PROJECT REPORT

On

Automatic Time Table Generator

Prepared by TEAM-6

Table of Contents

1. Introduction

1.1 System Introduction

An Automatic time table generator is a Web-based Application Software that will help to generate time table for our institute automatically. Also, user can manage data like adding, updating details, deleting and viewing of faculty, programs and courses.

1.2 System Objective

The purpose of developing Automatic Time Table Generator is to automate the process of developing time table for our institute.

1.3 System Scope

The scope of Automatic Timetable Generator is to provide easy and efficient system for automating the process of developing timetable. The main scope of our system is to generate feasible time table by taking required inputs.

1.4 Motivation

Currently, timetable for our institute is manually prepared by a committee which includes UG convener, PG convener, Dean of Academic Programs, Professor Puneet Bhateja and Professor Asim Banerjee. As the process of generating time table is time and effort consuming, it creates workload on committee members as they have other responsibilities also.

This project aims at the development of a web based application software which will help our institute to generate timetable for our college. This web app is developed in PHP which will work with the database developed in MYSQL.

1.5 Technology & Literature Overview

- **To develop Automatic TimeTable Generator technology use for**

    1. **Web Application :** HTML5, CSS3, PHP
    2. **Database:** MySQL

- **Literature**

    1. Automatic TimeTable Generator and its paperwork.

2. System Requirement Study

2.1 Functional Requirements

Automatic TimeTable Generator involves the following functions.

1. **Web Application**

    This web application is mainly intended to only one type of user.

    i.e. **ADMIN**

1. **Login :** The admin can log in to his dashboard using login page which is the first page provided in the Web App System.

2. **Manage Faculty:** Admin can manage faculties. He/She can add faculty, update faculty, delete faculty and view them.

3. **Manage Programs:** Admin can manage programs. He/She can add programs, update programs, delete programs and view them.

4. **Manage Courses:** Admin can manage Courses. He/She can add courses, update courses, delete courses and view them.

5. **Generate TimeTable:** For generating the time table, Admin has to give the Student Registration details and course faculty details and this system generate the feasible timetable accordingly.

6. **Logout:** Admin can log out easily.

2.2 Non-Functional Requirements

1. **Performance:** Time taken to generate the time table and satisfying the constraints are two major performance areas which must be taken care of.

2. **Availability:** This Web Application is only available to the authorized user to view and manage details of faculty, programs, classrooms and courses and accordingly generate the timetable.

3. **Maintainability:** The system can be maintained without any hindrance and it shall respond as fast as possible in generating the timetable.

4. **Security:** The security is one of the primary requirement of any application.Only authorized user can generate time table and create, update and manage details regarding faculties, courses and programs.

3. System Analysis

3.1 Study of System

3.1.1 Study of current system

Currently the time table is prepared by Dean of Academic Programs, PG convenor, UG convenor, Professor Puneet Bhateja and Professor Asim Banerjee manually. They have to keep track of slots, faculties and courses. This makes it more time consuming.

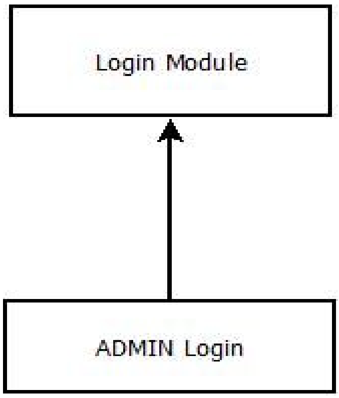
3.1.2 Proposed New System

So we are going to make an Automatic time table generator which is a Web-based Application Software that will help to generate time table for our institute automatically. It takes data like Faculty Names, Courses and Programs and generate the feasible time table. Timetabling is a task of satisfying some constraints. These constraints are hard constraints and soft constraints. In this project hard constraints have been taken care of strictly and it has been ensured that soft constraints are as well followed as much as possible.

3.2 System Modules & Description

3.2.1 Web Module

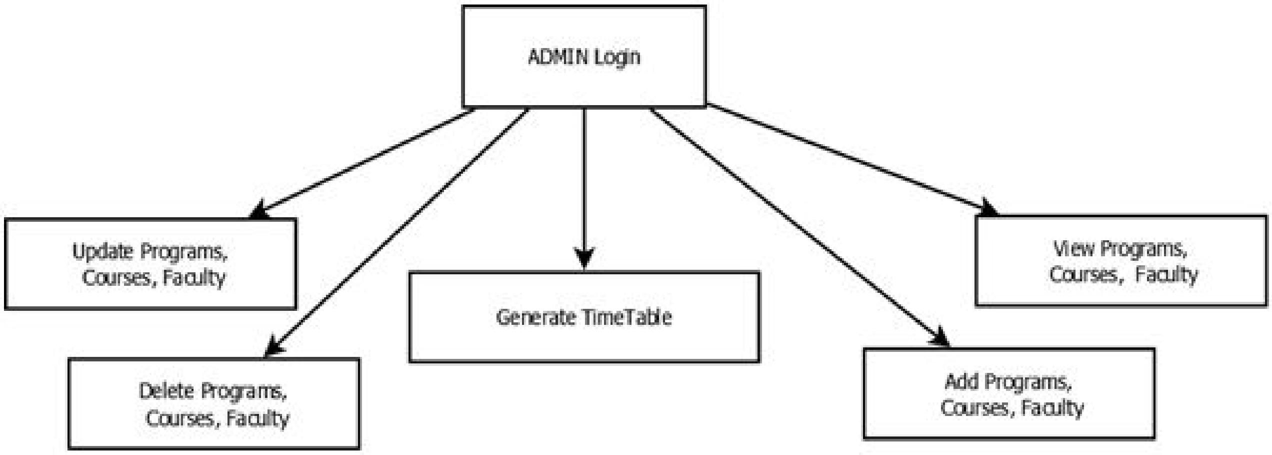
1. **Login Module**



Description:

The user is required to login using the username and the password, to access the web-app. The login credentials are verified, and checked if the user is a valid user or not. Based on that, the features are made available on the web-app for access.

2. ADMIN Module


ADMIN Login
Update Programs, Courses, Faculty
Delete Programs, Courses, Faculty
Generate TimeTable
View Programs, Courses, Faculty
Add Programs, Courses, Faculty

Description:

1. **Add Programs, Courses, Faculty :**

   Using this feature, admin can add any new program, courses or faculty.

2. **Update Programs, Courses, Faculty :**

   Using this feature, admin can update the details of any existing program, courses or faculty.

3. **Delete Programs, Courses, Faculty :**

   Using this feature, admin can delete any existing program, courses or faculty.

4. **View Programs, Courses, Faculty :**

   Using this feature, admin can view the lists of program, courses or faculty.

5. **Generate Time Table :**

   Using this feature, Admin has to give the Student Registration details and course faculty details and this system generate the feasible timetable accordingly.

3.3 System Characteristics

- The web app will have the dependency for making of the time table of the lectures in the system.

- The central objective of the system is to develop an interface that will ease the process of generating time table. The system will make process hassle free for the end user.

- The system will also help to manage faculties, courses and programs.

3.4 Feasibility Study

3.4.1 Economical feasibility:

Economic feasibility defines that whether development of this project is financially beneficial for client and developer or not.

As we are developing this application in a PHP Technology and it is an open source technology so all the required tools and server configuration are freely available. So our system is Economically Feasible.

3.4.2 Operational feasibility:

Operational Feasibility defines the whether all the client requirements are satisfied or not.

We studied our system and gathered requirements in detail for the system by interviewing every stakeholder. Based on that study we are able to develop the system.

3.4.3 Technical Feasibility:

We develop this system in PHP Technology. We are using following tools and technology.

Front End : Technology - HTML5, CSS3, Bootstrap
Tool - Sublime Text

Back End : Technology - Php, MySQL
Server - Apache

We have all necessary tools available to develop project so our system is Technically feasible.

4. Project Management

4.1 Project Planning

The People

We have a group of 10 people and the requirement gathering is done by all of us.

After that, the Project is divided into 3 Part are as follows

1. Designing
2. Coding & Database

3. Documentation

We assign 3 people for Designing, 4 for Coding & Database 3 for Documentation.

Project Area	**No of People**
Designing	**3**
Coding & Database	**4**
Documentation	**3**

4.2 Project Estimation

For estimating a project, we are estimating the system using COCOMO model. In which we select our system as Semi-detached System and the parameter value of a and b is 3.0 and 1.12 respectively.

Count of Lines of Code of our system is 1.905K

Also Project Complexity is as follows :-

Complexity	Nominal	1.00
Storage	Nominal	1.00
Experience	Very Low	1.29
Programmer Capability	Low	1.17

Now, **Effort Adjustment Factor** is calculated as follows :-

EAF = 1.00*1.00*1.29*1.17

= 1.5093

Initial effort Estimation :-

$$\mathbf{E_i} = \mathbf{a*(KLOC)^b}$$
$$= 3.0*(1.905)^{1.12}$$
$$= 6.17452612$$
$$= 6.17PM$$

E(Effort Estimation) = 6.17*1.5093

= 9.3124 PM

Overall Scheduling :-

For overall scheduling using semi-detached type we need to use following equation.

M (Months) = $2.5*(E_i)^{0.35}$

= $2.5*(9.3124)^{0.35}$

= 5.5

4.3 Project Scheduling

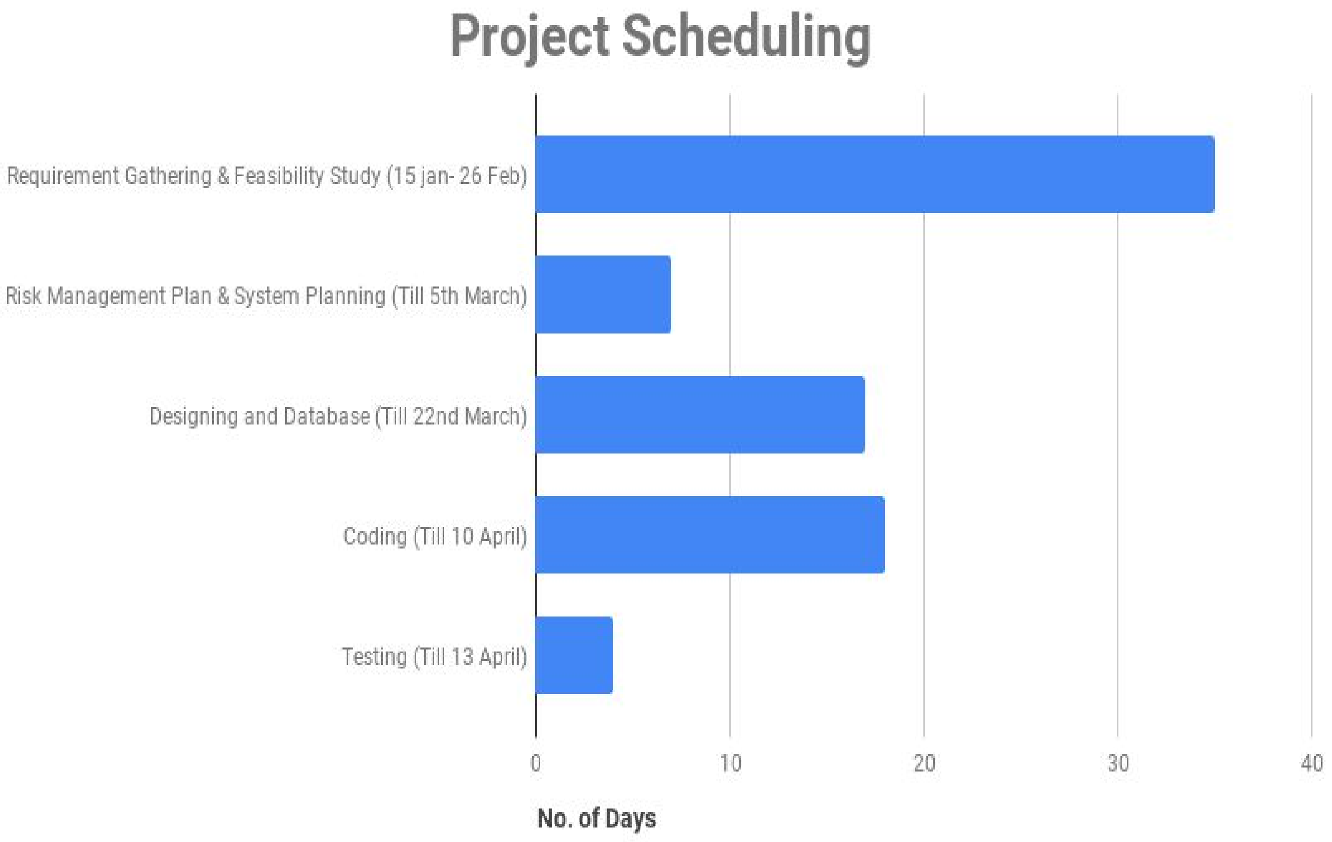


Fig. 4.3 Project Timeline

4.4 Risk Management Plan

Sr. No	Risk	Risk Type	Probability of occurrence	Impact	Impact Type	Impact (1-5)	Priority (P*I)	Response
1.	No one in our group has done this type of project before	Technical	0.5	Because no one in group has develop before so if an issue or error occur then it will take more time to resolve.	Moderate to High	3	1.5	Every team member is taking this in a positive manner as all the members are willing to learn something from this project.
2.	Product is complex to implement	Technical/ Product	0.4	It fails to run or possibly give no or wrong output.	High	4	1.6	We have spent maximum of our time in finding logic.
3.	Fail to address priority constraints	Product	0.6	If we fail to consider hard constraints, then a faculty can assign two lectures at same time in two different classes. If we fail to consider soft constraints, then a faculty might have to sacrifice with his choices.	Moderate to High	2	1.2	The first priority will be given to the hard constraint and then soft constraint because as of now also the manual system follows the same.
4.	Problem is not properly understood by the team	Team	0.3	We may deliver different product than expected.	High	5	1.5	We will make sure to get information from various people and interpret the solution they want.
5.	Team members may fail to understand some requirements	Team	0.3	Quality of the product may suffer.	Moderate	2	0.6	As we are 10 people in the team and we are gathering the requirements from many people we would be able to understand proper requirements and

								also we can solve the doubts about any requirement by asking the stakeholders and get it properly.
6.	Availability of team members	Team	0.3	Weak understanding among team members, less coordination and time management will become difficult.	Low	1	0.3	Discussion regarding general issues can be done online in group and discussion regarding important topics are done through meetings which are stated prior.
7.	Competition	Business	0.6	If our competitor provides better solution than us, then our product might not be use.	Moderate	3	1.8	We will try to make a web application which is user friendly so that even non-tech people can use.
8.	Strategic risk	Business	0.5	If strategies we decide become less efficient then objectives are not easily met.	Moderate	2	1	We can discuss the strategies with our faculty or TA's and get their guidance to work in proper way.
9.	Training human resource	Business	0.3	It will require more effort to train other team member who are not fully aware of technology so it will lead in consumption of time and effort. The team members who are aware of the technology may fail to guide another member properly	Moderate	2	0.6	Team members are ready to give extra time to learn new language in which project will be made.
10.	Lack of commitment	Team	0.2	Increase the workload on other team	Low	1	0.2	Active team members will guide and enforce

				members and work will not proceed as plan.				non-participating members to accomplish a given work in particular time.
11.	Difficulty in integration of modules	Technical/Product	0.4	Leads to rework.	Moderate	3.5	1.4	To resolve this problem before merging the individual modules, we will first ensure that all the individual modules work properly without any error and also ensure that they give expected outcome.

5. System Diagrams

5.1 Flow Chart

Web App

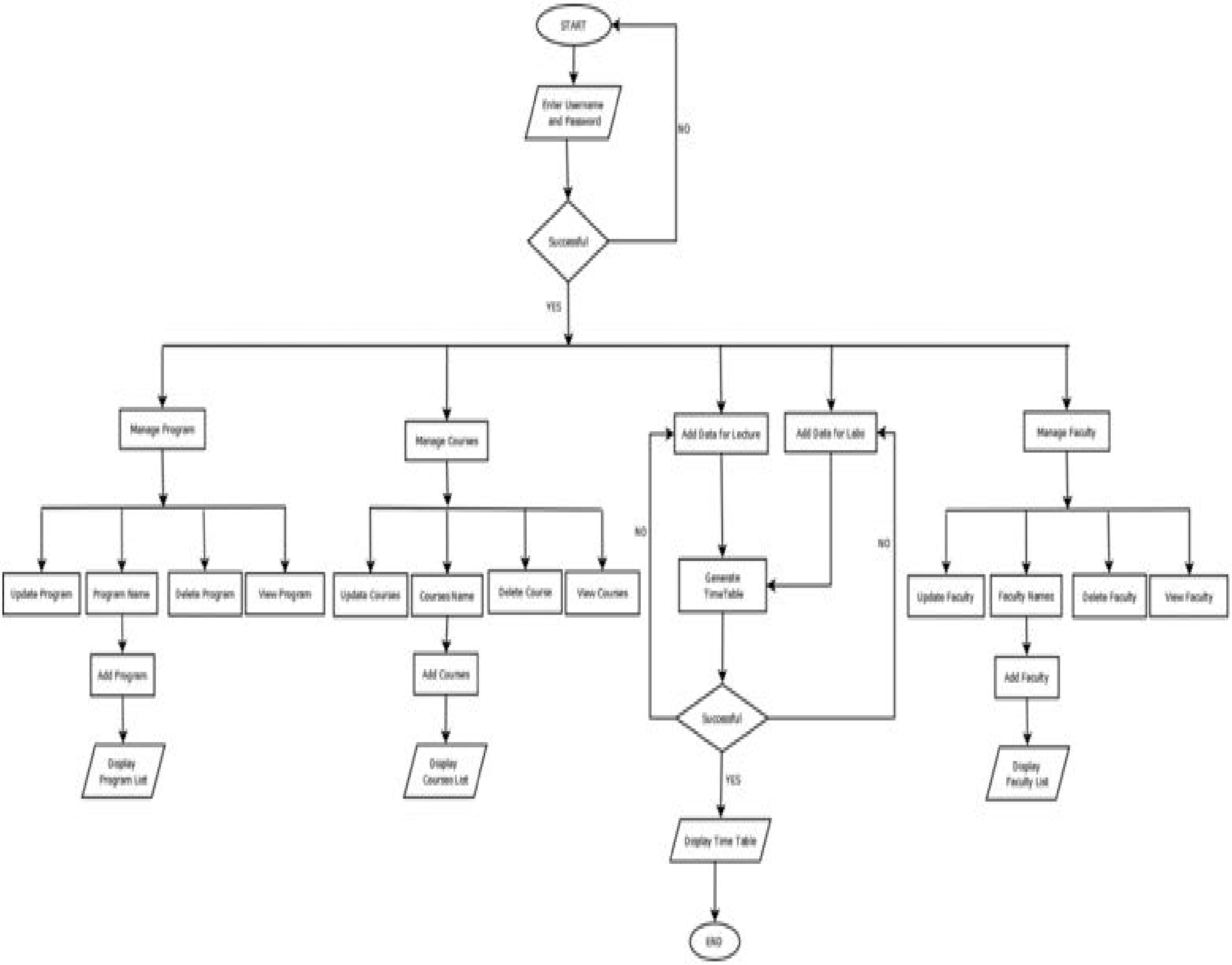


Fig. 5.1.1 Flow Chart of the Web App

5.2 Use Case Diagram

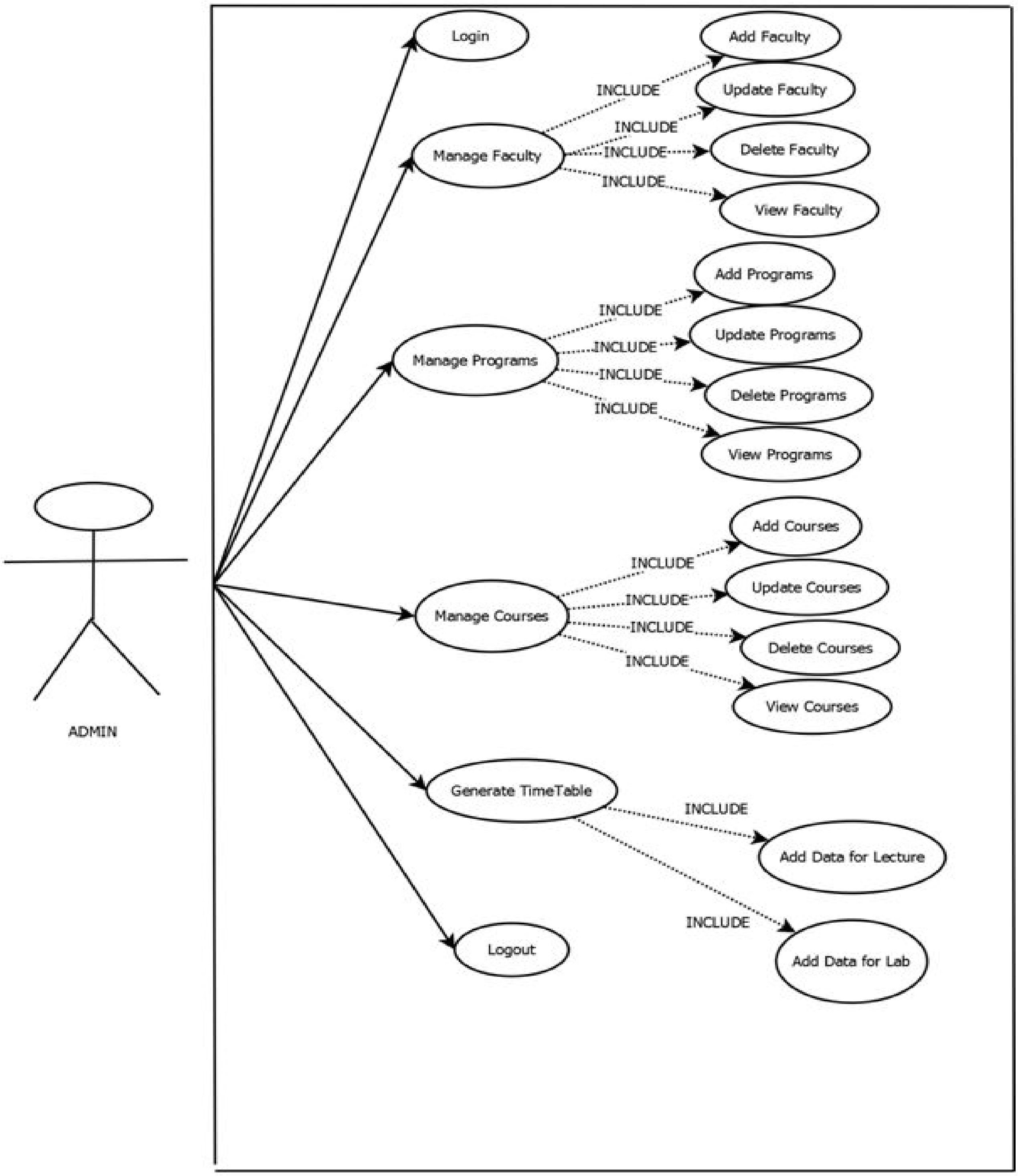


Fig 5.2.1 Use Case Diagram

5.3 Data Flow Diagram

5.3.1 Zero Level DFD

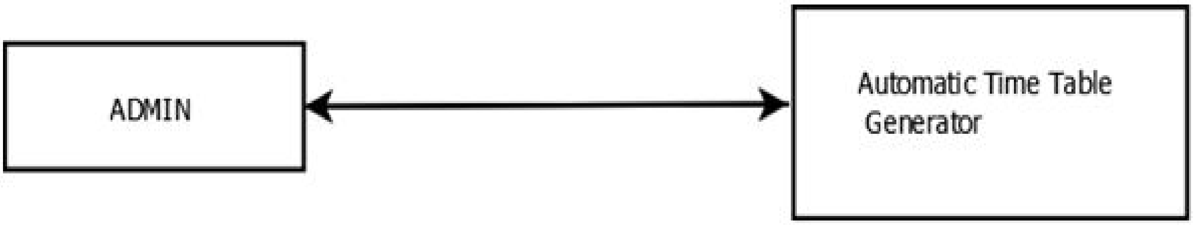


Fig 5.3.1 Zero Level DFD

5.3.2 Level 1 DFD

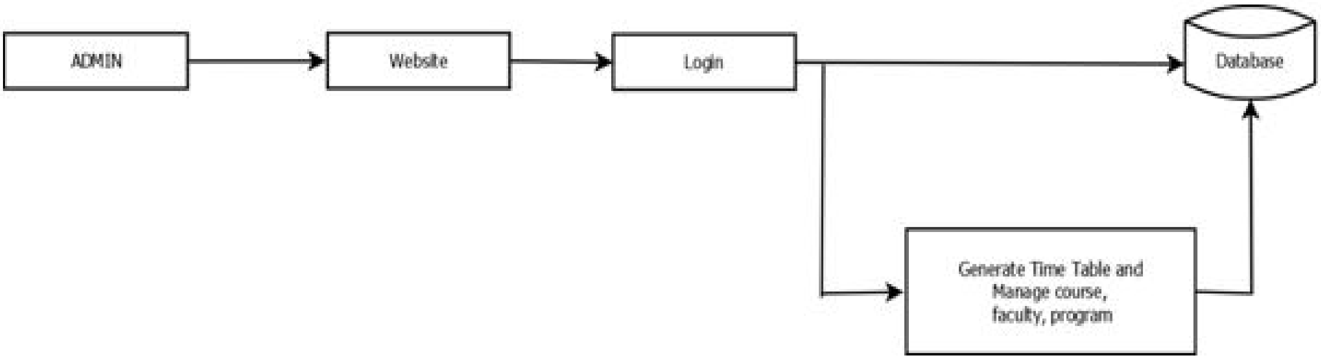


Fig 5.3.2 Level 1 DFD

5.3.3 Level 2 DFD

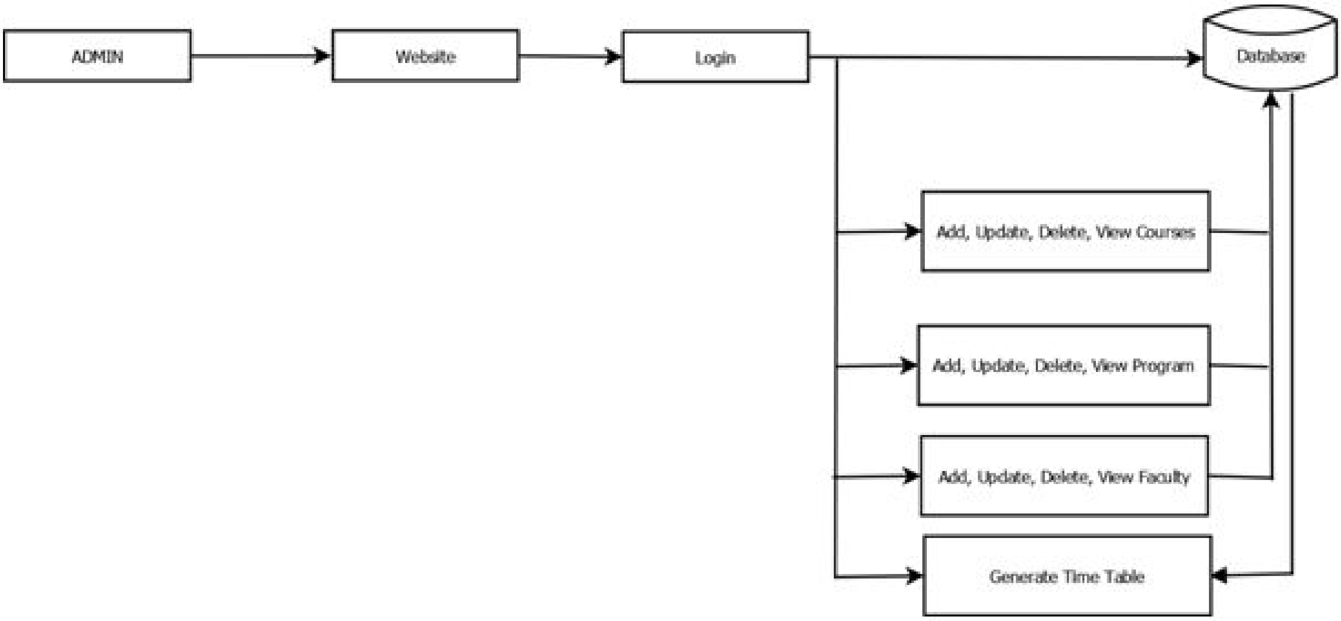

ADMIN
Website
Login
Database
Add, Update, Delete, View Courses
Add, Update, Delete, View Program
Add, Update, Delete, View Faculty
Generate Time Table

Fig 5.3.2 Level 2 DFD

5.4 ER Diagram

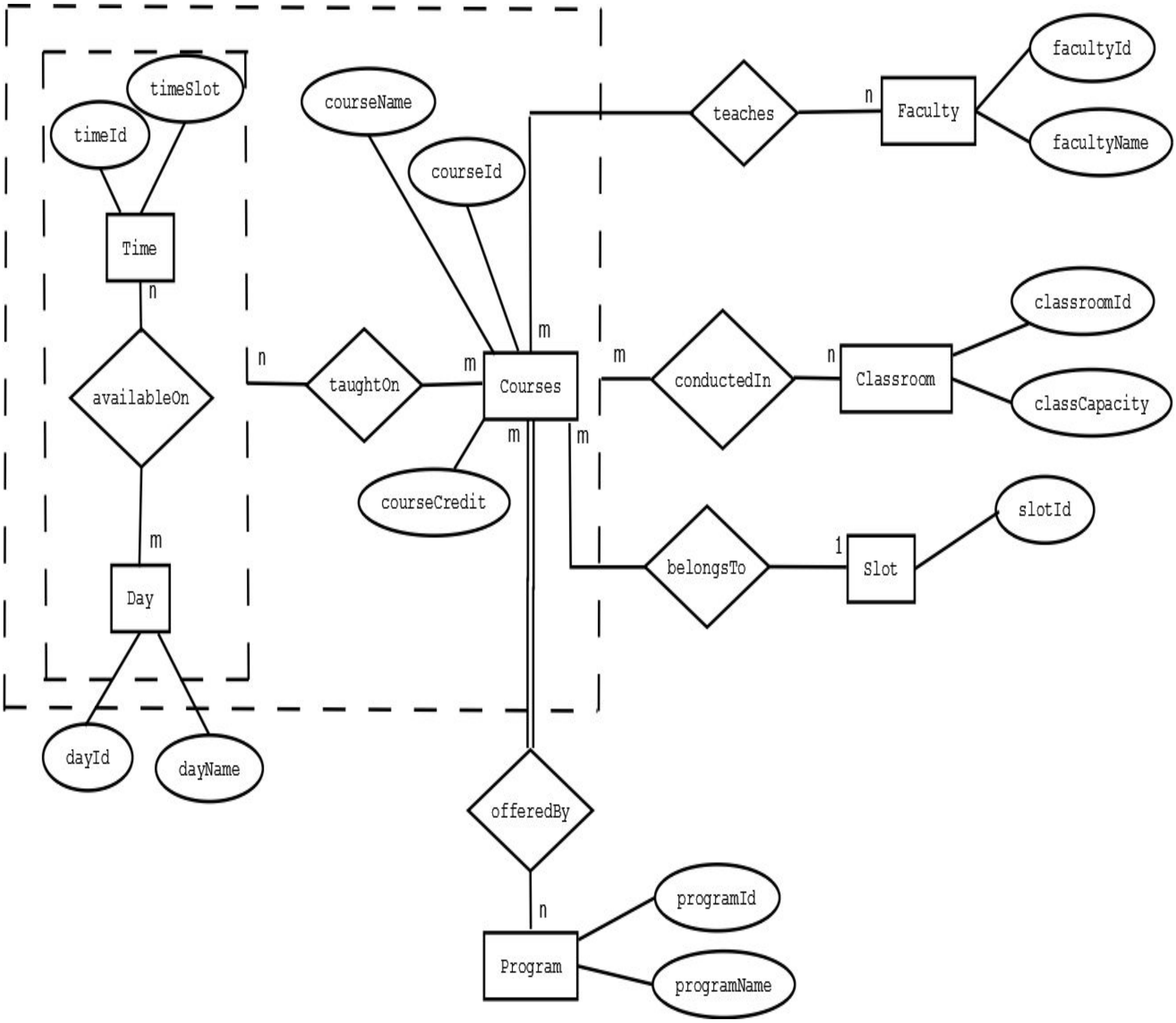


Fig 5.4.1 ER Diagram

6. Data Dictionary

6.1 Program

Attribute Name	Data Type	Field Size
i_programid	Int	11
c_programname	varchar	45

6.2 Course

Attribute Name	Data Type	Field Size
c_courseid	Varchar	10
c_coursename	Varchar	100
i_coursecredit	Int	11
i_programid	Int	11

6.3 Faculty

Attribute Name	Data Type	Field Size
c_facultyid	varchar	10
c_facultyname	varchar	45

6.4 Classroom

Attribute Name	Data Type	Field Size
c_classroomid	Varchar	10
i_classroomcapacity	Int	11

6.5 Admin

Attribute Name	Data Type	Field Size
c_username	Varchar	20
c_password	Varchar	10

6.6 Course_Faculty

Attribute Name	Data Type	Field Size
c_courseid	Varchar	10
c_facultyid	Varchar	10

6.7 Day

Attribute Name	Data Type	Field Size
i_dayid	Int	11
c_day	Varchar	10

6.8 Day_time_Classroom

Attribute Name	Data Type	Field Size
i_dayid	Int	11
i_timeid	Int	11
c_classroomid	Varchar	10
c_courseid	Varchar	10

6.9 Day_time_Course

Attribute Name	Data Type	Field Size
i_dayid	Int	11
i_timeid	Int	11
c_courseid	Varchar	10

6.10 Program_Classroom

Attribute Name	Data Type	Field Size
i_pcid	Int	11
i_programid	Int	11
c_classroomid	Varchar	10

6.11 Time Slot

Attribute Name	Data Type	Field Size
i_timeid	Int	11
i_timeslot	Int	11

6.12 Slot_course

Attribute Name	Data Type	Field Size
i_slotid	Int	11
c_courseid	Varchar	10

6.13 Slot

Attribute Name	Data Type	Field Size
i_slotid	Int	11

7. Implementation Planning

7.1 Implementation Environment

A fully-compatible environment is been chosen for the implementation of the system which will ensure a functionally consistent interface for the user with no dependency on the environment or the type of the system they use to access it.

The web app is built with the technologies that are widely supported by all the major desktop environment wherein Windows and Mac OS.

7.2 Use Case Module Specification

7.2.1 Login/Logout

Description: This use case allows the admin to login or logout to the system.

Actor: ADMIN

Input: Username and Password

Output: Depending on the validity of the username and password passed in the system, the admin will be logged into the system or rejected

7.2.2 Manage Faculty

7.2.2.1 Add faculty

Description: This use case allows the admin to add any new faculty (who joined the institute) in the system.

Actor: ADMIN

Input: Give faculty ID, name and type.

Output: Display newly updated list of faculties.

7.2.2.2 Update faculty

Description: This use case allows the admin to update any information of existing faculty in the system.

Actor: ADMIN

Input: Updated details of faculty.

Output: Display newly updated list of faculties.

7.2.2.3 Delete faculty

Description: This use case allows the admin to delete any existing faculty from the system.

Actor: ADMIN

Input: Give Faculty ID.

Output: Display newly updated list of faculties.

7.2.2.4 View faculty

Description: This use case allows the admin to view the faculty list from the system.

Actor: ADMIN

Input: No input required

Output: Display the list of faculties.

7.2.3 Manage Programs

7.2.3.1 Add Programs

Description: This use case allows the admin to add any new program in the system.

Actor: ADMIN

Input: Give Program ID and Program Name.

Output: Display newly updated list of Programs.

7.2.3.2 Update Programs

Description: This use case allows the admin to update any information of existing programs in the system.

Actor: ADMIN

Input: Updated details of Program.

Output: Display newly updated list of Programs.

7.2.3.3 Delete Programs

Description: This use case allows the admin to delete any existing Programs from the system.

Actor: ADMIN

Input: Give Program ID.

Output: Display newly updated list of Programs.

7.2.3.4 View Programs

Description: This use case allows the admin to view the Programs list from the system.

Actor: ADMIN

Input: No input required

Output: Display the list of programs.

7.2.4 Manage Courses

7.2.4.1 Add Courses

Description: This use case allows the admin to add any new course in the system.

Actor: ADMIN

Input: Give Course ID and Course name.

Output: Display newly updated list of courses.

7.2.4.2 Update Courses

Description: This use case allows the admin to update any information of existing courses in the system.

Actor: ADMIN

Input: Updated details of Courses.

Output: Display newly updated list of courses.

7.2.4.3 Delete Courses

Description: This use case allows the admin to delete any existing courses from the system.

Actor: ADMIN

Input: Give Course ID.

Output: Display newly updated list of courses.

7.2.4.4 View Courses

Description: This use case allows the admin to view the courses list from the system.

Actor: ADMIN

Input: No input required

Output: Display the list of courses.

7.2.5 Generate Time Table

Description: This use case allows the admin to generate time table for lecture or lab.

Actor: ADMIN

Input: Excel sheet of Student Registration Details and course faculty details.

Output: Display generated time table.

7.3 Data Coding Standards

A standard approach for the coding of the system has been adopted to achieve simplicity and maintainability in the development.

The Web-App is developed in PHP which will allow the data transactions in a unified and standard way that makes it easier for the developer to code and maintain the system in long run of time.

8. Snapshots

8.1 Login Page

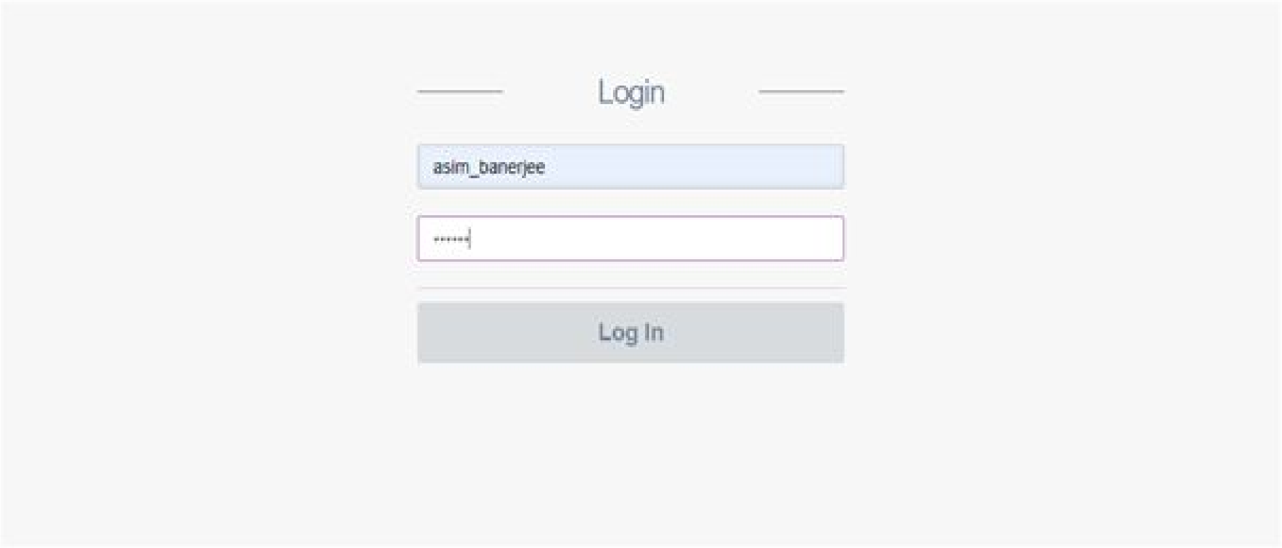
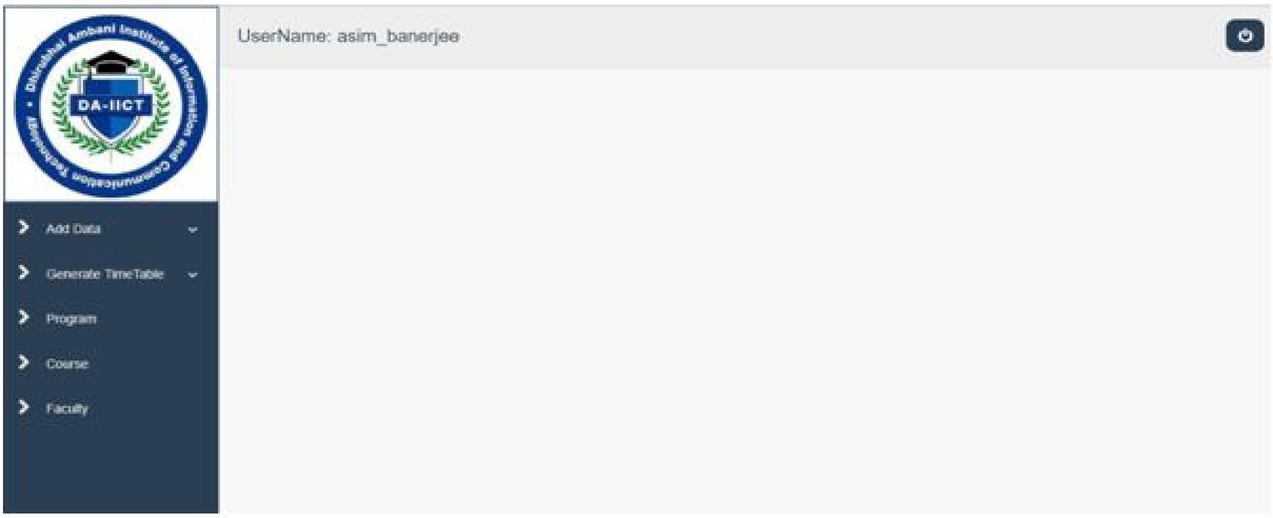


Fig 8.1.1 Login Page

8.2 Dashboard

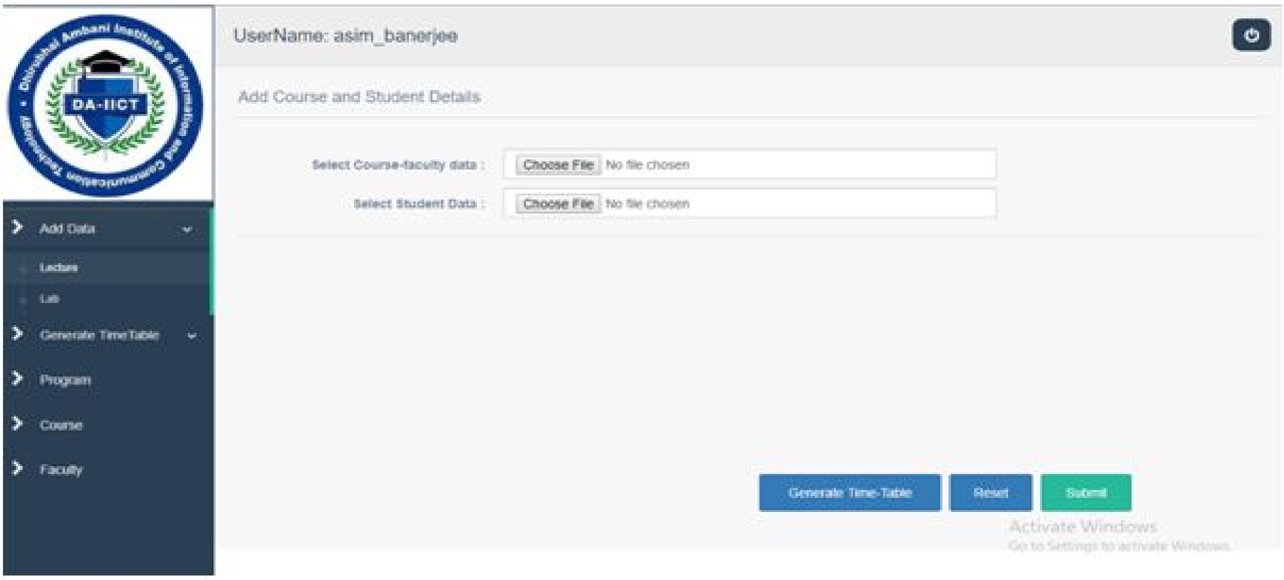


UserName: asim_banerjee

DA-IICT
Dhirubhai Ambani Institute of Information and Communication Technology
Add Data
Generate TimeTable
Program
Course
Faculty

Fig 8.2.1 Dashboard

8.3 Add Data -> Lecture



UserName: asim_banerjee

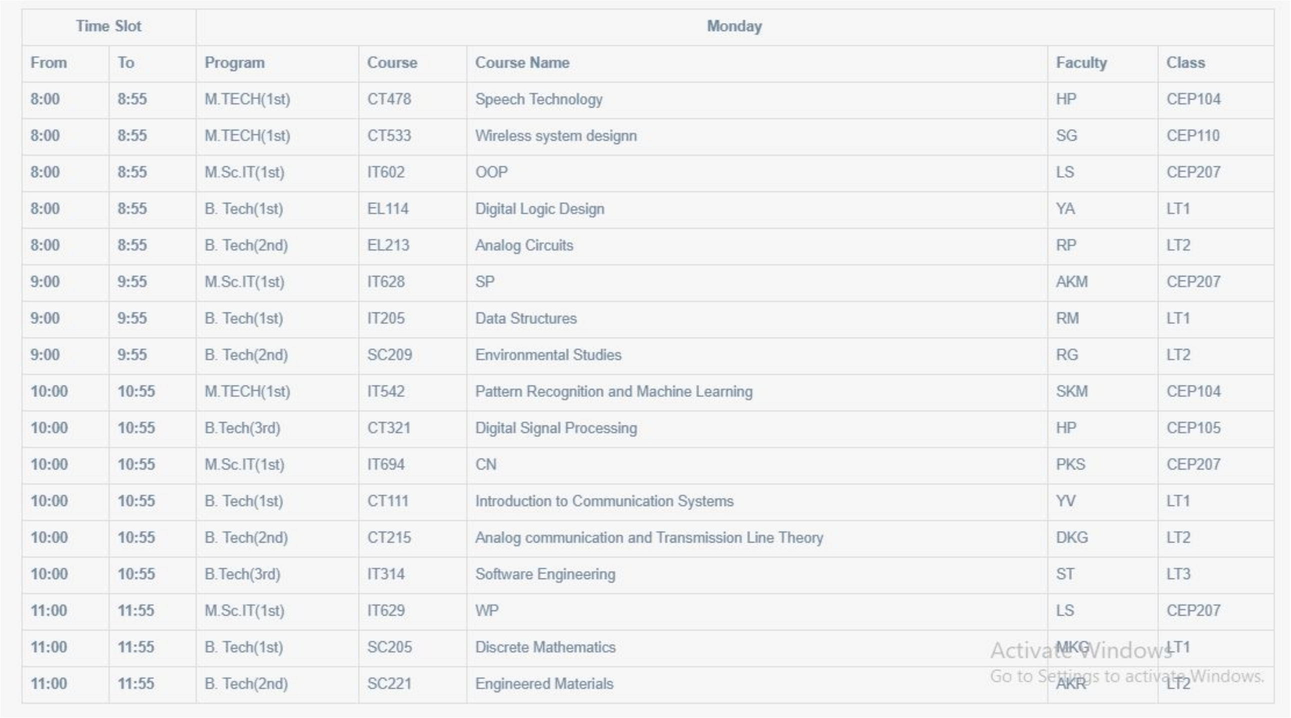
Add Course and Student Details

Select Course-faculty data : Choose File No file chosen

Select Student Data : Choose File No file chosen

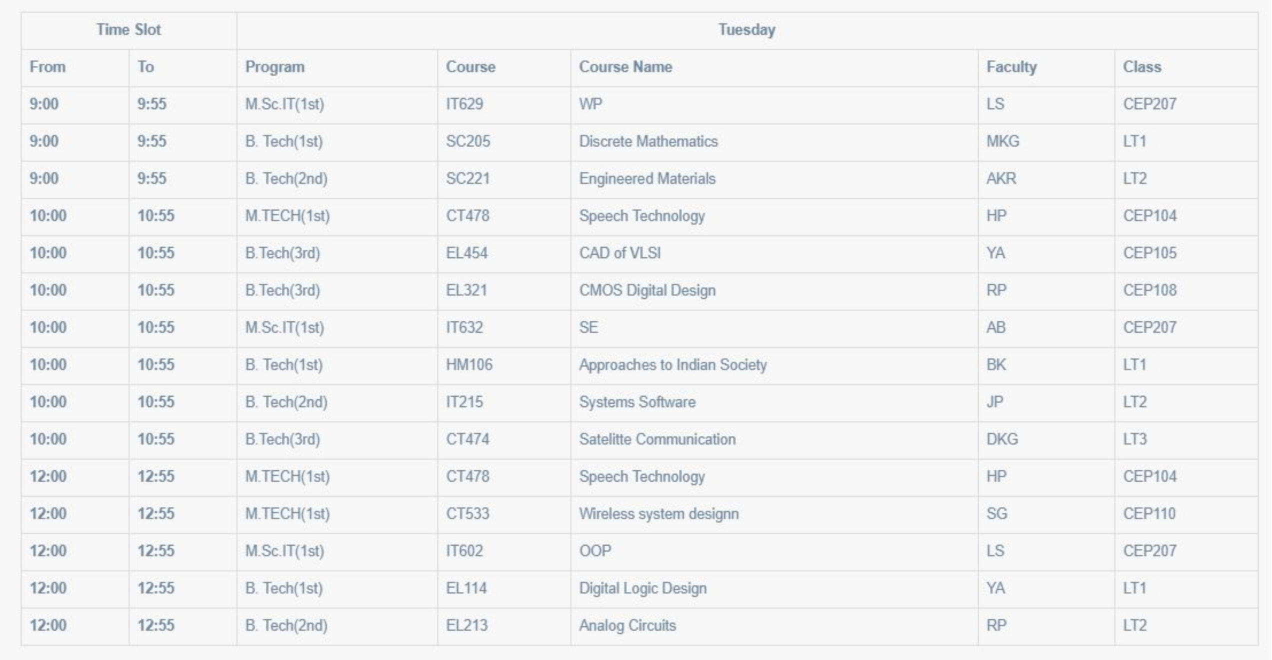
Fig 8.3.1 Add data to Lecture

8.4 After choosing required file and generating time table



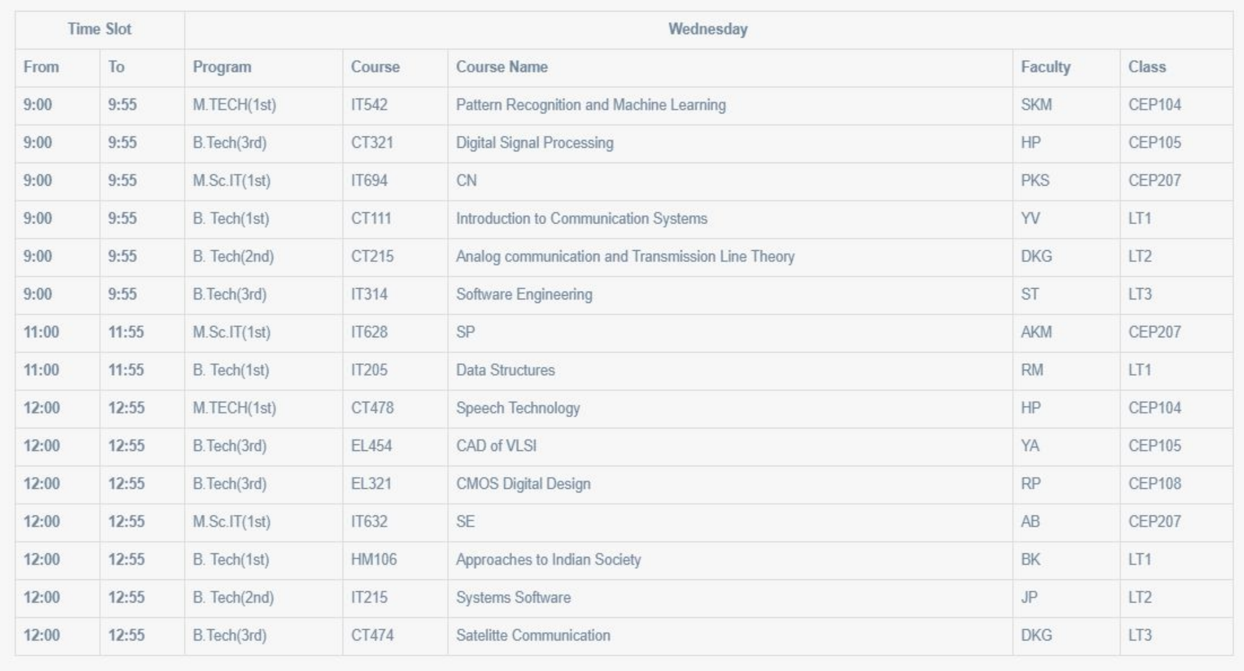
Time Slot		Monday				
From	To	Program	Course	Course Name	Faculty	Class
8:00	8:55	M.TECH(1st)	CT478	Speech Technology	HP	CEP104
8:00	8:55	M.TECH(1st)	CT533	Wireless system designn	SG	CEP110
8:00	8:55	M.Sc.IT(1st)	IT602	OOP	LS	CEP207
8:00	8:55	B. Tech(1st)	EL114	Digital Logic Design	YA	LT1
8:00	8:55	B. Tech(2nd)	EL213	Analog Circuits	RP	LT2
9:00	9:55	M.Sc.IT(1st)	IT628	SP	AKM	CEP207
9:00	9:55	B. Tech(1st)	IT205	Data Structures	RM	LT1
9:00	9:55	B. Tech(2nd)	SC209	Environmental Studies	RG	LT2
10:00	10:55	M.TECH(1st)	IT542	Pattern Recognition and Machine Learning	SKM	CEP104
10:00	10:55	B.Tech(3rd)	CT321	Digital Signal Processing	HP	CEP105
10:00	10:55	M.Sc.IT(1st)	IT694	CN	PKS	CEP207
10:00	10:55	B. Tech(1st)	CT111	Introduction to Communication Systems	YV	LT1
10:00	10:55	B. Tech(2nd)	CT215	Analog communication and Transmission Line Theory	DKG	LT2
10:00	10:55	B.Tech(3rd)	IT314	Software Engineering	ST	LT3
11:00	11:55	M.Sc.IT(1st)	IT629	WP	LS	CEP207
11:00	11:55	B. Tech(1st)	SC205	Discrete Mathematics	MKG	LT1
11:00	11:55	B. Tech(2nd)	SC221	Engineered Materials	AKR	LT2

Fig 8.4.1 After generating time table_Monday



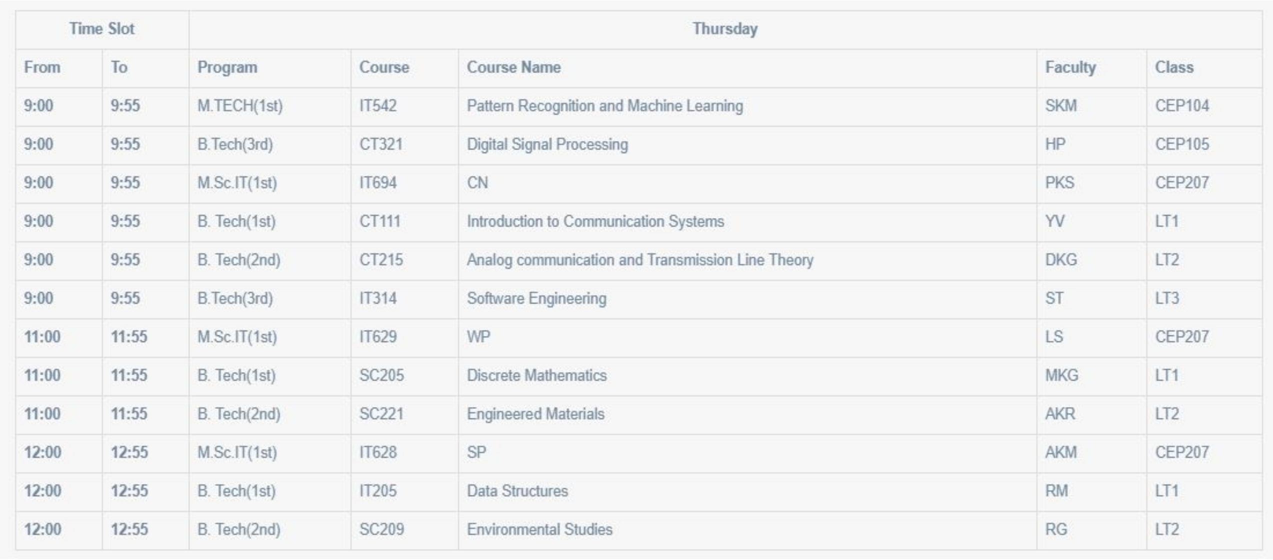
Time Slot		Tuesday				
From	To	Program	Course	Course Name	Faculty	Class
9:00	9:55	M.Sc.IT(1st)	IT629	WP	LS	CEP207
9:00	9:55	B. Tech(1st)	SC205	Discrete Mathematics	MKG	LT1
9:00	9:55	B. Tech(2nd)	SC221	Engineered Materials	AKR	LT2
10:00	10:55	M.TECH(1st)	CT478	Speech Technology	HP	CEP104
10:00	10:55	B.Tech(3rd)	EL454	CAD of VLSI	YA	CEP105
10:00	10:55	B.Tech(3rd)	EL321	CMOS Digital Design	RP	CEP108
10:00	10:55	M.Sc.IT(1st)	IT632	SE	AB	CEP207
10:00	10:55	B. Tech(1st)	HM106	Approaches to Indian Society	BK	LT1
10:00	10:55	B. Tech(2nd)	IT215	Systems Software	JP	LT2
10:00	10:55	B.Tech(3rd)	CT474	Satelitte Communication	DKG	LT3
12:00	12:55	M.TECH(1st)	CT478	Speech Technology	HP	CEP104
12:00	12:55	M.TECH(1st)	CT533	Wireless system designn	SG	CEP110
12:00	12:55	M.Sc.IT(1st)	IT602	OOP	LS	CEP207
12:00	12:55	B. Tech(1st)	EL114	Digital Logic Design	YA	LT1
12:00	12:55	B. Tech(2nd)	EL213	Analog Circuits	RP	LT2

Fig 8.4.2 After generating time table_Tuesday



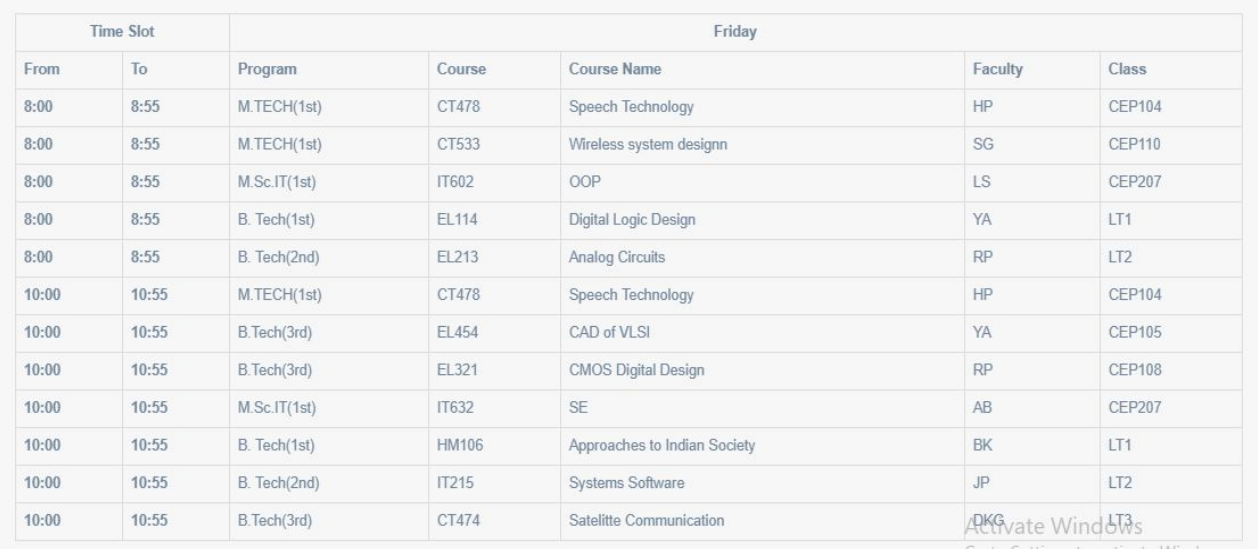
Time Slot		Wednesday				
From	To	Program	Course	Course Name	Faculty	Class
9:00	9:55	M.TECH(1st)	IT542	Pattern Recognition and Machine Learning	SKM	CEP104
9:00	9:55	B.Tech(3rd)	CT321	Digital Signal Processing	HP	CEP105
9:00	9:55	M.Sc.IT(1st)	IT694	CN	PKS	CEP207
9:00	9:55	B. Tech(1st)	CT111	Introduction to Communication Systems	YV	LT1
9:00	9:55	B. Tech(2nd)	CT215	Analog communication and Transmission Line Theory	DKG	LT2
9:00	9:55	B.Tech(3rd)	IT314	Software Engineering	ST	LT3
11:00	11:55	M.Sc.IT(1st)	IT628	SP	AKM	CEP207
11:00	11:55	B. Tech(1st)	IT205	Data Structures	RM	LT1
12:00	12:55	M.TECH(1st)	CT478	Speech Technology	HP	CEP104
12:00	12:55	B.Tech(3rd)	EL454	CAD of VLSI	YA	CEP105
12:00	12:55	B.Tech(3rd)	EL321	CMOS Digital Design	RP	CEP108
12:00	12:55	M.Sc.IT(1st)	IT632	SE	AB	CEP207
12:00	12:55	B. Tech(1st)	HM106	Approaches to Indian Society	BK	LT1
12:00	12:55	B. Tech(2nd)	IT215	Systems Software	JP	LT2
12:00	12:55	B.Tech(3rd)	CT474	Satelitte Communication	DKG	LT3

Fig 8.4.3 After generating time table_Wednesday



Time Slot		Thursday				
From	To	Program	Course	Course Name	Faculty	Class
9:00	9:55	M.TECH(1st)	IT542	Pattern Recognition and Machine Learning	SKM	CEP104
9:00	9:55	B.Tech(3rd)	CT321	Digital Signal Processing	HP	CEP105
9:00	9:55	M.Sc.IT(1st)	IT694	CN	PKS	CEP207
9:00	9:55	B. Tech(1st)	CT111	Introduction to Communication Systems	YV	LT1
9:00	9:55	B. Tech(2nd)	CT215	Analog communication and Transmission Line Theory	DKG	LT2
9:00	9:55	B.Tech(3rd)	IT314	Software Engineering	ST	LT3
11:00	11:55	M.Sc.IT(1st)	IT629	WP	LS	CEP207
11:00	11:55	B. Tech(1st)	SC205	Discrete Mathematics	MKG	LT1
11:00	11:55	B. Tech(2nd)	SC221	Engineered Materials	AKR	LT2
12:00	12:55	M.Sc.IT(1st)	IT628	SP	AKM	CEP207
12:00	12:55	B. Tech(1st)	IT205	Data Structures	RM	LT1
12:00	12:55	B. Tech(2nd)	SC209	Environmental Studies	RG	LT2

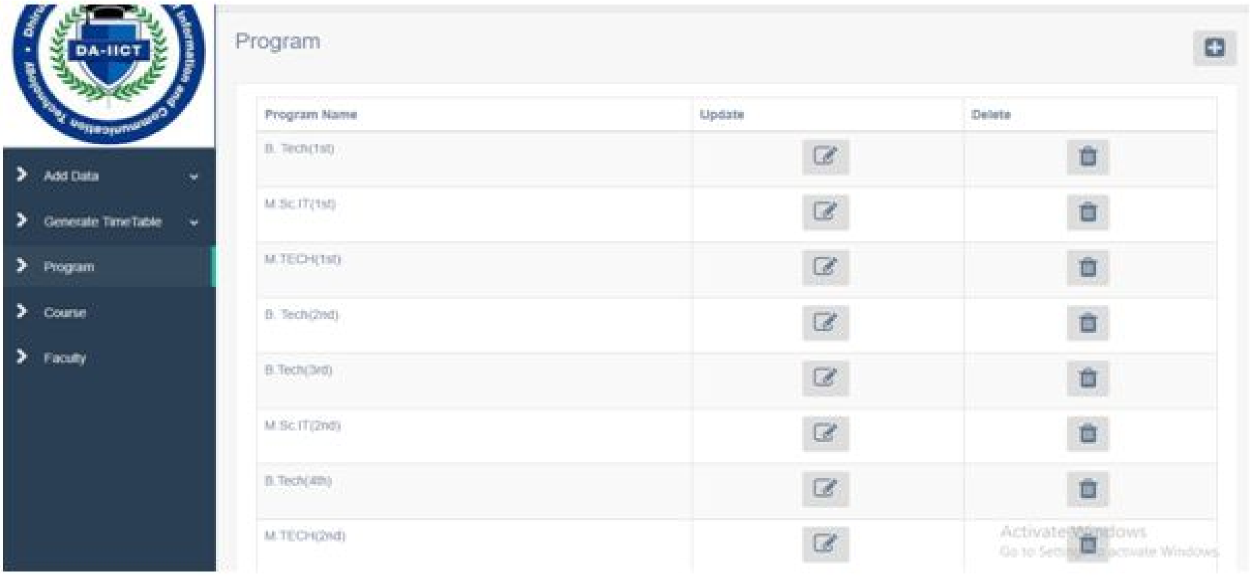
Fig 8.4.4 After generating time table_Thursday



Time Slot		Friday				
From	To	Program	Course	Course Name	Faculty	Class
8:00	8:55	M.TECH(1st)	CT478	Speech Technology	HP	CEP104
8:00	8:55	M.TECH(1st)	CT533	Wireless system designn	SG	CEP110
8:00	8:55	M.Sc.IT(1st)	IT602	OOP	LS	CEP207
8:00	8:55	B. Tech(1st)	EL114	Digital Logic Design	YA	LT1
8:00	8:55	B. Tech(2nd)	EL213	Analog Circuits	RP	LT2
10:00	10:55	M.TECH(1st)	CT478	Speech Technology	HP	CEP104
10:00	10:55	B.Tech(3rd)	EL454	CAD of VLSI	YA	CEP105
10:00	10:55	B.Tech(3rd)	EL321	CMOS Digital Design	RP	CEP108
10:00	10:55	M.Sc.IT(1st)	IT632	SE	AB	CEP207
10:00	10:55	B. Tech(1st)	HM106	Approaches to Indian Society	BK	LT1
10:00	10:55	B. Tech(2nd)	IT215	Systems Software	JP	LT2
10:00	10:55	B.Tech(3rd)	CT474	Satelitte Communication	DKG	LT3

Fig 8.4.5 After generating time table_Friday

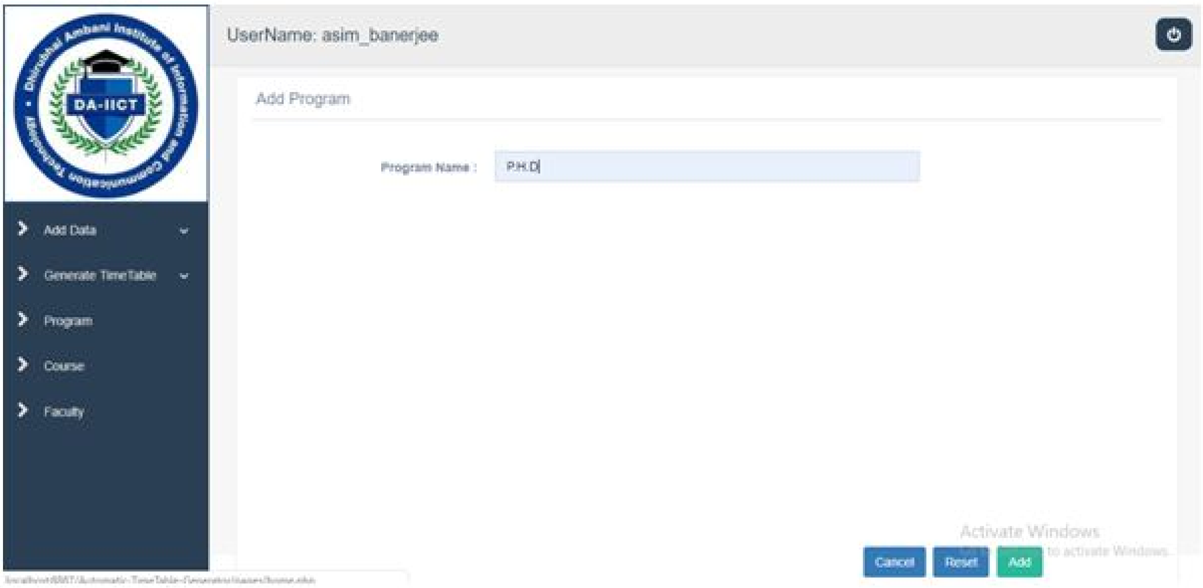
8.5. View, Add, Update and Delete Program



Program

Program Name	Update	Delete
B. Tech(1st)		
M.Sc.IT(1st)		
M.TECH(1st)		
B. Tech(2nd)		
B.Tech(3rd)		
M.Sc.IT(2nd)		
B.Tech(4th)		
M.TECH(2nd)		

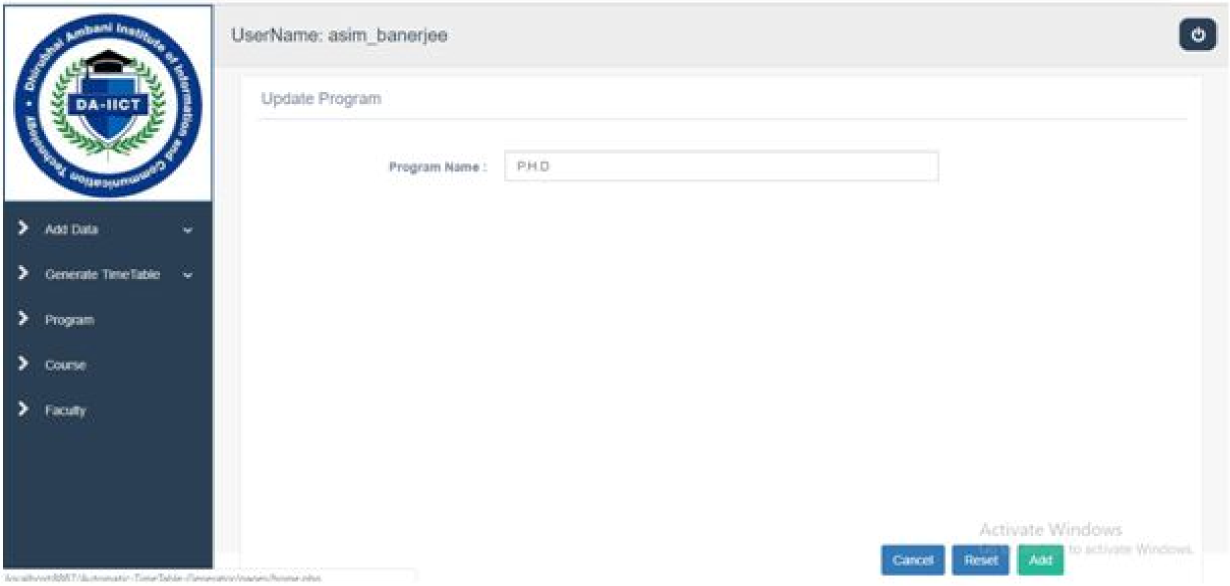
Fig 8.5.1 Add,Update,Delete,View Program


UserName: asim_banerjee

Add Program

Program Name : P.H.D

Fig 8.5.2 Add Program

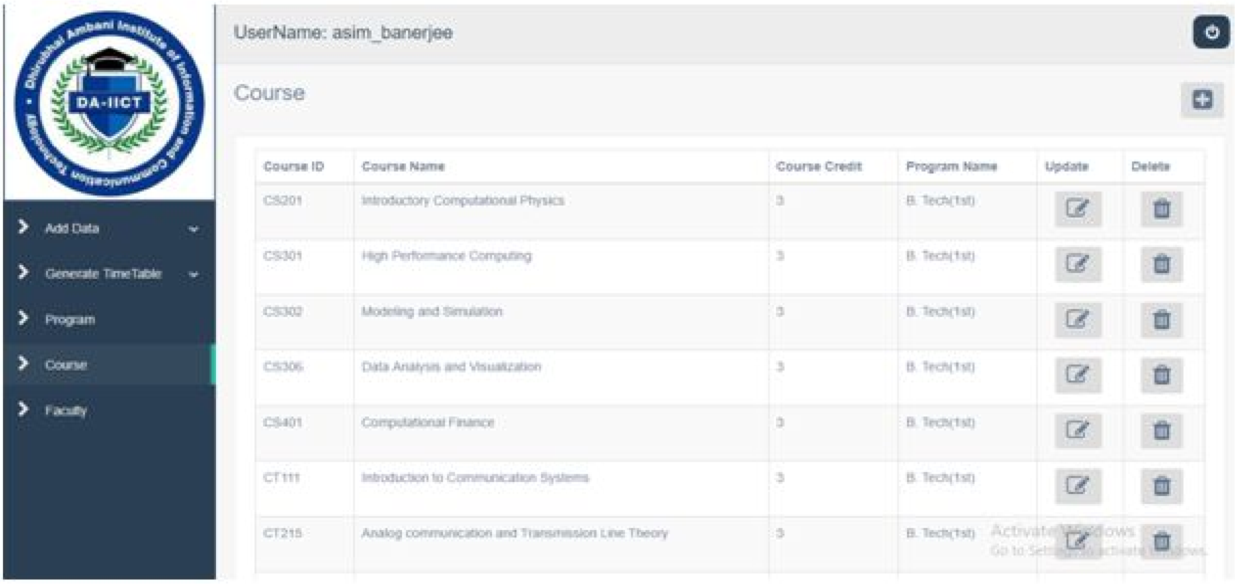

UserName: asim_banerjee

Update Program

Program Name : P.H.D

Fig 8.5.3 Update Program

8.6 View, add, update and delete a course

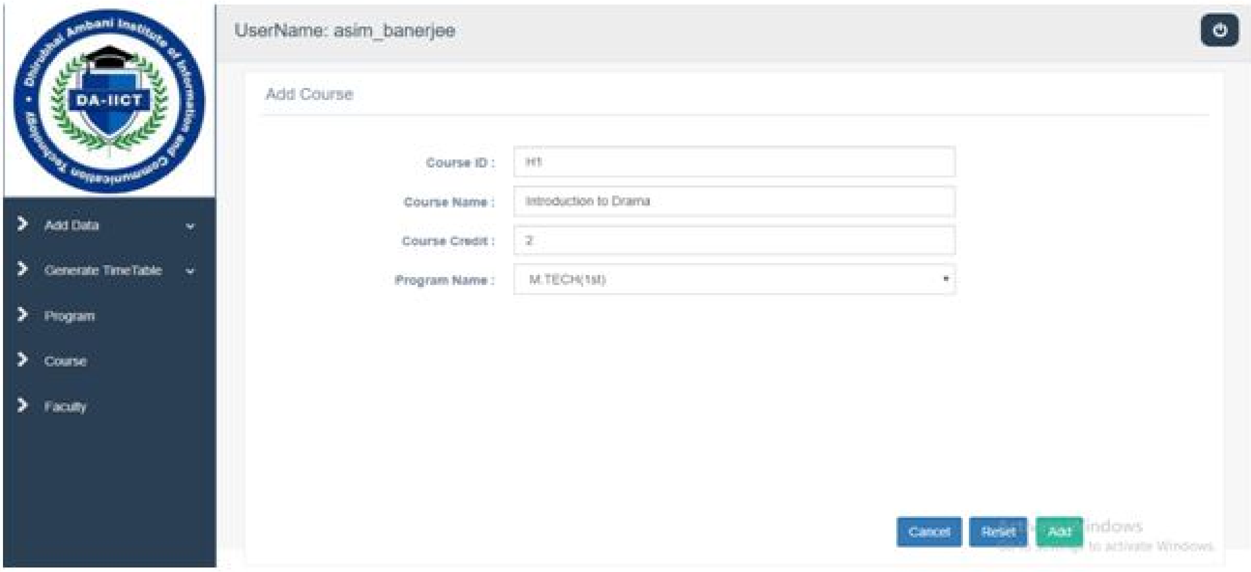


UserName: asim_banerjee

Course

Course ID	Course Name	Course Credit	Program Name	Update	Delete
CS201	Introductory Computational Physics	3	B. Tech(1st)		
CS301	High Performance Computing	3	B. Tech(1st)		
CS302	Modeling and Simulation	3	B. Tech(1st)		
CS306	Data Analysis and Visualization	3	B. Tech(1st)		
CS401	Computational Finance	3	B. Tech(1st)		
CT111	Introduction to Communication Systems	3	B. Tech(1st)		
CT215	Analog communication and Transmission Line Theory	3	B. Tech(1st)		

Fig 8.6.1 Add,Update,Delete,View Course



UserName: asim_banerjee

Add Course

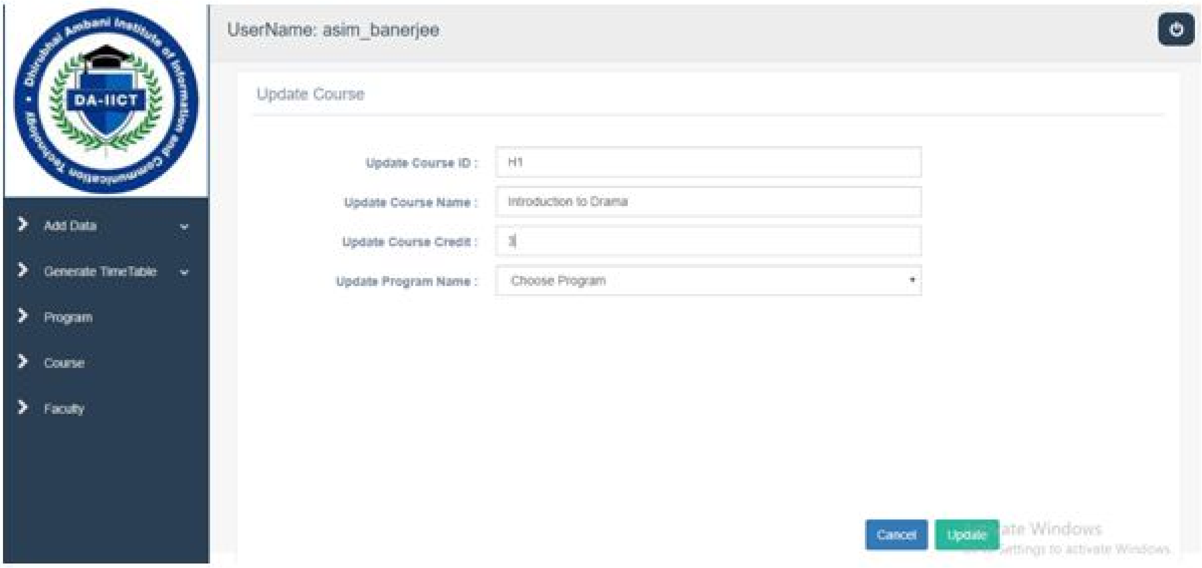
Course ID : H1

Course Name : Introduction to Drama

Course Credit : 2

Program Name : M.TECH(1st)

Fig 8.6.2 Add Course



Update Course

Update Course ID : H1

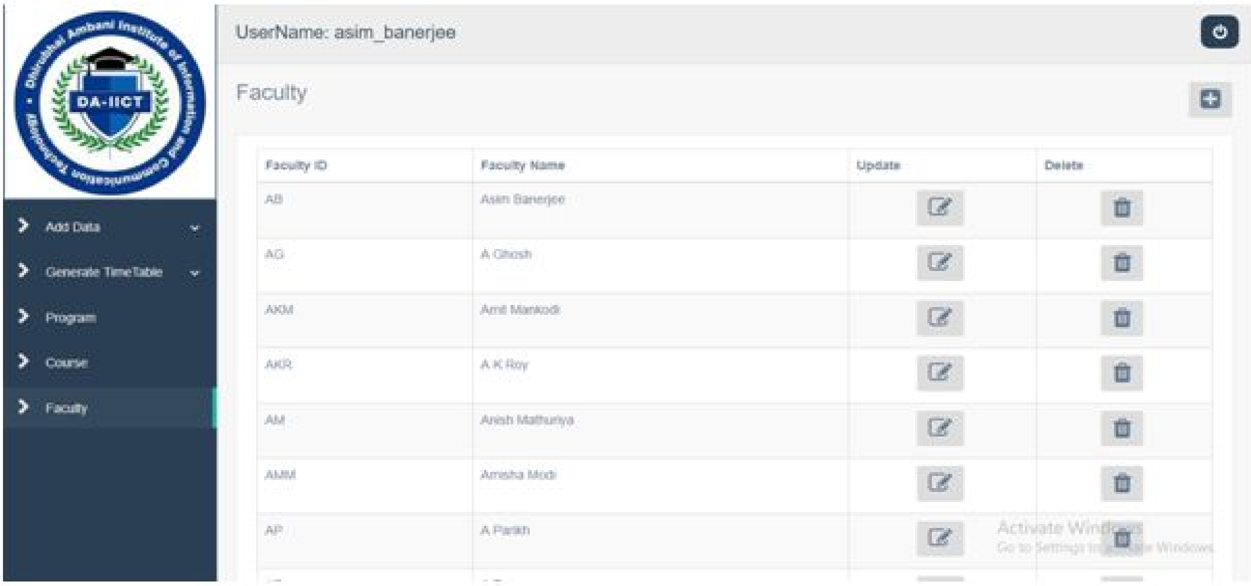
Update Course Name : Introduction to Drama

Update Course Credit : 3

Update Program Name : Choose Program

Fig 8.6.2 Update Course

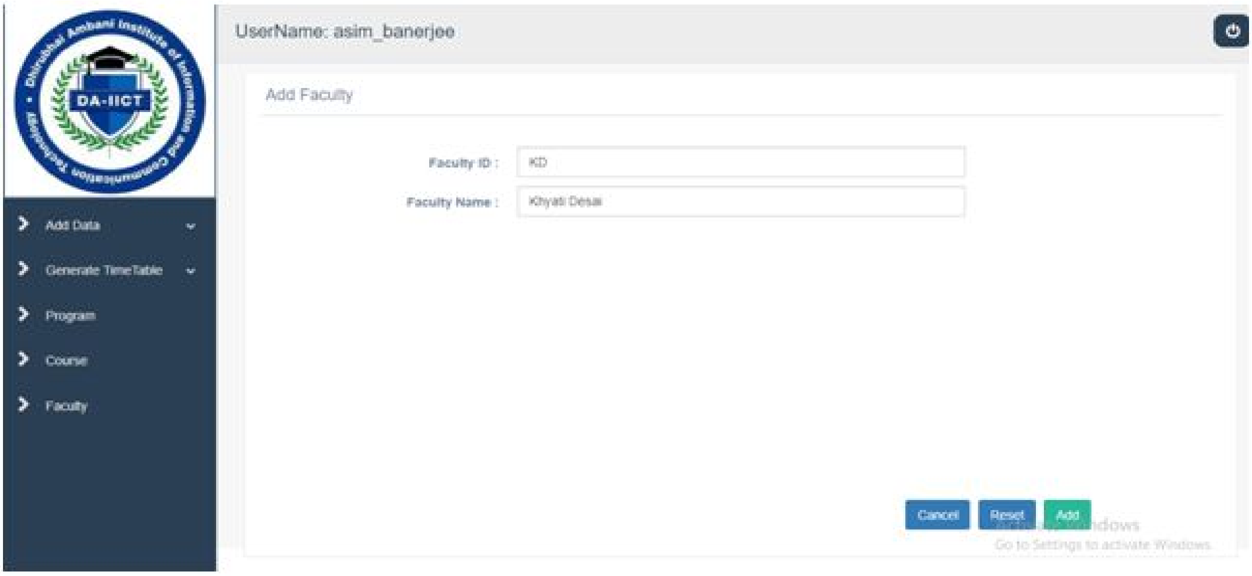
8.7. View, add, update and delete faculty



Faculty

Faculty ID	Faculty Name	Update	Delete
AB	Asim Banerjee		
AG	A Ghosh		
AKM	Amit Mankodi		
AKR	A K Roy		
AM	Anish Mathuriya		
AMM	Amisha Modi		
AP	A Parikh		

Fig 8.7.1 Add,Update,Delete,View Faculty



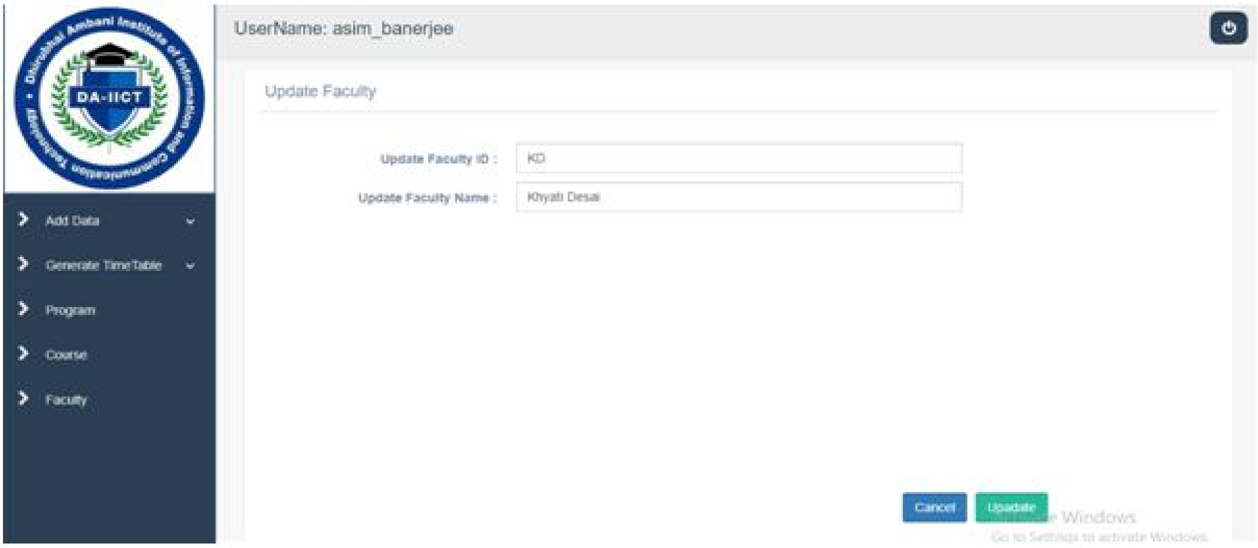
UserName: asim_banerjee

Add Faculty

Faculty ID : KD

Faculty Name : Khyati Desai

Fig 8.7.2 Add Faculty



UserName: asim_banerjee

Update Faculty

Update Faculty ID : KD

Update Faculty Name : Khyati Desai

Fig 8.7.3 Update Faculty

(Snapshot link)

9. Testing

9.1 Testing Plan

9.1.1 Introduction

This document is a test plan for the Automatic Lecture Time Table Generation System produced by the Project team. It describes the testing strategy and approach to testing the team will use to verify that the application meets the established requirements of the software.

9.1.2 Objective

- Meets the requirements and specifications of the software.
- Supports the intended software functions and achieves the required software standards.

9.1.3 Functional Scope

The modules in the scope of testing for the Automatic Lecture Time Table Generation System Testing are mentioned in the document attached in the following path:

Functional Requirement	**Expected Output**
Admin login with id and password	Should be redirected to dashboard successfully
Admin selects excel document which contains data regarding courses and student ids of those who have registered in those courses.	Should be able to select the appropriate excel document

Admin generate time table for lecture	Should generate the lecture time table from given data in excel document successfully
Admin generate time table for lab	Should generate the lab time table from given data in excel document successfully
Admin retrieve program information	Should be able to see all the programs conducted in institute
Admin add program	Should be able to add new programs in database
Admin update a program by giving ID	Should be able to update program detail of given ID in the database.
Admin delete program by giving ID	Should be able to remove the program with given ID from the database.
Admin retrieve course information	Should be able to see all the courses conducted in institute
Admin add course	Should be able to add new courses in database
Admin update a course by giving ID	Should be able to update course detail of given ID in the database.

Admin delete course by giving ID	Should be able to remove the course with given ID from the database.
Admin retrieve faculty information	Should be able to see all the faculties conducted in institute
Admin add faculty	Should be able to add new faculties in database
Admin update a faculty by giving ID	Should be able to update faculty detail of given ID in the database.
Admin delete faculty by giving ID	Should be able to remove the faculty with given ID from the database.

9.1.4 Item Pass/Fail Criteria

9.1.4.1 Item Pass Criteria

Functional Requirements	**Pass Criteria**
Admin login with id and password	Admin needs to enter correct id and password
Admin selects excel document which contains data regarding courses and student ids of those who have	Admin should select correct file containing data for generating time table.

registered in those courses.	
Admin retrieve program information	Must be redirected to Program page displaying list of programs
Admin add program	Admin should enter the correct data in input fields
Admin update a program by giving ID	Only existing selected record should be updated
Admin delete program by giving ID	Selected recorded must be removed from database
Admin retrieve course information	Must be redirected to Course page displaying list of courses
Admin add course	Must enter the proper details in input fields
Admin update a course by giving ID	Selected recorded must be updated in database
Admin delete course by giving ID	Selected entry must be removed from database
Admin retrieve faculty information	Must be redirected to Faculty page displaying list of faculties
Admin add faculty	Should enter proper details in input fields
Admin update a faculty by giving ID	Only selected entry gets changed in database

Admin delete faculty by giving ID	Only selected entry gets removed from database

9.1.4.2 Item Fail Criteria

Functional Requirements	Fail Criteria
Admin login with id and password	Any of the two are incorrect than login fails.
Admin selects excel document which contains data regarding courses and student ids of those who have registered in those courses.	If wrong file has been selected it will not be able to generate time table.
Admin retrieve program information	Program page does not loads displaying list of courses.
Admin add program	Proper data is not entered in input fields
Admin update a program by giving ID	Program to be updated is not present in database
Admin delete program by giving ID	Record is not present in database
Admin retrieve course information	Course page does not loads displaying list of courses.
Admin add course	Proper data is not entered in input fields

Admin update a course by giving ID	Course to be updated is not present in database
Admin delete course by giving ID	Record is not present in database
Admin retrieve faculty information	Faculty page does not loads displaying list of faculties.
Admin add faculty	Proper required data is not entered in required fields
Admin update a faculty by giving ID	Selected record to be updated is not present in database
Admin delete faculty by giving ID	Record to be deleted is not present in database

9.1.5 Environmental Needs

Sr. No.	Environment
1.	User should have any device which can access browser and should match required browser version.
2.	User should have internet connection.

9.2 Testing Methodology and Test Cases

9.2.1 Reactive test Strategy:

In this test strategy testing will be done after designing and coding has been completed.

Test Cases:

Functional Requirement	Message	Pass/Fail
Admin login with id and password	Displays Dashboard	Pass
Admin enters ID starting with digit	ID cannot start with a digit	Pass
Admin enters ID ending with ./@	ID cannot end with dot/@	Pass
Admin enters wrong ID/Password	Invalid Username or Password	Pass
Admin enters only ID	Enter valid password	Pass
Admin enters only password	Enter valid ID	Pass
Admin selects excel document which contains data regarding courses, student ids etc	Generates Time Table	Pass
Admin selects wrong file	Please choose excel file only	Pass

Admin clicks Generate Time Table without selecting any file	Please select the data file	Pass
Admin clicks Generate Time Table Tab	Generates Time Table	Pass
Admin clicks reset button	Clears all the data from input fields	Pass
Admin clicks Program tab	Displays list of all the programs in institute.	Pass
Admin does not enters program ID	Please enter valid Program ID	Pass
Admin does not enters program name	Please enter valid Program Name	Pass
Admin fills all the fields properly	Record successfully inserted	Pass
Admin clicks Add button without filling mandatory fields	Displays message of filling that required field	Pass
Admin enters proper data in input fields for update	Displays updated list	Pass
Admin clicks Upload button without filling mandatory fields	Please, fill the required fields	Pass

Admin clicks Delete button	Successfully deletes the record and displays the updated list	Pass
Admin clicks Course tab	Displays list of all the courses in all the programs	Pass
Admin does not enters ID	Enter valid Course ID	Pass
Admin does not enters name	Enter valid Course Name	Pass
Admin does not enters Course Credit	Enter valid Course Credit	Pass
Admin does not selects program	Please Select a Program	Pass
Admin fills all the fields	Record successfully inserted	Pass
Admin clicks Add button without filling any mandatory field	Displays message of filling that required field	Pass
Admin enters proper data in input fields for update	Displays updated list	Pass
Admin clicks Upload button without filling mandatory fields	Displays message of filling that required field	Pass

Admin clicks Delete button	Successfully deletes the record and displays the updated list	Pass
Admin clicks the Faculty tab	Displays list of all the faculties in the institute	Pass
Admin does not enters Faculty ID	Enter valid faculty ID	Pass
Admin does not enters Faculty name	Enter valid faculty name	Pass
Admin fills all the fields	Record successfully inserted	Pass
Admin clicks Add button without filling any mandatory field	Displays message of filling that required field	Pass
Admin enters proper data in input fields for update	Recorded successfully updated	Pass
Admin clicks Upload button without filling any mandatory field	Displays message of filling that required field	Pass
Admin clicks Delete button	Successfully deletes the record and displays updated list	Pass
Admin clicks the Classroom tab	Displays list of classrooms in cep block	Pass

Admin does not enters classroom ID	Enter valid classroom ID	Pass
Admin fills all the fields	Record successfully inserted	Pass
Admin clicks Add button without filling any mandatory field	Displays message of filling that required field	Pass
Admin enters proper data in input fields for update	Recorded successfully updated	Pass
Admin clicks Upload button without filling any mandatory field	Displays message of filling that required field	Pass
Admin clicks Delete button	Successfully deletes the record and displays updated list	Pass

10. Conclusions

10.1 Conclusions

This time table system is a companion for our institution to improve their time table generating system with an automated system. Additional features include faculty management, program management and course management and will greatly help to manage time slots efficiently.

10.2 Limitations and Future Scope

Various features in the Automatic Time Table Generating System can be worked upon to achieve greater efficiency and rigidness in use and functioning of the system.

Few of the features that could be worked upon in future to improve Automatic Time Table Generating System are:

- Algorithm can be improved to generate time table in lesser amount of time with greater efficiency.
- Functionality for generating time table for lab could also be added later.