

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN

=====***=====



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài:

XÂY DỰNG HỆ THỐNG GIÁM SÁT
BÃI GỬI XE QUA CAMERA

Sinh viên thực hiện : BÙI THU HẰNG

Mã sinh viên: 1041060287

Lớp: Công nghệ thông tin 3

Khóa: 10

Giảng viên hướng dẫn : TS. ĐỖ VĂN TUẤN

Hà Nội 2019

LỜI NÓI ĐẦU

Với sự gia nhập WTO vào ngày 7 tháng 11 năm 2006, Việt Nam chính thức bước vào sân chơi thương mại toàn cầu. Từ bước nhảy vượt bậc đó, đời sống vật chất của con người nâng cao đã đến. Theo thống kê của cục đăng kiểm Việt Nam tính đến tháng 1 năm 2019 nước ta có trên 3.309.469 ô tô, ước tính mỗi năm tăng từ 8-10%. Tỷ lệ nghịch với sự gia tăng phương tiện giao thông chính là cơ sở hạ tầng phục vụ cho các phương tiện giao thông. Thêm vào đó sự mọc lên hàng ngàn chung cư, các cao ốc hay các trung tâm thương mại tại các thành phố lớn như thành phố Hồ Chí Minh, Hà Nội, Đà Nẵng đòi hỏi một lượng lớn các bãi giữ xe và hiệu quả làm việc của các bãi giữ xe luôn ở mức cao.

Chính vì nhu cầu cấp thiết về bãi giữ xe về cả số lượng lẫn chất lượng, cho nên với mục đích cải thiện và giải quyết vấn đề mà cộng đồng xã hội đã và đang quan tâm, đề tài **“Xây dựng hệ thống giám sát bãi giữ xe bằng Camera”** đưa ra hướng giải quyết tốt nhất tại thời điểm này. Đề tài này giúp cho việc quản lý thống kê và kiểm soát bãi giữ xe một cách dễ dàng hơn, ứng dụng những kỹ thuật tiên tiến và giúp cho bãi giữ xe hoạt động tối đa năng suất và đạt hiệu quả cao.

Trong quá trình thực hiện đồ án, với sự giúp đỡ của TS. Đỗ Văn Tuấn, em đã hoàn thành đồ án trong thời gian cho phép. Em xin chân thành cảm ơn thầy đã luôn giúp đỡ, đóng góp, chỉ bảo tận tình để em hoàn thành đồ án thật tốt. Bên cạnh đó, em cũng xin cảm ơn các thầy (cô) giáo trong khoa vì đã truyền đạt cho em những kiến thức nền tảng để em hoàn thành được đồ án.

Tuy nhiên vì kiến thức chuyên môn còn hạn hẹp không thể tránh được những thiếu sót. Em mong các thầy cô trong khoa công nghệ thông tin có thể đóng góp thêm ý kiến để em có thể hoàn thiện đồ án của mình hơn.

Em xin chân thành cảm ơn!

Hà Nội, ngày 09 tháng 05 năm 2019

Sinh viên

Bùi Thu Hằng

MỤC LỤC

LỜI NÓI ĐẦU	2
Danh mục hình ảnh.....	5
Danh sách bảng	7
Các từ viết tắt sử dụng trong đồ án.....	8
PHẦN I: MỞ ĐẦU	9
1.1 Giới thiệu vấn đề:	9
1.2 Mục tiêu đề tài:.....	9
PHẦN II: NỘI DUNG.....	10
Chương 1. Tìm hiểu về GUI Framework WPF	10
1.1 Tổng quan về WPF.....	10
1.2 Nền tảng thống nhất để xây dựng giao diện người dùng	10
1.3 Khả năng làm việc chung giữa người thiết kế giao diện và lập trình viên.....	13
1.4 Hệ quản trị cơ sở dữ liệu	15
1.4.1 Giới thiệu về SQL Server 2017:	15
1.4.2 SQL là ngôn ngữ của cơ sở dữ liệu quan hệ:	16
1.4.3 Vai trò của SQL	17
Chương 2. Quy trình xử lý nhận dạng biển số	19
2.1 Giới thiệu thư viện EmguCV	19
2.2 Quy trình xử lý nhận dạng biển số xe	19
Chương 3. Tìm hiểu đề tài và phân tích thiết kế hệ thống	21
3.1 Khảo sát hệ thống.....	21
3.1.1 Khảo sát sơ bộ.....	21
3.1.2 Khảo sát chi tiết	22
3.2 Phân tích thiết kế hệ thống	22
3.2.1 Xây dựng biểu đồ Use case.....	22
3.2.2 Biểu đồ Use case tổng thể.....	24
3.2.3 Biểu đồ use case phân rã.....	24
3.2.4 Đặc tả ca sử dụng.....	29
3.2.5 Biểu đồ trình tự	39
3.4 Thiết kế cơ sở dữ liệu.....	53
3.4.1 Xác định các thực thể.....	53

3.4.2	Mô hình quan hệ	57
3.5	Thiết kế giao diện.....	58
3.5.1	Giao diện đăng nhập	58
3.5.2	Giao diện màn hình chính.....	58
3.5.3	Giao diện quản lý xe vào	59
3.5.4	Giao diện quản lý xe ra	59
3.5.5	Giao diện quản lý xe trong bãi.....	60
3.5.6	Giao diện quản lý loại xe	61
3.5.7	Giao diện quản lý thẻ gửi xe	61
3.5.8	Giao diện quản lý xe gửi theo tháng	62
3.5.9	Giao diện quản lý nhân viên	62
3.5.10	Giao diện thống kê doanh thu	63
KẾT LUẬN		64
Tài liệu tham khảo:		65

Danh mục hình ảnh

Hình 2.1. 1 Giao diện quản nhà hàng bằng WPF.....	12
Hình 2.1. 2 Quá trình làm việc giữa lập trình viên và người thiết kế giao diện khi sử dụng XAML	14
Hình 2.2. 1 Mô hình tổng quan bài toán nhận dạng biển số xe	20
Hình 2.3. 1 Sơ đồ quy trình gửi xe tại bãi gửi xe thông minh	21
Hình 2.3. 2 Biểu đồ các use case tổng thể.....	24
Hình 2.3. 3 Biểu đồ phân rã đăng nhập.....	24
Hình 2.3. 4 Biểu đồ phân rã quản lý tài khoản	25
Hình 2.3. 5 Biểu đồ phân rã quản lý xe trong bãi	25
Hình 2.3. 6 Biểu đồ phân rã quản lý xe tháng.....	26
Hình 2.3. 7 Biểu đồ phân rã quản lý loại xe	26
Hình 2.3. 8 Biểu đồ phân rã quản lý thẻ gửi xe	27
Hình 2.3. 9 Biểu đồ phân rã quản lý xe vào.....	27
Hình 2.3. 10 Biểu đồ phân rã quản lý xe ra	28
Hình 2.3. 11 Biểu đồ phân rã thống kê doanh thu	28
Hình 2.3. 12 Biểu đồ trình tự đăng nhập.....	39
Hình 2.3. 13 Biểu đồ trình tự thống kê doanh thu doanh thu	39
Hình 2.3. 14 Biểu đồ trình tự sửa loại xe	40
Hình 2.3. 15 Biểu đồ trình tự thêm loại phương tiện.....	41
Hình 2.3. 16 Biểu đồ trình tự xóa loại phương tiện	42
Hình 2.3. 17 Biểu đồ trình tự sửa tài khoản.....	43
Hình 2.3. 18 Biểu đồ trình tự thêm tài khoản	44
Hình 2.3. 19 Biểu đồ trình tự xóa tài khoản.....	45
Hình 2.3. 20 Biểu đồ trình tự sửa vé xe	46
Hình 2.3. 21 Biểu đồ trình tự thêm vé xe.....	46
Hình 2.3. 22 Biểu đồ trình tự xóa vé xe	47
Hình 2.3. 23 Biểu đồ trình tự thêm xe vào bãi.....	48
Hình 2.3. 24 Biểu đồ xe ra khỏi bãi	49

Hình 2.3. 25 Biểu đồ thêm xe gửi tháng	50
Hình 2.3. 26 Biểu đồ sửa xe gửi tháng.....	51
Hình 2.3. 27 Biểu đồ xóa xe gửi tháng	52
Hình 2.3. 28 Biểu đồ mô hình cơ sở dữ liệu vật lý	57
Hình 2.3. 29 Giao diện chức năng đăng nhập.....	58
Hình 2.3. 30 Giao diện màn hình chính	58
Hình 2.3. 31 Giao diện chức năng quản lý xe vào.....	59
Hình 2.3. 32 Giao diện chức năng quản lý xe ra.....	59
Hình 2.3. 33 Giao diện chức năng thông tin xe trong bãi.....	60
Hình 2.3. 34 Giao diện quản lý loại xe	61
Hình 2.3. 35 Quản lý thẻ gửi xe	61
Hình 2.3. 36 Quản lý xe tháng	62
Hình 2.3. 37 Giao diện quản lý nhân viên	62
Hình 2.3. 38 Giao diện thống kê doanh thu	63

Danh sách bảng

Bảng 2.1. 1 Bảng so sánh giữa các nền tảng.....	11
Bảng 2.3. 1 Các tác nhân trong hệ thống	22
Bảng 2.3. 2 Các use case trong hệ thống	23
Bảng 2.3. 3 Bảng nhân viên	53
Bảng 2.3. 4 Bảng hóa đơn.....	53
Bảng 2.3. 5 Bảng nhật ký xe ra.....	54
Bảng 2.3. 6 Bảng nhật ký xe vào	54
Bảng 2.3. 7 Bảng loại xe	55
Bảng 2.3. 8 Bảng thẻ gửi xe.....	55
Bảng 2.3. 9 Bảng xe trong bãi.....	55
Bảng 2.3. 10 Bảng xe tháng	56
Bảng 2.3. 11 Bảng phân quyền	56

Các từ viết tắt sử dụng trong đồ án

STT	Ký hiệu	Ý nghĩa
1	WPF	Windows Presentation Foundation
2	API	Application Programming Interface – Giao diện lập trình ứng dụng
3	XAML	eXtensible Application Markup Language
4	GDI+	Là giao diện lập trình (API) của Window
5	SQL	Structured Query Language
6	RDBMS	Relational Database Management System
7	UC	Use case

PHẦN I: MỞ ĐẦU

1.1 Giới thiệu vấn đề:

Hệ thống giám sát bãi gửi xe bằng camera là một phần mềm cung cấp cho nhân viên trông giữ xe, quản lý những chức năng cần thiết, để tiến hành quản lý hoạt động của bãi gửi xe.

Đối với nhân viên trông giữ xe hệ thống cho phép tìm kiếm thông tin các xe gửi trong bãi, nhập hay đối chiếu thông tin xe với thẻ gửi xe khi xe vào bãi gửi hay khi lấy xe ra khỏi bãi gửi xe. Nhân viên trông xe có thể thống kê lượng xe hiện tại đang trong bãi để tiếp tục nhận xe vào gửi.

Đối người quản lý, sẽ thống kê doanh thu hàng tháng, quản lý người dùng hệ thống, quản lý bảng giá cho các loại phương tiện, quản lý các loại phương tiện, quản lý thẻ gửi xe.

1.2 Mục tiêu đề tài:

- Quản lý người dùng, phân quyền người dùng.
 - Cập nhật các thông tin về người dùng.
 - Cập nhật thông tin về các phương tiện.
 - Thống kê, cập nhật doanh thu.
 - Tìm kiếm thông tin gửi xe thông qua biển số trên phần mềm.
 - Quản lý thông tin xe vào, xe ra.
 - Quản lý thông tin thẻ.
-

PHẦN II: NỘI DUNG

Chương 1. Tìm hiểu về GUI Framework WPF

1.1 Tổng quan về WPF

WPF viết tắt của Windows Presentation Foundation, là hệ thống API mới hỗ trợ việc xây dựng giao diện đồ họa trên nền Windows. WPF được xem như thế hệ kế tiếp của WinForms, WPF tăng cường khả năng lập trình giao diện của lập trình viên bằng cách cung cấp các API cho phép tận dụng những lợi thế về đa phương tiện hiện đại. Là một bộ phận của .NET Framework 3.0, WPF sẵn có trong Windows Vista và Windows Server 2008. Đồng thời WPF cũng có thể hoạt động trên nền Windows XP Service Pack 2 hoặc mới hơn và cả Windows Server 2003.

WPF được xây dựng nhằm vào ba mục tiêu cơ bản:

- Cung cấp một nền tảng thống nhất để xây dựng giao diện người dùng.
- Cho phép người lập trình và người thiết kế giao diện làm việc cùng nhau một cách dễ dàng.
- Cung cấp một công nghệ chung để xây dựng giao diện người dùng trên cả Windows và trình duyệt Web.

1.2 Nền tảng thống nhất để xây dựng giao diện người dùng

Trước khi WPF ra đời, việc tạo giao diện người dùng theo những yêu cầu mô tả ở ví dụ trên đòi hỏi sử dụng rất nhiều công nghệ khác nhau. Để tạo form, các control và các tính năng kinh điển khác của một giao diện đồ họa Windows, thông thường lập trình viên sẽ chọn Windows Forms, một phần của .NET Framework. Nếu cần hiển thị văn bản, Windows Forms có một số tính năng hỗ trợ văn bản trực tiếp hoặc có thể sử dụng Adobe's PDF để hiển thị văn bản có khuôn dạng cố định.

Đối với hình ảnh và đồ họa 2 chiều, lập trình viên sẽ dùng GDI+, một mô hình lập trình riêng biệt có thể truy nhập qua Windows Forms. Để hiển thị video hay phát âm thanh, lập trình viên phải sử dụng Windows Media Player, và với đồ họa 3 chiều, lập trình viên phải dùng Direct3D, một thành phần

chuẩn khác của Windows. Chính vì vậy quá trình phát triển giao diện người dùng theo yêu cầu trở nên phức tạp, đòi hỏi lập trình viên quá nhiều kỹ năng công nghệ.

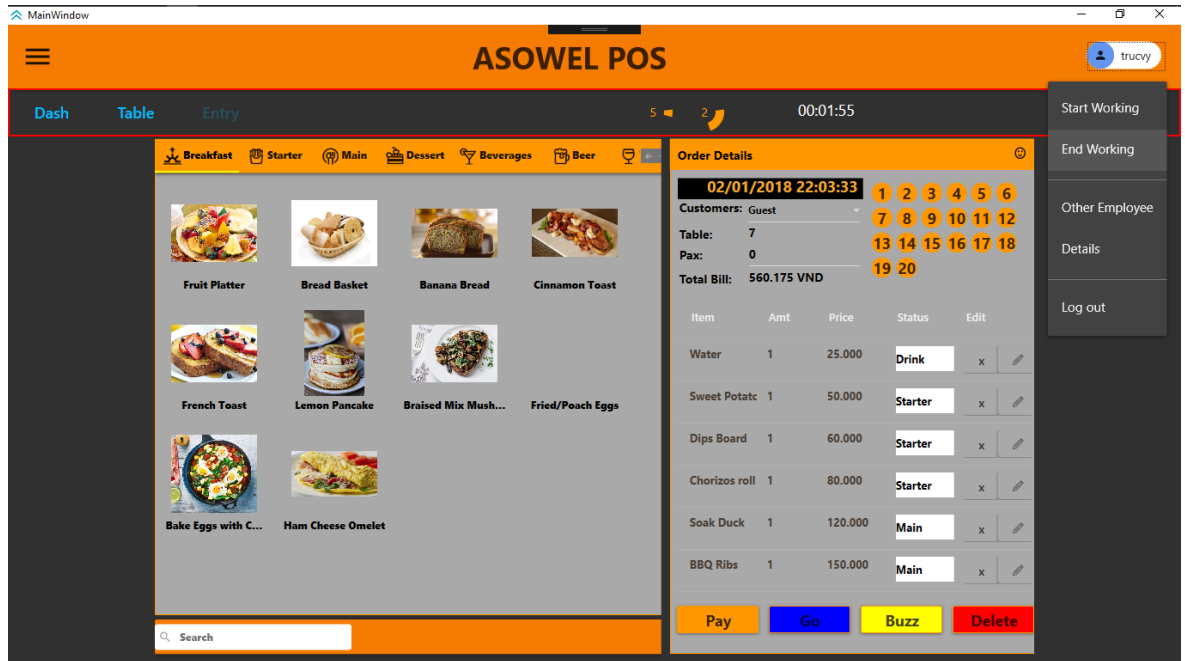
Sự nổi bật của WPF so với các nền tảng khác được thể hiện qua **bảng 2.1.1** dưới đây.

	Windows Forms	PDF	Windows Forms/ GDI+	Windows Media Player	Direct3D	WPF
Giao diện đồ họa	X					X
On-screen văn bản	X					X
Fixed-format văn bản		X				X
Hình ảnh			X			X
Video và âm thanh				X		X
Đồ họa 2 chiều			X			X
Đồ họa 3 chiều					X	X

Bảng 2.1. 1 Bảng so sánh giữa các nền tảng

WPF là giải pháp phù hợp nhất nhằm giải quyết tất cả những vấn đề công nghệ nêu trên hay nói cách khác, WPF cung cấp nhiều tính năng lập trình giao diện trong cùng một công nghệ đơn nhất. Điều này giúp cho quá trình tạo giao diện người dùng trở nên dễ dàng hơn đáng kể.

Hình bên dưới cho thấy một giao diện quản lý nhà hàng có sự kết hợp của hình ảnh, văn bản, đồ họa 2 chiều/3 chiều và nhiều thông tin trực quan khác. Tất cả đều được tạo ra bằng WPF – lập trình viên không cần viết code để sử dụng các công nghệ chuyên biệt như GDI+ hay Direct3D.



Hình 2.1. 1 Giao diện quản lý nhà hàng bằng WPF

Việc tạo ra một giao diện người dùng hiện đại không chỉ là việc hợp nhất các công nghệ sẵn có khác nhau. Nó còn thể hiện ở việc tận dụng lợi điểm của card đồ họa hiện đại. Để giải phóng những hạn chế của đồ họa bitmap, WPF dựa hoàn toàn trên đồ họa vector, cho phép hình ảnh tự động thay đổi kích thước để phù hợp với kích thước và độ phân giải của màn hình mà nó được hiển thị.

Bằng việc hợp nhất tất cả các công nghệ cần thiết để tạo ra một giao diện người dùng vào một nền tảng đơn nhất, WPF đơn giản hóa đáng kể công việc của lập trình viên giao diện. Với việc yêu cầu lập trình viên học một môi trường phát triển duy nhất, WPF góp phần làm giảm chi phí cho việc xây dựng và bảo trì ứng dụng. Và bằng việc cho phép tích hợp đa dạng nhiều cách biểu diễn thông tin trên giao diện người dùng, WPF góp phần nâng cao

chất lượng, theo đó là giá trị công việc, của cách thức người dùng tương tác với ứng dụng trên Windows.

1.3 Khả năng làm việc chung giữa người thiết kế giao diện và lập trình viên

Trong thực tế, việc xây dựng một giao diện người dùng phức hợp như trong ví dụ về ứng dụng quản lý bệnh nhân trên đòi hỏi những kỹ năng ít thấy ở những lập trình viên đơn thuần, mà chỉ có thể tìm thấy ở những người thiết kế giao diện chuyên nghiệp. Nhưng câu hỏi đặt ra là làm sao để người thiết kế và lập trình viên có thể làm việc cùng nhau?

Thông thường người thiết kế giao diện sử dụng một công cụ đồ họa để tạo ra những ảnh tĩnh về cách bố trí giao diện trên màn hình. Những hình ảnh này sau đó được chuyển tới lập trình viên với nhiệm vụ tạo ra mã trình để hiện thực hóa giao diện đã thiết kế. Đôi lúc vẽ ra một giao diện thì đơn giản với người thiết kế, nhưng để biến nó thành hiện thực có thể là khó khăn hoặc bất khả thi với lập trình viên.

Hạn chế về công nghệ, sức ép tiến độ, thiếu kỹ năng hoặc đơn giản là bất đồng quan điểm có thể khiến lập trình viên không đáp ứng được đầy đủ yêu cầu từ người thiết kế. Do vậy, điều cần thiết ở đây là một cách thức để hai nhóm công tác độc lập này có thể làm việc với nhau mà không làm thay đổi chất lượng của giao diện đã thiết kế.

Để thực hiện được điều này WPF đưa ra ngôn ngữ đặc tả eXtensible Application Markup Language (viết tắt là XAML). XAML định ra một tập các phần tử XML như Button, TextBox, Label, v.v... nhằm định nghĩa các đối tượng đồ họa tương ứng như nút bấm, hộp thoại, nhãn, v.v... và nhờ đó cho phép mô tả chính xác diện mạo của giao diện người dùng. Các phần tử XAML cũng chứa các thuộc tính, cho phép thiết lập nhiều tính chất khác nhau của đối tượng đồ họa tương ứng.

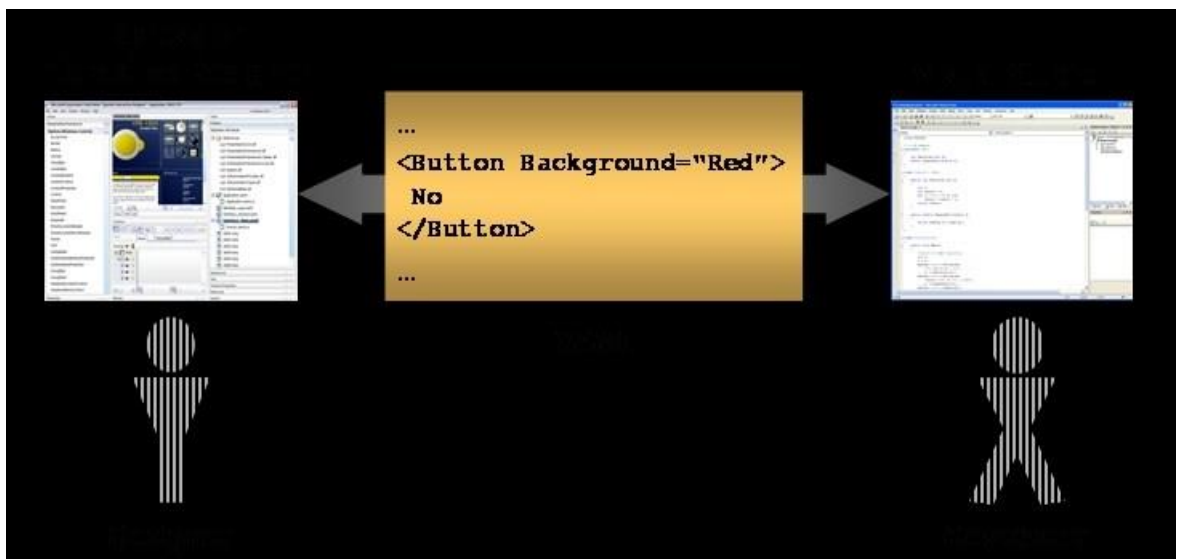
Ví dụ đoạn mã sau sẽ tạo ra một nút bấm màu đỏ “Click me”:

```
<Button x:Name= “btn” Content= “Click me”/>
```

Mỗi phần tử XAML lại tương ứng với một lớp WPF, và mỗi thuộc tính của phần tử đó lại tương ứng với thuộc tính hay sự kiện của lớp này. Chẳng hạn, nút bấm màu đỏ trong ví dụ trên có thể tạo bằng C# code như sau:

```
Button btn = new Button();  
Btn.Background = Brushes.Red;  
Btn.Content = "Click me";
```

Nếu như mọi thứ có thể biểu diễn bằng XAML thì cũng có thể biểu diễn bằng đoạn mã, thì câu hỏi đặt ra là XAML có ý nghĩa gì? Câu trả lời là việc xây dựng các công cụ sinh và sử dụng các đặc tả bằng XML dễ dàng hơn nhiều so với xây dựng một công cụ tương tự làm việc với đoạn mã. Bởi vậy, XAML mở ra một cách thức tốt hơn để lập trình viên và người thiết kế làm việc với nhau. *Hình 2.1.2* minh họa quá trình này.



Hình 2.1. 2 Quá trình làm việc giữa lập trình viên và người thiết kế giao diện khi sử dụng XAML

Người thiết kế có thể mô tả giao diện người dùng và tương tác với nó thông qua một công cụ, chẳng hạn như Microsoft Expression Interactive Designer. Chỉ tập trung vào việc định ra diện mạo và cảm quan cho giao diện đồ họa WPF, công cụ này sinh các đoạn mô tả giao diện thể hiện qua ngôn

ngữ XAML. Lập trình viên sau đó sẽ nhập đoạn mô tả XAML đó vào môi trường lập trình, chẳng hạn như Microsoft Visual Studio.

Thay vì lập trình viên phải tái tạo lại giao diện từ đầu dựa trên một ảnh tĩnh mà người thiết kế cung cấp, bản thân các đoạn XAML này sẽ được Microsoft Visual Studio biên dịch để tái tạo thành giao diện đồ họa đúng theo mô tả. Lập trình viên chỉ tập trung vào việc viết mã trình cho giao diện được sinh ra, chẳng hạn như xử lý các sự kiện, theo những chức năng đề ra của ứng dụng.

Việc cho phép người thiết kế và lập trình viên làm việc chung như vậy sẽ hạn chế những lỗi phát sinh khi hiện thực hóa giao diện từ thiết kế. Thêm vào đó, nó còn cho phép hai nhóm công tác này làm việc song song, khiến mỗi bước lập trong quy trình phát triển phần mềm ngắn đi và việc phản hồi được tốt hơn. Vì cả hai môi trường đều có khả năng hiểu và sử dụng XAML, ứng dụng WPF có thể chuyển qua lại giữa hai môi trường phát triển để sửa đổi hay bổ sung giao diện. Với tất cả những lợi điểm này, vai trò của người thiết kế trong việc xây dựng giao diện được đặt lên hàng đầu.

1.4 Hệ quản trị cơ sở dữ liệu

1.4.1 Giới thiệu về SQL Server 2017:

SQL Server 2017 là một hệ thống quản lý cơ sở dữ liệu (Relational Database Management System (RDBMS) sử dụng Transact – SQL để trao đổi dữ liệu giữa Client computer và SQL Server computer. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server 2017 hiện đang là phiên bản mới nhất của SQL Server và nó cũng như các phiên bản trước đây đều có những ưu điểm để người dùng lựa chọn như:

- **Sự lựa chọn về ngôn ngữ nền tảng:** Hiện tại trên hệ điều hành Windows, Linux và Docker bạn có thể xây dựng các ngôn ngữ bạn chọn, tại bộ nhớ máy cá nhân và trên điện toán đám mây.

- **Hiệu suất hàng đầu trong ngành:** Tận dụng khả năng mở rộng đột phá, hiệu suất và tính sẵn sàng cho các ứng dụng và kho dữ liệu thông minh, quan trọng.

- **Thời gian thực:** SQL Server 2017 có được những hiểu biết chuyên sâu về công việc bạn đang làm với các phân tích thời gian thực tới đa 1 triệu dự đoán / giây.

- **BI di động đầu cuối:** Biến dữ liệu thô thành các báo cáo có ý nghĩa có thể được gửi đến bất kỳ thiết bị nào với giá bằng một phần tư chi phí cho các giải pháp tự phục vụ khác.

Các phiên bản của SQL Server 2017:

- **SQL Server 2017 on premises:** Là một phiên bản dùng để xây dựng các ứng dụng thông minh, quan trọng cho nhiệm vụ bằng cách sử dụng nền tảng dữ liệu lại có thể mở rộng cho khối lượng công việc đòi hỏi sự khẩn trương.

- **SQL Server 2017 in the cloud:** Phiên bản này tận dụng tính sẵn sàng cao, bảo mật và sự thông minh của cơ sở dữ liệu SQL Azure và sử dụng công cụ SQL quen thuộc mà không cần sự phức tạp trong quản lý cơ sở hạ tầng.

- **Developer:** SQL Server 2017 Developer là phiên bản miễn phí đầy đủ tính năng, được cấp phép sử dụng làm cơ sở dữ liệu thử nghiệm và phát triển trong môi trường phi sản xuất.

- **Express:** SQL Server 2017 Express là phiên bản miễn phí của SQL Server, lý tưởng để phát triển và sản xuất cho máy tính để bàn, web và các ứng dụng máy chủ nhỏ.

1.4.2 SQL là ngôn ngữ của cơ sở dữ liệu quan hệ:

SQL, viết tắt của Structured Query Language, là công cụ sử dụng để tổ chức, quản lý và truy xuất dữ liệu được lưu trữ trong các cơ sở dữ liệu. SQL là một hệ thống ngôn ngữ bao gồm tập các câu lệnh sử dụng để tương tác với cơ sở dữ liệu quan hệ.

Khả năng của SQL vượt xa so với một công cụ truy xuất dữ liệu, mặc dù đây là mục đích ban đầu khi SQL được xây dựng nên và truy xuất dữ liệu vẫn

còn là một trong những chức năng quan trọng của nó. SQL được sử dụng để điều khiển tất cả các chức năng mà một hệ quản trị cơ sở dữ liệu cung cấp cho người dùng bao gồm:

- Định nghĩa dữ liệu: SQL cung cấp khả năng định nghĩa các cơ sở dữ liệu, các cấu trúc lưu trữ và tổ chức dữ liệu cũng như mối quan hệ giữa các thành phần dữ liệu.
- Truy xuất và thao tác dữ liệu: Với SQL, người dùng có thể dễ dàng thực hiện các thao tác truy xuất, bổ sung, cập nhật và loại bỏ dữ liệu trong các cơ sở dữ liệu.
- Điều khiển truy cập: SQL có thể được sử dụng để cấp phát và kiểm soát các thao tác của người sử dụng trên dữ liệu, đảm bảo sự an toàn cho cơ sở dữ liệu.
- Đảm bảo toàn vẹn dữ liệu: SQL định nghĩa các ràng buộc toàn vẹn trong cơ sở dữ liệu nhờ đó đảm bảo tính hợp lệ và chính xác của dữ liệu trước các thao tác cập nhật cũng như các lỗi của hệ thống.

Như vậy, có thể nói rằng SQL là một ngôn ngữ hoàn thiện được sử dụng trong các hệ thống cơ sở dữ liệu và là một thành phần không thể thiếu trong các hệ quản trị cơ sở dữ liệu.

- Mặc dù SQL không phải là một ngôn ngữ lập trình như C, C++, Java, v.v... xong các câu lệnh mà SQL cung cấp có thể được nhúng vào trong các ngôn ngữ lập trình nhằm xây dựng các ứng dụng tương tác với cơ sở dữ liệu.
- Khác với các ngôn ngữ lập trình quen thuộc như C, C++, Java, v.v... SQL là ngôn ngữ có tính khai báo. Với SQL người dùng chỉ cần mô tả các yêu cầu cần phải thực hiện trên cơ sở dữ liệu mà không cần phải chỉ ra cách thức thực hiện các yêu cầu như thế nào. Chính vì vậy SQL là ngôn ngữ dễ tiếp cận và dễ sử dụng.

1.4.3 Vai trò của SQL

Bản thân SQL không phải là một hệ quản trị cơ sở dữ liệu, nó không thể tồn tại độc lập.

SQL thực sự là một phần của hệ quản trị cơ sở dữ liệu, nó xuất hiện trong các hệ quản trị cơ sở dữ liệu với vai trò ngôn ngữ và là công cụ giao tiếp giữa người sử dụng và hệ quản trị cơ sở dữ liệu.

Trong hầu hết các hệ quản trị cơ sở dữ liệu quan hệ, SQL có những vai trò như sau:

- SQL là ngôn ngữ truy vấn có tính tương tác: Người sử dụng có thể dễ dàng thông qua các trình tiện ích để gửi các yêu cầu dưới dạng các câu lệnh SQL đến cơ sở dữ liệu và nhận kết quả trả về từ cơ sở dữ liệu.
 - SQL là ngôn ngữ lập trình cơ sở dữ liệu: Các lập trình viên có thể nhúng các câu lệnh SQL vào trong các ngôn ngữ lập trình để xây dựng nên các chương trình ứng dụng giao tiếp với cơ sở dữ liệu.
 - SQL là ngôn ngữ quản trị cơ sở dữ liệu: Thông qua SQL, người quản trị cơ sở dữ liệu có thể quản lý được cơ sở dữ liệu, định nghĩa các cấu trúc lưu trữ dữ liệu, điều khiển truy cập cơ sở dữ liệu, v.v...
 - SQL là ngôn ngữ cho các hệ thống khách/chủ (client/server): Trong các hệ thống cơ sở dữ liệu khách/chủ, SQL được sử dụng như là công cụ để giao tiếp giữa các trình ứng dụng phía máy khách với máy chủ cơ sở dữ liệu.
 - SQL là ngôn ngữ truy cập dữ liệu trên Internet: Cho đến nay, hầu hết các máy chủ Web cũng như các máy chủ trên Internet sử dụng SQL với vai trò là ngôn ngữ để tương tác với dữ liệu trong các cơ sở dữ liệu.
 - SQL là ngôn ngữ cơ sở dữ liệu phân tán: Đối với các hệ quản trị cơ sở dữ liệu phân tán, mỗi một hệ thống sử dụng SQL để giao tiếp với các hệ thống khác trên mạng, gửi và nhận các yêu cầu truy xuất dữ liệu với nhau.
 - SQL là ngôn ngữ sử dụng cho các cổng giao tiếp cơ sở dữ liệu: Trong một hệ thống mạng máy tính với nhiều hệ quản trị cơ sở dữ liệu khác nhau, SQL thường được sử dụng như là một chuẩn ngôn ngữ để giao tiếp giữa các hệ quản trị cơ sở dữ liệu.
-

Chương 2. Quy trình xử lý nhận dạng biển số

2.1 Giới thiệu thư viện EmguCV

EmguCV là một thư viện xử lý ảnh mạnh dành riêng cho ngôn ngữ C#. Cho phép gọi được chức năng của OpenCV từ .NET.

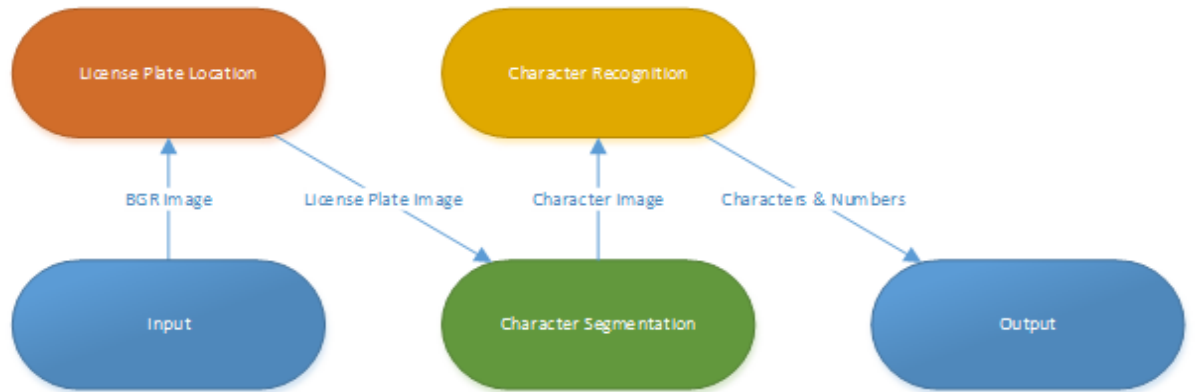
EmguCV tương thích với ngôn ngữ lập trình như: C#, VB, VC++, Iron Python, v.v...

EmguCV có thể chạy trên Windows, Linux, Mac OS, iOS, Android và Windows Phone.

Ưu điểm của EmguCV:

- EmguCV được viết hoàn toàn bằng C#. Có thể chạy trên bất kỳ nền tảng hỗ trợ bao gồm iOS, Android, Windows Phone, hệ điều hành Mac OS X và Linux.
- EmguCV có thể được sử dụng từ nhiều ngôn ngữ khác nhau, bao gồm C#, VB.NET, C++ và Iron Python.
- Nhận dạng ảnh: nhận dạng khuôn mặt, các vật thể, v.v...
- Xử lý ảnh: khử nhiễu, điều chỉnh độ sáng, v.v...
- Nhận dạng cử chỉ.
- Sự lựa chọn để sử dụng hình ảnh lớp hoặc trực tiếp gọi chức năng từ OpenCV.
- V.v...

2.2 Quy trình xử lý nhận dạng biển số xe



Hình 2.2. 1 Mô hình tổng quan bài toán nhận dạng biển số xe

Các bước tiến hành:

- Chuẩn hóa ảnh đầu vào: Ảnh đầu vào cần được chuẩn hóa kích thước 640 x 480 trước khi được xử lý xác định vùng biển số.
- Xác định vùng biển số: Xác định vùng biển số là công việc xác định gần chính xác vị trí 4 đỉnh của khung chữ nhật nhỏ nhất chứa toàn bộ biển số xe từ ảnh đầu vào.
- Bóc tách ký tự: Vùng biển số sau khi được phát hiện thường chứa rất nhiều các đối tượng ngoài ký tự như mép biển số hay vỏ xe, đinh ốc, v.v... Do đó việc bóc tách ký tự ở đây chính là việc lọc bỏ toàn bộ hoặc hầu hết các đối tượng dư thừa và định vị được vị trí của từng ký tự trên biển nhằm phục vụ cho tiến trình nhận dạng ký tự cuối cùng.
- Nhận dạng ký tự: Đưa ảnh của từng ký tự rời rạc cần được đối chiếu với một thư viện dữ liệu có sẵn trong hệ thống để đưa ra kết quả biển số cuối cùng.

Chương 3. Tìm hiểu đề tài và phân tích thiết kế hệ thống

3.1 Khảo sát hệ thống



Hình 2.3. 1 Sơ đồ quy trình gửi xe tại bãi gửi xe thông minh

3.1.1 Khảo sát sơ bộ

3.1.1.1 Khái niệm

Hệ thống bãi gửi xe thông minh là một hệ thống tích hợp các công nghệ tiên tiến và thiết bị hiện đại như: giám sát bằng hệ thống camera, kiểm soát lưu lượng xe vào - ra bằng hệ thống thẻ từ, nhận diện biển số ...

Là giải pháp mang tính an toàn cao, giảm thiểu được rủi ro thất thoát tài chính và tăng cao hiệu quản lý.

3.1.1.2 Mục tiêu

Thiết kế một hệ thống giữ xe tự động sử dụng công nghệ quét thẻ. Khi vào gửi xe, người gửi quét thẻ của mình đồng thời hệ thống chụp biển số và gương mặt của chủ xe. Tất cả chỉ xảy ra trong vòng 4 đến 5 giây. Lúc lấy xe ra người gửi quét thẻ lần nữa và hệ thống sẽ hiện lên hình ảnh chụp lúc gửi xe. Nhân viên giữ xe so sánh, nhận dạng và quyết định cho xe ra.

3.1.1.3 Nhiệm vụ

Tạo ra một hệ thống gửi xe tự động có thể hoạt động được và sau này có thể phát triển thêm để áp dụng được vào thực tế sử dụng cho các trường học, siêu thị và các bãi gửi xe công cộng.

3.1.2 Khảo sát chi tiết

Hoạt động của nhân viên: Đầu tiên nhân viên cần phải đăng nhập vào hệ thống. Khi chạy chương trình sẽ hiển thị cửa sổ đăng nhập. Nhân viên nhập tên người dùng và mật khẩu do người quản lý cung cấp. Sau khi đăng nhập, phần mềm quản lý bãi gửi xe vào trang chủ và sẽ phân quyền sử dụng các chức năng đối với người dùng tùy thuộc vào quyền đã được phân trước.

Hoạt động quản lý xe ra vào của bãi gửi xe: Hệ thống có thể quản lý 2 làn xe ra, vào cùng lúc. Có thể đối chiếu làn xe theo ý của người quản lý bãi gửi xe thông minh chụp ảnh của xe và biển số xe vào ra bãi gửi xe. Tự động nhận diện biển số khi xe vào ra. Khi xuất hiện tình trạng không khớp biển số hay chủ xe thì sẽ có tiếng còi báo và chuông cảnh báo do cho nhân viên giữ xe thao tác trên phần mềm.

3.2 Phân tích thiết kế hệ thống

3.2.1 Xây dựng biểu đồ Use case

3.2.1.1 Tác nhân trong hệ thống

STT	Tên Actor	Ý nghĩa
1	Người quản lý (Admin)	Người quản lý là người điều hành, toàn quyền quản lý hệ thống phần mềm với đầy đủ chức năng.
2	Nhân viên (User)	Nhân viên có thể truy cập vào phần mềm và sử dụng một số chức năng được phân quyền trong phần mềm.

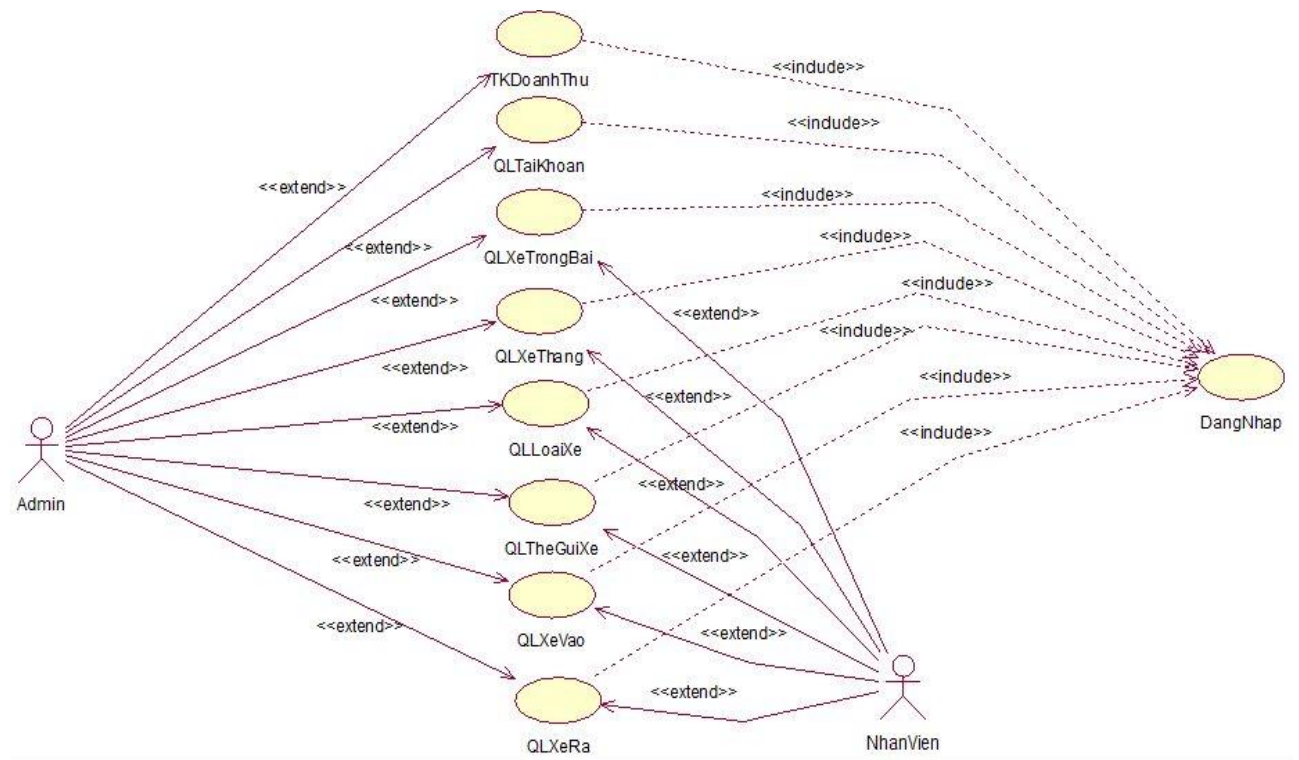
Bảng 2.3. 1 Các tác nhân trong hệ thống

3.2.1.2 Các use case trong hệ thống

STT	Tên Usecase	Ý nghĩa
1	Đăng nhập	UC này mô tả chức năng đăng nhập vào hệ thống của Admin và nhân viên trông xe.
2	Quản lý tài khoản	UC này mô tả chức năng do admin thực hiện bao gồm nhập mới thông tin, sửa thông tin, xóa thông tin, tìm kiếm tài khoản và đổi mật khẩu đăng nhập.
3	Quản lý xe trong bãi	UC này mô tả chức năng liệt kê danh sách các xe hiện tại đang có trong bãi và cho phép người dùng tìm kiếm thông tin xe qua biển số xe.
4	Quản lý xe tháng	UC này mô tả chức năng do admin thực hiện nhập mới thông tin, sửa thông tin, xóa thông tin của các xe đăng ký vé tháng và với tài khoản nhân viên có thể xem danh sách các xe đăng ký vé tháng.
5	Quản lý loại xe	UC này mô tả chức năng do admin thực hiện nhập mới thông tin, sửa thông tin, xóa thông tin.
6	Quản lý thẻ gửi xe	UC mô tả chức năng nhập mới thông tin, sửa thông tin, xóa thông tin của thẻ gửi xe.
7	Thống kê doanh thu	UC mô tả chức năng do admin thực hiện thống kê doanh thu theo từng tháng, năm.
8	Quản lý xe vào	UC mô tả chức năng nhập thông tin xe và cho xe vào bãi.
9	Quản lý xe ra	UC mô tả chức năng nhập thông tin thẻ khách đưa và cho xe ra.

Bảng 2.3. 2 Các use case trong hệ thống

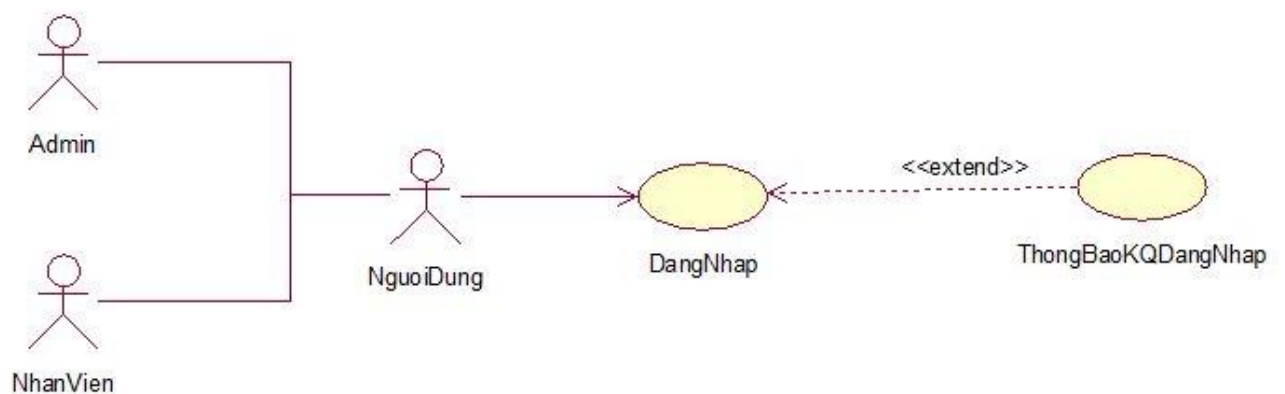
3.2.2 Biểu đồ Use case tổng thể



Hình 2.3. 2 Biểu đồ các use case tổng thể

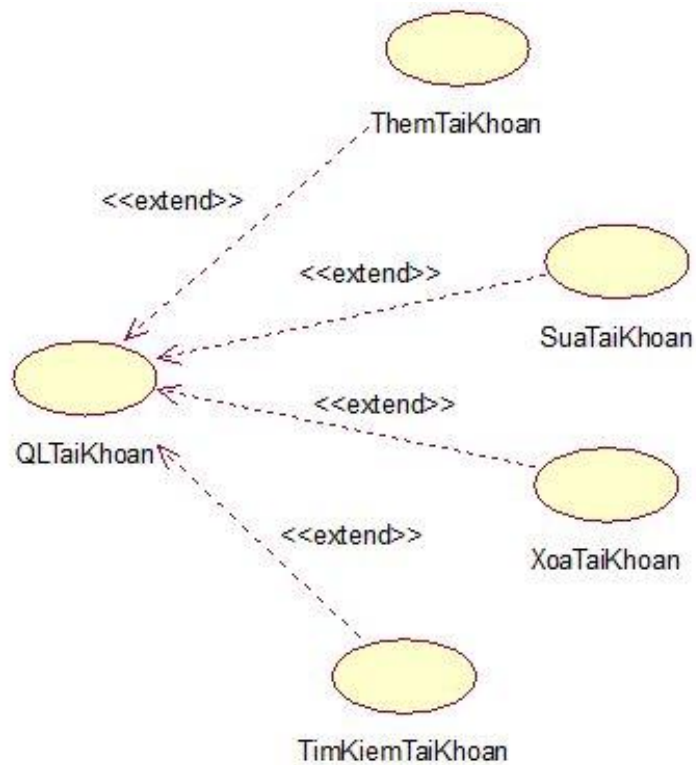
3.2.3 Biểu đồ use case phân rã

3.2.3.1 Biểu đồ phân rã chức năng đăng nhập



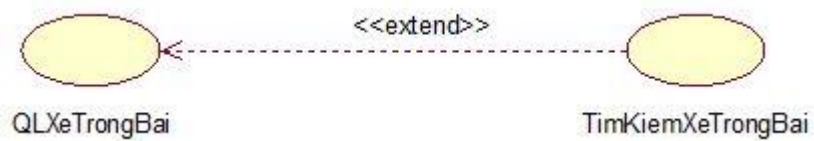
Hình 2.3. 3 Biểu đồ phân rã đăng nhập

3.2.3.2 Biểu đồ phân rã chức năng quản lý tài khoản



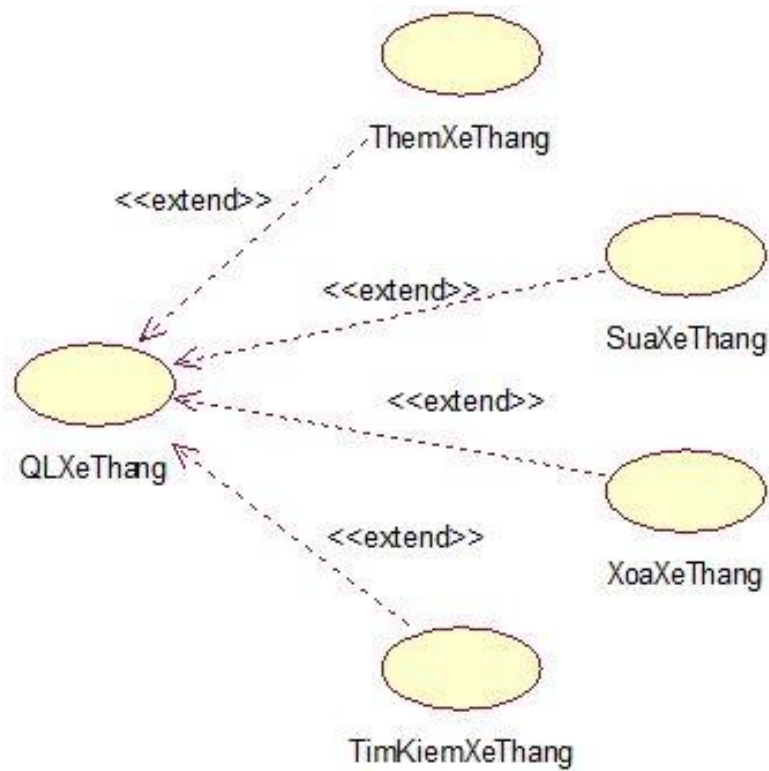
Hình 2.3. 4 Biểu đồ phân rã quản lý tài khoản

3.2.3.3 Biểu đồ phân rã chức năng quản lý xe trong bãi



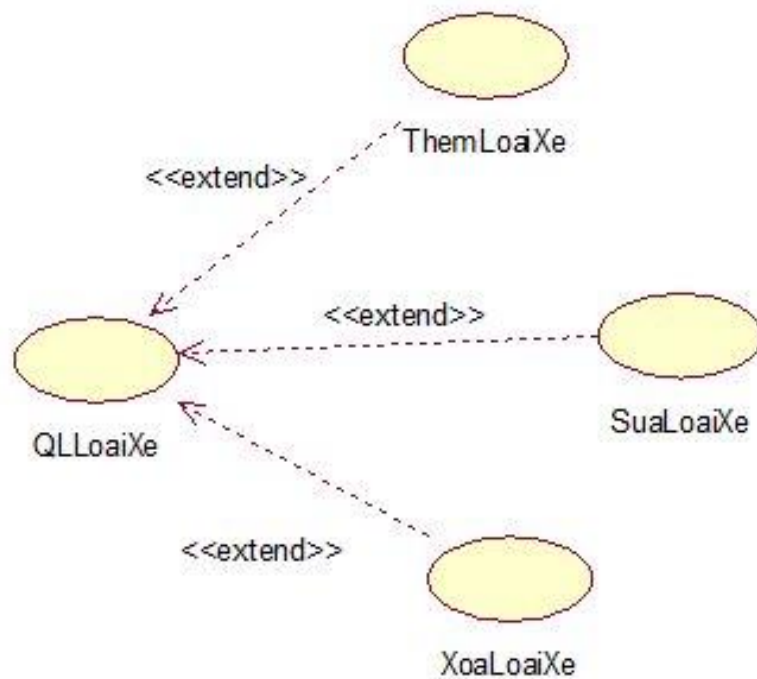
Hình 2.3. 5 Biểu đồ phân rã quản lý xe trong bãi

3.2.3.4 Biểu đồ phân rã chức năng quản lý xe gửi theo tháng



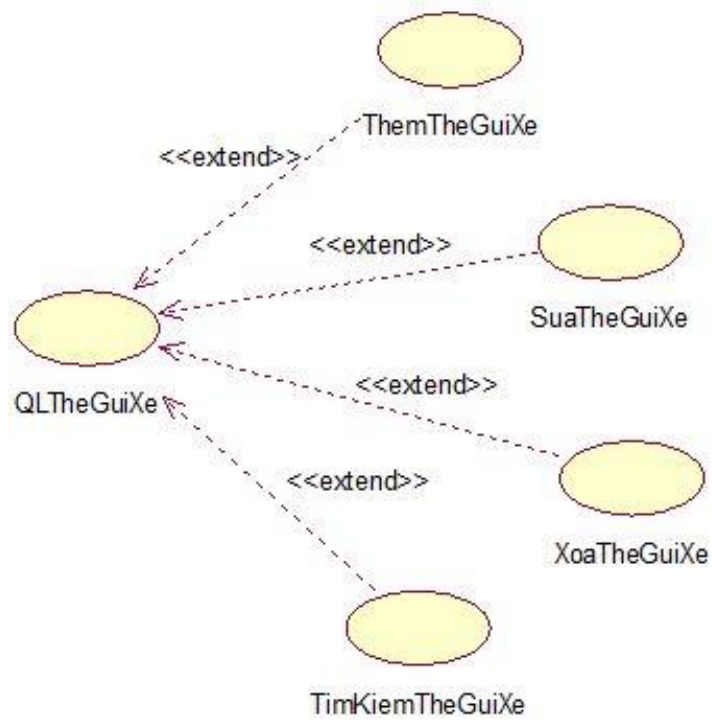
Hình 2.3. 6 Biểu đồ phân rã quản lý xe tháng

3.2.3.5 Biểu đồ phân rã chức năng quản lý loại xe



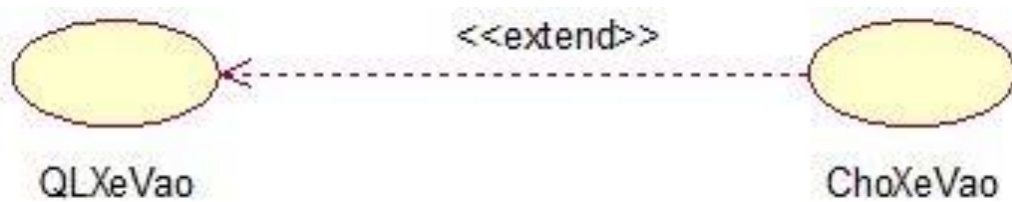
Hình 2.3. 7 Biểu đồ phân rã quản lý loại xe

3.2.3.6 Biểu đồ phân rã chức năng quản lý thẻ gửi xe



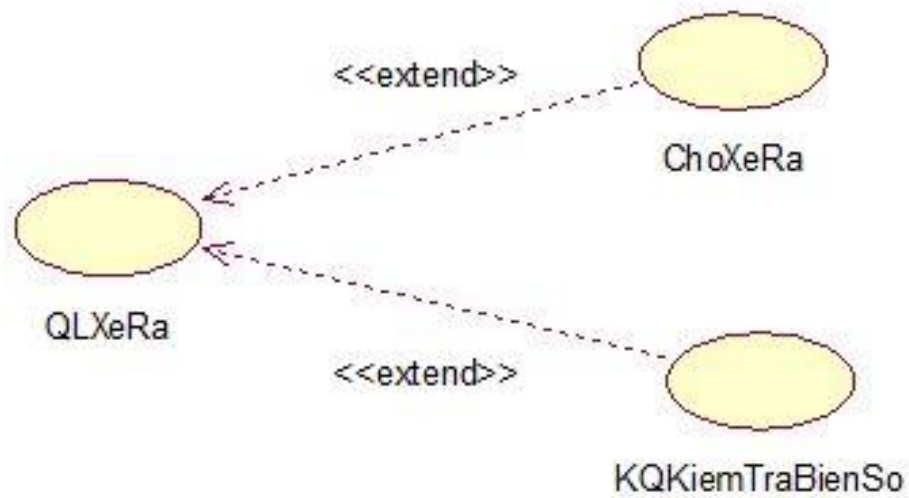
Hình 2.3. 8 Biểu đồ phân rã quản lý thẻ gửi xe

3.2.3.7 Biểu đồ phân rã chức năng quản lý xe vào



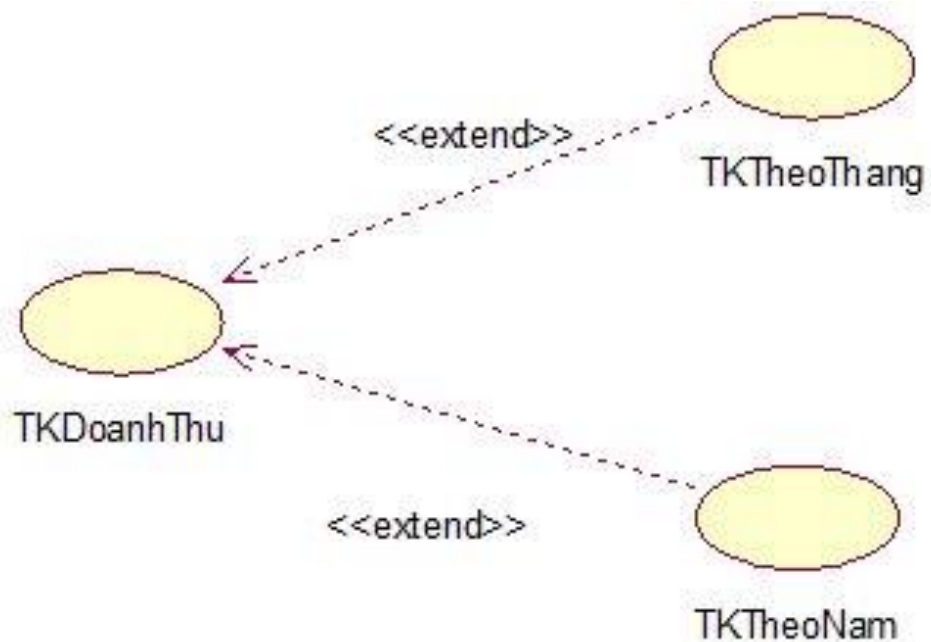
Hình 2.3. 9 Biểu đồ phân rã quản lý xe vào

3.2.3.8 Biểu đồ phân rã chức năng quản lý xe ra



Hình 2.3. 10 Biểu đồ phân rã quản lý xe ra

3.2.3.9 Biểu đồ phân rã thống kê doanh thu



Hình 2.3. 11 Biểu đồ phân rã thống kê doanh thu

3.2.4 Đặc tả ca sử dụng

3.2.4.1 Use case: Đăng nhập

Tên use case: Đăng nhập.

Mô tả vắn tắt: Use case này cho phép người dùng (nhân viên và admin) đăng nhập vào hệ thống bằng cách nhập tên đăng nhập và mật khẩu để truy cập các chức năng của hệ thống.

Tác nhân: Admin, nhân viên

Luồng sự kiện:

- **Luồng sự kiện chính:**

Bước 1: Use case này bắt đầu khi tác nhân mở chương trình phần mềm. Hệ thống sẽ hiển thị một màn hình đăng nhập.

Bước 2: Người dùng nhập tên đăng nhập bằng chính tên được cấp bởi hệ thống đăng kí và mật khẩu vào các trường tương ứng và nhấn nút “**Đăng nhập**”. Hệ thống sẽ kiểm tra thông tin tài khoản có tồn tại hay không và ứng với quyền người dùng nào. Nếu nhập sai tài khoản hoặc mật khẩu đăng nhập thì thực hiện luồng rẽ nhánh A1.

- **Luồng rẽ nhánh A1:** Nhập sai tài khoản hoặc mật khẩu đăng nhập. Hệ thống sẽ hiển thị thông báo lỗi. Người dùng có thể chọn đăng nhập hoặc hủy đăng nhập khi đó ca sử dụng sẽ kết thúc.

Hậu điều kiện: Nếu việc đăng nhập thành công người dùng sẽ đăng nhập được vào hệ thống và thực hiện chức năng ứng với quyền của mình.

3.2.4.2 Use case: Quản lý tài khoản

Tên use case: Quản lý tài khoản

Mô tả vắn tắt: Use case này cho phép người admin cập nhật thông tin nhân viên trong hệ thống bao gồm thêm, sửa, xóa, tìm kiếm cho các tài khoản người dùng trong hệ thống.

Tác nhân: Admin.

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện:

- **Luồng sự kiện chính:** Use case bắt đầu khi admin muốn thêm, sửa, xóa hoặc đổi mật khẩu cho người dùng trong hệ thống. Hệ thống sẽ hiển thị danh sách các tài khoản người dùng hiện có trong hệ thống và các lựa chọn thêm, sửa, xóa, tìm kiếm. Admin sẽ lựa chọn công việc mà mình muốn thực hiện.

- **Thêm người dùng:** Admin muốn thêm tài khoản người dùng mới vào hệ thống thì chỉ cần nhấn nút “**Thêm**”. Sau đó admin sẽ nhập đầy đủ thông tin bắt buộc. Sau khi admin nhập đầy đủ các thông tin cần thiết nếu admin muốn lưu thông tin vừa nhập thì kích nút “**Lưu**” còn ngược lại thì tiếp tục thực hiện luồng rẽ nhánh A1. Sau khi kích nút “**Lưu**” hệ thống sẽ kiểm tra dữ liệu nhập vào ở luồng rẽ nhánh A2, nếu thông tin nhập vào hợp lệ thì hệ thống sẽ thêm thành công và kết thúc use case.

- **Sửa người dùng:** Admin muốn sửa thông tin tài khoản người dùng bất kỳ hiện có trong hệ thống thì chỉ cần chọn tài khoản người dùng cần sửa. Sau đó nhập các thông tin cần sửa vào trong hệ thống. Admin muốn lưu thông tin vừa sửa thì kích nút “**Lưu**” còn ngược lại thì thực hiện luồng rẽ nhánh A3. Sau khi kích nút “**Lưu**” hệ thống sẽ kiểm tra dữ liệu nhập vào và thực hiện luồng rẽ nhánh A2. Sửa thành công thì kết thúc use case.

- **Xóa thông tin:** Tác nhân chỉ cần chọn nhân viên muốn xóa rồi sau đó kích nút “**Xóa**” và kích nút “**Lưu**” khi chắc chắn muốn xóa còn ngược lại thì thực hiện luồng rẽ nhánh A4. Hệ thống sẽ kiểm tra dữ liệu rằng buộc thực hiện tại luồng rẽ nhánh A5. Xóa thành công, kết thúc use case.

- **Tìm kiếm:** Admin muốn tìm kiếm một nhân viên trong hệ thống. Admin chỉ cần chọn nhập tài khoản cần tìm kiếm và kích nút “**Tìm**”. Hệ thống sẽ truy suất dữ liệu và đưa ra các thông tin về tài khoản mà tác nhân muốn tìm kiếm (Luồng rẽ nhánh A6). Tìm kiếm thành công thì kết thúc use case.

- **Luồng sự kiện rẽ nhánh**

○ **Luồng rẽ nhánh A1:** Admin không muốn thêm tài khoản người dùng mới thì kích nút “**Hủy**” trên giao diện. Thêm không thành công, kết thúc use case.

○ **Luồng rẽ nhánh A2:** Hệ thống kiểm tra dữ liệu nhập vào có sai sót, đưa ra thông báo “**Lưu không thành công**”. Yêu cầu nhập lại.

○ **Luồng rẽ nhánh A3:** Người tác nhân không muốn thay đổi thông tin tài khoản người dùng, kích nút “**Hủy**” trên giao diện. Thay đổi không thành công, kết thúc use case.

○ **Luồng rẽ nhánh A4:** Tác nhân không muốn xóa tài khoản, ấn nút “**Hủy**” trên giao diện. Xóa không thành công, kết thúc use case

○ **Luồng rẽ nhánh A5:** Hệ thống kiểm tra ràng buộc dữ liệu tài khoản không được xóa, đưa ra thông báo “**Không thể xóa**”, use case kết thúc.

○ **Luồng rẽ nhánh A6:** Hệ thống không tìm thấy tài khoản mà admin muốn tìm kiếm thì sẽ hiện ra một thông báo là không tìm thấy tài khoản.

Hậu điều kiện: Nếu ca sử dụng được thực hiện thành công thì thông tin tài khoản sẽ được thêm, sửa, xóa, tìm kiếm còn lại các trường hợp khác thông tin vẫn được giữ nguyên.

3.2.4.3 Use case: Quản lý thẻ gửi xe

Tên use case: Quản lý thẻ gửi xe

Mục đích: Giúp admin có thể cập nhật lại thông tin về các thẻ gửi xe trên hệ thống.

Tác nhân: Admin, nhân viên

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện:

- **Luồng sự kiện chính:** Sau khi đăng nhập thành công, tác nhân chọn “**Quản lý thẻ gửi xe**”. Hệ thống sẽ hiển thị danh sách các thẻ gửi xe hiện có trong hệ thống và các lựa chọn thêm, sửa, xóa, tìm kiếm. Tác nhân sẽ lựa chọn công việc mà mình muốn thực hiện.

○ **Thêm thẻ gửi xe:** Tác nhân muốn thêm một thẻ gửi xe mới vào hệ thống thì chỉ cần nhấn nút “**Thêm**”. Sau đó nhập đầy đủ các thông tin bắt buộc. Sau khi tác nhân nhập đầy đủ thông tin thì kích nút “**Lưu**” để lưu thông tin mới còn ngược lại hệ thống thực hiện luồng rẽ nhánh A1. Hệ thống sẽ kiểm tra dữ liệu nhập vào và thực hiện luồng rẽ nhánh A2, nếu hợp lệ thì sẽ thêm thành công và kết thúc use case.

○ **Sửa thông tin thẻ gửi xe:** Tác nhân muốn sửa thông tin thẻ gửi xe. Tác nhân chỉ cần chọn thẻ gửi xe muốn sửa trên danh sách và sửa các thông tin trong hệ thống và kích nút “**Lưu**” khi muốn lưu lại thông tin mình vừa sửa còn ngược lại hệ thống sẽ thực hiện luồng rẽ nhánh A3. Hệ thống sẽ kiểm tra dữ liệu nhập vào và thực hiện luồng rẽ nhánh A2. Sửa thành công thì kết thúc use case.

○ **Xóa thông tin thẻ gửi xe:** Tác nhân muốn xóa một thẻ gửi xe. Tác nhân chỉ cần chọn thẻ gửi xe muốn xóa rồi ấn nút “**Xóa**” sau đó tiếp tục kích nút “**Lưu**” khi chắc chắn muốn xóa thẻ gửi xe đó còn ngược lại hệ thống thực hiện luồng rẽ nhánh A4. Hệ thống sẽ kiểm tra dữ liệu ràng buộc và thực hiện luồng rẽ nhánh A5. Xóa thành công thì kết thúc use case.

○ **Tìm kiếm thông tin thẻ gửi xe:** Use case này bắt đầu khi tác nhân muốn tìm kiếm một thẻ gửi xe có trong hệ thống. Tác nhân chỉ cần nhập mã thẻ và kích nút “**Tìm kiếm**”. Hệ thống sẽ xử lý và đưa ra các thông tin về loại vé xe và ngược lại sẽ thực hiện luồng rẽ nhánh A7.

- **Luồng sự kiện rẽ nhánh**

○ **Luồng rẽ nhánh A1:** Tác nhân không muốn thêm mới thì chỉ cần nhấn nút “**Hủy**” trên giao diện thêm. Thêm không thành công thì kết thúc use case.

○ **Luồng rẽ nhánh A2:** Hệ thống kiểm tra dữ liệu nhập vào, đưa ra thông báo “**Lưu không thành công / Sửa không thành công**”. Yêu cầu nhập lại.

○ **Luồng rẽ nhánh A3:** Tác nhân không muốn sửa thông tin về thẻ gửi xe thì chỉ cần nhấn nút “**Hủy**” trên màn hình. Sửa không thành công thì kết thúc use case.

○ **Luồng rẽ nhánh A4:** Tác nhân không muốn xóa thông tin về một thẻ gửi xe thì chỉ cần nhấn nút “**Hủy**” trên màn hình giao diện. Xóa không thành công thì kết thúc use case.

○ **Luồng rẽ nhánh A5:** Hệ thống sẽ kiểm tra ràng buộc dữ liệu về thẻ gửi xe muốn xóa, đưa ra thông báo “**Không thể xóa**”, use case kết thúc.

Hậu điều kiện: Nếu ca sử dụng được thực hiện thành công thì thông tin thẻ gửi xe được thêm, sửa, xóa, tìm kiếm ngược lại các trường hợp khác thông tin vẫn được giữ nguyên.

3.2.4.4 Use case: Quản lý loại xe

Tên use case: Quản lý loại xe

Mô tả vắn tắt: Use case này cho phép tác nhân quản lý các loại xe có trong bãi gửi xe của hệ thống bao gồm nhập, sửa, xóa, tìm kiếm các loại xe.

Tác nhân: Admin, nhân viên

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện

• **Luồng sự kiện chính:** Use case này bắt đầu khi tác nhân muốn nhập, sửa, xóa, tìm kiếm thông tin về các loại xe có trong hệ thống. Hệ thống sẽ hiển thị danh sách các loại xe hiện có trong hệ thống và các lựa chọn thêm, sửa, xóa, tìm kiếm. Tác nhân sẽ lựa chọn công việc mà mình muốn thực hiện.

○ **Nhập thông tin:** Tác nhân muốn nhập thông tin mới vào hệ thống thì chỉ cần nhấn nút “**Thêm**”. Sau đó nhập đầy đủ các thông tin bắt buộc. Sau khi tác nhân nhập đầy đủ thông tin thì kích nút “**Lưu**” để lưu thông tin mới còn ngược lại hệ thống thực hiện luồng rẽ nhánh A1. Hệ thống sẽ kiểm tra dữ liệu nhập vào và thực hiện luồng rẽ nhánh A2, nếu hợp lệ thì sẽ thêm thành công và kết thúc use case.

○ **Sửa thông tin:** Tác nhân muốn sửa thông tin về một loại xe trong hệ thống. Tác nhân chỉ cần chọn loại xe muốn sửa sau đó nhập các thông tin cần sửa vào trong hệ thống và kích nút “**Lưu**” để lưu lại thông tin cần sửa ngược lại luồng rẽ nhánh A3. Hệ thống sẽ kiểm tra dữ liệu nhập vào hệ thống thực hiện luồng rẽ nhánh A2. Sửa thành công thì kết thúc use case.

○ **Xóa thông tin:** Tác nhân chỉ cần chọn loại xe muốn xóa rồi sau đó kích nút “**Xóa**” để chắc chắn xóa loại xe vừa chọn tác nhân tiếp tục kích nút lưu ngược lại hệ thống thực hiện luồng rẽ nhánh A4. Hệ thống sẽ kiểm tra dữ liệu rằng buộc hệ thống thực hiện luồng rẽ nhánh A5. Xóa thành công, kết thúc use case.

• **Luồng sự kiện rẽ nhánh**

○ **Luồng rẽ nhánh A1:** Tác nhân không muốn thêm thông tin mới, ấn nút “**Hủy**” trên giao diện. Thêm không thành công, kết thúc use case.

○ **Luồng rẽ nhánh A2:** Hệ thống kiểm tra dữ liệu nhập vào có sai sót, đưa ra thông báo “**Lưu không thành công**”. Yêu cầu nhập lại.

○ **Luồng rẽ nhánh A3:** Tác nhân không muốn thay đổi thông tin loại xe, ấn nút “**Hủy**” trong giao diện. Thay đổi không thành công, kết thúc use case.

○ **Luồng rẽ nhánh A4:** Tác nhân không muốn xóa một loại xe trong bãi gửi xe, kích nút “**Hủy**” trên giao diện. Xóa không thành công, kết thúc use case.

○ **Luồng rẽ nhánh A5:** Hệ thống kiểm tra rằng buộc dữ liệu loại xe không được xóa, đưa ra thông báo “**Không thể xóa**”. Use case kết thúc.

○ **Luồng rẽ nhánh A7:** Hệ thống không tìm thấy loại xe mà tác nhân muốn tìm kiếm thì đưa ra thông báo không tìm thấy.

Hậu điều kiện: Nếu ca sử dụng được thực hiện thành công thì thông tin loại xe sẽ được thêm, sửa, xóa, tìm kiếm còn lại các trường hợp khác thông tin vẫn được giữ nguyên.

3.2.4.5 Use case: Quản lý xe vào

Tên use case: Quản lý xe vào

Mô tả vắn tắt: Use case này cho phép nhân viên có thể thêm các xe vào trong hệ thống khi mà có xe vào gửi trong bãi đậu xe.

Tác nhân: Admin, nhân viên

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện:

- **Luồng sự kiện chính:** Use case này bắt đầu khi tác nhân kích vào nút “**Quản lý xe vào**” trên menu chính của hệ thống. Hệ thống sẽ hiển thị ra màn hình quản lý xe vào. Tác nhân chỉ cần mở ảnh biển số mà camera đã chụp. Hệ thống sẽ xử lý hình ảnh và lấy ra được thông tin của biển số xe. Tác nhân sẽ chọn loại xe và đưa thẻ cho khách hàng và nhấn nút xác nhận (Luồng rẽ nhánh A1). Nếu thêm thành công thì hệ thống sẽ báo cập nhật thông tin thành công và kết thúc use case (Luồng rẽ nhánh A2).

- **Luồng sự kiện rẽ nhánh:**

- **Luồng rẽ nhánh A1:** Tác nhân không muốn thêm xe vào trong bãi gửi xe thì chỉ cần thoát khỏi chức năng quản lý xe vào.

- **Luồng rẽ nhánh A2:** Nếu hệ thống không thêm thành công thì sẽ báo lỗi ra màn hình hiển thị để tác nhân biết và sửa lại để có thể thêm thành công xe vào bãi gửi xe.

3.2.4.6 Use case: Quản lý xe ra

Tên use case: Quản lý xe ra

Mô tả vắn tắt: Use case này cho phép nhân viên có thể kiểm soát được các xe ra khỏi bãi gửi xe.

Tác nhân: Nhân viên

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện

- **Luồng sự kiện chính:** Use case này bắt đầu khi tác nhân kích vào nút “**Quản lý xe ra**” trên menu chính của hệ thống. Hệ thống sẽ hiển thị ra màn hình quản lý xe ra. Tác nhân chỉ cần mở ảnh biển số mà camera đã chụp lúc xe ra. Hệ thống sẽ xử lý hình ảnh và lấy ra được thông tin của biển số xe và các thông tin của xe lúc xe đi vào. Tác nhân sẽ lấy lại vé xe mà chủ xe đưa và xác nhận lại xem vé xe có trùng khớp với biển số lúc mà xe đi vào hay không (Luồng rẽ nhánh A1). Tác nhân ấn xác nhận và hệ thống sẽ xử lý và đưa ra thông báo hoàn tất (Luồng rẽ nhánh A2)

- **Luồng sự kiện rẽ nhánh**

- **Luồng rẽ nhánh A1:** Nếu vé xe không trùng khớp với biển số thì hệ thống sẽ đưa ra thông báo và tác nhân sẽ kiểm tra và xử lý.

- **Luồng rẽ nhánh A2:** Khi người chủ xe muốn quay lại bãi đậu xe thì người tác nhân chỉ cần thoát khỏi màn hình quản lý xe ra.

3.2.4.7 Use case: Quản lý xe tháng

Tên use case: Quản lý xe tháng

Mô tả vắn tắt: Use case này cho phép tác nhân cập nhật thông tin xe gửi theo tháng trong hệ thống bao gồm thêm, sửa, xóa, tìm kiếm cho các xe gửi theo tháng trong hệ thống.

Tác nhân: Admin, nhân viên

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện

- **Luồng sự kiện chính:** Use case này bắt đầu khi tác nhân muốn nhập, sửa, xóa, tìm kiếm thông tin về các xe gửi theo tháng có trong hệ thống. Hệ thống sẽ hiển thị danh sách các xe gửi theo tháng hiện có trong hệ thống và các lựa chọn thêm, sửa, xóa, tìm kiếm. Tác nhân sẽ lựa chọn công việc mà mình muốn thực hiện.

- **Nhập thông tin:** Tác nhân muốn nhập thông tin mới vào hệ thống thì chỉ cần nhấn nút “**Thêm**”. Sau đó nhập đầy đủ các thông tin bắt buộc. Sau

khi tác nhân nhập đầy đủ thông thì kích nút **“Lưu”** để lưu thông thể mới còn ngược lại hệ thống thực hiện luồng rẽ nhánh A1. Hệ thống sẽ kiểm tra dữ liệu nhập vào và thực hiện luồng rẽ nhánh A2, nếu hợp lệ thì sẽ thêm thành công và kết thúc use case.

- **Sửa thông tin:** Tác nhân muốn sửa thông tin về một xe gửi theo tháng trong hệ thống. Tác nhân chỉ cần chọn xe muốn sửa sau đó nhập các thông tin cần sửa vào trong hệ thống và kích nút **“Lưu”** để lưu lại thông tin cần sửa ngược lại luồng rẽ nhánh A3. Hệ thống sẽ kiểm tra dữ liệu nhập vào hệ thống thực hiện luồng rẽ nhánh A2. Sửa thành công thì kết thúc use case.

- **Xóa thông tin:** Tác nhân chỉ cần chọn xe gửi theo tháng muốn xóa rồi sau đó kích nút **“Xóa”** để chắc chắn xóa xe vừa chọn tác nhân tiếp tục kích nút **“Lưu”** ngược lại hệ thống thực hiện luồng rẽ nhánh A4. Hệ thống sẽ kiểm tra dữ liệu ràng buộc hệ thống thực hiện luồng rẽ nhánh A5. Xóa thành công, kết thúc use case.

- **Luồng sự kiện rẽ nhánh**

- **Luồng rẽ nhánh A1:** Tác nhân không muốn thêm thông tin mới, ấn nút **“Hủy”** trên giao diện. Thêm không thành công, kết thúc use case.

- **Luồng rẽ nhánh A2:** Hệ thống kiểm tra dữ liệu nhập vào có sai sót, đưa ra thông báo **“Lưu không thành công”**. Yêu cầu nhập lại.

- **Luồng rẽ nhánh A3:** Tác nhân không muốn thay đổi thông tin xe gửi theo tháng, kích nút **“Hủy”** trong giao diện. Thay đổi không thành công, kết thúc use case.

- **Luồng rẽ nhánh A4:** Tác nhân không muốn xóa một xen gửi theo tháng trong hệ thống, kích nút **“Hủy”** trên giao diện. Xóa không thành công, kết thúc use case.

- **Luồng rẽ nhánh A5:** Hệ thống kiểm tra ràng buộc dữ liệu xe không được xóa, đưa ra thông báo **“Không thể xóa”**. Use case kết thúc.

- **Luồng rẽ nhánh A7:** Hệ thống không tìm thấy thông tin xe mà tác nhân muốn tìm kiếm thì đưa ra thông báo không tìm thấy.

Hậu điều kiện: Nếu ca sử dụng được thực hiện thành công thì thông tin xe gửi theo tháng sẽ được thêm, sửa, xóa còn lại các trường hợp khác thông tin vẫn được giữ nguyên.

3.2.4.8 Use case: Quản lý xe trong bãi

Tên use case: Quản lý xe trong bãi

Mô tả vắn tắt: Use case này cho phép tác nhân tìm kiếm thông tin xe có trong bãi gửi xe tại thời điểm tìm kiếm.

Tác nhân: Admin, nhân viên

Tiền điều kiện: Tác nhân phải đăng nhập vào hệ thống trước khi thực hiện use case này.

Luồng sự kiện

- **Luồng sự kiện chính:** Use case này bắt đầu khi tác nhân muốn tìm kiếm thông tin về các xe gửi theo tháng có trong hệ thống. Hệ thống sẽ hiển thị danh sách các xe đang gửi trong bãi gửi xe trong hệ thống. Tác nhân có thể kích con trỏ chuột lên trực tiếp các ảnh của xe sau đó phía trên giao diện sẽ hiển thị thông tin chi tiết khi xe vào bãi. Hoặc tác nhân có thể tìm kiếm thông tin chi tiết khi xe vào bãi bằng cách nhập biển số xe của xe cần tìm vào ô nhập thông tin sau đó kích nút “**Tìm kiếm**” (Luồng rẽ nhánh A1).

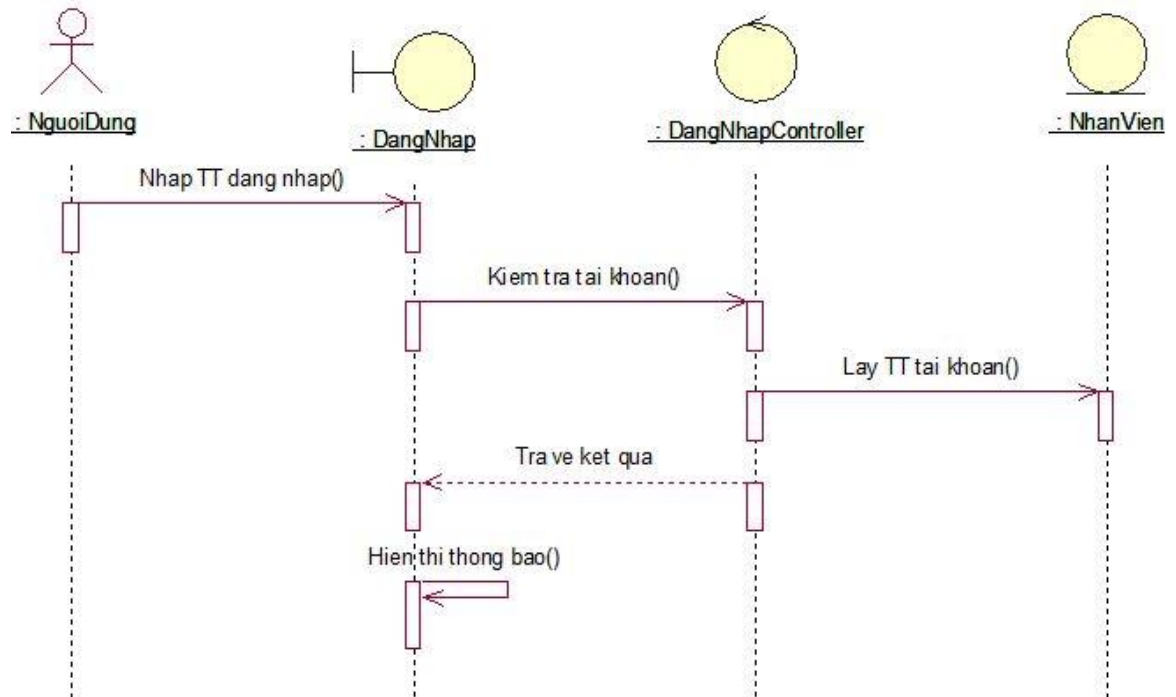
- **Luồng sự kiện rẽ nhánh**

- **Luồng rẽ nhánh A1:** Hệ thống không tìm thấy thông tin của xe mà tác nhân muốn tìm kiếm thì đưa ra thông báo không tìm thấy xe.

Hậu điều kiện: Nếu ca sử dụng được thực hiện thành công thì thông tin của xe sẽ được tìm kiếm còn lại các trường hợp khác thông tin vẫn được giữ nguyên.

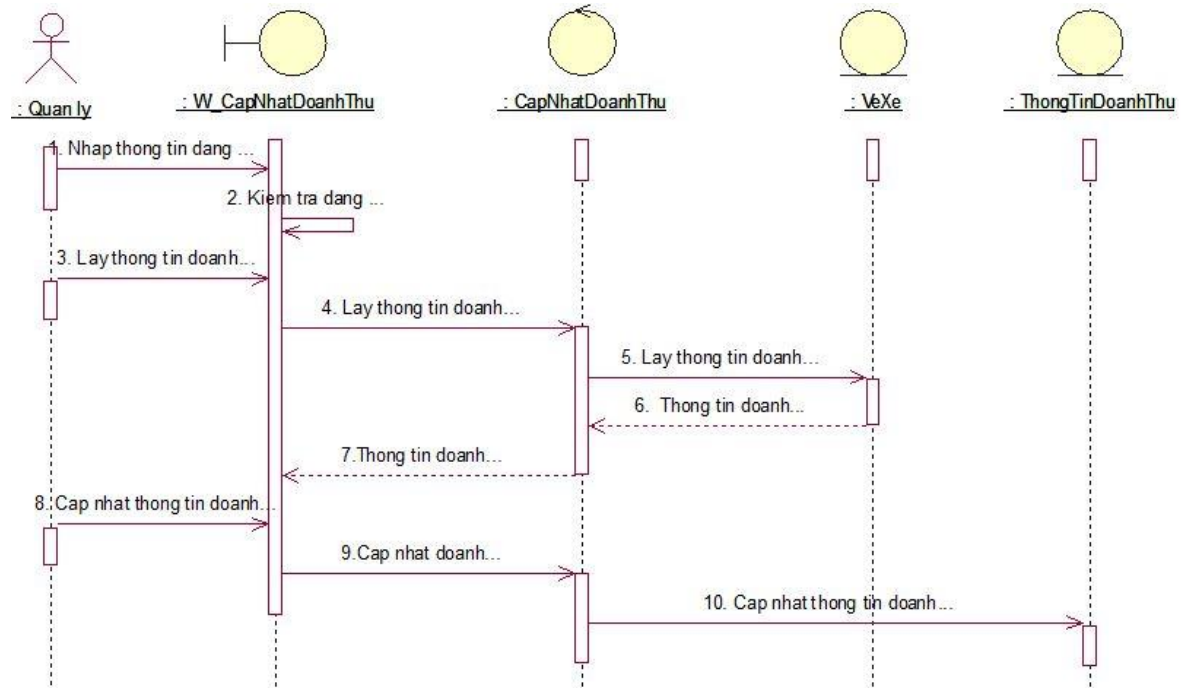
3.2.5 Biểu đồ trình tự

3.2.5.1 Biểu đồ trình tự chức năng đăng nhập



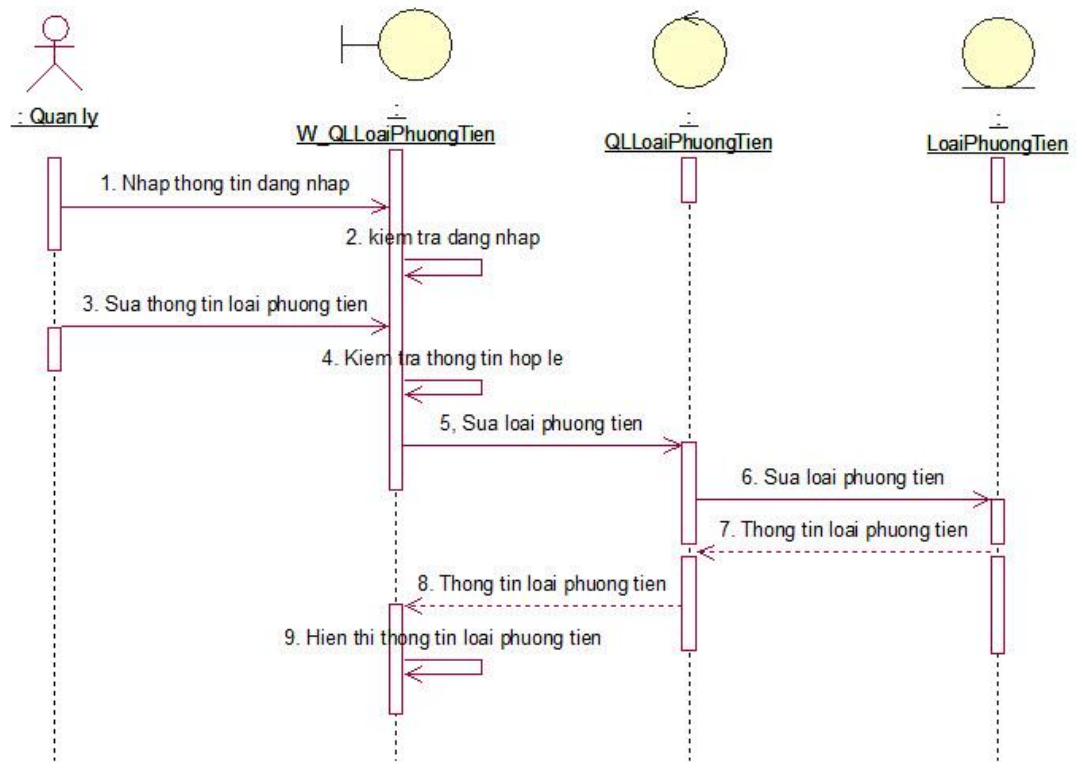
Hình 2.3. 12 Biểu đồ trình tự đăng nhập

3.2.5.2 Biểu đồ trình tự chức năng thống kê doanh thu

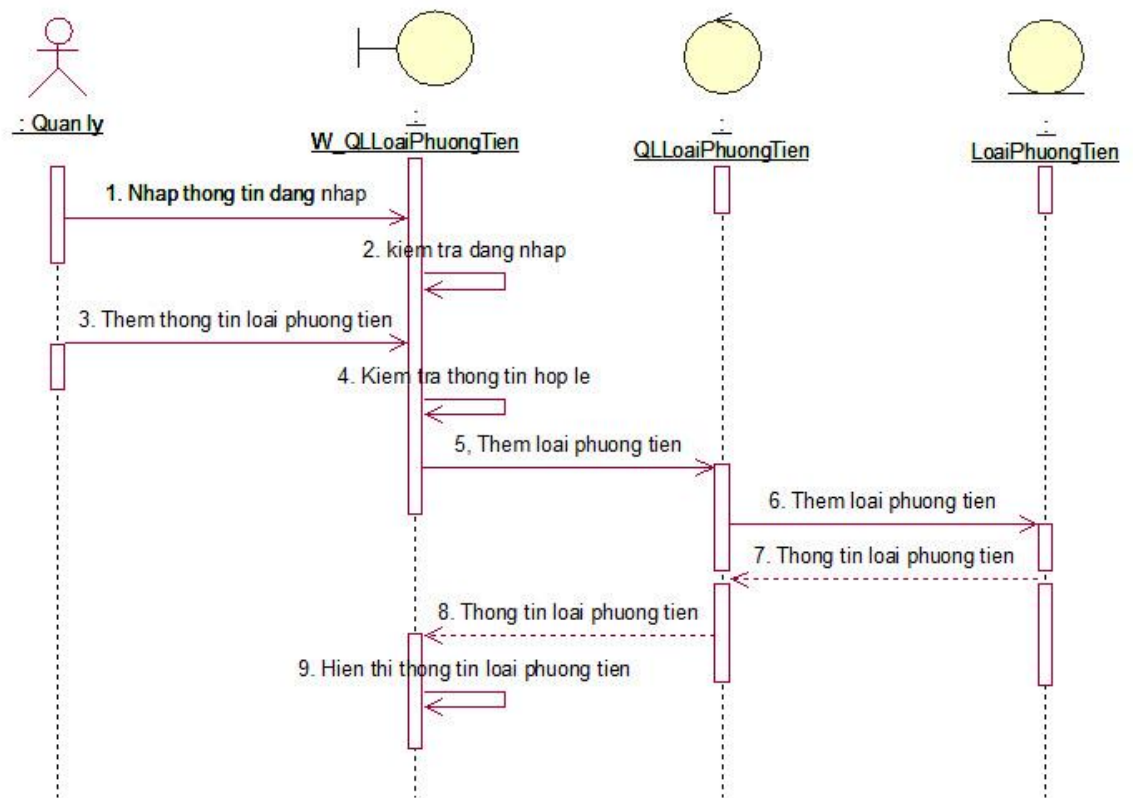


Hình 2.3. 13 Biểu đồ trình tự thống kê doanh thu doanh thu

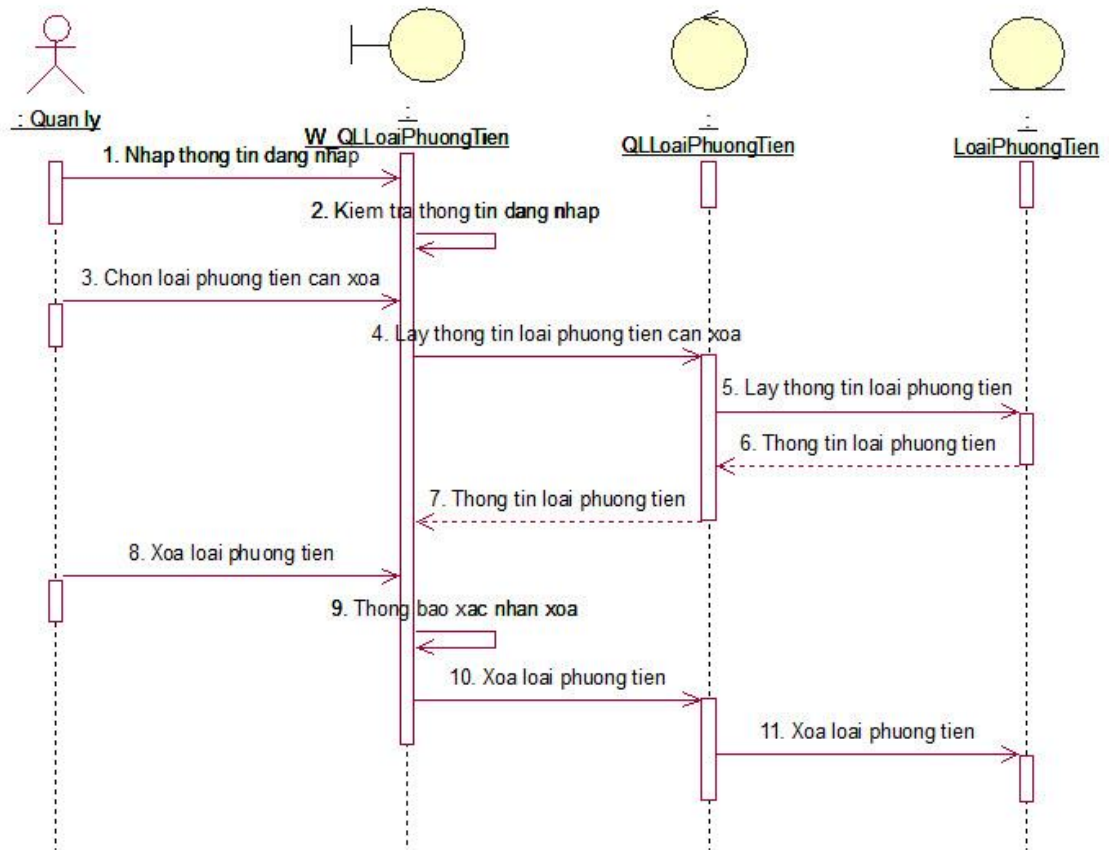
3.2.5.3 Biểu đồ trình tự chức năng quản lý loại xe



Hình 2.3. 14 Biểu đồ trình tự sửa loại xe

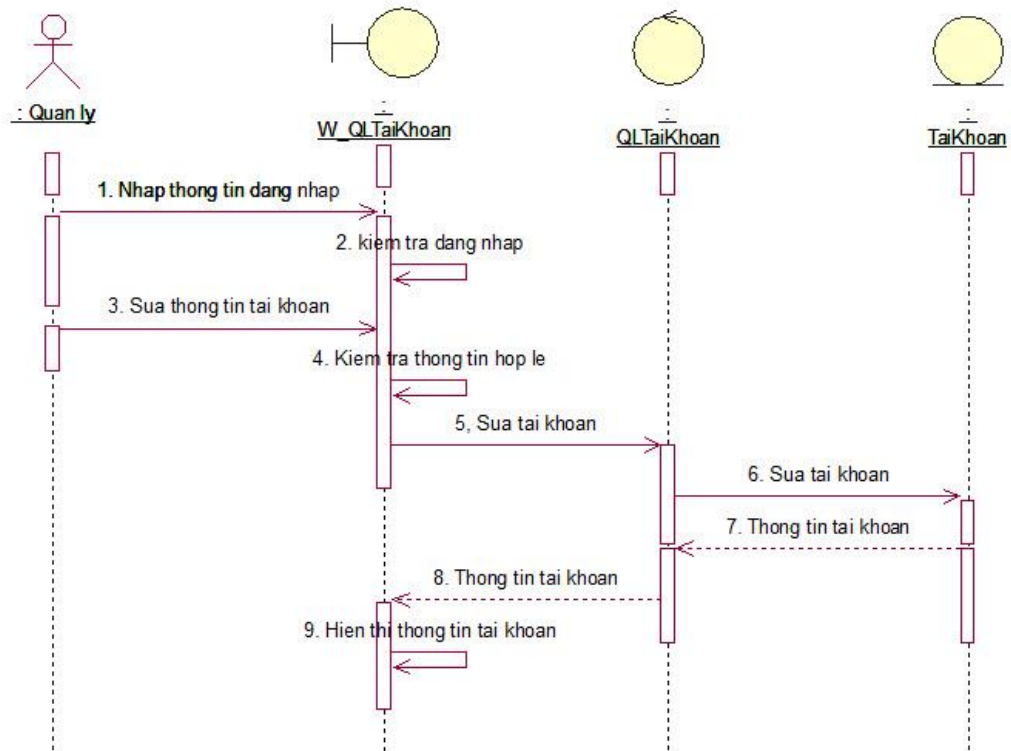


Hình 2.3. 15 Biểu đồ trình tự thêm loại phương tiện

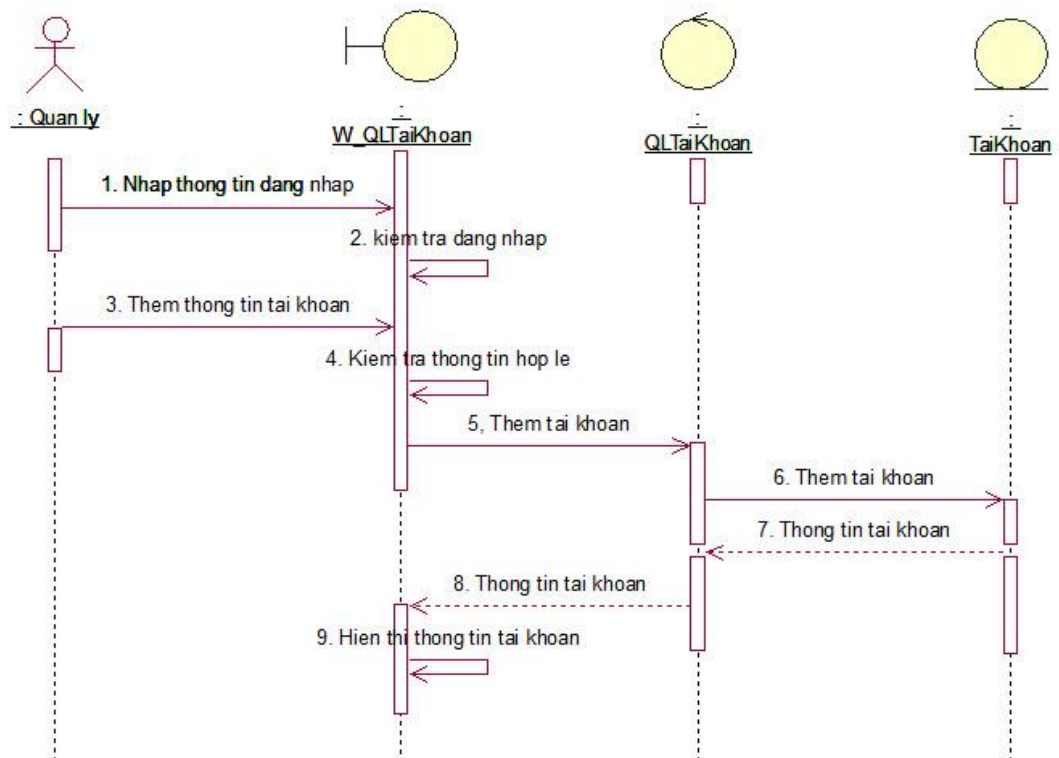


Hình 2.3. 16 Biểu đồ trình tự xóa loại phương tiện

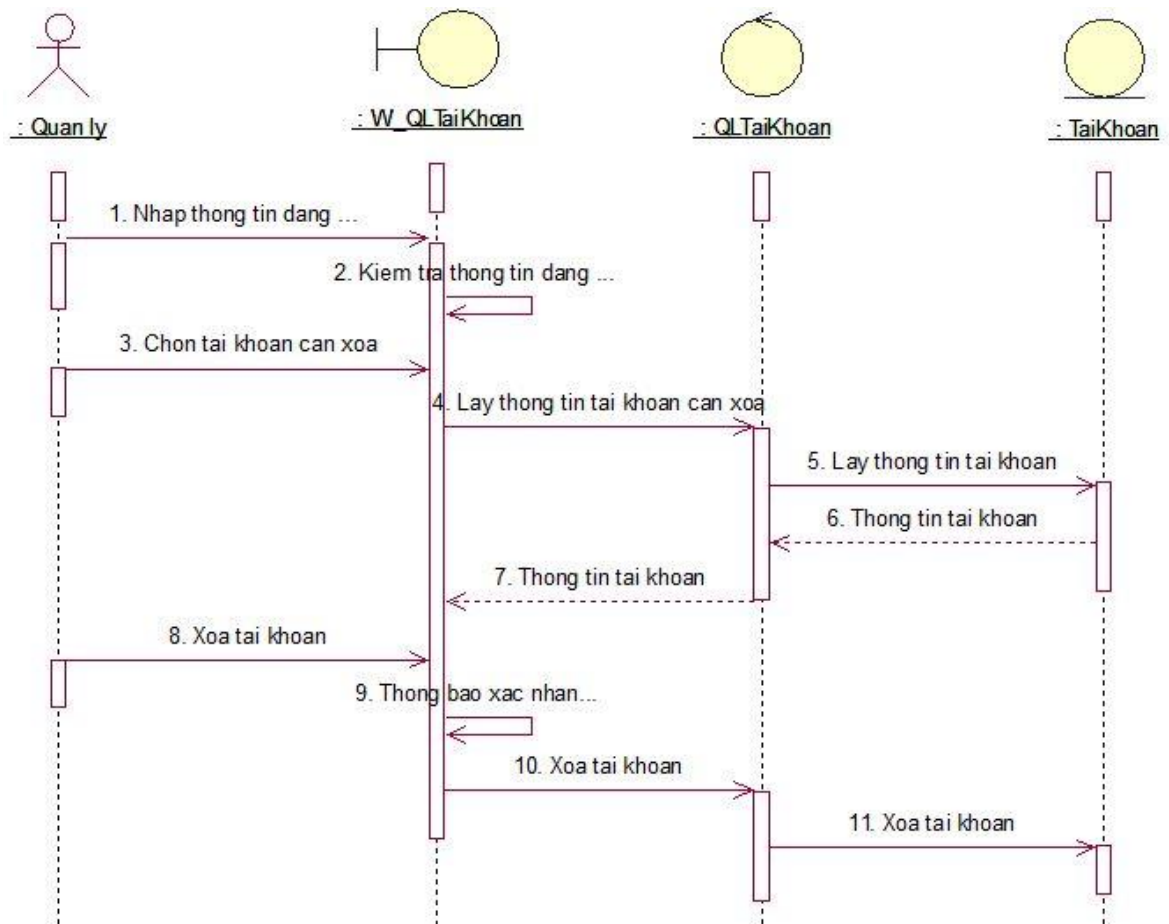
3.2.5.5 Biểu đồ trình tự chức năng quản lý tài khoản



Hình 2.3. 17 Biểu đồ trình tự sửa tài khoản

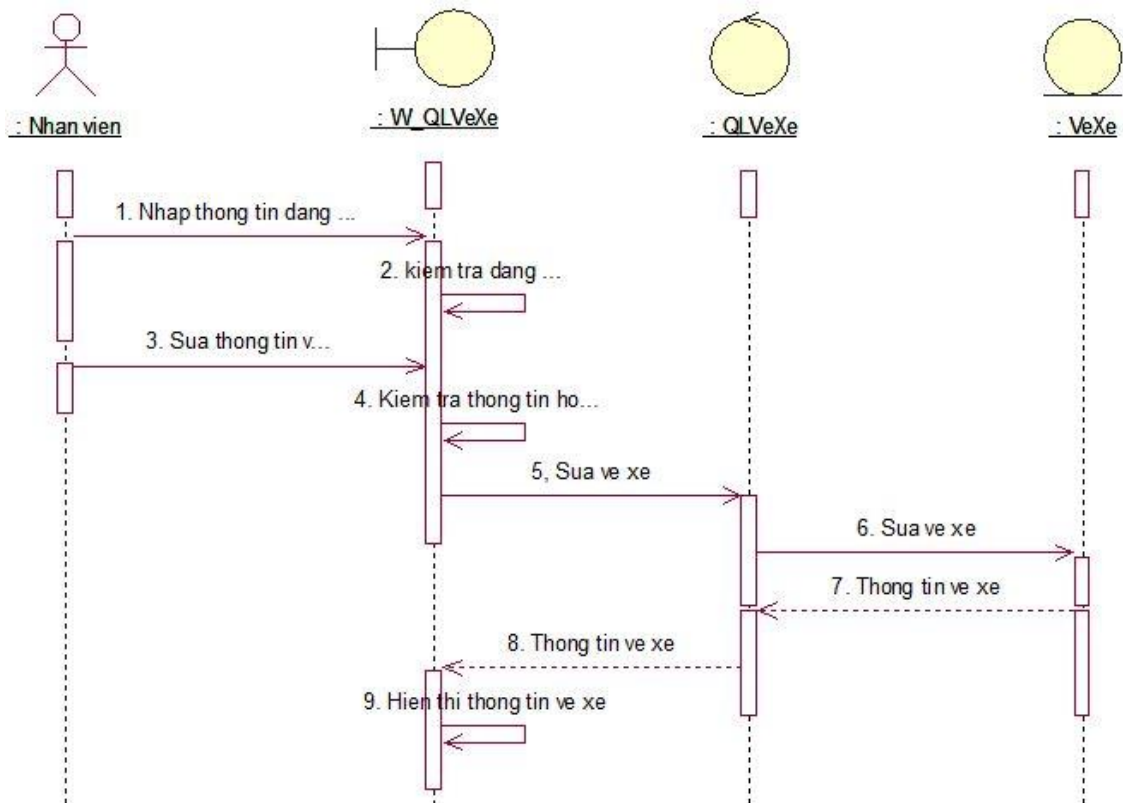


Hình 2.3. 18 Biểu đồ trình tự thêm tài khoản

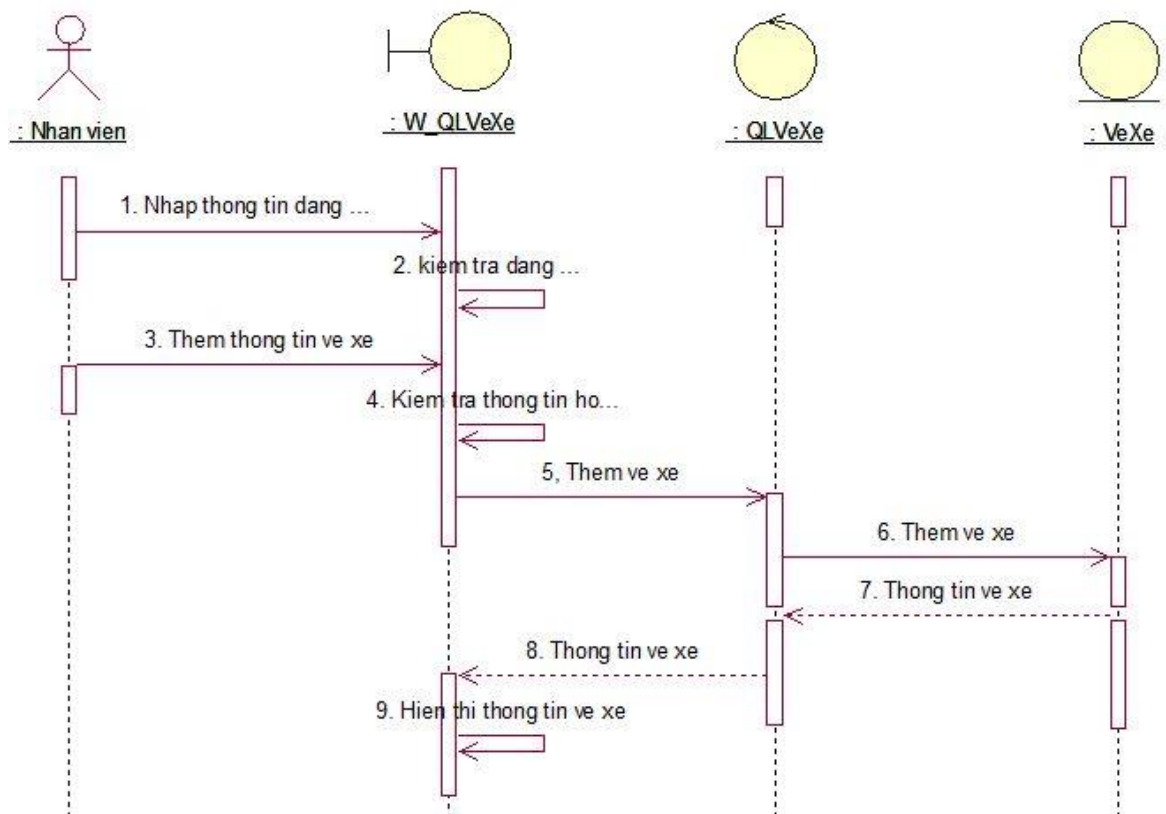


Hình 2.3. 19 Biểu đồ trình tự xóa tài khoản

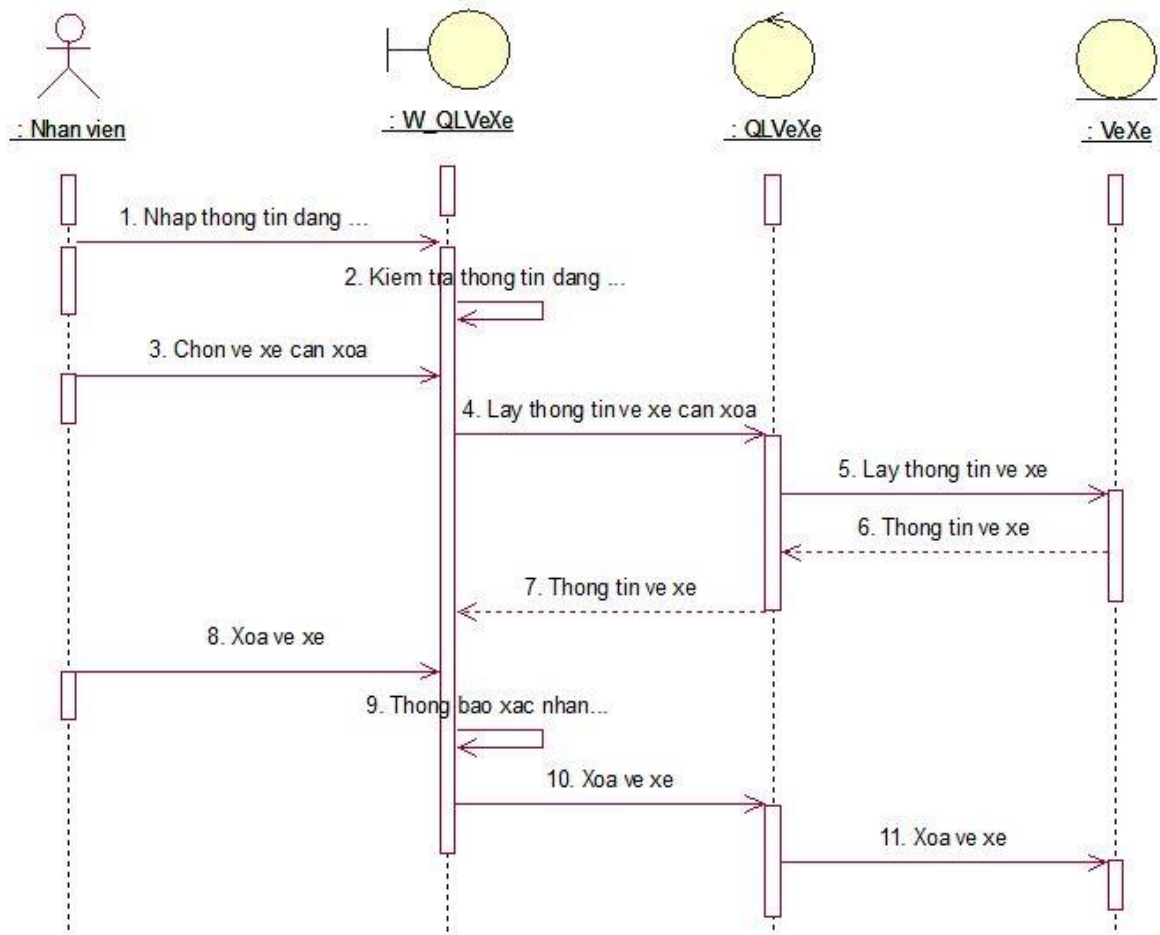
3.2.5.6 Biểu đồ trình tự chức năng quản lý thẻ gửi xe



Hình 2.3. 20 Biểu đồ trình tự sửa vé xe

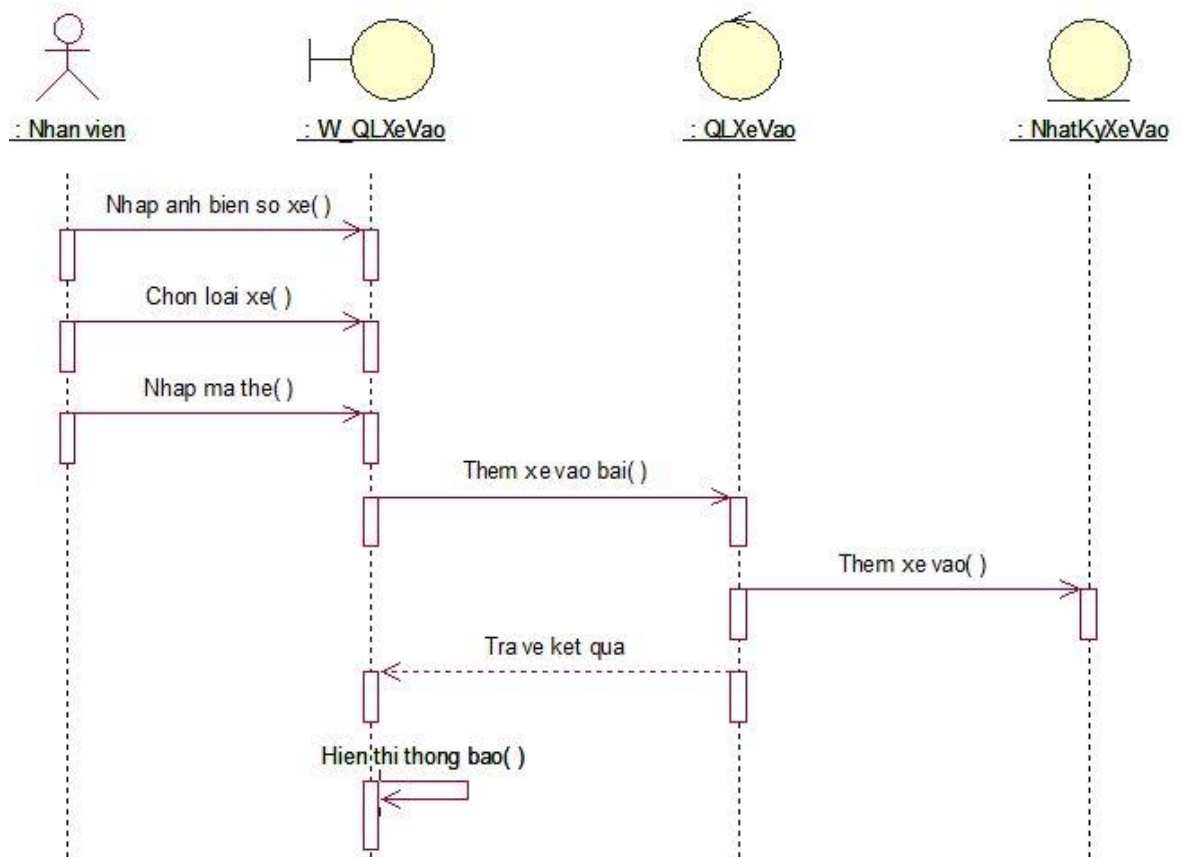


Hình 2.3. 21 Biểu đồ trình tự thêm vé xe



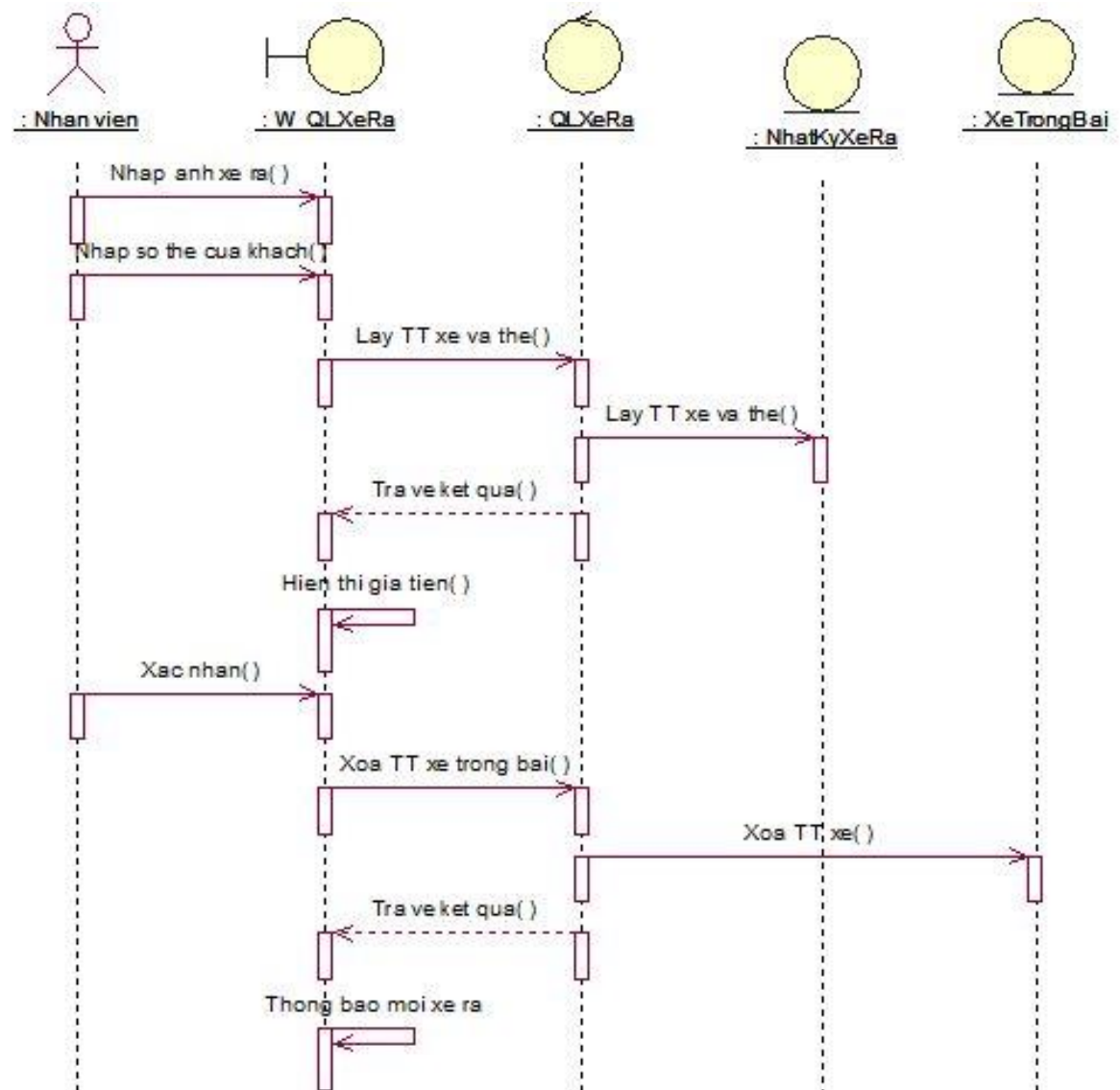
Hình 2.3. 22 Biểu đồ trình tự xóa vé xe

3.2.5.7 Biểu đồ trình tự chức năng quản lý xe vào



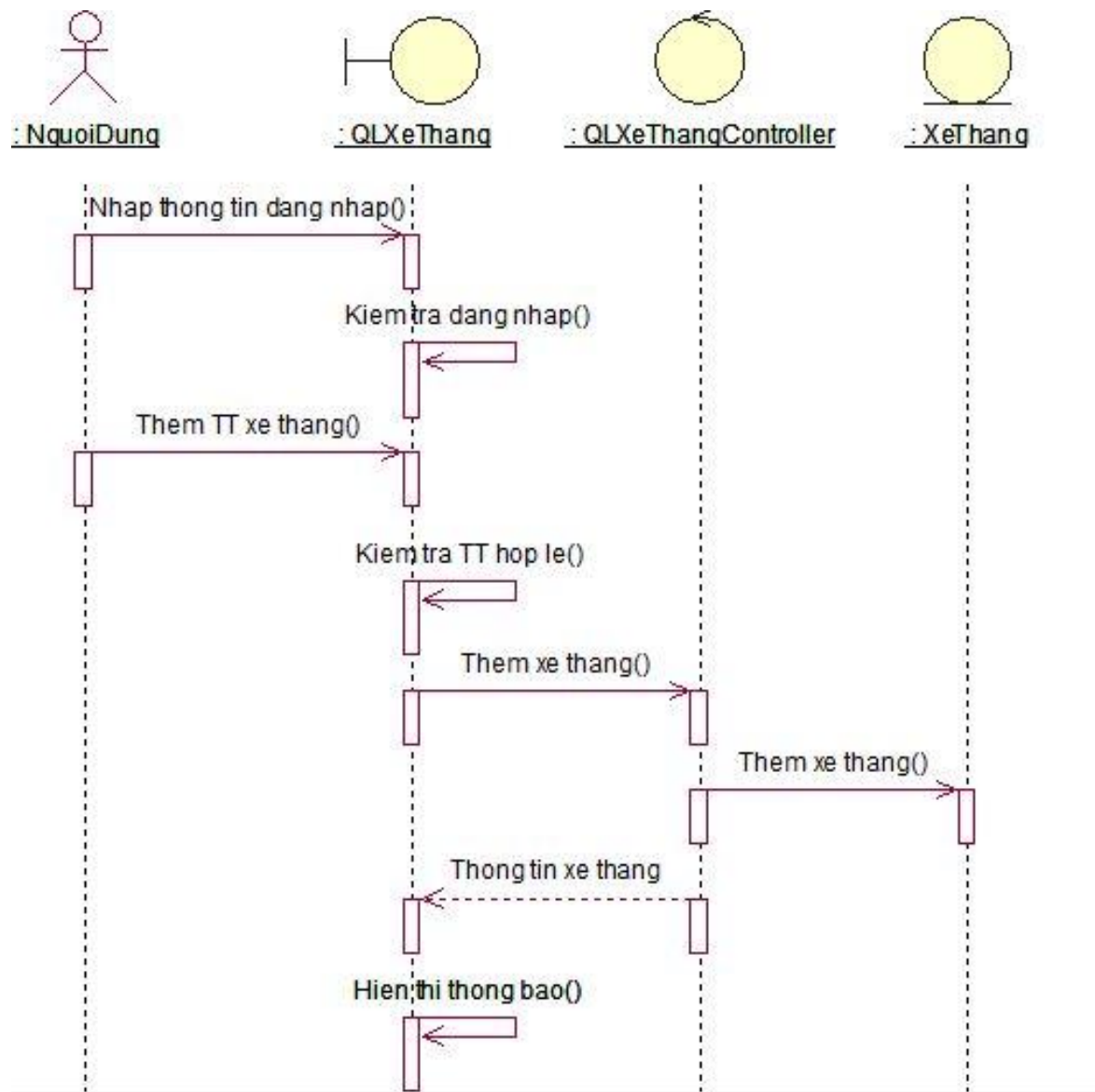
Hình 2.3. 23 Biểu đồ trình tự thêm xe vào bãi

3.2.5.8 Biểu đồ trình tự chức năng quản lý xe ra

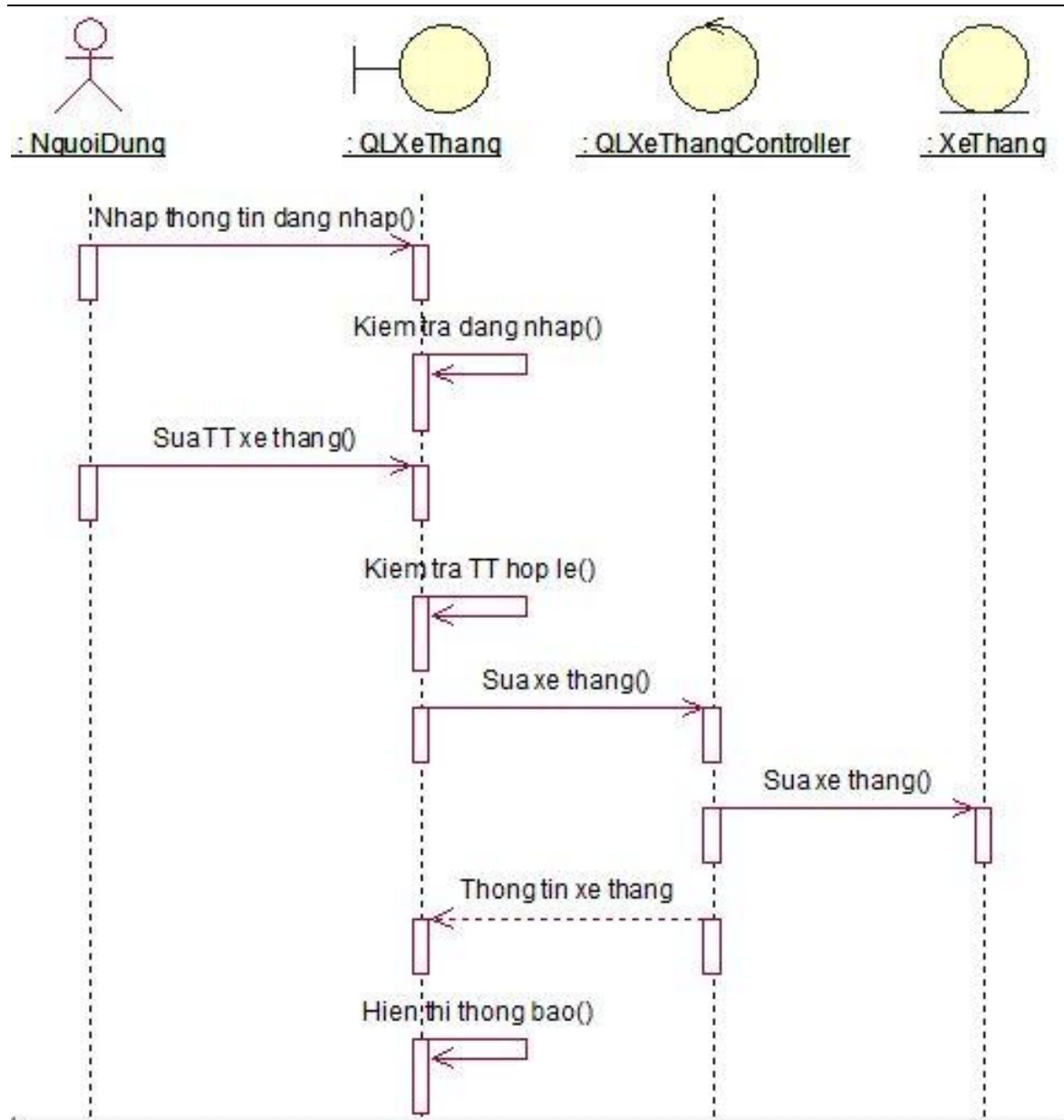


Hình 2.3. 24 Biểu đồ xe ra khỏi bãi

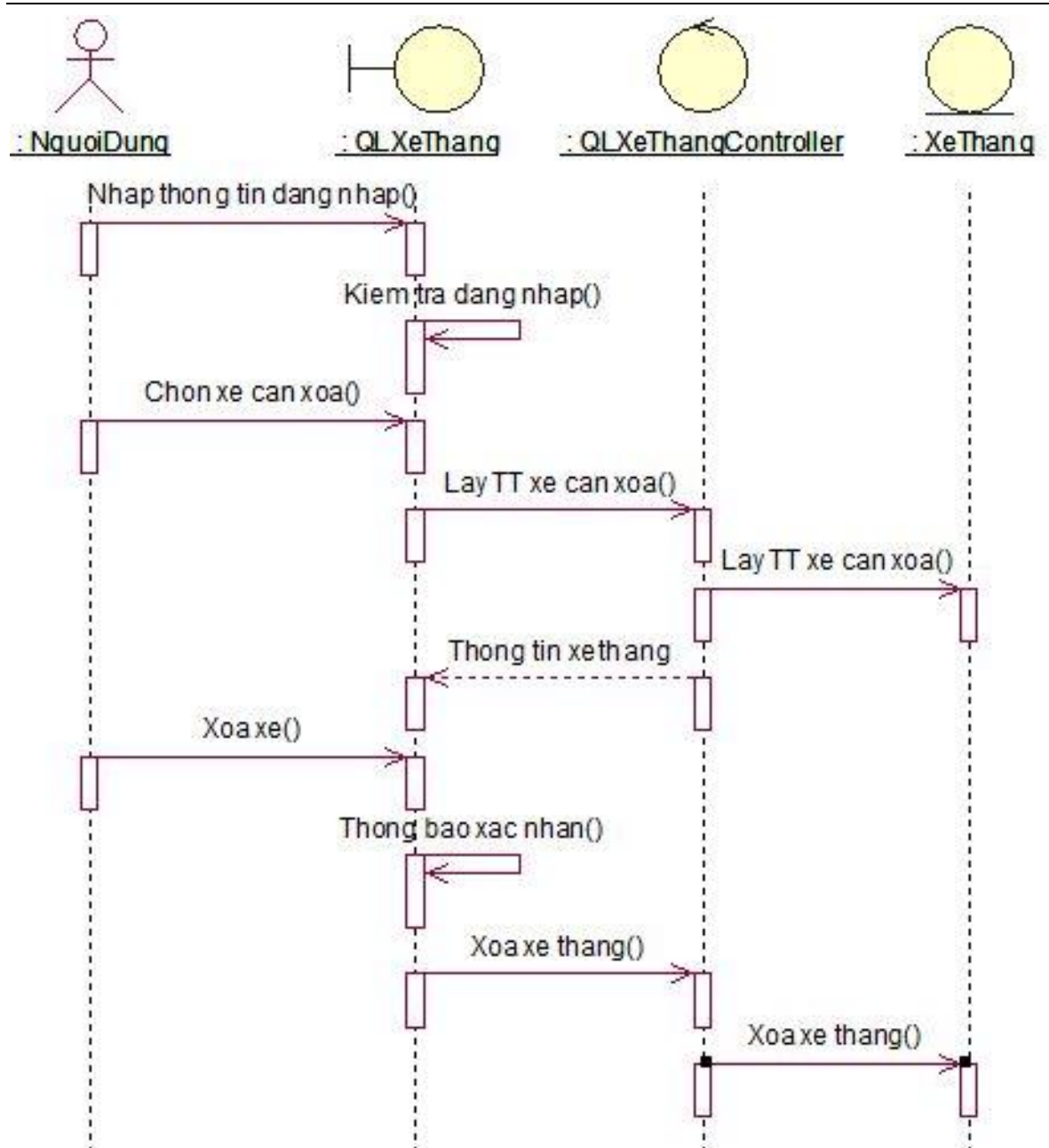
3.2.5.9 Biểu đồ trình tự chức năng quản lý xe tháng



Hình 2.3. 25 Biểu đồ thêm xe gửi tháng



Hình 2.3. 26 Biểu đồ sửa xe gửi thàng



Hình 2.3. 27 Biểu đồ xóa xe gửi thàng

3.4 Thiết kế cơ sở dữ liệu

3.4.1 Xác định các thực thể

3.4.1.1 Bảng chứa thông tin nhân viên

Thuộc tính	Kiểu dữ liệu	Mô tả	Null
ID	Nvarchar(50)	Khóa chính	Not
MatKhau	Nvarchar(MAX)		Not
HoTen	Nvarchar(MAX)		Not
SĐT	Nvarchar(MAX)		
Email	Nvarchar(MAX)		
NgaySinh	Date		
DiaChi	Nvarchar(MAX)		

Bảng 2.3. 3 Bảng nhân viên

3.4.1.2 Bảng chứa thông tin hóa đơn thanh toán

Thuộc tính	Kiểu dữ liệu	Mô tả	Null
IDHoaDon	Int	Khóa chính	Not
ThoiGian	DateTime		Not
TongTien	Int		Not
BienSoXe	Varxhar(MAX)		Not
IDLoaiXe	Int	Khóa ngoại	Not
IDNhanVien	Nvarchar(MAX)	Khóa ngoại	Not
IDTheGuiXe	Int	Khóa ngoại	Not
Mô tả	Nvarchar(MAX)		

Bảng 2.3. 4 Bảng hóa đơn

3.4.1.3 Bảng chứa thông tin xe ra

Thuộc tính	Kiểu dữ liệu	Mô tả	Null
IDNhatKyXeRa	Int	Khóa chính	Not
ThoiGian	Int		
BienSoXe	Int		Not
IDLoaiXe	Int	Khóa ngoại	Not
IDTheGuiXe	Int	Khóa ngoại	Not
IDNhanVien	Nvarchar(MAX)	Khóa ngoại	Not
UrlAnh	Nvarchar(MAX)		

Bảng 2.3. 5 Bảng nhật ký xe ra

3.4.1.4 Bảng chứa thông tin xe vào

Thuộc tính	Kiểu dữ liệu	Mô tả	Null
IDNhatKyXeVao	Int	Khóa chính	Not
ThoiGian	Int		
BienSoXe	Int		Not
IDLoaiXe	Int	Khóa ngoại	Not
IDTheGuiXe	Int	Khóa ngoại	Not
IDNhanVien	Nvarchar(MAX)	Khóa ngoại	Not
UrlAnh	Nvarchar(MAX)		

Bảng 2.3. 6 Bảng nhật ký xe vào

3.4.1.5 Bảng chứa thông tin của loại xe

Thông tin	Thuộc tính	Mô tả	Null
IDLoaiXe	Int		Not
TenLoaiXe	Nvarchar(MAX)		Not
Mô tả	Nvarchar(MAX)		
DonGia	Int		Not

Bảng 2.3. 7 Bảng loại xe

3.4.1.6 Bảng chứa thông tin của thẻ gửi xe

Thông tin	Thuộc tính	Mô tả	Null
IDLoaiThe	Int	Khóa chính	Not
TenLoaiThe	Nvarchar(MAX)		Not
Mô tả	Nvarchar(MAX)		

Bảng 2.3. 8 Bảng thẻ gửi xe

3.4.1.7 Bảng chứa thông tin của xe đang trong bãi

Thông tin	Thuộc tính	Mô tả	Null
IDXeTrongBai	Nvarchar(50)	Khóa chính	Not
ThoiGianVao	DateTime		Not
ThoiGianRa	DateTime		Not
IDTheGuiXe	Int	Khóa ngoại	Not
IDLoaiXe	Int	Khóa ngoại	Not
UrlAnh	Nvarchar(MAX)		Not

Bảng 2.3. 9 Bảng xe trong bãi

3.4.1.8 Bảng chứa thông tin của xe gửi theo tháng

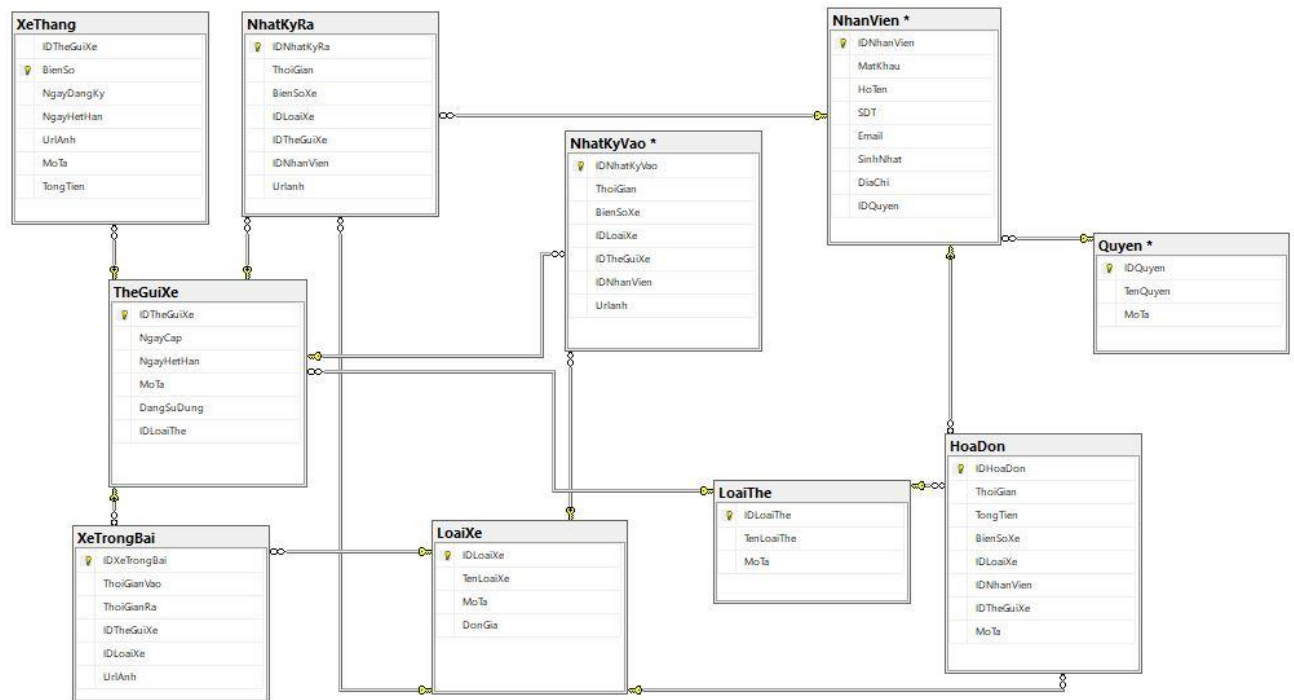
Thông tin	Thuộc tính	Mô tả	Null
IDXeTrongBai	Nvarchar(50)	Khóa chính	Not
ThoiGianVao	DateTime		Not
ThoiGianRa	DateTime		Not
IDTheGuiXe	Int	Khóa ngoại	Not
IDLoaiXe	Int	Khóa ngoại	Not
UrlAnh	Nvarchar(MAX)		Not

Bảng 2.3. 10 Bảng xe tháng**3.4.1.9 Bảng phân quyền**

Thông tin	Thuộc tính	Mô tả	Null
IDQuyền	Nvarchar(50)	Khóa chính	Not
TenQuyền	DateTime		Not
MoTa	Nvarchar(MAX)		Not

Bảng 2.3. 11 Bảng phân quyền

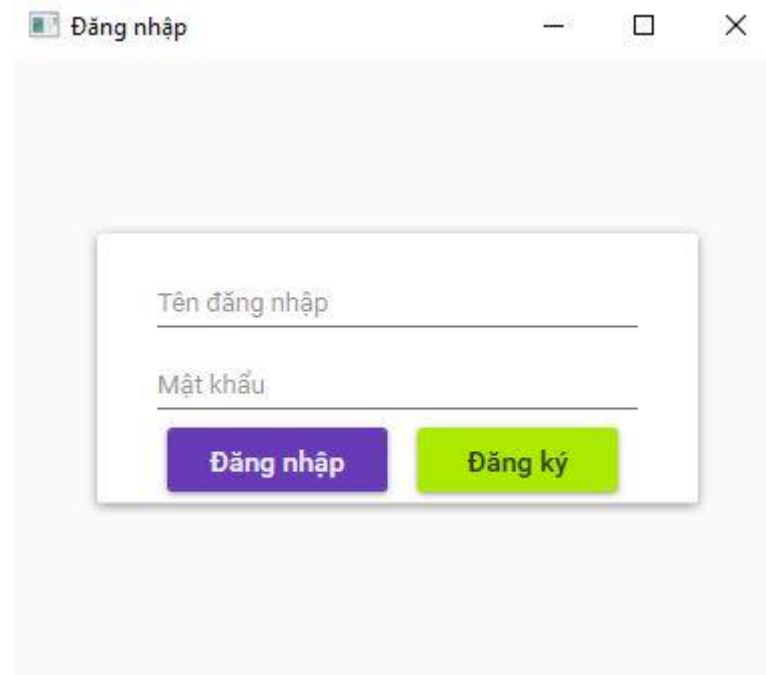
3.4.2 Mô hình quan hệ



Hình 2.3. 28 Biểu đồ mô hình cơ sở dữ liệu vật lý

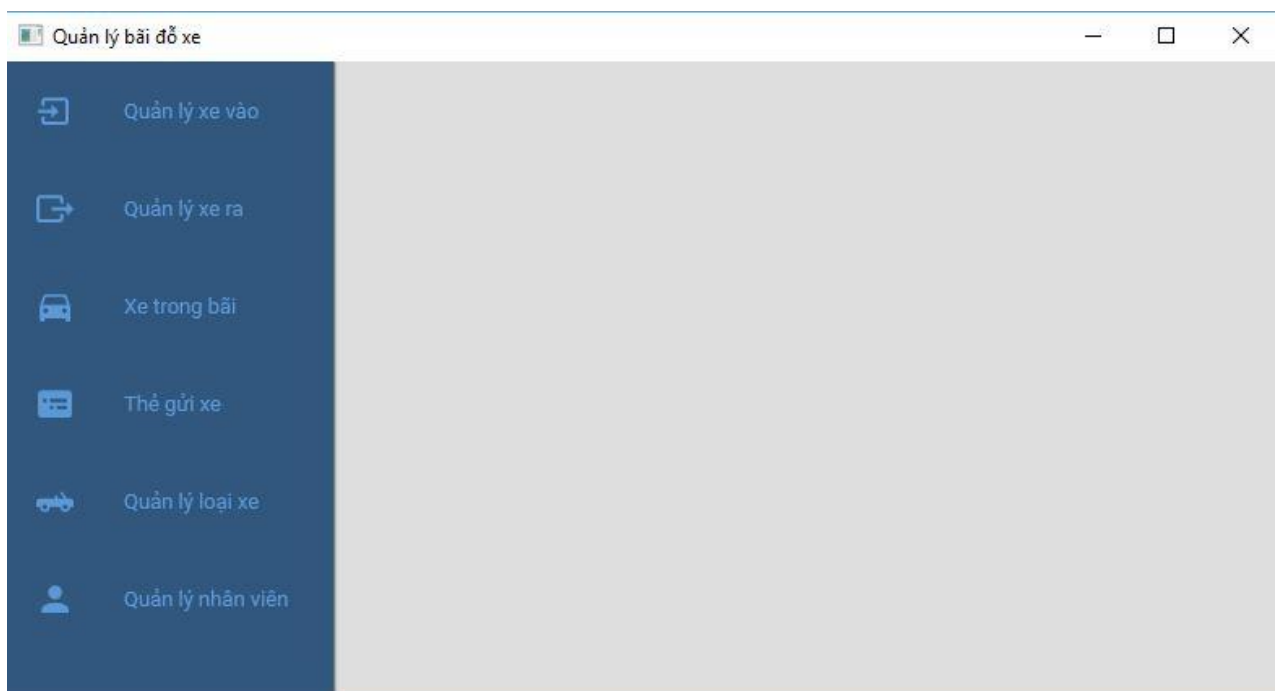
3.5 Thiết kế giao diện

3.5.1 Giao diện đăng nhập



Hình 2.3. 29 Giao diện chức năng đăng nhập

3.5.2 Giao diện màn hình chính



Hình 2.3. 30 Giao diện màn hình chính

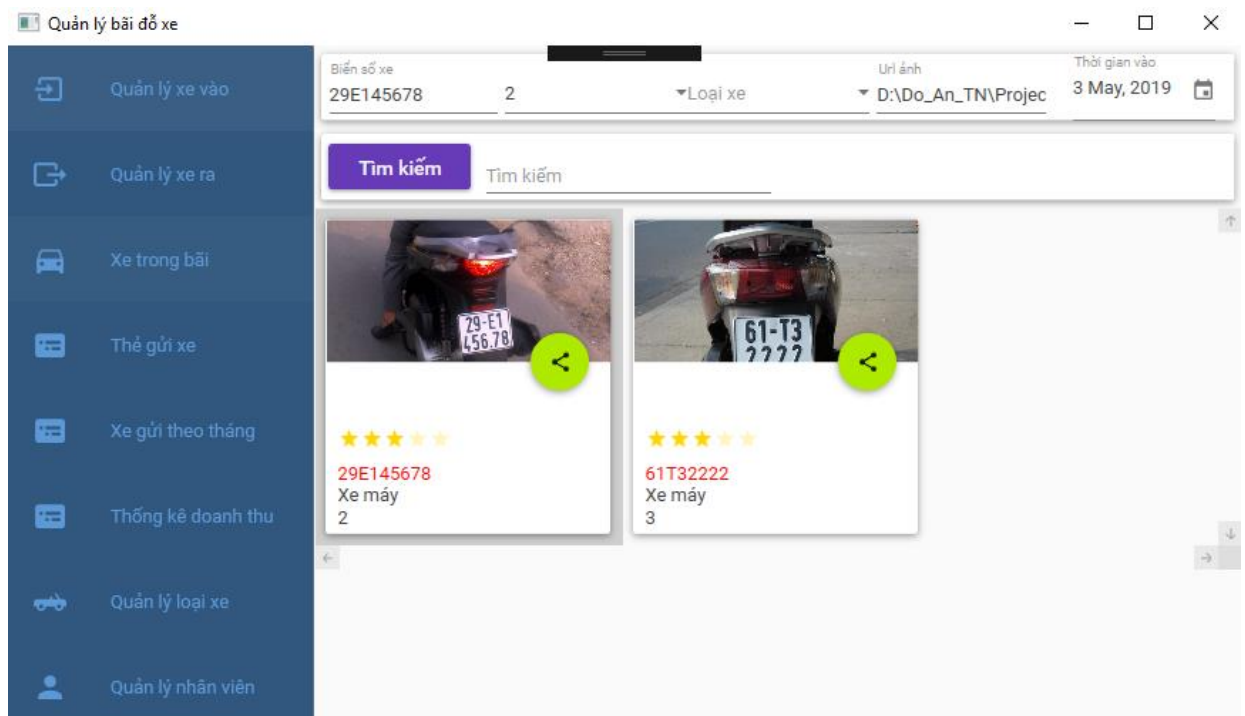
3.5.3 Giao diện quản lý xe vào

Hình 2.3. 31 Giao diện chức năng quản lý xe vào

3.5.4 Giao diện quản lý xe ra

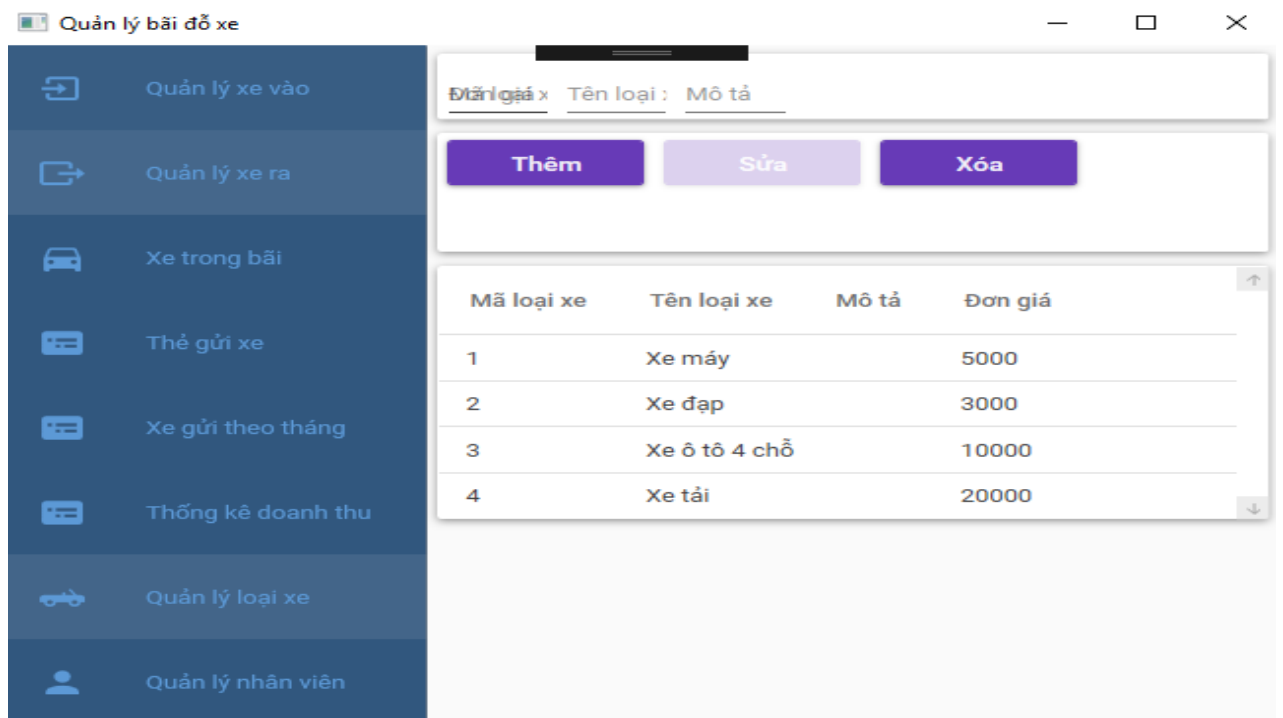
Hình 2.3. 32 Giao diện chức năng quản lý xe ra

3.5.5 Giao diện quản lý xe trong bãi



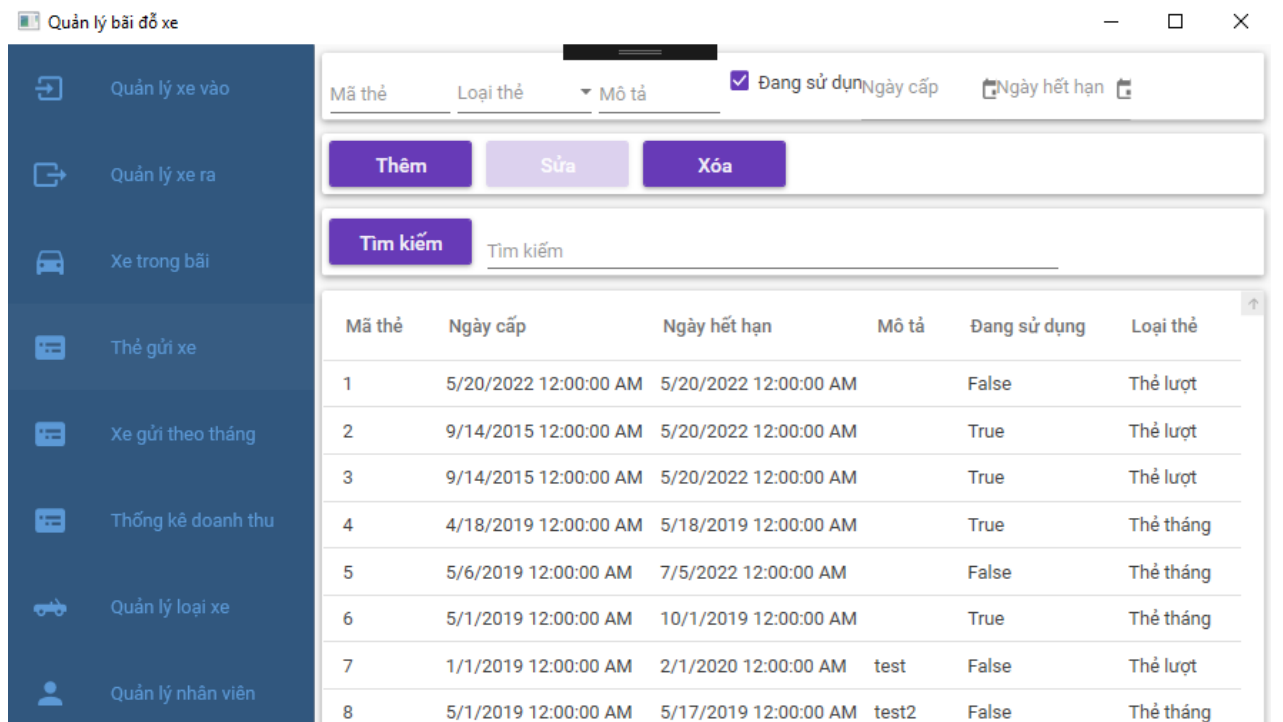
Hình 2.3. 33 Giao diện chức năng thông tin xe trong bãi

3.5.6 Giao diện quản lý loại xe



Hình 2.3. 34 Giao diện quản lý loại xe

3.5.7 Giao diện quản lý thẻ gửi xe



Hình 2.3. 35 Quản lý thẻ gửi xe

3.5.8 Giao diện quản lý xe gửi theo tháng

Quản lý bãi đỗ xe

Quản lý xe vào | Quản lý xe ra | Xe trong bãi | **Thẻ gửi xe** | Xe gửi theo tháng | Thống kê doanh thu | Quản lý loại xe | Quản lý nhân viên

Mã thẻ tháng | Biến số xe | Ngày đăng ký | Ngày hết hạn | Mô tả | Url ảnh | Tổng tiền

Thêm **Sửa** **Xóa**

Tim kiếm Tim kiếm

Mã thẻ	Biến số xe	Ngày đăng ký	Ngày hết hạn	Mô tả	Url ảnh	Tổng tiền
4	15C112345	5/1/2019 12:00:00 AM	6/1/2019 12:00:00 AM	test	D:\Do_An_TN\Project\Image\images (1).jpg	250000

Hình 2.3. 36 Quản lý xe tháng

3.5.9 Giao diện quản lý nhân viên

Quản lý bãi đỗ xe

Quản lý xe vào | Quản lý xe ra | Xe trong bãi | Thẻ gửi xe | Xe gửi theo tháng | Thống kê doanh thu | Quản lý loại xe | **Quản lý nhân viên**

Mã nhân viên | Họ và tên | Số điện thoại | Email | Địa chỉ | Mật khẩu | Sinh nhật | Quyền

Thêm **Sửa** **Xóa**

Tim kiếm Tim kiếm

Mã nhân viên	Họ và tên	Số điện thoại	Email	Địa chỉ	Mật khẩu	Sinh nhật	Quyền
admin	Admin	00000000	admin@gmail.com		19951995	1/1/1997 12:00:00 AM	
admin2	Bùi Thu Hằng	0384399862	aaa@gmail.com	Trăng Hưng Đạo	123456		
lehaisonmath6	Lê Hải Sơn	03812312321	lehaisonmath6@gmail.com	Hà Nội	123456		

Hình 2.3. 37 Giao diện quản lý nhân viên

3.5.10 Giao diện thống kê doanh thu



Hình 2.3. 38 Giao diện thống kê doanh thu

KẾT LUẬN

Phần mềm quản lý bãi gửi xe được xây dựng trên ngôn ngữ lập trình kết hợp nền tảng WPF sử dụng SQL server 2017. Phần mềm đã đưa ra nhiều chức năng giúp cho nhân viên trông xe và người quản lý có thể quản lý bãi gửi xe một cách dễ dàng hơn nhờ các chức năng cơ bản như sau:

- Nhân viên trông xe:
 - Quản lý xe vào gửi và xe lấy ra.
 - Quản lý việc thu phí với xe gửi theo lượt.
 - Quản lý thẻ xe.
 - Quản lý loại xe.
 - Quản lý xe tháng.
 - Quản lý xe đang trong bãi.
- Người quản trị:
 - Quản lý xe vào gửi và xe lấy ra.
 - Quản lý thẻ xe.
 - Quản lý loại xe.
 - Quản lý xe tháng.
 - Quản lý xe đang trong bãi.
 - Quản lý tài khoản nhân viên trong hệ thống
 - Cập nhật doanh thu theo ngày, tháng.

Bên cạnh những lợi ích thì phần mềm vẫn chưa thể đáp ứng đủ các chức năng trong một hệ thống quản lý bãi gửi xe hoàn chỉnh. Việc sử dụng WPF đã phần nào mang lại sự thân thiện hơn cho người sử dụng và cũng góp phần làm tăng hiệu quả công việc.

Trong tương lai, sản phẩm hướng tới là phù hợp với hầu hết tất cả các loại xe và ở tất cả những nơi có lượng xe gửi cao. Nâng cao tốc độ dựa trên xử lý trong code và database.

Tài liệu tham khảo:

- <https://www.howkteam.vn/>
 - <https://phamtuantech.com/wpf-la-gi/>
 - <https://www.stdio.vn/articles/gioi-thieu-thu-vien-emgucv-va-cach-tich-hop-vao-net-362>
-