INTEGRATED UNIVERSITY ADMISSION PLATFORM

Submitted by: Rimsha Fiayaz

Supervised by: Ms Ifrah Farrukh Khan

Department of Computer Science

Quaid-i-Azam University Islamabad

Session (2014-2018)

Acknowledgement

First of all, I would like to express my sincere and humble gratitude to Almighty ALLAH whose blessings and guidance has been a real source of all achievements in my life. I am thankful to my parents and my family for their prayers and all kinds of support during my educational career. They were really there to help me whatever they could do. I really appreciate their caring attitude and guidance.

I pay homage to my supervisor Ms. Ifrah Farrukh Khan. This work would not have been possible without her guidance, support and encouragement. Under her guidance I successfully overcame many difficulties and learned a lot.

Rimsha Fiayaz.

Abstract

Integrated University Admission Platform is a web-based tool that facilitates students and institutions in admission process. The students can make their personal and educational profile and apply in any institution. The system also facilitates institutions as it makes the admission process easy by automatic generation of merit lists. The system provides a single platform for all the institutions which helps admission seeking students. IUAP is advantageous to manual admission system which is a time taking process and involves huge manpower whereas the online admission system ensures accurate and very fast computerized information.

Table of Contents

List of Figures

List of Tables

Definitions, Acronyms, and Abbreviations

IUAP: Integrated University Admission Platform

Php: hypertext preprocessor

GUI: Graphical User Interface

WWW: World Wide Web

HTTP: Hyper Text Transfer Protocol

SPMP: Software Project Management Plan

SRS: Software Requirement Specification

UC: Use case

UCD: Use Case Description

SDD: Software Design Description

STD: Software Test Documentation

AD: Activity Diagram

Chapter 1
Software Project Management Plan

1.1: Introduction

This chapter provides the description of software approach and milestones associated with each step. It also covers what tools and techniques are used to develop software. It also provides description of which process model will be used to develop the system.

1.1.1: Product Overview

The goal of IUAP is to deliver product that allows candidates/students to apply in their favorite institutions as well as institutions easy access to student's data and sending notifications to admitted students for roll no slips and entry test dates and to suggest universities to students who have not been yet admitted in any institution.

1.1.2: Project Deliverables

The project deliverables incudes:

- Software project management plan (SPMP).
- Software requirement specification(SRS).
- Software design description(SDD).
- Implementation.
- Software test documentation(STD).

1.2: Project Organization

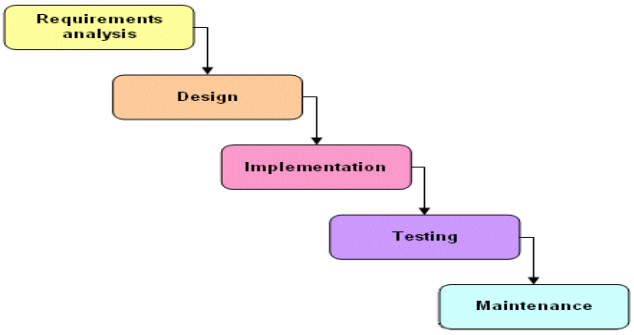
This section contains software process model which will used throughout this project, roles and responsibilities of stakeholders and tools and techniques to develop the product.

1.2.1: Software Process Model

In this project waterfall model will be used because:

- Requirements are clear.
- Product definition is stable.
- No ambiguous requirements.
- Very simple to implement.

In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases. Phases are processed and completed one at a time.



1.2.2: Roles and Responsibilities

The author is doing this project alone so all responsibilities are on her regarding this project.

1.2.3: Tools and Techniques

These tools used for developing the product:

- Notepad++.
- wamp server.
- Web Browser.
- Sql database.
- Microsoft visio for diagrams.
- Project libre for poject plan.

1.3: Project Management Plan

This section describes how the project will be manage, what are its tasks, deliverables, milestones etc.

1.3.1: Tasks

The project has the following tasks:

1.3.1.1: Problem Understanding

- **Description**
  First problem definition is must.
- **Resources Needed**
  Rimsha Fiayaz

- **Dependencies and Constraints**
  None

- **Risks and Contingencies**
  None

1.3.1.2: Analysis and Requirement

- **Description**
  In this task analysis would be done and SRS will be included.
- **Deliverables**
  SRS
- **Resources Needed**
  Rimsha Fiayaz
- **Dependencies and Constraints**
  Software Project Management Plan
- **Risks and Contingencies**
  None

1.3.1.3: Develop System Design

- **Description**
  In this task of software will meet the requirements. Also describes the rationale of design decisions taken will be included.
- **Deliverable**
  DSD
- **Resources Needed**
  Rimsha Fiayaz
- **Dependencies and Constraints**
  Analysis and Requirement
- **Risks and Contingencies**
  None

1.3.1.4: Software Test Documentation

- **Description**
  In this task plan and specifications to verify and validate the software and the results will be included.
- **Deliverables**
  STD
- **Resources Needed**
  Rimsha Fiayaz
- **Dependencies and Constraints**
  Develop System Design
- **Risks and Contingencies**
  None

1.3.1.5: Software Implementation

- **Description**
  In this we develop the plan how the project will be implemented.

- **Deliverables**
  STD
- **Resources Needed**
  Rimsha Fiayaz
- **Dependencies and Constraints**
  Develop System Design
- **Risks and Contingencies**
  None

1.3.2: Assignments

The project is on individual basis so all the assignments are on the author's end.

1.3.3: Timetable

This section describes the time table of project as how the project tasks would be accomplished.

Documentation schedule and time table

It describes the time table and plan of documentation.

		Name	Duration	Start	Finish
1		**Analysis and Requirement**	**40 days?**	**9/20/17 8:00 AM**	**10/29/17 5:00 PM**
2		**Identify Requirements**	**40 days?**	**9/20/17 8:00 AM**	**10/29/17 5:00 PM**
3		Problem Definition	3 days?	9/20/17 8:00 AM	9/22/17 5:00 PM
4		Review CaseStudy	2 days?	9/23/17 8:00 AM	9/24/17 5:00 PM
5		Define Requirement	3 days?	9/25/17 8:00 AM	9/27/17 5:00 PM
6		Meet Stakeholders	1 day?	9/28/17 8:00 AM	9/28/17 5:00 PM
7		**Define Usecases**	**31 days?**	**9/29/17 8:00 AM**	**10/29/17 5:00 PM**
8		Write Usecases	4 days?	9/29/17 8:00 AM	10/2/17 5:00 PM
9		Draw Usecase Diagram	3 days?	10/3/17 8:00 AM	10/5/17 5:00 PM
10		Review Requirement and Usecases	2 days?	10/6/17 8:00 AM	10/7/17 5:00 PM
11		**Develop SRS**	**20 days?**	**10/8/17 8:00 AM**	**10/27/17 5:00 PM**
12		Define Functional and Nonfunctional Requirements	4 days?	10/8/17 8:00 AM	10/11/17 5:00 PM
13		Write Use case Description	4 days?	10/12/17 8:00 AM	10/15/17 5:00 PM
14		Define Software System Attributes	2 days?	10/14/17 8:00 AM	10/15/17 5:00 PM
15		**Database Requirements**	**13 days?**	**10/8/17 8:00 AM**	**10/20/17 5:00 PM**
16		Define Entities	1 day?	10/16/17 8:00 AM	10/16/17 5:00 PM
17		Identify Relationships	3 days?	10/8/17 8:00 AM	10/10/17 5:00 PM
18		Develop ERD	4 days?	10/17/17 8:00 AM	10/20/17 5:00 PM
19		Review Requirements	3 days?	10/21/17 8:00 AM	10/23/17 5:00 PM
20		finalize SRS	2 days?	10/24/17 8:00 AM	10/25/17 5:00 PM
21		**Review SRS**	**1 day?**	**10/27/17 8:00 AM**	**10/27/17 5:00 PM**
22		Meet Stakeholders	1 day?	10/27/17 8:00 AM	10/27/17 5:00 PM
23		First deliverable	2 days?	10/28/17 8:00 AM	10/29/17 5:00 PM
24		**Develop System Design**	**59 days?**	**10/30/17 8:00 AM**	**12/27/17 5:00 PM**
25		**System Architectural Design**	**59 days?**	**10/30/17 8:00 AM**	**12/27/17 5:00 PM**
26		Develop Architectural Design	5 days?	10/30/17 8:00 AM	11/3/17 5:00 PM
27		Review Architectural Design	2 days?	11/4/17 8:00 AM	11/5/17 5:00 PM
28		**Data Design**	**52 days?**	**11/6/17 8:00 AM**	**12/27/17 5:00 PM**
29		Define Database Architecture	4 days?	11/6/17 8:00 AM	11/9/17 5:00 PM
30		Normalize ERD	3 days?	11/10/17 8:00 AM	11/12/17 5:00 PM
31		**Detail Design**	**28 days?**	**11/13/17 8:00 AM**	**12/10/17 5:00 PM**
32		Create Sequence Diagram	5 days?	11/13/17 8:00 AM	11/17/17 5:00 PM
33		Create Class Diagram	4 days?	11/18/17 8:00 AM	11/21/17 5:00 PM
34		**Interface Design**	**20 days?**	**11/21/17 8:00 AM**	**12/10/17 5:00 PM**
35		Develop Interface Design	4 days?	11/21/17 8:00 AM	11/24/17 5:00 PM

fyp - page1

		Name	Duration	Start	Finish
36		Review Interface design	2 days?	11/25/17 8:00 AM	11/26/17 5:00 PM
37		**Evaluate Design**	**12 days?**	**11/27/17 8:00 AM**	**12/8/17 5:00 PM**
38		Validate Design	5 days?	11/27/17 8:00 AM	12/1/17 5:00 PM
39		Verify Design	5 days?	12/2/17 8:00 AM	12/6/17 5:00 PM
40		Review and Refine Design	2 days?	12/7/17 8:00 AM	12/8/17 5:00 PM
41		Design Phase Completed	2 days?	12/9/17 8:00 AM	12/10/17 5:00 PM
42		2nd Deliverable	1 day?	12/27/17 8:00 AM	12/27/17 5:00 PM
43		**Software Test Documentation**	**16 days?**	**12/11/17 8:00 AM**	**12/26/17 5:00 PM**
44		System Overview	2 days?	12/11/17 8:00 AM	12/12/17 5:00 PM
45		Test Plan	4 days?	12/13/17 8:00 AM	12/16/17 5:00 PM
46		Test Cases	7 days?	12/17/17 8:00 AM	12/23/17 5:00 PM
47		Review Test Documentation	3 days?	12/24/17 8:00 AM	12/26/17 5:00 PM

Figure 1.1 DOCUMENTATION TIMETABLE

Gantt chart

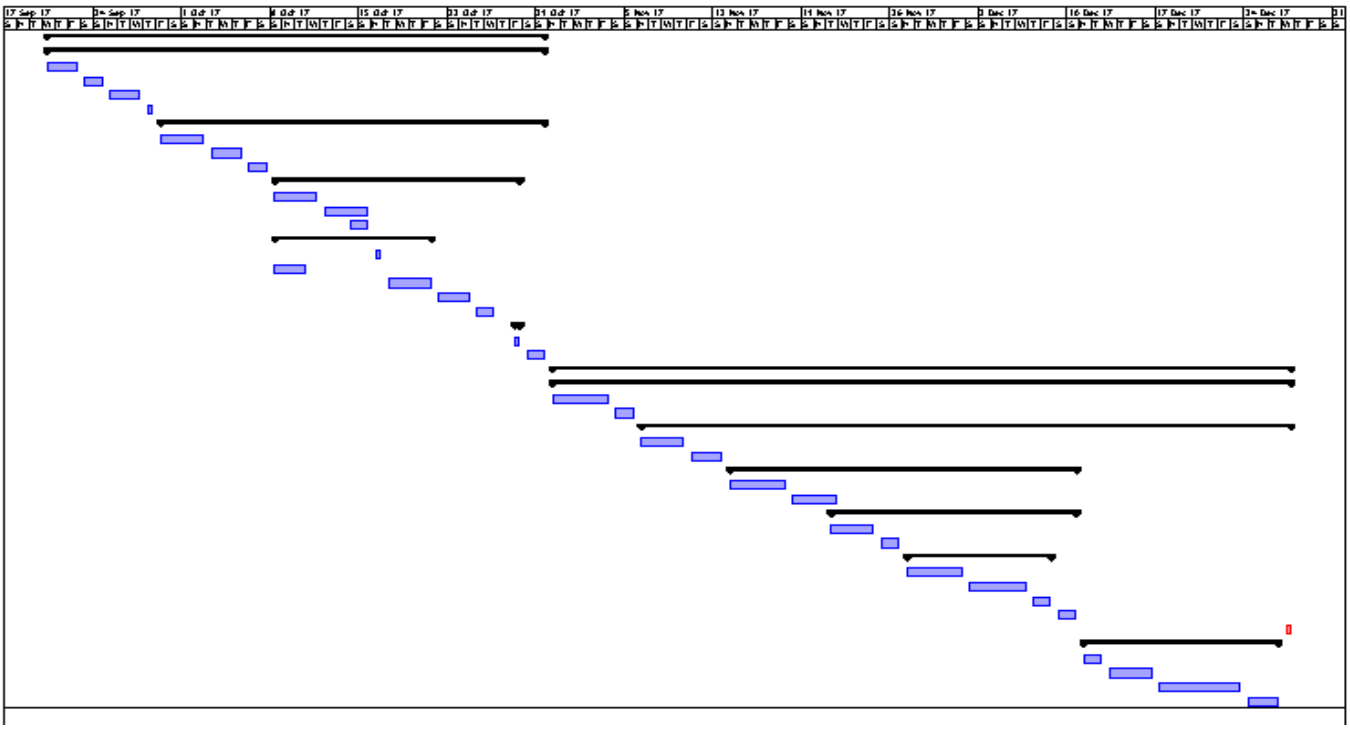


FIGURE 1.2 DOCUMENTATION GANTT CHART

Documentation schedule and time table

It describes the time table and plan of implementation.

		Name	Duration	Start	Finish
1		**Student panel**	**100 days?**	**2/1/18 8:00 AM**	**6/20/18 5:00 PM**
2		account creation module	10 days?	2/1/18 8:00 AM	2/14/18 5:00 PM
3		profile module	7 days	2/15/18 8:00 AM	2/23/18 5:00 PM
4		search programe module	9 days	2/24/18 8:00 AM	3/8/18 5:00 PM
5		search uni module	9 days	3/9/18 8:00 AM	3/21/18 5:00 PM
6		apply for programe module	15 days	3/22/18 8:00 AM	4/11/18 5:00 PM
7		**university admin pane**	**50 days?**	**4/12/18 8:00 AM**	**6/20/18 5:00 PM**
8		account creation module	10 days?	4/12/18 8:00 AM	4/25/18 5:00 PM
9		profile module	7 days	4/26/18 8:00 AM	5/4/18 5:00 PM
10		send notification module	18 days	5/6/18 8:00 AM	5/30/18 5:00 PM
11		**web admin panel**	**16 days**	**5/30/18 8:00 AM**	**6/20/18 5:00 PM**
12		account creation mo...	5 days	5/30/18 8:00 AM	6/5/18 5:00 PM
13		profile module	3 days	6/6/18 8:00 AM	6/8/18 5:00 PM
14		suggest uni module	5 days	6/8/18 8:00 AM	6/14/18 5:00 PM
15		implementation docu...	5 days	6/14/18 8:00 AM	6/20/18 5:00 PM
16		integration	5 days	2/1/18 8:00 AM	2/7/18 5:00 PM

FIGURE 1.3 IMPLEMENTATION TIMETABLE

Gantt chart:

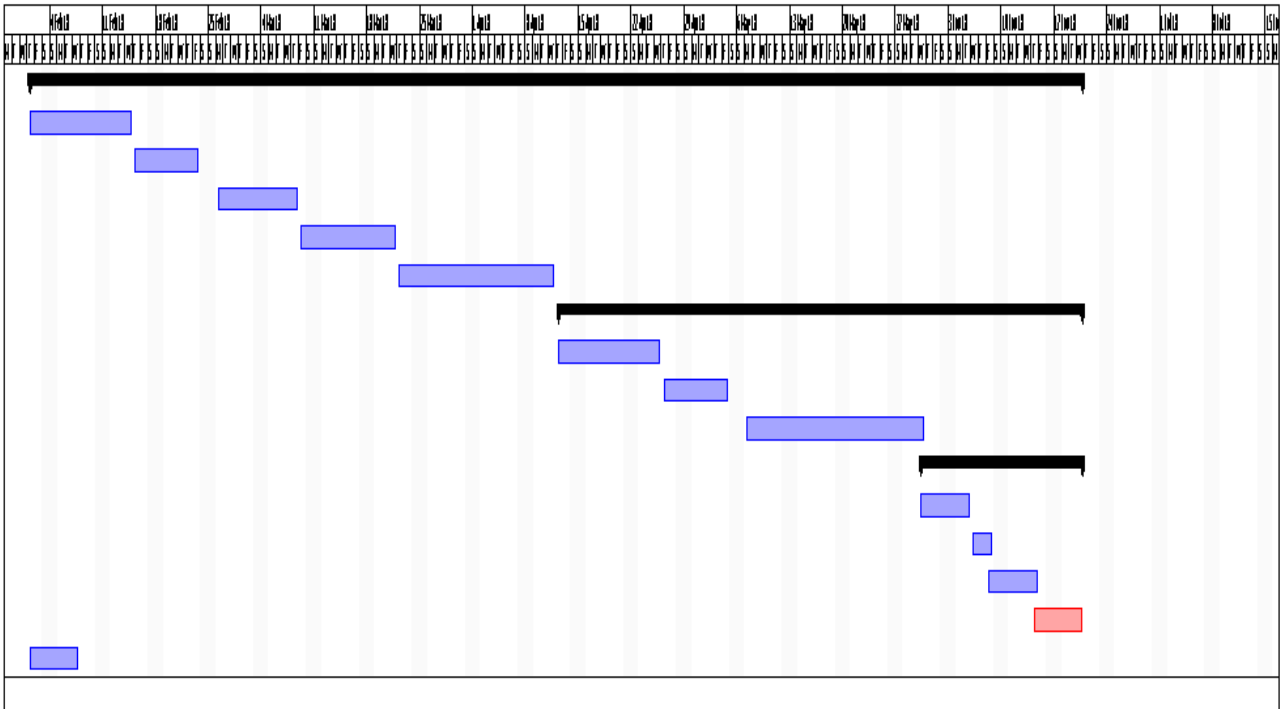


FIGURE 1.4 IMPLEMENTATION GANTT CHART

Chapter 2
SOFTWARE REQUIREMENT SPECIFICATION

2.1: Introduction

The integrated university admission platform (IUAP) is a web-based tool whose goal is to automate the Academic Institute's admission structure and its related operation and functionality. The objective of the initiative is to provide support to the admission seeking candidates by providing a faster, transparent and easy way of keeping records and use them for reference and further proceedings. Universities and educational institutions are also at a major advantage when it comes to an online admission process. IUAP is advantageous to manual admission system which is a time taking process and involves huge manpower whereas the online admission system ensures accurate and very fast computerized information with easy backup maintenance

2.1.1: System overview

The IUAP (integrated university admission platform) provides an interface for students to seek knowledge about their favorite programs in different universities. The student interface will allow students to make their personal and educational profile s well as apply in different educational institutions.

2.1.2: Purpose

The purpose of this document is to present a detailed description of IUAP (integrated university admission platform). It will explain the purpose and features of the software, the interfaces of the software, what the software will do and the constraints under which it must operate. It will illustrate the purpose and complete declaration for the development of system

2.1.3: Scope

- Will let registered students to apply for admission in different educational institutions
- Allow registered institutions to specify degrees and courses it is offering and the eligibility requirements for each program.
- Allow Student to make personal and academic profile and upload personal and educational information.
- Allow student to search for favorite degree program and apply for that program.
- Allow admin to register their program and upload all the information about that program.

2.1.4: STAKE HOLDERS

Following are the stake holders of IUAP:

- Students.
- Web admin.
- Super admin

2.2: Specific requirements

This section describes all requirements of the system. It gives a detailed description of the system and all its features.

2.2.1: External Interface Requirements

External interface requirements specify hardware, software, or database elements with which a system or component must interface. This section covers the details of external interface requirements.

2.2.1.1: User Interfaces

IUAP has a friendly user interface, user guide (user documentation) must be sufficient to educate the users on how to use this product without any problems or difficulties. IUAP should be designed for ease of use, providing appropriate error messages for user inputs. The background will be light in color with dark colored font to enhance the contrast and visibility.

2.2.1.2: Hardware Interfaces

Keyboard and mouse is used to input and monitor is to display output.

2.2.1.3: Software Interfaces

The IUAP will use windows operating system, xamp server, SQL database and a web browser for display.

2.2.1.4: Communication Interfaces:

As the system is internet based, therefore it will require some standard networking protocols for communication. These protocols are usually installed automatically by the operating system running on server.

- HTTP: It is a protocol used by the WWW service to make communication possible between a web server and a web browser.
- TCP/IP: It is a protocol used to communicate data all around the web.

2.2.2: Software Product Features

IUAP has many features. Each user panel has its own features. Some features are provided for only particular type of user and only these users have access to these features.

IUAP has the following features:

- **Web admin panel**

Web admin has the following features:

- Register account
- Login
- Logout
- Make program profile
- View program profile
- Update program profile
- View applicants of a program
- View merit list
- Change password

- **Student panel**

Student panel has the following features:

- Login.
- Logout.
- Register account.
- Make profile.
- View profile.
- Update profile.
- search program.
- Apply for program.
- View notifications.

- **Super Admin Panel**
  - Make program live
  - Approve web admin

2.3: Use case diagram

Use case diagram describes the actors, use cases and their relationship among them. It represents the user interaction and visualization of use cases. It is a graphical depiction of the interaction among the elements of system.

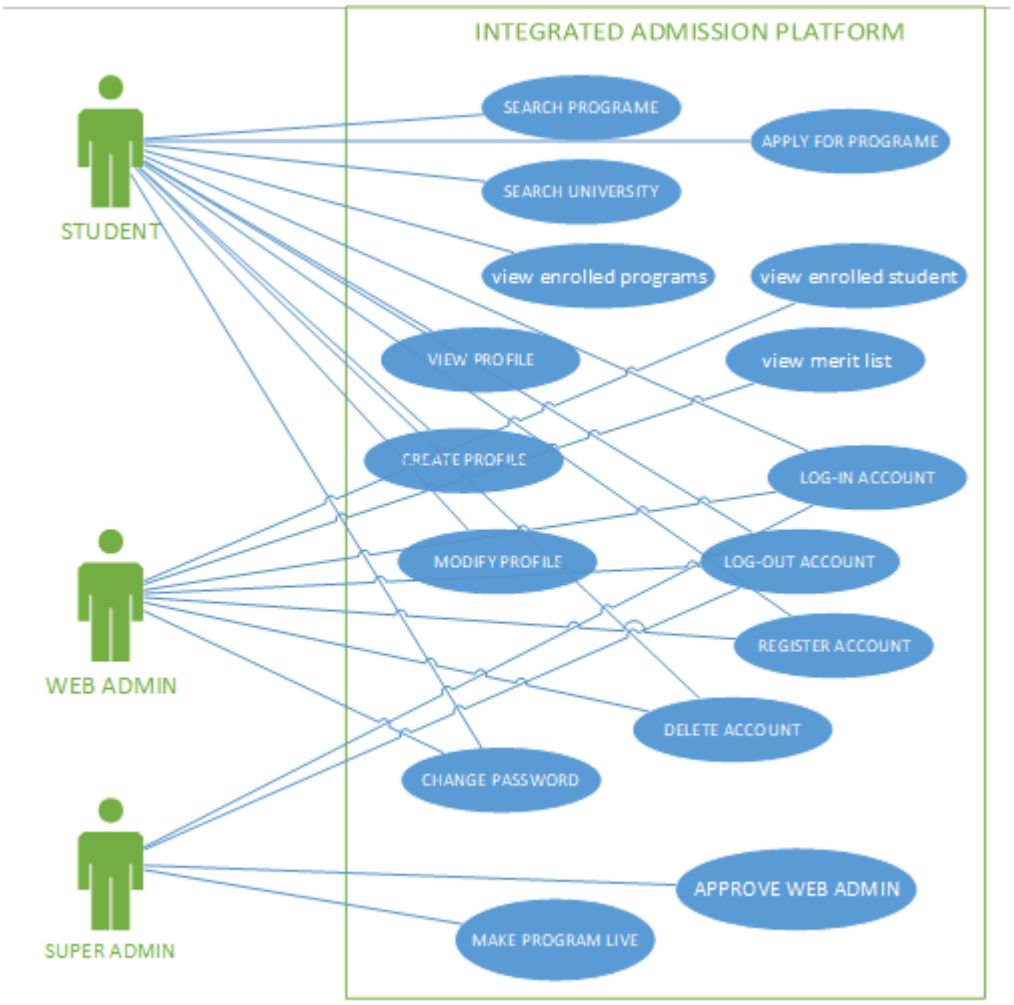


FIGURE 2.1 USE CASE DIAGRAM

2.3.1: USE CASES

Use cases can be employed during several stages of software development, such as planning system requirements, validating design, testing software, and creating an outline for online help and user manuals.

There are some use cases of this website which helps in developing it. The full description of use cases are given below:

UC-1: Register Account

This is the scenario of register account where user can create account. For this user first have to press create account button. After it user fills the required information of user and submits the

form. And if server and internet work properly then the login credentials would be sent to user's email address. And system displays the message that account created successfully and login credentials were sent to user's email address.

TABLE 2.1 UCD FOR REGISTER ACCOUNT

UC-1: Register Account	
Primary actor	User (web admin, student)
Goal in context	User will be able to create new user's account.
Pre-condition	User must be logged in.
Post-condition	New user's account has been created successfully.
Main Success Scenario	1) User presses the create account button. 2) User inputs account details. 3) User presses "Insert" button. 4) Username and password would be sent to given email address. 5) System displays account's registered message.
Alternative Scenario	4a) User submits information without filling all required fields. a) System prompts user to fill all required fields. 4b) Server is down. 5a) Internet is not available
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-2: Log-in account

This is the scenario of login where user can login in to account. User enters the login credentials such as username and password. After it user press login button and if the credentials are credentials are correct then home screen will appear. Every user has its own home screen/profile

TABLE 2.2 UCD FOR LOG-IN ACCOUNT

UC-2: Log-in account	
Primary actor	User (student, web admin)
Goal in context	User will be able to login account.
Pre-condition	User must have an account.
Post-condition	User has logged in successfully.
Main Success Scenario	1) User enters login credentials. 2) User press login button. 3) System display home screen.
Alternative Scenario	1a) User enters wrong credentials. a) System prompts user to enter correct username. 2a) User submits information without filling all required fields. a) System prompts user to fill all required fields. 2b) Server is down.
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-3: Log-out account

This is the scenario of logout where user can logout from his/her account. User press logout button and if the session expires successfully then login screen will appear. And if the sever is down then the user will not able to logout from his/her account until server works properly. In order to login again user must have to enter his/her credentials. And if these credentials are correct then user will be able to login again.

TABLE 2.3 UCD FOR LOG-OUT ACCOUNT

UC-3: Log-out account	
Primary actor	User (student, web admin)
Goal in context	User will be able to logout account.
Pre-condition	User must be logged in.
Post-condition	User has been logged out successfully.
Main Success Scenario	1) User presses “Logout” button. 2) System displays login screen.
Alternative Scenario	1a) Server is down.
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-4: Create profile

This is the scenario of create profile where user can make their profile in order to apply for different programs. For this user first have to login in to his/her account then press create profile button. After it user fills the required information and submits the form. And if server and internet work properly then the login credentials would be sent to user's email address. And system displays the message that profile created successfully and login credentials were sent to user's email address.

TABLE 2.4 UCD FOR CREATE PROFILE

UC-4: Create Profile	
Primary actor	User (student, web admin)
Goal in context	User will be able to create personal profile
Pre-condition	User must be logged in.
Post-condition	New user's profile has been created successfully.
Main Success Scenario	1- User presses the create profile button 2- User inputs account details. 3- User presses "insert" button. 4- Username and password would be sent to given email address 5- System displays profile registered message.
Alternative Scenario	4a) User submits information without filling all required fields. a) System prompts user to fill all required fields. 4b) Server is down. 5a) Internet is not available
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-5: Modify profile information

This is the scenario of update account where user can update his/her profile. For this student first have to login in to his/her account then press edit button. After it student modifies the information and submits the form. And if server works properly then the modified information is saved in the database. And system displays the message that account updated successfully.

TABLE 2.5 UCD FOR MODIFY PROFILE INFORMATION

UC-5: Modify profile information	
Primary actor	User (student, web admin)
Goal in context	User will be able to update account.
Pre-condition	User must be logged in.
Post-condition	User updated the account successfully.
Main Success Scenario	1) User presses the edit button. 2) User modifies the account information. 3) User presses “Update” button. 4) System displays profile updated message
Alternative Scenario	4a) User saves information without filling all required fields. a) System prompts user to fill all required fields. 4b) Server is down.
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-6: View Profile

This is the scenario of view profile where user can view his/her profile information. For this user first have to login in to his/her account, and click on view profile information, if server works properly user profile will be displayed

TABLE 2.6 UCD FOR VIEW PROFILE

UC-6: View Profile	
Primary actor	User (student, web admin)
Goal in context	User will be able to view profile details.
Pre-condition	User must be logged in.
Post-condition	User has viewed profile successfully.
Main Success Scenario	1) User presses the view button. 2) System displays profile details
Alternative Scenario	.1a) Server is down.
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-7: Apply for Program

This is the scenario of apply program where user can apply for his/her favorite program. For this user first have to login in to his/her account then search for his/her favorite program and then click on apply program, If server works properly then student application for program is submitted successfully and system displays the message that user applied successfully.

TABLE 2.7 UCD FOR APPLY FOR PROGRAME

UC-7: Apply for Program	
Primary actor	User (student)
Goal in context	User will be able to apply for specific program
Pre-condition	User must be logged in.
Post-condition	User applied for the specific program successfully.
Main Success Scenario	1) Student presses the apply button. 2) System prompts user for confirmation. 3) Student confirms the operation. 4) System displays that user has applied successfully
Alternative Scenario	3a) User not confirms the operation. 4a) Server is down.
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UC-8: Change Password

This is the scenario of update password where user can update his/her password. For this user first have to login in to his/her account then press change password button. After it user enter his/her old and new password and submits the form. And if the information is correct and server works properly then the modified information is saved in the database

TABLE 2.8 UCD FOR CHANGE PASSWORD

UC-8: Change Password	
Primary actor	User (student)
Goal in context	User will be able to update the password.
Pre-condition	User must be logged in.
Post-condition	User updated the password successfully.
Main Success Scenario	1- User presses the change password option. 2- User enter old password. 3- User enter new password 4- User presses “Update” button. 5- System prompts that password is updated.
Alternative Scenario	3a) User enters wrong password. 4a) User saves information without filling all required fields. a) System prompts user to fill all required fields. 5b) Old password not matched b) System prompts user to enter correct password 5c) Server is down.
Special Requirements	None.
Technology	1- Mouse and Keyboard for input 2- Screen for display output 3- Internet connection
Frequency	Many times, a day.

UC-9: Search program

This is the scenario of search or filter the program. For this user first have to login in to his/her account then write program name in search option. If the information is correct and server works properly then system displays the records.

TABLE 2.9 UCD FOE SEARCH PROGRAM

UC-9: Search Program	
Primary actor	User (student)
Goal in context	User will be able to search program
Pre-condition	User must be logged in.
Post-condition	User has searched/filtered program successfully.
Main Success Scenario	1) User inputs program name in search bar. 2) System shows universities which offer that program.
Extensions	2a) User inputs incorrect information 3a) Server is down.
Special Requirements	None.
Technology	1. Mouse and Keyboard for input 2. Screen for display output 3. Internet connection
Frequency	Many times a day.

UCD-10: Search university

This is the scenario of search or filter the university information. For this user first have to login in to his/her account then write university name in search option. If the information is correct and server works properly then system displays the records.

TABLE 2.10 UCD FOR SEARCH UNIVERSITY

UC-10: Search University	
Primary actor	User (student)
Goal in context	User will be able to search university
Pre-condition	User must be logged in.
Post-condition	User has searched/filtered university information successfully.
Main Success Scenario	3) User inputs university name in search bar. 4) System shows university information
Extensions	2a) User inputs incorrect information 3a) Server is down.
Special Requirements	None.
Technology	4. Mouse and Keyboard for input 5. Screen for display output 6. Internet connection
Frequency	Many times a day.

UCD-11: View program

This is the scenario of view all program. For this user first have to login in to his/her account then click on program. If the server works properly then system displays the records.

TABLE 2.11 UCD For View Program

UC-11: View Program	
Primary actor	User (student, web admin)
Goal in context	User will be able to view all programmes
Pre-condition	User must be logged in.
Post-condition	User has viewed programmes successfully.
Main Success Scenario	5) User clicks on programme option. 6) System shows all programme information
Extensions	3a) Server is down.
Special Requirements	None.
Technology	7. Mouse and Keyboard for input 8. Screen for display output 9. Internet connection
Frequency	Many times a day.

UCD-12: View Applicants of a Program

This is the scenario of view all enrolled students of the program. For this user first have to login in to his/her account then click on enrolled students' button. If the server works properly then system displays the records.

TABLE 2.12 UCD For View Applicants of a Program

UC-12: View Applicants of a Program	
Primary actor	User (web admin)
Goal in context	User will be able to view all applicants of the program
Pre-condition	User must be logged in.
Post-condition	User has viewed applicants of a program successfully.
Main Success Scenario	7) User clicks on applicants option. 8) System shows all applicants information
Extensions	3a) Server is down.
Special Requirements	None.
Technology	10. Mouse and Keyboard for input 11. Screen for display output 12. Internet connection
Frequency	Many times a day.

UCD-13: View Merit List

This is the scenario of view merit list of the program. For this user first have to login in to his/her account then click on merit list option. If the server works properly then system displays the records.

TABLE 2.13 UCD For View Merit List

UC-13: View Merit List	
Primary actor	User (web admin)
Goal in context	User will be able to view merit list of the program
Pre-condition	User must be logged in.
Post-condition	User has viewed merit list successfully.
Main Success Scenario	9) User clicks on merit list option. 10) System shows merit list information
Extensions	3a) Server is down.
Special Requirements	None.
Technology	13. Mouse and Keyboard for input 14. Screen for display output 15. Internet connection
Frequency	Many times a day.

UCD-14: View Enrolled Program

This is the scenario of view merit list of the program. For this user first have to login in to his/her account then click on merit list option. If the server works properly then system displays the records.

TABLE 2.14 UCD For View Enrolled Program

UC-14: View Enrolled Program	
Primary actor	User (student)
Goal in context	User will be able to view enrolled programs
Pre-condition	User must be logged in.
Post-condition	User has viewed enrolled programs successfully.
Main Success Scenario	11) User clicks on enrolled programs option. 12) System shows enrolled programs information
Extensions	3a) Server is down.
Special Requirements	None.
Technology	16. Mouse and Keyboard for input 17. Screen for display output 18. Internet connection
Frequency	Many times a day.

UCD-15: Make Program Live

This is the scenario of view merit list of the program. For this user first have to login in to his/her account then click on merit list option. If the server works properly then system displays the records.

TABLE 2.15 UCD For Make Program Live

UC-15: Make Program Live	
Primary actor	User (super admin)
Goal in context	User will be able to make program live.
Pre-condition	User must be logged in.
Post-condition	User has made program live successfully.
Main Success Scenario	13) User clicks on active program option. 14) System makes the program live
Extensions	3a) Server is down.
Special Requirements	None.
Technology	19. Mouse and Keyboard for input 20. Screen for display output 21. Internet connection
Frequency	Many times a day.

UCD-16: Approve Web admin

This is the scenario of view merit list of the program. For this user first have to login in to his/her account then click on merit list option. If the server works properly then system displays the records.

TABLE 2.16 UCD For Approve Web Admin

UC-16: Approve Web Admin	
Primary actor	User (super admin)
Goal in context	User will be able to approve admin
Pre-condition	User must be logged in.
Post-condition	User has approved the admin successfully.
Main Success Scenario	15) User clicks on approve admin option. 16) System approves the admin
Extensions	3a) Server is down.
Special Requirements	None.
Technology	22. Mouse and Keyboard for input 23. Screen for display output 24. Internet connection
Frequency	Many times a day.

2.3.1.2: System Sequence Diagrams

A system sequence diagram is a picture that shows, for a particular scenario of a use case, the events that external actors generate, their order, and inter-system events. All systems are treated as a black box; the emphasis of the diagram is events that cross the system boundary from actors to systems.

SSD 1: Register Account

The scenario is about the create account in which user(student) create their personal accounts

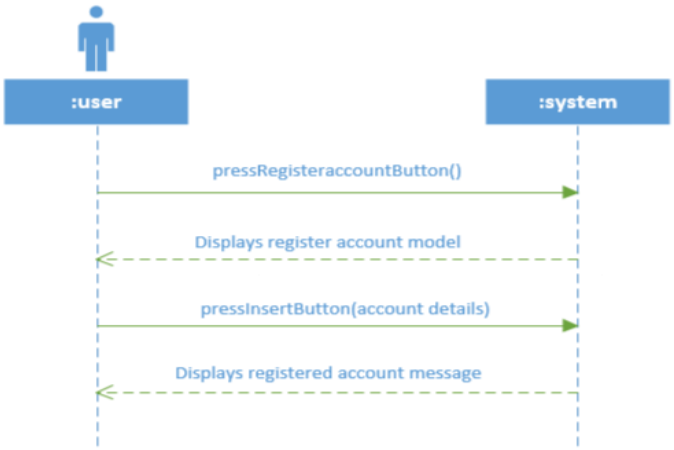


Figure 2.2 SSD for register account

SSD 2: Login account

The scenario is about the login in which user (student) enter the login credentials and then press login button and in response home screen will appear.

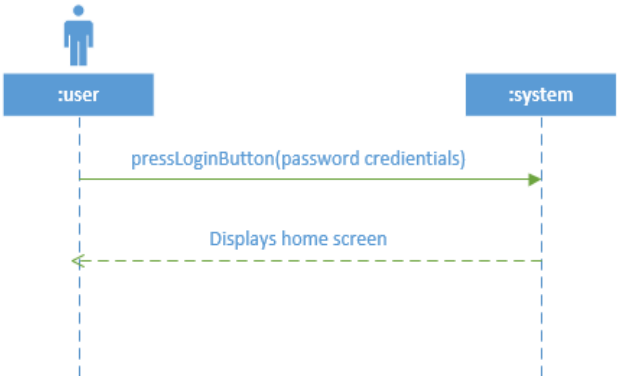


Figure 2.3 SSD for login account

SSD 3: Logout account

The scenario is about the logout account in which user(student) press logout button and in response login screen will appear.

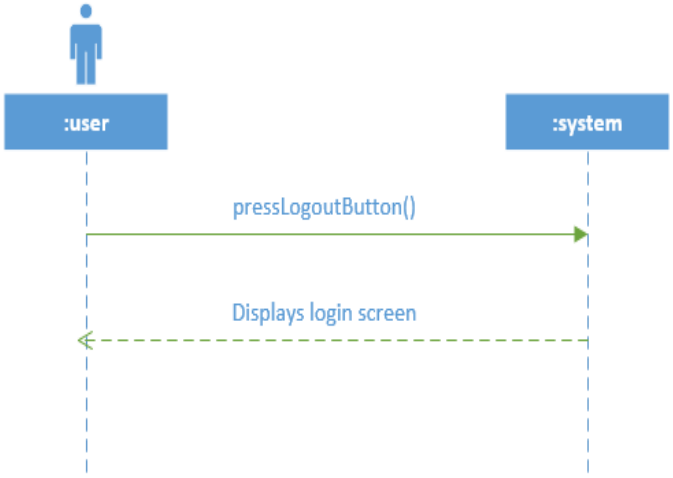


Figure 2.4 SSD for logout account

SSD 4: Create profile

The scenario is about create profile in which user(student) create his/her profile and if the profile is created successfully success message will be displayed

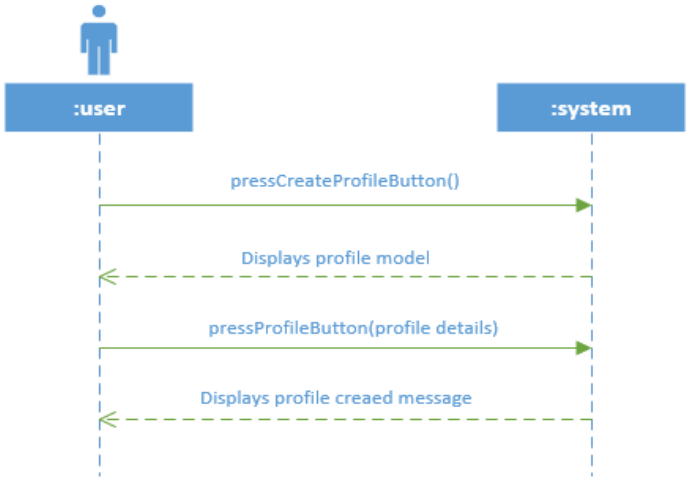


Figure 2.5 SSD for create profile

SSD 5: Modify profile

The scenario is about the update profile in which user update his/her profile and if the details are updated successfully, success message will be displayed

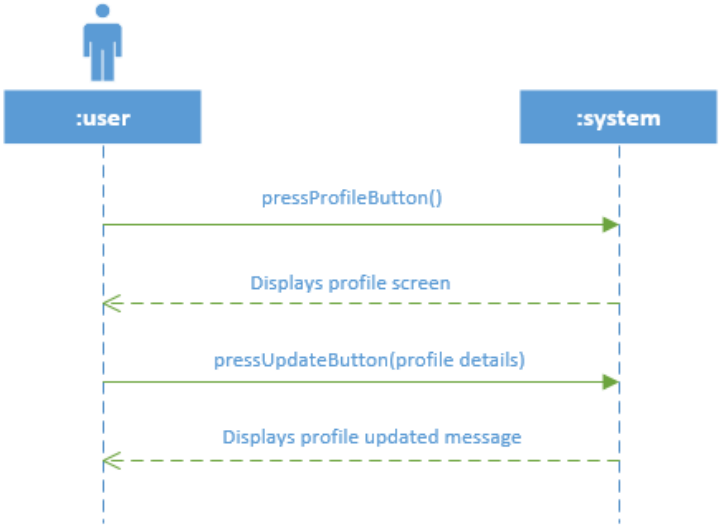


Figure 2.6 SSD for modify profile

SSD 6: View profile

The scenario is about the view profile details in which user views details of his/her profile

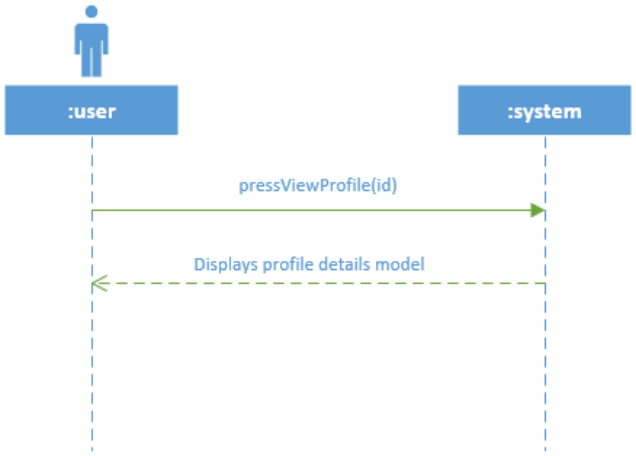


Figure 2.7 SSD for view profile

SSD 7: Apply for program

The scenario is which user(student) will be able to apply for the programme and if the application for programme is successful, success message will be displayed

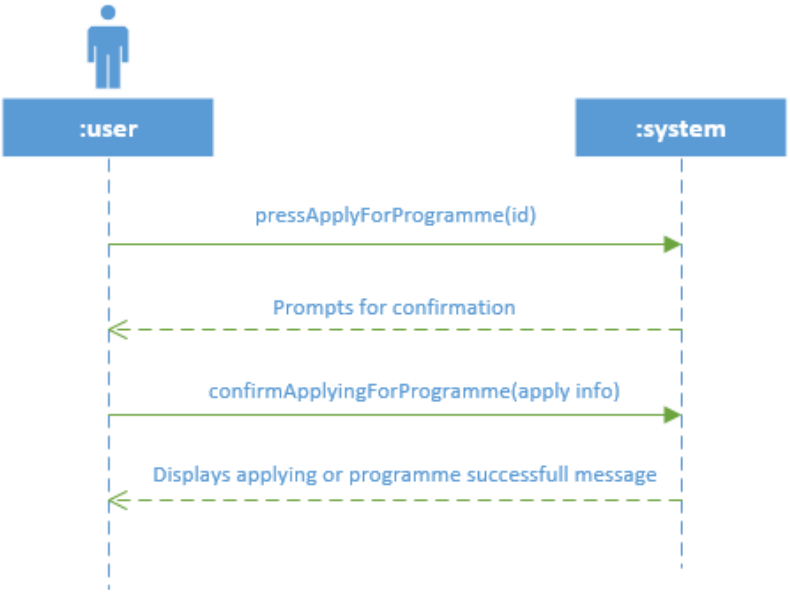


Figure 2.9 SSD for apply for program

SSD 8: Change password

The scenario is about the change password in which user(student) change his/her password.

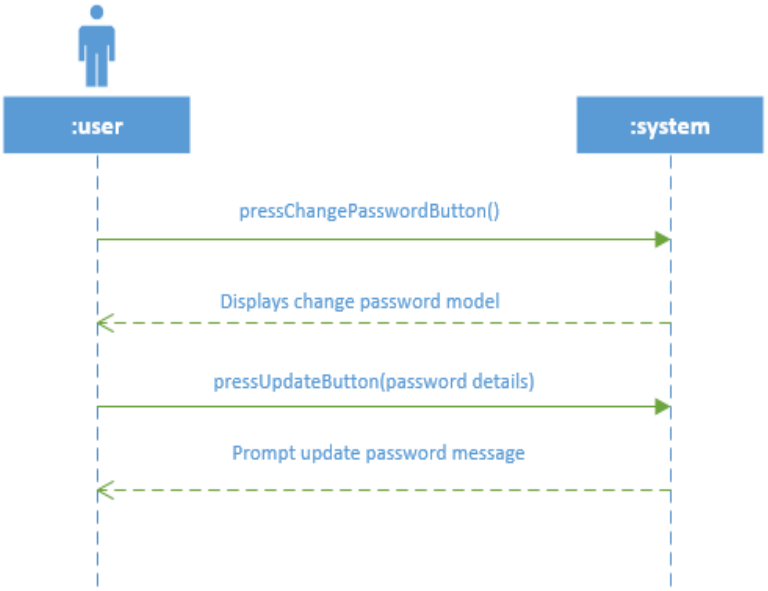


Figure 2.10 SSD for change password

SSD 9: Search for program

The scenario is about the search for programme in which user(student) search for his/her desired programme

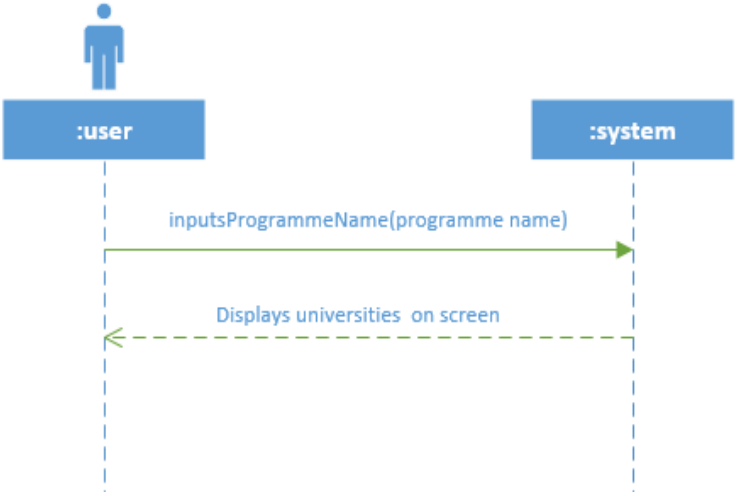


Figure 2.11 SSD for search for program

SSD 10: Search university

The scenario is about the search university in which user(student) search for his/her desired university

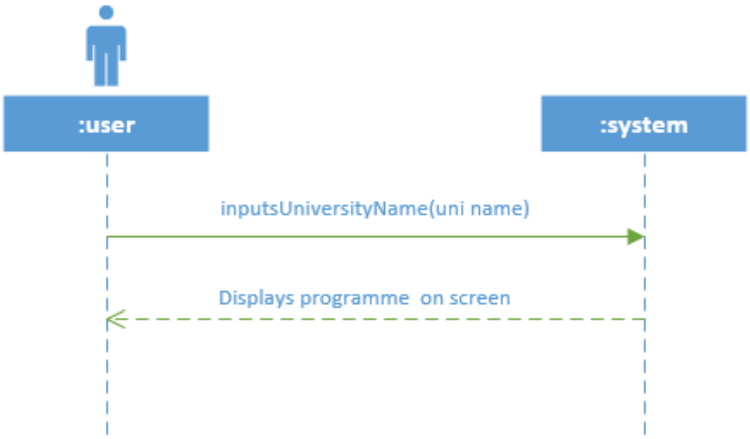


Figure 2.12 SSD for search university

SSD 11: View program

The scenario is about the programme in which user click on the programme option and view all the programmes offered.

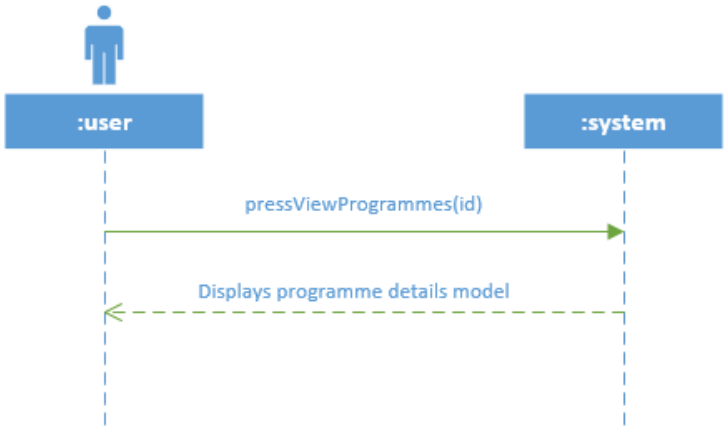


Figure 2.15 SSD for view program

2.3.2: SOFTWARE SYSTEM ATTRIBUTES

Within system engineering, software attributes are realized non-functional requirements used to evaluate the performance of a system.

- **RELIABLITY**
  The system should never crash or hang, other than as a result of an operating system error. The system will provide graceful degradation in the face of network delays and failures
- **AVALIABILITY**

  System should be available to the users all the time

- **SECURITY**
  Privacy between workers is also important to the client because uncertified people using the product may utilize the ability to enter another person profile with negative intent.
  - Access to the various subsystems will be protected by a user authentication (log in screen) that requires user name and password.
  - System have different types of users and every user has access constraints

- **MAINTAINABILITY**

The system code should be written to allow for future possible upgrades. Code should be fully commented. Each method will include a description of its functionality to help in future additions.

- **PORTABILITY**

The IUAP is a web application and to ensure portability, it will be developed in PHP language so it should operate in all famous browsers that includes Firefox, Google Chrome and Internet Explorer

- **PERFORMANCE**

Performance requirements define acceptable response times for system functionality.

- The load time for user interface screens shall take no longer than five seconds.
- The verification of log in information should not take more than five seconds.
- Queries return results should not take more than five seconds.
- User's interfaces should update quickly.
- MySQL database needs to have the capacity to grow.

2.3.3: DATABASE REQUIREMENTS

MySQL database will use for this system to store all information. This database will serve as the backbone of the IUAP It will allow the user information to be captured, stored and then display in various forms to each user. Tables communicate and share information, which will facilitate data search ability, organizing and reporting.

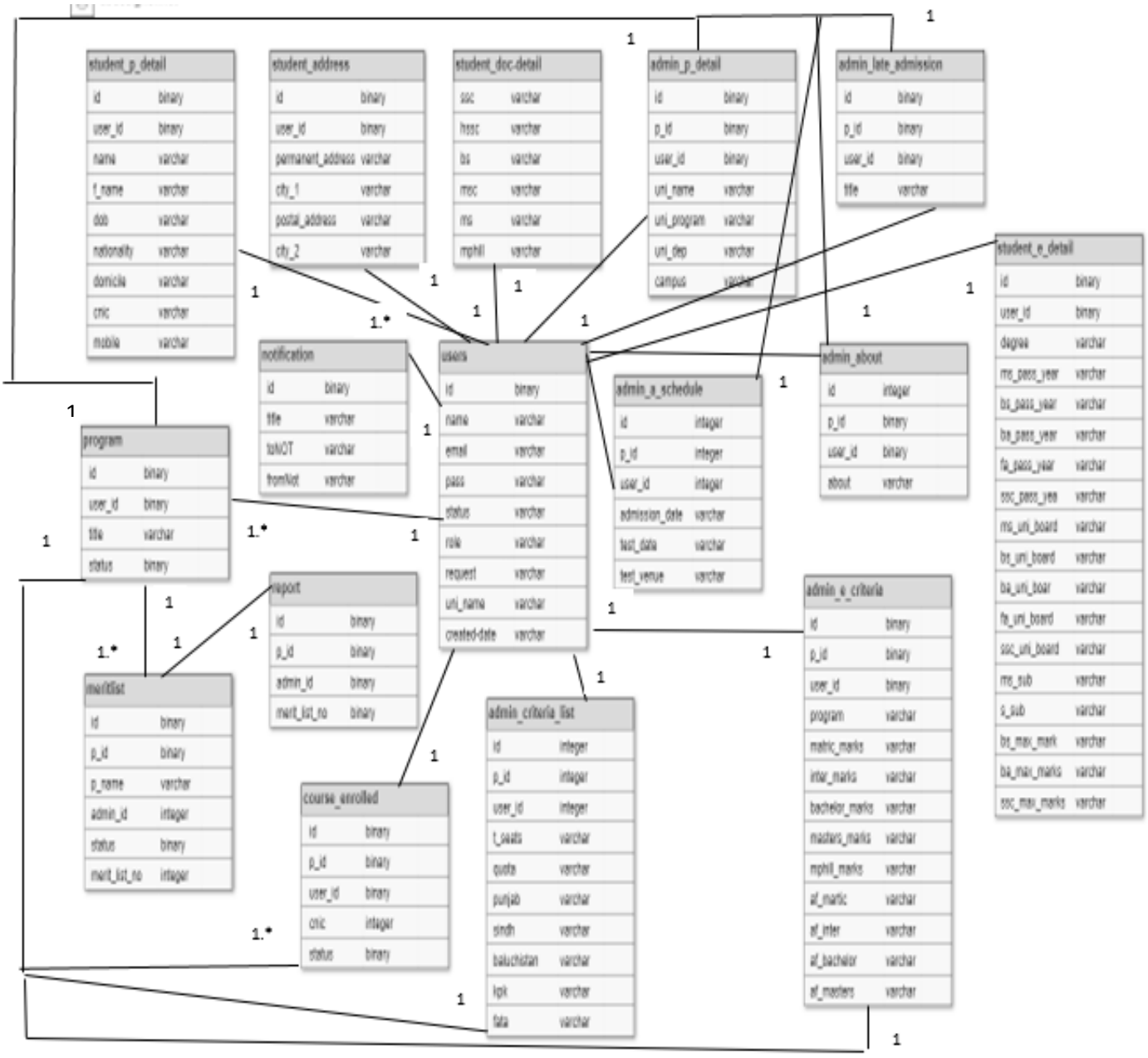


Figure 2.16 Database Design

Chapter 3
Software Design Description

3.1: Introduction

The software design document tracks the necessary information required to effectively define architecture and system design Product Overview. This document is meant to equip the reader with a solid understanding of the inner workings of the IUAP.

3.1.1: Design Overview

The software design document provides design details of IUAP**.** The document contains a complete low-level description of the IUAP, providing insight into the structure and design of each component.

3.1.2: Domain Model

Domain model is an object model of problem domain. It is based on real world classes/concepts and their relationships, that is used to identify the relationships among all the entities within the scope of problem domain

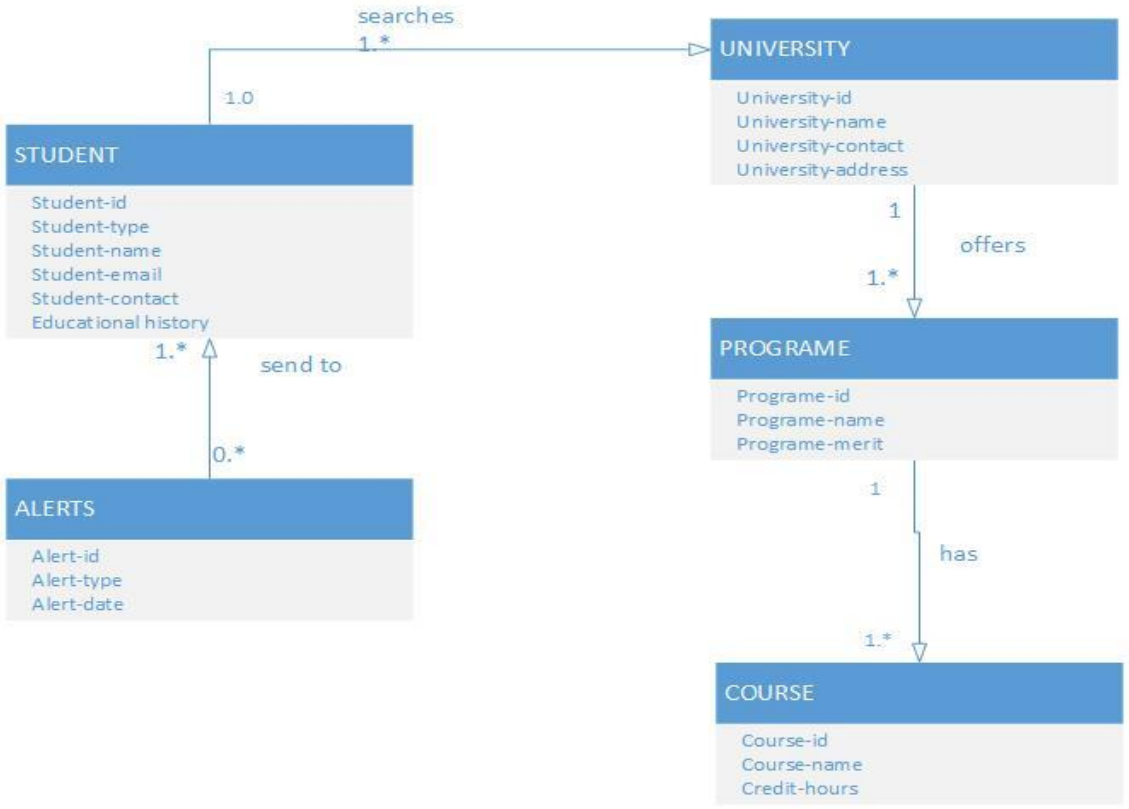


Figure 3.1 Domain model

3.1.3: Activity diagram

Activity diagram is used to describe the dynamic aspects of the system. It is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

AD 1: Login account

In Figure 3.2 a User (web admin, student) enter login credentials such as email and password and then press login button. And if the information is correct then the home screen will appear.

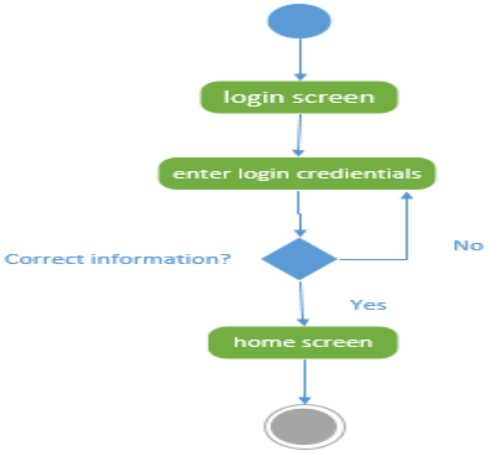


Figure 3.2 AD for login account

AD 2: Logout account

In Figure 3.3 a User (web admin, student) press logout button and if the session destroys then the user will able to logout from account otherwise he/she will remain login in to system.

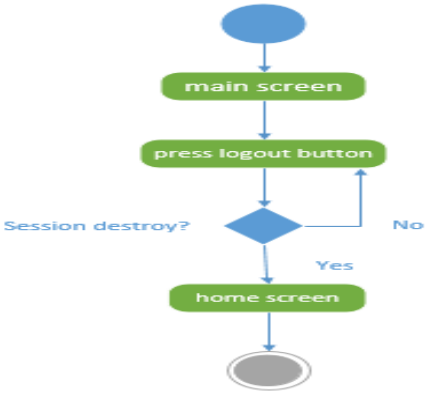


Figure 3.3 AD for logout account

AD 3: Account Operations(student)

In Figure 3.4 a User (student) press profile button then selects option such as update information, view current information or change password. The user can perform any operation and will get a response.

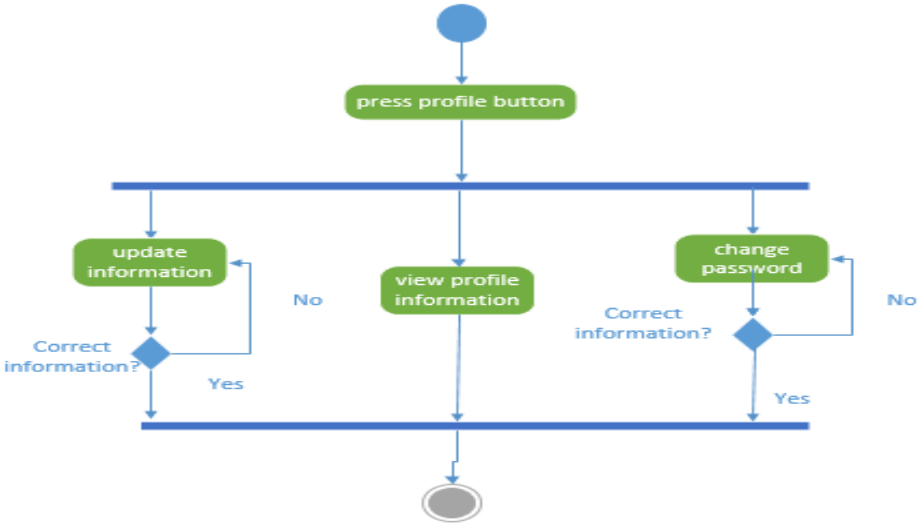


Figure 3.4 AD for account operations

AD 4: Profile operations

In Figure 3.5 a User (student) go to the home screen and then selects option such as view university, view program, search university, search program or apply for program. The user can perform any operation and will get a response.

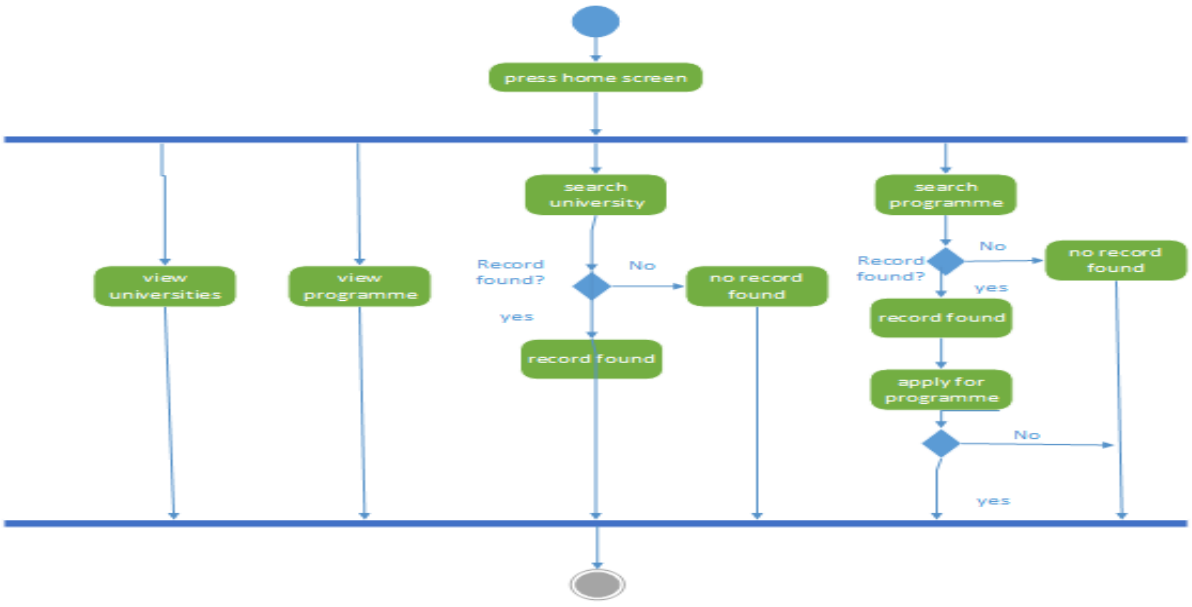


Figure 3.5 AD for profile operations

AD 5: Register account

In Figure 3.7 a User (web admin, student) register account by providing the registration details. And if the information is correct then the home screen will appear.

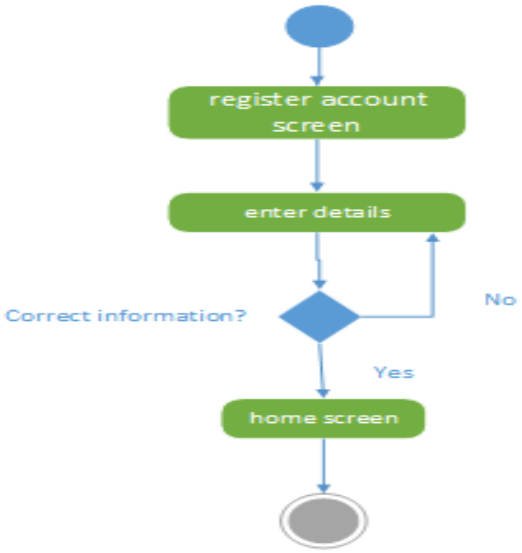


Figure 3.7 AD for register account

3.2: System Architectural Design

This section provides the process for identifying the sub-system making up a system and the framework for sub-system control and communication. It involves identifying major system components and their communications. The system decomposes into several sub systems there is a control relationship between different parts of the system. The identified sub-systems decompose into modules. It enables software engineers to describe in predictable way

3.2.1: Chosen System Architecture

In this section, we describe main architecture of the system. Modules of the system are described at abstract level. The basic architecture of this system is 3-tier model. A 3-tier model uses the client/server computing model.

It has three layers:

- Presentation Layer.
- Application Layer.
- Data Link Layer.

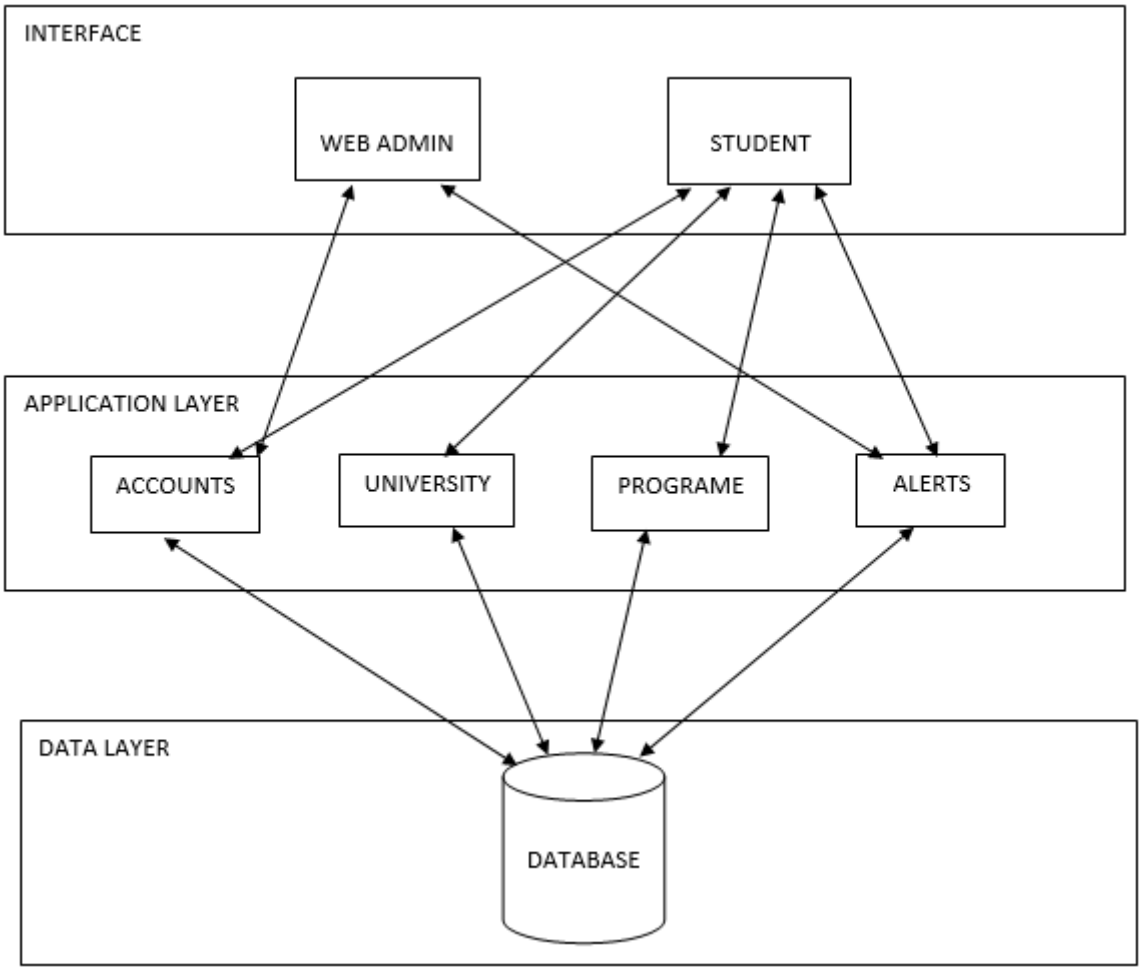


Figure 3.9 Architecture diagram

3.2.2: Detailed Description of Components

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system.

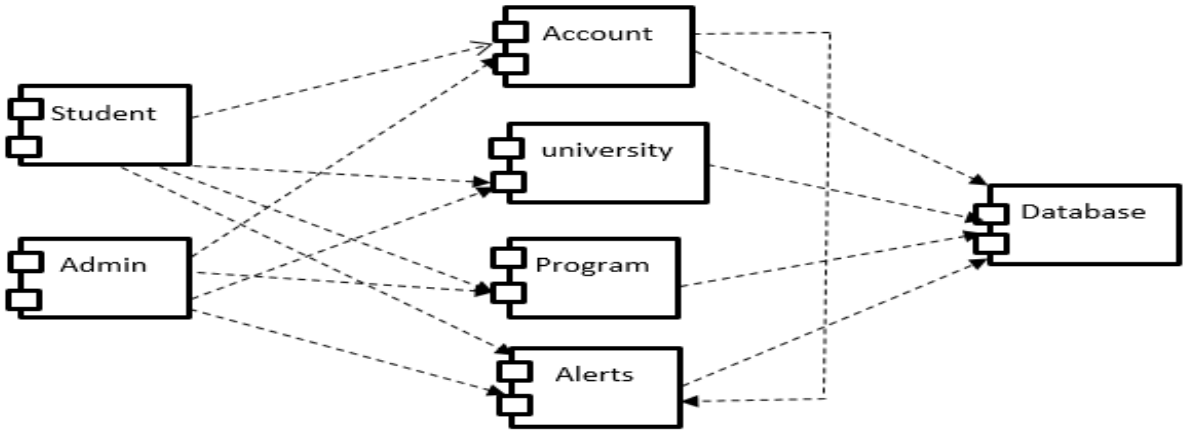


Figure 3.10 Component diagram

- **STUDENT**

  This component interacts with account, university, program and alerts

- **ADMIN**

  This component interacts with account, university, program and alerts

- **ACCOUNT**

  This component has all detail of accounts of every user.

- **UNIVERSITY**

  This component contains details of all universities

- **PROGRAM**

  This component contains detail of all program

- **ALERTS**

  This component contain notification to be send to the users

- **DATABASE**

  All components interact with this component through which they access all the store information.

3.3: CLASS DIAGRAM

A class diagram is a static structure diagram that describes the structure of a system by showing the system's classes, their attributes and the relationships among objects.

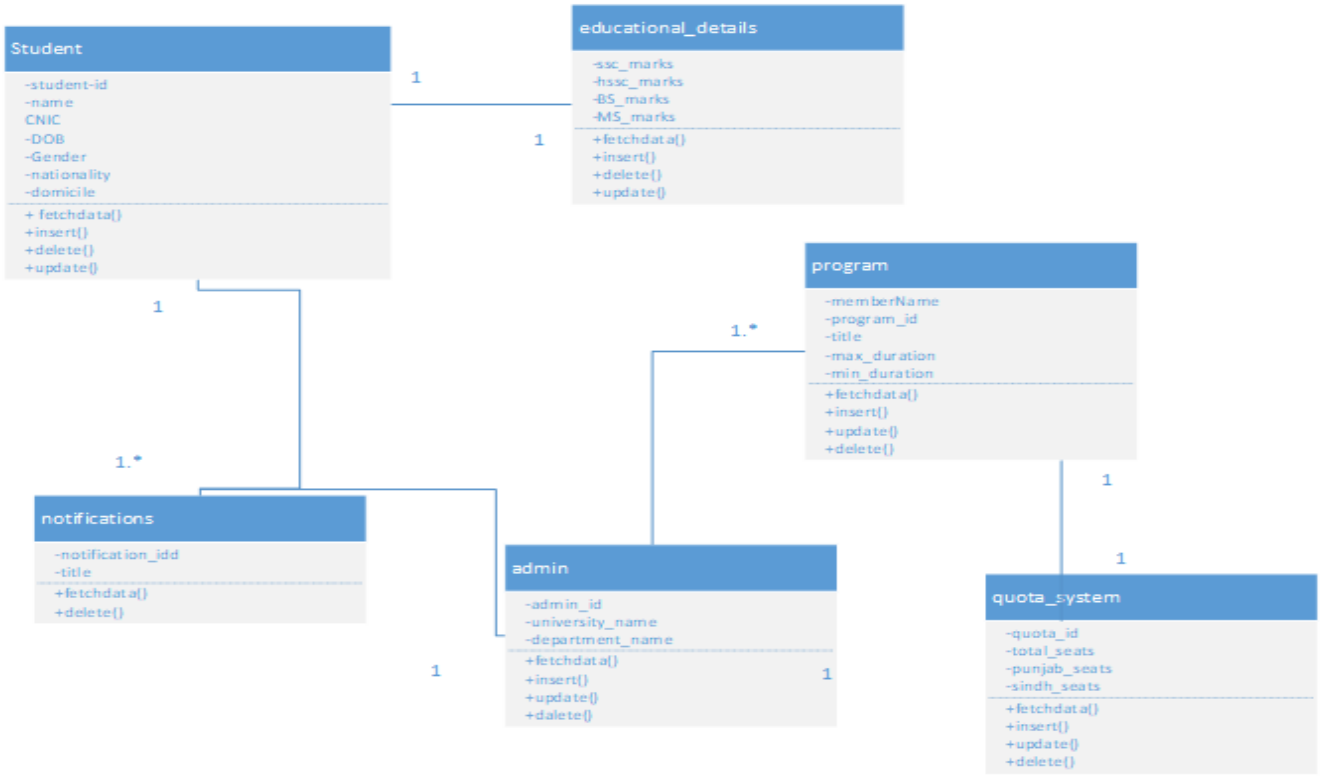


Figure 3.11 class diagram

Chapter 4
Software Implementation Document

4.1: Introduction

This document describes the project implementation for developing the project planner and scheduler.

4.1.1: Language Selection

The project implements in the following languages:

- PHP 5

  Used for PHP is a general-purpose scripting language that is especially suited to server-side web development.

- MySQL 127.0.0.1

  Use for database

- HTML/CSS

  Used for designing of web pages

- JavaScript/JQuery

  Use for scripting and validation.

4.1.2: Tools Selection

- Notepad++
- Xamp Server
- Web Browser
- Sql server

4.2: Application interface

Here are some proposed interfaces of the application:

Home Screen

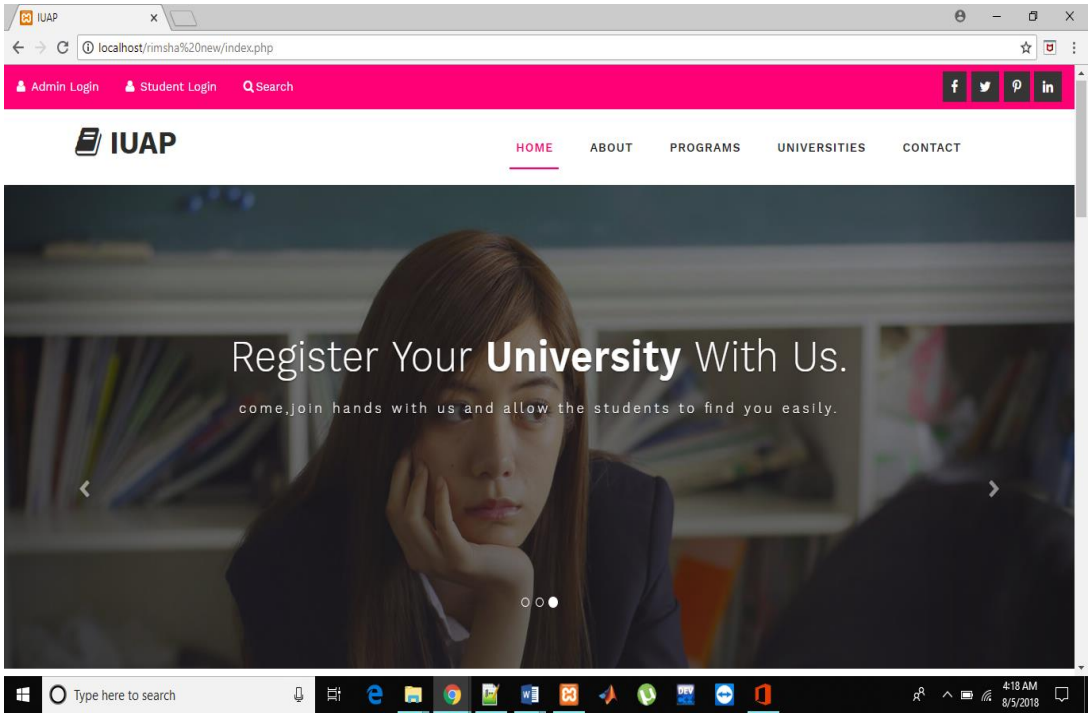


Figure 4.1 Home Screen

Login screen

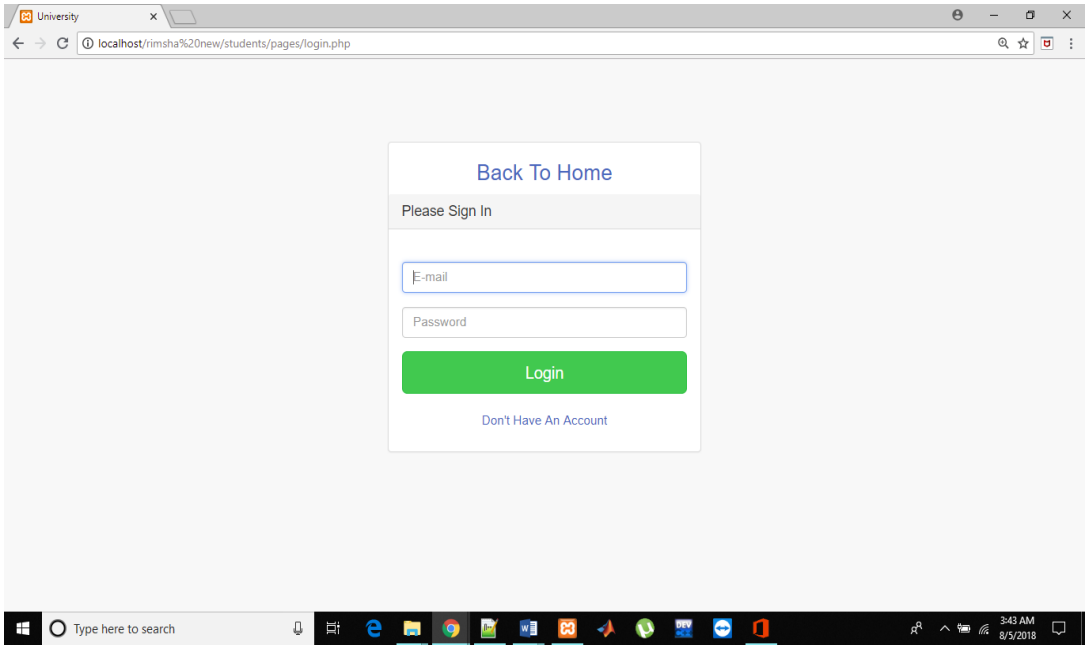


Figure 4.2 Login screen

Register account screen

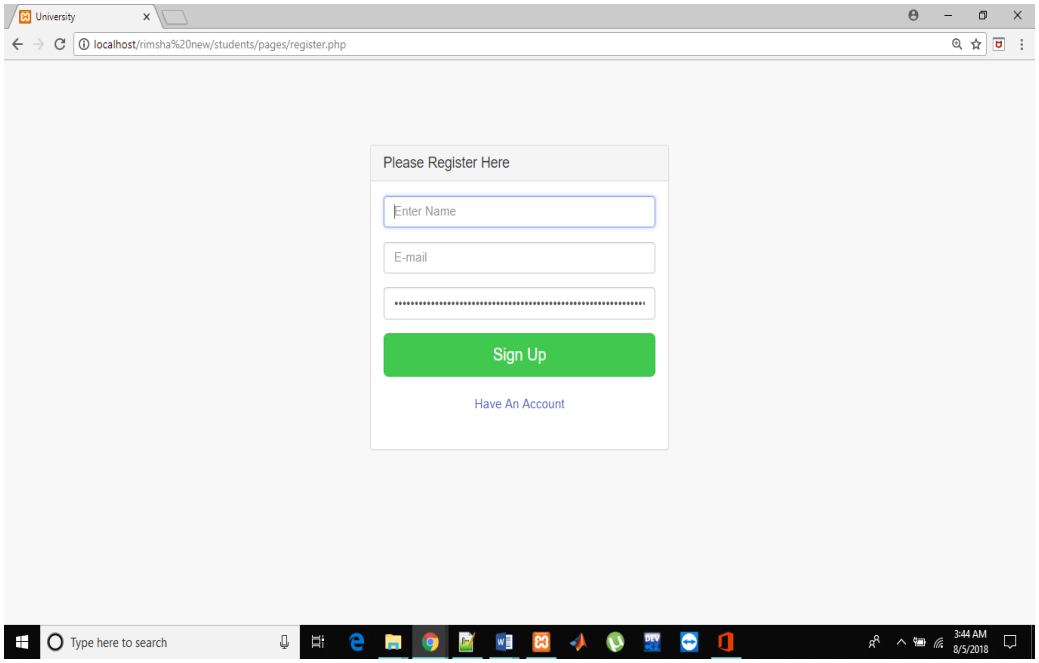


Figure 4.3 register account screen

Enrolled courses screen

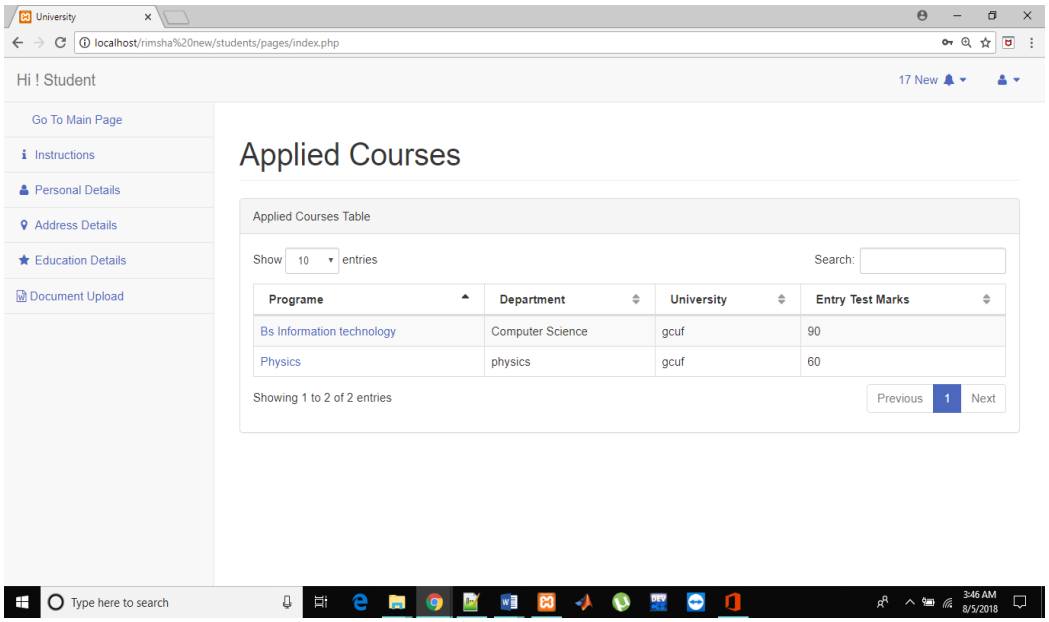


Figure 4.4 Enrolled Courses Screen

Student personal detail screen

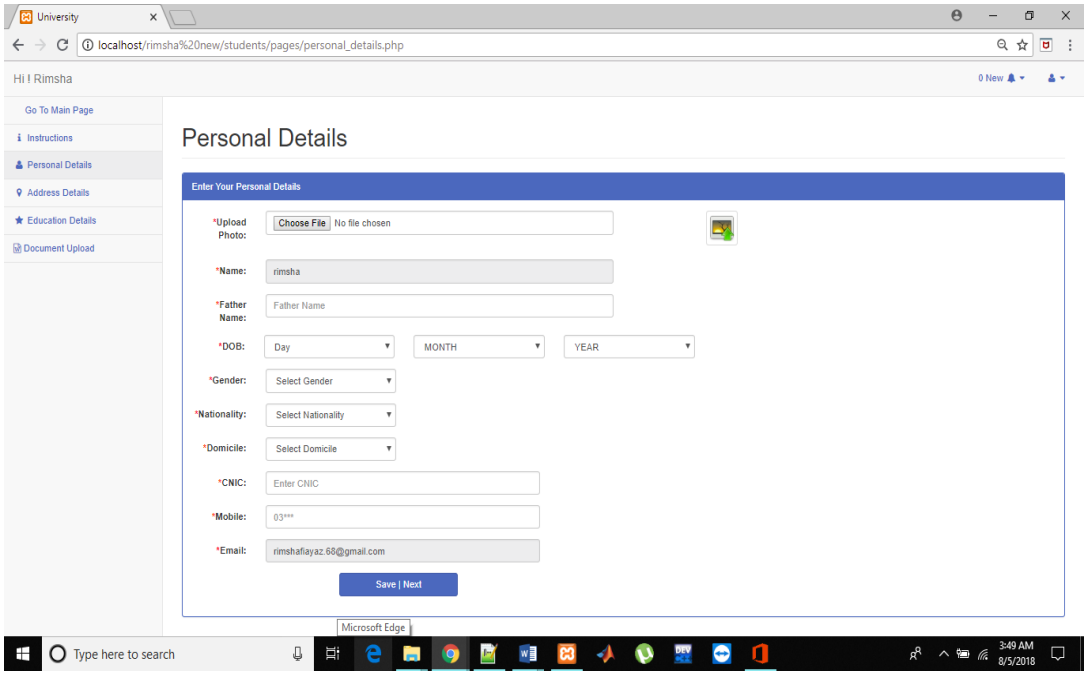


Figure 4.5 Student Personal Detail screen

Student Address Detail Screen

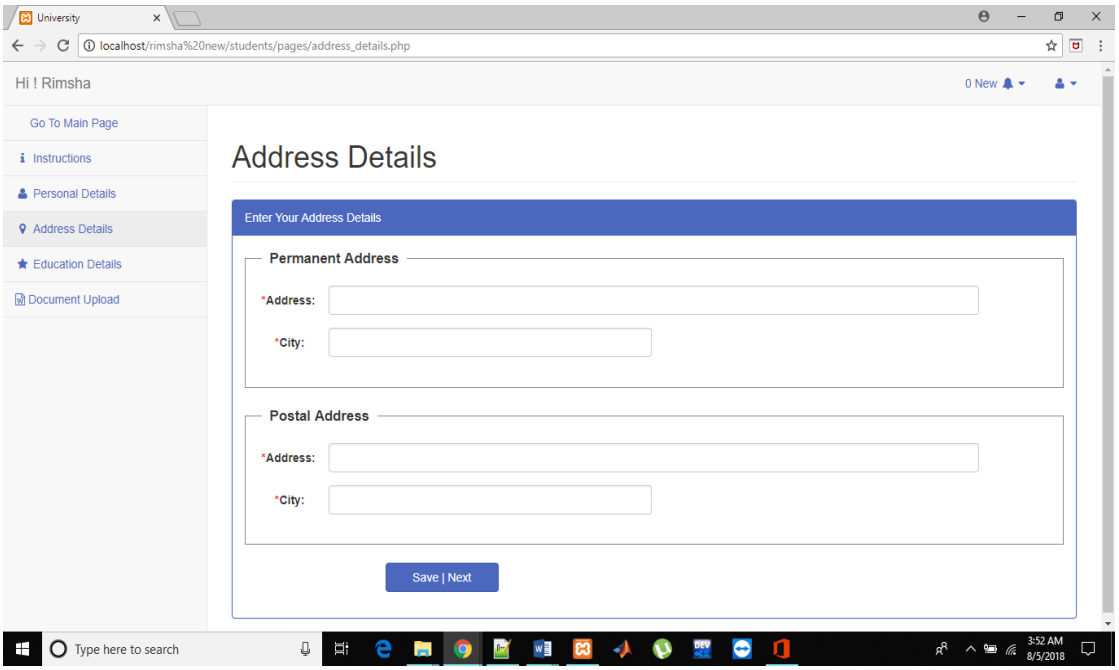


Figure 4.6 Student Address Detail Screen

Student Notification Screen

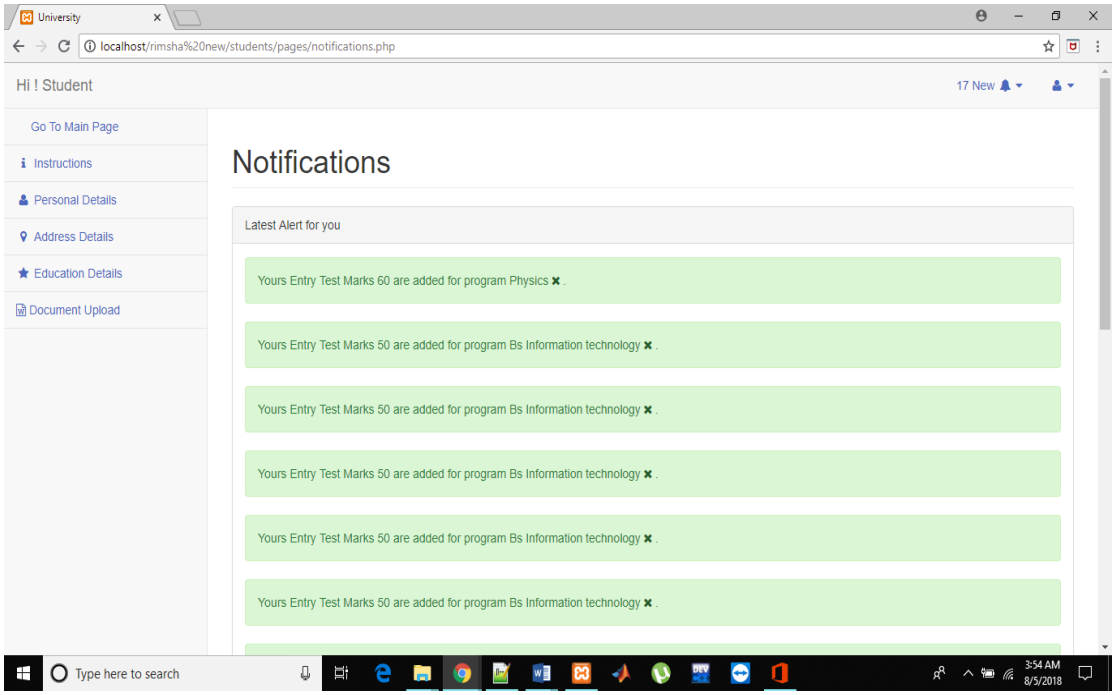


Figure 4.7 Student Notification Screen

Account Setting Screen

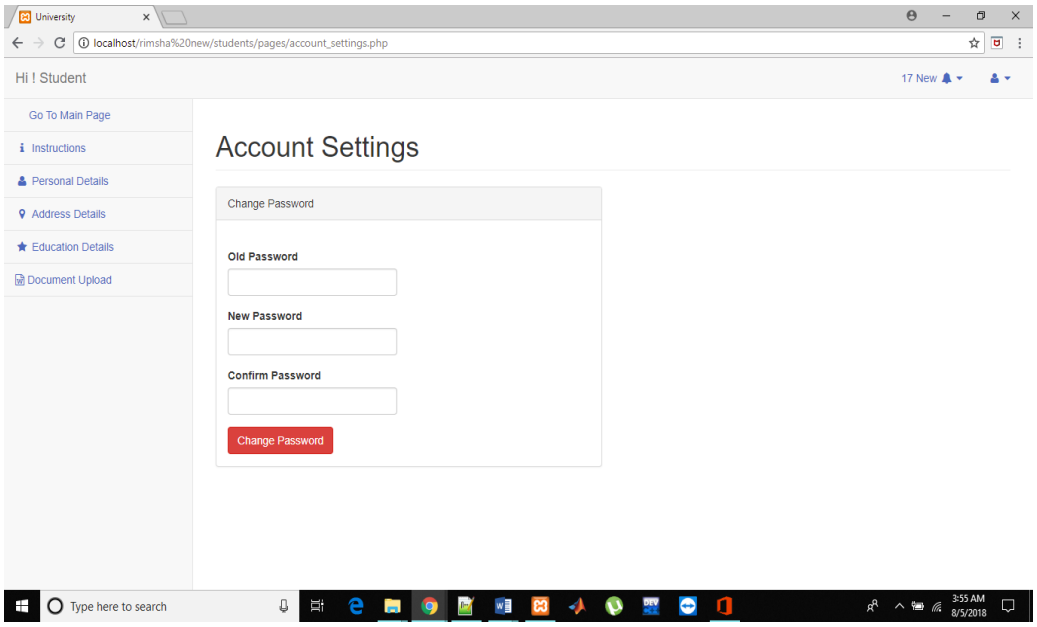


Figure 4.8 Account Setting Screen

Admin Dashboard Screen

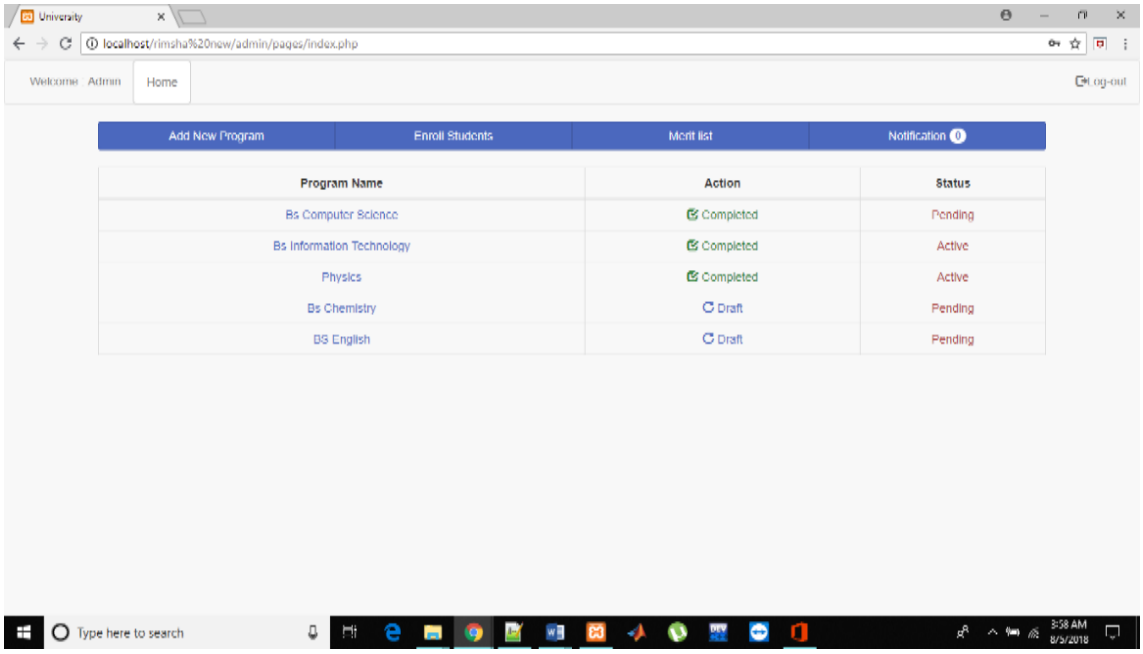


Figure 4.9 Admin Dashboard Screen

Add New Program Screen

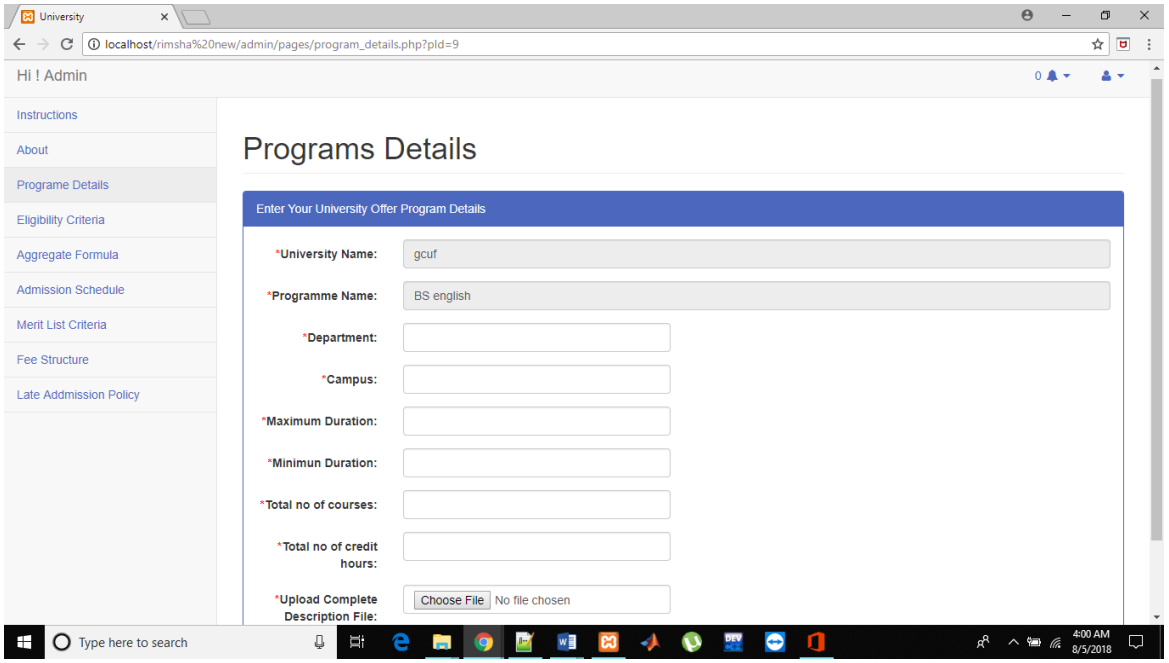


Figure 4.10 Add New Program Screen

Enrolled Students Screen

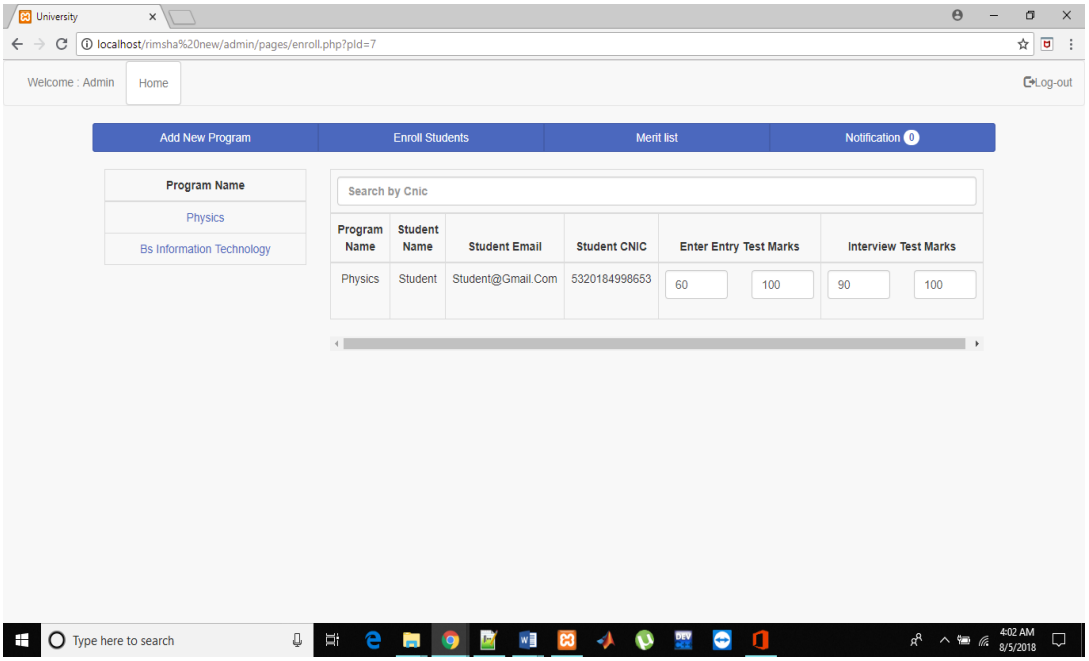


Figure 4.11 Enrolled Students Screen

Super Admin Dashboard Screen

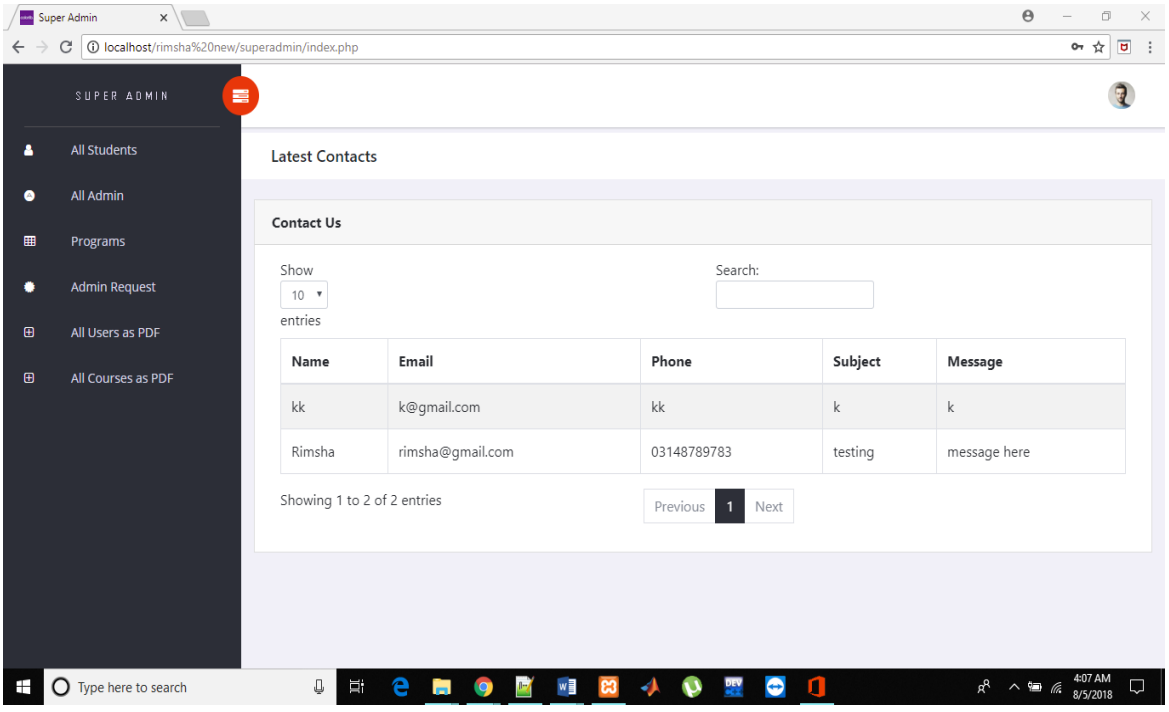


Figure 4.12 Super Admin Dashboard Screen

View All Web Admin Screen

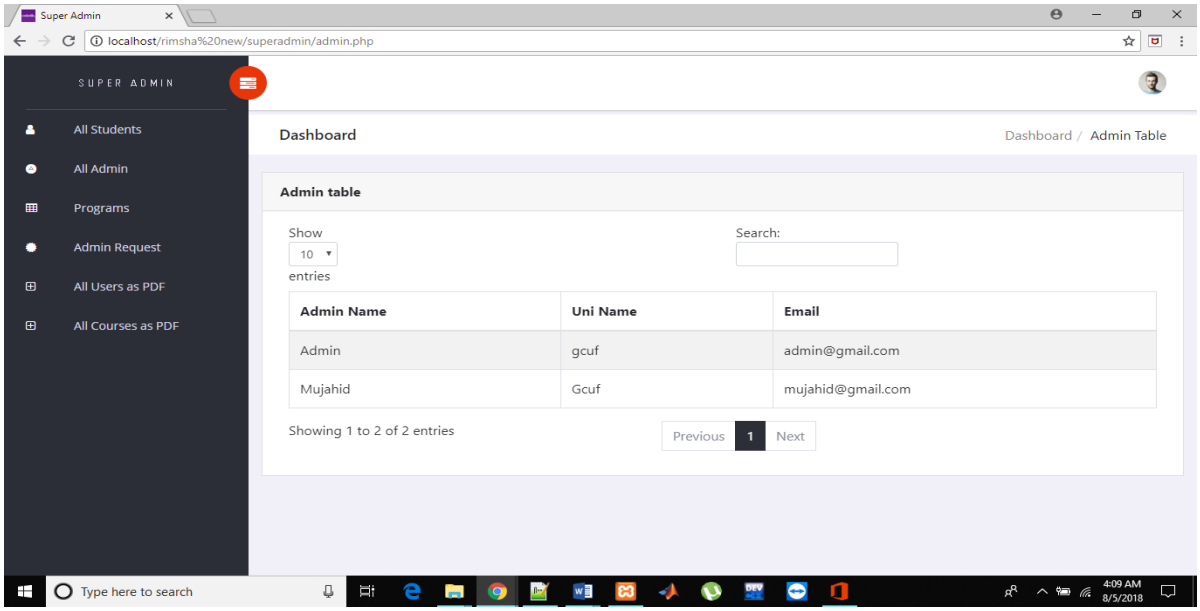


Figure 4.13 View All Web Admin Screen

View Active Program Screen

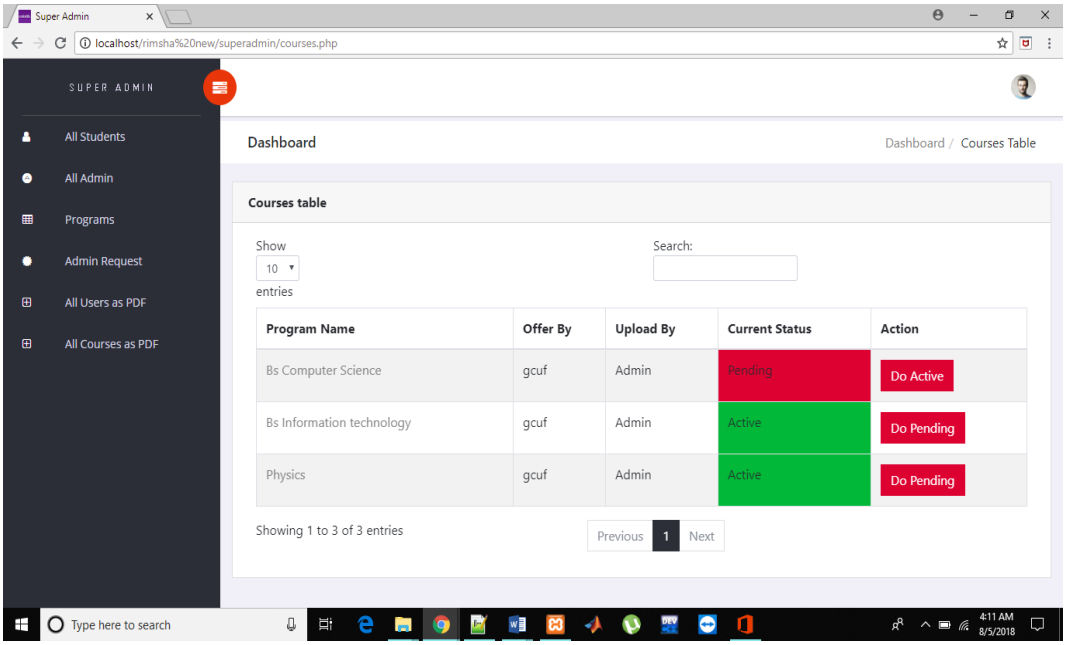


Figure 4.14 View Active Program Screen

Approve Admin Screen

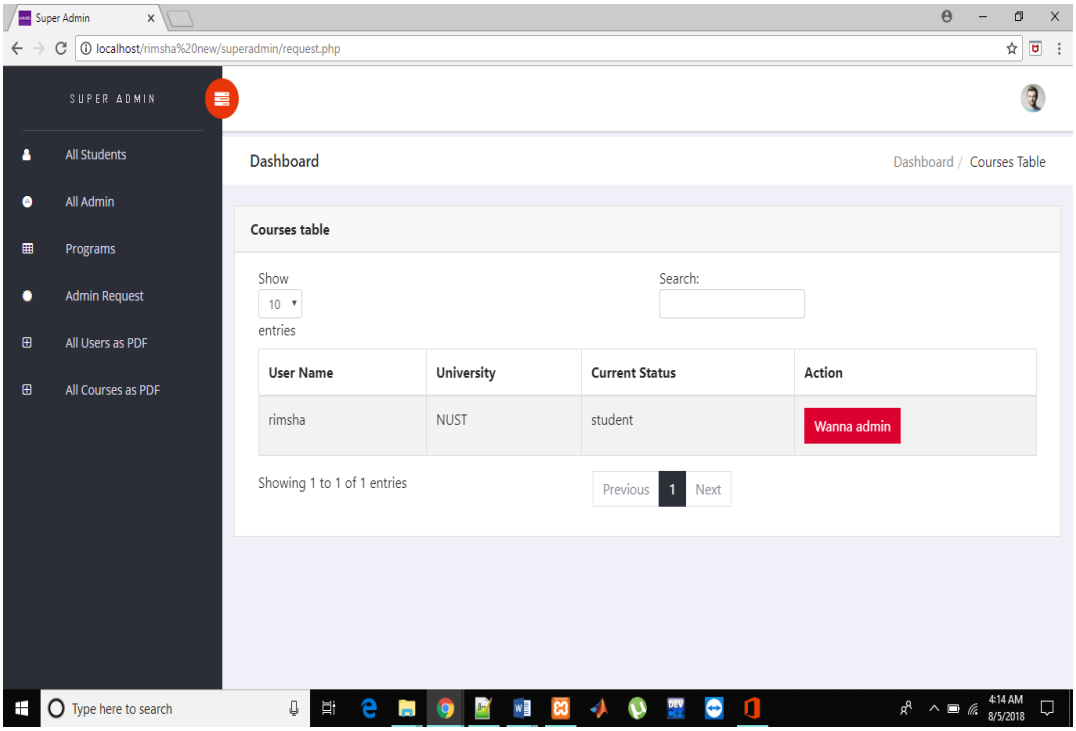


Figure 4.15 Approve Admin Screen

Chapter 5
Software Test Documentation

5.1: Introduction

Software test document is a type of document under which tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application.

This section describes the objectives and extent of the tests.

5.1.1: System Overview

This section, focusing on the structural aspects of testing, provides an overview of the system in terms of the components that are tested during the acceptance test.

5.1.2: Test Approach

The choice of test approaches or test strategy is one of the most powerful factor in the success of the test effort and the accuracy of the test plans and estimates. This factor is under the control of the testers and test leaders.

5.1.3: Acceptance test

We will follow the acceptance test approach to determine whether or not the software system has met the requirement specifications. The main purpose of this test is to evaluate the system's compliance with the business requirements and verify if it has met the required criteria for delivery to end users

Acceptance test is useful, because**:**

- It captures user requirements in a directly verifiable way,
- It identifies problems which unit or integration tests might have missed.
- It they provide an overview on how “done” the system is.

5.2: Test Plan

The test plan focusing on the functional aspects of testing, identifies all features and

combinations of features to be tested. It also describes all those features that are not to be tested and the reasons for not testing them.

5.2.1: Features to be tested

The following is a list of areas to be focused on during testing of the application.

- Register account
- Log-in account
- Log-out account
- Create profile
- Modify profile
- View profile
- Apply for program
- Change password
- Search program
- Search universities
- View program

5.2.2: Features not to be tested

- Power consumption:
  This feature is not to be test because it is not in our scope.
- Memory usage:
  This feature is not test that how much memory consumed by IUAP.

5.2.3: Testing Tools and Environment

Following tools and environments are used for testing:

- Web browser.
- PC/Laptop.
- Windows operating system.

5.3: Test Cases

A test case describes an input, action, or event and an expected response, to determine if a feature of a software application is working correctly. A test case may contain particulars such as test case id, test case name, purpose, input data requirements, steps, and expected results.

TC-1: Register account

Table 5.1 is a test case for register account. This test case tells us about testing of create account scenario.

Table 5. 1 TC for Register Account

TC-1: Register account	
Actor	User (student, web admin)
Purpose	to create an account
Steps	1. User login to account 2. User press the create account button 3. User fills requires information. 4. User press create account option
Inputs	User enter correct information.
	User enter wrong information.
Expected result	Account created successfully.
	Account not created
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-2: Log-in account

Table 5.2 is a test case for login. This test case tells us about testing of login scenario.

Table 5.2 TC for Login Account

TC-2: Log-in account:	
Actor	User (web admin, student)
Purpose	Login to account
Steps	1. Open login page 2. User enter requires information. 3. User press login button
Inputs	User enter correct username and password
	User enter wrong user name and password
Expected result	User login to account
	User not logged in
Actual result	User successfully login in to system
	User not logged in
Pass/Fail	Pass
	Pass

TC-3: Log-out account

Table 5.3 is a test case for logout. This test case tells us about testing of logout scenario.

Table 5.3 TC for Logout Account

TC-3: Log-out Account	
Actor	User(student, web admin)
Purpose	Logout from account
Steps	1. Open login page 2. click on logout button
Inputs	User press logout button
	User click on logout and enter the URL of the same site in next tab.
Expected result	Login screen will appear
	Site should display the login page
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-4: Create profile

Table 5.4 is a test case for create profile scenario. This test case tells us about testing of creating profile scenario.

Table 5.4 TC for Create Profile

TC-4: Create Profile	
Actor	User (student)
Purpose	to create profile
Steps	1. User login to account 2. User press the create profile 3. User fills the information. 4. User submit button.
Inputs	User enters the profile information
	User does not enter the information
Expected result	Profile would be Created.
	Profile not Created.
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-5: Modify Profile

Table 5.5 is a test case for view modify profile information. This test case tells us about testing of modify profile scenario.

Table 5.5 TC for Modify Profile

TC-5: Modify profile	
Actor	User (student)
Purpose	to update profile
Steps	1. User login to account 2. User press the profile button. 3. User fills requires information. 4. User press update option
Inputs	User enter correct information.
	User enter wrong information.
Expected result	Profile updated successfully.
	Profile not updated.
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-6: View Profile

Table 5.6 is a test case for view profile details. This test case tells us about testing of view profile details scenario.

Table 5.6 TC for View Profile

TC-6: View Profile	
Actor	User (Student)
Purpose	to view profile details
Steps	1. User login to account 2. User press view profile button 3. System display profile details
Inputs	User press view profile button
Expected result	User view profile details successfully.
Actual result	As expected
Pass/Fail	Pass

TC-7: Apply for Program

Table 5.8 is a test case for applying for a programme scenario. This test case tells us about applying for a programme scenario.

Table 5.8 TC for Apply for program

TC-8: Apply for Program	
Actor	User (student)
Purpose	To apply for program
Steps	1. User login to account 2. User press the apply for program button 3. User fills the information. 4. User press submit button.
Inputs	User enters the correct information
	User enters the wrong information
Expected result	User has successfully applied for program
	User has not successfully applied for program
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-8: Change password

Table 5.9 is a test case for change password. This test case tells us about testing of change password scenario.

Table 5.9 TC for Change Password

TC-9: Change password	
Actor	User (web admin)
Purpose	to update password
Steps	1. User login to account 2. User press the change password button. 3. User fills requires information. 4. User press update button
Inputs	User enter correct information.
	User enter wrong information.
Expected result	Password updated successfully.
	Password not updated.
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-9: Search/Filter program

Table 5.10 is a test case for search/filter project. This test case tells us about testing of search/filter project scenario.

Table 5.10 TC for Search/Filter Program

TC-11: Search/Filter program	
Actor	User (student)
Purpose	to search/filter program
Steps	1. User login to account 2. User inputs the information. 3. User press the search/filter program.
Inputs	User selects the option.
Expected result	Program would be searched/filtered.
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-10: Search/Filter university

Table 5.11 is a test case for search/filter university. This test case tells us about testing of search/filter university scenario.

Table 5. 11 TC for Search/Filter University

TC-11: Search/Filter university	
Actor	User (student)
Purpose	to search/filter university
Steps	4. User login to account 5. User inputs the information. 6. User press the search/filter university.
Inputs	User selects the option.
Expected result	university would be searched/filtered.
Actual result	As expected
	As expected
Pass/Fail	Pass
	Pass

TC-11: View universities

Table 5.13 is a test case for view university details. This test case tells us about testing of view universities.

Table 5.13 TC for View University

TC-13: View University	
Actor	User (student)
Purpose	to view universities
Steps	1. User login to account 2. User press view universities button 3. System display universities
Inputs	User press view universities button
Expected result	User view all universities successfully.
Actual result	As expected
Pass/Fail	Pass

TC-12: View program

Table 5.14 is a test case for view program details. This test case tells us about testing of view program.

Table 5.14 TC for View Program

TC-14: View program	
Actor	User (student)
Purpose	to view programs
Steps	4. User login to account 5. User press view program button 6. System display program
Inputs	User press view program button
Expected result	User view all program successfully.
Actual result	As expected
Pass/Fail	Pass

TC-12: View enrolled students:

Table 5.15 is a test case for view program details. This test case tells us about testing of view program.

Table 5.15 TC for View Enrolled Students

TC-14: View enrolled students	
Actor	User (web admin)
Purpose	To view enrolled students.
Steps	7. User login to account 8. User press view enrolled students button 9. System display enrolled students
Inputs	User press view enrolled students button
Expected result	User view all enrolled students successfully.
Actual result	As expected
Pass/Fail	Pass

Chapter 6
Conclusions and Future Enhancements

6.1: Introduction

This document describes the project conclusions and future enhancements i.e. what type of new features can be added with time.

6.2: Summary

This application allows students to apply in different institutions with ease as well as facilitate universities by automating calculating merit of different students and generating merit lists

6.3: Conclusions

- Students are now able to apply in different institutions by eliminating burden of collecting forms from different institutions.
- Institutions are at much ease because of automatic merit list generations and automated sending alerts to admitted students.
- Search are able to search for their favorite institution and program at one platform.

6.4: Future Enhancements

In future application can be enhanced by:

- IUAP is web-based application but in future it can developed for android and iOS devices.
- The alerts or notifications can be sent on mobile phones.