**BÁO CÁO LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

**NHÓM 2 : SUS@**

**GVHD: TS.Trịnh Tuấn Đạt**

1. **THÀNH VIÊN**
2. Đào Nhật Minh 20200392 (Leader)
3. Nguyễn Giang Nam 20205103
4. Nguyễn Sỹ Việt Nhật 20205011
5. Trần Thị Quỳnh Tiên 20205032
6. Nguyễn Thu Trang 20205035
7. Trần Anh Tuấn 20205041
8. **PHÂN CHIA CÔNG VIỆC**
9. **Xây dựng bài toán:**
10. Thu thập dữ liệu- > crawl data
11. Xây dựng biểu đồ lớp -> uml, usecase, classcase
12. Thiết kế đối tượng
13. Xây dựng UI
14. **Phần trăm đóng góp:**

| Đào Nhật Minh | 20 | Trần Anh Tuấn | 15 |
| --- | --- | --- | --- |
| Nguyễn Giang Nam | 20 | Nguyễn Thu Trang | 15 |
| Trần Thị Quỳnh Tiên | 15 | Nguyễn Sỹ Việt Nhật | 15 |

1. **THU THẬP DỮ LIỆU (Crawl Data)**

Dựa trên tìm kiếm các web thông tin về triều đại lịch sử, nhóm tập trung chủ yếu làm về 3 triều đại lớn của Việt Nam: nhà Lý, nhà Trần và nhà Lê. Trong đó:

Câu hỏi 1: Bao nhiêu nhân vật lịch sử, lễ hội, di tích,..?

→ Data crawl có 15 nhân vật,

Câu hỏi 2: Bao nhiêu thuộc tính dữ liệu cho mỗi thực thể?

→ Dynasty (5 thuộc tính); Event (6 thuộc tính); War (7 thuộc tính); Festival (3 thuộc tính); People (4 thuộc tính); King (9 thuộc tính); FamousPeople (6 thuộc tính);

Câu hỏi 3: Bao nhiêu liên kết giữa các thực thể?

→ Giữa các thực thể có 9 liên kết với nhau.

Câu hỏi 4: Lấy data của bao nhiêu nguồn? là những nguồn nào? Mỗi nguồn lấy được data gì?

→Dữ liệu được lấy từ nguồn: <https://dbpedia.org> Dữ liệu thu thập được bao gồm triều đại (Dynasty), nhân vật (nhân vật nổi tiếng - FamousPeople, vua - King), sự kiện lịch sử (sự kiện - Event, trận chiến - War) và lễ hội (Festival), di tích (Địa danh - Place, thủ đô - Capital, sông - River và một số địa danh liên quan khác).

1. **Ngôn ngữ**

Sử dụng SPARQL (phát âm là ‘sparkle’, là ngôn ngữ và giao thức truy vấn tiêu chuẩn cho dữ liệu mở liên kết (Linked Open Data) trên web hoặc trong cơ sở dữ liệu đồ họa ngữ nghĩa (còn được gọi là RDF triplestore - bộ 3 RDF), cho phép những người sử dụng truy vấn thông tin từ các cơ sở dữ liệu hoặc bất kỳ nguồn dữ liệu nào có thể được ánh xạ tới RDF.

**Ưu điểm:**

* Chỉ cần đưa vào các câu lệnh, toàn bộ quá trình thu thập đều diễn ra tự động (ưu thế hơn so với sử dụng JSOUP do JSOUP không có tính tự động, người dùng phải can thiệp, chọn lọc thông tin từ các trang web, mỗi web lấy 1 vài thông tin chứa trong các tab file html).
* Có khả năng phân loại và sắp xếp trên thông tin đầu ra. Dễ dùng tái sử dụng, chỉ cần thay thế, sửa chữa 1 chút có thể lấy thêm dạng thông tin mới. Đây là ưu thế tuyệt đối của SPARQL, đa số các ngôn ngữ truy vấn hỗ trợ rất ít chức năng gộp nhóm và tổng hợp.
* Cho phép những người sử dụng truy vấn thông tin từ các cơ sở dữ liệu hoặc bất kỳ nguồn dữ liệu nào có thể được ánh xạ tới RDF…

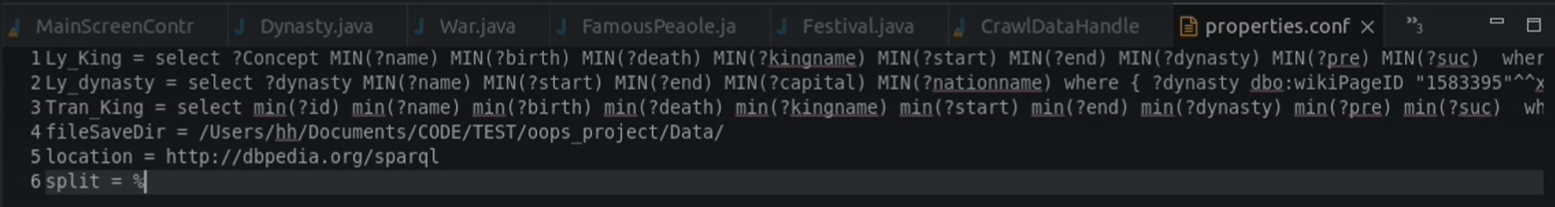
1. **Thu thập dữ liệu**

* Quá trình thu thập dữ liệu gồm 2 bước:
* Load data từ internet.

Dữ liệu được lấy từ <https://dbpedia.org>

Sau khi load dữ liệu từ internet về, dữ liệu được lưu dưới dạng bảng query bằng ngôn ngữ SPARQL.

* Chuyển về dạng file json và lưu trữ.
* Các câu lệnh trong file properties.conf cho phép load data vào các biến dữ liệu có trong từng object tương ứng.

**

Cấu trúc cơ bản của file dữ liệu json (mẫu) bao gồm:

Id = 1457828

Tên thật: Trần Cảnh

Ngày sinh: 1218 - 07 - 17

Ngày mất: 1277 - 05 - 05

Tên hiệu: Trần Thái Tông

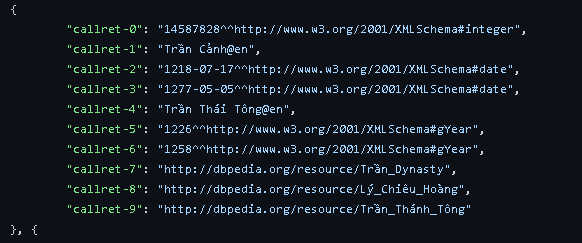
Thời gian trị vì:

* năm bắt đầu trị vì: 1226
* năm kết thúc trị vì: 1258

Triều đại: Trần\_Dynasty

Người tiền vị: Lý\_Chiêu\_Hoàng

Người kế vị: Trần\_Thánh\_Tông



Mã nguồn thu thập và xử lý dữ liệu được lưu trong 2 package: Crawler và Handler.

Package Crawler: chứa các class Constant và LoadCache.

* Class Constant: chứa các từ khóa được sử dụng lặp đi lặp lại nhiều lần, tên các thư mục chứa data được crawl về; lưu thông tin lấy từ file properties.
* Class LoadCache: tải bộ đệm danh sách Vua (King) và Triều đại (Dynasty)

Package Handler: chứa 2 lớp CrawlDatatHandle và ReadDataHandle.

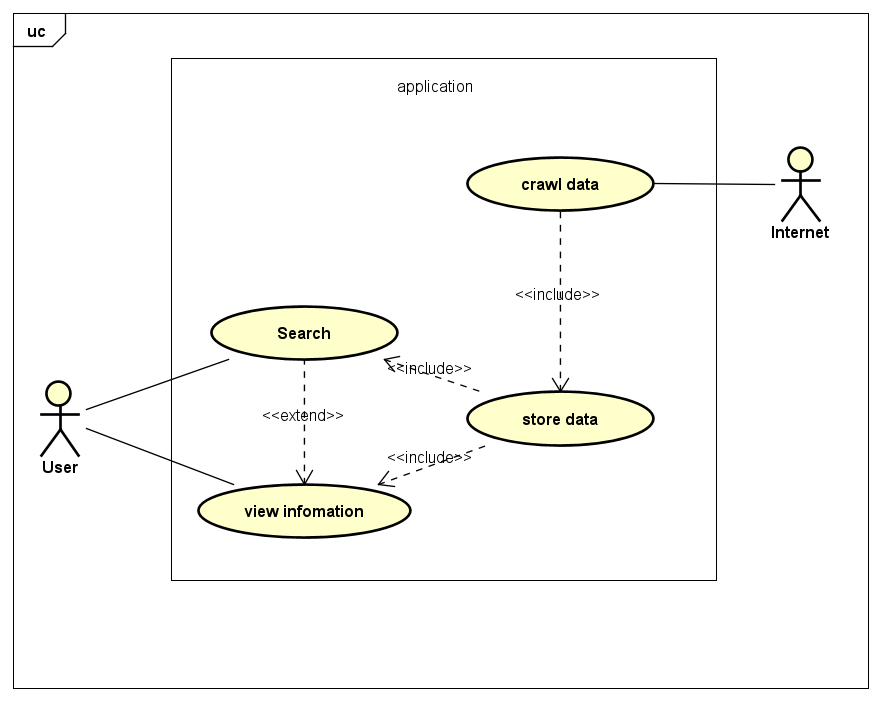
* Load conf (load toàn bộ thông tin từ file properties lên code, load tương ứng vào các từ khóa sử dụng nhiều lần trong file constant.java.
* Crawl data làm tương tự như sql, tạo 1 query, execution bằng ngôn ngữ sparqlService (load 2 thông tin là location: (dbpedia.org) kho dữ liệu được lấy query câu lệnh thực hiện truy vấn key và value.
* Lấy các result bằng lệnh ResultSet (như câu lệnh sql, lệnh này chứa các dòng dữ liệu của database, tham chiếu nó là các trường dữ liệu, là các cột), xuất toàn bộ thông tin trong ResultSet về thành String bằng hàm ExportToString (là hàm chuyển data từ dạng ResultSet thành bố cục file json.
* Xuất từ data dạng String đưa ra file để lưu trữ tại local của mình(data đưa vào gồm list - thông tin vừa chuyển, fileSaveDir (trong properties.conf) đường dẫn thư mục lưu trữ data crawl về + tên thư mục).
* Đóng execution và clear Params.
* Đọc và trích xuất thông tin cần lấy trong file properties.conf.

=> Toàn bộ data crawl trên web sau khi select được đưa về dạng ResultSet, đưa về dạng String rồi lưu dưới dạng file json, từ đấy load thông tin lên thành các object đưa lên màn hình.

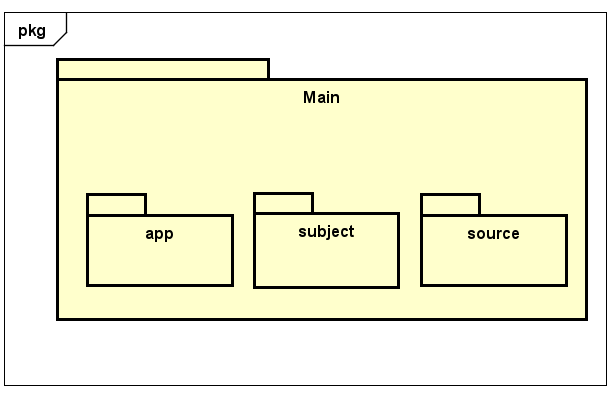
1. **BIỂU ĐỒ**

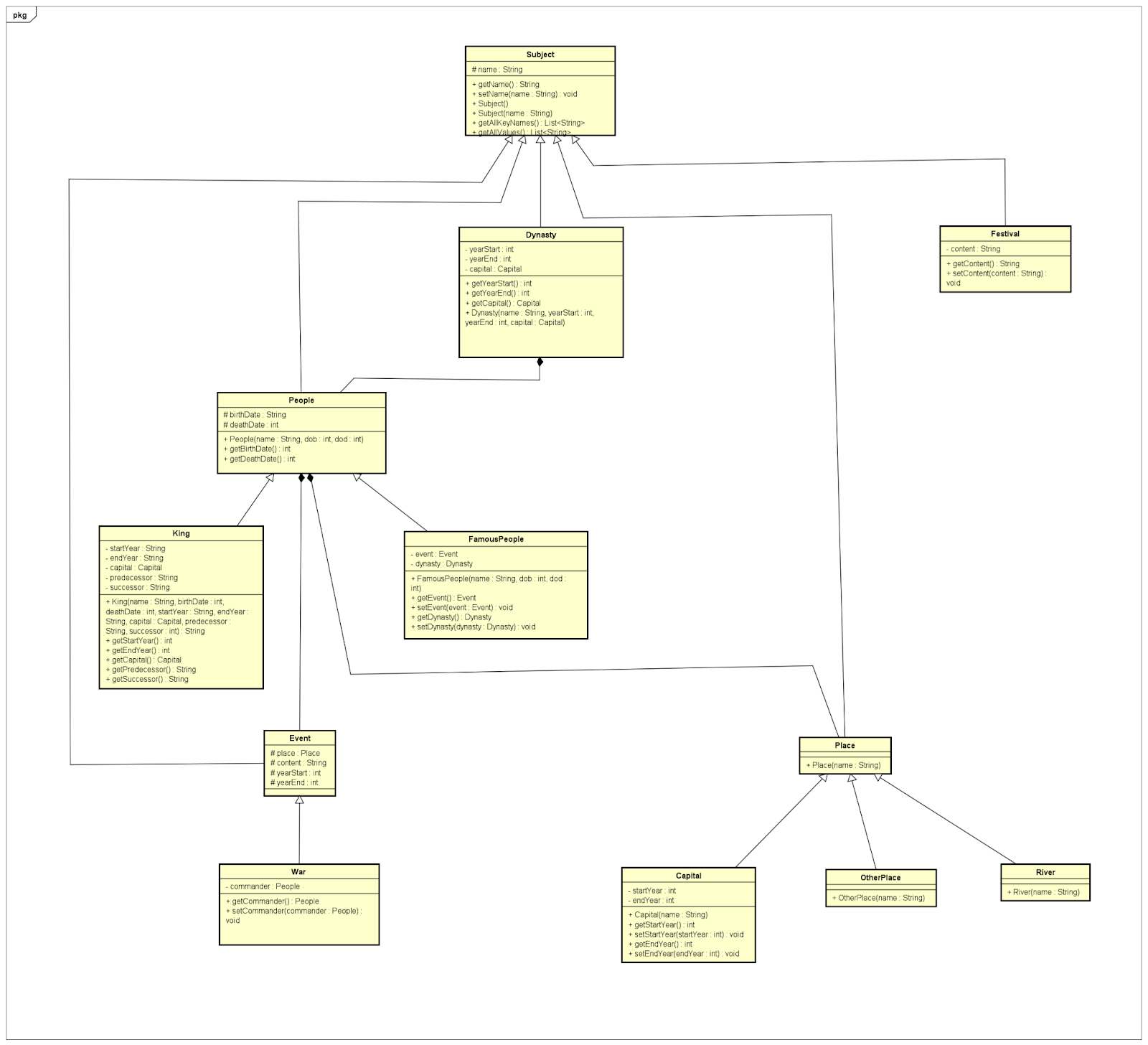
**Biểu đồ UML:**

* Biểu đồ usecase bao gồm các chức năng: lấy dữ liệu từ internet và lưu dữ liệu. User có chức năng tìm kiếm và xem thông tin:

****

* Biểu đồ package bao gồm package lớn Main trong đó chứa 3 package nhỏ: app, source, subject.

**

Biểu đồ lớp (Class Diagram) của package subject: **

1. **THIẾT KẾ ĐỐI TƯỢNG**

Source code (main) bao gồm 3 package: subject, app, source;

1. **Package subject:**

* Gồm 5 child package: dynasty, event, person, place và festival (Đối tượng có thể chứa thông tin về triều đại lịch sử, nhân vật, di tích, sự kiện hoặc là lễ hội) và lớp Subject. Tất cả lớp đối tượng khác đều kế thừa từ lớp Subject.

1. **Lớp Subject:**

Trong lớp Subject gồm tên đối tượng (name) và một danh sách xâu list<String> chứa từ khóa (getAllKeyNames) các trường thông tin và 1 danh sách chứa giá trị của nó (getAllValues).

1. **Package dynasty:** chứa class Dynasty kế thừa từ class Subject.

* Trong 1 triều đại (dynasty), sẽ có các dữ liệu về tên (countryName) thủ đô (capital), năm tồn tại (bao gồm năm bắt đầu - yearStart và năm kết thúc - yearEnd).
* Xây dựng phương thức khởi tạo cho các biến để lấy dữ liệu từ database.
* Thủ đô capital được kế thừa từ lớp place.

1. **Package event:**

* Gồm 2 class: Sự kiện (Event) và Chiến tranh (War).
* Class Event: Trong mỗi sự kiện (event) sẽ chứa các biến tên sự kiện (content), địa điểm (place), năm bắt đầu (yearStart), năm kết thúc (yearEnd).
* Class War: Chiến tranh được kế thừa từ lớp Event, thêm tên người chỉ huy.

1. **Package person:**

* Gồm 3 class : People, FamousPeaole và King;
* Lớp trừu tượng People: cung cấp thông tin về các nhân vật bao gồm tên (name), ngày sinh (birthDate) và ngày mất (deathDate).
* King: kế thừa từ lớp People, gồm các thông tin tên, năm trị vì (năm bắt đầu, năm kết thúc), thủ đô.
* FamousPeople (nhân vật nổi tiếng): kế thừa từ lớp People, gồm các thông tin về triều đại và sự kiện liên quan. Thông tin về nhân vật như tên, ngày sinh, ngày mất được kế thừa từ lớp people. Thêm dữ liệu về sự kiện liên quan và triều đại bằng cách ghi đè lại list<String> getAllKeyNames và getAllValues.

1. **Package place:**

* Gồm 4 class địa điểm (Place), thủ đô (Capital), sông (river) và địa điểm liên quan khác (OtherPlace).
* Class Place, River và OtherPlace đều chỉ chứa tên của các địa điểm (name).

Class Capital chứa tên của thủ đô(name) và thêm năm đóng đô (năm bắt đầu và năm kết thúc).

Cả ba class Capital, River và OtherPalce đều được kế thừa từ class Place.

1. **Package festival:**

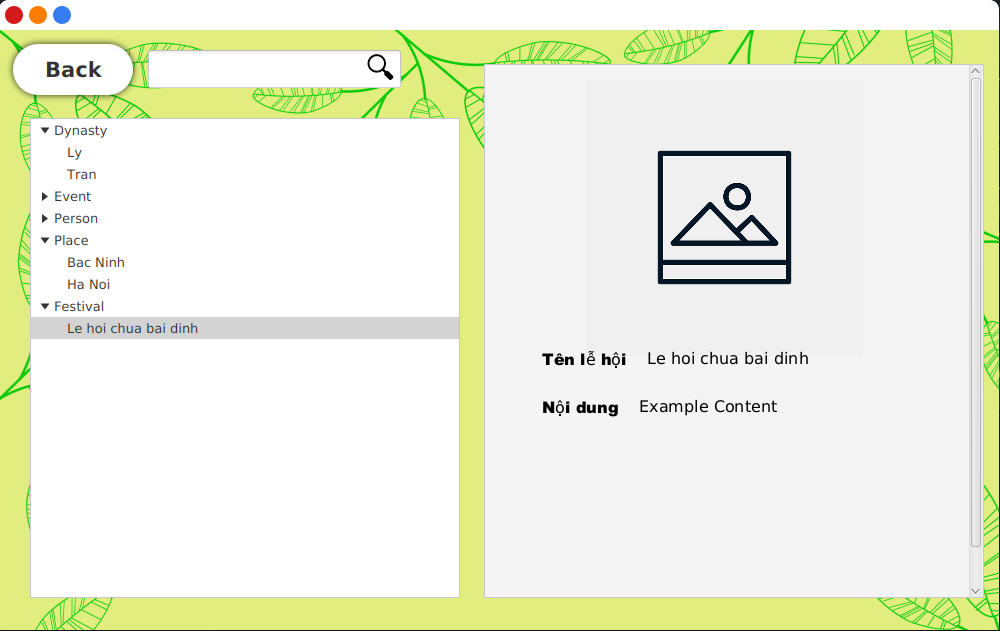
* Chỉ chứa tên và nội dung một số lễ hội, kế thừa class Subject.

1. **Thiết kế đồ họa**

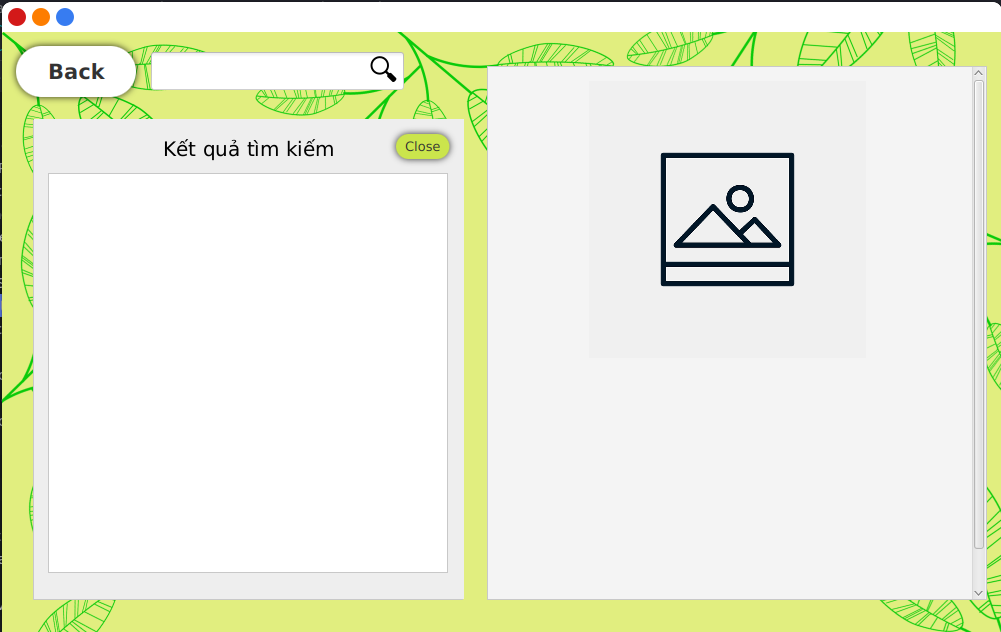
* Giao diện chính:
* Giao diện bắt đầu: bao gồm các chức năng bắt đầu, lấy dữ liệu và thoát:



* Màn hình bắt đầu gồm chức năng chọn thông tin và tìm kiếm thông tin:



* Màn hình sau khi tìm kiếm:



**Ⅵ. KỸ THUẬT LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG ÁP DỤNG**

* **Tính đóng gói (Encapsulation):** không cho phép người dùng trực tiếp tác động đến dữ liệu trong đối tượng mà thông qua các phương thức đối tượng cung cấp.

Ví dụ: các lớp dynasty, war, festival,.. tạo mộ trường của lớp private và cung cấp khả năng truy cập trường này thông qua các phương thức public.

* **Tính kế thừa (Inheritance):** kế thừa, tái sử dụng phương thức, thuộc tính của lớp cơ sở; thừa hưởng những gì lớp cha có và cho phép.

Ví dụ: các đối tượng đều được kế thừa từ lớp Subject.

* **Tính đa hình (Polymorphism):** cho phép các chức năng khác nhau được thực thi khác nhau trên các đối tượng khác nhau.

Ví dụ: King và FamousPeople đều kế thừa từ lớp People

**Các công nghệ, thuật toán hay ho được sử dụng?**

* **sparql**(ngôn ngữ và giao thức truy vấn): như đã nói ở phần ngôn ngữ trong phần thu thập dữ liệu, toàn bộ đều tự động, đưa vào các câu lệnh như mô hình jsoup(chọn các trang web, mỗi web lấy 1 vài thông tin bằng các tab file html, không có tính tự động, ta phải can thiệp, chọn lọc thông tin).

**Hướng dẫn sử dụng và demo chương trình**

[Video demo](https://drive.google.com/drive/folders/1rJzescm-Tz-8jcwGF6TDvrW82FChNCtE?usp=sharing)