MAKERERE****UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATION SCIENCES

DEPARTMENT OF NETWORKS

BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING (YEAR 2)

RECESS TERM 2 (BSE 2301)

SOFTWARE REQUIREMENTS SPECIFICATIONS FOR:

GROUP 11

GROUP MEMBERS

|  |  |  |
| --- | --- | --- |
| NAME | REGISTRATION NUMBER | STUDENT NUMBER |
| KEKIRUNGA JEAN | 16/U/5850/PS | 216004629 |
| MUGISA BEST | 16/U/7279/PS | 216018395 |
| KISIGA TIMOTHY | 16/U/6173/PS | 216003537 |

PROJECT LEADER: KISIGA TIMOTHY

ADVISOR: NOAH KANGE

SOFTWARE REQUIREMENTS SPECIFICATION

For

IOS MOBILE APP ANALYSIS DOCUMENT

Version 1.0 approved

Prepared by Kekirunga Jean, Kisiga Timothy, Mugisa Best

GROUP 11

26th June, 2018

Table of Contents

[1. Introduction 4](#_Toc517865617)

[1.1 Purpose 4](#_Toc517865618)

[1.2 Document Conventions 4](#_Toc517865619)

[1.3 Intended audience and Reading suggestions 4](#_Toc517865620)

[1.4 References 4](#_Toc517865621)

[1.5 Product scope 4](#_Toc517865622)

[2. Overall description 5](#_Toc517865623)

[2.1 Product perspective 5](#_Toc517865624)

[2.2 Product functions 5](#_Toc517865625)

[2.3 User classes and characteristics 7](#_Toc517865626)

[2.4 Operating environment 7](#_Toc517865627)

[2.5 Design and implementation constraints 7](#_Toc517865628)

[2.6 User documentation 7](#_Toc517865629)

[3. External interface requirements 7](#_Toc517865630)

[3.1 User interfaces 7](#_Toc517865631)

[3.2 Hardware interfaces 8](#_Toc517865632)

[3.3 Software interfaces 8](#_Toc517865633)

[3.4 Communication interfaces 8](#_Toc517865634)

[4. System features 9](#_Toc517865635)

[4.1 Get the trending applications 9](#_Toc517865636)

[4.1.1 Description and policy 9](#_Toc517865637)

[4.1.2 Stimulus and Response sequences 9](#_Toc517865638)

[4.1.3 Functional requirement 9](#_Toc517865639)

[4.2 Categorize the apps on the system 9](#_Toc517865640)

[4.2.1 Description and priority 9](#_Toc517865641)

[4.2.2 Stimulus/ response sequences 9](#_Toc517865642)

[4.2.2 Functional requirement 10](#_Toc517865643)

[5. Other Non-functional Requirements 10](#_Toc517865644)

[5.1 Performance requirements 10](#_Toc517865645)

[5.2 Safety requirements 10](#_Toc517865646)

[5.3 Security requirements 10](#_Toc517865647)

[5.4 Software quality attributes 10](#_Toc517865648)

[5.5 Business rules 10](#_Toc517865649)

[6. Appendix: Glossary 11](#_Toc517865650)

[7. Appendix B 11](#_Toc517865651)

# Introduction

# 1.1 Purpose

The purpose of this SRS document is to present a detailed description of the different functionalities of the system we are going to develop and implement, a software that studies the relationship between app details and user ratings in order to help Apple developers study the features of the most downloaded apps on the Apple app store and also to increase the user ratings for their apps.

The document will explain the purpose, features, interfaces, functional requirements, and non-functional requirements of the system.

The document is intended for the stakeholders of this project that is developers, Apple store administration, developers of the system, more importantly the latter.

# 1.2 Document Conventions

This document follows the IEEE format; bold faced font has been used for emphasis, headings and sub headings. Highlighted words are used in the glossary and italicized text is used in the diagram labelling.

# 1.3 Intended audience and Reading suggestions

This document is intended for all individuals, both the IOS users and IOS mobile applications developers. The intended users mentioned above will be in position to get insights and patterns on the most trending IOS apps based on different features (details) such as size, price, etc.

This software is an IOS mobile application analysis system that analyses the IOS mobile application tools to have an insight and determine the patterns from the Apple app store be able to determine which apps are trending according to user rating.

# References

* SRS template 2016
* R package from github, with devtools::install\_github(“ramamet/applestoreR”)

# 1.5 Product scope

The software that is going to be developed studies the relationship between app details and user ratings.

**Benefits**

It helps Apple developers study the features of the most downloaded apps on the Apple app store and also to increase the user ratings for their apps.

**Objectives**

* To get the top trending apps on IOS app store.
* To increase the user ratings for applications on the Apple IOS app store

# 2. Overall description

In this part, background information about specific requirements of the system will be provided briefly. General issues that affect the product and outline of the functional requirements will be mentioned, too. In short, this section will mainly give information about product perspective, product functions, constraints, assumptions and dependencies.

# 2.1 Product perspective

The system being developed is a new self-contained software product. It is to be developed for all individuals interested in graphical visualisation. It handles various graph formats and supports them.

It is an open source project and has a very active development team to support it and provide feedback to the intended users.

# 2.2 Product functions

The system is going to perform the different functionalities.

* Login a user.
* Display apps and app details.
* Arrange the apps in the order of their creation and update.
* Keep an updated list of the 20 most trending apps.
* Categorise the apps on the system for example Games, Music, Social media.
* Update information like new apps from the apple store.
* Fetch comments (feedback)from the users



Use case diagram

# 2.3 User classes and characteristics

This software will be used by mostly IOS mobile applications developers in order for them to be able analyse how the different application features affect the user rating. This will eventually enable them to wisely decide on which features to give their applications when developing.

# 2.4 Operating environment

The software will be able to run on any personnel computer and smart phone that is connected to the internet.

# 2.5 Design and implementation constraints

The software will only be developed on 2 Windows 10 HP laptops and Toshiba.

Millions of data will be needed to test the software. At this stage developers will need huge amount of disk space and clusters.

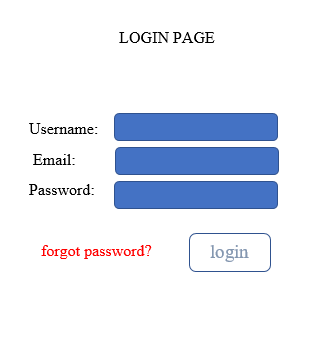
# 2.6 User documentation

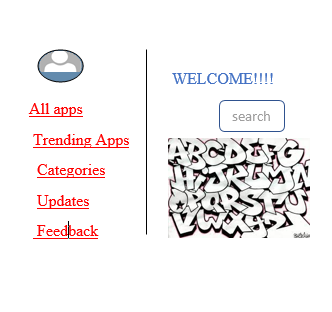
User manuals and video tutorials shall be delivered along with the software to enable the user understand how to use the system.

# 3. External interface requirements

# 3.1 User interfaces

The user will be able to access a login screen with space for username, email address and password. After login, the user will access the home page that has all the functionalities of the system as shown in the diagram.

Login page



Home page

# 3.2 Hardware interfaces

The system should work on any connected devices. The devices should have a limit requirement to make the software to run effectively. The processor speed and internet speed should be high.

# 3.3 Software interfaces

The system will be majorly coded in R with some html and css. The web side will be designed with html and css.

# 3.4 Communication interfaces

The system will be communicating with the users email address, with notifications of updates.

# 4. System features

# 4.1 Get the trending applications

# 4.1.1 Description and policy

The system is able to get ios applications that are trending. The trending applications are based on user ratings which are determined by the users’ experience while using the various application features.

This feature is of high priority because it enables ios application developers to know which applications on the market are most used depending on their ratings

# 4.1.2 Stimulus and Response sequences

The user clicks on the application icon which prompts the system to open the login page.the user then enters their login details, the system verifies the details and if they match, the user is directed to the home page after clicking the login button else the user is told to re enter details. The home page contains links to all applications, trending applications, application categories, updates, and feed back. when the user then clicks on the trending applications link, the system directs some to a list of the 20 most applications.

# 4.1.3 Functional requirement

REQ 1

NAME : listing the most trending applications according to the user rating in the order of highest to lowest.

DESC: in the user enters invalid details during the login, the system is in position to ask the user to enter their details again.

# 4.2 Categorize the apps on the system

# 4.2.1 Description and priority

The system scans the apps on the system and categorizes them in their corresponding groups for example games, music, shopping, and entertainment, among others.

# 4.2.2 Stimulus/ response sequences

The home page contains links to all applications, trending applications, application categories, updates, and feedback. when the user then clicks on the trending applications link, the system directs some to a list of the 20 most applications.

# 4.2.2 Functional requirement

REQ 2

NAME : categorizing the apps on the system.

DESC: in the user enters invalid details during the login, the system is in position to ask the user to enter their details again.

# 5. Other Non-functional Requirements

# 5.1 Performance requirements

The software is only going to operate on personal computers and smartphones. The devices have to be connected to the internet to access the services provided by the system. For a user to go on and use the system, he or she must create an account.

# 5.2 Safety requirements

The user can only have one account therefore an email address can only be associated with one user.

# 5.3 Security requirements

The user of the system is required to reset his or her password and username after a period of 6 months.

# 5.4 Software quality attributes

The product is expected to be available, adaptable, correct, flexible, maintainable, portable, reliable, robust, reusable, testable and usable.

# 5.5 Business rules

Only the administrator is responsible for all updates made to the system.

# 6. Appendix: Glossary

|  |  |
| --- | --- |
| word | meaning |
| app | Application |
| IOS | I phone Operating System |
| IEEE | Institution of Electrical and Electronics Engineers |
| REQ | Requirements |
|  |  |

# 7. Appendix B



Context diagram



Level zero diagram