

**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN**



**BÁO CÁO ĐỒ ÁN**  
**MÔN: PHÂN TÍCH THIẾT KẾ PHẦN MỀM**

**ĐỀ TÀI**  
**VIẾT MÔ TẢ DỰ ÁN VÀ TẠO USE CASE DIAGRAM**  
**BẰNG STARUML**

GVHD: TS. Nguyễn Tấn Cầm

Nhóm sinh viên thực hiện:

1. Hoàng Gia Minh

MSSV: 22520861

2. Võ Nhật Phương

MSSV: 22521172

**TP. Hồ Chí Minh, tháng 4 năm 2024**

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

....., ngày... tháng.....năm 20...

**Người nhận xét**

*(Ký tên và ghi rõ họ tên)*

## BẢNG PHÂN CÔNG, ĐÁNH GIÁ THÀNH VIÊN:

*Bảng 1: Bảng phân công, đánh giá thành viên*

Họ và tên	MSSV	Phân công	Đánh giá
Hoàng Gia Minh	22520861	- Tuần 3: + Tìm hiểu đề tài: Viết mô tả dự án và tạo Use case diagram bằng StarUML về Ứng dụng đặt chỗ đỗ xe  + Mô tả dự án  + Tìm hiểu các chức năng cần có cho dự án Ứng dụng đặt chỗ đỗ xe  + Mục tiêu đề tài	- Tuần 3: 100%
		- Tuần 4: + Use case diagram  + Các bước xây dựng Use case diagram	- Tuần 4: 100%
		+ Vẽ Use case diagram Ứng dụng đặt chỗ đỗ xe	- Tuần 5: 100%
		- Tuần 5: + Đặc tả Use case trong Use case diagram Quản lý bãi đỗ xe  + Đặc tả Use case trong Use case diagram Quản lý đơn đặt chỗ đỗ xe	- Tuần 6: 100%
			- Tuần 7: 100%

		<ul style="list-style-type: none"> <li>- Tuần 6: <ul style="list-style-type: none"> <li>+ Activity Diagram</li> <li>+ Sequence Diagram</li> <li>+ Ưu điểm, Phương hướng phát triển</li> </ul> </li> <li>- Tuần 7: <ul style="list-style-type: none"> <li>+ Slide</li> </ul> </li> </ul>	
Võ Nhật Phương	22521172	<ul style="list-style-type: none"> <li>- Tuần 3: <ul style="list-style-type: none"> <li>+ Tìm hiểu đề tài: Viết mô tả dự án và tạo Use case diagram bằng starUML về Ứng dụng đặt chỗ đỗ xe</li> <li>+ Giới thiệu đề tài</li> <li>+ Lời cảm ơn</li> <li>+ Tìm hiểu các chức năng cần có cho dự án Ứng dụng đặt chỗ đỗ xe</li> </ul> </li> <li>- Tuần 4: <ul style="list-style-type: none"> <li>+ Các thành phần của Use case diagram</li> <li>+ Vẽ Use case diagram Ứng dụng đặt chỗ đỗ xe</li> </ul> </li> <li>- Tuần 5: <ul style="list-style-type: none"> <li>+ Đặc tả Use case trong Use case</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>- Tuần 3: 100%</li> <li>- Tuần 4: 100%</li> <li>- Tuần 5: 100%</li> <li>- Tuần 6: 100%</li> <li>- Tuần 7: 100%</li> </ul>

		<p>diagram Đặt chỗ đỗ xe</p> <p>+ Đặc tả Use case trong Use case diagram Quản lý tài khoản</p> <p>- Tuần 6:</p> <p>+ Class Diagram</p> <p>+ Sequence Diagram</p> <p>+ Nhược điểm, Kết luận</p> <p>- Tuần 7:</p> <p>+ Slide</p>	
--	--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

## LỜI CẢM ƠN

Trong thời gian học tập và làm báo cáo đề tài “VIẾT MÔ TẢ DỰ ÁN VÀ TẠO USE CASE DIAGRAM BẰNG STARUML” này, chúng em muốn gửi những lời cảm ơn và biết ơn chân thành nhất tới tất cả những người đã hỗ trợ, giúp đỡ nhóm về kiến thức và tinh thần trong quá trình thực hiện đồ án.

Trước hết, nhóm chúng em xin gửi lời cảm ơn tới thầy TS. Nguyễn Tấn Cầm - Giảng viên Khoa Khoa học và Kỹ thuật Thông tin, Trường Đại học Công nghệ Thông tin ĐHQG HCM. Thầy là người đã trực tiếp hướng dẫn, nhận xét, giúp đỡ chúng em trong suốt quá trình thực hiện đồ án.

Chúng em cũng xin gửi lời cảm ơn đến gia đình, bạn bè, người thân đã giúp đỡ động viên rất nhiều trong quá trình học tập và làm đồ án đúng thời gian quy định.

Do thời gian thực hiện có hạn, kiến thức còn nhiều hạn chế nên đồ án thực hiện chắc chắn không tránh khỏi những thiếu sót nhất định. Chúng em rất mong nhận được ý kiến đóng góp của thầy cô giáo và các bạn để nhóm có thêm kinh nghiệm và tiếp tục hoàn thiện đồ án của mình.

Nhóm chúng em xin chân thành cảm ơn tất cả mọi người!

*TP. Hồ Chí Minh, ngày 14 tháng 4 năm 2024*

# MỤC LỤC

<b>LỜI CẢM ƠN.....</b>	<b>6</b>
<b>CHƯƠNG 1: TỔNG QUAN DỰ ÁN.....</b>	<b>10</b>
1. Giới thiệu đề tài.....	10
2. Mục tiêu đề tài.....	10
3. Mô tả dự án.....	11
<b>CHƯƠNG 2: USE CASE DIAGRAM.....</b>	<b>14</b>
1. Tổng quan về Use Case Diagram.....	14
1.1. Use case diagram.....	14
1.2. Các thành phần của Use case diagram.....	15
1.2.1. Use case.....	15
1.2.2. Actor.....	17
1.2.3. Communication link.....	17
1.2.4. Boundary of system.....	19
1.3. Các bước xây dựng Use case diagram.....	19
2. Vẽ Use Case Diagram cho Ứng dụng đặt chỗ đỗ xe.....	21
2.1. Use case diagram Hệ thống tìm bãi đỗ xe.....	21
2.2. Use case diagram Quản lý bãi đỗ xe.....	22
2.3. Use case diagram Quản lý đơn đặt chỗ đỗ xe.....	23
2.4. Use case diagram Đặt chỗ đỗ xe.....	24
2.5. Use case diagram Quản lý tài khoản.....	25
3. Đặc tả Use case.....	25
3.1 Use case diagram Quản lý bãi đỗ xe.....	25
3.1.1 Use case Xem danh sách bãi đỗ xe.....	25
3.1.2 Use case Thêm bãi đỗ.....	26
3.1.3 Use case Xóa bãi đỗ.....	27
3.1.4 Use case Cập nhật thông tin bãi đỗ.....	28
3.1.5 Use case Xem danh sách chỗ đỗ xe.....	29
3.1.6 Thêm chỗ đỗ xe mới.....	29
3.1.7 Use case Xóa chỗ đỗ xe.....	30
3.1.8 Use case Cập nhật thông tin chỗ đỗ xe.....	31
3.1.9. Use case Xem trạng thái bãi đỗ.....	32
3.1.10 Use case Cập nhật trạng thái bãi đỗ.....	33

3.2 Use case diagram Quản lý đơn đặt chỗ đỗ xe.....	34
3.2.1 Xem danh sách đơn đặt chỗ đỗ xe.....	34
3.2.2 Cập nhật trạng thái đơn đặt chỗ đỗ xe.....	35
3.2.3 Use case Xem chi tiết các đơn đặt chỗ đỗ xe.....	36
3.2.4 Xem doanh số đặt chỗ đỗ xe.....	36
3.2.5 Xem danh sách bãi đỗ xe yêu thích.....	37
3.3 Use case diagram Đặt chỗ đỗ xe.....	38
3.3.1 Use case Xem danh sách bãi đỗ xe.....	38
3.3.2 Tìm kiếm bãi đỗ xe.....	39
3.3.3 Use case Xem chi tiết bãi đỗ xe.....	39
3.3.4 Use case Đánh giá.....	40
3.3.5 Use case Đặt chỗ đỗ xe.....	41
3.3.6 Use case Xem lịch sử đặt chỗ đỗ xe.....	43
3.4 Use case diagram Quản lý tài khoản.....	43
3.4.1 Use case Đăng nhập.....	43
3.4.2 Use case Đăng xuất.....	44
3.4.3 Use case Đăng ký.....	45
3.4.4 Use case Quên mật khẩu.....	46
3.4.5 Xem thông tin chi tiết tài khoản.....	47
3.4.6 Use case Chỉnh sửa thông tin.....	48
3.4.7 Use case Xem thông báo.....	48
3.4.8 Xem danh sách thông tin tài khoản.....	49
3.4.9 Xem chi tiết thông tin tài khoản.....	50

### **CHƯƠNG 3: ACTIVITY DIAGRAM, CLASS DIAGRAM, SEQUENCE**

<b>DIAGRAM.....</b>	<b>51</b>
1. Activity Diagram.....	51
1.1 Tổng quan về Activity Diagram.....	51
1.2. Các thành phần cơ bản của Activity Diagram:.....	51
1.3 Cách vẽ Activity Diagram.....	53
1.4 Activity Diagram cho Use case Đặt chỗ đỗ xe.....	55
2. Class Diagram.....	56
2.1 Tổng quan về Class Diagram.....	56
2.2 Các tính chất cơ bản của Class Diagram.....	56
2.3 Mối quan hệ trong Class Diagram.....	57



2.4 Access Modifier trong Class Diagram.....	57
2.5 Multiplicity trong Class Diagram.....	58
2.6 Các bước xây dựng Class Diagram.....	58
2.7 Class Diagram trong Ứng dụng đặt chỗ đỗ xe.....	59
3. Sequence Diagram.....	59
3.1. Tổng quan về Sequence Diagram.....	59
3.2. Các thành phần cơ bản của Sequence Diagram:.....	60
3.3. Cách vẽ Sequence Diagram.....	62
3.4 Sequence Diagram của Use case Đặt chỗ đỗ xe.....	63
<b>CHƯƠNG 4: ƯU, NHƯỢC ĐIỂM, HƯỚNG PHÁT TRIỂN HỆ THỐNG.....</b>	<b>65</b>
1. Ưu điểm.....	65
2. Nhược điểm.....	66
3. Phương hướng phát triển.....	66
<b>CHƯƠNG 5: KẾT LUẬN.....</b>	<b>68</b>
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>69</b>

# CHƯƠNG 1: TỔNG QUAN DỰ ÁN

## 1. Giới thiệu đề tài

Thời đại ngày nay, dân số tăng nhanh đang là một vấn đề đáng quan tâm, đặc biệt là tại các thành phố lớn. Dân cư trở nên đông đúc hơn dẫn đến nhu cầu sử dụng phương tiện giao thông cũng tăng cao. Sự gia tăng dân số không chỉ ảnh hưởng đến cơ sở hạ tầng, môi trường sống mà còn đặt ra nhiều thách thức trong việc cung cấp các dịch vụ công cộng và tiện ích cho cư dân.

Với mật độ dân cư ngày càng tăng, nhu cầu sử dụng phương tiện giao thông như ô tô, xe máy, xe bus cũng tăng lên đáng kể. Điều này không chỉ gây áp lực lên hệ thống giao thông công cộng mà còn tạo ra tình trạng kẹt xe, ô nhiễm môi trường và tai nạn giao thông. Trong các đô thị lớn, việc tìm thấy một chỗ đỗ xe cho xe ô tô không phải điều dễ dàng. Chính vì thế việc tạo ra một ứng dụng hỗ trợ tìm bãi đỗ xe cho ô tô trở thành nên vô cùng cấp thiết.

Với lý do đó, nhóm chúng em đã chọn đề tài “VIẾT MÔ TẢ DỰ ÁN VÀ TẠO USE CASE DIAGRAM ỨNG DỤNG ĐẶT CHỖ ĐỖ XE” để tìm hiểu tổng quan và tìm ra cách giải quyết vấn đề này.

## 2. Mục tiêu đề tài

Dự án này nhằm mục đích phát triển một hệ thống đặt chỗ đỗ xe hiệu quả, dễ sử dụng và linh hoạt thông qua việc viết mô tả dự án và vẽ Use Case Diagram bằng StarUML. Mục tiêu chính của dự án bao gồm:

- Mô tả dự án chi tiết: Xây dựng một tài liệu mô tả dự án đầy đủ và chi tiết, bao gồm mô tả các yêu cầu, chức năng, tính năng và các thành phần cần thiết cho hệ thống đặt chỗ đỗ xe.

- Phân tích yêu cầu: Hiểu rõ các yêu cầu của khách hàng và người dùng, cung cấp các chức năng và tính năng phù hợp để đáp ứng nhu cầu thực tế và cải thiện trải nghiệm người dùng.
- Thiết kế Use Case Diagram: Sử dụng StarUML để thiết kế và vẽ Use Case Diagram, thể hiện các tác nhân, các Use Case và các tương tác giữa chúng, nhằm mô tả rõ ràng luồng làm việc của hệ thống đặt chỗ đỗ xe.
- Tối ưu hóa trải nghiệm người dùng: Xác định và thiết kế các tính năng như tìm kiếm, đặt chỗ, quản lý tài khoản và các tính năng gợi ý để cung cấp trải nghiệm người dùng tốt nhất.
- Phát triển hệ thống linh hoạt và mở rộng: Thiết kế hệ thống có khả năng linh hoạt và mở rộng, dễ dàng thích nghi với các yêu cầu mới và các tính năng mở rộng trong tương lai.
- Tạo tài liệu chất lượng: Tạo ra tài liệu dự án và tài liệu hướng dẫn sử dụng chất lượng, dễ hiểu và dễ sử dụng cho các nhà phát triển và người dùng cuối.

Tóm lại, mục tiêu của dự án là xây dựng một hệ thống đặt chỗ đỗ xe đáp ứng được các yêu cầu và mong đợi của khách hàng, mang lại giá trị và tiện ích tốt nhất cho người dùng cuối.

### **3. Mô tả dự án**

Chúng ta cần xây dựng một ứng dụng nhằm giúp người dùng tìm kiếm, đặt chỗ và quản lý việc đỗ xe một cách thuận tiện và nhanh chóng. Ứng dụng cũng có các chức năng cho người quản trị để họ có thể quản lý các bãi đỗ xe, quản lý khách hàng, quản lý các đánh giá từ người dùng, ...

Yêu cầu cụ thể:

- Đối với khách hàng:
  - Ứng dụng hỗ trợ tối đa cho khách hàng trong việc tìm kiếm các bãi đỗ xe

để tìm được chỗ đỗ gần nhất.

- Khuyến nghị cho người dùng các bãi đỗ xe có nhiều dịch vụ ưu đãi và được đánh giá cao bởi các khách hàng khác.
- Khách hàng có thể xem danh sách các bãi đỗ xe sau khi hoàn tất việc tìm kiếm cũng như xem thông tin chi tiết của mỗi bãi đỗ như tên, địa chỉ, giá cả, đánh giá, khoảng cách, thời gian hoạt động, sức chứa, các tiện ích kèm theo...
- Ứng dụng cũng cho phép người dùng gửi những bình luận, đánh giá về dịch vụ của bãi đỗ xe để giúp bãi đỗ cải thiện tốt hơn.
- Sau khi tìm được bãi đỗ thích hợp, khách hàng có thể chọn đặt chỗ với những yêu cầu cụ thể. Khi đưa xe ra khỏi bãi đỗ, hệ thống sẽ tính toán số tiền mà khách hàng phải trả theo giá cả và thời gian đỗ. Hệ thống sẽ hiển thị chi tiết hóa đơn và cung cấp nhiều phương thức thanh toán như thanh toán trên công thanh toán, bằng thẻ tín dụng, ví điện tử...
- Ứng dụng cũng hiển thị các thông báo nhắc nhở cho khách hàng về thời gian bãi đỗ đóng cửa hay thông báo về các chương trình ưu đãi để khách hàng có trải nghiệm tốt nhất.
- Hệ thống lưu lại lịch sử đỗ xe và thống kê doanh số của bãi đỗ xe theo tuần, theo tháng... để tổng hợp và phân tích nhu cầu sử dụng của khách hàng nhằm cải tiến chiến lược phát triển của mình.
- Để trải nghiệm ứng dụng một cách tốt nhất, khách hàng có thể đăng ký cho mình một tài khoản và đăng nhập vào tài khoản đó mỗi lần sử dụng. Khách hàng được phép quản lý tài khoản của mình thông qua các tác vụ như xem thông tin tài khoản, cập nhật thông tin tài khoản, quản lý mật khẩu...

- Đối với người quản trị:

- Ứng dụng cho phép người quản trị thực hiện các chức năng quản lý như: quản lý bãi đỗ xe, quản lý tài khoản, quản lý đơn đặt chỗ đỗ xe...

- Ngoài ra, ứng dụng còn hỗ trợ các chức năng liên quan đến tổng hợp thống kê:
  - Thống kê doanh thu các bãi đỗ xe, số lượt cho gửi mỗi tháng, hiển thị dạng biểu đồ
  - Thống kê các bình luận, đánh giá về các bãi đỗ xe
- Yêu cầu về bảo mật
  - Tất cả các form nhập phải được kiểm soát dữ liệu một cách hợp lý
  - Khách hàng chưa đăng nhập thì không được bình luận về dịch vụ
  - Chỉ các nhân viên quản trị mới được thực hiện các chức năng quản trị

## CHƯƠNG 2: USE CASE DIAGRAM

### 1. Tổng quan về Use Case Diagram

#### 1.1. Use case diagram

Use case diagram trong Unified Modeling Language (UML) là một công cụ mô hình hóa quan trọng cung cấp cái nhìn tổng quan về các tương tác giữa người dùng và hệ thống. Nó giúp các bên liên quan hiểu rõ mục tiêu và phạm vi của dự án phần mềm.

Một use case diagram gồm các thành phần sau:

- Actor: Đại diện cho người dùng hoặc các thực thể bên ngoài có khả năng tương tác với hệ thống.
- Use case: Đại diện cho các chức năng hoặc tính năng mà các actors sẽ sử dụng để tương tác với hệ thống. Mỗi use case biểu diễn một tác vụ hoặc một yêu cầu cụ thể.
- Communication link: Kết nối giữa Actor và Use case để thể hiện sự tương tác giữa hệ thống và người dùng.
- Boundary of system: Thể hiện phạm vi của các Use case và hiển thị ranh giới của hệ thống.

Use case diagram thường được sử dụng ở giai đoạn đầu của quá trình phát triển phần mềm cho các mục đích sau:

- Xác định ngữ cảnh của một hệ thống: Bao gồm các tác nhân (actors) và các tương tác giữa hệ thống và tác nhân đó.
- Thu thập yêu cầu của một hệ thống: Định hình rõ ràng mục tiêu và phạm vi của dự án.
- Xác thực kiến trúc hệ thống: Chỉ ra cách mà các chức năng của hệ thống tương tác với nhau và với các tác nhân bên ngoài.
- Hướng dẫn thực hiện và tạo ra các ca kiểm thử: Chỉ ra các tác vụ cụ thể mà hệ

thống phải thực hiện và cách mà các tác nhân sẽ tương tác với hệ thống, từ đó hỗ trợ việc tạo ra các ca kiểm thử.

- Phát triển bởi các nhà phân tích và các chuyên gia: Đảm bảo rằng các yêu cầu và mô hình hóa phản ánh đúng nhu cầu và yêu cầu của doanh nghiệp hoặc lĩnh vực cụ thể [1].

## **1.2. Các thành phần của Use case diagram**

### **1.2.1. Use case**

Use case là một thuật ngữ được sử dụng phổ biến trong lĩnh vực phát triển phần mềm và quản lý dự án. Nó mô tả một tình huống về cách thức hệ thống sẽ được sử dụng bởi người dùng, bao gồm các chức năng, hoạt động và tương tác giữa hệ thống, người dùng trong một ngữ cảnh nhất định.

Mục đích:

- Hiểu rõ các yêu cầu và đặc điểm của hệ thống.
- Xác định các hành vi và kịch bản sử dụng của người dùng.
- Xác định các yếu tố quan trọng trong việc phát triển phần mềm.
- Cung cấp một cách tiếp cận hợp lý để phân tích và mô tả các tác động của hệ thống.

Cấu trúc cơ bản của một Use Case bao gồm các thành phần sau:

- Tên Use Case: Mô tả ngắn gọn và súc tích về tình huống cụ thể mà Use Case đang mô tả. Ví dụ: "Đăng nhập", "Tìm kiếm sản phẩm", "Thanh toán", v.v.
- Mô tả: Giải thích chi tiết về mục đích và chức năng của Use Case. Mô tả các hoạt động và tương tác giữa hệ thống và người dùng hoặc các thành phần khác trong tình huống được mô phỏng. Làm rõ các bước, luồng làm việc và kết quả mong đợi.
- Tác nhân (Actor): Thành phần bên ngoài tương tác với hệ thống trong Use Case. Có thể là người dùng cuối, một hệ thống khác, hoặc bất kỳ thực thể nào có tương

tác với hệ thống. Mỗi Use Case có ít nhất một tác nhân liên quan.

- **Tiền điều kiện (Precondition):** Trạng thái hoặc điều kiện mà hệ thống phải đáp ứng trước khi Use Case được thực hiện. Thường liên quan đến trạng thái ban đầu của hệ thống hoặc các hoạt động trước đó.
- **Luồng chính (Main sequence):** Mô tả chi tiết về các bước cơ bản để thực hiện Use Case theo thứ tự logic và dễ hiểu.
- **Luồng thay thế (Alternative sequence):** Phương án thay thế cho một số bước trong luồng điều khiển (nếu có).
- **Hậu điều kiện (Postcondition):** Trạng thái hoặc điều kiện mà hệ thống phải đạt được sau khi Use Case được thực hiện thành công. Thường liên quan đến trạng thái cuối cùng của hệ thống hoặc các hoạt động kế tiếp.

Use Case có thể được phân loại theo nhiều cách khác nhau, tùy thuộc vào mục đích sử dụng và ngữ cảnh của dự án. Dưới đây là một số phương pháp phân loại phổ biến:

- Theo vai trò:
  - **Use Case chính:** Mô tả các chức năng cốt lõi và quan trọng nhất của hệ thống, đáp ứng trực tiếp nhu cầu của người dùng.
  - **Use Case hỗ trợ:** Bổ sung và hỗ trợ cho Use Case chính, cung cấp các chức năng phụ hoặc tương tác hỗ trợ.
- Theo tính năng:
  - **Use Case mở rộng:** Mở rộng chức năng của Use Case chính, bổ sung các chức năng hoặc luồng làm việc tùy chọn.
  - **Use Case thay thế:** Cung cấp phương pháp hoặc lựa chọn khác để đạt được mục tiêu tương tự trong Use Case chính.
- Theo tác nhân:
  - **Use Case hệ thống:** Mô tả tương tác giữa hệ thống với các thành phần khác như hệ thống bên ngoài, cơ sở dữ liệu, dịch vụ web, v.v.
  - **Use Case người dùng:** Mô tả hoạt động và tương tác của người dùng với hệ thống.



thống để đạt được mục tiêu cụ thể [2].

### **1.2.2. Actor**

Trong biểu đồ Use case diagram, Actor là một thành phần quan trọng đại diện cho các thực thể bên ngoài hệ thống mà tương tác với các chức năng của hệ thống. Dưới đây là một số điểm cần lưu ý về Actor:

- Actor tương tác trực tiếp với các chức năng của hệ thống
- Tên của mỗi Actor được đặt bằng một danh từ để mô tả rõ vai trò hoặc chức năng của họ trong hệ thống.
- Actor đại diện cho các vai trò hoặc nhóm người trong hoạt động kinh doanh, nhưng không nhất thiết phải là cá nhân cụ thể.
- Một Actor có thể đảm nhận nhiều vai trò khác nhau trong các hệ thống khác nhau, tùy thuộc vào ngữ cảnh cụ thể.
- Actor là người khởi đầu hoặc kích hoạt các Use case bằng cách thực hiện các hành động cụ thể.
- Actor có trách nhiệm cung cấp dữ liệu và tương tác với hệ thống (đầu vào) và họ có kỳ vọng về kết quả hoặc đầu ra từ hệ thống [1].

### **1.2.3. Communication link**

Một đường thẳng có thể đại diện cho tương tác của một Actor với một Use case hoặc tương tác giữa các Use case. Actor có thể kết nối với các Use case và những người khác thông qua một loạt các liên kết giao tiếp đại diện cho các mối quan hệ khác nhau như sau:

- Liên kết (Associations): Quan hệ liên kết biểu diễn các tương tác giữa Actor với Use case.
- Kế thừa (Generalizations):
  - Liên kết kế thừa biểu diễn các mối quan hệ cha con giữa các Use case với nhau.

- Use case con là một sự cải tiến của use case cha.
- Biểu diễn dưới dạng một mũi tên có đầu mũi tên hình tam giác. Use case cha được kết nối ở đầu của mũi tên. Đầu còn lại của mũi tên được kết nối với use case con.
- Mở rộng (Extensions):
  - Liên kết mở rộng biểu diễn các chức năng tùy chọn trong hệ thống. Những liên kết này có thể kết nối Actor với các Use case hoặc nhiều Use case liên quan mà không phụ thuộc vào nhau về chức năng.
  - Biểu diễn bằng một mũi tên có đường gạch chấm. Đầu mũi tên chỉ đến use case cơ sở và use case con được kết nối ở đầu còn lại của mũi tên.
  - Thẻ <<extend>> xác định mối quan hệ mở rộng.
- Bao gồm (Inclusions):
  - Khi một use case được mô tả sử dụng chức năng của một use case khác, mối quan hệ giữa các use case được đặt tên là quan hệ bao gồm hoặc sử dụng.
  - Một mối quan hệ sử dụng từ use case cơ sở đến use case con chỉ ra rằng một trường hợp của use case cơ sở sẽ bao gồm hành vi được chỉ định trong use case con.
  - Biểu diễn bằng một mũi tên có đường gạch chấm. Đầu mũi tên chỉ đến use case con và use case cha được kết nối ở đầu còn lại của mũi tên.
  - Thẻ <<include>> xác định mối quan hệ là một mối quan hệ bao gồm [1][3] .

#### **1.2.4. Boundary of system**

Boundary of system được sử dụng để xác định phạm vi của hệ thống mà chúng ta đang thiết kế. Các đối tượng nằm ngoài hệ thống này có tương tác với hệ thống được xem là các Actor.

- Boundary of system có thể là toàn bộ hệ thống như được xác định trong tài liệu yêu cầu.
- Đối với các hệ thống lớn và phức tạp, mỗi mô-đun có thể là một system boundary. Ví dụ, đối với hệ thống ERP cho một tổ chức, mỗi phân hệ như nhân sự, tính lương, kế toán,... sẽ tương ứng với một boundary of system.
- Có thể hình thành boundary of system cho các trường hợp sử dụng cụ thể cho từng chức năng của hệ thống [1].

#### **1.3. Các bước xây dựng Use case diagram**

Để xây dựng một Use case diagram hoàn chỉnh, chúng ta cần tiến hành 3 bước sau:

- Bước 1: Tìm Actor

Các câu hỏi sau có thể giúp bạn xác định các Actor của hệ thống:

- Ai sử dụng hệ thống?
- Ai cài đặt hệ thống?
- Ai khởi động hệ thống?
- Ai duy trì hệ thống?
- Ai tắt hệ thống?
- Hệ thống nào khác sử dụng hệ thống này?
- Ai nhận thông tin từ hệ thống này?
- Ai cung cấp thông tin cho hệ thống?
- Có điều gì xảy ra tự động vào một thời điểm cụ thể không?

- Bước 2: Xác định các Use case

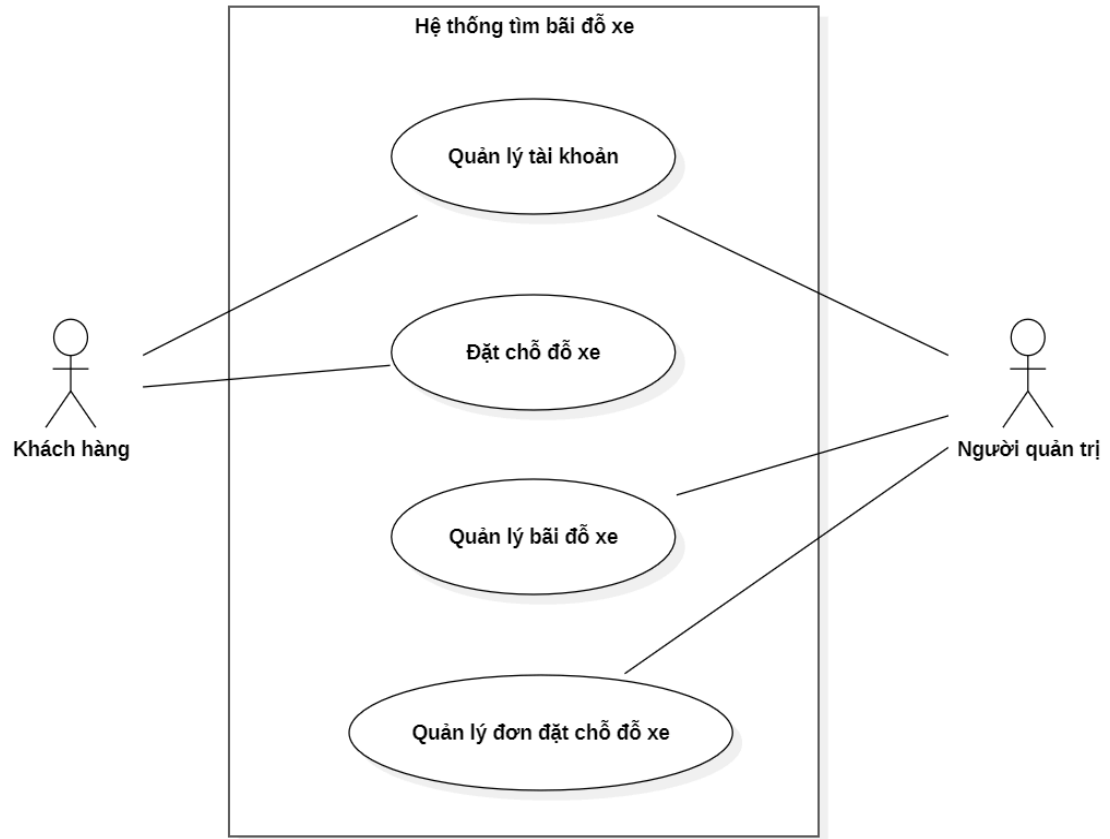
Các câu hỏi sau có thể được đặt để xác định các Use Case, sau khi các Actor đã được xác định):

- Các chức năng nào mà các Actor cần từ hệ thống?
  - Hệ thống có lưu trữ thông tin không? Các Actor nào sẽ tạo, đọc, cập nhật hoặc xóa thông tin này?
  - Hệ thống cần thông báo cho một Actor về các thay đổi trong trạng thái nội bộ không?
  - Có các sự kiện bên ngoài nào mà hệ thống phải biết đến không? Actor nào cần thông báo cho hệ thống về những sự kiện đó?
- Bước 3: Xác định mối quan hệ

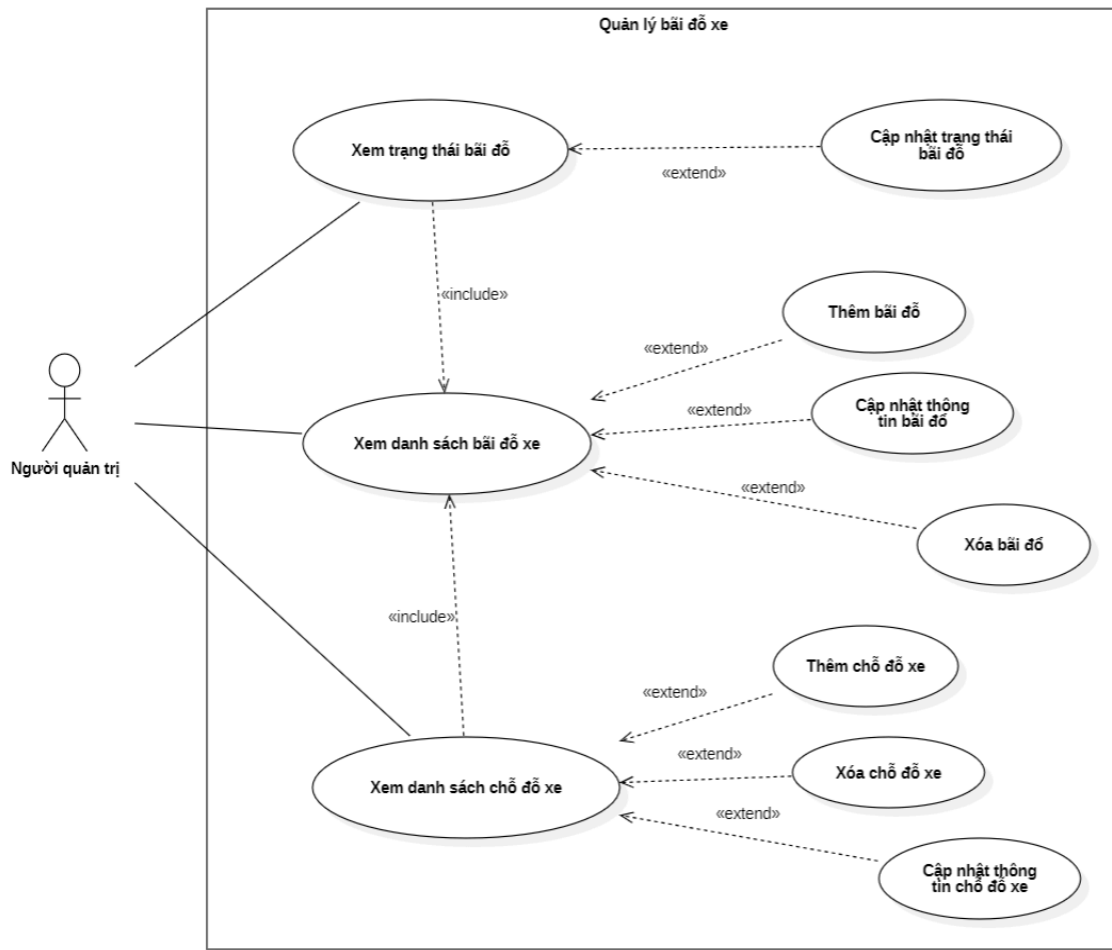
Chúng ta tiến hành phân tích và xác định các loại mối quan hệ giữa các Actor và Use Case, giữa các Actor với nhau cũng như giữa các Use Case với nhau. Sau đó, chúng ta sẽ nối các phần này lại với nhau để tạo thành Use Case Diagram hoàn chỉnh [1].

## 2. Vẽ Use Case Diagram cho Ứng dụng đặt chỗ đỗ xe

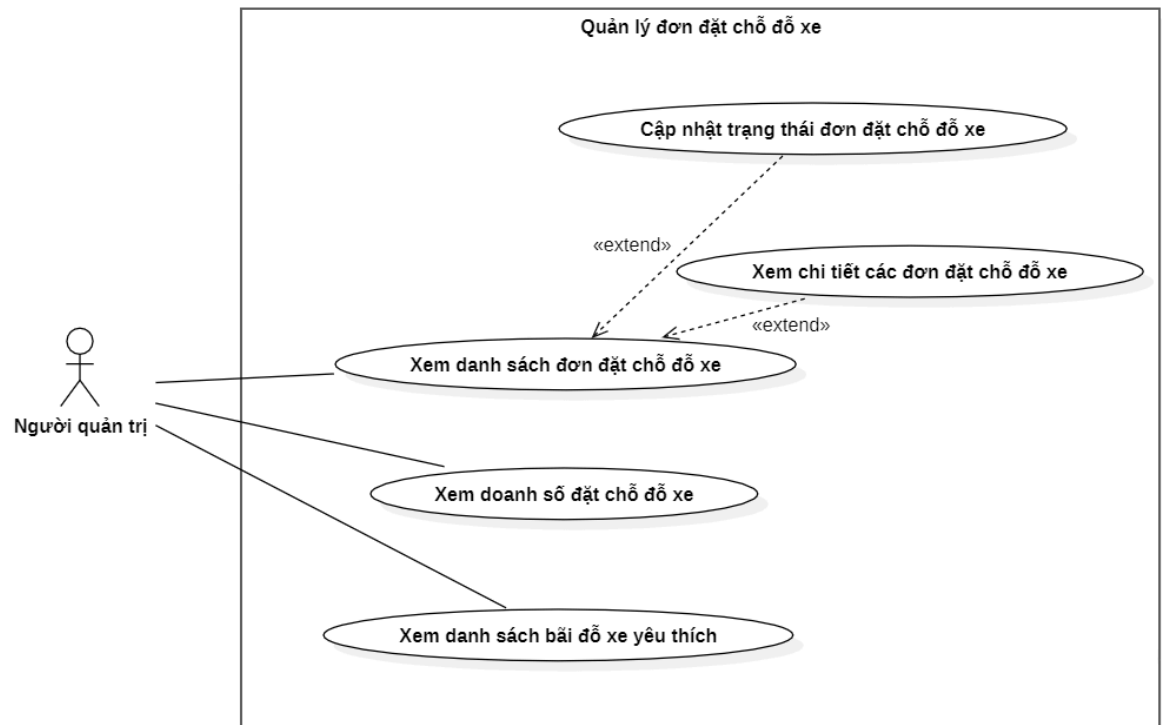
### 2.1. Use case diagram Hệ thống tìm bãi đỗ xe



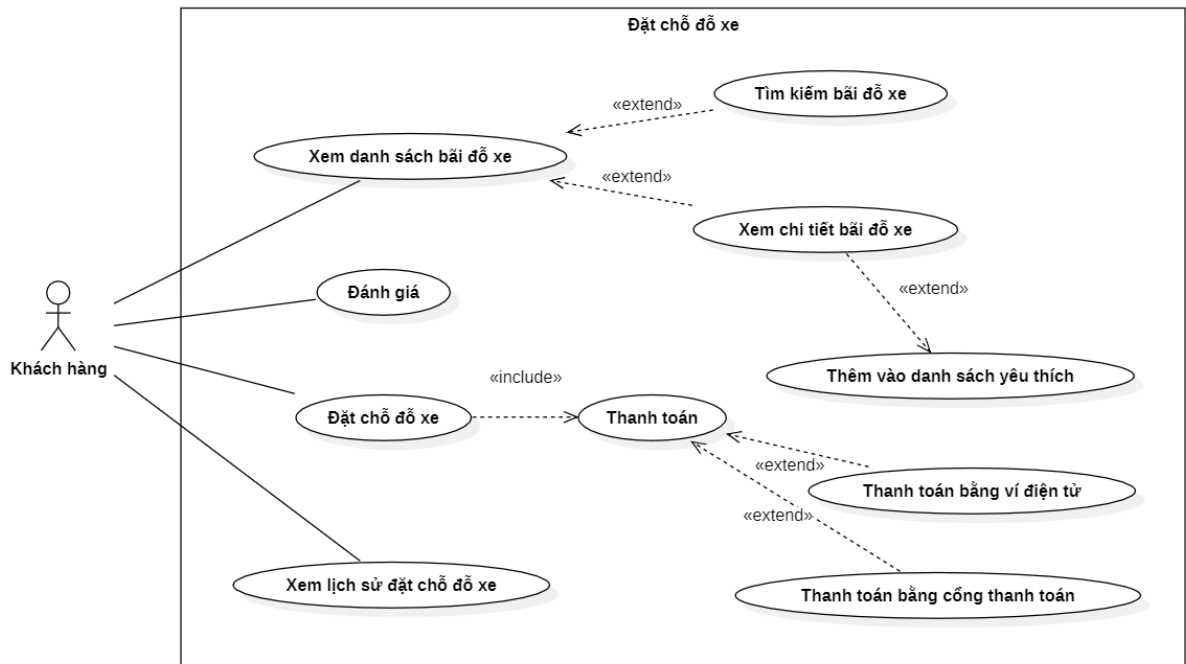
## 2.2. Use case diagram Quản lý bãi đỗ xe



### 2.3. Use case diagram Quản lý đơn đặt chỗ đỗ xe

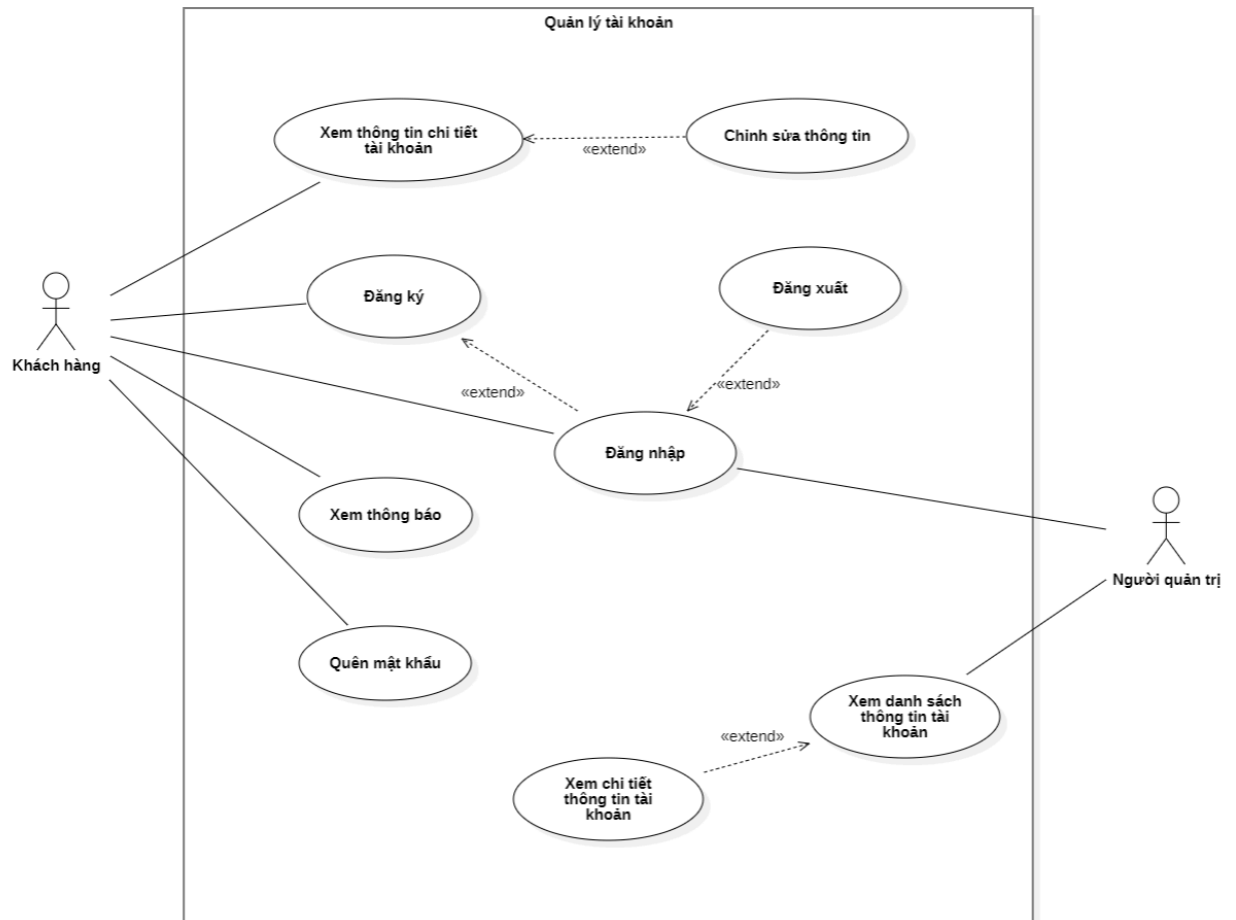


## 2.4. Use case diagram Đặt chỗ đỗ xe





## 2.5. Use case diagram Quản lý tài khoản



## 3. Đặc tả Use case

### 3.1 Use case diagram Quản lý bãi đỗ xe

#### 3.1.1 Use case Xem danh sách bãi đỗ xe

Name	Xem danh sách bãi đỗ xe
Description	Cho phép Người quản trị xem danh sách các bãi đỗ xe có trên hệ thống
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào hệ thống

Postcondition	Người quản trị xem được danh sách bãi đỗ xe có trên hệ thống
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn chức năng “Xem danh sách bãi đỗ xe”</li> <li>2. Hệ thống hiển thị danh sách các bãi đỗ xe có trong hệ thống</li> <li>3. Người quản trị xem danh sách bãi đỗ xe thành công</li> </ol>
Alternative Flow	2.1 Nếu không có bãi đỗ xe nào trong hệ thống thì hiển thị thông báo "Không có bãi đỗ xe nào".

### 3.1.2 Use case Thêm bãi đỗ

Name	Thêm bãi đỗ
Description	Cho phép Người quản trị thêm thông tin của một bãi đỗ xe mới trên ứng dụng
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách bãi đỗ xe
Postcondition	Người quản trị thêm thông tin của bãi đỗ xe thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn vào chức năng “Thêm bãi đỗ xe mới”</li> <li>2. Hệ thống hiển thị form để Người quản trị cập nhật các thông tin về bãi đỗ xe mới</li> <li>3. Người quản trị điền các thông tin về bãi đỗ xe mới bao gồm: Địa chỉ, loại hình, giá cả, số lượng chỗ đỗ,...Nhấn</li> </ol>

	<p>nút “Lưu” để áp dụng thay đổi</p> <p>4. Hệ thống kiểm tra thông tin bãi đỗ xe Người quản trị vừa thêm có hợp lệ hay không.</p> <p>5. Thông tin hợp lệ, hệ thống thêm thông tin bãi đỗ xe mới vào cơ sở dữ liệu của hệ thống và hiển thị thông báo xác nhận việc thêm thành công.</p>
Alternative Flow	5.1 Thông tin không hợp lệ -> Hệ thống yêu cầu Người quản trị điền lại thông tin trước khi tiếp tục.

### 3.1.3 Use case Xóa bãi đỗ

Name	Xóa bãi đỗ
Description	Cho phép Người quản trị xóa thông tin của một bãi đỗ xe đã có trên ứng dụng.
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách bãi đỗ xe
Postcondition	Người quản trị xóa thông tin của bãi đỗ xe thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một bãi đỗ xe cụ thể và chọn vào chức năng “Xóa bãi đỗ xe”</li> <li>2. Hệ thống yêu cầu xác nhận từ người quản trị trước khi thực hiện xóa bãi đỗ xe</li> <li>3. Người quản trị xác nhận việc xóa bãi đỗ xe</li> <li>4. Hệ thống thực hiện xóa bãi đỗ xe và hiển thị thông báo</li> </ol>

	xác nhận việc xóa thành công.
Alternative Flow	2.1 Bãi đỗ xe đó đang có đặt chỗ đang chờ, hệ thống hiển thị cảnh báo và yêu cầu xác nhận lại từ người quản trị trước khi tiếp tục xóa.

### 3.1.4 Use case Cập nhật thông tin bãi đỗ

Name	Cập nhật thông tin bãi đỗ
Description	Cho phép Người quản trị cập nhật thông tin của một bãi đỗ xe cụ thể.
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách bãi đỗ xe
Postcondition	Người quản trị cập nhật thông tin của bãi đỗ xe thành công
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một bãi đỗ xe cụ thể và chọn vào chức năng “Cập nhật thông tin bãi đỗ xe”</li> <li>2. Hệ thống hiển thị form để Người quản trị cập nhật các thông tin về bãi đỗ xe đã chọn</li> <li>3. Người quản trị điền thông tin mới vào các trường cần cập nhật và nhấn nút “Lưu” để áp dụng thay đổi</li> <li>4. Hệ thống kiểm tra thông tin Người quản trị vừa cập nhật có hợp lệ hay không.</li> <li>5. Thông tin hợp lệ, hệ thống cập nhật thông tin bãi đỗ xe mới vào cơ sở dữ liệu của hệ thống và hiển thị thông báo xác nhận việc cập nhật thành công.</li> </ol>

Alternative Flow	5.1 Thông tin không hợp lệ -> Hệ thống yêu cầu Người quản trị điền lại thông tin trước khi tiếp tục.
------------------	------------------------------------------------------------------------------------------------------

### 3.1.5 Use case Xem danh sách chỗ đỗ xe

Name	Xem danh sách chỗ đỗ xe
Description	Cho phép Người quản trị xem danh sách các chỗ đỗ xe có sẵn trong một bãi đỗ xe cụ thể
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đã truy cập vào xem danh sách bãi đỗ xe
Postcondition	Người quản trị xem danh sách chỗ đỗ xe thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một bãi đỗ xe cụ thể sau đó chọn xem danh sách chỗ đỗ xe của bãi đỗ xe đó.</li> <li>2. Hệ thống hiển thị danh sách các chỗ đỗ xe trong bãi đỗ xe được chọn, bao gồm thông tin về số thứ tự chỗ đỗ, tình trạng (trống/đã đặt), vv.</li> </ol>
Alternative Flow	2.1 Nếu không có thông tin chi tiết về chỗ đỗ xe trong bãi đỗ đã được chọn, hệ thống sẽ hiển thị thông báo "Không có thông tin về chỗ đỗ xe trong bãi đỗ này".

### 3.1.6 Thêm chỗ đỗ xe mới

Name	Thêm chỗ đỗ xe mới
------	--------------------

Description	Cho phép Người quản trị thêm chỗ đỗ xe mới trong một bãi đỗ xe cụ thể
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đã truy cập vào xem danh sách chỗ đỗ xe
Postcondition	Người quản trị thêm thông tin của chỗ đỗ xe mới thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn “Thêm chỗ đỗ xe mới” trong danh sách chỗ đỗ xe</li> <li>2. Hệ thống hiển thị form để Người quản trị cập nhật các thông tin về chỗ đỗ xe mới</li> <li>3. Người quản trị điền các thông tin về bãi đỗ xe mới bao gồm: số thứ tự, vị trí,...Nhấn nút “Lưu” để áp dụng thay đổi</li> <li>4. Hệ thống kiểm tra thông tin chỗ đỗ xe Người quản trị vừa thêm có hợp lệ hay không.</li> <li>5. Thông tin hợp lệ, hệ thống thêm thông tin chỗ đỗ xe mới vào cơ sở dữ liệu của hệ thống và hiển thị thông báo xác nhận việc thêm thành công.</li> </ol>
Alternative Flow	5.1 Thông tin không hợp lệ -> Hệ thống yêu cầu Người quản trị điền lại thông tin trước khi tiếp tục.

### 3.1.7 Use case Xóa chỗ đỗ xe

Name	Xóa chỗ đỗ xe
------	---------------

Description	Cho phép Người quản trị xóa chỗ đỗ xe mới trong một bãi đỗ xe cụ thể
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đã truy cập vào xem danh sách chỗ đỗ xe
Postcondition	Người quản trị xóa thông tin của chỗ đỗ xe thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một chỗ đỗ xe cụ thể trong danh sách chỗ đỗ xe rồi chọn “Xóa chỗ đỗ xe”.</li> <li>2. Hệ thống yêu cầu xác nhận từ người quản trị trước khi thực hiện xóa chỗ đỗ xe</li> <li>3. Người quản trị xác nhận việc xóa chỗ đỗ xe</li> <li>4. Hệ thống thực hiện xóa chỗ đỗ xe và hiển thị thông báo xác nhận việc xóa thành công.</li> </ol>
Alternative Flow	2.1 Nếu chỗ đỗ xe đó đang được sử dụng, hệ thống hiển thị cảnh báo và yêu cầu xác nhận lại từ Người Quản trị trước khi tiếp tục xóa.

### 3.1.8 Use case Cập nhật thông tin chỗ đỗ xe

Name	Cập nhật thông tin chỗ đỗ xe
Description	Cho phép Người quản trị cập nhật thông tin của một chỗ đỗ xe cụ thể trong một bãi đỗ
Actor	Người quản trị

Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách chỗ đỗ xe
Postcondition	Người quản trị cập nhật thông tin của chỗ đỗ xe thành công
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một chỗ đỗ xe cụ thể trong danh sách chỗ đỗ xe rồi chọn “Cập nhật thông tin chỗ đỗ xe”</li> <li>2. Hệ thống hiển thị form để Người quản trị cập nhật các thông tin về chỗ đỗ xe đã chọn</li> <li>3. Người quản trị điền thông tin mới vào các trường cần cập nhật và nhấn nút “Lưu” để áp dụng thay đổi</li> <li>4. Hệ thống kiểm tra thông tin Người quản trị vừa cập nhật có hợp lệ hay không.</li> <li>5. Thông tin hợp lệ, hệ thống cập nhật thông tin chỗ đỗ xe mới vào cơ sở dữ liệu của hệ thống và hiển thị thông báo xác nhận việc cập nhật thành công.</li> </ol>
Alternative Flow	5.1 Thông tin không hợp lệ -> Hệ thống yêu cầu Người quản trị điền lại thông tin trước khi tiếp tục.

### 3.1.9. Use case Xem trạng thái bãi đỗ

Name	Xem trạng thái bãi đỗ
Description	Cho phép Người quản trị xem trạng thái hiện tại của một bãi đỗ xe trong ứng dụng.
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách các bãi đỗ xe.



Postcondition	Người quản trị xem được trạng thái hiện tại của bãi đỗ xe trên ứng dụng.
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một bãi đỗ xe cụ thể trong danh sách và chọn chức năng "Xem trạng thái".</li> <li>2. Hệ thống hiển thị trạng thái hiện tại của bãi đỗ xe như đang hoạt động, đóng cửa, số lượng chỗ đỗ trống, số lượng chỗ đã được đặt chỗ,...</li> </ol>
Alternative Flow	

### 3.1.10 Use case Cập nhật trạng thái bãi đỗ

Name	Xem trạng thái bãi đỗ
Description	Cho phép Người quản trị cập nhật trạng thái của một bãi đỗ xe trong ứng dụng.
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem trạng thái của một bãi đỗ xe cụ thể
Postcondition	Trạng thái của bãi đỗ xe đã được cập nhật thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn “Cập nhật trạng thái” cho bãi đỗ xe</li> <li>2. Hệ thống hiển thị giao diện cho phép Người quản trị cập nhật thông tin trạng thái của bãi đỗ xe như số lượng chỗ đỗ trống, số lượng chỗ đã được đặt chỗ,...</li> <li>3. Người quản trị cập nhật thông tin trạng thái theo yêu cầu</li> </ol>

	<p>4. Hệ thống kiểm tra thông tin Người quản trị vừa cập nhật có hợp lệ hay không.</p> <p>5. Thông tin hợp lệ, hệ thống cập nhật thông tin trạng thái bãi đỗ xe vào hệ thống và hiển thị thông báo xác nhận việc cập nhật thành công.</p>
Alternative Flow	5.1 Thông tin không hợp lệ -> Hệ thống yêu cầu Người quản trị điền lại thông tin trước khi tiếp tục.

### 3.2 Use case diagram Quản lý đơn đặt chỗ đỗ xe

#### 3.2.1 Xem danh sách đơn đặt chỗ đỗ xe

Name	Xem danh sách đơn đặt chỗ đỗ xe
Description	Cho phép Người quản trị xem danh sách đơn đặt chỗ đỗ xe có trên hệ thống
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào hệ thống
Postcondition	Người quản trị xem được danh sách đơn đặt chỗ đỗ xe có trên hệ thống
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn chức năng “Xem danh sách đơn đặt chỗ đỗ xe”</li> <li>2. Hệ thống hiển thị danh sách đơn đặt chỗ đỗ xe có trong hệ thống</li> <li>3. Người quản trị xem danh sách đơn đặt chỗ đỗ xe thành công</li> </ol>

Alternative Flow	2.1 Nếu không có đơn đặt chỗ đỗ xe nào trong hệ thống thì hiển thị thông báo "Không có đơn đặt chỗ đỗ xe nào".
------------------	----------------------------------------------------------------------------------------------------------------

### 3.2.2 Cập nhật trạng thái đơn đặt chỗ đỗ xe

Name	Cập nhật trạng thái đơn đặt chỗ đỗ xe
Description	Cho phép Người quản trị cập nhật trạng thái đơn đặt chỗ đỗ xe có trên hệ thống
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào hệ thống và đang xem danh sách đơn đặt chỗ đỗ xe
Postcondition	Người quản trị cập nhật trạng thái đơn đặt chỗ đỗ xe thành công
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một đơn đặt chỗ đỗ xe</li> <li>2. Người quản trị chọn chức năng “Cập nhật trạng thái đơn đặt chỗ đỗ xe”</li> <li>3. Hệ thống hiển thị thông tin đơn đặt chỗ và các tùy chọn trạng thái để Người quản trị lựa chọn</li> <li>4. Thông tin đơn đặt chỗ đỗ xe hợp lệ</li> <li>5. Người quản trị chọn trạng thái “Đã xác nhận” cho đơn đặt chỗ đỗ xe đã chọn</li> <li>6. Hệ thống cập nhật trạng thái “Đã xác nhận” cho đơn đặt chỗ đỗ xe.</li> </ol>
Alternative Flow	<p>4.1 Thông tin đơn đặt chỗ đỗ xe không hợp lệ -&gt; Người quản trị chọn trạng thái “Bị từ chối” cho đơn đặt chỗ đỗ xe đó.</p> <p>4.2 Thông tin đơn đặt chỗ cần thời gian để xác nhận thêm -&gt;</p>

	Người quản trị chọn trạng thái “Đang xử lý” cho đơn đặt chỗ đỗ xe đó.
--	-----------------------------------------------------------------------

### 3.2.3 Use case Xem chi tiết các đơn đặt chỗ đỗ xe

Name	Xem chi tiết các đơn đặt chỗ đỗ xe
Description	Cho phép Người quản trị xem thông tin chi tiết các đơn đặt chỗ đỗ xe
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách đơn đặt chỗ đỗ xe
Postcondition	Người quản trị xem được thông tin chi tiết các đơn đặt chỗ đỗ xe
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn một đơn đặt chỗ đỗ xe cụ thể để xem chi tiết</li> <li>2. Người quản trị chọn chức năng “Xem chi tiết đơn đặt chỗ”</li> <li>3. Hệ thống hiển thị thông tin chi tiết của đơn đặt chỗ Người quản lý đã chọn như số điện thoại người đặt, giờ đặt, đặt bãi nào...</li> </ol>
Alternative Flow	

### 3.2.4 Xem doanh số đặt chỗ đỗ xe

Name	Xem doanh số đặt chỗ đỗ xe
Description	Cho phép Người quản trị xem doanh số đặt chỗ đỗ xe của hệ

	thống
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào hệ thống
Postcondition	Người quản trị xem được doanh số đặt chỗ đỗ xe của hệ thống
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn chức năng “Xem doanh số đặt chỗ đỗ xe”</li> <li>2. Hệ thống hiển thị các tùy chọn để xem doanh số</li> <li>3. Người quản trị chọn các tùy chọn để xem doanh số như xem theo tuần, tháng, năm; xem theo loại xe, xem theo giờ đặt...</li> <li>4. Hệ thống lọc thông tin theo các tùy chọn của Người quản trị và hiển thị doanh số theo yêu cầu</li> </ol>
Alternative Flow	

### 3.2.5 Xem danh sách bãi đỗ xe yêu thích

Name	Xem danh sách bãi đỗ xe yêu thích
Description	Cho phép Người quản trị xem danh sách bãi đỗ xe yêu thích của các khách hàng trong hệ thống
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng
Postcondition	Người quản trị xem được danh sách bãi đỗ xe yêu thích của khách hàng

Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị chọn chức năng “Xem danh sách bãi đỗ xe yêu thích”</li> <li>2. Người quản trị chọn danh sách bãi đỗ xe yêu thích của một khách hàng.</li> <li>3. Hệ thống hiển thị danh sách bãi đỗ xe yêu thích của khách hàng đã chọn.</li> <li>4. Người quản trị xem danh sách bãi đỗ xe yêu thích để phân tích cho các chiến lược trong tương lai</li> </ol>
Alternative Flow	

### 3.3 Use case diagram Đặt chỗ đỗ xe

#### 3.3.1 Use case Xem danh sách bãi đỗ xe

Name	Xem danh sách bãi đỗ xe
Description	Cho phép Khách hàng hoặc Người quản trị xem danh sách các bãi đỗ xe có trên hệ thống
Actor	Người quản trị, Khách hàng
Precondition	Người quản trị hoặc Khách hàng đã đăng nhập vào hệ thống
Postcondition	Người quản trị hoặc Khách hàng xem được danh sách bãi đỗ xe có trên hệ thống
Main Flow	<ol style="list-style-type: none"> <li>4. Người quản trị hoặc Khách hàng chọn chức năng “Xem danh sách bãi đỗ xe”</li> <li>5. Hệ thống hiển thị danh sách các bãi đỗ xe có trong hệ thống</li> </ol>

	6. Người quản trị hoặc Khách hàng xem danh sách bãi đỗ xe thành công
Alternative Flow	2.1 Nếu không có bãi đỗ xe nào trong hệ thống thì hiển thị thông báo "Không có bãi đỗ xe nào".

### 3.3.2 Tìm kiếm bãi đỗ xe

Name	Tìm kiếm bãi đỗ xe
Description	Cho phép Khách hàng tìm kiếm bãi đỗ xe gần địa điểm mình muốn tới trên ứng dụng
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản và đăng nhập vào ứng dụng
Postcondition	Khách hàng đã tìm thấy bãi đỗ xe phù hợp với yêu cầu hoặc không tìm thấy kết quả phù hợp
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng nhập địa chỉ hoặc vị trí cụ thể mà mình muốn tìm kiếm bãi đỗ xe trên thanh tìm kiếm của ứng dụng</li> <li>2. Hệ thống hiển thị kết quả tìm kiếm với danh sách các bãi đỗ xe phù hợp với yêu cầu của Khách hàng</li> </ol>
Alternative Flow	2.1 Nếu không có bãi đỗ xe nào phù hợp hoặc không có kết quả trả về, hệ thống hiển thị thông báo "Không tìm thấy kết quả phù hợp".

### 3.3.3 Use case Xem chi tiết bãi đỗ xe

Name	Xem chi tiết bãi đỗ xe
------	------------------------

Description	Cho phép Khách hàng xem thông tin chi tiết của một bãi đỗ xe cụ thể.
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản và đăng nhập vào ứng dụng
Postcondition	Khách hàng xem được thông tin chi tiết của một bãi đỗ xe mong muốn.
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng chọn một bãi đỗ xe cụ thể để xem chi tiết</li> <li>2. Khách hàng chọn chức năng “Xem chi tiết bãi đỗ xe”</li> <li>3. Hệ thống hiển thị thông tin chi tiết của bãi đỗ xe, bao gồm tên, mô tả, địa điểm, trạng thái, giá cả...</li> <li>4. Khách hàng có thể thêm bãi đỗ xe đã chọn vào danh sách yêu thích của mình (nếu muốn)</li> <li>5. Hệ thống thêm bãi đỗ xe Khách hàng chọn vào danh sách yêu thích của Khách hàng (nếu có)</li> </ol>
Alternative Flow	

### 3.3.4 Use case Đánh giá

Name	Đánh giá
Description	Cho phép Khách hàng đánh giá bãi đỗ xe sau khi họ đã sử dụng dịch vụ đỗ xe tại đó.
Actor	Khách hàng
Precondition	Khách hàng đã đặt đơn đỗ xe và đã trải nghiệm dịch vụ tại bãi đỗ xe.



Postcondition	Khách hàng đã gửi đánh giá về bãi đỗ xe thành công trên ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng truy cập chức năng "Đánh giá bãi đỗ xe".</li> <li>2. Hệ thống hiển thị danh sách các đơn đặt đỗ xe đã được Khách hàng thực hiện.</li> <li>3. Người dùng chọn đơn đặt đỗ xe mà mình muốn đánh giá.</li> <li>4. Hệ thống hiển thị giao diện cho phép Khách hàng nhập đánh giá và phản hồi về trải nghiệm của mình tại bãi đỗ xe đó.</li> <li>5. Khách hàng nhập nội dung đánh giá của mình vào các trường tương ứng.</li> <li>6. Khách hàng nhấn nút "Gửi" để hoàn tất quá trình đánh giá</li> </ol>
Alternative Flow	

### 3.3.5 Use case Đặt chỗ đỗ xe

Name	Đặt chỗ đỗ xe
Description	Cho phép Khách hàng đặt chỗ đỗ xe trước qua ứng dụng
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản, đăng nhập vào ứng dụng và tìm

	kiểm được bãi đỗ xe phù hợp
Postcondition	Đơn đặt chỗ đỗ xe được tạo và Khách hàng đã đặt chỗ đỗ xe thành công, nhận được hóa đơn đặt chỗ đỗ xe gửi về Email
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng chọn bãi đỗ xe mình muốn đặt chỗ trước trên ứng dụng</li> <li>2. Khách hàng nhập các thông tin cần thiết: biển số, thời lượng đỗ xe, phương thức thanh toán</li> <li>3. Khách hàng xác nhận đặt chỗ và click “Đặt chỗ”</li> <li>4. Hệ thống chuyển hướng tới trang thanh toán phụ thuộc vào từng loại phương thức</li> <li>5. Khách hàng thanh toán hóa đơn</li> <li>6. Nếu thông tin thanh toán hợp lệ, hệ thống hiển thị hóa đơn đặt chỗ đỗ xe và gửi hóa đơn về Email Khách hàng</li> </ol>
Alternative Flow	<p>4.1 Nếu có lỗi xảy ra trong quá trình đặt chỗ như không thể tìm thấy chỗ trống, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu Khách hàng thử lại hoặc điều chỉnh thông tin đặt chỗ.</p> <p>4.2 Nếu thông tin đặt chỗ không hợp lệ hoặc không đầy đủ, hệ thống yêu cầu Khách hàng điều chỉnh thông tin trước khi tiếp tục.</p> <p>6.2 Thông tin thanh toán không hợp lệ -&gt; Hệ thống thông báo lỗi</p> <p>6.3 Giao dịch không thành công do lỗi kết nối, hết hạn -&gt; Hệ thống thông báo lỗi</p>

### 3.3.6 Use case Xem lịch sử đặt chỗ đỗ xe

Name	Xem lịch sử đặt chỗ đỗ xe
Description	Cho phép Khách hàng xem lại lịch sử các đơn đặt chỗ đỗ xe đã thực hiện trước đó trên ứng dụng.
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản, đăng nhập vào ứng dụng
Postcondition	Khách hàng đã xem được lịch sử đặt chỗ đỗ xe của tài khoản mình.
Main Flow	<ol style="list-style-type: none"><li>1. Khách hàng truy cập vào chức năng "Lịch sử đặt chỗ đỗ xe".</li><li>2. Hệ thống hiển thị danh sách các đơn đặt chỗ đỗ xe mà Khách hàng đã thực hiện trước đó</li><li>3. Người dùng chọn một đơn đặt chỗ đỗ xe cụ thể để xem chi tiết.</li></ol>
Alternative Flow	

### 3.4 Use case diagram Quản lý tài khoản

#### 3.4.1 Use case Đăng nhập

Name	Đăng nhập
Description	Cho phép Khách hàng hoặc Người quản trị đăng nhập vào tài khoản của mình khi bắt đầu sử dụng ứng dụng
Actor	Khách hàng, Người quản trị

Precondition	Khách hàng phải đăng ký tài khoản trước trên ứng dụng Người quản trị đã được cấp tài khoản sử dụng ứng dụng
Postcondition	Khách hàng hoặc Người quản trị đăng nhập thành công và có quyền truy cập ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng hoặc Người quản trị chọn chức năng “Đăng nhập” trên ứng dụng</li> <li>2. Hệ thống hiển thị các trường nhập thông tin gồm số điện thoại, mật khẩu</li> <li>3. Khách hàng hoặc Người quản trị nhập thông tin đăng nhập vào các trường thông tin tương ứng và click “Đăng nhập”</li> <li>4. Hệ thống kiểm tra tính hợp lệ của thông tin đăng nhập nhằm xác thực tài khoản</li> <li>5. Nếu thông tin đăng nhập là hợp lệ, Khách hàng hoặc Người quản trị đã đăng nhập thành công và được phép truy cập ứng dụng.</li> </ol>
Alternative Flow	5.1 Thông tin đăng nhập không hợp lệ -> Hệ thống báo lỗi tới người dùng (sai tên đăng nhập, sai mật khẩu, tài khoản không tồn tại,...)

### 3.4.2 Use case Đăng xuất

Name	Đăng xuất
Description	Cho phép Khách hàng hoặc Người quản trị đăng xuất khỏi tài khoản của mình trên ứng dụng

Actor	Khách hàng, Người quản trị
Precondition	Khách hàng hoặc Người quản trị đã có tài khoản và đăng nhập vào tài khoản trên ứng dụng
Postcondition	Khách hàng hoặc Người quản trị đăng xuất thành công khỏi ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng hoặc Người quản trị chọn chức năng “Đăng xuất” trên ứng dụng</li> <li>2. Hệ thống xác nhận yêu cầu đăng xuất</li> <li>3. Thoát khỏi giao diện hiện tại, đưa ra giao diện đăng nhập</li> </ol>
Alternative Flow	

### 3.4.3 Use case Đăng ký

Name	Đăng ký
Description	Cho phép Khách hàng đăng ký tài khoản mới khi bắt đầu sử dụng ứng dụng
Actor	Khách hàng
Precondition	Khách hàng chưa có tài khoản trên hệ thống
Postcondition	Khách hàng đăng ký thành công và có thể sử dụng tài khoản mới để truy cập vào ứng dụng
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng chọn chức năng “Đăng ký” trên ứng dụng</li> <li>2. Hệ thống hiển thị các trường thông tin cần cung cấp bao gồm tên, địa chỉ, số điện thoại, email và mật khẩu.</li> </ol>

	<ol style="list-style-type: none"> <li>Khách hàng nhập thông tin cá nhân vào các trường thông tin tương ứng và nhấn chọn “Đăng ký”</li> <li>Hệ thống kiểm tra tính hợp lệ của thông tin đăng ký</li> <li>Thông tin đăng ký hợp lệ, Khách hàng đăng ký thành công vào hệ thống tìm chỗ đỗ xe.</li> </ol>
Alternative Flow	5.1 Thông tin đăng ký không hợp lệ -> Hệ thống báo lỗi tới Khách hàng và yêu cầu nhập lại thông tin.

#### 3.4.4 Use case Quên mật khẩu

Name	Quên mật khẩu
Description	Cho phép Khách hàng lấy lại mật khẩu tài khoản khi quên
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản trên ứng dụng
Postcondition	Khách hàng tạo mật khẩu mới cho tài khoản thành công
Main Flow	<ol style="list-style-type: none"> <li>Khách hàng click “Quên mật khẩu” trên màn hình hiển thị của ứng dụng</li> <li>Hệ thống yêu cầu khách hàng nhập số điện thoại mà khách hàng đã đăng ký tài khoản trước đó</li> <li>Khách hàng nhập số điện thoại và click “Tạo lại mật khẩu”</li> <li>Hệ thống gửi mã xác thực về số điện thoại của Khách hàng</li> <li>Khách hàng điền mã xác thực đã nhận được trong SMS điện thoại vào ô xác nhận</li> </ol>

	<p>6. Nếu khách hàng nhập đúng mã xác thực thì hệ thống sẽ hiển thị ô thông tin “Mật khẩu mới” và “Nhập lại mật khẩu mới” để Khách hàng tạo mật khẩu mới</p> <p>7. Hệ thống xác nhận mật khẩu mới thành công</p>
Alternative Flow	<p>5.1 Khách hàng không nhận được mã xác thực -&gt; Yêu cầu hệ thống gửi lại mã</p> <p>6.1 Khách hàng nhập sai mã xác thực -&gt; Hệ thống thông báo mã sai và yêu cầu Khách hàng nhập lại</p>

### 3.4.5 Xem thông tin chi tiết tài khoản

Name	Xem thông tin chi tiết tài khoản
Description	Cho phép Khách hàng xem chi tiết thông tin cá nhân trong tài khoản của mình
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản và đăng nhập vào ứng dụng
Postcondition	Khách hàng đã xem được chi tiết thông tin cá nhân trong tài khoản
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng click “Tài khoản”</li> <li>2. Hệ thống hiển thị chi tiết thông tin cá nhân của Khách hàng, bao gồm Họ tên, Email, SĐT, Địa chỉ, Phương tiện,...</li> </ol>
Alternative Flow	

### 3.4.6 Use case **Chỉnh sửa thông tin**

Name	Chỉnh sửa thông tin
Description	Cho phép Khách hàng chỉnh sửa thông tin cá nhân trong tài khoản của mình
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản và đăng nhập vào ứng dụng
Postcondition	Khách hàng cập nhật thành công thông tin tài khoản
Main Flow	<ol style="list-style-type: none"><li>1. Khách hàng click vào tùy chọn “Tài khoản” và chọn nút “Chỉnh sửa thông tin tài khoản”</li><li>2. Hệ thống hiển thị form để Khách hàng chỉnh sửa các thông tin cá nhân của mình</li><li>3. Khách hàng điền thông tin mới vào các trường cần chỉnh sửa và nhấn nút “Lưu” để áp dụng thay đổi</li><li>4. Hệ thống kiểm tra thông tin Khách hàng vừa cập nhật có hợp lệ hay không.</li><li>5. Thông tin hợp lệ và hệ thống cập nhật thông tin khách hàng vừa thay đổi vào cơ sở dữ liệu của hệ thống.</li></ol>
Alternative Flow	<p>5.1 Thông tin không hợp lệ -&gt; Hệ thống yêu cầu Khách hàng nhập lại thông tin.</p>

### 3.4.7 Use case **Xem thông báo**

Name	Xem thông báo
------	---------------



Description	Cho phép Khách hàng xem các thông báo mới và thông báo đã đọc trên ứng dụng.
Actor	Khách hàng
Precondition	Khách hàng đã có tài khoản và đăng nhập vào ứng dụng
Postcondition	Khách hàng đã xem được các thông báo mới và cập nhật trên ứng dụng.
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng truy cập chức năng "Thông báo"</li> <li>2. Hệ thống hiển thị danh sách các thông báo mới và các thông báo đã đọc trước đó của tài khoản</li> <li>3. Khách hàng có thể chọn một thông báo cụ thể để xem chi tiết</li> </ol>
Alternative Flow	

#### 3.4.8 Xem danh sách thông tin tài khoản

Name	Xem danh sách thông tin tài khoản
Description	Cho phép Người quản trị có thể xem danh sách thông tin các tài khoản của Khách hàng
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng
Postcondition	Người quản trị đã xem được danh sách thông tin tài khoản của các khách hàng
Main Flow	<ol style="list-style-type: none"> <li>1. Người quản trị truy cập vào phần quản lý tài khoản khách</li> </ol>

	<p>hàng trong hệ thống.</p> <p>2. Người quản trị chọn chức năng ‘Xem danh sách thông tin tài khoản’</p> <p>3. Hệ thống hiển thị danh sách các tài khoản khách hàng, bao gồm các thông tin như số điện thoại, email, thông tin cá nhân và các thông tin liên quan khác.</p>
Alternative Flow	

### 3.4.9 Xem chi tiết thông tin tài khoản

Name	Xem chi tiết thông tin tài khoản
Description	Cho phép Người quản trị xem chi tiết thông tin tài khoản của các Khách hàng trong hệ thống
Actor	Người quản trị
Precondition	Người quản trị đã đăng nhập vào ứng dụng và đang xem danh sách thông tin tài khoản
Postcondition	Người quản trị xem chi tiết thông tin tài khoản của các Khách hàng thành công
Main Flow	<ol style="list-style-type: none"> <li>1. Khách hàng chọn một tài khoản để xem thông tin chi tiết</li> <li>2. Khách hàng chọn chức năng “Xem chi tiết thông tin tài khoản”</li> <li>3. Hệ thống hiển thị thông tin chi tiết tài khoản của Khách hàng đã chọn</li> </ol>
Alternative Flow	

## **CHƯƠNG 3: ACTIVITY DIAGRAM, CLASS DIAGRAM, SEQUENCE DIAGRAM**

### **1. Activity Diagram**

#### **1.1 Tổng quan về Activity Diagram**

Activity Diagram được sử dụng để minh họa luồng điều khiển trong hệ thống và đề cập các bước liên quan đến việc thực hiện Use case. Chúng ta có thể mô tả cả quá trình xử lý tuần tự và xử lý đồng thời các hoạt động bằng Activity Diagram, tức là Activity Diagram tập trung vào điều kiện của luồng hoạt động và trình tự xảy ra.

Lợi ích của Activity Diagram:

- Chứng minh tính logic của một thuật toán.
- Mô tả các bước thực hiện trong một Use case UML.
- Minh họa một quy trình kinh doanh hoặc quy trình làm việc giữa người dùng và hệ thống
- Đơn giản hóa và cải thiện mọi quy trình bằng cách làm rõ các trường hợp sử dụng phức tạp.
- Mô hình hóa các thành phần kiến trúc phần mềm, chẳng hạn như phương pháp, chức năng và hoạt động [4].

#### **1.2. Các thành phần cơ bản của Activity Diagram:**


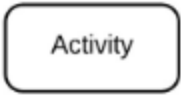

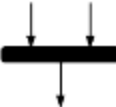

Một số thành phần phổ biến nhất của Activity Diagram bao gồm:




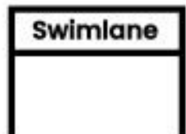
- Action: Một bước trong hoạt động trong đó người dùng hoặc phần mềm thực hiện một tác vụ nhất định. Các hành động được ký hiệu bằng hình chữ nhật có cạnh tròn.
- Decision node: Một nhánh có điều kiện trong luồng được biểu thị bằng hình

thời. Nó bao gồm một đầu vào duy nhất và hai hoặc nhiều đầu ra.

- Control flows: Các luồng điều khiển từ hoạt động này đến hoạt động khác
- Start node: Tượng trưng cho sự bắt đầu của hoạt động. Nút bắt đầu được thể hiện bằng một vòng tròn màu đen.
- End node: Thể hiện bước cuối cùng trong hoạt động. Nút cuối được thể hiện bằng một vòng tròn màu đen có viền ngoài [4].

Một số ký hiệu trong Activity Diagram:

Ký hiệu	Tên	Mô tả
	Start symbol	Thể hiện sự bắt đầu của một quy trình làm việc trong Activity Diagram.
	Activity symbol	Chỉ ra các hoạt động tạo nên một quy trình được mô hình hóa.
	Connector symbol	Hiện thị luồng định hướng hoặc luồng điều khiển của hoạt động.
	Joint symbol/ Synchronization bar	Kết hợp hai hoạt động đồng thời và tổng hợp chúng thành một luồng duy nhất.
	Fork symbol	Chia một luồng hoạt động thành hai hoạt động đồng thời.

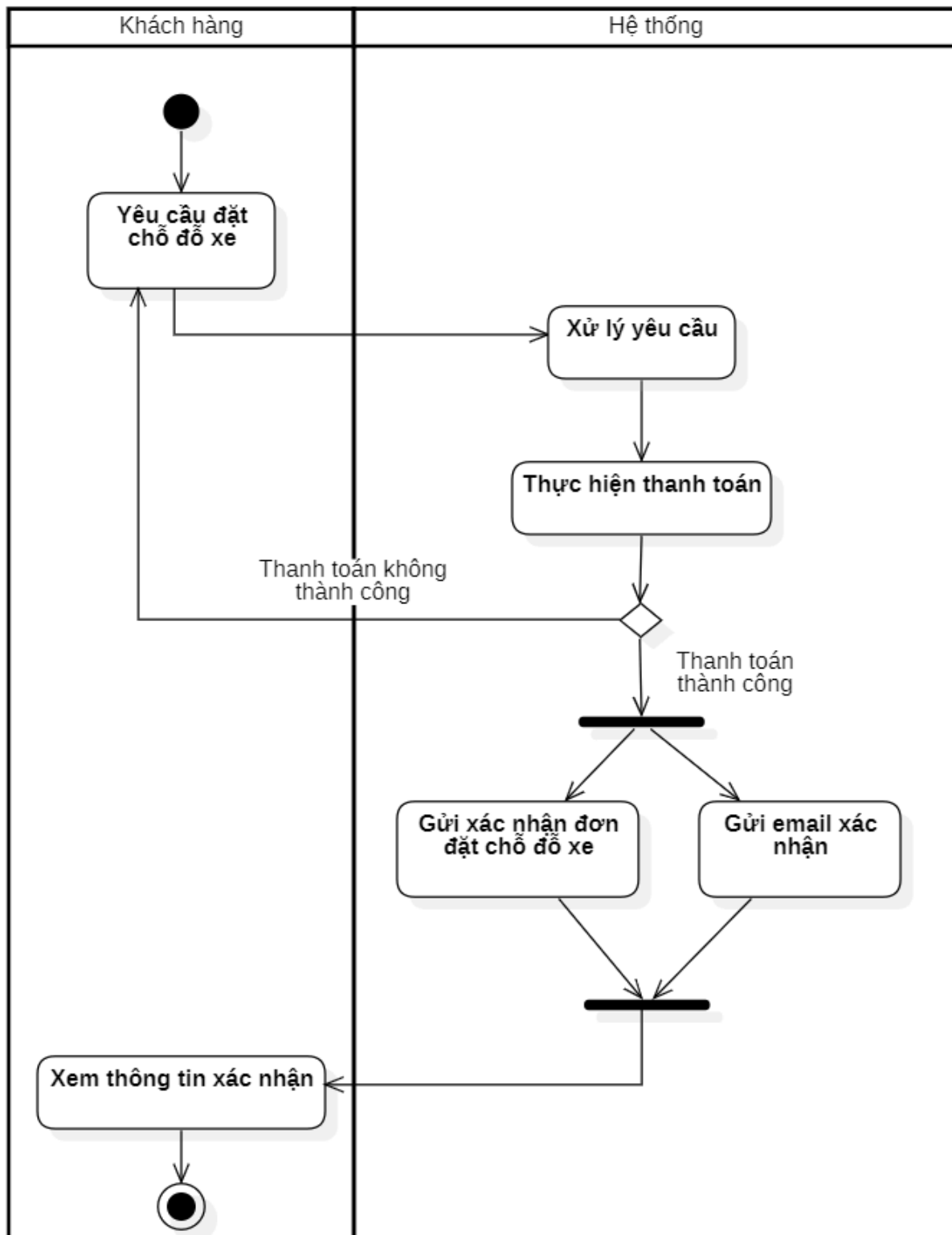
	Decision symbol	Thể hiện một quyết định và luôn có ít nhất hai đường dẫn phân nhánh với văn bản điều kiện để cho phép người dùng xem các tùy chọn.
	Note symbol	Ghi chú cho các hoạt động, điều kiện...
	End symbol	Đánh dấu trạng thái kết thúc của một hoạt động và thể hiện sự hoàn thành của tất cả các luồng của một quy trình.
	Swimlane	Nhóm các hoạt động của một đối tượng vào cùng một đường bơi của đối tượng đó.

### 1.3 Cách vẽ Activity Diagram

- Bước 1. Xác định trạng thái ban đầu và trạng thái cuối cùng:
  - Xác định nơi quá trình của bạn bắt đầu (trạng thái ban đầu) và nơi kết thúc (trạng thái cuối cùng). Ví dụ: nếu bạn đang lập mô hình quy trình pha một tách trà, trạng thái ban đầu có thể là “Chưa chuẩn bị trà” và trạng thái cuối cùng có thể là “Trà đã sẵn sàng”.
- Bước 2. Xác định các hoạt động trung gian cần thiết:
  - Xác định các bước hoặc hành động cần thiết để đi từ điểm bắt đầu đến điểm kết thúc.
  - Đây là những hoạt động hoặc nhiệm vụ cần được thực hiện.

- Bước 3. Xác định các điều kiện hoặc ràng buộc:
  - Xem xét các điều kiện hoặc hoàn cảnh có thể ảnh hưởng đến tiến trình hoạt động.
  - Đây là những yếu tố quyết định khi chuyển từ hoạt động này sang hoạt động khác.
- Bước 4. Vẽ sơ đồ với các ký hiệu phù hợp:
  - Thể hiện các trạng thái, hoạt động và điều kiện đã xác định một cách trực quan bằng cách sử dụng các ký hiệu và ký hiệu thích hợp trong sơ đồ hoạt động. Sơ đồ này đóng vai trò là bản đồ trực quan về quy trình của bạn, hiển thị dòng chảy từ trạng thái này sang trạng thái khác [5].

#### 1.4 Activity Diagram cho Use case Đặt chỗ đỗ xe



## **2. Class Diagram**

### **2.1 Tổng quan về Class Diagram**

Class Diagram là một biểu đồ lớp trong ngôn ngữ mô hình hóa thống nhất (UML). Nó là một loại biểu đồ cấu trúc tĩnh mô tả cấu trúc của một hệ thống bằng cách hiển thị các lớp của hệ thống, các thuộc tính của chúng, các thao tác (hoặc phương thức) và mối quan hệ giữa các đối tượng.

Mục đích của Class Diagram:

- Hiển thị cấu trúc tĩnh của các lớp trong hệ thống.
- Cung cấp một ký hiệu cơ bản cho các biểu đồ cấu trúc khác được quy định bởi UML.
- Hữu ích cho các nhà phát triển và các thành viên khác trong nhóm.
- Các nhà phân tích kinh doanh có thể sử dụng biểu đồ lớp để mô hình hóa hệ thống từ góc nhìn kinh doanh.

Một Class Diagram được tạo thành từ:

- Một tập hợp các lớp
- Một tập hợp các mối quan hệ giữa các lớp [6].

### **2.2 Các tính chất cơ bản của Class Diagram**

Một Class Diagram bao gồm ba phần:

- Tên lớp (Class name): Tên của lớp xuất hiện trong phân vùng đầu tiên.
- Thuộc tính của lớp (Attribute):
  - Các thuộc tính được hiển thị trong phân vùng thứ hai.
  - Loại thuộc tính được hiển thị sau dấu hai chấm.
  - Các thuộc tính tương ứng với biến thành viên (biến dữ liệu) trong mã.

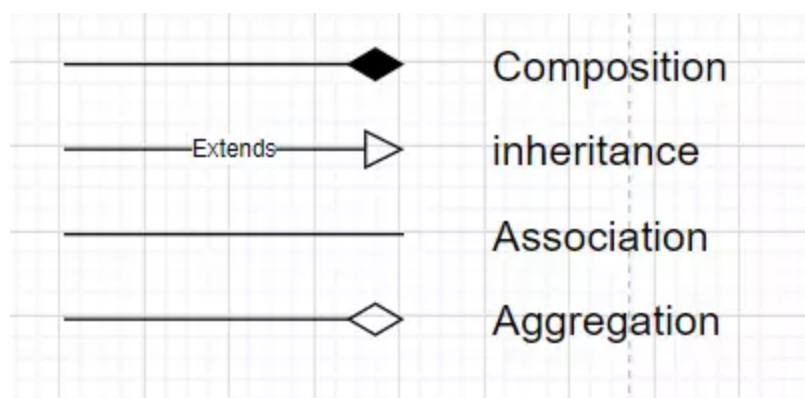


- Các phương thức của lớp (Operation):
  - Các phương thức được hiển thị trong phân vùng thứ ba. Chúng là các chức năng mà lớp cung cấp.
  - Loại trả về của một phương thức được hiển thị sau dấu hai chấm ở cuối của phương thức.
  - Loại trả về của các tham số phương thức được hiển thị sau dấu hai chấm theo sau tên tham số [6].

### 2.3 Mối quan hệ trong Class Diagram

Một lớp có thể tham gia vào một hoặc nhiều mối quan hệ với các lớp khác. Một mối quan hệ có thể là một trong các loại sau:

- Inheritance: 1 class kế thừa từ 1 class khác.
- Association: 2 class có liên hệ với nhau nhưng không chỉ rõ mối liên hệ.
- Composition: Đối tượng tạo từ class A mất thì đối tượng tạo từ class B sẽ mất.
- Aggregation: Đối tượng tạo từ class A mất thì đối tượng tạo từ class B vẫn tồn tại độc lập [7].



### 2.4 Access Modifier trong Class Diagram

Sử dụng để đặc tả phạm vi truy cập cho các Attribute và Operation của một class (Cấp

quyền cho các class khác sử dụng Attribute và Operation của class này).

Có 4 lựa chọn phạm vi truy cập:

- Private ( - ): Chỉ các đối tượng được tạo từ class này mới có thể sử dụng.
- Public ( + ): Mọi đối tượng đều có thể sử dụng.
- Protected ( # ): Chỉ các đối tượng được tạo từ class này và class kế thừa từ class này mới có thể sử dụng.
- Package/Default: Các đối tượng được tạo từ class trong lớp cùng gói có thể sử dụng [7].

## 2.5 Multiplicity trong Class Diagram

Sử dụng để thể hiện quan hệ về số lượng giữa các đối tượng được tạo từ các class trong Class Diagram:

- 0...1: 0 hoặc 1
- n : Bắt buộc có n
- 0...\* : 0 hoặc nhiều
- 1...\* : 1 hoặc nhiều
- m...n: có tối thiểu là m và tối đa là n[7].

## 2.6 Các bước xây dựng Class Diagram

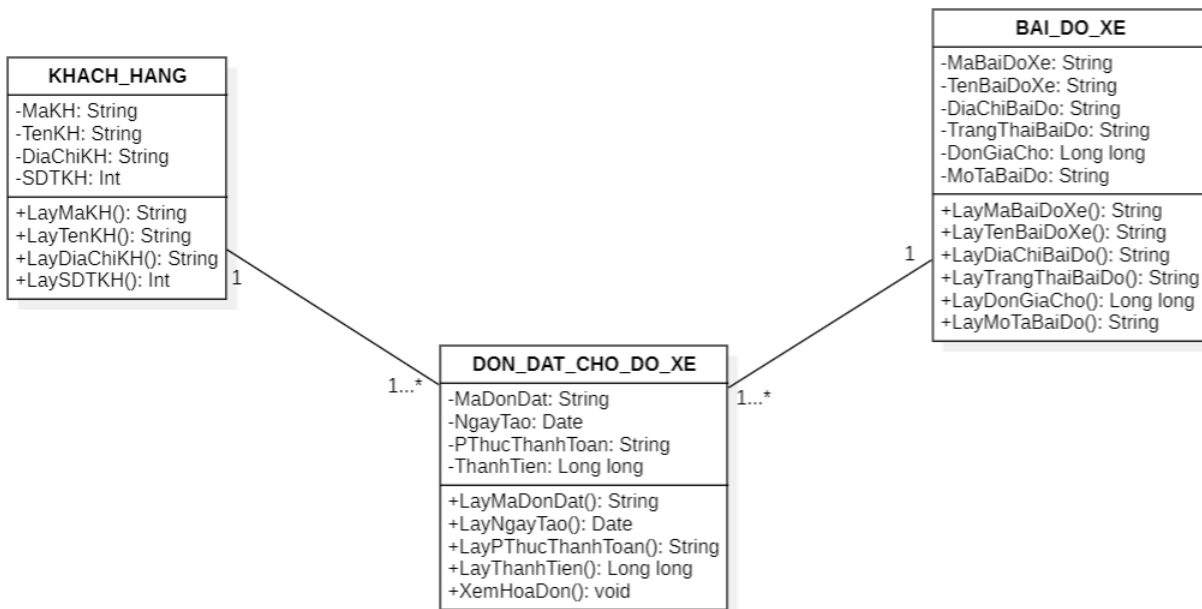
- Bước 1: Tìm các Classes dự kiến: Nghiên cứu kỹ các yêu cầu, Use Case và nghiên cứu kỹ các hệ thống tương tự để xác định các lớp dự kiến thông qua việc xác định các đối tượng có trong hệ thống.
- Bước 2: Xác định thuộc tính và quan hệ cho các lớp: Chúng ta bổ sung các thuộc tính cho các lớp và phân tích quan hệ của chúng.
- Bước 3: Bổ sung phương thức cho các lớp: Phương thức là các hành động mà đối

tượng sinh ra từ lớp đó có thể thực hiện trong hệ thống.

- Bước 4: Thiết kế chi tiết các thuộc tính và phương thức cho lớp: Khi đã có được Class Diagram, bạn cần thiết kế chi tiết các lớp bằng cách đặc tả các thuộc tính và phương thức của nó.
  - Đặc tả thuộc tính: chúng ta xác định kiểu dữ liệu và kích thước.
  - Đặc tả phương thức: chúng ta xác định dữ liệu đầu vào, dữ liệu đầu ra [8].

## 2.7 Class Diagram trong Ứng dụng đặt chỗ đỗ xe

Class Diagram cho Quy trình đặt chỗ đỗ xe



## 3. Sequence Diagram

### 3.1. Tổng quan về Sequence Diagram

Sequence Diagram là sơ đồ tương tác trình bày chi tiết cách thực hiện các hoạt động. Chúng nắm bắt sự tương tác giữa các đối tượng trong bối cảnh hợp tác. Sequence Diagram tập trung vào thời gian và chúng hiển thị thứ tự tương tác một cách trực quan bằng cách sử dụng trục tung của sơ đồ để biểu thị thời gian những tin nhắn nào được gửi

và khi nào.

Lợi ích của Sequence Diagram:

- Trình bày chi tiết các Use case UML.
- Lập mô hình logic của một quy trình, chức năng hoặc hoạt động phức tạp.
- Xem cách các đối tượng và thành phần tương tác với nhau để hoàn thành một quy trình.
- Lập kế hoạch và hiểu chức năng chi tiết của một kịch bản hiện tại hoặc tương lai [9].

Sequence Diagram hiển thị các phần tử khi chúng tương tác theo thời gian. Chúng được sắp xếp theo đối tượng (theo chiều ngang) và thời gian (theo chiều dọc):

- Chiều đối tượng: Trục ngang thể hiện các yếu tố tham gia vào quá trình tương tác. Thông thường, các đối tượng tham gia thao tác được liệt kê từ trái sang phải tùy theo thời điểm chúng tham gia vào chuỗi thông báo. Tuy nhiên, các phần tử trên trục ngang có thể xuất hiện theo bất kỳ thứ tự nào
- Chiều thời gian: Trục dọc thể hiện tiến trình thời gian (hoặc tiến triển) tính từ trên xuống dưới [10].

### **3.2. Các thành phần cơ bản của Sequence Diagram:**

- Actor
  - Thể hiện các thực thể tương tác với hệ thống và nằm bên ngoài hệ thống
  - Thể hiện vai trò của người dùng con người, phần cứng bên ngoài hoặc các chủ thể khác.
- Lifeline
  - Một lifeline đại diện cho một đối tượng tham gia vào tương tác.
  - Mỗi lifeline có một đường nét đứt dùng để đại diện cho sự trôi qua của thời

gian và hiển thị các sự kiện tuần tự xảy ra với một đối tượng trong quá trình lập biểu đồ.

- Activation box

- Một hình chữ nhật mỏng trên lifeline biểu thị khoảng thời gian mà một đối tượng đang thực hiện một thao tác.
- Phần trên và phần dưới của hình chữ nhật được căn chỉnh tương ứng với thời gian bắt đầu và hoàn thành

- Call message

- Một thông báo xác định một giao tiếp cụ thể giữa các Lifeline của một tương tác.
- Call message là loại tin nhắn thể hiện lời kêu gọi vận hành lifeline mục tiêu.

- Return message

- Một thông báo xác định một giao tiếp cụ thể giữa các Lifeline của một tương tác.
- Return message là một loại tin nhắn thể hiện việc chuyển thông tin trở lại người gọi của một tin nhắn tương ứng trước đó.

- Self message

- Một thông báo xác định một giao tiếp cụ thể giữa các Lifeline của một tương tác.
- Self message là một loại tin nhắn dùng để gọi thông điệp đến cùng một lifeline.

- Create message

- Một thông báo xác định một giao tiếp cụ thể giữa các Lifeline của một tương tác.

- Create message là một loại tin nhắn thể hiện sự khởi tạo của một lifeline mới.
- Destroy message
  - Một thông báo xác định một giao tiếp cụ thể giữa các Lifeline của một tương tác.
  - Destroy message là loại bản tin thể hiện yêu cầu hủy bỏ vòng đời của lifeline mục tiêu.
- Note
  - Một ghi chú (bình luận) cung cấp khả năng đính kèm các nhận xét khác nhau vào các thành phần. Một nhận xét không mang ý nghĩa ngữ nghĩa nhưng có thể chứa thông tin hữu ích cho người lập mô hình.
- Alternative
  - Tượng trưng cho sự lựa chọn (thường loại trừ lẫn nhau) giữa hai hoặc nhiều chuỗi tin nhắn.
  - Để thể hiện các lựa chọn thay thế, hãy sử dụng hình chữ nhật được gắn nhãn với đường đứt nét bên trong.
- Loop
  - Được sử dụng để lập mô hình các kịch bản nếu/thì, tức là một tình huống sẽ chỉ xảy ra trong những điều kiện nhất định [10].

### **3.3. Cách vẽ Sequence Diagram**

- Bước 1: Xác định một tập hợp các đối tượng sẽ tham gia vào quá trình cộng tác chung (hoặc kịch bản ca sử dụng)
  - Nếu bạn rút ra sơ đồ trình tự dựa trên một kịch bản của một ca sử dụng, trước tiên hãy chọn các kịch bản thông thường

- Bạn nên biết (các) tác nhân chính kích hoạt ca sử dụng
- Bước 2: Xem xét điểm đầu tiên của kịch bản (điểm đầu tiên của luồng sự kiện của trường hợp sử dụng)
- Bước 3: Xem xét hệ thống cần phải làm gì để phản hồi với actor, khi actor gửi tin nhắn đến hệ thống
  - Hệ thống cần xử lý những gì trước khi nhận được phản hồi tin nhắn trả về từ hệ thống?
  - Bằng cách này, bạn sẽ xác định được các đối tượng và các thông điệp được gửi cho kịch bản cụ thể đó và bạn cũng có thể sử dụng những thông tin này làm cơ sở để rút ra sơ đồ lớp theo từng bước.
- Bước 4: Lập lại từng điểm của tình huống (hoặc luồng sự kiện) cho đến khi bạn hoàn thành tất cả các điểm trong tình huống [11].

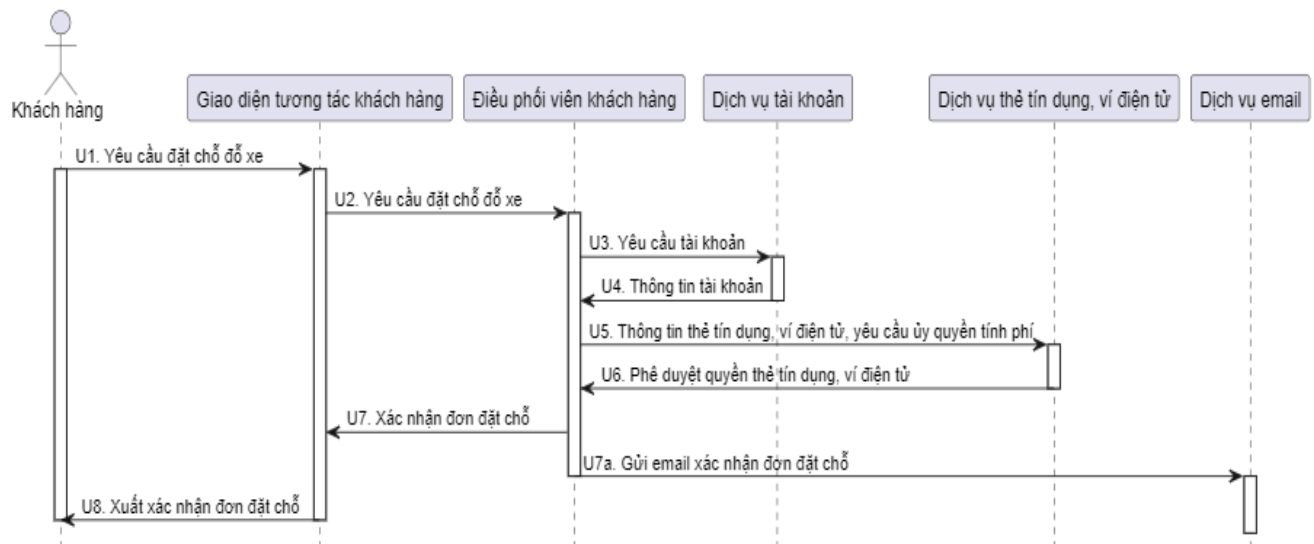
### 3.4 Sequence Diagram của Use case Đặt chỗ đỗ xe

Luồng thông điệp của Thao tác đặt chỗ đỗ xe:

- U1. Khách hàng gửi yêu cầu đặt chỗ đỗ xe đến giao diện tương tác khách hàng trên ứng dụng
- U2. Giao diện tương tác khách hàng gửi yêu cầu đặt chỗ đỗ xe đến điều phối viên khách hàng
- U3. Điều phối viên gửi yêu cầu tài khoản đến dịch vụ tài khoản
- U4: Dịch vụ tài khoản gửi thông tin tài khoản, bao gồm chi tiết thẻ tín dụng, ví điện tử của khách hàng đến điều phối viên khách hàng
- U5. Điều phối viên khách hàng gửi thông tin thẻ tín dụng, ví điện tử của khách hàng và yêu cầu ủy quyền tính phí đến dịch vụ thẻ tín dụng, ví điện tử
- U6. Dịch vụ thẻ tín dụng, ví điện tử gửi phê duyệt ủy quyền thẻ tín dụng, ví điện tử

tử đến điều phối viên khách hàng

- U7. Điều phối viên khách hàng gửi xác nhận đơn đặt chỗ đỗ xe đến giao diện tương tác khách hàng
- U7a. Điều phối viên khách hàng gửi email xác nhận đơn đặt chỗ đỗ xe đến khách hàng qua dịch vụ email
- U8. Giao diện tương tác khách hàng xuất xác nhận đơn đặt chỗ đỗ xe cho khách hàng





## CHƯƠNG 4: ƯU, NHƯỢC ĐIỂM, HƯỚNG PHÁT TRIỂN ỨNG DỤNG

### 1. Ưu điểm

- Đối với khách hàng:
  - Giúp tiết kiệm thời gian và công sức:
    - Tìm kiếm chỗ đỗ xe nhanh chóng và dễ dàng, không cần phải đi đến từng nơi một để tìm.
    - Đặt chỗ trước giúp đảm bảo có chỗ đỗ xe, đặc biệt là trong giờ cao điểm hoặc tại các khu vực đông dân cư.
    - Thanh toán trực tuyến tiện lợi, không cần phải mang theo tiền mặt.
    - Nhận thông báo nhắc nhở về thời gian đỗ xe sắp hết, giúp tránh trường hợp bị phạt.
  - Lựa chọn thông minh:
    - Xem được thông tin chi tiết của từng bãi đỗ xe như vị trí, giá cả, dịch vụ đi kèm, hình ảnh, đánh giá của người dùng khác.
    - So sánh các lựa chọn khác nhau để tìm được chỗ đỗ xe phù hợp nhất với nhu cầu và ngân sách của mình.
  - Trải nghiệm tốt hơn:
    - Gửi bình luận, đánh giá về dịch vụ của bãi đỗ xe, giúp góp phần nâng cao chất lượng dịch vụ.
    - Xem lịch sử đỗ xe của mình để theo dõi chi tiêu và quản lý việc sử dụng dịch vụ.
    - Quản lý tài khoản cá nhân, thay đổi thông tin, mật khẩu dễ dàng.
- Đối với người quản trị:
  - Nâng cao hiệu quả quản lý:
    - Quản lý tập trung các bãi đỗ xe, tài khoản khách hàng, đơn đặt chỗ, thanh toán.

- Theo dõi doanh thu, số lượt đỗ xe, đánh giá của khách hàng theo thời gian.
- Báo cáo thống kê trực quan, giúp đánh giá hiệu quả hoạt động và đưa ra quyết định quản lý phù hợp.
- Cải thiện chất lượng dịch vụ:
  - Phân tích nhu cầu của khách hàng thông qua các dữ liệu thu thập được.
  - Đưa ra các chiến lược phát triển dịch vụ phù hợp, đáp ứng nhu cầu thị trường.
  - Nâng cao chất lượng dịch vụ, thu hút khách hàng và tăng doanh thu.

## **2. Nhược điểm**

Mặc dù ứng dụng đặt chỗ đỗ xe mang lại nhiều tiện ích và lợi ích cho khách hàng nhưng vẫn tồn tại một số nhược điểm cần được cân nhắc và cải thiện:

- Vấn đề về tính tương thích: Ứng dụng có thể không tương thích hoặc hoạt động không mượt mà trên một số loại thiết bị di động và hệ điều hành, gây ra sự phiền toái cho khách hàng khi trải nghiệm.
- Hạn chế về sự đa dạng của bãi đỗ xe: Ứng dụng hiện chỉ mới cập nhật thông tin các bãi đỗ xe ở trong các thành phố lớn như Hồ Chí Minh, Đà Nẵng,...gây ra sự thiếu lựa chọn cho người dùng.
- Vấn đề về sự chính xác của thông tin: Thông tin về giá cả, sức chứa hoặc thời gian hoạt động của một số bãi đỗ xe có thể có sai sót do không được cập nhật.

## **3. Phương hướng phát triển**

- Đối với vấn đề về tính tương thích:
  - Phát triển ứng dụng đa nền tảng: Hỗ trợ ứng dụng trên nhiều hệ điều hành phổ biến như Android, iOS, Windows Phone.

- Thiết kế giao diện ứng dụng linh hoạt: Tương thích với nhiều kích thước màn hình và độ phân giải khác nhau.
- Tiến hành kiểm tra ứng dụng kỹ lưỡng trên nhiều thiết bị khác nhau: Đảm bảo ứng dụng hoạt động mượt mà và ổn định trên tất cả các thiết bị.
- Đối với hạn chế về sự đa dạng của bãi đỗ xe:
  - Mở rộng hợp tác với các chủ bãi đỗ xe: Khuyến khích các chủ bãi đỗ xe ở nhiều khu vực khác nhau tham gia vào hệ thống.
  - Cung cấp các ưu đãi hấp dẫn cho các chủ bãi đỗ xe: Thu hút các chủ bãi đỗ xe tham gia vào hệ thống, đặc biệt là các bãi đỗ xe ở khu vực ngoại thành.
  - Tăng cường quảng bá ứng dụng: Giới thiệu ứng dụng đến người dùng ở nhiều khu vực khác nhau.
- Đối với vấn đề về sự chính xác của thông tin:
  - Cập nhật thông tin thường xuyên: Thu thập thông tin mới từ các chủ bãi đỗ xe và cập nhật lên hệ thống.
  - Cho phép người dùng phản hồi về thông tin: Tạo kênh thông tin để người dùng phản hồi về những sai sót trong thông tin của các bãi đỗ xe.
  - Xác minh thông tin: Hợp tác với các bên thứ ba uy tín để xác minh thông tin của các bãi đỗ xe.
- Một số hướng phát triển khác:
  - Nâng cao tính bảo mật của hệ thống: Đảm bảo an toàn thông tin cá nhân và thanh toán của người dùng.
  - Cung cấp dịch vụ khách hàng tốt: Hỗ trợ người dùng giải quyết các vấn đề gặp phải khi sử dụng ứng dụng.
  - Thu thập phản hồi của người dùng: Lắng nghe ý kiến của người dùng để cải thiện ứng dụng ngày càng tốt hơn.

## **CHƯƠNG 5: KẾT LUẬN**

Trong đồ án này, chúng em đã mô tả dự án và thiết kế các Use Case Diagram để mô hình hóa quy trình hoạt động của Ứng dụng đặt chỗ đỗ xe. Dự án này nhằm mục đích tạo ra một bản mô tả tổng quan về các tác nhân, chức năng và tương tác trong hệ thống. Qua đó giúp hiểu rõ hơn về quy trình hoạt động của hệ thống và làm cơ sở cho việc phát triển, triển khai dự án.

Trong quá trình thực hiện đồ án, chúng em đã nhận thấy sự quan trọng của việc xây dựng Use Case Diagram để phân tích và thiết kế một phần mềm trong thực tế. Chúng em hy vọng rằng kinh nghiệm từ việc thực hiện dự án này sẽ giúp chúng em phát triển kỹ năng nghiên cứu và phát triển phần mềm một cách chuyên nghiệp hơn trong tương lai.

## TÀI LIỆU THAM KHẢO

- [1] “What is Use Case Diagram?” Accessed: Apr. 06, 2024. [Online]. Available: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>
- [2] “Use Case là gì? Cách xây dựng sơ đồ Use Case hiệu quả.” Accessed: Apr. 06, 2024. [Online]. Available: <https://stringee.com/vi/blog/post/use-case-diagram-la-gi>
- [3] “Use Case Diagram: Definition and Examples,” Indeed Career Guide. Accessed: Apr. 06, 2024. [Online]. Available: <https://www.indeed.com/career-advice/career-development/use-case-diagram>
- [4] “UML Activity Diagram Tutorial,” Lucidchart. Accessed: Apr. 07, 2024. [Online]. Available: <https://www.lucidchart.com/pages/uml-activity-diagram>
- [5] “Activity Diagrams | Unified Modeling Language (UML),” GeeksforGeeks. Accessed: Apr. 07, 2024. [Online]. Available: <https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/>
- [6] “Sơ đồ lớp là gì?” Accessed: Apr. 07, 2024. [Online]. Available: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/>
- [7] “Tìm Hiểu Về Cách Thiết Kế Class Diagram - Viblo.” Accessed: Apr. 07, 2024. [Online]. Available: <https://viblo.asia/p/tim-hieu-ve-cach-thiet-ke-class-diagram-L4x5xLyY5BM>
- [8] iViettech Education, “Thực hành xây dựng Class Diagram,” iViettech. Accessed: Apr. 07, 2024. [Online]. Available: <https://iviettech.vn/viettech/916-thuc-hanh-xay-dung-classs-diagarm.html>
- [9] “UML Sequence Diagram Tutorial,” Lucidchart. Accessed: Apr. 07, 2024. [Online]. Available: <https://www.lucidchart.com/pages/uml-sequence-diagram>
- [10] “What is Sequence Diagram?” Accessed: Apr. 07, 2024. [Online]. Available: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/>
- [11] “Hướng dẫn về sơ đồ trình tự.” Accessed: Apr. 07, 2024. [Online]. Available: <https://online.visual-paradigm.com/diagrams/tutorials/sequence-diagram-tutorial/>