Real Time Biometric based Students Attendance Management System

By

M. Adeel Waris

01071911005

Supervised By

Dr. Umer Rashid

Department of Computer Science

Quaid-i-Azam University

Islamabad

Batch (2019-2021)

ACKNOWLEDGEMENT

Primarily, I would like to thank Allah Almighty, who made me able to work on this project and showered his blessings upon me. Then I would like to thank several people for their support. A special thanks to my supervisor Dr. Umer Rashid whose valuable guidance has been the ones that helped me patched this project. His suggestions and his instructions has served as the major contributor towards the completion of the project. Then I would like to thank my family who supported me during this journey. Last but not the least I would like to thank my classmates who have helped me a lot.

ABSTRACT

Face is the representation of one’s identity. Hence, we have proposed a real-time biometric student attendance management system. It is a web application to record and manage students' attendance in real time scenarios. The student’s attendance will be marked in real-time via a biometric device installed in each classroom. The Staffs can see an immediate list of students present in a class. Also, the Staffs and administration can log-in to perform attendance related management activities via the application.

The students can only view their attendance records and notifications. The application also sends present, absent, short attendance notifications to the students via SMS, email, and WhatsApp messages. The admin may check all attendance data, see the defaulter list, search for student attendance by name, search class attendance, and generate required reports. The application maintains a complete record of sessions, classes, Staffs, and students for administrative purposes.

Contents

Chapter 1
Software Project Management Plan

1.1 Introduction

This chapter first introduces the description of Real Time Biometric Student Attendance Management System. It highlights the problem that has been addressed in this work along with the designed and developed solution. It also elaborates project organization and project planning. Finally, this chapter explains the scope and objectives of the project.

1.2 Problem Statement

Traditional student attendance marking technique is often facing a lot of trouble. The face recognition student attendance system emphasizes its simplicity by eliminating classical student attendance marking technique such as calling student names or checking respective identification cards. There are not only disturbing the teaching process but also causes distraction for students during exam sessions. Apart from calling names, attendance sheet is passed around the classroom during the lecture sessions. The lecture class especially the class with a large number of students might find it difficult to have the attendance sheet being passed around the class. Thus, face recognition student attendance system is proposed in order to replace the manual signing of the presence of students which are burdensome and causes students get distracted in order to sign for their attendance. Furthermore, the face recognition based automated student attendance system able to overcome the problem of fraudulent approach and lecturers does not have to count the number of students several times to ensure the presence of the students.

1.3 Project Description

A real-time biometric student attendance management system is a web application to record and manage students' attendance in real time scenarios. The student's attendance will be marked in real-time via a biometric device installed in each classroom. The Staffs can see an immediate list of students present in a class. Also, the Staffs and administration can log-in to perform attendance related management activities via the application.

The students can only view their attendance records and notifications. The application also sends present, absent, short attendance notifications to the students via SMS, email, and WhatsApp messages. The admin may check all attendance data, see the defaulter list, search for student attendance by name, search class attendance, and generate required reports. The application maintains a complete record of sessions, classes, Staffs, and students for administrative purposes.

1.4 Major Functionalities

Main functionalities of this tool are:-

a) To detect the face of student for attendance.
b) To record and manage student's attendance.
c) App will send notifications to student in case of absent or short.
d) Allow students to check for their attendance record.
e) To let Staff see list of students present in class.
f) To provide a comprehensive and details view to admin.

1.5 Objective

Main objectives to develop this tool are:-

a) To detect the student face through camera.
b) To record and manage student's attendance
c) To record sessions, classes, Staffs and students data for administrative purpose.
d) To provide the teacher an interface where he can see all student's attendance w.r.t to their Programs.

1.6 Tools and Techniques

It would be web based application and I will use following tools in development:-

a) Html, CSS, Javascript
b) Project Libre (for Plan)
c) Start UML (for Diagrams)
d) MS Word (for Documentation)
e) Python, Machine Learning, Django
f) mySQL (Database)

1.7 Users

Mainly this application will be used by:

a) **Students**:
   Students who are registered in the app will be able to use it. First time student will be assigned an ID/Password by administration after registration and then they will simply login. And they can see their attendance w.r.t to their Programs.

b) Staff:
   Administration will assign ID/Password to Staff first time and then they will simply login. And Staff will be able to capture attendance through single click and will be able to see overall attendances w.r.t to their specified Programs.

c) Administration:
   Admin will be able to add Programs, students and Staffs. And will be able to see and generates respected reports.

1.8 Software Process Model

Waterfall model will be used for the development of this project because of the following reasons:-

a) Requirements are clearly defined.
b) It is easy to manage due to the rigidity of model.

1.9 Project Management Plan

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

- **Requirements Analysis Phase:**

- **Task 1: Identify Requirements**
  - **Task Description**
    The initial step in the development of this project is the identification of requirements. The requirements include functional and non-functional requirements.
  - **Deliverables and Milestones**
    Requirements are collected and reviewed.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: MS Word
    Hardware: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- **Task 2: Define Use cases**
  - **Task Description**
    This task includes defining and writing use cases and making use case diagram.
  - **Deliverables and Milestones**
    Use cases are written and reviewed.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: MS Word, Star UML
    Hardware: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- **Task 3: Develop Domain Model**
  - **Task Description**
    This task includes making domain model for the system.
  - **Deliverables and Milestones**
    Domain model is reviewed.

- **Resources Needed**
  People: Adeel Waris, Dr Umar
  Software: Star UML
  Hardware: Laptop
- **Dependencies and Constraints**
  None
- **Risk and Contingencies**
  None

- **Task 4: Develop ERD**
  - **Task Description**
    This task includes making database design in the form of ERD.
  - **Deliverables and Milestones**
    ERD is reviewed.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar
    Software: Star UML
    Hardware: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- **Task 5: Develop Complete SRS**
  - **Task Description**
    This task includes making Software Requirements Specification document which contains the description of functional and non-functional requirements.
  - **Deliverables and Milestones**
    SRS document is reviewed.
    Requirement Analysis phase is complete.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: MS Word
    Hardware: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- **Design Phase:**

- **Task 1: Develop Design**
  - **Task Description**
    This task includes the development of architectural design and detailed design of the system.
  - **Deliverables and Milestones**
    Architecture diagram is reviewed.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar
    Software: Star UML
    Hardware: Laptop
  - **Dependencies and Constraints**
    The development of designs requires the completion of previous phase.
  - **Risk and Contingencies**
    None

- **Task 2: Develop Interfaces**
  - **Task Description**
    This task includes develop the expected interfaces of the project.
  - **Deliverables and Milestones**
    Interfaces are reviewed.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar
    Software: Marvel App Online
    Hardware: Laptop
  - **Dependencies and Constraints**
    Architectural Design of the system should be complete.
  - **Risk and Contingencies**
    None

- **Task 3: Develop Class and Sequence Diagrams**
  - **Task Description**
    This task includes develop the Class and Sequence Diagrams.
  - **Deliverables and Milestones**
    Class and Sequence Diagram is reviewed.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: Star UML, MS Word
    Hardware: Laptop
  - **Dependencies and Constraints**
    Architectural Design of the system should be complete.
  - **Risk and Contingencies**

    None

- **Task 4: Evaluate Design**
  - **Task Description**
    This task includes evaluation and verification of the design.
  - **Deliverables and Milestones**
    Design phase is completed.
    Design document is reviewed
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: MS Word
    Hardware: Laptop
  - **Dependencies and Constraints**
    Architectural Design and interfaces of the system should be complete.
  - **Risk and Contingencies**
    None
- **Task 5: Develop Software Test Documentation**
  - **Task Description**
    This task includes defining test cases for the system.
  - **Deliverables and Milestones**
    Test cases are reviewed.
    Design phase is complete
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: MS Word
    Hardware: Laptop
  - **Dependencies and Constraints**
    To define the test cases, design phase should be complete.
  - **Risk and Contingencies**
    None

• Implementation Phase

- **Task 1: System Implementation**
  - **Task Description**
    This phase includes the development of application.
  - **Deliverables and Milestones**
    Implementation of application is complete.
  - **Resources Needed**
    People: Adeel Waris, Dr Umar Rashid
    Software: pyCharm, MS Word, MySQL, Visual Studio code
    Hardware: Laptop
  - **Dependencies and Constraints**
    The development of application requires the completion of design phase.
  - **Risk and Contingencies**

    None

1.10 Timetable and Gantt chart

This section describes the timetable of the Project.

		Name	Duration	Start	Finish
1		**Attendance Managem...**	**175.75 days**	**12/10/20 8:00 AM**	**8/12/21 3:00 PM**
2		Project Desc. Finalization	10 days	12/10/20 8:00 AM	12/23/20 5:00 PM
3		Project Undestanding	20 days	12/24/20 8:00 AM	1/20/21 5:00 PM
4		**Management Plan**	**21 days**	**12/10/20 8:00 AM**	**1/7/21 5:00 PM**
5		Define Scope and obje...	5 days	12/10/20 8:00 AM	12/16/20 5:00 PM
6		Define Users	5 days	12/17/20 8:00 AM	12/23/20 5:00 PM
7		Choose process model	3 days	12/24/20 8:00 AM	12/28/20 5:00 PM
8		Gantt chart	3 days	12/29/20 8:00 AM	12/31/20 5:00 PM
9		Plan	4 days	1/1/21 8:00 AM	1/6/21 5:00 PM
10		Write and Review	21 days	12/10/20 8:00 AM	1/7/21 5:00 PM
11		**SRS**	**22 days**	**1/8/21 8:00 AM**	**2/8/21 5:00 PM**
12		Identify software syst...	2 days	1/8/21 8:00 AM	1/11/21 5:00 PM
13		Identify use cases	5 days	1/12/21 8:00 AM	1/18/21 5:00 PM
14		Use case diagram	3 days	1/19/21 8:00 AM	1/21/21 5:00 PM
15		Review use case diagram	1 day	1/22/21 8:00 AM	1/22/21 5:00 PM
16		Use case text	7 days	1/25/21 8:00 AM	2/2/21 5:00 PM
17		Domain model	3 days	2/3/21 8:00 AM	2/5/21 5:00 PM
18		Review	1 day	2/8/21 8:00 AM	2/8/21 5:00 PM
19		**Software Design Cha...**	**20.25 days**	**2/3/21 3:00 PM**	**3/3/21 5:00 PM**
20		Architecture Diagram	3 days	2/9/21 8:00 AM	2/11/21 5:00 PM
21		Sequence Diagram	3 days	2/12/21 8:00 AM	2/16/21 5:00 PM
22		Review	1 day	2/17/21 8:00 AM	2/17/21 5:00 PM
23		Identify Classes	3 days	2/18/21 8:00 AM	2/22/21 5:00 PM
24		Class Diagram	3 days	2/23/21 8:00 AM	2/25/21 5:00 PM
25		Define User Interface	3 days	2/26/21 8:00 AM	3/2/21 5:00 PM
26		Review User Interfaec	1 day	3/3/21 8:00 AM	3/3/21 5:00 PM
27		Overall Review	1 day	2/3/21 3:00 PM	2/4/21 3:00 PM
28		**Implemetation**	**123 days**	**2/4/21 3:00 PM**	**7/27/21 3:00 PM**
29		Machine Leaning	30 days	2/4/21 3:00 PM	3/18/21 3:00 PM
30		review and update	1.5 days	3/18/21 3:00 PM	3/22/21 10:00 AM
31		Font end interface	30 days	3/22/21 10:00 AM	5/3/21 10:00 AM
32		Review and update	3 days	5/3/21 10:00 AM	5/6/21 10:00 AM
33		Backend Interface	30 days	5/6/21 10:00 AM	6/17/21 10:00 AM
34		Review and Update	3 days	6/17/21 10:00 AM	6/22/21 10:00 AM
35		Project Final Integration	10 days	7/13/21 3:00 PM	7/27/21 3:00 PM
36		**Software Testing**	**12 days**	**7/27/21 3:00 PM**	**8/12/21 3:00 PM**
37		Identify test cases	5 days	7/27/21 3:00 PM	8/3/21 3:00 PM
38		Testing	5 days	8/3/21 3:00 PM	8/10/21 3:00 PM
39		Review	2 days	8/10/21 3:00 PM	8/12/21 3:00 PM

Figure 1.1 Project Plan Time Table

1.11 Project Management Plan (Gantt chart) - a

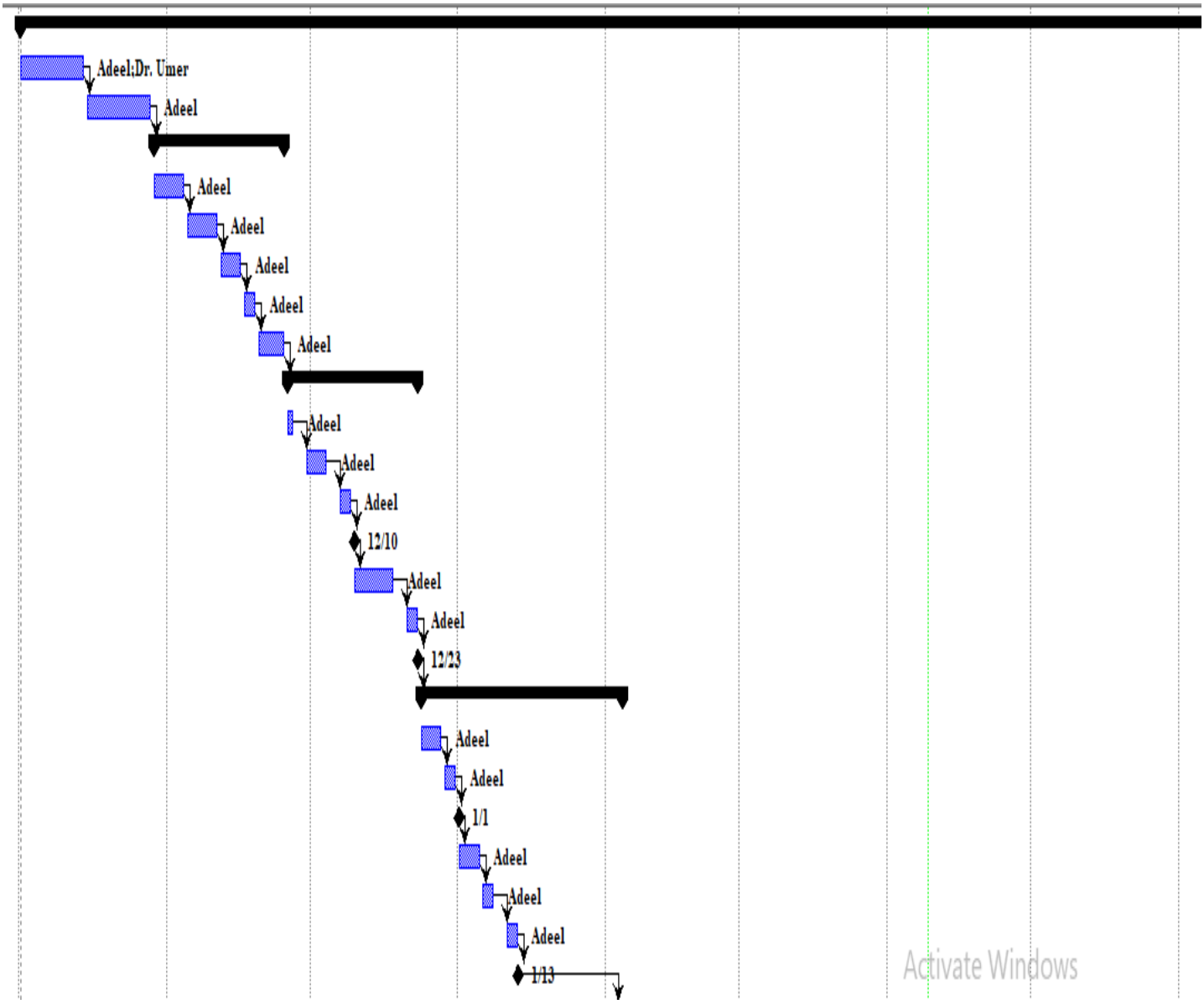


Figure 1.2 Project Management Plan (Gantt chart - a) of Student Management System

1.12 Project Management Plan (Gantt chart) - b

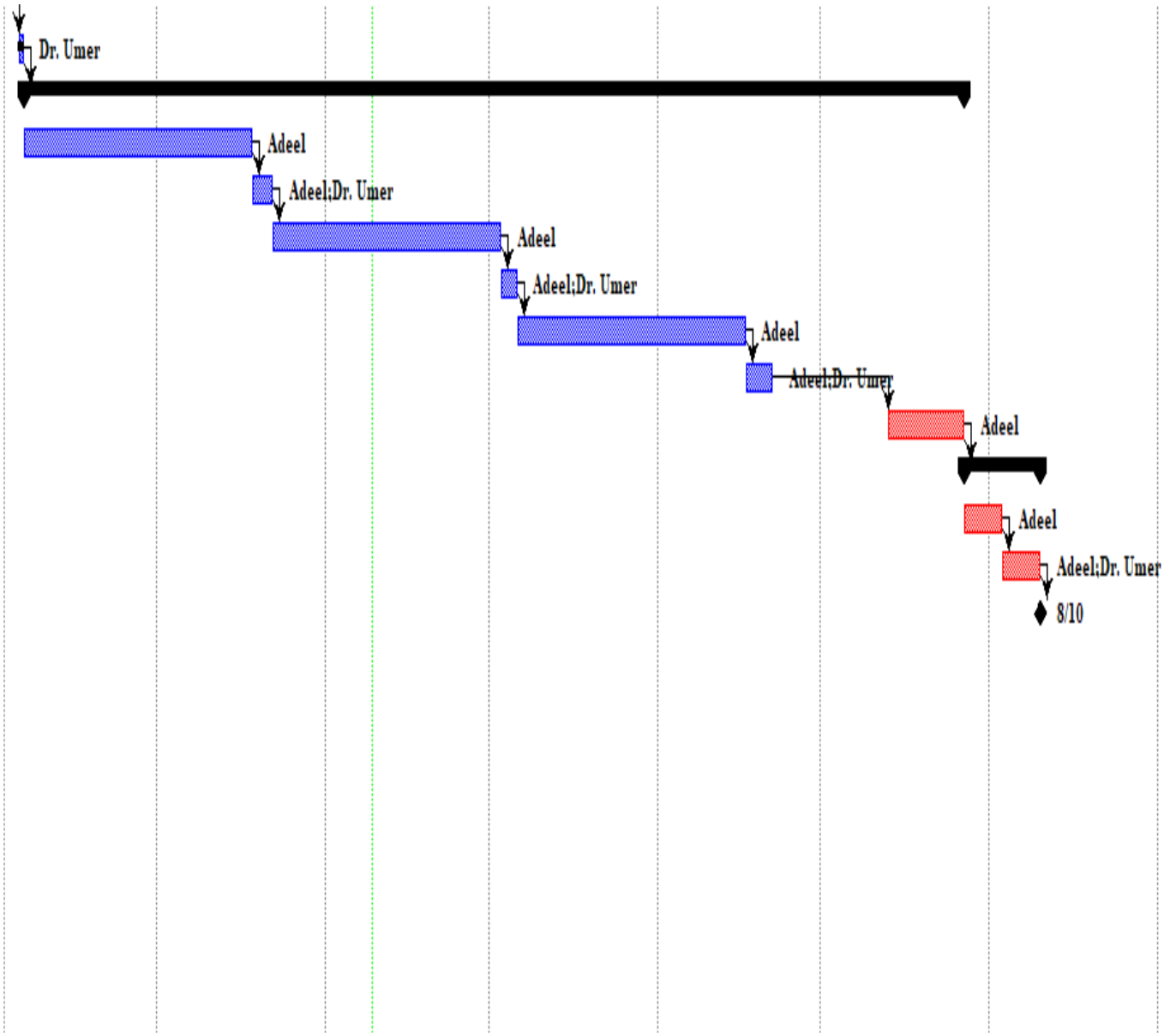


Figure 1.3 Project Management Plan (Gantt chart - b) of Student Management System

1.13 Summary

This chapter has briefly introduced the project description, scope, objectives and project management plan. This has also described the software process model which will be followed throughout the design and implementation process. Now in chapter 2, project's functional and non-functional requirements will be described.

Chapter 2
Software Requirements Specification

2.1 Introduction

This chapter first describes the software system attributes. It highlights the user characteristics and major constraints of the tool. It also elaborates the use cases and system sequence diagrams. Finally, this chapter explains the domain model and database design of the project.

2.2 Functional Requirements

- System will provide a proper interface for all Actors.
- System will provide an interface for student where he/she will be able to check their attendances and their Programs.
- System will provide an interface for Staff where Staff will be able to mark attendance through biometric device.
- Staff will also be able to check Programs and their attendance data
- System will provide an interface for admin where he will be able to generate different reports accordingly
- System will provide an option to the admin where admin will be able to add new student.
- System will provide an option to the admin where admin will be able to add new Staff
- System will provide an option to the admin where admin will be able to add new Class.
- System will provide an option to the admin where admin will be able to add new Program.
- System will provide an option to the admin where admin will be able to manage students, Programs, classes and Staffs.
- All Id's and Passwords for students and Staffs will be generated by Admin for security reasons.
- System will provide Login and Logout interface to all users.

2.3 Non-Functional Requirements

System attributes are properties and characteristics of a system that describes the overall project performance. These attributes can be changed with evolvement of new values in the system. System should possess the following attributes:-

2.3.1 Reliability

Tool should be reliable. There should be no occurrence of the failure. The system should be able to work properly all-time, i.e., to the extent to which it works as and when needed. The system should give proper response against every mistake performed by user.

2.3.2 Availability

Tool should be available all the time with right occurrence of each resource.at any given time. System should be available to user at any time. It also includes the availability of internet connection and resources the system is providing.

2.3.3 Portability

This is an android-based application. User can carry this tool anywhere they go but they must have an android device. Anyone who has an android device will be able to use this tool for their preferences. It runs on any of the android device having internet access.

2.3.4 Performance

Performance includes, project should be completed on time, made on allocated budget , can deal with numerous users at a time. Then system is said to have a good performance.

2.3.5 Security

User should only be able to access the system through his personal android device where he can use this tool. No members in the network can access the personal account/profile of any other user. The tool has its own login credentials to use it. User must enter his login credentials before using the system.

2.3.6 Maintainability

In some cases, maintainability involves continuous improvement in the system, learning from the past in order to improve the ability to maintain systems, or improve the reliability of systems on the basis of maintenance experience. The application should be easy to extend. The code should be written in a way that it favors implementation of new functions.

2.4 User Characteristics

This tool can be used by all the users.

User should be

- Familiar with basic knowledge of web applications
- Internet literate.

2.5 Constraints

User must have PC/Laptop having at least 2 GB of RAM and 500 MB of free disk space to smoothly run this application.

2.6 Major Inputs and Outputs

Major inputs and output of the application are described as follows:-

2.6.1 Major Inputs

Major inputs of the application are:-

a) Camera Input
b) Students Data
c) Staff Data
d) Login credentials
e) Personal Account Data (Name, Email, Phone)
f) Admin Data
g) Classes and Sessions Data

2.6.2 Major Outputs

Major outputs of the application are:-

a) Attendance taken
b) Notifications
c) Visual Data

2.7 Use Cases List

Use cases list, Use case Diagram and their Descriptions are as follows:

- Login
- Add Staff
- Manage Staff
- Add Student
- Manage Student
- Add Program
- Manage Program
- Add Class
- Mange Class
- Mark Attendance
- View Attendance
- Send Notifications
- View Notifications
- View Attendance Reports
- Capture Frames

2.8 Use Case Diagram

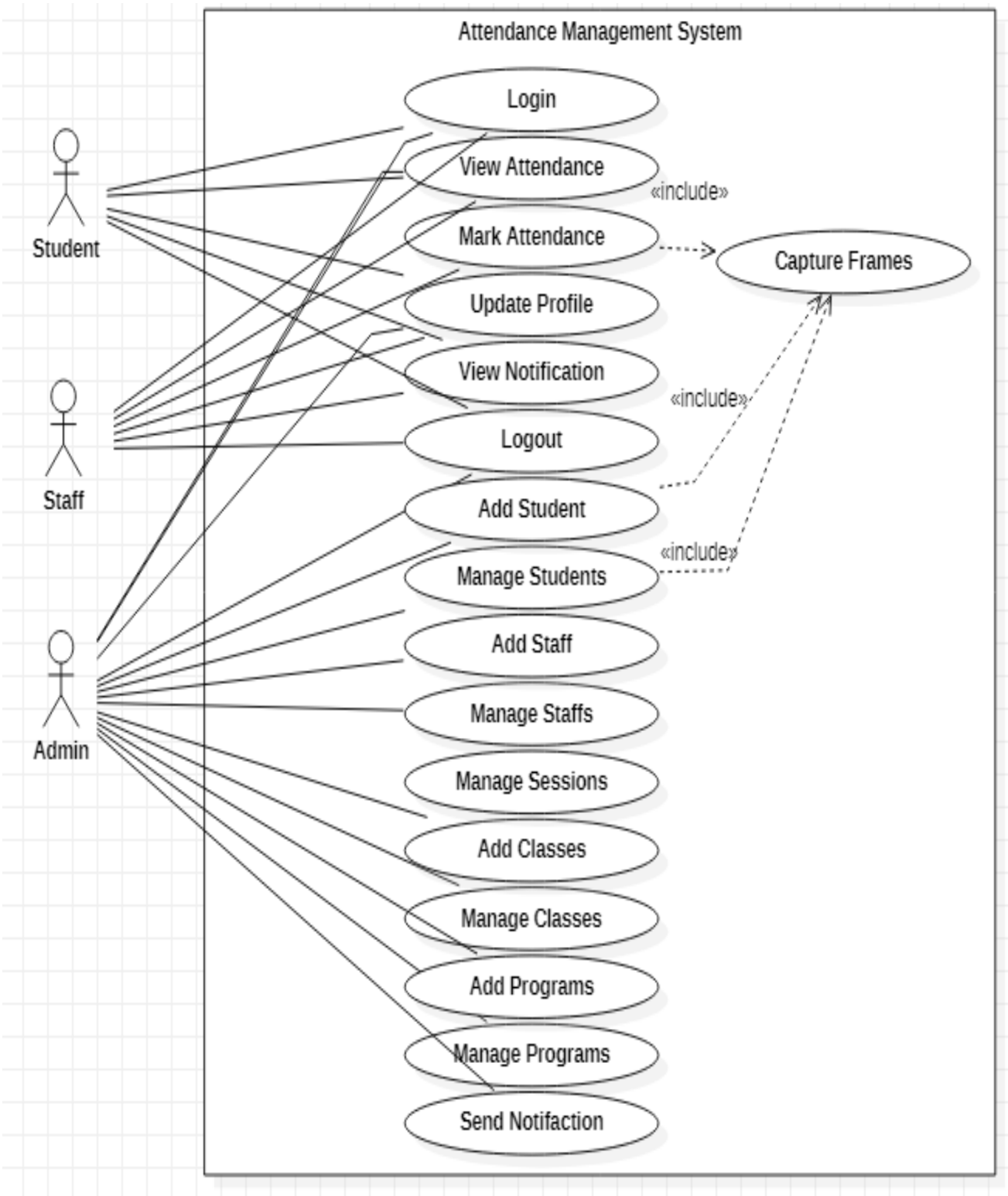


Figure 4 Use case Diagram of Student Management Syetm

2.9 Use Case Description

Above mentioned use cases are described in details as follows.

2.9.1 Login

Student/Staff/Admin can go to the log in option where he/she will enter username and password and others required information

Table 1 UCD for Login

ID	UC1
Name	Login
Primary Actor	Student, Admin, Staff
Pre-Condition	1) User has a Laptop/PC 2) User has already an assigned account 3) User's account has been verified by Admin.
Post Condition	1) User has login to the web application successfully. 2) System has displayed the homepage.
Main Success Scenario	1) User opens the application. 2) System displays the login screen. 3) User enters its credentials. 4) System validates and shows home page of application.
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application. 4a) User entered wrong username/password. 1. System prompts to enter username/password again. 2. User again enters the username/password and selects sign in option. 4b) User leaves required fields blank. 1. System prompts user to fill required fields. 2. User fills the required fields and selects sign in option again.
Frequency	Could be nearly continuous

2.9.2 Send Notification

Admin will be able to send notification to staff and students

Table 3 UCD for Send Notification

ID	UC3
Name	Send Notification
Primary Actor	Admin
Pre-Condition	Admin has selected Send Notification option
Post Condition	Admin has sent the notification successfully
Main Success Scenario	1) Admin selects the Send Notification 2) System displays the all staff and srudent data 3) Admin select specified user 4) System display a new form to type message 5) Admin type the message and select Send
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application. 2a) Attendance data is not available Contact admin for support
Frequency	Could be nearly continuous

2.9.3 View Attendance

Actor can see attendance data of student w.r.t to Programs and time

Table 3 UCD for View Attendance

ID	UC3
Name	View Attendance
Primary Actor	Admin, Staff,Student
Pre-Condition	1) User has selected Attendance Data
Post Condition	User has viewed the attendance desired data
Main Success Scenario	1) User selects the View Attendance 2) System displays the attendance data.
Alternative flows or Extensions	***Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application. 2a) Attendance data is not available Contact admin for support
Frequency	Could be nearly continuous

2.9.4 View Notification

Staff/Student will be able to View notification to staff and students

Table 3 UCD for View Notification

ID	UC3
Name	View Notification
Primary Actor	Staff, student
Pre-Condition	Actor has selected View Notification option
Post Condition	Actor read all the notifications
Main Success Scenario	1) Actor selects the View Notification 2) System displays the all the notifications sent to him
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application. 2a) Attendance data is not available Contact admin for support
Frequency	Could be nearly continuous

2.9.5 Mark Attendance

The Attendance will be marked via biometric channel when user will select the mark attendance option

ID	UC6
Name	Mark Attendance
Primary Actor	Staff
Pre-Condition	1) Staff has logged in to the web application
Post Condition	1) Attendance is marked
Main Success Scenario	1) User has selected Mark Attendance. 2) System will mark the attendance using biometric channel connected to the computer
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.6 Capture Frames

When admin will select the Capture Frames then Video Camera will be open to capture frames from video.

Table 5 UCD for Capture Frames

ID	UC7
Name	Capture Frames
Primary Actor	Admin, Staff
Pre-Condition	2) User has logged in to the web application
Post Condition	Frames has been Captured successfully.
Main Success Scenario	3) User has selected Capture Frames 4) Camera has started
Alternative flows or Extensions	***Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.7 Logout

Users can log out from the system by selecting the logout option.

Table 11 UCD for Logout

ID	UC8
Name	Logout
Primary Actor	Student, Staff, Admin
Pre-Condition	1) User has logged in to the system.
Post Condition	1) User has logged out from the application successfully. 2) System has displayed the login page.
Main Success Scenario	1) User selects logout option. 2) System displays the login screen.
Alternative flows or Extensions	***Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.8 Add New Student

Admin adds a new student with all required information

Table 10 UCD for add new student

ID	UC10
Name	Add New Student
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	1) Student has been added successfully
Main Success Scenario	1) Admin selects add new Student option 2) System displays the form where all information has to be added 3) Admin will fill the form and will select proceed 4) System will complete the registration
Alternative flows or Extensions	***Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.9 Add New Staff

Admin adds a new Staff with all required information

Table 11 UCD for add new Staff

ID	UC11
Name	Add New Staff
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	1) Staff has been added successfully
Main Success Scenario	1) Admin selects add new Staff option 2) System displays the form where all information has to be added 3) Admin will fill the form and will select proceed 4) System will complete the registration
Alternative flows or Extensions	***Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.10 Add New Program

Admin adds a new Program with all required information

Table 12 UCD for add new program

ID	UC12
Name	Add New Program
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	Program has been added successfully
Main Success Scenario	1) Admin selects add new program option 2) System displays the form where all information has to be added 3) Admin will fill the form and will select proceed 4) System will complete the registration
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.11 Add New Class

Admin adds a new Class with all required information

Table 13 UCD for add new Class

ID	UC13
Name	Add New Class
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	Class has been added successfully
Main Success Scenario	1) Admin selects add new Class option 2) System displays the form where all information has to be added 3) Admin will fill the form and will select proceed 4) System will complete the registration
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.12 Manage Students

Admin can Manage any students from registered list.

Table 14 UCD for Manage students

ID	UC14
Name	Manage Students
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	Student has been Updated/Deleted successfully
Main Success Scenario	1) Admin selects Manage Students option 2) System displays list consisting of all registered student. 3) Admin will select the specified student from list. 4) System will display the specified student data. 5) Admin will Manage that data 6) System will Update/Delete the specified student
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.13 Manage Staffs

Admin can edit or delete any Staff from registered list.

Table 15 UCD for Manage Staffs

ID	UC15
Name	Manage Staff
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	Student has been Updated/Deleted successfully
Main Success Scenario	1) Admin selects Manage Staff option 2) System displays list consisting of all registered Staff. 3) Admin will select the specified Staff from list. 4) System will display the specified Staff data. 5) Admin will Manage that data 6) System will Update/Delete the specified Staff
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.14 Manage Programs

Admin can edit or delete any Program from registered list.

Table 16 UCD for Manage Program

ID	UC16
Name	Manage Programs
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	Student has been Updated/Deleted successfully
Main Success Scenario	1) Admin selects Manage Programs option 2) System displays list consisting of all registered Program. 3) Admin will select the specified student from list. 4) System will display the specified Program data. 5) Admin will Manage that data 6) System will Update/Delete the specified Program
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.9.15 Manage Classes

Admin can edit or delete any class from registered list.

Table 17 UCD for Manage classes

ID	UC17
Name	Manage Classes
Primary Actor	Admin
Pre-Condition	1) Admin is logged in to the system.
Post Condition	Student has been Updated/Deleted successfully
Main Success Scenario	1) Admin selects Manage class option 2) System displays list consisting of all registered class. 3) Admin will select the specified student from list. 4) System will display the specified class data. 5) Admin will Manage that data 6) System will Update/Delete the specified class
Alternative flows or Extensions	*****Server down or Internet link down 1. User waits until internet and server is recovered. *Anytime system fails. 1. User restarts the application.
Frequency	Could be nearly continuous

2.10 System Sequence Diagrams

An interaction diagram which shows the sequence of interactions between the external actor and the system and the events generated by these actors is called System Sequence Diagram.

2.10.1 Mark Attendance

Figure 5 shows the system sequence diagram for Mark Attendance.

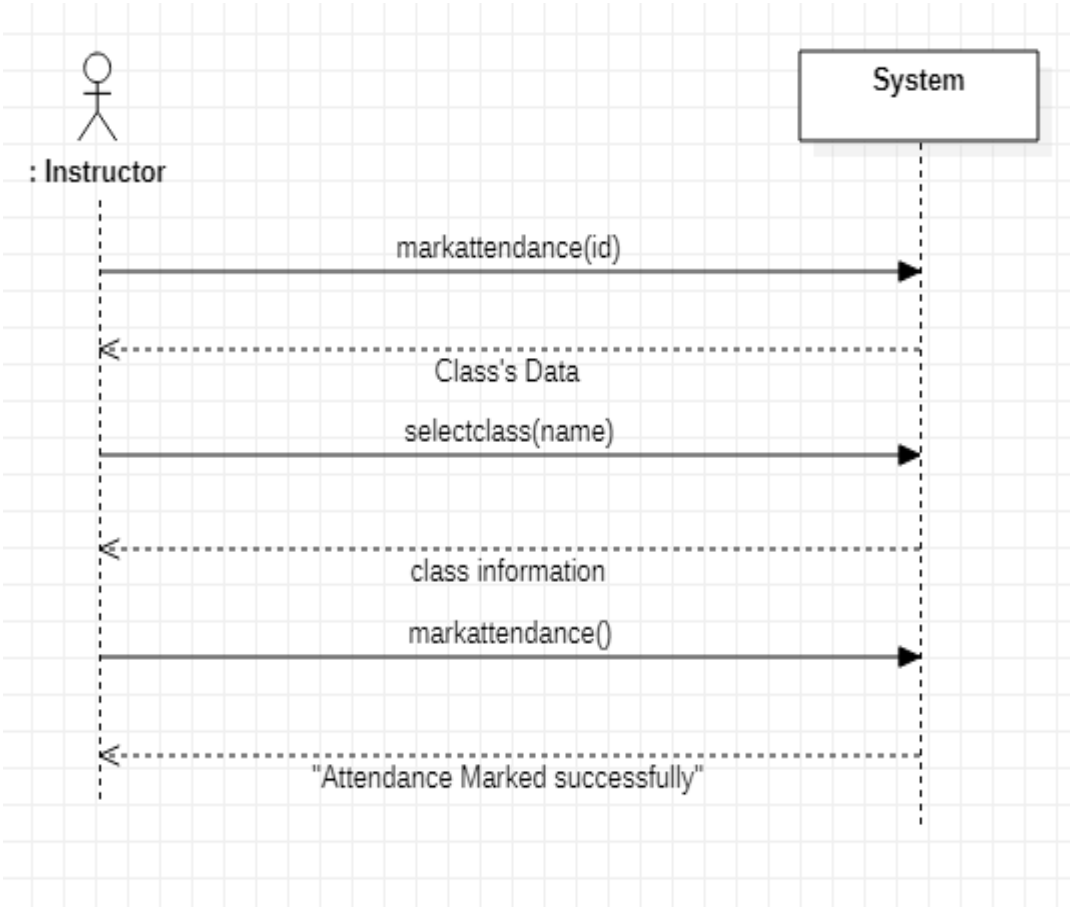


Figure 5 SSD for Mark Attendance

2.10.2 Login

Figure 6 shows the system sequence diagram of Login.

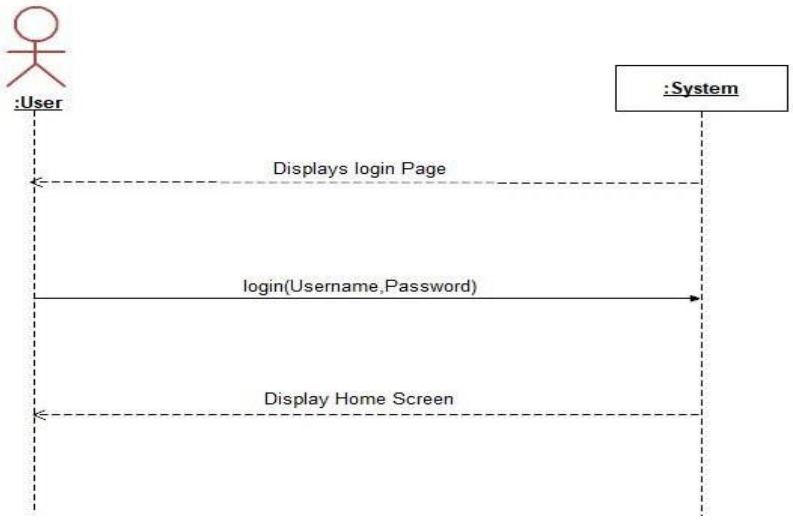


Figure 7 SSD for Login

2.10.3 Add new Student

Figure 8 shows the system sequence diagram of Add new Student.

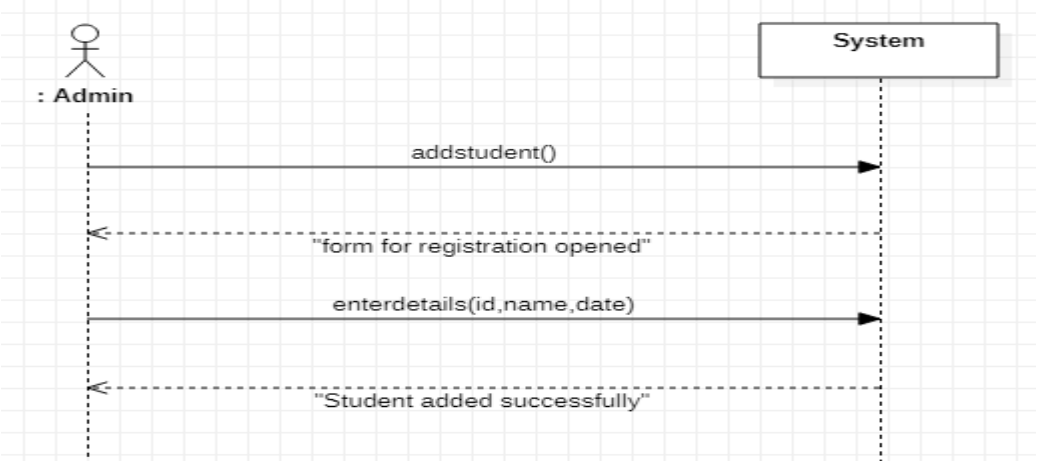


Figure 8 SSD for Add new student

2.10.4 Add new Staff

Figure 9 shows the system sequence diagram of Add new Staff.

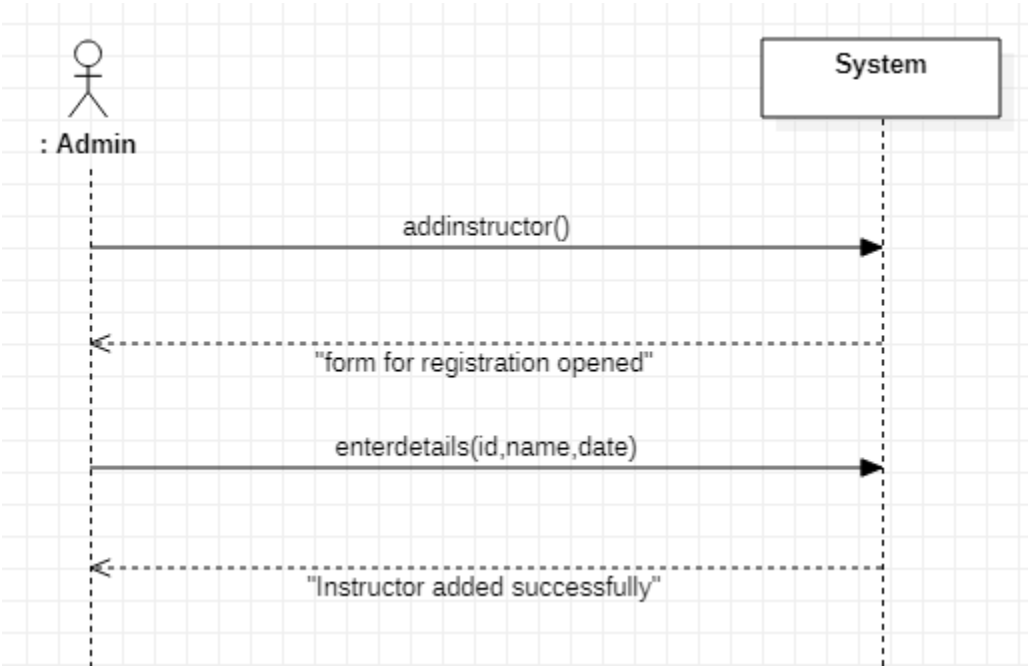


Figure 9 SSD for Add New Staff

2.10.5 Add new Program

Figure 10 shows the system sequence diagram of Add new Program.

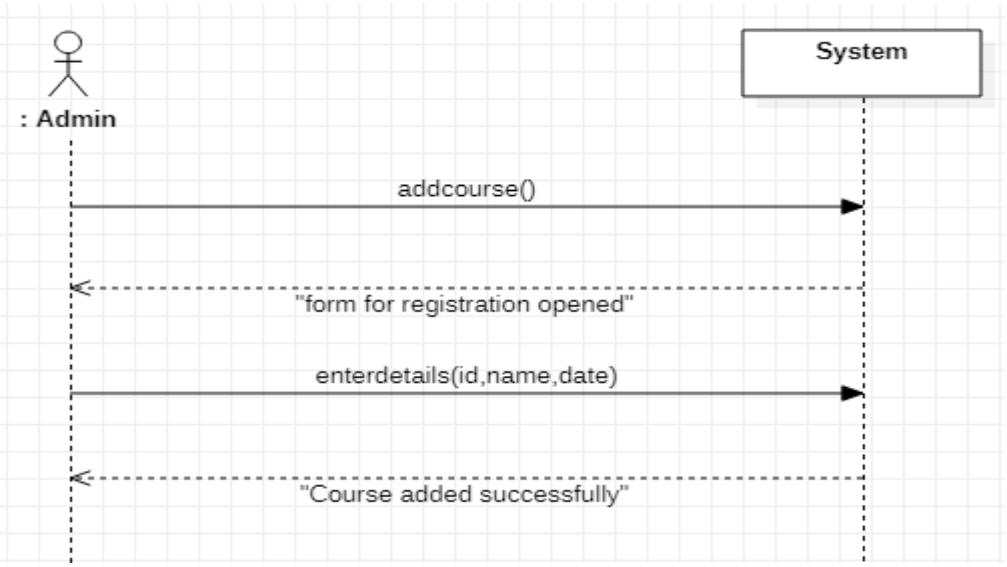


Figure 10 SSD for Add new Program

2.10.6 Add new Class

Figure 11 shows the system sequence diagram of Class.

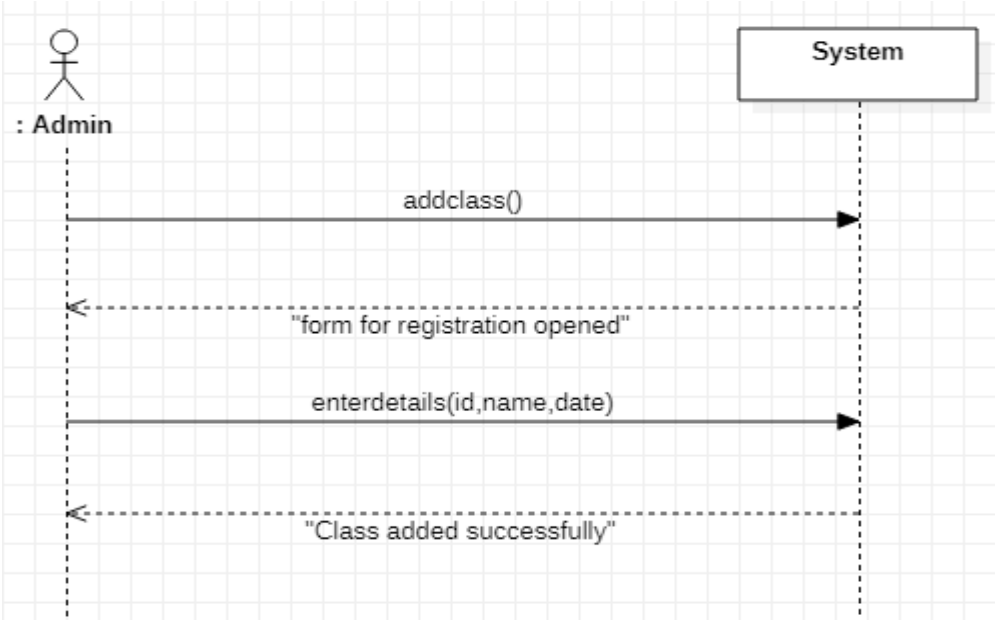


Figure 11 SSD for Class

2.10.7 Manage Students

Figure 12 shows the system sequence diagram of Manage Students.

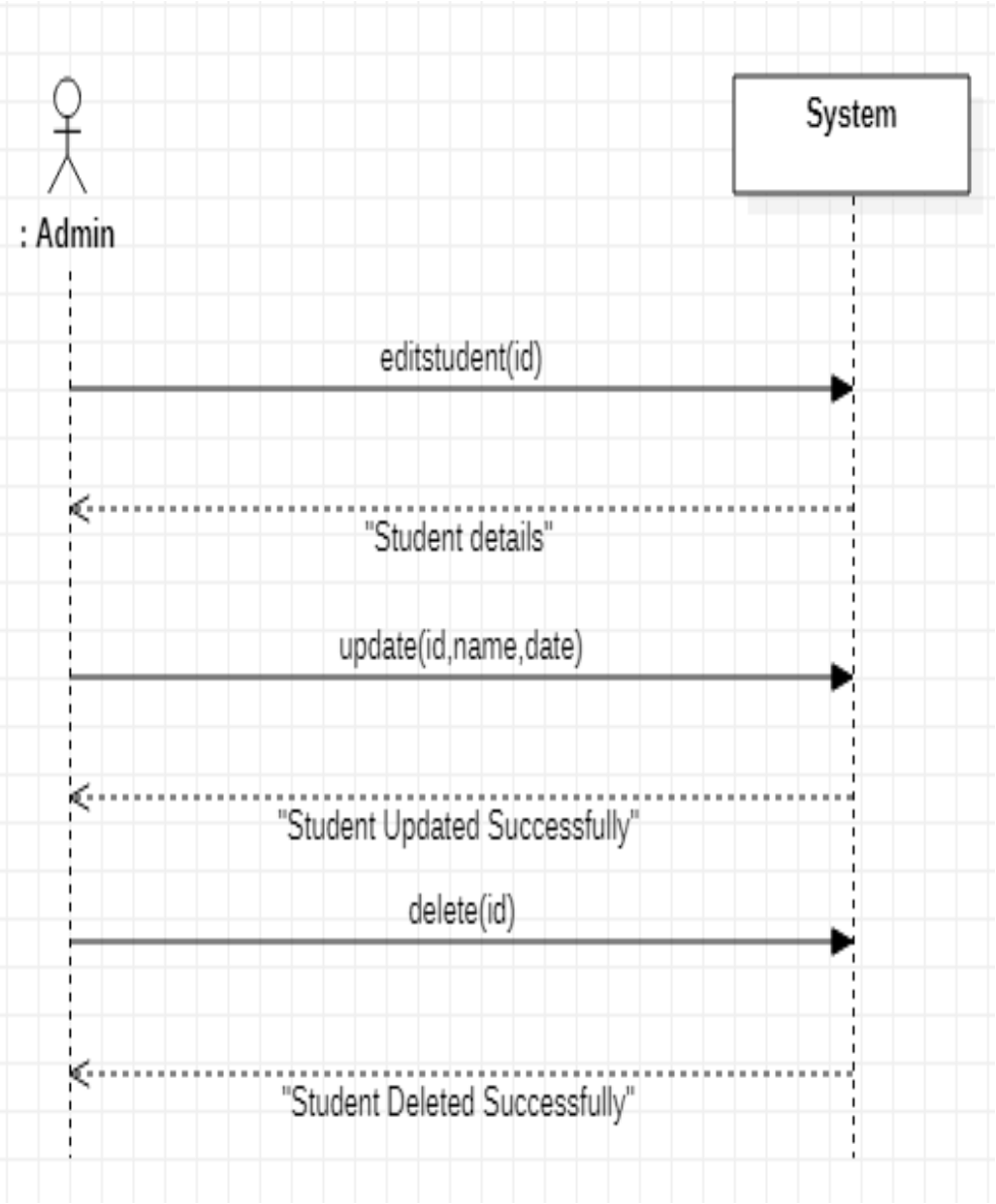


Figure 12 SSD for Manage Students

2.10.8 Manage Staffs

Figure 13 shows the system sequence diagram of Manage Staffs.

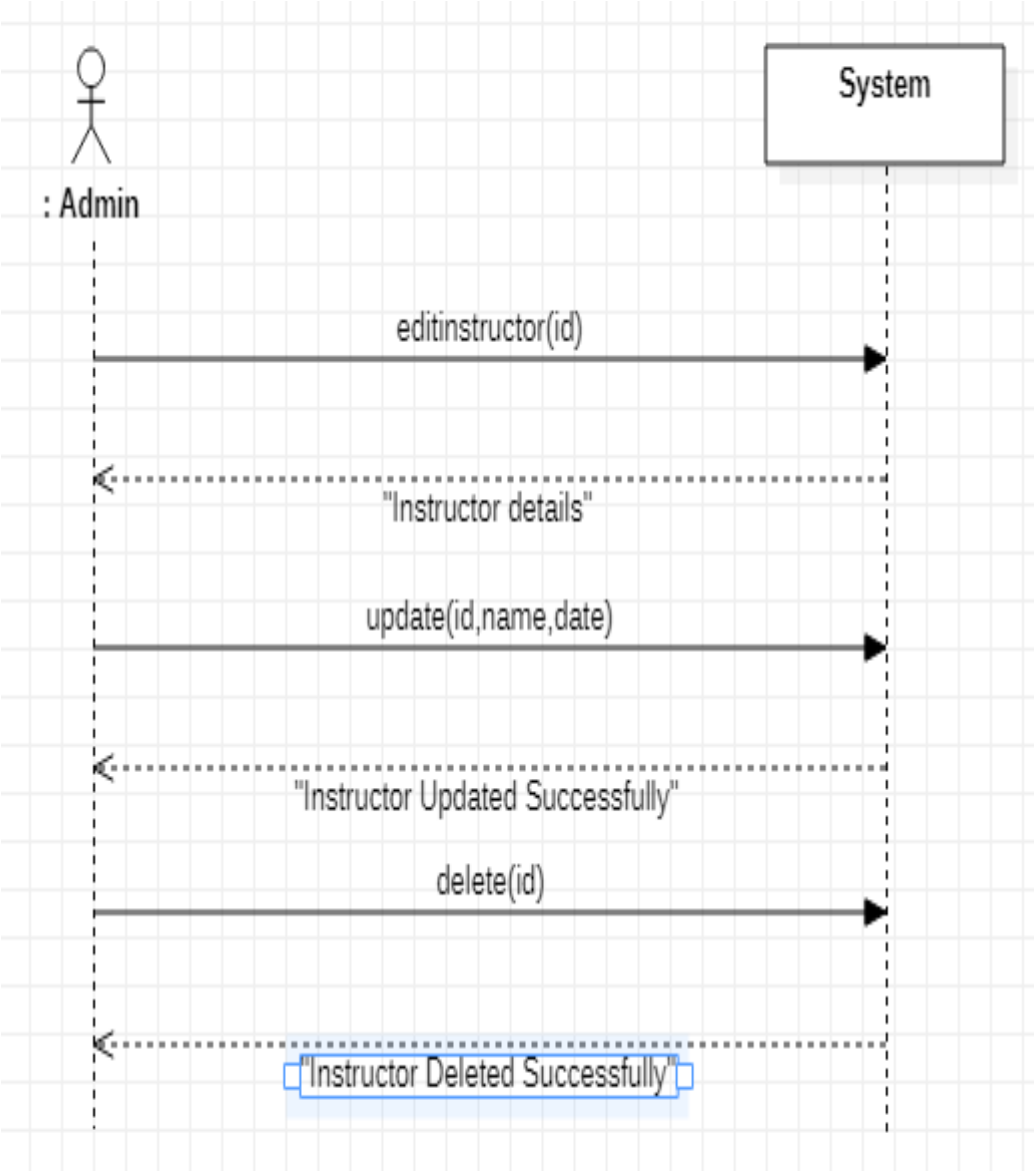


Figure 13 SSD for Manage Staffs

2.10.9 Manage Programs

Figure 14 shows the system sequence diagram of Manage Programs.

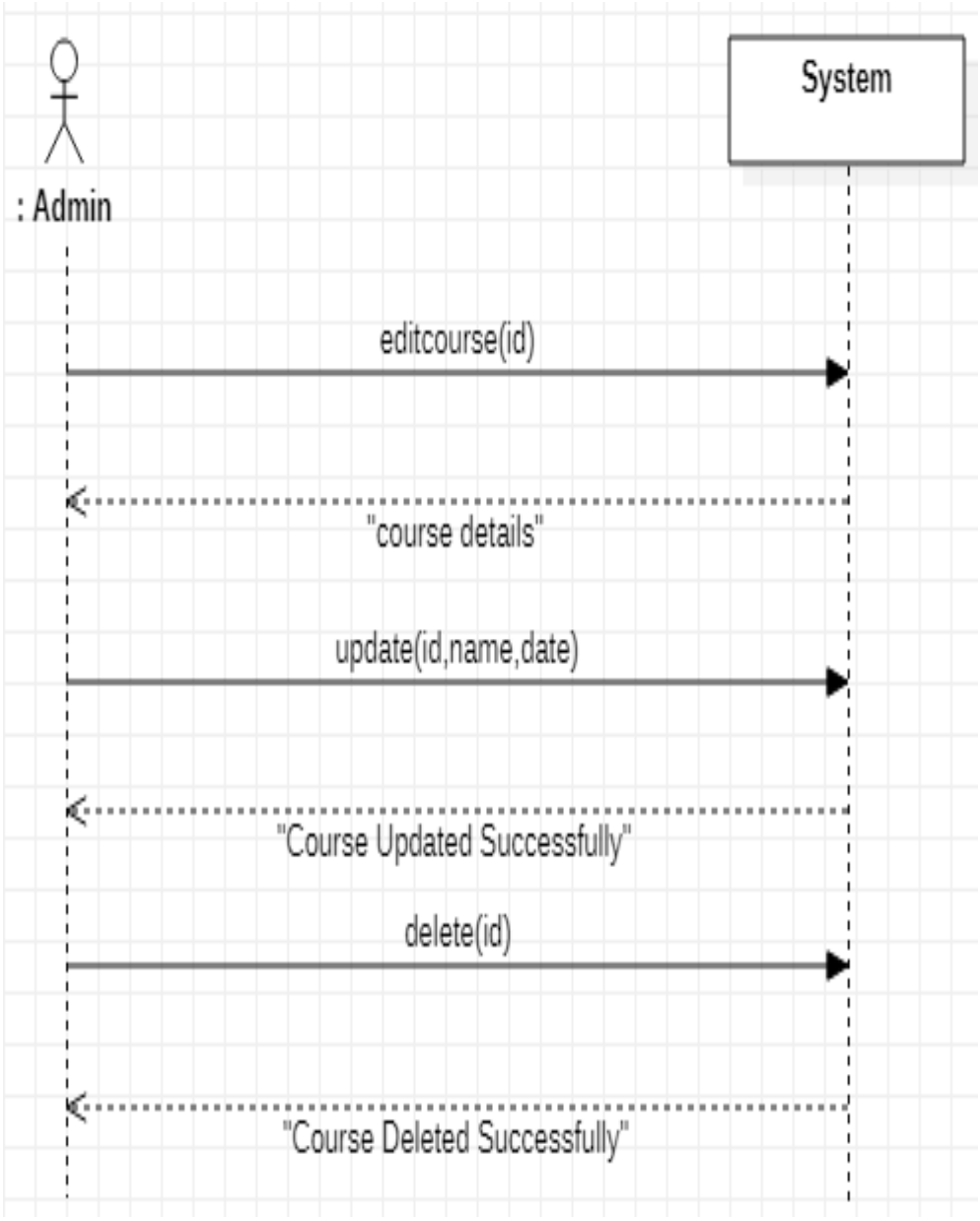


Figure 14 SSD for Manage Programs

2.10.10 Logout

Figure 15 shows the system sequence diagram of Logout.

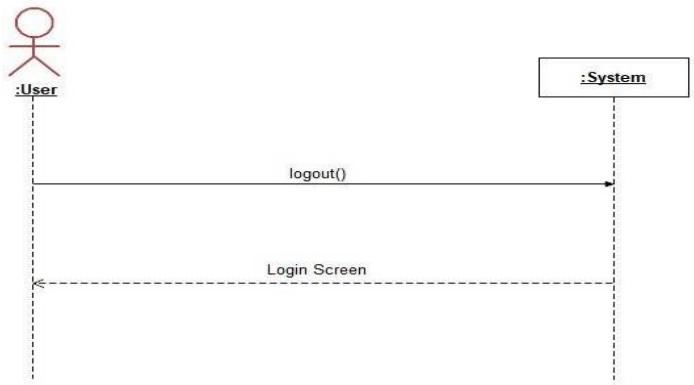


Figure 15 SSD for Logout

2.10.11 Manage Classes

Figure 16 shows the system sequence diagram of Manage Classes.

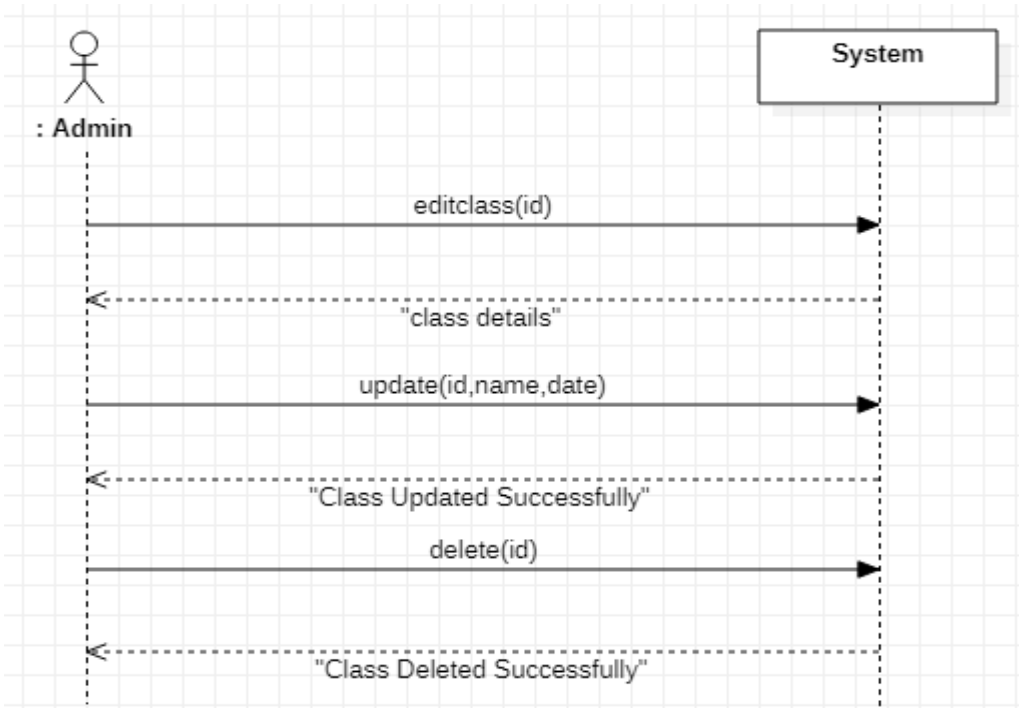


Figure 16 SSD for Manage Classes

2.11 Domain Model

Domain model is the visual representation of the decomposition of a domain into individual conceptual classes or objects. It is a way to describe and model real world entities and the relationships between them.

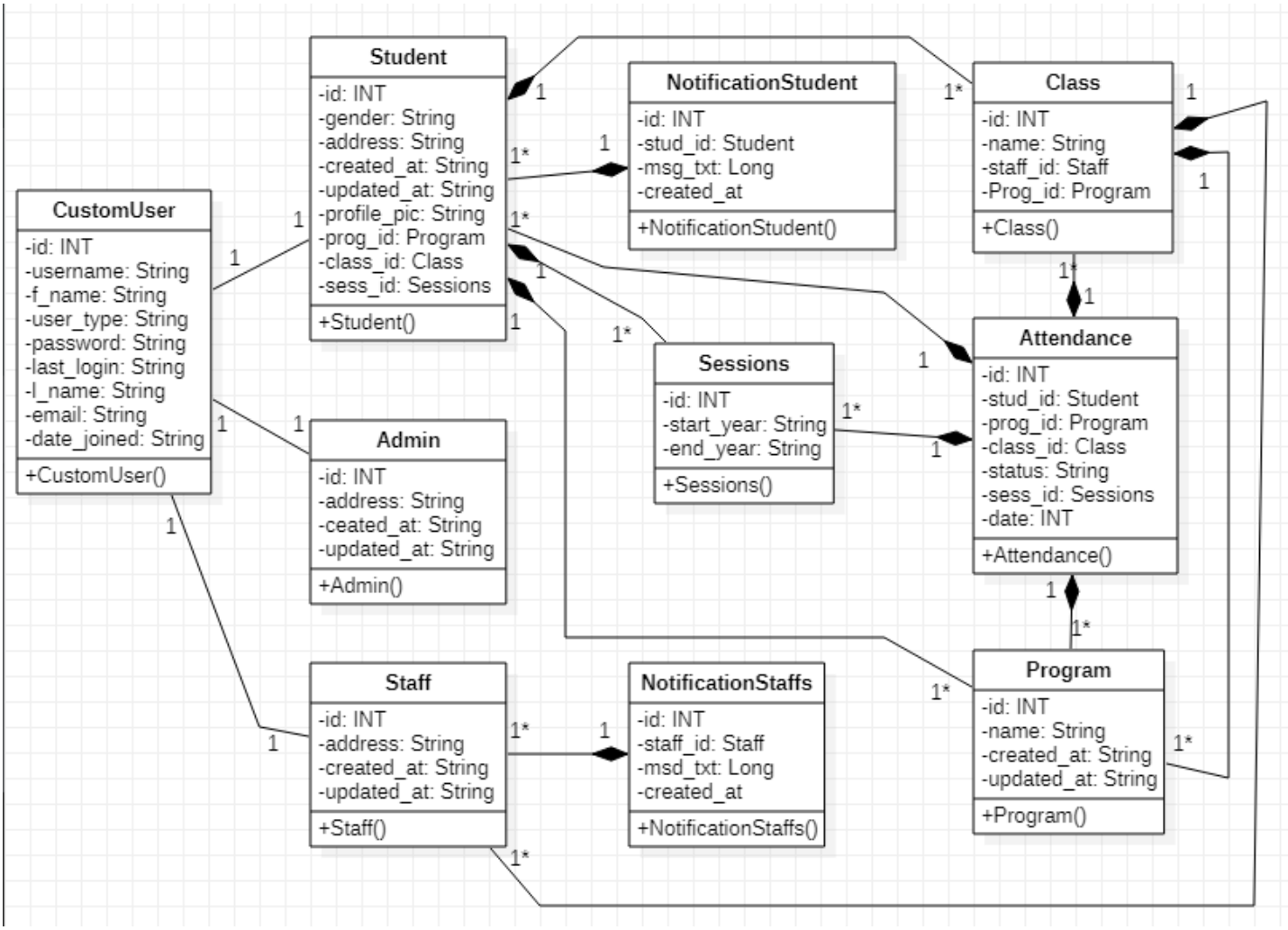


Figure 19 Domain Model of Student Management System

2.12 Database Design

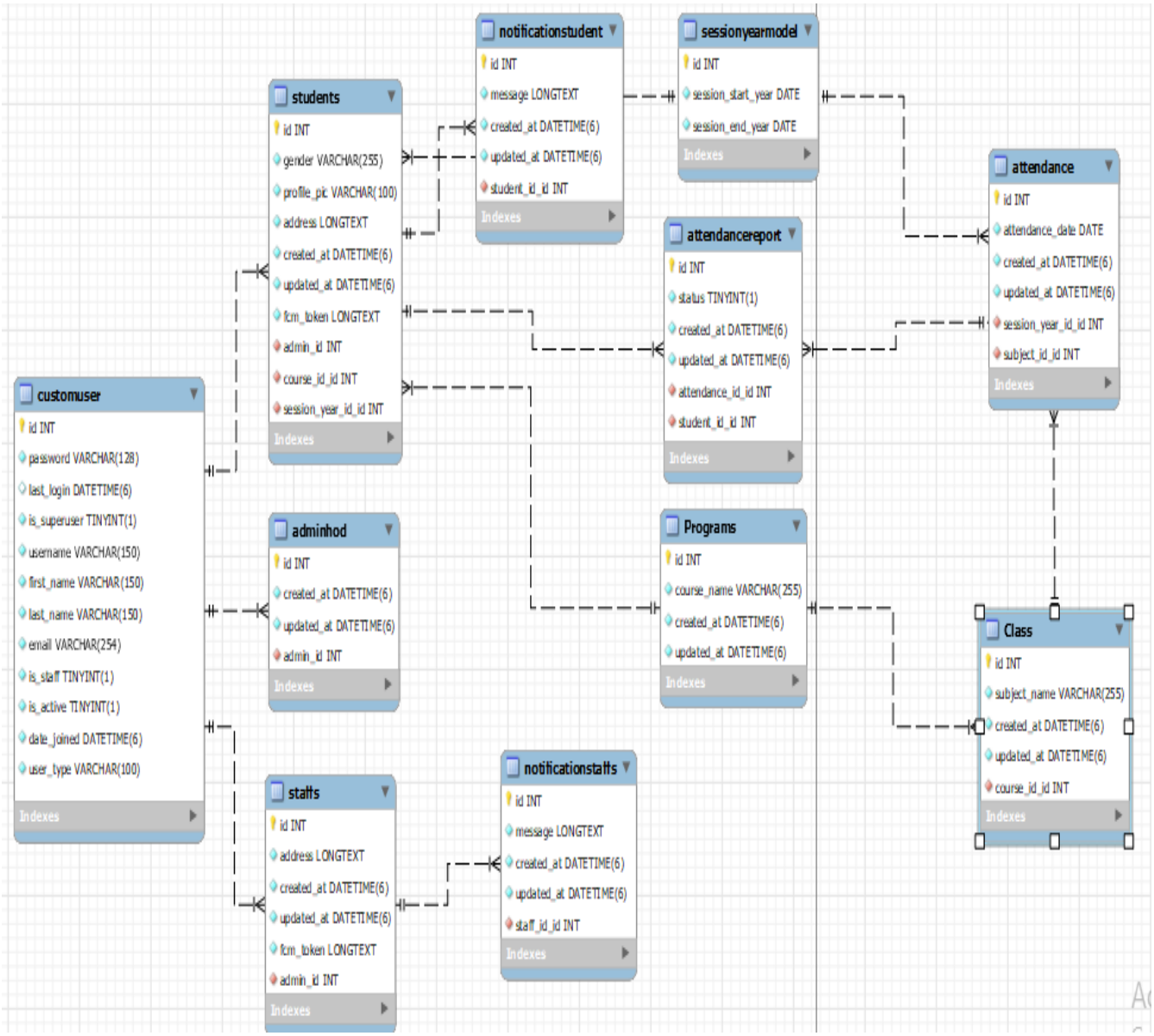


Figure 20 Database Design for Student Management System

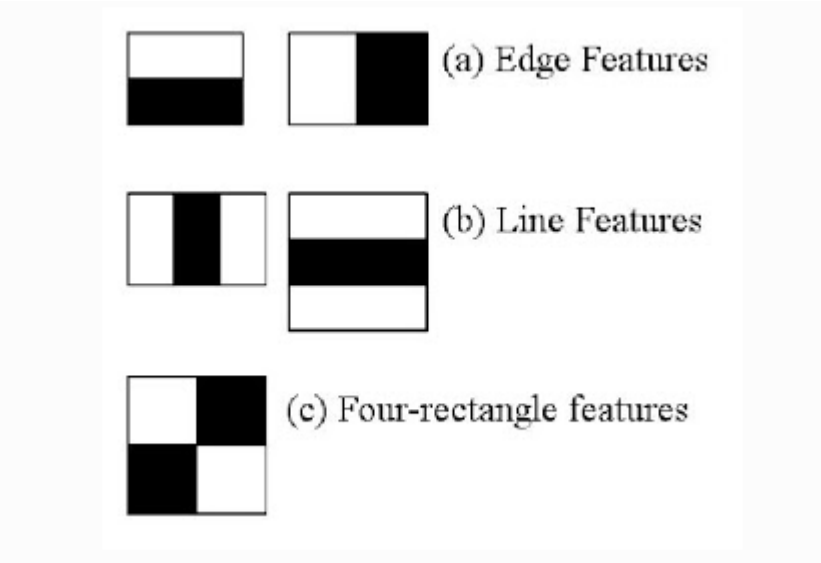
2.13 System functionalities

For Face Detection, and Recognition we used OpenCv Library along with Haar Feature-based Cascade Classifiers.

2.13.1 Basics of Face Recognition

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

Here we will work with face detection. Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier. Then we need to extract features from it. For this, haar features shown in below image are used. They are just like our convolutional kernel. Each feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle.



2.13.2 Collecting Data Sets

Using OpenCV inbuilt function we start capturing frames from webcam camera and then using harr cascade classifiers we detect face from camera and resize according to that and then we convert these frames to gray scale and save to the directory with username. here below are the samples collected using this technique.

Figure 21 Data Set

2.13.3 Training Model

After creating data sets next step is to train your model. We used OpenCV LBPHFaceRecognizer function. It has following parameters

Radius: The radius used for building the Circular Local Binary Pattern. The greater the radius, the smoother the image but more spatial information you can get.

Neighbors: The number of sample points to build a Circular Local Binary Pattern from. An appropriate value is to use 8 sample points. Keep in mind: the more sample points you include, the higher the computational cost.

Grid_x: The number of cells in the horizontal direction, 8 is a common value used in publications. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector.

Grid_y: The number of cells in the vertical direction, 8 is a common value used in publications. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector.

Threshold: The threshold applied in the prediction. If the distance to the nearest neighbor is larger than the threshold, this method returns -1.

The Circular Local Binary Patterns (used in training and prediction) expect the data given as grayscale images, use cvtColor to convert between the color spaces. And after training the model trained file is stored in directory for further process.

2.13.4 Face Recognition

When staff selects take attendance than by using OpenCV videocapture() function it starts capturing frames and with that it reads that trained file from directory using imread() function. The sample of recognition is given below.

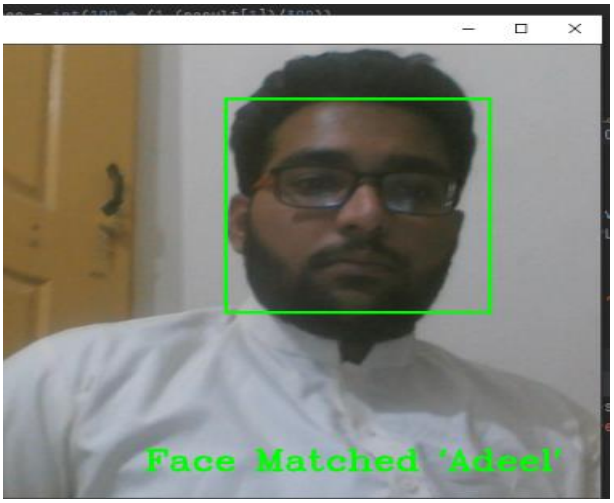


Figure 22 Face Detection

2.14 Overview

This chapter described the functional requirements i.e. use cases and their flow i.e. system sequence diagrams. Also it describes the skeleton of our database that how it would look like when implemented and what would be the relationship among the entities in the form of entity relationship diagram (ERD) and the domain model and the inner model for face recognition is also explained here.

Chapter 3
Software Design Description

3.1 Introduction

Software Design Description (SDD) is the representation of a software design which is used for communicating design information of a system to all the stake holders. It shows how the software system will be structured to satisfy the requirements.

3.2 User Interface Design

It is the process in which we create the prototype of the screen images. In this section, we draw the interface based on our requirement. And later during the implementation we will use them for the actual development.

3.2.1 Login In Interface

This is the proposed login interface design through which user of this application will login to it.

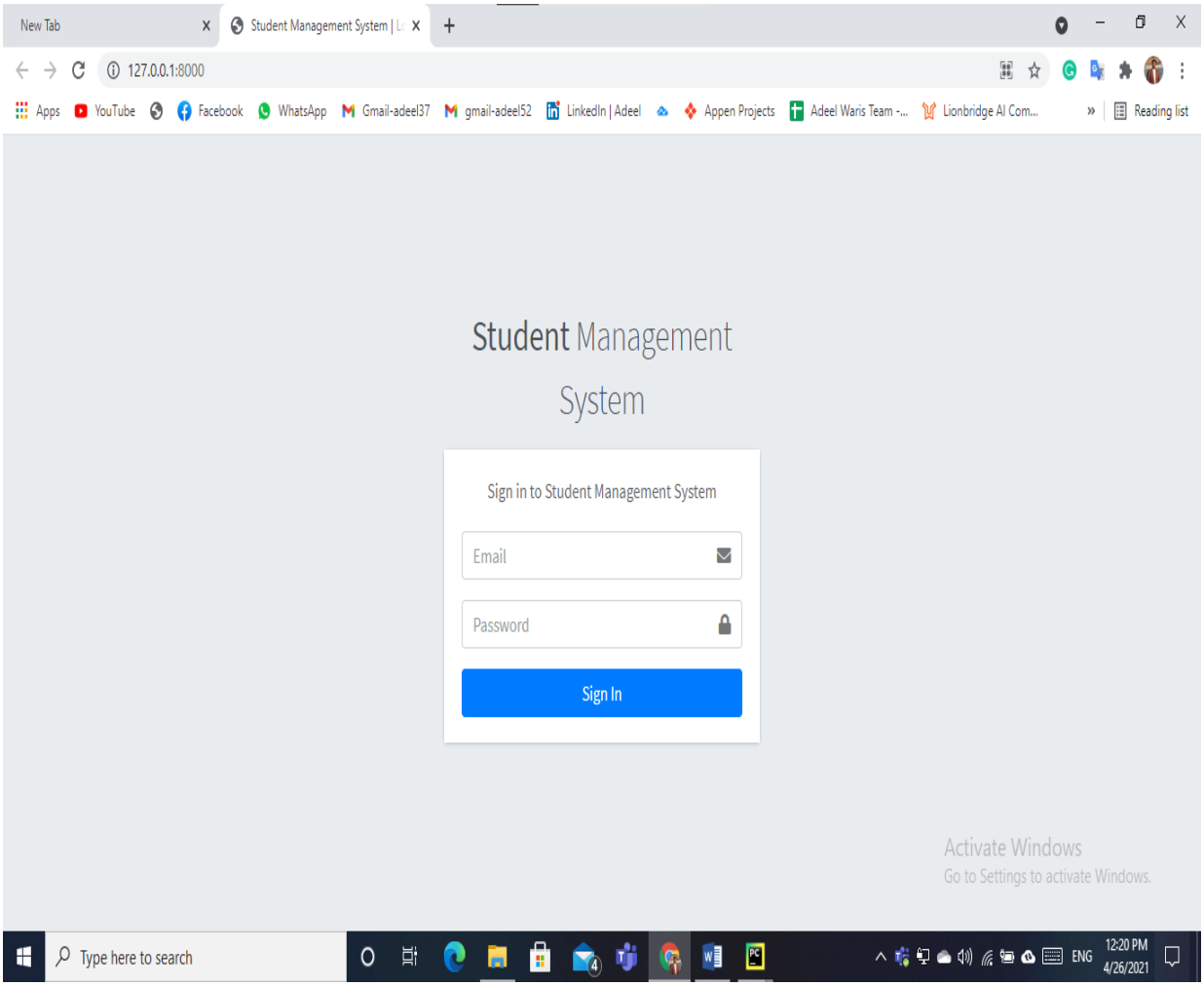


Figure 23 Interface Design Login Page

3.2.2 Admin Homepage

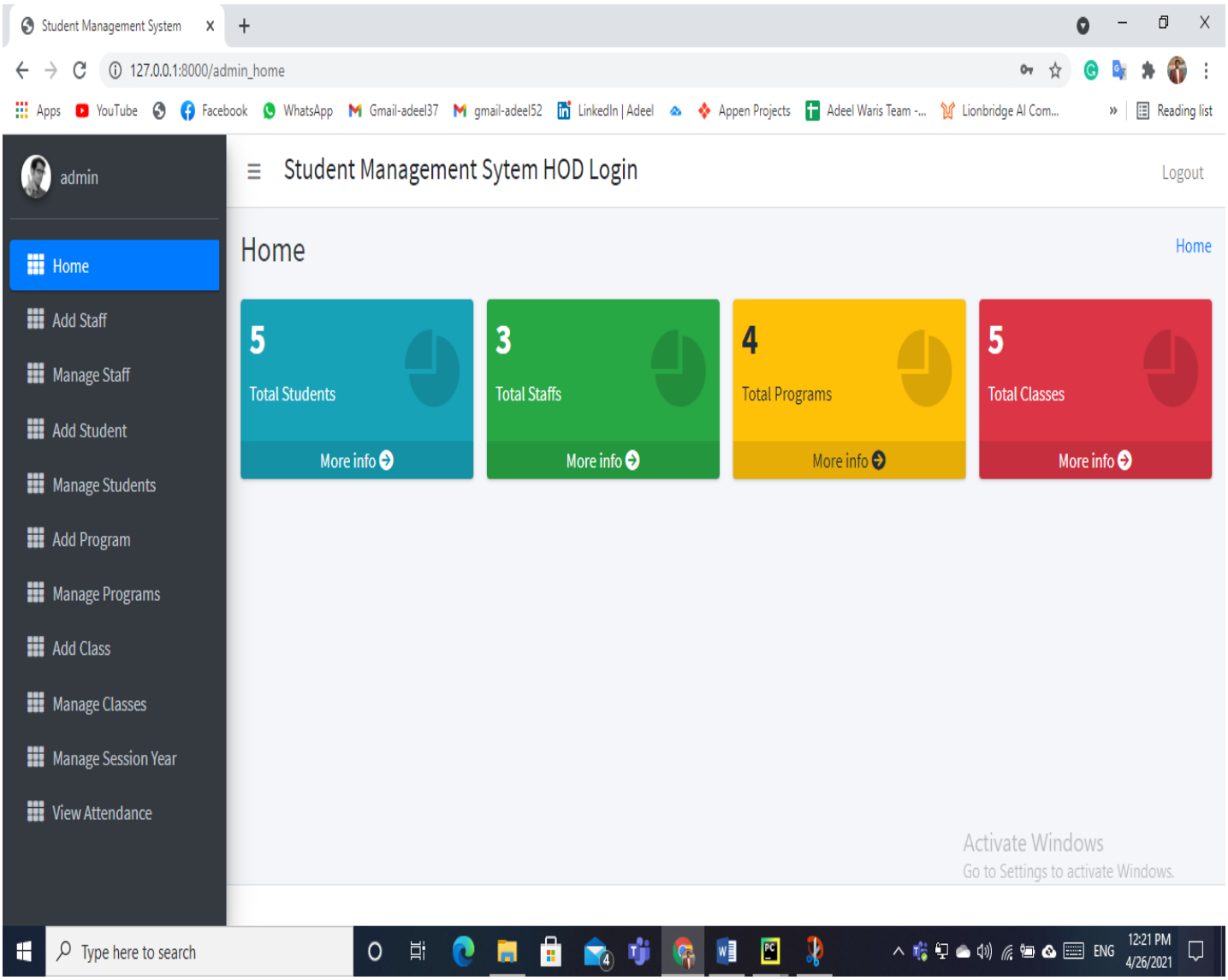


Figure 24 Interface Design Admin Home Page

3.2.3 Staff Home Page

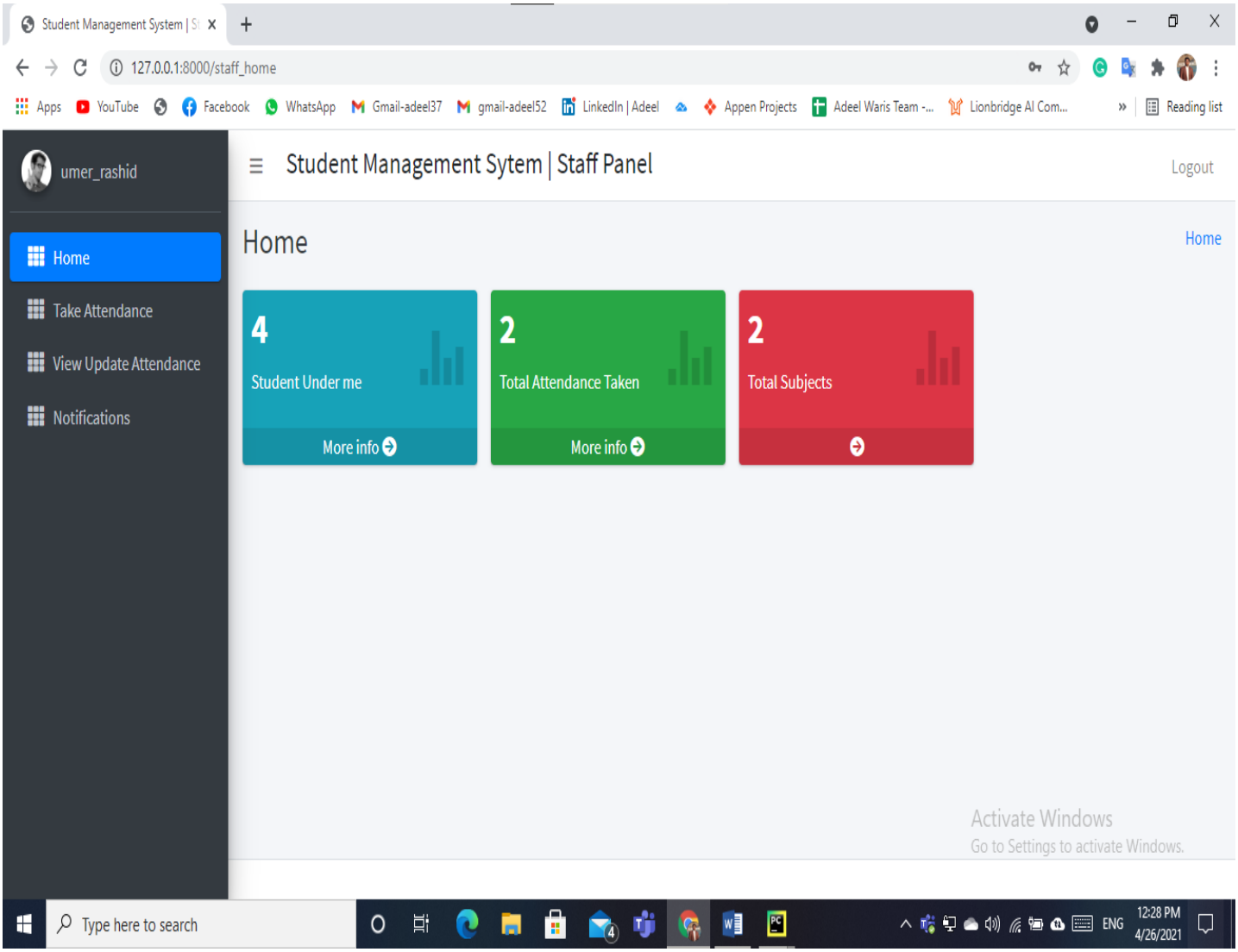


Figure 25 Interface Design Staff Home Page

3.2.4 Student Home Page

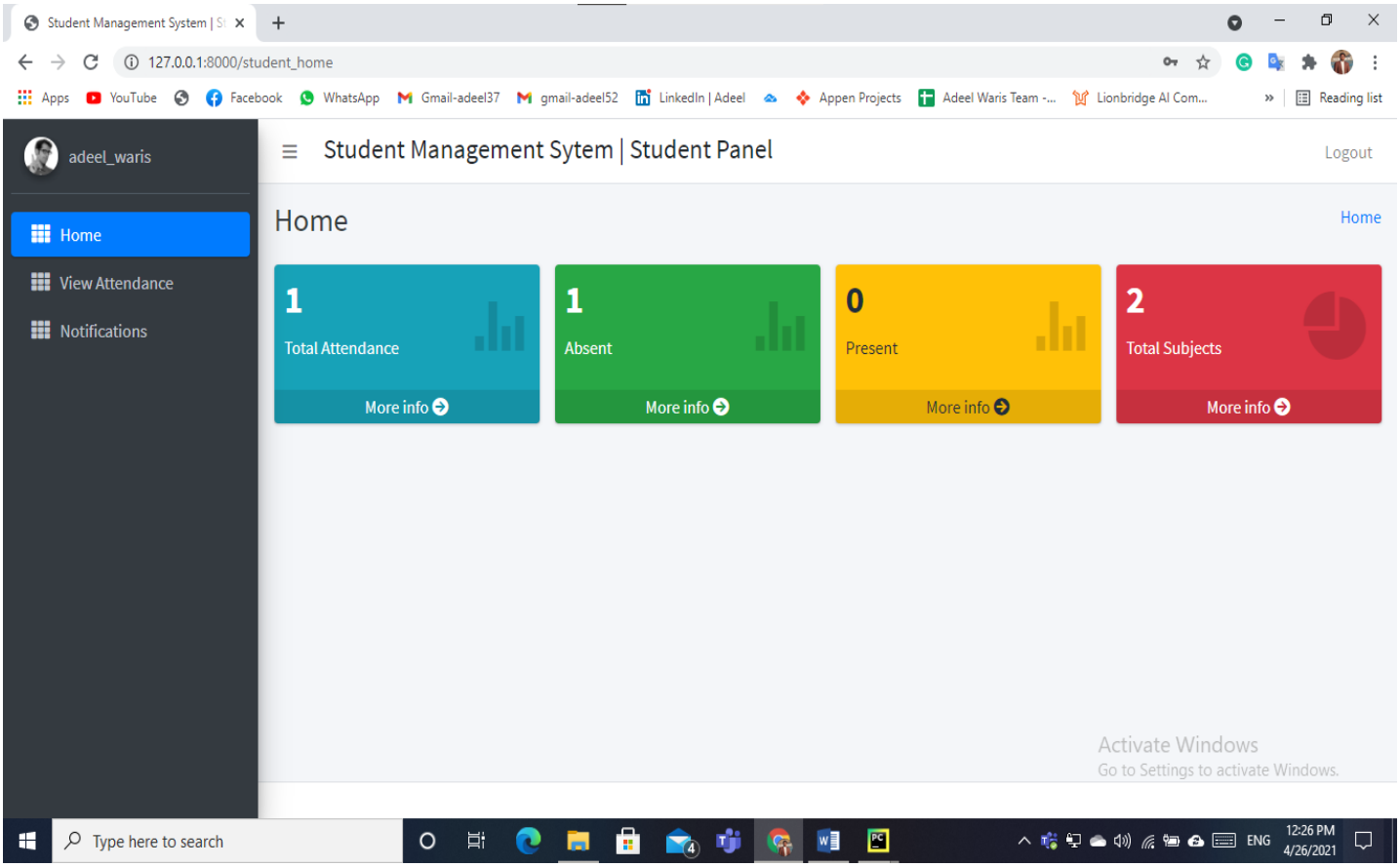
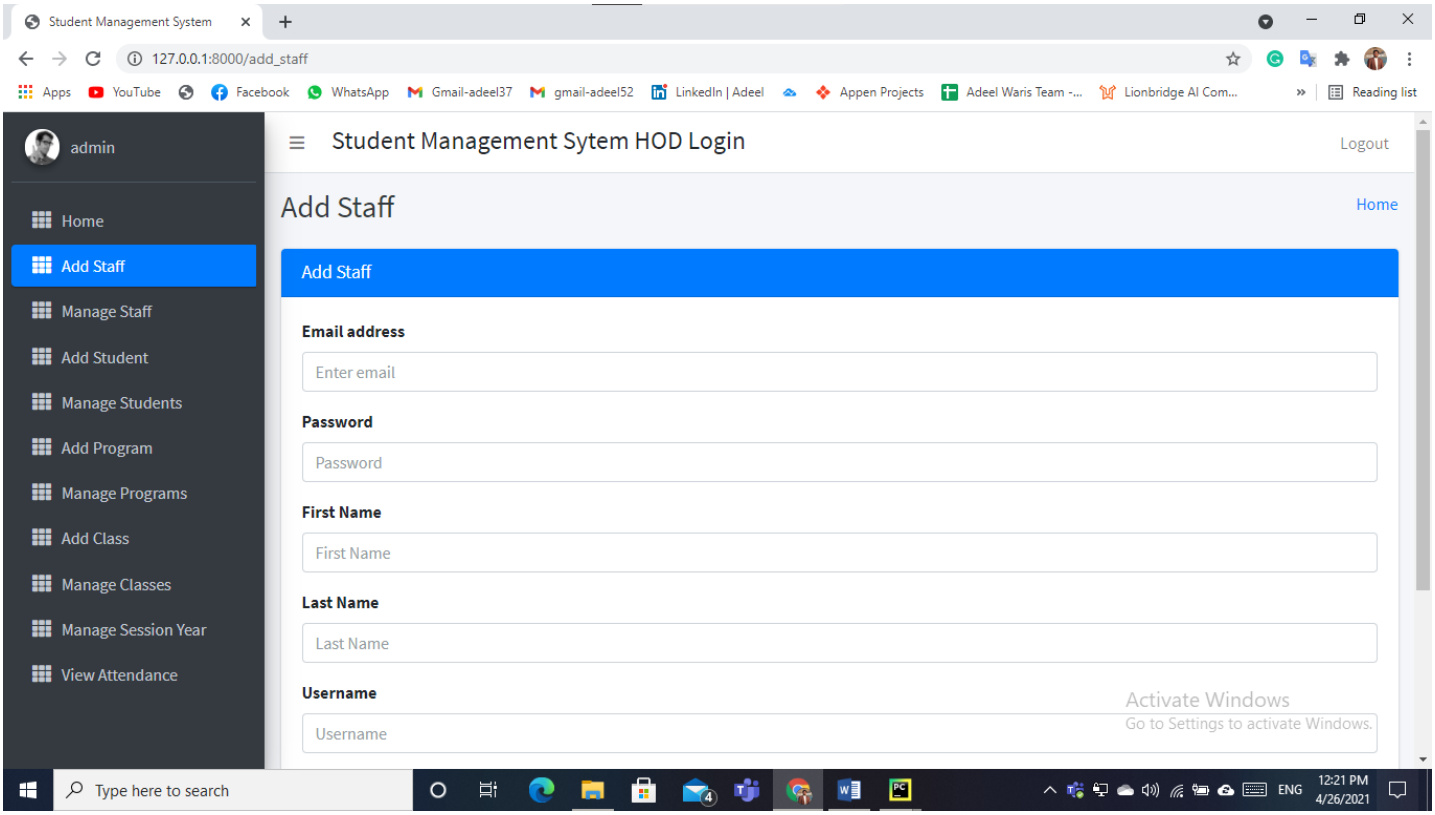
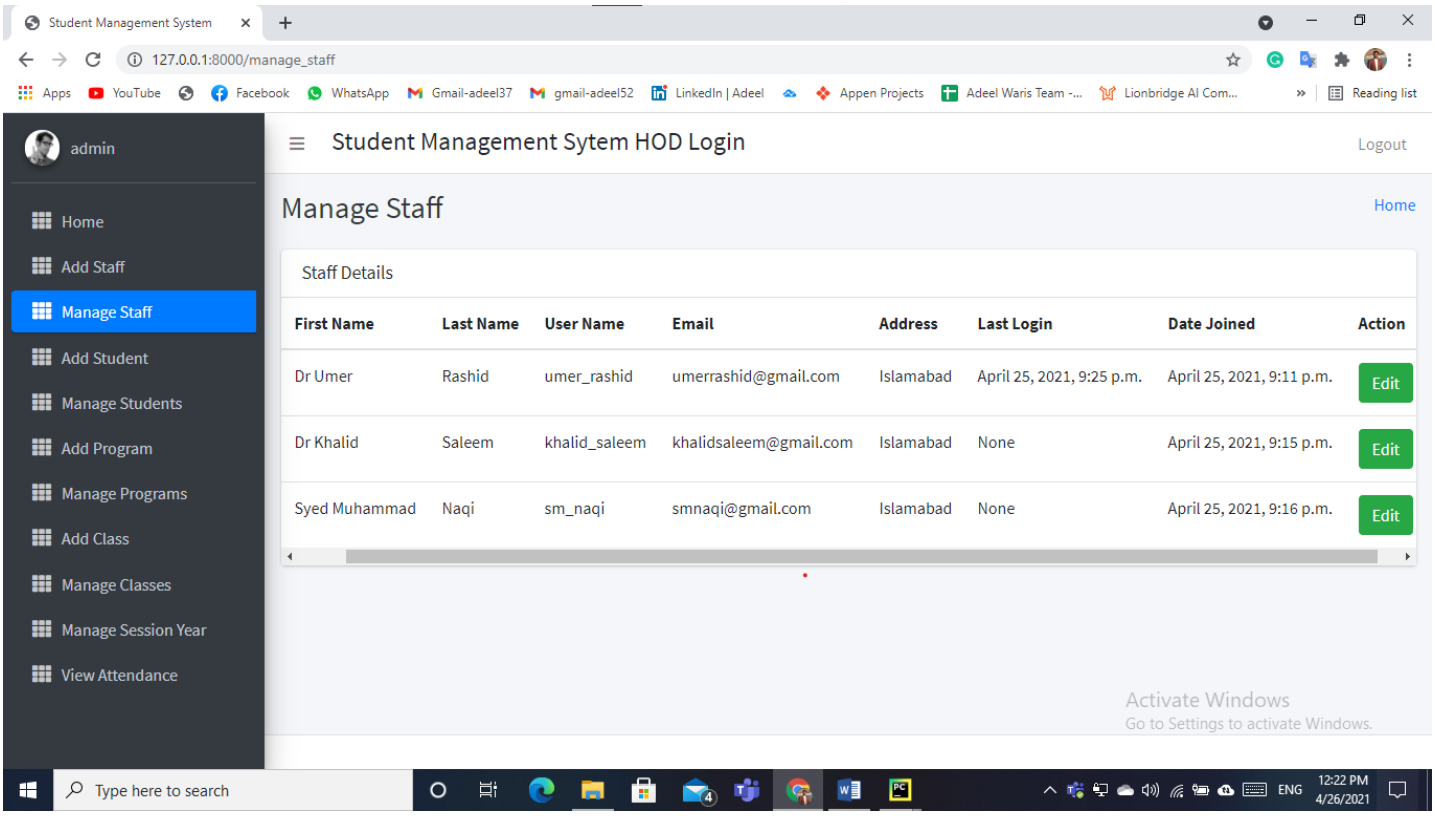


Figure 26 Interface Design Student Home Page

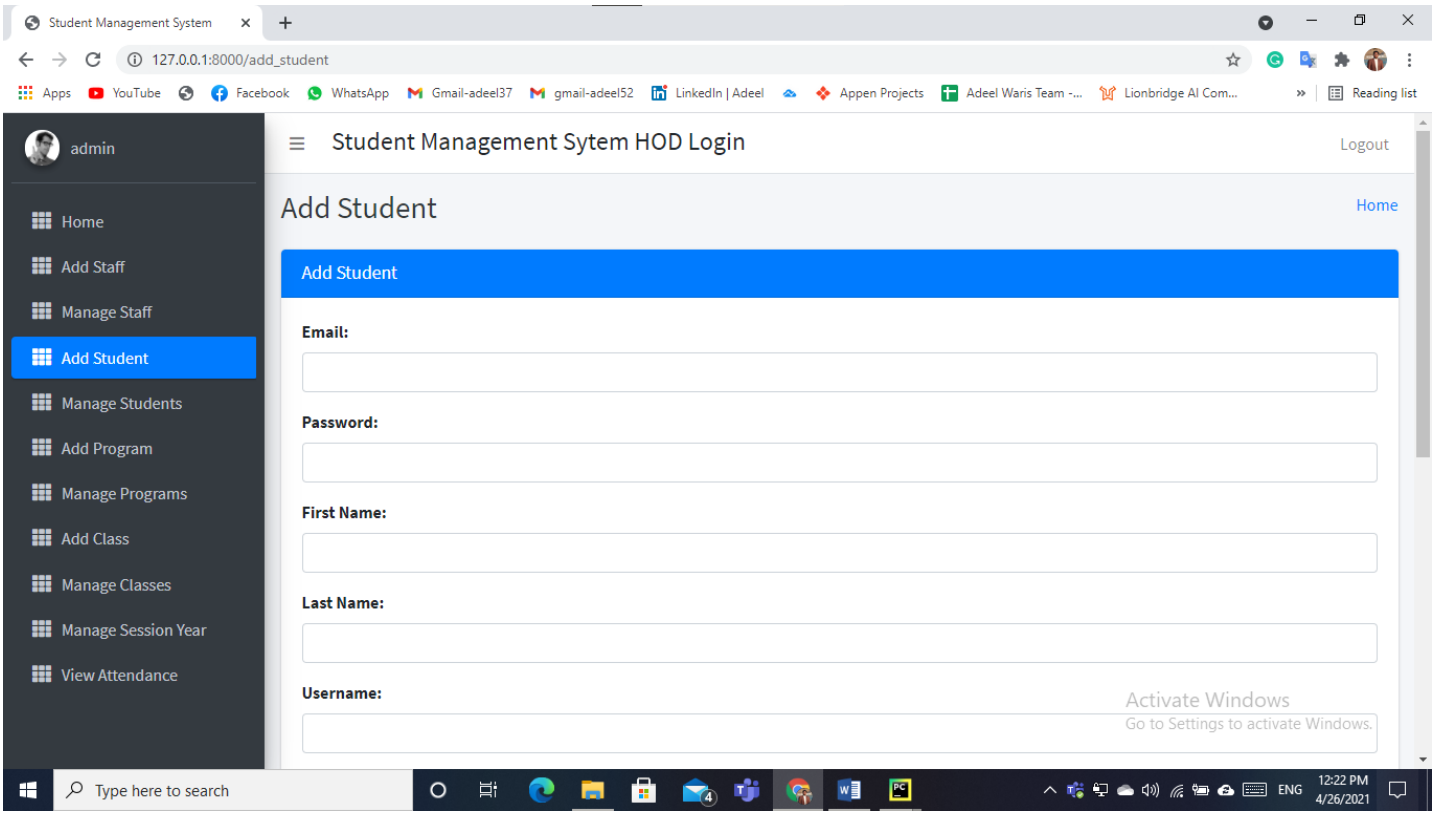
3.2.5 Add Staff



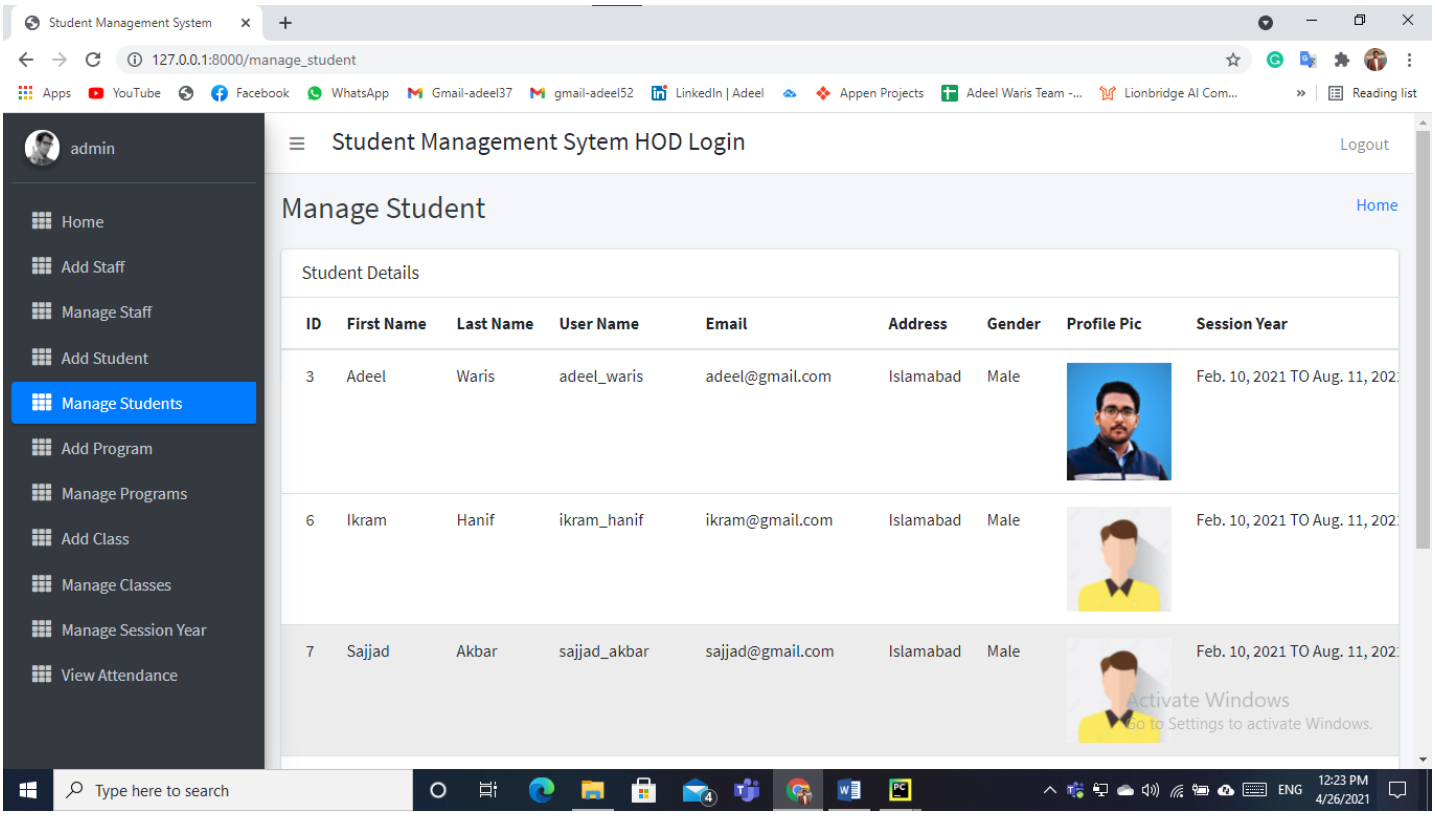
3.2.6 Manage Staff



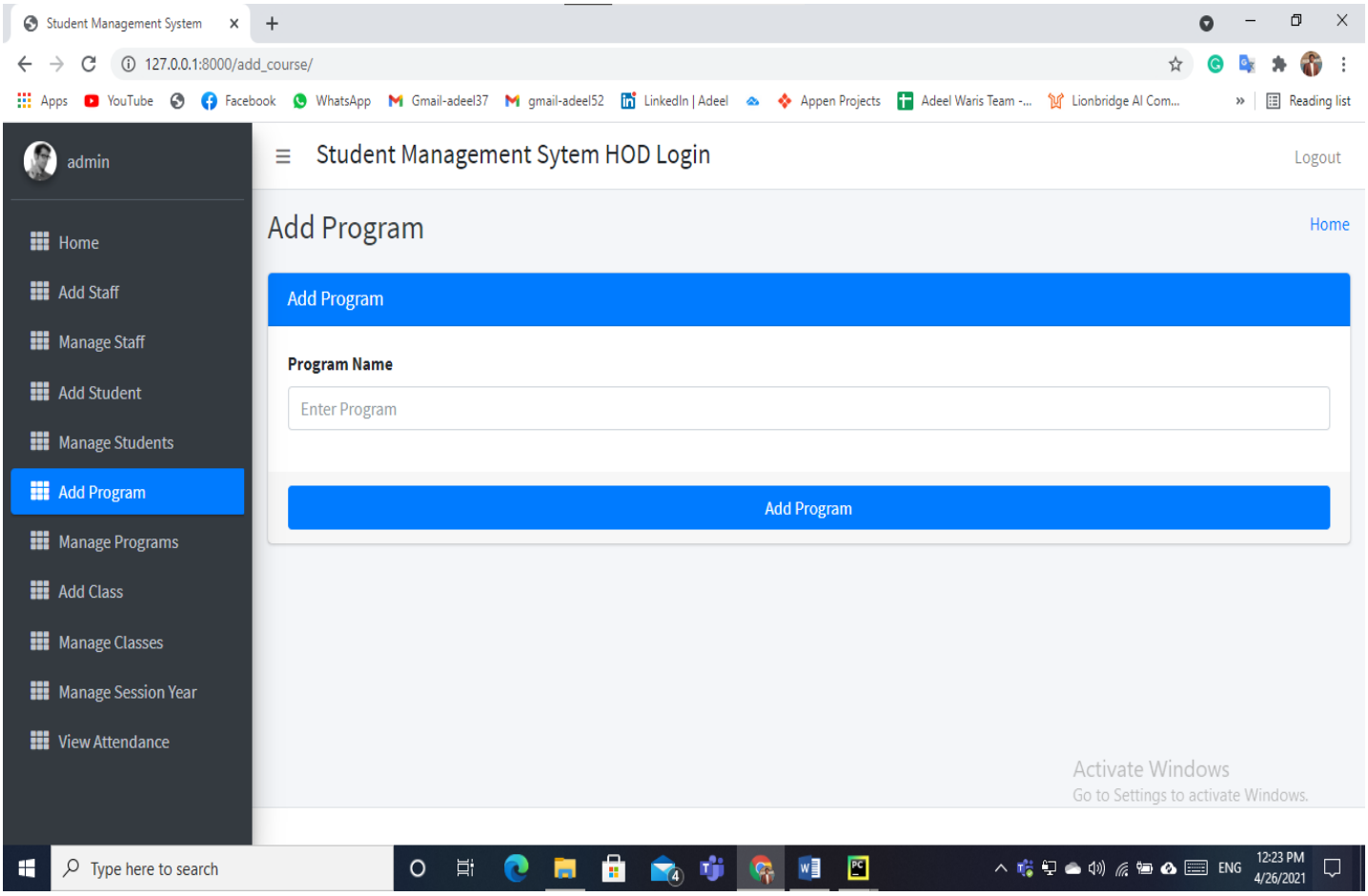
3.2.7 Add Student



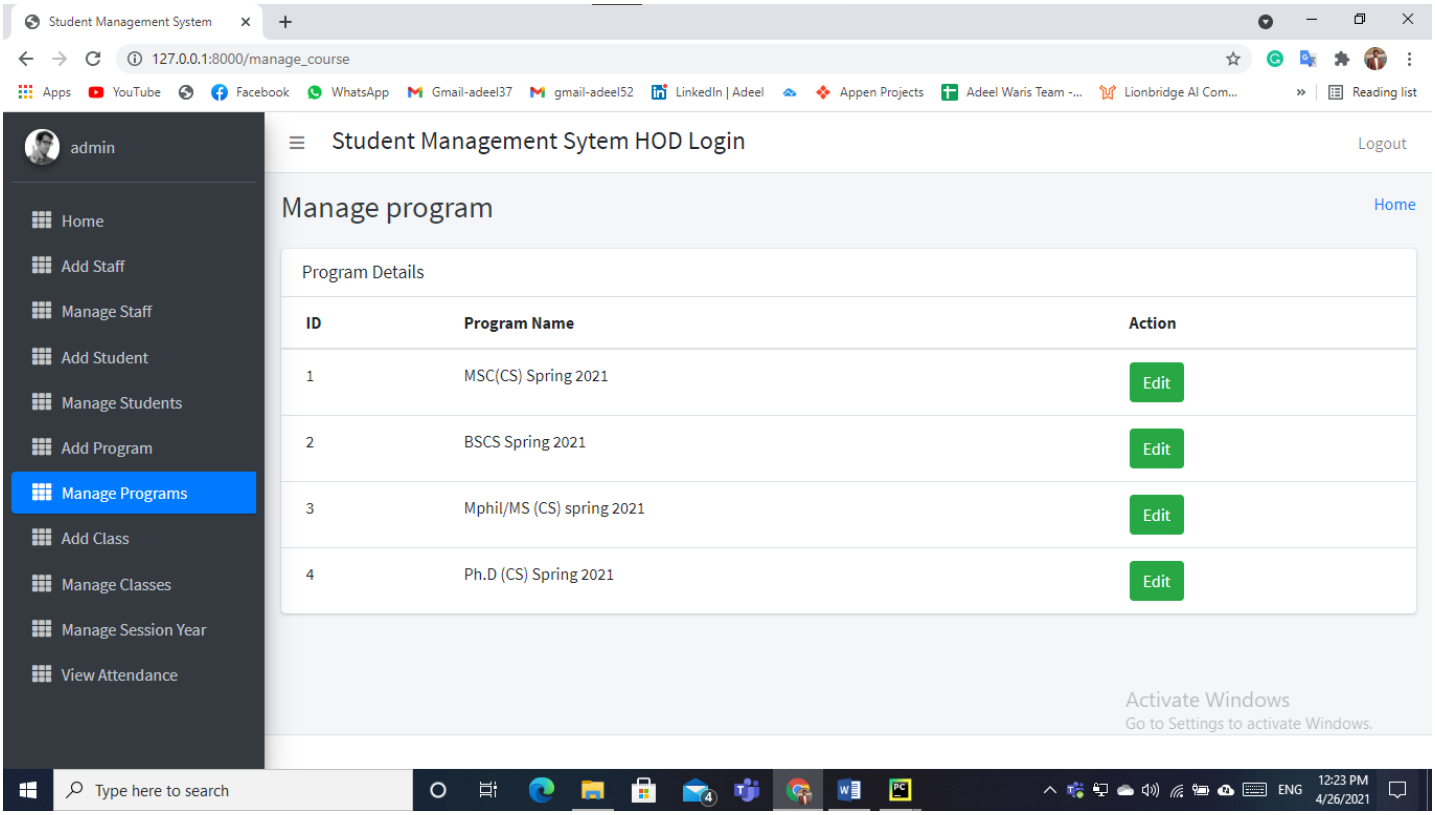
3.2.8 Manage Student



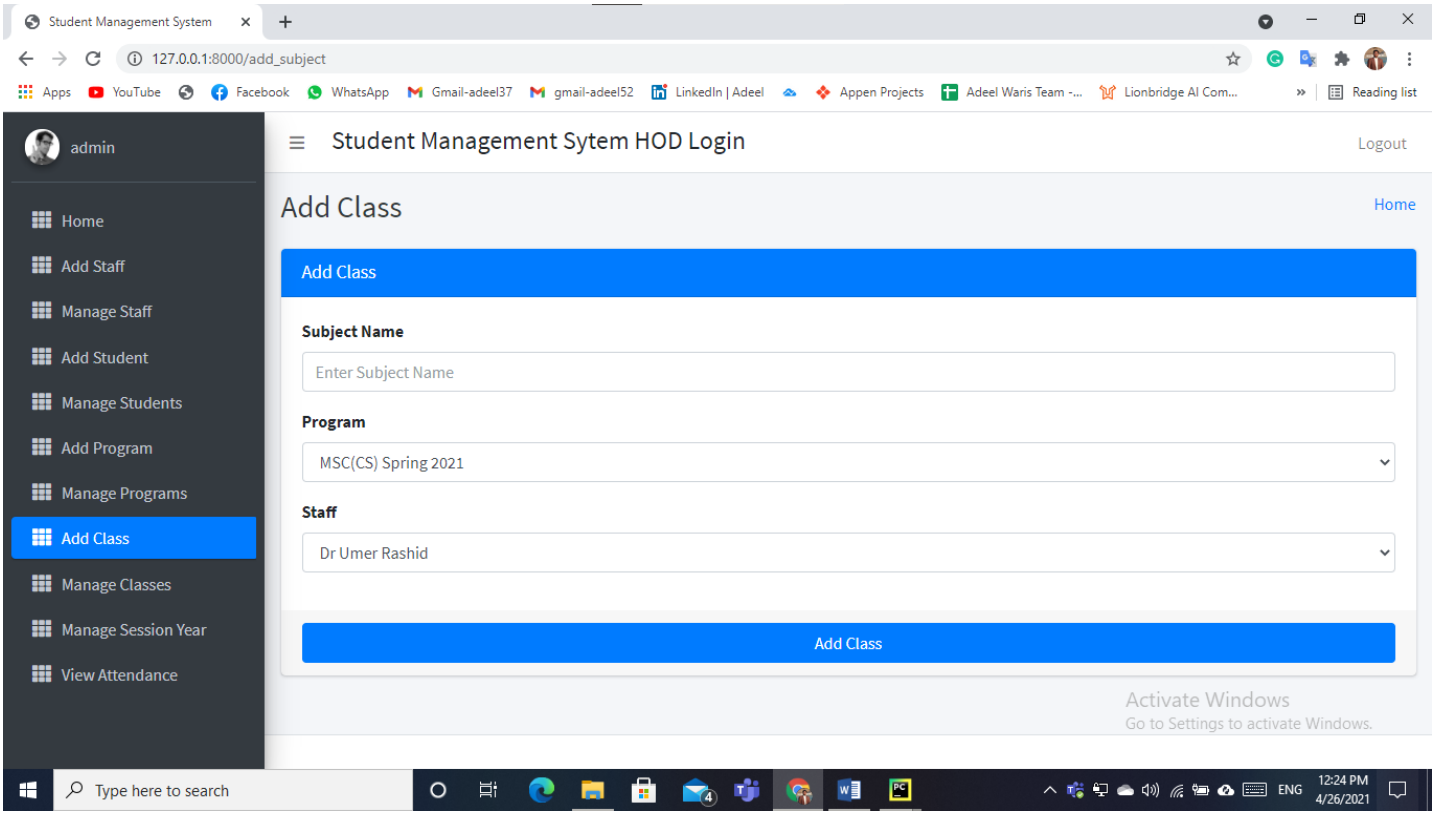
3.2.9 Add Programs



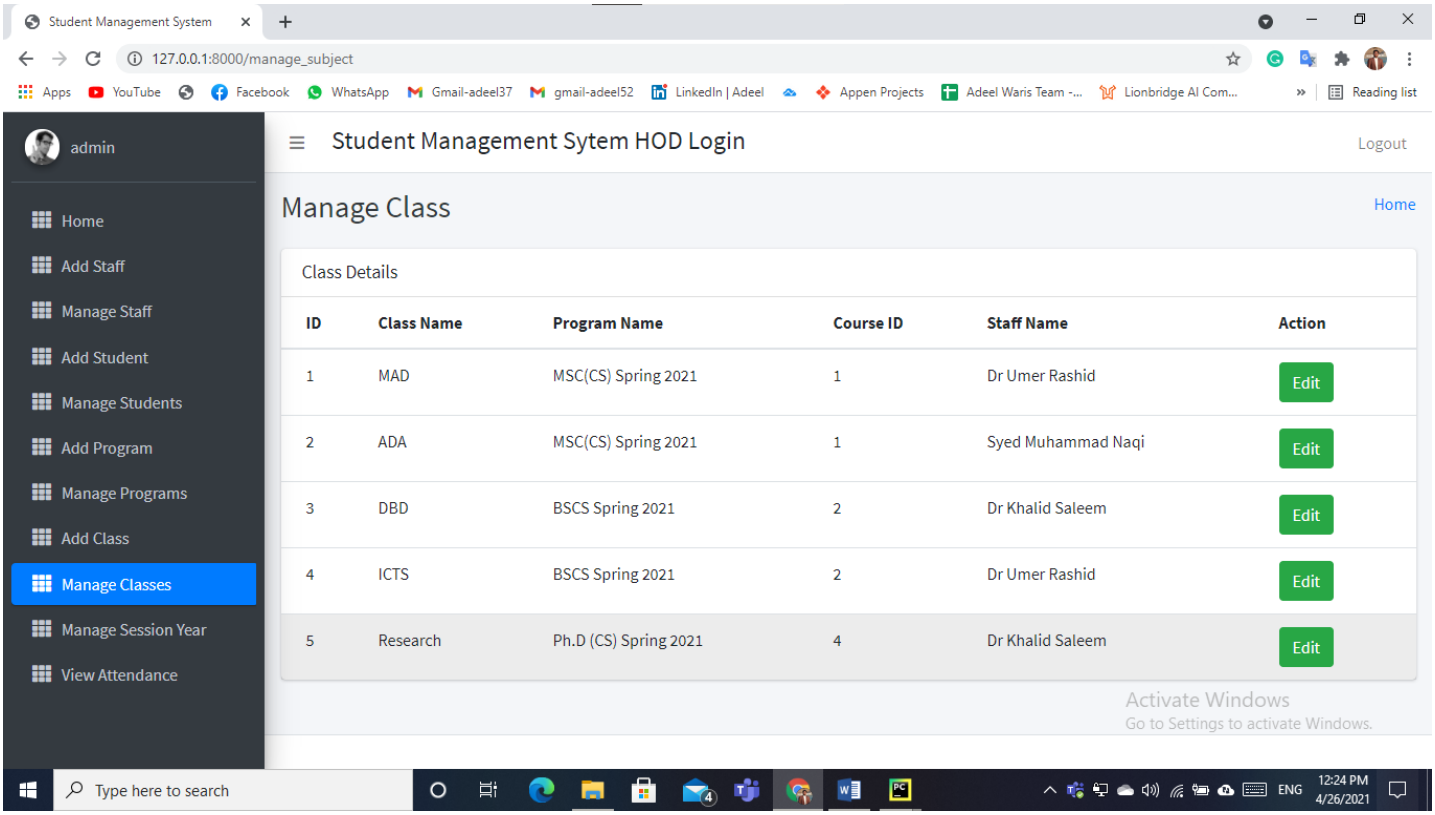
3.2.10 Manage Programs



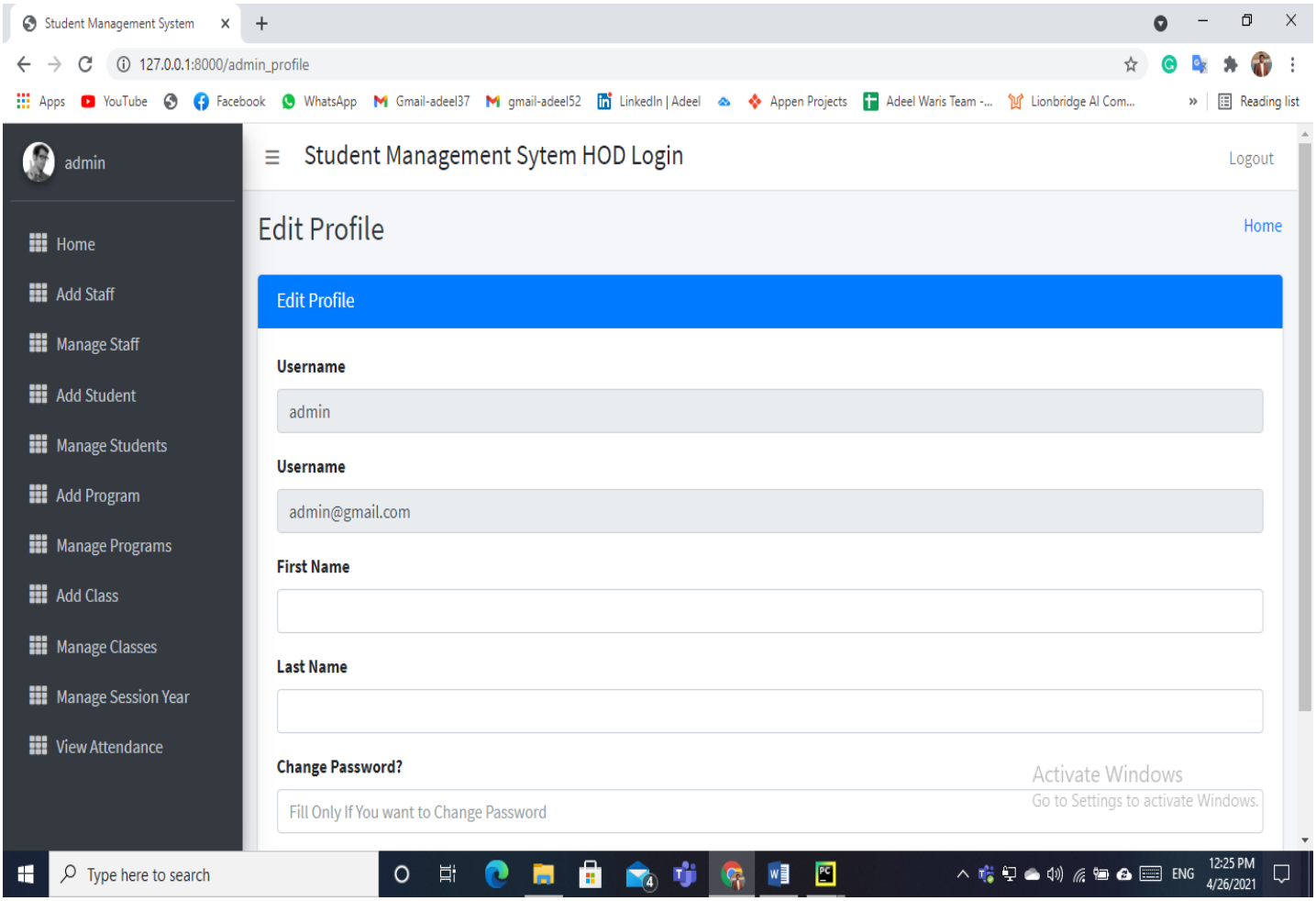
3.2.11 Add Class



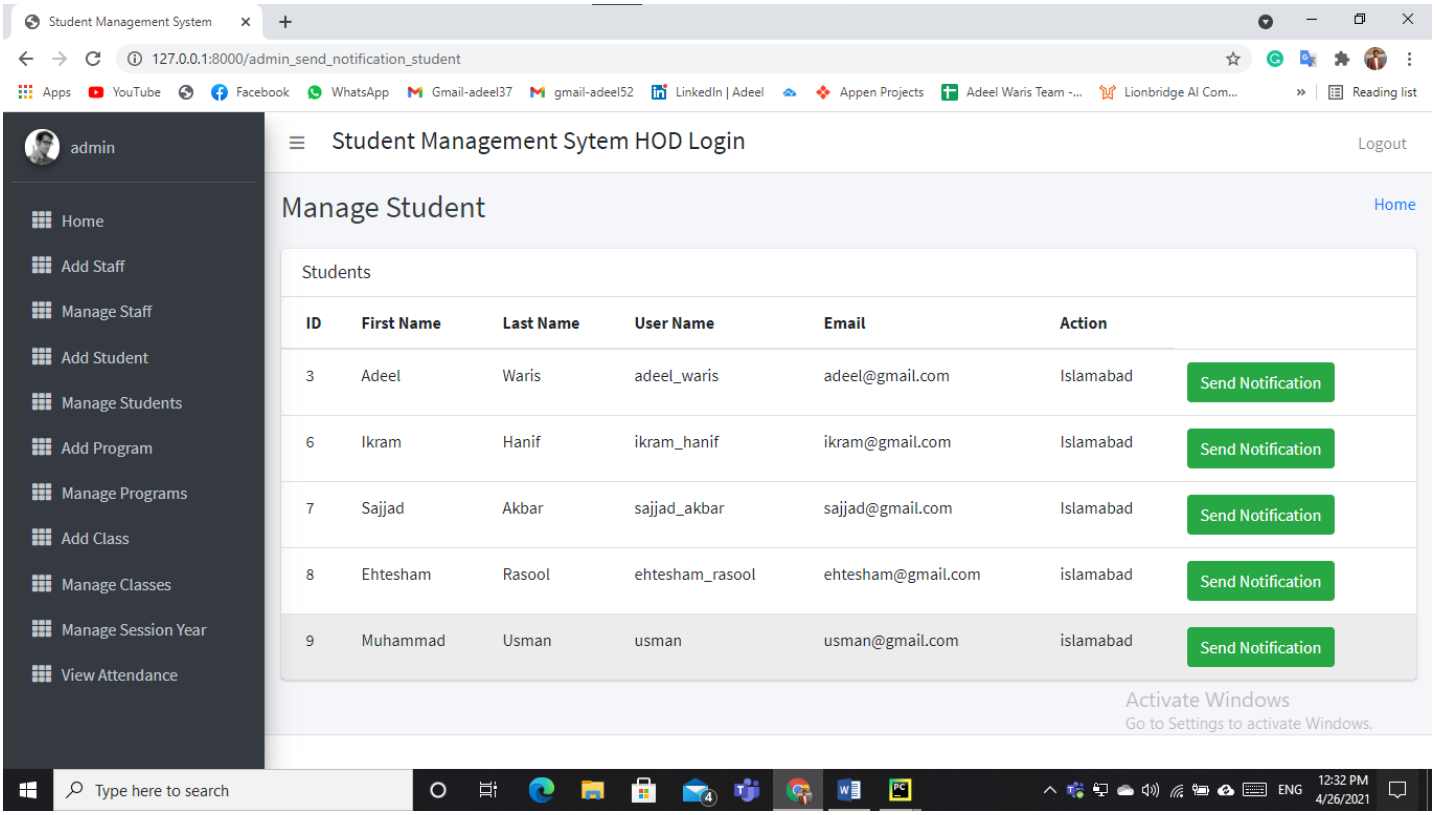
3.2.12 Manage Class



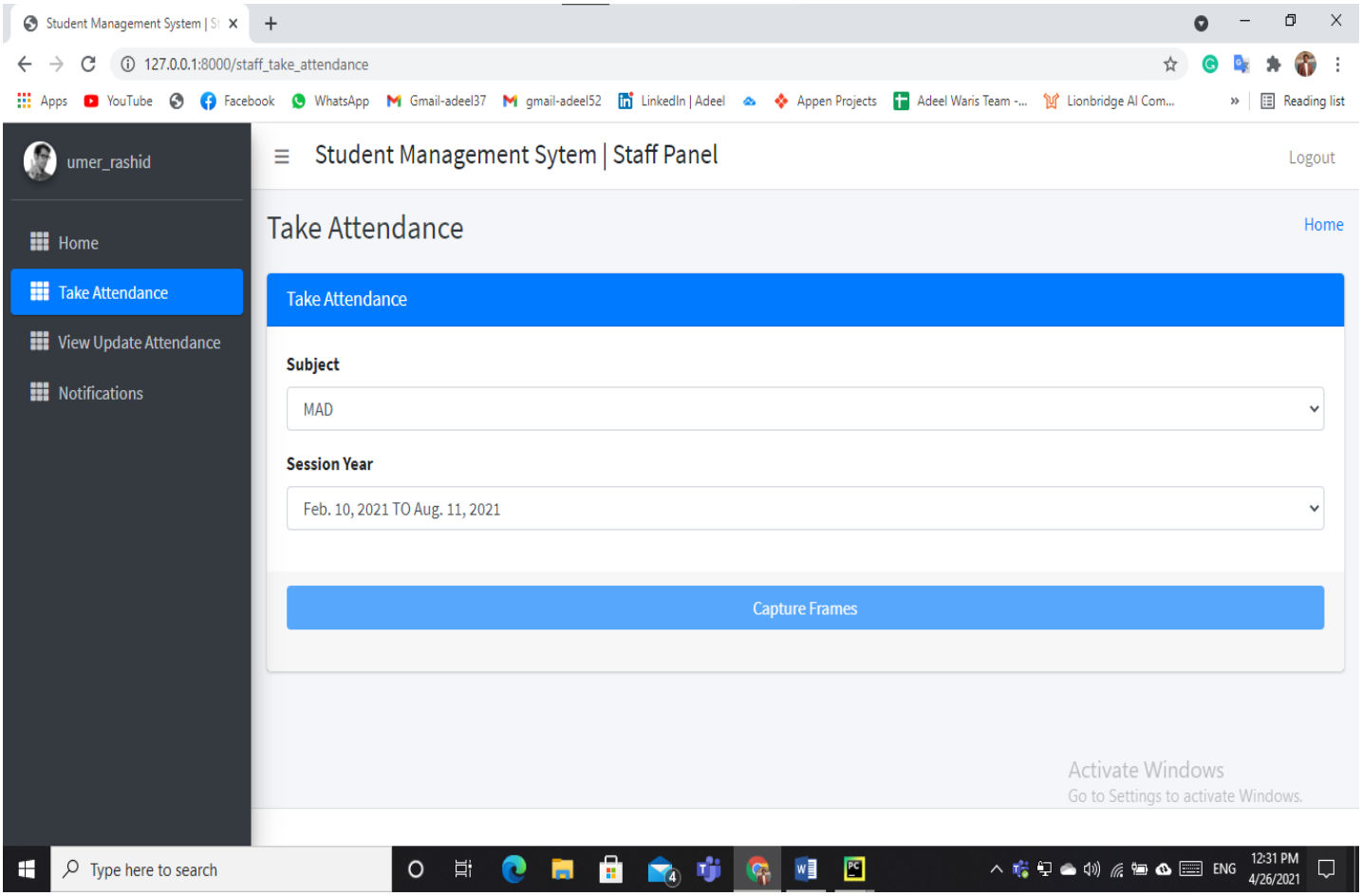
3.2.13 Edit Profile



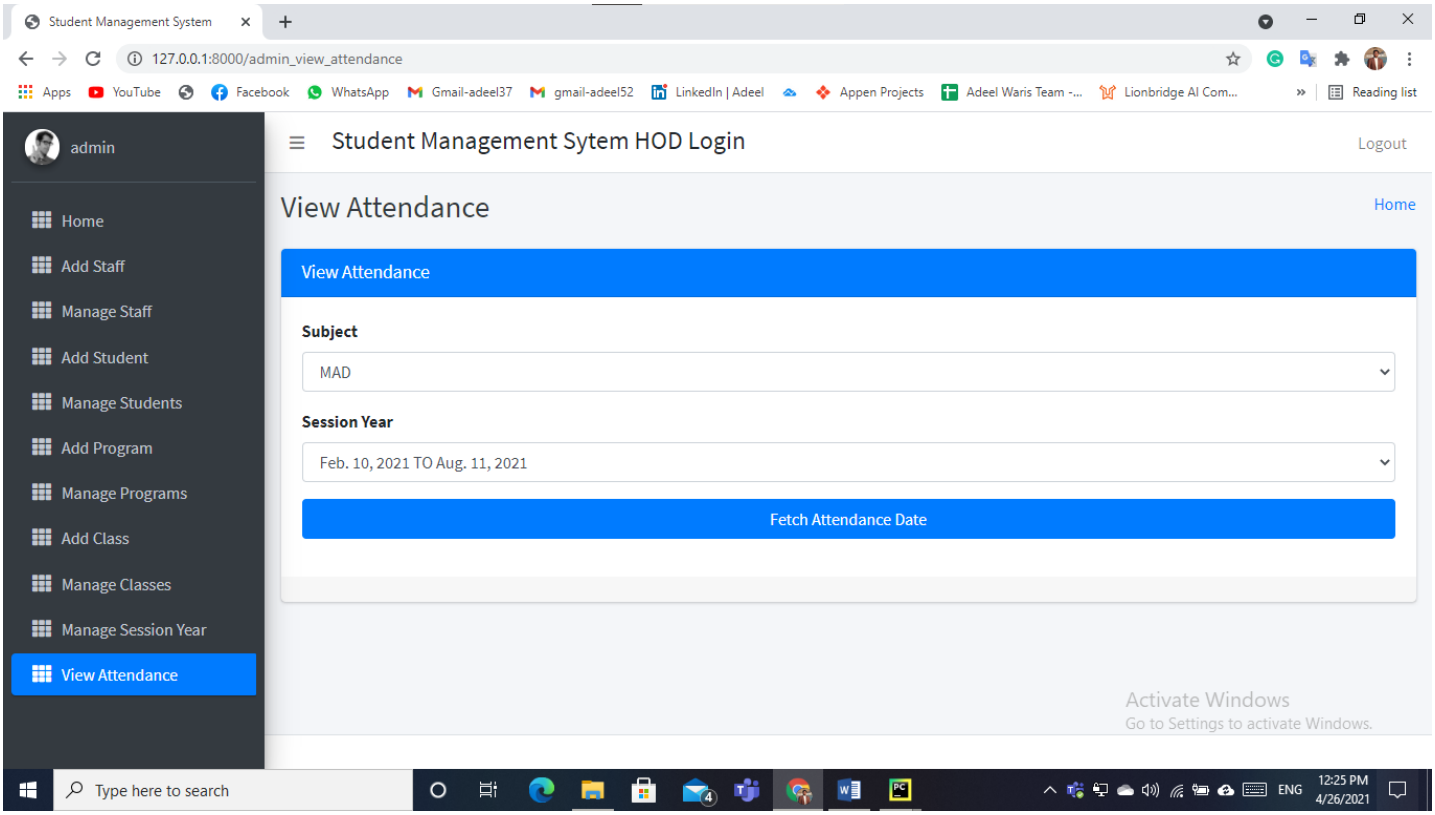
3.2.14 Send Notifications



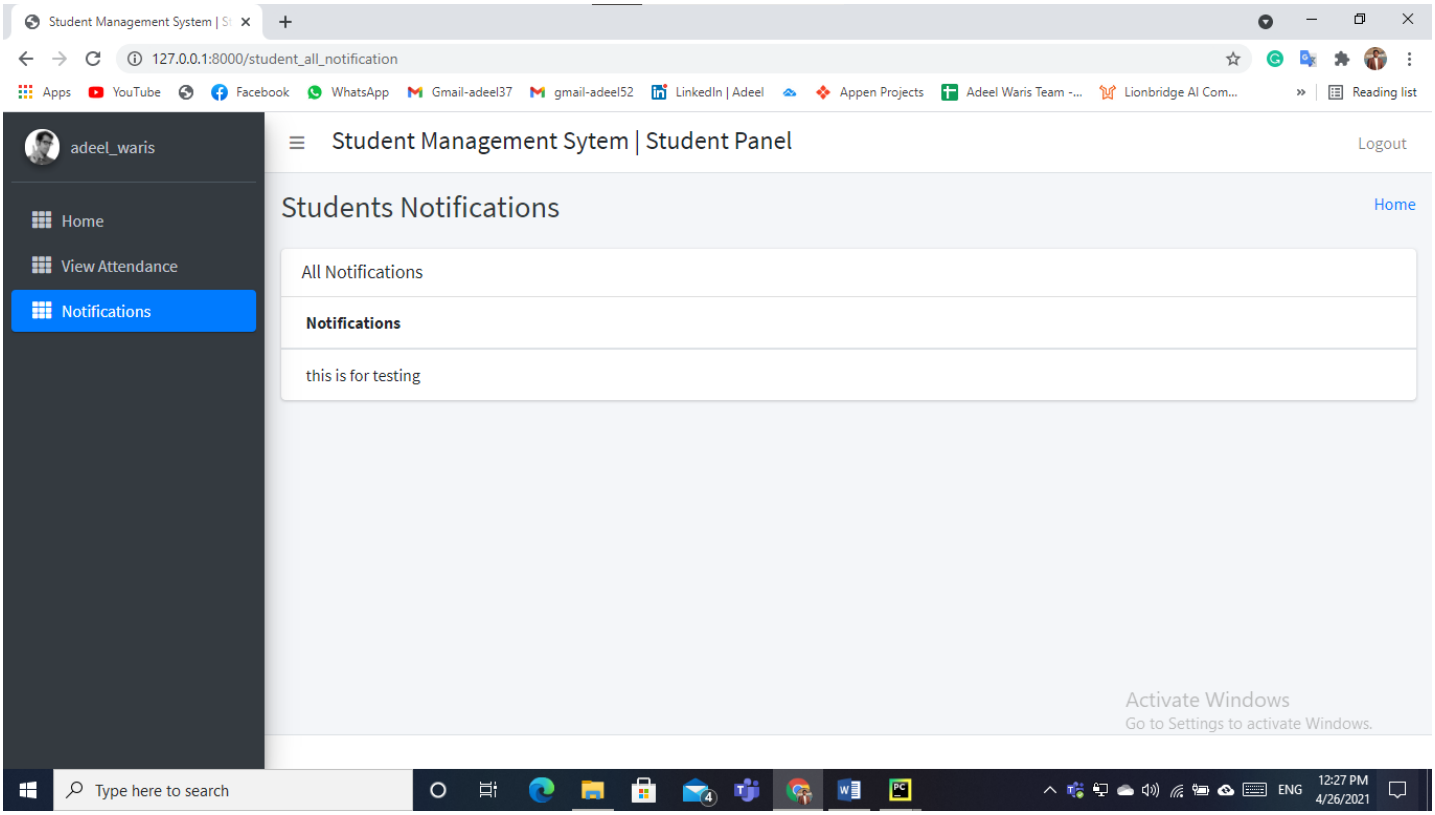
3.2.15 Mark Attendance



3.2.16 View Attendance



3.2.17 View Notifications



3.3 Sequence Diagrams

A Sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are developed against the use cases.

3.3.1 Sequence Diagram Login

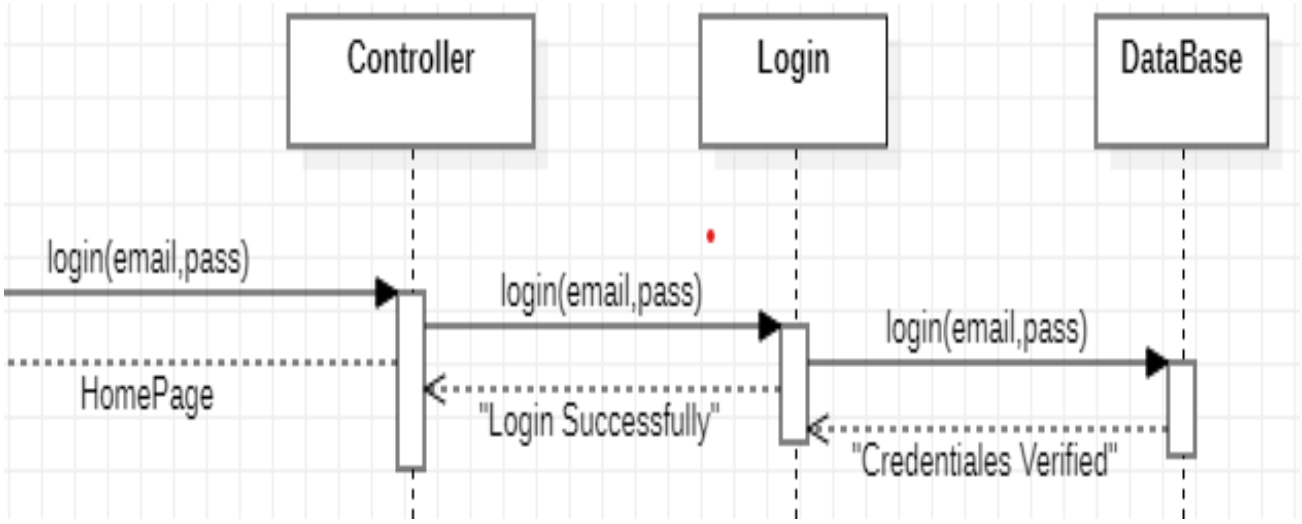


Figure 31Sequence Diagram Login

3.3.2 Sequence Diagram Add/Manage Staff

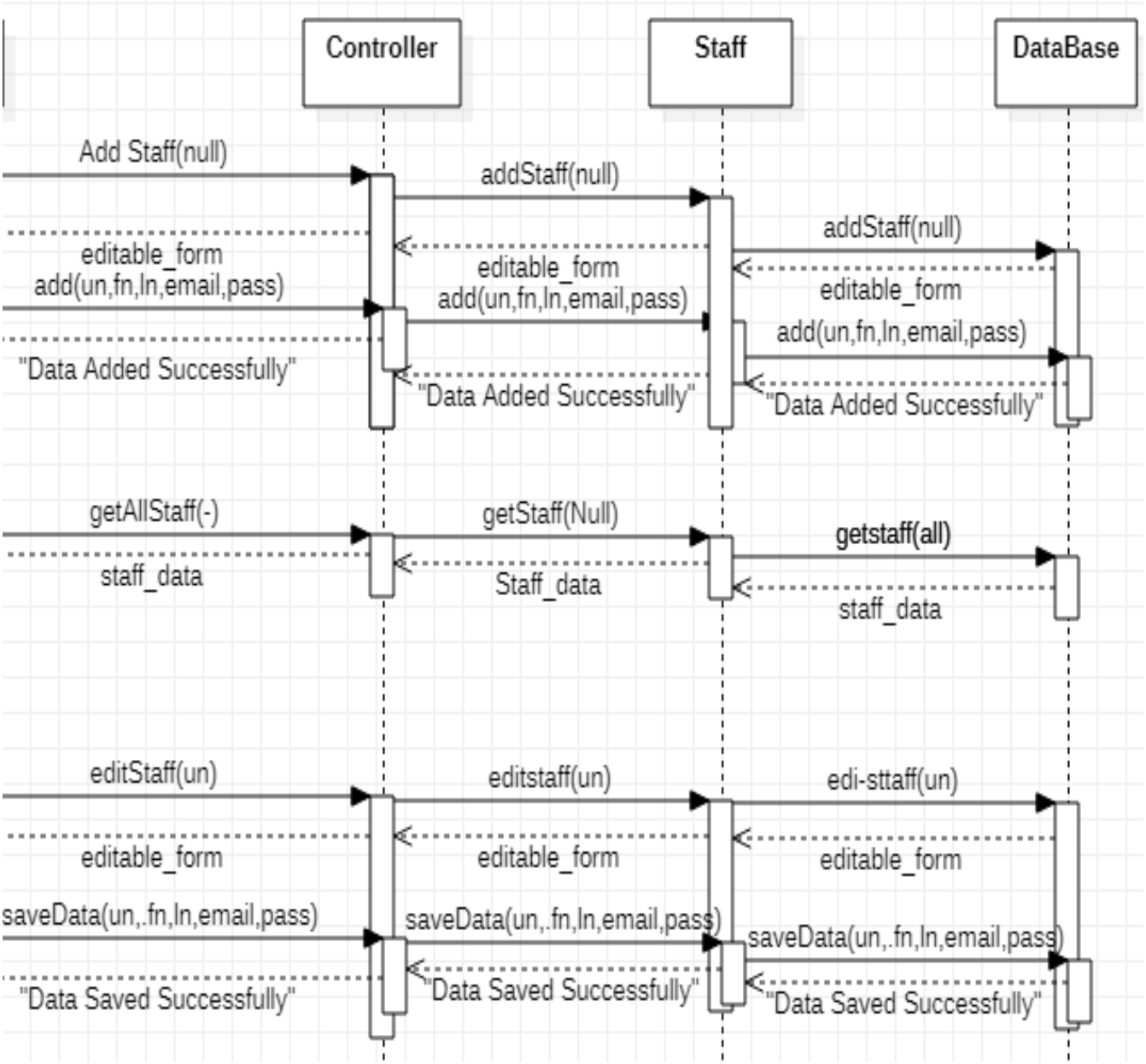


Figure 32Sequence Diagram Add/Manage Staff

3.3.3 Sequence Diagram Add/Manage Student

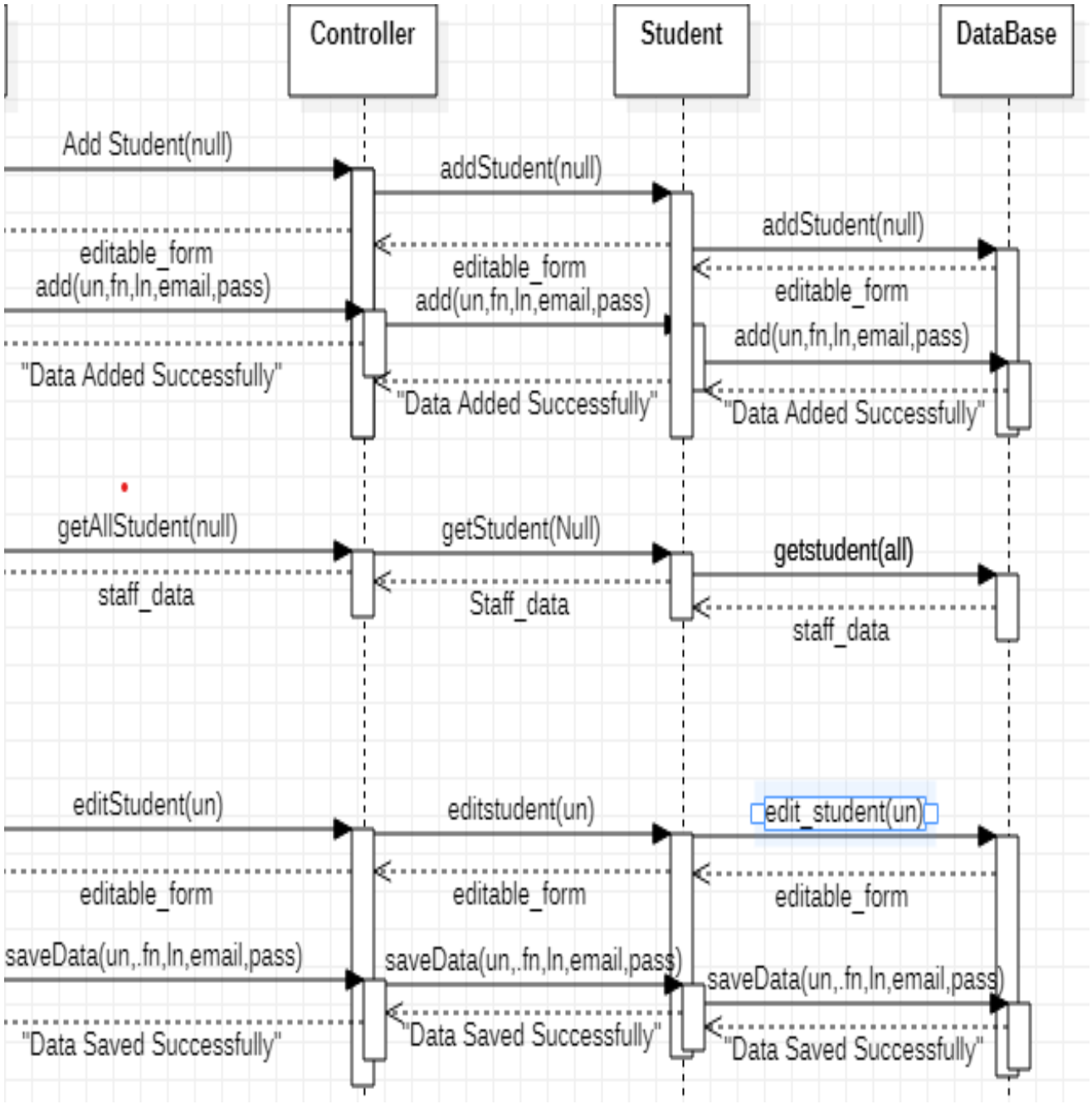


Figure 33Sequence Diagram Add/Manage Students

3.3.4 Sequence Diagram Add Program

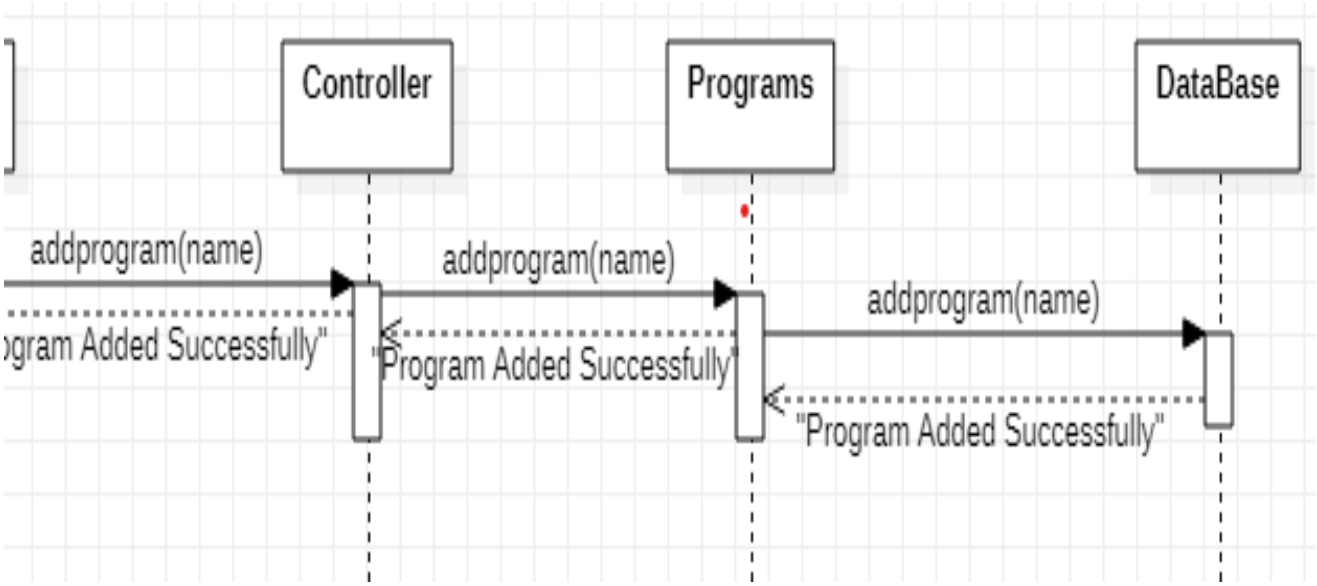


Figure 34Sequence Diagram Add Program

3.3.5 Sequence Diagram Add Class

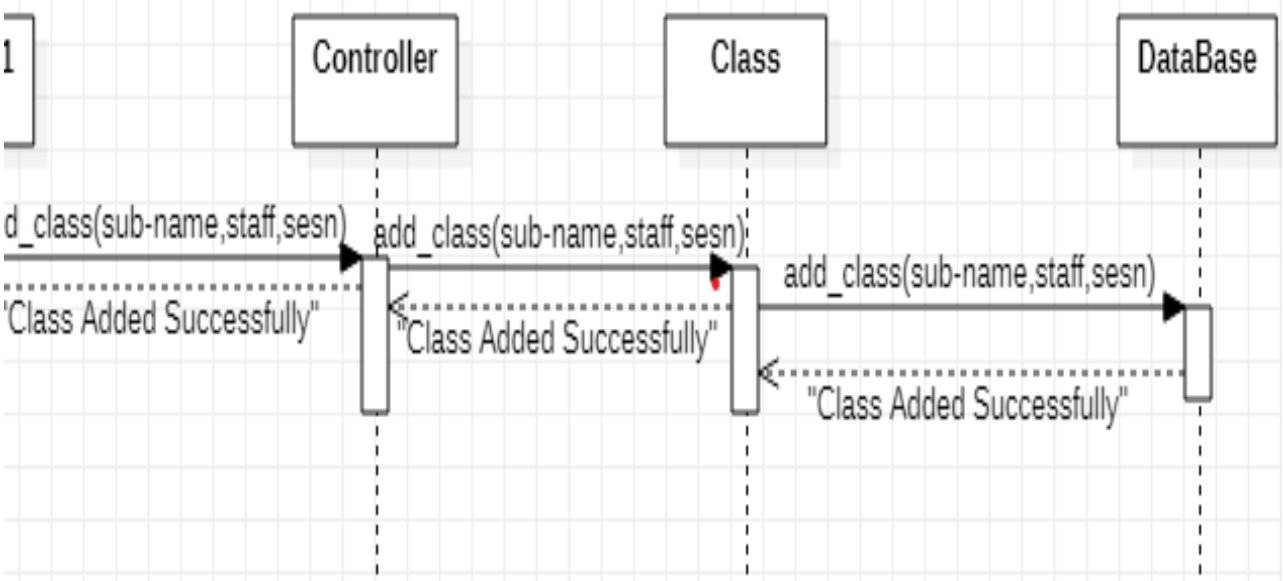


Figure 35Sequence Diagram Add Class

3.3.6 Sequence Diagram Mark Attendance

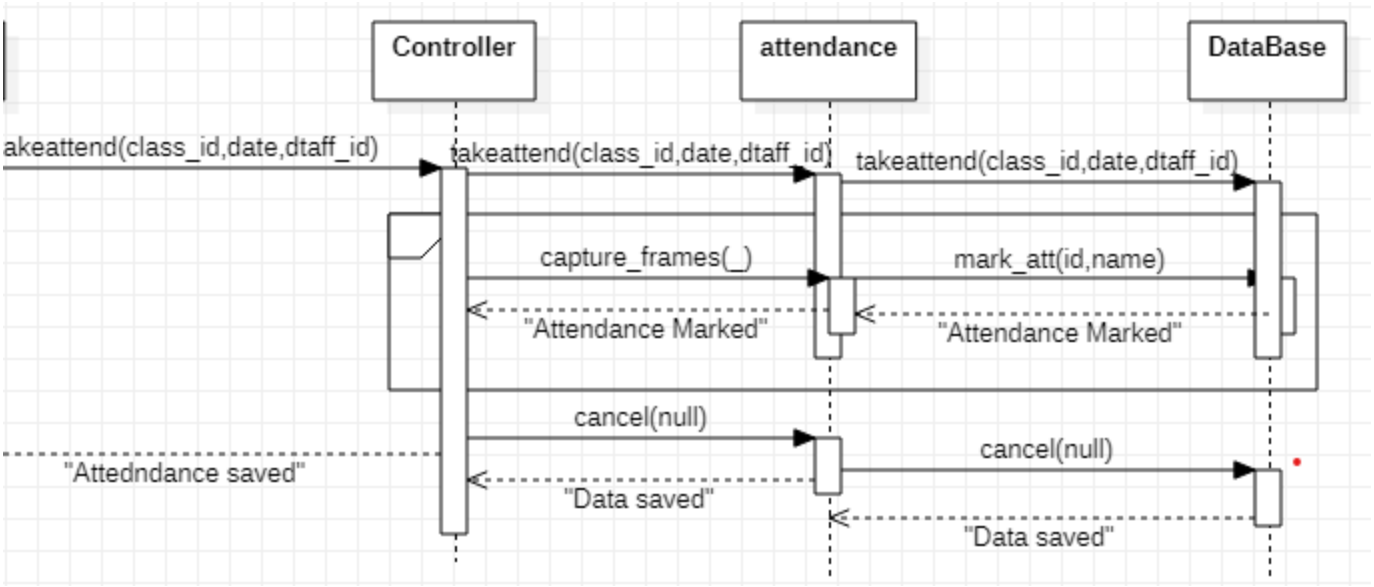


Figure 36Sequence Diagram Mark attendance

3.3.7 Sequence Diagram View/Send Notifications

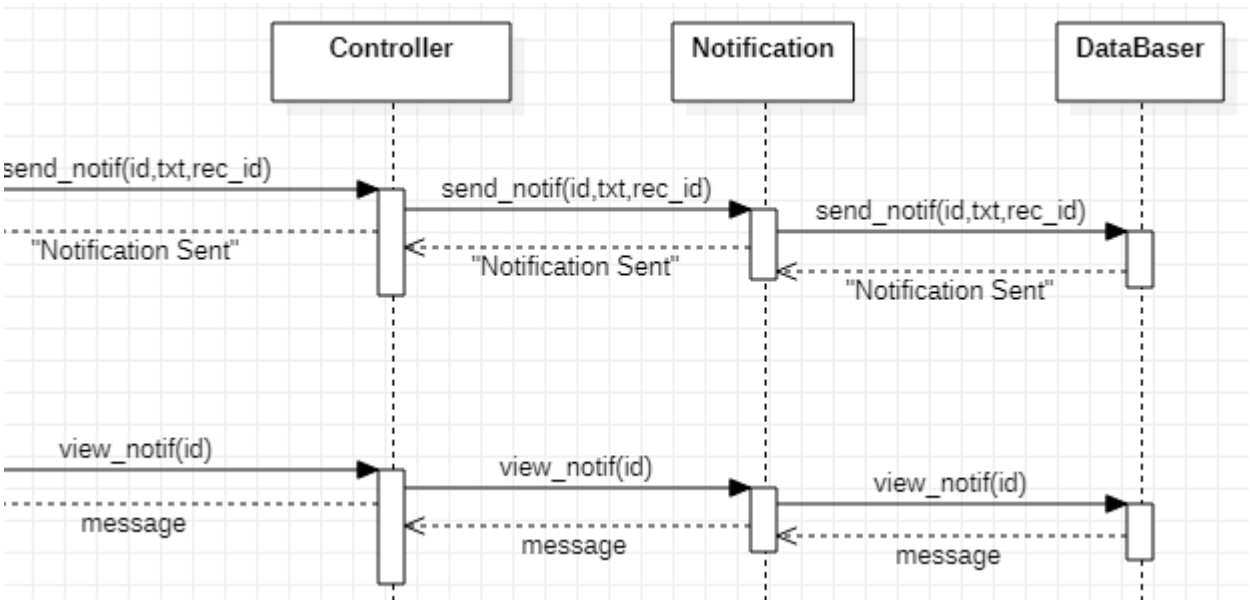


Figure 36Sequence Diagram view//send notifications

3.4 Class Diagram

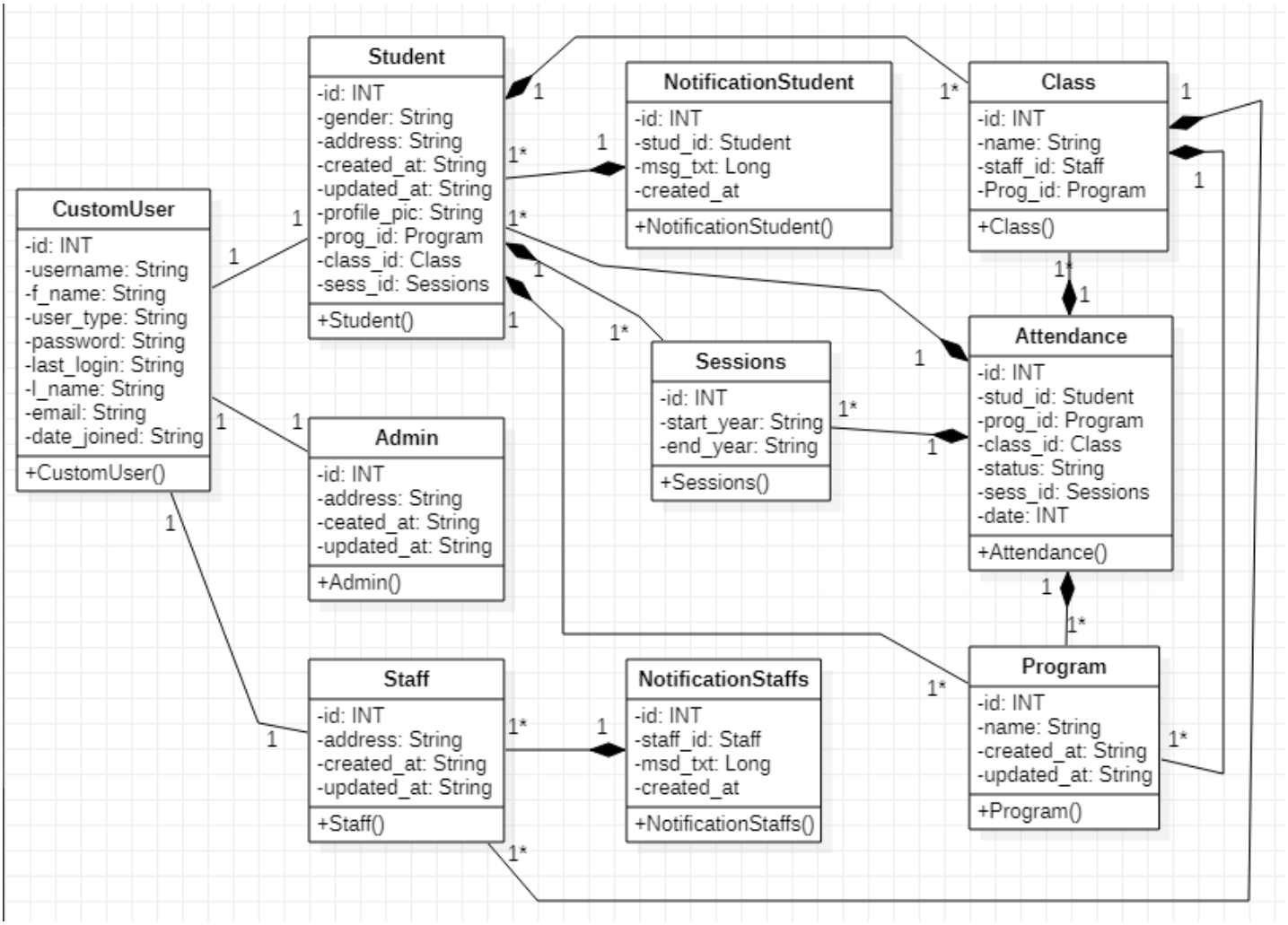


Figure 37 Class Diagram

Chapter 4
Software Test Documentation

4.1 Introduction

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

4.2 Test Plan

Test planning is an activity that ensures that there is initially a list of tasks and milestones in a baseline plan to track the progress of the project. Test plan determines the scope and the risks that need to be tested and are not to be tested. Deciding fail and pass criteria.

4.3 Features to be tested

Features to be tested are as following.

- Login
- Add Staff
- Add Student
- Add Program
- Add Class
- Send Notification Staff
- Mark attendance
- Student View Attendance
- Staff Logout

4.4 Test Cases

4.4.1 Login

ID	T1
Description	User is registered and login into its account.
Tester	User
Setup	User starts the application.
Instructions	1. Selects Login option. 2. Enter username as “adeel@gmail.com” and password as “adeel”. 3. Select ‘login’
Expected Results	Home page is displayed.
Actual Result	As expected
Status	Pass

4.1.2 Add Staff

ID	T2
Description	Admin will add staff successfully
Tester	Admin
Setup	Admin starts the application and login to the application.
Instructions	1. Select Add Staff option. 2. Enter email, username, first-name, last-name, password and address. 3. Select Add Staff option.
Expected Results	Message “Staff added Successfully”
Actual Result	As expected
Status	Pass

4.1.3 Add Student

ID	T3
Description	Admin will add student
Tester	Admin
Setup	Admin starts the application and login to the application.
Instructions	1. Selects Add Student option. 2. Enter first-name, last-name, username, email, pic-frames, password, select session and select program. 3. Select Add
Expected Results	Message “Student Added Successfully”
Actual Result	As expected
Status	Pass

4.1.4 Add Class

ID	T4
Description	Admin will add Class
Tester	Admin
Setup	Admin starts the application and login to the application.
Instructions	1. Selects Add Class option. 2. Enter subject-name, select program and then select staff member. 3. Select Add
Expected Results	Message “Class Added Successfully”
Actual Result	As expected
Status	Pass

4.1.5 Add Program

ID	T5
Description	Admin will add Program
Tester	Admin
Setup	Admin starts the application and login to the application.
Instructions	4. Selects Add Program option. 5. Enter program name. 6. Select Add.
Expected Results	Message “Program Added Successfully”
Actual Result	As expected
Status	Pass

4.1.6 Send Notification Staff

ID	T6
Description	Admin will add Program
Tester	Admin
Setup	Admin starts the application and login to the application and there must be at least one staff member added.
Instructions	1. Selects Send Notification Staff 2. Selects staff member and press send message option. 3. The write message in a pop-up window. 4. Selects send message.
Expected Results	Message “Message Sent”
Actual Result	As expected
Status	Pass

4.1.7 Mark Attendance

ID	T7
Description	Staff Mark the attendance for class
Tester	Staff
Setup	Staff logs in to the application and must have a class
Instructions	1. Select Take Attendance from side bar. 2. Select class, session and date 3. Select Capture Frames
Expected Results	Video Camera open and start detecting faces.
Actual Result	As expected
Status	Pass

4.1.8 Student View Attendance

ID	T6
Description	Student will view all his attendances.
Tester	Student
Setup	Student starts the application and login to the application.
Instructions	5. Selects View Attendance. 6. Selects class name, select start date and end date. 7. Selects fetch data
Expected Results	All his attendance shown
Actual Result	As expected
Status	Pass

4.1.9 Staff Logout

ID	T8
Description	Staff logs out from the application.
Tester	Staff
Setup	Staff is logged in to the application.
Instructions	Selects logout option.
Expected Results	Login page is displayed.
Actual Result	As expected
Status	Pass

Chapter 5
Conclusion and Future Enhancement

5.1 Introduction

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

5.2 Summary and Conclusions

This project ease the Management of Educational institutes as well as their staff member and their students. Admin can keep an eye on Attendances, Staff and Students. Everything is under his control. Staff can also keep an eye on his/her subjects and overall subject attendances and mark attendance in a very perspective and healthy manner with no unfair means. Students can also looks at his performance throughout his session.

Furthermore, this product is a Web based application specifically designed for web based users which also may be extended to the Android/IOS application for all Mobile Phone users over the time.

5.3 Future Enhancements

In future, the application can be enhanced by:

a) IOS application.
b) Android Application.
c) Fully automated with timetable added.
d) Parents/Guardians Interface can be added.
e) Email/SMS System can be added.
f) Proper Leave Management System

References

[1]: IEEE.IEEE Std 830-1998.IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998

[2]: https://docs.opencv.org/3.4/javadoc/org/opencv/face/LBPHFaceRecognizer.html

[3]: https://docs.opencv.org/3.4/da/d60/tutorial_face_main.html

[4]: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html: http://www.ndma.gov.pk/tools/vkc/vkc/gis/