LUMS Campus Navigation System

Software Requirements Specification

Version 1.0

28th February, 2013

Group 3
Aroosha Ahmad
Hafiz Salman Asif
Muhammad Junaid Khalid
Muhammad Nauman Minhas

Prepared for
CS360—Software Engineering
Instructor: Dr. Hamid Abdul Basit
Spring 2013

Revision History

Date	Description	Author	Comments
28th Feb '13	Version 1.0	Aroosha Ahmad	First Revision

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Group3	Lead Software Eng.	
	Dr. Hamid Abdul Basit	Instructor, CS360	

Table of Contents

1. Introduction

1.1 Problem Statement

In universities with large campuses, one is not aware of the location of every office and auditorium on campus. One can simply not afford to get lost in the LUMS academic block before an 8AM examination. Similarly, visiting every wing of the academic block to find the correct office is extremely tiresome. At moments like these, it would be extremely convenient to have an automated map on your mobile, which would instantly give you the shortest path to your desired location on campus.

1.2 Purpose

This SRS describes the requirements and specifications of LUMS Campus Navigation System. It explains the functional features of the navigation system, along with interface details, design constraints, special technologies used and related considerations such as reliability and performance characteristics. The SRS is intended for LUMS students, staff and guests.

This document is also intended for prospective software developers, who are interested in extending this project to campuses of other universities.

1.3 Scope

The LUMS Campus Navigation System is intended to provide the users with a smart map of LUMS; much like the Google maps system. The important characteristic of this system is that it works indoor as well, unlike other GPS/navigation systems.

The basic functionality of the system is simple; given your current location on the campus, it will give you the shortest route to your desired location on campus. We will develop the system as a mobile application, for easy, fast and on-the-spot use. Now, the students/staff will not have to worry about finding a particular room/office on campus and save themselves the time and the hassle.

One of the biggest advantages of this application is for the guests at LUMS. LUMS hosts around ten events every year which involve the participation from over one thousand delegates per event, from all-over Pakistan. These participants are entirely new to the campus and hence, can use this application to find their way around.

This system and document also aims at helping other software developers who are interested in providing navigation systems for other universities or buildings.

1.4 Definitions, Acronyms, and Abbreviations

SRS: Software Requirements Specification

Wi-Fi: Wireless Fidelity

CPU: Central Processing Unit

Java ME: Java Micro Edition

SQL: Structural Query Language

GPS: Global Positioning System

SDK: System Development Kit

ARP: Address Resolution Protocol (ARP) is a telecommunications protocol used for resolution of network layer addresses into link layer addresses.

MAC Address: A Media Access Control address (MAC address) is a unique identifier assigned to network interfaces for communications on the physical network segment.

Routers: Routers are small physical devices that join multiple networks together.

Android: Android is an operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers.

Encryption: Encryption is the process of encoding information in such a way that eavesdroppers or hackers cannot read it, but only authorized parties can.

API: Application Programming Interface

Block Level Location: This means the location of the building e.g. Academic Block or SBASSE/SSE building.

Place Location: The location of the indoor places e.g. the Discussion-Rooms (DRs) or auditoriums.

1.5 References

http://en.wikipedia.org/wiki/Main_Page
https://maps.google.com/

1.6 Overview

The rest of the SRS examines the specifications of the LUMS Campus Navigation System. Section 2 of the SRS presents the general factors that affect the LUMS Campus Navigator and its requirements, such as user characteristics and project constraints. Section 3 outlines the detailed, specific functional, performance, system and other related requirements of the LUMS Campus Navigator. Supporting information about appendices is provided in Section 3.

2. Overall Description

2.1 Product Perspective

This project has two main parts. In the first part of the project, a mobile phone will be used to run the map application and reveal the changes in the user interface according to the information coming from the server. The connection between the server and the mobile phone will be provided via wireless (Wi-Fi).

The second part involves a server. We will be using the MAC addresses of the wire-less routers in LUMS, in our processing. MAC addresses of routers are considered as private and sensitive information; therefore, we cannot afford to make this information available to the users. In order to do this, we have to do all the processing on the server and keep this sensitive information encrypted on the server-side.

Secondly, mobile devices do not have a powerful CPU; they cannot handle complicated image manipulating processes and calculations. Therefore, a server computer is needed to process the information coming from mobile devices; as the CPU of a computer is powerful enough. Hence, the second part of the project will be developed on a computer which will be used as a server doing some image processing operations and calculations. It will have a database containing information about the wireless network of LUMS i.e. the routers.

2.2 Product Functions

As the name of the application implies, the main purpose of the system is to provide On-Campus navigation in LUMS. The basic functionality of the system can be summarized in three parts.

It provides you with a complete and detailed map of LUMS with the added detail of the main LUMS Academic block. Firstly, the application tracks your current location on campus and shows it on the map so that you are never lost on campus. This is the default setting of this application.

The second functionality is that if you specify your destination location to the navigator, it shows you the shortest possible route from your current location to your destination.

The third functionality is that you manually enter a location in LUMS, and the application will show you it's location on campus. You can also specify two locations, e.g. A and B, and it will give you the shortest route between the two.

2.3 User Characteristics

The users of the LUMS Campus Navigator are the general public. We can safely assume that they have little or no technical knowledge; therefore, the application should be as user friendly as possible and accessible to maximum number of people interacting with LUMS. In order to achieve this, we have decided to develop a mobile application, since; almost everyone has a mobile these days.

2.4 General Constraints

The current constraints on the project mainly revolve around the software to be used. We are using the Google Maps API for the outdoor locations and major buildings in LUMS. Some of the locations are not tagged as yet and thus, we have to customize it according to our needs.

Secondly, we are using the Google Maps as our images with the incorporation of image of the academic block; therefore, we have to customize it according to our needs. As we are planning to provide navigation inside the LUMS academic block; we have to extend and customize the Google Maps API according to our needs.

Another issue to be resolved is that GPS is not functional indoors. Therefore, we have to develop a system for indoor navigation that can at least tell the floor the user is on but the precision of the output will be largely dependent upon the signals of the Wi-Fi routers (that can be tempered, weakened or spoofed easily). Therefore, in cases where the output will not be reliable we will ask the user to enter the floor number manually.

We have decided upon developing an Android application, since, it is the technology of today's world. However, this is the first time we are working with this technology; therefore, it will take us time to get comfortable with it.

Another issue that we have to tackle is that the indoor maps of the Academic Block of LUMS are outdated, which means, that we have to update them ourselves, by making rounds of the Academic Block and marking the changes in the map especially the change in location of the offices.
Also, for the academic block, we will have to take updated map and find a way to convert it into a graph with relevant nodes and paths.

Lastly, as we are developing this project as part of a course, we are greatly constrained by time. Therefore, we are only providing indoor navigation for the LUMS academic block and not the other buildings e.g. SSE (School of Sciences and Engineering) and SDSB (Suleman Dawood

School of Business). However, we will be providing complete navigation for the entire campus in terms of outdoor locations.

2.5 Assumptions and Dependencies

A number of factors that may affect the requirements specified in the SRS include:

- The users have smartphones with android operating system, 2.3 gingerbread or above.
- Wi-Fi is available on the 'entire' campus. Indoors as well as outdoor.
- GPS is enabled on their phone and it is working.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Tapping on the icon of the application will open the application. A LUMS map will appear on the screen, with a menu on the left side indicating the different options that the navigator provides. Tapping on a location twice will give you the route from your current location to that location that you tapped.

3.1.2 Hardware Interfaces

Hardware interfaces include Android Smartphones (with touch screens), Wi-Fi transceiver, Wi-Fi routers and a computer for the server-side.

3.1.3 Software Interfaces

We will be using the following software:

- Google Maps API.
- Android API.
- Java SDK 7.1
- MySQL database.

We will be basing our project on the client-server model where computations and sensitive information will be retained on the server-side.

3.1.4 Communications Interfaces

- Internet
- Wi-Fi

3.2 Functional Requirements/Use-Cases

The functional requirements and specifications of the LUMS Campus Navigator are represented in terms of the following use cases.

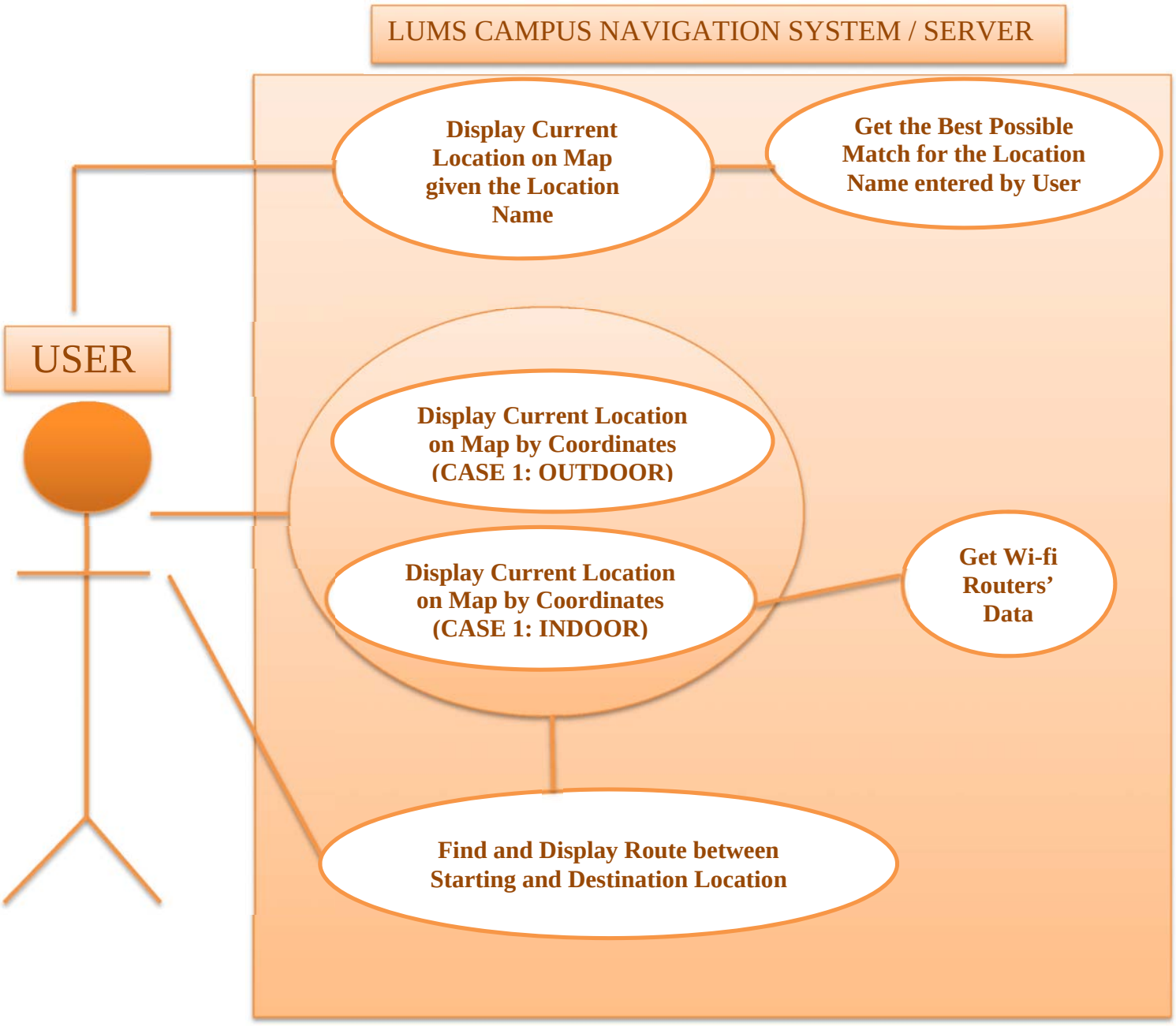

Figure 1. Use Case Diagram for the LUMS Campus Navigation System

3.2.1 Get the best possible match for the Location Name entered by user

Use case name	*Get the best possible match for the Location Name entered by user.*
Trigger	*User enters the name of a location (the name entered may not be the exact keyword in our locations' database)*
Precondition	*The names of all the major locations in the LUMS campus are saved in database.*
Basic Path	*1. The location name entered is sent to the server.* *2. Server searches for the best possible match for the location name entered by the user.* *3. The found location name is then sent back to our application.*
Exception path	*If location is not found in the database notify the user. Also, suggest the user at least three closest possible matches and allow the user to select one of them.*
Post Condition	*The name is finalized on confirmation by the user.*

3.2.2 Display Current Location on Map Given the Location Name

Use case name	*Display Current Location on Map Given the Location Name*
Trigger	*Location name is provided*
Precondition	*Application is connected to Wi-Fi. All the location names of LUMS campus in Google maps are tagged according to the location names in our database.*
Basic Path	*1. Location name is provided.* *2. Application then invokes the Google API to find and display the user's location on the map.*
Post Condition	*User current location is shown on the map by a pointer.*
Others	*Map information includes map of the campus with user current location being shown as an arrow head.*

3.2.3 Display Current Location on Map by Coordinates (CASE 1: OUTDOOR)

Use case name	*Display Current Location on Map by Coordinates (CASE 1: OUTDOOR)*

Trigger	*User presses button to find his/her current location*
Precondition	*Application is connected to Wi-Fi and GPRS internet or GPS*
Basic Path	*1. User presses current location button.* *2. Application gets the GPS coordinates from the GPS service.* *3. Application then invokes the Google API for locating and displaying the user's location on the map.*
Post Condition	*User current location is shown on the map by a pointer.*
Others	*Map information includes map of the campus with user current location being shown as an arrow head.*

3.2.4 Display Current Location on Map by Coordinates (CASE 2: INDOOR)

Use case name	*User Current location on Map by Coordinates (CASE 2: INDOOR)*
Trigger	*User presses the current location button*
Precondition	*Application is connected to the Wi-Fi routers installed in LUMS and their MAC addresses are saved in our database.*
Basic Path	*1.User presses current location button.* *2.Different available MAC addresses are provided.* *3.Application sends this information to application server.* *4.Server processes the information and sends back the name of the location and floor number.* *5.Application displays this location on the map.*
Exception Path	*If the application is not able to track your position, then, the user will have to enter the current location manually.*
Post Condition	*User current location is shown on the map.*
Others	*Map information includes map of the campus with user current location being shown as an arrow head.*

3.2.5 Get Wi-Fi Routers Data

Use case name	*Get Wi-Fi Router Data*

Trigger	*User presses the current location button*
Precondition	*Application is connected to the Wi-Fi routers installed in LUMS and their MAC addresses are saved in our database.*
Basic Path	*1.Application will use ARP and different network level protocols to find the MAC addresses of the Wi-Fi routers.* *2.It will also measure the signal strengths of different routers and will associate them with their respective routers MAC addresses.* *3.In the end it will format this data in a specific manner so that we may send this data to server for further processing and computation.*
Post Condition	*Gives the formatted data back as "String"*

3.2.6 Find and Display Route between Starting and Destination Location

Use case name	*Find and Display Route between Starting and Destination Location*
Trigger	*User provides for starting and destination location (starting location can be automatically found by GPS service if available)*
Precondition	*All the location names of LUMS campus in Google Maps have been tagged correctly in accordance with the location names in application database on server and the provided name are also in accordance with these names.*
Basic Path	1. *Checks whether the provided locations are block level or indoor location.* 2. *If any of the provided location is not a block level location then application finds the block level location associated with the the provided indoor location/locations.* 3. *Then application invokes the Google Map API with block level starting and destination location as an argument to find and display the rout on the map.* 4. *For indoor location application will have its own algorithm for finding and displaying the rout on map.*
Post Condition	*route will be displayed on the map by a coloured line with starting and destination location highlighted by the different coloured pointers*

3.3 Non-Functional Requirements

3.3.1 Performance

The purpose of the LUMS Campus Navigator is to save time and effort; hence, the performance of the application is one of the major priorities. All the output will take less than a second to generate.

3.3.2 Reliability

The LUMS Campus Navigator relies heavily on a good Wi-Fi system and the correct functioning of GPS. Therefore, reliability might be compromised if the above two components are not working.

3.3.3 Availability

The LUMS Campus Navigator is specifically being designed keeping the availability requirements in mind i.e. we are designing it on Android since, it is the most common and abundant technology of the day.

3.3.4 Security

As stated above, the security concerns involve the MAC addresses of the routers, which is sensitive information. Specifically for this purpose, we are using the client-server model, and we keep this sensitive information, encrypted on the server-side.

3.3.5 Maintainability

The application will require maintenance if there three major changes:

- If LUMS expands its campus, we'll have to make sure that the area/buildings are marked on Google Maps, since; we will be using them for our outdoor navigation.
- Similarly, if the location of the offices is changed in the academic block, we will have to make those changes on our map and graph it accordingly.
- MAC addresses of routers are unique; therefore, if one of the routers are changed or added, we will have to update our router's database accordingly.

3.3.6 Portability

LUMS Campus Navigator is available on your mobile phones and hence, is just as portable as your mobile phone is.

3.4 Design Constraints

In order to keep the MAC addresses private, we have to introduce the client-server model, which is additional work. Maintaining the server would add up to this extra work.

LUMS Campus Navigator will be developed as an Android system.

3.5 Logical Database Requirements

We will be using a MySQL database for the keeping all the information about the routers.

4. Change Management Process

If there are additions/ changes in the LUMS campus, in terms of the outdoor map, we will have to update Google Maps for that.

If there are changes in the indoor map i.e. the map of the academic block, we will have to update the graph that we have made out of the academic block map.

If there are additions/changes in the routers being currently used, the information will need to be updated in the router's database.

All these permissions will be granted by the LUMS administration and the developers of this project.