
Software Requirements Specification

Version 1.0 approved

Prepared by

Group Name: CSC18/20

NAKYEWA IRENE
ABILA RAPHEAL
AGABA DAVIS
MWAITA JOSHUA

16/U/851
16/U/2673/PS
16/U/2812/PS
16/U/7890/PS

nakvewa205@gmail.com
rphlabila@gmail.com
davisag67@gmail.com
Jmwaita95@gmail.com

Mentor: **prof. Engineer Bainomugisha**
Course: **CSC 1304 Practical Skills Development**
Date: **24/6/2018**

Contents

REVISIONS.....	III
1 INTRODUCTION.....	4

1.1	DOCUMENT PURPOSE.....	4
1.2	PRODUCT SCOPE	4
1.3	INTENDED AUDIENCE AND DOCUMENT OVERVIEW .. ERROR! BOOKMARK NOT DEFINED.	
1.4	DEFINITIONS, ACRONYMS AND ABBREVIATIONS	5
1.5	DOCUMENT CONVENTIONS	7
1.6	REFERENCES AND ACKNOWLEDGMENTS.....	8
2	OVERALL DESCRIPTION	10
2.1	PRODUCT PERSPECTIVE	10
2.2	PRODUCT FUNCTIONALITY.....	11
2.3	USERS AND CHARACTERISTICS	12
2.4	OPERATING ENVIRONMENT	13
2.5	DESIGN AND IMPLEMENTATION CONSTRAINTS	14
2.6	USER DOCUMENTATION.....	15
2.7	ASSUMPTIONS AND DEPENDENCIES	15
3	SPECIFIC REQUIREMENTS	16
3.1	EXTERNAL INTERFACE REQUIREMENTS	16
3.2	FUNCTIONAL REQUIREMENTS	19
3.3	BEHAVIOUR REQUIREMENTS.....	21
4	OTHER NON-FUNCTIONAL REQUIREMENTS	23
4.1	PERFORMANCE REQUIREMENTS	23
4.2	SAFETY AND SECURITY REQUIREMENTS	24
4.3	SOFTWARE QUALITY ATTRIBUTES	25
5	OTHER REQUIREMENTS	28
	APPENDIX B - GROUP LOG.....	30

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
.1/1	Nakyewa Irene,Agaba Davis	Overall Description given	20/06/2018
2/2	Abilla Rapheal,Mwaita Joshua	Updating and verifications	22/06/2018

1 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.

1.1 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. Geofencing allows 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.

1.2 Product Scope

Geofencing allows an administrator to setup triggers so when a device enters(or exits) the boundaries defined by the administrator , an alert is 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.

1.3 Targeted Audience and Document Overview

The intended audience for this app are basically the clients(users who can access the mobile application) 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.

1.4 Definitions, Acronyms and Abbreviations

1.4 Targeted Audience and Document Overview

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]

1.5 Document Conventions

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
- Line spacing: 1.5

1.6 References and Acknowledgments

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
- [5] Cuellar, J.R. 2002. Location Information Privacy. B. Srikaya (Ed.), Geographic Location in the Internet. Kluwer Academic Publishers. 179-208.
<https://www.webopedia.com/TERM/G/geo-fence.html>
- [6] Dey, A. K., G. D. Abowd. 1999. Towards a Better Understanding of Context and Context-Awareness. Technical Report GIT-GVU-99-22, Georgia Institute of Technology
- [7] Farrell, T., R. Cheng, and K. Rothermel. 2007. Energy-efficient Monitoring of Mobile Objects with Uncertainty-aware Tolerances. Proceedings of the 11th International Database Engineering and Applications Symposium (IDEAS '07). Washington, DC, USA. IEEE Computer Society. 129–140.
- [8] Gruteser, M., D. Grunwald. 2003. Anonymous Usage of Location-based Services Through Spatial and Temporal Cloaking. Proceedings of the ACM/USENIX International Conference on Mobile Systems, Applications and Services (MobiSys '03), San Francisco, California, USA, 31-42

Acknowledgements

We thank the almighty God who has enabled us to start working on our project and we pray he enable us finish it well and successful .Thank you lord.

We are also thankful to the department of computer science because of whom we have gained confidence in mobile application development and it has also enhanced in our professional skills so as to become competent in this this field.

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

We are grateful to our project guides for their support and guidance we respect them for their understanding of students problems and their ability to solve them with ease.

We are grateful to our department head **prof. Engineer Bainomugisha** and as well as our supervisor for his support and guidance in this project we are developing thank you , we have started out this project with excitement of embarking on a new and long journey , advancing with heaps and bounds everyday and exploring new avenues that the field of computerized operation offers .

2 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.

2.1 Product Perspective

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,

The GPS will provide the mobile application with locations of both the user 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. [3]

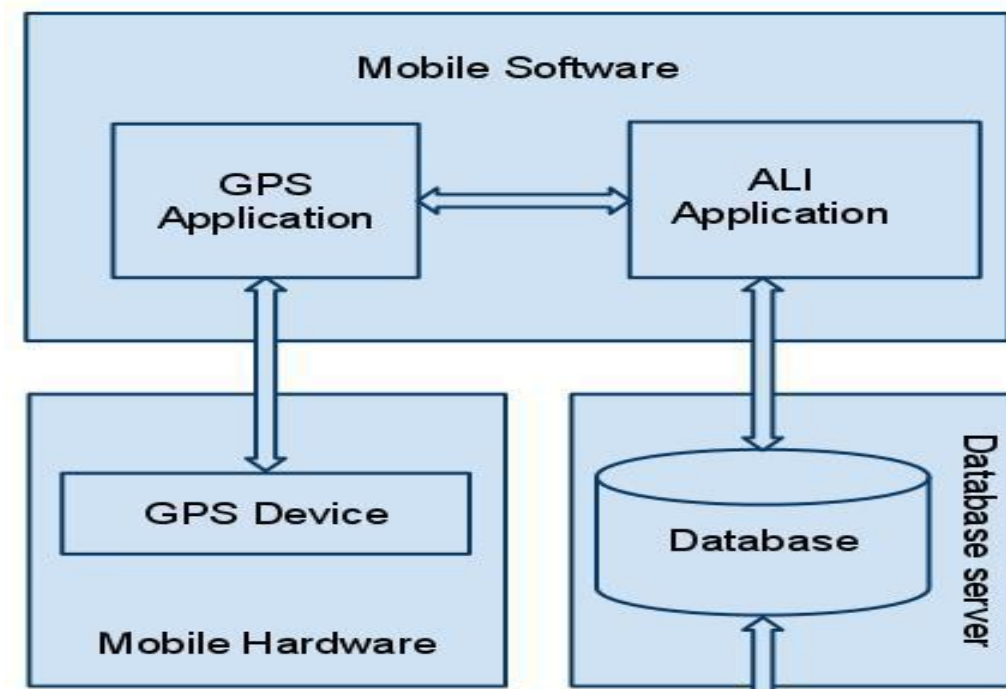


Figure 1 - Block diagram

2.2 Product Functionality

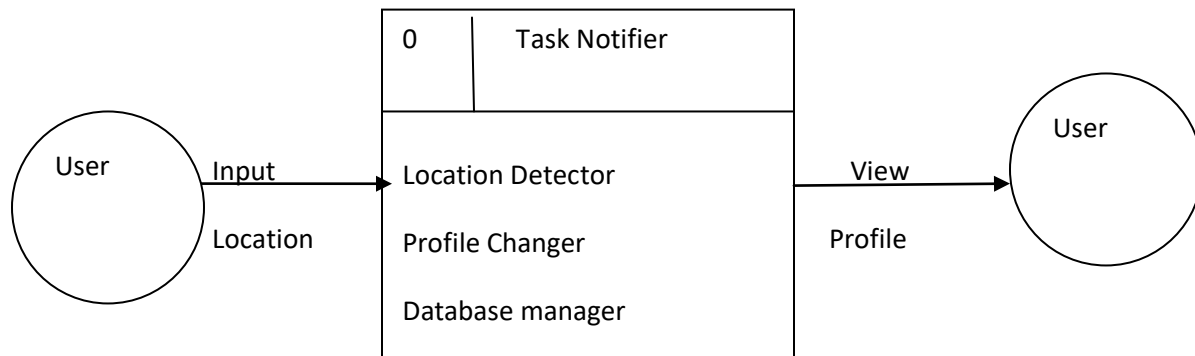
Functions of Geofences

- Ability to define areas in the physical world, where you want your application users to receive a message, when they arrive at a specific place. This could be an advertising, or a piece of information.

- The system allows a user to build the fence digitally around the area on the map by drawing. When your customers walk inside the fenced area,
- The app will be able to detect the fence, and send them a notification.
- The system can display all the items or set locations from the geofence

Data flow Diagram

Fig2:DFD Level-0



2.3 Users and Characteristics

Software designers: These are the most obvious users of the system. They will use the system as a means of buying out the design of Notifier of which as yet unimplemented system. In over case these users can be specified as our team members only.

Software developers: These are the people that take the model generated by the designers and implement it in the code. They may also use the system to identify the design of the existing system in order to maintain it.

The End users (students, professionals, Lecturers (course facilitators))

- ✓ Students: They use our application for their personal profile settings use and works regarding the personalization.
- ✓ Facilitators: They use this application for their own use and for their professional use.
- ✓ Professional: They use this application for auto advanced auto system and make them free to set all work time by time.

2.4 Operating Environment

As This smart phone application, it requires information to be stored .All the above information stored should be accurate and so this information can be used in future for verification.

Development of the application is done by android development tools using java language.

- ✓ Platform: Android
- ✓ Database: Mysql
- ✓ Map API: Google Maps API V2
- ✓ IDE:Android Studio
- ✓ Emulator: Android SDK[1]

- ✓ Android Smart phone for Real Simulation.

2.5 Design and Implementation 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.
- ✓ Battery usage should be high during GPS connection

2.6 User Documentation

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.

2.7 Assumptions and Dependencies

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.

3 Specific Requirements

3.1 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.

3.1.1 User Interfaces

- Settings: The first interface when the user opens the application is for settings, this interface mainly need action by the user, in this view, geofences are enabled and the connection will be enabled between GPS. Location and ringer permissions are also enabled by the user and he is required to add a location which will later be used when Geofencing is applied
- Maps view: This interface enables a user to search for any location or choose a nearby location one the maps are displayed and when there's internet connection.
- Messaging :This view will also be implemented so that a user can be able to send a text message to another user in case they are in the same boundary/ fence
- Basically most features are implemented in those two graphical user interfaces

Sample Screenshots of the Graphical User Interfaces

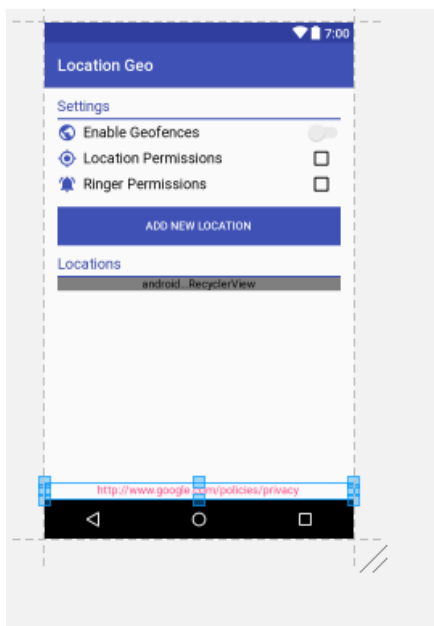


Fig3 .Create a geofence

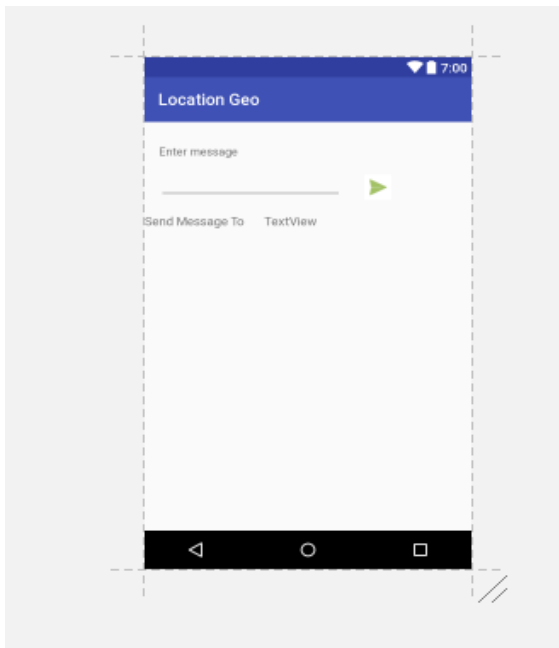


Fig 4.Send a text message

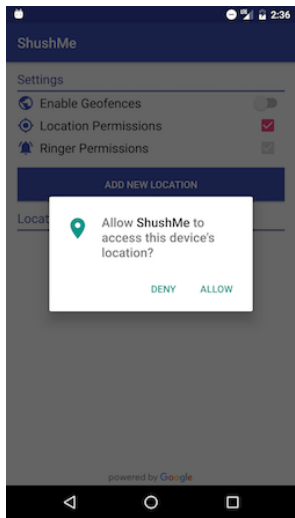


Fig 5.Enable a geofence

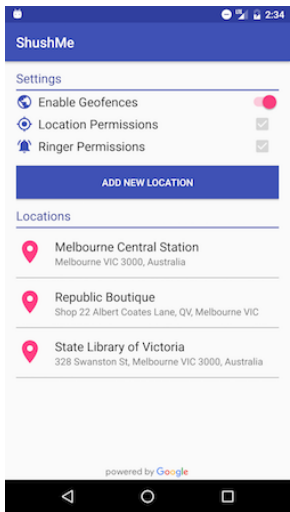


Fig 6:Allowing access user location

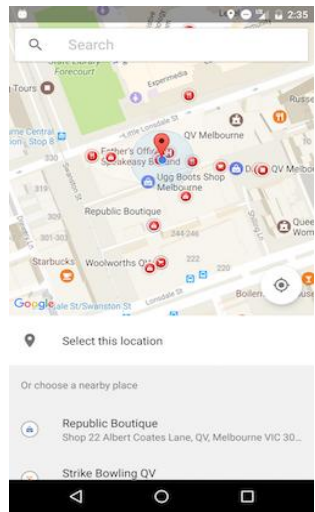


fig6:display geofence

3.1.2 Hardware Interfaces

3.1.3 Developer Hardware and software characteristics

✓ Hardware

- Intel Pentium Processors at 2.5GHz and upgraded
- 512MB RAM minimum(Recommended 1GB RAM)
- Any android smart phone which inbuilt GPS Receiver

✓ Software

- OS:Windows 7 and upgraded version
- Android Studio
- Android 2.3 edited SDK
- Sqlite(Backend)
- Java(Front-end)

3.1.3.1 End-user(Client)Hardware and Software characteristics

➤ Hardware

- ✓ GPS Enabled android smartphone

➤ Software

- ✓ Task Notifier Application

3.1.4 Software Interfaces

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.

3.1.5 Communications Interfaces

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

3.2 Functional Requirements

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

3.2.1 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 Receiving Notifications

DESC: The notification permission must be enabled

RAT: To communicate with someone with in a geofence

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

3.3 Behaviour Requirements

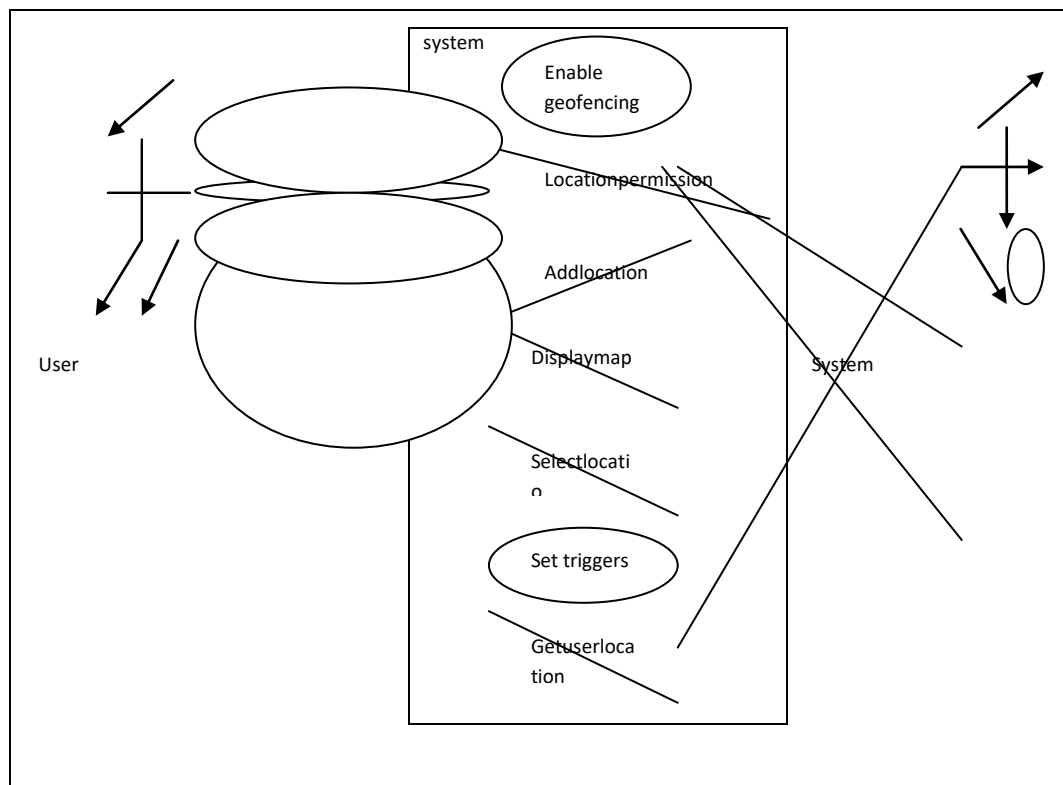
This section describes the different users and how they interact with the system including the necessary requirements /inputs and outputs of the system

3.3.1 Use Case View

Fig: Overview of Geofencing



Fig:Use case diagram of Task notifier app



4 Other Non-functional Requirements

4.1 Performance Requirements

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: Response Time

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: System Dependability

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

4.2 Safety and Security Requirements

4.2.1 Communication Security

The information between two users can only be allowed and carried out if they are within the same boundary

4.2.2 Safety of the input information

Once someone has uploaded his schedule, it shouldn't be edited by someone else, and someone can't get updates if not in the geofence.

4.2.3 Security on user events

The dates and time set for a user event shouldn't be changed by someone else and the notification should be enriched to that very user.

4.3 Software Quality Attributes

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

4.3.1 Reliability

ID: QR5

TAG: System Reliability 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: System Availability

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: Communication Security

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

5 Other Requirements

5.1 LITERATURE REVIEWS

5.1.1 Description of the technology used

- ✓ GPS is used to track the location of the user .As a GPS device ,the smart phone is used this application
- ✓ Using a smart phone ,the current location can be utilized for further requirements
- ✓ The user interface is designed in XML
- ✓ User can easily this application and need an internet connection to use the application.

5.1.2 Major Constituents of the system are

- ✓ GPS tracking Device: This is provided by the users smart phone
- ✓ User interface: The UI determines how the use will interact with the application and access the required functionality.

5.2 Closing thoughts

Geofencing is a profitable tool for connecting with consumers. By developing an app with Geofencing technology, you give your business a competitive edge. If done properly, this marketing tactic can build loyalty, generate real-time engagement, and maximize path-to-purchase conversions.

5.3Critical Considerations For ImplementingGeofencing

- It shouldn't be dependent on GPS – GPS will drain your app users' battery in no time.
- Geofencing software should be optimized for accuracy and battery usage – Choose a system that has a well-optimized balance between the two, not ones that claim to be super duper accurate; you can bet you'll find battery draining on the flip side.

- **Collect Data** – When it comes to interpreting data, you can't neglect offline behavior. Use these insights to target users in a more personalized way, and go beyond the average push notification.
- **Manage geofencing efforts** – A good Geofencing solution should come equipped with a management dashboard and API, which allows for easy creation and optimization of Geofences.
- **Don't spam your users** – Dense urban areas have a lot of Geofences. To avoid sending too many notifications it's critical to research and employ anti-spam tools.

5.4 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.

5.5 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

Appendix B - Group Log

This section contains the minutes of the group meeting with our facilitator for evaluation and progress.

Minutes of Fridays presentation on 22nd, June, 2018

Date and time: 22nd, June, 2018

Project: Task Notifier App

Location: LLT 2A

Supervisor: prof: Engineer Bainomugisha

Team Members:

1. NAKYEWA IRENE
2. AGABA DAVIS
3. ABILA RAPHAEL
4. MWAITA JOSHUA

Group Name: CSC18/20

Roll calling:

Topics

1. **Brief introduction About the app**
2. **Expectations of the user from the system**
 - ✓ User sets up an appointment.
 - ✓ User can set up multiple appointments
 - ✓ The system should allow the user to classify the appointments

- ✓ The user should be able to edit, delete and update an appointment
- ✓ The system should be able to set the reminder before the real time of the task execution happens.

3. Next week expectation

- ✓ Implementing the Geo-fencing component
- ✓ Working on the location component and the database.

For the Group meetings, we OOAD (Object Oriented Analysis and Design) for SDLC and used the UML (Unified Modeling Language) for analysis and design of the system.

We Discussed and decided to plan our project in the following ways

- ✓ During the first two days we had to study the various problems
- ✓ During the second week we had to start in analysis of the problems
- ✓ From the third week up to the end of the session we perform te design and implementation of the system.

Appendix B - Group Log

DATE	NAMES OF MEMEBERS	EACH CONTRIBUTION ON EVERY SECTION REQUIREMENT
18/06/2018	Nakyewa Irene	Document overview, product scope
	Abila Rapheal	product perspective, document overview, interfaces

	Mwaita Joshua	Product perspective, product functionality, interfaces
	Agaba Davis	Document purpose, product perspective, interfaces
	Abila Rapheal	Product scope, product functionality
DATE	NAMES OF MEMEBERS	EACH CONTRIBUTION ON EVERY SECTION REQUIREMENT
21/06/2018	Nakyewa Irene	Safety and security requirements, communication in interface
	Agaba Davis	Communication interfaces, functional requirement
	Abila Rapheal	User case view, functional requirements
	Mwaita Joshua	Functional requirements, user case view,
	Nakyewa Irene	Safety and security requirements, user case view
GENERAL CONCLUSION	ALL MEMBERS	Intended audience, document convention, users and characteristics, operating environment, design and implementation constraints, user documentation, assumptions and dependences, hardware interfaces, software interfaces, software quality attributes, performance requirement and communication interface.