

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



BÁO CÁO BÀI TẬP LỚN MÔN HỌC
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Đề tài:

Lịch sử Việt Nam

Giảng viên hướng dẫn : GV. Trịnh Tuấn Đạt
Nhóm sinh viên thực hiện : Nhóm 7
Thành viên : Nguyễn Khắc Thái Bình - 20204944
Phạm Bá Đồng - 20200156
Nguyễn Anh Đức - 20200167
Nguyễn Sơn Đức - 20204952
Trần Văn Hiếu - 20200231
Chu Bá Hiếu - 20205080

Lời mở đầu

Ngày nay, việc ứng dụng công nghệ thông tin đã trở nên phổ biến trong hầu hết mọi cơ quan, doanh nghiệp, trường học đặc biệt là việc áp dụng các giải pháp tin học trong công tác quản lý hoặc để xây dựng những hệ thống phục vụ cho một mục đích nào đó.

Trong công nghệ thông tin thì không thể không nhắc đến lập trình hướng đối tượng - một môn học rất quan trọng mà các kỹ sư công nghệ thông tin cần phải nắm rõ. Trong bài báo cáo này chúng em sẽ trình bày về bài tập lớn môn lập trình hướng đối tượng.

Chúng em xin gửi lời cảm ơn đến thầy Trịnh Tuấn Đạt đã truyền đạt cho chúng em nhiều kiến thức về môn lập trình hướng đối tượng cũng như nhiệt tình chỉ dẫn và có những gợi ý để hoàn thành đồ án môn học.

Tuy có nhiều cố gắng nhưng trong quá trình làm đồ án không tránh khỏi những sai sót. Chúng em rất mong nhận được các ý kiến đóng góp từ các thầy cô và bạn bè để chúng em rút kinh nghiệm thực hiện tốt hơn ở những lần sau.

Phân chia công việc

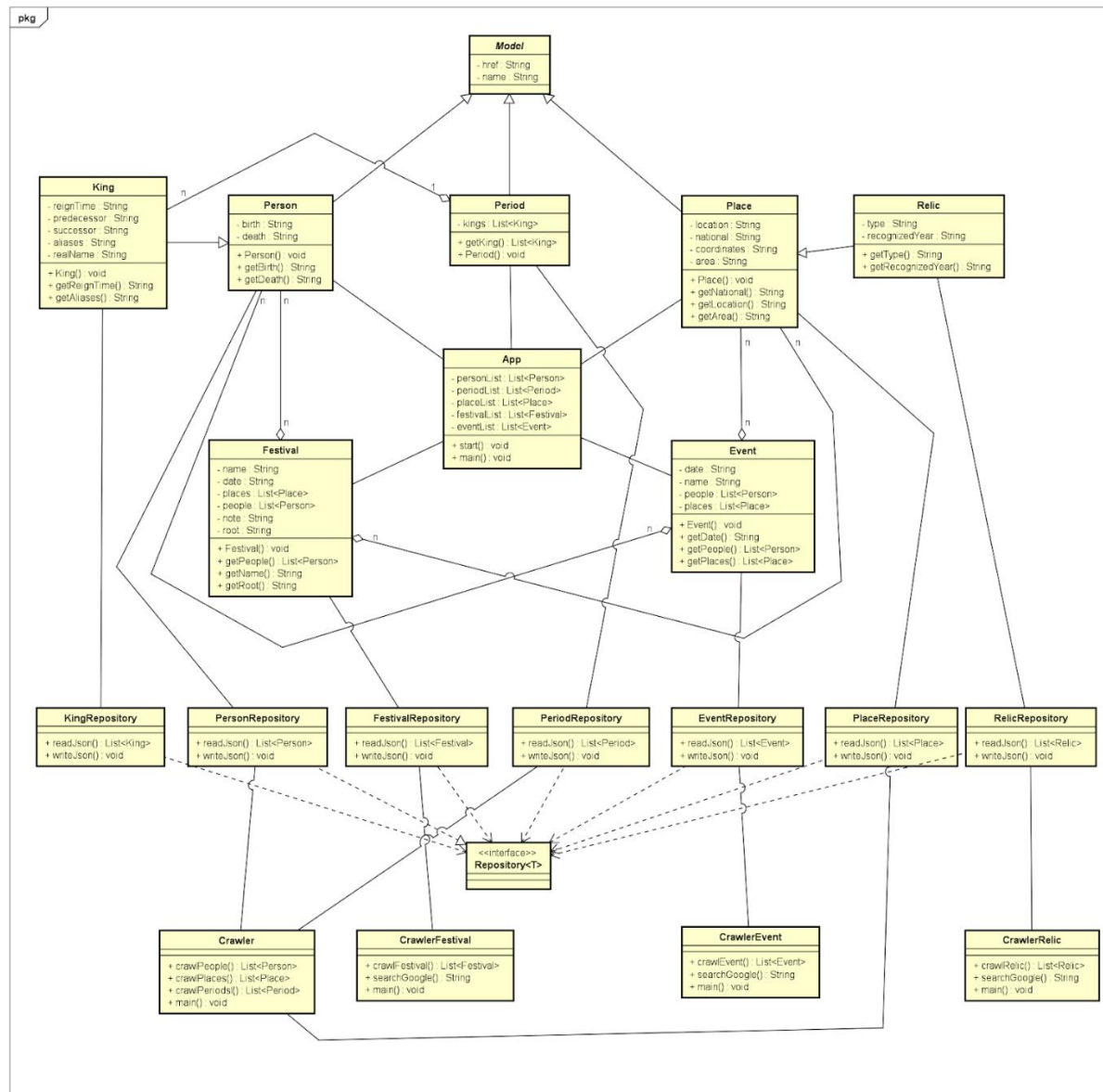
Họ và tên	Công việc	Phần trăm
Nguyễn Khắc Thái Bình	Controller , UML	16%
Phạm Bá Đồng	Lấy dữ liệu , Báo cáo	18%
Nguyễn Anh Đức	Controller , UML	16%
Nguyễn Sơn Đức	Lấy dữ liệu , Repository	18%
Trần Văn Hiếu	Giao diện, slide	16%
Chu Bá Hiếu	Báo cáo , Repository	16%

MỤC LỤC

1. Class Diagram và Package Diagram	5
1.1)Class Diagram	5
1.2)Package Diagram	6
2. Mô tả thiết kế	7
2.1)Tính đóng gói	7
2.2)Tính kế thừa	7
2.3)Tính đa hình.....	7
2.4)Tính trừu tượng.....	8
3. Crawler	8
3.1)Crawler	8
3.2)CrawlerEvent	8
3.3)CrawlerFestival	8
3.4)CrawlerRelic.....	9
3.5)Class abstract CrawlGoogle	9
4. Controller and GUI.....	9
4.1)Menu	9
4.2)Event	9
4.3)Festival	12
4.4)People	13
4.5)Period	15
4.6)Place	16
5. Một số công nghệ sử dụng.....	18
5.1)Một số thư viện đã sử dụng.....	18
5.2)Một số kỹ thuật hay ho	18

1. Class Diagram và Package Diagram

1.1)Class Diagram

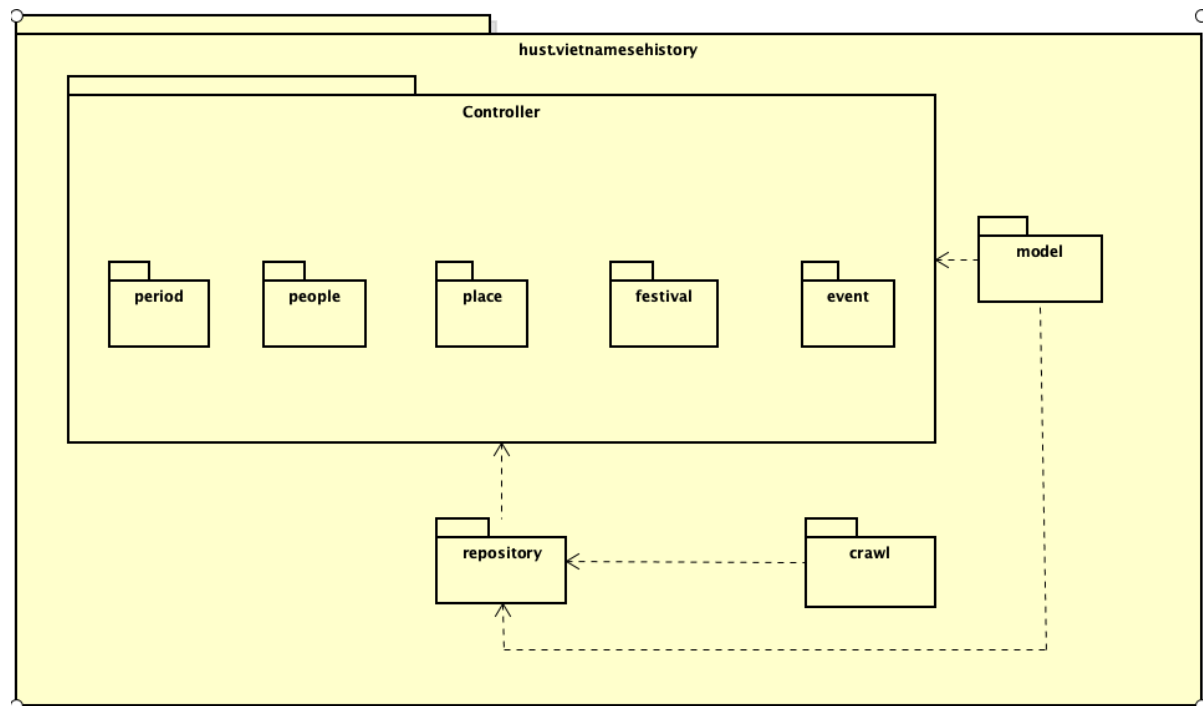


Biểu đồ sơ đồ lớp trên gồm những thành phần lớp sau:

- **Model:** là lớp trừu tượng được kế thừa bởi các lớp person, period và place. Gồm có 2 thuộc tính là name(tên) và href(liên kết)
- **Person:** là đối tượng người, trong đó gồm các thuộc tính birth(năm sinh), death(năm mất) và các constructor và getter setter tương ứng

- **King:** là đối tượng người đặc biệt là vua, kế thừa của lớp person, trong đó gồm các thuộc tính reignTime(thời gian trị vì), predecessor(tiền nhiệm), successor(kế nhiệm), aliases(tên hiệu), realName(tên thật) và các constructor và getter setter tương ứng
- **Period:** là đối tượng thời kỳ, trong đó gồm các thuộc tính kings(danh sách các vua liên quan) và các constructor và getter setter tương ứng
- **Place:** là đối tượng địa điểm, trong đó gồm các thuộc tính national(quốc gia), location(vị trí), coordinates(tọa độ), area(diện tích) các constructor và getter setter tương ứng
- **Relic:** là đối tượng địa danh, kế thừa của lớp place trong đó gồm các thuộc tính type(kiểu địa danh), RecognizedYear(năm phát hiện), các constructor và getter setter tương ứng
- **Festival:** là đối tượng lễ hội, trong đó gồm các thuộc tính name(tên), date(ngày), places(các địa điểm liên quan), people(các nhân vật liên quan), note(ghi chú) và root(nguồn gốc) và các constructor và getter setter tương ứng
- **Event:** là đối tượng sự kiện trong đó gồm các thuộc tính name(tên), date(ngày), places(các địa điểm liên quan), people(các nhân vật liên quan) và các constructor và getter setter tương ứng
- **App:** là chương trình chạy chính của bài

1.2) Package Diagram



Biểu đồ sơ đồ gói trên gồm những thành phần gói sau:

- **model**: chứa các đối tượng **period**, **people**, **place**, **festival**, **event**, **king**, **relic** và lớp trừu tượng **model**.
- **crawler**: chứa các methods để lấy dữ liệu trên các đường dẫn được nêu ở phần crawler phía dưới.
- **repository**: chứa các methods để đọc và ghi dữ liệu dưới dạng json
- **controller**: Bao gồm các file để điều khiển các file fxml (có 5 package nằm trong controller là **period**, **people**, **place**, **festival**, **event** chứa các file controller của từng màn)

Việc chia dự án làm 4 phần như này sẽ thuận tiện chia việc, bởi vì các phần **crawler**, **repository** và **controller** không cần phải làm việc chung với nhau vì đây là những đầu việc riêng biệt.

2. Mô tả thiết kế

2.1) Tính đóng gói

Các dữ liệu liên quan đến từng thực thể được đưa vào lớp riêng, với thuộc tính **private**, và chỉ được thay đổi và sử dụng thông qua **getter**, **setter** và **constructor**. Việc áp dụng tính đóng gói giúp tiện cho việc quản lý và sử dụng, hơn nữa còn đảm bảo an toàn dữ liệu cho các lớp.

2.2) Tính kế thừa

Ở package **model** lớp **King** được kế thừa lớp **Person**, lớp **Relic** được kế thừa lớp **Place** vì lớp **King**, **Relic** đều được xây dựng dựa trên định nghĩa lớp **Person**, **Place**, thừa hưởng các thuộc tính, phương thức của **Person**, **Place** và ngoài ra còn bổ sung thêm các thành phần mới.

Việc này giúp tái sử dụng mã nguồn một cách tối ưu, tận dụng được mã nguồn, hỗ trợ việc bảo trì code đơn giản hơn.

2.3) Tính đa hình

Ở package **repository**, các lớp repository của các thực thể đều được implement từ interface **Repository**. Các lớp repository này dùng để đọc và ghi dữ liệu của thực

thể từ file json, do đó chúng đều có 2 phương thức là readJson và writeJson, nhưng với mỗi thực thể khác nhau thì việc đọc và ghi cũng sẽ khác nhau. Ở các package controller thì có sử dụng interface Iniztializable để lấy dữ liệu và hiển thị vào các bảng

Việc áp dụng tính đa hình giúp không phải giảm bớt sự phức tạp trong quá trình thiết kế và phát triển.

2.4) Tính trừu tượng

Ở package **model**, lớp trừu tượng **Model** là lớp có các thuộc tính và phương thức được dùng chung có các lớp **Period**, **Person**, **Place**, **King**, **Relic**.

Ở package **crawler**, lớp trừu tượng **CrawlerGoogle** có phương thức searchGoogle() dùng chung cho các lớp **CrawlerEvent** và **CrawlerFestival**.

Việc áp dụng tính trừu tượng giúp cho giảm bớt sự phức tạp trong quá trình thiết kế và phát triển.

3. Crawler

3.1)Crawler

Trong lớp Crawler thì nhóm chúng em sử dụng dữ liệu ở trang <https://nguoikesu.com/> để lấy thông tin về các nhân vật lịch sử, địa danh và các thời kỳ lịch sử của Việt Nam bằng cách sử dụng thư viện jsoup, tổng cộng có **1452** nhân vật lịch sử, **769** địa danh và **35** thời kỳ lịch sử được lưu vào các file people.json, places.json và periods.json .

3.2)CrawlerEvent

Trong phần crawl event, nhóm chúng em sử dụng dữ liệu ở đường dẫn : [Niên biểu lịch sử Việt Nam-Wikipedia Tiếng Việt](#) được **179** sự kiện lịch sử của Việt Nam bằng cách sử dụng thư viện jsoup . Các sự kiện được liên kết với những người và địa điểm liên quan đến sự kiện đó bằng methods searchGoogle so với dữ liệu đã lấy được ở people.json và places.json .

3.3)CrawlerFestival

Trong phần crawl festival , nhóm chúng em đã sử dụng dữ liệu ở đường dẫn:

[Lễ hội Việt Nam – Wikipedia tiếng Việt](#) được **52** lễ hội của Việt Nam bằng cách sử dụng thư viện jsoup . Các lễ hội được liên kết với những địa điểm và người liên quan đến địa điểm đó thông qua methods searchGoogle so với dữ liệu đã lấy được ở people.json và places.json .

3.4)CrawlerRelic

Trong phần crawl Relic, nhóm chúng em đã sử dụng dữ liệu ở đường dẫn : [Danh sách Di tích quốc gia Việt Nam – Wikipedia tiếng Việt](#) được **1599** di tích quốc gia ở Việt Nam bằng cách sử dụng thư viện jsoup . Các di tích được kế thừa từ các địa danh đã được crawl ở trên (hiện tại bọn em vẫn chưa map được với lớp place)

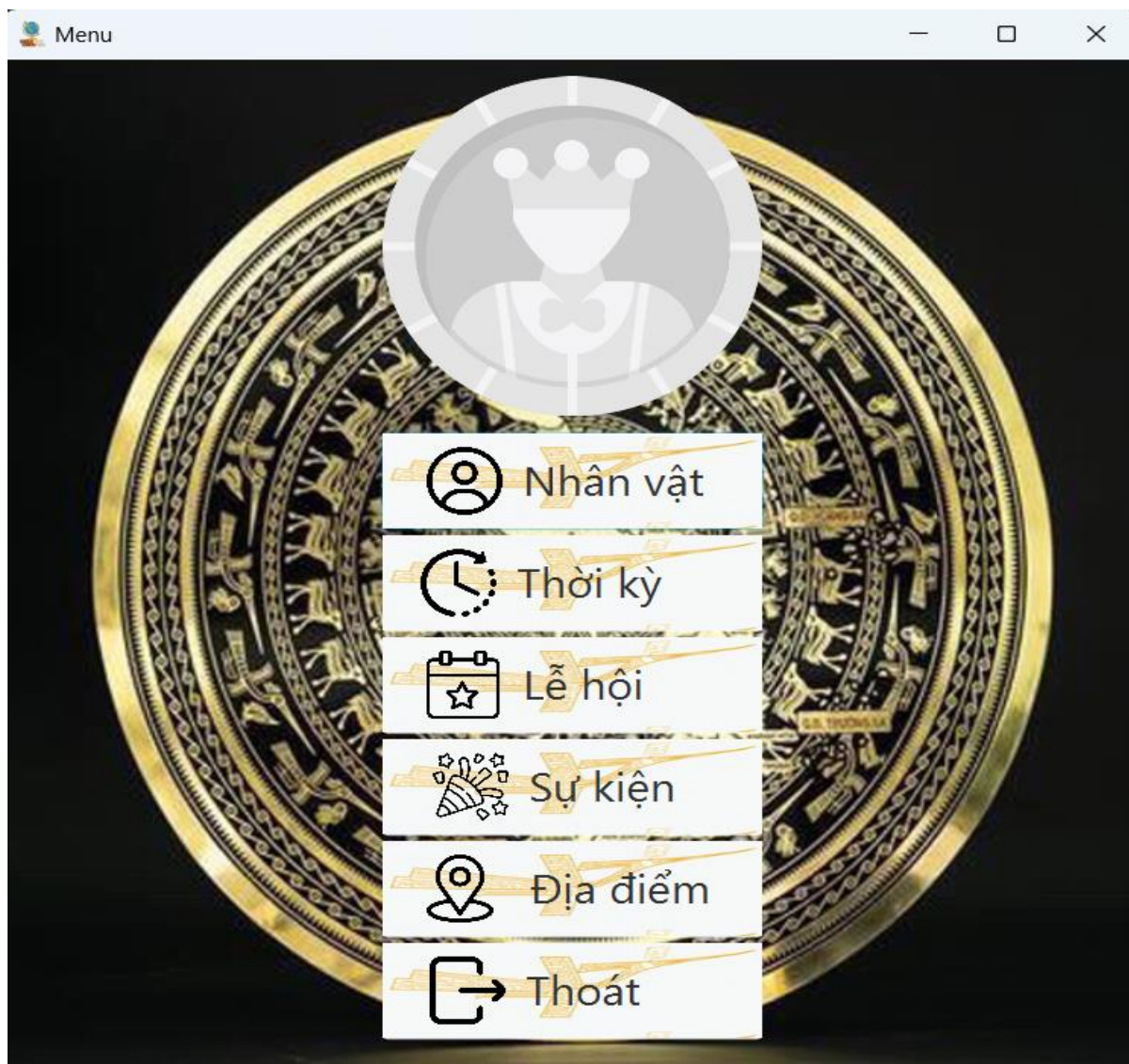
3.5)Class abstract CrawlGoogle

Method searchGoogle được sử dụng ở crawl Festival và Event thì được viết trong class abstract crawlGoogle và được kế thừa bởi 2 class là CrawlEvent và CrawlGoogle .

Method searchGoogle sử dụng thư viện Jsoup và tận dụng công cụ tìm kiếm Google để map nhân vật và địa danh đã được lấy ở people.json và places.json lại với các festivals và events , searchGoogle có kiểu trả về là một string có dạng “/dia-danh/...” hoặc “/nhan-vat/...” để có thể liên kết với model person và place hoặc trả về chuỗi rỗng “” nếu như không có nhân vật hoặc địa danh đó trong trang web <https://nguoiikesu.com/>.

4. Controller and GUI

4.1)Menu



Giao diện Menu

Thao tác với Menu:

Nhân vật => hiển thị màn giao diện **People**

Thời kỳ => hiển thị màn giao diện **Period**

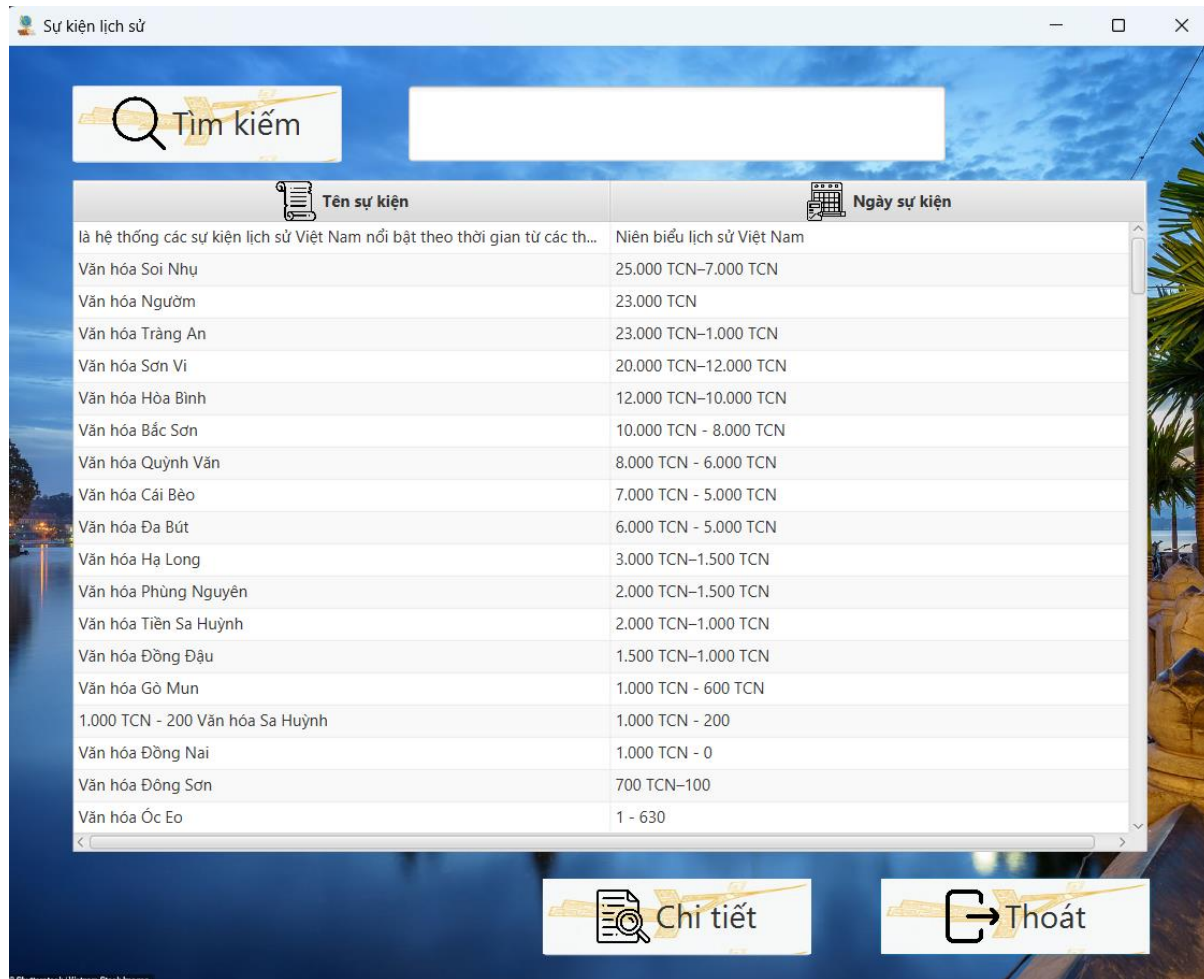
Lễ hội => hiển thị màn giao diện **Festival**

Sự kiện => hiển thị màn giao diện **Event**

Địa điểm => hiển thị màn giao diện **Place**

Thoát => kết thúc chương trình

4.2)Event



Giao diện Event

Thao tác và hiển thị trong giao diện Event:

Ô trống để nhập thông tin cần tìm kiếm (có phân biệt hoa thường)

Tìm kiếm => tìm những từ khóa có chứa thông tin đã nhập trong ô trống

Bảng => hiển thị các cột chứa thuộc tính của **Event**

Chi tiết => chọn một **Event** để hiển thị thêm thông tin của **Event** đó

Thoát => hiển thị giao diện màn **Menu**



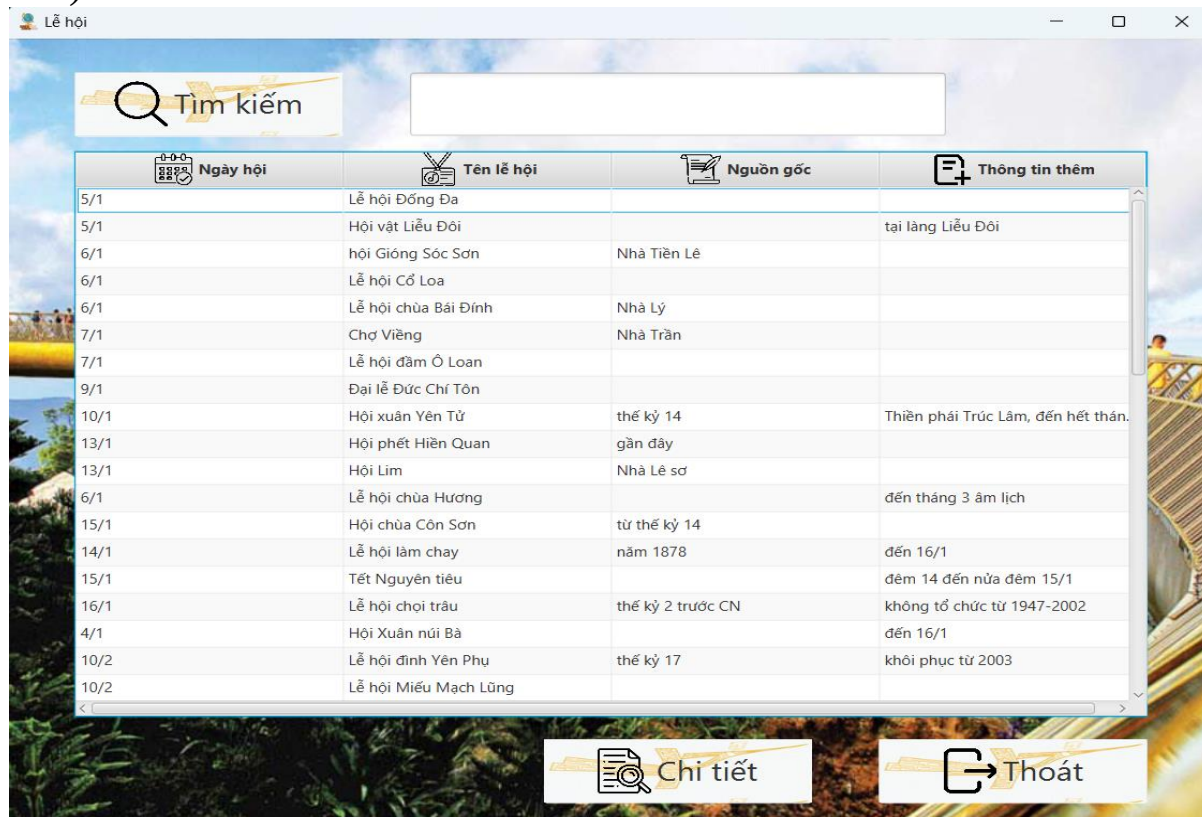
Giao diện EventDetail

Thao tác và hiển thị trong giao diện EventDetail:

Giao diện **hiển thị đầy đủ** các thuộc tính của **Event** đã chọn ở giao diện màn **Event**

Thoát => hiển thị giao diện màn **Event**

4.3) Festival



Giao diện Festival

Thao tác và hiển thị trong giao diện Festival:

Ô trống để nhập thông tin cần tìm kiếm (có phân biệt hoa thường)

Tìm kiếm => tìm những từ khóa có chứa thông tin đã nhập trong ô trống

Bảng => hiển thị các cột chứa thuộc tính của **Festival**

Chi tiết => chọn một **Festival** để hiển thị thêm thông tin của **Festival** đó

Thoát => hiển thị giao diện màn **Menu**



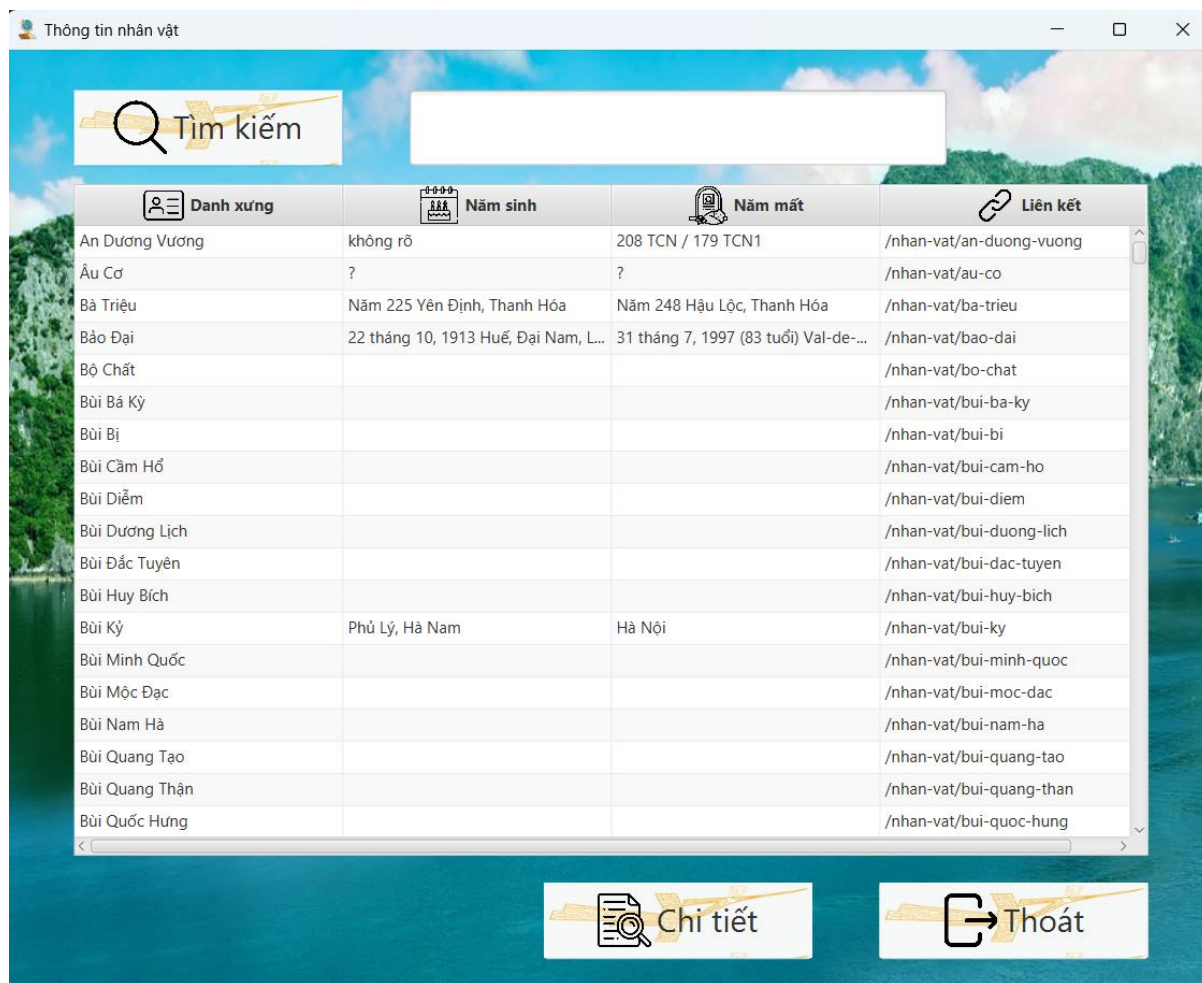
Giao diện FestivalDetail

Thao tác và hiển thị trong giao diện FestivalDetail:

Giao diện **hiển thị đầy đủ** các thuộc tính của **Festival** đã chọn ở giao diện màn **Festival**

Thoát => hiển thị giao diện màn **Festival**

4.4) People



Giao diện People

Thao tác và hiển thị trong giao diện People:

Ô trống để nhập thông tin cần tìm kiếm (có phân biệt hoa thường)

Tìm kiếm => tìm những từ khóa có chứa thông tin đã nhập trong ô trống

Bảng => hiển thị các cột chứa thuộc tính của **People**

Chi tiết => chọn một **People** để hiển thị thêm thông tin của **People** đó

Thoát => hiển thị giao diện màn **Menu**



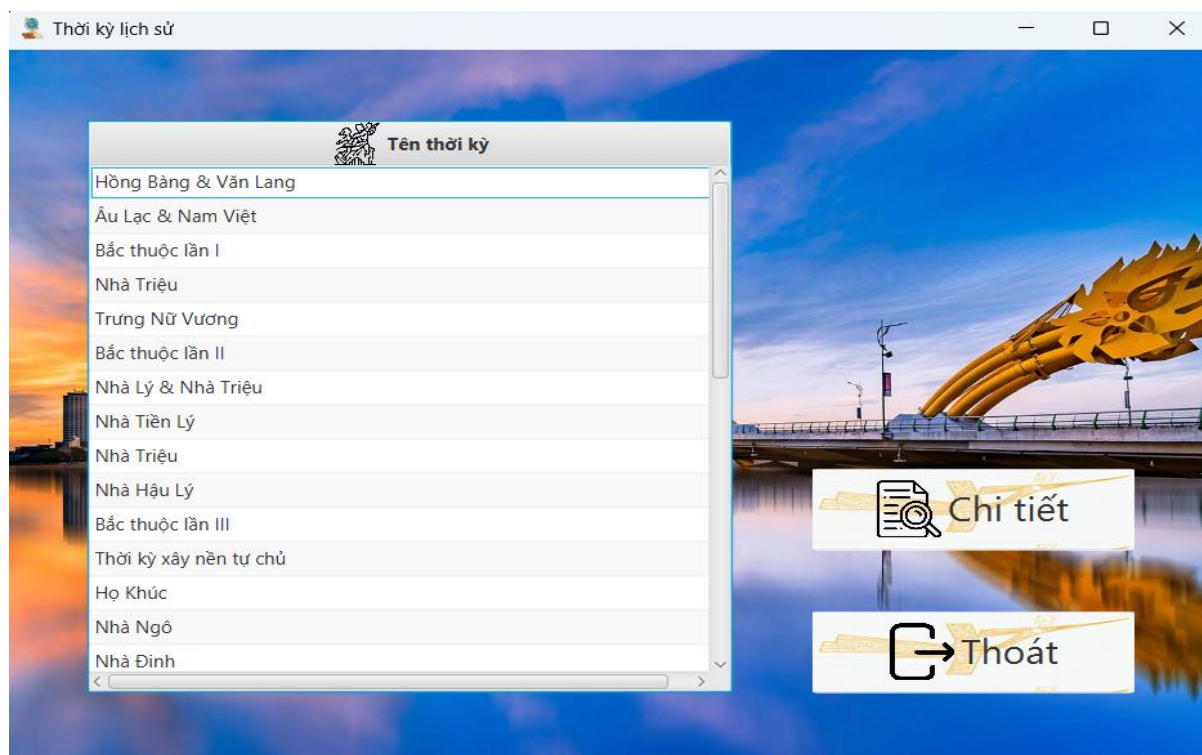
Giao diện PeopleDetail

Thao tác và hiển thị trong giao diện PeopleDetail:

Giao diện **hiển thị đầy đủ** các thuộc tính của **People** đã chọn ở giao diện màn **People**

Thoát => hiển thị giao diện màn **People**

4.5)Period



Giao diện Period

Thao tác và hiển thị trong giao diện Period:

Bảng => hiển thị các cột chứa thuộc tính của **Period**

Chi tiết => chọn một **Period** để hiển thị thêm thông tin của **Period** đó

Thoát => hiển thị giao diện màn Menu

[illegible]

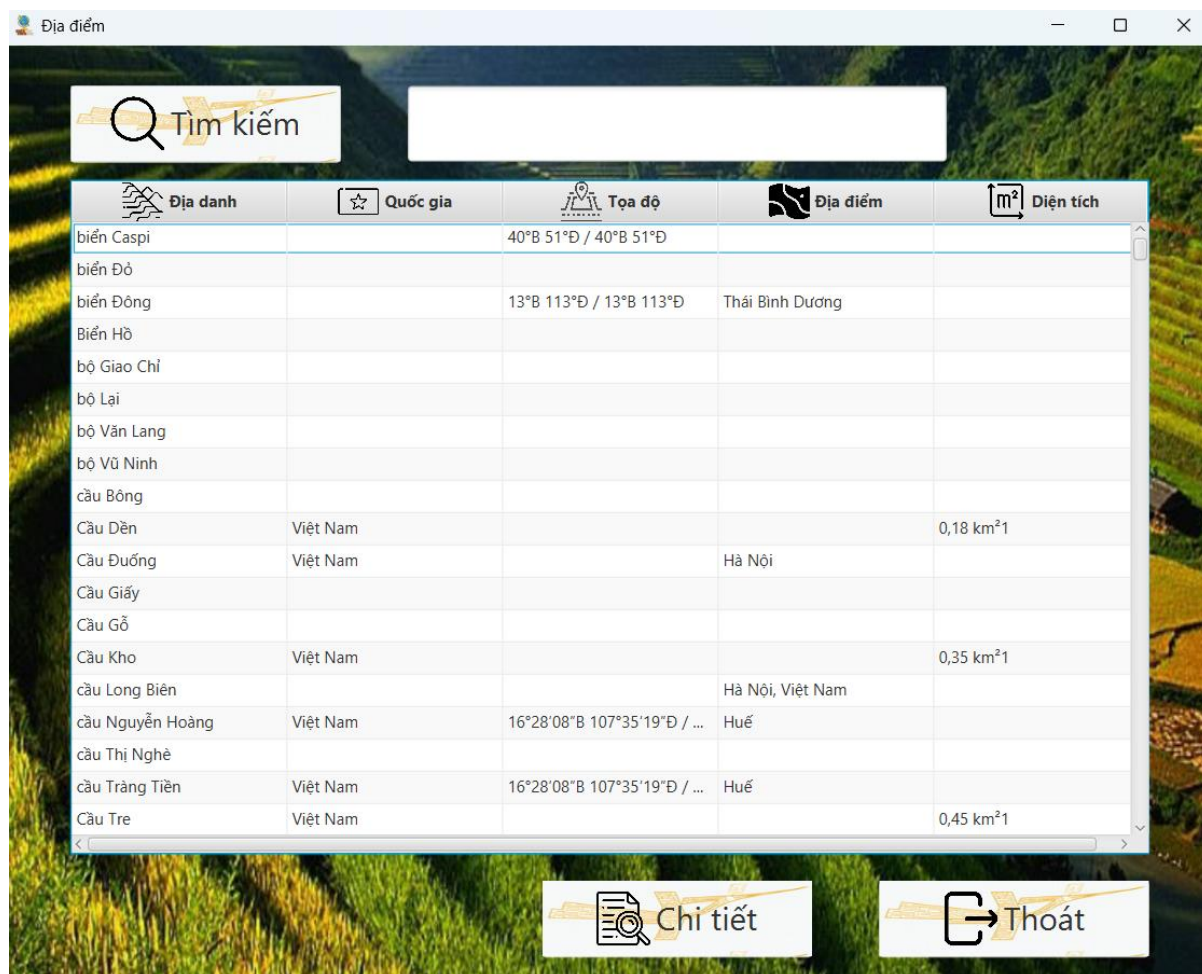
Giao diện PeriodDetail

Thao tác và hiển thị trong giao diện PeriodDetail:

Giao diện **hiển thị đầy đủ** các thuộc tính của **Period** đã chọn ở giao diện màn **Period**

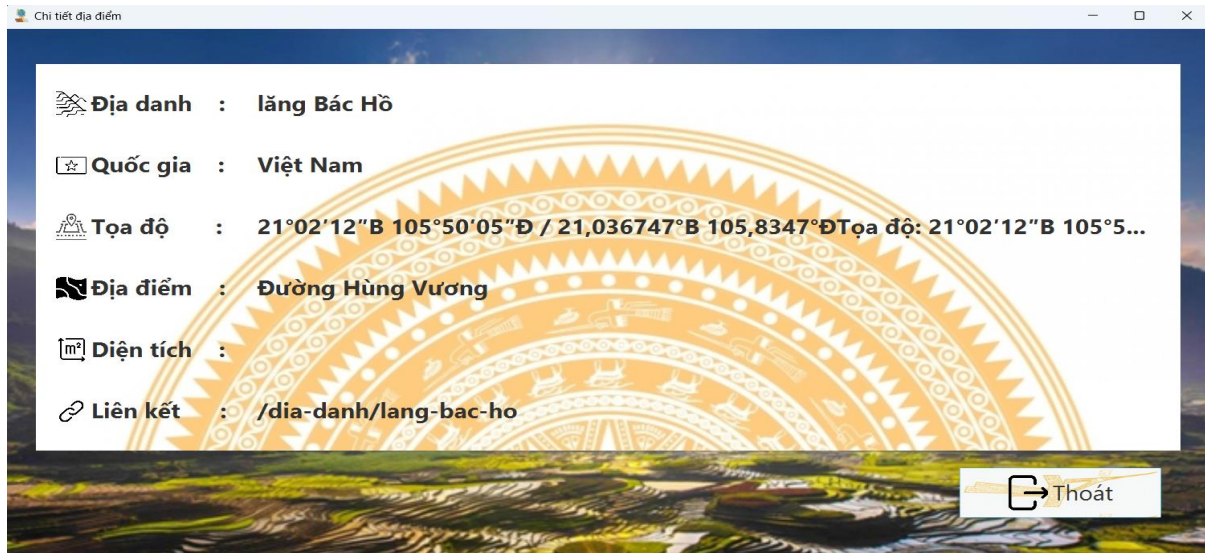
Thoát => hiển thị giao diện màn Period

4.6)Place



Giao diện Place

Thao tác và hiển thị trong giao diện Place:**Ô trống** để nhập thông tin cần tìm kiếm (có phân biệt hoa thường)**Tìm kiếm** => tìm những từ khóa có chứa thông tin đã nhập trong ô trống**Bảng** => hiển thị các cột chứa thuộc tính của **Place****Chi tiết** => chọn một **Place** để hiển thị thêm thông tin của **Place** đó**Thoát** => hiển thị giao diện màn **Menu**



Giao diện PlaceDetail

Thao tác và hiển thị trong giao diện PlaceDetail:

Giao diện **hiển thị** đầy đủ các thuộc tính của **Place** đã chọn ở giao diện màn **Place**

Thoát => hiển thị giao diện màn **Place**

5. Một số công nghệ sử dụng

5.1) Một số thư viện đã sử dụng

- Sử dụng thư viện jsoup để lấy dữ liệu từ các trang web về .
- Sử dụng thư viện json và jackson để in và đọc các file json .
- Sử dụng thư viện fxml để hiển thị chương trình chạy .

5.2) Một số kỹ thuật hay ho

- Sử dụng mô hình MVC và thêm phân repository (có nhiệm vụ đọc và ghi json) để dễ dàng làm việc, dễ mở rộng dự án và quản lý dự án .
- Sử dụng công cụ Google để liên kết dữ liệu với nhau .
- Sử dụng tính chất SOLID để thiết kế (chủ yếu là tính chất S và O).