



COMSATS University Islamabad (CUI)

Curriculum Management and Administration System

By

Raja Shaiyan Azad Khan	CIIT/FA16-BSE-084/ISB
Noor us Sahar	CIIT/SP17-BSE-011/ISB
Iqra Fareed	CIIT/SP17-BSE-037/ISB

Supervisor

Dr. Muhammad Waqar

Co- Supervisor

Dr. Hassan Ali Khattak

Bachelor of Science in Software Engineering (2017-2021)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.



COMSATS University Islamabad (CUI)

Curriculum Management and Administration System

**A project presented to
COMSATS University Islamabad**

**In partial fulfillment
of the requirement for the degree of**

Bachelor of Science in Software Engineering (2017-2021)

By

Raja Shaiyan Azad Khan	CIIT/FA16-BSE-084/ISB
Noor us Sahar	CIIT/SP17-BSE-011/ISB
Iqra Fareed	CIIT/SP17-BSE-037/ISB

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Raja Shaiyan Azad Khan

Noor us Sahar

Iqra Fareed

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (CS) “Curriculum Management and Administration System” was developed by **Syed Shaiyan Azad Khan (CIIT/FA16-BSE-084)** , **Noor us Sahar (CIIT/SP17-BSE-011)** and **Iqra Fareed (CIIT/SP17-BSE-037)** under the supervision of “Dr. Muhammad Waqar” and Co-supervisor “Dr. Hassan Ali Khattak” and that in their opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Sciences.



11-oct-2020

Supervisor

Co-Supervisor

External Examiner

Head of Department
(Department of Computer Science)

Executive Summary

In Universities and educational institutes, there are a lot of tasks that are to be done by the admin, faculty, coordinator which take up a lot of time if done manually. Appointing more people for different small tasks causes a lot of problems such as increased employee hiring, accuracy problem, trust, no proof for later use, and also the fact that a human can remember things till a certain time limit. Hence, there is a need for a system that reduces the workload of a user and also maintains the records over time.

The aim of this project is to provide services to admin and faculty of Shaheed Benazir Bhutto university to perform their manual tasks automatically, we have provided them a platform related to information technology where they will benefit from this system by using its unique service related to their requirements. Users will be able to provide and avail different functionalities on a single platform such as admissions, course enrollment, student record, grading, scheduling, reports, communication, faculty records maintenance. They are will three kind users faculty, admin and coordinators each of them will have their unique rolls and they will have access to only those tasks which are related to them. The software will prove to be efficient and helpful as they were performing these tasks manually.

Our system will address three major aspects that have to be considered when developing a single platform for three different kinds of users. Our system is a web application where users will be able to perform tasks on a friendly and easy-to-use UI.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Dr. Muhammad Waqar”. Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are deeply indebted to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

Raja Shaiyan Azad Khan

Noor us Sahar

Iqra Fareed

Abbreviations

SRS	Software Requirement Specification
UC	Use Case
TC	Test Case
MERN	MongoDB Express React NodeJS
SDS	Software Design Specification

Table of Contents

Contents

1	Introduction.....	1
1.1	Brief Overview.....	1
1.2	Relevance to Course Modules.....	1
1.3	Project Background.....	1
1.4	Literature Review.....	2
1.5	Analysis from Literature Review.....	2
1.1	Methodology and Software Lifecycle for this Project	2
2	Problem Definition.....	3
3	Requirement Analysis.....	4
3.1	Use Cases Diagram(s).....	4
3.1.1	Use Cases Admin.....	4
3.1.2	Use Cases Applicant	5
3.1.3	Use Cases Coordinator.....	6
3.1.4	Use Cases Faculty	7
3.1.5	Use Cases All Actors	8
3.2	Detailed Use Case	9
3.2.2	Use Cases of Module-1:	10
3.2.3	Use Cases of Module-2:	39
3.2.4	Use Cases of Module-3:	44
3.2.5	Use Cases of Module-4:	50
3.2.6	Use Cases of Module-5:	54
3.2.7	Use Cases of Module-6:	60
3.2.8	Use Cases of Module-7:	63
3.2.9	Use Cases of Module-8:	72
3.2.10	Use Cases of Module-9:	74
3.3	Functional Requirements	77
3.3.1	FRS Module-1:.....	77
3.3.2	FRS Module-2:.....	110
3.3.3	FRS MODULE-3:.....	114
3.3.4	FRS Module:4	123
3.3.5	FRS Module 5	126
3.3.6	FRS Module 6	131
3.3.7	FRS Module 7:	133
3.3.8	FRS Module 8:	143
3.3.9	FRS Module 9:	149
3.4	Non-Functional Requirements	153
3.4.1	Usability	153
3.4.2	Responsiveness	154
3.4.3	Performance	154
3.4.4	Scalability.....	154
3.4.5	Security	154
3.4.6	Reliability.....	154

4	Design and Architecture	155
4.1	System Architecture	155
4.2	Data Representation	155
4.2.1	Admission	155
4.2.2	Announcement	156
4.2.3	Applicant	157
4.2.4	Course	162
4.2.5	Department	165
4.2.6	Enrollment	165
4.2.7	Programme	166
4.2.8	Section	167
4.2.9	Student profile	168
4.2.10	Survey Forms	171
4.2.11	Survey Responses	172
4.2.12	User	172
4.2.13	Calendar Event	173
4.2.14	Faculty	174
4.2.15	Faculty Profile	174
4.3	Process Flow/Representation	177
4.3.1	Admission	177
4.3.2	Maintain Course Detail	178
4.3.3	Enrollment	179
4.3.4	Class Scheduling	180
4.3.5	Grading	181
4.3.6	Student Academic History	182
4.3.7	Faculty Record	183
4.3.8	Communication	184
4.3.9	Report Analysis	185
4.4	Design Models	186
4.4.1	Structural Diagrams	186
5	Implementation	197
5.1	Algorithm	197
5.2	User Interface	198
5.2.1	Landing Page	198
5.2.2	Admin Portal	198
5.2.3	Applicant Portal	207
5.2.4	Coordinator Portal	211
5.2.5	Faculty Portal	215
6	Testing and Evaluation	222
6.1	Manual Testing	222
6.1.1	Unit Testing	222
6.1.2	Integration Testing	282
7	Conclusion and Future Work	283
7.1	Conclusion	283
7.2	Future Work	283
8	References	283

List of Figures

Figure 3-1 : Use Case Admin.....	4
Figure 3-2 : Use Case Applicant.....	5
Figure 3-3 : Use Case Coordinator	6
Figure 3-4 : Use Case Faculty	7
Figure 3-5 : Use case All Actors.....	8
Figure 4-1: Process flow admission.....	177
Figure 4-2 Process flow Maintain Course Detail.....	178
Figure 4-3 Process Flow Enrollment	179
Figure 4-4 Process flow : Class Scheduling	180
Figure 4-5 Process Flow Grading	181
Figure 4-6 Process Flow Student Academic History	182
Figure 4-7 Process Flow Faculty Record.....	183
Figure 4-8 Process Flow Communication Chatbot.....	184
Figure 4-9 Process Flow Report Analysis	185
Figure 4-10 Level 0 DFD.....	186
Figure 4-11 Level 1 DFD.....	187
Figure 4-12 Level 2 DFD Admission Management	188
Figure 4-13 Level 2 DFD Course Management	189
Figure 4-14 Level 2 DFD Enrollment Management.....	190
Figure 4-15 Level 2 DFD Timetable Management.....	191
Figure 4-16 Level 2 DFD Grade Management	192
Figure 4-17 Level 2 DFD Student academic record	193
Figure 4-18 Level 2 DFD Faculty Management.....	194
Figure 4-19 Level 2 DFD Communication Management	195
Figure 4-20 Class Diagram	196
Figure 5-1 Landing Page.....	198
Figure 5-2 Admin Login Page	198
Figure 5-3 Admin Dashboard	199
Figure 5-4 Admin Faculty Page	199
Figure 5-5 Admin Faculty Page 2	200
Figure 5-6 Admin Coordinator Page.....	200
Figure 5-7 Admin Coordinator Page 2.....	201
Figure 5-8 Admin Applicant Page	201
Figure 5-9 Admin Student Page.....	202
Figure 5-10 Admin Student Page 2.....	202
Figure 5-11 Admin Admission Page.....	203
Figure 5-12 Admin Department Page	203
Figure 5-13 Admin Program Page	204
Figure 5-14 Admin Announcement Page	204
Figure 5-15 Admin Merit List Page.....	205
Figure 5-16 Admin Survey Page.....	205

Figure 5-17 Admin Survey Page 2.....	206
Figure 5-18 Admin Settings Page	206
Figure 5-19 Applicant Login Page.....	207
Figure 5-20 Applicant registration Page	207
Figure 5-21 Applicant Personal Details Page	208
Figure 5-22 Applicant Personal Page 2	208
Figure 5-23 Applicant Income Details Page	209
Figure 5-24 Applicant Education Details Page.....	209
Figure 5-25 Applicants Education Page 2	210
Figure 5-26 Applicants Settings Page.....	210
Figure 5-27 Applicants Settings Page 2.....	211
Figure 5-28 Coordinator Login Page	211
Figure 5-29 Coordinator Dashboard	212
Figure 5-30 Coordinator Manage Applicants	212
Figure 5-31 Coordinator Manage Students Page	213
Figure 5-32 Coordinator Manage Announcements Page.....	213
Figure 5-33 Coordinator Manage Sections	214
Figure 5-34 Coordinator Manage Courses.....	214
Figure 5-35 Faculty Login Page	215
Figure 5-36 Faculty Dashboard	215
Figure 5-37 Faculty Dashboard 2	216
Figure 5-38 Faculty Profiles Page.....	216
Figure 5-39 Faculty Self Profile Page.....	217
Figure 5-40 Faculty Edit Profile Page	217
Figure 5-41 Faculty Add Education Page	218
Figure 5-42 Faculty Add Research Paper Page	218
Figure 5-43 Faculty Add Experience Page	219
Figure 5-44 Faculty Calendar Page.....	219
Figure 5-45 Faculty Calendar Page 2.....	220
Figure 5-46 Faculty Grades Page 1.....	220
Figure 5-47 Faculty Grades Page 2.....	221
Figure 5-48 Faculty Grades Page 3.....	221
Figure 5-49 Faculty Survey Page.....	222

1 Introduction

This is an online system that makes it easier for faculty and admin to perform their daily tasks by automating the major activities that are being done manually. Faculty can register students online, upload and download assignments, see their attendance, upload lectures and quizzes of their general and hybrid courses and result cards. With the help of this system, we can perform online surveys to get reviews and feedback for improvement. This system will cater to many problems faced by the administration systems, under one umbrella. It can track and update records of fee submission, salary payment, and other financial tasks. This system will allow users to upload and download data within certain limits (faculty can only upload field relevant content). User interfaces are designed according to the targeted audience, such as different interfaces will be provided for admin and faculty according to their roles and authority.

1.1 Brief Overview

Our system will address three major aspects that have to be considered when developing software that provides a single platform to three different kinds of users who have three different kinds of roles and access to the system as each of them has different tasks to perform. First, our system will provide an admin portal where the admin can perform all the tasks they are supposed to do. This portal will provide a platform where the admin will be able to access, handle and maintain all the data that is only specific to them that particular portal will focus on the admin domain. Secondly, there is a coordinator portal where the tasks under the coordinator will be focused on, and thirdly there is a faculty portal faculty which has cover all the faculty related data and functionalities, which can be accessed and modified by faculty. Our system will provide all the required functionalities which are so easy to perform.

1.2 Relevance to Course Modules

This project is a result of courses we have studied throughout our degree like Human-Computer Interaction for UI development, Software Engineering Concepts for architecture and design of the system, Software Construction for good and efficient coding practices, Algorithm and Data structures for complex logics at the backend and frontend, Databases for how to store data in an effective manner and Web Engineering for how to host and deploy a system which can run in many types of system environments.

1.3 Project Background

In the modern-day and age providing services online has become a popular trend. Any person who knows basic computer and knows a little bit English can perform their job on technology but there still are so many institutes who are performing their jobs manually which can be automated. It makes their system efficient and saves so much time and energy. Data privacy and security can also be promised this way. So, this is a system which is providing a single platform

with different interfaces to make all those manual works automated. When the concept of automation comes many people think that this is going to be more complex and non-technical people will not be able to use them, but it is otherwise. The system is designed in such a way that any person with some basic knowledge can easily use the system. In fact, the system makes it easier for them as it was before. As we all know in COVID-19 types nothing can be done without technical platforms

1.4 Literature Review

- CU Online: This is a system where all students and faculty and admin are provided with a platform where they can access and modify data as per their needs. All tasks related to these actors can be performed on this system
- CMS Nust: This is pretty similar system it provides almost same functionalities as per mentioned above but it has to different dedicated platforms.

1.5 Analysis from Literature Review

Application Name	Weakness	Proposed Project Solution
CU Online	Simple but reduced functionality offered to admin.	Interactive and easy to learn user interface with more functionality.
CMS nust	Weakness is that courses are dealt on another portal one platform is not dealing major functionality.	Combining both the portal functionalities and making a unified system.

1.1 Methodology and Software Lifecycle for this Project

We have followed the procedural design methodology and the Iterative process model for the development of this system

1.1.1 Rationale behind the Selected Methodology

We have followed the Procedural design methodology because we are using ReactJS which mostly uses functional-based components that reduce the need to create classes and objects, although we did use some class-based components they were very few to mention. The iterative process model has been followed because many of the requirements for the project are clear and defined as stated in the scope document and if any new requirements are identified they will also be included in the project as well.

2 Problem Definition

2.1 Problem Statement

In this era of technology, there are many problems faced by the usage of manual systems such as calculating percentages, updating records, maintaining backups, fast searching and many other issues are faced by fast-growing organizations, every semester hundreds of new students are enrolled and a great number of them graduate while many students are promoted or they repeat the semester. Usually, institutes save this kind of data in excel sheets but again there is not one specified and customized system that can handle all the fields at one platform. Admin can't broadcast the notifications to all individuals at once in current systems and it is not easy for the admin to perform these operations since manual functionality requires more time and effort. Systems like these are currently absent in many institutions but they have some limitations like (sending separate notification systems regarding updates and announcements in the institutions and dealing with major administration-based activities at one platform), which we'll rectify in our project. The major and most common problems are that they are very sensitive to human errors or mistakes like the wrong entry of student's credentials etc.

2.2 Deliverables and Development Requirements

Following are the deliverables of the project

- Project Scope Document
- Software Requirement Specification Document (SRS)
- Software Design Document (SSD)
- 40% implementation
- 70% implementation
- 100% implementation
- Final Project Report

2.3 Current System

Application Name	Weakness	Proposed Project Solution
Cu Online	Simple but reduced functionality offered to admin.	Interactive and easy to learn user interface with more functionality.
CMS NUST	Weakness is that courses are dealt on another portal one platform is not dealing major functionality.	Combining both the portal functionalities and making a unified system.

3 Requirement Analysis

3.1 Use Cases Diagram(s)

3.1.1 Use Cases Admin

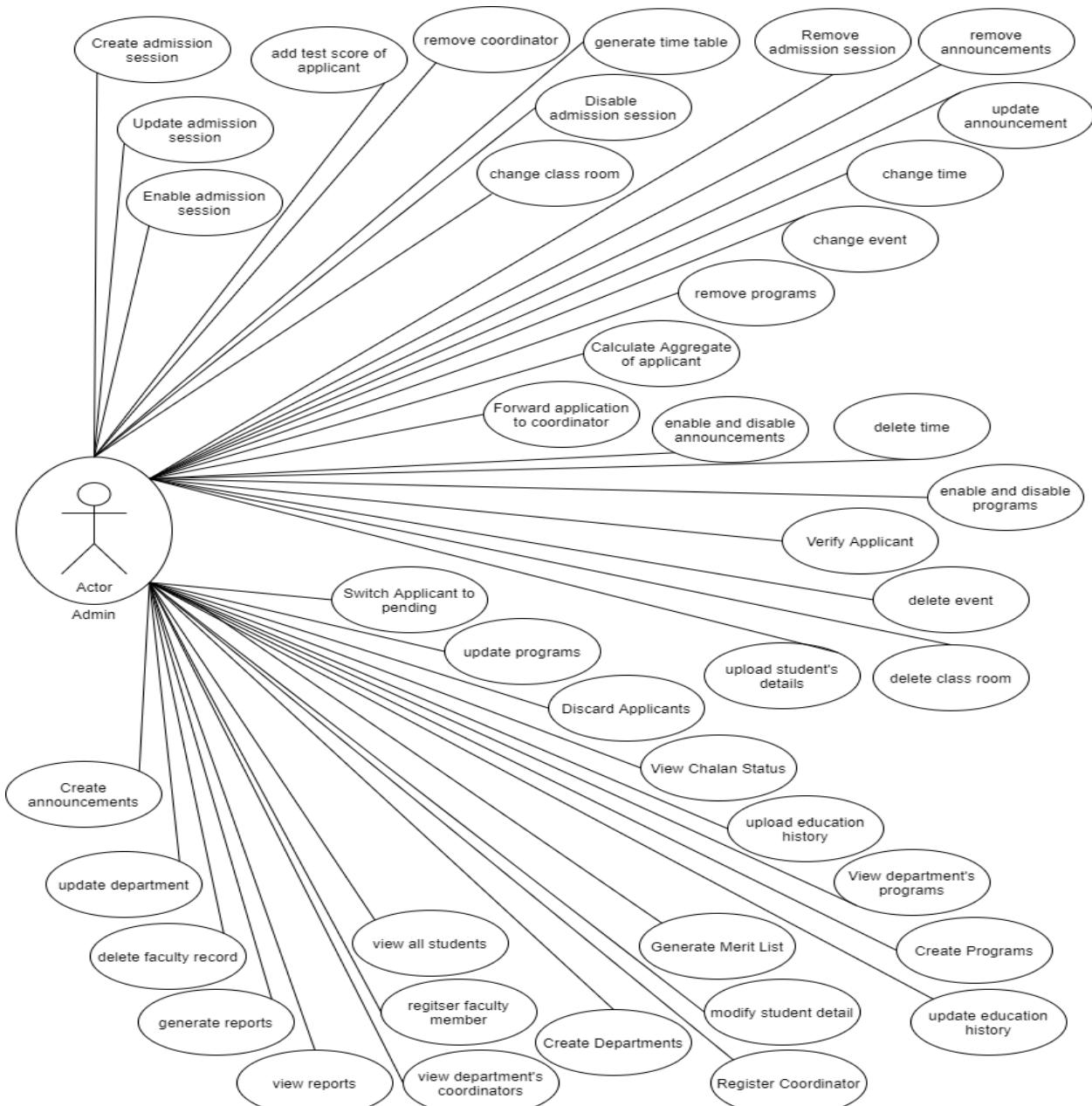


Figure 3-1 Use Case Admin

3.1.2 Use Cases Applicant

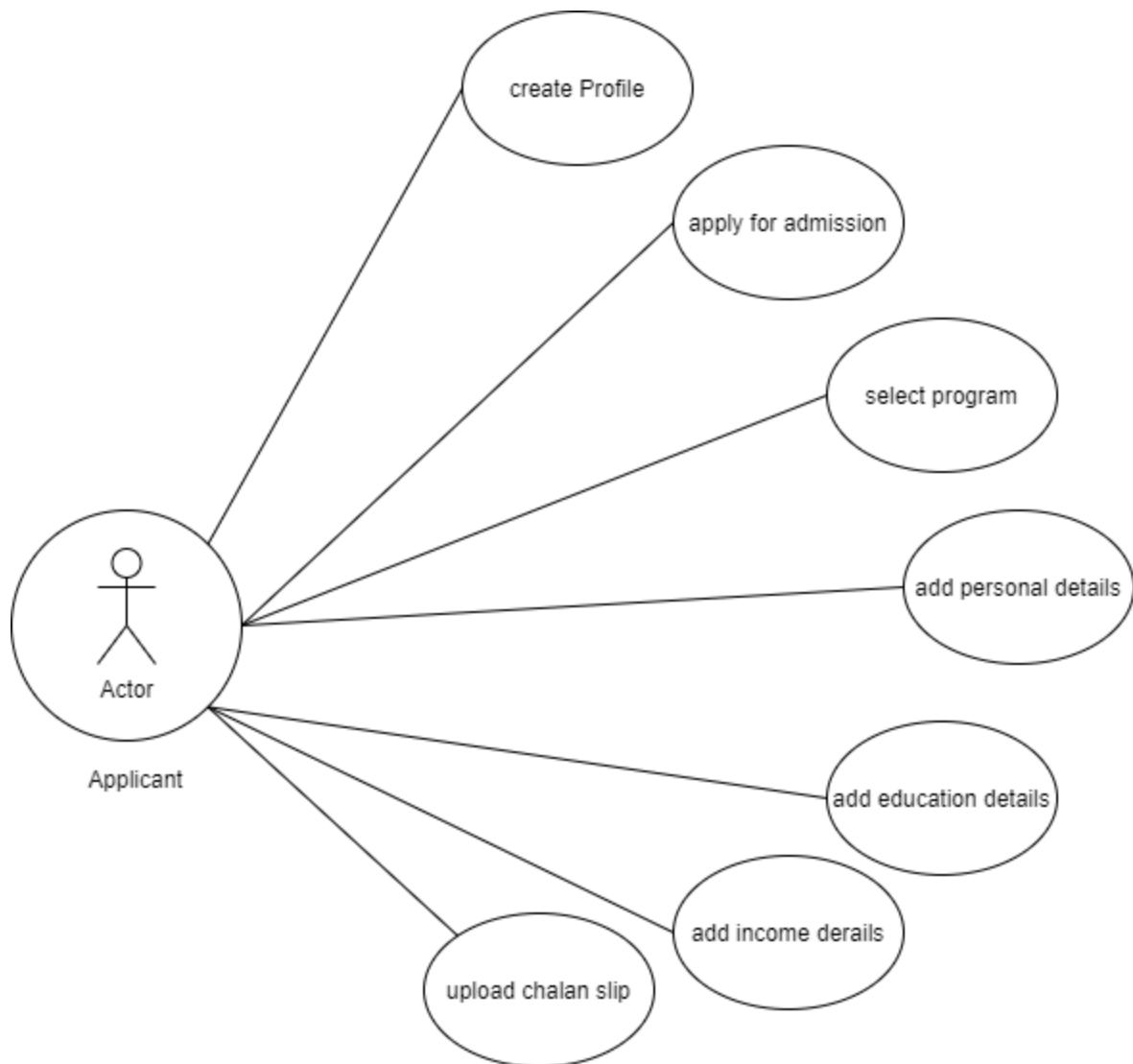


Figure 3-2 : Use Case Applicant

3.1.3 Use Cases Coordinator

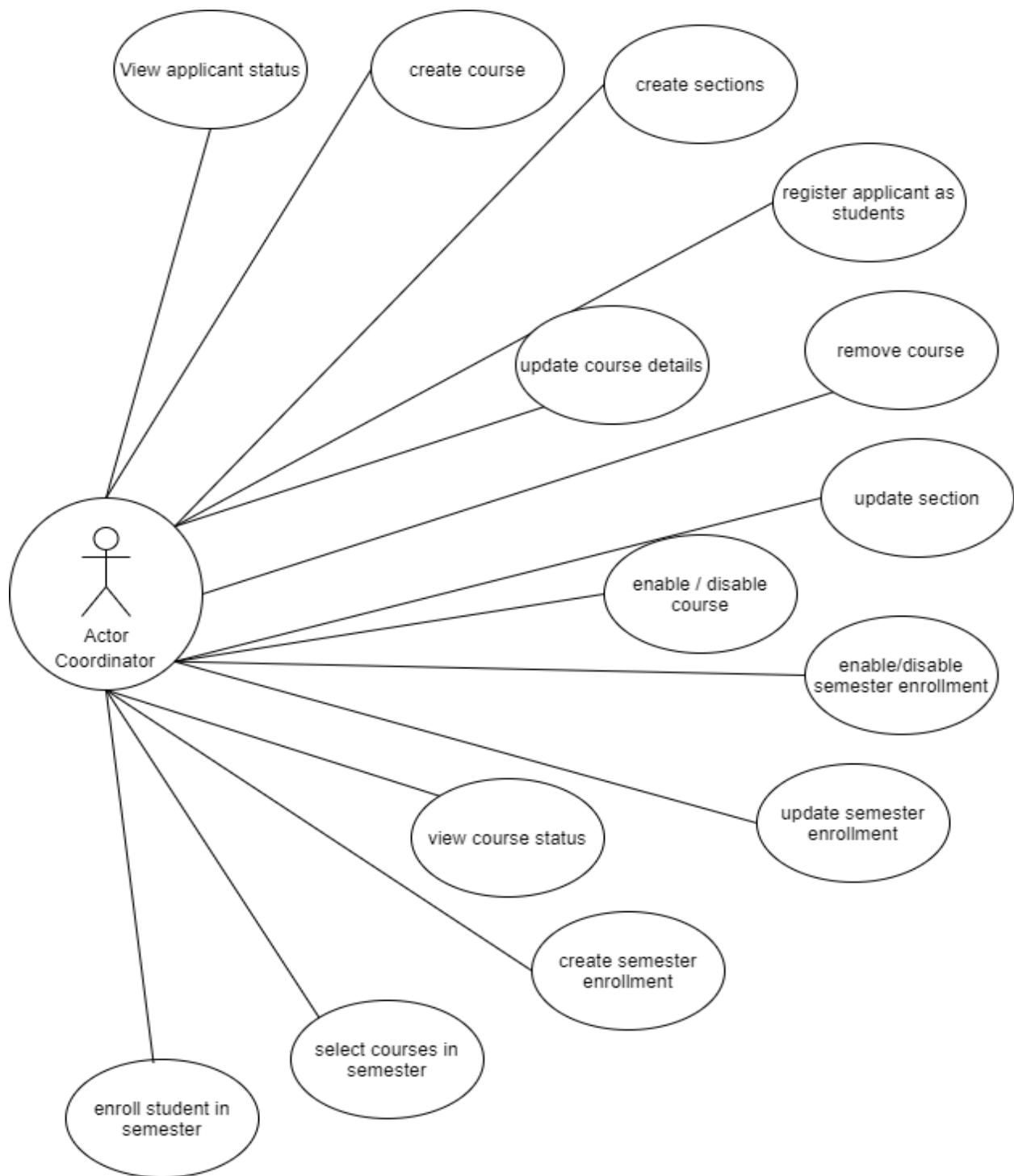


Figure 3-3 : Use Case Coordinator

3.1.4 Use Cases Faculty

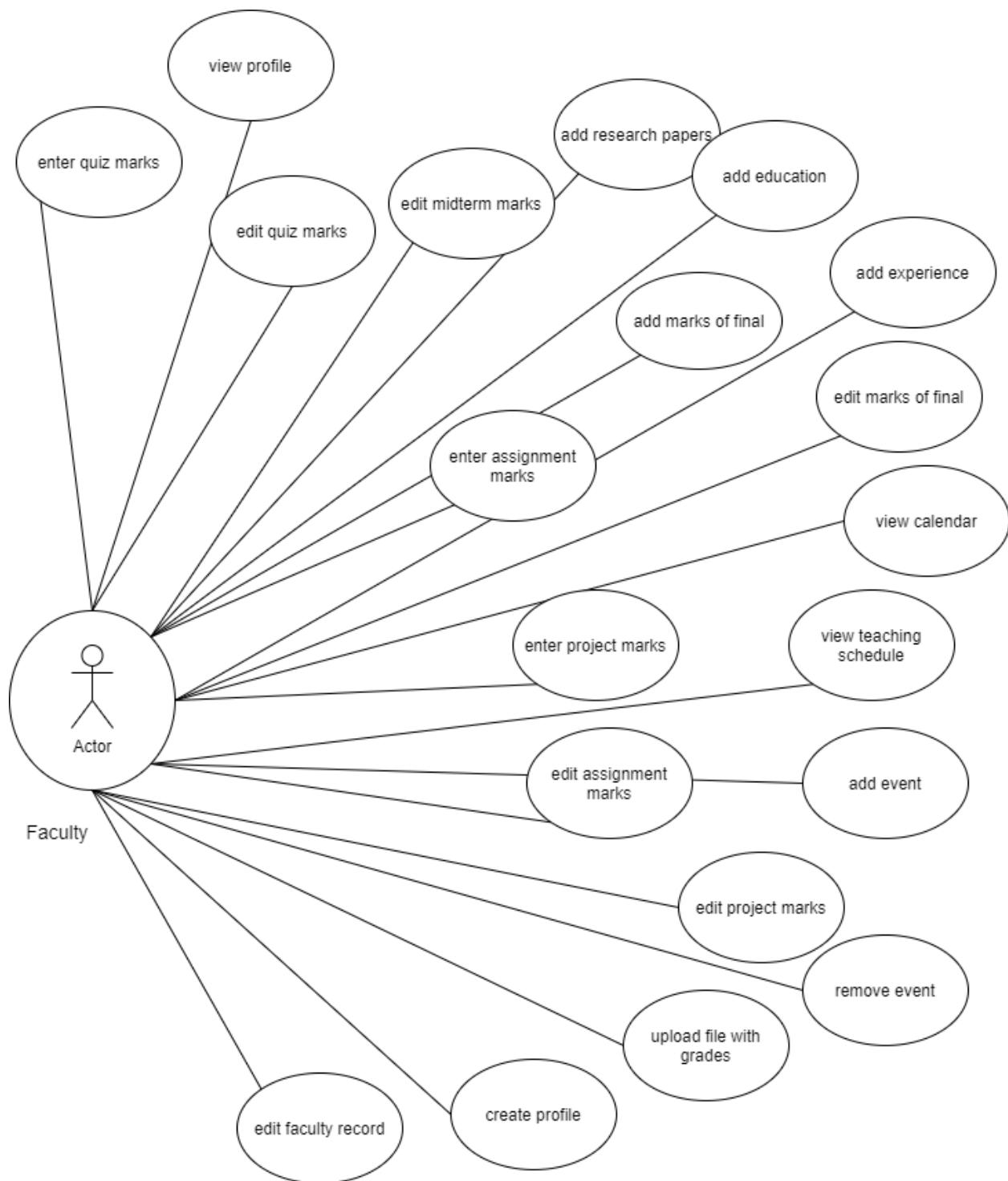


Figure 3-4 : Use Case Faculty

3.1.5 Use Cases All Actors

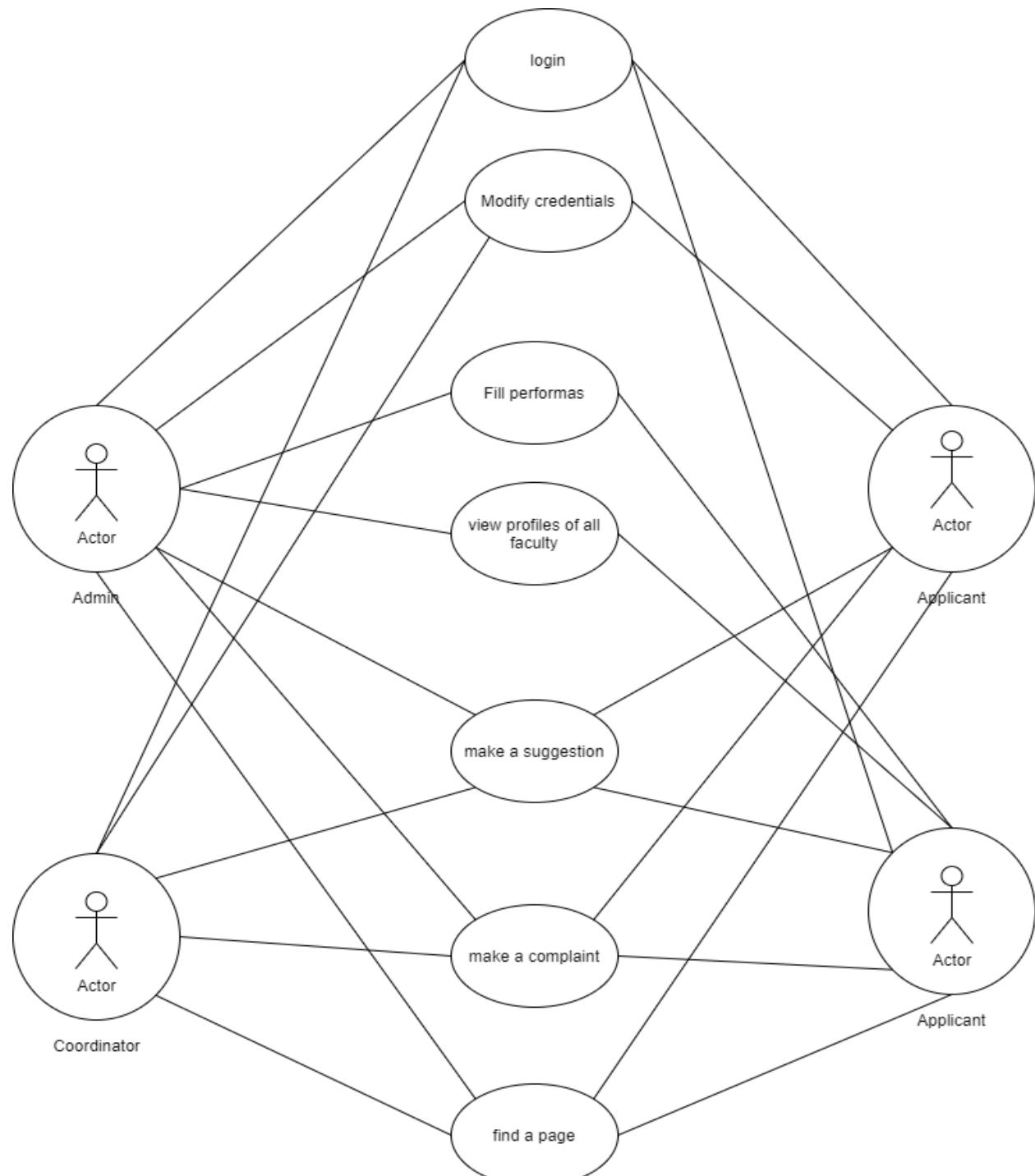


Figure 3-5 : Use case All Actors

3.2 Detailed Use Case

3.2.1.1 UC-1: Access Control

Use Case ID	UC-1
Use Case Name	Access Control
Actors	Primary Actor: Admin, Coordinator
Description	An authorized admin will register and sign in all the enrolled students and teachers in the portal and allot them credentials to make access through the portal
Trigger	New students and teachers are enrolled and are required to have their own login.
Preconditions	PRE-1: Admin is logged in the system.
Postconditions	POST-1: Login credentials are emailed to the student or teacher enrolled. POST-2: A message is displayed “User signed up”
Normal Flow	<ol style="list-style-type: none"> 1. Admin selects the option between student and faculty. 2. Admin enters the credentials to sign up the user. 3. System validates the credentials 4. System sends a verification mail to the user who is signed up by the admin. 5. Admin is displayed a message “User signed up”
Alternative Flows	<p>3a. In normal flow 3, if the system doesn't validate the credentials.</p> <ol style="list-style-type: none"> 1. User is displayed a message “Wrong credentials, enter again” 2. User is shown the sign up form again. 3. Normal flow continues from point 2.
Exceptions	N/A
Business Rules	Only authorized and enrolled members can have access
Assumptions	N/A

3.2.1.2 UC-2: Login

Use Case ID	UC-2
Use Case Name	Login
Actors	Primary Actor: Admin, Coordinator, Faculty
Description	An admin wants to get access to the portal so he/she will have to login the system to get access to facilities.com.
Trigger	User will click on the login button
Preconditions	PRE-1: User is registered..
Postconditions	POST-1: User is logged in the system. POST-2: Home page is displayed to the user.
Normal Flow	<ol style="list-style-type: none"> 1. User enter his/her password and username 2. User presses enter or selects the submit button.

	<ul style="list-style-type: none"> 3. Username and password are verified by the system. 4. The home page is displayed.
Alternative Flows	<ul style="list-style-type: none"> 3a. If the username or password are not correct. <ul style="list-style-type: none"> 1. User is shown a message “Username or password incorrect.” 2. User is prompted to enter the credentials again. 3. Normal flow continues from point 1.
Exceptions	N/A
Business Rules	Only enrolled students and faculty can login into the system
Assumptions	N/A

3.2.2 Use Cases of Module-1:

3.2.2.1 UC-3: Create Admission Session

Use Case ID	UC-3
Use Case Name	Create Admission Session
Actors	Primary Actor : Admin
Description	The admin will be able to create the admission session.
Trigger	The admin will click on the create admission session button.
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to create admission sessions.
Postconditions	POST-1: The Admission session is created. POST-2: The Admission session is enabled by default. POST-3: The Applicant can apply for the Admission session.
Normal Flow	<ul style="list-style-type: none"> 1. Admin selects the create admission session option 2. System shows a form with the fields to create admission session 3. Admin enters the details 4. Admin clicks on submit 5. Admission session gets created on the dashboard 6. Admission session is enabled by default 7. Applicant can apply for the session
Alternative Flows	N/A
Exceptions	6a. If the end date has passed, the session will automatically turn status to disabled <ul style="list-style-type: none"> 1. Session can be enabled manually. 2. Normal flow continues from step 7 of normal flow.
Business Rules	N/A
Assumptions	N/A

3.2.2.2 UC-4: Set Start and End Date of Admission Sessions

Use Case ID	UC-4
Use Case Name	Set Start and End Date of Admission Sessions
Actors	Primary Actor : Admin
Description	The admin will be able to set the start and end date of the admission according to the schedule.
Trigger	The admin will select the set date of Admissions .
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to set the dates.
Postconditions	POST-1: The start date is set POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> Admin selects the set date of the admissions. System shows a form with start and end date to enter Admin enters the dates System asks for confirmation Admin saves. Admin displays the start and end Date .
Alternative Flows	<p>3a. If the admin enters th date such that date is behind the start date</p> <ol style="list-style-type: none"> Generate an error and prompt the user to enter again. Normal flow continues from step 2.
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.3 UC-5: Create Department

Use Case ID	UC-5
Use Case Name	Create Department
Actors	Primary Actor : Admin
Description	The admin will be able to create the departments
Trigger	The admin will click on create department
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to create the department PRE-3: Admin is on manage department section

Postconditions	POST-1: Department is created POST-2: Success message is displayed POST-3: Programs can be added to the departments
Normal Flow	<ol style="list-style-type: none"> 1. The Admin selects create department 2. System shows the form to fill in the department 3. The Admin completes the form and clicks on submit button 4. System display the success message and displays the created department
Alternative Flows	<p>3a. If the department is already created</p> <ol style="list-style-type: none"> 1. Department can be updated 2. System continues from step 3 of normal flow
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.4 UC-6: Update Department

Use Case ID	UC-6
Use Case Name	Update Department
Actors	Primary Actor : Admin
Description	The admin will be able to update the departments fields
Trigger	The admin will click on update department button
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to update the department PRE-3: Admin is on manage department section
Postconditions	POST-1: Department fields are updated POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on update department button 2. System will display the admin the fields of department to be updated 3. Admin will update the required fields 4. Admin will click on submit 5. System will display success message indicating department fields are updated 6. Updated fields will be seen on manage department section
Alternative Flows	N/A

Exceptions	<p>3a. If there are no fields that have been updated</p> <ol style="list-style-type: none"> 1. System shall display the message indicating that no change has been done 2. The use case will be terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.5 **UC-7: View Departments Coordinator**

Use Case ID	UC-7
Use Case Name	View Departments Coordinator
Actors	Primary Actor : Admin
Description	The admin will be able to view the departments coordinator after coordinators have been created
Trigger	The admin will click on coordinators button next to respective department
Preconditions	PRE-1: Admin is logged in
Postconditions	POST-1: Department coordinators list can be viewed
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on view department coordinator 2. System will redirect the Admin to the list of Coordinators of the department 3. Admin can view the coordinators inside a department
Alternative Flows	N/A
Exceptions	<p>3a. If there are no coordinators that exists</p> <ol style="list-style-type: none"> 3. System shall display the message indicating that no coordinators have been found 4. Admin will click on back button and go back to the previous page
Business Rules	N/A
Assumptions	N/A

3.2.2.6 UC-8: View Departments Program

Use Case ID	UC-8
Use Case Name	View Departments Program
Actors	Primary Actor : Admin
Description	The admin will be able to view the departments programs after programs have been created
Trigger	The admin will click on programs button next to respective department
Preconditions	PRE-1: Admin is logged in
Postconditions	POST-1: Department programs list can be viewed
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on view department programs 2. System will redirect the Admin to the list of Programs of the department 3. Admin can view the programs inside a department
Alternative Flows	N/A
Exceptions	<p>3a. If there are no programs that exists</p> <p>5. System shall display the message indicating that no programs have been found</p> <p>6. Admin will click on back button and go back to the previous page</p>
Business Rules	N/A
Assumptions	N/A

3.2.2.7 UC-9: Create Program

Use Case ID	UC-9
Use Case Name	Create Program
Actors	Primary Actor : Admin
Description	The admin will be able to create the program
Trigger	The admin will click on create program
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to create the program PRE-3: Admin is on manage program section
Postconditions	POST-1: Program is created POST-2: Success message is displayed POST-3: Applicant can select the program they can apply for

Normal Flow	<ol style="list-style-type: none"> 1. The Admin selects create program 2. System shows the form to fill in for the creating program 3. The Admin completes the form and clicks on submit button 4. Created programs can be seen on the manage program tab and applicant apply tab for them to apply into the programs they desire.
Alternative Flows	3a. If the program is already created <ol style="list-style-type: none"> 1. Program can be updated 2. System continues from step 3 of normal flow
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.8 ***UC-10: Update Program***

Use Case ID	UC-10
Use Case Name	Update Program
Actors	Primary Actor: Admin
Description	The Admin can update program after it has been created, by clicking on the update button next to the respective program
Trigger	The Admin will click on update button
Preconditions	PRE-1: Admin is logged in PRE-2: Program exists in order to be updated
Postconditions	POST-1: Programs details will be modified POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin clicks on update button 2. Admin updates any of the required fields of the program 3. Admin clicks on submit 4. System displays a success message indicating the fields have been updated successfully 5. System will redirect the Admin to manage programs section 6. System will display the modifications made on the program details
Alternative Flows	N/A
Exceptions	3a. If there are no fields that have been updated

	<ol style="list-style-type: none"> 1. System shall display the message indicating that no change has been done 2. The use case will be terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.9 ***UC-11: Enable Program***

Use Case ID	UC-11
Use Case Name	Enable Program
Actors	Primary Actor : Admin
Description	The Admin can enable the status of program by id
Trigger	The Admin will click on enable button
Preconditions	PRE-1: Admin is logged in PRE-2: Program exists or is disabled in order to be enabled
Postconditions	POST-1: Program status will be enabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Admin clicks on the Enable button next to the respective program they desire to enable 2. System enables the program 3. System displays the success message indicating the program enabled
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.10 ***UC-12: Disable Program***

Use Case ID	UC-12
Use Case Name	Disable Program

Actors	Primary Actor : Admin
Description	The Admin can disable the status of program by id
Trigger	The Admin will click on disable button
Preconditions	PRE-1: Admin is logged in PRE-2: program is disabled or exists in order to be enabled
Postconditions	POST-1: Program status will be disabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Admin clicks on the Disable button next to the respective program which is enabled 2. System disables the program 3. System displays the success message indicating the program has been disabled
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.11 **UC-13: Remove Program**

Use Case ID	UC-13
Use Case Name	Remove Program
Actors	Primary Actor : Admin
Description	The Admin can remove the Program by id
Trigger	The Admin will click on remove button
Preconditions	PRE-1: Admin is logged in PRE-2: Program exists in order to be removed
Postconditions	POST-1: Program will be removed POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin desires to remove the program 2. Admin clicks on remove button next to a program 3. System will display a message to confirm in order to remove a remove 4. Admin confirms to remove the program 5. System removes the program
Alternative Flows	N/A

Exceptions	3a. Admin does not confirm to remove the program 1. The use case will be terminated.
Business Rules	N/A
Assumptions	N/A

3.2.2.12 UC-14: Create Announcement Feed

Use Case ID	UC-14
Use Case Name	Create Announcement Feed
Actors	Primary Actor : Admin
Description	The admin will be able to create the announcement feed
Trigger	The admin will click on create announcement feed
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to create the announcement feed PRE-3: Admin is on manage announcement section
Postconditions	POST-1: Announcement feed is created POST-2: Success message is displayed POST-3: Applicant can select the announcement feed they can apply for
Normal Flow	1. The Admin selects create announcement feed 2. System shows the form to fill in for the creating announcement feed 3. The Admin completes the form and clicks on submit button 4. Created feeds can be seen on the manage announcement tab and applicants can view them on the landing page
Alternative Flows	3a. If the announcement feed is already created 1. Announcement feed can be updated 2. System continues from step 3 of normal flow
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.13 UC-15: Update Announcement Feed

Use Case ID	UC-15
Use Case Name	Update Announcement Feed
Actors	Primary Actor : Admin
Description	The Admin can update announcement feed after it has been created, by clicking on the update button next to the respective announcement feed
Trigger	The Admin will click on update button
Preconditions	PRE-1: Admin is logged in PRE-2: Announcement feed exists in order to be updated
Postconditions	POST-1: Announcement feed details will be modified POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin clicks on update button 2. Admin updates any of the required fields of the announcement feed 3. Admin clicks on submit 4. System displays a success message indicating the fields have been updated successfully 5. System will redirect the Admin to manage announcement feed section 6. System will display the modifications made on the announcement feed details
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.14 UC-16: Enable Announcement Feed

Use Case ID	UC-16
Use Case Name	Enable Announcement Feed
Actors	Primary Actor : Admin
Description	The Admin can enable the status of feed by id
Trigger	The Admin will click on enable button
Preconditions	PRE-1: Admin is logged in PRE-2: Announcement feed exists or is disabled in order to be enabled

Postconditions	POST-1: Announcement feed status will be enabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Admin clicks on the Enable button next to the respective feed they desire to enable 2. System enables the feed 3. System displays the success message indicating the announcement feed has been enabled
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.15 **UC-17: Disable Announcement Feed**

Use Case ID	UC-17
Use Case Name	Disable Announcement Feed
Actors	Primary Actor : Admin
Description	The Admin can disable the status of feed by id
Trigger	The Admin will click on disable button
Preconditions	PRE-1: Admin is logged in PRE-2: Announcement feed is disabled or exists in order to be enabled
Postconditions	POST-1: Announcement feed status will be disabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Admin clicks on the Disable button next to the respective feed which is enabled 2. System disables the announcement feed 3. System displays the success message indicating the feed has been disabled
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A

Assumptions	N/A
--------------------	-----

3.2.2.16 **UC-18: Remove Announcement Feed**

Use Case ID	UC-18
Use Case Name	Remove Announcement Feed
Actors	Primary Actor : Admin
Description	The Admin can remove the feed by id
Trigger	The Admin will click on remove button
Preconditions	PRE-1: Admin is logged in PRE-2: Announcement feed exists in order to be removed
Postconditions	POST-1: Announcement feed will be removed POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin desires to remove the announcement feed 2. Admin clicks on remove button next to a feed 3. System will display a message to confirm in order to remove a feed 4. Admin confirms to remove the respective feed 5. System removes the feed
Alternative Flows	N/A
Exceptions	<p>3a. Admin does not confirm to remove the feed</p> <p>2. The use case will be terminated.</p>
Business Rules	N/A
Assumptions	N/A

3.2.2.17 **UC-19: Register Coordinator**

Use Case ID	UC-19
Use Case Name	Register Coordinator
Actors	Primary Actor : Admin
Description	The admin will be able to register a coordinator
Trigger	The admin will click on add coordinator button

Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to register coordinator PRE-3: Admin is on manage coordinator section
Postconditions	POST-1: Coordinator is registered POST-2: Success message is displayed POST-3: Coordinator can login to their portal
Normal Flow	<ol style="list-style-type: none"> 1. Admin will navigate to Manage Coordinator section 2. Admin will click on Add Coordinator 3. System will display the registration form 4. Admin will fill in the registration form 5. Admin will click on submit 6. System will display a success message 7. Registered Coordinator list can be seen on the Manage Coordinator section
Alternative Flows	N/A
Exceptions	<p>5a. If the credentials of Coordinator are same as the created coordinator</p> <ol style="list-style-type: none"> 1. The system will display the error message indicating that the user already exists 2. The use case will resume from step 4 of Normal Flow
Business Rules	N/A
Assumptions	N/A

3.2.2.18 UC-20: Remove Coordinator

Use Case ID	UC-20
Use Case Name	Remove Coordinator
Actors	Primary Actor : Admin
Description	The Admin can remove the coordinator by id
Trigger	The Admin will click on remove button
Preconditions	PRE-1: Admin is logged in PRE-2: Coordinator feed exists in order to be removed
Postconditions	POST-1: Coordinator will be removed POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin desires to remove the coordinator

	<ol style="list-style-type: none"> 2. Admin clicks on remove button next to a coordinator 3. System will display a message to confirm in order to remove a coordinator 4. Admin confirms to remove the coordinator 5. System removes the coordinator
Alternative Flows	N/A
Exceptions	<p>3a. Admin does not confirm to remove the coordinator</p> <p>3. The use case will be terminated.</p>
Business Rules	N/A
Assumptions	N/A

3.2.2.19 **UC-21: Generate Merit List**

Use Case ID	UC-21
Use Case Name	Generate Merit List
Actors	Primary Actor: Admin
Description	The admin will be able to generate a merit list of students after they have given their entry test and their total aggregate has been calculated.
Trigger	The admin will click on generate merit list button
Preconditions	PRE-1: Admin is logged in PRE-2: Admin is authorized to generate merit list PRE-3: Admin is on their dashboard
Postconditions	POST-1: Merit list is generated POST-2: Success message is displayed POST-3: Applicant will be displayed a notification for acceptance on their dashboard. POST-4: Applicant can upload the fee challan of the respective program
Normal Flow	<ol style="list-style-type: none"> 1. Admin will navigate to Dashboard 2. Admin will create the session 3. Admin will enter student university test score and calculate their aggregate 4. Admin will click on the generate merit list button on dashboard 5. System will generate merit list of all the applicants having their test scores above that of minimum criteria set 6. Admin will be able to view the generated Merit list on Manage List

	section
Alternative Flows	N/A
Exceptions	<p>3a. If Admin has not yet calculated the merit list of any applicant and clicks on generate merit list</p> <ol style="list-style-type: none"> 1. The system shall generate an empty list 2. The use case resumes from step 5 of normal flow
Business Rules	N/A
Assumptions	N/A

3.2.2.20 UC-22: Add Test Score

Use Case ID	UC-22
Use Case Name	Add Test Score
Actors	Primary Actor : Admin
Description	Admin will be able to add test score of the applicants who have applied for a program and they have appeared in the university examination
Trigger	Admin will be able to add test score of the applicants
Preconditions	<p>PRE-1: Admin is logged in</p> <p>PRE-2: Applicant has given the entry test</p> <p>PRE-3: Admin has received the entry test scores of Applicant</p>
Postconditions	<p>POST-1: Applicant's test score has been added</p> <p>POST-2: Success message is displayed</p> <p>POST-3: Applicant's aggregate can be calculated</p>
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on view button next to the applicant and go on their details section 2. Admin will enter the test score under aggregate section 3. Admin will click on the add test score 4. System will display a message indicating the test scores have been updated
Alternative Flows	N/A
Exceptions	<p>3a. If Admin has already added the test score for the Applicant</p> <ol style="list-style-type: none"> 1. They will be updating the test score if desired

	2. The use case will resume from step 3 of normal flow
Business Rules	N/A
Assumptions	N/A

3.2.2.21 UC-23: Calculate Aggregate of the Applicant

Use Case ID	UC-23
Use Case Name	Calculate Aggregate of the Applicant
Actors	Primary Actor : Admin
Description	Admin will be able to calculate aggregate of the applicants who have applied for a program and their test score has been added
Trigger	Admin will click on the calculate aggregate button
Preconditions	PRE-1: Admin is logged in PRE-2: Applicant has given the entry test PRE-3: Admin has received the entry test scores of Applicant PRE-4: Admin has added the test score of Applicant
Postconditions	POST-1: Applicant's aggregate will be calculated POST-2: Success message is displayed POST-3: Admin can request system to generate merit list
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on view button next to the applicant and go on their details section 2. Admin will click on the calculate aggregate button below the test score button 3. System will calculate the aggregate of the applicant 4. System will display the aggregate in aggregate section
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.22 UC-24: Forward Applicant Details to the Coordinator

Use Case ID	UC-24
Use Case Name	Forward Applicant Details to the Coordinator
Actors	Primary Actor : Admin
Description	Admin will be able to forward the applicant details to the coordinator, so that they can view the status of the applicant and register them as student after verification
Trigger	Admin will click on the forward applicant button
Preconditions	PRE-1: Admin is logged in PRE-2: Applicants aggregate has been calculated PRE-3: Applicants aggregate is above the minimum required aggregate of the program
Postconditions	POST-1: Applicant's details will be forwarded to Coordinator POST-2: Coordinator will be able to view Applicant's details
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on forward applicant button 2. System will change the status of applicant to forwarded 3. System will forward applicant details to coordinator 4. Coordinator can view the details of the Applicant forwarded
Alternative Flows	N/A
Exceptions	<p>1a. If the applicant aggregate has not yet calculated</p> <ol style="list-style-type: none"> 1. Applicant will not be forwarded 2. System will display error message indicating the applicant has been forwarded
Business Rules	N/A
Assumptions	N/A

3.2.2.23 UC-25: Verify Applicant

Use Case ID	UC-25
Use Case Name	Verify Applicant
Actors	Primary Actor: Admin
Description	Admin will be able to verify applicant after they have uploaded their submitted fee challan image

Trigger	Admin will click on the verify button
Preconditions	PRE-1: Admin is logged in PRE-2: Applicants has uploaded their submitted challan image
Postconditions	POST-1: Applicant's status will be changed to verified POST-2: Coordinator will be able to view Applicant's status in order to register them as a student
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on status to view the submitted challan image 2. System will display the challan image of submitted program fee 3. Admin will save the changes 4. Admin will click on verify 5. System will verify the applicant 6. Coordinator can view the status of the applicant as verified
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.24 UC-26: Change Applicant Status to Pending

Use Case ID	UC-26
Use Case Name	Change Applicant Status to Pending
Actors	Primary Actor: Admin
Description	Admin will be able to change applicant's status to pending in case they have not uploaded the submitted fee challan on the announced date for admissions closing.
Trigger	Admin will click on the pending button
Preconditions	PRE-1: Admin is logged in PRE-2: Applicants has not yet uploaded their submitted challan image
Postconditions	POST-1: Applicant's status will be changed to pending POST-2: Coordinator will be able to view Applicant's status pending applicants will not be registered
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on status to view the submitted challan image 2. System will not display challan image 3. Admin will save the changes 4. Admin will click on pending button 5. System will change the status of applicant as pending

	6. Coordinator can view the status of the applicant as pending
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.25 UC-27: Discard Applicant

Use Case ID	UC-27
Use Case Name	Discard Applicant
Actors	Primary Actor: Admin
Description	Admin will be able to change applicant's status to discarded in case they have not uploaded the submitted fee challan on the announced last date.
Trigger	Admin will click on the discard button
Preconditions	PRE-1: Admin is logged in PRE-2: Applicants has not yet uploaded their submitted challan image PRE-3: Last date for uploading submitted challan fee has been announced by the Admin
Postconditions	POST-1: Applicant's status will be changed to discarded POST-2: Discarded Applicants will be moved to discarded list of applicants
Normal Flow	<ol style="list-style-type: none"> 1. Admin will click on status to view the submitted challan image 2. System will not display challan image 3. Admin will save the changes 4. Admin will click on discard button 5. System will change the status of applicant as discarded 6. System will push the applicant to discarded list
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.26 UC-28: Create Profile

Use Case ID	UC-28
Use Case Name	Create Profile
Actors	Primary Actor: Applicant
Description	Applicants will be able to create their profile by filling in their personal details, income details, education details and upload their profile picture in the settings menu.
Trigger	Applicants will click on apply for the program and select programs after which they will click proceed.
Preconditions	PRE-1: Applicant is logged in PRE-2: Applicant is has not applied for any program earlier PRE-3: Applicant profile does not exists
Postconditions	POST-1: Applicant's data has been submitted POST-2: Success message is displayed POST-3: Applicant picture and their details can be seen by faculty and coordinator
Normal Flow	<ol style="list-style-type: none"> 1. Applicant will click on apply now option on the registration page 2. After applicant has completed registration, they will click on apply for program option 3. Applicants will click on proceed and they will be redirected to personal details form, income details form and educational details form. 4. After completing these forms, they will be redirected to Settings tab and they will upload their profile picture 5. After which all their details can be seen by the Admin and Coordinator
Alternative Flows	N/A
Exceptions	<p>3a. If Applicant has already applied for a program</p> <p>3. They will not be able to re-apply again</p> <p>4. The use case will resume from step 5 of normal flow</p>
Business Rules	N/A
Assumptions	N/A

3.2.2.27 UC-29: Modify Credentials

Use Case ID	UC-29
Use Case Name	Modify Credentials
Actors	Primary Actor: Admin, Coordinator, Applicant
Description	The users will be able to modify their credentials which includes their name and password
Trigger	The users will click on settings tab indicating their desire to change their credentials
Preconditions	PRE-1: User is logged in PRE-2: User is authorized to modify their credentials
Postconditions	POST-1: Users name/password will be changed POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. After the user has logged into their portals 2. They can navigate to the settings tab 3. Here they can update their username or password as desired 4. After updating their username or password they can click on submit button 5. System will display a success message indicating the username or password has been updated
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.28 UC-30: Update Admission Session

Use Case ID	UC-30
Use Case Name	Update Admission Session
Actors	Primary Actor: Admin
Description	The Admin can update admission session after it has been created, by clicking on the update button next to the respective session
Trigger	The Admin will click on update button
Preconditions	PRE-1: Admin is logged in PRE-2: Admission session exists in order to be updated

Postconditions	POST-1: Admission sessions details will be modified POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin clicks on update button 2. Admin updates any of the required fields of the Admission session 3. Admin clicks on submit 4. System displays a success message indicating the fields have been updated successfully 5. System will redirect the Admin to Dashboard 6. System will display the modifications made on the session details
Alternative Flows	N/A
Exceptions	<p>3a. If there are no fields that have been updated</p> <ol style="list-style-type: none"> 7. System shall display the message indicating that no change has been done 8. The use case will resume from step 4 of normal flow
Business Rules	N/A
Assumptions	N/A

3.2.2.29 UC-31: Enable Admission Session

Use Case ID	UC-31
Use Case Name	Enable Admission Session
Actors	Primary Actor : Admin
Description	The Admin can enable the status of Admission Session by id
Trigger	The Admin will click on enable button
Preconditions	PRE-1: Admin is logged in PRE-2: Admission session exists or is disabled in order to be enabled
Postconditions	POST-1: Admission sessions status will be enabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Admin clicks on the Enable button next to the respective admission session they desire to enable 2. System enables the admission session 3. System displays the success message indicating the session enabled
Alternative Flows	N/A

Exceptions	<p>3a. If the end date for the session has passed the session status will be disabled by default and will not be enabled</p> <ol style="list-style-type: none"> 1. The use case will be terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.30 UC-32: Disable Admission Session

Use Case ID	UC-32
Use Case Name	Disable Admission Session
Actors	Primary Actor : Admin
Description	The Admin can disable the status of Admission Session by id
Trigger	The Admin will click on disable button
Preconditions	PRE-1: Admin is logged in PRE-2: Admission session is disabled or exists in order to be enabled
Postconditions	POST-1: Admission sessions status will be disabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Admin clicks on the Disable button next to the respective admission session which is enabled 2. System disables the admission session 3. System displays the success message indicating the session has been disabled
Alternative Flows	N/A
Exceptions	<p>3a. If the end date for the session has passed the session status will be disabled by default</p> <ol style="list-style-type: none"> 2. The use case will be terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.31 UC-33: Remove Admission Session

Use Case ID	UC-33
Use Case Name	Remove Admission Session
Actors	Primary Actor : Admin
Description	The Admin can remove the Admission Session by id
Trigger	The Admin will click on remove button
Preconditions	PRE-1: Admin is logged in PRE-2: Admission session exists in order to be removed
Postconditions	POST-1: Admission sessions will be removed POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Admin desires to remove the session 2. Admin clicks on remove session 3. System will display a message to confirm in order to remove a session 4. Admin confirms to remove the session 5. System removes the session
Alternative Flows	N/A
Exceptions	<p>3a. Admin does not confirm to remove the session</p> <p>4. The use case will be terminated</p>
Business Rules	N/A
Assumptions	N/A

3.2.2.32 UC-34: Apply for Admission

Use Case ID	UC-34
Use Case Name	Apply for Admission
Actors	Primary Actor: Applicant
Description	The Applicant can apply for the admissions once the Apply Now option is visible on the landing page. Applicants can also view the announcement to check for open registrations.
Trigger	The Applicant will click on Apply Now button
Preconditions	PRE-1: Applicant is not logged in yet PRE-2: Admission session is enabled and created

	PRE-3: Applicant has not already applied for a program
Postconditions	<p>POST-1: Applicant will be registered and will be able to select a program they desire to proceed with</p> <p>POST-2: Success message is displayed</p>
Normal Flow	<ol style="list-style-type: none"> 1. Applicant clicks on Apply Now Button 2. Applicant fills all the required fields 3. Applicant clicks on register now 4. Applicant is redirected to registration page where they can register into the session 5. System will redirect the applicant after registration to their dashboard
Alternative Flows	N/A
Exceptions	<p>2a. Applicant has missed any of the fields</p> <ol style="list-style-type: none"> 1. System will request applicant to fill in all the required fields 2. Applicant re-enters the missing field 3. System will proceed from step 3 of normal flow
Business Rules	N/A
Assumptions	N/A

3.2.2.33 UC-35: Choose a Program

Use Case ID	UC-35
Use Case Name	Choose a Program
Actors	Primary Actor: Applicant
Description	Applicants can choose more than one program to apply for, they can select the list of programs and click on proceed.
Trigger	The Applicant will select the list of programs from the checkboxes
Preconditions	<p>PRE-1: Applicant is logged in</p> <p>PRE-2: Admission session is enabled and created</p> <p>PRE-3: Applicant has not already applied for a program</p> <p>PRE-4: Applicant has not yet selected any program and not yet clicked on proceed now button.</p>
Postconditions	POST-1: List of programs will be added to applicants Applied Programs section
Normal Flow	<ol style="list-style-type: none"> 1. Applicant chooses the program they desire to apply for

	<ol style="list-style-type: none"> 2. Applicant clicks on the checkboxes to select the program 3. System assigns the programs to them, for application process
Alternative Flows	N/A
Exceptions	<p>2a. Applicant has already applied for the program</p> <ol style="list-style-type: none"> 1. The use case gets terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.34 UC-36: Add Personal Details

Use Case ID	UC-36
Use Case Name	Add Personal Details
Actors	Primary Actor: Applicant
Description	Applicants can add their personal details once they have applied for a program.
Trigger	The Applicant will click on proceed button after choosing program they want to apply for
Preconditions	PRE-1: Applicant is logged in PRE-2: Admission session is enabled and created PRE-3: Applicant has not already applied for a program
Postconditions	POST-1: Applicants personal details will be updated
Normal Flow	<ol style="list-style-type: none"> 1. Applicant clicks on proceed button after choosing the list of programs they desire to get enrolled in 2. System displays the personal details form 3. Applicant adds their personal details 4. Applicant clicks next button 5. System continues to the next form for Income details
Alternative Flows	N/A
Exceptions	<p>1a. Applicant has already applied for the program</p> <ol style="list-style-type: none"> 1. The use case gets terminated
Business Rules	N/A

Assumptions	N/A
--------------------	-----

3.2.2.35 **UC-37: Add Income Details**

Use Case ID	UC-37
Use Case Name	Add Income Details
Actors	Primary Actor: Applicant
Description	Applicants can add their income details once they have applied for a program.
Trigger	The Applicant will click on next button after completing the personal details section
Preconditions	PRE-1: Applicant is logged in PRE-2: Applicant has completed their personal details
Postconditions	POST-1: Applicants income details will be updated
Normal Flow	<ol style="list-style-type: none"> 1. Applicant clicks on next button after filling in their personal details 2. System displays the income details form 3. Applicant adds their income details 4. Applicant clicks next button 5. System continues to the next form for Education details
Alternative Flows	N/A
Exceptions	<p>1a. Applicant has already applied for the program</p> <ol style="list-style-type: none"> 1. The use case gets terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.36 **UC-38: Add Education Details**

Use Case ID	UC-38
Use Case Name	Add Education Details
Actors	Primary Actor: Applicant
Description	Applicants can add their education details once they have applied for a

	program.
Trigger	The Applicant will click on next button after completing the education details section
Preconditions	PRE-1: Applicant is logged in PRE-2: Applicant has completed their income details
Postconditions	POST-1: Applicants education details will be updated
Normal Flow	<ol style="list-style-type: none"> 1. Applicant clicks on next button after filling in their income details 2. System displays the education details form 3. Applicant adds their education details 4. Applicant clicks next button 5. System displays a message to confirm submission of the details 6. Applicant confirms the submission 7. System redirects the Applicant to their dashboard
Alternative Flows	N/A
Exceptions	<p>1a. Applicant has already applied for the program</p> <ol style="list-style-type: none"> 1. The use case gets terminated
Business Rules	N/A
Assumptions	N/A

3.2.2.37 UC-39: Upload Submitted Challan Image

Use Case ID	UC-39
Use Case Name	Upload Submitted Challan Image
Actors	Primary Actor : Applicant
Description	Applicants can upload their submitted challan picture which can be viewed by the Admin for verification purposes.
Trigger	The Applicant will click on upload button
Preconditions	PRE-1: Applicant is logged in PRE-2: Applicant paid the fee amount
Postconditions	POST-1: Applicant submitted challan can be viewed by the admin
Normal Flow	<ol style="list-style-type: none"> 1. Applicant chooses the picture of submitted challan form 2. Applicant clicks on upload fee challan 3. System uploads the submitted challan picture with the applicant details

	4. Admin can view the status of submitted challan on Manage Applicant section
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.38 UC-40: View Challan Status

Use Case ID	UC-40
Use Case Name	View Challan Status
Actors	Primary Actor: Admin
Description	Admin can click on the status button on the Manage List section next to applicant, details in order to view their status for the fee submission.
Trigger	The Admin will click on view button
Preconditions	PRE-1: Admin is logged in PRE-2: Applicant paid the fee challan PRE-3: Applicant is forwarded
Postconditions	POST-1: Admin can verify or discard the pending applicant
Normal Flow	<ol style="list-style-type: none"> 1. Admin clicks on status button 2. System displays the image of the challan submitted by the applicant 3. Admin clicks on save button and proceeds to verify applicant
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.2.39 UC-41: View Applicants Status

Use Case ID	UC-41
--------------------	-------

Use Case Name	View Applicants Status
Actors	Primary Actor: Coordinator
Description	Coordinator can view the Applicants status by visiting the Manage Applicant section
Trigger	Admin will click on verify applicant
Preconditions	PRE-1: Coordinator is logged in PRE-2: Admin has forwarded the applicant
Postconditions	POST-1: Coordinator can view and generate merit list based on the applicant details.
Normal Flow	<ol style="list-style-type: none"> 1. Coordinator visits manage applicant section 2. System will display the applicant status as verified or pending 3. Coordinator can proceed on registering the student
Alternative Flows	<p>2a. Applicant status is pending</p> <ol style="list-style-type: none"> 1. Coordinator can wait for Admin to verify applicant for registration 2. The use case will proceed from step 3 of normal flow
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.3 Use Cases of Module-2:

3.2.3.1 UC-42: Create Course

Use Case ID	UC-42
Use Case Name	Create Course
Actors	Primary Actor: Coordinator
Description	The coordinator will be able to create the course
Trigger	The coordinator will click on create course
Preconditions	PRE-1: Coordinator is logged in PRE-2: Coordinator is authorized to create the course PRE-3: Coordinator is on manage course section
Postconditions	POST-1: Course is created

	<p>POST-2: Success message is displayed POST-3: Coordinator can add the courses to the sections POST-4: Coordinator can assign the course to the students</p>
Normal Flow	<ol style="list-style-type: none"> 1. The Coordinator selects add course 2. System shows the form to fill in for the creating course 3. The Coordinator completes the form and clicks on submit button 4. Created courses can be seen on the manage course tab and students can be assigned these courses.
Alternative Flows	<p>3a. If the course is already created</p> <ol style="list-style-type: none"> 1. Course can be updated 2. System continues from step 3 of normal flow
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.3.2 **UC-43: Update Course**

Use Case ID	UC-43
Use Case Name	Update Course
Actors	Primary Actor : Coordinator
Description	The Coordinator can update course after it has been created, by clicking on the update button next to the respective course
Trigger	The Coordinator will click on update button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Course exists in order to be updated
Postconditions	POST-1: Course details will be modified POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Coordinator clicks on update button 2. Coordinator updates any of the required fields of the course 3. Coordinator clicks on submit 4. System displays a success message indicating the fields have been updated successfully 5. System will redirect the Coordinator to manage course section 6. System will display the modifications made on the course details
Alternative Flows	N/A

Exceptions	<p>3a. If there are no fields that have been updated</p> <ol style="list-style-type: none"> 1. System shall display the message indicating that no change has been done 2. The use case will be terminated
Business Rules	N/A
Assumptions	N/A

3.2.3.3 *UC-44: Enable Course*

Use Case ID	UC-44
Use Case Name	Enable Course
Actors	Primary Actor : Coordinator
Description	The Coordinator can enable the status of course by id
Trigger	The Coordinator will click on enable button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Course exists or is disabled in order to be enabled
Postconditions	POST-1: Course status will be enabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. The Coordinator clicks on the Enable button next to the respective course they desire to enable 2. System enables the course 3. System displays the success message indicating the course enabled
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.3.4 UC-45: Disable Course

Use Case ID	UC-45
Use Case Name	Disable Course
Actors	Primary Actor : Coordinator
Description	The Coordinator can disable the status of course by id
Trigger	The Coordinator will click on disable button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Course is disabled or exists in order to be enabled
Postconditions	POST-1: Course status will be disabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 4. The Coordinator clicks on the Disable button next to the respective course which is enabled 5. System disables the course 6. System displays the success message indicating the course has been disabled
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.3.5 UC-46: Remove Course

Use Case ID	UC-46
Use Case Name	Remove Course
Actors	Primary Actor : Coordinator
Description	The Coordinator can remove the Course by id
Trigger	The Coordinator will click on remove button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Course exists in order to be removed
Postconditions	POST-1: Course will be removed POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Coordinator desires to remove the course 2. Coordinator clicks on remove button next to a course

	<ol style="list-style-type: none"> 3. System will display a message to confirm in order to remove a course 4. Coordinator confirms to remove the course 5. System removes the course
Alternative Flows	N/A
Exceptions	<p>3a. Admin does not confirm to remove the course</p> <p>5. The use case will be terminated.</p>
Business Rules	N/A
Assumptions	N/A

3.2.3.6 **UC-47: View Course Status**

Use Case ID	UC-47
Use Case Name	View Course Status
Actors	Primary Actor : Coordinator
Description	Coordinator can view the Course status by visiting the Manage Course section
Trigger	Coordinator will click on manage course section
Preconditions	PRE-1: Coordinator is logged in PRE-2: Coordinator has course created
Postconditions	POST-1: Coordinator can view and change the course status as offered or not
Normal Flow	<ol style="list-style-type: none"> 1. Coordinator visits manage course section 2. System will display the course status as offered or not offered 3. Coordinator can proceed on changing status of the course
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A

Assumptions	N/A
--------------------	-----

3.2.4 Use Cases of Module-3:

3.2.4.1 UC-48: Enroll Students in Semester

Use Case ID	UC-48
Use Case Name	Enroll Students in Semester
Actors	Primary Actor : Coordinator
Description	The Coordinator will be able to enroll Students into semester
Trigger	The Coordinator will click on manage course button next to an Applicant
Preconditions	PRE-1: Coordinator is logged in PRE-2: Coordinator is authorized to enroll student PRE-3: Coordinator is on manage student section
Postconditions	POST-1: Applicant is registered as Student POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> Coordinator will navigate to Manage Applicant section Coordinator will click on manage course button System will display the list of courses Coordinator will choose the desired courses Coordinator will click on save System will enroll the students into the Semester depending on the courses they choose to study
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.4.2 UC-49: Select a Course in Semester

Use Case ID	UC-49
Use Case Name	Select a Course in Semester
Actors	Primary Actor : Coordinator
Description	Coordinator can choose courses to add to the Students for which they will be taught in the respective sections
Trigger	The Coordinator will select the list of courses from the Manage Course
Preconditions	PRE-1: Coordinator is logged in PRE-2: Applicant has been registered as a student
Postconditions	POST-1: List of programs will be added to Students Assigned Course section
Normal Flow	<ol style="list-style-type: none"> Coordinator visits the Manage Students section Coordinator clicks on manage course next to each applicant Coordinator chooses the courses they desire to assign the student Coordinator clicks on save System assigns the courses to students, for application process
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.4.3 UC-50: Create Semester Enrollment

Use Case ID	UC-50
Use Case Name	Create Semester Enrollment
Actors	Primary Actor : Coordinator
Description	The coordinator will be able to create the semester for enrollment of students with respect to their courses and sections.
Trigger	The admin will click on the create enrollment semester button.
Preconditions	PRE-1: Coordinator is logged in PRE-2: Coordinator is authorized to create enrollment semester.
Postconditions	POST-1: The Enrollment semester is created. POST-2: The Enrollment semester is enabled by default. POST-3: The Applicant can be enrolled into a semester.
Normal Flow	<ol style="list-style-type: none"> Coordinator selects the create enrollment semester option System shows a form with the fields to create enrollment semester

	<ol style="list-style-type: none"> 3. Coordinator enters the details 4. Coordinator clicks on submit 5. Enrollment semester gets created on the dashboard 6. Enrollment semester is enabled by default 7. Students can be assigned to the semester
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.4.4 UC-51: Update Enrollment semester

Use Case ID	UC-51
Use Case Name	Update Enrollment semester
Actors	Primary Actor : Coordinator
Description	The Coordinator can update Enrollment semester after it has been created, by clicking on the update button next to the respective semester
Trigger	The Coordinator will click on update button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Enrollment semester exists in order to be updated
Postconditions	POST-1: Enrollment semester details will be modified POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 1. Coordinator clicks on update button 2. Coordinator updates any of the required fields of the Enrollment semester 3. Coordinator clicks on submit 4. System displays a success message indicating the fields have been updated successfully 5. System will redirect the Coordinator to Dashboard 6. System will display the modifications made on the semester details
Alternative Flows	N/A
Exceptions	<p>3a. If there are no fields that have been updated</p> <ol style="list-style-type: none"> 1. System shall display the message indicating that no change has been done 2. The use case will resume from step 4 of normal flow
Business Rules	N/A

Assumptions	N/A
--------------------	-----

3.2.4.5 **UC-52: Enable Enrollment semester**

Use Case ID	UC-52
Use Case Name	Enable Enrollment semester
Actors	Primary Actor : Coordiantor
Description	The Coordiantor can enable the status of Enrollment semester by id
Trigger	The Coordiantor will click on enable button
Preconditions	PRE-1: Coordiantor is logged in PRE-2: Enrollment semester exists or is disabled in order to be enabled
Postconditions	POST-1: Enrollment semester status will be enabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 4. The Coordiantor clicks on the Enable button next to the respective Enrollment semester they desire to enable 5. System enables the Enrollment semester 6. System displays the success message indicating the semester enabled
Alternative Flows	N/A
Exceptions	<p>3a. If the end date for the semester has passed the semester status will be disabled by default and will not be enabled</p> <p>3. The use case will be terminated</p>
Business Rules	N/A
Assumptions	N/A

3.2.4.6 **UC-53: Disable Enrollment semester**

Use Case ID	UC-53
Use Case Name	Disable Enrollment semester
Actors	Primary Actor : Coordinator
Description	The Coordinator can disable the status of Enrollment semester by id

Trigger	The Coordinator will click on disable button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Enrollment semester is disabled or exists in order to be enabled
Postconditions	POST-1: Enrollment semester status will be disabled POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 4. The Coordinator clicks on the Disable button next to the respective Enrollment semester which is enabled 5. System disables the Enrollment semester 6. System displays the success message indicating the semester has been disabled
Alternative Flows	N/A
Exceptions	<p>3a. If the end date for the session has passed the semester status will be disabled by default</p> <p>4. The use case will be terminated</p>
Business Rules	N/A
Assumptions	N/A

3.2.4.7 **UC-54: Update Section**

Use Case ID	UC-54
Use Case Name	Update Section
Actors	Primary Actor : Coordinator
Description	The Coordinator will be able to update the Section fields
Trigger	The Coordinator will click on update Section button
Preconditions	PRE-1: Coordinator is logged in PRE-2: Coordinator is authorized to update the Section PRE-3: Coordinator is on manage Section
Postconditions	POST-1: Section fields are updated POST-2: Success message is displayed
Normal Flow	<ol style="list-style-type: none"> 7. Coordinator will click on update Section button 8. System will display the admin the fields of Section to be updated 9. Coordinator will update the required fields 10. Coordinator will click on submit 11. System will display success message indicating Section fields are updated

	12. Updated fields will be seen on manage Section.
Alternative Flows	N/A
Exceptions	<p>3a. If there are no fields that have been updated</p> <p>9. System shall display the message indicating that no change has been done</p> <p>10. The use case will be terminated</p>
Business Rules	N/A
Assumptions	N/A

3.2.4.8 **UC-55: Register Applicant as Student**

Use Case ID	UC-55
Use Case Name	Register Applicant as Student
Actors	Primary Actor : Coordinator
Description	The Coordinator will be able to register a Student
Trigger	The Coordinator will click on register button next to an Applicant
Preconditions	<p>PRE-1: Coordinator is logged in</p> <p>PRE-2: Coordinator is authorized to register coordinator</p> <p>PRE-3: Coordinator is on manage student section</p>
Postconditions	<p>POST-1: Applicant is registered as Student</p> <p>POST-2: Success message is displayed</p>
Normal Flow	<ol style="list-style-type: none"> Coordinator will navigate to Manage Applicant section Coordinator will click on register button System will display the registration form Coordinator will fill in the registration form Coordinator will click on submit System will display a success message Registered Student list can be seen on the Manage Students section
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A

Assumptions	N/A
--------------------	-----

3.2.4.9 **UC-56: Create Section**

Use Case ID	UC-56
Use Case Name	Create Section
Actors	Primary Actor : Coordinator
Description	The Coordinator will be able to create the Section
Trigger	The Coordinator will click on create Section
Preconditions	PRE-1: Coordinator is logged in PRE-2: Coordinator is authorized to create the Section PRE-3: Coordinator is on manage Section
Postconditions	POST-1: Section is created POST-2: Success message is displayed POST-3: Courses can be added to the Section
Normal Flow	<ol style="list-style-type: none"> 1. The Coordinator selects create Section 2. System shows the form to fill in the Section 3. The Coordinator completes the form and clicks on submit button 4. System display the success message and displays the created Section
Alternative Flows	3a. If the Section is already created <ol style="list-style-type: none"> 1. Section can be updated 2. System continues from step 3 of normal flow
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.5 Use Cases of Module-4:

3.2.5.1 **UC-57: Generate Timetable**

Use Case ID:	UC-57
Use Case Name:	Generate Timetable
Actors:	Primary Actor: Admin
Description:	The Admin will be able to generate a timetable and set classroom, time slot and course name
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: A new timetable slot will be created
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will click on create new slot 3: Admin will enter all the fields on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.5.2 UC-58: *Change classroom*

Use Case ID:	UC-58
Use Case Name:	Change classroom
Actors:	Primary Actor: Admin
Description:	The Admin will be able to change classroom details in timetable
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: A new classroom will be allotted
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will click on edit icon. 3: Admin will enter all the fields on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.5.3 UC-59: Change time slot

Use Case ID:	UC-59
Use Case Name:	Change time slot
Actors:	Primary Actor: Admin
Description:	The Admin will be able to change time slot details in timetable
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: A new timetable slot will be set
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will edit icon 3: Admin will enter all the fields on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.5.4 UC-60: Change Event

Use Case ID:	UC-60
Use Case Name:	Change Event
Actors:	Primary Actor: Admin
Description:	The Admin will be able to change the event or course name for which they are setting the schedule
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: A new timetable slot will be set
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will edit icon 3: Admin will enter all the fields on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.5.5 ***UC-61: Delete classroom***

Use Case ID:	UC-61
Use Case Name:	Delete classroom
Actors:	Primary Actor: Admin
Description:	The Admin will be able to delete classroom details in timetable
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: That Classroom detail will be deleted
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will click on delete icon. 3: Admin will delete the particular field on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.5.6 ***UC-62: Delete classroom***

Use Case ID:	UC-62
Use Case Name:	Delete classroom
Actors:	Primary Actor: Admin
Description:	The Admin will be able to delete timeslot details in timetable
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: That Classroom detail will be deleted
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will click on delete icon. 3: Admin will delete the particular field on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.5.7 UC-63: Delete Event

Use Case ID:	UC-63
Use Case Name:	Delete Event
Actors:	Primary Actor: Admin
Description:	The Admin will be able to delete Event details in timetable
Trigger:	The admin will be able to click on the desired options to make changes
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in schedule
Postconditions:	POST-1: That Classroom detail will be deleted
Normal Flow:	1: Admin will go to schedule tab in dashboard. 2: Admin will click on delete icon. 3: Admin will delete the particular field on a form 4: Admin will save the details
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6 Use Cases of Module-5:

3.2.6.1 UC-64: Add Assignment Marks

Use Case ID:	UC-64
Use Case Name:	Add Assignment Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to add marks to the students for each subject's assignments
Trigger:	The Faculty member clicks on the "Mark Grades" option and then select assignment option
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in assignment marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the assignment tab

	4. Enter marks of each or particular subject 5. Then click on save button. 6. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.2 ***UC-65: Edit Assignment Marks***

Use Case ID:	UC-65
Use Case Name:	Edit Assignment Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to edit marks to the students for each subject's assignments
Trigger:	The Faculty member clicks on the “Mark Grades” option and then select assignment option and click on edit icon
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in assignment marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the assignment tab 4. Then click on edit icon 5. Enter marks of each or particular subject 6. Then click on save button. 7. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.3 ***UC-66: Add quiz Marks***

Use Case ID:	UC-66
---------------------	-------

Use Case Name:	Add quiz Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to add marks to the students for each subject's quiz marks
Trigger:	The Faculty member clicks on the "Mark Grades" option and then select quiz option
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in quiz marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the quiz tab 4. Enter marks of each or particular subject 5. Then click on save button. 6. System displays the message "Save Successful".
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.4 **UC-67: Edit quiz Marks**

v

Use Case ID:	UC-67
Use Case Name:	Edit quiz Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to edit marks to the students for each subject's quiz
Trigger:	The Faculty member clicks on the "Mark Grades" option and then select quiz option and click on edit icon
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in quiz marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the quiz tab 4. Then click on edit icon 5. Enter marks of each or particular subject 6. Then click on save button. 7. System displays the message "Save Successful".

Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.5 UC-68: Add midterm Marks

Use Case ID:	UC-68
Use Case Name:	Add midterm Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to add marks to the students for each subject's midterm marks
Trigger:	The Faculty member clicks on the “Mark Grades” option and then select midterm option
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in midterm marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the midterm tab 4. Enter marks of each or particular subject 5. Then click on save button. 6. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.6 UC-69: Edit midterm Marks

Use Case ID:	UC-69
Use Case Name:	Edit midterm Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to edit marks to the students for each subject's midterm

Trigger:	The Faculty member clicks on the “Mark Grades” option and then select midterm option and click on edit icon
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in midterm marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the midterm tab 4. Then click on edit icon 5. Enter marks of each or particular subject 6. Then click on save button. 7. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.7 **UC-70: Add project Marks**

Use Case ID:	UC-70
Use Case Name:	Add project Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to add marks to the students for each subject’s project marks
Trigger:	The Faculty member clicks on the “Mark Grades” option and then select project option
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in project marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the project tab 4. Enter marks of each or particular subject 5. Then click on save button. 6. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.8 UC-71: Edit project Marks

Use Case ID:	UC-71
Use Case Name:	Edit project Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to edit marks to the students for each subject's project
Trigger:	The Faculty member clicks on the "Mark Grades" option and then select project option and click on edit icon
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in midterm marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the project tab 4. Then click on edit icon 5. Enter marks of each or particular subject 6. Then click on save button. 7. System displays the message "Save Successful".
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.9 UC-72: Add final Marks

Use Case ID:	UC-72
Use Case Name:	Add final Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to add marks to the students for each subject's final marks
Trigger:	The Faculty member clicks on the "Mark Grades" option and then select final option
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in project marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter

	<ol style="list-style-type: none"> 3. Then select the final tab 4. Enter marks of each or particular subject 5. Then click on save button. 6. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.6.10 ***UC-73: Edit final Marks***

Use Case ID:	UC-73
Use Case Name:	Edit final Marks
Actors:	Primary Actor: Faculty
Description:	The Faculty will be able to edit marks to the students for each subject’s final
Trigger:	The Faculty member clicks on the “Mark Grades” option and then select final option and click on edit icon
Preconditions:	PRE-1: Logged into the system PRE-2: Have access to make changes in midterm marks
Postconditions:	POST-1: The marks will be uploaded and viewable by Faculty.
Normal Flow:	<ol style="list-style-type: none"> 1. Faculty is logged into the CMS. 2. Faculty will select particular semester they want to enter 3. Then select the final tab 4. Then click on edit icon 5. Enter marks of each or particular subject 6. Then click on save button. 7. System displays the message “Save Successful”.
Alternative Flows:	N/A.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.7 Use Cases of Module-6:

3.2.7.1 ***UC-74: Add student Record***

Use Case ID:	UC-74
Use Case Name:	Add student Record
Actors:	Primary Actor: Admin
Description:	The user needs to add a record.
Trigger:	The user selects add record button.
Preconditions:	PRE-1: User is logged in the portal.
Postconditions:	POST-1: Record is added in the database. POST-2: User is displayed a message “Record added Successfully”.
Normal Flow:	<ol style="list-style-type: none"> 1. User is shown a page with an add record button. 2. User selects an option of student, faculty or employees whoever he/she wants to add 3. User enters the credentials to add. 4. System verifies the information added. 5. System verifies the record doesn't exist already. 6. Record is added in the database. 7. Admin is displayed a message “Record added successfully.”.
Alternative Flows:	<p>4a . If the system doesn't verify the credentials added.</p> <ol style="list-style-type: none"> 1. User is displayed a message “Incorrect information.” 2. The fields with wrong information are colored red. 3. Normal flow continues from point 3. <p>5a. The record already exists.</p> <ol style="list-style-type: none"> 1. System generates a message “Record already exists” 2. System exists the use case.
Exceptions:	N/A
Business Rules	N/A
Assumptions:	N/A

3.2.7.2 UC-75: *Modify Record*

Use Case ID:	UC-75
Use Case Name:	Modify Record
Actors:	Primary Actor: Admin, Coordinator
Description:	The user needs to update an existing record.
Trigger:	The user selects modify record button.
Preconditions:	PRE-1: User is logged in the portal.
Postconditions:	POST-1: User is provided the option to update or delete the record selected.
Normal Flow:	<ol style="list-style-type: none"> 1. User is shown a page with a search option to search the record. 2. User selects an option of student 3. User enters the credentials through which he/she wants to search the record. 4. Record is displayed to the user.

	5. Option of update and delete is provided beneath the record searched..
Alternative Flows:	3a .If the user enters wrong credentials, or such credentials which do not exist in the database. 1. User is displayed a message “ Record doesn’t exist. Try again.” 2. User is prompted to enter other credentials. 3. Normal flow continues from point 2.
Exceptions:	N/A.
Business Rules	N/A.
Assumptions:	N/A.

3.2.7.3 ***UC-76: Delete Record***

Use Case ID:	UC-76
Use Case Name:	Delete Record
Actors:	Primary Actor: Admin
Description:	The user wants to delete the record of a student
Trigger:	User selects “Delete” button.
Preconditions:	PRE-1: A record is displayed which the user searched.
Postconditions:	POST-1: Record is deleted from the database. POST-2: Modify page is shown to the user.
Normal Flow:	1. User is asked for confirmation by showing a message “Are you sure you want to delete the record?” 2. User selects yes. 3. Record is deleted from the database. 4. User is shown message “Record Deleted Successfully” 5. User is shown the modify page.
Alternative Flows:	2a. User selects no 1. User is shown the record with the update and delete option.
Exceptions:	N/A
Business Rules	N/A
Assumptions:	N/A

3.2.7.4 ***UC-77: View Profile***

Use Case ID:	UC-77
Use Case Name:	View Profile
Actors:	Primary Actor: Admin,

Description:	The admin wants to view the student profile
Trigger:	User selects the display student button.
Preconditions:	PRE-1: User is logged in the system.
Postconditions:	POST-1: Student profile is displayed
Normal Flow:	<ol style="list-style-type: none"> 1. Log in the system 2. Select profiles option 3. Click on the profile you want to display
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	N/A

3.2.7.5 *UC-78: Display students*

Use Case ID:	UC-78
Use Case Name:	Display students
Actors:	Primary Actor: Admin, Coordinator
Description:	The admin wants to view the list of students in a semester.
Trigger:	User selects the display students button.
Preconditions:	PRE-1: User is logged in the system.
Postconditions:	POST-1: User is displayed a list of students..
Normal Flow:	<ol style="list-style-type: none"> 4. User is prompted to enter the semester no# of which he wants to view the students. 5. User enters the semester number. 6. User is displayed the list of students who are eligible for that semester
Alternative Flows:	N/A
Exceptions:	N/A
Business Rules	N/A
Assumptions:	N/A

3.2.8 Use Cases of Module-7:

3.2.8.1 *UC-79: Register Faculty*

Use Case ID	UC-79
Use Case Name	Register Faculty

Actors	Primary Actor: Admin
Description	The admin will be able to register a faculty member..
Trigger	The user clicks on “Register Faculty “ button
Preconditions	PRE-1: The faculty member has been hired
Postconditions	POST-1: The faculty member is registered.
Normal Flow	<ol style="list-style-type: none"> 1. The admin selects on register faculty button. 2. The admin is displayed a form with fields required to register the faculty. 3. The admin fills the forms and submits. 4. The system registers the faculty.
Alternative Flows	<p>3a. In step 3 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.2 **UC-80: Create Faculty**

Use Case ID	UC-80
Use Case Name	Create Faculty
Actors	Primary Actor: Faculty
Description	The faculty will be able to create his/her profile.
Trigger	The user clicks on “Create Profile “ button
Preconditions	PRE-1: The faculty member has logged in PRE-2: User doesn't have a profile.
Postconditions	POST-1: The faculty member's profile is created
Normal Flow	<ol style="list-style-type: none"> 1. The system displays a form with fields to be filled 2. The user fills the form 3. The user submits. 4. The system creates a profile.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A

Business Rules	N/A
Assumptions	N/A

3.2.8.3 *UC-81: Create Profile*

Use Case ID	UC-81
Use Case Name	Create Profile
Actors	Primary Actor: Faculty
Description	The faculty will be able to create his/her profile.
Trigger	The user clicks on “Create Profile “ button
Preconditions	PRE-1: The faculty member has logged in PRE-2: User doesn’t have a profile.
Postconditions	POST-1: The faculty member’s profile is created
Normal Flow	<ol style="list-style-type: none"> 1. The system displays a form with fields to be filled 2. The user fills the form 3. The user submits. 4. The system creates a profile.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.4 *UC-82: Edit Profile*

Use Case ID	UC-82
Use Case Name	Edit Profile
Actors	Primary Actor: Faculty
Description	The faculty will be able to edit his/her profile.
Trigger	The user clicks on “Edit Profile “ button
Preconditions	PRE-1: The faculty member has logged in PRE-2: User has a profile.

Postconditions	POST-1: The faculty member's profile is updated
Normal Flow	<ol style="list-style-type: none"> 1. The system displays a form with fields filled with previous data 2. The user fills the form 3. The user submits. 4. The system updates a profile.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 3. The system prompts the user to enter the required fields. 4. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.5 **UC-83: View Profile**

Use Case ID	UC-83
Use Case Name	View Profile
Actors	Primary Actor: Faculty
Description	The faculty will be able to view his/her profile.
Trigger	The user clicks on “Profile” button
Preconditions	PRE-1: The faculty member has logged in PRE-2: User has profile.
Postconditions	POST-1: The faculty member's profile is shown
Normal Flow	<ol style="list-style-type: none"> 1. User selects profile button on dashboard 2. User is shown his/her profile .
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.6 **UC-84: View Profiles of All Faculty**

Use Case ID	UC-84
Use Case Name	View Profiles of All Faculty
Actors	Primary Actor: Faculty , Admin

Description	The user will be able to view the profiles of all the faculty members.
Trigger	The user clicks on “Faculty Members “ button
Preconditions	PRE-1: The user has logged in
Postconditions	POST-1: The faculty members profile is shown
Normal Flow	<ol style="list-style-type: none"> 1. User selects “Faculty Member” button on dashboard 2. User is shown profiles of all faculty members..
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.7 *UC-85: Delete Profile*

Use Case ID	UC-85
Use Case Name	Delete Profile
Actors	Primary Actor: Admin
Description	The user will be able to delete a profile
Trigger	The user clicks on “Delete Profile “ button
Preconditions	PRE-1: The profiles are displayed
Postconditions	POST-1: The faculty member’s profile is deleted
Normal Flow	<ol style="list-style-type: none"> 1. User selects Delete button of member’s profile. 2. Profile is deleted.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.8 *UC-86: Add Research Paper*

Use Case ID	UC-86
Use Case Name	Add Research Paper
Actors	Primary Actor: Faculty
Description	The faculty will be able to add research papers
Trigger	The user clicks on “Add research Papers “ button
Preconditions	PRE-1: The faculty member has logged in

	PRE-2: User has a profile.
Postconditions	POST-1: The faculty member's profile is update
Normal Flow	<ol style="list-style-type: none"> 1. The system displays a form with fields to be filled 2. The user fills the form 3. The user submits. 4. The system updates a profile.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.9 *UC-87: Add Education*

Use Case ID	UC-87
Use Case Name	Add Education
Actors	Primary Actor: Faculty
Description	The faculty will be able to add education
Trigger	The user clicks on “Add education “ button
Preconditions	PRE-1: The faculty member has logged in PRE-2: User has a profile.
Postconditions	POST-1: The faculty member's profile is update
Normal Flow	<ol style="list-style-type: none"> 1. The system displays a form with fields to be filled 2. The user fills the form 3. The user submits. 4. The system updates a profile.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.10 UC-88: Add Experience

Use Case ID	UC-88
Use Case Name	Add Experience
Actors	Primary Actor: Faculty
Description	The faculty will be able to add experience
Trigger	The user clicks on “Add experience“ button
Preconditions	PRE-1: The faculty member has logged in PRE-2: User has a profile.
Postconditions	POST-1: The faculty member’s profile is update
Normal Flow	<ol style="list-style-type: none"> 1. The system displays a form with fields to be filled 2. The user fills the form 3. The user submits. 4. The system updates a profile.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.11 UC-89: View Calendar

Use Case ID	UC-89
Use Case Name	View Calendar
Actors	Primary Actor: Faculty
Description	The faculty will be able to view his/her calendar
Trigger	The user clicks on “Calendar “ button
Preconditions	PRE-1: The faculty member has logged in
Postconditions	POST-1: The faculty member’s calendar is shown
Normal Flow	<ol style="list-style-type: none"> 1. User selects calendar button on dashboard 2. User is shown his/her calendar .
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.12 UC-90: View Evaluation Result of Self

Use Case ID	UC-90
Use Case Name	View Evaluation Result of Self
Actors	Primary Actor: Faculty
Description	The faculty will be able to view his/her evaluation result
Trigger	The user clicks on “View Result “ button
Preconditions	PRE-1: The faculty member has logged in
Postconditions	POST-1: The faculty member’s evaluation result is shown
Normal Flow	<ol style="list-style-type: none"> 1. User selects profile button on dashboard 2. User is shown his/her result.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.13 UC-91: View Teaching Schedule

Use Case ID	UC-91
Use Case Name	View Teaching Schedule
Actors	Primary Actor: Faculty
Description	The faculty will be able to view his/her teaching schedule as assigned by admin.
Trigger	The user clicks on “Teaching Schedule “ button
Preconditions	PRE-1: The faculty member has logged in
Postconditions	POST-1: The faculty member’s profile is shown
Normal Flow	<ol style="list-style-type: none"> 1. User selects profile button on dashboard 2. User is shown his/her profile .
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.14 UC91: Add event

Use Case ID	UC-91
Use Case Name	Add event
Actors	Primary Actor: Faculty
Description	The faculty will be able to add an event to the calendar
Trigger	The user clicks on “Add event “ button
Preconditions	PRE-1: The faculty member has logged in
Postconditions	POST-1: The event is added
Normal Flow	<ol style="list-style-type: none"> 1. User selects add event button on calendar. 2. User enters the fields . 3. User submits. 4. System creates and event.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.15 UC-92: Delete event

Use Case ID	UC-92
Use Case Name	Delete event
Actors	Primary Actor: Faculty
Description	The faculty will be able to delete an event to the calendar
Trigger	The user clicks on Delete event “ button
Preconditions	PRE-1: The faculty member has logged in
Postconditions	POST-1: The event is deleted
Normal Flow	<ol style="list-style-type: none"> 1. User selects delete event button on calendar. 2. System delete an event.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.8.16 UC-93: View Evaluation Report of All Faculty

Use Case ID	UC-93
Use Case Name	View Evaluation Report of All Faculty
Actors	Primary Actor: Admin
Description	The user will be able to evaluation report of all the faculty members.
Trigger	The user clicks on “Evaluation Report “ button
Preconditions	PRE-1: The user has logged in
Postconditions	POST-1: The evaluation result is shown
Normal Flow	<ol style="list-style-type: none"> 1. User selects “Evaluation Report” button on dashboard 2. User is shown reports of all faculty members..
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.9 Use Cases of Module-8:

3.2.9.1 UC-94: Make a suggestion

Use Case ID	UC-94
Use Case Name	Make a suggestion
Actors	Primary Actor: Admin/ Faculty/ Coordinator /Applicant
Description	The user will be able to make a suggestion regarding the portal which will be sent to the email address of the university.
Trigger	User selects suggestion button
Preconditions	PRE-1: User is logged in PRE-2: User has opened the chatbot
Postconditions	POST-1: The suggestion mail is sent. POST-2: The system asks the user if he/she needs help with something else.
Normal Flow	<ol style="list-style-type: none"> 1. The user selects “Suggestion” button. 2. The system prompts the user to enter his/her name, email, suggestion. 3. The user enters the fields required 4. System asks the user if he/she wants to update 5. The user selects NO 6. The system asks the user if he/she wants to send mail. 7. The user select YES 8. The system send the mail

	<p>9. The system asks the user if he/she needs help with anything else.</p>
Alternative Flows	<p>In 4 of the normal flow , if the user wants to update a field and selects YES</p> <ol style="list-style-type: none"> 1. The system asks which field he/she wants to update. 2. The user selects the field to update. 3. System prompts the user to enter the updated value 4. The user enters the value to update to. 5. Normal flow continues from step 5.
Exceptions	N/A
Business Rules	N/A

3.2.9.2 UC-95: Make a complain

Use Case ID	UC-95
Use Case Name	Make a complain
Actors	Primary Actor: Admin/ Faculty/ Coordinator /Applicant
Description	The user will be able to make a complain regarding the portal which will be sent to the email address of the university.
Trigger	User selects suggestion button
Preconditions	PRE-1: User is logged in PRE-2: User has opened the chatbot
Postconditions	POST-1: The suggestion mail is sent. POST-2: The system asks the user if he/she needs help with something else.
Normal Flow	<ol style="list-style-type: none"> 1. The user selects “Complain” button. 2. The system prompts the user to enter his/her name, email, suggestion. 3. The user enters the fields required 4. System asks the user if he/she wants to update 5. The user selects NO 6. The system asks the user if he/she wants to send mail. 7. The user select YES 8. The system send the mail 9. The system asks the user if he/she needs help with anything else.
Alternative Flows	<p>In 4 of the normal flow , if the user wants to update a field and selects YES</p> <ol style="list-style-type: none"> 1. The system asks which field he/she wants to update. 2. The user selects the field to update. 3. System prompts the user to enter the updated value 4. The user enters the value to update to. 5. Normal flow continues from step 5.
Exceptions	N/A

Business Rules	N/A
Assumptions	N/A

3.2.9.3 ***UC-93: Find a Page***

Use Case ID	UC-96
Use Case Name	Find a Page
Actors	Primary Actor: Faculty , Admin, Coordinator , Applicant
Description	The user will be able to find a page through the chatbot
Trigger	User selects ‘Find a Page’ button in the chatbot.
Preconditions	PRE-1: User is logged in PRE-2: User is using the chatbot
Postconditions	POST-1: User is redirected to the page.
Normal Flow	<ol style="list-style-type: none"> 1. User selects ‘Find a Page’ button 2. System shows a set of pages. 3. User selects the page he/she wants to find 4. System redirects the user to the page.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.10 Use Cases of Module-9:

3.2.10.1 ***UC-97: Fill a Performa***

Use Case ID	UC-97
Use Case Name	Fill a Performa
Actors	Primary Actor: Faculty, Student
Description	The faculty will be able to fill a Performa in order to generate results.
Trigger	The user clicks on the proforma to be filled.
Preconditions	PRE-1: The user is logged in PRE-2: The Performa is open for filling
Postconditions	POST-1: The Performa is submitted
Normal Flow	<ol style="list-style-type: none"> 1. The user selects on the proforma wanting to be filed.

	<ol style="list-style-type: none"> 2. The user fills the fields given. 3. User submits.
Alternative Flows	<p>2a. In step 2 of normal flow , If the user leaves out an important or required field.</p> <ol style="list-style-type: none"> 1. The system prompts the user to enter the required fields. 2. The normal flow continues from step 2 with previous state
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.10.2 UC-98: *Generate Reports of Faculty*

Use Case ID	UC-98
Use Case Name	Generate Reports of Faculty
Actors	Primary Actor: Admin
Description	The admin will be able generate the results of students.
Trigger	The user clicks on “Generate Report of Student“ button
Preconditions	PRE-1: The proformas have been filled
Postconditions	POST-1: The report is generated
Normal Flow	<ol style="list-style-type: none"> 1. User selects on generate report of Students button 2. User selects the type of report to generate. 3. System generates the report.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.10.3 UC-99: *Generate Reports of Student*

Use Case ID	UC-99
Use Case Name	Generate Reports of Student
Actors	Primary Actor: Admin
Description	The admin will be able generate the results of students.
Trigger	The user clicks on “Generate Report of Faculty“ button
Preconditions	PRE-1: The proformas have been filled
Postconditions	POST-1: The report is generated
Normal Flow	<ol style="list-style-type: none"> 1. User selects on generate report of Faculty button

	<p>2. User selects the type of report to generate.</p> <p>3. System generates the report.</p>
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.10.4 UC-100: Upload Results

Use Case ID	UC-100
Use Case Name	Upload Results
Actors	Primary Actor: Admin
Description	The admin will be able to upload the results.
Trigger	The user clicks on “Upload Results“ button
Preconditions	PRE-1: The proformas have been filled PRE-2: The report is generated
Postconditions	POST-1: The result is uploaded
Normal Flow	1. User selects on upload result after a report has been generated. 2. The system uploads the result.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.10.5 UC-101: Change Graph Type

Use Case ID	UC-101
Use Case Name	Change Graph Type
Actors	Primary Actor: Admin
Description	The admin will be able to change the graph types of the reports generated.
Trigger	The user clicks on “Change Graph“ button
Preconditions	PRE-1: The reports have been generated
Postconditions	POST-1: The report graph is changed.
Normal Flow	1. The user selects change graph button 2. System asks the type of graph user wants 3. User selects the type he/she wants 4. The graph is changed and shown to the user.

Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.2.10.6 **UC-102: View Reports**

Use Case ID	UC-102
Use Case Name	View Reports
Actors	Primary Actor: Admin , Student, Faculty
Description	The user will be able to view the reports
Trigger	The user clicks on Reports button
Preconditions	PRE-1: The result has been uploaded
Postconditions	POST-1: The report is displayed
Normal Flow	<ol style="list-style-type: none"> 1. User selects the reports button 2. User is shown a list of reports that have been uploaded 3. User selects the report 4. Report is displayed.
Alternative Flows	N/A
Exceptions	N/A
Business Rules	N/A
Assumptions	N/A

3.3 Functional Requirements

3.3.1 FRS Module-1:

Identifier	FR-1
Title	Select Create Admission Session Button
Requirement	The admin shall be able to create an admission session.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to create the admission sessions
Business Rule (if required)	N/A
Dependencies	N/A

Priority	High
-----------------	------

Identifier	FR-2
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submits
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the Admission Session Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-3
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-4
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-5
-------------------	------

Title	Set date of Admissions
Requirement	The admin shall be able to select the set date of the admissions.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to select the date of the admissions..
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-6
Title	Enter Date
Requirement	The admin shall be able to enters the dates
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to select the date of the admissions..
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-7
Title	Display Date
Requirement	The admin shall be able to display the start and end Date
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to select the date of the admissions..
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-8
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-9
Title	Select Create Department Button
Requirement	The admin shall be able to create a department.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to create the department
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-10
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for creation of the department.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the Department Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-11
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-12
-------------------	-------

Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-13
Title	Select Update Department Button
Requirement	The admin shall be able to update a department.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to update the department
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-14
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for updation of the department.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-15
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.

Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-16
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-17
Title	View Department Coordinator List
Requirement	The Admin shall be able to view the list of departments coordinator
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to view the department coordinators list.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-18
Title	Display Department Coordinator List
Requirement	The System shall display the list of departments coordinators
Source	Raja Shaiyan Azad Khan
Rationale	To help user to see the list of department coordinators
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-19
Title	Display Department Programs List
Requirement	The System shall display the list of departments programs
Source	Raja Shaiyan Azad Khan
Rationale	To help user to see the list of department programs
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-20
Title	Display Department Program List
Requirement	The System shall display the list of departments programs
Source	Raja Shaiyan Azad Khan
Rationale	To help user to see the list of department programs
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-21
Title	Select Create Program Button
Requirement	The admin shall be able to create a program.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to create the program
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-22
-------------------	-------

Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for creation of the program.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the Program Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-23
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-24
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-25
Title	Select Update Program Button
Requirement	The admin shall be able to update a program.

Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to update the program
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-26
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for updatation of the program.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-27
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-28
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

required)	
Dependencies	N/A
Priority	High

Identifier	FR-29
Title	Select Enable Program Button
Requirement	The system shall change the status change of the selected program to enabled.
Source	Raja Shaiyan Azad Khan
Rationale	To allow applicants to apply into the specific program, being enabled.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-30
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-31
Title	Select Disable Program Button
Requirement	The system shall change the status change of the selected program to disabled.
Source	Raja Shaiyan Azad Khan
Rationale	To indicate that the current program is not being offered for application.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-32
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-33
Title	Select Remove Program Button
Requirement	The system shall remove the program from the program list
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to remove the program if it is not being offered any more.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-34
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-35
Title	Select Create Announcement Feed Button
Requirement	The admin shall be able to create an announcement feed.

Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to create the announcement feed.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-36
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for creation of the announcement feed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the announcement feed form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-37
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-38
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

required)	
Dependencies	N/A
Priority	High

Identifier	FR-39
Title	Select Update Announcement Feed Button
Requirement	The admin shall be able to update an announcement feed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to update the announcement feed.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-40
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for updation of the announcement feed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-41
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A

Dependencies	N/A
Priority	High

Identifier	FR-42
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-43
Title	Select Enable Button
Requirement	The system shall change the status change of the selected Announcement Feed to enabled.
Source	Raja Shaiyan Azad Khan
Rationale	To allow applicants to apply into the specific Announcement Feed, being enabled.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-44
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-45
Title	Select Disable Button
Requirement	The system shall change the status change of the selected Announcement Feed to disabled.
Source	Raja Shaiyan Azad Khan
Rationale	To indicate that the current Announcement Feed is not active.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-46
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-47
Title	Select Remove Button
Requirement	The system shall remove the feed from the feed list
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to remove the Announcement Feed if it is not important any more.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-48
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan

Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-49
Title	Select Add Coordinator Button
Requirement	The admin shall be able to add a coordinator.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to register a coordinator
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-50
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for registration of coordinator.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the Registration Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-51
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.

Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-52
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-53
Title	Select Remove Coordinator Button
Requirement	The system shall remove the coordinator from the coordinator list
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to remove the coordinator.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-54
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-55
Title	Select Generate Merit List Button
Requirement	The Admin shall be able to generate a list of students who are on the qualifying merit.
Source	Raja Shaiyan Azad Khan
Rationale	To allow applicants to get selected for registration.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-56
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-57
Title	Display Merit List
Requirement	The system shall be able to display the generated merit list of current session enabled.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to select the applicants that are on merit for registration purposes.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-58
Title	Select Add Test Score Button

Requirement	The admin shall be able to add the test score of each applicant.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the system to calculate aggregate of the students based on their test score
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-59
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Identifier	FR-60
Title	Select Generate Aggregate Button
Requirement	The admin shall be able to generate an aggregate after adding the applicant's test score button.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the system to calculate the aggregate of the applicant who scored in a test.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-61
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

required)	
Dependencies	N/A

Identifier	FR-62
Title	Display Aggregate
Requirement	The system shall be able to display the calculated aggregate of the applicant.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to forward applicants to the coordinator.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-63
Title	Select Forward Applicant Button
Requirement	The system shall be able forward applicants to the coordinator.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the coordinator to view applicants for verification status and registration purposes.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-64
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Identifier	FR-65
Title	Display Forwarded Applicant
Requirement	The system shall be able to display the forwarded applicant on the coordinators side.

Source	Raja Shaiyan Azad Khan
Rationale	To allow the coordinator to view the applicants that have been forwarded.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-66
Title	Select Verify Applicant Button
Requirement	The system shall be able to verify applicants in order to be registered as a student.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the coordinator to register applicants.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-67
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Identifier	FR-68
Title	Select Pending Button
Requirement	The system shall be able to set applicants status as pending.
Source	Raja Shaiyan Azad Khan
Rationale	To filter out the pending applicants that have not yet submitted the admission fee.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-69
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Identifier	FR-70
Title	Select Discard Button
Requirement	The system shall be able to move the applicants to discard applicants list.
Source	Raja Shaiyan Azad Khan
Rationale	To discard Applicants that have not yet paid the fee for the active admission session.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-71
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Identifier	FR-72
Title	Display Discarded Applicants List
Requirement	The system shall be able to display discarded applicants list.
Source	Raja Shaiyan Azad Khan
Rationale	To acquire the applicant if any of them submits their admission fee before the deadline.
Business Rule (if required)	N/A
Dependencies	N/A

Priority	High
-----------------	------

Identifier	FR-73
Title	Select Acquire Button
Requirement	The system shall be able to move the applicants from discard applicants list to Manage Applicants Section.
Source	Raja Shaiyan Azad Khan
Rationale	To acquire the discarded Applicants that have now paid the admission fee.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-74
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Identifier	FR-75
Title	Select Proceed Button
Requirement	The applicant shall be able to create their profile using their details.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the applicant to create the department
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-76
Title	Fill in the Personal Details Form

Requirement	The Applicant shall be able to fill the personal details form and submit the data for creation of their profile.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Applicant to fill in the Personal Detail Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-77
Title	Select Submit Button
Requirement	The Applicant shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-78
Title	Fill in the Income Details Form
Requirement	The Applicant shall be able to fill the income details form and submit the data for creation of their profile.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Applicant to fill in the Income Detail Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-79
Title	Select Submit Button
Requirement	The Applicant shall be able to submit the data filled in the form.

Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-80
Title	Fill in the Education Details Form
Requirement	The Applicant shall be able to fill the education details form and submit the data for creation of their profile.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Applicant to fill in the education Detail Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-81
Title	Select Submit Button
Requirement	The Applicant shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-82
Title	Display Confirmation Message
Requirement	The System shall display a confirmation message to the Applicant for the submission of education details.
Source	Raja Shaiyan Azad Khan

Rationale	To proceed with the submission of education details and complete applicant details and profile section.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-83
Title	Select Okay Button
Requirement	The Applicant shall be able to confirm submission of their details.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the applicant details to be visible to the Admin and Coordinator.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-84
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-85
Title	Select Settings Tab
Requirement	The Users shall be able to update their credentials.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the users to update their credentials
Business Rule (if required)	N/A

Dependencies	N/A
Priority	High

Identifier	FR-86
Title	Fill in the Form
Requirement	The User shall be able to fill the form and submit the data for updation of the credentials.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Users to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-87
Title	Select Submit Button
Requirement	The Users shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-88
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-89
Title	Select Update Admission Session Button
Requirement	The Admin shall be able to update an Admission Session.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the admin to update the Admission Session fields.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-90
Title	Fill in the Form
Requirement	The Admin shall be able to fill the form and submit the data for updation of the Admission Session.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to fill in the update form for Admission Session.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-91
Title	Select Submit Button
Requirement	The Admin shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-92
-------------------	-------

Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-93
Title	Select Enable Admission Session Button
Requirement	The system shall change the status change of the selected Admission Session to enabled.
Source	Raja Shaiyan Azad Khan
Rationale	To allow applicants to apply for the Admission Session, being enabled.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-94
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-95
Title	Select Disable Admission Session Button
Requirement	The system shall change the status change of the selected Admission Session to disabled.
Source	Raja Shaiyan Azad Khan
Rationale	To indicate that the current Admission Session is closed for application.

Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-96
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-97
Title	Select Remove Admission Session Button
Requirement	The system shall remove the Admission Session from the Session list
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to remove the Admission Session if it is not active anymore.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-98
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-99
Title	Select Apply For Program Button
Requirement	The Applicant shall be able to select the program they desire to apply for.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Applicants to proceed on with their registration process.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-100
Title	Display Programs List
Requirement	The System shall display the list of programs being offered.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Applicants to choose their desired programs to apply for.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-101
Title	Select Programs
Requirement	The Applicant shall be able to select the list of programs from the program list.
Source	Raja Shaiyan Azad Khan
Rationale	To provide applicants with the option to select the program they want to apply for.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-102
Title	Select Proceed Button
Requirement	The Applicant shall be able to select the proceed button to lead towards the Personal Details Form

Source	Raja Shaiyan Azad Khan
Rationale	To allow applicants to proceed towards personal details form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-103
Title	Select Upload Button
Requirement	The Applicant shall be able to upload challan images for the submitted challan form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to verify applicants for registration as students based on their challan status.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-104
Title	Select Submit Button
Requirement	The Applicant shall be able to submit their challan form image.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Applicant to submit all their challan form for verification purposes.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-105
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

required)	
Dependencies	N/A
Priority	High

Identifier	FR-106
Title	Select View Challan Button
Requirement	The Admin shall be able to click on the view button next to the applicant in the merit list to view their challan status.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to verify applicant based on challan status.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-107
Title	Display Submitted Challan Form
Requirement	The system shall display the submitted challan form by applicant to Admin for verification purposes.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Admin to verify applicant status.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-108
Title	Select Save option for the challan image
Requirement	The Admin shall be able to save the challan image of the student.
Source	Raja Shaiyan Azad Khan
Rationale	To keep record of the Applicants submitted challan form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

3.3.2 FRS Module-2:

Identifier	FR-109
Title	Select Create Course Button
Requirement	The Coordinator shall be able to create a course.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the coordinator to create the course
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-110
Title	Fill in the Form
Requirement	The Coordinator shall be able to fill the form and submit the data for creation of the course.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to fill in the Course Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-111
Title	Select Submit Button
Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-112
-------------------	--------

Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-113
Title	Select Update Course Button
Requirement	The Coordinator shall be able to update a course.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to modify the course details
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-114
Title	Fill in the Form
Requirement	The Coordinator shall be able to fill the form and submit the data for updation of the course.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-115
Title	Select Submit Button

Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-116
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-117
Title	Select Enable Course Button
Requirement	The system shall change the status change of the selected course to being offered.
Source	Raja Shaiyan Azad Khan
Rationale	To allow coordinators to select specific courses for applicants, being offered.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-118
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

required)	
Dependencies	N/A
Priority	High

Identifier	FR-119
Title	Select Disable Course Button
Requirement	The system shall change the status change of the selected course to not being offered.
Source	Raja Shaiyan Azad Khan
Rationale	To allow coordinators to select only specific courses for applicants, that are being offered.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-120
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-121
Title	Select Remove Course Button
Requirement	The system shall remove the course from the course list
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to remove the course if it is not being offered any more.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-122
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-123
Title	Display Course List
Requirement	The system shall be able to display the course list for the coordinator to select for each student.
Source	Raja Shaiyan Azad Khan
Rationale	To allow Coordinator to enroll students into specific courses.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

3.3.3 FRS MODULE-3:

Identifier	FR-124
Title	Select Courses
Requirement	The Coordinator shall be able to select the check boxes to add the specific courses to the student side and allow them to be enrolled.
Source	Raja Shaiyan Azad Khan
Rationale	To enroll registered students into their courses.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-125
-------------------	--------

Title	Select Save Button
Requirement	The Coordinator shall be able to save the list of courses to the assigned students.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the students to view the courses they are enrolled in.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-126
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-127
Title	Select Create Enrollment Semester Button
Requirement	The Coordinator shall be able to create an Enrollment Semester.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the coordinator to create the Enrollment Semester
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-128
Title	Fill in the Form

Requirement	The Coordinator shall be able to fill the form and submit the data for creation of the Enrollment Semester.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to fill in the Enrollment Semester Form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-129
Title	Select Submit Button
Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-130
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-131
Title	Select Update Enrollment Semester Button
Requirement	The Coordinator shall be able to update a Enrollment Semester.
Source	Raja Shaiyan Azad Khan

Rationale	To allow the Coordinator to update the Enrollment Semester
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-132
Title	Fill in the Form
Requirement	The Coordinator shall be able to fill the form and submit the data for updation of the Enrollment Semester.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-133
Title	Select Submit Button
Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-134
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

Dependencies	N/A
Priority	High

Identifier	FR-135
Title	Select Enable Enrollment Semester Button
Requirement	The system shall change the status change of the selected Enrollment Semester to enabled
Source	Raja Shaiyan Azad Khan
Rationale	To allow coordinators to enroll students into the specific semesters offering courses.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-136
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-137
Title	Select Disable Enrollment Semester Button
Requirement	The system shall change the status change of the selected Enrollment Semester to disabled.
Source	Raja Shaiyan Azad Khan
Rationale	To allow coordinators to enroll students into the specific courses offered in the enabled semester.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-138
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-139
Title	Select Update Section Button
Requirement	The Coordinator shall be able to update a Section.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to update the Section
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-140
Title	Fill in the Form
Requirement	The Coordinator shall be able to fill the form and submit the data for updation of the Section details.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to fill in the update form.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-141
Title	Select Submit Button
Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-142
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-143
Title	Display Updated Section
Requirement	The system shall be able to display the updated section details.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the coordinator to view the updated section details.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-144
Title	Select Register Button
Requirement	The Coordinator shall be able to register an applicant as a student.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to register a student.

Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-145
Title	Fill in the Form
Requirement	The Coordinator shall be able to fill the form and submit the data for registration of students.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to fill in the Registration Form for students.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-146
Title	Select Submit Button
Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-147
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Priority	High
-----------------	------

Identifier	FR-148
Title	Select Create Section Button
Requirement	The Coordinator shall be able to create a section for the enrollment of students.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the courses to be taught in their respective sections.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-149
Title	Fill in the Form
Requirement	The Coordinator shall be able to fill the form and submit the data for creation of the section.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the Coordinator to create the section for courses to be offered in them.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-150
Title	Select Submit Button
Requirement	The Coordinator shall be able to submit the data filled in the form.
Source	Raja Shaiyan Azad Khan
Rationale	To allow the user to submit all the data online.
Business Rule (if required)	N/A

required)	
Dependencies	N/A
Priority	High

Identifier	FR-151
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Raja Shaiyan Azad Khan
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

3.3.4 FRS Module:4

Identifier	FR-152
Title	Generate student schedule
Requirement	The admin shall be able to generate daily schedule and class routine.
Source	Iqra Fareed
Rationale	To allow the user to arrange meeting logs accurately on the availability time of faculty as well as students.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-153
Title	View Student Schedule
Requirement	The faculty shall be able to view students daily schedule and class routine.
Source	Iqra Fareed
Rationale	To view the arranged meeting logs accurately on the availability time of faculty as well as students.
Business Rule	None
Dependencies	None

Priority	High
-----------------	------

Identifier	FR-154
Title	Schedule Meetings
Requirement	The admin shall be able to schedule meetings with their co-workers, admins or other students
Source	Iqra Fareed
Rationale	To allow the user to resolve any queries.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-155
Title	Schedule Tasks
Requirement	The admin shall be able to schedule tasks that are to be completed as a faculty in due time.
Source	Iqra Fareed
Rationale	To allow the user to make students time table
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-156
Title	Change Classroom
Requirement	The Admin will be able to change classroom details in timetable
Source	Iqra Fareed
Rationale	To allow the user to change classroom
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-157
-------------------	--------

Title	Change Timeslot
Requirement	The Admin will be able to change time slot details in timetable
Source	Iqra Fareed
Rationale	To allow the user to change timeslot
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-158
Title	Change Event
Requirement	The Admin will be able to change the event or course name for which they are setting the schedule
Source	Iqra Fareed
Rationale	To allow the user to change event
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-159
Title	Delete Classroom
Requirement	The Admin will be able to delete classroom details in timetable
Source	Iqra Fareed
Rationale	To allow the user to delete classroom
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-160
Title	Delete Timeslot
Requirement	The Admin will be able to delete timeslot details in timetable
Source	Iqra Fareed

Rationale	To allow the user to delete timeslot
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-161
Title	Delete Event
Requirement	The Admin will be able to delete Event details in timetable
Source	Iqra Fareed
Rationale	To allow the user to delete event
Business Rule	None
Dependencies	None
Priority	High

3.3.5 FRS Module 5

Identifier	FR-161
Title	Add Assignment Marks
Requirement	The Faculty will be able to add marks to the students for each subject's assignments
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-162
Title	Edit Assignment Marks
Requirement	The Faculty will be able to edit marks to the students for each subject's assignments

Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-163
Title	Add quiz Marks
Requirement	The Faculty will be able to add marks to the students for each subject's quiz
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-164
Title	Edit quiz Marks
Requirement	The Faculty will be able to edit marks to the students for each subject's quiz
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-165
Title	Add Mid Marks
Requirement	The Faculty will be able to add midterm marks to the students for each subject's midterm
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None

Dependencies	None
Priority	High

Identifier	FR-166
Title	Edit Mid Marks
Requirement	The Faculty will be able to add midterm marks to the students for each subject's midterm
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-167
Title	Add Project Marks
Requirement	The Faculty will be able to add project marks to the students for each subject's project
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-168
Title	Edit Project Marks
Requirement	The Faculty will be able to edit project marks to the students for each subject's project
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-169
Title	Add Final Marks
Requirement	The Faculty will be able to add final marks to the students for each subject's final
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-170
Title	Edit Final Marks
Requirement	The Faculty will be able to edit final marks to the students for each subject's final
Source	Iqra Fareed
Rationale	To make it easy for the user to enter data
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-171
Title	Delete Marks
Requirement	The faculty member shall be able to delete marks sometimes by mistake the enter marks on the wrong site , or any other mistake can occur in this situation, so It is necessary for the user to have the facility to delete data.
Source	Iqra Fareed
Rationale	To give margin a mistake to the user
Business Rule	Marks can only be deleted before deadline of submission of result.
Dependencies	FR-66
Priority	High

Identifier	FR-172
Title	Import CSV File
Requirement	The faculty shall be able to import CSV files containing students record for each term after each term has finished or during that term.
Source	Iqra Fareed
Rationale	To allow the user to maintain student's information in the CMS for evaluation purposes.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-173
Title	Upload Assignment
Requirement	The user shall be able to upload Assignment based on which user shall be able to mark assessments.
Source	Iqra Fareed
Rationale	To allow the faculty to make marks available for evaluation.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-174
Title	Upload Lectures
Requirement	The user shall be able to upload Lectures on which their assessments are based.
Source	Iqra Fareed
Rationale	To allow the faculty to make study material for the students.
Business Rule	None
Dependencies	None
Priority	High

3.3.6 FRS Module 6

Identifier	FR-175
Title	Add student's Information
Requirement	The admin shall be able to add students to the selected. The system shall only display the students who are eligible to be allocated the course i.e., they have passed the pre-req etc
Source	Iqra Fareed
Rationale	To let the admin, select and add students who are eligible.
Business Rule	Only students who fulfill eligibility criteria are being added to the list.
Dependencies	None
Priority	High

Identifier	FR-176
Title	Update Profile
Requirement	The admin shall be able to update student's profile
Source	Iqra Fareed
Rationale	To allow the user to update their personal information
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-177
Title	Insert Details
Requirement	The user shall be able to insert the student's information
Source	Iqra Fareed
Rationale	To allow preparation of the course details for teaching timeline.
Business Rule	The teacher shall not be able to view and insert details of courses not allocated to him/her.
Dependencies	None
Priority	High

Identifier	FR-178
-------------------	--------

Title	Maintain data
Requirement	The admin shall be able to keep backup/ maintain data by uploading data to the CMS.
Source	Iqra Fareed
Rationale	To allow the user to keep a record of a student's academic career for each semester.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-179
Title	Add to database
Requirement	The system shall enter all the information gather from the person into the database to store that information permanently
Source	Iqra Fareed
Rationale	To make sure that the information submitted by the user is authentic system shall apply a verification test
Business Rule	None.
Dependencies	None
Priority	High

Identifier	FR-180
Title	Delete Record from Database
Requirement	The system shall delete the record from the database.
Source	Iqra Fareed
Rationale	To update the deleted record in the database as well.
Business Rule	Only the main admin can delete the record
Dependencies	FR-16
Priority	High

Identifier	FR-181
-------------------	--------

Title	Display Students
Requirement	The admin shall be able to view the list of students.
Source	Iqra Fareed
Rationale	To let the admin have a list of students to easily perform actions of registering a course or to print out a list of the students of the semester.
Business Rule	Only registered students in the merit list will be displayed onto the portal.
Dependencies	FR-16
Priority	High

Identifier	FR-182
Title	View Student Profile
Requirement	The faculty shall be able to view students information like previous academic record and other extra curricular activities performed by them.
Source	Iqra Fareed
Rationale	To allow the user to evaluate students performance throughout his degree.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-183
Title	View All Student Profile
Requirement	The admin shall be able to view students information like previous academic record and other extra curricular activities performed by them.
Source	Iqra Fareed
Rationale	To allow the user to evaluate students performance throughout his degree.
Business Rule	None
Dependencies	None
Priority	High

3.3.7 FRS Module 7:

Identifier	FR-184
-------------------	--------

Title	Register Faculty
Requirement	The admin shall be able to select on the register faculty button.
Source	Noor us Sahar
Rationale	To allow users to register faculty.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-185
Title	Form with Fields
Requirement	The admin shall be able to view a form with fields required to register the faculty.
Source	Noor us Sahar
Rationale	To allow users to register faculty.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-186
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-187
Title	Create Profile
Requirement	The faculty will be able to create his/her profile.
Source	Noor us Sahar
Rationale	To allow users to create his/her profile.
Business Rule (if required)	N/A
Dependencies	N/A

Priority	High
-----------------	------

Identifier	FR-188
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-189
Title	Edit Profile
Requirement	The faculty will be able to edit his/her profile.
Source	Noor us Sahar
Rationale	To allow users to edit his/her profile.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-190
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-191
Title	View Profile
Requirement	The faculty will be able to view his/her profile.
Source	Noor us Sahar

Rationale	To allow users to view his/her profile.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-192
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-193
Title	Faculty Member Board
Requirement	The user will be able to select “Faculty Member” button on dashboard
Source	Noor us Sahar
Rationale	To allow users to view his/her profile.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-194
Title	Profile of Faculty Member
Requirement	The faculty will be able to view profiles of all faculty members.
Source	Noor us Sahar
Rationale	To allow users to view his/her profile.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-195
-------------------	--------

Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-196
Title	Delete Profile
Requirement	The user will be able to delete a profile
Source	Noor us Sahar
Rationale	To allow users to view his/her profile.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-197
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-198
Title	Add Research Paper
Requirement	The faculty will be able to add research papers
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-199
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-200
Title	Add Education
Requirement	The faculty will be able to add education
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-201
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-202
Title	Add Experience
Requirement	The faculty will be able to add experience
Source	Noor us Sahar

Rationale

To allow users to add research papers

Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-203
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-204
Title	View Calendar
Requirement	The faculty will be able to view his/her calendar
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-205
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-206
Title	View Evaluation Result of Self

Requirement	The faculty will be able to view his/her evaluation result
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-207
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-208
Title	View Attendance
Requirement	The faculty will be able to view his/her attendance
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-209
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A

Priority	High
-----------------	------

Identifier	FR-210
Title	View Teaching Schedule
Requirement	The faculty will be able to view his/her teaching schedule as assigned by the admin.
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-211
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-212
Title	Add event
Requirement	The faculty will be able to add an event to the calendar
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-213
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.

Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-214
Title	Delete event
Requirement	The faculty will be able to delete an event to the calendar
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-215
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-216
Title	View Evaluation Report of All Faculty
Requirement	The user will be able to evaluate reports of all the faculty members.
Source	Noor us Sahar
Rationale	To allow users to add research papers
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-217
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

3.3.8 FRS Module 8:

Identifier	FR-218
Title	Suggestion Button
Requirement	The user selects “Suggestion” button.
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-219
Title	Entering Information
Requirement	The system prompts the user to enter his/her name, email, suggestion.
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-220
-------------------	--------

Title	Entering the Fields
Requirement	The user enters the fields required.
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-221
Title	Update Information
Requirement	System asks the user if he/she wants to update
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-222
Title	Selecting Option
Requirement	The user selects NO
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-223
Title	Confirmation for Mail
Requirement	The system asks the user if he/she wants to send mail.
Source	Noor us Sahar

Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-224
Title	Selecting Option
Requirement	The user select YES
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-225
Title	Sending Mail
Requirement	The system send the mail
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-226
Title	System Help
Requirement	The system asks the user if he/she needs help with anything else.
Source	Noor us Sahar
Rationale	To allow the user to give a suggestion
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-227
Title	Complain Button
Requirement	The user selects “Complain” button.
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-228
Title	Entering Information
Requirement	The system prompts the user to enter his/her name, email, suggestion.
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-229
Title	Entering the Fields
Requirement	The user enters the fields required.
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-230
Title	Update Information
Requirement	System asks the user if he/she wants to update
Source	Noor us Sahar

Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-231
Title	Selecting Option
Requirement	The user selects NO
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-232
Title	Confirmation for Mail
Requirement	The system asks the user if he/she wants to send mail.
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-233
Title	Selecting Option
Requirement	The user select YES
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-234
Title	Sending Mail
Requirement	The system send the mail
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-235
Title	System Help
Requirement	The system asks the user if he/she needs help with anything else.
Source	Noor us Sahar
Rationale	To allow the user to file a complaint.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-236
Title	Find a Page Button
Requirement	User selects 'Find a Page' button
Source	Noor us Sahar
Rationale	To allow the user to find a page.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-237
Title	Set of Pages
Requirement	System shows a set of pages.
Source	Noor us Sahar

Rationale	To allow the user to find a page.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-238
Title	Selecting Option
Requirement	User selects the page he/she wants to find
Source	Noor us Sahar
Rationale	To allow the user to find a page.
Business Rule	None
Dependencies	None
Priority	High

Identifier	FR-239
Title	Redirection
Requirement	System redirects the user to the page.
Source	Noor us Sahar
Rationale	To allow the user to find a page.
Business Rule	None
Dependencies	None
Priority	High

3.3.9 FRS Module 9:

Identifier	FR-240
Title	Fill a Performa
Requirement	The faculty will be able to fill a Performa in order to generate results.
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A

Dependencies	N/A
Priority	High

Identifier	FR-241
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-242
Title	Report of Students button
Requirement	The faculty will be able to select on generate report of Students button
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-243
Title	Type of Report
Requirement	The faculty will be able to select the type of report to generate
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-244
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A

required)	
Dependencies	N/A
Priority	High

Identifier	FR-245
Title	Faculty Button
Requirement	The faculty will be able to select on generate report of Faculty button
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-246
Title	Report to Generate
Requirement	The faculty will be able to select selects the type of report to generate
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-247
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-248
Title	Upload Results
Requirement	The admin will be able to upload the results.
Source	Noor us Sahar

Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-249
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-250
Title	Change Graph Type
Requirement	The admin will be able to change the graph types of the reports generated.v
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-251
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-252
Title	View Reports

Requirement	The user will be able to view the reports
Source	Noor us Sahar
Rationale	To allow users to create groups
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-253
Title	Generate Success Message
Requirement	The system shall be able to indicate the success of an action performed.
Source	Noor us Sahar
Rationale	To allow users to get feedback on completion of an action.
Business Rule (if required)	N/A
Dependencies	N/A
Priority	High

Identifier	FR-254
Title	Logout
Requirement	The system shall logout the user from the application.
Source	Client
Rationale	To allow the user to logout from the application.
Business Rule	None
Dependencies	None
Priority	High

3.4 Non-Functional Requirements

3.4.1 Usability

USE-1	A new user shall be easily able to learn how to use the system after 3-5 times of using.
--------------	--

USE-2	A trained user shall be able to perform tasks in less than 3 minutes at minimum and 5 minutes maximum depending on the task.
USE-3	90 percent of the users shall be able to perform major tasks without requiring any help.

3.4.2 Responsiveness

RESP-1	The searches performed within the system in any module will take no more than 2 seconds to process.
RESP-2	Any request in the system will be completed within 4 seconds.
RESP-3	The verification code will be sent within 3 seconds after the user has submitted the form in the account creation page.
RESP-4	The update of records in the database will not take more than 2 seconds.
RESP-5	Retrieval of data from the database will not take more than 2 seconds.

3.4.3 Performance

PER-1	The searches performed within the system in any module will take no more than 4 seconds to process.
PER-2	Any request in the system will be completed within 4 seconds.
PER-3	The verification code will be sent within 3 seconds after the user has submitted the form in the account creation page.

3.4.4 Scalability

SCB-1	The initial system shall be able to support up to a maximum of 20 users by hosting this application on a cloud hosting platform.
SCB-2	The initial system shall be able to handle up to 10 concurrent users at a time by hosting this application on a cloud hosting platform.

3.4.5 Security

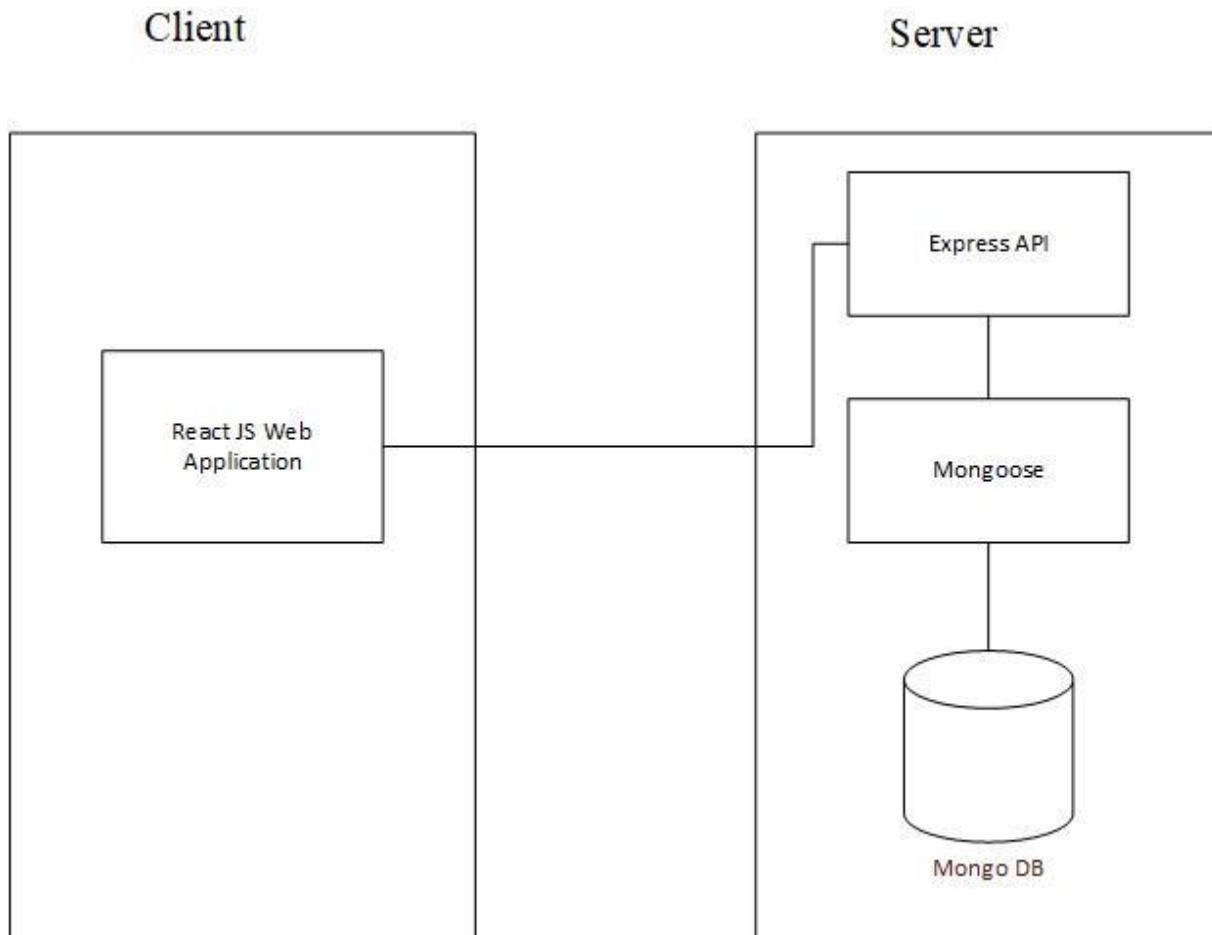
SEC-1	The system shall perform hashing of password to protect against unauthorized access.
--------------	--

3.4.6 Reliability

REL-1	The system shall be available at least 90 percent of the time by hosting this application on cloud hosting platform.
--------------	--

4 Design and Architecture

4.1 System Architecture



4.2 Data Representation

4.2.1 Admission

```
{  
  sessions: [  
    {  
      name: {
```

```
        type: String
    },
    startDate: {
        type: Date
    },
    endDate: {
        type: Date
    },
    timeStamp: {
        type: Date,
        default: Date.now()
    },
    status: {
        type: Boolean,
        default: true
    },
    meritList: [
        {
            applicantId: {
                type: mongoose.Schema.Types.ObjectId,
                ref: 'applicant'
            }
        }
    ],
    notification: {
        type: String
    }
},
fee: {
    amount: {
        type: Number
    },
    feePaid: {
        type: Boolean,
        default: false
    }
}
}
```

4.2.2 Announcement

```
{
```

```

feeds: [
  {
    name: {
      type: String
    },
    message: {
      type: String
    },
    startDate: {
      type: Date
    },
    endDate: {
      type: Date
    },
    status: {
      type: Boolean,
      default: true
    }
  }
]
}

```

4.2.3 Applicant

```

{
  // To keep track who is the actual user account associated with application object
  user: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'user'
  },
  // To keep track of what type of applicant the user is i.e. Bachelors or Masters
  type: {
    type: Number
  },
  // To keep track of status of applicant form
  status: {
    type: Number,
    default: 0
  },
  personalDetails: {
    name: {
      type: String
    }
  }
}

```

```
  },
  fatherName: {
    type: String
  },
  email: {
    type: String
  },
  // CNIC should remain like previous
  cnic: {
    number: {
      type: String
    },
    frontPicture: {
      type: String
    },
    backPicture: {
      type: String
    }
  },
  avatar: {
    type: String
  },
  address: {
    type: String
  },
  placeOfBirth: {
    type: String
  },
  dateOfBirth: {
    type: Date
  },
  phoneNumber: {
    type: String
  },
  domicile: {
    type: String
  }
},
incomeDetails: {
  monthlyIncome: {
    type: Number
  },
  minimumYearlyIncome: {
    type: Number
  }
},
```

```
educationDetails: {
    // Here is the details of the secondary education
    secondaryEducationDetails: {
        type: {
            type: Number
        },
        institute: {
            type: String
        },
        fieldOfStudy: {
            type: String
        },
        from: {
            type: Number
        },
        to: {
            type: Number
        },
        obtainedMarks: {
            type: Number
        },
        totalMarks: {
            type: Number
        },
        picture: {
            type: String
        }
    },
    // Here is the details of the intermediate education
    intermediateEducationDetails: {
        type: {
            type: Number
        },
        institute: {
            type: String
        },
        fieldOfStudy: {
            type: String
        },
        from: {
            type: Number
        },
        to: {
            type: Number
        },
        obtainedMarks: {
```

```

        type: Number
    },
    totalMarks: {
        type: Number
    },
    picture: {
        type: String
    }
},
// Here is the details of the bachelor education
bachelorEducationDetails: {
    institute: {
        type: String
    },
    fieldOfStudy: {
        type: String
    },
    from: {
        type: Number
    },
    to: {
        type: Number
    },
    cgpa: {
        type: Number
    },
    picture: {
        type: String
    }
},
ntsMarks: {
    type: Number
},
universityTestScore: {
    type: Number
},
totalAggregate: {
    type: Number
}
},
appliedPrograms: [
{
    programme: {
        type: mongoose.Schema.Types.ObjectId,
        ref: 'programme'
    }
}
]

```

```
        }
    ],
registeredCourses: [
    {
        course: {
            type: mongoose.Schema.Types.ObjectId,
            ref: 'course'
        }
    }
],
timeStamp: {
    type: Date,
    default: Date.now()
},
applicantVerified: {
    type: Boolean,
    default: false
},
applicantDiscarded: {
    type: Boolean,
    default: false
},
applicantForwarded: {
    type: Boolean,
    default: false
},
applicationForwarded: {
    type: Boolean,
    default: false
},
admission: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'admission'
},
feeDetails: {
    challanPicture: {
        type: String,
        default: null
    },
    amountPaid: {
        type: Number,
        default: 0
    }
},
isStudent: {
    type: Boolean,
```

```
    default: false
  }
}
```

4.2.4 Course

```
{
  name: {
    type: String,
    required: true
  },
  description: {
    type: String,
    required: true
  },
  creditHours: {
    type: Number,
    default: 0
  },
  fee: {
    type: Number,
    default: 0
  },
  category: {
    type: Number
  },
  type: {
    type: Number
  },
  compulsory: {
    type: Boolean,
    default: true
  },
  preRequisite: {
    type: String,
    default: null
  },
  minimumAttendanceRequiredForTerminals: {
    type: Number,
    default: 0
  },
  courseLearningOutcomes: {
    type: String,
  }
}
```

```
    required: true
  },
  programLearningOutcomes: {
    type: String,
    required: true
  },
  isOffered: {
    type: Boolean,
    default: true
  },
  isCompleted: {
    type: Boolean,
    default: false
  },
  lectures: {
    totalLectures: {
      type: Number,
      default: 0
    },
    lecturesTakenFaculty: {
      type: Number,
      default: 0
    },
    lecturesTakenStudent: {
      type: Number,
      default: 0
    }
  },
  assignments: {
    totalAssignments: {
      type: Number,
      default: 0
    },
    assignmentsTaken: {
      type: Number,
      default: 0
    }
  },
  quizzes: {
    totalQuizzes: {
      type: Number,
      default: 0
    },
    quizzesTaken: {
      type: Number,
      default: 0
    }
  }
}
```

```
        },
      },
      midTerms: {
        totalMidTerms: {
          type: Number,
          default: 0
        },
        midTermsTaken: {
          type: Number,
          default: 0
        }
      },
      labs: {
        totalLabs: {
          type: Number,
          default: 0
        },
        labsTakenFaculty: {
          type: Number,
          default: 0
        },
        labsTakenStudent: {
          type: Number,
          default: 0
        },
        labManual: {
          type: String,
          default: null
        },
        labProjectTotalMarks: {
          type: Number,
          default: 0
        },
        labTerminalTotalMarks: {
          type: Number,
          default: 0
        }
      },
      program: {
        type: mongoose.Schema.Types.ObjectId,
        ref: 'programme'
      },
      department: {
        type: mongoose.Schema.Types.ObjectId,
        ref: 'department'
      },
    },
```

```
    section: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'section'
    }
}
```

4.2.5 Department

```
{
  name: {
    type: String,
    required: true,
  },
  description: {
    type: String,
    required: true,
  },
}
```

4.2.6 Enrollment

```
{
  semesters: [
    {
      name: {
        type: String,
        required: true
      },
      description: {
        type: String,
        required: true
      },
      maximumBatchStrength: {
        type: Number,
        default: 0
      },
      status: {
        type: Boolean,
        default: true
      },
    }
},
```

```

startDate: {
  type: Date
},
endDate: {
  type: Date
},
program: {
  type: mongoose.Schema.Types.ObjectId,
  ref: 'programme'
},
courseList: [
  {
    courseId: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'course'
    }
  }
],
sectionList: [
  {
    sectionId: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'section'
    }
  }
]
}

```

4.2.7 Programme

```
{
  name: {
    type: String,
    required: true
  },
  description: {
    type: String,
    required: true
  },
  duration: {
    yearly: {
      type: Number,

```

```

        default: 0
    },
    semester: {
        type: Number,
        default: 0
    }
},
feePerSemester: {
    type: Number,
    required: true
},
criteria: {
    minPercentageOfEquivalence: {
        type: Number
    },
    minCGPA: {
        type: Number
    },
    categoryOfDegree: {
        type: Number
    }
},
department: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'department'
},
status: {
    type: Boolean,
    default: true
},
type: {
    type: Number
}
}

```

4.2.8 Section

```
{
  name: {
    type: String,
    required: true
  },
  maximumStrength: {

```

```

        type: Number,
        default: 0
    },
    currentNumberOfStudents: {
        type: Number,
        default: 0
    },
    semester: {
        type: mongoose.Schema.Types.ObjectId,
        ref: 'enrollment'
    }
}

```

4.2.9 Student profile

```

{
    user:{
        type:mongoose.Schema.Types.ObjectId,
        ref:'user'
    },
    Name:{
        type: String,
        required: true
    },
    Program:{
        type: String,
        required: true
    },
    fatherName:{
        type: String,
        required: true
    },
    registrationNo:{
        type: String,
        required: true,
        unique:true
    },
    Nationality:{
        type: String,
        required: true,
        unique:true
    },
    cnic:{
        type:String,

```

```
        required:true,
        unique:true
    },
    address: {
        District: {
            type: String,
            required:true
        },
        area: {
            type: String,
            required:true
        },
        city: {
            type: String,
            required:true
        }
    },

semester:{  
    type:Number,  
    required:true  
},  
section:{  
    type:String,  
    required:true  
},  
dateOfBirth:{  
    type:Date,  
    required:true  
},  
phone_number: {  
    type: Number,  
    required: true,  
},  
domicile:{  
    type:String,  
    required:true  
},  
school: {  
    type:String,  
    required:true  
},  
college:{
```

```
    type:String,
    required:true
},
Olevels_maticMarks:{
    type: Number,
    required: true
},
alevels_fscMarks:{
    type: Number,
    required: true
},
feildOfStudy:{
    //pre-eng // pre-med
    type: String,
    required: true
},
yearOfMariculatioin:{
    type: Date,
    required: true
},
yearOfIntermediate:{
    type: Date,
    required: true
},
overseasEducation:{
    type: String,
},
gpa:{
    type:mongoose.SchemaTypes.Decimal128,
    required:true
},
cgpa:{
    type: mongoose.SchemaTypes.Decimal128,
    required:true
},
department:{
    type: String,
    required:true
}
```

```
 },
courses: {
  type: [String],
  required: true
},
grades:{
  type:"String"
}
}
```

4.2.10 Survey Forms

```
{
surveys:
{
  type: Object,
  required: true
},
publish_date : {
  type : Date
}
,
end_date :{
  type: Date
}
}
```

4.2.11 Survey Responses

```
{  
  user_email: {  
    type: String,  
    required: true,  
  },  
  survey_id: {  
    type: String,  
    required: true,  
  },  
  title: {  
    type: String,  
    required: true,  
  },  
  satisfied: {  
    type: Number,  
    required: true,  
  },  
  not_satisfied: {  
    type: Number,  
    required: true,  
  },  
  neutral: {  
    type: Number,  
    required: true,  
  },  
  department: {  
    type: String,  
    // required: true  
  },  
  course: {  
    type: String,  
    // required: true  
  },  
  response: {  
    type: Object,  
  },  
}
```

4.2.12 User

```
{
```

```

name: {
  type: String,
  required: true
},
email: {
  type: String,
  required: true
},
type: {
  type: Number,
  required: true
},
password: {
  type: String,
  required: true
},
avatar: {
  type: String
},
date: {
  type: Date,
  default: Date.now()
},
description: {
  type: String
},
department: {
  type: mongoose.Schema.Types.ObjectId,
  ref: 'department'
}
}

```

4.2.13 Calendar Event

```
{
faculty: {
  type: mongoose.Schema.Types.ObjectId,
  ref: "faculty",
},
event: [
  {
    title: { type: String, required: true },
    start: { type: Date, default: Date.now },
  }
]
}
```

```
        end: { type: Date, default: Date.now },
    },
],
}
```

4.2.14 Faculty

```
{
  name : {
    type : String,
    require : true,
  },
  email : {
    type : String,
    required : true,
    unique : true
  },
  password : {
    type : String,
    required : true
  }
}
```

4.2.15 Faculty Profile

```
{
  faculty: {
    type: mongoose.Schema.Types.ObjectId,
    ref: "faculty",
  },
  department :{
    type : String
  },
  phone_number: {
```

```
    type: Number,
    required: true,
  },
  designation: {
    type: String,
    required: true,
  },
  website :{
    type: String
  },
  address: {
    type : String,
    // required: true
  },
  courses_teaching: {
    type: [String],
    required: true,
  },
  bio: {
    type: String,
  },
  date: {
    type: Date,
    default: Date.now,
  },
  research_papers: [
    {
      title: {
        type: String,
        required: true,
      },
      date: {
        type: Date,
        required: true,
      },
      author: {
        type: [String],
        required: true,
      },
      description: {
        type: String,
      },
    },
  ],
  education :[
```

```
{  
    institute : {  
        type : String,  
        required : true  
    },  
    degree : {  
        type : String,  
        required : true  
    },  
  
    from : {  
        type :Date,  
        required : true  
    },  
    to : {  
        type :Date,  
    },  
    grade :{  
        type : String,  
        required : true  
    }  
}  
},  
experience : [  
{  
    company : {  
        type : String,  
        required : true  
    },  
    designation : {  
        type : String,  
        required : true  
    },  
  
    from : {  
        type :Date,  
        required : true  
    },  
    to : {  
        type :Date,  
    },  
}
```

]
}

4.3 Process Flow/Representation

4.3.1 Admission

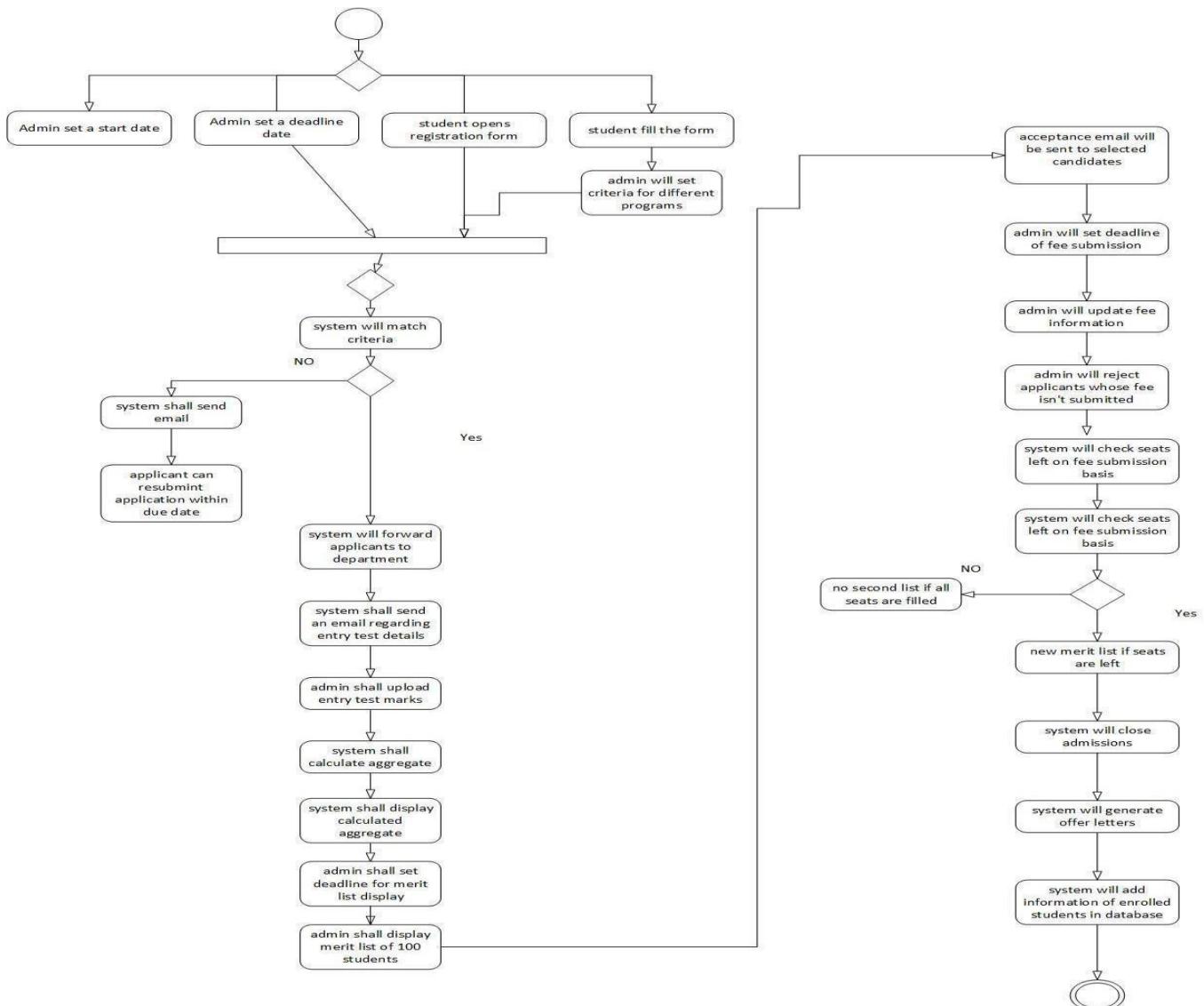


Figure 4-1: Process flow admission

4.3.2 Maintain Course Detail

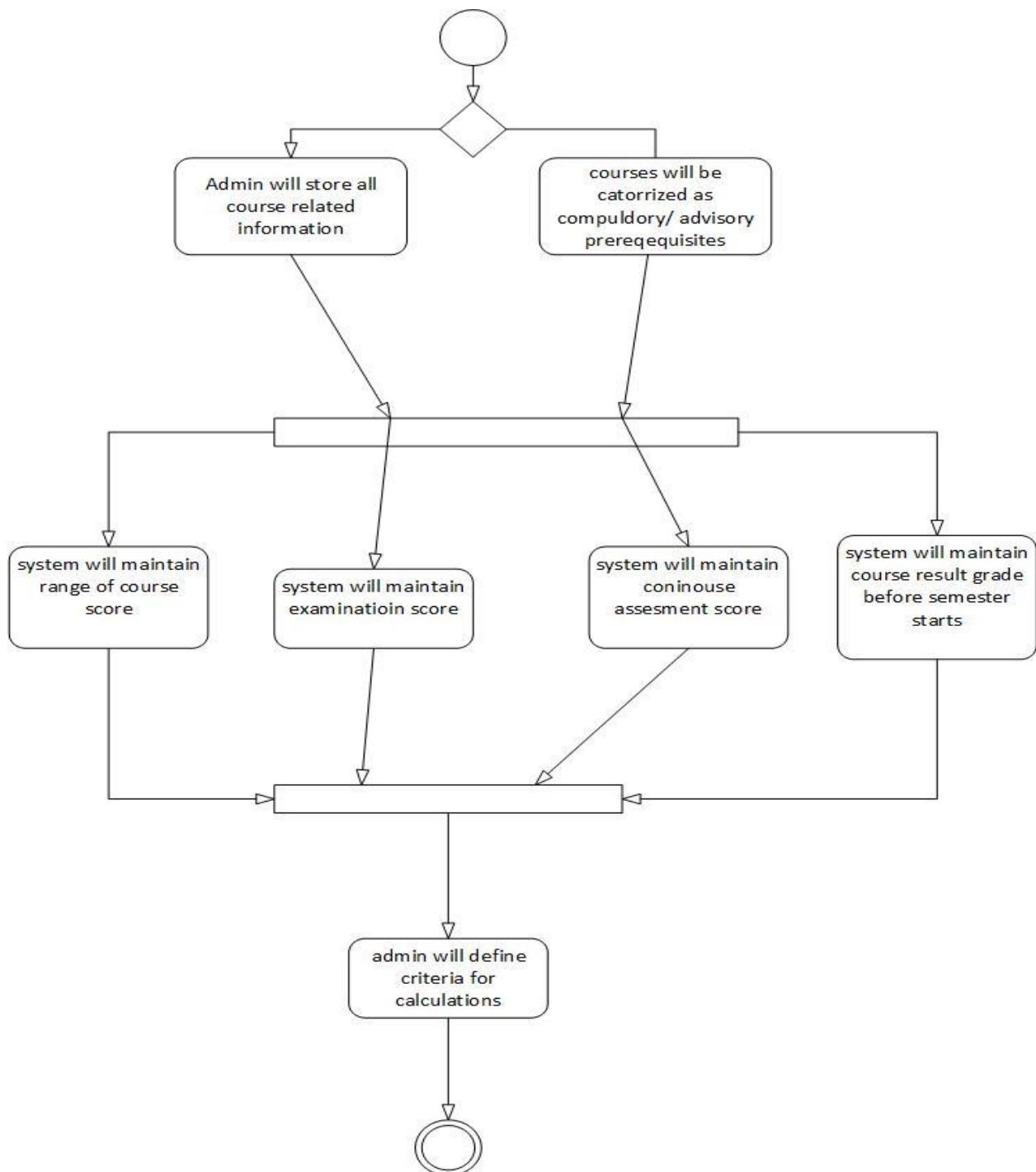


Figure 4-2 Process flow Maintain Course Detail

4.3.3 Enrollment

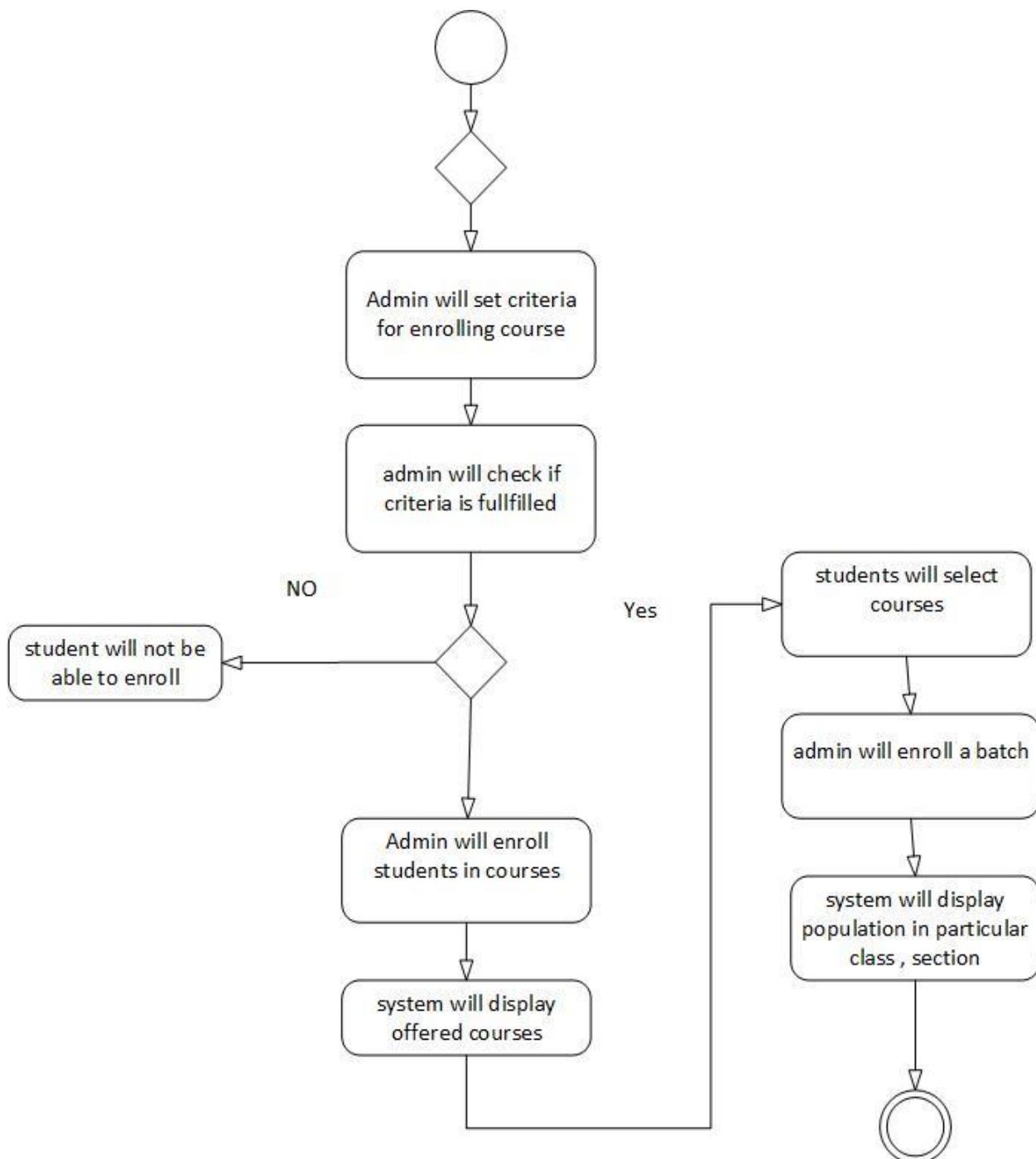


Figure 4-3 Process Flow Enrollment

4.3.4 Class Scheduling

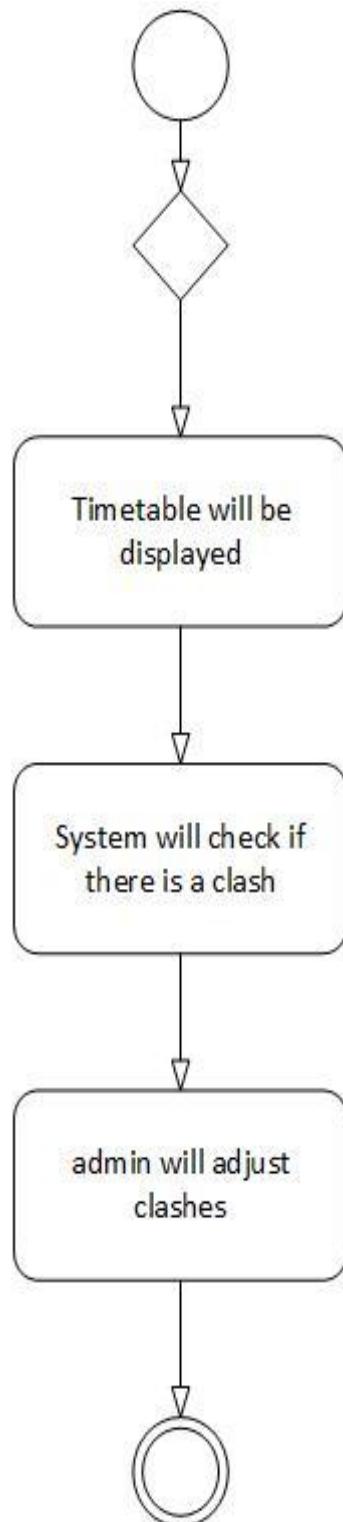


Figure 4-4 Process flow : Class Scheduling

4.3.5 Grading

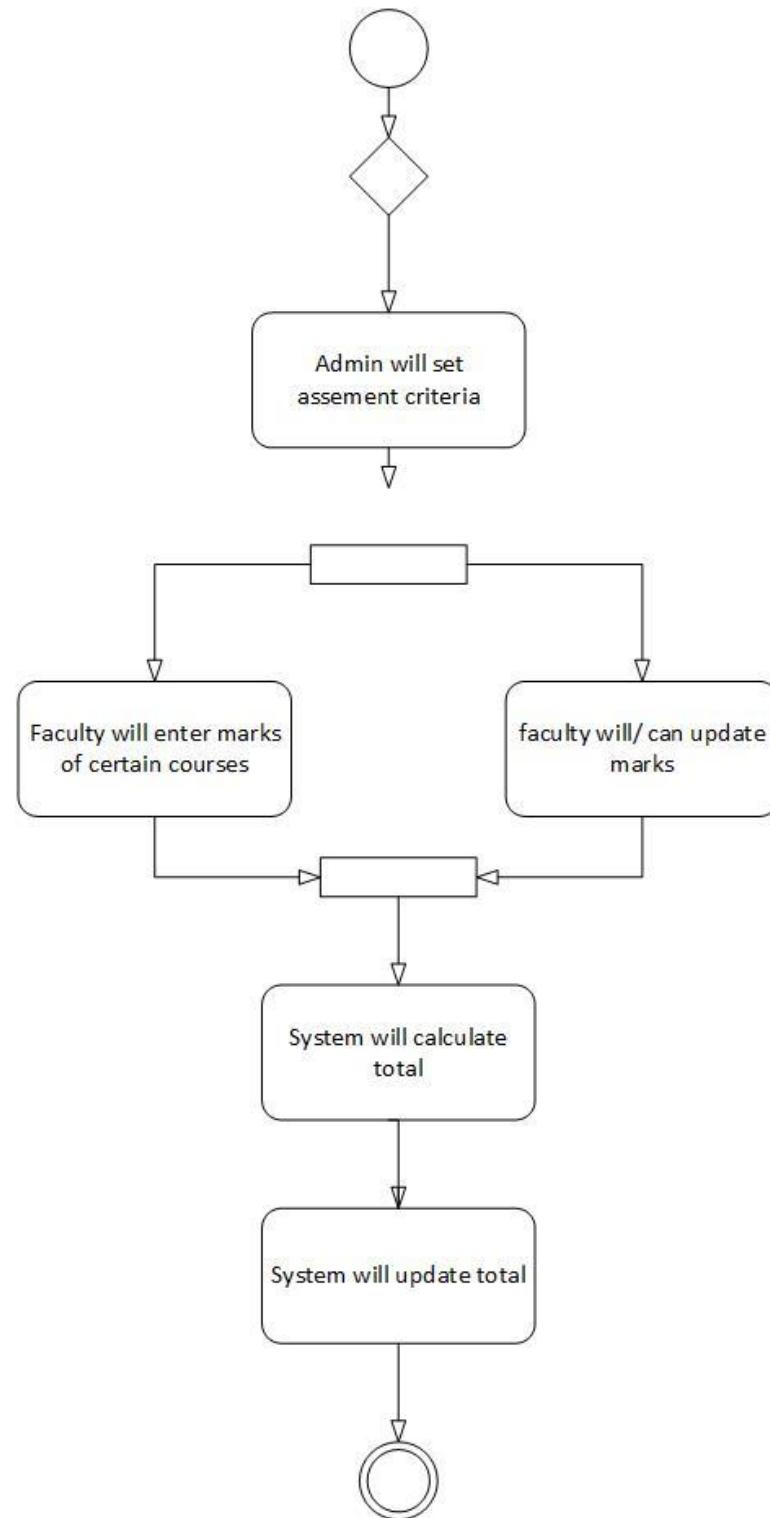


Figure 4-5 Process Flow Grading

4.3.6 Student Academic History

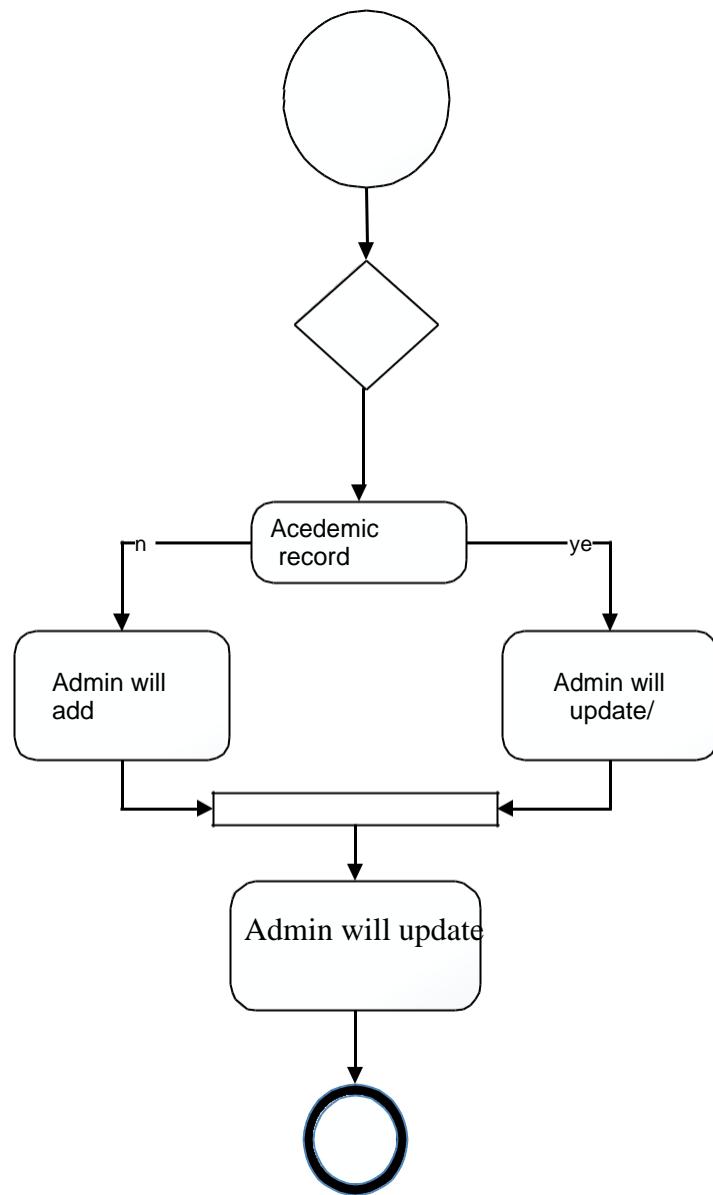


Figure 4-6 Process Flow Student Academic History

4.3.7 Faculty Record

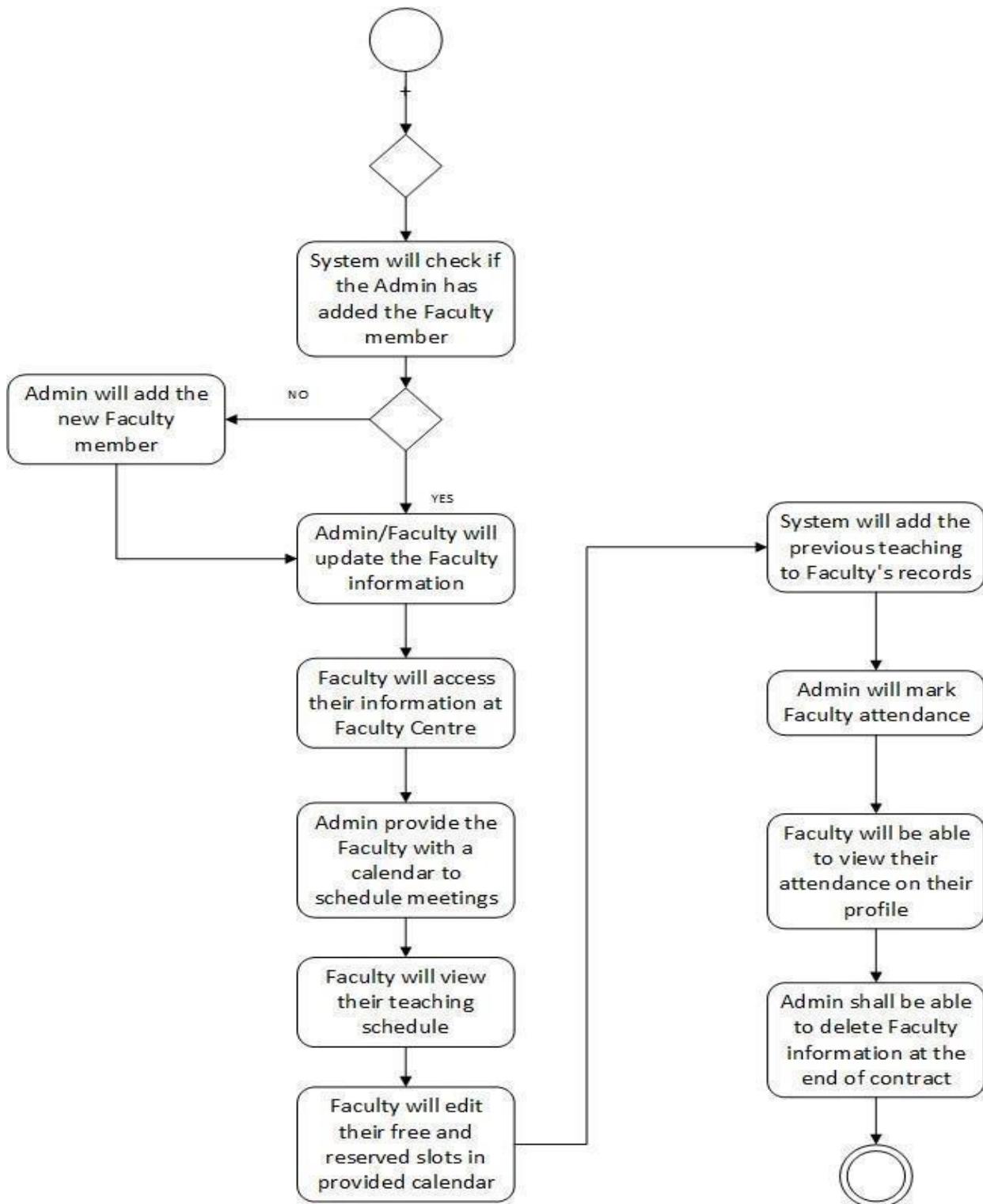


Figure 4-7 Process Flow Faculty Record

4.3.8 Communication

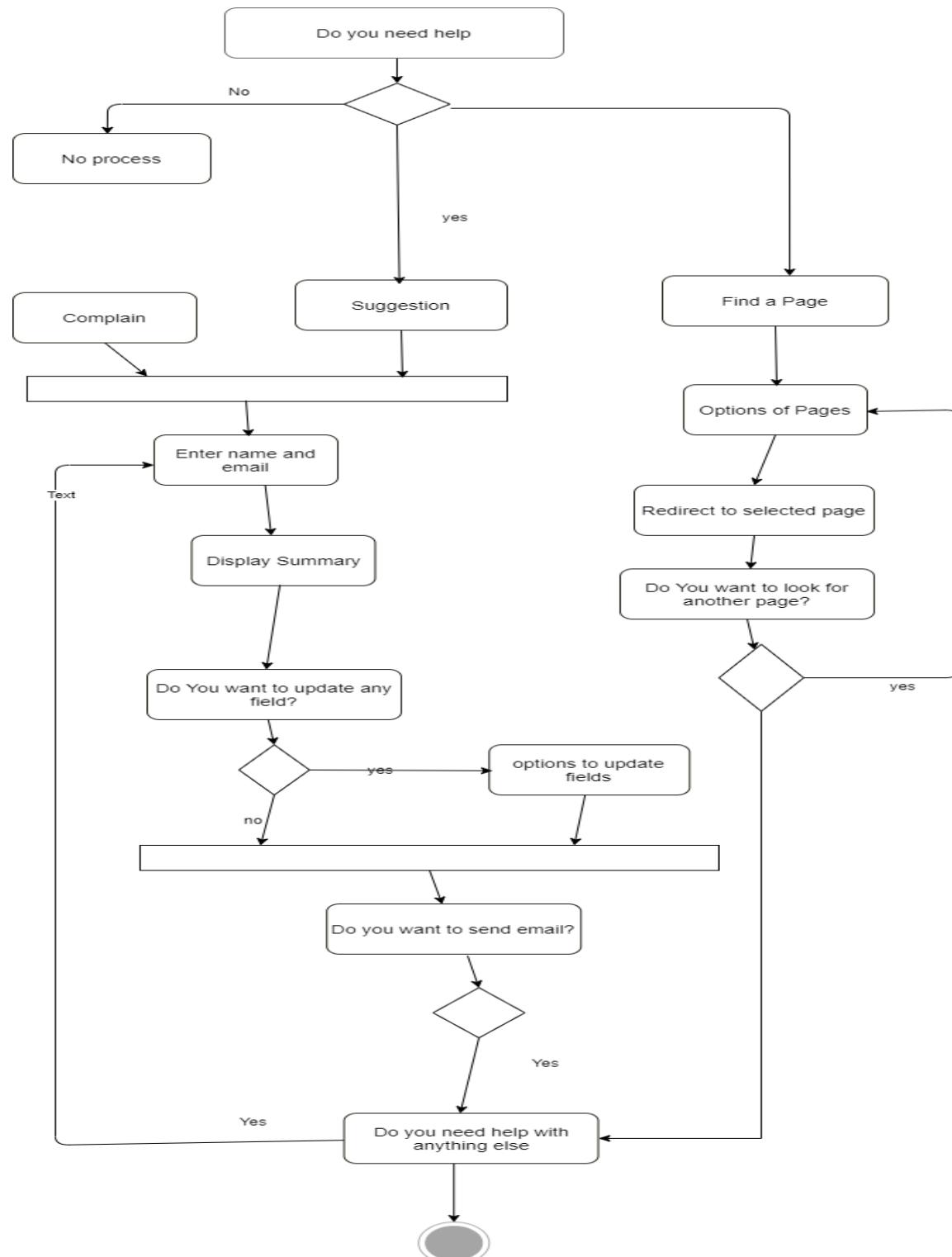


Figure 4-8 Process Flow Communication Chatbot

4.3.9 Report Analysis

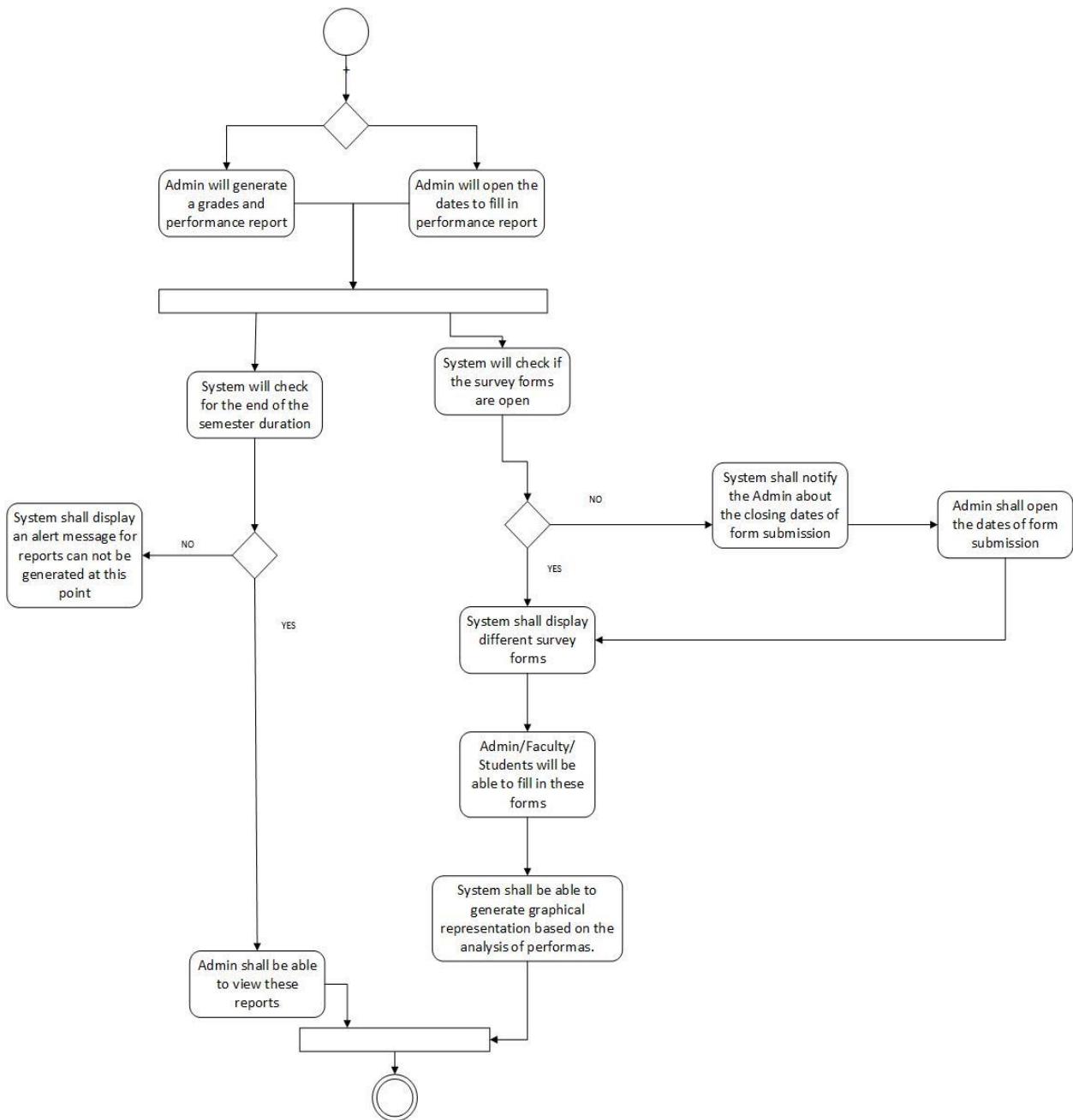


Figure 4-9 Process Flow Report Analysis

4.4 Design Models

4.4.1 Structural Diagrams

4.4.1.1 Level 0 DFD

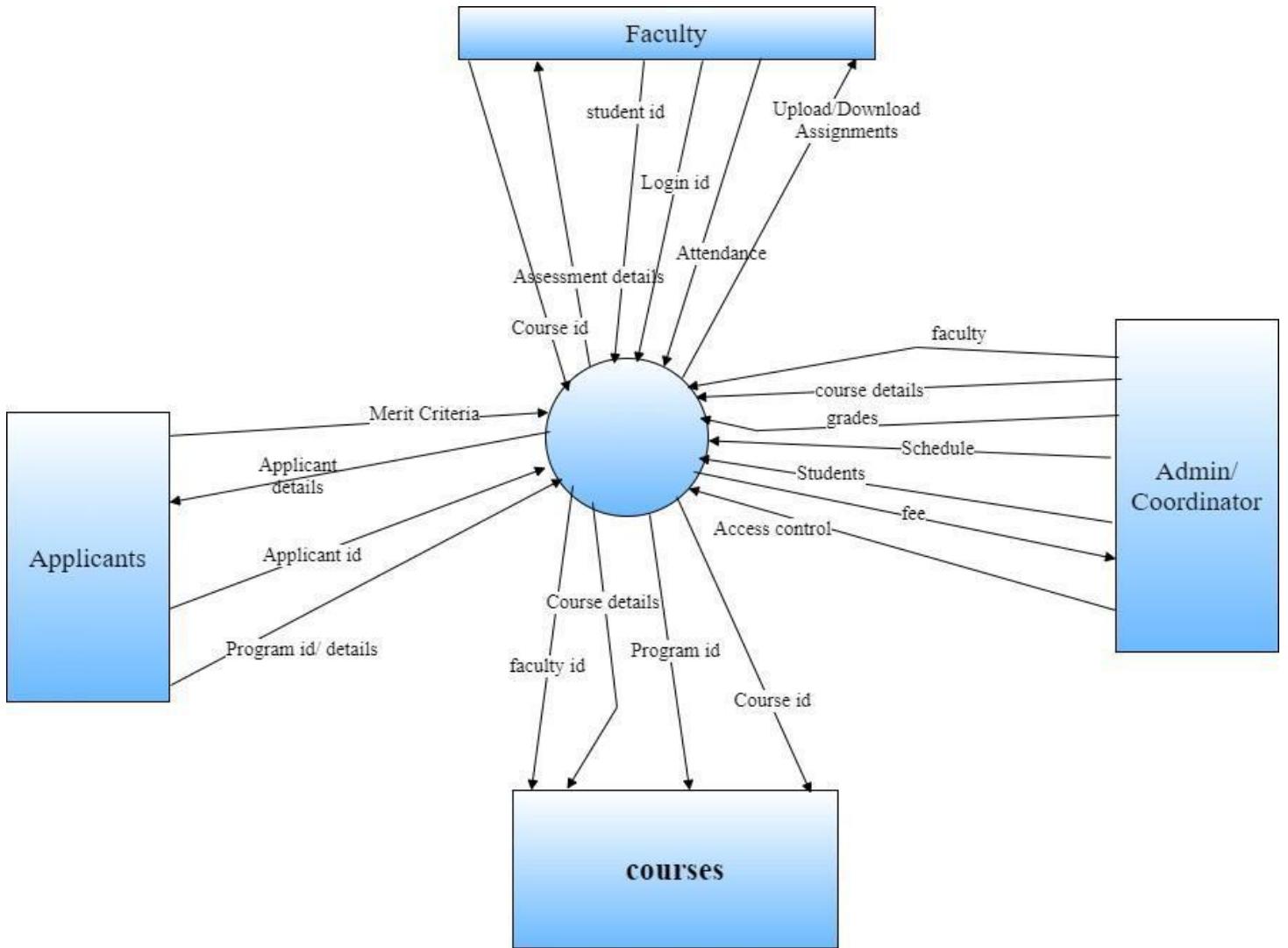


Figure 4-10 Level 0 DFD

4.4.1.2 Level 1 DFD

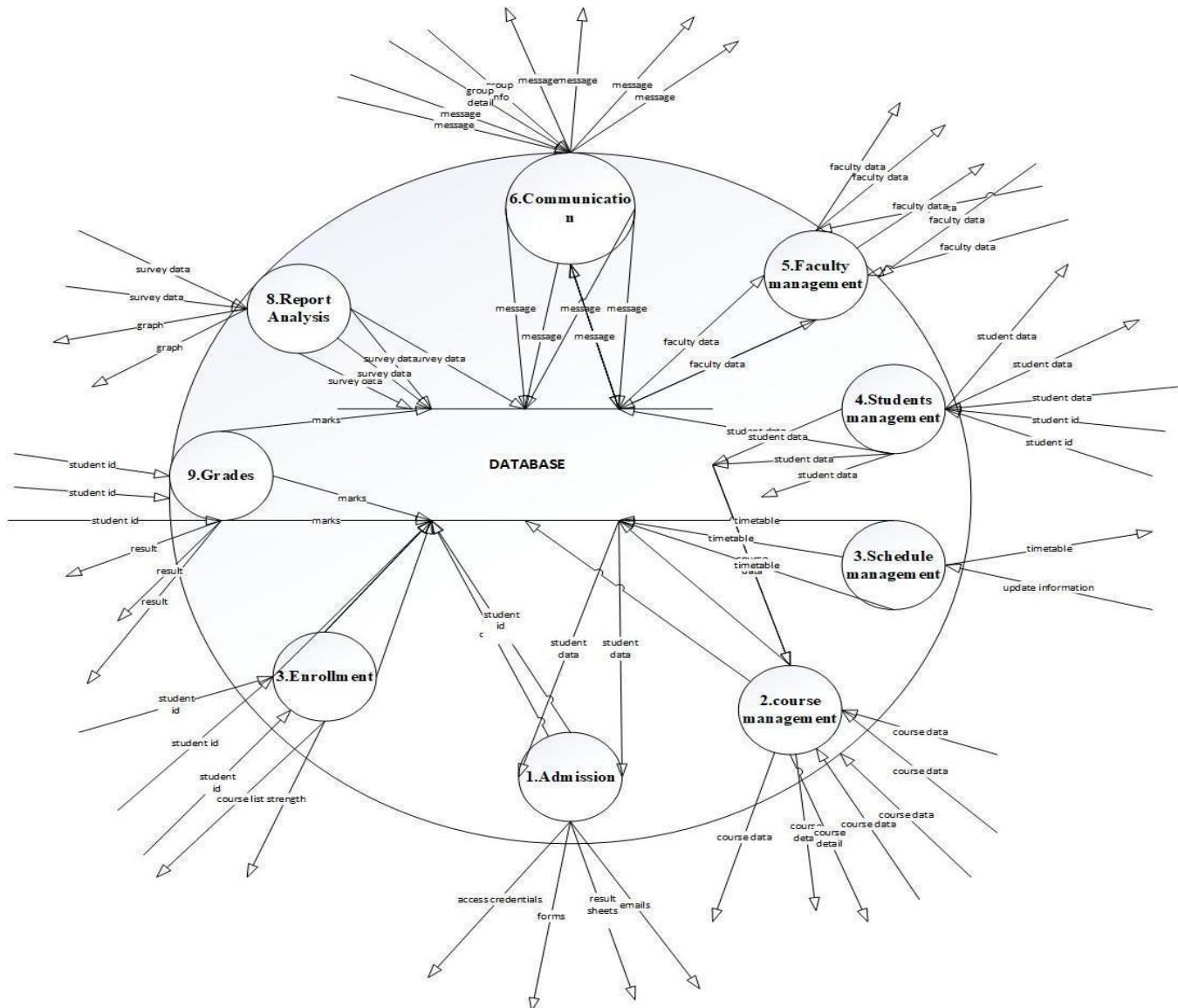


Figure 4-11 Level 1 DFD

4.4.1.3 Level 2 DFD Admission Management

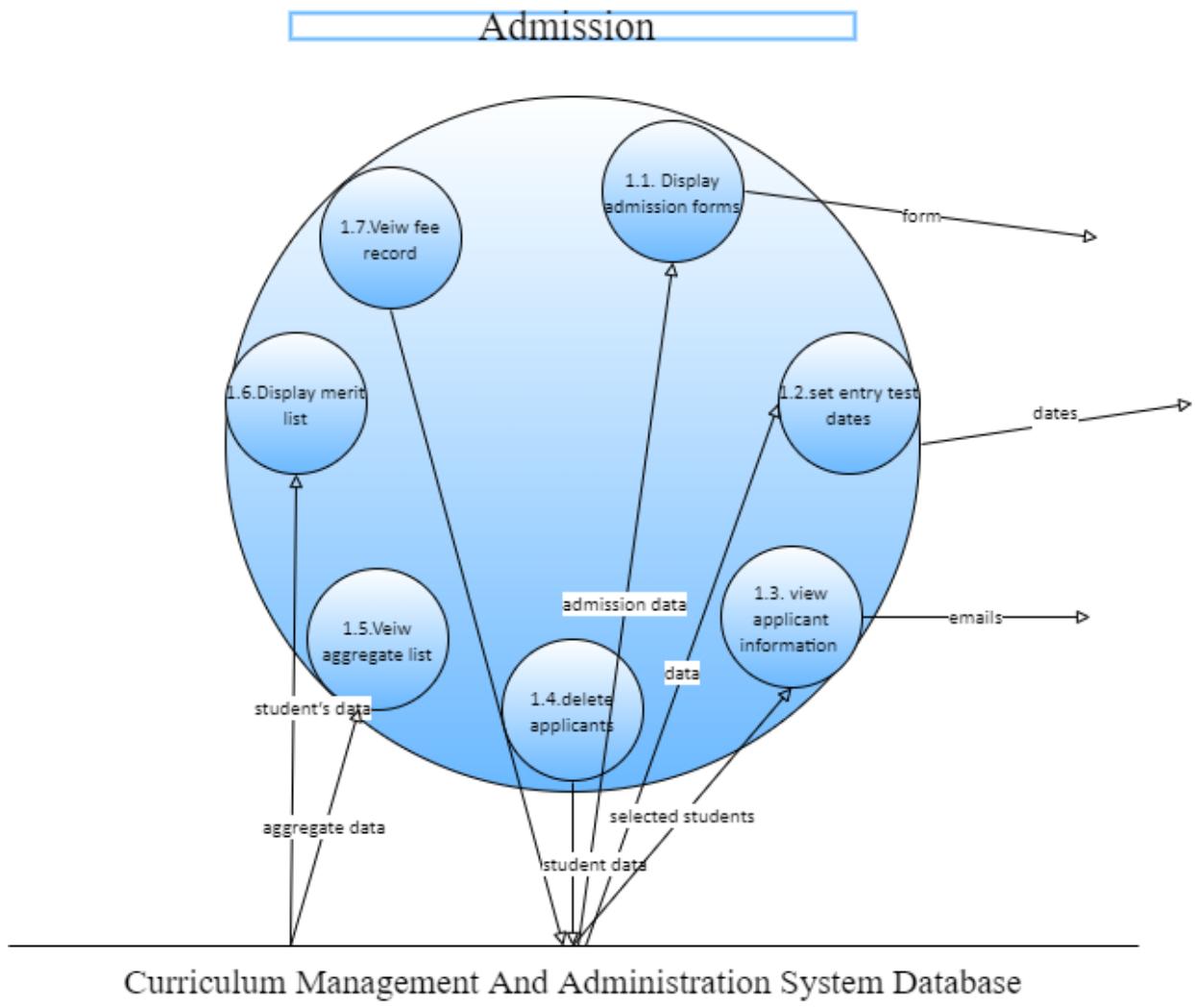


Figure 4-12 Level 2 DFD Admission Management

4.4.1.4 Level 2 DFD Course Management

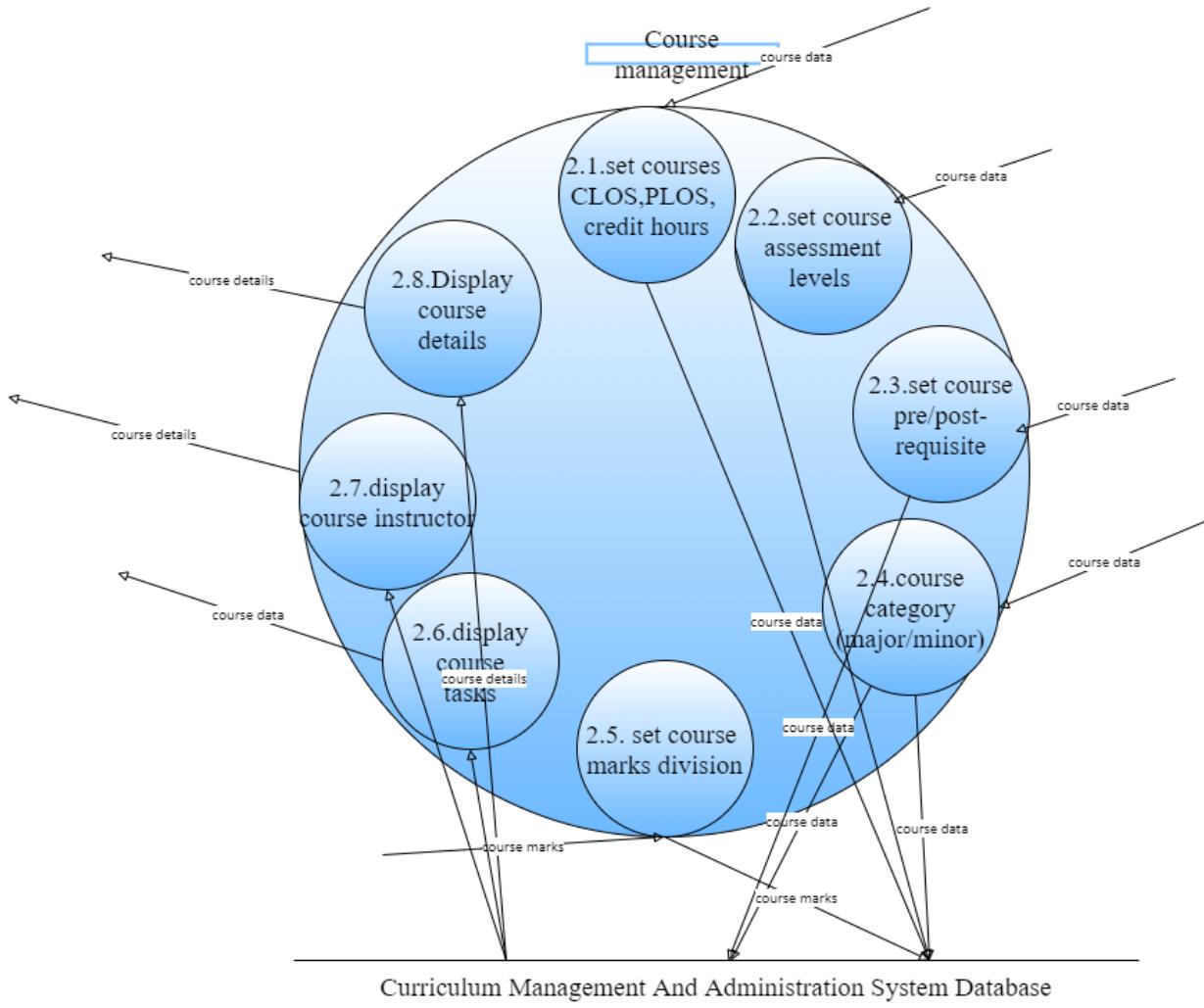


Figure 4-13 Level 2 DFD Course Management

4.4.1.5 Level 2 DFD Enrollment Management

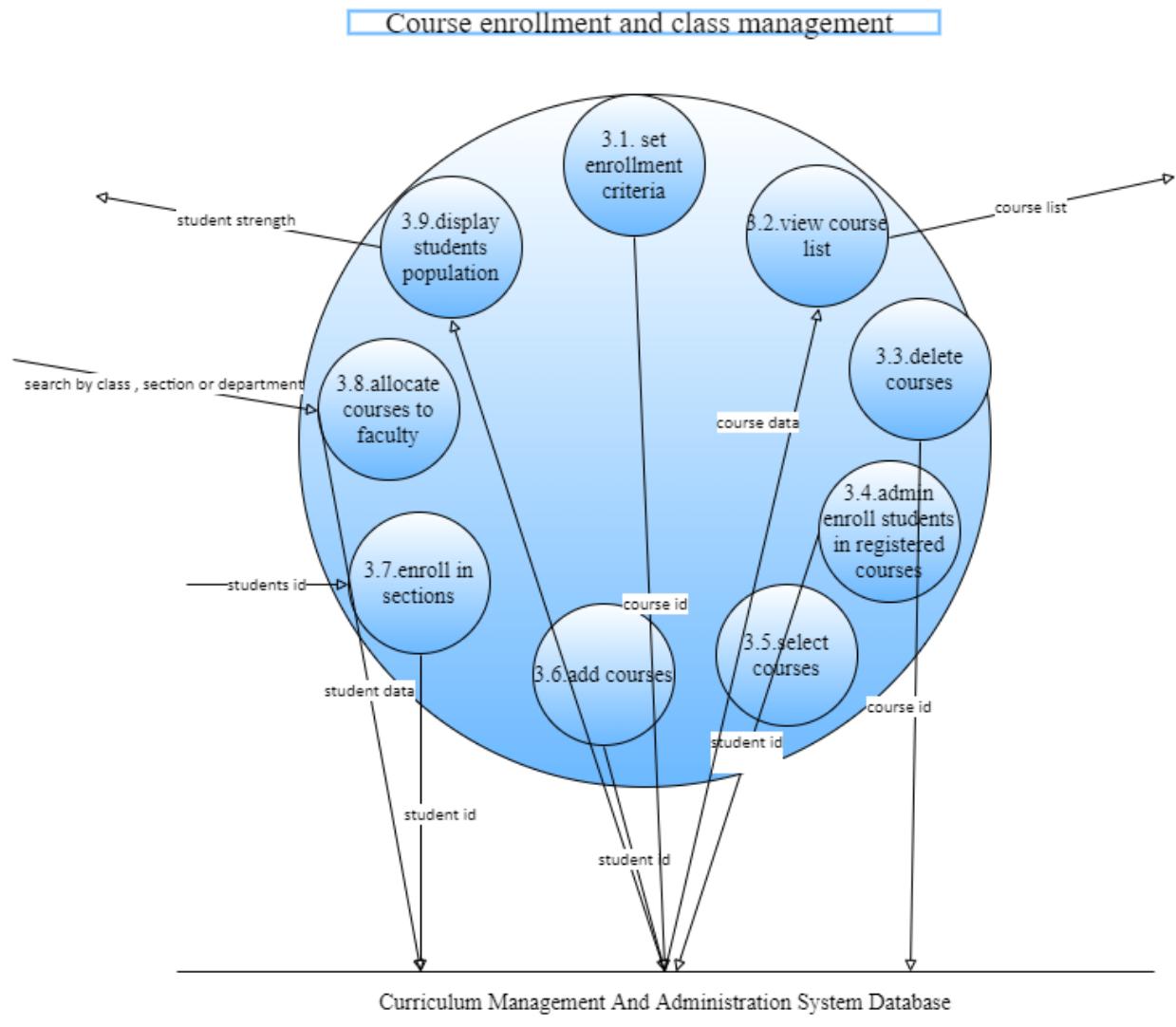


Figure 4-14 Level 2 DFD Enrollment Management

4.4.1.6 Level 2 DFD Timetable Management

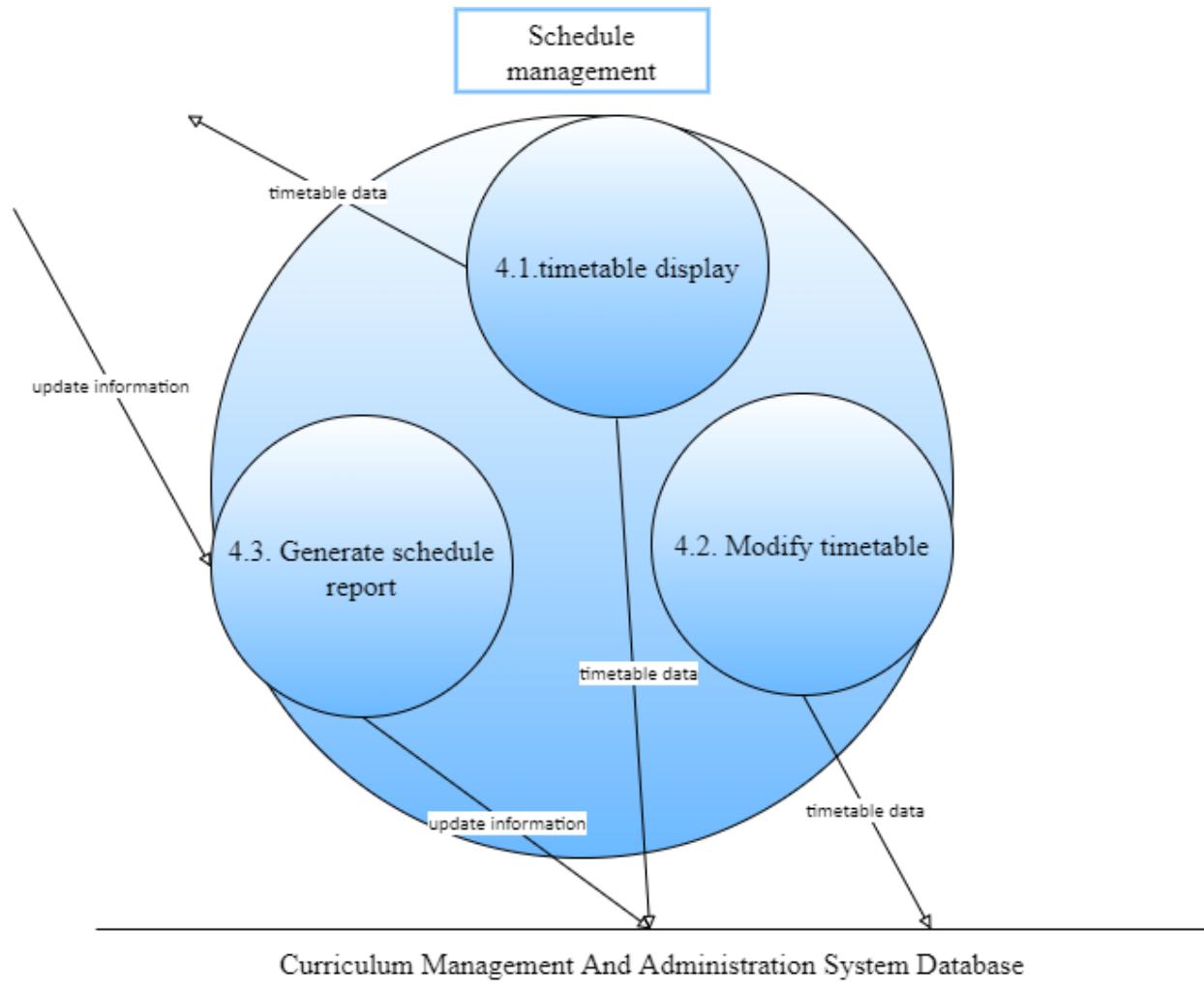


Figure 4-15 Level 2 DFD Timetable Management

4.4.1.7 Level 2 DFD Grade Management

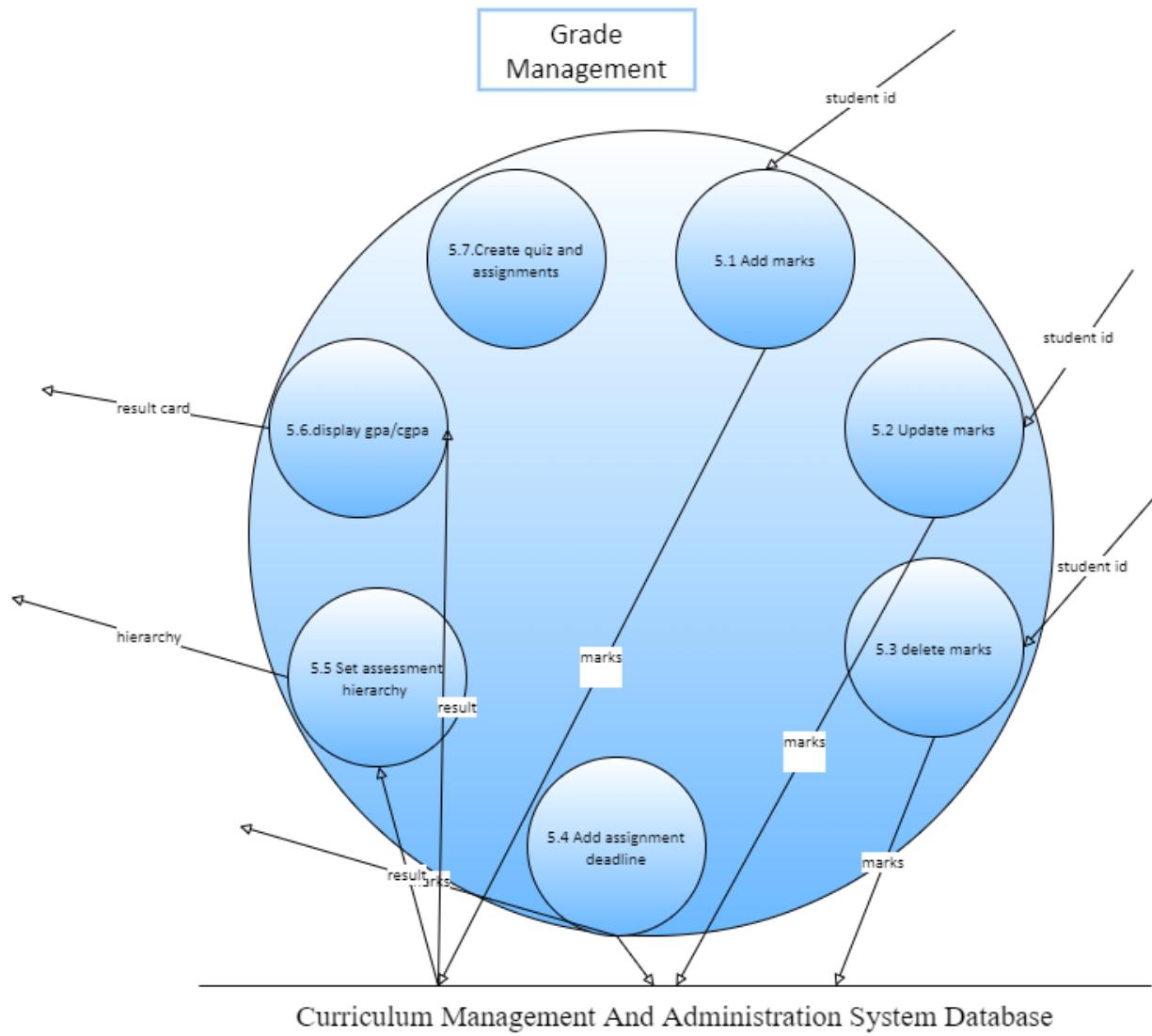


Figure 4-16 Level 2 DFD Grade Management

4.4.1.8 Level 2 DFD Student academic record

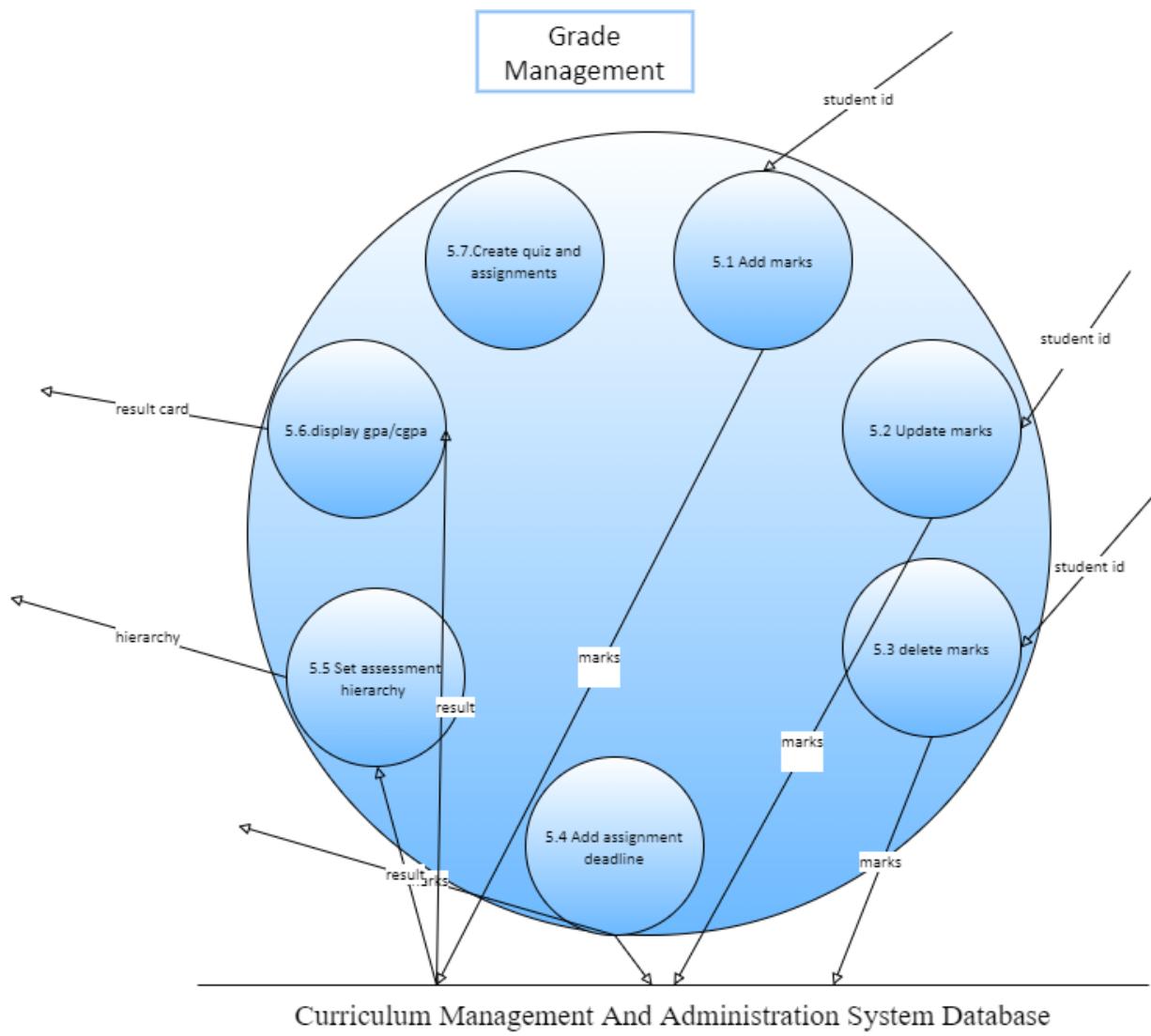


Figure 4-17 Level 2 DFD Student academic record

4.4.1.9 Level 2 DFD Faculty Management

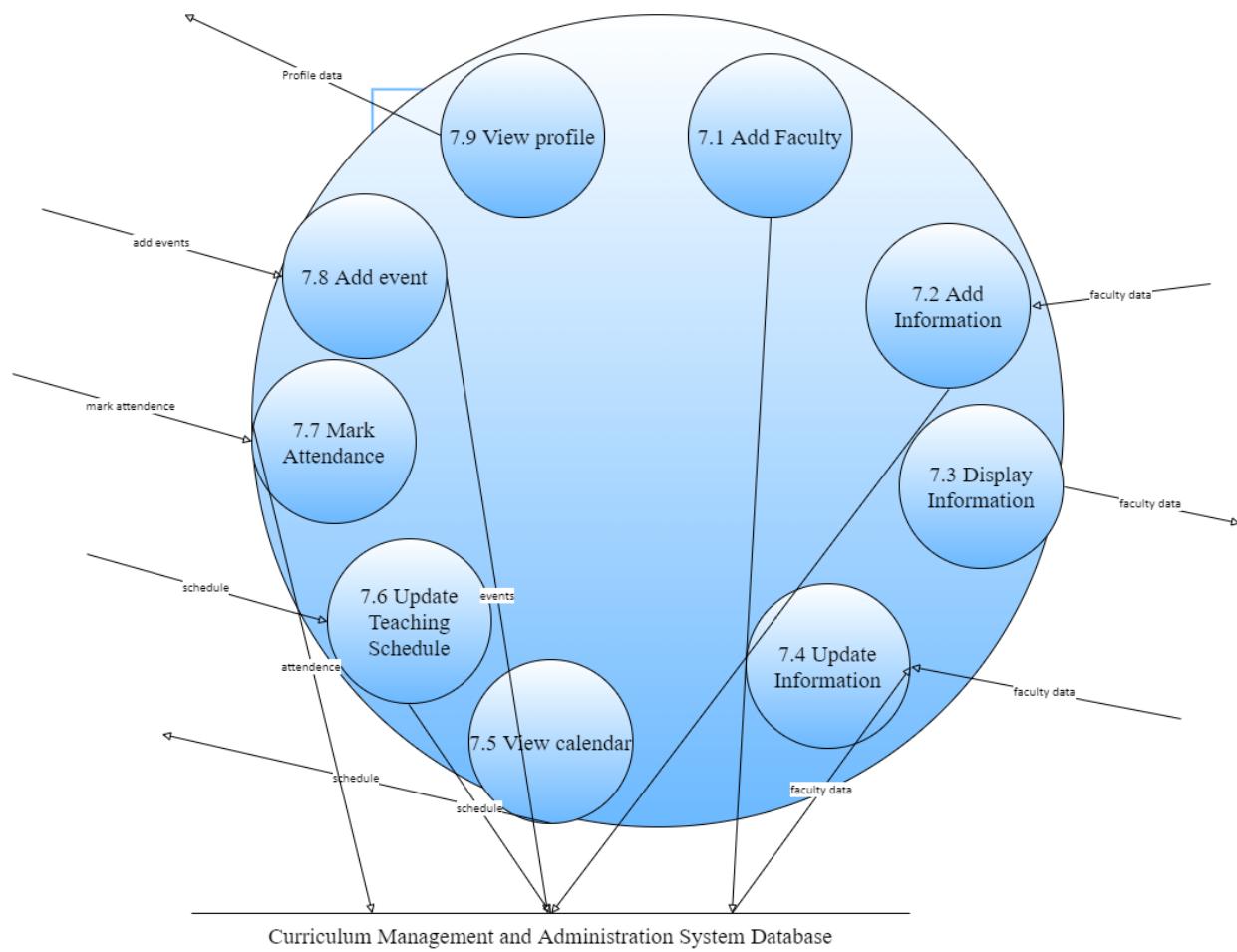


Figure 4-18 Level 2 DFD Faculty Management

4.4.1.10 Level 2 DFD Communication Management

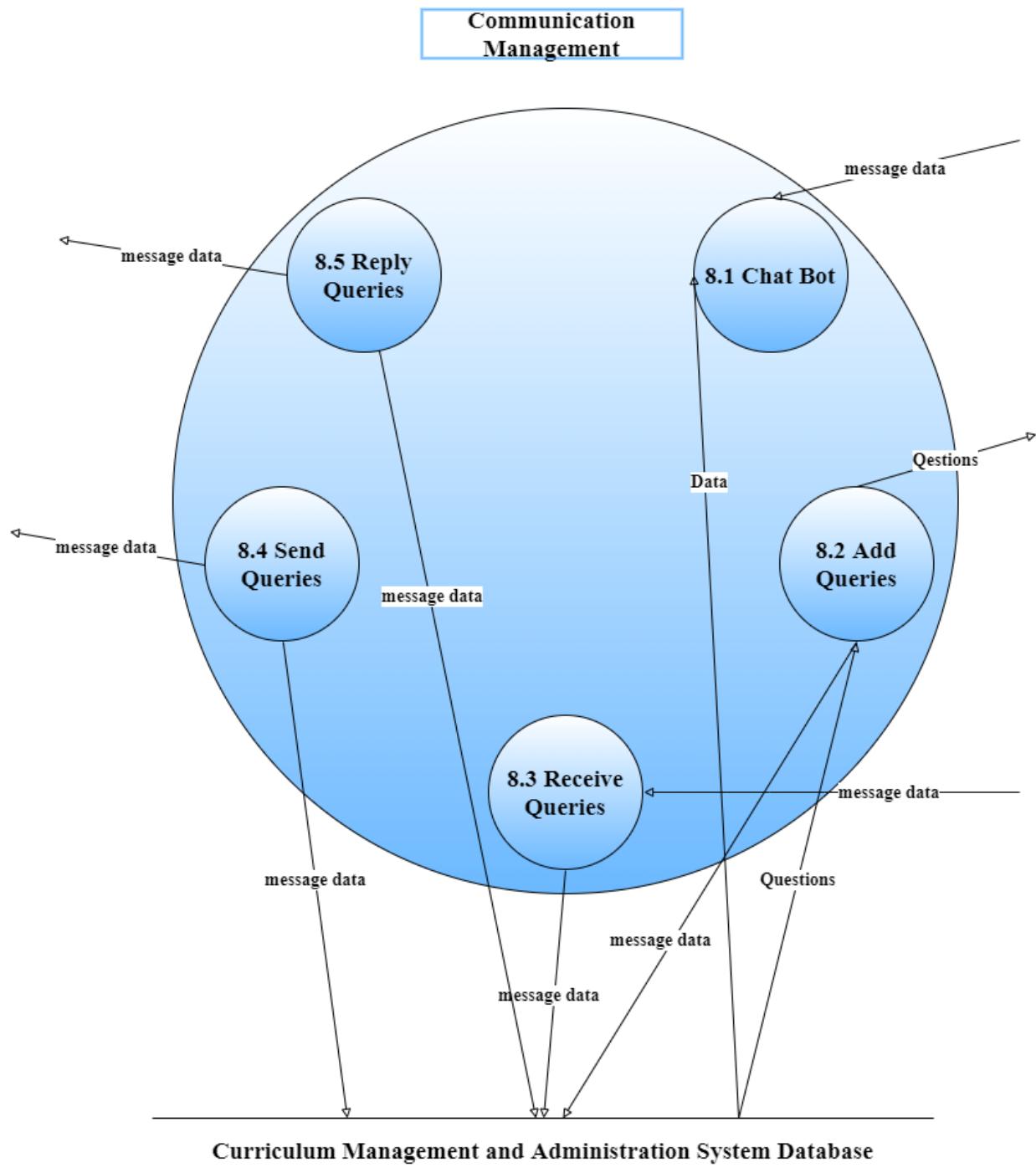


Figure 4-19 Level 2 DFD Communication Management

4.4.1.11 Class diagram

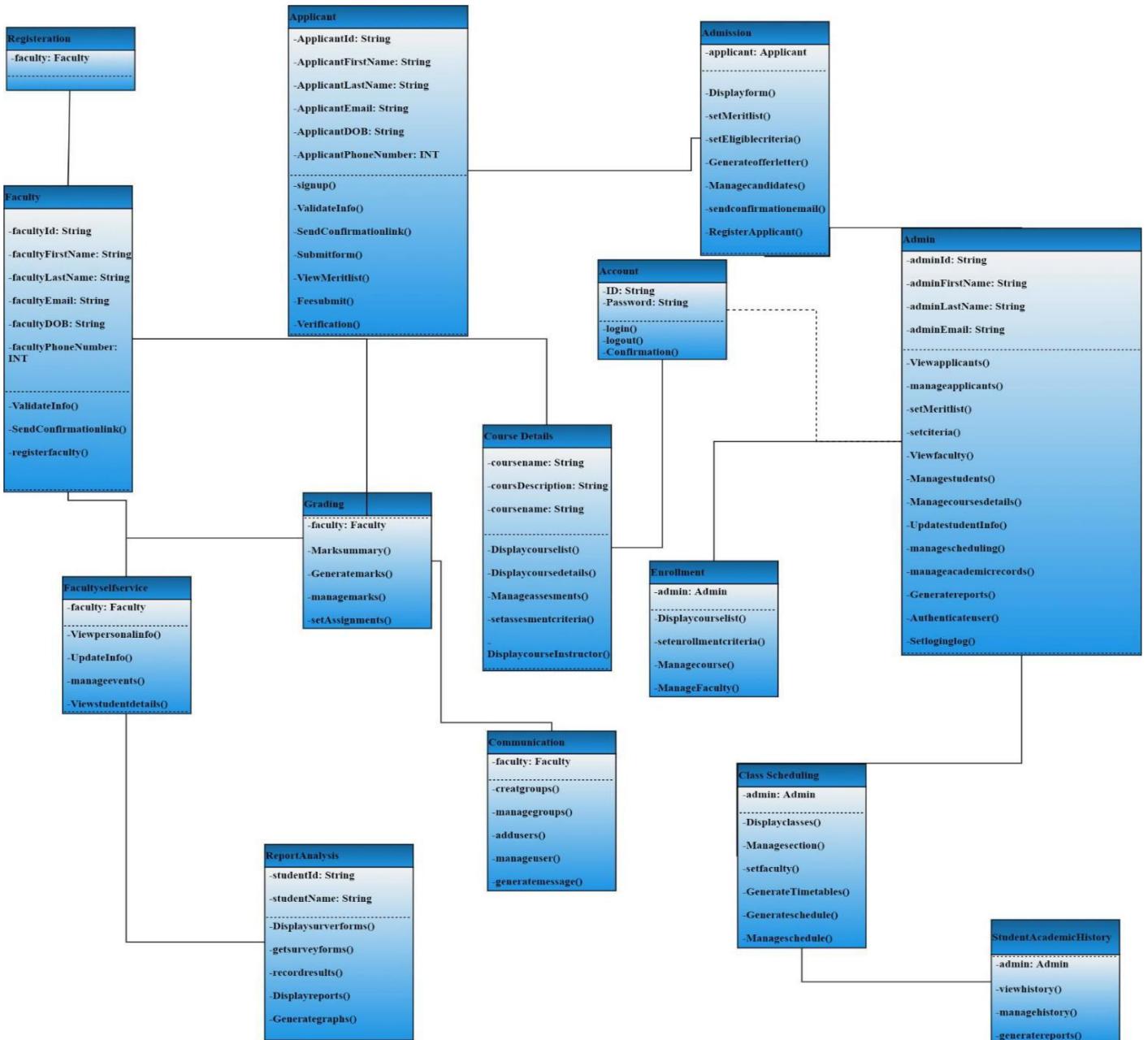


Figure 4-20 Class Diagram

5 Implementation

5.1 Algorithm

In our project, we used the object values for Aggregate Calculation Algorithm to calculate and push the aggregate data of the applicant depending on the test score he achieved during university test.

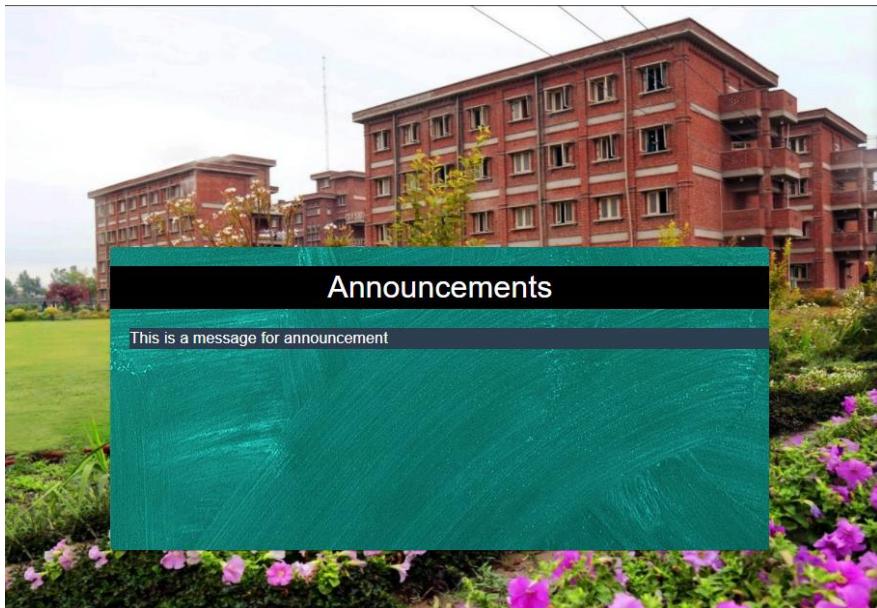
At first, we will do object field extraction from the Model and in which we will loop via the array applicant scores and check if they have already scored the intermediate or secondary marks or not. If no then the new calculated value will be appended to the default 0 value of the total test score. System will run the algorithm based on the test score feeded to the respective Applicant by Admin. Calculated results will be displayed below.

Pseudo Code:

```
applicantList.forEach(applicant => ({  
  
    applicant.sec_test_score = (applicant.secondary_education_obtained_marks /  
    applicant.secondary_education_total_marks) *100*0.1;  
  
    applicant.itm_test_score = (applicant.intermediate_education_obtained_marks /  
    applicant.intermediate_education_total_marks) *100*0.4;  
  
    applicant.uni_test_score = applicant.university_test_score * 0.5;  
  
    applicant.total_aggregate = applicant.sec_test_score + applicant.itm_test_score +  
    applicant.uni_test_score;  
  
    applicant.total_aggregate.toFixed(2);  
})
```

5.2 User Interface

5.2.1 Landing Page



Shaheed Benazir Bhutto University,
Peshawar

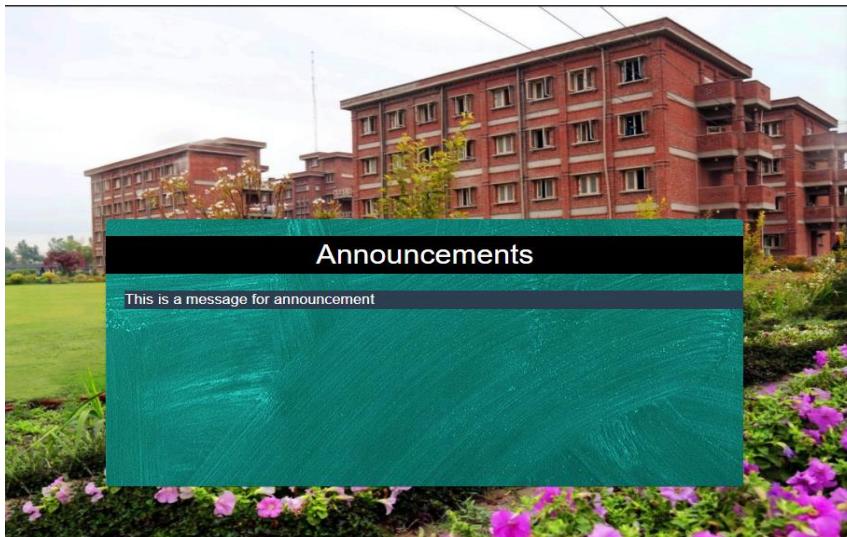


- [APPLICANT PORTAL](#)
- [FACULTY PORTAL](#)
- [COORDINATOR PORTAL](#)
- [ADMIN PORTAL](#)

Figure 5-1 Landing Page

5.2.2 Admin Portal

5.2.2.1 Login Page



Shaheed Benazir Bhutto University,
Peshawar



Sign in

[Not an Admin?](#)

Email Address *

Password *

Remember me

SIGN IN

Figure 5-2 Admin Login Page

5.2.2.2 Dashboard

The screenshot shows the Admin Portal dashboard. On the left is a sidebar with icons and labels: Schedule, Admission, Departments, Programs, Announcements, Merit List, Survey, and Settings. Below the sidebar is the URL: <https://bbu-cmas.herokuapp.com/admin>. The main area has a blue header bar with the text "Welcome". Below it are four orange and red buttons with "CLICK HERE" labels: "Manage Faculty", "Manage Coordinator", "Manage applicant", and "Manage Student". A yellow banner labeled "Calendar" contains a December 2020 calendar grid. The grid shows dates from Monday, December 30, to Sunday, January 6. A dark blue circular icon with a white speech bubble is located in the bottom right corner of the calendar area.

Figure 5-3 Admin Dashboard

5.2.2.3 Faculty

The screenshot shows the Admin Faculty page. The sidebar on the left is identical to the one in Figure 5-3. The main area has a blue header bar with the text "Manage Faculty" and a green "ADD FACULTY MEMBER" button. Below the header is a sub-header: "Below is a list of all the faculty members". A table lists six faculty members with columns for Name, Email, Designation, and a "VIEW" link. Each "VIEW" link is preceded by a small blue user icon. A dark blue circular icon with a white speech bubble is located in the bottom right corner of the table area.

Name	Email	Designation	
Noor us Sahar	noorussahr@gmail.com	noorussahr@gmail.com	VIEW
Asra Hashim	asrahashim@gmail.com	asrahashim@gmail.com	VIEW
kevin smith	kevin@gmail.com	kevin@gmail.com	VIEW
Syed Hashim Hussain	hashim@gmail.com	hashim@gmail.com	VIEW
sara smith	sara@gmail.com	sara@gmail.com	VIEW
sara smith	sara1@gmail.com	sara1@gmail.com	VIEW

Figure 5-4 Admin Faculty Page

Register Faculty

Fill in the information below to register a new Faculty member

Name *

Email *

Password *

SUBMIT

Figure 5-5 Admin Faculty Page 2

5.2.2.4 Coordinator

Manage Coordinators

Below is a list of all the coordinators

ADD COORDINATOR

S.No	Name	Email	Department	Actions
1	kevin smith	kevinsmith@gmail.com	CS	PROFILE REMOVE
2	Coordinator	coordinator@gmail.com	CS	PROFILE REMOVE

Figure 5-6 Admin Coordinator Page

The screenshot shows the 'Create Coordinator' page within the Admin Portal. The left sidebar contains links for Schedule, Admission, Departments, Programs, Announcements, Merit List, Survey, and Settings. The main content area has a blue header bar with the title 'Create Coordinator' and a sub-instruction 'Fill in the information below to register the coordinator'. Below this are four input fields: 'Name *' and 'Email *' in the top row, and 'Password *' and 'Confirm Password *' in the bottom row. There is also a 'Description *' field and a 'Department' dropdown menu. At the bottom are two buttons: a green 'SUBMIT' button and a blue 'BACK' button.

Figure 5-7 Admin Coordinator Page 2

5.2.2.5 *Applicants*

The screenshot shows the 'Manage Applicants' page within the Admin Portal. The left sidebar is identical to Figure 5-7. The main content area has a blue header bar with the title 'Manage Applicants' and a sub-instruction 'Below is the list of applicants that have applied this session'. Below this is a table with the following columns: S.No, Name, Father Name, Email, Secondary Education (%), Intermediate Education (%), and Actions. A dark blue circular icon with a white speech bubble is located in the bottom right corner of the main content area.

S.No	Name	Father Name	Email	Secondary Education (%)	Intermediate Education (%)	Actions
1	John Doe	John Doe Sr.	john.doe@example.com	85%	90%	

Figure 5-8 Admin Applicant Page

5.2.2.6 Student

The screenshot shows the 'Student's Academic Records' section of the Admin Portal. On the left, a sidebar lists various administrative functions: Schedule, Admission, Departments, Programs, Announcements, Merit List, Survey, and Settings. The main area displays a red button labeled 'Add Records in Student's Profiles' with a 'ADD INFORMATION' button below it. To the right is an orange button labeled 'Schedule' with a 'SCHEDULE' button below it. A red box contains the text 'Students' and a grey box contains 'VIEW ALL PROFILES'. Below these boxes, a message says 'No Profiles Found'. A blue header bar at the top reads 'Student's Academic Records'. A dark background image of a building is visible on the left.

Figure 5-9 Admin Student Page

The screenshot shows the 'Add Information to Student Profile' section. The sidebar on the left remains the same as in Figure 5-9. The main area has a blue header bar with the title 'Add Information to Student Profile' and a sub-instruction 'Add the fields'. It contains several input fields: 'Name *' (a large text input), 'Program *' (a large text input), 'Father's Name *' (a large text input), 'CNIC *' (a text input), 'Semester *' (a dropdown menu), 'Section *' (a dropdown menu), and 'Date of Birth *' (a text input). A blue header bar at the top reads 'Add Information to Student Profile'. A dark background image of a building is visible on the left.

Figure 5-10 Admin Student Page 2

5.2.2.7 Admission

The screenshot shows the 'Manage Admissions' section of the Admin Portal. On the left is a sidebar with icons for Schedule, Admission (which is highlighted in blue), Departments, Programs, Announcements, Merit List, Survey, and Settings. The main area has a title 'Manage Admissions' with the subtitle 'Create and enable/disable admission sessions'. Below this are two buttons: 'CREATE ADMISSION SESSION' and 'GENERATE MERIT LIST'. A table lists one admission session: S.No 1, Session Name 'Admission', Start Date '29-Oct-2020', End Date '24-Nov-2020'. Action buttons for each row include 'UPDATE' (green), 'DISABLE' (orange), and 'REMOVE' (red). A dark blue circular icon with a white speech bubble is in the bottom right corner.

Figure 5-11 Admin Admission Page

5.2.2.8 Department

The screenshot shows the 'Manage Departments' section of the Admin Portal. The sidebar is identical to Figure 5-11. The main area has a title 'Manage Departments' with the subtitle 'Below is a list of all the departments'. A button 'ADD DEPARTMENT' is at the top. A table lists four departments: 1. CS (Computer Science), 2. Mathematics, 3. Bio Chemistry, and 4. Microbiology. Each row includes 'COORDINATORS' (purple), 'PROGRAMS' (purple), and 'UPDATE' (green) buttons. A dark blue circular icon with a white speech bubble is in the bottom right corner.

Figure 5-12 Admin Department Page

5.2.2.9 *Programs*

The screenshot shows the 'Manage Programs' section of the Admin Portal. On the left, a sidebar lists various administrative functions: Schedule, Admission, Departments, Programs (which is selected and highlighted in blue), Announcements, Merit List, Survey, and Settings. The main content area has a blue header 'Manage Programs' with the sub-instruction 'Below is a list of all the programs offered to undergraduate and graduate applicants'. Below this are two tabs: 'UNDERGRADUATE PROGRAMS' (selected) and 'GRADUATE PROGRAMS'. Underneath the tabs are two buttons: 'ADD UNDERGRADUATE PROGRAM' (blue) and 'DISABLE ALL UNDERGRADUATE PROGRAMS' (orange). A message 'No programs found' is displayed. In the bottom right corner of the content area, there is a dark circular icon with a white speech bubble symbol.

Figure 5-13 Admin Program Page

5.2.2.10 *Announcement*

The screenshot shows the 'Manage Announcements' section of the Admin Portal. The sidebar on the left is identical to Figure 5-13, showing 'Announcements' as the selected item. The main content area has a blue header 'Manage Announcements' with the sub-instruction 'Create and enable/disable announcement feeds'. Below this is a button 'CREATE ANNOUNCEMENT FEED'. The central part of the page is a table with the following columns: S.No, Creator Name, Message, Start Date, End Date, and Actions. Two rows of data are present:

S.No	Creator Name	Message	Start Date	End Date	Actions
1	Noor us Sahar	This is a message for announcement	15-Dec-2020	15-Dec-2020	<button>UPDATE</button> <button>DISABLE</button> <button>REMOVE</button>
2	Coordinator	Congratulations !!! FYP is done	20-Dec-2020	20-Dec-2020	<button>UPDATE</button> <button>DISABLE</button> <button>REMOVE</button>

In the bottom right corner of the content area, there is a dark circular icon with a white speech bubble symbol.

Figure 5-14 Admin Announcement Page

5.2.2.11 Merit List

The screenshot shows the 'Manage Merit List' section of the Admin Portal. A message at the top states: 'Below is the merit list of all applicants who qualified for admission in this session. Here you can view their fee status and verify them for enrollment.' Below this, there are two tabs: 'UNDERGRADUATE LIST' (highlighted in blue) and 'GRADUATE LIST'. A small trash can icon is located to the right of the tabs. At the bottom, it says 'No data found'.

Figure 5-15 Admin Merit List Page

5.2.2.12 Survey

The screenshot shows the 'Surveys' section of the Admin Portal. It lists six survey types with 'VIEW' and 'GENERATE REPORT' buttons:

Survey Type	VIEW	GENERATE REPORT
Student Course Evaluation Questionnaire		
Survey of Graduating Students		
Research Student Progress Review Form		
Survey Of Department Offering Ph.D. Programs		
Alumni Survey		
Employer Survey		

Figure 5-16 Admin Survey Page

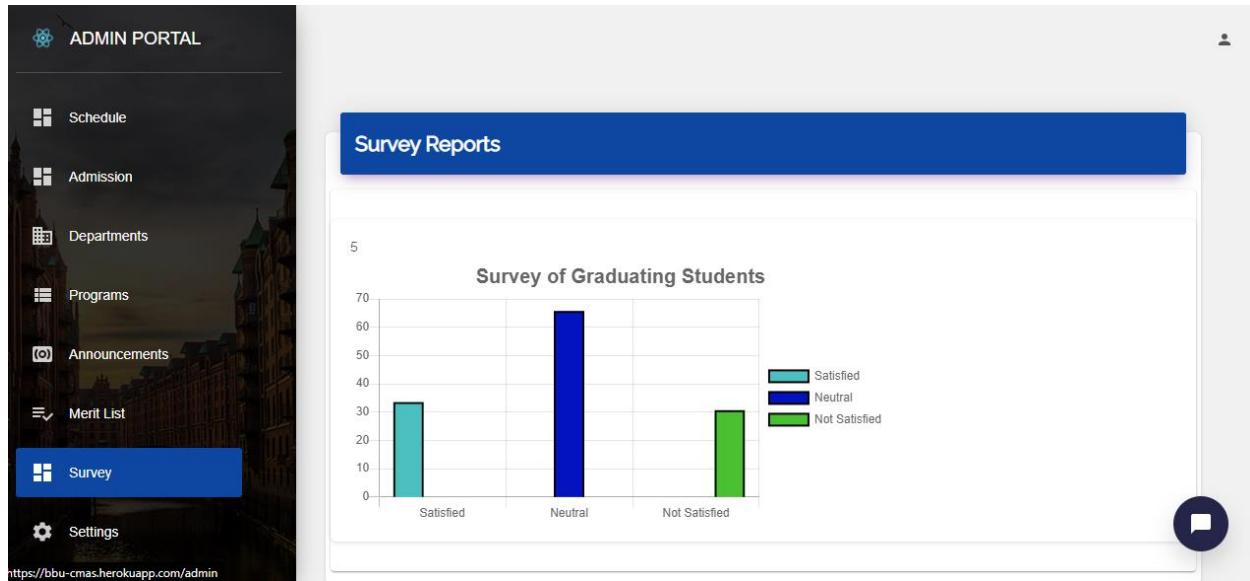


Figure 5-17 Admin Survey Page 2

5.2.2.13 *Settings*

The screenshot shows the 'Settings' page of the Admin Portal. It includes two main sections: 'Change Name' and 'Password'.

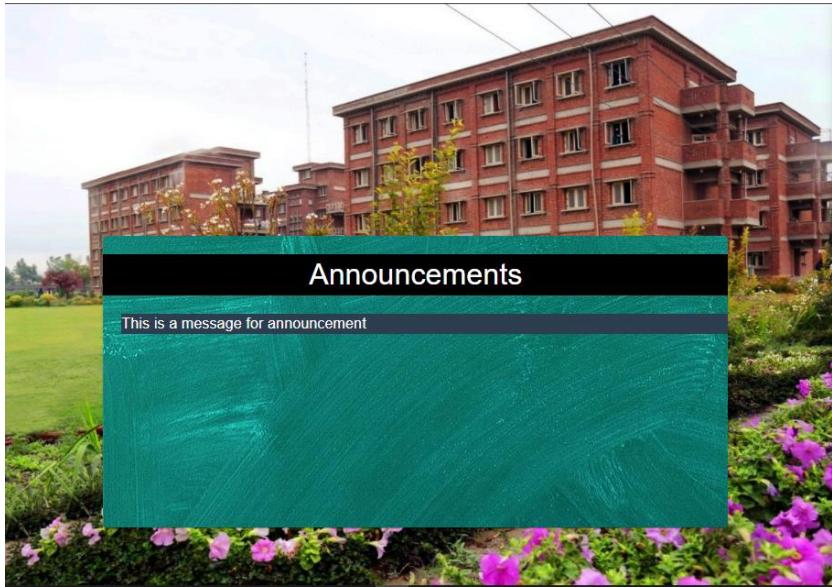
Change Name: A form to update your name. It has a 'Name' input field and a green 'UPDATE NAME' button.

Password: A form to update your password. It has 'Password' and 'Confirm Password' input fields, both marked with an asterisk, and a green 'UPDATE PASSWORD' button.

Figure 5-18 Admin Settings Page

5.2.3 Applicant Portal

5.2.3.1 Login Page



Shaheed Benazir Bhutto University,
Peshawar

 Sign in
Not an Applicant?

Email Address *

Password *

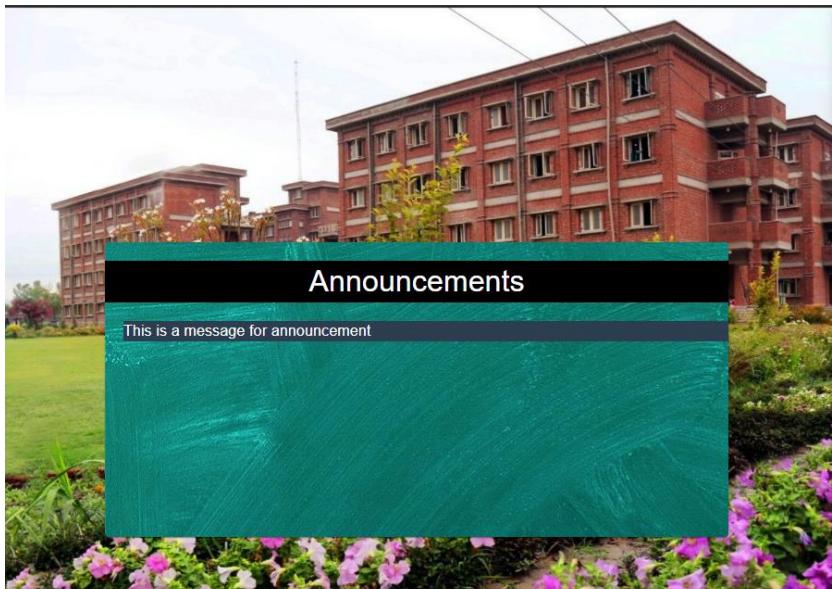
Remember me

SIGN IN

Don't have an account? [Sign Up](#)

Figure 5-19 Applicant Login Page

5.2.3.2 Registration Page



Shaheed Benazir Bhutto University,
Peshawar

 Sign Up
Not an Applicant?

Name *

Email *

Password *

Program applying for

REGISTER

Already have an account? [Sign In](#)

Figure 5-20 Applicant registration Page

5.2.3.3 Personal Details

The screenshot shows the 'Personal Details' section of the Applicant Portal. At the top, there's a blue header bar with the title 'Personal Details' and a sub-instruction 'Fill in the information below update your personal details'. Below this is a horizontal navigation bar with three tabs: 'Personal Details' (which is active and highlighted in blue), 'Income Details', and 'Education Details'. A large, light-gray placeholder icon of a person's head and shoulders is centered above the form fields. The form itself contains several input fields:

- 'Name *' and 'Father name *'
- 'CNIC / B-Form number *' with a 'Choose File' button and a note 'No file chosen'.
- 'CNIC / B-Form Back Picture' with a 'Choose File' button and a note 'No file chosen'.
- 'Email (For updates enter a valid email address)' containing the value 'applicant@gmail.com'.

On the right side of the page, there's a dark blue circular icon with a white speech bubble symbol.

Figure 5-21 Applicant Personal Details Page

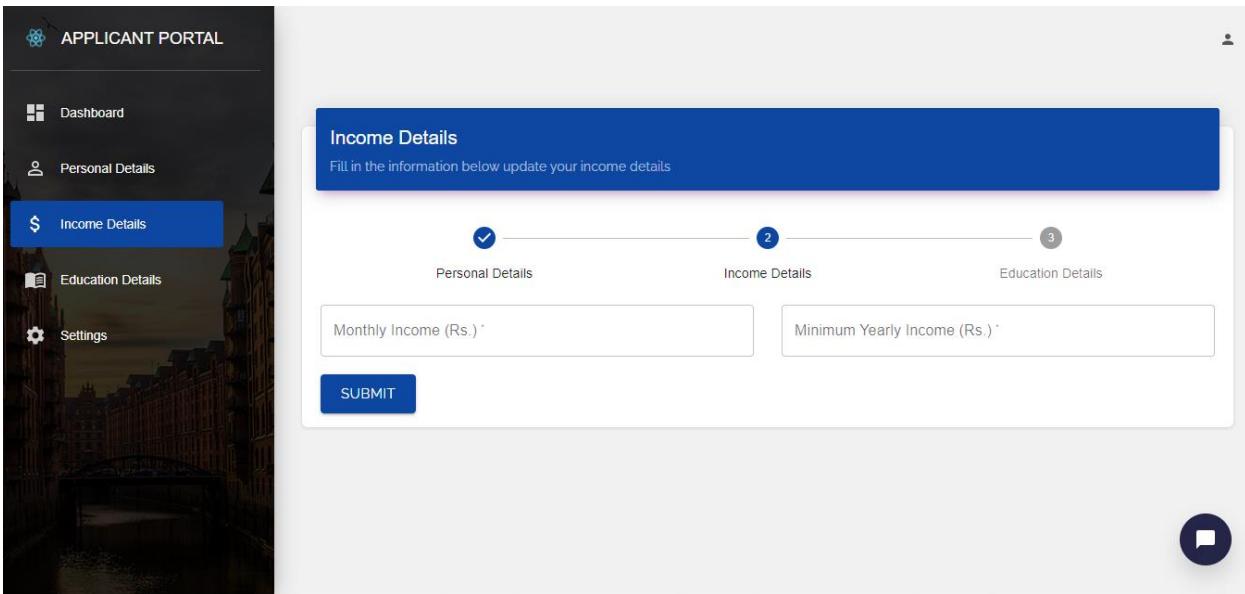
This screenshot shows the second page of the Applicant Personal Details process. The left sidebar remains the same as Figure 5-21. The main area displays the following data:

- 'CNIC / B-Form Front Picture' field containing the value '37404-6255523-8' with a 'Choose File' button and a note 'cnic front.jpg'. To its right is a thumbnail image of a Pakistani National Identity Card (CNIC) showing the front side.
- 'CNIC / B-Form Back Picture' field with a 'Choose File' button and a note 'cnic front.jpg'. To its right is a thumbnail image of the back side of the same CNIC card.
- 'Address *' field containing the value 'Pindi'.
- 'Contact No. *' field containing the value '03416999888'.
- 'Date of Birth *' field containing the value '20/12/2020'.
- 'Place of Birth *' field containing the value 'Islamabad'.
- 'Domicile *' field containing the value 'Islamabad'.

A large blue 'SUBMIT' button is located at the bottom left of the form area. On the right side, there's a dark blue circular icon with a white speech bubble symbol.

Figure 5-22 Applicant Personal Page 2

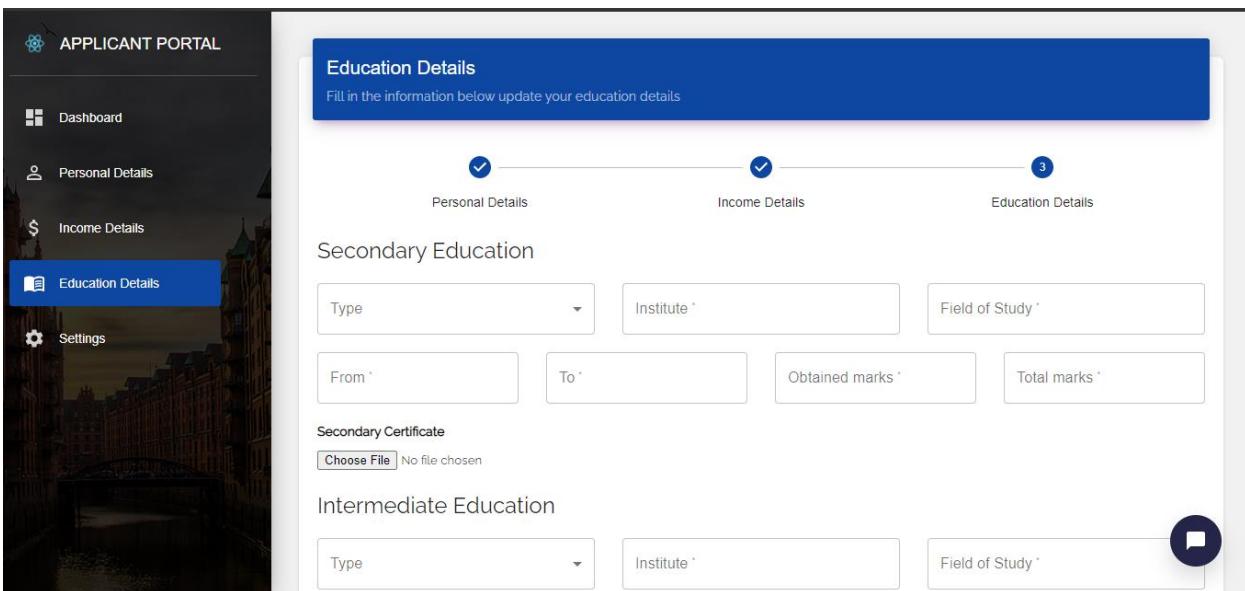
5.2.3.4 *Income Details*



The screenshot shows the 'Income Details' page of the Applicant Portal. The left sidebar has a dark background with icons for Dashboard, Personal Details, Income Details (which is highlighted in blue), Education Details, and Settings. The main content area has a white background with a blue header bar containing the title 'Income Details' and a sub-instruction 'Fill in the information below update your income details'. Below the header is a horizontal progress bar with three steps: 'Personal Details' (step 1, checked), 'Income Details' (step 2, checked), and 'Education Details' (step 3). There are two input fields: 'Monthly Income (Rs.) *' and 'Minimum Yearly Income (Rs.) *'. A blue 'SUBMIT' button is at the bottom left. A small blue speech bubble icon is in the bottom right corner.

Figure 5-23 Applicant Income Details Page

5.2.3.5 *Education Details*



The screenshot shows the 'Education Details' page of the Applicant Portal. The left sidebar has a dark background with icons for Dashboard, Personal Details, Income Details, Education Details (which is highlighted in blue), and Settings. The main content area has a white background with a blue header bar containing the title 'Education Details' and a sub-instruction 'Fill in the information below update your education details'. Below the header is a horizontal progress bar with three steps: 'Personal Details' (step 1, checked), 'Income Details' (step 2, checked), and 'Education Details' (step 3). The page is divided into sections for 'Secondary Education' and 'Intermediate Education'. Under Secondary Education, there are four input fields: 'Type' (dropdown), 'Institute' (text), 'Field of Study' (text), 'From' (text), 'To' (text), 'Obtained marks' (text), and 'Total marks' (text). Under Intermediate Education, there are three input fields: 'Type' (dropdown), 'Institute' (text), and 'Field of Study' (text). A blue speech bubble icon is in the bottom right corner.

Figure 5-24 Applicant Education Details Page

APPLICANT PORTAL

Secondary Education

Type Institute * Field of Study *

From * To * Obtained marks * Total marks *

Secondary Certificate

No file chosen

Intermediate Education

Type Institute * Field of Study *

From * To * Obtained marks * Total marks *

Intermediate Certificate

No file chosen

Figure 5-25 Applicants Education Page 2

5.2.3.6 *Settings*

APPLICANT PORTAL

Profile Picture

Upload your profile picture

No file chosen

Change Name

Figure 5-26 Applicants Settings Page

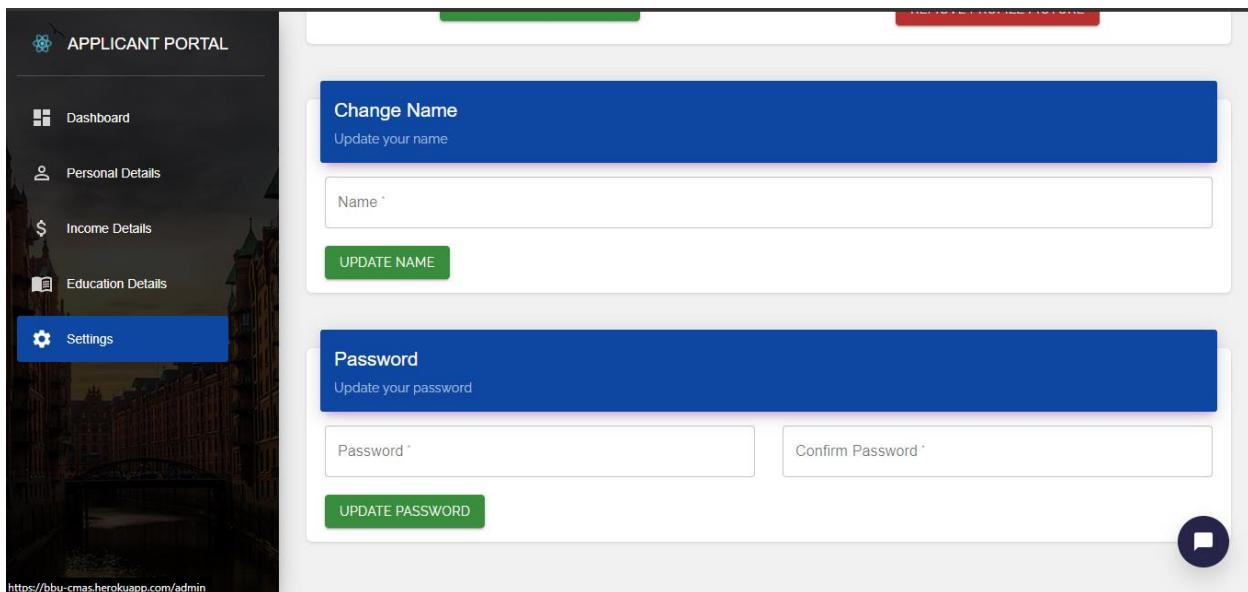


Figure 5-27 Applicants Settings Page 2

5.2.4 Coordinator Portal

5.2.4.1 Login Page

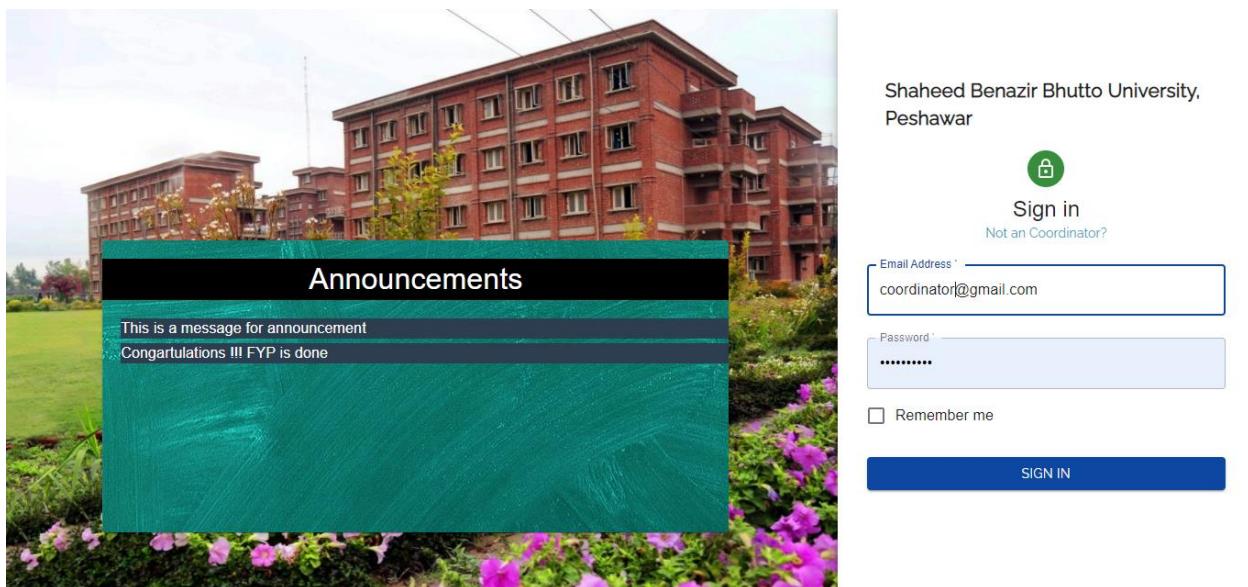


Figure 5-28 Coordinator Login Page

5.2.4.2 Dashboard/ Manage Enrollment Semester

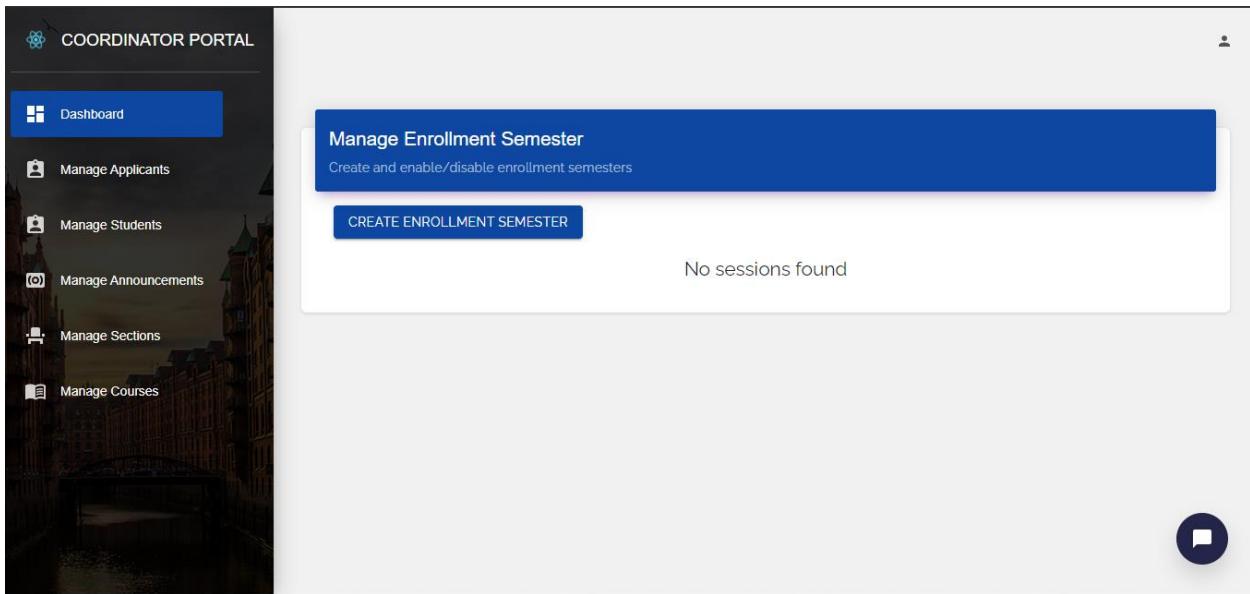


Figure 5-29 Coordinator Dashboard

5.2.4.3 Manage Applicants

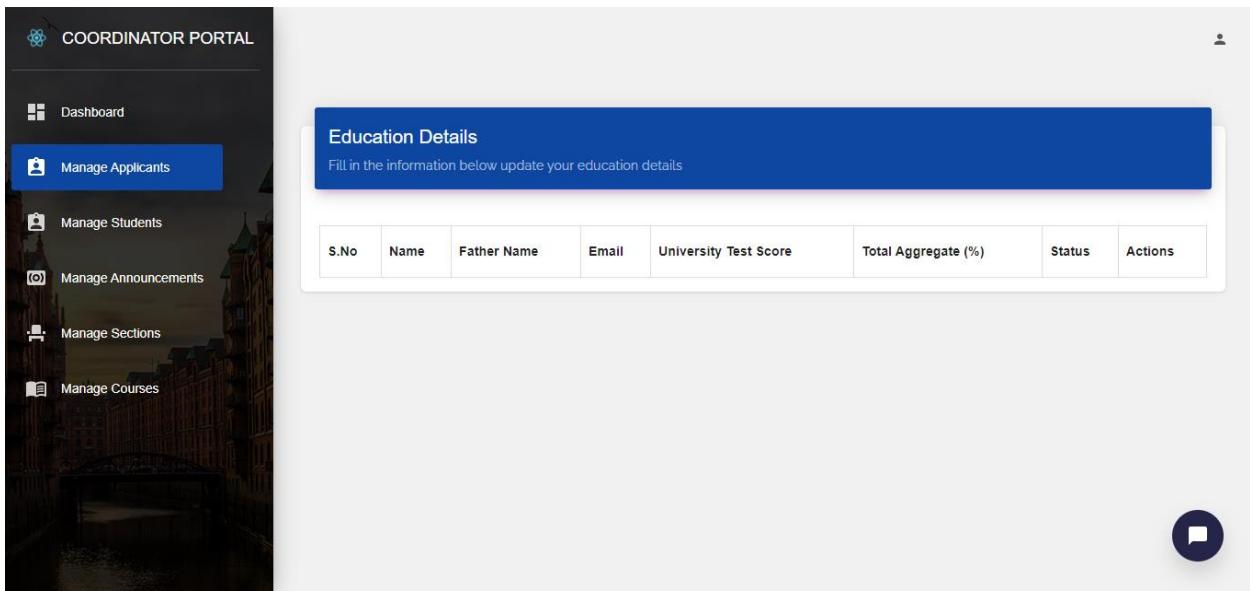
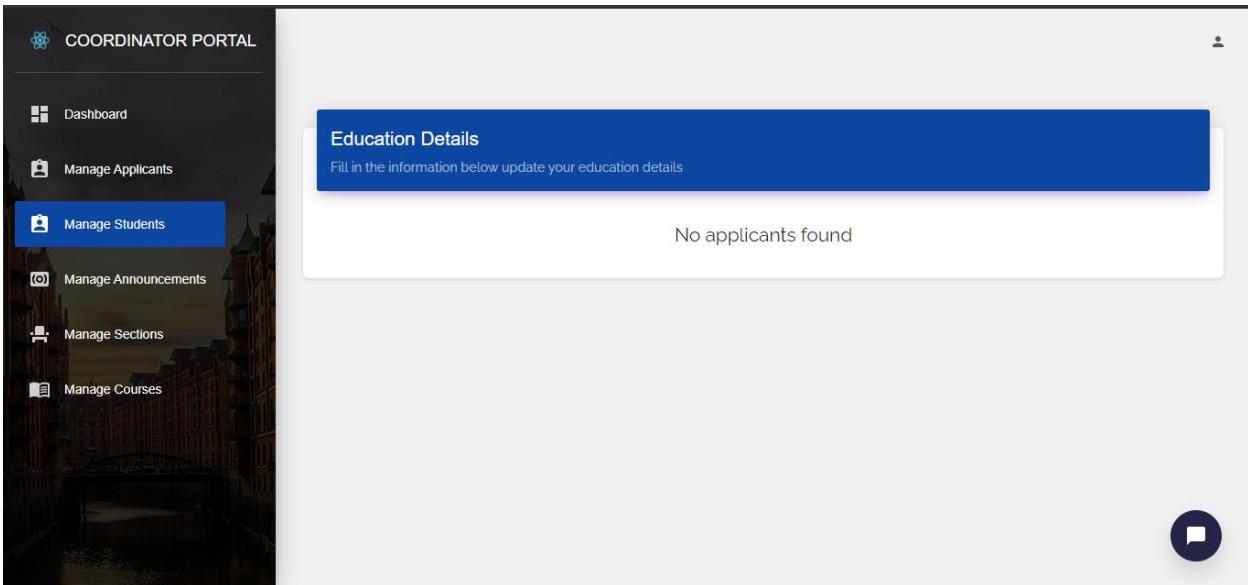


Figure 5-30 Coordinator Manage Applicants

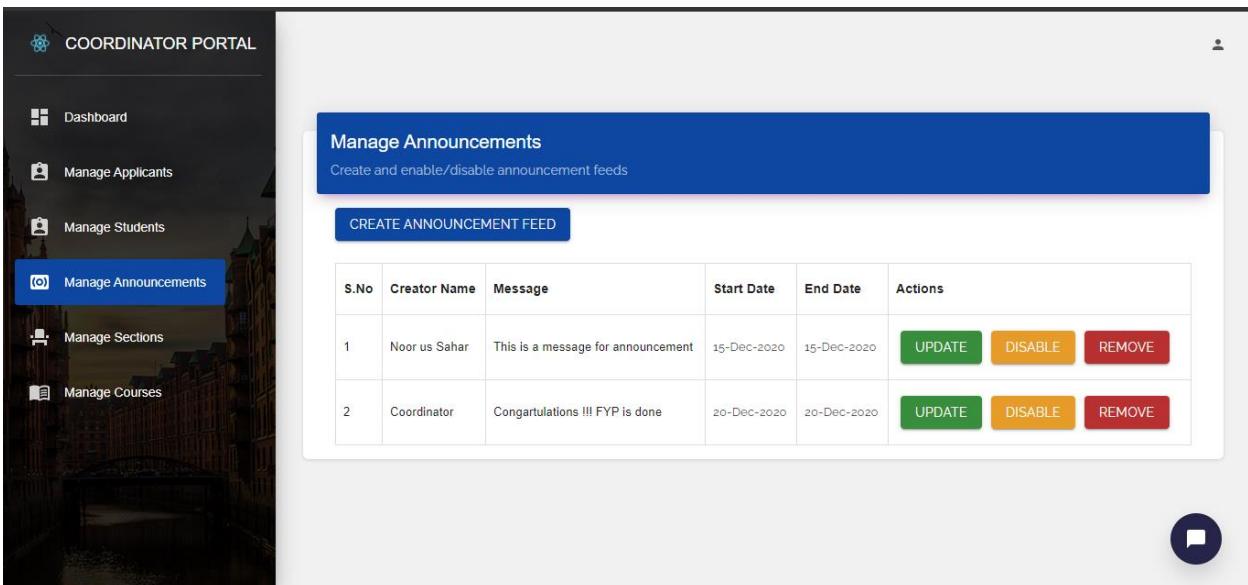
5.2.4.4 Manage Students



The screenshot shows the 'COORDINATOR PORTAL' interface. The left sidebar contains links: Dashboard, Manage Applicants, Manage Students (which is highlighted in blue), Manage Announcements, Manage Sections, and Manage Courses. The main content area has a blue header bar titled 'Education Details' with the sub-instruction 'Fill in the information below update your education details'. Below this, a message says 'No applicants found'. A dark background image of a building is visible behind the sidebar.

Figure 5-31 Coordinator Manage Students Page

5.2.4.5 Manage Announcements



The screenshot shows the 'COORDINATOR PORTAL' interface. The left sidebar contains links: Dashboard, Manage Applicants, Manage Students, Manage Announcements (which is highlighted in blue), Manage Sections, and Manage Courses. The main content area has a blue header bar titled 'Manage Announcements' with the sub-instruction 'Create and enable/disable announcement feeds'. Below this, a 'CREATE ANNOUNCEMENT FEED' button is visible. A table lists two announcements:

S.No	Creator Name	Message	Start Date	End Date	Actions
1	Noor us Sahar	This is a message for announcement	15-Dec-2020	15-Dec-2020	<button>UPDATE</button> <button>DISABLE</button> <button>REMOVE</button>
2	Coordinator	Congratulations !!! FYP is done	20-Dec-2020	20-Dec-2020	<button>UPDATE</button> <button>DISABLE</button> <button>REMOVE</button>

Figure 5-32 Coordinator Manage Announcements Page

5.2.4.6 *Manage Sections*

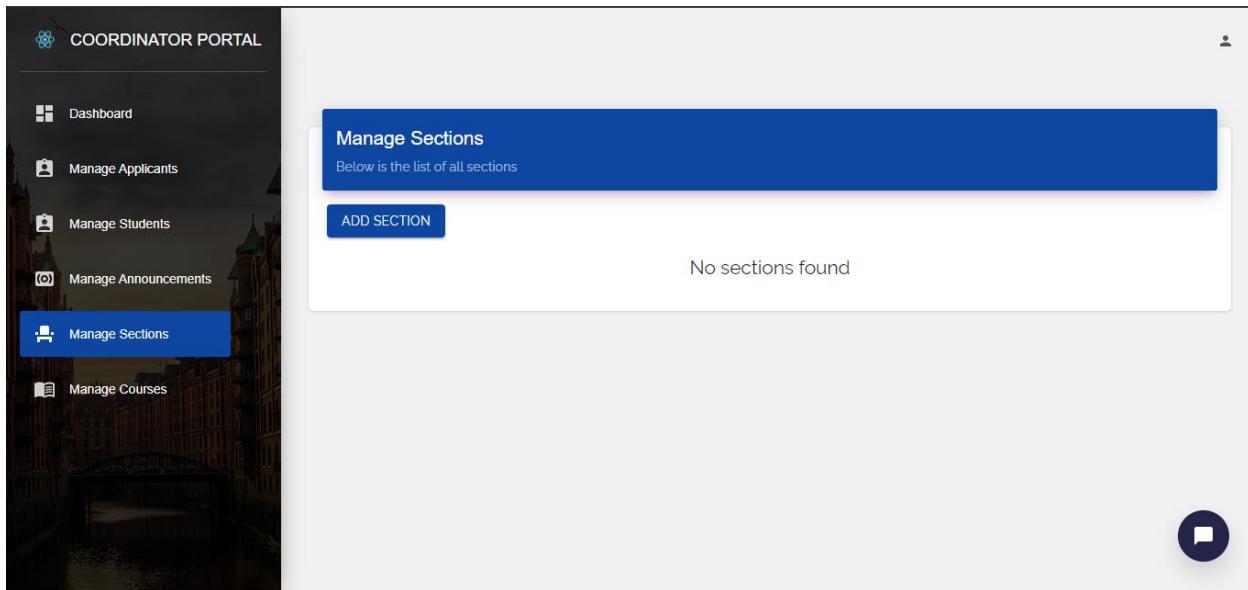


Figure 5-33 Coordinator Manage Sections

5.2.4.7 *Manage Courses*

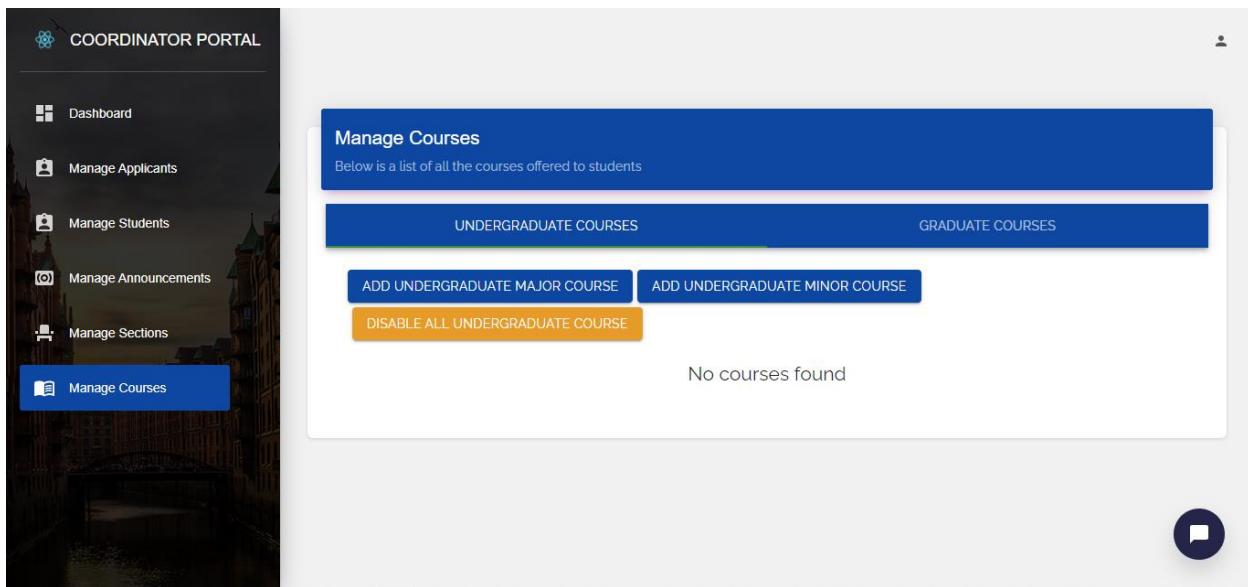


Figure 5-34 Coordinator Manage Courses

5.2.5 Faculty Portal

5.2.5.1 Login Page

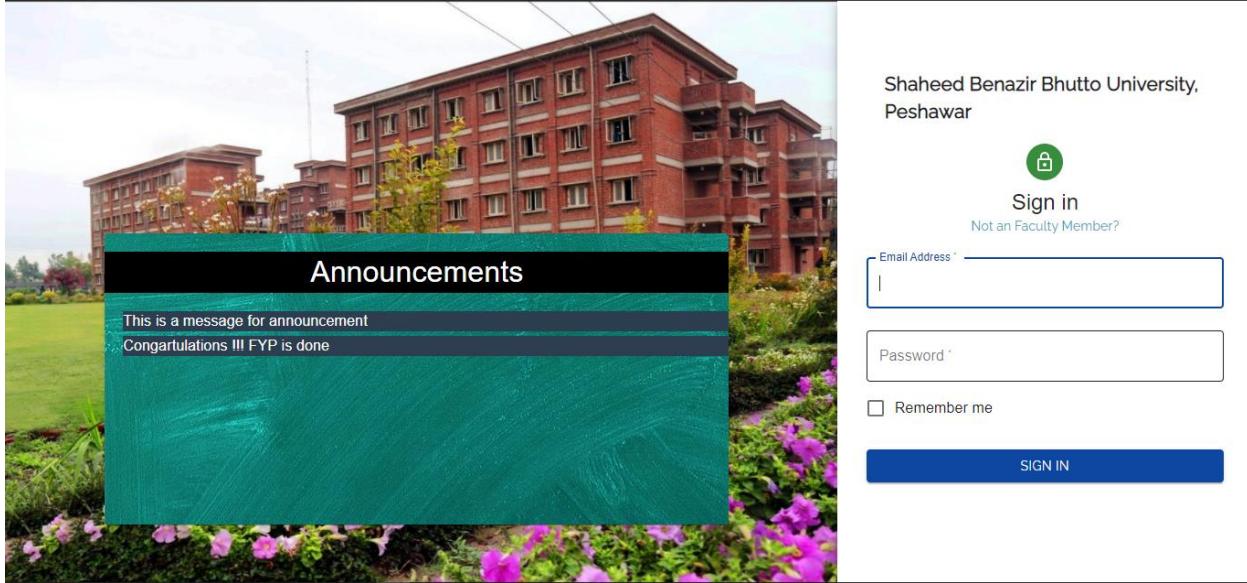


Figure 5-35 Faculty Login Page

5.2.5.2 Dashboard

The screenshot shows the Faculty Dashboard. On the left is a vertical sidebar with a dark background and white icons for "Dashboard", "Faculty Profiles", "Calendar", "Grades", and "Survey". The main area has a light gray background. At the top, a blue header bar displays the text "Welcome Noor us Sahar". Below the header, there are two main sections. The first section, titled "Profile" (highlighted in green), contains the user's name "Noor us Sahar", title "Assistant Professor", bio "Bio : My name is Noor us Sahar.", and teaching courses "Courses Teaching" listed as "vOOP" and "vDM". It also includes "VIEW PROFILE" and "EDIT PROFILE" buttons. The second section, titled "Add other Info In your Profile" (highlighted in cyan), contains three buttons: "ADD RESEARCH PAPERS", "ADD EDUCATION", and "ADD EXPERIENCE". At the bottom of the dashboard, there are three colored buttons: "Calendar" (orange), "Faculty Members" (red), and "VIEW ALL PROFILES" (white text on red).

Figure 5-36 Faculty Dashboard

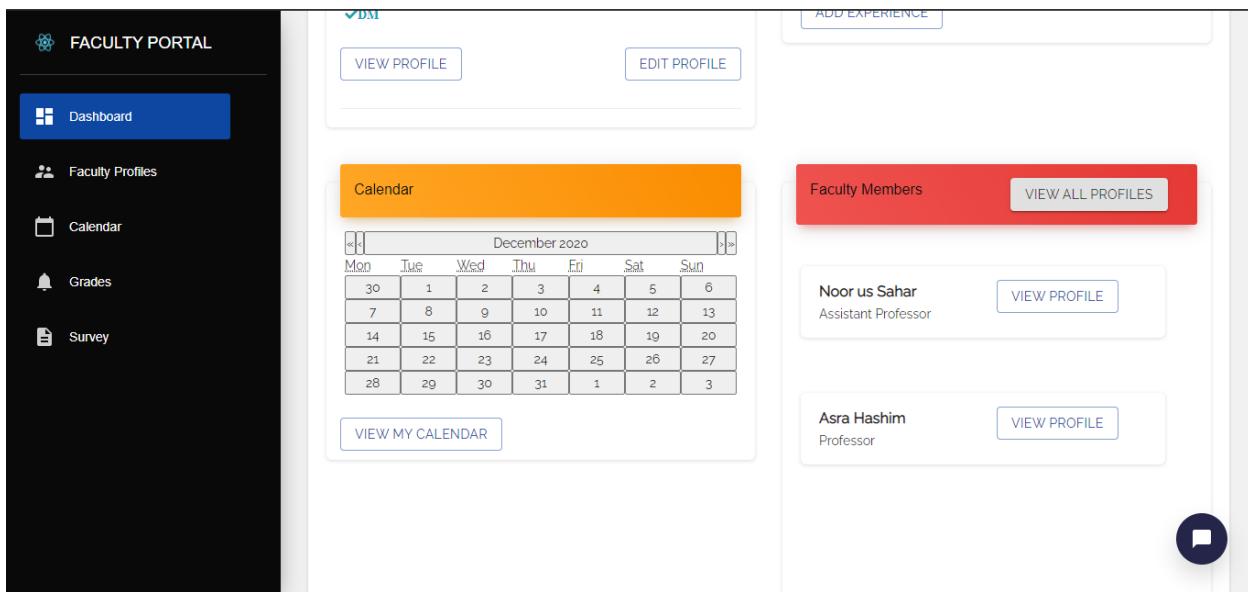


Figure 5-37 Faculty Dashboard 2

5.2.5.3 Faculty Profiles

Figure 5-38 Faculty Profiles Page

5.2.5.4 View Self Profile

The screenshot shows the Faculty Self Profile Page. On the left is a dark sidebar with the title 'FACULTY PORTAL' and five menu items: Dashboard, Faculty Profiles, Calendar, Grades, and Survey. The main content area features a blue header with a user icon and the name 'Noor us Sahar' followed by 'Assistant Professor'. Below the header is a message: 'My name is Noor us Sahar.' with a globe icon. A table titled 'Experience' lists two entries:

Company	Designation	From	To
yifkvvkj	developer	2020/10/29	2020/10/28
yifkvvkj	developer	2020/07/28	2020/08/20

A small dark circular icon with a white speech bubble is positioned to the right of the table.

Figure 5-39 Faculty Self Profile Page

5.2.5.5 Faculty Edit Profile

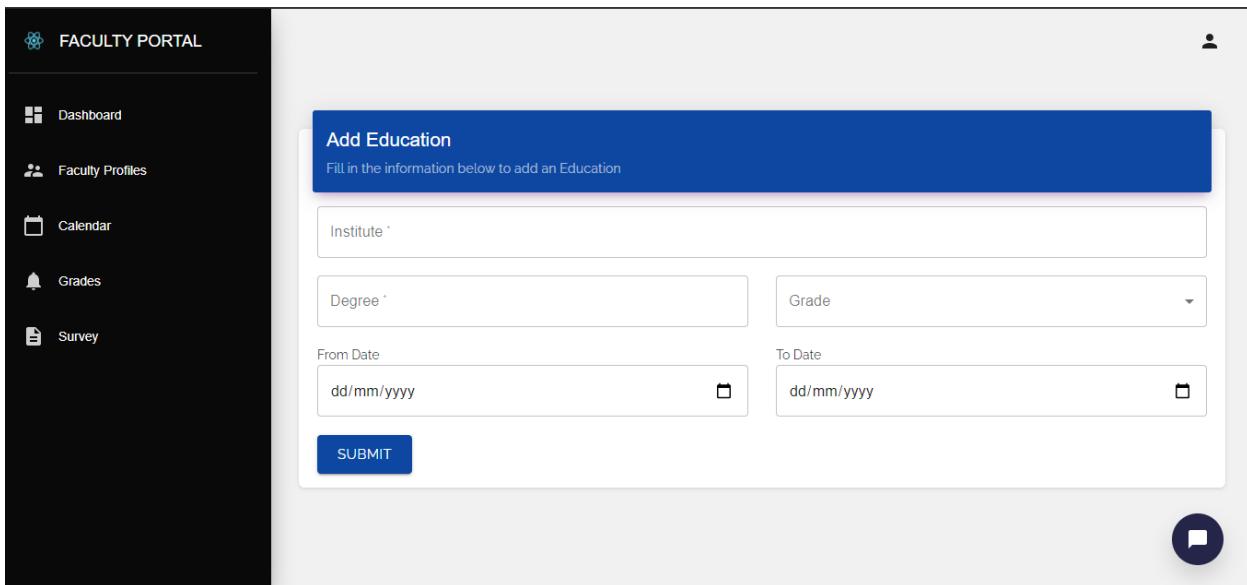
The screenshot shows the Faculty Edit Profile Page. The sidebar on the left is identical to Figure 5-39. The main area has a blue header with 'Your Profile' and 'Complete your profile' text. It includes tabs for 'ADD EXPERIENCE' and 'ADD RESEARCH PAPERS', and buttons for 'ADD EDUCATION' and 'SUBMIT'. The form fields are as follows:

- Designation: Assistant Professor
- Phone Number: 3416999888
- Website: <http://www.hotelviewforth.com>
- Address: Pindi
- Courses Teaching: OOP, DM
- Bio: My name is Noor us Sahar.

A small dark circular icon with a white speech bubble is positioned to the right of the bio field.

Figure 5-40 Faculty Edit Profile Page

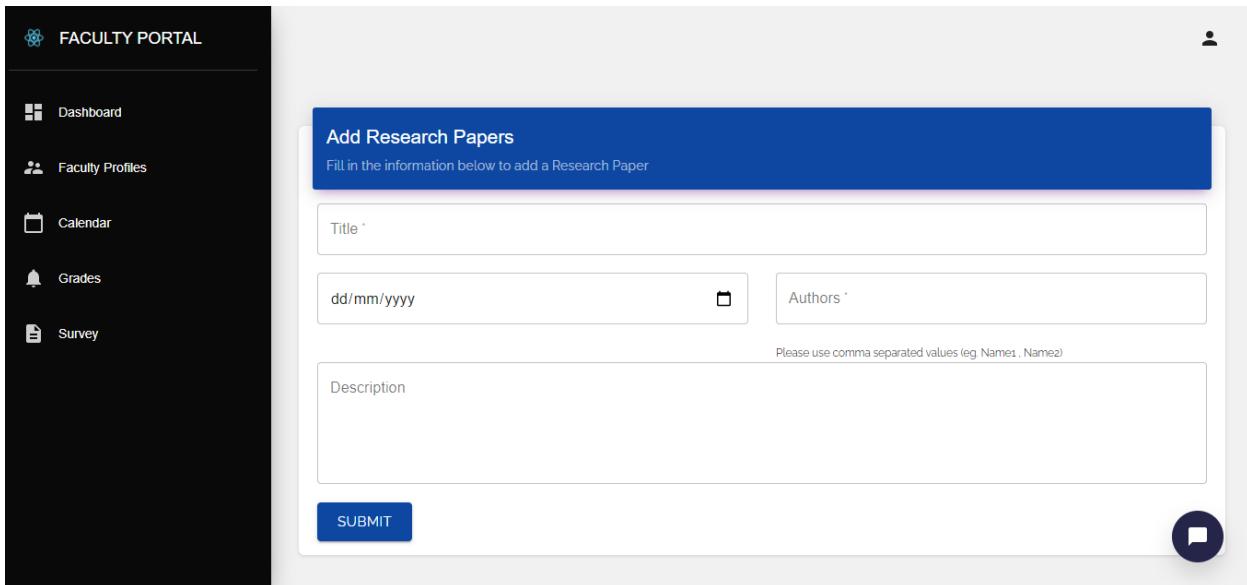
5.2.5.6 Add Education



The screenshot shows the 'Add Education' form within the Faculty Portal. The left sidebar contains links for Dashboard, Faculty Profiles, Calendar, Grades, and Survey. The main form has a blue header 'Add Education' with the sub-instruction 'Fill in the information below to add an Education'. It includes fields for 'Institute', 'Degree', 'Grade', 'From Date' (dd/mm/yyyy), and 'To Date' (dd/mm/yyyy). A 'SUBMIT' button is at the bottom.

Figure 5-41 Faculty Add Education Page

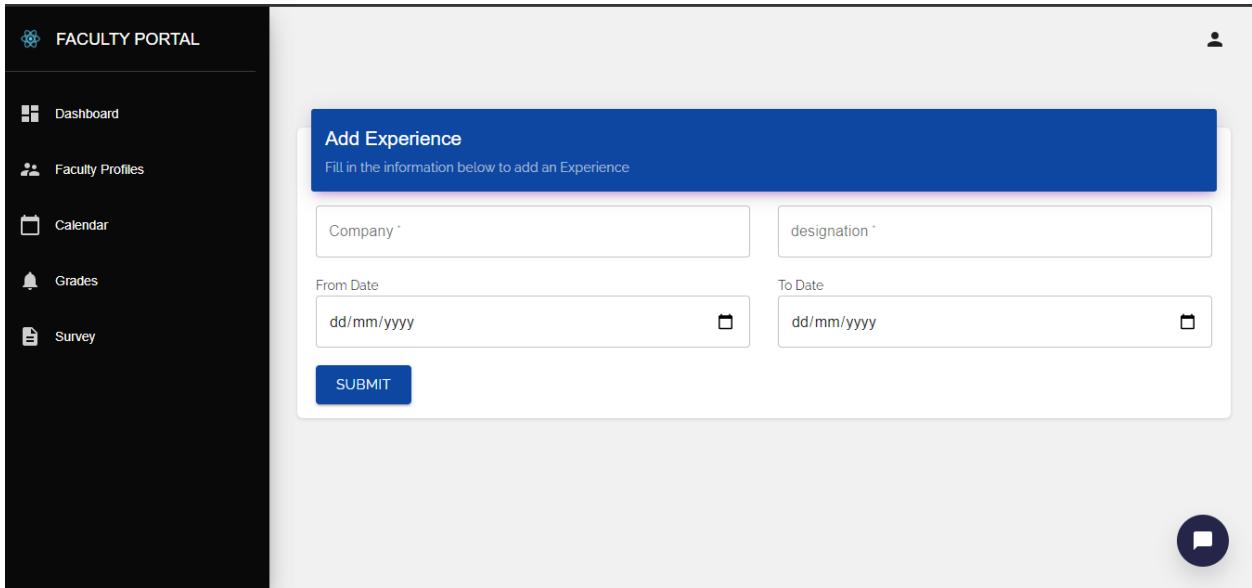
5.2.5.7 Add Research Paper



The screenshot shows the 'Add Research Papers' form within the Faculty Portal. The left sidebar contains links for Dashboard, Faculty Profiles, Calendar, Grades, and Survey. The main form has a blue header 'Add Research Papers' with the sub-instruction 'Fill in the information below to add a Research Paper'. It includes fields for 'Title', 'Date' (dd/mm/yyyy), 'Authors' (with a note: 'Please use comma separated values (eg. Name1, Name2)'), and 'Description'. A 'SUBMIT' button is at the bottom.

Figure 5-42 Faculty Add Research Paper Page

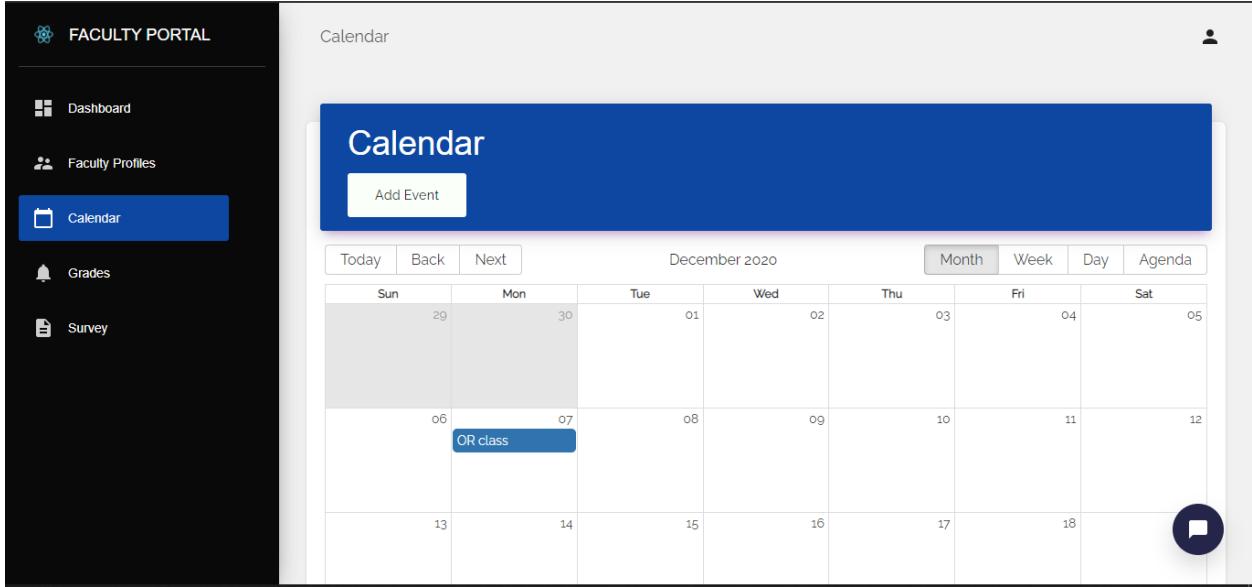
5.2.5.8 Add Experience



The screenshot shows the 'Add Experience' form within the Faculty Portal. The left sidebar includes links for Dashboard, Faculty Profiles, Calendar, Grades, and Survey. The main content area has a blue header 'Add Experience' with the sub-instruction 'Fill in the information below to add an Experience'. It contains four input fields: 'Company *' and 'designation *' (both required), 'From Date' (dd/mm/yyyy) with a calendar icon, and 'To Date' (dd/mm/yyyy) with a calendar icon. A large blue 'SUBMIT' button is at the bottom.

Figure 5-43 Faculty Add Experience Page

5.2.5.9 Calendar



The screenshot shows the Faculty Calendar page. The left sidebar has a 'Calendar' link highlighted in blue. The main content area features a blue header 'Calendar' with an 'Add Event' button. Below is a monthly calendar for December 2020. The days of the week are labeled from Sun to Sat. Specific dates are marked: Sunday 29, Monday 30, Tuesday 01, Wednesday 02, Thursday 03, Friday 04, Saturday 05, Sunday 06, Monday 07, Tuesday 08, Wednesday 09, Thursday 10, Friday 11, Saturday 12, Sunday 13, Monday 14, Tuesday 15, Wednesday 16, Thursday 17, Friday 18, and Saturday 19. An event titled 'OR class' is scheduled for Monday, December 7. Navigation buttons for Today, Back, Next, Month, Week, Day, and Agenda are available.

Figure 5-44 Faculty Calendar Page

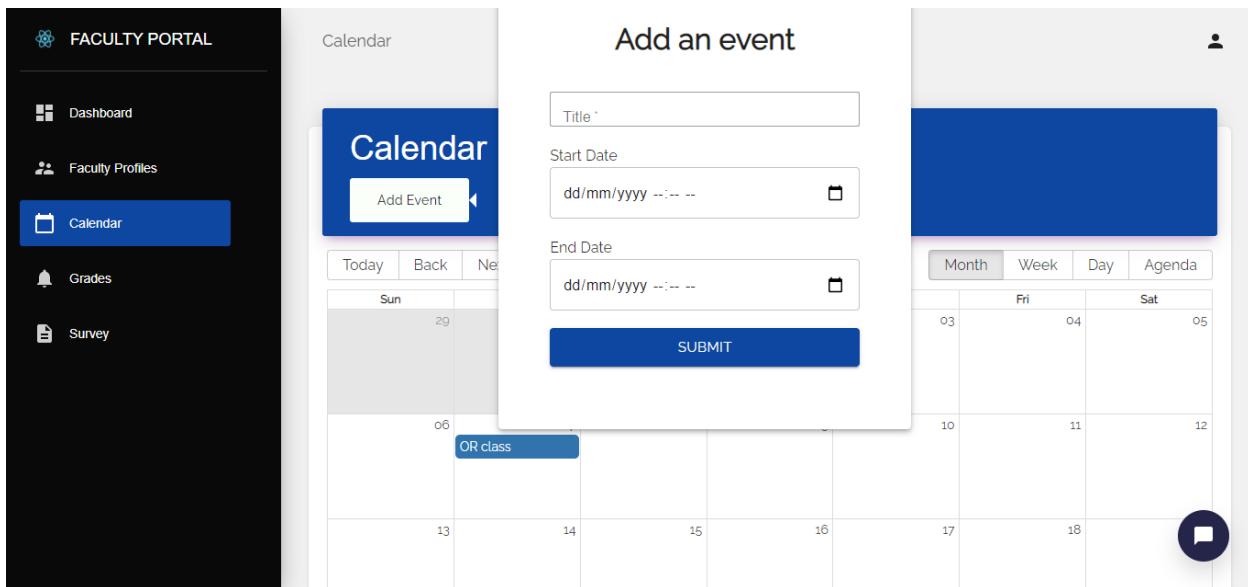


Figure 5-45 Faculty Calendar Page 2

5.2.5.10 *Grades*

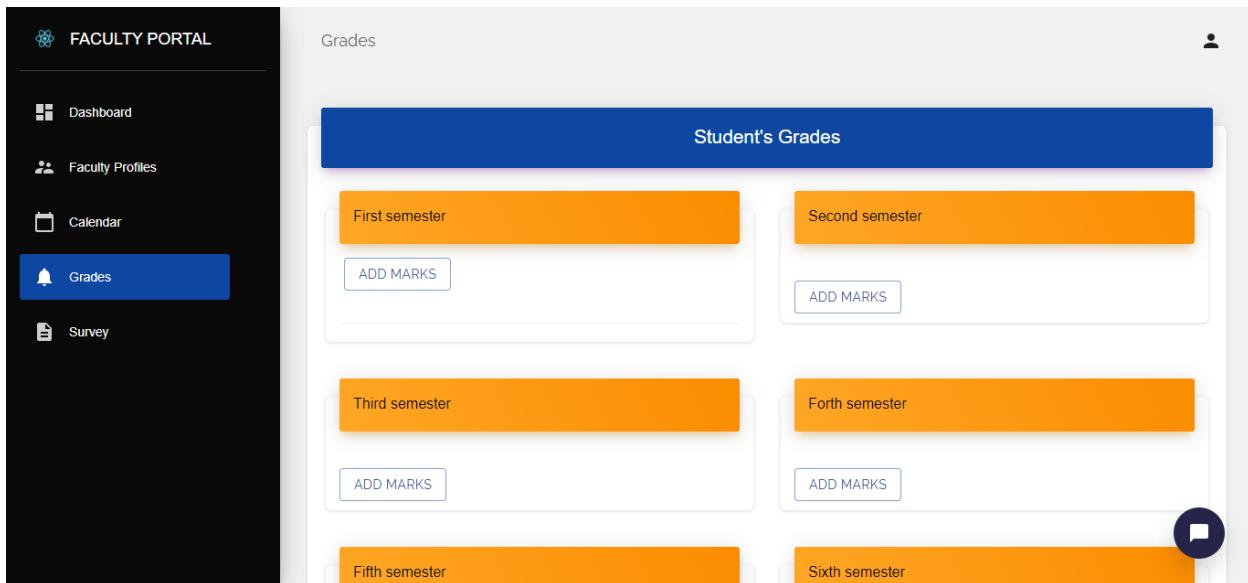


Figure 5-46 Faculty Grades Page 1

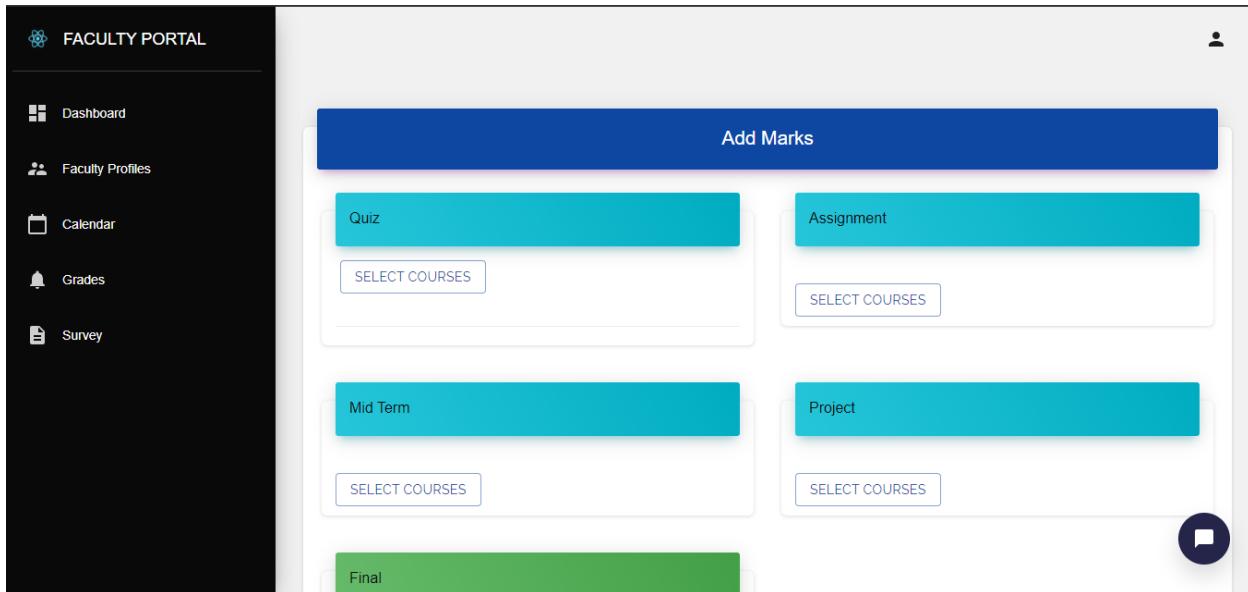


Figure 5-47 Faculty Grades Page 2

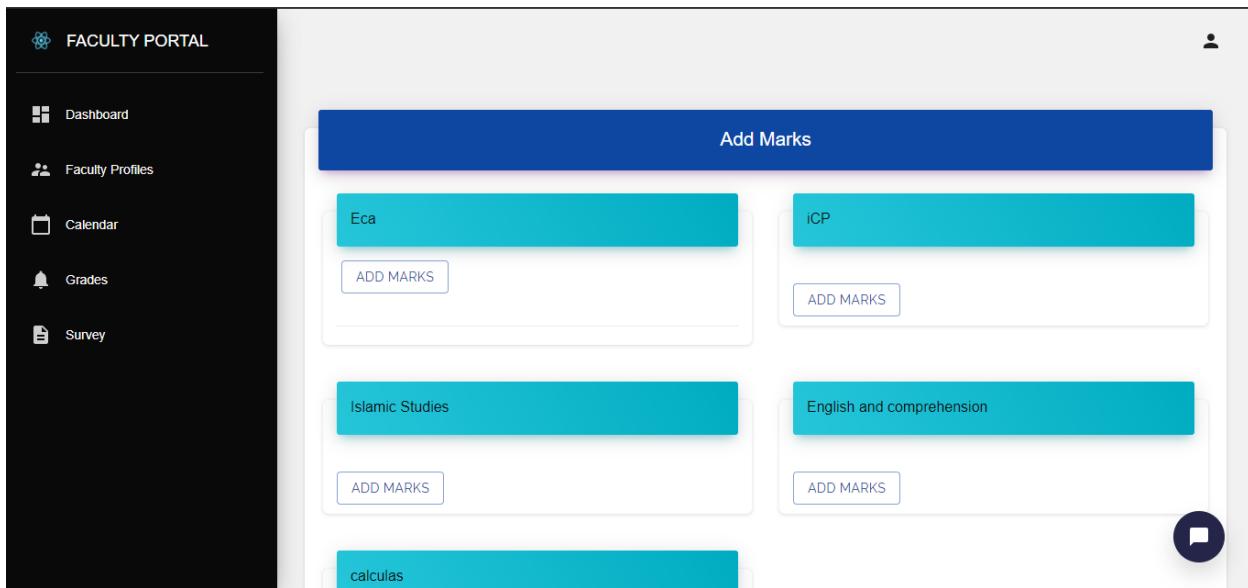


Figure 5-48 Faculty Grades Page 3

5.2.5.11 Survey

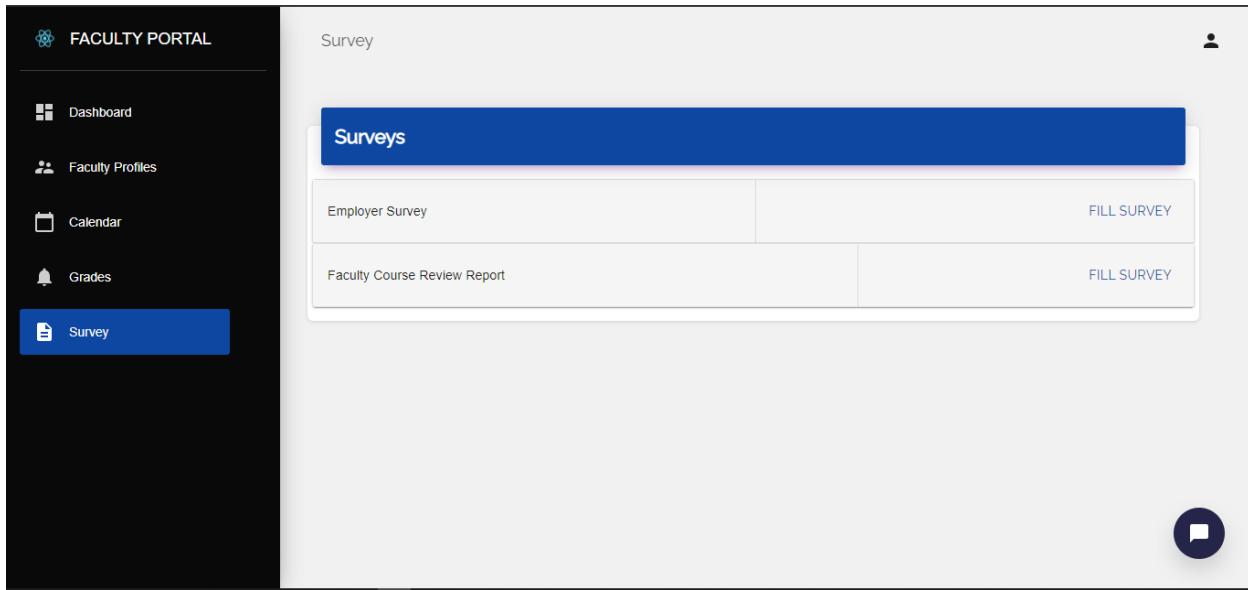


Figure 5-49 Faculty Survey Page

6 Testing and Evaluation

6.1 Manual Testing

6.1.1 Unit Testing

6.1.1.1 Create Account

Test Case ID	TC-1.1
Test Case Name	Check empty input fields
Test Data	Register form's fields
Pre-Condition	User is not currently registered
Actions	
User will click register with one or more empty fields	System shall display a message to fill in all the fields.
Result	
	User will fill in the fields and try again

Test verification: This test verifies the registration validity and account creation.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.2
Test Case Name	Check for valid email
Test Data	User email address
Pre-Condition	User shall have an existing email account
Actions	System Response
User will provide his email address	System shall verify email before creating an account if email exists, the account will be created.
Result	Users with an existing email will be able to create their account.

Test verification: This test verifies the email validity for a registration process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.3
Test Case Name	Successful Registration
Test Data	Registration fields
Pre-Condition	All the inputs are valid and entered in the form fields.
Actions	System Response
User will click on create account after filling the form.	The system shall redirect the ussr to their login form.
Result	No user's with the same email address can be registered.

Test verification: This test verifies the user can login to their portal after a successful registration process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.2 *Create Account (Applicant)*

Test Case ID	TC-1.4
Test Case Name	Check for the user to select a program they want to apply in.
Test Data	User's program they want to apply for
Pre-Condition	User shall not have any program selected earlier
Actions	System Response
Users will select a program they want to apply for.	Users shall check, if the user has selected any program, while they click on the register button.
Result	Users which have selected a program will be able to proceed with registration only.

Test verification: This test verifies the program has been selected by the user while registration.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.5
Test Case Name	Check for admissions open
Test Data	Admissions enabled
Pre-Condition	Admin must have enabled the admission sessions.
Actions	System Response
Applicants will click on apply now option.	System shall redirect the applicant to the register page.
Result	Applicants shall be able to register for applying into a program.

Test verification: This test case verifies that the applicant can apply for a program by registering their account.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.3 *Login*

Test Case ID	TC-1.6
Test Case Name	Disable login button upon submission
Test Data	Inputs
Pre-Condition	User must click login button once after entering the credentials
Actions	System Response
User will enter the required credentials and click submit button	System shall disable login button and process credential's validation
Result	The user shall not be able to interrupt validation process

Test verification: The user shall not be able to interfere or disrupt the validation process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.7
Test Case Name	Non-empty username and password
Test Data	Username and Password
Pre-Condition	User shall click login button
Actions	System Response
The user will click the login button with empty fields	System shall display notification on screen, requesting the user to enter their credentials
Result	The user will enter username and password

Test verification: This test assures that username and password fields are not left empty.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.8
Test Case Name	Invalid username and password
Test Data	Username and Password
Pre-Condition	User shall click login button

Actions	System Response
The user will click the login button with invalid credentials	System shall display an error on the screen of invalid credentials.
Result	The user will enter correct username and password

Test verification: This test assures that username and password fields are filled with correct input.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.9
Test Case Name	Redirect to homepage
Test Data	Username and Password
Pre-Condition	User shall click login button
Actions	System Response
The user will click the login button with valid credentials	System shall redirect the user to their respective portal
Result	The user will be able to access their portal

Test verification: This test insures that the user shall be able to access their respective portal

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.10
Test Case Name	Check for valid email address
Test Data	User email address
Pre-Condition	User shall have registered with the specific email address
Actions	System Response
User will provide his email address	System shall allow the user to access their portal
Result	Users with an existing email will be able to login to their account.

Test verification: This test verifies the email validity for a login process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.11
Test Case Name	Password length check
Test Data	Password
Pre-Condition	None
Actions	System Response
User will enter his/her password	System shall check if the password length is greater than or equal to 6
Result	Display the message indicating the user to enter a password with 6 or more characters.

Test verification: This test verifies the email validity for a login process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.4 *Settings*

Test Case ID	TC-1.12
Test Case Name	Upload profile picture success
Test Data	Profile picture
Pre-Condition	User has been logged into their portal User has selected a profile picture
Actions	System Response
User clicks on the choose profile picture option.	System will have an upload profile picture button, which after uploading, the user shall click in order to proceed with the successful profile picture upload process.
Result	User profile picture will be uploaded successfully.

Test verification: This test case verifies users can successfully upload the profile picture.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.13
Test Case Name	Upload profile picture failure
Test Data	Profile picture
Pre-Condition	User has been logged into their portal User has selected a profile picture
Actions	System Response
User clicks on the choose profile picture option. User clicks on the upload profile picture button.	System displays the error message the profile picture size is too large to be uploaded.
Result	System redirects the user to upload the profile picture of size below 10Mb.

Test verification: This test case verifies the system indicates the user to upload a profile picture having a smaller size.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.14
Test Case Name	Change name successful
Test Data	Name
Pre-Condition	User has been logged in User has entered a name while registration process
Actions	System Response
User clicks on change the name after entering their desired name	System will display the response of the user name has been changed
Result	Users name will be updated successfully

Test verification: This test case verifies users can successfully update their name.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.15
Test Case Name	Change name failed
Test Data	Name input field
Pre-Condition	User has not yet entered a name in the field
Actions	System Response
User clicks on update name button	System shall notify the user with an error to enter the required name to be updated
Result	User shall try again after entering a name for updating

Test verification: This test case ensures that the name input field is filled.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.16
Test Case Name	Change password success
Test Data	Password
Pre-Condition	None
Actions	System Response
User clicks on update password button after completing the required fields for password update	System shall display the response that the password has been changed
Result	Users password shall be updated successfully

Test verification: This test case ensures that the user password can be updated.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.17
Test Case Name	Change password failure
Test Data	Password
Pre-Condition	None
Actions	System Response

User will enter the password. User will enter the confirm password and click on change password	System shall notify the user that the passwords do not match.
Result	Users will enter the matching password and try again.

Test verification: This test case ensures that the user password can be updated.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.5 *Dashboard (Applicant)*

Test Case ID	TC-1.18
Test Case Name	Notification panel
Test Data	List of notifications
Pre-Condition	Admin has made any announcements.
Actions	System Response
User will log into their account	System shall display no messages if there are no announcements done from the admin side.
Result	Applicants will be notified if there are no notifications.

Test verification: This test case verifies that there are some notifications on the applicant side.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.19
Test Case Name	Apply for program success
Test Data	Programs
Pre-Condition	Applicant has not yet applied for any programs earlier
Actions	System Response
Applicants will click on either two of the buttons, depending on the program they wish to apply for.	System will redirect applicants to the program list page.

Result	Applicants shall be able to choose from the list of programs they have to apply for.
---------------	--

Test verification: This test case verifies applicants are able to apply for a suitable program they wish for.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.20
Test Case Name	Apply for program failure
Test Data	Programs
Pre-Condition	Applicant has already applied for the program
Actions	System Response
Applicant will click on the apply now button	System will disable the button for applying into a program
Result	Applicants shall be displayed a pop-up message that your application is already under process. You can not apply before the next session.

Test verification: This test case verifies applicants are able to apply for a suitable program they wish for.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.21
Test Case Name	Display application details of applicant
Test Data	Applicant form details
Pre-Condition	Applicant has already entered the form details
Actions	System Response
Applicant will input the application data	System will display the application data in the applicant details section
Result	Applicants will be able to see the form data that has been processed for evaluation on admission.

Test verification: This test case verifies applicants are able to view the data they have entered in the application form during applying for a program.

Testing Environment: Google Chrome
Tested by: Raja Shaiyan Azad Khan
Date: 10th December, 2020

Test Case ID	TC-1.22
Test Case Name	Apply for programs success
Test Data	Programs
Pre-Condition	Admin has enabled the programs to apply for
Actions	System Response
Applicant will select the programs from the programs list	System will add the programs to the applied programs list of the particular admin
Result	Applicant on pressing the proceed button shall be redirected to the Application form filling page.

Test verification: This test case verifies applicants are able to select the programs they wish to apply for.

Testing Environment: Google Chrome
Tested by: Raja Shaiyan Azad Khan
Date: 10th December, 2020

Test Case ID	TC-1.23
Test Case Name	Apply for programs failure
Test Data	Programs
Pre-Condition	Admin has disabled the programs
Actions	System Response
Applicant will select the program he wants to apply in	System will notify the applicant that there are no programs found with a back button.
Result	Applicant can go back to the dashboard. Applicant can recheck again for the program list to be displayed.

Test verification: This test case verifies applicants are able to select the programs they wish to apply for.

Testing Environment: Google Chrome
Tested by: Raja Shaiyan Azad Khan
Date: 10th December, 2020

Test Case ID	TC-1.24
Test Case Name	Program List
Test Data	Application input fields
Pre-Condition	Applicant has selected the programs he is applying for.
Actions	System Response
Applicant inputs in the required form fields.	System shall notify the applicant for the form status above. Indicating step 1 as the personal details section, step 2 as income details section and step 3 as education details section.
Result	Applicants can proceed onto filling in the income details section indicated by status bar.

Test verification: This test case verifies applicants are fill in their personal details form.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.25
Test Case Name	Check for empty application form input fields
Test Data	Empty application input fields
Pre-Condition	Application data has not been submitted completely
Actions	System Response
Applicant leaves an input field as incomplete.	System shall not proceed to the next step indicating the applicant to fill in all the input fields.
Result	Applicant will fill in the missing field and then try again.

Test verification: This test case verifies applicant fills in all the input fields of the application form.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.26
Test Case Name	Successful form submission
Test Data	Form completion status

Pre-Condition	Application data has been filled in completely.
Actions	System Response
Applicant clicks on the submit application form button and confirms the submission confirmation.	System shall display the applicant with a message of application submission successful.
Result	Applicants can now view the entry test venue.

Test verification: This test case verifies applicants can successfully apply for the program.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.27
Test Case Name	Successful form submission
Test Data	Form completion status
Pre-Condition	Application data has been filled in completely.
Actions	System Response
Applicant clicks on the submit application form button and confirms the submission confirmation.	System shall display the applicant with a message of application submission successful.
Result	Applicants can now view the entry test venue.

Test verification: This test case verifies applicants can successfully apply for the program.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.28
Test Case Name	View admission challan form
Test Data	Admission challan status
Pre-Condition	Admin has enabled the admission challan form Applicant is verified by the coordinator
Actions	System Response
Applicant clicks on the notification in the panel for printing the challan from	System redirects the applicant to a new webpage. System generates the challan form with a random

	PV and Bank number.
Result	Applicant can print the challan and fill their name and id number in order to assure the transaction is done successfully.

Test verification: This test case ensures that the admission fee challan for the semester can be printed by the applicant.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.29
Test Case Name	Upload admission challan form
Test Data	Admission challan submission status
Pre-Condition	Applicant has paid the fee amount
Actions	System Response
Applicant uploads the paid copy of the challan form	System will notify the applicant that challan has been uploaded successfully.
Result	Applicant will now be ready to get registered as student by coordinator.

Test verification: This test case ensures that the applicant is registered as a student.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.30
Test Case Name	Challan Form Verification
Test Data	Challan Form Data
Pre-Condition	Applicant have applied for a program
Actions	System Response
Applicant uploads the paid challan form.	System verifies that the challan form uploaded has been paid.
Result	System Fails to verify the paid Challan form

Test verification: This test case fails to ensure that the challan form submitted is verified

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan
Date : 10th December, 2020

6.1.1.6 *Dashboard (Admin)*

Test Case ID	TC-1.31
Test Case Name	Create admission session success
Test Data	Session name
Pre-Condition	Session has not been created before
Actions	System Response
Admin shall click on the create session button on dashboard.	System shall redirect the admin to a page. Admin shall enter the session name, start date and end date.
Result	System will notify the admin the admission session has been created successfully.

Test verification: This test case verifies the admission session is created successfully.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.32
Test Case Name	Create admission session failed
Test Data	Session name
Pre-Condition	Session has been created before
Actions	System Response
Admin enters the required fields and clicks on the create button.	System shall notify the admin that the session has already been created.
Result	Admin shall remove the current session and try again.

Test verification: This test case ensures that there is only 1 session per year.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.33
Test Case Name	Update admission session
Test Data	Input fields
Pre-Condition	Session has been already created
Actions	System Response
Admin clicks on the update session button. Admin fills in the session update form. Admin clicks on the update button.	System shall notify the admin that the session has been updated.
Result	Admin shall be able to view the updated session details.

Test verification: This test case ensures that the session details are updated.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.34
Test Case Name	Enable session
Test Data	Session status
Pre-Condition	Session has been created
Actions	System Response
Admin clicks on the enable session button.	System shall notify the admin that the session is enabled.
Result	Admin shall be able to disable the session later. Applicants shall be able to view the option to apply for the programs. System shall show the disable session button.

Test verification: This test case ensures it is enabled by the admin so that the applicants can apply for the program.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.35
---------------------	---------

Test Case Name	Disable session
Test Data	Session status
Pre-Condition	Session has been enabled
Actions	System Response
Admin clicks on the disable session button.	System shall notify the admin that the session is currently enabled
Result	Admin shall be able to enable the session later. Applicants shall be able to view the option to apply for the only programs that are enabled. System shall show the enable session button.

Test verification: This test case ensures programs are disabled and not offered to students.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.36
Test Case Name	Remove session
Test Data	Session name
Pre-Condition	Session has been created
Actions	System Response
Admin clicks on remove session button	System shall confirm the remove session process. Applicant confirms the removal of session.
Result	System removes the admission session.

Test verification: This test case ensures that the admin can remove the admission session they created earlier.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.37
Test Case Name	Generate the merit list

Test Data	Total aggregate Applicant's applied
Pre-Condition	Applicant have given the university test Admin has forwarded the applicants application to the coordinator for verification Coordinator has verified the applicants
Actions	System Response
Admin clicks generate merit list button	System generates the merit list System sends a notification to the applicants notification panel indicating that the applicant has been selected for the respective program in which their total aggregate was greater than the minimum required criteria.
Result	System displays the list of applicants qualified for this session.

Test verification: This test case ensures that the merit list of eligible students has been generated.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.7 *Manage Departments (Admin)*

Test Case ID	TC-1.38
Test Case Name	Add department success
Test Data	Department Name and Description
Pre-Condition	None
Actions	System Response
Admin clicks on the create department in department section on the side bar.	System shall redirect the admin to the department name and description form.
Result	Admin can enter the required fields and click on the submit button in order to create a department.

Test verification: This test case ensures that the admin can add departments to the system.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.39
Test Case Name	Add department failure
Test Data	Department Name and Description
Pre-Condition	Form fields are left empty
Actions	System Response
Admin clicks on the submit button.	System will notify the admin that they shall fill in all the required fields.
Result	Admin fills the input fields and then tries again.

Test verification: This test case ensures that the admin can complete all the input fields before submission of the form to create a department.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.40
Test Case Name	Update department
Test Data	Department Name and Description
Pre-Condition	Department must be existing in order to update
Actions	System Response
Admin clicks on the update department button.	System shall redirect the admin to the department name and description form with the previous details in order to update them.
Result	Admin can fill in the required fields and click on update in order to update the department details.

Test verification: This test case ensures that the admin can update existing departments.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.8 *Manage Programs (Admin)*

Test Case ID	TC-1.41
Test Case Name	Add undergraduate program success

Test Data	Form inputs
Pre-Condition	None
Actions	System Response
Admin clicks on the add undergraduate program button.	System shall redirect the admin to the form page where the admin can add an undergraduate program for students to apply in.
Result	Undergraduate programs will be added and shown during the application process to applicants if it is enabled.

Test verification: This test case ensures that the admin can add undergraduate programs to the system.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.42
Test Case Name	Add undergraduate program failure
Test Data	Form inputs
Pre-Condition	None
Actions	System Response
Admin leaves an input field empty	System shall notify the admin that the undergraduate program is not added due to a missing field.
Result	Admin shall enter all the required fields and try adding the undergraduate program again.

Test verification: This test case ensures that the admin can add undergraduate programs to the system.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.43
Test Case Name	Add graduate program success
Test Data	Form inputs
Pre-Condition	None

Actions	System Response
Admin clicks on the add graduate program button.	System shall redirect the admin to the form page where the admin can add a graduate program for students to apply in.
Result	Graduate programs will be added and shown during the application process to applicants if it is enabled.

Test verification: This test case ensures that the admin can add graduate programs to the system.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.44
Test Case Name	Add graduate program failure
Test Data	Form inputs
Pre-Condition	None
Actions	System Response
Admin leaves an input field empty	System shall notify the admin that the graduate program is not added due to a missing field.
Result	Admin shall enter all the required fields and try adding the graduate program again.

Test verification: This test case ensures that the admin can add graduate programs to the system.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.45
Test Case Name	Update program success
Test Data	Input fields
Pre-Condition	Program is already created
Actions	System Response
Admin clicks on the update button. Admin fills in the program update form. Admin clicks on the submit button.	System shall notify the admin that the program has been updated.

Result	Admin shall be able to view the updated program details.
---------------	--

Test verification: This test case ensures that the program details are updated.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.46
Test Case Name	Remove program
Test Data	Program name
Pre-Condition	Program exists
Actions	System Response
Admin clicks on remove button	System shall notify the admin that the program has been removed successfully. System updates the list of programs.
Result	Admin shall be able to view the updated list of programs.

Test verification: This test case ensures that the program can be removed.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.47
Test Case Name	Disable all programs
Test Data	Program status
Pre-Condition	More than one program is created
Actions	System Response
Admin clicks on disable all programs button	System shall notify the admin that all the programs are disabled. System shall display applicants with a message that there are no programs found. System shall change the button to enable the under action section of each program.
Result	Admin shall be able to enable them manually one by one.

Test verification: This test case ensures that all the program can be disabled at once.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.9 *Manage Coordinators (Admin)*

Test Case ID	TC-1.48
Test Case Name	Register Coordinator success
Test Data	Form input fields
Pre-Condition	Coordinator has not been registered yet
Actions	System Response
Admin clicks on add coordinator	System will redirect the admin to the form for registering a coordinator
Result	Admin will be able to register the coordinator after entering their required credentials.

Test verification: This test case ensures that the admin can register the coordinator.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.49
Test Case Name	Register Coordinator failure
Test Data	Form input fields
Pre-Condition	None
Actions	System Response
Admin clicks on add coordinator email in the form field	System will notify the admin that a coordinator with the same email exists in the same department.
Result	Admin will retry with a different email address.

Test verification: This test case ensures that the admin can not register the coordinator with the same email in the same department.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.50
Test Case Name	View coordinator profile
Test Data	Coordinator data
Pre-Condition	Coordinator has updated their details in the profile section.
Actions	System Response
Admin clicks on profile button	System will redirect the admin to the page containing the coordinator's profile details.
Result	Admin shall be able to view the coordinator's profile.

Test verification: This test case ensures that the admin can view the coordinators profile info.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.51
Test Case Name	Update coordinators details success
Test Data	Coordinator data
Pre-Condition	Coordinator has been created and assigned to a department
Actions	System Response
Admin clicks on update button	System will redirect the admin to the page containing coordinators details like (name, email and department) which can be updated by the admin.
Result	Admin shall be able to update the coordinators department and profile info.

Test verification: This test case ensures that the admin can update the coordinators details.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.52
Test Case Name	Remove the coordinator
Test Data	Coordinator email
Pre-Condition	Coordinator exists
Actions	System Response
Admin clicks on the remove coordinator button.	System will request confirmation of admin for the deletion of coordinator On receiving confirmation from admin, coordinator will be removed successfully
Result	Coordinator will be deleted permanently.

Test verification: This test case ensures that the admin can remove the coordinator.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.10 *Manage Applicants (Admin)*

Test Case ID	TC-1.53
Test Case Name	Display list of applicants
Test Data	Applicant's list
Pre-Condition	Applicant have applied for a program
Actions	System Response
Admin clicks manage applicant section	System displays the list of applicants that have applied for a program.
Result	Admin shall be able to view the applicant details and proceed with their application.

Test verification: This test case ensures that the admin proceeds processing the applicants admission process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.54
Test Case Name	Display applicant details
Test Data	Applicant details
Pre-Condition	Applicant have applied for a program
Actions	System Response
Adim clicks on view button next to admin name	System shall display the applicant details that they have entered earlier.
Result	Admin shall be able to proceed with forwarding application to the coordinator.

Test verification: This test case ensures that the admin proceeds processing the applicants admission process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.55
Test Case Name	Adding university test score
Test Data	Applicant university test score
Pre-Condition	Applicant gave the test Test results are calculated and available to admin
Actions	System Response
Admin enters the applicant's university test score. Admin clicks on add test score.	System indicates the admin with a message of test score added successfully
Result	Admin will proceed onto calculating the total aggregate of applicants.

Test verification: This test case ensures that the admin can enter the test score of the applicant.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.56
Test Case Name	Calculate total aggregate of the student

Test Data	Applicant university test score Applicant intermediate education score Applicant secondary education score Applicant nts test score
Pre-Condition	Applicant have applied for a program
Actions	System Response
Admin enters the applicant's university test score. Admin clicks on the calculate aggregate button	System will calculate the applicant's aggregate based on the previous result and university test score entered.
Result	Admin will proceed with forwarding the application depending on the applicant's info.

Test verification: This test case ensures that the admin proceeds onto calculating test score of applicant for processing the admission process.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.57
Test Case Name	Forward Application
Test Data	Applicant details Application forwarded status
Pre-Condition	Applicant total aggregate has been calculated
Actions	System Response
Admin clicks on the forward application button.	System fails to forward the applicant data.
Result	Admin fails to proceed with forwarding the application depending on the applicant's info.

Test verification: This test case fails to ensure the forwarding of the applicant's application to the coordinator.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.11 *Manage Payrolls (Admin)*

Test Case ID	TC-1.58
Test Case Name	Enable admission challan form
Test Data	Admission challan status
Pre-Condition	Merit list has been generated
Actions	System Response
Admin clicks on enable challan button	System generates a challan form with a random PV number and Bank number different for each applicant to apply for.
Result	Applicants can print challan form and submit it to the UBL bank.

Test verification: This test case ensures that the admission fee for the semester has been paid.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.59
Test Case Name	Display applicant submitted fee challan
Test Data	Admission challan status
Pre-Condition	Merit list has been generated
Actions	System Response
Admin clicks on enable challan button	System generates a challan form with a random PV number and Bank number different for each applicant to apply for.
Result	Applicants can print challan form and submit it to the UBL bank.

Test verification: This test case ensures that the admission fee for the semester has been paid.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.12 *Manage Applicants (Coordinator)*

Test Case ID	TC-1.60
Test Case Name	Display applicants details
Test Data	Admission challan status
Pre-Condition	Applicant has been forwarded by the admin
Actions	System Response
Coordinator clicks on the view button next to the applicant in the list.	System redirects the coordinator to the applicant detail section
Result	Coordinator can verify the applicant based on the application details

Test verification: This test case ensures that the coordinator can view the applicants details.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.61
Test Case Name	Verify applicant
Test Data	Applicant verified status
Pre-Condition	Applicant has been forwarded
Actions	System Response
Coordinator clicks on verify applicant button	System changes the applicant status to be verified as true.
Result	Applicant gets verified.

Test verification: This test case ensures that the applicant gets verified.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

Test Case ID	TC-1.62
Test Case Name	Register applicant as student
Test Data	Applicant form data

Pre-Condition	Applicant has paid the fee challan
Actions	System Response
Coordinator clicks on the register button. Coordinator enters the required fields for the registration process.	System registers the applicant as a student.
Result	Applicant is now a registered as a student

Test verification: This test case ensures that the applicant is registered as a student successfully.

Testing Environment: Google Chrome

Tested by: Raja Shaiyan Azad Khan

Date: 10th December, 2020

6.1.1.13 *Add Personal information with valid credentials*

Test Case ID	TC-1.63
Test Case Name	Add student's personal Information with valid credentials
Test Data	Name fatherNam e cnic semester dateOfBirth h phone_number er domicile registrationN o Nationality, Area City District
Pre-Condition	Admin login into system
Action s	System Response
Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list
Result	Student's information is added

Test verification: Test verifies that admin can add student's data into record

Testing Environment: Google Chrome
Tested by: Iqra Fareed
Date: 10th December, 2020

6.1.1.14 : *Add Personal information with invalid credentials*

Test Case ID	TC-1.64
Test Case Name	Add student personal Information with invalid credentials
Test Data	Name fatherNam e cnic semester dateOfBirth h phone_numb er domicile registrationN o Nationality, Area City District
Pre-Condition	Admin login into system
Action s	System Response
Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list
Result	Student's information is added

Test verification: Test verifies that admin can add student's data into record
Testing Environment: Google Chrome
Tested by: Iqra Fareed
Date: 10th December, 2020

6.1.1.15 : *Add Primary education information with valid credentials*

Test Case ID	TC-1.65
Test Case Name	Add primary education information with valid credentials

Test Case Data	School Olevels_matic Marks College feildOfStudy yearOfMariculation yearOfIntermediate overseasEducation						
Pre-Condition	Admin login into system						
	<table border="1"> <thead> <tr> <th>Actions</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>Enter required fields Enter submit button</td><td>All fields are added in students record System displayed the student in record list</td></tr> <tr> <td>Result</td><td>Student's information is added</td></tr> </tbody> </table>	Actions	System Response	Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list	Result	Student's information is added
Actions	System Response						
Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list						
Result	Student's information is added						

Test verification: Test verifies that admin can add student's data into record

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.16 : *Add Primary information with invalid credentials*

Test Case ID	TC-1.66						
Test Case Name	Add student Information						
Test Case Data	School Olevels_matic Marks College feildOfStudy yearOfMariculation yearOfIntermediate overseasEducation						
Pre-Condition	Admin login into system						
	<table border="1"> <thead> <tr> <th>Action s</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>Enter required fields Enter submit button</td><td>All fields are added in students record System displayed the student in record list</td></tr> <tr> <td>Result</td><td>Student's information is added</td></tr> </tbody> </table>	Action s	System Response	Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list	Result	Student's information is added
Action s	System Response						
Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list						
Result	Student's information is added						

Test verification: Test verifies that admin can add student's data into record

Testing Environment: Google Chrome

Tested by: Iqra Fareed
Date: 10th December, 2020

6.1.1.17 : Add tertiary education information with valid credentials

Test Case ID	TC-1.67
Test Case Name	Add student Information
Test Case Data	semester registration Gpa Cgpa Dep Courses section
Pre-Condition	Admin login into system
Action s	System Response
Enter required fields Enter submit button	All fields are added in students record System displayed the student in record list
Result	Student's information is added

Test verification: Test verifies that admin can add student's data into record

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.18 : Add tertiary education with invalid credentials

Test Case ID	TC-1.68
Test Case Name	Add information with invalid credentials
Test Data	semester registrationN o Gpa Cgpa Departme nt Courses section
Pre-Condition	Admin login into system

Action	System Response
Enter required fields with wrong data type Enter Submit button	Box outline will turn red and show an error msg
Result	Infromation will not be submitted

Test verification: Test verifies that admin can add only valid information in student's profile

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.19 : *Leave required input field*

Test Case ID	TC-1.69
Test Case Name	Leave required input field
Test Data	Name Program fatherNam e cnic semester dateOfBirth h phone_number er domicile registrationN o Nationality, Gpa Cgpa Departme nt Course s Area City District School Olevels_maticMar ks College feildOfStudy yearOfMariculatio

	in yearOfIntermediat e section
Pre-Condition	Admin login into system
Action s	System Response
Enter required fields Enter Submit button	Box outline will turn red and show an error msg
Result	Infromation will not be submitted

Test verification: Test verifies that admin enters all the required input fields

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.20 : *View student profile*

Test Case ID	TC-1.70
Test Case Name	View Student Profile
Test Data	NaN
Pre-Condition	Admin login into system
Action s	System Response
Enter view profile button	Particular profile will be opened
Result	Student's profile will be displayed

Test verification: Test verifies that admin view student's profile

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.21 : view all student's profile

Test Case ID	TC-1.71
Test Case Name	View All Student's Profile
Test Data	NaN
Pre-Condition	Admin login into system
Action s	System Response
Enter view all profile button	All profiles list will be opened
Result	All Student's profile list will be displayed

Test verification: Test verifies that admin view student's profile

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.22 : Edit profile

Test Case ID	TC-1.72
Test Case Name	Edit Profile
Test Data	Student's record that needs to be edited
Pre-Condition	Admin login into system Student's profile already created
Action s	System Response
Enter edit profile button Edit data that needs to be changed Enter Submit button	profile will be edited
Result	Student's profile will be edited

Test verification: Test verifies that admin can edit a student's profile

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.23 : *View Schedule*

Test Case ID	TC-1.73	
Test Case Name	View Schedule	
Test Data	View student's timetable	
Pre-Condition	Admin login into system Schedule already set	
Action s	System Response	
Enter schedule button	Schedule of all semesters will be displayed	
Result	Schedule displayed	

Test verification: Test verifies that admin can view students schedule

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.24 : *Add new timeslot in Schedule*

Test Case ID	TC-1.74	
Test Case Name	Add new timeslot	
Test Data	Admin will add new class schedule in the timetable	
Pre-Condition	Admin login into system	
Action s	System Response	
Enter schedule button Double tap on the schedule cell Form will displayed Fill the subject name , class name , time slot Click save	A new class schedule will be added in the time table	
Result	New class timetable will be displayed in schedule	

Test verification: Test verifies that admin can add new class in timetable

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.25 : *Edit new timeslot in Schedule*

Test Case ID	TC-1.75
Test Case Name	Edit timeslot
Test Data	Admin will edit class schedule in the timetable
Pre-Condition	Admin login into system
Action s	System Response
Enter schedule button Double tap on the particular schedule cell Form will displayed change the subject name , class name , time slot Click save	class schedule will be edited in the time table
Result	class timetable will be edited in schedule

Test verification: Test verifies that admin can edit class in timetable

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 10th December, 2020

6.1.1.26 : *Delete timeslot in Schedule*

Test Case ID	TC-1.76
Test Case Name	Delete timeslot
Test Data	Admin will delete class schedule in the timetable
Pre-Condition	Admin login into system
Action s	System Response
Enter schedule button Double tap on the particular schedule cell Form will displayed Click on delete icon Click save	class schedule will be deleted in the time table
Result	class timetable will be deleted in schedule

Test verification: Test verifies that admin can delete class in timetable
Testing Environment: Google Chrome
Tested by: Iqra Fareed
Date: 10th December, 2020

6.1.1.27 1.95: *Select Semester you want to enter your marks*

Test verification: Test verifies that faculty can select desired semester they want to enter marks
Testing Environment: Google Chrome

Test Case ID	TC-1.77
Test Case Name	Select Semester you want to enter your marks
Test Data	The faculty will select the desired semester
Pre-Condition	Faculty login into system
Action s	System Response
Login in system Select addmarks Form will displayed Click on desire semester icon Click save	Desire semester will be opened
Result	Desired semester will be opened

Tested by: Iqra Fareed
Date: 10th December, 2020

6.1.1.28 *Select Assessment type you want to enter your marks*

Test Case ID	TC-1.78
Test Case Name	Select Assessment you want to enter your marks
Test Data	The faculty will select the desired semester
Pre-Condition	Faculty login into system
Action s	System Response

Login in system Select addmarks Form will displayed Click on desire Assessment icon Click save	Desire Assessment type will be opened
Result	Desired assessment will be opened

Test verification: Test verifies that faculty can select desired assessment type they want to enter marks

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 19th December, 2020

6.1.1.29 *Select Course you want to enter your marks*

Test Case ID	TC-1.79
Test Case Name	Select course you want to enter your marks
Test Data	The faculty will select the desired semester
Pre-Condition	Faculty login into system
Action s	System Response
Login in system Select addmarks Form will displayed Click on desire course icon Click save	Desire course will be opened
Result	Desired course will be opened

Test verification: Test verifies that faculty can select desired course they want to enter marks **Testing Environment:** Google Chrome

Tested by: Iqra Fareed

Date: 19th December, 2020

6.1.1.30 *Add marks*

Test Case ID	TC-1.80
---------------------	---------

Test Case Name	Add marks
Test Data	The faculty will Add marks of their student
Pre-Condition	Faculty login into system
Action s	System Response
Login in system Select addmarks Form will displayed Click on desire Semester icon Click on desire assessment type icon Click on desire course icon Add marks Click save	Marks will be added in database
Result	Marks will be added in student's data

Test verification: Test verifies that faculty can add marks in student's data

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 19th December, 2020

6.1.1.31 *Delete marks*

Test Case ID	TC-1.81
Test Case Name	Delete marks
Test Data	The faculty will delete marks of their student
Pre-Condition	Faculty login into system
Action s	System Response
Login in system Select addmarks Form will displayed Click on desire Semester icon	Marks will be deleted in database

Click on desire assessment type icon Click on desire course icon Delete marks Click save	
Result	Marks will be deleted in student's data

Test verification: Test verifies that faculty can delete marks in student's data

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 19th December, 2020

6.1.1.32 *Edit marks*

Test Case ID	TC-1.82
Test Case Name	Edit marks
Test Data	The faculty will edit marks of their student
Pre-Condition	Faculty login into system
Action	System Response
Login in system Select addmarks Form will displayed Click on desire Semester icon Click on desire assessment type icon Click on desire course icon Edit marks Click save	Marks will be edit in database
Result	Marks will be edit in student's data

Test verification: Test verifies that faculty can edit marks in student's data

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 19th December, 2020

6.1.1.33 *Upload Files*

Test Case ID	TC-1.83	
Test Case Name	Upload file	
Test Data	The faculty will be able to upload csv file	
Pre-Condition	Faculty login into system	
Action s	System Response	
Login in system Select addmarks Form will displayed Click on upload file Select the file Click upload	Csv file will be uploaded	
Result	CSV file will be uploaded	

Test verification: Test verifies that faculty can upload a file

Testing Environment: Google Chrome

Tested by: Iqra Fareed

Date: 19th December, 2020

6.1.1.34 *Login with Valid Credentials*

Test Case ID	TC-1.84	
Test Case Name	Login with Valid Credentials	
Test Data	Email: example@gmail.com Password: noor123	
Pre-Condition	The faculty is registered	
Actions	System Response	
1. Entered the required fields. 2. Click on submit button.	1. The system logged in the user. 2. The system displayed the dashboard.	
Result	User is logged in the system.	

Test verification: This test verifies that faculty can login with right credentials

Testing Environment: Google Chrome

Tested by : Noor us Sahar
Date: 10th December, 2020

6.1.1.35 *Login with Invalid Credentials*

Test Case ID	TC-1.85
Test Case Name	Login with Invalid Credentials
Test Data	Invalid Credentials: Email : example1@gmail.com password:noor123 email : example@gmail.com password:123 Email: wrong@gmail.com password: 123
Pre-Condition	The faculty email is not registered
Actions	System Response
1. Entered the required fields. 2. Click on submit button.	1. The system generated an alert of invalid credentials
Result	User is not logged in the system

Test verification: This test verifies that faculty cannot login with invalid credentials i.e., email or password.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.36 *Logout*

Test Case ID	TC-1.86
Test Case Name	Logout
Test Data	N/a
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
Click the logout button in navbar	The system logs out the user and shows the login page
Result	User is not logged out of the system

Test verification: This test verifies that faculty can logout of the portal.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.37 *Create Profile with Valid Credentials*

Test Case ID	TC-1.87
Test Case Name	Create Profile with Valid Credentials
Test Data	Designation: Professor, Phone Number: 03122222222, Website : xyz.com Address : Pindi Bio : This is new bio
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1. Click on create profile button on dashboard 2. Fill the fields 3. Click on submit	1. The system generates a “Profile Created” message. 2. The system redirected to dashboard 3. The created profile is shown on the dashboard.
Result	Profile is created.

Test verification: This test verifies that faculty can create its profile.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.38 *Create Profile with Invalid Credentials*

Test Case ID	TC-1.88
Test Case Name	Create Profile with Invalid Credentials
Test Data	Designation: Professor, Phone Number: 122222222, Website : Address : Pindi Bio : This is new bio
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1. Click on create profile button on	1. The system generates a “Invalid Credentials”

dashboard 2. Fill the fields 3. Click on submit	message.
Result	Profile is not created.

Test verification: This test verifies that faculty cannot create its profile with Invalid Credentials

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.39 *Edit Profile*

Test Case ID	TC-1.89
Test Case Name	Edit Profile
Test Data	Bio: This is the new bio
Pre-Condition	The faculty is logged in the portal and has a profile
Actions	System Response
1.Click on edit Profile Button on dashboard. 2. Enter the fields you want to change. 3. Click on submit	1. System generates “Profile Updated” message. 2. System redirects to dashboard.
Result	Profile is updated

Test verification: This test verifies that faculty can update the profile.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.40 *View Profile*

Test Case ID	TC-1.90
Test Case Name	View Profile
Test Data	N/A
Pre-Condition	The faculty is logged in the portal and has a profile
Actions	System Response
1.Click on view Profile Button on dashboard.	1. System shows the profile.

Result	Profile is displayed
---------------	----------------------

Test verification: This test verifies that faculty can view the profile.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.41 *Add Research Paper*

Test Case ID	TC-1.91
Test Case Name	Add Research Paper
Test Data	Title: Research Date: 01/01/2020 Authors: Noor, Iqra Description: This is the research paper
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on add research paper button on dashboard or in edit profile page. 2. Input the fields. 3. Click on Submit	1. System generates “Research Paper Added” message. 2. System redirects to dashboard.
Result	Research Paper is added

Test verification: This test verifies that faculty can add a research paper to his/her profile.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.42 *Add Education*

Test Case ID	TC-1.92
Test Case Name	Add Education
Test Data	Institute: Comsats University Islamabad Degree: BSE Grade: A From Date: 03/02/2017

	To Date: 20/01/2021
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on add education button on dashboard or in edit profile page. 2. Input the fields. 3. Click on Submit	1. System generates “Education Added” message. 2. System redirects to dashboard.
Result	Education is added

Test verification: This test verifies that faculty can add education to his/her profile.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.43 *Add Experience*

Test Case ID	TC-1.93
Test Case Name	Add Experience
Test Data	Company: XYZ Designation: Developer From Date: 10/02/2019 To Date: 10/04/2020
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on add experience button on dashboard or in edit profile page. 2. Input the fields. 3. Click on Submit	1. System generates “Experience Added” message. 2. System redirects to dashboard.
Result	Experience is added

Test verification: This test verifies that faculty can add Experience to his/her profile.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.44 *View Calendar*

Test Case ID	TC-1.94
Test Case Name	View Calendar

Test Data	N/A
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on “View My Calendar” from dashboard or “Calendar” from sidebar.	1. System displays the calendar page.
Result	Calendar is displayed.

Test verification: This test verifies that faculty can view his/her calendar

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.45 *Add Event*

Test Case ID	TC-1.95
Test Case Name	Add Event
Test Data	Title: New Event Start Date and Time: 20/10/2020 11:59 PM End Date and Time : 22/10/2020 11:59PM
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on “Add Event” on Calendar Page. 2. Input the fields. 3. Click on submit	1. System adds the event. 2. System redirects the user to dashboard.
Result	Event is created.

Test verification: This test verifies that faculty can add an event on his/her calendar.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.46 *View All Profiles of Faculties*

Test Case ID	TC-1.96
Test Case Name	View All Profiles

Test Data	N/A
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on “View All Profiles” from dashboard or “Faculty Profiles” from sidebar.	1. System displays the Profiles page , with profiles of all faculty members.
Result	Profiles are displayed.

Test verification: This test verifies that faculty can view Profiles of all faculty member.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.47 *View Profile of Another Faculty Member*

Test Case ID	TC-1.97
Test Case Name	View Profile of Another Faculty Member
Test Data	N/A
Pre-Condition	The faculty is logged in the portal.
Actions	System Response
1.Click on “View Profile” button in front of the faculty member you want to view the profile of.	1. System displays the Profile of the Faculty Member
Result	Profile is displayed.

Test verification: This test verifies that faculty can view Profile of one faculty member.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.48 *View All Surveys*

Test Case ID	TC-1.98
---------------------	---------

Test Case Name	View All Surveys
Test Data	N/A
Pre-Condition	The user is logged in the portal.
Actions	
1.Click on “Surveys” button in the sidebar	1. System displays the list of all the surveys
Result	Surveys are displayed

Test verification: This test verifies that user can view list of all surveys.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.49 *View/ Fill Surveys*

Test Case ID	TC-1.99
Test Case Name	View/ Fill Surveys
Test Data	N/A
Pre-Condition	The user is logged in the portal.
Actions	
1.Click on “View” button in front of the survey to be viewed and filled. 2. Fill the survey 3. Click on complete .	1. System displays survey to be filled. 2. System generates “Response Saved Successfully” after completion of survey. 3. System redirects to the dashboard.
Result	Survey Response is saved and survey can be filled.

Test verification: This test verifies that user can view and fill the survey to save a response.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 10th December, 2020

6.1.1.50 *Fill Surveys leaving Required Fields Empty*

Test Case ID	TC-1.100
Test Case Name	View/ Fill Surveys

Test Data	N/A
Pre-Condition	The user is logged in the portal.
Actions	System Response
1.Click on “View” button in front of the survey to be viewed and filled. 2. Fill the survey leaving the fields empty 3. Click on complete.	1. System displays survey to be filled. 2. System prompts the user to input the required fields by highlighting them and showing a message “Please Answer the question”
Result	Survey Response cannot be saved.

Test verification: This test verifies that user has to fill all the required fields in order to complete the survey.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 14th December, 2020

6.1.1.51 *View Report*

Test Case ID	TC-1.101
Test Case Name	View Report
Test Data	N/A
Pre-Condition	The user is logged in the portal.
Actions	System Response
1.Click on “View Report” button in front of the survey.	1. System displays the reports.
Result	Survey report is displayed

Test verification: This test verifies that user can view the reports of the surveys.

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 14th December, 2020

6.1.1.52 *Complain Using chatbot*

Test Case ID	TC-1.102
Test Case Name	Complain using chatbot

Test Data	Name : Noor Email : noor@gmail.com Complain : This is a complain
Pre-Condition	The user is logged in the portal.
Actions	System Response
1.Click on Chatbot icon 2.Select complain from options 3. Enter the credentials 4.System asks for confirmation 5.Confirm sending email.	1. System displays a message “Mail sent” 2. Email sent to the admin.
Result	Email sent to the admin

Test verification: This test verifies that user can complain using the chatbot

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 14th December, 2020

6.1.1.53 *Suggestion Using chatbot*

Test Case ID	TC-1.103
Test Case Name	Suggestion using chatbot
Test Data	Name : Noor Email : noor@gmail.com Complain : This is a suggestion
Pre-Condition	The user is logged in the portal.
Actions	System Response
1.Click on Chatbot icon 2.Select suggestion from options 3. Enter the credentials 4.System asks for confirmation 5.Confirm sending email.	1. System displays a message “Mail sent” 2. Email sent to the admin.
Result	Email sent to the admin

Test verification: This test verifies that user can make a suggestion using the chatbot

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 14th December, 2020

6.1.1.54 *Find a Page Using chatbot*

Test Case ID	TC-1.104	
Test Case Name	Find a Page using chatbot	
Test Data	N/A	
Pre-Condition	The user is logged in the portal.	
Actions		System Response
1.Click on Chatbot icon 2. Click on find page from options. 3.System shows options of pages to select from.		1. System Redirects to the selected page.
Result	Redirected to selected page	

Test verification: This test verifies that user can redirect to the selected page

Testing Environment: Google Chrome

Tested by : Noor us Sahar

Date: 14th December, 2020

6.1.1.55 *Manage Courses (Coordinators)*

Test Case ID	TC-105	
Test Case Name	Add undergraduate courses success	
Test Data	Form inputs	
Pre-Condition	None	
Actions		System Response
Coordinators click on the add undergraduate courses button.		System shall redirect the coordinators to the form page where the coordinator can add an undergraduate courses for students to apply in.
Result	Undergraduate courses will be added and shown	

	during the application process to applicants if it is enabled.
--	--

Test verification: This test case ensures that the coordinators can add undergraduate courses to the system.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date : 19th Dec , 2020

Test Case ID	TC-106
Test Case Name	Add undergraduate courses failure
Test Data	Form inputs
Pre-Condition	None
Actions	System Response
Coordinator leaves an input field empty	System shall notify the coordinator that the undergraduate program is not added due to a missing field.
Result	Coordinator shall enter all the required fields and try adding the undergraduate courses again.

Test verification: This test case ensures that the coordinator can add undergraduate courses to the system.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-107
Test Case Name	Add graduate courses success
Test Data	Form inputs
Pre-Condition	None
Actions	System Response
Coordinator clicks on the add graduate courses button.	System shall redirect the coordinator to the form page where the coordinator can add a graduate courses for students to apply in.
Result	Graduate courses will be added and shown during the application process to applicants if it is enabled.

Test verification: This test case ensures that the coordinator can add graduate courses to the system.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-108
Test Case Name	Add graduate courses failure
Test Data	Form inputs
Pre-Condition	None
Actions	System Response
Coordinator leaves an input field empty	System shall notify the coordinator that the graduate course is not added due to a missing field.
Result	Coordinator shall enter all the required fields and try adding the graduate courses again.

Test verification: This test case ensures that the coordinator can add graduate courses to the system.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-109
Test Case Name	Update courses success
Test Data	Input fields
Pre-Condition	Course is already created
Actions	System Response
Coordinator clicks on the update button. Coordinator fills in the courses update form. coordinators clicks on the submit button.	System shall notify the coordinator that the course has been updated.
Result	coordinator shall be able to view the updated course details.

Test verification: This test case ensures that the course details are updated.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-110
Test Case Name	Remove course
Test Data	Course name
Pre-Condition	Course exists
Actions	System Response
coordinators click on remove button	System shall notify the coordinator that the course has been removed successfully. System updates the list of courses.
Result	Coordinator shall be able to view the updated list of courses.

Test verification: This test case ensures that the course can be removed.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-111
Test Case Name	Disable all courses
Test Data	Courses status
Pre-Condition	More than one course is created
Actions	System Response
coordinator clicks on disable all courses button	System shall notify the coordinator that all the courses are disabled. System shall display applicants with a message that there are no courses found. System shall change the button to enable the under action section of each course.
Result	Coordinator shall be able to enable them manually one by one.

Test verification: This test case ensures that all the courses can be disabled at once.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

6.1.1.56 *Manage Sections (Coordinators)*

Test Case ID	TC-112
Test Case Name	Add sections success
Test Data	Section Name and Description
Pre-Condition	None
Actions	System Response
Coordinators clicks on the create section on the side bar.	System shall redirect the coordinators to the section name and description form.
Result	Coordinator can enter the required fields and click on the submit button in order to create section.

Test verification: This test case ensures that the coordinators can add sections to the system.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-113
Test Case Name	Add section failure
Test Data	Section Name and Description
Pre-Condition	Form fields are left empty
Actions	System Response
coordinator clicks on the submit button.	System will notify the coordinator that they shall fill in all the required fields.
Result	Coordinator fills the input fields and then tries again.

Test verification: This test case ensures that the coordinator can complete all the input fields before submission of the form to create a section.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-114
Test Case Name	Update Section
Test Data	Section Name and Description

Pre-Condition	Section must be existing in order to update
Actions	System Response
Coordinators clicks on the update section button.	System shall redirect the coordinator to the section name and description form with the previous details in order to update them.

Test verification: This test case ensures that the coordinator can update existing sections.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

6.1.1.57 *Manage Semester (Coordinator)*

Test Case ID	TC-115
Test Case Name	Check for semester enrollment
Test Data	Semester Enrollment enabled
Pre-Condition	Coordinators must have enabled the semester sessions.
Actions	System Response
Coordinators will click on apply now option.	System shall redirect the coordinator to the semester page.
Result	Coordinators shall be able to manage semester enrollments.

Test verification: This test case verifies that the coordinator can manage semesters.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-116
Test Case Name	Create semester enrollment session
Test Data	Session name
Pre-Condition	Semester Enrollment Session has not been created before
Actions	System Response

Coordinator shall click on the create enrollment semester session button on dashboard.	System shall redirect the coordinator to a page. Coordinator shall enter the enable semester enrollment semester session name, start date and end date.
Result	System will notify the coordinator the system enrollment session has been created successfully.

Test verification: This test case verifies the semester enrollment session is created successfully.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-117
Test Case Name	Create semester enrollment session failed
Test Data	Session name
Pre-Condition	Session has been created before
Actions	System Response
Coordinators enters the required fields and clicks on the create button.	System shall notify the coordinators that the enrollment semester session has already been created.
Result	Coordinator shall remove the current semester enrollment session and try again.

Test verification: This test case ensures that there are only 2 semester enrollment session per year.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

Test Case ID	TC-118
Test Case Name	Update semester enrollment session
Test Data	Input fields
Pre-Condition	Session has been already created
Actions	System Response
Coordinator clicks on the update enrollment session button.	System shall notify the coordinator that the semester enrollment session has been updated.

Coordinator fills in the semester enrollment session update form. Coordinator clicks on the semester enrollment update button.	
Result	Coordinator shall be able to view the updated semester enrollment session details.

Test verification: This test case ensures that the semester enrollment session details are updated.

Testing Environment: Curriculum Management and Administration System

Tested by: Raja Shaiyan Azad Khan

Date: 19th Dec , 2020

6.1.2 Integration Testing

No.	Test Case/Test Script	Test Data	Expected Result	Actual Result	Pass/Fail/Not Executed/ Suspended
1.	Manipulate Applicant's Aggregate	Applicant's Test Score	System will take the applicant's test score, added by the Admin and calculate the total aggregate.	As expected	Pass
2.	Manipulate Applicant's status	Applicant Type	System will confirm applicant's registration as a student once registered by coordinator.	As expected.	Pass
3.	Display List of Faculty Profiles	Faculty Accounts	System will fetch all the faculty accounts registered and display their profiles at the Admin side.	As expected	Pass
4	Check the link in between Login and Respective User Portal	Login Credentials	System will check the authorization credentials of the user being logged in and redirect them to their respective portals.	As expected	Pass

5.	Assign Courses to Registered Students	Student's Registered Course List	System will append the courses to the registered course list of Applicants. Which can be seen on the students side of the data.	As expected	Pass
----	---------------------------------------	----------------------------------	---	-------------	------

7 Conclusion and Future Work

7.1 Conclusion

Our system will fulfill the requirements of admin, coordinator, and faculty and provide them a platform where they can perform their tasks with ease. The system provides three different portals where admin, coordinator, and faculty can perform different tasks with different access criteria where they can access and modify required data with privacy and integrity

7.2 Future Work

In the near future, we are looking to enhance the overall functionality of the system by adding more options in the functionalities. Where we can offer auto-generated data and involve more modules that can integrate external APIs. We are also looking to include more categories i.e. banking modules where the student can submit their fee online. Aside from this many UI improvements are needed which will be added in the future. We are looking forward to automating our schedule system so we can get an auto-generated timetable.

8 References

1. Lyda M.S. Lau, Jayne Curson, Richard Drew, Peter Dew and Christine Leigh, (1999), Use Of VSP Resource Rooms to Support Group Work in a Learning Environment, ACM 99, pp- 2. **(Journal paper example)**
2. Hideyuki Nakanishi, Chikara Yoshida, Toshikazu Nishimura and TuruIshada, (1996), FreeWalk: Supporting Casual Meetings in a Network, pp 308-314 **(paper on web)**
<http://www.acm.org/pubs/articles/proceedings/cscw/240080/p308-nakanishi.pdf>

3. Ali Behforooz & Frederick J. Hudson, (1996), Software Engineering Fundamentals, Oxford University Press. Chapter 8, pp255-235. (**book reference example**)
4. Page Author, Page Title, <http://www.bt.com/bttj/archive.htm>, Last date accessed. (**web site**)