ITU NOTIFIER

SWE600 Team Project

Submitted By

Bhavana Mamane

Bhavya Sahay

Disha Updhyay

Priyanka Jadhav

Shweta Sahu

Contents

1. GLOSSARY, ABSTRACT AND CONCEPT 7

1.1 Glossary **7**

1 .2 Project Abstract **8**

1 .3 Project Concept **9**

2 INTRODUCTION 9

2 .1 Introduction **9**

2 .2 Project Background **10**

2. 3 Problem Statement **10**

2.4 Proposed System: **11**

2.5 Identification Need **13**

2.6 Project Scope **13**

2.7 Unique Features of System **14**

3 PROJECT SCHEDULE 15

4 REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATIONS 16

4.1 Requirement Gathering 16

4.2 Requirement Specification 18

4.3 Functional Requirement 19

4.4 Non Functional Requirement 22

4.5 SDLC Model Used 26

4.6 Functional Specification Requirement 29

5 SYSTEM DESIGN 30

5.0 Introduction 30

5.1 Architecture Overview 30

5.2 System Architecture Diagram: 32

5.3 Detail Design 33

5.3 Methodology 35

6 DATABASE DESIGN 37

6.0 Database Design **37**

6.1 Data Dictionary **37**

6.2 Entity Relationship Diagrams **41**

7 UML DIAGRAMS 43

7.1 Use Case Diagram – Registration **43**

7.2 Use Case Diagram – Login **45**

7.3 Use Case Diagram – Announcement **47**

7.4 Use Case Diagram – Subscription **49**

7.5 Use Case Diagram – Broadcast **51**

7.6 Activity Diagram **53**

7.8 Class Diagram **57**

7.9 State Diagram **58**

7.10 Component Diagram **59**

7.11 Collaboration Diagram **61**

8. RISK MANAGEMENT 62

8.1 Purpose Of Risk Management Plan 62

8.2 Risk Management Procedure 63

8.2.2 Risk Analysis 63

8.3 Risk Response Planning 65

8.4 Risk Monitoring and Controlling 67

9 QUALITY AND TEST PLAN 71

9.1 Scope 71

9.2Test Plan Objective 71

9.3 Entry Criteria: 72

9.4 Exit Criteria: 72

9. 6 Testing Type 73

9.7 System Validation 76

9.8 Test Plan 78

9.9 Test Case – Registration 82

9.10 Test Case – Login 88

9.11 Test Case – User Home Page **91**

9.12 Test Case – Announcement **92**

9.13 Test Case – Subscription **94**

9.14 Test Case – Broadcast Message **96**

9.15 Test Case – View Profile **97**

10 INTERFACES /SNAPSHOTS 99

10.1 Login Page **99**

11 SUMMARY, FUTURE SCOPE AND OPPORTUNITIES 99

11.0 Introduction **99**

11.1 Summary **99**

11.2 Future Scope **99**

11.3 Opportunities and Lesion Learned **100**

11.6 References **100**

1. Glossary, Abstract and Concept

1.1 Glossary

Student: A user who is registered as a Student.

Professor: A user who is registered as a Professor.

Librarian: A user who is registered as Librarian.

SGA: Student Government Association.

SGA Staff: A user who is registered as SGA staff.

ISO: International Student Office.

ISO Staff: A user who is registered as ISO staff.

ITU: International Technological University.

Admin: A user who is registered as an Admin.

DB: Database.

SDLC: Software Development Life Cycle.

FCM: Firebase Cloud Messaging.

1 .2 Project Abstract

The ITU Notifier is a mobile app that can be used to send and receive online notices issued by ITU on the mobile phones of registered users.

A notice board is an essential part of an educational organization which students and staff can use for making announcements and providing information. ITU Notifier is an electronic notice and announcement system which will bring the ITU notice board virtually to your mobile phones. It will enable staff and students to send and receive notification alerts. This app will not only help to keep the users updated on what is going on in the college but will also make it easy them to broadcast their messages anywhere anytime.

Key features of ITU Notifier will be:

1. User registration: Students and staff will have to first register to the app with their ITU email id to use it.
2. Notification post: Registered users will be able to send notifications from the app.
3. Notification alert: Students and staff will be notified of new postings via notification alerts.
4. User response: Users will be able to respond to notices.
5. User roles: There will be a set of pre-defined roles in the system.
6. Groups: There will be different groups for example groups for different ongoing courses, groups for classes and so on. Users will have the option to post notices to any specific group instead of broadcasting to the entire college.
7. Notice categories: Every notice will have a category for example, advertisement, lost and found, information etc.

1 .3 Project Concept

Purpose:

The purpose of proposed system is to overcome all the limitation and drawback existing system. The ITU Notifier is user friendly android application. The main objective of this application is simplicity, easy to use and convenient for all college staff and students. It will help everyone who would be part of college to collect information about all the events going on in college premises.

The main purpose **of ITU Notifier** can be enumerated as follows:

1) Easy and simple way to broadcast message

2) Helps you to update with what's going on in college

3) Good way to advertise about the coaching, courses and different tuitions

4) Different group categories will be available, use can follow different group notice board.

5) Easily notified for any lost and found in college.

6) Faster discrimination of notices regarding education, technical events, cultural events.

2 Introduction

2 .1 Introduction

The purpose of proposed system is to overcome all the limitation and drawback existing system. The ITU Notifier is user friendly android application. The main objective of this application is simplicity, easy to use and convenient for all college staff and students. It will help everyone who would be part of college to collect information about all the events going on in college premises.

2 .2 Project Background

ITU is a university provides excellence in education for the future leaders of Silicon Valley corporations. It offers different courses certification programs and many more professional courses. Like every university or college, here also lot of events take place. Notice board is always full of information. As we all know notice board is the easiest way to broadcast the message to all students, professor basically all members who are directly or indirectly involved in days todays college activity. So considering importance of Notice board, in this modern era it's a need to access this notice board from anywhere anytime. So this brings up this app called ITU Notifier. Below are few points which are major background points behind building this application.

1) Manual maintenance of notices is a tedious job.so easy to broadcast notices and easy to access.

2) Giving the facility to convey message to all students anytime and anywhere.

3) Making students upload about all the events and activities going to the college

4) Students will not require standing in the crowd to see the notice. There will be no fighting in order to see the notice first. Everyone will be first who can see notice as soon as it broadcasts.

5) Saves lot of time.

6) Professor can also post updates regarding upcoming courses or exams to specific group of student's or all students based on the requirement.

7) Admin staff can also broadcast notice as per their requirement to group of students or professors.

2. 3 Problem Statement

To develop a mobile application that will help you receiving the notices from the college anywhere and anytime. Earlier there was a problem that notices were pasted on notice board. If there holiday next day, nobody will be able to read it. Moreover any update to website is also very difficult .And it's very common that we feel lazy to go and update the website data. The more easy way is just

type a message sitting wherever and click a button to send and done, your message will be broadcasted. It will notify all the registered users including staff, students, admin etc..

Currently our college has manual system to put notices on notice board. As nobody has time to rush in order to read the notices on noticeboard, so its outdated now.

Limitation of existing system:

1) **Order of Data:** Notice can get out of order on traditional notice board system. If someone accidently put some data in the wrong place, it can lead to lost data. Automated notice management system allow users to quickly check whether information already exists somewhere in the system which helps avoid problems like redundant data.
2) **Complexity:** Automated systems are less complex than manual system for handling notices which can make it easier for untrained people to access and manipulate data. Anyone having basic knowledge of Mobile can use this automated system.
3) **Inconsistency of Data:** There will be unavailability for future use, since notice might get misplaced during manual notice management. Difficulty to keep track of all notices using manual method**.**
4) **Damage:** Manual notices stack are vulnerable to damage, destruction and theft in ways that digital notices are not. A company may back up its digital data both on site ensuring security if the office building suffered a fire or any type of disaster. A manual database however many only exists at one place without any copies. As a result manual database will vulnerable to fire and natural disaster.

   In addition while access time in manual database system information must be found by hand rather than electronically. While a digital database will typically allow user to search the entire database system for specific information within fraction of second. On the other hand someone looking in manual system may have to spend hours for searching for particular records.

2.4 Proposed System:

Proposed system will be able to do the following:

1) **To eliminate wastage time and energy**: ITU E-Notifier will be able to save lot of time and paper. It directs both teacher and students energy to one thing at a time by placing proper persons at their proper places at proper time. Everything will be instantaneous.
2) **To avoid duplication and Overlapping:** This application will help to remove duplicity of notices. Admin person can manage the notices based on the different categories. Student and staff will be having the correct information.
3) **To ensure attention of students to each and every notice:** This app ensures that everyone has kind attention to every notice and updates going on college. There will be buzz at each and every notice to drive the attention of students, professors and respective concerned groups.
4) **To bring the system into college life:** This would be the easiest method to notify all the concerned members at right time. In absence of this application it is quite difficult to inform students at right time.
5) **Searching particular Notice:** This application will allow all students and professors to search notices based on different fields like notice date, title, category.
6) **Free Service**: It gives free service to notify all the students. There will be no cost involved for sending notifications.
7) **Prevent Crowd in college**: As we see there is always crowd near notice board. Someone may miss the information if he/she is absent on that particular days. With this application there will be no crowd near notice board. There is no need to visit college especially for notices, one can access anywhere anytime. This way no notices can be missed by students, professors and college staff members.
8) **Automatically Updated Dashboard**: The dashboard of notice is automatically updated when new message arrives.
9) **Anytime anywhere service**: With this application notices will be delivered anytime anywhere. There is no restriction of time to send to read notices.
10) **Keeping Notices at one place**: This application allow you to have notices in one place only.

2.5 Identification Need

1. As we discussed earlier that manual maintenance of a notices is a tedious job. So to enhance the ease of working, we go for this package.
2. Giving the facility to convey messages to all students anytime and anywhere.
3. Making students updated about all the events and activities going on in the college.
4. The student will not require to stand in the crowd to see the notice. There will be no issue of fighting in order to see the notice first. Everyone is first to see that notice inside their own mobile phone anywhere and anytime.
5. The least but most important it saves time.
6. Utilizing less man power. As there are many persons involved in circulating the message. With this application, only one person is required to post the notice. Rest of the man power is saved in the entire process.

2.6 Project Scope

”ITU Notifier” is a mobile application i.e an Internet based Mobile application. A mobile application is a computer program designed to run on smartphones, tablet computers and other mobile devices. Apps are usually available through application distribution platforms, which began appearing in 2008 and are typically operated by the owner of the mobile operating system, such as the Apple App Store, Google Play, Windows Phone Store, and BlackBerry App World.
Mobile apps were originally offered for general productivity and information retrieval, including email, calendar, contacts, and stock market and weather information. However, public demand and the availability of developer tools drove rapid expansion into other categories, such as mobile games, factory automation, GPS and location-based services, banking, order-tracking, ticket purchases and recently mobile medical apps. The explosion in number and variety of apps made discovery a challenge, which in turn led to the creation of a wide range of review, recommendation, and curation sources, including blogs, magazines, and dedicated online app-discovery services. Mobile application development is the process by which application software is developed for low-power

handheld devices, such as personal digital assistants, enterprise digital assistants or mobile phones. These applications can be pre-installed on phones during manufacturing, downloaded by customers from various mobile software distribution platforms, or delivered as web applications using server-side or clientside processing (e.g. JavaScript) to provide an ”application-like” experience within a Web browser. Application software developers also have to consider a lengthy array of screen sizes, hardware specifications and configurations because of intense competition in mobile software and changes within each of the platforms. Mobile app development has been steadily growing, both in terms of revenues and jobs created. The popularity of mobile apps has continued to rise, as their usage has become increasingly prevalent across mobile phone users.

2.7 Unique Features of System

The unique features of this application are as follow:

1. Google FireBase Message Notification: Google firebase Messaging has been used to broadcast notices to all the students who are registered with this application. It will help you receive the notice even if you application is closed. So it implements anywhere anytime notifications. Google Firebased Messaging (FCM) is a free service for sending messages to Android devices. FCM messaging can greatly enhance the user experience. Your application can stay up to date without wasting battery power on waking up the radio and polling the server when there are no updates. Also, FCM allows you to attach up 1,000 recipients to a single message, letting you easily contact large user bases quickly when appropriate, while minimizing the work load on your server.

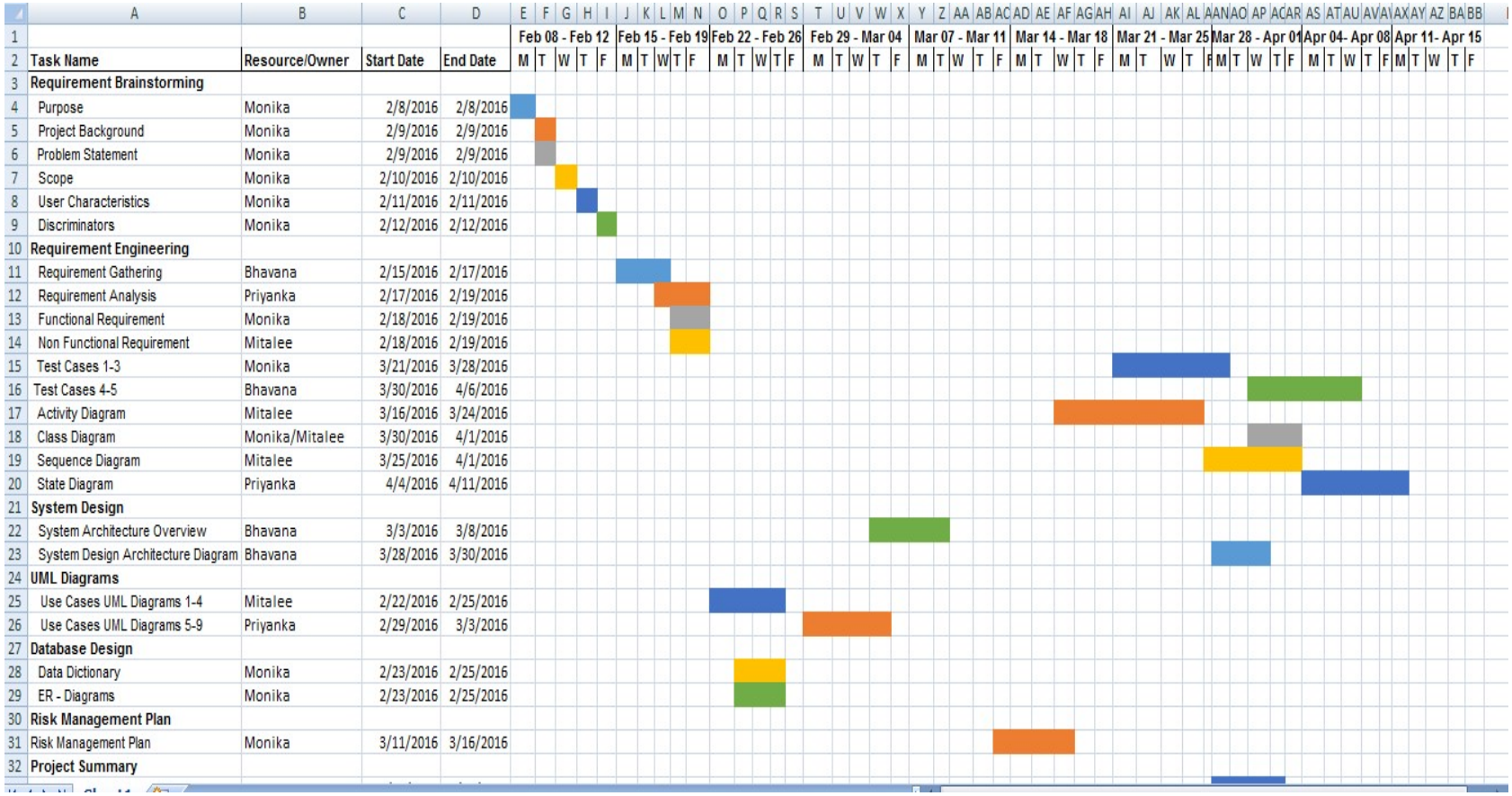
Scalable for all universities or schools

The solution scales to fit a single school with few categories or go up to all categories and all departments linked to school.

Workflow efficiency

No more manual maintain of Notices are required. Easy and simple way to post notices and read notices and keep everyone updated with what all events are happening around.

3 Project Schedule



Task Name	Resource/Owner	Start Date	End Date
Requirement Brainstorming			
Purpose	Monika	2/8/2016	2/8/2016
Project Background	Monika	2/9/2016	2/9/2016
Problem Statement	Monika	2/9/2016	2/9/2016
Scope	Monika	2/10/2016	2/10/2016
User Characteristics	Monika	2/11/2016	2/11/2016
Discriminators	Monika	2/12/2016	2/12/2016
Requirement Engineering			
Requirement Gathering	Bhavana	2/15/2016	2/17/2016
Requirement Analysis	Priyanka	2/17/2016	2/19/2016
Functional Requirement	Monika	2/18/2016	2/19/2016
Non Functional Requirement	Mitalee	2/18/2016	2/19/2016
Test Cases 1-3	Monika	3/21/2016	3/28/2016
Test Cases 4-5	Bhavana	3/30/2016	4/6/2016
Activity Diagram	Mitalee	3/16/2016	3/24/2016
Class Diagram	Monika/Mitalee	3/30/2016	4/1/2016
Sequence Diagram	Mitalee	3/25/2016	4/1/2016
State Diagram	Priyanka	4/4/2016	4/11/2016
System Design			
System Architecture Overview	Bhavana	3/3/2016	3/8/2016
System Design Architecture Diagram	Bhavana	3/28/2016	3/30/2016
UML Diagrams			
Use Cases UML Diagrams 1-4	Mitalee	2/22/2016	2/25/2016
Use Cases UML Diagrams 5-9	Priyanka	2/29/2016	3/3/2016
Database Design			
Data Dictionary	Monika	2/23/2016	2/25/2016
ER - Diagrams	Monika	2/23/2016	2/25/2016
Risk Management Plan			
Risk Management Plan	Monika	3/11/2016	3/16/2016
Project Summary			

4 Requirement Analysis and System Specifications

4.1 Requirement Gathering

4.1.1 Feasibility Study

Depending on the results of the initial investigation, the survey is expanded to a more detailed feasibility study. Feasibility study is a test of system proposal ac- cording to its work ability, impact on the organization, ability to meet user needs, and effective use of resources. The objective for this phase is not to solve the problem but to acquire a sense of scope. During the study, the problem definition is crystallized and aspects of the problem to be included in the system are deter- mined.

Mobile Application Development Systems are capital investments because resources are being spent currently in order to achieve benefits to be received over a period of time following completion. There should be a careful assessment of each project before it is begun in terms of economic justification, technical feasibility, operational impact and adherence to the master development plan. We started the project by listing the possible queries that the user might want to be satisfied. And on these lines we guided the project further.

The three main points, kept in mind at the time of project, are:

_ Possible (To build it with the given technology and resources)

_ Affordable (given the time and cost constraints of the organization)

_ Acceptable (for use by the eventual users of the system

The three major areas to be considered while determining the feasibility of a project are:

1. Technical Feasibility: The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipment's have the technical capacity to hold the data required to use the new system?

- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of Secure Infrastructure Implementation System. The current system developed is technically feasible. It is a web based user interface. Thus it provides an easy access to the users. The databases purpose is to create, establish and maintain a work- flow among various entities in order to facilitate all concerned users in their various capacities OR roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hardware requirements for the development of this project are not many and are already available as free as open source. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing a fast feed- back to the users irrespective of the number of users using the system.

2. Operational Feasibility: Under this category of service we conduct a study to analysis and determine whether your need can be fulfilled by using a proposed solution. The result of our operational feasibility Study will clearly outline that the solution proposed for your business is operationally workable and conveniently solves your problems under consideration after the proposal is implemented. We would precisely describe how the system will interact with the systems and persons around. Our feasibility report would provide results of interest to all stakeholders. It will do as per the needs of the business requirements.

3. Timeline Feasibility: It is important to understand that a need must be fulfilled when it has to be. Some otherwise feasible and highly desirable projects can become non-feasible due to very restrictive timeline constraints. This fact makes it imperative that milestones are clearly linked to the timeline and projects are well conceived with safe unforeseen margins. We make sure that we strictly follow what has been stated above.

Observation

Before design a system for the clinic, do visit to the clinic to do an observation on how is their daily business operation It is an ordinary clinic which done everything manually without any technology help Observed and record down in detail how is the progress flow from patient register on the counter until the patient take medicine on the dispensary counter Through the progress, noticed that there's a lot of paper needed to use in order to complete this progress We asked question to them when we found out some problem that we do not truly understand on it and the staff was kindly answer every question and explain to me in detail Furthermore, the staff had shown us several type of document that they usually do for reference purpose Most of the document is done manually in paper by the counter staff includes calculation on stock, generate receipt, generate MC and etc

Interview

End user is the one who are going to have the interaction with the system, thus interview the end user is one of the most effective ways on gathering facts Throughout the interview session we can get the exact information and definite requirement that the end user expected on the system Students and Admin staff ,Professor as they may have different requirement as the category that they carry out is different from each other However, the current model of the university is having old traditional notice board system. So when it come to the design stage, must consider the simple and easy way of keep all students updated with the notices.

4.2 Requirement Specification

4.2.1 Data Requirements

Data requirement is meant to be the data that will be used in our application. Data required in this project is all notices that need to be conveyed to the user. This application also require the username and passwords of persons in order to register them and sending notification about updates. So two main requirements are:

--Notice Details

--User

4.3 Functional Requirement

In order to make this application functional, we require the following:

Download mobile application:

A user should be able to download the mobile an application through either an application store or similar service on the mobile phone. The application should be free to download.

ITUN001: User registration

The system shall allow guest user/visitor to browse the Medicine information and Physician information

- Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application.
- The user must provide password and e-mail address. The user can choose to provide a regularly used phone number.
- Email address should be ITUs email address. Email id other than students.itu.edu domain should not be allowed to register.

ITUN002: User Login:

- Given that a user has registered, then the user should be able to log in to the mobile application with email id and password.

- The log-in information will be stored on the phone and in the future the user should be logged in automatically.

ITUN003: Account Preferences:

- User should be able to subscribe and unsubscribe different categories for receives notices. Like lost and found, ITU Talent ,SGA ,ISO

ITUN004: Reset Password:

- Given that a user has registered, then the user should be able to retrieve his/her password by e-mail.

ITUN005: Dashboard:

- Given that a user is logged in to the mobile application, then the first page that is shown should be the dashboard page. The user should be able to see all the college notices based on the category subscription.

ITUN006: Search Notice:

- The user should be able to search for a notice by its title, date posted, and category. For example, if a user types fee, all the notices having fee in their content get displayed.

ITUN007: Selecting a Notice:

- A user should be able to select any notice from list view. The click on particular notice will take him to notice details of that particular notice.

ITUN008: Navigating back to Notices List:

- The user should be able to navigate back to notices list from the notice details section. This is required to give a good user experience.

ITUN009 Deleting Notices:

- The user should have the option to delete the unnecessary notices from his phone, by ticking them one by one and then deleting those in one go. This way, user can save this phone memory from unrequired notices

ITUN0010: Posting Notices:

- The admin, student and all available role users of this application should be able to post the notices. He should be able to add a picture within notices. That picture can be taken either from gallery or by using the camera of the mobile phone.

ITUN0011: Notification Alert:

- All the registered users should be able to have a ping or notification on their mobile phone whenever a new notice is posted.

ITUN0012: Dashboard:

- Newly posted notices will be flashed on the dashboard page based on the logged in user category subscription.

ITUN0013: Database:

The system shall use the Mysql Database

ITUN0014: Operating System.

The Development environment shall be Android Studio and supporting SDK for Mobile development

4.4 Non Functional Requirement

4.4.1 Performance

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

Prominent search feature:

- The search feature should be prominent and easy to find for the user.

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.

4.4.2 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.

4.4.3 System Dependability:

Following are the requirements that an application require from the device/mobile on which it is installed.

Internet Permission:

Application developed, require full internet permissions of mobile so that it can fetch notices from the server. At the same time, it should be able to receive buzz or notification tone whenever new notice is posted by admin.

System Tools:

This application require various system tools to be used. For example, it requires Camera of mobile in order to click the image and post in into notice. It also require system tool that prevents it from sleeping.

Hardware Control:

It uses vibrator of mobile phone whenever any notification arrives.

4.4.4 Maintainability Requirements

Following are the maintainability requirement of ITU Notifier mobile application:

Application Extendibility:

The application should be easy to extend. The code should be written in a way that it favors implementation of new functions. It is requires in order for future functions to be implemented easily to the application.

Application Testability:

Test environments should be built for the application to allow testing of the applications different functions.

4.4.5 Security

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.

Admin Login Account Security:

If an admin tries to log in to the web portal with a non-existing account then the admin should not be logged in. The admin should be notified about log-in failure.

Admin Account Security:

There should be security of admin accounts. An admin and IP address should not be able to log-into the web portal for a certain time period after three times of failed log-in attempts.

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 is occupied, the user should be asked to choose a different user name.

4.4.6 Look and Feel Requirements:

Easy to Use

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.

Soft Sound Notification:

The sound for notification should be very soft. It should not disturb the peers with a loud note. Everything should be sober in this application.

4.4.7 Validations:

Validations are important for all type of applications. There should be a way to validate the user input first before sending the user request to the server. Following are the validations implemented in proposed system:

User Password Validation:

The application should check the user and password fields before sending any request to the server. It should check whether the fields are filled or not. if fields are not filled up, user should be instructed to fill up the fields before moving further. in this way, there will be less traffic on the server.

Validations during Registration:

There are a lot of validations that needs to be implemented in the application. They are as follow:

1. First and Last Name of User:

The first and last name of user should be not null. Also first letter of first and last name should be in uppercase.

2. Username:

The username should be the email id of ITU. Email id other than studnets.itu.edu won't be able to register here. Only ITU students will be able to register with this app.

3. Password:

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.

5. Mobile Number:

The mobile number should be of only ten digits. No more, no less than that.

6 Validating During Posting Notices:

The application should validate the notice posting fields before posting any notice. It should check whether title and description fields are filled or not. if not, it should tell the user to fill up the required fields while posting the notice.

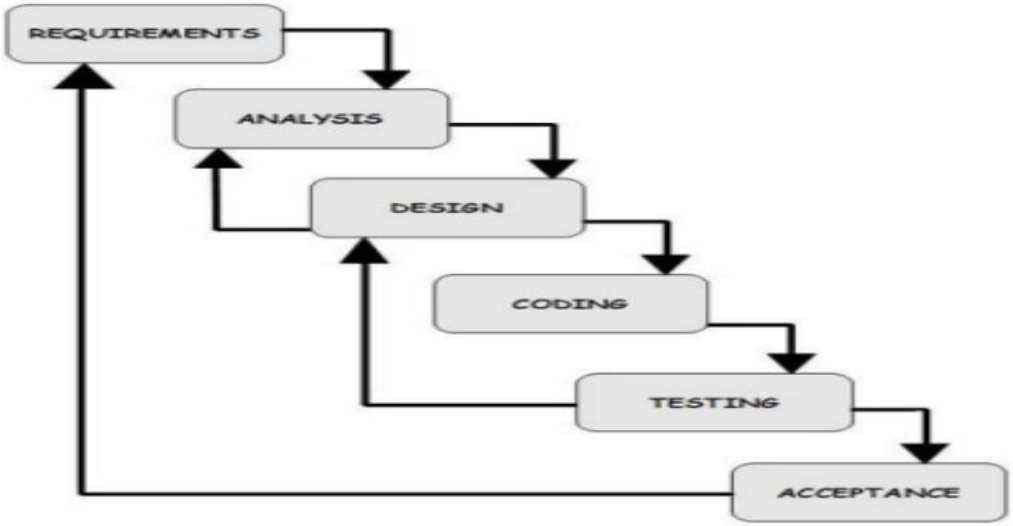
7. Reset Password Validation:

The application should check that user has entered the username or email in the given filed before pressing the reset password button.

4.5 SDLC Model Used

This section describes the project as per the various stages of the Software Development life cycle. The model of software development life cycle used in this project is the waterfall method. The Waterfall Method is comprised of a series of very definite phases, each one run intended to be started sequentially only after the last has been completed, with one or more tangible deliverables produced at the end of each phase of the waterfall method of SDLC. Essentially, it starts with a heavy, documented, requirements

planning phase that outlines all the requirements for the project, followed by sequential phases of design, coding, test-casing, optional documentation, verification (alpha-testing), validation (beta-testing), and finally deployment/release.



1. Requirement Analysis:

Existing system is time consuming and it makes difficultlt to convey huge amount of users about any event, class or seminar almost instantly. Also there is always a big crowd in front of noticeboard. So it was hectic to read any useful instruction and information.

Thus all the problems of the existing system are summarized and proposing a new system that works as an online application.

It is a value added solution to the problem. It resolves all the problems stated above. It will provide simple interface to the user to operate on and convey the intended users about events almost instantly, anytime and anywhere.

2. Design:

It includes translation of the requirements specified in the SRS into a logical structure that can be implemented in a programming language. The output of the design phase is a design document that acts as an input for all the subsequent SDLC phases. The design of this app is simple and user friendly containing six main activities, namely:

(a) Register

(b) Login

(c) Dashboard

(d) Details of Notices

(e) Account Preference

(f) Reset Password

3. Coding/Implementation:

It includes translation of the requirements specified in the SRS into a logical structure that can be implemented in a programming language. The output of the design phase is a design document that acts as an input for all the subsequent SDLC phases. The project is implemented using the Android virtual devise (AVD). This emulator helped to implement the project in a real-like environment and sketch out the details of how it will work on a real hardware. Each activity is linked with another and interconnectivity is transparent and smooth.

4. Testing:

It includes detection of errors in the application. The testing process starts with a test plan that recognizes test-related activities, such as test case generation, testing criteria, and resource allocation for testing. The code is tested and mapped against the design document created in the design phase. The output of the testing phase is a test report containing errors that occurred while testing the application.

Testing of the project has not been done on real hardware and also on the emulator or software environment. Testing has been done for each of the individual activities of the project.

5. Maintenance:

It includes implementation of changes that software might undergo over a period of time, or implementation of new requirements after the software is deployed at the customer location. The maintenance phase also includes handling the residual errors that may exist in the software even after the testing phase. The project maintenance is low cost and efficient as user will get this application at free of cost and also this application is shared over network, therefore maintenance is little bit difficult.

4.6 Functional Specification Requirement

S.No.	Requirements(Functional)
FR1	ITU User should be able to register account with credential(s). First name of User Last Name of User Unique registration account Password
FR2	ITU User should be able to login with unique id and updated password in system.
FR3	ITU User should be able to send notification to ITU domain or different domain.
FR4	ITU User should be able to receive notification from same domain or different domain.
FR5	ITU User should be able to save important link and information i.e. create notification and should be

	able to use it to send email.
FR6	ITU User should be able to read all notifies.
FR7	ITU User should be able to send attachment(jpeg, gif,png,pdf, ms office, zip) in ITU Notifies
FR8	ITU User should be able to get relevant broadcast message as soon as its published
FR9	ITU User should be able to organize notifies under different category

5 System Design

5.0 Introduction

The Architecture Overview includes models that describe the high level logical architecture A key section of the document is to provide a list of Architecture decisions that are being made. This outlines the key decisions made for the ITU Notifier architecture and the pros/cons of alternatives

5.1 Architecture Overview

5.1.1 Solution Objectives

ITU Notifier aims at improving the Broadcasting and posting Notices process for day to day activities like Lost and found, Cultural events happening in the college/university. ITU Notifier is a Andriod application and it provides the most easiest system for the university by supporting all types of users: Students, Professor, Admin Staff,Librarian, and all other people associated with university.

The system has provision to all the users the complete and simple process for keeping everyone updated with all the events happening in college. Professor can also announce important exam or course related information using this port very easily

5.1.2 Target Audience, Users and Usage

The target audience for ITU Notifier can be broken down in 6 major categories:

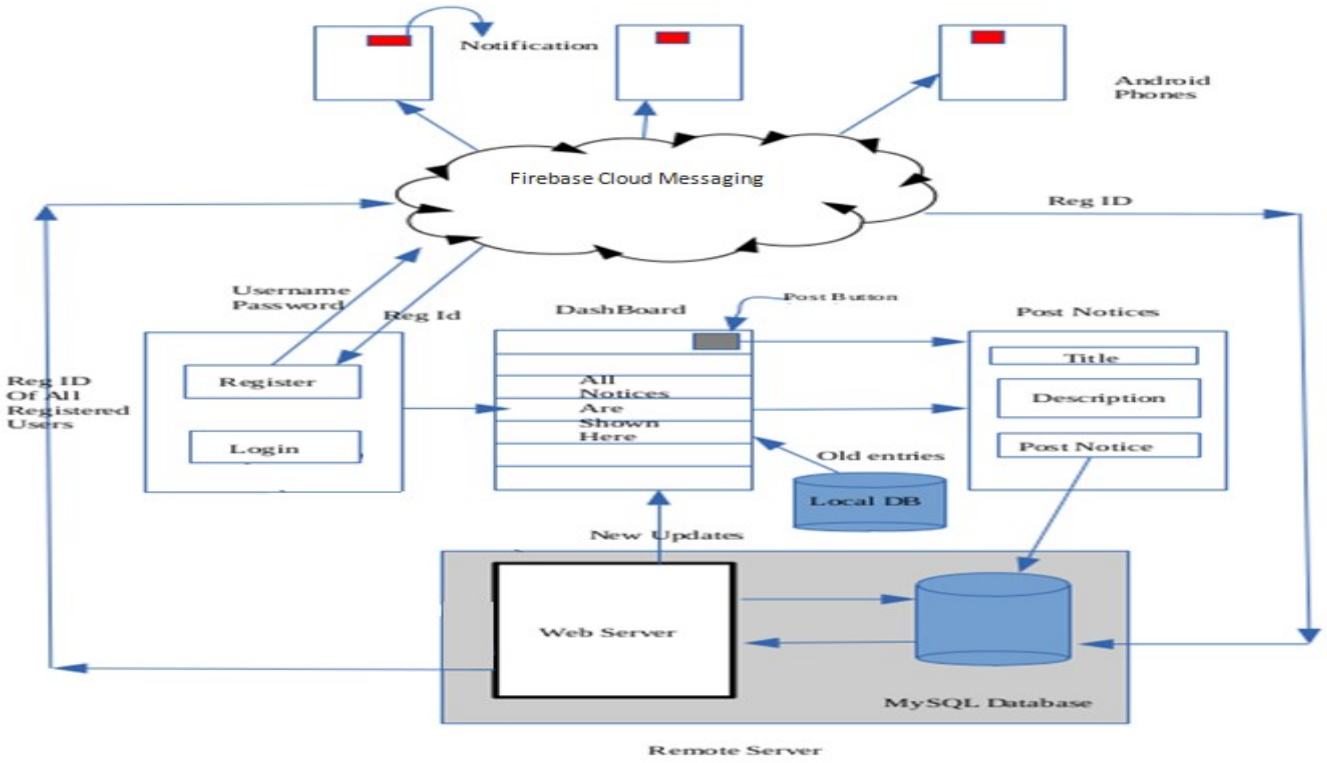
- Students
- Professors
- Admin Staff
- ISO/SGA Staff
- Library staff
- System Administrator

The table below describes the users:

Use	**Role**	**Est Usage**
System Administrator	Administer site specific parameters	Regular
Students	Subscribe for different category for receiving and broadcasting notices.	Regular
Professor	Broadcast course or exam related notices.	Regular

ISO/SGA Staff	Can post cultural events happening in university and ISO related announcements	Regular
Library Staff	Can post library related announcements	Regular

5.2 System Architecture Diagram:



5.2.1 What is Firebase?

Firebase is a cloud services provider. It provides a mobile platform that helps you quickly develop high-quality apps.

Firebase Cloud Messaging

- Firebase Cloud Messaging (FCM) is the new version of Google Cloud Messaging (FCM) under the Firebase brand. It inherits FCM's core infrastructure to make sure we continue to deliver messages reliably on Android, iOS and Chrome.
- FCM is a cross-platform messaging solution that lets you reliably deliver messages and notifications at no cost.

Firebase Auth

Firebase Auth is a service that can authenticate users using only client-side code. It supports social login providers Facebook, GitHub, Twitter and Google. Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase. Firebase automatically stores your users' credentials securely (using bcrypt)

5.3 Detail Design

The detailed design of this application is as follow:

1. Registering a User:

The first step in this application is to get the users registered to both FCM Server and to Remote Web Server. For this, user will provide all the necessary details and press the register button. The request will first go to Google Firebase Server. FCM Server will provide the registration id for that device. After that, all the information along with registration id is stored on Web Server and the user gets registered.

2. User Login:

After registering, the user is allowed to log in. Username and password after validating at client side, is sent to server side to authentication. After authentication response is sent by the server to client, and then user gets logged in.

3. Viewing the Notices:

At the first time, when you are using this application for the first time, it will fetch all the notices from server. In all the other case, all previous notices are fetched from application's own database stored inside client mobile. It then checks for new notices from the server. If there are new notices on the server, it will fetch all those notices.

4. Searching a Notice:

The user is able to search the notice in listview depending on the title of the notice. It helps user to get the desired notice instantly.

5. Deleting a Notice:

If the user does not want some notices, he/she can delete it from their phone. There will be no effect on server entry.

6. Posting a Notice:

If a user is an admin, he is able to post the notice. In order to post the notices, he has three option. One option is that, he can post a simple text notice. Another option allows him to send some attachment image with the notice. In this, he has two options. Either he can pick the image from the gallery or he can click a picture on the spot by using camera. After that, press the post button to post the notice.

7. Notification Buzz:

As soon as the admin post a notice, the script will run with which request is made by FCM Server to WebServer for all the registered IDs. After getting all the registered IDs, notification is sent to all the users registered with this application. Notification has a tune and vibration that runs whenever there is a notification received by the user from FCM Server.

8. Reset Password:

This application also has the facility to reset the password. If one user has forgot his password, he/she can rest the password by giving his username or email address. The user will be given a page in which he can set his new password. Forgotten password will be updated with the new one on the server.

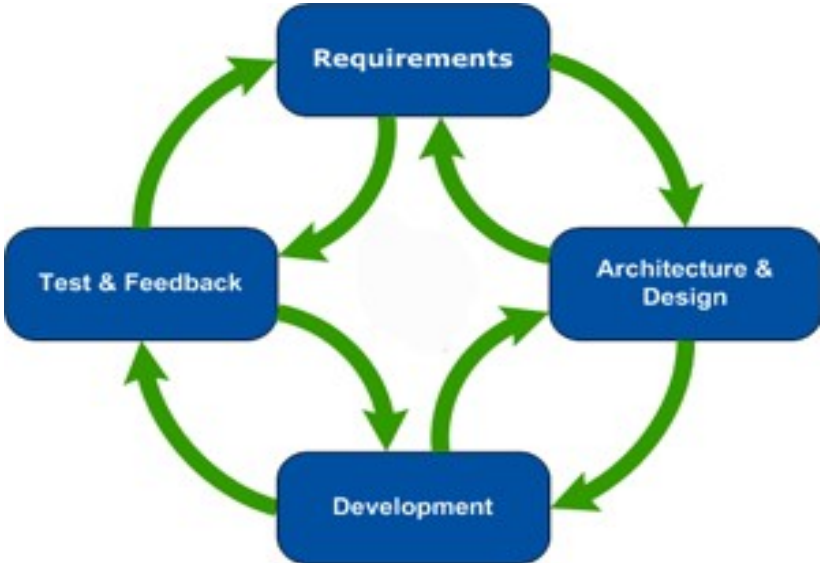
5.3 Methodology

The methodology used in project is Agile Software Development. Agile Software Development methodology is used especially for software development, that is characterized by the division of tasks into short phases of work and frequent reassessment and adaptation of plans. It is for a project that needs extreme agility in requirements. The key features of agile are its short termed delivery cycles (sprints), agile requirements, dynamic team culture, less restrictive project control and emphasis on real-time communication. Agile software development is a group of software development methods based on iterative and incremental development, in which requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. It promotes adaptive planning, evolutionary development and delivery, a time-boxed iterative approach, and encourages rapid and flexible response to change. It is a conceptual framework that promotes foreseen tight iterations throughout the development cycle.

The Manifesto of Agile Software Development context are:

- Individuals and interactions In agile development, self-organization and motivation are important,as are interactions like co-location and pair programming.

- Working software will be more useful and welcome than just presenting documents to clients in meetings.
- Customer collaboration Requirements cannot be fully collected at the beginning of the software development cycle, therefore continuous customer or stakeholder involvement is very important.
- Responding to change agile development is focused on quick responses to change and continuous development.



6 Database Design

6.0 Database Design

A relational database design was used to design the database. A relational database management system (RDBMS) is an excellent tool for organizing large amount of data and defining the relationship between the datasets in a consistent and understandable way. A RDBMS provides a structure which is flexible enough to accommodate almost any kind of data Relationships between the tables were defined by creating special columns (keys), which contain the same set of values in each table. The tables can be joined in different combinations to extract the needed data. A RDBMS also offered flexibility that enabled redesign and regeneration of reports from the database without need to re-enter the data. Data dictionaries were used to provide definitions of the data used; these included the final data structures for the various tables and their corresponding data fields, description and sizes. The android application interfaces were developed using Java and XML with support of structured query language (SQL) and MYSQL. The database used at the Back-end is MySQL database.

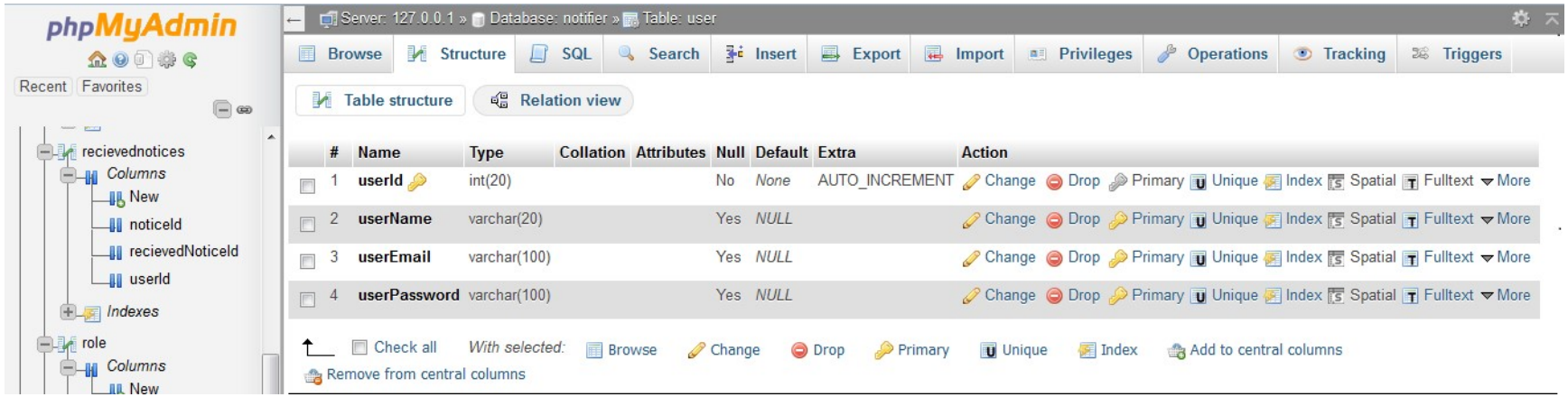
SQL is a language used to create, manipulate, examine and manage relational databases. SQL was standardized in 1992 so that a program could communicate with most database systems without having to change the SQL commands. Unfortunately one must connect to the database before sending SQL commands and each database vendor has a different interface as well as different extensions of SQL.

6.1 Data Dictionary

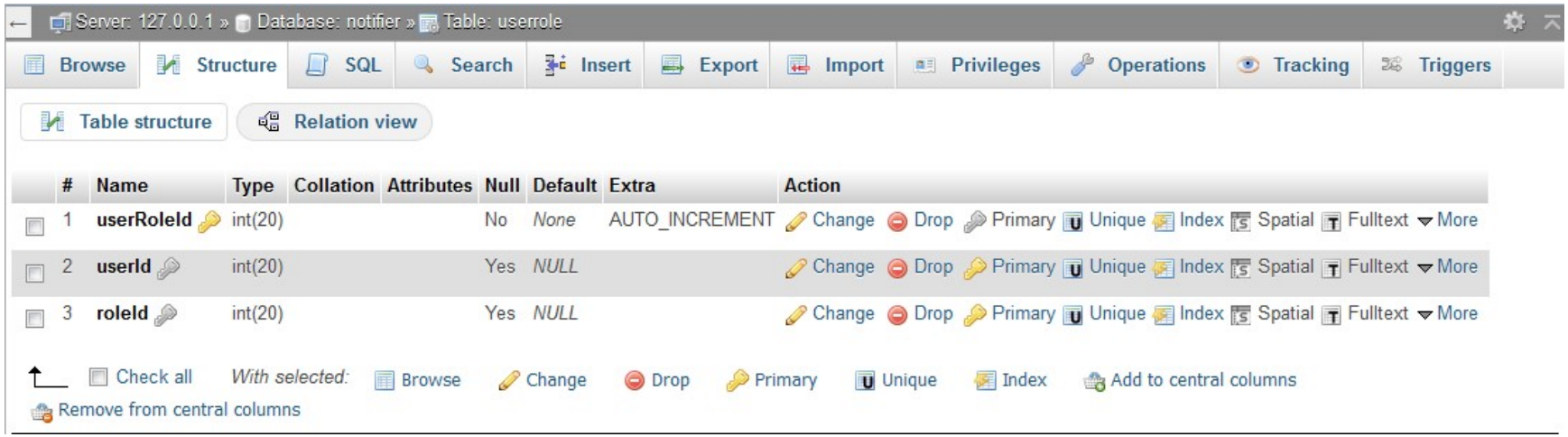
This section contains different relational tables, entities, attributes and data types

Database design used at the backend server has following tables:

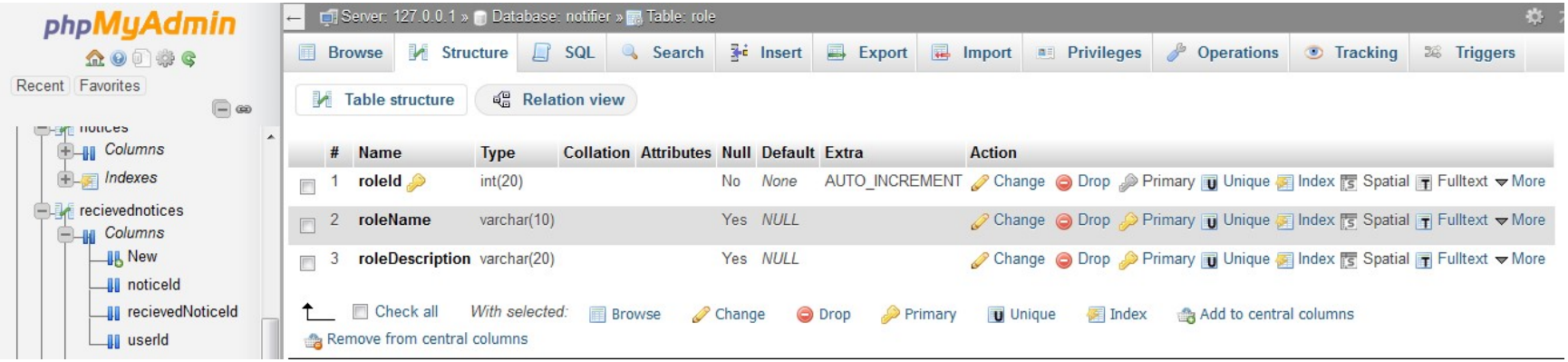
6.1.1 Table User: Details about User



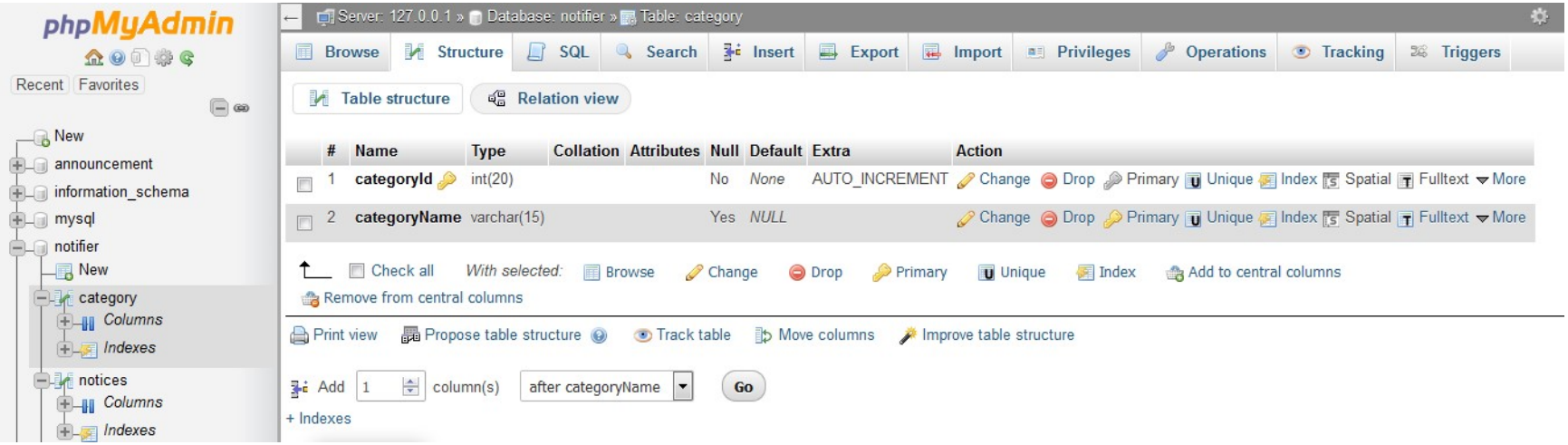
6.1.2 Table UserRole: Details about User Role



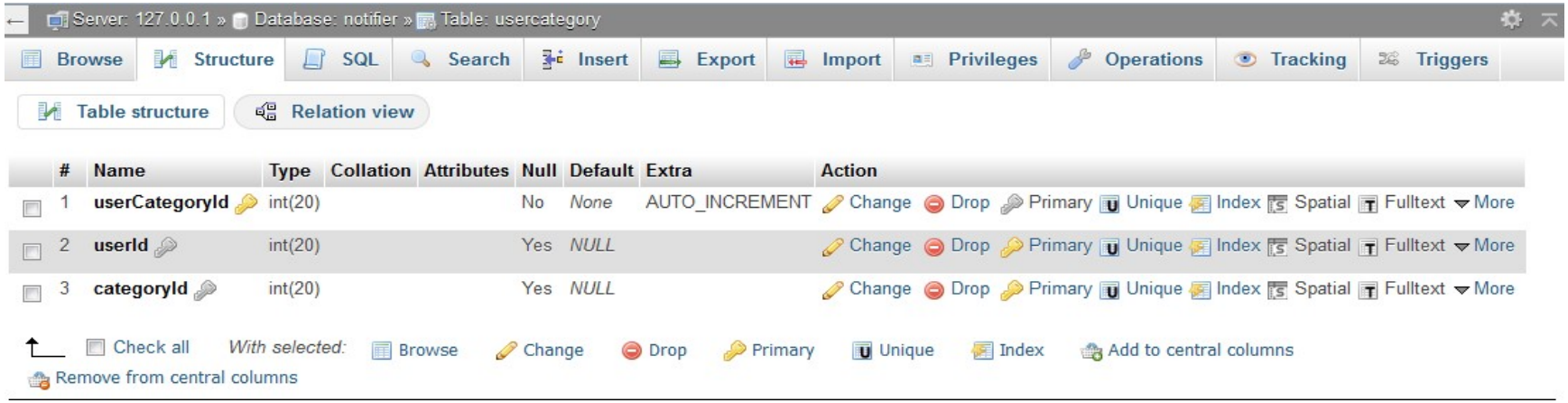
6.1.3 Table Role: Details about Role



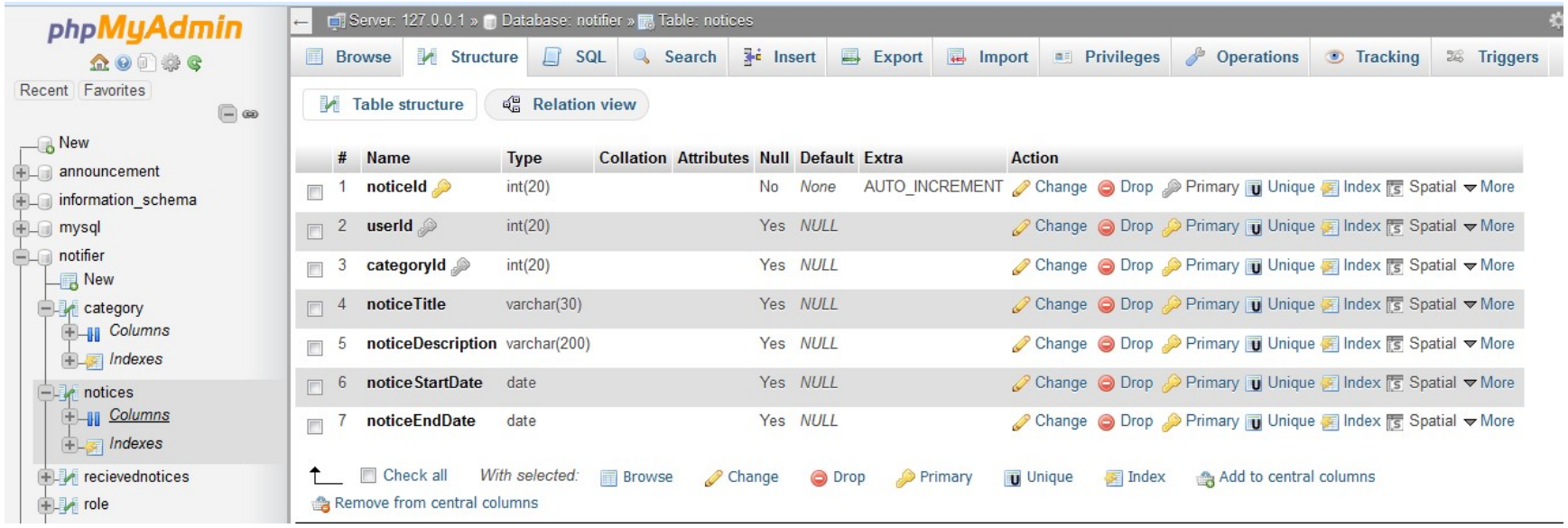
6.1.4 Table Category: Details about notice category



6.1.5 Table UserCategory: Details about category subscribed by user



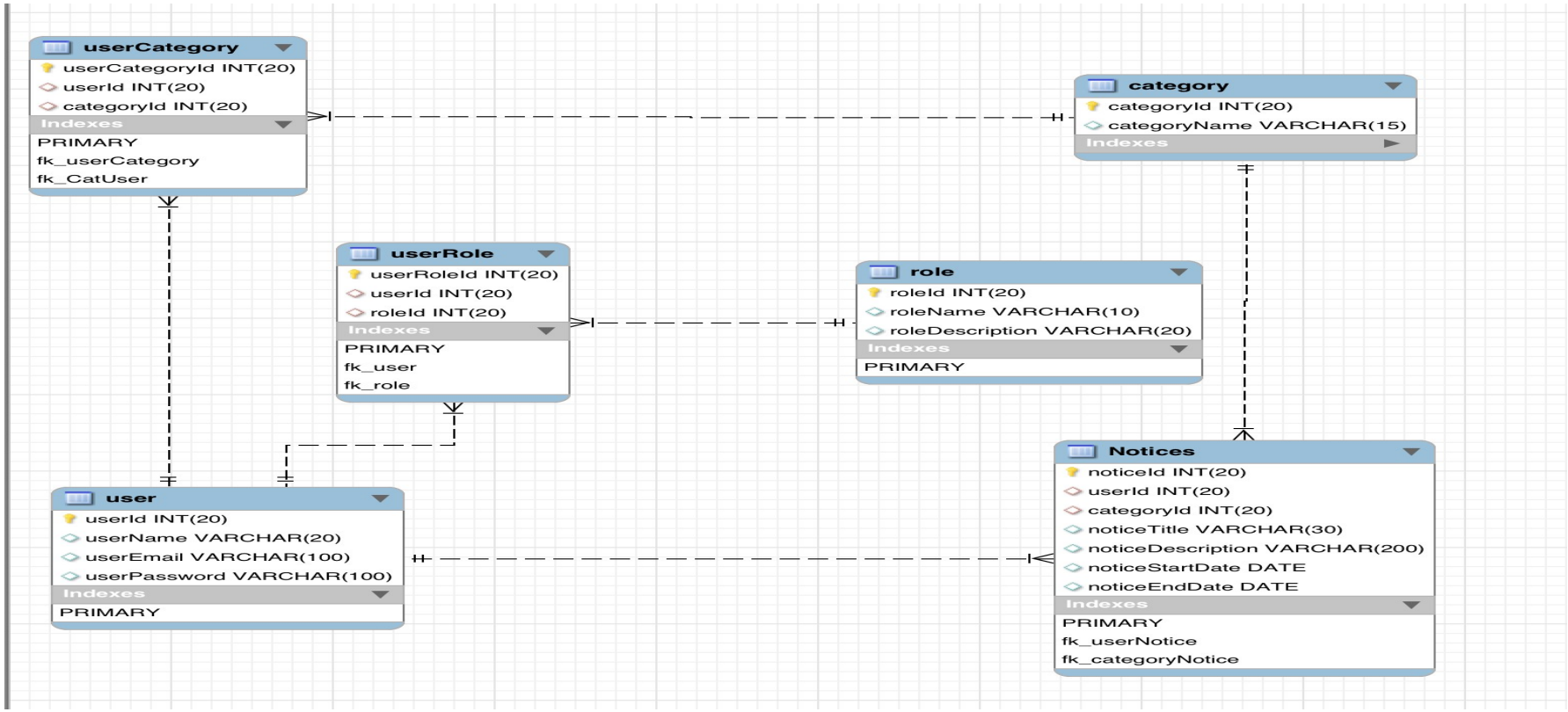
6.1.6 Table Notices: Details about Notices



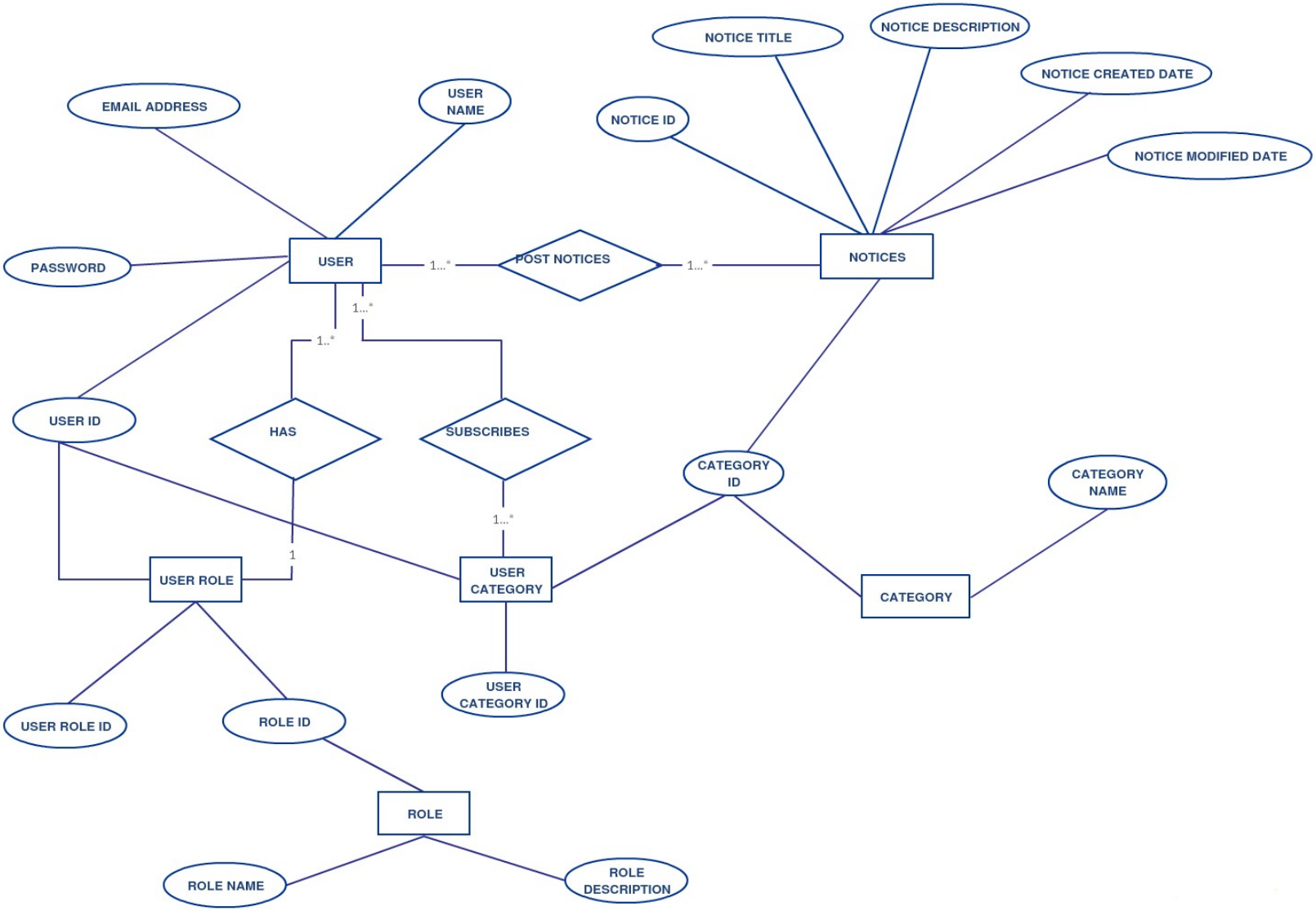
6.2 Entity Relationship Diagrams

Entity Relationship diagrams is a specialized graphics that illustrate the interrelationship between entities in a database. It helps to design database in efficient way.

6.2.1 ER-Diagram 1

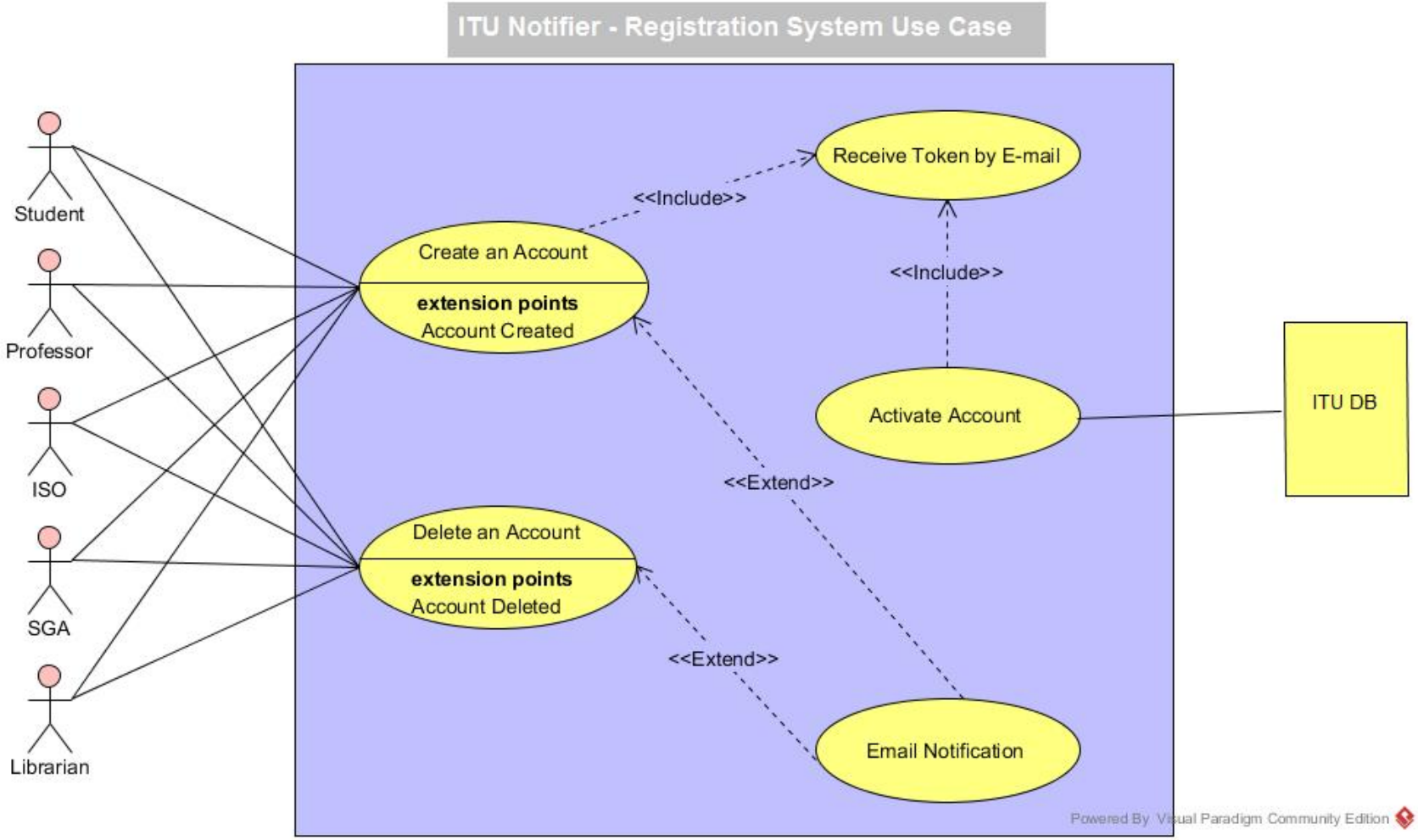


6.2.2 ER-Diagram 2



7 UML Diagrams

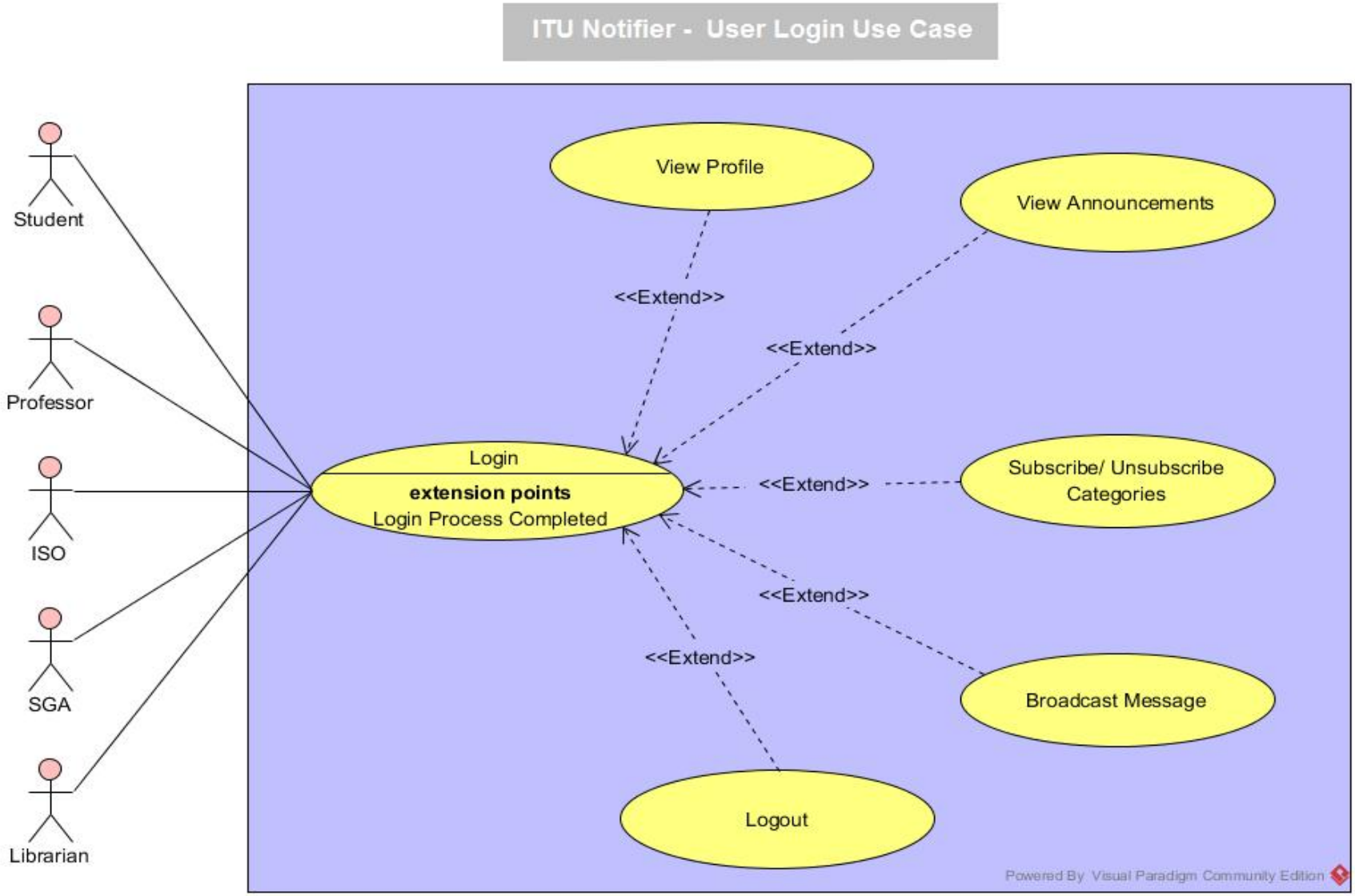
7.1 Use Case Diagram – Registration



7.1.1 Use Case Description - Registration

Name	User registers ITU email id with *ITU Notifier* System	
Use Case Story	A user registering for the first time on the *ITU Notifier* application on a mobile phone. The user inputs first name, last name username, password, ITU email id, and mobile number. The user receives a token from the system by email notification from system. The user enters token in the mobile app. The user account is validated and account details are stored in the ITU database The user receives an email confirmation from the system. The user tries to delete the account. The user receives an email confirmation from the system	
Actor(s)	Student, Professor, ISO, SGA, Librarian	
Pre-Condition	User has *ITU Notifier* app installed in mobile phone and valid ITU email id	
Post-Condition	User account is registered to system and can receive notices from *ITU Notifier*	
Associated Package	Email	
Updated/Created	SS (8/14/2016)	
Basic Flow:		
ID	**User Action**	**System Response**
1	User input details for new account registration	System sends a token to user's email.
2	User enters token in *ITU Notifier* app	System validates account and sends user confirmation email. ITU database stores newly registered account records.
3	User deletes account from app	System sends email notification to user. ITU database deletes user records.

7.2 Use Case Diagram – Login



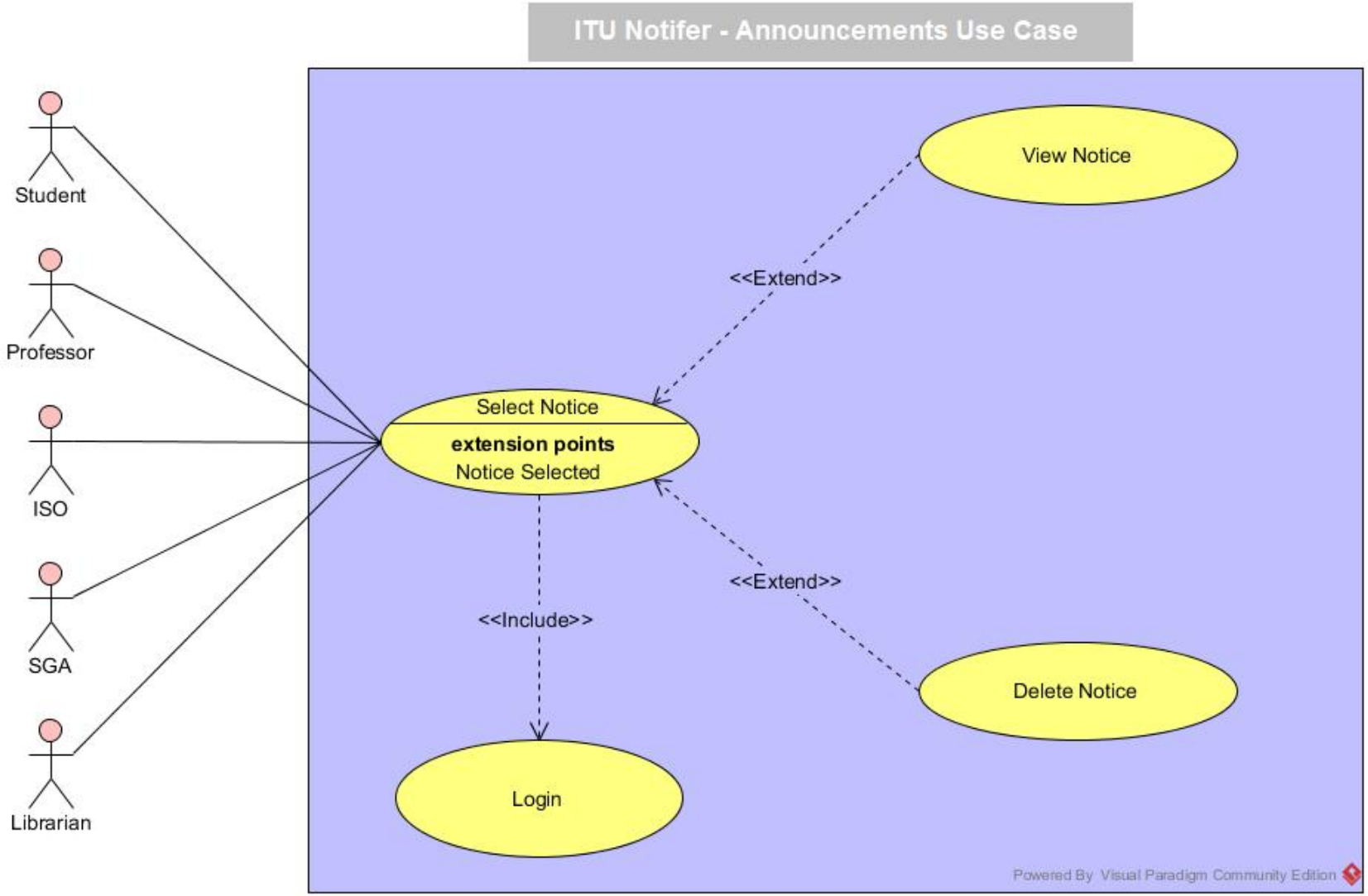
7.2.1 Use Case Description - Login

Use Case Name	*ITU Notifier* – User Login
Use Case Story	A user has logged to the *ITU Notifier* application on a mobile phone. The user inputsITU email id and password and presses login button. After successful login, the user can "view announcements", "subscribe/unsubscribe categories" and "broadcast message" options
Actor(s)	Student, Professor, ISO, SGA, Librarian
Pre-Condition	User has a registeredaccount in *ITU Notifier* app
Post-Condition	User account is logged in to system and can perform all functionalities on *ITU Notifier*app.
Associated Package	Email
Updated/Created	SS (8/14/2016)

Basic Flow:

ID	User Action	System Response
1	User logs in to application system.	System verifies the account credentials in ITU database and logs the user into the app.
2	User can "view announcements"	System generates the notice list and displays it to the user
3	User can "subscribe/unsubscribe categories"	System displays the category list and lets user select/deselect categories
4	User can "broadcast message"	System displays "post message" window
5	User can "View Profile"	System displays the "User Profile"
6	User can "Log Out"	System allow the user to log out from ITU notifier app

7.3 Use Case Diagram – Announcement

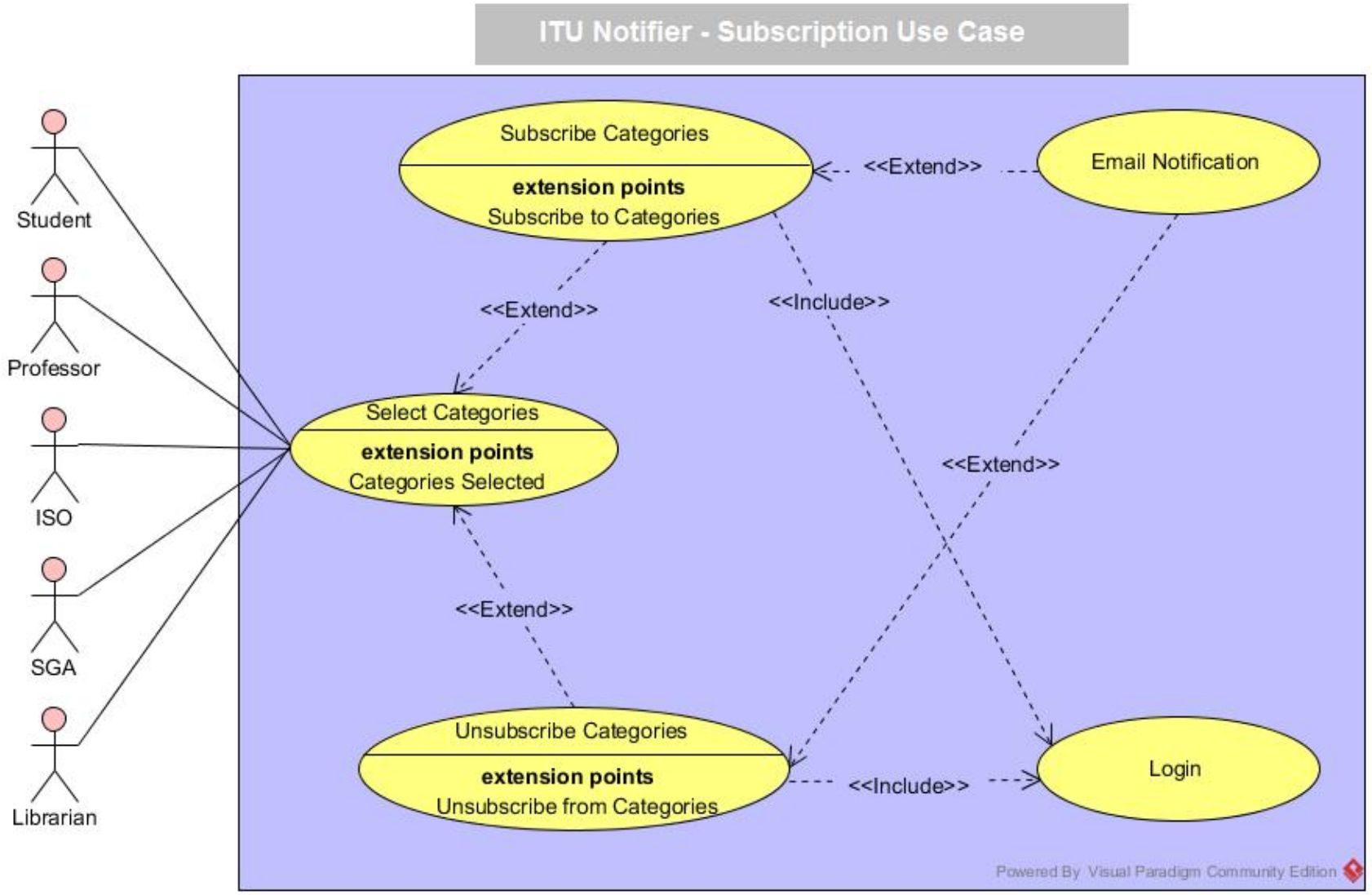


7.3.1 Use Case Description - Announcement

Use Case Name	*ITU Notifier* – Announcement		
Use Case Story	A user is viewing the "announcement" page of the *ITU Notifier* application on a mobile phone. The user can "select notice" and "search notice"		
Actor(s)	Student, Professor, ISO, SGA, Librarian		
Pre-Condition	User has already logged in to *ITU Notifier* app		
Post-Condition	User is able to search and select notice. The user is able to view and delete notices.		
Associated Package	Email		
Priority		**Rank**	
Updated/Created	SS (8/14/2016)		
Basic Flow:			

ID	**User Action**	**System Response**
1	User selects notice	System displays a checked checkbox against notice. allow the user to view all notices , delete any notice and search any specific notice
2	User clicks the notice to read	System displays notice in pre-formatted window
3	User deletes the notice	System deletes notice locally in the mobile phone app.
4	User searches notice using keywords of choice	System searches notice archive using keywords provided and displays results

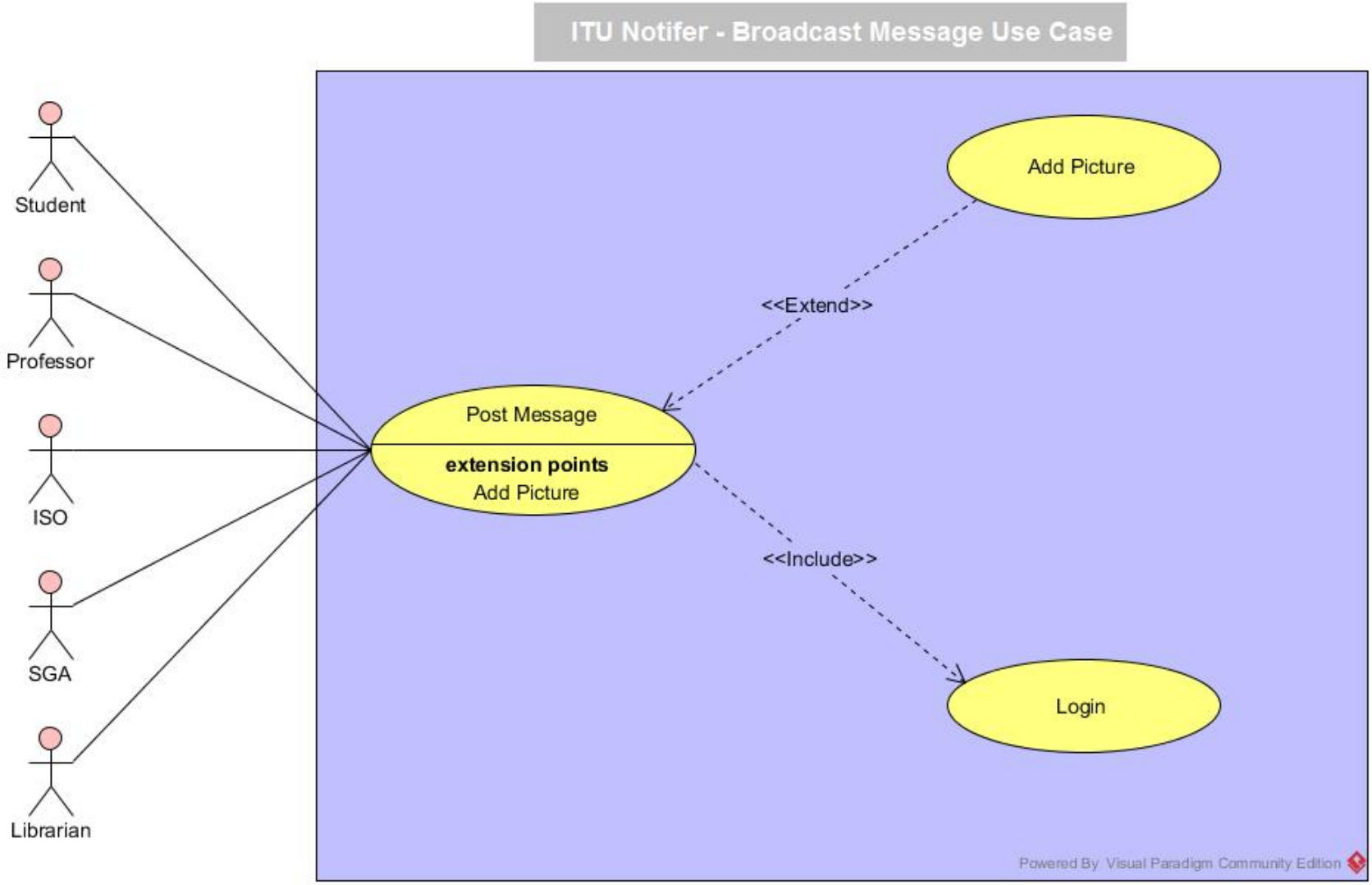
7.4 Use Case Diagram – Subscription



7.4.1 Use Case Description - Subscription

Use Case Name	*ITU Notifier* – Subscription		
Use Case Story	A user has logged into the system and wants to subscribe to other categories of interest. The user selectsfew categories for subscription. The user also unsubscribes from few categories. The user is notified of all changes by email made to subscription list.		
Actor(s)	Student, Professor, ISO, SGA, Librarian		
Pre-Condition	User haslogged into the app		
Post-Condition	User made changes to selected categories in subscription list		
Associated Package	Email		
Priority		**Rank**	
Updated/Created	SS (8/14/2016)		
Basic Flow:			
ID	**User Action**	**System Response**	
1	User selectscategories from announcement	System shows the selected categories in the subscription list.	
2	User subscribes or unsubscribes categoriesdisplayed the list	System sends an email notification to the user as confirmation about their action.	

7.5 Use Case Diagram – Broadcast



7.5.1 Use Case Description - Broadcast

Use Case Name	*ITU Notifier* – Broadcast		
Use Case Story	A user is already logged into the app and trying to post a notice.The user posts the notice with “title” and “description”. The user adds a picture with this post.		
Actor(s)	Student, Professor, ISO, SGA, Librarian		
Pre-Condition	User has already logged into the app		
Post-Condition	User posts an announcement		
Associated Package	Email		
Priority		**Rank**	
Updated/Created	SS (8/14/2016)		
Basic Flow:			

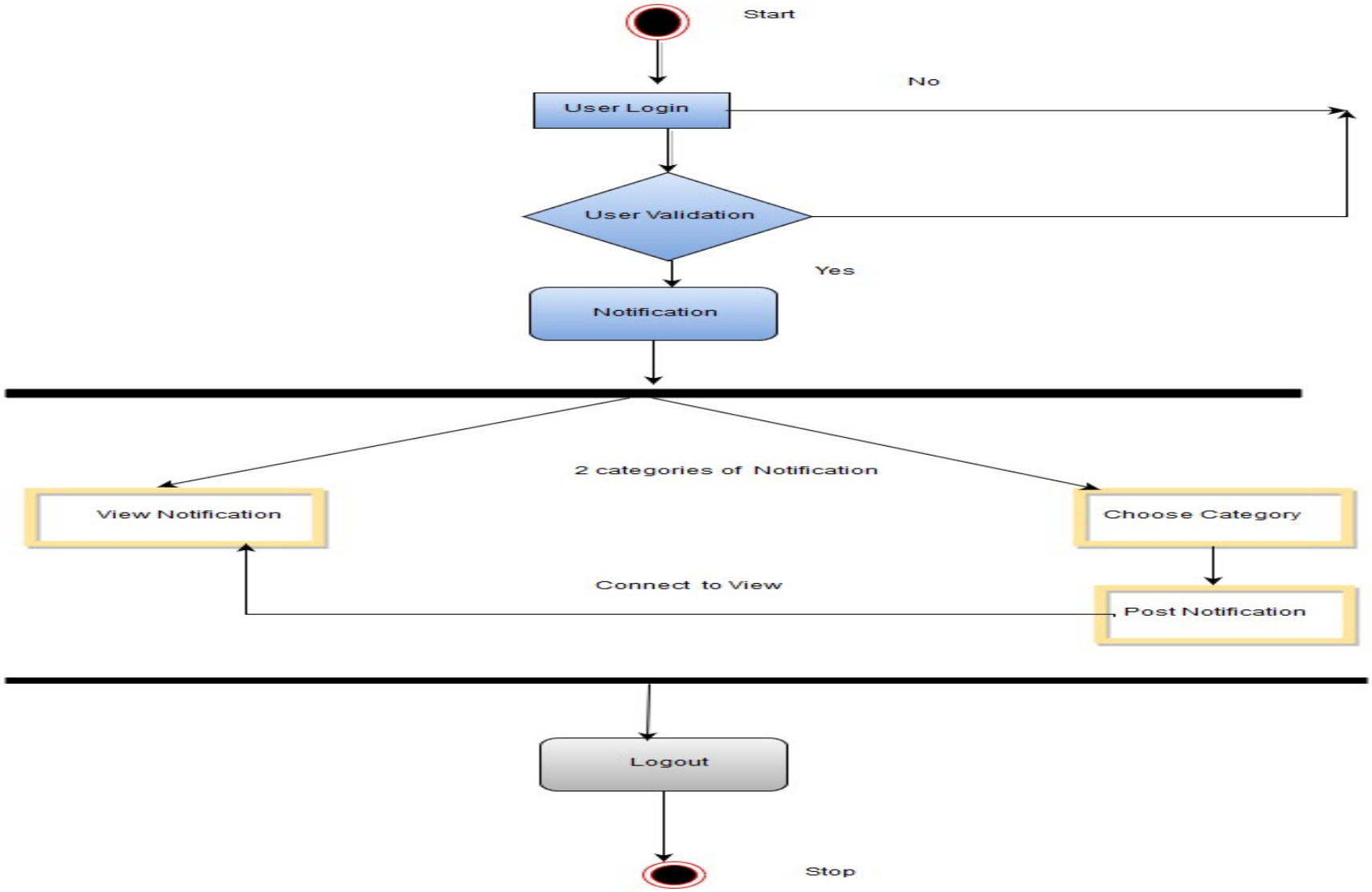
ID	**User Action**	**System Response**
1	User posts a message through the application system	System accepts the posts and broadcast the post for every stakeholders
2	User attaches pictures to the post	System accepts and saves picture with the post and enables it to be displayed through notifications.

7.6 Activity Diagram

Activity diagram are graphical illustrations or logical workflow of stepwise activities of ITU Notifies along with action which support for options and iteration .In UML diagram activity diagram are made from UML diagram different shapes and connected through arrows. Activity diagram also shows about control flow in these diagram

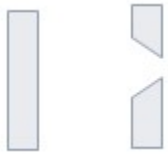
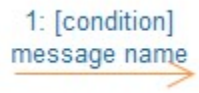
	In order to operate major task of operation in project is describe through it
	In UML diagram decision activity support condition .It connect previous activity too. Diamond shape recommend decision about yes and no in flow
	Initial state show initial stage of activity workflow .It represent “start” stage in flow diagram
	This sign also used to show stop sign in these activity
	This arrow sign shows connectivity in each steps

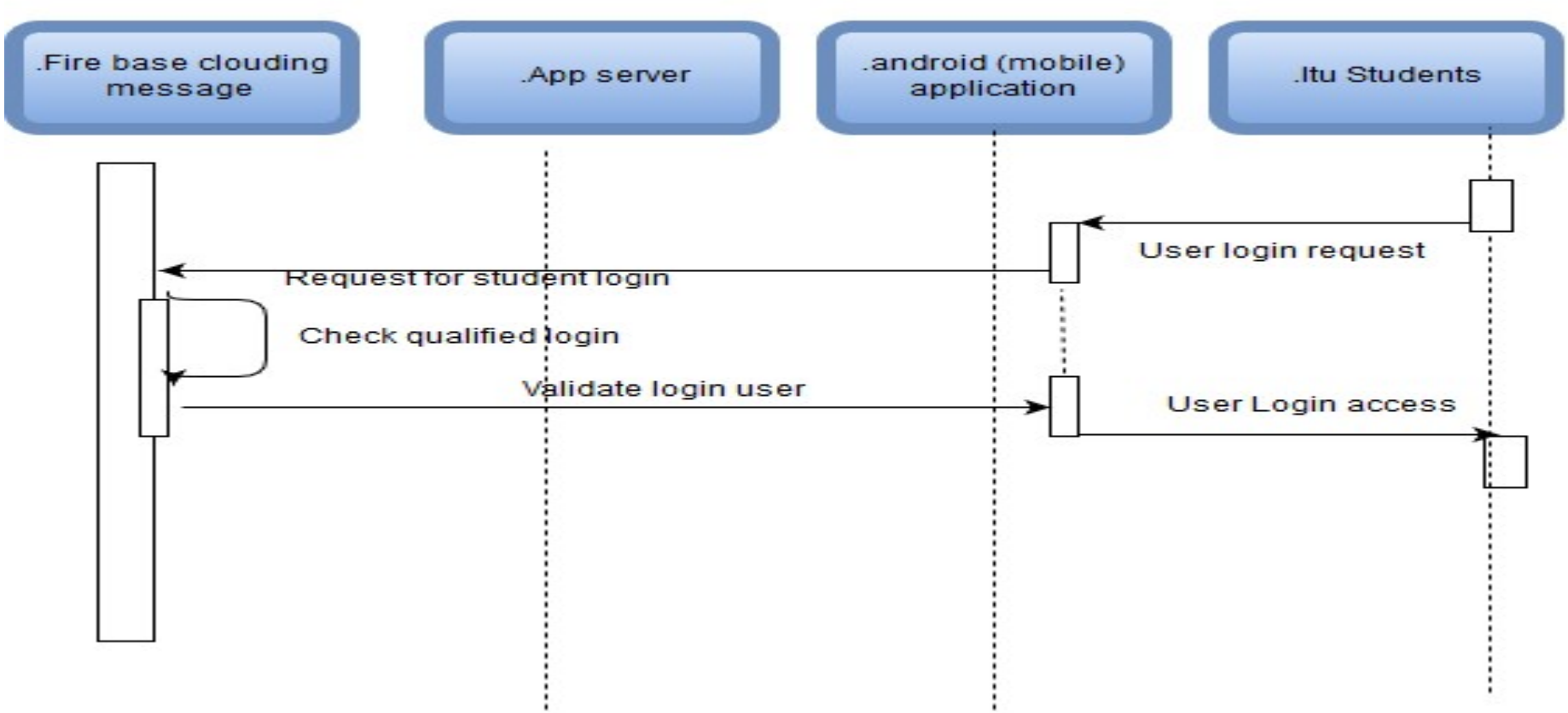
7.6.1 Login Activity Diagram



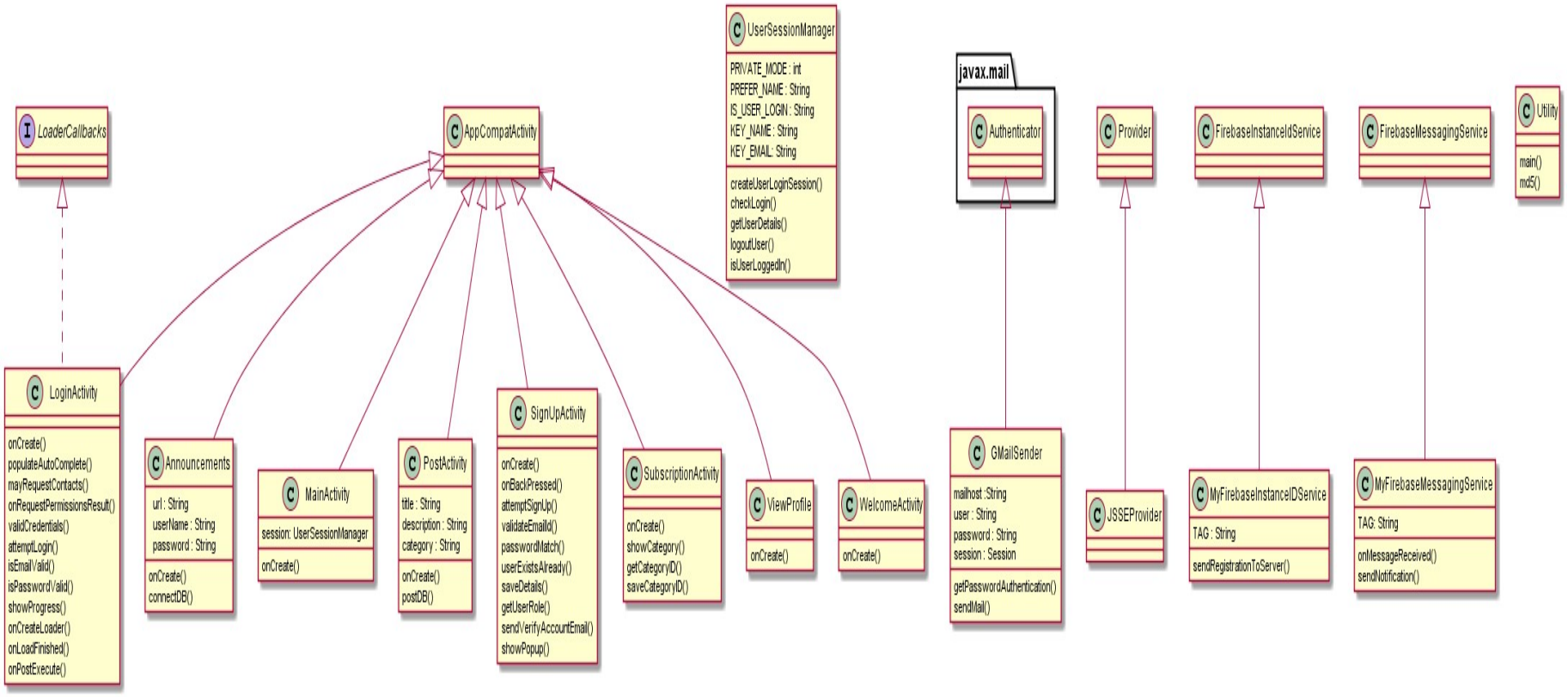
7.7 Sequence Diagram

A sequence diagram is outline that describe the time ordering of messages

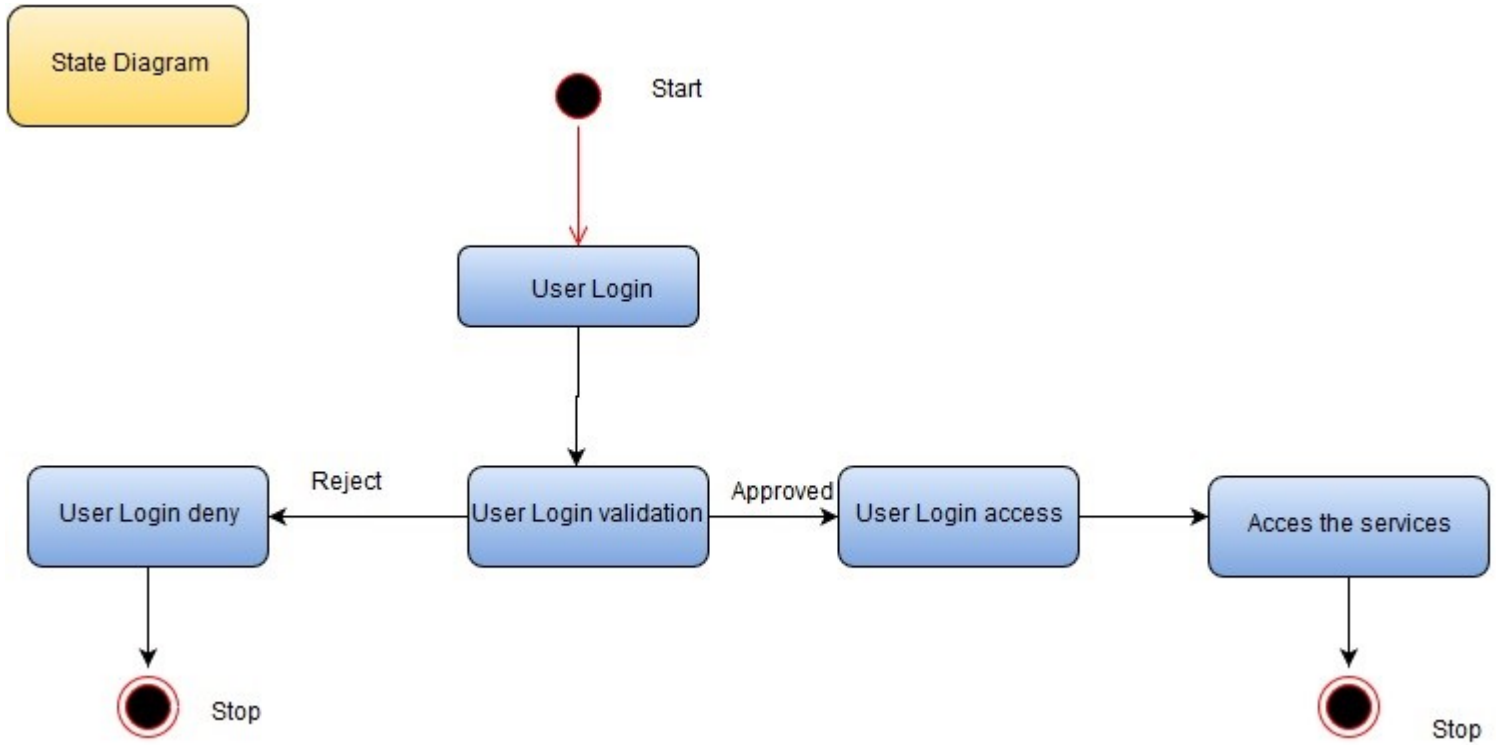
	Activate and activate breaks
	Object are model element which determine instance of class
	Message sent and return
	Lifeline which specify respectively instance in each interaction


.Fire base clouding message
.App server
.android (mobile) application
.Itu Students
User login request
Request for student login
Check qualified login
Validate login user
User Login access

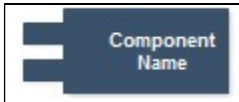
7.8 Class Diagram



7.9 State Diagram

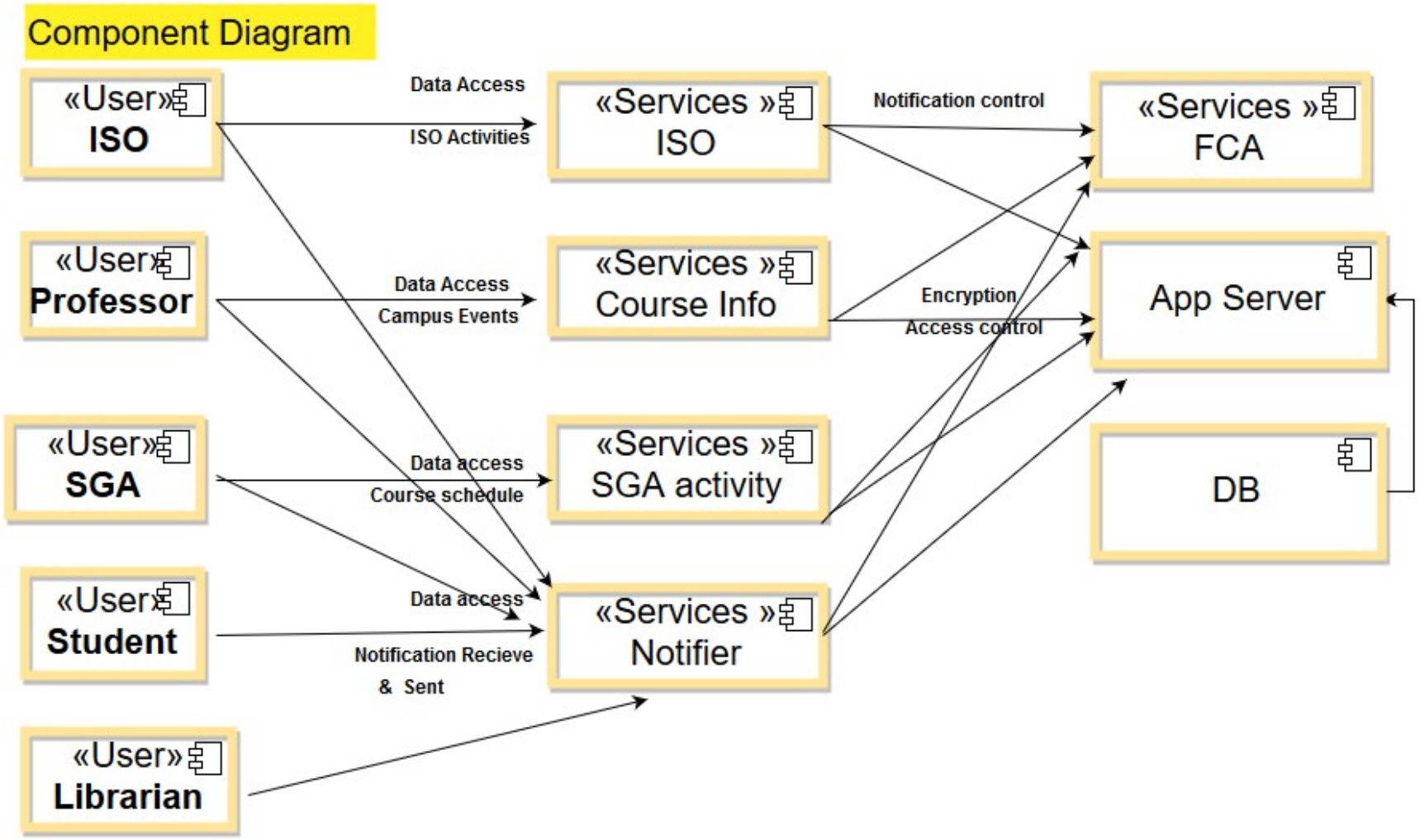


7.10 Component Diagram

	Different component are described with sign .It shows modular part of a system. It also specify behavior in needed interfaces
	Dependency shows relationship of one element of user or another element (activities) and supplier

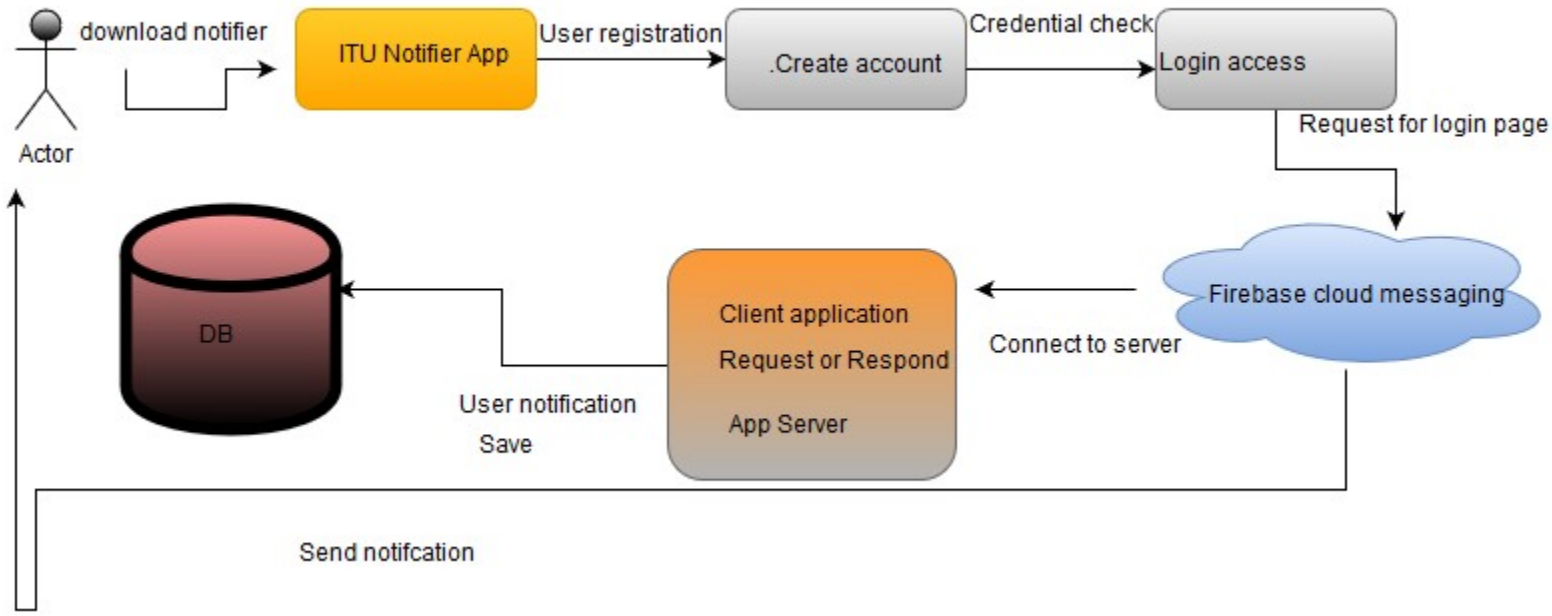
Abbreviation used in diagrams in diagram are ISO –International student officer, SGA- Student Government Association FCA-Firebase clouding Application/Messaging and DB is Database

Component diagram also shows dependency of user for different notification while interacting different departments .It also show backend activity DB connect to App server


Component Diagram
«User»
ISO
«User»
Professor
«User»
SGA
«User»
Student
«User»
Librarian
Data Access
ISO Activities
Data Access
Campus Events
Data access
Course schedule
Data access
Notification Recieve
& Sent
«Services »
ISO
«Services »
Course Info
«Services »
SGA activity
«Services »
Notifier
Notification control
Encryption
Access control
«Services »
FCA
App Server
DB

7.11 Collaboration Diagram

These diagram shows how object are model element which represent to class through different stage.The Collobration diagram shows FCM plays essential role in sending message to actor and responding for login page request



8. Risk Management

8.1 Purpose Of Risk Management Plan

The purpose of risk management is to identify potential problems before they occur so that risk-handling activities may be planned and invoked as needed across the life of the product or project to mitigate adverse impacts on achieving objectives.

Risk management is a continuous, forward-looking process that is an important part of business and technical management processes. Risk management should address issues that could endanger achievement of critical objectives. A continuous risk management approach is applied to effectively anticipate and mitigate the risks that have critical impact on the project.

Effective risk management includes early and aggressive risk identification through the collaboration and involvement of relevant stakeholders. Strong leadership across all relevant stakeholders is needed to establish an environment for the free and open disclosure and discussion of risk.

Although technical issues are a primary concern both early on and throughout all project phases, risk management must consider both internal and external sources for cost, schedule, and technical risk. Early and aggressive detection of risk is important because it is typically easier, less costly, and less disruptive to make changes and correct work efforts during the earlier, rather than the later, phases of the project.

8.2 Risk Management Procedure

Risk management can be divided into three parts: defining a risk management strategy; identifying and analyzing risks; and handling identified risks, including the implementation of risk mitigation plans when needed.

8.2.1 Risk Identification

Risk management requires you to identify potential risks; risk being anything that can possibly harm or have a negative impact on the project. Risk managers generally approach the search for potential risk from two distinct angles: source analysis and problem analysis. Source analysis seeks to look at the potential sources of risk whereas problem analysis looks at specific individual problems that could arise

8.2.2 Risk Analysis

Once risks have been identified, the next logical step in risk management is assessment. Risk assessment, as mentioned earlier, measures the probability of an identified risk actually taking place, as well as the amount of loss that would be suffered were the risk to actually occur. Loss and probability are usually placed in a prioritized list, with those risks that are most probable and that stand to generate the most loss given the most attention. In reality, a lot of guess work goes into this phase of risk management as at times it is almost impossible to evaluate and know the true likelihood as to whether a potential risk will occur or not.

The Risk Impact/Probability Chart provides a useful framework that helps decide which risks need attention. The Risk Impact/Probability Chart is based on the principle that a risk has two primary dimensions:

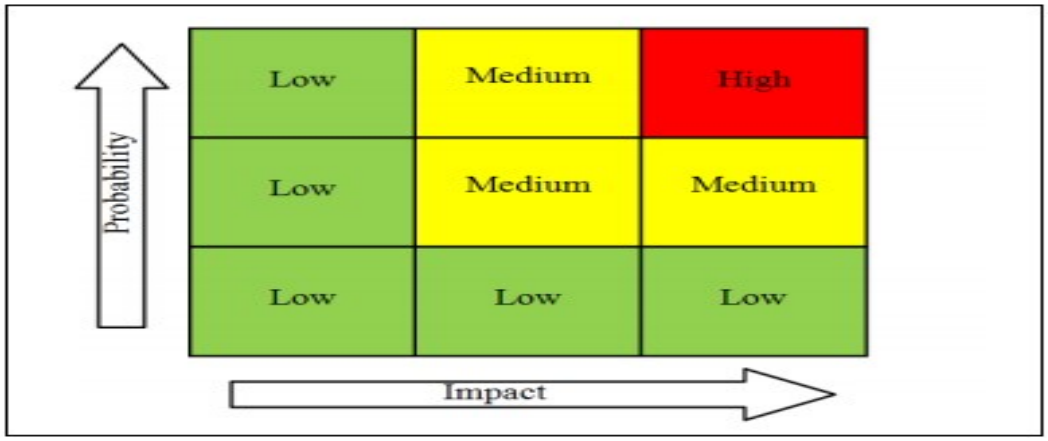
8.2.2.1 Probability

A risk is an event that "may" occur. The probability of it occurring can range anywhere from just above 0 percent to just below 100 percent. (Note: It can't be exactly 100 percent, because then it would be a certainty, not a risk. And it can't be exactly 0 percent, or it wouldn't be a risk.)

8.2.2.2 Impact

A risk, by its very nature, always has a negative impact. However, the size of the impact varies in terms of cost and impact on health, human life, or some other critical factor.

The chart allows you to rate potential risks on these two dimensions. The probability that a risk will occur is represented on one axis of the chart – and the impact of the risk, if it occurs, on the other. You use these two measures to plot the risk on the chart. This gives you a quick, clear view of the priority that you need to give to each. You can then decide what resources you will allocate to managing that particular risk. The basic form of the Risk Impact/Probability Chart is shown in figure below.



8.3 Risk Response Planning

Potential risks Identified in project and the proposed measures:

1. Incompetent management (lack the necessary qualifications and skills, poor organization)
   - diversification of company management
   - the involvement of external consultant

2. External risk factors (factors outside the direct control of the organization)
   - development of catastrophic scenarios, recovery plans
   - creating team for crisis management
   - data storage in a safe place

3. Information technology (disruption can be caused by internal factors (employees) or external factors (viruses))
   - authorized of documentation and requires access to system and data
   - limiting access to system and to data

4. Lack of marketing (inaccurate, delayed or unavailable information concerning prices, products, advertising, sales support)
   - the introduction of marketing research
   - evaluation of pricing strategies compared to competitive products and prices
   - evaluate the effectiveness of advertising and sales promotion

5. The organization is lagging behind in technological development (management does not have access to information associated with the current technological development)

- monitoring of competitors
- technical literature
- visits to seminars, conferences, business meetings, exhibitions
- regular summary information on technological development

6. High staff turnover (problems in the organization, for example: wages under industry-average, lack of investment in development of the employees, wrong systems for monitoring)

- encouraging vertical communication
- keeping active company culture

7. Failure safety (lack of knowledge regarding the laws for the protection and safety at work)

- obtain competent legal advice for business relating to the company for the protection and safety
- periodic verification of the legal advice they are applied to the procedures and security measures

8. Organization lacks quality strategy (incomplete or inaccurate information regarding the changes about competition, products and customer preferences)

- development of strategic plans for top management, which will include a vision of the organization
- periodic review directions and priorities set top management to ensure they are still valid.

8.4 Risk Monitoring and Controlling

The proposed measures are very good prevention as to avoid future potential risks. These measures can provides enhanced planning and better decision making, rather than crisis management.

Potential risk		Elimination of potential risk	Risks in matrix grid
1	Technical capability of the product does not meet the expectations and wishes to the customers, the product does not have assumption that will succeed in the market	The risks of the project are clearly defined in advance, before beginning Through market research	**Medium risk**
2	Complicated design of product for manufacturing, the expenses of the product development process exceeded the limits and budget forecast	Project managers must achieve balance among these resource constraints: time, funds, and required quality	**High risk**
3	The durability of the product development process is longer as it was planned and therefore the product could not enter to the market in a right time	Anticipating changes in technology and estimating the product life cycle	**Low risk**
4	The problems in manufacturing are caused very complicated and complexity of the product, increase the cost of producing the product	Set a limit on the amount of risk, that will be accepted	**Low risk**

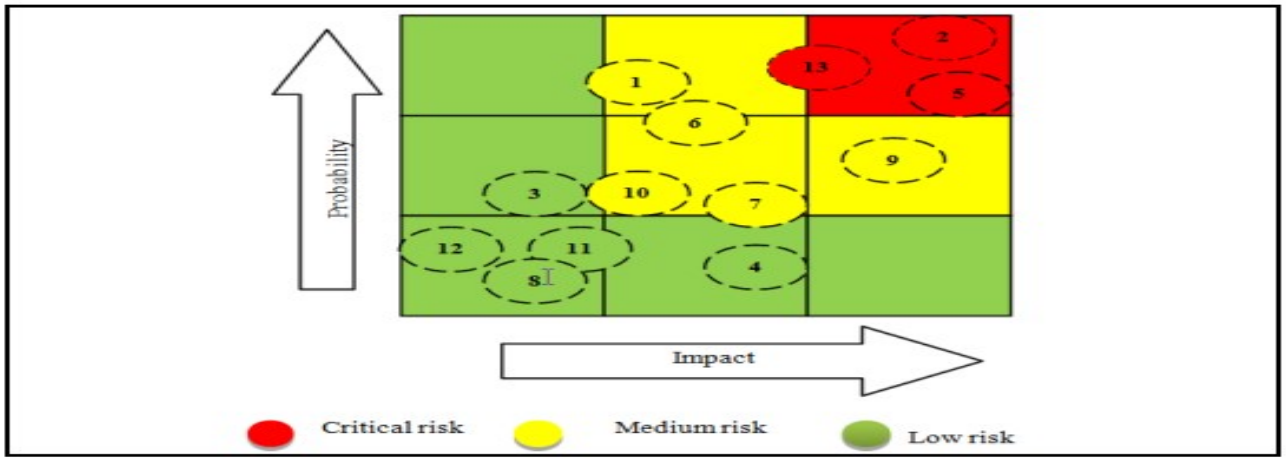
5	Critical people left the project	Improved communications through project Increased motivation through: financial rewards, gift vouchers, vouchers for birthday, staff training, career advancement, services paid by company: foreign visits, courses, training Change the project manager in project management	**High risk**
6	Critical resources needed for the project was not available at the right times	Developing plans and practices, that notify the most serious risks Planning and monitoring of risks by introducing standardized connection between risk management processes, enabling quick respond to changes in other people involved in the development and take into account changes the objectives of the project	**Medium risk**
7	The project missed critical milestones	Compliance methods at each stage in process of product development Improve the process more accurate by redistributing of means Development alternative methods - each partial goal can be achieved by several methods	**Medium risk**
8	Cost of the product exceeded the market expectations, exceeded budget	Anticipating and responding to expected competitor reaction. Periodic and detailed estimates throughout the project Inclusion of a	**Low risk**

		project manager who has experience with audits and inspections that are done before assigning tasks	
9	Introduction of new tools, technology, or processes during the project	Testing of new tools, technology or processes for a specified period Analysis of internal and external environment (market, customers, competitors, strengths and weaknesses of the company)	**Medium risk**
10	The competence of the product development team is not at this level as it was expected and there is a lack of key competence	Involvement of an experienced project manager to team	**Medium risk**
11	The project management team don´t follow the best practices and rules in project management	Review the decision to accept best alternative	**Low risk**
12	The subcontractors and supplies do not fulfill the schedule	Effective communication, regular exchange of information between subcontractors and supply	**Low risk**
13	The mistakes in design cause problems in manufacturing. There are misunderstandings between designers and manufacture. Redesign is expensive and takes additional time	Optimizing the cost of risk management and reducing losses due to incorrect decisions	**High risk**

Indication of the potential risks identified in the matrix grid

Risk #	Probability of Occurrence	Risk #	Impact of risk
1.	12	1.	5,5
2.	14,5	2.	14,5
3.	5,5	3.	4
4.	2	4.	9
5.	10,5	5.	14,8
6.	9,8	6.	7
7.	5	7.	9
8.	1	8.	3,5
9.	7	9.	12,5
10.	6	10.	6
11.	3,5	11.	4,8
12.	3,5	12.	2
13.	12	13.	10

In the figure below we can see identified and assessed risks based of probability and impact. These risks are situated in matrix grid.



9 Quality and Test Plan

9.1 Scope

The scope of testing is to test the operating characteristics of Android application that runs on Mobile devices .The test are organized by requirement category such as usability, functionality, security etc. The procedure for carrying out testing in terms of preparation of test cases, test environment set up, defects logging and reporting

9.2Test Plan Objective

The Software Test Plan (STP) is designed to prescribe the scope, approach, resources, and schedule of all testing activities. The plan will identify items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources and schedule required to complete testing. The purpose of the software test plan is such as:

- To achieve the correct code and ensure all Functional and Design
- Identify requirement for this clinic and the software that should be tested.
- List the recommended test requirements (high level).
- Requirements are implemented as specified in the documentation.
- Identify the required resources and provide an estimate of the test efforts.
- List the deliverable elements of the test activities.
- To provide a procedure for Unit and System Testing.
- To identify the test methods for Unit and System Testing.

9.3 Entry Criteria:

1) Development of the application is complete

2) All the modules of ITU Notifier should be unit tested.

3) Release of software to the test environment.

4) Test cases are reviewed

5) Dedicated test resource are allocated.

6)Test environment is set up

7) Test environment is set up

8) Approved test bed to carry out testing.

9.4 Exit Criteria:

1) All planned test cases are executed and 95% are passed and remaining 5% does not impact critical functionality.

2) Defect are fixed in the code and tracked to closure

3) All test results have been evaluated and accepted.

9.5 Test Process

- Identify the requirements to be tested. All test cases shall be derived using the current design specification.
- Identify particular test to use to test each module.
- Identify the expected results for each test.
- Perform the test.
- Document the test data, test cases used during the testing process.

9. 6 Testing Type

9.6.1 Unit Testing

Validation testing is one kind of testing which will test on whether the system is able to handle the wrong data that entered by the user or not. It also use to ensure that the data entered by the user is relevant and in the correct format or correct data types. For this testing, we will test whether the system can handle the data that entered by the user. For example, if the user is required to enter the numeric data but they are enter the string data, will the system able to track the data entered by user and prompt error message to inform the user to enter the correct one. This testing is very important because the wrong data entered may cause the system error or accidentally Insert the wrong data into the database and misleading the system.

Units to be tested:

User Registration

Dashboard

Students Login

Search Notices

Post Notices

Professor Login

Admin Login

9.6.2 Integration Testing

We carried out integration testing after different modules had been put together to make a complete system. Integration was aimed at ensuring that modules are compatible and they can be integrated to form a complete working system. For example we tested to ensure that when a user is logged in, he/she is linked to the appropriate page, and also could access the database.

9.6.3 User Acceptance Testing

User Acceptance Testing is a testing to determine whether the system that had been developed is satisfied and accepted by the user or not. It will carry out after all testing are done and will test by the user who will use the system. In this testing, the user also can decide whether the acceptance level to the system or perhaps the user can provide some feedback to the programmer so that the programmer can make enhancement on the system

1) Broadcast Notices by student Login

2) Broadcast Notices by Professor Login

3) Subscribe for different categories

4) Receive Notices for subscribed category Notices

5) Search for Notices

9.6.4 System Validation

Validation testing is one kind of testing which will test on whether the system is able to handle the wrong data that entered by the user or not. It also use to ensure that the data entered by the user is relevant and in the correct format or correct data types. For this testing,

we will test whether the system can handle the data that entered by the user. For example, if the user is required to enter the numeric data but they are enter the string data, will the system able to track the data entered by user and prompt error message to inform the user to enter the correct one. This testing is very important because the wrong data entered may cause the system error or accidentally insert the wrong data into the database and misleading the system.

9.6.5 Regression Testing:

This is an additional step, and is done prior to taking up system testing which is to test new functionality. Regression testing consists of running a set of standard tests to ensure that old functionality has not been broken by new functionality. Regression tests are also run if a new release is made after fixing a number of defects.

9.6.6 Performance Testing

Verify response time to access ITU Notifier App.

Verify response time to access system by Student, admin login.

Vision Document: "The system shall provide access to the fetch notices from Database with no more than a 10 second latency."

Supplementary Specification, "The system shall provide access to the view notice with no more than a 10 second latency."

9.6.7 Load Testing

- Verify system response when loaded with 200 logged on students.
- Verify system response when 50 simultaneous students' accesses and post notices.

9.6.8 Stress Testing

Write script which will post notices continuously for one week with an interval of 1hr.

9.7 System Validation

Validation testing is one kind of testing which will test on whether the system is able to handle the wrong data that entered by the user or not. It also use to ensure that data entered by the user is relevant and in the correct format or correct data types. For this testing, we will test whether the system can handle the data that entered by the user. For example, if the user is required to enter the numeric data but they are enter the string data, will the system able to track the data entered by user and prompt error message to inform the user to enter the correct one. This testing is very important because the wrong data entered may cause the system error or accidentally insert the wrong data into the database and misleading the system.

Pass-Fail Condition: It is expected that an application must pass all the tests in each test category to be successful.

Test Report:

For each report, the following information is provided.

1) Name of the application

2) Version number of the application.

3) Device used for testing

For each error reported, following information is provide.

1) Description of error

2) Frequency of occurrence of error: Systematic or Random or Once

3) Location of error in the application

4) Steps to reproduce the error

Test Metrics:

Following metrics will be captured and reported as part of test:

- Summary Report
- Test Design effort
- Test execution effort
- Number of Test Cases executed
- Number of Defects and their classification
- Test Coverage (Number of test cases executed/Number planned)

Deliverables:

Test Plan

Test case document

Test Results with pass/fail status of each test case and list of issues**.**

9.8 Test Plan

Significant feature/Scenarios to be tested:

S.no.	Subject	Description	Trace	Expected Result
1	Registration	Validate that user registration is successful in the following way: a) First name- lower case, upper case, Mixed cases b) Last name of user- lower case, upper case, Mixed case c)id of user- Lower case, upper case, Mixed cases, alphanumeric, 3 char, 18 char, 36 char d) password-Lower case, upper case, Mixed case, alphanumeric, 3 char, 18 char, 36 char	FR1	Registration should be successful for ITU notifies
2	Registration	Validate that registration is unsuccessful in the following scenarios: a) First name- blank, numerical value, special char like % etc,37 char b) Last name- blank, numeric value, special char like %, any space in the char,37 char c)id- duplicate(reusing same id), blank, starts with number or beginning with special char, 1char, 37 char, not using @ d) Password-duplicate email address, blank, begin	FR1	Registration should fail.

		with number, begin with special char, 2 char, 37 char		
3	Login	Validate that ITU user login is successful with right user name and right password. Also, validate that login is unsuccessful in the following scenarios: a) User name- blank, wrong, used any special character b) Password- blank, wrong ,or any uppercase c) either user name or password used are correct and incorrect respectively.	FR2	Login should be successful when user name and password are both accurate. Or It may be unsuccessful if information is incorrect

4	Send Notifies	Validate that user can send notifies through these app successful in the following way: a) User name of user -lower case, upper case ,Mixed cases b) Last name of user- lower case, upper case, Mixed case c)id of user- Lower case, upper case, Mixed cases, alphanumeric, 3 char, 18 char, 36 char d) password-Lower case, upper case, Mixed case, alphanumeric, 3 char, 18 char, 36 char e) send notification - blank, wrong	FR3	Successful login will support to send notifies
5	Receive Notification	Validate that user can received notifies through these app successful in the following way: a)Correct user and password b)Login shows notification date and time ascending way	FR4	Receive notification successful
6	Save notification	Validate that user can received notifies through these app successful in the following way: a)Correct user and password b)Login shows notification date and time ascending way c)allows to save any notification through these app	FR5	Save notification successful

7	Read all Notifies	Validate that user can read all notified through these app successful in the following way: a)Correct user and password b)Login will organized read and unread notifies	FR6	Sort read and unread notifies
8	Send attachment	Validate that user can send attachment through notifies successful in the following way: a)Correct user and password b)Login can send attachment up to 2.5k c)notifies support jpeg,png,gif,pdf,MS word and zip file	FR7	Send attachment successful through it
9	Quick visibility of notification	Validate that user can view notification through notifies successful in the following way: b)Correct user and password b)Login in these app will broadcast message display within second	FR8	Able to view notification successfully as it broadcast
10	Feature to organize	Validate that user can view notification and organized category through notifies successful in the following way: b)Correct user and password b)Login in these app feature for admin ,campus activity,ITU present, course ,ISO and others	FR9	Notifies feature support to organized notification

9.9 Test Case – Registration

Project Name	ITU - Notifier
Screen Name	Registration
Date of creation	8/15/2016
Date of review	8/20/2016

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority
ASE_IN_9.9.1.1	Verification of starting of the ITU-Notifier system	Application is installed on an android mobile	Open the app in the android mobile phone	The application is opened successfully	High
			Close the application	The application is closed successfully	High
ASE_IN_9.9.1.2	Verification of the user interface of the ITU-Notifier system	Application is opened	Verify UI of the ITU-Notifier system	The UI of the app is being displayed as follows: 1) The app name " ITU-Notifier" is being displayed at the top left side of the application. 2) It shows following two input fields : a)"E-mail" b)"Password" 3) It shows following two buttons: a)"Sign In" b)"Register"	Medium

ASE_IN_9.9.1.3	Verification of the functionality of the "Register" button	Application is opened	Click on the "Register" button	A new screen for account registration is displayed.	High
ASE_IN_9.9.1.4	Verfication of UI for Register Account Screen	The new UI for registration is being displayed	Verify UI of Registration Screen	1) Application Name " ITU-Notifier" is being displayed at the top left side of the application 2) A new UI for registration is displayed with following input fields and a "Register" Button at the bottom of screen : a) "First Name" b) "Last Name" c) "User Name" d) "Password" e) "E-mail ID" f) "Mobile No"	High
ASE_IN_9.9.1.5	Verification of functionality of "First name" input field	The new UI for registration is being displayed	Click on the "First Name" input field	The "First Name"" tab is ready to accept inputs from user	High
			Write the "First Name" in this input field using allowed characters (a-z), (A-Z)	1) The first name is be displayed as '(a-z) and (A-Z) ' for each character input 2) The legal "first name" should be accepted.	High
ASE_IN_9.9.1.6	Verification of functionality of	The new UI for registration is	Click on the "Last Name" input field	The "Last Name" tab is ready to accept inputs from user	High

	"Last Name" input field	being displayed	write the "Last Name" in this input field using allowed characters (a-z), (A-Z)	1) The last Name is be displayed as '(a-z) and (A-Z) ' for each character input 2) The legal "last name" should be accepted.	High
ASE_IN_9.9.1.7	Verification of functionality of "username" input feld	The "First Name" and "Last Name" input fields have been entered	Click on the "User Name" input field	The "User Name" tab is be ready to accept inputs from user	High
			write the "User Name" in this input field using allowed characters (a-z), (A-Z) and (0-9) with minimum of 8 characters (along with digits).	1) The "User Name" is displayed as '(a-z) and (A-Z) and (0-9) ' for each character input 2) The legal "User Name" should be accepted.	High
ASE_IN_9.9.1.8	Verify functionality of "Password" input field	The "First Name", "Last Name" and "User Name" input field has been entered	Click on the "Password" tab	The "Password" tab allows user to input password	High
			write the password in this tab using allowed characters (a-z), (A-Z) and (0-9) with minimum of 8 characters (along with digits).	1) The password is displayed as '*' for each character input. 2) The legal password should be accepted.	High

			write the password using non-allowed characters or with wrong length of character in this tab	1) The password will not be displayed and alert should be prompted to user "Use characters in lowercase (a-z), uppercase (A-Z) or numerals(0-9). Minimum length is 8 characters" 2) The password field should turn blank	High
ASE_IN_9.9.1.9	Verify the correct format(local-port@domain) of email id in the "e-mail" input field.	The "First Name", "Last Name", "User Name" and "Password" input field has been entered	Click on the "E-mail" input field	The "E-mail" tab is ready to accept inputs from user	High
			write the email id with proper format in this input field as follows : 1)write the "local-port" for e.g. "john.perry" 2)write"@" symbol followed by "local-port" for e.g. "john.perry@" 3)write the domain part followed by the "local-port" as well as the "@" symbol for e.g. john.perry@gmail.com	The "E-mail" input field accepts it.	High
			write the email id with wrong format in this input field	The "E-mail" input field field doesn't accept it and an error message is prompted to the user	High

ASE_IN_9.9.1.10	Verify the functionality of the "Mobile Number" input field	The "First Name", "Last Name", "User Name", "Password" and "E-mail Id" input field has been entered	Click on the "Mobile No" input field	The "Mobile No" input field allows user to input password	High
			write the password in this tab using allowed 10 digits from 0 to 9	1) The mobile no is displayed as '0 to 9' for each character input. 2) The legal password is accepted.	High
			write the password using non-allowed characters or with wrong length of character in this tab	1) The password will not be displayed and alert is prompted to user "Use 10 numbers of characters in numerals (0-9) only. 2) The password field turns blank	High
ASE_IN_9.9.1.11	Verify functionality of "Register" Button	The "First Name", "Last Name", "User Name", "Password", "E-mail Id", and "Mobile number" input fields have been entered	Click "Register" Button	The app screen shows a screen for "Token".	High
ASE_IN_9.9.1.12	Verification of functionality of "Token" Screen	"Register" button has been clicked	Verify UI	1) The app screen displays a notification "A token has been sent to your email , please enter the token no: (6 digits)". 2) A token input field is displayed followed by a "Submit" button.	High

ASE_IN_9.9.1.13	Verification of functionality of "Token" input field	"Token" verification screen is being displayed	Enter Token received by email.	Token field is entered with valid digits.	High
ASE_IN_9.9.1.14	Verification of functionality of "Submit" button	"Token" verification screen is being displayed	Click "Submit" button with correct token	1) The submitted token is verified with the app system. 2) On validation, the account is registered and stored on the ITU database. 3) The message " Account Activated" is prompted on screen.	High
			Click "Submit" button with incorrect token	1) The submitted token is verified with the app system. 2) On invalid detection, the message "Invalid Token number" is prompted on screen.	High

9.10 Test Case – Login

Project Name	ITU - Notifier
Screen Name	Login Screen
Date of creation	8/15/2016
Date of review	8/22/2016

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority
ASE_IN_9.9.2.1	Verification of the user interface of the ITU-Notifier system	Application is opened and User is registered on the app	Verify UI of the ITU-Notifier system	The UI of the app is displayed as follows: 1) The app name " ITU-Notifier" is displayed at the top left side of the application. 2) It shows following two input fields : a)"E-mail" b)"Password" 3) It is followed by two buttons: a)"Sign In" b)"Register"	Medium
ASE_IN_9.9.2.2	Verify the correct format(local-	The new UI for login is being displayed	Click on the "E-mail" input field	The "E-mail" tab is ready to accept inputs from user	High

	port@domain) of email id in the "e-mail" input field.		write the email id with proper format in this input field as follows : 1)write the "local-port" for e.g. "john.perry" 2)write"@" symbol followed by "local-port" for e.g. "john.perry@" 3)write the domain part followed by the "local-port" as well as the "@" symbol for e.g. john.perry@gmail.com	The "E-mail" input field accepts it.	High
			write the email id with wrong format in this input field	The "E-mail" input field field isn't accept it and an error message is prompted to the user	High
ASE_IN_9.9.2.3	Verify functionality of "Password" input field	The "E-mail" input field has been entered	Click on the "Password" tab	The "Password" tab allows user to input password	High
			write the password in this tab using allowed characters (a-z), (A-Z) and (0-9) with minimum of 8 characters (along with digits).	1) The password is displayed as '*' for each character input. 2) The legal password is accepted.	High

			write the password using non-allowed characters or with wrong length of character in this tab	1) The password isn't displayed and alert is prompted to user "Use characters in lowercase (a-z), uppercase(A-Z) or numerals(0-9). Minimum length is 8 characters" 2) The password field turns blank	High
ASE_IN_9.9.2.4	Verify functionality of "Sign In" Button	Valid email id and pasword have been entered	Click the "Sign In" Button	The "Sign In" button initiates login validation	High
ASE_IN_9.9.2.5	Validation of login credentials	"Sign in" button has been clicked	Validate the login id and password with the ITU Notifier database.	1) If validated, a new screen is opened with following buttons: 1) "Announcements" 2) "Broadcast Message" 3) "Subscription" 4) "View Profile" 5) "Log Out" 2) If not validated, alert is prompted "Invalid user id or password"	High

9.11 Test Case – User Home Page

Project Name	ITU - Notifier
Screen Name	User Home Page
Date of creation	8/15/2016
Date of review	8/20/2016

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority
ASE_IN_9.9.3.1	Verification of the user interface of the ITU-Notifier system for Broadcast Screen	Application is opened and User is signed in to the app	Verify UI of the ITU-Notifier system for user home page	The UI of the app is being displayed as follows: 1) The app name " ITU-Notifier" is being displayed at the top left side of the application. 2) Following buttons is being displayed :- a) Announcement b) Broadcast Message c) Subscribe d) View Profile e) Logout	Medium
ASE_IN_9.9.3.2	Verification of the functionality of the "Broadcast Message" button	Application is opened and the user is viewed the UI screen of user home page	Click on the "Broadcast Message" button	1)The button is selected. 2)A new UI is opened for "Broadcast Message" screen	High

ASE_IN_9.9.3.3	Verification of the functionality of the "Subscribe" button	Application is opened and the user is viewed the UI screen of user home page	Click on the "Subscribe" button	1)The button is selected. 2)A new UI is opened for Subscribe screen	High
ASE_IN_9.9.3.4	Verification of the functionality of the "View Profile" button	Application is opened and the user is viewed the UI screen of user home page	Click on the "View Profile" button	1)The button is selected. 2)A new UI is opened for "View Profile" screen	High
ASE_IN_9.9.3.5	Verification of the functionality of the "Log Out" button	Application is opened and the user is viewed the UI screen of user home page	Click on the "Log Out" button	1)The button is selected. 2)A new UI islogged out from the ITU Notifier app.	High

9.12 Test Case – Announcement

Project Name	**ITU - Notifier**
Screen Name	**Registration**
Date of creation	**8/15/2016**
Date of review	**8/20/2016**

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority

ASE_IN_9.9.4.1	Verification of the user interface of the ITU-Notifier system for Announcement Screen	Application is opened and User is logged in to the app	Verify UI of the ITU-Notifier system for Announcement Screen	The UI of the app is displayed as follows: 1) The app name " ITU-Notifier" is displayed at the top left side of the application. 2) It shows "Delete" button at the middle just after the top left 3) It shows different announcements with "check box"	Medium
ASE_IN_9.9.4.2	Verification of the functionality of the "check box" button of the ITU-Notifier system	Application is opened and User is able to view all the announcements	Click on the "check box" button respect to each announcements	The "check box" is selected.	High
ASE_IN_9.9.4.3	Verification of the functionality of the "Delete" button	Application is opened and The particular announcement is got selected	Click on the "Delete" button	1)This allows the user to delete the selected announcements from the screen	High

9.13 Test Case – Subscription

Project Name	ITU - Notifier
Screen Name	Subscription
Date of creation	8/15/2016
Date of review	8/20/2016

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority
ASE_IN_9.9.5.1	Verification of the user interface of the ITU-Notifier system for Subscription Screen	Application is opened and User is signed in to the app	Verify UI of the ITU-Notifier system for Subscription Screen	The UI of the app is being displayed as follows: 1) The app name " ITU-Notifier" is being displayed at the top left side of the application. 2) List of categories is being displayed with their respective check box options 3) "Subscription" button is being displayed just after all categories	Medium
ASE_IN_9.9.5.2	Verification of the functionality of the "check box" button of	Application is opened and User is able to view all the announcements	Check on the "check box" button respect to each announcements	The "check box" is get selected with respective category.	High

	the ITU-Notifier system		Uncheck on the "check box" button respect to each announcements	1) The "check box" is get unselected with respective category. 2) A message is prompt as "Unsubscribe Successfully" with a button prompt "ok"	High
ASE_IN_9.9.5.3	Verification of the functionality of the "Subscribe" button	Application is opened and the categories has selected according to users choice	Click on the "Subscribe" button	1)The button is selected. 2)A message is prompt as "Subscribed Successfully" with a button prompt "ok"	High

9.14 Test Case – Broadcast Message

Project Name	ITU - Notifier
Screen Name	Broadcast Message
Date of creation	8/15/2016
Date of review	8/20/2016

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority
ASE_IN_9.9.6.1	Verification of the user interface of the ITU-Notifier system for Broadcast Screen	Application is opened and User is signed in to the app	Verify UI of the ITU-Notifier system for Broadcast Screen	The UI of the app is being displayed as follows: 1) The app name " ITU-Notifier" is being displayed at the top left side of the application. 2) List of categories is being displayed with a drop down menu. 3)Following input fields is being displayed with column headers 'Titles' and 'Description' 3) "Broadcast" button is being displayed just after all categories	Medium
ASE_IN_9.9.6.2	Verify funtionality of drop down menu	Drop down box is correctly displayed with text "Sell"	Click the drop down box	The drop down box is shown following options: "Sell", "Accommodation", "Lost and Found" and "Fall 2016 Software Engineering"	High
		The drop down menu is being displayed	Select "any" tab from the drop down menu list	The tab is selected for particular category.	High

ASE_IN_9.9.6.3	Verify functionality of "Title" input Field	"Title" input field is showing the selected	Click on “Titles" input field	1)This allows the user to input the title of the notification. 2)This is accepted by the "Title" input field	High
ASE_IN_9.9.6.4	Verify functionality of "Description" input Field	"Titles" input field is displayed with respective title	Click on “Description" input field	This allows the user to input the description of the notification.	High
			Enter the notes for "Description" input field within 200 words (word limit).	The contents of appointment notes is displayed properly.	High
ASE_IN_9.9.6.5	Verify functionality of "Add Picture" button	Filled "Titles" and "description" input field is displayed	Click on"Add Picture" button	1)This allows the user to attach any picture related to the respective notification. 2)This is accepted by the "Add Picture" button	High
ASE_IN_9.9.6.6	Verify functionality of "Broadcast" button	Filled Titles and "description" input field is displayed with attached picture	Click on “Broadcast" button	1)This allows the user to broadcast his notification into ITU notifier app. 2)This is accepted by the "Broadcast" button	High

9.15 Test Case – View Profile

Project Name	ITU - Notifier

Screen Name	View Profile
Date of creation	8/15/2016
Date of review	8/20/2016

Test case ID	Test Objective	Precondition	Steps:	Expected result	Test Priority
ASE_IN_9.9.7.1	Verification of the user interface of the ITU-Notifier system for View Profile Screen	Application is opened and User is signed in to the app	Verify UI of the ITU-Notifier system for View Profile Screen	The UI of the app is being displayed as follows: 1) The app name " ITU-Notifier" is being displayed at the top left side of the application. 2) Following user detail are being displayed with field data:- a. First Name b. Last Name c. User Name d. E-mail Id e. Subscribe Categories	High

10 Interfaces /Snapshots

10.1 Login Page:

11 Summary, Future Scope And Opportunities

11.0 Introduction

This chapter describes discuss the objectives of the system stipulated in earlier chapter, limitation of the system conclusion and recommendation of the system

11.1 Summary

ITU Notifier is an electronic notice and announcement system which will bring the ITU notice board virtually to your mobile phones. It will enable staff and students to send and receive notification alerts. This app will not only help to keep the users updated on what is going on in the college but will also make it easy them to broadcast their messages anywhere anytime.

11.2 Future Scope

The future scope of the project is that it can be used as any news giving application for college. This application of E-Notifier can be further extended to include following features.

1. Documents and PDF files :
   The attachments can be further improved to include PDF files or Doc files.
2. Feedback :
   Feedback about the notices can be taken. It can increase communication among connected members and any issue can be easily sorted out on the spot.

11.3 Opportunities and Lesion Learned

During the course of this project, we got an opportunity to learn Software engineering concepts. We got a chance to go through complete software development lifecycle including requirement engineering, designing, implementation, testing and acceptance. We learned android development along with system design, system modeling and risk managements.

11.6 References

1. Ian Sommerville, PEARSON . "Software Engineering - 9th Edition"

Available from:

"http://web.info.uvt.ro/~cmindruta/Curs_IS_SwEng/Software%20Engineering%20(9th%20Edition)%20by%20Ian%20Sommerville.pdf"

2. Online Diagram Software to draw Flowcharts, UML & more | creately

Available from:

"http://creately.com/"

3. Android Development:

“http://www.tutorialspoint.com/android/”

“https://developers.google.com/”

“https://developer.android.com”

“http://www.androidlearning.in/android-push-notifications-using-firebase-cloud-messaging-fcm-php-mysql/”

4. Trello Project Management Tool

Available from:

"https://trello.com/"

5. Bapco, W. (2008) "information management breakdown" information management in organizations.

5. UML Activity Diagram Symbols. (n.d.). Retrieved August 31, 2016, from https://www.edrawsoft.com/uml-activity-symbols.php

6. Draw.io - free flowchart maker and diagrams online. (n.d.). Retrieved August 31, 2016, from https://www.draw.io/