ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA CÔNG NGHỆ THÔNG TIN

Tel. (84-511) 3736949, Fax. (84-511) 3842771

Website: [itf.dut.edu.vn](mailto:itf.ud.edu.vn), E-mail: cntt@dut.udn.vn



ĐỒ ÁN TỐT NGHIỆP KỸ SƯ

NGÀNH CÔNG NGHỆ THÔNG TIN

MÃ NGÀNH: D480201

**ĐỀ TÀI :**

XÂY DỤNG ỨNG DỤNG HỔ TRỢ XE MÁY TRÊN NỀN TẢNG ANDROID

SINH VIÊN : LÊ VIẾT NHỰT

MÃ SINH VIÊN : 102120108

LỚP : 12T1

CBHD : TS. TRƯƠNG NGỌC CHÂU

Đà Nẵng, 06/2016

LỜI CẢM ƠN

Em xin chân thành cảm ơn thầy TS.Trương Ngọc Châu.- người trực tiếp hướng dẫn và giúp em rất nhiều để hoàn thành đồ án tốt nghiệp. Em cũng xin gởi lời cảm ơn đến toàn thể thầy cô giáo của đã truyền đạt kiến thức suốt 5 năm qua để e có kiến thức để thực hiện tốt đồ án tốt nghiệp lần này.

Em xin cảm ơn !

**Họ và tên sinh viên**

**Lê Viết Nhựt**

**LỜI CAM ĐOAN**

*Em xin cam đoan :*

1. *Nội dung trong luận văn này là do tôi thực hiện dưới sự hướng dẫn trực tiếp của TS.Trương Ngọc Châu.*
2. *Các tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.*
3. *Nếu có những sao chép không hợp lệ, vi phạm, em xin chịu hoàn toàn trách nhiệm.*

*Sinh viên thực hiện*

*Lê Viết Nhựt*

**NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN**

.............................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................. ..........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN

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

**MỤC LỤC**

[**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT** 13](#_Toc482798892)

[**1.1.** **LÝ THUYẾT ANDROID** 13](#_Toc482798893)

[***1.1.1.*** ***Lịch sử Android*** 13](#_Toc482798894)

[***1.1.2.*** ***Kiến trúc Android*** 14](#_Toc482798895)

[***1.1.3.*** ***Các kiến thức trong lập trình Android*** 15](#_Toc482798896)

[**1.2.** **LÝ THUYẾT NODEJS, MONGODB** 18](#_Toc482798897)

[***1.2.1.*** ***NodeJS là gì?*** 18](#_Toc482798898)

[***1.2.2.*** ***Đặc tính của NodeJS*** 18](#_Toc482798899)

[***1.2.3.*** ***MongoDB*** 19](#_Toc482798900)

[**1.3.** **PHÁT BIỂU BÀI TOÁN** 20](#_Toc482798901)

[**1.4.** **KẾT CHƯƠNG** 20](#_Toc482798902)

[**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG** 21](#_Toc482798903)

[**2.1.** **GIỚI THIỆU** 21](#_Toc482798904)

[**2.2.** **PHÂN TÍCH HIỆN TRẠNG** 21](#_Toc482798905)

[**2.3.** **PHÂN TÍCH YÊU CẦU** 21](#_Toc482798906)

[***2.3.1.*** ***Đối tượng sử dụng*** 21](#_Toc482798907)

[***2.3.2.*** ***Các yêu cầu để xây dựng hệ thống hoàn thiện*** 21](#_Toc482798908)

[***2.3.3.*** ***Công nghệ sử dụng*** 22](#_Toc482798909)

[**2.4.** **THIẾT KẾ HỆ THỐNG** 22](#_Toc482798910)

[***2.4.1.*** ***Phân tích chức năng hệ thống*** 22](#_Toc482798911)

[***2.4.2.*** ***Các biểu đồ*** 28](#_Toc482798912)

[***2.4.3.*** ***Thiết kế cơ sở dữ liệu*** 32](#_Toc482798913)

[**2.5.** **KẾT CHƯƠNG** 35](#_Toc482798914)

[**CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ** 36](#_Toc482798915)

[**3.1.** **MÔ HÌNH TRIỂN KHAI** 36](#_Toc482798916)

[***3.1.1.*** ***Mô hình triển khai*** 36](#_Toc482798917)

[***3.1.2.*** ***Công cụ hổ trợ*** 36](#_Toc482798918)

[**3.2.** **KẾT QUẢ CHƯƠNG TRÌNH** 37](#_Toc482798919)

[***3.2.1.*** ***Màn hình đăng nhập, đăng ký*** 37](#_Toc482798920)

[***3.2.2.*** ***Màn hình chính của ứng dụng*** 37](#_Toc482798921)

[***3.2.3.*** ***Màn hình tìm kiếm địa điểm theo vị trí hiện tại và địa điểm bất kỳ trên google maps***…………………………………………………………………………………….38](#_Toc482798922)

[***3.2.4.*** ***Màn hình hiển thị địa điểm đã được tìm kiếm, vẽ đường đi và tính khoảng cách thời gian, khoảng cách*** 38](#_Toc482798923)

[***3.2.5.*** ***Màn hình chọn một tài xế để thực hiện chuyến đi*** 39](#_Toc482798924)

[***3.2.6.*** ***Màn hình thông tin tài xế được chọn*** 39](#_Toc482798925)

[***3.2.7.*** ***Màn hình hiển thị lịch sử chuyến đi của người dùng*** 40](#_Toc482798926)

[***3.2.8.*** ***Màn hình hiển thị đánh giá chuyến đi của người dùng*** 40](#_Toc482798927)

[***3.2.9.*** ***Màn hình hiển thị tài xe được yêu thích của người dùng*** 41](#_Toc482798928)

[***3.2.10.*** ***Màn hình hiển thị thông tin cần thiết để trở thành tài xế*** 41](#_Toc482798929)

[**3.2.11.** ***Màn hình hiển thị địa điểm sửa chữa xe máy*** 42](#_Toc482798930)

[***3.2.12.*** ***Màn hình đăng ký địa điểm sửa chữa xe máy*** 42](#_Toc482798931)

[***3.2.13.*** ***Màn hình hiển thị phản hồi, đóng góp ý kiến*** 43](#_Toc482798932)

[**3.3.** **NHẬN XÉT ĐÁNH GIÁ KẾT QUẢ** 43](#_Toc482798933)

[**3.4.** **KẾT CHƯƠNG** 43](#_Toc482798934)

**DANH SÁCH HÌNH ẢNH**

**DANH SÁCH TỪ VIẾT TẮC**

**MỞ ĐẦU**

1. **TỔNG QUAN ĐỀ TÀI**

Trong thời đại công nghệ thông tin đang phát triển như hiện nay thì số lượng người sử dụng smartphone ngày càng gia tăng. Qua đó nhu cầu người dùng cũng tăng theo, do đó yêu cầu nhà phát triển phải biết đáp ứng như được nhu cầu của người dùng.

Dựa vào nhu cầu người dùng, em đã chọn đề liên quan đến một dịch vụ đã có từ lâu, đó chính là dịch vụ xe ôm. Xe ôm – một hình thức chuyên chở khá phổ biến và rất Việt Nam. Người hành nghề xe ôm trước đây thường là những người nghèo, không có được công việc lao động cụ thể, họ thường tập trung ở những nơi đông đúc như bến xe, trường học, chợ để đưa đón khách. Tuy nhiên dịch vụ xe ôm này thường đi kèm một số mặt tiêu cực như là:

* Khách đi xe ôm hay bị chủ xe chặt chém về giá cả.
* Đa số người không thích cách giành dựt khách giữa những người hành nghề xe ôm với nhau
* Một bộ bận xe ôm có thái độ phục vụ không tốt đối với khách.

Từ những mặt tiêu cực còn tồn đọng trên, em đã nảy ra ý tưởng xây dựng nên một hệ thống xe ôm thông minh. Vậy hệ thống xe ôm thông minh có điểm gì hay ?

**Thứ nhất**, khách hàng không phải lo bị các tài xế xe ôm chặt chém về giá, vì trước khi thực hiện chuyến đi của mình, hệ thống xe thông báo cho khách hàng biết giá tiền mà mình cần phải trả cho chuyến đi đó theo đúng bảng giá đã được quy định.

**Thứ hai**, khách hàng có thể dự do lựa chọn tài xế xe ôm nào sẽ cũng mình thực hiện chuyến đi đó. Không xảy ra tình trạng tranh giành khách hàng giữa những tài xế với nhau

**Thứ ba**, khách hàng sẽ nhận được sự phục vụ rất chuyên nghiệp từ các tài xế. Đa phần tài xế trong hệ thống xe ôm thông minh là những người ăn mặc lịch sự, thái độ phục vụ tốt và đi kèm với xe xịn.

**Thứ tư**, từ vai trò khách hàng sử dụng dịch vụ dến việc trở thành tài xê của hệ thống xe ôm thông cũng cũng cực gì đơn giản. Góp phần tăng thu nhập cá nhân.

Hãy đến với hệ thống xe ôm của chúng tôi, bạn sẽ được di chuyển khắp mọi ngõ ngách của thành phố một cách an toàn nhất, chưa kể hạn chế được tình trạng kẹt xe như hiện nay.

Ngoài việc xây dụng một hệ thống xe ôm thông minh, trong ứng dụng chúng tôi đã tích hợp thêm cho người dùng có thể tìm kiếm địa điểm sửa xe máy đang hoạt động mà gần với người dùng nhất.

Đề tài đồ án của em có tên *“Xây dựng ứng dụng hổ trợ xe máy Sbike trên nền tảng Android”*

1. **MỤC ĐÍCH Ý NGHĨA ĐỀ TÀI**
   1. ***Mục đích***

Mục đích của đề tài là em muốn phát tạo nên một ứng dụng kết nối người dùng lại với nhau trên một nền tảng di động khá là hot hiện nay. Qua đó sẽ giúp em nâng cao khả năng tư suy, sáng tạo và nghiên cứu những công nghệ mới đáp ứng yêu cầu đề tài đặt ra.

* 1. ***Ý nghĩa***

Đề tài phát triển nhằm tạo ra một hệ thống thông minh, đáp ứng được yêu cầu thực tiễn cũng như nhu cầu ngày càng cao của người dùng. Nếu đề tài được phổ biến rộng rãi em tin chắc sẽ góp phần làm xã hội trở nên văn minh, năng động hơn.

1. **PHƯƠNG PHÁP THỰC HIỆN**
   1. ***Khảo sát thực tế***

Thực tế cho thấy rằng nhu cầu của người dùng ngày một tăng cao, các dịch vụ yêu cầu phải được thực hiện chuyên nghiệp, an toàn và có chất lượng tốt. Do đó yêu cầu các dịch vụ phải đáp ứng được những nhu cầu người dùng đặt ra nếu không muốn bị đào thải.

* 1. ***Khảo sát lý thuyết***

Trong quá trình nghiên cứu và tìm hiểu, em thấy cần phải khảo sát một số vấn đề lý thuyết để phục vụ cho đề tài

* Nắm vững ngôn ngữ lập trình Java
* Nắm vững kiến thức lập trình Android
* Hiểu được cách hoạt động của mô hình Clien/Server cũng như xây dựng một webservice
* Hiểu được quy tắc lập trình NodeJS, kết hợp với CSDL MongoDB
* Nắm vững yêu cầu và cách thức để phân tích và thiết kế hệ thống
* Cài đặt và triển trai ứng dụng trên nền tảng di động cũng như web.
  1. ***Phạm vi đề tà***

Đề tài được phát triển trên nền tảng Android 5

1. **BỐ CỤC CỦA ĐỒ ÁN**

Đồ án được chia làm các phần như sau:

**Mở đầu** : Giới thiệu tổng quan đề tài, mục đích, tính cấp thiết và ý nghĩa thực tiễn của đề tài. Trình bày các phương pháp để thực hiện đề tài như là: khảo sát thực tiễn, khảo sát lý thuyết và phạm vi đề tài.

**Chương 1** : Mô tả các cơ sở lý thuyết, kiến thức nền tảng chung của các công nghê sẽ được áp dụng để xây dụng ứng dụng thỏa mãn yếu cầu đề tài.

**Chương 2** : Trình bày phương pháp phân tích và thiết kế hệ thống. Như là phân tích các chức năng chính, các biểu đồ use case, biểu đồ lớp và biểu đồ tuần tự cũng như việc thiết kế cơ sở dữ liệu

**Chương 3 :** Trình bày các mô hinh triển khai, các công nghê và công cụ được sử dụng để xây dựng hệ thống. Kết thúc chương là danh sách hình ảnh giao diện của ứng dụng.,

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

* 1. **LÝ THUYẾT ANDROID**
     1. ***Lịch sử Android***

Ban đầu, Android là hệ điều hành cho các thiết bị cầm tay dựa trên lõi Linux do công ty Android Inc. (California, Mỹ) thiết kế. Công ty này sau đó được Google mua lại vào năm 2005 và bắt đầu xây dựng Android Platform. Các thành viên chủ chốt ở Android Inc. gồm có: Andy Rubin, Rich Miner, Nick Sears, and Chris White..

****

Hinh 1. Adroid timeline

Và sau tiếp, vào cuối năm 2007, thuộc về Liên minh Thiết bị Cầm tay Mã Nguồn mở (Open Handset Alliance) gồm các thành viên nổi bật trong ngành viễn thông và thiết bị cầm tay như:

Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, T-Mobile, ARM Holdings, Atheros Communications, Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, and Vodafone Group,… Mục tiêu của Liên minh này là nhanh chóng đổi mới để đáp ứng tốt hơn cho nhu cầu người tiêu dùng và kết quả đầu tiên của nó chính là nền tảng Android. Android được thiết kế để phục vụ nhu cầu của các nhà sản xuất thiết, các nhà khai thác và các lập trình viên thiết bị cầm tay.

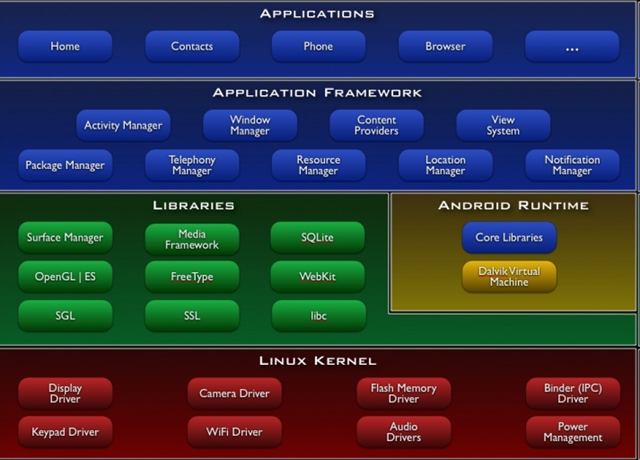
Phiên bản SDK lần đầu tiên phát hành vào tháng 11 năm 2007, hãng T-Mobile cũng công bố chiếc điện thoại Android đầu tiên đó là chiếc T-Mobile G1, chiếc smartphone đầu tiên dựa trên nền tảng Android. Một vài ngày sau đó, Google lại tiếp tục công bố sự ra mắt phiên bản Android SDK release Candidate 1.0. Trong tháng 10 năm 2008, Google được cấp giấy phép mã nguồn mở cho Android Platform.

Khi Android được phát hành thì một trong số các mục tiêu trong kiến trúc của nó là cho phép các ứng dụng có thể tương tác được với nhau và có thể sử dụng lại các thành phần từ những ứng dụng khác. Việc tái sử dụng không chỉ được áp dụng cho các dịch vụ mà nó còn được áp dụng cho cả các thành phần dữ liệu và giao diện người dùng.

Vào cuối năm 2008, Google cho phát hành một thiết bị cầm tay được gọi là Android Dev Phone 1 có thể chạy được các ứng dụng Android mà không bị ràng buộc vào các nhà cung cấp mạng điện thoại di động. Mục tiêu của thiết bị này là cho phép các nhà phát triển thực hiện các cuộc thí nghiệm trên một thiết bị thực có thể chạy hệ điều hành Android mà không phải ký một bản hợp đồng nào. Vào khoảng cùng thời gian đó thì Google cũng cho phát hành một phiên vản vá lỗi 1.1 của hệ điều hành này. Ở cả hai phiên bản 1.0 và 1.1 Android chưa hỗ trợ soft-keyboard mà đòi hỏi các thiết bị phải sử dụng bàn phím vật lý. Android cố định vấn đề này bằng cách phát hành SDK 1.5 vào tháng Tư năm 2009, cùng với một số tính năng khác. Chẳng hạn như nâng cao khả năng ghi âm truyền thông, vật dụng, và các live folder..

* + 1. ***Kiến trúc Android***

Mô hình sau thể hiện một cách tổng quát các thành phần của hệ điều hành Android. Mỗi một phần sẽ được đặc tả một cách chi tiết dưới đây.



Hinh 2. Kiến trúc Android

**Kiến trúc android chia làm 4 tầng**:

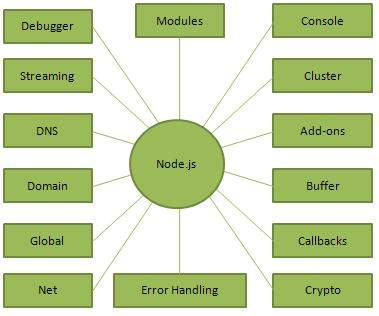
* + - * Tầng 1: **Tầng Application**
        + Tầng ở trên cùng cách xa với phần cứng nhất: Chứa các ứng dụng mà lập trình viên phát triển như : browser, Contacts, media.
      * Tầng 2: **Application Framework**
        + Activity Manager - quản lý vòng đời của các ứng dụng.​
        + Windows Manager - quản lý form của các ứng dụng.
        + Content Providers - cho phép các ứng dụng truy cập dữ liệu từ các ứng dụng khác hoặc để chia sẻ dữ liệu của riêng ứng dụng​
        + Google xây dựng cho các developer để phát triển các ứng dụng của họ trên Android chỉ bằng cách gọi các API.
        + View UI - để xây dựng layout của ứng dụng bao gồm: list view, text field, button, dialog, form …​
        + Resource Manager - cung cấp cách thức truy cập đến non-code resources như các asset, graphic, image, music, video …​
        + Notification Manager - cho phép tất cả các ứng dụng hiển thị thông báo của mình trên hệ điều hành.​
      * Tầng 3: **Libraries**
        + Là các thư viện được viết bằng ngôn ngữ C/C++ sẽ được các developer phát triển ứng dụng android thông qua tầng Android Framework. Có thể kể ra đây một số thư viện quen thuộc với các lập trình viên như:Media Libraries,Surface Manager, LibWebCore,…
      * Tầng 4: **Kernel Linux Layer**
        + Dựa trên Kernel Linux version 2.6 bởi nó cung cấp các trình điều khiển các thiết bị phần cứng(driver), quản lý tiến trình, quản lý tài nguyên, bảo mật,...
    1. ***Các kiến thức trong lập trình Android***
* Activity
  + Là 1 trong 4 thành phần quan trọng nhất của Android.
  + Thuật ngữ Activity chỉ một việc mà người dùng có thể thực hiện trong một ứng dụng Android. Do gần như mọi activity đều tương tác với người dùng, lớp Activity đảm nhận việc tạo ra một cửa sổ (window) để người lập trình đặt lên đó một giao diện UI với setContentView(View).
  + Một activity có thể mang nhiều dạng khác nhau: Một cửa sổ toàn màn hình (full screen window), một cửa sổ floating (với windowsIsFloating) hay nằm lồng bên trong 1 activity khác (với ActivityGroup).
* Intent
  + Theo định nghĩa của Google, Intent là một miêu tả về một hoạt động cần được thực hiện. Còn nói một cách đơn giản và dễ hiểu hơn, Intent là một cơ cấu cho phép truyền thông điệp giữa các thành phần của 1 ứng dụng và giữa các ứng dụng với nhau.
  + Intent được chia làm 2 loại:
    - Explicit Intents: intent đã được xác định thuộc tính component, nghĩa là đã chỉ rõ thành phần sẽ nhận và xử lý intent. Thông thường intent dạng này sẽ không bổ sung thêm các thuộc tính khác như action, data. Explicit Intent thương được sử dụng để khởi chạy các activity trong cùng 1 ứng dụng
    - Implicit Intents: Intent không chỉ rõ component xử lý, thay vào đó nó bổ sung thông tin trong các thuộc tính. Khi intent được gửi đi, hệ thống sẽ dựa vào những thông tin này để quyết định component nào thích hợp nhất để xử lý nó.
* User Interface
  + Android UI Controls: bao gồm các thành phần mà người sử dụng có thể nhìn thấy và tương tác được, bao gồm: Button, EditText, TextView, CheckBox, Radio Button,...
  + Android UI Layout: dùng để chứa các các view và tổ chức các view đó theo một trật tự nhất định, bao gồm các layout phổ biết như: Relative Layout, Linear Layout, Frame Layout, Listview, Gridview ...
* Fragment
  + Fragment là một phần giao diện người dùng hoặc hành vi của một ứng dụng. Fragment có thể được đặt trong Activity, nó có thể cho phép thiết kế activity với nhiều mô-đun. Có thể nói Fragment là một loại sub-Activity.
  + Fragment cũng có layout của riêng của nó, cũng có các hành vi và vòng đời riêng.
  + Chúng ta có thể thêm hoặc xóa Fragment trong một Activity trong khi Activity này đang chạy.
  + Có thể kết hợp nhiều Fragment trong một Activity để xây dựng giao diện người dùng đa khung.
* Service
  + Service là một trong 4 thành phần chính của android. Service không có giao diện chuyên dùng để thực hiện một nhiệm vụ nào đó được thực hiện ngầm dưới background mà không cần tương tác đến giao diện ví dụ như: chơi nhạc nền, download file, xử lý tính toán…
* Content provider
  + Content providers là thành phần cung cấp dữ liệu từ một ứng dụng đến một ứng dụng khác dựa trên các Request. Mỗi Request được xử lý bằng các phương thức của class ContentResolver. Một Content Provider có thể sử dụng các cách lưu trữ dữ liệu khác nhau, dữ liệu có thể được lưu trữ trong databases, file, thậm chí thông qua Network.
  + Content Providers cho phép bạn tập trung dữ liệu ở một nơi và các ứng dụng khác nhau sẽ truy xuất vào nó khi cần thiết. Content Provider hoạt động rất giống với một cơ sở dữ liệu, và bạn có thể truy vấn nó, chỉnh sửa nội dung, cũng như là thêm xóa các nội dung sử dụng các phương thức: insert(), update(), delete(), query(). Trong nhiều trường hợp dữ liệu được lưu trữ trong SQLite.
* Threading & Multithreading.
  + Thread có nghĩa là luồng, luồng ở đây là một đơn vị thực hiện đồng thời.
  + Multi Thread - Đa luồng: là một phương pháp "giả tạo" các lệnh thực thi cùng một lúc (ít nhất là trong thời điểm hiện tại) giúp cho người dùng có cảm giác chương trình thực thi nhanh hơn. Về mặt lập trình thì nó giúp chuơng trình thực thi nhiều tác vụ có thể coi như đồng thời, nhờ đó ứng dụng vẫn có thể tuơng tác tốt với người dùng trong khi đang bận làm 1 việc gì đó
  + Chúng ta có thể thực hiện đa luồng trong Android bằng nhiều cách khác nhau: Handler, Asyntask, Loader,..
  1. **LÝ THUYẾT NODEJS, MONGODB**
     1. ***NodeJS là gì?***

NodeJS là một mã nguồn được xây dựng dựa trên nền tảng Javascript V8 Engine, nó được sử dụng để xây dựng các ứng dụng web như các trang video clip, các forum và đặc biệt là trang mạng xã hội phạm vi hẹp. NodeJS là một mã nguồn mở được sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới. NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ WIndow cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.

* + 1. ***Đặc tính của NodeJS***

Qua phần tìm hiểu **NodeJS là gì** mình có giới thiệu một đặc tính rất quan trọng đó là Realtime, tuy nhiên vẫn còn khá nhiều đặc tính mà bạn cần phải biết trước khi học NodeJS.

* **Không đồng bộ**: Tất cả các API của NodeJS đều không đồng bộ (*none-blocking*), nó chủ yếu dựa trên nền của NodeJS Server và chờ đợi Server trả dữ liệu về. Việc di chuyển máy chủ đến các API tiếp theo sau khi gọi và cơ chế thông báo các sự kiện của Node.js giúp máy chủ để có được một phản ứng từ các cuộc gọi API trước (Realtime).
* **Chạy rất nhanh**: NodeJ được xây dựng dựa vào nền tảng V8 Javascript Engine nên việc thực thi chương trình rất nhanh.
* **Đơn luồng nhưng khả năng mở rộng cao**: Node.js sử dụng một mô hình luồng duy nhất với sự kiện lặp. cơ chế tổ chức sự kiện giúp các máy chủ để đáp ứng một cách không ngăn chặn và làm cho máy chủ cao khả năng mở rộng như trái ngược với các máy chủ truyền thống mà tạo đề hạn chế để xử lý yêu cầu. Node.js sử dụng một chương trình đơn luồng và các chương trình tương tự có thể cung cấp dịch vụ cho một số lượng lớn hơn nhiều so với yêu cầu máy chủ truyền thống như Apache HTTP Server.
* **Không đệm**: NodeJS không đệm bất kì một dữ liệu nào và các ứng dụng này chủ yếu là đầu ra dữ liệu.
* **Có giấy phép**: NodeJS đã được cấp giấy phép bởi [MIT License](https://raw.githubusercontent.com/joyent/node/v0.12.0/LICENSE).

****

Hinh 3. Đặc tính NodeJS

* + 1. ***MongoDB***

MongoDB là một trong những cơ sở dữ liệu mã nguồn mở NoSQL phổ biến nhất được biết bằng C++. Tính đến tháng 2/2015, MongoDB được xếp thứ 4 trong số các hệ thống cơ sở dữ liệu phổ biến nhất. Nó được pahts triển bởi công ty 10gen sau này được biết đến với tên MongoDB Inc..

MongoDB là cơ sở dữ liệu hướng tài liệu, nó lưu trữ dữ liệu trong các document dạng JSON với schema động rất linh hoạt. Nghĩa là bạn có thể lưu các bản ghi mà không cần lo lắng về cấu trúc dữ liệu như là số trường, kiểu của trường lưu trữ. Tài liệu MongoDB tương tự như các đối tượng JSON.

MongoDB là một cơ sở dữ liệu NoSQL hỗ trợ đa nền tảng, nó có thể chạy trên Windows, Linux và Mac...Nó hỗ trợ hầu hết các ngôn ngữ lập trình phổ biến như C#, Java, PHP, Javascript...và các môi trường phát triển khác nhau.

* **So sánh giữa SQL DB schema và Mongo DB schema**

| **SQL DB** | **MongoDB** |
| --- | --- |
| Table | Collection |
| Row | Document |
| Column | Field |
| Joins | Embeded documents, linking |
| Primary key | Primary key (mặc định là \_id do chính Mongo tạo) |

* 1. **PHÁT BIỂU BÀI TOÁN**

Bài toán đặt ra là làm thế nào để xây dựng một webservice bằng NodeJS, sử dụng MongoDB để lưu cơ sở dữ liệu. Để từ đó các máy Android đóng vai trò là các client có thể kết nối và tương tác với service thông qua các API trả về từ webservice.

* 1. **KẾT CHƯƠNG**

Nội dung chính của chương I là trình bày các cơ sở lý thuyết, công nghê cùng với những kiến thức nền tảng được sử dụng để thực hiện và triển khai đồ án đạt hiệu quả nhất.

**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

* 1. **GIỚI THIỆU**

Phân tích và thiết kế hệ thống (PT&TKHT) là quy trình để phát triển và bảo trì một hệ thống ứng dụng với mục đích nâng cao hiệu qủa của công việc. PT&TKHT luôn là bước đầu tiên và quan trọng nhất của bất kỳ hệ thống phần mềm nào. Nó đóng vai trò cốt lõi cho sự thành công hay thất bại của một hệ thống.

Việc PT&TKHT góp phần làm rõ các chức năng, đối tượng sử dụng cũng như cách thức để thực hiện các chức năng đó. Thông qua chúng ta sẽ có cái nhìn tổng quát về đề tài để lựa chọn công nghệ, kiến thống để phát triển hệ thống sao cho phù hợp nhất.

* 1. **PHÂN TÍCH HIỆN TRẠNG**

Kế hoạch của đề tài được thực hiện theo lần lượt các bước:

* Khảo sát, tìm hiểu thực tế, xác định yêu cầu, nhiệm vụ, mục đích của đề tài.
* Định hướng phát triển, môi trường làm việc, giải quyết bài toán.
* Phân tích yêu cầu chức năng chính của đề tài.
* Thiết kế hệ thống
* Tiến hành phát triển, lập trình theo các chức năng đã phân tích.
* Kiểm thử
* Cài đặt
* Bảo trì
  1. **PHÂN TÍCH YÊU CẦU**
     1. ***Đối tượng sử dụng***

Ứng dụng hướng tới một số nhóm người dùng:

* + - Người dùng muốn đi đến địa điểm nào đó bằng xe máy, nhưng chưa biết phải tìm kiếm xe ôm tại đâu.
    - Người dùng đang có xe máy và muốn thêm thu nhập từ xe máy của mình mà không bị ràng buộc về thời gian làm việc
    - Người dùng đang cần tìm kiến địa điểm sửa xe máy mọi lúc mọi nơi khi xe bị hỏng.
    1. ***Các yêu cầu để xây dựng hệ thống hoàn thiện***
       1. **Yêu cầu chức năng**
    - Hiển thị được bản đồ lên giao diện người dùng
    - Đăng ký tài khoản và đăng nhập tài khoản thông qua các tài khoản mạng xác hội Facebook cũng như Gmail
    - Cho phép người dùng tìm kiếm địa điểm trên bản đồ.
    - Cho phép người dùng thực hiện cuộc gọi với các tài xê đang hoạt động.
    - Gợi ý đường đi, ước lượng thời gian quảng đường cũng như giá tiền phải trả cho mỗi chuyến đi.
    - Xem bảng giá tính tiền
    - Hiển thị lịch sử các chuyến đi, danh sách các tài xế được yêu thích
    - Người dùng có thể đóng góp ý kiến của mình về hệ thống cũng như thái độ của các tài xế bằng cách đánh giá trức tiếp sau mỗi chuyến đi của mình.
    - Trở thành một phần của hệ thống bằng cách đăng ký những thông tin cần thiết cho hệ thống.
    - Tìm kiếm cũng như đăng ký địa điểm sửa xe máy, dịch vụ sửa xe máy lưu động.
    - Đóng góp ý kiến cho chúng tôi.
      1. **Yêu cầu phi chứng năng**

Phần mềm được xây dựng phải thỏa mãn các yêu cầu sau:

* + Cho phép cài đặt được trên các điện thoại chạy hệ điều hành Android
  + Đảm bảo hệ thống chạy ổn định, an toàn
  + Giao diện người dùng thiết kế đơn giản, thân thiện và dễ sử dụng
    1. ***Công nghệ sử dụng***

Danh sách công nghệ được sử dụng để xây dựng ứng dụng trên.

* + Android SDK
  + RESTful Webservice
  + NodeJS với MongoDB
  1. **THIẾT KẾ HỆ THỐNG**
     1. ***Phân tích chức năng hệ thống***
        1. **Đăng ký**

Để có thể vào giao diện của ứng dụng yêu cầu người dùng phải có tài khoản đăng nhập. Có hai cách để đăng ký tài khoản :

* Đăng ký theo form trên ứng dụng
* Đăng ký bằng tài khoản Facebook hoặc Gmail
* **Đăng nhập theo form trên ứng dụng**
  + - Luồng dữ liệu vào :

+ Họ và tên người dùng

+ Email người dùng

+ Số điện thoại

+ Mật khẩu đăng nhập

+ Xác nhận mật khẩu

* + - Tao thác xử lý

+ Sau khi kiểm tra tính hợp lệ của dữ liệu đầu vào:

* Họ và tên người dùng không thể để trống
* Email người dùng nhập phải đúng định dạng của email
* Số điện thoại không thể để trống
* Mật khẩu có độ dài nằm trong khoảng 6-20 ký
* Xác nhận mật khẩu phải giống mật khẩu đã nhập ở trên.

+ Khi các tài dữ liệu đầu vào đã hợp lệ thì chúng ta gởi lên server xử lý.

* + - Luồng dữ liệu ra

+ Nếu đăng ký thành công hệ thống sẽ tự động chuyển sang màn hình đăng nhập với email đăng ký được nhập sẵn.

+ Nếu email đăng ký đã tồn tại thì thông báo tài khoản đã tồn tại.

* **Đăng ký với tài khoản Facebook và Gmail**

Hầu hết người dùng đều có tài khoản Facebook hay Gmail, người dùng có thể sử dụng các tài khoản đó đăng ký tài khoản trên hệ thống. Chỉ cần click vào nút bấm Facebook hoặc Gmail trên màn hình đăng nhập thì người dùng có thể ủy quyền cho ứng dụng sử dụng thông tin từ các tài khoản Facebook, Gmail để đăng ký tài khoản trên hệ thống.

Khi đăng ký thành công thì ứng dụng tự động chuyển sang màn hình chính.

* + - 1. **Đăng nhập**

Người dùng sử dụng tài khoản đã được đăng ký để đăng nhập vào hệ thống

* + - Luồng dữ liệu vào

+ Email đăng nhập

+ Mật khẩu đăng nhập

* + - Thao tác xử lý

Sau khi kiêm tra tính hợp lệ của dữ liệu đầu vào

* Email phải đúng định dạng
* Passowrd phải có độ dài từ 6-20 ký tự

Người dùng bấm nút Đăng nhập để gởi dữ liệu lên server xử lý.

* + - Luồng dữ liệu ra

**+** Nếu đăng nhập thành công, ứng dụng sẽ tự động chuyển sang màn hình chính (màn hình maps).

**+** Nếu đăng nhập thất bại thì thống báo lỗi ra màn hình.

Đối với tài khoản được đăng ký bằng tài khoản Facebook, Gmail thì khi vào màn hình đăng nhập sẽ được hệ thống kiểm tra, nếu dữ liệu hợp lệ sẽ được tự động đăng nhập

* + - 1. **Tìm kiếm và thực hiện chuyến đi**

Đây là một chức năng chính của đề tại. Chức năng cho phép người dùng có thể tìm kiếm những tài xế đang hoạt động dựa theo vị trí hiện tại của người dùng.

Người dùng có thể chọn bất kỳ tài xế nào đang hoạt động để thực hiện chuyến đi của mình.

Khi người dùng tìm kiếm địa điểm mình cần đến thì hệ thống sẽ tự động gợi ý cho người dùng một số địa điểm ở gần nơi bạn đang đứng theo nhiều chủ đề khác nhau, như: trường đại học, bệnh viện và nhà hàng. Ngoài ra người dùng có thể tìm kiếm trực tiếp địa điểm mình cần đến trên google maps.

Sau khi người dùng chọn được địa điểm mình cần đến thì hệ thống sẽ tính quãng đường bạn cần đi là khoảng bao nhiêu kilomet, từ đó sẽ ước lượng số tiền bạn cần phải trả cho chuyến đi của mình. Giá tiền sẽ được quy định trong bảng giá của hệ thống, người dùng có thể kiếm tra lại giá tiền nếu muốn.

Nếu người dùng nhận thấy giá tiền phù hợp với quãng đường mà mình sắp đi thì co thể gọi cho tài xế gần mình nhất để tiết kiệm thời gian duy chuyển.

* + - Luồng dữ liệu vào

+ Vị trí hiện tại của người dùng (kinh độ, vĩ độ)

+ Địa điểm người dùng cần đến (kinh độ, vĩ độ)

* + - Luồng dữ liệu ra

+ Độ dài quãng đường mà mình sắp đi ()

+ Thời gian và giá tiền cần phải trả cho chuyến đi

Nếu người dùng thực hiện thành công chuyến đi thì chi tiết chuyến đi sẽ được lưu vào lịch sử chuyến đi của bạn.

* + - 1. **Xem bảng giá hệ thống**

Hệ thông sẽ đưa ra một bảng giá quy định cách tính giá tiền cho mỗi chuyến đi. Bảng giá được chia làm hai mốc:

* + - Hai kilomet đầu tiên
    - Sau hai kilomet đầu tiên

Tuy nhiên bảng giá sẽ được thay đổi tùy theo nhu cầu, khu vực hoặc thời điểm. Đặc biệt những chuyến xe sau 11 giờ khuya đến 5 giờ sáng sẽ được tình thêm cước phí phụ.

* + - 1. **Đánh giá sau chuyến đi**

Sau mỗi chuyến đi của người dùng thì hệ thống sẽ tự động gởi thông báo về điện thoại của người dùng nhằm yêu cầu người dùng đánh giá về chuyến đi của mình như thế nào? Thái độ phục vụ của tài xế ?

Có 3 mức để người dùng đánh giá là rất tốt, tốt và chưa tốt. Nếu bạn chọn mức đánh giá rất tốt và tốt thì không cần để lại bình luận của mình, nhưng nếu bạn chọn mức đánh giá là chưa tốt thì bạn phải thêm lý do vì sao chưa tốt. Dựa vào kết quả đánh giá, hệ thống có thể đánh giá lại chất lượng dịch vụ và thái độ phục vụ khách của từng tài chế góp phần đưa dịch vụ của chúng ta ngày càng chuyên nghiệp và phát triển hơn.

* + - Luồng dữ liệu vào

+ Lịch sử chuyến đi

+ Mức đánh giá của người dùng cho chuyến đi của mình.

+ Bình luận của người dùng (Có hoặc không tùy vào mức đánh giá)

* + - Luồng dữ liệu ra

+ Nếu bạn gởi đánh giá của chuyến đi thành công thì hệ thống sẽ lưu lại để ban quản trị hệ thống dựa vào kết quả đánh giá để xử lý các tài xế bị đánh giá chưa tốt.

* + - 1. **Trở thành tài xế**

Ứng dụng kết nối những người dùng lại với nhau, vậy làm thế nào để kết nối người dùng? Bằng chế độ trở thanh tài xê thì bất kì người dùng nào cũng có thể trở thành tài xế, chỉ cần cung cấp một số thông tin cần thiết để bảo tính an toàn cho người sử dụng thì bạn đã trở thành một tài xế thực thụ. Khi trở thành tài xế bạn có thể kiếm thêm thu nhập cho bản thân thông qua các chuyến đi.

* + - Luồng dữ liệu vào

+ Số chứng minh nhân dân – Ngày cấp – Nơi cấp

+ Mã số giấy phép lái xe

+ Số seri giấy phép lái xe

+ Số biển số xe

Sau khi kiểm tra tính hợp lệ của dữ liệu, dữ liệu được gởi lên server.

* + - Luồng dữ liệu ra

+ Sau khi dữ liệu được gởi lên server thì ban quản trị sẽ kiểm tra tính chính xác của dữ liệu mà người dùng cũng cấp. Sau khi kiểm tra dữ liệu hệ thống sẽ thông báo về cho người dùng tình trạng dữ liệu đăng kí. Nếu dữ liệu cung cấp chính xác thì ban quản trị sẽ cho phép bạn trở thành tài xế, nếu không hợp lệ thì hệ thống cũng sẽ thông báo lỗi về cho người dùng để người dùng có thể kiểm tra và bổ sung lại thông tin cho chính xác sớm nhất có thể.

+ Một khi bạn đã thành tài xê thì bạn có thể tham gia vào mạng lưới tài xế của hệ thống bằng cách bật chế độ tài xế khi bạn có thời gian. Hệ thống sẽ thêm bạn vào danh sách hoạt động để một khi người dùng muốn thực hiện chuyến đi sẽ thấy được vị trí của bạn.

* + - 1. **Yêu thích**

Ứng dụng cho phép người dùng có thể yêu thích một tãi xế bất kì. Có thể bạn đã đi với tài xế này một lần và bạn thích thái độ phục vụ của họ, bạn muốn lưu lại để sau này có thể dễ dàng tìm kiếm hơn. Ngoài ra, nếu một tài xế có lượt yêu thích nhiều đồng nghĩa với việc tài xế đó sẽ có cơ hội đươc chọn cho chuyến đi của người dùng hơn. Thông qua lượt yêu thích sẽ góp phần làm cho hệ thống ngày 1 thông minh hơn. Bạn có thể yêu thích hoặc bỏ yêu thích một tài xế nào mà bạn đã yêu thích trước đó.

* + - Luồng dữ liệu vào

+ Người dùng

+ Tài xế

+ Tình trạng yêu thích (Yêu thích hoặc hủy bỏ yêu thích)

* + - Luồng dữ liệu ra

+ Khi bạn yêu thích một tài xê thì hệ thống sẽ tự động thêm tài xê đó vào danh sách yêu thích của bạn và ngược lại, khi bạn bỏ yêu thích thì tài xế đó sẽ bị xóa khỏi danh sách yêu thích.

+ Khi truy cập vào danh sách yêu thích, trạng thái hoạt động của tài xế sẽ được hiển thị, đang hoạt động hoặc không hoạt động.

* + - 1. **Tìm kiếm nơi sửa xe máy**

Nếu bạn đi chơi xa mà chẳng may xe máy của bạn bị hỏng ở một nơi xa lạ, một nơi vắng vẻ nào đó hoặc là trong đem khuya chẳng han, hẳn bạn đang băng khoăn không biết phải tìm nơi sửa xe ở đâu? Vậy hãy cài ứng dụng này ngay trên smartphone của bạn. Ứng dụng sẽ cũng cấp cho bạn tất cả các cửa hàng sửa xe máy gần bạn nhất, kể cả hệ thống sửa xe máy lưu động cũng được tìm thấy. Vậy là bạn có thể đi chơi xa mà không sợ hỏng xe. Nếu bạn thấy giá cả sửa chữa bất hợp lý thì có thể gởi phản hồi lại cho hệ thống, để hệ thống xử lý vi phạm của các cửa hàng sửa chữa.

* + - 1. **Đăng tin sửa xe máy**

Nếu bạn có cửa hàng xe máy hoặc đang là thợ sửa xe máy lưu động và bạn muốn nhiều người biết đến thì hãy đến với ứng dụng của chúng tôi. Hệ thống sẽ cho phép bạn đăng ký thông tin cửa hàng của bạn. Như vậy cửa hàng của bạn sẽ được nhiều người biết đến, nhưng những thông tin của bạn cung cấp yêu cầu phải chính xác nếu không sẽ không được hệ thống phê duyệt.

* + - Luồng dữ liệu vào

+ Tên cửa hàng hoặc tên người sửa chữa lưu động

+ Địa chỉ

+ Số điện thoại

+ Giờ hoạt động (Open Time – Close Time)

+ Vị trí (kinh độ, vĩ độ)

Kiểm tra tính hợp lệ của dữ liệu trong form đăng ký

+ Tên cửa hàng hoặc tên người sửa chữa không được bỏ trống

+ Địa chỉ không được bỏ trống

+ Số điện thoại không được bỏ trống

+ Giờ hoạt động (Open Time , Close Time) không được để trống

+ Bạn phải chọn vị trí của cửa hàng bằng cách chạm trên bản đồ để đánh dấu chính xác vị trí.

Sau khi dữ liệu đầu vào đã thỏa mãn được tính hợp lệ của dữ liệu thì khi bấm nút Đăng ký dữ liệu sẽ được gởi lên server.

* + - Luồng dữ liệu ra.

Nếu kết quả trả về sau đăng ký là thành công thì hệ thống sẽ gởi thông báo thành công đồng thời sẽ chuyển màn hình sang danh sách cửa hàng sửa chữa. Ngược lại nếu đăng ký thất bại thì hệ thống cũng sẽ gởi thông báo lỗi đã xảy ra trong quá trình đăng ký.

* + - 1. **Gởi phản hồi**

Cách tốt nhất làm cho ứng dụng ngày càng phát triển chính là sự đóng góp ý kiến cho hệ thống. Bạn có thể nêu lên ý kiến về ứng dụng, chỗ nào bạn thấy chưa hợp lý thì hãy vui lòng gởi cho chúng tôi. Mọi đóng góp của bạn sẽ được chúng tôi xem xét kỹ lưỡng để góp phần làm cho ứng dụng hoàn thiện hơn.

* + - Luồng dữ liệu vào

+ Người đóng góp ý kiến

+ Tiêu đề

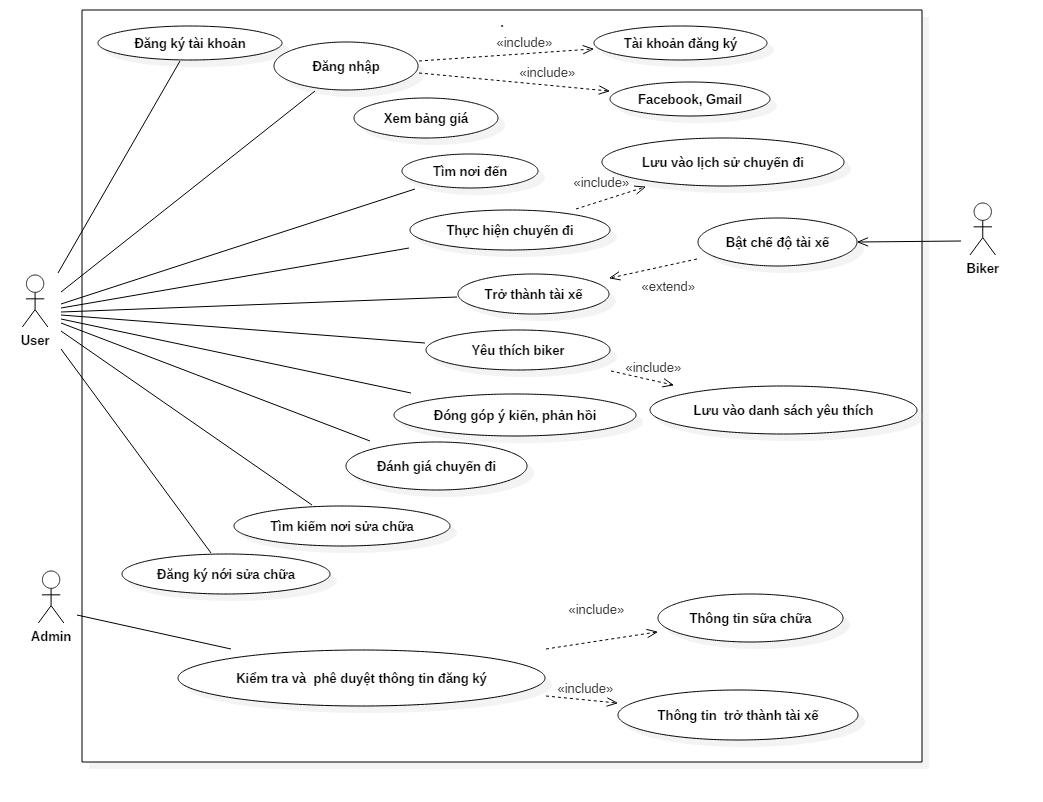
+ Nội dung

Cần kiểm tra tính hợp lệ của nội dung trước khi gởi dữ liệu lên server. Nội dung góp ý không thể bỏ trống.

* + - Luồng dữ liệu ra

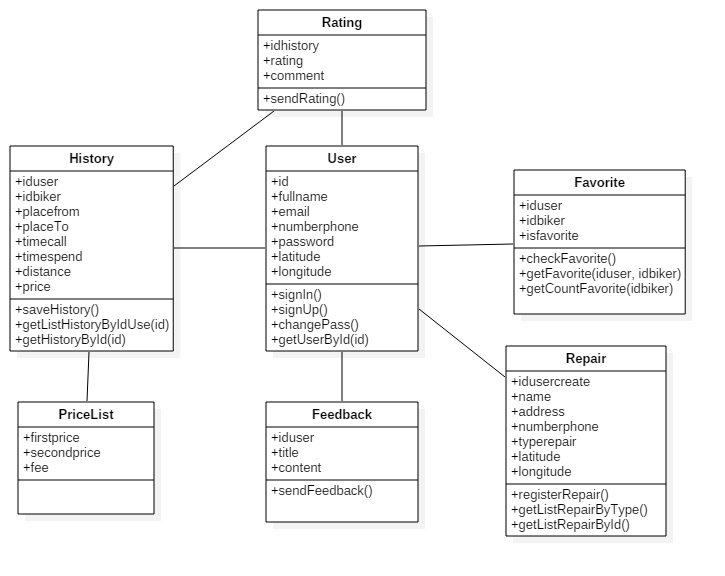
+ Khi bạn đóng góp ý kiến thành công thì ứng dụng sẽ tự động chuyển sang màn hình chính.

* + 1. ***Các biểu đồ***
       1. **Biểu đồ usercase**



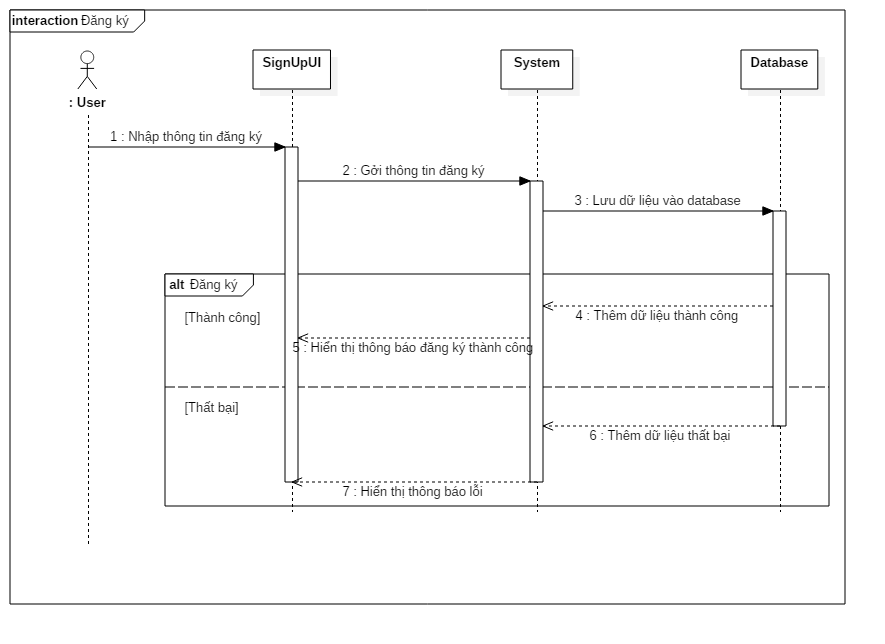
Hinh 4.Biểu đồ usercase

* + - 1. **Biểu đồ lớp**

****

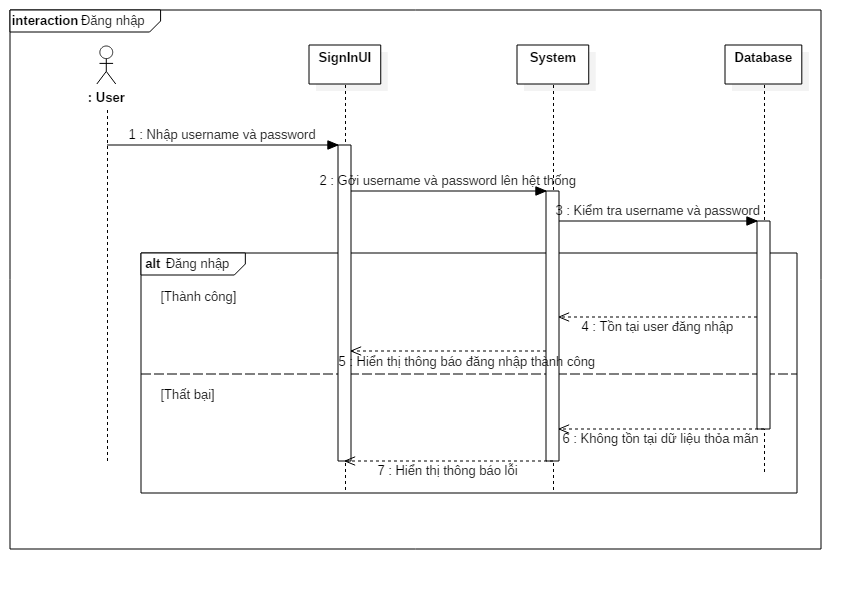
Hinh 5. Biểu đồ lớp

* + - 1. **Biểu đồ tuần tự**
* **Biểu đồ tuần tự chức năng đăng ký**

****

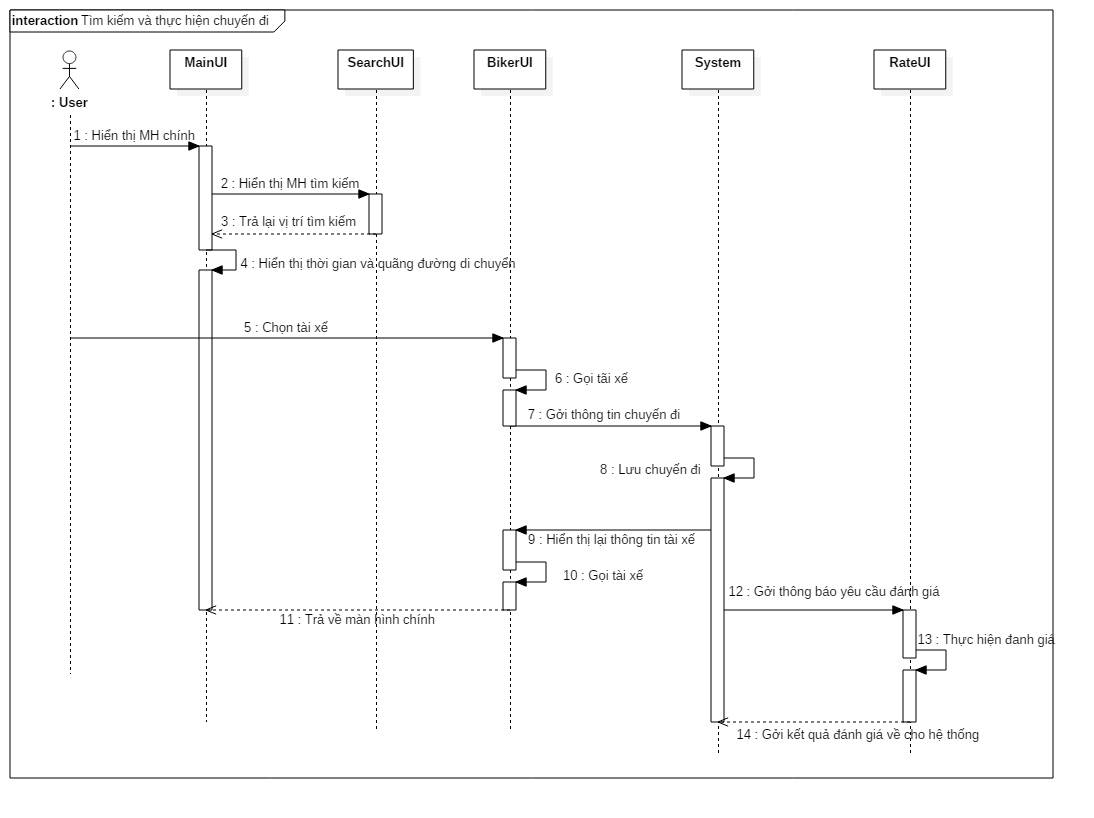
Biểu đồ tuần tự chức năng đăng ký tài khoản

* **Biểu đồ tuần tự chức năng đăng nhập**

****

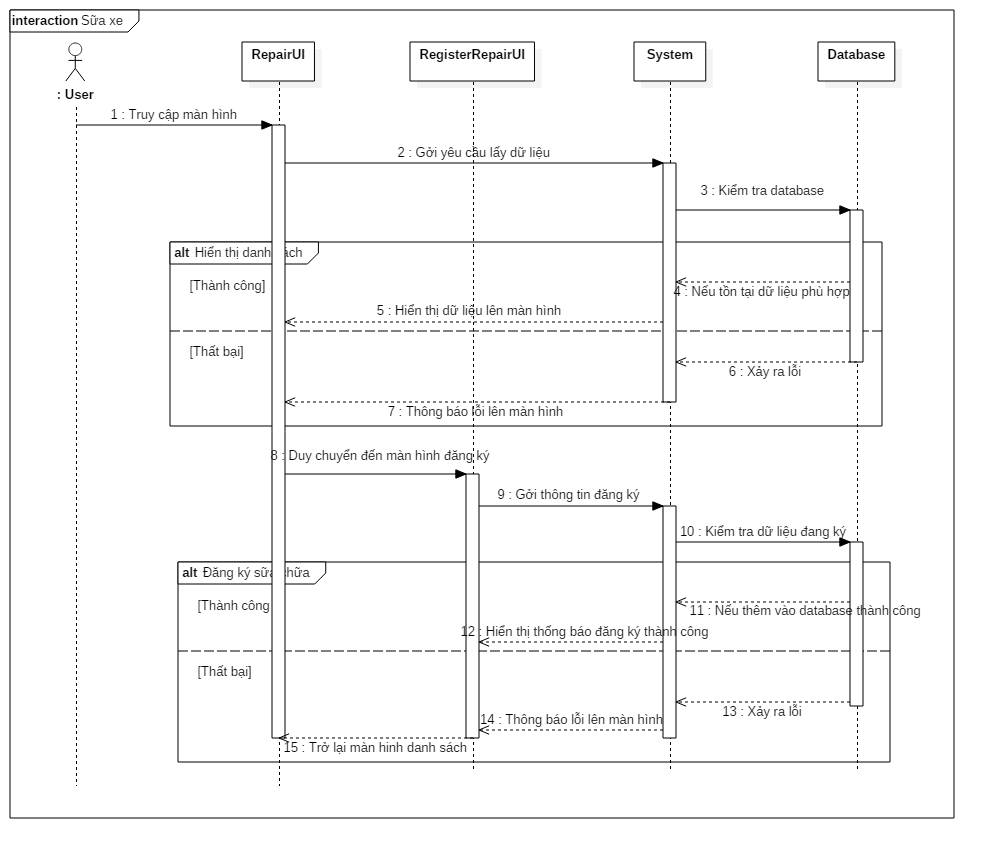
Biểu đồ tuần tự chức năng đăng nhập

* **Biểu đồ tuần tự chức năng tìm kiếm và thực hiện chuyến đi**

****

Biểu đồ tuần tự chức năng tìm kiếm và thực hiện chuyến đi

* **Biểu đồ tuần tự chức năng tìm kiếm và đăng ký địa điểm sửa chữa**



Biểu đồ tuần tự chức năng tìm kiếm và đăng ký địa điểm sửa chữa xe máy

* + 1. ***Thiết kế cơ sở dữ liệu***

|  |  |  |
| --- | --- | --- |
| Collection UserMeta | | |
| Field | Type | Description |
| fullname | String |  |
| email | String |  |
| password | String |  |
| access\_token | String |  |
| image\_avatar\_path | String |  |
| latitude | Number |  |
| longitude | Number |  |
| number\_phone | String |  |
| identification\_number | String |  |
| identification\_place | String |  |
| identification\_date | String |  |
| identification\_before\_path | String |  |
| identification\_after\_path | String |  |
| driving\_license\_number | String |  |
| driving\_license\_seri | String |  |
| number\_card | String |  |
| type\_user | Number |  |
| is\_driving | Number |  |
| is\_approved | Number |  |
| is\_become | Number |  |

|  |  |  |
| --- | --- | --- |
| Collection Feedback | | |
| Field | Type | Description |
| id\_user | String |  |
| title | String |  |
| content | String |  |

|  |  |  |
| --- | --- | --- |
| Collection HistoryMeta | | |
| Field | Type | Description |
| id\_user | String |  |
| id\_biker | String |  |
| time\_call | String |  |
| place\_from | String |  |
| latitude\_from | Number |  |
| longitude\_from | Number |  |
| place\_to | String |  |
| latitude\_to | Number |  |
| longitude\_to | Number |  |
| distance | Number |  |
| price | Number |  |
| time\_spend | Number |  |

|  |  |  |
| --- | --- | --- |
| Collection FavoriteMeta | | |
| Field | Type | Description |
| id\_user | String |  |
| id\_biker | String |  |
| is\_favorite | Number |  |
| timestamp\_created | Date |  |
| timestamp\_updated | Date |  |

|  |  |  |
| --- | --- | --- |
| Collection RepairMeta | | |
| Field | Type | Description |
| name | String |  |
| number\_phone | String |  |
| address | String |  |
| latitude | Number |  |
| longitude | Number |  |
| type\_repair | Number |  |
| id\_user\_created | String |  |
| time\_open | String |  |
| time\_close | String |  |
| timestamp\_created | Date |  |
| timestamp\_updated | Date |  |

|  |  |  |
| --- | --- | --- |
| Collection RateMeta | | |
| Field | Type | Description |
| id\_history | String |  |
| rating | String |  |
| comment | String |  |

**\*\* Đối với MongoDB thí mỗi collection đề có một Field “\_id”đóng vài là Primary Key, và là được thêm mặc định khi khởi tạo collection.**

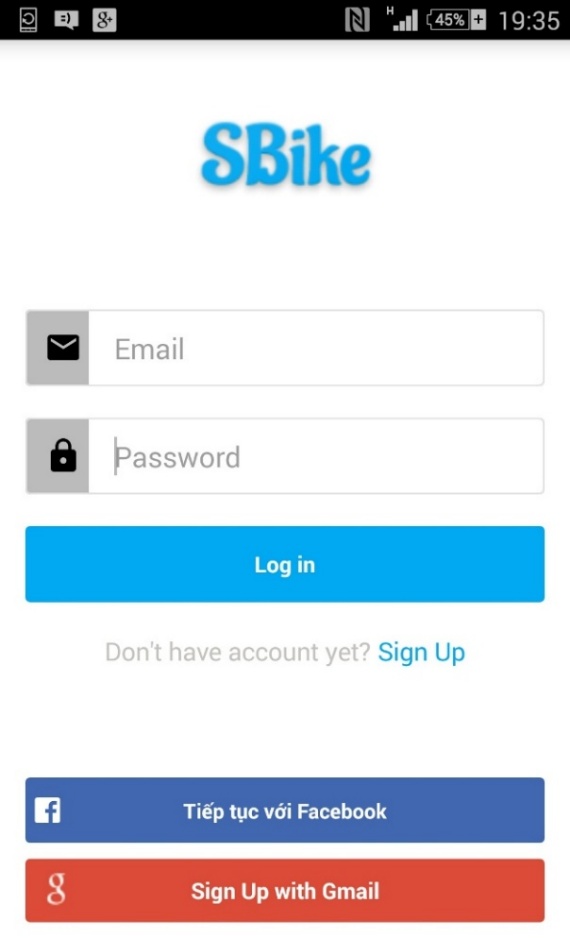
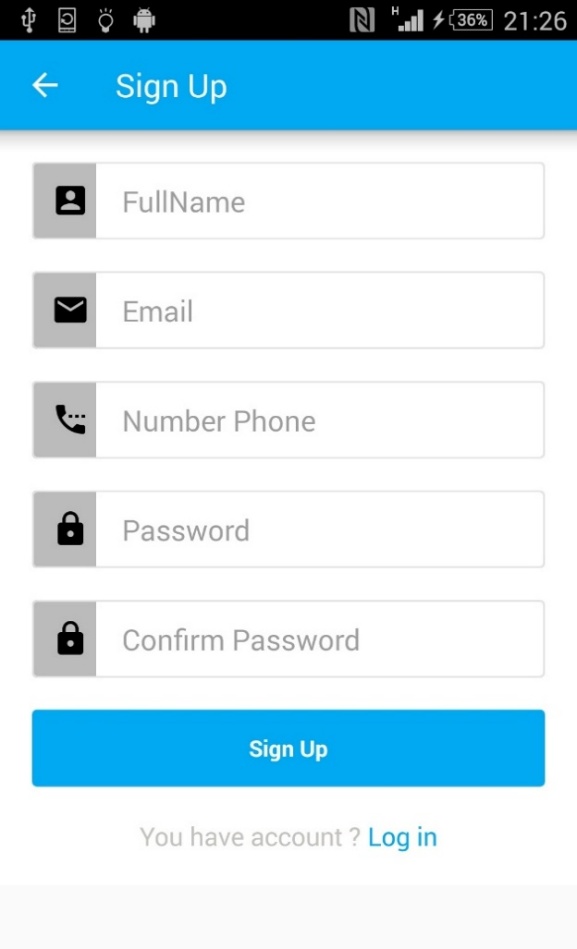
* 1. **KẾT CHƯƠNG**

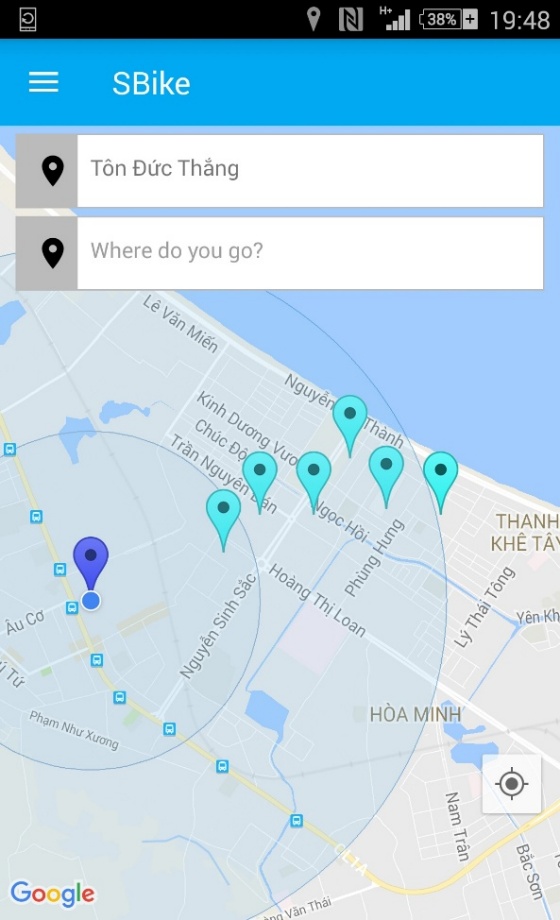
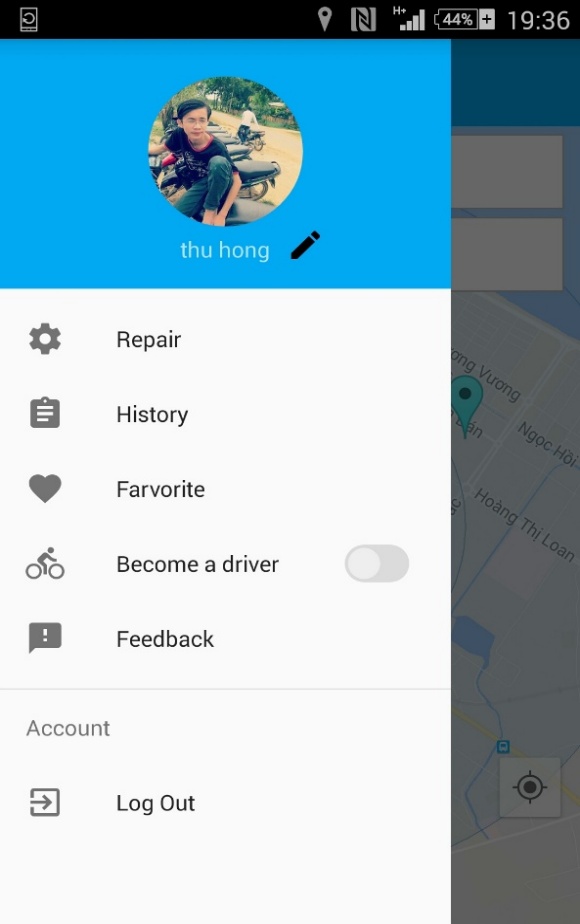
Nội dung chính của chương này là phân tích hệ thống, để xác định đối tượng người dùng, công nghệ sử dụng. Sau đó thiết kế hệ thống bằng các phân tích các chức năng chính, các biểu đồ và thiết kế cớ sở dữ liệu.

**CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ**

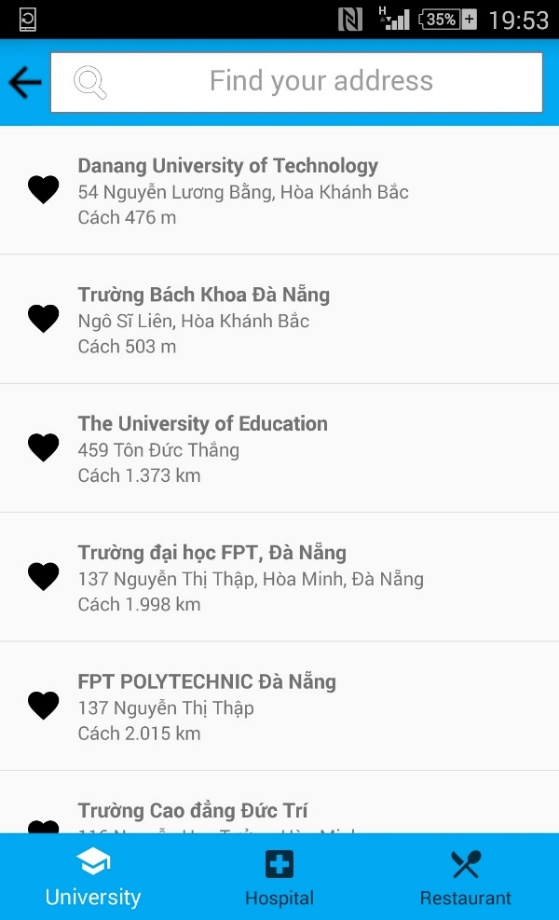
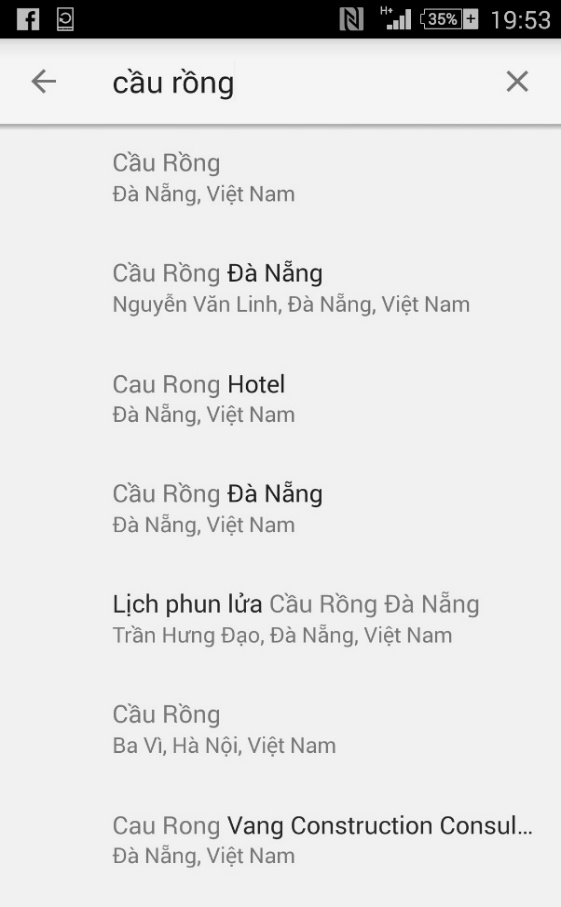
* 1. **MÔ HÌNH TRIỂN KHAI**
     1. ***Mô hình triển khai***
     + Khảo sát thông tin và tính thực tiễn của đề tài đối với người dùng
     + Phân rã từng chức năng của đề tài
     + Thiết kế giao diện chương trình
     + Tìm hiểu Google Maps API, Google Place API, RESTful webservice với NodeJS và MongoDB
     + Thực hiện phân tích và thiết kế database.
     + Phát triển đề tài trên nền tảng Android
     1. ***Công cụ hổ trợ***

Một số công cụ dùng để hổ trợ cho việc phát triển ứng dụng:

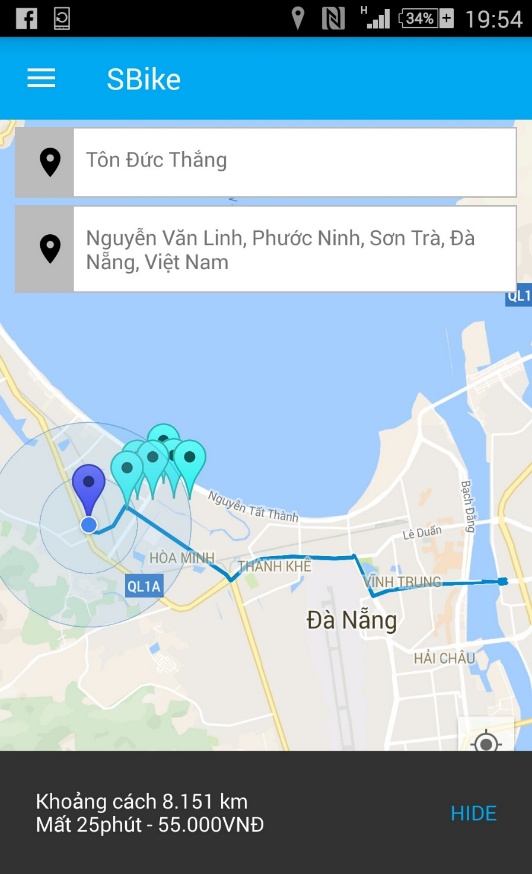
* + - Android Studio 2.3
    - Visual Studio Code
    - Postman
    - Robomongo
  1. **KẾT QUẢ CHƯƠNG TRÌNH**
     1. ***Màn hình đăng nhập, đăng ký***
     2. ***Màn hình chính của ứng dụng***

****

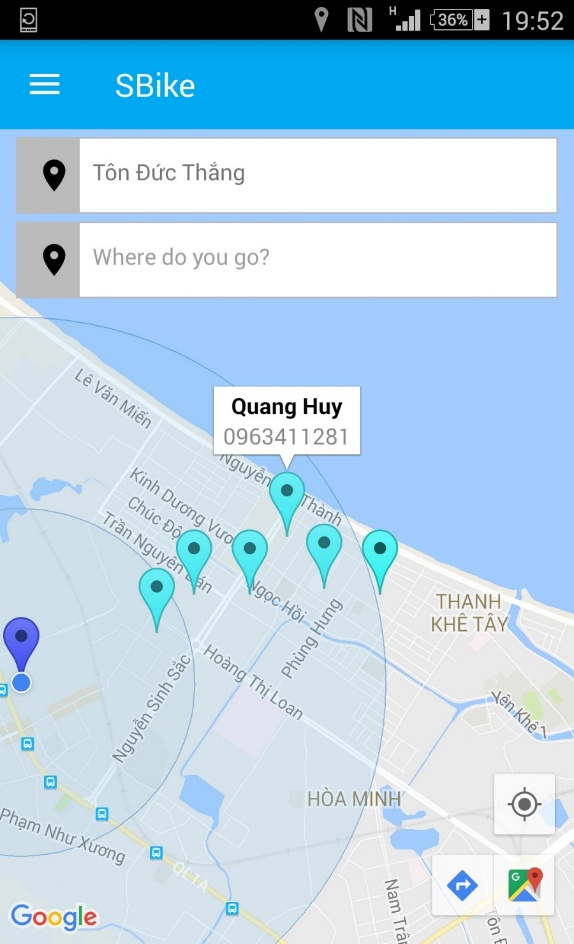
* + 1. ***Màn hình tìm kiếm địa điểm theo vị trí hiện tại và địa điểm bất kỳ trên google maps.***

****

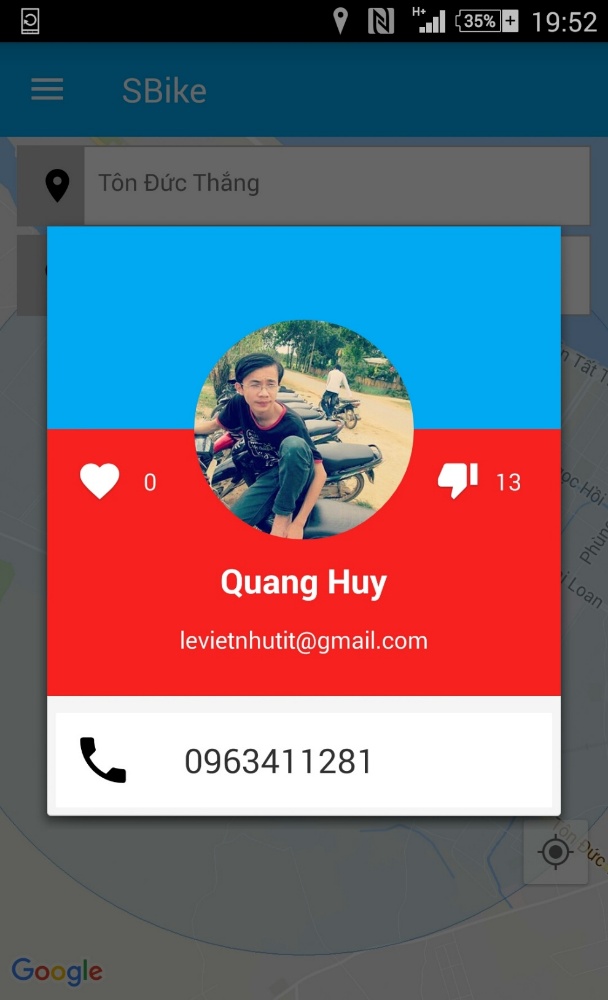
* + 1. ***Màn hình hiển thị địa điểm đã được tìm kiếm, vẽ đường đi và tính khoảng cách thời gian, khoảng cách***

****

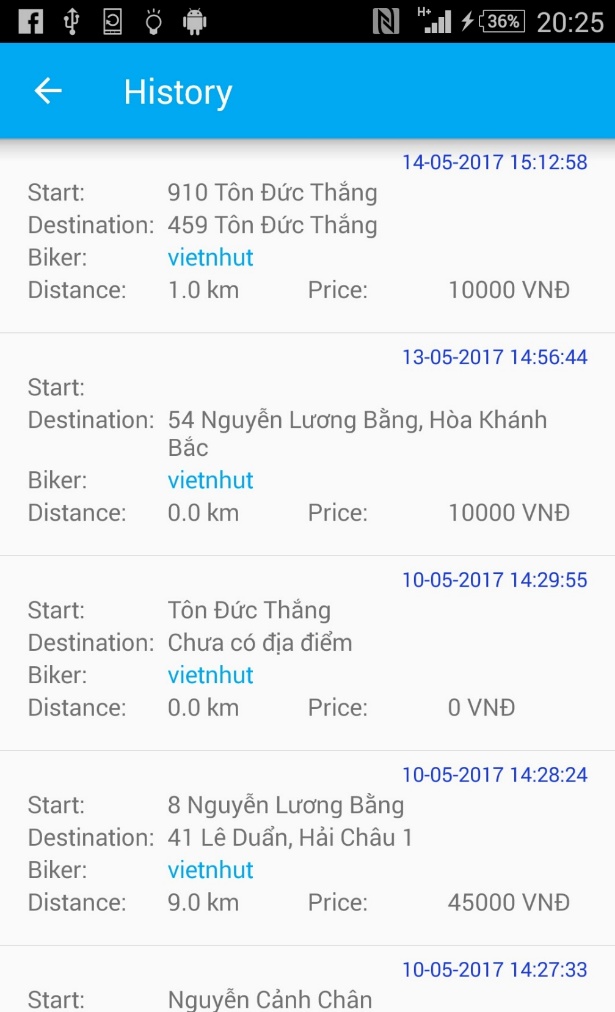
* + 1. ***Màn hình chọn một tài xế để thực hiện chuyến đi***

****

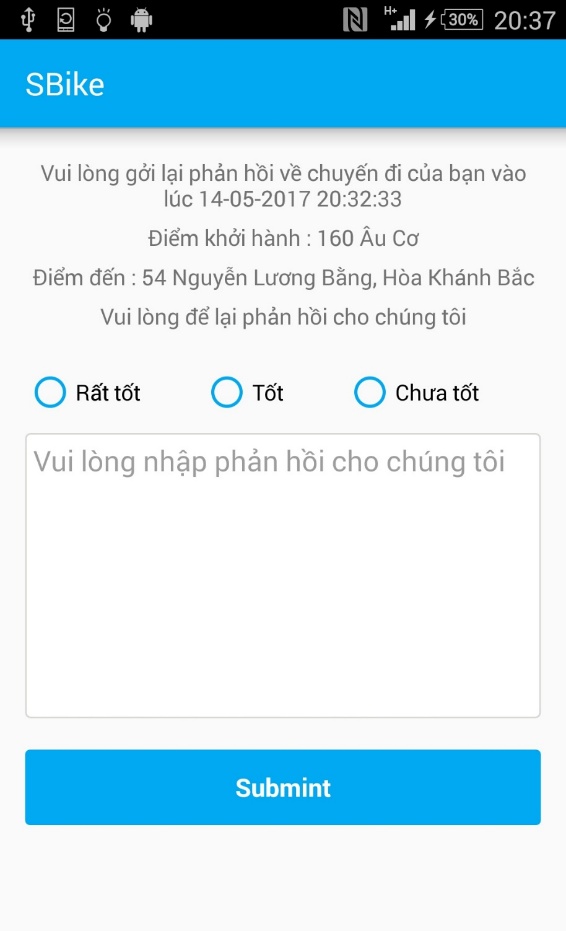
* + 1. ***Màn hình thông tin tài xế được chọn***

****

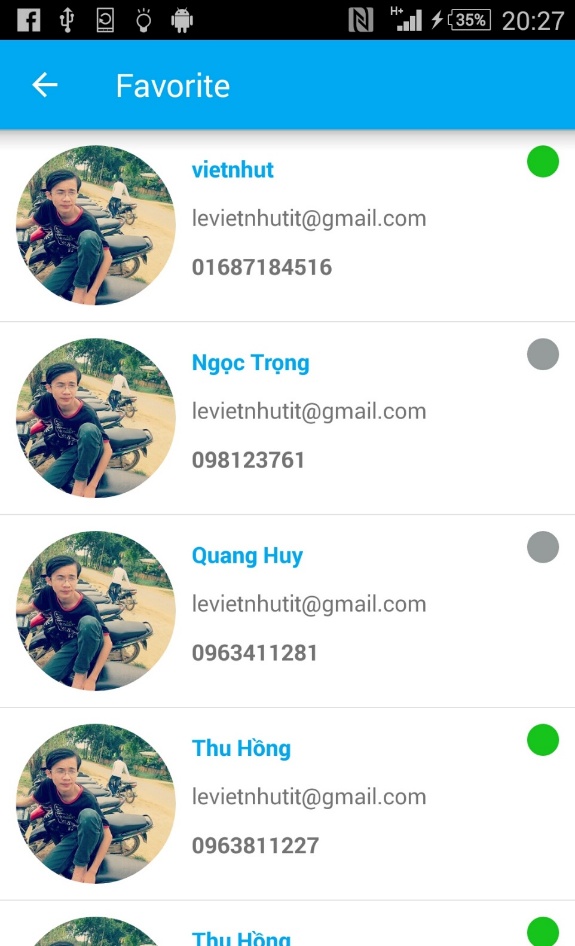
* + 1. ***Màn hình hiển thị lịch sử chuyến đi của người dùng***

****

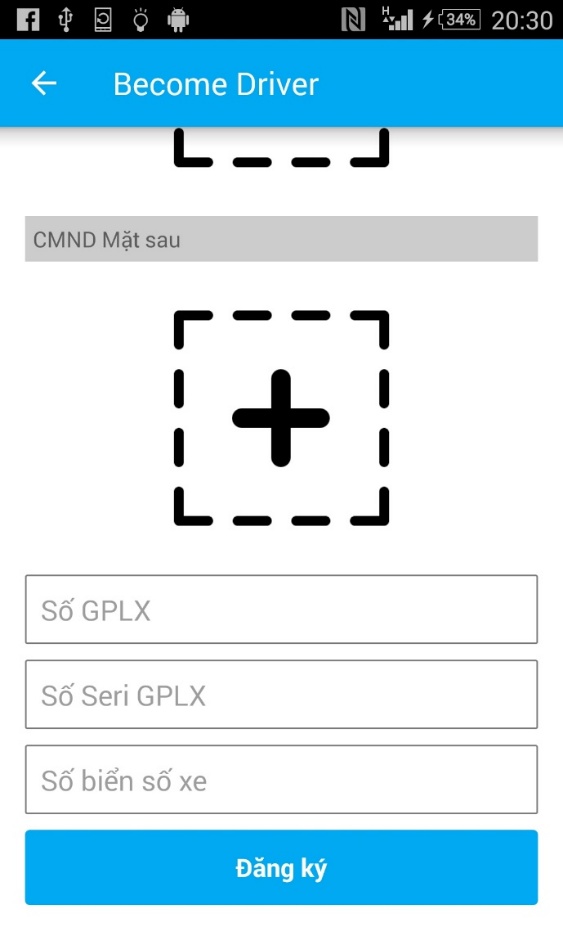
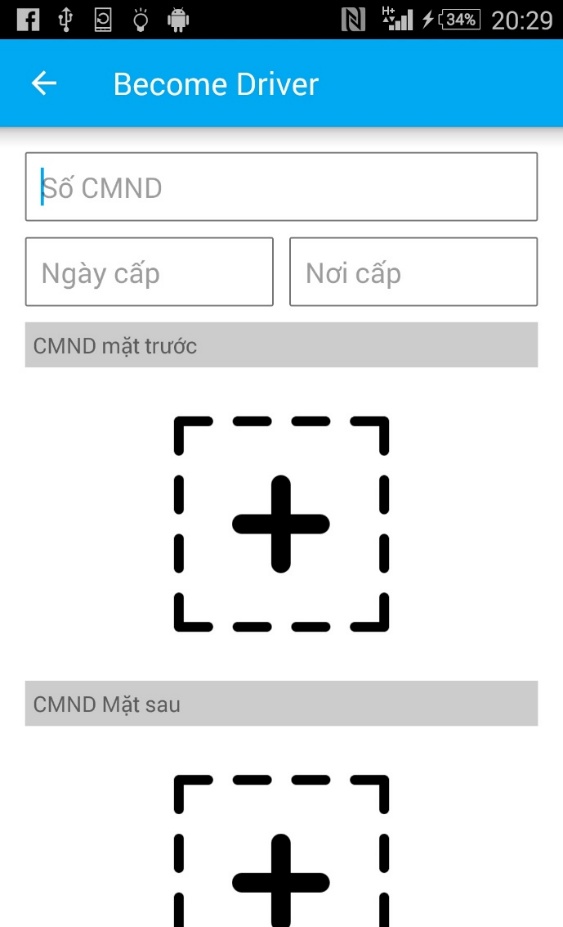
* + 1. ***Màn hình hiển thị đánh giá chuyến đi của người dùng***

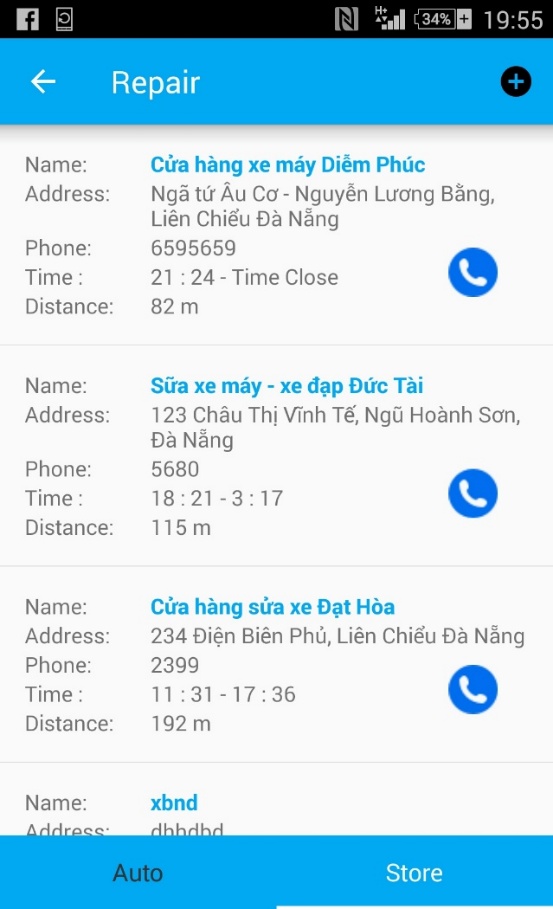
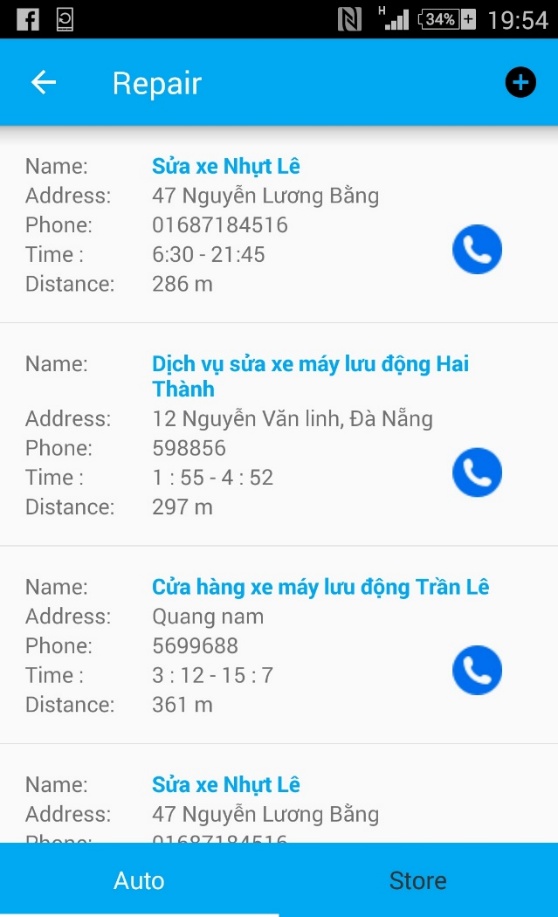
****

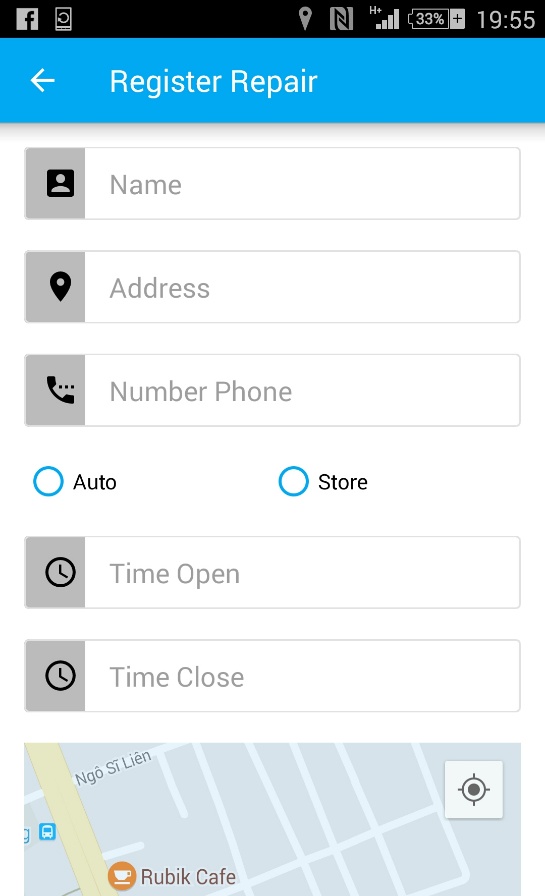
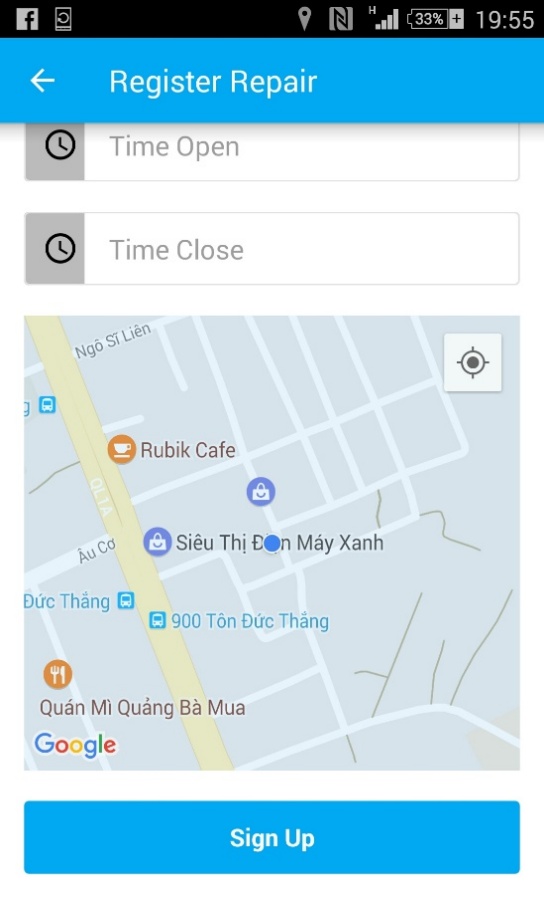
* + 1. ***Màn hình hiển thị tài xe được yêu thích của người dùng***

****

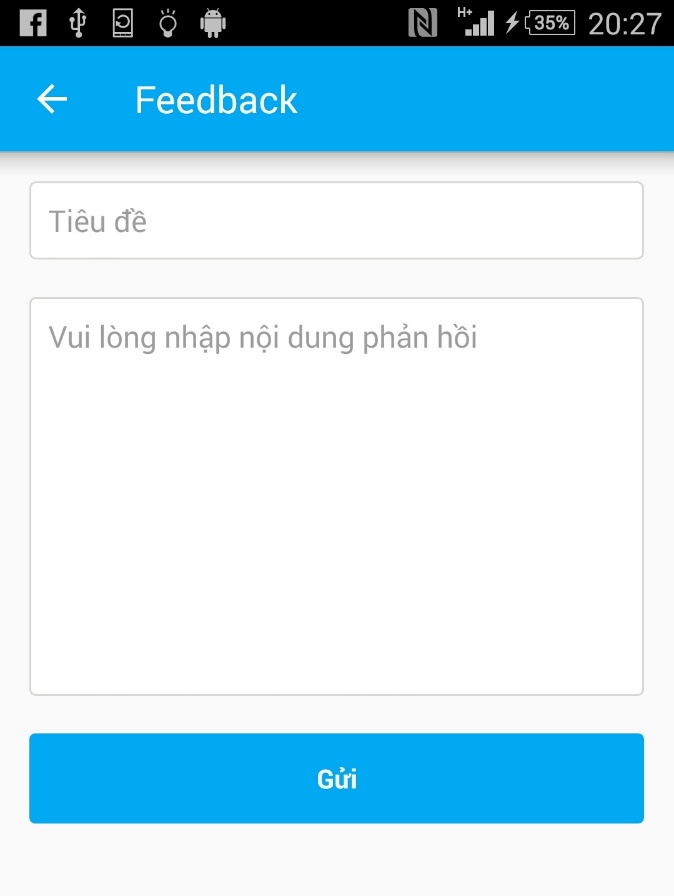
* + 1. ***Màn hình hiển thị thông tin cần thiết để trở thành tài xế***

****

* + 1. *****Màn hình hiển thị địa điểm sửa chữa xe máy***
    2. ***Màn hình đăng ký địa điểm sửa chữa xe máy***

****

* + 1. ***Màn hình hiển thị phản hồi, đóng góp ý kiến***

****

* 1. **NHẬN XÉT ĐÁNH GIÁ KẾT QUẢ**

Ứng dụng đã phần nào đáp ứng được một số yêu cầu đặt ra ban đầu từ lúc phân tích, thiết kế hệ thống. Các chức năng chính của hệ thống đảm bảo hoạt động tốt. Ứng dụng đã được triển khai và cài đặt trên hệ điều hành Android.

Tuy nhiên, ứng dụng còn một số hạn chế như sau :

* + - Giao diện trên ứng dụng chưa được bắt mắt cho lắm
    - Hiệu suất làm việc của ứng dụng chưa được cao
    - Khả năng áp dụng thực tiễn còn gặp nhiều khó khăn
  1. **KẾT CHƯƠNG**

Chương này trình bày mô hình triển trai và giới thiệu một số công cụ hổ trợ cho việc phát triển ứng dung. Hiển thị kết qua demo của một số màn hình ứng dụng.

**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

1. **KẾT QUẢ ĐẠT ĐƯỢC**
   * + Qua thời gian thực hiện đồ án em đã có nhiều tiến độ hơn trước. Như là:

**+** Hiểu sâu về kiến thức lập trình trong Java cũng như Android

**+** Biết cách sử dụng Google Maps API kết hợp với android và một số kiến thức liên quan đến nghiệp vụ xử lý trên maps.

**+** Biết cách xây dựng webservice bằng NodeJS và MongoDB.

**+** Nâng cao trình độ phân tích và thiết kế ứng dụng.

**+** Học cách đặt vấn đề và giải quyết các yêu cầu của đồ án tốt nghiệp, nâng cao khả năng tư duy sáng tạo trong lập trình.

**+** Cải thiện khả năng thiết kế giao diện.

1. **KIẾN NGHỊ VÀ HƯỚNG PHÁT TRIỂN**
   * + Do ứng dụng được thiết kế theo mô hình clien/server nên nếu nhiều client cùng truy cập một thời điểm thì hệ thống sẽ bị chậm và có thể gây chết hệ thống.
     + Gặp nhiều khó khăn trong quá trình phân tích và thiết kế hệ thống, việc thiết sót hoặc chưa tối ưu là điều không thể tránh khỏi.
     + Nếu có thể em sẽ phát triển quy mô mạng lưới tài xế xe máy hơn nữa, cho phép người dùng có thể thanh toán cước phí thông qua thẻ để đảm bảo tính an toàn cho người dùng. Phát triển ứng dụng trên các nền tảng mobile khác như iOS, WindowPhone.

**TÀI LIỆU THAM KHẢO**

* 1. Dave MacLean, Satya Komatineni, Grant Allen. *Pro Android 5*.
  2. <http://developer.android.com/guide/index.html>
  3. <http://stackoverflow.com/>
  4. <https://developers.google.com/maps/documentation/android-api/?hl=vi>
  5. <https://nodejs.org/en/>
  6. <https://www.mongodb.com/>