

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**



**BÀI TẬP LỚN**

**Học kỳ: 01 Năm học: 2024-2025**

**Lớp Tín chỉ: PTUTBD.03.K13.13.LH.C04.1\_LT**

**HỌC PHẦN: PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG**

**CHỦ ĐỀ 1: ỨNG DỤNG QUẢN LÝ**

**ĐỀ TÀI 5: XÂY DỰNG ỨNG DỤNG QUẢN LÝ BẾN XE KHÁCH TRÊN NỀN TẢNG ANDROID SỬ DỤNG ANDROID STUDIO**

**Giảng viên giảng dạy: ThS. Trần Mạnh Đông**

**Danh sách sinh viên thực hiện:**

STT	Mã sinh viên	Sinh viên thực hiện	Lớp hành chính
1	20222398	Đỗ Nhật Minh	DCCNTT 13.10.13
2	20222494	Phạm Tuấn Anh	DCCNTT 13.10.13
3	20222492	Hoàng Tiến Huy	DCCNTT 13.10.13
4	20222384	Nguyễn Công Minh	DCCNTT 13.10.13

**Bắc Ninh, năm 2024**

**PHIẾU CHẤM THI BÀI TẬP LỚN**

**CHỦ ĐỀ 1: ỨNG DỤNG QUẢN LÝ**

**ĐỀ TÀI 5: XÂY DỰNG ỨNG DỤNG QUẢN LÝ BẾN XE KHÁCH TRÊN NỀN  
TẢNG ANDROID SỬ DỤNG ANDROID STUDIO**

TT	TIÊU CHÍ ĐÁNH GIÁ	THANG ĐIỂM	Đỗ Nhật Minh	Phạm Tuấn Anh	Hoàng Tiến Huy	Nguyễn Công Minh
<b>I</b>	<b>Chức năng</b>	<b>3</b>				
1	Ứng dụng hoạt động ổn định, không gặp lỗi nghiêm trọng.	0,5				
2	Các chức năng chính theo chủ đề được triển khai đầy đủ và chính xác.	0,5				
3	Giao diện thân thiện, dễ sử dụng, hỗ trợ người dùng thực hiện các tác vụ một cách thuận tiện.	0,5				
4	Triển khai thêm các tính năng phụ trợ hoặc cải tiến để tăng tính hữu ích và trải nghiệm người dùng.	0,5				
5	Ứng dụng có khả năng xử lý dữ liệu lớn hoặc các tác vụ phức tạp.	0,5				
6	Tích hợp các công nghệ mới hoặc các API bên ngoài để mở rộng chức năng.	0,5				
<b>II</b>	<b>Chất lượng code</b>	<b>2</b>				

TT	TIÊU CHÍ ĐÁNH GIÁ	THANG ĐIỂM	Đỗ Nhật Minh	Phạm Tuấn Anh	Hoàng Tiến Huy	Nguyễn Công Minh
1	Tổ chức code tổ chức thành các module, package rõ ràng, dễ quản lý và bảo trì. Sử dụng tên biến, tên hàm, tên lớp có ý nghĩa và tuân thủ quy tắc đặt tên	0,5				
2	Tính đúng đắn và hiệu quả	0,5				
3	Sử dụng các kỹ thuật lập trình hướng đối tượng như kế thừa, đa hình, lớp trừu tượng, interface để tăng tính tái sử dụng và mở rộng của code.	0,5				
4	Thiết kế code linh hoạt, dễ dàng thêm hoặc thay đổi chức năng trong tương lai.	0,5				
<b>II</b>	<b>Thiết kế giao diện</b>	<b>2</b>				
1	Giao diện đẹp, hài hòa, sử dụng màu sắc và hình ảnh phù hợp. Tuân thủ các nguyên tắc thiết kế Material Design (Android).	0,5				
2	Bố cục rõ ràng, các thành phần được sắp xếp hợp lý, dễ nhìn.	0,5				

TT	TIÊU CHÍ ĐÁNH GIÁ	THANG ĐIỂM	Đỗ Nhật Minh	Phạm Tuấn Anh	Hoàng Tiến Huy	Nguyễn Công Minh
3	Điều hướng giữa các màn hình dễ dàng và trực quan.	0,5				
4	Cung cấp phản hồi cho người dùng khi thực hiện các thao tác.	0,5				
III	<b>Tính sáng tạo và khác biệt</b>	<b>1</b>				
1	<b>Ý tưởng độc đáo</b>	0,5				
2	<b>Cải tiến công nghệ</b>	0,5				
IV	<b>Báo cáo và trình bày</b>	<b>2</b>				
1	Báo cáo đầy đủ, chi tiết, trình bày rõ ràng, mạch lạc, tuân thủ định dạng yêu cầu.	0,5				
2	Mô tả đầy đủ các yêu cầu, phân tích, thiết kế, cài đặt, kiểm thử và đánh giá.	0,5				
3	Trình bày tự tin, rõ ràng, truyền đạt tốt nội dung và ý tưởng của dự án.	0,5				
4	Demo ứng dụng mượt mà, thể hiện rõ các tính năng chính.	0,5				
<b>TỔNG ĐIỂM BẢNG SỐ:</b>		10				

ĐIỂM BÀI TẬP LỚN

TT	Mã sinh viên	Sinh viên thực hiện	Điểm BTL	
			Điểm số	Điểm chữ
1	20222398	Đỗ Nhật Minh		
2	20222494	Phạm Tuấn Anh		
3	20222492	Hoàng Tiến Huy		
4	20222384	Nguyễn Công Minh		

Cán bộ chấm thi 1

(Ký và ghi rõ họ tên)

Cán bộ chấm thi 2

(Ký và ghi rõ họ tên)

## LỜI NÓI ĐẦU

Nhóm chúng em xin kính chào quý thầy cô bộ phận khảo thí cùng với thầy Trần Mạnh Đông- người trực tiếp hướng dẫn em trong bài báo cáo này. Lời đầu tiên cho phép nhóm chúng em gửi đến thầy, cô lời chúc sức khỏe. Nhóm chúng em là nhóm 1 lớp PTUTBD.03.K13.13.LH.C04.1\_LT, nhóm chúng em xin được phép trình bày bài báo cáo ngay sau đây.

Đầu tiên em xin có đôi lời về thầy, thầy là người rất nhiệt tình trong giảng dạy và cũng rất quan tâm chúng em trong tất cả mọi việc. Mỗi khi lớp có bạn nghỉ thầy đều hỏi các bạn trong lớp về lý do bạn nghỉ cũng như lo lắng cho bạn rằng sẽ không theo kịp chúng em. Thầy là người cầm tay chỉ việc từng bạn trong lớp, ai không biết phần nào thầy cũng rất ân cần dạy.

Tiếp theo đây em xin được trình bày về bài báo cáo của nhóm em. Theo sự tìm hiểu của em, hiện nay vấn đề quản lý đối với tất cả công ty, các bến xe nói chung và với bến xe Mỹ Đình luôn là vấn đề nhức nhối với các ban lãnh đạo. Vì nhận thấy được sự bất cập này, nhóm chúng em có ý tưởng tạo ra 1 app trên android để giúp việc quản lý bến xe cho ban quản lý và việc đặt vé của khách hàng sẽ trở nên dễ dàng hơn. Rất mong app của chúng em nhận được sự chấp thuận từ thầy cô!

Bài báo cáo của chúng em sẽ nói lên thực tế của bến xe khách hiện nay, cũng như bến xe Hà Nội nói chung và đưa ra giải pháp khắc phục nó. Sau khi tìm hiểu thực trạng của bến xe, chúng em đã nảy ra ý tưởng tạo ứng dụng đặt vé và quản lý bến xe.

Theo thống kê sơ bộ ban đầu, trung bình một ngày bến đón hơn 11.000 lượt khách mỗi ngày và hơn 4.000.000 lượt khách mỗi năm với hơn 106 đơn vị vận tải phục vụ. Với lượng khách đông như vậy, sự quản lý bến xe cũng trở nên rất khó khăn và là vấn đề nan giải của đội ngũ quản lý. Vì vậy chúng em muốn góp chút sức mọn để việc quản lý trở nên dễ dàng hơn.

Ý tưởng ban đầu của chúng em là tạo ra ứng dụng đặt vé cho người dùng bao gồm các chức năng cơ bản như xem lịch trình xe, tra cứu vé đã đặt, đặt vé và thanh toán. Với ứng dụng quản lý bao gồm quản lý nhân sự (Employee management), quản lý xe (Car management), quản lý khách (Customer management), quản lý dịch vụ (Service management),... Hiện nay ứng dụng của chúng em đã bao gồm những chức năng nhất định. Với ứng dụng đặt vé dành cho khách hàng, khách hàng có thể tra cứu lịch trình xe, tra cứu nhà xe, tìm kiếm vé, đặt vé và thanh toán trực tiếp hoặc online. Với ứng dụng quản lý dành cho bến xe, ngoài các chức năng như thêm, sửa, xóa dữ liệu ứng dụng còn có thể tìm kiếm dữ liệu. Dữ liệu sẽ được lưu trữ trong hệ tổng cơ sở dữ liệu MySQL. Bài báo cáo của chúng em còn bao gồm các sơ đồ, biểu đồ cần thiết để xây dựng bài như biểu đồ phân rã chức năng, sơ đồ lớp, biểu đồ trình tự, biểu đồ hoạt động,...

Trên đây chúng em đã trình bày ý tưởng cũng như nội dung bài báo cáo của chúng em, chúng em xin cam đoan số liệu và kết quả nghiên cứu trong đồ án này là trung thực và chưa hề được sử dụng để bảo vệ một học vị nào. Mọi sự giúp đỡ cho việc

thực hiện bài báo cáo này đã được cảm ơn và các thông tin trích dẫn trong đề án đã được chỉ rõ nguồn gốc rõ ràng và được phép công bố.

Chúng em xin chân thành cảm ơn thầy cô đã đọc và góp ý cho chúng em!

Hà Nội, ngày ... tháng ... năm 2024

Nhóm sinh viên thực hiện

(Ký và ghi rõ họ tên)

# MỤC LỤC

DANH MỤC TỪ VIẾT TẮT .....	10
DANH MỤC HÌNH VẼ .....	11
DANH MỤC BẢNG BIỂU .....	13
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT .....	14
1.1 Tổng quan về ứng dụng cho thiết bị di động .....	14
1.1.1 Giới thiệu chung về xu hướng phát triển ứng dụng cho thiết bị di động .....	14
1.1.2 Tổng quan về các hệ điều hành thiết bị di động, các công cụ lập trình .....	15
1.1.3 Hệ điều hành Android .....	19
1.1.4 Công cụ lập trình Android Studio .....	21
1.2. Phát triển ứng dụng trên nền tảng Android sử dụng Java .....	22
1.2.1. Cấu trúc ứng dụng Android .....	22
1.2.2. Các thành phần phát triển ứng dụng Android .....	25
CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....	28
2.1 Giới thiệu bài toán .....	28
2.1.1 Giới thiệu về đề tài .....	28
2.1.2 Yêu cầu bài toán .....	28
2.2 Sơ đồ chức năng của phần mềm .....	34
2.3 Mô hình hoá yêu cầu .....	39
2.3.1 Mô hình UC tổng quát .....	39
2.3.2 Mô hình UC hoạt động của khách hàng .....	40
2.3.3 Mô hình UC quản lý nhân viên .....	43
2.3.4 Mô hình UC quản lý xe .....	44
2.3.5 Mô hình UC quản lý đặt vé .....	45
2.3.6 Mô hình UC quản lý khách hàng .....	46
2.3.7 Mô hình UC quản lý dịch vụ .....	48
2.3.8 Mô hình UC quản lý cơ sở vật chất .....	49
2.3.9 Mô hình UC đăng nhập .....	50
2.3 Mô hình hoá khái niệm .....	51
2.3.1 Class Diagram .....	51
2.3.2 Activity Diagram .....	52
2.3.3 Sequence Diagram .....	61
2.4 Mô hình hoá kiến trúc hệ thống .....	66



2.5 Mô hình cơ sở dữ liệu .....	67
<b>CHƯƠNG 3: CÀI ĐẶT CHƯƠNG TRÌNH</b> .....	<b>71</b>
3.1 Tổ chức chương trình .....	71
3.2. Các chức năng chương trình .....	81
3.3. Giao diện chính của chương trình .....	83
3.3.1 Giao diện main.....	84
3.3.2 Giao diện trang chủ.....	85
3.3.3 Giao diện đăng ký khách hàng.....	87
3.3.4 Giao diện đăng nhập khách hàng .....	89
3.3.5 Giao diện đăng nhập quản trị viên .....	90
3.3.6 Giao diện đăng nhập nhân viên .....	92
3.3.7 Giao diện chọn chức năng nghiệp vụ cho nhân viên .....	93
3.3.8 Giao diện chọn chức năng nghiệp vụ cho quản trị viên .....	95
3.3.9 Giao diện quản lý xe .....	96
3.3.10 Giao diện quản lý nhân viên .....	98
3.3.11 Giao diện quản lý khách .....	100
3.3.12 Giao diện quản lý dịch vụ .....	103
3.3.13 Giao diện quản lý đặt vé .....	105
3.3.13 Giao diện quản lý cơ sở vật chất.....	107
<b>CHƯƠNG 4: KẾT LUẬN</b> .....	<b>110</b>
4.1 Kết quả đạt được .....	110
4.2 Hạn chế.....	110
4.3 Hướng phát triển .....	110
<b>TÀI LIỆU THAM KHẢO</b> .....	<b>111</b>
<b>PHỤ LỤC</b> .....	<b>112</b>

## DANH MỤC TỪ VIẾT TẮT

Từ viết tắt	Ý nghĩa
CSVC	CƠ SỞ VẬT CHẤT
UC	USECASE

## DANH MỤC HÌNH VẼ

Hình 1. 1 Xu hướng ứng dụng thiết bị di động giai đoạn 2014-2023 .....	14
Hình 1. 2 Các hệ điều hành hiện nay .....	16
Hình 1. 3 Công cụ lập trình Andoid Studio .....	17
Hình 1. 4 Công cụ lập trình Andromo .....	18
Hình 1. 5 Công cụ lập trình Appypie .....	19
Hình 1. 6 Công cụ lập trình Android Studio .....	21
Hình 1. 7 Tạo lập 1 Activity mới .....	23
Hình 1. 8 Tạo một tệp XML Drawable Resource .....	23
Hình 2. 1 Biểu đồ phân rã chức năng hệ thống .....	34
Hình 2. 2 Biểu đồ Usecase tổng quát .....	39
Hình 2. 3 Biểu đồ UC khách hàng .....	40
Hình 2. 4 UC quản lý nhân viên .....	43
Hình 2. 5 UC quản lý xe .....	44
Hình 2. 6 UC quản lý đặt vé .....	45
Hình 2. 7 UC quản lý khách hàng .....	46
Hình 2. 8 UC quản lý dịch vụ .....	48
Hình 2. 9 UC quản lý cơ sở vật chất .....	49
Hình 2. 10 UC đăng nhập .....	50
Hình 2. 11 Class Diagram .....	51
Hình 2. 12 Biểu đồ hoạt động đăng nhập .....	52
Hình 2. 13 Biểu đồ hoạt động đăng ký .....	53
Hình 2. 14 Biểu đồ hoạt động khách hàng đặt vé .....	54
Hình 2. 15 Biểu đồ hoạt động quản lý nhân viên .....	55
Hình 2. 16 Biểu đồ hoạt động quản lý xe .....	56
Hình 2. 17 Biểu đồ hoạt động quản lý khách .....	57
Hình 2. 18 Biểu đồ hoạt động quản lý dịch vụ .....	58
Hình 2. 19 Biểu đồ hoạt động quản lý đặt vé .....	59
Hình 2. 20 Biểu đồ hoạt động quản lý cơ sở vật chất .....	60
Hình 2. 21 Biểu đồ tuần tự đăng nhập .....	61
Hình 2. 22 Biểu đồ tuần tự đăng ký .....	61
Hình 2. 23 Biểu đồ tuần tự người dùng đặt vé .....	62
Hình 2. 24 Biểu đồ tuần tự quản lý khách .....	63
Hình 2. 25 Biểu đồ tuần tự quản lý khách .....	63
Hình 2. 26 Biểu đồ tuần tự quản lý xe .....	64
Hình 2. 27 Biểu đồ tuần tự quản lý dịch vụ .....	64
Hình 2. 28 Biểu đồ tuần tự quản lý đặt vé .....	65
Hình 2. 29 Biểu đồ tuần tự quản lý csvc .....	65
Hình 2. 30 Mô hình kiến trúc hệ thống .....	66
Hình 2. 31 Mô hình cơ sở dữ liệu .....	67
Hình 3. 1 Giao diện main .....	84
Hình 3. 2 Giao diện trang chủ .....	85
Hình 3. 3 Giao diện đăng ký tài khoản khách hàng .....	87

Hình 3. 4 Giao diện đăng nhập tài khoản khách hàng.....	89
Hình 3. 5 Giao diện đăng nhập tài khoản quản trị viên.....	90
Hình 3. 6 Giao diện đăng nhập tài khoản nhân viên .....	92
Hình 3. 7 Giao diện chọn nghiệp vụ nhân viên .....	93
Hình 3. 8 Giao diện chọn nghiệp vụ quản trị viên .....	95
Hình 3. 9 Giao diện quản lý nhân viên.....	96
Hình 3. 10 Giao diện quản lý nhân viên.....	98
Hình 3. 11 Giao diện quản lý khách.....	100
Hình 3. 11 Giao diện quản lý dịch vụ.....	103
Hình 3. 11 Giao diện quản lý đặt vé.....	105
Hình 3. 11 Giao diện quản lý cơ sở vật chất .....	107

## DANH MỤC BẢNG BIỂU

Bảng 2. 1 Mô tả biểu đồ phân rã chức năng .....	35
Bảng 2. 2 Đặc tả UC đăng ký .....	40
Bảng 2. 3 Đặc tả UC Đăng nhập .....	41
Bảng 2. 4 Đặc tả UC tìm kiếm .....	41
Bảng 2. 5 Đặc tả UC đặt vé .....	41
Bảng 2. 6 Đặc tả UC thanh toán .....	42
Bảng 2. 7 Đặc tả UC quản lý nhân viên .....	43
Bảng 2. 8 Đặc tả UC quản lý xe .....	44
Bảng 2. 9 Đặc tả UC quản lý đặt vé .....	45
Bảng 2. 10 Đặc tả UC quản lý khách hàng.....	46
Bảng 2. 11 Đặc tả UC quản lý dịch vụ .....	48
Bảng 2. 12 Đặc tả UC quản lý cơ sở vật chất.....	49
Bảng 2. 13 Đặc tả UC đăng nhập .....	50
Bảng 2. 14 Quản lý xe .....	67
Bảng 2. 15 Quản lý nhân viên .....	68
Bảng 2. 16 Quản lý khách .....	68
Bảng 2. 17 Quản lý đặt vé .....	69
Bảng 2. 18 Quản lý dịch vụ .....	69
Bảng 2. 19 Quản lý cơ sở vật chất.....	70
Bảng 2. 20 Quản lý tài khoản admin .....	70
Bảng 2. 21 Quản lý tài khoản nhân viên xe.....	70
Bảng 3. 1 Bảng đặc tả giao diện main .....	84
Bảng 3. 2 Bảng đặc tả giao diện trang chủ .....	86
Bảng 3. 3 Bảng đặc tả giao diện đăng ký tài khoản khách hàng .....	87
Bảng 3. 4 Bảng đặc tả giao diện đăng ký tài khoản khách hàng .....	89
Bảng 3. 5 Bảng đặc tả giao diện đăng ký tài khoản quản trị viên .....	91
Bảng 3. 6 Bảng đặc tả giao diện đăng ký tài khoản nhân viên.....	92
Bảng 3. 7 Bảng đặc tả giao diện đăng ký tài khoản nhân viên.....	94
Bảng 3. 8 Bảng đặc tả giao diện đăng ký tài khoản nhân viên.....	95
Bảng 3. 9 Bảng đặc tả giao diện quản lý xe .....	97
Bảng 3. 10 Bảng đặc tả giao diện quản lý nhân viên .....	99
Bảng 3. 11 Bảng đặc tả giao diện quản lý khách hàng .....	101
Bảng 3. 11 Bảng đặc tả giao diện quản lý dịch vụ .....	104
Bảng 3. 11 Bảng đặc tả giao diện quản lý đặt vé .....	106
Bảng 3. 11 Bảng đặc tả giao diện quản lý cơ sở vật chất .....	108

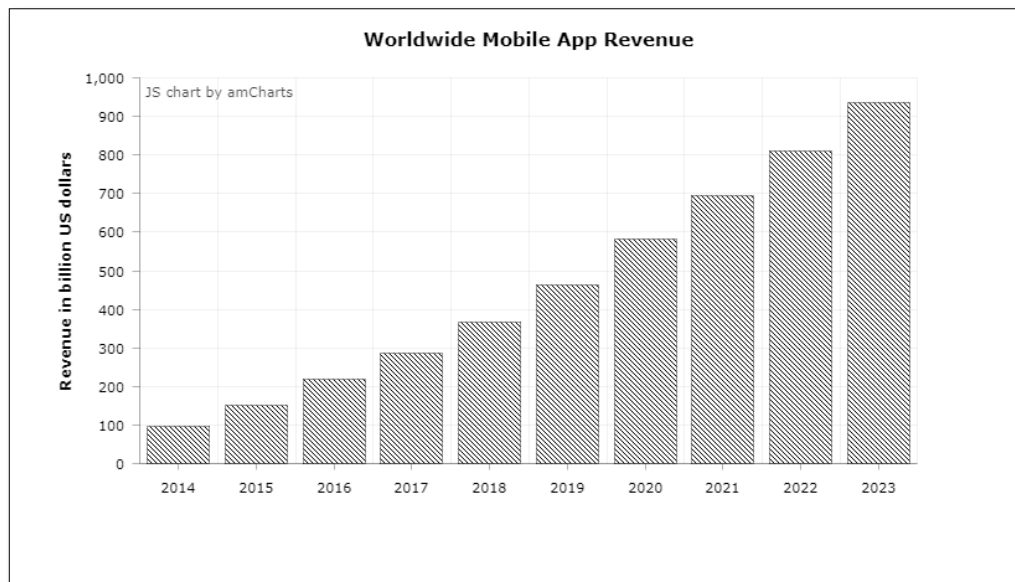
# CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

## 1.1 Tổng quan về ứng dụng cho thiết bị di động

### 1.1.1 Giới thiệu chung về xu hướng phát triển ứng dụng cho thiết bị di động

Hiện nay, dưới thời đại công nghệ số, hiện đại số, thiết bị di động và vật bất li thân của rất nhiều người không chỉ đối với giới trẻ mà còn cả người già, người trung niên hay kể cả trẻ em. Bởi vậy mà xu hướng mới sẽ phát sinh.

Theo số liệu thống kê mới nhất, ngành ứng dụng di động tiếp tục có mức tăng trưởng đáng kể. Tổng doanh thu trên thị trường ứng dụng được dự đoán sẽ đạt 475,90 tỷ USD vào năm 2022, với tốc độ tăng trưởng hàng năm (CAGR 2022-2027) là 8,58%, dẫn đến giá trị thị trường dự kiến là 755,5 tỷ USD vào năm 2027. Doanh thu mua hàng trong ứng dụng (IAP) trên thị trường ứng dụng dự kiến sẽ đạt 204,90 tỷ USD vào năm 2022, trong khi doanh thu ứng dụng phải trả phí dự kiến sẽ đạt 5,25 tỷ USD vào năm 2022. Doanh thu quảng cáo trên thị trường ứng dụng cũng dự kiến sẽ đạt 265,80 tỷ USD vào năm 2022. Số lượt tải xuống trên thị trường ứng dụng dự kiến sẽ đạt 235,30 tỷ lượt tải xuống vào năm 2022, với doanh thu trung bình trên mỗi lượt tải xuống dự kiến là 2,02 USD. Một so sánh toàn cầu cho thấy phần lớn doanh thu được tạo ra ở Trung Quốc, với dự kiến là 166,60 tỷ USD vào năm 2022. Những số liệu thống kê này nêu bật sự tăng trưởng liên tục và tiềm năng của ngành ứng dụng di động, mang lại cơ hội thú vị cho các doanh nghiệp và nhà phát triển để tạo ra các ứng dụng di động sáng tạo và có tác động.<sup>[8]</sup>



Hình 1. 1 Xu hướng ứng dụng thiết bị di động giai đoạn 2014-2023

Khi chúng ta bước vào năm 2024, ngành phát triển ứng dụng di động nhanh chóng phát triển với những xu hướng mới đang nổi lên. Một số xu hướng phát triển ứng dụng di động mới nhất cần chú ý vào năm 2024 bao gồm việc áp dụng mạng 5G, mạng này sẽ cung cấp tốc độ dữ liệu nhanh hơn và độ trễ thấp, cho phép các nhà phát triển xây dựng các ứng dụng tương tác và phản hồi nhanh hơn. Một xu hướng khác cần chú ý là việc sử dụng trí tuệ nhân tạo và học máy, điều này sẽ giúp các nhà phát triển tạo ra các ứng dụng thông minh hơn và cá nhân hóa hơn. Internet of Things (IoT) cũng sẽ tiếp tục đóng một vai trò quan trọng trong việc phát triển ứng dụng di động, với nhiều ứng dụng tích hợp với các thiết bị thông minh như thiết bị đeo và thiết bị gia dụng thông minh. Cuối cùng, nhu cầu về nền tảng phát triển low-code hoặc no-code sẽ tiếp tục tăng, giúp các chuyên gia không chuyên về kỹ thuật dễ dàng tạo và triển khai ứng dụng của riêng họ mà không cần phụ thuộc vào nhà phát triển.

### **1.1.2 Tổng quan về các hệ điều hành thiết bị di động, các công cụ lập trình**

#### **a, Hệ điều hành thiết bị di động**

- Hệ điều hành di động (Mobile Operating System - Mobile OS) là phần mềm quản lý phần cứng và tài nguyên phần mềm trên các thiết bị di động như điện thoại thông minh (smartphone), máy tính bảng (tablet), đồng hồ thông minh (smartwatch), và các thiết bị cầm tay khác. Hệ điều hành di động kết hợp các tính năng của một hệ điều hành cho máy tính cá nhân với các tính năng khác hữu ích cho việc sử dụng di động hoặc cầm tay; thường bao gồm hầu hết các chức năng được coi là cần thiết trong các hệ thống di động hiện đại như: màn hình cảm ứng, mạng thiết bị di động, Bluetooth, Wi-Fi Protected Access, Wi-Fi, Hệ thống Định vị Toàn cầu (GPS), máy ảnh số cho phép chụp ảnh và quay video, nhận dạng tiếng nói, thu âm, chơi nhạc, kết nối trường gần, và đèn hồng ngoại điều khiển từ xa. Đến cuối năm 2016, hơn 430 triệu điện thoại thông minh đã được bán với 81,7 % chạy nền tảng Android, 17,9 % chạy iOS, 0,3 % chạy Windows 10 Mobile (hiện các thiết bị chạy nền tảng này không còn được bán trên thị trường) và các hệ điều hành khác chiếm 0,1%.[1] Android còn phổ biến hơn so với hệ điều hành Windows, hệ điều hành phổ biến nhất trên máy tính để bàn. Lượng sử dụng điện thoại thông minh (thậm chí chưa bao gồm máy tính bảng) đã nhiều hơn cả lượng máy tính để bàn đang sử dụng (nhu cầu sử dụng máy tính tổng thể đã giảm xuống 44,9% trong quý I năm 2017).[9]



Hình 1. 2 Các hệ điều hành hiện nay

- Sự phát triển của hệ điều hành di động phản ánh sự phát triển của điện thoại di động, máy tính bảng, PDA và điện thoại thông minh. Giai đoạn đầu là cuộc chiến tranh giành thị phần của nhiều hãng sản xuất với rất nhiều nền tảng khác nhau. Nhưng với sự ra đời của iOS và Android vào năm 2007, cuộc chiến dần ngã ngũ với sự thống trị tuyệt đối của Android và iOS trên các điện thoại di động, máy tính bảng, đồng hồ thông minh và TV thông minh. Windows 10 Mobile và BlackBerry 10 là những nền tảng cạnh tranh cuối cùng cũng tuyên bố ngừng phát triển. Do iOS là độc quyền bởi Apple, các nhà sản xuất di động khác buộc phải lựa chọn Android. Đến quý 1 năm 2018, thị phần Android là 85.9%, 14.1% còn lại là iOS.<sup>[9]</sup>

- Hiện nay hệ điều hành di động phổ biến là android được phát triển bởi Google với nền tảng là Linux-based với ngôn ngữ lập trình đa dạng như Java, Kotlin, C++,... Và được phát hành trên cửa hàng ứng dụng Google Play Store. Với các tính năng về mã nguồn mở, đa dạng thiết bị, hệ sinh thái phong phú với hàng triệu ứng dụng và dịch vụ từ Google như Gmail, Google Maps, YouTube.

- Hệ điều hành iOS: Nền tảng iOS được tạo ra bởi Apple. Vì vậy nếu bạn phát triển một ứng dụng trên iOS thì đồng nghĩa nó sẽ hoạt động trên toàn bộ hệ sinh thái của Apple như iPhone và iPad. Để tạo ứng dụng di động trên hệ điều hành iOS, bạn cần có tài khoản dành cho nhà phát triển Apple. Đồng thời bạn cũng sẽ cần cài đặt Xcode IDE trên máy tính Mac. Nếu bạn cài đặt trên máy tính hệ Windows thì có thể nó sẽ không thể tạo được cũng như tính năng gỡ lỗi không chính xác được. Công cụ Xcode có các tính năng trình soạn thảo mã, trình mô phỏng, trình gỡ lỗi và SDK. Nó được trang bị hầu hết mọi thứ cần để lập trình ứng dụng cho tất cả các thiết bị của Apple. Có hai ngôn ngữ lập trình mobile app dành riêng để phát triển iOS gồm Objective-C và Swift .

- Đa nền tảng: Đa nền tảng tức là một phần mềm hay chương trình máy tính có khả năng hoạt động trên nhiều nền tảng. Ví dụ như một app nào đó có thể hoạt động được trên cả hệ điều hành Android và iOS. Để lập trình mobile app đa nền tảng, bạn sẽ cần học thêm một số công cụ đa nền tảng phổ biến. Đây là những framework mà qua đó



một ứng dụng di động có thể chạy trên nhiều nền tảng như Android, iOS và Windows. Các nhà phát triển chỉ cần viết mã một lần và có thể chạy nó ở bất kỳ đâu cho bất kỳ nền tảng nào họ muốn.

+ Xamari: Công cụ này được phát hành bởi Microsoft. Vì vậy để sử dụng được nó bạn cần bỏ ra một khoảng chi phí để có thể sử dụng được. Về mặt kỹ thuật, Xamarin không phải là một ngôn ngữ. Đó là một nền tảng phát triển mã nguồn mở cho các ứng dụng iOS, Android và Windows. Đây là một nền tảng .NET sử dụng C # làm ngôn ngữ cốt lõi của nó.

+ React Native: React Native sử dụng JavaScript để giao tiếp với các chức năng được tạo sẵn có nguồn gốc từ khung của họ. Điều này cho phép bạn thao tác trên giao diện người dùng, thu thập dữ liệu và truy xuất dữ liệu để bạn có thể chia sẻ cho người dùng. Nhìn chung, điều này có nghĩa là bạn đang phụ thuộc rất nhiều vào JavaScript để có thể thao tác. Một số thương hiệu lâu đời đã sử dụng Framework này để phát triển ứng dụng dành cho thiết bị di động của họ; chúng bao gồm Walmart, Uber và Skype

+ Appcelerator: Giống như React Native, Appcelerator cũng sử dụng JavaScript để giao tiếp với các chức năng với một riêng. Appcelerator Titanium giúp bạn có thể tạo các ứng dụng gốc cho iOS, Android và Windows.

#### b, Công cụ lập trình

- Android Studio là một môi trường phát triển tích hợp (IDE) chính thức của Google để lập trình ứng dụng Android. Android Studio hỗ trợ Java, Kotlin, C++ và cung cấp các công cụ mạnh mẽ như trình giả lập, hệ thống build dựa trên Gradle, và công cụ kiểm tra hiệu suất. Android Studio dành cho Google Android Development được ra mắt vào ngày 16 tháng 5 năm 2013, trong sự kiện I/O 2013 của Google. IDE viết tắt của Integrated Development Environment, được gọi là "Môi trường thiết kế hợp nhất" hay "Môi trường gỡ lỗi hợp nhất" (tiếng Anh: Integrated Debugging Environment) là một loại phần mềm máy tính có công dụng giúp đỡ các lập trình viên trong việc phát triển phần mềm. Để phát triển ứng dụng đầu tiên của mình, bạn cần tải xuống Android Studio cho nền tảng ưa thích (Windows, Mac OS X hoặc Linux) từ trang web dành cho nhà phát triển Android. Android Studio có thể hỗ trợ phát triển và thử nghiệm ứng dụng của bạn trên thiết bị thực hoặc trình giả lập.



Hình 1. 3 Công cụ lập trình Andoid Studio

- Andromo: Nền tảng phát triển ứng dụng di động Andromo đã khẳng định vị thế của họ trong cộng đồng lập trình viên Android như một công cụ phát triển phổ biến hàng đầu. Một trong những ưu điểm nổi bật của ứng dụng này là giao diện và thao tác đơn giản, mang lại cho người dùng sự dễ dàng trong quá trình viết App. Bên cạnh đó, nền tảng này còn cung cấp cơ hội miễn phí cho người mới bắt đầu làm quen với việc phát triển ứng dụng một cách dễ dàng và nhanh chóng. Trong quá trình sử dụng, bạn chỉ cần thực hiện theo ba bước cơ bản từ việc tạo dự án đến việc thêm tính năng, đồ họa và nội dung thông qua việc điền biểu mẫu. Điều này vô cùng tiện lợi để bạn có thể tạo nên những ứng dụng độc đáo và mang tính cá nhân. Còn lại, các công việc phức tạp khác sẽ được Andromo thực hiện. Tuy nhiên, cần lưu ý rằng công cụ này cũng tồn tại một số hạn chế. Ví dụ, nền tảng này không hỗ trợ tính năng xem trước ứng dụng, giúp bạn kiểm tra trước trải nghiệm của người dùng. Hơn nữa, việc tải ứng dụng lên Google Play Store cũng không được hỗ trợ trực tiếp từ Andromo. Thêm vào đó, để nâng cấp và quản lý nhiều hơn 50 ứng dụng, bạn sẽ phải trả một khoản phí hàng tháng là 8 USD để có thể tiếp tục sử dụng các tính năng cao cấp của nền tảng này.



Hình 1. 4 Công cụ lập trình Andromo

- Appypie: Appypie là một nền tảng tạo ứng dụng miễn phí, mang đến cho người dùng khả năng tạo ra các ứng dụng cho các hệ điều hành phổ biến như IOS, Android và Windows. Một điểm đáng chú ý là công cụ này có khả năng đưa ứng dụng lên cửa hàng iTunes hay Google Play dễ dàng thông qua việc sử dụng giao diện kéo và thả. Bên cạnh đó, việc viết App dựa trên HTML5 và các nền tảng khác sẽ được hoàn thiện dễ dàng hơn với nhiều chức năng đa dạng. Điều này cho phép bạn gửi thông báo đẩy, theo dõi phân tích thời gian thực, sử dụng GPS để định vị và tận dụng nguồn thu từ quảng cáo. Đồng thời, bạn cũng có khả năng liên kết ứng dụng với các trang web, âm thanh và radio, tạo ra sự tương tác đa phương tiện đa dạng. Với Appypie, bạn có tùy chọn tạo ứng dụng miễn phí hoặc chọn phiên bản trả phí với mức giá 33 đô la mỗi tháng. Điều này tạo cơ hội linh hoạt cho bạn để tùy chỉnh theo nhu cầu và ngân sách của dự án.



Hình 1. 5 Công cụ lập trình Appypie

### 1.1.3 Hệ điều hành Android

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Android Inc. với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005.

Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào năm 2008.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache.[9] Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do. Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi.[10] Tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới, vượt qua Symbian OS vào quý 4 năm 2010, và được các công ty công nghệ lựa chọn khi họ cần một hệ điều hành không nặng nề, có khả năng tinh chỉnh, và giá rẻ chạy trên các thiết bị công nghệ cao thay vì tạo dựng từ đầu. Kết quả là mặc dù được thiết kế để chạy trên điện thoại và máy tính bảng, Android đã xuất hiện trên TV, máy chơi game và các thiết bị điện tử khác. Bản chất mở của Android cũng khích lệ một đội ngũ đông đảo lập trình viên và những người đam mê sử dụng mã nguồn mở để tạo ra những dự án do cộng đồng quản lý. Những dự án này bổ sung các tính năng cao cấp

cho những người dùng thích tìm tòi hoặc đưa Android vào các thiết bị ban đầu chạy hệ điều hành khác.

Android chiếm 87,7% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 2 năm 2017, với tổng cộng 2 tỷ thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là "cuộc chiến điện thoại thông minh" giữa các công ty công nghệ.

Từ năm 2011, Android đã là hệ điều hành bán chạy nhất trên toàn cầu trên điện thoại thông minh và từ năm 2013 trên máy tính bảng. Tính đến tháng 5 năm 2021, nó có hơn ba tỷ người dùng hàng tháng, là hệ điều hành có cài đặt nhiều nhất trên thế giới, và tính đến tháng 1 năm 2021, Cửa hàng Google Play có hơn 3 triệu ứng dụng. Android 13, được phát hành vào ngày 15 tháng 8 năm 2022, là phiên bản mới nhất, và phiên bản Android 12.1/12L mới nhất bao gồm những cải tiến đặc biệt cho điện thoại gập, máy tính bảng, màn hình có kích thước lớn như máy tính để bàn 1080p và Chromebook.

Về giao diện người dùng của Android dựa trên nguyên tắc tác động trực tiếp, sử dụng cảm ứng chạm tương tự như những động tác ngoài đời thực như vuốt, chạm, kéo giãn và thu lại để xử lý các đối tượng trên màn hình. Sự phản ứng với tác động của người dùng diễn ra gần như ngay lập tức, nhằm tạo ra giao diện cảm ứng mượt mà, thường dùng tính năng rung của thiết bị để tạo phản hồi rung cho người dùng. Những thiết bị phản cứng bên trong như gia tốc kế, con quay hồi chuyển và cảm biến khoảng cách được một số ứng dụng sử dụng để phản hồi một số hành động khác của người dùng, ví dụ như điều chỉnh màn hình từ chế độ hiển thị dọc sang chế độ hiển thị ngang tùy theo vị trí của thiết bị, hoặc cho phép người dùng lái xe đua bằng xoay thiết bị, giống như đang điều khiển vô-lăng.

Android có lượng ứng dụng của bên thứ ba ngày càng nhiều, được chọn lọc và đặt trên một cửa hàng ứng dụng như Google Play hay Amazon Appstore để người dùng lấy về, hoặc bằng cách tải xuống rồi cài đặt tập tin "APK" từ trang web khác. Các ứng dụng trên Play Store cho phép người dùng duyệt, tải về và cập nhật các ứng dụng do Google và các nhà phát triển thứ ba phát hành. Play Store được cài đặt sẵn trên các thiết bị thỏa mãn điều kiện tương thích của Google. Ứng dụng sẽ tự động lọc ra một danh sách các ứng dụng tương thích với thiết bị của người dùng, và nhà phát triển có thể giới hạn ứng dụng của họ chỉ dành cho những nhà mạng cố định hoặc những quốc gia cố định vì lý do kinh doanh. Nếu người dùng mua một ứng dụng mà họ cảm thấy không thích, họ được hoàn trả tiền sau 15 phút kể từ lúc tải về và một vài nhà mạng còn có khả năng mua giúp các ứng dụng trên Google Play, sau đó tính tiền vào trong hóa đơn sử dụng hàng tháng của người dùng. Đến tháng 9 năm 2012, có hơn 675.000 ứng dụng dành cho Android, và số lượng ứng dụng tải về từ Play Store ước tính đạt 25 tỷ.

Bộ nhớ của các thiết bị Android có thể được mở rộng bằng các thiết bị phụ như thẻ SD. Android nhận dạng hai loại bộ nhớ phụ: bộ nhớ di động (được sử dụng theo mặc định) và bộ nhớ có thể áp dụng. Bộ nhớ di động được coi là một thiết bị lưu trữ bên ngoài. Bộ nhớ có thể áp dụng, được giới thiệu trên Android 6.0, cho phép bộ nhớ trong

của thiết bị được mở rộng bằng thẻ SD, coi nó như một phần mở rộng của bộ nhớ trong. Điều này có nhược điểm là ngăn không cho thẻ nhớ được sử dụng với thiết bị khác trừ khi nó được định dạng lại.

#### 1.1.4 Công cụ lập trình Android Studio

Android Studio là một môi trường tích hợp phát triển (Integrated Development Environment – IDE) được phát triển bởi Google dành cho việc phát triển ứng dụng trên nền tảng Android. Nó cung cấp một loạt các công cụ và tính năng để giúp nhà phát triển xây dựng và triển khai ứng dụng Android một cách dễ dàng.



Hình 1. 6 Công cụ lập trình Android Studio

Công cụ này cung cấp một giao diện trực quan và các trình biên dịch, trình gỡ lỗi, trình tạo giao diện người dùng, trình quản lý phiên bản và nhiều công cụ khác. Nó hỗ trợ viết code trong ngôn ngữ Java hoặc Kotlin và tích hợp sẵn các thư viện và công cụ phát triển Android. Một trong những tính năng quan trọng của công cụ này là khả năng sử dụng Gradle, một hệ thống quản lý dự án mạnh mẽ. Gradle cho phép bạn quản lý phụ thuộc, xây dựng, kiểm thử và đóng gói ứng dụng Android một cách linh hoạt và hiệu quả.

#### Android ra đời như thế nào?

Android Studio đã được công bố lần đầu tiên vào năm 2013 tại hội nghị Google I/O và chính thức phát hành rộng rãi vào năm 2014. Trước đó, các nhà phát triển ứng dụng Android thường sử dụng các công cụ như Eclipse IDE hoặc IDE của Java để phát triển ứng dụng.

Sau khi được công bố công cụ này đã trở thành một công cụ phát triển ứng dụng Android chính thức và được Google khuyến nghị sử dụng. Nó đã thay thế Eclipse IDE và trở thành môi trường phát triển chính cho các dự án Android.

Đối với những người có kinh nghiệm trong phát triển phần mềm, việc tạo ứng dụng trở nên dễ dàng hơn. Công cụ này cung cấp nhiều tính năng và công cụ hỗ trợ như trình biên dịch thông minh, gỡ lỗi tiện lợi, thiết kế giao diện trực quan và nhiều tính năng khác giúp tăng năng suất và hiệu quả trong quá trình phát triển ứng dụng Android.



Về ưu điểm của Android Studio:

- Đây là môi trường phát triển phần mềm chính thức của Google, đây cũng chính là chủ sở hữu của hệ điều hành Android.
- Công cụ này cung cấp một loạt các công cụ phát triển và tính năng hỗ trợ cho quá trình phát triển ứng dụng Android. Điều này bao gồm trình biên dịch, trình gỡ lỗi, trình quản lý dự án, thiết kế giao diện và nhiều tính năng khác để tăng năng suất và hiệu quả của nhà phát triển.
- Giao diện người dùng thân thiện và dễ sử dụng, giúp tăng hiệu suất và giảm thời gian cần thiết để phát triển ứng dụng.
- Hỗ trợ bởi một số tài liệu tham khảo và hướng dẫn phong phú. Có sẵn các tài liệu chính thức từ Google, cùng với các diễn đàn lập trình viên Android và các nguồn tài liệu trực tuyến khác để giúp nhà phát triển tìm hiểu và giải quyết các vấn đề phát triển.
- Công cụ này được hỗ trợ thông qua các khóa học đào tạo về lập trình Android cơ bản và nâng cao. Các khóa học này giúp nhà phát triển nắm vững các khái niệm và kỹ năng cần thiết để phát triển ứng dụng Android chất lượng.

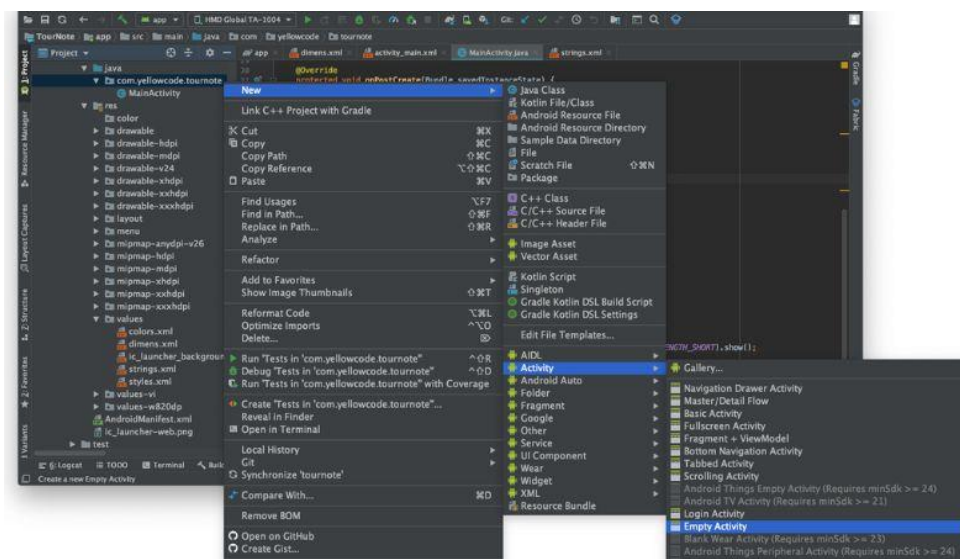
Ngoài những ưu điểm trên Android Studio cũng có những nhược điểm như:

- Đây là công cụ chiếm lượng lớn dữ liệu trong không gian bộ nhớ máy tính nếu được cài đặt.
- Việc kiểm tra hoạt động thông qua giả lập gây đơ, lag, giật máy và tiêu tốn pin.

## 1.2. Phát triển ứng dụng trên nền tảng Android sử dụng Java

### 1.2.1. Cấu trúc ứng dụng Android

Cấu trúc của một ứng dụng Android bao gồm nhiều thành phần khác nhau, mỗi thành phần có vai trò riêng trong việc tạo ra và điều khiển ứng dụng. Khi phát triển một ứng dụng Android, các thành phần này sẽ hoạt động cùng nhau thông qua một khung ứng dụng (Application Framework) để cung cấp trải nghiệm người dùng liền mạch. Các thành phần cơ bản của ứng dụng Android.



### Hình 1. 7 Tạo lập 1 Activity mới

Một ứng dụng Android được cấu trúc từ bốn thành phần chính:

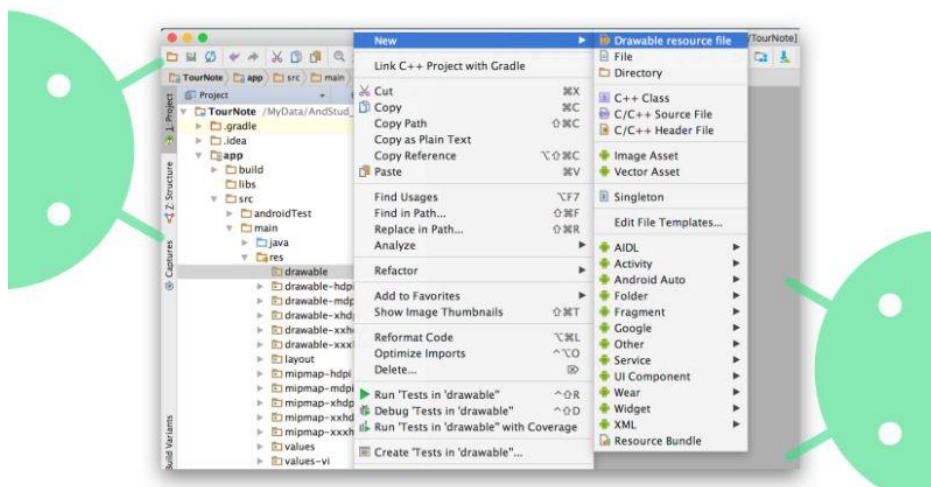
- Activity (Hoạt động): Activity là một thành phần đại diện cho một màn hình trong ứng dụng. Mỗi Activity có giao diện riêng và quản lý các tương tác của người dùng trên giao diện đó. Khi người dùng tương tác với ứng dụng (như nhấn nút hoặc vuốt màn hình), Activity sẽ xử lý các sự kiện này và phản hồi theo yêu cầu của người dùng. Ví dụ: Màn hình chính của ứng dụng hoặc màn hình đăng nhập đều có thể là một Activity.

- Service (Dịch vụ): Service là một thành phần chạy ngầm (background) mà không có giao diện người dùng. Service thường được sử dụng để thực hiện các tác vụ dài hạn hoặc tác vụ không yêu cầu tương tác trực tiếp với người dùng. Ví dụ: Phát nhạc trong nền, đồng bộ dữ liệu từ máy chủ, hoặc xử lý các tác vụ nặng.

- Broadcast Receiver (Bộ nhận phát sóng): Broadcast Receiver là thành phần cho phép ứng dụng nhận và xử lý các thông báo phát sóng từ hệ thống hoặc từ các ứng dụng khác. Các broadcast này có thể là sự kiện hệ thống (như thay đổi kết nối mạng, mức pin thấp) hoặc sự kiện tùy chỉnh. Ví dụ: Ứng dụng có thể lắng nghe sự kiện khi thiết bị sạc pin hoặc khi mạng Wi-Fi được kết nối.

- Content Provider (Nhà cung cấp nội dung): Content Provider cho phép chia sẻ dữ liệu giữa các ứng dụng. Đây là thành phần chịu trách nhiệm quản lý quyền truy cập đến một tập hợp dữ liệu cụ thể, có thể là dữ liệu lưu trữ trong cơ sở dữ liệu SQLite, tập tin, hoặc trên web. Ví dụ: Danh bạ điện thoại, thư viện ảnh của hệ thống là các ví dụ về Content Provider.

- Tập tin cấu hình AndroidManifest.xml: AndroidManifest.xml là tập tin cấu hình quan trọng của mỗi ứng dụng Android. Tập tin này chứa thông tin mô tả ứng dụng và các thành phần của nó, đồng thời xác định quyền hạn ứng dụng cần sử dụng.



Hình 1. 8 Tạo một tập XML Drawable Resource

+ Các thông tin quan trọng trong AndroidManifest.xml bao gồm:

- Khai báo Activity, Service, Broadcast Receiver, Content Provider.
- Quyền truy cập: Ví dụ, nếu ứng dụng cần truy cập Internet, bạn phải khai báo quyền "android.permission.INTERNET".
- Thông tin về ứng dụng: Bao gồm tên ứng dụng, biểu tượng, phiên bản, v.v.
- Intent Filter: Xác định loại thông tin mà Activity hoặc Service có thể xử lý.

- Thành phần giao diện người dùng: Android hỗ trợ nhiều cách để xây dựng giao diện người dùng, chủ yếu là sử dụng XML để định nghĩa bố cục (layout) và Java hoặc Kotlin để lập trình hành vi cho giao diện.

- Layout: Layout là cách tổ chức các thành phần giao diện (widgets) trong một Activity. Mỗi layout là một tệp XML định nghĩa vị trí và cách hiển thị của các thành phần như nút, văn bản, hình ảnh.

- Các loại layout phổ biến:

- LinearLayout: Xếp các thành phần theo chiều ngang hoặc chiều dọc.
- RelativeLayout: Xếp các thành phần dựa trên mối quan hệ của chúng với nhau hoặc với bố cục cha.
- ConstraintLayout: Mạnh mẽ và linh hoạt hơn, cho phép tạo giao diện phức tạp mà không cần quá nhiều lớp lồng ghép.
- View và ViewGroup
  - View: Là thành phần cơ bản nhất trong giao diện người dùng. Tất cả các thành phần như Button, TextView, ImageView đều là các lớp con của View.
  - ViewGroup: Là container chứa các Views hoặc ViewGroups khác. Nó quyết định cách các thành phần con được sắp xếp và hiển thị trên màn hình.
- Luồng hoạt động của ứng dụng Android (Application Lifecycle)

- Mỗi ứng dụng Android có một vòng đời hoạt động cụ thể, đặc biệt là với các Activity. Điều này giúp quản lý tài nguyên hệ thống và đảm bảo rằng ứng dụng hoạt động trơn tru ngay cả khi chạy nhiều ứng dụng cùng lúc.

- Vòng đời của Activity: Vòng đời của một Activity bao gồm các trạng thái chính và các phương thức tương ứng được gọi khi Activity thay đổi trạng thái. Các phương thức quan trọng bao gồm:

- onCreate(): Được gọi khi Activity được khởi tạo lần đầu. Đây là nơi khởi tạo các tài nguyên, thiết lập giao diện và các thành phần liên quan.
- onStart(): Được gọi khi Activity chuẩn bị hiển thị với người dùng.
- onResume(): Được gọi khi Activity bắt đầu tương tác với người dùng (hoạt động ở phía trước).
- onPause(): Được gọi khi Activity tạm dừng tương tác với người dùng (nhưng vẫn còn trong bộ nhớ).



- onStop(): Được gọi khi Activity không còn hiển thị với người dùng.
- onDestroy(): Được gọi khi Activity bị hủy và tài nguyên được giải phóng.
- Vòng đời của Service:
  - Started Service: Bắt đầu khi ứng dụng khác yêu cầu và chạy ngầm cho đến khi hoàn thành nhiệm vụ hoặc bị dừng.
  - Bound Service: Cho phép các ứng dụng khác kết nối và tương tác trực tiếp, thường dừng lại khi không còn ứng dụng nào kết nối.
- Các tài nguyên của ứng dụng (Resources): Ứng dụng Android sử dụng nhiều loại tài nguyên khác nhau ngoài mã nguồn như chuỗi văn bản, hình ảnh, màu sắc, âm thanh, video, và bố cục giao diện. Tất cả các tài nguyên này được lưu trữ trong thư mục res/ của dự án và quản lý thông qua R.java, một tập tin tự động tạo ra để tham chiếu tài nguyên trong mã nguồn.
- Các thư mục tài nguyên quan trọng:
  - res/layout/: Chứa các tệp XML định nghĩa bố cục giao diện.
  - res/values/: Chứa các giá trị như chuỗi văn bản (strings.xml), màu sắc (colors.xml), kích thước (dimens.xml), và kiểu dáng (styles.xml).
  - res/drawable/: Chứa các tệp hình ảnh hoặc tệp XML cho các đối tượng đồ họa.
- Cơ sở dữ liệu trong Android:
  - SQLite là hệ quản trị cơ sở dữ liệu được tích hợp sẵn trong Android, cho phép lưu trữ dữ liệu ứng dụng một cách hiệu quả.
  - Android cũng cung cấp Room là một thư viện ORM giúp tương tác với cơ sở dữ liệu SQLite dễ dàng hơn thông qua các đối tượng Java/Kotlin.
- Các thư viện hỗ trợ phát triển: Android cung cấp nhiều thư viện hỗ trợ phát triển, bao gồm:
  - Jetpack: Bộ thư viện hỗ trợ phát triển giao diện, quản lý dữ liệu, điều hướng ứng dụng, và tương tác với hệ thống.
  - Retrofit: Thư viện cho việc gọi API (RESTful) dễ dàng và mạnh mẽ.
  - Glide và Picasso: Thư viện tải và hiển thị hình ảnh từ mạng hoặc từ bộ nhớ cục bộ.
  - LiveData và ViewModel: Hỗ trợ quản lý và quan sát dữ liệu sống động, đảm bảo dữ liệu được cập nhật theo thời gian thực mà không cần phải làm mới Activity hoặc Fragment.

### 1.2.2. Các thành phần phát triển ứng dụng Android

- Thiết kế giao diện người dùng (UI) cho ứng dụng Android là một phần quan trọng để mang lại trải nghiệm tốt cho người dùng. Android cung cấp nhiều công cụ và thành phần để xây dựng giao diện đẹp và linh hoạt. Dưới đây là các thành phần chính trong thiết kế giao diện ứng dụng Android:

- + Trong phát triển ứng dụng Android, Activity là thành phần cơ bản nhất, đại diện cho một màn hình giao diện người dùng (UI). Mỗi Activity quản lý các tương tác với người dùng trên màn hình và có vòng đời riêng. Ví dụ, ứng dụng có thể có các Activity riêng biệt cho màn hình đăng nhập, màn hình chính hoặc danh sách sản phẩm.
- + Fragment là một phần của giao diện người dùng bên trong Activity, giúp thiết kế giao diện linh hoạt hơn, đặc biệt trên các thiết bị có màn hình lớn như tablet. Fragment có thể được tái sử dụng trong nhiều Activity và dễ dàng quản lý hơn. Ví dụ, một Fragment có thể hiển thị danh sách sản phẩm, trong khi Fragment khác hiển thị chi tiết sản phẩm khi được chọn.
- + Service là thành phần chạy ngầm trong ứng dụng, thực hiện các tác vụ lâu dài mà không cần giao diện người dùng. Điều này rất hữu ích cho các tác vụ nền như phát nhạc hoặc tải dữ liệu từ Internet. Ví dụ, một ứng dụng phát nhạc sẽ sử dụng Service để tiếp tục phát khi người dùng chuyển sang ứng dụng khác.
- + Broadcast Receiver là thành phần giúp ứng dụng lắng nghe và phản hồi các sự kiện từ hệ thống hoặc các ứng dụng khác. Ví dụ, Broadcast Receiver có thể nhận thông báo khi có sự thay đổi về kết nối mạng hoặc khi pin yếu, từ đó ứng dụng có thể thực hiện các hành động phù hợp.
- + Content Provider cho phép chia sẻ dữ liệu giữa các ứng dụng một cách an toàn và có tổ chức. Nó quản lý việc truy cập dữ liệu tập trung và cung cấp giao diện để các ứng dụng khác đọc hoặc ghi dữ liệu. Ví dụ, ứng dụng danh bạ sử dụng Content Provider để các ứng dụng khác có thể truy cập thông tin liên hệ.
- + View và ViewGroup là các thành phần tạo nên giao diện người dùng trong Android. View là các thành phần UI cơ bản như nút bấm, ô nhập liệu, trong khi ViewGroup là container chứa các View, như LinearLayout hoặc RelativeLayout. Chúng giúp xây dựng và tổ chức giao diện người dùng một cách hiệu quả.
- + Intent là cơ chế để giao tiếp giữa các thành phần trong Android như Activity, Service, và BroadcastReceiver. Intent giúp điều hướng giữa các màn hình hoặc khởi chạy một Service. Ví dụ, khi người dùng nhấn nút đăng nhập, Intent có thể chuyển từ màn hình đăng nhập sang màn hình chính của ứng dụng.
- + Manifest File (AndroidManifest.xml) là tệp cấu hình chính của ứng dụng Android. Tệp này khai báo tất cả các thành phần của ứng dụng như Activity, Service, và các quyền truy cập cần thiết. Đây là thành phần bắt buộc và giúp hệ thống hiểu cách ứng dụng hoạt động.
- + Resources là các tệp tài nguyên không phải mã nguồn, như giao diện XML, hình ảnh, và chuỗi văn bản. Chúng được lưu trữ trong thư mục res và giúp tách biệt giao diện với logic xử lý, tạo điều kiện cho việc quốc tế hóa và tùy chỉnh giao diện dễ dàng hơn.
- + Gradle Build System là công cụ quản lý việc xây dựng và đóng gói ứng dụng. Gradle giúp tự động hóa quy trình biên dịch, quản lý dependencies và cấu hình dự án. Tệp build.gradle được sử dụng để khai báo các thư viện cần thiết cho ứng dụng.

- + Android Jetpack là bộ thư viện hỗ trợ phát triển ứng dụng hiện đại, giúp đơn giản hóa nhiều tác vụ thường gặp. Jetpack bao gồm các thành phần như LiveData, ViewModel, Room (cơ sở dữ liệu) và Navigation, giúp cải thiện kiến trúc và hiệu suất ứng dụng.
- + Cuối cùng, Kotlin và Java là hai ngôn ngữ lập trình phổ biến nhất để phát triển ứng dụng Android. Kotlin hiện được khuyến nghị vì cú pháp hiện đại, ngắn gọn, và tích hợp tốt với các công cụ của Android, trong khi Java vẫn được sử dụng rộng rãi nhờ sự ổn định và phổ biến.

## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 2.1 Giới thiệu bài toán

#### 2.1.1 Giới thiệu về đề tài

- Ý tưởng ban đầu của chúng em là tạo ra ứng dụng đặt vé cho người dùng bao gồm các chức năng cơ bản như xem lịch trình xe, tra cứu vé đã đặt, đặt vé và thanh toán. Với ứng dụng quản lý bao gồm quản lý nhân sự (Employee management), quản lý xe (Car management), quản lý khách (Customer management), quản lý dịch vụ (Service management),... Hiện nay ứng dụng của chúng em đã bao gồm những chức năng nhất định. Với ứng dụng đặt vé dành cho khách hàng, khách hàng có thể tra cứu lịch trình xe, tra cứu nhà xe, tìm kiếm vé, đặt vé và thanh toán trực tiếp hoặc online. Với ứng dụng quản lý dành cho bên xe, ngoài các chức năng như thêm, sửa, xóa dữ liệu ứng dụng còn có thể tìm kiếm dữ liệu. Dữ liệu sẽ được lưu trữ trong hệ thống cơ sở dữ liệu MySQL. Bài báo cáo của chúng em còn bao gồm các sơ đồ, biểu đồ cần thiết để xây dựng bài như biểu đồ phân rã chức năng, sơ đồ lớp, biểu đồ trình tự, biểu đồ hoạt động,...

- Đề tài "Xây dựng ứng dụng quản lý bên xe khách trên nền tảng Android sử dụng PHP và Android Studio" của chúng em tập trung vào việc phát triển một ứng dụng di động cho Android, cho phép quản lý bên xe và các hoạt động liên quan đến các chuyến xe, vé, hành khách và tài xế. Mục tiêu của ứng dụng là giúp các nhà quản lý bên xe, nhân viên bán vé, tài xế và hành khách có thể tương tác và quản lý thông tin liên quan một cách thuận tiện và hiệu quả.

- Ứng dụng sẽ có hai phần chính:

- + Ứng dụng di động (Android): Dành cho người dùng cuối như hành khách và tài xế để theo dõi thông tin chuyến xe, vé, lịch trình, và tình trạng xe.
- + Backend (PHP): Xử lý và lưu trữ dữ liệu từ ứng dụng Android, thực hiện các chức năng quản lý bên xe, vé, và cập nhật tình trạng xe theo thời gian thực.

Mục tiêu :

- Tăng cường trải nghiệm người dùng: Giao diện thân thiện, dễ sử dụng, bắt mắt.
- Tối ưu hóa quy trình quản lý: Quản lý được lịch trình xe, khách hàng, vé xe, dịch vụ, cơ sở vật chất. Tối ưu các trường quản lý tránh dài dòng, gây lẫn, rối trong quá trình quản lý.
- Tăng cường bảo mật: Tổ chức phân quyền cho từng nhân viên quản lý riêng, không thể xem mục không thuộc phạm vi quản lý của mình. Từng nhân viên phải có tài khoản riêng, yêu cầu mật khẩu riêng.
- Hỗ trợ chăm sóc khách hàng: Khách hàng có thể gọi vào số điện thoại hoặc nhắn tin lên fanpage, email của trang chủ để phản ánh trải nghiệm của mình.

#### 2.1.2 Yêu cầu bài toán

Để đạt được mục tiêu nghiên cứu, đề tài "Quản lý bên xe khách" sẽ triển khai một kế hoạch xây dựng cụ thể như sau:

Bước đầu tiên của đề tài là nghiên cứu cơ sở và thu thập thông tin về các hệ thống quản lý bến xe hiện có trên thế giới. Qua đó, em sẽ tìm hiểu sâu về các vấn đề và thách thức mà các bến xe đang phải đối mặt, cũng như những giải pháp công nghệ đã được áp dụng để cải thiện hiệu quả quản lý và phục vụ khách hàng. Trong đó bao gồm cả việc tìm hiểu ý kiến của khách hàng và các nhà phân phối, quản lý.

Bước tiếp theo là thiết kế hệ thống: Dựa trên thông tin thu thập được, đề tài sẽ xác định và phân tích các yêu cầu cụ thể của người dùng và các nhà quản lý bến xe. Em sẽ thiết kế một hệ thống thông minh và tích hợp các chức năng quản lý như:

- Về phía nhà tuyển dụng:
  - + Quản lý nhân viên bến xe
  - + Quản lý xe và vị trí xe
  - + Quản lý khách hàng
  - + Quản lý cơ sở hạ tầng bến xe
  - + Quản lý dịch vụ bến xe
  - + Quản lý lịch trình và đặt vé trực tuyến.
  - + Quản lý vé
- Về phía người dùng:
  - + Đặt vé
  - + Xem danh sách tuyến xe, nhà xe
  - + Xem thông tin mua vé
  - + Xem mức giá vé
  - + Thanh toán giá vé trực tuyến

Ngoài ra chúng em cần kết nối cơ sở dữ liệu:

- Sử dụng các lớp và phương thức trong Android Studio để kết nối đến cơ sở dữ liệu đã chọn.
- Tạo các đối tượng để thực hiện các truy vấn SQL (SELECT, INSERT, UPDATE, DELETE).
- Xây dựng lớp xử lý nghiệp vụ
- Tạo các lớp đại diện cho các đối tượng trong ứng dụng.
- Mỗi lớp có các thuộc tính tương ứng với các trường trong cơ sở dữ liệu và các phương thức để thực hiện các thao tác liên quan đến đối tượng đó.

- Các phương thức này sẽ gọi các truy vấn SQL thông qua lớp kết nối cơ sở dữ liệu để lấy hoặc cập nhật dữ liệu.

Sau đó chúng em Sử dụng các công cụ và điều khiển có sẵn trong Netbeans để tạo giao diện cho ứng dụng. Tạo các form, button, textbox, label, panel, table, ... để hiển thị và nhập liệu. Viết mã để xử lý các sự kiện trên giao diện

Sau khi hoàn tất thiết kế, chúng tôi sẽ tiến hành phát triển các phần mềm và ứng dụng cần thiết cho hệ thống. Sử dụng các công nghệ hiện đại như ASP.NET, Java, và MySQL, em sẽ xây dựng các thành phần như ứng dụng web, ứng dụng di động và cơ sở dữ liệu quản lý thông tin đa dạng.

Tiếp theo đó, để đảm bảo hiệu quả và tính ứng dụng của đề tài, chúng em sẽ tiến hành đánh giá kết quả dựa trên các chỉ số như sự hài lòng của người dùng, giảm thiểu thời gian chờ đợi và cải thiện năng suất của các bên xe. Các phản hồi từ người dùng và nhà quản lý sẽ được sử dụng để điều chỉnh và cải tiến hệ thống, từ đó nâng cao hiệu quả và tính ứng dụng trong thực tế. Kiểm thử đơn vị: Kiểm tra từng đơn vị mã nhỏ nhất (hàm, phương thức) để đảm bảo chúng hoạt động đúng như mong đợi.

- Viết các trường hợp kiểm thử để kiểm tra các trường hợp nhập liệu khác nhau và kết quả trả về của hàm.
- Chạy các trường hợp kiểm thử và so sánh kết quả thực tế với kết quả mong đợi.
- Kiểm thử tích hợp: Kiểm tra sự tương tác giữa các module, lớp, thành phần khác nhau của ứng dụng.
- Kết hợp các module đã được kiểm thử đơn vị và kiểm tra xem chúng có hoạt động cùng nhau một cách chính xác không.
- Sử dụng các kỹ thuật như kiểm thử từ trên xuống, kiểm thử từ dưới lên hoặc kiểm thử luồng điều khiển.
- Kiểm thử hệ thống: Kiểm tra toàn bộ ứng dụng trên môi trường gần giống với môi trường thực tế để đảm bảo ứng dụng đáp ứng được các yêu cầu về chức năng và phi chức năng.
- Kiểm thử chức năng: Kiểm tra xem tất cả các chức năng của ứng dụng có hoạt động đúng như yêu cầu không.
- Kiểm thử hiệu năng: Kiểm tra xem ứng dụng có hoạt động đủ nhanh và ổn định dưới tải trọng lớn không.
- Kiểm thử khả dụng: Kiểm tra xem ứng dụng có dễ sử dụng và thân thiện với người dùng không.

- Sửa lỗi: Tìm và sửa các lỗi (bug) trong ứng dụng.
- Sử dụng các công cụ debug của Netbeans để theo dõi quá trình thực thi của chương trình, kiểm tra giá trị của các biến, tìm ra vị trí gây lỗi.
- Phân tích các thông báo lỗi (error message) để hiểu rõ nguyên nhân gây lỗi.
- Sửa mã nguồn để khắc phục lỗi.
- Chạy lại các trường hợp kiểm thử để đảm bảo lỗi đã được sửa hoàn toàn.
- Cuối cùng, chúng em sẽ tổng kết kết quả nghiên cứu và viết báo cáo chi tiết về đề tài. Báo cáo này sẽ trình bày những phát hiện quan trọng, những thành công đạt được và những bài học rút ra từ quá trình nghiên cứu và triển khai. Đồng thời, em cũng sẽ trình bày báo cáo tại lớp để chia sẻ kinh nghiệm và học hỏi từ giảng viên và các bạn trong lớp. Cài đặt hệ điều hành và các phần mềm cần thiết trên máy chủ.
- Cấu hình môi trường để ứng dụng có thể chạy được.
- Xây dựng bản cài đặt.
- Sử dụng công cụ Netbeans hoặc các công cụ của bên thứ ba để tạo bản cài đặt cho ứng dụng.
- Bản cài đặt nên bao gồm tất cả các tệp cần thiết (tệp thực thi, thư viện, tệp cấu hình) và các bước cài đặt tự động.
- Cài đặt và kiểm tra.
- Kiểm tra kỹ lưỡng các chức năng của ứng dụng trên môi trường mới để đảm bảo chúng hoạt động đúng.
- Hoàn thiện.
- Tối ưu hóa hiệu năng: tìm và sửa các đoạn code gây ảnh hưởng đến hiệu năng. Cải thiện tốc độ truy vấn cơ sở dữ liệu. Giảm thiểu việc sử dụng tài nguyên hệ thống.
- Cải thiện giao diện người dùng: điều chỉnh giao diện dựa trên phản hồi của người dùng. Làm cho giao diện trực quan, dễ sử dụng hơn.
- Bổ sung tính năng: thêm các tính năng mới nếu cần thiết và có đủ thời gian.

Chúng em rất mong nhận được lời góp ý từ các thầy cô để website của chúng em thêm hoàn thiện hơn, có tính ứng dụng cao hơn, tốt hơn, tiếp cận được với người dùng. Cảm ơn thầy cô rất nhiều!

### 2.1.2.1 Yêu cầu chức năng

- Trong các bến xe hiện nay, việc quản lý và tổ chức, sắp xếp các xe, khách hàng luôn là vấn đề được quan tâm. Đặc biệt với những bến xe lớn như bến xe Mỹ Đình hay bến xe Gia Lâm với lượng xe ra vào rất đông mỗi giờ, mỗi ngày thì việc quản lý chúng càng trở nên phức tạp và được quan tâm. Quan tâm nhất có lẽ là quá trình bán vé, người quản lý luôn gặp khó khăn trong việc kiểm soát xe, lịch trình, hành khách mua vé, trang thiết bị,... Khách hàng luôn phải xếp hàng chờ đợi, vào những thời gian cao điểm như Tết, các dịp lễ 2/9, 30/04,... lượng khách mua vé đi các tỉnh đông gấp nhiều lần gây ra ùn tắc, xếp hàng chật chội trong vài giờ liền cũng chưa chắc mua được vé. Vì vậy, chúng em muốn đưa tới một trải nghiệm tốt hơn cho khách hàng, giảm thiểu thực trạng ấy. Hệ thống của chúng em hướng tới việc quản lý khách hàng, nhân viên, xe ra/ vào bến, cơ sở vật chất và dịch vụ của bến xe. Hệ thống sẽ bao gồm các chức năng chính như: quản lý khách, quản lý nhân viên, quản lý xe, quản lý đặt vé, quản lý dịch vụ, quản lý cơ sở vật chất và website đặt vé của khách hàng.

- Các chức năng này làm cho hoạt động của bến xe diễn ra một cách ổn định, tối ưu được thời gian đặt mua vé, giúp người quản lý làm việc dễ dàng hơn.

### 2.1.2.2 Yêu cầu phi chức năng

#### a) Yêu cầu về tốc độ xử lý

Tốc độ xử lý như: Đăng kí, đăng nhập vào hệ thống, tìm kiếm phải nhanh và chính xác, các trang có giao diện thống nhất tránh sự nhập nhằng, rắc rối tránh sự rườm rà không cần thiết. Thông tin gửi về cho khách hàng nhanh chóng tránh trường hợp phải chờ đợi lâu

#### b) Yêu cầu về an toàn và bảo mật

- Chế độ backup dữ liệu: Hệ thống cần có chế độ lưu lại số lần truy cập của khách hàng.
- Người dùng có thể vào xem thông tin mà không cần đăng kí hay hay đăng nhập tài khoản. Trường hợp họ muốn sử dụng chức năng đặt vé hay thanh toán... bắt buộc phải đăng nhập. Nếu chưa có tài khoản người dùng phải đăng kí tài khoản
- Các chức năng nhập, sửa, xóa thông tin chỉ có người quản trị được sử dụng.
- Đảm bảo máy chủ luôn hoạt động và kết nối với mạng Internet, cập nhật và đưa các thông tin lên mạng một cách nhanh nhất
- Để đăng kí tài khoản cần phải có các thông tin sau:
  - Nhập địa chỉ email: Địa chỉ này sẽ được hệ thống kiểm tra xem có thực sự tồn tại hay không. Địa chỉ mail có đúng định dạng không, nếu không đúng sẽ đưa ra thông báo lỗi yêu cầu người dùng nhập lại.
  - Nhập hai lần Password: Password là một dãy kí tự yêu cầu phải từ sáu kí tự trở lên không có các kí tự đặc biệt. Nếu password đầu và cuối không trùng nhau thì hiện thông báo lỗi: “Bạn nhập password không trùng nhau, hãy nhập lại”.
  - Một số các thông tin khác yêu cầu người dùng nhập hoặc xác nhận nếu thiếu sẽ cảnh báo: “Bạn phải điền đầy đủ thông tin”.
- Bảo mật chống truy cập trái phép từ bên ngoài: Mật khẩu của người quản trị khách hàng sẽ được mã hóa trong khi nhập và trong cả CSDL. Những thông tin quan trọng bên trong mã code sẽ không được hiển thị trên thanh địa chỉ mà được che dấu để tránh những phần tử xấu xâm nhập vào trang web và sửa đổi nội dung của nó.
- Bảo mật dữ liệu trên đường truyền và máy chủ: được bảo mật một cách triệt để.



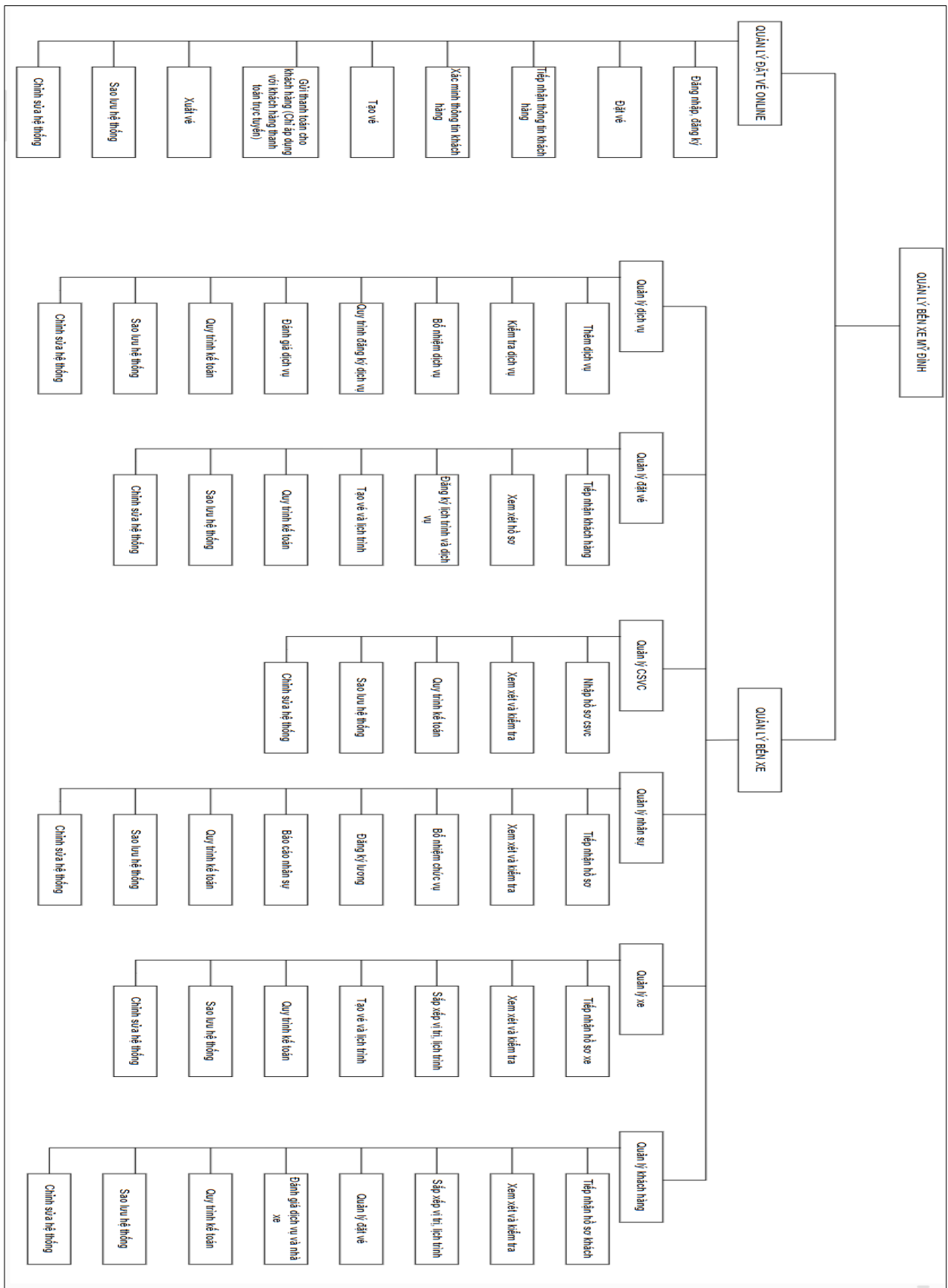
- Quản lý người sử dụng và lưu trữ dấu vết: Hệ thống quản lý người sử dụng thông qua tài khoản khi họ đăng kí thành viên. Khi khách hàng đăng kí xong với email và mật khẩu thì hệ thống xác nhận bằng cách gửi về mail của họ một yêu cầu và yêu cầu họ phải vào mail để xác nhận lại, tránh tình trạng đăng kí mail ảo

c) Giao diện người sử dụng

- Giao diện người dùng phải thân thiện, dễ sử dụng đối với cả những người chưa có kinh nghiệm làm việc trên Internet. Màu sắc trong trang web phải hài hòa tránh sự lộn xộn. Giao diện phải kích thích sự hứng thú và tò mò của bất kì khách ghé thăm nào khi đã mở trang ra xem một lần.

- Nội dung hiển thị trên trang web phải phong phú, chính xác, cập nhật thường xuyên những thông tin về việc làm mới nhất để thu hút người xem. Tránh những trang web chết.

## 2.2 Sơ đồ chức năng của phần mềm



Hình 2. 1 Biểu đồ phân rã chức năng hệ thống

Bảng 2. 1 Mô tả biểu đồ phân rã chức năng

Tên chức năng	Mô tả
Quản lý ứng dụng đặt vé online	<ul style="list-style-type: none"> <li>- Đăng nhập, đăng ký: Cho phép người dùng tạo tài khoản để đặt vé, theo dõi lịch trình xe, thuận tiện cho người dùng.</li> <li>- Đặt vé: Sau khi có tài khoản người dùng có thể đặt vé trực tiếp đến nhà xe hoặc liên hệ trực tiếp vào đường dây nóng để trực tiếp đặt vé và thương lượng giá cả.</li> <li>- Tiếp nhận thông tin khách: Tiếp nhận thông tin hồ sơ của khách, vé khách đã đặt, đã đi (Nếu khách đã đăng ký vé).</li> <li>- Xác minh thông tin khách hàng: Xác minh lại thông tin khách hàng như số điện thoại, họ và tên, giới tính, độ tuổi, vé đã đặt, ... tránh sai sót, nhầm lẫn.</li> <li>- Tạo vé: Tạo ra mẫu vé xe cho khách hàng, mỗi người, mỗi tuyến xe sẽ có một loại vé, mẫu vé khác nhau.</li> <li>- Gửi thanh toán cho khách hàng: Đối với khách hàng thanh toán trước sẽ có mã QR để thanh toán trước số tiền xe.</li> <li>- Xuất vé: Tạo vé và đưa đến vị trí khách hàng.</li> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chỉnh sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho khách hàng.</li> </ul>
	<p style="text-align: center;">Quản lý nhân sự</p> <ul style="list-style-type: none"> <li>- Tiếp nhận hồ sơ: Phỏng vấn nhân sự và tiếp nhận hồ sơ.</li> <li>- Xem xét và kiểm tra: Xem xét lại và kiểm tra nhân sự với nhiều tiêu chí xét tuyển dụng nghiêm ngặt.</li> <li>- Bổ nhiệm chức vụ: Sau khi thông qua vòng sơ tuyển nhân sự sẽ được bổ nhiệm chức vụ làm việc đúng nhu cầu và thực lực hiện có của nhân viên</li> <li>- Quy trình kế toán: Xem xét lương, thưởng, lợi ích, BHXH, BHYT, ...</li> </ul>

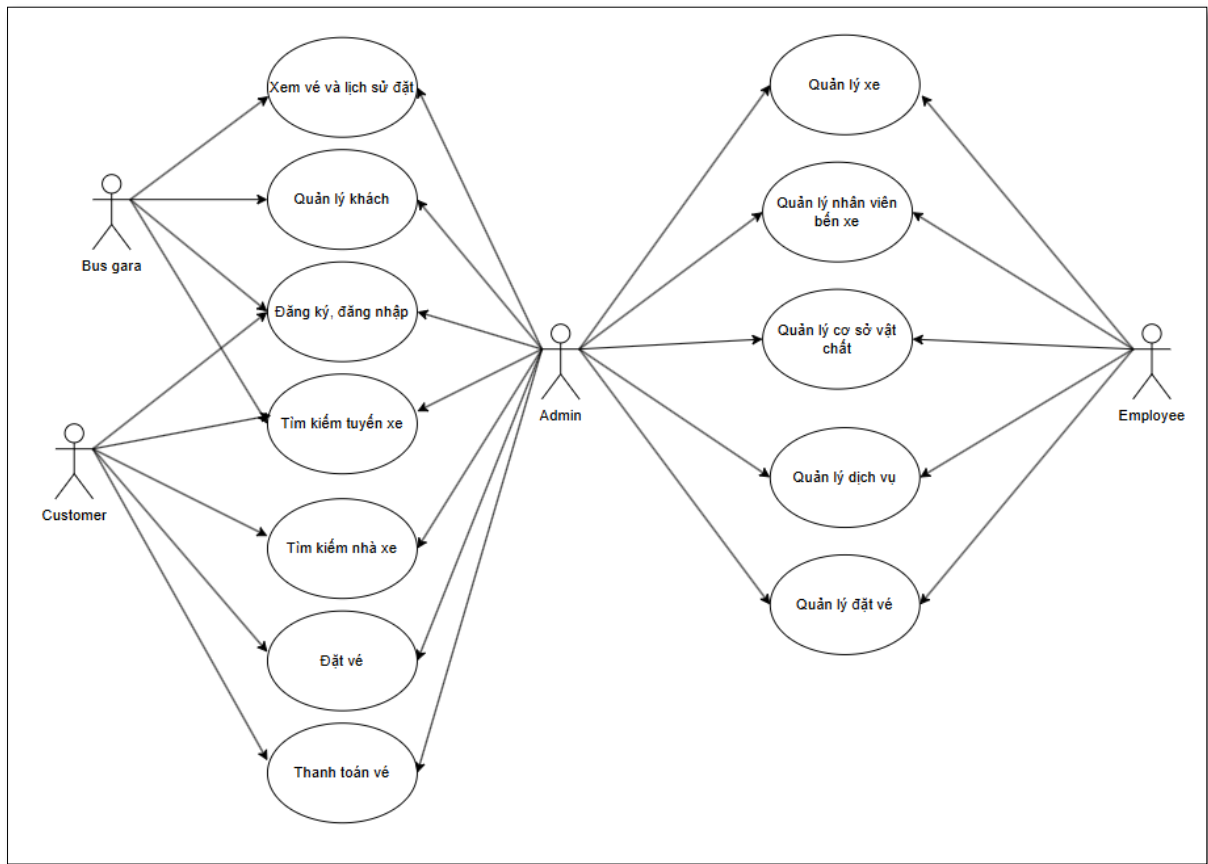
Quản lý bến xe khách		<ul style="list-style-type: none"> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chính sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho nhân viên</li> </ul>
	Quản lý khách	<ul style="list-style-type: none"> <li>- Tiếp nhận hồ sơ khách hàng</li> <li>- Xem xét và kiểm tra: Xem xét hồ sơ khách hàng, kiểm tra lại tránh sai sót, nhầm lẫn gây ảnh hưởng đến trải nghiệm khách hàng.</li> <li>- Sắp xếp vị trí, lịch trình: Sắp xếp vị trí khách hàng (Khách hàng mới, khách hàng tiềm năng, ...), sắp xếp lịch trình đi theo vé khách đặt.</li> <li>- Quản lý đặt vé: Quản lý vé mà khách hàng đã mua trước đó, đảm bảo tính bảo mật của khách hàng.</li> <li>- Đánh giá dịch vụ: Khách hàng đánh giá dịch vụ và nhà xe để cải thiện trải nghiệm vào những lần tiếp theo.</li> <li>- Quy trình kế toán: Thiết lập thanh toán cho khách hàng, trả lại công cho nhà xe và phân phát lương cho nhân viên.</li> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chính sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho nhân viên.</li> </ul>
	Quản lý xe	<ul style="list-style-type: none"> <li>- Tiếp nhận hồ sơ: Tiếp nhận hồ sơ của nhà xe.</li> <li>- Xem xét, kiểm tra: Xem xét và kiểm tra thông tin nhà xe, tiểu sử, các lái và phụ xe, ...</li> <li>- Sắp xếp vị trí, lịch trình: Sắp xếp vị trí đỗ cố định và tham khảo lịch trình chạy của từng xe, thu thập lại.</li> </ul>

		<ul style="list-style-type: none"> <li>- Tạo vé và lịch trình: Tạo vé cố định của xe, thêm lịch trình cho vé xe, tạo sự thuận tiện cho khách, cũng như nhà xe.</li> <li>- Quy trình kế toán: Tính toán giá tiền, gửi đến khách hàng và nhà xe.</li> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chỉnh sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho nhân viên.</li> </ul>
	Quản lý đặt vé	<ul style="list-style-type: none"> <li>- Tiếp nhận khách hàng: Tiếp nhận hồ sơ khách hàng và thêm vào danh sách.</li> <li>- Xem xét hồ sơ: Xem xét hồ sơ khách hàng, hỗ trợ sửa hồ sơ cho khách.</li> <li>- Đăng ký lịch trình và dịch vụ: Đăng ký lịch trình và dịch vụ theo nhu cầu của khách hàng.</li> <li>- Tạo vé và lịch trình: Tạo lập vé cho khách ghi rõ lịch trình, họ tên và một số điều cần lưu ý trên vé.</li> <li>- Quy trình kế toán: Tính toán giá tiền, gửi đến khách hàng, hệ thống bên xe và nhà xe.</li> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chỉnh sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho nhân viên.</li> </ul>
	Quản lý dịch vụ	<ul style="list-style-type: none"> <li>- Thêm dịch vụ: Thêm mới dịch vụ cho khách hàng.</li> <li>- Kiểm tra dịch vụ: Kiểm tra dịch vụ có phù hợp với nhu cầu và túi tiền của khách hay không?</li> <li>- Bỏ nhiệm dịch vụ: Tạo hệ thống các dịch vụ.</li> <li>- Quy trình đăng ký dịch vụ: Hiển thị quy trình đăng ký dịch vụ cho khách hàng.</li> </ul>

		<ul style="list-style-type: none"> <li>- Đánh giá dịch vụ: Đánh giá dịch vụ theo trải nghiệm của khách hàng và của nhà xe để cân đối tránh trải nghiệm không tốt.</li> <li>- Quy trình kế toán: Tính toán giá tiền, gửi đến khách hàng và nhà xe.</li> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chỉnh sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho nhân viên.</li> </ul>
	Quản lý cơ sở vật chất	<ul style="list-style-type: none"> <li>- Nhập hồ sơ: Thêm mới csvc, ghi rõ ràng các mục ngày nhập, cấp, thời gian bảo dưỡng, giá trị.</li> <li>- Xem xét: Xem xét quá trình bảo dưỡng, thu/chi.</li> <li>- Quy trình kế toán: Tính toán giá tiền, gửi đến khách hàng và nhà xe.</li> <li>- Sao lưu hệ thống: Lập kế hoạch sao lưu, chọn phương pháp sao lưu, thực hiện sao lưu định kỳ, kiểm tra và xác nhận sao lưu, lưu trữ bản sao dự phòng, kiểm tra định kỳ và cập nhật.</li> <li>- Chỉnh sửa thông tin: Xác định nhu cầu chỉnh sửa, tìm kiếm và truy cập thông tin cho sinh viên, thực hiện chỉnh sửa, xác nhận và kiểm tra lại, lưu trữ thông tin mới, thông báo cho nhân viên.</li> </ul>

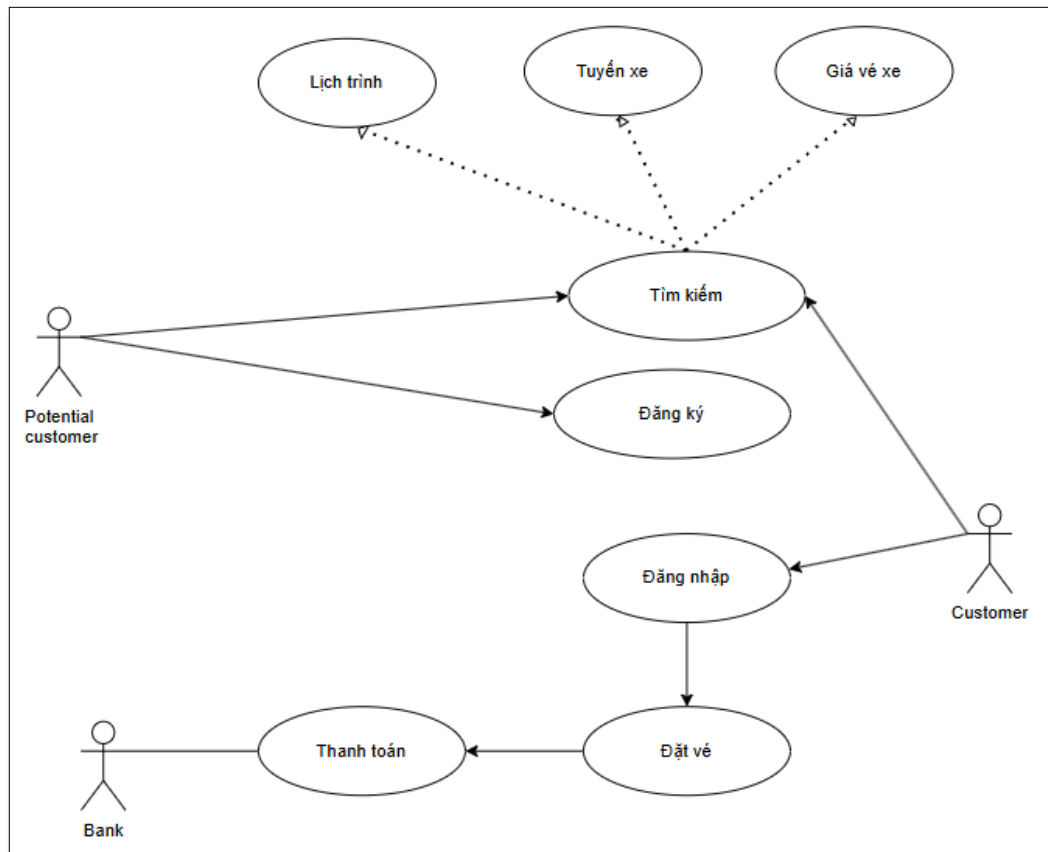
## 2.3 Mô hình hoá yêu cầu

### 2.3.1 Mô hình UC tổng quát



Hình 2. 2 Biểu đồ Usecase tổng quát

### 2.3.2 Mô hình UC hoạt động của khách hàng



Hình 2. 3 Biểu đồ UC khách hàng

Bảng 2. 2 Đặc tả UC đăng ký

Tên UC	Đăng ký
Tác nhân (Actor)	Potential customer (Khách hàng tiềm năng)
Điều kiện	Người dùng chưa có tài khoản
Mô tả	Khách hàng cần đăng ký tài khoản để đặt vé online.
Luồng sự kiện chính	1. Người sử dụng chọn chức năng “Đăng ký”. 2. Hệ thống hiển thị form “Đăng ký”. 3. Người dùng nhập thông tin yêu cầu, gửi thông tin đến hệ thống. 4. Hệ thống kiểm tra. Nếu đúng thì cho phép truy cập chức năng đăng nhập vào hệ thống. Nếu sai hệ thống báo lỗi tài khoản đã tồn tại và yêu cầu người dùng nhập lại thông tin.
Luồng sự kiện phụ (ngoại lệ)	- Không nhập đủ thông tin hệ thống báo lỗi. - Thông tin đã tồn tại hệ thống báo lỗi.



Bảng 2. 3 Đặc tả UC Đăng nhập

Tên UC	Đăng nhập
Tác nhân (Actor)	Potential customer (Khách hàng tiềm năng), Customer (Khách hàng)
Điều kiện	Người dùng đã có tài khoản
Mô tả	Khách hàng cần đăng nhập tài khoản để đặt vé online.
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Người sử dụng chọn chức năng “Đăng nhập”.</li> <li>2. Hệ thống hiển thị form “Đăng nhập”.</li> <li>3. Người dùng nhập thông tin yêu cầu, gửi thông tin đến hệ thống.</li> <li>4. Hệ thống kiểm tra. Nếu đúng thì cho phép truy cập các chức năng đăng nhập vào hệ thống. Nếu sai hệ thống báo lỗi sai mật khẩu hoặc tên đăng nhập</li> </ol>
Luồng sự kiện phụ (ngoại lệ)	<ul style="list-style-type: none"> <li>- Không nhập đủ thông tin hệ thống báo lỗi.</li> <li>- Thông tin nhập bị sai hệ thống báo lỗi.</li> </ul>

Bảng 2. 4 Đặc tả UC tìm kiếm

Tên UC	Tìm kiếm
Tác nhân (Actor)	Potential customer (Khách hàng tiềm năng), Customer (Khách hàng)
Điều kiện	Người dùng đã truy cập vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống, vào giao diện trang chủ và chọn tìm kiếm theo yêu cầu.
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Người dùng vào giao diện trang chủ</li> <li>2. Người dùng chọn một trong các từ khóa sau để tìm kiếm: <ol style="list-style-type: none"> <li>2.1 Lịch trình</li> <li>2.2 Giá vé</li> <li>2.3 Tuyến xe</li> </ol> </li> <li>3. Hệ thống hiển thị giao diện người dùng đã chọn.</li> </ol>
Luồng sự kiện phụ (ngoại lệ)	

Bảng 2. 5 Đặc tả UC đặt vé

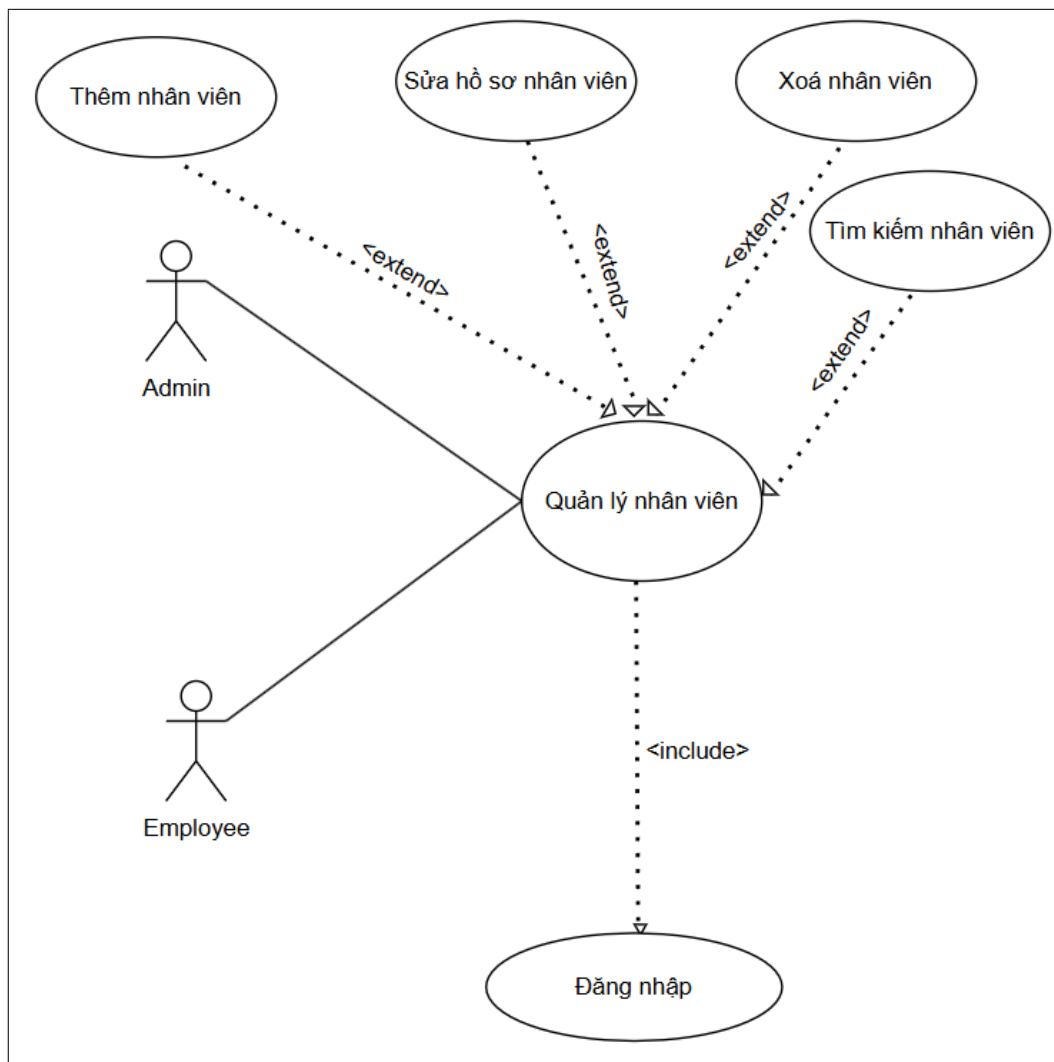
Tên UC	Đặt vé
Tác nhân (Actor)	Customer (Khách hàng)

Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng đăng nhập vào hệ thống, xem lịch chiếu và đặt vé
Luồng sự kiện chính	1. Người dùng xem lịch trình, chọn tuyến xe, nhà xe, dịch vụ. 2. Người dùng chọn giá vé, nhà xe rồi thanh toán.
Luồng sự kiện phụ (ngoại lệ)	

Bảng 2. 6 Đặc tả UC thanh toán

Tên UC	Thanh toán
Tác nhân (Actor)	Customer (Khách hàng)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống, xem lịch trình, giá vé, nhà xe và chọn đặt vé.
Mô tả	Sau chọn đặt vé, người dùng cần chọn một trong các phương thức thanh toán online để thanh toán vé.
Luồng sự kiện chính	1. Người dùng ấn thanh toán 2. Hệ thống hiện thị giao diện thanh toán với nhiều phương thức thanh toán. 3. Người dùng chọn một trong các phương thức thanh toán, điền thông tin thanh toán. 4. Hệ thống kiểm tra dữ liệu và thông báo đã thanh toán thành công.
Luồng sự kiện phụ (ngoại lệ)	Nhập sai thông tin thanh toán hệ thống sẽ báo lỗi và yêu cầu nhập lại.

### 2.3.3 Mô hình UC quản lý nhân viên



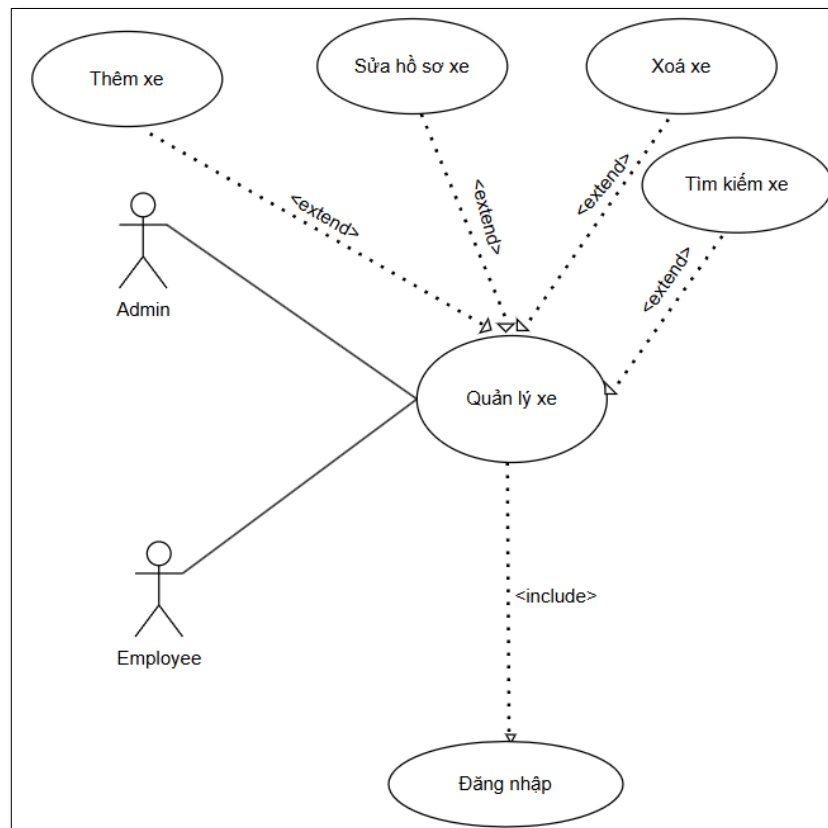
Hình 2. 4 UC quản lý nhân viên

Bảng 2. 7 Đặc tả UC quản lý nhân viên

Tên UC	Quản lý nhân viên
Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống và chọn chức năng quản lý nhân viên. Người dùng có thể sử dụng các chức năng như: thêm, tìm, sửa thông tin và xóa nhân viên.
Luồng sự kiện chính	1. Người dùng đăng nhập vào hệ thống. 2. Người dùng tìm chức năng quản lý nhân viên. 3. Người dùng chọn một trong các chức năng sau để thực hiện:

	3.1 Tìm nhân viên 3.2 Thêm nhân viên 3.3 Sửa thông tin nhân viên 3.4 Xóa nhân viên 4.Hệ thống hiển thị giao diện chức năng người dùng đã chọn.
Luồng sự kiện phụ (ngoại lệ)	

#### 2.3.4 Mô hình UC quản lý xe



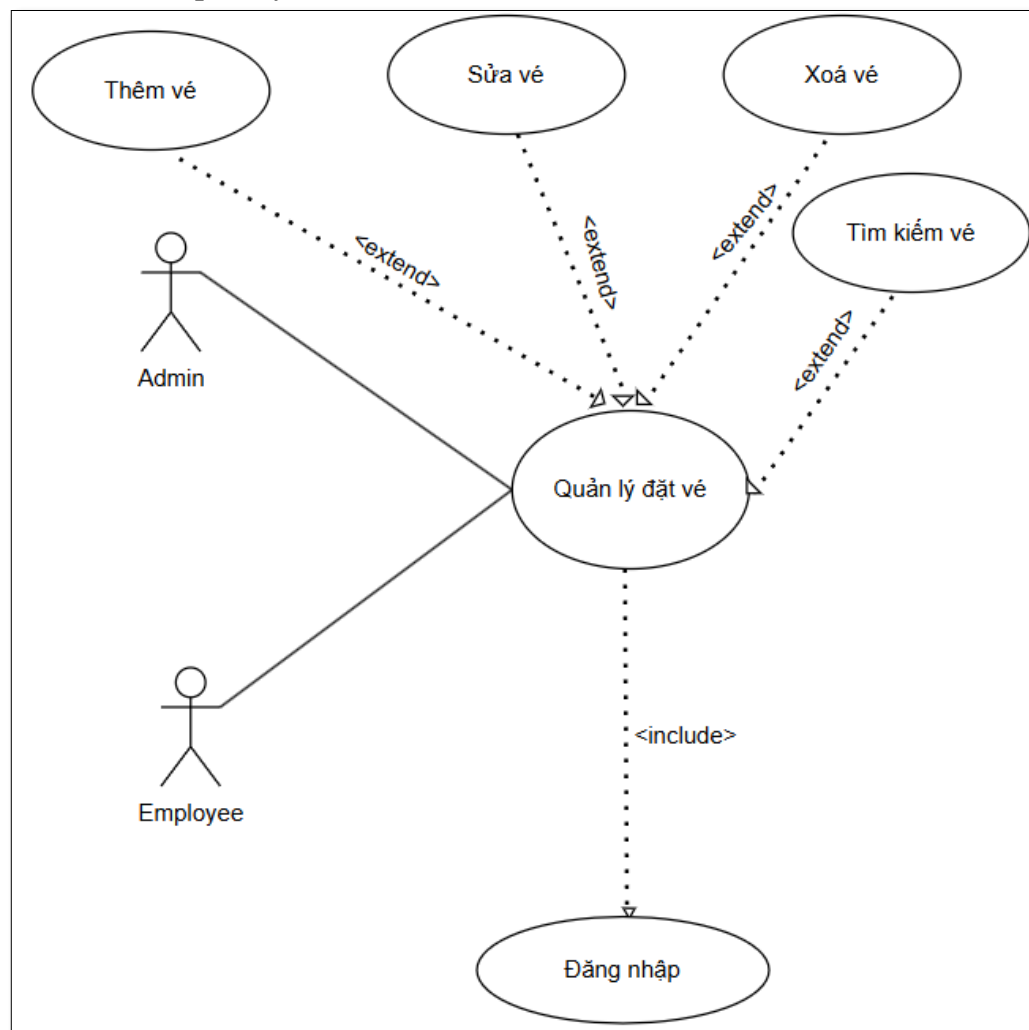
Hình 2. 5 UC quản lý xe

Bảng 2. 8 Đặc tả UC quản lý xe

Tên UC	Quản lý xe
Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống và chọn chức năng quản lý xe. Người dùng có thể sử dụng các chức năng như: thêm, tìm, sửa thông tin và xóa xe.
Luồng sự kiện chính	1. Người dùng đăng nhập vào hệ thống. 2. Người dùng tìm chức năng quản lý xe. 3. Người dùng chọn một trong các chức năng sau để thực hiện:

	3.1 Tìm xe 3.2 Thêm xe 3.3 Sửa thông tin xe 3.4 Xóa xe 4.Hệ thống hiển thị giao diện chức năng người dùng đã chọn.
Luồng sự kiện phụ (ngoại lệ)	

#### 2.3.5 Mô hình UC quản lý đặt vé



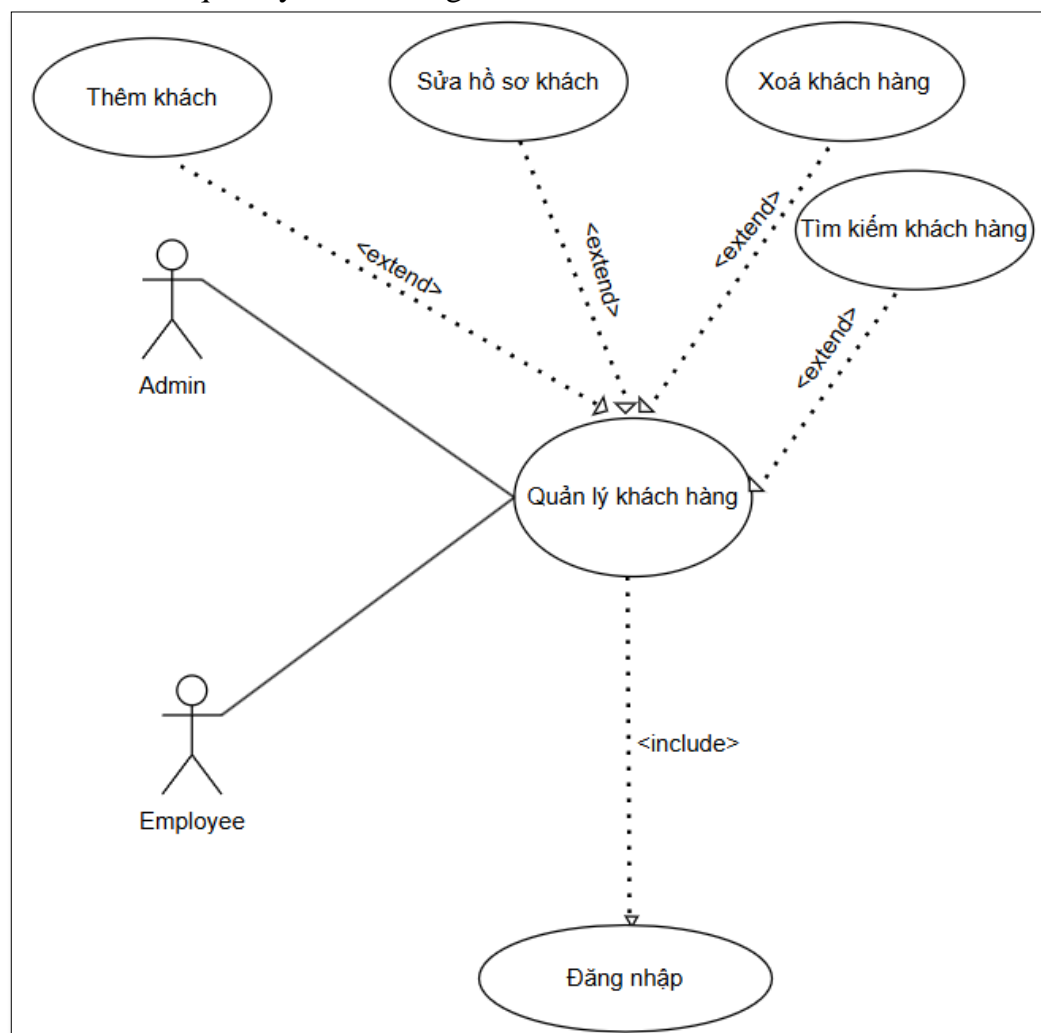
Hình 2. 6 UC quản lý đặt vé

Bảng 2. 9 Đặc tả UC quản lý đặt vé

Tên UC	Quản lý đặt vé
Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống và chọn chức năng quản lý đặt vé. Người

	dùng có thể sử dụng các chức năng như: thêm, tìm, sửa thông tin và xóa đặt vé.
Luồng sự kiện chính	1. Người dùng đăng nhập vào hệ thống. 2. Người dùng tìm chức năng quản lý đặt vé. 3. Người dùng chọn một trong các chức năng sau để thực hiện: 3.1 Tìm vé 3.2 Thêm vé 3.3 Sửa thông tin vé 3.4 Xóa vé 4. Hệ thống hiển thị giao diện chức năng người dùng đã chọn.
Luồng sự kiện phụ (ngoại lệ)	

### 2.3.6 Mô hình UC quản lý khách hàng



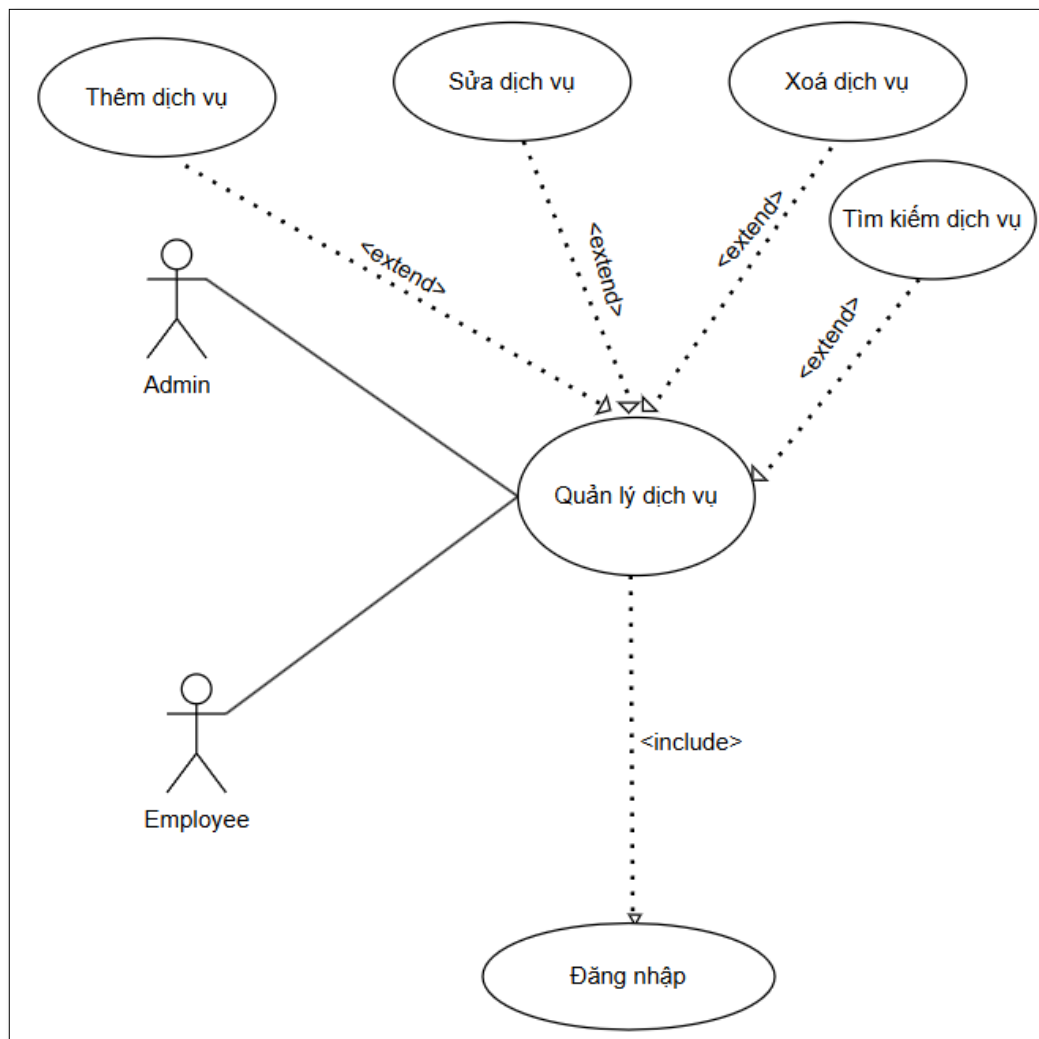
Hình 2. 7 UC quản lý khách hàng

Bảng 2. 10 Đặc tả UC quản lý khách hàng

Tên UC	Quản lý khách hàng
--------	--------------------

Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống và chọn chức năng quản lý khách hàng. Người dùng có thể sử dụng các chức năng như: thêm, tìm, sửa thông tin và xóa khách hàng.
Luồng sự kiện chính	1. Người dùng đăng nhập vào hệ thống. 2. Người dùng tìm chức năng quản lý khách hàng. 3. Người dùng chọn một trong các chức năng sau để thực hiện: 3.1 Tìm khách hàng 3.2 Thêm khách hàng 3.3 Sửa thông tin khách hàng 3.4 Xóa khách hàng 4. Hệ thống hiển thị giao diện chức năng người dùng đã chọn.
Luồng sự kiện phụ (ngoại lệ)	

### 2.3.7 Mô hình UC quản lý dịch vụ



Hình 2. 8 UC quản lý dịch vụ

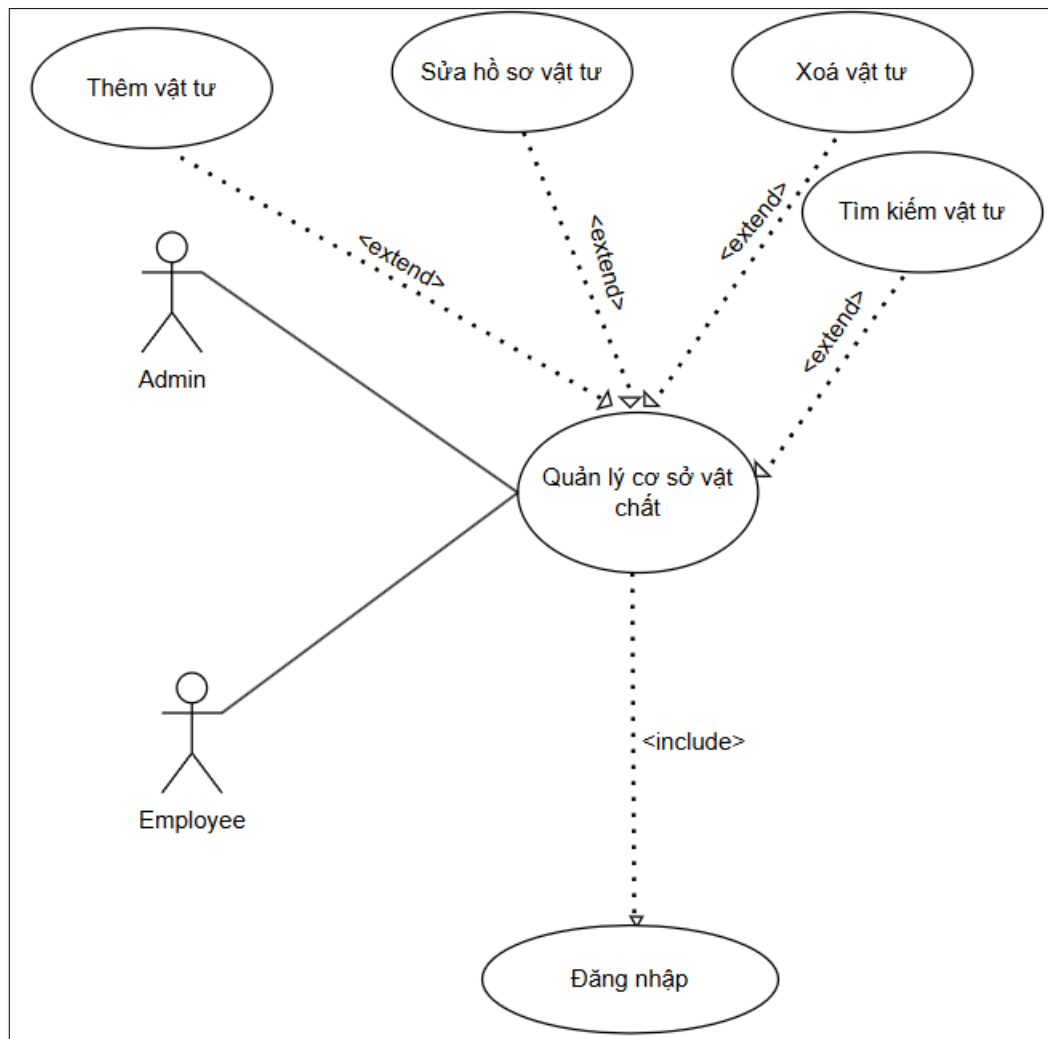
Bảng 2. 11 Đặc tả UC quản lý dịch vụ

Tên UC	Quản lý dịch vụ
Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống và chọn chức năng quản lý dịch vụ. Người dùng có thể sử dụng các chức năng như: thêm, tìm, sửa thông tin và xóa dịch vụ.
Luồng sự kiện chính	1. Người dùng đăng nhập vào hệ thống. 2. Người dùng tìm chức năng quản lý dịch vụ. 3. Người dùng chọn một trong các chức năng sau để thực hiện: 3.1 Tìm dịch vụ



	3.2 Thêm dịch vụ 3.3 Sửa thông tin dịch vụ 3.4 Xóa dịch vụ 4.Hệ thống hiển thị giao diện chức năng người dùng đã chọn.
Luồng sự kiện phụ (ngoại lệ)	

#### 2.3.8 Mô hình UC quản lý cơ sở vật chất



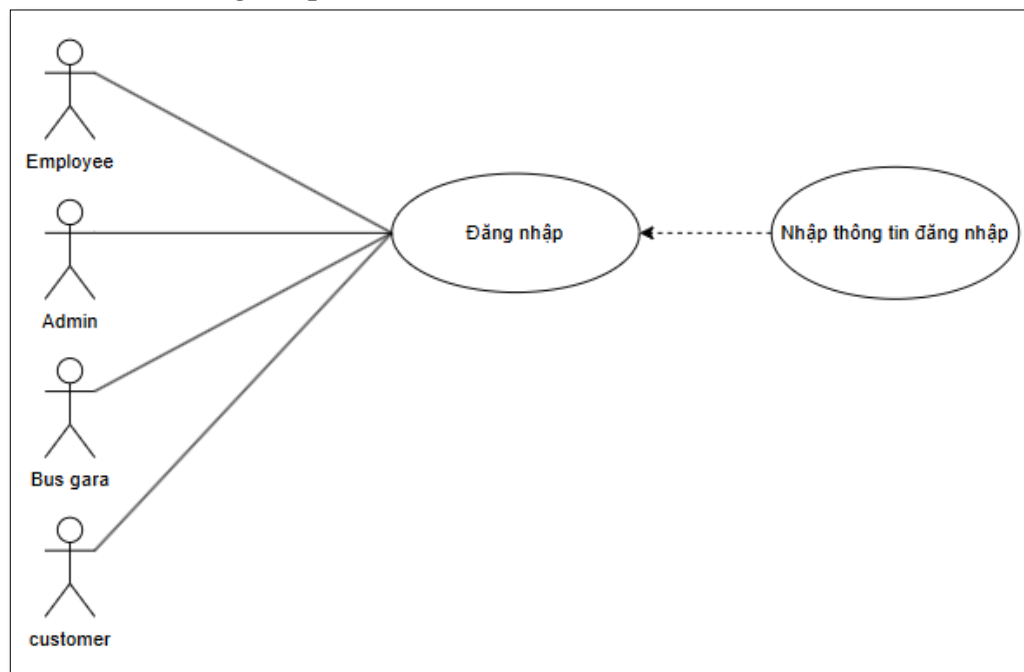
Hình 2. 9 UC quản lý cơ sở vật chất

Bảng 2. 12 Đặc tả UC quản lý cơ sở vật chất

Tên UC	Quản lý cơ sở vật chất
Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên)
Điều kiện	Người dùng đã đăng nhập thành công vào hệ thống.
Mô tả	Người dùng truy cập vào hệ thống và chọn chức năng quản lý cơ sở vật chất. Người dùng có thể sử dụng các chức năng

	nếu: thêm, tìm, sửa thông tin và xóa cơ sở vật chất.
Luồng sự kiện chính	1. Người dùng đăng nhập vào hệ thống. 2. Người dùng tìm chức năng quản lý cơ sở vật chất. 3. Người dùng chọn một trong các chức năng sau để thực hiện: 3.1 Tìm vật tư 3.2 Thêm vật tư 3.3 Sửa thông tin vật tư 3.4 Xóa vật tư 4. Hệ thống hiển thị giao diện chức năng người dùng đã chọn.
Luồng sự kiện phụ (ngoại lệ)	

### 2.3.9 Mô hình UC đăng nhập



Hình 2. 10 UC đăng nhập

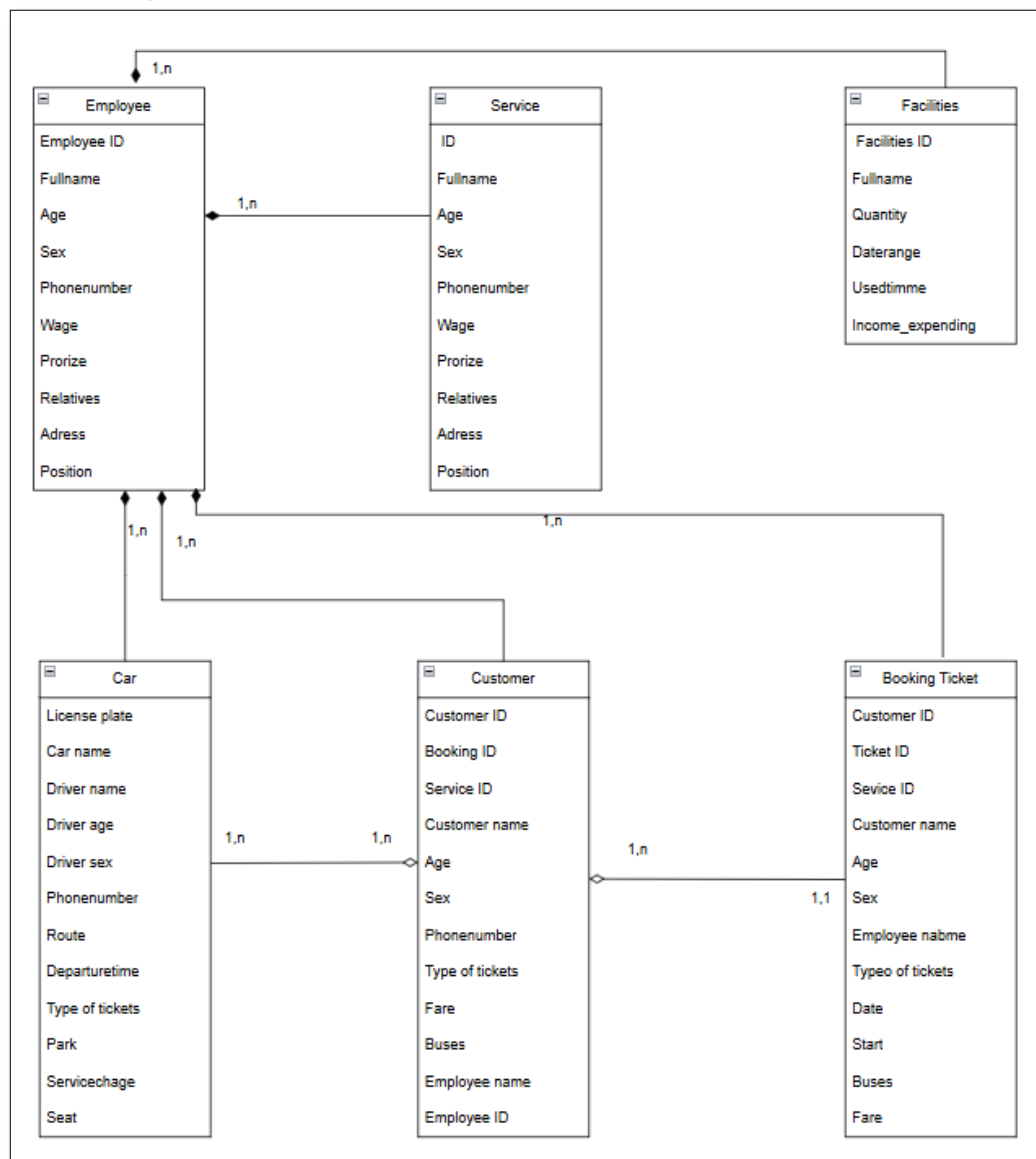
Bảng 2. 13 Đặc tả UC đăng nhập

Tên UC	Đăng nhập
Tác nhân (Actor)	Admin (Quản lý), Employee (Nhân viên), Customer (Khách hàng)
Điều kiện	Người dùng đã có tài khoản đăng nhập.
Mô tả	Người dùng cần nhập đủ tên đăng nhập và mật khẩu.
Luồng sự kiện chính	1. Người sử dụng chọn chức năng “Đăng nhập” 2. Hệ thống hiển thị form “Đăng nhập”.

	<p>3. Người dùng nhập “Tên đăng nhập” và “mật khẩu”, gửi thông tin đến hệ thống.</p> <p>4. Hệ thống kiểm tra. Nếu đúng thì cho phép truy cập và sử dụng các chức năng theo đúng quyền hạn. Nếu sai hệ thống báo lỗi và yêu cầu người dùng nhập lại thông tin.</p>
Luồng sự kiện phụ (ngoại lệ)	

## 2.3 Mô hình hoá khái niệm

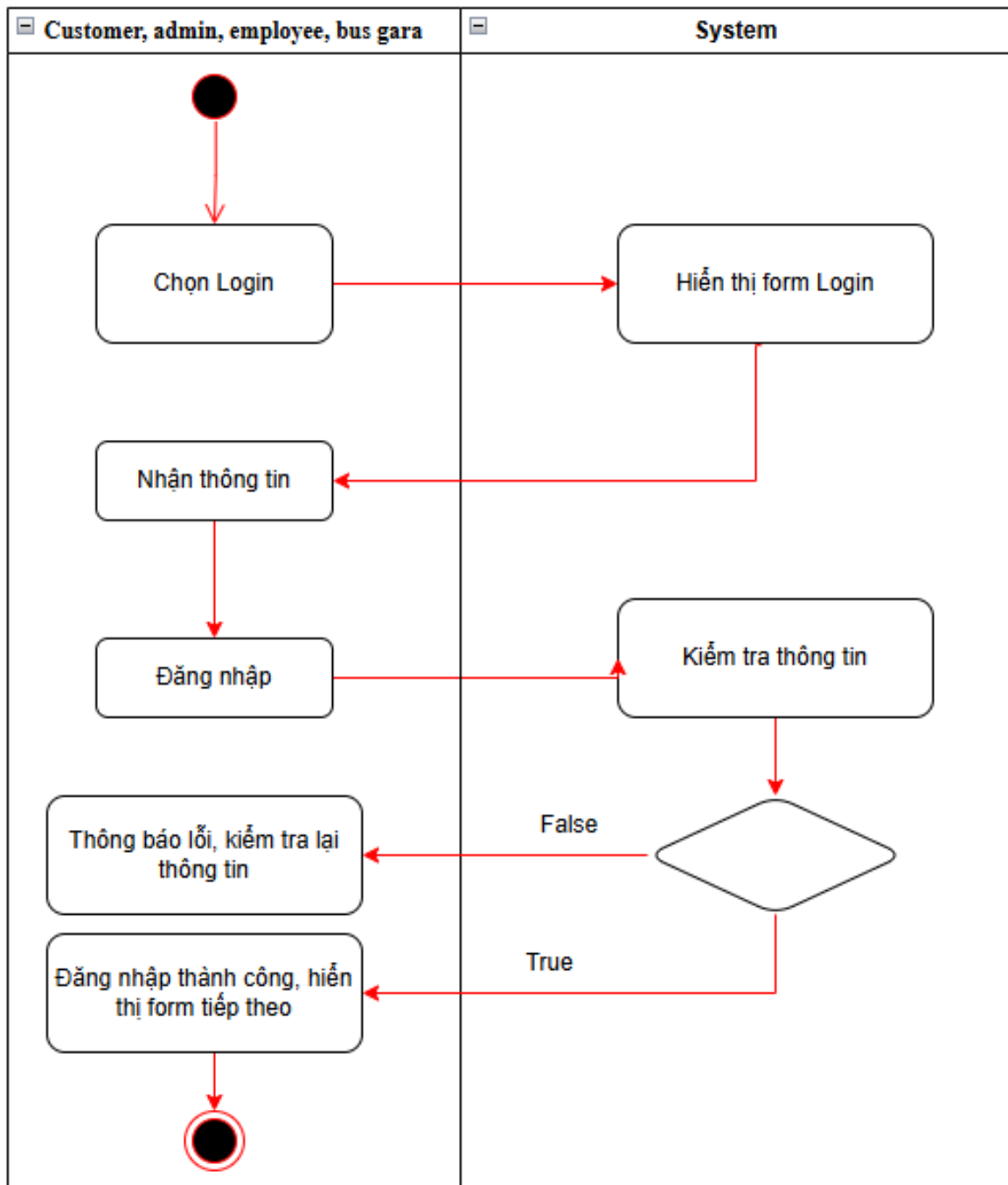
### 2.3.1 Class Diagram



Hình 2. 11 Class Diagram

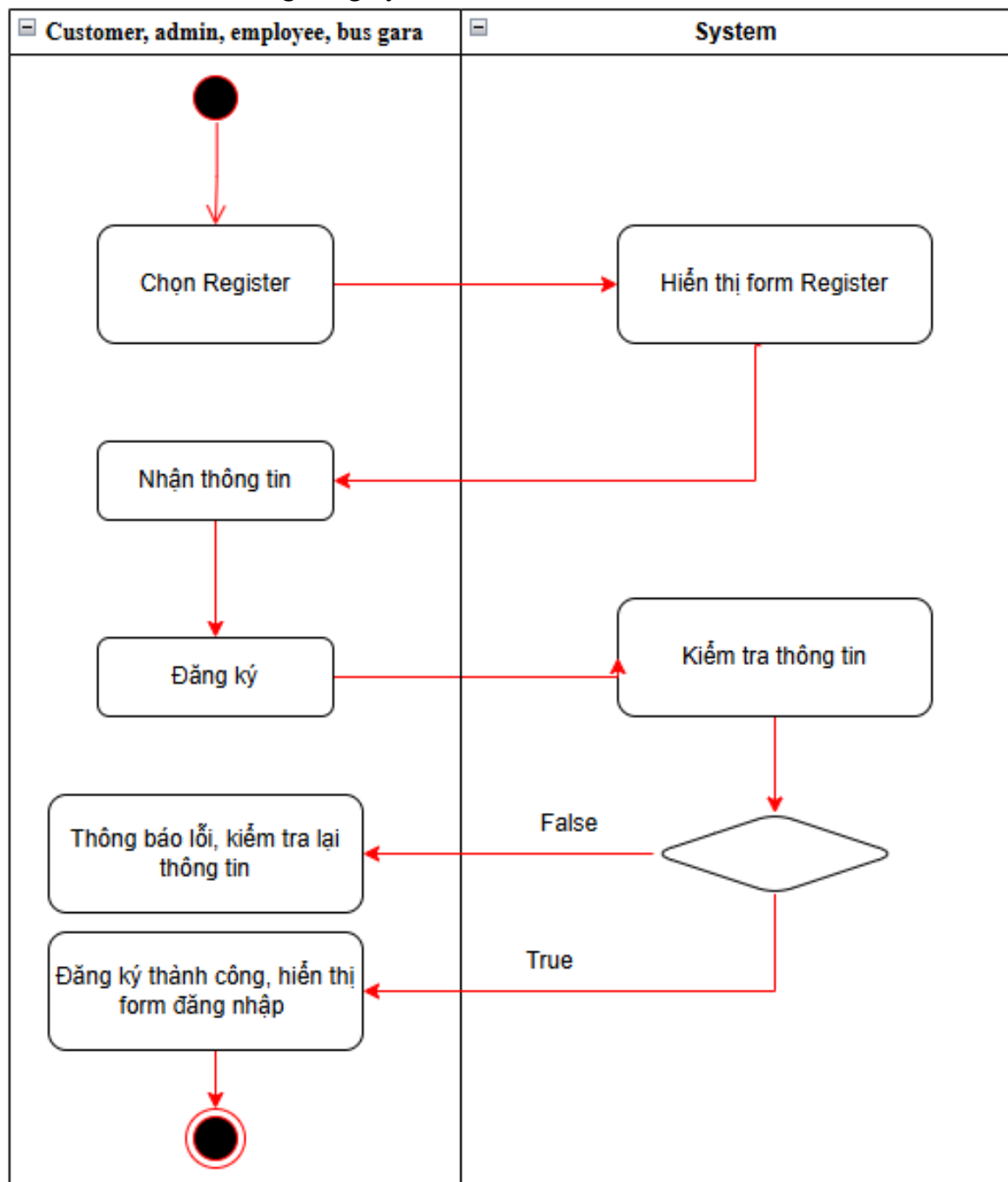
## 2.3.2 Activity Diagram

### 2.3.2.1 Biểu đồ hoạt động đăng nhập



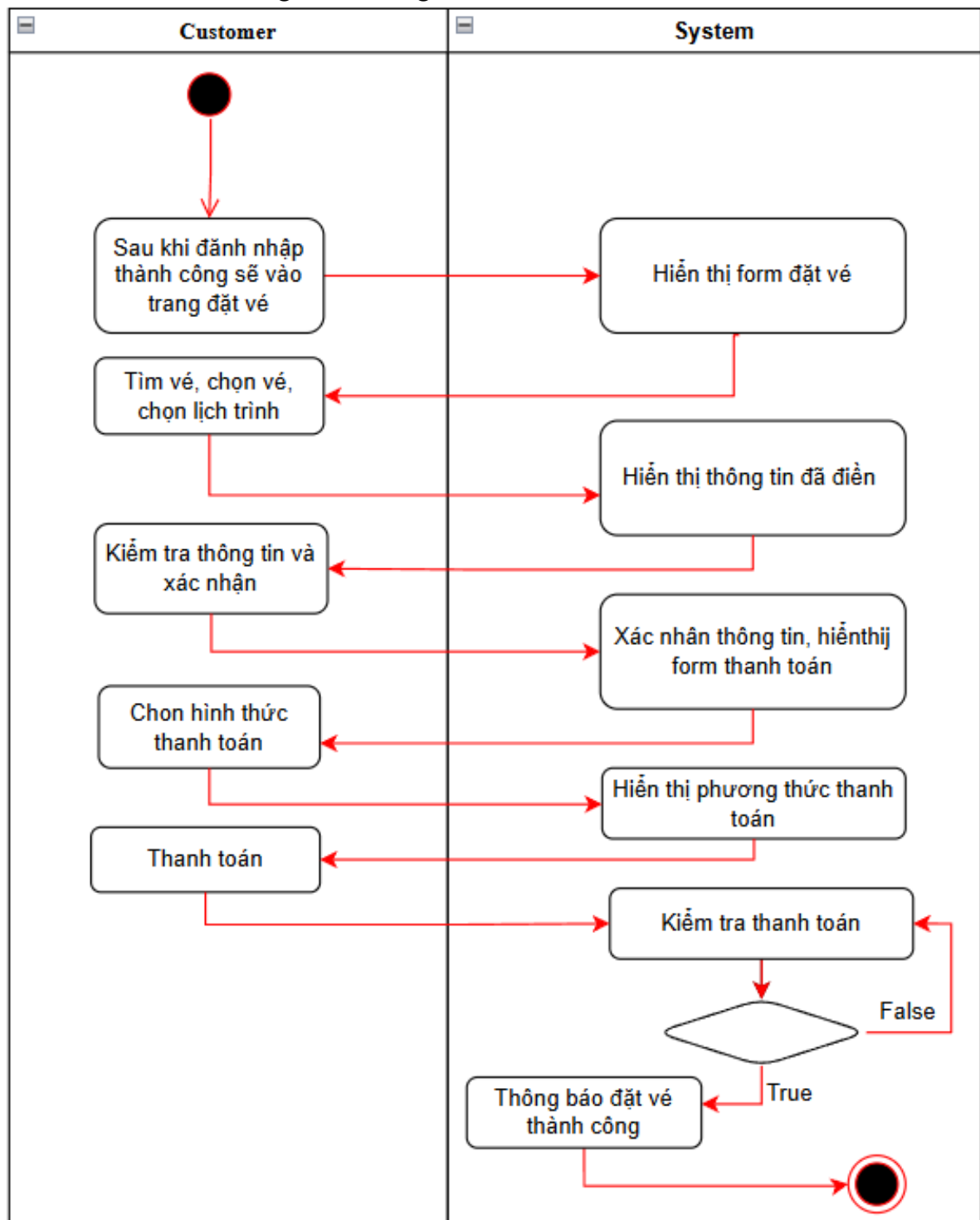
Hình 2. 12 Biểu đồ hoạt động đăng nhập

### 2.3.2.2 Biểu đồ hoạt động đăng ký



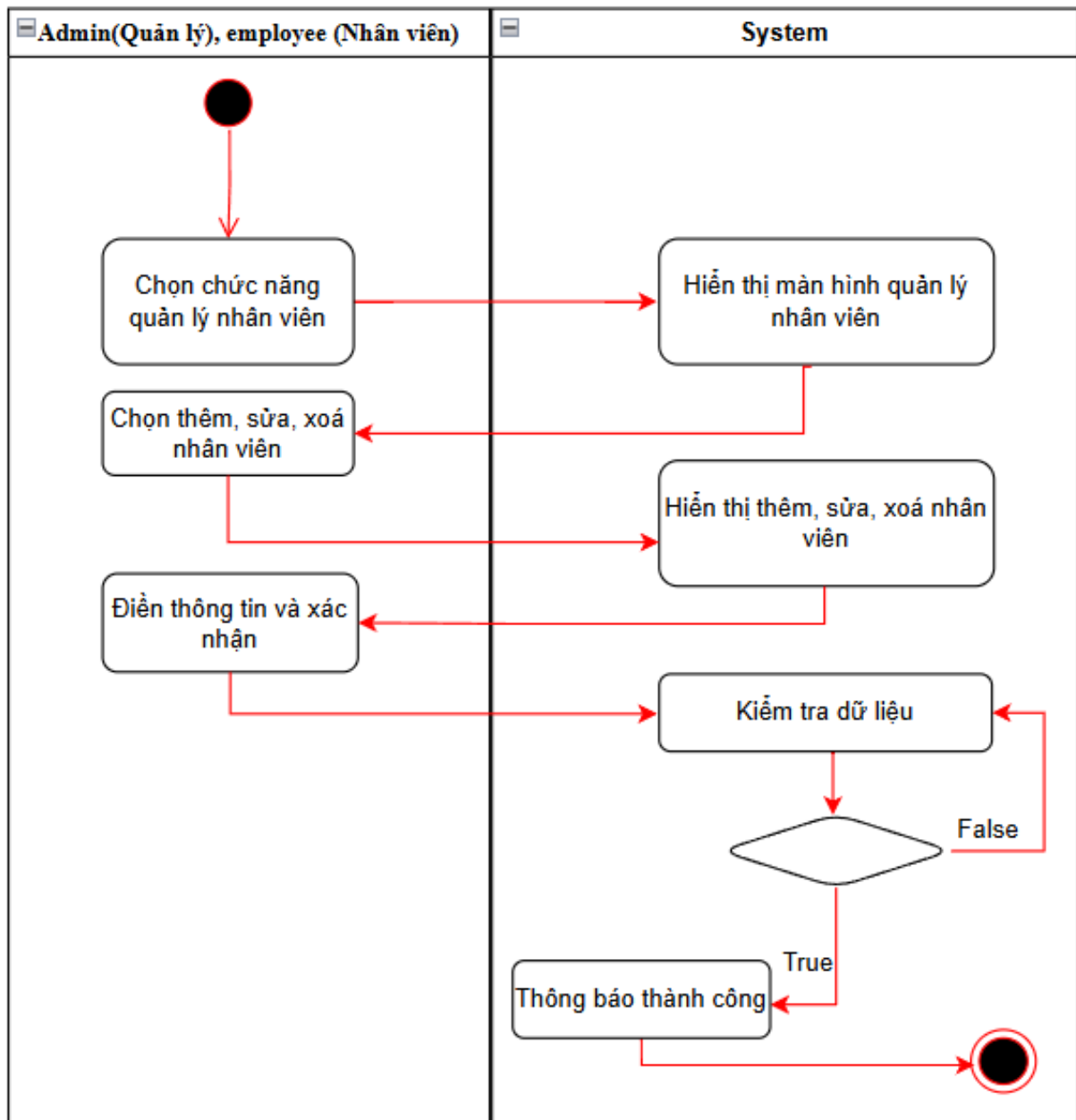
Hình 2. 13 Biểu đồ hoạt động đăng ký

### 2.3.2.3 Biểu đồ hoạt động khách hàng đặt vé



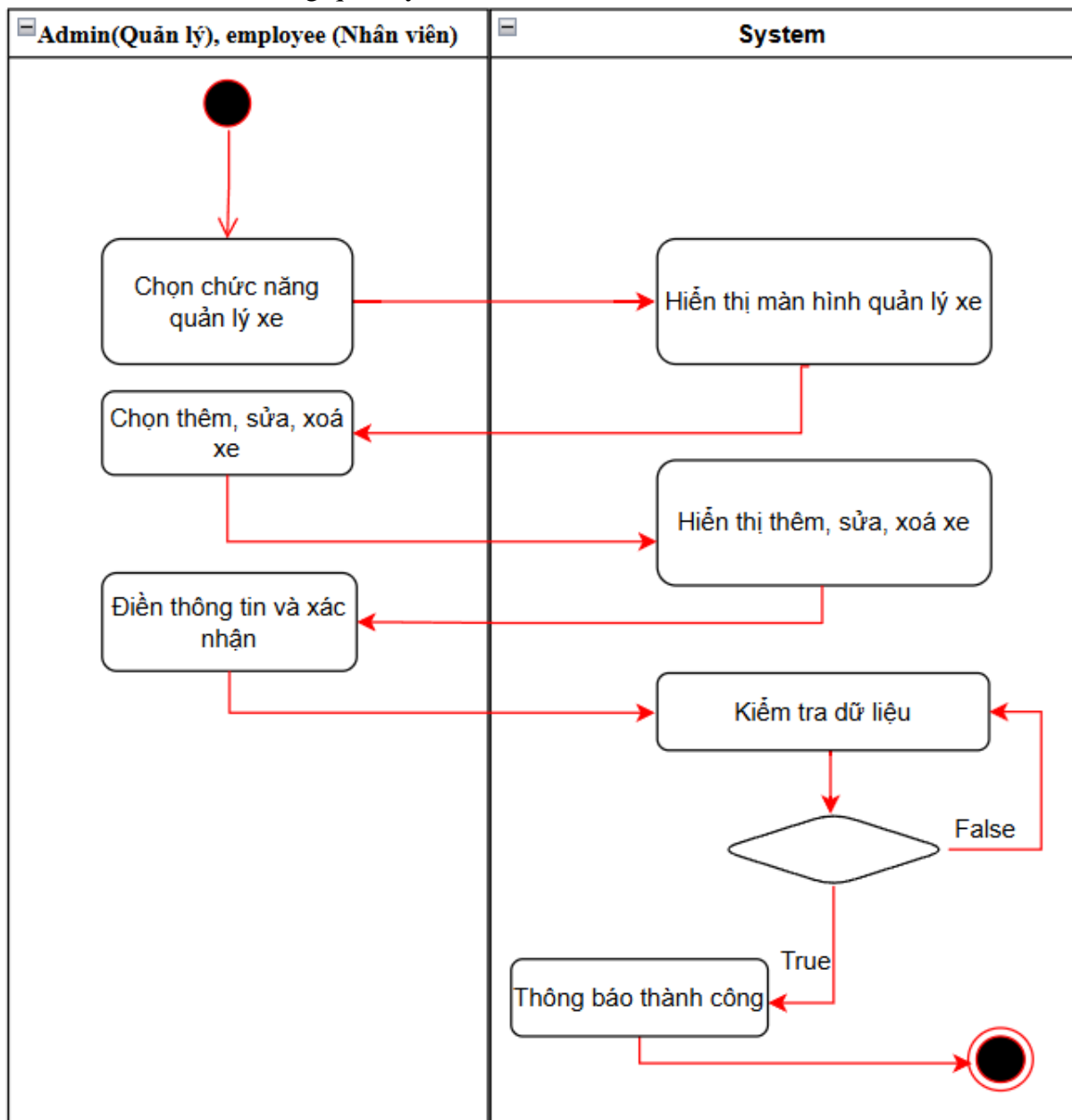
Hình 2. 14 Biểu đồ hoạt động khách hàng đặt vé

#### 2.3.2.4 Biểu đồ hoạt động quản lý nhân viên



Hình 2. 15 Biểu đồ hoạt động quản lý nhân viên

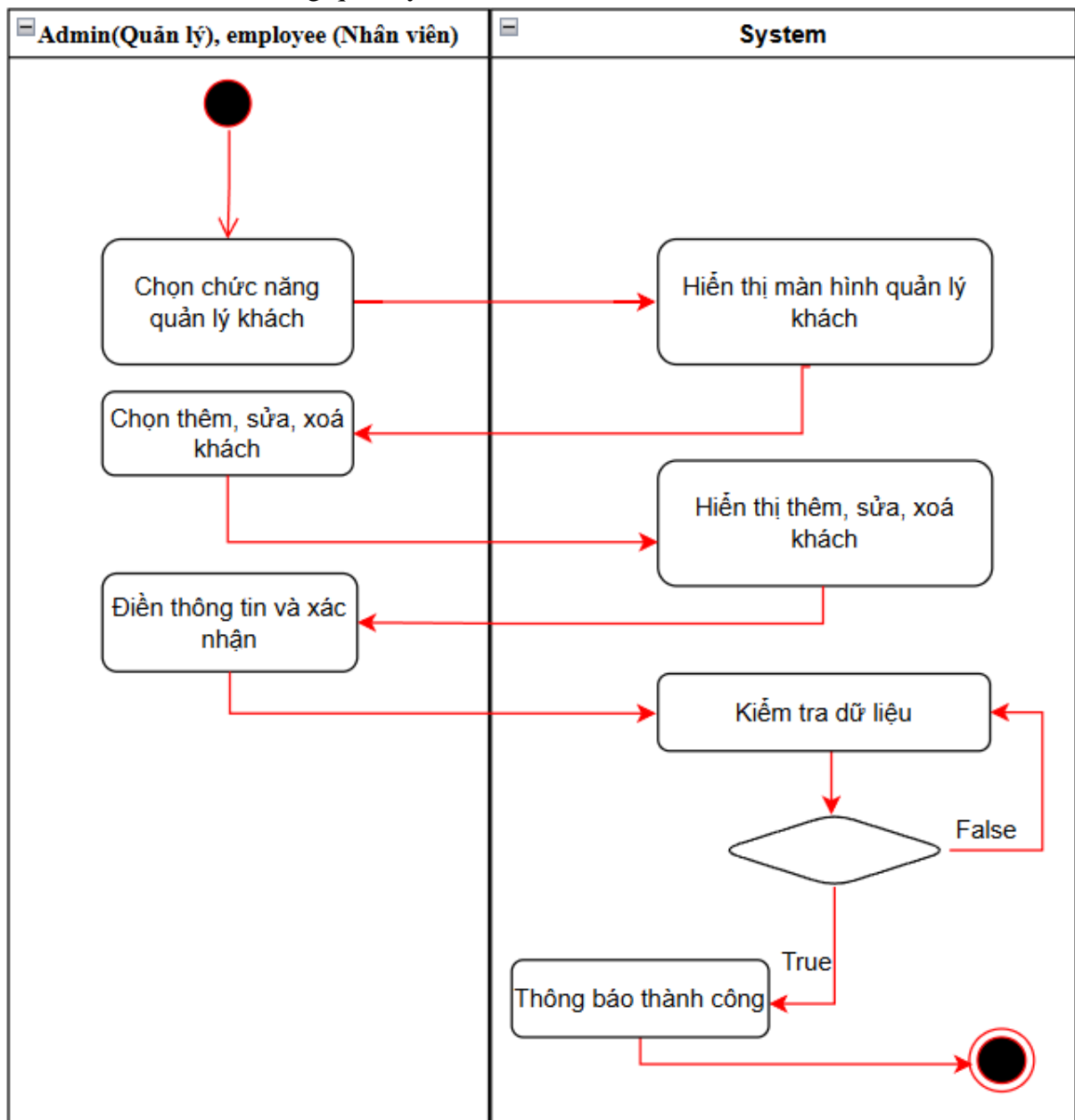
### 2.3.2.5 Biểu đồ hoạt động quản lý xe



Hình 2. 16 Biểu đồ hoạt động quản lý xe

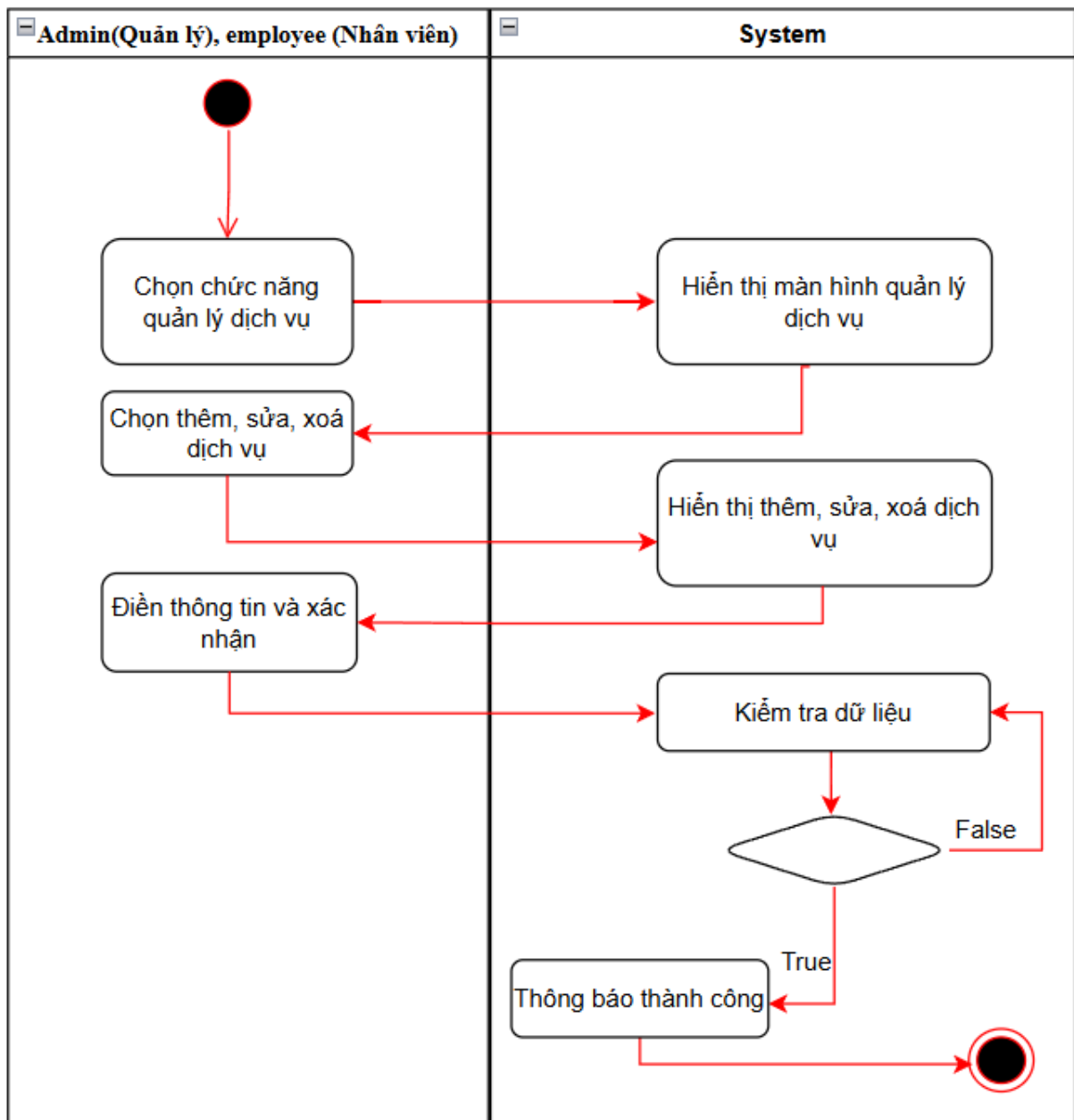


### 2.3.2.6 Biểu đồ hoạt động quản lý khách



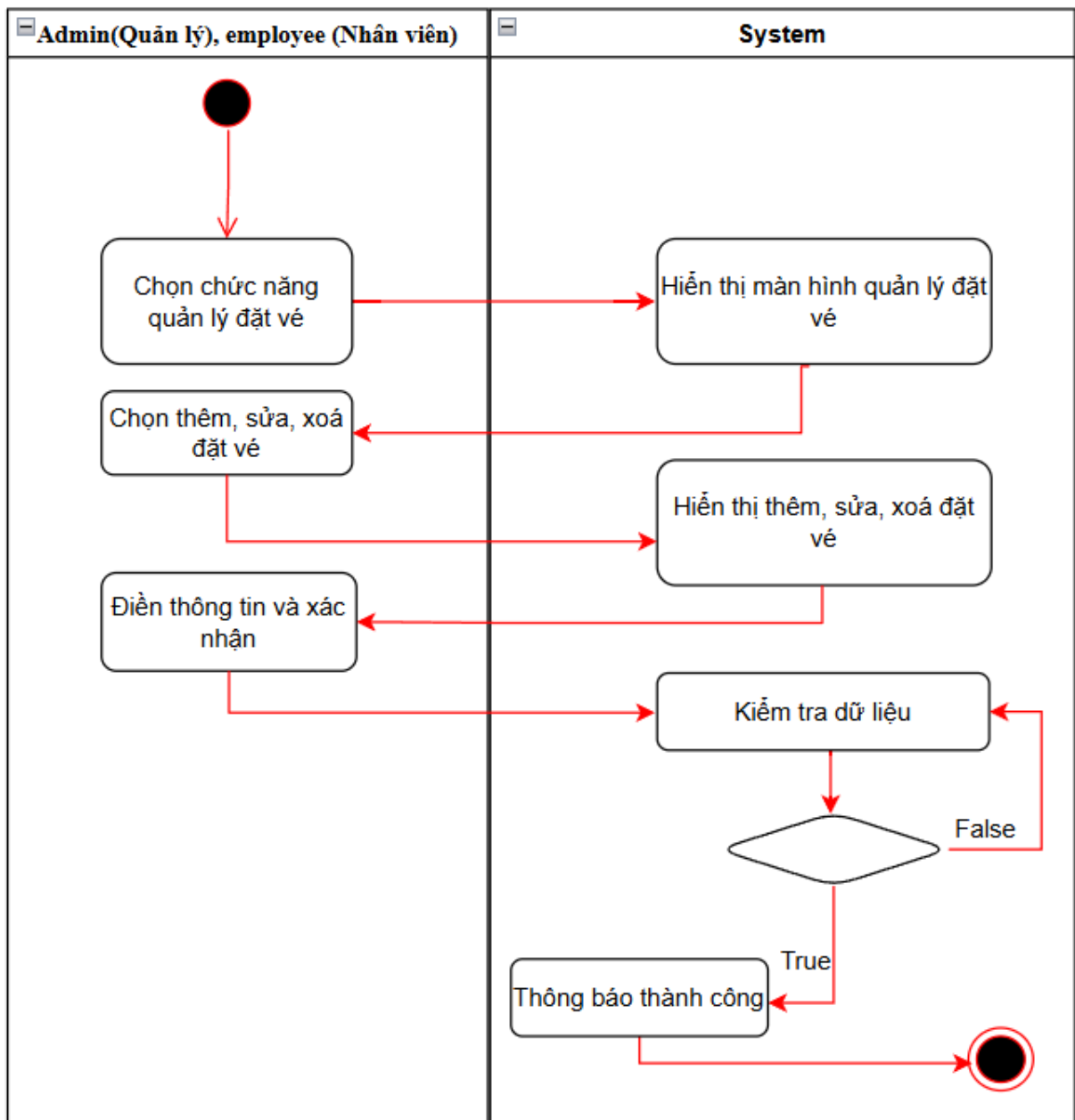
Hình 2. 17 Biểu đồ hoạt động quản lý khách

### 2.3.2.7 Biểu đồ hoạt động quản lý dịch vụ



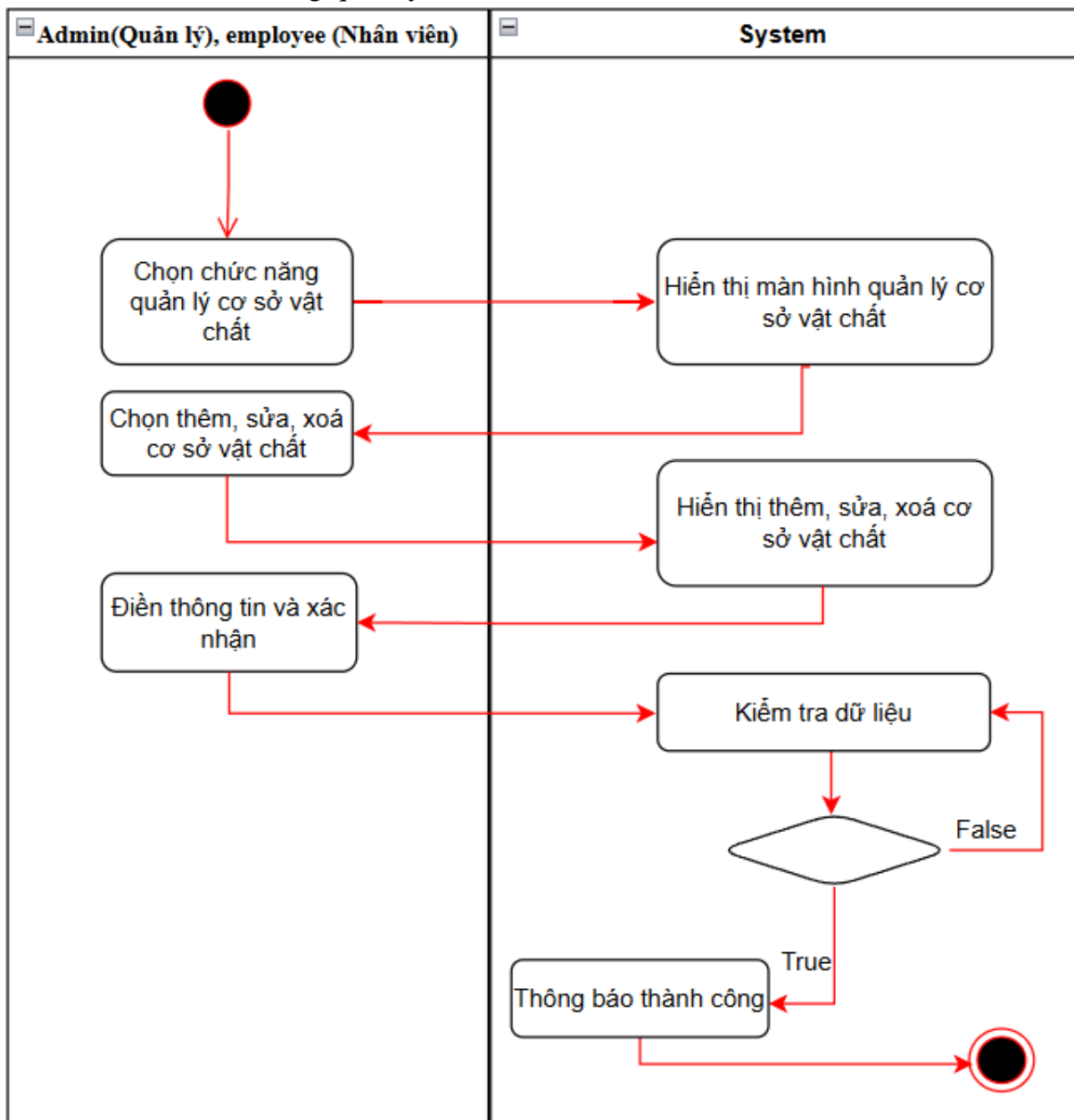
Hình 2. 18 Biểu đồ hoạt động quản lý dịch vụ

### 2.3.2.8 Biểu đồ hoạt động quản lý đặt vé



Hình 2. 19 Biểu đồ hoạt động quản lý đặt vé

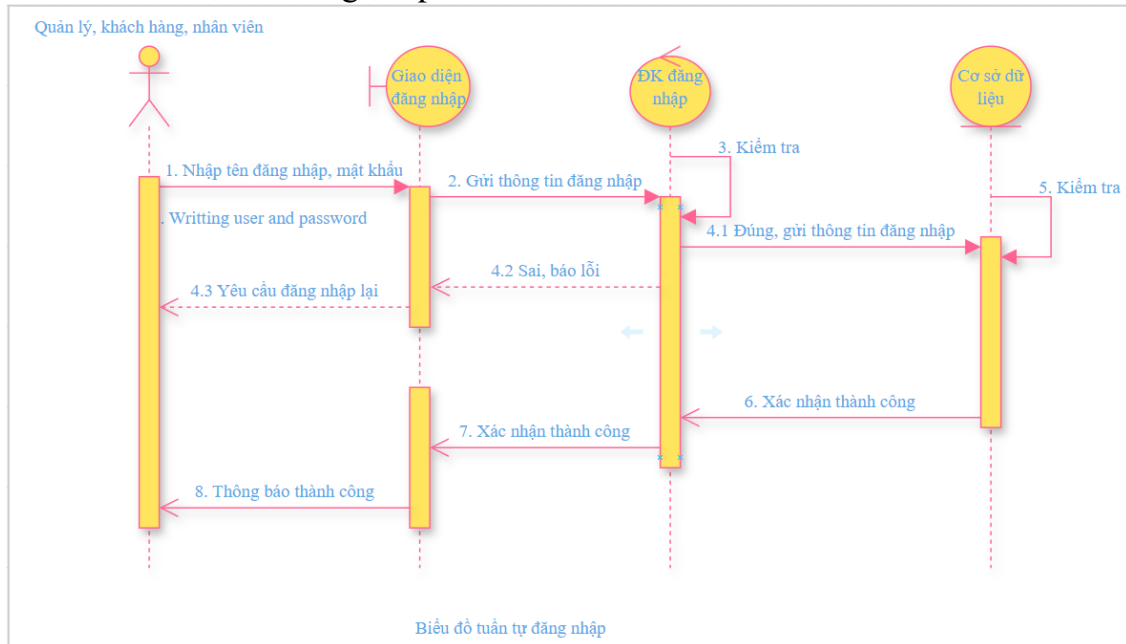
### 2.3.2.9 Biểu đồ hoạt động quản lý cơ sở vật chất



Hình 2. 20 Biểu đồ hoạt động quản lý cơ sở vật chất

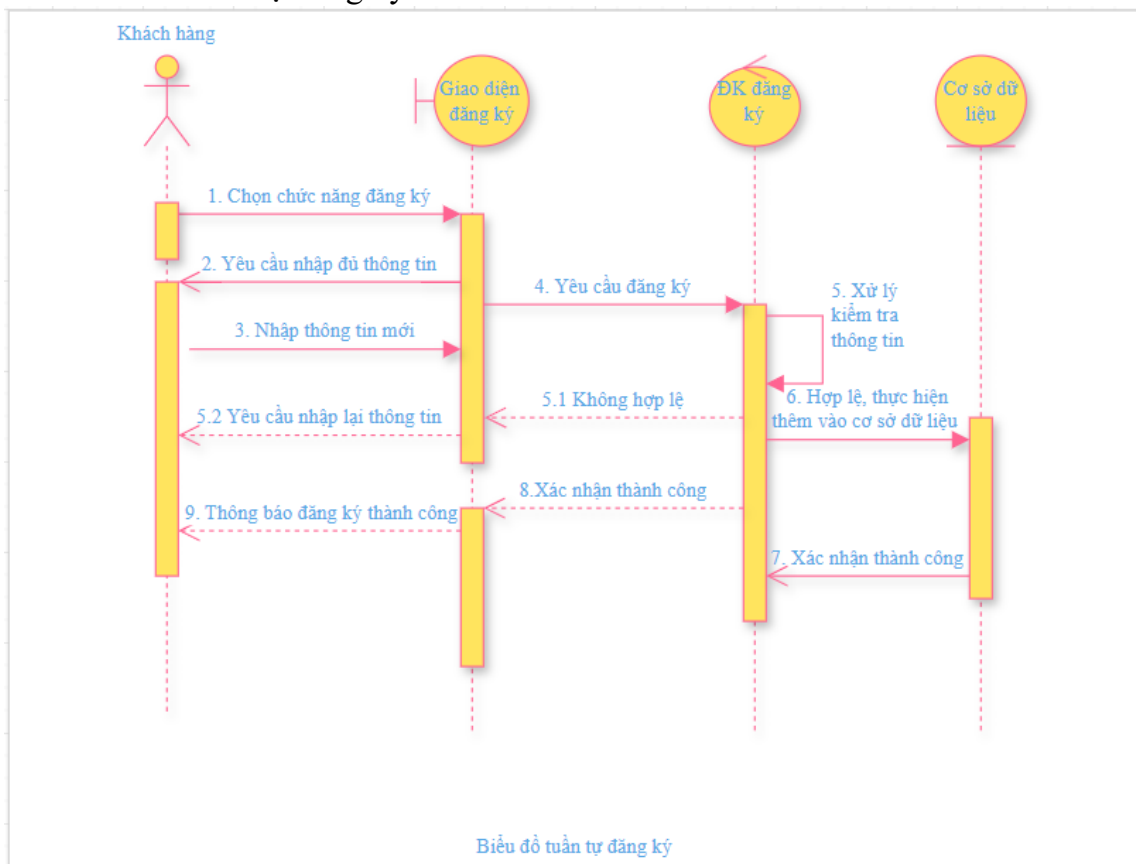
### 2.3.3 Sequence Diagram

#### 2.3.3.1 Biểu đồ tuần tự đăng nhập



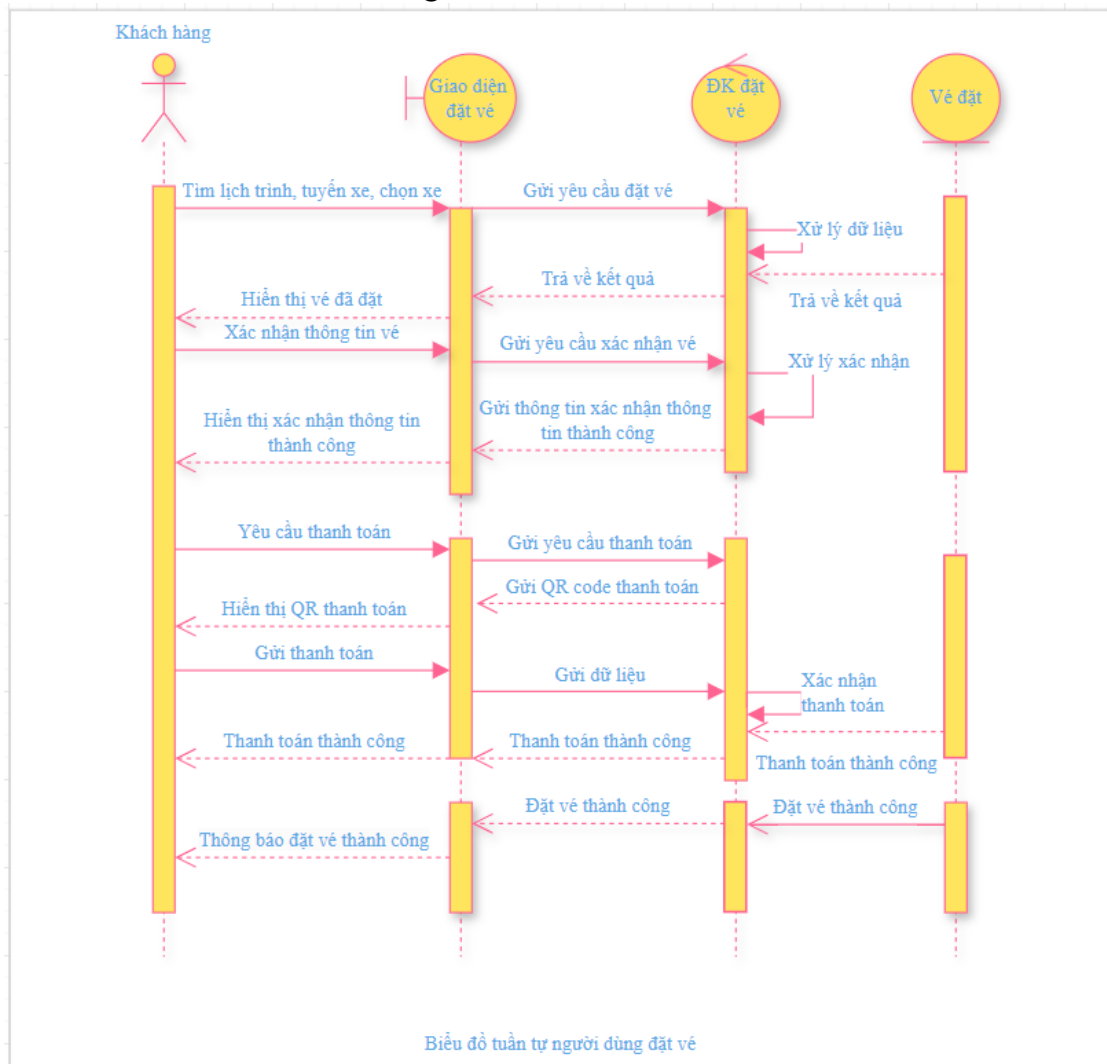
Hình 2. 21 Biểu đồ tuần tự đăng nhập

#### 2.3.3.2 Biểu đồ tuần tự đăng ký



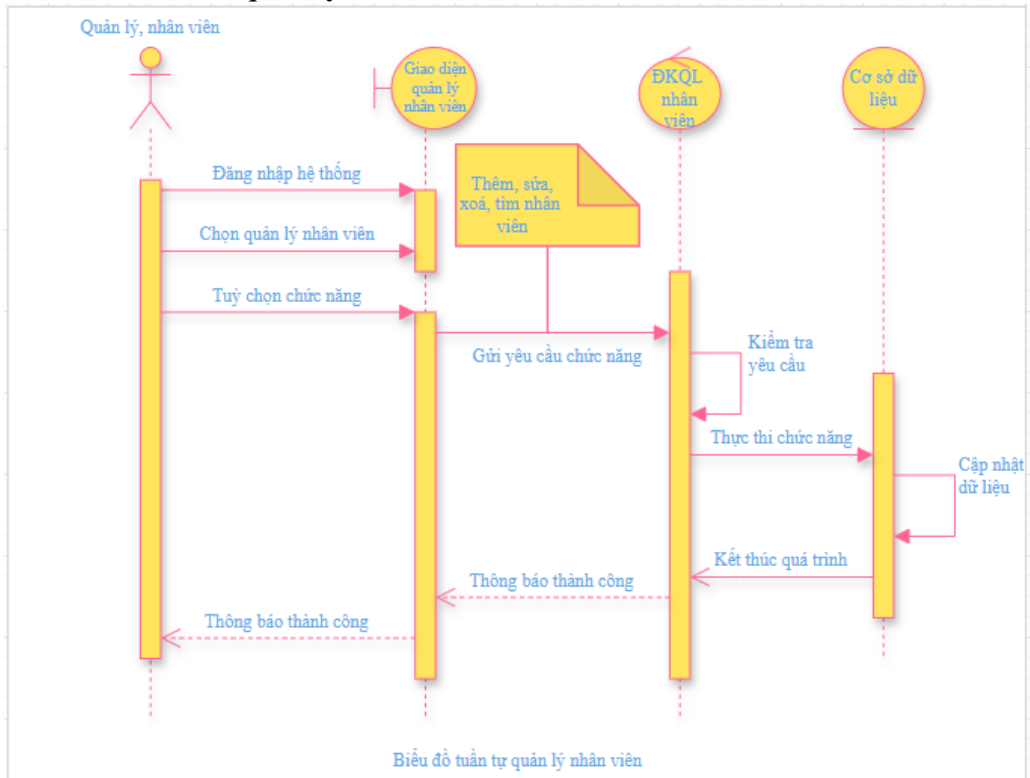
Hình 2. 22 Biểu đồ tuần tự đăng ký

### 2.3.3.3 Biểu đồ tuần tự khách hàng đặt vé



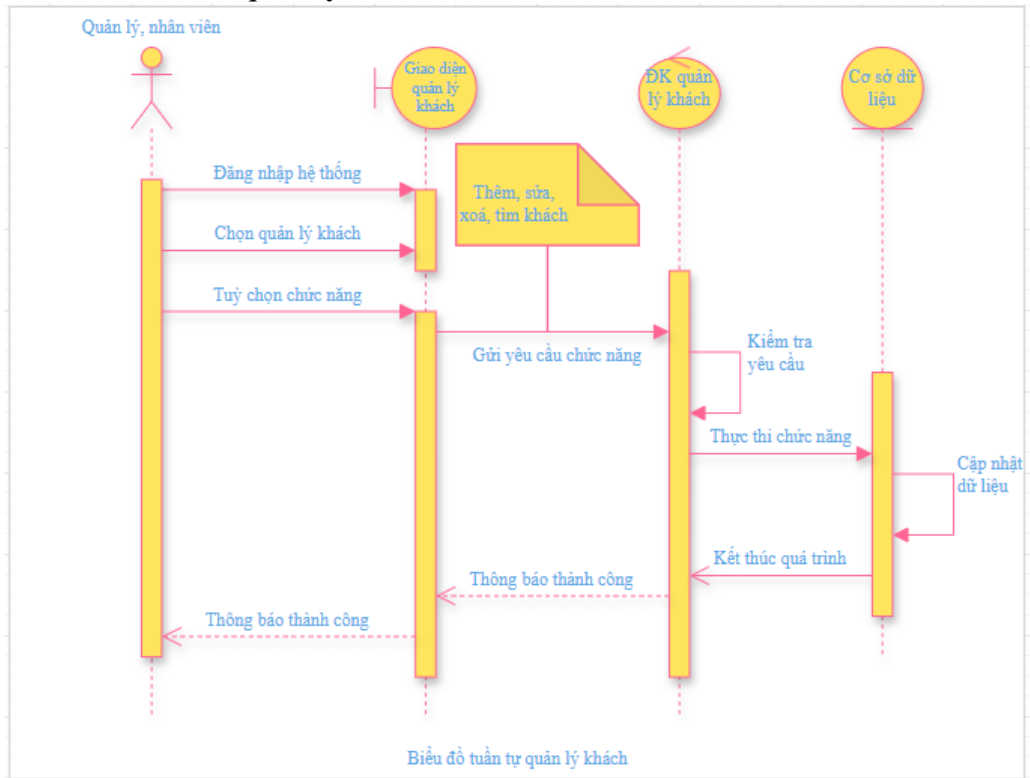
Hình 2. 23 Biểu đồ tuần tự người dùng đặt vé

### 2.3.3.4 Biểu đồ tuần tự quản lý nhân viên



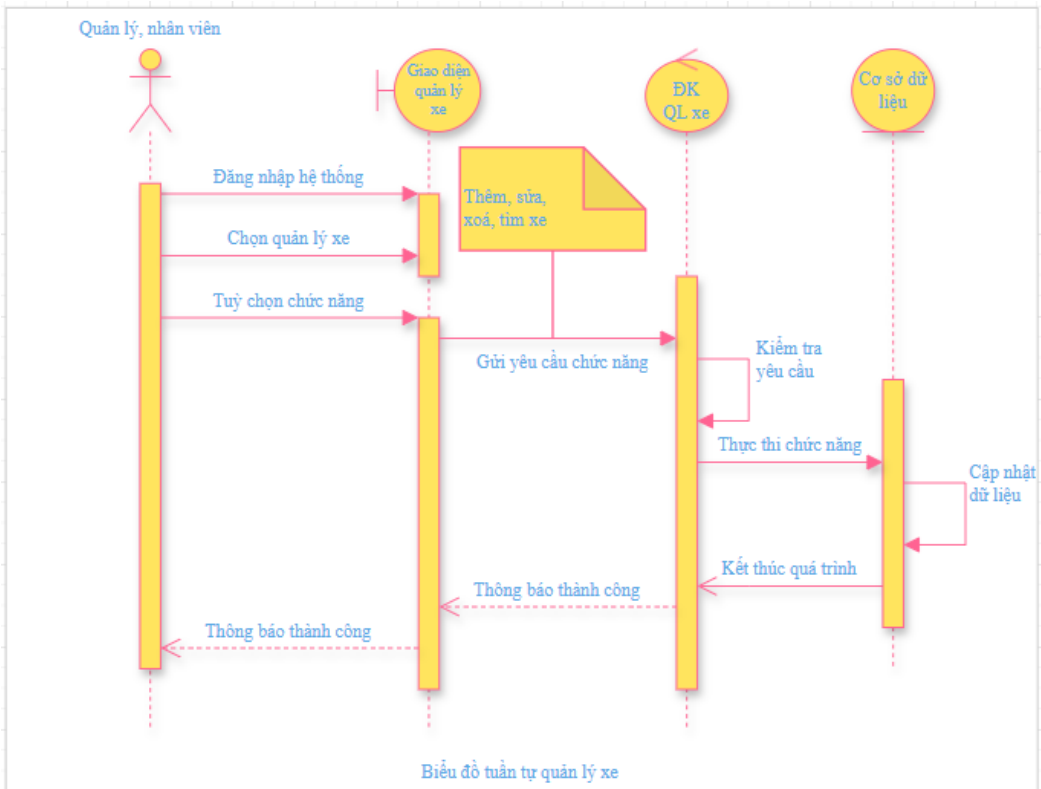
Hình 2. 24 Biểu đồ tuần tự quản lý khách

### 2.3.3.5 Biểu đồ tuần tự quản lý khách



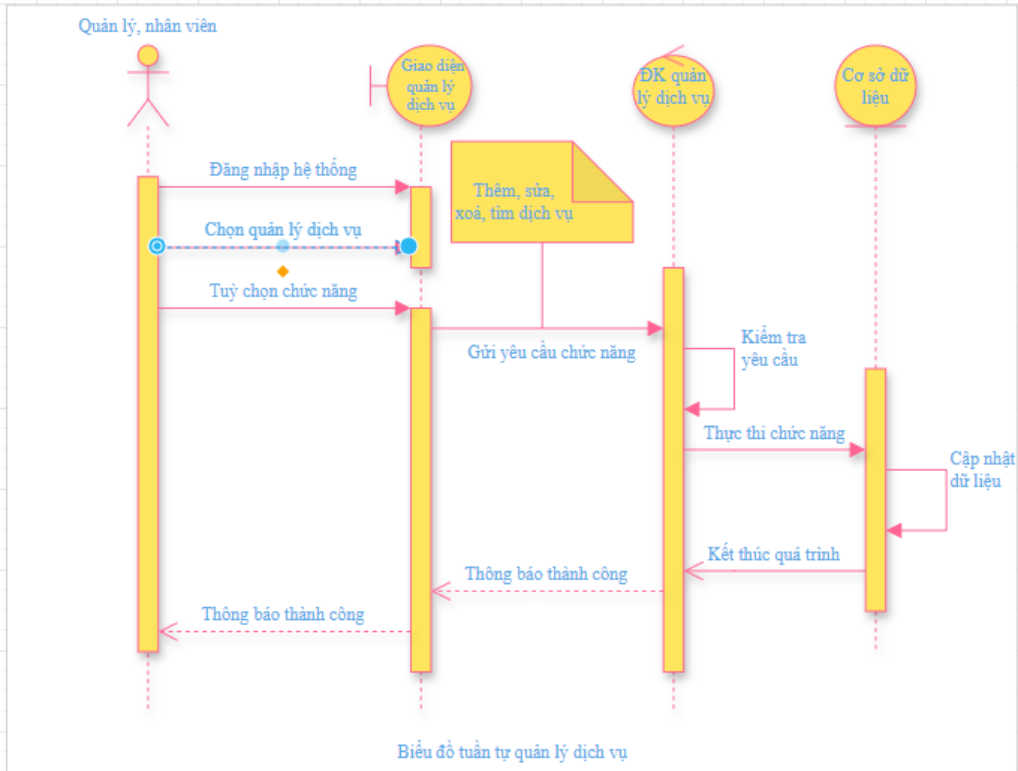
Hình 2. 25 Biểu đồ tuần tự quản lý khách

### 2.3.3.6 Biểu đồ tuần tự quản lý xe



Hình 2. 26 Biểu đồ tuần tự quản lý xe

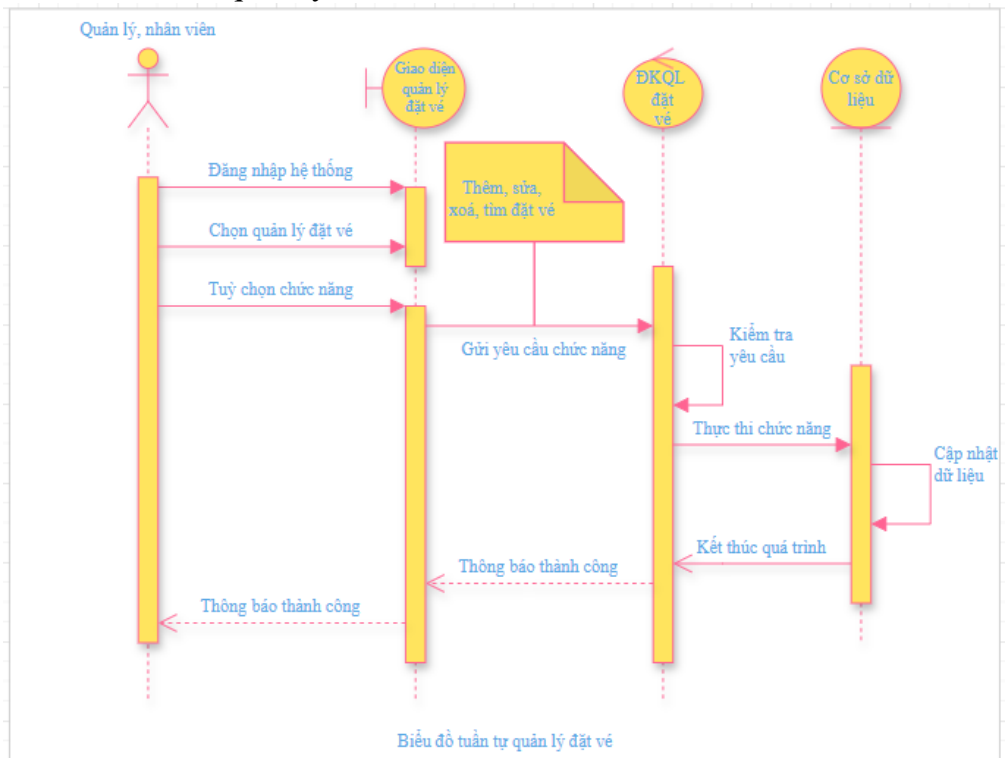
### 2.3.3.7 Biểu đồ tuần tự quản lý dịch vụ



Hình 2. 27 Biểu đồ tuần tự quản lý dịch vụ

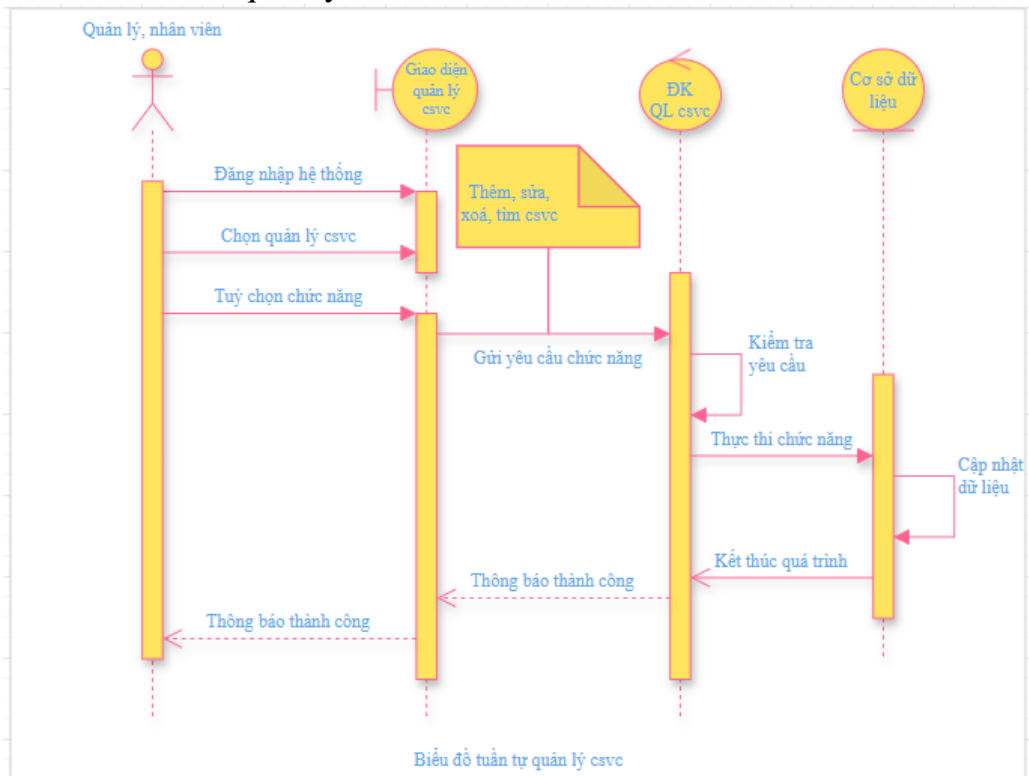


### 2.3.3.8 Biểu đồ tuần tự quản lý đặt vé



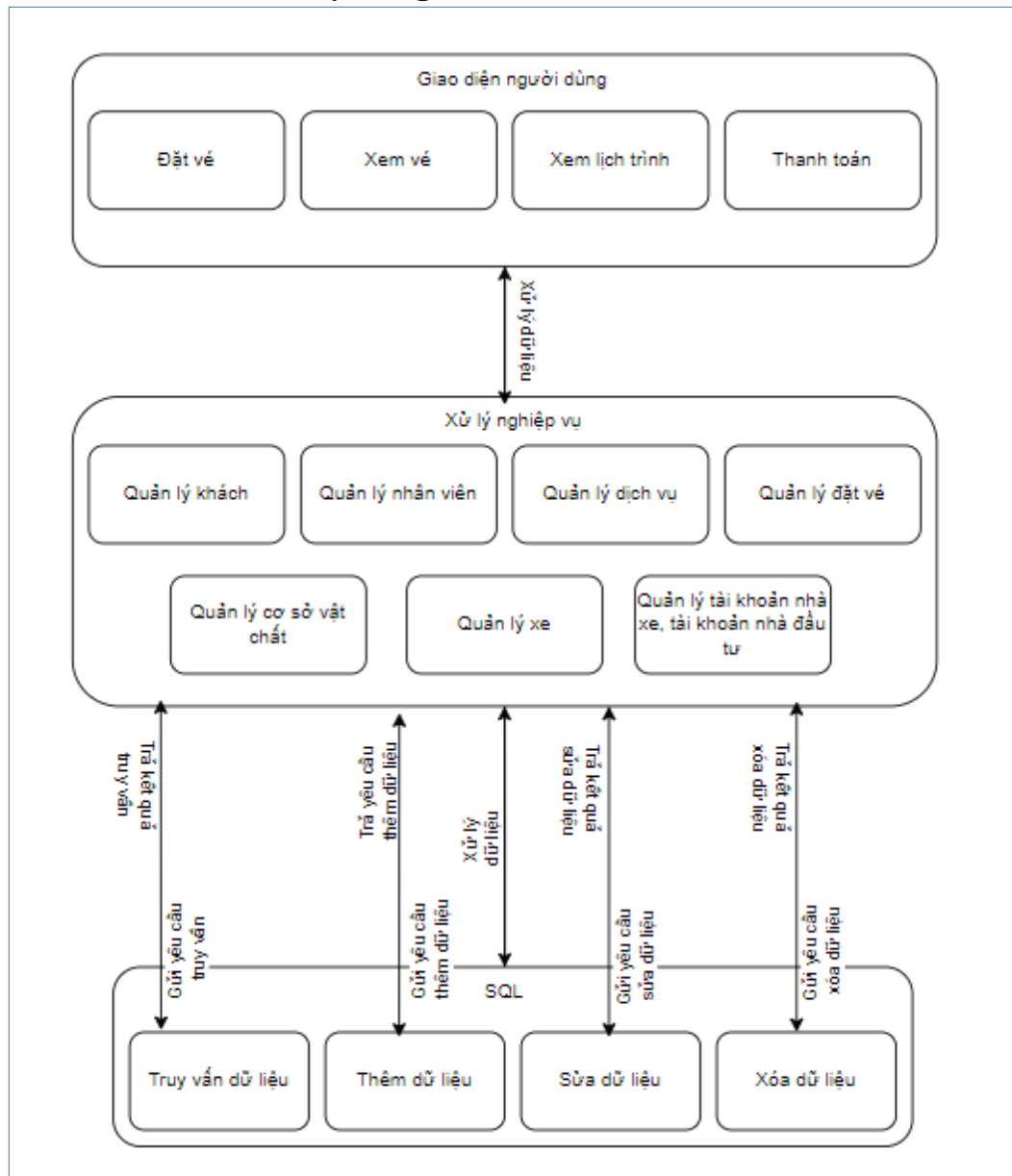
Hình 2. 28 Biểu đồ tuần tự quản lý đặt vé

### 2.3.3.9 Biểu đồ tuần tự quản lý cơ sở vật chất



Hình 2. 29 Biểu đồ tuần tự quản lý csvc

## 2.4 Mô hình hoá kiến trúc hệ thống

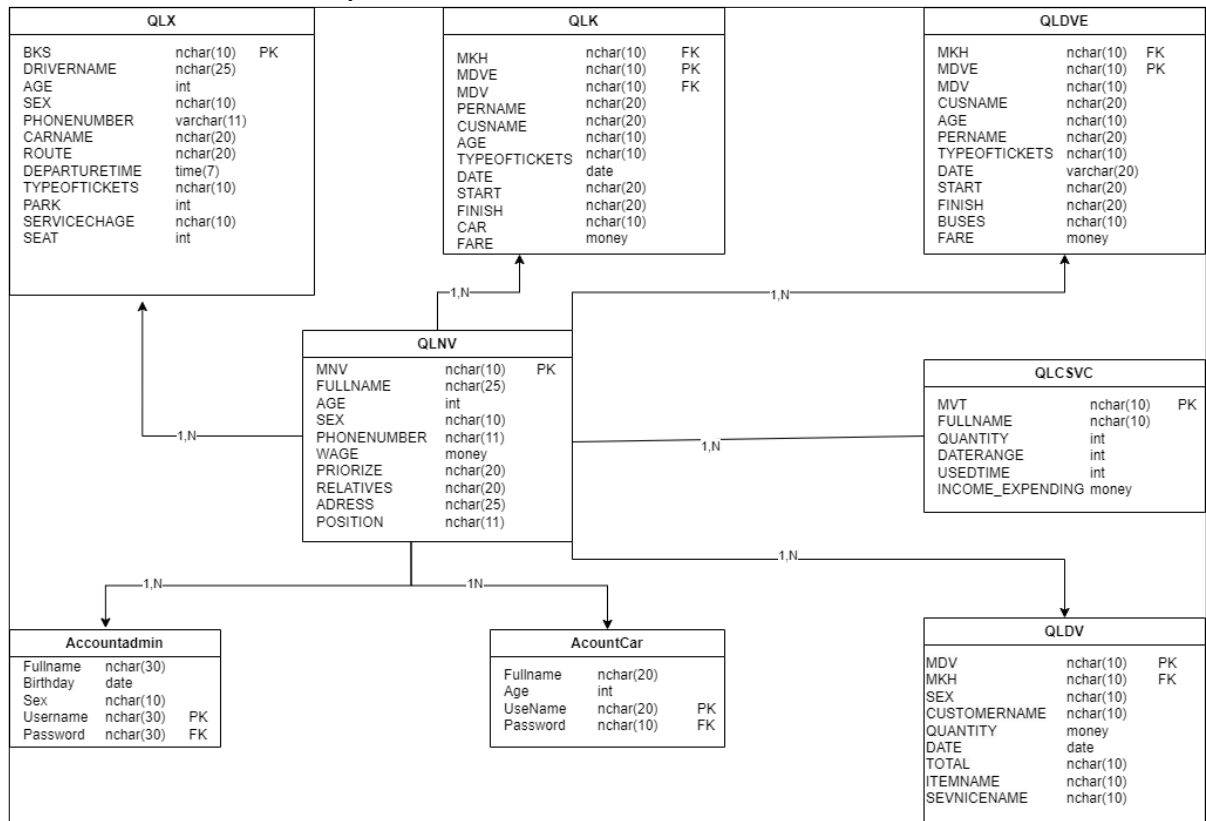


Hình 2. 30 Mô hình kiến trúc hệ thống

➤ Mô tả mô hình kiến trúc hệ thống:

- Tầng Giao Diện: Kết nối với người dùng (hành khách, tài xế, nhân viên quản lý bến xe).
- Tầng Nghiệp Vụ: Xử lý yêu cầu, kiểm tra các điều kiện nghiệp vụ và điều phối dữ liệu.
- Tầng Dữ Liệu: Lưu trữ thông tin, hỗ trợ truy vấn dữ liệu.

## 2.5 Mô hình cơ sở dữ liệu



Hình 2. 31 Mô hình cơ sở dữ liệu

Mô tả cơ sở dữ liệu:

+ Quản lý xe:

Bảng 2. 14 Quản lý xe

Thuộc tính	Phương thức
BKS	nchar(10)
DRIVERNAME	nchar(25)
AGE	int
SEX	nchar(10)
PHONENUMBER	varchar(11)
CARNAME	nchar(20)
ROUTE	nchar(20)
DEPARTURETIME	time(7)
TYPEOFTICKETS	nchar(10)
PARK	int

SERVICECHAGE	nchar(10)
SEAT	int

+ Quản lý nhân viên:

Bảng 2. 15 Quản lý nhân viên

Thuộc tính	Phương thức
MNV	nchar(10)
FULLNAME	nchar(25)
AGE	int
SEX	nchar(10)
PHONENUMBER	nchar(11)
WAGE	money
PRIORIZE	nchar(20)
RELATIVES	nchar(20)
ADRESS	nchar(25)
POSITION	nchar(11)

+ Quản lý khách:

Bảng 2. 16 Quản lý khách

Thuộc tính	Phương thức
MKH	nchar(10)
MDVE	nchar(10)
MDV	nchar(10)
CUSNAME	nchar(20)
AGE	int
SEX	varchar(11)
PHONENUMBER	nchar(11)
TYPEOFTICKETS	nchar(10)
FARE	money
BUSES	nchar(20)

PERNAME	nchar(20)
MNV	nchar(10)

+ Quản lý đặt vé:

Bảng 2. 17 Quản lý đặt vé

Thuộc tính	Phương thức
MKH	nchar(10)
MDVE	nchar(10)
MDV	nchar(10)
CUSNAME	nchar(20)
AGE	nchar(10)
PERNAME	nchar(20)
TYPEOFTICKETS	nchar(10)
DATE	varchar(20)
START	nchar(20)
FINISH	nchar(20)
BUSES	nchar(10)
FARE	money

+ Quản lý dịch vụ:

Bảng 2. 18 Quản lý dịch vụ

Thuộc tính	Phương thức
MDV	nchar(10)
MKH	nchar(10)
SEX	nchar(10)
CUSNAME	nchar(10)
AGE	int
QUANTITY	money
TOTAL	int
ITEMNAME	nchar(10)

SEVNICENAME	nchar(10)
-------------	-----------

+ Quản lý cơ sở vật chất:

Bảng 2. 19 Quản lý cơ sở vật chất

Thuộc tính	Phương thức
MVT	nchar(10)
FULLNAME	nchar(10)
QUANTITY	int
DATERANGE	date
USEDTIME	date
INCOME_EXPENDING	money
MVT	nchar(10)

+ Quản lý tài khoản admin:

Bảng 2. 20 Quản lý tài khoản admin

Thuộc tính	Phương thức
Fullname	nchar(30)
Birthday	date
Sex	nchar(10)
Username	nchar(30)
Password	nchar(30)

+ Quản lý tài khoản nhân viên xe:

Bảng 2. 21 Quản lý tài khoản nhân viên xe

Thuộc tính	Phương thức
Fullname	nchar(20)
Age	int
UseName	nchar(20)
Password	nchar(10)

## CHƯƠNG 3: CÀI ĐẶT CHƯƠNG TRÌNH

### 3.1 Tổ chức chương trình

Hệ thống trên được xây dựng theo mô hình MVC (Model-View-Controller), giúp tách biệt rõ ràng giữa phần xử lý dữ liệu, giao diện và điều khiển luồng xử lý. Việc tổ chức chương trình theo mô hình này giúp hệ thống dễ dàng bảo trì và mở rộng trong tương lai.

#### 3.1.1. Gói Model

Gói Model chứa các lớp thực hiện xử lý dữ liệu, như quản lý thông tin xe, tài xế, tuyến đường, v.v.

Ví dụ: Lớp CarManager với các phương thức addCar, deleteCar, updateCar.

+ Addcar:

```
public String addCar(String plate, String bus, String driver, String age, String sex,
String phone, String fare, String route, String sit, String sevice) {
    String urlString = BASE_URL + "addCar.php";
    JSONObject postData = new JSONObject();
    try {
        postData.put("bienso", plate);
        postData.put("tenxe", bus);
        postData.put("tentaixe", driver);
        postData.put("tuoi", age);
        postData.put("gioitinh", sex);
        postData.put("sdt", phone);
        postData.put("phi", fare);
        postData.put("tuyen", route);
        postData.put("ghengoi", sit);
        postData.put("mdv", sevice);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return sendPostRequest(urlString, postData);
}
```

+ Deletecar:

```
public String deleteCar(int id) {
    String urlString = BASE_URL + "deleteCar.php";
    JSONObject postData = new JSONObject();
    try {
        postData.put("bienso", id);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```

    }
    return sendPostRequest(urlString, postData);
}

```

+ Editcar:

```

public String updateCar( String plate, String bus, String driver, String age, String
sex, String phone, String fare, String route, String sit, String sevice) {
    String urlString = BASE_URL + "updateCar.php";
    JSONObject postData = new JSONObject();
    try {
        postData.put("bienso", plate);
        postData.put("tenxe", bus);
        postData.put("tentaixe", driver);
        postData.put("tuoi", age);
        postData.put("gioitinh", sex);
        postData.put("sdt", phone);
        postData.put("phi", fare);
        postData.put("tuyen", route);
        postData.put("ghengoi", sit);
        postData.put("mdv", sevice);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return sendPostRequest(urlString, postData);
}

```

### 3.1.2. Gói View

Gói View chứa các lớp xử lý giao diện người dùng, như Add\_Car hoặc View\_Car. Các lớp này bao gồm logic hiển thị và tương tác với người dùng thông qua các thành phần như EditText, Button, Spinner, v.v.

+ Add\_car:

```

public class Add_Car extends AppCompatActivity {
    private EditText EtBienso, EtNhaxe, EtTaixe, EtSdt, EtGia;
    private RadioButton RdNam, RdNu;
    private Spinner SnAge, SnRoute, SnSev, SnSit, SN;
    private Button btnAdd, btnEdit, btnDelete;
    private ImageView Exit;
    private CarManager car;
    private FloatingActionButton fabEdit, fabDelete ,Fab;
    private Animation openFab, closeFab, fromBottom, toBottom ;
    private boolean clicked = false;
}

```



```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    EdgeToEdge.enable(this);
    setContentView(R.layout.activity_add_car);
    ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v,
insets) -> {
        Insets systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars());
        v.setPadding(systemBars.left, systemBars.top, systemBars.right,
systemBars.bottom);
        return insets;
    });
    openFab = AnimationUtils.loadAnimation(this, R.anim.open);
    closeFab = AnimationUtils.loadAnimation(this, R.anim.close);
    fromBottom = AnimationUtils.loadAnimation(this, R.anim.bottom);
    toBottom = AnimationUtils.loadAnimation(this, R.anim.bottom2);
    Fab = findViewById(R.id.fab);
    SN = findViewById(R.id.spinner_qlx);
    EtBienso = findViewById(R.id.edtxtbks);
    EtNhaxe = findViewById(R.id.edtxtname);
    EtTaixe = findViewById(R.id.edtxtdrivername);
    EtSdt = findViewById(R.id.edtxtphone);
    EtGia = findViewById(R.id.edtxtfare);
    RdNu = findViewById(R.id.rdiofemale);
    RdNam = findViewById(R.id.rdiomale);
    SnSev = findViewById(R.id.spinner_mdv);
    SnRoute = findViewById(R.id.spinner_buses);
    SnAge = findViewById(R.id.spinner_age);
    SnSit = findViewById(R.id.spinner_seat);
    btnAdd = findViewById(R.id.BtnAdd);
    fabEdit = findViewById(R.id.fab_edit);
    fabDelete = findViewById(R.id.fab_delete);
    Exit = findViewById(R.id.btnex);
}

```

+ Các tương tác add\_car:

```

Fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        onAddButtonClicked();
    }
}

```

```

});

btnAdd.setOnClickListener(v -> {
    String plate = EtBienso.getText().toString();
    String bus = EtNhaxe.getText().toString();
    String driver = EtTaixe.getText().toString();
    String phone = EtSdt.getText().toString();
    String sex = RdNam.isChecked() ? "Nam" : "Nữ";
    String age = SnAge.getSelectedItem().toString();
    String fare = EtGia.getText().toString();
    String route = SnRoute.getSelectedItem().toString();
    String sit = SnSit.getSelectedItem().toString();
    String sevice = SnSev.getSelectedItem().toString();

    if (plate.isEmpty() || bus.isEmpty() || sex.isEmpty() || phone.isEmpty() ||
driver.isEmpty()) {
        Toast.makeText(Add_Car.this, "Vui lòng điền tất cả các trường",
Toast.LENGTH_SHORT).show();
        return;
    }

    new Thread() -> {
        String response = car.addCar(plate, bus, driver, age, sex, phone, fare,
route, sit, sevice );
        runOnUiThread() -> Toast.makeText(Add_Car.this, "Add Response: " +
response, Toast.LENGTH_SHORT).show());
    }).start();
});

fabDelete.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String idStr = EtBienso.getText().toString();
        if (idStr.isEmpty()) {
            Toast.makeText(Add_Car.this, "Vui lòng nhập ID",
Toast.LENGTH_SHORT).show();
            return;
        }

        int id = Integer.parseInt(idStr);

        new Thread() -> {
            String response = car.deleteCar(id);

```

```

        runOnUiThread() -> Toast.makeText(Add_Car.this, "Delete Response: "
+ response, Toast.LENGTH_SHORT).show());
    }).start();
    }
});

fabEdit.setOnClickListener(v -> {
    String plate = EtBienso.getText().toString();
    if (plate.isEmpty()) {
        Toast.makeText(Add_Car.this, "Vui lòng nhập ID",
Toast.LENGTH_SHORT).show();
        return;
    }
    String bus = EtNhaxe.getText().toString();
    String driver = EtTaixe.getText().toString();
    String phone = EtSdt.getText().toString();
    String sex = RdNam.isChecked() ? "Nam" : "Nữ";
    String age = SnAge.getSelectedItem().toString();
    String fare = EtGia.getText().toString();
    String route = SnRoute.getSelectedItem().toString();
    String sit = SnSit.getSelectedItem().toString();
    String sevice = SnSev.getSelectedItem().toString();

    new Thread() -> {
        String response = car.updateCar(plate, bus, driver, age, sex, phone, fare,
route, sit, sevice );
        runOnUiThread() -> Toast.makeText(Add_Car.this, "Update Response: "
+ response, Toast.LENGTH_SHORT).show());
    }).start();
});

```

+ view\_car:

```

public class View_Car extends AppCompatActivity {
    private ListView listView;
    private ArrayList<String> carList;
    private ArrayAdapter<String> adapter, Adapter;
    private Spinner spin;
    private ImageView Exit;
    private Button btnDelete, btnEdit;
    private CarManager car;
    private SearchView searchView;
    private ArrayList<String> filteredList = new ArrayList<>();
}

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    EdgeToEdge.enable(this);
    setContentView(R.layout.activity_view_car);
    ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v,
insets) -> {
        Insets systemBars =
insets.getInsets(WindowInsetsCompat.Type.systemBars());
        v.setPadding(systemBars.left, systemBars.top, systemBars.right,
systemBars.bottom);
        return insets;
    });
    searchView = findViewById(R.id.SeachCar);
    Exit = findViewById(R.id.btnback);
    listView = findViewById(R.id.list);
    carList = new ArrayList<>();
    spin = findViewById(R.id.spinner_view_car);
    adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1,
carList);

    listView.setAdapter(adapter);

```

+ Các tương tác view\_car:

- Chuyển đổi giữa add\_car và view\_car:

```

String[] options = {"Chọn Chức năng:", "Add Car", "View Car"};

// Tạo Adapter cho Spinner
ArrayAdapter<String> adapter = new ArrayAdapter<>(this,
android.R.layout.simple_spinner_item, options);

adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_it
em);
spin.setAdapter(adapter);

spin.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        switch (position) {
            case 1:

```

```

        Intent intentB = new Intent(View_Car.this, Add_Car.class);
        startActivity(intentB);
        break;

    case 2:

        break;

    default:

        break;
    }
}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});

```

- Thoát:

```

Exit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intentB = new Intent(View_Car.this, SelectAd.class);
        startActivity(intentB);
    }
});

```

- Sự kiện khi bấm vào 1 item trên listview:

```

listView.setOnItemClickListener((parent, view, position, id) -> {

    String selectedItem = carList.get(position);
    Intent intent = new Intent(View_Car.this, Add_Car.class);
    intent.putExtra("carData", selectedItem);
    startActivity(intent);
});

searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener()
{
    @Override
    public boolean onQueryTextSubmit(String query) {

```

```

        return false;
    }

    @Override
    public boolean onQueryTextChange(String newText) {
        filterList(newText);
        return true;
    }
});
}

```

### 3.1.3. Gói Controller

Gói Controller chịu trách nhiệm kết nối giữa Model và View. Nó bao gồm các phương thức xử lý dữ liệu và cập nhật giao diện người dùng.

### 3.1.4. Gói Service

Gói Service chịu trách nhiệm quản lý các dịch vụ xử lý dữ liệu giữa ứng dụng và cơ sở dữ liệu, ví dụ như kết nối MySQL, truy vấn SQL.

+ Yêu cầu POST với dữ liệu JSON tới một API:

```

private static final String BASE_URL = "http://192.168.1.11/qlbx/phpCar/";

private String sendPostRequest(String urlString, JSONObject postData) {
    HttpURLConnection conn = null;
    try {
        URL url = new URL(urlString);
        conn = (HttpURLConnection) url.openConnection();
        conn.setRequestMethod("POST");
        conn.setRequestProperty("Content-Type", "application/json; utf-8");
        conn.setRequestProperty("Accept", "application/json");
        conn.setDoOutput(true);

        try (OutputStream os = conn.getOutputStream()) {
            byte[] input = postData.toString().getBytes("utf-8");
            os.write(input, 0, input.length);
        }

        int responseCode = conn.getResponseCode();
        if (responseCode == HttpURLConnection.HTTP_OK) {
            try (BufferedReader br = new BufferedReader(new
InputStreamReader(conn.getInputStream(), "utf-8"))) {
                StringBuilder response = new StringBuilder();
                String responseLine;
                while ((responseLine = br.readLine()) != null) {

```

```

        response.append(responseLine.trim());
    }
    return response.toString();
}
} else {
    return "HTTP Error: " + responseCode;
}

} catch (Exception e) {
    e.printStackTrace();
    return "Error: " + e.getMessage();
} finally {
    if (conn != null) {
        conn.disconnect();
    }
}
}
}

```

+ Kết nối tới cơ sở dữ liệu:

```

<?php

$servername = "localhost";
$username = "root";
$password = "";
$dbname = "qlbxmd";

// Kết nối tới database
$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    echo json_encode(["error" => "Connection failed: " . $conn->connect_error]);
    exit();
}

```

+ Các câu lệnh truy vấn:

- Thêm:

```

$data = json_decode(file_get_contents("php://input"));

// Kiểm tra dữ liệu đầu vào
if (!isset($data->biensoi, $data->tenxe, $data->tentaixe, $data->tuoi, $data->gioitinh,
    $data->sdt, $data->phi, $data->tuyen, $data->ghengoi, $data->mdv)) {

```

```

    echo json_encode(["error" => "Invalid input data"]);
    exit();
}

// Gán giá trị từ JSON vào các biến PHP
$bienso = $data->bienso;
$tenxe = $data->tenxe;
$tentaixe = $data->tentaixe;
$tuoi = $data->tuoi;
$gioitinh = $data->gioitinh; // Nam/Nữ
$sdt = $data->sdt;
$phi = $data->phi;
$tuyen = $data->tuyen;
$ghengoi = $data->ghengoi;
$mdv = $data->mdv;

// Chuẩn bị câu lệnh SQL
$stmt = $conn->prepare("INSERT INTO cars (bienso, tenxe, tentaixe, tuoi, gioitinh,
sdt, phi, tuyen, ghengoi, mdv) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)");
$stmt->bind_param("sssisssss", $bienso, $tenxe, $tentaixe, $tuoi, $gioitinh, $sdt,
$phi, $tuyen, $ghengoi, $mdv);

// Thực thi câu lệnh SQL và kiểm tra kết quả
if ($stmt->execute()) {
    echo json_encode(["message" => "Add car Successfully"]);
} else {
    echo json_encode(["error" => "Lỗi: " . $stmt->error]);
}
// Đóng kết nối
$stmt->close();
$conn->close();
?>

```

- Xóa:

```

$data = json_decode(file_get_contents("php://input"));
$bienso = $data->bienso;
$sql = "DELETE FROM cars WHERE bienso=$bienso";
if ($conn->query($sql) === TRUE) {
    echo json_encode(["message" => "Xoa thanh cong"]);
} else {
    echo json_encode(["error" => "Lỗi: " . $conn->error]);
}
$conn->close();

```



```
?>
```

- Sửa:

```
$data = json_decode(file_get_contents("php://input"));
$bienso = $data->bienso;
$tenxe = $data->tenxe;
$tentaixe = $data->tentaixe;
$tuoi = $data->tuoi;
$gioitinh = $data->gioitinh; // Nam/Nữ
$sdt = $data->sdt;
$phi = $data->phi;
$tuyen = $data->tuyen;
$ghengoi = $data->ghengoi;
$mdv = $data->mdv;
$sql = "UPDATE cars SET tenxe='$tenxe', tentaixe='$tentaixe', tuoi='$tuoi',
gioitinh='$gioitinh', sdt='$sdt', phi='$phi', tuyen='$tuyen', ghengoi='$ghengoi',
mdv='$mdv' WHERE bienso=$bienso";
if ($conn->query($sql) === TRUE) {
    echo json_encode(["message" => "Cap nhat thong tin thanh cong"]);
} else {
    echo json_encode(["error" => "Lỗi: " . $conn->error]);
}
$conn->close();
?>
```

## 3.2. Các chức năng chương trình

### 3.2.1. Quản lý tài khoản

Mô tả: Cho phép quản lý thông tin đăng nhập của người dùng (nhân viên hoặc quản trị viên) trên hệ thống.

Các chức năng chi tiết:

- Đăng nhập: Kiểm tra thông tin tài khoản từ cơ sở dữ liệu để xác thực.
- Đổi mật khẩu: Cho phép người dùng thay đổi mật khẩu sau khi đăng nhập thành công.

Phân quyền: Phân biệt các quyền hạn (ví dụ: quản trị viên có thể thao tác toàn bộ dữ liệu, nhân viên chỉ được thực hiện các chức năng liên quan).

Yêu cầu: Hệ thống phải mã hóa mật khẩu trước khi lưu trữ trong cơ sở dữ liệu.

### 3.2.2. Quản lý xe

Mô tả: Hỗ trợ quản lý danh sách xe bus bao gồm việc thêm mới, sửa đổi, và xóa thông tin xe.

Các chức năng chi tiết:

- Thêm xe mới: Nhập thông tin xe như biển số, tên xe, tài xế, số ghế, tuyến đường, giá vé, v.v.
- Sửa thông tin xe: Cập nhật thông tin xe khi có thay đổi (ví dụ: đổi tài xế, cập nhật tuyến đường).
- Xóa xe: Loại bỏ thông tin xe khỏi hệ thống.
- Tìm kiếm: Tìm kiếm nhanh xe dựa trên biển số hoặc tuyến đường.

Yêu cầu:

- Đảm bảo không trùng biển số khi thêm mới xe.
- Cảnh báo nếu cố gắng xóa một xe đang hoạt động trên tuyến.

### **3.2.3. Quản lý nhân viên**

Mô tả: Hỗ trợ quản lý thông tin cá nhân và công việc của nhân viên.

Các chức năng chi tiết:

- Thêm nhân viên: Nhập thông tin như họ tên, tuổi, giới tính, số điện thoại, và vị trí công việc.
- Cập nhật thông tin: Chỉnh sửa thông tin khi nhân viên thay đổi công việc hoặc thông tin liên lạc.
- Xóa nhân viên: Xóa khỏi danh sách nhân viên khi không còn làm việc.

Phân quyền và theo dõi: Gán quyền hạn cho nhân viên (ví dụ: tài xế, phụ xe).

Yêu cầu: Thông báo khi xóa nhân viên để tránh sai sót.

### **3.2.4. Quản lý cơ sở vật chất**

Mô tả: Hỗ trợ quản lý các vật tư của biển đang hoạt động.

Các chức năng chi tiết:

- Thêm vật tư: Ghi nhận các thông tin của vật tư muốn nhập về
- Cập nhật tuyến đường: Chỉnh sửa thông tin vật tư khi có thay đổi (ví dụ: điều chỉnh số lượng, giá cả).
- Xóa tuyến đường: Loại bỏ vật tư không còn tồn tại.
- Tìm kiếm vật tư: Tìm nhanh vật tư dựa trên mã hoặc tên vật tư đó.

### **3.2.5. Quản lý vé xe**

Mô tả: Quản lý thông tin vé xe bus cho khách hàng.

Các chức năng chi tiết:

- Đặt vé: Nhập thông tin khách hàng và thông tin vé như tuyến đường, ngày giờ, số ghế, và giá vé.
- Theo dõi vé: Hiện thị danh sách vé đã đặt, sắp xếp theo ngày giờ hoặc tuyến đường.

Yêu cầu:

- Đảm bảo không trùng số ghế trên cùng một chuyến xe.
- Thông báo nếu hủy vé trong thời gian ngắn trước giờ khởi hành.

### **3.2.6. Quản lý dịch vụ**

Mô tả: Quản lý các dịch vụ của biển cho khách hàng.

Các chức năng chi tiết:

- Thêm dịch vụ: Nhập thông tin dịch vụ mới của bến.
- Xóa hoặc sửa: chỉnh sửa các loại dịch vụ khi có thay đổi.
- Tìm kiếm: tìm kiếm các loại dịch vụ đang có.

Yêu cầu:

- Đảm bảo không trùng số ghế trên cùng một chuyến xe.
- Thông báo nếu hủy vé trong thời gian ngắn trước giờ khởi hành.

### **3.3. Giao diện chính của chương trình**

3.3.1 Giao diện main



Hình 3. 1 Giao diện main

Bước 1: Khi người dùng mở ứng dụng lên người dùng sẽ thấy main, ở đây người dùng sẽ chọn “Next” để vào trang chủ.

Bảng 3. 1 Bảng đặc tả giao diện main

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	Logo	ImageView	Logo
2	btncus	Button	Chuyển sang trang chủ
3	TextView1	TextView	Không
4	TextView2	TextView	Không

5	TextView3	TextView	Không
6	TextView4	TextView	Không

### 3.3.2 Giao diện trang chủ



Hình 3. 2 Giao diện trang chủ

Bước 1: Khi người dùng mở ứng dụng lên người dùng sẽ thấy trang chủ, ở đây người dùng sẽ chọn chức năng.

Bước 2: Người dùng chọn chức năng:

2.1 Nếu bạn là nhân viên, chọn Employee ứng dụng sẽ chuyển sang trang chọn chức năng nghiệp vụ nhân viên

2.2 Nếu bạn là khách hàng, chọn Customer

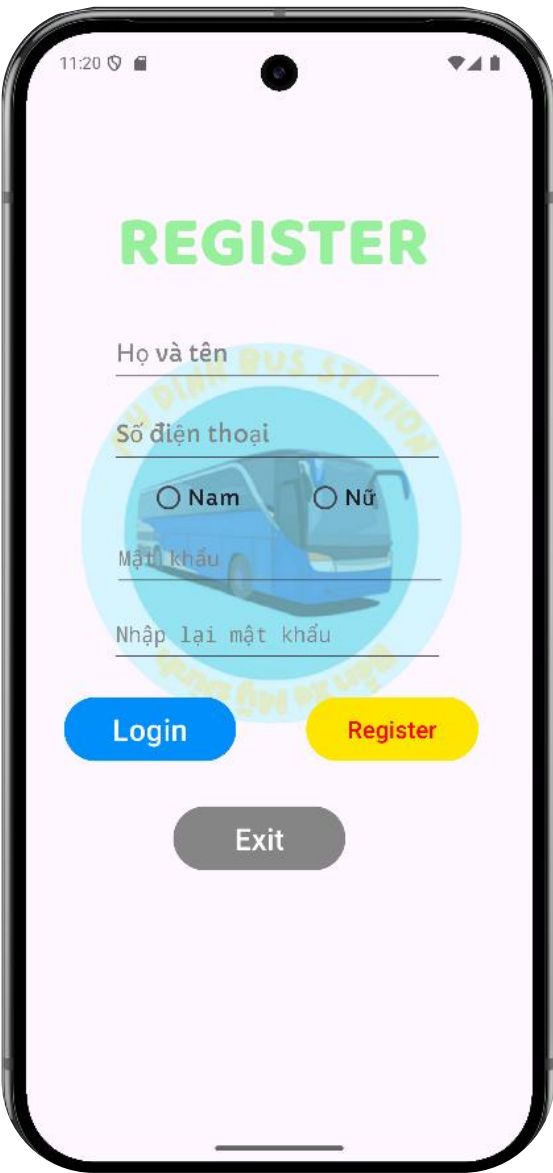
2.3 Nếu bạn là quản trị viên chọn Admin

2.4 Nếu bạn là nhà xe, chọn bus gara

Bảng 3. 2 Bảng đặc tả giao diện trang chủ

STT	Tên đối tượng	Kiểu	Ý nghĩa
7	Logo	ImageView	Logo
8	ImageView1	ImageView	Không
9	ImageView2	ImageView	Không
10	btnemployee	Button	Chuyển sang trang chọn nghiệp vụ dành cho nhân viên
11	btncus	Button	Chuyển sang trang đặt vé cho khách hàng
12	btnadmin	Button	Chuyển sang trang đăng nhập dành cho quản trị viên
13	btnbusdriver	Button	Chuyển sang trang đăng nhập dành cho nhân viên nhà xe
14	ImageView3	ImageView	Không
15	ImageView4	ImageView	Không
16	ImageView5	ImageView	Không
17	ImageView6	ImageView	Không
18	TextView1	TextView	Không
19	TextView2	TextView	Không
20	TextView3	TextView	Không
21	TextView4	TextView	Không
22	TextView6	TextView	Không

3.3.3 Giao diện đăng ký khách hàng



Hình 3. 3 Giao diện đăng ký tài khoản khách hàng

Bước 1: Người dùng nhập đầy đủ thông tin hiển thị trên màn hình hệ thống

Bước 2: Nhấn “Register” để đăng ký

Bước 3: Nếu bạn đã có tài khoản rồi hãy chọn “Login” để đăng nhập, nếu bạn không phải khách hàng hoặc không muốn đăng ký tài khoản hãy nhấn “Exit” để thoát

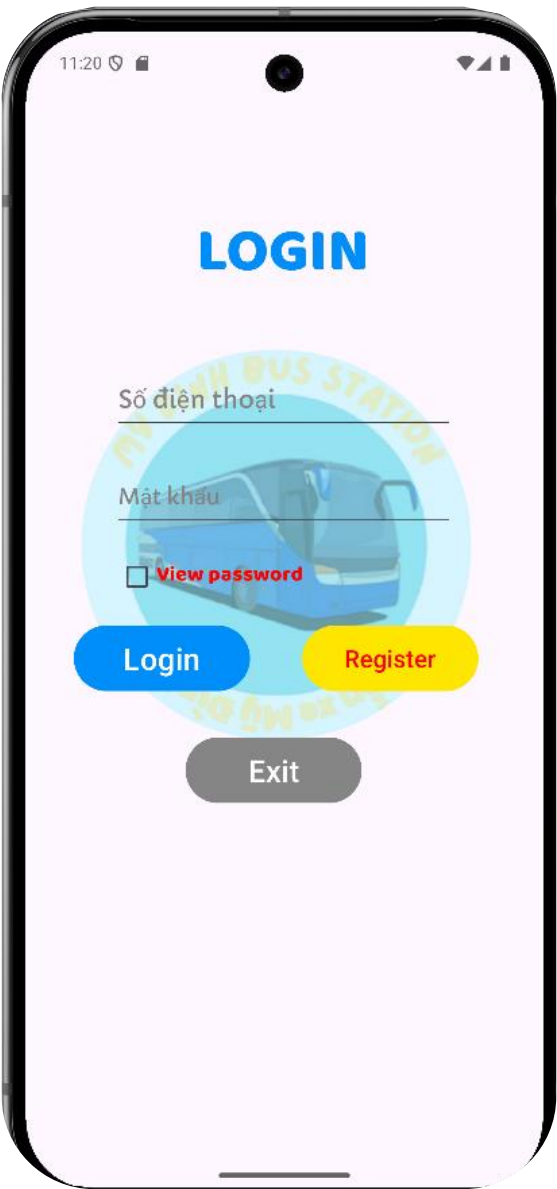
Bảng 3. 3 Bảng đặc tả giao diện đăng ký tài khoản khách hàng

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	ImageView hiển thị logo chìm

2	TextView	TextView	Không
3	edtxtuser	Plain Text	Điền số điện thoại khách hàng
4	edtxtname	Plain Text	Điền họ và tên khách hàng
5	edtxtpass	Plain Text	Điền mật khẩu tài khoản của khách hàng
6	edtxtcfpass	Plain Text	Điền lại mật khẩu tài khoản của khách hàng
7	rdiomale	Radio	Chọn giới tính khách hàng
8	rdiofemale	Radio	Chọn giới tính khách hàng
9	btnlogin	Button	Chuyển sang trang đăng nhập
10	btnregister	Button	Đăng ký tài khoản
11	btnexit	Button	Thoát



3.3.4 Giao diện đăng nhập khách hàng



Hình 3. 4 Giao diện đăng nhập tài khoản khách hàng

Bước 1: Người dùng nhập đầy đủ thông tin hiển thị trên màn hình hệ thống.

Bước 2: Nhấn “Login” để đăng nhập.

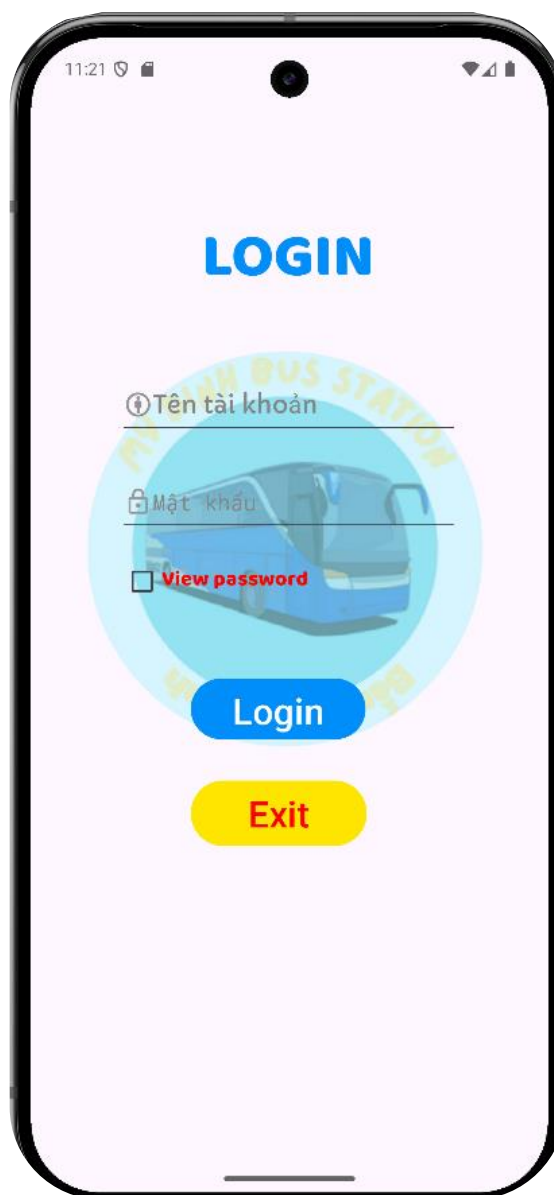
Bước 3: Nếu bạn chưa có tài khoản rồi hãy chọn “Register” để đăng ký, nếu bạn không phải khách hàng hoặc không muốn đăng ký tài khoản hãy nhấn “Exit” để thoát.

Bảng 3. 4 Bảng đặc tả giao diện đăng ký tài khoản khách hàng

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	ImageView hiển thị logo chìm
2	TextView	TextView	Không

3	edtxtuser	Plain Text	Điền số điện thoại khách hàng
4	edtxtpass	Plain Text	Điền mật khẩu tài khoản của khách hàng
5	btnlogin	Button	Chuyển sang trang đăng nhập
6	btnregister	Button	Đăng ký tài khoản
7	btnexit	Button	Thoát
8	viewpass	CheckBox	Hiện thị mật khẩu

### 3.3.5 Giao diện đăng nhập quản trị viên



Hình 3. 5 Giao diện đăng nhập tài khoản quản trị viên

Bước 1: Người dùng nhập đầy đủ thông tin hiển thị trên màn hình hệ thống.

Bước 2: Nhấn “Login” để đăng nhập.

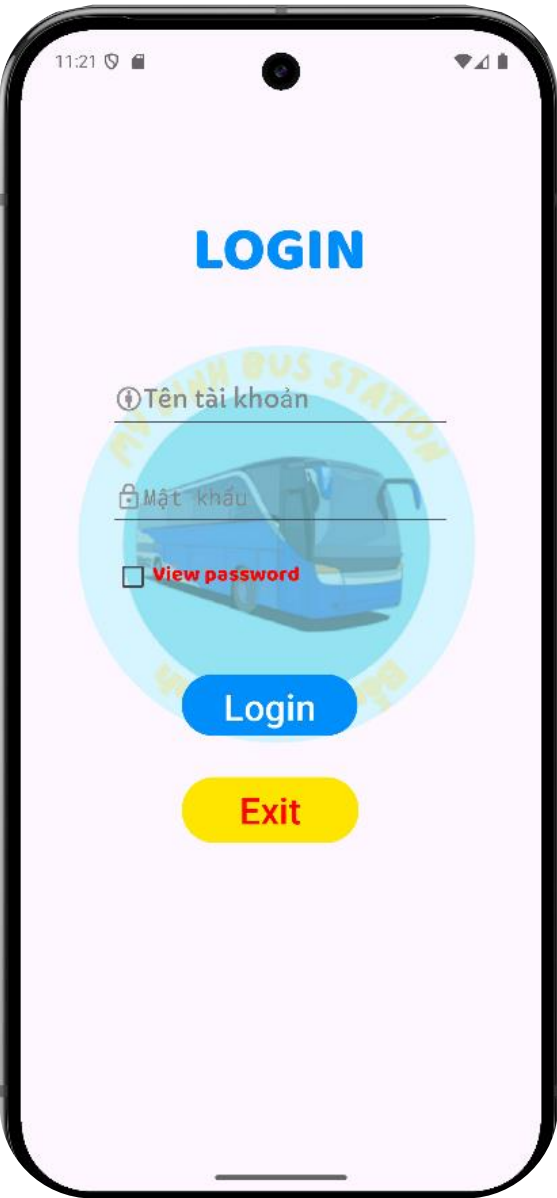
Bước 3: Nếu bạn không phải quản trị viên hoặc không muốn đăng nhập tài khoản hãy nhấn “Exit” để thoát.

Chức năng: Sử dụng để điền thông tin quản trị viên đăng nhập tài khoản.

Bảng 3. 5 Bảng đặc tả giao diện đăng ký tài khoản quản trị viên

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	ImageView hiển thị logo chìm
2	TextView	TextView	Không
3	edtxtuser	Plain Text	Điền tên đăng nhập quản trị viên
4	edtxtpass	Plain Text	Điền mật khẩu tài khoản của quản trị viên
5	btnlogin	Button	Chuyển sang trang đăng nhập
6	btnexit	Button	Thoát
7	viewpass	CheckBox	Hiển thị mật khẩu

3.3.6 Giao diện đăng nhập nhân viên



Hình 3. 6 Giao diện đăng nhập tài khoản nhân viên

Bước 1: Người dùng nhập đầy đủ thông tin hiển thị trên màn hình hệ thống.

Bước 2: Nhấn “Login” để đăng nhập.

Bước 3: Nếu bạn không phải nhân viên hoặc không muốn đăng nhập tài khoản hãy nhấn “Exit” để thoát.

Bảng 3. 6 Bảng đặc tả giao diện đăng ký tài khoản nhân viên

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	ImageView hiển thị logo chìm

2	TextView	TextView	Không
3	edtxtuser	Plain Text	Điền tên đăng nhập quản trị viên
4	edtxtpass	Plain Text	Điền mật khẩu tài khoản của quản trị viên
5	btnlogin	Button	Chuyển sang trang đăng nhập
6	btnexit	Button	Thoát
7	viewpass	CheckBox	Hiện thị mật khẩu

### 3.3.7 Giao diện chọn chức năng nghiệp vụ cho nhân viên



Hình 3. 7 Giao diện chọn nghiệp vụ nhân viên

Bước 1: Nhân viên sau khi chọn “Employee” ở trang chủ sẽ được ứng dụng đưa đến trang chọn chức năng nghiệp vụ như trên.

Bước 2: Nhân viên lựa chọn chức năng nghiệp vụ của mình.

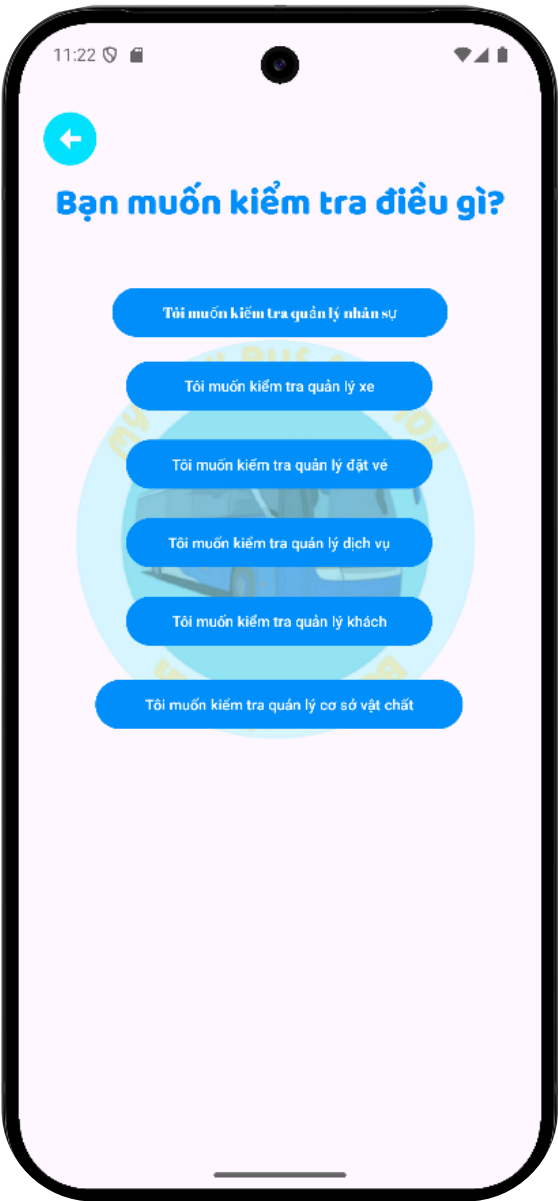
Bước 3: Hệ thống hiển thị trang đăng nhập để nhân viên đăng nhập tài khoản của mình

Chức năng: Sử dụng để chọn chức năng nghiệp vụ của nhân viên bên xe.

Bảng 3. 7 Bảng đặc tả giao diện đăng ký tài khoản nhân viên

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	ImageView hiển thị logo chìm
2	TextView	TextView	Không
3	btne1	Button	Chọn chức năng nghiệp vụ cho nhân viên
4	btne2	Button	Chọn chức năng nghiệp vụ cho nhân viên
5	btne3	Button	Chọn chức năng nghiệp vụ cho nhân viên
6	btne4	Button	Chọn chức năng nghiệp vụ cho nhân viên
7	btne5	Button	Chọn chức năng nghiệp vụ cho nhân viên
8	btne6	Button	Chọn chức năng nghiệp vụ cho nhân viên

3.3.8 Giao diện chọn chức năng nghiệp vụ cho quản trị viên



Hình 3. 8 Giao diện chọn nghiệp vụ quản trị viên

Bước 1: Nhân viên sau khi chọn “Admin” ở trang chủ sẽ được ứng dụng đưa đến trang chọn chức năng nghiệp vụ như trên.

Bước 2: Quản trị viên lựa chọn chức năng nghiệp vụ của mình.

Bước 3: Hệ thống hiển thị trang đăng nhập để quản trị viên đăng nhập tài khoản của mình

Chức năng: Sử dụng để chọn chức năng nghiệp vụ của quản trị viên bến xe.

Bảng 3. 8 Bảng đặc tả giao diện đăng ký tài khoản nhân viên

STT	Tên đối tượng	Kiểu	Ý nghĩa
-----	---------------	------	---------

1	ImageView	ImageView	ImageView hiển thị logo chìm
2	TextView	TextView	Không
3	btnad1	Button	Chọn chức năng nghiệp vụ cho quản trị viên
4	btnad2	Button	Chọn chức năng nghiệp vụ cho quản trị viên
5	btnad3	Button	Chọn chức năng nghiệp vụ cho quản trị viên
6	btnad4	Button	Chọn chức năng nghiệp vụ cho quản trị viên
7	btnad5	Button	Chọn chức năng nghiệp vụ cho quản trị viên
8	btnad6	Button	Chọn chức năng nghiệp vụ cho quản trị viên

### 3.3.9 Giao diện quản lý xe

11:23

QUẢN LÝ NHÂN VIÊN  
(EMPLOYEE MANAGEMENT)

Select Activity  
View Employees  
Add Employee

MNV: Employee ID

Tên nhân viên: Fullname

Tuổi: Age

Giới tính: ☐ Nam ☐ Nữ

Số điện thoại: Phone Number

Lương: Wage VNĐ

Chức vụ: Position

Thêm nhân viên Sửa Xóa

Hình 3. 9 Giao diện quản lý nhân viên



### 1. Thêm xe:

Bước 1: Điền đầy đủ thông tin của xe.

Bước 2: Nhấn “Thêm” để thêm nhân viên vào danh sách và cập nhật vào database.

### 2. Sửa thông tin xe:

Bước 1: Sau khi chọn nhân viên cần sửa ở mục View Car. Nhân viên sẽ được hệ thống đưa về trang Add Car với thông tin của xe cần sửa.

Bước 2: Nhân viên chỉnh sửa thông tin xe và nhấn sửa.

Bước 3: Thông tin xe được cập nhật vào View Car và database.

### 2. Xoá xe:

Bước 1: Sau khi chọn xe cần xoá ở mục View Car. Nhân viên sẽ được hệ thống đưa về trang Add Car với thông tin của xe cần xoá.

Bước 2: Nhân viên nhấn xoá để xoá xe.

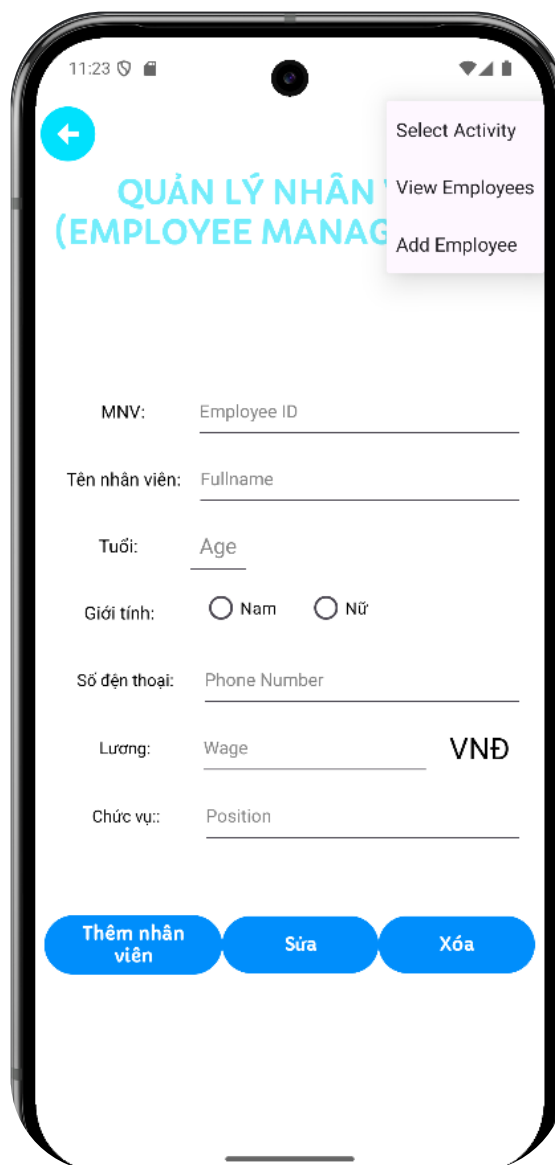
Bước 3: Thông tin xe được cập nhật vào View Car và database.

Bảng 3. 9 Bảng đặc tả giao diện quản lý xe

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	Thoát khỏi trang quản lý nhân viên
2	TextView1	TextView	Không
3	TextView2	TextView	Tên thông tin cần điền
4	TextView3	TextView	Tên thông tin cần điền
5	TextView4	TextView	Tên thông tin cần điền
6	TextView5	TextView	Tên thông tin cần điền
7	TextView6	TextView	Tên thông tin cần điền
8	TextView7	TextView	Tên thông tin cần điền
9	TextView8	TextView	Tên thông tin cần điền
10	TextView9	TextView	Tên thông tin cần điền
11	txtmnv	Plain Text	Điền thông tin nhân viên
12	txtname	Plain Text	Điền thông tin nhân viên
13	txtage	Plain Text	Điền thông tin nhân viên
14	txtwage	Plain Text	Điền thông tin nhân viên

15	txtphone	Plain Text	Điền thông tin nhân viên
16	txtposition	Plain Text	Điền thông tin nhân viên
17	spinner1	Spinner	Chọn trang quản lý hoặc xem danh sách xe
18	btnadd	Button	Thêm xe
19	btnupdate	Button	Sửa xe
20	btndelete	Button	Xoá xe
21	rdiomale	Radio	Chọn giới tính
22	rdiofemale	Radio	Chọn giới tính

### 3.3.10 Giao diện quản lý nhân viên



Hình 3. 10 Giao diện quản lý nhân viên

### 1. Thêm nhân viên:

Bước 1: Điền đầy đủ thông tin của nhân viên.

Bước 2: Nhấn “Thêm” để thêm nhân viên vào danh sách và cập nhật vào database.

### 2. Sửa thông tin nhân viên:

Bước 1: Sau khi chọn nhân viên cần sửa ở mục View Employee. Nhân viên sẽ được hệ thống đưa về trang Add Employee với thông tin của nhân viên cần sửa.

Bước 2: Nhân viên nhân sự chỉnh sửa thông tin nhân viên và nhấn sửa.

Bước 3: Thông tin nhân viên được cập nhật vào View Employee và database.

### 2. Xóa nhân viên:

Bước 1: Sau khi chọn nhân viên cần xóa ở mục View Employee. Nhân viên sẽ được hệ thống đưa về trang Add Employee với thông tin của nhân viên cần xóa.

Bước 2: Nhân viên nhân sự nhấn xóa để xóa nhân viên.

Bước 3: Thông tin nhân viên được cập nhật vào View Employee và database.

Bảng 3. 10 Bảng đặc tả giao diện quản lý nhân viên

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	Thoát khỏi trang quản lý nhân viên
2	TextView1	TextView	Không
3	TextView2	TextView	Tên thông tin cần điền
4	TextView3	TextView	Tên thông tin cần điền
5	TextView4	TextView	Tên thông tin cần điền
6	TextView5	TextView	Tên thông tin cần điền
7	TextView6	TextView	Tên thông tin cần điền
8	TextView7	TextView	Tên thông tin cần điền
9	TextView8	TextView	Tên thông tin cần điền
10	TextView9	TextView	Tên thông tin cần điền
11	txtmnv	Plain Text	Điền thông tin nhân viên
12	txtname	Plain Text	Điền thông tin nhân viên
13	txtage	Plain Text	Điền thông tin nhân viên
14	txtwage	Plain Text	Điền thông tin nhân viên

15	txtphone	Plain Text	Điền thông tin nhân viên
16	txtposition	Plain Text	Điền thông tin nhân viên
17	spinner1	Spinner	Chọn trang quản lý hoặc xem danh sách nhân viên
18	btnadd	Button	Thêm nhân viên
19	btnupdate	Button	Sửa nhân viên
20	btndelete	Button	Xoá nhân viên
21	rdiomale	Radio	Chọn giới tính
22	rdiofemale	Radio	Chọn giới tính

### 3.3.11 Giao diện quản lý khách



Hình 3. 11 Giao diện quản lý khách

#### 1. Thêm khách hàng:

Bước 1: Điền đầy đủ thông tin của khách hàng.

Bước 2: Nhấn “Thêm” để thêm khách hàng vào danh sách và cập nhật vào database.

2. Sửa thông tin khách hàng:

Bước 1: Sau khi chọn khách hàng cần sửa ở mục View Customer. Nhân viên sẽ được hệ thống đưa về trang Add Customer với thông tin của khách hàng cần sửa.

Bước 2: Nhân viên chỉnh sửa thông tin khách hàng và nhấn sửa.

Bước 3: Thông tin khách hàng được cập nhật vào View Customer và database.

2. Xóa khách hàng:

Bước 1: Sau khi chọn khách hàng cần xóa ở mục View Customer. Nhân viên sẽ được hệ thống đưa về trang Add Customer với thông tin của khách hàng cần xóa.

Bước 2: Nhân viên nhấn xóa để xóa khách hàng.

Bước 3: Thông tin khách hàng được cập nhật vào View Customer và database.

Bảng 3. 11 Bảng đặc tả giao diện quản lý khách hàng

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	Thoát khỏi trang quản lý nhân viên
2	TextView1	TextView	Không
3	TextView2	TextView	Tên thông tin cần điền
4	TextView10	TextView	Tên thông tin cần điền
5	TextView3	TextView	Tên thông tin cần điền
6	TextView4	TextView	Tên thông tin cần điền
7	TextView5	TextView	Tên thông tin cần điền
8	TextView6	TextView	Tên thông tin cần điền
9	TextView7	TextView	Tên thông tin cần điền
10	TextView8	TextView	Tên thông tin cần điền
11	TextView9	TextView	Tên thông tin cần điền
12	txtmkh	Plain Text	Điền thông tin khách hàng
13	txtname	Plain Text	Điền thông tin khách hàng
14	txtfare	Plain Text	Điền thông tin khách hàng

15	txtphone	Plain Text	Điền thông tin khách hàng
16	spinner1	Spinner	Chọn trang quản lý hoặc xem danh sách khách hàng
17	spinner_age	Spinner	Chọn thông tin
18	spinner_mdv	Spinner	Chọn thông tin
19	spinner_mdve	Spinner	Chọn thông tin
20	spinner_buses	Spinner	Chọn thông tin
21	btnadd	Button	Thêm khách hàng
22	btnupdate	Button	Sửa khách hàng
23	btndelete	Button	Xoá khách hàng
24	rdiomale	Radio	Chọn giới tính
25	rdiofemale	Radio	Chọn giới tính

### 3.3.12 Giao diện quản lý dịch vụ

11:35

← Selec..

## QUẢN LÝ DỊCH VỤ SERVICE MANAGEMENT

MDV:

Tên dịch vụ:

Giá vé  VNĐ

**Thêm dịch vụ** **Sửa** **Xóa**

Hình 3. 12 Giao diện quản lý dịch vụ

#### 1. Thêm dịch vụ:

Bước 1: Điền đầy đủ thông tin của dịch vụ.

Bước 2: Nhấn “Thêm” để thêm dịch vụ vào danh sách và cập nhật vào database.

#### 2. Sửa thông tin dịch vụ:

Bước 1: Sau khi chọn dịch vụ cần sửa ở mục View Service. Nhân viên sẽ được hệ thống đưa về trang Add Service với thông tin của dịch vụ cần sửa.

Bước 2: Nhân viên chỉnh sửa thông tin dịch vụ và nhấn sửa.

Bước 3: Thông tin khách hàng được cập nhật vào View Service và database.

#### 2. Xóa dịch vụ:

Bước 1: Sau khi chọn dịch vụ cần xoá ở mục View Sevice. Nhân viên sẽ được hệ thống đưa về trang Add Service với thông tin của dịch vụ cần xoá.

Bước 2: Nhân viên nhấn xoá để xoá dịch vụ.

Bước 3: Thông tin dịch vụ được cập nhật vào View Sevice và database.

Bảng 3. 12 Bảng đặc tả giao diện quản lý dịch vụ

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	Thoát khỏi trang quản lý nhân viên
2	TextView1	TextView	Không
3	TextView2	TextView	Tên thông tin cần điền
4	TextView10	TextView	Tên thông tin cần điền
5	TextView3	TextView	Tên thông tin cần điền
6	TextView4	TextView	Tên thông tin cần điền
7	txtmkh	Plain Text	Điền thông tin dịch vụ
8	txtname	Plain Text	Điền thông tin dịch vụ
9	txtfare	Plain Text	Điền thông tin dịch vụ
10	btnadd	Button	Thêm dịch vụ
11	btnupdate	Button	Sửa dịch vụ
12	btndelete	Button	Xoá dịch vụ
13	spinner1	Spinner	Chọn trang quản lý hoặc xem danh sách dịch vụ



### 3.3.13 Giao diện quản lý đặt vé

11:34

Selec..

## QUẢN LÝ ĐẶT VÉ TICKETS MANAGEMENT

MDVE: Ticket ID      MKH: Customer ID

Tên khách hàng: Customer name

Tuổi:                           ☐ Nam      ☐ Nữ

Tên nhân viên: Fullname

Tuổi:                           ☐ Nam      ☐ Nữ

Số điện thoại: Phone Number

Giá vé: Fare      VNĐ

Tuyến xe: Bến xe Mỹ Đình - Bến xe An..      ▼

Số ghế: 4 chỗ      ▼      MDV: DV01      ▼

**Thêm**      **Sửa**      **Xóa**

Selected Item: 4 chỗ

Hình 3. 13 Giao diện quản lý đặt vé

#### 1. Thêm đặt vé:

Bước 1: Điền đầy đủ thông tin của đặt vé.

Bước 2: Nhấn “Thêm” để thêm đặt vé vào danh sách và cập nhật vào database.

#### 2. Sửa thông tin đặt vé:

Bước 1: Sau khi chọn đặt vé cần sửa ở mục View Ticket. Nhân viên sẽ được hệ thống đưa về trang Add Ticket với thông tin của vé cần sửa.

Bước 2: Nhân viên chỉnh sửa thông tin đặt vé và nhấn sửa.

Bước 3: Thông tin vé được cập nhật vào View Ticket và database.

#### 2. Xóa vé:

Bước 1: Sau khi chọn vé cần xoá ở mục View Ticket. Nhân viên sẽ được hệ thống đưa về trang Add Ticket với thông tin của dịch vụ cần xoá.

Bước 2: Nhân viên nhấn xoá để xoá vé.

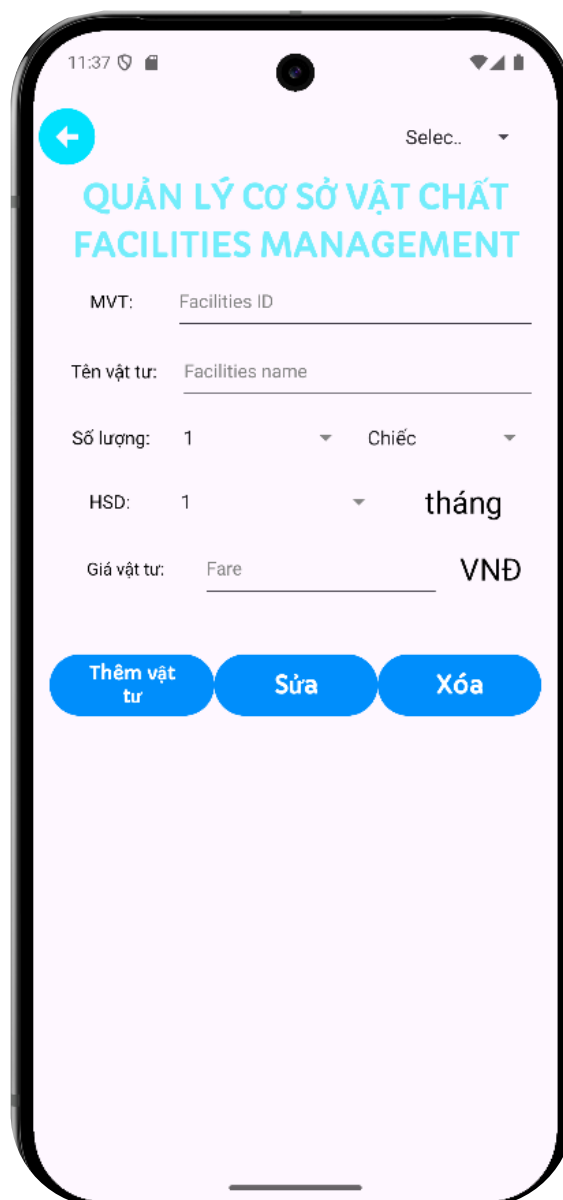
Bước 3: Thông tin vé được cập nhật vào View Ticket và database.

Bảng 3. 13 Bảng đặc tả giao diện quản lý đặt vé

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	Thoát khỏi trang quản lý nhân viên
2	TextView1	TextView	Không
3	TextView2	TextView	Tên thông tin cần điền
4	TextView10	TextView	Tên thông tin cần điền
5	TextView3	TextView	Tên thông tin cần điền
6	TextView4	TextView	Tên thông tin cần điền
7	TextView5	TextView	Tên thông tin cần điền
8	TextView6	TextView	Tên thông tin cần điền
9	TextView7	TextView	Tên thông tin cần điền
10	TextView8	TextView	Tên thông tin cần điền
11	TextView9	TextView	Tên thông tin cần điền
12	TextView11	TextView	Tên thông tin cần điền
13	txtmkh	Plain Text	Điền thông tin đặt vé
14	txtename	Plain Text	Điền thông tin đặt vé
15	txtfare	Plain Text	Điền thông tin đặt vé
16	txtmdve	Plain Text	Điền thông tin đặt vé
17	txtphone	Plain Text	Điền thông tin đặt vé
18	txtcusname	Plain Text	Điền thông tin đặt vé
19	txteage	Plain Text	Điền thông tin đặt vé
20	txtcage	Plain Text	Điền thông tin đặt vé
21	btnadd	Button	Thêm vé
22	btnupdate	Button	Sửa vé

23	btndelete	Button	Xoá vé
24	spinner1	Spinner	Chọn trang quản lý hoặc xem danh sách đặt vé
25	spinner_mdv	Spinner	Chọn mã dịch vụ
26	spinne_buses	Spinner	Chọn tuyến xe khách
27	spinner_seat	Spinner	Chọn xe

### 3.3.13 Giao diện quản lý cơ sở vật chất



Hình 3. 14 Giao diện quản lý cơ sở vật chất

#### 1. Thêm cơ sở vật chất:

Bước 1: Điền đầy đủ thông tin của cơ sở vật chất.

Bước 2: Nhấn “Thêm” để thêm cơ sở vật chất vào danh sách và cập nhật vào database.

## 2. Sửa thông tin cơ sở vật chất:

Bước 1: Sau khi chọn cơ sở vật chất cần sửa ở mục View Facility. Nhân viên sẽ được hệ thống đưa về trang Add Ticket với thông tin của cơ sở vật chất cần sửa.

Bước 2: Nhân viên chỉnh sửa thông tin cơ sở vật chất và nhấn sửa.

Bước 3: Thông tin cơ sở vật chất được cập nhật vào View Facility và database.

## 2. Xóa cơ sở vật chất:

Bước 1: Sau khi chọn cơ sở vật chất cần xóa ở mục View Facility. Nhân viên sẽ được hệ thống đưa về trang Add Facility với thông tin của cơ sở vật chất cần xóa.

Bước 2: Nhân viên nhấn sự nhấn xóa để xóa cơ sở vật chất.

Bước 3: Thông tin cơ sở vật chất được cập nhật vào View Facility và database.

Bảng 3. 14 Bảng đặc tả giao diện quản lý cơ sở vật chất

STT	Tên đối tượng	Kiểu	Ý nghĩa
1	ImageView	ImageView	Thoát khỏi trang quản lý nhân viên
2	TextView1	TextView	Không
3	TextView2	TextView	Tên thông tin cần điền
4	TextView10	TextView	Tên thông tin cần điền
5	TextView3	TextView	Tên thông tin cần điền
6	TextView4	TextView	Tên thông tin cần điền
7	TextView5	TextView	Tên thông tin cần điền
8	TextView6	TextView	Tên thông tin cần điền
9	TextView7	TextView	Tên thông tin cần điền
10	txtmvt	Plain Text	Điền thông tin cơ sở vật chất
11	txtname	Plain Text	Điền thông tin cơ sở vật chất
12	txtfare	Plain Text	Điền thông tin cơ sở vật chất
13	btnadd	Button	Thêm csvc
14	btnupdate	Button	Sửa csvc
15	btndelete	Button	Xóa csvc

16	spinner1	Spinner	Chọn trang quản lý hoặc xem danh sách csvc
17	spinner_sl	Spinner	Chọn thông tin cơ sở vật chất
18	spinne_hsd	Spinner	Chọn thông tin cơ sở vật chất
19	spinner_loai	Spinner	Chọn thông tin cơ sở vật chất

## CHƯƠNG 4: KẾT LUẬN

### 4.1 Kết quả đạt được

- Tìm hiểu rõ về Android Studio
- Tìm hiểu đầy đủ về PHP
- Tìm hiểu đầy đủ về MySQL
- Phân tích thiết kế hệ thống đầy đủ
- Giao diện hệ thống được thiết kế đơn giản thân thiện

Kết quả mà chúng em đạt được tuy chưa được tốt nhưng cũng đã là thành quả của sự cố gắng của cả nhóm. Qua đây chúng em đã có thêm kiến thức về học phần. Chúng em đã biết thêm rất nhiều kiến thức mới, biết cách phân chia công việc thêm phần công bằng và phù hợp hơn.

Ứng dụng này có thể dễ dàng mở rộng thêm các tính năng khác như tích hợp hệ thống thanh toán trực tuyến hay quản lý dịch vụ bổ sung cho bến xe. Nhìn chung, dự án đã thể hiện khả năng ứng dụng hiệu quả của lập trình hướng đối tượng trong việc giải quyết các bài toán nghiệp vụ thực tiễn.

### 4.2 Hạn chế

- Giao diện người dùng chưa trực quan.
- Quản lý dữ liệu chưa tích hợp với cơ sở dữ liệu lớn.
- Thiếu các tính năng mở rộng như thanh toán trực tuyến.
- Một số phần nội dung chưa được hoàn thiện hoặc cập nhật đầy đủ, chưa thể tiếp cận người dùng.

### 4.3 Hướng phát triển

- Bài báo cáo là quá trình thực hiện việc nghiên cứu và phát triển hệ thống quản lý bến xe khách. Hiện tại, nhóm chúng em đã phân tích và xây dựng các chức năng chính của hệ thống, bên cạnh đó cũng đã thiết kế giao diện của hệ thống. Vì vậy hướng phát triển tiếp theo là đi đến việc xây dựng một hệ thống hoàn chỉnh dựa vào kết quả báo cáo trên.. Tuy nhiên còn một số chức năng mà hệ thống cần có như: quản lý cơ sở vật chất, quản lý dự vụ ( đồ ăn, nước uống),... Từ đó xây dựng một hệ thống hoàn chỉnh mang tính ứng dụng cao và được sử dụng rộng rãi.
- Tích hợp thanh toán nhiều hình thức, sử dụng trên cả điện thoại và những thiết bị khác
- Tối ưu hoá tìm kiếm
- Xuất được hoá đơn thanh toán cho khách hàng
- Đưa ra được bảng thống kê doanh thu

## TÀI LIỆU THAM KHẢO

1. Professional Mobile Application Development – Jeff McWherter and Scott Gowell (ISBN:978-1118203903).
2. Android Programming Tutorials: Easy-To-Follow Training-Style Exercises on Android Application Development – Mark Lawrence Murphy (ISBN: 978-0981678023).
3. Christian Keur, Aaron Hillegass, iOS Programming: The Big Nerd Ranch Guide, Big Nerd Ranch, Inc, 2015.
4. Wei Meng Lee, Beginning Android Development
5. <https://developer.android.com/guide/>
6. Gail Rahn Frederick, Rajesh Lal, Phát triển web cho smartphone, 2016.
7. <https://codelearn.io/sharing/xu-huong-phat-trien-mobile-app-tam-nhin-toi-2023>
8. <https://appmaster.io/vi/blog/xu-huong-phat-trien-ung-dung-di-dong-hang-dau>
9. [https://vi.wikipedia.org/wiki/H%E1%BB%87\\_%C4%91i%E1%BB%81u\\_h%C3%A0nh\\_di\\_%C4%91%E1%BB%99ng](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh_di_%C4%91%E1%BB%99ng)
10. [https://vi.wikipedia.org/wiki/Android\\_\(h%E1%BB%87\\_%C4%91i%E1%BB%81u\\_h%C3%A0nh\)#:~:text=Android%20l%C3%A0%20m%E1%BB%99t%20h%E1%BB%87%20%C4%91i%E1%BB%81u,mua%20l%E1%BA%A1i%20v%C3%A0o%20n%C4%83m%202005](https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)#:~:text=Android%20l%C3%A0%20m%E1%BB%99t%20h%E1%BB%87%20%C4%91i%E1%BB%81u,mua%20l%E1%BA%A1i%20v%C3%A0o%20n%C4%83m%202005)

## PHỤ LỤC

### BẢNG PHÂN CÔNG NHIỆM VỤ VÀ ĐÁNH GIÁ KẾT QUẢ CÁC THÀNH VIÊN TRONG NHÓM

STT	Họ và tên	Vai trò	Nhiệm vụ	Đánh giá
1	Đỗ Nhật Minh	Nhóm trưởng	- Chịu trách nhiệm phân công, đảm bảo tiến độ, chất lượng - Thực hiện code giao diện - Hỗ trợ test	35%
2	Phạm Tuấn Anh	Thành viên chính	- Thực hiện code các chức năng giao diện - Thực hiện liên kết database	35%
3	Hoàng Tiến Huy	Thành viên chính	- Thực hiện code các chức năng giao diện	15%
4	Nguyễn Công Minh	Thành viên	- Thực hiện hoàn thiện tổng hợp báo cáo - Hỗ trợ test	15%