Software Requirements Specification

for

Language Filter and Abuse Detection System (LFADS)

Version 1.0 approved

Prepared by SEHRISH ASLAM
(EP19101073)
SYEDA FATIMA AHSAN
(EP19101085)
KIRON ISMAIL
(EP19101031)

UBIT(UOK)

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of Language Filter and Abuse Detector System (LFADS). It will explain the objective and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

1.2 Document Conventions

ABBREVIATION	DEFINITION
LFADS	Language Filter and Abuse Detector System
SRS	Software Requirement Specification
DB	Database
SDD	Software Design Description
GUI	Graphical User Interface
API	Application programming interface
JVM	Java Virtual Machine
USB	Universal Serial Bus

1.3 Intended Audience and Reading Suggestions

Users: This document is intended for users and customers to make them insure that this document is well meeting their needs.

Project Manager: Who overlooks the whole project and ensures that it contain all information required for planning the project.

Developers: The project developers will refer to the SRS to make sure that they develop exactly what the customer requires.

Tester: To ensure that the requirements are understandable from functionality point of view so that he can test the software and validate it's working.

Maintenance: who are interested in working on the project by further developing it or fix existing bugs.

The next section, the Overall Description section, of this document gives an overview of the functionality of the product. Followed by Requirements Specification which is written primarily for the developers and describes in technical terms the details of the functionality of the product. In the later part, it describes the non-functional or informal requirements and is used to establish a context for the technical requirements specification in the next chapter. Different sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

1.4 Product Scope

This software system will be a Language and Abuse Detection System for social media user mainly targeting Facebook users. This system will be designed to minimize the use of abusive and slang word and promoting clean language over social platforms by providing tools to assist the users to review their posts, which would otherwise have to be performed manually and is inconvenient when dealing with long posts. More specifically, this system is designed to allow a user to analyze, and generate a report of their latest posts on their Facebook timeline. The software will facilitate user with an option to download the report. The system also contains a relational database containing a list of User Accounts, Posts, and Reports.

1.5 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

2. Overall Description

2.1 Product Perspective

LFADS is abuse or slang language detecting system that automatically analyses users' posts or comments once the token is submitted and generates report which also contain a censored version of the concerned. Anyone who is interested in controlling their hate speech or just wants to know their contents more properly can use them as a means of analysing data. The use of GRAPH API provides the system an access to user's Facebook account. The system is developed to run on various operating systems provided JVM is installed on the system.

2.2 Product Functions

Following are some functions that the system must perform:

Allow Users to create an Account.

Authenticate User when they log-in.

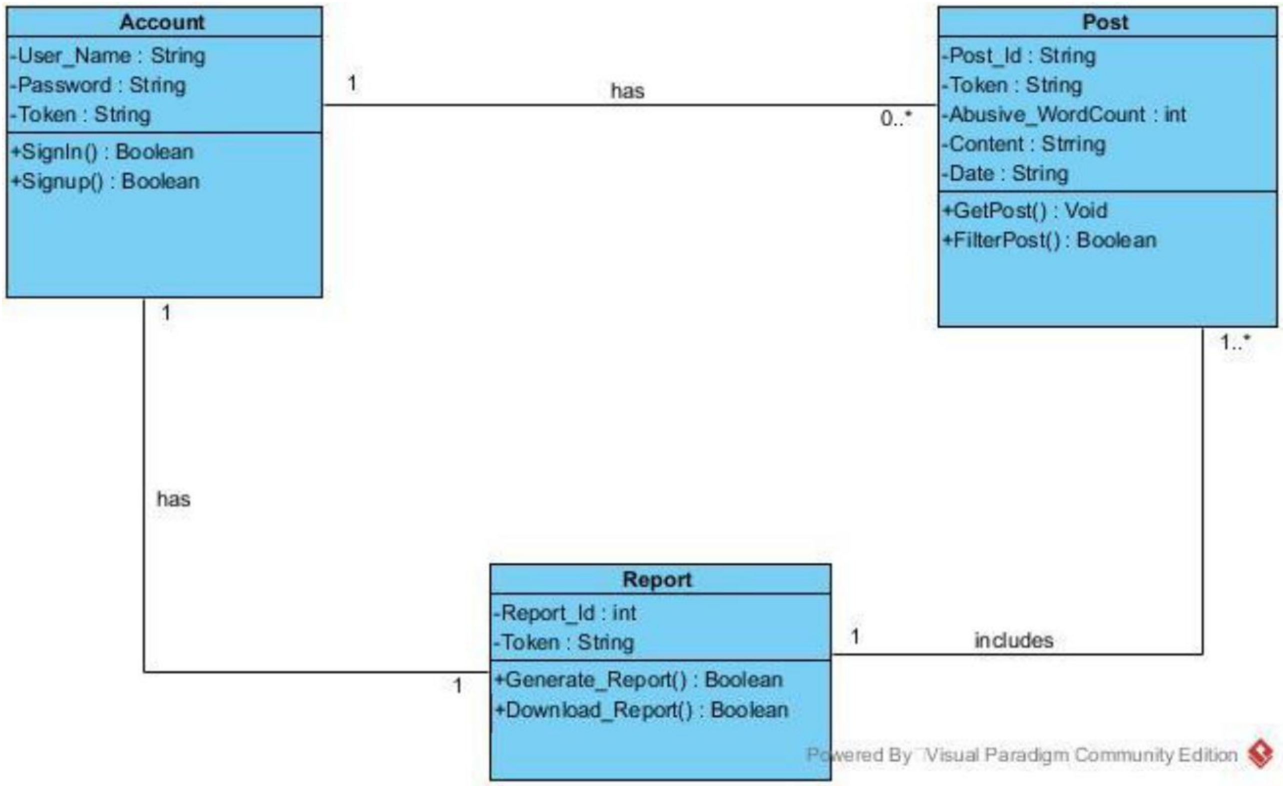
Fetch Posts from Facebook timeline.

Analyze posts.

Generate posts.

Allows user to view and download posts.

The major features of LFADS are as shown in below class diagram:



2.3 User Classes and Characteristics

Mainly, the system targets Facebook users. Users of the system should be able to analyse posts from their timeline and review the analysis. The system will support following user privileges, allowing user to able to:

Create accounts

Generate report

View report

Download Report

With each account, there is a limited number of posts that can be fetched and analysed at a time

2.4 Operating Environment

Operating environment for the LFADS is as listed below:

Operating system: Windows (ver. XP, 7, 8, 10), MAC and LINUX.

Database: JVA'S built-in database

Platform: Java API: Graph API, JDBC API

2.5 Design and Implementation Constraints

LFADS is developed in Java. It uses Java Swing for its visualization engine and has been built on top of the Net Beans Platform. It uses a modular design where every feature is wrapped into

a separate module and the modules depend on each other through well-written APIs. The most important API used is graph API which allows to read and write data to and from the Facebook social graph. And JDBC API which allows access to JAVA's built-in database.

2.6 User Documentation

A user manual will be provided during installation and set process which will guide the in operating the system. Moreover, the system shall provide a 'help' button to open a dialogue box that explains how to proceed.

2.7 Assumptions and Dependencies

LFADS is developed in Java and therefore requires JVM to be installed on the user's system. This applies to Windows and Linux users. On Mac OS X, Java is bundles with the application. Moreover, the system uses GRAPH API to fetch user's posts from Facebook, therefore user must have a Facebook account. Since, GRAPH API is a third party API, so, in the future, any changes in the API might affect the way posts are fetch.

3. External Interface Requirements

3.1 User Interfaces

The front end user interfaces will be developed using Java Graphical User Interface. Following will be the main screens available to the user:

Login/Signup Page.

Account Token Entry Page.

Help Button.

Button for Report Generation.

Report (of posts) displaying Page.

Whereas for backend, *Net beans*' own Java DB will be used to store all the data regarding user information and its account's posts

3.2 Hardware Interfaces

All devices with Windows (ver. XP, 7, 8, 10), MAC or LINUX will be able to run this software.

A browser supporting HTML5 and JavaScript/JQuery will be used to fetch a specific account token from Facebook via Graph API.

3.3 Software Interfaces

Following are the software used for the Language detection and filter software.

Software Used	**Description**
Java	We will be using java language because of its more interactive support to

	implement our project.
Net beans	We will use Net beans as a platform to run our Java Program.
Operating System	We have chosen Windows operating system for its best support and user-friendliness.
Database	We will be using Java’s Net beans inbuilt Java DB using JDBC API because of its convenience and ease of use.

3.4 Communications Interfaces

This project will be using web browser for fetching data from Facebook, hence supporting all types of web browsers and using communication standard HTTP**.** We are using simple electronic forms for Login and Signup and JDBC for communication with the database.

4. System Features

This section demonstrates LFADS’ most prominent features and explains how they can be used and the results they will give back to the user.

4.1 Sign up

4.1.1 Description and Priority

Priority Level: Medium
The second priority goes to sign up form where users’ information is recorded and stored in database. This feature is the source of account generation any user has to perform when they first use the system. In sign up form there will be many mandatory fields required for users to fill their information in. There might be facility of optional field or to select their choice from multiple options. After filling the form, user account will be created.

4.1.2 Stimulus/Response Sequences

All mandatory fields have to be filled after which ‘sign up’ button is to be pressed.

4.1.3 Functional Requirements

REQ-1: The all associated fields are visible to the user. REQ-2: No
mandatory field is left empty.
REQ-3: The user must be able to create an account.

4.2 Log in

4.1.1 Description and Priority

Priority Level: Medium

The user's ability to access the system's tools depends on this features. After the system starts, there will be login form. Through this form only authenticated users can login the system by entering their username and password. Without logging in, the user cannot avail the system.

4.1.2 Stimulus/Response Sequences

User has to fill to fields, that is, username and password. After which a 'Sign in' button has to be pressed.

4.1.3 Functional Requirements

REQ-1: The user must be able to log into his/her account. REQ-2: User must be authenticated properly before entering account. REQ-3: A specific user entry always leads to a specific account only. That is, no user can user into another user's account using his/her username and password.

4.3 Report Generation

4.1.1 Description and Priority

Priority Level: High

This feature one of the most important feature which circles around the main objective of the system. This feature fetches user's posts, analyse it and generate a detailed report about it. The report may contain report number, creation date, number of abusive words found and a censored version of the post.

4.1.2 Stimulus/Response Sequences

User enters token in the 'token field' and clicks on the 'Generate Report Button'.

4.1.3 Functional Requirements

REQ-1: The token provided by the user must link the system to the respective Facebook account.

REQ-2: The post must be filtered out properly on the basis of its content.

4.4 Download Report

4.1.1 Description and Priority

Priority Level: Low
This feature allows users to download generated report.

4.1.2 Stimulus/Response Sequences

User clicks on the ‘Download Report Button’.

4.1.3 Functional Requirements

REQ-1: A report is downloaded on the user’s pc.
REQ-2: The report must be of either .txt format or .docx format.

5. Other Non-functional Requirements

5.1 Performance Requirements

The system will be able to support 5 simultaneous users.

The mean time to generate the report will not take more than 1 second.

The mean time to download the report will not take more than 3 seconds. The user will be Signed Up/Signed In within 0.5 seconds.

Redundant information will not be stored in the database hence minimizing the fetch time to not more than 0.7 seconds.

5.2 Safety Requirements

We will back up the users’ data to a secondary storage (e.g. USB) in case there is an extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash or system crash. This will help us to retrieve the data, otherwise lost.

5.3 Security Requirements

Every user will have access to only its own information/data.

No sensitive information of a user including its account’s token will be breached by any other user.

A user will only be able to download its own report.

Every user needs to sign into the program, hence each session is password protected.

5.4 Software Quality Attributes

AVAILABILITY: The system will be available at all times to fetch and filter posts.

CORRECTNESS: The system shall run the algorithm correctly and filter only abusive words.

MAINTAINABILITY: The system will be expanded to accommodate more users simultaneously.

USABILITY: The system must analyse maximum accounts for filtration and in minimum time.

HELP: The system shall provide a 'help' button to open a dialogue box that explains how to proceed. This help page will be accessible once user is signed in

5.5 Business Rules

User will have the role to generate and enter token itself and then wait for the analyzed result upon requesting. This token will have to be fetched via Graph API which will connect the user to its Facebook account. User can then download the generated report. All of this process must satisfy usability requirement and give ease of access and high response time to the user.

6. Other Requirements

Some other requirements are as follow:

Software might be expanded to incorporate other social media platforms as well.(like Instagram etc.).

The software should be able to deploy on partial and complete infrastructure. The software should be delivered with complete source code.

The software will extend itself to provide a web interface to the user to generate the token within the system.

The software should be able to block an account if certain number of abusive posts is reached.

The software should be delivered with proper manual and documentation.

Appendix A: Glossary

Words	Meanings
Abuse	Foul/ Inappropriate Language.
API	Application Programming Interface to a software component or system defining how other components/systems can use it.
Availability	System is available to the user when it wants to use it.
Context	The circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood.
Constraint	Limitation.
Deployment	The action of bringing the system into use.
Functional Requirement	Requirement that defines the function or behavior of the system.
Graph API	Facebook API used to get access to a particular account and it’s contents.
Implementation	The process of putting the system design into execution.
Maintainability	Process of maintaining the system to either extend it, to handle errors and exceptions or to maximize its efficiency etc.
Non-Functional Requirement	Describes the general characteristics of a system.
SDD	Software Design Description document is a written description of a software product, that a software designer writes in order to give a software development team overall guidance to the architecture of the software project.

Scope	Scope is the defined features and functions of a product, or the scope of work needed to finish a project. Scope involves getting information required to start a project, and the features the product would have that would meet its stakeholders requirements.
SRS	A software requirements specification is a description of a software system to be developed. It is modeled after business requirements specification, also known as a stakeholder requirements specification.
Stakeholder	Anyone effected directly or indirectly by the system.
Usability	Usability can be described as the capacity of a system to provide a condition for its users to perform the tasks safely, effectively, and efficiently while enjoying the experience.
URD	User Requirements Document that specifies what the user expects the software to be able to do.