CSIS0801 Final Year Project 2014 – 2015

FYP14007
A Smart Phone Application to Assist Driving

Group Final Report

Supervisor: Dr. Chim T W
Second Examiner: Dr. Yiu S M

Students:
Leung Kin Ning, Chester 3035040891
Lee Jin Man, Jonathan 3035040011
Wong Kwan Wai, Kewea 2013718589
# Table of Contents

CHANGE HISTORY.................................................................................................................. 4  
ABSTRACT........................................................................................................................................ 4  
SECTION 1 PREFACE................................................................................................................. 5  
  1.1 PURPOSE.......................................................................................................................... 5  
  1.2 SCOPE.................................................................................................................................. 5  
  1.3 TEAM MEMBER AND STAKEHOLDER.............................................................................. 6  
  1.4 CREDIT.................................................................................................................................. 6  
SECTION 2 PROJECT OVERVIEW.............................................................................................. 7  
  2.1 BACKGROUND.................................................................................................................... 7  
  2.2 RELATED WORK............................................................................................................... 8  
  2.3 OBJECTIVE........................................................................................................................ 10  
  2.4 STRATEGIES.................................................................................................................... 11  
  2.5 SMART PHONE APPLICATION ANALYSIS...................................................................... 12  
  2.6 AMENDMENT OF PROJECT SCOPE FROM PROJECT PLAN........................................... 13  
  2.7 PROJECT DELIVERABLES................................................................................................. 16  
  2.8 MAJOR WORK PRODUCTS............................................................................................... 17  
  2.9 PROJECT RESPONSIBILITY............................................................................................. 18  
  2.10 MAJOR MILESTONES....................................................................................................... 19  
SECTION 3 PRODUCT DESIGN................................................................................................... 20  
  3.1 ITEMS OF THE PRODUCT FUNCTION............................................................................ 20  
  3.2 MAJOR PRODUCT FUNCTION DESCRIPTIONS............................................................ 21  
  3.3 HARDWARE REQUIREMENT............................................................................................ 24  
  3.4 SYSTEM ARCHITECTURE................................................................................................. 24  
      3.4.1 Overview..................................................................................................................... 24  
      3.4.2 Database design.......................................................................................................... 31  
  3.5 SOFTWARE PROCESS MODEL........................................................................................... 33  
  3.6 MONITORING AND CONTROLLING MECHANISM......................................................... 35  
  3.7 TESTING............................................................................................................................ 36  
SECTION 4 IMPLEMENTATION AND PROJECT DELIVERABLES SUMMARY................................. 37  
  4.1 PROJECT DELIVERABLE................................................................................................... 37  
  4.2 THE APPLICATION........................................................................................................... 37  
      4.2.1 Development Toolkit..................................................................................................... 37  
      4.2.2 Technology Used......................................................................................................... 39  
      4.2.3 User Interface ............................................................................................................. 40
4.2.4 Overall Application Design................................................................. 45
4.2.5 Summary of Technical Skills Used in Functions................................... 46
4.3 DIFFICULTIES AND POSSIBLE SOLUTIONS........................................... 49
  4.3.1 Challenges in Designing a Speech-enabled Application......................... 49
  4.3.2 Application design ............................................................................... 50
  4.3.3 Code sharing conflict ........................................................................... 51
4.4 FUTURE DEVELOPMENT........................................................................... 53
  4.4.1 Starting point relocation when travelling the route .............................. 53
  4.4.2 Speaking out the road names when travelling the route ....................... 53
  4.4.3 iOS development .................................................................................. 54
  4.4.4 Button selection by voice ...................................................................... 54

SECTION 5 SCHEDULE................................................................................... 55
SECTION 6 BALANCE SHEET AND RESOURCE ALLOCATION......................... 56
REFERENCES...................................................................................................... 57
APPENDICES...................................................................................................... 58
  APPENDIX I: MAIN FLOW OF THE APPLICATION (General Overview) ............ 58
  APPENDIX II: MAIN FLOW OF THE APPLICATION (Front-end View)................ 69
  APPENDIX III: MAIN FLOW OF THE APPLICATION (Back-end View).............. 60
  APPENDIX IV: DATABASE TABLE STRUCTURE........................................... 61
  APPENDIX V: TEST PLAN............................................................................. 62
Change History

<table>
<thead>
<tr>
<th>Version</th>
<th>Change Data</th>
<th>Author</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2015/04/07</td>
<td>Jonathan Lee</td>
<td>Draft</td>
</tr>
<tr>
<td>0.2</td>
<td>2015/04/15</td>
<td>Chester Leung</td>
<td>Revision on draft</td>
</tr>
<tr>
<td>0.3</td>
<td>2015/04/16</td>
<td>Kewea Wong</td>
<td>Revision on draft</td>
</tr>
<tr>
<td>0.4</td>
<td>2015/04/17</td>
<td>Kewea Wong</td>
<td>Revision on draft</td>
</tr>
<tr>
<td>0.5</td>
<td>2015/04/18</td>
<td>Chester Leung</td>
<td>Revision on draft</td>
</tr>
<tr>
<td>1.0</td>
<td>2015/04/19</td>
<td>Jonathan Lee</td>
<td>Final version</td>
</tr>
</tbody>
</table>

Abstract

Mobile applications are becoming more and more useful in our life. It is not difficult to observe that people use a variety of mobile applications anywhere and anytime. For example, a majority of people use WhatsApp, a famous mobile application nowadays, for communication. Also, there are many utility applications and game applications in the market. Applications play a very important role in the society. They do not only provide us a platform of entertainment and communication, but also assistance to users. In this project, we will integrate a voice recognition function to a mobile application.

Our project aims to provide drivers a more convenient way to obtain the instant road conditions and traffic information. We know that tying is painful, especially when you are driving. With the aid of the voice recognition function, we hope that drivers can retrieve useful traffic information in a user-friendly way. At the same time, they can pay more attention on the road conditions while driving. We expect our mobile application can assist drivers efficiently.
Section 1: Preface

1.1 Purpose

In this document, we are going to describe the finalized situation of our project - the smart application to assist driving, Final Year Project for Department of Computer Science, The University of Hong Kong in 2014-2015.

The contents of this document includes the background and overview of the project, the system design, deliverables, difficulties and limitations of the project as well as the ideas for further development.

1.2 Scope

Smart Application to Assist Driving is a Final Year Project for Department of Computer Science, The University of Hong Kong in 2014-2015. It aims to provide a new and convenient way for drivers to retrieve desired traffic information while driving. The software application is designed for driving use and thus its target audience is set to be drivers.

The main feature of our application is the voice recognition function which can surely bring convenient to drivers. Also, some other features such as route plotting are added to make the application more informative. The project team believes that through this project, voice technology can be integrated with the traditional driving application in an interactive way.
1.3 Team members and Stakeholder

This Final Year Project comprises of 3 students, 1 supervisor and 1 second examiner. The detailed information of the team is shown in Table 1.1 below.

<table>
<thead>
<tr>
<th>School</th>
<th>The University of Hong Kong</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Name</td>
<td>Smart Application to Assist Driving</td>
</tr>
<tr>
<td>Project Website</td>
<td><a href="http://i.cs.hku.hk/fyp/2014/fyp14007/">http://i.cs.hku.hk/fyp/2014/fyp14007/</a></td>
</tr>
<tr>
<td>Supervisor</td>
<td>Dr. Chim T W</td>
</tr>
<tr>
<td>Second Examiner</td>
<td>Dr. Yiu S M</td>
</tr>
<tr>
<td>Team Members</td>
<td>Leung Kin Ning, Chester (Contact person)</td>
</tr>
<tr>
<td></td>
<td>Lee Jin Man, Jonathan</td>
</tr>
<tr>
<td></td>
<td>Wong Kwan Wai, Kewea</td>
</tr>
<tr>
<td>Contact Mail</td>
<td><a href="mailto:kknleung@connect.hku.hk">kknleung@connect.hku.hk</a></td>
</tr>
</tbody>
</table>

Table 1.1

1.4 Credit

Android and Google Play are trademarks of Google Inc.

Spring Framework is a trademark of Spring Technologies protected by applicable trademark copyright and other intellectual property laws.
Section 2: Project Overview

2.1 Background

In this century, private cars seem not luxury goods when compared to the past. According to the statistical data provided by the Transport Department of Hong Kong, the number of licensing of vehicles has an increasing trend from about 37000 in 2009 to about 42000 in 2013. It infers that more and more people will have driving licenses and they can go everywhere by their own car.

![Number of Licensing of Vehicles](image)

*Fig. 2.1 – Statistical Data from the Transport Department of Hong Kong*

Because of the increasing number of drivers in the society, a number of mobile apps are available in the market to assist driving. Some mobile apps can be used as a black box to record information for car accident investigation while some
provide functions that the smart phone camera can be used to record road conditions and the GPS receiver can be used to log vehicle speed.

We think that the market of driving mobile app still have a potential for development. Although there are a variety of user-friendly driving apps in app store for drivers, nearly most of them are needed to input commands by touching on the screen on the mobile phone.

Thus, in order to provide a more convenient way for drivers to input commands to the driving app, we are going to embed a voice recognition function so that drivers can retrieve traffic information by saying some commands to the app.

### 2.2 Related Work

In our society, there have been some driving assistant devices such as GPS navigation system that make use of synchronized voice to assist drivers when they are driving. For example, the
driver can input a location where he or she are going to arrive, then the system
tell the driver with the aid of synchronized voice the details of the destination
when the driver is driving to the location such as the time left to arrive at the
destination, the distance to the destination and which road the driver should
choose in order to arrive at the location in a shorter time. This type of system
indeed brings an advantage to drivers and become very popular nowadays.

Besides, many new brands of mobile phones today have been installed a
voice recognition function such as Siri used in Iphone and Vlingo used
in most of the Android’s mobile phones. The function provides a
more convenient way for users to input commands to the phone. For
example, the mobile phone can play a song when a user just simply says “Play
music” to the phone.

All of these examples show the potential of using voice as a medium in inputting
commands and retrieving information in nowadays society is tremendous and
feasible, and the value is considerable.
2.3 Objective

In this project, we want to develop an app with a user-friendly interface and a convenient way to provide assistant to drivers. We observe that many car accidents happened in the past resulted from drivers which used their cell phones while driving.

Also, many taxi drivers use mobile phones while driving in recent years in Hong Kong. This issue raises the society’s awareness about the safety of passengers. Some legislative council members suggest regulating the problem by legislation. Thus, the main goal of our project is “Hands on the wheel, eyes on the road”.

Our app will offer drivers personalized access to content and services through deep voice integration, while ensuring seamless connectivity and end-to-end delivery of traffic information.

By using our driving app, we hope that drivers can pay more attention on the road conditions and reduce the number of car accidents resulting from using mobile apps when driving. Also, through using the app, drivers can easily get the on-time information about the traffic and road conditions in Hong Kong whenever they want.
2.4 Strategies

In our project, we have 3 main objectives. To achieve our objectives, we have identified some strategies to achieve each of the following:

In order to achieve Objective 1 – Develop a driving assistance application with user-friendly interface with simple operation, we will provide some handy ways such as buttons or a list of choices for drivers inputting data into our application and then get the desired information they want. We hope that the design of the application can be as simple as possible so that drivers can get the desired information quickly and conveniently.

To fulfill Objective 2 – Incorporate a convenient functionality into the application, we are going to embed the voice recognition function into the mobile application. We know that typing is very inconvenient and troublesome, especially when you are driving. We hope this assistive feature will be helpful and useful for the drivers.

Lastly, to accomplish objective 3 – Provide the comprehensive and updated traffic information and road conditions to drivers, data will be retrieved from the government websites time to time. Then, analysis on the data will be carried out and then the useful information will be available to the drivers. Besides, some useful tools will be specifically available such as map for users to view the shortest route when they are going to travel from one place to another place.
2.5 Smart Phone Application Analysis

Worldwide Smartphone OS Market Share
(Share in Unit Shipments)

<table>
<thead>
<tr>
<th>Period</th>
<th>Android</th>
<th>iOS</th>
<th>Window Phone</th>
<th>BlackBerry OS</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q4 2011</td>
<td>76.6 %</td>
<td>19.7 %</td>
<td>2.8 %</td>
<td>0.4 %</td>
<td>0.5 %</td>
</tr>
<tr>
<td>Q4 2012</td>
<td>78.2 %</td>
<td>17.5 %</td>
<td>3.0 %</td>
<td>0.6 %</td>
<td>0.8 %</td>
</tr>
<tr>
<td>Q4 2013</td>
<td>70.4 %</td>
<td>20.9 %</td>
<td>2.6 %</td>
<td>3.2 %</td>
<td>2.9 %</td>
</tr>
<tr>
<td>Q4 2014</td>
<td>52.8 %</td>
<td>23.0 %</td>
<td>1.5 %</td>
<td>8.1 %</td>
<td>14.6 %</td>
</tr>
</tbody>
</table>

Table 2.1 Smart Phone OS analysis, source: IDC, 2014 Q4

According to the above table, we can easily notice that Android operating system has dominated the smartphone market for several years. As major hardware and software developers such as LG and Samsung choose Android as the development platform, we have decided to develop our application on Android platform. We believe that the trend that Android dominates the OS market share will continue.

On the other hand, as we observe that iOS is the second popular operating system worldwide, we may develop our application on iOS platform in order to cater the needs of different drivers.
2.6 Amendment of Project Scope from Project Plan

After reviewing the project in these few months, some features were modified or added:

- Shortest path and the corresponding route information: Previously, we use the information of 600 something points from the transport department to find the shortest path between the starting location and destination. We initially implement the Breath-First-Search (BFS) algorithm to find the shortest path. However, we find that the shortest path between 2 points and its corresponding route information can be obtained by the web services provided by the transport department in a more accurate way. We believe that the algorithm used by the web service uses time as a weight. By showing all the possible routes between starting location and destination, the time used by each route is calculated and the total time used by each route will be compared and the route with the smallest time used will be selected as the shortest path.

Actually, after collecting the opinions at the first presentation, we are going to implement the Dijkstra’s algorithm to find the shortest path between 2 points. However, due to limited time and the main focus of our project is the voice recognition function, we prefer to allocate more time on the implementation on the voice recognition function and make our application with a high quality.
- Database table storing the road information from transport department: The table is now used to find the total time and total speed required to travel a road. A table with the main road names, the total time to travel the road and the total speed required to travel the road will be generated from the records in the database table and the table will be displayed on the application to provide the main road information to users.

- GPS function: GPS function is implemented to locate the starting point of the user and the starting point will be displayed on the map.

- Map function: A map is used by calling the Google API. It provides a graphical display of the route to the user.

- Route plotting function: It shows the route between the inputted destination and the starting point located by the GPS function of the user.

- Traffic news function: Traffic news is obtained from RTHK periodically. The traffic news is outputted by synthetic voice to drivers. Also, the traffic news related to the destination will be displayed and spoken out by the application.

- Database implementation: One database table is added for the storage of traffic news from the RTHK. The application will retrieve traffic news from the database table and display the news to the user.
On the other hand, a few features were removed:

- **Map with 600 something points:** This map will be removed. In our application, a map at present will only show the desired route and a point indicated the starting location.

- **Touch input method:** The application is voice-controlled. It means that user can only input data to the application by voice and the application will give back the desired traffic information to the user by synthetic voice. No touch input method is implemented as we hope that drivers can pay full attention on driving. We aim at keeping the input method as simple as possible and think that voice input is sufficient and it is the most appropriate input method in the application.
2.7 Project Deliverables

Our project is divided into four phases. In each phase, there are different deliverables and our team has achieved the targets of each phase.

<table>
<thead>
<tr>
<th>Phase &amp; Period</th>
<th>Deliverables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase 1 – Inception</td>
<td>Detailed project plan: It describes the overview, objective and schedule of the project in detail.</td>
</tr>
<tr>
<td></td>
<td>Project webpages: It introduces the general information of the team and the project to the public</td>
</tr>
<tr>
<td>(25 September 2014)</td>
<td></td>
</tr>
<tr>
<td>Phase 2 – Elaboration and Research</td>
<td>Preliminary implementation: Many researches are conducted in order to find the best way to develop the application. Basic functions and user interfaces as well as the voice recognition function are initially implemented.</td>
</tr>
<tr>
<td>(21 January 2015)</td>
<td>Detailed interim report: It describes the status, progress and implementations of the project at the stage.</td>
</tr>
<tr>
<td>Phase 3 – Construction and test</td>
<td>Construction: The voice recognition function is integrated into the application successfully. Also, other functionalities such as route plotting and display of traffic news are completed.</td>
</tr>
<tr>
<td>(1 April 2015)</td>
<td>Test: A lot of tests were carried out to test the validity of the functionalities implemented on the application throughout this period.</td>
</tr>
</tbody>
</table>
Phase 4 – Finalize  
(18 April 2015)

Finalized implementation: The finalized version of the application is released.

Final report:
It describes the detailed implementation and the system design of our application. Also, challenges and improvement will be included in the project.

### 2.8 Major Work Products

In this project, there are two major work products – an Android application and a project web page.

The Android application is compatible with all devices with suitable Android operating system and it runs on Android 3.0 to Android 4.2. However, all of us install the application to our smart phones with Android 4.2.

The webpages contain all general information of the project such as the information of our group, the working schedule and news.
2.9 Project Responsibility

<table>
<thead>
<tr>
<th>Team member</th>
<th>Major Tasks</th>
</tr>
</thead>
</table>
| Leung Kin Ning  | ➢ Android application  
                    ➢ Project management and related task  
                    ➢ “Route plotting” function on map  
                    ➢ Voice recognition function  
                    ➢ Version control  |
| Lee Jin Man     | ➢ Database system  
                    ➢ Documentation  
                    ➢ Graphics and publication  
                    ➢ User Interface  
                    ➢ Testing and quality control  
                    ➢ Project web page  |
| Wong Kwan Wai   | ➢ Android application  
                    ➢ Basic system implementation  
                    ➢ Server configuration  
                    ➢ Web-based system  |

The above tasks specified to each member are the main tasks for which each member is responsible. Indeed, throughout the whole FYP, we work very closely together to finish all of the tasks. All of us actively participated in both front-end and back-end design and development process. So that every member can have a better knowledge and overall understanding of how our system works besides one’s main tasks.
2.10 Project Milestones

The project is divided into four phases and there are two major milestones.

The first major milestone is the initial implementation of the Android application. It can be called the starting phase of the project. All functions are initially implemented including the voice recognition function and other basic functions. Also, the user interface and the icon of the application are designed.

The second major milestone is the finalized tested implementation. It can be called the finishing phase of the project. All of the works are finished and all functions in the application are tested thoroughly.
Section 3: Product Design

3.1 Items of the Product Function

<table>
<thead>
<tr>
<th>Major product function</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application features: Voice Recognition function</td>
<td>To allow user to input the destination to the application in order to obtain the desired traffic information. The traffic information will be spoken out by synthetic voice</td>
</tr>
<tr>
<td>Application features: Route Plotting function</td>
<td>To display the route to the user</td>
</tr>
<tr>
<td>Application features: Map function</td>
<td>To give the user a graphical display of the route and let them easily know how to reach the destination from the current location</td>
</tr>
<tr>
<td>Application features: Traffic news function</td>
<td>To display the latest five traffic news to the user</td>
</tr>
<tr>
<td>Application features: Travelling distance (in km) function</td>
<td>To display the travelling distance (in km) from the starting location to the destination</td>
</tr>
<tr>
<td>Application features: Travelling time (in minute) function</td>
<td>To display the travelling time from the starting location to the destination</td>
</tr>
<tr>
<td>Application features: GPS function</td>
<td>To capture the present location of users</td>
</tr>
<tr>
<td>Basic system implementation</td>
<td>To provide fundamental structure of the app</td>
</tr>
<tr>
<td>Database: Client and server API</td>
<td>To store the road information retrieved from the Transport department and to store the traffic news retrieved from the RTHK</td>
</tr>
<tr>
<td>User interface</td>
<td>To provide a user-friendly interface for user</td>
</tr>
</tbody>
</table>

Table 3.1 Items of the product Function
3.2 Major Product Function Descriptions

In this section, the major product functions in the application will be described in detail.

3.2.1 Application features: “Voice Recognition function”

The voice recognition function is the major function in the application. It involves almost in every sub-functions in the application.

First, in the location inputted situation, it is used to recognize and analyze the location spoken by the user to the application. Once it received the location, it will speak out the location again to the user in order to make sure the location received is the same as the location the user desired. When the location is correct, the user have to confirm the location. Once the voice engine receives the confirmation, it will analyze the location and the desired route will be displayed on the map.

Second, the voice recognition function is used to speak out the route information and the traffic news to the user by synthetic voice. The function will clearly speak out the steps to arrive the desired location inputted from the user from the present location of the user. After that, the function will speak out the latest 5 traffic news to the user.

3.2.2 Application features: “Route Plotting Function”

After receiving the location from the user, a route will be displayed on the map. It is the shortest path route from the present location of the user to the user’s desired location. The route is made in order to give a graphical display to the user that they can have a clear understanding of the route.

3.2.3 Application features: “Map Function”

The map is obtained by calling the google map API. The map is used to show the route with the starting location as the present location of the user and the destination inputted by the user.
3.2.4  Application features: “Traffic News Function”
After the destination is inputted to the application, in addition to the route on the map and the information of the route, the application will display the latest 5 traffic news on the screen from the database. The traffic news is obtained from the radio of Hong Kong (RTHK). The traffic news will be fetched from the RTHK periodically and the data will be stored in the database.

3.2.5  Application features: “Travelling Distance (in km) Function”
Similar to the traffic news function, the total travelling distance (in km) from the starting location to the destination will be displayed on the screen after the destination is inputted to the application. However, the travelling distance (in km) is obtained from the transport department given the starting location and the destination. As the shortest path route is displayed, the travelling distance (in km) is also the shortest travelling distance when compared to other routes with the same starting location and destination.

3.2.6  Application features: “Travelling Time (in minute) Function”
Similar to the travelling distance (in km) function, the travelling time (in minute) is obtained from the transport department given the starting location and the destination. As the shortest path route is displayed, the travelling time (in minute) is also the shortest travelling time when compared to other routes with the same starting location and destination.

3.2.7  Application features: “GPS Function”
The GPS function is used to mark the location of the user and it is further used as the starting location which is an important parameter in successfully obtaining the information of the desired route.
3.2.8 Database

There are two database tables which are used to store the data obtained from the radio of Hong Kong (RTHK) and the traffic information of 600 coordinate points from the Transport department respectively. The first database table consists of two fields which save the time the traffic news issued and the content of the traffic news respectively. On the other hand, the second database table contains seven fields and each record saves the traffic information of a specific road given the coordinate points in Hong Kong.
3.3 Hardware Requirement

In this project, there are some hardware requirements:
1. Device with Android OS installed is a must to install the application.
2. Device with network access is a must for transmission of data.
3. Device with GPS function opened is required to locate the starting point of the user.

3.4 System Architecture

3.4.1 Overview

3.4.1.1 Model-view-controller

Model-view-controller (MVC) is a software architectural design pattern for implementing user interfaces. It divides software into three interconnected sections, such that the system’s logic and the presentation part can be separated from each other.

In details, the main component, model, would capture the behavior of the application independent of the user interface and manage the data and logic of the application. While for the controller, it accepts and converts the input to commands for the model or view component. And for the view section, it displays the information to the user.
3.4.1.2 RESTful Web Services

REST stands for Representational State Transfer, which is a software architecture style. It consists of guidelines and practices as scalable web services can be created. And for a web service that is based on REST, it would be called as a RESTful service. In fact, most of the RESTful services communicate through the Hypertext Transfer Protocol (HTTP) as the web browsers can retrieve web pages and send data to remote servers. In details, data and functionality are considered as resources and are accessed using Uniform Resource Identifiers (URIs), typically links on the Web. For the resources, they are manipulated by using a fixed set of operations while they can be decoupled from their representation so that their content can be accessed in a variety of formats, such as HTML, XML, plain text, JSON etc.

3.4.1.3 Inversion of control container (dependency injection)

Inversion of control (IoC) is a design in which custom-written portions of a computer program receive the control from a generic, reusable library, such that the reusable code can invoke the custom or task-specific code. In other words, when a class, say, A,
has to instantiate an object of class B, the code within class A has to state explicitly that an object from class B would be created. However, by defining information in a XML file and with the help of IoC, the program just has to declare an objects of B, and the container would instantiate and injects the object from B to A automatically.

To implement inversion of control, dependency injection is one of the options. The approach injects or passes the dependencies (or services) into a dependent object (or client) rather than requiring the client to build or find the service.

The approach helps isolate the client from the impact of design changes and defects, which can increase modularity of the program and make it extensible.

### 3.4.1.4 Data Access Management

For Java to connect database for accessing data, the Java Database Connectivity (JDBC) API is the most common way to achieve the goal. The API defines how a client may access a database and provides methods for querying and updating data in a database. By collaborating with the JDBC Driver Manager which can act as a connection factory for creating JDBC connections, the workload of accessing different type of databases can be lessened.

### 3.4.1.5 Task Execution and Scheduling

The asynchronous execution and scheduling of tasks mean to execute tasks to run periodically. By adopting such an approach, certain jobs or tasks can be carried out at an exactly specified time or at regular time intervals. In fact, some native Java classes and methods are provided for developers to schedule tasks for future execution in a background thread. Meanwhile, developers can employ an open source job
scheduling library, Quartz, which can create schedules for executing jobs within any Java application. Through various implementations, a task can be scheduled and triggered at a particular time.

### 3.4.2 Spring Framework

#### 3.4.2.1 Model-view-controller

The Spring Framework model-view-controller (MVC) framework provides a flexible platform which is based on the typical standard of MVC to the developers, as developers can easily handle the connection between each component within the MVC architecture.

Under the Spring Framework design, when there is incoming request, the request will be passed from the front controller to the backend one. After handling the request, the response would be sent to the view component for rendering purpose. And finally, the control would return to the user.
3.4.2.2 RESTful Web Services

The Spring Framework provides some annotations to reduce the work of creating a RESTful service. Upon the Framework design, for building a RESTful web service, HTTP requests are handled by a controller.

All in all, the key difference between a traditional MVC controller and the RESTful web service controller is the way that the HTTP response body is created. Rather than relying on a view technology to perform server-side rendering of the data to HTML, the RESTful web service controller simply returns and converts the object into JSON format.

3.4.2.3 Inversion of control container (dependency injection)

The inversion of control container is one of the main characteristic of the Spring Framework. The service provides a means of configuring and managing the Java objects, beans, which are classes that encapsulate many objects into a single object. In other words, the container can manage an object life cycle through a series of processes including creating, calling initialization methods and configuring the objects.

In details, developer may simply describe how the beans should be created and which services and components must be called within the Spring configuration files, the container would handle the whole control process by itself.
As a whole, the container provides a consistent mechanism to configure applications and integrates with almost all Java environments. It makes the code easy to maintain and easier to test through using the container.

3.4.2.4 Data Access Management

The database connection in the Spring Framework addresses common difficulties that developers would face when working with databases in applications. Several extensive supports for working with the databases and the JDBC driver are provided, such as database resources can be automatically acquired or released, database objects can be retrieved from connection pool wrappers etc. All the features can be accessed by simply using the tailor-made template classes specified by the Framework.

In fact, the template class employed to use within the project is called JdbcTemplate class. It is the central class within the JDBC core package of Spring. The class handles the creation and release of resources and performs some basic tasks of the core JDBC workflow such as SQL statement creation, execution and extraction of the returned parameter values. The sole duty leaving to developer is to provide SQL statement and extract result set within the code.

Therefore, as the Framework takes care of all the low-level and tedious details of the JDBC API, the workload for handling the database connection has been greatly reduced.
3.4.2.5 Task Execution and Scheduling

Actually, the Framework boiled the process down to something very simple. Again, with the help of annotation support for both task scheduling and asynchronous method execution, the whole process can be simplified.

By using the annotations, methods would be invoked between the successive start times of each invocation. It means the task can be carried out and complete its duty in a periodical term.
3.4.2 Database design

In this project, we have designed two tables to store the data of the application. The first one is used to store the traffic news from the RTHK while the second one is used to store the traffic information of a specific road from transport department. The detailed design of each database table will be illustrated below.

**Table 1 (road_news)**

This table contains 4 fields and the fields store the id which is used to indicate each of the traffic news and is an auto-increment variable, the date that the traffic news is released, the time which the traffic news is released and the detail of the traffic news. The detailed information of each field in the table can be found in Appendix III.
**Table 2 (road data)**

This table contains 19 fields and the fields store the road which is specified in the road number, the starting point, the ending point, the Chinese name of the road, the English name of the road, the region the road is in, the category of the road, the Chinese starting and ending address of the road, the English starting and ending address of the road, the latitude of the starting and ending point, the longitude of the starting and ending point, the distance of the road, the traffic speed specified of the road, the road saturation level and the overview polyline points. The detailed information of each field in the table can be found in Appendix III.

After collecting the data from the transport department, the data will be stored in the database. Also, when drivers need to know the traffic information of a specific road, data will be retrieved from the database.
3.5 Software Process Model

In the software development cycle, we know that a software is developed through several steps in the cycles. From the graph 3.1 above, we know that eight main steps are needed in software development.

As the development cycle divides the software development work into distinct phases (or stages) and each stage contain the specific activities. Better planning and management of the development of the application can be achieved.

At the initial stage, a project plan was released to give a general direction of the project. Then, we move to the planning stage to start considering how to achieve the objective in the project. After that, we do some research and start developing the application. In each cycle, requirements as well as opinions are gathered and
analyzed followed by the implementation and testing of the application. At the end of the cycle, review is made on the application and the requirements to see whether improvement can be made on the application to make the application in a good quality.

At the end of the project, the finalized application is released together with the final report.
3.6 Monitoring and Controlling Mechanism

For documentation storage, there are several free cloud servers available nowadays such as SkyDrive, Dropbox and Google drive. In the project, Google drive is used to store the document, graphics and materials. We use different folders to store different types of documents. For example, one is for storing android APK and one is for storing database data.

For the version control of the application, we have considered the use of Dropbox, Github and Bitbucket at first. However, files are easily conflicted when using Dropbox and the setup of GitHub is complicated. We decided to use Bitbucket at last.

Bitbucket is a web-based hosting service for products that use either the Mercurial (since launch) or Git (since October 2011) revision control systems. It allows us to perform version control of the application easily. If there are any errors found in the latest version, we can always return back to the previous version. Besides, it is a free, simple and powerful software.

Also, in order to prevent the problem of code merging error, we have use a bug list to record the progress of each task and any bugs associated with it.
3.7 Testing

We have several system acceptance testing after a function is implemented in order to ensure that each function is implemented in a correct and accurate way. Smart phones with Android OS installed, including Sony and Samsung phone, and Samsung Galaxy Tab 10.1 are used in each testing. Before the start of each test, lists of test cases are constructed. In addition to test the accuracy of the information displayed on the screen of the application, other criteria such as loading time and transition time of data are observed during each test.

Apart from system acceptance testing, user acceptance testing is also carried out after the implementation of each function of the application. As we know that whether an application is successful or not depends on the number of users use the application. The larger the number of user uses the application, the more successful the application is and vice versa. Also, sometimes, developers will miss some crucial parts that are advantageous to users. Thus, the opinion from users is very important. We invite some of our friends with Android devices to test the application and ask them to give some feedbacks. We find that the feedbacks are very useful and valuable to help improve the quality of the application so as to cater the needs of different users and make the application in a more user-friendly way.

In user acceptance testing, a test plan is provided for users to let them be familiar with the testing purposes and the procedures. The test plan can be found in Appendix II.
Section 4: Implementation and Project Deliverables Summary

4.1 Project Deliverable

There are two deliverables including the final report and the final tested implementation in the construction phase. The final tested implementation and technical skills used in the project are illustrated in details in this group final report. On the other hand, different functions are explained in details in the individual reports of each group member which are suggested to be viewed.

4.2 The Application

4.2.1 Development Tools

4.2.1.1 Git (version control)

Git is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows. We use this as a platform to share the source codes among each other. The source codes will be stored in the BitBucket and each of us can fetch and commit the source codes and know about the progress of every member.

4.2.1.2 Spring tool suite (customized eclipse)

We adopted the Spring Tool Suite in the Eclipse platform to develop our application. The Spring Tool Suite is an Eclipse-based development environment that is customized for developing Spring applications. It provides a ready-to-use environment to implement, debug, run, and deploy your Spring applications,
including integrations for Git and Gradle.

### 4.2.1.3 Gradle

Gradle is a build automation system. Gradle can automate the building, testing, publishing, deployment and more of software packages or other types of projects such as generated static websites, generated documentation or indeed anything else.

We adopted Gradle to be our build automation system because Gradle combines the power and flexibility of Ant with the dependency management and conventions of Maven into a more effective way to build.

### 4.2.1.4 Android Studio

Android Studio is the official IDE for Android application development, based on IntelliJ IDEA. With its flexible Gradle-based build system, we can build variants and generate multiple .apk file. Also, the studio offers code templates to help us build common app features and rich layout editor with support for drag and drop theme editing.

We adopted Android Studio over Eclipse for our development because Android Studio offers better syntax highlighting, refactoring tools and better build system.

### 4.2.1.5 Apache Tomcat

Apache Tomcat is an open-source web server and servlet container developed by the Apache Software Foundation (ASF). We adopted Apache Tomcat as our web server for serving web services requests and running scheduled operations.

### 4.2.1.6 MySQL Database

We adopted MySQL Database in our project to store the data collected from the transport department and the RTHK
4.2.2 Technology used

4.2.2.1 Restful Web Services

RESTful web services are built to work best on the Web. Representational State Transfer (REST) is an architectural style that specifies constraints, such as the uniform interface. In the REST architectural style, data and functionality are considered resources and are accessed using Uniform Resource Identifiers (URIs). In the REST architecture style, clients and servers exchange representations of resources by using a standardized interface and protocol.

4.2.2.2 Spring Framework

The Spring Framework is an open source application framework and inversion of control container for the Java platform. We utilized some of the services provided the framework, such as Dependency Injection, Spring MVC web application and RESTful web service framework.

4.2.2.3 Google Maps Android API

With the Google Maps Android API, we can add maps based on Google Maps data to our application. The API automatically handles access to Google Maps servers, data downloading, map display, and response to map gestures. We make use of API calls to add markers and polylines a basic map. Each marker is associated with a road ID which is provided to our server in order to obtain traffic conditions of certain roads.
4.2.3 User Interface

4.2.3.1 Design Strategy and Benefits

In designing the interface of the application, we have one key factor that is to keep the operation of the application as simple as possible so as to provide a user-friendly way for user to use the application. Other factors include:

4.2.3.2 Main Page

Once user click on the application icon in their phone’s screen, he will directly launch on the main page of the application. With the aim to keep our application as simple as possible, our application consists of only one page. In the main page, you will see a map on the top together with the Start and Reset at the bottom. The Start button is used to start the application’s service while the Reset button is used to restart the application’s service. A red indicator is present on the map to locate the current situation of the user.

Fig 4.1 Main page of the application
4.2.3.3 Voice Input Procedure

Once the user click on the **Start** button, a microphone box will pop up which is used to receive the user’s voice. The user can then speak out the desired destination to the application.

![Pop up box](image)

Fig 4.2 Pop up box
4.2.3.4 Map and Route Plotting

After finishing the procedures of receiving and analyzing the destination inputted by the user, a map will indicate the user's starting location by a red indicator. Also, a route in blue color will be shown on the map.

4.2.3.5 Route Information – Shortest Path Display

The blue route shown on the map is the shortest path from the starting location of the user to the destination. It is the shortest path route when compared to other possible routes between the two points.

4.2.3.6 Route Information – Travelling Distance Display

The travelling distance of the corresponding shortest path is displayed at the bottom right hand side of the main page.
4.2.3.7 Route Information – Travelling Time Display
The travelling time of the corresponding shortest path is displayed at the bottom middle side of the main page.

![Fig 4.5 Travelling time]

4.2.3.8 Route Information – Destination
The destination of the corresponding shortest path is displayed at the bottom left hand side of the main page.

![Fig 4.6 Destination]

4.2.3.9 Route Information
The name of the roads that the shortest path route consists of will be displayed in a tabular form on the left hand side of the main page showing from the road near the starting location to the road near the destination. This shows to user how to arrive at the destination from the starting location.

![Fig 4.7 Route information]
4.2.3.10 Traffic News Display

The corresponding traffic news according to the destination will be displayed on the right hand side. For example, if the destination is Glocester Road, the traffic news related to Glocester Road will be selected from the database table and displayed on the application.

![Traffic news display](image)

Fig 4.8 Traffic news
4.2.4 Overall Application Design

In the project, we applied the 3-tier system architecture in which presentation, application processing, and data management functions are physically separated. By applying this client-server architecture, the application can be created with a high flexibility as an application is separated into tiers, different functions can be modified independently without affecting other functions or even reworking the entire application. This brings convenient to us in developing the application.

Also, we apply the thin-client technology in the project. We hope to minimize the processing burden of the target user’s phone as lower as possible so most data processing works are performed in the server.

We know that there are different types of Android phones with different standards nowadays. However, some of the Android phones’ processing power are low and they have low random access memory (RAM) in their central processing unit (CPU). In order to cater for more low specification phone, we aim at keeping our application as simple as possible and the application mainly provides the 2 most important functions – the speech recognition function and the text-to-speech function to users. Besides, the application will call the web services of the back-end server to retrieve the corresponding information or services. Through this way, we think that the bandwidth used by the phone reduces as the bandwidth used in exchanging messages between the application and the phone to retrieve the data is much smaller than retrieving the data by the application directly.
4.2.5 Summary of technical skills used in functions

4.2.5.1 Overview – Speech Recognition & Text-to-Speech

Android has built-in APIs for speech recognition and Text-to-Speech (TTS). The speech recognition API allows devices to collect audio from users and convert it to text. TTS allows the device to go in the reverse direction, converting text into audio.

Speech Recognition Features

- It utilizes a remote server to process audio recorded by the device. This creates a small delay in the recognition and makes it require Internet access to function.
- Android’s speech input dialog is the standard interface for collecting speech.
- Android packages the recognition result into a List<String> of potential speech-to-text conversions with a confidence score attached to each.

Text-to-Speech (TTS) Features

- Supports a limited set of languages.
- Each language has only one voice.
- Runs on the device without an Internet connection.
- Sometimes mispronounces words, but there is no way to change pronunciation. Fortunately, sometimes new releases of Android contain updates to the TTS functionality that improves pronunciation.
4.2.5.1.1 Speech Recognition

Using the speech recognition API involves sending an Intent with the various actions and extras defined in RecognizerIntent. To implement speech recognition, our app extends SpeechRecognizerActivity and configures the appropriate RecognizerIntent, interprets recognition results, and handles with any errors.

- SpeechRecognizingActivity: An abstract Activity to handle interpreting the onActivityResult() response.

The Speech Recording Process

1. App sends a RecognizerIntent.
2. User waits up to several seconds.
3. Speech prompt dialog appears with a prompt.
5. App records until a minimum time passes and it hears silence for long enough.
7. If an error occurs, the device plays a beeping sound, vibrates, and displays a retry dialog.
8. Android returns results to the app via onActivityResult().

Configuring and Processing the Result

When the user completes a recognition, the recognizer returns some results within an Intent that contains the results.
4.2.5.1.2 Text-to-Speech

To use TTS, the app must perform the following steps:

1. Initialize the TextToSpeech object. Verify that the device supports the desired language, download additional data if necessary, and wait for an asynchronous TTS engine initialization process to complete.

2. Operate the TextToSpeech API to play speech.

3. Implement an Activity that handles managing the TextToSpeech life cycle as well as any user interactions that are required during initialization or speaking.

Playing Speech with TTS

TextToSpeech manages a play queue and plays each piece of audio one at a time as it receives it. To add speech to the end of the playback queue, our app invokes speak() of TextToSpeech with text to speak and queue mode QUEUE_ADD as parameters.

4.2.5.2 Application feature: “Route Plotting Function”

Polyline The Polyline class defines a set of connected line segments on the map. A Polyline object consists of a set of LatLng locations, and creates a series of line segments that connect those locations in an ordered sequence.


4.3 Difficulties and Possible Solutions

4.3.1 Challenge in Designing a Speech-enabled Application

Allowing users to speak to their devices is challenging to design and implement. Android supports speech input and output with APIs for speech recognition and TTS. Using the APIs effectively is only part of the task. The other part is designing and implementing a complete voice user interface (VUI) with all its supporting components.

A VUI is a user interface that utilizes a user’s speech as input and synthesized speech for output. A VUI consists of a set of voice actions, where each voice action allows the user to perform a certain task.

Constrain Speech Input to Increase Accuracy

Apps cannot easily understand unconstrained speech. Human speech is highly variable and entails a large vocabulary. Although this breadth makes speech extremely expressive, it also makes it difficult for an app understand, and hence unable to take full advantage of humans’ ability to communicate.

Therefore, when designing a VUI, we spent a great effort in minimizing the complexity of allowed speech input.
Prompt the Users so They Know What to Say

A prompt is text spoken or displayed to users. In our app, we use the conventions of speech to suggest what the user should say in a reply. For example, a prompt such as “Your destination is The University of Hong Kong. Please confirm.” would encourage the user to say “confirm” in response.

Confirm Success and Help Users Recover from Errors

Speaking a voice command is risky because it could fail for many reasons. Errors are possible in a VUI due to the uncertainty of the recognition accuracy. We need to provide a way for users to recover from any mistakes they or the app make. At the same time, we don’t want to overwhelm users with too many confirmations, which can be tiring. We also need to allow users to confirm that their voice command was successful.

Therefore, we prompt for explicit confirmation. For example, “Your destination is The University of Hong Kong. Please confirm.”

4.3.2 Application design

In this project, our aim is “Hand on wheel, eye on road”. We hope that drivers can pay full attention on the road conditions in order to prevent the occurrence of unnecessary traffic accidents.

Although our application contains only 2 buttons which are the “start” and “reset” button, our ultimate aim is that driver can use our application without clicking on any of the buttons on the screen. The button may be only used as the
last measure to control the application if the voice recognition function is malfunction. However, it is difficult to avoid drivers clicking on the buttons with the present of buttons. Although it can implement an application with no buttons appeared on the main page, it is difficult to implement such an application and lots of functional testing needed to perform to ensure the accuracy and validity of the functions. Although our application contains only 2 buttons which are the “start” and “reset” button, our ultimate aim is that driver can use our application without clicking on any of the buttons on the screen. The button may be only used as the last measure to control the application if the voice recognition function is malfunction. However, it is difficult to make the drivers use our application totally by voice as they have to start the services by clicking on the “start” button or reset the application by clicking on the “reset” button.

4.3.3 Code sharing conflict

As we develop the application together, we share the codes using the online Bitbucket system. However, problem arises in version control easily because of code conflicts. Most of the time, codes are inter-dependent. For example, one portion of code in implementing a function A depends on the other portion of code. Sometimes, when one of us is deal with a portion of code in function A but he has not finished the code yet or the code still consists of some bugs, normally function A is not finished implementing at this time. When the other one who has already finished the other portion of code in implementing function A and try to merge the two portions of codes of function A together and compile the code. Error will occur and the large amount of time is wasted in fixing the code merging error.
In order to prevent the situation described above, we think of an idea that we can divide a big function into many small portions of independent tasks and each of us is responsible for some of the tasks. Also, we should avoid working on the same task at the same time.

Also, an online document bug list can be created in Google drive to track the progress of each task. When some bugs in codes have not been fixed, the bug will be recorded in the bug list with the corresponding file name. This provides an efficient method to handle the code merging problem.
4.4 Future Development

4.4.1 Starting point relocation when travelling the route

In order to make the application more user-friendly to users, we plan to implement a function that when the user is travelling on the shortest path, the route displayed on the map will shorten when time passes. In other words, the GPS function will relocate the current location of the user periodically when user is heading to the destination. This function gives a clear idea to the user about how far is the user’s current position from the destination.

Also, sometimes drivers will not strictly follow the shortest path and it will cause a deviation from the normal shortest path route on the map. The relocation of starting point function allows user to know whether they strictly follow the shortest route or not. If they find their location deviates from the route, they can click on the “Reset” button and choose another shortest path route from their current location.

However, due to limited time, this function will be put in the future development.

4.4.2 Speaking out the road names when travelling the route

Sometimes, when drivers are on the way to the destination, they do not know whether they have already passed the road displayed on the route information table on the screen.

A possible way to deal with this problem is once the user passes the road, the application will speak out the next road the user has to pass. This ensures the user can arrive at the destination through passing all the roads specified in the shortest path. This function requires the GPS function which needs to track the current location of the user. Once the GPS function matches the location to the location on the route information table, the application will give a signal to the
user that a specific road is passed and speak out the next road in the table.

4.4.3 iOS development

According to table 2.1 on page 12, we know that iOS has the second largest market share in the smartphone OS worldwide. In order to cater for the need of more people, we plan to develop an iOS version of the application.

However, the programming language and framework used to build an iOS application is significant different from that used to build an Android application. Large amount of time is needed to implement this idea.

4.4.4 Button selection by voice

In order to avoid drivers clicking on the phone’s screen while driving, we are planning to add a new function that drivers can select the “start” and “reset” button by voice instead of by touch. We suggested that drivers can use earphone while driving and use their voice to control the application. Also, as many earphones have voice cancellation function to eliminate the noisy nowadays, this can increase the accuracy of the voice recognition function.
## Section 5: Schedule

<table>
<thead>
<tr>
<th>Period</th>
<th>Description of Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>11th – 18th September</td>
<td>Initial Planning of the project&lt;br&gt; - Feasibility study&lt;br&gt; - Project scope Determination&lt;br&gt; - Gather user requirement&lt;br&gt; - Work plan Development</td>
</tr>
<tr>
<td>20th September – 25th October</td>
<td>Project first deliverables&lt;br&gt; - Preparation of project plan&lt;br&gt; - Preparation of web page</td>
</tr>
<tr>
<td>4th October – 25th October</td>
<td>Preliminary implementation&lt;br&gt; - Architecture design&lt;br&gt; - Database &amp; GUI design&lt;br&gt; - Database &amp; GUI construction&lt;br&gt; - Program construction</td>
</tr>
<tr>
<td>1th November – 15th November</td>
<td>Preparation of first presentation and interim report</td>
</tr>
<tr>
<td>15th January – 21th January</td>
<td>Project second deliverables&lt;br&gt; - Preliminary implementation&lt;br&gt; - Detailed interim report</td>
</tr>
<tr>
<td>23th January</td>
<td>First presentation</td>
</tr>
<tr>
<td>28th January – 10th April</td>
<td>Project third deliverables&lt;br&gt; - Program implementation&lt;br&gt; - Program testing</td>
</tr>
<tr>
<td>19th April</td>
<td>Project fourth deliverables&lt;br&gt; - Finalized implementation&lt;br&gt; - Final report</td>
</tr>
<tr>
<td>20th April – 3rd May</td>
<td>Preparation of final presentation and project exhibition</td>
</tr>
<tr>
<td>4th May</td>
<td>Final presentation</td>
</tr>
<tr>
<td>5th May</td>
<td>Project exhibition</td>
</tr>
</tbody>
</table>
Section 6: Balance Sheet and Resource Allocation

A maximum of HKD4,000, financed the project by the Department of Computer Science, The University of Hong Kong as budget, was spent as below:

<table>
<thead>
<tr>
<th>Item</th>
<th>Amount (HKD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Machine Borrowing</td>
<td>700</td>
</tr>
</tbody>
</table>

Table 6.1 Table of balance sheet
References

GovHK: Traffic Speed Map, Retrieved October 11, 2014 from

Google Maps API: Android, Retrieved October 13, 2014 from
https://developers.google.com/maps/documentation/android

Geodetic Survey Section: Web-based Transformation Tool, Retrieved October 19, 2014 from

Google Developers: Google Maps API, Retrieved October 19, 2014 from
https://developers.google.com/maps/?hl=zh-tw

GitHub Gist: Graph Adjacent List, Retrieved October 21, 2014 from
https://gist.github.com/tnhansel/11441647

Jackson contains JSON data-binding functionality and tree-model for Jackson Data Processor, Retrieved March 28, 2015 from
https://github.com/FasterXML/jackson-databind/

Great Android Sensing Toolkit (GAST) is a toolkit for using Android’s Sensing capabilities., Retrieved from 29, 2015 from
https://github.com/gast-lib/gast-lib
Appendix I: Main Application Flow (General Overview)

Main flow chart:

Driver’s View

- Open the app. while driving
- Input the desired destination to the app. by voice
- Drivers receive the traffic information and route plotting on map
- Drivers receive the traffic news in chronological order
- Close the app. when it is not used

Application’s View

- Analyze the voice commands
- Desired traffic information is obtained
- Retrieve the appropriate traffic data from database
- Use synchronized voice to output the information to drivers
- Display the information on the screen
- Get the desired traffic information from the Transport Department
- Store the data into a database
- Retrieves certain traffic news from the RTHK website (periodically)

1. Route information
2. Travelling distance (in km)
3. Travelling time (in minute)
Appendix II: Main Application flow (Front-end view)

This flow chart illustrates the main flow in the app. The entire process, from inputting the data into the phone to get the desired information from the phone, will be shown in a user-friendly way.

- Open the application
- A box will pop up and the user has to input the destination at the time
- The voice engine tries to recognize the voice input
  - Recognize the voice input successfully?
    - Yes
      - A box will pop up and the user has to confirm the destination inputted
    - No
      - Change destination
- The destination will be translated into coordinates by Google API
  - The destination’s coordinates together with the starting location’s coordinates obtained will be used in the web services provided by transport department to obtain the shortest path between the 2 points
- Route information
- Route on map
- Traffic News
  - Traffic news obtained from RTHK will be stored in database
  - Display
  - Display
  - The destination will be translated into coordinates by Google API
Appendix III: Main Application flow (Back-end view)

Restful Web Service by HTTP GET request

JSON

Web server

Fetch traffic information

Get traffic information

Store Data

Get Data

Transport department, HKSAR

Fetch traffic news

Get traffic news

DB server
Appendix IV: Database table structure

The structure of the 2 database tables will be illustrated below.

Table 1 (road_news)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>AutoNumber</td>
<td>11</td>
<td>Primary key of the table. It is used to indicate the traffic news</td>
</tr>
<tr>
<td>date</td>
<td>Text</td>
<td></td>
<td>The date the news is released</td>
</tr>
<tr>
<td>time</td>
<td>Text</td>
<td></td>
<td>The time the news is released</td>
</tr>
<tr>
<td>news</td>
<td>Text</td>
<td></td>
<td>The detail of the news</td>
</tr>
</tbody>
</table>

Table 2 (road_data)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>link_id</td>
<td>Varchar</td>
<td>13</td>
<td>Road ID</td>
</tr>
<tr>
<td>start_node</td>
<td>Int</td>
<td>6</td>
<td>Starting point</td>
</tr>
<tr>
<td>end_node</td>
<td>Int</td>
<td>6</td>
<td>Ending point</td>
</tr>
<tr>
<td>name_tc</td>
<td>Varchar</td>
<td>125</td>
<td>Chinese name of the road</td>
</tr>
<tr>
<td>name_en</td>
<td>Varchar</td>
<td>70</td>
<td>English name of the road</td>
</tr>
<tr>
<td>region</td>
<td>Varchar</td>
<td>2</td>
<td>Region the road is in</td>
</tr>
<tr>
<td>road_type</td>
<td>Varchar</td>
<td>11</td>
<td>Category of the road</td>
</tr>
<tr>
<td>start_address_tc</td>
<td>Varchar</td>
<td>46</td>
<td>Chinese starting address of the road</td>
</tr>
<tr>
<td>start_address_en</td>
<td>Varchar</td>
<td>72</td>
<td>Chinese ending address of the road</td>
</tr>
<tr>
<td>start_location_lat</td>
<td>Decimal</td>
<td>9,7</td>
<td>Latitude of the starting point</td>
</tr>
<tr>
<td>start_location_lng</td>
<td>Decimal</td>
<td>10,7</td>
<td>Longitude of the starting point</td>
</tr>
<tr>
<td>end_address_tc</td>
<td>Varchar</td>
<td>52</td>
<td>English starting address of the road</td>
</tr>
<tr>
<td>end_address_en</td>
<td>Varchar</td>
<td>72</td>
<td>English ending address of the road</td>
</tr>
<tr>
<td>end_location_lat</td>
<td>Decimal</td>
<td>9,7</td>
<td>Latitude of the ending point</td>
</tr>
<tr>
<td>end_location_lng</td>
<td>Decimal</td>
<td>10,7</td>
<td>Longitude of the ending point</td>
</tr>
<tr>
<td>distance</td>
<td>Varchar</td>
<td>7</td>
<td>Distance of the road</td>
</tr>
<tr>
<td>traffic_speed</td>
<td>Int</td>
<td>3</td>
<td>Traffic speed specified of the road</td>
</tr>
<tr>
<td>road_saturation_level</td>
<td>Varchar</td>
<td>15</td>
<td>Road saturation level</td>
</tr>
<tr>
<td>overview_polyline_points</td>
<td>Varchar</td>
<td>100</td>
<td>Overview polyline points</td>
</tr>
</tbody>
</table>
Appendix V: Test plan

This test plan is distributed to the testers for user acceptance testing.

<table>
<thead>
<tr>
<th>Version</th>
<th>Change Date</th>
<th>Author</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>25/03/2015</td>
<td>Jonathan Lee</td>
<td>Initial version</td>
</tr>
</tbody>
</table>

1. Introduction
This document describes the procedures used for testing the application, JDK on Android devices with Android 3.0 to Android 4.2.

1.1. Scope
This document outlines the testing procedures for the testing of JDK, which is a product from Smart Application to Assist Driving, Final Year Project for Department of Computer Science, The University of Hong Kong in 2014-2015.

In the testing, “JDK.apk” is provided and installed in the Android devices, including tablets and smart phones. However, in this project, we mainly focus on Android smart phones.

The purpose of the testing is to evaluate on the performance of the application and collect opinions from the users. Also, test cases will be provided to the users to help them finish the testing. Any unexpected results in the testing will be marked for future amendment.

2. Detail
In this part, detailed description of each testing steps are provided.

2.1. General description of testing procedures
In performing each test case, testers should follow the steps below to install the application, perform the testing and record the result:

2.1.1. Ensure the device with network access before starting the testing procedures
2.1.2. Connect the device to user-preferred computer
2.1.3. Load “JDK.apk” into the device
2.1.4. Find and tap “JDK.apk” in the device for installation
2.1.5. Tap the application icon to launch the application
2.1.6. If launched successfully, run the test cases in section 2.2 one by one and record the result – either success or fail
2.1.6.1. If failed to launch, take a screen shot
2.1.6.2. If the application has no response in any test cases, record the issue in the record table in Section 3 and take a screen shot
2.1.7. Record all the result of test cases in the result table in Section 3
2.1.8. Send the result to jlee1213@connect.hku.hk

2.2. Test cases
Please open the GPS function on the device before performing the test cases below.

2.2.1. “GPS” function
Steps:
1. Click “Start” from the start page to enter the main page
2. Say the destination as “Causeway Bay” after hearing the instruction to enter the destination from the device
3. Say “Confirm” to the device after receiving a confirmation message from the device
4. Look at the map to check whether the starting location marked at the map is the same as your actual location
5. Click the “Reset” button to finish the test

2.2.2. “Voice Recognition” function
Steps:
1. Click “Start” from the start page to enter the main page
2. Say the destination as “Causeway Bay” after hearing the instruction to enter the destination from the device
3. Say “Confirm” to the device after receiving a confirmation message from the device
4. Listen to the destination spoken out by the device and check whether the destination is the desired one
5. Listen to the route information spoken out by the device and check whether they are same as those displayed on the screen.
6. Listen to the traffic news spoken out by the device and check whether they are same as those displayed on the screen.
7. Listen to the travelling time spoken out by the device and check whether it is same as the one displayed on the screen.
8. Listen to the travelling distance spoken out by the device and check whether it is same as the one displayed on the screen.
9. Click the “Reset” button to finish the test
2.2.3. “Route Plotting” function

Steps:
1. Click “Start” from the start page to enter the main page
2. Say the destination as “Causeway Bay” after hearing the instruction to enter the destination from the device
3. Say “Confirm” to the device after receiving a confirmation message from the device
4. Look at the map to check whether the line plotting on the map indicating the true starting location and true destination or not
5. Click the “Reset” button to finish the test

2.2.4. “Map” function

Steps:
1. Click “Start” from the start page to enter the main page
2. Check if the map is present in the page or not
3. Click the “Reset” button to finish the test

2.2.5. “Traffic News” function

Steps:
1. Click “Start” from the start page to enter the main page
2. Say the destination as “Causeway Bay” after hearing the instruction to enter the destination from the device
3. Say “Confirm” to the device after receiving a confirmation message from the device
4. Check if the traffic news are present in the page or not
5. Count if there are totally 5 latest traffic news displayed in the page or not
6. Click the “Reset” button to finish the test

2.2.6. “Travelling Distance (in km)” function

Steps:
1. Click “Start” from the start page to enter the main page
2. Say the destination as “Causeway Bay” after hearing the instruction to enter the destination from the device
3. Say “Confirm” to the device after receiving a confirmation message from the device
4. Check if the traveling distance is present in the page or not
5. Click the “Reset” button to finish the test
2.2.7. “Travelling Time (in minute)” function
Steps:
1. Click “Start” from the start page to enter the main page
2. Say the destination as “Causeway Bay” after hearing the instruction to enter the destination from the device
3. Say “Confirm” to the device after receiving a confirmation message from the device
4. Check if the traveling time is present in the page or not
5. Click the “Reset” button to finish the test

2.2.8. User-proposed destination testing
Repeat 2.2.1 – 2.2.7 by using a self-proposed destination (e.g. Mong Kok)

2.2.9. User Interface test
Test the graphical user interface as you preferred and check for any unexpected user interface behavior

2.2.10. Load test
Repeat any of the test cases 2.2.1 – 2.2.7 for three times per testing set (1 set includes 1 default destination testing and 1 user-proposed destination testing)

3. Result table
The table is used for recording the testing results of each test case. Actually, there are 2 result tables, 1 is for recording the result for default destination (i.e. Causeway Bay) and the other is for recording the result for user-proposed destination.

Result table 1 – Default destination (Causeway Bay)
- Please write down the location you performed the test:
  ______________________________________________________________
- Please write down the time you performed the test:
  ______________________________________________________________

<table>
<thead>
<tr>
<th>Launching (Please record the launching time in second in the comment box)</th>
<th>Success (✓) / Fail (✗)</th>
<th>Comment / Reason for failure</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Success (✓) / Fail (✗)</td>
<td>Comment / Reason for failure</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>“GPS” function (Please record the location marked by GPS function in the map in the comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Voice Recognition” function (Please specify the problem in details and clearly if any problem(s) is found)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Route Plotting” function</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Map” function</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Traffic News” function (Please specify the times of the latest 5 traffic news in the comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Travelling Distance” function (Please specify the travelling distance (in km) in the comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Travelling Time” function (Please specify the travelling time (in minute) in the comment box)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Result table 2 – User-proposed destination**

- Please write down the location you performed the test:
  ________________________________________________________________

- Please write down the time you performed the test:
  ________________________________________________________________

<table>
<thead>
<tr>
<th>Activity</th>
<th>Success (✓) / Fail (✗)</th>
<th>Comment / Reason for failure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Launching (Please record the launching time in second in comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Success (✓) / Fail (✗)</td>
<td>Comment / Reason for failure</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>“GPS” function (Please record the location marked by GPS function in the map in the comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Voice Recognition” function (Please specify the problem in details and clearly if any problem(s) is found)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Route Plotting” function</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Map” function</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Traffic News” function (Please specify the times of the latest 5 traffic news in the comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Travelling Distance” function (Please specify the travelling distance (in km) in the comment box)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Travelling Time” function (Please specify the travelling time (in minute) in the comment box)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**General result table**

<table>
<thead>
<tr>
<th>Test</th>
<th>Success (✓) / Fail (✗)</th>
<th>Comment / Reason for failure</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Interface test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Load test (please state the test case used in the comment box)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Other opinion / suggestion / bug report**