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](#_Toc518326911)

[1.1 Purpose 4](#_Toc518326912)

[1.2 Document Conventions 5](#_Toc518326913)

[1.3 Intended audience and Reading suggestions 5](#_Toc518326914)

[1.4 References 5](#_Toc518326915)

[1.5 Product scope 5](#_Toc518326916)

[2. Overall description 5](#_Toc518326917)

[2.1 Product perspective 6](#_Toc518326918)

[2.2 Product functions 6](#_Toc518326919)

[2.3 User classes and characteristics 6](#_Toc518326920)

[2.4 Operating environment 6](#_Toc518326921)

[2.5 Design and implementation constraints 6](#_Toc518326922)

[2.6 User documentation 6](#_Toc518326923)

[3. External interface requirements 7](#_Toc518326924)

[3.1 User interfaces 7](#_Toc518326925)

[3.2 Hardware interfaces 7](#_Toc518326926)

[3.3 Software interfaces 7](#_Toc518326927)

[3.4 Communication interfaces 7](#_Toc518326928)

[4. System features 7](#_Toc518326929)

[4.1 Login 7](#_Toc518326930)

[4.1.1 Description and policy 7](#_Toc518326931)

[4.1.2 Stimulus and Response sequences 7](#_Toc518326932)

[4.1.3 Functional requirement 8](#_Toc518326933)

[4.2 Apps Display 8](#_Toc518326934)

[4.2.1 Description and priority 8](#_Toc518326935)

[4.2.2 Stimulus/ response sequence 8](#_Toc518326936)

[4.2.2 Functional requirement 8](#_Toc518326937)

[4.3 Trending 8](#_Toc518326938)

[4.3.1 Description and policy 8](#_Toc518326939)

[4.3.2 Stimulus / Response 8](#_Toc518326940)

[4.3.3 Functional requirement 9](#_Toc518326941)

[4.4 Recommendations 9](#_Toc518326942)

[4.4.1 Description and priority 9](#_Toc518326943)

[4.4.2 Stimulus / Response 9](#_Toc518326944)

[4.4.3 Functional Requirement 9](#_Toc518326945)

[4.5 Information Update 9](#_Toc518326946)

[4.5.1 Description and policy 9](#_Toc518326947)

[4.5.2 Stimuli/ Response 10](#_Toc518326948)

[4.5.3 Functional Requirement 10](#_Toc518326949)

[4.6 Feedback 10](#_Toc518326950)

[4.6.1 Description and policy 10](#_Toc518326951)

[4.6.2 Stimuli/ Response 10](#_Toc518326952)

[4.6.3 Functional requirement 10](#_Toc518326953)

[5. Other Non-functional Requirements 10](#_Toc518326954)

[5.1 Performance requirements 10](#_Toc518326955)

[5.2 Safety requirements 11](#_Toc518326956)

[5.3 Security requirements 11](#_Toc518326957)

[5.4 Software quality attributes 11](#_Toc518326958)

[5.5 Business rules 11](#_Toc518326959)

[6. Appendix: Glossary 12](#_Toc518326960)

**List of figures**

[figure 1: Context diagram 1 7](#_Toc518382414)

[figure 2: Use case diagram 1 8](#_Toc518382556)

[figure 3: Login page 1 9](#_Toc518382847)

[figure 4: Home page 1 10](#_Toc518382397)

[figure 5: level zero diagram 1 16](#_Toc518382281)

# 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 IOS mobile app analysis system is unified for the use of all IOS mobile app users and developers all over the world.

The system requests the users to log in to use the system, either as an administrator or a public user. After authentication, the system allows the user to view the different functionalities which are categorized as, Apps, Trending, Recommendations, Notifications, Comments, Update. The system allows the user to add, edit, and delete information from their profile. The system also allows the users to invite their friends.

The system restricts the users from accessing all the above mentioned features depending on the user’s level of permissions. The administrator has a special login and is the only one allowed to make changes to the information in the system, except for updating profiles, which is for all users.

The system accepts data from the files of previous inventory system and the new data added by the user.

# 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, Entertainment, Shopping, etc.
* Investigate the user interactions with the system.
* Do app recommendations.
* Update information like new apps from the apple store.
* Fetch comments (feedback)from the users

**DIAGRAM SHOWING THE CONTEXT DIAGRAM OF IOS MOBILE APP ANALYSIS SYSTEM**



figure 1: Context diagram 1

**DIAGRAM SHOWING THE USE CASE FOR IOS MOBILE APP ANALYSIS SYSTEM**



figure 2: Use case diagram 1

# 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 Windows 10 laptops. Millions of data will be needed to test the software.

# 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.

**DIAGRAM SHOWING THE LOGIN PAGE FOR IOS MOBILE APP ANALYSIS SYSTEM**

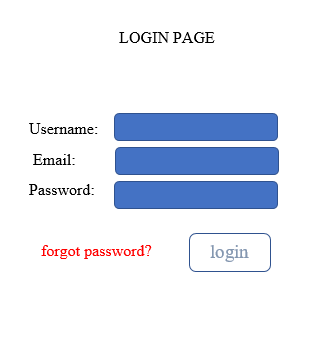


figure 3: Login page 1

**DIAGRAM SHOWING THE HOME PAGE OF THE IOS MOBILE APP ANALYSIS SYSTEM**

# 

figure 4: Home page 1

# 3.2 Hardware interfaces

The system has no hardware interfaces.

# 3.3 Software interfaces

The systems user interface will be integrated with a web browser. The client side gathers information from users, investigates some actions of the users, and provides the connection with the server.

The server side system will hold the entire data in a graph database and must include all functionality to perform operations on this database, receives requests from the clients, evaluate, create and send recommendations.

# 3.4 Communication interfaces

The system requires http to communicate with the server. The system can be configured to be accessed via any available port.

The web based UI is the only means of communication between the user and the system. The system is accessible through all popular well browsers that interact with JSP and HTML pages.

# 4. System features

# 4.1 Login

# 4.1.1 Description and policy

This feature allows the user to enter into the application. The user is required to provide username, password and email address.

# 4.1.2 Stimulus and Response sequences

The user opens his or her browser and searches for the IOS mobile app analysis system. A sign up page will pop on the screen, requiring the user to enter his password, username and email address. After authentication user will have access to main menu. Availability of menu functions depends on user’s level.

# 4.1.3 Functional requirement

**REQ 1**

**NAME**: Login a User

**DESC**: the user enters their password, username, email address. In case these details are incorrect, the user will be denied access and asked to enter correct details, with an option of recovering his details via his email address.

# 4.2 Apps Display

# 4.2.1 Description and priority

This feature contains all the apps and their details for example price, size, among others.

# 4.2.2 Stimulus/ response sequence

Once given access to the home page, the user clicks on the “Apps” link. There the user will see the entire list of apps on the Apple app store. This feature has a search button which enables quick locating of apps. These apps come with a clear description of their details including their ratings.

# 4.2.2 Functional requirement

**REQ 2**

**NAME**: Displaying apps and app details.

**DESC:** the user is able to view all the apps on the Apple app store plus their details

**REQ 3**

**NAME:** Categorize the apps on the system.

**DESC**: The system is able to put all apps in their respective categories, for example Music, Games, Communication, etc.

# 4.3 Trending App

# 4.3.1 Description and policy

This feature contains a list of the twenty most trending apps on the IOS market.

# 4.3.2 Stimulus / Response

The user clicks the “Trending” link. The user will then will access a list of the twenty most trending apps. This list is determined by the system basing on the user ratings of the apps. Take note that the list may change continuously as new apps come up daily with better functionalities.

# 4.3.3 Functional requirement

**REQ 4**

**NAME:** Keeping an updated list of the 20 most trending apps

**DESC:** The system keeps an up to date list of the most trending apps on the store.

# 4.4App Recommendations

# 4.4.1 Description and priority

The feature contains app recommendations for the user following the interactions of the user with the system, especially to with the apps.

# 4.4.2 Stimulus / Response

The user taps the “Recommendations” link. A list of recommended apps will be displayed. The system comes up with these recommendations using Content Based Filtering, CBF. This technique scans through the information on the app and selects those apps with extra functionalities that the user could need rather than on user ratings, that are relevant to the users’ interests. This also helps to discover new apps among over whelming numbers of alternatives.

# 4.4.3 Functional Requirement

**REQ 5**

**NAME:** Investigating user

**DESC**: The system follows the interactions of the user with the system and sends them to the server. The server stores these evaluations.

**REQ 6**

**NAME:** Displaying App recommendations

**DESC:** The server interprets the content of the evaluation and comes up with the appropriate recommendations, which it forwards to the user’s account.

# 4.5 Information Update

# 4.5.1 Description and policy

This feature enables the administrator to add or delete information from the system.

# 4.5.2 Stimuli/ Response

The administrator clicks “Update” on the menu page and adds any information, or deletes information accordingly.

# 4.5.3 Functional Requirement

**REQ 7**

**NAME**: Update information

**DESC**: The administrator adds or deletes information from the system.

# 4.6 Feedback

# 4.6.1 Description and policy

The user is able to make comments following the services provided by the system. The administrator views these comments.

# 4.6.2 Stimuli/ Response

The user clicks “Comments” and a comment bar pops where the user is able to write any comments about anything to do with use of the system.

# 4.6.3 Functional requirement

**REQ 8**

**NAME**: Fetching Feedback

**DESC:** The user is able to make comments.

# 5. Other Non-functional Requirements

# 5.1 Performance requirements

The response time is expected to be fast to increase efficiency, with a maximum waiting time of thirty seconds. This prevents the user from assuming that the system is down.

The system will allow a hundred users at a time. The incoming requests will be queued, until there are slots for them to use the system.

The system should have a very low consumption of power.

The system should be able to perform failure handling that is the system components may fail independently of others so system components must be built so they can handle the failure of other components they depend on.

# 5.2 Safety requirements

The system will be integrated with Data Integrity Gateway tools to perform data cleaning to detect, eliminate and correct all errors and inconsistencies.

# 5.3 Security requirements

The user of the system is required to create an account and also reset his or her password and username frequently to ensure unauthorised access.

# 5.4 Software quality attributes

**Portability**

The system should support new versions of the related browsers. The administrative and server technologies should be standard and supported by most platforms.

**Maintainability**

The system will be well documented to enable proper maintenance and in cases of further development and change of team members, they can still follow up.

**Reliability**

The system should work reliably, with automatic backup and recovery features. In case of unexpected termination of a session, the unsaved data should be recovered without loss and displayed to the respective users.

**Availability**

The entire system should be available round the year, except for a periodic maintenance. The maintenance period should be pre-scheduled and short. The users should be reminded of the unavailability period, well in advance.

**Testability**

The system will be tested with various data sets, and also tested for syntax and semantic errors, which will be debugged to increase efficiency.

# 5.5 Business rules

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

# 6. Appendix: Glossary

|  |  |
| --- | --- |
| Term/ Acronym | Description/ Meaning |
| app | Application |
| IOS | I phone Operating System |
| IEEE | Institution of Electrical and Electronics Engineers |
| REQ | Requirements |
| DESC | Description |

**LEVEL ZERO DIAGRAM FOR THE IOS MOBILE APP ANALYSIS SYSTEM**



figure 5: level zero diagram 1