**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------🙡🕮🙣----------**

****

**BÀI TẬP LỚN**

**MÔN PHÁT TRIỂN PHẦN MỀM HƯỚNG CẤU PHẦN**

**ĐỀ TÀI: XÂY DỰNG HỆ THỐNG QUẢN LÝ GIA PHẢ THEO HƯỚNG CẤU PHẦN TRÊN NỀN TẢNG JSP/SERVLET**

**Giáo viên : ThS. Nguyễn Thái Cường**

**Nhóm: 17**

**Lớp: ĐH KTPM 3 – K11**

**Hà Nội - 2019**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------🙡🕮🙣----------**

****

**BÀI TẬP LỚN**

**MÔN PHÁT TRIỂN PHẦN MỀM HƯỚNG CẤU PHẦN**

**ĐỀ TÀI: XÂY DỰNG HỆ THỐNG QUẢN LÝ GIA PHẢ THEO HƯỚNG CẤU PHẦN TRÊN NỀN TẢNG JSP/SERVLET**

**Giáo viên: ThS. Nguyễn Thái Cường**

**Nhóm: 17**

**Sinh viên thực hiện: Phạm Văn Hiểu**

**Phạm Hải Dương**

**Phạm Quốc Cường**

**Lớp: ĐH KTPM 3 – K11**

**Hà Nội – 2019**

**MỤC LỤC**

[PHẦN I. MỞ ĐẦU 1](#_Toc25401057)

[**1.** **Tên đề tài** 1](#_Toc25401058)

[**2.** **Lý do chọn đề tài** 1](#_Toc25401059)

[**3.** **Mục đích** 1](#_Toc25401060)

[**4.** **Mục tiêu** 2](#_Toc25401061)

[**5.** **Bố cục đề tài** *(nội dung chính: 3 chương)* 2](#_Toc25401062)

[**6.** **Phương pháp** 2](#_Toc25401063)

[PHẦN II. NỘI DUNG 3](#_Toc25401064)

[**CHƯƠNG 1. PHÂN TÍCH THIẾT KẾ HỆ THỐNG** 3](#_Toc25401065)

[**1.1.** **Biểu đồ use case** 3](#_Toc25401066)

[1.1.1. Biểu đồ use case chính 3](#_Toc25401067)

[1.1.2. Biểu đồ use case mức 2 3](#_Toc25401068)

[1.1.2.1. Use case “Đăng nhập” 3](#_Toc25401069)

[1.1.2.2. Use case “Quản lý gia phả” 4](#_Toc25401070)

[1.1.2.3. Use case “Xem thông tin gia phả” 4](#_Toc25401071)

[1.1.2.4. Use case “Quản lý nội dung” 5](#_Toc25401072)

[**1.2.** **Mô tả use case** 5](#_Toc25401073)

[1.2.1. Use case: LogIn (Đăng nhập) 5](#_Toc25401074)

[1.2.2. Use case: Manage Genealogy (Quản lý gia phả) 7](#_Toc25401075)

[1.2.3. Use case: See Genealogy Information (Xem thông tin gia phả) 8](#_Toc25401076)

[1.2.4. Use case: Manage Content (Quản lý nội dung) 9](#_Toc25401077)

[**1.3.** **Phân tích use case** 10](#_Toc25401078)

[1.3.1. Phân tích use case “Đăng nhập” 10](#_Toc25401079)

[1.3.2. Phân tích use case “Quản lý gia phả” 12](#_Toc25401080)

[1.3.3. Phân tích use case “Xem thông tin gia phả” 17](#_Toc25401081)

[1.3.4. Phân tích use case “Quản lý nội dung” 19](#_Toc25401082)

[**CHƯƠNG 2. THIẾT KẾ CƠ SỞ DỮ LIỆU** 21](#_Toc25401083)

[**2.1.** **Thiết kế cơ sở dữ liệu - Ánh xạ lớp sang bảng** 21](#_Toc25401084)

[**2.2.** **Các bảng sau khi được ánh xạ** 21](#_Toc25401085)

[**CHƯƠNG 3. THIẾT KẾ GIAO DIỆN** 24](#_Toc25401086)

[**3.1.** **Giao diện** 24](#_Toc25401087)

[**CHƯƠNG 4. THIẾT KẾ THƯ VIỆN, API** 28](#_Toc25401088)

[**4.1** **Cách tạo và sử dụng thư viện** 28](#_Toc25401089)

[4.1.1 Tạo file jar 28](#_Toc25401090)

[4.1.2 Import vào project 30](#_Toc25401091)

[**4.2** **Thiết kế thư viện** 30](#_Toc25401092)

[**4.2.1** Thư viện DAO 30](#_Toc25401093)

[**4.2.2** Thư viện Filter 31](#_Toc25401094)

[**4.2.3** Thư viện Mapper 31](#_Toc25401095)

[**4.2.4** Thư viện Utils 32](#_Toc25401096)

[**4.3** **Thiết kế API** 32](#_Toc25401097)

# PHẦN I. MỞ ĐẦU

## **Tên đề tài**

***Xây dựng Hệ thống quản lý gia phả theo hướng cấu phần trên nền tảng JSP/SERVLET.***

## **Lý do chọn đề tài**

Cùng với dòng chảy của thời gian loài người đã sinh sôi nảy nở và tồn tại trên trái đất này. Trải qua nhiều thế hệ con người đã không ngừng phát triển và tạo dựng nền móng văn minh cho nhân loại. Qua bao nhiêu thế kỉ với các giai đoạn xã hội khác nhau, các dòng họ đã có sự quản lý dòng họ của mình qua cuốn Gia Phả và được lưu truyền từ đời này sang đời khác, từ thế hệ này sang thế hệ khác. Trước đây việc quản lý dữ liệu, những thông tin về mỗi cá nhân trong dòng họ đều được thực hiện bằng tay. Điều này gây ra nhiều khó khăn và rất tốn thời gian khi ta muốn tra cứu thông tin về ai đó trong dòng họ, và cũng thật rắc rối rất khi dòng họ này lại có thêm thành viên mới sinh ra, hay dòng họ này lại tìm một người nào đó bị thất lạc.

Cuốn Gia Phả lúc này lại trở lên dài rất khó quản lý, rất khó kiểm soát. Điều gây phiền hà nhất ở đây là khi muốn thay đổi thông tin của cá nhân nào đó trong họ thì cuốn Gia Phả đó lại phải viết lại từ đầu, hay là tạo ra cuốn Gia Phả mới. Đây quả là một sự bất tiện không mong muốn.

Ngày nay với sự phát triển không ngừng của Công nghệ thông tin nhóm em đã tìm tòi, học hỏi, thu nhập các thông tin, tài liệu và phát triển **“*Xây dựng Hệ thống quản lý gia phả theo hướng cấu phần trên nền tảng JSP/SERVLET*”**. Hệ thống Quản lý gia phả này giúp tra cứu các thông tin về cuốn Gia Phả một các nhanh chóng hơn, tiện lợi, chính xác và tiết kiệm thời gian hơn.

## **Mục đích**

Đề tài **“Hệ thống Quản lý gia phả một dòng họ”** nhằm áp dụng công cụ, kiến thức và kỹ thuật, xây dựng hệ thống đi vào hoạt động giúp cho mọi người có thể xem thông tin về các dòng họ cũng như có một cuốn gia phả của riêng dòng họ mình.

## **Mục tiêu**

* Tìm kiếm được dự án thực tế để có điều kiện trải nghiệm thực tế.
* Xây dựng hệ thống giúp tra cứu thông tin gia phả một cách nhanh chóng.
* Giúp người dùng và người quản trị thao tác nghiệp vụ nhanh chóng dễ dàng.
* Giảm thời gian khi xây dựng một cuốn gia phả.
* Thay đổi thông tin gia phả một cách dễ dàng, nhanh chóng.

## **Bố cục đề tài** *(nội dung chính: 3 chương)*

* *Chương 1: Phân tích thiết kế hệ thống*
* Biểu dồ Use Case
* Mô tả Use Case
* Phân tích Use Case
* *Chương 2: Thiết kế cơ sở dữ liệu*
* *Chương 3:*

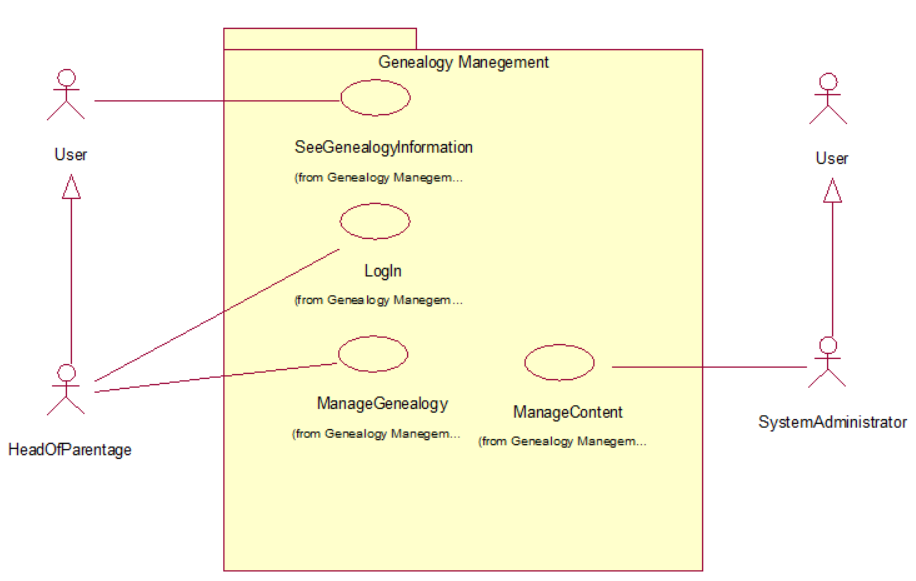
## **Phương pháp**

* Sử dụng kỹ thuật, công cụ hỗ trợ để phân tích UML và lập trình.
* Áp dụng kiến thức tìm hiểu được để xây dựng và phát triển hệ thống.

# PHẦN II. NỘI DUNG

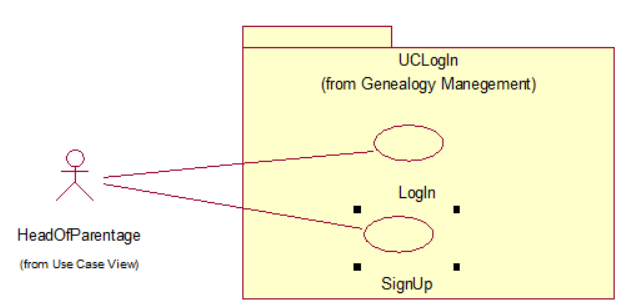
## **CHƯƠNG 1. PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

* 1. **Biểu đồ use case**
     1. **Biểu đồ use case chính**

****

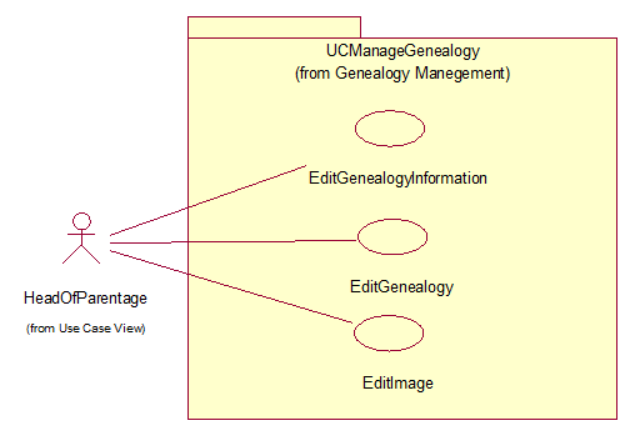
*Hình 1.1 Biểu đồ use case chính*

* + 1. **Biểu đồ use case mức 2**
       1. **Use case “Đăng nhập”**

****

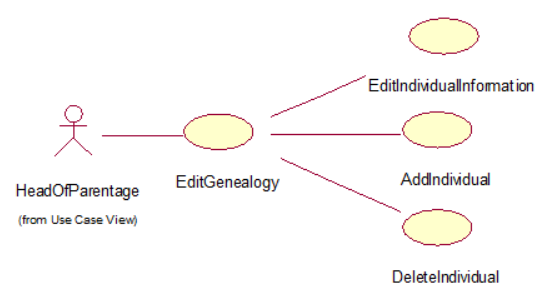
*Hình 1.2 Biểu đồ use case Đăng nhập*

* + - 1. **Use case “Quản lý gia phả”**

****

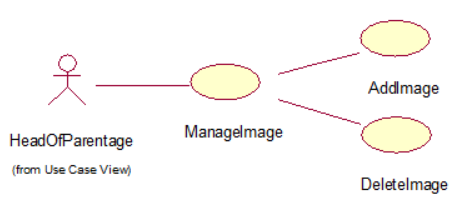
*Hình 1.3 Biểu đồ use case “Quản lý gia phả”*

* + - * 1. **Use case “Sửa gia phả”**



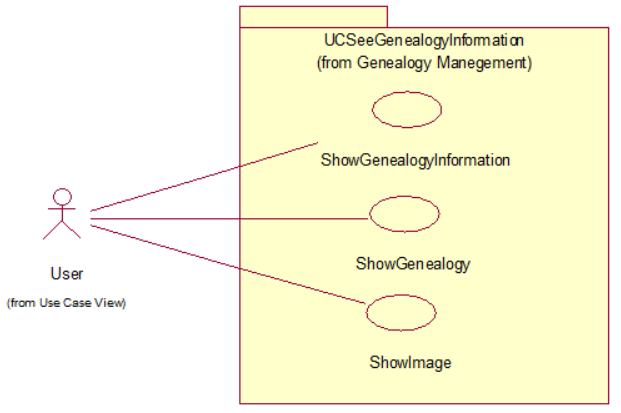
*Hình 1.4 Biểu đồ use case “Sửa gia phả”*

* + - * 1. **Use case “Sửa hình ảnh”**



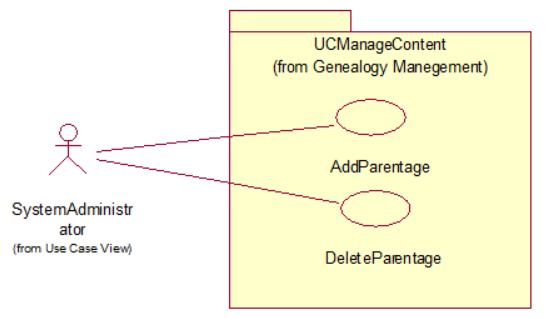
*Hình 1.5 Biểu đồ use case “Sửa hình ảnh”*

* + - 1. **Use case “Xem thông tin gia phả”**

****

*Hình 1.6 Biểu đồ use case “Xem thông tin gia phả”*

* + - 1. **Use case “Quản lý nội dung”**



*Hình 1.7 Biểu đồ use case “Quản lý nội dung”*

* 1. **Mô tả use case**
     1. **Use case: LogIn (Đăng nhập)**
        1. **Đăng nhập**
* **Mô tả vắn tắt:** Use case này cho phép người dùng đăng nhập hệ thống bằng cách nhập tên đăng nhập và mật khẩu để truy nhập các chức năng của hệ thống.
* **Luồng sự kiện:**
  + **Luồng cơ bản:**
    - 1. Use case này bắt đầu khi người dùng kích vào nút “Đăng nhập” trên Menu chinh. Hệ thống hiển thị một màn hình Đăng nhập.
      2. Người dùng nhập tên đăng nhập và mật khẩu và các trường tương ứng và ấn nút “Đăng nhập”. Hệ thống kiểm tra chi tiết người dùng trong bảng “ACCOUNT” va hiển thị Menu lựa chọn các chức năng tương ứng với quyền của người dùng. Use case kết thúc.
  + **Luồng rẽ nhánh:**
    1. Hủy bỏ: Tại bước 2 trong luồng cơ bản, nếu người dùng kích vào nút “Hủy bỏ” thì use case kết thúc.
    2. Sai tên hoặc mật khẩu: Tại bước 2 trong luồng cơ bản, nếu người dùng nhập sai tên đăng nhập hay mật khẩu thì hệ thống sẽ thồn báo lỗi. Người dùng nhập sai tên đăng nhập hay mật khẩu thì hệ thống sẽ thông báo lỗi. Người dùng có thể nhập lại và tiếp tục bước 2 hoặc chọn hủy bỏ để kết thúc use case.
    3. Không kết nối được với cơ sở dữ liệu: Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.
* **Các yêu cầu đặc biệt:**

Use case này được dùng để phân quyền của người dùng theo tên đăng nhập. Các quyền được được kiểm soát theo vai trò khách hàng, người quản trị.

* **Tiền điều kiện:**

Người dùng cần phải có một tài khoản trong hệ thống. Nếu use case thành công thì người dùng sẽ đăng nhập được vào hệ thống và có thể sử dụng các chức năng ứng với vai trò của mình, ngược lại trạng thai của hệ thống không thay đổi .

* **Hậu điều kiện:**

Nếu use case thành công thì người dùng sẽ đăng nhập vào hệ thống và có thể sử dụng các chức năng ứng với vai trò của mình, ngược lại trạng thái của hệ thống không thay đổi.

* **Điểm mở rộng:**

Không có.

* + - 1. **Đăng ký**
* **Mô tả vắn tắt:** Use case này cho phép người dùng tạo một tài khoản thành viên để có thể sử dụng đầy đủ các chức năng của khách hàng.
* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng kích chọn vào nút “Đăng ký” trong Menu đăng nhập. Hệ thống hiển thị màn hình tạo tài khoản.
2. Người dùng nhập thông tin cần thiết vào các trường tương ứng.
3. Nhấn vào nút “Đăng ký”.
4. Hệ thống sẽ lưu thông tin đăng ký của người dùng vào bảng “ACCOUNT” trong cơ sở dữ liệu. Và hiển thị màn hình kết quả sau khi đăng ký thành công.
5. Use case kết thúc.
   * **Luồng rẽ nhánh:**
     1. Tại bước 5 trong luồng cơ bản nếu người dùng nhập thông tin không chính xác thì hệ thống sẽ thông báo người dùng nhập không chính xác.
     2. Hệ thống sẽ hiển thị lại form đăng ký yêu cầu người dùng nhập lại thông tin.
     3. Nếu người dùng đồng ý thì quay lại bước 3 của luồng cơ bản, nếu không thì click chọn “Hủy” và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Không có.

* **Hậu điều kiện:**

Nếu use case thành công thì người dùng sẽ trở thành thành viên của hệ thống.

* **Điểm mở rộng:**

Không có

* + 1. **Use case: Manage Genealogy (Quản lý gia phả)**
* **Mô tả vắn tắt:** Use case này cho phép người dùng (Trưởng họ) quản lý thông tin về dòng họ của mình.
* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng click vào nút “Chỉnh sửa” trong gia phả mà họ đang quản lý. Hệ thống hiển thị màn hình quản lý gia phả.
2. Người dùng chọn công việc cần sửa, nhập thông tin cần thiết vào các trường tương ứng.
3. Click vào nút “Lưu thông tin”.
4. Hệ thống sẽ lưu thông tin của gia phả vào bảng “PARENTAGE” trong cơ sở dữ liệu. Và hiển thị thông báo sau khi sửa thành công.
5. Use case kết thúc.
   * **Luồng rẽ nhánh:**
     1. Không kết nối với cơ sở dữ liệu: tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không thể kết nối với cơ sở dữ liệu thì hệ thống sẽ báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Chỉ có người dùng là trưởng họ mới có thể thực hiện use case này.

* **Tiền điều kiện:**

Người dùng đã có tài khoản và đăng nhập vào hệ thống.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

* + 1. **Use case: See Genealogy Information (Xem thông tin gia phả)**
* **Mô tả vắn tắt:** Use case này cho phép người dùng xem thông tin gia phả của từng dòng họ.
* **Luồng sự kiện:**
  + **Luồng cơ bản:**
    - 1. Use case bắt đầu khi người dùng click vào một gia phả bất kỳ. Hệ thống sẽ lấy thông tin gia phả từ bảng “PARENTAGE” trong cơ sở dữ liệu và hiển thị thông tin chi tiết về dòng họ đó.
      2. Thông tin: Người dùng click “Thông tin chung” hệ thống sẽ hiển thị thông tin tổng quan của gia phả lên màn hình.
      3. Phả đồ: Người dùng click “Phả đồ” hệ thống sẽ hiển thị danh sách các người con trong dòng họ.
      4. Ảnh: Người dung click “Album ảnh” hệ thống sẽ hiển thị những hình ảnh về dòng họ.
      5. Khi người dùng click nút “Trang chủ” use case kết thúc.
  + **Luồng rẽ nhánh:**
    1. Không kết nối được tới cơ sở dữ liệu: Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu hệ thống sẽ hiển thị một thông bao lỗi và use case kết thúc.
* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Không có.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

* + 1. **Use case: Manage Content (Quản lý nội dung)**
* **Mô tả vắn tắt:** Use case này cho phép người quản trị hệ thống thêm gia phả được lấy từ các nguồn trên mạng hoặc xóa đi.
* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người quản trị đăng nhập vào hệ thống. Hệ thống sẽ hiển thị một danh sách các dòng họ được thu thập từ trên mạng.
2. Thêm: Người quản trị click nút ‘Lưu lại’ đối với gia phả muốn thêm. Hệ thống sẽ lưu thông tin của gia phả đó vào bảng “PARENTAGE” và “INDIVIDUAL”.
3. Xóa: Người quản trị click nút ‘Xóa’ với gia phả muốn xóa. Hệ thống sẽ xóa gia phả trong bảng “PARENTAGE” và “INDIVIDUAL”.
4. Use case kết thúc khi người quản trị thực hiện chức năng khác.
   * **Luồng rẽ nhánh:**
     1. Trùng tên: Tại bước 1 trong luồng cơ bản , nếu người quản trị nhập trùng tên với các tài khoản có trước thì hệ thống sẽ báo lỗi. Người quản trị có thể nhập lại và tiếp tục bước 2 hoặc chọn hủy bỏ để kết thúc use case.
     2. Không kết nối với cơ sở dữ liệu: tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không thể kết nối với cơ sở dữ liệu thì hệ thống sẽ báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Chỉ có người quản trị hệ thống mới có thể thực hiện use case này.

* **Tiền điều kiện:**

Người quản trị hệ thống phải đăng nhập vào hệ thống.

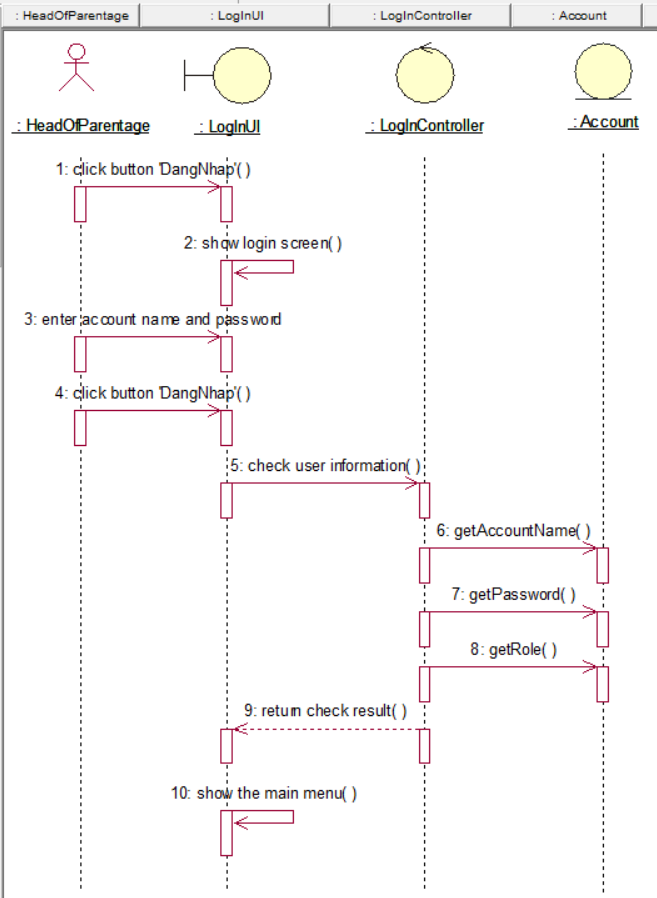
* **Hậu điều kiện:**

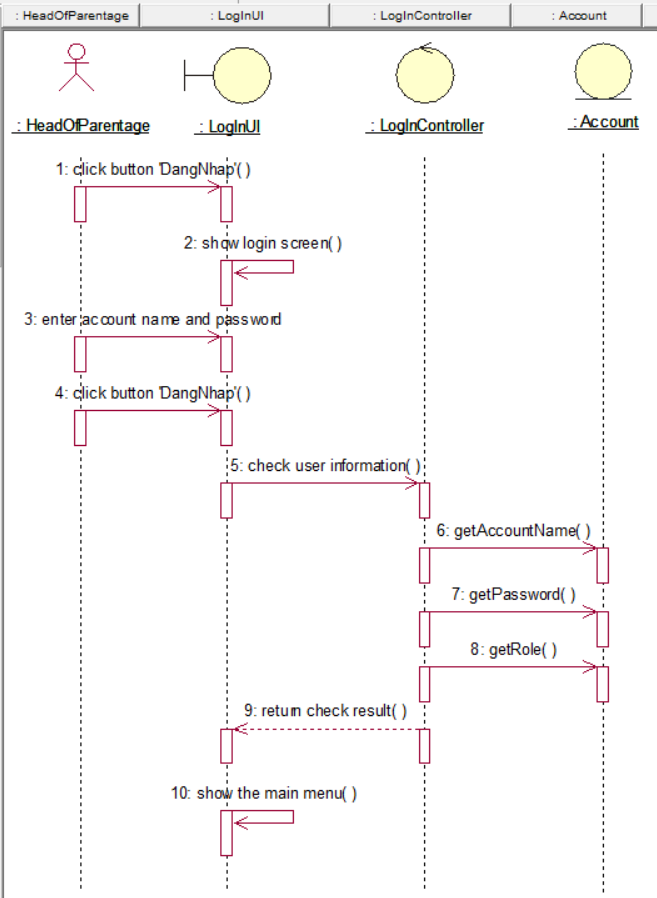
Không có.

* **Điểm mở rộng:**

Không có.

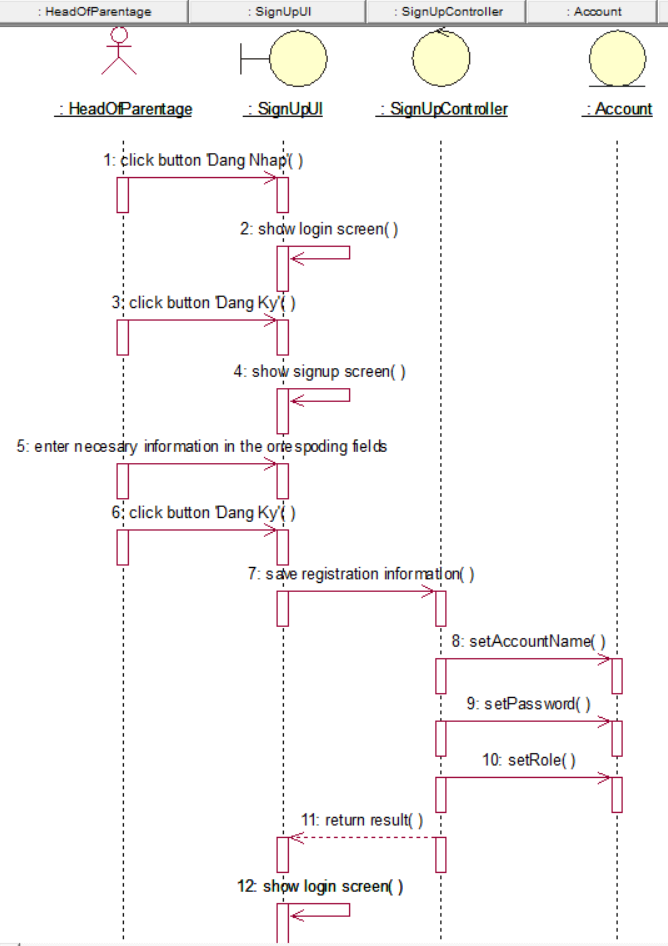
* 1. **Phân tích use case**
     1. **Phân tích use case “Đăng nhập”**
        1. **Biểu đồ trình tự chức năng “Đăng nhập”**

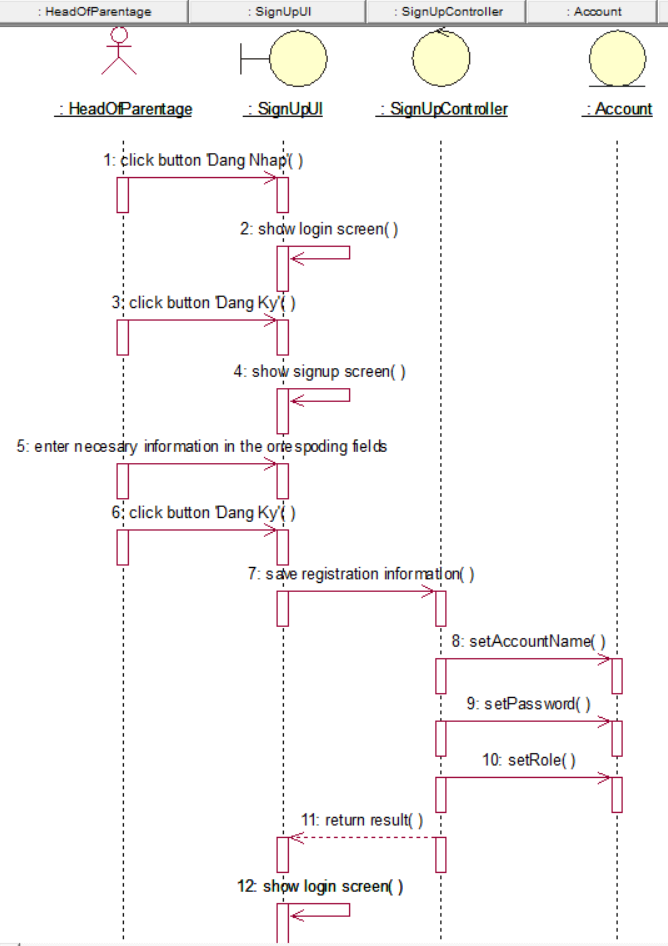
****

****

*Hình 1.8 Biểu đồ trình tự chức năng “Đăng nhập”*

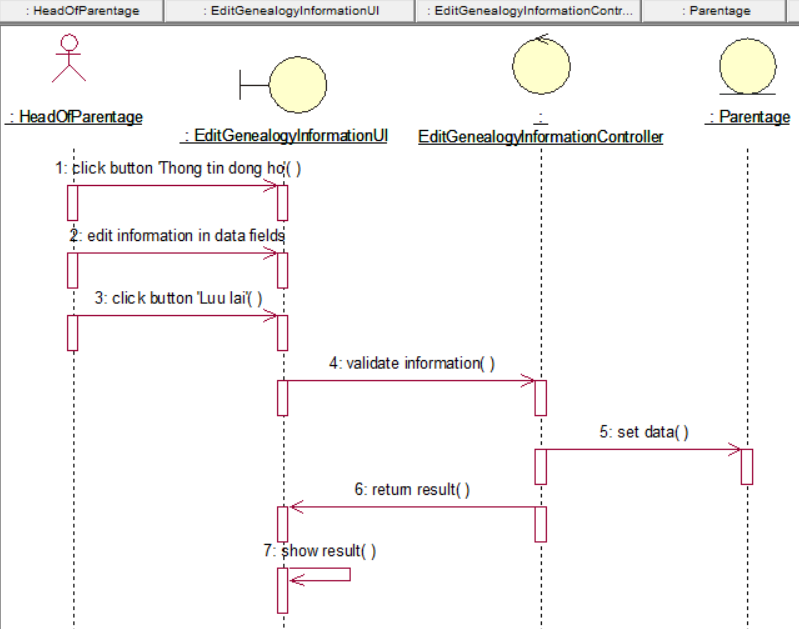
* + - 1. **Biểu đồ trình tự chức năng “Đăng ký”**

****

****

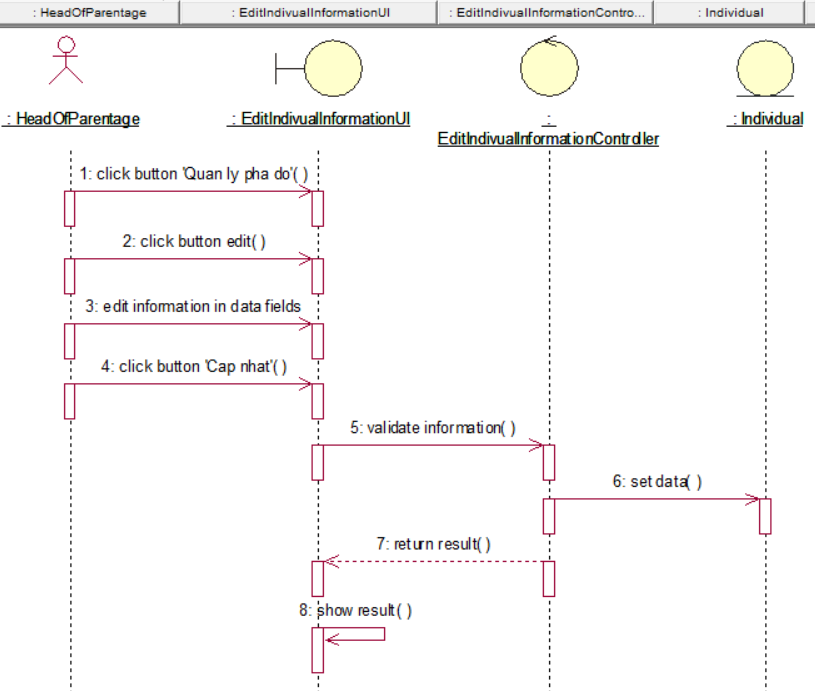
*Hình 1.9 Biểu đồ trình tự chức năng “Đăng ký”*

* + 1. **Phân tích use case “Quản lý gia phả”**
       1. **Biểu đồ trình tự chức năng “Sửa thông tin gia phả”**



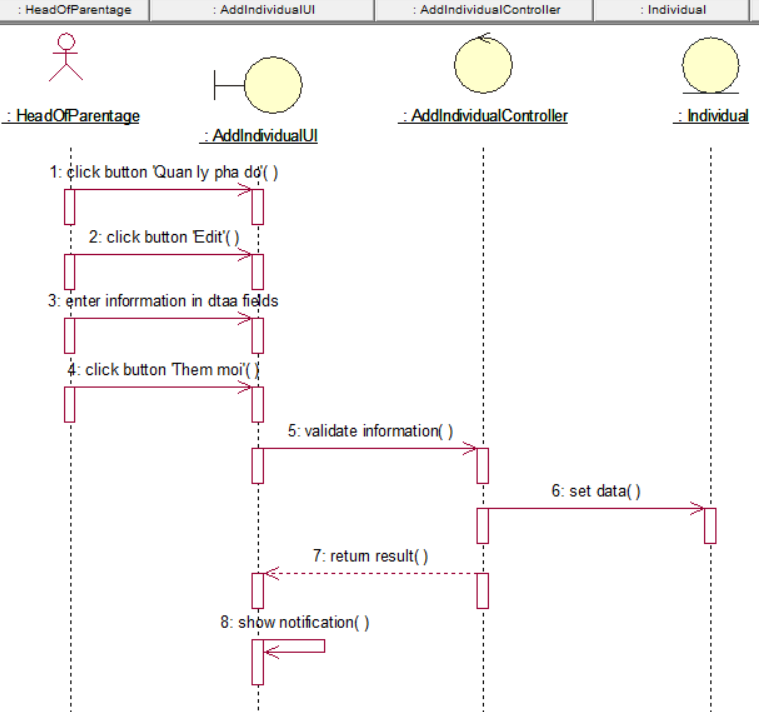
*Hình 1.10 Biểu đồ trình tự chức năng “Chỉnh sửa thông tin gia phả”*

* + - 1. **Use case “Sửa gia phả”**
         1. **Biểu đồ trình tự chức năng “Sửa thông tin thành viên”**



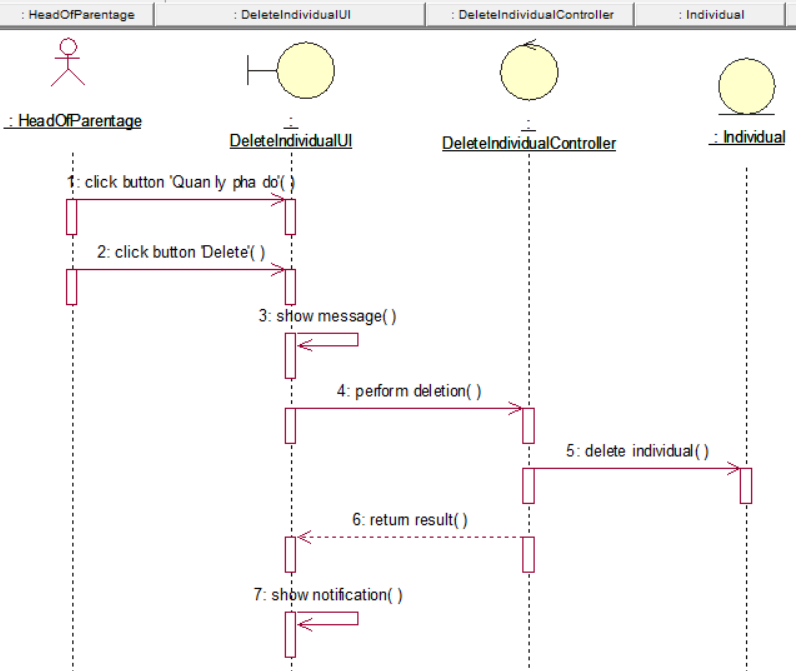
*Hình 1.11 Biểu đồ trình tự chức năng “Sửa thông tin thành viên”*

* + - * 1. **Biểu đồ trình tự chức năng “Thêm thành viên”**



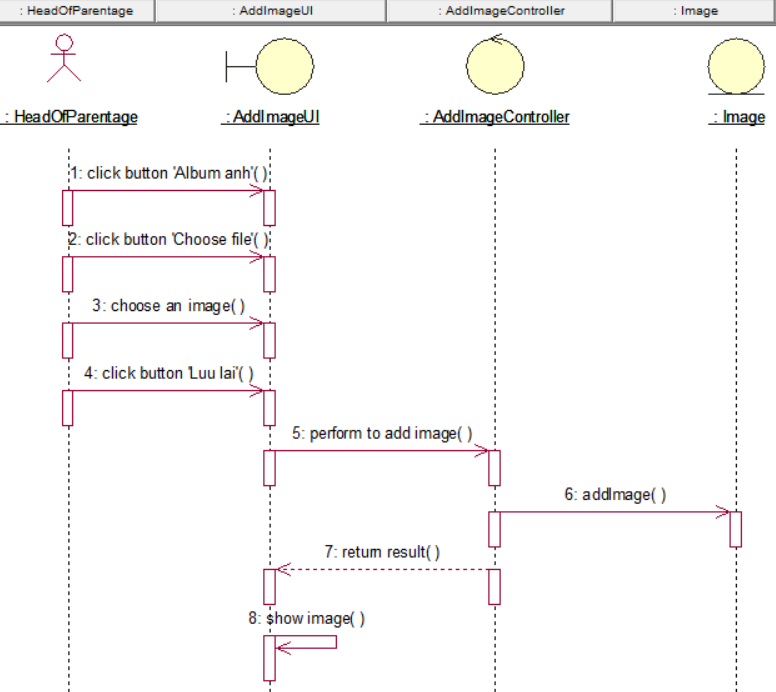
*Hình 1.12 Biểu đồ trình tự chức năng “Thêm thành viên”*

* + - * 1. **Biểu đồ trình tự chức năng “Xóa thành viên”**



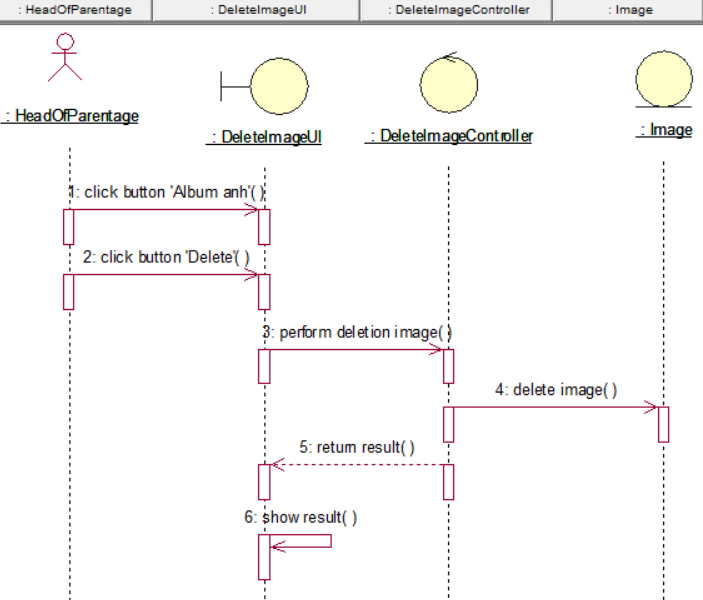
*Hình 1.13 Biểu đồ trình tự chức năng “Xóa thành viên”*

* + - 1. **Use case “Quản lý hình ảnh”**
         1. **Biểu đồ trình tự chức năng “Thêm ảnh”**



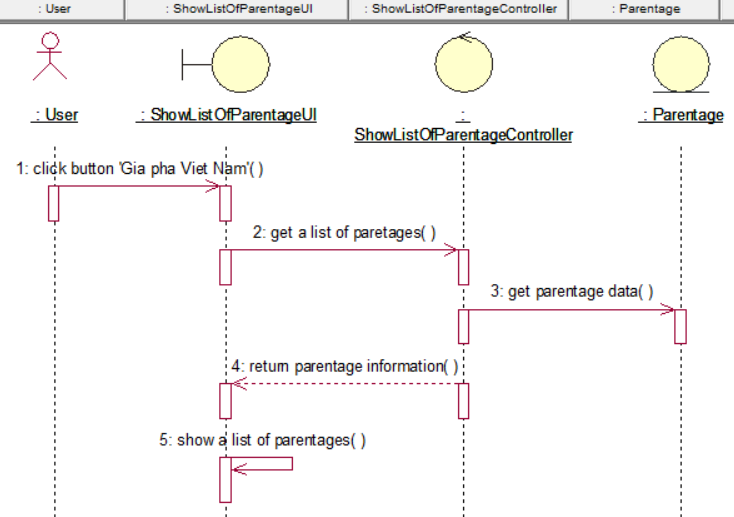
*Hình 1.14 Biểu đồ trình tự chức năng “Thêm ảnh”*

* + - * 1. **Biểu đồ trình tự chức năng “Xóa ảnh”**



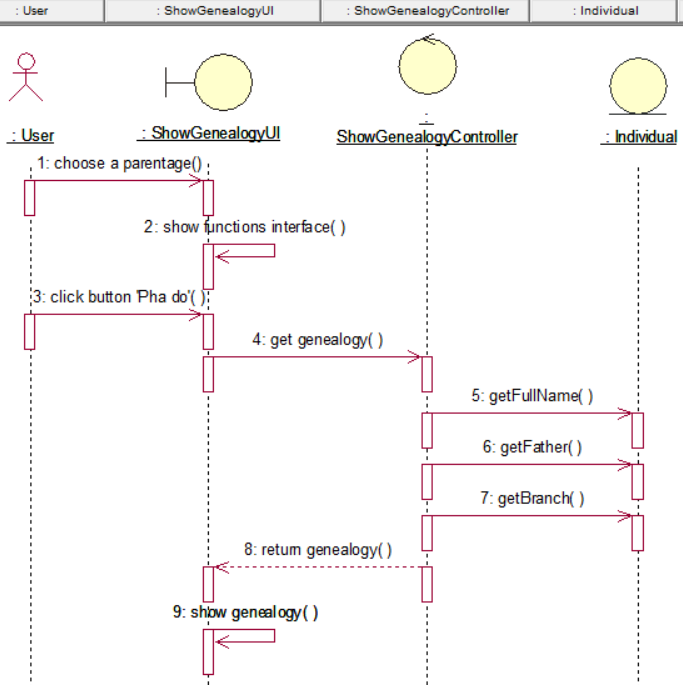
*Hình 1.15 Biểu đồ trình tự chức năng “Xóa ảnh”*

* + 1. **Phân tích use case “Xem thông tin gia phả”**
       1. **Biểu đồ trình tự chức năng “Xem thông tin dòng họ”**



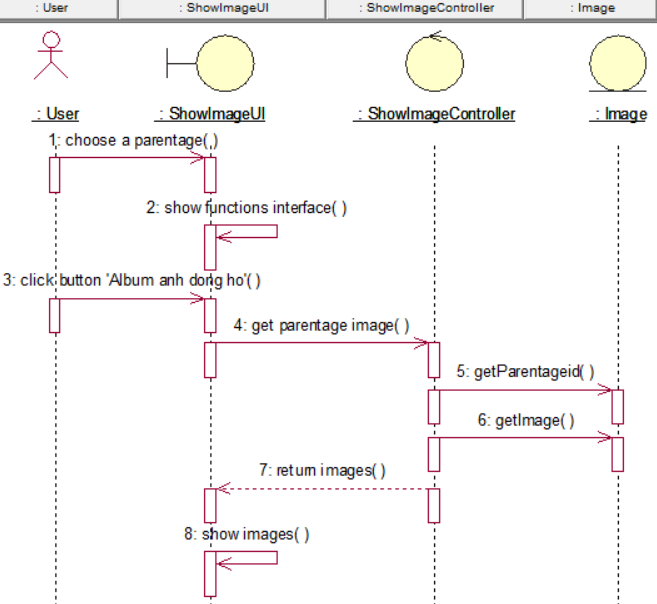
*Hình 1.16 Biểu đồ trình tự chức năng “Xem thông tin dòng họ”*

* + - 1. **Biểu đồ trình tự chức năng “Xem phả đồ”**



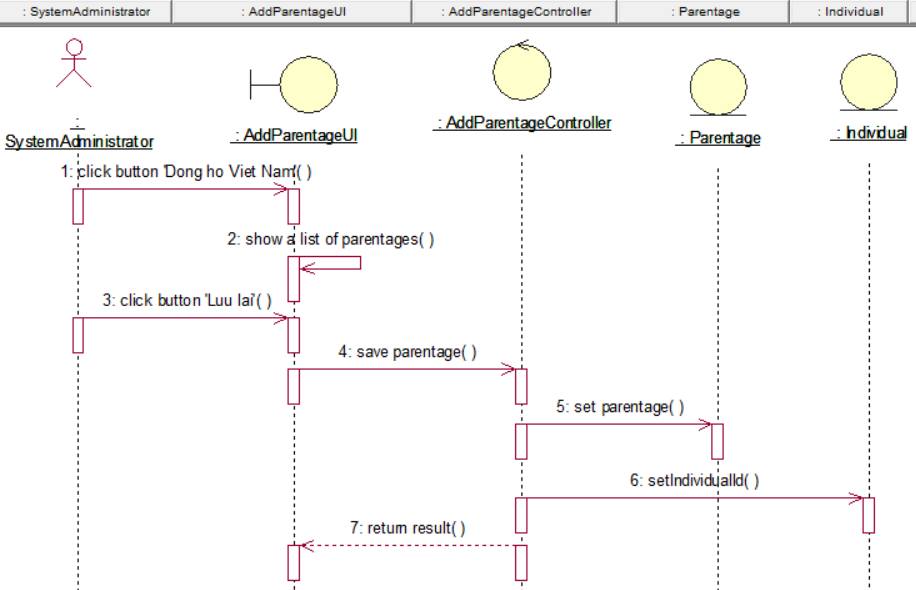
*Hình 1.17 Biểu đồ trình tự chức năng “Xem phả đồ”*

* + - 1. **Biểu đồ trình tự chức năng “Xem hình ảnh”**



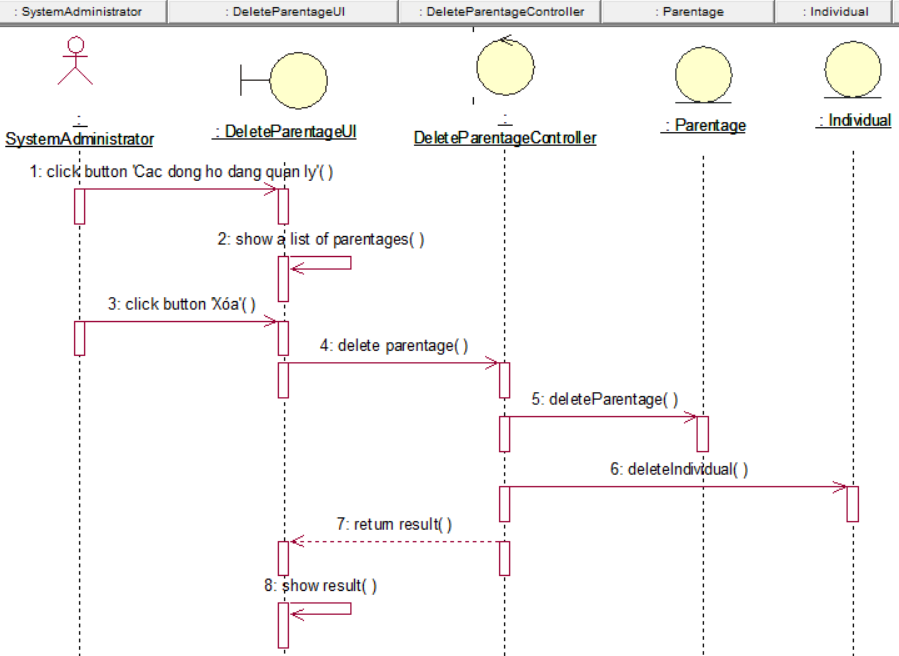
*Hình 1.18 Biểu đồ trình tự chức năng “Xem hình ảnh”*

* + 1. **Phân tích use case “Quản lý nội dung”**
       1. **Biểu đồ trình tự chức năng “Thêm dòng họ”**



*Hình 1.19 Biểu đồ trình tự chức năng “Thêm dòng họ”*

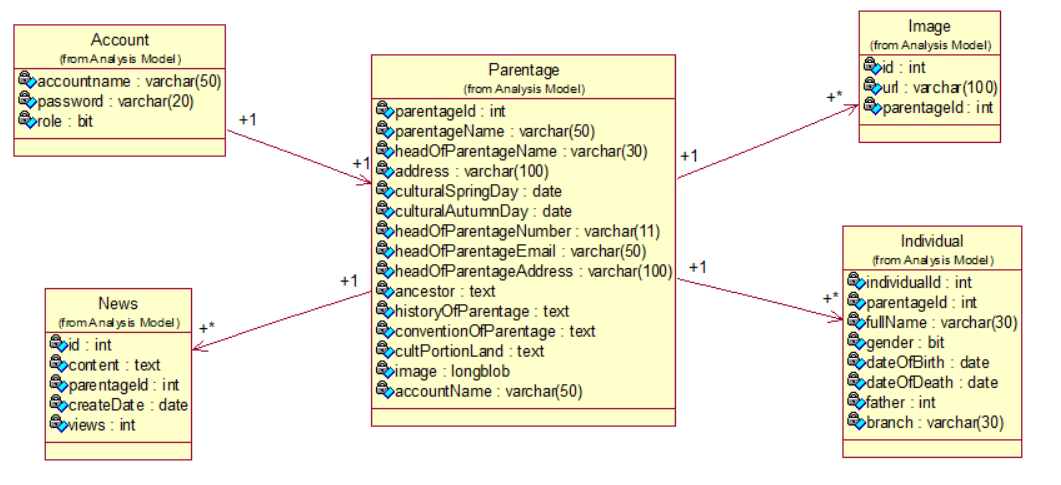
* + - 1. **Biểu đồ trình tự chức năng “Xóa dòng họ”**



*Hình 1.20 Biểu đồ trình tự chức năng “Xóa dòng họ”*

## **CHƯƠNG 2. THIẾT KẾ CƠ SỞ DỮ LIỆU**

* 1. **Thiết kế cơ sở dữ liệu - Ánh xạ lớp sang bảng**
     1. **Mô hình các lớp**



*Hình 2.1 Mô hình các lớp thực thể*

* + 1. **Nguyên tắc ánh xạ**
* Ánh xạ các lớp trong biểu đồ thực thể theo nguyên tắc:
  + Mỗi lớp thực thể sẽ được ánh xạ thành một bảng cơ sở dữ liệu.
* Quan hệ giữa Association các lớp:
  + Quan hệ 1-n: Chuyển khóa chính bên 1 sang bên nhiều làm khóa ngoại, loại bỏ các thuộc tính bên nhiều suy ra được từ khóa ngoại; chuyển các thuộc tính còn lại ở hai bên thành trường dữ liệu trong bảng.
  + Quan hệ n-n: Chuyển khóa chính hai bên nhiều sang lớp liên kết, loại bỏ các thuộc tính trong lớp liên kết mà suy ra được từ khóa ngoại, chuyển lớp liên kết thành bảng, chuyển hai lớp liên kết thành hai bảng, chuyển các thuộc tính còn lại thành trường dữ liệu.
  1. **Các bảng sau khi được ánh xạ**

**Bảng account:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | account\_name | Varchar (100) | Not null | Primary key |
| 2 | password | Varchar (100) | Not null |  |
| 3 | role | int | Not null |  |
| 4 | status | int | Not null |  |

**Bảng image:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | id | Int | Not null | Primary key |
| 1 | image\_name | Varchar (255) |  |  |
| 2 | url | Varchar (100) |  |  |
| 3 | parentage\_id | Int | Not null | Foreign key |

**Bảng individual:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | individual\_id | bigint | Not null | Primary key |
| 2 | parentage\_id | bigint | Not null | Foreign key |
| 3 | fullname | Varchar (100) |  |  |
| 4 | gender | bit |  |  |
| 5 | date\_of\_birth | date |  |  |
| 6 | date\_of\_death | date |  |  |
| 7 | father | int |  |  |
| 8 | branch | Varchar (255) |  |  |

**Bảng parentage:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | parentage\_id | bigint | Not null | Primary key |
| 2 | parentage\_name | int |  |  |
| 3 | head\_of\_parentage\_name | Varchar (100) |  |  |
| 4 | cultural\_spring\_day | date |  |  |
| 5 | cultural\_autumn\_day | date |  |  |
| 6 | head\_of\_parentage\_number | Varchar (10) |  |  |
| 7 | address | Varchar (100) |  |  |
| 8 | head\_of\_parentage\_email | Varchar (100) |  |  |
| 9 | ancestor | text |  |  |
| 10 | history\_of\_parentage | text |  |  |
| 11 | account\_id | bigint | Not null | Foreign key |

**Bảng news:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | id | bigint | Not null | Primary key |
| 2 | content | text | Not null |  |
| 3 | parentage\_id | int | Not null |  |
| 4 | create\_date | date |  |  |
| 5 | views | int |  |  |
| 6 | title | text |  |  |

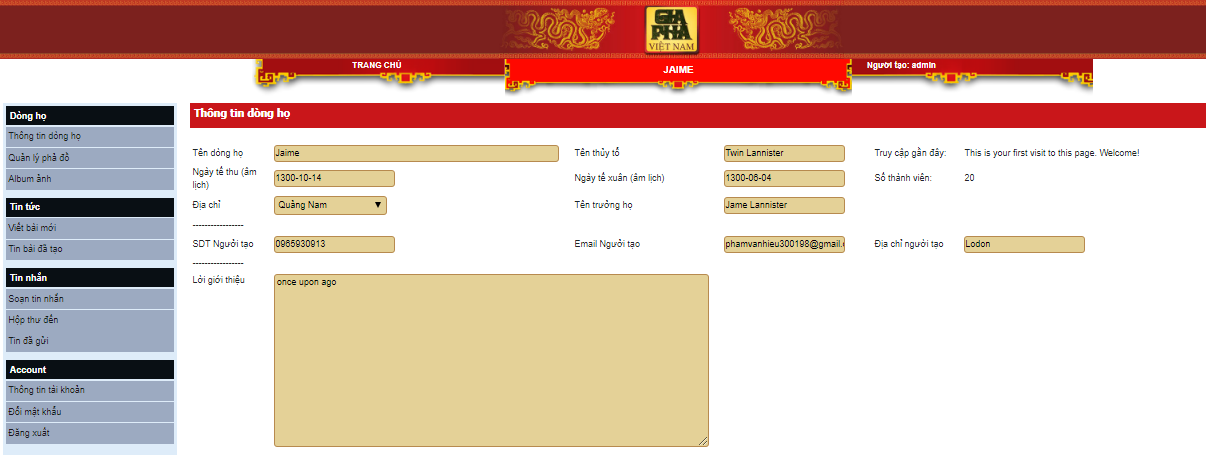
## **CHƯƠNG 3. THIẾT KẾ GIAO DIỆN**

* 1. **Giao diện**
     1. **Trang chủ**



*Hình 3.1 Giao diện quản lý thông tin dòng họ*

* + 1. **Quản lý thông tin dòng họ**



*Hình 3.2 Giao diện quản lý thông tin dòng họ*

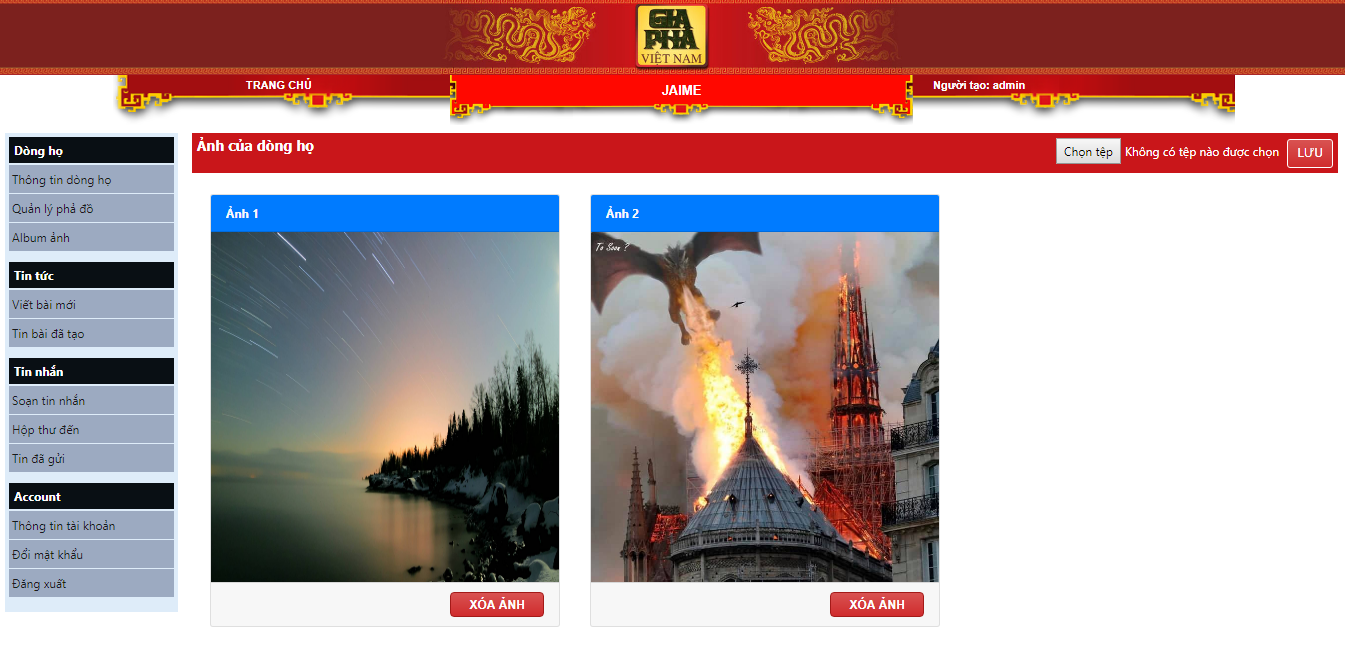
* + 1. **Quản lý phả đồ**





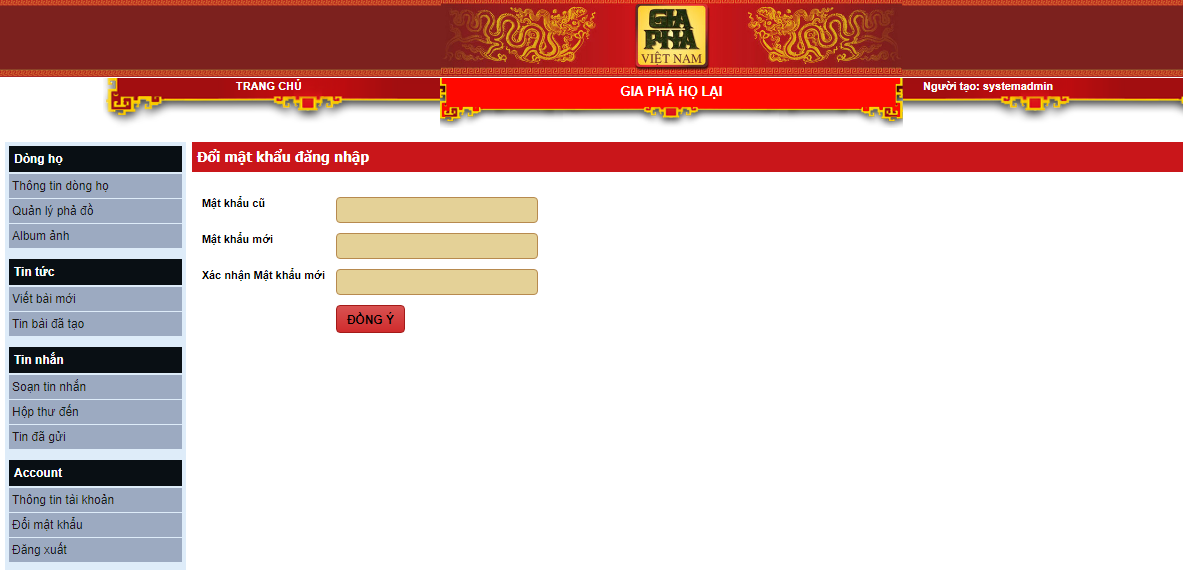
*Hình 3.3 Giao diện**quản lý phả đồ*

* + 1. **Quản lý hình ảnh**



*Hình 3.4 Giao diện quản lý hình ảnh*

* + 1. **Quản lý tài khoản**



*Hình 3.5 Giao diện quản lý tài khoản*

* + 1. **Xem thông tin dòng họ**



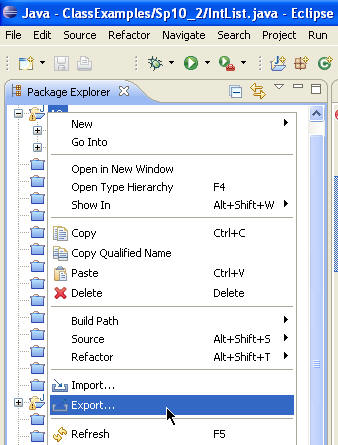


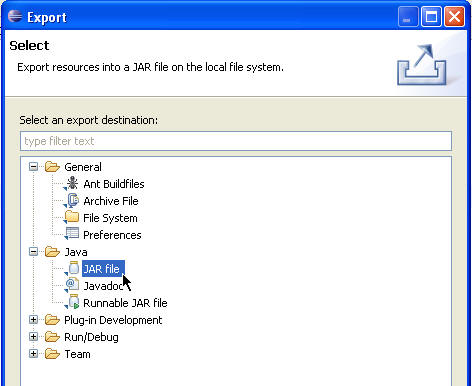


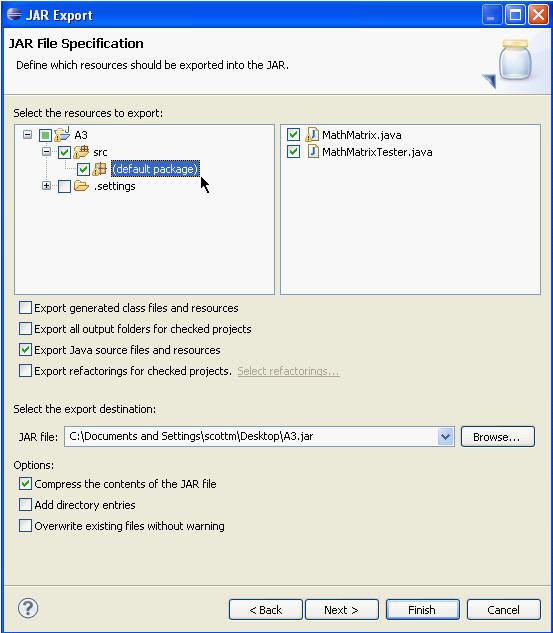
## **CHƯƠNG 4. THIẾT KẾ THƯ VIỆN, API**

* 1. **Cách tạo và sử dụng thư viện**
     1. Tạo file jar
* B1: Kích chuột phải vào project chọn “Export”.
* B2: Chọn định dạng file export, ở đây ta chọn “Jar file”.
* B3: Chọn folder muốn export và nơi lưu và nhấn “Finish”.

**Hình ảnh minh họa**

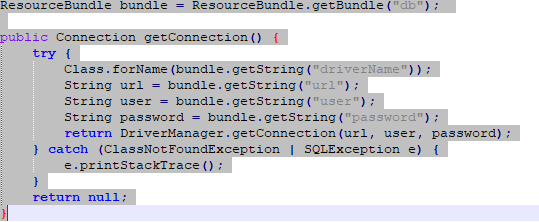


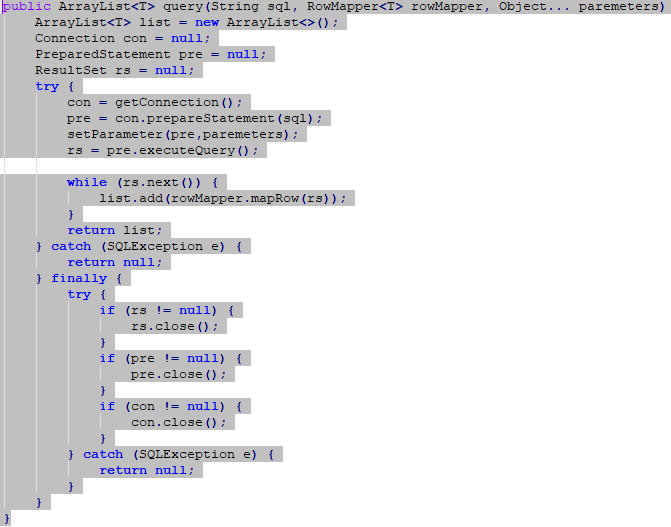




* + 1. Import vào project
* B1: Kích chuột phải vào project
* B2: Chọn Build Path
* B3: Chọn Configure Build Path
* B4: Chọn Libraries và chọn Add External JARS
* B5: Chọn file Jar vừa tạo bên trên
* B6: Chọn Apply
  1. **Thiết kế thư viện**
     1. Thư viện DAO

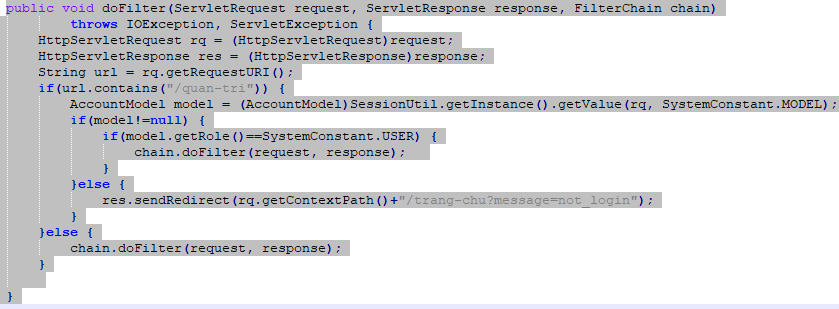
Chức năng: kết nối cơ sở dữ liệu và thực thi câu truy vấn sql





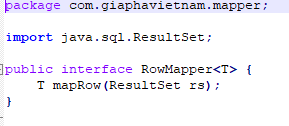
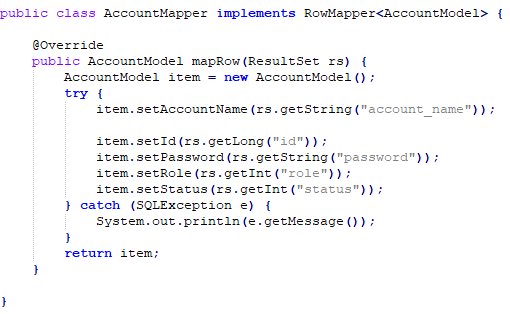
* + 1. Thư viện Filter

Chức năng: kiểm tra đăng nhập và phân quyền



* + 1. Thư viện Mapper

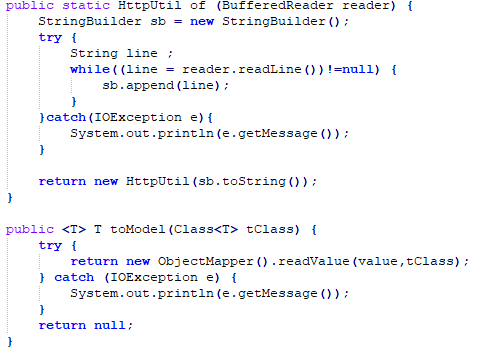
Chức năng: trả về model từ reaultset

* + 1. Thư viện Utils

Chức năng: Gồm các hàm chức năng.

Ví dụ: hàm trả về model lấy dữ liệu Json từ client.



* 1. **Thiết kế API**

API xây dựng cho các bảng để lấy dữ liệu và thao tác với cơ sở dữ liệu gồm các phương thức doGet(lấy dữ liệu), doPut(sửa thông itn), doPost(thêm thông tin), doDelete(xóa trường trong bảng).

**Ví dụ API của Individual**

