. Bên cạnh đó, View cũng có chức năng ghi nhận hoạt động của người dùng để tương tác với Controller. Tuy nhiên, View không có mối quan hệ trực tiếp với Controller, cũng không được lấy dữ liệu từ Controller mà chỉ hiển thị yêu cầu chuyển cho Controller mà thôi.

Controller : Bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua view. Từ đó, Controller đưa ra dữ liệu phù hợp với người dùng. Bên cạnh đó, Controller còn có chức năng kết nối với model.

Ưu điểm: Mô hình đơn giản, dễ dàng kiểm tra lỗi, nhẹ và tiết kiệm băng thông do không sử dụng viewstate.