**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**KHOA/VIỆN ……………..…………..**

**---🙠**🕮**🙢---**



**TIỂU LUẬN**

**Đề tài:……………………………………………………………**

***Giáo viên hướng dẫn: ………………………...***

***Sinh viên thực hiện:……………………..............***

***Nhóm: …………………………………………***

***Lớp:……………………………………………***

***Năm học …… - ……***

Mục Lục

[Hệ thống quản lý bán vé xem phim trực tuyến 2](#_Toc183416794)

[1. Tìm hiểu tổng quan về bài toán 2](#_Toc183416795)

[1.1 Quy trình vận hành của hệ thống bán vé tại các rạp phim 2](#_Toc183416796)

[1.2 Vấn đề mà khách hàng và quản lý rạp chiếu gặp phải 2](#_Toc183416797)

[1.3 Nghiên cứu các hệ thống bán vé trực tuyến hiện có 2](#_Toc183416798)

[2. Tìm hiểu tổng quan về công nghệ liên quan 3](#_Toc183416799)

[2.1 Backend với ASP.NET Core 3](#_Toc183416800)

[2.2 Frontend với ReactJS 3](#_Toc183416801)

[2.3 Cơ sở dữ liệu PostgreSQL 3](#_Toc183416802)

[2.4 Tích hợp mã QR và thanh toán trực tuyến 3](#_Toc183416803)

[3. Phân tích thiết kế 4](#_Toc183416804)

[3.1 Phân tích yêu cầu từ người dùng và quản lý hệ thống 4](#_Toc183416805)

[3.2 Thiết kế cơ sở dữ liệu 5](#_Toc183416806)

[3.3 Thiết kế API RESTful 9](#_Toc183416807)

[3.4 Wireframe và Mockup giao diện 10](#_Toc183416808)

[Kết luận 10](#_Toc183416809)

[4. Mô tả sơ đồ Use Case và đặc tả luồng chức năng 10](#_Toc183416810)

[4.1 Sơ đồ Use Case tổng quan hệ thống 10](#_Toc183416811)

[4.2 Sơ đồ Use Case chi tiết 11](#_Toc183416812)

[Sơ đồ Use Case đăng nhập và phân loại người dùng 11](#_Toc183416813)

[Sơ đồ Use Case quản lý thêm/sửa/xóa phim 12](#_Toc183416814)

[Sơ đồ Use Case quản lý thêm lịch chiếu 14](#_Toc183416815)

[Sơ đồ Use Case đặt vé và đồ ăn/uống 15](#_Toc183416816)

[Sơ đồ Use Case báo cáo doanh thu 17](#_Toc183416817)

# Hệ thống quản lý bán vé xem phim trực tuyến

## 1. Tìm hiểu tổng quan về bài toán

### 1.1 Quy trình vận hành của hệ thống bán vé tại các rạp phim

Hệ thống bán vé xem phim hiện tại ở các rạp chiếu phim thường có hai hình thức chính: bán vé trực tiếp tại quầy và bán vé trực tuyến qua website hoặc ứng dụng.   
- Bán vé tại quầy: Nhân viên hỗ trợ khách hàng chọn phim, suất chiếu, ghế ngồi, và thanh toán trực tiếp. Hình thức này gặp hạn chế trong giờ cao điểm do thời gian chờ lâu và dễ gây ra nhầm lẫn trong khâu giao tiếp.   
- Bán vé trực tuyến: Khách hàng tự chọn phim, suất chiếu và ghế ngồi qua giao diện ứng dụng hoặc website. Thanh toán thường được thực hiện qua các phương thức ngân hàng, ví điện tử hoặc tiền mặt khi nhận vé. Hệ thống này mang lại tiện lợi nhưng đôi khi gặp phải lỗi giao diện, quá tải máy chủ hoặc ghế không cập nhật trạng thái chính xác.

### 1.2 Vấn đề mà khách hàng và quản lý rạp chiếu gặp phải

- Đối với khách hàng:  
 - Khó khăn trong việc chọn ghế trong giờ cao điểm do ghế đã được đặt mà không cập nhật kịp thời.  
 - Quá trình thanh toán phức tạp hoặc thiếu các phương thức thanh toán linh hoạt.  
 - Không nhận được thông tin đầy đủ về phim, lịch chiếu, hoặc các chương trình khuyến mãi.  
- Đối với quản lý rạp chiếu:  
 - Khó khăn trong việc theo dõi doanh thu theo thời gian thực.  
 - Vấn đề khi đồng bộ trạng thái ghế và lịch chiếu trên hệ thống.  
 - Phân tích dữ liệu khách hàng chưa hiệu quả, gây khó khăn trong việc ra quyết định kinh doanh.

### 1.3 Nghiên cứu các hệ thống bán vé trực tuyến hiện có

Các hệ thống bán vé trực tuyến nổi bật như CGV, Lotte Cinema, Galaxy Cinema đã giải quyết phần lớn nhu cầu của khách hàng. Tuy nhiên, vẫn còn một số hạn chế như:  
- Khả năng tùy chỉnh giao diện và tính năng theo từng rạp còn hạn chế.  
- Các chương trình ưu đãi chưa được tích hợp liền mạch vào hệ thống.  
- Thiếu tính năng gợi ý phim hoặc suất chiếu phù hợp dựa trên hành vi khách hàng.

## 2. Tìm hiểu tổng quan về công nghệ liên quan

### 2.1 Backend với ASP.NET Core

ASP.NET Core được chọn để phát triển backend vì:  
- Hỗ trợ tốt cho việc xây dựng API RESTful.  
- Hiệu suất cao và dễ dàng mở rộng.  
- Tích hợp nhiều công cụ bảo mật và quản lý xác thực người dùng như Identity Framework.

### 2.2 Frontend với ReactJS

ReactJS là thư viện JavaScript mạnh mẽ, lý tưởng cho việc xây dựng giao diện người dùng:  
- Cung cấp khả năng tạo UI động, phản hồi nhanh với trải nghiệm người dùng mượt mà.  
- Hỗ trợ mạnh mẽ các thư viện như React Router cho điều hướng và Redux cho quản lý trạng thái.

### 2.3 Cơ sở dữ liệu PostgreSQL

PostgreSQL là lựa chọn phù hợp cho hệ thống:  
- Hỗ trợ mạnh mẽ trong việc quản lý cơ sở dữ liệu quan hệ.  
- Cung cấp các tính năng mở rộng như JSONB, giúp lưu trữ dữ liệu linh hoạt hơn.  
- Tích hợp tốt với ASP.NET Core và hỗ trợ các công cụ ORM như Entity Framework.

### 2.4 Tích hợp mã QR và thanh toán trực tuyến

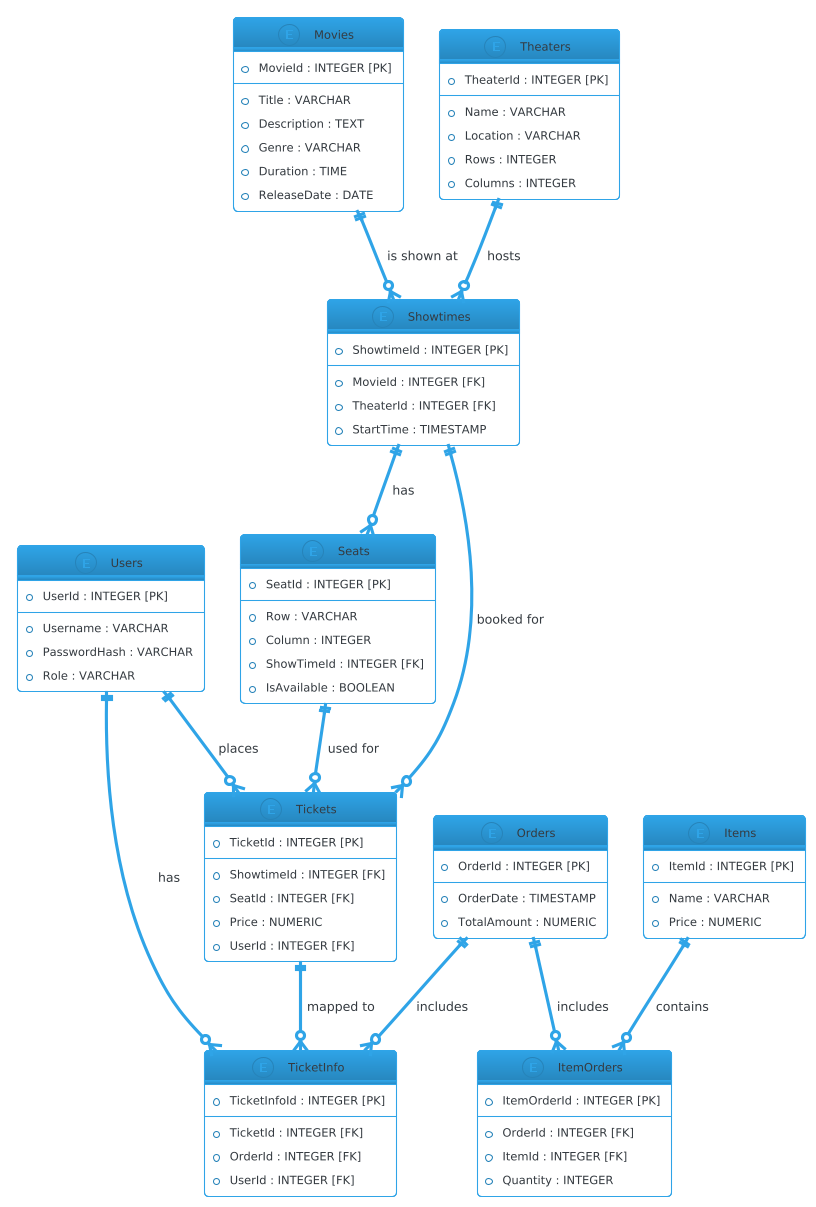
- Mã QR: Được sử dụng để hiển thị thông tin vé, giúp khách hàng dễ dàng quét mã tại cổng vào mà không cần xuất trình vé giấy.  
- Thanh toán trực tuyến: Tích hợp cổng thanh toán qua ngân hàng, ví điện tử (Momo, ZaloPay) để tối ưu hóa trải nghiệm mua sắm.

## 3. Phân tích thiết kế

### 3.1 Phân tích yêu cầu từ người dùng và quản lý hệ thống

- Yêu cầu từ người dùng:  
 - Giao diện dễ sử dụng, cho phép chọn ghế nhanh chóng.  
 - Hỗ trợ nhiều hình thức thanh toán.  
 - Nhận được thông báo qua email hoặc ứng dụng di động sau khi đặt vé thành công.  
- Yêu cầu từ quản lý:  
 - Báo cáo doanh thu theo ngày, tháng, quý.  
 - Theo dõi trạng thái ghế và lịch chiếu theo thời gian thực.  
 - Quản lý danh mục phim, lịch chiếu, và khuyến mãi.

### 3.2 Thiết kế cơ sở dữ liệu



Hệ thống cơ sở dữ liệu bao gồm các bảng chính:  
- Movies: Quản lý thông tin phim (tên phim, thể loại, thời lượng, mô tả).  
- Showtimes: Quản lý lịch chiếu (thời gian, rạp chiếu, bộ phim được chiếu).  
- Seats: Quản lý trạng thái ghế (trống, đã đặt, khóa tạm thời).  
- Tickets: Quản lý thông tin vé (ghế, người mua, suất chiếu).  
- Orders: Quản lý đơn hàng (đồ ăn/uống đi kèm, tổng tiền).  
- Users: Quản lý thông tin người dùng.

Theo đó các bảng có thông tin chi tiết như sau:

**- Bảng Movies**

* **Mô tả**: Quản lý thông tin về các bộ phim được chiếu tại rạp.
* **Thuộc tính chính**: MovieId (Khóa chính), Title, Genre, Duration, Description, ReleaseDate.
* **Quan hệ**:
  + **1:N** với bảng Showtimes: Một bộ phim có thể có nhiều lịch chiếu (mối quan hệ giữa MovieId trong Movies và Showtime.MovieId).

**- Bảng Showtimes**

* **Mô tả**: Quản lý thông tin lịch chiếu, bao gồm thời gian chiếu, rạp chiếu, và bộ phim được chiếu.
* **Thuộc tính chính**: ShowtimeId (Khóa chính), StartTime, MovieId, TheaterId.
* **Quan hệ**:
  + **N:1** với bảng Movies: Lịch chiếu gắn với một bộ phim duy nhất.
  + **N:1** với bảng Theaters: Một lịch chiếu thuộc về một rạp cụ thể.
  + **1:N** với bảng Seats: Một lịch chiếu có thể có nhiều ghế (mối quan hệ giữa ShowtimeId trong Showtimes và Seat.ShowtimeId).
  + **1:N** với bảng Tickets: Một lịch chiếu có thể liên kết với nhiều vé được đặt.

**- Bảng Seats**

* **Mô tả**: Quản lý trạng thái ghế trong từng suất chiếu, bao gồm trạng thái trống, đã đặt, hoặc tạm thời khóa.
* **Thuộc tính chính**: SeatId (Khóa chính), Row, Column, IsAvailable, ShowtimeId.
* **Quan hệ**:
  + **N:1** với bảng Showtimes: Ghế thuộc về một suất chiếu cụ thể.
  + **1:1** với bảng Tickets: Một ghế chỉ có thể liên kết với một vé (nếu đã đặt).

**- Bảng Tickets**

* **Mô tả**: Quản lý thông tin vé đã được đặt, bao gồm ghế, người mua, và suất chiếu.
* **Thuộc tính chính**: TicketId (Khóa chính), SeatId, ShowtimeId, UserId.
* **Quan hệ**:
  + **N:1** với bảng Users: Một vé thuộc về một người dùng cụ thể.
  + **N:1** với bảng Showtimes: Một vé liên kết với một suất chiếu cụ thể.
  + **1:1** với bảng Seats: Một vé chỉ gắn với một ghế duy nhất.
  + **1:N** với bảng TicketInfo: Một vé có thể liên kết với nhiều thông tin bổ sung, chẳng hạn như đồ ăn đi kèm.

**- Bảng Orders**

* **Mô tả**: Quản lý đơn hàng, bao gồm đồ ăn, thức uống được đặt kèm với vé xem phim.
* **Thuộc tính chính**: OrderId (Khóa chính), UserId, TotalAmount.
* **Quan hệ**:
  + **N:1** với bảng Users: Một đơn hàng thuộc về một người dùng cụ thể.
  + **1:N** với bảng ItemOrders: Một đơn hàng có thể bao gồm nhiều món đồ ăn hoặc thức uống.
  + **1:N** với bảng TicketInfo: Một đơn hàng có thể liên kết với nhiều vé (thông qua bảng TicketInfo).

**- Bảng Users**

* **Mô tả**: Quản lý thông tin người dùng, bao gồm khách hàng và quản trị viên.
* **Thuộc tính chính**: UserId (Khóa chính), Username, PasswordHash, Role.
* **Quan hệ**:
  + **1:N** với bảng Orders: Một người dùng có thể tạo nhiều đơn hàng.
  + **1:N** với bảng Tickets: Một người dùng có thể đặt nhiều vé.
  + **1:N** với bảng TicketInfo: Một người dùng có thể liên quan đến nhiều thông tin vé.

**- Bảng ItemOrders**

* **Mô tả**: Quản lý chi tiết các món đồ ăn/thức uống trong từng đơn hàng.
* **Thuộc tính chính**: ItemOrderId (Khóa chính), OrderId, ItemId, Quantity.
* **Quan hệ**:
  + **N:1** với bảng Orders: Một món ăn thuộc về một đơn hàng cụ thể.
  + **N:1** với bảng Items: Mỗi món ăn là một loại cụ thể từ danh sách Items.

**- Bảng Items**

* **Mô tả**: Quản lý danh sách đồ ăn và thức uống được cung cấp tại rạp.
* **Thuộc tính chính**: ItemId (Khóa chính), Name, Price.
* **Quan hệ**:
  + **1:N** với bảng ItemOrders: Một món ăn có thể xuất hiện trong nhiều đơn hàng.

**- Bảng Theaters**

* **Mô tả**: Quản lý thông tin rạp chiếu phim.
* **Thuộc tính chính**: TheaterId (Khóa chính), Name, Location, Rows, Columns.
* **Quan hệ**:
  + **1:N** với bảng Showtimes: Một rạp chiếu có thể có nhiều lịch chiếu.

**- Bảng TicketInfo**

* **Mô tả**: Liên kết giữa vé và đơn hàng để quản lý thông tin chi tiết về vé và đồ ăn đi kèm.
* **Thuộc tính chính**: TicketInfoId (Khóa chính), TicketId, OrderId, UserId.
* **Quan hệ**:
  + **N:1** với bảng Tickets: Một thông tin vé gắn với một vé duy nhất.
  + **N:1** với bảng Orders: Một thông tin vé có thể thuộc về một đơn hàng cụ thể.
  + **N:1** với bảng Users: Một thông tin vé liên kết với một người dùng.

### 3.3 Thiết kế API RESTful

API RESTful được chia thành các nhóm chính:  
- Movie API: Cung cấp danh sách phim, chi tiết phim.  
- Showtime API: Trả về danh sách lịch chiếu theo phim hoặc rạp.  
- Seat API: Trả về trạng thái ghế theo lịch chiếu.  
- Order API: Xử lý đặt đồ ăn/uống, lưu thông tin thanh toán.  
- Ticket API: Tạo và quản lý vé xem phim.

### 3.4 Wireframe và Mockup giao diện

- Trang đặt vé: Hiển thị danh sách ghế, hỗ trợ chọn nhiều ghế một lúc.  
- Trang thanh toán: Hiển thị danh sách ghế đã chọn và các tùy chọn đồ ăn/uống.  
- Trang quản lý: Cung cấp biểu đồ doanh thu, trạng thái lịch chiếu và ghế.

## Kết luận

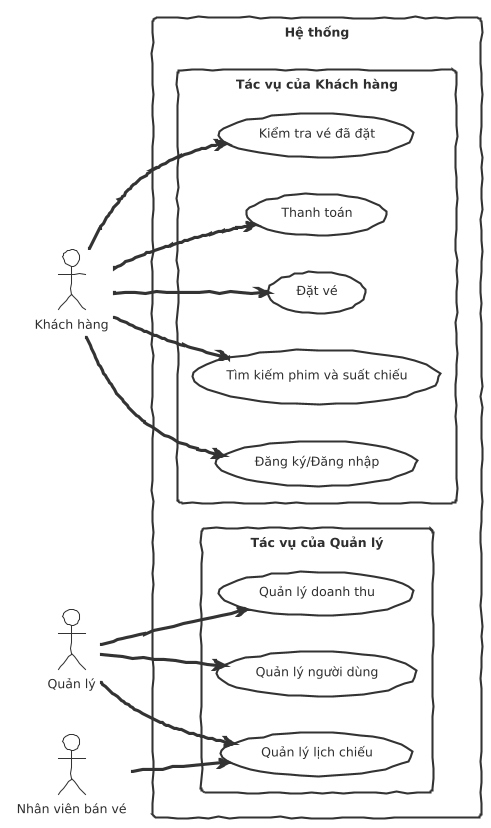
Việc xây dựng hệ thống quản lý bán vé xem phim không chỉ giải quyết các vấn đề hiện tại mà còn mang lại trải nghiệm mượt mà hơn cho khách hàng và tối ưu hóa hoạt động quản lý cho rạp chiếu phim. Bằng cách áp dụng công nghệ hiện đại như ASP.NET Core, ReactJS và PostgreSQL, hệ thống này hứa hẹn trở thành một công cụ hiệu quả, hỗ trợ tốt cho cả người dùng và nhà quản lý.

# 4. Mô tả sơ đồ Use Case và đặc tả luồng chức năng

Phần này trình bày các sơ đồ Use Case và luồng chức năng chính của hệ thống quản lý bán vé xem phim trực tuyến.   
Mỗi sơ đồ được giải thích chi tiết để thể hiện rõ cách hệ thống đáp ứng các yêu cầu từ người dùng và quản trị viên.

## 4.1 Sơ đồ Use Case tổng quan hệ thống

Sơ đồ Use Case tổng quan của hệ thống thể hiện các chức năng chính mà hệ thống cung cấp cho các loại người dùng khác nhau: khách hàng và quản trị viên.

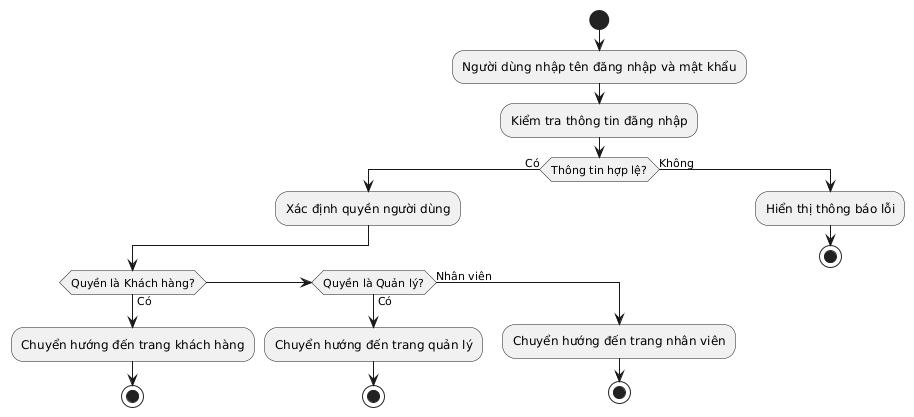


## 4.2 Sơ đồ Use Case chi tiết

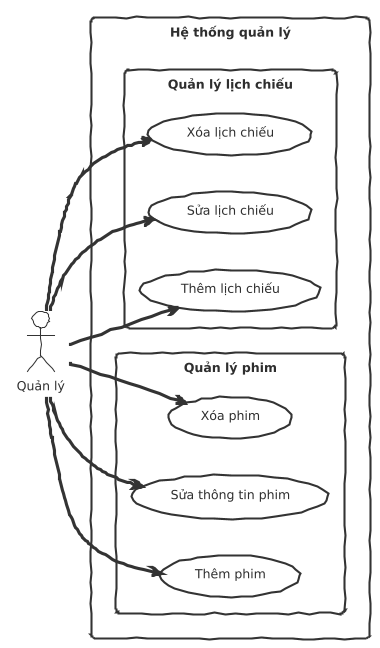
Các sơ đồ chi tiết bao gồm: Đăng nhập, đặt vé, đặt đồ ăn/uống, quản lý của admin, và báo cáo.

### Sơ đồ Use Case đăng nhập và phân loại người dùng

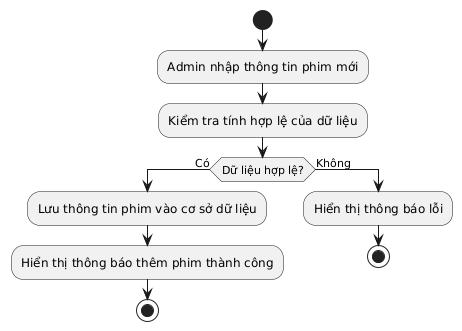
Mô tả: Đây là sơ đồ chi tiết cho sơ đồ use case đăng nhập và phân loại người dùng.



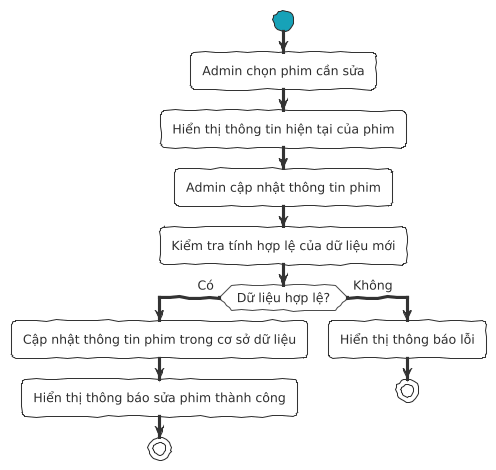
### Sơ đồ Use Case quản lý thêm/sửa/xóa phim



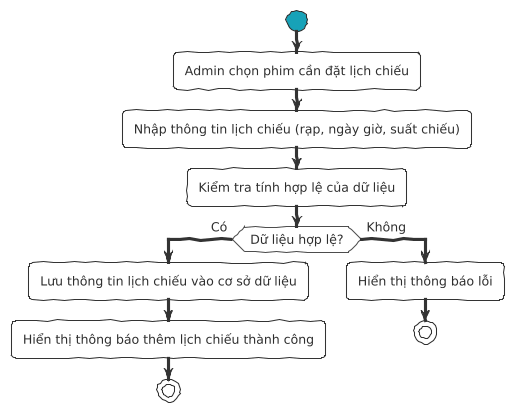
Chức năng thêm phim



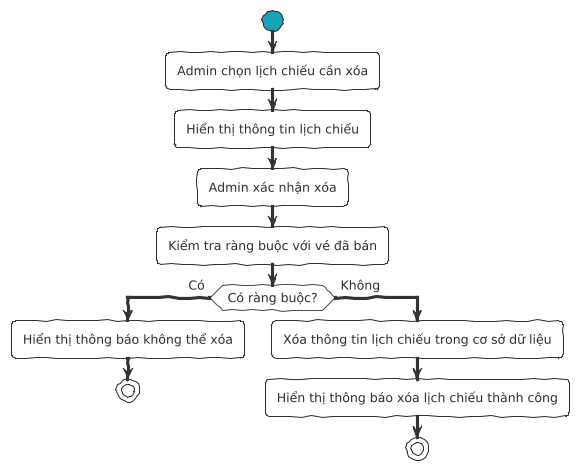
Chức năng sửa thông tin phim



### Sơ đồ Use Case quản lý thêm lịch chiếu

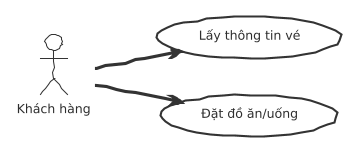


Chức năng xoá lịch chiếu

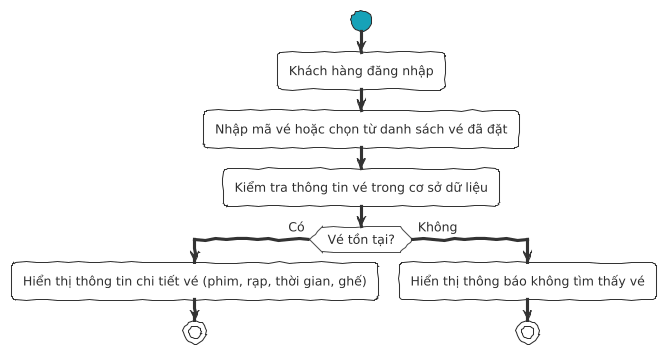


### Sơ đồ Use Case đặt vé và đồ ăn/uống

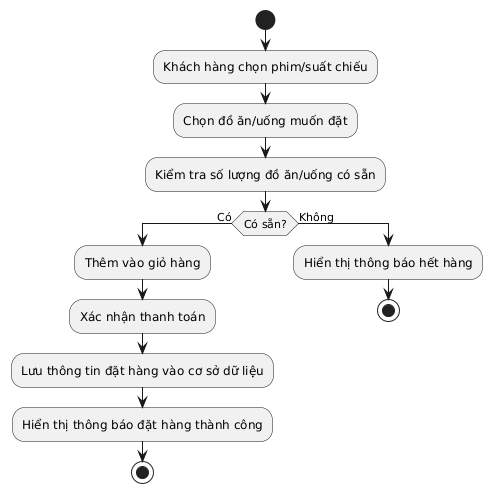
Lấy thông tin vé và đồ ăn



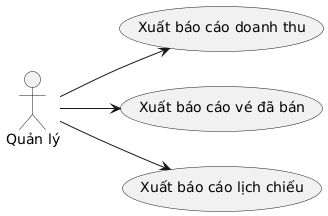
Lấy thông tin vé chi tiết



Đặt đồ ăn



### Sơ đồ Use Case báo cáo doanh thu



Xuất báo cáo doanh thu



**Sơ đồ đặc tả luồng chức năng: Xuất báo cáo vé đã bán**

