

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO CUỐI KỲ
MÔN CÔNG NGHỆ JAVA

XÂY DỰNG HỆ THỐNG QUẢN LÝ ĐẶT VÉ XEM PHIM ONLINE

Người hướng dẫn: **Ths NGUYỄN THANH PHONG**

Người thực hiện: **HOÀNG THỊ MAI HOA – 52000445**

TRẦN THỊ NGỌC ÁNH – 52000008

ĐOÀN THỊ KIM YẾN - 52000172

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO CUỐI KỲ
MÔN CÔNG NGHỆ JAVA

XÂY DỰNG HỆ THỐNG QUẢN LÝ

ĐẶT VÉ XEM PHIM ONLINE

Người hướng dẫn: **Ths NGUYỄN THANH PHONG**

Người thực hiện: **HOÀNG THỊ MAI HOA – 52000445**

TRẦN THỊ NGỌC ÁNH – 52000008

ĐOÀN THỊ KIM YẾN – 52000172

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022

LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành nhất đến Thầy Nguyễn Thanh Phong đã tận tình chỉ dạy chúng em trong môn học “Công nghệ Java”. Những kiến thức tiếp thu được từ Thầy không chỉ là nền tảng cho quá trình thực hiện báo cáo mà đó còn là hành trang quý báu cho chúng em trong tương lai. Tuy nhiên do kiến thức là vô hạn, báo cáo có thể không tránh khỏi những thiếu sót nhất định. Chúng em rất mong được tiếp thu những ý kiến đóng góp của Thầy để có thể hoàn thiện hơn. Cuối cùng, chúc Thầy có thật nhiều sức khỏe, ngày càng gặt hái được nhiều thành công trong sự nghiệp.

Chúng em xin chân thành cảm ơn!

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày 11 tháng 12 năm 2022

(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày 11 tháng 12 năm 2022

(kí và ghi họ tên)

TÓM TẮT

Ngày nay, chúng ta đang sống trong một môi trường phát triển vượt bậc về các công nghệ hiện đại, sẽ không quá ngạc nhiên khi hiện tại người ta có thể ra ngoài chỉ cần mang theo một chiếc smartphone là có thể giải quyết mọi vấn đề. Với một nền văn minh tiên tiến như vậy thì nhu cầu giải trí, giảm bớt áp lực cho bản thân ngày càng cao, trong đó, xem phim là một hình thức giải trí vô cùng được săn đón. Dựa theo nhu cầu của thị trường, nhóm chúng tôi đã lựa chọn đề tài “Xây dựng hệ thống quản lý đặt vé xem phim online” để giúp việc quản lý thông tin khách hàng trực tuyến được thuận lợi hơn cũng như quản lý doanh thu của rạp được dễ dàng hơn. Nhóm chúng tôi lựa chọn phát triển phần mềm theo mô hình MVC và sử dụng ngôn ngữ lập trình Java.

MỤC LỤC

HỆ THỐNG QUẢN LÝ ĐẶT VÉ XEM PHIM ONLINE

LỜI CẢM ƠN	i
TÓM TẮT	iii
DANH MỤC CÁC HÌNH VẼ.....	3
CHƯƠNG 1 – KẾ HOẠCH XÂY DỰNG.....	4
1. Các bước xây dựng	4
2. Thời gian thực hiện	4
CHƯƠNG 2 – PHÂN TÍCH HỆ THỐNG	5
1. Phân tích nhu cầu	5
2. Đặt tả hệ thống	5
3. Xác định thực thể	6
4. Đặc tả Use Case	6
5. Mô hình Use Case	16
CHƯƠNG 3 – THIẾT KẾ HỆ THỐNG	20
1. Class diagram.....	20
2. Sequence diagrams.....	21
3. Database design	24
CHƯƠNG 4 – KIẾN TRÚC HỆ THỐNG	26
1. Giới thiệu về kiến trúc hệ thống được sử dụng trong dự án	26
1.1. Mô hình MVC.....	26
1.2. Mô hình 3 lớp.....	27
2. Công nghệ sử dụng	29
CHƯƠNG 5 - CODING CONVENTION	32
1. Quy tắc đặt tên.....	32
2. Quy tắt số lượng.....	33

3.	Quy tắc xuống hàng	33
4.	Quy tắc comment	33
5.	Quy tắc WhiteSpace.....	34
5.1.	White Space – thụt đầu dòng	34
5.2.	White Space – Dòng trống.....	34
6.	Quy tắc ngoặc nhọn	35
CHƯƠNG 6 – TESTING.....		36
1.1.	Kế hoạch đánh giá.....	36
1.1.1.	Tiêu chí của đánh giá.....	36
2.1.	Unit Test.....	37
2.1.1.	Kiểm thử giao diện đăng nhập cho khách hàng.....	37
2.1.2.	Kiểm thử chức năng đăng kí	38
BẢNG KPI		43
TÀI LIỆU THAM KHẢO		45

DANH MỤC CÁC HÌNH VẼ

Hình 5.1. Use Case tổng quát.....	16
Hình 5.2. Use Case Đăng ký	16
Hình 5.3. Use Case Đăng nhập	17
Hình 5.4. Use Case Đặt vé	18
Hình 5.5. Use Case Thêm phim	18
Hình 1.1. Class diagram	20
Hình 2.1. Sequence Đăng nhập	21
Hình 2.2. Sequence Đặt vé	22
Hình 2.3. Sequence Đăng ký	23
Hình 3.1.1. Mô hình ERD	24
Hình 3.2.1. Mô hình quan hệ.....	25
Hình 2.1. Mô hình kết hợp giữa mô hình MVC và mô hình 3 lớp	29
Hình 2.2. Sơ đồ luồng đi của Spring Boot	31

CHƯƠNG 1 – KẾ HOẠCH XÂY DỰNG

1. Các bước xây dựng

Bước 1: Xác định mục tiêu dự án.

- Mục tiêu xây dựng hệ thống quản lý đặt vé xem phim online với các chức năng hỗ trợ người dùng như đăng ký tài khoản, đăng nhập, xem danh sách phim, đặt vé. Ngoài ra, nếu đăng nhập bằng tài khoản của admin, người dùng có thể quản lý các danh sách phim, quản lý phòng chiếu, suất chiếu.

Bước 2: Thiết lập lộ trình triển khai, phân bổ tài nguyên.

- Dự án được triển khai theo 5 giai đoạn:
 - Giai đoạn 1: Tìm hiểu đề tài, khảo sát thực tế.
 - Giai đoạn 2: Đặc tả hệ thống, vẽ các sơ đồ liên quan, phân chia công việc cụ thể cho từng thành viên.
 - Giai đoạn 3: Phát triển phần mềm.
 - Giai đoạn 4: Tổng hợp nội dung.
 - Giai đoạn 5: Kiểm tra và hoàn thành dự án.

Bước 3: Quản lý rủi ro phát sinh.

- Trong quá trình làm việc không tránh khỏi những rủi ro phát sinh, ví dụ như thời gian còn hạn chế, kiến thức không đủ,... Dự đoán được điều đó, nhóm chúng tôi đã phân chia công việc cụ thể cho từng thành viên và sắp xếp thời gian hợp lý.

Bước 4: Trình bày kế hoạch và tiếp nhận phản hồi.

2. Thời gian thực hiện

- Giai đoạn 1 (10/11 – 13/11): Tìm hiểu đề tài, khảo sát thực tế.
- Giai đoạn 2 (14/11 – 20/11): Đặc tả hệ thống, vẽ các sơ đồ liên quan, phân chia công việc cụ thể cho từng thành viên.
- Giai đoạn 3 (21/11 – 11/12): Phát triển phần mềm.
- Giai đoạn 4 (12/12 – 15/12): Tổng hợp nội dung.
- Giai đoạn 5 (15/12 – 16/12): Kiểm tra và hoàn thành dự án.

CHƯƠNG 2 – PHÂN TÍCH HỆ THỐNG

1. Phân tích nhu cầu

Ngày nay khi xã hội ngày càng phát triển thì nhu cầu giải trí, giảm bớt áp lực cho bản thân ngày càng cao. Trong đó, xem phim là một hoạt động vô cùng được săn đón và chưa bao giờ lỗi thời. Thế nhưng, bạn đã bao giờ đến quầy mua vé nhưng phải chờ đợi rất lâu, thậm chí phải ra về vì số lượng người mua quá đông? Giải pháp cho vấn đề nan giải trên chính là đặt vé trực tuyến ngay tại nhà bằng cách truy cập vào trang web hoặc tải ứng dụng của hệ thống rạp chiếu phim uy tín. Mô hình đặt vé xem phim trực tuyến không những giúp chúng ta tiết kiệm thời gian xếp hàng chờ đợi để có thể xem những siêu phẩm bom tấn mà còn giúp chúng ta có thể chọn lựa chỗ ngồi theo ý thích, hơn nữa bạn còn có thể dễ dàng tra cứu lịch chiếu phim tại các cụm rạp gần khu vực bạn sống để tiện việc di chuyển. Vì những lý do trên nên nhóm chúng tôi quyết định chọn đề tài xây dựng một hệ thống quản lý đặt vé xem phim online.

2. Đặt tả hệ thống

Hệ thống hướng tới các đối tượng là khách hàng và những nhân viên quản lý trong rạp. Hệ thống có các chức năng chính bao gồm: Đăng ký, đăng nhập, đăng xuất, đặt vé, quản lý thông tin khách hàng. Các chức năng này giúp khách hàng có thể dễ dàng mua vé ở bất cứ đâu hoặc bất cứ khi nào, giải quyết được những hạn chế của việc đặt vé trực tiếp.

Mỗi người dùng khi sử dụng hệ thống để đặt vé đều phải đăng kí tài khoản, các thông tin của người dùng được lưu trên hệ thống bao gồm: Mã người dùng, họ tên, địa chỉ, số điện thoại, ngày sinh, email, mật khẩu. Sau khi đăng kí thành công tài khoản, mỗi người dùng đều có tài khoản để có thể đăng nhập, thông tin mỗi tài khoản bao gồm: mã tài khoản, số tiền đã giao dịch. Người dùng có thể cập nhật thông tin cá nhân, lấy lại mật khẩu hoặc đổi mật khẩu khi cần thiết.

Hệ thống bao gồm nhiều rạp, thuận lợi cho người dùng có thể tùy ý chọn được rạp chiếu thuận lợi nhất cho mình, các thông tin rạp bao gồm: Mã rạp, tên rạp, địa chỉ rạp. Mỗi rạp phim bao gồm nhiều phòng chiếu, thông tin mỗi phòng bao gồm: Mã phòng, số phòng, số lượng ghế tối đa, phân loại phòng. Mỗi phòng chiếu có nhiều ghế để người dùng có thể chọn vị trí xem cho mình, thông tin ghế bao gồm: Mã ghế, tên ghế, phân loại.

Khách hàng phải đặt vé trước khi phim được chiếu, nhân viên tiến hành kiểm tra vé trước khi để khách hàng vào rạp, thông tin về mỗi vé bao gồm: Mã vé, số ghế ngồi, giá vé, ngày mua vé, tên phim, tên phòng chiếu, giờ bắt đầu. Ngoài ra, khi đặt vé, khách hàng có thể lựa chọn các dịch vụ ăn uống của rạp. Sau khi đặt vé và thanh toán thành công, hệ thống sẽ gửi đến bạn thông tin về hóa đơn mà bạn đã đặt bao gồm các thông tin về mã hóa đơn, ngày đặt vé, các thông tin về số lượng vé, số lượng các dịch vụ đã sử dụng và đơn giá.

3. Xác định thực thể

Sau khi tìm hiểu và khảo sát thực tế tại các rạp chiếu phim, chúng tôi đã xác định các tác nhân sử dụng phần mềm như sau:

- Admin: Người quản lý hệ thống.
- Thành viên: Khách hàng thực hiện đăng ký thành viên tại website.

Dựa vào các tác nhân trên, website của chúng tôi gồm 2 module chính: một module cho người nhân viên rạp và một module cho khách hàng xem.

4. Đặc tả Use Case

Các tác nhân tham gia hệ thống:

- Khách hàng: Là người mua vé để xem phim tại rạp. Khách hàng có thể mua vé trực tiếp tại rạp hoặc có thể mua gián tiếp thông qua chức năng đặt vé qua mạng mà hệ thống có hỗ trợ.

- Nhân viên:

- Là người có nhiệm vụ quản lý các bộ phim có tại rạp. Có thể thêm, xóa, sửa thông tin phim tại rạp. Luôn cập nhật những bộ phim bom tấn ăn khách nhất với đầy đủ thông tin cần thiết của bộ phim đó.
- Là người có nhiệm vụ quản lý các phòng chiếu có tại rạp. Có thể thêm, xóa, sửa thông tin phòng tại rạp.
- Là người có nhiệm vụ quản lý các thông tin ghế trong phòng chiếu tại rạp. Có thể thêm, xóa, sửa thông tin ghế tại rạp.

Các chức năng chính của hệ thống:

- Xem thông tin phim: Cho phép khách hàng có thể xem được các thông tin của các bộ phim mà rạp đã, đang và sắp chiếu. Các thông tin gồm: Tên phim, thể loại, thời lượng, nội dung, ngày khởi chiếu...
- Đặt vé: Hệ thống trợ giúp cho khách hàng có thể trực tiếp đặt mua vé online.
- Quản lý phòng chiếu: Hệ thống giúp cho nhân viên quản lý phòng chiếu có thể quản lý tất cả các phòng chiếu của rạp dựa trên các thuộc tính của các phòng chiếu.
- Quản lý phim: Hỗ trợ nhân viên quản lý phim có thể quản lý kho tư liệu phim của rạp dựa trên các thuộc tính của từng bộ phim.
- Quản lý thông tin khách hàng: Chức năng hỗ trợ quản trị viên biết được các thông tin cũng như số lượng khách hàng của rạp.
- Đăng nhập hệ thống: Nhân viên, quản trị viên cần đăng nhập vào hệ thống để thực hiện chức năng của họ.
- Quản lý bán vé: Chức năng này hỗ trợ cho nhân viên quản lý vé có thể kiểm soát số lượng vé đã bán ra và thông tin của việc bán vé một cách hiệu quả và nhanh chóng nhất.

Các thực thể chính:

- Thực thể phim: Mỗi phim đều có các thuộc tính: Mã phim, tên phim, poster, thời lượng, thể loại, nội dung, trailer, ngày khởi chiếu.
- Thực thể phòng chiếu: Rạp chiếu phim có nhiều phòng chiếu, mỗi phòng chiếu được quản lý theo các thuộc tính sau: Mã phòng, số phòng, phân loại phòng.
- Thực thể vé: Là vé xem phim được tạo ra để hiển thị thông tin phục vụ cho nhu cầu xem phim của khách hàng trong từng buổi chiếu phim tại rạp.
- Thực thể lịch chiếu: Là lịch mà trong đó thể hiện đầy đủ thông tin của một buổi chiếu phim tại rạp. Trong lịch chiếu bao gồm các thông tin thời gian chiếu phim, phòng chiếu và phim được chiếu trong thời gian đó.
- Thực thể hóa đơn: Hóa đơn được lập khi khách hàng thực hiện các thao tác đặt, mua vé xem phim qua online. Sau khi hoàn tất thủ tục thanh toán hệ thống sẽ in hóa đơn cho khách hàng.

Mô tả chi tiết Use Case.

Tên usecase	ĐĂNG NHẬP
Mô tả	Diễn tả cách người dùng đăng nhập vào hệ thống.
Tác nhân	Nhân viên, khách hàng.
UC liên quan	Đăng ký.
Tiền điều kiện	Có tài khoản người dùng.
Hậu điều kiện	<ul style="list-style-type: none"> - Thành công: Người dùng đăng nhập được vào hệ thống. - Lỗi: Người dùng không đăng nhập được vào hệ thống và hệ thống không bị thay đổi.
Luồng sự kiện chính	<p>Chức năng này bắt đầu khi người dùng muốn đăng nhập vào hệ thống:</p> <ol style="list-style-type: none"> 1. Hệ thống yêu cầu người dùng nhập tên và mật khẩu. 2. Người dùng nhập tên và mật khẩu.

	3. Hệ thống xác thực thông tin hợp lệ sẽ cho phép người dùng vào hệ thống.
Luồng sự kiện phát sinh	Tên đăng nhập/mật khẩu không hợp lệ: Nếu trong luồng chính, người dùng nhập sai tên hoặc mật khẩu, hệ thống hiển thị một thông báo lỗi. Người dùng có thể lựa chọn quay lại thời điểm bắt đầu luồng chính hoặc hủy đăng nhập, lúc này chức năng kết thúc.
Yêu cầu đặc biệt khác	Hệ thống có kết nối Internet.

Tên usecase	QUẢN LÝ PHÒNG CHIẾU
Mô tả	Usecase cho phép nhân viên quản lý có thể quản lý thông tin phòng chiếu.
Tác nhân	Nhân viên.
Tiền điều kiện	Nhân viên phải đăng nhập vào hệ thống.
Hậu điều kiện	<ul style="list-style-type: none"> - Thành công: Phòng chiếu có thông tin chính xác. - Lỗi: Hệ thống báo lỗi, thông tin, dữ liệu về các phòng chiếu không được cập nhật.
Luồng sự kiện chính	<p>Chức năng này bắt đầu khi nhân viên quản lý phòng chiếu muốn thực hiện các thao tác quản lý thông tin, dữ liệu về phòng chiếu:</p> <ol style="list-style-type: none"> 1. Hệ thống hiển thị menu lựa chọn bao gồm: <ul style="list-style-type: none"> - Thêm phòng chiếu. - Cập nhật thông tin phòng chiếu. - Xóa phòng chiếu. 2. Nhân viên sẽ chọn chức năng mà mình muốn thực hiện: <ul style="list-style-type: none"> - Nếu nhân viên quản lý phòng chiếu chọn “Tạo phòng chiếu”, luồng con tạo phòng chiếu sẽ được thực thi.

	<ul style="list-style-type: none"> - Nếu nhân viên quản lý phòng chiếu chọn “Cập nhật thông tin phòng chiếu”, luồng con cập nhật phòng chiếu sẽ được thực thi. - Nếu nhân viên quản lý phòng chiếu chọn “Xóa phòng chiếu”, luồng con xóa phòng chiếu sẽ được thực thi. <p>3. Nhân viên thao tác sử dụng menu:</p> <ul style="list-style-type: none"> - Luồng tạo phòng chiếu: <ol style="list-style-type: none"> 1. Hệ thống yêu cầu nhân viên nhập thông tin phòng chiếu: Mã phòng, số phòng, số lượng ghế, phân loại. 2. Hệ thống yêu cầu nhân viên xác nhận. 3. Nhân viên quản lý phòng chiếu xác nhận. 4. Hệ thống sinh một mã phòng chiếu mới cho phòng chiếu vừa tạo. 5. Hệ thống lưu lại thông tin dữ liệu của phòng chiếu mới. - Luồng xóa phòng chiếu: <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách các phòng chiếu đang sử dụng. 2. Nhân viên quản lý phòng chiếu chọn phòng chiếu muốn xóa. 3. Hệ thống yêu cầu nhân viên quản lý phòng chiếu xác nhận việc xóa phòng chiếu. 4. Nhân viên quản lý phòng chiếu xác nhận xóa. 5. Hệ thống xóa bỏ thông tin dữ liệu của phòng chiếu đó. - Luồng cập nhật thông tin phòng chiếu: <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách các phòng chiếu đang sử dụng.
--	---

	<ol style="list-style-type: none"> 2. Nhân viên quản lý phòng chiếu chọn phòng chiếu cần cập nhật. 3. Hệ thống tìm và hiển thị thông tin phòng chiếu đó. 4. Nhân viên quản lý phòng chiếu cập nhật các thông tin cho phòng chiếu đó. 5. Hệ thống yêu cầu nhân viên quản lý phòng chiếu xác nhận việc cập nhật thông tin cho phòng chiếu đó. 6. Nhân viên quản lý phòng chiếu xác nhận cập nhật. 7. Hệ thống lưu lại các thông tin dữ liệu vừa được cập nhật của phòng chiếu.
Luồng sự kiện phát sinh	<ul style="list-style-type: none"> - Không tìm thấy phòng chiếu: Trong “Sửa thông tin phòng chiếu”, “Xóa phòng chiếu” một phòng chiếu với mã nhập vào không chính xác nên hệ thống sẽ yêu cầu nhập lại hoặc hủy bỏ thao tác. - Xóa không thành công: Nếu trong luồng “Xóa phòng chiếu” nhân viên quản lý không muốn xóa phòng chiếu nên hủy bỏ thao tác xóa. Thông tin dữ liệu của phòng chiếu trong hệ thống sẽ không bị xóa. - Tạo phòng chiếu không thành công: Nếu trong luồng “Tạo phòng chiếu” nhân viên quản lý không muốn tạo phòng chiếu nên hủy bỏ thao tác tạo phòng chiếu mới. - Cập nhật không thành công: Nếu trong luồng “Cập nhật phòng chiếu” nhân viên quản lý không muốn cập nhật phòng chiếu nên hủy bỏ thao tác cập nhật. Thông tin dữ liệu của phòng chiếu trong hệ thống sẽ không thay đổi.
Yêu cầu đặc biệt khác	Hệ thống có kết nối Internet.

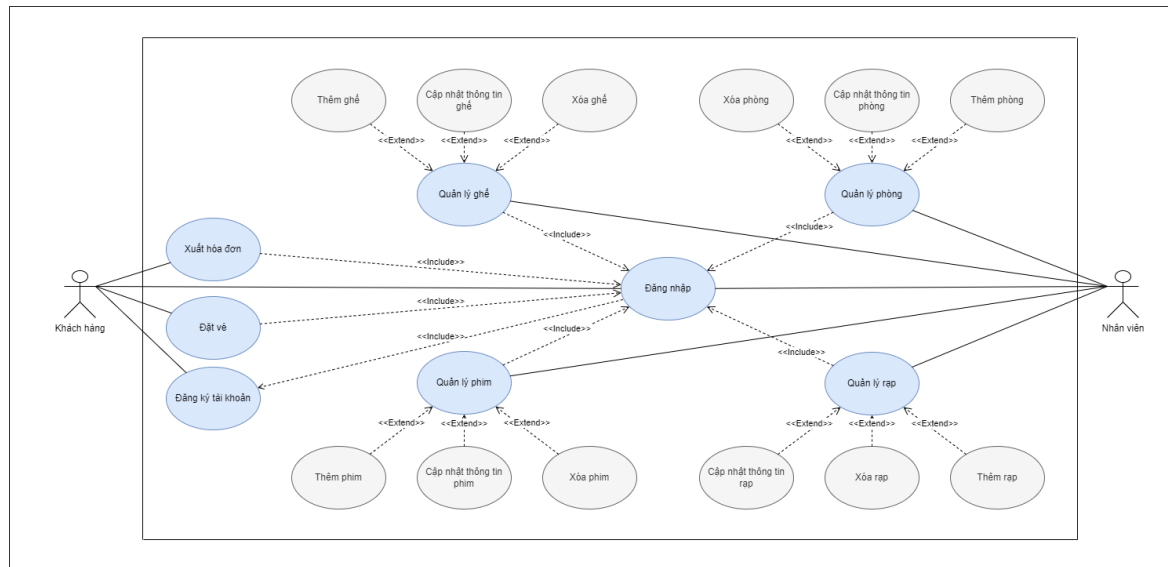
Tên usecase	ĐẶT VÉ
Mô tả	Chức năng này cho phép khách hàng đặt mua vé khi muốn xem phim hiện có tại rạp. Khách hàng có thể chọn phim, lịch chiếu, loại vé, số lượng. Nhờ chức năng này mà khách hàng có thể dễ dàng mua được vé mà không cần phải đến tận rạp xếp hàng chờ mua vé.
Tác nhân	Khách hàng.
Tiền điều kiện	Khách hàng có nhu cầu mua vé để xem phim.
Hậu điều kiện	<ul style="list-style-type: none"> - Thành công: Phiếu đặt vé sẽ được in ra và khách hàng đã có được vé của bộ phim mình muốn xem. - Lỗi: Khách hàng không đặt được vé, không lập phiếu đặt vé.
Luồng sự kiện chính	<p>Chức năng này bắt đầu khi khách hàng đã xem thông tin, lịch chiếu của phim và chọn “Đặt vé”.</p> <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách các bộ phim sắp chiếu của rạp. 2. Khách hàng chọn một bộ phim muốn xem. 3. Hệ thống hiển thị danh sách các lịch chiếu của bộ phim mà khách hàng đã chọn. 4. Khách hàng lựa chọn một buổi chiếu phù hợp và chọn “Mua vé” để tiếp tục. 5. Hệ thống hiển thị danh sách các loại vé tương ứng với bộ phim đã chọn. Ứng với mỗi loại vé sẽ có số lượng để khách hàng chọn. 6. Khách hàng chọn số lượng vé tương ứng với loại vé đã đặt. 7. Hệ thống cập nhật các thông tin đã chọn vào phiếu đặt vé.

	<p>8. Khách hàng chọn các chức năng: “Đặt thêm” để đặt thêm vé cho bộ phim khác. Quá trình được lặp lại từ đầu; “Tiếp tục” để chuyển sang bước tiếp theo của quy trình đặt vé.</p> <p>9. Hệ thống hiển thị form để khách hàng điền thông tin cá nhân (nếu không là thành viên hoặc chưa đăng nhập vào hệ thống), nơi giao hàng và chọn hình thức thanh toán phù hợp.</p> <p>10. Khách hàng điền đầy đủ thông tin.</p> <p>11. Hệ thống hiển thị phiếu đặt vé.</p> <p>12. Khách hàng chọn “Thanh toán” để hoàn tất quy trình đặt vé.</p> <p>13. Hệ thống in ra phiếu đặt vé.</p>
Luồng sự kiện phát sinh	Nếu khi hiện phiếu đặt vé mà khách hàng chọn: “HỦY” thì khách hàng sẽ hủy bỏ quá trình đặt vé; “THANH TOÁN” thì sẽ chuyển sang bước thanh toán tiền theo hình thức thanh toán mà khách hàng đã chọn trong phiếu đặt vé.
Yêu cầu đặc biệt khác	Hệ thống có kết nối Internet.
Tên usecase	QUẢN LÝ PHIM
Mô tả	Chức năng này cho phép nhân viên quản lý phim quản lý danh sách phim trong hệ thống. Bao gồm thêm, cập nhật và xóa phim từ hệ thống.
Tác nhân	Nhân viên
Tiền điều kiện	Nhân viên phải đăng nhập vào hệ thống trước khi chức năng bắt đầu.
Hậu điều kiện	<ul style="list-style-type: none"> - Thành công: Phim sẽ được thêm, xóa hoặc cập nhật. - Lỗi: Trạng thái hệ thống không thay đổi.

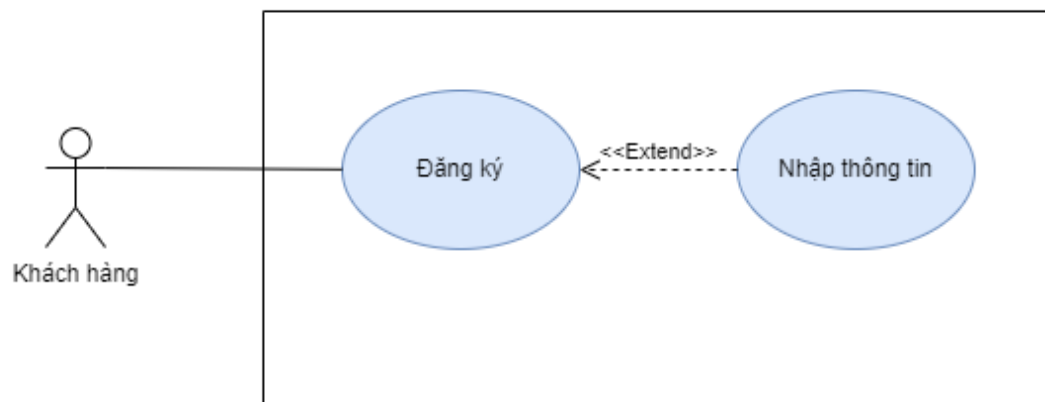
<p>Luồng sự kiện chính</p>	<p>Chức năng này bắt đầu khi nhân viên quản lý phim muốn thực hiện các thao tác để quản lý thông tin, dữ liệu về loại phim của rạp:</p> <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách các loại phim và các chức năng lựa chọn bao gồm: Tạo loại phim; cập nhật loại phim; xóa loại phim. 2. Khi nhân viên quản lý phim chọn chức năng muốn thực hiện: <ul style="list-style-type: none"> - Nếu nhân viên quản lý phim chọn “Thêm phim”, luồng con “Thêm phim” sẽ được thực thi. - Nếu nhân viên quản lý phòng chiếu chọn “Cập nhật thông tin phim”, luồng con “Cập nhật thông tin phim” sẽ được thực thi. - Nếu nhân viên quản lý phòng chiếu chọn “Xóa phim”, luồng con “Xóa phim” sẽ được thực thi. 3. Nhân viên thao tác sử dụng menu: <ul style="list-style-type: none"> - Luồng thêm phim: <ol style="list-style-type: none"> 1. Hệ thống yêu cầu nhân viên quản lý phim nhập thông tin phim. 2. Nhân viên quản lý phim nhập vào các thông tin tương ứng. 3. Hệ thống yêu cầu nhân viên xác nhận. 4. Nhân viên quản lý phim xác nhận việc thêm phim. 5. Hệ thống sinh một mã loại phim mới cho loại phim vừa tạo. 6. Hệ thống lưu lại thông tin dữ liệu của phim mới. - Luồng xóa phim: <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách phim.
----------------------------	---

	<ol style="list-style-type: none"> 2. Nhân viên quản lý phim chọn phim muốn xóa. 3. Hệ thống nhắc nhân viên quản lý phim xác nhận việc xóa phim. 4. Nhân viên quản lý phim xác nhận xóa phim. 5. Hệ thống xóa bỏ thông tin dữ liệu của phim đó. <p>- Luồng cập nhật thông tin phim:</p> <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách phim. 2. Nhân viên quản lý phim chọn phim cần cập nhật. 3. Hệ thống tìm và hiển thị form cập nhật thông tin phim đó. 4. Nhân viên quản lý phim cập nhật các thông tin cho phim đó. 5. Hệ thống yêu cầu nhân viên quản lý phim xác nhận cập nhật thông tin cho phim đó. 6. Nhân viên quản lý phim xác nhận. 7. Hệ thống lưu lại các thông tin dữ liệu vừa cập nhật.
Luồng sự kiện phát sinh	<ul style="list-style-type: none"> - Xóa không thành công: Nếu trong luồng “Xóa phim”, nhân viên quản lý phim quyết định không xóa phim, lệnh xóa bị hủy và luồng chính được khởi động lại tại thời điểm ban đầu. - Cập nhật không thành công: Nếu trong luồng “Cập nhật thông tin phim”, nhân viên quản lý phim không muốn sửa thông tin phim nên hủy bỏ thao tác sửa, thông tin dữ liệu loại phim trong hệ thống không bị thay đổi. - Thêm không thành công: Nếu trong luồng “Thêm phim”, nhân viên quản lý phim không muốn thêm phim nên hủy bỏ thao tác thêm phim mới, trạng thái hệ thống không thay đổi.
Yêu cầu đặc biệt khác	Hệ thống có kết nối Internet.

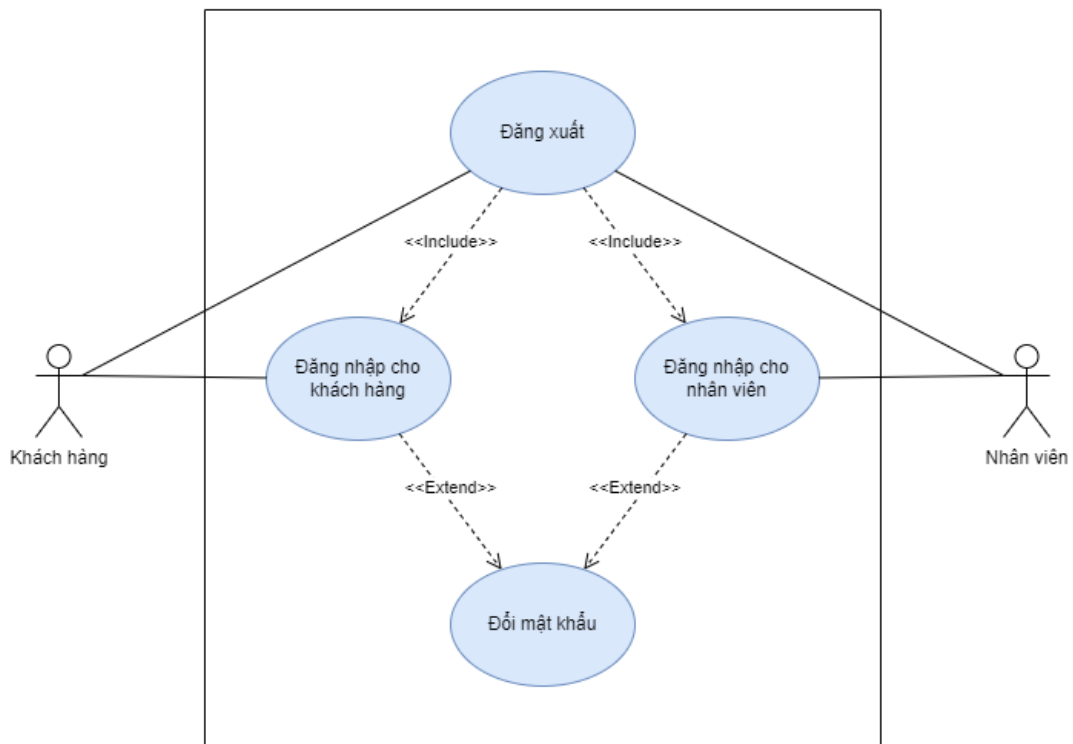
5. Mô hình Use Case



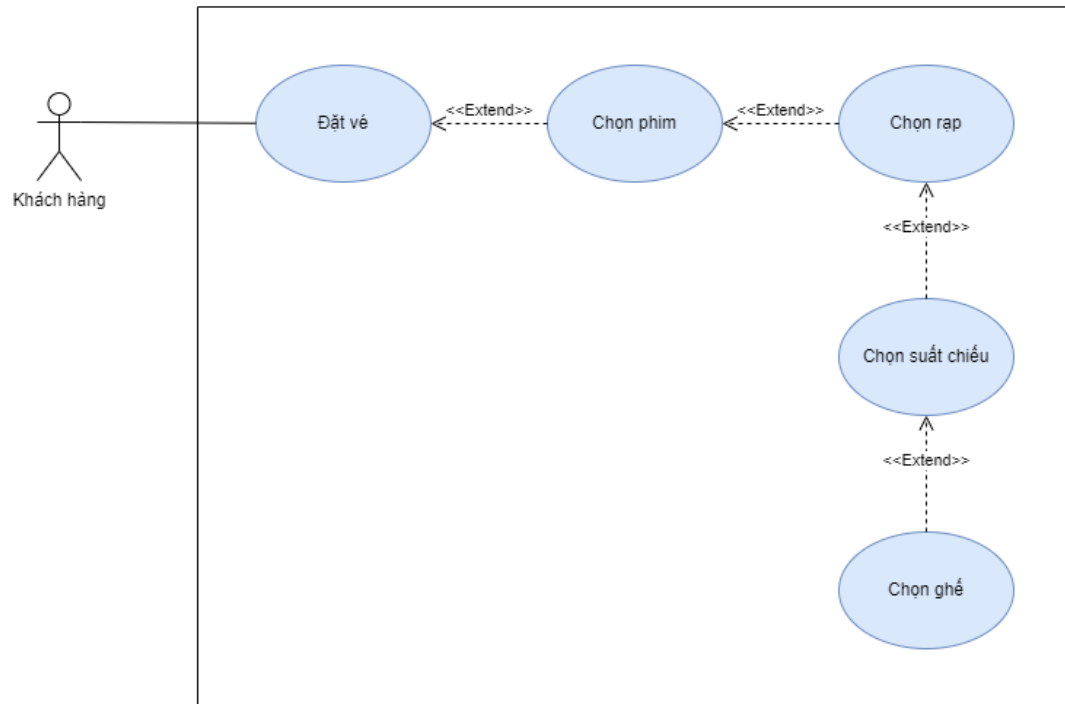
Hình 5.1. Use Case tổng quát



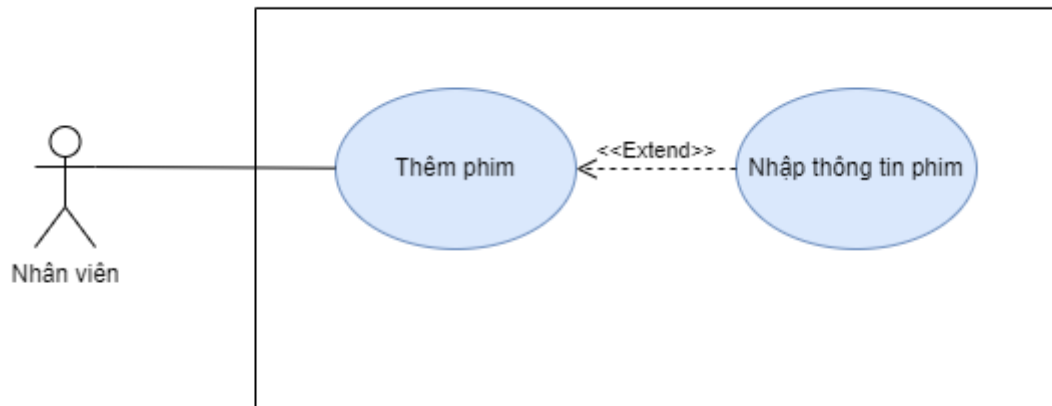
Hình 5.2. Use Case Đăng ký



Hình 5.3. Use Case Đăng nhập



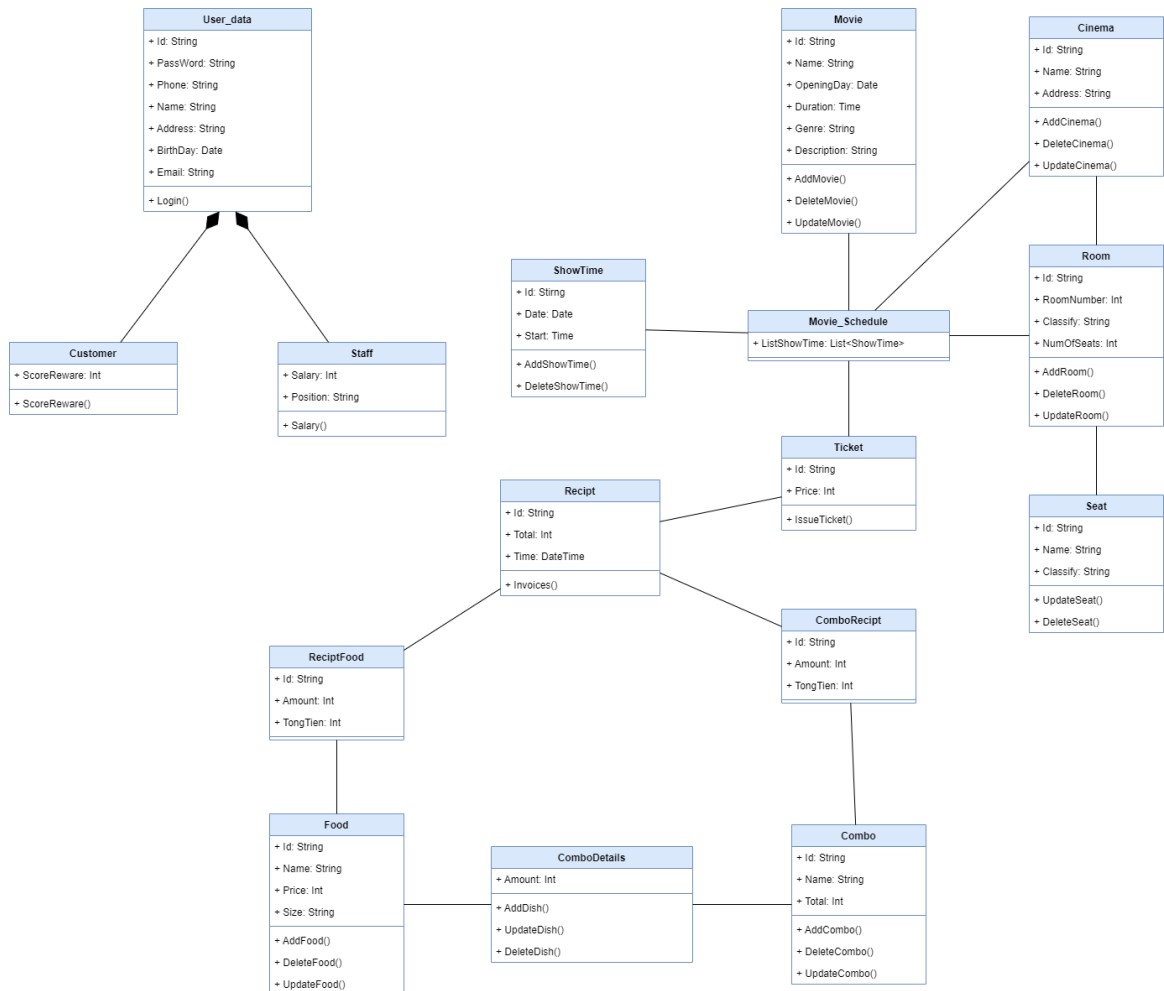
Hình 5.4. Use Case Đặt vé



Hình 5.5. Use Case Thêm phim

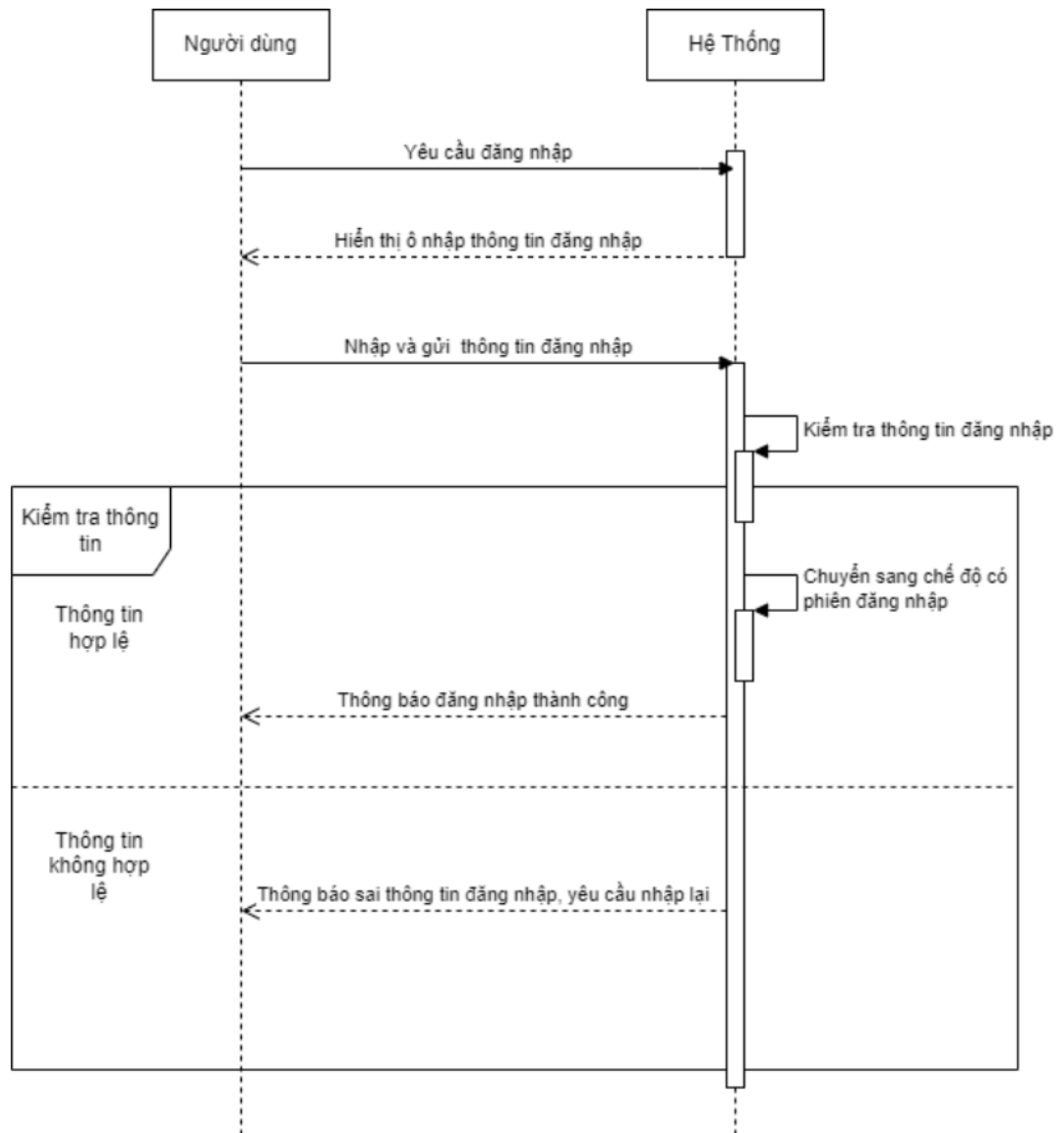
CHƯƠNG 3 – THIẾT KẾ HỆ THỐNG

1. Class diagram

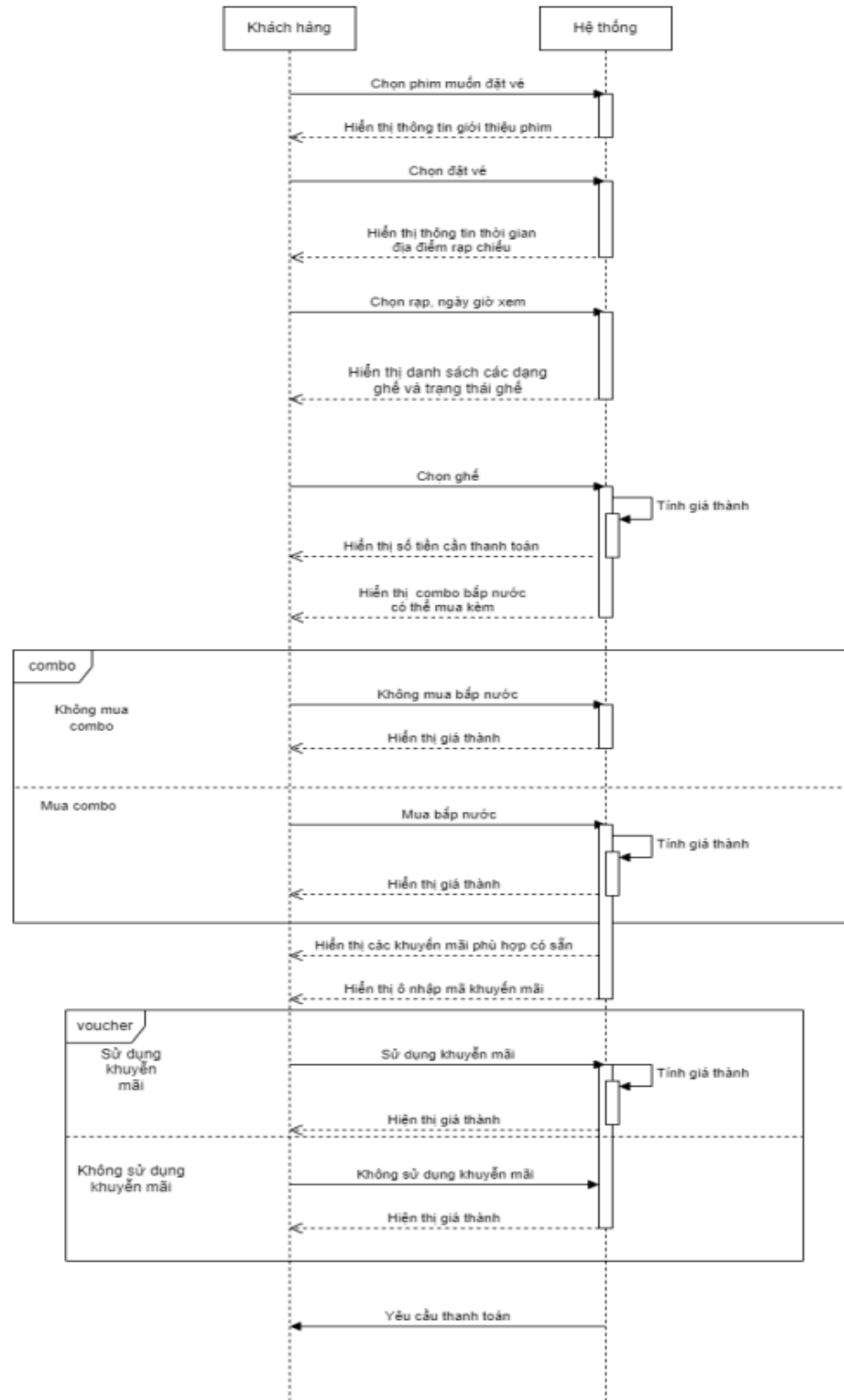


Hình 1.1. Class diagram

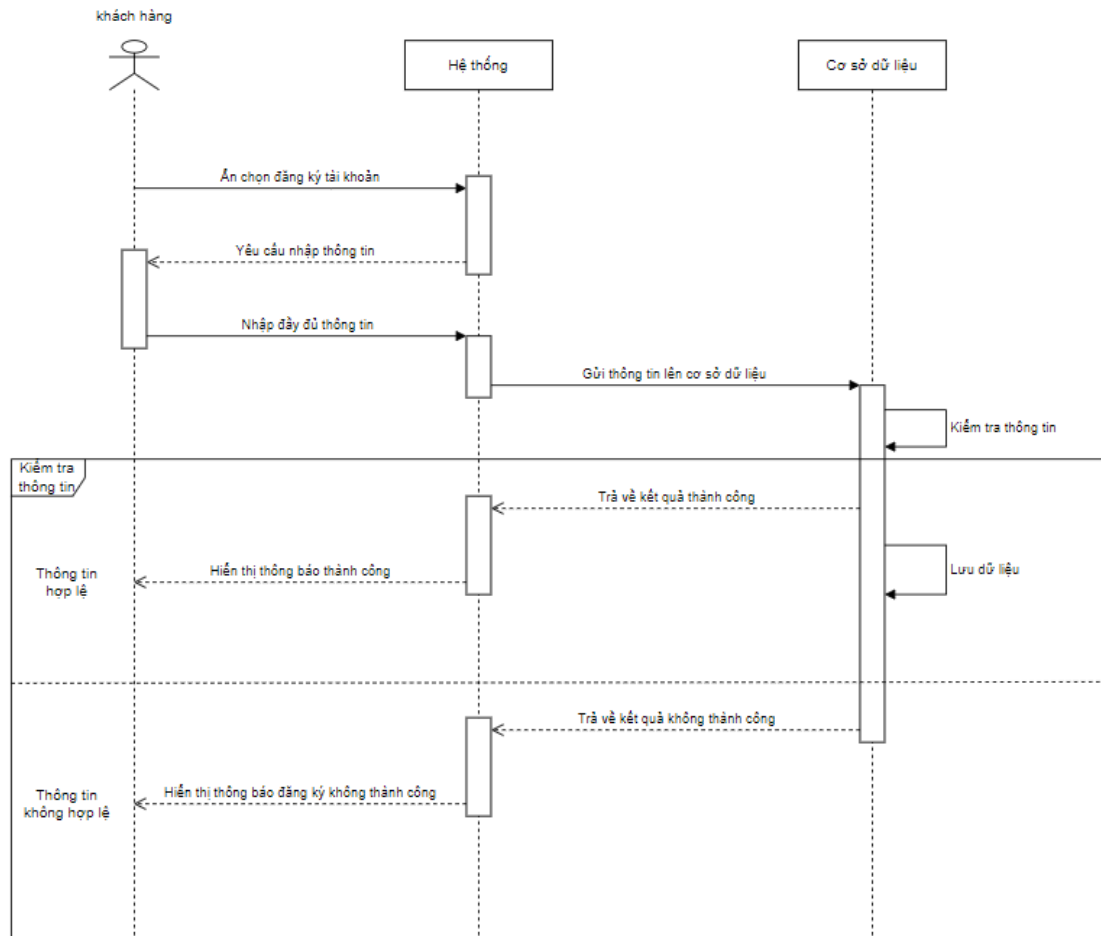
2. Sequence diagrams



Hình 2.1. Sequence Đăng nhập



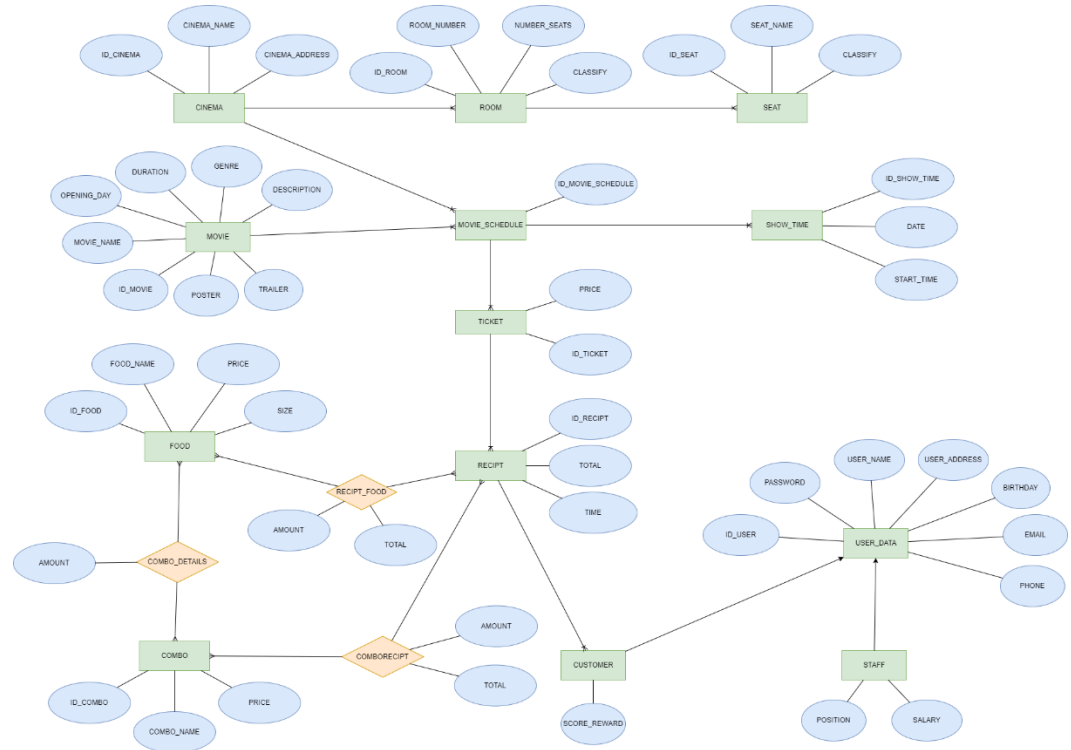
Hình 2.2. Sequence Đặt vé



Hình 2.3. Sequence Đăng ký

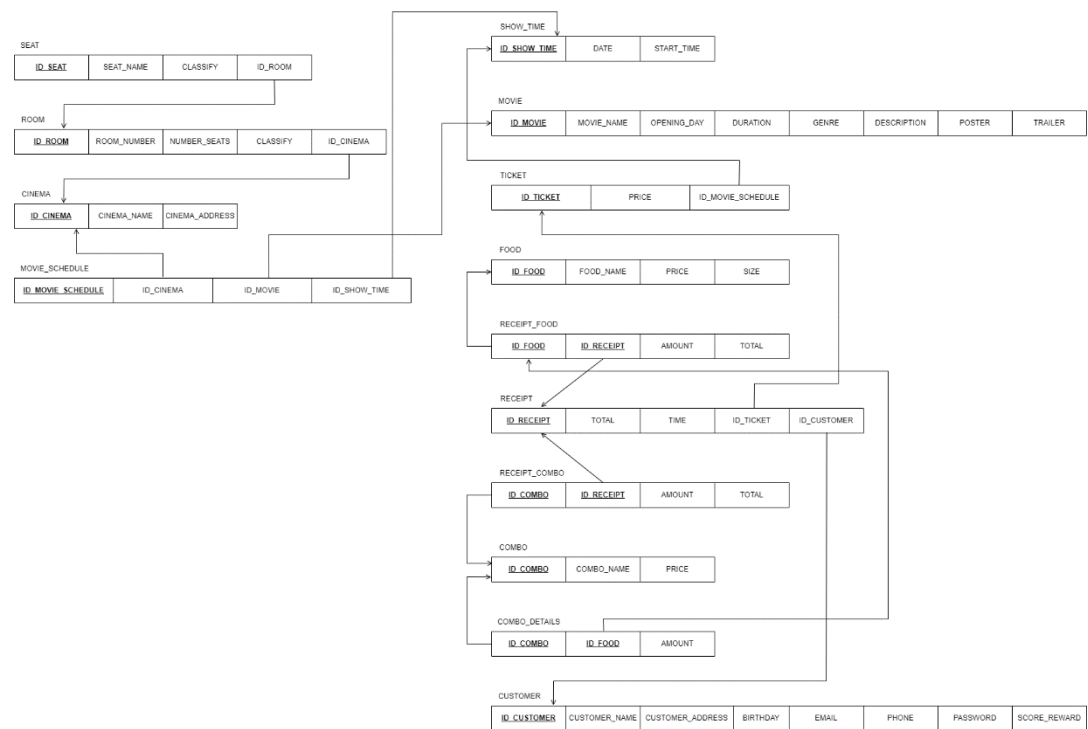
3. Database design

3.1. Mô hình ERD



Hình 3.1.1. Mô hình ERD

3.2. Mô hình quan hệ



Hình 3.2.1. Mô hình quan hệ

Bảng 2.1: So Sánh Spring MVC, Spring Boot và Spring Security

CHƯƠNG 4 – KIẾN TRÚC HỆ THỐNG

1. Giới thiệu về kiến trúc hệ thống được sử dụng trong dự án

Trang web được xây dựng dựa trên mô hình Model-View-Controller (MVC) kết hợp mô hình 3 lớp.

1.1. Mô hình MVC

Mô hình MVC là một mẫu kiến trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View, Controller. Mỗi thành phần kiến trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng. MVC tách lớp logic nghiệp vụ và lớp hiển thị ra riêng biệt.

1.1.1. Model trong MVC

Model chứa một cấu trúc dữ liệu có chức năng lưu trữ toàn bộ các thông tin dữ liệu của một ứng dụng. Trong mô hình MVC, Model đóng vai trò kết nối cho hai thành phần View và Controller. Model có chức năng vận chuyển thông tin từ nội hàm để hiển thị đến người dùng thông qua màn hình và xử lý các thông tin để người dùng dễ dàng tiếp cận nhất.

Model hoàn toàn độc lập với các thành phần còn lại trong MVC và nó chứa các tác vụ cần thiết nhất cho quá trình lập trình.

1.1.2. View trong MVC

View là thành phần liên quan đến giao diện mà người dùng trải nghiệm. Thông qua View, người dùng sẽ thực hiện các thao tác tìm kiếm, sử dụng thông tin trên trang web.

View thực hiện nhiệm vụ tạo tương tác với người dùng và hiển thị các kết quả từ tầng Controller. Đồng thời, View cũng thực hiện việc tiếp nhận các hoạt động, yêu cầu của người dùng để chuyển đến Controller xử lý.

1.1.3. Controller

Controller thực hiện chức năng kết nối tương tác giữa View và Model. Nó định nghĩa các lệnh và thực hiện xử lý các lệnh trong hệ thống. Controller điều khiển

hành động của người dùng từ View và tương tác với Model để chuyển tải thông tin cần thiết đến người dùng.

1.1.4. Luồng đi trong mô hình MVC

- Khi có một yêu cầu gửi từ phía khách hàng tới server, bộ phận Controller sẽ có nhiệm vụ nhận yêu cầu và xử lý yêu cầu đó. Nếu cần, nó sẽ gọi đến Model.
- Sau khi xử lý xong, toàn bộ kết quả sẽ được đẩy về phần View.
- View sẽ hiển thị lại thông tin đã xử lý cho người dùng.

1.2. Mô hình 3 lớp

Mô hình 3 lớp hay còn được gọi là mô hình Three Layer(3-Layer), mô hình này ra đời nhằm phân chia các thành phần trong hệ thống, các thành phần cùng chức năng sẽ được nhóm lại với nhau và phân chia công việc cho từng nhóm để dữ liệu không bị chồng chéo và chạy lộn xộn.

Mô hình này phát huy hiệu quả nhất khi bạn xây dựng một hệ thống lớn, việc quản lý code và xử lý dữ liệu lỗi dễ dàng hơn.

1.2.1. Thành phần

1.2.1.1. Presentation Layer (GUI)

Lớp này có nhiệm vụ chính là giao tiếp với người dùng. Nó gồm các thành phần giao diện (winform, webform, ...) và thực hiện các công việc như nhập liệu, hiển thị dữ liệu, kiểm tra tính đúng đắn dữ liệu trước khi gọi lớp Business Logic Layer (BLL).

1.2.1.2. Business Logic Layer (BLL)

Layer này phân ra 2 thành nhiệm vụ:

Đây là nơi đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, xử lý chính nguồn dữ liệu từ Presentation Layer trước khi truyền xuống Data Access Layer và lưu xuống hệ quản trị CSDL.

Đây còn là nơi kiểm tra các ràng buộc, tính toán vẹn và hợp lệ dữ liệu, thực hiện tính toán và xử lý các yêu cầu nghiệp vụ, trước khi trả kết quả về Presentation Layer.

1.2.1.3. Data Access Layer (DAL)

Lớp này có chức năng giao tiếp với hệ quản trị CSDL như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu (tìm kiếm, thêm, xóa, sửa,...).

Lý do chọn mô hình MVC:

- Đầu tiên, nhắc tới ưu điểm mô hình MVC thì đó là băng thông nhẹ vì không sử dụng viewstate nên khá tiết kiệm băng thông. Việc giảm băng thông giúp website hoạt động ổn định hơn.
- Kiểm tra đơn giản và dễ dàng, kiểm tra lỗi phần mềm trước khi bàn giao lại cho người dùng.
- Một lợi thế chính của MVC là nó tách biệt các phần Model, Controller và View với nhau.
- Sử dụng mô hình MVC chức năng Controller có vai trò quan trọng và tối ưu trên các nền tảng ngôn ngữ khác nhau
- Ta có thể dễ dàng duy trì ứng dụng vì chúng được tách biệt với nhau.
- Có thể chia nhiều developer làm việc cùng một lúc. Công việc của các developer sẽ không ảnh hưởng đến nhau.
- Hỗ trợ TTD (test-driven development). Chúng ta có thể tạo một ứng dụng với unit test và viết các won test case.
- Phiên bản mới nhất của MVC hỗ trợ trợ thiết kế responsive website mặc định và các mẫu cho mobile. Chúng ta có thể tạo công cụ View của riêng mình với cú pháp đơn giản hơn nhiều so với công cụ truyền thống.

Tuy nhiên, MVC chỉ mô tả luồng đi của dữ liệu, nó không nói rõ như code đặt ở đâu (ở Model, View hay Controller), rồi lưu trữ Model vào database kiểu gì,... Do đó, đối với ứng dụng hoàn chỉnh cần kết hợp cả mô hình MVC và 3 lớp lại với nhau.

2. Công nghệ sử dụng

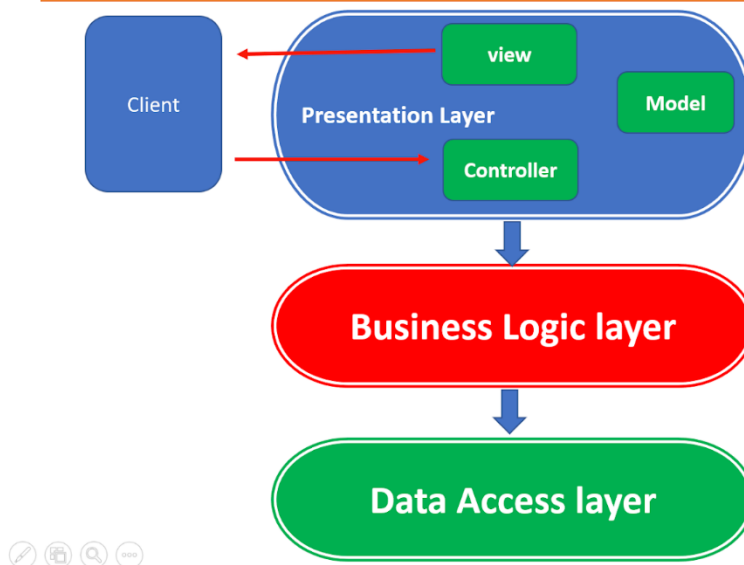
Công nghệ được sử dụng trong project chính là Spring Boot.

Spring Boot là gì?

Spring Boot là một framework vi mô mã nguồn mở được duy trì bởi Pivotal. Framework này cung cấp cho các nhà phát triển Java một nền tảng để bắt đầu với một ứng dụng Spring cấu hình tự động. Với nó, các nhà phát triển có thể bắt đầu nhanh chóng mà không mất thời gian chuẩn bị và cấu hình ứng dụng Spring của họ.

Cấu trúc source code của Spring Boot được dựa trên hai mô hình là mô hình MVC và mô hình 3 lớp.

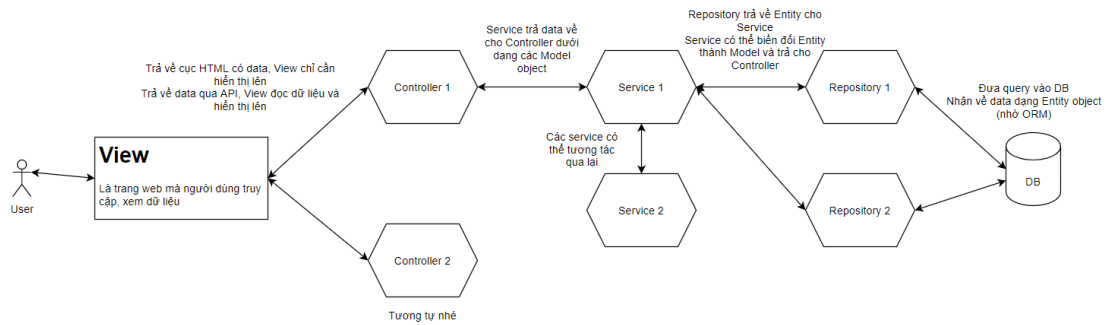
Three-Tier architecture vs MVC pattern



Hình 2.1. Mô hình kết hợp giữa mô hình MVC và mô hình 3 lớp

Kết hợp hai mô hình lại, chúng ta có được ứng dụng Spring Boot hoàn chỉnh, gồm các thành phần sau:

- Controller: trả về View (có chứa data sẵn, dạng trang HTML), hoặc Model thể hiện dưới dạng API cho View (View viết riêng bằng React, Vue, hoặc Angular).
- Service: chứa các code tính toán, xử lý. Khi Controller yêu cầu, thì Service tương ứng sẽ tiếp nhận và cho ra dữ liệu trả cho Controller (trả về Model). Controller sẽ gửi về View như trên.
- Repository: Service còn có thể tương tác với service khác, hoặc dùng Repository để gọi DB. Repository là thẳng trực tiếp tương tác, đọc ghi dữ liệu trong DB và trả cho service.
- Model chỉ đơn giản là các đối tượng được Service tính toán xong trả về cho Controller.
- View thì có 2 loại, một là dạng truyền thống là trả về 1 cục HTML có data rồi. Lúc này Controller sẽ pass dữ liệu vào View và return về (Spring MVC có JSP hoặc template engine như Thymeleaf làm điều đó).
- View dạng 2 là dạng View tách riêng (không thuộc về project Spring boot). Thường có trong các hệ thống dùng API. View sẽ được viết riêng bằng React, Angular,... Controller sẽ đưa dữ liệu Model thông qua API cho View, và cũng nhận lại các yêu cầu qua API luôn.



Hình 2.2. Sơ đồ luồng đi của Spring Boot

Lý do chọn Spring Boot:

- Spring Boot sẽ tự động cấu hình, gần như mọi thứ đã trở nên dễ dàng và giúp người lập trình tập trung vào việc code.
- Spring Boot giúp quản lý từng Microservice một cách dễ dàng.
- Annotation Based Configuration là một tính năng đặc lực hỗ trợ tạo lập bean thay vì phải XML.
- Servlet sẽ được nhúng sẵn trong Spring Boot, nên có thể bật và chạy Server Tomcat dễ dàng hơn.
- Spring Boot rất linh hoạt để có thể chạy cấu hình Java Beans, XML và kể cả là Database Transaction (giao dịch hoặc trao đổi).
- Gia tăng được năng suất trong quá trình lập trình.
- Giảm ở mức tối thiểu thời gian lập trình.
- Giúp người dùng mặc dù không có nhiều kiến thức lập trình vẫn có thể xây dựng được một ứng dụng.

CHƯƠNG 5 - CODING CONVENTION

1. Quy tắc đặt tên

3 cú pháp đặt tên có thể áp dụng và tuân theo:

- Cú pháp CamelCase (cú pháp lạc đà): Quy ước ký tự đầu tiên viết thường, những ký tự đầu tiên của chữ cái tiếp theo viết hoa.
- Cú pháp Pascal Case: Chữ cái đầu tiên viết hoa.
- Cú pháp Snake: Các chữ cái đều viết thường và các từ được phân tách bởi dấu gạch dưới.

Quy tắc đặt tên:

- Tên biến, tên lớp thường là cụm danh từ, danh từ hoặc tính từ. Khi đặt tên phải có ý nghĩa, không đặt theo kiểu viết tắt, không đặt tên chung chung, tên tối nghĩa.
- Ngoài ra một số trường hợp, tên biến cần phải thể hiện rõ kiểu dữ liệu của biến đó. Ví dụ: biến có kiểu List thì nên đặt tên là movieList, biến có kiểu Set thì nên đặt là movieSet, biến có kiểu Map thì nên đặt tên là movieMap, biến có kiểu Array thì nên đặt tên là studentArray,...
- Tên biến, hàm sẽ được đặt theo cú pháp lạc đà. Ví dụ: movieName, description,...
- Tên lớp đặt theo cú pháp Pascal. Ví dụ: User, Movie,...
- Tên Class nên có thêm những từ có hậu tố phía sau để thể hiện rõ hơn mục đích của Class đó, chẳng hạn như DataController, SecurityConfig,...
- Tên Package tuân theo quy tắc chung ở trên và phải viết thường. Ví dụ: com.example.demo.controller, com.example.demo.model,...
- Các hằng số đặt theo UPPER_CASE.

Sử dụng biến: Tránh tình trạng khai báo biến mà không sử dụng.

Import thư viện sử dụng: Chỉ import thư viện sử dụng cần thiết. Không sử dụng import tất cả.

2. Quy tắc số lượng

- Các hàm không được quá 30 dòng.
- Các lớp không vượt quá 500 dòng.
- Mỗi hàm không được vượt quá 5 tham số.
- Hàm chỉ thực hiện một việc duy nhất.
- 1 Dòng chỉ chứa 1 biến.
- Mỗi dòng không được quá 80 ký tự và các câu lệnh được lồng vào nhau không quá 4 cấp.

3. Quy tắc xuống hàng

- Nếu một hàm có nhiều cấp lồng nhau, mỗi cấp nên xuống dòng.
- Các đoạn code bằng cấp nên ở cùng một cột với nhau, dòng xuống hàng nên bắt đầu cùng cấp với dòng phía trên.
- Xuống hàng trước các toán tử (ví dụ như +, -, *, ?,...).

Ví dụ:

```
public SecurityFilterChain securityFilterChain(HttpSecurity http)
throws Exception {
    http
        .authorizeHttpRequests((requests) -> requests
            .antMatchers("/login", "/register", "/home",
                "/", "/images/**", "/saveData", "/css/**").permitAll()
            .anyRequest().authenticated()
        )
        .formLogin((form) -> form
            .loginPage("/login")
            .permitAll()
            .defaultSuccessUrl("/login?success=true")
            .failureUrl("/login?success=false")
        )
        .logout((logout) -> logout.permitAll());

    return http.build();
}
```

4. Quy tắc comment

- Hạn chế dùng comment để giải thích code hoặc chú thích những sự thật hiển nhiên.
- Không comment khi đóng thẻ/ngoặc:
- Xóa code không dùng, dùng comment out đoạn code đó.
- Comment cảnh báo hậu quả.
- Comment làm rõ ý nghĩa của code (nếu logic phức tạp).

5. Quy tắc WhiteSpace

5.1. White Space – thụt đầu dòng

- Xác định một đơn vị thụt đầu dòng cho toàn bộ mã nguồn của chương trình.
 - o 1 đơn vị thụt đầu dòng = 1 tab(*).
 - o Hoặc, 1 đơn vị thụt đầu dòng = 5 khoảng trắng.
- Nên dùng tab thay cho khoảng trắng.
 - o Đỡ tốn công nhập quá nhiều lần khoảng trắng.
 - o Có thể tùy chỉnh một đơn vị tab ứng với bao nhiêu khoảng trắng tùy ý.
- Hai dòng code cách nhau một bậc thì sẽ cách nhau một đơn vị thụt đầu dòng.

5.2. White Space – Dòng trống

- Những dòng code có quan hệ với nhau (cùng thực hiện một công việc) thì gom lại thành một block.
- Nghĩa là không có dòng trống giữa các đoạn code như trên.
- Hai block code thì cách nhau ít nhất một dòng trống.
- Đặt khoảng trắng sau dấu phẩy và dấu chấm phẩy.
- Đặt khoảng trắng xung quanh các toán tử.

6. Quy tắc ngoặc nhọn

Theo tiêu chuẩn Java: dấu “{” phải được đặt cùng dòng với các câu if, for, while,...

CHƯƠNG 6 – TESTING

A. STATIC TEST

1.1. Kế hoạch đánh giá

1.1.1. Tiêu chí của đánh giá

Tiêu chí đặt ra là kiểm tra tính đúng đắn của các tài liệu khác như: tài liệu về cơ sở dữ liệu, các tính năng trong hệ thống, dự đoán các lỗi có thể xảy ra trong hệ thống. Đồng thời kiểm tra xem các chức năng trong hệ thống có hoạt động đúng với yêu cầu của khách hàng đã đề ra hay chưa. Kiểm tra về độ tin cậy của hệ thống.

Kiểm tra xem các tài liệu có được viết đúng theo quy chuẩn. Các lỗi dễ xảy ra khi viết tài liệu như: không rõ ràng, không minh bạch, trừu tượng,...

1.1.2. Các lỗi tiềm năng

1.1.2.1. Lỗi tiềm năng trong hệ thống

- Lỗi thực thi sai logic.
- Lỗi sql injection: ứng dụng bị tấn công bởi các kỹ thuật sql.
- Lỗi không thể kết nối Database.
- Lỗi nhập sai thông tin.
- Lỗi thực thi sai logic.
- Lỗi xung đột giữa các dữ liệu.

1.1.2.2. Lỗi tiềm năng trong tài liệu cơ sở dữ liệu

- Lỗi thiếu ràng buộc khóa ngoại.
- Thiếu các ràng buộc về ngày tháng và kiểu dữ liệu.
- Lỗi nhập sai thông tin tiếng Việt.
- Lỗi về sai kiểu dữ liệu.
- Lỗi không có khóa chính.

B. DYNAMIC TEST

2.1. Unit Test

2.1.1. Kiểm thử giao diện đăng nhập cho khách hàng

STT	Mô tả Testcase	Các bước thực hiện	Kết quả mong đợi	Kết quả thực tế	Đánh giá
1	Để trống các trường dữ liệu khi đăng nhập.	1. Không nhập thông tin các trường. 2. Nhấn chọn “Đăng nhập”.	Không cho phép người dùng đăng nhập và hiển thị thông báo lỗi.	Không cho phép người dùng đăng nhập và hiển thị thông báo lỗi.	Đạt
2	Kiểm tra dữ liệu từng đầu vào.	Input: Username: htmaihoa792002gmail.com	Không cho phép người dùng đăng nhập và hiển thị thông báo lỗi.	Không cho phép người dùng đăng nhập và hiển thị thông báo lỗi “Vui lòng nhập đúng email”.	Đạt
3	Kiểm tra dữ liệu từng đầu vào.	Input: Username: htmaihoa792002@gmail.com Mật khẩu:	Không cho phép người dùng đăng nhập và	Không cho phép người dùng đăng nhập và	Đạt

		Maihoa792002	hiển thị thông báo lỗi.	hiển thị thông báo lỗi “Tên đăng nhập hoặc mật khẩu không chính xác”	
4	Nhập đúng và đầy đủ thông tin.	Input: Username: htmaihoa792002@gmail.com Mật khẩu: maihoa792002	Cho phép đăng nhập thành công và hiển thị giao diện Trang chủ.	Cho phép đăng nhập thành công và hiển thị giao diện Trang chủ.	Đạt

2.1.2. Kiểm thử chức năng đăng kí

STT	Mô tả Testcase	Các bước thực hiện	Kết quả mong đợi	Kết quả thực tế	Đánh giá
1	Để trống toàn bộ các trường thông tin khi đăng ký.	1. Không nhập thông tin các trường. 2. Nhấn chọn “Đăng ký”.	Không cho phép người dùng đăng ký và hiển thị thông	Không cho phép người dùng đăng nhập và hiển thị từng thông báo lỗi với nội dung	Đạt.

			báo lỗi cho từng trường thông tin.	phù hợp với lỗi mà người dùng gặp phải.	
2	Kiểm tra dữ liệu email có được dùng để đăng ký trước đó hay chưa.	Input: - Email: htmaihoa792002@gmail.com - Họ và tên: Hoàng Thị Mai Hoa. - Mật khẩu: 123456 - Nhập lại mật khẩu: 123456 - Số điện thoại: 0705426215 - Ngày sinh: 07/09/2002	Không cho phép người dùng đăng ký và hiển thị thông báo lỗi.	Không cho phép người dùng đăng ký và hiển thị thông báo lỗi “Email đăng kí đã tồn tại”.	Đạt
2	Kiểm tra mật khẩu xác nhận	Input: - Mật khẩu đầu vào: 12345678 - Mật khẩu xác nhận: 1234567	Không cho phép người dùng đăng ký và hiển thị thông báo lỗi	Không cho phép người dùng đăng ký và hiển thị thông báo nhập lại mật khẩu.	Đạt
3	Kiểm tra email đăng kí có	1. Input 1: Gmail: htmaihoa792002gmail.com	Không cho phép người dùng ký	Không cho phép người dùng đăng ký và hiển thị	Đạt

	hợp lệ hay không.		nhập và hiển thị thông báo lỗi.	thông báo lỗi “Email đăng kí không hợp lệ”	
		2. Input 2: Gmail: Htmaihoa792002@gmail	Không cho phép người dùng ký nhập và hiển thị thông báo lỗi.	Không cho phép người dùng đăng ký và hiển thị thông báo lỗi “Email đăng kí không hợp lệ”.	Đạt
		3. Input 3: Gmail: @gmail.comhtmaihoa792002	Không cho phép người dùng ký nhập và hiển thị thông báo lỗi.	Không cho phép người dùng ký nhập và hiển thị thông báo lỗi.	Đạt

5	Kiểm tra ngày sinh hợp lệ.	Input: 16/12/2022	Không cho người dùng đăng ký và hiển thị thông báo lỗi	Cho phép người dùng đăng kí	Không đạt
		Input: 16/12/2002	Cho phép đăng ký và hiển thị hộp thoại thông báo đăng ký thành công.	Cho phép đăng ký và hiển thị hộp thoại thông báo đăng ký thành công với nội dung “Bạn đã đăng ký thành công”.	Đạt
6	Kiểm tra khi đăng ký thành công.	Input: Input: - Email: maihoa792002@gmail.com - Họ và tên: Hoàng Thị Mai Hoa.	Cho phép đăng ký và hiển thị hộp thoại	Cho phép đăng ký và hiển thị hộp thoại thông báo đăng ký thành công	Đạt

		<ul style="list-style-type: none"> - Mật khẩu: 123456 - Nhập lại mật khẩu: 123456 - Số điện thoại: 0705426215 - Ngày sinh: 07/09/2002 	thông báo đăng ký thành công.	với nội dung “Đăng ký thành công !”.	
--	--	---	--	---	--

BẢNG KPI

MSSV	Họ và tên	Nội dung công việc	Đánh giá KPI
52000445	Hoàng Thị Mai Hoa (Nhóm trưởng)	<ul style="list-style-type: none"> - Xác định thực thể trong mô hình ERD. - Tạo Project Demo - Tạo project github - Thiết kế kiến trúc hệ thống, công nghệ được sử dụng, giải thích được lý do tại sao chọn kiến trúc và công nghệ này. - Hoàn thành mục Coding Convention và Testing trong báo cáo cuối kỳ. - Phần code demo: Chức năng đăng nhập, đăng kí, quản lý phim. 	90%
52000008	Trần Thị Ngọc Ánh	<ul style="list-style-type: none"> - Mô tả kế hoạch xây dựng dự án, thời gian thực hiện, mô hình phát triển phần mềm được sử dụng. - Viết đặc tả Use case. - Vẽ Sequence diagram. - Phần code demo: Front-end chức năng đặt vé xem phim + Chức năng quản lý thông tin lịch chiếu, ghế, Hiển thị thông tin khách hàng. 	80% (Không hoàn thành phần code)

52000172	Đoàn Thị Kim Yến	<ul style="list-style-type: none"> - Vẽ các sơ đồ: ERD, Mô hình quan hệ, sơ đồ usecase, class diagram - Tổng hợp nội dung file báo cáo. - Phần code demo: Chức năng hiển thị thông tin phim, tìm kiếm thông tin, backend chức năng đặt vé xem phim. - Thiết kế database. - Thiết kế trang chủ. 	80% (Không hoàn thành phần code)
----------	------------------	---	-------------------------------------

TÀI LIỆU THAM KHẢO

1. Model-view-controller Wikipedia,
<https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
2. What are the benefits of MVC, <https://blog.iandavis.com/2008/12/what-are-the-benefits-of-mvc/>
3. Spring Framework Wikipedia,
https://en.wikipedia.org/wiki/Spring_Framework
4. Spring MVC Tutorial, <https://www.javatpoint.com/spring-mvc-tutorial>
5. Spring MVC Tutorial | Full Course – YouTube,
<https://www.youtube.com/watch?v=g2b-NbR48Jo>
6. Spring Security, <https://spring.io/projects/spring-security>