

---

**Nhóm 14**

---

**Web đặt vé xem phim**  
**Software Architecture Document**

**Version 1.0**

<b>Web đặt vé xem phim</b>	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

## Revision History

Date	Version	Description	Author
28/Nov/22	1.0	Tài liệu được viết lần đầu.	Trần Anh Tuấn

<b>Web đặt vé xem phim</b>	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

## Table of Contents

1.	Introduction	4
2.	Architectural Goals and Constraints	4
3.	Use-Case Model	4
4.	Logical View	5
4.1	Class Diagram overview	6
4.2	Component: Backend – Service	6
4.2.1	Class Diagram:	6
4.2.2	Detail:	6
4.3	Component: Backend - Controller	7
4.3.1	Class Diagram:	7
4.3.2	Detail:	7
4.4	Component: Backend - Route	7
4.4.1	Class Diagram:	7
4.4.2	Detail:	7
4.5	Component: Frontend - Services	7
4.5.1	Class Diagram:	7
4.5.2	Detail	7
4.6	Component: Frontend - Controller and UI	8
4.6.1	Class Diagram:	8
4.6.2	Detail:	8
4.7	Data Structure Class Diagram	9
5.	Deployment	10
6.	Implementation View	10

Web đặt vé xem phim	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

# Software Architecture Document

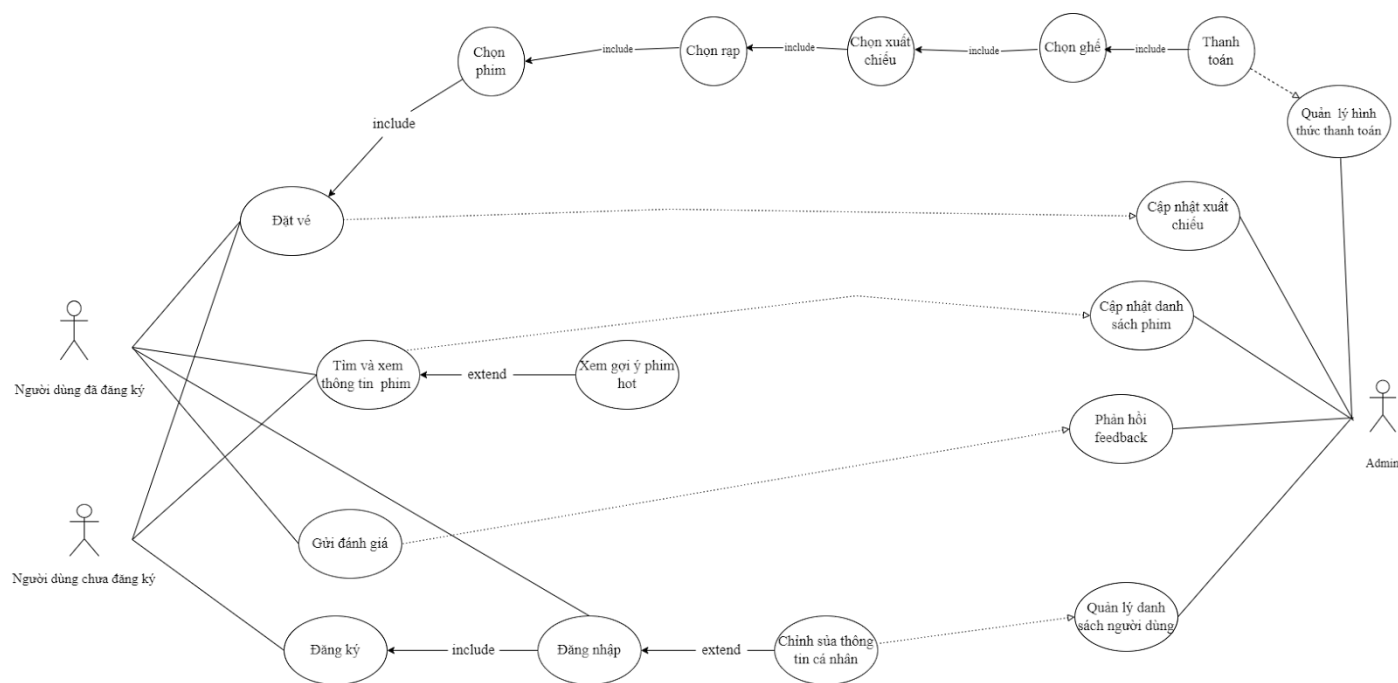
## 1. Introduction

- Tài liệu mô tả các component của ứng dụng và logical view dựa trên mô hình use-case được mô tả trong tài liệu Use-case specifications document và cũng chỉ ra cách triển khai ứng dụng trong tình huống thực tế.

## 2. Architectural Goals and Constraints

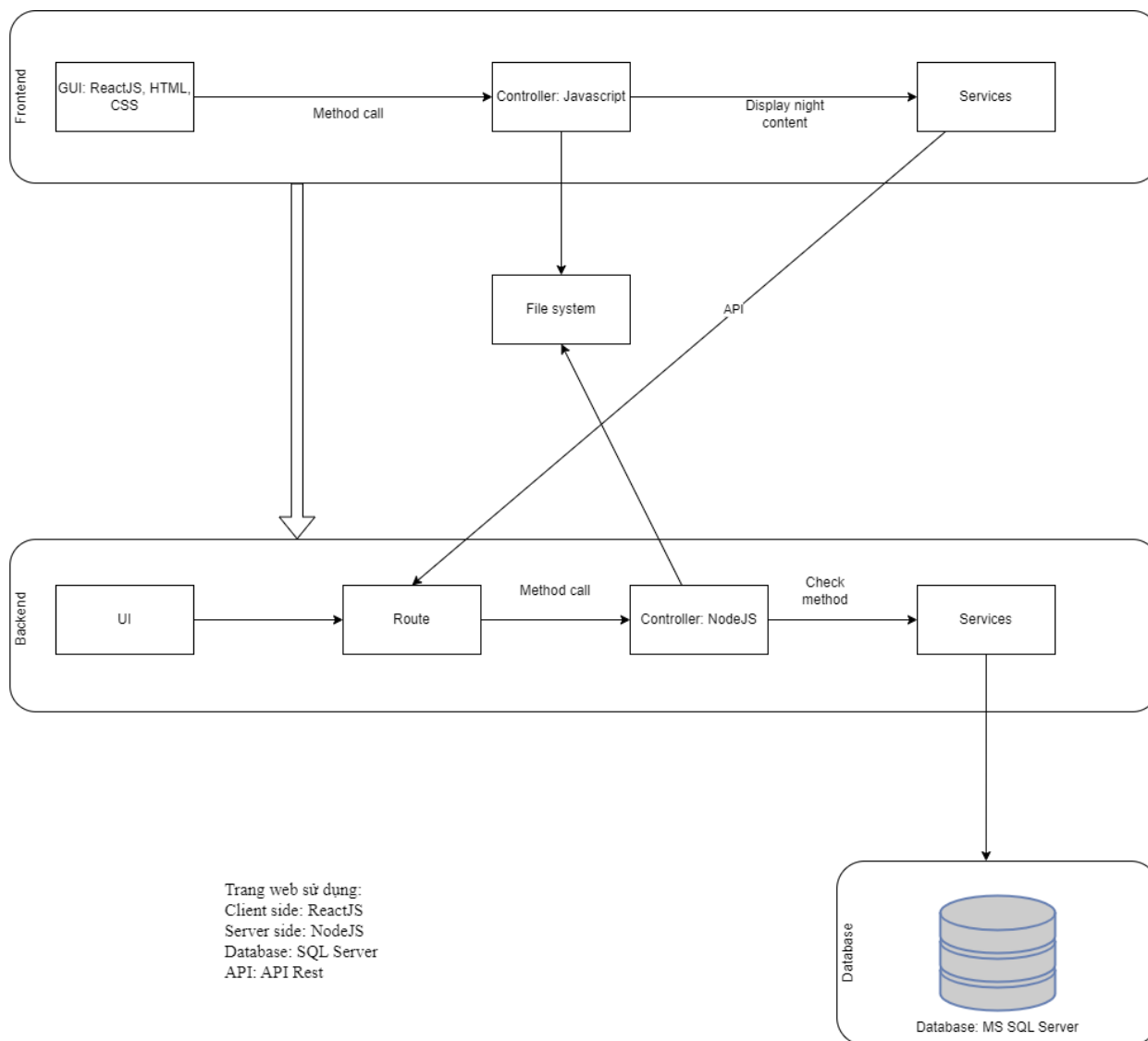
- **Quyền riêng tư và bảo mật thông tin của người dùng:** Tất cả thông tin cá nhân của người dùng như số điện thoại, email, số CCCD hoặc tài khoản ngân hàng đều được mã hóa. Chúng tôi chỉ phân tích các hành động của người dùng chẳng hạn như thống kê các phim được đặt vé nhiều trong tháng để đề xuất các phim hot nên xem cho người dùng.
- **Yêu cầu kết nối:** Người dùng chỉ cần một ứng dụng web như Google Chrome, Microsoft Edge hoặc Firefox để sử dụng trang web này.
- **Chiến lược phát triển:** Đầu tiên, nhóm sẽ thiết kế giao diện người dùng và các thành phần của nó. Trước khi xây dựng các tính năng khác, nhóm tập trung hoàn thiện chức năng hiển thị danh sách phim, cho phép người dùng chọn và đặt vé cũng như các chức năng liên quan - cơ sở dữ liệu chứa thông tin đặt vé, quản trị cơ sở dữ liệu kiểm soát.

## 3. Use-Case Model



Web đặt vé xem phim	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

## 4. Logical View



Nhóm sử dụng kiến trúc 3-tier cho project này:

Frontend: gồm 3 phần

- Phần GUI: Nhóm sử dụng các ngôn ngữ và frameworks để phát triển như: ReactJS, HTML, CSS. Layer này hiển thị giao diện người dùng và người dùng có thể thấy thanh điều hướng, tên phim, poster của phim,...
- Phần Controller: Dùng để điều khiển và chứa các hàm để gọi từ GUI Layer, sử dụng JavaScript để định nghĩa hàm và phát triển các phương thức
- Phần Services: Sử dụng API để giao tiếp với Back-end

Backend:

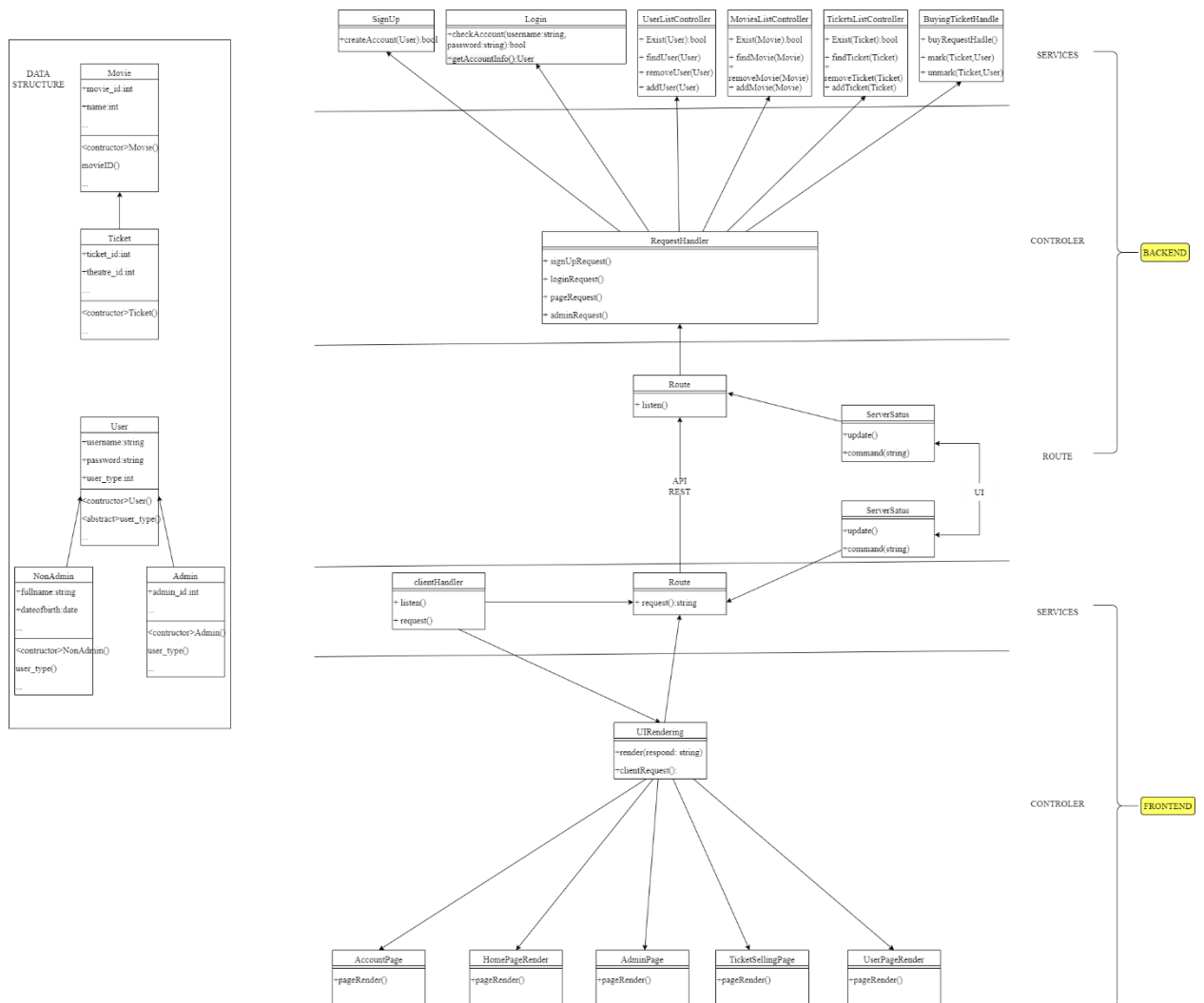
- UI: Sử dụng Command Line Interface.
- Route Layer: Phát hiện API chung để xử lý, điều phối, phân phối lời gọi đến các hàm khác, sử dụng JavaScript framework để tạo với API.

<b>Web đặt vé xem phim</b>	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

- Controller: điều khiển và gọi phương thức từ client đến Server và ngược lại, sử dụng NodeJS.
- Services: Truy vấn đến database và phản hồi tất cả yêu cầu từ Back-end.

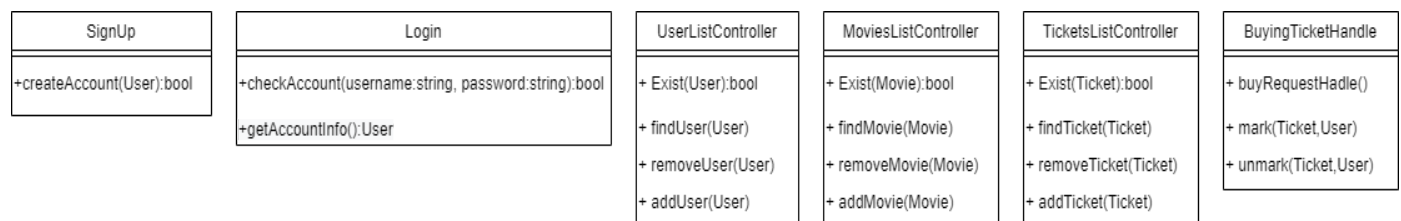
DataBase: Sử dụng SQL Server để truy vấn cơ sở dữ liệu.

#### 4.1 Class Diagram overview



#### 4.2 Component: Backend – Service

##### 4.2.1 Class Diagram:



<b>Web đặt vé xem phim</b>	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

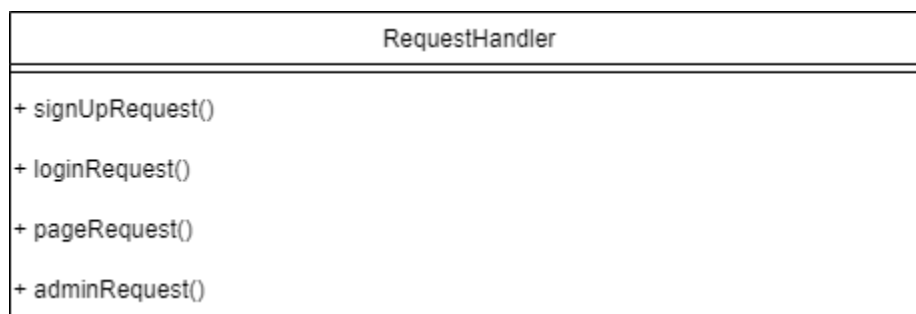
#### 4.2.2 Detail:

Backend service chủ yếu truy cập cơ sở dữ liệu cho các hoạt động từ giao diện người dùng, tất cả dịch vụ này sẽ được sử dụng bởi backend controllers.

- UserController: dùng để thêm hoặc xóa người dùng khỏi cơ sở dữ liệu.
- MovieListController: dùng để thêm bớt phim trong cơ sở dữ liệu.
- BuyingTicketHandle: dùng để đánh dấu và xử lý yêu cầu đặt vé phim.

### 4.3 Component: Backend - Controller

#### 4.3.1 Class Diagram:

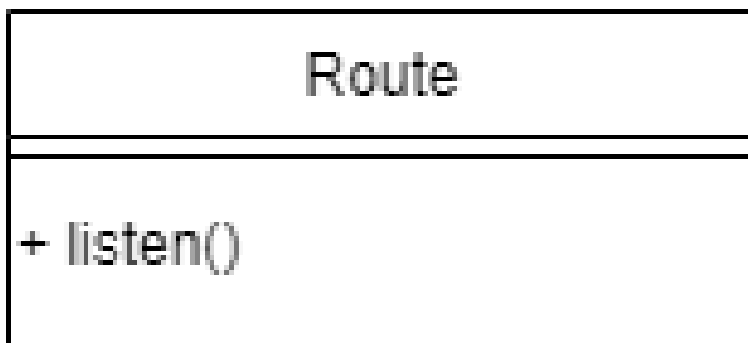


#### 4.3.2 Detail:

Component này sẽ xử lý yêu cầu đến từ Route component, đồng thời truy cập hệ thống tập tin nếu cần; mỗi phương thức trong RequestHandler sẽ xử lý một tác vụ cụ thể.

### 4.4 Component: Backend - Route

#### 4.4.1 Class Diagram:



#### 4.4.2 Detail:

Thành phần này sẽ xử lý các yêu cầu API đến từ Front-end và phân tích nó thành một lệnh hoạt động cho hệ thống Back-end để hoạt động. Khi một yêu cầu hoàn tất, Route sẽ phản hồi cho phía Front-end. Đây cũng sẽ xử lý bất kỳ lệnh nào mà đối tượng cần dùng

Web đặt vé xem phim	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

#### 4.5 Component: Frontend - Services

##### 4.5.1 Class Diagram:



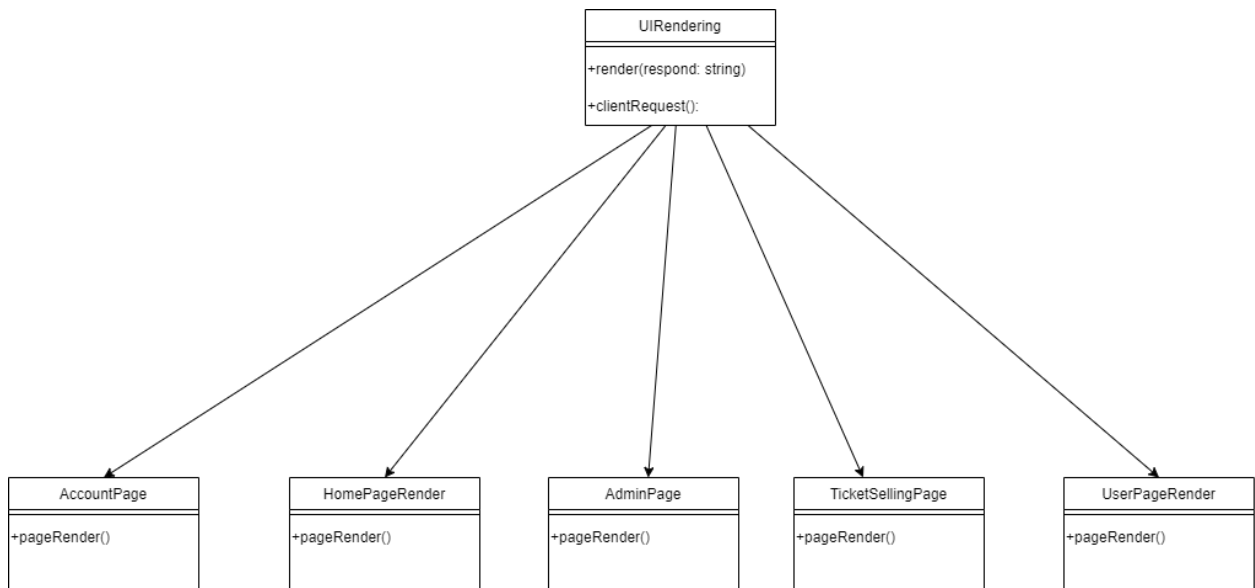
##### 4.5.2 Detail

Thành phần service sẽ xử lý mọi yêu cầu của người dùng và tương tác trực tiếp với Backend.

- clientHandler: nhận yêu cầu đến và phân tích nó thành renderUI hoặc Route cho yêu cầu.
- Route: sẽ gửi yêu cầu API về phía Backend và chờ phản hồi.

#### 4.6 Component: Frontend - Controller and UI

##### 4.6.1 Class Diagram:



##### 4.6.2 Detail:

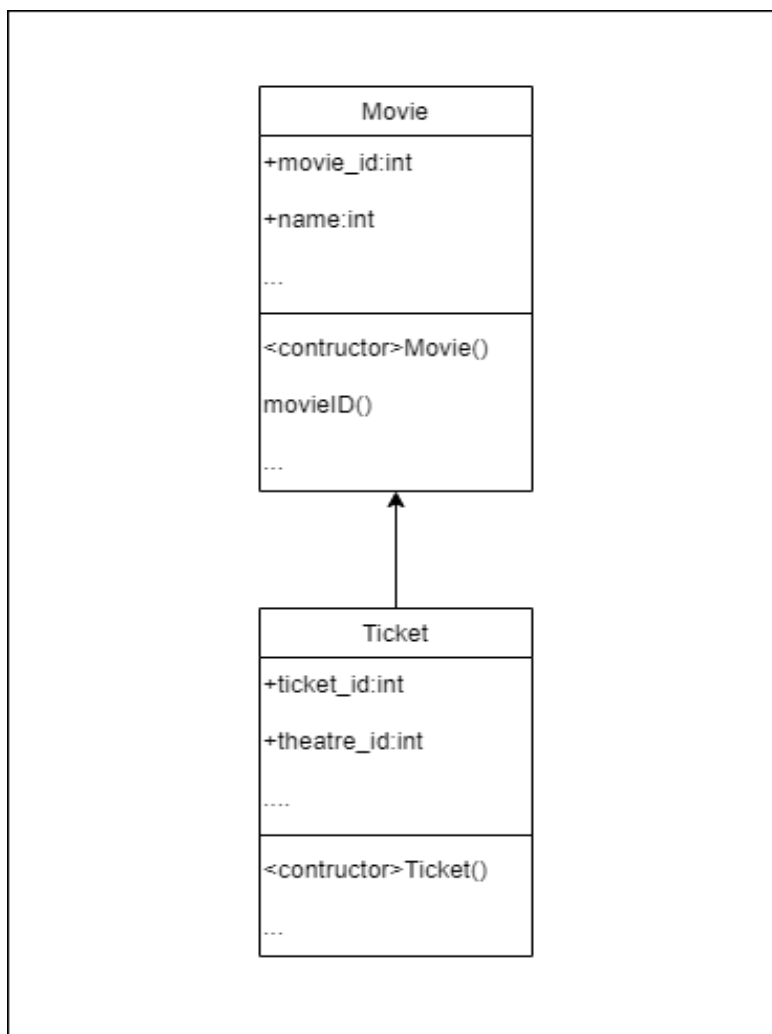
Controller sẽ hiển thị GUI cho người dùng bằng cách sử dụng yêu cầu của người dùng và dữ liệu nhận được từ phía phụ trợ

- UIRendering: xử lý phần GUI
- AccountPage: xử lý việc render trang tài khoản người dùng
- HomePageRender: xử lý việc render trang người dùng
- UserPageRender: xử lý việc render trang người dùng
- AdminPage: xử lý việc render trang admin
- TicketSellingRender: xử lý việc render trang bán vé

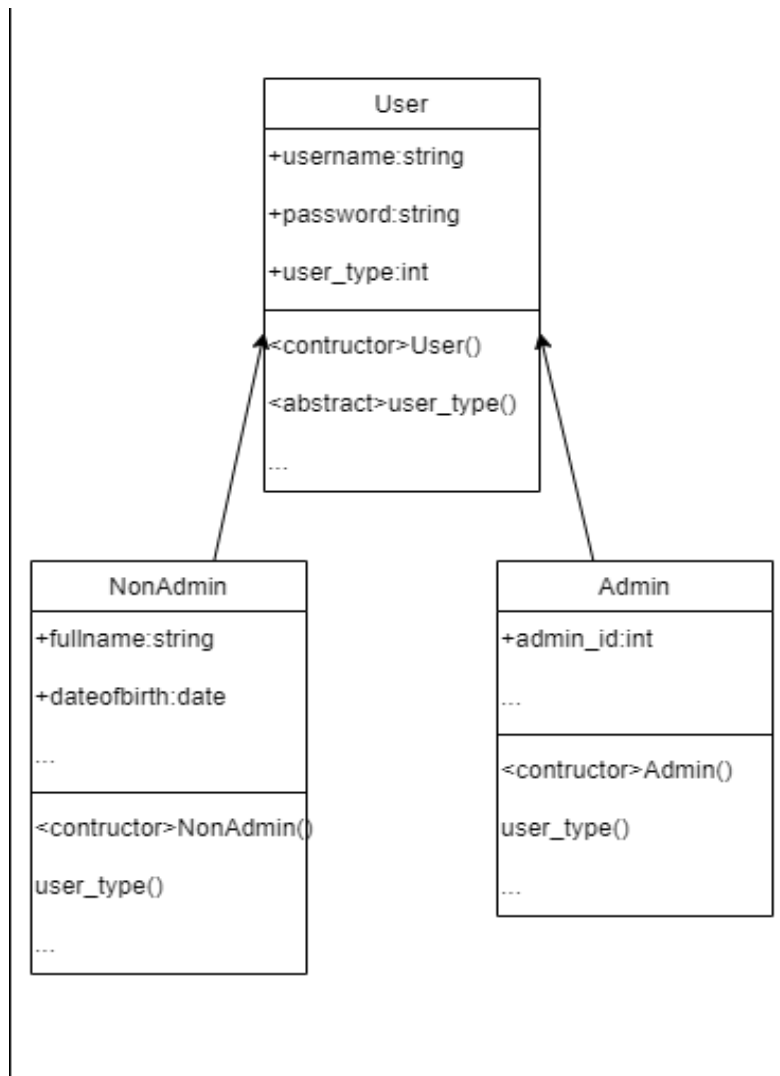


Web đặt vé xem phim	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	

#### 4.7 Data Structure Class Diagram



Web đặt vé xem phim	Version: 1.0
Software Architecture Document	Date: 28/Nov/22
SAD1.0	



## 5. Deployment

## 6. Implementation View