Daily activity application linked with public transportation system

Zeiyad Fathi Sfaxsi, Mahmood Sarzil, Karishma Patel A/P Balvantry, Fariza Aqila Bt. zulkifly

SESSION 2016

FACULTY OF COMPUTING AND INFORMATICS

MULTIMEDIA UNIVERSITY

MAY 2016

Daily activity application linked with public transportation system

BY

Zeiyad Fathi Sfaxsi, Mahmood Sarzil, Karishma Patel A/P Balvantry, Fariza Aqila Bt. zulkifly

SESSION 2016

The copyright of this thesis belongs to the author under the terms of the Copyright Act 1987 as qualified by Regulation 4(1) of the Multimedia University Intellectual Property Regulations. Due acknowledgment shall always be made of the use of any material contained in, or derived from, this thesis.

Declaration

I hereby declare that the work in this thesis have been done by myself and no portion of the work contained in this thesis has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

_____

The Killers
Faculty of Computing & Informatics
Multimedia University
Date: 17th May 2016

iii

Acknowledgements

We would like to express our deepest appreciation to all those who provided us the possibility to complete this report. A special gratitude we give to our lecturer, Mr. Wan Ruslan Yusoff, whose contribution in stimulating suggestions and encouragement, helped us to coordinate our project especially in learning and practicing software engineering principles and activities.

Furthermore we would also like to acknowledge with much appreciation the senior students, who helped us by providing us with their experience.

A special thanks goes to our friends, who helped us by assembling the parts and giving suggestions about the task.

Many thanks go to our parents who are the reason of us being able to study in MMU. Moreover, they have invested their full effort in guiding us in achieving our goals.

Last but not least, we have to appreciate the guidance given by other lecturers as well as the workers especially in the faculty of computing and informatics.

Contents

List of Figures

List of Tables

Listings

Abbreviations and Acronyms

FCI	Faculty of Computing and Informatics
MMU	Multimedia University
MTS	My Transportation Schedule (Software name)
Q.A.	Quality Assurance

Management Summary

Procedural design is considered the low level algorithmic design of software codes or computer programs. It covers the design of procedures, that is, the inner details of the steps and instructions the program are to execute to achieve their intended objectives.

Procedural design activities can also be interpreted as the design activities at the level of software components. The software constructor will use these procedural designs and write software codes to implement and execute those designs.

Chapter 1

Software Project Planning

1.1 Introduction

1.1.1 Project Brief Description

MTS (My Transportation Schedule) is a daily activity application that could be installed in any Android (An operating system works in mobiles and tablets) device. It focuses on the people who are using the public transportation system in Cyberjaya.

The buses among cyberjaya will be provided by a GPS tracker that would track the bus and update the database. The user will be able to organize his timetable through MTS.

The user is only required to insert the time, the date and location in Cyberjaya of the event, MTS garanties that the user will not be late on any event, by choosing the best route considering the density as well as the traffic. MTS is

designed to be able to alarm the user if the target destination cannot be reached on time.

1.1.2   MTS is an IoT Software

Internet of Things (IoT) is a connection of physical objects such as devices, buildings, vehicles, and items embedded through electronics, sensors, softwares, and network connectivity that are allowing all these items to get and exchange data [1].

A GPS device is considered as a sensor for tracking the object's location which is one of the main necessity for an IoT application [2].

Our software uses GPS trackers that are installed to the buses that allows the exchange of data. The software, connects through the GPS with the use of a sim card that allows the GPS to connect to the server that transfers data [3].

1.1.3   Project Objectives

- Minimizing the chance of being late.

- Avoid wasting time waiting in the bus station for the bus to come.

- Build a software with a good user experience (UX).

1.2 Project Scope

MTS covers only buses in Cyberjaya. For further prototypes, MTS might be extended to have a wider coverage and more public transportation methods such as trains and Taxis.

For starting prototypes, MTS has only four engineers working on it. Which leads to make the starting prototypes used by only few hundreds users to ease the proccess of finding and fixing bugs.

For the first prototypes, the application would be bassed on Android operation system.

1.3 Installation

MTS needs two servers that are the main server and the backup server to be installed. The backup server will be a copy of the main server. The backup server will be in service automatically if the main server stopped working. Figure 1.1 shows the relationship between servers and their databases.

The Internet

Backup Server Backup Database

Main Server Main Database
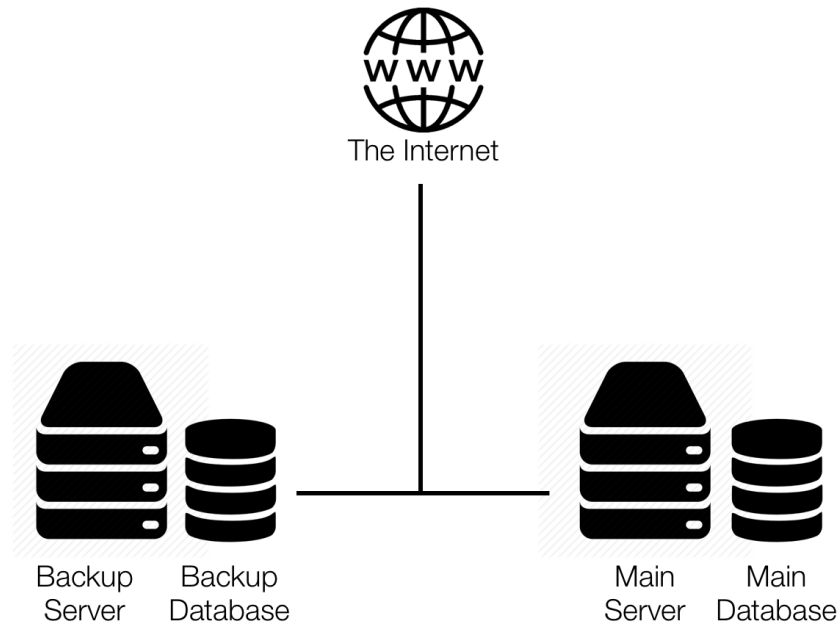
FIGURE 1.1: Gantt Chart: Estimated Timeline for a prototype

A chip equipped with a GPS tracker and a SIM card will be installed to every public transportation bus in cyberjaya. The GPS tracker will manage the location tracking function which gets every bus's location from the GPS Satellite and update it in the database, while the sim card will be providing the chip with the Internet connection.

1.4 Execution

The first two prototypes will be available only for the developers; For the sake of developing and improving. MTS will be available in Google Play after the release of the third prototype. For the ease of testing the prototype, 200 users

will be capable of using MTS by then. However, the software will be available for all cyberjaya users after the delivery of the fifth prototype.

During execution, feedbacks will be noted from stakeholders in order to list the following prototype's requirements.

1.5 Feedbacks and updates

At the first two prototypes the developers will list their feedbacks. By the third prototype the users will be able to use and give their feedback about MTS. A feedback is considered as soon as it's received and the update will be held to the next prototype to be out.

Chapter 2

Estimated Budget

2.1   Hardware itmes cost

Below is a table of items needed and it's prices.

Items needed for the project				
Item	**Description**	**Alternative**	**Purpose of choose**	**Price**
CPU	Intel Xeon	AMD Athlon	Cost	389.39$
RAM	Kingston 16 GB	Corsair 16 GB	Quality	76.39$
HDD	Seagat 1 TB	Kingsnot 1 TB	Cost	49.99$
GPS tracker	Hossenchip	HCT Micro	Cost	16.99$
SSL certificate	PositiveSSL from SSLs.com	godaddy	Cost	4.99$
Others	Keyboard, Mouse, screen . . . etc	-	-	290$
Total				727.75$

TABLE 2.1: Items needed for the project

Because we are going to use two servers - main and backup servers - the cost would be **1421.52$**

Cyberjaya has aproxmatily thirty-six buses price of GPS tracker multiplayed with thirty-six has to be added. The total price of GPS tracker is **611.64$**.

Total cost is **2033.16$**.

Chapter 3

Estimated Timeline

3.1 Estimated Timeline

Because MTS will be developed by using prototyping model, It is estimated that the thired prototype would be stable enough to be published in the Google Play - The market of android applications -.

The second and first prototype are to find bugs and security issues in earlier time which reduce the risk and minimize the cost of fixing.

Each prototype needs at least twenty-one weeks to be fully deployed. Furthermore, the work of a following prototype would be established on the nineteenth week of previous prototype.

3.1.1 Gantt Chart

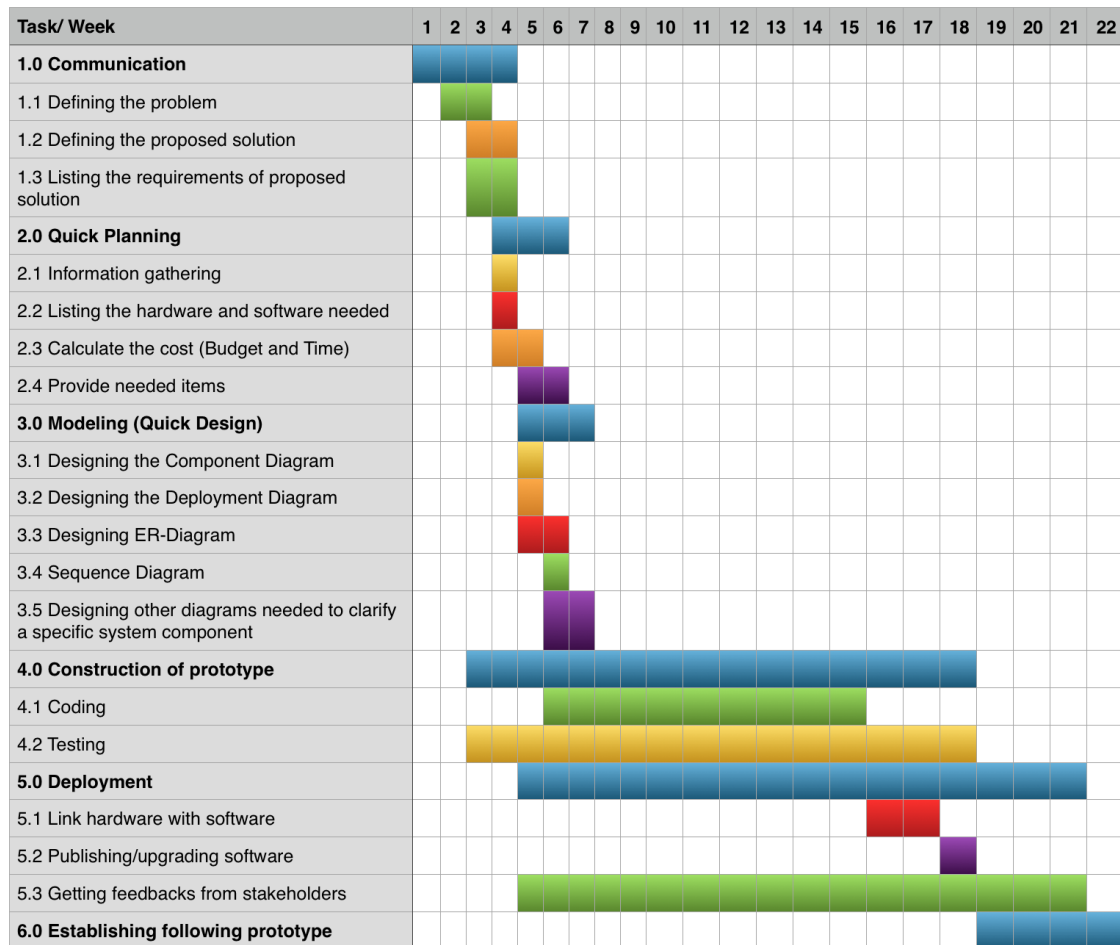Below the Gantt Chart discripes estimated timeline for each prototype.

Task/ Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1.0 Communication																						
1.1 Defining the problem																						
1.2 Defining the proposed solution																						
1.3 Listing the requirements of proposed solution																						
2.0 Quick Planning																						
2.1 Information gathering																						
2.2 Listing the hardware and software needed																						
2.3 Calculate the cost (Budget and Time)																						
2.4 Provide needed items																						
3.0 Modeling (Quick Design)																						
3.1 Designing the Component Diagram																						
3.2 Designing the Deployment Diagram																						
3.3 Designing ER-Diagram																						
3.4 Sequence Diagram																						
3.5 Designing other diagrams needed to clarify a specific system component																						
4.0 Construction of prototype																						
4.1 Coding																						
4.2 Testing																						
5.0 Deployment																						
5.1 Link hardware with software																						
5.2 Publishing/upgrading software																						
5.3 Getting feedbacks from stakeholders																						
6.0 Establishing following prototype																						

FIGURE 3.1: Gantt Chart: Estimated Timeline for a prototype

Chapter 4

Project Organization and Structure

4.1 Manpower

This project is handled by four Software engineering students. That leads everyone to handle more than one position which produce a higher quality product after each prototype.

4.2 Project Organization and Structure

Team members are having a role rotation plan that designed to enhance the product by team members abilities. Below is the organization chart shows basic roles:
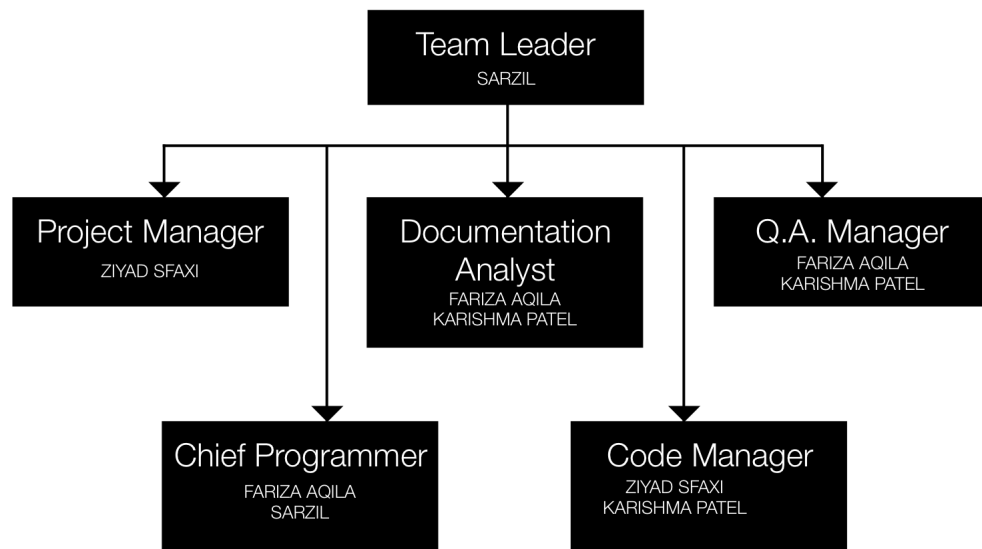

FIGURE 4.1: Organization Chart

Below is a table that descripe every role:

ORGANIZATION	DESCRIPTION
Team Leader	 • Serve as main contact with the managers. • Ensuring the team is consistently delivering working software to the standards the department expects. • Ensuring the team is self-organising so that we take collective responsibility for the work we do.
Project Manager	 • Rotate roles to maximize the result of team abilities. • Coordinate all those resources and the team's tasks – to make sure that work is done in the proper sequence with a minimum of time. • Ensuring the scope project is delivered and the body of work is accomplished.

Q.A. Manager	• Assures the viability, functionality and effectioness of essential tools. • Anticipates program release problems and takes coorective action, escalation as needed, to resolve and achieve commitments.
Documentation Analyst	• Ensure that all documents have no errors in filenames or submissions. • Perform evaluations and document audits.
Chief Programmer	Convert the design and to a programming langauge.
Code Manager	Maintain the record of code files and the make file(s)

TABLE 4.1: Table of organization roles

Chapter 5

Software Process Model

5.1 Prototyping model

5.1.1 Description

Prototyping model has five main phases which are listed below:

Communication: Gathering and discusing the requirements from and with
   users.

Quick Planning: Establishes a plan for software engneering work, addresses
   technical tasks, resources, work products, and work schedule.

Modeling (Quick Desgin): A quick plan that brings together customer
   requirements, business needs and technical considerations.

Construction of prototype: Combines code generation and testing uncover
errors.

Deployment, Delivery, and Feedback: Involves delivery of software to the
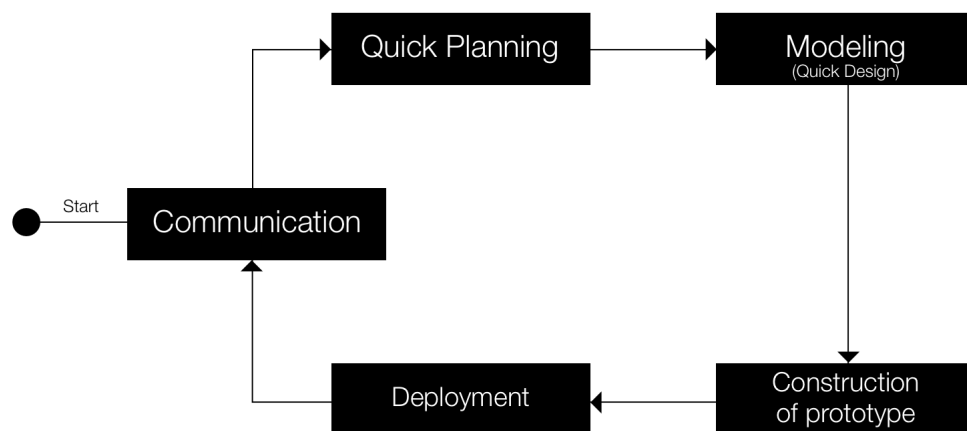customers (Users) evalution and feedback.

FIGURE 5.1: Prototype Model Diagram

5.1.2 Reasons for choosing prototyping model

By end of 2016 the population in Cyberjaya will be about 100,00, and most of people in Cyberjaya uses buses as their cheap and easy transportation method [4]. That makes MTS highly required and might be used by most of people who live in Cyberjaya. For that reason, bugs would be found easily and bugs would need a quick fix to reduce the damages and failures.

An advantage of Prototyping model is that the quicker user feedback is available which leads to better solutions and fast error fixing. In other words, users are actively involved in the development.

MTS is a system that needs to have interactions with the end users, therefore, Prototype model should be used. Moreover, MTS is an online system which has direct interfaces with many users at the same time. MTS might has a high amount of interaction with the end-users so prototype model is best compared to other models.

Chapter 6

Software Project Design

6.1 Use-Case of project

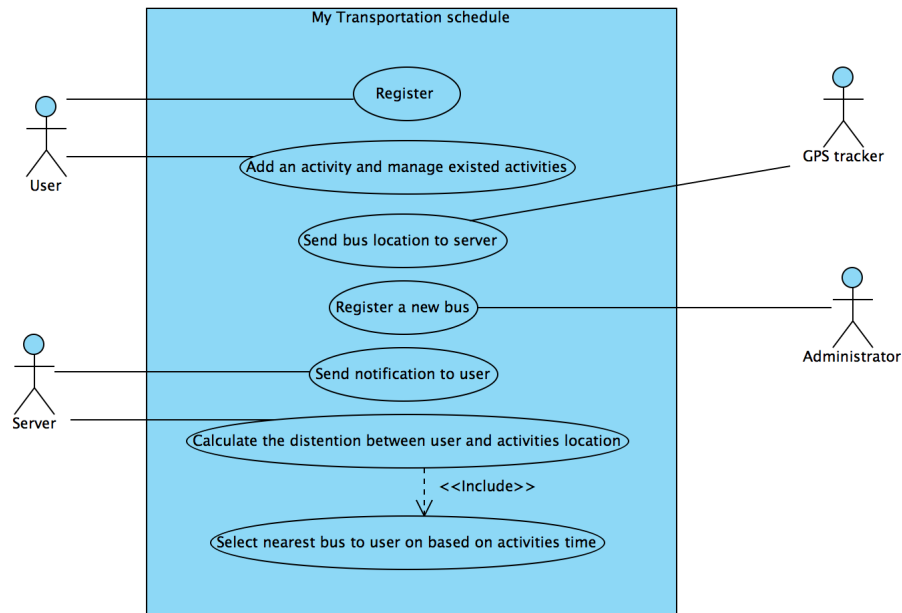Below the use-case daigram shows the use-cases of the project.

My Transportation schedule

Register

Add an activity and manage existed activities

Send bus location to server

Register a new bus

Send notification to user

Calculate the distention between user and activities location

<<Include>>

Select nearest bus to user on based on activities time

User

Server

GPS tracker

Administrator

FIGURE 6.1: Use-cases of the project

6.2 Component Diagram

Below is the Component Diagram shows the main software component and the interfaces between them.
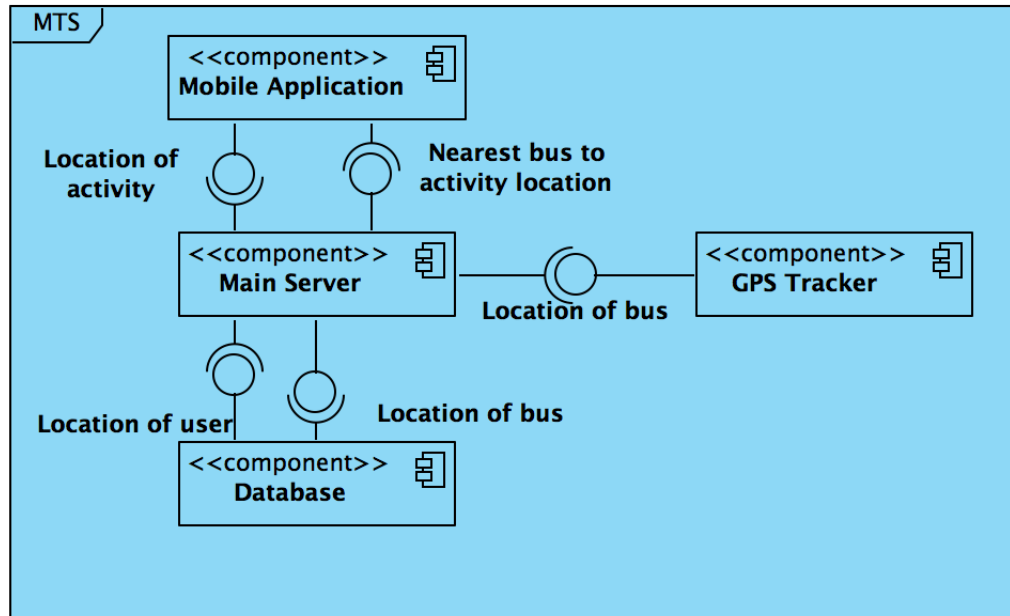
FIGURE 6.2: Component Diagram

6.3 Sequence Diagram

Below is the Sequence Diagram shows the attraction between user and the software, and how the software works overall.
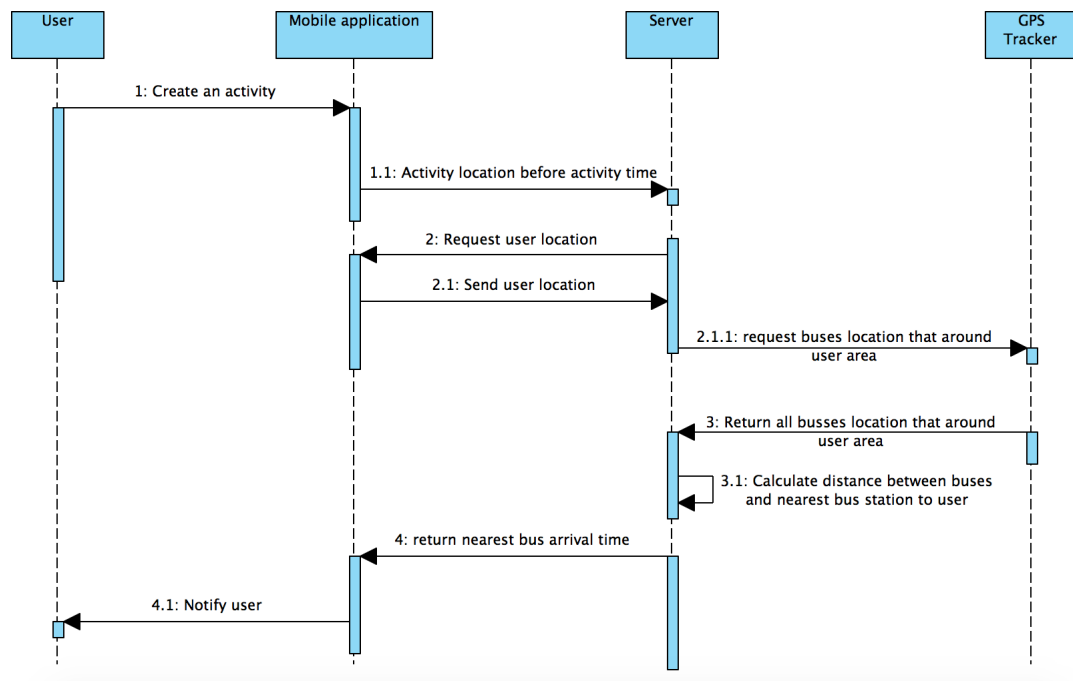

FIGURE 6.3: Sequence Diagram

Chapter 7

Conclusions

7.1 Conclusions

MTS was made for those who are facing troubles catching buses on time,
missing the bus will lead you to wait for the next bus to come.
Moreover, MTS will help user to be early on events they set using the
application. Our developer team is working hard to deliver the a high quality
product using the skills that we earned during our studies about software
engineering.

7.2 Recommendations for Future Work

- MTS domain will be extended to cover kuala lumpur area. MTS is
  currently limited to cybejaya users, extending the domain will gain the

software more users.

- MTS will include taxis as well as trains for full user experience. By including taxis and trains MTS will be more

7.3 Project Repository

https://github.com/zizo95/MTSN

Bibliography

[1] Wikipedia. *Internet of Things (IoT)*, May 11th 2016. URL
https://en.wikipedia.org/wiki/Internet_of_Things. Retrived on
12/05/2016.

[2] Paul Rubens. *Internet of Things (IoT)*, May 11th 2016. URL
http://www.cio.com/article/2843814/developer/
how-to-develop-applications-for-the-internet-of-things.html.
Retrived on 12/05/2016.

[3] Texas Instruments. *Challenges in the Internet of Things*, May 11th 2016.
URL
http://www.ti.com/ww/en/internet_of_things/iot-challenges.html.
Retrived on 12/05/2016.

[4] Wikipedia. *Cyberjaya*, May 16th 2016. URL
https://en.wikipedia.org/wiki/Cyberjaya. Retrived on 16/05/2016.