ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

4/1/2019

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA CÔNG NGHỆ PHẦN MỀM

**BÁO CÁO ĐỒ ÁN CUỐI KÌ**

ĐỀ TÀI : MEGA CINEMA

**Giảng viên hướng dẫn :**

Thầy Huỳnh Tuấn Anh

**Lớp :**

SE214.J22.PMCL

**Sinh viên thực hiện :**

16521470 – Huỳnh Hạ Vy

16520175 – Vũ Nguyên Đăng

16521273 – Đỗ Ngọc Bích Trâm

Dương Phi Long

Lớp :

IS201.J23.PMCL

Sinh viên thực hiện :

16521470 – Huỳnh Hạ Vy

16520175 – Vũ Nguyên Đăng

16521273 – Đỗ Ngọc Bích Trâm

LỜI CÁM ƠN

Thành phố Hồ Chí Minh, ngày 2 tháng 4 năm 2019

Trên thực tế không có sự thành công nào mà không được gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp từ mọi người. Trong suốt thời gian từ khi bắt đầu học tập ở giảng đường đại học đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ vô cùng quý báo từ thầy cô, gia đình và bạn bè. Với lòng biết ơn sâu sắc nhất, em xin gửi đến thầy Huỳnh Tuấn Anh – trường Đại học Công Nghệ Thông Tin đã cùng với tri thức và tâm huyết của mình để truyền đạt cho chúng em vốn kiến thức vô cùng quý báo trong suốt quãng thời gian học tập.

Và đặc biệt, chúng em xin chân thành cám ơn thầy đã tận tâm hướng dẫn chúng em qua từng buổi học trên lớp cũng như những buổi nói chuyện, thảo luận về các vấn đề trong việc xây dựng, thực hiện đồ án. Nếu không có những lời hướng dẫn, dạy bảo của thầy thì em nghĩ đề tài này của chúng em khó có thể hoàn thiện được. Một lần nữa, em xin chân thành cám ơn thầy, cô.

Đồ án được thực hiện trong khoảng thời gian không quá ngắn, nhưng cũng không quá dài. Bước đầu đi vào thực tế, tìm hiểu về đề tài, kiến thức, kỹ năng của chúng em còn nhiều hạn chế, thiếu sót và còn rất nhiều bỡ nhỡ. Do vậy, sẽ không thể tránh khỏi những thiếu sót trong cách tìm hiểu, thực hiện cũng như trình bày. Em rất mong nhận được sự đóng góp ý kiến từ quý thầy, cô để đồ án được hoàn thiện và đạt được kết quả tốt hơn. Đồng thời, thông qua đó chúng em cũng có thể đúc kết được những kinh nghiệm quý báu cho bản thân trong các đề tài, dự án tiếp theo.

Sau cùng, em kính chúc thầy thật dồi dào sức khỏe, giàu niềm vui và luôn tràn đầy năng lượng để có thể tiếp tục thực hiện sứ mệnh cao quý của mình – truyền đạt tri thức cho các thế hệ tương lai.

Xin chân thành cám ơn quý thầy !

Nhóm sinh viên

NHẬN XÉT CỦA GIẢNG VIÊN

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………….

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

…………………………………………………………………………………………………………………………………………………………………………………………

MỤC LỤC

[**I.** **TỔNG QUAN ĐỀ TÀI** 8](#_Toc11792071)

[**1.** **ĐẶT VẤN ĐỀ** 8](#_Toc11792072)

[**2.** **GIẢI PHÁP** 10](#_Toc11792073)

[**3.** **MỤC TIÊU** 10](#_Toc11792074)

[**4.** **PHẠM VI** 10](#_Toc11792075)

[**5.** **CÔNG CỤ SỬ DỤNG** 11](#_Toc11792076)

[**6.** **BỐ CỤC BÁO CÁO** 11](#_Toc11792077)

[**II.** **XÁC ĐỊNH VÀ PHÂN TÍCH YÊU CẦU** 11](#_Toc11792078)

[**1.** **KHẢO SÁT HIỆN TRẠNG** 11](#_Toc11792079)

[**1.1.** **PHỎNG VẤN** 12](#_Toc11792080)

[**1.2.** **HIỆN TRẠNG TỔ CHỨC** 18](#_Toc11792081)

[**1.3.** **CÁC QUY TRÌNH NGHIỆP VỤ** 18](#_Toc11792082)

[**1.3.1.** **HOẠT ĐỘNG CỦA PHÒNG KINH DOANH, DỰ ÁN** 18](#_Toc11792087)

[**1.3.2.** **HOẠT ĐỘNG CỦA PHÒNG KẾ TOÁN – TÀI CHÍNH** 20](#_Toc11792088)

[**1.3.3.** **HOẠT ĐỘNG CỦA PHÒNG KỸ THUẬT** 21](#_Toc11792089)

[**1.3.4.** **HOẠT ĐỘNG CỦA BAN ĐIỀU HÀNH** 21](#_Toc11792090)

[**2.** **PHÂN TÍCH YÊU CẦU** 22](#_Toc11792094)

[**2.1.** **YÊU CẦU CHỨC NĂNG** 22](#_Toc11792097)

[**2.1.1.** **YÊU CẦU LƯU TRỮ** 22](#_Toc11792098)

[**2.1.2.** **YÊU CẦU VỀ TÍNH NĂNG** 22](#_Toc11792099)

[**2.2.** **YÊU CẦU PHI CHỨC NĂNG** 23](#_Toc11792100)

[**3.** **LỰA CHỌN PHƯƠNG ÁN THIẾT KẾ** 23](#_Toc11792101)

[**3.1.** **CÁC YÊU CẦU CỦA HỆ THỐNG** 23](#_Toc11792102)

[**3.1.1.** **CÁC YÊU CẦU CĂN BẢN, BẮT BUỘC** 23](#_Toc11792103)

[**3.1.2.** **CÁC YÊU CẦU CẦN THIẾT** 24](#_Toc11792104)

[**3.1.3.** **CÁC YÊU CẦU MONG MUỐN** 24](#_Toc11792105)

[**3.2.** **CÁC RÀNG BUỘC CỦA HỆ THỐNG** 25](#_Toc11792106)

[**3.3.** **CÁC PHƯƠNG ÁN ĐỀ XUẤT CHO HỆ THỐNG** 25](#_Toc11792107)

[**3.4.** **CHỌN LỰA PHƯƠNG ÁN THIẾT KẾ** 26](#_Toc11792108)

[**III. THIẾT KẾ HỆ THỐNG** 28](#_Toc11792109)

[**1.** **SƠ ĐỒ USECASE** 28](#_Toc11792110)

[**2.** **DANH SÁCH CÁC TÁC NHÂN CỦA HỆ THỐNG** 28](#_Toc11792111)

[**3.** **DANH SÁCH USECASE** 29](#_Toc11792112)

[**4.** **ĐẶC TẢ USECASE VÀ SƠ ĐỒ HOẠT ĐỘNG (ACTIVITY DIAGRAM)** 30](#_Toc11792113)

[**4.1.** **USECASE ĐĂNG NHẬP** 30](#_Toc11792115)

[**4.2.** **USECASE QUẢN LÝ KHO HÀNG** 31](#_Toc11792116)

[**4.2.1.** **USECASE THÊM SẢN PHẨM VÀO KHO** 31](#_Toc11792121)

[**4.2.2.** **USECASE CẬP NHẬT THÔNG TIN SẢN PHẨM TRONG KHO** 32](#_Toc11792122)

[**4.2.3.** **USECASE XÓA SẢN PHẨM TRONG KHO** 34](#_Toc11792123)

[**4.2.4.** **USECASE TRA CỨU SẢN PHẨM TRONG KHO** 36](#_Toc11792124)

[**4.3.** **USECASE QUẢN LÝ ĐẶT MUA VÉ** 37](#_Toc11792125)

[**4.4.** **USECASE QUẢN LÝ ĐẶT MUA BẮP NƯỚC VÀ CÁC SẢN PHẨM KHÁC** 39](#_Toc11792126)

[**4.5.** **USECASE XEM LỊCH SỬ GIAO DỊCH** 41](#_Toc11792127)

[**4.6.** **USECASE QUẢN LÝ PHIM** 42](#_Toc11792128)

[**4.6.1.** **USECASE THÊM MỚI PHIM** 42](#_Toc11792133)

[**4.6.2.** **USECASE CẬP NHẬT THÔNG TIN PHIM** 43](#_Toc11792134)

[**4.6.3.** **USECASE XÓA PHIM** 44](#_Toc11792135)

[**4.6.4.** **USECASE TRA CỨU PHIM** 46](#_Toc11792136)

[**4.7.** **USECASE QUẢN LÝ LỊCH CHIẾU** 47](#_Toc11792137)

[**4.7.1.** **USECASE THÊM MỚI LỊCH CHIẾU** 47](#_Toc11792139)

[**4.7.2.** **USECASE CẬP NHẬT LỊCH CHIẾU** 48](#_Toc11792140)

[**4.7.3.** **USECASE XÓA LỊCH CHIẾU** 49](#_Toc11792141)

[**4.7.4.** **USECASE TRA CỨU LỊCH CHIẾU** 51](#_Toc11792142)

[**4.8.** **USECASE QUẢN LÝ PHÒNG CHIẾU** 52](#_Toc11792143)

[**4.8.1.** **USECASE THÊM MỚI PHÒNG CHIẾU** 52](#_Toc11792145)

[**4.8.2.** **USECASE CẬP NHẬT PHÒNG CHIẾU** 53](#_Toc11792146)

[**4.8.3.** **USECASE XÓA PHÒNG CHIẾU** 54](#_Toc11792147)

[**4.8.4.** **USECASE TRA CỨU PHÒNG CHIẾU** 56](#_Toc11792148)

[**4.9.** **USECASE QUẢN LÝ NHÂN VIÊN** 57](#_Toc11792149)

[**4.9.1.** **USECASE THÊM MỚI NHÂN VIÊN** 57](#_Toc11792150)

[**4.9.2.** **USECASE CẬP NHẬT THÔNG TIN NHÂN VIÊN** 58](#_Toc11792151)

[**4.9.3.** **USECASE XÓA NHÂN VIÊN** 60](#_Toc11792152)

[**4.9.4.** **USECASE TRA CỨU NHÂN VIÊN** 61](#_Toc11792153)

[**4.10.** **USECASE LẬP BÁO CÁO DOANH THU** 62](#_Toc11792154)

[**5.** **SƠ ĐỒ TRÌNH TỰ (SEQUENCE DIAGRAM) VÀ SƠ ĐỒ LỚP (CLASS DIAGRAM)** 63](#_Toc11792155)

[**5.1.** **ĐĂNG NHẬP** 63](#_Toc11792157)

[**5.1.1.** **SEQUENCE DIAGRAM** 63](#_Toc11792160)

[**5.1.2.** **CLASS DIAGRAM** 63](#_Toc11792161)

[**5.2.** **QUẢN LÝ KHO HÀNG** 64](#_Toc11792162)

[**5.2.1.** **THÊM SẢN PHẨM VÀO KHO** 64](#_Toc11792164)

[**5.2.1.1.** **SEQUENCE DIAGRAM** 64](#_Toc11792165)

[**5.2.1.2.** **CLASS DIAGRAM** 66](#_Toc11792166)

[**5.2.2.** **CẬP NHẬT THÔNG TIN SẢN PHẨM TRONG KHO** 66](#_Toc11792167)

[**5.2.2.1.** **SEQUENCE DIAGRAM** 66](#_Toc11792168)

[**5.2.2.2.** **CLASS DIAGRAM** 67](#_Toc11792169)

[**5.2.3.** **XÓA SẢN PHẨM TRONG KHO** 68](#_Toc11792170)

[**5.2.3.1.** **SEQUENCE DIAGRAM** 68](#_Toc11792171)

[**5.2.3.2.** **CLASS DIAGRAM** 69](#_Toc11792172)

[**5.2.4.** **TRA CỨU SẢN PHẨM TRONG KHO** 70](#_Toc11792173)

[**5.2.4.1.** **SEQUENCE DIAGRAM** 70](#_Toc11792174)

[**5.2.4.2.** **CLASS DIAGRAM** 71](#_Toc11792175)

[**5.3.** **QUẢN LÝ ĐẶT MUA VÉ** 72](#_Toc11792176)

[**5.3.1.** **SEQUENCE DIAGRAM** 72](#_Toc11792178)

[**5.3.2.** **CLASS DIAGRAM** 72](#_Toc11792179)

[**5.4.** **QUẢN LÝ ĐẶT MUA BẮP NƯỚC VÀ CÁC SẢN PHẨM KHÁC** 73](#_Toc11792180)

[**5.4.1.** **SEQUENCE DIAGRAM** 73](#_Toc11792190)

[**5.4.2.** **CLASS DIAGRAM** 74](#_Toc11792191)

[**5.5.** **XEM LỊCH SỬ GIAO DỊCH** 75](#_Toc11792192)

[**5.5.1.** **SEQUENCE DIAGRAM** 75](#_Toc11792203)

[**5.5.2.** **CLASS DIAGRAM** 76](#_Toc11792204)

[**5.6.** **QUẢN LÝ PHIM** 77](#_Toc11792205)

[**5.7.** **QUẢN LÝ LỊCH CHIẾU** 85](#_Toc11792206)

[**5.7.1.** **THÊM MỚI LỊCH CHIẾU** 85](#_Toc11792219)

[**5.7.1.1.** **SEQUENCE DIAGRAM** 85](#_Toc11792220)

[**5.7.1.2.** **CLASS DIAGRAM** 86](#_Toc11792221)

[**5.7.2.** **CẬP NHẬT THÔNG TIN LỊCH CHIẾU** 87](#_Toc11792222)

[**5.7.2.1.** **SEQUENCE DIAGRAM** 87](#_Toc11792223)

[**5.7.2.2.** **CLASS DIAGRAM** 88](#_Toc11792224)

[**5.7.3.** **XÓA LỊCH CHIẾU** 89](#_Toc11792225)

[**5.7.3.1.** **SEQUENCE DIAGRAM** 89](#_Toc11792226)

[**5.7.3.2.** **CLASS DIAGRAM** 90](#_Toc11792227)

[**5.7.4.** **TRA CỨU THÔNG TIN LỊCH CHIẾU** 91](#_Toc11792228)

[**5.7.4.1.** **SEQUENCE DIAGRAM** 91](#_Toc11792229)

[**5.7.4.2.** **CLASS DIAGRAM** 92](#_Toc11792230)

[**5.8.** **QUẢN LÝ PHÒNG CHIẾU** 93](#_Toc11792231)

[**5.8.1.** **THÊM MỚI PHÒNG CHIẾU** 93](#_Toc11792245)

[**5.8.1.1.** **SEQUENCE DIAGRAM** 93](#_Toc11792246)

[**5.8.1.2.** **CLASS DIAGRAM** 94](#_Toc11792247)

[**5.8.2.** **CẬP NHẬT THÔNG TIN PHÒNG CHIẾU** 95](#_Toc11792248)

[**5.8.2.1.** **SEQUENCE DIAGRAM** 95](#_Toc11792249)

[**5.8.2.2.** **CLASS DIAGRAM** 96](#_Toc11792250)

[**5.8.3.** **XÓA PHÒNG CHIẾU** 97](#_Toc11792251)

[**5.8.3.1.** **SEQUENCE DIAGRAM** 97](#_Toc11792252)

[**5.8.3.2.** **CLASS DIAGRAM** 98](#_Toc11792253)

[**5.8.4.** **TRA CỨU PHÒNG CHIẾU** 99](#_Toc11792254)

[**5.8.4.1.** **SEQUENCE DIAGRAM** 99](#_Toc11792255)

[**5.8.4.2.** **CLASS DIAGRAM** 100](#_Toc11792256)

[**5.9.** **QUẢN LÝ NHÂN VIÊN** 101](#_Toc11792257)

[**5.9.1.** **THÊM MỚI NHÂN VIÊN** 101](#_Toc11792264)

[**5.9.1.1.** **SEQUENCE DIAGRAM** 101](#_Toc11792265)

[**5.9.1.2.** **CLASS DIAGRAM** 102](#_Toc11792266)

[**5.9.2.** **CẬP NHẬT NHÂN VIÊN** 103](#_Toc11792267)

[**5.9.2.1.** **SEQUENCE DIAGRAM** 103](#_Toc11792268)

[**5.9.2.2.** **CLASS DIAGRAM** 104](#_Toc11792269)

[**5.9.3.** **XÓA NHÂN VIÊN** 105](#_Toc11792270)

[**5.9.3.1.** **SEQUENCE DIAGRAM** 105](#_Toc11792271)

[**5.9.3.2.** **CLASS DIAGRAM** 106](#_Toc11792272)

[**5.9.4.** **TRA CỨU NHÂN VIÊN** 107](#_Toc11792273)

[**5.9.4.1.** **SEQUENCE DIAGRAM** 107](#_Toc11792274)

[**5.9.4.2.** **CLASS DIAGRAM** 108](#_Toc11792275)

[**5.10.** **BÁO CÁO DOANH THU** 109](#_Toc11792276)

[**5.10.1.** **SEQUENCE DIAGRAM** 109](#_Toc11792278)

[**5.10.2.** **CLASS DIAGRAM** 110](#_Toc11792279)

[**5.11.** **CLASS DIAGRAM TỔNG** 111](#_Toc11792280)

[**5.12.** **ĐẶC TẢ DIAGRAM** 112](#_Toc11792281)

[**6.** **ÁNH XẠ DỮ LIỆU QUAN HỆ TỪ MÔ HÌNH HƯỚNG ĐỐI TƯỢNG** 136](#_Toc11792282)

[**6.1.** **MÔ HÌNH QUAN HỆ** 136](#_Toc11792284)

[**6.2.** **MÔ TẢ CÁC KIỂU DỮ LIỆU** 136](#_Toc11792285)

[**7.** **THIẾT KẾ GIAO DIỆN** 141](#_Toc11792286)

[**7.1.** **ĐĂNG NHẬP** 141](#_Toc11792287)

[**7.2.** **GIAO DIỆN CHÍNH** 141](#_Toc11792288)

[**7.3.** **QUẢN LÝ LỊCH CHIẾU** 142](#_Toc11792289)

[**7.4.** **QUẢN LÝ PHIM** 142](#_Toc11792290)

[**7.5.** **QUẢN LÝ PHÒNG CHIẾU** 143](#_Toc11792291)

[**7.6.** **QUẢN LÝ KHO** 143](#_Toc11792292)

[**7.7.** **BÁN VÉ XEM PHIM** 144](#_Toc11792293)

[**7.8.** **BÁN BẮP NƯỚC VÀ CÁC SẢN PHẨM KHÁC** 145](#_Toc11792294)

[**7.9.** **XEM LỊCH SỬ GIAO DỊCH** 146](#_Toc11792295)

[**7.10.** **QUẢN LÝ NHÂN VIÊN** 148](#_Toc11792296)

[**7.11.** **BÁO CÁO VÀ BIỂU ĐỒ** 149](#_Toc11792297)

[**III.** **KẾT LUẬN:** 149](#_Toc11792298)

1. TỔNG QUAN ĐỀ TÀI
2. ĐẶT VẤN ĐỀ

Chúng ta đang sống trong một thời đại mới, thời đại phát triển rực rỡ của CNTT. CNTT đã ở một bước phát triển cao đó là số hóa tất cả các dữ liệu thông tin, luân chuyển mạnh mẽ và kết nối tất cả chúng ta lại với nhau. Mọi loại thông tin, số liệu âm thanh, hình ảnh có thể được đưa về dạng kỹ thuật số để bất kỳ máy tính nào cũng có thể lưu trữ, xử lý và chuyển tiếp cho nhiều người. Những công cụ và sự kết nối của thời đại kỹ thuật số cho phép chúng ta dễ dàng thu thập, chia sẻ thông tin và hành động trên cơ sở những thông tin này theo phương thức hoàn toàn mới, kéo theo hàng loạt sự thay đổi về các quan niệm, các tập tục, các thói quen truyền thống, và thậm chí cả cách nhìn các giá trị trong cuộc sống. CNTT đến với từng người dân, từng người quản lý, nhà khoa học, người nông dân, bà nội trợ, học sinh tiểu học, …. Vv Không có lĩnh vực nào, không có nơi nào không có mặt của CNTT. Công nghệ thông tin là một trong các động lực quan trọng nhất của sự phát triển, ứng dụng và phát triển công nghệ thông tin ở nước ta nhằm góp phần giải phóng sức mạnh vật chất, trí tuệ và tinh thần của toàn dân tộc, thúc đẩy công cuộc đổi mới, phát triển nhanh và hiện đại hoá các ngành kinh tế, tăng cường năng lực cạnh tranh của các doanh nghiệp, hỗ trợ có hiệu quả cho quá trình hội nhập kinh tế quốc tế, nâng cao chất lượng cuộc sống của mọi người.

Nhận thấy được những ích lợi của công nghệ thông tin, các doanh nghiệp kinh doanh, các công ty giải trí cũng đã và đang áp dụng công nghệ thông tin vào việc quản lý thông tin. Thay vào đó, với sự phát triển nhanh chóng của ngành công nghiệp giải trí, các rạp chiếu phim đang dần được xây dựng ngày càng nhiều trên thế giới để phục vụ cho nhu cầu giải trí thiết yếu của khán giả. Tuy nhiên, việc quản lý hệ thống tại các hệ thống rạp chiếu luôn gặp nhiều khó khăn. Vì vậy, việc tạo ra hệ thống quản lí rạp phim nhằm hỗ trợ việc quản lý một cách tốt nhất hệ thống rạp chiếu, nhân viên, thiết bị, sự kiện, .. vv là một nhu cầu tất yếu và giúp cho việc quản lý hệ thống rạp chiếu ngày càng trở nên đơn giản hơn, nhanh chóng hơn và dễ dàng hơn.

Để giải quyết vấn đề trên, cần một giải pháp mang tính thực tiễn, cần xây dựng nên một hệ thống phục vụ công tác quản lý các rạp chiếu phim, đồng thời tạo nên kênh thông tin chính thức để quảng bá thương hiệu, thông tin, tin tức, .. vv đến với người dùng. Hệ thống được xây dựng phải là một hệ thống quản lý tích hợp, lưu trữ thông tin tập trung và dễ dàng tương tác với người dùng. Bên cạnh đó cũng cần chuẩn hóa qui trình các công việc, để việc ứng dụng công nghệ thông tin thực sự có hiệu quả.

1. GIẢI PHÁP

Từ thực tế như vậy, nhóm chúng tôi đã quyết định thực hiện đề tài này với mục tiêu xây dựng một hệ thống phần mềm giúp tự động hóa tối đa các giai đoạn và các nghiệp vụ quản lý rạp chiếu phim, nhằm khắc phục các hạn chế của các hệ thống quản lí hiện nay.

1. MỤC TIÊU

Xây dựng ứng dụng hỗ trợ quản lý và quảng bá Hệ thống rạp chiếu phim Mega Cinema đáp ứng những yêu cầu sau:

* Quản lý quá trình nhập xuất thông tin của phim, lịch chiếu và các thông tin của rạp phim một cách nhanh chóng và dễ dàng.
* Quản lý và kết xuất báo cáo đáng tin cậy và tự động.
* Quản lý đầy đủ thông tin rạp chiếu, phòng chiếu.
* Quản lý thông tin và giao dịch của khách hàng.
* Quản lý quá trình đặt vé kết hợp với các sự kiện khuyến mãi và giải quyết

vấn đề truy xuất đồng thời trong quá trình đặt vé.

* Thông tin được lưu trữ trong cơ sở dữ liệu, giúp truy xuất, sao lưu và phục hồi dễ dàng, đồng bộ, nhanh chóng và đáng tin cậy.

1. PHẠM VI

Do thời gian thực hiện đồ án có hạn, việc xây dựng ứng dụng quản lý hệ thống rạp chiếu phim Mega Cinema sẽ tập trung vào các chức năng chính bao gồm: quản lý thông tin phim, quản lý các thông tin rạp chiếu, phòng chiếu, mua bán vé và các sản phẩm bắp nước, quản lý lịch chiếu, quản lý nhân viên và quản lý thông tin cũng như giao dịch của người dùng.

1. CÔNG CỤ SỬ DỤNG

Trong quá trình thực hiện đồ án, nhóm đã sử dụng một số phần mềm phục vụ cho việc tìm hiểu và xây dựng đề tài bao gồm:

--‐ My SQL: Hệ quản trị CSDL quan hệ.

--‐ Netbeans IDE 8.2: IDE để xây dựng app.

--‐ StarUML: vẽ phân tích các dạng sơ đồ.

1. BỐ CỤC BÁO CÁO

* Tổng quan về đề tài.

Giới thiệu tống quan nội dung đề tài. Bao gồm: lý do lựa chọn đề tài, nội dung và phạm vi nghiên cứu của đề tài, bố cục báo cáo.

* Xác định và phân tích yêu cầu

Trình bày sơ lược về sơ đồ tổ chức của rạp chiếu phim Mega Cinema và những qui trình trong công tác quản lý rạp chiếu phim. Từ đó đưa ra phương pháp tiếp cận và giải quyết vấn đề.

* Thiết kế hệ thống

Trình bày chi tiết quá trình phân tích và thiết kế hệ thống dựa trên ngôn ngữ mô hình hóa UML.

* Kết luận

1. XÁC ĐỊNH VÀ PHÂN TÍCH YÊU CẦU
2. KHẢO SÁT HIỆN TRẠNG
   1. PHỎNG VẤN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| KẾ HOẠCH PHỎNG VẤN TỔNG QUAN  Hệ thống: Mega Cinema  Người lập: Huỳnh Hạ Vy Ngày lập: 14/03/2019 | | | | |
| STT | Chủ đề | Yêu cầu | Ngày bắt đầu | Ngày kết thúc |
| 1 | Quy trình quản lý phim | Nắm rõ quy trình lưu trữ phim, phân suất chiếu và định giá vé. | 14/03/2019 | 14/03/2019 |
| 2 | Quy trình hoạch định khuyến mãi và tin tức | Nắm rõ quy trình đề ra và áp dụng các sự kiện khuyến mãi trong rạp phim | 14/03/2019 | 14/03/2019 |
| 3 | Quy trình quản lý khách hàng | Nắm được quy trình lưu trữ và khai thác thông tin khách hàng để làm tốt công tác chăm sóc khách hàng | 14/03/2019 | 14/03/2019 |
| 4 | Quy trình thống kê | Biết rõ quy trình thực hiện các báo cáo lượng vé và thức ăn bán được, dựa vào dữ liệu trong hệ thống | 14/03/2019 | 12/03/2019 |
| 5 | Hệ thống máy móc, phần mềm | Tìm hiểu kỹ về tài nguyên máy móc, trang thiết bị, phần mềm, hệ điều hành đang sử dụng của hệ thống | 14/03/2019 | 14/03/2019 |

Bảng 2.1. Mẫu kế hoạch phỏng vấn

|  |  |
| --- | --- |
| **BẢNG KẾ HOẠCH PHỎNG VẤN**  Hệ thống: Mega Cinema | |
| *Người được phỏng vấn*: Ngô Tấn Phát | *Phân tích viên*: Huỳnh Hạ Vy |
| *Vị trí/phương tiện*: The coffee house | * *Thời gian bắt đầu*: 8h ngày 14/03/2019 * *Thời gian kết thúc*: 11h30 ngày 14/03/2019 |
| *Mục tiêu:*  Thu thập và hiểu được những quy trình hoạt động của hệ thống rạp phim |  |
| *Chi tiết buổi phỏng vấn*  --‐ Giới thiệu  --‐ Tổng quan về hệ thống  --‐ Tổng quan về buổi phỏng vấn:  Chủ đề 1: Quy trình quản lý phim  Chủ đề 2: Quy trình hoạch định khuyến mãi và tin tức  Chủ đề 3: Quy trình quản lý khách hàng  Chủ đề 4: Quy trình thống kê, báo cáo  Chủ đề 5: Hệ thống máy móc, phần mềm | *Thời gian ước lượng:*  --‐ 5 phút  --‐ 15 phút  --‐ 3 tiếng |
| Quan sát tổng quan |  |
| Phát sinh ngoài dự kiến |  |

Bảng 2.2. Bảng kế hoạch phỏng vấn

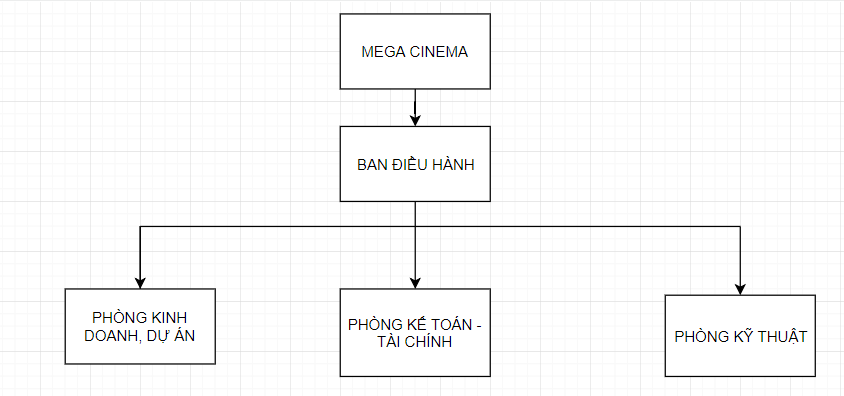
|  |  |  |
| --- | --- | --- |
| Chủ đề 1 | Câu hỏi 1: Việc phân chia suất chiếu của rạp cho từng phòng như thế nào ? | Việc phân chia suất chiếu được thực hiện dựa trên thời lượng của từng phim, thông thường thì sẽ dài hơn thời lượng của phim 30 phút để các nhân viên dọn vệ sinh phòng chiếu. |
| Câu hỏi 2: Thời gian cách quãng giữa các phòng chiếu như thế nào để tránh tình trạng lượng khách ra vào phòng chờ quá đông ? | Thời gian cách quãng giữa các phòng cách nhau khoảng 1 giờ nhằm khi phim kết thúc lượng khách ra phòng chờ không quá đông để không ảnh hưởng tới việc phục vụ khách hàng. |
| Câu hỏi 3: Giá vé đối với những phim bom tấn có thay đổi như thế nào ? | Đối với các phim bom tấn thì giá vé sẽ được áp dụng tăng 5% so với mức giá quy định và không áp dụng các hình thức khuyến mãi. |
| Chủ đề 2 | Câu hỏi 1: Ngoài hình thức khuyến mãi giảm giá vé còn có khuyến mãi nào khác không? | Ngoài hình thức khuyến mãi giá vé thì nhân các dịp lễ đặc biệt sẽ có những hình thức tặng quà cho các khách hàng thân thiết nhằm giúp cho việc chăm sóc khách hàng diễn ra tốt hơn. |
| Câu hỏi 2: Tin tức cần được cập nhật như thế nào? | Tin tức phải được cập nhật thường xuyên để khách hàng có thể dễ dàng nắm bắt được. |
| Chủ đề 3 | Câu hỏi 1: Cách tính điểm của khách hàng như thế | Khách hàng sẽ được chia làm nhiều cấp bậc khác nhau theo các hạng mục điểm. Việc tăng điểm sẽ |

|  |  |  |
| --- | --- | --- |
|  | nào và việc tăng điểm theo hạng mục ra sao? | dựa trên hóa đơn khách hàng thanh toán vào việc mua vé. Cụ thể là cứ mỗi 1000 đồng thì khách hàng tích lũy được 1 điểm. |
| Câu hỏi 2: Khi khách hàng thành viên trong thời gian quá lâu không sử dụng thẻ thì thông tin có còn được lưu trữ hay không? | Khách hàng là yếu tố chính cho việc hoạt động của rạp phim. Vì vậy việc lưu trữ thông tin khách hàng là cần thiết, bất kể rằng khách hàng có thường xuyên quan tâm đến rạp chiếu phim hay không. |
| Câu hỏi 3: Khách hàng thành viên theo các cấp bậc có độ ưu tiên khác nhau không ? | Khách hàng thành viên sẽ được phân theo các cấp độ và có độ ưu tiên khác nhau:  0 – 1499 : Thành viên mới chỉ được tích lũy điểm  và quy đổi thành quà tặng.  1500 – 2999 : Thành viên thân thiết được chiết khấu 5% trên hóa đơn.  Trên 3000: Thành viên VIP sẽ được chiết khấu 10% trên hóa đơn. |
| Câu hỏi 4: Trường hợp khách hàng chưa có thẻ thành viên thì có lưu thông tin hay không ? | Khách hàng khi không có thẻ thành viên sẽ được lưu chung là khách hàng vãng lai trên các hóa đơn nhưng không lưu các thông tin chi tiết như khách hàng thành viên. Khách hàng chưa là thành viên thì không được phép sử dụng chức năng đặt vé online, mà chỉ có thể mua trực tiếp tại rạp. |

|  |  |  |
| --- | --- | --- |
| Chủ đề 4 | Câu hỏi 1: Thời gian thống kê lượng vé bán được định kỳ là bao lâu? | Việc thống kê lượng vé bán được sẽ định kỳ theo thời gian 1 tuần và một khoảng thời gian do người quản lý cung cấp nếu cần cho việc báo cáo số liệu |
| Câu hỏi 2: Hệ thống cần những thống kê nào ? | Hệ thống gồm những thống kê:  Thống kê vé bán ra theo từng sản phẩm.  Thống kê doanh thu của rạp phim theo thời gian xác định. |
| Chủ đề 5 | Câu hỏi 1: Việc kiểm tra các thiết bị của rạp phim được thực hiện theo định kỳ bao lâu ? | Việc kiểm tra các trang thiết bị của rạp phim được kiểm tra theo định kỳ khi từng suất chiếu kết thúc nhằm đảm bảo cho việc các trang thiết bị luôn được đảm bảo hoạt động tốt. |
| Câu hỏi 2: Khi hệ thống hư hỏng thì sẽ được xử lý như thế nào? | Khi hệ thống bị hư hỏng thì sẽ dời phòng chiếu sang một phòng chiếu dự phòng và tiến hành sửa chữa hệ thống thiết bị ngay nhằm đảm bảo đúng tiến độ hoạt động của rạp phim. |
| Câu hỏi 3: Hệ thống máy tính có phần cứng như thế nào ? | Hiện tại hệ thống rạp phim đã trang bị một máy server và hệ thống máy tính dành cho các nhân viên. |
| Câu hỏi 4: Các cụm rạp khác nhau có hình thức quản lý ( giá vé, khuyến mãi …) khác nhau hay không? | Hiện tại quy mô rạp còn nhỏ nên việc quản lý là như nhau nhưng sau này sẽ có xem xét tình hình để có hình thức quản lý khác nhau. Và phân ra thành các cụm rạp để khách hàng có thể đặt vé qua mạng. |

Bảng 2.3. Bảng câu hỏi và ghi nhận trả lời

* 1. HIỆN TRẠNG TỔ CHỨC



* 1. CÁC QUY TRÌNH NGHIỆP VỤ

1. 3. 1. HOẠT ĐỘNG CỦA PHÒNG KINH DOANH, DỰ ÁN

* **Nghiệp vụ Quản lí phim:** là việc mua bản quyền phim từ các nguồn sản xuất và tiến hành phân phối đến khách hàng để phục vụ cho công việc kinh doanh của rạp. Bao gồm các quy trình nhập phim, định giá vé, phân chia suất chiếu và các quy trình khác.
* Quy trình nhập phim
* Nhập phim từ nguồn sản xuất phim
* Nhập thông tin phim và lưu trữ vào cơ sở dữ liệu: Phim sau khi được nhập về sẽ được lưu trữ các thông tin vào hệ thống quản lý. Các thông tin bao gồm: tên phim, poster, thời lượng chiếu, đạo diễn, diễn viên, năm sản xuất, nước sản xuất, thể loại, nội dung phim, trailer phim.
* Quy trình định giá vé
* Định giá vé
* Giá vé mặc định ban đầu đối với khách vãng lai và thành viên mới, chưa áp dụng khuyến mãi
* Nhận thông tin khuyến mãi từ phòng Marketing và định lại giá vé theo thông tin khuyến mãi
* Nhập giá vé vào cơ sở dữ liệu
* Quy trình phân suất chiếu
* Phân chia suất chiếu
* Nhập thông tin suất chiếu: nhập thông tin suất chiếu vào hệ thống quản lý. Một bộ phim có thể có nhiều suất chiếu, thông tin từng suất chiếu bao gồm: phòng chiếu, thời gian chiếu.
* Các quy trình khác
* **Nghiệp vụ Marketing:** là việc đề ra các chiến lược thu hút thêm khách hàng mới và chương trình khuyến mãi phục vụ cho lợi ích của họ. Gồm quy trình hoạch định khuyến mại và các quy trình khác.
* Quy trình hoạch định khuyến mãi
* Đề ra các sự kiện khuyến mãi
* Thông tin tới bộ phận định giá vé, bộ phận chăm sóc khách hàng và các bộ phận liên quan đến sự kiện khuyến mãi
* Các quy trình khác
* **Nghiệp vụ Chăm sóc khách hàng** là việc tạo mối quan hệ thân thiết giữa rạp phim và khách hàng, thông báo và giúp khách hàng cập nhật những thông tin mới của rạp phim, quan trọng hơn hết là các sự kiện khuyến mãi. Gồm quy trình quản lý khách hàng và các quy trình khác.
* Quy trình quản lý khách hàng
* Lưu trữ thông tin khách hàng: chỉ những khách hàng đăng kí làm thẻ thành viên mới được lưu trữ thông tin
* Giải đáp thắc mắc khách hàng
* Thực hiện các sự kiện do bộ phận Marketing đề ra
* Các quy trình khác
* **Nghiệp vụ Bán hàng** là việc phục vụ khách hàng đến xem phim và là việc chính để duy trì được hoạt động của rạp phim. Gồm quy trình bán vé và quy trình bán thức ăn.
* Quy trình bán vé
* Khách hàng đến quầy bán vé chọn phim, suất chiếu và ghế ngồi trực tiếp tại quầy bán vé. Nhân viên chọn phim, mở màn hình ghế ngồi cho khách xem trực tiếp và chọn, xuất vé, tính tiền cho khách hàng.
* Quy trình bán bắp nước và các sản phẩm khác
* Khách hàng đến quầy thức ăn mua bắp, nước theo thực đơn định sẵn. Nhân viên lấy bắp, nước và các sản phẩm khác tính tiền, sau đó xuất hóa đơn cho khách hàng.
* **Các nghiệp vụ khác.**
  + 1. HOẠT ĐỘNG CỦA PHÒNG KẾ TOÁN – TÀI CHÍNH
* **Nghiệp vụ Thống kê** là việc thực hiện các báo cáo dựa vào dữ liệu nhằm hỗ trợ cho Phòng Kinh Doanh theo dõi hoạt động của rạp phim. Gồm quy trình tính lương và thanh toán lương nhân viên, tính, quy trình thống kê lượng vé bán ra và thức ăn, các quy trình khác.
* Quy trình tính lương và thanh toán nhân viên
* Lấy thông tin hoạt động của nhân viên từ hệ thống quản lý
* Xác định thời gian làm việc của nhân viên
* Tính lương của nhân viên và cập nhật cơ sở dữ liệu vào hệ thống quản lý
* Quy trình thống kê lượng vé bán ra và thức ăn
* Lấy thông tin bán vé và thức ăn từ hệ thống quản lý bán hàng
* Tính toán tổng số tiền thu vào từ việc bán vé và thức ăn
* Thống kê số tiền theo từng thời điểm và cập nhật vào cơ sở dữ liệu
* Các quy trình khác
* **Các nghiệp vụ khác**
  + 1. HOẠT ĐỘNG CỦA PHÒNG KỸ THUẬT
* **Nghiệp vụ Quản lý rạp chiếu** là việc hỗ trợ về phần kỹ thuật của quy trình chiếu phim phục vụ khách hàng, đồng thời quản lý các trang thiết bị của rạp chiếu, tránh việc hỏng hóc hay thất lạc. Gồm quy trình quản lý thông tin phòng chiếu và các quy trình khác.
* Quy trình quản lý thông tin phòng chiếu
* Nhập, lưu trữ và quản lý các thông tin về phòng chiếu vào hệ thống quản lý, bao gồm: loại phòng, sơ đồ ghế, máy chiếu, âm thanh, diên tích, tình trạng, mã bảo vệ, trang thiết bị khác.
* Các quy trình khác
* **Nghiệp vụ khác.**
  + 1. HOẠT ĐỘNG CỦA BAN ĐIỀU HÀNH
* **Nghiệp vụ Quản lý nhân sự** giúp cho trưởng bộ phận giám sát được hoạt động của tất cả nhân viên trong rạp phim. Gồm quy trình quản lý thông tin nhân viên, quy trình quản lý hoạt động nhân viên và các quy trình khác.
* Quy trình quản lý thông tin nhân viên
* Tuyển nhân viên
* Nhập thông tin nhân viên
* Cập nhật thông tin nhân viên
* Xóa nhân viên
* Quy trình quản lý hoạt động nhân viên
* Quản lý giờ làm và lịch sử hoạt động của nhân viên
* Báo cáo định kì hoạt động của nhân viên
* Các quy trình khác
* **Các nghiệp vụ khác**

2. PHÂN TÍCH YÊU CẦU
4. 1. YÊU CẦU CHỨC NĂNG
      1. YÊU CẦU LƯU TRỮ

Tính năng lưu trữ là yêu cầu quan trọng nhất của hệ thống là cơ sở để thực hiện phần lớn các tính năng của hệ thống, để đảm bảo việc quản lý rạp phim có hiệu quả hệ thống cần lưu trữ những thông tin sau:

* Thông tin về phim: tên phim, poster, trailer, thời lượng, đạo diễn, năm sản xuất, ngày khởi chiếu, độ tuổi quy định, rating, quốc gia, nội dung, thể loại.
* Thông tin về người dùng: tên người dùng, giới tính, ngày sinh, địa chỉ, điện thoại, email, cmnd, ngay đăng ký, mật khấu, loại người dùng.
* Thông tin về khuyến mãi: tên khuyến mãi, hình, nội dung, ngày bắt đầu, ngày kết thúc.
* Thông tin về tin tức: tiêu đề, nội dung, ngày đăng.
* Thông tin về lịch chiếu: tên phim, ngày chiếu, phòng chiếu, giờ chiếu, giá, rạp chiếu, loại suất chiếu
  + 1. YÊU CẦU VỀ TÍNH NĂNG

Ngoài chức năng lưu trữ, hệ thống còn phải đáp ứng được những yêu cầu sau:

* Chức năng truy vấn: hệ thống phải luôn đáp ứng các yêu cầu truy vấn từ người dùng như: tìm thông tin phim, tìm thông tin người dùng, thông tin khuyến mãi, thông tin lịch chiếu, thông tin tin tức…
* Chức năng thống kê, báo cáo: giúp nhân viên quản lý theo dõi tình hình chung của rạp chiếu phim.
* Chức năng cập nhật: hệ thống phải đảm bảo chức năng cập nhật được thuận tiện và tránh tối đa lỗi do sơ xuất của người sử dụng.
* Các chức năng tự động: giúp người dùng thuận tiện hơn trong thao tác như tự động tính tiền vé, tự động phát sinh mã vé, ...
  1. YÊU CẦU PHI CHỨC NĂNG
* Yêu cầu về giao diện: sử dụng đa ngôn ngữ. Giao diện quản lý gần gũi, đơn giản. Giao diện dùng cho người dùng thân thiện, sinh động, tương tác cao.
* Yêu cầu chất lượng:
* Tính tiến hóa:
* Dễ dàng trong lúc nâng cấp hoặc tăng các module tiện ích.
* Tính tiện dụng:
* Hệ thống có giao diện trực quan, thân thiện và dễ sử dụng.
* Các đặc tả và hướng dẫn sử dụng rõ ràng.
* Tính hiệu quả:
* Ứng dụng thể hiện tốt lịch chiếu của các rạp phim.
* Ứng dụng hoạt động ổn định và đáng tin cậy.
* Tính tương thích:
* Hoạt động tốt với tất cả các hệ điều hành phổ biến hiện nay .

1. LỰA CHỌN PHƯƠNG ÁN THIẾT KẾ
   1. CÁC YÊU CẦU CỦA HỆ THỐNG
      1. CÁC YÊU CẦU CĂN BẢN, BẮT BUỘC

|  |  |
| --- | --- |
| Y1 | Quản lý lưu trữ tất cả thông tin cần thiết cho hệ thống (phim, nhân viên, khuyến mãi, tin tức, …). |
| Y2 | Tìm kiếm, tra cứu các thông tin cần để xử lý một cách tức thời. |
| Y3 | Thanh toán vé, bắp nước, ... vv |
| Y4 | Thực hiện được các thống kê, báo biểu cần thiết. |
| Y5 | Các hóa đơn, biểu mẫu, báo cáo,… có thể in ra giấy. |

Bảng 2.4. Bảng các yêu cầu căn bản, bắt buộc

* + 1. CÁC YÊU CẦU CẦN THIẾT

|  |  |
| --- | --- |
| Y6 | Có thể tính toán, thống kê tại thời điểm yêu cầu. |
| Y7 | Tự động cập nhật số lượng vé còn lại và các vị trí trống của phòng chiếu tương ứng với phim được chọn. |

Bảng 2.5. Bảng các yêu cầu cần thiết

* + 1. CÁC YÊU CẦU MONG MUỐN

|  |  |
| --- | --- |
| Y8 | Dễ dàng trong lúc nâng cấp hoặc tăng các module tiện ích. |
| Y9 | App hoạt động ổn định và đáng tin cậy. |
| Y10 | Hoạt động tốt với tất cả hệ điều hành phổ biến hiện nay |
| Y11 | App nhẹ, thời gian load nhanh. |

Bảng 2.5. Bảng các yêu cầu mong muốn

* 1. CÁC RÀNG BUỘC CỦA HỆ THỐNG

|  |  |
| --- | --- |
| R1 | Tổng chi phí để thực hiện ứng dụng không được vượt quá 80 triệu. |
| R2 | Hệ thống cần hoạt động trơn tru sau 4 tháng triển khai. |
| R3 | Giao diện quản lý gần gũi, đơn giản, thân thiện, sinh động, tương tác cao. |
| R4 | Nhân viên quản lý và giám đốc phải được toàn quyền xử lý và giám sát hệ thống. |
| R5 | Xây dựng và phát triển trên CSDL đã có của phần mềm quản lý rạp phim. |

Bảng 2.6. Bảng các ràng buộc của hệ thống

* 1. CÁC PHƯƠNG ÁN ĐỀ XUẤT CHO HỆ THỐNG

|  |  |  |  |
| --- | --- | --- | --- |
| Tiêu chuẩn | Phương án A | Phương án B | Phương án C |
| Yêu cầu  Y1 → Y5 | Có (trừ Y5 không đáp ứng) | Có đầy đủ | Có đầy đủ |
| Y6 | Có | Có | Có |
| Y7 | Không | Có | Có |
| Y8 | Có | Có | Có |
| Y9 | Không | Khó có thể | Có |
| Y10 | Không | Có | Có |
| Y11 | Không | Có | Có |
| Ràng buộc R1 | 45 | 65 | 90 |
| R2 | 3 tháng | 4 tháng | 6 tháng |
| R3 | Đáp ứng tương đối | Đáp ứng được | Đáp ứng được |
| R4 | Đáp ứng được | Đáp ứng được | Đáp ứng được |
| R5 | Đáp ứng | Đáp ứng được | Không đáp ứng được |

Bảng 2.7. Bảng các phương án đề xuất cho hệ thống

* 1. CHỌN LỰA PHƯƠNG ÁN THIẾT KẾ

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Điều kiện | Trọng số | Phương án A | | Phương án B | | Phương án C | |
|  |  | Tỉ lệ | Điểm | Tỉ lệ | Điểm | Tỉ lệ | Điểm |
| Yêu cầu: | | | | | | | |
| Y1 → Y5 | *25* | 4 | 100 | 5 | 125 | 5 | 125 |

A close up of a piece of paper

Description automatically generated

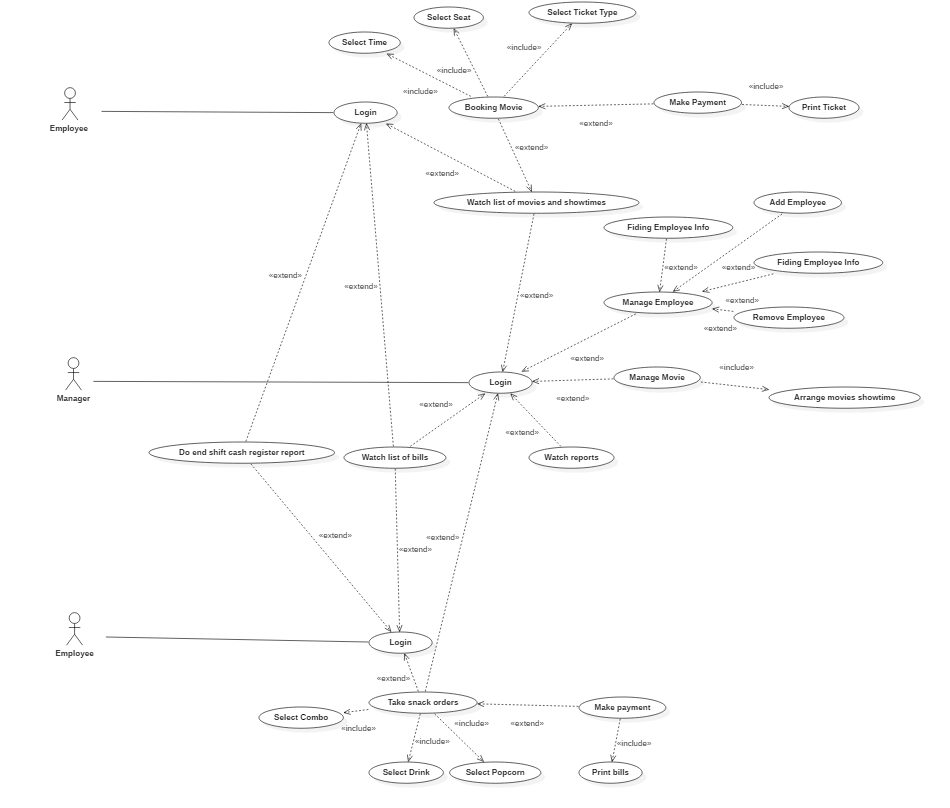
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Ràng buộc | | | | | | | |
| R1 | *20* | 5 | 100 | 4 | 80 | 2 | 40 |
| R2 | *10* | 5 | 50 | 5 | 50 | 3 | 30 |
| R3 | *7* | 3 | 21 | 4 | 28 | 4 | 28 |
| R4 | *4* | 3 | 12 | 4 | 16 | 4 | 16 |
| R5 | *9* | 5 | 45 | 5 | 45 | 2 | 18 |
|  | ***50*** |  | ***228*** |  | ***219*** |  | ***132*** |
| Tổng | 100 |  | 389 |  | 441 |  | 379 |

Bảng 2.8. Bảng chọn lựa phương án thiết kế

Dựa vào bảng điểm trên phương án thiết kế được chọn là phương án B.

III. THIẾT KẾ HỆ THỐNG

1. SƠ ĐỒ USECASE



1. DANH SÁCH CÁC TÁC NHÂN CỦA HỆ THỐNG

|  |  |  |
| --- | --- | --- |
| STT | Tác nhân của hệ thống | Ý nghĩa |
| 1 | Người quản trị | Người quản trị hệ thống |
| 2 | Người dùng | Nhân viên của rạp phim |

Bảng 3.1. Bảng danh sách các tác nhân của hệ thống

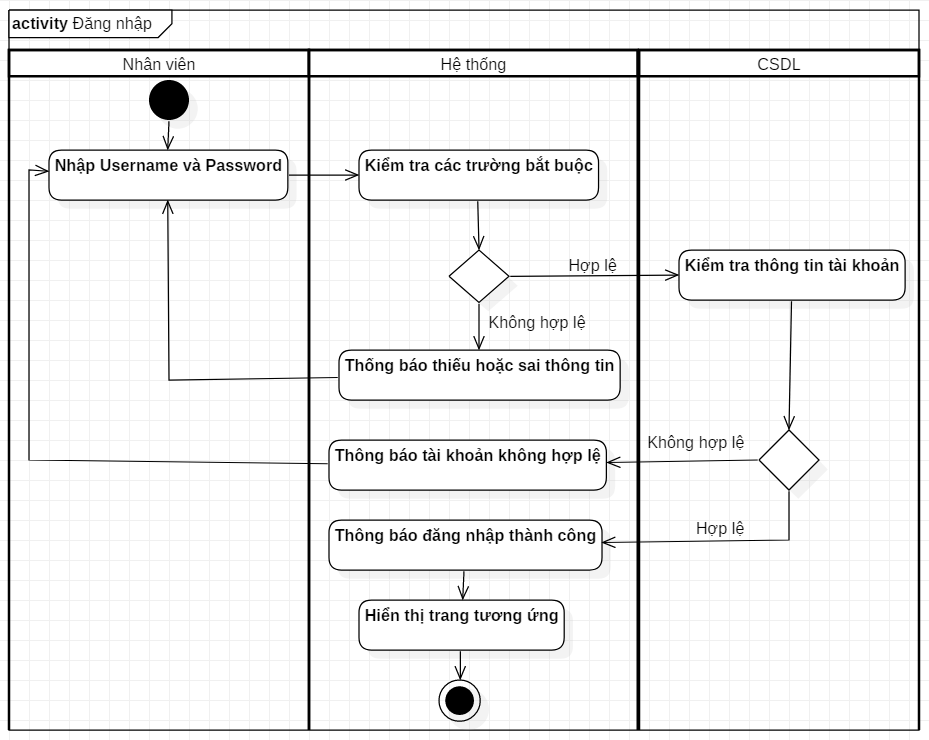
1. DANH SÁCH USECASE

|  |  |  |
| --- | --- | --- |
| STT | Usecase | Ý nghĩa |
| 1 | Đăng nhập | Người dùng đăng nhập vào hệ thống |
| 2 | Quản lý kho hàng | Người dùng quản lý kho và các sản phẩm trong kho. |
| 3 | Quản lý đặt mua vé | Nhân viên bán vé theo nhu cầu của khách hàng |
| 4 | Quản lý đặt mua bắp, nước và các sản phẩm khác | Nhân viên bán bắp, nước và các sản phẩm khác (ly đựng nước, lịch năm, … vv) theo nhu cầu của khách hàng |
| 5 | Xem lịch sử giao dịch | Người quản trị, nhân viên xem lịch sử các giao dịch gần nhất |
| 6 | Quản lý phim | Thêm, sửa, xóa và cập nhật thông tin phim |
| 7 | Quản lý lịch chiếu | Thêm, sửa, xóa và cập nhật thông tin lịch chiếu |
| 8 | Quản lý phòng chiếu | Thêm, sửa, xóa và cập nhật thông tin phòng chiếu bao gồm các trang thiết bị trong phòng chiếu |
| 9 | Quản lý nhân viên | * Thêm, sửa, xóa, cập nhật và tra cứu thông tin nhân viên của rạp * Phân quyền truy cập cho nhân viên |
| 10 | Báo cáo doanh thu | Lập báo cáo doang thu theo khoảng thời gian được yêu cầu |

Bảng 3.2. Bảng danh sách Usecase

1. ĐẶC TẢ USECASE VÀ SƠ ĐỒ HOẠT ĐỘNG (ACTIVITY DIAGRAM)
2. 1. USECASE ĐĂNG NHẬP

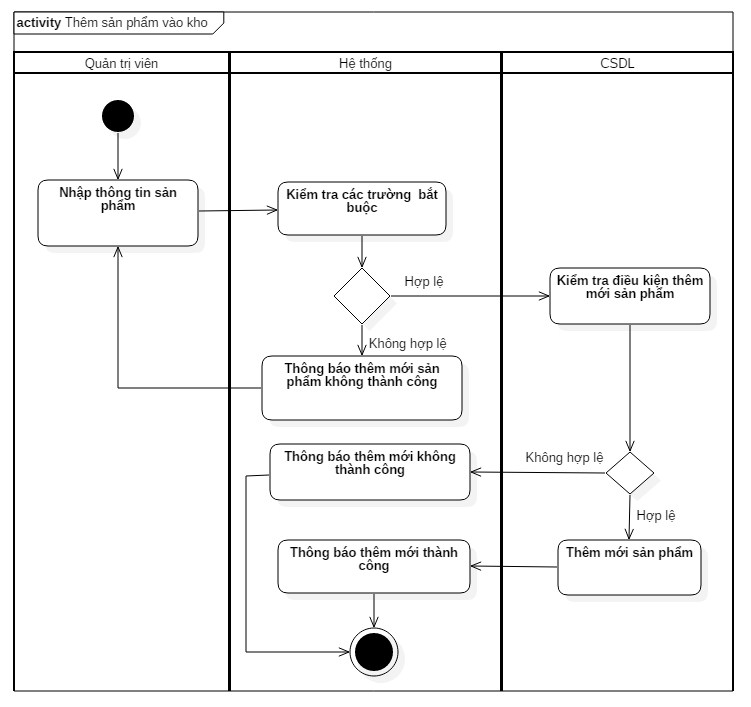
* Tên usecase: Đăng nhập
* Mô tả: Quản trị viên, nhân viên đăng nhập vào hệ thống.
* Luồng sự kiện:



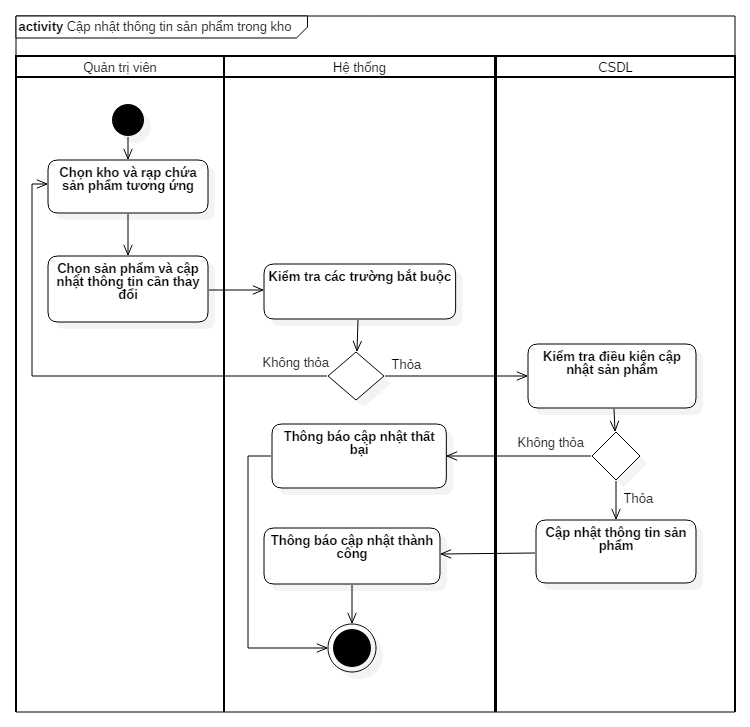
* Tiền điều kiện: Không
* Hậu điều kiện:
* Hợp lệ: Hiển thị thông báo đăng nhập thành công
* Không hợp lệ : Hiển thị thông báo tài khoản không hợp lệ
* Mối quan hệ: Nhân viên, quản trị viên và hệ thống quản lý rạp phim
  1. USECASE QUẢN LÝ KHO HÀNG

2. 2. 1. USECASE THÊM SẢN PHẨM VÀO KHO

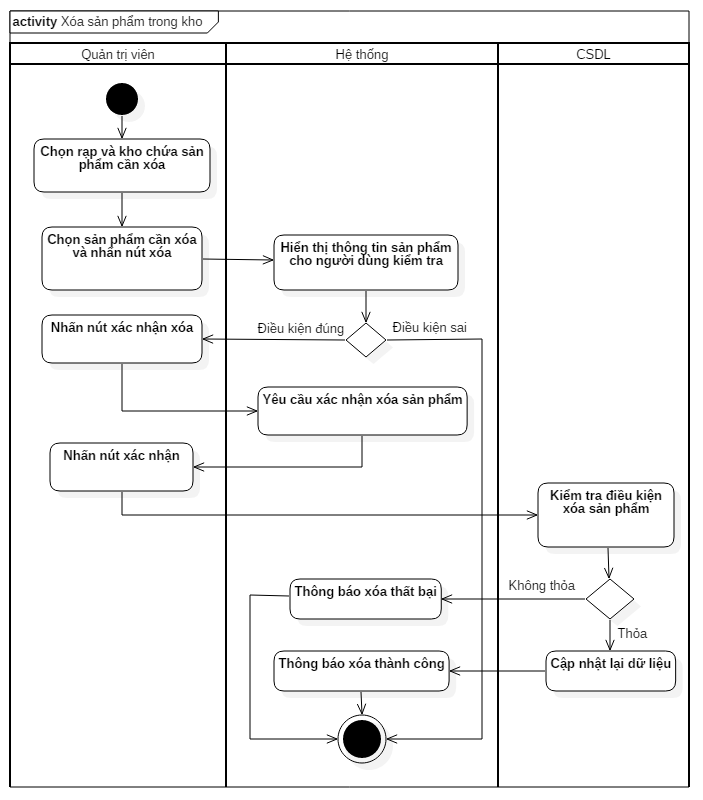
* Tên usacase : Thêm sản phẩm vào kho
* Mô tả : Quản trị viên chọn rạp, kho và thêm sản phẩm vào rạp, kho tương ứng.
* Luồng sự kiện:



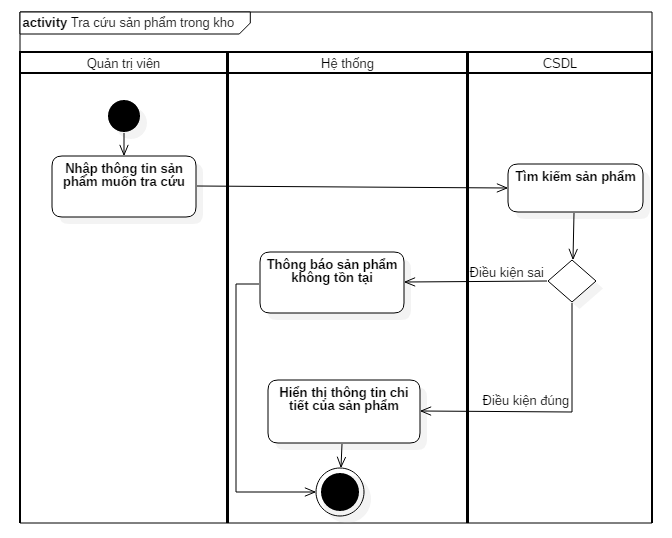
* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống.
* Hậu điều kiện:
* Nếu thỏa : Cập nhật thông tin xuống CSDL và hiển thị thông báo đã thêm sản phẩm vào kho thành công.
* Nếu không thỏa : Hiển thị thông báo thêm sản phẩm vào kho không thành công.
* Mối quan hệ: Quản trị viên với hệ thống quản lý kho
  + 1. USECASE CẬP NHẬT THÔNG TIN SẢN PHẨM TRONG KHO
* Tên usecase : Cập nhật thông tin sản phẩm trong kho
* Mô tả : Quản trị viên cập nhật thông tin sản phẩm trong kho của rạp được chọn.
* Luồng sự kiện:



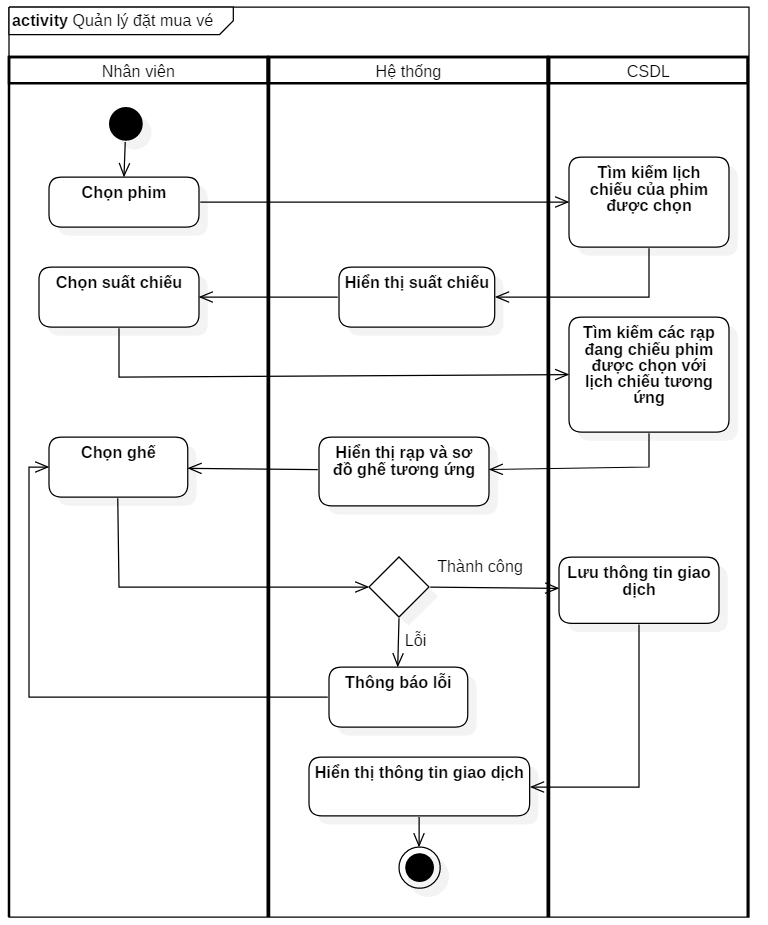
* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Cập nhật thông tin xuống CSDL và hiển thị thông báo cập nhật thành công ra màn hình.
* Nếu không thỏa : Thông báo cập nhật thất bại.
* Mối quan hệ: Quản trị viên với hệ thống quản lý kho
  + 1. USECASE XÓA SẢN PHẨM TRONG KHO
* Tên usecase: Xóa sản phẩm trong kho
* Mô tả : Quản trị viên chọn rạp, kho chứa sản phẩm tương ứng được chọn và xóa sản phẩm đó
* Luồng sự kiện:



* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : hiển thị thông báo ra màn hình và cập nhật dữ liệu xuống CSDL
* Nếu không thỏa : hiển thị thông báo xóa sản phẩm thất bại
  + 1. USECASE TRA CỨU SẢN PHẨM TRONG KHO
* Tên usecase: Tra cứu sản phẩm trong kho
* Mô tả : Quản trị viên chọn rạp, kho chứa sản phẩm tương ứng được chọn và xem thông tin sản phẩm đó
* Luồng sự kiện:



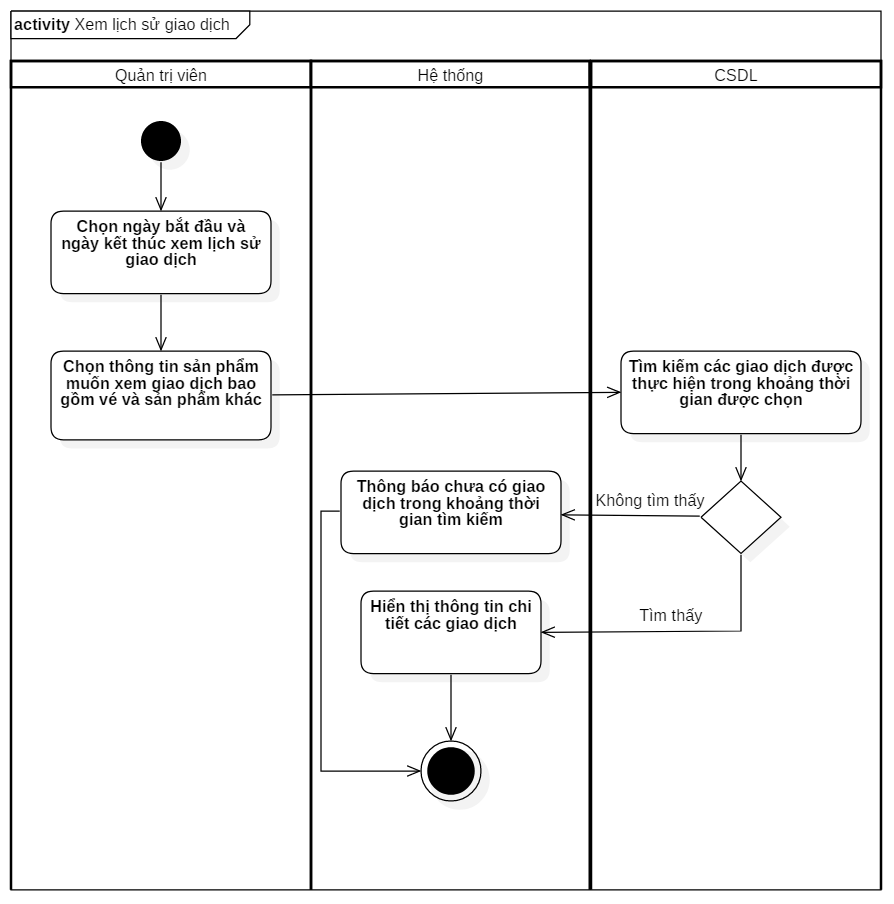
* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống.
* Hậu điều kiện:
* Nếu thỏa : Hiển thị thông tin chi tiết sản phẩm
* Nếu không thỏa : Hiển thị thông báo sản phẩm không tồn tại
* Mối quan hệ : Quản trị viên và hệ thống quản lý kho
  1. USECASE QUẢN LÝ ĐẶT MUA VÉ
* Tên usecase: Quản lý đặt mua vé
* Mô tả: Nhân viên đặt mua vé cho khách
* Luồng sự kiện:



* Tiền điều kiện: Nhân viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện: Hiển thị và lưu thông tin giao dịch vào CSDL
* Mối quan hệ: Nhân viên và hệ thống quản lí lịch chiếu
  1. USECASE QUẢN LÝ ĐẶT MUA BẮP NƯỚC VÀ CÁC SẢN PHẨM KHÁC
* Tên usecase: Quản lý đặt mua bắp nước và các sản phẩm khác
* Mô tả: Nhân viên đặt mua bắp nước và các sản phẩm khác cho khách
* Luồng sự kiện:



* Tiền điều kiện: Nhân viên đã có tài khoản và đăng nhập thành công vào tài khoản.
* Hậu điều kiện: Hiển thị và lưu thông tin giao dịch xuống CSDL
* Mối quan hệ: Nhân viên và hệ thống quản lí kho, quản lý rạp
  1. USECASE XEM LỊCH SỬ GIAO DỊCH
* Tên usecase: Xem lịch sử giao dịch
* Mô tả: Quản trị viên xem lịch sử giao dịch trong khoảng thời gian được chọn
* Luồng sự kiện:



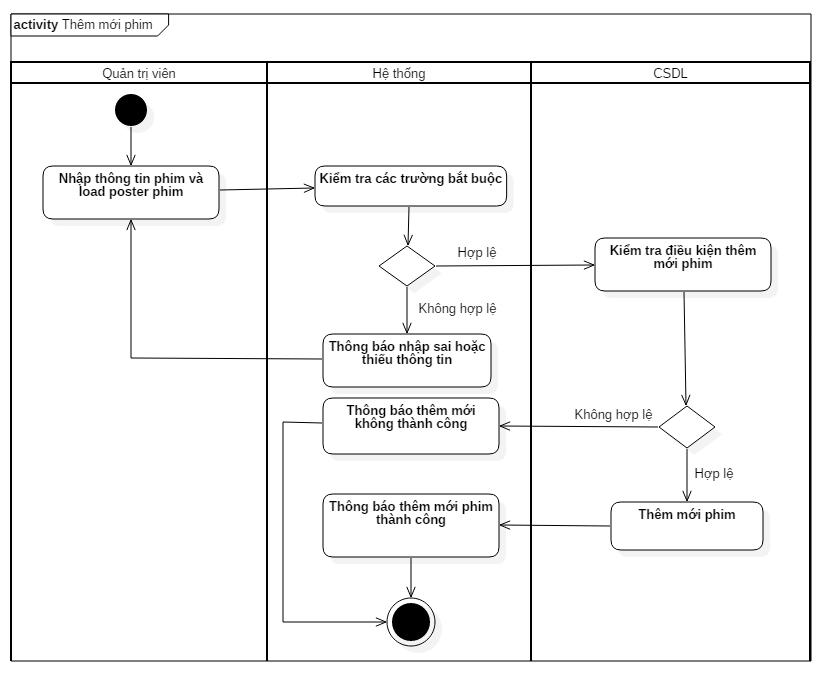
* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : hiển thị thông tin các giao dịch trong khoảng thời gian được chọn
* Nếu không thỏa : thông báo chưa có giao dịch trong khoảng thời gian được chọn
* Mối quan hệ: Quản trị viên và hệ thống quản lý nghiệp vụ xem lịch sử giao dịch
  1. USECASE QUẢN LÝ PHIM


  5. 1. USECASE THÊM MỚI PHIM

- Tên usecase: Thêm mới phim

- Mô tả: Quản trị viên thực hiện thêm mới phim.

- Luồng sự kiện:



- Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống.

- Hậu điều kiện:

* Nếu thỏa: Thông báo thêm phim thành công và cập nhật dữ liệu xuống CSDL.
* Nếu không thỏa: Thông báo thêm phim không thành công.

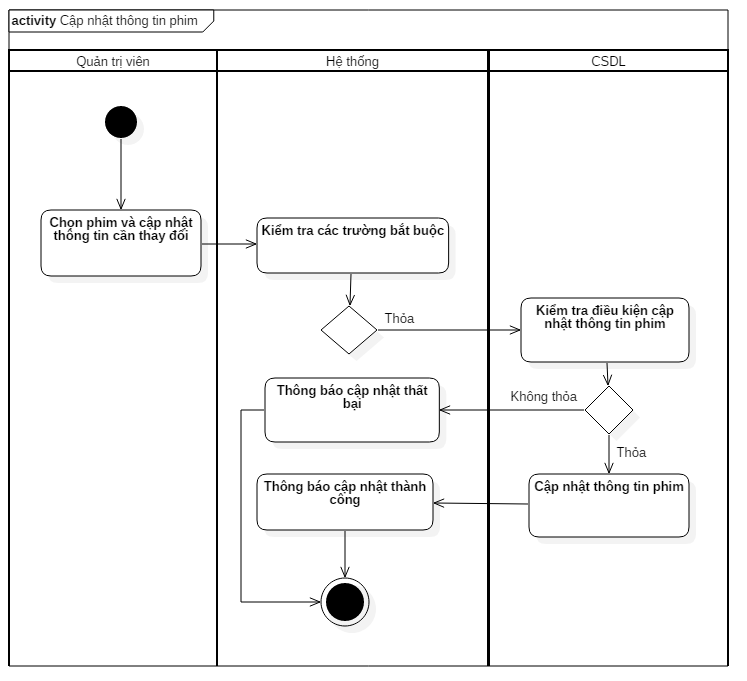
- Mối quan hệ: quản trị viên và hệ thống quản lý phim.

* + 1. USECASE CẬP NHẬT THÔNG TIN PHIM

- Tên usecase: Cập nhật thông tin phim

- Mô tả: Quản trị viên cập nhật thông tin phim

- Luồng sự kiện:



- Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống.

- Hậu điều kiện:

* Nếu thỏa: Thông báo cập nhật phim thành công và cập nhật thông tin xuống CSDL.
* Nếu không thỏa: Thông báo cập nhật phim không thành công.

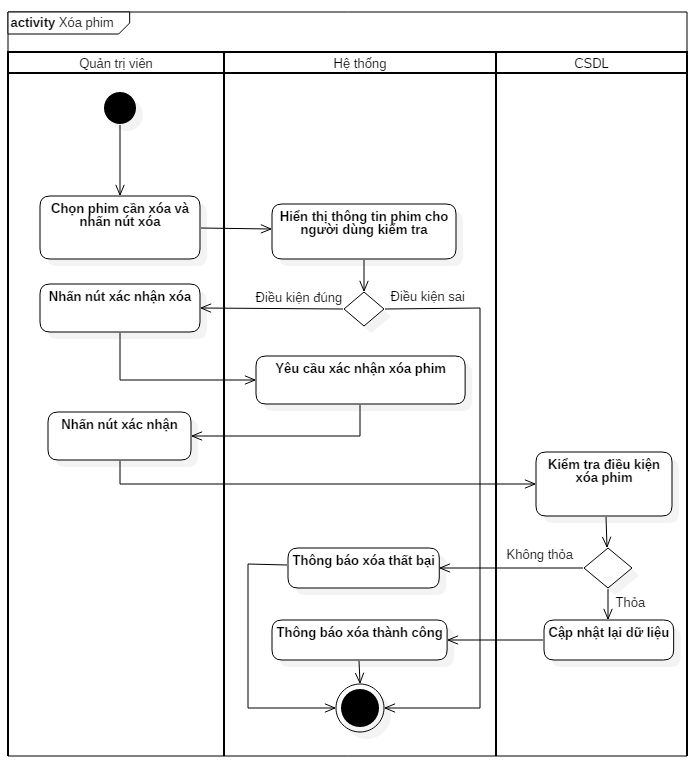
- Mối quan hệ: Quản trị viên và hệ thống quản lý phim.

* + 1. USECASE XÓA PHIM

- Tên usecase: Xóa phim

- Mô tả: Quản trị viên xóa phim.

- Luồng sự kiện:



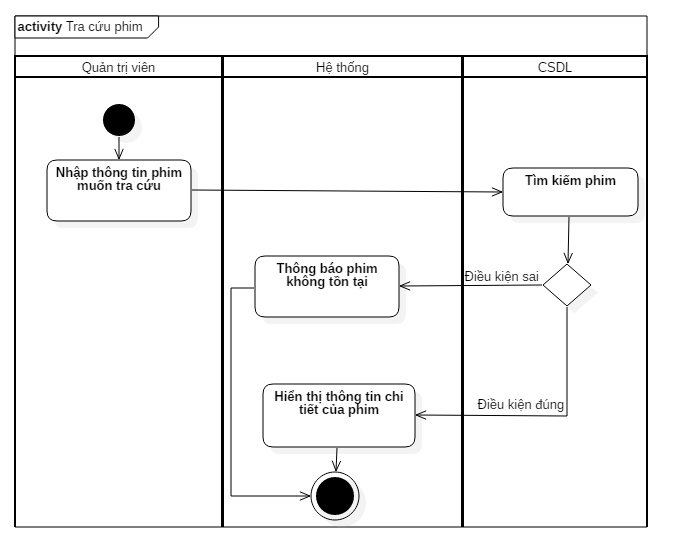
- Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống.

- Hậu điều kiện:

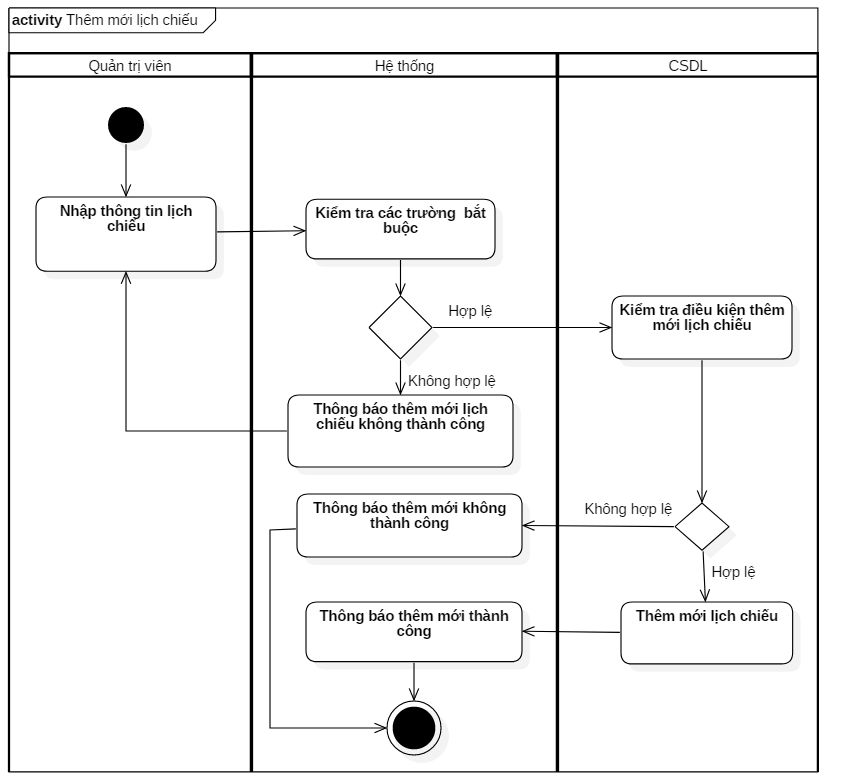
* Nếu thỏa : Thông báo xóa phim thành công và cập nhật lại CSDL.
* Nếu không thỏa : Thông báo xóa phim không thành công.

- Mối quan hệ: quản trị viên và hệ thống quản lý phim

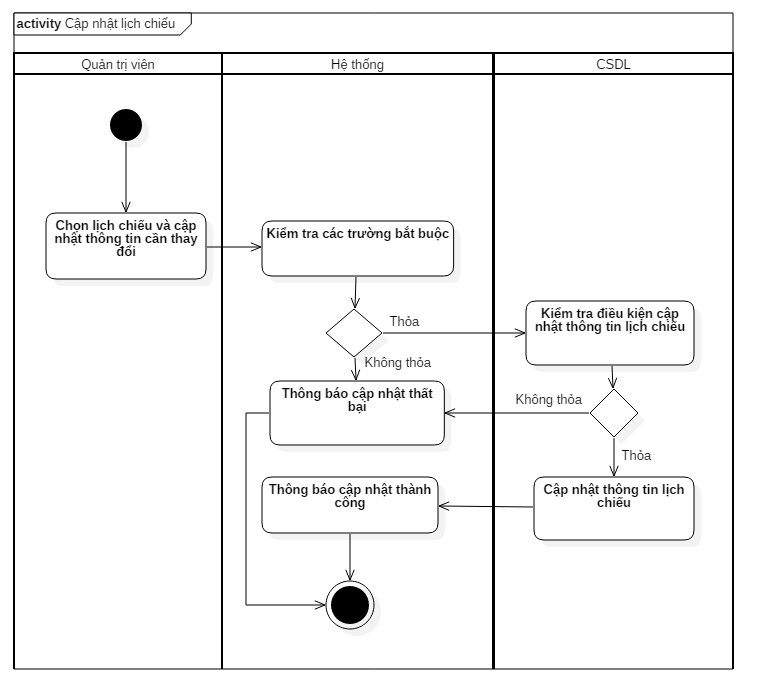
* + 1. USECASE TRA CỨU PHIM
* Tên usecase: Sửa thông tin người dùng
* Mô tả: Người dùng vào chức năng chỉnh sửa thông tin cá nhân và tiến hành chỉnh sửa những thông tin cần thiết.
* Luồng sự kiện:



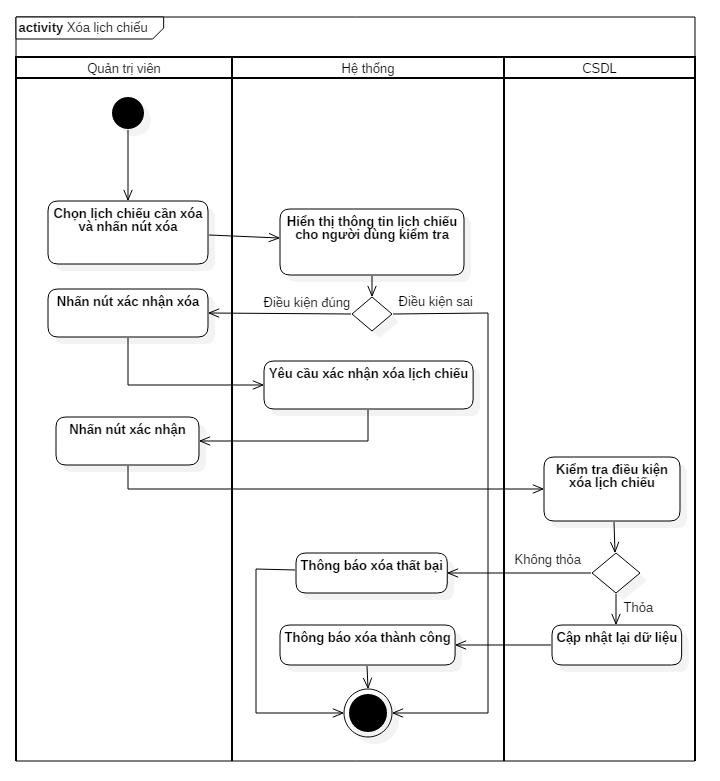
* Tiền điều kiện: Người dùng đã có tài khoản đăng nhập
* Hậu điều kiện: Hệ thống chuyển sang trang thông tin người dùng
* Mối quan hệ: Người dùng và hệ thống quản lý nhân viên
  1. USECASE QUẢN LÝ LỊCH CHIẾU
  2. 1. USECASE THÊM MỚI LỊCH CHIẾU
* Tên usecase : Thêm mới lịch chiếu
* Mô tả : Quản trị viên thực hiện thêm mới lịch chiếu chop him
* Luồng sự kiện :



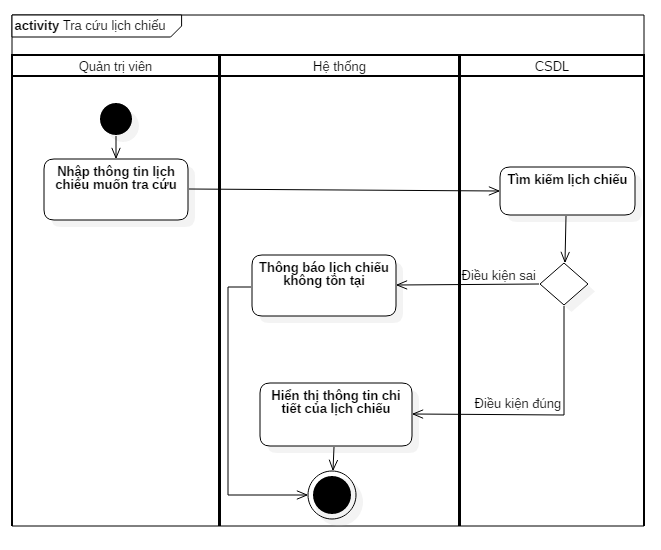
* Tiền điều kiện : Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện :
* Nếu thỏa: Thông báo thêm mới thành công và cập nhật thông tin xuống CSDL.
* Nếu không thỏa: Thông báo thêm mới không thành công.
* Mối quan hệ : Quản trị viên và hệ thống quản lý lịch chiếu
  + 1. USECASE CẬP NHẬT LỊCH CHIẾU
* Tên usecase : Cập nhật lịch chiếu
* Mô tả : Quản trị viên cập nhật lịch chiếu
* Luồng sự kiện :



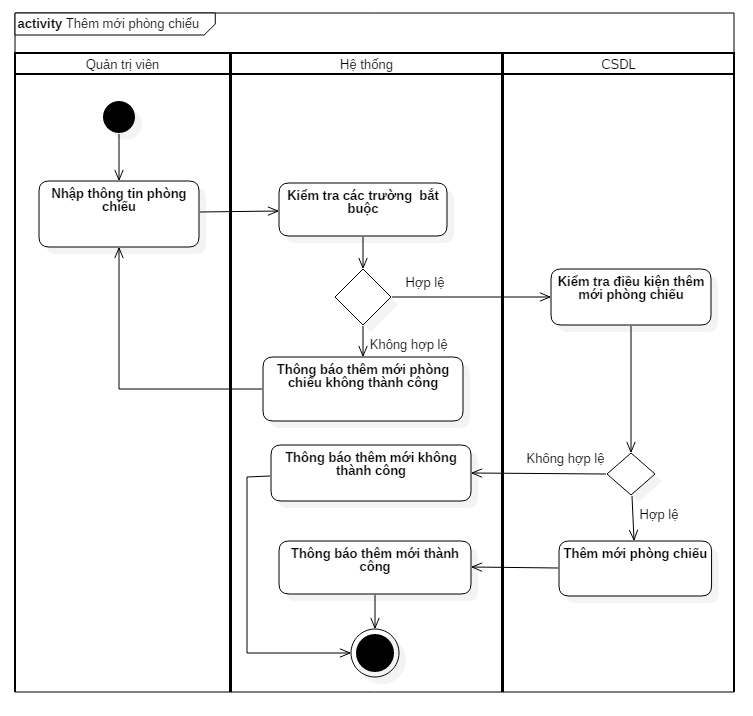
* Tiền điều kiện : Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện :
* Nếu thỏa: Thông báo cập nhật thành công và cập nhật thông tin xuống CSDL.
* Nếu không thỏa : Thông báo cập nhật thất bại
* Mối quan hệ : Quản trị viên và hệ thống quản lí lịch chiếu
  + 1. USECASE XÓA LỊCH CHIẾU
* Tên usecase: Xóa lịch chiếu
* Mô tả: Quản trị viên xóa lịch chiếu
* Luồng sự kiện:



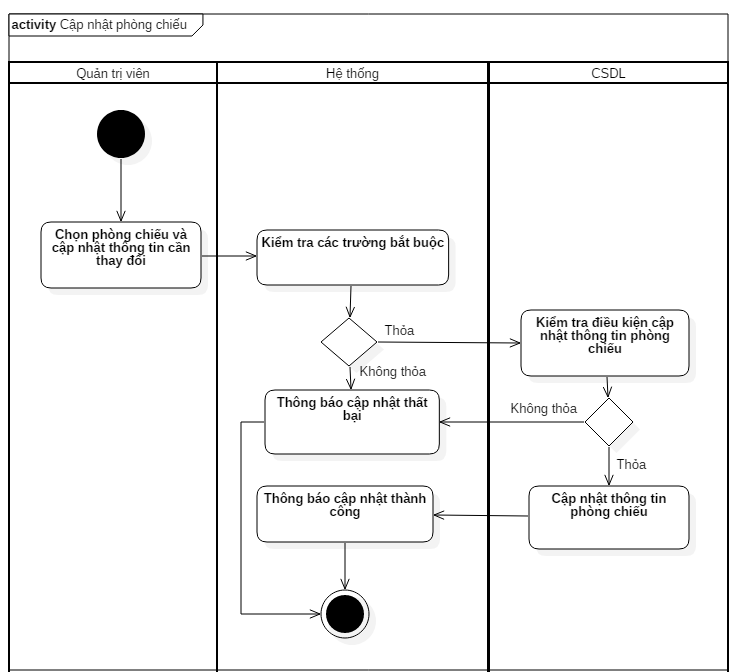
* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Thông báo xóa thành công và cập nhật thông tin xuống CSDL
* Nếu không thỏa : Thông báo xóa thất bại.
* Mối quan hệ: Người quản trị và hệ thống quản lý lịch chiếu
  + 1. USECASE TRA CỨU LỊCH CHIẾU
* Tên usecase: Tra cứu lịch chiếu
* Mô tả: Quản trị viên tra cứu để xem thông tin lịch chiếu
* Luồng sự kiện:



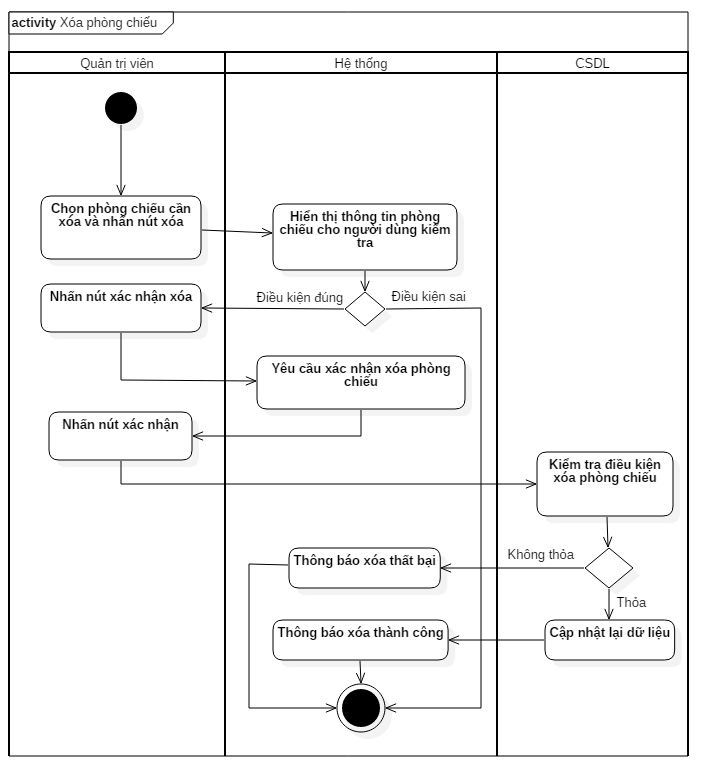
* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Hiển thị thông tin chi tiết lịch chiếu.
* Nếu không thỏa : Thông báo lịch chiếu không tồn tại.
* Mối quan hệ: Người quản trị và hệ thống quản lý lịch chiếu
  1. USECASE QUẢN LÝ PHÒNG CHIẾU
  2. 1. USECASE THÊM MỚI PHÒNG CHIẾU
* Tên usecase: Thêm mới phòng chiếu
* Mô tả: Quản trị viên thêm mới phòng chiếu
* Luồng sự kiện:



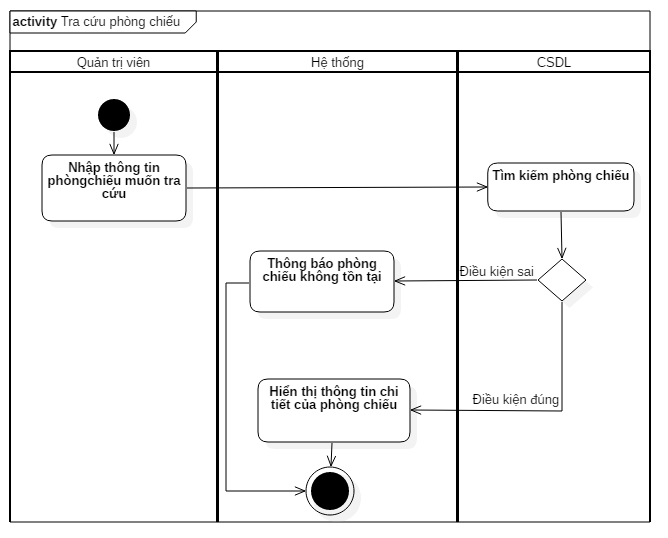
* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Thông báo thêm mới thành công và cập nhật thông tin xuống CSDL
* Nếu không thỏa : Thông báo thêm mới thất bại
* Mối quan hệ: Người quản trị và hệ thống quản lý phòng chiếu
  + 1. USECASE CẬP NHẬT PHÒNG CHIẾU
* Tên usecase: Cập nhật phòng chiếu
* Mô tả: Quản trị viên cập nhật thông tin phòng chiếu
* Luồng sự kiện:



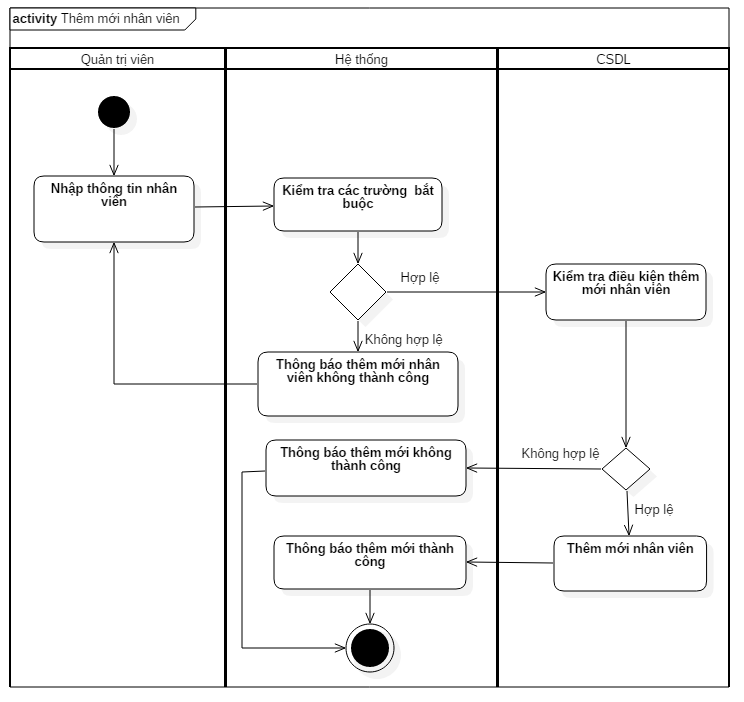
* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Thông báo cập nhật thành công và cập nhật thông tin xuống CSDL
* Nếu không thỏa : Thông báo cập nhật thất bại
* Mối quan hệ: Người quản trị và hệ thống quản lý phòng chiếu
  + 1. USECASE XÓA PHÒNG CHIẾU
* Tên usecase: Xóa phòng chiếu
* Mô tả: Quản trị viên xóa phòng chiếu
* Luồng sự kiện:



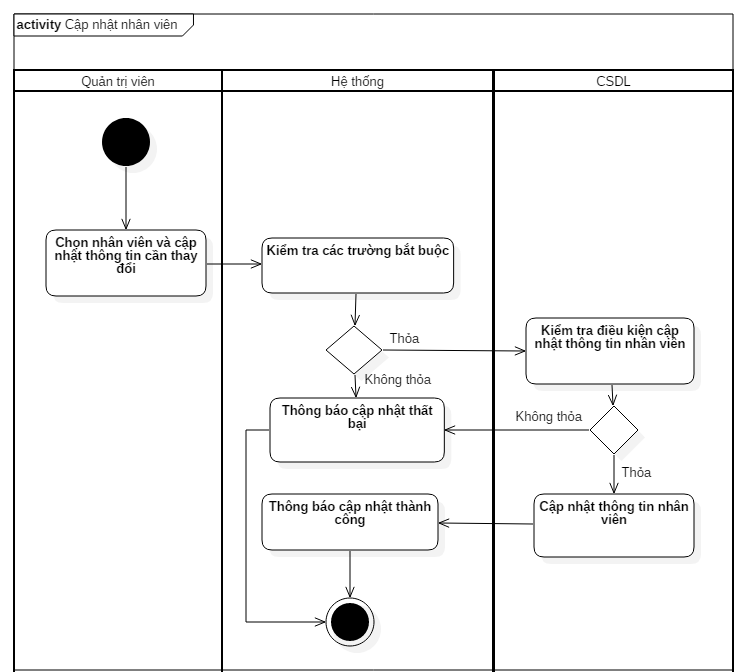
* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Thông báo xóa phòng chiếu thành công và cập nhật thông tin xuống CSDL
* Nếu không thỏa : Thông báo xóa không thành công
* Mối quan hệ: Người quản trị và hệ thống quản lý phòng chiếu
  + 1. USECASE TRA CỨU PHÒNG CHIẾU
* Tên usecase: Tra cứu phòng chiếu
* Mô tả: Quản trị viên thực hiện tra cứu thông tin phòng chiếu
* Luồng sự kiện:



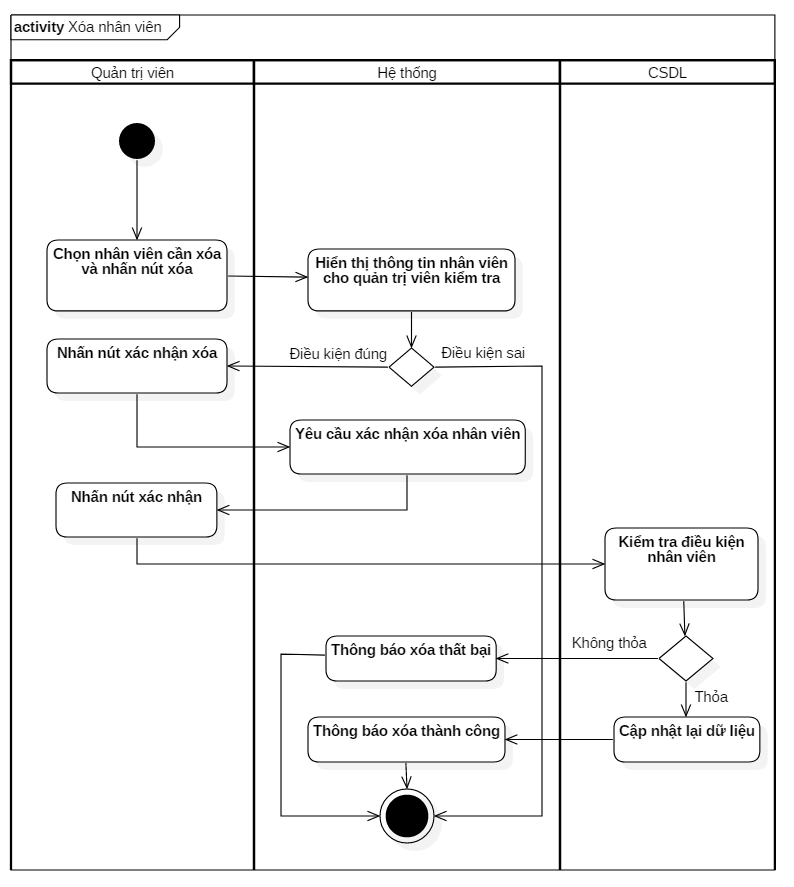
* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : Hiển thị thông tin chi tiết phòng chiếu
* Nếu không thỏa : Thông báo phòng chiếu không tồn tại.
* Mối quan hệ: Người quản trị và hệ thống quản lý phòng chiếu.
  1. USECASE QUẢN LÝ NHÂN VIÊN
     1. USECASE THÊM MỚI NHÂN VIÊN
* Tên usecase: Thêm mới nhân viên
* Mô tả : Quản trị viên thêm mới nhân viên
* Luồng sự kiện:



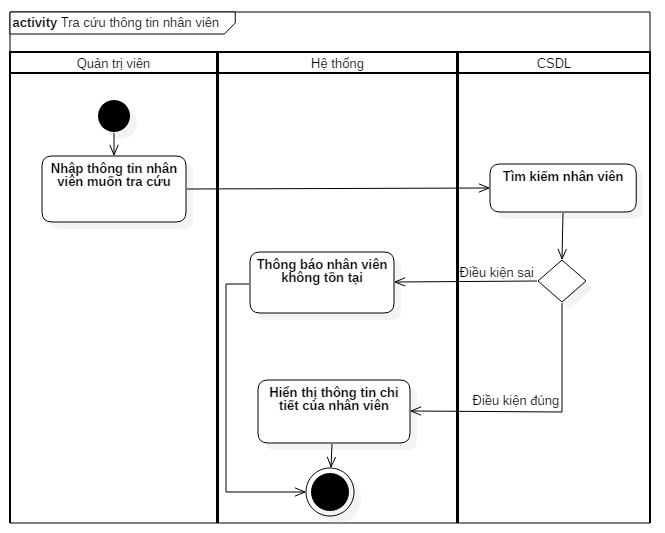
* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa: Thông tin được cập nhật xuống cơ sở dữ liệu và hiển thị thông báo cập nhật thông báo thêm mới nhân viên thành công.
* Nếu không thỏa: Hiển thị thông báo thêm mới nhân viên thất bại.
* Mối quan hệ: Quản trị viên và hệ thống quản lý nhân viên
  + 1. USECASE CẬP NHẬT THÔNG TIN NHÂN VIÊN
* Tên usecase: Cập nhật thông tin nhân viên
* Mô tả: Quản trị viên cập nhật thông tin cho nhân viên
* Luồng sự kiện:



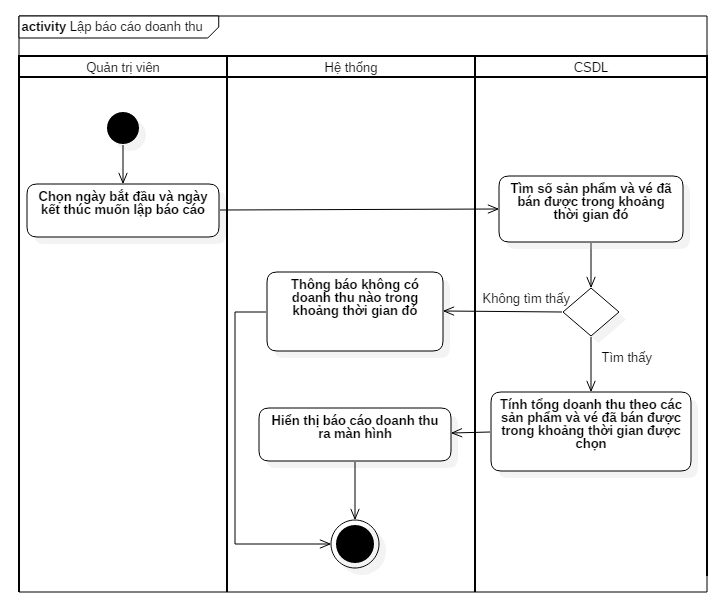
* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa: Thông tin được cập nhật xuống cơ sở dữ liệu và thông báo đã cập nhật thành công ra màn hình.
* Nếu không thỏa: Hiển thị thông báo cập nhật không thành công ra màn hình.
* Mối quan hệ: Quản trị viên và hệ thống quản lý nhân viên
  + 1. USECASE XÓA NHÂN VIÊN
* Tên usecase: Xóa nhân viên
* Mô tả: Quản trị viên xóa nhân viên
* Luồng sự kiện:



* Tiền điều kiện: Quản trị viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu thỏa : thông báo xóa thành công, thông tin được cập nhật xuống CSDL.
* Nếu không thỏa: thông báo xóa không thành công.
* Môi quan hệ: Quản trị viên và hệ thống quản lý nhân viên
  + 1. USECASE TRA CỨU NHÂN VIÊN
* Tên usecase: Tra cứu nhân viên
* Mô tả: Quản trị viên tra cứu thông tin nhân viên
* Luồng sự kiện:

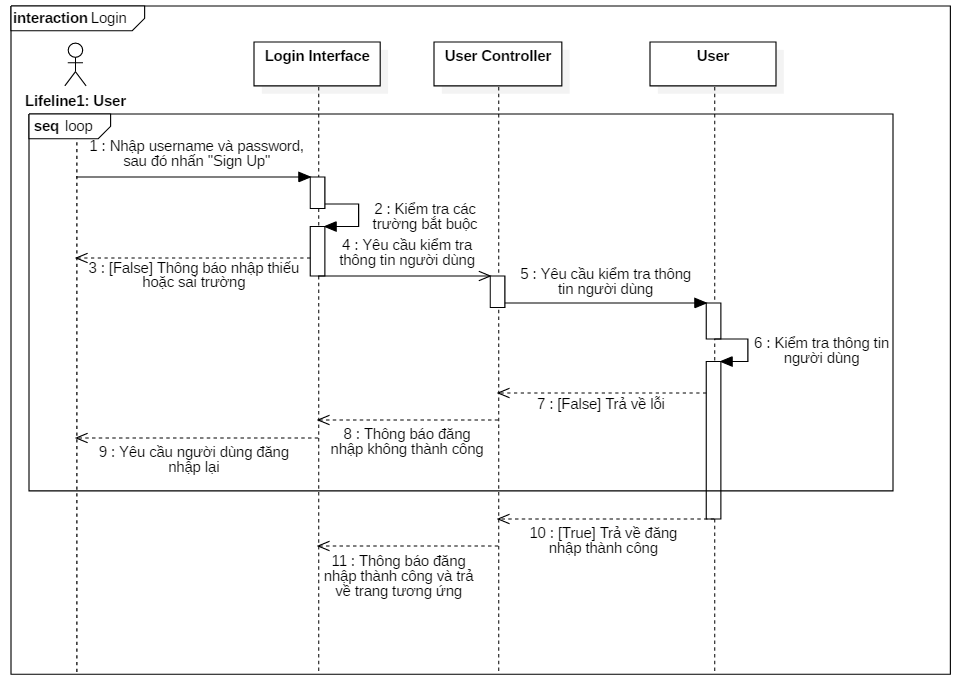


* Tiền điều kiện: Người quản trị, nhân viên đã đăng nhập thành công vào hệ thống
* Hậu điều kiện:
* Nếu tìm thấy: Hiển thị thông tin chi tiết của nhân viên
* Nếu không tìm thấy: Hiển thị thông báo nhân viên không tồn tại
* Mối quan hệ: Người quản trị, nhân viên và hệ thống quản lý nhân viên
  1. USECASE LẬP BÁO CÁO DOANH THU
* Tên usecase: Lập báo cáo doanh thu
* Mô tả: Quản trị viên lập báo báo doanh thu theo khoảng thời gian được nhập
* Luồng sự kiện:

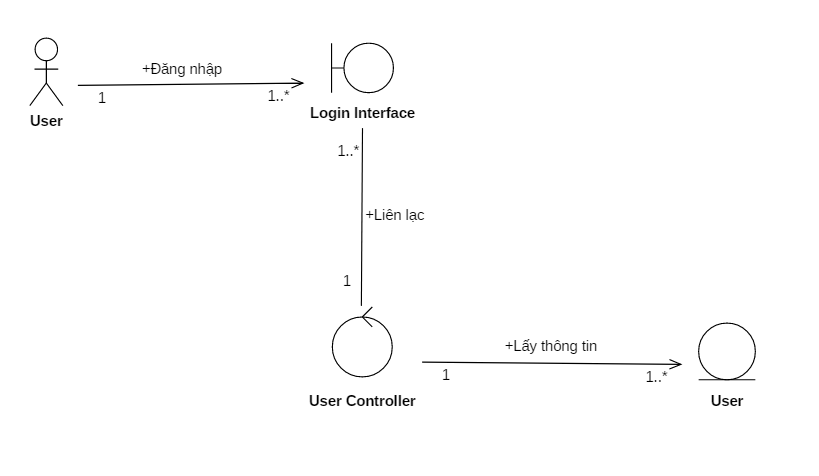


* Tiền điều kiện: Người quản trị đăng nhập thành công vào hệ thống
* Hậu điều kiện: Hiển thị báo cáo doanh thu ra màn hình

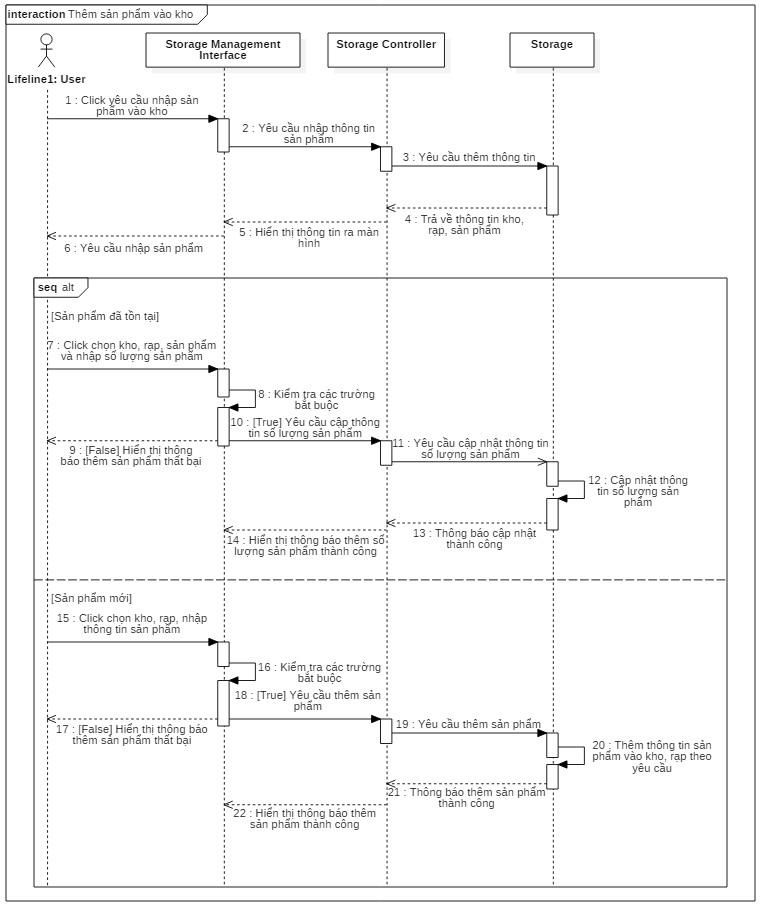
1. SƠ ĐỒ TRÌNH TỰ (SEQUENCE DIAGRAM) VÀ SƠ ĐỒ LỚP (CLASS DIAGRAM)
2. 1. ĐĂNG NHẬP
3. 1. 1. SEQUENCE DIAGRAM



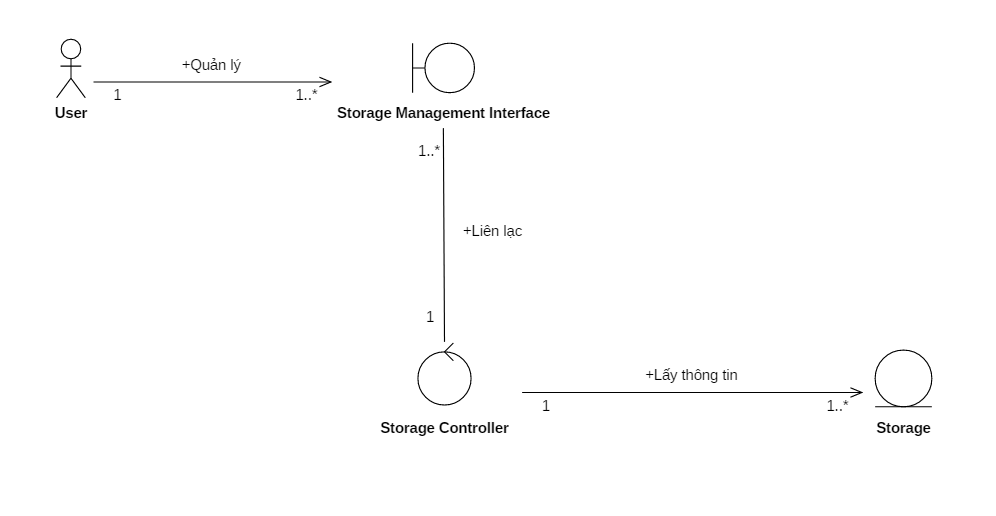
* + 1. CLASS DIAGRAM



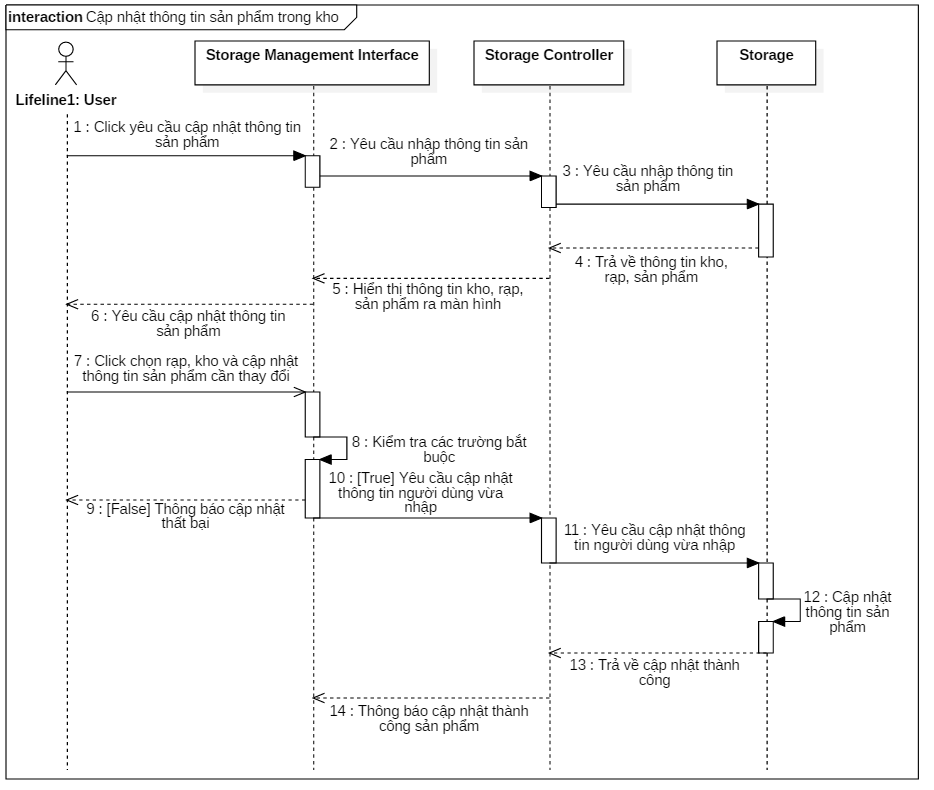
* 1. QUẢN LÝ KHO HÀNG
  2. 1. THÊM SẢN PHẨM VÀO KHO
        1. SEQUENCE DIAGRAM



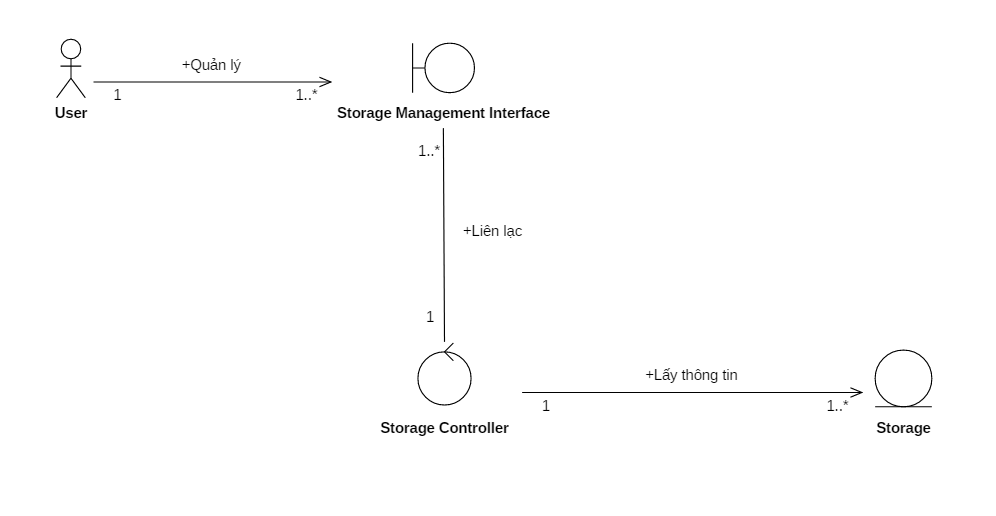
* + - 1. CLASS DIAGRAM



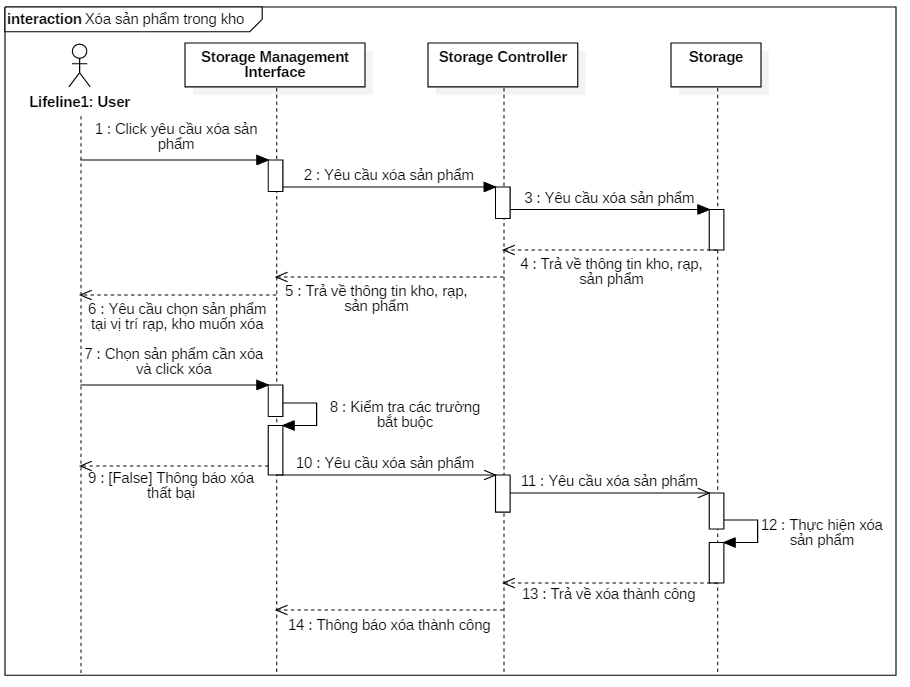
* + 1. CẬP NHẬT THÔNG TIN SẢN PHẨM TRONG KHO
       1. SEQUENCE DIAGRAM



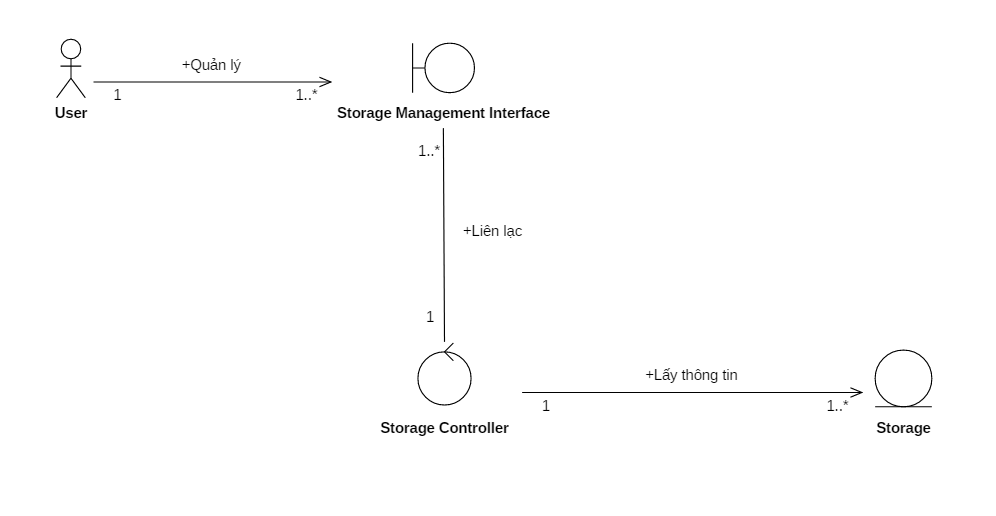
* + - 1. CLASS DIAGRAM



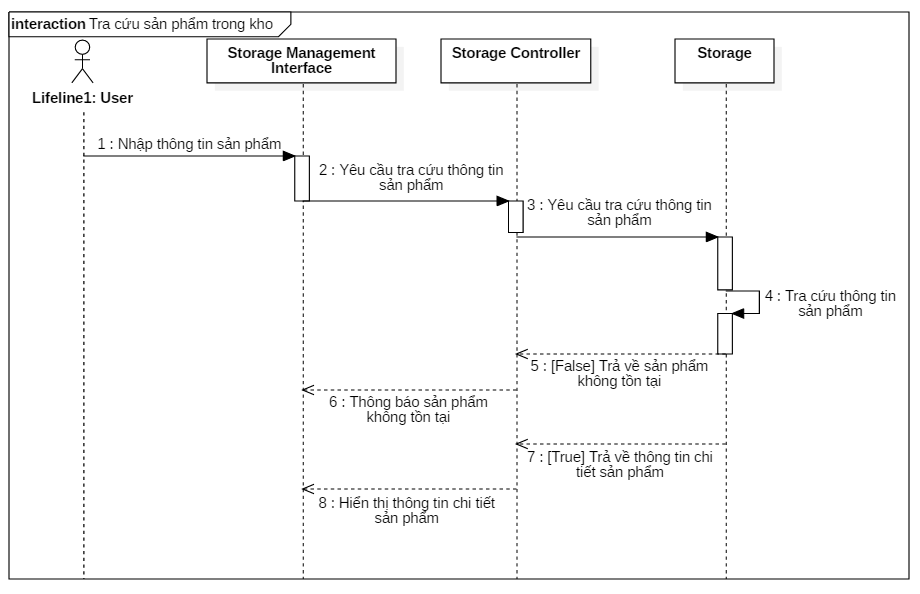
* + 1. XÓA SẢN PHẨM TRONG KHO
       1. SEQUENCE DIAGRAM



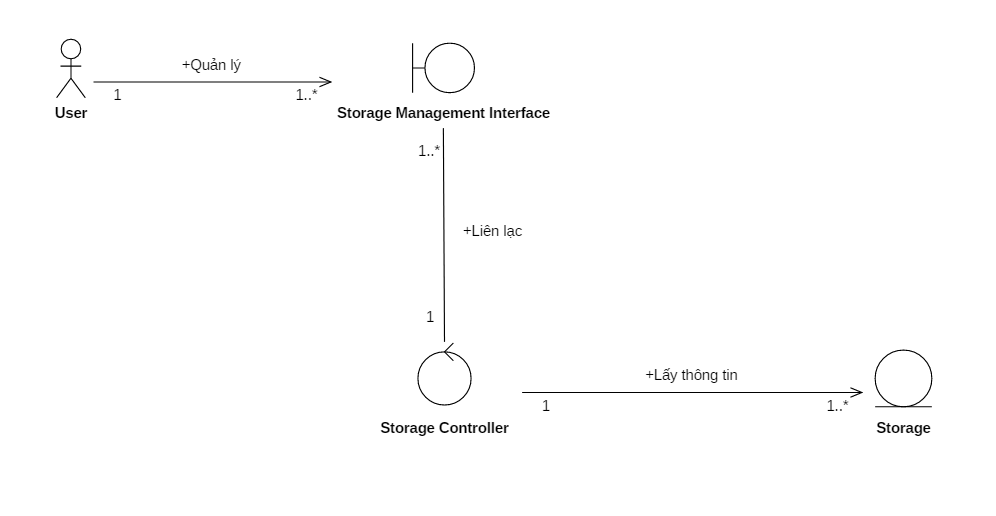
* + - 1. CLASS DIAGRAM



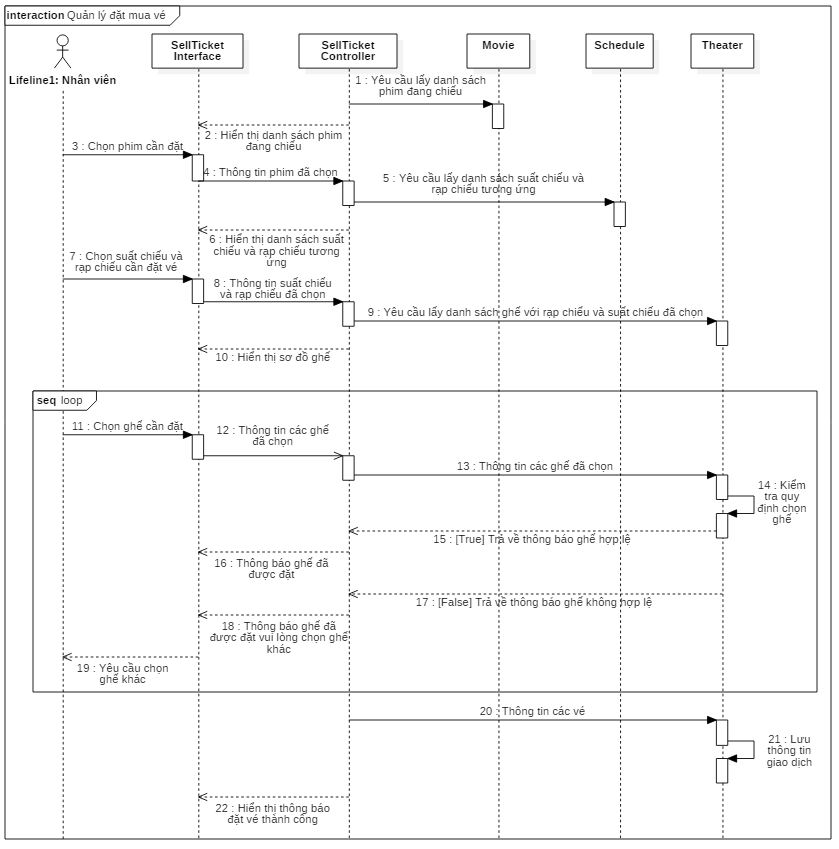
* + 1. TRA CỨU SẢN PHẨM TRONG KHO
       1. SEQUENCE DIAGRAM



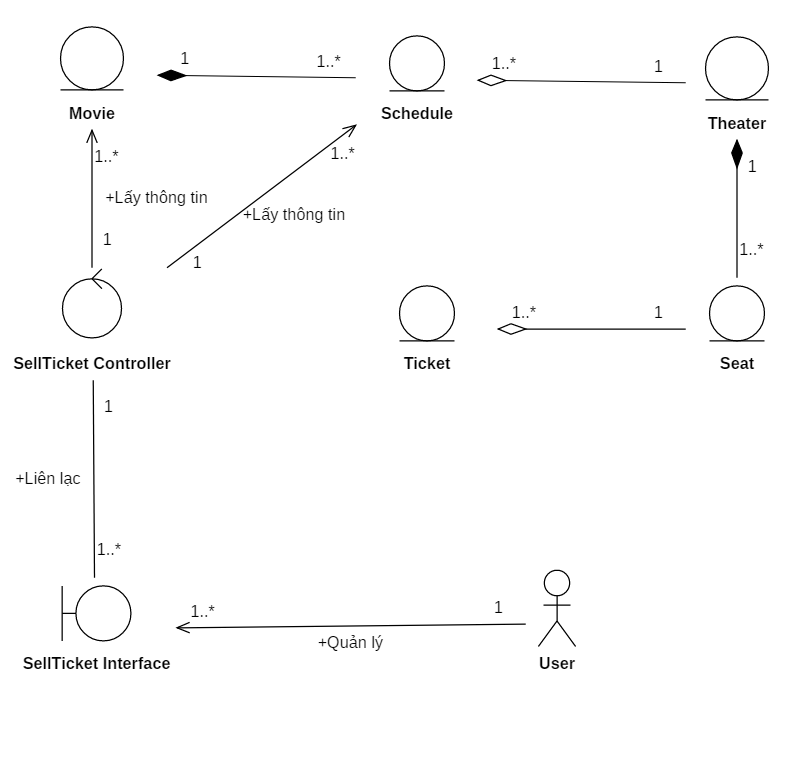
* + - 1. CLASS DIAGRAM



* 1. QUẢN LÝ ĐẶT MUA VÉ
  2. 1. SEQUENCE DIAGRAM

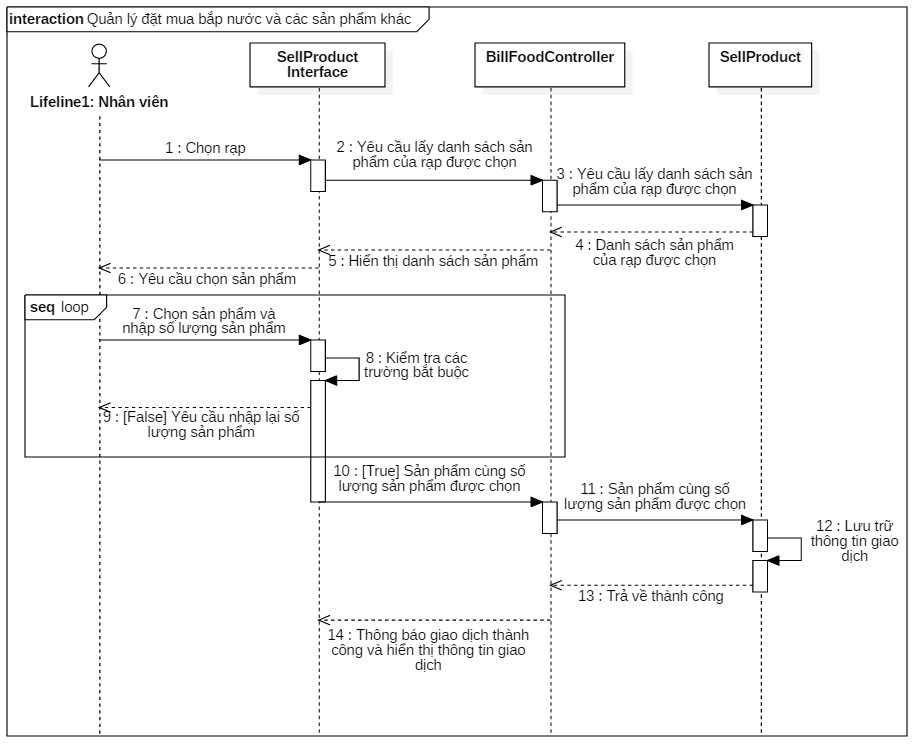


* + 1. CLASS DIAGRAM

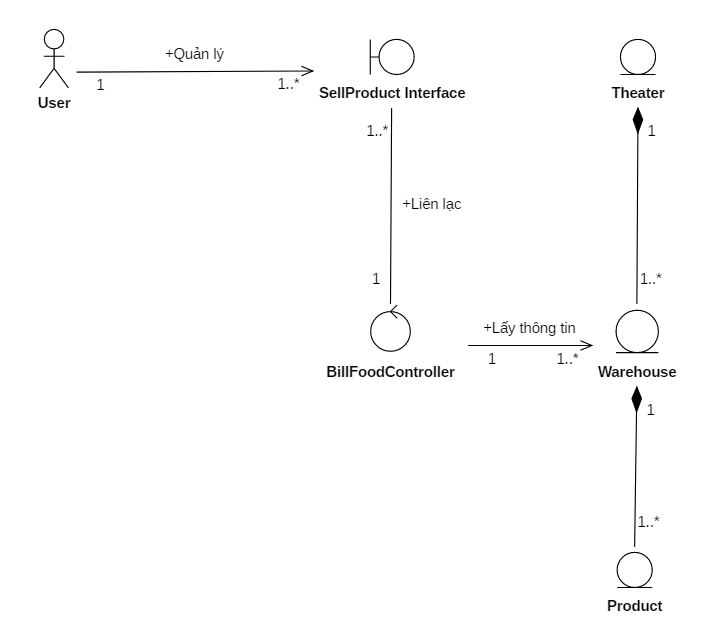


* 1. QUẢN LÝ ĐẶT MUA BẮP NƯỚC VÀ CÁC SẢN PHẨM KHÁC

5. 4. 1. SEQUENCE DIAGRAM

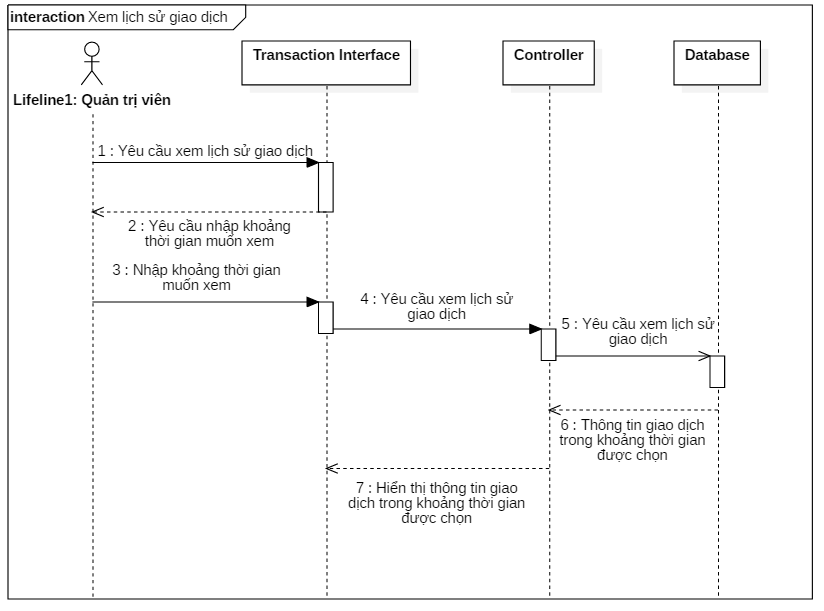


* + 1. CLASS DIAGRAM

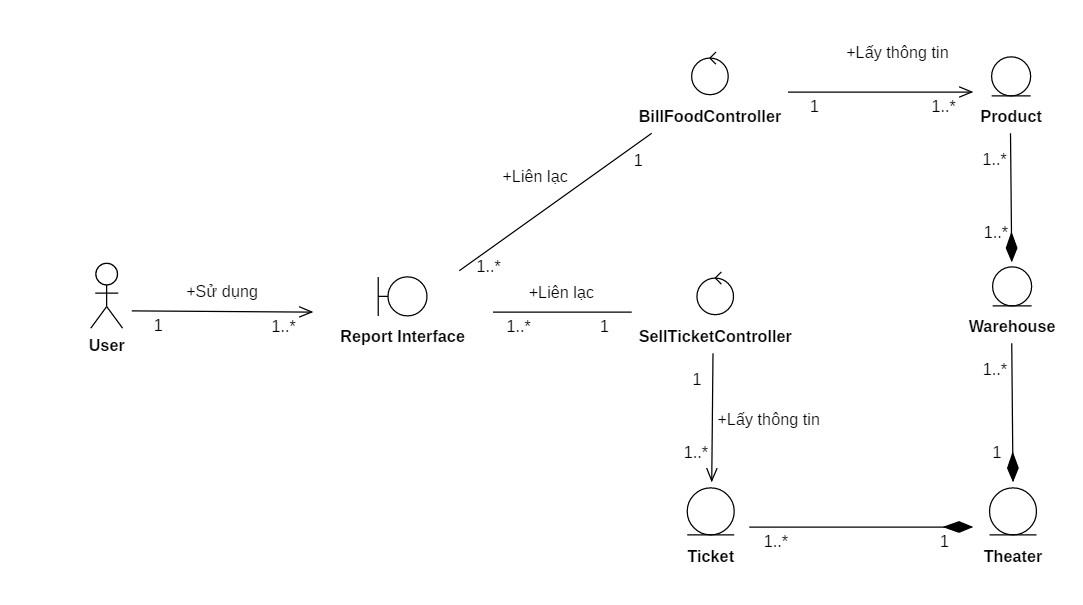


* 1. XEM LỊCH SỬ GIAO DỊCH

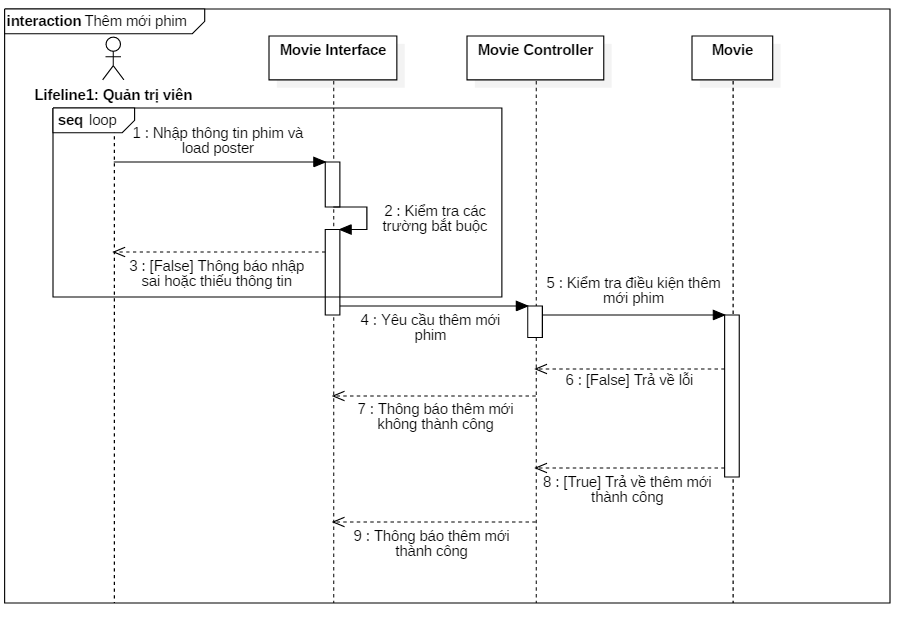
5. 5. 1. SEQUENCE DIAGRAM



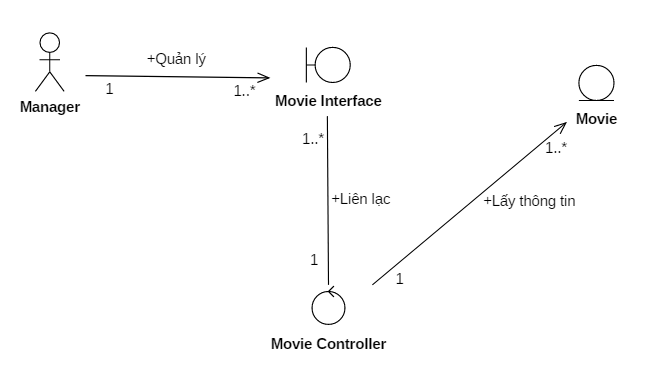
* + 1. CLASS DIAGRAM



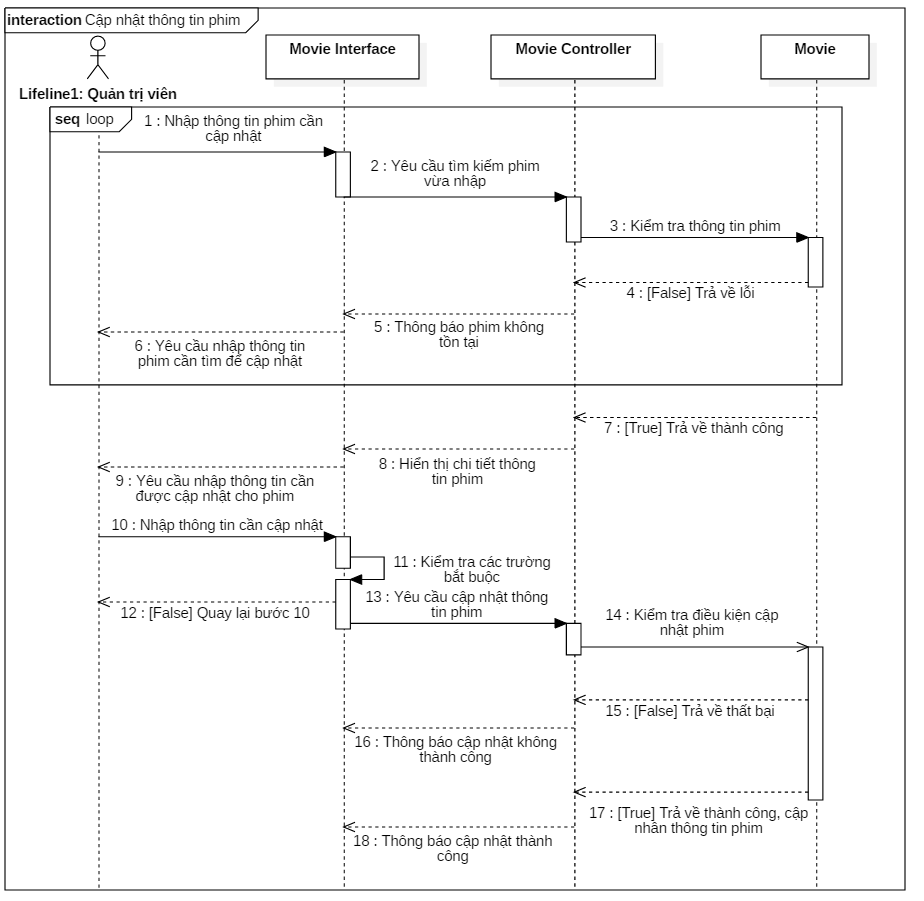
* 1. QUẢN LÝ PHIM
     1. THÊM MỚI PHIM
        1. SEQUENCE DIAGRAM



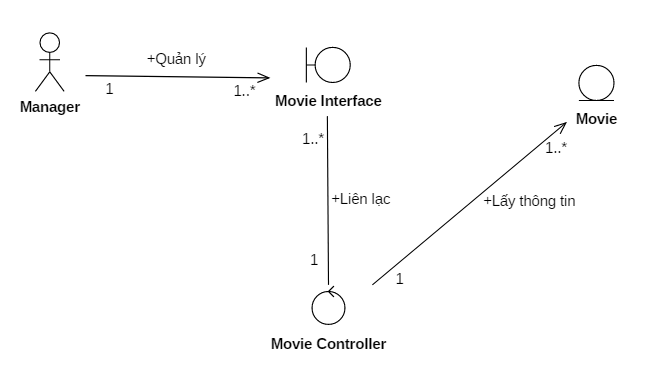
* + - 1. CLASS DIAGRAM



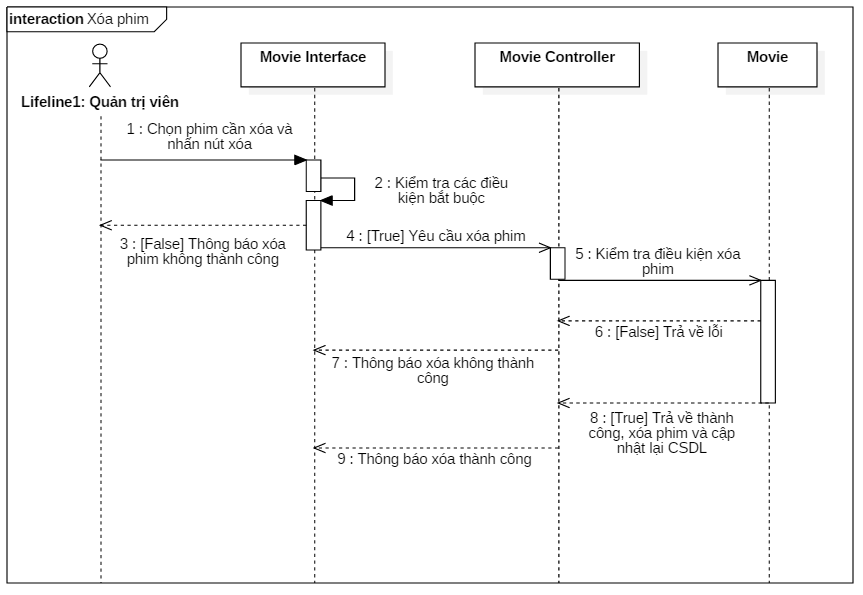
* + 1. CẬP NHẬT THÔNG TIN PHIM
       1. SEQUENCE DIAGRAM



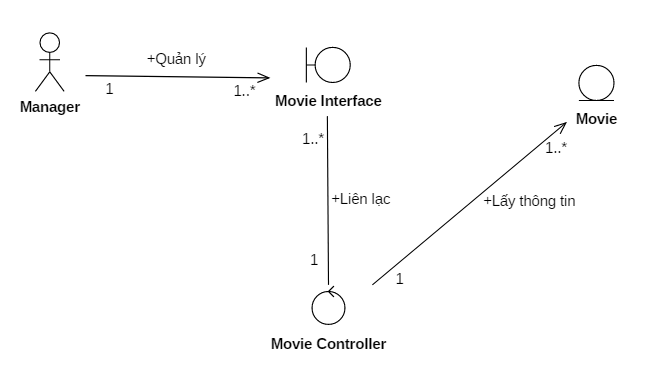
* + - 1. CLASS DIAGRAM



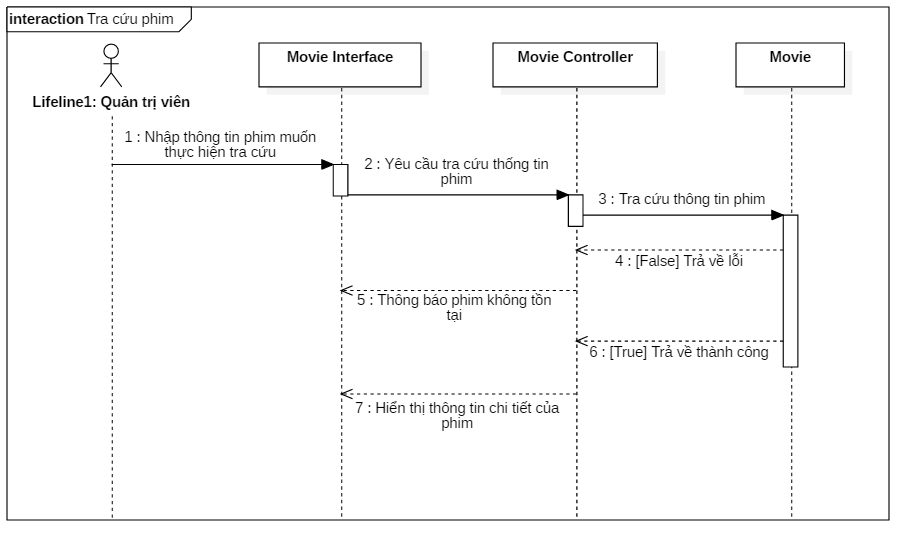
* + 1. XÓA PHIM
       1. SEQUENCE DIAGRAM



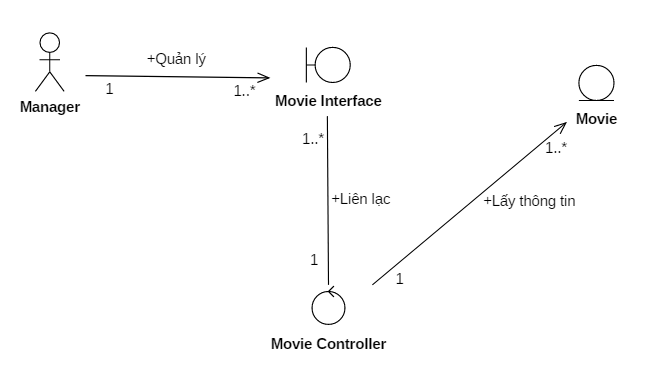
* + - 1. CLASS DIAGRAM



* + 1. TRA CỨU PHIM
       1. SEQUENCE DIAGRAM

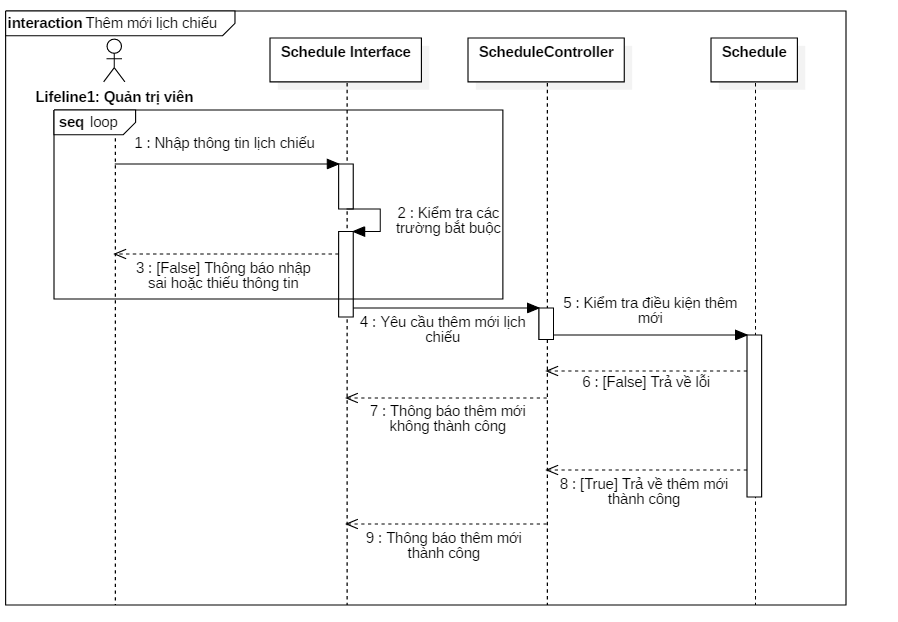


* + - 1. CLASS DIAGRAM

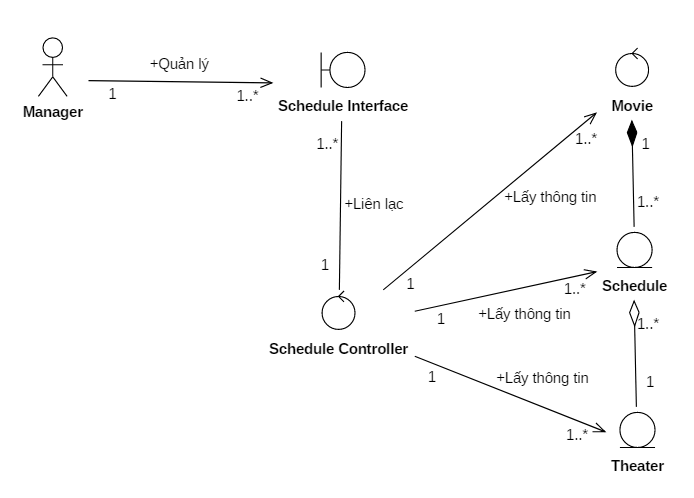


* 1. QUẢN LÝ LỊCH CHIẾU

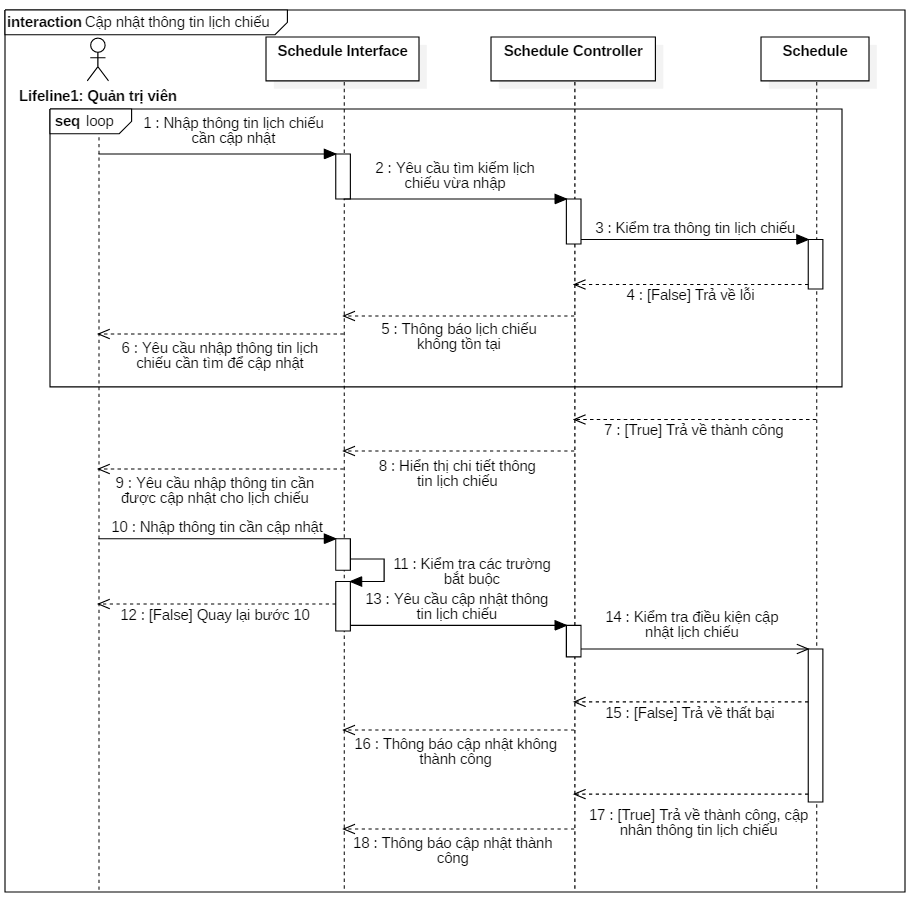
5. 7. 1. THÊM MỚI LỊCH CHIẾU
         1. SEQUENCE DIAGRAM



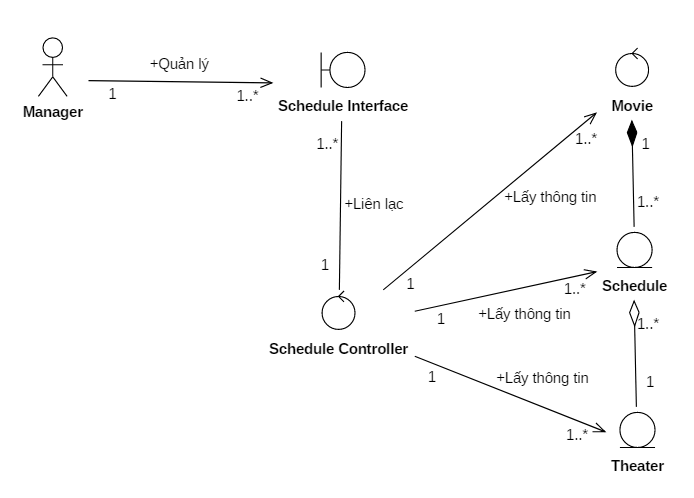
* + - 1. CLASS DIAGRAM



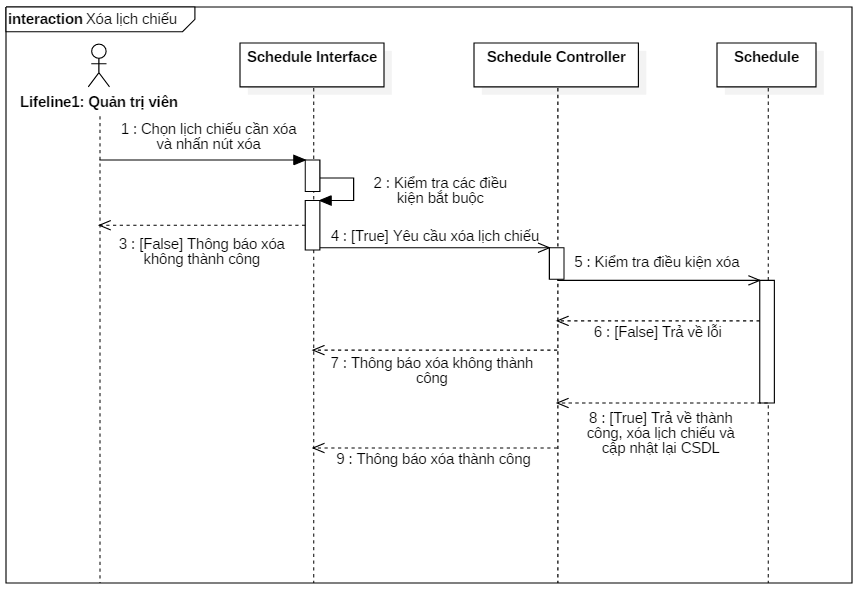
* + 1. CẬP NHẬT THÔNG TIN LỊCH CHIẾU
       1. SEQUENCE DIAGRAM



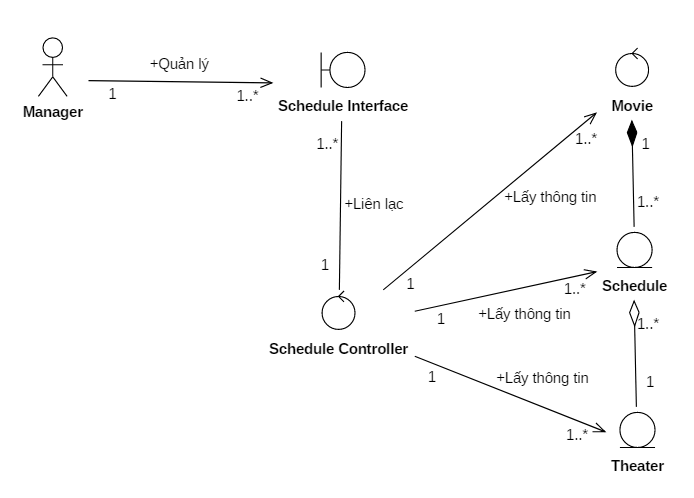
* + - 1. CLASS DIAGRAM



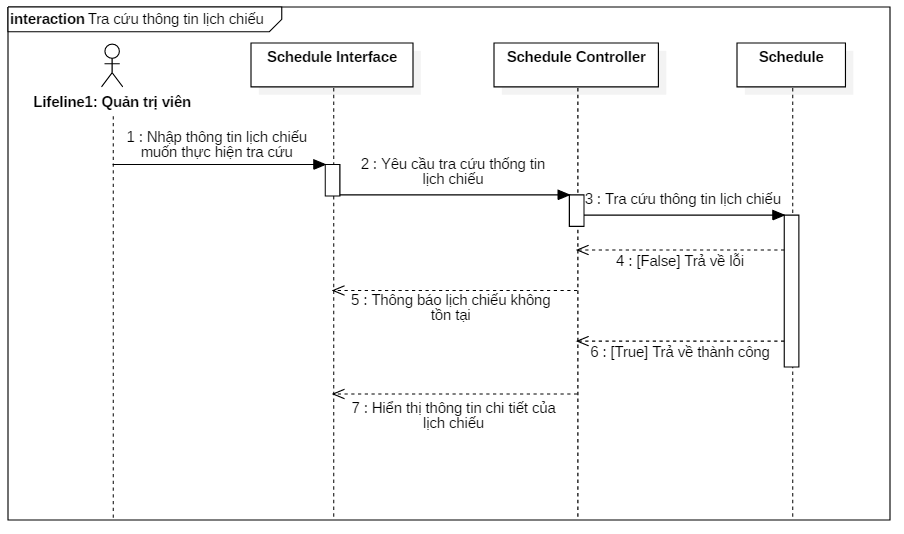
* + 1. XÓA LỊCH CHIẾU
       1. SEQUENCE DIAGRAM



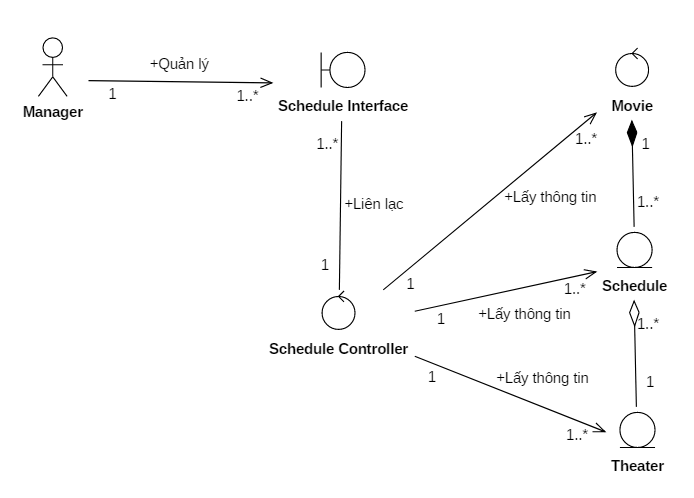
* + - 1. CLASS DIAGRAM



* + 1. TRA CỨU THÔNG TIN LỊCH CHIẾU
       1. SEQUENCE DIAGRAM



* + - 1. CLASS DIAGRAM

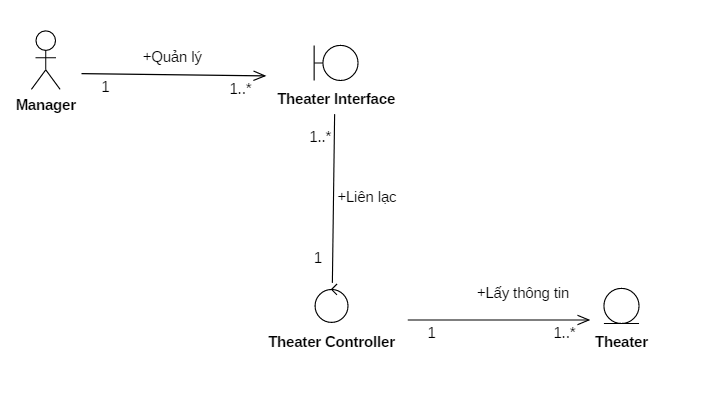


* 1. QUẢN LÝ PHÒNG CHIẾU

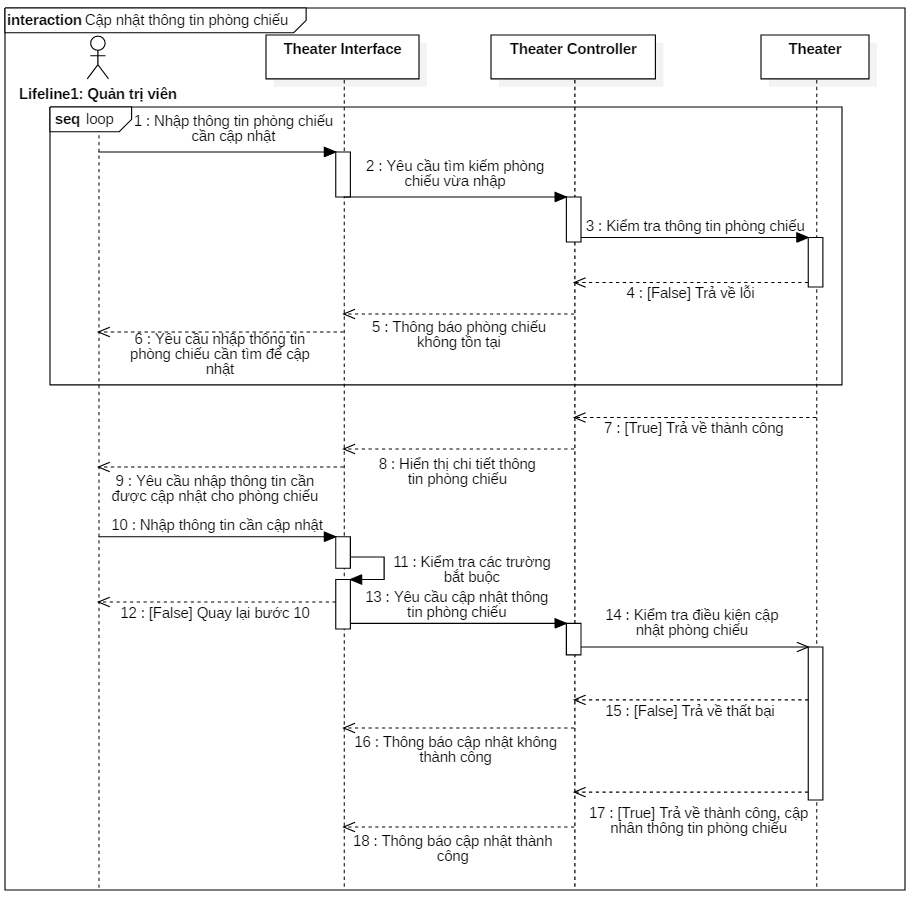
5. 8. 1. THÊM MỚI PHÒNG CHIẾU
         1. SEQUENCE DIAGRAM



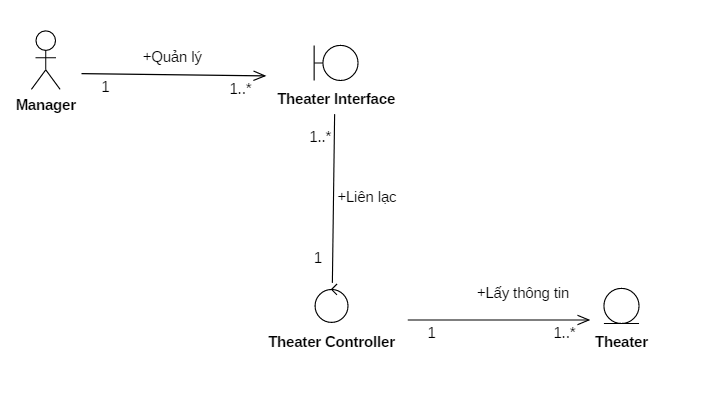
* + - 1. CLASS DIAGRAM



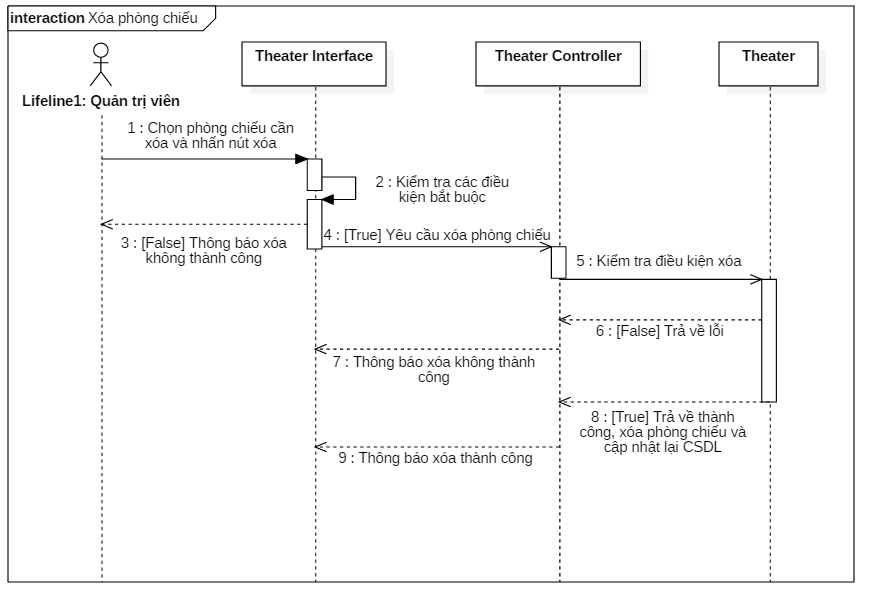
* + 1. CẬP NHẬT THÔNG TIN PHÒNG CHIẾU
       1. SEQUENCE DIAGRAM



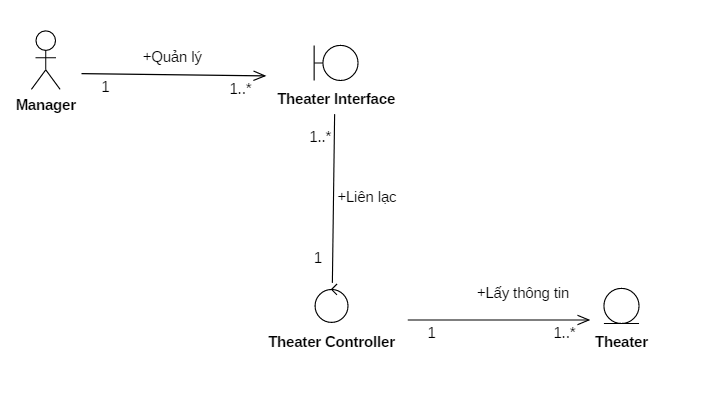
* + - 1. CLASS DIAGRAM



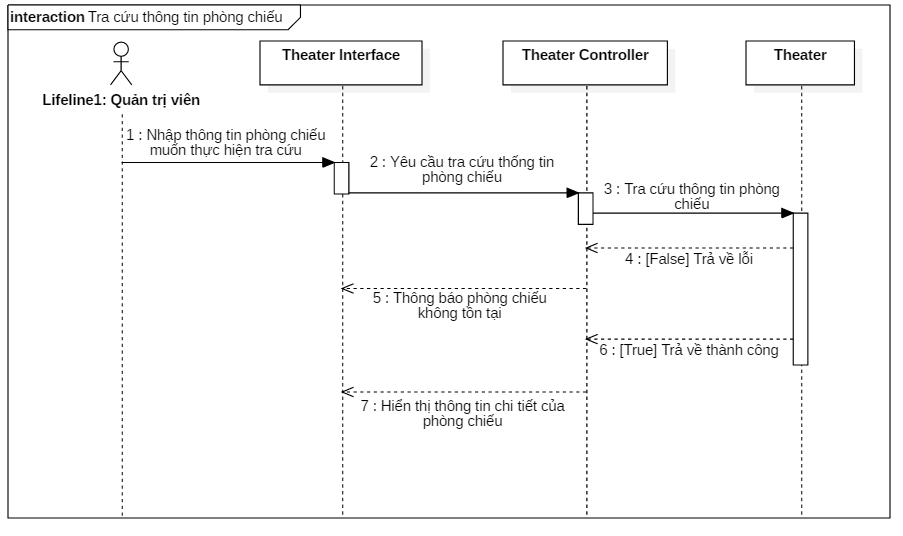
* + 1. XÓA PHÒNG CHIẾU
       1. SEQUENCE DIAGRAM



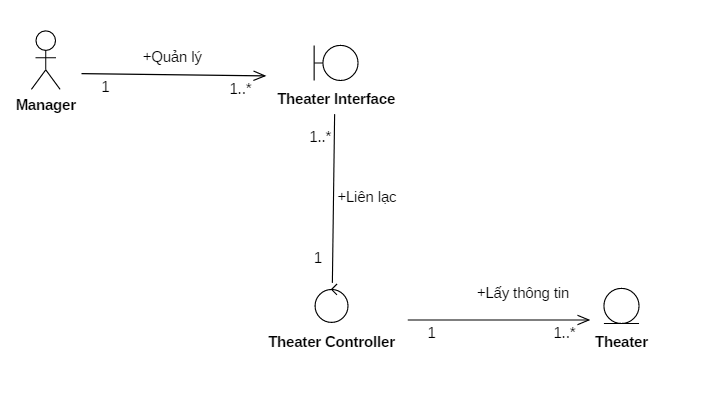
* + - 1. CLASS DIAGRAM



* + 1. TRA CỨU PHÒNG CHIẾU
       1. SEQUENCE DIAGRAM



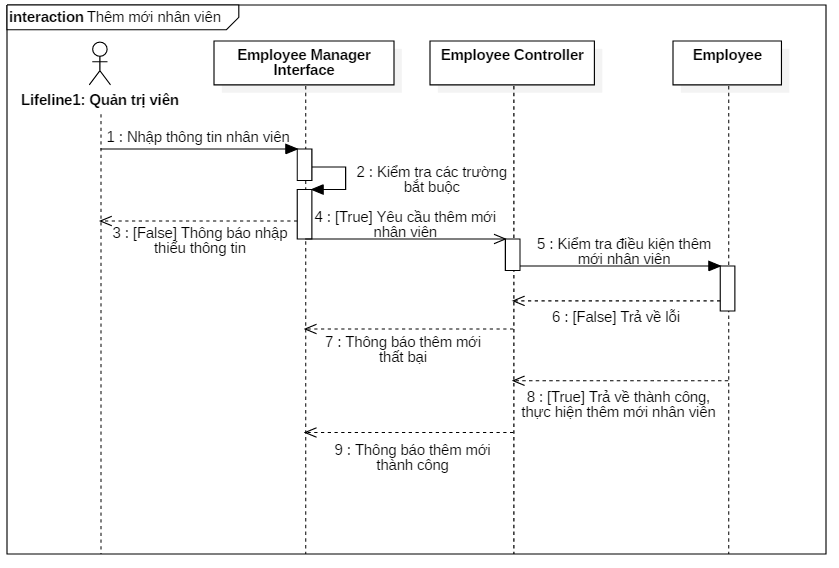
* + - 1. CLASS DIAGRAM



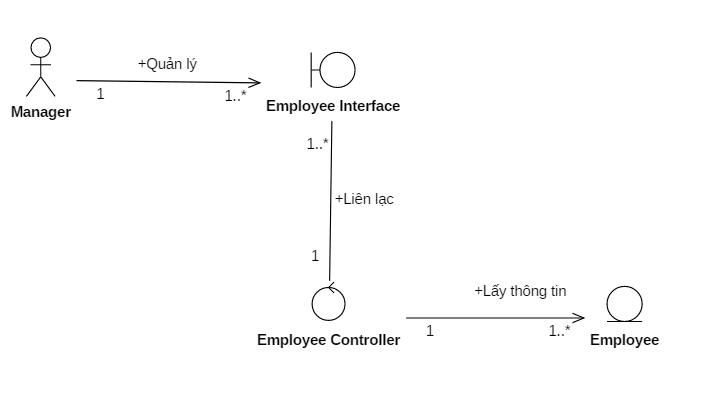
* 1. QUẢN LÝ NHÂN VIÊN



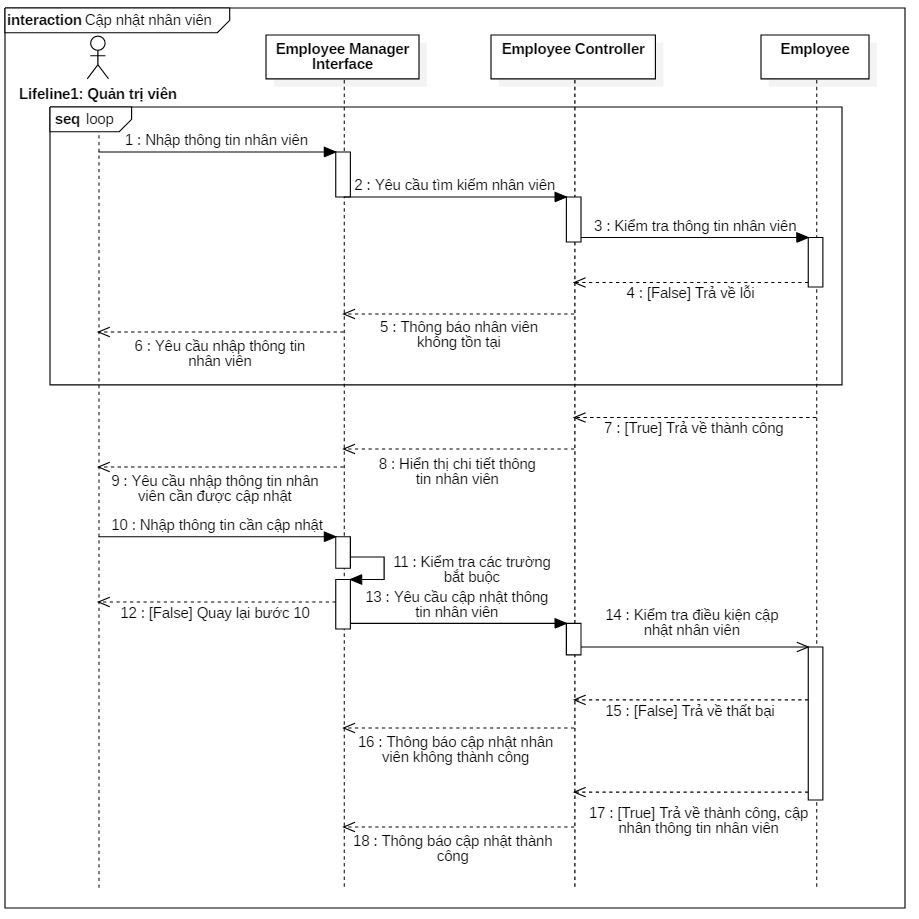

  7. 1. THÊM MỚI NHÂN VIÊN
        1. SEQUENCE DIAGRAM



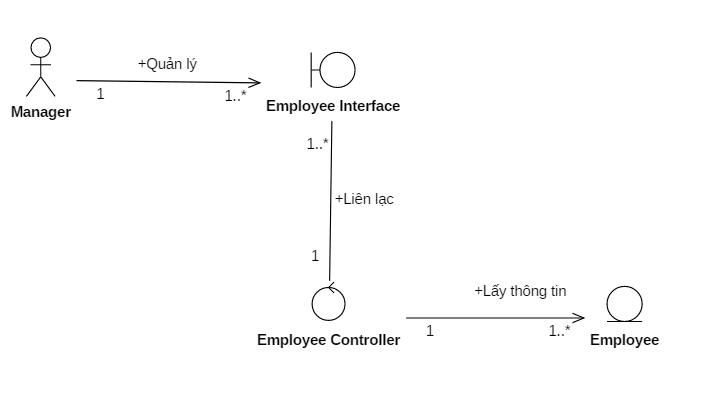
* + - 1. CLASS DIAGRAM



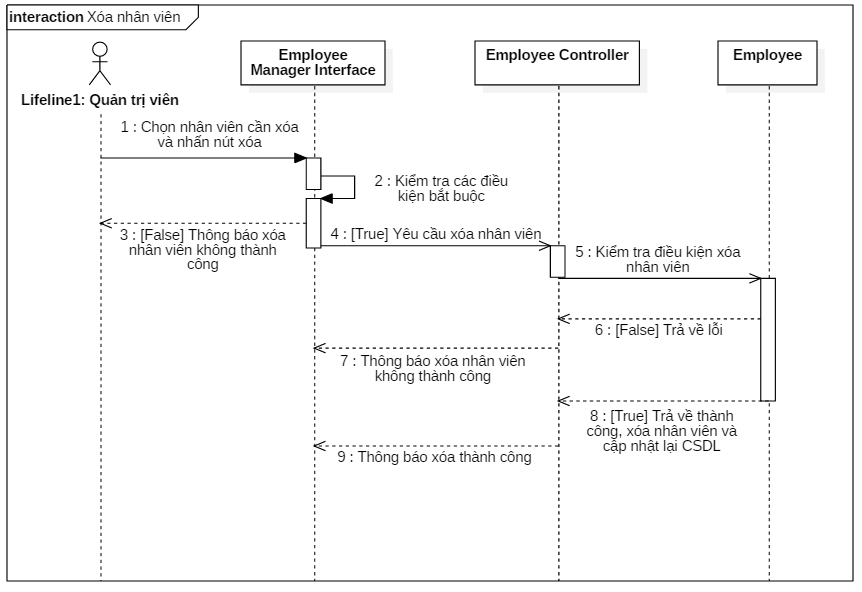
* + 1. CẬP NHẬT NHÂN VIÊN
       1. SEQUENCE DIAGRAM



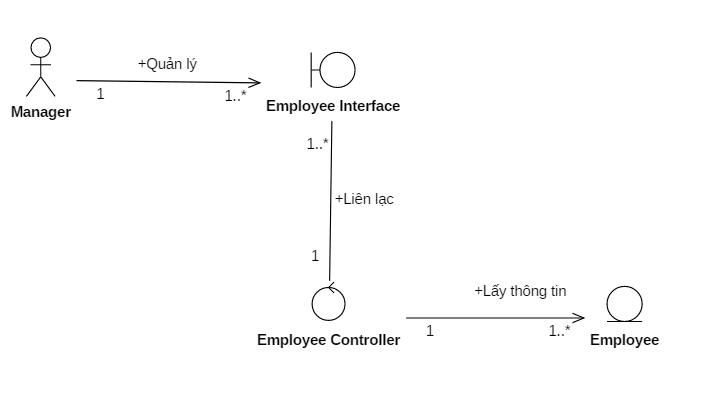
* + - 1. CLASS DIAGRAM



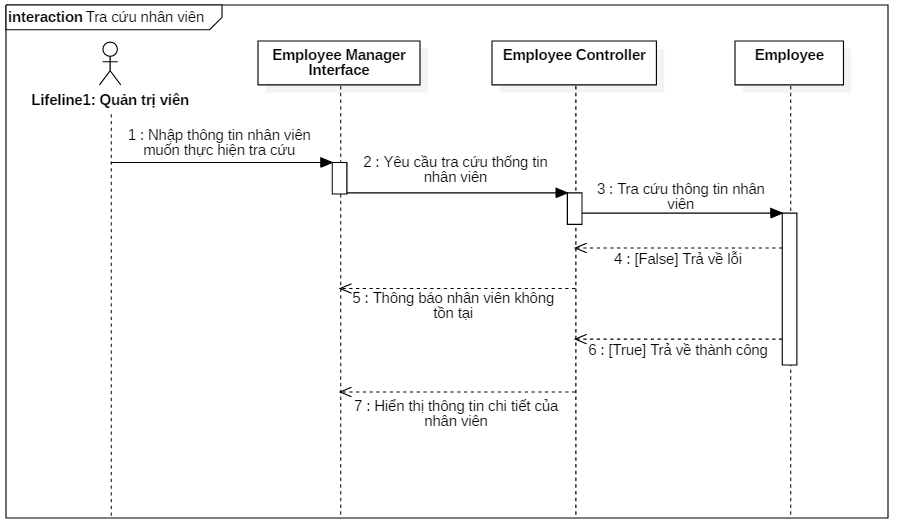
* + 1. XÓA NHÂN VIÊN
       1. SEQUENCE DIAGRAM



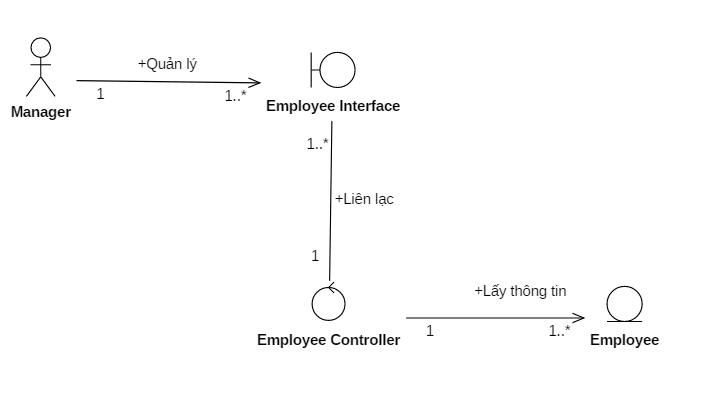
* + - 1. CLASS DIAGRAM



* + 1. TRA CỨU NHÂN VIÊN
       1. SEQUENCE DIAGRAM



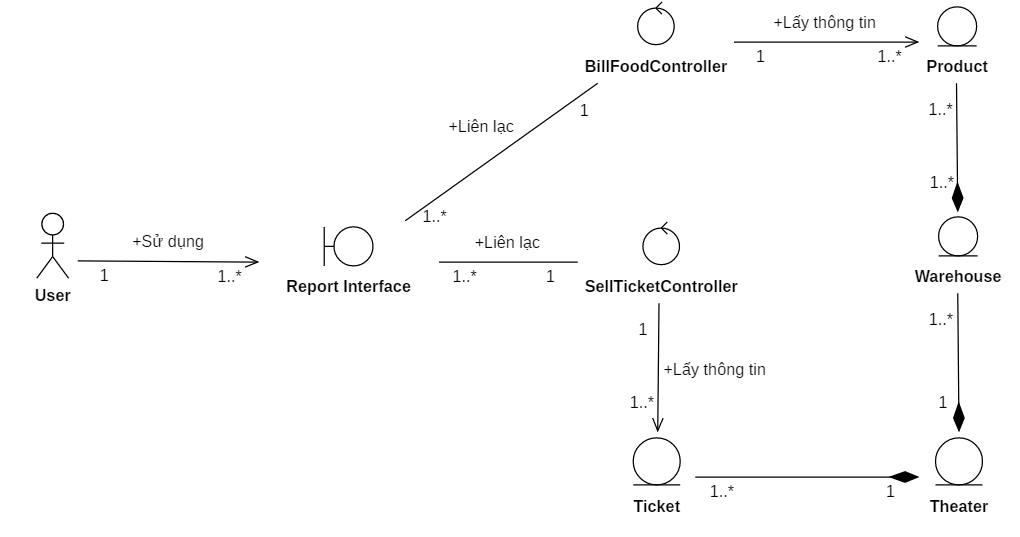
* + - 1. CLASS DIAGRAM



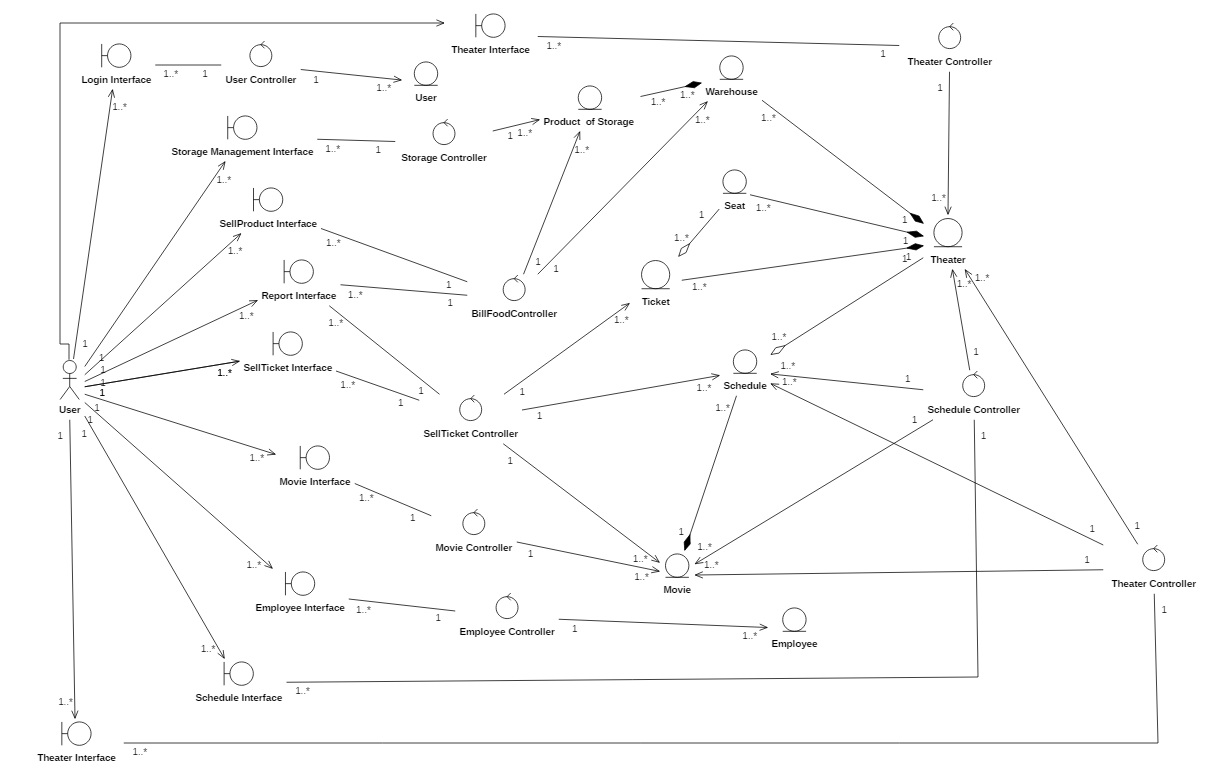
* 1. BÁO CÁO DOANH THU
  2. 1. SEQUENCE DIAGRAM



* + 1. CLASS DIAGRAM



* 1. CLASS DIAGRAM TỔNG



* 1. ĐẶC TẢ DIAGRAM

|  |  |  |
| --- | --- | --- |
| Employee | | |
| Attribute | Tên biến | Ý nghĩa |
|  | id: int | Mã id của nhân viên |
|  | access: int | Phân quyền truy cập cho nhân viên |
|  | name: String | Tên nhân viên |
|  | Addr: String | Địa chỉ của nhân viên |
|  | password: String | Password tài khoản của nhân viên |
|  | Email : String | Email của nhân viên |
|  | Tel : String | Số điện thoại của nhân viên |
|  | no: int | Số thứ tự của nhân viên |
| Operation | Tên hàm | Ý nghĩa |
|  | getId(): int | Lấy mã id của nhân viên |
|  | setId(int id): void | Gán mã id mới cho nhân viên |
|  | getAccess(): int | Lấy mã phân quyền cho nhân viên |
|  | setAccess(int access): void | Gán mã phân quyền cho nhân viên |
|  | getName(): String | Lấy tên nhân viên |
|  | setName(String name): void | Gán tên mới cho nhân viên |
|  | getAddr (): String | Lấy địa chỉ nhân viên |
|  | setAddr(String addr): void | Gán địa chỉ mới cho nhân viên |
|  | getPassword(): String | Lấy password nhân viên |
|  | setPassword(String password): void | Gán password mới cho nhân viên |
|  | getTel(): String | Lấy số điện thoại cho nhân viên |
|  | setTel(String tel): Void | Gán số điện thoại cho nhân viên |
|  | getEmailt(): String | Lấy email nhân viên |
|  | setEmail(String email): void | Gán email mới cho nhân viên |
|  | getNo() : int | Lấy số thứ tự cho nhân viên |
|  | setNo(int no) : void | Gán số thứ tự cho nhân viên |
|  | insert(Employee emp): int | Thêm 1 nhân viên mới vào database |
|  | update(Employee emp): int | Chỉnh sửa 1 nhân viên trong database |
|  | delete(int id): int | Xóa 1 nhân viên trong database |
|  | select(String condt): List<Employee> | Lấy danh sách nhân viên phù hợp với điều kiện, nếu còn rỗng thì lấy danh sách tất cả nhân viên có trong database |

|  |  |  |
| --- | --- | --- |
| EmployeeController | | |
| Operation | Tên hàm | Ý nghĩa |
|  | add(Theater theater): int | Báo model thêm 1 nhân viên mới vào database |
|  | modify(Theater theater) | Báo model chỉnh sửa 1 nhân viên trong database |
|  | delete(int ID) | Báo model xóa 1 nhân viên trong database |
|  | Fetch(String condt) : List<Employee> | Lấy danh sách nhân viên |
|  | FetchAll() : List<Employee> | Lấy danh sách nhân viên phù hợp với điều kiện, nếu condt null thì lấy danh sách nhân viên có trong database |

|  |  |  |  |
| --- | --- | --- | --- |
| Movie | | | |
| Attribute | Tên biến | | Ý nghĩa |
|  | id: Integer | | Mã id của phim |
|  | name: String | | Tên phim |
|  | duration: Integer | | Độ dài phim |
|  | image: byte[] | | Hình ảnh của phim |
|  | status: Integer | | Trạng thái của phim |
| Operation | Tên hàm | | Ý nghĩa |
|  | getId(): Integer | | Lấy mã id của phim |
|  | setId(int id): void | | Gán giá trị id của phim |
|  | getName(): String | | Lấy tên phim |
|  | setName(String name): void | | Gán tên cho phim |
|  | getDuration(): Integer | | Lấy độ dài của phim |
|  | setDuration(int dur): void | | Gán giá trị độ dài phim |
|  | getImage(): byte[] | | Lấy hình ảnh của phim dưới dạng chuỗi byte |
|  | setImage(byte[] img): void | | Gán hình ảnh cho phim |
|  | getStatus(): Integer | | Lấy trạng thái của phim |
|  | setStatus(int status): void | | Đặt trạng thái cho phim |
|  | insert(Movie movie): Integer | | Thêm một phim vào cơ sở database |
|  | update(Movie movie): Integer | | Chỉnh sửa thông tin của phim trong database |
|  | delete(int id): Integer | | Xóa phim trong database với mã id của phim cho trước |
|  | select(String condt): List<Movie> | | Lấy ra danh sách movie phù hợp với điều kiện, nếu condt null thì lấy danh sách movie có trong database |
|  | selectScheduledMovie(String condt): List<Movie> | | Lấy danh sách movie đã có lịch chiếu |
| Movie Controller |
| Attribute | Tên biến | Ý nghĩa | | |
|  | movieM: Movie | Model của phim | | |
| Operation | Tên hàm | Ý nghĩa | | |
|  | add(Movie movie): Integer | Báo cho model thêm 1 phim | | |
|  | modify(Movie movie): Integer | Báo cho model chỉnh sửa 1 phim | | |
|  | delete(int id): Integer | Báo cho model xóa 1 phim dựa trên id | | |
|  | selectAll(): List<Movie> | Báo cho model lấy danh sách tất cả các phim có trong database | | |
|  | select(String condt): List<Movie> | Báo cho model lấy danh sách các phim phù hợp với điều kiện có trong database | | |
|  | selectScheduledMovie(String condt): List<Movie> | Báo cho model lấy danh sách các phim có lịch chiếu trong database | | |
| Sale |
| Attribute | Tên biến | | Ý nghĩa | | |
|  | id: Integer | | Mã id của phiếu sale | | |
|  | empId: Integer | | Mã id của nhân viên làm phiếu sale | | |
|  | time: Date | | Thời gian lập phiếu sale | | |
|  | payment: float | | Số tiền khách trả cho phiếu sale | | |
|  | change: float | | Số tiền thối lại cho khách | | |
|  | type: Integer | | Kiểu thanh toán | | |
|  | status: Integer | | Trạng thái của phiếu sale | | |
|  | prices: float | | Tổng tiền thanh toán của phiếu sale | | |
| Operation | Tên hàm | | Ý nghĩa | | |
|  | getId(): Integer | | Lấy giá trị mã id của phiếu sale | | |
|  | set(int id): void | | Gán giá trị mã id của phiếu sale | | |
|  | getEmpId(): Integer | | Lấy giá trị mã id nhân viên lập phiếu sale | | |
|  | setEmpId(int empId): void | | Gán giá trị mã id nhân viên lập phiếu sale | | |
|  | getTime(): Integer | | Lấy thời gian lập phiếu sale | | |
|  | setTime(Date time) | | Gán giá trị thời gian lập phiếu sale | | |
|  | getPayment(): float | | Lấy giá trị số tiền thanh toán của khách | | |
|  | setPayment(float payment): void | | Gán giá trị số tiền thanh toán của khách | | |
|  | getChange(): float | | Lấy giá trị số tiền thối lại cho khách | | |
|  | setChange(float change): void | | Gán giá trị số tiền thối lại cho khách | | |
|  | getType(): Integer | | Lấy kiểu thanh toán của khách cho phiếu sale | | |
|  | setType(int type): void | | Gán kiểu thanh toán của khách cho phiếu sale | | |
|  | getStatus(): Integer | | Lấy trạng thái của phiếu sale | | |
|  | setStatus(int status): void | | Gán trạng thái của phiếu sale | | |
|  | insert(Sale sale): Integer | | Thêm 1 phiếu sale vào database | | |
|  | update(Sale sale): Integer | | Chỉnh sửa 1 phiếu sale trong database | | |
|  | delete(int id): Integer | | Xóa 1 phiếu sale trong database | | |
|  | select(String condt): List<Sale> | | Lấy danh sách phiếu sale phù hợp với điều kiện, nếu condt rỗng thì lấy danh sách tất cả các phiếu sale có trong database | | |
|  | doSale(List<Ticket> tickets, Sale sale): boolean | | Lập phiếu sale dựa trên danh sách ticket khách đã mua | | |

|  |  |  |
| --- | --- | --- |
| SaleController | | |
| Attribute | Tên biến | Ý nghĩa |
|  | saleM: Sale | Model của phiếu sale |
| Operation | Tên hàm | Ý nghĩa |
|  | add(Sale sale): int | Báo model thêm 1 phiếu sale |
|  | modify(Sale sale): int | Báo model chỉnh sửa 1 phiếu sale |
|  | delete(Sale sale): int | Báo model xóa 1 phiếu sale |
|  | selectAll(): List<Sale> | Báo model lấy danh sách tất cả các phiếu sale có trong database |
|  | select(String condt): List<Sale> | Báo model lấy danh sách các phiếu sale phù hợp với điều kiện trong database |
|  | doSale(List<Ticket> tickets, Sale sale): boolean | Báo model lập phiếu sale dựa trên danh sách ticket khách đã mua |

|  |  |  |
| --- | --- | --- |
| SaleItem | | |
| Attribute | Tên biến | Ý nghĩa |
|  | id: int | Mã id của SaleItem |
|  | ticketId: int | Mã id của ticket |
|  | saleId: int | Mã id của phiếu sale |
|  | price: float | Giá của ticket |
| Operation | Tên hàm | Ý nghĩa |
|  | getId(): int | Lấy mã id |
|  | setId(int id): void | Gán giá trị cho mã id |
|  | getTicketId(): int | Lấy mã id của ticket |
|  | setTicketId(int ticketId): void | Gán giá trị cho mã id của ticket |
|  | getSaleId(): int | Lấy mã id của phiếu sale |
|  | setSaleId(int saleId): void | Gán giá trị cho mã id của phiếu sale |
|  | getPrice(): float | Lấy giá của ticket |
|  | setPrice(float price): void | Gán giá trị cho giá của ticket |
|  | insert(SaleItem saleItem): int | Thêm 1 saleItem vào database |
|  | update(SaleItem saleItem): int | Chỉnh sửa 1 saleItem trong database |
|  | delete(int id): int | Xóa 1 saleItem trong database |
|  | select(String condt): List<SaleItem> | Lấy danh sách saleItem phù hợp với điều kiện, nếu condt rỗng thì lấy danh sách tất cả các saleItem có trong database |

|  |  |  |  |
| --- | --- | --- | --- |
| SaleItemController | | | |
| Attribute | Tên biến | Ý nghĩa |
|  | saleItemM: SaleItem | Model của saleItem |
| Operation | Tên hàm | Ý nghĩa |
|  | add(SaleItem saleItem): int | Báo model thêm 1 saleitem vào database |
|  | modify(SaleItem saleItem): int | Báo model chỉnh sửa 1 saleitem trong database |
|  | delete(int id): int | Báo model xóa 1 saleitem trong database |
|  | selectAll(): List<SaleItem> | Báo model lấy danh sách tất cả các saleitem có trong database |
|  | select(String condt): List<SaleItem> | Báo model lấy danh sách saleitem phù hợp với điều kiện trong database |

|  |  |  |
| --- | --- | --- |
| Ticket | | |
| Attribute | Tên biến | Ý nghĩa |
|  | id: int | Mã id của ticket |
|  | seatId: int | Mã id của ghế ngồi |
|  | scheduleId: int | Mã id của lịch chiếu |
|  | price: float | Giá của ticket |
|  | status: int | Trạng thái của ticket |
|  | lockedTime: Date | Thời gian lập ticket |
|  | movieName: String | Tên phim |
|  | schedule: Schedule | Model của lịch chiếu |
|  | seat: Seat | Model của ghế ngồi |
|  | currentLockedTime: Date | Thời gian lập đã khóa của ticket hiện tại |
| Operation | Tên hàm | Ý nghĩa |
|  | getId(): int | Lấy mã id của vé |
|  | setId(int id): void | Gán giá trị cho mã id |
|  | getSeatId(): int | Lấy mã id của ghế ngồi |
|  | setSeatId(int seatId): void | Gán giá trị mới cho mã id của ghế ngồi trong vé |
|  | getScheduleId(): int | Lấy mã id của lịch chiếu |
|  | setScheduleId(int scheduleId): void | Gán giá trị mới cho mã id của lịch chiếu trong vé |
|  | getPrice(): float | Lấy giá của vé |
|  | setPrice(float price): void | Gán giá trị mới cho giá của vé |
|  | getStatus(): int | Lấy trạng thái của vé |
|  | setStatus(int status): void | Gán giá trị mới cho trạng thái của vé |
|  | getLockedTime(): Date | Lấy thời gian lập vé mới |
|  | setLockedTime(Date lockedTime): void | Gán giá trị mới cho thời gian lập vé |
|  | getSchedule(): Schedule | Lấy model của lịch chiếu |
|  | setSchedule(Schedule schedule): void | Gán model mới của lịch chiếu |
|  | getSeat(): Seat | Lấy model của ghế |
|  | setSeat(Seat seat): void | Gán model mới của ghế |
|  | getMovieName(): String | Lấy tên phim |
|  | setMovieName(String movieName): void | Gán tên phim mới trong vé |
|  | getCurrentLockedTime(): Date | Lấy thời gian lập trước đó của vé |
|  | setCurrentLockedTime(Date currentLockedTime): void | Gán giá trị mới cho thời gian lập vé |
|  | insert(Ticket ticket): int | Thêm 1 vé mới vào database |
|  | update(Ticket ticket): int | Chỉnh sửa 1 vé trong database |
|  | delete(int id): int | Xóa 1 vé trong database |
|  | select(String condt): List<Ticket> | Lấy danh sách vé phù hợp với điều kiện, nếu còn rỗng thì lấy danh sách tất cả các vé có trong database |

|  |  |  |
| --- | --- | --- |
| TicketController | | |
| Attribute | Tên biến | Ý nghĩa |
|  | ticketM: Ticket | Model của vé |
| Operation | Tên hàm | Ý nghĩa |
|  | add(Ticket ticket): int | Báo model thêm 1 vé vào database |
|  | modify(Ticket ticket): int | Báo model chỉnh sửa 1 vé trong database |
|  | delete(int ID): int | Báo model xóa 1 vé trong database |
|  | select(String condt): List<Ticket> | Báo model lấy danh sách vé phù hợp với điều kiện trong database |
|  | selectAll(): List<Ticket> | Báo model lấy danh sách tất cả các vé có trong database |
| SellTicketController | | | |
| Attribute | Tên biến | Ý nghĩa | |
|  | ticketList: List<Ticket> |  | |
| Operation | Tên hàm | Ý nghĩa | |
|  | makeNewTicket(Schedule sched, Seat seat): Ticket | Báo TicketController thêm 1 vé mới vào database | |
|  | makeNewSale(): void | Tạo danh sách vé mua mới | |
|  | addTicket(Ticket ticket): void | Thêm 1 vé vào danh sách vé mua | |
|  | removeTicket(Ticket ticket): void | Xóa 1 vé trong danh sách vé mua | |
|  | removeAllTicket(): void | Xóa hết vé trong danh sách vé mua | |
|  | getInfo(): String | Xuất thông tin mua vé | |
|  | isTicketSelected(Ticket ticket): boolean | Kiểm tra xem vé có được chọn không | |
|  | clearSale(): void | Xóa danh sách vé mua cũ và tạo danh sách vé mua mới | |
|  | doSale(Sale sale): int | Báo SaleController lập 1 phiếu sale mới dựa trên danh sách vé mua | |

|  |  |  |
| --- | --- | --- |
| Schedule | | |
| Attribute | Tên biến | Ý nghĩa |
|  | id: int | Mã id của lịch chiếu |
|  | theaterId: int | Mã id của phòng chiếu |
|  | movieId: int | Mã id của phim |
|  | time: Date | Thời gian chiếu |
|  | price: double | Giá lịch chiếu phim |
| Operation | Tên hàm | Ý nghĩa |
|  | getId: int | Lấy mã id của lịch chiếu |
|  | setId(int id): void | Gán giá trị mới cho mã id của lịch chiếu |
|  | getTheaterId(): int | Lấy mã id của phòng chiếu |
|  | setTheaterId(int theaterId): void | Gán giá trị mới cho mã id phòng chiếu của lịch chiếu |
|  | getMovieId(): int | Lấy mã id của phim |
|  | setMovieId(int movieId): void | Gán giá trị mới cho mã id phim của lịch chiếu |
|  | getTime(): Date | Lấy thời gian chiếu |
|  | setTime(Date time): void | Gán thời gian chiếu mới cho lịch chiếu |
|  | getPrice(): double | Lấy giá của lịch chiếu |
|  | setPrice(double price): void | Gán giá trị mới cho giá của lịch chiếu |
|  | insert(Schedule schedule): int | Thêm 1 lịch chiếu mới vào database |
|  | update(Schedule sched): int | Chỉnh sửa 1 lịch chiếu trong database |
|  | delete(int id): int | Xóa 1 lịch chiếu trong database |
|  | select(String condt): List<Schedule> | Lấy danh sách lịch chiếu phù hợp với điệu kiện, nếu condt rỗng thì lấy danh sách tất cả lịch chiếu có trong database |

|  |  |  |
| --- | --- | --- |
| ScheduleController | | |
| Attribute | Tên biến | Ý nghĩa |
|  | scheduleM: Schedule | Model của lịch chiếu |
| Operation | Tên hàm | Ý nghĩa |
|  | add(Schedule schedule): int | Báo model thêm 1 lịch chiếu vào database |
|  | modify(Schedule schedule): int | Báo model chỉnh sửa 1 lịch chiếu trong database |
|  | delete(int id): int | Báo model xóa 1 lịch chiếu trong database |
|  | selectAll(): List<Schedule> | Báo model lấy danh sách tất cả các lịch chiếu có trong database |
|  | select(String condt): List<Schedule> | Báo model lấy danh sách các lịch chiếu phù hợp với điều kiện trong database |

|  |  |  |
| --- | --- | --- |
| Seat | | |
| Attribute | Tên biến | Ý nghĩa |
|  | id: int | Mã id của ghế |
|  | theaterId: int | Mã id phòng chiếu của ghế |
|  | row: int | Vị trí hàng của ghế |
|  | column: int | Vị trí cột của ghế |
|  | status: int | Trạng thái của ghế |
| Operation | Tên hàm | Ý nghĩa |
|  | getId(): int | Lấy mã id của ghế |
|  | setId(int id): void | Gán giá trị mã id mới cho ghế |
|  | getTheaterId(): int | Lấy mã id phòng chiếu |
|  | setTheaterId(int theaterId): void | Gán giá trị mã id phòng chiếu mới cho ghế |
|  | getRow(); int | Lấy vị trí hàng của ghế |
|  | setRow(int row): void | Gán vị trí hàng mới cho ghế |
|  | getColumn(): int | Lấy vị trí cột của ghế |
|  | setColumn(int column): int | Gán vị trí cột mới cho ghế |
|  | getStatus(): int | Lấy trạng thái của ghế |
|  | setStatus(int status): void | Gán trạng thái mới cho ghế |
|  | insert(Seat seat): void | Thêm 1 ghế mới vào database |
|  | update(Seat seat): int | Chỉnh sửa 1 ghế trong database |
|  | delete(int id): int | Xóa 1 ghế trong database |
|  | select(String condt): List<Seat> | Lấy danh sách ghế phù hợp với điều kiện, nếu condt rỗng thì lấy danh sách tất cả ghế có trong database |

|  |  |  |
| --- | --- | --- |
| SeatController | | |
| Attribute | Tên biến | Ý nghĩa |
|  | seatM: Seat | Model của ghế |
| Operation | Tên hàm | Ý nghĩa |
|  | add(Seat seat): int | Báo model thêm 1 ghế vào database |
|  | modify(Seat seat) | Báo model chỉnh sửa 1 ghế trong database |
|  | delete(int ID) | Báo model xóa 1 ghế trong database |
|  | select(String condt): List<Seat> | Báo model lấy danh sách ghế phù hợp với điều kiện trong database |
|  | selectAll():List<Seat> | Báo model lấy danh sách tất cả ghế có trong database |

|  |  |  |
| --- | --- | --- |
| Theater | | |
| Attribute | Tên biến | Ý nghĩa |
|  | id: int | Mã id của phòng chiếu |
|  | cinemtaId: int | Mã id của rạp chiếu |
|  | name: String | Tên phòng chiếu |
|  | rowCount: int | Số lượng hàng ghế của phòng chiếu |
|  | colCount: int | Số lượng cột ghế của phòng chiếu |
|  | status: int | Trạng thái phòng chiếu |
| Operation | Tên hàm | Ý nghĩa |
|  | getId(): int | Lấy mã id của phòng chiếu |
|  | setId(int id): void | Gán mã id mới cho phòng chiếu |
|  | getCinemaID(): int | Lấy mã id của rạp chiếu |
|  | setCinemaID(int cinemaID): void | Gán mã id rạp chiếu mới cho phòng chiếu |
|  | getName(): String | Lấy tên phòng chiếu |
|  | setName(String name): void | Gán tên mới cho phòng chiếu |
|  | getStatus(): int | Lấy trạng thái của phòng chiếu |
|  | setStatus(int status): void | Gán trạng thái mới cho phòng chiếu |
|  | getRowCount(): int | Lấy số lượng hàng ghế của phòng chiếu |
|  | setRowCount(int rowCount): void | Gán số lượng hàng ghế mới cho phòng chiếu |
|  | getColCount(): int | Lấy số lượng cột ghế của phòng chiếu |
|  | setColCount(int colCount): void | Gán số lượng cột ghế mới cho phòng chiếu |
|  | insert(Theater theater): int | Thêm 1 phòng chiếu mới vào database |
|  | update(Theater theater): int | Chỉnh sửa 1 phòng chiếu trong database |
|  | delete(int id): int | Xóa 1 phòng chiếu trong database |
|  | select(String condt): List<Theater> | Lấy danh sách phòng chiếu phù hợp với điều kiện, nếu condt rỗng thì lấy danh sách tất cả phòng chiếu có trong database |
|  | createSeats(Theater theater): int | Tạo các ghế mới có trong phòng chiếu |
|  | deleteSeats(Theater theater): int | Xóa các ghế có trong phòng chiếu |

|  |  |  |
| --- | --- | --- |
| TheaterController | | |
| Attribute | Tên biến | Ý nghĩa |
|  | theaterM: Theater | Model của phòng chiếu |
| Operation | Tên hàm | Ý nghĩa |
|  | loadTheaters(): ArrayList<Theater> | Báo model lấy danh sách tất cả phòng chiếu và lưu dưới dạng ArrayList |
|  | add(Theater theater): int | Báo model thêm 1 phòng chiếu mới vào database |
|  | modify(Theater theater) | Báo model chỉnh sửa 1 phòng chiếu trong database |
|  | delete(int ID) | Báo model xóa 1 phòng chiếu trong database |
|  | select(String condt): List<Theater> | Báo model lấy danh sách phòng chiếu phù hợp với điều kiện trong database |
|  | selectAll():List<Theater> | Báo model lấy danh sách tất cả phòng chiếu và lưu dưới dạng List |
|  | createSeats(Theater theater): int | Báo model tạo các ghế mới cho phòng chiếu |

|  |  |  |
| --- | --- | --- |
| Warehouse | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã nhà kho |
| -CinemaId: integer | Nhà kho thuộc cinema |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã kho |
| +setId(id: integer): void | Gán mã kho |
| +getCinemaId(): integer | Lấy mã cinema |
| +setCinemaId(cinemaid: integer): void | Gán mã cinema |
| +select(): List<Warehouse> | Lấy danh sách nhà kho |
| +insert(warehouse: Warehouse): void | Ghi nhà kho vào database |
| +update(warehouse: Warehouse): void | Cập nhập kho trong database |
| +delete(id: integer): void | Xóa kho khỏi database |
| +getNextID(): int | Lấy mã kho tiếp theo |

|  |  |  |
| --- | --- | --- |
| Cinema | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã cinema |
| -Name: String | Tên cinema |
| -Location: String | Địa điểm |
| -Status: integer | Tình trạng |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã cinema |
| +setId(id: integer): void | Gán mã cinema |
| +getName(): String | Lấy tên cinema |
| +setName(name: String): void | Gán tên cinema |
| +getLocation(): String | Lấy địa điểm |
| +setLocation(location: String): void | Gán địa điểm |
| +getStatus(): integer | Lấy tình trạng |
| +setStatus(status: integer): void | Gán mã tình trạng |
| +select(): List< Cinema > | Lấy danh sách cinema |
| +insert(cinema: Cinema): void | Ghi cinema vào database |
| +update(cinema: Cinema): void | Cập nhập cinema trong database |
| +delete(id: integer): void | Xóa cinema khỏi database |
| +getNextID(): int | Lấy mã cinema tiếp theo |
| +IDtoName(id: integer): String | Lấy tên đơn vị tính tương ứng với mã |

|  |  |  |
| --- | --- | --- |
| Unit | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã đơn vị tính |
| -Name: String | Tên đơn vị tính |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã đơn vị tính |
| +setId(id: integer): void | Gán mã đơn vị tính |
| +getName(): String | Lấy tên đơn vị tính |
| +setName(name: String): void | Gán tên đơn vị tính |
| +select(): List<Unit> | Lấy danh sách đơn vị tính |
| +IDtoName(id: integer): String | Lấy tên đơn vị tính tương ứng với mã |

|  |  |  |
| --- | --- | --- |
| Storage | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã sản phẩm |
| -WarehouseId: integer | Mã nhà kho |
| -productName: String | Tên sản phẩm |
| -Quantity: integer | Số lượng |
| -Unit: integer | Mã đơn vị tính |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã sản phẩm |
| +setId(id: integer): void | Gán mã sản phẩm |
| +getWarehoused(): integer | Lấy mã nhà kho |
| +setWarehouse (warehouse: integer): void | Gán mã nhà kho |
| +getproductName (): String | Lấy tên sản phẩm |
| +setproductName (name: String): void | Gán tên sản phẩm |
| +getQuantity (): integer | Lấy số lượng |
| +setQuantity (quantity: integer): void | Gán số lượng |
| +getUnit (): integer | Lấy mã đơn vị tính |
| +setUnit (id: integer): void | Gán mã đơn vị tính |
| +select(): List<Storage> | Lấy danh sách hàng trong kho |
| +insert(storage: Storage): void | Ghi hàng trong kho vào database |
| +update(storage: Storage): void | Cập nhập hàng trong kho trong database |
| +delete(id: integer): void | Xóa hàng trong kho khỏi database |
| +getNextID(): int | Lấy mã kho tiếp theo |
| +add(storage: Storage, quantity: integer): void | Cập nhập số lượng tồn |
| +selectProduct(): List<Storage> | Lấy danh sách sản phẩm tồn kho |

|  |  |  |
| --- | --- | --- |
| Status | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã tình trạng |
| -Name: String | Tên tình trạng |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã tình trạng |
| +setId(id: integer): void | Gán mã tình trạng |
| +getName(): String | Lấy tên tình trạng |
| +setName(name: String): void | Gán tên tình trạng |
| +select(): List<Status> | Lấy danh sách tình trạng |

|  |  |  |
| --- | --- | --- |
| SellReport | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã báo cáo thu |
| -CinemaId: integer | Mã rạp |
| -productTotal: integer | Tổng thu sản phẩm |
| -ticketTotal: integer | Tổng thu vé |
| -reportDay: Date | Ngày báo cáo |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã báo cáo thu |
| +setId(id: integer): void | Gán mã báo cáo thu |
| +getCinemaId (): integer | Lấy mã rạp |
| +setCinemaId (cinema: integer): void | Gán mã rạp |
| +getproductTotal (): integer | Lấy tổng thu sản phẩm |
| +setproductTotal (total: integer): void | Gán tổng thu sản phẩm |
| +getticketTotal (): integer | Lấy tổng thu vé |
| +setticketTotal (total: integer): void | Gán tổng thu vé |
| +getreportDay (): Date | Lấy ngày báo cáo |
| +setreportDay (date: Date): void | Gán ngày báo cáo |
| +select(): List< SellReport > | Lấy danh báo cáo thu |
| +insert(sellReport: SellReport): void | Ghi báo cáo thu vào database |
| +update(sellReport: SellReport): void | Cập nhập báo cáo thu trong database |
| +delete(id: integer): void | Xóa báo cáo thu khỏi database |
| +getcurDayID(date: Date): integer | Lấy mã báo cáo thu ngày hiện tại |
| +selectAPeriod(begin: Date, end: Date): List<SellReport> | Lấy danh sách báo cáo thu trong một khoảng thời gian |

|  |  |  |
| --- | --- | --- |
| FoodDetail | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã chi tiết hóa đơn sản phẩm |
| -BillId: integer | Mã hóa đơn sản phẩm |
| -productId: integer | Mã sản phẩm |
| -Price: integer | Đơn giá |
| -Quantity: integer | Số lượng |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã chi tiết hóa đơn sản phẩm |
| +setId(id: integer): void | Gán mã chi tiết hóa đơn sản phẩm |
| +getBillId (): integer | Lấy mã hóa đơn sản phẩm |
| +setBillId (bill: integer): void | Gán mã hóa đơn sản phẩm |
| +getproductId (): integer | Lấy mã sản phẩm |
| +setproductId (total: integer): void | Gán mã sản phẩm |
| +getPrice (): integer | Lấy đơn giá |
| +setPrice (price: integer): void | Gán đơn giá |
| +getQuantity (): integer | Lấy số lượng |
| +setQuantity (quan: integer): void | Gán số lượng |
| +select(): List< FoodDetail > | Lấy danh chi tiết hóa đơn sản phẩm |
| +insert(fooddetail: FoodDetail): void | Ghi chi tiết hóa đơn sản phẩm vào database |
| +update(fooddetail: FoodDetail): void | Cập chi tiết hóa đơn sản phẩm trong database |
| +delete(id: integer): void | Xóa chi tiết hóa đơn sản phẩm khỏi database |

|  |  |  |
| --- | --- | --- |
| BillFood | | |
| Attribute | Tên biến | Ý nghĩa |
| -Id: integer | Mã hóa đơn sản phẩm |
| -CinemaId: integer | Mã rạp |
| -productTotal: integer | Tổng tiền |
| -Time: Date | Ngày xuất |
| Operation | Tên hàm | Ý nghĩa |
| +getId(): integer | Lấy mã hóa đơn sản phẩm |
| +setId(id: integer): void | Gán mã hóa đơn sản phẩm |
| +getCinemaId (): integer | Lấy mã rạp |
| +setCinemaId (cinema: integer): void | Gán mã rạp |
| +getTotal (): integer | Lấy tổng tiền |
| +setTotal (total: integer): void | Gán tổng tiền |
| +getTime (): Date | Lấy ngày xuất |
| +setTime (date: Date): void | Gán ngày xuất |
| +select(): List< BillFood > | Lấy danh hóa đơn sản phẩm |
| +insert(billfood: BillFood): void | Ghi hóa đơn sản phẩm vào database |
| +update(billfood: BillFood): void | Cập nhập hóa đơn sản phẩm trong database |
| +delete(id: integer): void | Xóa hóa đơn sản phẩm khỏi database |

|  |  |  |
| --- | --- | --- |
| WarehouseController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadWarehouses(): List <Warehouse> | Lấy danh sách kho |
| +insertWarehouses(id: integer, cinemaid: integer): void | Thêm kho |
| +updateWarehouses(id: integer, cinemaid: integer): void | Cập nhập kho |
| +deleteWarehouses(id: integer): void | Xóa kho |
| +getNextID(): integer | Lấy ID tiếp theo |

|  |  |  |
| --- | --- | --- |
| CinemaController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadCinema (): List <Cinema> | Lấy danh sách cinema |
| +insertCinema (id: integer, name: String, location: String, status: integer): void | Thêm cinema |
| +updateCinema (id: integer, name: String, location: String, status: integer): void | Cập nhập cinema |
| +deleteCinema (id: integer): void | Xóa cinema |
| +getNextID(): integer | Lấy ID tiếp theo |

|  |  |  |
| --- | --- | --- |
| UnitController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadUnit (): List <Unit> | Lấy đơn vị tính |
| +IDtoName(id: integer): String | Lấy tên của ID tương ứng |

|  |  |  |
| --- | --- | --- |
| StorageController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadStorage (): List <Storage> | Lấy danh sách hàng trong |
| +insertStorage (id: integer, warehouseId: integer, Name: String, Quantity: integer, Unit: integer): void | Thêm danh sách hàng trong kho |
| +updateStorage (id: integer, warehouseId: integer, Name: String, Quantity: integer, Unit: integer): void | Cập nhập hàng trong kho |
| +deleteStorage (id: integer): void | Xóa hàng kho |
| +IDtoName(id: integer): String | Lấy tên của ID tương ứng |
| +AddQuantity(id: integer, warehouseId: integer, Quantity: integer) | Thêm hàng vào kho |

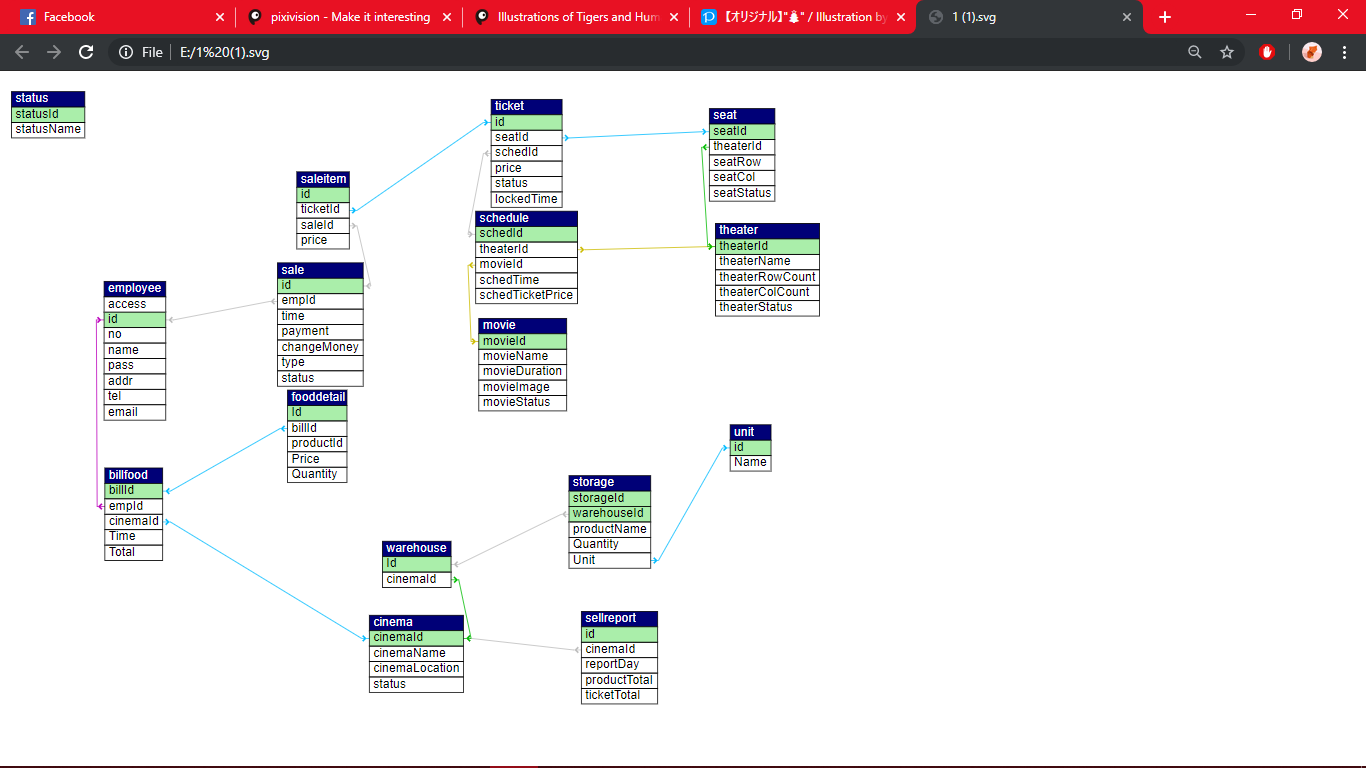
|  |  |  |
| --- | --- | --- |
| UnitController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadUnit (): List <Unit> | Lấy đơn vị tính |

|  |  |  |
| --- | --- | --- |
| SellReportController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadSellReport (begin: Date, end: Date): List <SellReport> | Lấy danh báo cáo thu trong một khoảng thời gian |
| +insertSellReport (id: integer, cinemaid: integer, time: Date, productTotal: integer, ticketTotal: integer): void | Thêm báo cáo thu |
| +updateSellReport (id: integer, productTotal: integer, ticketTotal: integer): void | Cập nhập báo cáo thu |
| +deleteSellReport (id: integer): void | Xóa báo cáo thu |
| +getCurDateID(date: Date): integer | Lấy ID ngày hiện tại |
| +getNextID(): integer | Lấy ID tiếp theo |

|  |  |  |
| --- | --- | --- |
| FoodDetailController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadFoodDetail (): List <FoodDetail> | Lấy danh sách chi tiết hóa đơn sản phẩm |
| +insertFoodDetail (id: integer, billId: integer, productid: integer, price: integer, quantity: integer): void | Thêm chi tiết hóa đơn sản phẩm |
| +updateFoodDetail (id: integer, billId: integer, productid: integer, price: integer, quantity: integer): void | Cập nhập chi tiết hóa đơn sản phẩm |
| +deleteFoodDetail (id: integer): void | Xóa chi tiết hóa đơn sản phẩm |
| +getNextID(): integer | Lấy ID tiếp theo |

|  |  |  |
| --- | --- | --- |
| BillFoodController | | |
| Operation | Tên hàm | Ý nghĩa |
| +loadBillFood (): List <BillFood> | Lấy danh sách hóa đơn sản phẩm |
| +insertBillFood (id: integer, cinemaId: integer, date: Date, total: integer): void | Thêm chi tiết sản phẩm |
| +updateBillFood (id: integer, cinemaId: integer, date: Date, total: integer): void | Cập nhập hóa đơn sản phẩm |
| +deleteBillFood (id: integer): void | Xóa hóa đơn sản phẩm |
| +getNextID(): integer | Lấy ID tiếp theo |

1. ÁNH XẠ DỮ LIỆU QUAN HỆ TỪ MÔ HÌNH HƯỚNG ĐỐI TƯỢNG
2. 1. MÔ HÌNH QUAN HỆ



* 1. MÔ TẢ CÁC KIỂU DỮ LIỆU
* TABLE billfood

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | billId | int(11) | Khác null | auto\_increment | mã Id của bill |
| 2 | empId | int(11) | Khác null |  | mã Id của nhân viên |
| 3 | cinemaId | int(11) | Khác null |  | mã Id của cụm rạp |
| 4 | Time | datetime | Khác null |  | thời gian lập bill |
| 5 | Total | int(11) | Khác null |  | tổng tiền của bill |

* TABLE cinema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | cinemaId | int(11) | Khác null |  | mã Id của cụm rạp |
| 2 | cinemaName | varchar(100) | Khác null |  | tên cụm rạp |
| 3 | cinemaLocation | varchar(100) | Khác null |  | vị trí của cụm rạp |
| 4 | status | int(11) | Khác null |  | trạng thái cụm rạp |

* TABLE employee

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | access | int(11) | Khác null |  | quyền truy cập của nhân viên |
| 2 | id | int(11) | Khác null | auto\_increment | mã Id của nhân viên |
| 3 | no | int(11) |  |  | số thứ tự nhân viên |
| 4 | name | varchar(100) | Khác null |  | tên nhân viên |
| 5 | pass | varchar(100) |  |  | password của nhân viên |
| 6 | addr | varchar(200) |  |  | địa chỉ của nhân viên |
| 7 | tel | varchar(20) |  |  | số điện thoại của nhân viên |
| 8 | email | varchar(100) |  |  | email của nhân viên |

* TABLE fooddetail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | Id | int(11) | Khác null | auto\_increment | mã Id chi tiết hóa đơn |
| 2 | billId | int(11) | Khác null |  | mã Id của bill |
| 3 | productid | int(11) | Khác null |  | mã Id của sản phẩm |
| 4 | Price | int(11) | Khác null |  | giá của sản phẩm |
| 5 | Quantity | int(11) | Khác null |  | số lượng sản phẩm |

* TABLE movie

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | movieId | int(11) | Khác null | auto\_increment | mã Id của phim |
| 2 | movieName | varchar(200) | Khác null |  | tên phim |
| 3 | movieDuration | int(11) | Khác null |  | thời lượng phim |
| 4 | movieImage | longblob |  |  | hình ảnh của phim |
| 5 | movieStatus | int(11) | Khác null |  | trạng thái của phim |

* TABLE sale

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | id | int(11) | Khác null | auto\_increment | mã Id của bill bán vé |
| 2 | empId | int(11) | Khác null |  | mã Id của nhân viên |
| 3 | time | datetime | Khác null |  | thời gian lập bill bán vé |
| 4 | payment | int(11) | Khác null |  | số tiền khách thanh toán |
| 5 | changeMoney | int(11) | Khác null |  | số tiền thối lại cho khách |
| 6 | type | int(11) |  |  | kiểu thanh toán |
| 7 | status | int(11) | Khác null |  | trạng thái bill bán vé |

* TABLE saleitem

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | id | int(11) | Khác null | auto\_increment | mã id của chi tiết bill bán vé |
| 2 | ticketId | int(11) | Khác null |  | mã id của vé |
| 3 | saleId | int(11) | Khác null |  | mã id của bill bán vé |
| 4 | price | int(11) | Khác null |  | số tiền của vé |

* TABLE schedule

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | schedId | int(11) | Khác null | auto\_increment | mã id của thời gian chiếu |
| 2 | theaterId | int(11) | Khác null |  | mã id của rạp |
| 3 | movieId | int(11) | Khác null |  | mã id của phim |
| 4 | schedTime | datetime | Khác null |  | thời gian bắt đầu chiếu |
| 5 | schedTicketPrice | int(11) | Khác null |  | giá tiền của thời gian chiếu |

* TABLE seat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | seatId | int(11) | Khác null | auto\_increment | mã id của ghế |
| 2 | theaterId | int(11) | Khác null |  | mã id của rạp |
| 3 | seatRow | int(11) | Khác null |  | vị trí theo hàng của ghế |
| 4 | seatCol | int(11) | Khác null |  | vị trí theo cột của ghế |
| 5 | seatStatus | int(11) | Khác null |  | trạng thái của ghế |

* TABLE sellreport

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | id | int(11) | Khác null | auto\_increment | mã id của bản báo cáo |
| 2 | cinemaId | int(11) | Khác null |  | mã id của cụm rạp |
| 3 | reportDay | date | Khác null |  | ngày lập bản báo cáo |
| 4 | productTotal | int(11) | Khác null |  | tổng số tiền bán bắp nước |
| 5 | ticketTotal | int(11) | Khác null |  | tổng số tiền bán vé |

* TABLE status

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | statusId | int(11) | Khác null |  | mã id của trạng thái |
| 2 | statusName | varchar(100) | Khác null |  | tên trạng thái |

* TABLE storage

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | storageId | int(11) | Khác null |  | mã id của sản phẩm |
| 2 | warehouseId | int(11) | Khác null |  | mã id của nhà kho |
| 3 | productName | varchar(100) | Khác null |  | tên sản phẩm |
| 4 | Quantity | int(11) | Khác null |  | số lượng sản phẩm tồn |
| 5 | Unit | int(11) | Khác null |  | đơn vị tính |

* TABLE theater

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | theaterId | int(11) | Khác null | auto\_increment | mã id của rạp |
| 2 | theaterName | varchar(100) | Khác null |  | tên rạp |
| 3 | theaterRowCount | int(11) | Khác null |  | số hàng ghế của rạp |
| 4 | theaterColCount | int(11) | Khác null |  | số cột ghế của rạp |
| 5 | theaterStatus | int(11) | Khác null |  | trạng thái của rạp |

* TABLE ticket

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | id | int(11) | Khác null | auto\_increment | mã id của vé |
| 2 | seatId | int(11) | Khác null |  | mã id của ghế |
| 3 | schedId | int(11) | Khác null |  | mã id của thời gian chiếu |
| 4 | price | int(11) | Khác null |  | giá vé |
| 5 | status | int(11) | Khác null |  | trạng thái của vé |
| 6 | lockedTime | datetime |  |  | thời gian lập vé |

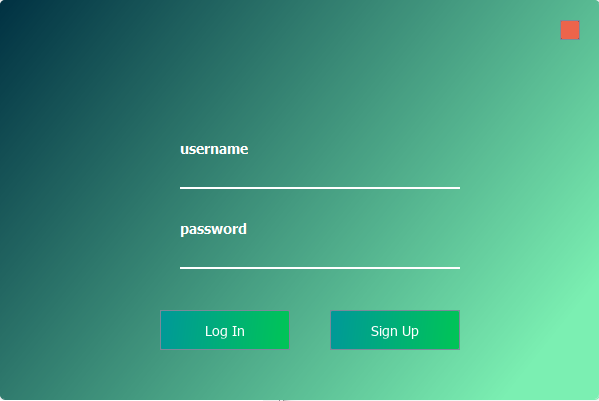
* TABLE unit

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | id | int(11) | Khác null | auto\_increment | mã id của đơn vị tính |
| 2 | Name | varchar(100) | Khác null |  | tên đơn vị tính |

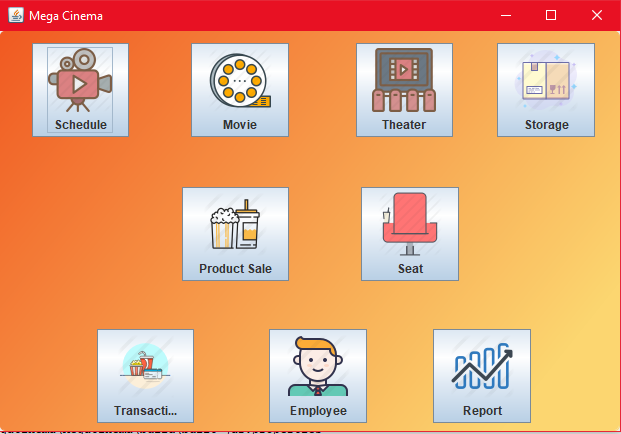
* TABLE warehouse

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu | Ràng buộc | Giá trị khởi động | Ý nghĩa / Ghi chú |
| 1 | id | int(11) | Khác null | auto\_increment | mã id của nhà kho |
| 2 | cinemaId | int(11) | Khác null |  | mã id của cụm rạp |

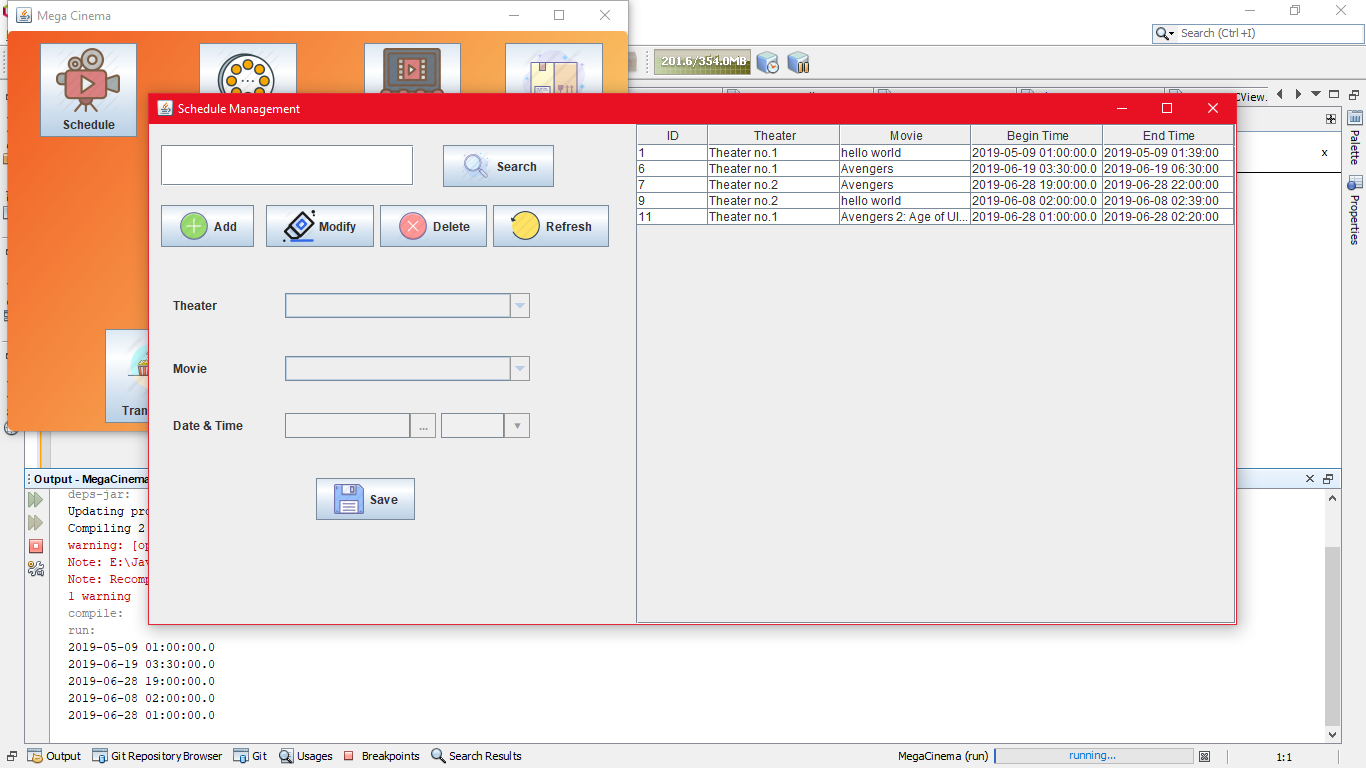
1. THIẾT KẾ GIAO DIỆN
   1. ĐĂNG NHẬP



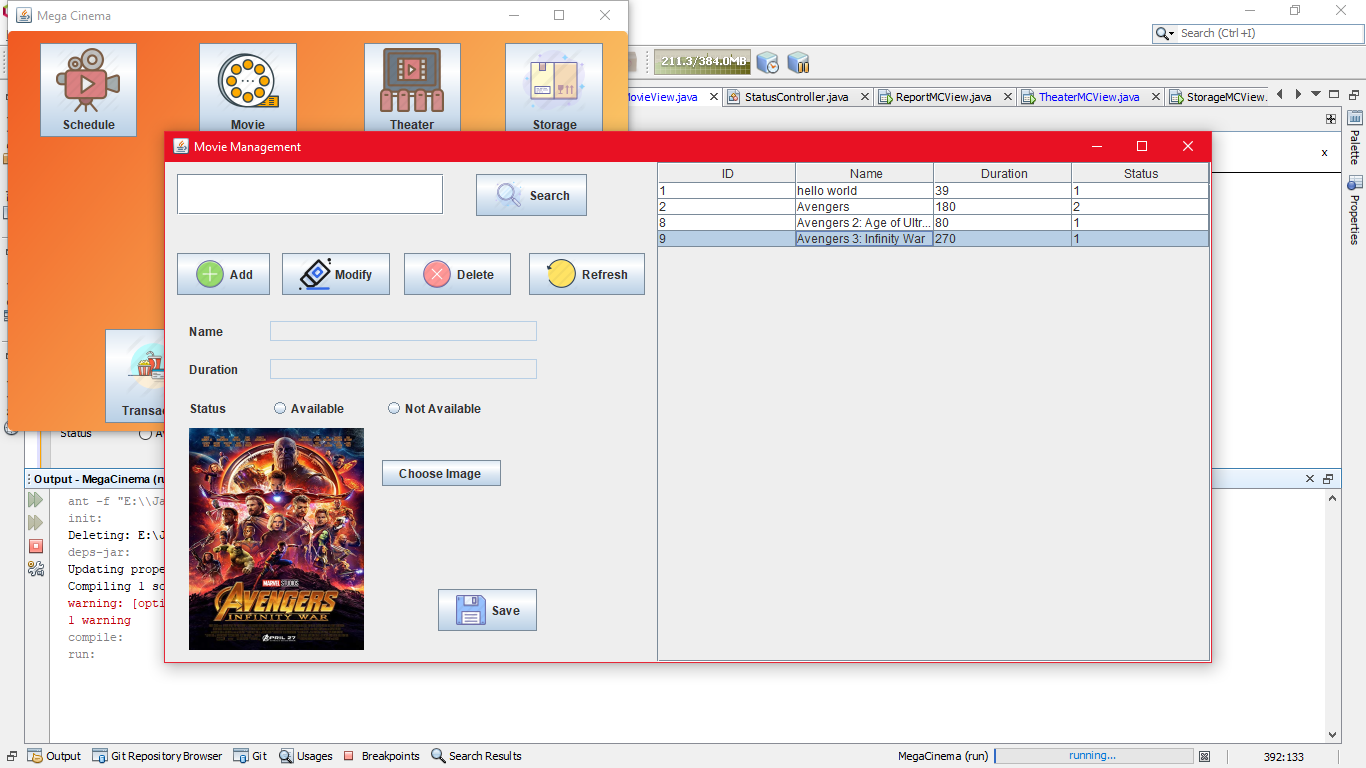
* 1. GIAO DIỆN CHÍNH



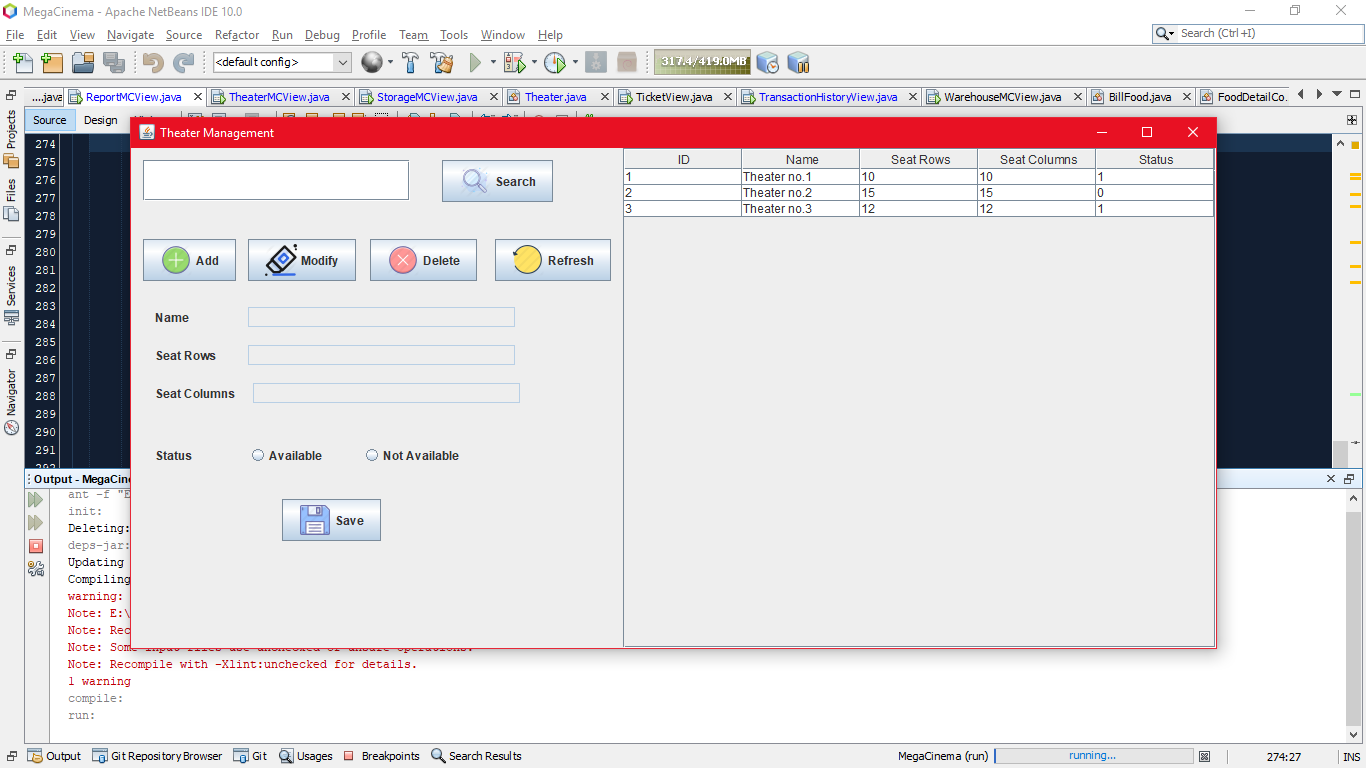
* 1. QUẢN LÝ LỊCH CHIẾU



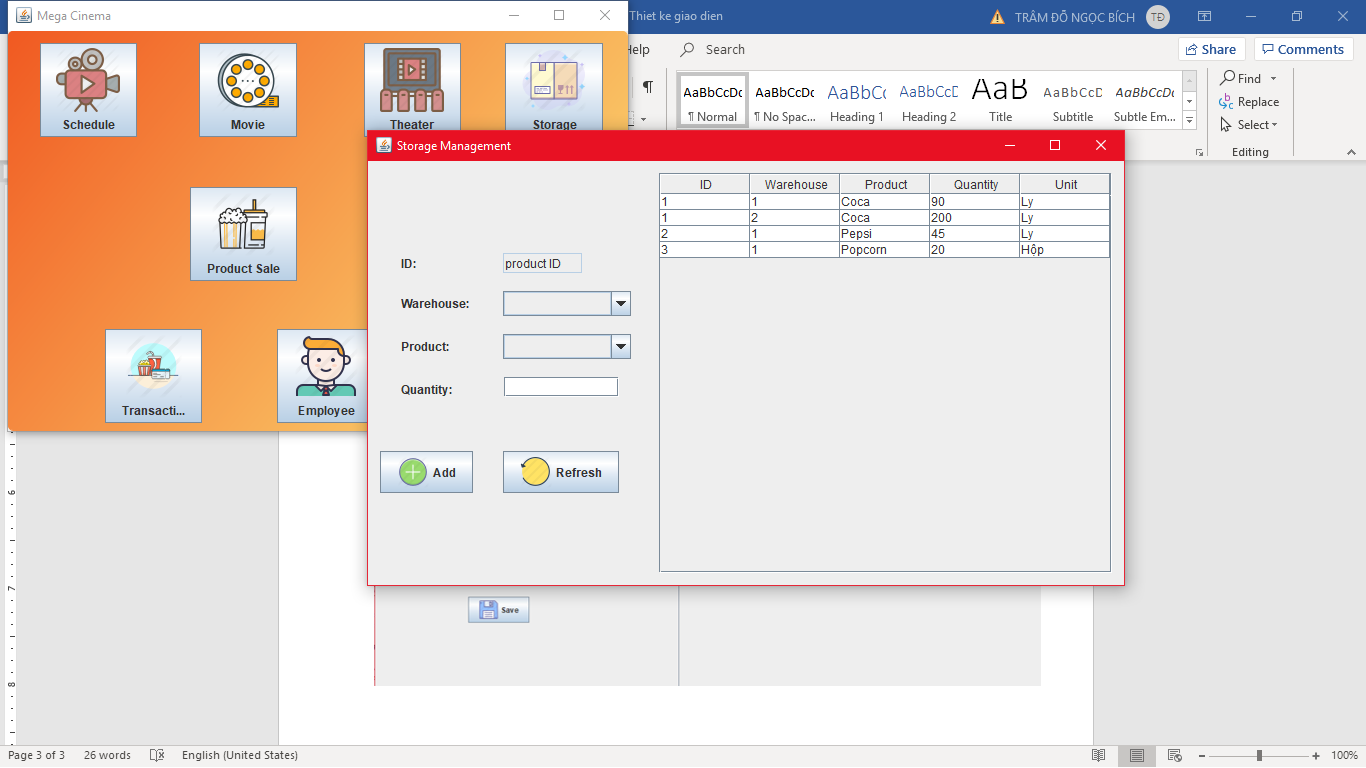
* 1. QUẢN LÝ PHIM



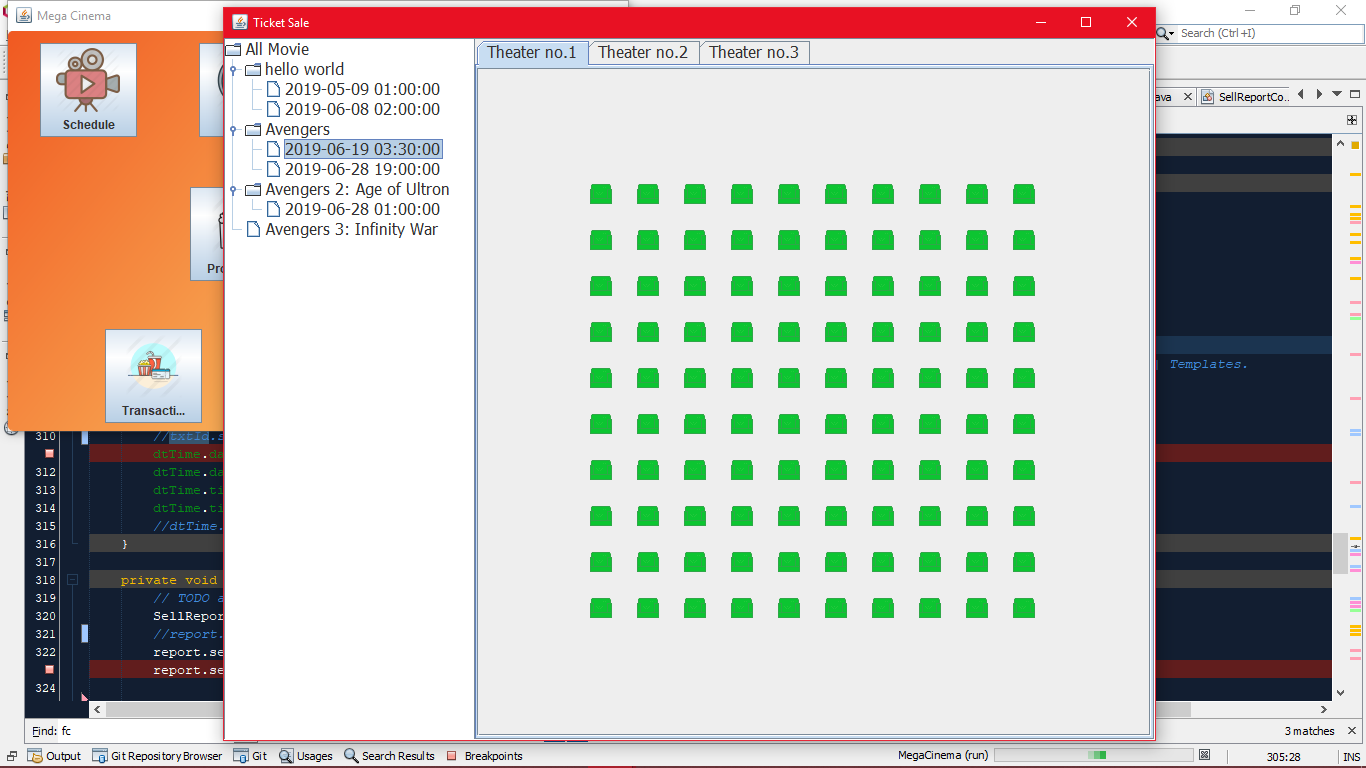
* 1. QUẢN LÝ PHÒNG CHIẾU



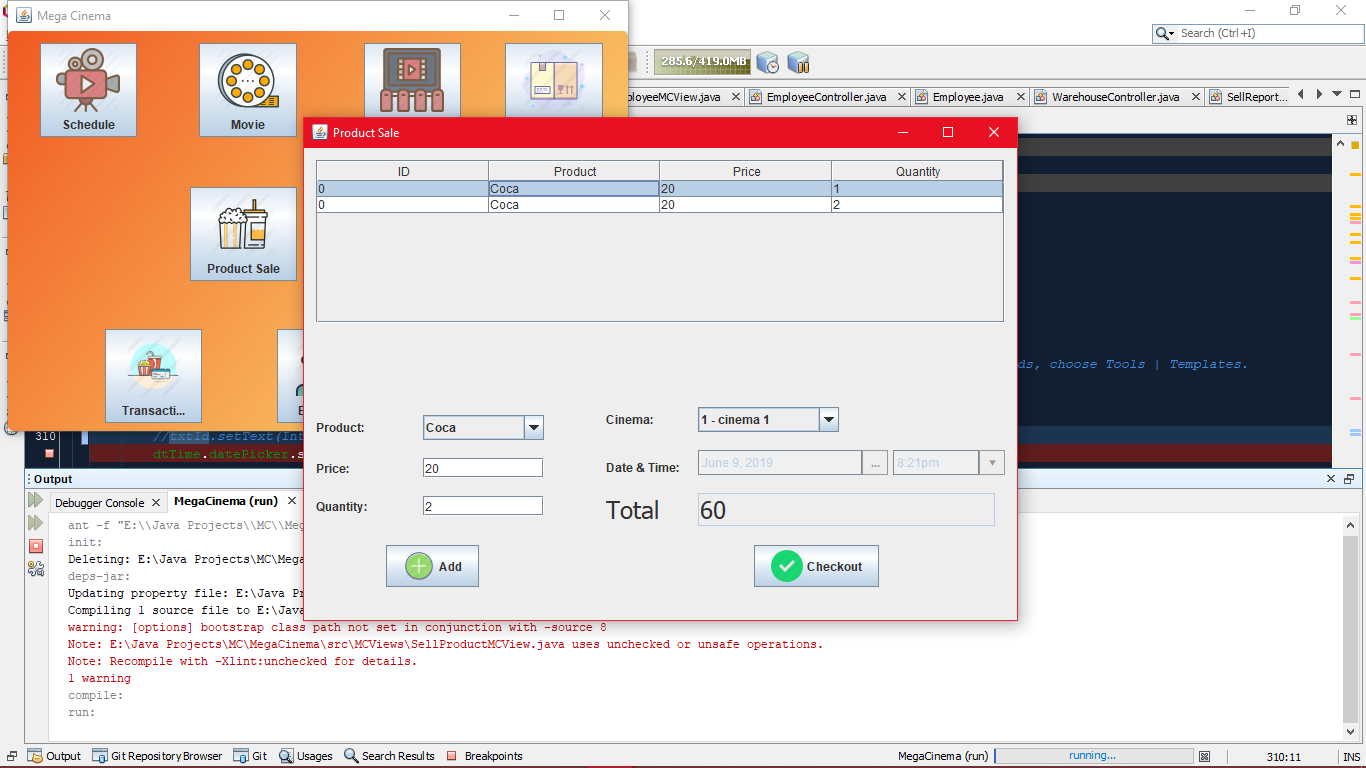
* 1. QUẢN LÝ KHO



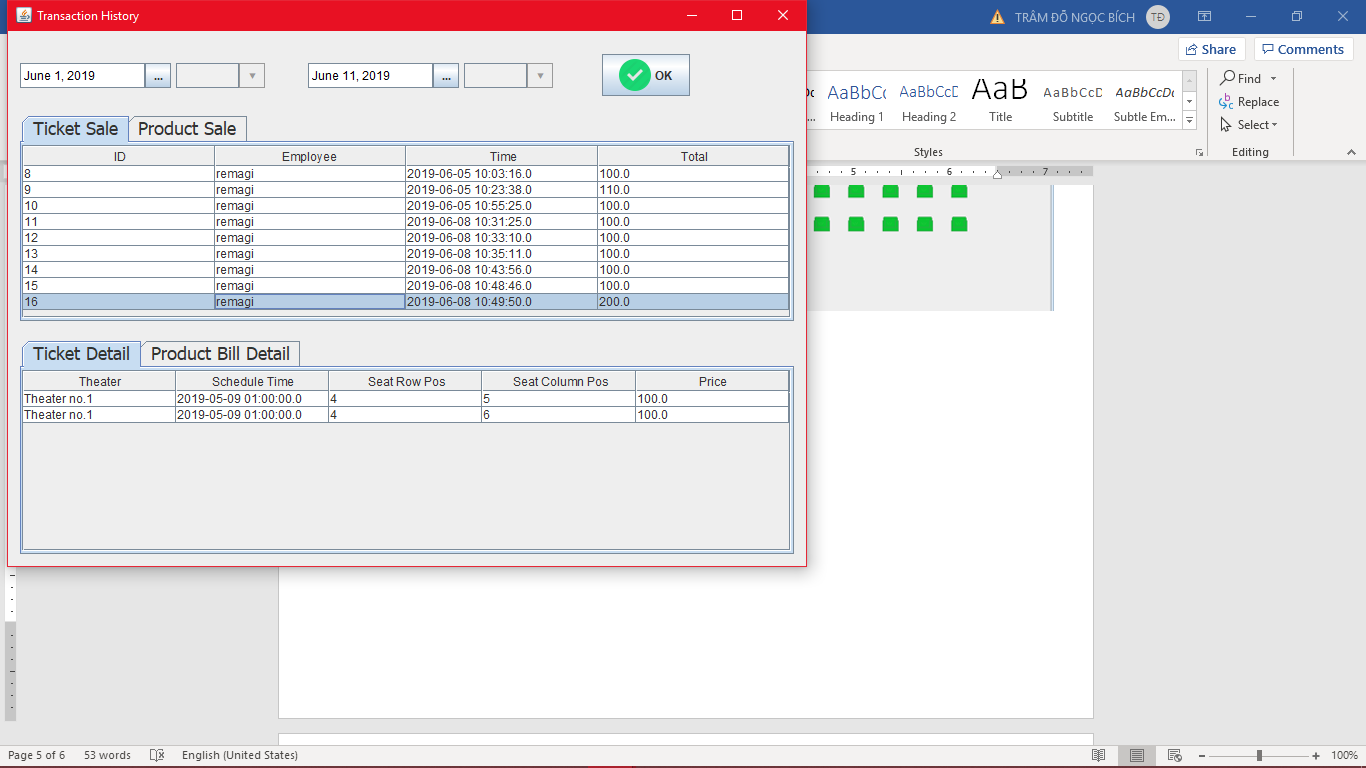
* 1. BÁN VÉ XEM PHIM

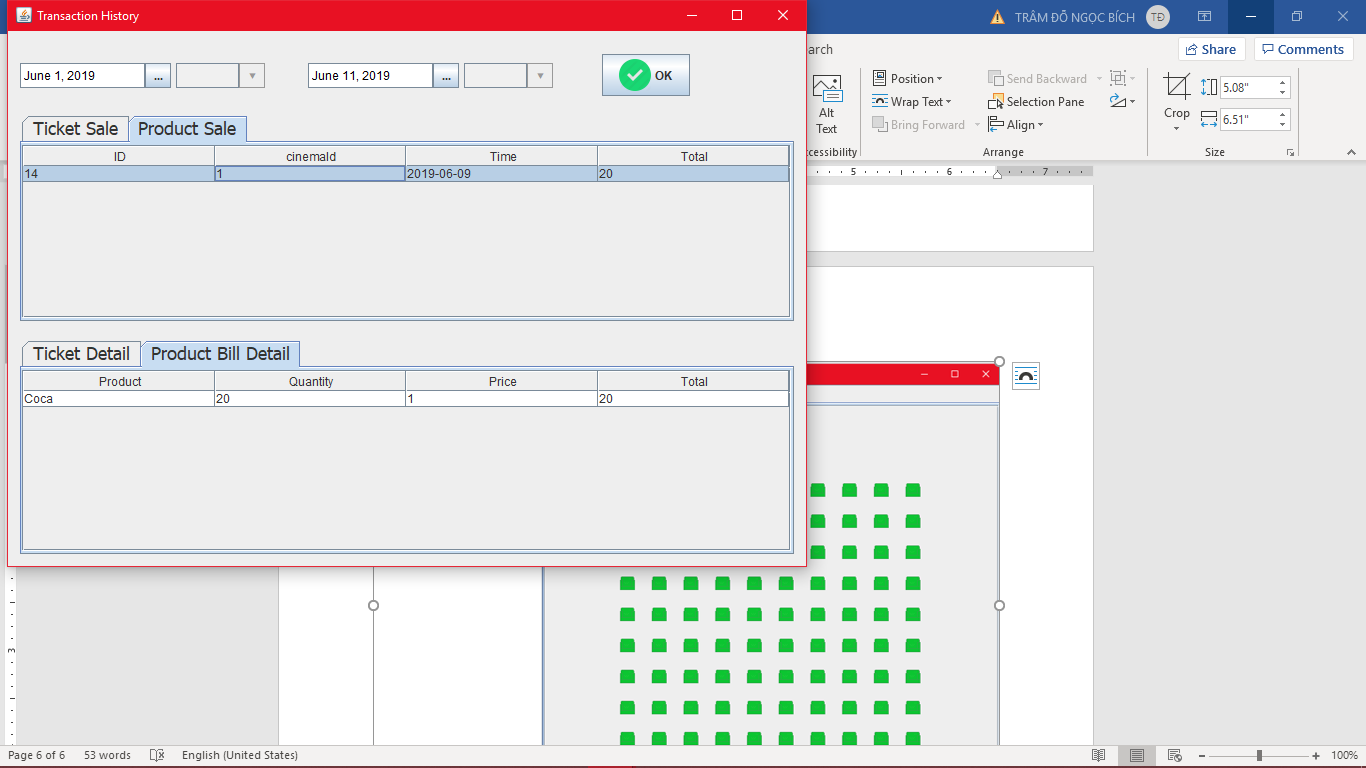


* 1. BÁN BẮP NƯỚC VÀ CÁC SẢN PHẨM KHÁC

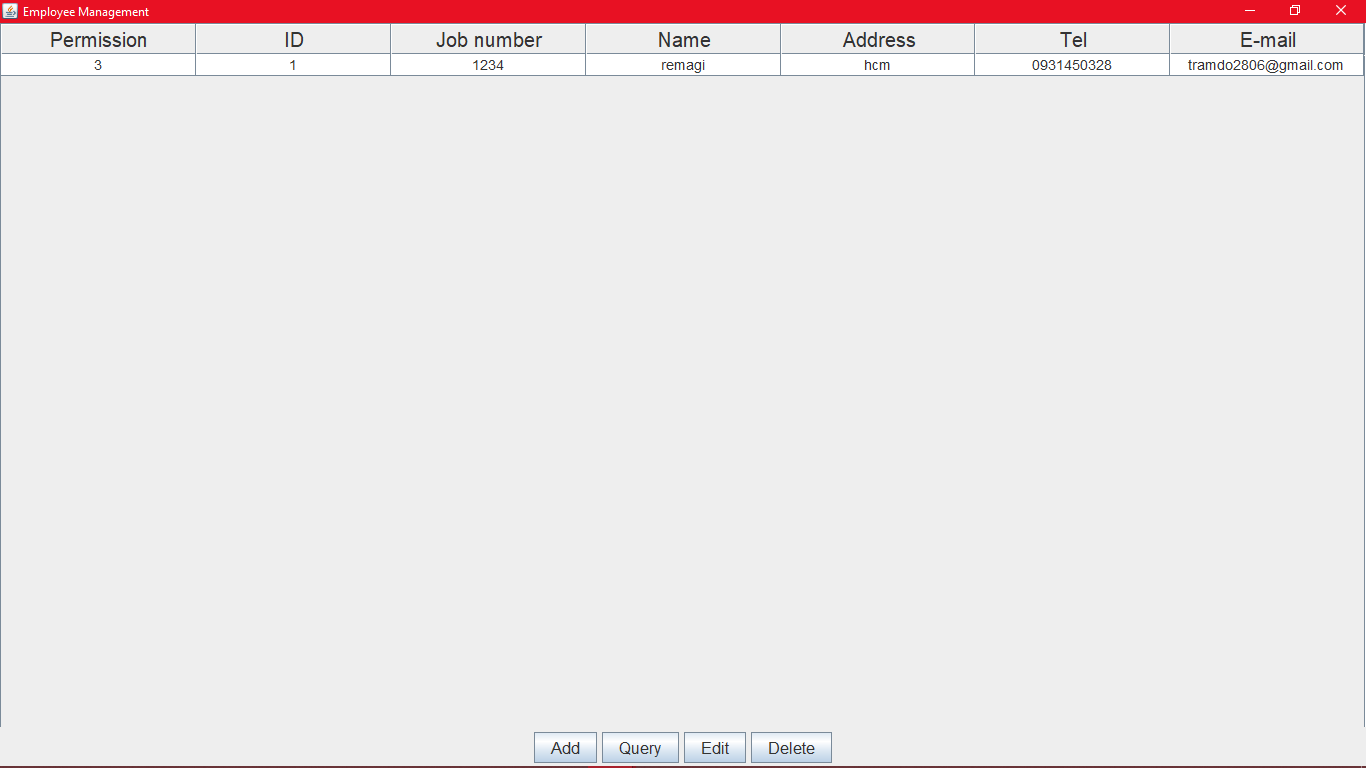


* 1. XEM LỊCH SỬ GIAO DỊCH

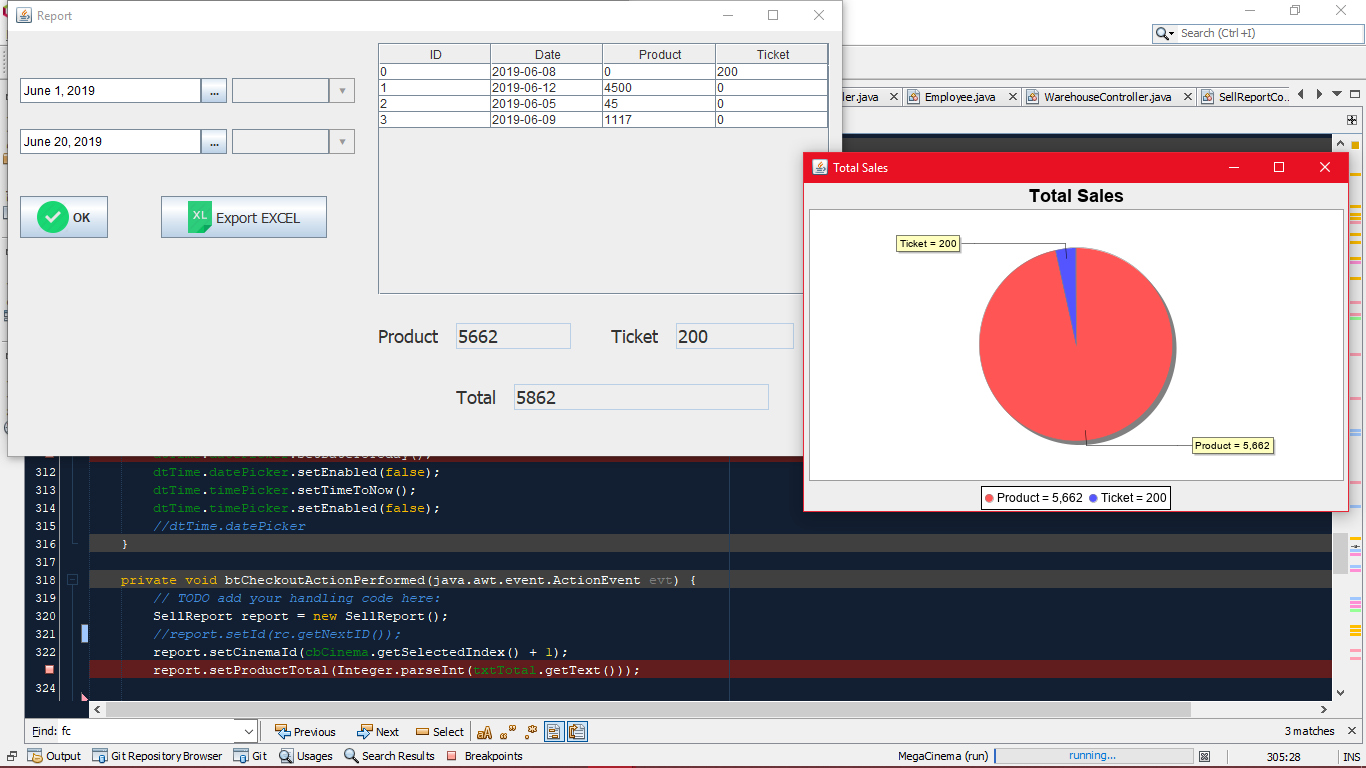




* 1. QUẢN LÝ NHÂN VIÊN

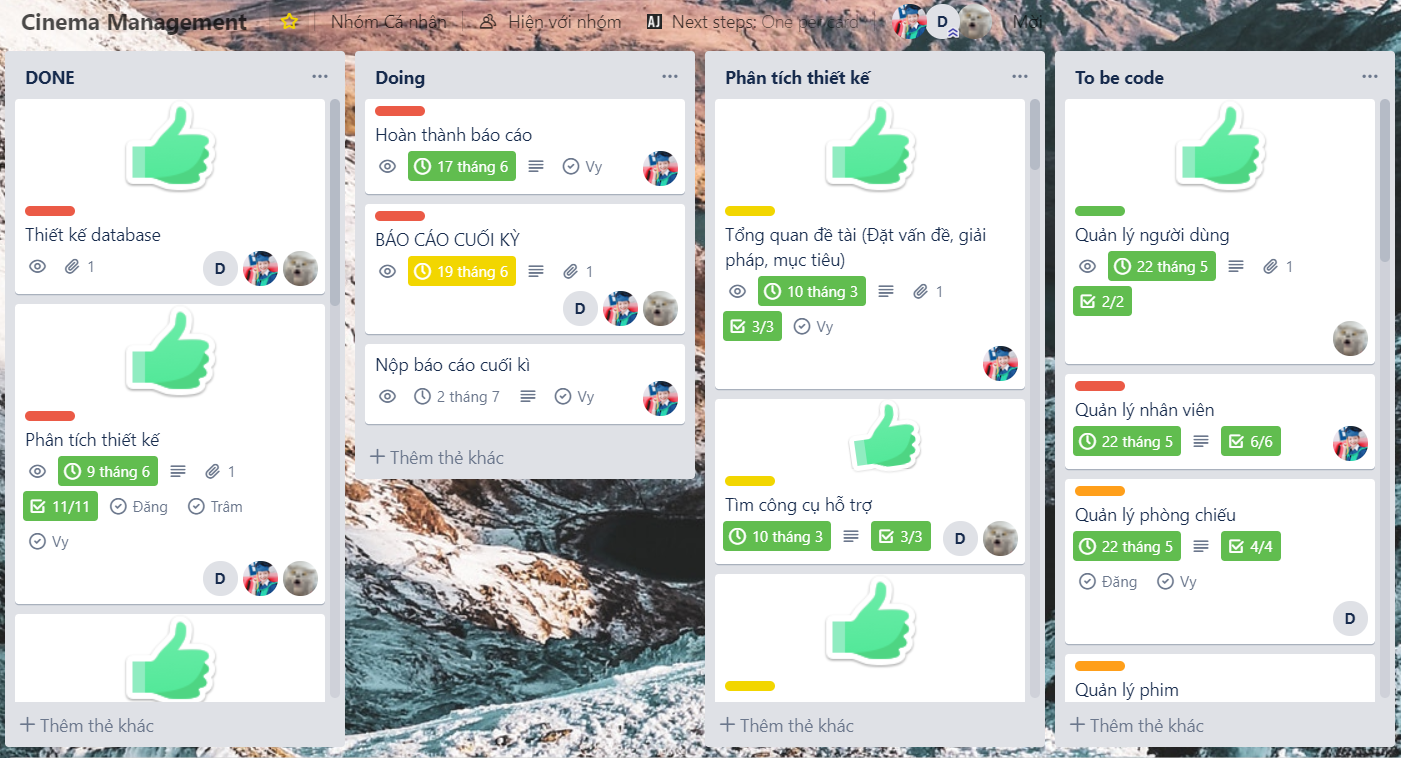


* 1. BÁO CÁO VÀ BIỂU ĐỒ



1. KẾT LUẬN:

* Bảng phân chia công việc nhóm : <https://trello.com/b/Ri9naGdU/cinema-management>



* Môi trường phát triển và triển khai ứng dụng :
* **Môi trường phát triển ứng dụng :**
* Hệ điều hành : Window 10
* Hệ quản trị CSDL : mySQL
* Công cụ xây dựng ứng dụng : NetBeans IDE 8.2
* **Môi trường triển khai ứng dụng :**
* Hệ điều hành : đa nền tảng
* Cần cài đặt mySQL
* **Kết quả đạt được :**

Quản lý rạp chiếu phim sử dụng hệ thống thông tin không phải là một hướng đi mới. Tuy nhiên, hiện tại vẫn còn rất nhiều cơ sở kinh doanh rạp chiếu phim nhỏ chỉ quản lý bằng cách thủ công. Nhận thấy đây là một đề tài hay và thiết thực có thể ứng dụng rộng rãi. Chính vì vậy nhóm tác giả đã quyết định chọn đề tài nhằm tạo ra hệ thống quản lý rạp chiếu phim trên nền tảng APP giúp ích cho việc quản lý đơn giản hơn, tiện ích hơn và giảm tải lượng công việc thủ công..

* **Hướng phát triển :**

Nhóm tác giả nhận định còn rất nhiều việc cần phải làm để hoàn thiện hệ thống quản lý rạp chiếu phim. Chính vì vậy nhóm đặt ra mục tiêu để phát triển đề tài trong tương lai như sau:

-­‐ Chức năng quản lý khách hàng, phân loại khách hàng và ưu đãi cho khách hàng theo từng hạng mục, có quà tặng sinh nhật cho khách hàng, ,..vv.

-­‐ Phân loại nhiều loại ghế, giường nằm,... để khách hàng có nhiều sự lựa chọn khi đến xem phim.

-­‐ Xây dựng kênh liên lạc giữa các khách với hệ thống rạp phim, bằng việc kết nối với các mạng xã hội thông dụng.

-­‐ Mở rộng các sản phẩm tại rạp phim : ly nước hình dạng các chủ đề hot theo phim, ống hút và ly bảo vệ môi trường, .. vv

--- Tạo bảng chấm công, quản lý giờ làm và lịch nghỉ phép cho nhân viên