

AUTOMATED TIME TABLE **SCHEDULAR**

By:

**CAPT M. AHSIN RIAZ
CAPT KHALIL AWAIS BUTT
CAPT EHTESHAM UL HAQ
CAPT ARSLAN ASHRAF
(BESE 22A)**

**Supervised by:
DR. AYESHA MAQBOOL**

**DEPT OF CSE
MILITARY COLLEGE OF SIGNALS, NUST**

Abstract

Automated timetable makers are a commonly used practice in the academic industry, where generating a clash free timetable and sharing among various stakeholders is NP hard problem. Although it is tried and tested to be a reliable mode of generating and sharing a timetable, it has some limitations in terms of availability, efficiency and effectiveness etc. The idea of the automated timetable generator can be replicated to make date sheet generator, to schedule a class presentation, appointments, reservations and bookings etc. We propose the creation of such a system. This Automated timetable maker is created with the aim to generate and manage the timetable using Artificial intelligence and sharing timetable for voting among the faculty members and sharing the finalized version of timetable after voting with the students. Using this system, each teacher and student can view their timetable once it is finalized for a given semester but they cannot edit them. Automated Timetable Maker generates timetable for each class and teacher, in keeping with the availability calendar of teachers, availability and capacity of physical resources (such as classrooms, computer labs, and lecture halls) and rules applicable at different classes, semesters, teachers and subjects level. The project was realized and envisaged based on the concept of automated timetable generation. We are very hopeful that such a system will help various departments in a highly engaging, interactive and effective manner.

Table of Contents

List of Figures	12
List of Tables	13
Chapter 1: Introduction	16
1.1 Background	19
1.2 Motivations and Challenges	20
1.3 Goals and Objectives	20
1.4 Solution Overview	21
1.5 Report Outline	21
Chapter 2: Literature / Market Survey	22
2.1 Introduction	23
2.2 Literature Review/Technologies Overview	24
2.3 Summary	27
Chapter 3: Requirement Analysis	28
3.1 Introduction	29
3.2 Problem Scenarios	30
3.3 Functional Requirements	31
3.4 Non-Functional Requirements	32
Chapter 4: System Design	34
4.1 Introduction	35
4.2 Architectural Design	36
4.3 Detailed Design	37
Chapter 5: Implementation	107
5.1 Endeavour	108
5.2 Flow Control/Pseudo codes	108
5.3 Components, Libraries, Web Services and stubs	115
5.4 Best Practices / Coding Standards	116
5.5 Deployment Environment	117
5.6 Summary	118
Chapter 6: Testing and Evaluation	119

6.1 Introduction	120
6.2 List of Test Scenarios	120
6.3 Performance and Evaluation	155
6.4 Summary	155

References

Appendices

List of Figures

Figure 4. 1 Architecture Design.....	30
Figure 4. 2 Architecture View Diagram	31
Figure 4. 3 UseCase Diagram	33
Figure 4. 4: Entity Relationship Diagram.....	70
Figure 4. 5: Class Diagram	71
Figure 4.5. 1 Login of SuperAdmin.....	72
Figure 4.5. 2 Login of Admin	73
Figure 4.5. 3 Login of Teacher	74
Figure 4.5. 4 Login of Student	75
Figure 4.5. 5 View Timetable by Admin	77
Figure 4.5. 6 View Timetable by Teacher	78
Figure 4.5. 7 View Timetable by student.....	79
Figure 4.5. 8 View notification by admin	81
Figure 4.5. 9 View Notification by Teacher	82
Figure 4.5. 10 View Notification by Student.....	83
Figure 4.5. 11 Send notification by admin.....	84
Figure 4.5. 12 Manage Resources	85
Figure 4.5. 13 Manage Teacher	86
Figure 4.5. 14 Add teacher.....	87
Figure 4.5. 15 Update teacher	88
Figure 4.5. 16 Import Students	89
Figure 4.5. 17 Export Courses	90
Figure 4.5. 18 View Venues	91
Figure 4.5. 19 Delete Clashes	92
Figure 4.5. 20 Manage Admin	93
Figure 4.5. 21 Add Admin	94
Figure 4.5. 22 Delete Admin.....	95
Figure 4.5. 23 Update Admin	96
Figure 4.5. 24 View Admin	97
Figure 4.5. 25 Generate Timetable	98
Figure 4.5. 26 Voting for timetable	99
Figure 4.5. 27 Report Clash	100
Figure 4.5. 28 Add preferences.....	101
Figure 4.5. 29 Evaluate Timetable.....	101
Figure 5.2. 1 Flow control of super admin.....	104
Figure 5.2. 2 Flow control of Admin	105
Figure 5.2. 3 Flow control of tecaher.....	106
Figure 5.2. 4 Flow control of student.....	107
Figure 5.2. 5 Flow Control of voting mechanism.....	108
Figure 5.2. 6 Flow control of generate timetable.....	109
Figure 5.5. 1 Deployment Diagram	113

List of Tables

Table 2. 1: Comparison of existing systems	17
Table 4. 1 Login of Super Admin	34
Table 4. 2 Login of Admin	34
Table 4. 3 Login of Teacher.....	35
Table 4. 4 Login of Student	35
Table 4. 5 View Timetable by SuperAdmin	36
Table 4. 6 View Timetable by Admin	36
Table 4. 7 View Timetable by Teacher.....	37
Table 4. 8 View Timetable by Student	38
Table 4. 9 View Notification by Super Admin	38
Table 4. 10 View Notification by Admin	39
Table 4. 11 View Notification by Teacher.....	39
Table 4. 12 View Notification by Student	40
Table 4. 13 Send Notification by Admin	40
Table 4. 14 Manage resources	41
Table 4. 15 Manage teachers	41
Table 4. 16 Add Teacher.....	42
Table 4. 17 update teachers.....	43
Table 4. 18 Manage students	43
Table 4. 19 Add students	44
Table 4. 20 update students.....	44
Table 4. 21 Manage courses.....	45
Table 4. 22 Add courses.....	46
Table 4. 23 Update Courses	46
Table 4. 24 Manage Venues.....	47
Table 4. 25 Add Venues.....	48
Table 4. 26 Update venues.....	48
Table 4. 27 Import resources	49
Table 4. 28 Import teachers	50
Table 4. 29 Import students	50
Table 4. 30 Import courses.....	51
Table 4. 31 Import venues	51
Table 4. 32 Import clashes	51
Table 4. 33 Import preferences	52
Table 4. 34 Export resources	52
Table 4. 35 Export teachers	53
Table 4. 36 Export students	53
Table 4. 37 Export courses.....	54
Table 4. 38 Export venues	54
Table 4. 39 Export clashes	55
Table 4. 40 Export preferences	55

Table 4. 41 View resources.....	56
Table 4. 42 View teachers.....	56
Table 4. 43 view students.....	57
Table 4. 44 View courses.....	57
Table 4. 45 view venues.....	58
Table 4. 46 View clashes	58
Table 4. 47 View preferences	58
Table 4. 48 Delete resources.....	59
Table 4. 49 Delete teachers.....	59
Table 4. 50 Delete students.....	60
Table 4. 51 Delete courses	60
Table 4. 52 Delete venues.....	61
Table 4. 53 Delete clashes	61
Table 4. 54 Delete preferences.....	62
Table 4. 55 Manage Admin	62
Table 4. 56 Add Admin	63
Table 4. 57 Delete Admin.....	63
Table 4. 58 Update Admin.....	64
Table 4. 59 View Admin.....	65
Table 4. 60 Generate Timetable.....	65
Table 4. 61 Voting for timetable.....	66
Table 4. 62 Report Clashes	67
Table 4. 63 Add preferences	67
Table 4. 64 Evaluate timetable.....	68

Chapter 1: Introduction

Chapter 1: Introduction

Many state-of-the-art and cutting-edge universities in the world rely on manual timetable. The overreliance of this practice in academic institutes is causing many slots, teacher and students clashes. Resource utilization is a routine task in a large number of daily life systems. In many systems, we have resources such as people, venues, preferences, priorities, time and equipment etc. that has to be efficiently managed in order to achieve smooth operation. Some daily life examples are time table, date sheet, appointments, reservations and bookings etc.

In this context, we have the issue of time table generation, which takes weeks if not months to achieve, and the final version still consists of clashes and compromises. A time table that considers the preferences and choices of all stakeholders is always desired, but unfortunately this is not the case at this campus.

Coming across the problems of manual timetable generation, timetable and date sheet management or scheduling is a big issue. The manual system of time table generation with large number of students requires a lot of time and as it is manually done so it requires a lot of effort and have many related viewing issues there is no proper resource utilization which results in increased chances of occurrence of deadlock and Clashes.

Nowadays, technology is moving towards advancement, and people and organizations prefer a digital presence over a physical one. In a world where virtual reality is actually being practiced, we observe that almost every company (and even individuals) has their own web pages and web sites. Online systems have been very successful in overcoming the problems with traditional practices by ensuring higher amount of access, availability, efficiency and effectiveness. We propose the use of such practices to create Automated Timetable Maker for our CSE department.

Our end goal is to develop a timetable that is free of clashes, allows the students to take the courses of their choice, and caters for teacher' s preferences as well. Also, we aim to maximize the utilization of resources. We aim to use search-based strategies for the purpose. In search-based algorithms, we take a set of inputs and give a fitness function that we want to maximize or minimize. The algorithm runs in successions and terminates when it finds a near optimal solution. It is important to note that we cannot guarantee an ideal solution using search-based strategies; however, we can find a number of solutions that are close to being optimal. Our motivation is to build a consistent timetable so that the student can take multiple courses of their own choice. It will also reduce the burden from coordinator. It saves a lot of time. Reduce stress from students, teachers and coordinators.

Different users shall have different level of access to the content. In the context of a academic institutes, there shall be following users of the software.

- **Super Admin:** Super Administrator is the Administrator that will sign up Administrators and are responsible for managing Administrators.
- **Admin:** Administrator is the stake holder that has complete rights, privileges and access of the system.
- **Teachers:** The teachers often have to share critical information like attendance and grading material etc. They also have to update the students about their respective availability hours and days. For that, they can use the system.
- **Students:** They are the stakeholders who acquire the maximum benefit from the system. They can view, update and share information.

1. Admin:

- Import/Export file formats
- Manage resources (rooms, courses)
- Manage teachers and students

- Manage preferences
- Generate Timetable
- Upload for voting/ voting manager
- Constraints manager (check break time)
- Generate notification

2. Super Admin:

- Sign up administrator

3. Teacher:

- Manage profile
- View timetable
- Add preferences
- Vote for timetable
- View notification

4. Student:

- View Profile
- Manage Profile
- Register

1.1 Background

Currently our academic institutes has manual system of generating a timetable. It's outdated now. As nobody have time to stand in rush in order to remove clashes again and again and update the time table accordingly. First manual systems were used which causes a lot of difficulties for students, teachers and Coordinator.

Limitations of Existing System:

1. Order of Data: Automated Timetable Maker allows Admin to quickly check courses, teachers and students and can monitor the class scheduling and management easily, which helps to keep the data ordered. Different views module in our system help to avoid data redundancy and show only concerned data to user i.e. if the student is of 7th semester it will show only his/her registered courses timetable to the student.

2. Complexity: Online system is less complex than manual system of generating timetable, which can make it easier for untrained people to access and manipulate data. Anyone having the basic knowledge of websites can work on this system.

3. Inconsistency of data: There will be unavailability for future use, since timetable might get misplaced during manual management. So timetable won't be preserved properly for future use.

4. Damage: Manual paper stack are vulnerable to damage, destruction and theft in ways that digital databases are not.

5. Editing and Communication: Manual timetable do not allow users to easily edit data or information, it creates a lot of mess while updating the manual timetable. Manual timetables often cannot be edited directly, forcing users to make new copies. To circulate timetable on paper, users must require peons and other staff. Automated timetable maker is a web application allows admin to edit time table as desired by faculty in less time.

1.2 Motivation and Challenges

Almost all leading institutions in Pakistan currently lack an Online Timetable maker. Though some have taken the aid of third party websites like Gmail to interact, it comes at the cost of mixing one' s social life with professional. Keeping this in mind, educational institutes will find this software extremely useful. The “ Automated Time table Maker” is web-based software, with supplementary application software, that aims to aid the institutes by providing automated timetable generation.

There is no time table generator in most of the universities of Pakistan. It is a basic need having automated timetable generator in the Universities to lessen the burden of admin for creating time tables again and again in order to eradicate clashes.

The academic institutes admin can now easily send all the versions of timetable to all the faculty members for voting. Students can easily access only his/her timetable therefore avoiding messy timetable and searching only his/her registered courses timetable. The management authorities can automatically set to activate/deactivate a timetable uploaded for voting, for a given time period thus helping in automatizing the posting and removal of timetables after the given time period. The management authorities can also manually activate/deactivate the timetable in case the academic institutes/department wants to remove the timetable due to clashes.

1.3 Goals and Objectives

Currently academic institutes administration is generating a manual time table and circulates a paper based timetable. A printed black and white timetable is pasted on the notice board to announce students their timetable. The method indeed consumes the administrative and clerical time of the academic institutes. The project will open the new powerful channel of sharing timetable that was generated using AI algorithms between faculty, academic institutes management and students. The system will be operated by Administrator and other staff to view and vote for best timetable. Also the aim of this project is to develop the framework so that it is easy to extend in the future i.e.

Automated Datasheet Maker. The proposed system's objectives are to overcome all the limitations and drawbacks of the existing system.

The primary aim of the Automated Timetable Maker Software project is to create a fully functional Timetable Generation system which will efficiently handle all of its assigned tasks.

1.4 Solution Overview

Our project is a **Automated Timetable Maker** in which we schedule our timetable. Our end goal is to develop a timetable that is free of clashes, allows the students to take the courses of their choice, and caters for teacher's preferences as well. Also, we aim to maximize the utilization of resources. We aim to use search-based strategies for the purpose. In search-based algorithms, we take a set of inputs and give a fitness function that we want to maximize or minimize. The algorithm runs in successions and terminates when it finds a near optimal solution. It is important to note that we cannot guarantee an ideal solution using search-based strategies; however, we can find a number of solutions that are close to being optimal. Our motivation is to build a consistent timetable so that the student can take multiple courses of their own choice. It will also reduce the burden from coordinator. It saves a lot of time. Reduce stress from students, teachers and coordinators.

1.5 Report Overview

This report is divided into different chapters to explain the process and methodologies applied to carry out this project. It includes market surveys in which we take into account all similar apps. The report also consists of the requirement analysis phase and the design phase which try to give a complete picture of the software we are about to make.

Chapter 2: Literature/Market Survey

Chapter 2: Literature/Market Survey

2.1 Introduction

We are trying to develop automated software which helps to generate an automated timetable. By looking at the existing systems we understand that all institutions/organizations have their own timetable making, managing and maintaining strategies which most the times are done manually. As it is time and effort consuming process.

In this chapter we will discuss the literary elements like features of various infamous automated timetable maker; what they have done, what we are doing and how our project is similar or different than theirs and why. This survey helped us classify our project under the category of providing services, organize our idea to achieve our goals, perceive the big picture of this area. In order to cater the needs of our stake holders it is necessary to have some background knowledge about similar kinds of system currently present in the market.

2.2 Literature Review/Technologies Overview

While conducting a literature review of the existing systems, we came across a number of applications. There are many existing systems of timetable available. But there are some problems with all of these. We are making automated timetable maker which overcome all the problems that was in the existing systems.

Existing systems in literature are:

- 1) CELCAT Timetabler
- 2) FET – Free Timetable Software
- 3) Mimosa

These systems are summarized in the table below:

Table 2. 1: Comparison of existing systems

Features	CELCAT	FET	Mimosa	Automated Maker	Timetable
Automated	True	True	True	True	
Check Conflict Prevention	True	True	True	True	
File format	False	True	False	True	
Check Teacher Preferences	False	False	False	True	
Voting	False	False	False	True	
Check Pre- requisites	False	False	False	True	
Notification	False	False	False	True	
Different views for enrolled students	False	False	True	True	

1. CELCAT

CELCAT (pronounced Sell-Cat) has been in production since 1989 and is used in over 30 countries. A full CELCAT site license includes the following software depending on license options. Timetabler Server - the connection between the SQL Server and the user software.

A timetable is a plan of the times at which events are scheduled to take place. Within CELCAT, timetables are constructed as grids where days are divided into blocks of time. These are referred to as periods. Events are created within a timetable grid when one or more resources are scheduled to take place during one or more periods. A user-friendly Event Window is used to assign resources to an event or part of an event, and to alter or set the weeks that the event runs over. A Resource can be a module, room, staff, groups, students, equipment and teams in CELCAT Terminology.

We cannot download a timetable from CELCAT in different file formats. There is no module of voting, checking the teacher preferences and checking the pre requisite courses. It does not send notifications and there is no option for different views for enrolled students available.

2. FET

Generating a timetable is very random process. So it might happen that generating a timetable just need a few seconds and generating a timetable with the same dataset need the next time several minutes. I can't say how much time is needed to your timetable, but I know datasets that need several hours to solve even on a modern computer.

We can download a timetable from FET in different file formats. But there is no module of voting, checking the teacher preferences and checking the pre requisite courses. It does not send notifications and there is not different views for enrolled students available.

3. Mimosa

Creating and maintaining timetables is often a complex task for both people and software. The technical side of Mimosa is kept as simple and as self-contained as possible. The technology is based on a collection of efficient optimization algorithms.

We cannot download a timetable from Mimosa in different file formats. There is no module of voting, checking the teacher preferences and checking the prerequisite courses. It does not send notifications but there are different views for enrolled students available.

4. Automated Timetable Maker

Our system solves all the problems which were in the existing systems. We will develop a timetable that is free of clashes, allows the students to take the courses of their choice, and caters for teacher's preferences as well. Also, we aim to maximize the utilization of resources. We use search-based strategies for the purpose. In search-based algorithms, we take a set of inputs and give a fitness function that we want to maximize or minimize. The algorithm runs in successions and terminates when it finds a near optimal solution. It is important to note that we cannot guarantee an ideal solution using search-based strategies; however, we can find a number of solutions that are close to being optimal. We are going to build a consistent timetable so that the student can take multiple courses of their own choice. It will also reduce the burden from coordinator. It saves a lot of time. We also send notification about it and there is also a voting mechanism in it.

Technical Overview

The project is based on the on web application and its interface with PHP technology. To implement the project we need to know some basic concepts of it. The basic concepts of the project consist of the following:

- **PHP:**

PHP: Hypertext Preprocessor", is an open-source, reflective programming language used mainly for developing server-side applications and dynamic web content.

- **XAMPP:**

XAMPP is a free open source cross platform web server package consisting of Apache Http server, MySQL database and interpreters for scripts written in PHP and Perl programming language.

- **MySQL:**

MySQL is an open source RDMS which manages the data contained within the databases for use in web applications, and is a central component of the widely used LAMP open source web application software stack. MySQL is used as Web Server database for storing all the incoming notices.

2.3 Summary

Automated Timetable maker is an Internet based Web Application that helps to develop a timetable that is free of clashes, allows the students to take the courses of their choice, and caters for teacher' s preferences as well. Also, we aim to maximize the utilization of resources. We use search-based strategies for the purpose. In search-based algorithms, we take a set of inputs and give a fitness function that we want to maximize or minimize. The algorithm runs in successions and terminates when it finds a near optimal solution. It is important to note that we cannot guarantee an ideal solution using search-based strategies; however, we can find a number of solutions that are close to being optimal. We are going to build a consistent timetable so that the student can take multiple courses of their own choice. It will also reduce the burden from coordinator. It saves a lot of time. Reduce stress from students, teachers and coordinators. We also sends notification about it and there is also a voting mechanism in it.

Although there are many timetable maker existing today but the features of our system are entirely different than the existing ones. Existing systems have limited

functionality where as our system has a broad variety of features. This chapter contains the session of market survey in which briefly discussed about the existing systems.

Chapter 3: Requirement Analysis

Chapter 3:

Requirement Analysis

3.1 Introduction

In systems engineering and software engineering, **requirements analysis** encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, *analyzing, documenting, validating and managing* software or system requirements.

Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design. Requirements are descriptions of the services that a software system must provide and the constraints under which it must operate.

Requirements can range from high-level abstract statements of services or system constraints to detailed mathematical functional specifications. The Requirement Engineering (RE) is the most important phase of the Software Development Life Cycle (SDLC). This phase is used to translate the imprecise, incomplete needs and wishes of the potential users of software into complete, precise and formal specifications. The specifications act as the contract between the software users and the developers. Therefore the importance of Requirement Engineering is enormous to develop effective software and in reducing software errors at the early stage of the development of software. Since Requirement Engineering (RE) has great role in different stages of the SDLC, its consideration in software development is crucial.

This chapter will give a brief overview of the requirement gathering, elicitation and analysis phase and the requirements which we thought were fit to be included in the Functional and Non-functional category. Functional requirements which are statements of services that the system should provide, how the system should react to particular inputs and how the system should behave in particular situations. Whereas non-functional

requirements which are constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, etc.

3.2 Problem Scenarios

Currently our academic institutes has manual system of generating a timetable. It' s outdated now. As nobody have time to stand in rush in order to remove clashes again and again and update the time table accordingly. First manual systems were used which causes a lot of difficulties for students, teachers and Coordinator.

The manual system of time table generation with large number of students is:

- ✓ Time consuming
- ✓ Requires a lot of effort
- ✓ Tedious
- ✓ Viewing issues

There is no proper resource utilization which results in:

- ✓ Increased chances of occurrence of deadlock and
- ✓ Clashes

There are many existing system present but there is no such system which could cater the need of all users in our academic institutes. Our system solves all the problems which were in the existing systems. We will develop a timetable that is free of clashes, allows the students to take the courses of their choice, and caters for teacher' s preferences as well. Also, we aim to maximize the utilization of resources. We use search-based strategies for the purpose. In search-based algorithms, we take a set of inputs and give a fitness function that we want to maximize or minimize. The algorithm runs in successions and terminates when it finds a near optimal solution. It is important to note that we cannot guarantee an ideal solution using search-based strategies; however, we can find a number of solutions that are close to being optimal. We are going to build a consistent timetable so that the student can take multiple courses of their own choice. It will also reduce the burden from coordinator. It saves a lot of time. Reduce stress from students, teachers and coordinators. We also send notification about it and our system “ Automated Timetable Maker” also has a voting mechanism in it.

3.3 Functional Requirements

Our system automated Timetable maker has four types of users:

- Super Admin
- Admin
- Teacher
- Students

All these users have their own rights and privileges according to their designation. Super Administrator is responsible for managing Administrators. Admin will have all the rights and privileges because admin will be the main controller of our system. Similarly teachers and students have privileges assigned to them.

3.3.1 Super Admin

FR.1 Super admin shall be able to login in to his/her account. Admin will have his/her specific employee ID and a password with the help of these two credentials he will be able to log into his/her account.

FR.2 Super admin shall be able to manage administrators.

3.3.2 Admin

FR.1 Admin will be able to login in to his/her account. Admin will have his/her specific employee ID and a password with the help of these two credentials he will be able to enter into his account.

FR.2 Admin will be able to import/export timetable in different file formats.

FR.3 Admin has a right to manage resources (students, teachers, rooms, courses).

FR.4 Admin has a right to manage (add, and update) all resources.

FR.5 Admin can manage teacher' s preferences.

FR.6 Admin will be able to generate multiple timetables.

FR.7 Admin has a privilege to upload different timetables for voting.

FR.8 Admin will be able to manage different constraints (check break time).

FR.9 Evaluate timetable (Admin will be able to generate notification to teachers and student when best time table is selected after voting).

FR.10 Admin has a right to view resources.

FR.11 Admin has a right to delete resources.

3.3.3 Student

FR.1 Student shall be able to login and register him/herself.

FR.2 Student will be able to view and manage profile.

FR.3 Student will be able to get notification about timetable.

FR.4 Student will be able to view notification

FR.5 Student will have a privilege to view timetable in customized form (according to registered courses).

FR.6 Student will report Clashes.

3.3.4 Teacher

FR.1 Teacher shall be able to login or register him or herself, only after registering themselves with the help of their specific employee ID.

FR.2 Teacher will be able to view and manage profile

FR.3 Teacher will be able to get notification about time table

FR.4 Teacher will have a privilege to view timetable in customized form (according to registered courses)

FR.5 Teacher has a right to vote for best timetable

FR.6 Teacher has a right to add preferences

FR.7 Teacher will able to view notification

3.4 Non-Functional Requirements

Non-functional requirements describe the constraints on the services and/or functions offered by the system and constraints on the development process and standards.

3.4.1 Accuracy

The system will generate the near to optimal solution. The timetable and the notifications will be sent to respective teachers and students. No student will receive the timetable other than enrolled students.

3.4.2 Usability in terms of Learnability and Operability

The system must have an easy to use and understandable user interface. This will ensure that the user, for which this system is intended, will be able to achieve a level of proficiency with the system with minimum effort in a very short period of time.

- System will be easy to learn and understand.
- System will be easy to use and minimum effort will be required to use the system.

Chapter 4: System Design

Chapter 4:

System Design

4.1 Introduction:

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

In the Software Development Life Cycle (SDLC), software design comes after the requirement analysis. Software design is very important for better understanding and implementation of the system. Software Design describes how the software will operate in the form of user interfaces, design patterns and software architecture. Software analysis diagrams and designs are simply the design of software. It applies systematic and engineered approach for software realization. Unlike requirement analysis, where focus is on "What", the design use to focus on "How" part of the system. This chapter includes all the high level and low level design details of the software.

4.2 Architecture Design

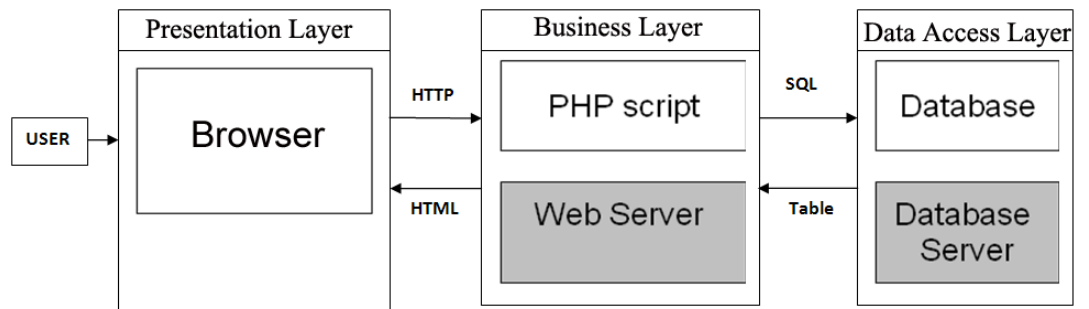
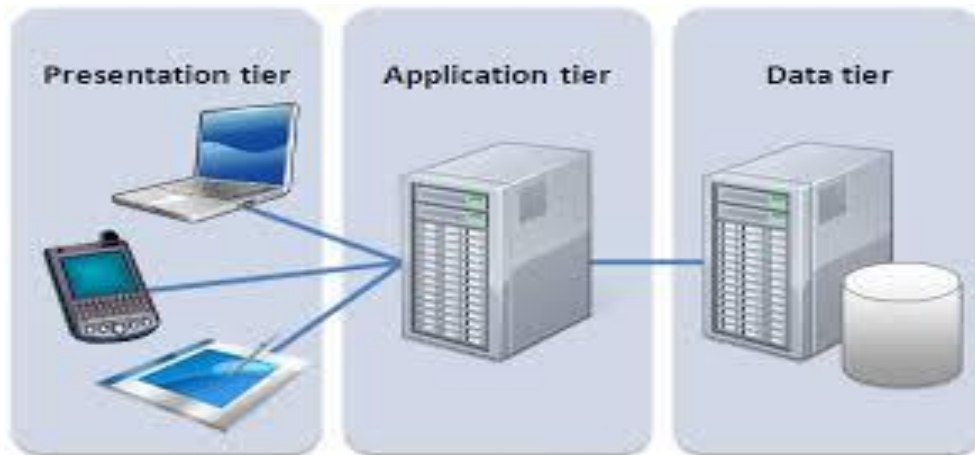


Figure 4. 1 Architecture Design

4.2.1 Architecture View Diagram

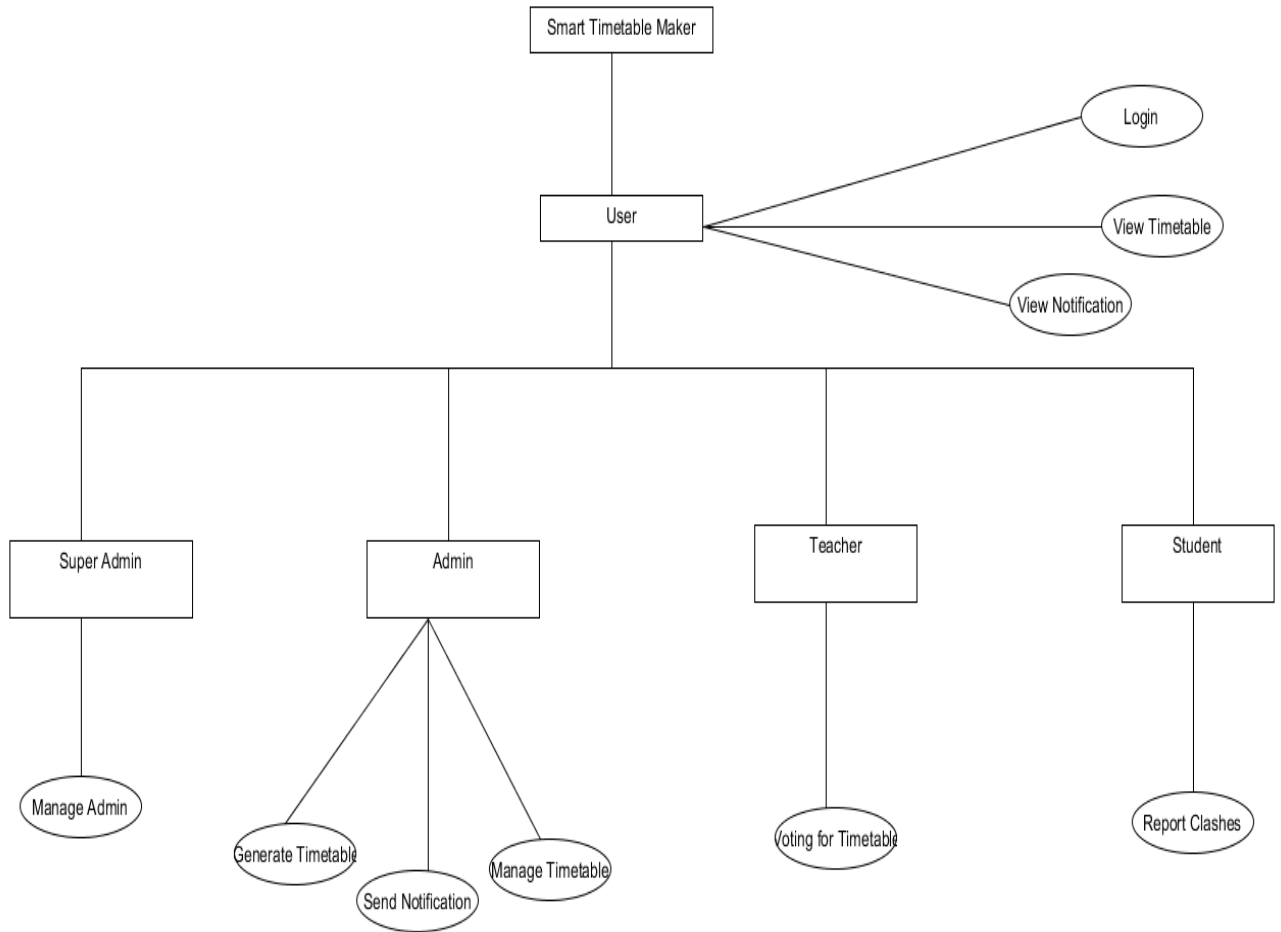


Figure 4. 2 Architecture View Diagram

4.3 Detailed Design

Detailed design of the system is the last design activity before implementation begins. The hardest design problems must be addressed by the detailed design. The detailed design is still an abstraction as compared to source code, but should be detailed enough to ensure that translation to source is a precise mapping instead of a rough interpretation.

4.3.1 Use case Diagram

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analyzed to gather its functionalities use cases are prepared and actors are identified.

Now when the initial task is complete use case diagrams are modeled to present the outside view.

So in brief, the purposes of use case diagrams can be as follows:

- Used to gather requirements of a system.
- Used to get an outside view of a system.
- Identify external and internal factors influencing the system.
- Show the interacting among the requirements are actors.

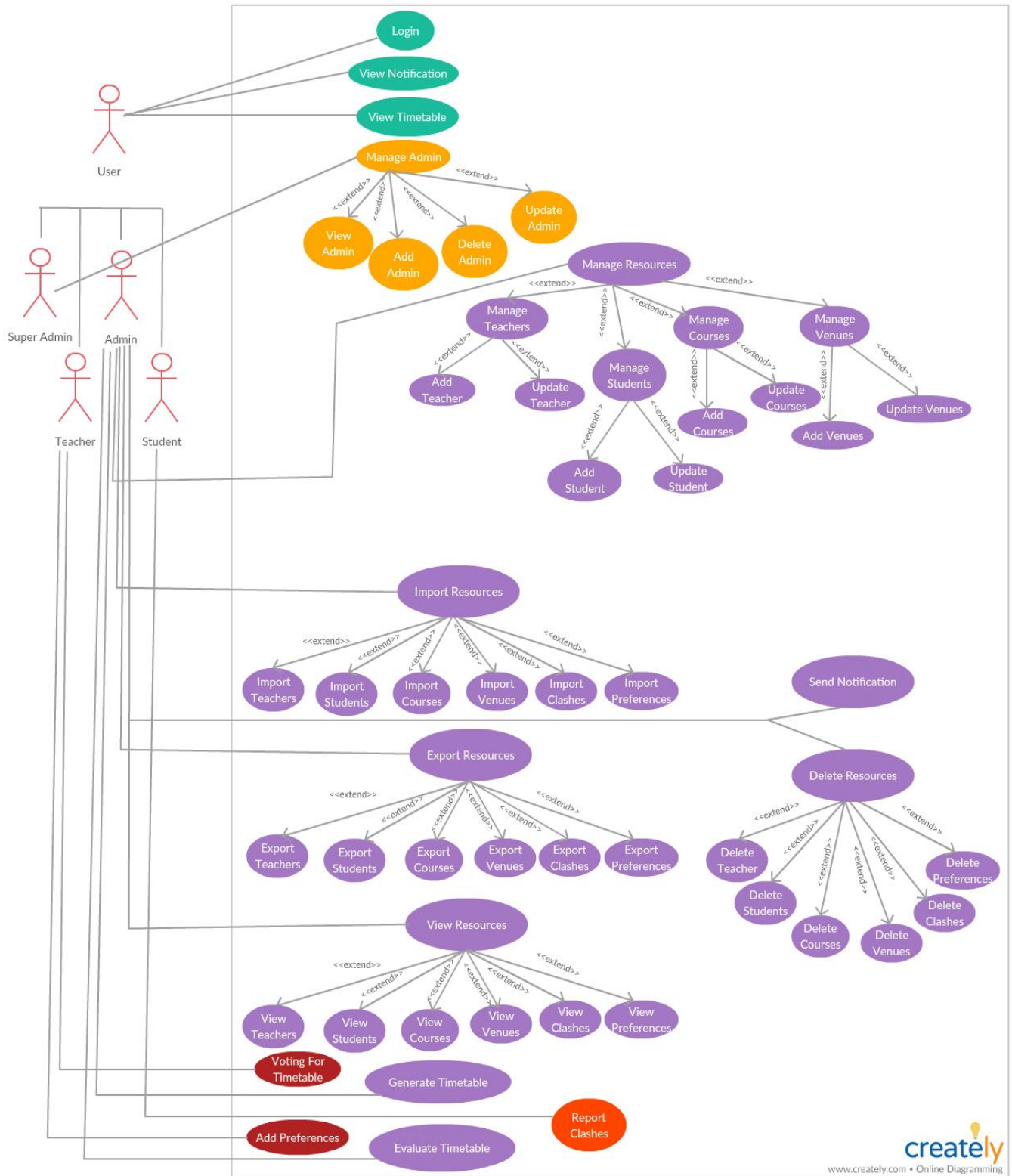


Figure 4. 3 Use Case Diagram

4.3.2 Use case Description

4.3.2.1 Login of Super Admin

Table 4. 1 Login of Super Admin

ID:	05
Title:	Login of Super Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Super Admin
Description:	A Super Admin who has signed up already, log in to his/her account to see all the details.
Pre-conditions:	Super Admin should have signed up.
Post conditions:	Super Admin should have successfully login.
Main Success Scenario:	<ol style="list-style-type: none">1. Super Admin clicks on “ sign in” button.2. System displays “ username” and “ password” field.3. Super Admin fills these fields.4. Super Admin clicks "Login" button.5. System displays all details.
Extensions:	May be Super Admin give wrong information and system give login error.
Special Requirements:	System should provide proper placeholder for sign up form.
Frequency of Use:	Can be used whenever you want to login.
Technology and data variations list	Login information is entered by keyboard.

4.3.2.2 Login of Admin

Table 4. 2 Login of Admin

ID:	06
Title:	Login of Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Admin
Description:	Admin who has signed up already, log in to his/her account to see all details.
Pre-conditions:	Admin should have signed up.
Post conditions:	Admin should have successfully login.
Main Success Scenario:	<ol style="list-style-type: none">1. Admin clicks on “ sign in” button.2. System displays “ username” and “ password” field.3. Admin fills these fields.4. Admin clicks "Login" button.5. System displays all details.

Extensions:	May be Admin give wrong information and system give login error.
Special Requirements:	System should provide proper placeholder for sign up form.
Frequency of Use:	Can be used whenever you want to login.
Technology and data variations list	Login information is entered by keyboard.

4.3.2.7 Login of Teacher

Table 4. 3 Login of Teacher

ID:	07
Title:	Login of Teacher
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Teacher
Description:	A Teacher who has signed up already, log in to his/her account to see all details.
Pre-conditions:	Teacher should have signed up.
Post conditions:	Teacher should have successfully login.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Teacher clicks on “ sign in" button. 2. System displays “ username” and “ password” field. 3. Teacher fills these fields. 4. Teacher clicks "Login" button. 5. System displays all details.
Extensions:	May be Teacher give wrong information and system give login error.
Special Requirements:	System should provide proper placeholder for sign up form.
Frequency of Use:	Can be used whenever you want to login.
Technology and data variations list	Login information is entered by keyboard.

4.3.2.8 Login of Student

Table 4. 4 Login of Student

ID:	08
Title:	Login of Student
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Student
Description:	A Student who has signed up already, log in to his/her account to see timetable.

Pre-conditions:	Student should have signed up.
Post conditions:	Student should have successfully login.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Student clicks on “ sign in” button. 2. System displays “ username” and “ password” field. 3. Student fills these fields. 4. Student clicks "Login" button. 5. System displays all details.
Extensions:	May be Student give wrong information and system give login error.
Special Requirements:	System should provide proper placeholder for sign up form.
Frequency of Use:	Can be used whenever you want to login.
Technology and data variations list	Login information is entered by keyboard.

4.3.2.9 View Timetable by Super Admin

Table 4. 5 View Timetable by SuperAdmin

ID:	09
Title:	View Timetable by Super Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Super Admin
Description:	Super Admin has log in to his/her account to see timetable in STTM.
Pre-conditions:	Super Admin should have login into his/her account.
Post conditions:	Super Admin should have successfully view the timetable.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Super Admin clicks on “ View Timetable” button. 2. System displays the window which shows timetable.
Extensions:	May be Super Admin click on wrong button.
Special Requirements:	System should provide proper button information for viewing timetable.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Timetable is clicked by mouse.

4.3.2.10 View Timetable by Admin

Table 4. 6 View Timetable by Admin

ID:	10
Title:	View Timetable by Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Admin

Description:	Admin has log in to his/her account to see timetable in STTM.
Pre-conditions:	Admin should have login into his/her account.
Post conditions:	Admin should have successfully view the timetable.
Main Success Scenario:	1. Admin clicks on “ View Timetable” button. 2. System displays the window which shows timetable.
Extensions:	May be Admin click on wrong button.
Special Requirements:	System should provide proper button information for viewing timetable.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Timetable is clicked by mouse.

4.3.2.11 View Timetable by Teacher

Table 4.7 View Timetable by Teacher

ID:	11
Title:	View Timetable by Teacher
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Teacher
Description:	Teacher has log in to his/her account to see timetable in STTM.
Pre-conditions:	Teacher should have login into his/her account.
Post conditions:	Teacher should have successfully view the timetable.
Main Success Scenario:	1. Teacher clicks on “ View Timetable” button. 2. System displays the window which shows timetable.
Extensions:	May be Teacher click on wrong button.
Special Requirements:	System should provide proper button information for viewing timetable.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Timetable is clicked by mouse.

4.3.2.12 View Timetable by Student

Table 4. 8 View Timetable by Student

ID:	12
Title:	View Timetable by Student
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Student
Description:	Student has log in to his/her account to see timetable in STTM.
Pre-conditions:	Student should have login into his/her account.
Post conditions:	Student should have successfully view the timetable.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Student clicks on “ View Timetable” button. 2. System displays the window which shows timetable.
Extensions:	May be Student click on wrong button.
Special Requirements:	System should provide proper button information for viewing timetable.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Timetable is clicked by mouse.

4.3.2.13 View Notification by Super Admin

Table 4. 9 View Notification by Super Admin

ID:	13
Title:	View Notification by Super Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Super Admin
Description:	Super Admin has log in to his/her account to see notification of STTM.
Pre-conditions:	Super Admin should have login into his/her account.
Post conditions:	Super Admin should have successfully view the notification.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Super Admin click on view notification button. 2. System displays all notifications. 3. Super Admin click on notification from STTM. 4. System displays the notification details.
Extensions:	May be Super Admin click on wrong button.
Special Requirements:	System should provide proper information about it.
Frequency of Use:	Can be viewed whenever you want to view.

Technology and data variations list	View Notification is clicked by mouse.
--	--

4.3.2.14 View Notification by Admin

Table 4. 10 View Notification by Admin

ID:	14
Title:	View Notification by Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Admin
Description:	Admin has log in to his/her account to see notification of STTM.
Pre-conditions:	Admin should have login into his/her account.
Post conditions:	Admin should have successfully view the notification.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin click on view notification button. 2. System displays all notifications. 3. Admin click on notification from STTM. 4. System displays the notification details.
Extensions:	May be Admin click on wrong button.
Special Requirements:	System should provide proper information about it.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Notification is clicked by mouse.

4.3.2.15 View Notification by Teacher

Table 4. 11 View Notification by Teacher

ID:	15
Title:	View Notification by Teacher
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Teacher
Description:	Teacher has log in to his/her account to see notification of STTM.
Pre-conditions:	Teacher should have login into his/her account.
Post conditions:	Teacher should have successfully view the notification.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Teacher click on view notification button. 2. System displays all notifications. 3. Teacher click on notification from STTM. 4. System displays the notification details.
Extensions:	May be Teacher click on wrong button.
Special Requirements:	System should provide proper information about it.
Frequency of Use:	Can be viewed whenever you want to view.

Technology and data variations list	View Notification is clicked by mouse.
--	--

4.3.2.16 View Notification by Student

Table 4. 12 View Notification by Student

ID:	16
Title:	View Notification by Student
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Student
Description:	Student has log in to his/her account to see notification of STTM.
Pre-conditions:	Student should have login into his/her account.
Post conditions:	Student should have successfully view the notification.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Student click on view notification button. 2. System displays all notifications. 3. Student click on notification from STTM. 4. System displays the notification details.
Extensions:	May be Student click on wrong button.
Special Requirements:	System should provide proper information about it.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Notification is clicked by mouse.

4.3.2.17 Send Notification by Admin

Table 4. 13 Send Notification by Admin

ID:	17
Title:	Send Notification by Admin
Scope:	Automated Timetable Maker (STTM)
Level:	User goal
Primary Actor:	Admin
Description:	Admin has log in to his/her account to send notification of STTM.
Pre-conditions:	Admin should have login into his/her account.
Post conditions:	Admin should have successfully send the notification.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin generates timetable. 2. System displays the timetable. 3. System sends the notification to all register teachers for voting. 4. System opens votes for a certain timeframe. 5. Teacher votes for timetable. 6. System displays the final timetable after voting.

	7. System then send notification to teachers and students both.
Extensions:	May be Student click on wrong email.
Special Requirements:	System should provide proper information about it.
Frequency of Use:	Can be viewed whenever you want to view.
Technology and data variations list	View Notification is clicked by mouse.

4.3.2.18 Manage Resources

Table 4. 14 Manage resources

ID:	18
Title:	Manage Resources
Description:	Admin is allowed to manage resources. Admin can manage teachers, students, courses, venues.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully managed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ manage resources" from the menu. 2. System displays manage teachers, manage students, manage courses, and manage venues. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while managing resources.
Frequency of Use:	Frequently used when the details is required to modified.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.19 Manage Teachers

Table 4. 15 Manage teachers

ID:	19
Title:	Manage Teachers
Description:	Admin is allowed to manage teachers. Admin can add/update teacher.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully managed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ manage teachers" from the menu. 2. System displays add teachers and update teachers. 3. Admin selects one of the options.

	4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while managing teachers.
Frequency of Use:	Frequently used when the details is required to modified.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.20 Add Teachers

Table 4. 16 Add Teacher

ID:	20
Title:	Add teachers
Description:	Admin is allowed to manage teachers. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Teacher' s details were successfully added.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ add teachers" from the manage teachers. 2. System displays information to be added i.e. first name, last name, id, age, gender, contact details, address etc. 3. Admin inserted the required information of teacher. 4. System checks for all fields and enabled the add button. 5. Admin presses add button. 6. System displays “ successfully added” message on screen.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. Admin inserted information in invalid format. 5a. Admin inserted the record or part of information that is already in the database i.e. redundant information.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when new teacher is hired and his/her details are required to be added.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.21 Update Teachers

Table 4. 17 update teachers

ID:	21
Title:	Update teacher
Description:	Admin is allowed to manage teachers. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Teacher’ s details were successfully updated.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ update teachers” from the manage teachers. 2. System displays the teacher information. 3. Admin inserted the updated information of teacher i.e. his contact number, his address, designation etc. 4. System displays confirmation message “ updated successfully” .
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 2b. Admin inserted invalid information. 3a. Admin inserted information in invalid format 4a. confirmation message took too long to display.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when teacher’ s specific detail is required to be updated.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.22 Manage Students

Table 4. 18 Manage students

ID:	22
Title:	Manage Students
Description:	Admin is allowed to manage students. Admin can add/update students.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully managed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ manage students “ from the menu. 2. System displays add students and update students. 3. Admin selects one of the options.

	4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while managing students.
Frequency of Use:	Frequently used when the details is required to modified.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.23 Add Students

Table 4. 19 Add students

ID:	23
Title:	Add students
Description:	Admin is allowed to manage students. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Student' s details were successfully added.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ add students” from the manage students. 2. System displays information to be added i.e. first name, last name, Cms, semester, contact details, address etc. 3. Admin inserted the required information of students. 4. System checks for all fields and enabled the add button. 5. Admin presses add button. 6. System displays “ successfully added” message on screen.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. Admin inserted information in invalid format. 5a. Admin inserted the record or part of information that is already in the database i.e. redundant information.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when new students are registered and his/her details are required to be added.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.24 Update Students

Table 4. 20 update students

ID:	24
------------	----

Title:	Update students
Description:	Admin is allowed to manage students. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Student’ s details were successfully updated.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ update students” from the manage students. 2. System displays the student information. 3. Admin inserted the updated information of students i.e. his contact number, his address, semester etc. 4. System displays confirmation message “ updated successfully” .
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 2b. Admin inserted invalid information. 3a. Admin inserted information in invalid format 4a. confirmation message took too long to display.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when student’ s specific detail is required to be updated.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.25 Manage Courses

Table 4. 21 Manage courses

ID:	25
Title:	Manage Courses
Description:	Admin is allowed to manage courses. Admin can add/update courses.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully managed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ manage courses “ from the menu. 2. System displays add courses and update courses. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special	There should be no redundancy of data while managing courses.

requirements:	
Frequency of Use:	Frequently used when the details is required to modified.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.26 Add Courses

Table 4. 22 Add courses

ID:	26
Title:	Add courses
Description:	Admin is allowed to manage courses. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Course' s details were successfully added.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ add courses" from the manage courses. 2. System displays information to be added i.e. course code, course title, credit hours, semester offered etc. 3. Admin inserted the required information of courses. 4. System checks for all fields and enabled the add button. 5. Admin presses add button. 6. System displays “ successfully added” message on screen.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. Admin inserted information in invalid format. 5a. Admin inserted the record or part of information that is already in the database i.e. redundant information.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when new courses are registered and its details are required to be added.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.27 Update Courses

Table 4. 23 Update Courses

ID:	27
Title:	Update courses

Description:	Admin is allowed to manage courses. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Course' s details were successfully updated.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ update courses" from the manage courses. 2. System displays the courses information. 3. Admin inserted the updated information of courses i.e. his course code, title etc. 4. System displays confirmation message “ updated successfully” .
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 2b. Admin inserted invalid information. 3a. Admin inserted information in invalid format 4a. confirmation message took too long to display.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when course' s specific detail is required to be updated.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.28 Manage Venues

Table 4. 24 Manage Venues

ID:	28
Title:	Manage Venues
Description:	Admin is allowed to manage venues. Admin can add/update venues.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully managed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ manage venues “ from the menu. 2. System displays add venues and update venues. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while managing venues.
Frequency of Use:	Frequently used when the details is required to modified.

Technology and data variations list:	Information is modified by using keyboard.
---	--

4.3.2.29 Add Venues

Table 4. 25 Add Venues

ID:	29
Title:	Add venues
Description:	Admin is allowed to manage venues. Admin can add and update any details.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Venue' s details were successfully added.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ add venues” from the manage venues. 2. System displays information to be added i.e. room no, type, capacity etc. 3. Admin inserted the required information of venues. 4. System checks for all fields and enabled the add button. 5. Admin presses add button. 6. System displays “ successfully added” message on screen.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. Admin inserted information in invalid format. 5a. Admin inserted the record or part of information that is already in the database i.e. redundant information.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when new venues are available and its details are required to be added.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.30 Update Venues

Table 4. 26 Update venues

ID:	30
Title:	Update venues
Description:	Admin is allowed to manage venues. Admin can add and update any details.
Primary Actor:	Admin

Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Venue' s details were successfully updated.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ update venues” from the manage venues. 2. System displays the venues information. 3. Admin inserted the updated information of venues i.e. his room no, type etc. 4. System displays confirmation message “ updated successfully” .
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 2b. Admin inserted invalid information. 3a. Admin inserted information in invalid format 4a. confirmation message took too long to display.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when venue' s specific detail is required to be updated.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.31 Import Resources

Table 4. 27 Import resources

ID:	31
Title:	Import resources
Description:	Admin is allowed to Import resources. Admin can import teachers, students, courses, venues, preferences and clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ Import resources “ from the menu. 2. System displays import teachers, students, courses, venues, preferences and clashes. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Import resources.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.32 Import Teachers

Table 4. 28 Import teachers

ID:	32
Title:	Import teachers
Description:	Admin is allowed to Import teachers.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ Import teachers “ from the import resources. 2. System displays import teacher’ s button. 3. Admin presses the button. 4. System imported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Importing.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.33 Import Students

Table 4. 29 Import students

ID:	33
Title:	Import students
Description:	Admin is allowed to Import students.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ Import students “ from the import resources. 2. System displays import student’ s button. 3. Admin presses the button. 4. System imported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Importing.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.34 Import Courses

Table 4. 30 Import courses

ID:	34
Title:	Import courses
Description:	Admin is allowed to Import courses.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	1. Admin selects “ Import courses “ from the import resources. 2. System displays import courses button. 3. Admin presses the button. 4. System imported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Importing.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.35 Import Venues

Table 4. 31 Import venues

ID:	35
Title:	Import venues
Description:	Admin is allowed to Import venues.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	1. Admin selects “ Import venues “ from the import resources. 2. System displays import venues button. 3. Admin presses the button. 4. System imported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Importing.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.36 Import Clashes

Table 4. 32 Import clashes

ID:	36
Title:	Import clashes

Description:	Admin is allowed to Import clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	1. Admin selects “ Import clashes “ from the import resources. 2. System displays import clashes button. 3. Admin presses the button. 4. System imported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Importing.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.37 Import Preferences

Table 4. 33 Import preferences

ID:	37
Title:	Import preferences
Description:	Admin is allowed to Import preferences.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully imported.
Main Success Scenario:	1. Admin selects “ Import preferences “ from the import resources. 2. System displays import preferences button. 3. Admin presses the button. 4. System imported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Importing.
Frequency of Use:	Frequently used when the details is required to imported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.38 Export Resources

Table 4. 34 Export resources

ID:	38
Title:	Export resources
Description:	Admin is allowed to export resources. Admin can export teachers, students, courses, venues, preferences and clashes.
Primary Actor:	Admin

Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully exported.
Main Success Scenario:	1. Admin selects “ Export resources “ from the menu. 2. System displays export teachers, students, courses, venues, preferences and clashes. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while export resources.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.39 Export Teachers

Table 4. 35 Export teachers

ID:	39
Title:	Export teachers
Description:	Admin is allowed to export teachers.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully exported.
Main Success Scenario:	1. Admin selects “ export teachers “ from the export resources. 2. System displays export teacher’ s button. 3. Admin presses the button. 4. System exported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Exporting.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.40 Export Students

Table 4. 36 Export students

ID:	40
Title:	Export students
Description:	Admin is allowed to Export students.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.

Post conditions:	All details were successfully exported.
Main Success Scenario:	1. Admin selects “ Export students “ from the export resources. 2. System displays export student’s button. 3. Admin presses the button. 4. System exported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while exporting.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.41 Export Courses

Table 4. 37 Export courses

ID:	41
Title:	Export courses
Description:	Admin is allowed to Export courses.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully exported.
Main Success Scenario:	1. Admin selects “ Export courses “ from the export resources. 2. System displays export courses button. 3. Admin presses the button. 4. System exported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while exporting.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.42 Export Venues

Table 4. 38 Export venues

ID:	42
Title:	Export venues
Description:	Admin is allowed to Export venues.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully exported.
Main Success Scenario:	1. Admin selects “ Export venues “ from the export resources. 2. System displays export venues button. 3. Admin presses the button.

	4. System exported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while Exporting.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.43 Export Clashes

Table 4. 39 Export clashes

ID:	43
Title:	Export clashes
Description:	Admin is allowed to export clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully exported.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ Export clashes “ from the export resources. 2. System displays export clashes button. 3. Admin presses the button. 4. System exported successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while exporting.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.44 Export Preferences

Table 4. 40 Export preferences

ID:	44
Title:	Export preferences
Description:	Admin is allowed to Export preferences.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully exported.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ Export preferences “ from the export resources. 2. System displays export preferences button. 3. Admin presses the button. 4. System exported successfully.
Extensions:	2a. May be system took too long to respond

Special requirements:	There should be no redundancy of data while exporting.
Frequency of Use:	Frequently used when the details is required to exported.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.45 View Resources

Table 4. 41 View resources

ID:	45
Title:	View resources
Description:	Admin is allowed to view resources. Admin can view teachers, students, courses, venues, preferences and clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ view resources “ from the menu. 2. System displays view teachers, students, courses, venues, preferences and clashes. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while view resources.
Frequency of Use:	Frequently used when the details is required to viewed.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.46 View Teachers

Table 4. 42 View teachers

ID:	46
Title:	View teachers
Description:	Admin is allowed to view teachers.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ view teachers “ from the view resources. 2. System displays teacher’ s detail. 3. Admin view details.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while viewing.
Frequency of Use:	Frequently used when the details is required to viewed.

Technology and data variations list:	Information is modified by using keyboard.
---	--

4.3.2.47 View Students

Table 4. 43 view students

ID:	47
Title:	View students
Description:	Admin is allowed to view students.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	1. Admin selects “ view students “ from the view resources. 2. System displays student’ s details. 3. Admin view details.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while viewing.
Frequency of Use:	Frequently used when the details is required to viewed.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.48 View Courses

Table 4. 44 View courses

ID:	48
Title:	View courses
Description:	Admin is allowed to view courses.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	1. Admin selects “ view courses “ from the view resources. 2. System displays courses details. 3. Admin view details.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while viewing.
Frequency of Use:	Frequently used when the details is required to viewed.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.49 View Venues

Table 4. 45 view venues

ID:	49
Title:	View venues
Description:	Admin is allowed to view venues.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	1. Admin selects “ view venues “ from the view resources. 2. System displays venues details. 3. Admin view details.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while viewing.
Frequency of Use:	Frequently used when the details is required to viewed.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.50 View Clashes

Table 4. 46 View clashes

ID:	50
Title:	View clashes
Description:	Admin is allowed to view clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	1. Admin selects “ view clashes “ from the view resources. 2. System displays clashes details. 3. Admin view details.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while viewing.
Frequency of Use:	Frequently used when the details is required to viewed.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.51 View Preferences

Table 4. 47 View preferences

ID:	51
Title:	View preferences
Description:	Admin is allowed to view preferences.
Primary Actor:	Admin

Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully viewed.
Main Success Scenario:	1. Admin selects “ view preferences “ from the view resources. 2. System displays preferences details. 3. Admin view details.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while viewing.
Frequency of Use:	Frequently used when the details is required to viewed.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.52 Delete Resources

Table 4. 48 Delete resources

ID:	52
Title:	Delete resources
Description:	Admin is allowed to delete resources. Admin can delete teachers, students, courses, venues, preferences and clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully deleted.
Main Success Scenario:	1. Admin selects “ delete resources “ from the menu. 2. System displays delete teachers, students, courses, venues, preferences and clashes. 3. Admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while delete resources.
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.53 Delete Teachers

Table 4. 49 Delete teachers

ID:	53
Title:	Delete teachers
Description:	Admin is allowed to delete teachers.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.

Post conditions:	All details were successfully deleted.
Main Success Scenario:	1. Admin selects “ delete teachers “ from the delete resources. 2. System displays teacher’ s detail. 3. Admin view details and presses delete button. 4. System deleted it successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while deleting.
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.54 Delete Students

Table 4. 50 Delete students

ID:	54
Title:	Delete students
Description:	Admin is allowed to delete students.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully deleted.
Main Success Scenario:	1. Admin selects “ delete students “ from the delete resources. 2. System displays student’ s details. 3. Admin view details and presses delete button. 4. System deleted it successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while deleting.
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.55 Delete Courses

Table 4. 51 Delete courses

ID:	55
Title:	Delete courses
Description:	Admin is allowed to delete courses.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully deleted.
Main Success Scenario:	1. Admin selects “ delete courses “ from the delete resources. 2. System displays courses details.

	3. Admin view details and presses delete button. 4. System deleted it successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while deleting.
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.56 Delete Venues

Table 4. 52 Delete venues

ID:	56
Title:	Delete venues
Description:	Admin is allowed to delete venues.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully deleted.
Main Success Scenario:	1. Admin selects “ delete venues “ from the delete resources. 2. System displays venues details. 3. Admin view details and presses delete button. 4. System deleted it successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while deleting.
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.57 Delete Clashes

Table 4. 53 Delete clashes

ID:	57
Title:	Delete clashes
Description:	Admin is allowed to delete clashes.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully deleted.
Main Success Scenario:	1. Admin selects “ delete clashes “ from the delete resources. 2. System displays clashes details. 3. Admin view details and presses delete button. 4. System deleted it successfully.
Extensions:	2a. May be system took too long to respond
Special	There should be no redundancy of data while deleting.

requirements:	
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.58 Delete Preferences

Table 4. 54 Delete preferences

ID:	58
Title:	Delete preferences
Description:	Admin is allowed to delete preferences.
Primary Actor:	Admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	All details were successfully deleted.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin selects “ delete preferences “ from the delete resources. 2. System displays preferences details. 3. Admin view details and presses delete button. 4. System deleted it successfully.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while deleting.
Frequency of Use:	Frequently used when the details is required to deleted.
Technology and data variations list:	Information is modified by using keyboard.

4.3.2.59 Manage Admin

Table 4. 55 Manage Admin

ID:	59
Title:	Manage admin
Description:	Super admin is allowed to manage admins. Super admin can add, delete, view and update any admin’ s details.
Primary Actor:	Super admin
Pre-conditions:	He/she should have a valid authentication details.
Post conditions:	Admin’ s details were successfully added/updated/removed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Super admin selects “ manage admin" from the menu. 2. System displays add, update and delete admin’ s information 3. Super admin selects one of the options. 4. System displays related fields.
Extensions:	2a. May be system took too long to respond
Special requirements:	There should be no redundancy of data while managing admins.
Frequency of Use:	Frequently used when admin details is required to modified/add/removed.

Technology and data variations list:	Information is modified by using keyboard.
---	--

4.3.2.60 Add Admin

Table 4. 56 Add Admin

ID:	60
Title:	Add admin
Description:	Super admin is allowed to manage admins. Super admin can add, delete, view and update any admin' s details.
Primary Actor:	Super admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Admin' s details were successfully added.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Super admin selects “ add admin” from the manage admin. 2. System displays information to be added i.e. first name, last name, contact details, address etc. 3. Super admin inserted the required information of admin. 4. System checks for all fields and enabled the add button. 5. Super admin presses add button. 6. System displays “ successfully added” message on screen.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. super admin inserted information in invalid format. 5a. super admin inserted the record or part of information that is already in the database i.e. redundant information.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when new admin is hired and his details are required to be added.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.61 Delete Admin

Table 4. 57 Delete Admin

ID:	61
Title:	Delete admin
Description:	Super admin is allowed to manage admins. Super admin can add, delete, view and update any admin' s details.
Primary Actor:	Super admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.

Post conditions:	Admin' s details were successfully removed.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Super admin selects “ delete admin" from the manage admin. 2. System displays enter admin id to be removed 3. Super admin inserted the required information of admin. 4. System displays confirmation message of deletion. 5. Super admin presses confirm button. 6. System displays “ successfully deleted” message on screen.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. super admin inserted invalid ID of admin. 3b. super admin inserted the record or part of information that is already been deleted. 4a. confirmation message took too long to display.
Special requirements:	There should be proper placeholders for all the input fields.
Frequency of Use:	Used only when admin detail is required to be removed from list of employed admins.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.62 Update Admin

Table 4. 58 Update Admin

ID:	62
Title:	Update admin
Description:	Super admin is allowed to manage admins. Super admin can add, delete, view and update any admin' s details.
Primary Actor:	Super admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Admin' s details were successfully updated.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Super admin selects “ update admin" from the manage admin. 2. System displays enter admin id to which you want to update information. 3. Super admin inserted the updated information of admin i.e. his contact number, his marital status etc. 4. System displays confirmation message “ update successful” .
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 2b. super admin inserted invalid ID of admin. 3a. super admin inserted information in invalid format 4a. confirmation message took too long to display.
Special requirements:	There should be proper placeholders for all the input fields.

Frequency of Use:	Used only when admin' s specific detail is required to be updated .
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.63 View Admin

Table 4. 59 View Admin

ID:	63
Title:	View admin
Description:	Super admin is allowed to manage admins. Super admin can add, delete, view and update any admin' s details.
Primary Actor:	Super admin
Pre-conditions:	He/she should have a valid authentication details and logged into his account.
Post conditions:	Admin' s details were successfully viewed.
Main Success Scenario:	1. Super admin selects “ view admin" from the manage admin. 2. System displays the information. 3. Super admin viewed the details.
Extensions:	2a. System took too long to respond.
Special requirements:	There should be proper view for all the input fields.
Frequency of Use:	Used only when admin' s specific detail is required to be viewed.
Priority:	Maximum priority
Technology and data variations list:	Information is added by using keyboard.

4.3.2.64 Generate Timetable

Table 4. 60 Generate Timetable

ID:	64
Title:	Generate timetable
Description:	It is admin' s responsibility to generate timetable that should be free of any clashes i.e. student, venue, teacher, course clashes. Timetable should also take teacher' s preferences into account.
Primary Actor:	Admin
Pre-conditions:	He/she should have all the courses and number of enrolled students and their semesters.
Post conditions:	Timetable should be generated that must be free of any type of clashes.
Main Success Scenario:	1. Admin clicks generate timetable from menu. 2. System loaded courses, venues, students, and teachers.

	<ol style="list-style-type: none"> 3. Admin selects number of timetable to be generated. 4. System displays “ successfully generated” message. 5. Admin can view different timetables. 6. System asks to select three timetables. 7. Admin selects timetable. 8. System displays “ successfully selected” message. 9. Admin now send these to teachers for voting. 10. System displays “ successfully send “ message.
Extensions:	<ol style="list-style-type: none"> 2a. System took too long to respond. 3a. Admin is unable to confirm. 4a. System shows timeout message due to confirmation delay by admin. 6a. System took too long to respond. 7a. Admin is unable to select. 9a. System took too long to respond.
Special requirements:	There should be proper input fields and there should not be any data redundancy , unique ID’ S should be assigned to teachers and students to avoid security concerns i.e. authentication
Frequency of Use:	More frequently used when to generate the time table at start of each session.
Technology and data variations list:	Fields should be entered using keyboard.

4.3.2.65 voting for timetable

Table 4. 61 Voting for timetable

ID:	65
Title:	Voting for timetable
Description:	Teacher can vote for the timetable that is best suited to him/her.
Primary Actor:	Teacher
Pre-conditions:	He/she should be notified with all the versions of timetable generated by admin.
Post conditions:	Voting for best suited timetable was done.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin sends all versions of timetable generated to all the teachers. 2. System displays confirmation message “ successfully sent” 3. A teacher views all the versions of timetable. 4. System displays options to vote for the viewed timetable. 5. Teacher sees the time table best suited to him/her and give votes to all versions of timetable accordingly. 6. System sends voting response from teacher to Admin.
Extensions:	<ol style="list-style-type: none"> 2a. System displays network error message. 3a. Teacher could not view all the versions of timetable because of weak internet signals.

	6a. System could not send voting response to admin because of poor network.
Special requirements:	Voting should be opened for particular timespan and results will get locked after expiry of timespan and timetable with highest votes will be finalized.
Frequency of Use:	Used frequently because teacher have to vote for timetables.
Technology and data variations list:	Voting should be done using mouse to give stars to best suited timetable to him/her.

4.3.2.66 Report Clashes

Table 4. 62 Report Clashes

ID:	66
Title:	Report clashes
Description:	As student will see the time table he/she can report clash to administration.
Primary Actor:	Student
Pre-conditions:	Students were notified with finalized version of timetable to the students, teachers of the department.
Post conditions:	Student successfully reported the clashes/clash.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Student found clash in his/her time table. 2. System displays form to report clash. 3. Student report clash to Admin via filling form 4. Student then press submit button. 5. Notification than send to Admin. 6. System displays confirmation message “ successfully reported” .
Extensions:	4a. System displays timespan to report clashes has been expired you cannot proceed.
Special requirements:	There should be proper timespan to report clash/clashes. After expiry of timespan, no request of clashes will be entertained.
Frequency of Use:	Used only when the students have clashes and they have reported the clashes to be resolved in particular timespan
Technology and data variations list:	Notification should be sent via keyboard.

4.3.2.67 Add preferences

Table 4. 63 Add preferences

ID:	67
Title:	Add preferences
Description:	Teacher can add his/her preferences

Primary Actor:	Teacher
Pre-conditions:	Teachers successfully logged in.
Post conditions:	Teacher successfully added the preferences.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Teacher wants to add preferences. 2. System displays add preferences form. 3. Teacher can add preferences by filling form data. 4. Teacher then presses submit button to confirm. 5. Notification than send to Admin. 6. System displays confirmation message “ successfully added” .
Extensions:	4a. System displays timespan to add preferences has been expired you cannot proceed.
Special requirements:	There should be proper timespan to add preferences. After expiry of timespan, no request of preferences will be entertained.
Frequency of Use:	Used only when the teachers have some preferences.
Technology and data variations list:	Information should be sent via keyboard.

4.3.2.68 Evaluate Timetable

Table 4. 64 Evaluate timetable

ID:	68
Title:	Evaluate Timetable
Description:	Admin can evaluate timetable
Primary Actor:	Admin
Pre-conditions:	Admin successfully logged in.
Post conditions:	Admin successfully evaluated.
Main Success Scenario:	<ol style="list-style-type: none"> 1. Admin wants to review voting result. 2. System displays voting result data 3. Admin selects the timetable with the highest votes. 4. Admin send the selected timetable to all students and teachers. 5. System displays “ Sent successfully” message.
Extensions:	4a. System displays timespan to add preferences has been expired you cannot proceed.
Special requirements:	There should be proper timespan to add preferences. After expiry of timespan, no request of preferences will be entertained.
Frequency of Use:	Used only when the teachers have some preferences.
Technology and data variations list:	Information should be sent via keyboard.

4.3.3 Entity Relationship Diagram (ERD)

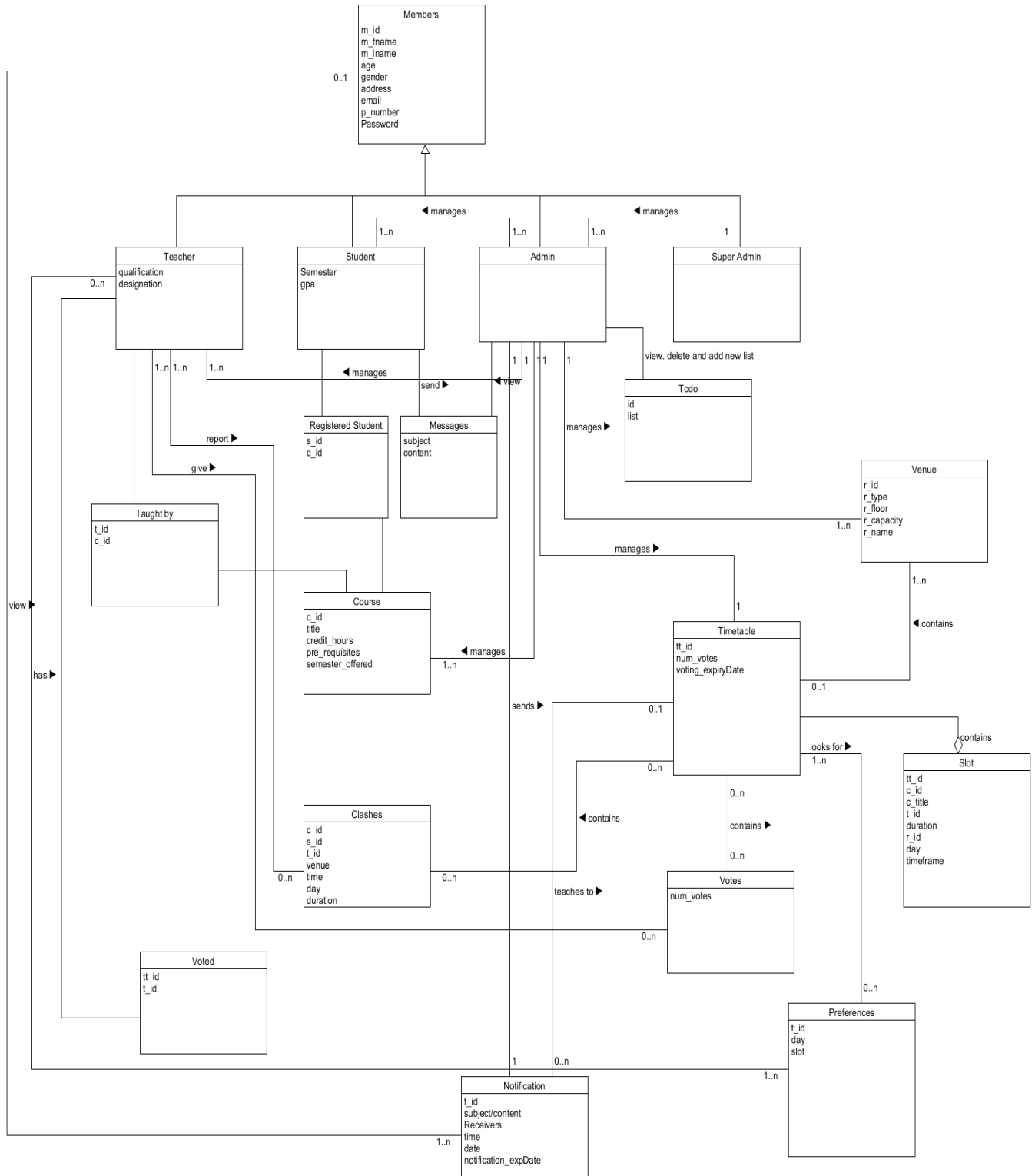


Figure 4. 4: Entity Relationship Diagram

4.3.4 Class Diagram

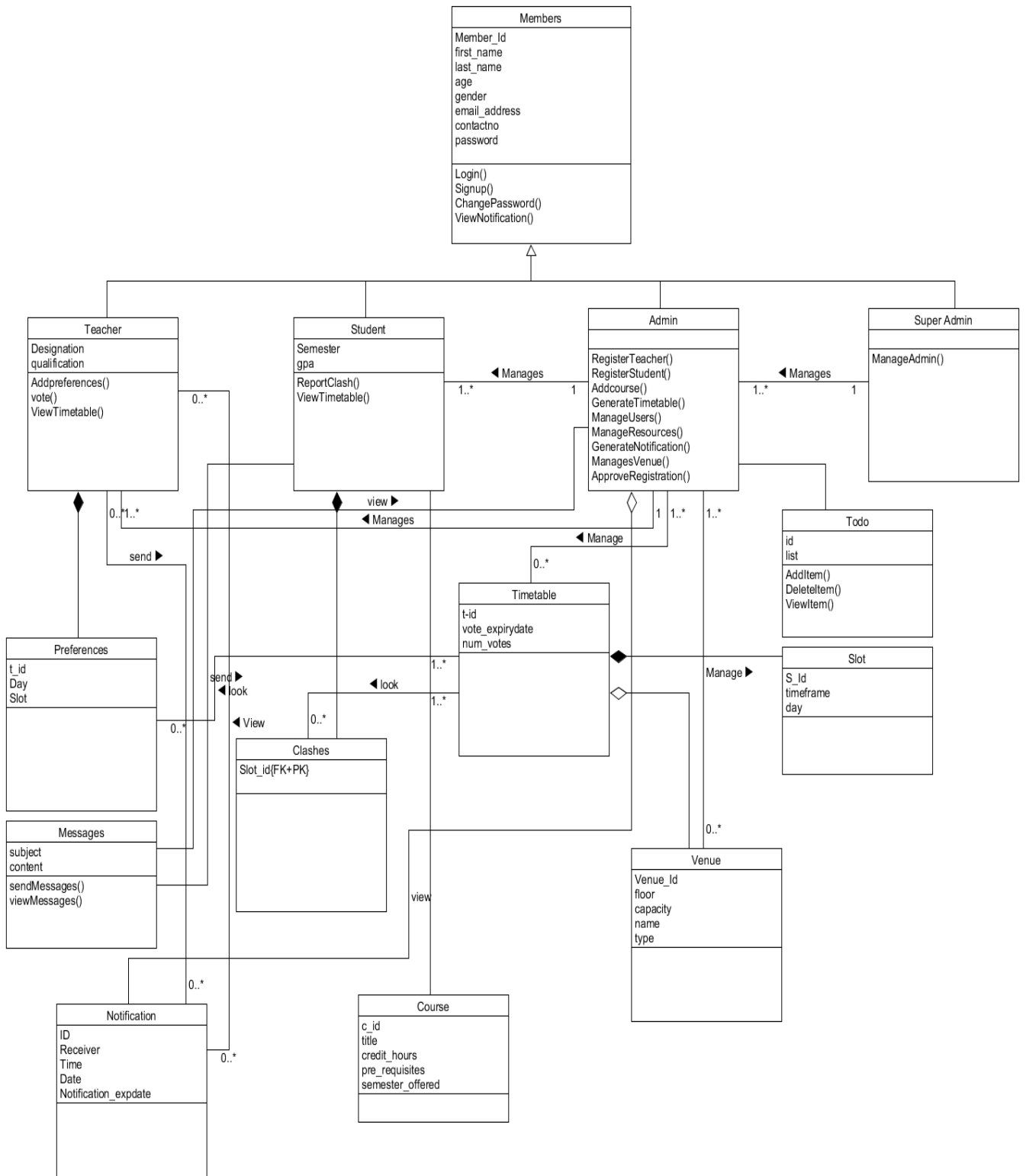


Figure 4. 5: Class Diagram

4.3.5 System Sequence Diagram

4.3.5.1 Login of Super Admin

Main Success Scenario:

1. Super Admin clicks on “ sign in” button.
2. System displays “ username” and “ password” field.
3. Super Admin fills these fields.
4. Super Admin clicks "Login" button.
5. System displays all details.

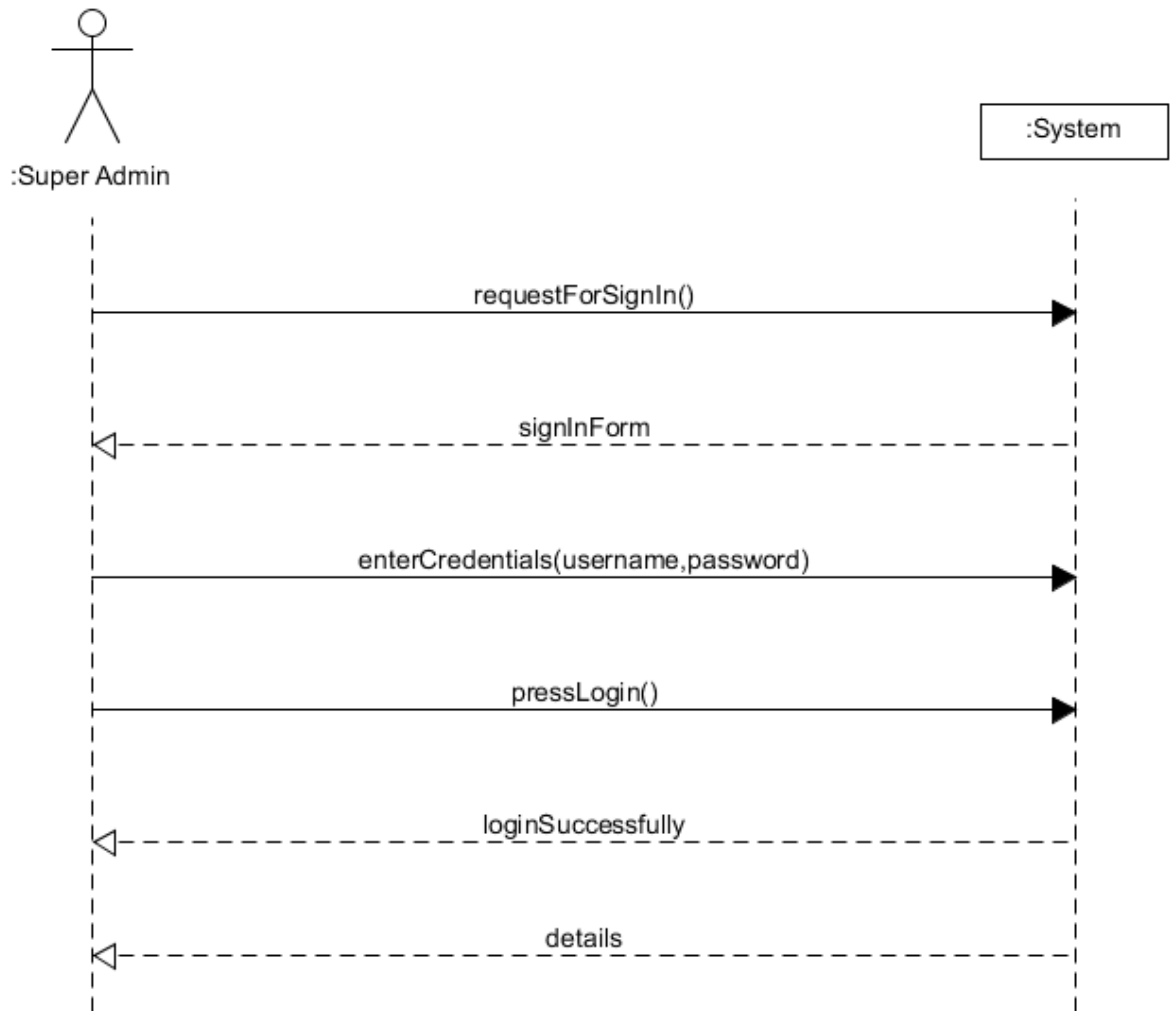


Figure 4.5. 1 Login of SuperAdmin

4.3.5.2 Login of Admin

Main Success Scenario:

1. Admin clicks on “ sign in” button.
2. System displays “ username” and “ password” field.
3. Admin fills these fields.
4. Admin clicks "Login" button.
5. System displays all details.

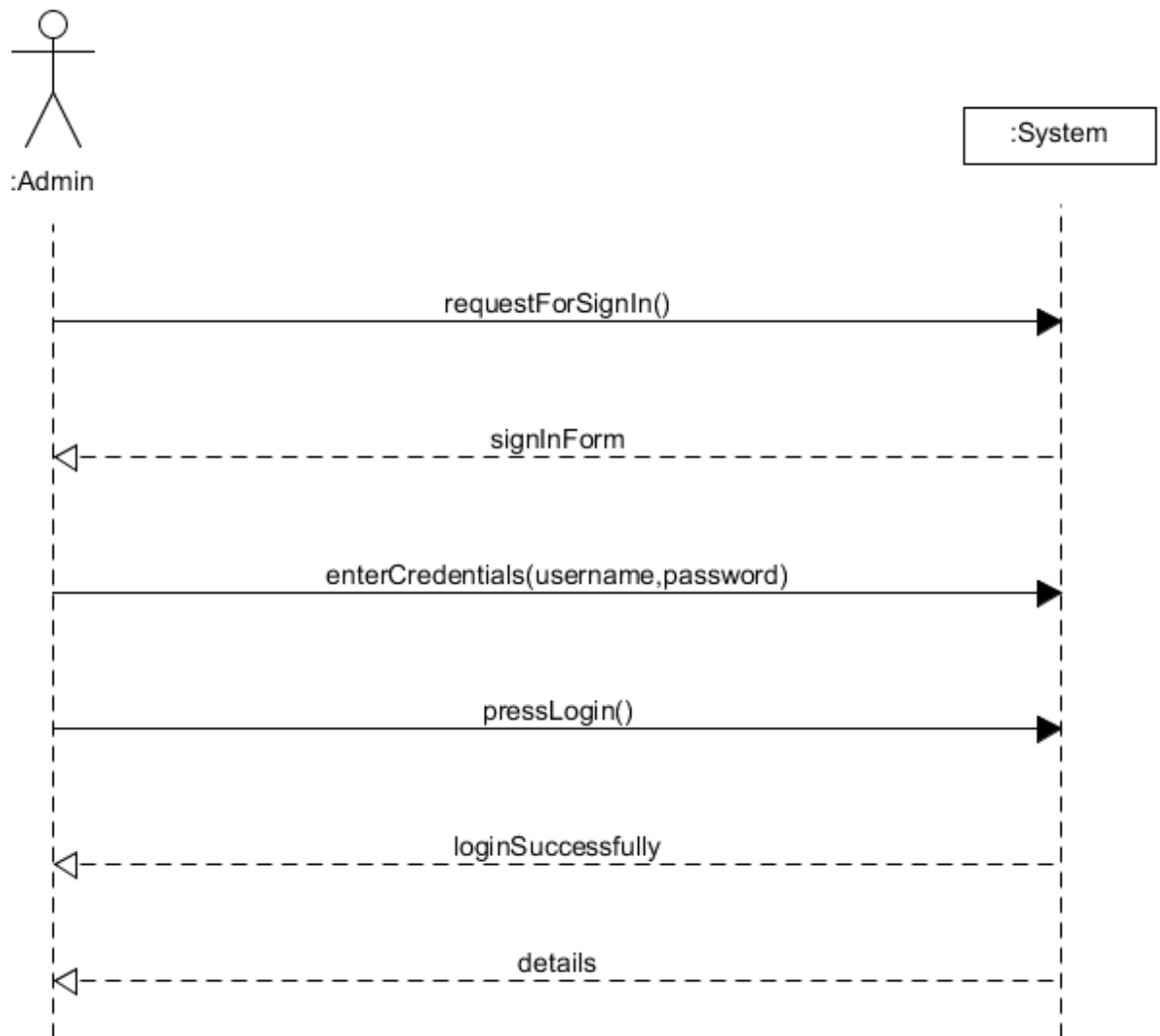


Figure 4.5. 2 Login of Admin

4.3.5.3 Login of Teacher

Main Success Scenario:

1. Teacher clicks on “ sign in” button.
2. System displays “ username” and “ password” field.
3. Teacher fills these fields.
4. Teacher clicks "Login" button.
5. System displays all details.

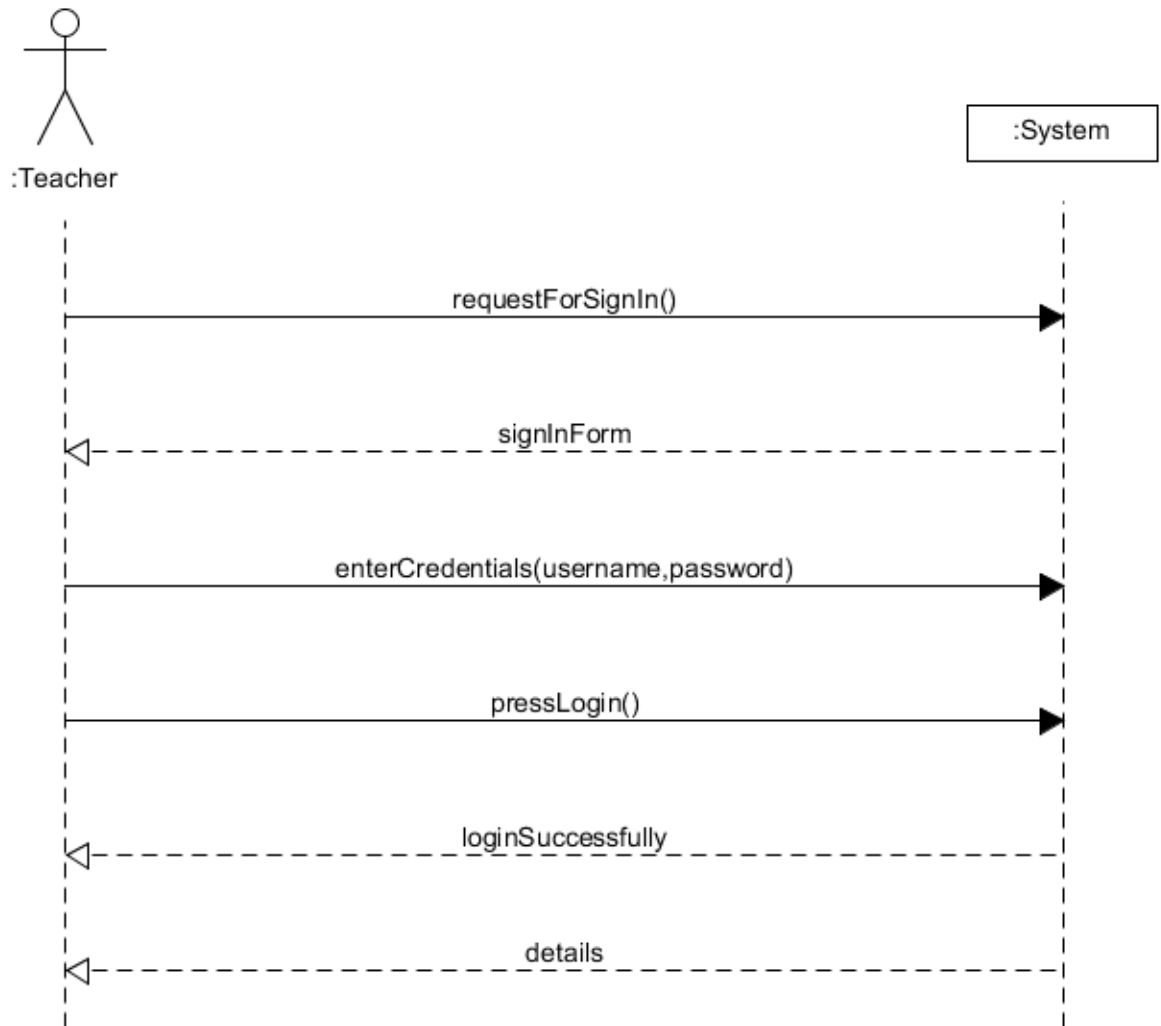


Figure 4.5. 3 Login of Teacher

4.3.5.4 Login of Student

Main Success Scenario:

1. Student clicks on “ sign in” button.
2. System displays “ username” and “ password” field.
3. Student fills these fields.
4. Student clicks "Login" button.
5. System displays all details.

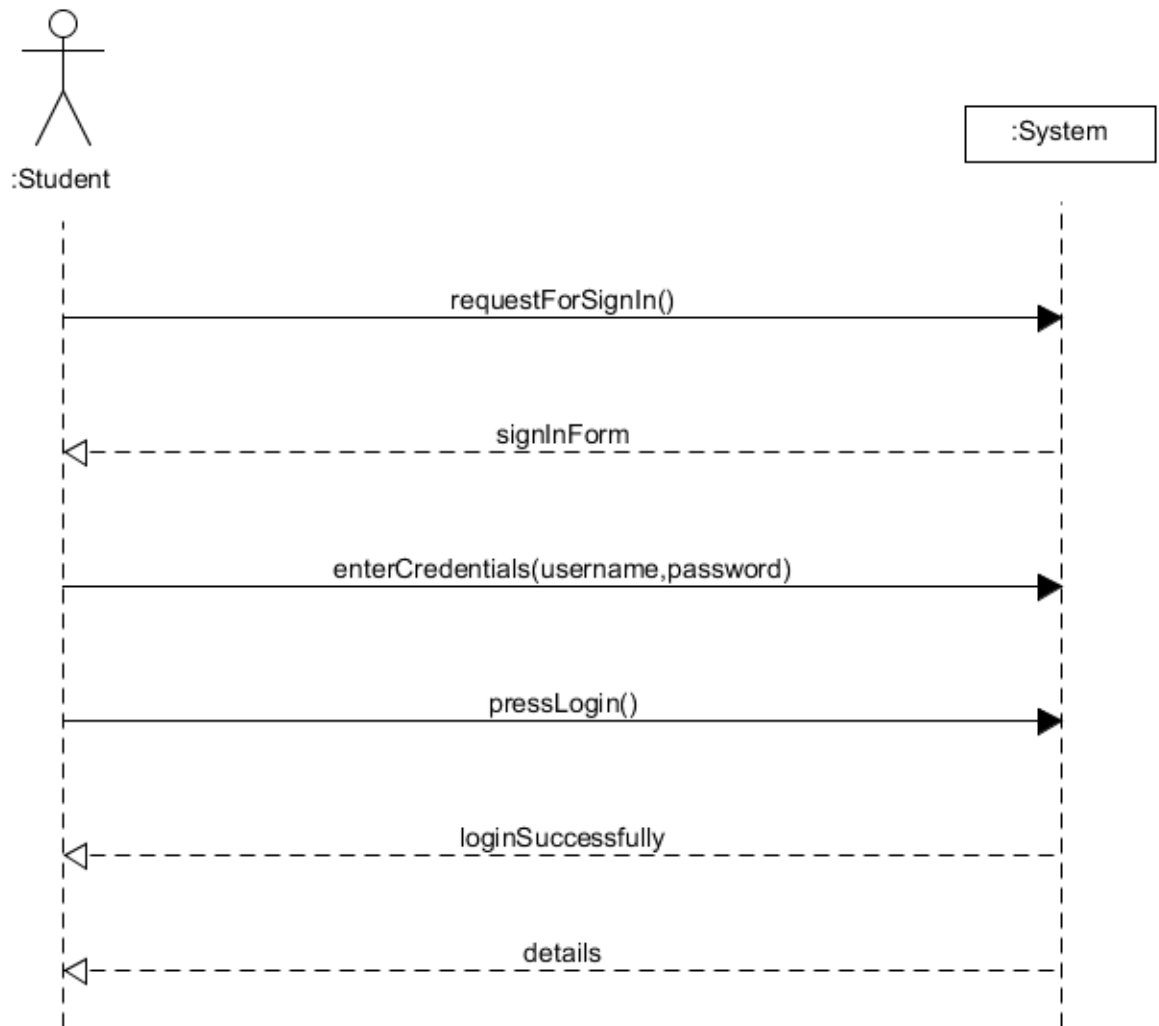


Figure 4.5. 4 Login of Student

4.3.5.5 View Timetable by Super Admin

Main Success Scenario:

1. Super Admin clicks on “ View Timetable” button.
2. System displays the window which shows timetable.

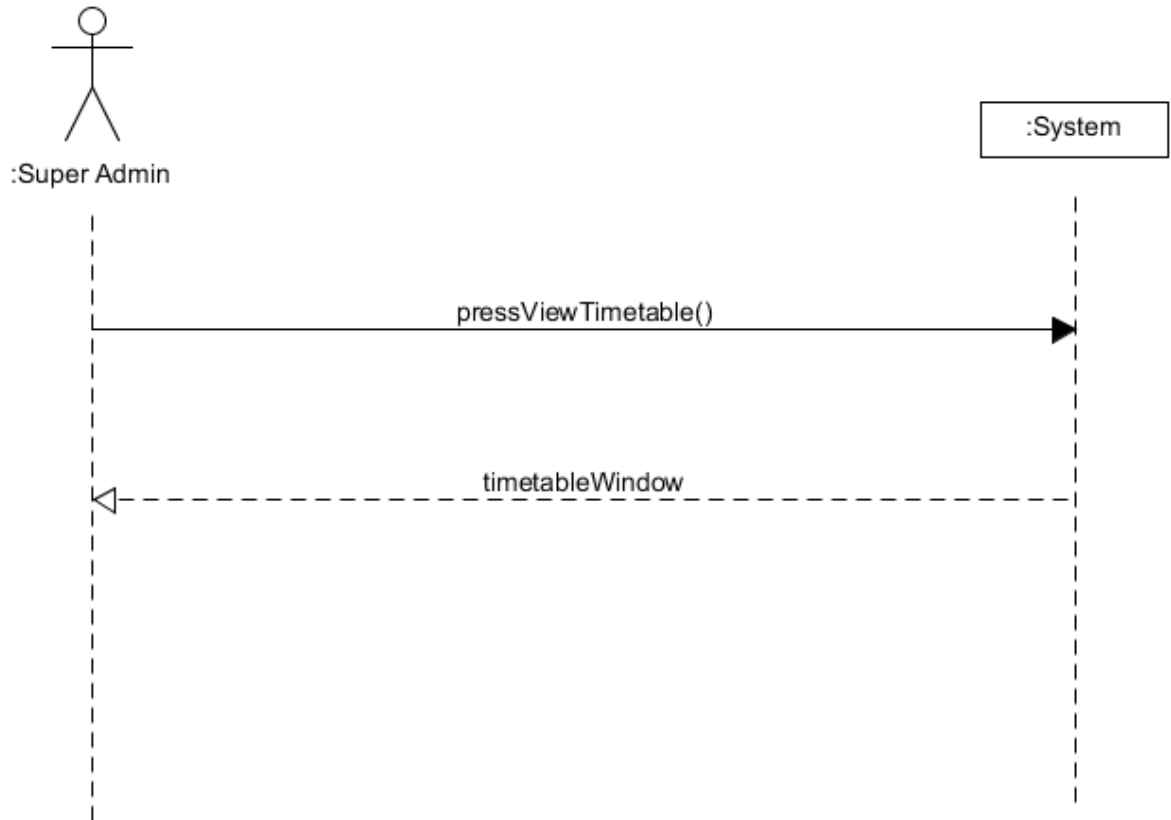


Figure 4.5.5 View Timetable by SuperAdmin

4.3.5.6 View Timetable by Admin

Main Success Scenario:

1. Admin clicks on “ View Timetable” button.
2. System displays the window which shows timetable

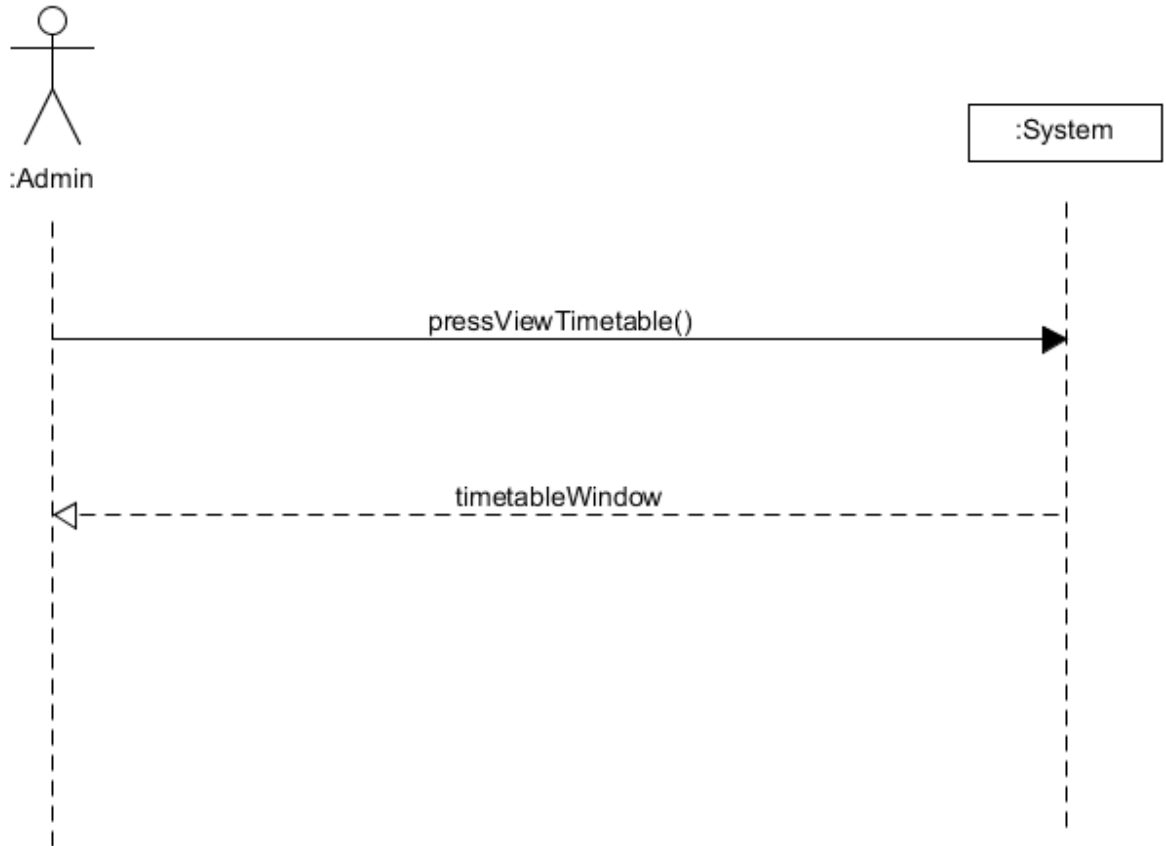


Figure 4.5. 5 View Timetable by Admin

4.3.5.7 View Timetable by Teacher

Main Success Scenario:

1. Teacher clicks on “ View Timetable” button.
2. System displays the window which shows timetable.

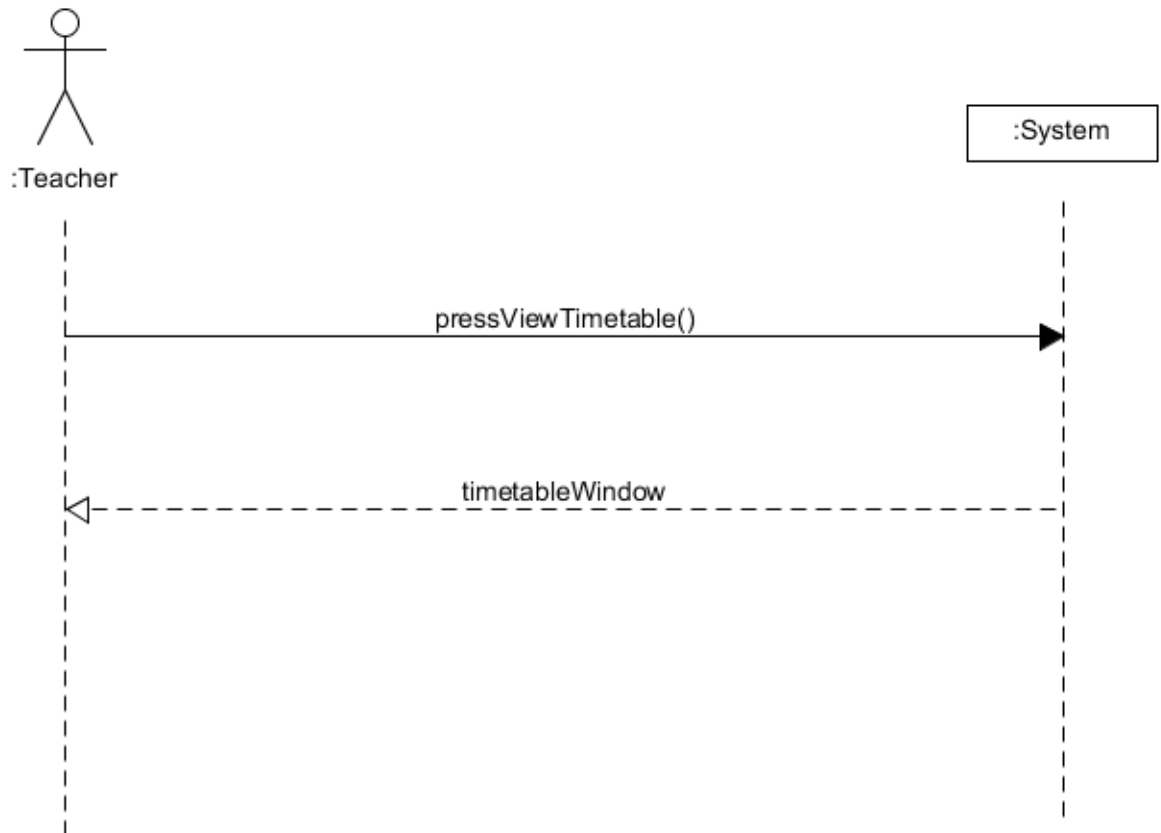


Figure 4.5. 6 View Timetable by Teacher

4.3.5.8 View Timetable by Student

Main Success Scenario:

1. Student clicks on “ View Timetable” button.
2. System displays the window which shows timetable.

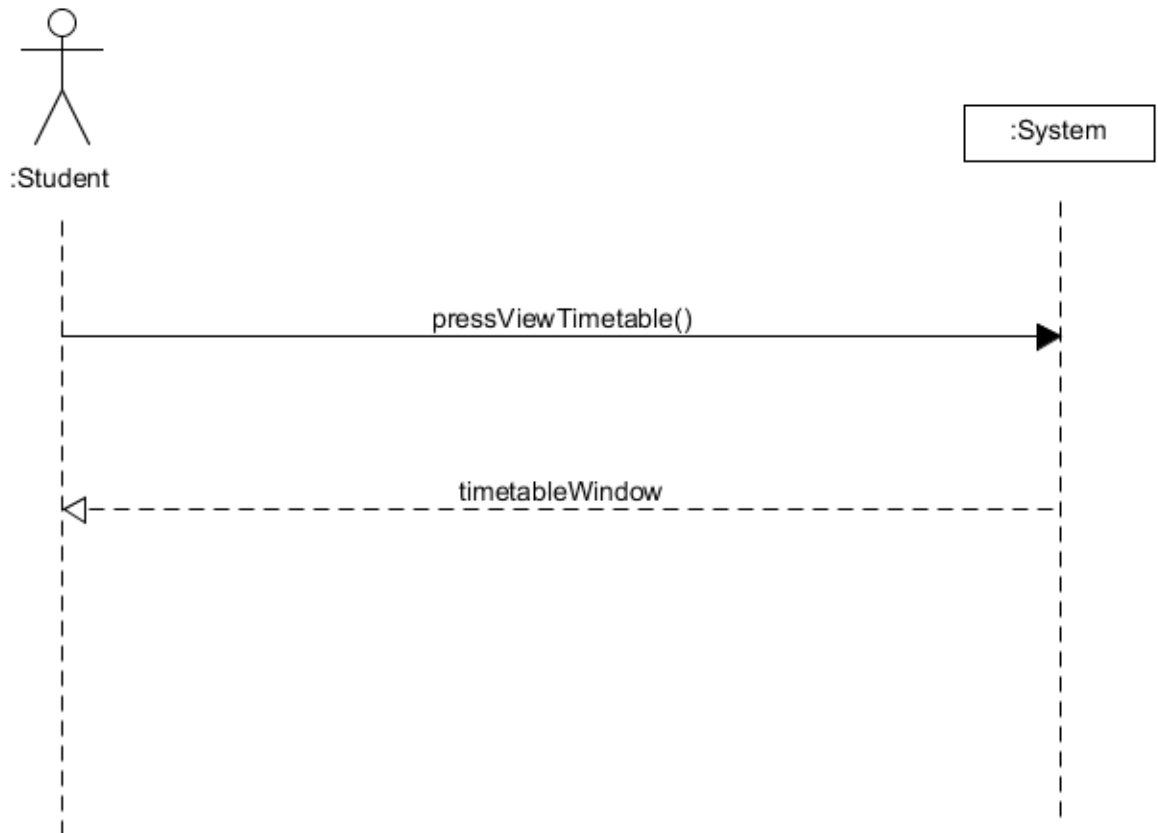


Figure 4.5. 7 View Timetable by student

4.3.5.9 View Notification by Super Admin

Main Success Scenario:

1. Super Admin clicks on view notification button.
2. System displays all notifications.
3. Super Admin clicks on notification from STTM.
4. System displays the notification details.

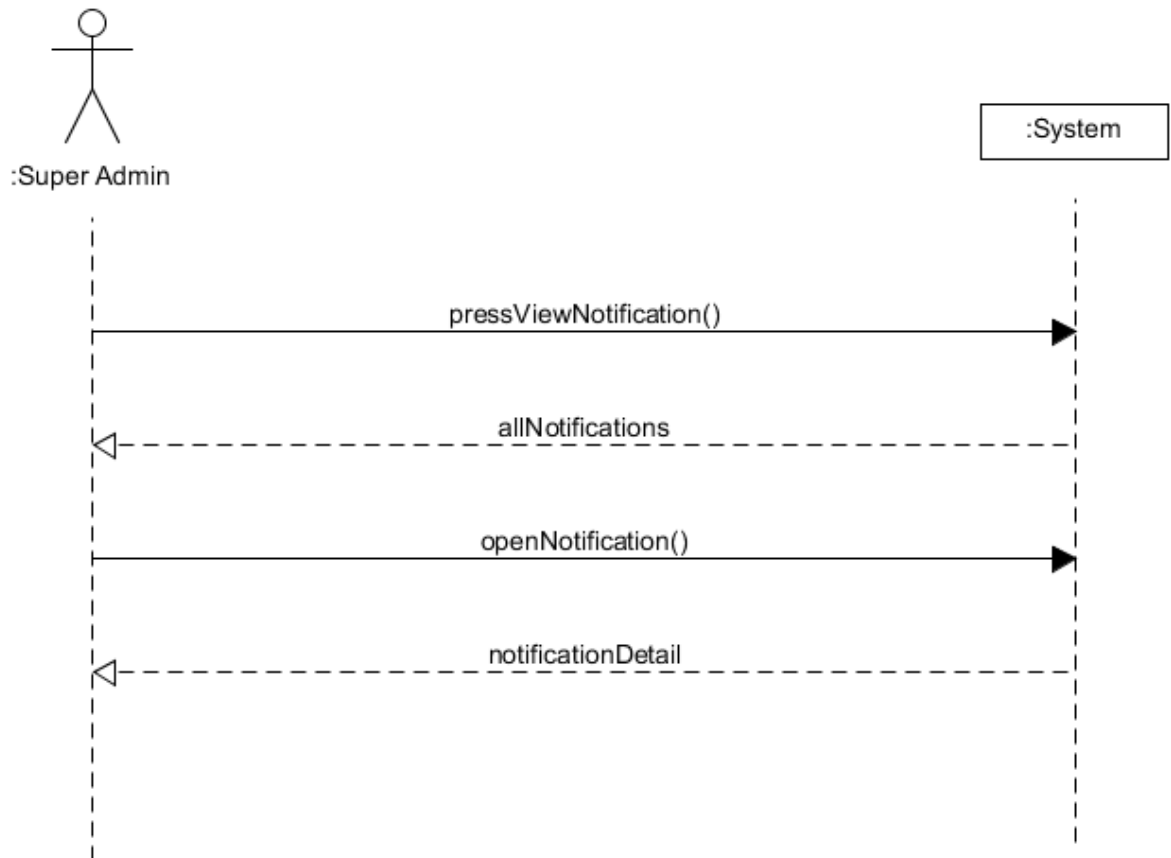


Figure 4.5.9 View Notification by superadmin

4.3.5.10 View Notification by Admin

Main Success Scenario:

1. Admin click on view notification button.
2. System displays all notifications.
3. Admin click on notification from STTM.
4. System displays the notification details.

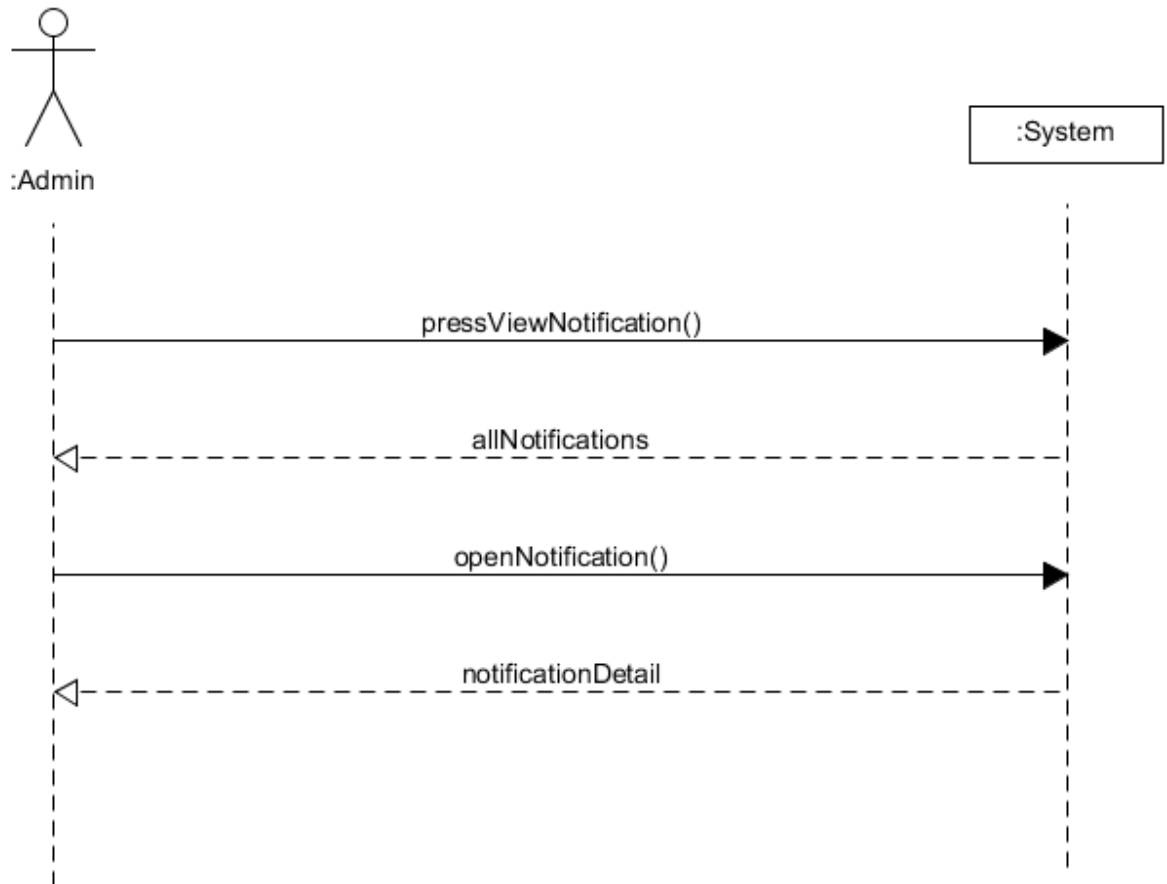


Figure 4.5. 8 View notification by admin

4.3.5.11 View Notification by Teacher

Main Success Scenarios:

1. Teacher clicks on view notification button.
2. System displays all notifications.
3. Teacher clicks on notification from STTM.
4. System displays the notification details.

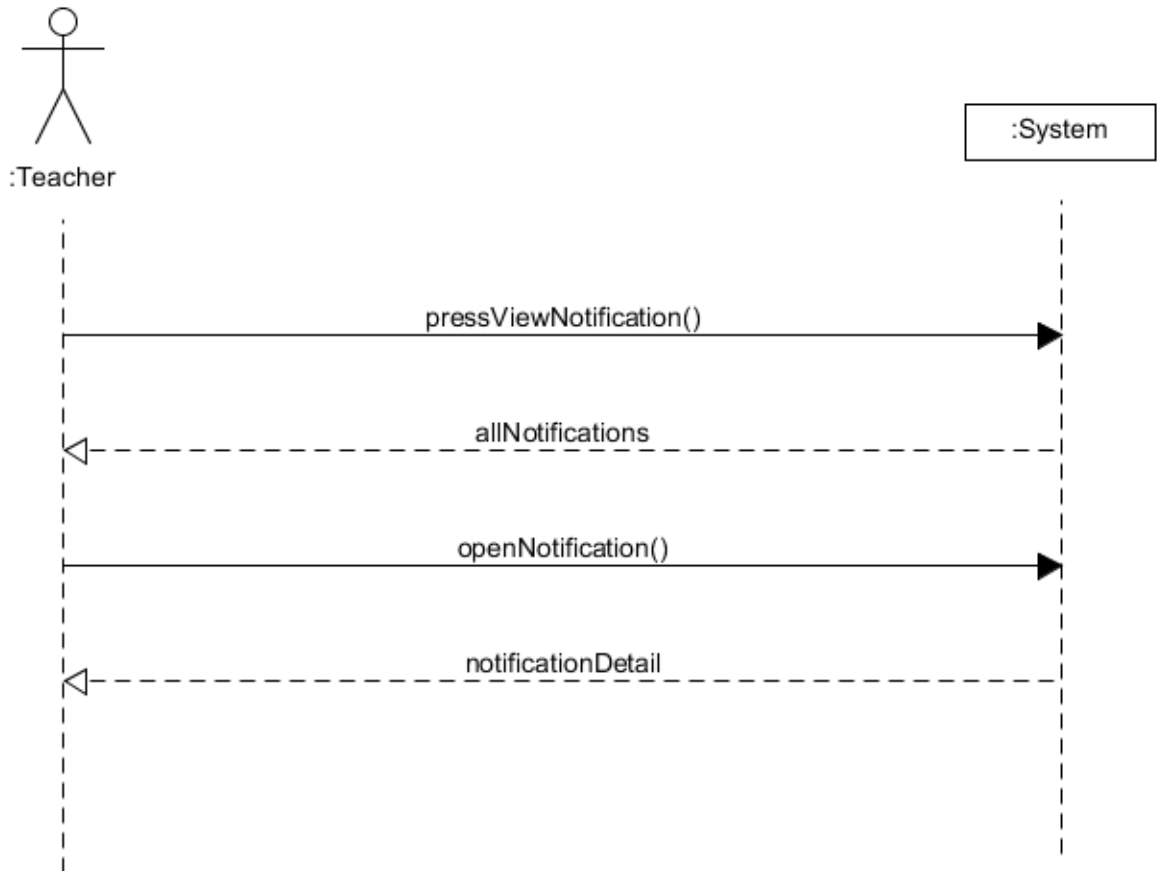


Figure 4.5. 9 View Notification by Teacher

4.3.5.12 View Notification by Student

Main Success Scenario:

1. Student click on view notification button.
2. System displays all notifications.
3. Student click on notification from STTM.
4. System displays the notification details.

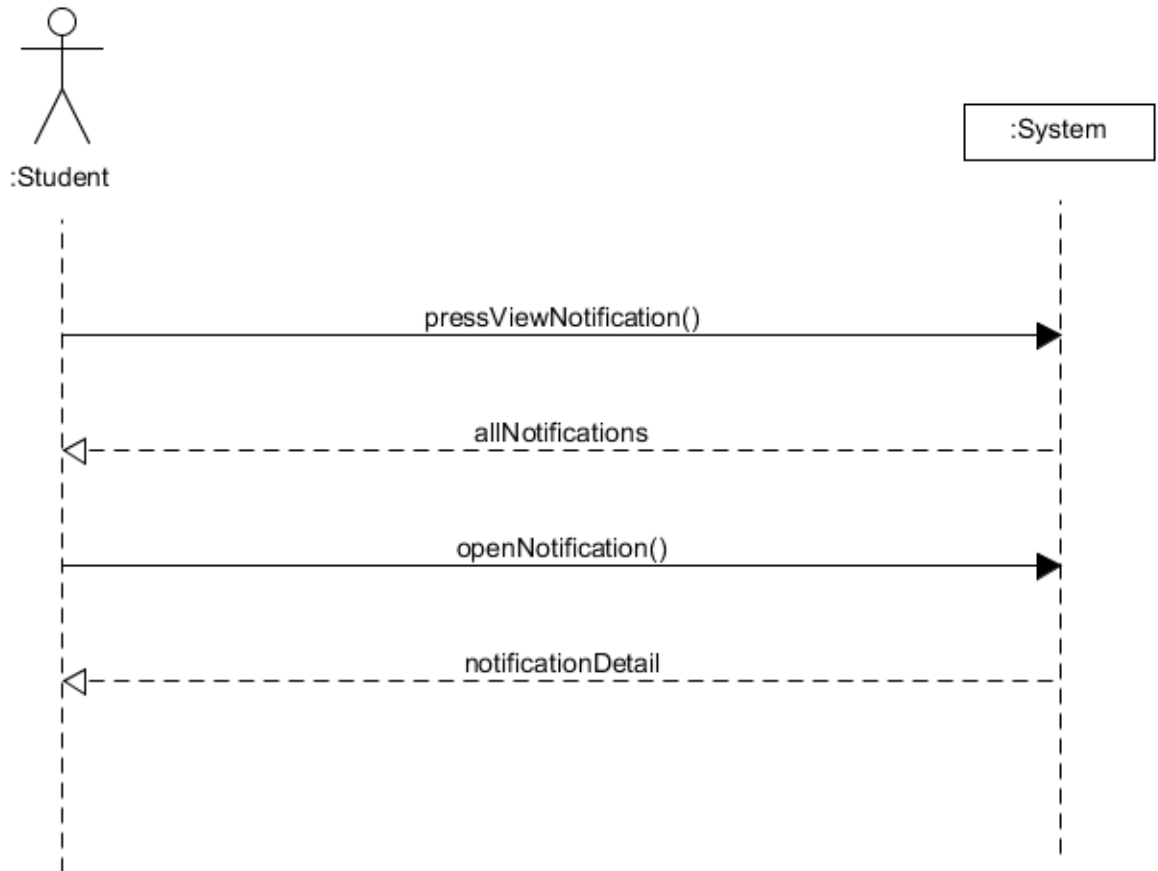


Figure 4.5. 10 View Notification by Student

4.3.5.13 Send Notification by Admin

Main Success Scenario:

1. Admin generates timetable.
2. System displays the timetable.
3. System sends the notification to all register teachers for voting.
4. System opens votes for a certain timeframe.
5. Teacher votes for timetable.
6. System displays the final timetable after voting.
7. System then sends notification to teachers and students both.

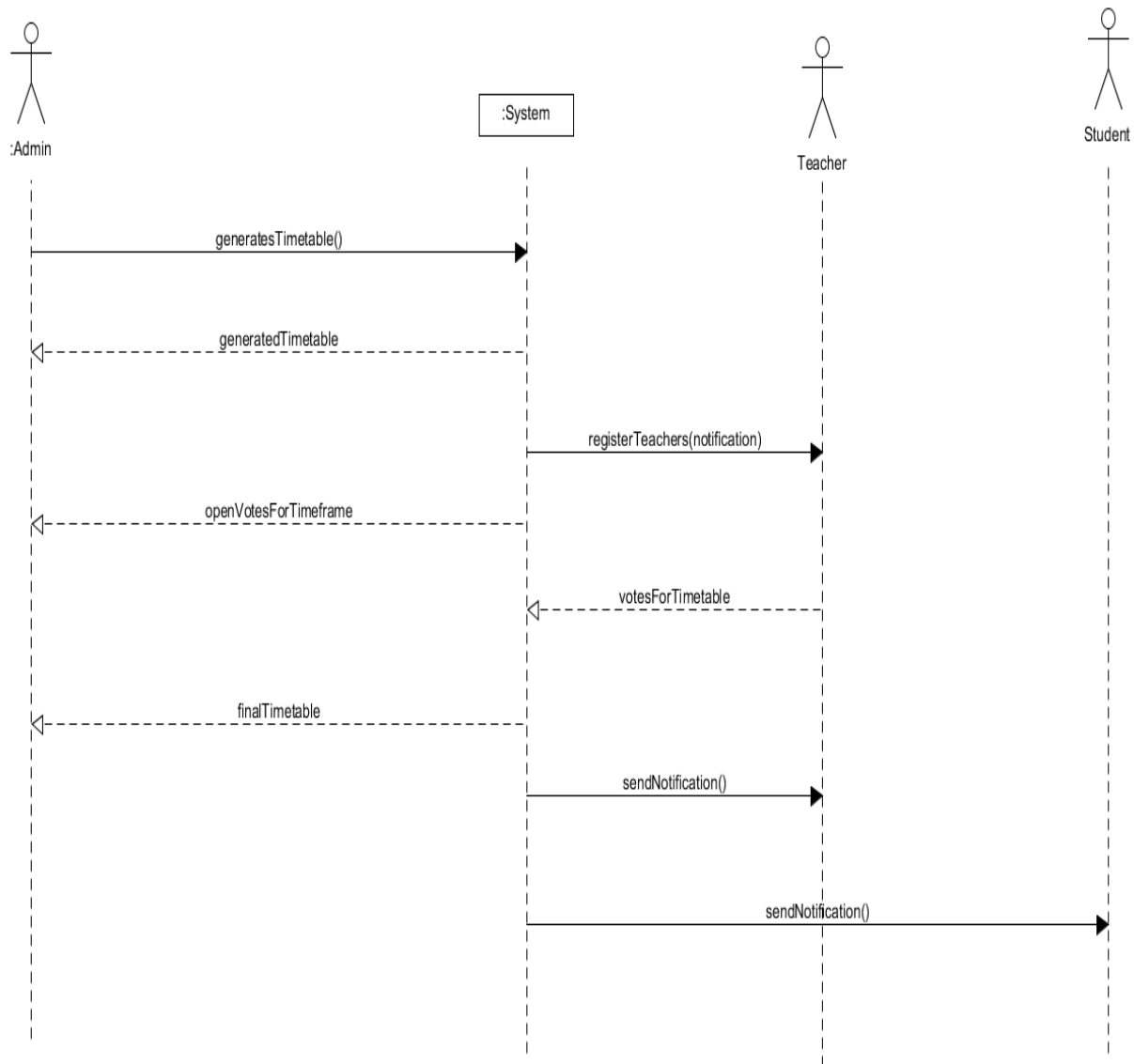


Figure 4.5. 11 Send notification by admin

4.3.5.14 Manage Resources

Main Success Scenario:

1. Admin selects “ manage resources" from the menu.
2. System displays manage teachers, manage students, manage courses, and manage venues.
3. Admin selects one of the options.
4. System displays related fields.

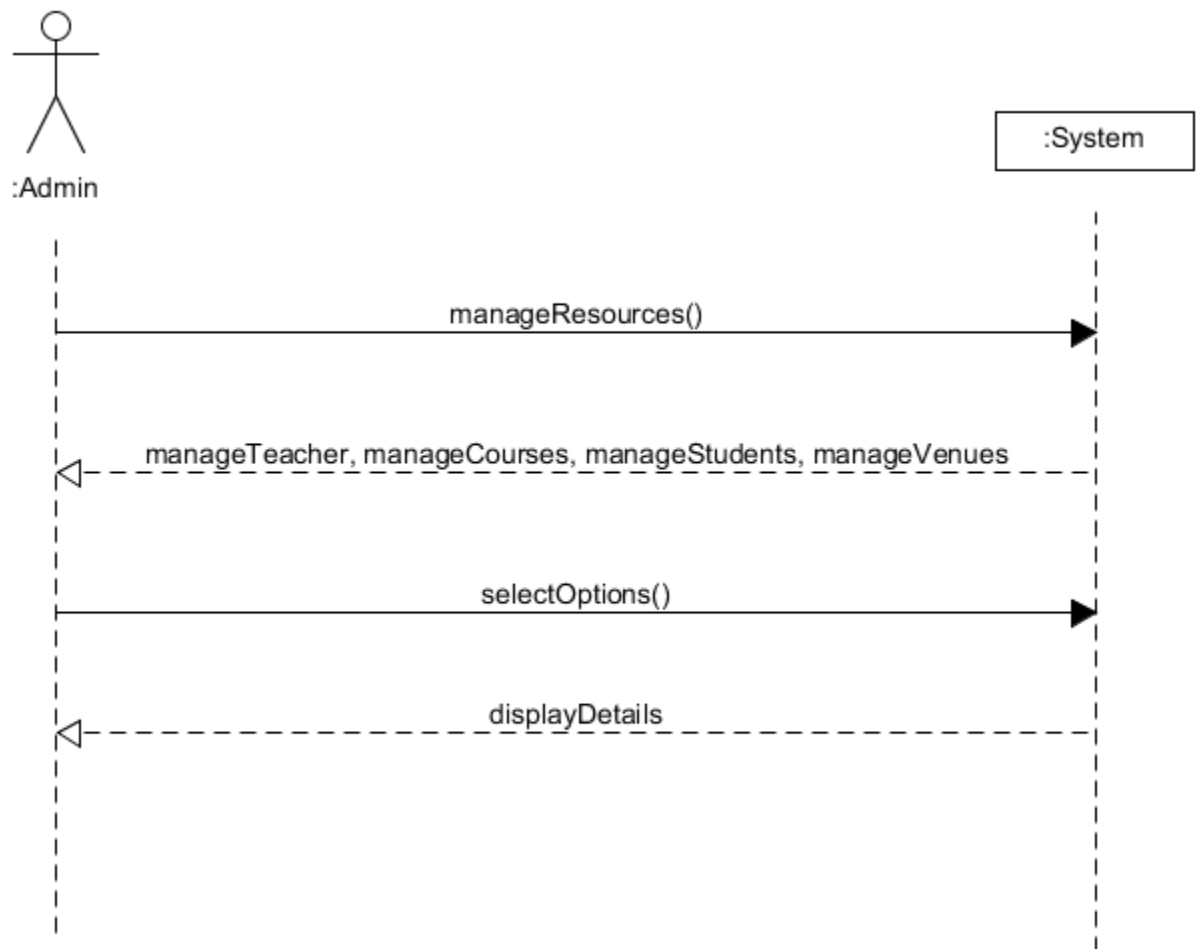


Figure 4.5. 12 Manage Resources

4.3.5.15 Manage Teacher

Main Success Scenario:

1. Admin selects “ manage teachers" from the menu.
2. System displays add teachers and update teachers.

3. Admin selects one of the options.
4. System displays related fields.

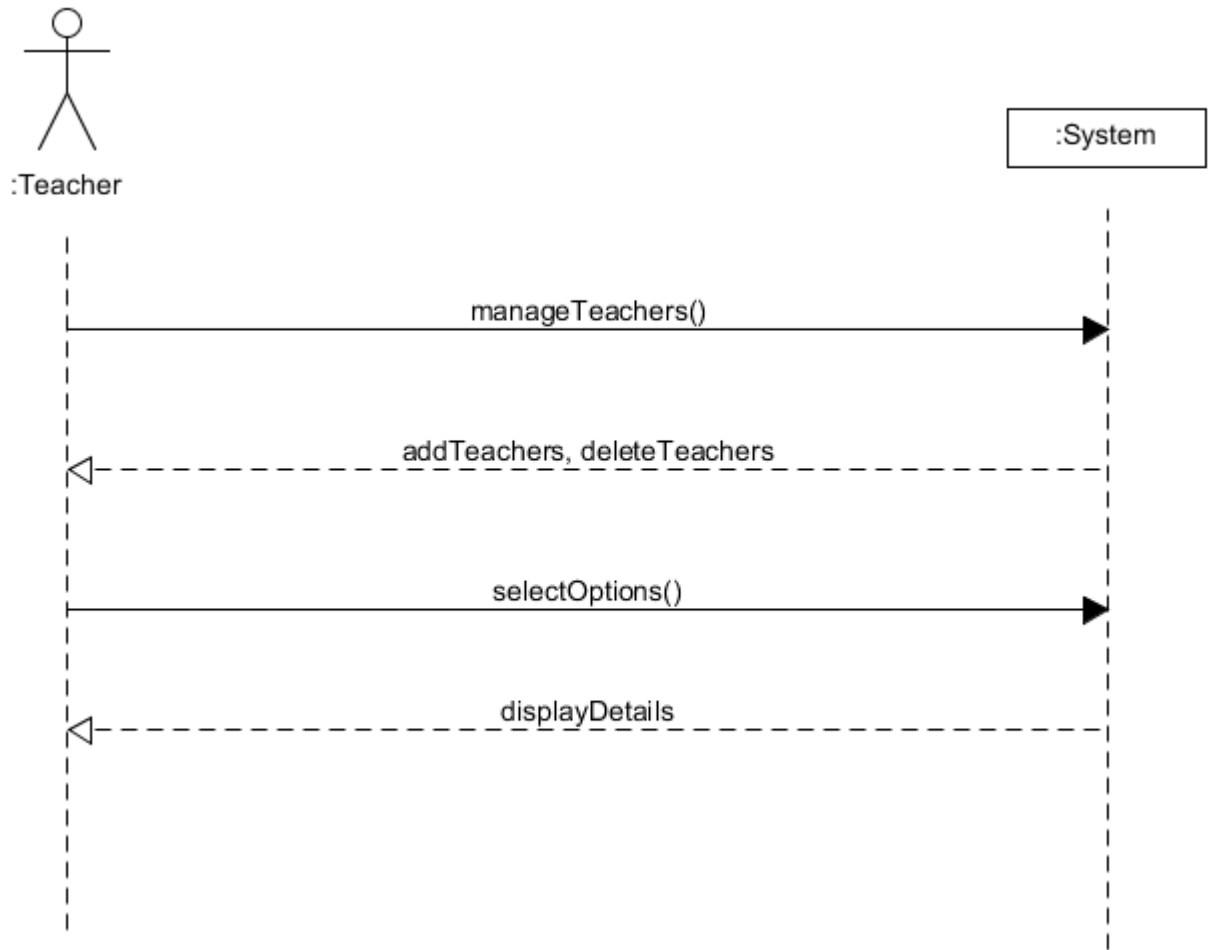


Figure 4.5. 13 Manage Teacher

4.3.5.16 Add Teacher

Main Success Scenario:

1. Admin selects “ add teachers” from the manage teachers.
2. System displays information to be added i.e. first name, last name, id, age, gender, contact details, address etc.
3. Admin inserted the required information of teacher.
4. System checks for all fields and enabled the add button.
5. Admin presses add button.
6. System displays “ successfully added” message on screen.

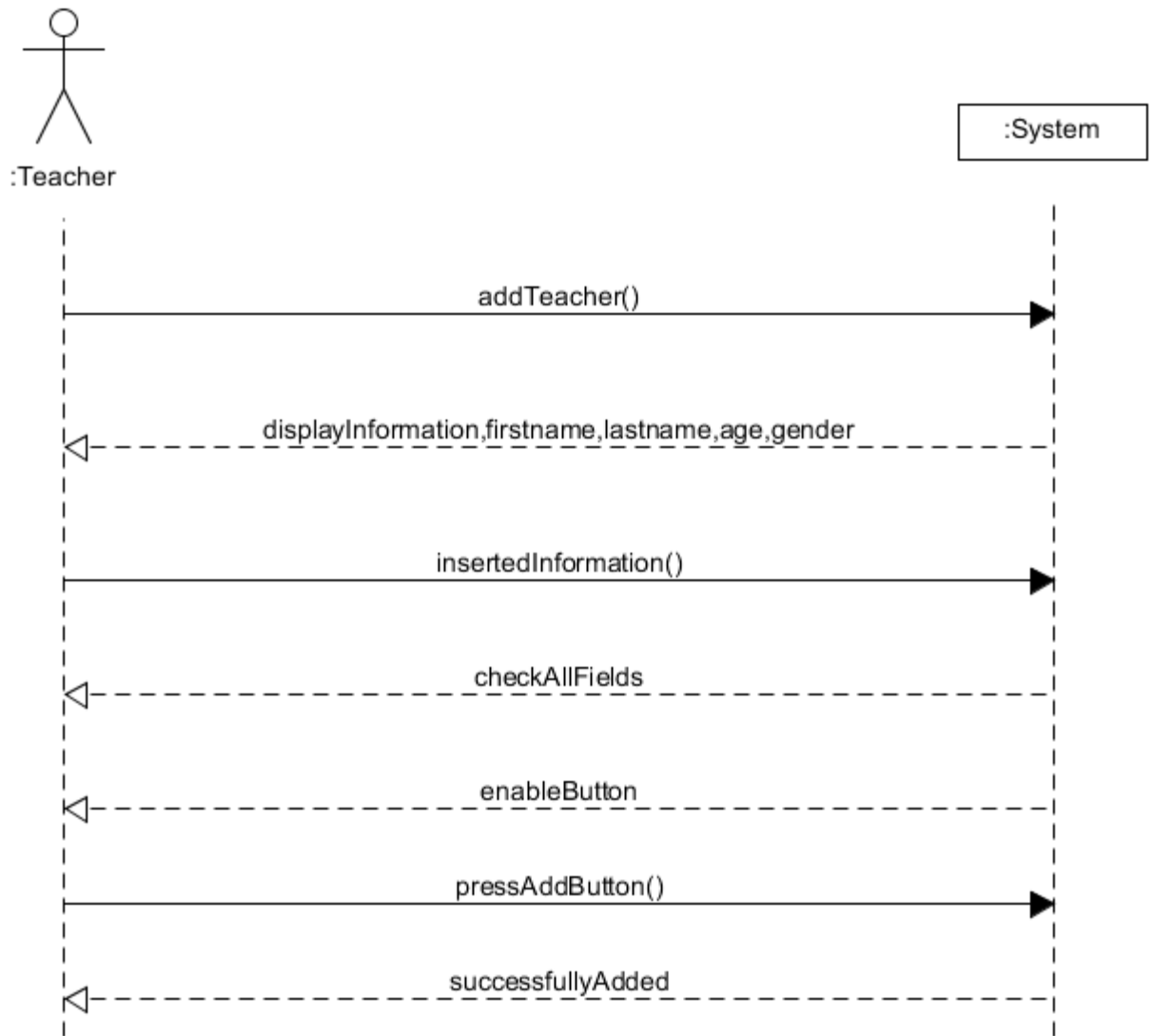


Figure 4.5. 14 Add teacher

4.3.5.17 Update Teacher

Main Success Scenario:

1. Admin selects “ update teachers” from the manage teachers.
2. System displays the teacher information.
3. Admin inserted the updated information of teacher i.e. his contact number, his address, designation etc.
4. System displays confirmation message “ updated successfully” .

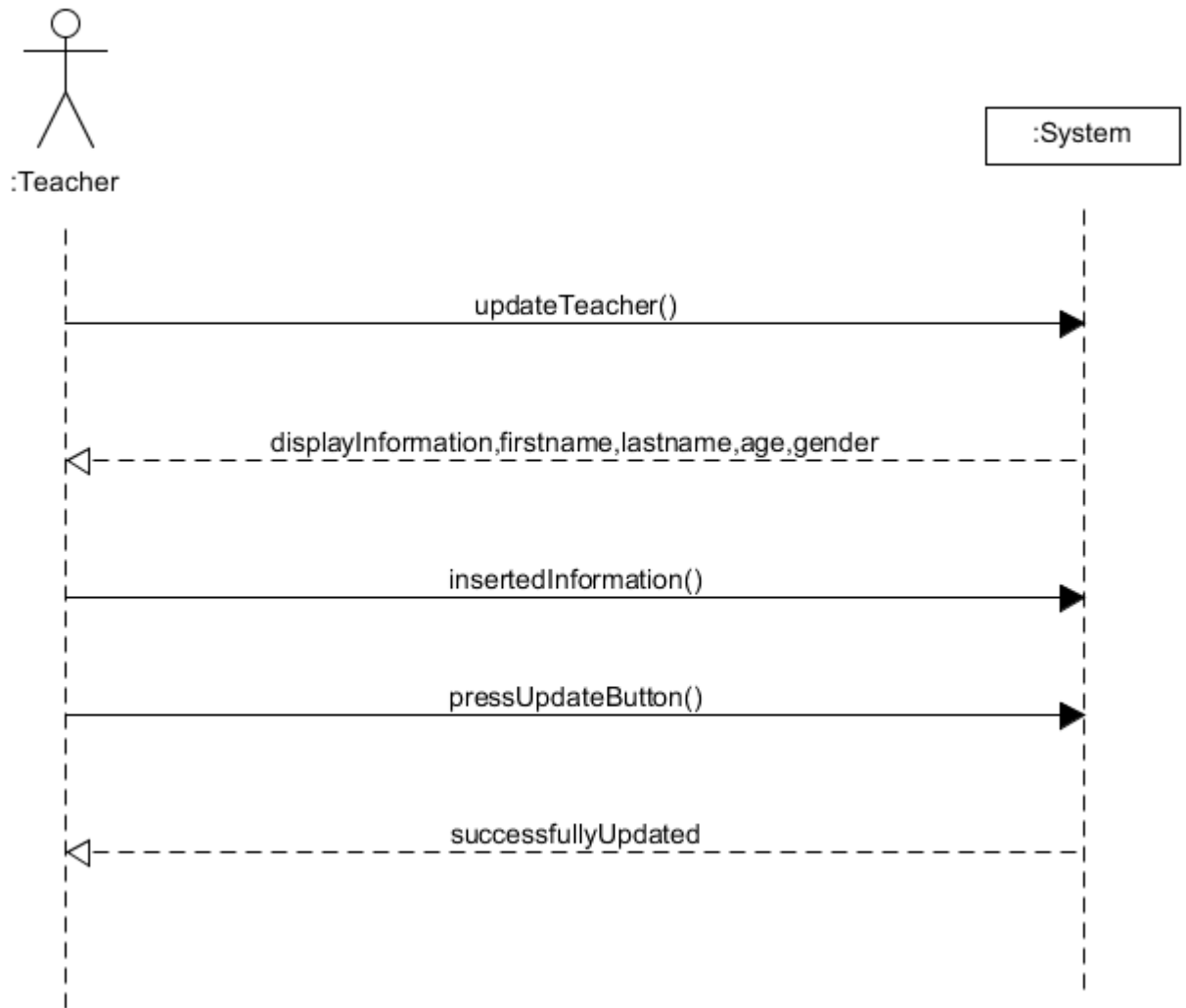


Figure 4.5. 15 Update teacher

4.3.5.18 Import Students

Main Success Scenario:

1. Admin selects “ Import students “ from the import resources.
2. System displays import student’ s button.
3. Admin presses the button.
4. System imported successfully.

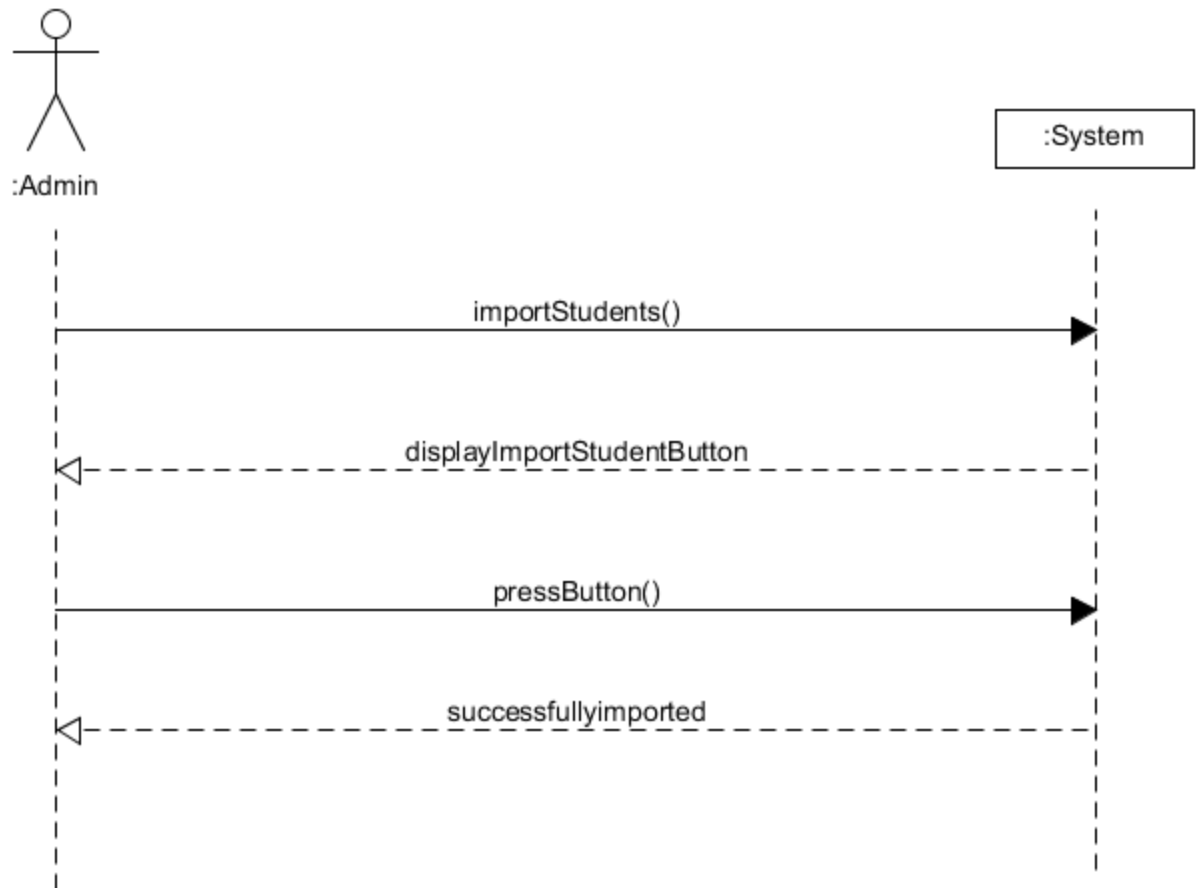


Figure 4.5. 16 Import Students

4.3.5.19 Export Courses

Main Success Scenario:

1. Admin selects “ Export courses “ from the export resources.
2. System displays export courses button.
3. Admin presses the button.
4. System exported successfully.

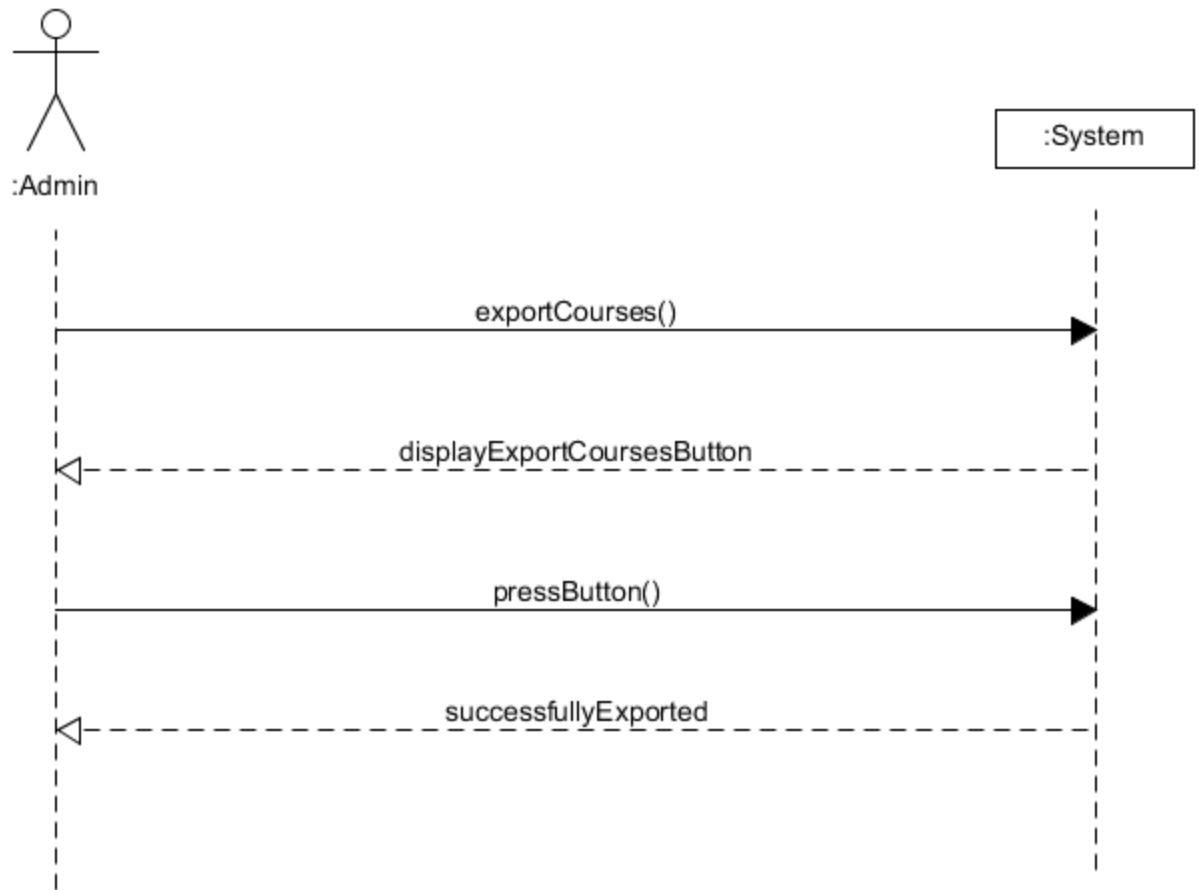


Figure 4.5. 17 Export Courses

4.3.5.20 View Venues

Main Success Scenario:

1. Admin selects “ view venues “ from the view resources.
2. System displays venues details.
3. Admin view details.

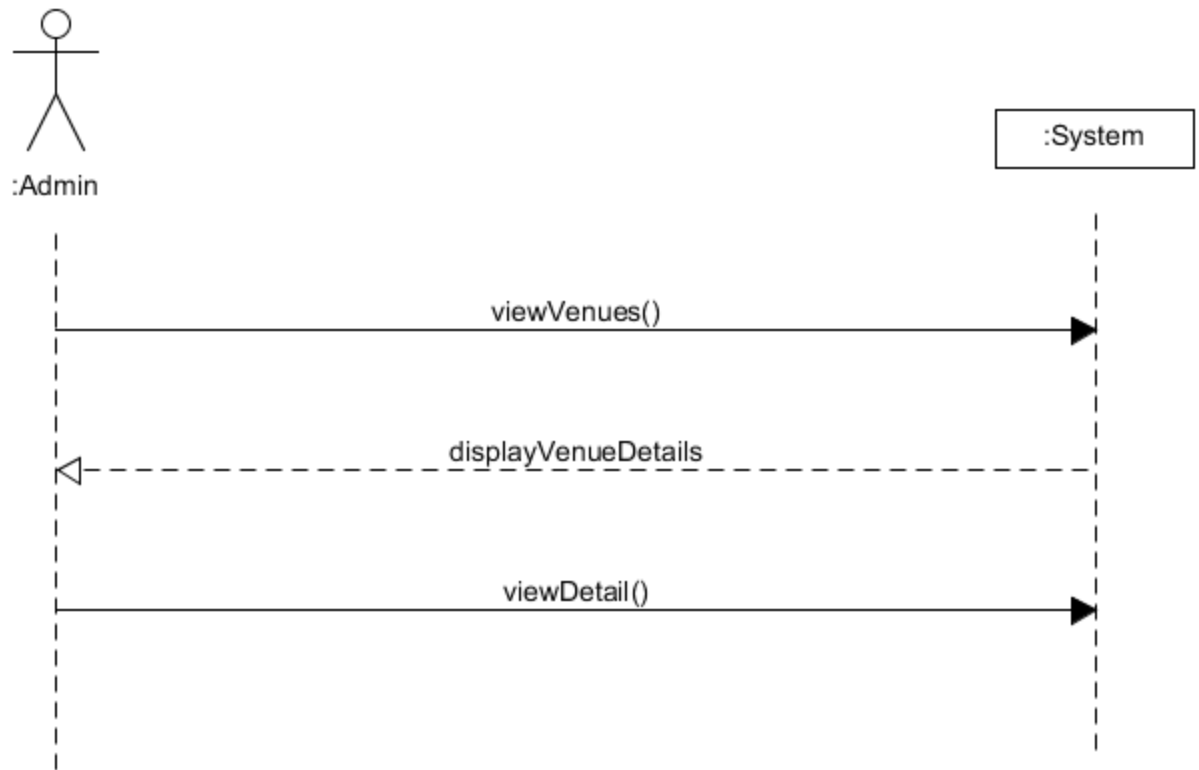


Figure 4.5. 18 View Venues

4.3.5.21 Delete Clashes

Main Success Scenario:

1. Admin selects “ delete clashes “ from the delete resources.
2. System displays clashes details.
3. Admin view details and presses delete button.
4. System deleted it successfully.

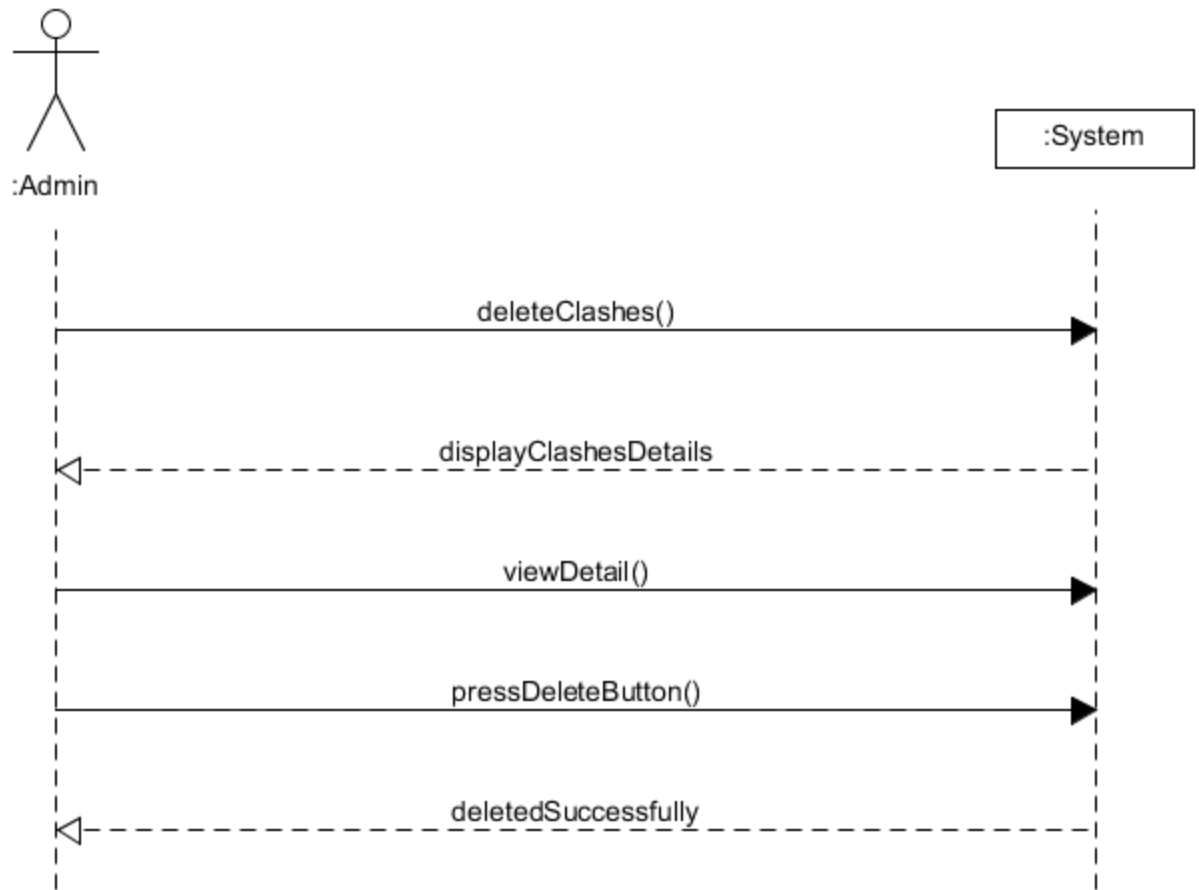


Figure 4.5. 19 Delete Clashes

4.3.5.22 Manage Admin

Main Success Scenario:

1. Super admin selects “ manage admin" from the menu.
2. System displays add, update and delete admin’ s information
3. Super admin selects one of the options.
4. System displays related fields.

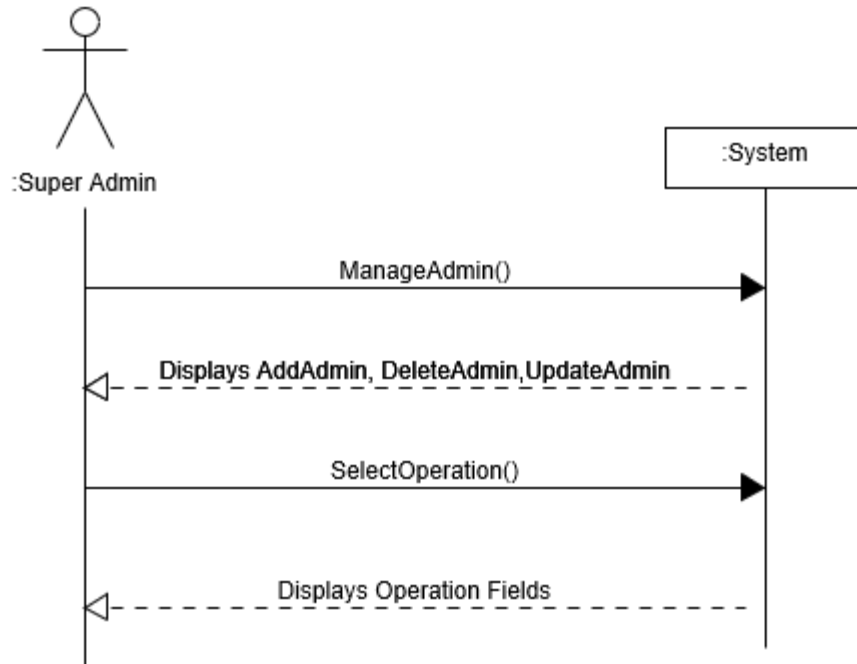


Figure 4.5. 20 Manage Admin

4.3.5.23 Add Admin

Main Success Scenario

1. Super admin selects “ add admin” from the manage admin.
2. System displays information to be added i.e. first name, last name, CNIC, contact details, address etc.
3. Super admin inserted the required information of admin.
4. System checks for all fields and enabled the add button.
5. Super admin presses add button.
6. System displays “ successfully added” message on screen.

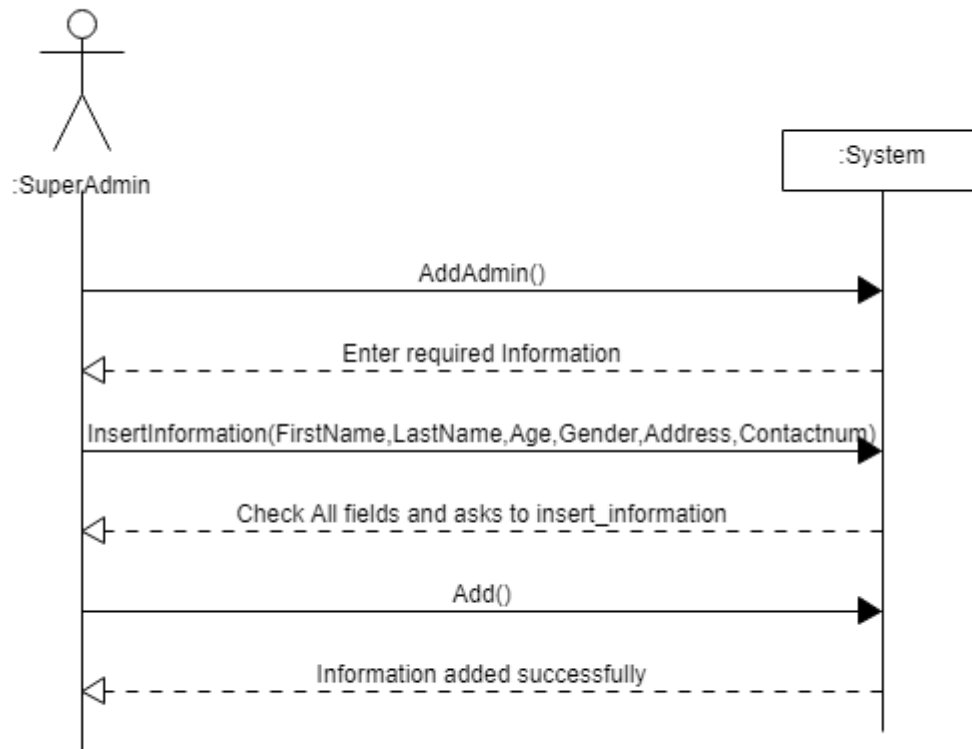


Figure 4.5. 21 Add Admin

4.3.5.24 Delete Admin

Main Success Scenario

1. Super admin selects “ delete admin” from the manage admin.
2. System displays enter admin id to be removed
3. Super admin inserted the required information of admin.
4. System displays confirmation message of deletion.
5. Super admin presses confirm button.
6. System displays “ successfully deleted” message on screen.

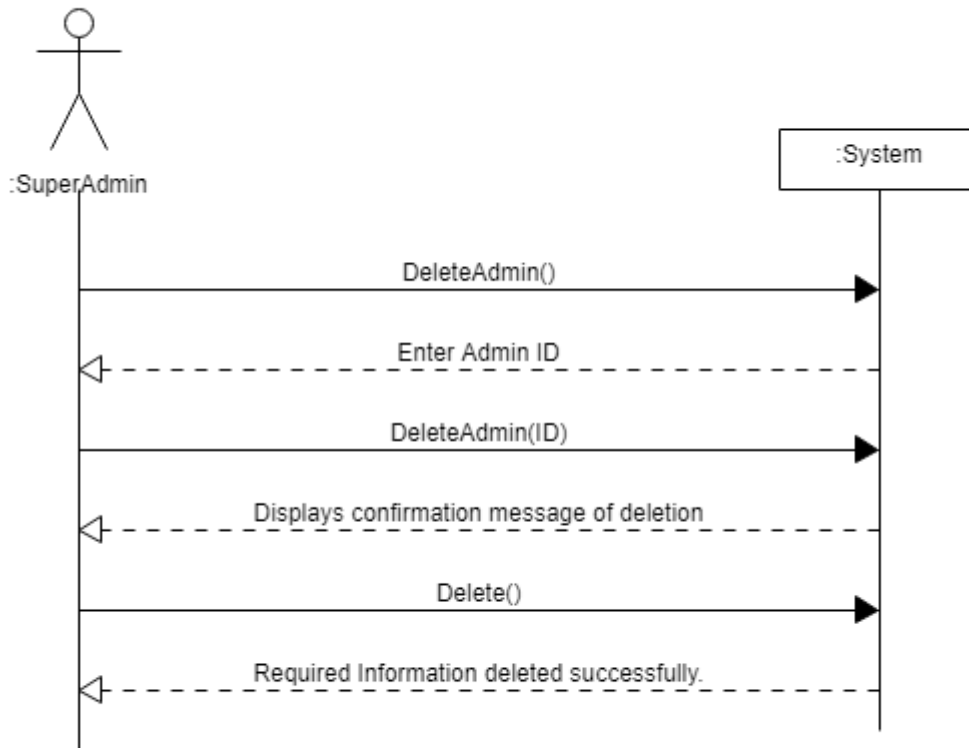


Figure 4.5. 22 Delete Admin

4.3.5.25 Update Admin

Main Success Scenario

1. Super admin selects “ update admin” from the manage admin.
2. System displays enter admin id to which you want to update information.
3. Super admin inserted the updated information of admin i.e. his contact number, his marital status etc.
4. System displays confirmation message “ update successful” .

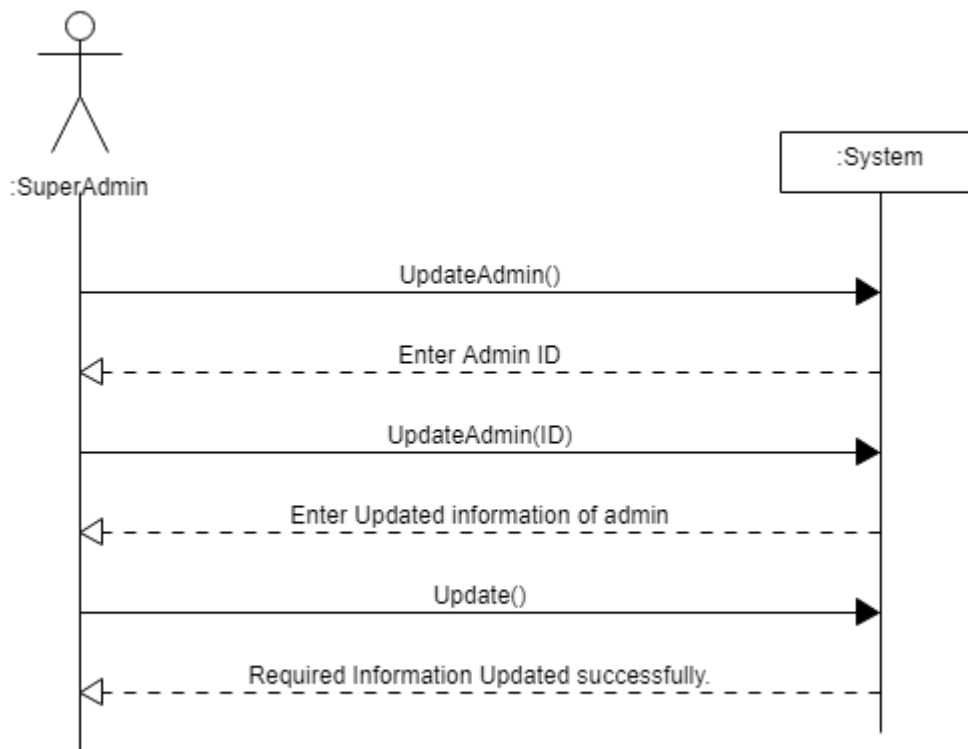


Figure 4.5. 23 Update Admin

4.3.5.26 View Admin

Main Success Scenario:

1. Super admin selects “ view admin” from the manage admin.
2. System displays the information.
3. Super admin viewed the details.

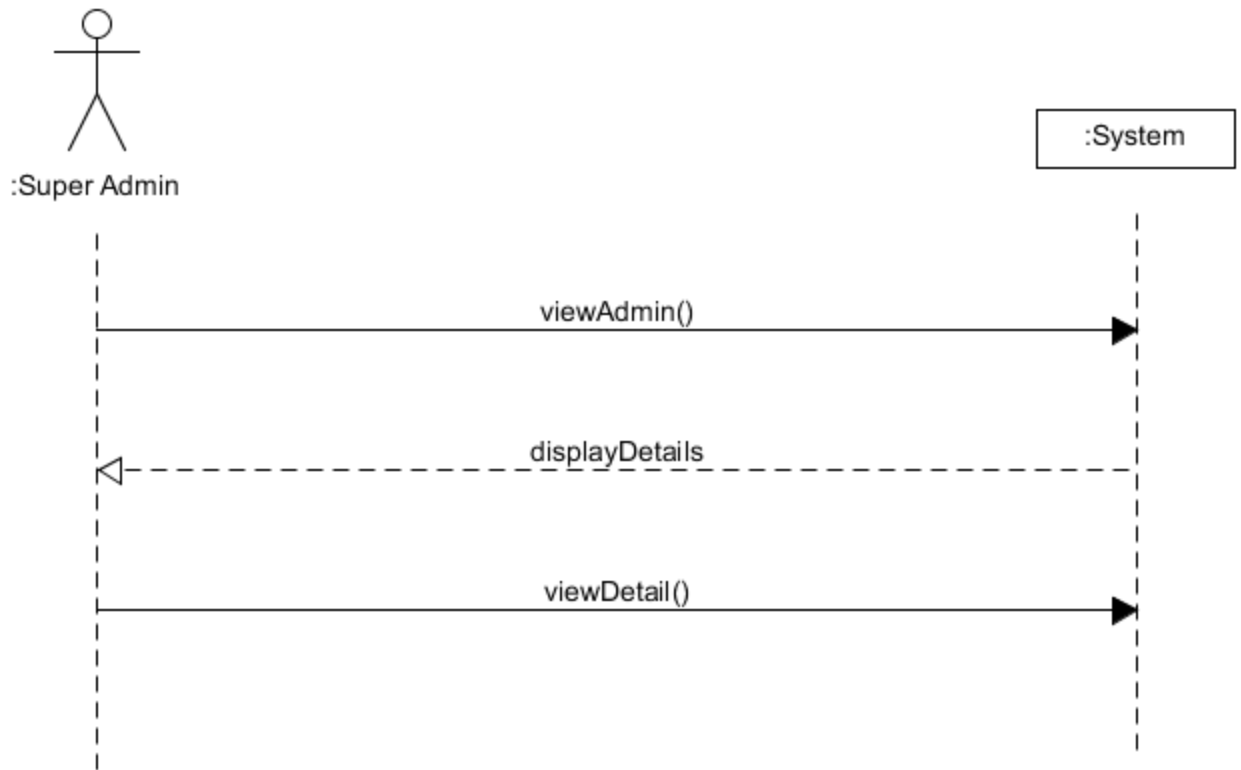


Figure 4.5. 24 View Admin

4.3.5.27 Generate Timetable

Main Success Scenario

1. Admin clicks generate timetable from menu.
2. System loaded courses, venues, students, and teachers.
3. Admin selects number of timetable to be generated.
4. System displays “ successfully generated” message.
5. Admin can view different timetables.
6. System asks to select three timetables.
7. Admin selects timetable.
8. System displays “ successfully selected” message.
9. Admin now send these to teachers for voting.
10. System displays “ successfully send “ message.

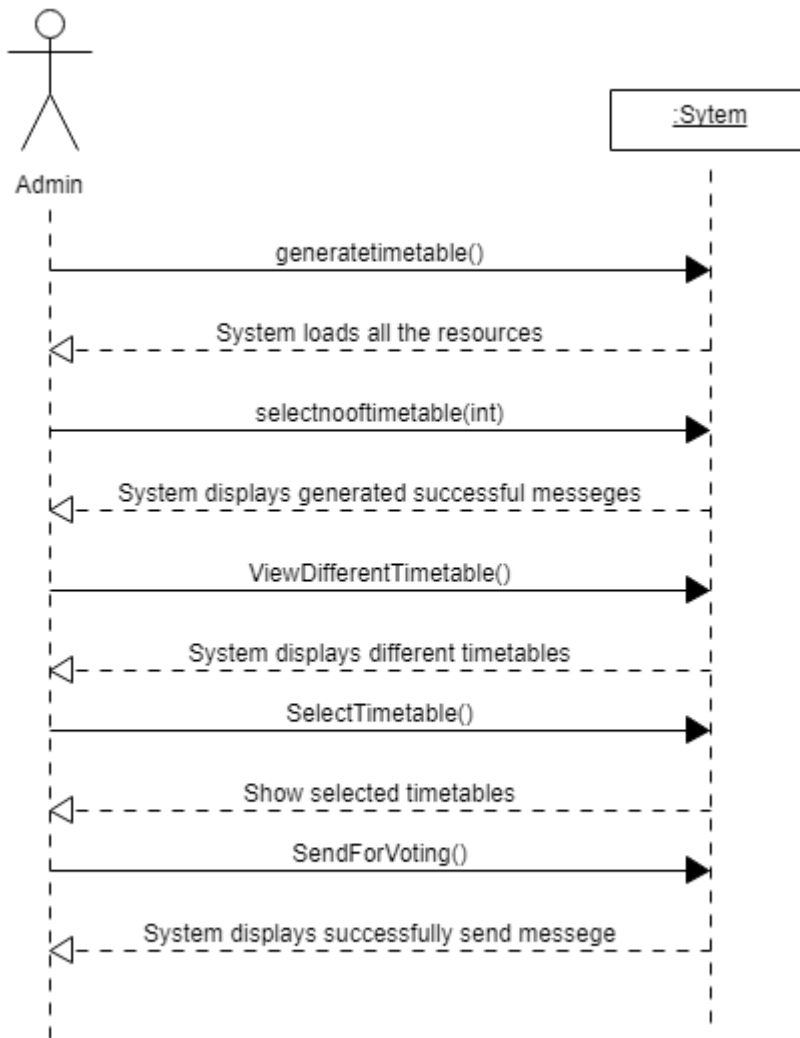


Figure 4.5. 25 Generate Timetable

4.3.5.28 Voting for timetable

Main success Scenario

1. Admin sends all versions of timetable generated to all the teachers.
2. System displays confirmation message “ successfully sent”
3. A teacher views all the versions of timetable.
4. System displays options to vote for the viewed timetable.
5. Teacher sees the time table best suited to him/her and give votes to all versions of timetable accordingly.

6. System sends voting response from teacher to Admin.

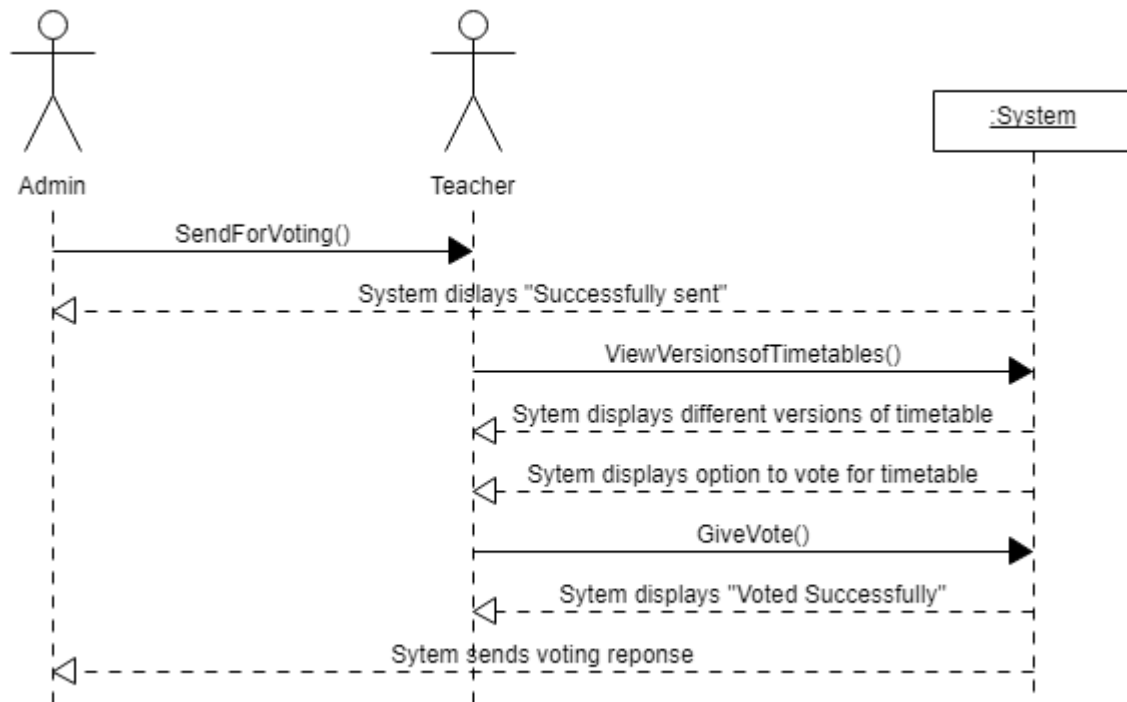


Figure 4.5. 26 Voting for timetable

4.3.5.29 Report Clash

Main success Scenario

1. Student found clash in his/her time table.
2. System displays form to report clash.
3. Student report clash to Admin via filling form
4. Student then press submit button.
5. Notification than send to Admin.
6. System displays confirmation message “ successfully reported” .

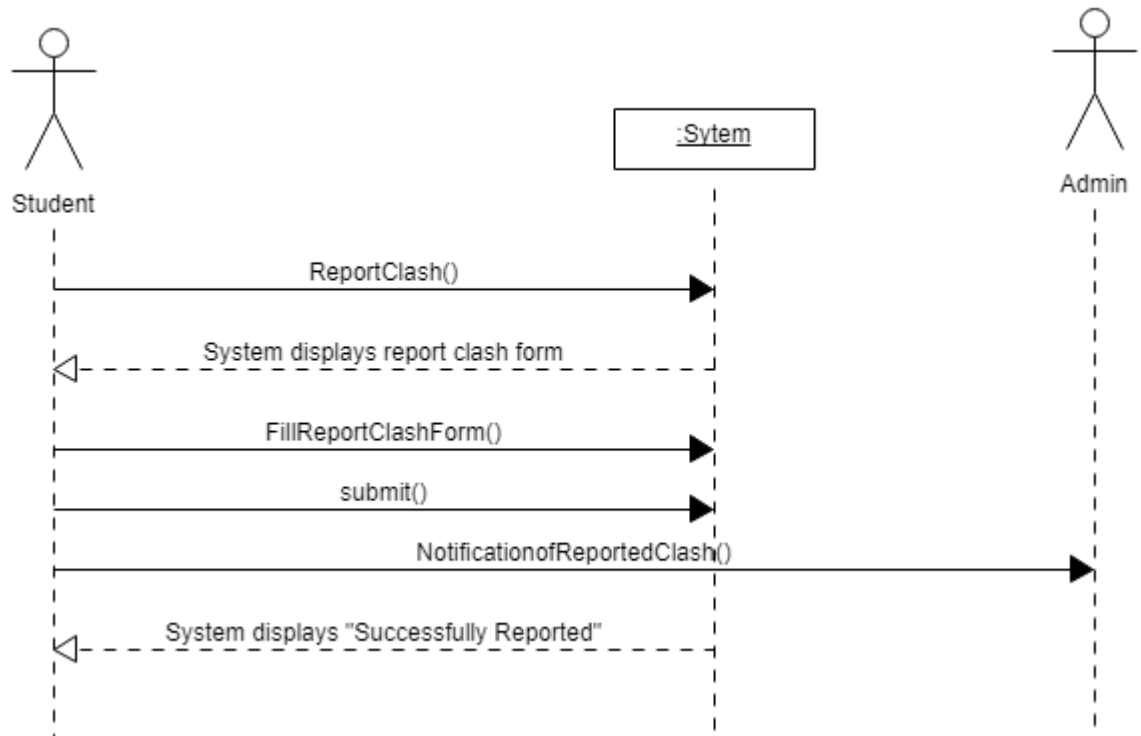


Figure 4.5. 27 Report Clash

4.3.5.30 Add preferences

Main Success Scenario:

1. Teacher wants to add preferences.
2. System displays add preferences form.
3. Teacher can add preferences by filling form data.
4. Teacher then presses submit button to confirm.
5. Notification than send to Admin.
6. System displays confirmation message “ successfully added” .

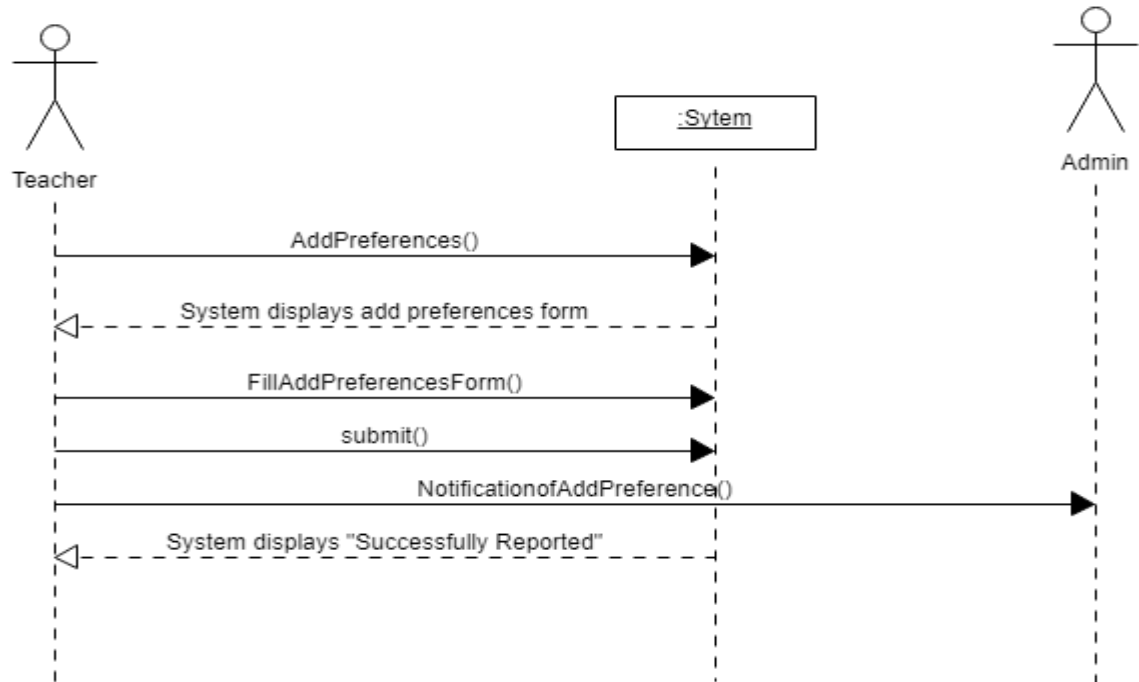


Figure 4.5. 28 Add preferences

4.3.5.31 Evaluate Timetable

Main Success Scenario:

1. Admin wants to review voting result.
2. System displays voting result data
3. Admin selects the timetable with the highest votes.
4. Admin send the selected timetable to all students and teachers.
5. System displays “ Sent successfully” message.

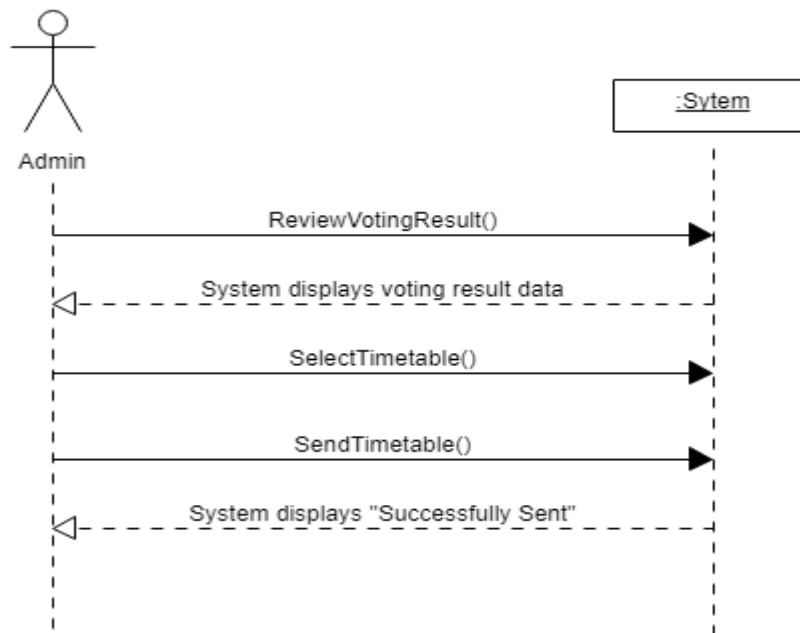


Figure 4.5. 29 Evaluate Timetable

Chapter 5: Implementation

In this chapter we would discuss in detail about the implementation of this project. The discussion will include the basic work flow of the tool, coding techniques and practices applied, programming languages, implementation environment used.

5.1 Endeavour

Team	Work
Syeda Zarwa Faiz	Super admin Panel <ul style="list-style-type: none"> • manage Admin
Shumaila Rehman	Admin Panel <ul style="list-style-type: none"> • Manage resources(Admin, Students, Teachers, Venues, Courses, Clashes, Preferences)
Maria Shoaib	Student Panel <ul style="list-style-type: none"> • Report Clashes
	Teacher Panel <ul style="list-style-type: none"> • Voting
	Timetable Generation
	Database

Way of Working	
Documentation	Microsoft Word
Usecase Diagram	Creatly
SSD, ERD, Class Diagram	UMLET
Implementation	Brackets
Backup	Google Drive
Coordination	Bitbucket
WBS	Project management tool (project professional)

5.2 Flow Control/Pseudo codes

Flow control is the management of data flow between computers or devices or between nodes in a network so that the data can be handled at an efficient pace. A flow control diagram is used to describe the control flow of a process or review. A flow control diagram can consist of a subdivision to show sequential steps, with if-then else conditions, repetition, and/or case conditions. Our system has four main modules, i.e. Super Admin, Admin, student and teacher. Following flow charts shall show the flow control of “ Automated Timetabler Maker” .

5.2.1 Super Admin

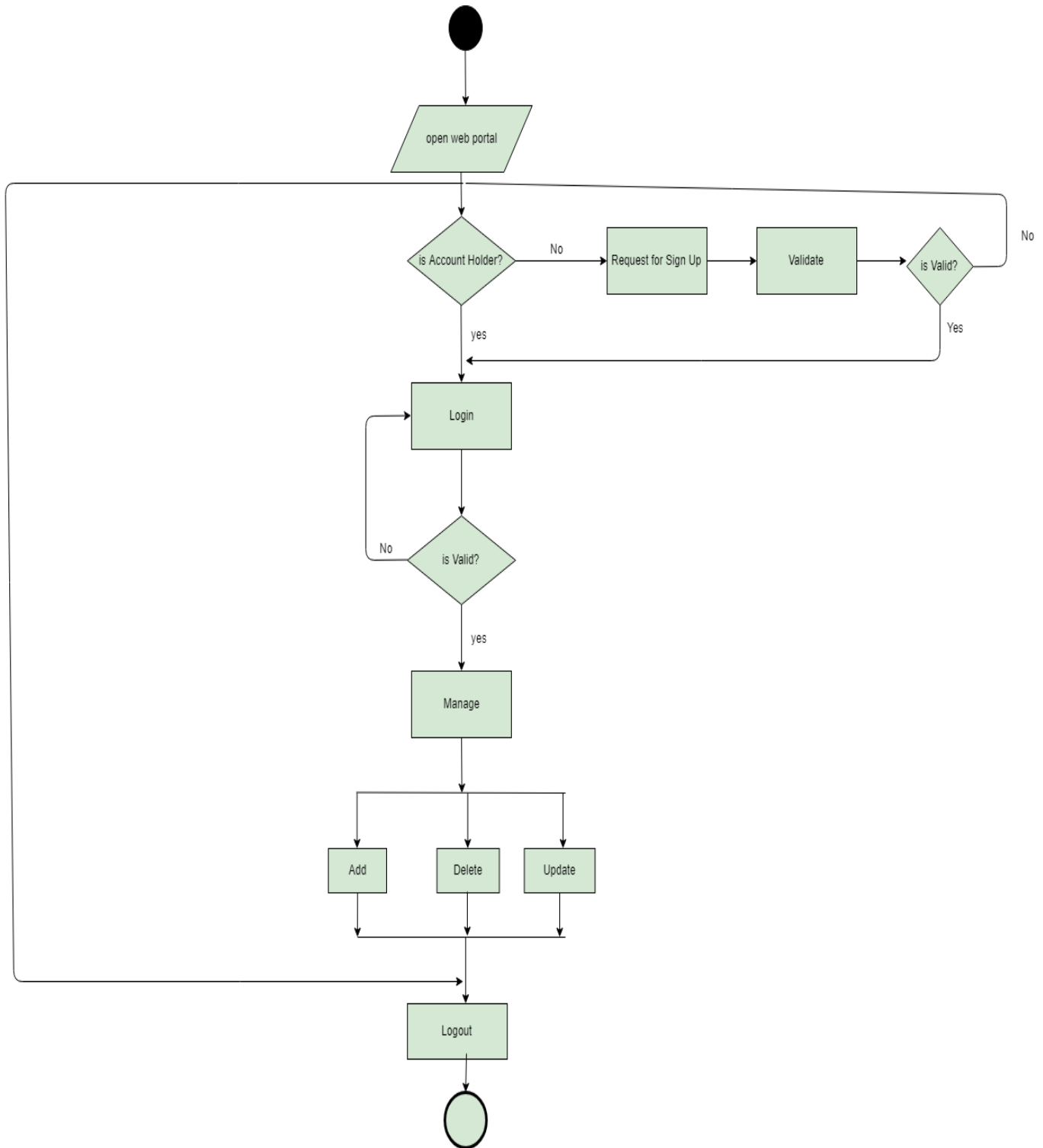


Figure 5.2. 1 Flow control of super admin

5.2.2 Admin

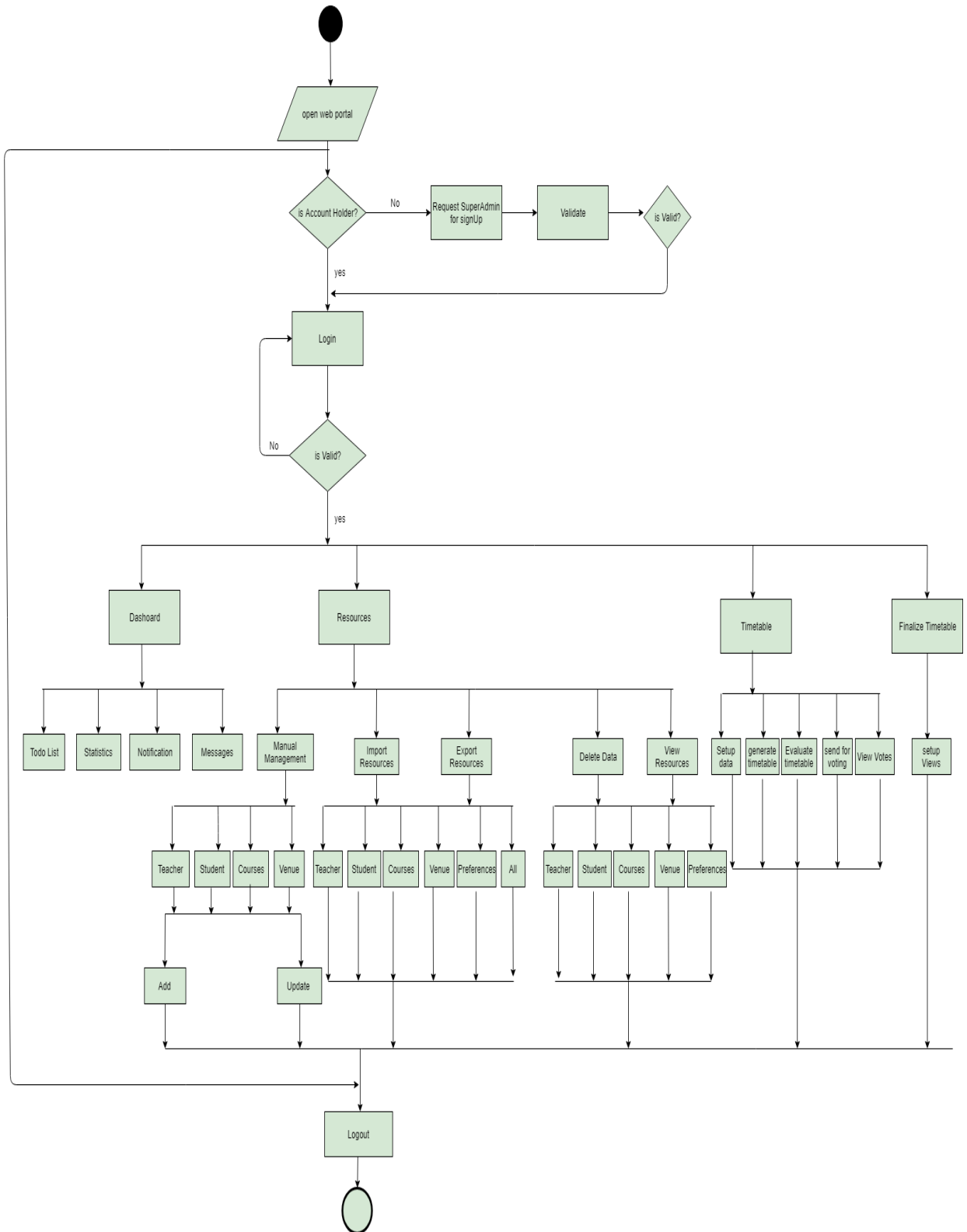


Figure 5.2. 2 Flow control of Admin

5.2.3 Teacher

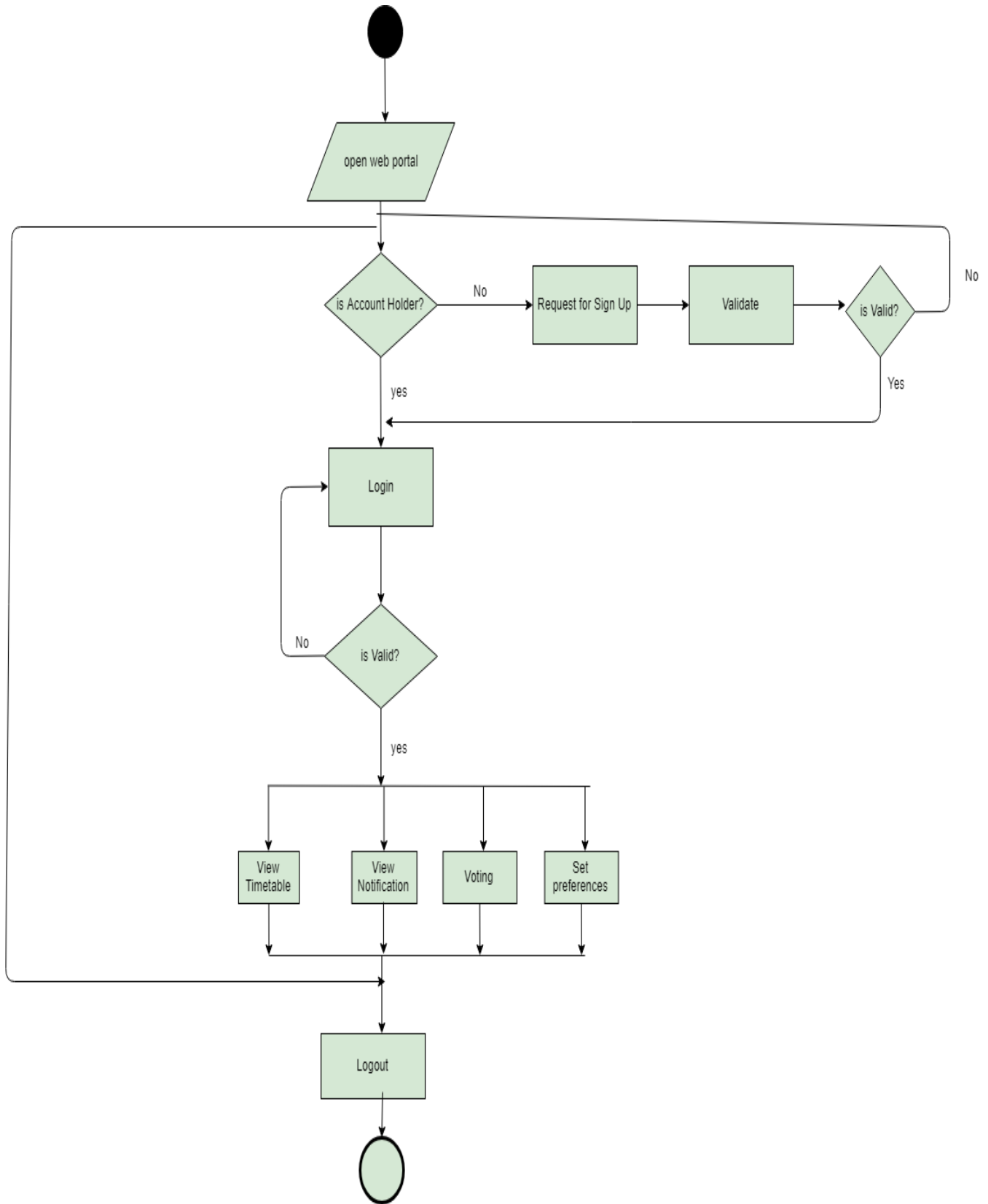


Figure 5.2. 3 Flow control of tecaher

5.2.4 Student

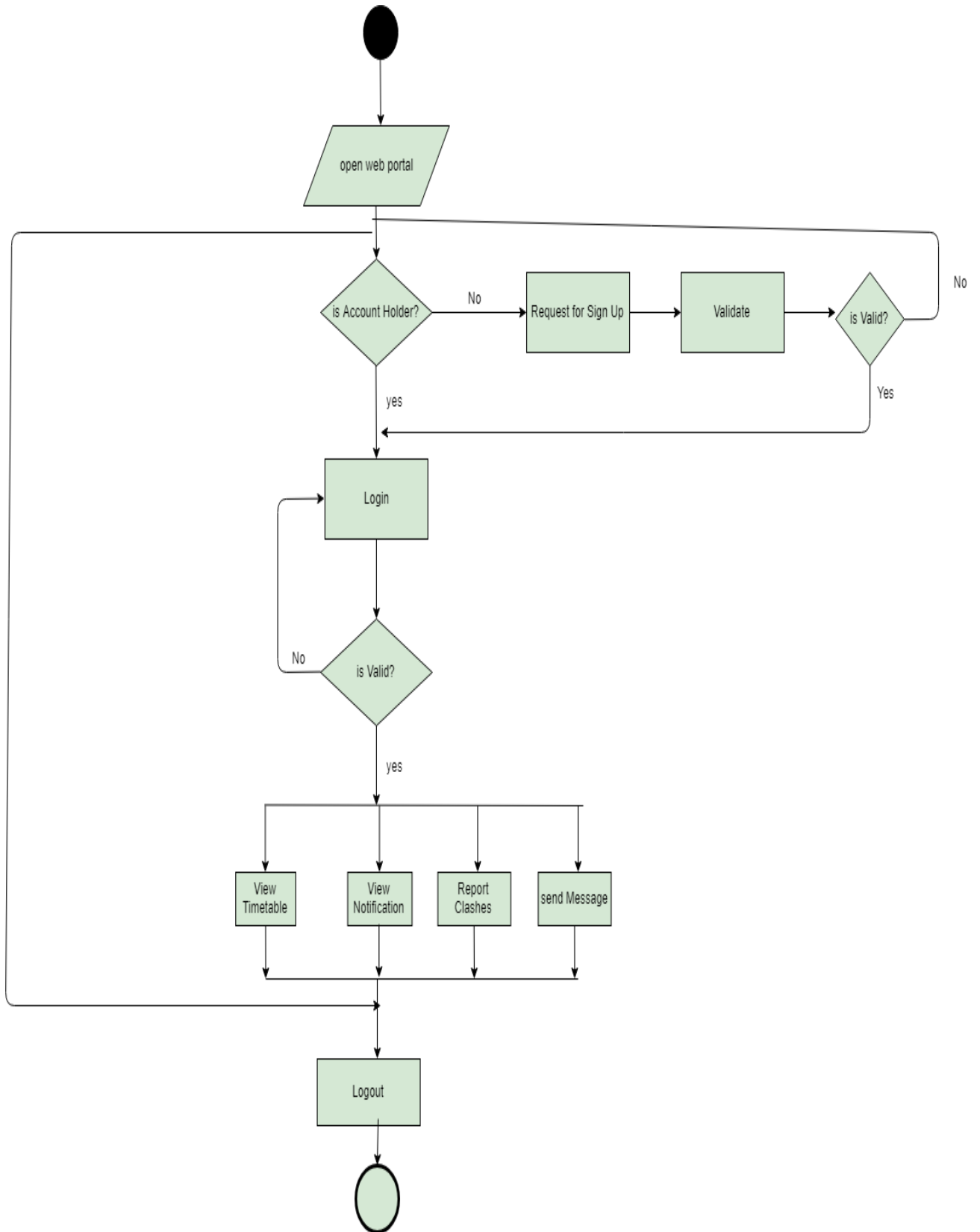


Figure 5.2. 4 Flow control of student

5.2.5 Voting

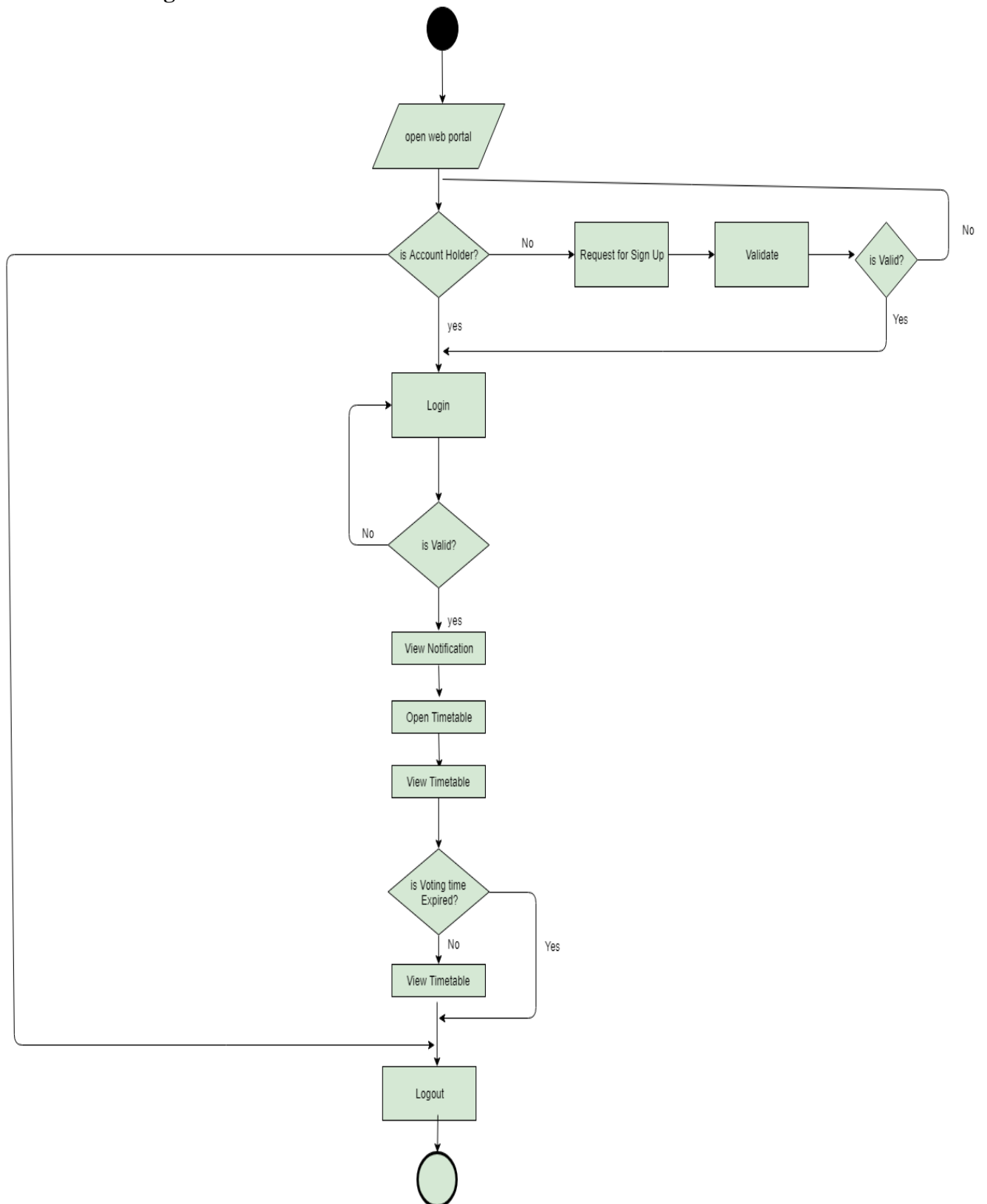


Figure 5.2. 5 Flow Control of voting mechanism

5.2.6 Generate Timetable

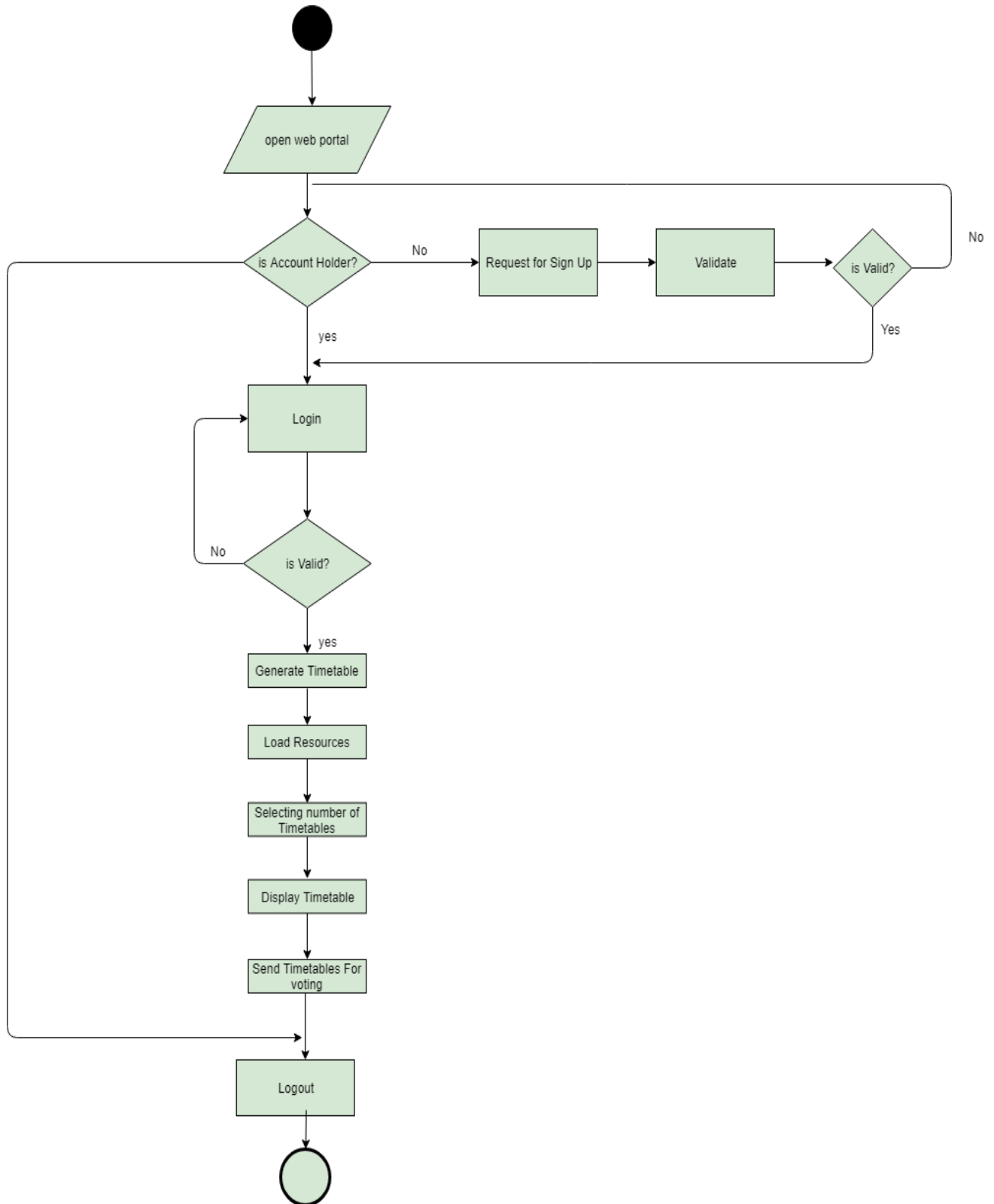


Figure 5.2. 6 Flow control of generate timetable

5.3 Components, Libraries, Web Services and stubs

Components in software development are a generic term which can be used purposely according to the software being developing. Components sometimes refer to the hardware components that are required by the software being developing and can also stand for the software components required. We can think of a component as the subset of a module or in other words one module can have multiple components inside it.

Libraries are the pre-written set of classes, written in some programming language we will be using to assist our development. In computer science, a **library** is a collection of non-volatile resources used by computer programs, often for **software** development. These may include configuration data, documentation, help data, message templates, pre-written code and subroutines, classes, values or type specifications. They will help us in achieving some of our desired functionalities (relevant to graphical UI) without explicitly writing their code. The choice of using any library depends upon the level of advantages it provides as compared to other similar ones.

Web services provide a standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks.

Stub is a small program routine that substitutes for a longer program, possibly to be loaded later or that is located remotely. For example, a program that uses Remote Procedure Calls (RPC) is compiled with **stubs** that substitute for the program that provides a requested procedure.

“Automated Timetable Maker” is a system developed using PHP approach with bootstrap framework and MySQL database. PHP provides set of core libraries, components, and controls which provides access to system functionality. Bootstrap is a free and open-source front-end web framework for designing websites it contains HTML- and CSS-based design templates for forms, buttons, navigation and other interface components. We are using Bootstrap version 3.0 which emphasizing responsive design by default.

Language, Framework and Platform

The technologies used for this project are:

- HTML5
- CSS3
- PHP
- JavaScript
- JQuery
- Bootstrap
- MySQL
- AJAX
- WAMP/XAMPP
- UMLET
- BRACKET
- VISIO

5.4 Best Practices / Coding Standards

As we mentioned earlier “ STTM ” uses PHP, Bootstrap framework which provides PHP coding standards and core libraries that contain predefined scripts as best practices for the development. Use of these libraries makes “ STTM” responsive, modifiable and reliable as it uses libraries like JSON libraries. For coding styles we are following complete standards i.e. for attributes using Camel Casing but this may not applicable for database attributes and for functions used Pascal Casing. “ STTM” uses some other technologies like Ajax, JavaScript and Query for data requests, validations for database (MySQL).

The following conventions are followed:

1. Naming convention

- a. Pages: All in lowercase with dashed separators.

- b. Variable: Follow camel case.
2. Commenting convention
 - a. Begin comment text with an uppercase letter
 3. Try-catch in exception handling where necessary

5.5 Deployment Environment

“ STTM” is a web based system that runs on internet. For using this system, Internet connection is mandatory. No common user can access this system only authorized and registered users admin can access it through internet. First data move from client side to server side then passed to database over internet. It can be deployed at web server like HTTP.

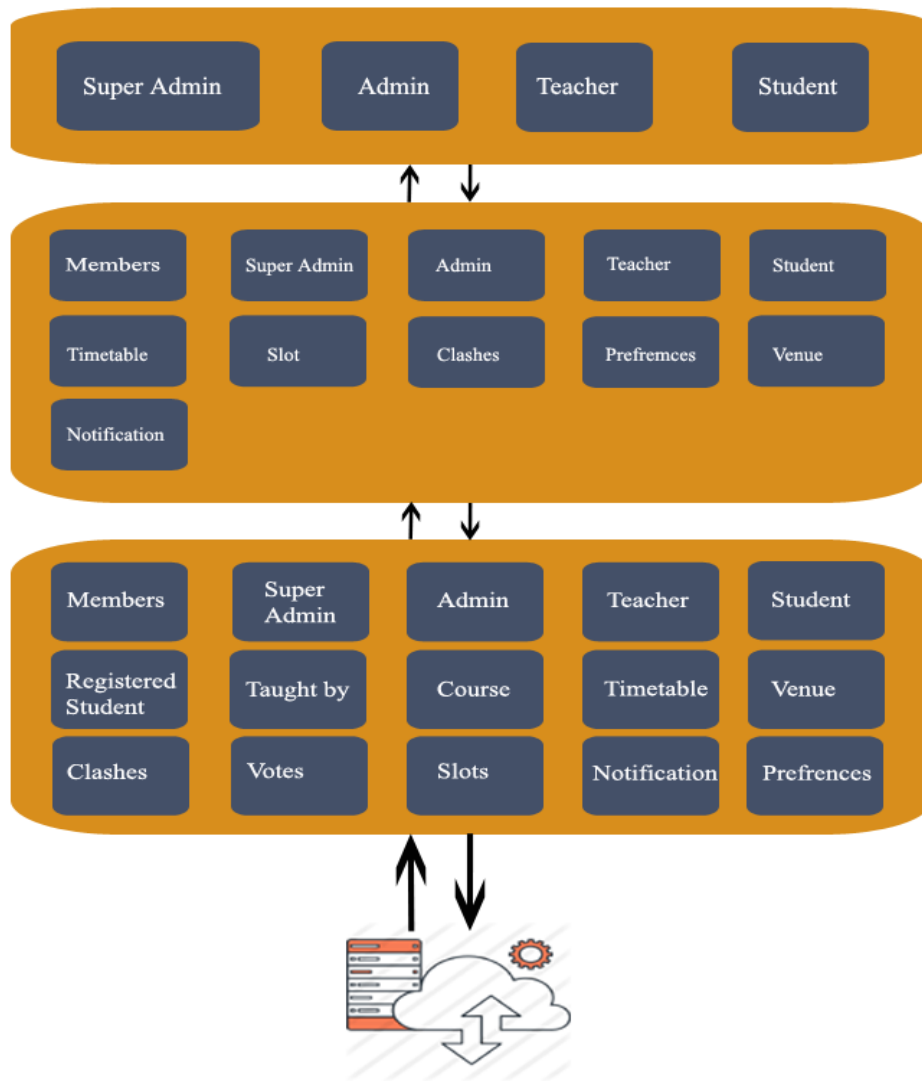


Figure 5.5. 1 Deployment Diagram

5.6 Summary

The main focus of this chapter is on monitoring; flow of events and data. Chapter explains which users interact with the system, its control flows and system response. Diagram shows how the system responds against inputs and what possible options are available in the system for it. Moreover it describes the libraries and technologies that were used in the development of “ STTM” . We have also discussed the coding techniques and best practices on which the software is implemented.

Chapter 6: Testing and Evaluation

6.1 Introduction: Describe which aspects you are focusing on when testing software and hardware environment.

Software testing is a process of executing a program or application with the intent of finding the software bugs and to check whether the actual results match the expected results ensuring a defect free software system. The technique followed to test ‘ Automated Timetable Maker’ system is Black boxing. Black box testing is a method of software testing that examines the functionality of an application based on the specifications and is also known as Specifications based testing. Internal system design is not considered in this type of testing. Tests are based on requirements and functionality.

6.2 List of Test Scenario

[TC- 1] Login of Admin

Test Case #:	TC – 1
System:	Automated Timetable Maker
Test Case Name:	Login of Admin
Related Requirement:	FR-3.3.2.1
Short Description:	Test that admin can login
Designed Date:	2/12/2018
Execution Date:	2/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin opens the system.	The system should display home page.	The system displays home page.	pass

2	Admin click on his own admin panel.	The system should display a login form.	The system display login form.	pass
3	Admin enter his username and password.	The system should login after validating.	The system login admin.	pass

Post-Condition:

Admin successfully login in the system.

[TC- 2] View Timetable by Student

Test Case #:	TC – 2
System:	Automated Timetable Maker
Test Case Name:	View Timetable by Student
Related Requirement:	FR-3.3.3.5
Short Description:	Test that student can view timetable.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Student has an account in automated timetable maker.
2. Student must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Student clicks on view timetable.	The system should display	The system displays	pass

		timetable page.	timetable page.	
--	--	-----------------	-----------------	--

Post-Condition:

Student successfully viewed the timetable.

[TC- 3] View Notification by Teacher

Test Case #:	TC – 3
System:	Automated Timetable Maker
Test Case Name:	View Notification by Teacher
Related Requirement:	FR-3.3.4.7
Short Description:	Test that teacher can view notification.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Teacher has an account in automated timetable maker.
2. Teacher must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Teacher clicks on view notifications.	The system should display notification page.	The system displays notification page.	pass

Post-Condition:

Teacher successfully viewed the notification.

[TC- 4] Manage Admin

Test Case #:	TC – 4
System:	Automated Timetable Maker
Test Case Name:	Manage Admin
Related Requirement:	FR-3.3.1.2
Short Description:	Test that super admin can manage admin.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Super Admin has an account in automated timetable maker.
2. Super Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Super Admin click on manage Admin.	The system should display a dropdown including add admin, delete admin, view admin, update admin.	The system displays a dropdown including add admin, delete admin, view admin, update admin.	Pass
2	Super Admin clicks on Add Admin.	The system should display add admin page.	The system displays add admin page.	Pass

3	Super Admin clicks on Delete Admin.	The system should display delete admin page.	The system displays delete admin page.	Pass
4	Super Admin clicks on Update Admin.	The system should display update admin page.	The system displays update admin page.	Pass
5	Super Admin clicks on view Admin.	The system should display view admin page.	The system displays view admin page.	Pass

Post-Condition:

Super Admin successfully managed Admin.

[TC- 5] Add Admin

Test Case #:	TC – 5
System:	Automated Timetable Maker
Test Case Name:	Add Admin
Related Requirement:	FR-3.3.1.2
Short Description:	Test that super admin can add admin.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Super Admin has an account in automated timetable maker.
2. Super Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Super Admin click on manage Admin.	The system should display a dropdown including add admin, delete admin, view admin, update admin.	The system displays a dropdown including add admin, delete admin, view admin, update admin.	Pass
2	Super Admin clicks on Add Admin.	The system should display add admin page.	The system displays add admin page.	Pass
3	Super Admin enters information in the form.	The system should enter data.	The system enters data.	Pass
4	Super Admin click on add button.	The system should display message successfully added and should store in database.	The system displays message successfully added and stored in database.	Pass

Post-Condition:

Super Admin successfully add Admin.

[TC- 6] Delete Admin

Test Case #:	TC – 6
---------------------	--------

System:	Automated Timetable Maker
Test Case Name:	Delete Admin
Related Requirement:	FR-3.3.1.2
Short Description:	Test that super admin can delete admin.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Super Admin has an account in automated timetable maker.
2. Super Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Super Admin click on manage Admin.	The system should display a dropdown including add admin, delete admin, view admin, update admin.	The system displays a dropdown including add admin, delete admin, view admin, update admin.	Pass
2	Super Admin clicks on Delete Admin.	The system should display delete admin page.	The system displays delete admin page.	Pass
3	Super Admin search for specific admin.	The system should display his/her details.	The system displays his/her details.	Pass

4	Super Admin click on delete button.	The system should display message successfully deleted and should be deleted from database.	The system displays message successfully deleted and deleted from database.	Pass
---	-------------------------------------	---	---	------

Post-Condition:

Super Admin successfully deleted Admin.

[TC- 7] Update Admin

Test Case #:	TC – 7
System:	Automated Timetable Maker
Test Case Name:	Update Admin
Related Requirement:	FR-3.3.1.2
Short Description:	Test that super admin can update admin.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Super Admin has an account in automated timetable maker.
2. Super Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Super Admin click	The system	The system	Pass

	on manage Admin.	should display a dropdown including add admin, delete admin, view admin, update admin.	displays a dropdown including add admin, delete admin, view admin, update admin.	
2	Super Admin clicks on update Admin.	The system should display update admin page.	The system displays update admin page.	Pass
3	Super Admin search for specific admin.	The system should display his/her details.	The system displays his/her details.	Pass
4	Super Admin enter details.	The system should enter details.	The system enters details.	
5	Super Admin click on update button.	The system should display message successfully updated and should be updated from database.	The system displays message successfully updated and updated in database.	Pass

Post-Condition:

Super Admin successfully updated Admin.

[TC- 8] View Admin

Test Case #:	TC – 8
---------------------	--------

System:	Automated Timetable Maker
Test Case Name:	View Admin
Related Requirement:	FR-3.3.1.2
Short Description:	Test that super admin can view admin.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Super Admin has an account in automated timetable maker.
2. Super Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Super Admin click on manage Admin.	The system should display a dropdown including add admin, delete admin, view admin, update admin.	The system displays a dropdown including add admin, delete admin, view admin, update admin.	Pass
2	Super Admin clicks on View Admin.	The system should display view admin page.	The system displays view admin page.	Pass
3	Super Admin search for specific admin.	The system should display his/her details.	The system displays his/her details.	Pass

Post-Condition:

Super Admin successfully viewed Admin.

[TC- 9] Manage Resources

Test Case #:	TC – 9
System:	Automated Timetable Maker
Test Case Name:	Manage Resources
Related Requirement:	FR-3.3.2.3
Short Description:	Test that admin can manage resources.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on manage resources.	The system should display a dropdown including manage teachers, manage students, manage venues, manage courses.	The system displays a dropdown including manage teachers, manage students, manage venues,	Pass

			manage courses.	
2	Admin clicks on manage teachers.	The system should display add teachers and update teachers page.	The system displays add teachers and update teacher' s page.	Pass
3	Admin clicks on manage students.	The system should display add students and update students page.	The system displays add students and update student' s page.	Pass
4	Admin clicks on manage courses.	The system should display add courses and update courses page.	The system displays add courses and update courses page.	Pass
5	Admin clicks on manage venues.	The system should display add venues and update venues page.	The system displays add venues and update venues page.	Pass

Post-Condition:

Admin successfully managed resources.

[TC- 10] Manage Students

Test Case #:	TC – 10
System:	Automated Timetable Maker
Test Case Name:	Manage Students

Related Requirement:	FR-3.3.2.4
Short Description:	Test that admin can manage students.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on manage resources.	The system should display a dropdown including manage teachers, manage students, manage venues, manage courses.	The system displays a dropdown including manage teachers, manage students, manage venues, manage courses.	Pass
2	Admin clicks on manage students.	The system should display add students and update students page.	The system displays add students and update student' s page.	Pass
3	Admin clicks on add	The system	The system	Pass

	students.	should display add students page.	displays add student' s page.	
4	Admin clicks on update students.	The system should display update students page.	The system displays update student' s page.	Pass

Post-Condition:

Admin successfully managed students.

[TC- 11] Add Students

Test Case #:	TC – 11
System:	Automated Timetable Maker
Test Case Name:	Add Students
Related Requirement:	FR-3.3.2.4
Short Description:	Test that admin can add students.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.
3. Admin must be in mange students.

Steps	Action	Expected Output	Actual output	Pass/Fail
-------	--------	-----------------	---------------	-----------

1	Admin clicks on Add Students.	The system should display add students page.	The system displays add student' s page.	Pass
2	Admin enters information in the form.	The system should enter data.	The system enters data.	Pass
3	Admin click on add button.	The system should display message successfully added and should store in database.	The system displays message successfully added and stored in database.	Pass

Post-Condition:

Admin successfully add Students.

[TC- 12] Update Students

Test Case #:	TC – 12
System:	Automated Timetable Maker
Test Case Name:	Update Students
Related Requirement:	FR-3.3.2.4
Short Description:	Test that admin can update students.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.
3. Admin must be in manage students.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin clicks on update Students.	The system should display update students page.	The system displays update student' s page.	Pass
3	Admin search for specific student.	The system should display his/her details.	The system displays his/her details.	Pass
4	Admin enter details.	The system should enter details.	The system enters details.	
5	Admin click on update button.	The system should display message successfully updated and should be updated from database.	The system displays message successfully updated and updated in database.	Pass

Post-Condition:

Admin successfully updated Students.

[TC- 13] Import Resources

Test Case #:	TC – 13
System:	Automated Timetable Maker
Test Case Name:	Import Resources
Related Requirement:	FR-3.3.2.2
Short Description:	Test that admin can import resources.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on import resources.	The system should display a dropdown including import teachers, import students, import venues, import courses, import clashes, import preferences.	The system displays a dropdown including import teachers, import students, import venues, import courses, import clashes, import preferences.	Pass
2	Admin clicks on import students.	The system should display import students page.	The system displays import student' s page.	Pass

3	Admin clicks on import courses.	The system should display import courses page.	The system displays import courses' page.	Pass
4	Admin clicks on import teachers.	The system should display import teachers page.	The system displays import teachers' page.	Pass
5	Admin clicks on import venues.	The system should display import venues page.	The system displays import venues' page.	Pass
6	Admin clicks on import clashes.	The system should display import clashes page.	The system displays import clashes' page.	Pass
7	Admin clicks on import preferences.	The system should display import preferences page.	The system displays import preference' s page.	Pass

Post-Condition:

Admin successfully imported resources.

[TC- 14] Import Teachers

Test Case #:	TC – 14
System:	Automated Timetable Maker
Test Case Name:	Import Teachers
Related Requirement:	FR-3.3.2.2

Short Description:	Test that admin can import teachers.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on import resources.	The system should display a dropdown including import teachers, import students, import venues, import courses, import clashes, import preferences.	The system displays a dropdown including import teachers, import students, import venues, import courses, import clashes, import preferences.	Pass
2	Admin clicks on import teachers.	The system should display import teachers page.	The system displays import teachers' page.	Pass
3	Admin clicks on import teacher' s button.	The system should import teachers into	The system imports teachers into database.	Pass

		database.		
4	Admin check database.	The system should display imported teachers.	The system displays imported teachers.	Pass

Post-Condition:

Admin successfully imported teachers.

[TC- 15] Export Resources

Test Case #:	TC – 15
System:	Automated Timetable Maker
Test Case Name:	Export Resources
Related Requirement:	FR-3.3.2.2
Short Description:	Test that admin can export resources.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on export resources.	The system should display a dropdown	The system displays a dropdown	Pass

		including export teachers, export students, export venues, export courses, export clashes, export preferences.	including export teachers, export students, export venues, export courses, export clashes, export preferences.	
2	Admin clicks on export students.	The system should display export students page.	The system displays export student' s page.	Pass
3	Admin clicks on export courses.	The system should display export courses page.	The system displays export courses' page.	Pass
4	Admin clicks on export teachers.	The system should display export teachers page.	The system displays export teachers' page.	Pass
5	Admin clicks on export venues.	The system should display export venues page.	The system displays export venues' page.	Pass
6	Admin clicks on export clashes.	The system should display export clashes page.	The system displays export clashes' page.	Pass
7	Admin clicks on export preferences.	The system should display export preferences page.	The system displays export preference' s page.	Pass

Post-Condition:

Admin successfully exported resources.

[TC- 16] Export Courses

Test Case #:	TC – 16
System:	Automated Timetable Maker
Test Case Name:	Export Courses
Related Requirement:	FR-3.3.2.2
Short Description:	Test that admin can export courses.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on export resources.	The system should display a dropdown including export teachers, export students, export venues, export	The system displays a dropdown including export teachers, export students, export venues, export	Pass

		courses, export clashes, export preferences.	courses, export clashes, export preferences.	
2	Admin clicks on export courses.	The system should display export courses page.	The system displays export courses' page.	Pass
3	Admin clicks on export course' s button.	The system should export courses from database to a folder.	The system exports courses from database to a folder.	Pass
4	Admin check database.	The system should display exported courses.	The system displays exported courses.	Pass

Post-Condition:

Admin successfully exported courses.

[TC- 17] View Resources

Test Case #:	TC – 17
System:	Automated Timetable Maker
Test Case Name:	View Resources
Related Requirement:	FR-3.3.2.10
Short Description:	Test that admin can view resources.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on view resources.	The system should display a dropdown including view teachers, view students, view venues, view courses, view clashes, view preferences.	The system displays a dropdown including view teachers, view students, view venues, view courses, view clashes, view preferences.	Pass
2	Admin clicks on view students.	The system should display view students page.	The system displays view student' s page.	Pass
3	Admin clicks on view courses.	The system should display view courses page.	The system displays view courses' page.	Pass
4	Admin clicks on view teachers.	The system should display view teachers page.	The system displays view teachers' page.	Pass

5	Admin clicks on view venues.	The system should display view venues page.	The system displays view venues' page.	Pass
6	Admin clicks on view clashes.	The system should display view clashes page.	The system displays view clashes' page.	Pass
7	Admin clicks on view preferences.	The system should display view preferences page.	The system displays view preference' s page.	Pass

Post-Condition:

Admin successfully viewed resources.

[TC- 18] View Clashes

Test Case #:	TC – 18
System:	Automated Timetable Maker
Test Case Name:	View Clashes
Related Requirement:	FR-3.3.2.10
Short Description:	Test that admin can view clashes.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.

2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on view resources.	The system should display a dropdown including view teachers, view students, view venues, view courses, view clashes, view preferences.	The system displays a dropdown including view teachers, view students, view venues, view courses, view clashes, view preferences.	Pass
2	Admin clicks on view clashes.	The system should display view clashes page.	The system displays view clashes' page.	Pass

Post-Condition:

Admin successfully viewed clashes.

[TC- 19] Delete Resources

Test Case #:	TC – 19
System:	Automated Timetable Maker
Test Case Name:	Delete Resources
Related Requirement:	FR-3.3.2.11

Short Description:	Test that admin can delete resources.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on delete resources.	The system should display a dropdown including delete teachers, delete students, delete venues, delete courses, delete clashes, and delete preferences.	The system displays a dropdown including delete teachers, delete students, delete venues, delete courses, delete clashes, and delete preferences.	Pass
2	Admin clicks on delete students.	The system should display delete students page.	The system displays delete student' s page.	Pass
3	Admin clicks on delete courses.	The system should display delete courses	The system displays delete courses' page.	Pass

		page.		
4	Admin clicks on delete teachers.	The system should display delete teachers page.	The system displays delete teachers' page.	Pass
5	Admin clicks on delete venues.	The system should display delete venues page.	The system displays delete venues' page.	Pass
6	Admin clicks on delete clashes.	The system should display delete clashes page.	The system displays delete clashes' page.	Pass
7	Admin clicks on delete preferences.	The system should display delete preferences page.	The system displays delete preference' s page.	Pass

Post-Condition:

Admin successfully deleted resources.

[TC- 20] Delete Preferences

Test Case #:	TC – 20
System:	Automated Timetable Maker
Test Case Name:	Delete Preferences
Related Requirement:	FR-3.3.2.11
Short Description:	Test that admin can delete preferences.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on delete resources.	The system should display a dropdown including delete teachers, delete students, delete venues, delete courses, delete clashes, and delete preferences.	The system displays a dropdown including delete teachers, delete students, delete venues, delete courses, delete clashes, and delete preferences.	Pass
2	Admin clicks on Delete preferences.	The system should display delete preferences page.	The system displays delete preferences page.	Pass
3	Admin search for specific preferences.	The system should display its details.	The system displays its details.	Pass
4	Admin click on delete button.	The system should display message successfully	The system displays message successfully	Pass

		deleted and should be deleted from database.	deleted and deleted from database.	
--	--	--	------------------------------------	--

Post-Condition:

Admin successfully deleted preferences.

[TC- 21] Generate Timetable

Test Case #:	TC – 21
System:	Automated Timetable Maker
Test Case Name:	Generate Timetable
Related Requirement:	FR-3.3.2.6
Short Description:	Test that admin can generate timetable.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Admin has an account in automated timetable maker.
2. Admin must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on generate timetable.	The system should loaded courses, teachers, venues and students.	The system load courses, teachers, venues and students.	Pass

2	Admin selects number of timetable to be generated.	The system should generate the selected number of timetable.	The system displays generated timetable.	Pass
3	Admin click on view different timetables.	The system should display timetable.	The system displays timetable.	Pass
4	Admin select three timetables from all generated.	The system should select three and save.	The system selected three and stored.	Pass
5	Admin send timetable for voting.	The system should send timetable to teachers.	The system sent timetable to teachers.	Pass

Post-Condition:

Admin successfully generated timetable.

[TC- 22] Voting for timetable

Test Case #:	TC – 22
System:	Automated Timetable Maker
Test Case Name:	Voting for timetable
Related Requirement:	FR-3.3.4.5
Short Description:	Test that teacher can vote for timetable.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Teacher has an account in automated timetable maker.
2. Teacher must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Teacher click on view timetable.	The system should display timetable.	The system displays timetable.	Pass
2	Teacher vote timetable.	The system should store in database and should send to admin.	The system stored in database and sent to admin.	Pass

Post-Condition:

Teacher successfully voted.

[TC- 23] Report Clashes

Test Case #:	TC – 23
System:	Automated Timetable Maker
Test Case Name:	Report Clashes
Related Requirement:	FR-3.3.3.6
Short Description:	Test that student can report clashes.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Student has an account in automated timetable maker.
2. Student must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Student click on report clashes.	The system should display report clashes page.	The system displays report clashes page.	Pass
2	Student write details in the form and click on Report clashes button.	The system should store details in database and should send to admin.	The system stored details in database and sent to admin.	Pass

Post-Condition:

Student successfully reported clash.

[TC- 24] Add Preferences

Test Case #:	TC – 24
System:	Automated Timetable Maker
Test Case Name:	Add Preferences
Related Requirement:	FR-3.3.4.6
Short Description:	Test that teacher can add preferences.
Designed Date:	3/12/2018
Execution Date:	3/12/2018

Pre-Condition:

1. Teacher has an account in automated timetable maker.
2. Teacher must be logged in.

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Teacher click on add preferences.	The system should display add preferences page.	The system displays add preferences page.	Pass
2	Student write details in the form and click on add preferences button.	The system should store details in database and should send to admin.	The system stored details in database and sent to admin.	Pass

Post-Condition:

Teacher successfully added preferences.

[TC- 25] Evaluate Timetable

Test Case #:	TC – 25
System:	Automated Timetable Maker
Test Case Name:	Evaluate Timetable
Related Requirement:	FR-3.3.2.9
Short Description:	Test that admin can evaluate timetable.

Designed Date:	3/12/2018
Execution Date:	3/12/2018

<p><u>Pre-Condition:</u></p> <ol style="list-style-type: none"> 1. Admin has an account in automated timetable maker. 2. Admin must be logged in.
--

Steps	Action	Expected Output	Actual output	Pass/Fail
1	Admin click on evaluate timetable.	The system should display voting result on evaluate timetable page.	The system displays voting result on evaluate timetable page.	Pass
2	Admin selects timetable with highest votes.	The system should select timetable.	The system selects timetable.	Pass
3	Admin send final to timetable to teacher and students.	The System should send timetable to teachers and students.	The System sent timetable to teachers and students.	Pass

<p><u>Post-Condition:</u></p> <p>Admin successfully evaluated timetable.</p>

6.3 Performance and Evaluation: Consist of results and comparisons

The test cases written were performed to evaluate whether the system conforms to the intended functionality. The results showed that all the test cases performed were passed and that the actual behavior matched the expected behavior.

6.4 Summary

It helped us gain confidence in our system. All the test cases that were performed showed that the system behaved as expected. It helped us to decide whether the system is acceptable or needs further refinement.

Reference and Bibliography

- [1] <https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3>
- [2] <https://www.sciencedirect.com/science/article/pii/S0304397508002028>
- [3] https://ac.els-cdn.com/S0304397501001827/1-s2.0-S0304397501001827-main.pdf?_tid=e17b45a0-d717-43b3-8194-64060dd6acac&acdnat=1544680652_2303f14468ce1ab0eed68682f06bc92e
- [4] <https://timetable.cranfield.ac.uk/CDS.html>
- [5] <https://lalescu.ro/liviu/fet/>
- [6] <http://www.mimosasoftware.com/>