

Dr. BR Ambedkar National Institute of Technology Jalandhar

SOFTWARE REQUIREMENTS SPECIFICATIONS DOCUMENT

**Software Engineering Laboratory**

Notesify

A Student’s Assistant in Learning

Rajnish Kumar

17103069

Rubal Khehra

17103074

Rupinderjit Kaur

17103075

Contents

[**Document Revision History** 2](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050965)

[**1.** **The Software Requirements Specifications Outline** 2](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050966)

[**1.1** **Purpose and Scope** 2](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050967)

[**1.2** **Document Conventions** 2](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050968)

[**1.3** **Definitions, Acronyms, and Abbreviations** 3](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050969)

[**Definitions** 3](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050970)

[**Acronyms and Abbreviations** 3](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050971)

[**1.4** **Intended Audience and Reading Suggestions (Overview)** 4](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050972)

[**1.5** **References** 4](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050973)

[**2.** **Overall Description** 4](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050974)

[**2.1** **Product Perspective** 4](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050975)

[**2.2** **Product Functions** 5](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050976)

[**2.3** **Operating Environments** 5](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050977)

[**2.4** **User Documentation** 5](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050978)

[**2.5** **General Constraints, Assumptions and Dependencies** 6](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050979)

[**3.** **Specific Requirements** 6](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050980)

[**3.1** **External Interface Requirements** 6](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050981)

[**3.1.1** **User Interfaces** 6](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050982)

[**3.1.2** **Hardware Interfaces** 8](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050983)

[**3.1.3** **Software Interfaces** 8](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050984)

[**3.1.4** **Communication Interfaces** 9](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050985)

[**3.2** **Functional Requirements** 9](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050986)

[**3.3** **Non-Functional Requirements** 12](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050987)

[**3.4** **Design and Implementation Constraints** 12](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050988)

[**3.5** **Other Requirements** 12](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050989)

[**4.** **To Be Determined List** 12](file:///C:\Users\hp\Downloads\Notesify%20SRSupdated.docx#_Toc19050990)

# **Document Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Author** | **Dates** | **Status and Description** |
| 1 | Rupinder, Rubal & Rajnish | 10-09-2019 | Created First Half of SRS |
| 2 | Rupinder, Rubal & Rajnish | 24-09-2019 | Completed SRS Version-1 |

# **The Software Requirements Specifications Outline**

## **Purpose and Scope**

‘Notesify’ is an *app* that aims to provide a common platform for current students, their seniors, alumni, and teachers to interact with each other and share their knowledge. Its purpose is to provide free study material and guidance to students readily at their doorsteps.

The audience of a *doc.* can also testify its quality by rating it and commenting on it. It will not only testify the genuineness of a *doc.* but also promote it for others to read. Users can also elaborate or correct a *doc.* (if it is modifiable as chosen by the owner of the document). This way a reader can know which *docs.* are highly used and suggested by others.

It provides a way to collect *info.* about any emerging topics and *tech.* as conveyed by the volunteering mentors currently working in the industry who have joined our interface. Therefore, a student can remain updated about the evolving *tech.* standards in the industry.

The *app* will also provide a common forum where anyone can ask a *que.* and anyone can answer it.

This *app* is as beneficial to a mentor as to the students because it provides the mentor with an audience who rates, supports and promotes his/her study material. It will not only boost his/her confidence but also allows him/her to improve his/her knowledge as he/she tries to create higher quality *docs.* and answer others’ *que.*

Major Goals of the Project-

* Easy access to useful and relevant *info*.
* Availability of material within seconds as soon as the mentor uploads it.
* Readily resolve *que.* of a user.
* To provide the correct *info.* the users on easy terms.
* Platform to provide an audience to mentor to promote his/her ideas.

## **Document Conventions**

|  |  |
| --- | --- |
| **Attributes** | **Description** |
| In Single Quotes | Name of the *App.* |
| Font Size 16 | First Heading |
| Font Size 13 | Second Heading |
| Font Size 12 | Third Heading |
| Font Size 11 and Italic | Fourth Heading |
| Italic | Abbreviations |
| Bold | Main Points |
| Superscript Number | Number of a *Ref.* |

## **Definitions, Acronyms, and Abbreviations**

### **Definitions**

* Student: The user for whose assistance this *app* was developed. It is a user who will only have access to read, copy and save *info*. He/she can’t upload any data, to upload data he has to change his status to mentor. He further has the rights to rate, comment and give feedback. He could also start a chat with any mentor.
* Mentor: This is the most important user or we can say the backbone of our *app.* Without him/her, the functionalities of the *app* are at stake. As mentors only can provide the study material for students to access. Mentors have been given all rights as given to students. In addition to that he/she has the ability to upload any kind of material, he/she wants to share.
* Volunteering Mentor: It is the mentor who is volunteering to resolve *que*. of the users. This mentor has his/her chatbox functional through which he/she can interact with the students to answer their *que.* They will receive notifications whenever a student asks them a *que.* through their chatbox.
* Feedback collector: These are the developers of the *app* to whom the feedback or *que.* will be reported. They have the right to read the *que.* and respond to them.
* Genuine *Info.*: It includes those *docs.* and links that contain *info.* that is verified by other users. Genuineness od a *doc.* can be judged by the rating and comments on the document. If a *doc.* is not found correct, it can be reported and the *admins* will take care of it.
* Genuine user: It means that if this user(mentor) provides any content about a topic then that content is supposed to be genuine. Genuineness of a mentor is judged by the average rating of all his/her uploaded contents.
* Developer/*admin*s: Persons who are the creators of this *app* and manage it.

### **Acronyms and Abbreviations**

|  |  |
| --- | --- |
| **Acronyms and Abbreviations** | **Description** |
| Admin | Administrator |
| API | Application Programming Interface |
| App | Android Application |
| CR | Class Representative |
| Docs. | Documents |
| DB | Database |
| DBMS | Database Management System |
| ER | Entity-Relationship |
| FAQ | Frequently Asked Questions |
| FSR | Feasibility Study Report |
| GUI | Graphical User Interface |
| Info. | Information |
| JDBC | Java Database Connectivity |
| JDK | Java Development Tool Kit |
| min | Minimum |
| PDF | Portable Document Format |
| PPT | PowerPoint |
| Que. | Questions or Queries |
| Ref. | References |
| Reqms. | Requirements |
| SQL | Structured Query Language |
| SRS | Software Requirements Specifications Documents |
| Tech. | Technologies |
| XML | Extensible Mark-up Language |

## **Intended Audience and Reading Suggestions (Overview)**

The intended readers of the document would be

* Developers
* Project Managers
* Users
* Testers
* Documentation Writers
* Client: - The general user of the app with restricted privileges.
* Group Admin/Seniors: - Special users heading a group with special privileges.

## **References**

1. <https://drive.google.com/drive/u/0/folders/0B-GCOGyxmJAtV3FXdmgxRXlnb3c>
2. Feasibility Study Report
3. <https://www.slideshare.net/harshavi57/srs-example-16099692>
4. <https://www.slideshare.net/KrishnasaiGudavalli/software-requirements-specification-17173967>

# **Overall Description**

## **Product Perspective**

‘Notesify’ is inspired by an existing documents-providing facility available to the students in our college. We are provided with a Google Drive Link**¹** created by the seniors of our department where they have uploaded some relevant books' pdfs and manuals. But this Google Drive Link**¹** lacks many facilities that we aim to provide through our project. Though it provides some important books and notes of each subject but it is not interactive. Through ‘Notesify’ users can rate and comment on any document and also ask *que.* from their mentors and seniors. In addition, ‘Notesify’ provides many other facilities for the students to make their lives much easier.

**EXISTING SYSTEM**

* Google Drive Link**1** to provide curricular related notes, books *pdfs*, lab manuals, etc.
* Available to all irrespective of any criteria.

**DRAWBACKS**

* Once created, not updated
* No facility to eliminate irrelevant information
* Other than developer, no other person can contribute to the application
* No online communication facility
* No feedback system

**PROPOSED SYSTEM**

* Android *app*
* Includes topics other than curriculum
* Online chatbox
* Rating and commenting Facility
* Search box for easier searching
* Online registration form

**OUR PLANS**

* To provide the facility of online availability of material within seconds
* To provide mentors to students
* To provide the facility of online communication with peers and mentors
* To provide the facility to promote one’s valuable *info.*
* To provide authenticated access to *docs*
* To provide genuine *info.* through rating (likes) and comments
* To provide a platform to ask *que.* openly without any restriction
* Any number of users can access a *doc.* simultaneously
* To provide a platform to help others

## **Product Functions**

* It provides an easy to use *GUI*.
* Allows users to log in to their accounts.
* It allows the users to view already posted materials by other users classified according to departments and subjects for easier searching and access. Search Bar is also available to search about any *doc*. randomly. There are no eligibility criteria to read any *doc*.
* Allows a user to post-study materials in the form of *pdf, doc* files, *URLs, ppts*, etc, He/she can choose to make his/her *doc.* modifiable by others or not.
* It allows a user to search for other people working in the fields of his/her interest.
* It allows a user to ask *que.* in the forum or directly communicate with the required mentor if that he/she has permitted to volunteer for assisting to *que.* on a chatbox.
* Allows a user who is the *CR* of a class to form a group of his/her classmates where he/she can post any poll *que.* to which his/her classmates can respond to take a decision.
* It allows a user to rate and comment on a *doc.*
* It allows a user to give feedback about the *app* which is helpful in the improvement of the *app.*
* Keeps the *info.* of a user secured.

## **Operating Environments**

For developers: -

* 1280 X 800 *min* screen resolution.
* 3GB RAM *min*, 8GB RAM recommended, plus 1GB for the Android Emulator.

For user: -

* Android users should have KitKat (4.4-4.4.4) or above version with internet access.

## **User Documentation**

‘Notesify’ is a very easy to use *app*. Users can use most of its features by simply clicking on different icons or buttons. We will maintain the user *docs* from time to time during the development and host it on the play store/internet so that the user can have detailed info about the *app*. If there are any commonly asked *que.* about the working of the *app* then we will provide their answers in the FAQs panel.

## **General Constraints, Assumptions and Dependencies**

#### Constraints

* Only those mentors will be asked to answer the *que.* of the students who have volunteered for it. Other mentors can just upload *docs.* and refrain from answering *que.* if they can’t due to their busy schedules.
* Volunteers can choose if they only want to answer the *que.* of the students of their alma mater or any student.
* Only the user who has uploaded a *doc.* can modify it. Others can only download it and not modify it. However, if the user is open to suggestions, he/she can choose to make it modifiable.

#### Assumptions

* It is assumed that we will get enough volunteers to help the students.
* It is assumed that most people use android phones.
* It is assumed that the *docs.* uploaded by mentors are correct and helpful.
* It is assumed that students will have faith in the *docs.* uploaded by other mentors.
* It is assumed that the students will be willing to provide feedback about *docs.* read by them and about the *app’*s working.

#### Dependencies

1. The functioning of our project is mostly dependant on the availability and competency of the mentors because if there are no mentors available to upload any *docs.* then we won’t have anything to show to the users. To lessen this dependency, we will take the following measures if required.

* Get the docs. verified from other mentors before they are uploaded.
* If mentors are not available, then to encourage more mentors to volunteer, we will pay them for each *que.* they answer and the *docs.* they upload. These payments can then be realized from the students who want to access the paid *docs.* and get their *que.* answered from a certain mentor.

# **Specific Requirements**

## **External Interface Requirements**

### **User Interfaces**

The User Interface will be a *GUI* that will be created using Android Studio. Some of the main pages/windows in the *GUI* will be-

#### Login Page

The Login Page will appear every time the *app* is opened. It will take Email ID and Password of the user as input and login if credentials are verified. It will then display the Home Page. We will also provide the Forgot Password functionality to the user in case he forgets his password. In this case, a Password Reset link will be sent to the verified email account of the user.

If the user is new to our application then he can create a new account, which will open the Registration Page.

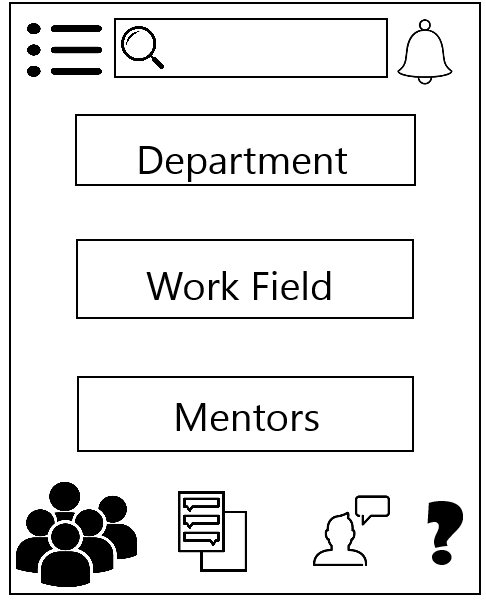


#### Registration Page

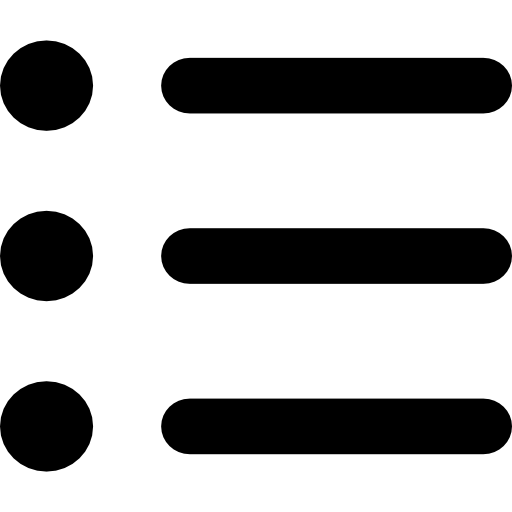
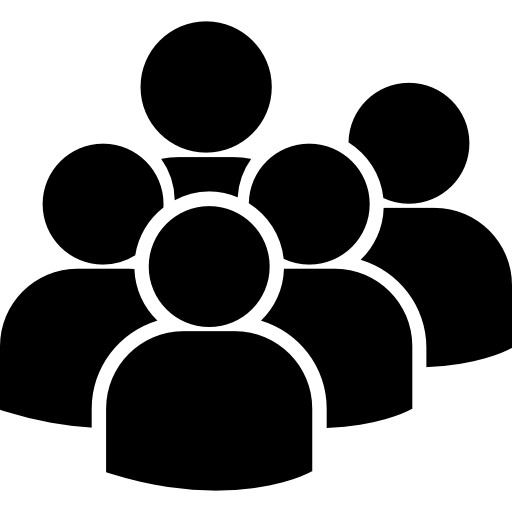
The Registration Page will ask the user to fill his/he credentials such as name, contact *info.*, college name, course, education field, etc. An alumnus will be further asked for *info.* like- current place of work, role, their Projects, etc. It will help us filter out the results for the students searching for guidance in various fields.

#### Home Page

Login Page will direct a user to the Home Page. This is the main page where a student can explore various features of the *app*.



Components of this page are-

*  - Clicking on this icon will open Options Menu which will include options-
* Edit Profile- Here the user can change his credentials like email id, phone, password, etc.
* Change Status- Here the user can change his mentor status. If a student thinks he/she has gained enough knowledge and experience in any field then he can volunteer to become a mentor.
* Followings- This shows the list of all those mentors who a student follows.
* Followers- If a user is a mentor then this shows a list of all the students who are following him/her.
* Settings- Here the User can change any settings related to the *app* like Font Size, Background Theme, Notification Settings, etc.
* *FAQ*- Here a user can read answers for all the Frequently Asked Questions.
* Sign Out- Click here to sign out from your account.
* - Clicking on this Notification Bell icon, a user can see if any of the Mentors or Teachers who he/she follows have posted any new *doc* or replied to his/her *que.*
*  - Clicking on this icon will open the Class Group of the Student, where a student can track any of the class activities like any assignment due or any other kind announcements. Here he\she will also be able to participate in any decision making in the class through the Polling Mechanism.
*  - Clicking on this option will open the My Posts page where a user can see the posts that he/she has pinned for future readings. The user can upload new *docs* and also edit his/her already uploaded posts.
* - Clicking on this icon will open the feedback form where a user can send feedback to the administrators suggesting any improvements in the *app.*
* **?** - Clicking on this iconwill open the Queries Block where a Student can post *que.* openly for anyone to answer.
* Pressing on the ‘Department’ Button will list various departments from where the student can select the required department from where he/she can further select the Semester. Each Semester will include various courses. Available courses will have different course materials like Notes, *PDF*s, Books, *PPT*s, etc.

* Pressing on the ‘Work Field’ Button will list various Fields of Work. Each field of work will have some active mentors and faculty working in those fields. We can Select any mentor and his/her profile will pop up.

The *app* will include several other pages also.

### **Hardware Interfaces**

Not Applicable

### **Software Interfaces**

The connection of the Mobile *app* with the *DB* will be realized by the use of *JDBC* Driver, which is an *API* for Java that defines how a client may access the *DB*.

Our *app* will use several services provided by Google for security and authentication purposes such as the Email verification of the user will be done through their google accounts. To use these services, we will use *APIs* provided by Google or any other vendor as the need may arise. The mentors can provide YouTube links to lectures and tutorials. To open those videos, we will use the YouTube *API* which is also provided by Google.

### **Communication Interfaces**

## **Functional Requirements**

This section includes the requirements that specify all the fundamental actions of the *app*.

1. **SIGN UP-**

Purpose: To create a new user and get his credentials.

Inputs: name, contact *info.*, college name, course, education field, password, etc.

Outputs: record of user inserted in the *DB* after verification through email.

1. **LOG IN-**

Purpose: To check the validity of the user by verifying the user ID and password in the *DB*.

Inputs: user name or email ID and password.

Outputs: opens the Home page for the user.

1. **RETRIEVE PASSWORD-**

Purpose: In case the user forgets his/her password he must be able to retrieve or reset it.

Inputs: Email ID and answer to a security *que*.

Outputs: Send a Reset Password link to the verified email ID of the user.

1. **LOG OUT-**

Purpose: To close a user’s account and open the Login/Sign Up window when the user clicks on the Log Out button.

Inputs: Event of clicking the Logout Button.

Outputs: Close all currently open windows and open the Login/Sign Up window.

1. **EDIT PROFILE-**

Purpose: Allows the user to change his/her credentials or password.

Inputs: New values for fields- name, contact *info.*, college name, course, education field, password, etc.

Outputs: Update the field entries in the *DB*.

1. **SEARCH CONTENT-**

Purpose: Users can search for any *docs* through the Search Bar on the Home page.

Inputs: Name of a *doc* or any topic.

Outputs: All the matching *docs*, lectures or posts.

1. **SEARCH BY DEPARTMENT-**

Purpose: a user belonging to a certain department may want to search for the subjects and topics in his curriculum semester-wise. He can do so by clicking on the Department Button on the Home page which will further have options for various departments in the form of Buttons which will open various courses related to the department and then their topics semester-wise.

Inputs: Event of clicking the Department Button.

Outputs: Opens another window containing various Buttons for different departments.

1. **SEARCH WORK FIELD-**

Purpose: a user interested in a certain work field may want to search for *docs* or mentors related to that field of work or research area.

Inputs: Event of clicking the Work Fields Button.

Outputs: Opens another window containing a list of various Buttons for different fields of work for which *docs* and mentors are available.

1. **SEARCH MENTORS-**

Purpose: a user may want to search for a particular mentor or see the list of available mentors.

Inputs: Event of clicking the Mentors Button.

Outputs: Opens another window containing a list of various Buttons for different mentors available.

1. **CHANGE MENTORSHIP STATUS-**

Purpose: a user who is already registered as a student may want to become a mentor if he/she has enough knowledge and work experience or a student already a mentor may want to stop being a mentor for reasons such as lack of time to answer *que* of other students.

Inputs: Click on Change Status Button in the Options Menu.

Outputs: Inserts/Deletes User Record in/from the Mentors Table in *DB*.

1. **RECEIVE NOTIFICATIONS-**

Purpose: a user will be notified if a mentor he/she follows has posted some new *doc* or has answered his/her *que.* A mentor will be notified if some student has asked him/her a *que.*

Inputs: Event of clicking the Notifications Bell Icon on Home page.

Outputs: Opens another window containing a list of various notifications clicking on which the user will be directed to the related content.

1. **POST A DOCUMENT-**

Purpose: a mentor can post *docs*, tutorials or lectures.

Inputs: Upload the *doc* on the cloud and provide its link.

Outputs: Add the details of the post in the associated table in *DB* and send notifications to students that follow the mentor.

1. **ASK QUERY FROM A MENTOR-**

Purpose: a user can ask a mentor a *que* from the chatbox available in the window of that particular mentor. The user simply has to send a message.

Inputs: Enter the *que* in the chatbox and click the Send button.

Outputs: Sends a notification to the mentor. The mentor can see the *que* by opening his/her own chatbox and send the reply in a similar way.

1. **ASK QUERY OPENLY-**

Purpose: a user can ask a *que* openly in the forum from where everyone can see that *que* and answer it. Such *que*s are posted in the Queries Block, a Button to open the same is available on the Home page.

Inputs: Enter the *que* in the Query Block and post it.

Outputs: Add the details of the *que* in the Queries Table in the DB. Whenever any user will open his/her Queries Block he/she can also see the *ques* posted by other users.

1. **POST A POLL IN CLASS GROUP-**

Purpose: a CR of a class can post any poll in his class group. Every user who is a member of that class can see the poll question and choose an option from the available options.

Inputs: Enter Poll Question and valid options for it.

Outputs: All group members are sent notification of the Poll Question.

1. **VOTE FOR A POLL-**

Purpose: a class group member can vote for a Poll Question posted in his/her class group.

Inputs: select one of the available options for the answer.

Outputs: send the user’s choice to the CR of the class group.

1. **SEE POLL RESULTS-**

Purpose: CR of a class group will receive votes of other class group members. He/she can see the results of the poll in a structured format such as a pie chart.

Inputs: votes of other class group members.

Outputs: Aggregate of the votes to create a pie chart displaying the number of votes in favour of a particular option.

1. **RATING AND COMMENTS-**

Purpose: a user can rate a *doc* or comment on it based on his/her experience after studying that *doc.*

Inputs: Enter the Rating out of 5 or enter a comment.

Outputs: Ratings and Comments along with their other details are added in the Rating and Comments Tables in the *DB* respectively.

1. **FEEDBACK-**

Purpose: a user can provide feedback about the *app* and suggest any improvements which the maintenance team can see and act on it accordingly.

Inputs: Enter Feedback in the Feedback window that can be opened by clicking on the Feedback icon on the Home page.

Outputs: Inserts Feedback details in the feedback Table in the *DB.*

## **Non-Functional Requirements**

1. The user’s data must be secured and only authenticated users must be allowed to use the *app* features.
2. The software should be able to respond to a large number of users simultaneously.
3. Other users must not be able to hamper the *docs* posted by one user.
4. The user’s data and history must be maintained. A user must be able to find the docs once searched by looking into his/her history for easier access.
5. The *app* should not take much time to search or open a *doc* since quick searching is the main motto of this *app*.

## **Design and Implementation Constraints**

* Eclipse and Android Studio have to be installed to develop the *app*.
* Time is the major constraint on implementation. We may not be able to implement all of the above-specified features and functionalities.
* Lack of knowledge regarding communication protocols and network security may limit the security features of our *app*.

## **Other Requirements**

Not Applicable

# **To Be Determined List**

Not Applicable