REAL TIME BUS TRACKING SYSTEM

MUHAMMAD ADDEIB BIN MOHD SUBARI

BACHELOR OF COMPUTER SCIENCE (SOFTWARE ENGNEERING)

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name   : MUHAMMAD ADDEIB BIN MOHD SUBARI

Date of Birth   : 13 MARCH 1996

Title   : REAL TIME BUS TRACKING SYSTEM

Academic Session   : 2018/2019

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☑	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_____

    (Student's Signature)   (Supervisor's Signature)

      960313435055   Dr. Adzhar Bin Kamaludin

New IC/Passport Number   Name of Supervisor

Date: 8 JANUARY 2019   Date: 8 JANUARY 2019

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name	Muhammad Addeib Bin Mohd Subari
Thesis Title	Real Time Bus Tracking System

Reasons	(i)
	(ii)
	(iii)

Thank you.

Yours faithfully,

_____
      (Supervisor's Signature)

Date: 8 JANUARY 2019

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and, in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering) with Honors.

_____

(Supervisor's Signature)

Full Name   : DR. ADZHAR BIN KAMALUDIN

Position   : SENIOR LECTURER

Date   : 8 JANUARY 2019

_____

(Co-supervisor's Signature)

Full Name   :

Position   :

Date   :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

_____

(Student's Signature)

Full Name   : MUHAMMAD ADDEIB BIN MOHD SUBARI

ID Number   : CB15133

Date   : 8 JANUARY 2019

REAL TIME BUS TRACKING SYSTEM

MUHAMMAD ADDEIB BIN MOHD SUBARI

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Software Engineering)

Faculty of Computer System & Software Engineering

UNIVERSITI MALAYSIA PAHANG

JANUARY 2019

ACKNOWLEDGEMENTS

ABSTRAK

Rapid Kuantan adalah jenama bas milik Prasarana Malaysia Berhad untuk menjalankan perkhidmatan bas di Kuantan, Pahang, Malaysia. Apabila menaiki bas, orang ramai ingin tahu masa ketibaan mereka kerana kadang-kadang jadual bas tidak mempunyai kebolehpercayaan dari segi ketepatan masa. Salah satu masalah utama ialah pengguna bas tidak mengetahui masa sebenar bas ketibaan. Pengurusan bas hanya menyediakan jadual waktu bas untuk pengguna. Pengguna bas perlu keluar sebelum perhentian bas untuk menunggu bas. Ini kerana mereka tidak mahu ketinggalan bas yang mereka targetkan pada jadual waktu bas. Kadang-kadang, pengguna bas perlu menunggu terlalu lama untuk bas ketibaan kerana mereka tidak tahu masa yang tepat bas akan tiba dan ketika mereka ingin mengejar masa untuk alasan tertentu mereka akan menjadi marah. Dengan membazirkan banyak masa menunggu bas, masa itu sebenarnya boleh melakukan sesuatu yang bermanfaat. Objektif utama adalah menerapkan teknologi pengesanan GPS ke dalam sistem pengangkutan bas. Ini adalah untuk mencadangkan kepada admin bas untuk memasang aplikasi Sistem Penjejakan Bas yang dapat menjejaki kedudukan sebenar bas. Seterusnya, aplikasi ini akan menghantar data kedudukan bas ke pangkalan data dan data akan dipaparkan sebagai ikon bas bergerak di peta. Objektif seterusnya adalah untuk menyediakan lokasi sebenar bas yang berfungsi bergerak. Lokasi yang tepat akan membantu pengguna menguruskan masa mereka kerana tidak menunggu terlalu lama untuk bas. Sistem Penjejakan Bas Masa Sebenar telah dibangunkan dengan menggunakan kaedah Rapid Application Development (RAD). Ini kerana kaedah ini boleh membuat proses pembangunan pesat. Permohonan ini akan digunakan oleh ramai orang pada masa akan datang. Manfaat aplikasi ini dapat mengurangkan masa menunggu bas dan membantu orang untuk menetapkan waktu apa yang mereka akan tiba di destinasi.

ABSTRACT

Rapid Kuantan is a bus brand owned by Prasarana Malaysia Berhad to conduct bus services in Kuantan, Pahang, Malaysia. When travelling in buses, public want to know their exact arrival time because sometimes the bus schedule does not have the reliability in terms of punctuality. One of the main problems is the bus's user did not know the exact time of the arrival bus. Bus management only provide bus's timetable for the user. The bus's user needs to go out earlier to the bus stop to wait for the bus. This is because they did not want to miss the bus that they target at the bus's timetable. Sometimes, the bus's user needs to wait for too long for the arrival bus because they did not know the exact time the bus will arrive and when they want to chasing time for the certain reason they will become angry. By wasting a lot of time waiting for the bus, that time actually can do something else beneficially. The main objective is to apply GPS tracking technology into bus transportation system. This is to propose to the bus admin to install the Bus Tracking System application that able to track the actual position of the bus. Next, this application will be sending position data of the bus to database and the data will be display as moving bus icon on the map. Next objective is to provide the exact location of the moving working bus. The exact location will help user to manage their time for not waiting too long for the bus. Real Time Bus Tracking System was developing by using Rapid Application Development (RAD) methodology. It is because this method can make rapid development process. This application will be used by many people in the future. The benefits of this application are it can reduce bus waiting time and help people to set what time they will arrive at destination.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVATION	TITLE
API	Application Programming Interface
APP	Application
APU	Asia Pacific University
GPS	Global Positioning System
IDE	Integrated Development Environment
NIU	Northern Illinois University
RAD	Rapid Application Development
RFID	Radio Frequency Identification
RTBTS	Real Time Bus Tracking System
SDD	Software Design Document
SDLC	Software Development Life Cycle
SRS	Software Requirement Specification
Wi-Fi	Wireless Fidelity

CHAPTER 1

INTRODUCTION

1.1   INTRODUCTION

Rapid Kuantan is a bus services owned by Prasarana Malaysia Berhad to conduct bus services in Kuantan, Pahang, Malaysia. Rapid Kuantan was launched on 1 December 2012.(MyRapid, 2012).

When travelling in buses, public want to know their exact arrival time because sometimes the bus schedule does not have the reliability in terms of punctuality. Thus, the bus arrival information system is a service that aim to provide a friendlier bus service. This project has been proposing to develop a prototype application for public to track the position of the bus by installing the application that been develop through out of this project. This application is able to receive and send GPS data, and bus distance.

1.2   PROBLEM STATEMENT

There is no easy way for travellers to check the location of buses in real time. Bus travellers want to know the bus's accurate arrival times because they tend to maximize their personal time to wait for the bus. Peoples usually go to the bus stop early to wait for the bus. Although the bus timetable is available for free on the Web, bus operators often do not follow, and another factor have been proved that the schedule provides limited information to the public. The peoples will disappoint when their miss a bus trip before, and the next bus service will cause them late for their arrangement. A long wait at the bus stop can make people angry and will make them unwilling to take the bus for the next day. In the end, they decide to use their own transport to go to other places, which resulted in increased traffic jam.

One of the main problems is the bus's user did not know the exact time of the arrival bus. Bus management only provide bus's timetable for the user. The bus's user needs to go out earlier to the bus stop to wait for the bus. This is because they did not want to miss the bus that they target at the bus's timetable. Sometimes, the user needs to wait for too long for the arrival bus because they did not know the exact time the bus will arrive and when they want to chasing time for the certain reason, they will become angry. By wasting a lot of time waiting for the bus, that time actually can do something else beneficially.

1.3   OBJECTIVE

The aim of this project is to develop an application what able to track the location of the bus by using GPS technology. In order to achieve this aim, the following objectives are listed:

   i.   To study how GPS can connect with application

   ii.   To design and implement the real time application that can track bus location.

   iii.   To test the proposed application in term of functionality.

1.4   SCOPE

The scope of this project is limited on bus trip number 100. GPS will be use as main technology in this project. The position of the bus is based on real time database.

1.5   THESIS ORGANIZATION

This thesis consists of five (5) chapters. Chapter 1 shall discuss about the introduction of the project. Chapter 2 would be Literature Review that discuss about the comparison with the other project that already exist and the compared technology used. Following by Chapter 3, Methodology shall discuss about methodology of the project. Chapter 4 will discuss about the implementation on develop this project. Also, testing will be performed in this chapter too. For the last Chapter is Conclusion for this project that will provides the summary of the report for this project.

CHAPTER 2

LITERATURE REVIEW

2.1   INTRODUCTION

There are many bus transportation services in every country, but many of them are lack of management system. In these days, majority of bus services are not performing the real time bus tracking system. This is because they only provide scheduled timetable to follow which are not accurate in real time. By having the GPS tracking system, it will describe the location of the current bus.

However, some countries have already succeeded in implemented the real time bus tracking system. There are many universities in oversea country that already implemented the bus tracking system for their shuttle bus service.

2.2   STUDIES ON EXISTING SYSTEMS

There are many existing bus tracking systems implemented by other universities. The university that already implemented this bus tracking system are Rice University, Northern Illinois University (NIU) and Asia Pacific University (APU) (LOONG, 2013).

2.2.1   Bus Tracking System by Asia Pacific University (APU)

Figure 2-1   Screenshot for APU Road map bus tracking system

Figure 2-2   Screenshot for APU Satellite imagenary bus tracking system

The APU bus tracking systems that are showed in Figure 2-1 and Figure 2-2 are developed by Malaysia premier private university – Asia Pacific University (APU). This system is built in APU webpage and embedded with Google Map. It allows APU student to track the bus traffic status anywhere as long they have device that connect to internet.

The bus icons are shown on the map of the system and it indicated the real time position on where the bus is. Those icons shown include with bus number and plate number of the bus. The bus number is shown to ensure that student track the right bus when it reaches the bus stop. Each bus will have the different route and different timetable to follow. The timetable will be shown to students about the bus arrival time at the bus stop.

However, this system is still not complete yet because the timetable in this system is the departure time of moving bus. So, the still cannot provide the real arrival time for each bus. Luckily, the student can see the icon of bus is moving to the bus stop on the web page. The webpage will refresh every 10 seconds.

Next, there are many destination buses stop of a bus route that is shown on the web page. However, the most destination bus stop will only appear on the web page and not the whole destination bus stop. There are many small bus stops that do not show on the map. This situation will be difficult to the new student if they on small bus stop but the system did not show the bus stop icon on the web page. The problems will be solved if there is bus stop icon on the web page so it can show the moving bus approach them.

2.2.2 Bus Tracking System by Northern Illinois University (NIU)

Figure 2-3   Screenshot for Route 7 in NIU bus tracking system

Figure 2-4   Screenshot for Route 11 in NIU bus tracking system

The Figure 2-3 and Figure 2-4 show the bus tracking system that has been developed by Northern Illinois University. This system is working without Google Map but they using jpg map that convert from Google Map. They are using this method because it easy to show different bus route on different map. As example in Figure 2-3 is showing the route 7 and Figure 2-4 is showing the route 11 for the bus route destination (Etaspot, 2012).

In both Figure 2-3 and Figure 2-4, there is a bus icon that represents the moving bus. The information will pop up when the user clicks the bus icon with the mouse cursor. It will show how many minutes the bus will arrive at the bus stop. The dot alongside the route in the map represents the bus stop. If there are buses in no service, the bus icon will not be display on the map.

The uniqueness for NIU bus tracking system is the web page is updating the bus position without refresh the web site. The system is using the exact time programming to update the bus position through the bus icon in the map without refresh the web page itself. Second uniqueness for this bus tracking system is the user can pick the route where they want to go. For example, if the user is on the route 11, they should pick "Route 11" destination to see position of the bus. This method can show the bus route and current bus position to the user clearly.

However, this webpage does not show the bus timetable or bus arrival time to the public. The webpage just shows the estimate time the bus will arrive to the nearest bus stop for example "less than 2 min". It is a problem if the user does not know the bus timetable. In addition, it does not show the bus capacity and bus not in service. It will help the student if the bus is full, then the student needs to go to other route to find the bus with low capacity. In the web page the bus icon that do not moving represent out of service. If there are new users, they do not know whether the bus is on service or not. Bus status need to be put into the system to inform the user about the bus services.

2.2.3 Bus tracking system by Rice University

Figure 2-5   Screenshot for Rice University bus tracking system – Route Inner Loop

Figure 2-6   Screenshot for Rice University bus tracking system – Route Greater Loop

The Figure 2-5 and Figure 2-6 show the bus tracking system that has been developed by Rice University. This system works with Google Maps embedded in web page with the simple interface that shows the bus services information (Ride-System, 2013).

The web page is showing the map that contains the campus area of the Rice University and the position of each bus. There are many colours of bus icon has been shown on the map. Each colour has different bus route and bus services. With this method, students easily differentiate bus route and bus services that differ in colour. When the user clicks the bus icon on the map, the bus route will appear with arrival time for the nearest bus stop. And if a user clicks the tab on the right corner of the web page, it will show the information of the bus route, each bus stop and bus services whether it is out of service or otherwise.

In addition, in the right corner also there are two tabs which are "active" and "all". The "active" tab will show the moving bus. And the "all" tab will display all the bus routes including buses in service and not. This page consists the updating speed on bus position on the map. From the user point of view, they will see the moving bus icon in the map because the webpage doesn't reload to update the bus position and the high update system. The users also can zoom in and out of the map to see clearly the route they take.

2.3   COMPARISON ON EXISTING SYSTEM

Based on the research above, there are several existing systems that have been developed. Table 2.1 shows the comparison of existing systems that are similar in terms of their features, advantages and disadvantages.

Table 2.1   Comparison of existing system

Functions	Bus Tracking System by APU	Bus Tracking System by NIU	Bus Tracking System by Rice University
GPS module	Yes	Yes	Yes
Technology has been used	GPS & GSM	GPS & GPRS	GPS & GPRS
Platform	Web page	Web page & Mobile phone	Web page & Mobile phone
To detect location	GPS	GPS	GPS
Advantages	Provide real time bus position.	Provide real time bus position. Give the expected time to arrive.	Provide real time bus position. Give the expected time to arrive.
Disadvantages	Did not provide the expected arrival time. The website needs to refresh each 5 seconds.	Did not provide the capacity of the bus. Not mentioning either bus in service or not.	Did not provide the capacity of the bus.

2.4   STUDIES ON TECHNOLOGY INVOLVED

Studies on technologies are needed to build a system. The technologies are Global Positioning System (GPS) and Hybrid App. The following is the detail studies on this technology.

2.4.1   Global Positioning System (GPS)

There are many kinds of tracking system that exist. In Real Time Bus Tracking System, GPS is used because it can show the exact location of the bus. Since, every point above the Earth will always be in radio contact with at least 4 satellites, and then it can provide accurate information and cover wide area(Leick, Rapoport, & Tatarnikov, 2015).

In this Real Time Bus Tracking System, GPS system will be installed in the bus so that it can send the location of the bus to the satellite. GPS receiver in the smart phone will be active by the module and satellite system will send the position data to the phone by the GPS receiver. In every 2 seconds, the data will send to the database.

In this system, the bus driver will install the application Bus Tracking System that already embedded with GPS service in it. The GPS will receive the current location of longitude and latitude of the bus and send it to database. And then the data will be receiving by customer by showing moving bus icon. Customer also need to install this Bus Tracking System in order to see the current bus location.

2.4.2   Hybrid Application

In this section will show the mobile application framework that used in Real Time Bus Tracking System. Hybrid app has been selected to connect between database and mobile application. It is because hybrid app is quicker to build, easy to learn and cheaper that native app (eVideoTuition, 2014). Table 2.2 will show the comparison between native app, mobile web app and hybrid app.

Table 2.2   Comparison between native app, mobile well app and hybrid app

Functions	Native App	Mobile Web App	Hybrid App
Development Language	Native only	Web only	Native or web
User Interface	Good	Average	Good
Code Portability	Low	High	High
Performance	High	Low	Medium
Development Cost	High	Low	Medium
Maintenance	High	Low	Medium

CHAPTER 3

METHODOLOGY

3.1   INTRODUCTION

In this chapter, which is methodology it will describe more about Software Development Life Cycle (SDLC) process and methodology itself to be implemented to create a Real Time Bus Tracking System project. In order to plan, structure and control the process of developing system efficiently, the SDLC method need to be choose wisely. There are many SDLC methods that exist such as agile software development, waterfall, spiral, Rapid Application Development (RAD) and other. Each of them has their own pros, cons and works effectively in appropriate situations.

In Real Time Bus Tracking System, Rapid Application Development (RAD) methodology has been chosen in order to develop this system because of the ability to quickly adjust to shifting requirements in a fast-paced and constantly changing. With the ability to quickly adjust, it will make low development cost.

3.2 RAPID APPLICATION DEVELOPMENT (RAD)

RAD is a solid development process that will be low in development cost and using minimal planning in favour of rapid prototypes. RAD is focusing on gathering requirements through focus groups or workshops re-use of the existing prototypes, early testing of prototypes by the customer and continuous integration (Point, 2017).

In Rapid Application Development, the functions or components can be developed in parallel as prototypes. The developments are delivered, time boxed and then assembled into a working prototype. As the result, it will deliver the complete product in faster way.

The advantages on using RAD are increases reusability of components, reduced development time, quick initial reviews occur and encourage customer feedback. It is used when there is a need to develop a system that can modularized in short time. It is perfect to use in this project since it takes short time to develop this project. Figure 3-1 show the phase involve during the process for this project.

Figure 3-1   Rapid Application Development (RAD) Model

Phase 1: Planning Requirement Phase

In this phase, the requirement and planning are the first phase of this RAD model. The objectives of this phase are to collect and analyse the overall requirements according to priority in order to reduce the development time. The meeting with supervisor has been set is to discuss the project name for this Final Year Project. During this phase, the information requirements have been discussed. In addition, the problem statement, scope and method that need to be used also have been discussed in order to develop this Real Time Bus Tracking System project.

Phase 2: System Design Phase

In this phase, the requirement from phase 1 is being analyse and specified into software, hardware and system architecture. During this phase also consist the entire necessary diagram that need to draw for the Real Time Bus Tracking System project. The diagram such as context, use case and dialog diagram.

Phase 3: Development Phase

In this phase, prototype will be made and implemented in real environment. The prototype will be test on selected user and the system is evaluated by them. While testing the prototype, observation and interview will be performing to verify the acceptance of the prototype by users. Observation will be performed to observe the functions and features provided are matching with the user's requirement. Interview need to be done if the users did not satisfy with the prototype. Then, the interview needs to be made to acquire the new requirements from the user. Then, the process will loop back to follow the current requirement and make the new prototype until the user satisfied with the provided requirement.

Phase 4: Cutover Phase

In this phase, cutover is the final task in the Software Development Life Cycle (SDLC) implementation method. This phase will include the testing, data conversion, user training and changeover to the new system that is involve in this project. Final system will be developed based on the latest prototype that already been accepted by user acceptance test. After the final system is developed, final testing will be performed and it is free from any error. Next, final system will be deployed and implement to the real environment. As a result, the new system has been built, delivered and been put into better operation.

3.2.1   Context Diagram

Figure 3-2 below shows the context diagram for this Real Time Bus Tracking System project. Context diagram shows that the system is under consideration as a single high-level process and shows the relationship that this system has with other entities.

Figure 3-2   Context Diagram for Real Time Bus Tracking System

Based on Figure 3-2, Real Time Bus Tracking System is used by two main user which are customer and bus driver that using smartphone. Customer need to click the "Find Bus" button to see the moving bus. Customer can see the information of the bus driver. Customer also can choose the direction of the bus that they want. Then, customer can make report if anything happens while using the application. Customer also can give feedback about the application. The GPS will receive the longitude and latitude coordinates from the satellite and send the data to database.

3.2.2 Use Case Diagram

Figure 3-3 below shows the use case diagram for this Real Time Bus Tracking System project. Use cases are the set of services, functions and actions that the system needs to perform.

Figure 3-3   Use Case Diagram for Real Time Bus Tracking System

Based on the Figure 3-3, Real Time Bus Tracking System's use case are interacting with two actors, which are customer and bus driver. There are five use cases which are Login, Register, View Map, Profile and Feedback. Customer and bus driver can interact with all use case.

3.2.3 Dialog Diagram

Figure 3-4 below shows the dialog diagram for this Real Time Bus Tracking System (RTBTS) project. Figure 3-4 shows the dialog diagram of the sequence of interface for the android application.

Figure 3-4   Dialog Diagram for Real Time Bus Tracking System

Based on the Figure 3-4, Real Time Bus Tracking System will have 1 main function and 3 sub functions in application. The main function is Home page to view the map. In this first function, customer can click the button to see bus location. The first sub function is logout, where users can logout from the system and go back to login page. Second sub function is Feedback, it is for customer to send the feedback about the services and bus driver can only see the feedback from them. Last sub function is Profile, where both users can view and edit their profile.

3.2.4   Software Requirement Specification (SRS)

The Software Requirement Specification (SRS) will precisely describe about the Real Time Bus Tracking System that will be built. It is to know all the requirements needed to develop this system. Then, this SRS will help in making of design in Software Design Document (SDD). The detail of SRS is stated in Appendix B.

3.2.5   Software Design Document (SDD)

The Software Design Document (SDD) will precisely describe about the design of Real Time Bus Tracking System. The detail of SDD is stated in Appendix C.

3.3   HARDWARE AND SOFTWARE REQUIREMENT

The requirements of this system are divided into two parts which are hardware and software requirements.

3.3.1   HARDWARE REQUIREMENT

Table 3.1 show the list of hardware requirement that will be used in order to develop this project.

<p align="center">Table 3.1   Hardware Requirement</p>

Hardware	Purpose
Laptop	To prepare the documents, make a configuration on develop the system.
Smartphone	To run the android apps.
GPS	To receive the longitude and latitude coordinate.

3.3.2   SOFTWARE REQUIREMENT

Table 3.2 show the software system requirement that will be used in order to develop this project.

Table 3.2   Software Requirement

Software	Purpose
Microsoft Office Word 2010	A tool to prepare the documentation.
Web Browser	To access the internet to find the related information.
Android Studio IDE	A tool to develop mobile application and generate apk file to install in smartphone.
Sublime Text	A tool to write the coding.
Firebase	To store the data into database.
Google Map API	Where all the API about map.
Lucid Chart Online	To modelling and designing the diagrams and flowchart.
Microsoft Project	A tool to prepare the Gantt Chart of the project.

3.4   GANTT CHART

These projects need to be done in two sessions which are PSM 1 and PSM 2. The detail of the project timeline has been done using Gantt chart as shown in Appendix A.

3.5   CONCLUSION

As conclusion, this chapter explain about methodology that has been selected and been used to develop this Real Time Bus Tracking System project. Rapid Application Development (RAD) methodology is the perfect method for this project. RAD is the best for this project because of faster development process, high quality results and makes lower costs to build a system (Powell-Morse, 2016).

In order to develop the system, the hardware and software requirement has been analysed. In addition, the suitable hardware and software has been chosen and listed in this chapter. The chosen hardware and software are important to ensure that all requirements meet the system requirement.

CHAPTER 4

RESULT AND DISCUSSION

4.1   INTRODUCTION

In this chapter, it will discuss the detail about the design, implementation and the testing of the project. The design and the implementation will be based on the requirement of the project. This project will be developed by using Android Studio IDE and Google Map Console. The database of this project is using Firebase Database which is embedded in android studio. The testing will be doing in several stages in order to make all the function working perfect as expected.

4.2   DESIGN REQUIREMENTS

The design has been discussed with client and supervisor through storyboard. Then the design will be develop using android studio by using XML language. The functionality of design will be using JAVA language. The following interface are the design of this Bus Tracking System application, which are Login page, registration page, main page, feedback page and profile page. Also, this project consists two user which are customer and bus driver.

First of all, when the users open the application, the user need to choose either customer or bus driver like in Figure 4-1. After that, it will go to login page (Figure 4-2 & Figure 4-3) which request user to fill in their email and password to go to main page. If user did not have account yet, there is sign up button to go to registration page.

Figure 4-1   Figure 4-2   Figure 4-3

In this registration page (Figure 4-4 & Figure 4-5), customer need to fill in their name, email and password. For bus driver, they need to fill in their name, email, password and bus destination. After that, it will go to main page which show the map.

Figure 4-4   Figure 4-5

In this main page (Figure 4-6 & Figure 4-7), customer will have one button and two choice of bus destination. These buttons are to put request to see the location of the bus. When the customer pull request to see bus location, bus driver main page will get the location of customer. Then, customer main page will get the location of bus driver and distance between them will be displayed in meter. There are three buttons in both page which are log out, feedback and profile button. Button logout will lead to login page.

Figure 4-6   Figure 4-7

In this feedback page (Figure 4-8 & Figure 4-9), customer can view the detail about the bus company such as email, number phone and company name. And also, customer can send the feedback about the bus's service by fill in the form given and click the button to send the feedback into the database. In bus driver page, it only shows the feedback given by the customer by showing customer id, email, name and feedback.

Figure 4-8   Figure 4-9

In this profile page (Figure 4-10 & Figure 4-11), both customer and bus driver can view and edit their own profile. For customer page, it shows profile picture, name, and password. But for bus driver, it shows the same as customer and one additional option which is bus destination. And the button to save the changes that has been made into database.

Figure 4-10   Figure 4-11

4.3   IMPLEMENTING REQUIREMENTS

The implementation in this system only required one stage which is software implementation. In this project, it is needed to install proper android platform to make the system run smoothly. It is better to install platform that provided package which has many functions in it that don't need to install anything else. Android platform that is using in order to develop this project is Android Studio which using Java program language. Next, it using Firebase database which is real time database to store all data about this application. It just needs to paste the JSON file in the Android Studio. And also, using Google Map Console to make the maps and all the function in map working.

Figure 4-12   Registration page in Android Studio Coding

Figure 4-13   Data in Firebase Database

Figure 4-14   Dashboard of Google Map Console

4.4   TESTING

After implementation it will end up with testing to make sure all the function is working smoothly according to the requirement provided and make sure the application is free error. This testing also to ensure that the data passing between database and system are in correct path. The software testing method that need to be used is black box testing or also known as behavioral testing. This testing is to ensure that the error can be found.

It is same as user acceptance test that the system is been test by user. After that, they will fill the acceptance test and to find out either the system is free error. The user acceptance test (UAT) that has been test by user will be referring to Appendix D.

4.5   USER MANUAL

Refer to Appendix E.

4.6   CONCLUSION

Real Time Bus Tracking System (RTBTS) is developed by using Android Studio, Firebase database and Google Map Console. It developed using 2 implementation which is mobile application, database and Google Map Console implementation. This application developed using Android Studio that include JAVA, JSON, XML and other languages. The database used in this project is Firebase Database which is real time database. It can store data and get data in real time. Google Map API key will make the all functionality in map working just fine.

CHAPTER 5

CONCLUSION

5.1   INTRODUCTION

In this chapter, it will conclude everything about this project which include all chapter and future work for improvement of this system.

Real Time Bus Tracking System (RTBTS) is been developed to ensure customer know the exact location of the bus is. RTBTS is implemented by using Android Studio that using Google Map API key to make the map interface working and using Firebase Database for the database. The location of the bus will be sends into database as bus driver open the application, and the customer can take the data from database to locate where the current bus location is by choosing the destination and click track button. The information that customer can get are bus location, distance of the bus and the detail of bus driver. It will give the pros to user to track the bus location in current time.

In conclusion, the goal of building RTBTS will overcome the problem statement and improve the features of the bus.

5.2   CONSTRAINT

There are the lists of the following constraint in this project:

I.   Internet

This application needs to have internet connection to make it working, since all the data is from database. And also, both users need to have internet connection to send their location to database.

II.   Platform

This application can be installed on Android OS only. This application cannot be installed in window OS.

III. Database

Bus driver need to open the application first to make sure the location data is been updated on database, so that customer can know the current bus location.

IV. GPS

Both users need to enable their GPS services on the phone to make them know the current location.

5.3 FUTURE WORK

Although the system has been developed, it still needs to have improvement in order to satisfied the user. In the following:

I. Multiple bus driver can login at the same time and customer can see many bus icons without choosing the destination.
II. Line of bus route will be display to make customer know which bus they need to take and customer can see where the bus going next.
III. Data of bus location is automatically sent without the driver need to open the application.

REFERENCES

Etaspot. (2012). niupublic.etaspot.net.   Retrieved from http://niupublic.etaspot.net/

eVideoTuition. (2014). Native Apps versus Web-Apps und mobile Webapplikationen. *Apps mit PhoneGap entwickeln*, 8. Retrieved from https://www.slideshare.net/videotuition/mobile-application-landscape-native-hybrid-mobile-web-apps-introduction-comparison

Leick, A., Rapoport, L., & Tatarnikov, D. (2015). *GPS satellite surveying*: John Wiley & Sons.

LOONG, C. C. (2013). BUS TRACKING SYSTEM.   Retrieved from http://eprints.utar.edu.my/1178/1/IA-2013-1100590.pdf

MyRapid. (2012). MyRapid Your Public Transport Portal.   Retrieved from https://www.myrapid.com.my/corporate-information/who-we-are

Point, T. (2017). SDLC RAD Model.   Retrieved from www.tutorialspoint.com

Powell-Morse, A. (2016). What Is Rapid Application Development (RAD) and How Do You Use It?   Retrieved from https://airbrake.io/blog/sdlc/rapid-application-development

Ride-System. (2013). Live Tracker.   Retrieved from http://www.rice.ridesystems.net/

APPENDIX A
GANTT CHART

	Task Name	Durat	Start	Finish
1	**Requirement Planning**	**20 days**	**Wed 2/21/18**	**Tue 3/20/18**
2	Identify problem and scope project	9 days	Wed 2/21/18	Mon 3/5/18
3	Identify objective and requirement	3 days	Tue 3/6/18	Thu 3/8/18
4	Reviewing existing system	4 days	Fri 3/9/18	Wed 3/14/18
5	Feasibility study	4 days	Thu 3/15/18	Tue 3/20/18
6	**System Design**	**29 days**	**Thu 3/22/18**	**Tue 5/1/18**
7	Plan design	10 days	Thu 3/22/18	Wed 4/4/18
8	Redefine requirement	5 days	Thu 4/5/18	Wed 4/11/18
9	Review requirement	5 days	Thu 4/12/18	Wed 4/18/18
10	Design Document (SRS and SDD)	9 days	Thu 4/19/18	Tue 5/1/18
11	**Development**	**50 days**	**Fri 9/14/18**	**Thu 11/22/18**
12	Software and hardware installation	3 days	Fri 9/14/18	Tue 9/18/18
13	Software,hardware and database implementation	31 days	Wed 9/19/18	Wed 10/31/18
14	Unit and integration testing	5 days	Thu 11/1/18	Wed 11/7/18
15	Debugging 1st time	5 days	Thu 11/8/18	Wed 11/14/18
16	System testing	1 day	Thu 11/15/18	Thu 11/15/18
17	Debugging 2nd time	5 days	Fri 11/16/18	Thu 11/22/18
18	**Cutover**	**12 days**	**Fri 11/23/18**	**Mon 12/10/18**
19	User acceptance	3 days	Fri 11/23/18	Tue 11/27/18
20	Evaluation	3 days	Wed 11/28/18	Fri 11/30/18
21	Documentation	4 days	Mon 12/3/18	Thu 12/6/18
22	Report submission	2 days	Fri 12/7/18	Mon 12/10/18

APPENDIX B
SOFTWARE REQUIREMENT SPECIFICATION (SRS)

Version

1

REAL TIME BUS TRACKING SYSTEM

Faculty Computer System & Software Engineering (FSKKP)

Software Requirement Specification (SRS)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 PRODUCT DESCRIPTION

1.1 USER CHARACTERISTICS

The Real Time Bus Tracking System application has two main users which are user and bus driver. Table 1.1 shows the education level, background of experience and technical expertise.

Table 1-1   User Characteristic

User	Education Level	Background Experience
User and Bus Driver	Any education level	Know how to use mobile phone.

1.2 CONSTRAINTS

Below are the lists of constraints for Real Time Bus Tracking System:
  i.   Only Android OS user can install this application.
 ii.   The users need to have smartphone.
iii.   Language provided in this application is only in English.
 iv.   To use this application, smartphone need to have connection with internet.

2.0 INTERFACES REQUIREMENTS

2.1 USER INTERFACE

Table 2-1   User Interface

User Interface Name	Descriptions
Main Page	This page contains two button which is customer and driver. User can click either they customer or driver.
Login Page	In this page, both users need to login their email and password in order to proceed to the home page. The user can click the signup button to register.
Signup Page	For those who doesn't have account yet, they need to fill in the form provided. For customer page, they need to fill in their name, email and password. And for bus driver, they need to fill in their name, email, password and their destination. After that, it will automatically go to the home page.
Homepage	It will show the customer the map that has bus icon on it when find bus button has been clicked and bus driver can see where the customer is. It is showing the current of the moving bus and estimated distance among them. For customer, there are two radio buttons for bus destination on the bottom of the application that user needs to chooses to see the bus location. In addition, there are three buttons which are logout, feedback and profile for both users on top of the application which they can click to go to another page. When user click logout button, it will go to main page where the user needs to login again.
Feedback Page	For customer page, it will show the detail about the company and if they want to complaint or give complement about the services, they can fill in the form which are name, email and comment. Click submit to go to homepage. For bus driver page, it will show all the feedback that they receive from customer about their services.

Profile Page	This page allow user to edit their profile. This page also can allow user to put their picture profile.

2.2 HARDWARE INTERFACE

Not Available

2.3 SOFTWARE INTERFACE

Table 2-2 show the software interface used in Real Time Bus Tracking System.

Table 2-2   Software Interface

Software Interface	Function
Firebase	This work as database to access and store data in here. The Global Positioning System (GPS) will send the data to be store in database.
Google Map Console	This work as console to locate the longitude and latitude of both user and make all the function on the map working.
Operating System	This work to make the application run smoothly in the smartphone.

3.0 SOFTWARE PRODUCT FEATURES

3.1   USE CASE – LOGIN

Figure 3-1 and Table 3-1 show the information of use case Login.

Figure 3-1   Use Case Login

Table 3-1   Use Case Description for Login

Use Case ID	RTBTS_UC_01
Brief Description	This use case is used by user to login to the application.
Actor	Customer and bus driver.
Pre-Conditions	1. The users manage to see the login page 2. The users already register to the system.
Basic Flow	1. The use case begins when the users open the application via smartphone. 2. User fill in the email and password to login[E1] [E2]. 3. User click either login or signup button. 4. Use case end.
Alternative Flow	None
Exception Flow	[E1]: Login error 1. Email or password wrong. 2. Message "email or password wrong" will be displayed. 3. Continue to step basic flow 3. [E2]: Blank form

	1. Some of the form is blank. 2. Message "Please fill in the form" will be displayed. 3. Continue to step basic flow 3.
Post-Conditions	Users will enter the application and can use it.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix A-1: Sequence Diagram for Login

3.2 USE CASE – REGISTER

Figure 3-2 and Table 3-2 show the information of use case Register.

Figure 3-2   Use Case Register

Table 3-2   Use Case Description for Register

Use Case ID	RTBTS_UC_02
Brief Description	This use case is used by user to register to use the application.
Actor	Customer and bus driver.
Pre-Conditions	1. The users have email account.
Basic Flow	1. The use case begins when the users open the application via smartphone. 2. User fill in the form given which are name, email, password and destination for driver[E1].

	3. User click Register button to enter to register account. 4. Use case end.
Alternative Flow	None
Exception Flow	[E1]: Blank form 1. Some of the form is blank. 2. Message "Please fill in the form" will be displayed. 3. Continue to step basic flow 3.
Post-Conditions	1. Account created. 2. Users will enter the application and can use it.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix A-1: Sequence Diagram for Register

3.3   USE CASE – VIEW MAP

Figure 3-3 and Table 3-3 show the information of use case View Map.

Figure 3-3   Use Case View Map

Table 3-3   Use Case Description for View Map

Use Case ID	RTBTS_UC_03
Brief Description	This use case is used by user to view map.
Actor	Customer and bus driver.

Pre-Conditions	1. Users logged in to the system.
Basic Flow	1. System display the map.
	2. Customer can see the moving bus on the map[A1].
	3. Bus driver can see where the customer is.
	4. System will display the distance between customer and bus driver.
	5. Use case end.
Alternative Flow	[A1]: Bus destination
	1. Users choose one of the radio buttons to see the different bus.
	2. Users is able to see different bus direction.
	3. Continue to basic flow step 2.
Exception Flow	None
Post-Conditions	1. System will display the distance and expected time.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix
	A-1: Sequence Diagram for View Map

3.4 USER CASE – PROFILE

Figure 3-4 and Table 3-4 show the information of use case Profile.

Figure 3-4   Use Case Profile

Table 3-4   Use Case Description for Profile

Use Case ID	RTBTS_UC_04
Brief Description	This use case is use by users to edit their profile.
Actor	Customer and bus driver.
Pre-Conditions	1. Users already logged in to the system.
Basic Flow	1. System will display the profile page.
	2. Users will be able to upload their profile picture and edit their profile.
	3. Bus driver able to change the destination bus.
	4. Message "Profile updated" will appear.
	5. Use case end.
Alternative Flow	None
Exception Flow	None
Post-Conditions	1. Profile will be updated.
	2. Bus destination changed.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix A-2: Sequence Diagram for Profile

3.5   USE CASE – FEEDBACK

Figure 3-5 and Table 3-5 show the information of use case Feedback.

Figure 3-5   Use Case Feedback

Table 3-5   Use Case Description for Feedback

Use Case ID	RTBTS_UC_05
Brief Description	This use case is used by customer to view the contact of client or give feedback about the services. This use case also is used by bus driver to view the feedback by customer about the services.
Actor	Customer and bus driver
Pre-Conditions	1. Users already logged in to the system.

Basic Flow	Customer:
	  1. Customer give feedback about the service[E1].
	  2. Message "Submitted" will appear.
	  3. Use case end.
	Bus driver:
	  1. Driver able to see the list of feedback about the services.
	  2. Use case end.

Alternative Flow	None

Exception Flow	[E1]: Blank form
	  1. The form is blank.
	  2. Message "Please fill in the form" will be displayed.
	  3. Continue to basic flow step 2.

Post-Conditions	1. The report is saved in database for bus driver review.
Rules	None
Constraints	None
Sequence Diagram	Refer Appendix A-3: Sequence Diagram for Feedback

4.0 REQUIREMENTS TRACEBILITY

Table 4-1 shows the requirement traceability that is available in each use case description of Real Time Bus Tracking System.

Table 4-1   Requirement Traceability

Requirement ID	Requirement Details
RTBTS_UC_01_01	Users able to login to the system.
RTBTS_UC_02_01	Users able to register to the system.
RTBTS_UC_03_01	Customer able to see the moving icon bus in real time.
RTBTS_UC_03_02	Customer able to see the information of the bus.
RTBTS_UC_03_03	Customer able to choose the bus destination.
RTBTS_UC_03_04	Customer able to see the chosen moving icon bus.
RTBTS_UC_04_01	Users able to update their profile information.
RTBTS_UC_04_02	Bus driver able to change the bus destination manually.
RTBTS_UC_05_01	Customer able to see the contact of client.
RTBTS_UC_05_02	Customer able to report about the services.
RTBTS_UC_05_03	Bus driver able to see the report about the services.

5.0 SYSTEM REQUIREMENTS APPROVAL

	Name	Date
Authenticated by: _____ Developer	MUHAMMAD ADDEIB BIN MOHD SUBARI	
Approved by: _____ Client	SAIMAH ADHA BINTI HJ SABRI	

APPENDIX

Sequence diagram for use case Login

Appendix A 1 Login

Sequence diagram for use case Register

Appendix A 2 Register

Sequence diagram for use case View Map

Appendix A 3   View Map

Sequence diagram for use case Profile

Appendix A 4   Profile

Sequence diagram for use case Feedback

Appendix A 5   Feedback

APPENDIX C
SOFTWARE DESIGN DOCUMENT (SDD)

Version

1

REAL TIME BUS TRACKING SYSTEM

Faculty Computer System & Software
Engineering (FSKKP)

Software Design
Document (SDD)

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

1.0   DATABASE DESIGN

Figure 1 show the Entity Relationship Diagram (ERD) for Real Time Bus Tracking System and the relationship between 4 table in the database.

Figure 1   Entity Relationship Diagram (ERD) for Real Time Bus Tracking System

1.1   DATA DICTIONARY

1.1.1 MainPage

Table 1 show the data dictionary for MainPage table in Real Time Bus Tracking System (RTBTS).

Field Name	Description	Data Types	Constraint
System_ID	System identification number.	Varchar (20)	PK
BDriver_ID	Bus driver identification number	Varchar (20)	FK

Customer_ID	Customer identification number.	Varchar (20)	FK
Report_ID	Report identification number.	Varchar (20)	FK
Distance	Distance between users	Varchar (20)	-

Table 1   MainPage Data Dictionary

1.1.2 BusDriver

Table 2 show the data dictionary for the BusDriver table in (RTBS).

Field Name	Description	Data Types	Constraint
BDriver_ID	Bus driver identification number.	Varchar (20)	PK
BEmail	Bus driver email.	Varchar (20)	-
BName	Bus driver name	Varchar (20)	-
BPassword	Bus driver password	Varchar (20)	-
BDestination	Bus driver destination	Varchar (20)	-
BLongitude	Bus driver longitude position	Varchar (20)	-
BLatitude	Bus driver latitude position	Varchar (20)	-

Table 2   BusDriver Data Dictionary

1.1.3 Customer

Table 3 show the data dictionary for the Customer table in (RTBS).

Field Name	Description	Data Types	Constraint
Customer_ID	Customer identification number.	Varchar (20)	PK
CEmail	Customer email.	Varchar (20)	-
CName	Customer name	Varchar (20)	-
CPassword	Customer password	Varchar (20)	-
CDestination	Customer destination	Varchar (20)	-
CLongitude	Customer longitude	Varchar (20)	-

Field Name	Description	Data Types	Constraint
	position		
CLatitude	Customer latitude position	Varchar (20)	-
Report_ID	Report identification number	Varchar (20)	FK

Table 3   Bus Route Data Dictionary

1.1.4 Report

Table 4 show the data dictionary for Report table in (RTBS).

Field Name	Description	Data Types	Constraint
Report_ID	Report identification number	Varchar (20)	PK
CEmail	Customer email.	Varchar (20)	-
CName	Customer name	Varchar (20)	-
Customer_ID	Customer identification number.	Varchar (20)	FK
Report_Description	Customer report.	Varchar (20)	-

Table 4   Report Data Dictionary

2.0   PRELIMINARY DESIGN

2.1   SYSTEM ARCHITECTURE

This topic identifies the internal organizational structure of this application. The relationship among application and sub-app will be described.

2.1.1   STATIC ORGANIZATION

Figure 2 show the static organization for Real Time Bus Tracking System.

Figure 2   Static Organization of Real Time Bus Tracking System

1. RTBTS_REGISTER

This package is responsible to manage register function. It consists of the following class:

   I.   CustomerRegisterClass
  II.   BusDriverRegisterClass

2. RTBTS_LOGIN

This package is responsible to manage login function. It consists of the following class:

I. CustomerLoginClass

II. BusDriverLoginClass

3. RTBTS_VIEW_MAP

This package is responsible to manage view map function. It consists of the following class:

I. CustomerViewMapClass

II. BusDriverViewMapClass

4. RTBTS_FEEDBACK

This package is responsible to manage view map function. It consists of the following class:

I. CustomerFeedbackClass

II. BusDriverFeedbackClass

5. RTBTS_PROFILE

This package is responsible to manage feedback function. It consists of the following class:

I. CustomerProfileClass

II. BusDriverProfileClass

3.0   DETAILED DESIGN

This section divided into the following paragraphs and subgraphs to describe the detailed design.

3.1   RTBTS REGISTER

3.1.1   CustomerRegisterClass

Class Type   : Entity class

Responsibility: This class is responsible to save customer information into database.

Attributes: Customer_ID : String

Name   : String

Email   : String

Password   : String

Method: voidRegister   : To save the new customer into database.

3.1.2   BusDriverRegisterClass

Class Type   : Entity class

Responsibility : This class is responsible to save the bus driver information into database.

Attributes: BDriver_ID   : String

Name   : String

Email   : String

Password   : String

Destination   : String

Method: voidRegister   : To save the new bus driver into database.

3.2 RTBTS LOGIN

3.1.1 CustomerLoginClass

Class Type   : Entity class

Responsibility: This class is to responsible to get the data from database.

Attributes: Customer_ID : String

                Email   : String

                Password   : String

Method: voidLogin   : To verify the mail and password from database

3.1.2 BusDriverLoginClass

Class Type   : Entity class

Responsibility : This class is responsible to get the data from database.

Attributes: BDriver_ID   : String

                Email   : String

                Password   : String

Method: voidLogin   : To verify the email and password from database.

3.3   RTBTS VIEW MAP

3.3.1   CustomerViewMapClass

Class Type   : Entity class

Responsibility: This class is responsible to show the location of the bus driver.

Attributes: Customer_ID : String

                Name   : String

                Longitude   : String

                Latitude   : String

Method: voidRequest   : To request location data of bus from database to
                display into the map.

3.3.2   BusDriverViewMapClass

Class Type   : Entity class

Responsibility : This class is responsible to show the location of the customer.

Attributes: BDriver_ID   : String

                Name   : String

                Longitude   : String

                Latitude   : String

Method: voidRequest   : To request location data of customer from database to
                display into the map.

3.4 RTBTS FEEDBACK

3.4.1 CustomerFeedbackClass

Class Type   : Entity class

Responsibility: This class is responsible to control and save the feedback into
database.

Attributes: Customer_ID : String

Name   : String

Email   : String

Feedback   : String

Method: voidFeedback : To save the feedback into database.

3.4.2 BusDriverFeedbackClass

Class Type   : Entity class

Responsibility: This class is responsible to view the feedback from database.

Attributes: BDriver_ID   : String

Name   : String

Email   : String

Feedback   : String

Method: voidFeedback : To view the feedback from database.

3.5   RTBTS PROFILE

3.5.1   CustomerProfileClass

Class Type   : Entity class

Responsibility: This class is responsible to control and save the customer profile into
database.

Attributes: Customer_ID : String

Name   : String

Email   : String

Profile_Image : String

Method: voidProfile   : To display the customer information and save the data
into database.

3.5.2   BusDriverProfileClass

Class Type   : Entity class

Responsibility: This class is responsible to control and save the bus driver profile into
database.

Attributes: BDriver_ID   : String

Name   : String

Email   : String

Destination   : String

Profile_Image : String

Method: voidProfile   : To display the bus driver information and save the data
into database.

4.0   SYSTEM DESIGN APPROVAL

	Name	Date
Authenticated by: ——————————— Developer	MUHAMMAD ADDEIB BIN MOHD SUBARI	
Approved by: ——————————— Client	SAIMAH ADHA BINTI HJ SABRI	

APPENDIX D
USER ACCEPTANCE TEST (UAT)

Version

1

REAL TIME BUS TRACKING SYSTEM

Faculty Computer System & Software
Engineering (FSKKP)

USER ACCEPTANCE
TEST (UAT)

TABLE OF CONTENT

1.0   USER ACCEPTANCE TEST (UAT)

1.1   RESULT - Customer

Event	Test Data	Expected Result	Pass/Fail
Login into Real Time Bus Tracking System using customer user	Valid email and password	Success login	Pass
Sign up for customer into Real Time Bus Tracking System	The information that will be used for log in validation	Success fill in the requirement needed	Pass
Current coordinate received accuracy	Obtain accurate coordinate from device	Accurate coordinate	Pass
Bus location icon coordinate display	Obtain accurate coordinate for bus	Bus icon display	Pass
Distance between bus and customer display	To calculate the distance between users	Distance is accurate	Pass
Give feedback about services	To give feedback about this application	Customer can send the feedback successfully	Pass
Update customer profile	Can update the profile	Successfully update the profile	Pass
Database	To test data store into database	All data saved into database	Pass

1.2   RESULT- Bus Driver

Event	Test Data	Expected Result	Pass/Fail
Login into Real Time Bus Tracking System using bus driver user	Valid email and password	Success login	Pass
Sign up for bus driver into Real Time Bus Tracking System	The information that will be used for log in validation	Success fill in the requirement needed	Pass
Current coordinate received accuracy	Obtain accurate coordinate from device	Accurate coordinate	Pass
Customer icon coordinate display	Obtain accurate coordinate for customer	Customer icon display	Pass
View the feedback from customer	To view the feedback given	Bus driver can view the feedback	Pass
Update the profile of driver	Can update the driver profile	Successfully update the profile	Pass
Can change the bus destination	Can update the bus destination	Successfully update the bus destination	Pass
Database	To test data store into database	All data saved into database	Pass

2.0 SYSTEM REQUIREMENTS APPROVAL

	Name	Date
Authenticated by: _____ Developer	MUHAMMAD ADDEIB BIN MOHD SUBARI	
Approved by: _____ Client	SAIMAH ADHA BINTI HJ SABRI	
Approved by: _____ Client	ABU BIN MUHAMMAD RAMLI BUS DRIVER OF RAPID BUS	

APPENDIX E
USER MANUAL

REAL TIME BUS TRACKING SYSTEM

Faculty Computer System & Software
Engineering (FSKKP)

User Manual

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

1.0   GENERAL INFORMATION

1.1   System Overview

This project is to specify the requirement for Real Time Bus Tracking System. This system is a mobile application based. The user of this application are customer of the bus and the bus driver. The registered bus for this project is only rapid bus number 100.

1.2   Organizational of the manuals

This user manual is divided into three parts which are general information, system summary and the last part is user guidelines.

2.0   SYSTEM SUMMARY

2.1   System configuration

Real Time Bus Tracking System is the mobile application based. It requires internet connection and enable the GPS function in order to run the application.

2.2   User Access Level

Table 2.1 shows the user access level for Real Time Bus Tracking System.

Table 2.1   User Access Level

User Access Level On	Customer	Bus Driver
Login	/	/
Register	/	/
Home – View Map	/	/
Profile	/	/
Give Feedback	/	
View Feedback		/

3.0   USER GUIDELINES

3.1   User Interfaces

3.1.1   Splash and Main Page Interface

Figure 3.1 and Figure 3.2 show the splash screen and main page every time open this application.

Figure 3.1   Splash Screen

For Customer user

For Bus Driver User

Figure 3.2   Main Page Interface

3.1.2 Login and Register – Customer

Figure 3.3 shows the login interface that customer needs to fill to enter the home page. If the customer doesn't register yet, they need to fill in their email, name and password to register as shown in Figure 3.4.

Fill in customer email

Fill in email password

Login button

Register button

Figure 3.3   Customer Login Interface

Figure 3.4   Customer Register Interface

3.1.3 Home Interface – Customer

Figure 3.5 shows the Home page for customer. It shows the map with the customer current location. There is radio button for select the route and "Find Bus" button to find the bus. Figure 3.6 shows that real time bus location and bus driver information will appear when the customer clicks the "Find Bus" button. Figure 3.7 show the side menu in Home interface that will lead to profile, about us which is feedback and logout.

Figure 3.5   Customer Home Interface

Figure 3.6   Customer Home Interface – Bus Location

Figure 3.7   Customer Home Interface – Side Menu

3.1.4 Profile Interface – Customer

Figure 3.8 shows the profile interface. Customer can upload their picture, change name and put number phone.

Figure 3.8   Customer Profile Interface

3.1.5 Feedback Interface – Customer

Figure 3.9 shows the feedback interface for customer. In this page, customer can view the name, email and number phone of the company. Customer also can give feedback about the services.

Figure 3.9   Customer Feedback Interface

3.1.6   Login and Register – Bus Driver

Figure 3.10 shows the login interface that driver needs to fill to enter the home page. If the driver doesn't register yet, they need to fill in their email, name, password and bus destination to register as shown in Figure 3.11.

Figure 3.10   Bus Driver Login Interface

Figure 3.11   Bus Driver Register Interface

3.1.7   Home Interface - Bus Driver

Figure 3.12 shows the Home page for bus driver. It shows the map with the driver current location. Figure 3.13 shows that customer location with customer information will appear whenever GPS receive the request from the customer. Figure 3.14 show the side menu in home interface that will lead to profile, about us which is view feedback and logout.

Figure 3.12   Bus Driver Home Interface

Figure 3.13   Bus Driver Home Interface – Customer Location

Figure 3.14   Bus Driver Home Interface – Side Menu

3.1.8 Profile Interface - Bus Driver

Figure 3.15 shows the profile interface. Bus driver can upload their picture, change name, put number phone and change the bus destination.

Figure 3.15   Bus Drive Profile Interface

3.1.9   Feedback Interface - Bus Driver

Figure 3.16 shows the feedback interface for bus driver. In this page, driver can view all the feedback that customer give.

Figure 3.16   Bus Driver Feedback Interface