**Võ Thành Luân TÊN ĐỀ TÀI: XÂY DỰNG WEBSITE QUẢN LÝ RẠP CHIẾU PHIM CHO RẠP LOTTE ĐÀ NẴNG 2022**

**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT**

**KHOA CÔNG NGHỆ SỐ**

**ĐỒ ÁN TỐT NGHIỆP**

**ĐẠI HỌC**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG WEBSITE QUẢN LÝ RẠP CHIẾU PHIM ÁP DỤNG CHO RẠP LOTTE ĐÀ NẴNG**

Người hướng dẫn: **TS. Nguyễn Tấn Thuận**

Sinh viên thực hiện: **Võ Thành Luân**

Mã sinh viên: **1811505310324**

Lớp**: 18T3**

**Đà Nẵng, 06/2022**

**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT**

**KHOA CÔNG NGHỆ SỐ**

**ĐỒ ÁN TỐT NGHIỆP**

**ĐẠI HỌC**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG WEBSITE QUẢN LÝ RẠP CHIẾU PHIM ÁP DỤNG CHO RẠP LOTTE ĐÀ NẴNG**

Người hướng dẫn: **TS. Nguyễn Tấn Thuận**

Sinh viên thực hiện: **Võ Thành Luân**

Mã sinh viên: **1811505310324**

Lớp**: 18T3**

**Đà Nẵng, 06/2022**

NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN

NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN

TÓM TẮT

Tên đề tài: Xây dựng hệ thống website quản lý áp dụng cho rạp Lotte Đà Nẵng.

Sinh viên thực hiện: Võ Thành Luân

Mã SV: 1811505310324 Lớp: 18T3

Đồ án với đề tài Website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng được xây dựng bằng ngôn ngữ lập trình Java, sử dụng Spring Framework, Angular và sử dụng hệ quản trị cơ sở dữ liệu MySQL cùng các plugin hỗ trợ như Bootstrap, JQuery, Ajax.

Website bao gồm các chức năng chính dành cho người sử dụng không phải là khách hàng – thành viên như: đăng ký tài khoản, đăng nhập, xem danh sách các phim, tìm kiếm các phim theo nhiều thể loại, tìm kiếm các khuyến mãi của rạp, xem giá vé, lịch chiếu. Các chức năng dành cho người sử dụng là khách hàng thành viên cũng giống như khách hàng không phải là thàn viên, ngoài ra còn có: đặt vé, quản lý tài khoản, xem lịch sử đặt vé,… Đối với người sử dụng là nhân viên có các chức năng như: xem và tìm kiếm nhân viên, quản lý phim, quản lý bán vé, quản lý đặt vé, quản lý khuyến mãi,… Đối với người sử dụng là chủ rạp chiếu hoặc người quản lý thì ngoài các chức năng của nhân viên, có thể sử dụng các chức năng khác như xem thống kê, quản lý nhân viên, quản lý thành viên, quản lý phòng chiếu, quản lý suất chiếu,…

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Giảng viên hướng dẫn: TS. Nguyễn Tấn Thuận

Sinh viên thực hiện: Võ Thành Luân Mã SV: 1811505310324

***1. Tên đề tài:***

Xây dựng hệ thống website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng.

***2. Các số liệu, tài liệu ban đầu:***

* Dựa trên qui định upload tài liệu, các qui tắc đặt câu hỏi.
* Tài liệu tham khảo: <https://levunguyen.com/>

***3. Nội dung chính của đồ án:***

Quá trình xây dựng hệ thống website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng:

* Thu thập thông tin tài liệu liên quan và khảo sát thực tế.
* Phân tích thiết kế các chức năng của hệ thống.
* Phân tích thiết kế cơ sơ dữ liệu.
* Thiết kế giao diện cho các chức năng.
* Xây dựng hệ thống website.
* Kiểm thử chương trình.
* Hoàn thành báo cáo tổng hợp.

***4. Các sản phẩm dự kiến***

* Website quản lý phòng họp.
* File báo cáo hoàn chỉnh.

***5. Ngày giao đồ án: 14/03/2022***

***6. Ngày nộp đồ án: 14/06/2022***

|  |  |
| --- | --- |
|  | *Đà Nẵng, ngày tháng năm 2022* |
| **Trưởng Bộ môn** | **Người hướng dẫn** |

LỜI NÓI ĐẦU

Lời đầu tiên, em xin gửi lời cảm ơn chân thành đến quý thầy cô đã giúp đỡ chúng em thực hiện đề tài này. Đặc biệt là TS. Nguyễn Tấn Thuận đã tận tình giúp đỡ em trong suốt quá trình thực hiện đề tài tốt nghiệp này.

Đồng thời, em cũng xin cảm ơn quý thầy cô thuộc ngành Công nghệ thông tin – khoa Công nghệ số, trường Đại học Sư phạm Kỹ thuật - Đại học Đà Nẵng đã truyền đạt những kiến thức cần thiết và những kinh nghiệm quý báu cho chúng em trong suốt thời gian 4 năm trên giảng đường để em có thể thực hiện tốt đề tài này. Đặc biệt, em xin gởi lời cảm ơn chân thành tới TS. Hoàng Thị Mỹ Lệ, TS. Nguyễn Tấn Thuận, cùng ThS. Nguyễn Thị Hà Quyên - giáo viên chủ nhiệm lớp 18T3 đã giúp đỡ em rất nhiều trong quá trình học tập và công việc.

Trong quá trình thực hiện đề tài, do kiến thức và thời gian còn hạn chế nên không thể tránh khỏi những sai sót. Vì vậy em mong quý thầy, cô thông cảm và góp ý để em có thể hoàn thiện đề tài. Và những lời góp ý đó có thể giúp em có thể tránh được những sai lầm sau này.

Em xin chân thành cảm ơn!

Đà Nẵng, ngày tháng năm 2022

**Sinh viên thực hiện**

LỜI CAM ĐOAN

Em xin cam đoan:

* + Những nội dung trong luận văn này do em thực hiện dưới sự hướng dẫn của thầy Nguyễn Tấn Thuận.
  + Mọi tham khảo trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.
  + Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo, hay gian trá, em xin chịu hoàn toàn trách nhiệm.

Sinh viên thực hiện

MỤC LỤC

[NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN](#_Toc101277497)

[NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN](#_Toc101277498)

[TÓM TẮT](#_Toc101277499)

[NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP](#_Toc101277500)

[LỜI NÓI ĐẦU i](#_Toc101277501)

[LỜI CAM ĐOAN ii](#_Toc101277502)

[MỤC LỤC iii](#_Toc101277503)

[DANH SÁCH CÁC BẢNG, HÌNH VẼ iv](#_Toc101277504)

[DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT v](#_Toc101277505)

Trang

[MỞ ĐẦU 1](#_Toc102365816)

[CHƯƠNG I. TỔNG QUAN ĐỀ TÀI 2](#_Toc102365817)

[1.1. Tổng quan đề tài 2](#_Toc102365818)

[1.2. Mục tiêu đề tài 2](#_Toc102365819)

[1.3. Phạm vi nghiên cứu 3](#_Toc102365820)

[1.4. Phương pháp nghiên cứu 3](#_Toc102365821)

[1.5. Giải pháp công nghệ 3](#_Toc102365822)

[1.6. Đặc tả yêu cầu nghiệp vụ 3](#_Toc102365823)

[1.6.1. User 3](#_Toc102365824)

[1.6.2. Employee 3](#_Toc102365825)

[1.6.3. Admin (gồm các chức năng của nhân viên) 3](#_Toc102365826)

[CHƯƠNG II. CƠ SỞ LÝ THUYẾT 4](#_Toc102365827)

[2.1. Ngôn ngữ lập trình Java 4](#_Toc102365828)

[2.1.1. Java là gì? 4](#_Toc102365829)

[2.1.2. Lịch sử phát triển 4](#_Toc102365830)

[2.2. Spring Framework 5](#_Toc102365831)

[2.2.1. Đôi nét về lịch sử 5](#_Toc102365832)

[2.2.2. Tổng quan về Spring 5](#_Toc102365833)

[2.3. Tổng quan về hệ quản trị cơ sở dữ liệu MySQL 6](#_Toc102365834)

[2.4. Restful API (RestController Spring Boot) 7](#_Toc102365835)

[2.5. Bootstrap và responsive 7](#_Toc102365836)

[2.6. Typescript và Angular Framework 8](#_Toc102365837)

[2.6.1. TypeScript 8](#_Toc102365838)

[2.6.2. Angular 8](#_Toc102365839)

[CHƯƠNG III. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 10](#_Toc102365840)

[3.1. Khảo sát hệ thống. 10](#_Toc102365841)

[3.1.1. Khảo sát thực tế 10](#_Toc102365842)

[3.1.2. Khảo sát người dùng 10](#_Toc102365843)

[3.2. Đặc tả yêu cầu phần mềm 10](#_Toc102365844)

[3.2.1. Xác định các tác nhân 10](#_Toc102365845)

[3.2.2. Các yêu cầu chức năng 10](#_Toc102365846)

[3.2.3. Yêu cầu phi chức năng 11](#_Toc102365847)

[3.3. Biểu đồ Use Case 11](#_Toc102365848)

[3.3.1 Đăng ký và đăng nhập 11](#_Toc102365849)

[3.3.2 Quản lý tài khoản 11](#_Toc102365850)

[3.3.3 Xem danh sách phim 12](#_Toc102365851)

[3.3.4 Đặt vé 12](#_Toc102365852)

[3.3.5 Xem lịch chiếu 13](#_Toc102365853)

[3.3.6 Quản lý đặt vé 13](#_Toc102365854)

[3.3.7 Quản lý nhân viên 14](#_Toc102365855)

[3.3.8 Quản lý phòng chiếu 14](#_Toc102365856)

[3.3.9 Quản lý thành viên 14](#_Toc102365857)

[3.3.10 Quản lý phim 15](#_Toc102365858)

[3.3.11 Thống kê 15](#_Toc102365859)

[3.4. Scenario cho Use Case 16](#_Toc102365860)

[3.4.1 Scenario Use-case “Đăng ký và đăng nhập” 16](#_Toc102365861)

[3.4.2 Scenario Use-case “Quản lý tài khoản” 16](#_Toc102365862)

[3.4.3 Scenario Use-case “Xem danh sách và tìm kiếm phim” 17](#_Toc102365863)

[3.4.4 Scenario Use-case “Đặt vé” 17](#_Toc102365864)

[3.4.5 Scenario Use-case “Xem lịch chiếu phim” 18](#_Toc102365865)

[3.4.6 Scenario Use-case “Quản lý đặt vé” 18](#_Toc102365866)

[3.4.7 Scenario Use-case “Quản lý nhân viên” 19](#_Toc102365867)

[3.4.8 Scenario Use-case “Quản lý thành viên” 19](#_Toc102365868)

[3.4.9 Scenario Use-case “Quản lý phim” 20](#_Toc102365869)

[3.4.10 Scenario Use-case “Quản lý phòng chiếu” 20](#_Toc102365870)

[3.4.11 Scenario Use-case “Thống kê” 21](#_Toc102365871)

[3.5. Sơ đồ hoạt động 21](#_Toc102365872)

[3.5.1 Đăng ký và đăng nhập 21](#_Toc102365873)

[3.5.2 Quản lý tài khoản 22](#_Toc102365874)

[3.5.3 Xem và tìm kiếm phim 23](#_Toc102365875)

[3.5.4 Đặt vé 24](#_Toc102365876)

[3.5.5 Xem lịch chiếu 25](#_Toc102365877)

[3.5.6 Quản lý đặt vé 26](#_Toc102365878)

[3.5.7 Quản lý nhân viên 27](#_Toc102365879)

[3.5.8 Quản lý thành viên 28](#_Toc102365880)

[3.5.9 Quản lý phim 29](#_Toc102365881)

[3.5.10 Quản lý phòng chiếu 30](#_Toc102365882)

[3.5.11 Thống kê 31](#_Toc102365883)

[3.6. Sơ đồ ERD 32](#_Toc102365884)

[3.7. Thiết kế bảng cơ sở dữ liệu 33](#_Toc102365885)

[3.7.1 Bảng role 33](#_Toc102365886)

[3.7.2 Bảng account\_role 33](#_Toc102365887)

[3.7.3 Bảng account 33](#_Toc102365888)

[3.7.4 Bảng transaction\_history 34](#_Toc102365889)

[3.7.5 Bảng province 34](#_Toc102365890)

[3.7.6 Bảng district 34](#_Toc102365891)

[3.7.7 Bảng ward 34](#_Toc102365892)

[3.7.8 Bảng user 35](#_Toc102365893)

[3.7.9 Bảng column 35](#_Toc102365894)

[3.7.10 Bảng row 35](#_Toc102365895)

[3.7.11 Bảng seat 36](#_Toc102365896)

[3.7.12 Bảng category 36](#_Toc102365897)

[3.7.13 Bảng movie 36](#_Toc102365898)

[3.7.14 Bảng movie\_category 37](#_Toc102365899)

[3.7.15 Bảng show\_time 37](#_Toc102365900)

[3.7.16 Bảng room 37](#_Toc102365901)

[3.7.17 Bảng movie\_room 37](#_Toc102365902)

[3.7.18 Bảng room\_seat 38](#_Toc102365903)

[3.7.19 Bảng movie\_ticket 38](#_Toc102365904)

[3.7.20 Bảng ticket 38](#_Toc102365905)

[CHƯƠNG IV. KẾT QUẢ TRIỂN KHAI HỆ THỐNG WEBSITE 40](#_Toc102365906)

[4.1. Công cụ xây dựng 40](#_Toc102365907)

[4.2. Thiết kế giao diện 40](#_Toc102365908)

[4.2.1 Giao diện 40](#_Toc102365909)

[4.2.2 Giao diện 40](#_Toc102365910)

[4.2.3 Giao diện 40](#_Toc102365911)

[4.2.4 Giao diện 40](#_Toc102365912)

[4.2.5 Giao diện 40](#_Toc102365913)

[4.2.6 Giao diện 40](#_Toc102365914)

[4.2.7 Giao diện 40](#_Toc102365915)

[4.2.8 Giao diện 40](#_Toc102365916)

[4.2.9 Giao diện 40](#_Toc102365917)

[KẾT LUẬN 41](#_Toc102365918)

[TÀI LIỆU THAM KHẢO 42](#_Toc102365919)

[PHỤ LỤC 1](#_Toc102365920)

DANH SÁCH CÁC BẢNG, HÌNH VẼ

[Bảng 3.4.1 Scenario Use-case “Đăng ký và đăng nhập” 16](#_Toc102365960)

[Bảng 3.4.2 Scenario Use-case “Quản lý tài khoản” 17](#_Toc102365961)

[Bảng 3.4.3 Scenario Use-case “Xem danh sách và tìm kiếm phim” 17](#_Toc102365962)

[Bảng 3.4.4 Scenario Use-case “Đặt vé” 18](#_Toc102365963)

[Bảng 3.4.5 Scenario Use-case “Xem lịch chiếu phim” 18](#_Toc102365964)

[Bảng 3.4.6 Scenario Use-case “Quản lý đặt vé” 18](#_Toc102365965)

[Bảng 3.4.7 Scenario Use-case “Quản lý nhân viên” 19](#_Toc102365966)

[Bảng 3.4.8 Scenario Use-case “Quản lý thành viên” 19](#_Toc102365967)

[Bảng 3.4.9 Scenario Use-case “Quản lý phim” 20](#_Toc102365968)

[Bảng 3.4.10 Scenario Use-case “Quản lý phòng chiếu” 20](#_Toc102365969)

[Bảng 3.4.11 Scenario Use-case “Thống kê” 21](#_Toc102365970)

[Bảng 3.7.1 role 33](#_Toc102365971)

[Bảng 3.7.2 account\_role 33](#_Toc102365972)

[Bảng 3.7.3 account 33](#_Toc102365973)

[Bảng 3.7.4 transaction\_history 34](#_Toc102365974)

[Bảng 3.7.5 province 34](#_Toc102365975)

[Bảng 3.7.6 district 34](#_Toc102365976)

[Bảng 3.7.7 ward 34](#_Toc102365977)

[Bảng 3.7.8 property 35](#_Toc102365978)

[Bảng 3.7.9 column 35](#_Toc102365979)

[Bảng 3.7.10 row 35](#_Toc102365980)

[Bảng 3.7.11 seat 36](#_Toc102365981)

[Bảng 3.7.12 category 36](#_Toc102365982)

[Bảng 3.7.13 movie 36](#_Toc102365983)

[Bảng 3.7.14 movie\_category 37](#_Toc102365984)

[Bảng 3.7.15 show\_time 37](#_Toc102365985)

[Bảng 3.7.16 room 37](#_Toc102365986)

[Bảng 3.7.17 movie\_room 37](#_Toc102365987)

[Bảng 3.7.18 room\_seat 38](#_Toc102365988)

[Bảng 3.7.19 movie\_ticket 38](#_Toc102365989)

[Bảng 3.7.20 ticket 38](#_Toc102365990)

[Hình 2.2.2 : Mô hình Spring Framework Runtime 6](#_Toc102365991)

[Hình 3.3.1 Use case đăng ký và đăng nhập 11](#_Toc102365992)

[Hình 3.3.2 Use case quản lý tài khoản 12](#_Toc102365993)

[Hình 3.3.3 Use case xem danh sách và tìm kiếm phim 12](#_Toc102365994)

[Hình 3.3.4 Use case đặt vé 13](#_Toc102365995)

[Hình 3.3.5 Use case xem lịch chiếu 13](#_Toc102365996)

[Hình 3.3.6 Use case quản lý đặt vé 13](#_Toc102365997)

[Hình 3.3.7 Use case quản lý nhân viên 14](#_Toc102365998)

[Hình 3.3.8 Use case quản lý phòng chiếu 14](#_Toc102365999)

[Hình 3.3.9 Use case quản lý thành viên 15](#_Toc102366000)

[Hình 3.3.10 Use case quản lý phim 15](#_Toc102366001)

[Hình 3.3.11 Use case thống kê 16](#_Toc102366002)

[Hình 3.5.1 Activity “Đăng ký và đăng nhập” 22](#_Toc102366003)

[Hình 3.5.2 Activity “Quản lý tài khoản” 23](#_Toc102366004)

[Hình 3.5.3 Activity “Xem và tìm kiếm phim” 24](#_Toc102366005)

[Hình 3.5.4 Activity “Đặt vé” 25](#_Toc102366006)

[Hình 3.5.5 Activity “Xem lịch chiếu” 26](#_Toc102366007)

[Hình 3.5.6 Activity “Quản lý đặt vé” 27](#_Toc102366008)

[Hình 3.5.7 Activity “Quản lý nhân viên” 28](#_Toc102366009)

[Hình 3.5.8 Activity “Quản lý thành viên” 29](#_Toc102366010)

[Hình 3.5.9 Activity “Quản lý phim” 30](#_Toc102366011)

[Hình 3.5.10 Activity “Quản lý phòng chiếu” 31](#_Toc102366012)

[Hình 3.5.11 Activity “Thống kê” 32](#_Toc102366013)

[Hình 3.6 Sơ đồ ERD 33](#_Toc102366014)

[Hình 4.2.1 Giao diện 40](#_Toc102366015)

[Hình 4.2.2 Giao diện 40](#_Toc102366016)

[Hình 4.2.3 Giao diện 40](#_Toc102366017)

[Hình 4.2.4 Giao diện 40](#_Toc102366018)

[Hình 4.2.5 Giao diện 40](#_Toc102366019)

[Hình 4.2.6 Giao diện 40](#_Toc102366020)

[Hình 4.2.7 Giao diện 40](#_Toc102366021)

[Hình 4.2.8 Giao diện 40](#_Toc102366022)

[Hình 4.2.9 Giao diện 40](#_Toc102366023)

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

KÝ HIỆU:

CHỮ VIẾT TẮT:

MỞ ĐẦU

Với nhu cầu đặt vé phim online cho khách hàng và quản lý rạp chiếu cho chủ rạp thì việc làm cấp thiết đặt ra là cần xây dựng một hệ thống website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng nhằm tạo điều kiện thuận lợi cho khách hàng có thể tìm kiếm thông tin phim và đặt vé phim trực tuyến, chủ rạp có thể quản lý rạp của mình dễ dàng.

Phạm vi nghiên cứu đề tài: Website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng

Cấu trúc đồ án tốt nghiệp gồm các phần như sau:

- Mở đầu: Giới thiệu lý do và các mục tiêu mà đề tài cần giải quyết, phạm vi nghiên cứu của đề tài. Giới thiệu tóm tắt nội dung sẽ được trình bày trong các chương tiếp theo.

- Chương 1: Nghiên cứu, tìm hiểu và đưa ra lý do chọn đề tài, mục tiêu và mục đích khi xây dựng và phát triển đề tài. Xác định rõ phạm vi và đối tượng hướng đến, giải pháp công nghệ để triển khai, xây dựng đề tài, đồng thời phân tích đặc tả yêu cầu nghiệp vụ.

- Chương 2: Tìm hiểu, giới thiệu tổng quát các kiến thức về ngôn ngữ lập trình Java, TypeScript, Spring Framework, Angular và hệ quản trị cơ sở dữ liệu MySQL.

- Chương 3: Phân tích các tác nhân và chức năng của từng tác nhân, yêu cầu phi chức năng của hệ thống. Thiết kế sơ đồ use-case, sơ đồ hoạt động, sơ đồ ERD. Thiết kế cơ sở dữ liệu. Xây dựng kịch bản cho từng use-case trong hệ thống.

- Chương 4: Xây dựng giao diện và chức năng của hệ thống.

- Kết luận: Kết luận chung cho các chương trong đồ án. Trình bày những vấn đề đã giải quyết đồng thời trình bày hướng phát triển.

CHƯƠNG I. TỔNG QUAN ĐỀ TÀI

* 1. Tổng quan đề tài

Hiện nay, nhu cầu giải trí xem phim rạp tăng cao, các hệ thống rạp chiếu phim phát triển nhanh chóng, để phục vụ nhu cầu xem phim của khách hàng, việc thành lập một website quản lý rạp phim là cần thiết. Thông qua hệ thống website, khách hàng có thể cập nhật thông tin của những bộ phim mới nhất, đầy đủ và chi tiết cho từng suất chiếu hoặc nhanh chóng đặt mua vé trước (đặt vé trực tuyến) để xem những suất chiếu đầu tiên của các siêu phẩm điện ảnh.

Từ các do trên, em đã chọn đề tài “xây dựng website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng” làm hướng nghiên cứu cho đề tài.

* 1. Mục tiêu đề tài

Xây dựng website quản lý rạp chiếu phim cho rạp Lotte Đà Nẵng. Với các chức năng cụ thể:

\*\*\* Đối khách hàng chưa phải là thành viên: gồm các chức năng cơ bản

* + Đăng ký thành viên, đăng nhập hệ thống.
  + Xem danh sách phim, xem chi tiết phim.
  + Tìm kiếm phim (theo tên, theo ngày chiếu, theo suất chiếu, theo thể loại,…)
  + Xem danh sách các khuyến mãi.
  + Xem lịch chiếu phim

\*\*\* Đối với khách hàng là thành viên: giống với khách hàng chưa phải là thành viên và có thêm một số chức năng

* + Đặt vé phim online.
  + Quản lý hồ sơ cá nhân.
  + Xem lịch sử đặt vé

\*\*\* Đối với nhân viên: bao gồm các chức năng quản lý

* + Xem danh sách thành viên.
  + Tìm kiếm thành viên.
  + Quản lý phim, tìm kiếm phim.
  + Quản lý bán vé.
  + Quản lý đặt vé.
  + Quản lý khuyến mãi.

\*\*\* Đối với nhân viên: giống với nhân viên và có thêm một số chức năng

* + Xem thống kê (theo phim, theo thành viên, theo doanh thu,…)
  + Quản lý nhân viên.
  + Quản lý thành viên.
  + Quản lý phòng chiếu.
  + Quản lý suất chiếu
  1. Phạm vi nghiên cứu

Đề tài được áp dụng cho rạp chiếu phim Lotte Đà Nẵng và sau này có thể mở rộng phạm vi ra các rạp chiếu phim khác.

* 1. Phương pháp nghiên cứu
     + Phương pháp nghiên cứu lý thuyết thu thập thông tin qua internet, các tài liệu, sách để tìm được các cơ sở lý thuyết liên quan vấn đề mình nghiên cứu.
     + Phương pháp triển khai thực nghiệp: xây dựng website.
  2. Giải pháp công nghệ
     + Ngôn ngữ lập trình: Java - Spring Framework, TypeScript – Angular.
     + Hệ quản trị cơ sở dữ liệu: MySQL.
     + Công cụ hỗ trợ: IntelliJ IDEA, WebStorm, MySQL Workbench
  3. Đặc tả yêu cầu nghiệp vụ
     1. User
* Xem danh sách các phim, xem chi tiết thông tin phim.
* Tìm kiếm phim.
* Quản lý hồ sơ cá nhân.
* Xem lịch sử đặt vé.
* Đăng nhập, đăng xuất, đổi mật khẩu.
* Đặt vé phim.
* Xem giá vé, lịch chiếu, khuyến mãi.
  + 1. Employee
* Xem danh sách thành viên.
* Tìm kiếm thành viên.
* Quản lý bán vé.
* Quản lý đặt vé.
* Quản lý khuyến mãi.
* Quản lý hồ sơ cá nhân.
* Đăng nhập, đăng xuất, đổi mật khẩu.
  + 1. Admin (gồm các chức năng của nhân viên)
* Thống kê.
* Quản lý nhân viên.
* Quản lý thành viên.
* Quản lý phòng chiếu.
* Quản lý suất chiếu.

CHƯƠNG II. CƠ SỞ LÝ THUYẾT

2.1. Ngôn ngữ lập trình Java

2.1.1. Java là gì?

Java là một ngôn ngữ lập trình dạng lập trình hướng đối tượng (OOP). Khác với phần lớn ngôn ngữ lập trình thông thường, thay vì biên dịch mã nguồn thành mã máy hoặc thông dịch mã nguồn khi chạy, Java được thiết kế để biên dịch mã nguồn thành Bytecode, Bytecode sau đó sẽ được môi trường thực thi (Runtime Environment) chạy. Cú pháp Java được vay mượn nhiều từ C và C++ nhưng có cú pháp hướng đối tượng đơn giản hơn và ít tính năng xử lý cấp thấp hơn. Do đó việc viết một chương trình bằng Java dễ hơn, đơn giản hơn, đỡ tốn công sửa lỗi hơn. [1]

2.1.2. Lịch sử phát triển

Java được khởi đầu bởi James - Gosling và bạn đồng nghiệp ở Sun Microsystems năm 1991. Ban đầu ngôn ngữ này được gọi là Oak (có nghĩa là cây sồi) do bên ngoài cơ quan của ông Gosling có trồng nhiều loại cây này. Java được phát hành vào năm 1994. Sau khi Oracle mua lại công ty Sun Microsystems năm 2009 - 2010, Oracle đã mô tả họ là "người quản lý công nghệ Java với cam kết không ngừng để bồi dưỡng một cộng đồng tham gia và minh bạch". Lịch sử phiên bản java:

* JDK 1.0 (23 tháng 01, 1996)
* JDK 1.1 (19 tháng 2, 1997)
* J2SE 1.2 (Playground) 08 tháng 12, 1998
* J2SE 1.3 (Kestrel) 08 tháng 5, 2000
* J2SE 1.4.0 (Merlin) 06 tháng 02, 2002
* J2SE 5 (1.5.0) (Tiger) 30 tháng 9, 2004
* Java SE 6 (còn gọi là Mustang), được công bố 11 tháng 12 năm 2006.
* Java SE 7 (còn gọi là Dolphin), được bắt đầu từ tháng 8 năm 2006 và công bố ngày 28 tháng 7 năm 2011.
* Java SE 8 ngày 18 tháng 3 năm 2014
* Java SE 9 ngày 21 tháng 9 năm 2017
* Java SE 10 ngày 20 tháng 3 năm 2018
* Java SE 11 ngày 25 tháng 9 năm 2018
* Java SE 12 ngày 19 tháng 3 năm 2019
* Java SE 13 ngày 17 tháng 9 năm 2019
* Java SE 14 ngày 17 tháng 3 năm 2020
* Java SE 15 ngày 15 tháng 9 năm 2020
* Java SE 16 ngày 16 tháng 3 năm 2021
* Java SE 17 ngày 14 tháng 9 năm 2021

2.2. Spring Framework

2.2.1. Đôi nét về lịch sử

Vào năm 2002, Spring Framework phát hành phiên bản đầu tiên bởi Rod Johnson.

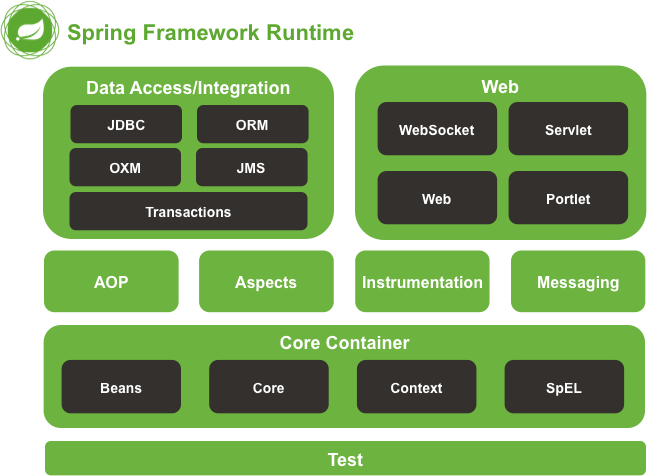
Việc xây dựng các ứng dụng doanh nghiệp trở nên đơn giản và dễ dàng hơn. Rod Johnson đã cho xuất bản một cuốn sách được phổ biến rộng rãi có nhan đề "J2EE Development without EJB". Điều này đã cho Spring Framework rất nhiều động lực để thay thế EJB.[2]

Ngày nay, Spring đã trở thành framework mã nguồn mở phổ biến nhất để xây dựng các ứng dụng doanh nghiệp. Cách tiếp cận thực tế ban đầu của Rod Johnson tiếp tục được phát triển và hướng tới một bộ công cụ hoàn chỉnh dành cho xây dựng các ứng dụng doanh nghiệp. Theo một số nguồn, trên 50% các ứng dụng web Java hiện nay đang sử dụng Spring Framework.

Để ngăn chặn sự phức tạp trong phát triển các ứng dụng, Spring Framework thường dựa trên các quan điểm như sau:

* Đơn giản hóa công việc phát triển thông qua việc sử dụng các đối tượng Java đơn giản hay còn được gọi là POJO (Plain Old Java Object).
* Nới lỏng ràng buộc giữa các thành phần thông qua việc sử dụng Dependency Injection và viết các interface.
* Tiếp cận lập trình khai báo bằng cách sử dụng các quy tắc (convention) và các khía cạnh (aspect) chung.
* Giảm thiểu các mã nghi thức và soạn sẵn (boilerplate) thông qua việc sử dụng các khuôn mẫu (template) và các khía cạnh.

2.2.2. Tổng quan về Spring

* Spring Core: Spring Core chính là thành phần trung tâm, cốt lõi của Spring Framework. Đây chính là nền tảng để xây dựng nên các thành phần khác.
* Spring Bean: Spring Bean là trung tâm của Spring Core và là trái tim của một ứng dụng Spring.
* Dependency Injection (DI): Dependency Injection (có thể dịch tiêm các thành phần phụ thuộc) là một sức mạnh nổi bật của Spring Framework.
* Spring Context: Spring Context mang mọi thứ lại với nhau.
* Spring Expression Language (SpEL): Spring Expression Language là một ngôn ngữ ngắn gọn giúp cho việc cấu hình Spring Framework trở nên linh hoạt hơn.
* Các dự án trong Spring Framework: Spring Framework là một tập hợp của nhiều dự án con.
* Spring MVC: Spring MVC được thiết kế dành cho việc xây dựng các ứng dụng nền tảng web.
* Spring Data: Cung cấp một cách tiếp cận đúng đắn để truy cập dữ liệu từ cơ sở dữ liệu quan hệ, phi quan hệ, map-reduce, …
* Spring Security: Dự án này cung cấp các cơ chế xác thực (authentication) và phân quyền (authorization) cho ứng dụng.
* Spring Boot: là một framework giúp phát triển cũng như chạy ứng dụng một cách nhanh chóng.
* Spring Batch: tạo các lịch trình (scheduling) và tiến trình (processing).
* Spring Integration: là một implementation của Enterprise Integration Patterns (EIP).
* Spring XD: đơn giản hóa công việc phát triển các ứng dụng Big Data.
* Spring Social: kết nối ứng dụng của bạn với các API bên thứ ba của Facebook, Twitter, Linkedin, ...

Hình 2.2.2 : Mô hình Spring Framework Runtime

2.3. Tổng quan về hệ quản trị cơ sở dữ liệu MySQL

* MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS, …
* MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).[3]

2.4. Restful API (RestController Spring Boot)

* Khác với @Controller là sẽ trả về một template. @RestController trả về dữ liệu dưới dạng JSON. Các đối tượng trả về dưới dạng Object sẽ được Spring Boot chuyển thành JSON.
* Các đối tượng trả về rất đa dạng, bạn có thể trả về List, Map, … Spring Boot sẽ convert hết chúng thành JSON, mặc định sẽ dùng Jackson converter để làm điều đó. Nếu bạn muốn API tùy biến được kiểu dữ liệu trả về, bạn có thể trả về đối tượng ResponseEntity của Spring cung cấp. Đây là đối tượng cha của mọi response và sẽ wrapper các object trả về.
* Vì xây dựng API, nên các thông tin từ phía Client gửi lên Server sẽ nằm trong Body, và cũng dưới dạng JSON luôn. Tất nhiên là Spring Boot sẽ làm giúp chúng ta các phần nặng nhọc, nó chuyển chuỗi JSON trong request thành một Object Java. bạn chỉ cần cho nó biết cần chuyển JSON thành Object nào bằng Annotation @RequestBody
* Ngoài thông tin trong Body của request, thì cái chúng ta cần chính là cái con số 12 nằm trong URL. Phải lấy được con số đó thì mới biết được đối tượng To-do cần thao tác là gì. Lúc đó Anotation @PathVariable tham chiến. [4]

2.5. Bootstrap và responsive

Bootstrap là front-end framework, là một bộ sưu tập miễn phí các công cụ để tạo ra các trang web và các ứng dụng web. Bootstrap bao gồm HTML và CSS dựa trên các mẫu thiết kế cho kiểu chữ, hình thức, các button và các thành phần giao diện khác, cũng như mở rộng tùy chọn JavaScript. Boostrap định nghĩa sẵn các class CSS giúp người thiết kế giao diện website tiết kiệm rất nhiều thời gian. Các thư viện Bootstrap có những đoạn mã sẵn sàng cho chúng ta áp dùng vào website của mình mà không phải tốn quá nhiều thời gian để tự viết. Với Bootstrap, việc phát triển giao diện website để phù hợp với đa thiết bị trở nên dễ dàng hơn bao giờ hết. Bootstrap cung cấp tính năng responsive và mobile first, nghĩa là làm cho trang web có thể tự co giãn để tương thích với mọi thiết bị khác nhau, từ điện thoại di động đến máy tính bảng, máy tính xách tay, máy tính để bàn,... Một khía cạnh khác là responsive web design làm cho trang web cung cấp được trải nghiệm tuyệt vời cho người dùng trên nhiều thiết bị, kích thước màn hình khác nhau. Một trang có thể hoạt động tốt bất kể sự biến đổi sẽ cung cấp một trải nghiệm người dùng tốt và nhất quán hơn một trang được thiết kế cho một loại thiết bị và kích thước màn hình cụ thể.[5]

2.6. Typescript và Angular Framework

2.6.1. TypeScript

* TypeScript (TS) là một superset của JavaScript (JS), được phát triển bởi Microsoft. Có thể transpile thành code JS để chạy trên môi trường của Browser hoặc Nodejs.
* Tuân thủ chặt chẽ specs mà ECMAScript (ES) đề ra, do đó tất cả những đoạn code hợp lệ trong JS thì sẽ hợp lệ ở TS.
* Support rất nhiều tính năng nâng cao trong các bản ES mới nhất.
* TypeScript đang được hỗ trợ rất mạnh, từ cộng đồng, IDE/Editor, đến các Library/Framework.

2.6.2. Angular

* Angular được xe là một open source (mã nguồn mở) hay frameworks miễn phí chuyên dụng cho công việc thiết kế web. Angular được phát triển từ những năm 2009 và được duy trì bởi Google. Frameworks này được xem là frameworks front end mạnh mẽ nhất chuyên dụng bởi các lập trình viên cắt HTML cao cấp.[6]
* Angular được ứng dụng rộng rãi với mục đích xây dựng project Single Page Application (SPA). Hiện tại, Version stable của Angular là Angular 9 (released on February 7, 2020) với TypeScript 3.6 và 3.7.
* Kiến trúc Angular:
* Component sử dụng data binding để lấy dữ liệu từ Component sang View (template). Chúng ta sử dụng HTML đặc biệt được biết đến là cú pháp Angular Template.
* Ở bên phải chúng ta có Angular Service. Angular Service cung cấp các service cho Component của chúng ta giống như lấy dữ liệu ra từ database sử dụng TaskService, các sự kiện logging trong ứng dụng sử dụng LoggerService và tạo request HTTP đến backend server sử dụng HTTP Service.
* Trách nhiệm cung cấp các thể hiện của Service cho Component gọi là Angular Injector. Nó inject các service vào component sử dụng Dependency Injection.
* Chúng ta có các Directive, giúp chúng ta thao tác với cấu trúc và style trong ứng dụng. Các directive giúp chúng ta transform DOM theo yêu cầu.
* Component: Trong một trang web đc xây dựng bằng angular, thì chia thành các khối hiển thị và mỗi khối có code xử lý business riêng, mỗi thành phần như vậy được gọi là component. Các thành phần: template + class|properties|method + metadata. Component gồm có: selector, templateUrl, styleUrl
* Template: Component cần một view để hiển thị. Template định nghĩa view. Template chỉ là một tập con của HTML, nó chỉ cho Angular biết làm sao để hiển thị view. Nó là một trang HTML chuẩn sử dụng các thẻ H1, H2 …
* Directive: Là thành phần trong ứng dụng Angular, được dung để tang sức mạnh và mở rộng tính năng cho HTML, Directive giúp người dung thay đổi cấu trúc của thuộc tính của thẻ HTML.
* Module: Angular App được chia thành các Module, chúng được gọi là NgModule. Trong NgModule có thể bao gồm: Component, Pipe, Directive, Service.

CHƯƠNG III. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1. Khảo sát hệ thống.

3.1.1. Khảo sát thực tế

Hiện nay, nhu cầu giải trí xem phim rạp tăng cao, việc quản lý đặt vé cho khách hàng và quản lý rạp chiếu trở nên khó khăn, vì vậy để phục vụ cho nhu cầu đặt vé xem phim, cập nhật thông tin của những bộ phim mà mình yêu thích và giúp cho việc quản lý rạp của chủ rạp trở nên dễ dàng thì việc thành lập một website quản lý rạp phim là hết sức cần thiết.

3.1.2. Khảo sát người dùng

Nhu cầu giải trí xem phim ngày càng cao, nhu đặt vé và tìm kiếm thông tin phim ngày càng tăng, khiến cho khách hàng rất mất thời gian và công sức cho việc đặt vé những bộ phim mình yêu thích.

3.2. Đặc tả yêu cầu phần mềm

3.2.1. Xác định các tác nhân

Đề tài gồm ba tác nhân chính

* Người dùng (User)
* Nhân viên (Employee)
* Người quản lý (Admin)

3.2.2. Các yêu cầu chức năng

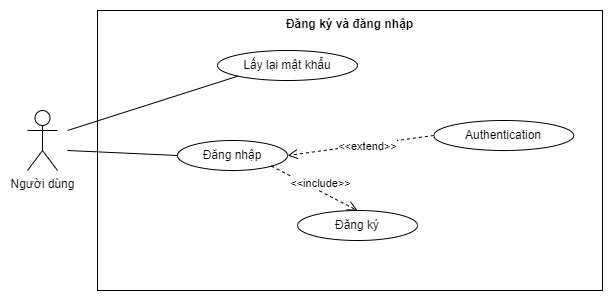
* User
* Xem danh sách phim, xem chi tiết thông tin phim.
* Tìm kiếm phim (lọc theo tên, thể loại, ngày chiếu,…).
* Xem giá vé, lịch chiếu, khuyến mãi.
* Đặt vé phim.
* Quản lý hồ sơ cá nhân.
* Xem lịch sử đặt vé.
* Đăng nhập, đăng xuất, đổi mật khẩu.
* Employee
* Xem danh sách thành viên.
* Tìm kiếm thành viên.
* Quản lý đặt vé.
* Quản lý bán vé.
* Quản lý khuyến mãi.
* Quản lý hồ sơ cá nhân.
* Đăng nhập, đăng xuất, đổi mật khẩu.
* Admin (gồm các chức năng của employee)
* Thống kê.
* Quản lý thành viên.
* Quản lý nhân viên.
* Quản lý phòng chiếu.
* Quản lý suất chiếu

3.2.3. Yêu cầu phi chức năng

* Giao diện, đẹp mắt, đơn giản, dễ sử dụng.
* Đảm bảo tính bảo mật, an toàn.
* Tốc độ xử lý nhanh chóng.

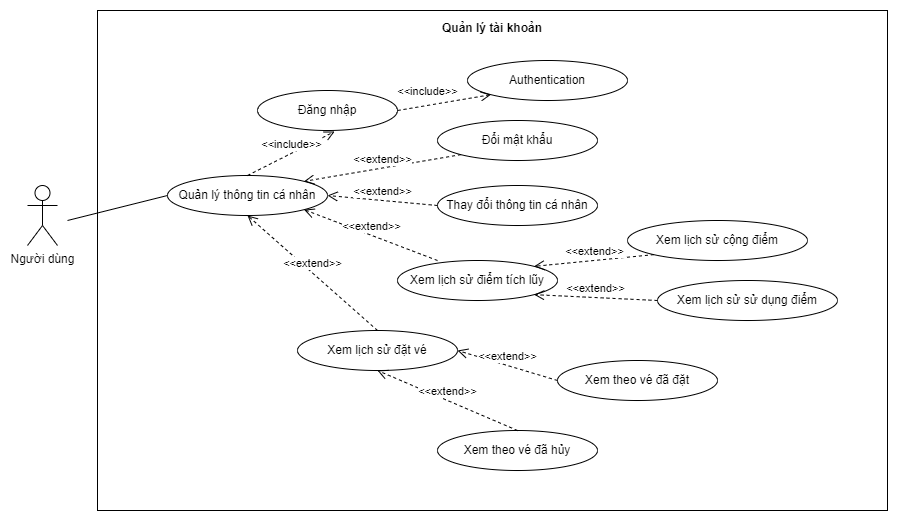
3.3. Biểu đồ Use Case

3.3.1 Đăng ký và đăng nhập



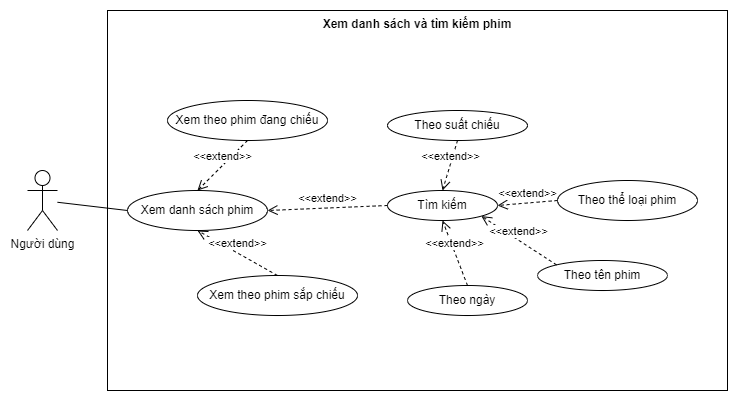
Hình 3.3.1 Use case đăng ký và đăng nhập

3.3.2 Quản lý tài khoản



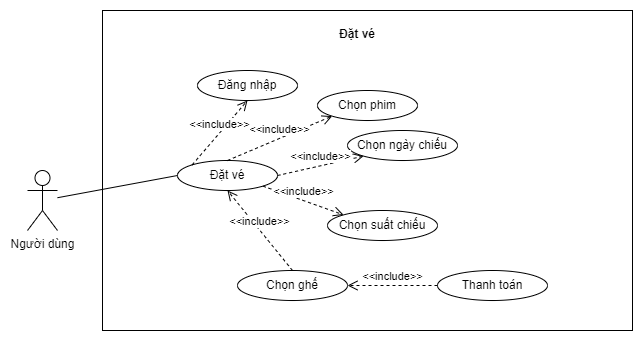
Hình 3.3.2 Use case quản lý tài khoản

3.3.3 Xem danh sách phim



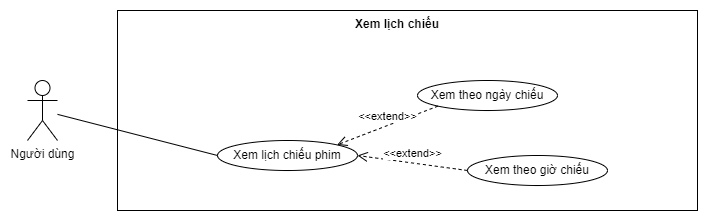
Hình 3.3.3 Use case xem danh sách và tìm kiếm phim

3.3.4 Đặt vé



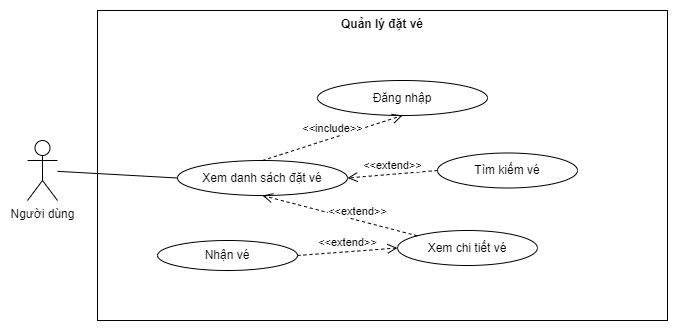
Hình 3.3.4 Use case đặt vé

3.3.5 Xem lịch chiếu



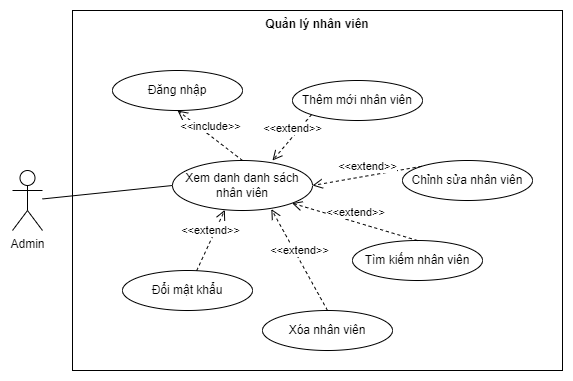
Hình 3.3.5 Use case xem lịch chiếu

3.3.6 Quản lý đặt vé



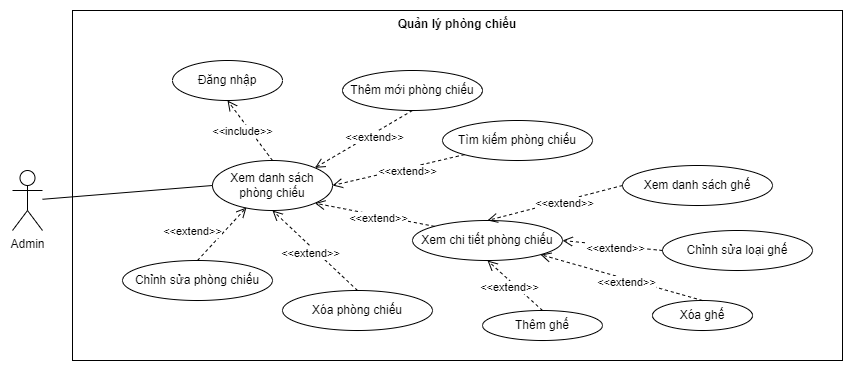
Hình 3.3.6 Use case quản lý đặt vé

3.3.7 Quản lý nhân viên



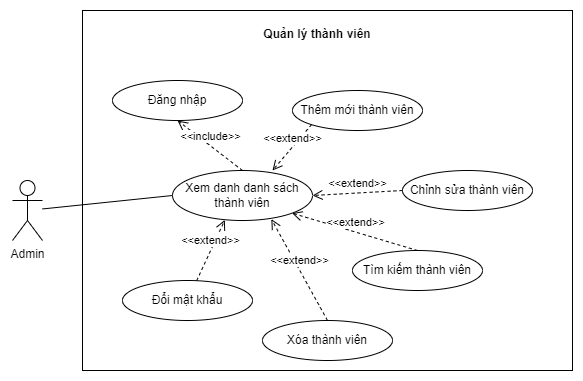
Hình 3.3.7 Use case quản lý nhân viên

3.3.8 Quản lý phòng chiếu



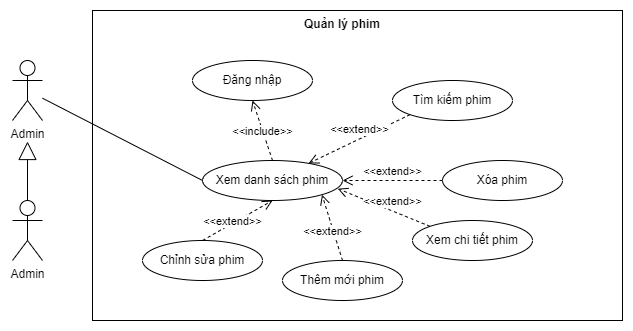
Hình 3.3.8 Use case quản lý phòng chiếu

3.3.9 Quản lý thành viên



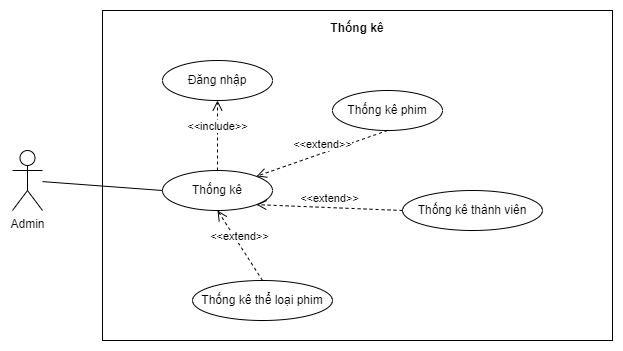
Hình 3.3.9 Use case quản lý thành viên

3.3.10 Quản lý phim



Hình 3.3.10 Use case quản lý phim

3.3.11 Thống kê



Hình 3.3.11 Use case thống kê

3.4. Scenario cho Use Case

3.4.1 Scenario Use-case “Đăng ký và đăng nhập”

Bảng 3.4.1 Scenario Use-case “Đăng ký và đăng nhập”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Đăng nhập |
|  | Description | Người dùng đã có tài khoản muốn đăng nhập vào hệ thống |
|  | Actors | User/Employee/Admin |
|  | Input | User/Employee/Admin đã có tài khoản  Thông tin tài khoản |
|  | Output | Hệ thống thông báo đăng nhập hệ thống thành công |
|  | Basic flow | 1. Actor vào ứng dụng 🡺 Bắt đầu use case 2. Actor nhập thông tin tài khoản (tên tài khoản, mật khẩu), hoặc chọn đăng nhập với Google 3. Actor nhấn nút đăng nhập 4. Hệ thống kiểm tra và trả về trang Trang chủ (đã đăng nhập) 🡺 Kết thúc use case |
|  | Alternative flow |  |
|  | Exception flow | 2.1. Actor thoát khỏi hệ thống 🡺 Use case kết thúc mà chưa được đăng nhập  3.1. Nhập sai tài khoản hoặc mật khẩu 🡺 Thông báo không tìm thấy tài khoản hoặc sai mật khẩu 🡺Quay lại bước 2 |

3.4.2 Scenario Use-case “Quản lý tài khoản”

Bảng 3.4.2 Scenario Use-case “Quản lý tài khoản”

|  |  |  |
| --- | --- | --- |
| **STT** | **Use case name** | **Quản lý tài khoản** |
|  | **Description** | Dùng để quản lý hồ sơ cá nhân của người dùng |
|  | **Actors** | User/Employee/Admin |
|  | **Input** | Actor đã đăng nhập thành công |
|  | **Output** | Chỉnh sửa thông tin thành công |
|  | **Basic flow** | 1. Actor nhấn vào “Tên User” ở thanh header của hệ thống sau đó nhấn “Hồ sơ cá nhân” 🡺 Use case bắt đầu 2. Thực hiện các thao tác cần thiết (Chỉnh sửa hồ sơ cá nhân hoặc Đổi mật khẩu) 3. Actor nhấn nút “Lưu” 4. Hệ thống lưu lại thông tin của người dùng sau khi thay đổi 🡺 Kết thúc Use case |
|  | **Alternative flow** |  |
|  | **Exception flow** | 2.1. Nếu Actor cập nhật thông tin không đúng định dạng 🡺 Thông báo thông tin cập nhật không hợp lệ  2.3. Nếu người dùng nhập mật khẩu cũ không khớp 🡺 Thông báo mật khẩu cũ không đúng |

3.4.3 Scenario Use-case “Xem danh sách và tìm kiếm phim”

Bảng 3.4.3 Scenario Use-case “Xem danh sách và tìm kiếm phim”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Xem danh sách và tìm kiếm phim |
|  | Description | Người dùng muốn xem danh sách các phim tại rạp hoặc tìm kiếm phim |
|  | Actors | Người dùng |
|  | Input | Người dùng truy cập vào trang chủ của hệ thống |
|  | Output | Hệ thống hiển thị danh sách các phim |
|  | Basic flow | 1. Actor truy cập hệ thống 🡺 Bắt đầu use case 2. Actor xem danh sách các phim trong trang chủ hoặc nhập các thông tin (Suất chiếu, Ngày chiếu, Tên, Thể loại) để tìm kiếm phim 3. Actor nhấn nút tìm kiếm 4. Hệ thống kiểm tra và trả kết quả theo yêu cầu tìm kiếm 🡺 Kết thúc use case |
|  | Alternative flow | 2.1. Actor chọn xem danh sách phim theo “Phim đang chiếu” hoặc “Phim sắp chiếu” 🡺 Hệ thống trả về danh sách phim phù hợp |
|  | Exception flow | 3.1. Actor nhập thông tin tìm kiếm không hợp lệ hoặc không có dữ liệu 🡺 Hệ thống trả về danh sách rỗng |

3.4.4 Scenario Use-case “Đặt vé”

Bảng 3.4.4 Scenario Use-case “Đặt vé”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Đặt vé |
|  | Description | Người dùng muốn đặt vé xem phim |
|  | Actors | User/Employee/Admin |
|  | Input | Người dùng đã đăng nhập vào hệ thống |
|  | Output | Hệ thống hiển thị đặt vé thành công |
|  | Basic flow | 1. Actor chọn “Tên phim” muốn đặt vé sau đó chọn “Ngày chiếu” và “Suất chiếu” 🡺 Bắt đầu use case 2. Actor chọn “Ghế ngồi” sau đó tiến hành “Thanh toán” 3. Actor nhập thông tin thanh toán và nhấn vào nút “Xác nhận thanh toán” 4. Hệ thống kiểm tra và trả về thông báo đặt vé thành công 🡺 Kết thúc use case |
|  | Alternative flow |  |
|  | Exception flow | 3.1 Actor nhấn hủy 🡺 Use case kết thúc |

3.4.5 Scenario Use-case “Xem lịch chiếu phim”

Bảng 3.4.5 Scenario Use-case “Xem lịch chiếu phim”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Xem lịch chiếu phim |
|  | Description | Dùng để hiển thị danh sách các phim được chiếu trong ngày giờ xác định |
|  | Actors | Người dùng |
|  | Input | Người dùng truy cập vào hệ thống |
|  | Output | Danh sách các phim được chiếu vào ngày giờ đã nhập |
|  | Basic flow | 1. Actor nhấn vào menu “Lịch chiếu” trên thanh header 🡺 Bắt đầu use case 2. Actor chọn “Ngày chiếu” và “Giờ chiếu” 3. Actor nhấn vào nút “Xem lịch chiếu” 4. Hệ thống hiển thị ra danh sách các phim sẽ được chiếu trong khung giờ đã nhập 🡺 Use case kết thúc |
|  | Alternative flow |  |
|  | Exception flow | 2.1. Người dùng không nhập đủ thông tin 🡺 Hiển thị thông báo “Vui lòng nhập đầy đủ thông tin” 🡺 Quay lại bước 2 |

3.4.6 Scenario Use-case “Quản lý đặt vé”

Bảng 3.4.6 Scenario Use-case “Quản lý đặt vé”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Quản lý đặt vé |
|  | Description | Cho phép người dùng xem thông tin về các vé đã đặt |
|  | Actors | User/Employee/Admin |
|  | Input | Actor đăng nhập thành công |
|  | Output | Hiển thị lịch sử đặt vé của thành viên sử dụng |
|  | Basic flow | 1. Actor nhấn vào “Tên User” trên thanh header sau đó chọn “Quản lý hồ sơ” 🡺 Bắt đầu use case 2. Nhấn vào menu “Quản lý đặt vé” ở side-bar 3. Hệ thống sẽ hiển thị ra lịch sử đặt vé của người dùng 🡺 Kết thúc Use case |
|  | Alternative flow | 3.1. Actor nhấn xem chi tiết vé 🡺 Hệ thống sẽ hiển thị thông tin chi tiết của vé đó |
|  | Exception flow |  |

3.4.7 Scenario Use-case “Quản lý nhân viên”

Bảng 3.4.7 Scenario Use-case “Quản lý nhân viên”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Quản lý nhân viên |
|  | Description | Dùng để quản lý danh sách các nhân viên của rạp |
|  | Actors | Admin |
|  | Input | Actor đã đăng nhập thành công với quyền admin |
|  | Output | Hiển thị danh sách các nhân viên |
|  | Basic flow | 1. Actor nhấn vào menu “Quản lý nhân viên” ở thanh side-bar 🡺 Bắt đầu use case 2. Actor thực hiển các thao tác cần thiết (xem chi tiết, thêm, sửa, xóa, tìm kiếm) đối với nhân viên 3. Hệ thống ghi nhận và cập nhật lại các chỉnh sửa của nhân viên nếu có 🡺 Kết thúc Use case |
|  | Alternative flow |  |
|  | Exception flow |  |

3.4.8 Scenario Use-case “Quản lý thành viên”

Bảng 3.4.8 Scenario Use-case “Quản lý thành viên”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Phản hồi |
|  | Description | Dùng để quản lý danh sách các thành viên của rạp |
|  | Actors | Employee/Admin |
|  | Input | Actor đặng nhập thành công với quyền Employee hoặc Admin |
|  | Output | Hiển thị danh sách thành viên của rạp |
|  | Basic flow | 1. Actor nhấn vào menu “Quản lý thành viên” ở thanh side-bar 🡺 Bắt đầu use case 2. Actor thực hiển các thao tác cần thiết (xem chi tiết, thêm, sửa, xóa, tìm kiếm) đối với thành viên 3. Hệ thống ghi nhận và cập nhật lại các chỉnh sửa của thành viên nếu có 🡺 Kết thúc Use case |
|  | Alternative flow |  |
|  | Exception flow | 2.1. Nếu Actor có quyền là Employee thì chỉ cho phép actor thực hiện các chức năng “Xem chi tiết”, “Tìm kiếm” thành viên mà thôi |

3.4.9 Scenario Use-case “Quản lý phim”

Bảng 3.4.9 Scenario Use-case “Quản lý phim”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Quản lý phim |
|  | Description | Dùng để quản lý danh sách phim của rạp |
|  | Actors | Employee/Admin |
|  | Input | Actor đăng nhập thành công với quyền Admin hoặc Employee |
|  | Output | Hiển thị danh sách phim của rạp |
|  | Basic flow | 1. Actor nhấn vào menu “Quản lý thành phim” ở thanh side-bar 🡺 Bắt đầu use case 2. Actor thực hiển các thao tác cần thiết (xem chi tiết, thêm, sửa, xóa, tìm kiếm) đối với phim 3. Hệ thống ghi nhận và cập nhật lại các chỉnh sửa của phim nếu có 🡺 Kết thúc Use case |
|  | Alternative flow |  |
|  | Exception flow | 2.1. Nếu Actor có quyền là Employee thì chỉ cho phép actor thực hiện các chức năng “Xem chi tiết”, “Tìm kiếm” phim mà thôi |

3.4.10 Scenario Use-case “Quản lý phòng chiếu”

Bảng 3.4.10 Scenario Use-case “Quản lý phòng chiếu”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Quản lý phòng chiếu |
|  | Description | Dùng để quản lý các phòng chiếu phim của rạp |
|  | Actors | Admin |
|  | Input | Actor đã đăng nhập thành công với quyền admin |
|  | Output | Hiển thị danh sách các phòng chiếu |
|  | Basic flow | 1. Actor nhấn vào menu “Quản lý phòng chiếu” ở thanh side-bar 🡺 Bắt đầu use case 2. Actor thực hiển các thao tác cần thiết (xem chi tiết, thêm, sửa, xóa, tìm kiếm) đối với phòng chiếu 3. Hệ thống ghi nhận và cập nhật lại các chỉnh sửa của phòng chiếu nếu có 🡺 Kết thúc Use case |
|  | Alternative flow | 2.1. Actor chọn “Xem chi tiết phòng chiếu” sau đó chọn “Quản lý ghế”  2.1.1. Actor thực hiện các thao tác cần thiết với ghế (thêm, sửa, xóa) của phòng chiếu  2.1.2. Hệ thống ghi nhận và cập nhật lại những thay đổi của ghế của phòng chiếu 🡺 Kết thúc Use case |
|  | Exception flow |  |

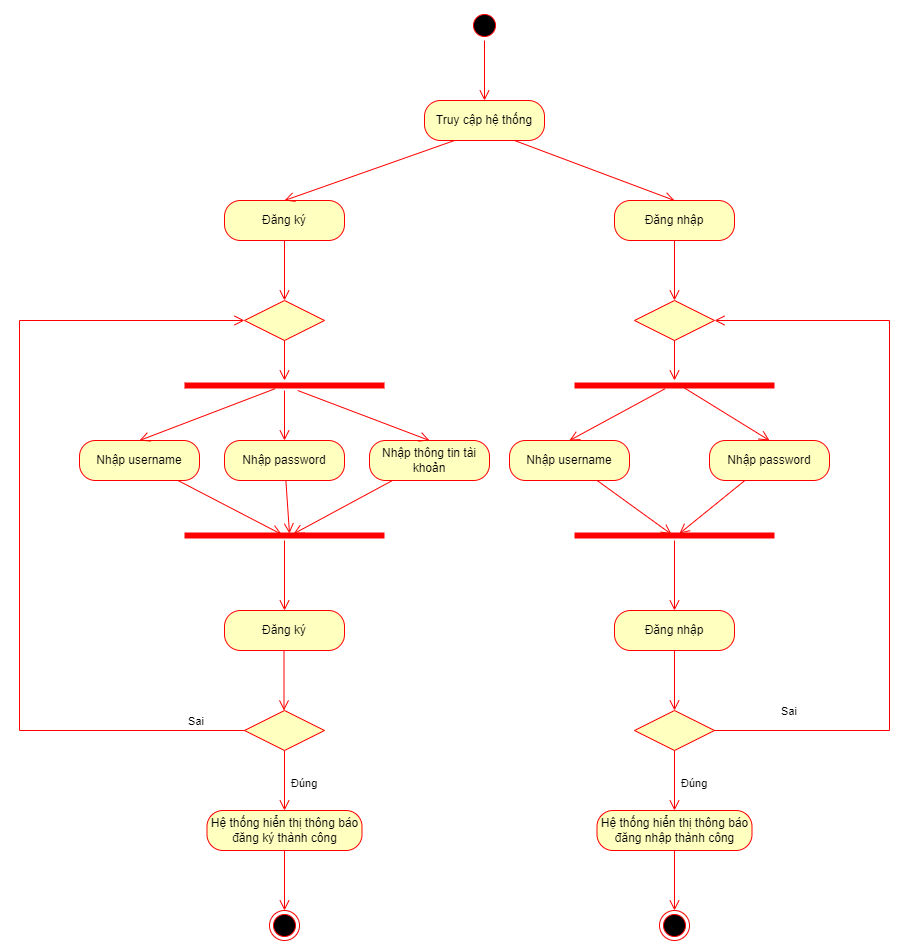
3.4.11 Scenario Use-case “Thống kê”

Bảng 3.4.11 Scenario Use-case “Thống kê”

|  |  |  |
| --- | --- | --- |
| STT | Use case name | Thống kê |
|  | Description | Thống kê các thông tin của rạp |
|  | Actors | Admin |
|  | Input | Actor đăng nhập thành công với quyền Admin |
|  | Output | Hiển thị bảng thống kê thông tin |
|  | Basic flow | 1. Actor nhấn vào menu “Thống kê” ở thanh side-bar 🡺 Bắt đầu use case 2. Actor chọn thông tin muốn thống kê, ngày bắt đầu và ngày kết thúc 3. Hệ thống sẽ tính toán và hiển thị số liệu thống kê 🡺 Kết thúc Use case |
|  | Alternative flow |  |
|  | Exception flow | 3.1. Nếu không tìm thấy số liệu thống kê 🡺 Hệ thống hiển thị số liệu rỗng |

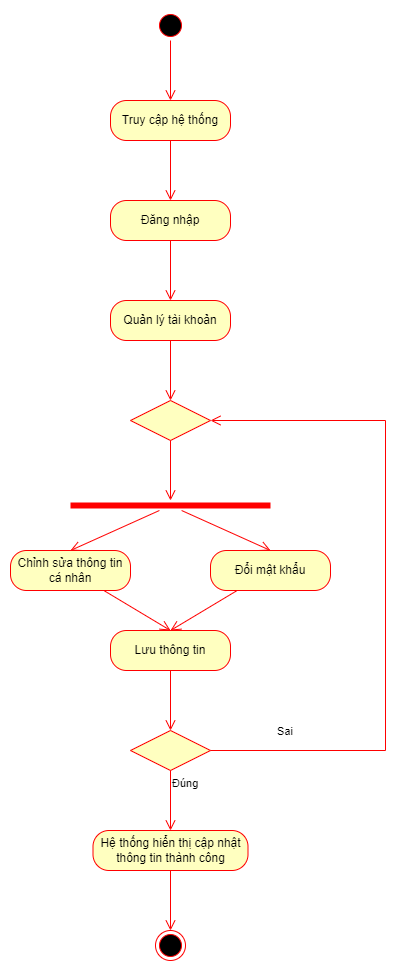
3.5. Sơ đồ hoạt động

3.5.1 Đăng ký và đăng nhập



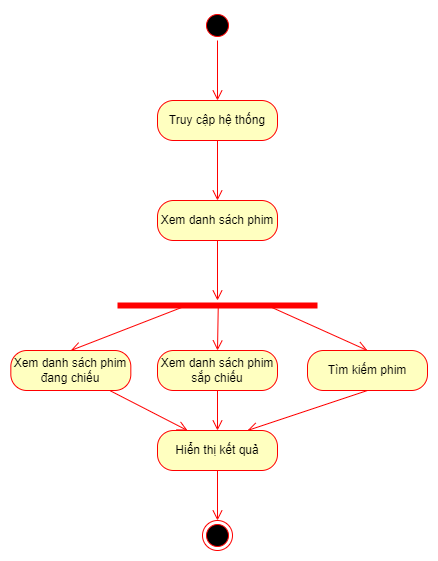
Hình 3.5.1 Activity “Đăng ký và đăng nhập”

* + 1. Quản lý tài khoản



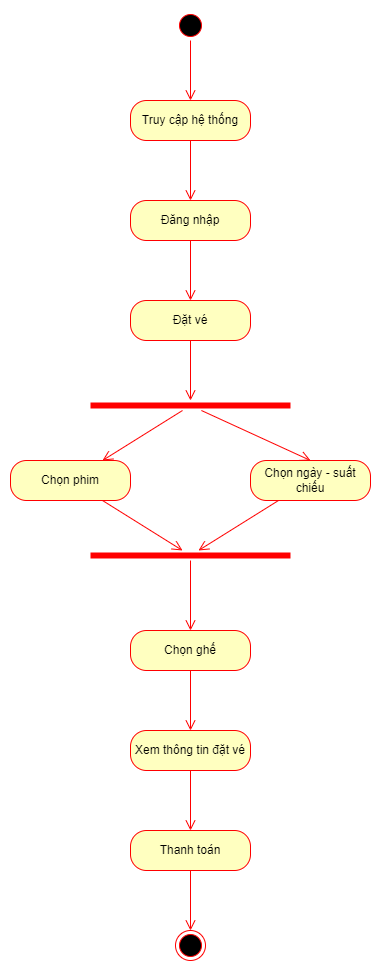
Hình 3.5.2 Activity “Quản lý tài khoản”

3.5.3 Xem và tìm kiếm phim



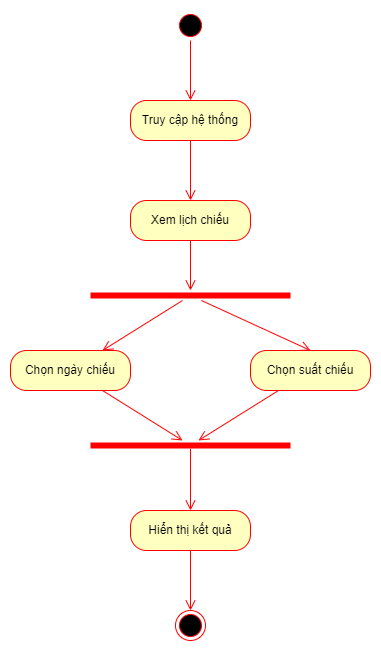
Hình 3.5.3 Activity “Xem và tìm kiếm phim”

3.5.4 Đặt vé



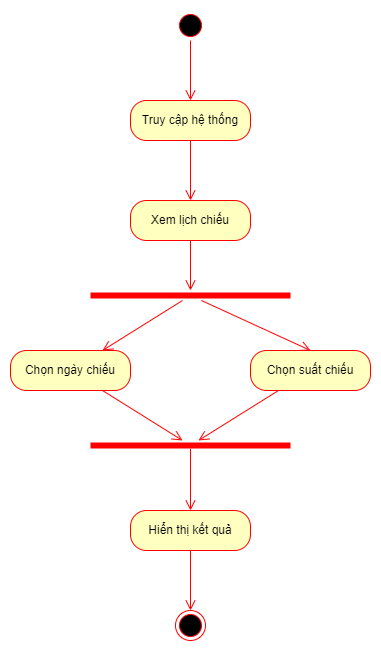
Hình 3.5.4 Activity “Đặt vé”

3.5.5 Xem lịch chiếu



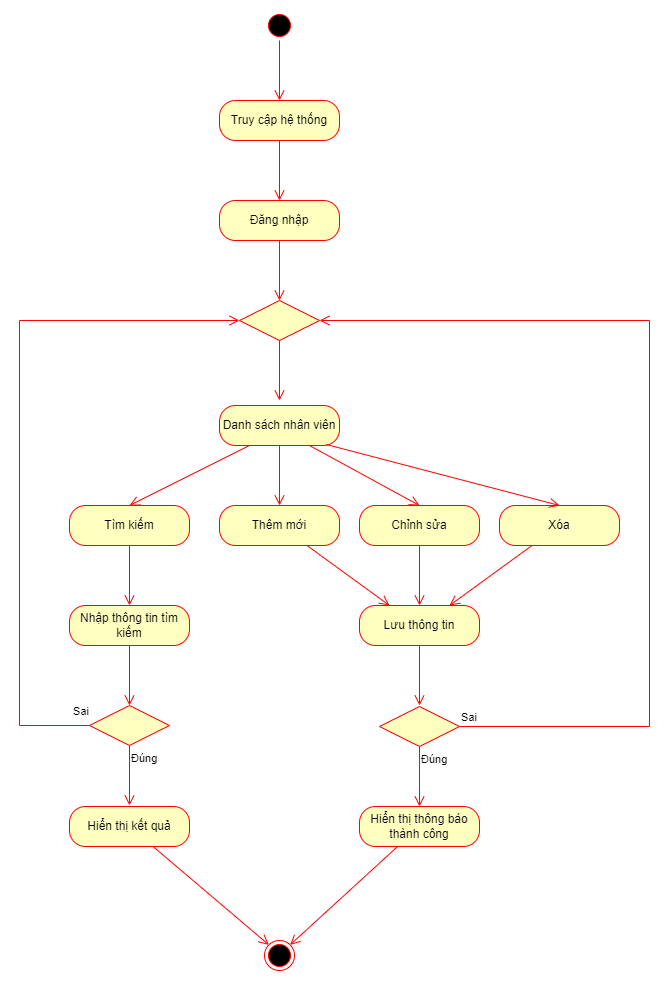
Hình 3.5.5 Activity “Xem lịch chiếu”

3.5.6 Quản lý đặt vé



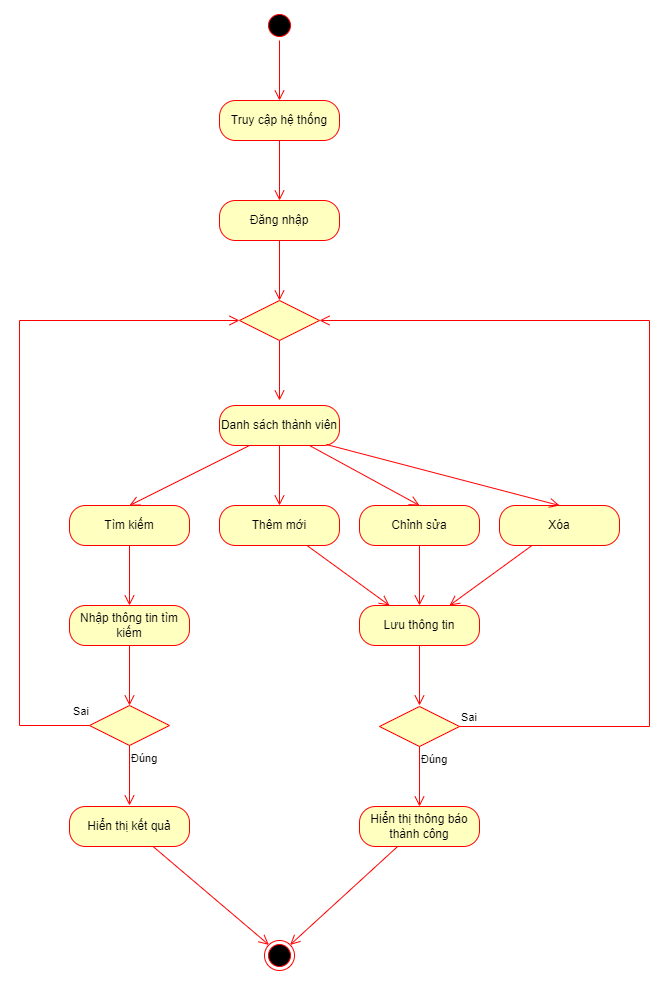
Hình 3.5.6 Activity “Quản lý đặt vé”

3.5.7 Quản lý nhân viên



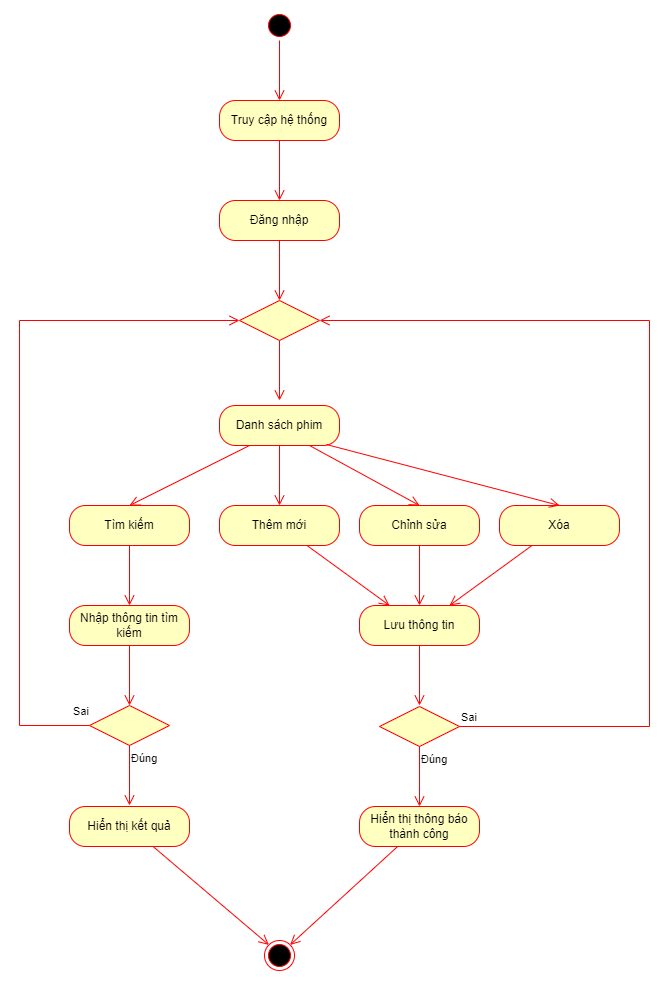
Hình 3.5.7 Activity “Quản lý nhân viên”

3.5.8 Quản lý thành viên



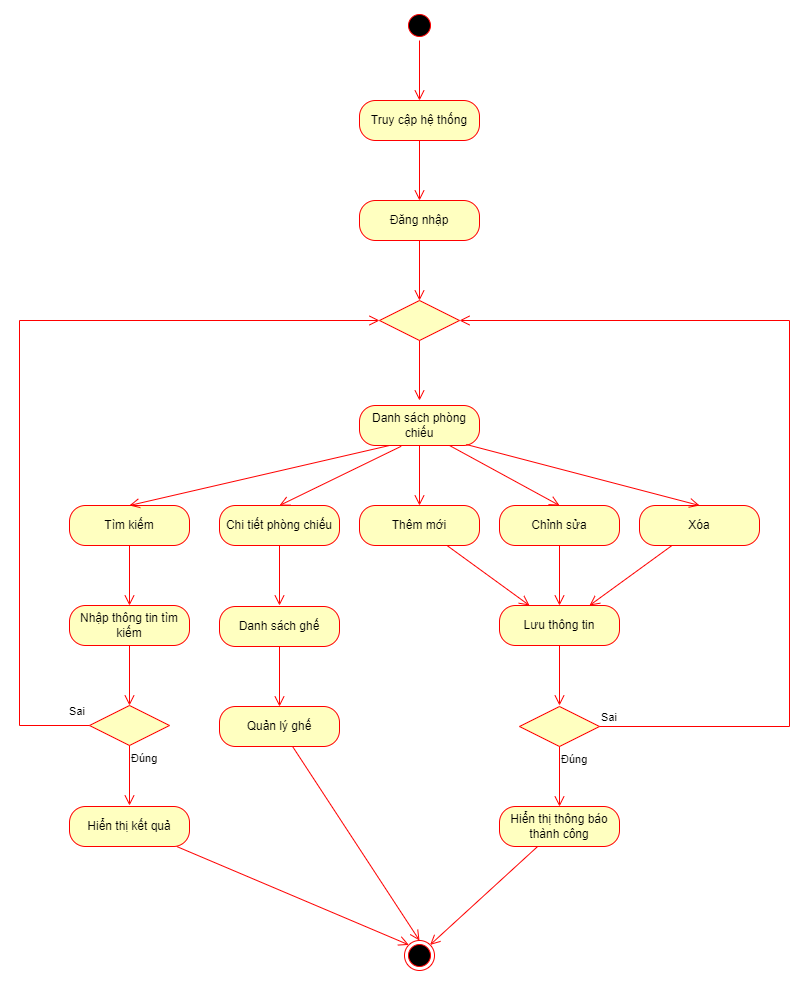
Hình 3.5.8 Activity “Quản lý thành viên”

3.5.9 Quản lý phim



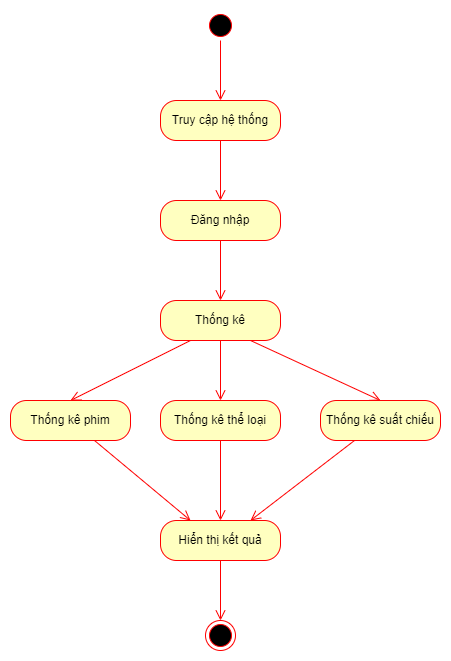
Hình 3.5.9 Activity “Quản lý phim”

3.5.10 Quản lý phòng chiếu



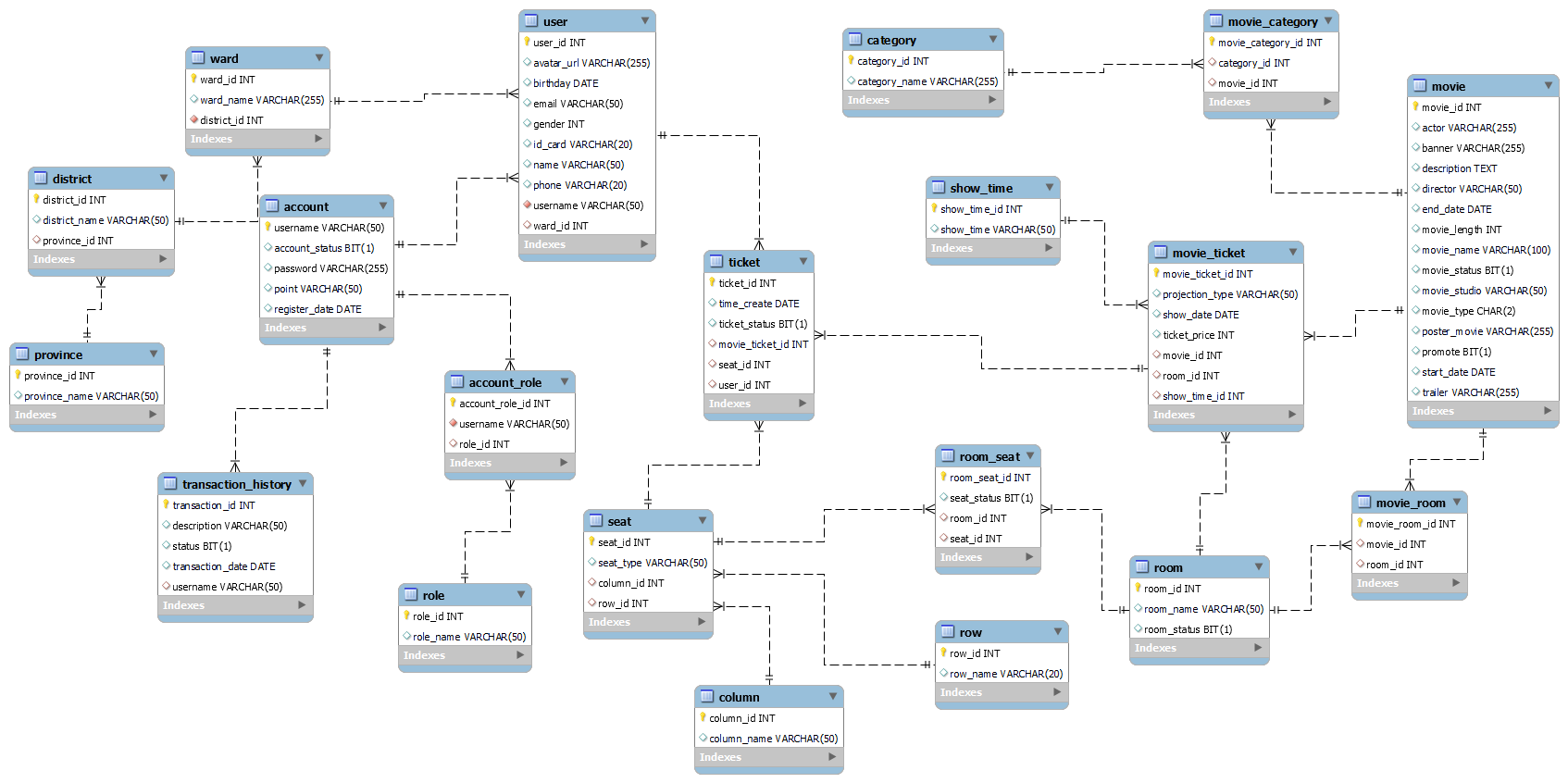
Hình 3.5.10 Activity “Quản lý phòng chiếu”

3.5.11 Thống kê



Hình 3.5.11 Activity “Thống kê”

3.6. Sơ đồ ERD



Hình 3.6 Sơ đồ ERD

3.7. Thiết kế bảng cơ sở dữ liệu

3.7.1 Bảng role

Bảng 3.7.1 role

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| role\_id | Mã quyền | INT | Khóa chính tự tăng |
| role\_name | Tên quyền | VARCHAR(50) |  |

3.7.2 Bảng account\_role

Bảng 3.7.2 account\_role

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| account\_role\_id | Mã tài quyền tài khoản | INT | Khóa chính |
| user\_name | Tên tài khoản | VARCHAR(50) | Khóa ngoại |
| role\_id | Mã quyền | INT | Khóa ngoại |

3.7.3 Bảng account

Bảng 3.7.3 account

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| username | Tên tài khoản | VARCHAR(50) | Khóa chính |
| account\_status | Trạng thái tài khoản | BIT(1) |  |
| password | Mật khẩu | VARCHAR(255) |  |
| point | Điểm tích lũy | VARCHAR(50) |  |
| register\_date | Ngày đăng ký | DATE |  |

3.7.4 Bảng transaction\_history

Bảng 3.7.4 transaction\_history

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| transaction\_id | Mã giao dịch | INT | Khóa chính tự tăng |
| description | Mô tả | VARCHAR(50) |  |
| status | Trạng thái | BIT(1) |  |
| transaction\_date | Ngày giao dịch | DATE |  |
| username | Tên tài khoản | VARCHAR(50) | Khóa ngoại |

3.7.5 Bảng province

Bảng 3.7.5 province

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| province\_id | Mã tỉnh | INT | Khóa chính tự tăng |
| province\_name | Tên tỉnh | VARCHAR(50) |  |

3.7.6 Bảng district

Bảng 3.7.6 district

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| district\_id | Mã quận/huyện | INT | Khóa chính tự tăng |
| district\_name | Tên quận/huyện | VARCHAR(50) |  |
| province\_id | Mã tỉnh | INT | Khóa ngoại |

3.7.7 Bảng ward

Bảng 3.7.7 ward

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| ward\_id | Mã phường/xã | INT | Khóa chính tự tăng |
| ward\_name | Tên tỉnh | VARCHAR(50) | Khóa ngoại |

3.7.8 Bảng user

Bảng 3.7.8 property

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| user\_id | Mã người dùng | INT | Khóa chính tự tăng |
| avatar\_url | Ảnh đại diện | VARCHAR(255) |  |
| birthday | Ngày sinh | DATE |  |
| email | Email | VARCHAR(50) |  |
| gender | giới tính | INT |  |
| id\_card | Số chứng minh thư | VARCHAR(20) |  |
| name | Tên người dùng | VARCHAR(50) |  |
| phone | Số điện thoại | VARCHAR(20) |  |
| username | Tên tài khoản | VARCHAR(50) | Khóa ngoại |
| ward\_id | Mã quận/huyện | INT | Khóa ngoại |

3.7.9 Bảng column

Bảng 3.7.9 column

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| column\_id | Mã cột | INT | Khóa chính tự tăng |
| column\_name | Tên cột | VARCHAR(50) |  |

3.7.10 Bảng row

Bảng 3.7.10 row

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| row\_id | Mã hàng | INT | Khóa chính tự tăng |
| row\_name | Tên hàng | VARCHAR(50) |  |

3.7.11 Bảng seat

Bảng 3.7.11 seat

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| seat\_id | Mã ghế | INT | Khóa chính tự tăng |
| seat\_type | Loại ghế | VARCHAR(50) |  |
| column\_id | Mã cột | INT | Khóa ngoại |
| row\_id | Mã hàng | INT | Khóa ngoại |

3.7.12 Bảng category

Bảng 3.7.12 category

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| category\_id | Mã thể loại | INT | Khóa chính tự tăng |
| category\_name | Tên thể loại | VARCHAR(255) |  |

3.7.13 Bảng movie

Bảng 3.7.13 movie

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| movie\_id | Mã phim | INT | Khóa chính tự tăng |
| actor | Diễn viên | VARCHAR(255) |  |
| banner | Ảnh banner | VARCHAR(255) |  |
| description | Mô tả phim | TEXT |  |
| director | Đạo diễn | VARCHAR(50) |  |
| end\_date | Ngày bắt đầu | DATE |  |
| movie\_length | Thời lượng | INT |  |
| movie\_name | Tên phim | VARCHAR(100) |  |
| movie\_status | Trạng thái | BIT(1) |  |
| movie\_studio | Studio làm phim | VARCHAR(50) |  |
| movie\_type | Loại chiếu | CHAR(2) |  |
| poster\_movie | Ảnh poster | VARCHAR(255) |  |
| promote | Phim hot | BIT(1) |  |
| start\_date | Ngày bắt đầu | DATE |  |
| trailer | Trailer phim | VARCHAR(255) |  |

3.7.14 Bảng movie\_category

Bảng 3.7.14 movie\_category

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| movie\_category\_id | Mã thể loại phim | INT | Khóa chính tự tăng |
| category\_id | Mã thể loại | INT | Khóa ngoại |
| movie\_id | Mã phim | INT | Khóa ngoại |

3.7.15 Bảng show\_time

Bảng 3.7.15 show\_time

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| show\_time\_id | Mã suất chiếu | INT | Khóa chính tự tăng |
| show\_tme | Suất chiếu | VARCHAR(50) |  |

3.7.16 Bảng room

Bảng 3.7.16 room

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| room\_id | Mã phòng | INT | Khóa chính tự tăng |
| room\_name | Tên phòng | VARCHAR(50) |  |
| room\_status | Trạng thái phòng | BIT(1) |  |

3.7.17 Bảng movie\_room

Bảng 3.7.17 movie\_room

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| movie\_room\_id | Mã phòng chiếu phim | INT | Khóa chính tự tăng |
| movie\_id | Mã phim | INT | Khóa ngoại |
| room\_id | Mã phòng | INT | Khóa ngoại |

3.7.18 Bảng room\_seat

Bảng 3.7.18 room\_seat

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| room\_seat\_id | Mã ghế của phòng | INT | Khóa chính tự tăng |
| seat\_status | Trạng thái ghế | BIT(1) |  |
| room\_id | Mã phòng | INT | Khóa ngoại |
| seat\_id | Mã ghế | INT | Khóa ngoại |

3.7.19 Bảng movie\_ticket

Bảng 3.7.19 movie\_ticket

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| movie\_ticket\_id | Mã vé phim | INT | Khóa chính tự tăng |
| projection\_type | Kiểu chiếu | VARCHAR(50) |  |
| show\_date | Ngày chiếu | DATE |  |
| ticket\_price | Giá vé | INT |  |
| movie\_id | Mã phim | INT | Khóa ngoại |
| room\_id | Mã phòng | INT | Khóa ngoại |
| show\_time\_id | Mã suất chiếu | INT | Khóa ngoại |

3.7.20 Bảng ticket

Bảng 3.7.20 ticket

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Ghi chú** |
| ticket\_id | Mã vé | INT | Khóa chính tự tăng |
| time\_create | Ngày tạo | DATE |  |
| ticket\_status | Trạng thái vé | BIT(1) |  |
| movie\_ticket\_id | Mã vé phim | INT | Khóa ngoại |
| seat\_id | Mã ghế | INT | Khóa ngoại |
| user\_id | Mã thành viên | INT | Khóa ngoại |

CHƯƠNG IV. KẾT QUẢ TRIỂN KHAI HỆ THỐNG WEBSITE

* 1. Công cụ xây dựng
* Hệ quản trị cơ sở dữ liệu (MySQL Workbench): thiết kế cơ sở dữ liệu.
* HTML, CSS, JS, Bootstrap: xây dựng giao diện chương trình.
* Spring Framework: xây dựng Back-end(sever) chương trình.
* Angular Framework: xây dựng Front-end chương trình
  1. Thiết kế giao diện

4.2.1 Giao diện

Hình 4.2.1 Giao diện

4.2.2 Giao diện

Hình 4.2.2 Giao diện

4.2.3 Giao diện

Hình 4.2.3 Giao diện

4.2.4 Giao diện

Hình 4.2.4 Giao diện

4.2.5 Giao diện

Hình 4.2.5 Giao diện

4.2.6 Giao diện

Hình 4.2.6 Giao diện

4.2.7 Giao diện

Hình 4.2.7 Giao diện

4.2.8 Giao diện

Hình 4.2.8 Giao diện

4.2.9 Giao diện

Hình 4.2.9 Giao diện

KẾT LUẬN

1. **Kết luận**
   1. **Về công nghệ**
      * Tìm hiểu và nắm bắt được các công cụ thiết kế web.
      * Biết được các thiết kế web động, trang web một trang cũng như cách tổ chức cơ sở dữ liệu.
      * Nắm vững lý thuyết và hiểu sâu hơn về Spring Framework, Angular.
      * Nắm vững lý thuyết và vận dụng thành thạo mô hình MVC, các kiến thức hỗ trợ trong quá trình xây dụng hệ thống: HTML, CSS, MySQL, Javascript, jQuery, …
      * Tích lũy được kinh nghiệm về xây dụng kế hoạch cho việc phát triển một phần mềm hoàn chỉnh, cách bố trì và quản lý thời gian sao cho hợp lý.
   2. **Về ứng dụng**
      * Xây dựng thành công “Website quản lý rạp chiếu phim” với các chức năng đáp ứng được nhu cầu đặt vé, xem thông tin phim, quản lý tài khoản cho thành viên, quản lý bán vé, quản lý phim, quản lý phòng chiếu, quản lý nhân viên,… cho chủ rap chiếu. Đảm bảo các nhu cầu đặt vé online cho người dùng và quản lý rạp chiếu phim cho người quản trị hệ thống.
2. **Hướng phát triển**
   * + Xây dựng thêm một số chức năng mới
     + Tối ưu hóa hệ thống để website hoạt động nhanh hơn.
     + Nâng cấp và hoàn thiện giao diện người dùng, các chức năng và tính bảo mật của hệ thống.

TÀI LIỆU THAM KHẢO

[1]. Java, https://vi.wikipedia.org/wiki/Java/

[2]. Spring Framework, https://levunguyen.com/hoc-lap-trinh-spring-web/

[3]. MySQL, https://levunguyen.com/co-so-du-lieu-mysql/

[4]. Restful API, https://topdev.vn/blog/restful-api-la-gi/

[5]. Boostrap, https://getbootstrap.com/docs/4.6/getting-started/introduction/

[6]. https://levunguyen.com/hoc-lap-trinh-angular/

PHỤ LỤC

1. **Cài đặt IntelliJ IDEA**

https://youtu.be/qvsYyTcSulA

1. **Cài đặt WebStorm**

https://youtu.be/b\_dkO0jM1Ks

1. **Thiết lập đường dẫn tới database**

Thiết lập trong file application.properties