

# **SOFTWARE REQUIREMENT SPECIFICATIONS**

## **Project Topic: [Note\_Taking App]**

**Submitted by: [Tshering Penjor 12190097]**

## 1. Introduction

The following section provides an overview of derived software requirements specification for the Note Taking App. To begin with, purpose of the document is presented and scope of the project is specified with the particular focus on what the resultant software will do. The document will also include functional requirement and non-functional requirements.

The introduction of mobile computing devices in classrooms ushered in a new set of note taking issues. Speed, legibility, and search ability are three positive attributes of digital note taking. Because of these advantages, some students may prefer digital note-taking over traditional handwritten notes. The digital way of note taking has not only ease the effort required to take the notes but also made it a secure and more dynamic than the usual traditional way of maintaining the notes. With the creation of mobile application, one can easily organize, make important notification and set a reminder using the mobile application which we are going to develop. Some of the benefits of digitally maintaining notes are the notes you store will stay organized and since if you note it digitally you won't require to write it on the paper, which will help contributing a good role in preserving the environment.

### A. Purpose

The purpose of the document is to give detailed description of the requirement for the Note Taking application. It will illustrate the purpose and complete declaration for the development of system. It will also explain system software requirements. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

### B. Scope

There are many current application for note taking in the market that provide wide range of functionalities and better features from one another. Existing application do not explicitly serve for needs of academia, rather they are more general purpose application that target a broader market, trying to gain as many customer as possible. Due to more advance features, usability range is low as most people are not aware of its proper usage and functionality. These gaps in existing application formed the starting point for this project which aimed to create an android based application called Note taking app which is simple but will provide basic features to maintain notes for academic students.

Scope of our project is School and College (BHUTAN) The android app is based on the student usage where it will help students to keep note of their Teachers, assignment dates (classwork and homework) and important events.

## 2. Requirements

### A. Functional Requirements

This section describes the functionality of Note taking application.

#### User Registration

When a new user uses the app they will need to register to the system. User will need to provide information such as Username, gender, phone number, email address and password. User profile can be edited after they have logged in to the app.

#### Note-taking platform interface

This app will allow you to write a note in a plain page and it will allow you to edit the existing note.

#### User profile

Registered user can edit their user profile depending upon their needs after they have logged into the app.

#### Logout

Registered user can log out of the app when desired.

### B. Non-functional Requirements

#### Usability

The mobile application will have a user friendly interface such that user will be familiar with the UI of the app in a short duration. This will be achieved by the use of graphical interface in combination with clear text and information.

#### Supportability

Note-taking app can be deployed in any android smartphone since majority of the population uses smartphones and android being the cheaper variant.

### C. Software Requirements

#### App Development Tools

- Java SE jdk1.8.0\_111 and above
- Android SDK-25 and above
- Android Studio version 2.3.3 and above
- Firebase

#### Documentation and Design Tools

- Latex
- MS Word
- Adobe XD (UI Design)

#### User Requirement

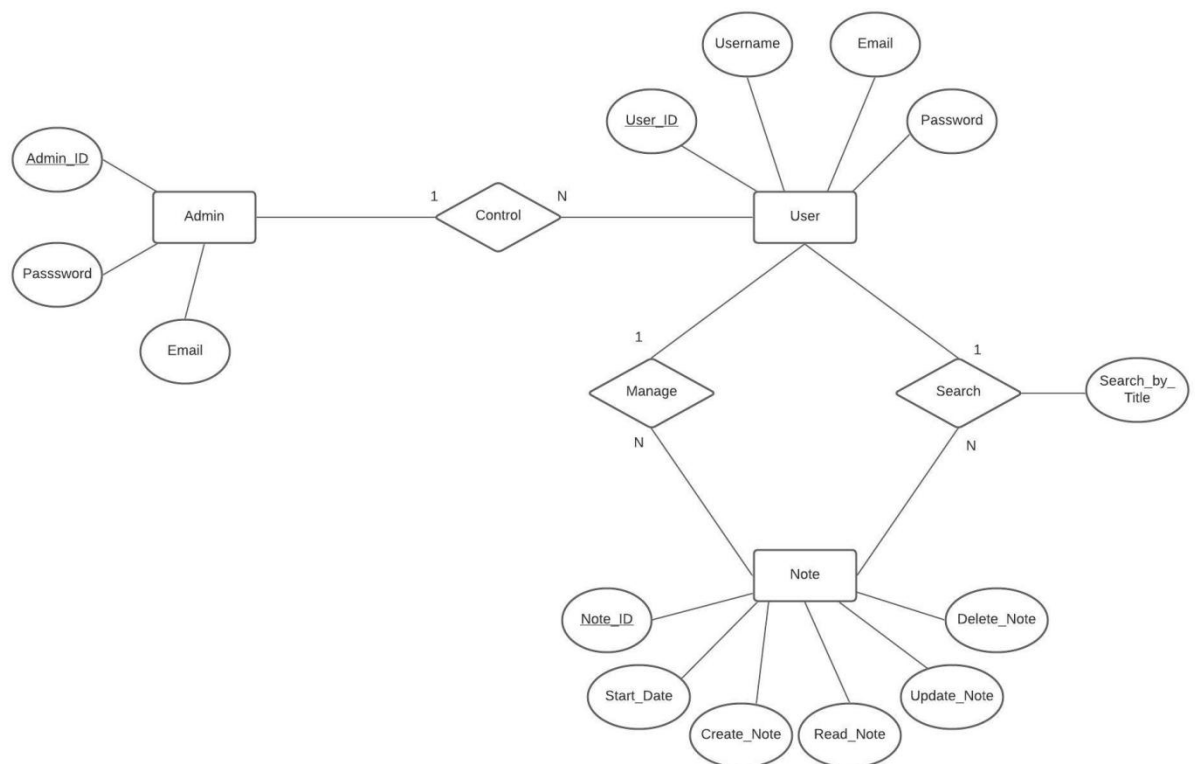
- Only devices with android version 6.0 and above is supported.
- Minimum of 2GB RAM.

### 3. Hardware Requirements

- Processor: dual core 64-bit, 2GHz Intel core i5
- RAM : 4GB
- Hard disk Space: 500Mb and 1GB for android SDK

### 4. System Design

#### Entity relationship diagram



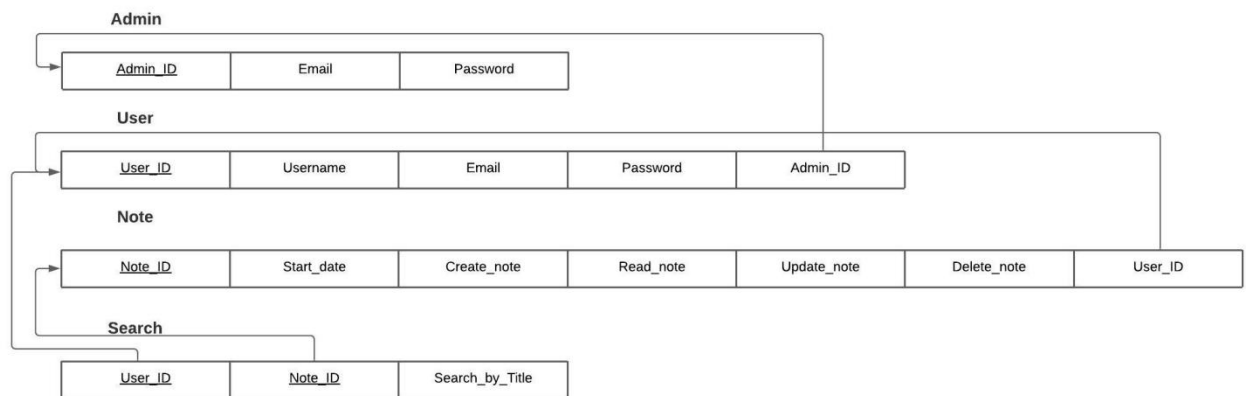
Following are the Entities, Attribute and Relationships:

- Entities - Admin, User and Note.
- Attributes - For Admin entity ( Admin\_ID (P.K), Email and Password). For User entity ( User\_ID (P.K), Username, Email and Password ). For Note entity ( Note\_ID (P.K), Start\_Date, Create\_Note, Read\_Note, Update\_Note, Delete\_Note ).

( Note\_ID (P.K), Start\_date, Create\_note, Read\_note, Update\_note and Delete\_note).

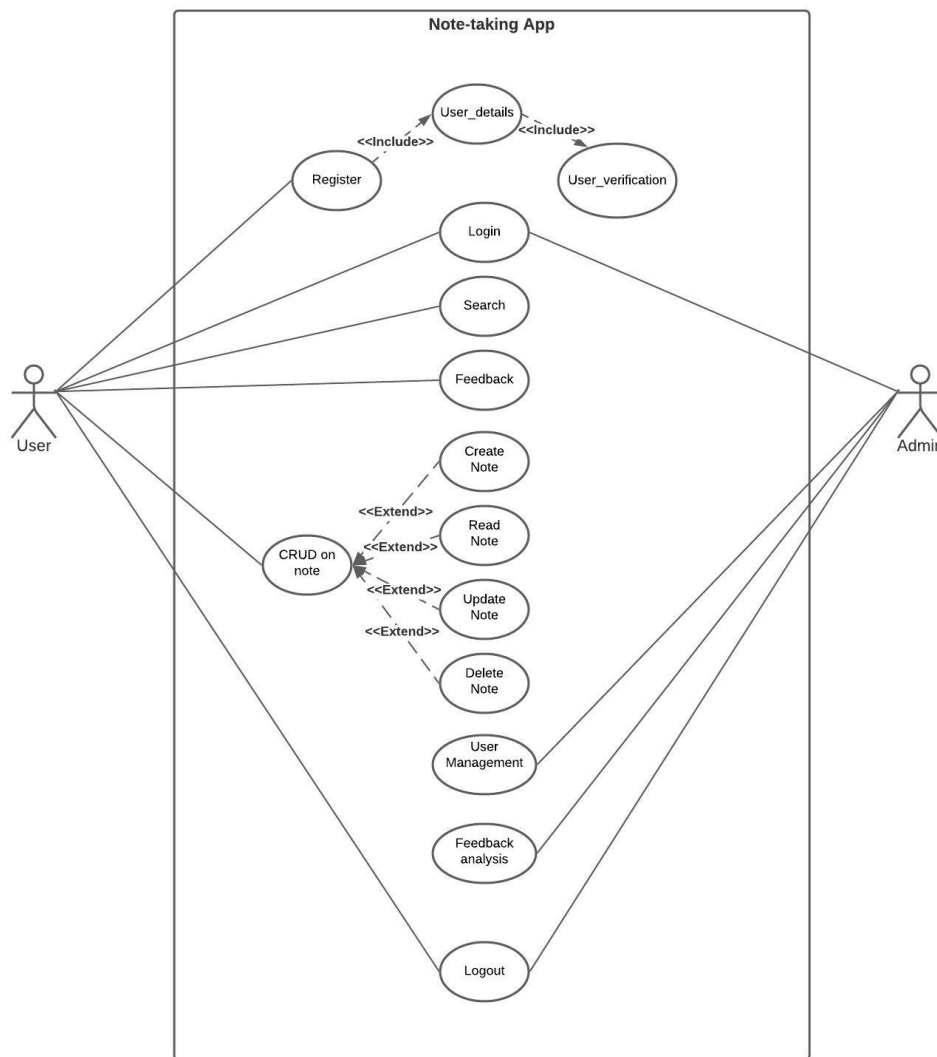
- Relationships - Between Admin and User : one to many relationship ( one admin can control many user and many user is being controlled by one admin ).  
Between User and Note : one to many relationship ( one user can manage many note and many note is being managed by one user, similarly one user can search one note and many note is being searched by many user ).

## Schema diagram



For every entity in a ER diagram I have created relation/table in a schema diagram. From ER diagram you can see that between admin and user there is one to many relationship so in schema diagram Admin relation/table contain Admin\_id as primary key in which Admin\_id act as a foreign key in user relation/table. These similar mapping rule apply to other entities have one to many relationship. In case of User and Note entity, from the ER diagram you can see that the user can search note in which new relation is being created since the search relationship have search\_by\_title attribute so a new relation search is being created in which the User\_id and Note\_id ( is foreign key in user and note relation ) acts both as primary key in Search relation/table.

## Usecase Diagram

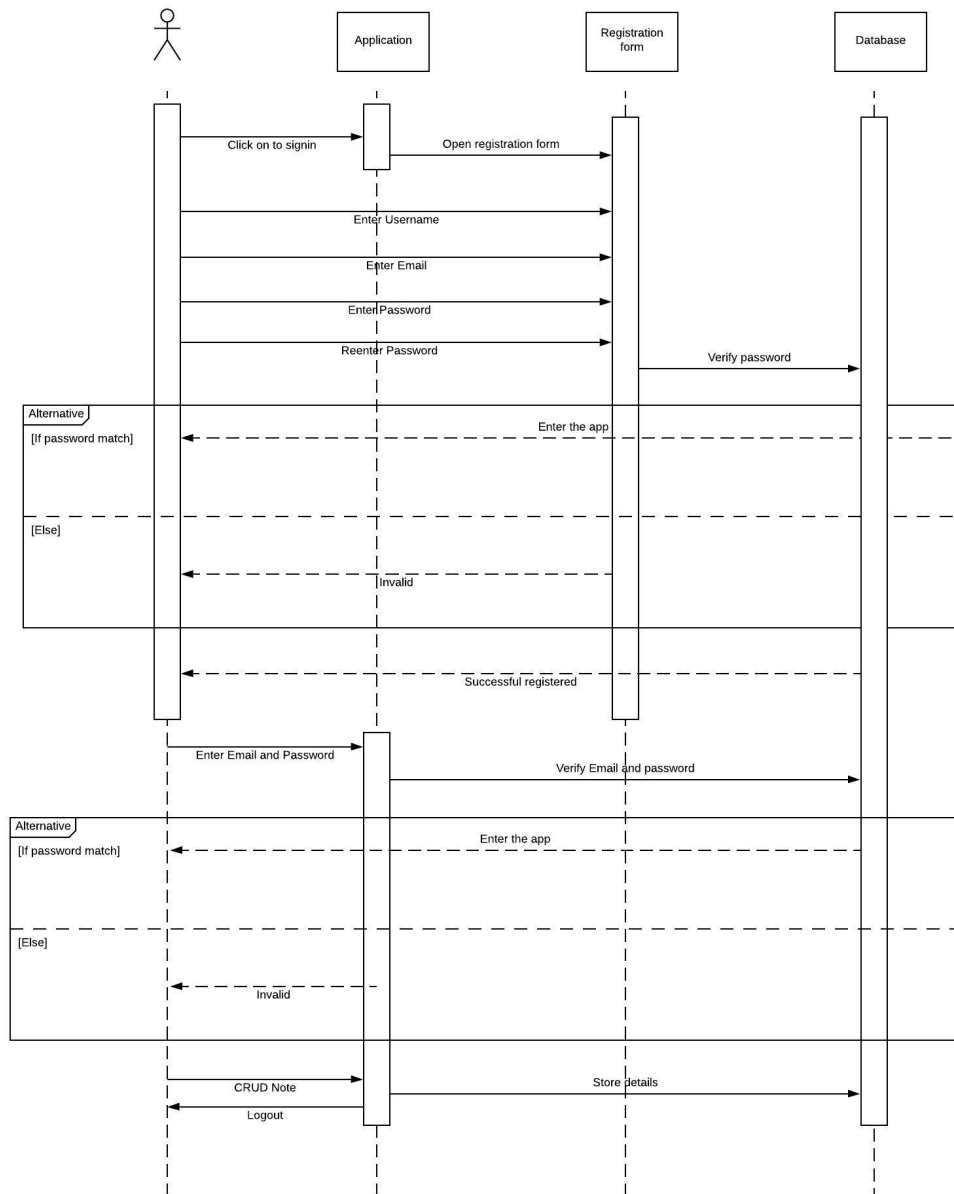


From the above usecase diagram it consist of two actor User as primary actor and Admin as secondary actor.

User can register and after registering user can login into the app. User can also search for a particular note by entering the title of the note. When user login into the app user can create, read, update and delete notes depending upon user choice. If the user have some issue with any features of the app user can give feedback. After using these app user can also logout when desired.

Similarly admin can manage user like admin can delete user account and so on. Admin can also view the feedback given by the users.

## Sequence Diagram



The above sequence diagram basically describe the interaction diagram because it describes how-and in what order-a group of object works together. So the above diagram describe whole interaction between user, application, registration form and database.