

ONLINE Notice Board

Submitted to GC University Lahore
In partial fulfillment of the requirements
For the award of degree of

Bachelor's
IN
Computer Science

By

NAME Rimsha Saleem & Murtaza Ali

SESSION 2013 -2017

ROLL No 049 –BSCS -2013 & 031 -BSCS -2013

GOVT. COLLEGE UNIVERSITY
LAHORE

DEPARTMENT OF COMPUTEFR SCIENCE

DECLARATION

We, **Murtaza Ali** Roll Number **031** and **Rimsha Saleem** Roll Number **049** student of BS. in the subject of Computer Science session 2013-2017, hereby declare that the matter printed in the thesis titled **ONLINE Notice Board** is my own work and has not been printed, published and submitted as Development work in any University, Research, thesis, Development Institution etc. In Pakistan or abroad.

Dated: _____

Signatures of Deponent

RESEARCH COMPLETION CERTIFICATE

Certified that the research work contained in this thesis titled **ONLINE Notice Board has** been carried out and completed by **Rimsha Saleem & Murtaza Ali** .Roll No **049-BSCS-2013 and 031-BSCS-2013** under my supervision.

Date

Supervisor Signature

Name _____

Submitted Through

Prof. _____

Chairperson

Department of _____

GC University Lahore.

_ Controller of

Examinations

GC University Lahore.



In the Name of Allah, the Most Beneficent, the Most Merciful

Read! And the Lord is most Honorable, And Most Benevolent.

Who taught (to write) by pen He taught man that which he knew not.

(Sura Al-Alaq 30:3-5) Al-Quran

ACKNOWLEDGEMENT

All praises to Almighty Allah Who make me able to complete this task. My words of special thanks to Dr.Asad Raza Kazmi, Director, Department of Computer Science for his guidance and support. Next we want to say special thanks Ms. Ayesha Atta, our research supervisor who played a major role in fulfillment of this thesis, without their valuable direction this might be an impossible for us. We would like to express our gratitude towards faculty and administration of GC University for their kind co-operation and encouragement which help us in completing this project. We would like to say Thanks to our beloved Parents who gave us everything we ever asked and helped us at every step of life and our friends also deserve many Thanks for their loving encouragement and prayers.

ABSTRACT

ONLINE Notice Board is based on website and android work where student sign up from android application and student request approved by admin. Admin can delete any request and approved students from website. Admin use website application while students use android application. Admin register student and send them notices semester wise, roll number wise, section wise, session wise, morning/evening shift wise notices to students. Notice can be Holiday notice, Result Mark Sheet, Fee Notice, job notice, Time Table notice, Sport notice, Date sheet notice. Main Admin has right to add sub admin. Sub Admin can also upload notice on the websites. Student can easily Download/View/View online notices on their mobile phones using network. By this application, there is no need to visit university notice board for the students. They can receive online notice board information by sitting at their home by notification. The project is solely developed for the institutes and universities

Contents

Chapter 1 Introduction to Project.....	III
1.1 Introduction.....	1
1.1.1 Purpose.....	1
1.1.2 Outcome of Project	2
1.1.3 Tools and Languages	2
Chapter 2 Literature Review	3
2.1 Existing Systems	3
2.1.1 Proposed System.....	3
Chapter 3 Software Requirement Specification.....	5
3.1 Document Purpose	5
3.2 Meeting with stakeholder.....	5
3.3 Scope of Development Project.....	5
3.4 Functional Requirements	6
3.5 Other Non-functional Requirements.....	6
3.5.1 Performance Requirement	6
3.6 Requirement attributes	7
3.7 User Requirement	7
3.8 Assumptions and Dependencies	7
3.9 System Requirement	7
3.9.1 Software Configuration.....	7
3.9.2 Hardware Configuration	8
3.10 Data Requirement	8
3.11 External Interface Requirement	8
Chapter 4 Project Design	9
4.1 Methodology	9
Figure 4.1 Process Model	9
4.2 Architecture Overview.....	10
4.2.1 Data Flow Diagrams	11
4.3 Design Description.....	12
Use Case.....	12
4.3.2 Activity Diagram	27
4.3.3 Sequence Diagram	38
4.3.4 Entity Relationship Diagram.....	39
4.3.5 Class Diagram.....	40
Chapter 5 Implementation.....	41

5.1 Development Stages.....	41
5.1.1 Feasibility Study	41
5.1.2 Information Collection.....	43
5.1.3 Lifecycle Description.....	43
5.2 User interface	45
5.2.1 Prototype	45
Chapter 6 Evaluation.....	57
6.1 Unit Testing	57
6.2 Functional Testing	61
6.2.1 Sign up Android Application	61
6.2.2 Login in Android Application.....	62
6.2.3 Download/view File.....	62
6.2.4 Logout Android Application.....	63
6.2.5 Verify Student Request	63
6.2.6 Delete Request	64
6.2.7 Delete Approved Student.....	64
6.2.8 Login web site.....	65
6.2.9 Logout web site.....	65
6.2.10 Register Student with Course.....	66
6.2.11 Upload File.....	66
6.2.12 Add Admin.....	67
6.2.13 Add Session	67
6.2.14 Add Section.....	68
Chapter 7 Conclusion and Future Work	69
7.1 Conclusion	69
7.2 Future Work.....	69
Bibliography	70
Admin	71
Student	71

LIST OF TABLES

Table #	Description	Page #
1.13	Tools and Languages	2
4.3.1.1	Narrative Sign Up android application	12
4.3.1.2	Narrative Log In android application	13
4.3.1.3	Narrative Download view file	14
4.3.1.4	Narrative Verify Student Request	15
4.3.1.5	Narrative Delete Request	16
4.3.1.6	Narrative Delete approved student	17
4.3.1.7	Narrative Logout Android Application	18
4.3.1.8	Narrative Log In web site	19
4.3.1.9	Narrative Logout Web site	20
4.3.1.10	Narrative Register student with course	21
4.3.1.11	Narrative Upload File	22
4.3.1.12	Narrative Add Admin	23
4.3.1.13	Narrative Add Session	24
4.3.1.14	Narrative Add Section	25
6.1.1	Sign up Android application	57
6.1.2	Verify Student Request	57
6.1.3	Login Android Application	58
6.1.4	Download view File	58
6.1.5	Logout Android Application	58
6.1.6	Delete Student Request	58
6.1.7	Log in Website	59
6.1.8	Add Admin	59
6.1.9	Add session	59
6.1.10	Registration Student with Course	60
6.1.11	Add Section	60
6.1.12	Send notice	60
6.1.13	Logout website	61
6.1.14	Delete approve students	61
6.2.1	Sign up Android Application	61

6.2.2	Login in Android Application	62
6.2.3	Download view File	62
6.2.4	Logout Android Application	63
6.2.5	Verify student request	63
6.2.6	Delete Request	64
6.2.7	Delete Approved Student	64
6.2.8	Log In web site	65
6.2.9	Logout web site	65
6.2.10	Register student with course	66
6.2.11	Upload file	66
6.2.12	Add Admin	67
6.2.13	Add Session	67
6.2.14	Add Section	68

LIST OF FIGURE

Figure #	Description	Page
Figure 3.2	Meeting with stakeholder	5
Figure 4.1	Process Model	9
Figure 4.2	System Architecture	10
Figure 4.3.3	Sequence Diagram	38
Figure 4.3.4	Entity Relationship Diagram	39
Figure 4.3.5	Class Diagram	40
Figure 5.1	Development stage	41
Figure 5.1.3	Software Development Life Cycle	44
Figure 5.2.1.1	Notice Menu	45
Figure 5.2.1.2	Session Wise Notice	45
Figure 5.2.1.3	Semester Wis eNotice	46
Figure 5.2.1.4	Section Wise Notice	46
Figure 5.2.1.5	Morning Evening Wise Notice	47
Figure 5.2.1.6	Roll Number Wise Notice	47
Figure 5.2.1.7	Course Wise Notice	48
Figure 5.2.1.8	Degree Wise Notice	48
Figure 5.2.1.9	Applied Student List	49
Figure 5.2.1.10	Approved Student list	49
Figure 5.2.1.11	Student Course Register	50
Figure 5.2.1.12	Add Admin	50
Figure 5.2.1.13	Add Session	51
Figure 5.2.1.14	List All Session	51
Figure 5.2.1.15	Add Section	52
Figure 5.2.1.16	List All Section	52
Figure 5.2.1.17	List All Notice	53
Figure 5.2.1.18	Student Login Menu	54
Figure 5.2.1.19	Student Sign UP Menu	55
Figure 5.2.1.20	Student Main Menu	56

Chapter 1 Introduction to Project

1.1 Introduction

“Online Notice Board” is the combination of web base and android base application which provide notification of university notice board to students. Students can get any relevant information from university like Holiday notice, Result Mark Sheet, Fee Notice, job notice, Time Table notice, Sport notice, Date sheet notice. The website admin upload the notice in website of university department. Only admin can upload notices on the website of notice board. Admin can add sub admin, section and semester. Student gets that notice notification in their mobile phones using network. Students can also download/view the uploaded notice which can be a file or image. The students have to sign up the app for this purpose once and also have to select the section whose they want to get information about.

1.1.1 Purpose

Online social network have been growing so quickly. Their uses are accessing every type of information. New generation is addicted to online network. They will prefer online automatic services rather than finding out related information from university notice board.

- By this application, there is no need to visit university notice board for the students. They can receive online notice board information by sitting at their home by notification.
- Sometimes the students are not in university when notice has uploaded about result, time table etc. By this App, They are able to receive notice information at any time. As in Pakistan, many holidays announce suddenly so admin can upload holiday notice immediately and students can get this notification.
- The main thing is student can easily download/view notices. So people fell very easy to use this app to receive images, files from university website admin about study purpose, awareness of holidays, awareness of job offering from any company or institute easily.
- Students don't need to spend any money for this App. Students just have to sign up the App by giving your roll number of university so that they can avail this facility. Students just have access to network to receive notifications.

- The system is solely developed for the institutes and universities. The website provides a platform to students to receive online notification easily.

1.1.2 Outcome of Project

This project eliminates the need of checking notice board of university department regularly for students. Notification of notices come to them by online system. This application will give 24*7 accesses to an online notification of notices. Admin of university department provides valuable data and information according to the studies. Application gives platform for admin to upload notice board file or image any time rather than attaching notices to the notice board of department. Students feel easy to use this application.

1.1.3 Tools and Languages

Platform	Microsoft Windows XP, Windows 7,8,10.
Back-end (Used)	SQL Server 2014, JAVA.
Front-end (Used)	HTML, CSS, Bootstrap 3, JAVASCRIPT, Visual C# Razor Engine
Web Server	IIS Express, web hosting service
Programming Language	Asp .Net, JAVA, XML
Development Tools	Microsoft Visual Studio 2015 Microsoft SQL Server 2014 Management Studio Android Studio
Team Size	2
Project Objectives	Online Notification App is for students to get notice board notices by notification on their mobile phones while online. Admin can upload the notification for students. Students can also download/view notices
Project Duration	One year
Guided by	Miss Ayesha Atta
Submitted To	Department of Computer Science Govt. College University Lahore

Chapter 2 Literature Review

2.1 Existing Systems

There is a lot of work done in the past regarding to the software of notice board. Notice board is uploaded on that web base software. These software or web base applications do not involve android work. These existing software's have some drawbacks which is listed as below

- Students do not get their information in mobile phones by notifications.
- Students need to find their related information from website which is more time consuming process.
- It was difficult for the student to search their relevant information in websites.
- Such software does not provide information about midterm result, Final Subject marks.
- They does not provide semester wise, roll number wise, section wise , session wise, morning/evening shift wise notices to students

2.1.1 Proposed System

The development of online notification system will enable the users too effectively, accurately and efficiently process and receive the relevant information from institution without too much time consumption. The main goal of application is to give platform to students to receive online notice board notification so that they can download/view it online. It is also a platform where admin can upload notification. It will also provide the users the ease of use. The major benefits that will be achieved by the online notice board to all three stakeholders are as following.

2.1.1.1 Students

The basic user and beneficiary of the system will be the current students, who are looking for the university notice board regularly for their information according to their studies and any other related purpose. The product will provide the following benefits to the students which are following.

➤ **Online notification**

Students can receive their related information from notice board by online notification. They can get online notification in their mobile phones that are uploaded by admin. This feature finishes the difficulty for students to visit their university notice board regularly.

➤ **Online sign Up/Sign In/Sign Out**

Students have to sign up once the android application for achieving this facility of online notice board notification. Students can easily sign in the application. They also can select the following options which they want to receive information about.

- Semester wise
- Roll number wise
- Morning/Evening wise
- Section wise
- Session wise

Sign out option is also available for the students.

➤ **Download/view old/new notification**

Sometimes Students need old notices for any purpose. There is no limitation that students only download/view current notices. They can go to their notice board list where all notices are listed. They can download/view any one of it by clicking on it.

➤ **Easy to use**

Students feel laziness to visit notice board sometimes. They prefer online notification which provides them information without using any physical energy.

2.1.1.2 Admin

➤ **Reduce time to update**

As in manual process, Admin had to place the notices on the notice board of university in paper form which was also a time consuming process. By this product, he can upload notices in website directly with in a second.

➤ **Free of cost**

The free service is provided by this system. There is no need of money to print out the notices. Admin can upload any file and image free of cost. Updates, on an online notice board portal for free can help in better belief on the site.

Chapter 3 Software Requirement Specification

3.1 Document Purpose

The purpose of this document is to account all functional, non-functional and other appropriate requirements of Software Application. Overall description and scope of Software is defined. Reader can surely understand all requirements which depict Software complete functionality. Assumptions which are taken from this system are determined in this document. Reader can also move on the purpose and reason for developing this Software by reading it.

3.2 Meeting with stakeholder

After conducting meeting with stakeholders we can conclude the following results



Figure 3.2 Meeting with stakeholder

The customer/users wants impressive theme. They want software for their department which will facilitate each student.

3.3 Scope of Development Project

We are providing a solution according to which the students will get their related notification on their mobile phone. They don't need to visit notice board to see any sort of notification placed by university. Individual user/student can be able to notify by their related information through mobile notification.

- Our project is based on web application as well as android application.
- It provides a portal b/w student and notification admin.
- The mobile application is only for students to receive notifications and download/view it.
- The web application interface only for the notification admin where the admin upload and send the notification to students only.

3.4 Functional Requirements

➤ Admin

- Ability to upload file for students.
- Ability to verify the student request.
- Ability to approved or delete student request.
- Ability to see number of applicants on the site

➤ Student

- Ability to sign up/ sign in / sign out the application
- Downloading of received file through mobile interface.
- Ability to see notice board list.

3.5 Other Non-functional Requirements

3.5.1 Performance Requirement

- The performance of the system should be fast and accurate.
- Online notice board shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period.
- The system should be able to handle large amount of data. Thus it should accommodate high number of student and their information without any fault.
- The system shall be available 24*7.
- Usability
- The system shall have a clean interface with only needed features, clear terminology and tool tips wherever necessary. Warnings or alerts shall be specified in clear way.

➤ Reliability

Reliability defined as a measure of the time between failures occurring in a system (measures how frequently the system fails), so that the system shall operate without any failure for a particular period of time

➤ Security Requirement

- System will use secured database
- System will have different types of users and every user has access constraints
- Proper user authentication should be provided.

3.6 Requirement attributes

- There may be multiple members creating the project, all of them will have the right to create changes to the system. But other users cannot do changes about design and functionality.
- The user shall be able to easily get the notification.
- The user shall be able to easily download/view any file.

3.7 User Requirement

The users of the system are students and admin. The proper user interface help and the guide to run the system must be sufficient to educate the users on how to use the system without any problems.

The admin provides certain facilities to the users in the form of

- Data migration i.e. whenever user registers for the first time then the data is stored in the server
- Request of the applied student must be verified and approved.
- The server must be maintained regularly and it has to be updated from time to time

3.8 Assumptions and Dependencies

The assumptions are:

- The coding should be error free
- The system should be user-friendly so that it is easy to use for the users
- The information of all files, data must be stored in a database
- The system should have more storage capacity and provide fast access to the database
- The Online notice board is running 24 hours a day

The dependencies are:

- The specific hardware and software due to which the product will be run.
- The end users should have proper understanding of the product.
- The information of all the users must be stored in a database that is accessible by the admin.

3.9 System Requirement

3.9.1 Software Configuration

- Visual Studio (Compile to build software)
- SQL Server (Database engine taking care of database of the software)
- Asp.net MVC (Microsoft web framework to build interactive websites using MVC design patterns)
- ASP.net Web API (Microsoft web framework to build rest API which is intermediate b/w server and mobile application)
- JAVA for mobile application development.
- SQL (Structure query language to create, manage the database)

3.9.2 Hardware Configuration

Processor: Pentium(R) Dual-core CPU

Hard Disk: 40GB

RAM: 2GB or more

3.10 Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. In this project the inputs will be the queries as fired by the users like selecting files for individual user or for all. Now the output will be the file selected by user.

3.11 External Interface Requirement

➤ GUI(Graphical User Interface)

The software provides good graphical interface for the user, performing the required task such as sign up, sign in, sign out, uploading and downloading any file.

- It allows user to quick download/view like word, .rar etc. file.
- All the modules provided with the software must fit into this graphical user interface and accomplish to the standard defined.
- The design should be simple and all the different interfaces should follow a standard template.

Chapter 4 Project Design

4.1 Methodology

Process Model

The process model that we used in our project development is “Incremental Process Model”. The incremental build model is a method of software development where the model is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements.

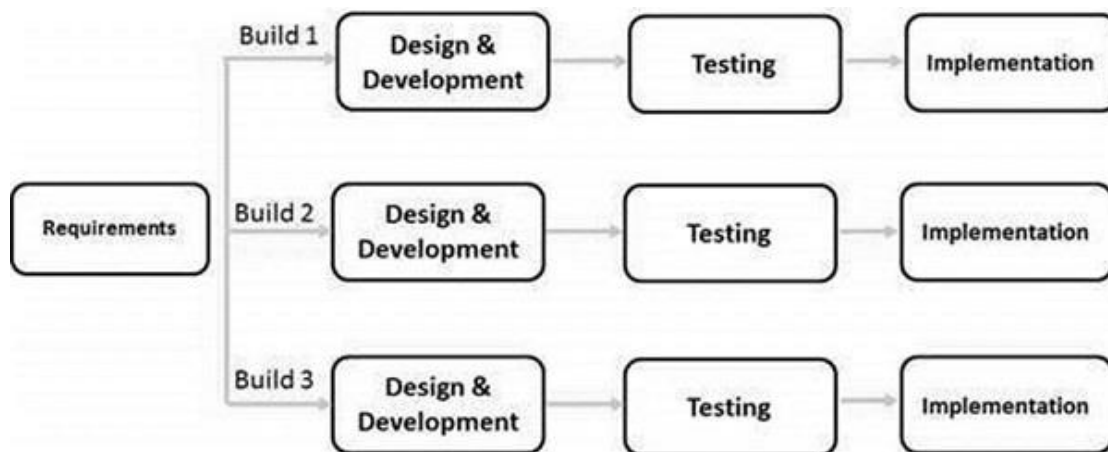


Figure 4.1 Process Model

4.2 Architecture Overview

The architecture of our Online Notice Board is a basic structural framework that identifies the major components of our system and the communications between these components.

Below is a general model for the whole system.

There are two types of users.

- Admin
- Students

This Diagram shows the architecture of our whole system. Besides the admins and students there is Database which stores the overall information.

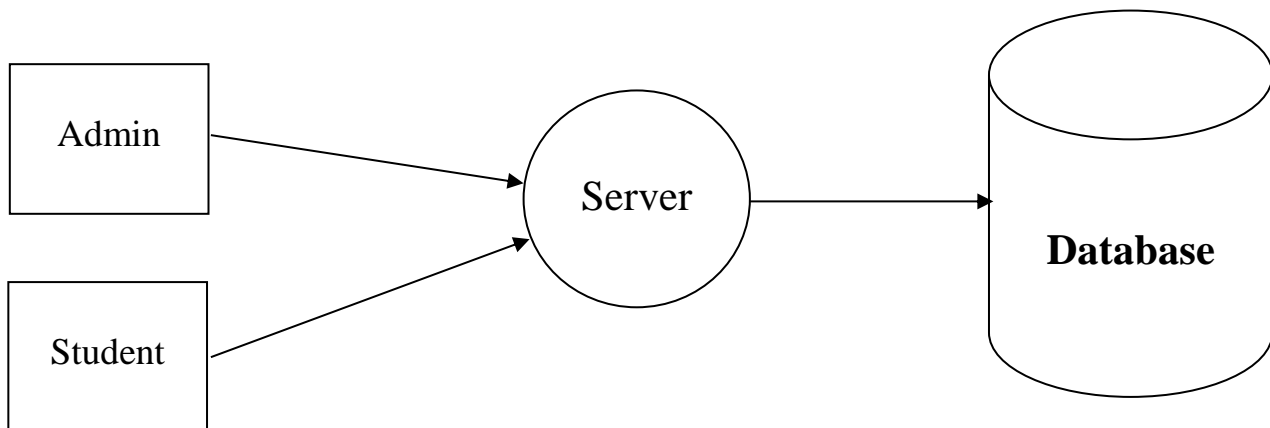


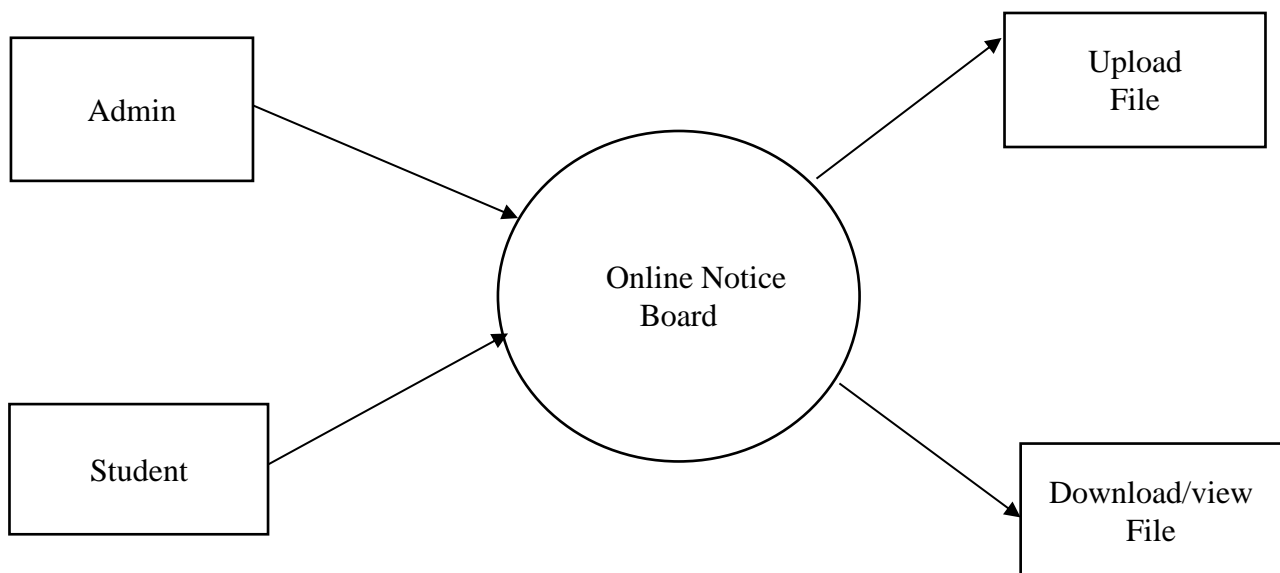
Figure 4.2 System Architecture

4.2.1 Data Flow Diagrams

A Data Flow Diagram shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored.

A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through the information, modeling its process aspects.

Level 0



4.3 Design Description

Use Case

4.3.1.1 Narrative Signup Android Application

This is online notification system named Online Notice Board. In this system students sign up android application for receiving notifications.

Use Case	Sign Up android application
Primary Actor	Student
Goal in Context	To make the account for online notifications.
Preconditions	Students should have application and has network connection for sign up.
Trigger	Student data will store in database.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Student selects name textbox and fills it by writing name. • Student selects Roll Number textbox and fills it by writing roll number. • Student selects 'Degree Title' drop down option. • Student selects session textbox and fills it by writing correct session. • Student selects semester textbox and fills it by writing correct semester. • Student select Morning / Evening drop down option. • Student Write Password • After filling form , student press sign up button
Exceptions	<ul style="list-style-type: none"> • If the student gives numeric data in name then it will not proceed. • If the student gives incorrect roll number, session format and semester greater than 8, it will give error. • Network is unavailable. • If student does not fill any asked information then sign up will not be succeeded

4.3.1.2 Narrative Login Android Application

In this case students have to sign up their account for receiving online notification.

Use Case	Log In android application
Primary Actor	Student
Goal in Context	To sign in the account for receiving online notification
Preconditions	Students should have network connection and account in android application.
Trigger	Student account will open.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Student select roll number textbox and write name. • Student selects 'Degree Title' drop down option. • Student selects session textbox and fills it by writing correct session. • Student Write Password • After filling form, student press sign In button
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • If the student gives numeric data in name then it will not proceed. • If the student gives incorrect roll number and session format it will give error. • If the student does not fill any asked information then sign in will not be succeeded.

4.3.1.3 Narrative Download/View File

In this case students can download/view any file which is uploaded by admin.

Use Case	Download/view file
Primary Actor	Student
Goal in Context	To download/view file of notice board.
Preconditions	Students should have network connection and account in android application.
Trigger	File starts to download/view.
Secondary Actors:	System
Scenario	<ul style="list-style-type: none"> • Student goes to the notice board list. • Select specific file which has notice title. • File start to Download/view on student mobile phones
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Not enough memory in student mobile phones. • Network is unavailable. • Server is unavailable.

4.3.1.4 Narrative Verify Student Request

This is online notification system named Online Notice Board. In this system, Admin verify the student's request so that they can able to get online notification.

Use Case	Verify student request.
Primary Actor	Admin
Goal in Context	To approve the student request
Preconditions	Students should have web application and has network connection for sign up and there must be appending request.
Trigger	Student data will store in database and go to the approved student table.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin login the web site. • In main menu, Click validate student button. • Click validate link. • Student will be validated.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable. • If admin try to validate already approved student's request.

4.3.1.5 Narrative Delete Request

This is online notification system named Online Notice Board. In this system Admin have right to delete the student request or any other request which is unknown.

Use Case	Delete Request
Primary Actor	Admin
Goal in Context	To delete request by admin which is unknown.
Preconditions	Admin should have network connection and log in the web site.
Trigger	Student request will be deleted and removed from database.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin login account of web site • Click validate student button. • Click delete link. • Applied student request will be deleted.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable.

4.3.1.6 Narrative Delete approved student.

This is online notification system named Online Notice Board. In this system admin can approve students and can also remove student to get any notification.

Use Case	Delete approved student.
Primary Actor	Admin
Goal in Context	To delete approved student
Preconditions:	Admin should has network connection and log in the web application
Trigger	Student data will remove from database.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin login the account of wed application. • Click approves student button. • Click delete approved student. • Specific Approved student will be deleted and will be removed from database.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable.

4.3.1.7 Narrative Logout Android Application.

This is online notification system named Online Notice Board. In this system, Student can logout the account.

Use Case	Logout Android Application.
Primary Actor	Student
Goal in Context	To logout student account.
Preconditions	Students should have android application, has network connection and already login the account.
Trigger	Student logout the account and go to login panel.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin click logout button. • Account will be logout.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable.

4.3.1.8 Narrative Login Website

This is online notification system named Online Notice Board. In this system Admin sign in web application so that he can approve student and upload notification on noticeboard.

Use Case	Log In web site
Primary Actor	Admin
Goal in Context	To login the account.
Preconditions	Admin should have network connection and have web application.
Trigger	Admin account will be open.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin go to login menu. • Enter ID. • Enter password. • Click log in Button. •
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Admin enter invalid ID, Password. • Server is unavailable.

4.3.1.9 Narrative Logout Website

This is online notification system named Online Notice Board. In this system, Admin can logout the account.

Use Case	Logout Web site
Primary Actor	Admin
Goal in Context	To logout admin account
Preconditions	Admin should have web application, has network connection and already login the account.
Trigger	Admin logout the account and go to login panel.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin click logout button. • Account will be logout.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable.

4.3.1.10 Narrative Register Student with Course

In this system, Admin register the student's either they are regular students or repeat students. Admin register the student subject wise.

Use Case	Register student with course
Primary Actor	Admin
Goal in Context	To register the student subject wise.
Preconditions	Admin should have web application, has network connection and already login the account and have approved student list.
Trigger	Student will register with subject wise and data will store in database.
Secondary Actors:	System
Scenario	<ul style="list-style-type: none"> • Admin goes to registration panel. • Enter course code in which student is enrolled. • Enter student roll number. • Select degree title from drop down option. • Enter session of student. • Select register type i-e. Regular student, repeater etc. • Click register button.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Admin enter wrong data. • Server is unavailable.

4.3.1.11 Narrative Upload File

In this system, Admin upload file on notice board for students.

Use Case	Upload File
Primary Actor	Admin
Goal in Context	To Upload notice/file
Preconditions	Admin should have web application, has network connection and have register student.
Trigger	File will be uploaded on website.
Secondary Actors:	System
Scenario	<ul style="list-style-type: none"> • Admin go to notice category selection panel. • Select categories. • Go to upload panel. • Input notice description. • Click send.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable •

4.3.1.12 Narrative Add Admin

In this system, Admin can add sub admin.

Use Case	Add Admin
Primary Actor	Main Admin
Goal in Context	To add sub admin
Preconditions	Admin should have web application, has network connection.
Trigger	Sub Admin data will be store on database.
Secondary Actors:	System
Scenario	<ul style="list-style-type: none"> • Admin go to add admin panel. • Fill form by entering employee ID, Name, Designation, admin type, password. • Click add
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • Server is unavailable •

4.3.1.13 Narrative Add Session

In this system, Admin can add session.

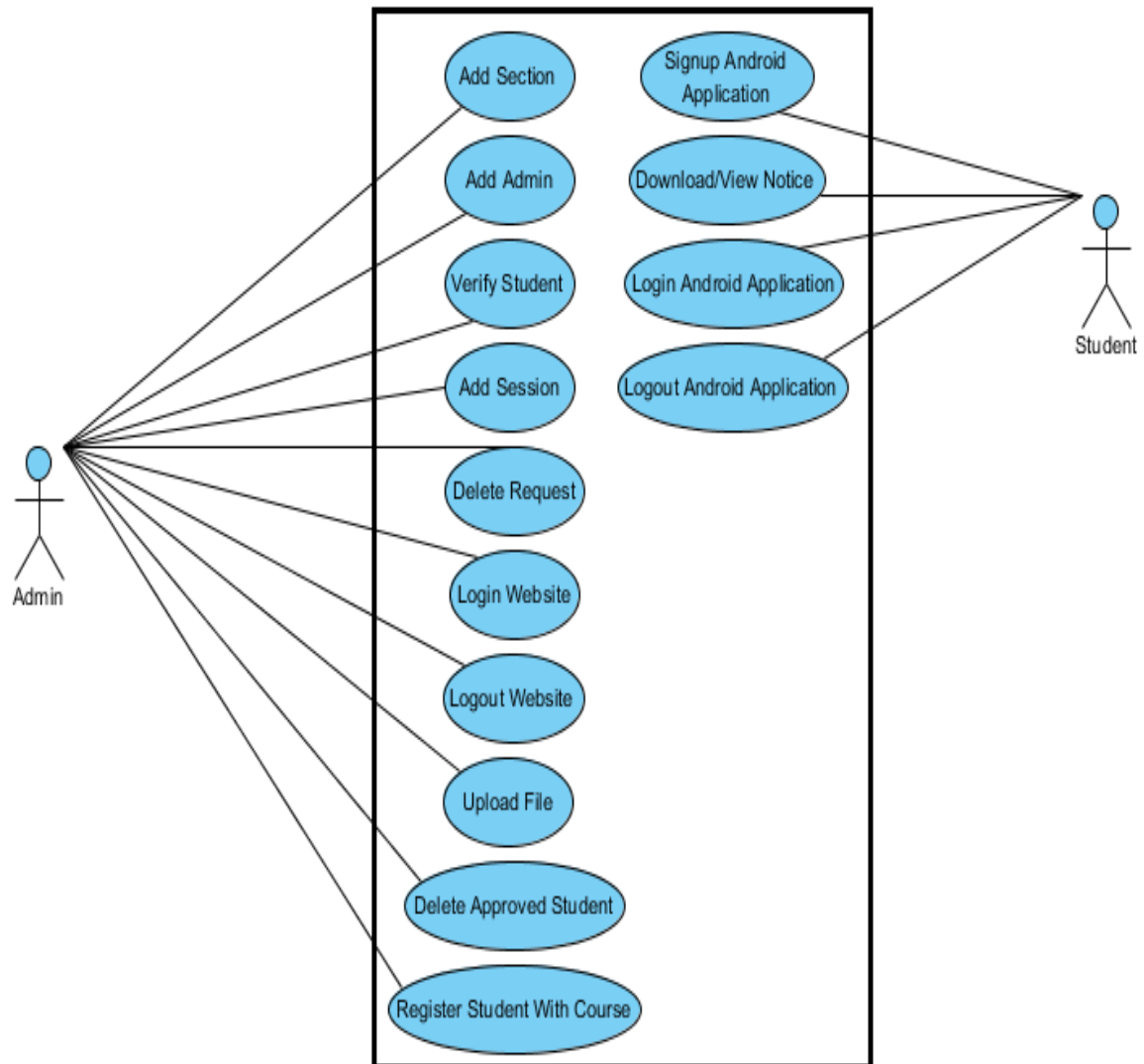
Use Case	Add session
Primary Actor	Admin
Goal in Context	To add session.
Preconditions	Admin should have web application, has network connection.
Trigger	New session will be added in database.
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin go Add session panel • Select degree title. • Enter session. • Click to create button
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • If admin try to enter preexisting session and invalid format. • Server is unavailable.

4.3.1.14 Narrative Add Section

In this system, Admin can add section.

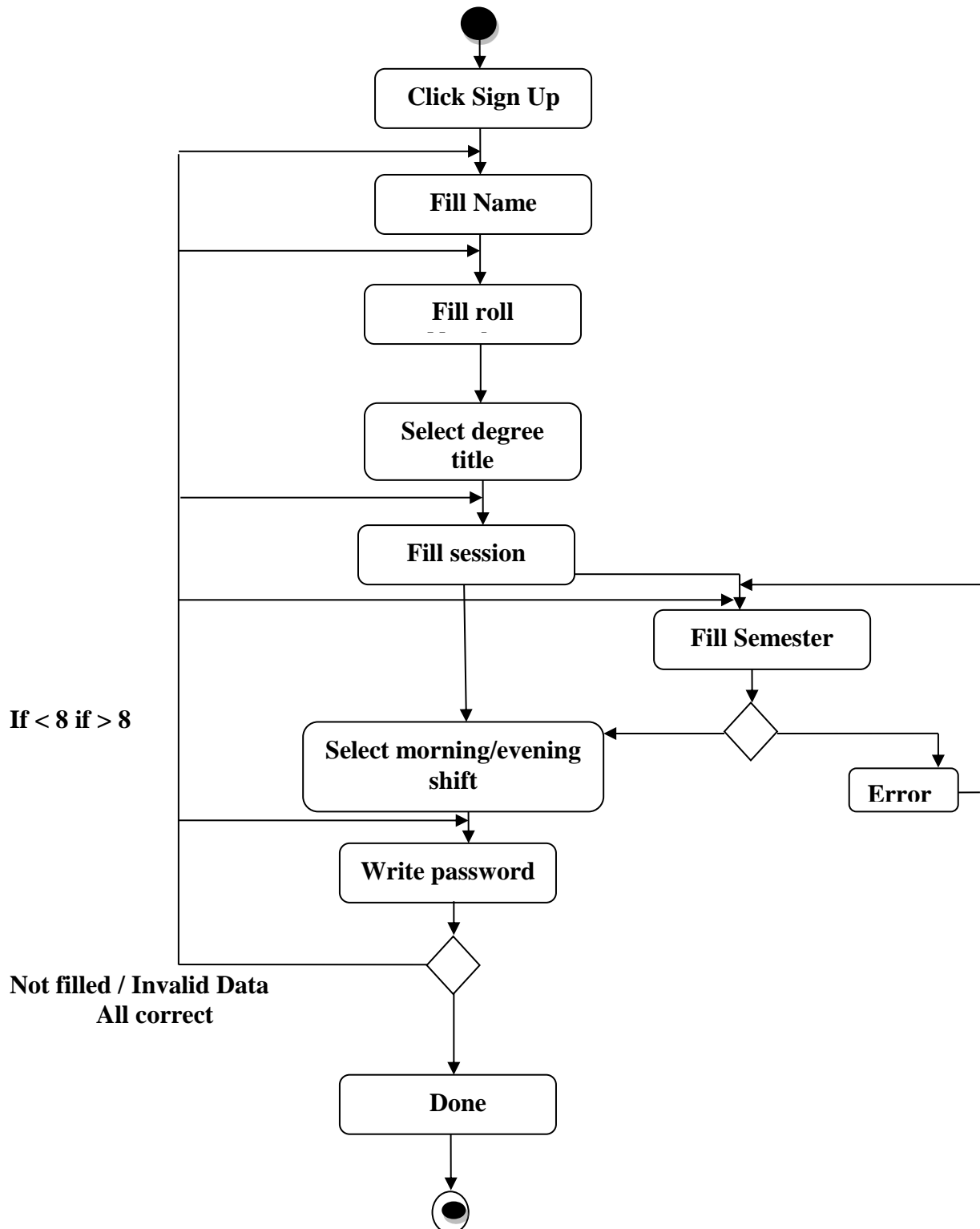
Use Case	Add section
Primary Actor	Admin
Goal in Context	To add section.
Preconditions	Admin should have web application, has network connection.
Trigger	New section will be added in database
Secondary Actors	System
Scenario	<ul style="list-style-type: none"> • Admin go Add section panel • Select degree title. • Enter session. • Click to create button.
Exceptions	<ul style="list-style-type: none"> • System is not in working condition. • Network is unavailable. • If admin try to enter preexisting section and invalid format. • Server is unavailable

4.3.1 Use Cases

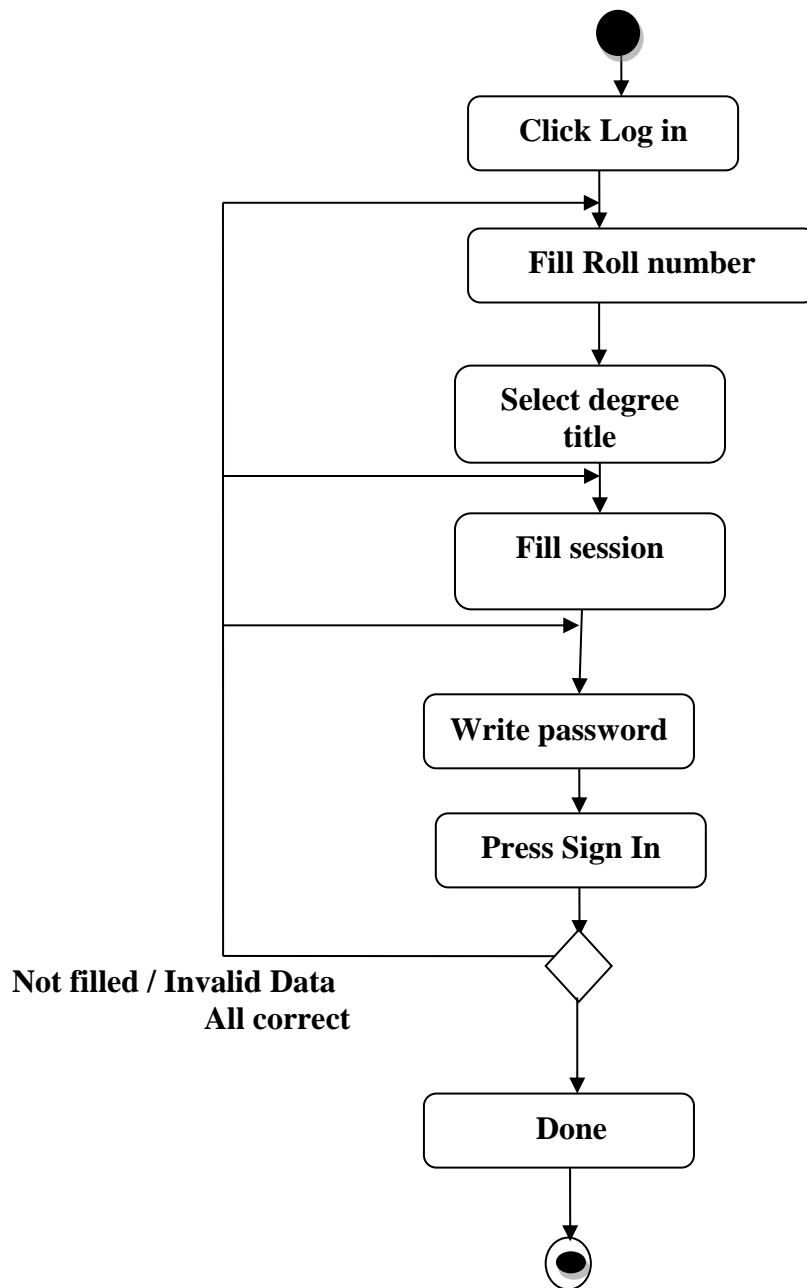


4.3.2 Activity Diagram

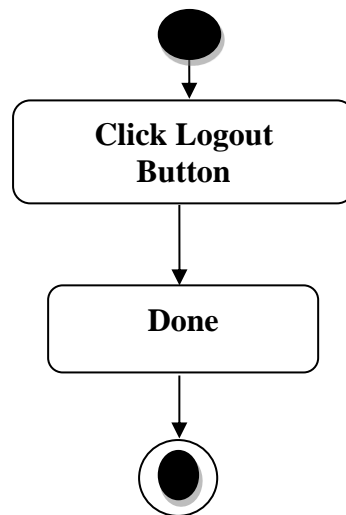
1. Sign Up Android Application



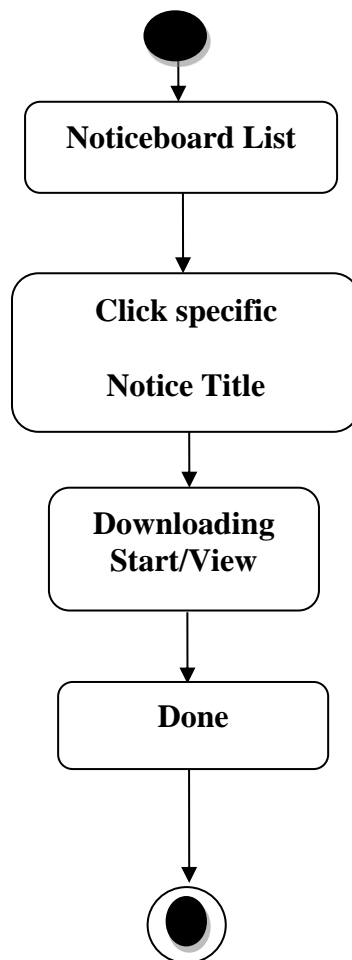
2. Login Android Application



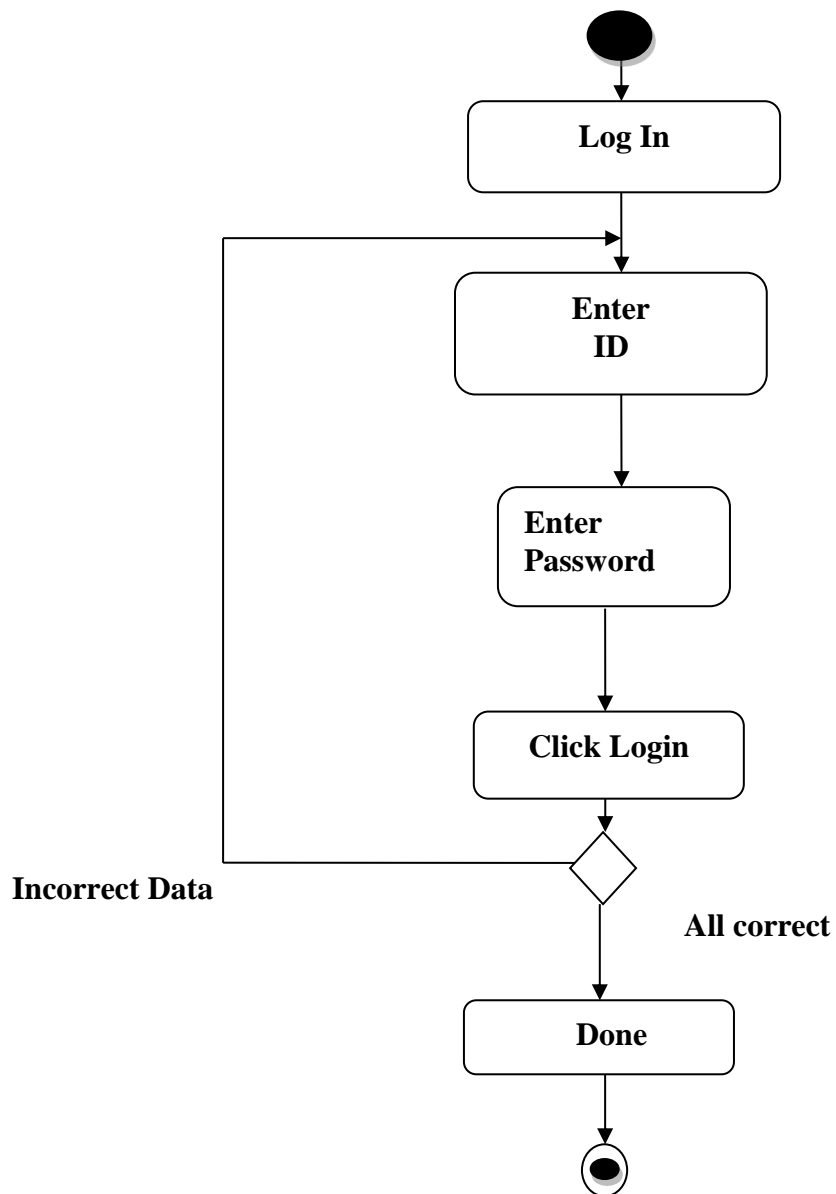
3. Logout Web site



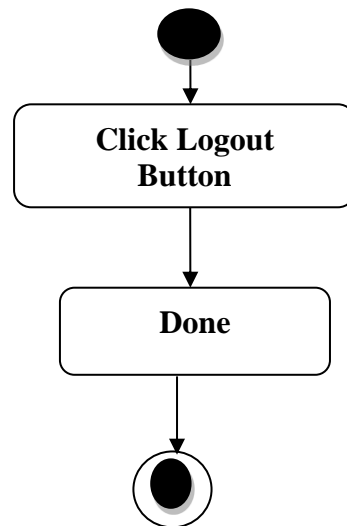
4. Download/view Notification



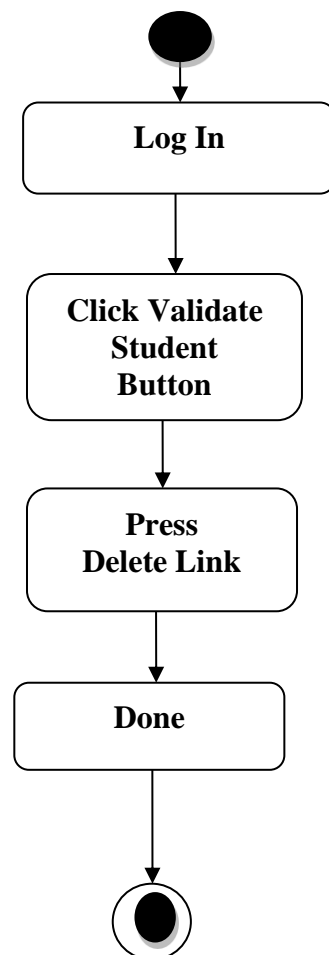
5. Log In web site



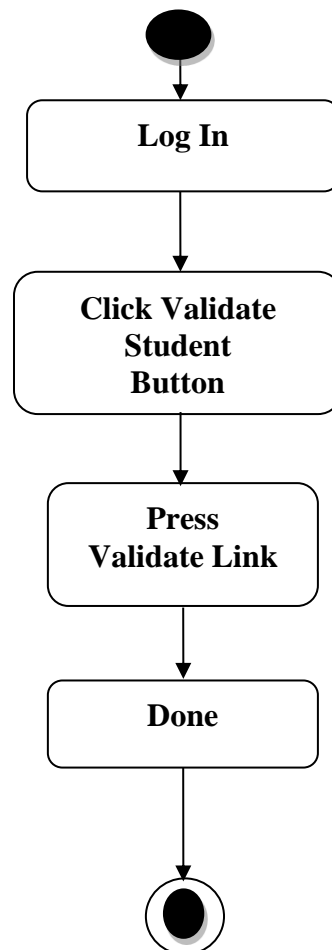
6. Logout Android Application



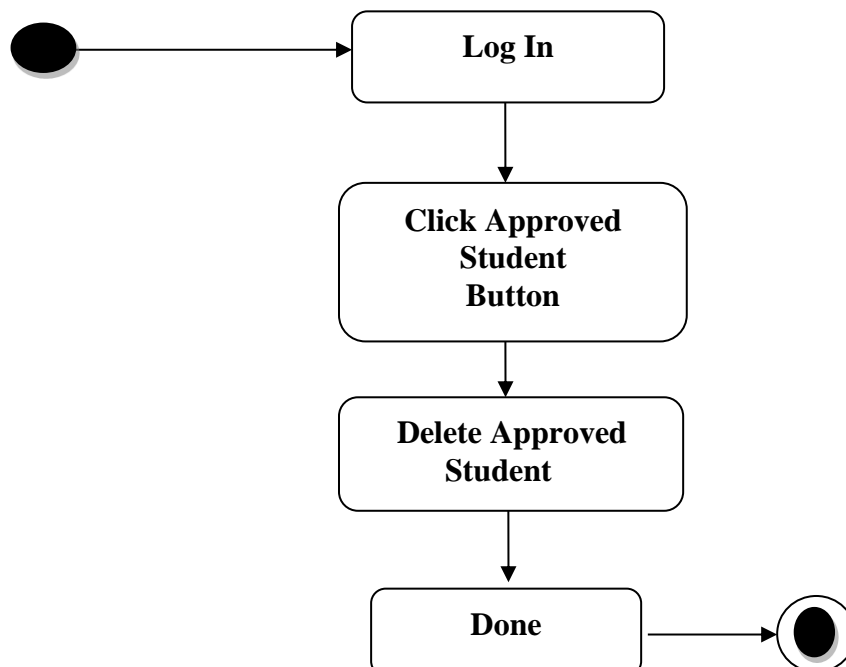
7. Delete Request



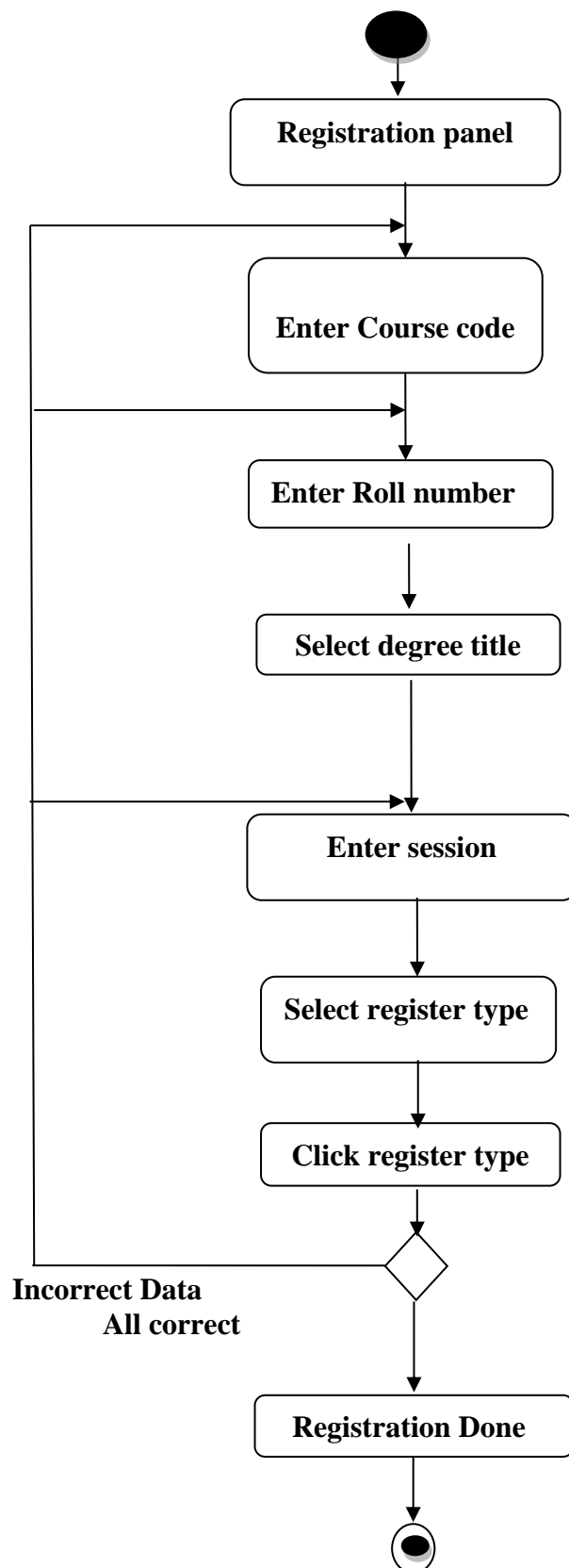
8. Verify Student Request



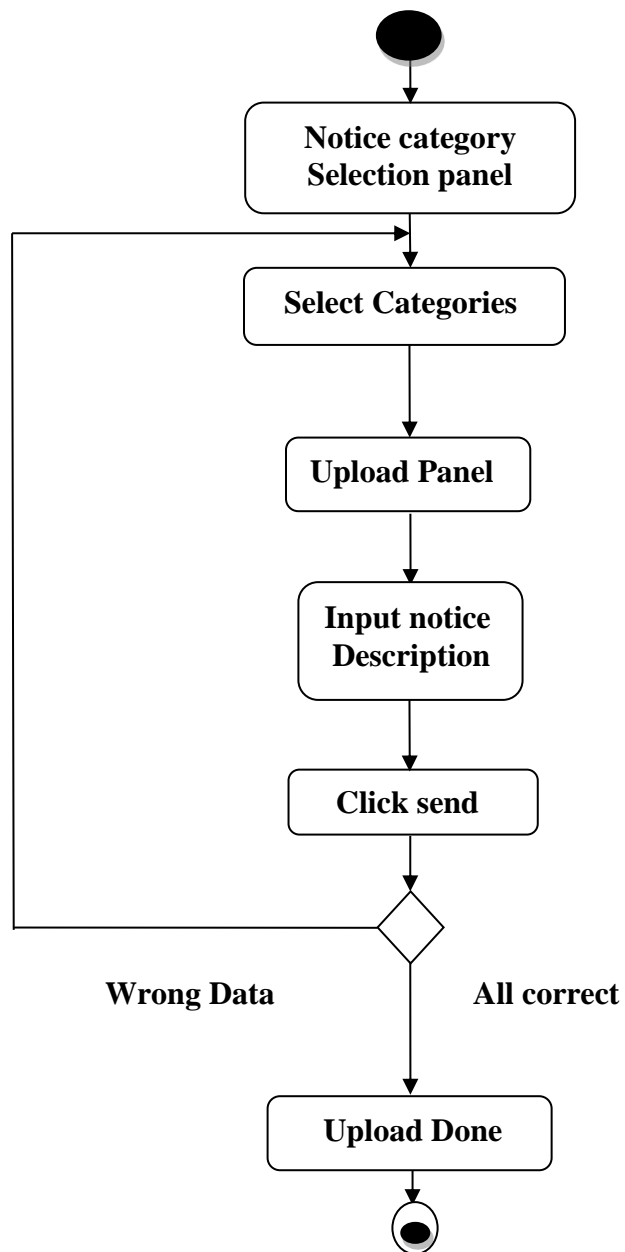
9. Delete Approved Student



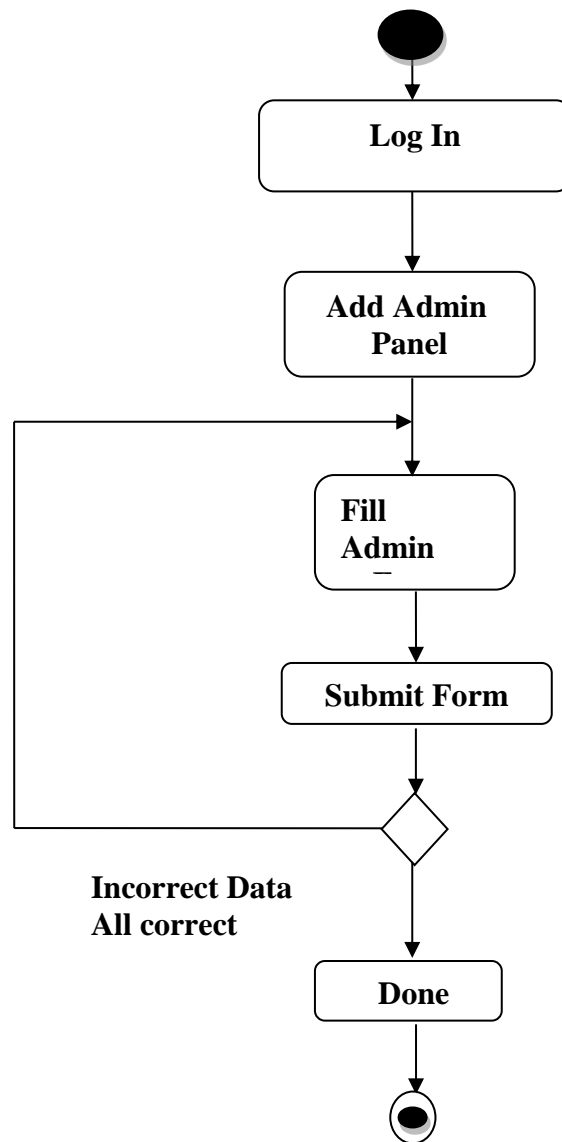
10. Register student with course



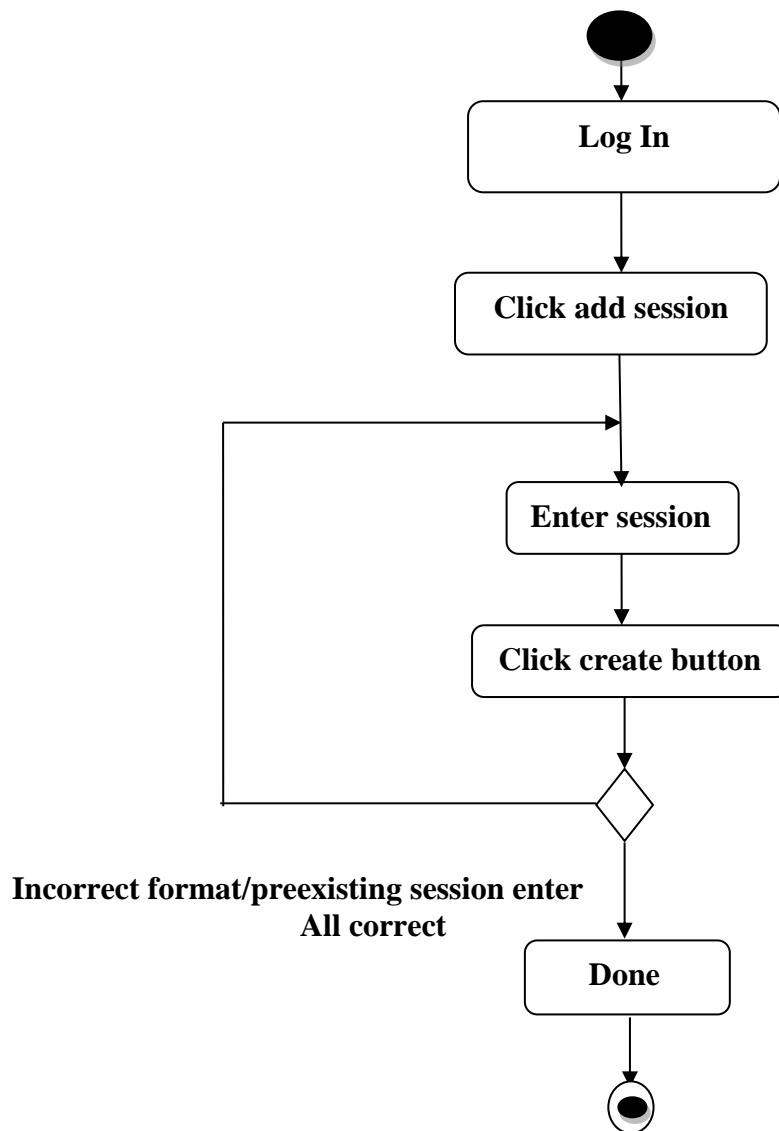
11. Upload File



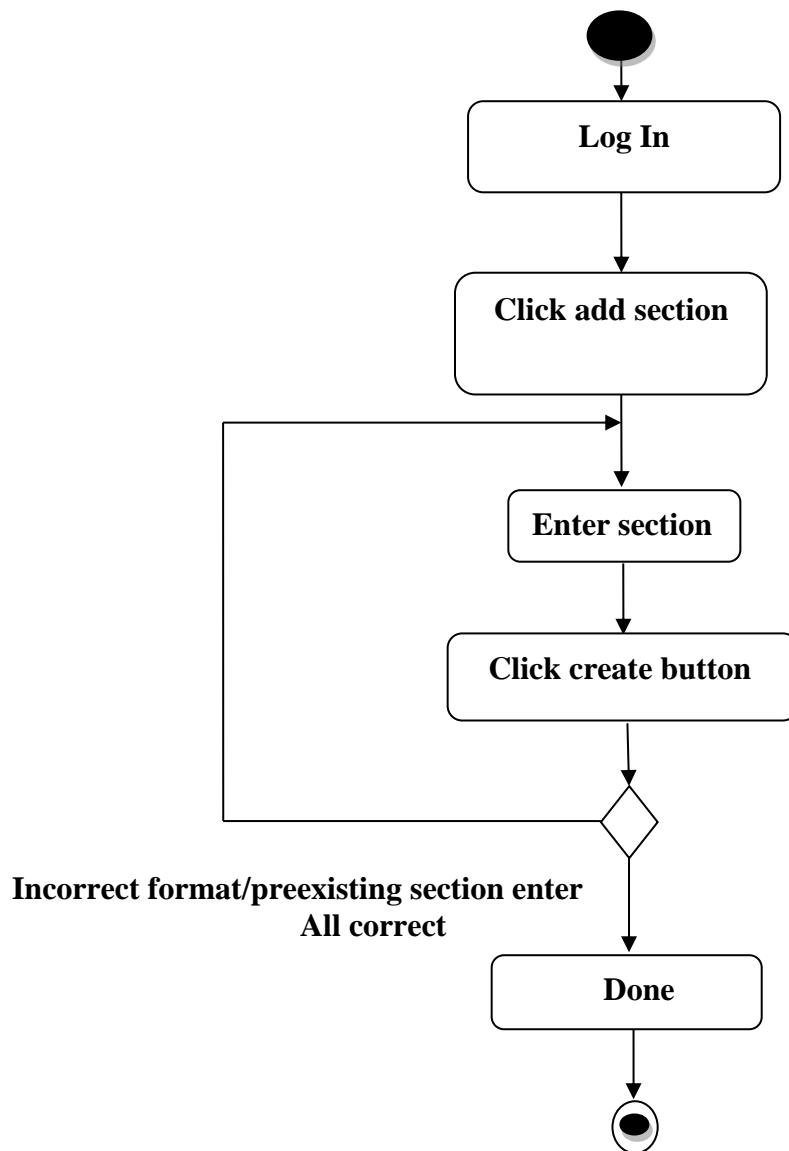
12. Add Sub Admin



13. Add Session



14. Add Section



4.3.3 Sequence Diagram

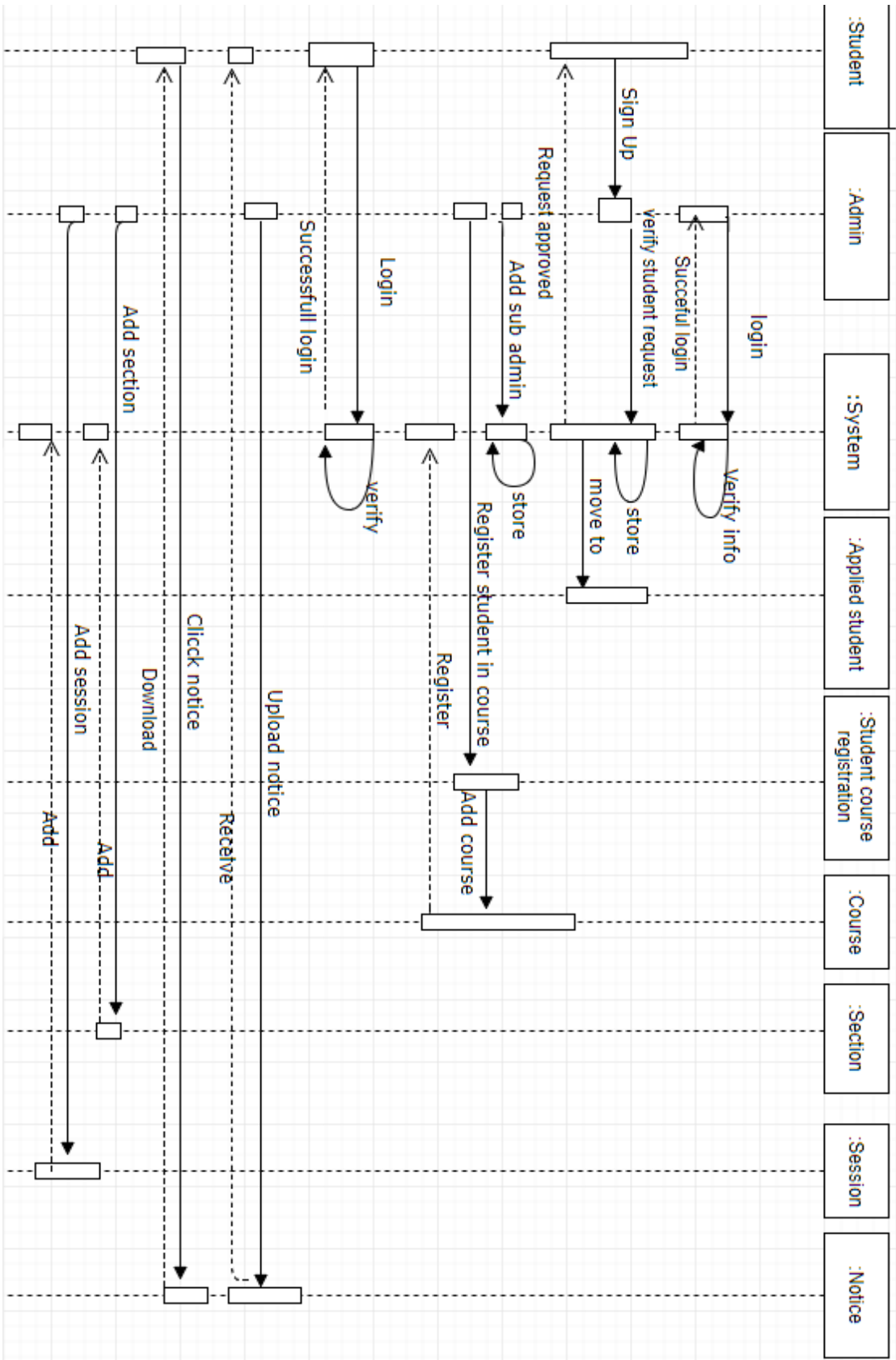


Figure 4.3.3 Sequence Diagram

4.3.4 Entity Relationship Diagram

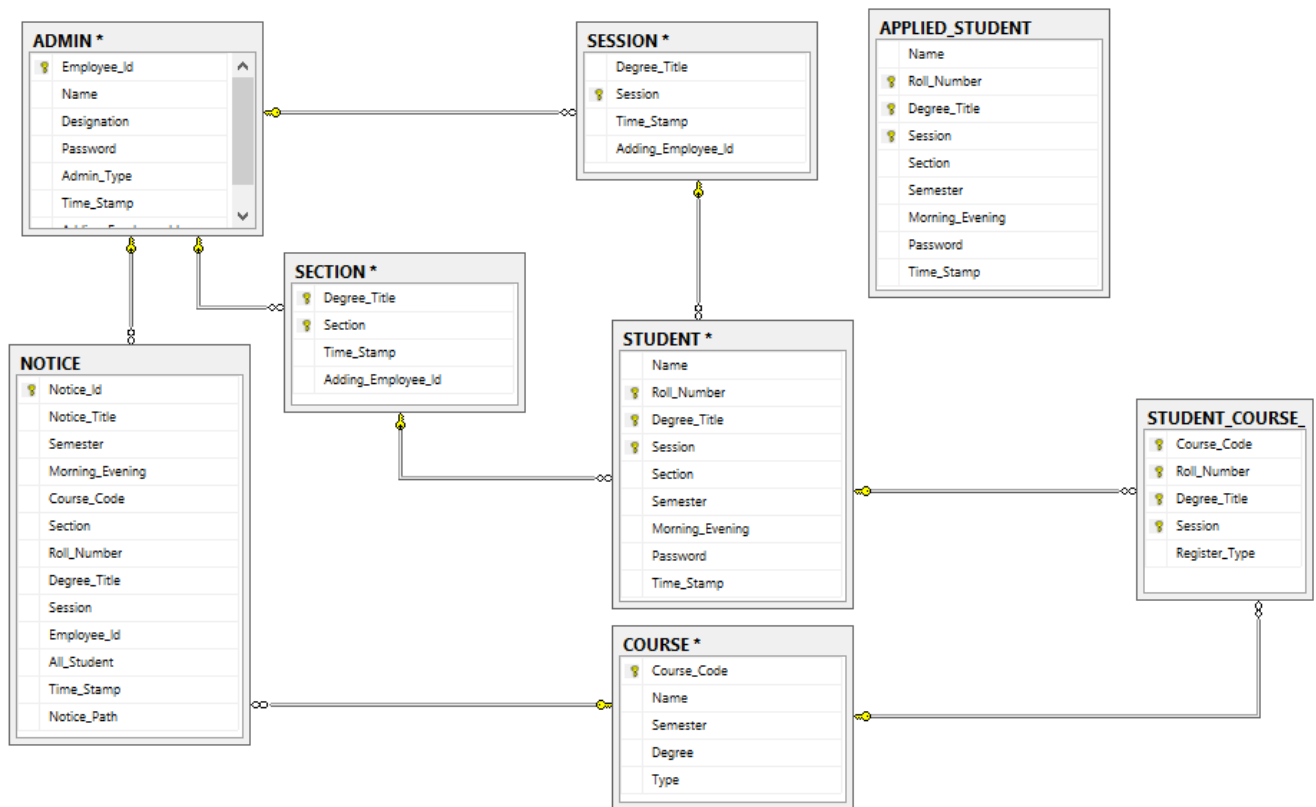


Figure 4.3.4 Entity Relationship Diagram

4.3.5 Class Diagram

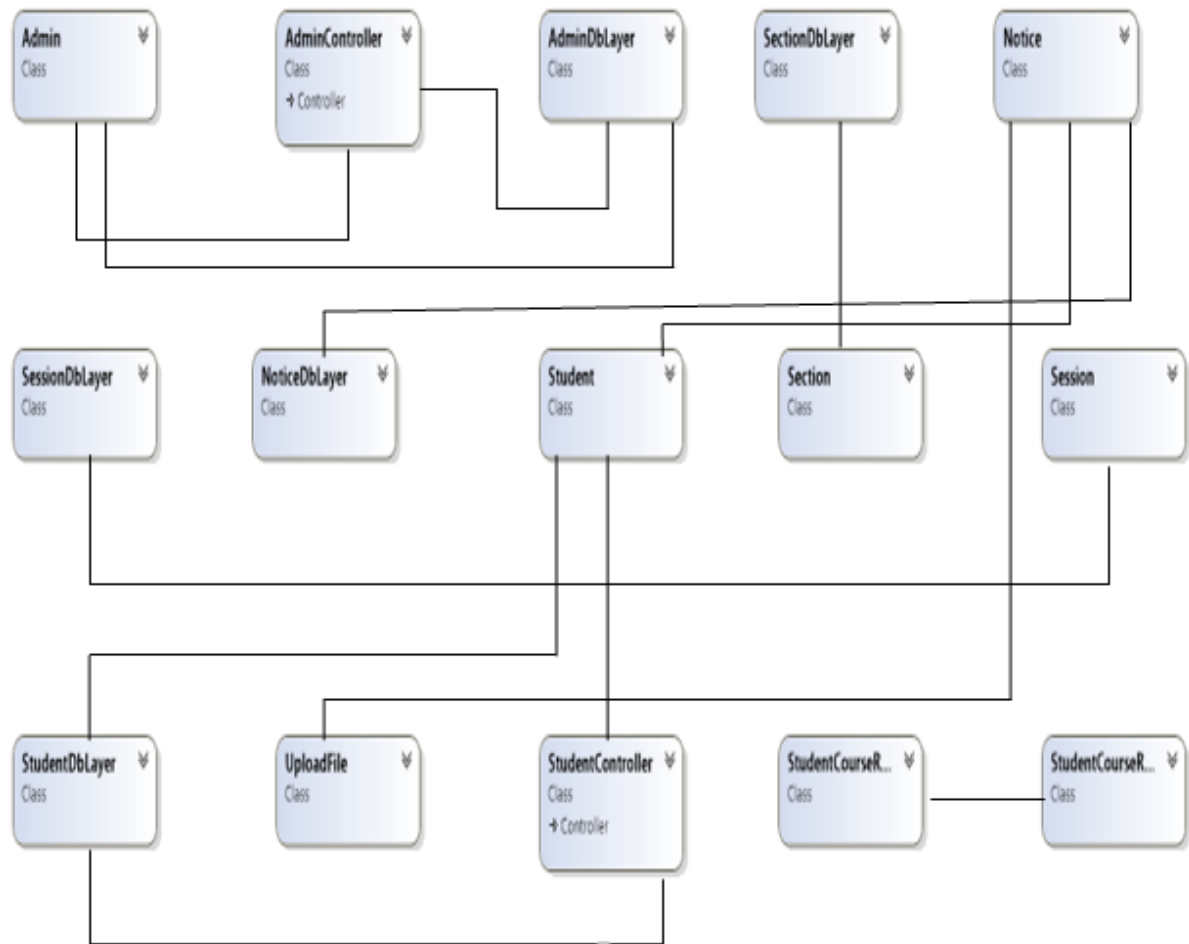


Figure 4.3.5 Class Diagram

Chapter 5 Implementation

5.1 Development Stages

Activity	1	2	3	4	5	6	7	8	9	10	11
Project proposal											
Feasibility study											
Req. analysis											
System design											
System coding											
testing											
Implementation											
Documentation											

Figure 5.1 Development stage

5.1.1 Feasibility Study

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the common public. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation

- Technical Feasibility
- Operation Feasibility
- Economic Feasibility

5.1.1.1 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 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 of users?

- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and 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 desktop based user interface. Thus it provides an easy access to the users. The database’s purpose is to create, establish and maintain a workflow among various objects in order to facilitate all concerned users in their various capacities or roles. Permission to the student would be granted based on the roles specified means that student can’t receive notice board notification before admin validation. Therefore, it provides the technical guarantee of accuracy, reliability. The work for the project is done with upload notification by admin and download/view notification by student.

5.1.1.2 Operational Feasibility

That will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following

- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

5.1.1.3 Economic Feasibility

A system can be developed technically and that will be used if installed must still be a good investment for the developers. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed

using the existing resources and technologies, there is nominal expenditure and economic feasibility for certain.

5.1.2 Information Collection

Once both the analyst and the client are convinced that a solution is possible and desirable, the analyst needs to collect information about the problem to understand the problem and the current solution better, and work out which areas need improving. There are several ways an analyst can gather information

Interviews: A small number of important people can be asked about the problems. Interviews are generally slow but it allows the analyst to ask more questions if something unplanned arises.

Questionnaires: These allow the analyst to ask opinions of a large number of people quickly. However, questionnaires are very restrictive because the questions must be planned in advance.

Meetings: Meetings strike a balance between the speed in questionnaires, and the flexibility of interviews. They allow a large number of people to have their opinions taken quickly. However, a few powerful individuals can control the meeting, and make the result biased.

Observation: Observation is possibly the most accurate form of information collection, but it is also the most time consuming. The analyst simply observes use of the current system, making notes about what the current solution does well, and what needs improving.

Documentation: The analyst collects all available documentation and studies it carefully. This will show how the system collects, processes, and presents data and information.

5.1.3 Lifecycle Description

➤ Requirements Analysis

- Our functional as well as non-functional requirements were very clear from the start. So this lifecycle suited our purpose the best.

➤ Design

- We were able to assist our project on the pre-design diagrams that were made purely on the basis of requirements we were provided with

➤ **Development:**

- We chose the modular approach to complete our project. It was fast, easy, efficient and effective in all manners whatsoever.

➤ **Testing:**

- Our project yielded our desired results according to the requirements we were provided with. Our project passed the unit testing and the accuracy was between 80-90%.

➤ **Maintenance:**

- We hope for the future that we will enhance our project to make it more efficient and error free (if there are still any).

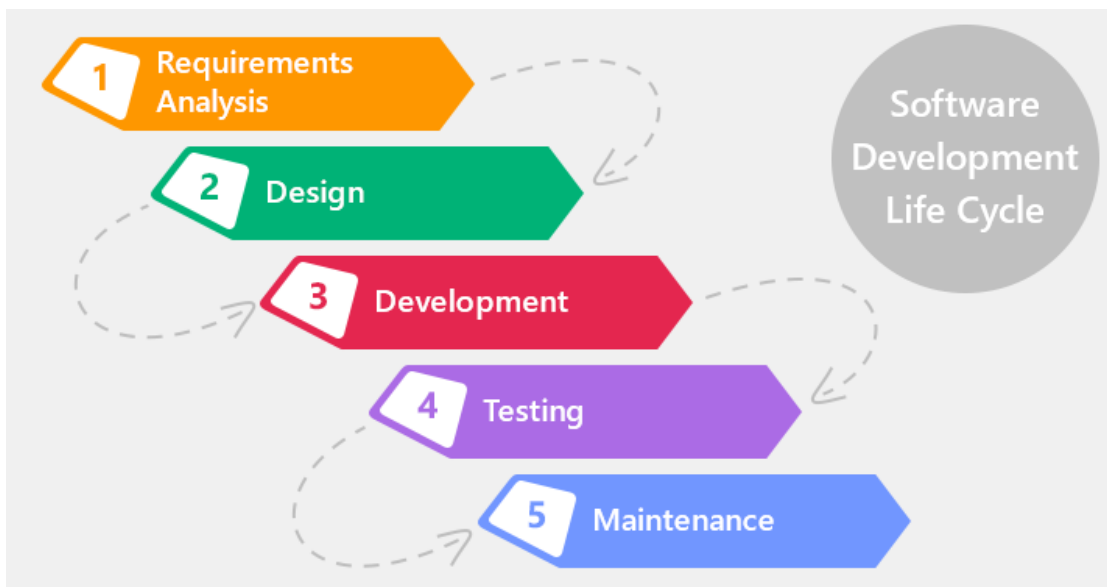


Figure 5.1.3 Software Development Life Cycle

5.2 User interface

5.2.1 Prototype

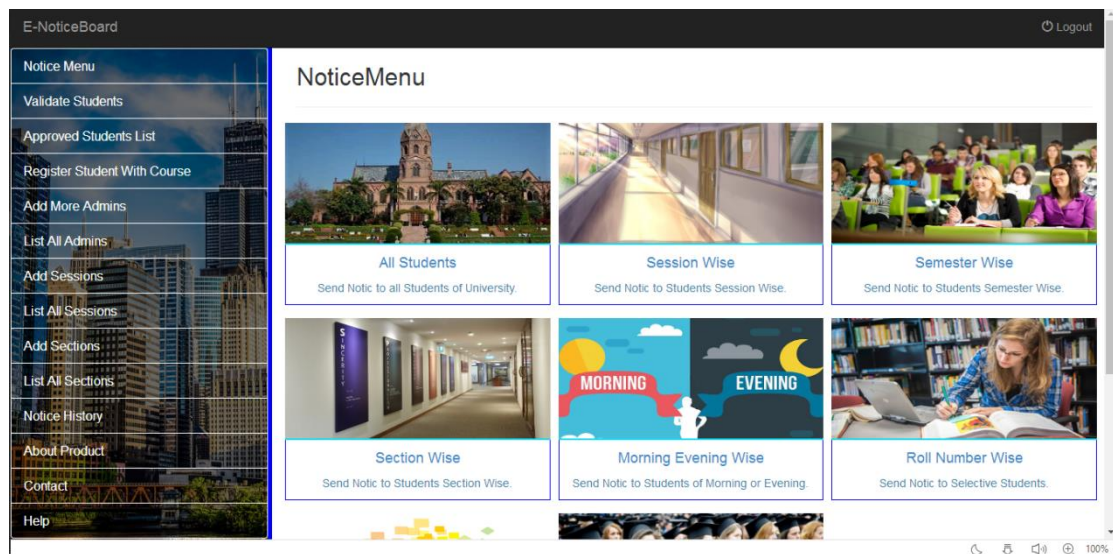


Figure 5.2.1.1 Notice Menu

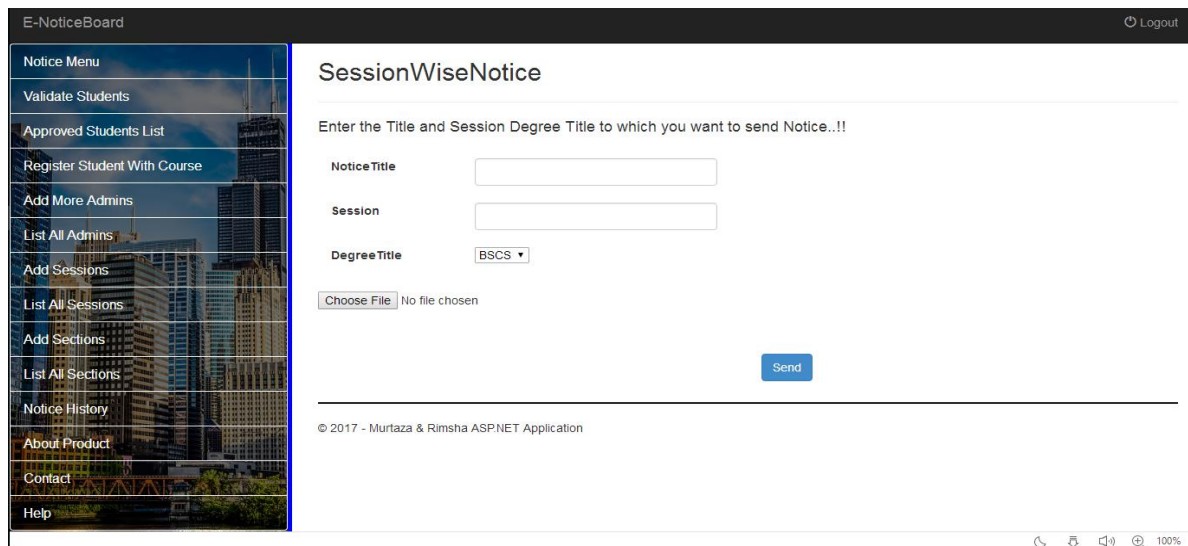


Figure 5.2.1.2 SessionWiseNotice

The screenshot shows the 'SemesterWiseNotice' form within the 'E-NoticeBoard' application. On the left is a vertical navigation menu with options: Notice Menu, Validate Students, Approved Students List, Register Student With Course, Add More Admins, List All Admins, Add Sessions, List All Sessions, Add Sections, List All Sections, Notice History, About Product, Contact, and Help. The main content area has a header 'SemesterWiseNotice' and a prompt 'Enter the Title and Semester to which you want to send the Notice..!!'. Below this are three input fields: 'NoticeTitle', 'Semester', and 'DegreeTitle' (which is a dropdown menu currently showing 'BSCS'). There is a 'Choose File' button with the text 'No file chosen' next to it, and a blue 'Send' button. At the bottom of the main area, it says '© 2017 - Murtaza & Rimsha ASP.NET Application'. The footer of the page contains navigation icons and a '100%' zoom indicator.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

SemesterWiseNotice

Enter the Title and Semester to which you want to send the Notice..!!

NoticeTitle

Semester

DegreeTitle BSCS ▼

Choose File No file chosen Send

© 2017 - Murtaza & Rimsha ASP.NET Application

100%

Figure 5.2.1.3 SemesterWiseNotice

The screenshot shows the 'SectionWiseNotice' form within the 'E-NoticeBoard' application. The layout is identical to the previous figure, with the same navigation menu on the left. The main content area has a header 'SectionWiseNotice' and a prompt 'Enter the Title, Section and Semester to which you want to send the Notice..!!'. Below this are four input fields: 'NoticeTitle', 'Section', 'Semester', and 'DegreeTitle' (a dropdown menu showing 'BSCS'). There is a 'Choose File' button with the text 'No file chosen' next to it, and a blue 'Send' button. At the bottom of the main area, it says '© 2017 - Murtaza & Rimsha ASP.NET Application'. The footer of the page contains navigation icons and a '100%' zoom indicator.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

SectionWiseNotice

Enter the Title, Section and Semester to which you want to send the Notice..!!

NoticeTitle

Section

Semester

DegreeTitle BSCS ▼

Choose File No file chosen Send

© 2017 - Murtaza & Rimsha ASP.NET Application

100%

Figure 5.2.1.4 SectionWiseNotice

The screenshot displays the E-NoticeBoard application interface. On the left is a vertical sidebar menu with the following items: Notice Menu, Validate Students, Approved Students List, Register Student With Course, Add More Admins, List All Admins, Add Sessions, List All Sessions, Add Sections, List All Sections, Notice History, About Product, Contact, and Help. The main content area is titled "MorningEveningWiseNotice". Below the title, it says "Enter the Title and Semester to which you want to send the Notice...!!". There are two input fields: "Notice Title" and "MorningEvening". Below these is a file upload section with a "Choose File" button and the text "No file chosen". A blue "Send" button is located to the right of the file upload section. At the bottom of the main area, there is a copyright notice: "© 2017 - Murtaza & Rimsha ASP.NET Application". The bottom of the page features a footer with icons for a clock, a document, a speaker, a magnifying glass, and a 100% zoom indicator.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

MorningEveningWiseNotice

Enter the Title and Semester to which you want to send the Notice...!!

Notice Title

MorningEvening

Choose File No file chosen Send

© 2017 - Murtaza & Rimsha ASP.NET Application

🕒 📄 🔊 🔍 100%

Figure 5.2.1.5 MorningEveningWiseNotice

The screenshot displays the E-NoticeBoard application interface. On the left is a vertical sidebar menu with the following items: Notice Menu, Validate Students, Approved Students List, Register Student With Course, Add More Admins, List All Admins, Add Sessions, List All Sessions, Add Sections, List All Sections, Notice History, About Product, Contact, and Help. The main content area is titled "RollNumberWiseNotice". Below the title, it says "Enter the Title and Full Roll Number to which you want to send the Notice...!!". There are four input fields: "Notice Title", "RollNumber", "DegreeTitle" (which is a dropdown menu currently showing "BSCS"), and "Session". Below these is a file upload section with a "Choose File" button and the text "No file chosen". A blue "Send" button is located to the right of the file upload section. At the bottom of the main area, there is a copyright notice: "© 2017 - Murtaza & Rimsha ASP.NET Application". The bottom of the page features a footer with icons for a clock, a document, a speaker, a magnifying glass, and a 100% zoom indicator.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

RollNumberWiseNotice

Enter the Title and Full Roll Number to which you want to send the Notice...!!

Notice Title

RollNumber

DegreeTitle BSCS ▼

Session

Choose File No file chosen Send

© 2017 - Murtaza & Rimsha ASP.NET Application

🕒 📄 🔊 🔍 100%

Figure 5.2.1.6 RollNumberWiseNotice

The screenshot shows the 'CourseWiseNotice' form within the 'E-NoticeBoard' application. On the left is a vertical sidebar menu with a cityscape background, containing links: Notice Menu, Validate Students, Approved Students List, Register Student With Course, Add More Admins, List All Admins, Add Sessions, List All Sessions, Add Sections, List All Sections, Notice History, About Product, Contact, and Help. The main content area has a dark header with 'E-NoticeBoard' and a 'Logout' link. Below the header, the title 'CourseWiseNotice' is displayed. A prompt reads 'Enter the Title and Course Code to which you want to send the Notice...!!'. There are two input fields: 'NoticeTitle' and 'CourseCode'. Below them is a file upload section with a 'Choose File' button, the text 'No file chosen', and a 'Send' button. At the bottom, a copyright notice states '© 2017 - Murtaza & Rimsha ASP.NET Application'. The footer includes navigation icons and a '100%' zoom indicator.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

CourseWiseNotice

Enter the Title and Course Code to which you want to send the Notice...!!

NoticeTitle

CourseCode

Choose File No file chosen

© 2017 - Murtaza & Rimsha ASP.NET Application

100%

Figure 5.2.1.7 CourseWiseNotice

The screenshot shows the 'DegreeWiseNotice' form within the 'E-NoticeBoard' application. The layout is identical to the previous figure, with the same sidebar menu and header. The title is 'DegreeWiseNotice'. The prompt reads 'Enter the Title and Degree Title to which you want to send the Notice...!!'. The input fields are 'NoticeTitle' and 'DegreeTitle', which has a dropdown menu currently showing 'BSCS'. The file upload section and 'Send' button are also present. The copyright notice and footer elements are the same as in Figure 5.2.1.7.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

DegreeWiseNotice

Enter the Title and Degree Title to which you want to send the Notice...!!

NoticeTitle

DegreeTitle

Choose File No file chosen

© 2017 - Murtaza & Rimsha ASP.NET Application

100%

Figure 5.2.1.8 DegreeWiseNotice

E-NoticeBoard Logout

Notice Menu

Validate Students

Approved Students List

Register Student With Course

Add More Admins

List All Admins

Add Sessions

List All Sessions

Add Sections

List All Sections

Notice History

About Product

Contact

Help

AppliedStudentList

[Back](#)

Name	RollNumber	DegreeTitle	Session	Section	Semester	MorningEvening	TimeStamp
Ali Murtaza	85	BSCS	2013-2017	A	8	E	Oct 08 2017/Sun/03:59 Validate Delete

© 2017 - Murtaza & Rimsha ASP.NET Application

Figure 5.2.1.9 AppliedStudentList

E-NoticeBoard Logout

Notice Menu

Validate Students

Approved Students List

Register Student With Course

Add More Admins

List All Admins

Add Sessions

List All Sessions

Add Sections

List All Sections

Notice History

About Product

Contact

Help

ApprovedStudentsList

[Back](#)

Name	RollNumber	DegreeTitle	Session	Section	Semester	MorningEvening	TimeStamp
Rabiya Jamil	31	BSCS	2013-2017	A	8	E	Oct 08 2017/Sun/04:00 Delete
Muneeb Java	32	MSCS	2013-2017	A	6	M	Oct 08 2017/Sun/04:00 Delete
Rimsha Saleem	49	BSCS	2013-2017	A	8	M	Oct 08 2017/Sun/11:55 Delete

© 2017 - Murtaza & Rimsha ASP.NET Application

Figure 5.2.1.10 ApprovedStudentList

The screenshot shows the 'StudentCourseRegister' page of the 'E-NoticeBoard' application. On the left is a vertical sidebar menu with options: Notice Menu, Validate Students, Approved Students List, Register Student With Course, Add More Admins, List All Admins, Add Sessions, List All Sessions, Add Sections, List All Sections, Notice History, About Product, Contact, and Help. The main content area is titled 'StudentCourseRegister' with a subtitle 'StudentCourseRegistration'. It contains a registration form with the following fields: 'CourseCode' (text input), 'RollNumber' (text input), 'DegreeTitle' (dropdown menu with 'BSCS' selected), 'Session' (text input), and 'RegisterType' (dropdown menu with 'Regular' selected). A 'Register' button is located below the 'RegisterType' field. A 'Back' link is positioned below the form. At the bottom of the main area, there is a copyright notice: '© 2017 - Murtaza & Rimsha ASP.NET Application'. The top of the page features a dark header with 'E-NoticeBoard' on the left and a 'Logout' button on the right. The bottom of the page includes a status bar with the URL 'localhost:24880/Admin/AdminMainMenu' and browser controls.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

StudentCourseRegister

StudentCourseRegistration

CourseCode

RollNumber

DegreeTitle BSCS ▼

Session

RegisterType Regular ▼

[Back](#)

© 2017 - Murtaza & Rimsha ASP.NET Application

localhost:24880/Admin/AdminMainMenu

Figure 5.2.1.11 StudentCourseRegister

The screenshot shows the 'AddAdmin' page of the 'E-NoticeBoard' application. The sidebar menu is identical to the previous page, with 'Register Student With Course' highlighted. The main content area is titled 'AddAdmin' with a subtitle 'Admin'. It contains a form for adding a new admin with the following fields: 'EmployeeId' (text input), 'Name' (text input), 'Designation' (text input), 'AdminType' (dropdown menu with 'Main' selected), and 'Password' (text input). An 'Add' button is located below the 'Password' field. A 'Back' link is positioned below the form. At the bottom of the main area, there is a copyright notice: '© 2017 - Murtaza & Rimsha ASP.NET Application'. The top of the page features a dark header with 'E-NoticeBoard' on the left and a 'Logout' button on the right. The bottom of the page includes a status bar with the URL 'localhost:24880/Admin/AdminMainMenu' and browser controls.

E-NoticeBoard Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

AddAdmin

Admin

EmployeeId

Name

Designation

AdminType Main ▼

Password

© 2017 - Murtaza & Rimsha ASP.NET Application

localhost:24880/Admin/AdminMainMenu

Figure 5.2.1.12 AddAdmin

E-NoticeBoard
Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

AddSession

Session

DegreeTitle BSCS

Session

Create

© 2017 - Murtaza & Rimsha ASP.NET Application

100%

Figure 5.2.1.13 AddSession

E-NoticeBoard
Logout

Notice Menu
Validate Students
Approved Students List
Register Student With Course
Add More Admins
List All Admins
Add Sessions
List All Sessions
Add Sections
List All Sections
Notice History
About Product
Contact
Help

ListAllSession

[Create New](#)

DegreeTitle	Session	TimeStamp	AddingEmployeeId	
BSCS	2018-2019	Oct 08 2017/Sun/04:07	Test	Delete
BSCS	2019-2021	Oct 08 2017/Sun/04:07	Test	Delete
MSCS	2013-2015	Oct 08 2017/Sun/04:08	Test	Delete
MSCS	2014-2016	Oct 08 2017/Sun/04:08	Test	Delete

© 2017 - Murtaza & Rimsha ASP.NET Application

100%

Figure 5.2.1.14 ListAllSession

E-NoticeBoard Logout

AddSection

Section

DegreeTitle BSCS

Section

Create

© 2017 - Murtaza & Rimsha ASP.NET Application

Figure 5.2.1.15 AddSection

E-NoticeBoard Logout

ListAllSection

DegreeTitle	Section	TimeStamp	AddingEmployeeId	
BSCS	A	Oct 08 2017/Sun/04:08	Test	Delete
BSCS	B	Oct 08 2017/Sun/04:08	Test	Delete
BSCS	C	Oct 08 2017/Sun/04:08	Test	Delete
MSCS	A	Oct 08 2017/Sun/04:08	Test	Delete
MSCS	B	Oct 08 2017/Sun/04:08	Test	Delete
MSCS	C	Oct 08 2017/Sun/04:08	Test	Delete
BSCS	E1	Oct 08 2017/Sun/04:09	Test	Delete

© 2017 - Murtaza & Rimsha ASP.NET Application

Figure 5.2.1.16 ListAllSection

E-NoticeBoard

Logout

Notice Menu

Validate Students

Approved Students List

Register Student With Course

Add More Admins

List All Admins

Add Sessions

List All Sessions

Add Sections

List All Sections

Notice History

About Product

Contact

Help

ListAllNotice

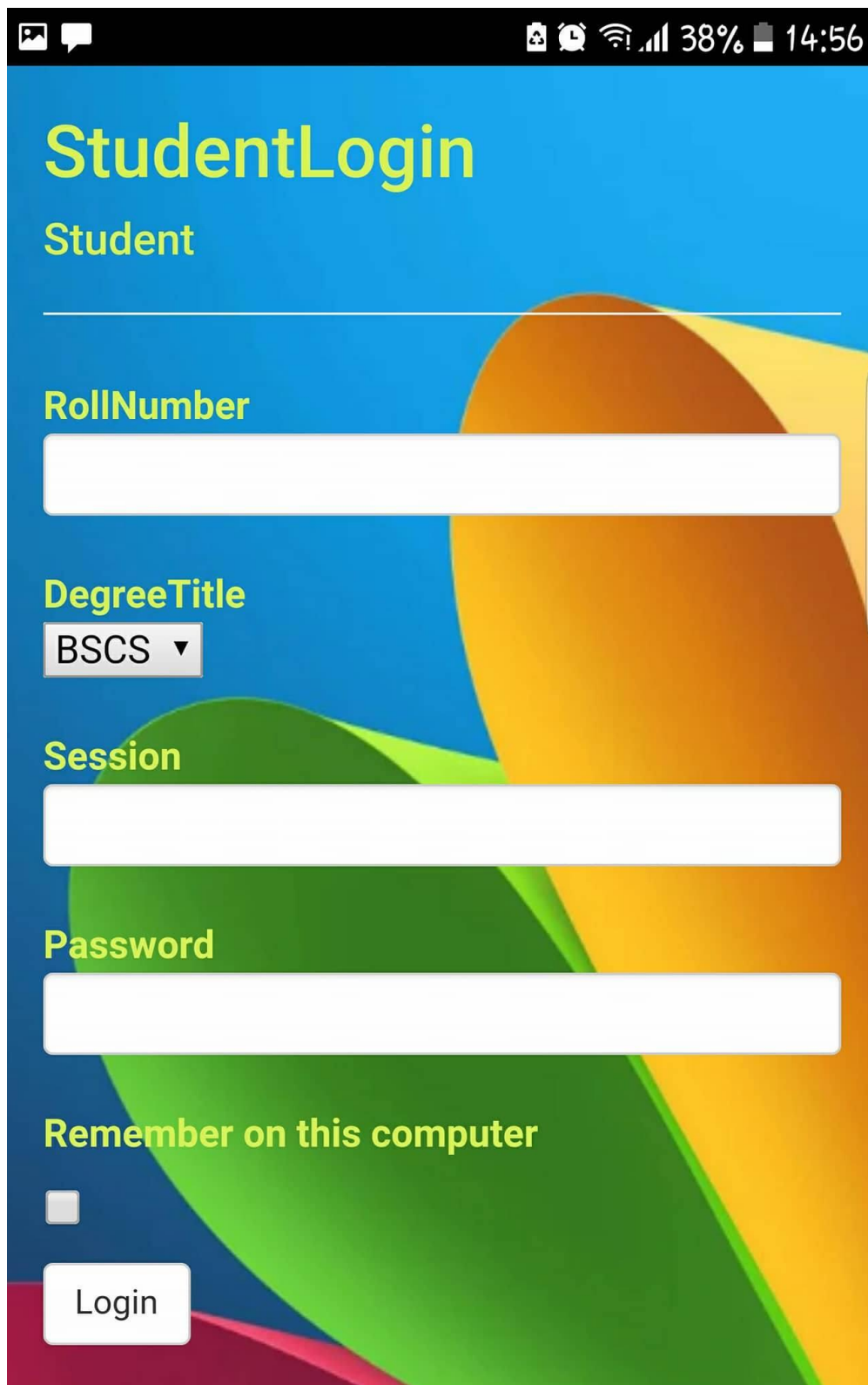
NoticeTitle	EmployeeId	TimeStamp	RollNo	Degree	Session	Section	Semester	Mor/Eve	CourseCode	AllStudent
Debug	Test	Oct 08 2017/Sun/04:22	0	None	None	None	0	None	None	Yes
abc	Test	Oct 08 2017/Sun/06:36	0	None	None	None	0	None	None	Yes
Holiday Notice	Test	Oct 08 2017/Sun/09:13	0	None	None	None	0	None	None	Yes
Fee	Test	Oct 08 2017/Sun/09:14	0	BSCS	2013-2017	None	0	None	None	None
fdj	Exyz	Oct 08 2017/Sun/09:42	0	BSCS	2013-2017	None	0	None	None	None
fdj	Exyz	Oct 08 2017/Sun/09:45	0	BSCS	2013-2017	None	0	None	None	None
fdj	Exyz	Oct 08 2017/Sun/09:45	0	BSCS	2013-2017	None	0	None	None	None

© 2017 - Murtaza & Rimsha ASP.NET Application

localhost:24580/Admin/AdminMainMenu

100%

Figure 5.2.1.17 ListAllNotice



A mobile application login screen titled "StudentLogin" with a subtitle "Student". The background features a blue sky and colorful abstract hills in orange, yellow, and green. The form includes a horizontal line, a "RollNumber" text input, a "DegreeTitle" dropdown menu with "BSCS" selected, a "Session" text input, a "Password" text input, a "Remember on this computer" checkbox, and a "Login" button. The top status bar shows icons for camera, messages, alarm, Wi-Fi, cellular signal, 38% battery, and the time 14:56.

StudentLogin
Student

RollNumber

DegreeTitle

BSCS ▼

Session

Password

Remember on this computer

☐

Login

Figure 5.2.1.18 Student Login Menu



A mobile application interface for student sign-up. The background features a vibrant, abstract design with overlapping circles in shades of blue, orange, green, and purple. The form is composed of several white input fields and a dropdown menu, each preceded by a label in a bold, yellow font. The top of the screen shows a black status bar with various icons and the time 14:56.

StudentSignUp

Student

Name

RollNumber

DegreeTitle

BSCS ▾

Session

Section

Semester

MorningEvening

Figure 5.2.1.19 Student Sign Up Menu

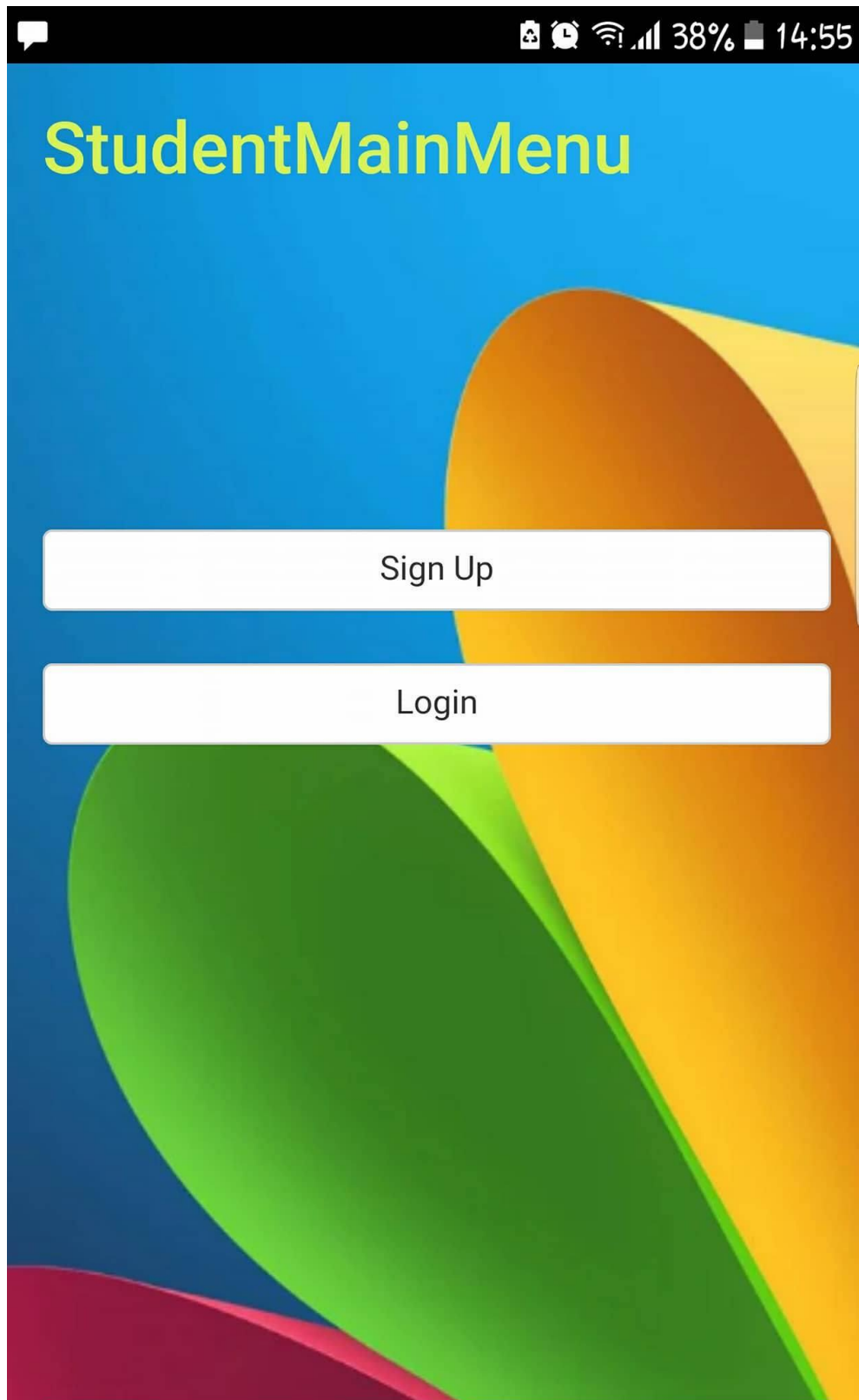


Figure 5.2.1.20 Student Main Menu

Chapter 6 Evaluation

6.1 Unit Testing

The first stage of the evaluation was the self-evaluation and technical testing which has been carried out without the help of external users, and this part included validating the server and client side code, testing the system with different kinds of inputs, debugging, and other design issues. After that, the system was tested working from different locations, platforms, and browsers. In this stage the system was valid. All the functional units or methods of the project were test individually by passing different values to the methods and then checking the values by using debugger tools for their correctness. In our project Unit testing was performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time for each module in Bottom-up testing short programs were used to execute the module and provided the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

6.1.1 Sign up Android Application

Field	Value	Valid/Invalid	Result
Roll number	Null	Invalid	This is a required field
Roll number	Wrong format	Invalid	Roll No. not in correct format
Roll number	031	Valid	Accepted
Degree title	BSCS/MSCS/PhD	Valid	Accepted
Session	Null	Invalid	This is a required field
Session	17-14	Invalid	Session is invalid
Session	2013-2017	Valid	Accepted
Password	12345	Valid	Create Account
Password	ABC123	Invalid	Incorrect Password

6.1.2 Verify Student Request

Field	Value	Valid/Invalid	Result
Verify Button	Click	Valid	Verified

6.1.3 Login Android Application

Field	Value	Valid/Invalid	Result
Name	Rabiya	Valid	Accepted
Roll number	Null	Invalid	This is a required field
Roll number	Wrong format	Invalid	Roll No. not in correct format
Roll number	031	Valid	Accepted
Degree title	BSCS/MSCS/PhD	Valid	Accepted
Session	Null	Invalid	This is a required field
Session	17-14	Invalid	Session is invalid
Session	2013-2017	Valid	Accepted
Semester	9	Invalid	Semester must between 1-8
Semester	8	Valid	Accepted
Semester	Null	Invalid	This is a required field
Morning/Evening	E/M	Valid	Accepted
Morning/Evening	Afternoon	Invalid	Type M or E for
Password	12345	Valid	Create Account

6.1.4 Download/view File

Field	Value	Valid/Invalid	Result
Download/view Link	Click	Valid	Download/viewing

6.1.5 Logout Android Application

Field	Value	Valid/Invalid	Result
Logout Button	Click	Valid	Logout

6.1.6 Delete Student Request

Field	value	Valid/Invalid	Result
Delete Button	Click	Valid	Student Deleted

6.1.7 Log in Website

Field	Value	Valid/Invalid	Result
Employee ID	Null	Invalid	This is a required field
Employee ID	Test	valid	Accepted
Employee ID	Abcd	Invalid	Login data is incorrect
Employee ID	Test123456	Invalid	Length must be 4
Password	12345	Valid	Create Account
Password	ABC123	Invalid	Incorrect Password

6.1.8 Add Admin

Field	Value	Valid/Invalid	Result
Employee Id	Null	Invalid	This is a required field
Employee Id	Test	valid	Accepted
Employee Id	Abcd	Invalid	Login data is incorrect
Employee Id	Test123456	Invalid	Length must be 4
Name	Komal	Valid	Accepted
Name	Null	Invalid	This is a required field
Designation	Lecturer	Valid	Accepted
Admin Type	Main/Sub	Valid	Accepted
Password	12345	Valid	Create Account
Password	ABC123	Invalid	Incorrect Password

6.1.9 Add Session

Field	Value	Valid/Invalid	Result
Session Name	Null	Invalid	This is a required field
Session Name	2013-2017	valid	Accepted
Section Name	Abcd	Invalid	Section data is incorrect
Degree Title	BSCS/MSCS/PhD	Invalid	Length must be 4

6.1.10 Registration Student with Course

Field	Value	Valid/Invalid	Result
Roll number	Null	Invalid	This is a required field
Roll number	Wrong format	Invalid	Roll No. not in correct format
Roll number	031	Valid	Accepted
Degree title	BSCS/MSCS/PhD	Valid	Accepted
Session	Null	Invalid	This is a required field
Session	17-14	Invalid	Session is invalid
Session	2013-2017	Valid	Accepted
Course code	CS-4215	Valid	Accepted
Course code	C1234	Invalid	Invalid course code
Course code	C1234	Invalid	Invalid course code

6.1.11 Add Section

Field	Value	Valid/Invalid	Result
Section Name	Null	Invalid	This is a required field
Section Name	A/B/E1	valid	Accepted
Section Name	Abcd	Invalid	Section data is incorrect
Degree Title	BSCS/MSCS/PhD	Invalid	Length must be 4

6.1.12 Send Notice

Field	Value	Valid/Invalid	Result
Notice Menu	Select Category	Valid	Move to Sending Page
Fill Required Fields	NULL	Invalid	This is a required field
Fill Required Fields	Fill Properly	Valid	Accepted
Choose File	Null	Invalid	Please Choose a File
Choose File	Wrong Extension	Invalid	File Extension is Invalid
Choose File	Correct Extension	Valid	Accepted
Send Button	Click	Valid	Notice Sent

6.1.13 Logout Website

Field	Value	Valid/Invalid	Result
Logout Button	Click	Valid	Logout

6.1.14 Delete Approve Students

Field	Value	Valid/Invalid	Result
Delete Button	Click	Valid	Student Deleted

6.2 Functional Testing

Functional testing is a quality assurance (QA) process and a type of black-box testing that bases its test cases on the specifications of the software component under test.

6.2.1 Sign up Android Application

Pre-Condition	Students should have application and has network connection for sign up.
Step Taken	<ul style="list-style-type: none"> • Student selects name textbox and fills it by writing name. • Student selects Roll Number textbox and fills it by writing roll number. • Student selects 'Degree Title' drop down option. • Student selects session textbox and fills it by writing correct session. • Student selects semester textbox and fills it by writing correct semester. • Student select Morning / Evening drop down option. • Student Write Password • After filling form , student press sign up button • filling form , student press sign In button
Expected Result	The system allows the user to sign up.
Result	Pass

6.2.2 Login in Android Application

Pre-Condition	Students should have network connection and account in android application.
Step Taken	<ul style="list-style-type: none"> • Student select roll number textbox and write name. • Student selects 'Degree Title' drop down option. • Student selects session textbox and fills it by writing correct session. • Student Write Password • After filling form , student press sign In button
Expected Result	The system allows the user to login.
Result:	Pass

6.2.3 Download/view File

Pre-Condition	Students should have network connection and account in android application.
Step Taken	<ul style="list-style-type: none"> • Student goes to the notice board list. • Select specific file which has notice title.
Expected Result	The system allows the user to down load file.
Result	Pass

6.2.4 Logout Android Application

Pre-Condition	Students should have android application, has network connection and already login the account.
Step Taken	<ul style="list-style-type: none"> • Admin click logout button • Account will be logout
Expected Result	The system allows the user to Log out.
Result:	Pass

6.2.5 Verify Student Request

Pre-Condition	Students should have web application and has network connection for sign up and there must be appending request.
Step Taken	<ul style="list-style-type: none"> • Admin login the web site • In main menu, Click validate student button • Click validate link
Expected Result:	The system allows the admin to verify student.
Result:	Pass

6.2.6 Delete Request

Pre-Condition	Admin should have network connection and log in the web site.
Step Taken	<ul style="list-style-type: none"> • Admin login account of web site • Click validate student button. • Click delete link.
Expected Result:	The system allows the admin to delete request.
Result:	Pass

6.2.7 Delete Approved Student

Pre-Condition	Admin should has network connection and log in the web application.
Step Taken	<ul style="list-style-type: none"> • Admin login the account of wed application. • Click approves student button. • Click delete approved student.
Expected Result	The system allows the admin to delete approved student.
Result	Pass

6.2.8 Login web site

Pre-Condition	Admin should have network connection and have web application.
Step Taken	<ul style="list-style-type: none"> • Admin go to login menu. • Enter ID. • Enter password. • Click log in Button..
Expected Result	The system allows the admin to login.
Result	Pass

6.2.9 Logout web site

Pre-Condition	Admin should have web application, has network connection and already login the account.
Step Taken	<ul style="list-style-type: none"> • Admin click logout button. • Account will be logout.
Expected Result	The system allows the admin to logout.
Result	Pass

6.2.10 Register Student with Course

Pre-Condition	Admin should have web application, has network connection and already login the account and have approved student list.
Step Taken	<ul style="list-style-type: none"> • Admin goes to registration panel. • Enter course code in which student is enrolled. • Enter student roll number. • Select degree title from drop down option. • Enter session of student. • Select register type i.e. regular student, repeater etc. • Click register button..
Expected Result	The system allows the admin to register student with course.
Result	Pass

6.2.11 Upload File

Pre-Condition	Admin should have web application, has network connection and have register student.
Step Taken	<ul style="list-style-type: none"> • Admin go to notice category selection panel. • Select categories. • Go to upload panel. • Input notice description. • Click send.
Expected Result	The system allows the admin to upload file.
Result	Pass

6.2.12 Add Admin

Pre-Condition	Admin should have web application, has network connection.
Step Taken	<ul style="list-style-type: none"> • Admin go to add admin panel. • Fill form by entering employee ID, Name, Designation, admin type, password. • Click add
Expected Result:	The system allows the main admin to add sub admin.
Result	Pass

6.2.13 Add Session

Pre-Condition	Admin should have web application, has network connection.
Step Taken	<ul style="list-style-type: none"> • Admin go Add session panel • Select degree title. • Enter session. • Click to create button.
Expected Result	The system allows the admin to add session.
Result	Pass

6.2.14 Add Section

Pre-Condition	Admin should have web application, has network connection.
Step Taken	<ul style="list-style-type: none">• Admin go Add section panel• Select degree title.• Enter section.• Click to create button.
Expected Result	The system allows the admin to add section.
Result	Pass

Chapter 7 Conclusion and Future Work

7.1 Conclusion

As a result of the Web revolution, Universities are going global by creating a Web presence and users are going online using mobile phones. Therefore we have developed an online university notice board system which give the notification to student of department as subject wise, roll number wise, session wise, subject wise, section wise, semester wise and student receive notices on their mobile phone and can download/view it easily, Admin upload the notices on web portal after registration of students.

7.2 Future Work

The system is newly developed system it has many stages to progress. There are many improvement and future ideas that can be applied as system is so much flexible so in future it can increase easily and new modules can be added easily. In this section we will discuss some of them.

On implementation side: In a real life application, where there is a heavy traffic on the server, it may be recommended to use other database systems like SQL Server Enterprise Editions, as it works seamlessly with .NET platform, and it is more scalable. We used SQL Server Express here for demonstrating purposes and for the reason of availability in our development environment.

University Level Work: This project is now developed for Department. In future we will expand this project and make it able for university level so that every student of university will get the notification department wise.

Bibliography

- Crockford, D. (2010). *Javascript The Good Parts*.
- Microsoft. (2017). *msdn.Microsoft*. Retrieved from Microsoft:
<https://msdn.microsoft.com>
- Pressman, R. S. (2009). *Software Engineering*.
- *PrimaryObject*. (2017). Retrieved from
<http://www.primaryobjects.com/2012/12/07/mvc-forms-authentication-and-storing-data-in-the-cookie/>
- Schildt, H. (2012). *Asp .Net Complete Reference*.
- StackOverflow. (2017). *Stackoverflow*. Retrieved from
<https://stackoverflow.com/questions/19128507/using-cookie-in-asp-net-mvc-4>
- StackOverFlow. (2017). *StackOverFlow*. Retrieved from
<https://stackoverflow.com/questions/304617/html-helper-for-input-type-file>
- Suehring, S. (2017). *SQL Sever*. John Wiley & Sons.
- *YouTube*. (2017). Retrieved from YouTube:
<https://www.youtube.com/watch?v=pe6JhgN7BWM>

Appendix

How to run the system?

To run the website, the required software is already present on the system, the server runs Microsoft .NET Framework 4.5.2, under the Web server IIS Express and Microsoft SQL server. To test it you have to register as an Admin or student. If you are a student, you must login from an android device having Online Notice Board application in it and if you are university employee, you must login from website using PC. You can use the following ready-to-use login credentials, which has been created for testing purpose:

Admin

- **Employee ID:** Test
- **Password:** 12345

Student

- **Roll number:** 031
- **Degree title:** BSCS
- **Session:** 2013-2017
- **Password:**12345

Using through Local Machine

To run the website on your local machine through a browser e.g. UC-Browser or Chrome etc. you need an operating system such as Windows running IIS Express with Microsoft .NET Framework 4.5.2, if it is not available on your local machine you can download/view it from the Microsoft ASP.NET Official site at: <http://asp.net/downloads/>. Make sure the Web server (IIS) is running. Navigate from your browser to:

<http://localhost:81/Admin/Login> or just <http://localhost:81> and you will see the Login home page. You can use the login information mentioned above or register with new credentials.

Note: the names of folders or drives may vary, you may need to change the name of the server and add or edit the port number depending on the configuration setting son your machine.

Using through Android Device

To run the android application on your android device your device must have installed the application in it. You can use the login information mentioned above or register with new credentials. Interface of application may vary with the change of android OS version. This application will confirm work properly in every aspect on android (5.0-Lollipop, 6.0-Marshmallow, 7.0-Nougat, and 8.0-Oreo).