# Software Requirements Specification

**for**

## ANDROID BASED COLLEGE ASSISTANT APPLICATION

**Version 1.0 approved**

**Prepared by CS17-33**

**24-03-2017**


| | | |
|---|---|---|
| **KAAHWAARMSTRONG** | **14/U/288** | **Kaahwaarmstrong2014@gmail.com** |
| **KAGGWA HAM** | **14/U/6759/PS** | **kaggwaham@gmail.com** |
| **SEGAWA DAVID BISASO** | **14/U/1115** | **bdsegawa@gmail.com** |

# Table of Contents



**Revision History**

| Name | Date | Reason for Changes | Version |
|---|---|---|---|
| **COLLEGE ASSISTANT APPLICATION** | **14/03/2017** | **Initial version** | **Version 1.0** |

## 1.Introduction

This section gives a scope description and overview of everything included in this SRS document about the Application. Also, the purpose for this document is described and a list of abbreviations and definitions is provided

### 1.1 Purpose

The purpose of this document is to give a detailed description of the functionalities and requirements for the "ANDROID BASED COLLEGE ASSISTANT "Application. This document will also cover each of the system's intended features, application's User Interface and interactions with other external applications. The document will also cover hardware, software, and various other technical. It will also explain constraints and systems. This document is primarily a reference for developing the first version of the application for the development team as well as the course facilitators for assessment of the progress of the app development.

### 1.2 Document Conventions

The format of this SRS is simple. Bold face text and indentation is used on general topics or main section titles and other specific points of explanation with text size of 14, 12 respectively. The remainder of the document will be written the standard font, new times roman italicized text is used to label explanatory comments and recognize diagrams.

### 1.3 Intended Audience and Reading Suggestions

- This document is intended for students, developers, project manager, users, testers, documentation writers etc. This document is organized as follows. The SRS has been organized approximately in order of increasing specificity
- Section 1: Introduction, provides a brief introduction to this document, the purpose, document conventions, intended audience, reading suggestions, product scope, and references.
- Section 2: Overall Description, provides brief general descriptions of the product and its functions, user classes and characteristics, operating environments, design and implementation constraints, assumptions and dependencies.
- Section 3: External Interface Requirements, provides the detailed information regarding user interfaces, hardware interfaces, software interfaces and communication interfaces. Section 4: System Features, provides the detailed description of various features of the system.
- Section 5: Other Nonfunctional requirements, provides information regarding performance requirements, safety requirements, security requirements, software quality attributes and business rules.

- Section 6: Other Requirements Provides other requirements that are not included in the above sections.

## 1.4 Product Scope

Specifically, the application is designed to communicate and to notify the updates from the college. The system also contains a relational database containing a list of registered students and other information. The application will facilitate communication between students and the faculties via database.

### 1.5 References

K. B. Lee, "Developing Mobile Collaborative Learning Application for Mobile Users. "International Journal of Interactive Mobile Technologies, Vol 5, No 4, 2011, pp. 42-48.

User Interface Guidelines. (n.d.). Retrieved March 10, 2017, from https://developer.android.com/guide/practices/ui_guidelines/index.html

IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", October 20, 1998.

## 2 Overall Description

### 2.1 Product Perspective

This system will consist of two parts: a mobile application and one web portal that will be used on only rare occasions for example transfer of heavy files. The college assistant application project is a new self-contained mobile application, in this project the app is on an android platform. The scope of the project encompasses both server and client-side functionalities, so both aspects are covered in detail with in this document. It will be implemented with java jdk, android sdk, and android studio.

### 2.2 PRODUCT FUNCTIONS

1. **User registration & welcome**
   - Only appears once (the first time the application is run).
   - Allows the user to register with the applications database.

2. **Push and SMS notifications**
   - Appear after any significant event is about to occur.

- Administrator can broadcast messages to all students.

**3. GPS tracking**

- Shows college locations and distances from a particular place to destination
- Utilizes Google Maps to display and find locations
- User can navigate the whole college map fragment in-built in the app and can switch into different map views.

**4. Battery saving application**

This application saves your battery too because the service implemented in application is not running all the time. Whenever FCM ping the mobile, only then it makes a broadcast to phone that initiates the service. In this way, it is saving your battery a lot.

**5. Anytime anywhere service**

With this application, notices will be delivered anytime and at any place. There is no restriction of time to send a notice.

## 2.3 User Classes and Characteristics

There are three types of users that interact with the application that is to say registered students, administrators and the guest users. Each of these three types of users has different use of the application so each of them has their own requirements.

- Administrators will register and then log into the application or web portal to post, view, search and read the updates and will then they will be sent to students in a notification form.
- Students will register and then log into the application to search for information about college. They will also read, share notices from the administrators.
- The last group of users are the guest users. These don't require to register or login into the application but they just press the view button and they will be able to access resources like the college map, about college, courses and the administration.

## 2.4 Operating Environment

The application will only be available for the Android (Linux- based) operating system operating systems. The application shall only be used with compatible android devices. The user shall use this application on Android OS 4.2 jelly beans (API 23) or any later versions of the Android OS. The

app will rely on several functionalities built into Android's Application Programming Interface (API), so ensuring appropriate usage of the API will be a major concern. Beyond that, the application is mainly relies on google play services to be specific the google maps. So, for proper functionality, the device should have updated google play services app installed thus indirectly requires a user to have a Gmail account with Google. Because the app database will be stored on an external data-base though the app is built with a local database, it heavily requires internet connection for most real-time updates and functionality. But requires minimal internal data storage but high speed internet for best performance. Best results will be archived for GPS enabled android devices.

## 2.5 Design and Implementation Constraints

Implementation language restrictions

- The programming language shall be Java for the main application. The programming language shall be SQL for the cloud targets database.

Resource limits.

- The users‟ device shall have a working data plan or Wi-Fi connection.
- The users‟ device shall have sufficient memory storage to install the application.
- The users‟ device shall have sufficient battery life to run the application.

## 2.6User Documentation

### 2.6.1 Tutorial

- The user shall be able to see a tutorial the first time they start the application. The tutorial will be a series of splash screens that illustrates how to use each feature. This tutorial will not automatically show more than one time There will always be a "help" button available in the menu if user needs revisit the tutorial.

### 2.6.2 Online Help

- There shall be contact information posted in "about" page. The user shall be able to contact the developers regarding any issues they encounter.

## 2.7 Assumptions and Dependencies

Dependencies

- The application shall be used with the assumption that the Android API and licensing agreement remains the same.

Software Component Dependencies

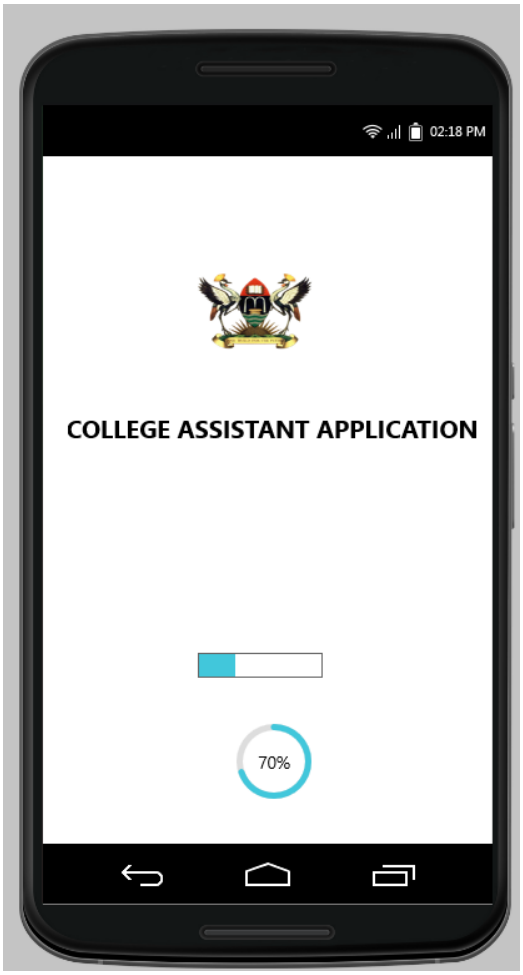
- The application shall be used with the assumption that the built in camera application operates correctly
- The application shall be used with the assumption that the device's network interface card and driver are operating correctly

# 3. External Interface Requirements

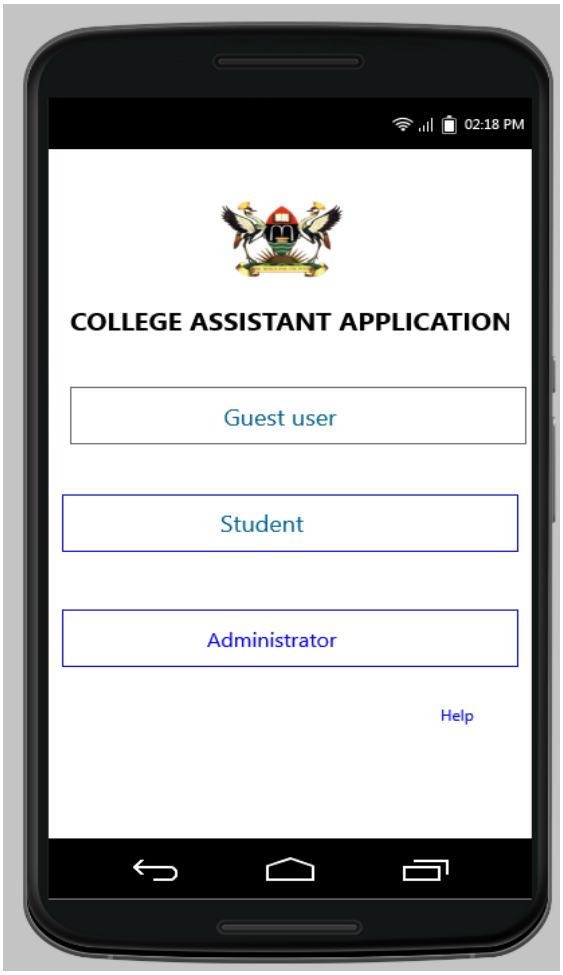
## 3.1User Interfaces

The interface will meet the following requirements to conform to the users‟ needs: It will be simple and easy to understand. Controls which allow the user to interact with the application will be clear and imply their functionality within the application.

1.Splash screen

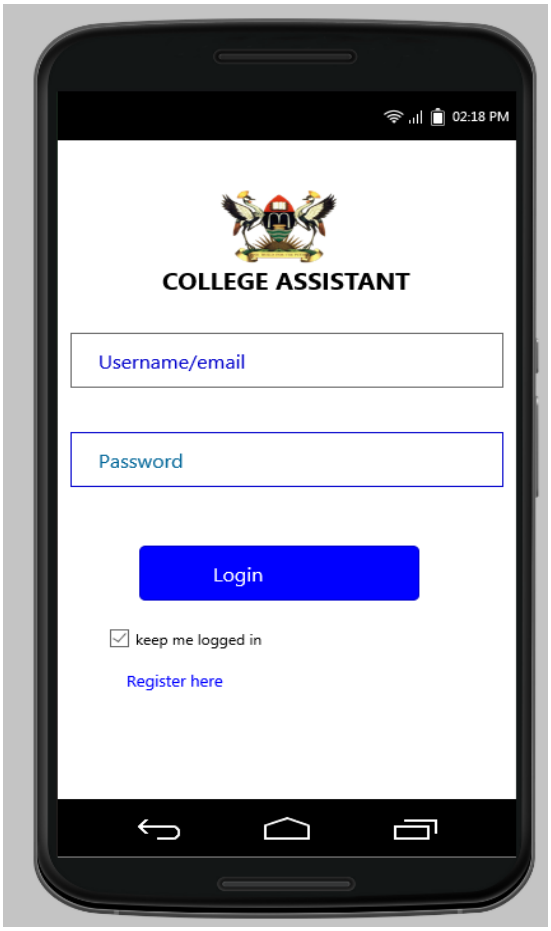



2. Users view

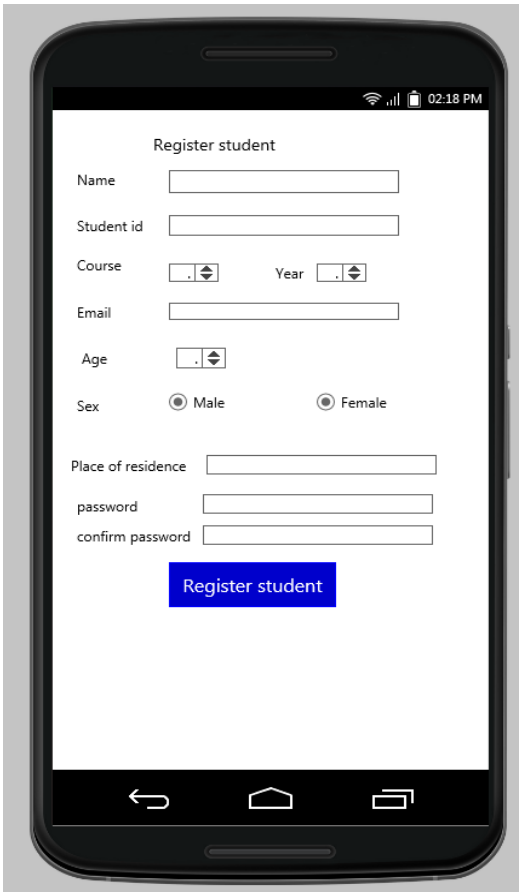


3. Guest user screen showing sliding tabs

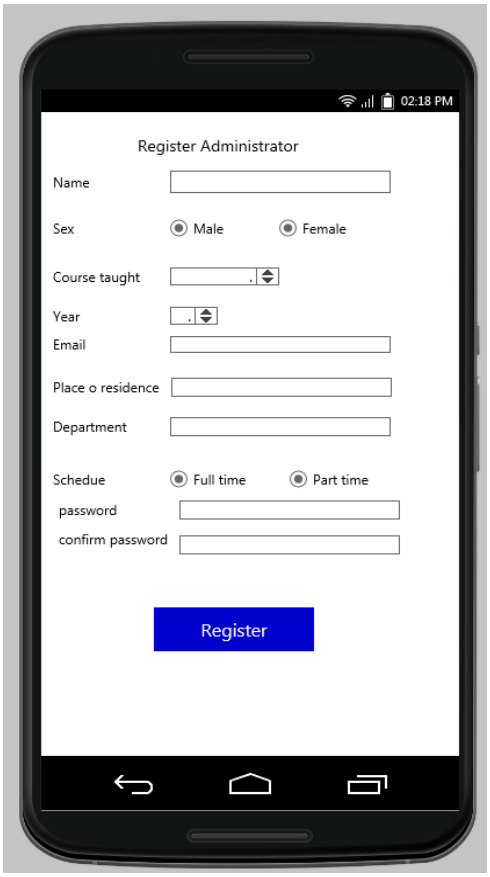

4.login screen for both student and administrator



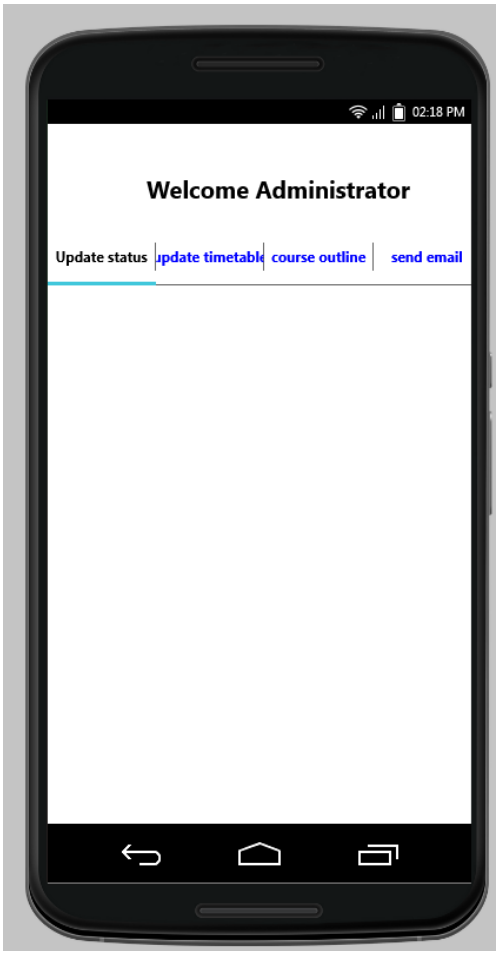
5.register student screen




6.register administrator screen



7.Welcome administrator screen




8.Welcome student screen


## *3.2* Hardware Interfaces

Android based college assistant application is a mobile application for the Android platform and hence is solely supported on Android mobile devices with aversion of jelly beans (4.2) and above and user should also have at least 5mb of space to save the application.

Messages, updates, real time notifications and data exchanged between the different devices are transmitted to and handled by the Firebase cloud messaging(FCM). It includes using SQLite, MSQL database. The application supports push messages both SMS and notifications inform of data between application and the main application server. Information will be sent using TCP/IP and

the HTTP protocol. The Android platform provides abstractions for all network communication interfaces and thus the hardware as well.

### 3.3 Software Interfaces

The app will have an Android operating systems which is developed using the Java JDK x64 bit version 8.0.50.13. Java Development Kit (JDK) contains the necessary tools to create, compile and package Android applications. And the Android SDK the latest version tools and android studio version 2.3.0 it is provided by Google. It is an integrated development environments (IDEs) to develop new applications, it also has default phone intents i.e. mailings, calling, messaging all at the cost of the local ISP

Android debug bridge (ADB):

The Android SDK contains the Android debug bridge, which is a tool that allows you to connect to a virtual or real android device, for the purpose of managing the device or debugging your application. Services needed include internet connection, it uses internet protocols (HTTP Connection) to store data.it uses two databases an external cloud database and a local SQLite database.

### 3.4 Communications Interfaces

The application communicates with the various databases including a web-based network server and software services via API function calls. Because the application will be written in Java, Java functions will make these calls to the APIs. The exact formats and protocols for incoming and outgoing messages should be abstracted by the APIs

The HTTP server will use a push protocol to push notifications of updates onto the Android phones. Furthermore, whenever a user opens the app a pull protocol will be used to retrieve and sync the latest updates from the server. The databases used include at the Back-end is SQLite. Client on Internet will use HTTP/HTTPS while on intranet will be using TCP/IP protocol.

## 4. System Features

Android college assistant application contains following features

1. Authentication
2. Notification
3. College Location

### 4.1 Authentication

- **Description and Priority**

  This feature will give the user a secure and simple login screen. The lecturer login screen is enabled for the college lecturer use only and is not accessible to any person. It has only limited and handful of input capability of limited number of administrator in the college. Any student cannot enter the routine settings and hamper the system.

- **Stimulus/Response Sequences**

  It will consist of two basic fields, Username inform of email and Password. A welcome message is boldly displayed at the screen defining the login is for the administrators only. There is a Login button for submitting the entered username and password. On successful entry, the user will be provided with the administrator control page to control all the settings of the database and on unsuccessful login the user is directed again to the same login page with an error message.

- **Functional Requirements**

  The most important function of the login page is to provide access only to the registered college lecturers and students. The application provides the admin role to lectures who will be modifying the different routines provided.

  REQ-1: Providing the registered lecturers to be listed as the administrators.

  REQ-2: Correct username and password to enter into the administrator controls.

### 4.2 Notification

- **Description and Priority**

  This feature will give the user real time update notification. The notification is enabled for students. Any registered student can get updated notifications intended for them as they are the obsolete users of the application.

- **Stimulus/Response Sequences**

  Only registered students can get updated notifications. Non-students are only provided with basic information about college campus, admissions procedures, courses offered.

- **Functional Requirements**

  The most important function of the notification is to provide important events and special communications at the college.

  REQ-1: For the first-time login, the application should have an internet connection to enable verification of the user account.

  REQ-2: The system provide procedure to follow /trouble shoot in case the user fails to login.

### 4.3 College Location

- **Description and Priority**

  The application provides Location services. An outsider can access the college by location.

- **Stimulus/Response Sequences**

  Application is connected to the Google maps, for route service.

- **Functional Requirements**

  This application requires a GPS-enabled android smartphone, Google play services with pre-loaded or online google maps for the location service

  REQ-1: For this functionality, the application should have an internet connection to enable connection to the online server.

  REQ-2: The application should have GPS enabled for accurate directions.

## 4.4 Registering user

- **Description and Priority**

  This is a highly priority feature which enables the user to create an account using the registration screen. This is done after installation and on the first run of the application. The assumption considered is that the person has not had an account before.

- **Stimulus/Response Sequences**

  When either the student or lecturer runs the application for the first time on his mobile phone. A registration screen is displayed prompting him/her to register.

  The screen consists of various fields i.e. name, student id, course, email, telephone number, age, sex etc. On successful entry, the user will click the register button and a toast message will be displayed showing status of registration.

- **Functional Requirements**

The most important function of the register screen register college lecturers and students. The application provides the reset password.

REQ-1: the application should have an internet connection to enable registration of the user account.

REQ-2: The system provides information showing registration status.

### 4.5 Uploading Files

- **Description and Priority**

  When the lecturer needs to upload, certain files inform of excel, pdf, word files. He uploads this document to this application's cloud server as a data source of the application. The feature is of a high priority to the lecturer

- **Stimulus/Response Sequences**

  While logged into the application user clicks on **"UPLOAD BUTTON**" while on the administrator screen. This can also be done using the web console/portal Clicks on.

- **Functional Requirements**

  **REQ-1:** The application must have an internet connection. uploaded files are stored on a cloud firebase server.

  **REQ-2:** Only pdfs, word, and excel files are accepted by the application.

### 4.6 deleting/editing notices

- **Description and Priority**

  The feature is of a low priority to the student and of a high priority to the administrator enables better management notices/updates. The administrator can delete, edit notices on the synchronized cloud server.

- **Stimulus/Response Sequences**

  When administrator is logged in and has internet connection. He uses the provided card views to edit or delete the pending notices on the server.

- **Functional Requirements**

  **REQ-1**: For this functionality, the application should have an internet connection to enable connection to the cloud server.

## 4.7 Add remainder

- **Description and Priority**

  This feature is of low priority enables the student to a set alarm for any due assignments, pending events.it improves time management of the student.

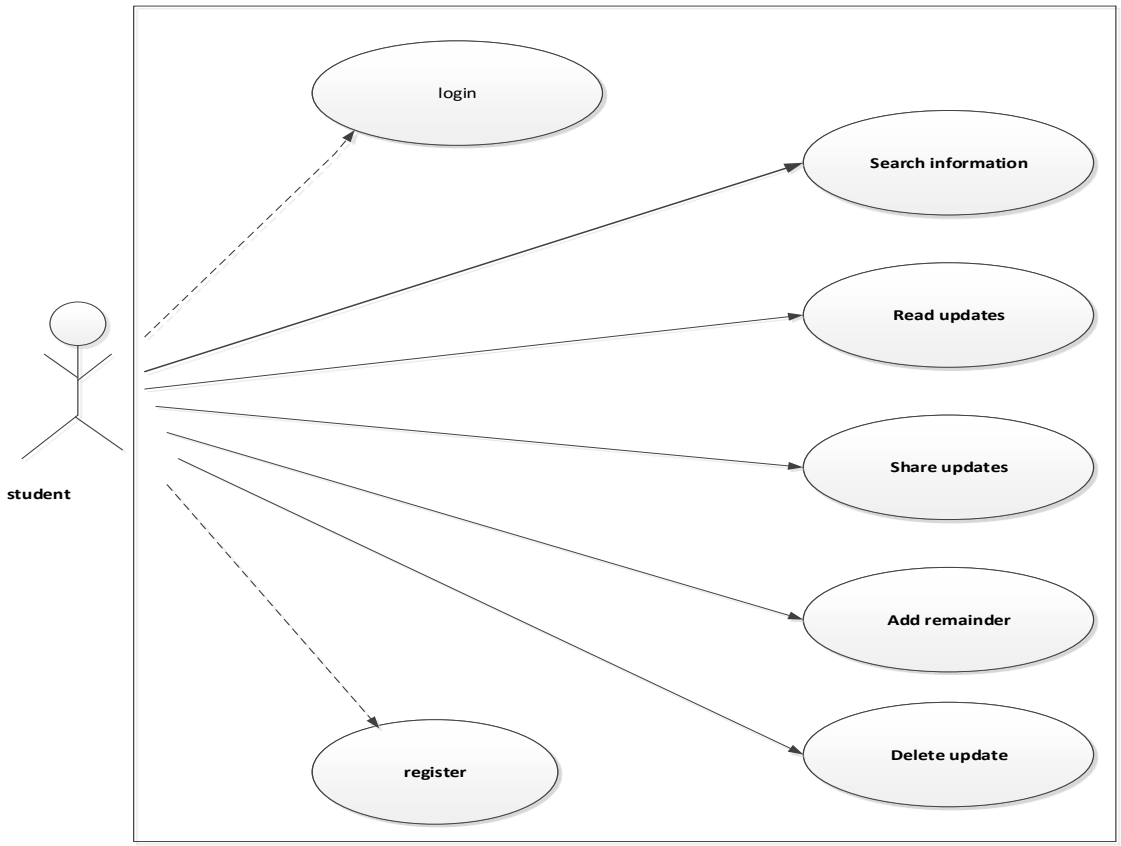
- **Stimulus/Response Sequences**

  While logged into the application in the welcome student screen the student can select date, time of the remainder and add notice to the alarm.

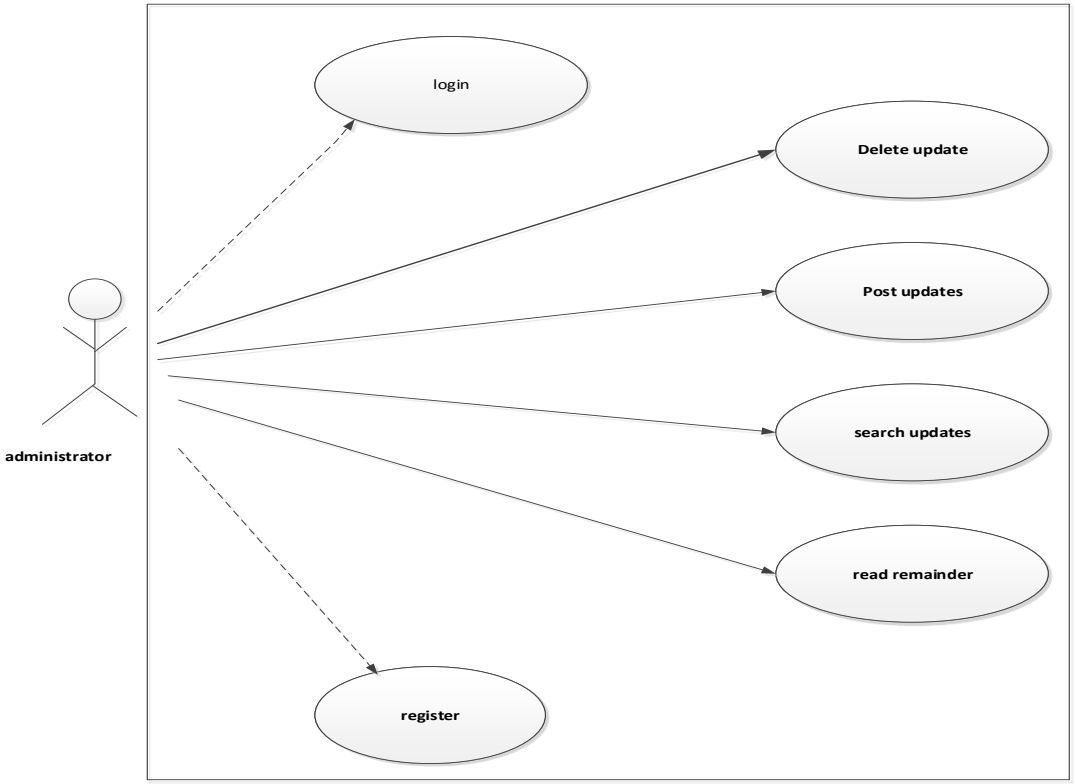
- **Functional Requirements**

  **REQ-1:** This is in the welcome student screen only

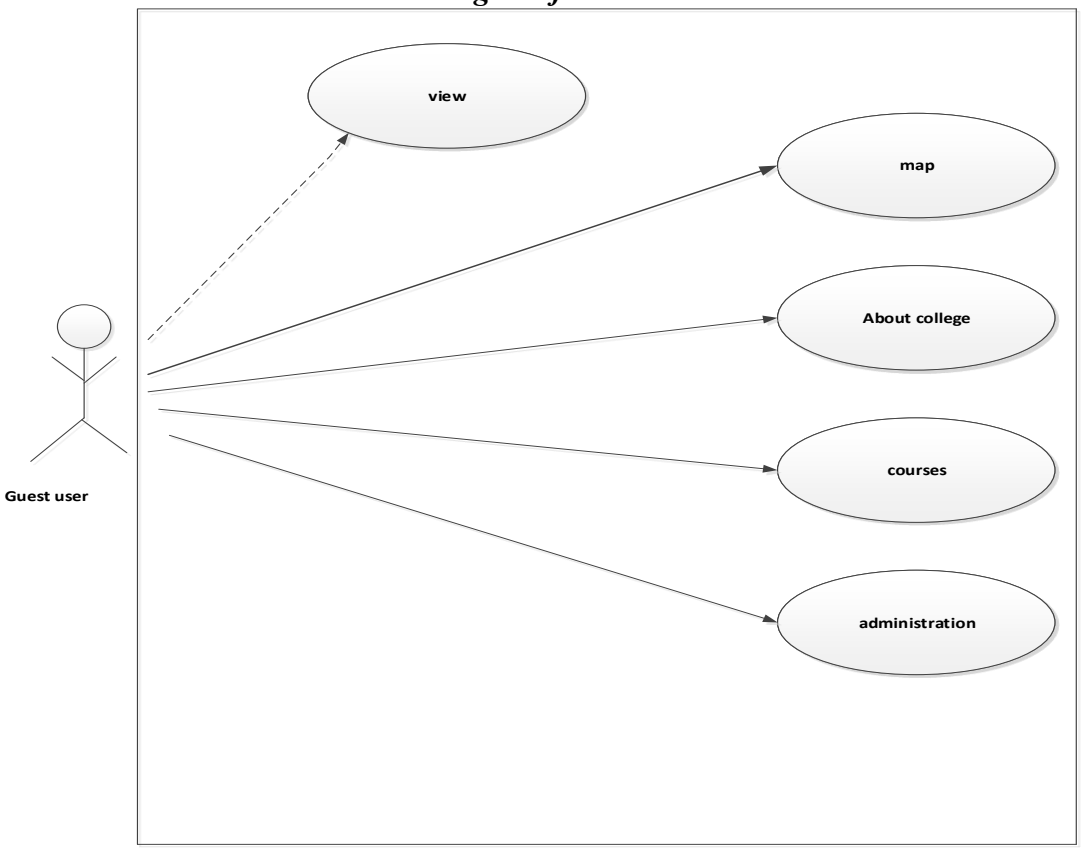
**Use case diagrams**




***Use case diagram for student***




***Use case diagram for administrator***




***Use case diagram for guest user***

# 5.Other Nonfunctional Requirements

## 5.1 Performance Requirements

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.

- **Real-Time**

  The application will provide up-to-date information. It should display the latest results at all times, and if it lags behind, the user should be notified.

- **System Resource Consumption**

  Resource consumption of this application should not reach an amount that renders the mobile device unusable. The application should be capable of operating in the background should the user wish to utilize other applications.

- **Multiple users**

  The software shall support use of multiple users at same time.

- **Usage of the Notice Information:**

  The notice link should be prominent and it should be evident that it is a usable link. Selecting the
  notice link should only take one click.

- **Response Time:**

  The response time should not be more than 5 seconds if user have a proper internet connection.

- **Fault Tolerance:**

  The fault tolerance of the system should be very good. If the system loses the connection to the Internet or the system gets some strange input, the user should be informed.

## 5.2 Safety Requirements

- User needs to sign in with their account to prove their identity (student) before using the application.
- User shall not use our application while driving or biking while finding directions for non-students for proper and accurate documents

## 5.3 Security Requirements

- Only recognized email addresses are required to verify the identity of the user upon opening the app. Also on opening the app, Authentication measures have been established for each user. Some features may require administrative password to use them.
- The security system features from having a login for the users to access the application's full features. So the chances of the application getting intruded are very less since the requirements below need to be satisfied :Login Requirements ,Password requirements(the password must contains one digit from 0-9, one lowercase character, one uppercase character, one special symbols in the list "#$%" and length of password must be at least 6 characters and maximum of 20. , Inactivity timeout, Email verification requirement(the application must validate and email address entered by the user before sending request to the server)
- Communication security:
  There should be security of the communication between the system and server. The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages. Every exchanged of information between client and server should be encrypted so that no one can track it.
- User Create Account Security:
  The security of creating account for users of the system should be maintained. If a user wants to create an account and the desired user name/email is occupied, the user should be asked to choose a different user name.
- Admin Login Account Security:
  If an admin tries to log in with a non-existing account for several times then the admin should not be logged in. The admin should be notified about log-in failure.

## 5.4 Software Quality Attributes

- **Reliability**
  The application will meet all of the functional requirements without any unexpected behavior. The application shouldn't at any time output, display incorrect or outdated information without alerting the user to potential errors and all data storage for user variables will be committed to the database at the time of entry.
- **Availability**
  The application will be available at all times on the user's Android device, as long as the device is in proper working order. The functionality of the application will depend on any external services such as internet access that are required. If those services are unavailable, the user should be alerted. All cached data will be rebuilt during every startup. There is no recovery of user data if it is lost. Default values of system data will be assigned when necessary.

- **Security**
  The software should never disclose any personal information of any users, and should collect no personal information from its own users.
- **Maintainability**
  The application users will be able to reset all options and all stored user variables to default settings
- **Portability**
  This software will be designed to run on any Android operating system version 4.2 or higher. The software will be forward compatible for all currently released Android operating system versions (up to 6.0)
- **Usability**
  The application should be easy to use. If any user is doing something wrong, he/she should be informed correctly what is going wrong behind the scene. There should be proper instructions for the user to use this application.
- **Testability:**
  Test environments should be built for the application to allow testing of the applications different functions.
- **Reusability**
  The application should be reusable by new students to the individuals to inform them about the college history for at least 10 years.

### 5.5 Business Rules

- The input system will allow for inputting numbers, operands, special symbols and letters of the alphabet and Only authorized members shall access the Authentication Logs
- Only users who have been granted permission and have authenticated, may access the provided privileges this applies to both student and administrator.
- The basic fact here is that material for upload and download is totally dependent on the user's discretion and network has no responsibility for that matter.

# 6.0 Other Requirements

A database for the application calls for a server side implementation that has information about the students, lecturers, and all the relationships involved. The following provides an example of information that may be stored in the database for example a student: full name, phone number, email, age, sex, student id etc. The external DB will be configured, and through use of PHP will allow interaction and processing in conjunction with the database. Processes to be done on the server include: pushing/pulling data, updating data, and generating notifications.
The android based college assistant application complies with quality assurance standards and keeps Personal information should be protected

## Appendix A: Glossary

### Definitions

| Definition | Description |
|---|---|
| Non-registered User | This is a user, which did not register to the application. He/she will have an ability to register to the application (i.e. to register as a Student, or as a lecturer) |

| | |
|---|---|
| Registered User | This is an abstract user role, which described additional allowed features for the users after they register to the application - like an ability access additional features of the application |
| Student | A student at pursing an academic award at the college activities. He/she will register to the application to get the services provided by the app. |
| Lecturer(admin) | The manager in the college assistant application. He/she will process update timetables, make communications to students from their parents |
| Guest user | This is unknown individual interested in knowing more about the college (He/she the to find location to the college, view about the college etc. |
| SQLite | SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is a compact library. |
| GPS | Global Positioning System |
| Linux | Linux is an open source operating system, with extensive documentation and user support. |

**Abbreviations**

| **abbreviation** | **Meaning** |
|---|---|
| SRS | Software Requirement Specifications |
| NFR | Nonfunctional requirement |
| FR | Functional Requirement |
| SDK | |
| JDK | Java Development Kit |
| GUI | Graphical User Interface |
| RAM | Random Access Memory |
| HTTP | Hypertext Transfer Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| HTML | Hypertext Markup Language |

## Appendix B: Analysis Models

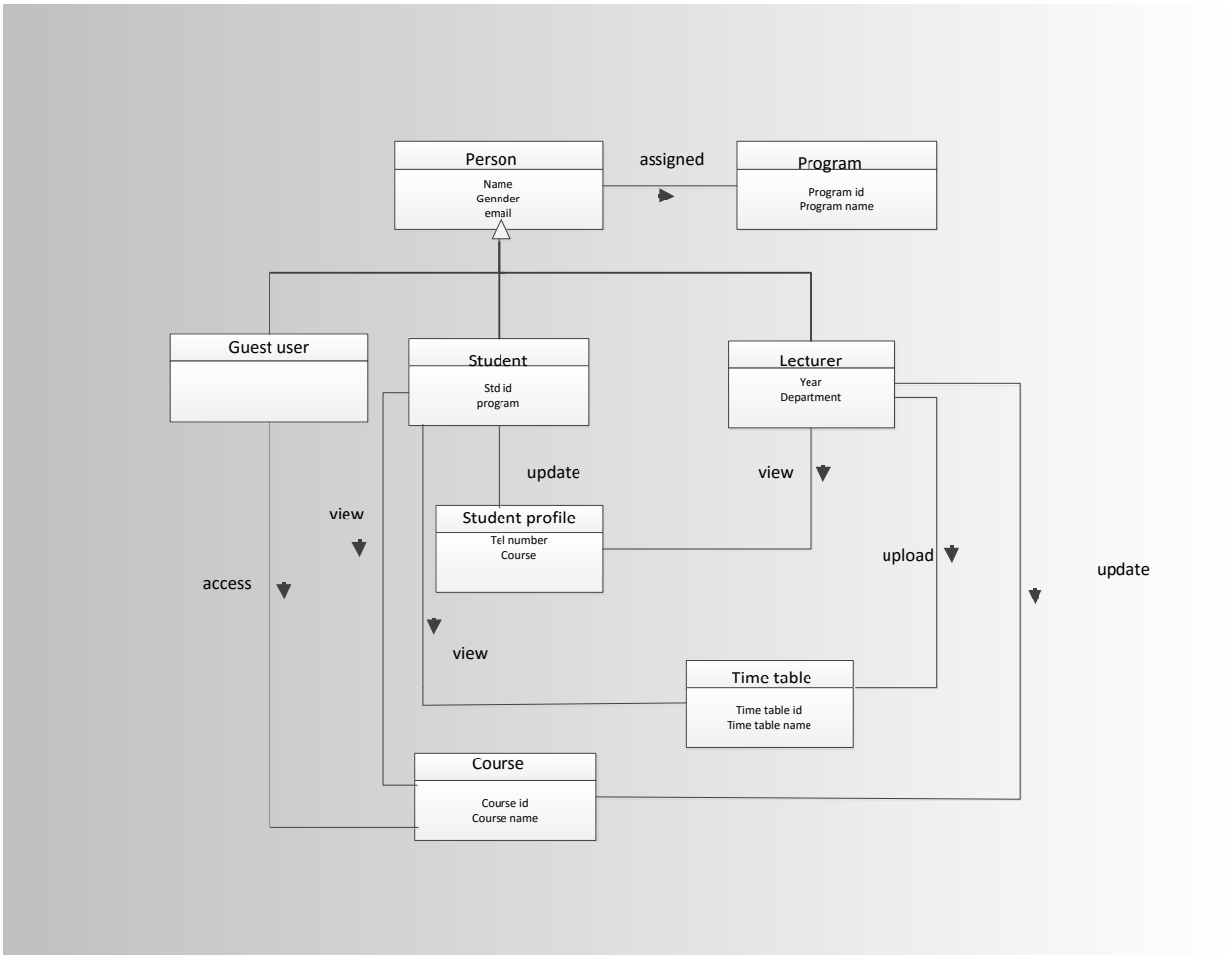



***Fig below shows entity-relationship diagrams***

## Appendix C: To Be Determined List

The document in this file is adopted from the IEEE *Guide to Software Requirements Specifications* (Std 830-1993).