Software Requirements Specification

For

Automatic Time Table Generator

Prepared by TEAM-6

Table of Contents

1. INTRODUCTION

1.1 Purpose

The main purpose of this document is to list all the requirements of the AUTOMATIC TIME TABLE GENERATOR project. It also helps us to collect and analyze the ideas gathered for this project. It will explain the purpose and features of the software, the interfaces of the software and what the software will do. This document is intended for users of the software and it also set stage for the design phase of the project.

1.2 Project Overview

An Automatic time table generator is a Web-based Application Software that will help to generate time table for our institute automatically. It takes an excel sheet containing data like courses, students enrolled in that particular course, etc. 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.

1.3 References

Some of the references used for preparing this document include:

1. DIA software is used to generate Flow diagram, Use case diagrams.
2. IEEE template for Software Requirements Specifications.
3. Software Engineering by Roger. S. Pressman.

2. OVERALL PRODUCT DESCRIPTION

2.1 Product Perspective

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 lectures and labs. This web app is developed in PHP which will work with the database developed in MYSQL.

2.2 Product Features

The Automatic Time Table Generator has a process model along which all the processes are handled for the admin. The first step, we have the student registration details and course-faculty details excel format so that when the system requires it, we provide it.

Now the process is followed by a sequence of steps:

1. Admin logs into the system with the username and password.
2. If successful login, he/she can manage faculties, courses and programs.
   If admin wants to generate the lecture timetable, he/she has to provide data for lecture.
   If admin wants to generate the lab timetable, he/she has to provide data for lab. (Here, data means student registration details and course-faculty details)
3. Now, Admin has to click on Generate Time Table (for lecture or lab) and view the timetable.

The flowchart diagram (Fig 2.1) below explains the system diagrammatically.

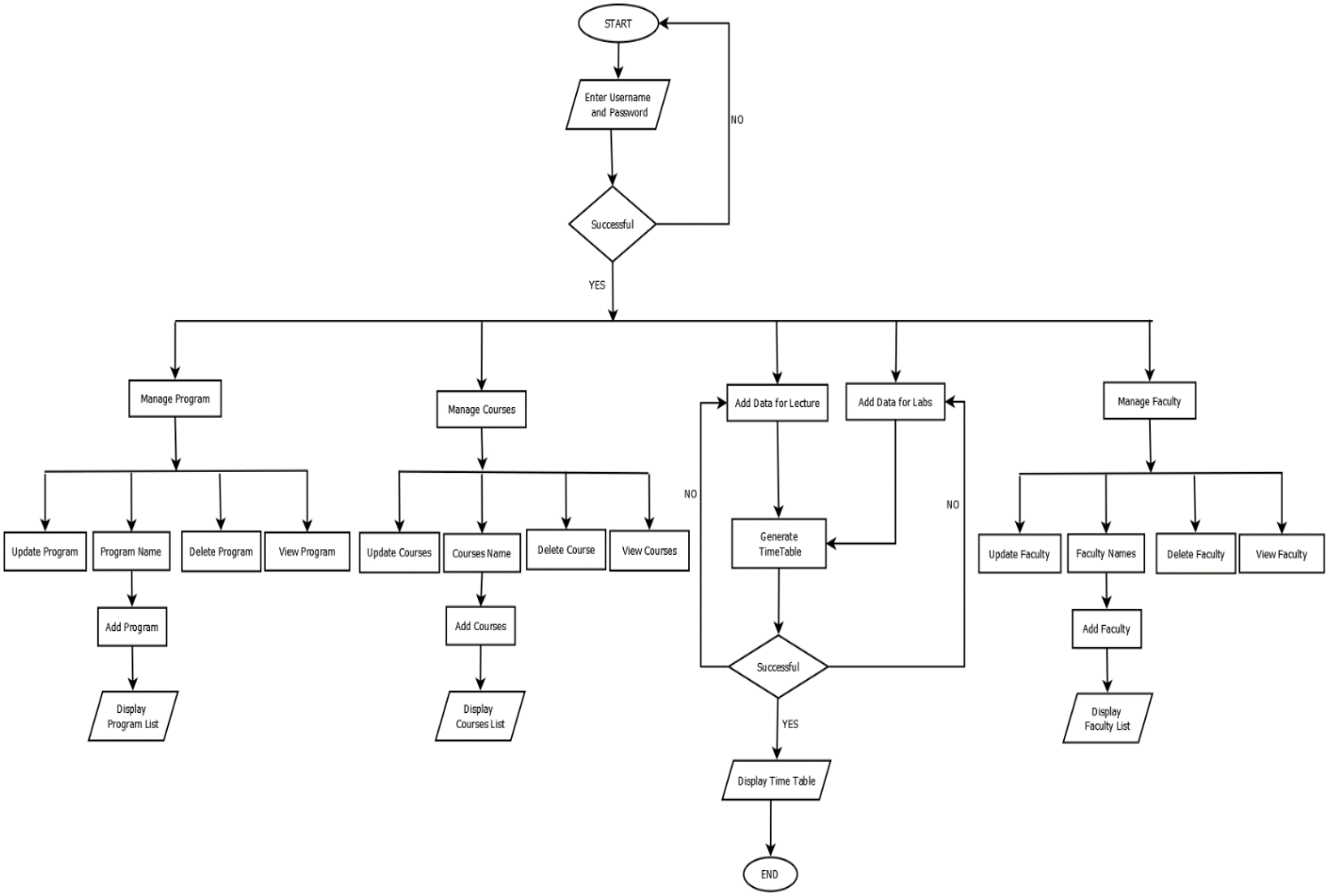

Fig. 2.1 Flow Chart of the Web App

2.2.1 ADMIN Activities

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 Time Table:** For generating the time table, Admin has to give the excel sheet containing student details and this system generate the feasible timetable accordingly.

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

The Use Case diagram (Fig 2.2) below displays the activities for the Admin on WEB-App.

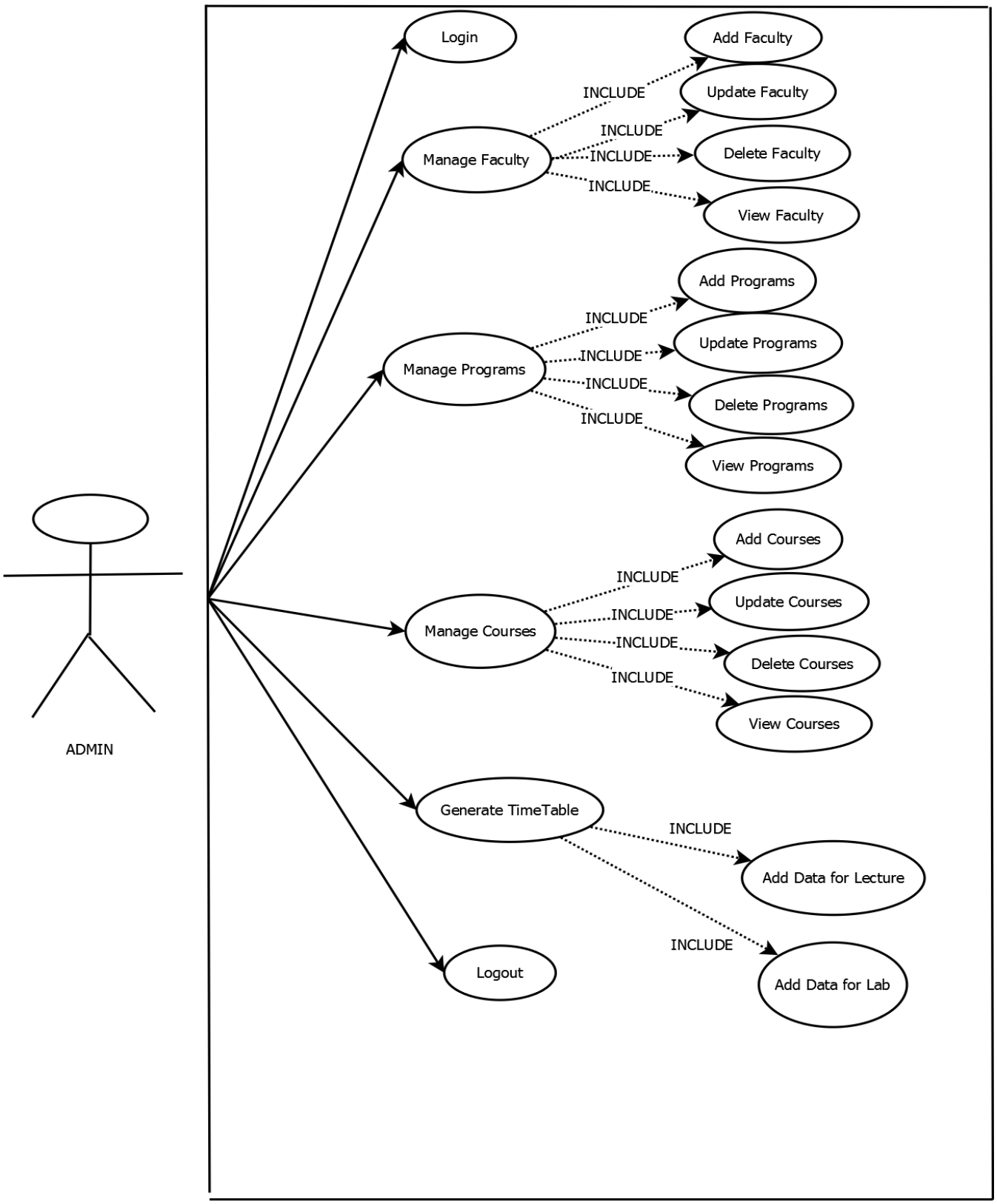

Fig 2.2 Use Case Diagram

2.3 User Characteristics

This project is mainly intended for only one type of audience.

1. ADMIN

3. Specific Requirements

3.1 External Interface

The interfaces that need to be attended while development of the Automatic Time Table Generator will be –

1. **User Interface**
2. **Hardware Interface**
3. **Software Interface**

3.1.1 User Interfaces

User Interface is the actual part of the system which allows the Admin to interact with the system. The user interface of this system contains only 1 interface.

1. **Web App**

   Web App is the User Interface for the Admin to manage and interact with the system.

3.1.2 Hardware Interfaces

The hardware interfaces requirements for the Web App are described below:-

Web App:

Server Side

a) RAM: 2 GB and above.
b) Hard disk: 250 GB and above.
c) Processor: P4 and above
d) OS: Windows
e) Server: Apache Server
f) Database: MySQL
g) Browser: Any Browser

Client Side

a) RAM: 2 GB and above.
b) Hard disk: 250 GB and above.
c) Processor: P4 and above
d) OS: Windows
e) Browser: Any Browser

3.1.3 Software Interfaces

The development of the Automatic Time Table Generating System will require the following software interfaces:

1) Laptop/ Personal Computer
2) Web Pages
3) MySQL Database
4) Programming in PHP

4. Functional Requirements

4.1 Use Case Model

Develop a system that can help the institute to make the timetable automatically.

The target actors of the System are:-

1. ADMIN

4.2 Use Cases

4.2.1 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.

4.2.2 Manage faculty

4.2.2.1 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.

4.2.2.2 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.

4.2.2.3 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.

4.2.2.4 View faculty

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

Actor: ADMIN

Input: -

Output: Display the list of faculties.

4.2.3 Manage Programs

4.2.3.1 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.

4.2.3.2 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.

4.2.3.3 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.

4.2.3.4 View Programs

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

Actor: ADMIN

Input: -

Output: Display the list of programs.

4.2.4 Manage Courses

4.2.4.1 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.

4.2.4.2 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.

4.2.4.3 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.

4.2.4.4 View Courses

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

Actor: ADMIN

Input: -

Output: Display the list of courses.

4.2.5 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.