Software Requirements Specification

for

Task Notifier app

Version 1.0 approved<X.X>

Prepared by

Group Name: <*place your group number here*>

|  |  |  |
| --- | --- | --- |
| NAKYEWA IRENE | 16/U/851 | [nakyewa205@gmail.com](mailto:nakyewa205@gmail.com) |
| ABILA RAPHEAL | 16/U/2673/PS | [rphlabila@gmail.com](mailto:rphlabila@gmail.com) |
| AGABA DAVIS | 16/U/2812/PS | [davisag67@gmail.com](mailto:davisag67@gmail.com) |
| MWAITA JOSHUA | 16/U/7890/PS | [Jmwaita95@gmail.com](mailto:Jmwaita95@gmail.com) |

|  |  |
| --- | --- |
| Mentor: | <*place your mentor’s name here (if known)>* |
| Course: | CSC 1304 Practical Skills Development |
| Date: | 6/25/2018 |

Contents

Revisions iii

1 Introduction 1

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 1

1.5 Document Conventions 1

1.6 References and Acknowledgments 2

2 Overall Description 3

2.1 Product Perspective 3

2.2 Product Functionality 3

2.3 Users and Characteristics 3

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

3 Specific Requirements 5

3.1 External Interface Requirements 5

3.2 Functional Requirements 6

3.3 Behaviour Requirements 6

4 Other Non-functional Requirements 7

4.1 Performance Requirements 7

4.2 Safety and Security Requirements 7

4.3 Software Quality Attributes 7

5 Other Requirements 8

Appendix B - Group Log 9

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| Draft Type and Number | Full Name | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# 

# *<In this template you will find text bounded by the “<>” symbols. This text appears in italics and is intended to guide you through the template and provide explanations regarding the different sections in this document. There are two types of comments in this document. These comments that are in black are intended specifically for that course. These comments that are in blue are more general and apply to any SRS. Please, make sure to delete all of the comments before submitting the document.*

# *The explanations provided below, do not cover all of the material, but merely, the general nature of the information you would usually find in SRS documents. It is based on the IEEE requirements and was adapted specifically for the needs of CSC 1304 Practical Skills Development course. Most of the sections in this template are required sections, i.e. you must include them in your version of the document. Failure to do so will result in marks deductions. Optional sections will be explicitly marked as optional.*

# Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

## Document Purpose

## The purpose of this document is to give a detailed description of the requirements for the “Geofencing app” (GA) software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team. Geofencingallows limitations to be established as to where Systems Manager devices are allowed to be when devices are detected outside of that area, alerts can be generated to notify administrators and designated profiles can be applied or removed.

## Product Scope

Geofencing allows an administrator to setup triggers so when a device enters(or exits) the boundaries defined by the administrator , an alert id issued .Many geo-fencing applications incorporate Google earth, allowing administrators to define on top of a satellite view of a specific geographical area .other applications define boundaries by longitude and latitude or through user-created and web-based maps.

Furthermore, the software needs both Internet and GPS connection to fetch and display results. All system information is maintained in a database, which is located on a web-server. The software also interacts with the GPS-Navigator software which is required to be an already installed application on the user’s mobile phone.

One of the most useful features of mobile devices is its location and thus it helps a user navigate, automatically switch to the time zone that he is in and allow for location targeted push notifications ,or give insights into whatever activities take place at a certain location.

GA is to be understood as a reference application that uses and validates the appropriateness of the interfaces, components and specifications that are defined for the geolocation and Google-services in the project.

## Intended Audience and Document Overview

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers (In your case it would probably be the “client” and the course facilitator). Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

The intended audience for this app are basically the clients and the facilitator as specified since they are the users who can be able to access the app but with time we would want to consider widening the audience .

The requirements to be able to access the app is an android phone or an emulator since its an android application .The remainder of this document includes three chapters and appendixes. The second one provides an overview of the system functionality and system interaction with other systems. This chapter also introduces different types of stakeholders and their interaction with the system. Further, the chapter also mentions the system constraints and assumptions about the product.

The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences.

The fourth chapter deals with the prioritization of the requirements. It includes a motivation for the chosen prioritization methods and discusses why other alternatives were not chosen. The Appendixes in the end of the document include the all results of the requirement prioritization and a release plan based on them.

## Definitions, Acronyms and Abbreviations

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.

TO DO: Please provide a list of all abbreviations and acronyms used in this document sorted in alphabetical order.>

**Table 1 – Definitions, Acronyms and Abbreviations**

|  |  |
| --- | --- |
| Term | Definition |
| user | Someone who interacts with the mobile phone application |
| Admin/Administrator | System administrator who is given specific permission for managing and controlling the system |
| GA | Geofencing App |
| GPS | Global Positioning System |
| GPS-Navigator | An installed software on mobile phone which could provide GPS connection and data, show locations on map and find paths from current position to defined destination |
| Application Store | An installed application on mobile phone which helps user to find new compatible applications with mobile phone platform and download them from Internet |
| Stakeholder | Any person who has interaction with the system who is not a developer. |
| DESC | Description |
| RAT | Rational |
| DEP | Dependency |
| TAG | A unique, persistent identifier contained in a PLanguage statement [2] |
| GIST | A short, simple description of the concept contained in a PLanguage statement [2] |
| PLAN | The level at which good success can be claimed contained in a PLanguage statement [2] |
| DEFINED | The official definition of a term contained in a PLanguage statement [2] |
| MUST | The minimum level required to avoid failure contained in a PLanguage statement [2] |
| WISH | A desirable level of achievement that may not be attainable through available means contained in a PLanguage statement [2] |
| METER | The process or device used to establish location on a SCALE contained in a PLanguage statement [2] |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Document Conventions

<In general this document follows the IEEE formatting requirements. Use Arial font size 11, or 12 throughout the document for text. Use italics for comments. Document text should be single spaced and maintain the 1” margins found in this template. For Section and Subsection titles please follow the template.

TO DO: Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. Sometimes, it is useful to divide this section to several sections, e.g., Formatting Conventions, Naming Conventions, etc.>

This document follows MLA Format. Bold-faced text has been used to emphasize section and sub-section headings. Highlighting is to point out words in the glossary and italicized text is used to label and recognize diagrams.

*Main Section Titles*

* Font: Times New Roman
* Face: Bold
* Size: 14

*Sub Section Titles*

* Font: Times New Roman
* Face: Bold
* Size: 12

*Other Text Explanations*

* Font: Times New Roman
* Face: Normal
* Size: 12

## References and Acknowledgments

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document.

TO DO: Use the standard IEEE citation guide for this section. An example citation guide can be found at this URL <http://www.ieee.org/documents/ieeecitationref.pdf> .>

<https://www.webopedia.com/TERM/G/geo-fence.html>

**References**

[1] IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

[2] Feldt R,”re\_lecture5b\_100914”, unpublished.

[3] Davis M A, “Just Enough Requirements Management: Where Software Development Meets Marketing”, New York, Dorset House Publishing, 2005.

[4] Karlsson J, “A Cost-Value Approach for Prioritizing Requirements”, Norges Teknisk-Naturvitenskapelige Uni. 1997

**Acknowledgements**

We thank the almighty God who has enabled us to work on our project .thank you lord

We highly appreciate and convey sincere gratitude to our instructors for being available doing their level best to make us excel.

And to our supervisor thanks for continuous support and allowing us continue with our project implementation.

# Overall Description

## This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

## Product Perspective

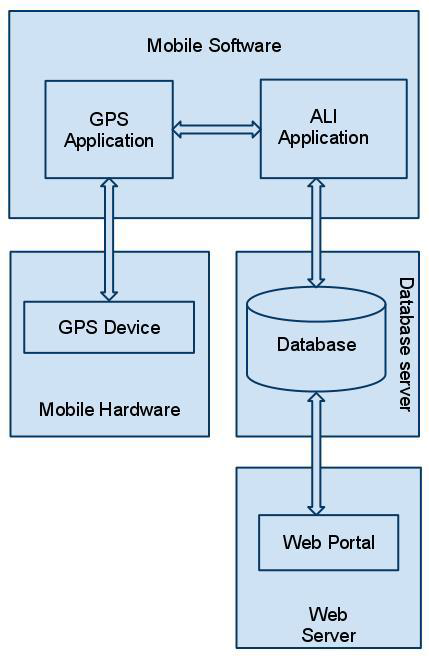
<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface. In this section it is crucial that you will be creative and provide as much information as possible.

TO DO: Provide at least one paragraph describing product perspective. Provide a general diagram that will illustrate how your product interacts with the environment and in what context it is being used.>

This system will consist of two parts: The mobile application that will be used get the location of the user and view information including the task to be implemented using a notification message .

The mobile application will need to communicate to a GPS application within the mobile phone, which in turn communicates with a physical GPS device to find the location of the user, see Figure1. The GPS will provide the mobile application with locations of both the user and the restaurants and the distance between them, but it will also provide maps and the functionality to display the application’s data on the map. The functionality provided by the GPS will be embedded into the application in order for the user to be able to use the functions in the application in a seamlessly manner.

The mobile application has some restrictions about the resource allocation. To avoid problems with overloading the operating system the application is only allowed to use 20 megabytes of memory while running the application. The maximum amount of hard drive space is also 20 megabytes.



**Figure 1 - Block diagram**

## Product Functionality

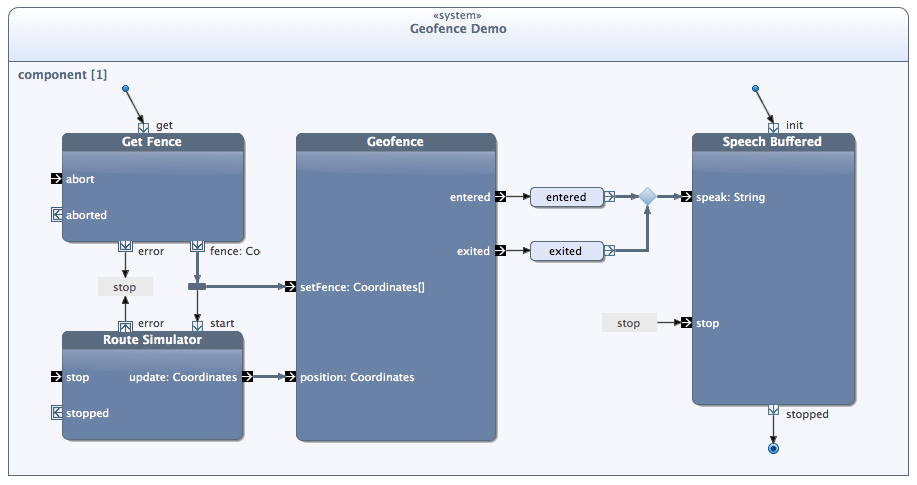
<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, will be effective.

TO DO:

1. Provide a bulleted list of all the major functions of the system

2. **(Optional)** Provide a Data Flow Diagram of the system to show how these functions relate to each other.>

Functions of Geofences



## Users and Characteristics

<Identify the various users that you anticipate will use this product. Users may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience.

TO DO:

1. Describe the pertinent characteristics of each user. Certain requirements may pertain only to certain users.

3. Distinguish the most important users for this product from those who are less important to satisfy.>

There are three types of users that interact with the system in this case : users of the mobile application or clients , administrators/developers,facilitators. Each of these three types of users has different use of the system so each of them has their own requirements.

The mobile application users/clients can only use the application to find the location and set a geofence,clearing a geofence .In order for the users to get a relevant search result there are multiple criteria the users can specify and all results matches all of those.

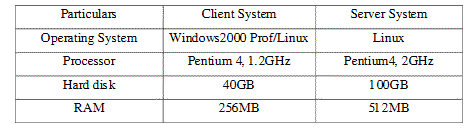
The administrators also only interact with the backend of the system . They are managing the overall system so there is no incorrect information within it.

## Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface

TO DO: As stated above, in at least one paragraph, describe the environment your system will have to operate in. Make sure to include the minimum platform requirements for your system. >

|  |  |  |
| --- | --- | --- |
| particulars | Client system | Server system |
| Operating system | Windows 200, prof/linux | Linux |
| Processor | Pentium 4, 1.2GHz | Pentium 4, 2GHz |
| Hard disk | 40GB | 100GB |
| RAM | 256MB | 512MB |



## Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).

TO DO: In this section you need to consider all of the information you gathered so far, analyze it and correctly identify at least 5 constraints.>

**Design constraints**

This section includes the design constraints on the software caused by the hardware.

***2.5.0 Hard drive space***

**ID: QR**

TAG: Hard DriveSpace

GIST: Hard drive space.

SCALE: The application’s need of hard drive space.

METER: MB.

MUST: No more than 20 MB.

PLAN: No more than 15 MB.

WISH: No more than 10 MB.

MB: DEFINED: Megabyte

***2.5.1 Application memory usage***

**ID: QR**

TAG: ApplicationMemoryUsage

GIST: The amount of Operate System memory occupied by the application.

SCALE: MB.

METER: Observations done from the performance log during testing

MUST: No more than 20 MB.

PLAN: No more than 16 MB

WISH: No more than 10 MB

Operate System: DEFINED: The mobile Operate System which the application is running on.

MB: DEFINED: Megabyte.

**Implementation constraints**

The mobile application is constrained by the system interface to the GPS navigation system within the mobile phone. Since there are multiple system and multiple GPS manufacturers, the interface will most likely not be the same for every one of them. Also, there may be a difference between what navigation features each of them provide.

The Internet connection is also a constraint for the application. Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function.

The mobile application will be constrained by the capacity of the database in case there’s some information to be captured by the system. Since the database is shared between both application it may be forced to queue incoming requests and therefore increase the time it takes to fetch data.

The mobile application is also designed to run on only android mobile phones and it cannot run on any other such as windows phones.

In case the application is installed on the mobile phone, it cannot be able to run and display the map when Google play services is not installed or updated.

## User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.

TO DO: You will not actually develop any user-manuals, but you need to describe what kind of manuals and what kind of help is needed for the software you will be developing. One paragraph should be sufficient for this section.>

The product is under development stage and requires a complete implemented prototype to explain the user documentation. Once the prototype is designed and implemented online manuals, user manuals can be provided.

## Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project.

TO DO: Provide a short list of some major assumptions that might significantly affect your design. For example, you can assume that your client will have a reliable internet connection (if your project requires an internet connection). Every number has a significant effect on the design of your system. >

One assumption about the product is that it will always be used on mobile phones that have enough performance. If the phone does not have enough hardware resources available for the application, for example the users might have allocated them with other applications, there may be scenarios where the application does not work as intended or even at all.

Another assumption is that the GPS components in all phones work in the same way. If the phones have different interfaces to the GPS, the application need to be specifically adjusted to each interface and that would mean the integration with the GPS would have different requirements than what is stated in this specification.

# Specific Requirements

## External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

### User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., Cancel) that will appear on every screen, error message display standards, and so on. Define the software components for which a user interface is needed.

TO DO: The least you can do for this section is to describe in words the different User Interfaces and the different screens that will be available to the user. Those who will be able to provide optional Graphical User Interface screenshots, will be rewarded by extra marks.>

Sample Screenshots of the Graphical User Interfaces

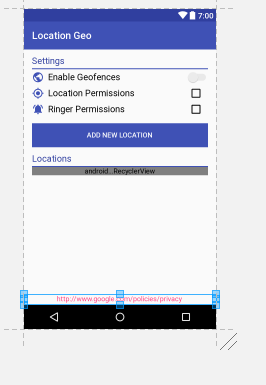
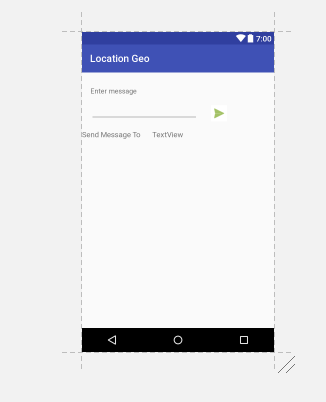
 

Fig2 .Create a geofence Fig 3.Send a text message

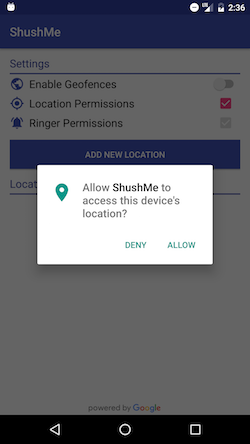
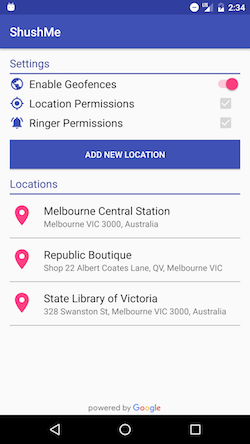
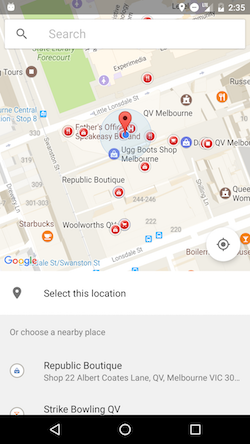
  

Fig 4.Enable a geofence Fig 5 Allowing access user location fig6. display geofence

### Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware. You are not required to specify what protocols you will be using to communicate with the hardware, but it will be usually included in this part as well.

TO DO: Please provide a short description of the different hardware interfaces. If you will be using some special libraries to communicate with your software mention them here. In case you have more than one hardware interface divide this section into subsections.>

### Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems (Android? iOS? Windows? Linux? Etc…), tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.

TO DO: The previous part illustrates some of the information you would usually include in this part of the SRS document. To make things simpler, you are only required to describe the specific

The mobile application communicates with the GPS application in order to get geographical information

about where the user is located and the visual representation of it, and with the database in order to get the information about the restaurants, see Figure 1. The communication between the database and the web portal consists of operation concerning both reading and modifying the data, while the communication between the database and the mobile application consists of only reading operations.

### Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.

TO DO: Do not go into too much detail, but provide 1-2 paragraphs were you will outline the major communication standards. For example, if you decide to use encryption there is no need to specify the exact encryption standards, but rather, specify the fact that the data will be encrypted and name what standards you consider using. >

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for both the mobile application and the web portal

## Functional Requirements

*< Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. This section is the direct continuation of section 2.2 where you have specified the general functional requirements. Here, you should list in detail the different product functions with specific explanations regarding every function.*

*TO DO: Break the functional requirements to several functional areas and divide this section into subsections accordingly. Provide a detailed list of all product operations related to these functional areas.*

This section includes the requirements that specify all the fundamental actions of the software system.

### User Class 1 - The User

***3.2.1.1 Functional requirement 1.1***

**ID: FR1**

TITLE: Download mobile application

DESC: A user should be able to download the mobile application through either an application store or similar service on the mobile phone. The application should be free to download.

RAT: In order for a user to download the mobile application.

DEP: None

***3.2.1.2 Functional requirement 1.2***

**ID: FR2**

TITLE: Download and notify users of new releases

DESC: When a new/updated version or release of the software is released, the user should check for these manually. The download of the new release should be done through the mobile phone in the same way as downloading the mobile application.

RAT: In order for a user to download a new/updated release.

DEP: FR1

***3.2.1.3 Functional requirement***

**ID: FR3**

TITLE: User registration - Mobile application

DESC: Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide user-name, password and e-mail address. The user can choose to provide a regularly used phone number which will also be implemented as part of the application.

RAT: In order for a user to register on the mobile application.

DEP: FR1

***3.2.1.4 Functional requirement***

**ID: FR4**

TITLE: Mobile application - Search result in a map view

DESC:

-There should be maximally 100 results displayed. The map view should have a default zoom.

- The map view should include a button that, when selected, should display different filtering options in a filtering menu.

RAT: The way results are displayed in a map.

DEP: FR1,FR3.

***3.2.1.5 Functional requirement***

**ID:FR5**

TITLE: Sending and Recieving Notifications

DESC:

RAT:

DEP:FR4

***3.2.1.6 Functional Requirement***

**ID: FR6**

TITLE: Mobile application - Search by destination

DESC: A user should be able to input a maximum and a minimum distance, according to his/her position. By default the minimum distance is set to 0 km and the maximum to 10 km. The user should be able to input a higher or lower maximum distance and a higher minimum distance than set by default. The result is displayed in a map view by default.

RAT: In order for a user to search by destination.

DEP: FR7

## Behaviour Requirements

### Use Case View

<A use case defines a goal-oriented set of interactions between external actors and the system under consideration. Since sometimes we will not be able to specify completely the behaviour of the system by just State Diagrams, we use use-cases to complete what we have already started in section 3.3.1.

TO DO: Provide a use case diagram which will encapsulate the entire system and all possible actors. Do not include detailed use case descriptions, but make sure to include a short description of what every use-case is, who are the actors in your diagram. For more information please refer to Wikipedia article <http://en.wikipedia.org/wiki/Use_case>

# Other Non-functional Requirements

## Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.

TODO: Provide at least 5 different performance requirements based on the information you collected from the client. For example you can say “1. Any mobile money transaction will not take more than 10 seconds, etc…>

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.

***4.1.4 Response time***

**ID: QR6**

TAG: ResponseTime

GIST: The fastness of the search

SCALE: The response time of a search

METER: Measurements obtained from 1000 searches during testing.

MUST: No more than 2 seconds 100% of the time.

WISH: No more than 1 second 100% of the time.

***4.1.5 System dependability***

**ID: QR8**

TAG: SystemDependability

GIST: The fault tolerance of the system.

SCALE: If the system loses the connection to the Internet or to the GPS device or the system gets some strange input, the user should be informed.

METER: Measurements obtained from 1000 hours of usage during testing.

MUST: 100% of the time.

***4.1.0 Prominent search feature***

**ID: QR1**

TITLE: Prominent search feature

DESC: The search feature should be prominent and easy to find for the user.

RAT: In order to for a user to find the search feature easily.

DEP: none

***4.1.1 Usage of the search feature***

**ID: QR2**

TITLE: Usage of the search feature

DESC: The different search options should be evident, simple and easy to understand.

RAT: In order to for a user to perform a search easily.

DEP: none

***4.1.2 Usage of the result in the side dropdown list view***

**ID: QR3**

TITLE: Usage of the result in the list view

DESC: The results displayed in the list view should be user friendly and easy to understand. Selecting an element in the result list should only take one click.

RAT: In order to for a user to use the list view easily.

DEP: none

***4.1.3 Usage of the result in the map view***

**ID: QR4**

TITLE: Usage of the result in the map view

DESC: The results displayed in the map view should be user friendly and easy to understand. Selecting a pin on the map should only take one click.

RAT: In order to for a user to use the map view easily.

DEP: none

## Safety and Security Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied. Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements.

TODO:

* Provide at least 3 different safety requirements based on your interview with the client or, on your research, and again you need to be creative here.
* Describe briefly what level of security is expected from this product by your client and provide a bulleted (or numbered) list of the major security requirements.>

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.

TODO: Use subsections (e.g., 4.3.1 Reliability, 4.3.2 Portability, etc…) provide requirements related to the different software quality attributes. Base the information you include in these subsections on the material you have learned in the class. Make sure, that you do not just write “This software shall be maintainable…” Indicate how you plan to achieve it, & etc…Do not forget to include such attributes as the design for change. Please note that you need to include at least 2 quality attributes, but it is the mere minimum and it will not receive the full marks.>

The requirements in this section specify the required reliability, availability, security and maintainability of the software system

***4.3.1 Reliability***

**ID: QR5**

TAG: SystemReliability 24

GIST: The reliability of the system.

SCALE: The reliability that the system gives the right result on a search.

METER: Measurements obtained from 1000 searches during testing.

MUST: More than 98% of the searches.

PLAN: More than 99% of the searches.

WISH: 100% of the searches.

***4.3.2 Availability***

**ID: QR6**

TAG: SystemAvailability

GIST: The availability of the system when it is used.

SCALE: The average system availability (not considering network failing).

METER: Measurements obtained from 1000 hours of usage during testing.

MUST: More than 98% of the time.

PLAN: More than 99% of the time.

WISH: 100% of the time.

**ID: QR7**

TITLE: Internet Connection

DESC: The application should be connected to the Internet.

RAT: In order for the application to communicate with the database.

DEP: none

**ID: QR8**

TITLE: GPS Connection

DESC: The application should be connected to the GPS device.

RAT: In order for the application to get the users location, the map and to calculate the distance.

DEP: none

***4.3 .3 Security***

**ID: QR9**

TAG: CommunicationSecurity

GIST: Security of the communication between the system and server.

SCALE: The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages.

METER: Attempts to get user-name and password through obtained messages on 1000 log-in session during testing.

MUST: 100% of the Communication Messages in the communication of a log-in session should be encrypted. Communication Messages: Defined: Every exchanged of information between client and server.

***4.3.4 Maintainability***

**ID: QR10**

TITLE: Application extendibility

DESC: The application should be easy to extend. The code should be written in a way that it favors implementation of new functions.

RAT: In order for future functions to be implemented easily to the application.

DEP: none

**ID: QR11**

TITLE: Application testability

DESC: Test environments should be built for the application to allow testing of the applications different functions.

RAT: In order to test the application.

DEP: none

***4.3.5 Portability***

**ID: QR12**

TITLE: Application portability

DESC: The application should be portable Android.

RAT: The adaptable platform for the application to run on.

DEP: none

# Other Requirements

<This section is **Optional.** Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

**5.0 Prioritization and Release Plan**

In order to get a view of how to divide the requirements into different releases and what requirements should be included in which release, a prioritization of the requirements is needed. This section discusses the choice of prioritization methods and gives a suggestion of how the release plan for these requirements could look like.

**5.1 Choice of prioritization method**

When prioritizing the requirements the ten most important ones were picked out first. This was done with a simple “1 to 10” ranking method, with one being “not important” and ten “very important”. Based on the elicitation meetings, and the perceived ideas of what was important to the different stakeholders, a number was set for each requirement. The numbers were then summed up for each requirement and the ten with the highest score were chosen to be prioritized with the cost value approach. The results, which are red-marked, can be seen in Appendix I and as shown, it turned out to be five functional requirements and five quality requirements. These requirements were then prioritized according to the cost value approach and the results can be viewed under Appendix II.

The remaining requirements were prioritized according to the “Five-Way Priority Scheme” as shown in Appendix III. This method was chosen since it gives the different stakeholders the same importance and has an enough wide range for determining which requirement is more important than the other [3]. However, in this prioritization process, the development team was not included as a stakeholder since the different features were not considered to be as important to them as for the other stakeholders.

Other methods for prioritization, such as the hundred-dollar test and the yes-no vote, were also considered. The hundred-dollar test is quite similar to the five-way priority scheme, since it also gives a wide range for ranking the requirements. However, it is more easily misused since someone could save all their money and put them on a requirement that they think is very important [3]. Others might not agree that this requirement is important but it might still get the most votes since one person cared about it [3].

The yes-no vote method might be fairly simple to carry out, however the range is too narrow. For instance, if two requirements are not very important it would be hard to determine which of those requirements that is more important than the other [3].

In conclusion, weighing the disadvantages and advantages of these methods against each other lead us to choose the five-way priority scheme.

**5.2 Release Plan**

The requirements were divided into three releases based on the prioritization and their dependencies. The three different releases were assembled so that each would work as a fully functional application.

In the first release the requirements that build up the foundation of the application were included, together with the most highly prioritized requirements and their dependencies.

The second release also includes important requirements. However, these requirements are not vital for a functional application. They are more suited to act as additional features that can contribute to making the software product more attractive.

The third release includes the requirements that can be afforded to discard if the project gets delayed or overruns the budget.

Appendix B - Group Log

<Please include here all the minutes from your group meetings and meetings with users, your group activities, and any other relevant information that will assist the Course Facilitator to determine the effort put forth to produce this document>