

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN TỐT NGHIỆP

**Tìm hiểu công nghệ Phonegap &
Google Maps API áp dụng xây dựng
ứng dụng hỗ trợ dịch vụ xe khách trên
thiết bị di động**

Giáo viên hướng dẫn: **Nguyễn Mạnh Sơn**
Sinh viên thực hiện: **Lê Văn Sơn**
SHSV : **B12LDCN070**
Lớp : **L12 – CN2**

MỤC LỤC

MỤC LỤC.....	2
LỜI NÓI ĐẦU	5
CHƯƠNG 1. TÌM HIỂU CÔNG NGHỆ PHONEGAP VÀ CÁC VĂN ĐỀ LIÊN QUAN	11
1.1 Phonegap.....	11
1.1.1 Vì sao lại lựa chọn PhoneGap	11
1.1.2 Tìm hiểu về Apache Cordova.....	13
1.2 WebService	14
1.2.1 Khái niệm	14
1.2.2 Một số đặc điểm cơ bản	14
1.2.3 Ưu – nhược điểm	15
1.2.4 Hoạt động của web service	16
1.2.5 JSON	16
1.3 Google Maps API	17
1.3.1 Google map	17
1.3.2 Google Maps API Web Services	18
1.3.3 Google Geocoding API.....	19
1.3.4 Google Distance Matrix API.....	20
1.3.5 Google Direction API	21
1.4 Tổng quan Android	22
1.4.1 Android là gì? Tại sao cần quan tâm tới nền tảng Android	22

1.4.2 Kiến trúc Android	24
CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG	25
2.1 Khảo sát yêu cầu hệ thống	25
2.1.1 Khảo sát sơ bộ	25
2.1.2 Khảo sát chi tiết	26
2.1.3 Xây dựng dự án	31
2.2 Phân tích hệ thống.....	33
2.2.1 Biểu đồ Use Case.....	33
2.2.2 Biểu đồ lớp	40
2.2.3 Biểu đồ trạng thái (State diagram).....	41
2.3 Thiết kế hệ thống	43
2.3.1 Biểu đồ tuần tự (sequence diagram)	43
2.3.2 Biểu đồ cộng tác (colaboration Diagram)	45
2.3.3 Biểu đồ lớp chi tiết.....	46
2.3.4 Thiết kế chi tiết.....	47
2.3.5 Biểu đồ triển khai	51
CHƯƠNG 3. CÀI ĐẶT VÀ TRIỂN KHAI	52
3.1 Các vấn đề kỹ thuật.....	52
3.2 Một số hình ảnh sau khi cài đặt :.....	53
3.2.1 Lái xe.....	53
3.2.2 Khách hàng.....	57
3.2.3 Server	63
CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	65

LỜI NÓI ĐẦU

Ngày nay, dịch vụ xe khách khắp mọi nơi trên đất nước đã và đang rất phát triển. Ngày càng nhiều các nhà cung cấp dịch vụ mở ra. Khách hàng chỉ cần tới bến xe, trạm dừng đỗ hoặc chỉ là ra bên lề đường vẫy một chiếc xe là có thể được phục vụ.

Qua xem xét thực trạng xe khách tại địa điểm khảo sát là dọc tuyến QL 1A từ thủ đô Hà Nội tới Thanh Hóa mà cụ thể là bến xe Mỹ Đình, Giáp Bát ... em thấy sự cung cấp dịch vụ xe khách là tương đối tốt, đáp ứng nhu cầu khách hàng. Nhưng bên cạnh đó tồn tại nhiều vấn đề như: khách hàng phải nhớ hoặc phải có cardvisit của hãng thì mới biết thông tin xe chạy, lộ trình... nếu không muốn mất thời gian tại bến xe hoặc ra nơi chờ. Bên cạnh đó thông tin không được trực quan, khi ra địa điểm bắt xe thì không biết có những xe khách nào gần vị trí của mình, nhiều khi khách hàng phải chờ rất lâu mà không biết chiếc xe cần hiện đang ở đâu, thông tin xe cũng như tài xế, hay làm sao bắt xe thuận tiện nhất, tốn ít chi phí nhất, hay có những vấn đề phát sinh như quên đồ, phàn nàn chất lượng dịch vụ thì khách hàng cũng rất khó biết thông tin để liên lạc lại .v.v.

Bên cạnh đó hiện nay, với sự phát triển của công nghệ mobile ngày càng mạnh mẽ thì những chiếc smartphone từ bình dân tới cao cấp trở nên rất phổ biến. Cùng với đó là hạ tầng mạng cũng dần hoàn thiện và hiện tại là 3G đã phủ sóng trên hầu khắp đất nước. Mặc dù các nền tảng di động là rất đa dạng, khó cho việc chuyển đổi khi ứng dụng nhưng nắm bắt được nhu cầu kết nối các nền tảng lại với nhau từ cuối năm 2012 bộ công cụ phát triển đa nền Phonegap (“viết một lần chạy khắp nơi”) đã chính thức ra đời.

Nhận thấy những vấn đề trên, cũng như tính ưu việt khi áp dụng khoa học công nghệ thông tin mà cụ thể là trên thiết bị di động. Em quyết định chọn đề tài “**Tìm hiểu công nghệ PhoneGap và Google Maps API áp dụng xây dựng hệ thống hỗ trợ dịch vụ xe khách trên thiết bị di động**”. Sự ra đời của hệ thống sẽ khắc phục những bất cập của hệ thống dịch vụ hiện nay. Dù hệ thống được xây dựng trên bộ công cụ phát triển đa nền Phonegap. Nhưng do phạm vi thời gian có hạn em xin phép chỉ trình bày hệ thống trên nền tảng Android.

Dù đã cố gắng nhưng sẽ không tránh khỏi những sai sót, em rất mong được sự gợi ý, chỉ bảo của thầy để hệ thống hoàn thiện và mang tính thực tiễn hơn.

Để hoàn thành đề tài này, em xin được gửi lời cảm ơn chân thành đến:

- ✓ Thầy giáo hướng dẫn đê tài – **Nguyễn Mạnh Sơn**, Giảng viên Khoa Công nghệ Thông tin học viện Bưu Chính Viễn Thông - đã hết lòng giúp đỡ, hướng dẫn, chỉ dạy tận tình để em hoàn thành được đê tài này.
- ✓ Ban quản lý bến xe Giáp Bát, Mỹ Đình, cũng như các nhà xe đã tạo điều kiện.
- ✓ Các bạn cùng học đã góp ý và giúp đỡ trong quá trình làm đê tài này.

Hà Nội, tháng 06 năm 2014

Lê Văn Sơn, L12CN2

DANH MỤC BIỂU ĐỒ

Biểu đồ 2.2.1-1: Biểu đồ use case tổng quát	33
Biểu đồ 2.2.1-2: Use case “cập nhật hãng xe”	34
Biểu đồ 2.2.1-3: Use case “cập nhật trạng thái xe”	35
Biểu đồ 2.2.1-4: Use case “tra cứu”	36
Biểu đồ 2.2.1-5: Use case “thống kê”	36
Biểu đồ 2.2.2-1: Biểu đồ lớp trong pha phân tích.....	40
Biểu đồ 2.2.3-1: biểu đồ trạng thái khách hàng.....	41
Biểu đồ 2.2.3-2: Biểu đồ trạng thái của xe khách.....	41
Biểu đồ 2.2.3-3: Biểu đồ trạng thái tài xế.....	42
Biểu đồ 2.3.1-1: Biểu đồ tuần tự cho use case “tra cứu”	43
Biểu đồ 2.3.1-2: Biểu đồ tuần tự use case “cap nhat trang thai xe”	44
Biểu đồ 2.3.2-1: Biểu đồ cộng tác use case cap nhat trang thai xe.....	45
Biểu đồ 2.3.2-2: Biểu đồ cộng tác use case tra cứu	45
Biểu đồ 2.3.3-1: Biểu đồ lớp chi tiết.....	46
Biểu đồ 2.3.4-1: Biểu đồ hoạt động phương thức tra cứu lớp xe khách	47
Biểu đồ 2.3.5-1: Biểu đồ triển khai hệ thống.....	51

DANH MỤC BẢNG

Bảng 2.1.2-1: Bảng khảo sát thực trạng đón trả xe khách.....	30
Bảng 2.2.1-1: Scenario cho use case tra cứu	37
Bảng 2.2.1-2: Scenario use case “cập nhật số ghế trống”	39
Bảng 2.2.1-3: scenario use case “cập nhật xe”	39
Bảng 2.3.4-1. Các thiết kế chi tiết của lớp hàng xe	48
Bảng 2.3.4-2: Bảng thiết kế chi tiết của lớp tài xế.....	49
Bảng 2.3.4-3: Bảng thiết kế chi tiết lớp xe khách.....	49
Bảng 2.3.4-4: Bảng thiết kế chi tiết lớp loại xe	49
Bảng 2.3.4-5: Bảng thiết kế chi tiết lớp lộ trình	50
Bảng 2.3.4-6: Bảng thiết kế chi tiết lớp lịch sử	50

DANH MỤC HÌNH VẼ

Hình 1.1.1-1. Mô hình PhoneGap	11
Hình 1.1.1-2. Một số sản phẩm thực hiện với PhoneGap	12
Hình 1.1.1-3. Diagram – PhoneGap Buil	12
Hình 1.2.1-1: WebService	14
Hình 1.2.3-1: Mô hình hoạt động của Web service	16
Hình 1.4.2-1. Kiến trúc Android	24
Hình 2.1.2-1. Phiếu khảo sát thực trạng đón xe khách	28
Hình 3.2.1-1. Login của lái xe	54
Hình 3.2.1-2. Giao diện chính cho lái xe.....	54
Hình 3.2.1-3. Form tắt trạng thái bắt khách.....	55
Hình 3.2.1-4: Form bật trạng thái bắt khách.....	56
Hình 3.2.1-5: Giao diện cập nhật ghé trống	56
Hình 3.2.1-6: Giao diện đăng xuất của driver	57
Hình 3.2.2-1. Form tra cứu thông tin	58
Hình 3.2.2-2: Giao diện hiển thị thông tin tra cứu.....	59
Hình 3.2.2-3: Form thông báo không tìm thấy kq tra cứu.....	59
Hình 3.2.2-4: Giao diện xem lộ trình xe khách	60
Hình 3.2.2-5: Location of Client.....	60
Hình 3.2.2-6. Cập nhật bản đồ	61
Hình 3.2.2-7. Đặt xe	61
Hình 3.2.2-8. Cập nhật thông tin.....	62
Hình 3.2.3-1. Giao diện home của web server (QTV).....	63

Hình 3.2.3-2. Giao diện đăng nhập nhân viên 64

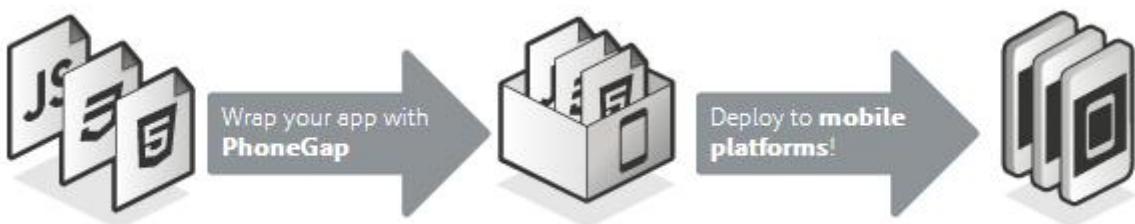
Hình 3.2.3-3. Giao diện đăng nhập của QTV 64

CHƯƠNG 1. TÌM HIỂU CÔNG NGHỆ PHONEGAP VÀ CÁC VẤN ĐỀ LIÊN QUAN

1.1 Phonegap

1.1.1 Vì sao lại lựa chọn PhoneGap

PhoneGap là một framework mã nguồn mở giúp nhanh chóng xây dựng nền tảng các ứng dụng với html5, javascript và css.



Hình 1.1.1-1. Mô hình PhoneGap

Phone Gap hỗ trợ phát triển các ứng dụng cho các thiết bị khác nhau như Iphone, Android, Windows Phone và nhiều framework hay ngôn ngữ khác. PhoneGap sử dụng các công nghệ web chuẩn cơ sở để xây dựng các ứng dụng web và thiết bị di động. Khi các ứng dụng Phonegap trở thành cách chuẩn phù hợp, cấu trúc của họ làm việc với các trình duyệt như họ phát triển.

PhoneGap được tải về bởi hơn một triệu lượt và được sử dụng bởi trên 400000 nhà phát triển. Hàng ngàn ứng dụng được phát triển sử dụng PhoneGap trên các kho ứng dụng điện thoại di động.

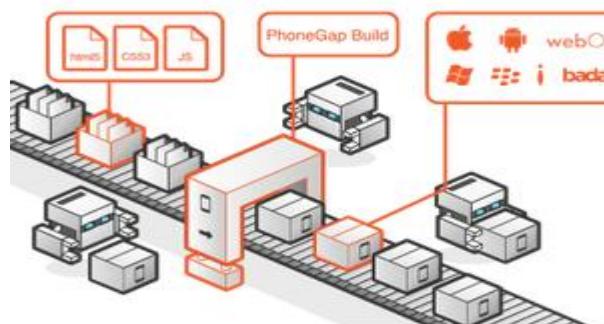


Hình 1.1.1-2. Một số sản phẩm thực hiện với PhoneGap

Ví dụ với phần mềm HealthTap – một phần mềm tốt nhất giúp tìm kiếm các câu trả lời của bác sĩ một cách nhanh chóng, tin cậy mọi lúc mọi nơi cho sức khỏe của bạn mà lại hoàn toàn miễn phí.

Mã nguồn PhoneGap được đóng góp bởi Apache Software Foundation (ASF) dưới tên **Apache Cordova** và được chia làm nhiều mức đề tài từ tháng 10/2012. Thông qua ASF, các thành phần phát triển của PhoneGap sẽ được đảm bảo mở rộng với nhiều ưu đãi cho các dự án. Nó sẽ thường được miễn phí và mã nguồn mở dưới các Apache License, version 2.0.

Diagram - PhoneGap Build
PNG, EPS



Hình 1.1.1-3. Diagram – PhoneGap Buil

Như trình bày ở trên thì với PhoneGap ta có thể built ứng dụng sang tới 7 hệ điều hành Mobile phổ biến này do sử dụng các ngôn ngữ cơ bản các hệ điều hành di động đều có thể đọc được là html5, javascript và css.

Đây là một hướng phát triển đúng đắn bởi nó có thể thu hút lượng người dùng sản phẩm của bạn lớn hơn. Khách hàng dùng nhiều dòng điện thoại vẫn có thể dùng sản phẩm của bạn chứ không hạn chế như các khách hàng dùng android thì không thể dùng được các ứng dụng của IOS .v.v.. Có một cách khác để tăng lượng người dùng trên tất cả các dòng điện thoại là với mỗi dòng bạn phát triển một phiên bản của sản phẩm tương ứng. Tuy nhiên việc này sẽ tốn kém của bạn nhiều thời gian, công sức và tiền bạc hơn. Điều đó là hiển nhiên.

Vậy tại sao ta lại không lựa chọn một nền tảng tiện dụng, đầy đủ như PhoneGap vào xây dựng, phát triển các dự án của mình.

1.1.2 *Tìm hiểu về Apache Cordova*

Apache Cordova là một tập hợp các API cho ứng dụng di động khả năng kết nối trực tiếp với thiết bị chức năng như là camera hay accelerometer từ JavaScript. Được kết hợp với một framework UI như Jquery mobile hay Dojo Mobile, Sencha Touch, Apache Cordova chấp nhận cho các ứng dụng trên smartphone phát triển dựa trên HTML, CSS và JS.

Khi sử dụng các API Cordova, một ứng dụng cần được xây dựng với một vài mã nguồn riêng (java, Objective C...) từ nhà phát triển ứng dụng. Thay vào đó công nghệ web được sử dụng, và chúng làm trong bản thân ứng dụng cục bộ (thông thường không trong một http server ở xa)

Và bởi vì các API JavaScript thích hợp với nhiều thiết bị nền tảng khác nhau và được xây dựng dựa trên các chuẩn về web nên ứng dụng có thể chạy trên các thiết bị khác nhau với những thay đổi rất nhỏ.

Ứng dụng sử dụng Cordova gồm các gói như các ứng dụng sử dụng nền tảng SDK và có thể tạo được các cài đặt khả dụng trên các kho lưu trữ ứng dụng của thiết bị khác.

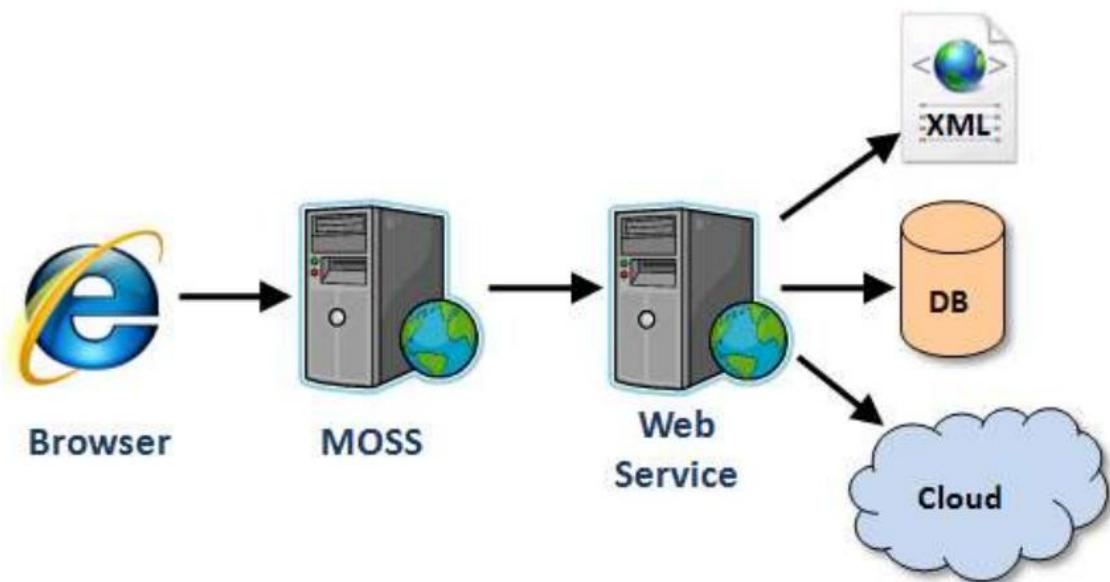
Cordova phát triển một tập hợp các thư viện JavaScript thống nhất có thể được gọi, với thiết bị cụ thể trả lại mã nguồn cho những thư viện JavaScript. Cordova có thể chạy được trên các nền tảng như IOS, Android, Blackberry, Windows Phone, Palm WebOS, Bada và Symbian.

1.2 WebService

1.2.1 Khái niệm

Theo định nghĩa của W3C (World Wide Web Consortium), Web Service là một hệ thống phần mềm được thiết kế để hỗ trợ khả năng tương tác giữa các ứng dụng trên máy tính khác thông qua mạng internet, giao diện chung và sự gắn kết của nó được mô tả bằng XML.

Web Service mô tả một cách thức chuẩn để tích hợp các ứng dụng dựa trên nền tảng Web, sử dụng các chuẩn mở XML (Extensible Markup Language), SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language)... Thông qua giao thức internet.



Hình 1.2.1-1: WebService

1.2.2 Một số đặc điểm cơ bản

Một webservice có thể được truy cập thông qua web.

Một webservice có giao diện dịch vụ. Giao diện này giúp cho webservice có thể được gọi bởi bất kỳ một ứng dụng nào hay bởi webservice nào khác. Giao diện dịch vụ là một tài liệu XML. Bởi vì XML có thể thể hiện rõ vai trò trong công nghệ trao đổi thông tin toàn cục (global exchange technology) được chấp nhận bởi phần lớn công nghệ hiện nay.

Các Dịch vụ web dùng giao thức tiêu chuẩn web để giao tiếp, không như COM,

RMI hay CORBA. XML được dùng để trao đổi thông tin giữa các chương trình ứng dụng và dịch vụ.

1.2.3 *Ưu – nhược điểm*

Ưu điểm:

Khả năng vượt firewall: Web Service hoạt động trên nền HTTP nên sử dụng port 80, là port mà firewall cho phép.

Hoạt động trên đa môi trường: Web Service dựa trên công nghệ XML được hỗ trợ bởi tất cả các hệ điều hành kể cả môi trường di động.

Tính linh hoạt, dễ chuyển đổi: Web Service chỉ bao gồm các hàm nên việc thay đổi dễ dàng và giảm chi phí đường truyền do truyền dữ liệu không kèm định dạng HTML.

Web Service cung cấp khả năng hoạt động rộng lớn với các ứng dụng phần mềm khác nhau chạy trên những nền tảng khách nhau.

Sử dụng giao thức và chuẩn mở. Giao thức và định dạng dữ liệu dựa trên văn bản (text), giúp các lập trình viên dễ dàng hiểu được.

Nâng cấp khả năng tái sử dụng.

Thúc đẩy đầu tư các hệ thống phần mềm đã tồn tại bằng cách cho phép các tiến trình/ chức năng nghiệp vụ đóng gói trong giao diện Web Service.

Tạo mối quan hệ tương tác lẫn nhau và mềm dẻo giữa các thành phần trong hệ thống, dễ dàng cho việc phát triển hệ thống phân tán.

Thúc đẩy hệ thống tích hợp, giảm sự phức tạp của hệ thống. hạ giá thành hoạt động, phát triển hệ thống nhanh và tương tác hiệu quả với hệ thống của các doanh nghiệp khác.

Nhược điểm:

Những thiệt hại lớn sẽ xảy ra vào khoảng thời gian chết của Web Service, giao diện không thay đổi, có thể lỗi nếu một khách hàng không được nâng cấp, thiếu các giao thức cho việc vận hành.

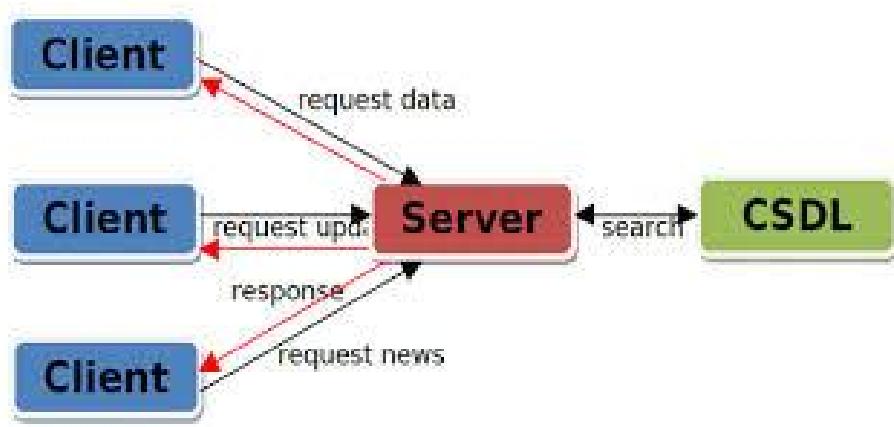
Có quá nhiều chuẩn cho Web Service khiến người dùng khó nắm bắt.

Phải quan tâm nhiều hơn về vấn đề an toàn và bảo mật của Web Service.

1.2.4 Hoạt động của web service

Một ứng dụng Web Service bao gồm 2 thành phần: Client và Server giao tiếp với nhau qua giao thức HTTP.

- Client gửi các yêu cầu thông qua các lời gọi hàm **HTTP Request** đến server.
- Server gửi các kết quả được thực thi ở các hàm thông qua **HTTP Response**



Hình 1.2.4-1: Mô hình hoạt động của Web service

Các kiểu dữ liệu trả về của Web Service cho các ứng dụng di động:

- SOAP
- XML
- JSON OBJECT

1.2.5 JSON

JSON là viết tắt của JavaScript Object Notation.

JSON là định dạng trao đổi dữ liệu văn bản dung lượng nhẹ.

JSON sử dụng cú pháp JavaScript để mô tả đối tượng dữ liệu, nhưng JSON là ngôn ngữ và nền tảng độc lập. Trình phân tích cú pháp(parsers) JSON và các thư viện JSON tồn tại cho nhiều ngôn ngữ lập trình khác nhau.

JSON được "tự mô tả" và dễ hiểu.

1.3 Google Maps API

1.3.1 *Google map*

1.3.1.1 Giới thiệu về Google map

Google Maps là một dịch vụ ứng dụng và công nghệ bản đồ trực tuyến trên web miễn phí được cung cấp bởi Google và hỗ trợ nhiều dịch vụ dựa vào bản đồ như Google Ride Finder và một số có thể dùng để nhúng vào các trang web của bên thứ ba thông qua Google Maps API. Nó cho phép thấy bản đồ đường sá, đường đi cho xe đạp, cho người đi bộ (những đường đi ngắn hơn 6.2 dặm[1]) và xe hơi, và những địa điểm kinh doanh trong khu vực cũng như khắp nơi trên thế giới. (*theo wiki*)

Google Map là một trong những ứng dụng phổ biến nhất của Google hiện nay. Khi mà việc sử dụng bản đồ giấy đã trở nên lỗi thời thì dịch vụ tìm kiếm bản đồ số, bản đồ vệ tinh, hệ thống GPS là hết sức cần thiết. Giờ đây chúng ta có thể dễ dàng truy cập vào Google Map dù bất cứ ở đâu. Không chỉ có nền tảng trên web, Google còn cung cấp các nền tảng và hướng dẫn giúp cho lập trình viên có thể thao tác với Service của Google để sử dụng tiện ích Maps này

1.3.1.2 Google map for mobile

Năm 2006, Google đã giới thiệu một ứng dụng Java có tên là Google Maps for Mobile. Mục đích để chạy trên bất kỳ thiết bị di động nào dựa có nền tảng Java. Rất nhiều tính năng chính của phiên bản Google Maps trên web có mặt trong ứng dụng.

Ngày 28 tháng 11 năm 2007, Google Maps for Mobile 2.0 được phát hành. Nó cung cấp một dịch vụ xác định vị trí giống như GPS nhưng vẫn hoạt động nếu thành phần GPS trong phần cứng không có hoạt không hoạt động. Vị trí hiện tại chỉ được bắt nếu thành phần GPS sẵn sàng hoạt động. Tính năng định vị không cần thành phần GPS dựa vào phần mềm xác định trạm phát sóng không dây và trạm BTS gần nhất. Sau đó phần mềm sẽ tìm trong cơ sở dữ liệu của nó vị trí của các điểm phát sóng WiFi và các trạm BTS mà nó biết được. Bằng cách xác định cường độ tín hiệu từ nhiều trạm phát sóng WiFi và BTS đã biết vị trí, phần mềm sẽ tính toán ra được vị trí hiện tại của người dùng.

Thứ tự ưu tiên tính toán vị trí như sau:

- Thiết bị bắt vị trí GPS
- Trạm WiFi

- Trạm BTS

Tính đến ngày 15 tháng 12 năm 2008, dịch vụ này có sẵn cho các nền tảng sau đây:

- Android
- iOS (iPhone, iPod Touch, iPad)
- PlayStation Vita 3G Models
- Windows Mobile (không hỗ trợ Windows Phone 7 từ 21/12/2011)
- Nokia/Symbian (S60 3rd & 5th)
- Symbian OS (UIQ v3)
- BlackBerry
- Điện thoại hỗ trợ Java-Platform (từ MIDP 2.0 trở lên)
- Palm OS (Centro và mới hơn)
- Palm webOS (Palm Pre và Palm Pixi)

Ngày 4 tháng 11, năm 2009, Google Maps Navigation đã được phát hành trên hệ điều hành Google Android 2.0 Eclair trên điện thoại Motorola Droid, thêm chức năng ra lệnh bằng giọng nói, báo cáo lưu lượng truy cập, và hỗ trợ xem đường phố. Phát hành ban đầu được giới hạn ở Hoa Kỳ. Dịch vụ này đã được đưa ra tại Anh vào ngày 20 tháng 4 năm 2010 và trong phần lớn của lục địa châu Âu vào ngày 09 tháng 6 năm 2010 (bao gồm Áo, Bỉ, Canada, Đan Mạch, Pháp, Đức, Ý, Hà Lan, Bồ Đào Nha, Tây Ban Nha, và Thụy Sĩ).

Trong tháng 3 năm 2011, Phó Chủ tịch Dịch vụ Location Service, Marissa Mayer thông báo Google đã cung cấp dịch vụ bản đồ đến 150 triệu người sử dụng. Trong tháng sáu năm 2012, Apple thông báo rằng họ sẽ loại bỏ Google Maps trên IOS 6 và sẽ được thay thế bằng dịch vụ bản đồ của riêng họ.

1.3.2 Google Maps API Web Services

Google Maps API Web Services là một tập các giao diện HTTP cung cấp thông tin địa lý cho ứng dụng. Google Maps API Web Services bao gồm:

- Directions API
- Distance Matrix API
- Elevation API
- Geocoding API
- Places API

Để làm việc tốt với hệ thống bản đồ của google, thì cần sử dụng một số các dịch vụ cần thiết để lập trình. Điểm chung của các dịch vụ này là đều chạy trên môi trường web và đều trả kết quả qua định dạng file JSON hoặc là XML.

1.3.3 Google Geocoding API

Geocoding là tiến trình chuyển từ địa chỉ (“Minh Khai, Hai Bà Trưng, Hà Nội”) qua tọa độ địa lý (37.423021, -122.083739) → mã hóa địa lý.

Geocoding request :

Truy cập dịch vụ geocoding qua đối tượng *google.maps.Geocoder*. Phương thức *Geocoder.geocode()* gửi request tới dịch vụ geocoding, qua đó một đối tượng *GeocodeRequest* chứa những điều kiện input và một phương thức callback.

```
{  
    address: string,  
    location: LatLng,  
    bounds: LatLngBounds,  
    region: string  
}
```

Geocoding responses:

```
results[]: {  
    types[]: string,  
    formatted_address: string,  
    address_components[]: {  
        short_name: string,  
        long_name: string,  
        postcode_localities[]: string,  
        types[]: string  
    },  
    partial_match: boolean,  
    geometry: {  
        location: LatLng,  
        location_type: GeocoderLocationType  
        viewport: LatLngBounds,  
        bounds: LatLngBounds  
    }  
}
```

Geo response – Status code:

- google.maps.GeocoderStatus.OK : kết quả trả về bình thường
- google.maps.GeocoderStatus.ZERO_RESULTS : địa chỉ không tồn tại hoặc giá trị tọa độ không đúng
- google.maps.GeocoderStatus.OVER_QUERY_LIMIT : thông báo đã gửi quá số lượng yêu cầu trong một thời gian
- google.maps.GeocoderStatus.REQUEST_DENIED : ứng dụng đã bị dịch vụ Distance Matrix API chặn
- google.maps.GeocoderStatus.INVALID_REQUEST : thông báo chuỗi request không hợp lệ

1.3.4 Google Distance Matrix API

Google Distance Matrix API là một dịch vụ cung cấp khoảng cách và thời gian di chuyển cho một ma trận các điểm gốc và điểm đích. Thông tin trả về phụ thuộc vào đường đi được đề nghị giữa điểm gốc và điểm đích theo tính toán của Google Distance Matrix API, bao gồm các giá trị khoảng cách và thời gian cho mỗi cặp.

Dịch vụ Google Distance Matrix API không cung cấp chi tiết đường đi. Muốn tìm thông tin chi tiết đường đi có thể sử dụng dịch vụ Google Direction API với chỉ một cặp điểm gốc và điểm đích.

Mỗi truy vấn gửi cho Distance Matrix API bị giới hạn số lượng phần tử, cứ mỗi một điểm gốc đến một điểm đích là một phần tử.

Số lượng phần tử bị giới hạn như sau:

- 100 phần tử cho mỗi truy vấn.
- 100 phần tử cho mỗi 10 giây.
- 2500 phần tử cho mỗi giờ.

Google MAP API trả phí sẽ có giới hạn lớn hơn:

- 625 phần tử cho mỗi truy vấn.
- 1000 phần tử cho mỗi 10 giây.
- 100000 phần tử cho mỗi giờ.

Các Distance Matrix API URL giới hạn ở 2048 ký tự trước khi mã hóa. Trong mỗi chuỗi Distance Matrix API URL sẽ có nhiều thông tin địa điểm được đưa vào vì vậy nên chú ý giới hạn này khi xây dựng chuỗi request URL. Sử dụng Distance Matrix API cần phải kết hợp với một bản đồ Google Map dùng để hiển thị các địa điểm một cách trực quan trên bản đồ

1.3.5 Google Direction API

Google Direction API là dịch vụ tính toán đường đi giữa 2 điểm sử dụng HTTP request. Thông tin đầu vào của điểm đầu, điểm kết thúc, điểm trung gian có thể ở dạng tên xác định địa điểm hay giá trị kinh độ - vĩ độ. Direction API có thể trả về nhiều hướng đi khác nhau sử dụng một tập các điểm trung gian.

Dịch vụ này được thiết kế để tính toán đường đi giữa những vị trí xác định rõ thông tin trên bản đồ Google Map. Vì quá trình tính toán đường đi mất thời gian và phức tạp (phụ thuộc vào khoảng cách hai điểm) nên bắt cứ khi nào có thể nên thực hiện công việc tính toán trước và lưu trữ tạm thời kết quả nhận được.

Có thể truyền vào một tham số giá trị địa chỉ kinh độ - vĩ độ. Nếu truyền vào địa chỉ, hệ thống Google Direction sẽ chuyển nó sang kinh độ - vĩ độ để tính toán.

Google Direction API cho phép 2500 truy vấn mỗi ngày. Mỗi lần tính toán hướng đi có thể sử dụng tối đa 9 điểm trung gian. Nếu có nhu cầu cao hơn, có thể sử dụng Google Direction API cho doanh nghiệp với tối đa 100000 truy vấn mỗi ngày và 23 điểm trung gian ở mỗi truy vấn.

Ngoài ra mỗi chuỗi URL truy vấn được giới hạn ở 2048 ký tự trước khi mã hóa. Một số truy vấn tìm đường có thể chứa thông tin các điểm dọc đường đi vì vậy cần lưu ý độ dài của nó.

Kết quả trả về của Google Direction API phải được sử dụng kết hợp với bản đồ Google Map. Tức là thông tin phải kết hợp với hiển thị đường đi trên bản đồ. Không nên sử dụng thông tin trả về mà không dung Goolge Map

```
{  
    origin: LatLng | String,  
    destination: LatLng | String,  
    travelMode: TravelMode,  
    transitOptions: TransitOptions,  
    unitSystem: UnitSystem,  
    durationInTraffic: Boolean,  
    waypoints[]: DirectionsWaypoint,  
    optimizeWaypoints: Boolean,  
    provideRouteAlternatives: Boolean,  
    avoidHighways: Boolean,  
    avoidTolls: Boolean  
    region: String  
}
```

Khi tính toán đường đi sử dụng Google Direction API, sử dụng các điểm trung gian để tính toán đường đi đến đích có đi qua những điểm trung gian đã quy định.

Có thể sử dụng nhiều điểm trung gian, mỗi điểm phân cách nhau bằng một ký tự “|”

Mặc định Google Direction sẽ tính toán đường đi qua các điểm trung gian dựa trên thứ tự sắp xếp của khai báo. Tuy nhiên có thể sử dụng tham số optimizeWaypoints “true” để G-Direction sắp xếp tối ưu nhất các điểm trung gian.

Rendering Directions: Status

Thông báo trạng thái kết quả hoặc thông tin sự cố

- OK : kết quả trả về bình thường
- NOT_FOUND : một điểm nhập vào nào đó không thể mã hóa địa lý để lấy tọa độ
- ZERO_RESULTS : không thể tìm thấy đường đi
- MAX_WAYPOINTS_EXCEEDED : thông báo vượt quá giới hạn các điểm trung gian được cho phép. (Bình thường là 9 điểm, nếu trả phí là 23)
- OVER_QUERY_LIMIT: thông báo đã gửi quá số lượng yêu cầu trong một khoảng thời gian
- REQUEST_DENIED : ứng dụng đã bị dịch vụ Google Direction chặn
- UNKNOWN_ERROR : lỗi không xác định

DirectionsResult chứa kết quả truy vấn directions. Chúng ta có thể tự xử lý hoặc chuyển qua cho đối tượng **DirectionsRenderer**, nó sẽ tự hiển thị kết quả trên bản đồ.

Trong kết quả trả về của Google Direction sẽ có nhiều thẻ route, một hoặc không có nếu không tìm thấy đường đi. Mỗi route chứa một tuyến đường đi riêng biệt từ điểm đầu tới điểm cuối.

Trong mỗi thẻ route sẽ có một hay nhiều thẻ leg phụ thuộc vào số lượng điểm trung gian yêu cầu.

1.4 Tổng quan Android

1.4.1 Android là gì? Tại sao cần quan tâm tới nền tảng Android

Android là một hệ điều hành dành cho thiết bị di động như điện thoại, máy tính bảng và netbooks. Android được phát triển bởi Google, dựa trên nền tảng Linux

kernel và các phần mềm nguồn mở. Ban đầu nó được phát triển bởi Android Inc (sau đó được Google mua lại) và gần đây nó trở thành một trong những phần mềm đứng đầu của liên minh OHA (Open Handset Alliance - với khoảng 78 thành viên bao gồm cả nhà sản xuất, nhà phát triển ứng dụng... cho thiết bị di động mà dẫn đầu là Google). Andorid được phát triển nhằm cạnh tranh với các hệ điều hành di động khác như iOS (Apple), BlackBerry OS, Windows Mobile (Microsoft), Symbian (Nokia), Samsung (Bada), WebOS (Palm)... Theo thống kê trong quý II năm 2010 tại Mỹ, hệ điều hành Android chiếm thị phần 33% (cao nhất) trong tổng số các hệ điều hành di động được bán ra, thứ 2 là BlackBerry OS 28% và iOS (Apple) xếp thứ 3 với 22%.

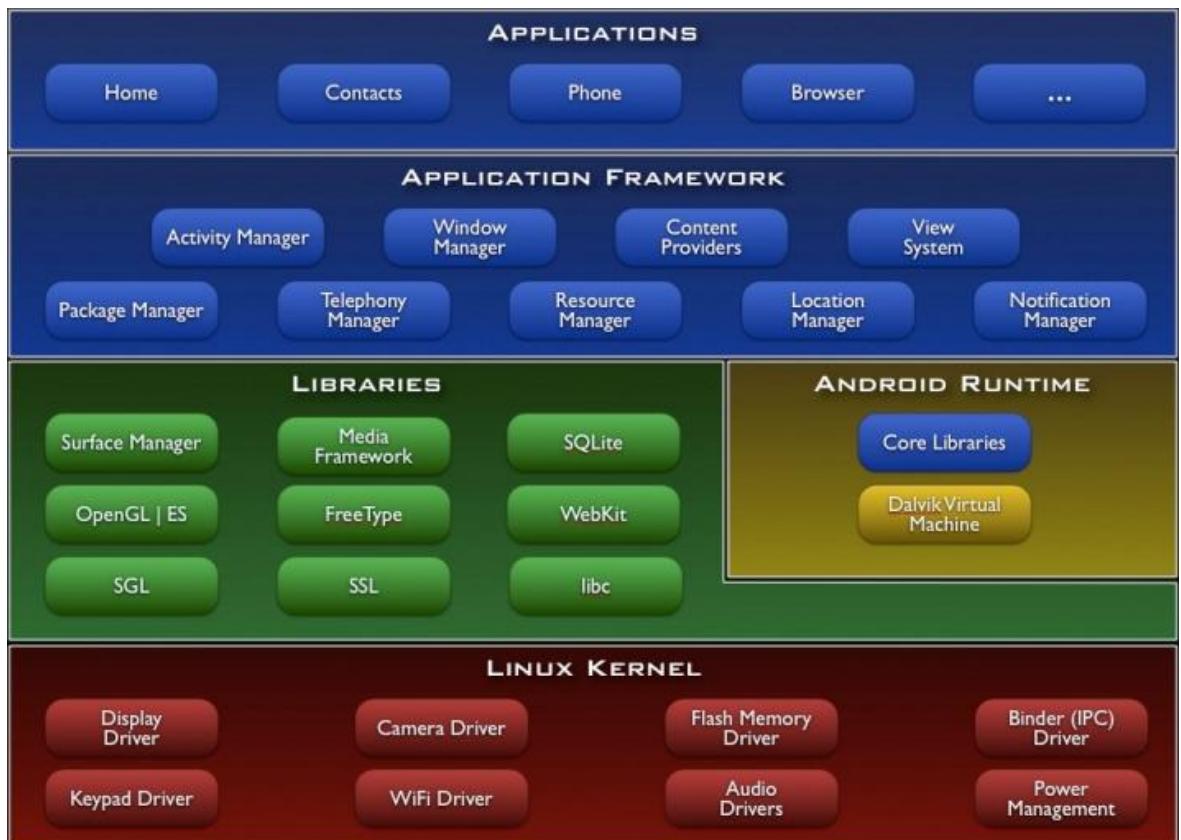
Tại sao cần quan tâm tới nền tảng Android?

Hiện nay trên thế giới có rất nhiều các nền tảng lập trình di động khác nhau, bao gồm Symbian, iPhone, Windows Mobile, BlackBerry, Java Mobile Edition, ... Vậy tại sao chúng ta lại cần quan tâm tới Android ? Tuy sinh sau đẻ muộn nhưng Android là nền tảng đầu tiên đem lại những điều sau:

- Một nền tảng hoàn toàn mở, tự do phát triển, dựa trên nhân Linux và mã nguồn mở. Các nhà chế tạo thiết bị cầm tay và các nhà phát triển phần mềm rất thích điều này vì họ có thể tự do tùy chỉnh lại nó và sử dụng nó mà không phải trả thêm chi phí bản quyền.
- Kiến trúc hệ thống dựa trên các thành phần rời rạc. Các lập trình viên có thể thay thế hoàn toàn một thành phần trong nền tảng bằng một phiên bản cải tiến khác của họ, điều này đem lại một xu hướng sáng tạo mới trong thế giới lập trình di động.
- Các dịch vụ mạnh mẽ được thiết kế bên trong: có thể kể đến các dịch vụ dựa trên vị trí hiện tại của người dùng sử dụng GPS cho phép ta cung cấp cho người dùng những trải nghiệm mới tùy vào nơi mà họ đang đứng, hoặc một hệ thống cơ sở dữ liệu sử dụng SQL cho phép ta lưu trữ dữ liệu và đồng bộ hóa chúng khi kết nối vào Internet.
- Vòng đời của ứng dụng được kiểm soát tự động. Các ứng dụng được cô lập lẫn nhau hoàn toàn bởi cơ chế bảo mật của Android. Điều này mang lại tính ổn định mà chưa một nền tảng nào hiện nay làm được. Người dùng không cần phải lo lắng xem ứng dụng nào đang chạy và ứng dụng nào cần phải tắt để các ứng dụng khác có thể hoạt động. Android được tối ưu cho các thiết bị có bộ nhớ thấp và ít năng lượng ở mức tốt nhất hiện nay.

- Hình ảnh và âm thanh chất lượng cao. Nhanh, mượt, hình ảnh được khử răng cưa cộng với khả năng tăng tốc cho đồ họa 3D bằng thư viện OpenGL cho phép nhiều loại ứng dụng doanh nghiệp và Game chạy được trên nền tảng này.
- Khả năng tương thích với nhiều loại phần cứng hiện nay và trong tương lai. Tất cả các chương trình được viết bằng Java và được thực thi bởi máy ảo Dalvik của Android, do đó mã nguồn của lập trình viên sẽ tương thích hoàn toàn với các kiến trúc chip như ARM, x86, ... Hỗ trợ một loạt các phương pháp nhập liệu như dùng bàn phím, cảm ứng, trackball. Giao diện người dùng có thể được tùy chỉnh theo độ phân giải và phương hướng của thiết bị.

1.4.2 Kiến trúc Android



Hình 1.4.2-1. Kiến trúc Android

CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

2.1 Khảo sát yêu cầu hệ thống

2.1.1 Khảo sát sơ bộ

Địa điểm khảo sát: tuyến xe ven quốc lộ 1A và đường Hồ Chí Minh từ bến xe Giáp Bát và Mỹ Đình - Hà Nội tới Nhu Xuân-Thanh Hóa.

Tài liệu, dữ liệu khảo sát: phiếu khảo sát, Card visit của nhà xe

Về phía nhà xe: Các nhà xe sẽ đi theo một số giờ nhất định trong ngày. Số lượng chuyến và khoảng cách giữa các xe chạy phụ thuộc vào lượng khách cũng như là số lượng xe của hãng. Ví dụ hãng xe khách Tiền Phương chạy tuyến Giáp Bát – Thanh Hóa thì 30p có một chuyến, còn tuyến Mỹ Đình – Thanh Hóa thì 60 phút một chuyến. Các xe sẽ chạy theo lộ trình cố định theo hãng quy định. Dọc đường khi có khách vẫy tay ra hiệu thì lái xe dừng xe bên đường và mở cửa cho khách lên xe. Sau đó, phụ xe sắp xếp chỗ ngồi và thu tiền phí và ghi lại điểm lên, điểm xuống của khách. Thường thì điểm đến của khách được mô tả từ khách và phụ xe dựa vào đặc điểm được mô tả và kinh nghiệm của mình để đoán được chỗ khách cần xuống. Khi khách đến điểm cần xuống thì khách hàng báo với phụ xe hoặc phụ xe sẽ gọi thông báo để khách xuống. Cứ liên tục bắt khách và trả khách như thế đến khi xe đầy chỗ thì ngừng bắt thêm khách. Đến bến xe thì xe trả khách và chờ đợi chuyến sau theo phân công của hãng xe.

Về phía khách hàng: Khi có nhu cầu đi xe thì khách hàng có thể có thông tin về các xe có lộ trình cùng với hướng mình đi tại các bến xe hoặc các quan sát các quảng cáo, lộ trình xe tại xe chạy qua hoặc hỏi người thân. Ví dụ khi bạn muốn đi từ Hà Nội về Thanh Hóa qua quốc lộ 1 thì có các xe của hãng xe Tiền Phương, Hoàng Long, Đông Lý, Anh Hào... có thể là lựa chọn của bạn. Tuy nhiên mỗi xe lại có một điểm đầu, điểm cuối là các bến xe khác nhau gây những khó khăn nhất định cho bạn. Khi đã xác định được tuyến xe cần đi thì bạn chọn địa điểm mà xe đi qua và đứng đợi xe đến. Quan sát thấy xe cần đi tới gần thì bạn ra tín hiệu lên xe (thường là vẫy tay) cho lái xe biết. Khi xe dừng thì bạn lên xe. Sau đó bạn cung cấp thông tin về điểm đến của mình cho phụ xe. Có thể miêu tả thêm thông tin điểm đến để phụ xe biết được điểm cần đến và thu vé xe. Khi đến điểm cần xuống khách hàng hỏi hoặc phụ xe thông báo để xuống xe.

2.1.2 Khảo sát chi tiết

Hệ thống dịch vụ xe khách truyền thông với mạng lưới dày đặc và sự đa dạng cũng như chất lượng phục vụ ngày một nâng cao đã mang lại cho khách hàng sự thuận tiện, thoải mái trong sử dụng.

Nhưng bên cạnh mặt tiêu cực vẫn còn tồn tại những hạn chế:

Về phía nhà xe:

- Thời gian xe chạy và lộ trình là cố định.
- Thông tin về xe có thể được cung cấp qua một thẻ card giới thiệu về tuyến xe.
- Không có cách nào thông báo tình trạng xe (còn bắt khách hay đã full ghế ngồi) cho các khách hàng đợi xe.
- Lái xe cứ chạy dọc theo tuyến đường được quy định của hãng và khi thấy khách ra tín hiệu thì dừng xe. Vì vậy lái xe không dự đoán được lúc nào có khách cũng như vị trí của khách đang chờ xe, khách muốn xuống xe để có được sự chủ động dừng xe hợp lý.
- Phụ xe phải mất thời gian nhất định và kinh nghiệm về tuyến đường mới có thể biết được chính xác khách muốn xuống ở đâu.
- Ngoài chờ khách ra thì nhà xe còn nhận chuyển phát nhanh một số hàng hóa nhẹ như giấy tờ, văn bản... và phụ xe phải liên tục gọi điện thông báo cho khách hàng giờ xe vào bến hoặc hẹn ở một vị trí nào đó. Điều này cũng gặp không ít khó khăn.

Về phía khách hàng:

- Với các tuyến mới đi thì không chủ động biết được xe nào đi qua, thời gian đi của xe là lúc nào. Muốn biết thông tin về các hãng xe phải tra cứu trên mạng hoặc đến các bến xe.
- Khi biết được hãng xe cần bắt thì lại không biết rõ là địa điểm nào có thể bắt được xe và thời gian đứng đợi bắt xe như thế nào cho hợp lý. Nếu đi quá sớm thì mất thời gian đợi và đi quá muộn thì có thể nhỡ xe.
- Không quan sát được lộ trình xe đi, nhất là khi gấp thời tiết xấu hoặc trời tối gây khó khăn trong việc chủ động bố trí điểm xuống xe cho thuận tiện. Có thể phải thường xuyên hỏi phụ xe gây khó chịu cho họ.
- Khách hàng phải nhớ các tuyến xe, thời gian xe chạy tại đầu bến, lộ trình xe. Hoặc khách hàng có thể xin được một thẻ card giới thiệu thông tin về nhà xe. Tuy nhiên có rất nhiều hãng xe, cũng như mỗi thẻ card

đó cũng không hề nhỏ gây khó khăn cho việc lưu trữ thông tin về các nhà xe.

- Nếu nhà xe thay đổi tuyến đường hoặc thời gian chạy thì không kịp cập nhật thông tin.
- Không biết được hết các tuyến xe có thể dùng cho lộ trình của mình.
- Nhiều khi có chờ đợi một chuyến xe mà không biết được xe đó đã đầy. Lại mất thời gian chờ chuyến xe sau gây lãng phí thời gian và sức lực.
- Tương tự khi bạn là người nhận bưu phẩm qua xe khách thì việc liên lạc nhận biết xe, xem xe đang ở đâu là tương đối khó khăn.

Sơ qua về thị trường smartphone(điện thoại thông minh) và tablet(máy tính bảng) tại Việt Nam:

- Tốc độ tăng trưởng thị trường điện thoại thông minh Việt Nam thuộc diện nhanh nhất Đông Nam Á. Cụ thể, 9 tháng đầu 2013, tốc độ tăng trưởng của thị trường này ở Việt Nam là 156% so với cùng kì năm ngoái. Xét về tổng giá trị sản phẩm bán ra, tăng 113% so với cùng kì. 6 tháng đầu năm 2013 đã có 206.000 tablet đến tay người tiêu dùng Việt Nam tăng 233%. (theo thống kê của GFK)
- Theo thống kê của Google, vào cuối quý II/2013, Việt Nam có hơn 17 triệu smartphone và số lượng này còn tăng mạnh trong thời gian tới.
- Việt Nam lọt vào top 10 về khía cạnh người dùng mới với 14,2 triệu người. Tỉ lệ người dùng smartphone lần đầu thì Việt Nam lọt top 3 là 82%.

PHIẾU KHẢO SÁT THỰC TRẠNG ĐÓN XE KHÁCH

Phản 1: Thực trạng đón xe khách của bạn là gì? (điền vào chỗ trống)

1. Thời gian đợi bắt xe trung bình của năm lần gần nhất của bạn là khoảng trongphút.
2. Lộ trình bạn hay tham gia là từ.....đến
3. Hãng xe khách bạn hay đi là hãng.....
4. Ước tính tỷ lệ bắt hụt xe của bạn là%
5. Thời điểm bạn bắt xe là vào lúc.....giờ.
6. Ý kiến bổ sung.....
.....

Phản 2: Nếu có một ứng dụng hỗ trợ đón xe khách bạn sẽ mong nó có chức năng gì? (Bôi đen để chọn)

1. Là một ứng dụng mobile
2. Chạy được trên nhiều dòng điện thoại như android, ios, blackberry...
3. Có giao diện bàn đồ trực quan để thấy được mình cũng như các xe đang chạy.
4. Có thể tra cứu được thông tin về các hãng xe.
5. Ý kiến bổ sung:.....

Chân thành cảm ơn bạn đã tham gia khảo sát!

Hình 2.1.2-1. Phiếu khảo sát thực trạng đón xe khách

Sau khi thực hiện lấy thông tin khảo sát là các sinh viên với tuyến ven quốc lộ 1 & đường HCM, điểm đến là Hà Nội, ta có được bảng tổng hợp như sau:

STT	TG đợi xe (phút)	Điểm xuất phát	Hãng xe hay đi	Tỷ lệ nhỡ xe (%)	Thời gian đón xe (buổi)	Ý kiến đóng góp
1	10	Yên Cát	Tiến Phương	10	Chiều	
2	15	Nông Cống	Đông Lý, Tiến Phương	10	Sáng	

3	20	TP Thanh Hóa	Tiến Phương, Đông Lý, Hoàn g Long	5	Sáng	
4	20	Dốc Xây	Tiến phương	20	Sáng	
5	10	TP Ninh Bình	Hoàng Long, Đông Lý,...	10	Chiều	
6	10	Hà Nam	Hải Âu, Hoàng Long...	15	Sáng, Chiều	
7	10	Cẩm Thủy	Anh Hào	40	Chiều	
8	15	Ngọc Lặc	Anh Hào	45	Tối	
9	10	Cúc Phương	Tự do	50	Chiều	
10	20	Ninh Bình	Tự do	40	Chiều	
11	15	Hòa Bình	Tự do	50	Chiều	
12	30	Hòa Bình	Tự Do	40	Sáng	
13	20	Lang Chánh	Tự do	50	Sáng	
14	15	Quan Sơn	Tự Do	45	Sáng	
15	20	Quan Sơn	Tự do	40	Chiều	
16	25	Quan Sơn	Hoàng Long	40	Cả ngày	
17	20	Quan Sơn	Tự Do	35	Sáng	
18	20	Quan Sơn	Tự do	55	Sáng	
19	30	Quan Sơn	Tự do	35		
20	25	Quan Sơn	Hoàng Long	50	Sáng	

21	50	Thọ Xuân	Hoàng Long	80	Chiều	
22	30	Thọ Xuân	Tự do	50	Sáng	
23	35	Thọ Xuân	Tự do	60	Cả ngày	
24	25	Thọ Xuân	Tự do	50	Sáng	
25	30	Thọ Xuân	Hải Âu	50	Sáng	
26	40	Thọ Xuân	Tự do	60	Chiều	
27	25	Thọ Xuân	Tự do	50	Sáng	
28	30	Thọ Xuân	Tự do	60	Sáng	
29	35	Thọ Xuân	Tự do	70	Sáng	
30	35	Hà Tây	Tự do	10	Sáng, Chiều	

Bảng 2.1.2-1: Bảng khảo sát thực trạng đón trả xe khách

Theo như một số số liệu thống kê lấy được từ khảo sát thực trạng đón xe khách trên tuyến Thanh Hóa – Hà Nội có một số điểm nhận xét như sau:

- Thời gian bắt xe trung bình từ 20 -25 phút, tỷ lệ nhỡ xe tương đối cao trong khoảng 40 – 45 %.
- Thời gian bắt xe trung bình, tỷ lệ nhỡ xe tăng dần theo khoảng cách điểm bắt so với bến xe. Khách hàng ở khu vực Hà Nội gần bến xe hơn nên tỷ lệ này nhỏ hơn. Tỷ lệ cao dần theo Ninh Bình, Thanh Hóa và Hòa bình. Cá biệt có trường hợp bắt xe gần một giờ đồng hồ với tỷ lệ nhỡ xe lên đến 80%.
- Khách hàng bắt xe theo một hãng có thời gian đón xe cùng tỷ lệ nhỡ xe cao hơn so với bắt xe theo lộ trình.
- Thời gian trong ngày không ảnh hưởng nhiều đến các tỷ lệ nhỡ xe cũng như thời gian bắt xe. Điều này do các hãng xe có thời gian phục vụ dài trong ngày.

- Chưa lấy được dữ liệu ngày lễ, tết, dịp đặc biệt. Dự kiến sẽ cao hơn nhiều nhất là các khu vực xa điểm đầu bến xe.

Ngoài ra phần điều tra ý kiến khách hàng thì đa số đều lựa chọn tất cả các thông tin mong muốn của một ứng dụng hỗ trợ đón xe khách nếu có.

- 90% chọn 1. Là một ứng dụng mobile
- 76% chọn 2. Chạy được trên nhiều dòng điện thoại như android, ios, blackberry...
- 85% chọn 3. Có giao diện bản đồ trực quan để thấy được mình cũng như các xe đang chạy.
- 60% chọn 4. Có thể tra cứu được thông tin về các hãng xe.

Đa số ý kiến bổ sung đều mong muốn ứng dụng có lượng xe phục vụ nhiều hơn và giao diện dễ sử dụng.

2.1.3 Xây dựng dự án

Hệ thống được tạo ra từ nhu cầu của người sử dụng dịch vụ xe khách, các lái xe khách và bản thân các nhà xe nhằm khắc phục những bất cập của hệ thống dịch vụ hiện nay.

Hệ thống được tích hợp phần mềm trên thiết bị di động, sử dụng qua kết nối wifi/3G cung cấp cho khách hàng dịch vụ hiệu quả nhất, thoải mái, chủ động, tiết kiệm, kiểm soát thông tin hoàn hảo. Lợi ích của khách hàng là số 1. Đối với lái xe khách, đây là công cụ hỗ trợ thuận tiện trong việc tiếp nhận các nhu cầu đặt xe, gọi xe một cách trực quan, tiết kiệm chi phí nhờ hệ thống dẫn đường, chủ động liên hệ khách hàng khi cần thiết, tăng độ tin cậy giữa lái xe và khách hàng.

Hệ thống có những tính năng nổi bật:

Đối với khách hàng:

- Thông tin trực quan về thông tin các xe hãng xe, lộ trình, giờ chạy... Hỗ trợ trên giao diện bản đồ.
- Hệ thống tự định vị nơi khách đang đứng và có thể cho phép chỉ đường ngắn nhất từ vị trí khách tới vị trí đón xe gần nhất.
- Khách hàng sẽ biết rõ thông tin xe khách phục vụ như biển số, loại xe, tên tài xế, hình ảnh, khoảng cách tới điểm bắt xe.
- Không phát sinh cước phí.
- Cho phép khách hàng thông báo nhu cầu đi xe của mình tới nhà xe và tài xế (đặt xe).

- Cho phép khách hàng đánh giá mức độ hài lòng với lái xe, giúp cải thiện chất lượng dịch vụ.
- Tự động lưu nhật ký hành trình khi kết thúc. Cho phép khách hàng liên lạc ngay với lái xe, hằng xe khi có vấn đề phát sinh.

Đối với tài xế:

- Dễ dàng cập nhật thông báo tới các khách hàng thông tin của xe, tình trạng của xe.
- Hỗ trợ bản đồ và định vị dẫn đường.
- Thông báo tới tài xế thông tin khách hàng có nhu cầu đi xe.

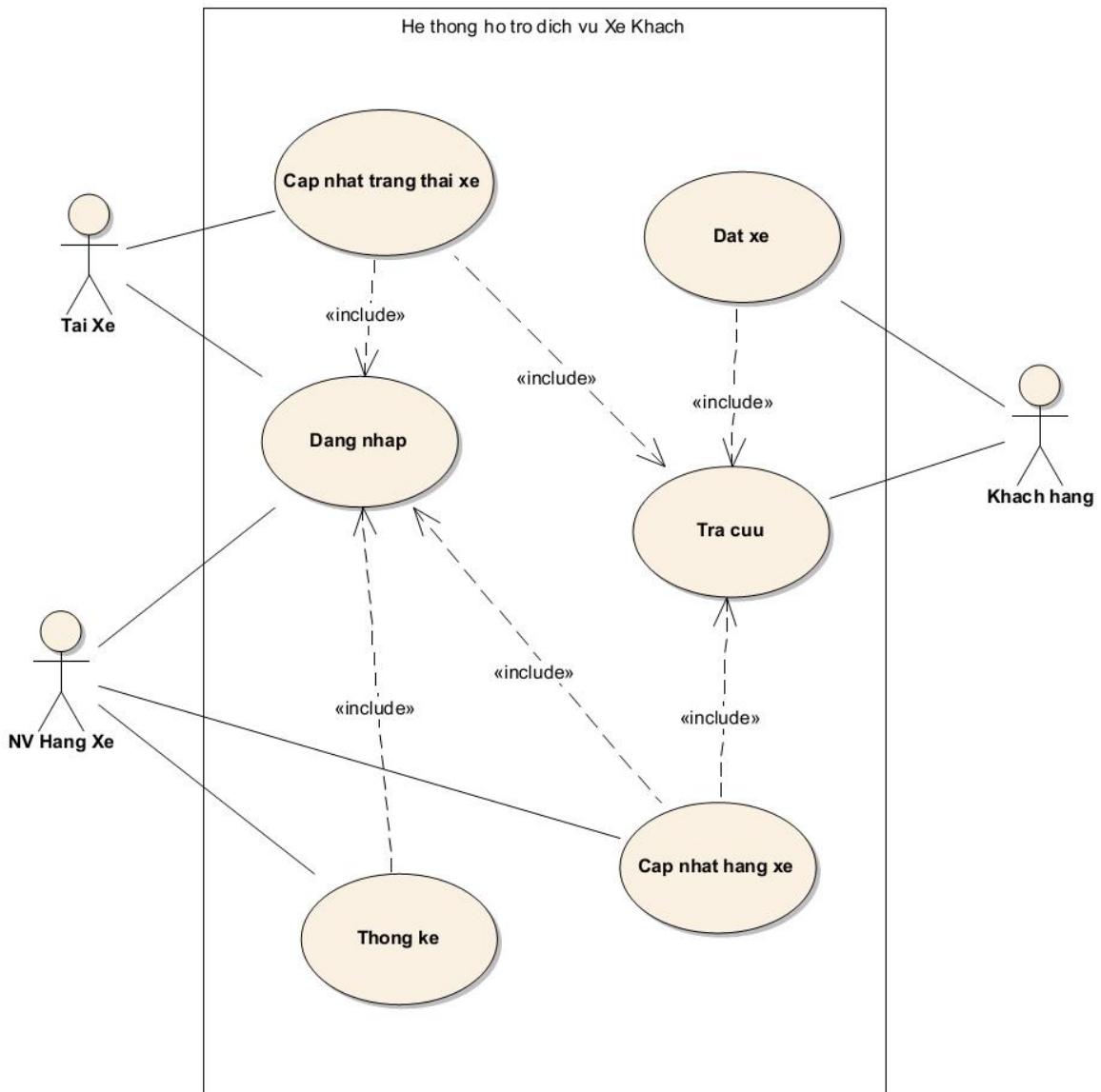
Vì hệ thống hướng tới người sử dụng công nghệ cao nên sẽ có những **hạn chế** nhất định như:

- Yêu cầu người dùng có smartphone hoặc tablet
- Thuận tiện ở những nơi có sóng wifi hoặc 3G mạnh
- Người dùng có tầm tri thức nào đó để hiểu và sử dụng hệ thống.

2.2 Phân tích hệ thống

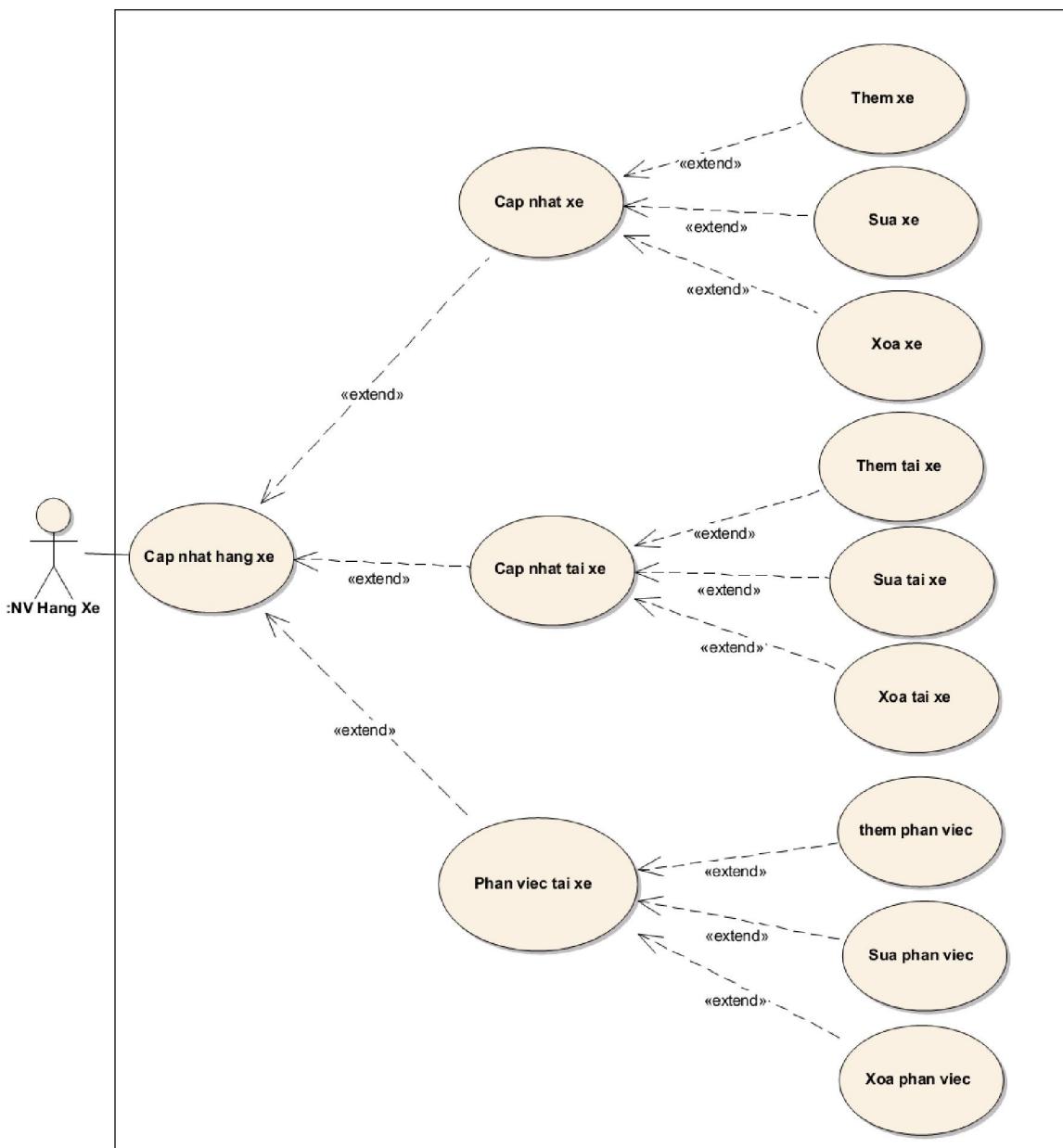
2.2.1 Biểu đồ Use Case

2.2.1.1 Use case tổng quát

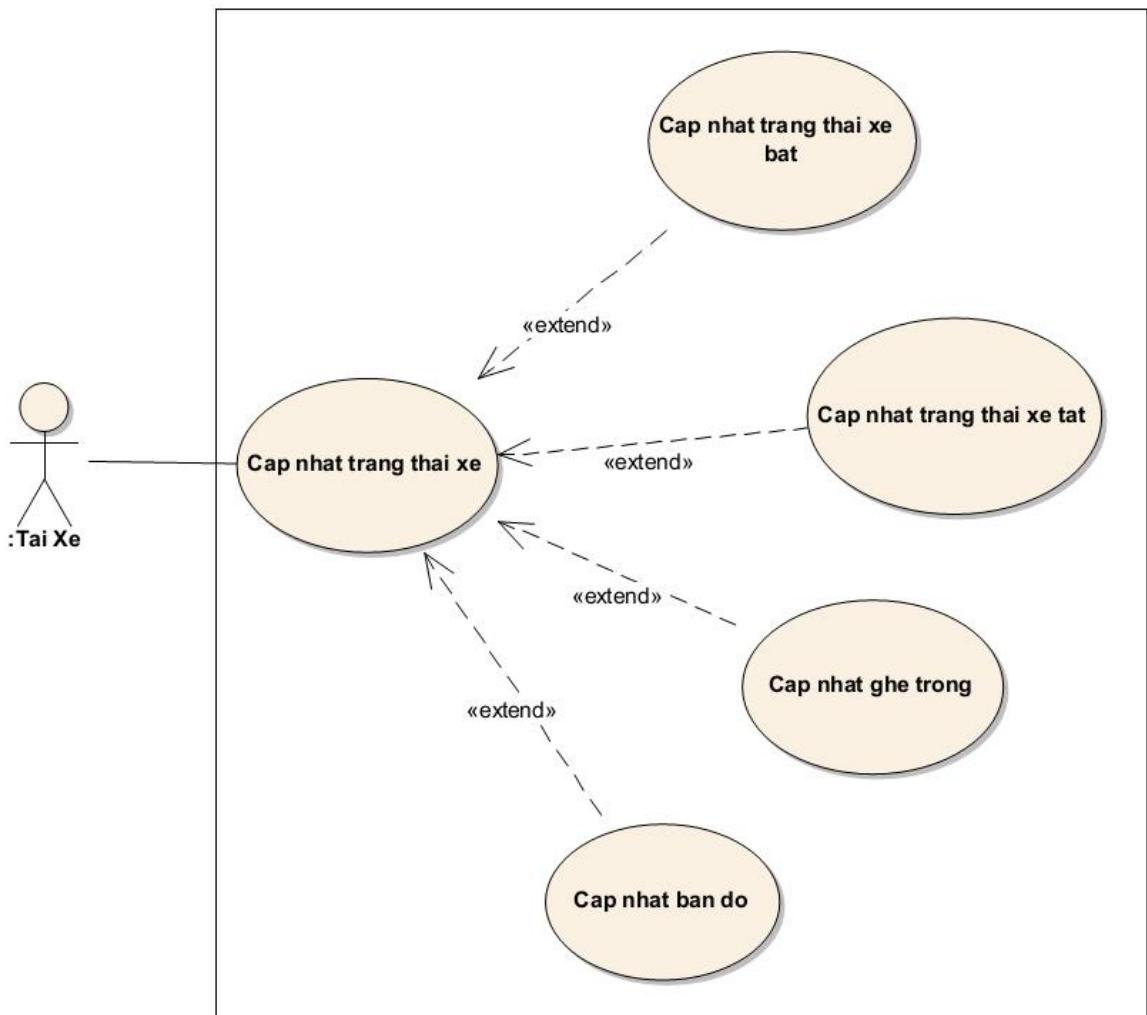


Biểu đồ 2.2.1-1: Biểu đồ use case tổng quát

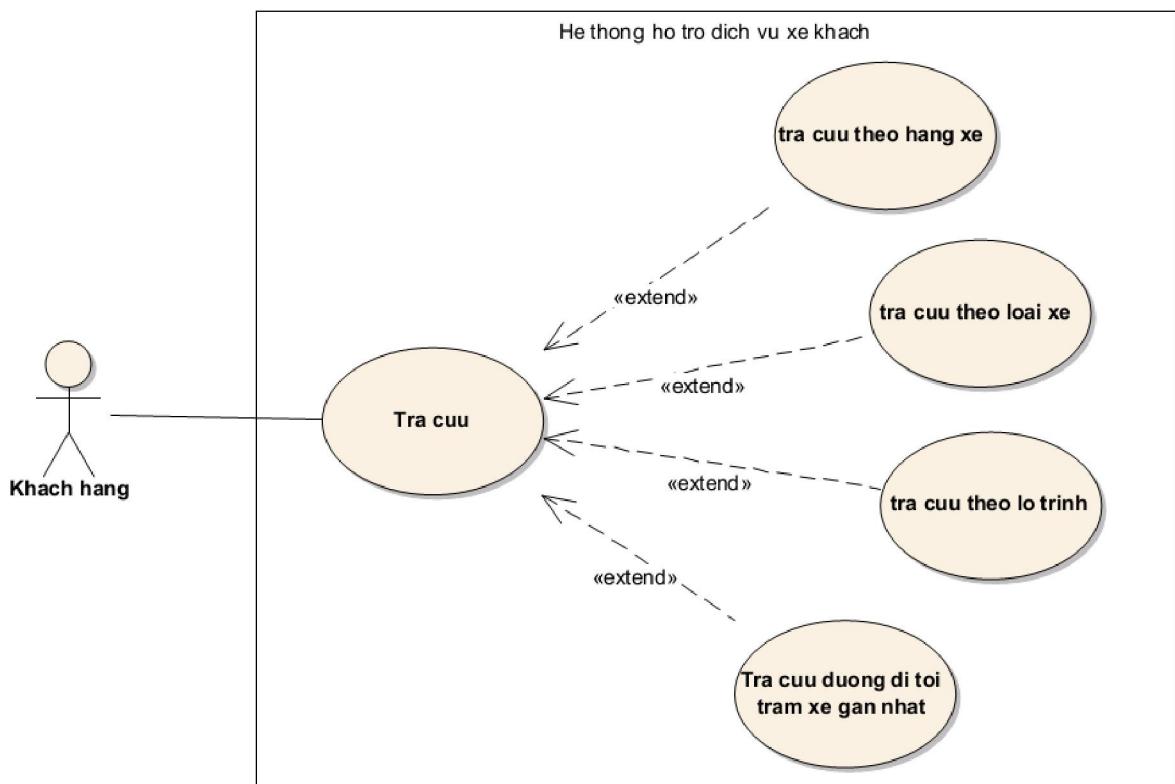
2.2.1.2 Phân rã Use case



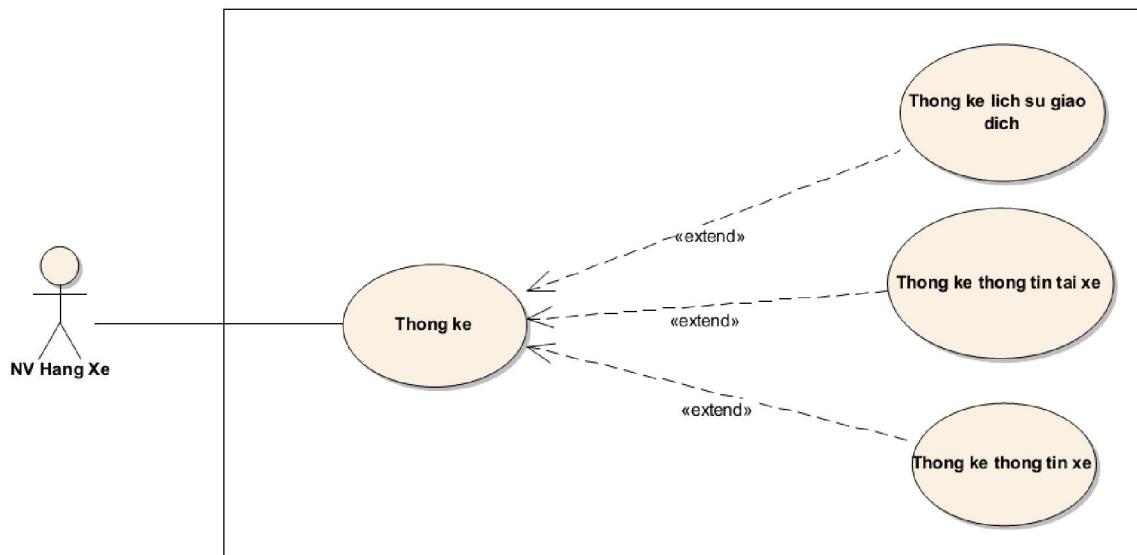
Biểu đồ 2.2.1-2: Use case “cập nhật hàng xe”



Biểu đồ 2.2.1-3: Use case “cập nhật trạng thái xe”



Biểu đồ 2.2.1-4: Use case “tra cứu”



Biểu đồ 2.2.1-5: Use case “thống kê”

2.2.1.3 Scenario

Đặc tả use case “tra cứu”

Tên UseCase	Tra cứu
Tác nhân chính	Khách hàng
Đảm bảo tối thiểu	Hệ thống loại bỏ các thông tin tìm kiếm thông báo cho khách hàng và quay lui lại bước trước
Đảm bảo thành công	Hiển thị các thông tin xe khách phù hợp, hiển thị trên bản đồ đường đi của xe và chỉ đường cho khách hàng tới trạm xe gần nhất
Kích hoạt	Khách hàng chọn chức năng tra cứu trong menu hoặc khách hàng bắt đầu mở ứng dụng

Chuỗi sự kiện chính:

1. Hệ thống hiển thị form tra cứu
2. Khách hàng nhập các thông tin tìm kiếm và nhấn nút tìm kiếm
3. Hệ thống kiểm tra thông tin và show kết quả cho khách hàng
4. Khách hàng xem thông tin tóm tắt, chi tiết và chọn chức năng xem chi tiết trên bản đồ hoặc chỉ đường tới trạm xe gần nhất
5. Hệ thống show bản đồ lộ trình đường đi của xe nếu khách hàng chọn chức năng xem chi tiết bản đồ. Show chỉ đường tới trạm bắt xe gần nhất nếu khách hàng chọn chức năng chỉ đường
6. Khách hàng thoát khỏi chức năng tra cứu (gọi, thoát khỏi hệ thống)

Ngoại lệ:

1. Hệ thống thông báo không có kết nối internet
2. Hệ thống thông báo cho khách hàng khi nhập không đúng format tìm kiếm
 - a. Hệ thống yêu cầu khách hàng nhập lại thông tin tìm kiếm
 - b. Khách hàng nhập lại thông tin tìm kiếm
3. Hệ thống không tìm thấy kết quả nào phù hợp
 - a. Hệ thống thông báo cho khách hàng
 - b. Quay lại trang tìm kiếm
5. Hệ thống thông báo không có dịch vụ định vị toàn cầu GPS
 - a. Hệ thống hỏi khách hàng có cài đặt lại GPS hay không
 - b. Khách hàng bật GPS
 - c. Hệ thống hiển thị vị trí hiện tại của khách hàng và chỉ đường tới trạm xe gần nhất.

Bảng 2.2.1-1: Scenario cho use case tra cứu

Đặc tả “cập nhật tài xế”

Tên Use case	Cập nhật trạng thái
Tác nhân chính	Tài xế

Mức	3
Tiền điều kiện	Tài xế đăng nhập vào hệ thống
Đảm bảo tối thiểu	Hệ thống thông báo lỗi và quay lui lại bước trước
Đảm bảo thành công	Hệ thống cập nhật thông tin trạng thái tài xế ~ xe vào CSDL
Kích hoạt	Tài xế nhấp chọn nút bật-tắt trạng thái trong menu

Chuỗi sự kiện chính:

1. Hệ thống hiển thị nút thay đổi trạng thái trên menu (header)
2. Tài xế click nút thay đổi trạng thái trên menu
3. Hệ thống thông báo xác nhận
4. Hệ thống xử lý cập nhật trạng thái vào CSDL
5. Hệ thống thông báo cho tài xế thay đổi trạng thái thành công
6. Tài xế xác nhận và thoát khỏi thông báo

Ngoại lệ:

4. Hệ thống thông báo gặp lỗi trong quá trình update
 - a. Một dialog hiện ra thông báo lỗi
 - b. Tài xế xác nhận và cập nhật trạng thái lại

Bảng 2.2.1.3.2: Scenario use case “cập nhật trạng thái”

Tên Use Case	Cập nhật số ghế trống
Tác nhân chính	Tài xế
Mức	3
Người chịu trách nhiệm	Quản lý hãng xe
Tiền điều kiện	Tài xế đăng nhập vào hệ thống
Đảm bảo tối thiểu	Hệ thống thông báo lỗi và quay lui lại bước trước
Đảm bảo thành công	Số ghế còn trống trên xe được cập nhật vào CSDL
Kích hoạt	Tài xế nhấp chọn nút “đang bật” trên menu header

Chuỗi sự kiện chính:

1. Hệ thống hiển thị nút “đang bật/tắt” trên menu header
2. Tài xế nhấp chọn nút
3. Hệ thống hiển thị form dialog cập nhật ghế trống cho xe
4. Tài xế nhập số ghế trống của xe
5. Hệ thống cập nhật số ghế trống hiện tại vào CSDL

6. Tài xế thoát khỏi chức năng cập nhật số ghế trống

Ngoại lệ:

1. Hệ thống thông báo không có kết nối internet cho thiết bị
 - a. Hệ thống hỏi người dùng có cài đặt lại network cho thiết bị hay không
 - b. Thoát hệ thống và chuyển qua phần cài đặt network
4. Hệ thống thông báo thông tin ghế tài xế nhập không hợp lệ
 - a. Hệ thống hỏi tài xế có muốn tiếp tục cập nhật số ghế trống cho xe không
 - b. Quay lại form cập nhật ghế trống cho xe
5. Hệ thống thông báo lỗi trong quá trình cập nhật

Bảng 2.2.1-2: Scenario use case “cập nhật số ghế trống”

Tên Use Case	Cập nhật xe
Tác nhân chính	Nhân viên hãng xe
Mức	2
Người chịu trách nhiệm	Quản lý hãng xe
Tiền điều kiện	Nhân viên đăng nhập hệ thống
Đảm bảo tối thiểu	Hệ thống loại bỏ các thao tác update thông tin của nhân viên và quay lại bước trước
Đảm bảo thành công	Thông tin về xe được update vào CSDL
Kích hoạt	Nhân viên hãng xe chọn mục cập nhật xe tại menu quản trị

Chuỗi sự kiện chính:

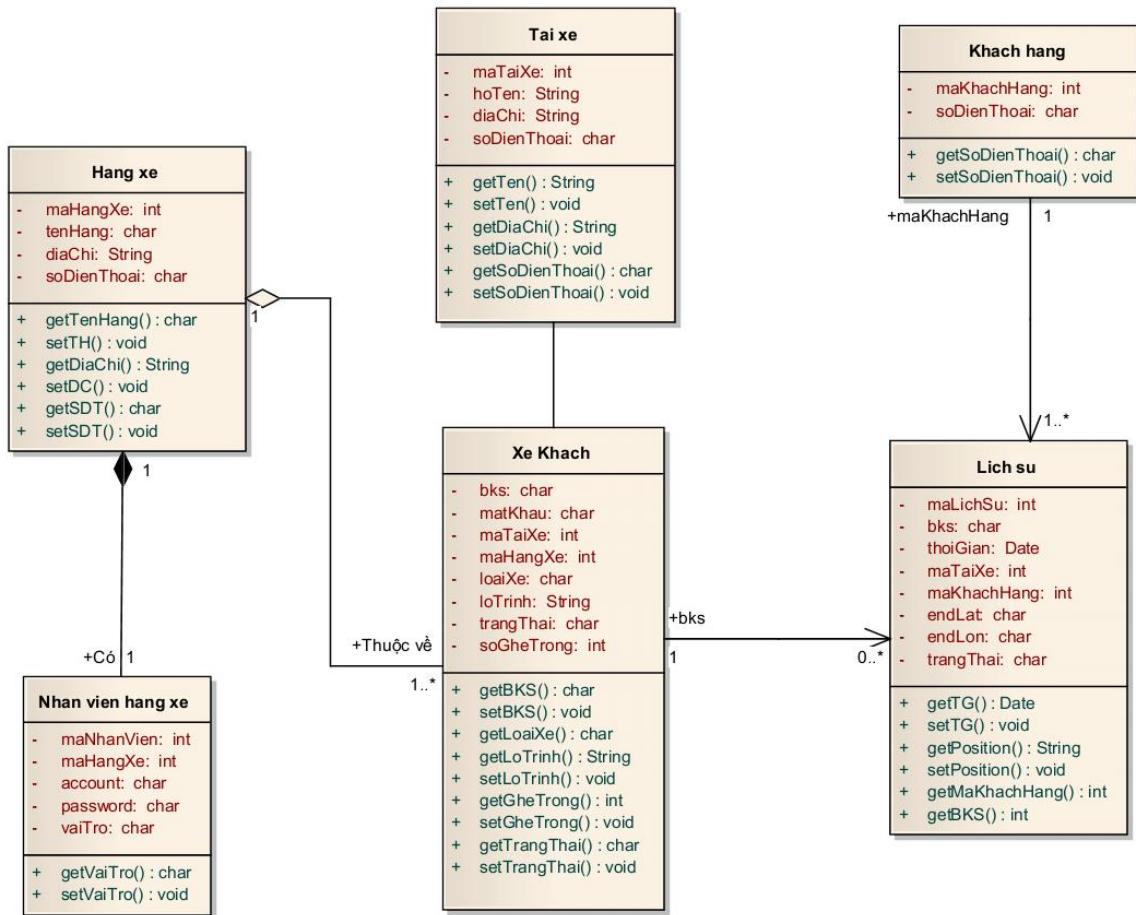
1. Hệ thống hiển thị form cập nhật xe
2. Nhân viên hệ thống nhập mới hoặc thay đổi thông tin xe nếu đã có, nếu là xóa thì chọn nút xóa xe
3. Hệ thống kiểm tra thông tin người dùng nhập và xác nhận lại vs nhân viên
4. Hệ thống thông báo đã cập nhật xe thành công
5. Nhân viên hãng xe thoát khỏi chức năng cập nhật xe

Ngoại lệ:

3. Hệ thống thông báo thao tác của người dùng không hợp lệ
 - a. Hệ thống hỏi nhân viên có tiếp tục cập nhật xe không
 - b. Nhân viên thao tác lại hoặc hủy cập nhật
4. Hệ thống thông báo quá trình cập nhật xảy ra lỗi

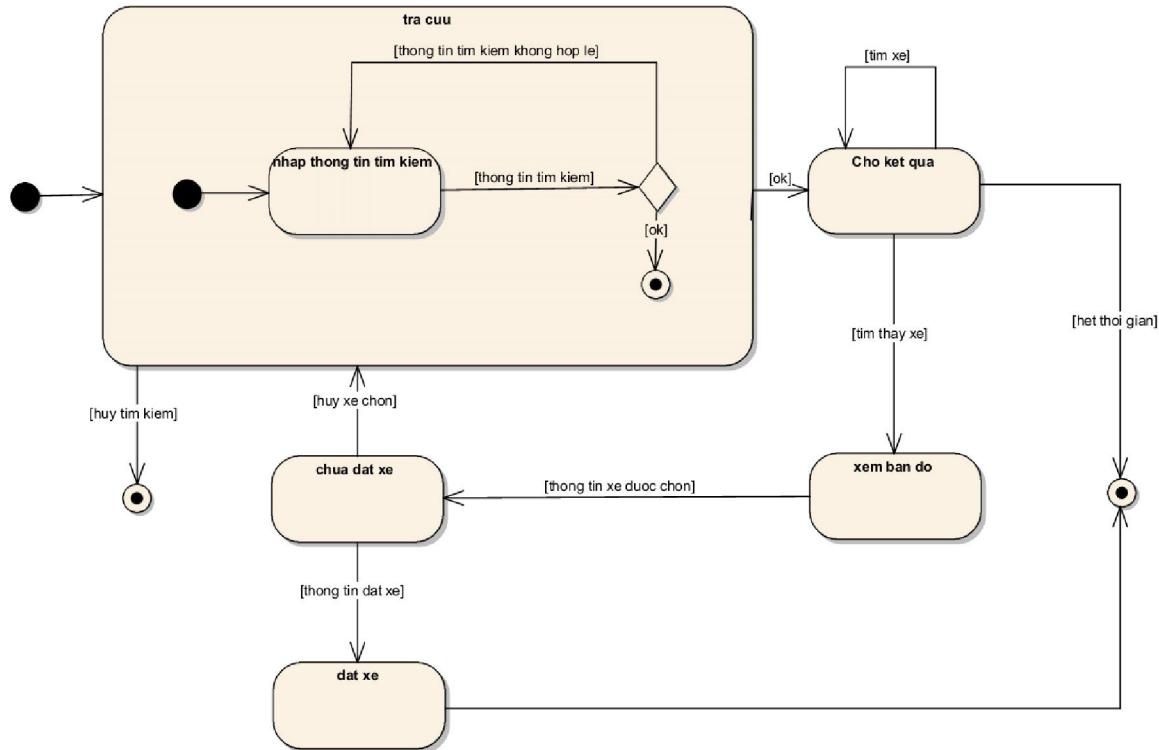
Bảng 2.2.1-3: scenario use case “cập nhật xe”

2.2.2 Biểu đồ lớp

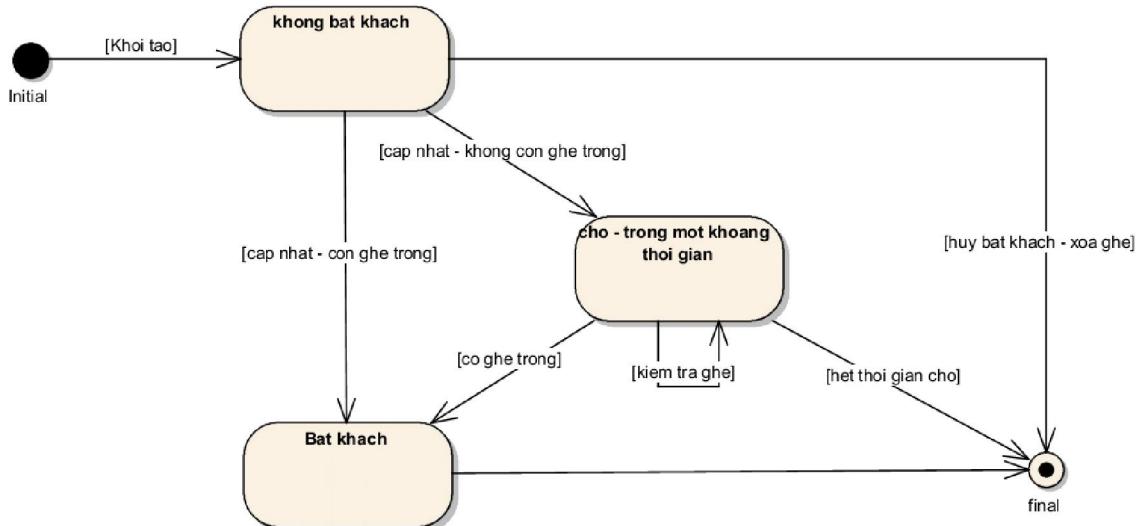


Biểu đồ 2.2.2-1: Biểu đồ lớp trong pha phân tích

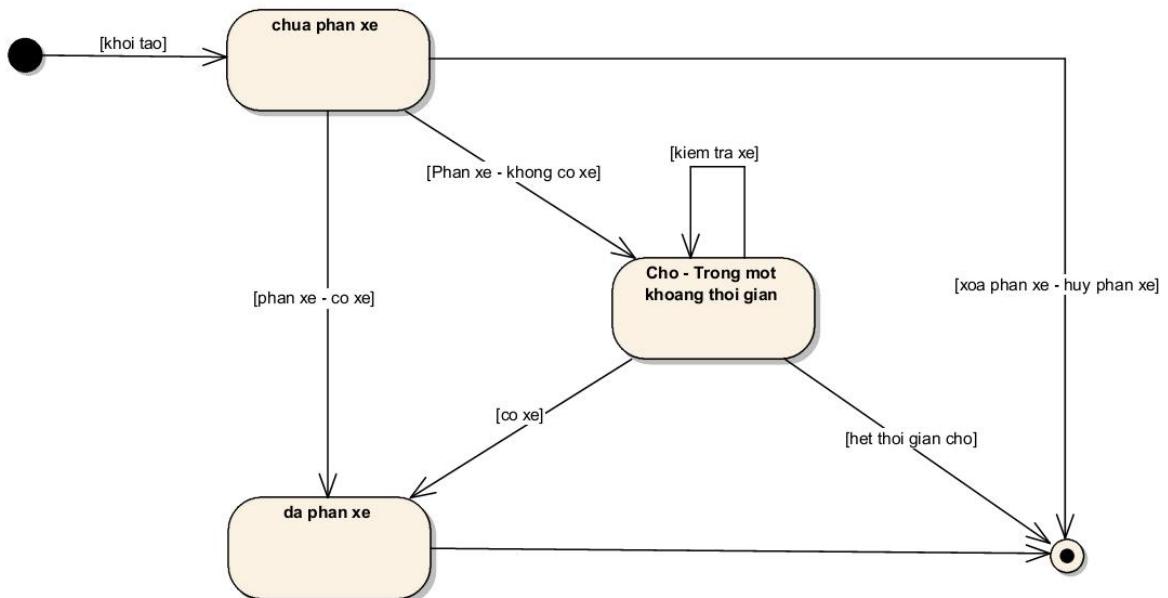
2.2.3 Biểu đồ trạng thái (State diagram)



Biểu đồ 2.2.3-1: biểu đồ trạng thái khách hàng



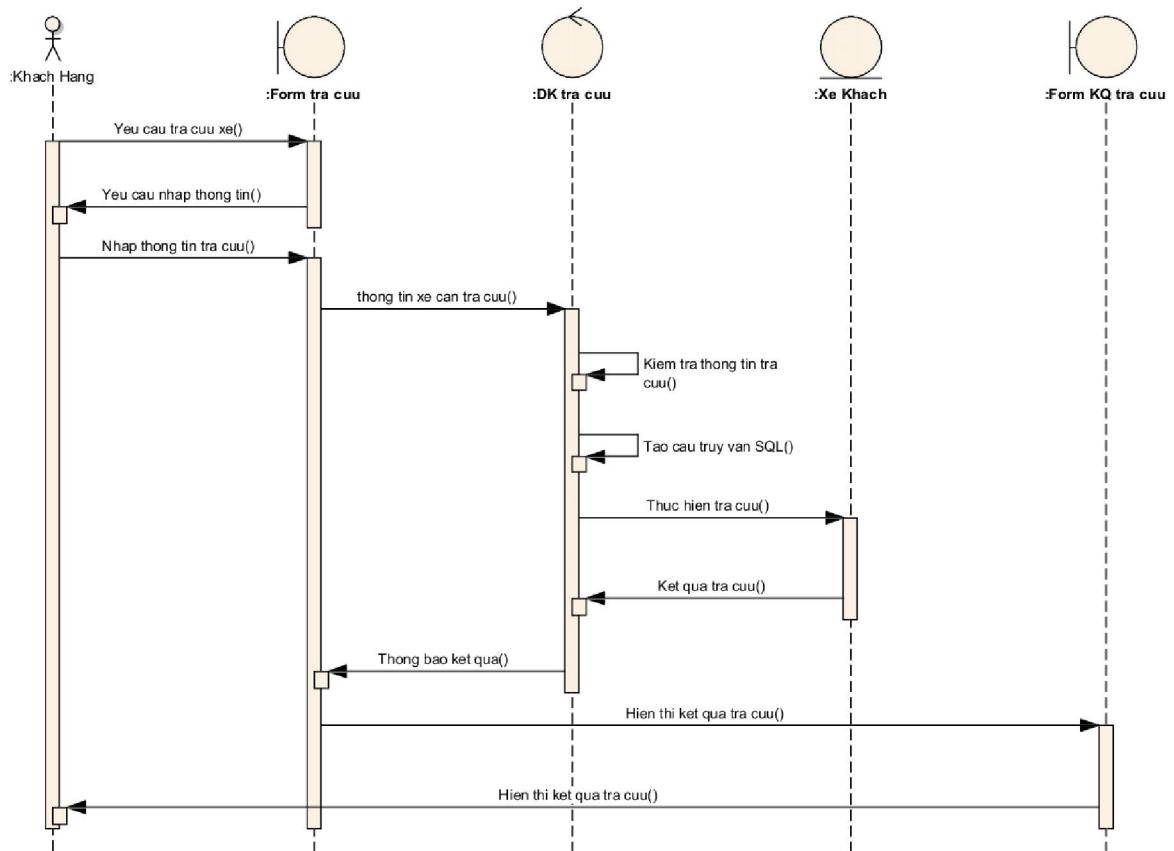
Biểu đồ 2.2.3-2: Biểu đồ trạng thái của xe khách



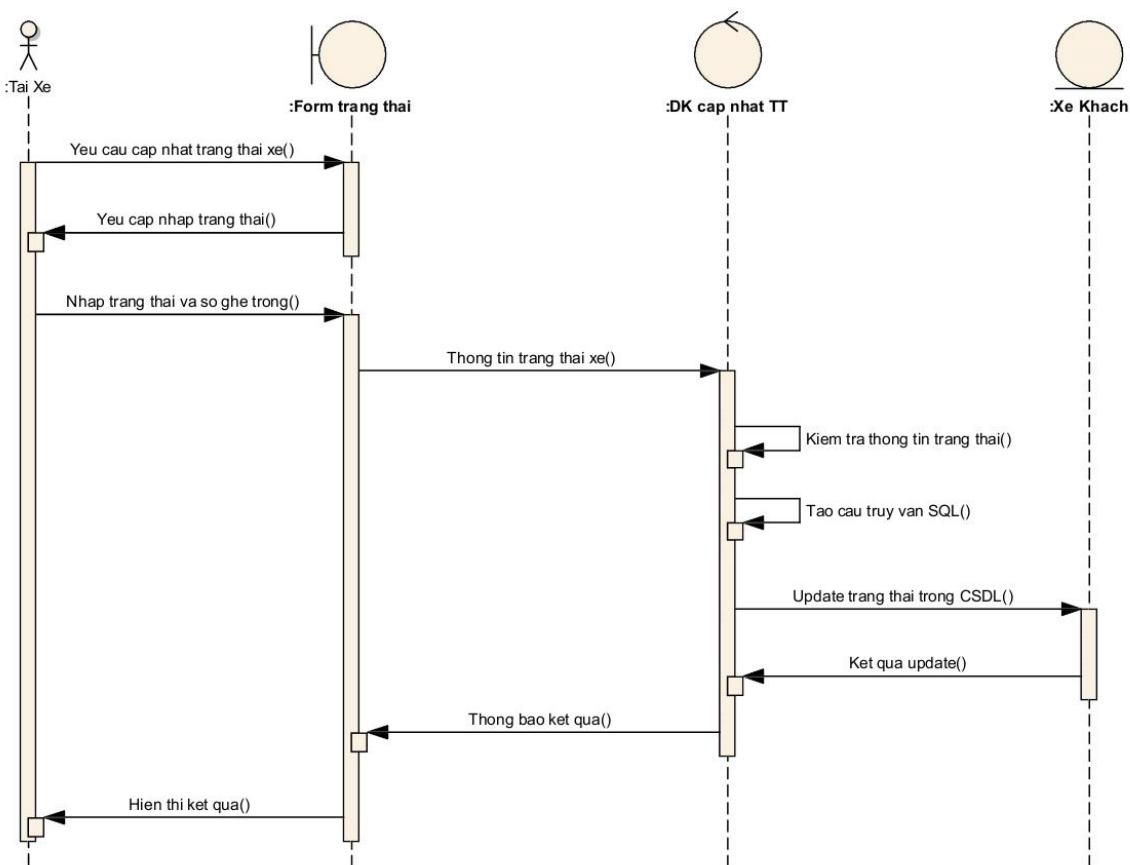
Biểu đồ 2.2.3-3: Biểu đồ trạng thái tài xế

2.3 Thiết kế hệ thống

2.3.1 Biểu đồ tuần tự (sequence diagram)

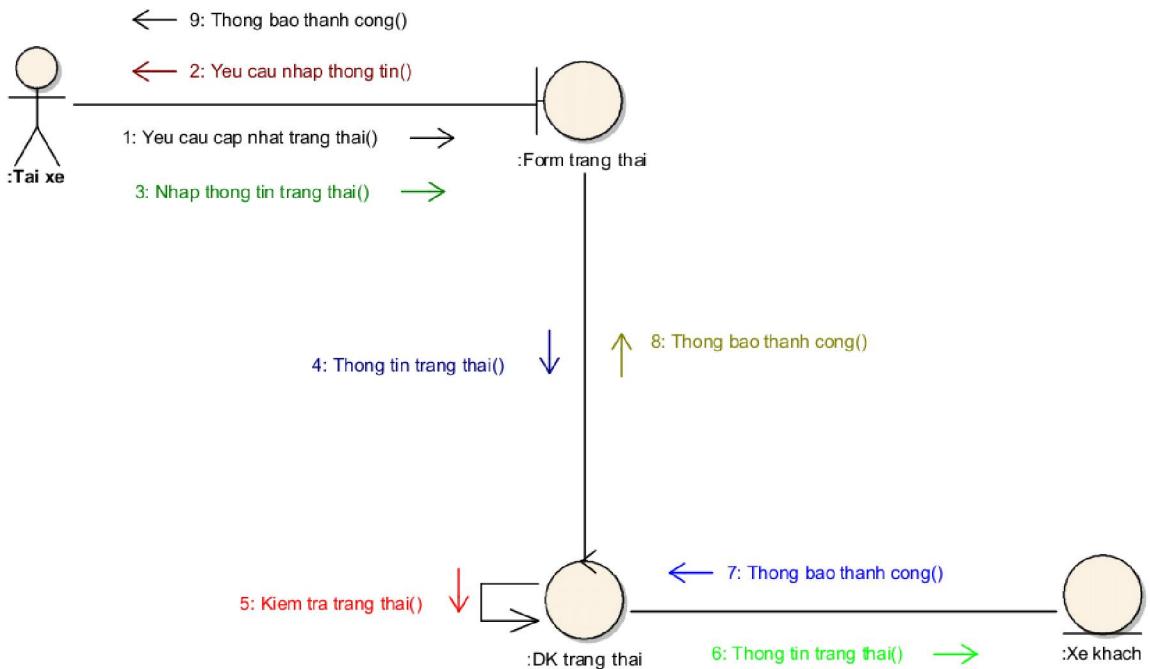


Biểu đồ 2.3.1-1: Biểu đồ tuần tự cho use case “tra cứu”

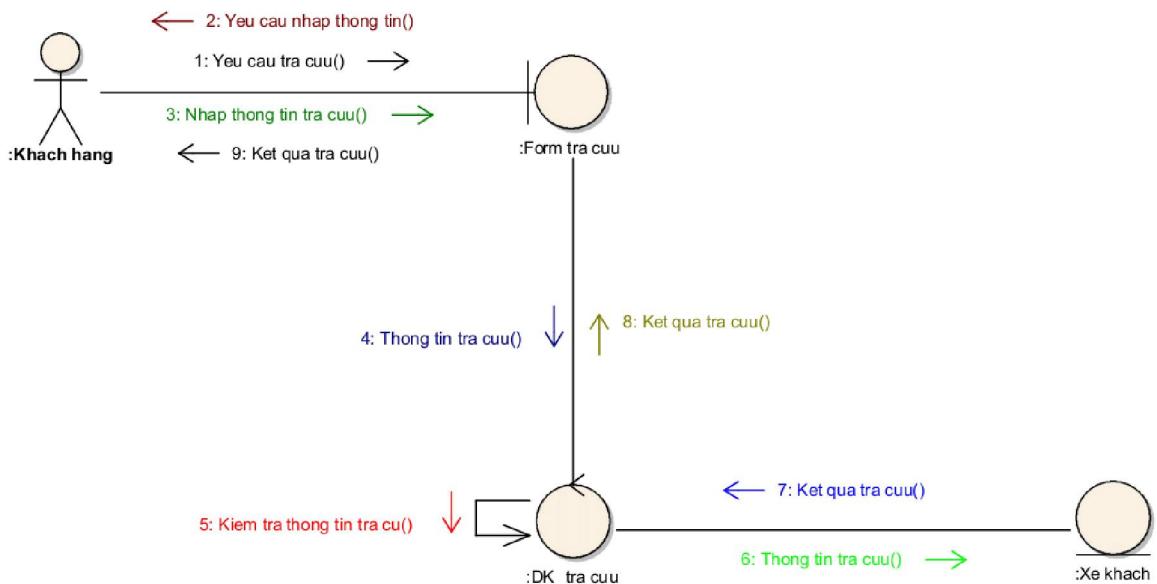


Biểu đồ 2.3.1-2: Biểu đồ tuần tự use case “cap nhat trang thai xe”

2.3.2 Biểu đồ cộng tác (colaboration Diagram)

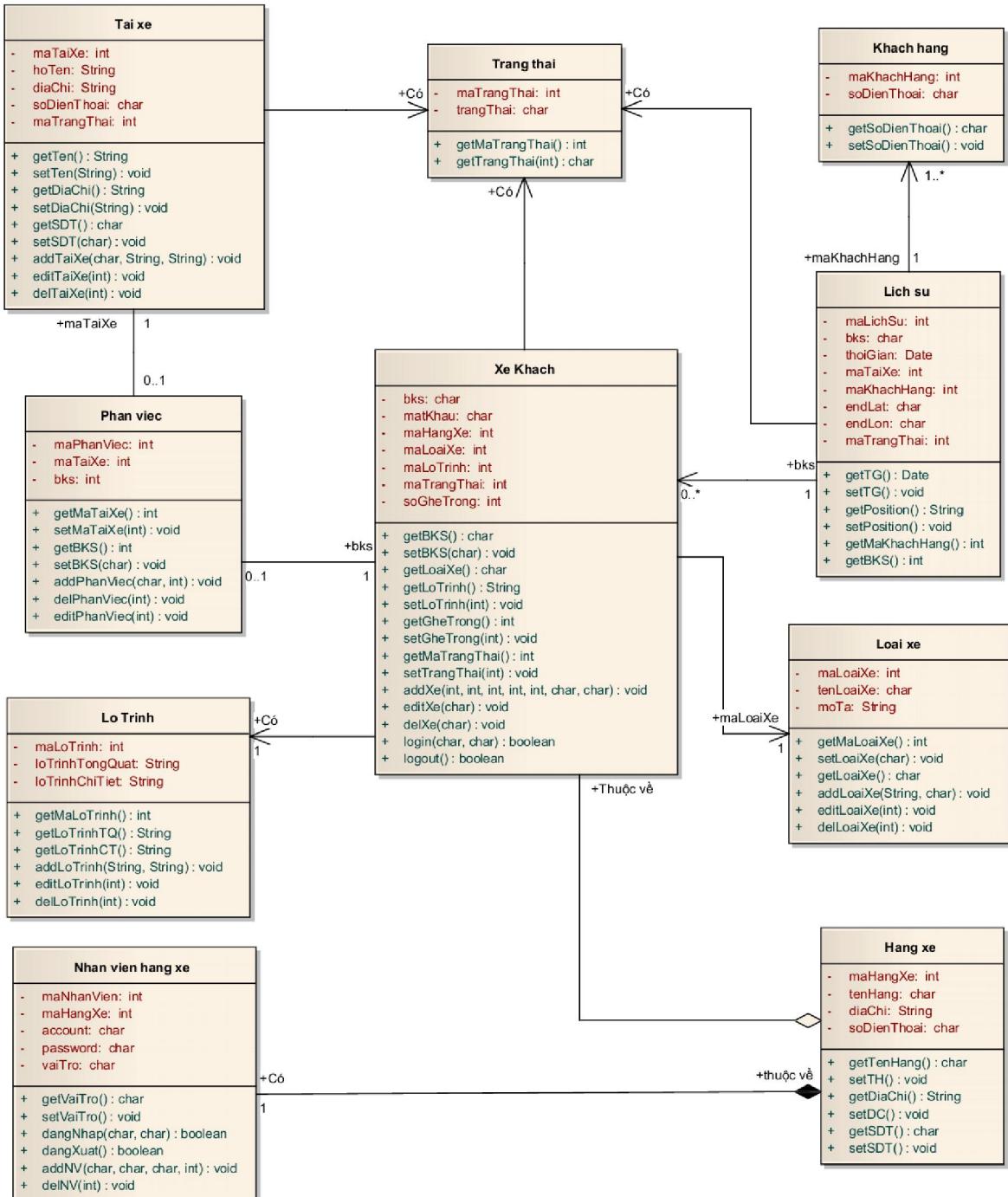


Biểu đồ 2.3.2-1: Biểu đồ cộng tác use case cap nhat trang thai xe



Biểu đồ 2.3.2-2: Biểu đồ cộng tác use case tra cứu

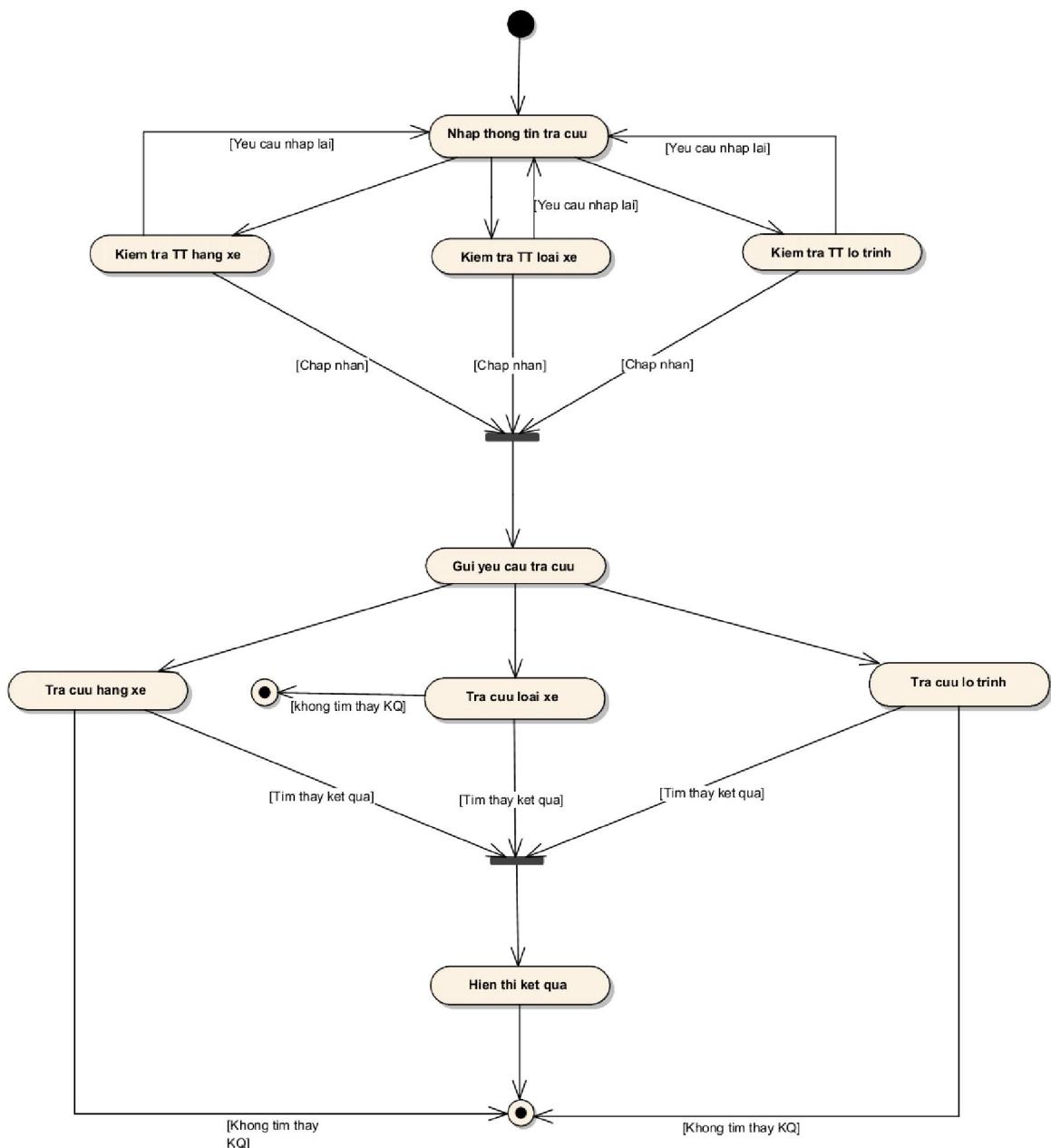
2.3.3 Biểu đồ lớp chi tiết



Biểu đồ 2.3.3-1: Biểu đồ lớp chi tiết

2.3.4 Thiết kế chi tiết

2.3.4.1 Biểu đồ hoạt động (Activity diagram)



Biểu đồ 2.3.4-1: Biểu đồ hoạt động phương thức tra cứu lớp xe khách

2.3.4.2 Xây dựng bảng thiết kế chi tiết

Hãng xe

Tên thuộc tính	Miêu tả	Kiểu	Phạm vi
<u>IDHangXe</u>	Mã số hãng. Mỗi hãng có một mã số riêng là duy nhất để phân biệt với nhau.	Int	private
TenHang	Tên hãng	Char	private
SoDienThoai	Số điện thoại hãng	Char	private
DiaChi	Địa phương hãng kinh doanh. Mỗi hãng xe thường hoạt động trên một số địa phương nhất định.	String	private

Bảng 2.3.4-1. Các thiết kế chi tiết của lớp hãng xe

Tài xế

Tên thuộc tính	Miêu tả	Kiểu	Phạm vi
IDTaiXe	Mã số tài xế. Mỗi tài xế có một mã duy nhất	Int	Private
HoTen	Họ và tên của tài xế	Char	Private
SoDienThoai	Mỗi tài xế có một số điện thoại di động riêng	Char	Private
DiaChi	Địa chỉ thường trú của tài xế	String	Private
IDTrangThai	Mã trạng thái. Trạng thái công việc của tài	Int	private

	xé. Đã phân việc hoặc chưa. Mặc định là chưa		
--	--	--	--

Bảng 2.3.4-2: Bảng thiết kế chi tiết của lớp tài xế

Xe khách

Tên thuộc tính	Miêu tả	Kiểu	Phạm vi
BKS	Biển kiểm soát hay biển số của xe. Mỗi xe có duy nhất một BKS	Char	Private
MatKhau	Mật khẩu đăng nhập của xe	Char	Private
IDHangXe	Xe thuộc hãng nào	Int	Private
IDLoaiXe	Xe thuộc loại gì	Int	Private
IDLoTrinh	Lộ trình của xe	Int	Private
IDTrangThai	Trạng thái của xe	Int	Private
SoGheTrong	Ghế trống trên xe	Int	private

Bảng 2.3.4-3: Bảng thiết kế chi tiết lớp xe khách

Loại xe

Tên thuộc tính	Miêu tả	Kiểu	Phạm vi
IDLoaiXe	Mỗi loại xe có một ID duy nhất	Int	Private
TenLoaiXe	Tên loại xe	Char	Private
MoTa	Mô tả loại xe	String	private

Bảng 2.3.4-4: Bảng thiết kế chi tiết lớp loại xe

Lộ trình

Tên thuộc tính	Miêu tả	Kiểu	Phạm vi
IDLoTrinh	Mỗi lộ trình có một ID duy nhất	Int	Private
LoTrinhTongQuat	Lộ trình tổng quát, đưa ra các điểm lớn mà xe qua.	String	Private
LoTrinhChiTiet	Lộ trình xe qua ngoài 2 điểm đầu và cuối còn thêm 7-8 điểm trung gian	String	private

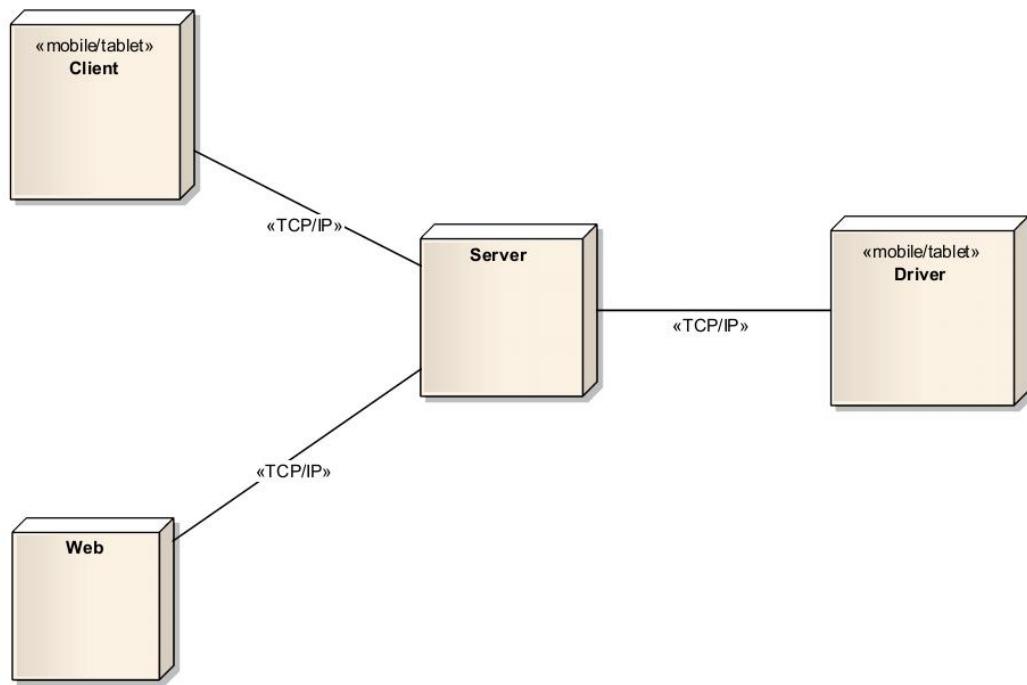
Bảng 2.3.4-5: Bảng thiết kế chi tiết lớp lộ trình

Lịch sử

Tên thuộc tính	Miêu tả	Kiểu	Phạm vi
IDLichSu	Mỗi một giao dịch, yêu cầu đặt xe của khách hàng có 1 ID duy nhất	Int	Private
BKS	Biển kiểm soát	Char	Private
Thời gian	Thời gian đặt xe	Date	Private
IDTaiXe	Tài xế	Int	Private
IDKhachHang	Khách hàng	Int	Private
endLat	Vĩ độ điểm cuối	Char	Private
endLong	Kinh độ điểm cuối	Char	Private
IDTrangThai	Trạng thái giao dịch	Int	private

Bảng 2.3.4-6: Bảng thiết kế chi tiết lớp lịch sử

2.3.5 Biểu đồ triển khai



Biểu đồ 2.3.5-1: Biểu đồ triển khai hệ thống

CHƯƠNG 3. CÀI ĐẶT VÀ TRIỂN KHAI

3.1 Các vấn đề kĩ thuật

Với hệ thống các bảng dữ liệu tương đối nhỏ và ít các trường cùng mong muốn đặt tại server online nên ta sử dụng hệ quản trị cơ sở dữ liệu MySQL cho thiết kế vì đây là một hệ thống mở tự do mà lại có chế độ bảo mật dữ liệu mạnh. Đồng thời MySQL cũng hỗ trợ tốt cho sự phát triển các ứng dụng hỗn hợp với tính linh hoạt và thực thi cao của mình.

CSDL sẽ gồm các bảng:

- ✚ QL các hãng xe
- ✚ QL các lái xe
- ✚ QL các xe khách
- ✚ QL loại xe
- ✚ QL nhân viên hãng xe
- ✚ QL phân việc cho lái xe
- ✚ QL khách hàng
- ✚ QL lịch sử
- ✚ QL trạng thái
- ✚ QL lộ trình

Hệ thống sẽ gồm 3 actors:

- ✚ Lái xe
- ✚ Khách hàng
- ✚ Quản trị viên

Code source sẽ gồm 3 gói:

- ✚ Server (dành cho những xử lý phía server)
- ✚ Client (dành cho những chức năng người dùng)
- ✚ Driver (dành cho những chức năng cho lái xe)

Công nghệ sử dụng:

- ✚ PhoneGap FrameWork

-  Android Platform
-  MySQL
-  HTML
-  CSS
-  PHP
-  Jquery
-  API Google
-  Bootstrap

Các yêu cầu hệ thống:

Phần cứng:

- SmartPhone chạy hệ điều hành Android 2.2 trở lên, có GPS...
- Server chứa dữ liệu.

Phần mềm:

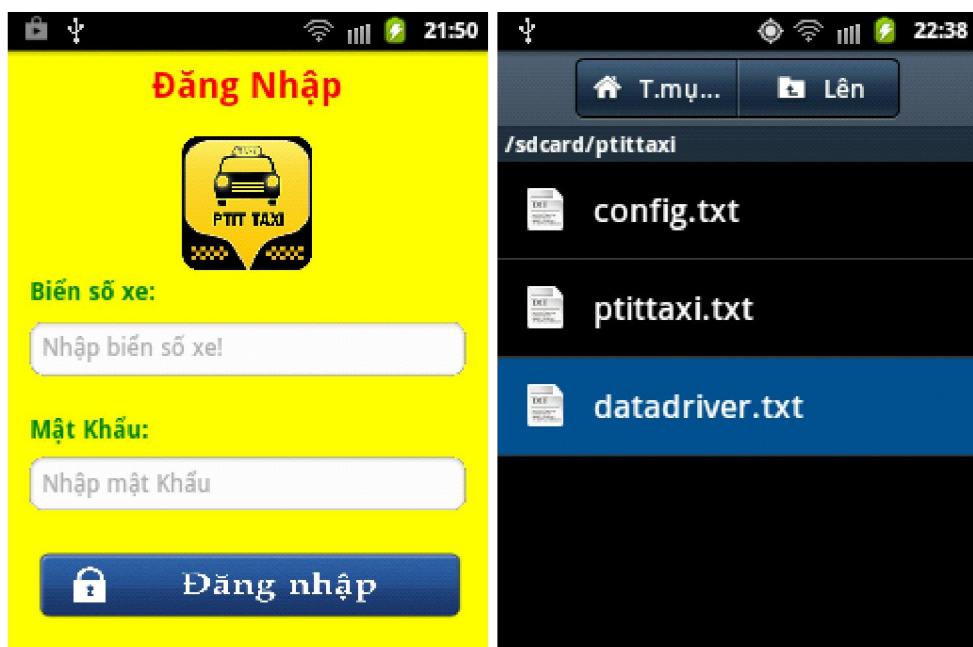
- Eclipse có cài đặt PhoneGap, Cordova, ADT,PDT...
- My SQL cài đặt phía server.
- Notepad++

3.2 Một số hình ảnh sau khi cài đặt :

3.2.1 Lái xe

Login:

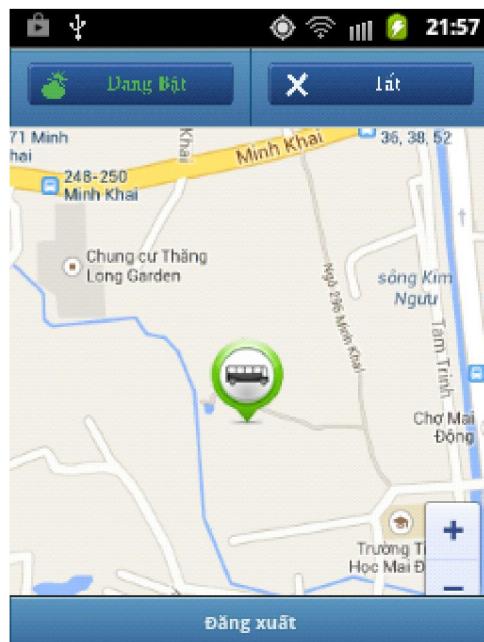
Lần đầu khi chưa login và có lịch sử login, thì trước khi sử dụng hệ thống, lái xe phải đăng nhập bằng thông tin tài khoản của chiếc xe mà tài xế đang điều khiển. Khi đăng nhập, thông tin sẽ lưu trong một file hệ thống ở trên điện thoại. Từ lần sử dụng sau tài xế không phải login lại.



Hình 3.2.1-1. Login của lái xe

Màn hình giao diện chính:

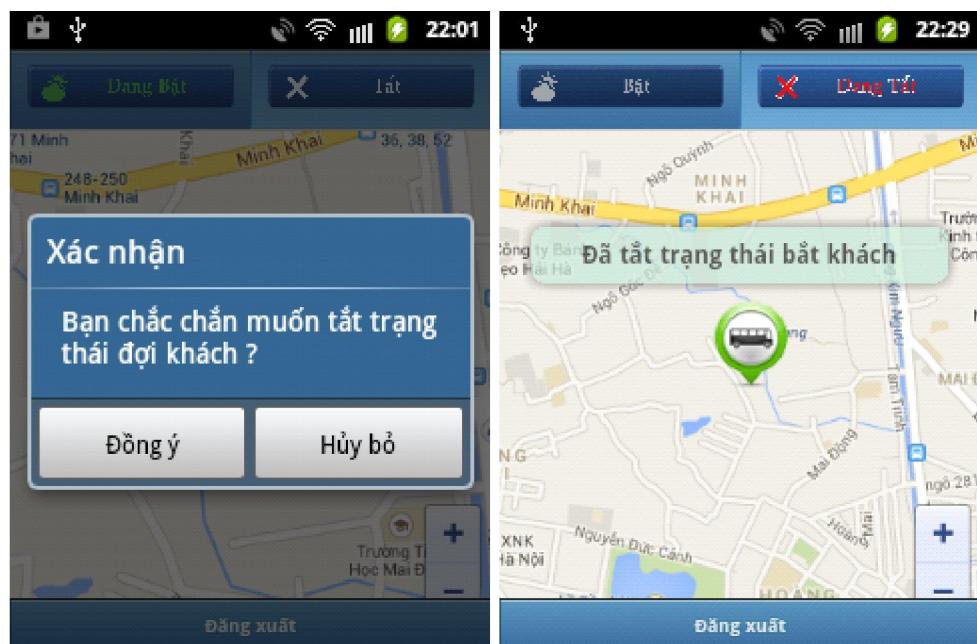
Nơi thao tác chính của tài xế bao gồm một bản đồ thể hiện vị trí của xe và form thay đổi trạng thái của xe khách



Hình 3.2.1-2. Giao diện chính cho lái xe

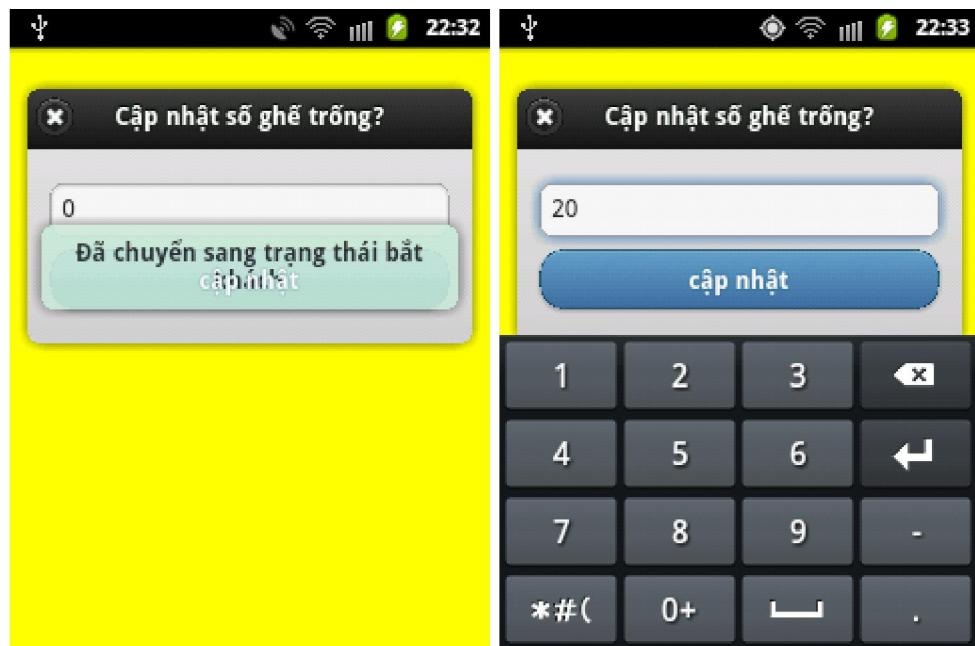
Thay đổi trạng thái “bắt khách” hoặc “không bắt khách”:

- Đầu vào: Y/c đổi trạng thái của lái xe, được kích hoạt đi khi tài xế ấn chọn button tắt (nếu trạng thái đang bật) hoặc bật (nếu trạng thái đang tắt) trên header của màn hình chính.
- Đầu ra: Trạng thái được đổi trong dữ liệu và màn hình hiển thị.
- Xử lý:
 - Kiểm tra tính đúng đắn của yêu cầu. Lái xe chỉ đang trong trạng thái bắt khách khi xe đang chạy.
 - Nếu thỏa mãn thì thay đổi trạng thái.
 - Cập nhật trạng thái mới của lái xe vào trong bảng “xe_khach”.



Hình 3.2.1-3. Form tắt trạng thái bắt khách

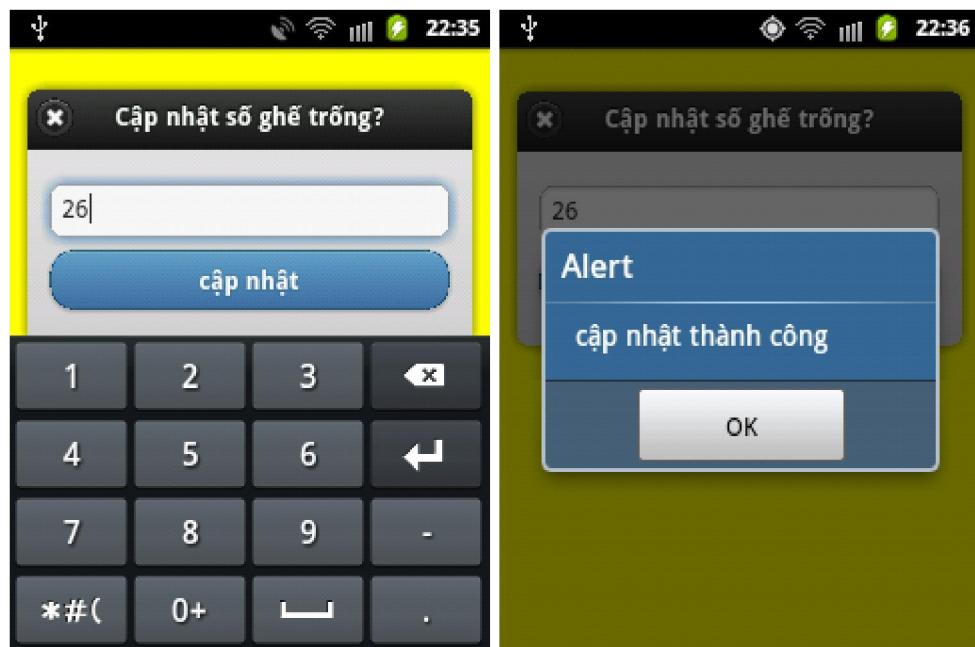
Khi bật trạng thái bắt khách, bắt buộc phải nhập số ghế còn trống trên xe:



Hình 3.2.1-4: Form bật trạng thái bắt khách

Thay đổi số ghế trống của xe:

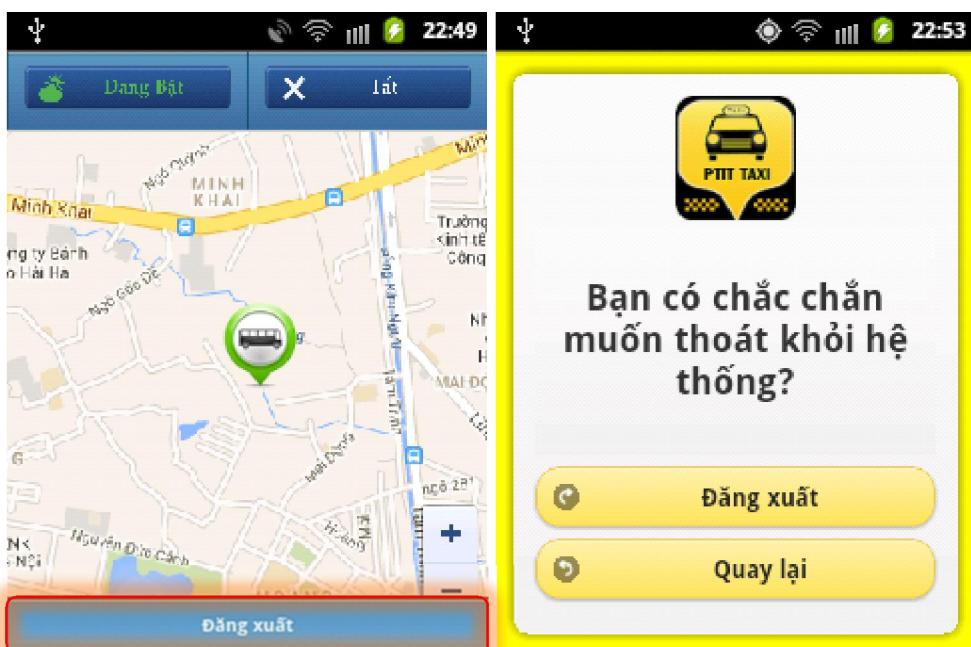
Tài xế cập nhật số ghế còn trống trên xe, tới khi nào bằng 0 thì xe chuyển trạng thái không bắt thêm khách nữa.



Hình 3.2.1-5: Giao diện cập nhật ghế trống

Đăng xuất:

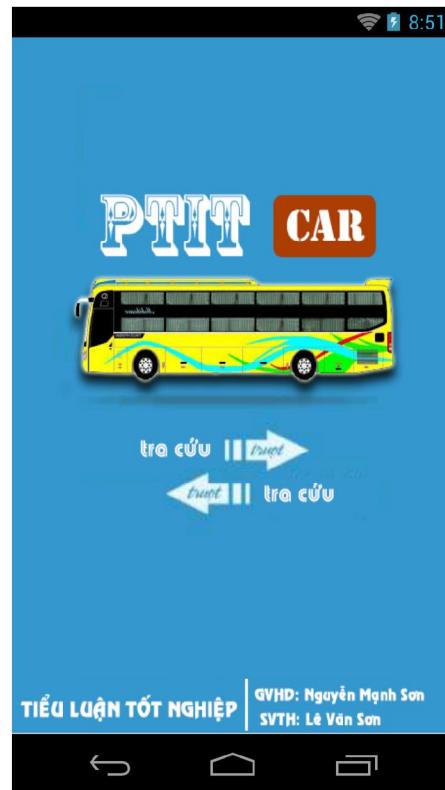
Kích hoạt khi tài xế click button đăng xuất ở footer màn hình chính



Hình 3.2.1-6: Giao diện đăng xuất của driver

3.2.2 Khách hàng

Màn hình trang chủ của app



Hình 3.2.2-1: Giao diện trang chủ

Form tra cứu thông tin

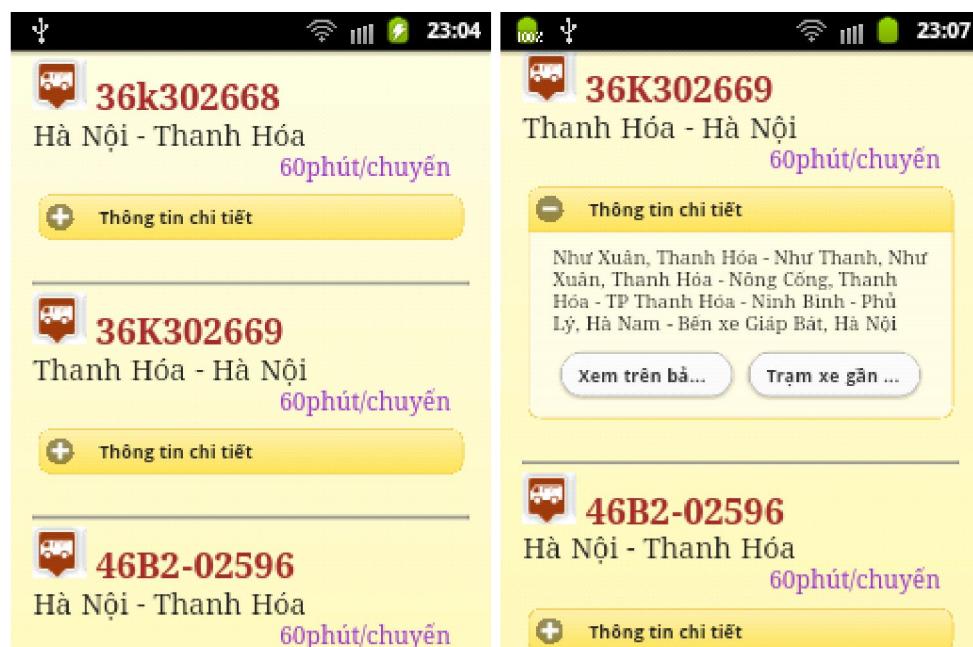
- Form này dùng để tra cứu tất cả thông tin về hãng xe, loại xe, tìm theo lộ trình xe khách.
- Input: hãng xe, loại xe theo sự lựa chọn ở đây là select box, hoặc theo lộ trình
- Output: toàn bộ thông tin về các xe khách phù hợp



Hình 3.2.2-2. Form tra cứu thông tin

Kết quả tra cứu:

Khi ấn vào nút tra cứu thông tin, nếu tìm được kết quả phù hợp, thì form thông tin về các xe phù hợp hiện ra, click nút thông tin chi tiết để xem chi tiết từng xe.



Hình 3.2.2-3: Giao diện hiển thị thông tin tra cứu

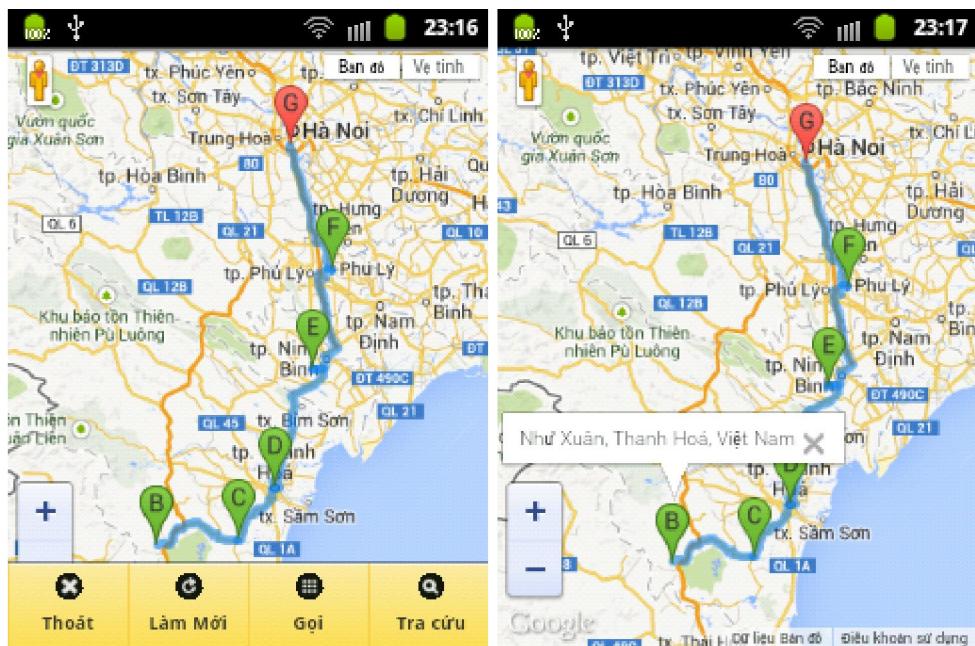
Nếu không có kết quả, hiện thông báo và quay về trang tra cứu:



Hình 3.2.2-4: Form thông báo không tìm thấy kq tra cứu

Xem lộ trình xe:

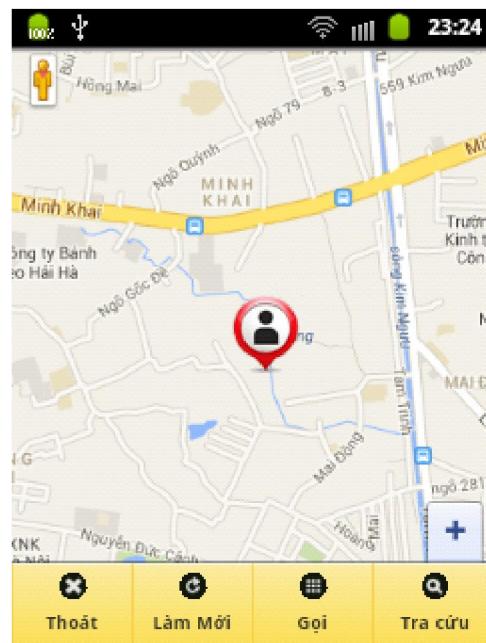
Ở giao diện xem chi tiết kết quả, nếu click nút xem bản đồ, thì bản đồ lộ trình xe đó hiển thị trực quan trên bản đồ.



Hình 3.2.2-5: Giao diện xem lộ trình xe khách

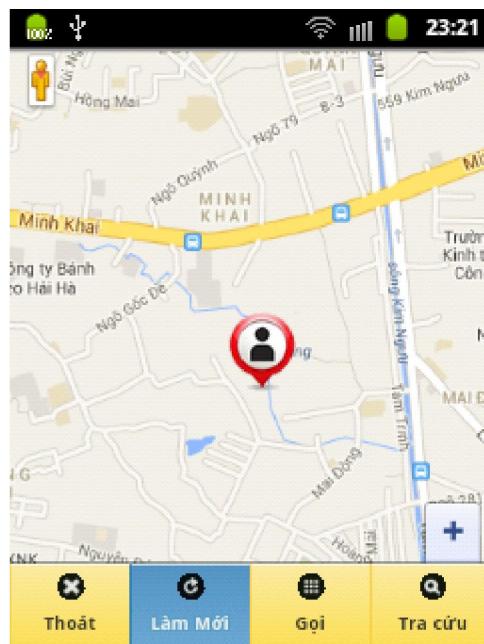
Định vị người dùng:

Với hệ thống GPS, khách hàng có thể xem trực quan vị trí của mình trên bản đồ:



Hình 3.2.2-6: Location of Client

Chọn nút “làm mới” để cập nhật lại vị trí:



Hình 3.2.2-7. Cập nhật bản đồ

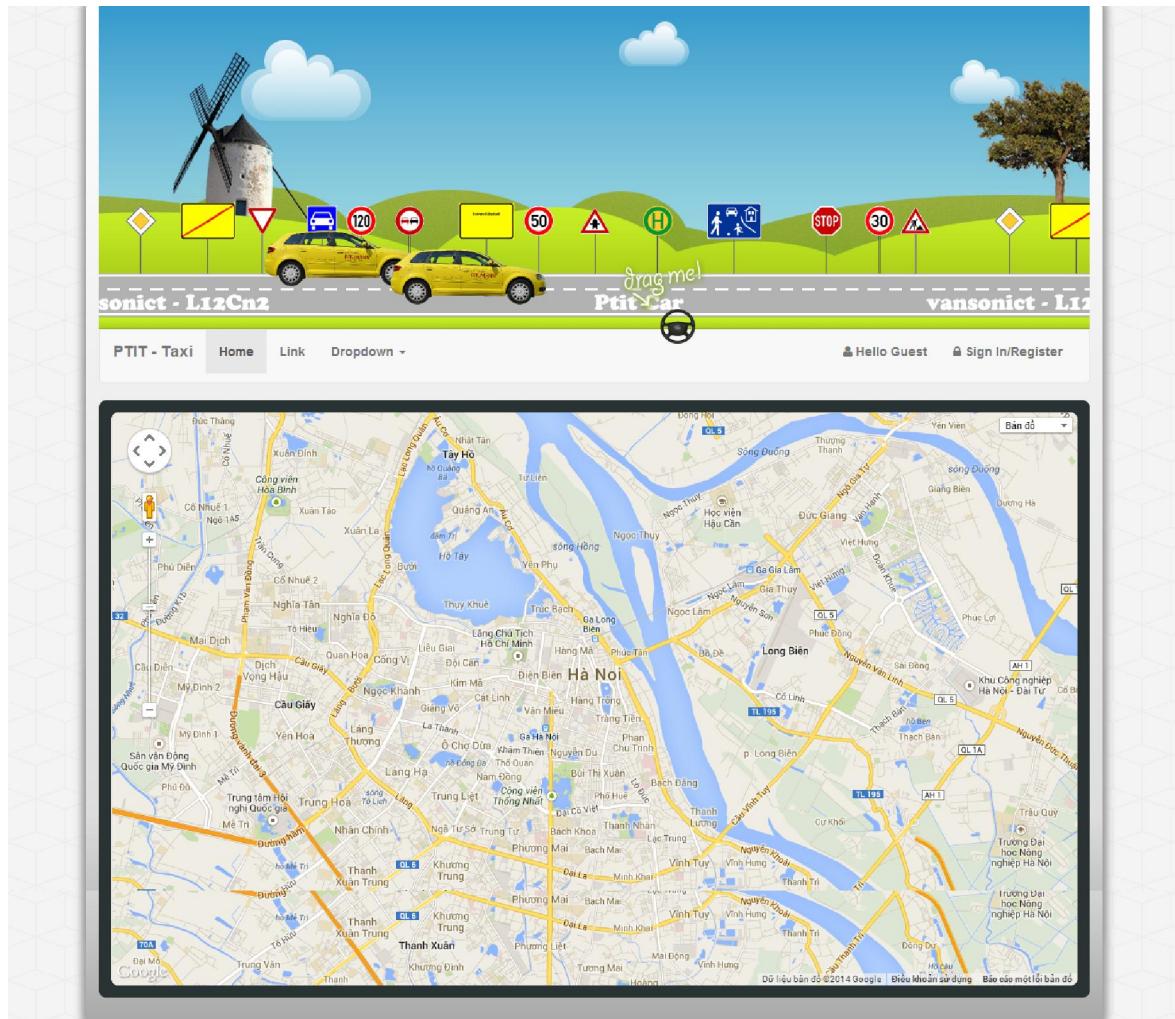


Hình 3.2.2-8. Đặt xe

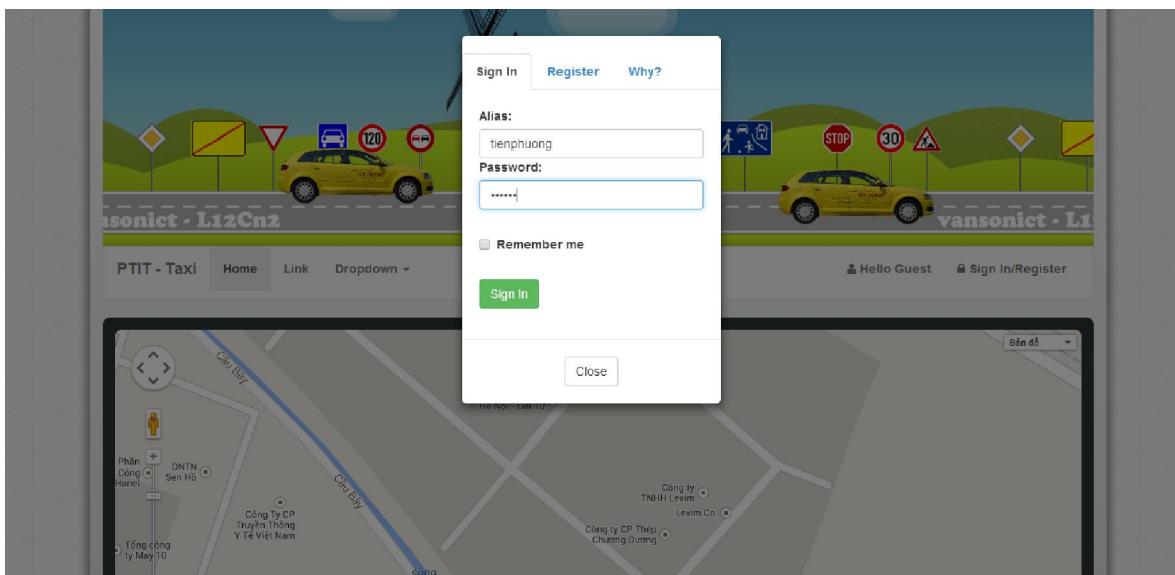


Hình 3.2.2-9. Cập nhật thông tin

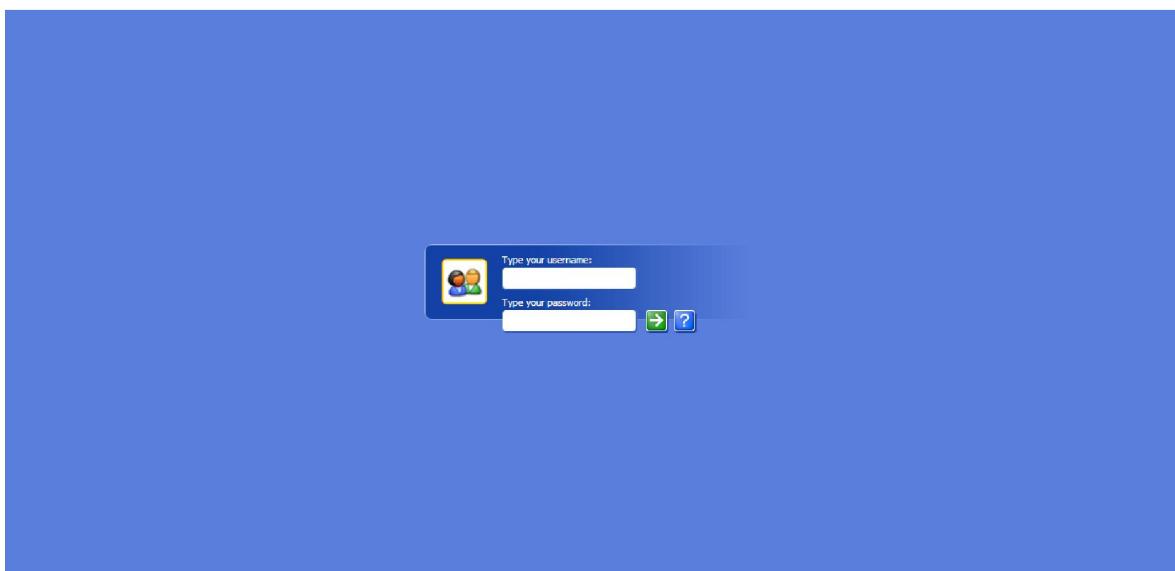
3.2.3 Server



Hình 3.2.3-1. Giao diện home của web server (QTV)



Hình 3.2.3-2. Giao diện đăng nhập nhân viên



Hình 3.2.3-3. Giao diện đăng nhập của QTV

CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau một thời gian tìm hiểu, nghiên cứu và thực hiện em đã cơ bản hoàn thành những mục tiêu của đề tài tiểu luận tốt nghiệp:

Tìm hiểu về công nghệ PhoneGap, đặc điểm, chức năng và ứng dụng của nó. Áp dụng được công nghệ PhoneGap trên nền tảng Android.

Thực hiện khảo sát dịch vụ xe khách hiện tại tại bến xe Giáp Bát, Mỹ Đình, ven tuyến QL1A, đường Hồ Chí Minh từ Hà Nội tới Thanh Hóa, rút ra những ưu nhược điểm còn tồn tại. Từ đó xác định mục tiêu và yêu cầu cần đạt được ở hệ thống mới.

Tiến hành phân tích thiết kế hệ thống và xây dựng cơ sở dữ liệu cho ứng dụng “Đón xe khách” (Ptit-taxi). Với các chức năng tra cứu, đặt xe, cập nhật thông tin... trực quan, thân thiện thích hợp với nhiều đối tượng. Sau đó, em đã tiến hành cài đặt và thực nghiệm ứng dụng với kết quả tương đối khả quan.

Tuy nhiên với khoảng thời gian có hạn và kiến thức còn ít ỏi đề tài nghiên cứu của em vẫn còn tồn tại một số nhược điểm hạn chế cần khắc phục: ứng dụng còn chậm, cơ sở dữ liệu nhỏ chỉ mang tính chất kiểm thử, phụ thuộc nhiều vào API của Google...

Hướng phát triển của ứng dụng mà em đặt ra muôn hướng tới là tiếp tục cải thiện nâng cao hơn nữa chất lượng sử dụng, mở rộng ra các loại hình khác như taxi và đưa ứng dụng vào thực tế trên nhiều nền tảng khác nhau.

Để đạt được những điều trên, em rất cảm ơn sự tận tình chỉ bảo của thầy cũng như sự giúp đỡ của các bạn cùng học. Em rất mong sẽ được nhận nhiều hơn nữa những ý kiến đóng góp của thầy và các bạn.

Em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

- [1] Dr.Jawahar, OverView of System Analysis & Design.
- [2] Trần Đình Quέ & Nguyễn Mạnh Sơn, Phân tích và thiết kế hệ thống thông tin, Học viện Bưu chính viễn thông
- [3] Nguyễn Văn Ba , Phân tích và thiết kế hệ thống thông tin, NXB Đại học Quốc gia Hà Nội năm 2006.
- [4] Th.S Nguyễn Hồng Phương , tài liệu CSDL_IT3090_handout về dữ liệu.
- [5] Các bài viết liên quan đến PhoneGap được tham khảo tại địa chỉ <http://docs.phonegap.com/guides>.
- [6] Các bài viết, hướng dẫn về ngôn ngữ lập trình android tại <http://developer.android.com>
- [7] Các bài viết, hướng dẫn về google maps api tại
<https://google-developers.appspot.com/maps/>,
<https://code.google.com/p/jquery-ui-map/>,
<https://developers.google.com/maps/>
- [8] Các bài viết, hướng dẫn về Jquery mobile tại
<http://jquerymobile.com/>
- [9] Các bài viết, hướng dẫn về Bootstrap tại
<http://getbootstrap.com/getting-started/>
- [10] Các bài viết, hướng dẫn về Spritely tại
<http://spritely.net/documentation/>