Software Requirements Specification

for

Bona fide

The Plagiarism Checker

Version 4.0 approved

Prepared by Ayush Kumar and Aadarsh Sahoo

Department of Computer Science & Engineering

Indian Institute of Technology Kharagpur

8th February 2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Aadarsh&Ayush	15/02/19	Appended the Use-Case Diagram.	2.0
Aadarsh&Ayush	13/03/19	Appended the Sequence & Class Diagrams.	3.0
Aadarsh&Ayush	12/04/19	Added some extra requirements & specifications	4.0

1. Introduction

1.1 Purpose and Need

Plagiarism is the "wrongful appropriation" and "stealing and publication" of another author's "language, thoughts, ideas, or expressions" and the representation of them as one's own original work. Plagiarism is a form of academic dishonesty or we can say that it's “Cheating”. With the rise and spread of the Internet and the development of advanced search engines, Plagiarism has become a very serious issue in today's educational system and professional environment.

A lot of Surveys were conducted all around the world and it was found that a considerable percentage of High School and College Students admitted having cheated on their work one or more times in some form. Detailed Statistics of the surveys could be found at .

The issues mentioned above calls for a solution. The detection of Plagiarism is near to impossible for a human being because of the enormous amount of data and resources available all around, this gave birth to the class of software known as Plagiarism Detection Softwares. The main goal of these systems is to promote and sustain the ‘value’ of ‘Intellectual Property’ and ‘Originality in Idea and Expression’.

1.2 Addressing the Need

The need can be addressed in several ways. The solution we are using in developing our software is by using the internet to search for the ‘matches’ between the sentences present in the document or text with the content available on the internet. This can be achieved by extracting the input text from the document and dividing them into groups/sentences, which could be searched in an advanced search engine like Google Search. Google Search with its extremely good PageRank Algorithm shows the possible websites with the searched content in an appropriate order. Now each website can be visited in order, for comparing the searched content and detecting any plagiarism involved. The output could be presented to the user with the addresses of the possible sources from where the information may be copied. The percentage of plagiarism present in a document could also be calculated and presented to the user. A ‘threshold plagiarism percentage value’ must be given by the user so as to classify the analyzed document as ‘Plagiarised’ or not.

1.3 Prospective Users or Intended Audience

This application software mainly targets Academia for its prospective users, who may use this to detect academic dishonesty and copying. The application can be used by anyone(e.g. a verification officer) who is in a position to evaluate or screen any kind of documents or publications, to determine the ‘authenticity’ of the submitted document as per the guidelines for the user.

1.4 Issues and Challenges to overcome

The issues and challenges that are going to arise while the development process of this software can be identified if we properly analyze the very basic principle which the software uses to detect plagiarism. Plagiarism detectors don't actually detect plagiarism, what they actually do is detect sections of identical texts i.e. it can only detect copying or similar phrases. So the following would arise as the prospective problems:

Synonym Matching: The person may copy from a source and change some of the words to their corresponding synonyms to get themselves screened from the application software.

Non-Verbatim Plagiarism: The user may rewrite, translate or otherwise redraft the content from the source and deceive the software system. This problem arises because Plagiarism detectors analyze the words, they don't analyze the content i.e. it can't see if you copied the idea or information even if you didn't copy the words. So this may let go of some serious plagiarised content undetected.

Common Phrasing: The document or content being analyzed is very likely to contain many common phrases in the English language, which may be reported by the software system as a match even though that might be just a coincidence. So we need to implement an Intelligent comparison system which is able to overcome the problem of common phrasing.

1.5 References

https://www.plagiarism.org/article/what-is-plagiarism
https://www.plagiarism.org/article/plagiarism-facts-and-stats
https://en.wikipedia.org/wiki/Plagiarism

2. Work Plan for the Project

2.1 Timeline

Week	Plan
Week-1	Preparation of SRS, RAS and FS related documents.
Week-2	Designing the Structure to write the program for the application software.
Week-3	Designing the Graphical User Interface for the application software.
Week-4	Designing the Algorithm.

Week-5	Designing the complete Backend of the application software.
Week-6	Testing of the Application and Error Correction.
Week-7	Refining the GUI and making it look better and more user friendly.
Week-8	Final testing

3. Functional Requirements

3.1 Application Startup

When the user opens the application the logo of the application must be displayed in the screen for 2 seconds with a welcome message and then proceed to the user login/sign-up window.
Input : When application icon is clicked.
Output : Application Logo on the Screen with a Welcome Message and then proceed to login/sign-up window.

3.2 User Registration

An User must be able to register for an account for the application after the user has provided the required credentials like email-id, name, password, mobile no., and IP Address(determined automatically).
Input : Details and Credentials of the user.
Output : A message notifying either successful or unsuccessful sign-up with reasons.

3.2.1 Username Verification

If the Username used in the sign-up process already exists with another user, there should be a prompt for the user to notify him/her about that.
Input : Username provided during registration.
Output : A message stating ‘Username already exists!’ if there is an account with the same Username.

3.2.2 Image Upload**

The user is asked to upload an image for the user profile. This step may be skipped by the user.
Input : Success message from Email Verification process.
Output : Ask to upload an Image if the user chooses to do so, else continue to the main window.

3.3 User Login

Allows the user to login to the user account whenever the user enters the username and password and allows the user to access the application software.
Input : Username and password by the user.
Output : Successful login on email-password matching, else an Error Message.

3.3.1 User Not Found

If Username entered by the user is not found on the server for login then there is a prompt to notify the user that the Username was not found and the user is asked to either sign-up or re-enter the email address.
Input : Email address from login window.
Output : Message of 'User not found!...Try Again' or redirect to the Registration Window option is shown.

3.3.2 Username-Password Mismatch

If the password entered by the user while login is not correct, the user is prompted about email-password mismatch and is asked to enter the password again.
Input : Username and password from login window.
Output : A message of 'Username-password mismatch...Enter password again!' if the password doesn't match with the email address entered.

3.4 Guest Login

If the user doesn't want to go through the login process and just want to directly use our software then the user can directly visit the session window on-click.
Input : Click on the Guest Login option.
Output : Open the Session Window.

3.5 Naming a Session

The user is asked to name the session once it is logged in as it will be saved in the session history feature.
Input : Title of the Session by the user.
Output : Stores the title of the search result and the date for future reference with no immediate message to the user.

3.6 Text Input Function

User is allowed to type any content which he/she would like to check for plagiarism and the entered text is sent to the temporary storage. This also shows the word count of the content entered.

Input : Content typed by the user using the keyboard to be checked for plagiarism
Output : Sends the text to the temporary memory and the Word Count of the content is displayed.

3.6.1 Word Limit Exceeded

If the word count of the entered text or the text in the document file exceeds 1000 then the user is prompted to reduce the word count to 1000 or less and then enter/upload again.
Note: This function has been kept with view to the speed of the application to show the result.
Input : Word Count from the entered text or from the document uploaded.
Output : Error message of word limit exceeded and asks the user to enter or upload the content again.

3.7 Upload File Function

Uploads the file from the user's computer when chosen by the user.
Input : Click on upload file button
Output : Ask the user to browse and select and upload the file to the temporary storage.

3.7.1 File Format Mismatch

If the file uploaded is in a format other than '.docx' or '.txt', user is prompted about it and asked to upload the file in the compatible format only.
Input : File format of the document uploaded in the upload file function.
Output : Error message if file format does not match the compatible one.

3.7.2 World Limit Exceeded

If the word count of the entered text or the text in the document file exceeds 1000 then the user is prompted to reduce the word count to 1000 or less and then enter/upload again.
Note: This function has been kept with view to the speed of the application to show the result.
Input : Word Count from the entered text or from the document uploaded.
Output : Error message of word limit exceeded and asks the user to enter or upload the content again.

3.8 Extract Function

After the user has entered or uploaded the content this function should separate/divide the content into divisions/sentences to make them ready to be searched in the search engine for matching and saves it in a separate file.
Input : Content Entered/Uploaded by the user.
Output : Sentences extracted and saves them in a separate file.

3.9 Search in the Search Engine Function

This function one-by-one takes the extracted sentences and searches them in the Google Search and loads the search engine result.
Input : Extracted Sentences.
Output : Search Engine Results.

3.10 Web Page Loader and Parser

This function one-by-one opens the web pages loaded by the search engine and each page is parsed and the text present in it is extracted from its HTML code and the text is Isolated.
Note: This function only checks the first 5 search results only for speed and time factors.
Input : Search result from the search engine.
Output : Extracted text from each web page.

3.11 The Sentence Searching Algorithm

This function searches the extracted text from the user content in the extracted text from the web page loader and parser. If any match is found then the address of the web page and the corresponding sentence is saved and noted in a separate file, else it just repeats the process for the rest of the sentences.
Input : Extracted text from the user as well as from the web page loader and parser
Output : Saves the web page address and the corresponding statement if a match is found in a separate file, else continues with the rest of the sentences.

3.12 Sentence-wise Result Function

It goes through the file created by the sentence searching algorithm, compares with the file created by the extract function and displays all the sentences along side with the status that whether it is a copied one or unique in the output box.
Input : The file created by the sentence search algorithm.
Output : Displays sentence wise result in the output box.

3.13 Matched Sources Display Function

User is shown the possible sources from where the content might be copied along with its website address.
Input : The file created by the sentence search algorithm.
Output : A list of all the web page addresses present in the file.

3.14 Final Result/Summary Display Function

This function calculates the percentage of plagiarism and uniqueness of the document/content of the user from the file created by the sentence searching algorithm and the file created by the

extract function.
Input : The files created by the sentence searching algorithm and the extract function.
Output : Summary of the user content: percentage of plagiarism and uniqueness present.

3.15 Report Download Function**

The user can download an analysis report of the content/document he/she checked for using the software. The document will contain the sentence wise result, matched source websites, percentage of plagiarism along with the title of the session in pdf format.
Input : Click on the Download Report button.
Output : A report having the details of the content searched for plagiarism like sentence wise result and matched sources to be downloaded in pdf format with the name of the file same as that of the session.

3.16 New Search Function

This option makes the application ready to conduct a search once again.
Input : Click on the New Search Button.
Output : Deletes all the temporary files created in the memory and takes the user to a Fresh Input Window with a new Session Name Prompt.

3.17 Session History Display Function**

Allows the user to see all the session details which were conducted by him/her till date.
Input : Click on the See Session History Button.
Output : Session History of the user with the Session Names and the Summary of each session.

3.18 Logout Function

The logout button which upon clicked asks the user if he/she wants to logout of the session. Upon approving yes, the user is logged out of the session with all the user data deleted except the session history and the window for login or sign-up appears.
Input : Click on logout button.
Output : Log-out from the software session upon ‘yes’ approval by the user, along with saving the session history and deleting the rest of the data.

**These are planned for Future Implementation.

4. Non-Functional Requirements

4.1 Performance Requirements

4.1.1 Response Time

The software should return the desired output after analysis in a reasonable amount of time, we keep that time as a maximum of 20 seconds.

4.1.2 System Dependability

It determines the fault tolerance of the system. If the system loses the connection to the Internet or the system gets some strange or invalid input or the system faces any random failure,then the user must be informed about it.

4.1.3 Prominent Results

The result displayed must be prominent and clearly specify the amount of plagiarism and the amount of unique content found in the input document.

4.2 Security Requirements

4.2.1 Secure Search

The software should search the internet for the content securely giving utmost priority to the privacy of the user's data.

4.2.2 Temporary Storage of the Extracted Text

The extracted text from the input document must be stored temporarily and securely and should be permanently deleted after the plagiarism detection session has been over.

4.2.3 User Profile should be Unique

Any user account registered for the software should be unique and no fake accounts should be present. Only one user account per IP Address should be allowed.

4.2.4 Secure Login

The system should be secure from malicious or forced login to access the software.

4.3 Design Constraints

4.3.1 Hard Drive Space

The software should not take more than 20MB of Hard Drive Space.

4.4 Software Quality Attributes

4.4.1 Reliability

The system should be reliable i.e. it should give right and accurate results for each session held by the user.

4.4.2 Internet Connectivity
The application must be connected to Internet Connection for performing the Login activities and Sentence Search for the analysis of the document.

4.4.3 Maintainability
The application should be easy to extend. The code should be written in such a way that it favors the implementation of new functions.

4.4.4 Portability
The application should be portable to different platforms i.e. it should be adaptable in different platforms.

5. External Interface/Environment Requirements

5.1 Hardware Interfaces

This application software does not have any designated hardware so no direct hardware interfaces are required.

5.2 Software Interfaces

5.2.1 Java

Java has been chosen as the programming language for the development of this application software. Java is a programming language and computing platform first released by Sun Microsystems in 1995. It is the underlying technology that powers state-of-the-art programs including utilities, games, and business applications. Java runs on more than 850 million personal computers worldwide, and on billions of devices worldwide, including mobile and TV devices (Oracle Technology Network, 2010), so this application is intended to run on all platforms that support JAVA without the need of recompilation as JAVA is intended to let developers **write once, run everywhere**. The version which has been chosen for this project is JRE System 1.7. The version used contains important enhancements to improve performance, stability and security of the Java applications. Java is known for his large number of libraries. Indeed, Sun provides a large number of frameworks and API in order to allow a lot of diversified uses. This is why Java was probably the best choice, at least the most suitable language, for the implementation of this project.

5.2.2 Java Libraries

Many Java Libraries have been used for execution of different and a variety of functional requirements for this application software.

5.2.3 Python Libraries

Python Libraries such as google and fuzzywuzzy are used for the development of the backend of the project which completely deals with web related activities and string matching algorithms.

5.2.4 Google Search

Google Search (Google Web Search) is one of the advanced and fastest search engines presently, which is owned by **Google Inc**. Google handles around 3.5 billion search queries of different types per day. It scans websites having a particular keyword to be searched and indexes the website as per the queries to the website. The PageRank Algorithm used by Google Search performs exceptionally well which would help us get our desired results very fast.
We will be using Google Search for searching the Extracted Text in the Internet and to look for the possible webpages where matches in the contents may be found.

6. Attractive System Features

This section notes down all the attractive system features we provide for our system software which would allow the user to give our software the highest preference among all our competitors.

6.1 We do not Save your Precious Data

We respect user data. Our software is designed in such a way that after analyzing the input text or document from the user, it permanently deletes the extracted text from all possible locations which were used for performing the search on the Internet.

6.2 We provide a FREE Trial with NO feature restriction!**

We provide a FREE Trial of 3 Plagiarism detection sessions to every registered user. This allows the user to experience the Very Accurate and Error Free Plagiarism Detection Sessions with all the features made available by our Application Software for FREE! The user can continue with the experience by subscribing for our premium membership for unlimited quality sessions.

6.3 Easy to Use

We have designed the application software interface in such a way that it is convenient to use by any user. It is very simple and robust at the same time. The simple User Interface allows the user to comfortably use the software without going through any user manual. The function of each option displayed in the interface is also mentioned aside concisely for better user experience.

6.4 Personalised Sessions with User Accounts**

We enable access to our application software only to a registered user. The user can register and create an account for him/her. This allows the user to experience personalized sessions as per the need of the user and also helps in secure text extraction and search.

6.5 Save your Session History**

We allow you to save your session history with only two features i.e. the name of the session provided by the user and the plagiarism percentage result of the input document, Nothing Else (No Other Data for Security Purposes), which can be very helpful for the user to refer in the future.

6.6 You may contact us for Advertisements

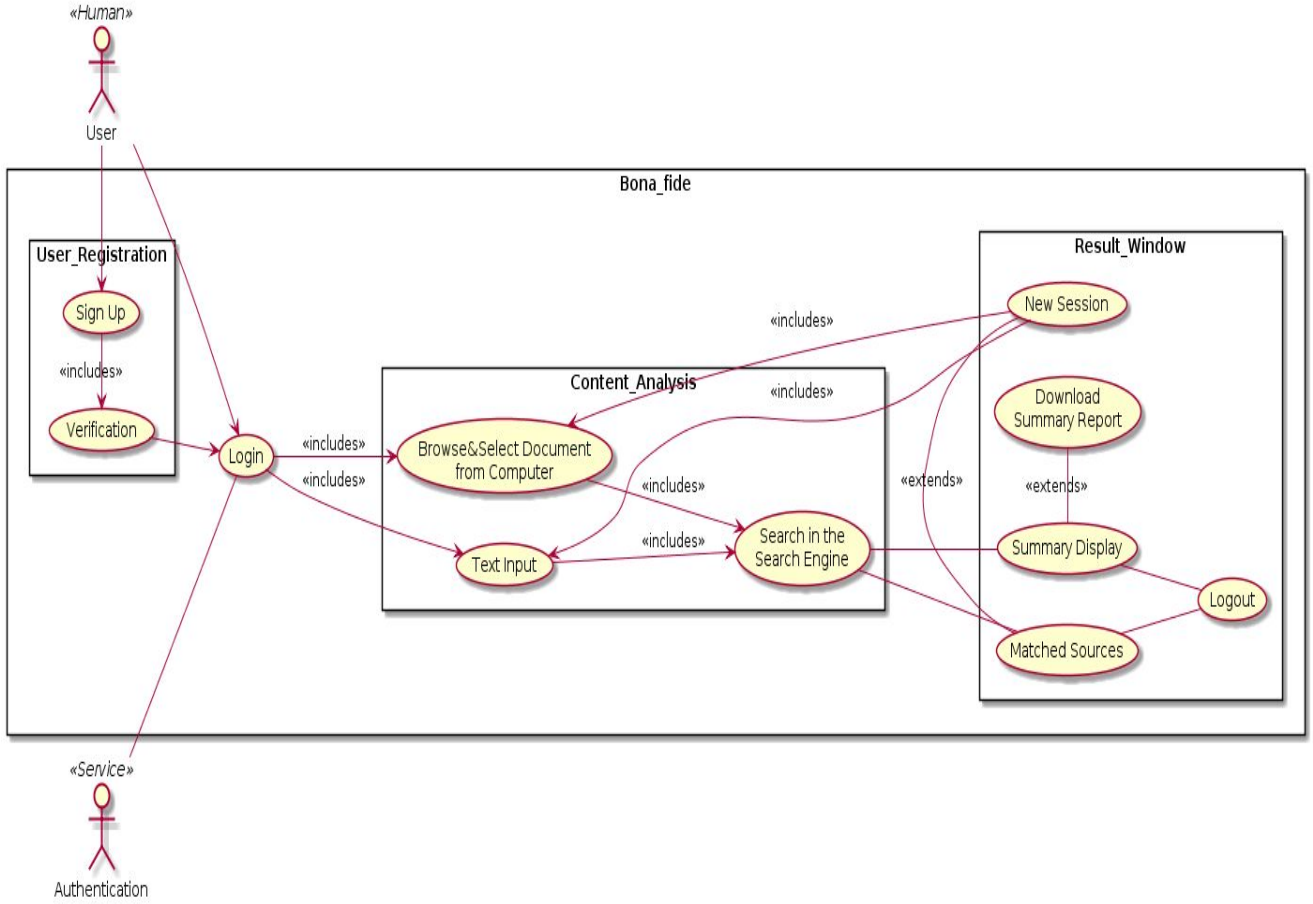
We allow the users to contact us if they wish to use our platform for their Advertisements.

**Planned to be Implemented in Future.

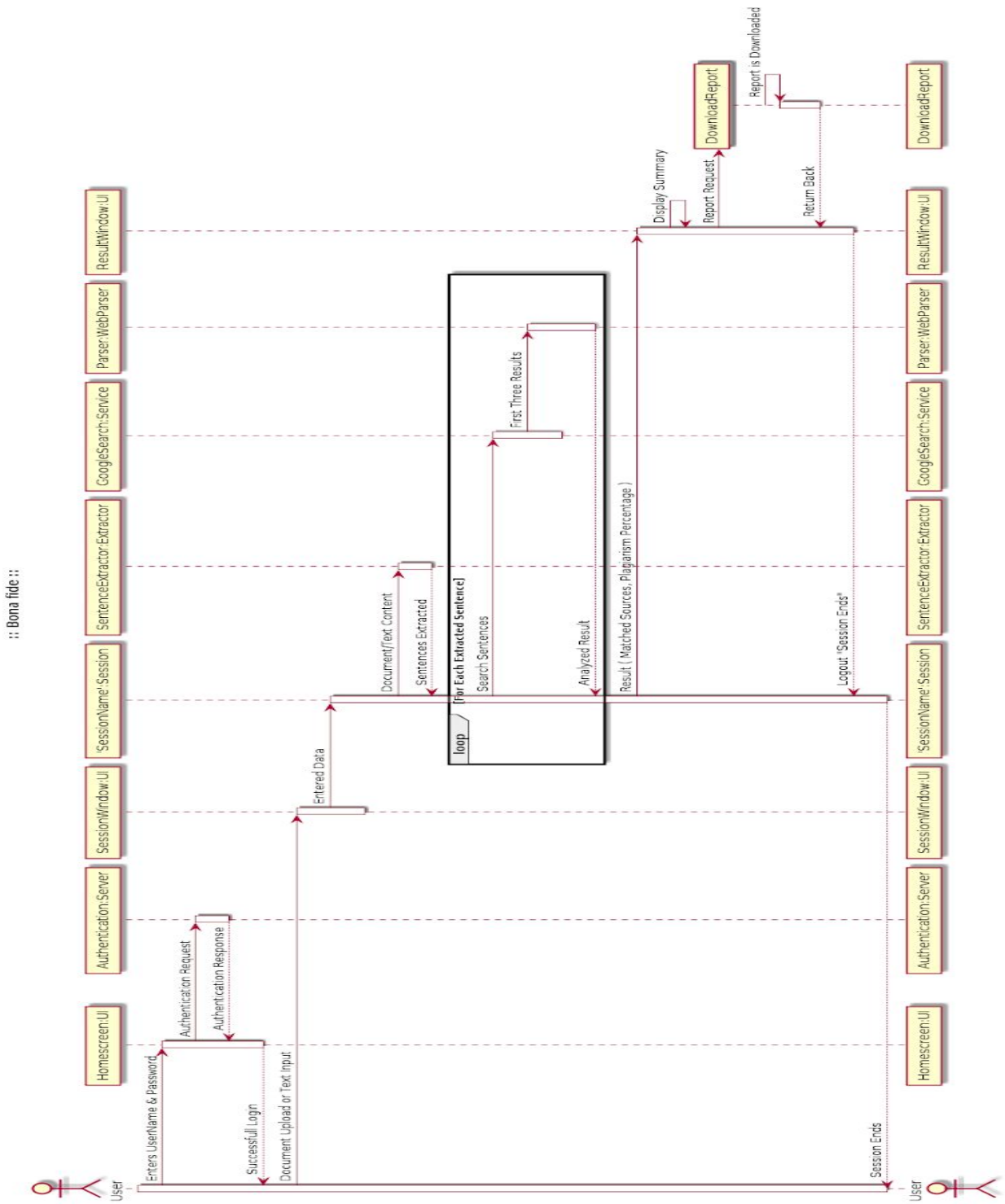
7. Estimated Cost for the Product

Almost all the Software Libraries and Resources we are going to use for the development of this application software are open-source so almost all of them are available for use free of cost. But it takes time to build such software and obviously, time is precious and in future we may need some funding for the maintenance of the application software. We will be charging a minimal amount of INR 2000 as a membership fee for each registered user, provided the user wants to subscribe for the premium version of the application in which the user is allowed to have any number of free Plagiarism Detection Sessions. So as an overall view, this software is Free. Our Goal is to provide the user with good and quality software which the user uses for the betterment of the society.

8. Use-Case Diagram



9. Sequence Diagram



10. Class Diagram