UCSC

Masters (MIT) Project Final Report

March 2018

Please tick relevant Stream	
MIT(General) : MIT3101	√
MIT(e-Learning) : MIT3111	
MIT(Multimedia) : MIT3121	

For MIT(General) Projects only	
Research Project	
Implementation Project	√

Project Title	easyBus – Online GPS Based Bus Tracking & Management System
Student Name	Nipuna Niroshana Kiriella
Registration No.	2015/MIT/024
Index No.	15550243
Supervisor's Name	Dr. K L Jayaratne

For Office Use ONLY

easyBus – Online GPS Based Bus Tracking & Management System

A dissertation submitted for the Degree of Master of Information Technology

N. N. Kiriella
University of Colombo School of Computing
2018

Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name:	Nipuna Niroshana Kiriella
Registration Number:	2015/MIT/024
Index Number:	15550243

Signature: Date:

This is to certify that this thesis is based on the work of

Mr./Ms.

under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name:

Signature: Date:

Abstract

The Public Bus Service in Sri Lanka has been cumbersome for the passengers due to practical problems such as the delay of busses and waste of time, inability to identify the exact point to get down from or get in to the bus, and the inability to decide the exact bus they need to get. The root cause of these problems in general, is the inability to track the buses matching their requirements in real-time to which presently no solution is provided. This project was carried out to introduce a Global Positioning System (GPS) based interfaces to track the busses matching the requirements of the passengers in real-time called 'easyBus', as a solution.

The 'easyBus' system detects the bus halts around a bus in its route using the current latitudes and longitude of the bus provided by the Android smartphone of the bus driver, and sends the current location of the bus to the relevant bus halt, and also sends to the bus the halts currently around it, in the relevant route. In addition, the system facilitates two search options to search online the closest bus which goes to the intended destination and the time schedules of the expected busses.

The suggested system was tested for the buses going in one direction in one route and it proved that it can be extended to multiple routes with both directions. This system reduces the time wastage of the passengers and increase their knowledge about the bus routes and the departure/arrival times of the buses.

.

Acknowledgement

I would like to express my deep gratitude to all my lecturers of University of Colombo School of Computing, for their patient guidance, enthusiastic encouragement and useful critiques of this project work. I would also like to thank Dr. K L Jayaratne, for his advice and assistance in keeping my progress on schedule as my project supervisor. My grateful thanks are also extended to my project co-ordinator Dr. (Mrs.) F H A M Premachandra for her guildelines for success of the project.

I would especially like to thank my best friends who helped me in finalizing this project within the limited time frame.

Nobody has been more important to me in the pursuit of this project than the members of my family. I would like to thank my parents, whose love and guidance are with me in whatever I pursue. They are the ultimate role models. Most importantly, I wish to thank my supportive sister.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

- Admin - Administrator
- API - Application Programming Interface
- App - Application
- DOB - Date of Birth
- EBTC - easyBus Test Case
- ER – Entity Relationship
- GPRS - General Packet Radio Service
- GPS - Global Positioning System
- GSM - Global System for Mobile Communications
- ID - Identity
- LCD - Liquid Crystal Display
- NIC - National Identity Card
- No - Number
- NTC - National Tranport Commission
- PW - Password
- Reg - Registration
- SLTB - Sri Lanka Transport Board
- SLTB - Sri Lanka Transport Board
- SMS - Short Message Service
- UN - User Name

Chapter 1: Introduction

This chapter provides the overview of the project and discuss the motivation, aims and objectives, scope and structure of the dissertation.

1.1 Motivation

In Sri Lanka there is no computerized system to track and manage the public bus service according to passengers' requirements. Despite of the development of the infrastructure facilities in Sri Lanka, this has resulted in Public Bus Service being cumbersome for the passengers due to practical problems such as the delay of busses and waste of time, inability to identify the exact point to get down from or get in to the bus, and the inability to decide the exact bus they need to get.

Therefore, it is utmost important to find a solution to overcome these practical problems to make the Public Bus service more comfortable, passenger-friendly and effective especially by increasing the knowledge of the passengers about the bus routes and of any means of tracking the bus they want to get and the time schedules on departures and arrival of the busses, because the mere development in infrastructure facilities would be obsolete if the passengers still find the Public Bus Service cumbersome and time consuming.

This system facilitates the users to track the buses they want online through GPS technology and thereby to overcome the above mentioned issues.

1.2 Aims and Objectives

The main objective of this project is to identify the locations of the buses according to the requirements of the passengers. To achieve this main objective, following sub objectives have to be achieved.

- To up-lift the living standards of the passengers.
- To facilitate online tracking of the buses.
- To facilitate to identify the point at which to get down from the bus, accurately.

- To prevent time-wasting due to the current situation of the buses such as traffic blocks, break downs, etc.
- To facilitate to identify accurately the bus which the passenger should get to reach the intended destination.
- To prevent time-wasting due to the lack of knowledge on the arrivals/departures of the buses.

1.3 Scope

The project intends to provide a user friendly and web based GPS tracking management system which mainly targets the passengers of the public bus service. A mobile application will also be developed to get the latitude and longitude of the buses. Based on the current latitude and longitude of a bus, the server application will process the data for the requirements of the users of the system.

Considering protection and accuracy of the system, following general functions are to be implemented for the administrators of the system.

- Administration login, monitoring and controlling of the system.

This project focuses on implementing the following main four functions for the passengers to overcome the problems in public bus service.

- Inside of the bus display the previous, current and next bus halts of the bus.
- Display at the bus halt the bus routes and the next bus halt locations of the arriving buses.
- Online search for the next bus to the intended destination according to the location of the bus halt at which the passenger is.
- Online search for the arrival/departure times of all the buses going to/through the intended destination.

This project is limited to one direction in one bus route due to the time constraints.

- Selected bus route number is 69 and the direction is Ratnapura → Kegalle for the easyBus system. (this can be expanded for other routes and other directions in future)

1.4 Structure of the Dissertation

Chapter 1 – Introduction

Introduces the background, problem statement, significance of the study, objectives, scope and limitation of the study and presents the framework to give readers a basic idea of this project.

Chapter 2 – Background

Literature review of similar systems available and technological review relevant to improve the system and their pros and cons will also be included.

Chapter 3 – Analysis and Design

Requirement gathering, analyzing and management techniques are discussed. Functional and non-functional requirements of the proposed system will also be included.

Initial design diagrams and the methodological approach used in the system designing, alternative approached and their justification for not using in the system are discussed.

Chapter 4 – Implementation

Appropriate coding and implementation tools and techniques which will be used developing the system are included.

Chapter 5 – User Evaluation and Testing

Developed system will be tested against the user requirements which are gathered at the beginning. Actual data from the client's environment will be used.

Chapter 6 – Conclusion and Future Work

This chapter discussed about the system enhancements which can be done in the future and also provide a summary note of the system achievements at last.

Chapter 2: Background

This chapter presents a concise summary of the background information to the implementation and discuss about existing similar systems.

2.1 Background Description

Presently, the Sri Lanka Transport Board [1] does not have a GPS based monitoring system to monitor SLTB buses. However, the National Transport Commission [2] uses a GPS monitoring system to monitor few private buses. Even though they use a very expensive GPS tracking device to track the private buses, they do not have any facility for passengers to track the buses according to the passenger requirements. Considering this situation, it is important to implement a new system to track any SLTB or private buses according to the requirements of the passengers under a low cost GPS tracker.

2.2 Existing Similar Systems

Real-time tracking and management of vehicles has been a field of interest for many researchers and developers and a lot of research work has been done for tracking systems. Recently the various anti-theft modules like steering wheel locked equipment, network tracking system and traditional electronic alarm are developed along with client identification and real time performance monitoring.

Vehicle tracking systems are commonly used by fleet operators for fleet management functions such as fleet tracking, routing, dispatch, on-board information and security. Along with commercial fleet operators, urban transit agencies use the technology for a number of purposes, including monitoring schedule adherence of buses in service, triggering changes of buses' destination sign displays at the end of the line (or other set location along a bus route), and triggering pre-recorded announcements for passengers. Further when these solutions are coupled with various sensors they can be used to monitor the temperature, fuel consumption, door closers etc.

In Sri Lanka, there are some vehicle tracking and monitoring systems available to buy with various features. Mainly telecommunication providers such as Mobitel and Dialog are in this business. Mobitel has two solutions [3] called 'mLocator' and 'mTrack' to track vehicles. And also Dialog has 'iLocate' [4] as a solution. Some web and software developing companies are also in this business. As an example of that Nexify Systems (Pvt) Ltd [5] has multiple solutions called 'Track My Car' [6] to track various types such as vehicle, people, pets, etc. These kind of tracking and monitoring systems share similar features most of the time.

There are some popular vehicle tracking and reservation systems are using in public transportation services in Sri Lanka. Nowadays, PickMe [7] is the most popular vehicle tracking and reservation system among Sri Lankan Passengers to booking taxis. Before Pickme, Kangaroo Cabs [8] was the most popular online taxi booking system. And also Uber [9] is a world famous vehicle tracking and reservation system also available in Sri Lanka to taxi booking. eTours Lanka (Pvt) Ltd has a bus tracking and reservation system [10] for passengers of public bus service. Those kind of tracking and reservation system also share smiler features.

Therefore, under this section discuss about Mobitel 'mTrack' solution and PickMe App to cover both monitoring and reservation of the vehicle tracking system exists in Sri Lanka.

2.2.1 mTrack Overview

The Mobitel mTrack solution provides an up-to-the-minute picture of vehicle fleets, whether you are at your desk, on your tablet computer, or using your smartphone. The solution has the capability to provide the location of your fleet vehicles in real time, 24 hours a day from anywhere, provide the historical status of each vehicle, including when it started up and shut down, as well as its idling status, location, speed, and more all via a user-friendly interface which assists companies in the effective management of their vehicle fleets.

The Process

mTrack is a GPS based solution for Vehicle tracking. This corporate solution consists of a base unit which carries a Mobitel SIM. Once installed in Vehicles, this unit shall send the location info via GPRS/SMS to a base system which in return shall interpret the info to a MAP or a log for user consumption.

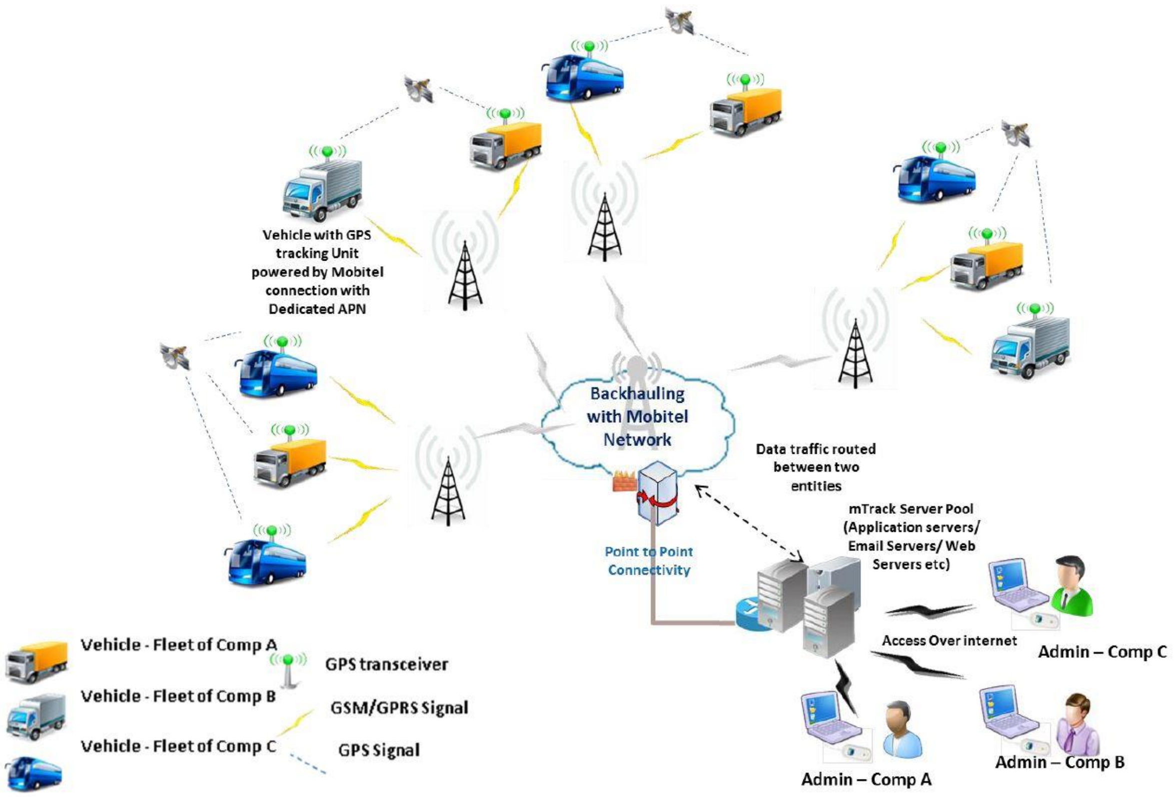

Vehicle with GPS tracking Unit powered by Mobitel connection with Dedicated APN
Backhauling with Mobitel Network
Data traffic routed between two entities
mTrack Server Pool (Application servers/ Email Servers/ Web Servers etc)
Point to Point Connectivity
Access Over internet
Admin – Comp C
Admin – Comp A
Admin – Comp B
Vehicle - Fleet of Comp A
Vehicle - Fleet of Comp B
Vehicle - Fleet of Comp C
GPS transceiver
GSM/GPRS Signal
GPS Signal

Figure 1: How to mTrack Works

Tracking device

The tracking device which shall be installed in the vehicles to be tracked shall have the following features:

- GPS capability
- Support GSM SIM with SMS and GPRS communication capability
- Fuel monitoring facility
- Ability to update at a frequency of once in 10 seconds
- Ability to store and forward location info in case of Low or no network reception
- Sunlight protective cover to stand in the local temperatures
- Operate in between 12-24V DC current
- Tamper proof SIM cover with detection

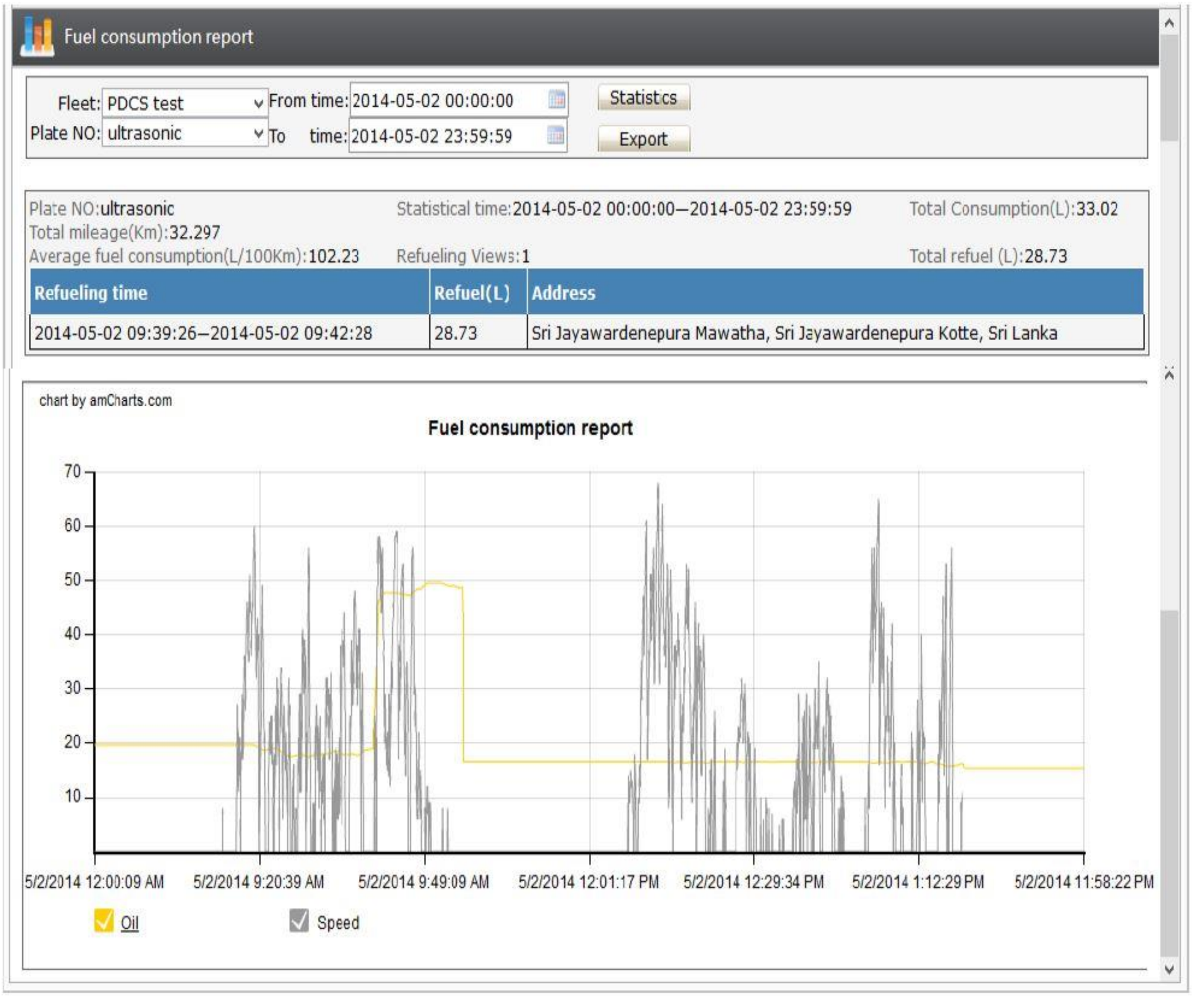
Fuel Sensor

This is an Ultrasonic Fuel Sensor, a state-of-the-art technology that is expected to revolutionize the monitoring mechanism for fuel consumption of vehicles in the most reliable and efficient manner for the first time in Sri Lanka.

The Ultrasonic type fuel sensor uses non-contact ultrasonic measuring technology which ensures high measuring accuracy. The special fuel sensor does not require any tampering of the fuel gage system and does not require the insertion of special measuring devices to the fuel tank which could lead to dangerous situations. The Ultrasonic does not require such tampering and is hence considered far safer, more accurate and user-friendly.

Suitable for all kinds of vehicles such as cars, vans, buses, prime movers and other forms of transportation which could digitally record vehicle refueling, fuel availability and minimize fuel theft, the sophisticated mechanism avoids wastage of resources and improves operating efficiency.

- Works only with square shaped fuel tanks
- High accurate measure and monitor of the fuel consumption of vehicles
- Work with GPS device
- Safest and most environmentally friendly fuel sensor service available in the market



Fleet: PDCS test
Plate NO: ultrasonic
From time: 2014-05-02 00:00:00
To time: 2014-05-02 23:59:59

Plate NO:ultrasonic
Total mileage(Km):32.297
Average fuel consumption(L/100Km):102.23
Statistical time:2014-05-02 00:00:00—2014-05-02 23:59:59
Refueling Views:1
Total Consumption(L):33.02
Total refuel (L):28.73

Refueling time	Refuel(L)	Address
2014-05-02 09:39:26—2014-05-02 09:42:28	28.73	Sri Jayawardenepura Mawatha, Sri Jayawardenepura Kotte, Sri Lanka

Figure 2: The Fuel Consumption Report

Base Server

All the tracking devices installed for this solution shall communicate with the Base server. The base server shall have interfaces to communicate with a variety of devices through SMS and GPRS in varying time frequencies.

The Base server shall maintain a registry of tracking devices set for monitoring

mTrack Info Site

The solution shall consist of a comprehensive user site which shall be accessible through internet.

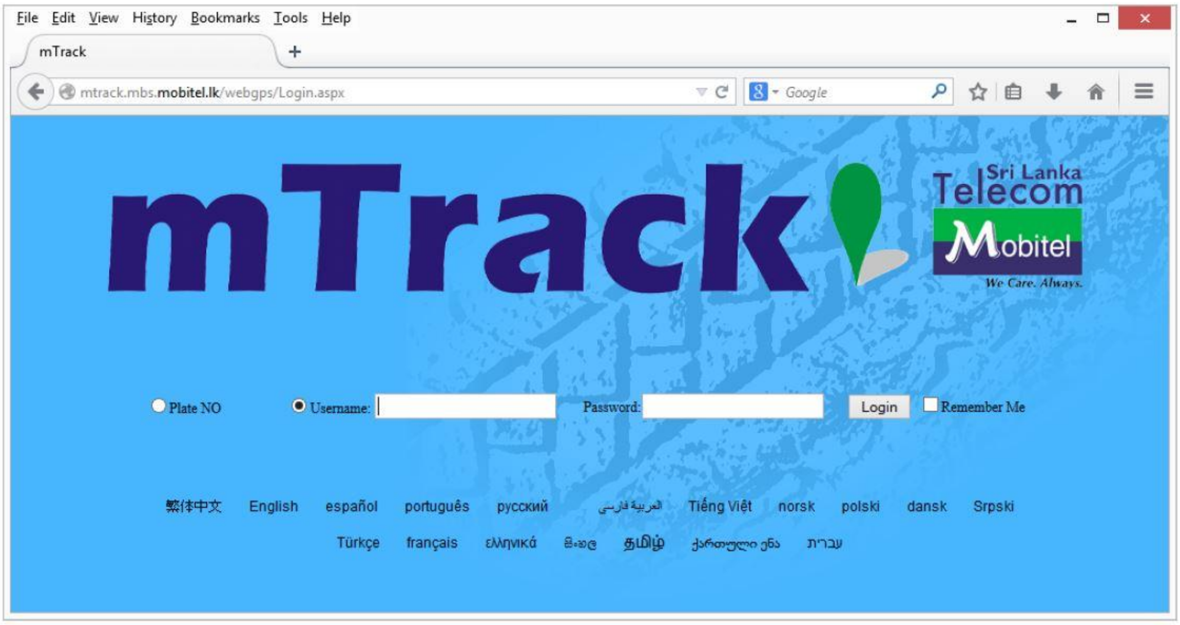
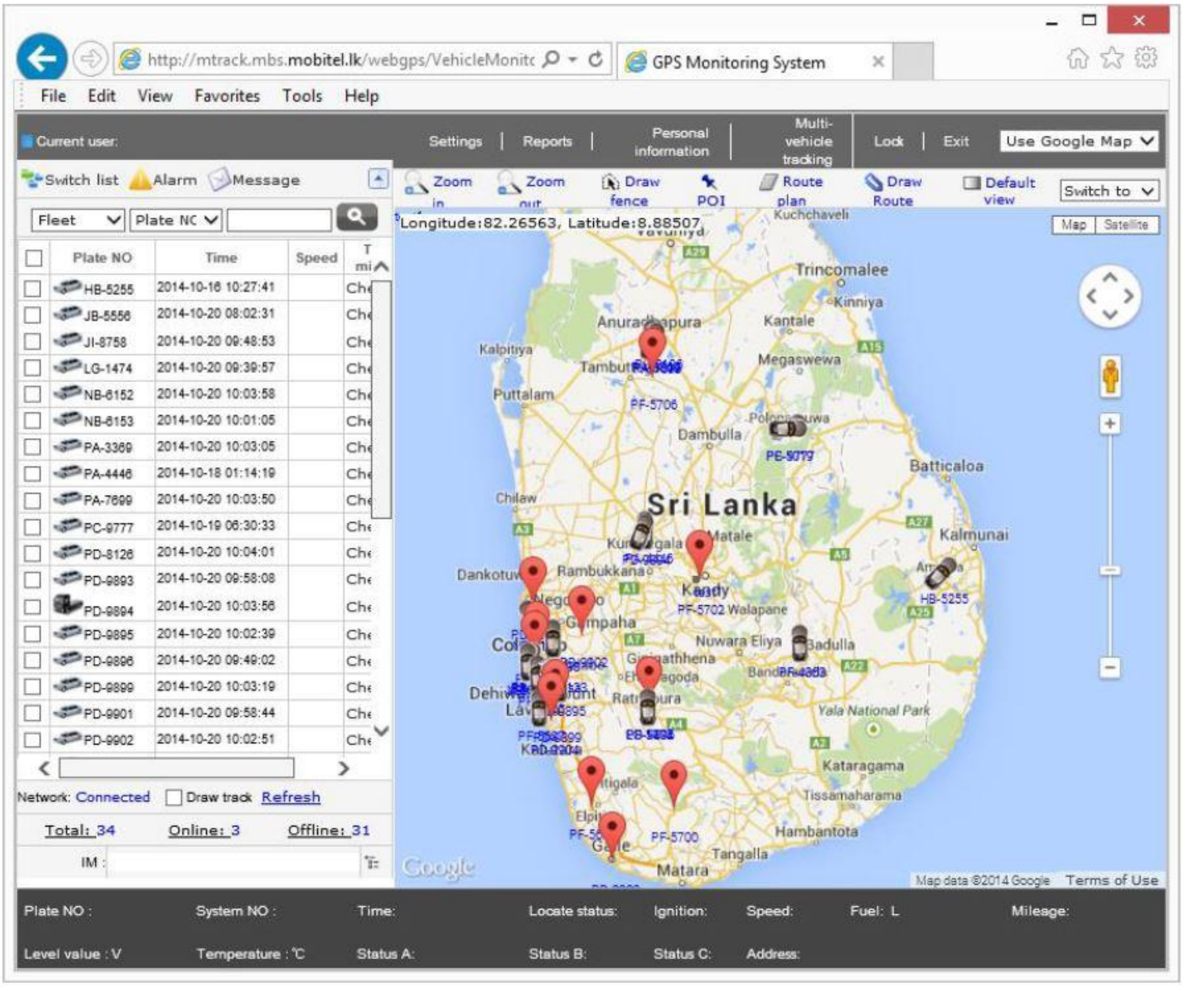


Figure 3: mTrack Login Page

The credentials shall be created by Mobitel and the login info shall be handed over to the designated corporate admin. Client admin shall enter the client ID and the Password to login to the site.

Users shall be able to track the devices assigned to them, generate reports and change password.



Longitude:82.26563, Latitude:8.88507

□	Plate NO	Time	Speed	T mi
□	HB-5255	2014-10-16 10:27:41		Che
□	JB-5556	2014-10-20 08:02:31		Che
□	JI-8758	2014-10-20 09:48:53		Che
□	LG-1474	2014-10-20 09:39:57		Che
□	NB-6152	2014-10-20 10:03:58		Che
□	NB-6153	2014-10-20 10:01:05		Che
□	PA-3369	2014-10-20 10:03:05		Che
□	PA-4446	2014-10-18 01:14:19		Che
□	PA-7699	2014-10-20 10:03:50		Che
□	PC-9777	2014-10-19 06:30:33		Che
□	PD-8126	2014-10-20 10:04:01		Che
□	PD-9893	2014-10-20 09:58:08		Che
□	PD-9894	2014-10-20 10:03:56		Che
□	PD-9895	2014-10-20 10:02:39		Che
□	PD-9896	2014-10-20 09:49:02		Che
□	PD-9899	2014-10-20 10:03:19		Che
□	PD-9901	2014-10-20 09:58:44		Che
□	PD-9902	2014-10-20 10:02:51		Che

Network: Connected □ Draw track Refresh

Total: 34 Online: 3 Offline: 31

IM :

Plate NO : System NO : Time: Locate status: Ignition: Speed: Fuel: L Mileage:

Level value : V Temperature : ℃ Status A: Status B: Status C: Address:

Figure 4: mTrack User Interface

Logged in users shall be able to track the vehicles that are assigned to them. There shall be the option to select the vehicles that need to be tracked within the assigned list. This can be one-many-all. Once selected, there shall be a map with an indication of the locations tracked.

More info

A mouse-over the tracked vehicle icon shall bring up a balloon with the following info:

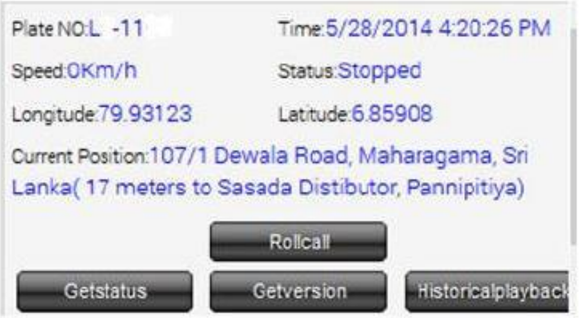


Plate NO:L -11 Time:5/28/2014 4:20:26 PM

Speed:0Km/h Status:Stopped

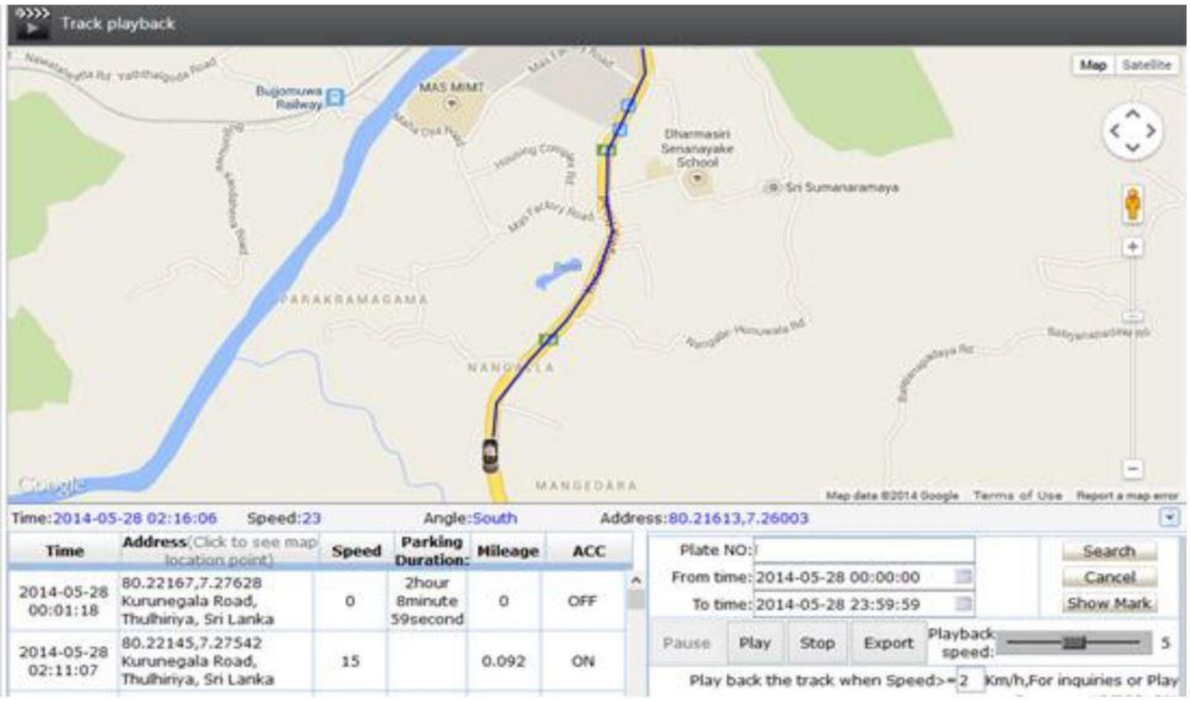
Longitude:79.93123 Latitude:6.85908

Current Position:107/1 Dewala Road, Maharagama, Sri Lanka(17 meters to Sasada Distibutor, Pannipitiya)

Figure 5: Live Status

Movement history

There shall be the option to view the track history of a specific vehicle(s). There shall be the option to select the vehicles for the plot and the period. The period shall be defined as start date and time and the end date and time. The route of the vehicle movement shall be plotted on the map with direction indicators. A mouse-over the route shall bring up a balloon with Date-time stamp and the vehicle number. If there is more than one vehicle selected, then each route shall be indicated with a unique color. This information shall be available up to 6 months.



Time:2014-05-28 02:16:06 Speed:23 Angle:South Address:80.21613,7.26003

Time	Address(Click to see map location point)	Speed	Parking Duration:	Mileage	ACC
2014-05-28 00:01:18	80.22167,7.27628 Kurunegala Road, Thulhiriya, Sri Lanka	0	2hour 8minute 59second	0	OFF
2014-05-28 02:11:07	80.22145,7.27542 Kurunegala Road, Thulhiriya, Sri Lanka	15		0.092	ON

Plate NO:

From time: 2014-05-28 00:00:00

To time: 2014-05-28 23:59:59

Playback speed: 5

Play back the track when Speed>=2 Km/h,For inquiries or Play

Figure 6: Historical Playback Report

Reports

The system shall have the capability to generate the following reports:

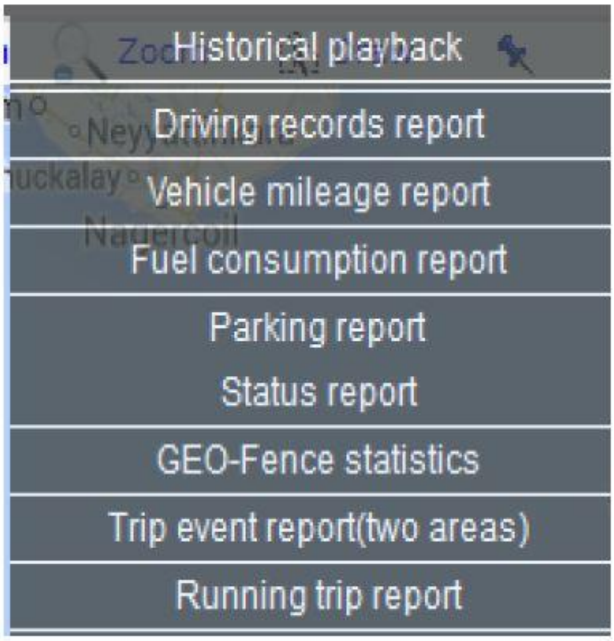
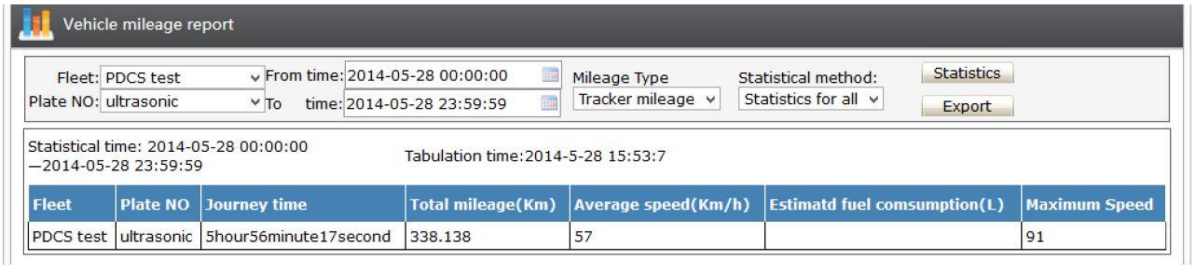


Figure 7: Reports

All these reports can be exported as an Excel data sheet simply by clicking on the export button in any report interface.



Vehicle mileage report

Fleet: PDCS test From time: 2014-05-28 00:00:00 Mileage Type Statistical method: Statistics

Plate NO: ultrasonic To time: 2014-05-28 23:59:59 Tracker mileage Statistics for all Export

Statistical time: 2014-05-28 00:00:00
—2014-05-28 23:59:59

Tabulation time:2014-5-28 15:53:7

Fleet	Plate NO	Journey time	Total mileage(Km)	Average speed(Km/h)	Estimatd fuel comsumption(L)	Maximum Speed
PDCS test	ultrasonic	5hour56minute17second	338.138	57		91

Figure 8: A Mileage Report

Note

mTrack uses GPS, GPRS, and ultrasonic technologies and the results given as location, mileage, and fuel consumption are best approximations to the actual readings. The accuracy may vary due to various technical and atmospheric conditions.

2.2.2 PickMe Overview

PickMe uses a smartphone installed tracking application to track their taxis. Each and every taxi driver has that device and they need to run their application when they are available. Otherwise they must stop the application running. Fare of the tour also calculated by the application automatically. PickMe facilitate two ways to reserve a taxi for passengers. One way is from their website and the other way is using their mobile App. PickMe App can detect the passengers' location and track the nearest available taxis automatically. They are showing in a map in the App with real-time movements. In the PickMe website, passenger need to enter pickup location, date and time for reserve a taxi. PickMe App has following features and technologies.

PickMe App is a locally built Live Taxi Hailing System that allows the Passenger to:

- Call a ride from anywhere in Sri Lanka.
- Call a ride at any time of the day.
- Call the closest taxi available – assuring the fastest service.
- Choose a ride - may it be a tuk tuk, a nano car or even a limo.
- Choose desired taxi service.
- Choose desired hire rates.
- Track the journey and share details with family and friends.
- Retrieve lost belongings.
- Calculate the fare accurately.
- Share a ride with a friend.

PickMe App requires permission from passenger's mobile device to access:

- Accounts - Required by Google Play Services to use the Google Map and other Google services.
- Phone / Device ID / Call Information - To directly call drivers once the trip is accepted.
- Camera / Photos / Media - To upload user's profile image.
- GPS / Network Location - To identify user's pickup location.
- SMS - To read the verification code during registration.
- Contacts - To be able to set an emergency contact.

2.3 Review of Systems

In the above mentioned systems, they have designed the software solutions for common purposes in vehicle transport industry. They have well organized web based automated vehicle transport management system with transport order management system. Concept is good but the requirement of public bus service is slightly different than these solutions.

According to mTracker there are many features implemented which are not important for passengers in public bus service. But it is best for monitor the buses. On the other hand, it is more complex and costly to implement in public bus service.

Considering PickMe, it is more similar to proposed easyBus of this project. As an example both PickMe and easyBus systems use GPS technology to track vehicles with the help of Google APIs. The main difference between PickMe and easyBus systems is that the PickMe system is designed specifically for private transportation whereas the easyBus system is designed to fit both private and public transportations. Therefore, even though both systems allow the passenger to track the nearest service vehicle to the passenger's current location, the difference is that in PickMe the nearest service vehicle will take the passenger to the intended destination whereas in easyBus system the nearest service bus may or not take the passenger to the intended destination depending on bus routes. To facilitate this, in contrast to PickMe service and in addition to tracking the nearest bus, the easyBus system also allows the passenger to choose/track the most suitable bus for his/her journey; ie. if the bus does not go to the intended destination, the passenger can track the bus which goes closer to the intended destination.

Chapter 3: Analysis and Design

3.1 Introduction to Analysis

Analysis is the most significant phase of the software develpoment life cycle. It will help developers to get clear idea about the problem domain and what are the user's requirements for the new system before start developing.

The way of required information gathering, analyzing of gathered information and management techniques of analyzing information are explained in this section.

In addition to that functional and non-functional requirements of the proposed system, difficulties in completing requirements and how to overcome them in the best possible way are also denoted. This section alos interprets what are the best method to develop the system.

3.2 Requirement Gathering and Fact Finding

Purpose of every system is to full fill needs of user/client. Those needs named as requirements of the system. To have the clear idea about the requirements, analyst must go through the existing system and must collect details from every stakeholders.

Gathering requirement can be done by following various fact finding techniques like interviews, questionnires, etc.

3.2.1 Background Reading / Inspect Documents

Having good understand of the organization's process will leads to get a good understanding about requirements.

You can have a good understanding in background of the reuirement via inspecting reports, and other documents relates with current process. It is used for design the database and identifies the fields of the tables' structure easily.

3.2.2 Interviewing

With this technique analyst can collect the information from groups or individuals. In this system, by interacting face to face with officers of SLTB and NTC, selected group of passengers, drivers and conductors, and their responses have provided valuable ideas about the usability of the system and current difficulties with the manual system.

3.2.3 Observation

By watching passengers and their normal work flow carrying out their operations to learn about system, helped to gather much needed information's for the development.

3.2.4 Questionnaire

Asking collection of questions from passengers that helps to identify the requirement of the system.

3.2.5 prototyping

By showing prototype of the system, functional and non-functional requirements of the system can be identified.

3.3 PACT Analysis

PACT (People, Activities, Contexts, Technologies) framework help to carry out the PACT analysis methodically and facilitate the designers to understand many aspects before rushing to a particular design. It helps to select the most suitable alternative design and develop successful products.

People use technologies to undertake activities in contexts describes in this section.

3.3.1 People

- Administrators and passengers are the main groups of the easyBus system.

- The admins should have a sound knowledge on the complete process of easyBus system and on the functionalities of it, to login through the admin login and to control.
- The admins should have a general knowledge on the bus service of Sri Lanka.
- The admins should have a general knowledge on the bus routes, halts and buses in Sri Lanka.
- The admin should have a sound knowledge on general English and general information technology.
- The passengers of the buses should know, at least, one of the three languages; Sinhala, Tamil or English.
- The passengers will have to check the display related to the easyBus system at the bus halts, from time to time, as it gets live updates.
- All the passengers should know their intended destinations properly.
- The passengers who use mobile phones and computers should have a sound knowledge on using the internet and on websites.
- The passengers should constantly check the display related to the easyBus system in the bus while travelling as it gets live updates.

3.3.2 Activities

- easyBus display inside the bus;
  - ➢ It will display the locations of the bus under three main headings namely; ‘previous’, ‘current’ and ‘next’ in order to facilitate the passenger to trace the bus and the names of the halts will also be displayed in all three language (Sinhala, Tamil and English).
  - ➢ Firstly, when the bus begins its journey, the name of the bus stand at which it starts the journey will be displayed under the ‘current’ heading. Under the ‘next’ heading it will be displayed the names of the next first halt. At this point nothing will be displayed under ‘previous’ heading.
  - ➢ After the journey started from the bus stand and from the first bus halt onwards, when the bus is going in between two different halts, nothing will be displayed under the ‘current’ heading. Under the ‘previous’ heading it will be displayed the last halt that had been passed by the bus and under the ‘next’ heading it will be displayed the name of the next closest halt.

    - When the bus arrives at a halt, the name of that halt will be displayed under the ‘current’ heading. Under ‘previous’ heading it will be displayed the last bus halt that had been passed by the bus and under ‘next’ heading it will be displayed the next closest bus halt.
    - When the bus arrives at its last stop, the name of that halt will be displayed under the ‘current’ heading. Nothing will be displayed under the ‘next’ heading and the name of the previous halt will be displayed under the ‘previous’ heading.

- easyBus display at bus halts;
    - On the display(s) at the bus halt(s), the routes and the numbers of the buses which are arriving to the current bus halt (only the buses that have started the journey) will be live updated. Those details will be moved on the display depending on how closer the bus is, to the current bus halt.
    - When the busses arrive to the bus halt, those arrived busses will be displayed in a separate column on the display.
    - The departed buses from the bus halt, will be removed from the display after their departure.

- Online searching for buses;
    - On the easyBus system website (bus search engine), the passengers will have to select their current bus halt or the nearest bus halt and search for their intended destination and then the easyBus show them the closest bus to their current bus halt or the nearest bus halt which would either go to their intended destination or which would go through the route of the intended destination.
    - On the easyBus system website (time schedule search engine), the passengers will have to select the starting point and the intended destination to search the time schedule of the buses heading to the intended destination through/from the starting point.

- Admin Login;
    - Logging in using the admin username and the password.
    - Adding new bus halts, new busses, new departure/arrival times and new bus routes to the easyBus system.
    - Make changes to the data in the easyBus system regarding the busses, departure/arrival times, routes and halts whenever necessary.

3.3.3 Contexts

- Busses will be on route at any time.
- Busses might get delayed due to the traffic, due to an accident or any other circumstance.
- Due to the competition between private buses and SLTB buses, buses may arrive early before the expected time.
- The bus may not arrive due to unavoidable circumstances such as breakdowns, accidents, natural disasters etc.
- The GPRS connection may drop down due to the lack of cell towers, technical errors or whether changes such as heavy rain.
- The GPS connection in buses may drop down due to the technical errors of the GPS trackers attached to the buses.
- The displays at bus halts or the displays which are attached to the buses may not work due to the dropping down of their power supply.
- The internet connection of the mobile users / computer users may drop down due to exceeding data or any other circumstance.
- Such connections can also be dropped down due to the dropping down of the power supply of the devices such as dead battery, etc.

3.3.4 Technologies

- GPS Technology;
    - ➢ To get the coordinates of the buses.
- GSM Technology;
    - ➢ To get the data and send the data.
- Google Maps;
    - ➢ To show the tracks of the buses.
- Microcontroller;
    - ➢ It is responsible for acting as an interface between GSM module and GSM receiver.
- LCD Display;
    - ➢ To display the details at the bus halts and inside the buses.

3.4 Requirement Analysis and Management

Requirement management is a critical task. The gathered requirements should be analyzed and must check for ambiguities to be solved, otherwise it will lead the system to a problematic stage later.

In this process of maintaining document of system, analyzing, tracing, prioritizing tasks and agreeing on requirement changes. Accepting new requirements unconditionally, will affect the project quality and it will lead to project delays.

3.5 Requirements gathered from Analysis

This section presents functional, non-functional and domain requirements above analysis.

3.5.1 Functional Requirements

Those are the major functions are to be implemented.

- Login API for the admin
- Monitoring and controlling API for the admin
- Display API of the previous, current and next bus halts, for the buses
- Display API of the bus routes and the next bus halt locations of the arriving buses for the bus halts
- Online searching API of the next bus to the intended destination for the passengers
- Online searching API of the arrival/departure times of all the buses going to/through the intended destination for the passengers

3.5.2 Non-functional Requirements

Those are to be implemented to maintain the quality of the system.

- Get the coordinates from any GPS tracker to the easyBus system
- Output of the display APIs fits on any LCD monitor
- Passengers can access the online searching APIs on any desktop/laptop or mobile device

- The details of the display APIs and the online searching APIs are in Sinhala and Tamil in addition to English for the passengers

3.5.3 Domain Requirements

Those requirements need for the accuracy of the system.

- The list of all the bus stops on route no. 69 (Kegalle - Ratnapura)
- The timetable and the identity; a name/number/code to make each bus identifiable separately, of all the busses going on route no. 69
- The list of all the other bus routes across the no. 69 route and the timetable and the identities of all the busses using those other bus routes across the no. 69 route
- All the data supplied to the ticketing machine related to the route no. 69
- Dynamic coordinates of all the moving buses on route no. 69
- Coordinates of all the bus stops on route no. 69

3.6 Introduction to Design

System design phase produces the design specification for the system to be implemented. Well drawn diagrams will make the designer’s work easy. For example having entity relation diagram will lead database designer’s work easy.

There are several methodologies that can be used to while developing a system like warerfall method, spiral method, agile methodology, etc.

For this system implementation, Rational Unified Process (RUP), an iterative development methodology is the best methodology. Selected because easyBus GPS Tracking and Management System have higher risk of changing through the project duration.

3.7 System Architecture

The system architecture for the system is given below in Figure 9.

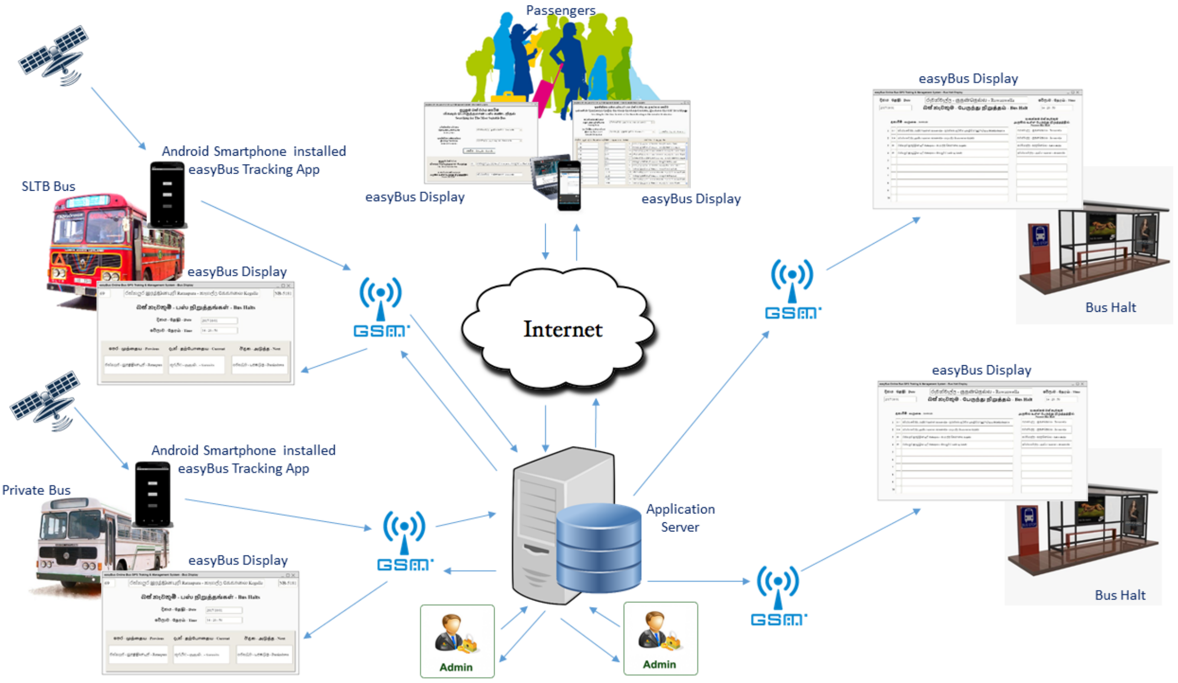


Figure 9: How easyBus Works

3.8 Development Approch

Rational Unified Process (RUP) is the methodological approach which will be used in the proposed system. It is an iterative software developmen framework created by Rational Software Cooperation (See Figure 10). This process consists of 4 phases. Namely,

1. Inception – Identify the system scope by gathering the user requirements correctly. Business case is established. Should be concerned about the cost/schedule estimate, risk assessment etc.

2. Elaboration – Mitigate the key risks identified from the previous stage. Develop an overall project plan.

3. Construction – Main focus is on the development of the codes.

4. Transitition – End user training and acceptance testing is carried out here.

Rational Unified Process (RUP)

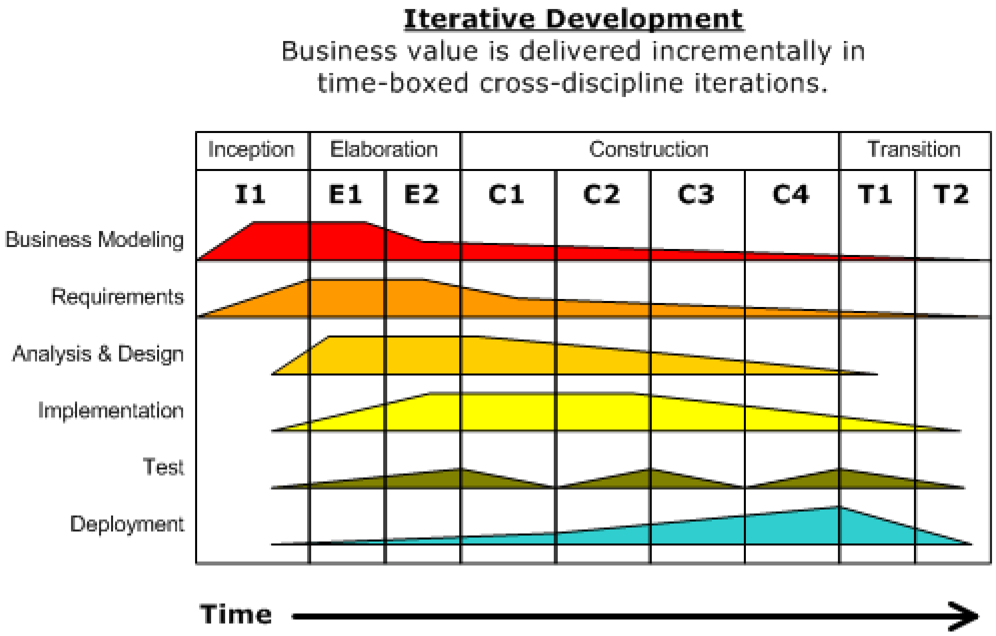


Figure 10: RUP

easyBus system is a web based solution on client sever architecture. And also it has a mobile application. System administrators/operators, drivers and passangers have acess from anywhere to the system.

3.2 Database Design

This section describes the database design usnig ER diagram and used its assumptions.

- Assumptions:
  - ➢ There can be several buses which start and finish the journey at the same time even though they should use different routes and might/might not have started their journey at the same bus stand/halt
  - ➢ There can be only one SLTB bus and one private bus which can start the journey at a similar time, from the same bus stand/halt, travel on the same route and arrive the destination at the same time
  - ➢ A bus can have more than one trip
  - ➢ A bus can travel only on one route

- One route can be used by more than one bus
- There should be atleast two halts for a route
- A halt should have at least on route
- It is mandatory for an administrator to manupulate the data relavant to time schedule, buses, routes, halts and also an administrator can omit doing so
- It is madatory for a bus to have a route and a time schedule
- It is madatory for a time schedule to have a bus

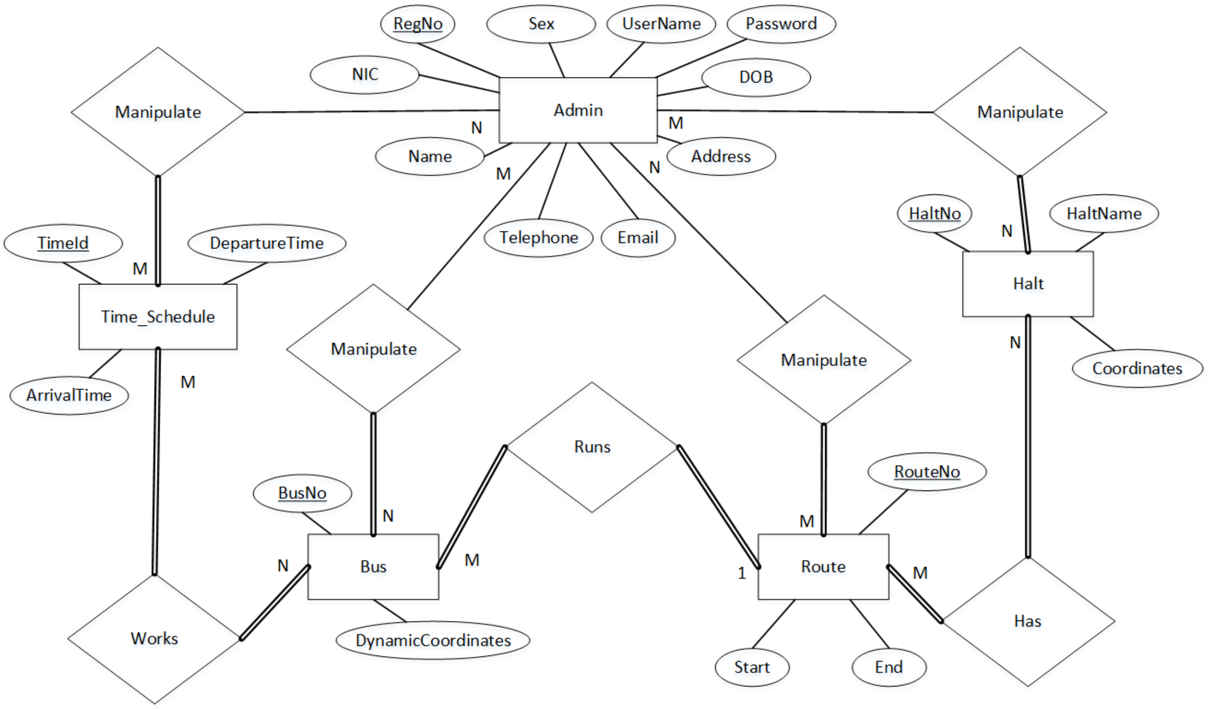


Figure 11: Entity Relationship (ER) Diagram

3.3 High-level Data Flow Diagram for the system

Mainly easyBus system has four entities such as Admin, Passanger, Bus Halt and Bus. Admin can login the system using username and the password. Then admin can view and edit the data of the system. According to the dynamic coordinates sends by the buses, system generate outputs to the relevent bus halt and the relevent bus. Request of the passanger also depend on the dynamic coordinates of the buses.

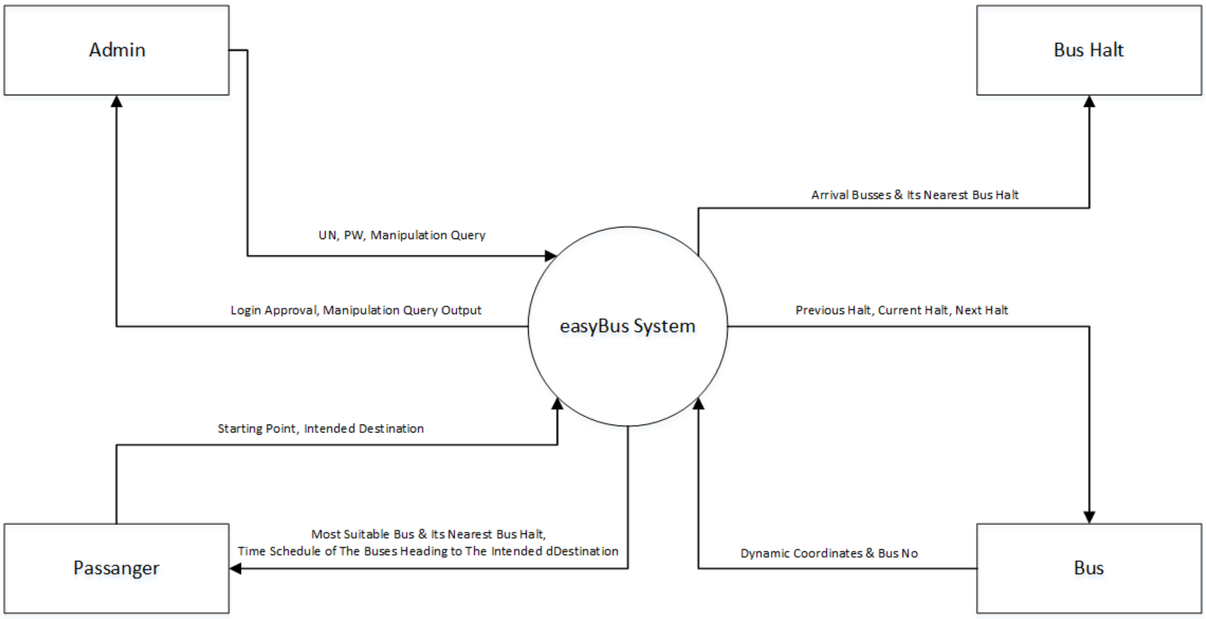
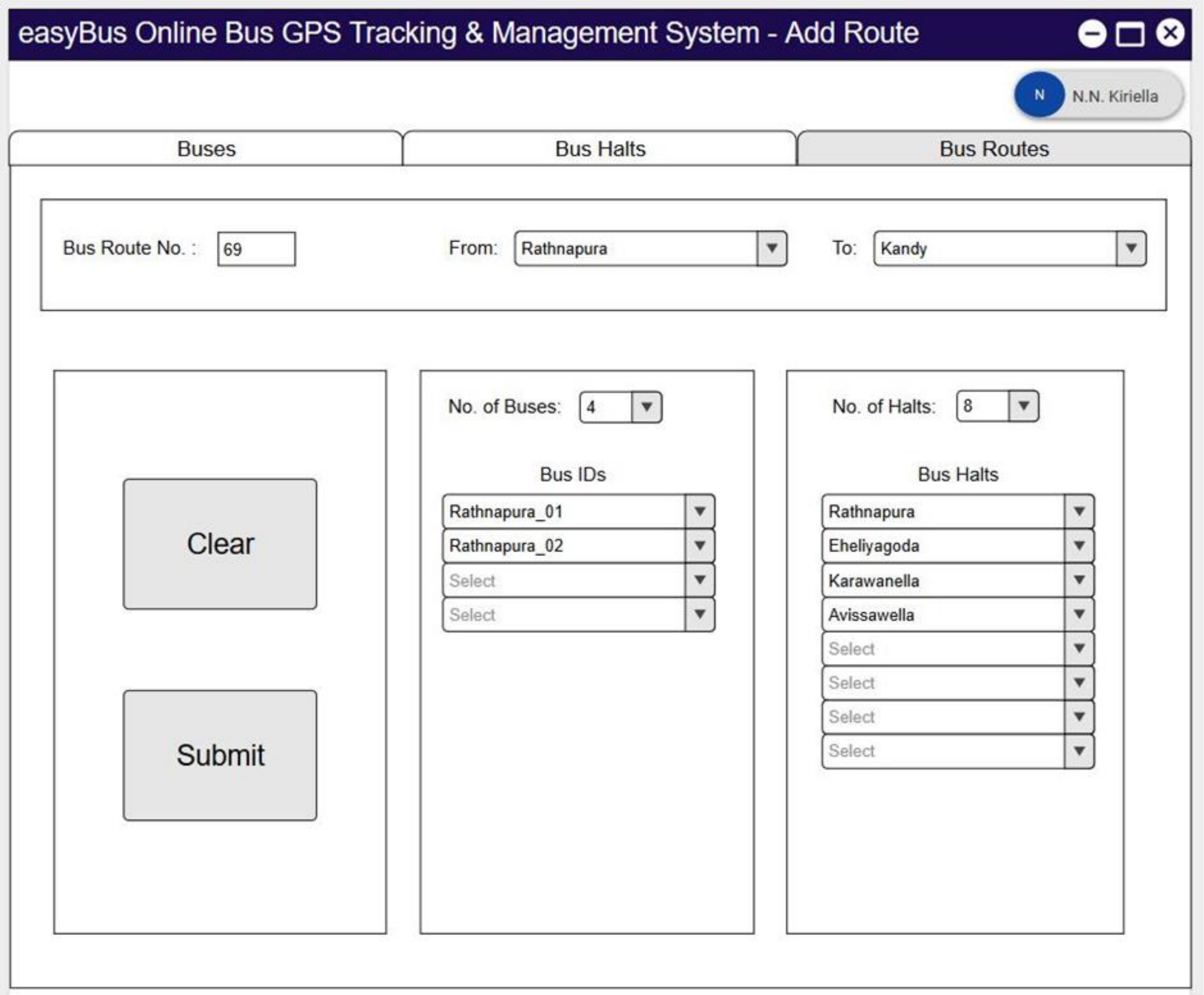

Admin
Bus Halt
Arrival Busses & Its Nearest Bus Halt
UN, PW, Manipulation Query
Login Approval, Manipulation Query Output
easyBus System
Previous Halt, Current Halt, Next Halt
Starting Point, Intended Destination
Most Suitable Bus & Its Nearest Bus Halt,
Time Schedule of The Buses Heading to The Intended dDestination
Passanger
Dynamic Coordinates & Bus No
Bus

Figure 12: Context Diagram

3.4 User Interface Design

Following data entry form is related to a part (a function) of the admin login interface of the '**easyBus – Online GPS Based Bus Tracking & Management System**'. The name of the system and the name of the window (function name) are on the title bar (top). Logged account of the admin is shows under the title bar. It shows drop down menu with log out option when click on it. There are 3 tabs (sub functions) in the window (main function) and the active one is called '**Bus Routes**'. First of all, admin should enter the details of buses and the bus halts when come to the bus routes tab. After that admin can add bus route details. Bus routes form contains 4 sections. '**Bus Route No.**' is a text field. '**From**', '**To**', '**No. Of Buses**', and '**No of Halts**' fields get values from drop down menus. Drop down menus load values from the data entered in previous sections (**Buses**, **Bus Halts**). '**Bus IDs**' rows and '**Bus Halts**' rows are auto generated based on values of '**No. of Buses**' and '**No. Of bus halts**'.



easyBus Online Bus GPS Tracking & Management System - Add Route

N.N. Kiriella

Bus Route No. : 69

From: Rathnapura

To: Kandy

No. of Buses: 4

Bus IDs

Rathnapura_01
Rathnapura_02
Select
Select

No. of Halts: 8

Bus Halts

Rathnapura
Eheliyagoda
Karawanella
Avissawella
Select
Select
Select
Select

Figure 13: Admin Login Interface - Add Route Function

If rows increase than the size of the section, scroll bar will appear in related section. Also drop down menus of ‘**Bus IDs**’ and ‘**Bus Halts**’ load values from the data entered in previous sections. ‘**Clear**’ button clears all the values of the form and the ‘**Submit**’ button send all the data in the form to the database. All fields must be filled in the form and if not filled, error message will pop up and required fields will be highlighted with red asterisk when click on ‘**Submit**’ button. Drop down menus of the form minimize the data entry errors. Also it helps to manage sizes of data fields well.

Please refer to Appendix A: User Interface Design to see other main user interface designs.

Chapter 4: Implementation

4.1 Introduction

During the implementation process CodeIgniter web framework used to build the code and web interfaces which were the outcomes of the design process. Database which is designed using the ER diagram was created using MySQL software. Mobile application was developed in Android Studio development environment.

4.2 Technological Review

The following Table 1 describes used technologies and reasons to select them for the development of the easyBus system.

Technology	Justification for the choice
PHP (language)	Fast load time, less expensive software, less expensive hosting, database flexibility, increased available programming talent, etc.
CodeIgniter (framework)	Faster execution time completes web development job quickly, file management, great security functions, simple configuration, less coding requirement & speedier development possible, great support from community, cache class, easy handling of errors, convenient template solutions, stage by stage testing in development stage, etc.
MySQL (database)	Quick-start capability, rock-solid reliability, high availability, on-demand scalability, etc.
Android (mobile operating system)	Wireless communications, mobility, portability, etc.

Table 1: Used technologies and reasons to select for web based easyBus system

4.2 User Interfaces

Under this section will explain the main interfaces of the implemented easyBus system. It will give the overall picture of the web interface and the mobile interface.

Figure 14 shows the view of an admin acount after login to the easyBus sytem in web browser. In the left hand side, there are five options; Users, Buses, Routes, Halts and Time Schedules to manupulate data in database tables. Front End option contains the main four APIs for the passangers. It helps to monitor two type of the displays of the system and test whole four APIs of the front-end with limited resourses. Any display of a bus can be view by entering the bus registaration number using Bus Display option. And also any display of a halt can be view by entering the halt number using Bus Halt Display option. Other two search engines of the system can be test using Bus Search Engine and Bus Time Schedule Search Engine options.

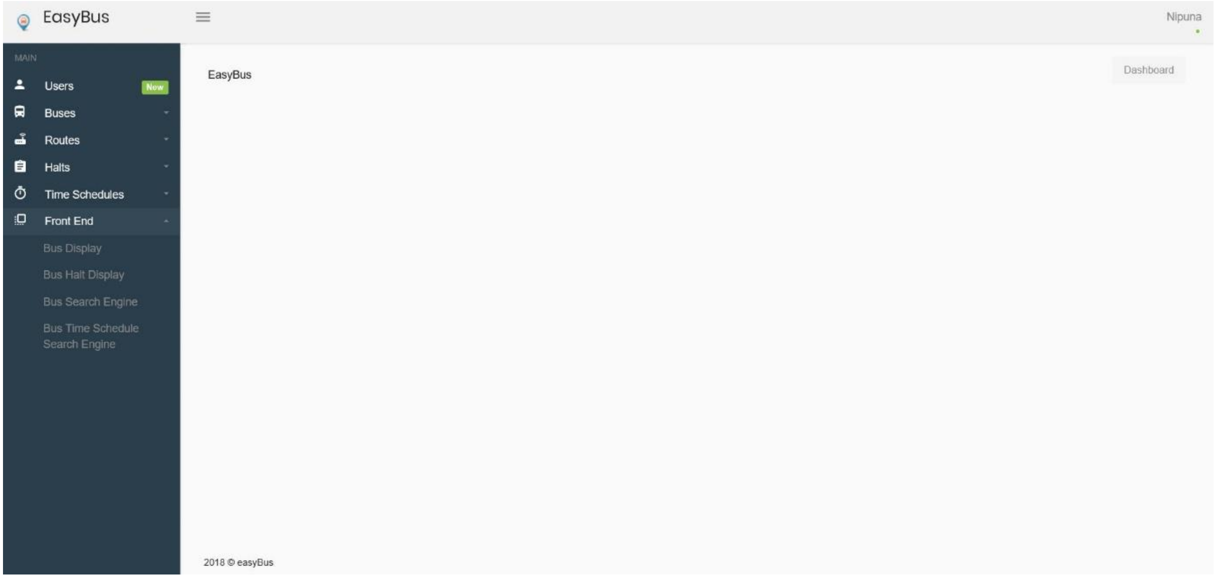


Figure 14: View of an admin account dashboard

Figure 15 shows the GPS tracking application installed in an Andoid smart phone. Any Android device with this application must be used by every buses registerd in the easyBus system. In the registeration process of a bus, registaraion number will be hard coded in the GPS tracking application. Therefor each and every GPS tracking applications will be unique. Start Trip button of the interface must be pressed before start the trip. Then the application start to send the coordinates of the current location of the bus with its hard corded registraion number and '1' as a flag to the online sever in every 5 seconds. Hard coded registraion numbers are used to identify the buses uniquely when reciving the current coordinates from the buses by the system. And also flag ‘1’ is used to identify activate buses. In this situation Start Trip button is disabled and End Trip button is enabled in automatically. In the end of the trip, driver must be pressed End Trip button to stop the execution of the GPS tracker. It will be enable the Start Trip button and disable the End Trip button in automatically. Also send send the coordinates of the current location of the bus with its hard corded registraion number and ‘0’ as a flag. Flag ‘0’ is used to identify passive buses. Driver can view the coordinates of the bus in anytime by pressing Show Location button.

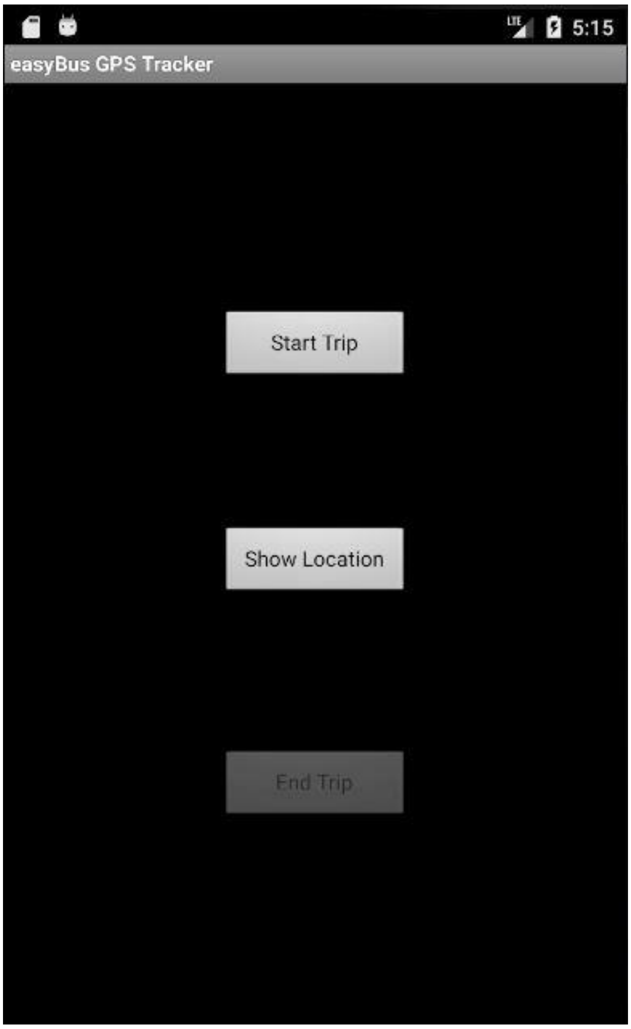
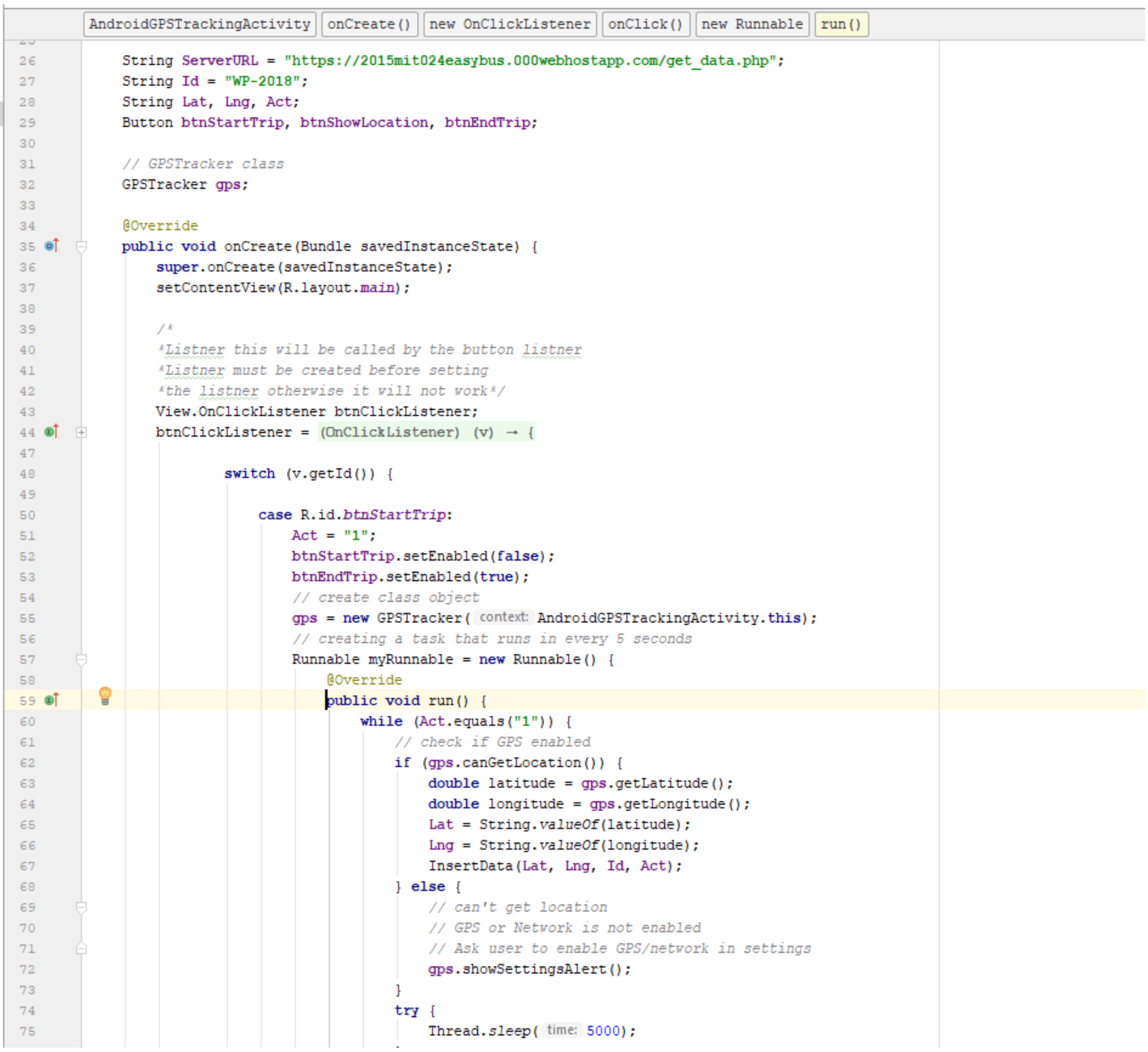


Figure 15: Interface of the Android mobile application

Other main implemented outputs are similar to designed interfaces shows in Appendix A: User Interface Design.

4.3 Code Segments

Most important part of the system is tracking the location of the buses. Following code segment shows how to get the location of the buses in the mobile application.



```
String ServerURL = "https://2015mit024easybus.000webhostapp.com/get_data.php";
String Id = "WP-2018";
String Lat, Lng, Act;
Button btnStartTrip, btnShowLocation, btnEndTrip;

// GPSTracker class
GPSTracker gps;

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);

    /*
    *Listner this will be called by the button listner
    *Listner must be created before setting
    *the listner otherwise it will not work*/
    View.OnClickListener btnClickListener;
    btnClickListener = (OnClickListener) (v) → {

            switch (v.getId()) {

                case R.id.btnStartTrip:
                    Act = "1";
                    btnStartTrip.setEnabled(false);
                    btnEndTrip.setEnabled(true);
                    // create class object
                    gps = new GPSTracker( context: AndroidGPSTrackingActivity.this);
                    // creating a task that runs in every 5 seconds
                    Runnable myRunnable = new Runnable() {
                        @Override
                        public void run() {
                            while (Act.equals("1")) {
                                // check if GPS enabled
                                if (gps.canGetLocation()) {
                                    double latitude = gps.getLatitude();
                                    double longitude = gps.getLongitude();
                                    Lat = String.valueOf(latitude);
                                    Lng = String.valueOf(longitude);
                                    InsertData(Lat, Lng, Id, Act);
                                } else {
                                    // can't get location
                                    // GPS or Network is not enabled
                                    // Ask user to enable GPS/network in settings
                                    gps.showSettingsAlert();
                                }
                                try {
                                    Thread.sleep( time: 5000);
```

Figure 16: Location Tracking Functions of The Mobile App

Chapter 5: User Evaluation and Testing

5.1 Test Plan

5.1.1 Introduction

The Test Plan is designed to prescribe the scope, approach, resources, and schedule of all testing activities of the easyBus system.

The plan identifies the items to be tested, the features to be tested, the types of testing to be performed, the resources and schedule required to complete testing, and the risks associated with the plan.

5.1.2 Test Strategy

5.1.2.1 Scope of Testing

Mainly, the following seven functions of the easyBus system are going to be tested.

- Display of the previous halt of a bus
- Display of the current halt of a bus
- Display of the next halt of a bus
- Display of arriving/arrived buses of a halt
- Nearest bus searching
- Time schedule searching
- Login API for the admin

Please refer to Appendix B: Scope of Testing to see detailed test modules of above functions.

5.1.2.2 Test Type

In the easyBus system, there are 3 types of testing which should be conducted.

- **Integration Testing** (Individual software modules are combined and tested as a group.)
- **System Testing**: Conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.
- **API testing**: Test all the APIs create for the software under tested.

5.1.2.3 Risk and Issues

Risk	Mitigation
Lack of GPS tracking devices	One GPS tracking device can be used for the testing. According to that, change the table values related to the data of other GPS tracking devices (virtual) in the database.
Lack of moving buses	Change the table values related to the movements of the busses (virtual) in the database.
Lack of display monitors	Display an one output screen of the easyBus system on the developer’s machine at a time.
Lack of actual coordinates of the bus halts	Use sample coordinates for the bus halts.

Table 2: Risk and Issues

5.1.2.4 Test Logistics

Who will test?

- The system will be tested by the developer as a passenger and as an administrator on the developer’s machine.

When will test occur?

- The tester will start the test execution when all the following inputs are ready.
    - Software is available for testing
    - Test specification is created
    - Test environment is built
    - Enough test data is collected or created for testing

5.1.3 Test Objective

The test objectives are to verify the functionality of the easyBus system, the system should focus on testing the tracking operations such as detect previous, current, and next halts of the buses…etc. to guarantee all these operations can work normally in real travelling environment.

5.1.4 Resource Planning

5.1.4.1 System Resource

No.	Resource	Description
1	Server	Need a database server which install MySQL Server & a web server which install Apache Server
2	Test tool	Develop a Test tool which can auto generate the test result to the pre-defined form and automated test execution
3	Network	An internet connection with the speed at least 9 Mbps
4	Computer	Run Windows 10, Ram 8 GB, CPU 2.9 GHz
5	Smartphone	Run Android (API level 23), Coordinates send & receive application

Table 3: System Resource

5.1.4.2 Human Resource

No.	Member	Task
1	Developer as a passenger	Execute all modules related to the passenger.
2	Developer as an administrator	Execute all modules related to the administrator.

Table 4: Human Resource

5.2 Test Cases

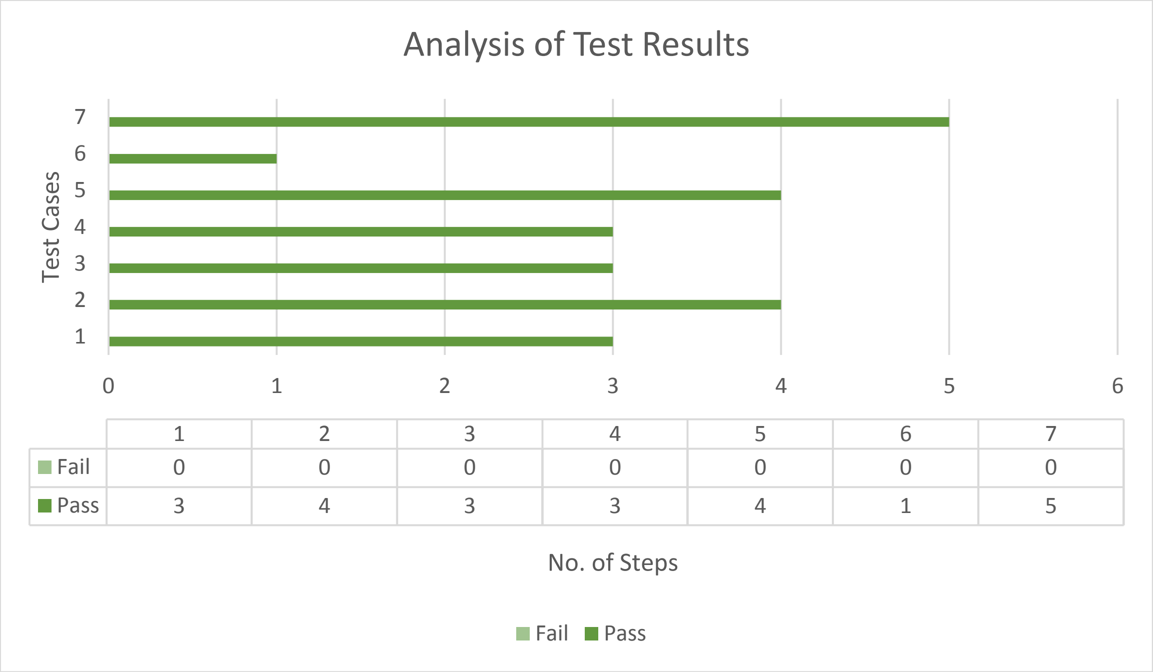
This section describes the main test cases related to the main modules of the easyBus system such as bus display, bus halt display, bus search engine, time schedule search engine and admin login with sample data and analysis of test results given by the main test cases.

Mainly, the following seven test cases was created for easyBus system.

1. EBTC 01 - Test the display of the previous halt of a bus.
2. EBTC 02 - Test the display of the current halt of a bus.
3. EBTC 03 - Test the display of the next halt of a bus.
4. EBTC 04 - Test the display of arriving/arrived buses of a halt.
5. EBTC 05 - Test the narest bus searching.
6. EBTC 06 - Test the time schedule searching.
7. EBTC 07 - Test the Login API for the admin.

Please refer to Appendix C: Test Cases to see the main detailed test cases.

Test results given by the main test cases can be analyze as Figure 17.



	1	2	3	4	5	6	7
Fail	0	0	0	0	0	0	0
Pass	3	4	3	3	4	1	5

Figure 17: Analysis of Test Results

According to Figure 17, all steps of the main test cases are successfully passed without any failure.

5.3 User Evaluation

This section describes how to do the user evaluation of the easyBus system by using questionnaires. Main user group of the system is the passanger group. Therefore, asking questions about the system from the passangers will help to evaluate the system. To do that, sample form of questioners will distribute among the passanger group. Please refer to Appendix D: User Evaluation to see the sample user evaluation form.

Chapter 6: Conclusion and Future Work

6.1 Conclusion

Passengers can prevent time-wasting due to the lack of knowledge on the arrivals/departures of the buses by the help of the bus time schedule search engine. They can easily find the arrivals/departures time of any bus according to their requirements.

Using the bus search engine, passengers can track any active bus in online for their requirement.

Display of a bus facilitate to identify the point at which to get down from the bus, accurately by showing previous, current, next halts of the bus. If a person missed the halt to get down, the person can easily recognize it and get down immediately at the next halt without going to a wrong direction by looking the previous halt. If bus is fully loaded passenger can be ready to get down by looking the next halt. Current halt always reminds get down now message.

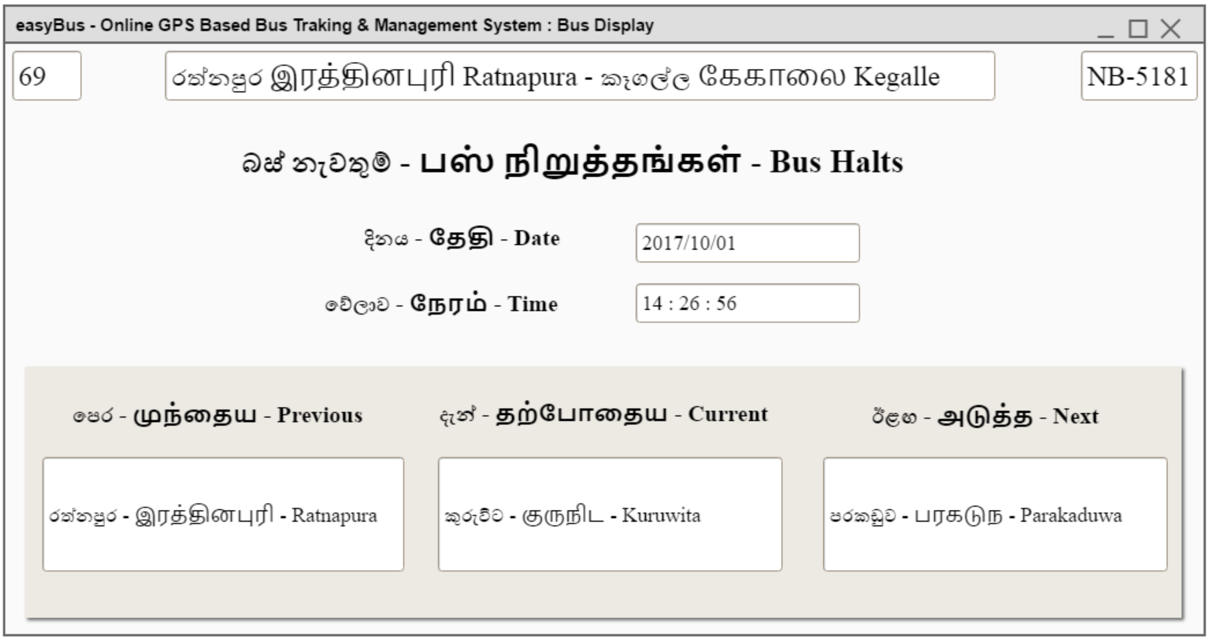
Display of a bus halt facilitate to identity accurately the bus which the passenger should get to reach the intended destination by showing all incoming buses to the halt.

All above facilities of the easyBus system up-lift the living standards of the passengers by preventing the common difficulties; getting delayed, wasting time on roads and missing the point at which they want to get down from or the bus they want to catch.

6.2 Future Work

The easyBus system is developed and tested for one route and one direction. In future, it can be extended to both directions based on the starting points of the buses and multiple routes by doing an adjustment of the halts and the route tables. Also features such as calculating the fare using a fare details table in the database, calculating the distance between two bus halts/stands, calculating the speed of a bus, finding the estimated time of the journey, finding the estimated arrival time of an approaching bus, viewing buses in google maps can be added to the system with the help of Google Maps APIs.

Appendix A: User Interface Design



easyBus - Online GPS Based Bus Traking & Management System : Bus Display

69

රත්නපුර இரத்தினபுரி Ratnapura - කෑගල්ල கேகாலை Kegalle

NB-5181

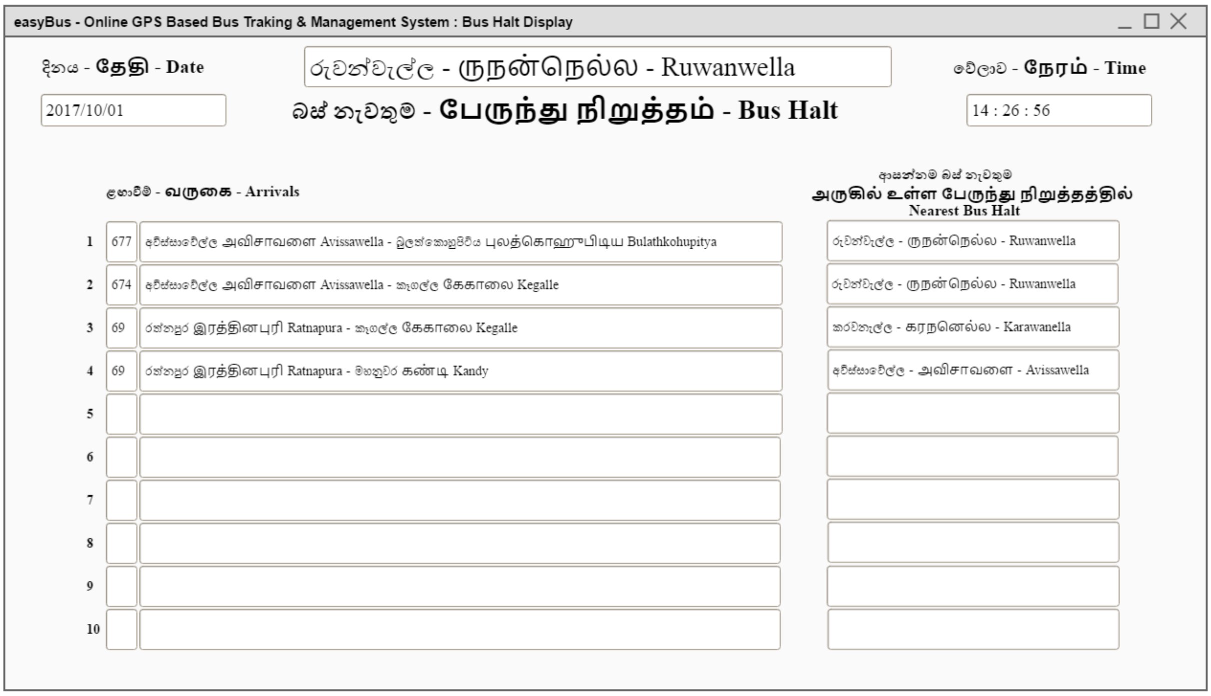
බස් නැවතුම් - **பஸ் நிறுத்தங்கள்** - **Bus Halts**

දිනය - **தேதி** - **Date** 2017/10/01

වේලාව - **நேரம்** - **Time** 14 : 26 : 56

පෙර - **முந்தைய** - **Previous**	දැන් - **தற்போதைய** - **Current**	ඊළඟ - **அடுத்த** - **Next**
රත්නපුර - இரத்தினபுரி - Ratnapura	කුරුවිට - குருநிட - Kuruwita	පරකඩුව - பரகடுந - Parakaduwa

Figure 18: Display API of the previous, current and next bus halts, for the buses



easyBus - Online GPS Based Bus Traking & Management System : Bus Halt Display

දිනය - **தேதி** - **Date** 2017/10/01

රුවන්වැල්ල - ருநன்நெல்ல - Ruwanwella

බස් නැවතුම - **பேருந்து நிறுத்தம்** - **Bus Halt**

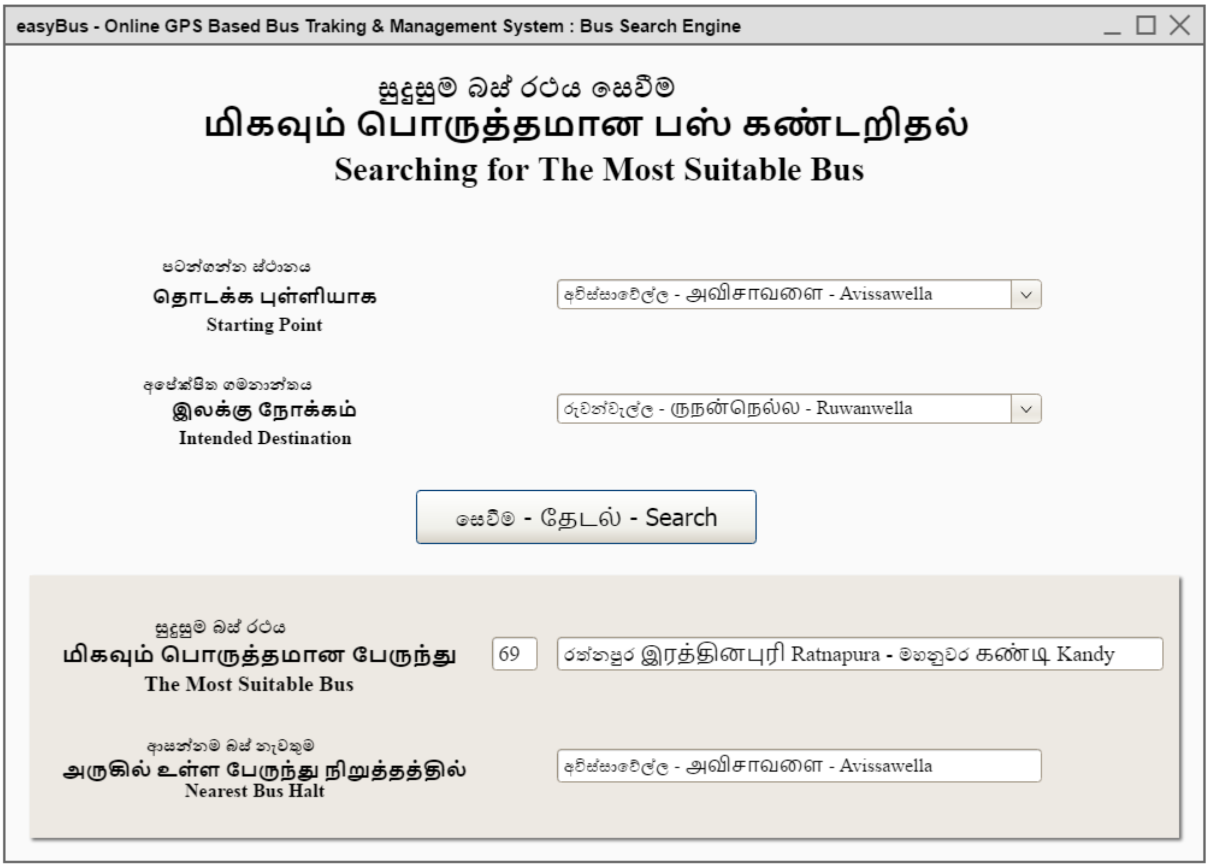
වේලාව - **நேரம்** - **Time** 14 : 26 : 56

ළඟාවීම් - **வருகை** - **Arrivals**

ආසන්නම බස් නැවතුම **அருகில் உள்ள பேருந்து நிறுத்தத்தில்** **Nearest Bus Halt**

1	677	අවිස්සාවේල්ල அவிசாவளை Avissawella - බුලත්කොහුපිටිය புலத்கொஹுபிடிய Bulathkohupitya	රුවන්වැල්ල - ருநன்நெல்ல - Ruwanwella
2	674	අවිස්සාවේල්ල அவிசாவளை Avissawella - කෑගල්ල கேகாலை Kegalle	රුවන්වැල්ල - ருநன்நெல்ல - Ruwanwella
3	69	රත්නපුර இரத்தினபுரி Ratnapura - කෑගල්ල கேகாலை Kegalle	කරවනැල්ල - கரநனெல்ல - Karawanella
4	69	රත්නපුර இரத்தினபுரி Ratnapura - මහනුවර கண்டி Kandy	අවිස්සාවේල්ල - அவிசாவளை - Avissawella
5			
6			
7			
8			
9			
10			

Figure 19: Display API of the bus routes and the next bus halt locations of the arriving buses for the bus halts



easyBus - Online GPS Based Bus Traking & Management System : Bus Search Engine

සුදුසුම බස් රථය සෙවීම
மிகவும் பொருத்தமான பஸ் கண்டறிதல்
Searching for The Most Suitable Bus

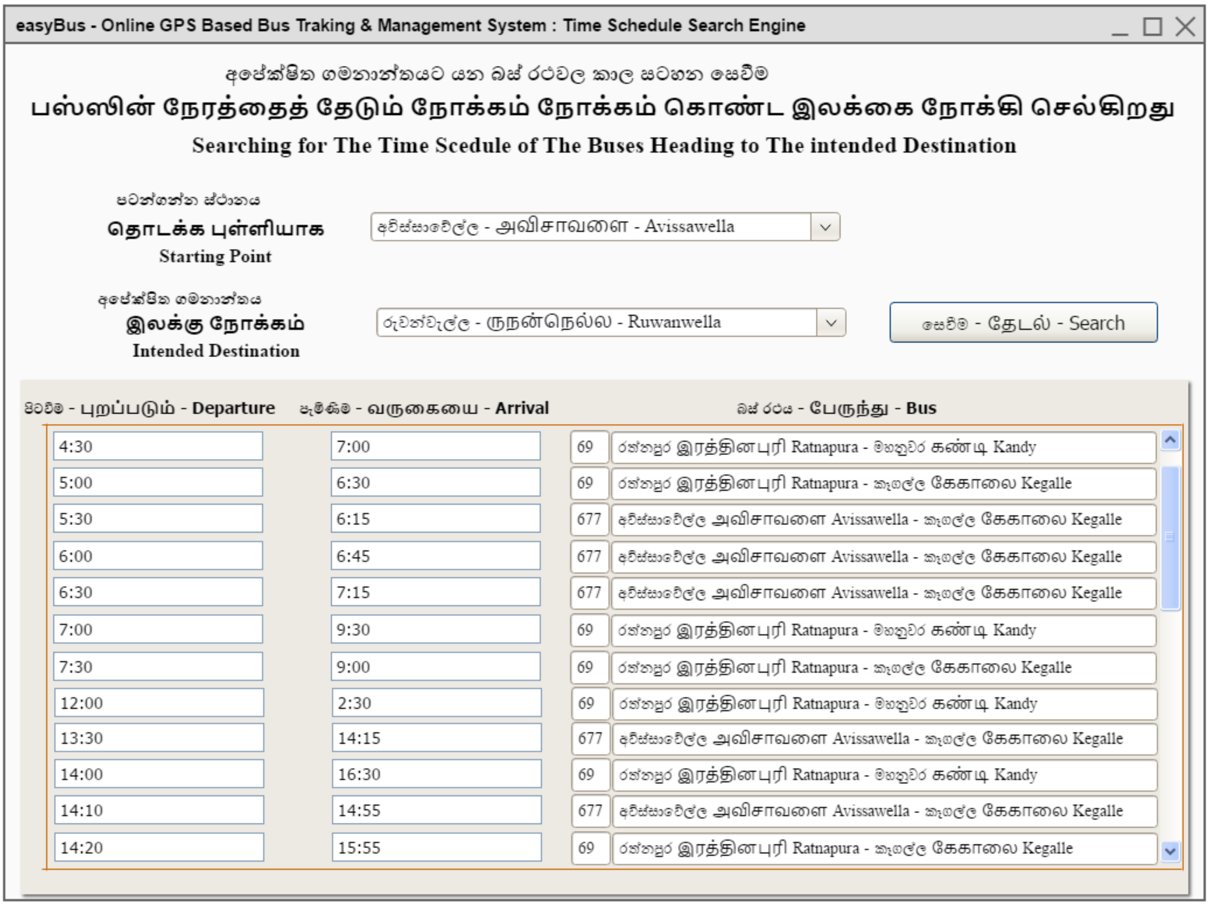
පටන්ගන්න ස්ථානය
தொடக்க புள்ளியாக
Starting Point: අවිස්සාවේල්ල - அவிசாவளை - Avissawella

අපේක්ෂිත ගමනාන්තය
இலக்கு நோக்கம்
Intended Destination: රුවන්වැල්ල - ருநன்நெல்ல - Ruwanwella

සුදුසුම බස් රථය
மிகவும் பொருத்தமான பேருந்து
The Most Suitable Bus: 69 | රත්නපුර இரத்தினபுரி Ratnapura - මහනුවර கண்டி Kandy

ආසන්නම බස් නැවතුම
அருகில் உள்ள பேருந்து நிறுத்தத்தில்
Nearest Bus Halt: අවිස්සාවේල්ල - அவிசாவளை - Avissawella

Figure 20: Online searching API of the next active bus to the intended destination for the passengers



easyBus - Online GPS Based Bus Traking & Management System : Time Schedule Search Engine

අපේක්ෂිත ගමනාන්තයට යන බස් රථවල කාල සටහන සෙවීම

பஸ்ஸின் நேரத்தைத் தேடும் நோக்கம் நோக்கம் கொண்ட இலக்கை நோக்கி செல்கிறது

Searching for The Time Scedule of The Buses Heading to The intended Destination

පටන්ගන්න ස්ථානය
தொடக்க புள்ளியாக
Starting Point: අවිස්සාවේල්ල - அவிசாவளை - Avissawella

අපේක්ෂිත ගමනාන්තය
இலக்கு நோக்கம்
Intended Destination: රුවන්වැල්ල - ருநன்நெல்ல - Ruwanwella

පිටවීම - புறப்படும் - Departure	පැමිණීම - வருகையை - Arrival	බස් රථය - பேருந்து - Bus	
4:30	7:00	69	රත්නපුර இரத்தினபுரி Ratnapura - මහනුවර கண்டி Kandy
5:00	6:30	69	රත්නපුර இரத்தினபுரி Ratnapura - කෑගල්ල கேகாலை Kegalle
5:30	6:15	677	අවිස්සාවේල්ල அவிசாவளை Avissawella - කෑගල්ල கேகாலை Kegalle
6:00	6:45	677	අවිස්සාවේල්ල அவிசாவளை Avissawella - කෑගල්ල கேகாலை Kegalle
6:30	7:15	677	අවිස්සාවේල්ල அவிசாவளை Avissawella - කෑගල්ල கேகாலை Kegalle
7:00	9:30	69	රත්නපුර இரத்தினபுரி Ratnapura - මහනුවර கண்டி Kandy
7:30	9:00	69	රත්නපුර இரத்தினபுரி Ratnapura - කෑගල්ල கேகாலை Kegalle
12:00	2:30	69	රත්නපුර இரத்தினபுரி Ratnapura - මහනුවර கண்டி Kandy
13:30	14:15	677	අවිස්සාවේල්ල அவிசாவளை Avissawella - කෑගල්ල கேகாலை Kegalle
14:00	16:30	69	රත්නපුර இரத்தினபுரி Ratnapura - මහනුවර கண்டி Kandy
14:10	14:55	677	අවිස්සාවේල්ල அவிசாவளை Avissawella - කෑගල්ල கேகாலை Kegalle
14:20	15:55	69	රත්නපුර இரத்தினபுரி Ratnapura - කෑගල්ල கேகாலை Kegalle

Figure 21: Online searching API of the arrival/departure times of all the buses going to/through the intended destination for the passengers

Appendix B: Scope of Testing

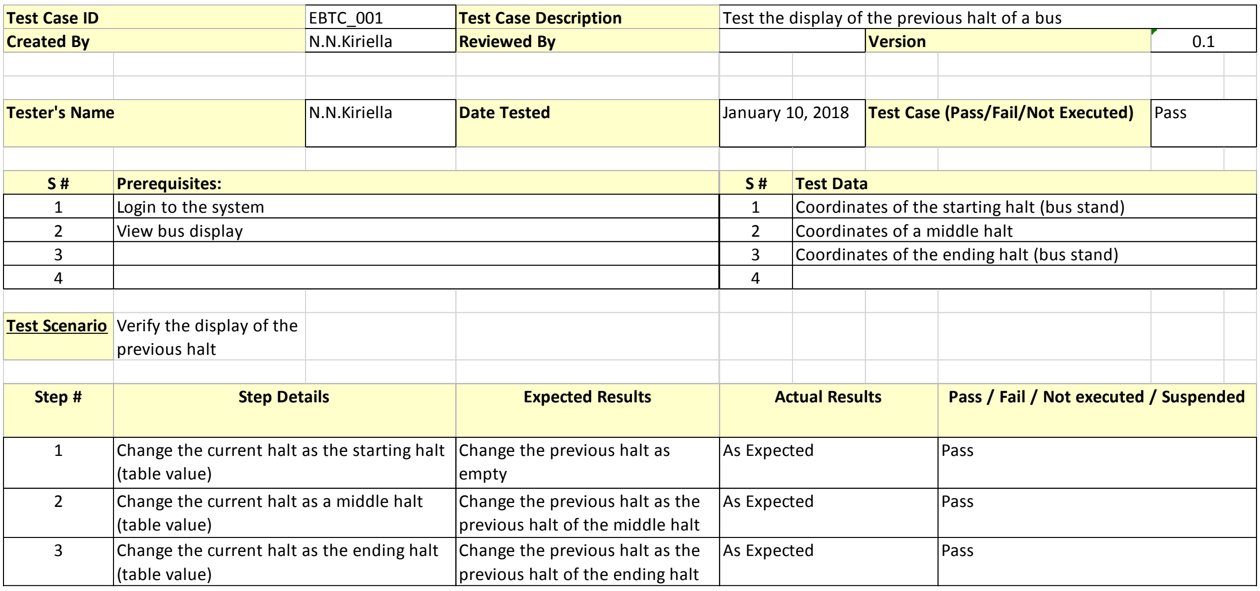
Module Name	Applicable Roles	Description
Display previous halt	Passenger	While being inside the bus, a passenger can, in real time, trace the previous bus halt at which his bus stopped.
Display current halt	Passenger	While being inside the bus, a passenger can, in real time, trace the current bus halt of his bus.
Display next halt	Passenger	While being inside the bus, a passenger can, in real time, trace the next bus halt of his bus.
Display the details of arriving/arrived busses	Passenger	A passenger can see the details of all the arriving buses and arrived busses to the bus halt in real-time according to the arriving order.
Search the nearest bus	Passenger	A passenger can search the nearest bus which goes to the intended destination accurately.
Search the time schedule	Passenger	A passenger can search the time schedule of all the busses which go to the intended destination.
Login	Administrator	An administrator can login to the system with username and password.
Add account	Administrator	An administrator can add a new admin account into the system.
Modify account	Administrator	An administrator can modify the details of an existing admin account in the system.
Delete account	Administrator	An administrator can delete an existing admin account in the system.
Search account	Administrator	An administrator can search for an existing admin account in the system.
View account	Administrator	An administrator can view all existing admin accounts in the system.

Add bus	Administrator	An administrator can add the details of a bus into the system.
Add halt	Administrator	An administrator can add the details of a bus halt into the system.
Add route	Administrator	An administrator can add the details of a bus route into the system.
Add time schedule	Administrator	An administrator can add the time schedule of a bus into the system.
Modify bus	Administrator	An administrator can modify the details of a bus in the system.
Modify halt	Administrator	An administrator can modify the details of a bus halt in the system.
Modify route	Administrator	An administrator can modify the details of a bus route in the system.
Modify time schedule	Administrator	An administrator can modify the time schedule of a bus in the system.
Delete bus	Administrator	An administrator can delete the details of a bus in the system.
Delete halt	Administrator	An administrator can delete the details of a bus halt in the system.
Delete route	Administrator	An administrator can delete the details of a bus route in the system.
Delete time schedule	Administrator	An administrator can delete the time schedule of a bus in the system.
Search bus	Administrator	An administrator can search for the details of a bus in the system.
Search halt	Administrator	An administrator can search for the details of a bus halt in the system.
Search route	Administrator	An administrator can search for the details of a route in the system.
View buses	Administrator	An administrator can view all the details of all busses in the system.
View halts	Administrator	An administrator can view all the details of all bus halts in the system.

View routes	Administrator	An administrator can view all the details of all bus routes in the system.
View bus display	Administrator	An administrator can view the display of a bus.
View bus halt display	Administrator	An administrator can view the display of a bus halt.

Table 5: Scope of Testing

Appendix C: Test Cases



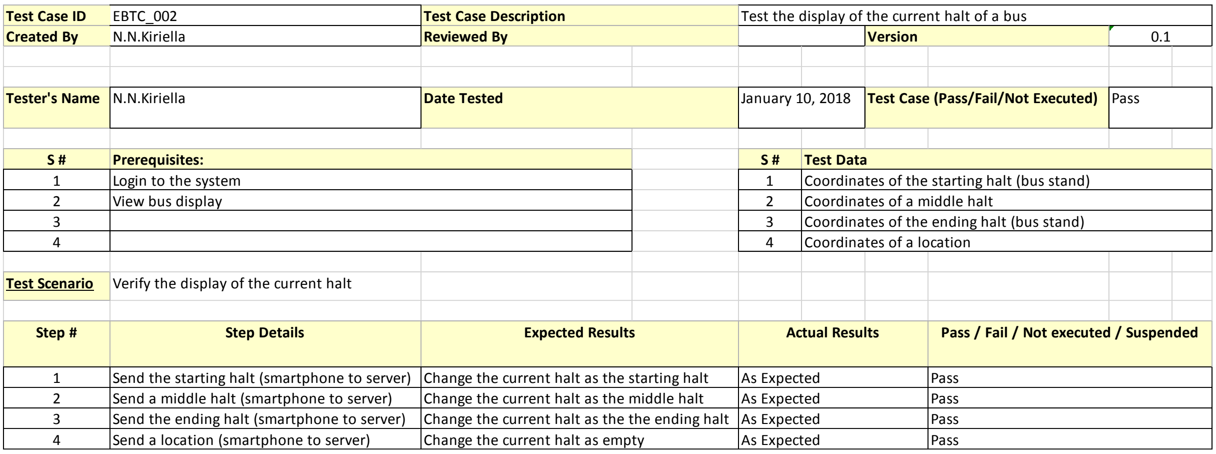
Test Case ID	EBTC_001	**Test Case Description**	Test the display of the previous halt of a bus		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case (Pass/Fail/Not Executed)**	Pass

S #	**Prerequisites:**	**S #**	**Test Data**
1	Login to the system	1	Coordinates of the starting halt (bus stand)
2	View bus display	2	Coordinates of a middle halt
3		3	Coordinates of the ending halt (bus stand)
4		4	

Test Scenario	Verify the display of the previous halt

Step #	**Step Details**	**Expected Results**	**Actual Results**	**Pass / Fail / Not executed / Suspended**
1	Change the current halt as the starting halt (table value)	Change the previous halt as empty	As Expected	Pass
2	Change the current halt as a middle halt (table value)	Change the previous halt as the previous halt of the middle halt	As Expected	Pass
3	Change the current halt as the ending halt (table value)	Change the previous halt as the previous halt of the ending halt	As Expected	Pass

Figure 22: Test Case 01



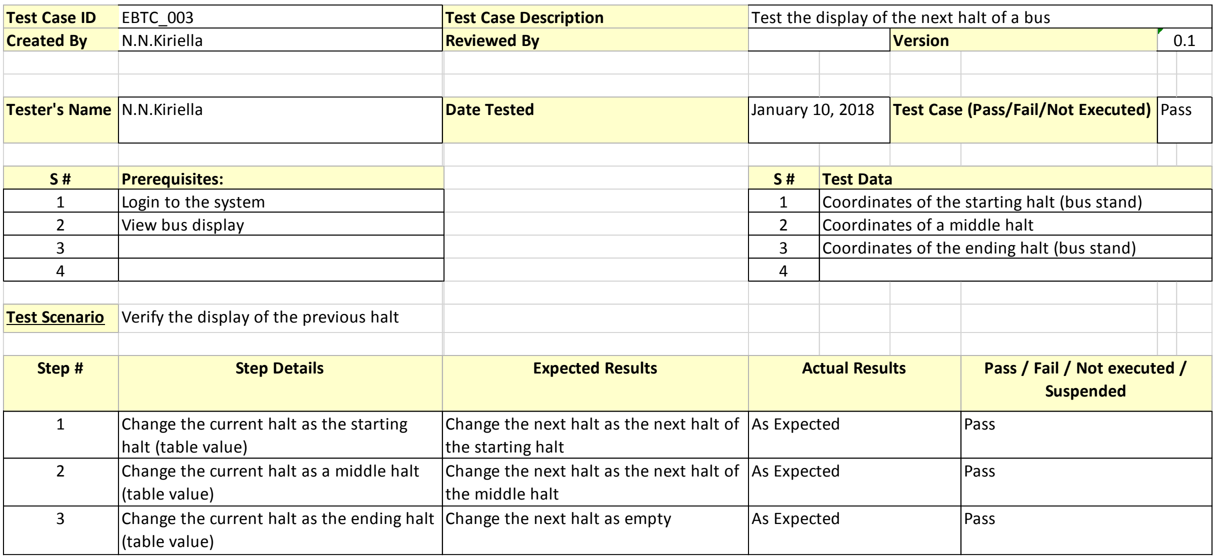
Test Case ID	EBTC_002	**Test Case Description**	Test the display of the current halt of a bus		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case (Pass/Fail/Not Executed)**	Pass

S #	**Prerequisites:**	**S #**	**Test Data**
1	Login to the system	1	Coordinates of the starting halt (bus stand)
2	View bus display	2	Coordinates of a middle halt
3		3	Coordinates of the ending halt (bus stand)
4		4	Coordinates of a location

Test Scenario	Verify the display of the current halt

Step #	**Step Details**	**Expected Results**	**Actual Results**	**Pass / Fail / Not executed / Suspended**
1	Send the starting halt (smartphone to server)	Change the current halt as the starting halt	As Expected	Pass
2	Send a middle halt (smartphone to server)	Change the current halt as the middle halt	As Expected	Pass
3	Send the ending halt (smartphone to server)	Change the current halt as the the ending halt	As Expected	Pass
4	Send a location (smartphone to server)	Change the current halt as empty	As Expected	Pass

Figure 23: Test Case 02



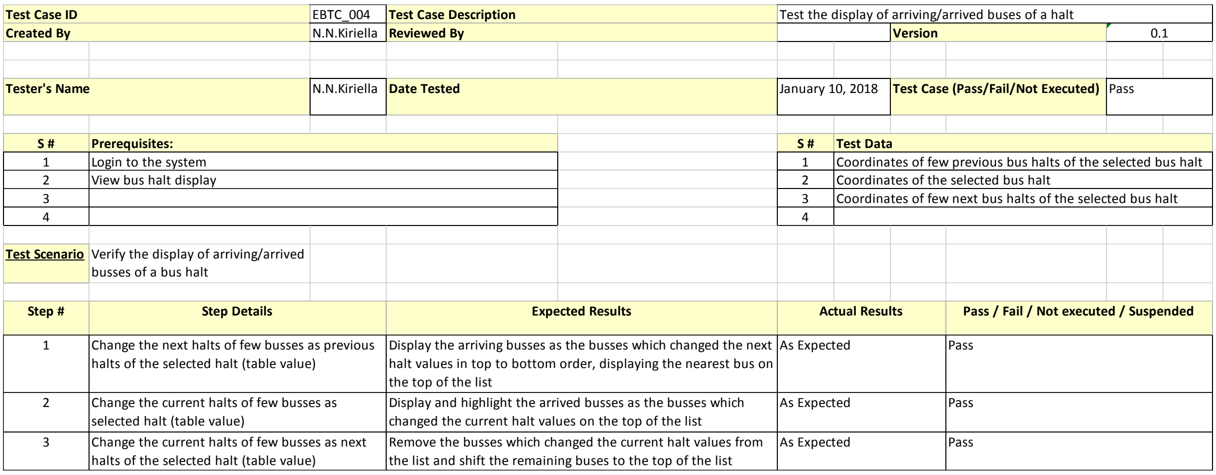
Test Case ID	EBTC_003	**Test Case Description**	Test the display of the next halt of a bus		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case (Pass/Fail/Not Executed)**	Pass

S #	Prerequisites:	S #	Test Data
1	Login to the system	1	Coordinates of the starting halt (bus stand)
2	View bus display	2	Coordinates of a middle halt
3		3	Coordinates of the ending halt (bus stand)
4		4	

Test Scenario Verify the display of the previous halt

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Change the current halt as the starting halt (table value)	Change the next halt as the next halt of the starting halt	As Expected	Pass
2	Change the current halt as a middle halt (table value)	Change the next halt as the next halt of the middle halt	As Expected	Pass
3	Change the current halt as the ending halt (table value)	Change the next halt as empty	As Expected	Pass

Figure 24: Test Case 03



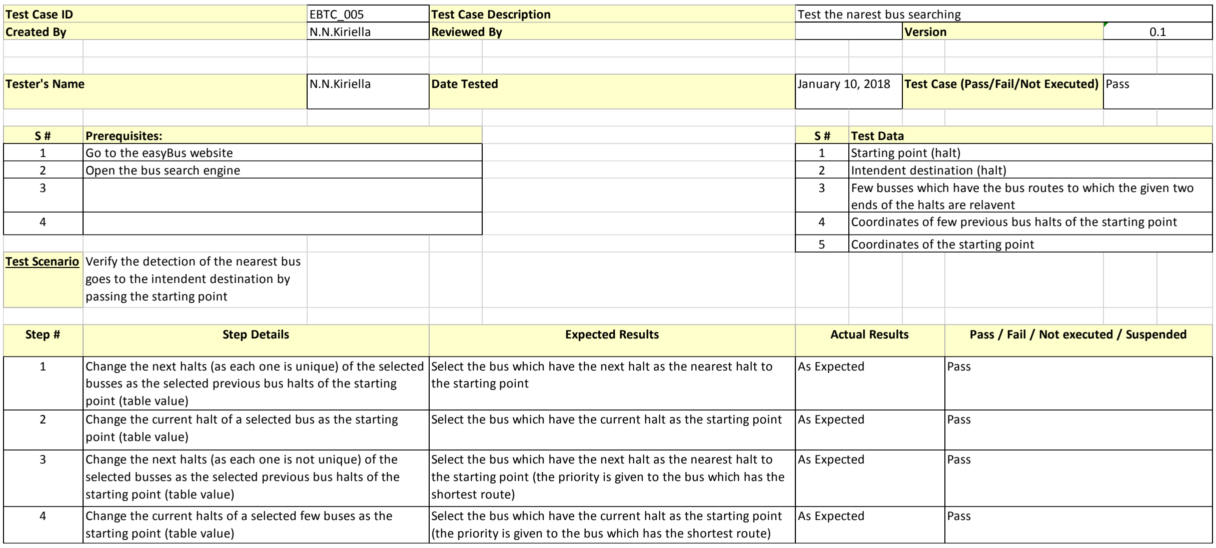
Test Case ID	EBTC_004	**Test Case Description**	Test the display of arriving/arrived buses of a halt		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case (Pass/Fail/Not Executed)**	Pass

S #	Prerequisites:	S #	Test Data
1	Login to the system	1	Coordinates of few previous bus halts of the selected bus halt
2	View bus halt display	2	Coordinates of the selected bus halt
3		3	Coordinates of few next bus halts of the selected bus halt
4		4	

Test Scenario Verify the display of arriving/arrived busses of a bus halt

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Change the next halts of few busses as previous halts of the selected halt (table value)	Display the arriving busses as the busses which changed the next halt values in top to bottom order, displaying the nearest bus on the top of the list	As Expected	Pass
2	Change the current halts of few busses as selected halt (table value)	Display and highlight the arrived busses as the busses which changed the current halt values on the top of the list	As Expected	Pass
3	Change the current halts of few busses as next halts of the selected halt (table value)	Remove the busses which changed the current halt values from the list and shift the remaining buses to the top of the list	As Expected	Pass

Figure 25: Test Case 04



Test Case ID	EBTC_005	**Test Case Description**	Test the narest bus searching		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case (Pass/Fail/Not Executed)**	Pass

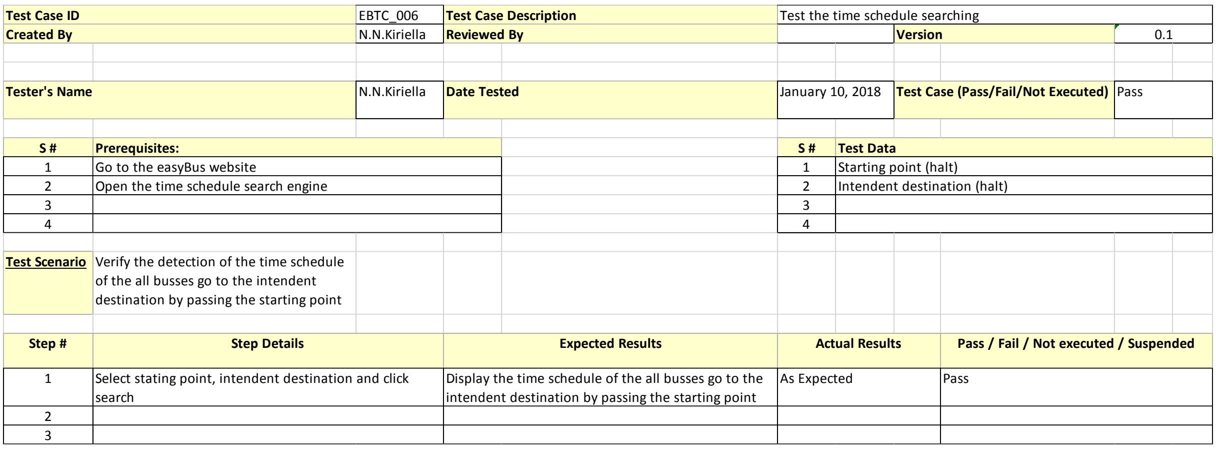
S #	Prerequisites:
1	Go to the easyBus website
2	Open the bus search engine
3	
4	

S #	Test Data
1	Starting point (halt)
2	Intendent destination (halt)
3	Few busses which have the bus routes to which the given two ends of the halts are relavent
4	Coordinates of few previous bus halts of the starting point
5	Coordinates of the starting point

Test Scenario	Verify the detection of the nearest bus goes to the intendent destination by passing the starting point

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Change the next halts (as each one is unique) of the selected busses as the selected previous bus halts of the starting point (table value)	Select the bus which have the next halt as the nearest halt to the starting point	As Expected	Pass
2	Change the current halt of a selected bus as the starting point (table value)	Select the bus which have the current halt as the starting point	As Expected	Pass
3	Change the next halts (as each one is not unique) of the selected busses as the selected previous bus halts of the starting point (table value)	Select the bus which have the next halt as the nearest halt to the starting point (the priority is given to the bus which has the shortest route)	As Expected	Pass
4	Change the current halts of a selected few buses as the starting point (table value)	Select the bus which have the current halt as the starting point (the priority is given to the bus which has the shortest route)	As Expected	Pass

Figure 26: Test Case 05



Test Case ID	EBTC_006	**Test Case Description**	Test the time schedule searching		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case (Pass/Fail/Not Executed)**	Pass

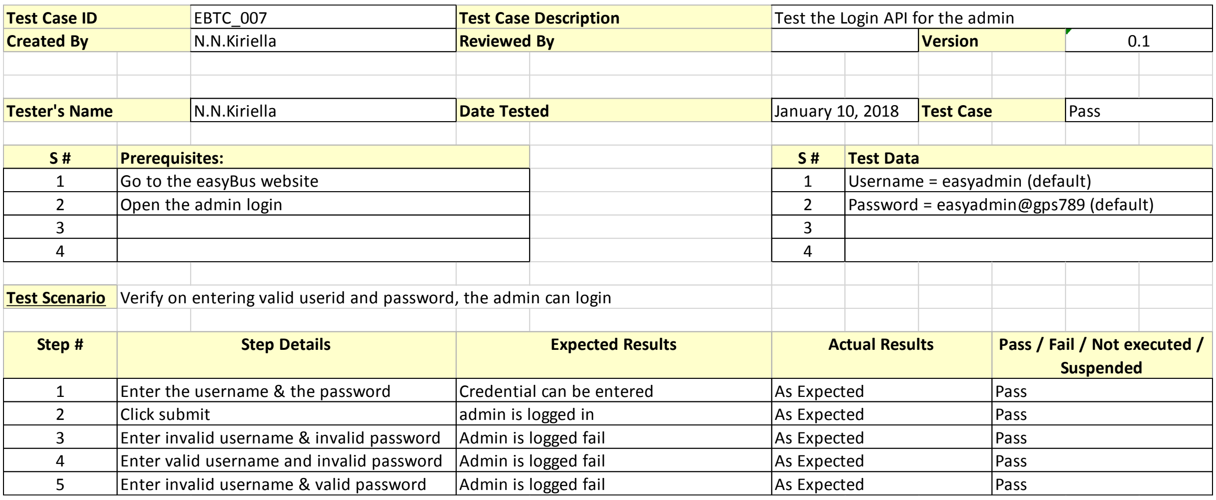
S #	Prerequisites:
1	Go to the easyBus website
2	Open the time schedule search engine
3	
4	

S #	Test Data
1	Starting point (halt)
2	Intendent destination (halt)
3	
4	

Test Scenario	Verify the detection of the time schedule of the all busses go to the intendent destination by passing the starting point

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Select stating point, intendent destination and click search	Display the time schedule of the all busses go to the intendent destination by passing the starting point	As Expected	Pass
2				
3				

Figure 27: Test Case 06



Test Case ID	EBTC_007	**Test Case Description**	Test the Login API for the admin		
Created By	N.N.Kiriella	**Reviewed By**		**Version**	0.1
Tester's Name	N.N.Kiriella	**Date Tested**	January 10, 2018	**Test Case**	Pass

S #	Prerequisites:
1	Go to the easyBus website
2	Open the admin login
3	
4	

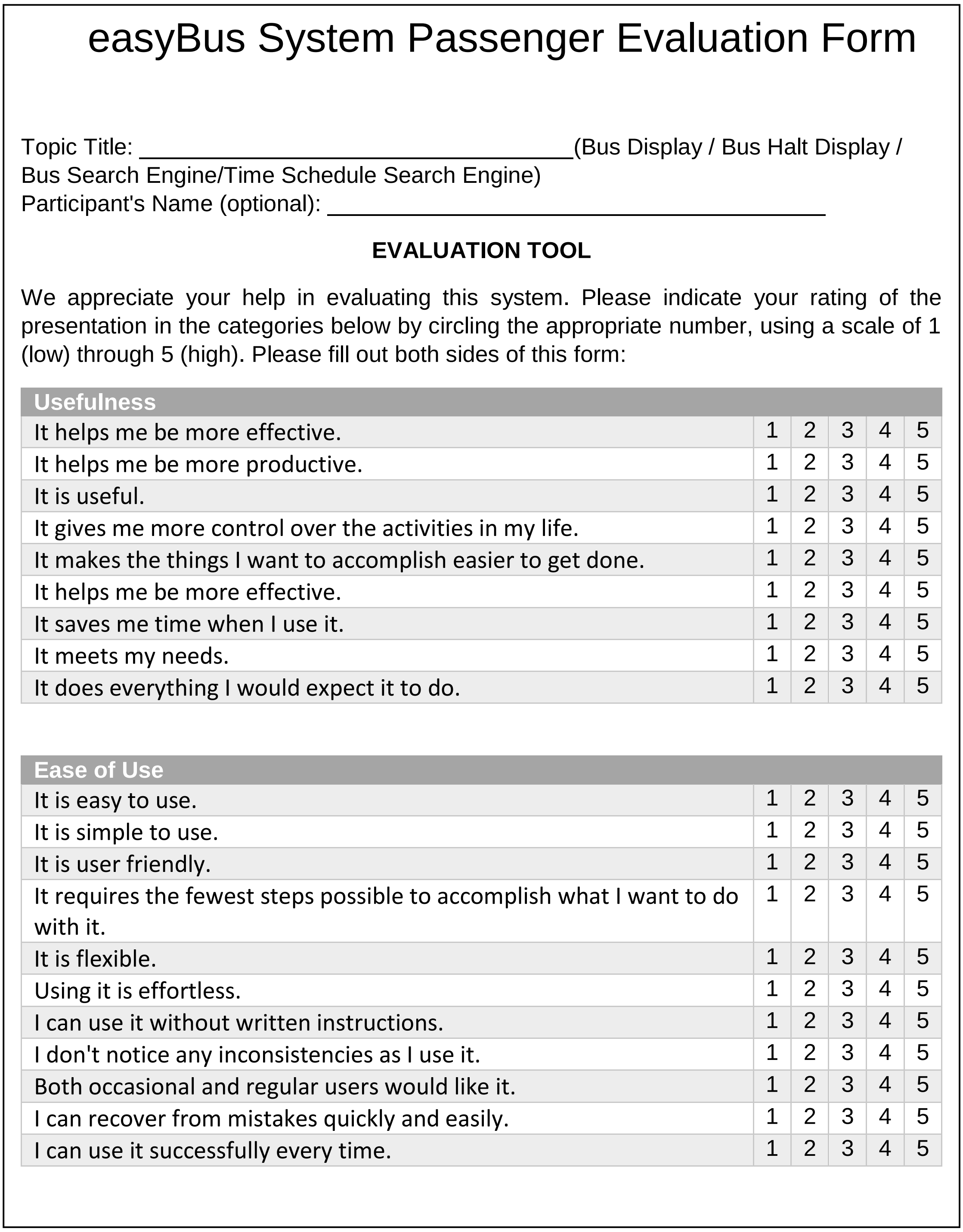
S #	Test Data
1	Username = easyadmin (default)
2	Password = easyadmin@gps789 (default)
3	
4	

Test Scenario Verify on entering valid userid and password, the admin can login

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Enter the username & the password	Credential can be entered	As Expected	Pass
2	Click submit	admin is logged in	As Expected	Pass
3	Enter invalid username & invalid password	Admin is logged fail	As Expected	Pass
4	Enter valid username and invalid password	Admin is logged fail	As Expected	Pass
5	Enter invalid username & valid password	Admin is logged fail	As Expected	Pass

Figure 28: Test Case 07

Appendix D: User Evaluation



easyBus System Passenger Evaluation Form

Topic Title: ___________________________________(Bus Display / Bus Halt Display / Bus Search Engine/Time Schedule Search Engine)
Participant's Name (optional): ____

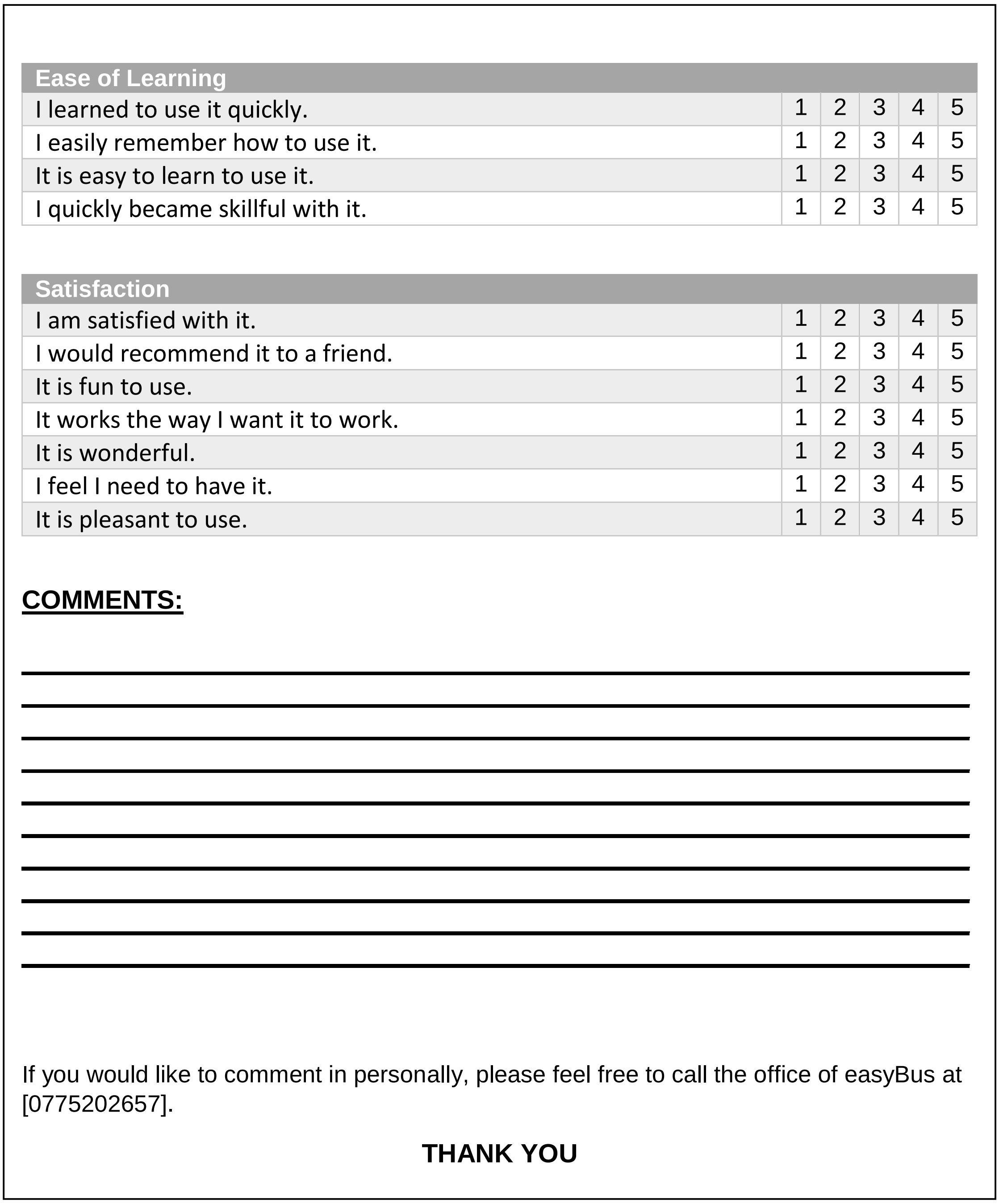
EVALUATION TOOL

We appreciate your help in evaluating this system. Please indicate your rating of the presentation in the categories below by circling the appropriate number, using a scale of 1 (low) through 5 (high). Please fill out both sides of this form:

Usefulness					
It helps me be more effective.	1	2	3	4	5
It helps me be more productive.	1	2	3	4	5
It is useful.	1	2	3	4	5
It gives me more control over the activities in my life.	1	2	3	4	5
It makes the things I want to accomplish easier to get done.	1	2	3	4	5
It helps me be more effective.	1	2	3	4	5
It saves me time when I use it.	1	2	3	4	5
It meets my needs.	1	2	3	4	5
It does everything I would expect it to do.	1	2	3	4	5

Ease of Use					
It is easy to use.	1	2	3	4	5
It is simple to use.	1	2	3	4	5
It is user friendly.	1	2	3	4	5
It requires the fewest steps possible to accomplish what I want to do with it.	1	2	3	4	5
It is flexible.	1	2	3	4	5
Using it is effortless.	1	2	3	4	5
I can use it without written instructions.	1	2	3	4	5
I don't notice any inconsistencies as I use it.	1	2	3	4	5
Both occasional and regular users would like it.	1	2	3	4	5
I can recover from mistakes quickly and easily.	1	2	3	4	5
I can use it successfully every time.	1	2	3	4	5

Figure 29: First Part of the Passenger Evaluation Form



Ease of Learning					
I learned to use it quickly.	1	2	3	4	5
I easily remember how to use it.	1	2	3	4	5
It is easy to learn to use it.	1	2	3	4	5
I quickly became skillful with it.	1	2	3	4	5

Satisfaction					
I am satisfied with it.	1	2	3	4	5
I would recommend it to a friend.	1	2	3	4	5
It is fun to use.	1	2	3	4	5
It works the way I want it to work.	1	2	3	4	5
It is wonderful.	1	2	3	4	5
I feel I need to have it.	1	2	3	4	5
It is pleasant to use.	1	2	3	4	5

COMMENTS:

If you would like to comment in personally, please feel free to call the office of easyBus at [0775202657].

THANK YOU

Figure 30: Last Part of the Passenger Evaluation Form

References

Online Sources

[1] Ministry of Transport & Civil Aviation. Sri Lanka Transport Board. [Online]. Available: http://www.transport.gov.lk/web/index.php/component/content/article/175.html.

[2] National Transport Commission. [Online]. Available: https://www.ntc.gov.lk/.

[3] Mobitel. Tracking Solutions. [Online]. Available: http://www.mobitel.lk/tracking-solutions.

[4] Dialog. iLocate – The low cost Vehicle and Team tracking solution. [Online]. Available: https://www.dialog.lk/blog/ilocate-the-low-cost-vehicle-and-team-tracking-solution/.

[5] Nexify Systems. [Online]. Available: https://nexify.lk/.

[6] Track My Car. Solutions. [Online]. Available: https://trackmycar.lk/tracking-solutions/.

[7] PickMe. [Online]. Available: https://pickme.lk/.

[8] Kangaroo Cabs. [Online]. Available: http://www.2588588.com/.

[9] Uber. [Online]. Available: https://www.uber.com/en-LK/.

[10] BusTicket.LK. [Online]. Available: https://busticket.lk/.