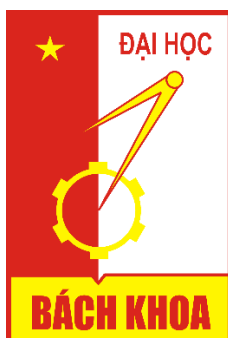


ĐẠI HỌC BÁCH KHOA HÀ NỘI  
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



**BÁO CÁO BÀI TẬP LỚN**  
**LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

**Đề tài:** Thu thập tự động dữ liệu về lịch sử Việt Nam qua các trang web và liên kết các dữ liệu để tìm kiếm, hiển thị

Giảng viên : **Trịnh Tuấn Đạt**

Học phần : IT3103

Mã lớp : 130934

Nhóm sinh viên thực hiện: Nhóm 17

STT	Họ và tên	MSSV
1	Vũ Minh Đăng	20205063
2	Phạm Trung Dũng	20205072
3	Nguyễn Duy Hùng	20204984
4	Nguyễn Văn Hảo	20204965
5	Nguyễn Thế Thương	20205031
6	Phạm Xuân Duy	20200114

**Hà Nội, 02 năm 2023**

## I. Phân công công việc và đóng góp

Họ và Tên	MSSV	Phân công công việc	Đóng góp
Vũ Minh Đăng	20205063	- Nhóm trưởng: Phân chia công việc, quản lý mã nguồn - Fix bug, quản lý cấu trúc project - Viết báo cáo	18%
Phạm Trung Dũng	20205072	- Lập trình thực thể Triều đại (Dynasty): Lấy và trả dữ liệu - Lập trình giao diện GUI	17%
Nguyễn Duy Hùng	20204984	- Lập trình thực thể Nhân vật (Character) + Vua (King): Lấy và trả dữ liệu - Vẽ biểu đồ lớp	18%
Nguyễn Văn Hảo	20204965	- Lập trình thực thể Sự kiện (Event): Lấy và trả dữ liệu - Lập trình giao diện GUI	17%
Nguyễn Thế Thương	20205031	- Lập trình thực thể Lễ hội (Festival): Lấy và trả dữ liệu - Vẽ biểu đồ phụ thuộc package	15%
Phạm Xuân Duy	20200114	- Lập trình thực thể Di tích (Relic): Lấy và trả dữ liệu - Slide và video demo	15%

## II. Thống kê dữ liệu

### 1. Dữ liệu đã lấy

Tổng hợp dữ liệu

×



Danh sách URL chính:

Web Triều đại : [https://vi.wikipedia.org/wiki/L%E1%BB%8Bch\\_s%E1%BB%AD\\_Vi%E1%BB%87t\\_Nam](https://vi.wikipedia.org/wiki/L%E1%BB%8Bch_s%E1%BB%AD_Vi%E1%BB%87t_Nam)

Web Lễ hội : [https://vi.wikipedia.org/wiki/L%E1%BB%85\\_h%E1%BB%99i\\_Vi%E1%BB%87t\\_Nam](https://vi.wikipedia.org/wiki/L%E1%BB%85_h%E1%BB%99i_Vi%E1%BB%87t_Nam)

Web Di tích : <http://dsvh.gov.vn/danh-muc-di-tich-quoc-gia-dac-biet-1752>

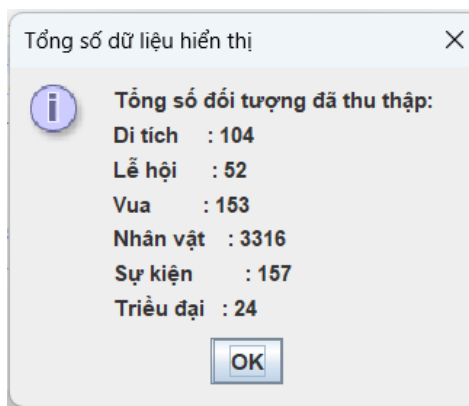
Web Vua : [https://vi.wikipedia.org/wiki/Vua\\_Vi%E1%BB%87t\\_Nam](https://vi.wikipedia.org/wiki/Vua_Vi%E1%BB%87t_Nam)

Web Nhân vật : <https://vansu.vn/viet-nam/viet-nam-nhan-vat?page=0>

Web Triều đại : <https://nguoiikesu.com/dong-lich-su>

Web Lễ hội : <https://www.couturetravelcompany.com/cac-le-hoi-o-viet-nam/>

OK



## 2. Thuộc tính dữ liệu

- Tất cả các thực thể đều có 3 thuộc tính chung: name, time, description (abstract class History)
- Vua (King) – mienHieu - 16: mienHieu, thuyHieu, nienHieu, tenHuy, triVi, tienNhiem, keNhiem, sinh, mat, trieuDai, anTang, thanPhu, thanMau
- Nhân vật (Character) - 6: otherName, place, era
- Triều đại (Dynasty) – 7: kingdom, captical, king(list), government
- Sự kiện (Event) - 3:
- Lễ hội (Festival) – 6: fisrtTime, place, character(list)
- Di tích (Relic) - 7: place, province, certifacte, reference

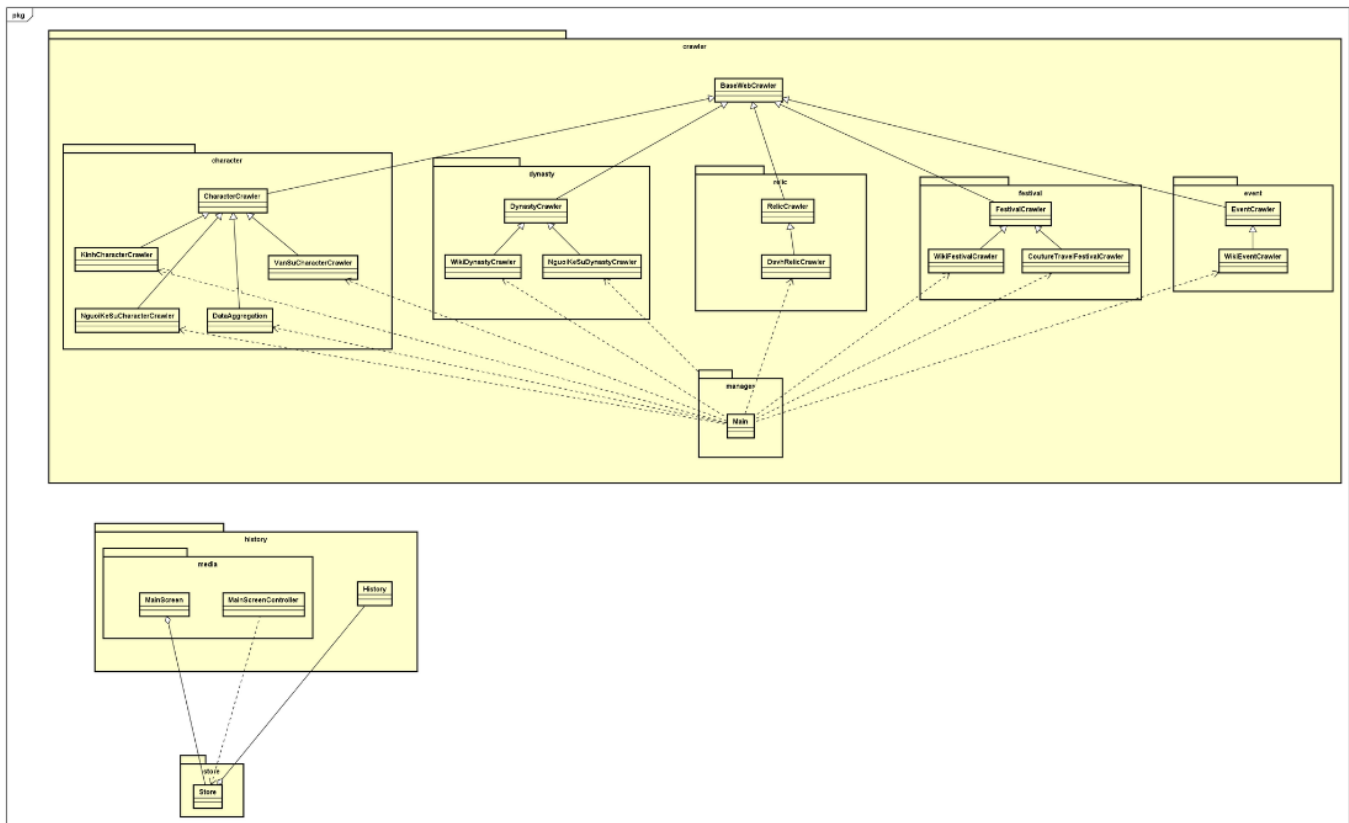
## 3. Liên kết dữ liệu

- Tổng số liên kết các thực thể 5:
  - Nhân vật – Triều đại
  - Vua – Nhân vật
  - Sự kiện – Nhân vật

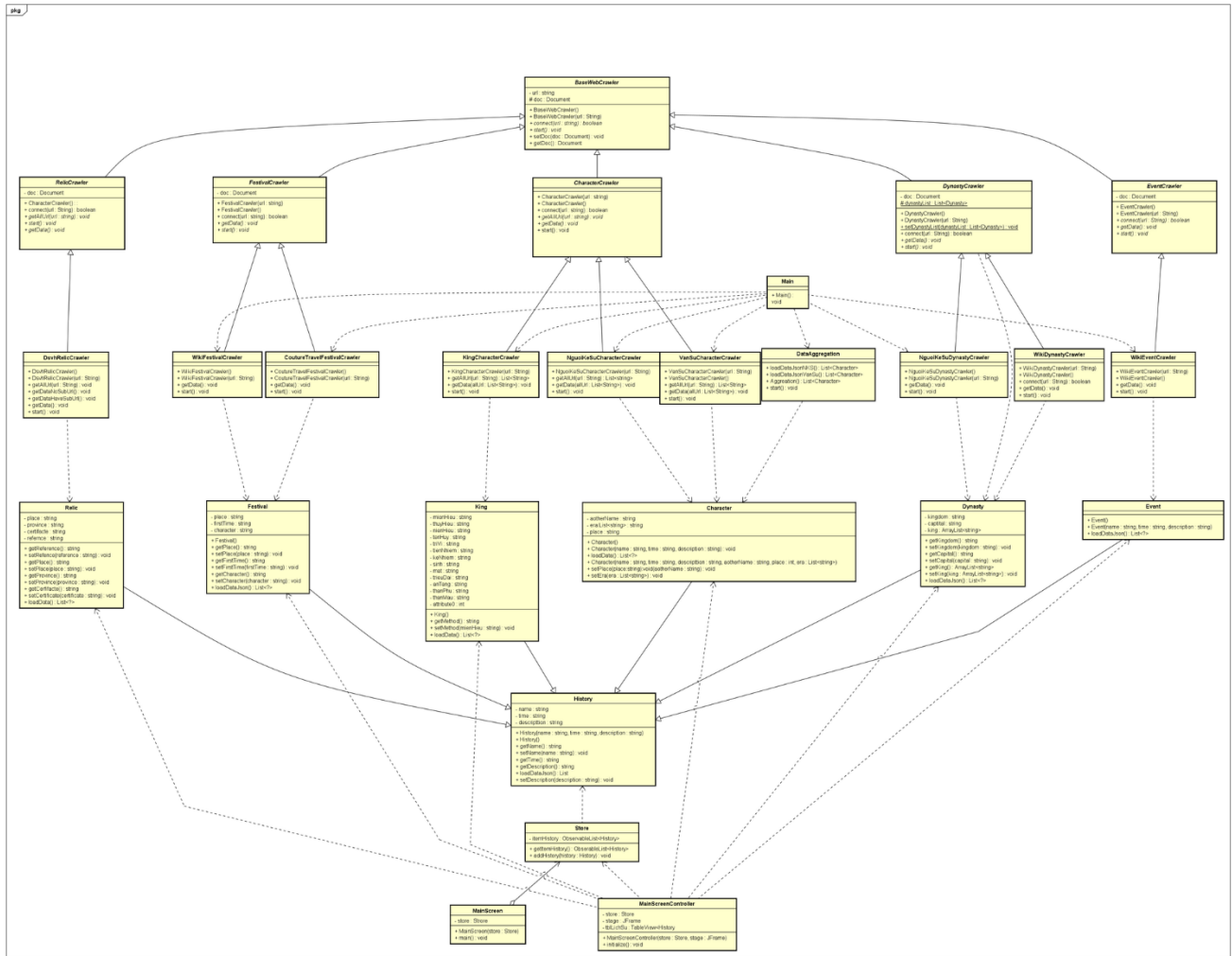
- Triều đại – Vua
- Lễ hội – Nhân vật

### III. Biểu đồ UML

#### 1. Biểu đồ phụ thuộc package

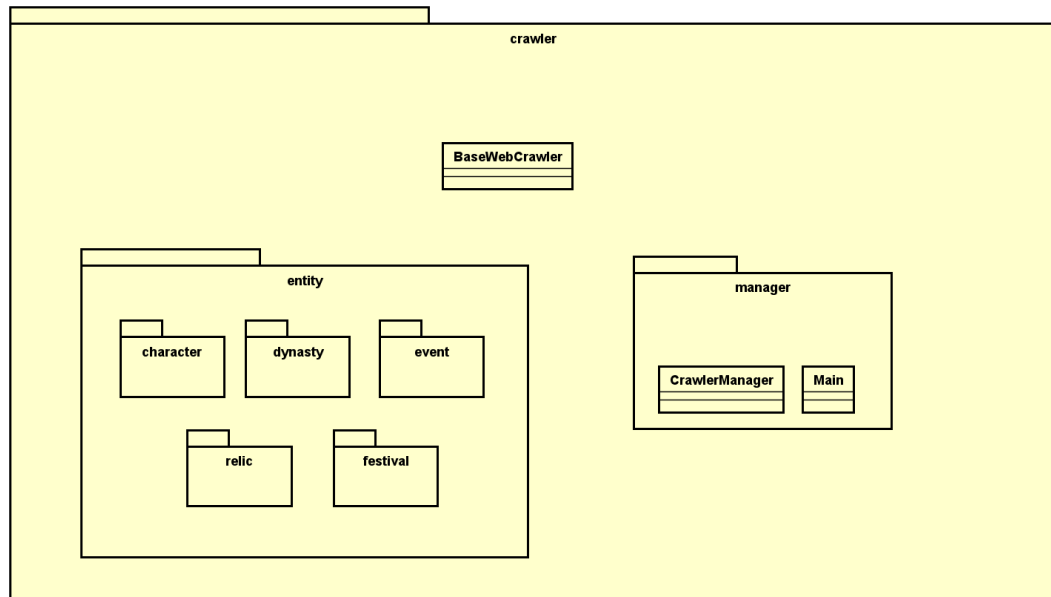


#### 2. Biểu đồ phụ thuộc lớp

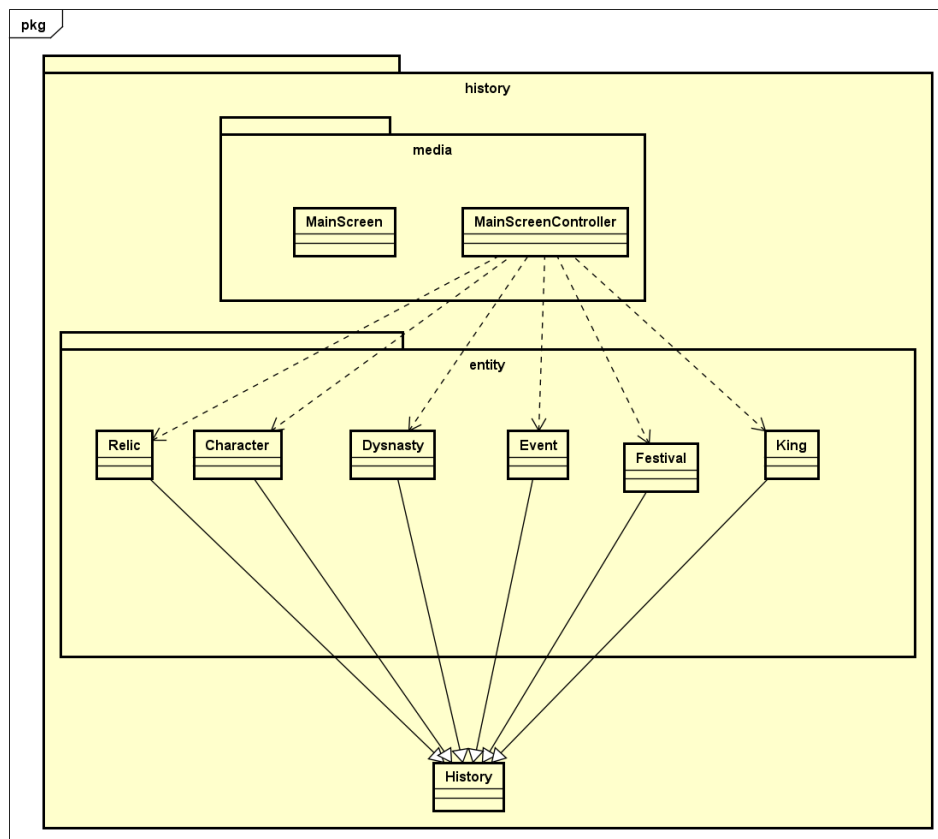


## IV. Phân tích thiết kế

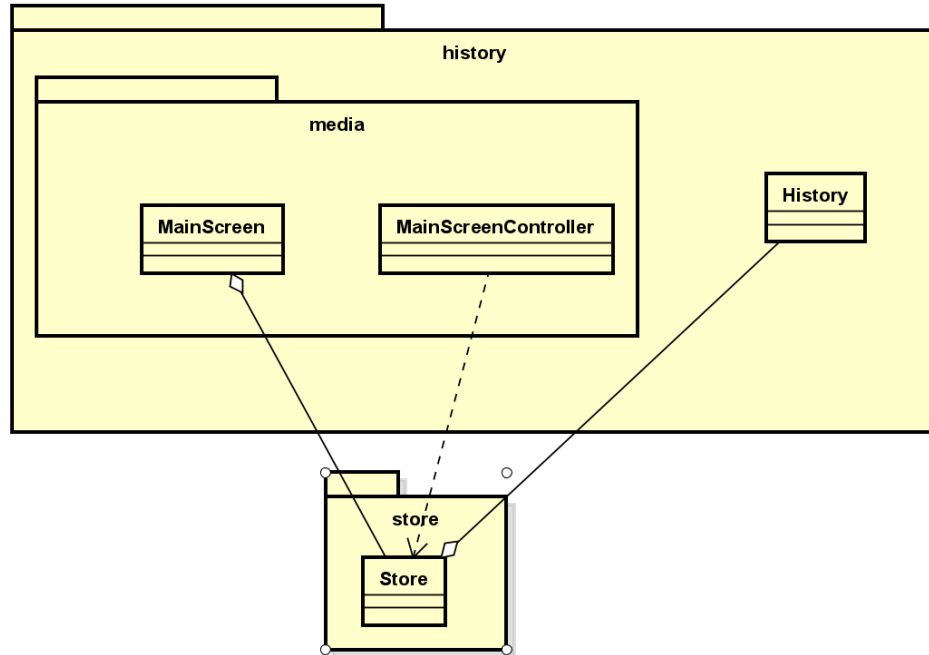
### 1. Phân tích biểu đồ phụ thuộc package



- Package crawler sẽ chứa 2 package: entity, manager và lớp class BaseWebCrawler.
- Trong package entity sẽ có 5 package là : character,dynasty,event,relic,festival.
- Trong package manager gồm : lớp CrawlerManager và lớp Main dùng để chạy CT.

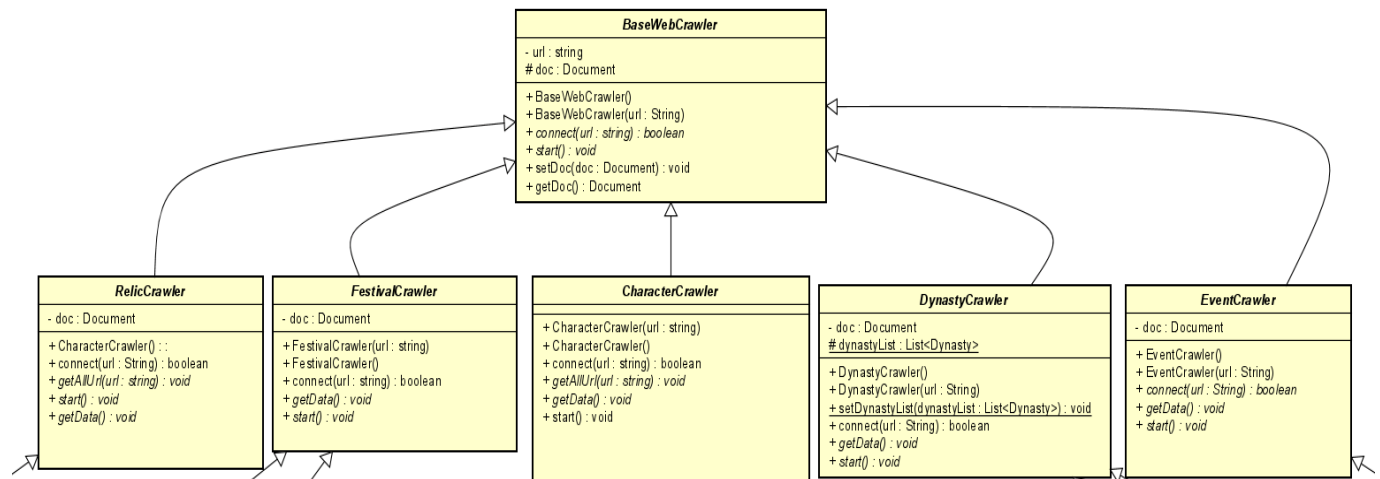


- Trong package history gồm 2 package :media , entity và lớp trừu tượng history.
- Trong package media gồm có lớp MainScreen và MainScreenController
- Trong package entity gồm 5 lớp: Relic,Character,Dynasty,Event,King,Festival

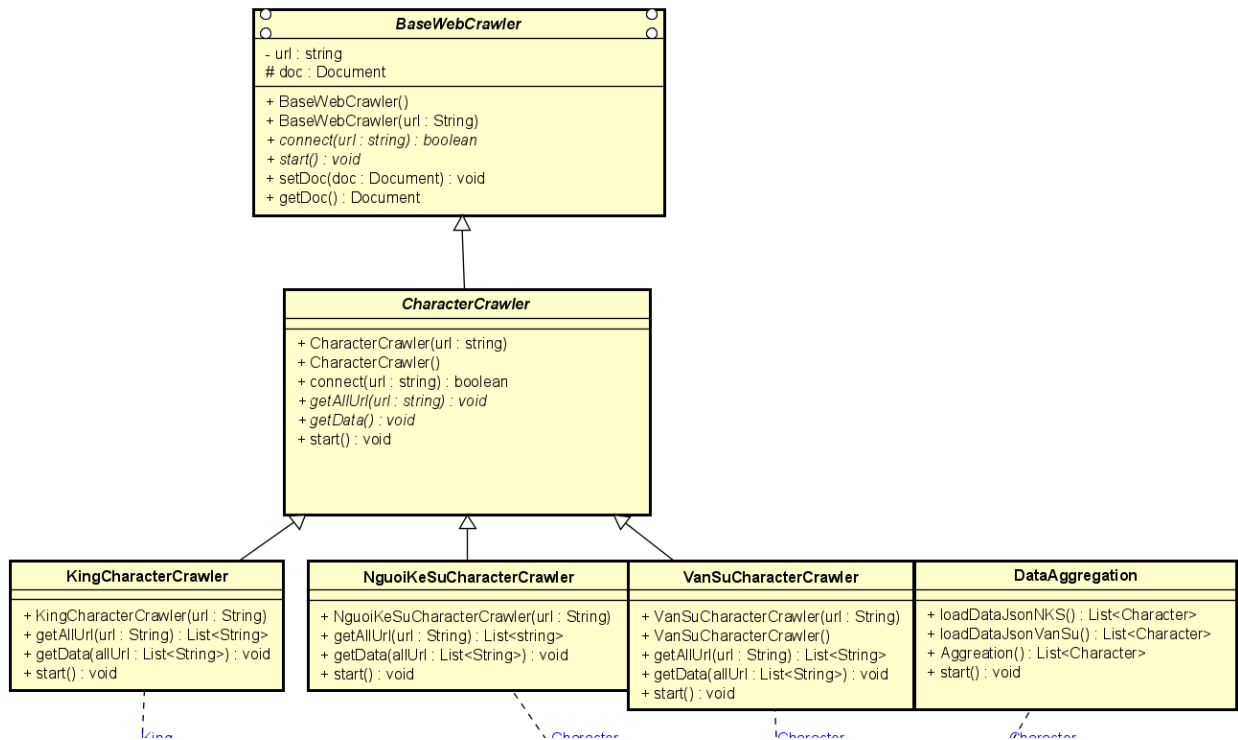


- Trong package store có lớp Store dùng để lưu trữ các đối tượng của lớp lịch sử

## 2. Phân tích biểu đồ lớp



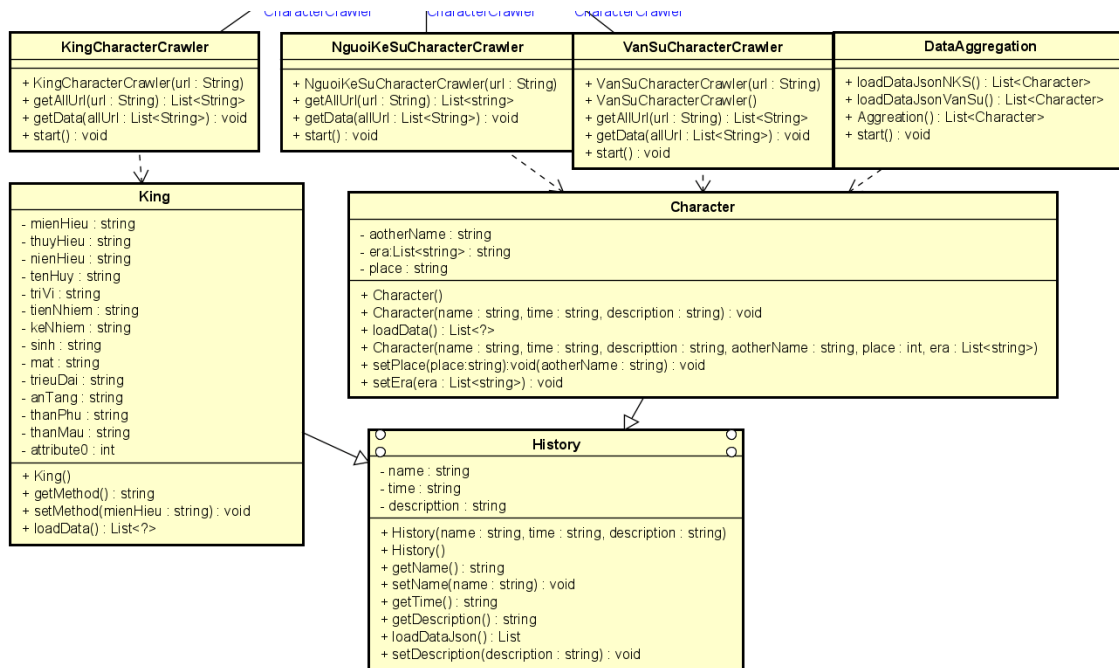
- 5 lớp trừu tượng gồm : RelicCrawler , FestivalCrawler, CharacterCrawler, DynastyCrawler,EventCrawler. Kế thừa từ BaseWebCrawler.



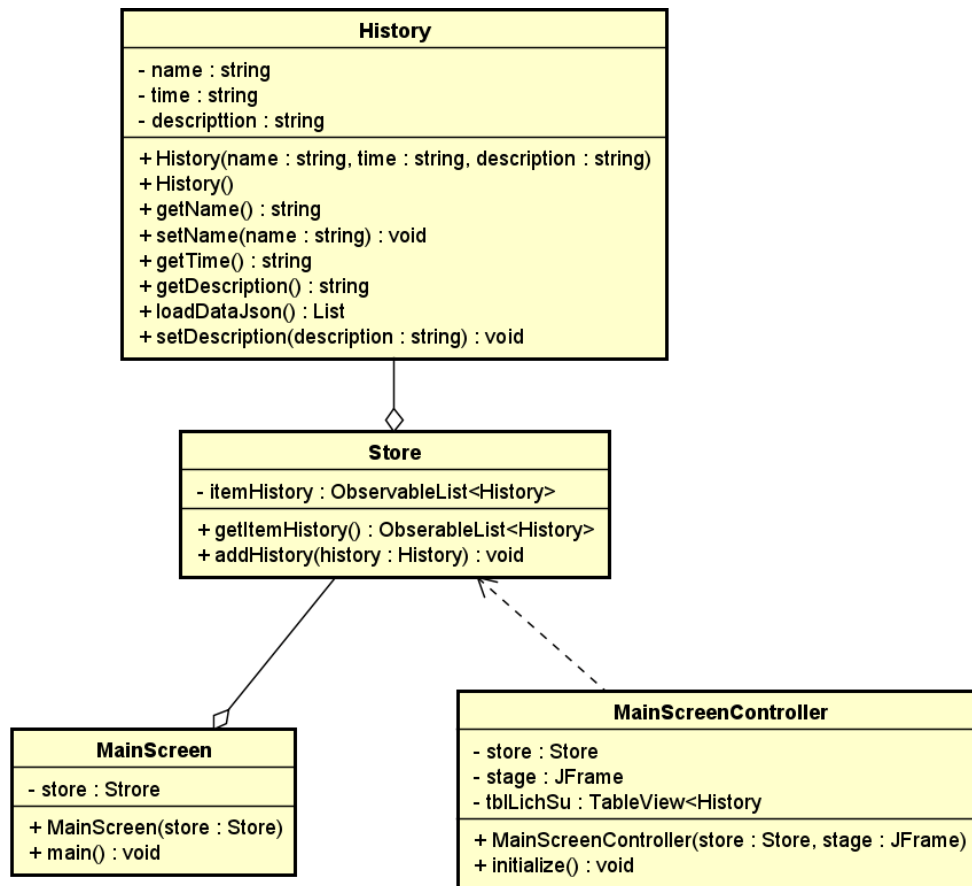
-3 lớp KingCharacterCrawler, NgươiKeSuCharacterCrawler, VanSuCharacterCrawler .

- Kế thừa từ lớp CharacterCrawler, dùng để đọc dữ liệu từ website. Lớp DataAggregation dùng để gộp các đối tượng cùng nhau lấy từ website về một đối tượng duy nhất.





-Lớp King và lớp Character kế thừa 3 thuộc tính từ lớp trừu tượng History.  
 Và 2 lớp này dùng để tạo đối tượng để lưu trữ các thuộc tính từ 3 lớp trên .



- Lớp Store dùng để lưu trữ các đối tượng History, Character,...
- Lớp MainScreen dùng để hiển thị các dữ liệu mình cần tìm kiếm.
- Lớp MainScreenController dùng để kiểm soát các sự kiện xảy ra
- Lớp CrawlerManager dùng để tổng hợp thông tin

## V. Kỹ thuật lập trình hướng đối tượng

### 1. Tính trừu tượng

- Sử dụng kỹ thuật lập trình bằng Interface (Giao diện) và Abstract Class (Lớp trừu tượng), cụ thể:

+ Lớp BaseWebCrawler là lớp trừu tượng: có phương thức trừu tượng (Abstract Method) là connect() để các lớp kế thừa sử dụng nó như một phương thức kết nối tới các trang web khác nhau mà không cần phải biết rõ nó là trang web nào, kết nối thành công sẽ tạo lấy được Document; phương thức trừu tượng thứ hai là start(), để các lớp con kế thừa và sử dụng cho việc chạy Crawl dữ liệu và sử dụng để gọi chung trong hàm Main

+ Lớp History là lớp trừu tượng chứa phương thức trừu tượng loadDataJson(), phương thức này được cài đặt ở các lớp con dùng để lấy dữ liệu từ file Json cho từng đối tượng; phương thức trừu tượng hienthi() để đưa ra dữ liệu chung nhất của 5 đối tượng là Name, Time và Description.

## **2. Tính đóng gói**

- Mỗi lớp đều có các thuộc tính private, đóng gói chúng lại vào class và chỉ được truy cập thông qua phương thức getter() và sửa đổi thông qua phương thức setter(). Việc đóng gói như vậy tránh những truy cập bất hợp pháp hoặc sẽ tiện lợi hơn khi sửa đổi code. Ví dụ như việc nếu để thuộc tính Name là public, khi muốn sửa đổi ta phải đi tìm tất cả những chỗ có thuộc tính này và sửa lại, trong khi đó với phương thức setter(), ta chỉ cần truyền vào Name mình muốn sửa.

## **3. Tính kế thừa**

- Tính kế thừa được thể hiện rõ nhất trong việc xây dựng class cho các thực thể Character, Dynasty, Festival... Vì chúng đều có chung thuộc tính đó là tên (Name), thời gian (Time) và mô tả (Description).

- Tính kế thừa cũng được thể hiện qua các class Crawler như CharacterCrawler, DynastyCrawler, FestivalCrawler... kế thừa từ class BaseWebCrawler với các thuộc tính url, doc do tất cả các class Crawler đều cần có url để kết nối và có Document để làm việc với file json.

- Các class Crawler nhỏ hơn của từng trang web cũng kế thừa từ class cha của chúng, ví dụ với Triều đại: DynastyCrawler là lớp cha có các phương thức connect() và start(), các lớp con như NguoikeSuDynastyCrawler hay WikiDynastyCrawler sẽ kế thừa các phương thức đó và thực thi vì chúng đều cần phải kết nối và ghi file json.

## **4. Tính đa hình**

- Tính đa hình có thể thấy thông qua việc ghi đè các phương thức từ abstract class với các cách cài đặt khác nhau. Ví dụ nhưng trong class History với phương thức trừu tượng cần ghi đè là loadDataJson(), thì với mỗi class con kế thừa nó như Dynasty, King, Event... ta đều phải ghi đè lại phương thức đó với các cách thực thi khác nhau vì mỗi file json ghi ra của từng đối tượng là khác nhau. Hoặc như việc ghi đè các phương thức

connect() và start() từ BaseWebCrawler với các cách cài đặt khác nhau ở lớp con do việc kết nối tới các url để crawl dữ liệu rất khác nhau, mỗi lớp sẽ tìm trên một đường link và việc kết nối cũng tương tự như vậy.

## **VI. Các công nghệ sử dụng**

- Maven: Các lớp thư viện được thêm qua maven giúp đồng bộ code (Tránh conflict các phiên bản, hay IDE)
- Java Coding Convention: các Class, package, tên biến, hàm được đặt tên theo tiêu chuẩn
- Xây dựng biểu đồ Class, package bằng ngôn ngữ mô hình hóa thống nhất UML, phần mềm astar
- Jsoup: Thư viện chuyên dụng cho việc thu thập dữ liệu qua các trang web, thêm qua Maven
- Xây dựng giao diện GUI với javafx Scene Builder
- Gson: Thư viện của Google cho Java để ghi hoặc đọc dữ liệu với file .json ra các Object;
- Git & Github: Quản lý mã nguồn qua Github, xem mã nguồn [tại đây](#)

## **VII. Hướng dẫn sử dụng**

- Các bước thực hiện lấy dữ liệu trên web và trả ra màn hình
  - Bước 1: Tìm đến class Main – đây là class thực hiện việc lấy dữ liệu và kết quả ra màn hình của cả dự án
  - Bước 2: Chạy hàm main và lựa chọn hành động
    - + Nếu chọn “Lấy dữ liệu” hệ thống sẽ chạy phần lấy dữ liệu từ trên các web về và lưu vào file json
    - + Nếu chọn “Hiển thị dữ liệu” hệ thống sẽ chạy phần hiển thị dữ liệu và tìm kiếm

## **VIII. [Video Demo](#)**