# Software Requirements Specification

# For

# Notes Sharing System

Version 1.0 approved

Prepared by:

Vinay Badhan

Ayush Singh Chauhan

Department of Information Technology

August 25, 2016

Table of Contents

Table of Contents [ii](#__RefHeading___Toc441230970)

Revision History [ii](#__RefHeading___Toc441230971)

1. Introduction [3](#__RefHeading___Toc441230972)

1.1 Purpose [3](#__RefHeading___Toc441230973)

1.2 Document Conventions [3](#__RefHeading___Toc441230974)

1.3 Intended Audience and Reading Suggestions [3](#__RefHeading___Toc441230975)

1.4 Product Scope [3](#__RefHeading___Toc441230976)

1.5 References [4](#__RefHeading___Toc441230977)

2. Overall Description [4](#__RefHeading___Toc441230978)

2.1 Product Perspective [4](#__RefHeading___Toc441230979)

2.2 Product Functions [5](#__RefHeading___Toc441230980)

2.3 User Classes and Characteristics 5

2.4 Operating Environment [6](#__RefHeading___Toc441230982)

2.5 Design and Implementation Constraints [6](#__RefHeading___Toc441230983)

2.6 User Documentation [6](#__RefHeading___Toc441230984)

2.7 Assumptions and Dependencies [7](#__RefHeading___Toc441230985)

3. External Interface Requirements [7](#__RefHeading___Toc441230986)

3.1 User Interfaces 7

3.2 Hardware Interfaces [11](#__RefHeading___Toc441230988)

3.3 Software Interfaces [11](#__RefHeading___Toc441230989)

3.4 Communications Interfaces [12](#__RefHeading___Toc441230990)

4. System Features [12](#__RefHeading___Toc441230991)

4.1 System Feature 1 [12](#__RefHeading___Toc441230992)

4.2 System Feature 2 (and so on) [13](#__RefHeading___Toc441230993)

5. Other Non-functional Requirements [14](#__RefHeading___Toc441230994)

5.1 Performance Requirements [14](#__RefHeading___Toc441230995)

5.2 Safety Requirements [14](#__RefHeading___Toc441230996)

5.3 Security Requirements [14](#__RefHeading___Toc441230997)

5.4 Software Quality Attributes 1[5](#__RefHeading___Toc441230998)

5.5 Business Rules 1[5](#__RefHeading___Toc441230999)

6. Other Requirements [15](#__RefHeading___Toc441231000)

Appendix A: Glossary [15](#__RefHeading___Toc441231001)

Appendix B: Analysis Models [16](#__RefHeading___Toc441231002)

Appendix C: To Be Determined List [18](#__RefHeading___Toc441231003)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **Notes Sharing System** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the **Notes Sharing System** are provided in this document.

## Purpose

## The Product that we have is Notes Sharing System, revision number being 1.0. It is a web application which provides online notes sharing services to everyone at their doorstep. The students in the college can connect through internet to get this service. This web application is more effective, quick in providing notes especially, to students who find it difficult to maintain the bulk of notes throughout the semester. This helps the students to maintain a proper record of notes which they can download and can even upload the notes on their own.

## Document Conventions

This document follows MLA Format. **Bold-faced** text has been used to emphasize section and sub-section headings. Highlighting is to point out words in the glossary and *italic* text is used to label and recognize diagrams.

## Intended Audience and Reading Suggestions

This document is to be read by the development team, the project managers, testers and documentation writers. The students may review the document to learn about the project and to understand the requirements. The SRS has been organized approximately in order of increasing specificity. The developers and project managers need to become intimately familiar with the SRS.

Others involved need to review the document as such:

Overall Description – This is a working document and, as such, is subject to change. In its initial form, it is incomplete by definition, and will require continuing refinement. Requirements may be modified and additional requirements may be added as development progresses and the system description becomes more refined.

## Product Scope

**Notes Sharing System** is a web application that allows everyone to access the notes that are submitted by the students themselves. They can download as well as upload notes, also it eliminates the task of handling the bulk of notes. The contribution to the notes repository not only benefit the student itself but also other students, especially the juniors also get an introduction of the subjects that they would come across in future. This gives them an upper hand over the tight time schedule that an engineering student has to go through.

## References

[1]"Django Tutorials for Beginners - YouTube", *Youtube.com*, 2016. [Online]. Available: https://www.youtube.com/playlist?list=PL6gx4Cwl9DGBlmzzFcLgDhKTTfNLfX1IK.

[2]"PHP Tutorials Playlist - YouTube", *Youtube.com*, 2016. [Online]. Available: https://www.youtube.com/playlist?list=PL442FA2C127377F07.

.

[3]"XAMPP Installers and Downloads for Apache Friends", *Apachefriends.org*, 2016. [Online]. Available: https://www.apachefriends.org/index.html.

[4]"HTML5 Introduction", *W3schools.com*, 2016. [Online]. Available: http://www.w3schools.com/html/html5\_intro.asp.

[5]"CSS3 Introduction", *W3schools.com*, 2016. [Online]. Available: http://www.w3schools.com/css/css3\_intro.asp.

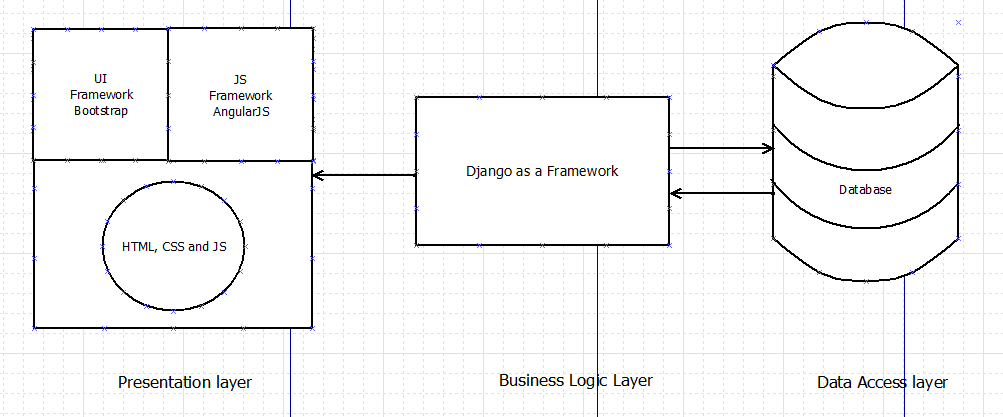
[6]"JavaScript Tutorial", *W3schools.com*, 2016. [Online]. Available: http://www.w3schools.com/js/.

# Overall Description

## Product Perspective

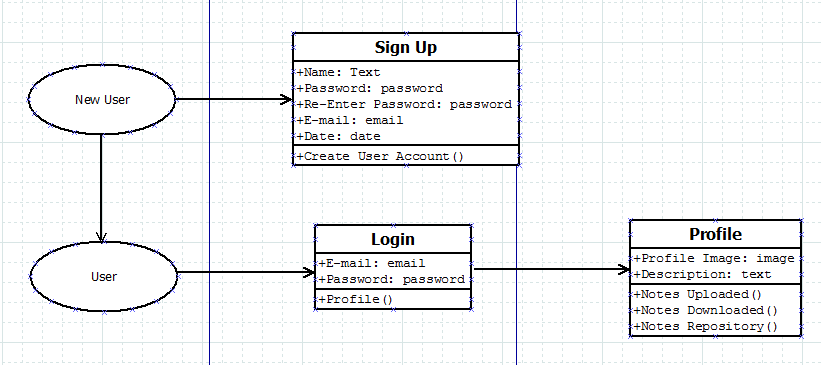
**Notes Sharing System** is based on an innovative idea which aims to ease the hectic task of maintaining the bulk of notes. Especially for college students like us who face sleepless nights before exam just to have a grasp of topics covered. The open collaboration repository of notes provided will sort out this problem faced by students.

Layer Architecture for the project



## Product Functions

* User Registration
* Login option
* Uploading Notes
* Downloading Notes
* Editing Profile
* Admin privilege



## User Classes and Characteristics

### Students:

Remote students most frequently use the device for notes sharing purposes as well as notes uploading and downloading. The students are not expected to have a high educational and proficiency level or technical expertise. Hence, the user interfaces is suitable for students (learners) or all ages irrespective of their geographical location.

### DBA:

The DBA is expected to have a field experience of at least 2 years as a DBA and an additional 2 years in the IT field. He/she has the privilege to update information in the database and technical expertise in database management. The DBA does not directly interact with the **Notes Sharing System**.

## Operating Environment

### Operating System:

* Windows XP
* Windows Vista
* Windows 7
* Windows 8,8.1
* Windows 10
* Linux (versions that supports Mozilla Firefox)
* Ubuntu (12.04 or later)

### Hardware:

* Screen resolution should be exactly1366 x 786.
* Screen size should be 15.6”.

### Server Side:

* Django as a backend framework
* Sqlite as a database

## Design and Implementation Constraints

1. Above mentioned software should be installed.
2. Limited to laptops with screen size of 15.6’’ only.
3. Text Editor: Brackets
4. UI Framework: Bootstrap
5. JS Framework: Angular JS
6. HTML5: Markup Language
7. CSS3: Stylesheets
8. Sqlite: Database
9. Django: Backend Framework
10. Encoding Standards: UTF8
11. Design Convention: DOM(Document Object Model)
12. Use of standard semantics.

## User Documentation

## Assumptions and Dependencies

1. It is assumed that users are making use of modern web browsers.
2. Currently **Notes Sharing System** display the contents correctly on 100% zoom.
3. Bootstrap should be installed along with fontawesome.css and mystyles.css
4. Users can download the project dependencies from Github profile

# External Interface Requirements

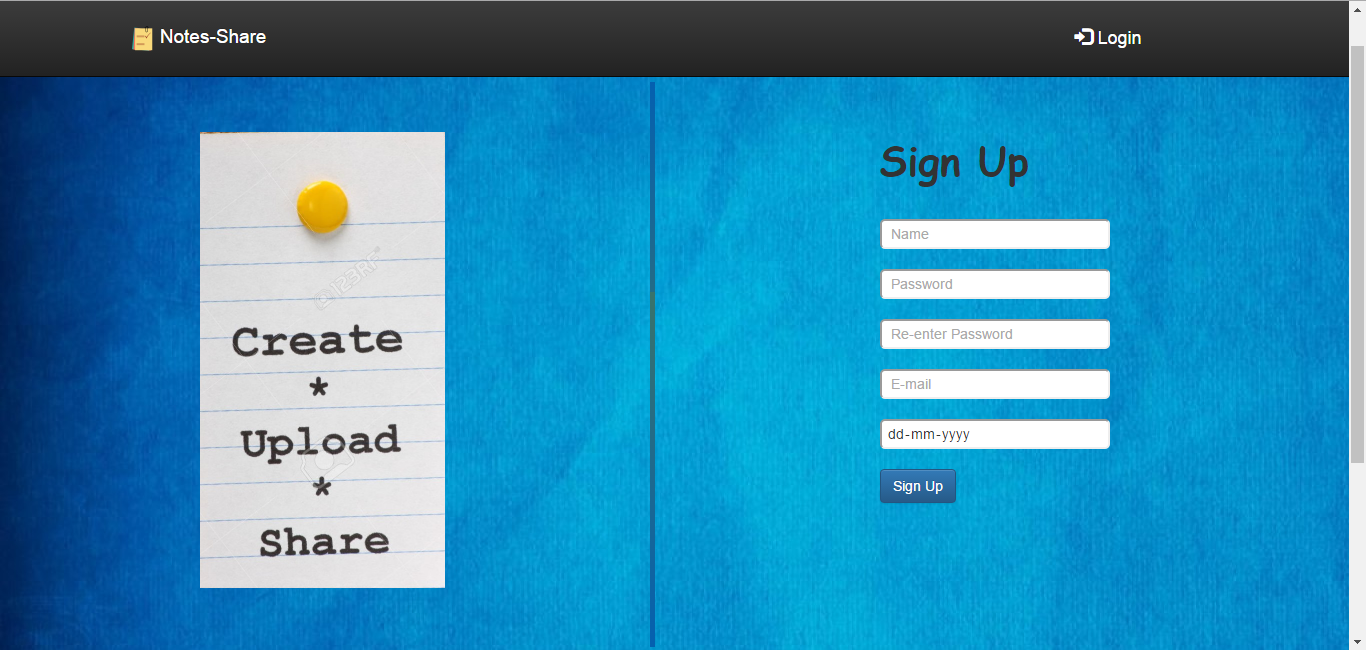
## User Interfaces

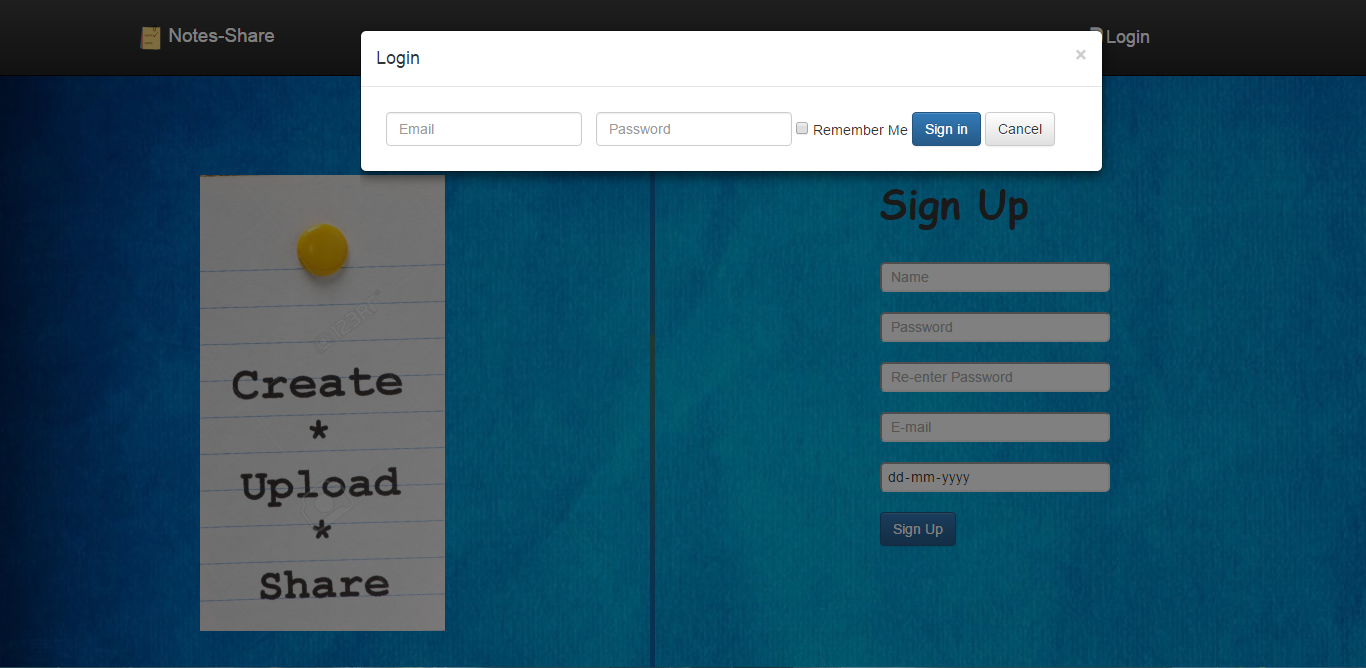
3.1.1 GUI Standards

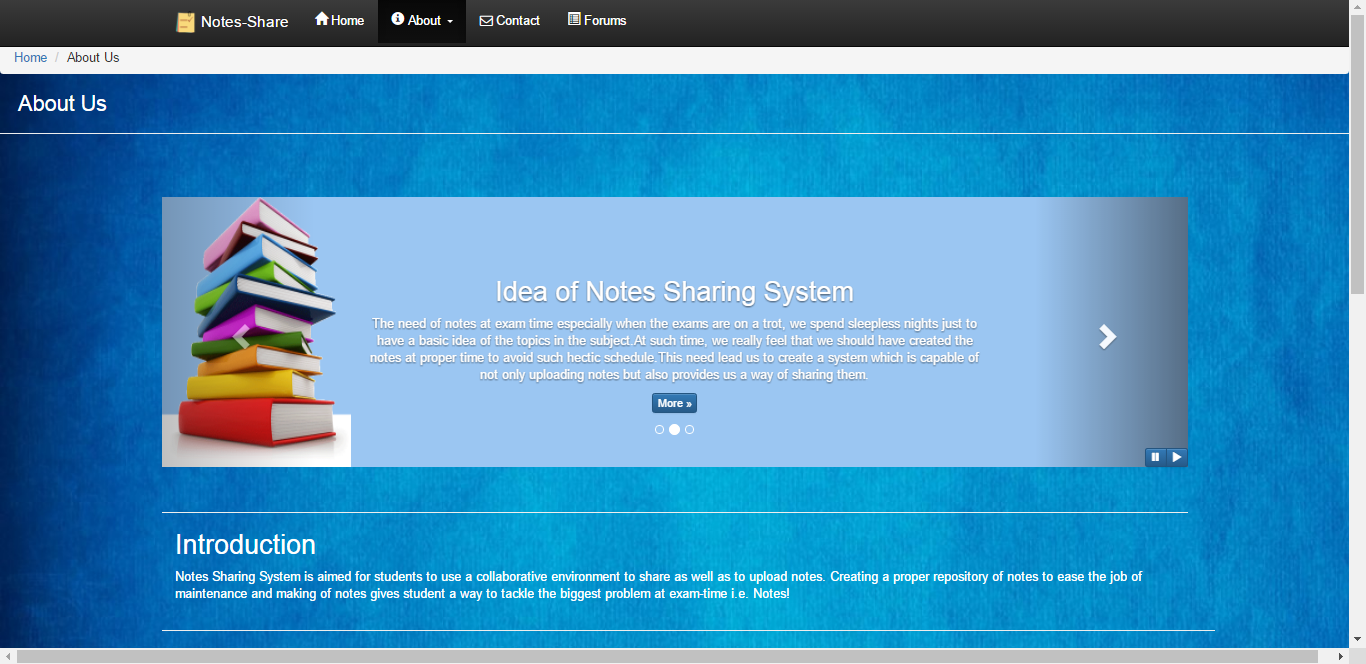
* UI frameworks like Bootstrap are extensively used to provide best possible interface for the users. The interactive elements in the various parts of the **Notes Sharing System** are the distinguishing feature in our project which sets it apart from other projects.
* Examples of previously mentioned interactive elements can be easily seen in the screenshots of the Frontend part of the project. In the following page screenshots of the **Notes Sharing System** are presented.
* UI framework provides a sandbox environment for a programmer to pour out his/her thought in real life.
* Dynamic behavior is also present within this project. For ex- in the second screenshot a “modal” JS plug-in is used for login. Similarly, in the third screenshot, an image of carousel (provides slides) can be seen easily. Along with the above mentioned features, a navbar (navigation bar) is also provided with the logo of our project on the left side of it.
* Footer is also present along with glyphicon icons to provide better readability and proper links are associated. About page gives information regarding the aim of our project along with its scope and gives a brief introduction of the team members in the project.
* Contact page provides geographical location along with the facility to contact us using email id or mobile number.

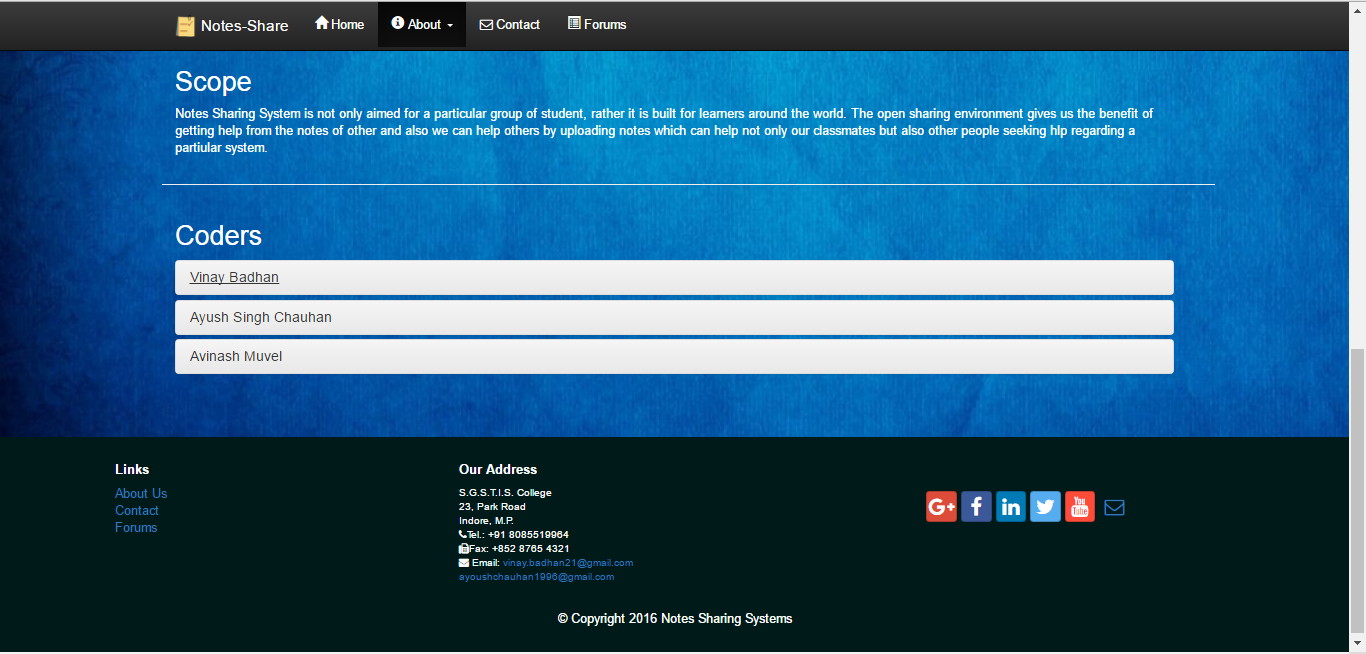
3.1.2 Design Constraints and Display Standards

* Screen Layout Constraints for laptops/ PCs is 1366\*786px.
* Running atmosphere must have CSS and JS; available on their systems to run the software.
* The computers must be equipped with web browsers mentioned above.
* A general knowledge of basic computer skills is required to use the product

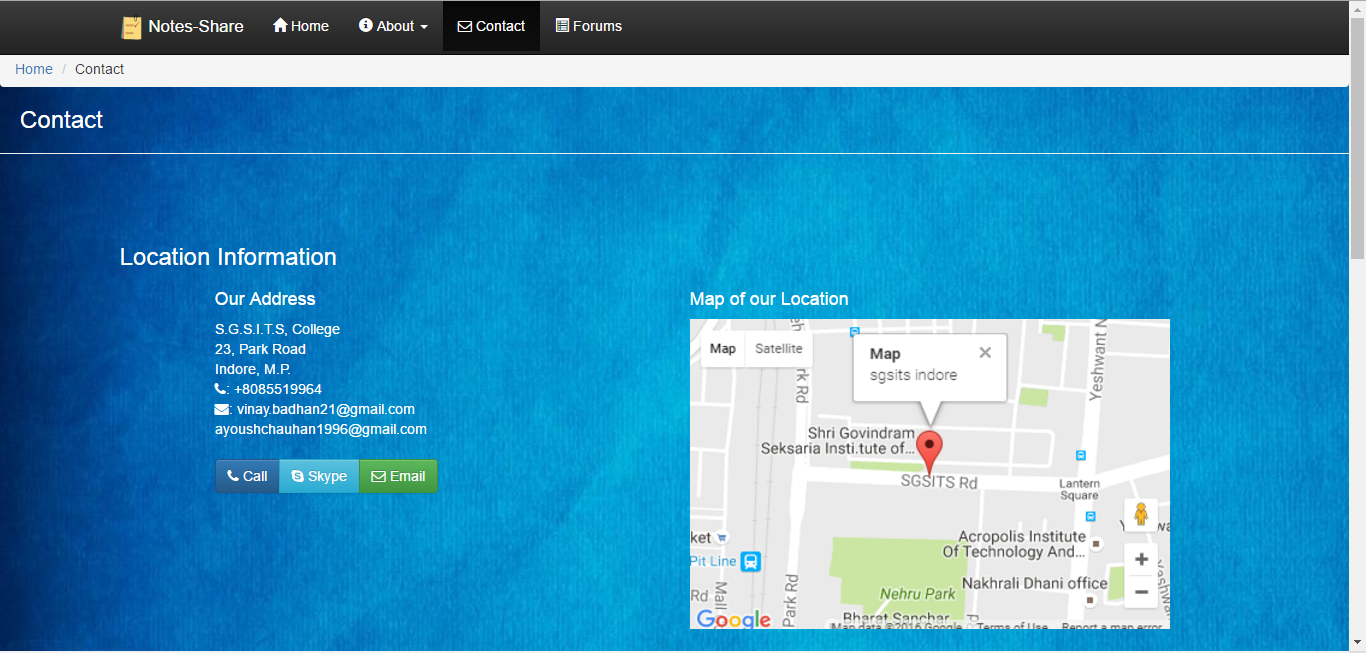
**Login Page**

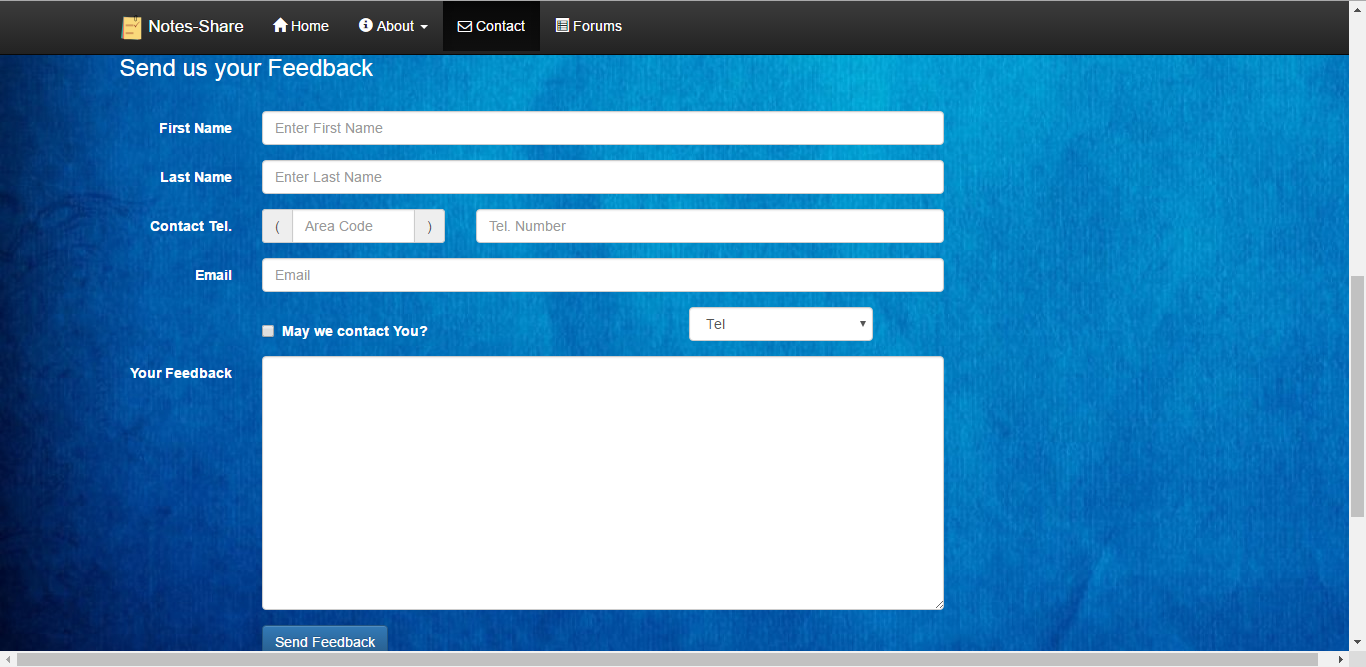


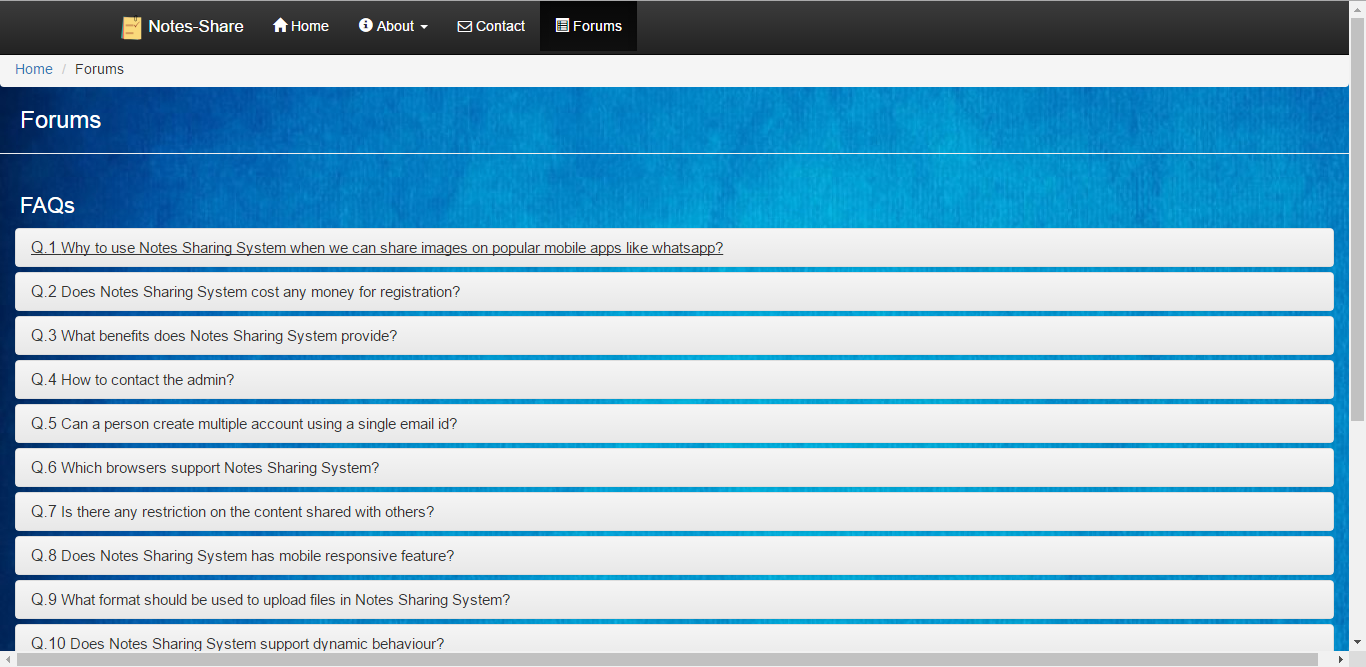
**About Page**

****

**Contact page**





**Forums Page**

## Hardware Interfaces

* Monitor screen – the software shall display information to the user via the monitor screen
* Mouse – the software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, command buttons and select options from menus.
* Keyboard – the software shall interact with the keystrokes of the keyboard. The keyboard will input data into the active area of the database.

## Software Interfaces

**Notes Sharing System** makes use of web browsers (like Mozilla Firefox/Google Chrome), to run and give software interface to its users. It uses Sqlite database to store the data and information provided by the user. It makes use of tools like CSS, Bootstrap, JS and Angular JS, and their respective libraries, for its implementation. We have used Brackets, as out text editor. Details provided by the user will be shared across the software components during Login and Signup. This **Notes Sharing System** makes use of File Browser while uploading content on it.

1. Server side An Apache web server will accept all requests from the client and forward it accordingly. A database will be hosted centrally using Sqlite.
2. Client side An OS which is capable of running a modern web browser which supports JavaScript and HTML5

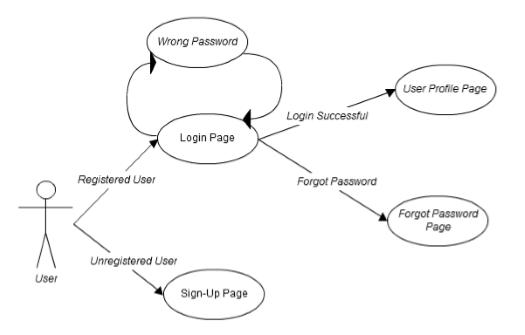
## Communications Interfaces

The HTPP or HTTPS protocol(s) will be used to facilitate communication between the client and server.

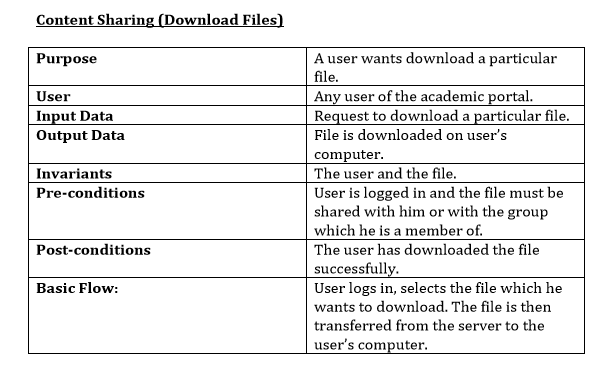
# C:\Users\vinay\Pictures\Screenshots\Screenshot (30).pngSystem Features

## C:\Users\vinay\Pictures\Screenshots\Screenshot (30) - Copy.pngUse Case Scenario 1 – User Login

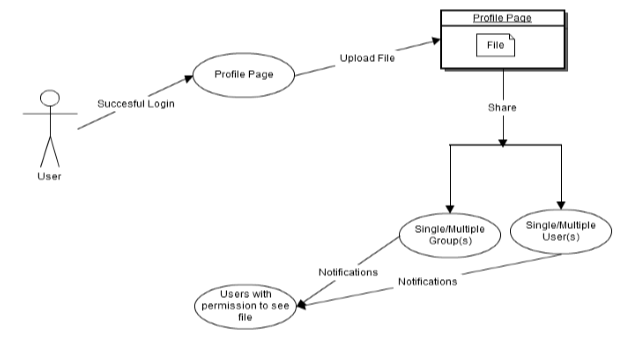
**User Login**

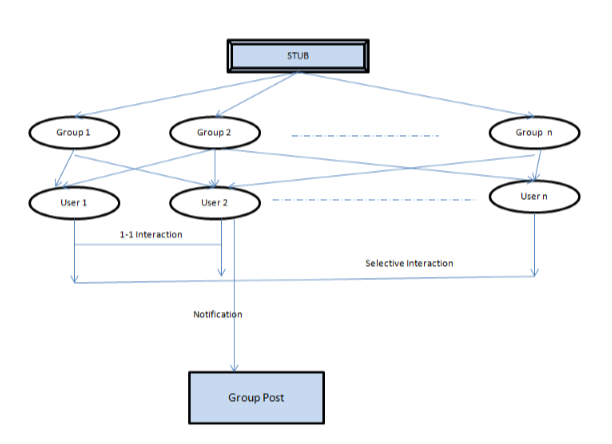


## Content Sharing:



**Content Sharing**

****

**Content Sharing Network**

**Notes Sharing System**

# Other Nonfunctional Requirements

## Performance Requirements

It is important that a substantial number of users be able to access the system at the same time, since the **Notes Sharing System** is important to the students that use it, especially at exam time.

## Safety Requirements

“Illegal” or “Copyright Content” should not be uploaded or posted anywhere or in any form on the **Notes Sharing System**.

## Security Requirements

* Passwords will be saved encrypted in the database in order to ensure the user's privacy.
* The user's IP will be logged.
* The system will be protected against vulnerabilities such as SQL injection attacks.
* “Illegal” or “Copyright” content should not be uploaded or posted anywhere or in any form on the **Notes Sharing System**. Any such actions will be resolved at Indore High Court jurisdiction.

## Software Quality Attributes

The software consists of following attributes:

* Created using mobile first approach by bootstrap UI framework.
* Dynamic behavior through Angular JS framework.
* The database should remain consistent at all times in case of an error.

## Reliability:

## The reliability of the overall program depends on the reliability of the separate components.

* + 1. **Availability:**

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. In case of a of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved with the Sqlite server and saved by the administrator.

* + 1. Maintenance**:**

Sqlite is used as a database. In case of a failure, a re-initialization of the program is recommended.

* + 1. **Portability:**

Components used are practically independent of the OS. So the **Notes Sharing System** provided independency of the OS that an user can choose.

## Business Rules

* Administrator can change the privileges for users
* Administrator can add or delete users.
* Individual users have the right to edit the content shared by them only.

# Other Requirements:

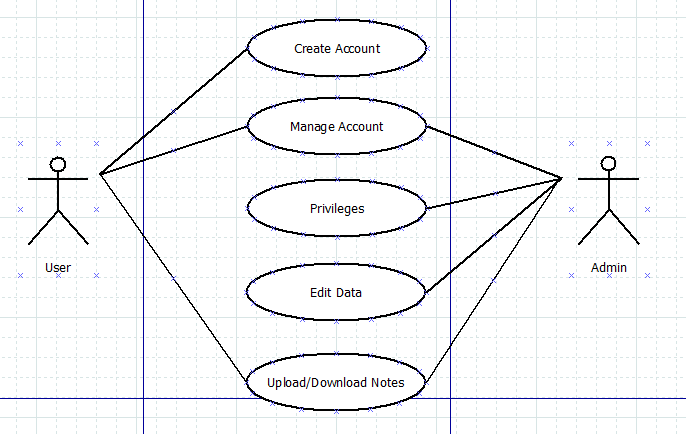
Appendix A: Glossary

|  |  |
| --- | --- |
| Word | Definition |
| Remote | Distant / Mobile |
| Backend | Server side of the software |
| DOM | Document Object Model |
| Semantics | Meaning of a text |
| Protocols | Set of rules |
| IP | Internet Protocol |
| SQL | Structured Query Language |
| Framework | A basic structure underlying a system |

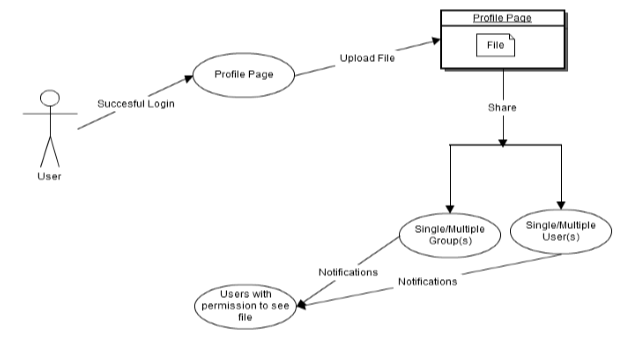
# 

Appendix B: Analysis Models

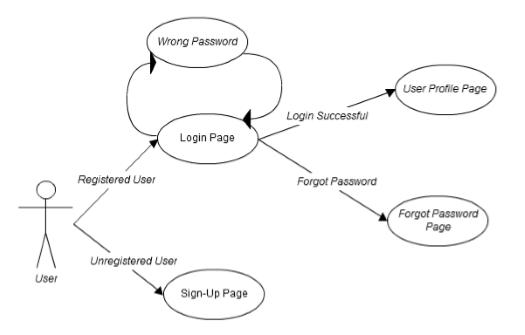
Use Case Diagram



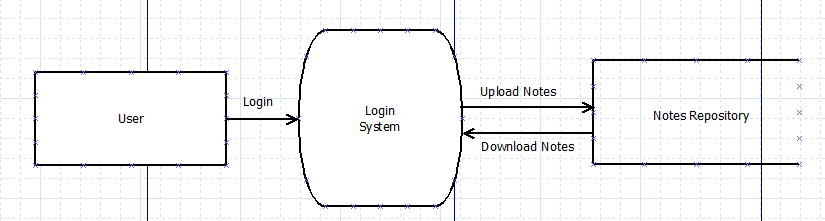
**Login Use Case Diagram**



**Use Case Diagram**



**Data Flow Diagram**

****

**Class Diagram**

