Daffodil International University

Department of Software Engineering

SWE 698 Project

Project Documentation

University Management System

Supervised by:

Mr. Mohammad Khaled Sohel

Assistance professor

Department of Software Engineering

Daffodil International University

Submitted by:

Abdikani Mohamed Ali Ahmed

ID: 191-44-178

Department of Software Engineering

Daffodil International University

©Daffodil International University

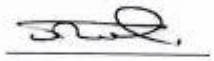
Approval

DECLARATION

I am hereby declared that, this project has been done by me under the supervision of **Md. Khaled Sohel, Assistant Professor, Department of SWE** at Daffodil International University.

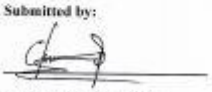
I also declare that this project and the documentation book on it are my own and have been generated by me.

Supervised by:



Md. Khalid Sohel
Assistant Professor
Department of Software Engineering
Daffodil International University

Submitted by:



Abdikani Mohamed Ali
ID: 191-44-178
Department of Software Engineering
Daffodil International University

©Daffodil International University

Acknowledgement

First of all, I am very much blessed as I have successfully reached the final semester. And so, this is why I am pleased with my almighty. From the very beginning of my university life, I have learned a lot about software engineering as well as core computer science-related knowledge from my course teachers. Moreover, they teach us ethics, morality, and politeness.

Besides, I am so much thankful to my parents that I was always supported by them. They always prioritize my opinion and inspired me as well.

I am also so grateful to my supervisor **Mr. Mohammad Khaled Sohel** for allowing me to work with this project. He always supports me to make this project successful. Especially, I am very thankful also to my all lecturers in the department.

©Daffodil International University

Table of Contents

©Daffodil International University

©Daffodil International University

List of Figures:

Chapter 1
Introduction

1.1 Project Overview

Today we have computers with large computing power and almost every business is going to take the advantages of using those technologies. But nowadays digital certifications itself become an essential component for every business infrastructure. Because it provides security and it can identify every unique individual. Besides, it also provides confidential communications to the users.

This university management system I developed it to solve the problems of the universities and make work smart instead of using papers now you can you use this system to do all the work, every university or educational institution has challenges to overcome and managing the information of students, faculties, registrations courses and staff at the management level, this system designed to assist strategic planning and will help to ensure that the university can meet the minimum standard of the university.

The main aim of this Project University Management System is to manage student registration as well as staff registration so that the teachers can submit student exam results online and take class attendances online during class sessions.

University Management System deals with the maintenance of university faculties, Departments, classes, courses, Courses, teachers and students within the university system. This UMS involved in automating university and manage student information and staff, the students can see their exam results in their portals.

This is an application that monitors students Results and class attendance, the student will have been registered and all the information of students can get after that by click.

1.2 Project Purpose

The main purpose of this project named “University Management System” is to make an automation system which might be helpful for thousands of students to get exam results as the shortest time and help the university to reduce the manual work for managing Students, Teachers, Courses, Classes, Attendance, and Results. So that’s why we are going to develop such as this project.

1.3 Background

I was working in university in my country, the university did not have any system that we can manage student like registration, exam results, and attendances we had faced a lot of problems, so when I learned the programming language I said to myself can you develop an application to that university because I know their needs very well, and help them to manage the student and results. I started working on this project since that and I hope I have solved 70% of the problem and I hope I will solve the remaining.

1.4 Benefits & Beneficiaries

Our system would be beneficial for some point of view. Now, I am mentioning those below:

- Our system helps the university to manage student's registration
- It helps to manage the course of the university and assign teacher for them
- This system also helps the university to manage classes • It is also helpful to manage faculties and departments
- It also helps the university to manage teachers and staff
- This system will help to submit exam results online
- This system will help from university to manage class attendance
- This system will also help students to get exam results online.

I have also mentioned some benefits as well as beneficiaries. So, I think this application is very much helpful for the University.

1.5 Goals

The main goals of this project are to develop an application based on web. That can students, staff, faculties, department, courses, results and attendance can be taken as online by accessing the application through internet, that 24/7 should be available and students can access to their portals.

1.6 Stakeholders

There are three types of stakeholders in our "University Management System". Such as:

- University System Admins
- University Teachers
- University students

Now, I will write a brief description of the stakeholders.

- **University System Admin:** university system admin can be (academics director, registrar, IT director) or can be any other staff of the university who will have permission to manage this system these are main activities: student registration, course registration, and teacher registration
- **University Teachers:** The teacher is one of the stakeholders of this system because they are the ones who will insert student exam results and take class attendance.
- **University students:** Students are also stakeholders of this system when the teacher inserts student exam results Student will log in to the student portal and see the exam results online and access the system through the portal.

1.7 Project Schedule

We need to prepare a scheduling plan to complete the project on time. It also refers to make communication with what task need to get done within timeframe.

1.7.1 Gantt Chart

Gantt chart is mainly a production control tools. It remained us to complete our assigned tasks within a certain period of time. For developing software, it is mostly used. Now I will show a Gantt chart for our project.

Activities		W 1	W 2	W 3	W 4	W 5	W 6	W 7	W 8	W 9	W 10	W 11	W 12	W 13	W 14	W 15	W 16
Planning	Ideas																
	Problem definition																
	Proposal planning																
Requirements	Requirement specification																
	Requirement analysis																
QA - 1	Quality assurance																
System design	Sketching																
	Design specification																
	Database design																
Implementation-1	Searching vehicles																
QA – 2	Test cases																
Implementation-2	Impose case & demerits																
Testing	Unit testing																
	Blackbox testing																
Delivery	Software release																

Scheduled time	
Buffered time	

Figure 1.1: Gantt Chart

1.7.2 Release Plan or Milestone

The release plan or milestones are given below:

Activities	Duration in week	Total week
Brainstorming	Week 1	1
Problem identification	Week 1, Week 2	2
Requirement specification	Week 2	1
Requirement analysis	Week 2	1
Sketching	Week 4	1
Design specification	Week 4	1
Database design	Week 5	1
Vehicle search	Week 5, Week 6, Week 7, Week 8	4
Quality assurance	Week 3,	1
Test case	Week 3, Week 7, Week 8, Week 9	4
Impose case & demerits	Week 10, Week 11, Week 12, Week 13	4
Unit testing	Week 11, Week 12	2
Black-box testing	Week 13, Week 14, Week 15	3
Software release	Week 16	1

©Daffodil International University

Chapter 2
Software Requirement Specification

2.1 Functional Requirements

Functional requirements refer to the functions which are mandatory to the system. Functional requirements must be able to perform on the software system. Every system must have some functional requirements. Now, we are going to mention functional requirements associating with our project.

2.1.1 User Registration

Requirements 1	**Faculty Registration**
Description	System Super Admin can only register the new User like Academic's, Registrar other system admins. Super Admin He can only manage the users in the system he can update delete users.
Stakeholders	System Super Admin

2.1.2 Faculty Registration

Requirements 2	**Faculty Registration**
Description	university Academic Director or System Admin they can only register the new faculty name, and also update the Faculty information.
Stakeholders	University Staff (Academics), System Admin

2.1.3 Department Registration

Requirements 3	Department Registration
Description	university Academic Director or System Admin they can only register the new Department and select the department Faculty, and also, they can update the Department information.
Stakeholders	University Staff (Academics), System Admin

2.1.4 Class Registration

Requirements 4	Class Registration
Description	university Academic Director or System Admin add the new Class name with class number and assign this class room in to his department and faculty, and also update the Class information.
Stakeholders	University Staff (Academics), System Admin

2.1.5 Semester Registration

Requirements 5	Semester Registration
Description	university Academic Director or System Admin add the new semester name that every course and students will register for it
Stakeholders	University Staff (Academics), System Admin

2.1.6 Registrar officer add Students data

Requirements 6	Registrar add new Student
Description	The university registrar officer is going to register a new student and add them in to the database, so that he will complete the registration form of the student, the register will select the student registered faculty, Department, Class and Semester with unique student id, student picture and save it. After that he can Edit it, update and Delete student Data from the database.
Stakeholders	University Staff (Registrar)

2.1.7 Teacher Registration

Requirements 7	Employee Registration
Description	university Academic Director or System Admin register the new Teachers of the university and give them unique Employee-ID. teachers have their own portal that the can add student results and take class attendance for his registered course only.
Stakeholders	University Staff (Academics), System Admin

2.1.8 Course Registration

Requirements 8	Course Registration
Description	university Academic Director or System Admin register the new course name with course code and select the faculty, department, class and semester for this course then assign it for teacher of this course. university Academic Director or System Admin can also update the course name, Course code and can change or re-assign this course to a new teacher.
Stakeholders	University Staff (Academics), System Admin

2.1.9 Exam Schedule

Requirements 9	Exam Schedule
Description	university Academic Director or System Admin can add the exam schedule for the university this exam schedule both Teachers and Student see and prepare for the exam. Is like exam notice.
Stakeholders	University Staff (Academics), System Admin

2.1.10 Teachers Submit Exam Results

Requirements 10	Teachers Submit Exam Results
Description	Teacher submit exam result as online with assignment marks, mid-term marks, presentation marks and final exam marks. Teacher will submit the result when every semester if finished, when he submits the results student can see their course result.
Stakeholders	University Staff (Teachers)

2.1.11 Teachers Take Class Attendance

Requirements 11	Teachers Take Class Attendance
Description	Teacher they will take every class attendance by online and submit it through online this activity of taking attendance every day for students the system will calculate the number of days that this student is present and generate the attendance marks, when the semester ends the teachers will generate these marks into student result sheet.
Stakeholders	University Staff (Teachers)

2.1.12 Teacher Course Student Attendance Report

Requirements 12	Class Attendance Report
Description	Teachers can see their course student's attendance report so that he can identify the students normally missing out the class.
Stakeholders	University Staff (Teachers)

2.1.13 Student Portal

Requirements 13	Student Portal
Description	Student portal this requirement is every student should have portal that they can see their exam results and exam schedule. Every student has unique portal, when they login the write their student id and password:12345 which is common
Stakeholders	Students

2.1.14 Students change their Password

Requirements 14	Students change their Password
Description	Student can change their student portal Password because the common password is not much secure so every student can change his password for whatever he likes.
Stakeholders	Students

2.2 Non-Functional Requirements

2.2.1 Performance

It is very important to maintain performance of any software system. To ensure performance, we need to maintain some steps. Now, I will explain some perspective by which we are going to enhance the performance of our project.

2.2.2 Speed & Latency

Speed and latency requirements must be ensured while retrieving data from the cloud server. When the Registrar needs to search student to update or when teacher wants to submit results, then the search result must show within seconds.

2.2.3 Capacity

The developed system by us must be capable to handle user data, provide accurate information, handling database, manage http request etc.

2.2.4 Maintainability & Supportability

It is very important to provide after service or support to the end users

2.2.5 Supportability

Supportability requirements may have related to some extends. Like:

- Testability
- Extensibility
- Adaptability
- Maintainability

- Compatibility
- Configurability
- Serviceability
- Install ability

Our application meets all of the above requirements related to supportability.

2.2.6 Security

Making software security as a requirement is very important. Software security requirements should be its functional requirement. Software security enforces security of an application system. Functionality related to software security can either be directly tested or observed.

While accessing to the system, each and every module must provide a central authentication mechanism. There is also a process to prevent entering into the system by ensuring hashed password for the unauthenticated users.

2.2.7 Accessibility

For accessing to our application system, there remains some authentication and authorization techniques. And every module of our system will provide it.

2.2.8 Integrity

Integrity requirements refers to a security system which ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to the malicious modification or accidental destruction. For that reason, we will store our user passwords as encrypted format which is impossible to decrypt. It is also called hashed password.

2.2.9 Privacy

It is very important to ensure privacy of the system users. Privacy requirements enhances to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database should be protected by the anonymous. Users are permitted to get access to those data which are being associated by them which can be ensured by the user log in system.

2.2.10 Usability and Human-Interaction

The main target of developing any system is to make the system user friendly and easy to usable for the end users

2.2.11 User Documentation

Documentation are mainly two types. One is internal documentation which is generally written by the application engineers. It is prepared to make development life cycle easier for the system engineers or system analysts.

2.2.12 Training

Training requirements involved in after service of any application. It is very necessary to properly train up end users to the system so that they would be capable to operate easily. After launching the full package to the market, firstly we provide training to the different end users like University Staff, Teachers and also Students.

2.2.13 UI/UX

UI/UX requirements mainly refers how the system will look like and how the user interface or graphical user interface of our system will display to the user.

2.2.14 Legal

Legal requirements normally refer to the terms and conditions or privacy policy of any organizations. The terms and condition of our application is that, no third-party software or person are allowed to engage to use our data for their business purposes.

Chapter 3
Requirement Analysis

3.1 Use Case Diagram

This Use Case Diagram is a graphic description of the interactions among the elements of University Management System. It represents the methodology used in the system analysis to identify, clarify, and organize system requirement of University Management System. The main actors of University Management System in this **Use Case Diagram** are: Super Admin, System User, Students who perform different type of use cases such as Student Registration, manage (faculties, Departments, Classes), Manage Course, Result Management, Taking Student Attendance, Student see their results, Manage Users and full University Management System Operations. Major elements of Use Case Diagram of University Management System are shown on the picture below.

- **Super Admin Entity:** Use Case of Super Admin are Student management, management of (faculties, Departments, Classes, Semesters), Course management, Result management, Attendance management, Manage Users and Full University Management System Operations.
- **System User Entity:** Use Case of System user are divided into two
    - **Academic Staff User:** this user is normal academic staffs like (Academic Director, Registrar and other officials) they are responsible for Registration Student management, management of (faculties, Departments, Classes, Semesters), Course management, Result management, Attendance management.
    - **Teacher User:** this is user is responsible to add the Student Results and take class attendance
- **Student Entity:** Use Cases of Student are for getting Exam results.

Use Case Diagram University Management System

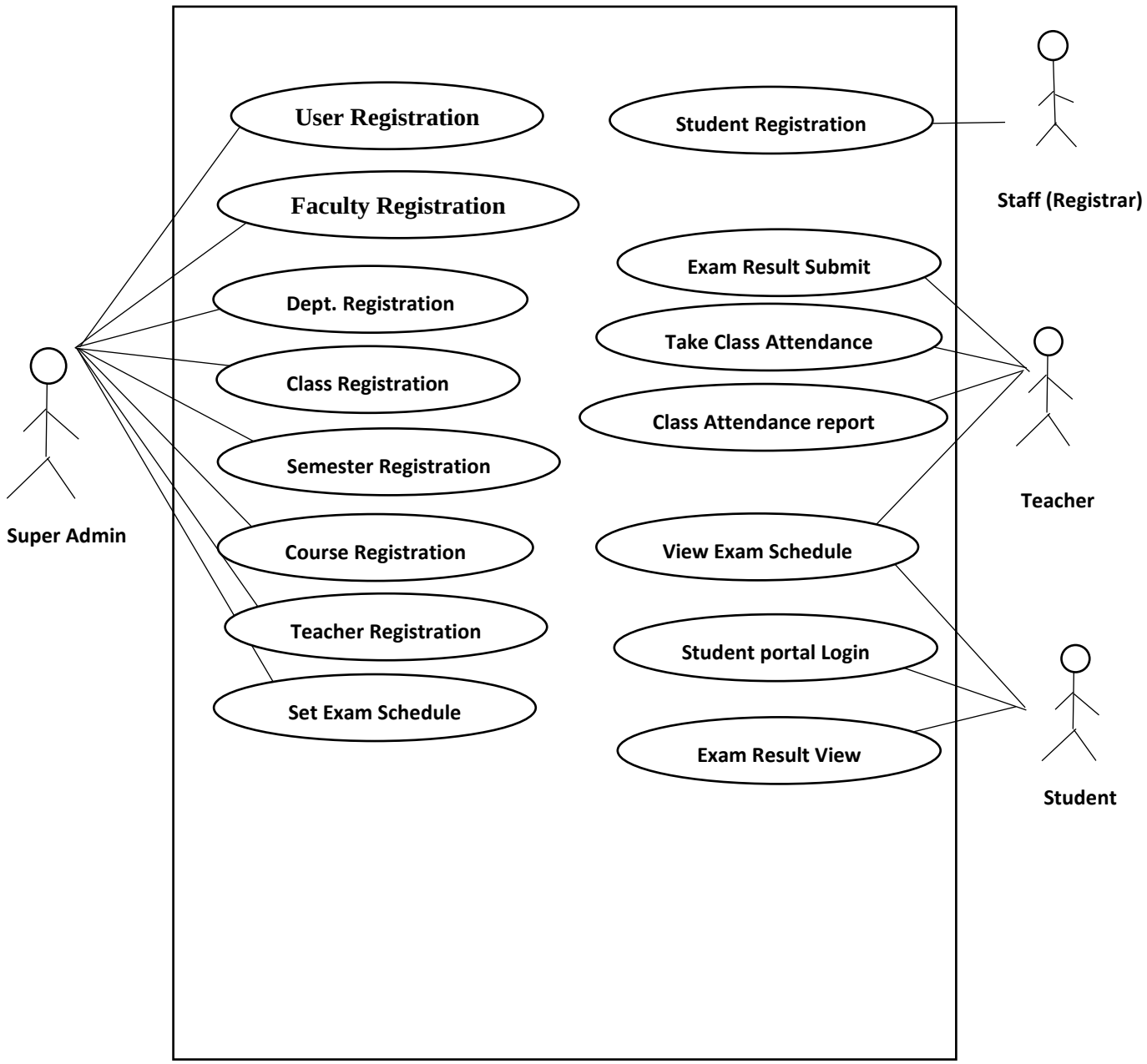


Figure 3.1: Use Case Diagram for University Management System

3.1.1 Users Registration

Use Case ID	UC001
Use Case Title	Users Registration

Description	System Super Admin has the authority to register new user like (system Admin, Academic and Registrar).
Actor	• Super Admin
Pre-condition	User are not registered
Post-condition	User registered and get access to the system
Flow	• Register new user with his full name • Register with email and password • Give permission to the user • Give role to user • Share the user with staff • Maintain the user logs

3.1.2 Faculty Registration

Use Case ID	**UC002**
Use Case Title	**Faculty Registration**
Description	System Super Admin and Academics has the authority to register new Faculty and update existing faculty
Actor	• Super Admin • Academic Director
Pre-condition	Faculties are not created
Post-condition	Faculties are created and These faculties created should be available in all other forms that needed to be filled. Like student Registration, Teacher and course
Flow	• Register faculty • Update existing faculty

3.1.3 Department Registration

Use Case ID	**UC003**

Use Case Title	Department Registration
Description	University department should be created.
Actor	• Super Admin • Academic Director
Pre-condition	Departments are not registered
Post-condition	Departments registered and the created Departments should be available in all other forms that needed to be filled. Like student Registration, Teacher and course
Flow	• Register Department • Update existing department

3.1.4 Class Registration

Use Case ID	UC004
Use Case Title	**Class Registration**
Description	University class should be created So, classes will have name and class No
Actor	• Super Admin • Academic Director
Pre-condition	Classes not added
Post-condition	Classes added into the system and they should be available in all other forms that needed to be filled. Like student Registration, Teacher and course
Flow	• Register Class • Update existing Class

3.1.5 Semester Registration

Use Case ID	UC005
Use Case Title	**Semester Registration**
Description	University semester should be created because the university works as semesterly every activity based on university semester.
Actor	• Super Admin • Academic Director
Pre-condition	New semester is not created
Post-condition	Semesters are created and should be available in all other forms that needed to be filled. Like student Registration, Teacher and course
Flow	• Register new Semester

3.1.6 Course Registration

Use Case ID	**UC006**
Use Case Title	**Course Registration**
Description	This course registration it helps to assign teachers their specified course that they are teaching and student when they are registering, they will choose their course they needed to take in this semester
Actor	• Super Admin • Academic Director • Teachers • Students
Pre-condition	Courses are not registered
Post-condition	Courses are registered and these courses registered will be available in all other forms that needed to be filled. Like student Registration, Teacher and course
Flow	• Register new course • Register the course with a code • Assign this course to teacher • Register this subject to students • The teacher will see only students registered to this course. • Update course teacher • Update course information

3.1.7 Teacher Registration

Use Case ID	**UC007**
Use Case Title	**Teacher Registration**
Description	Is to register a new teacher that can upload student exam results and take class attendance
Actor	• Super Admin • Academic Director • Teachers
Pre-condition	Teachers are not registered
Post-condition	Teachers are registered, the registered teachers will be submitting student exam result and take class attendance.
Flow	• Register new teacher • Give teacher username and password • Access his dashboard • Submit student results • Take class attendance • Update teacher profile • See student attendance report

3.1.8 Exam Schedule

Use Case ID	**UC008**
Use Case Title	**Exam Schedule**
Description	Is to announce examination schedule or when exams are going to start like mid-term exams and final exams.
Actor	• Academic Director
Pre-condition	Exam schedules are not set yet
Post-condition	Exam schedules are added, student and also the teacher will see the exam schedule for the semester
Flow	• Add new exam schedule for new semester • Students will see examination schedule • Teacher also will see exam schedule

3.1.9 Student Registration

Use Case ID	**UC009**
Use Case Title	**Student Registration**
Description	University registration office will register the new students for the new semester when they are registering the new student they have to be sure to select faculty, department, semester, class and select the course is going to take this semester this information of student will be available in all activity in system later so is very important.
Actor	• University Registration Staff • student
Pre-condition	Students are not registered
Post-condition	New students registered, when Student registered the teachers can take class attendance and insert results to that student and the student can see results from student Portal.
Flow	• Register new student • Confirm that the all field is filled and faculty, department, semesters, classes and courses selected • Test student portal is working • Update student information • Flow student results and attendance to ensure quality.

3.1.10 Teacher submit Student Result

Use Case ID	UC010
Use Case Title	**Teacher submit Student Result**
Description	The teacher will submit when semester ends the results for exam and upload online so that students can see online
Actor	• Teachers
Pre-condition	Exam results is not submitted
Post-condition	teacher submit the results, after that students can see their results in the student Portal.
Flow	• Teacher submit results • Ensure every student is available in the list • Ensure before teacher submit the results all information are right • Attendance will be generated from attendance summary. • Teacher will submit the results with specified period. • Teacher can update the results

3.1.11 Teacher take class Attendance

Use Case ID	UC011
Use Case Title	**Teacher take class Attendance**
Description	The teacher will take every class the attendance of the class and submit online
Actor	• Teachers
Pre-condition	Teacher take class attendance manually
Post-condition	Now the teacher takes class attendance through the system, the attendance taking every class will be converted into student's attendance results, the teachers will generate the marks from attendance summary and update the result of the students' course.
Flow	• Teacher take class attendance every class • Ensure that he take attendance proper way • See class attendance summary • Generate attendance marks into student results

3.1.12 Teacher see class attendance

Use Case ID	UC012
Use Case Title	**Teacher see class attendance**
Description	The teacher can see his course student's attendance report like every class how many days he presents and absents the students.
Actor	• Teachers • Academics
Pre-condition	Teacher see student class attendance manually
Post-condition	The teachers and academics staff will able to see detailed attendance report of specific course that they can see student's attendance report
Flow	• Teachers and academic staff see the attendance report.

3.1.13 Student Portal Login

Use Case ID	UC013
Use Case Title	**Student Portal Login**
Description	Student will have portal that they can see their results and see exam schedules.
Actor	• Students
Post-condition	Login into student portal
Flow	• Students login to their student Portal using their ID and given password: 12345.

3.1.14 Student see Exam Results

Use Case ID	UC014
Use Case Title	**Student see Exam Results**
Description	Students when they are login to their portal, they can see their results with semesters
Actor	• Students
Pre-condition	Students see exam results by paper attached in to the wall with the whole class. Also, student know exam schedules from the office.
Post-condition	Now students see their results online from student portal individually that means no one can see your results. And also, students can see exams schedules through the portal without going to office.

Flow	• Students see exam results by semesters. • Students see exam schedules.

3.2 Activity Diagram

We have prepared some activity diagram according to our use case. These activity diagrams are properly referring the flow of the individual conditions of our project.

3.2.1 User Registration

At beginning System Super admin will create the system users like academic Director, Registrar and also other system Admin. These users will have some Roles and Permission that they can create another user like teachers.

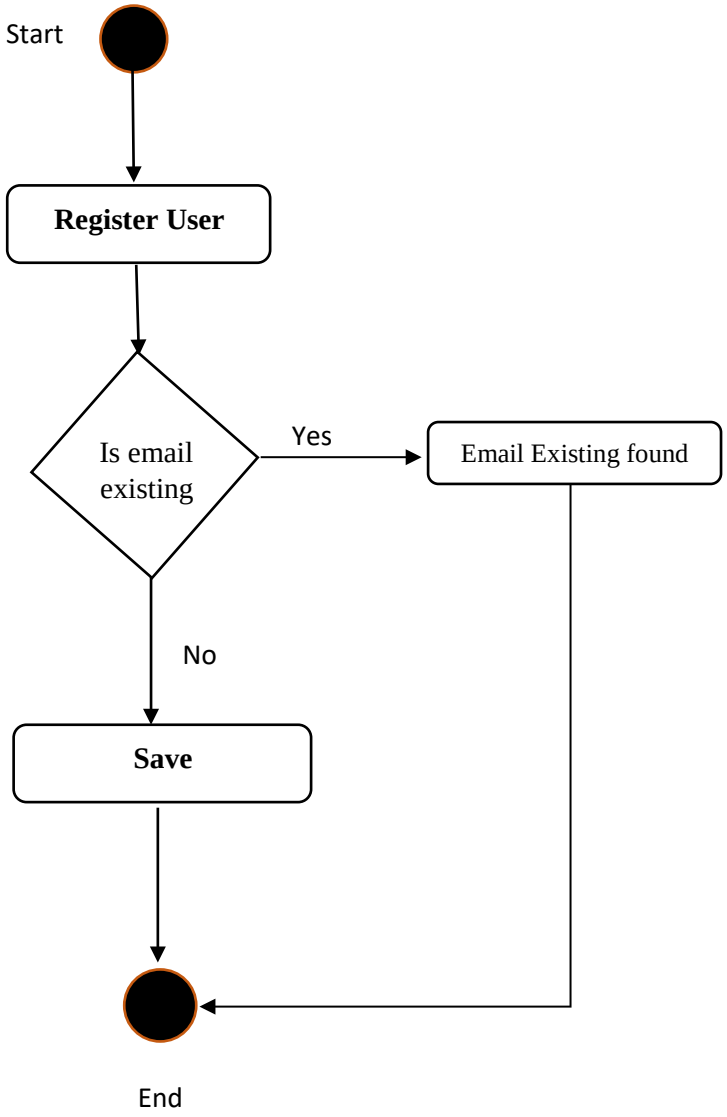


Figure 3.2: Users Registration

3.2.2 Faculty Registration

Second operation is to create faculty or register so that this faculty later on when students are going to register, they will select faculty. We will register all university faculties at one time if later on one faculty is added we can register also.

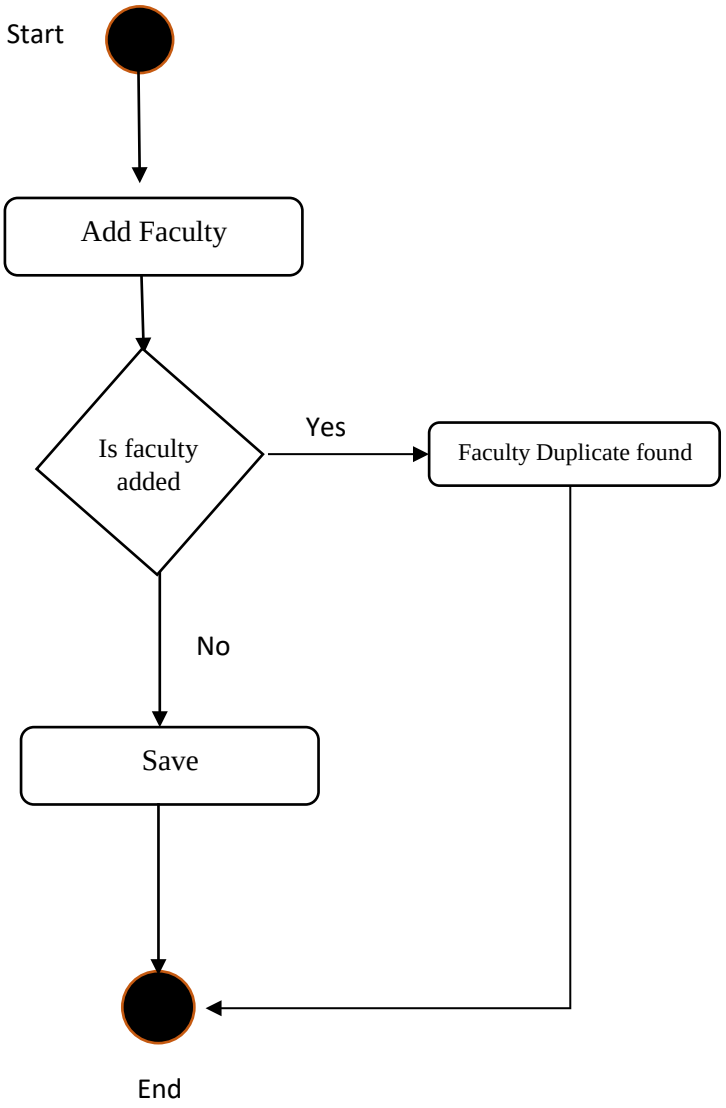


Figure 3.3: Faculty Registration

3.2.3 Department Registration

Third operation is to create departments of the university and align every department with her faculty, when the **Registrar** is registering the students it will select student Department. We will register all university Departments at one time if later on one Department is added on, we can register also.

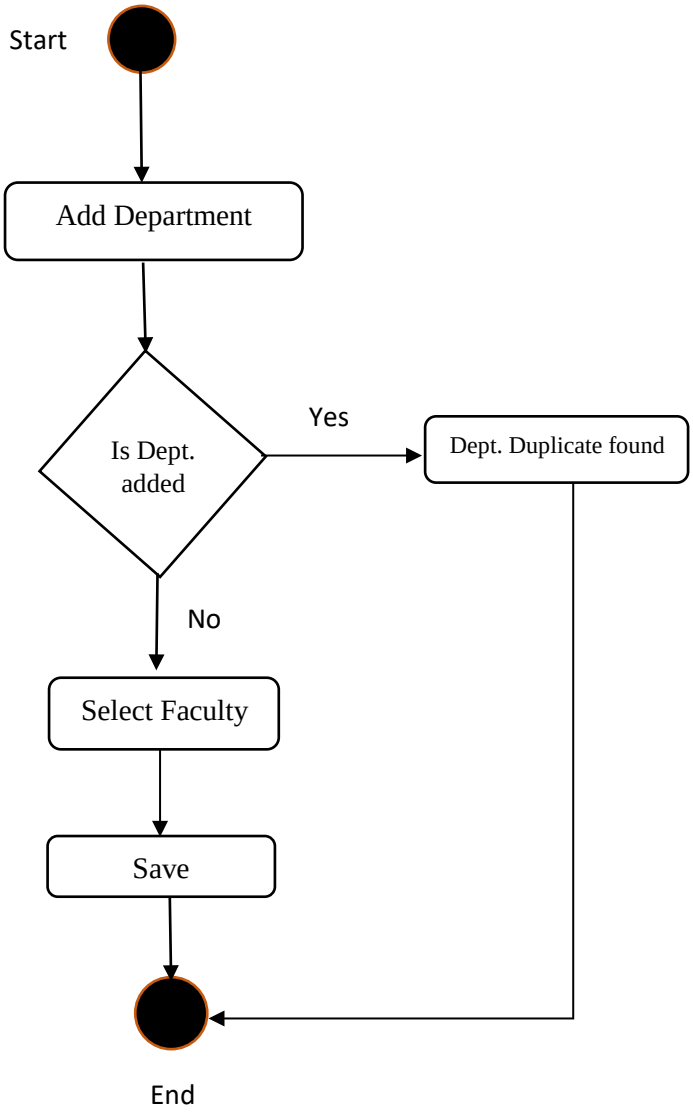


Figure 3.4: Department Registration

3.2.4 Class Registration

Fourth operation is to create university Classes and align every Class with specific Department and faculty, when the **Registrar** is going to register the students it will select student Faculty, Department and also classes so that this student knows which class he will take lessons. We will register all university Departments at one time if later on one Department is added on, we can register also.

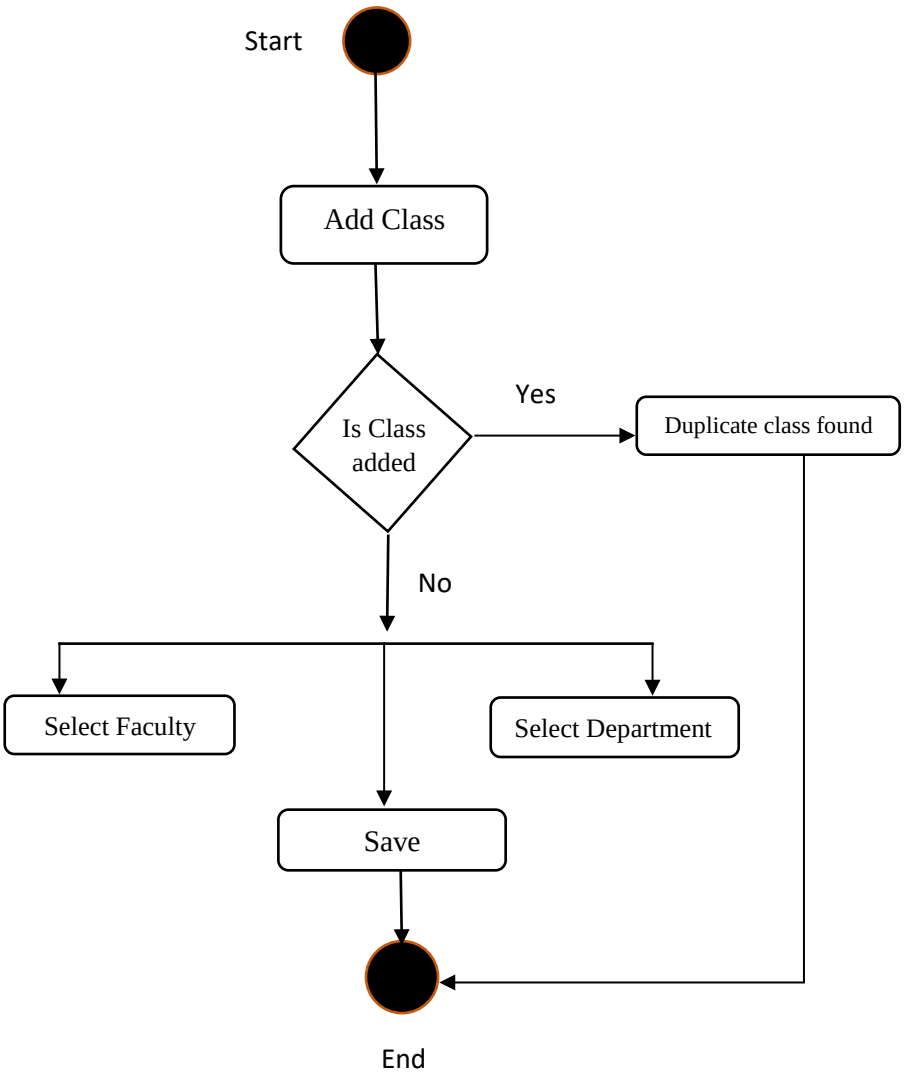


Figure 3.5: Class Registration

3.2.5 Add Semester

The next operation is to create university Semester because universities normally their colanders based on Semester, this semester will be use in all other forms like student registration form, teacher Registration, Result summation and taking class attendance .We will register the semester at beginning of each semester so that we can identify each activity which semester happened.

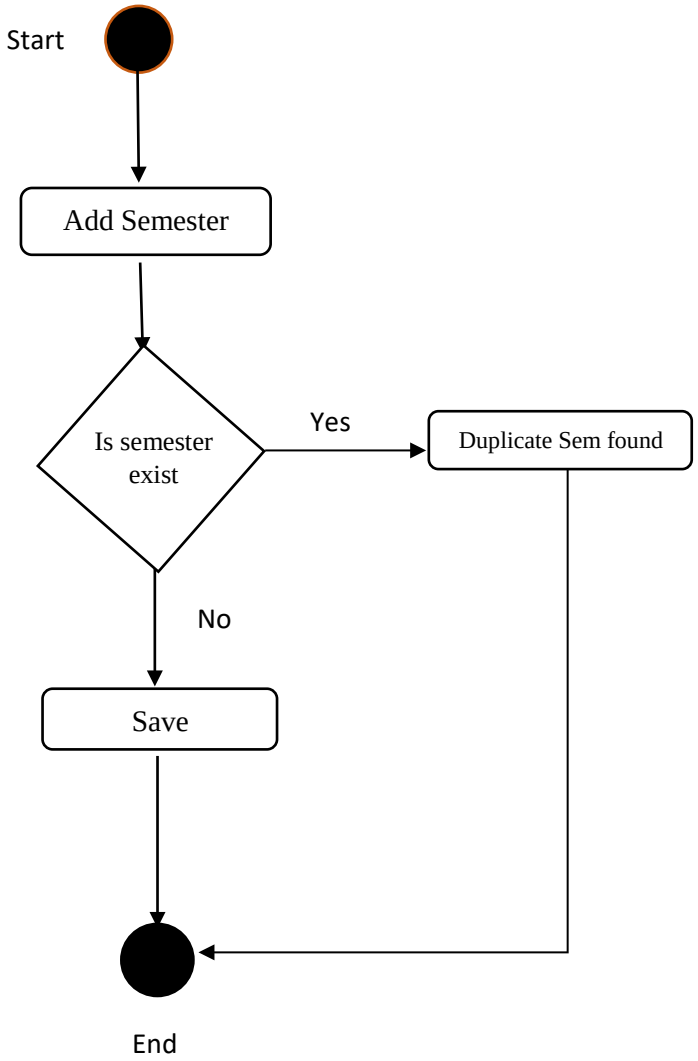


Figure 3.6: Add Semester

3.2.6 Course Registration

Course registration is to registrar the courses that will be teaching during the semester, select the course with faculty, Department, class and Semester, and assign each course Teacher that is going to teach this course.

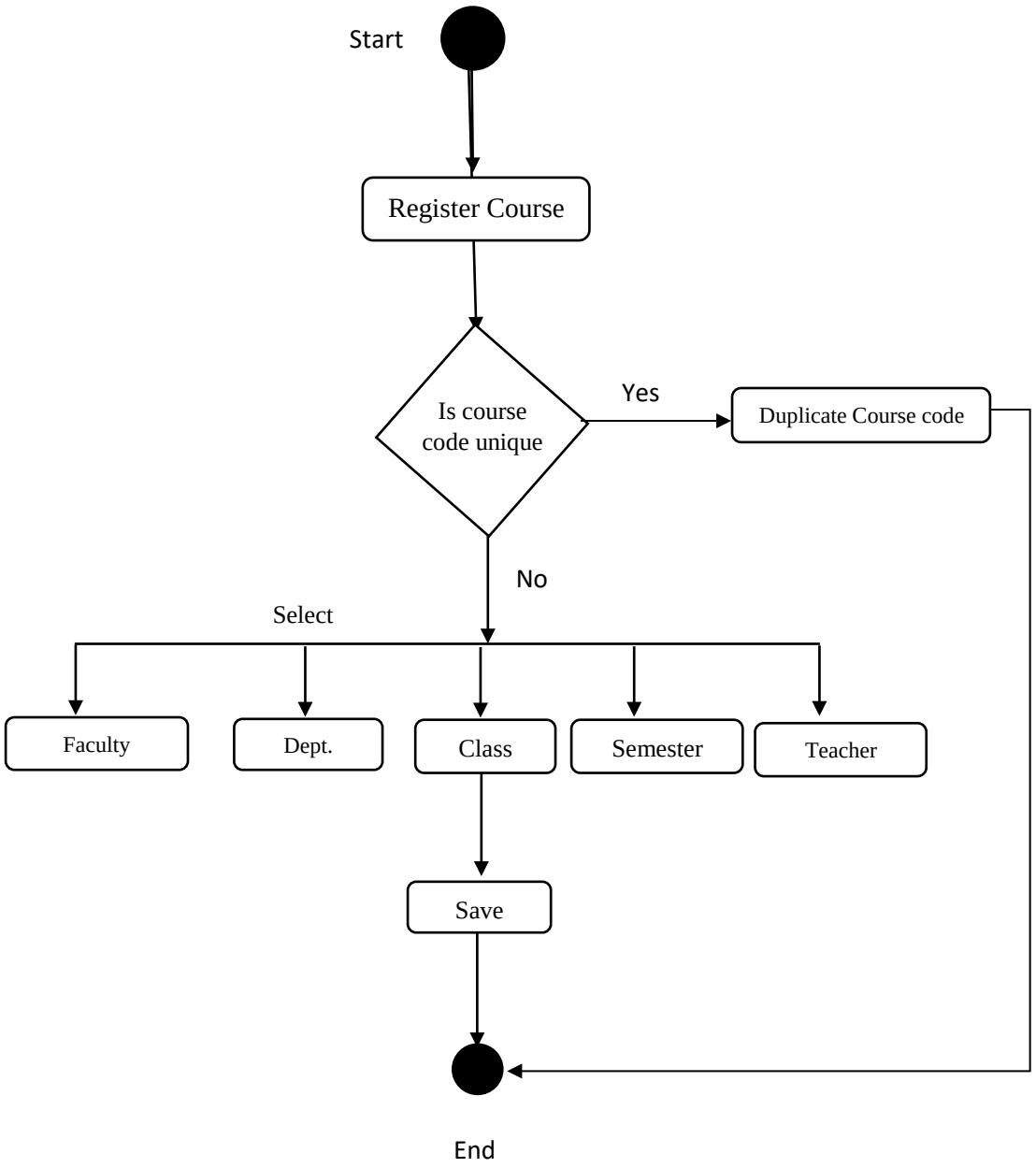


Figure 3.7: Course Registration

3.2.7 Teacher Registration

Teacher registration is one important feature in this application because is responsible to submit exam results and take class attendance. When Admin or Academics Director Register New Teacher it will user-name and password that he can login into his Dashboard and access result management and take attendance class in every day and see student attendance report.

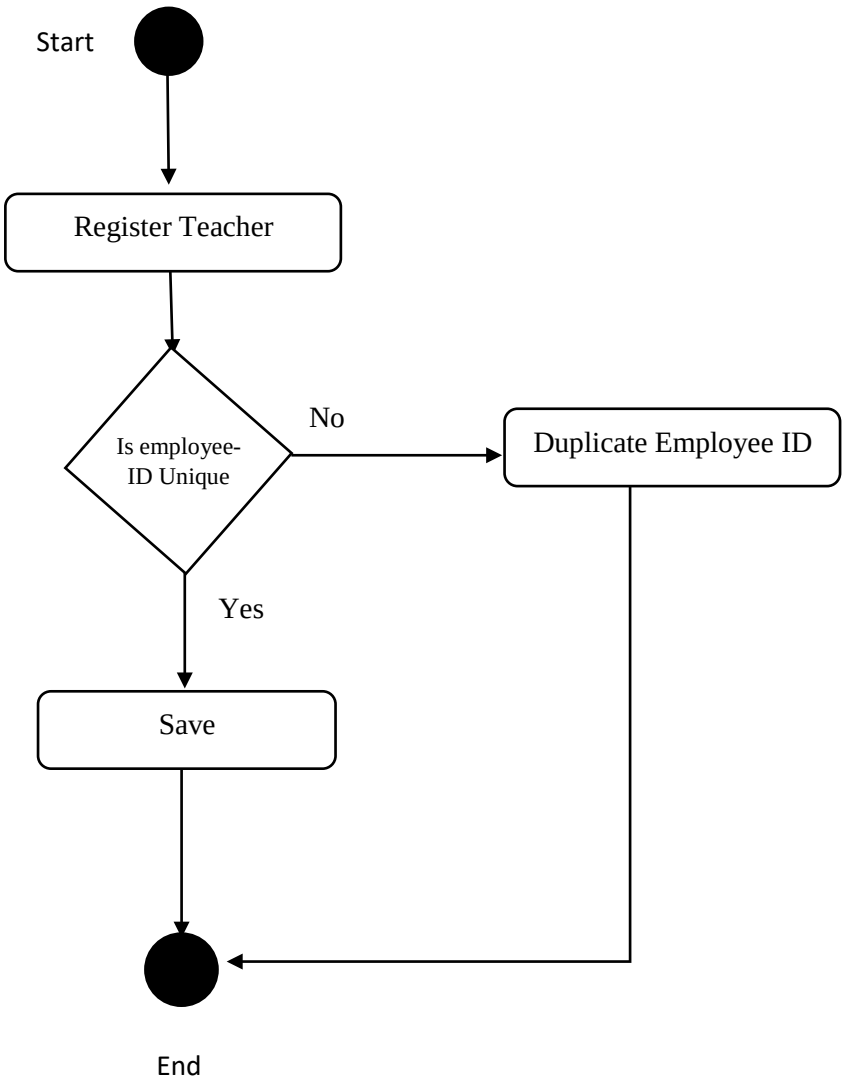


Figure 3.8: Teacher Registration

3.2.8 Student Registration

Student registration is the main activity of this system and largest information that needed to be submitted, the registrar is responsible to register all the students, in the registration form every student should have unique ID and Reg-NO and also registrar should select faculty, department, class, semester and courses that this students registered. Also, at the u have to give student User-name and Password that he will use to access the Portal, User-name will be his ID No and

password is common: 12345

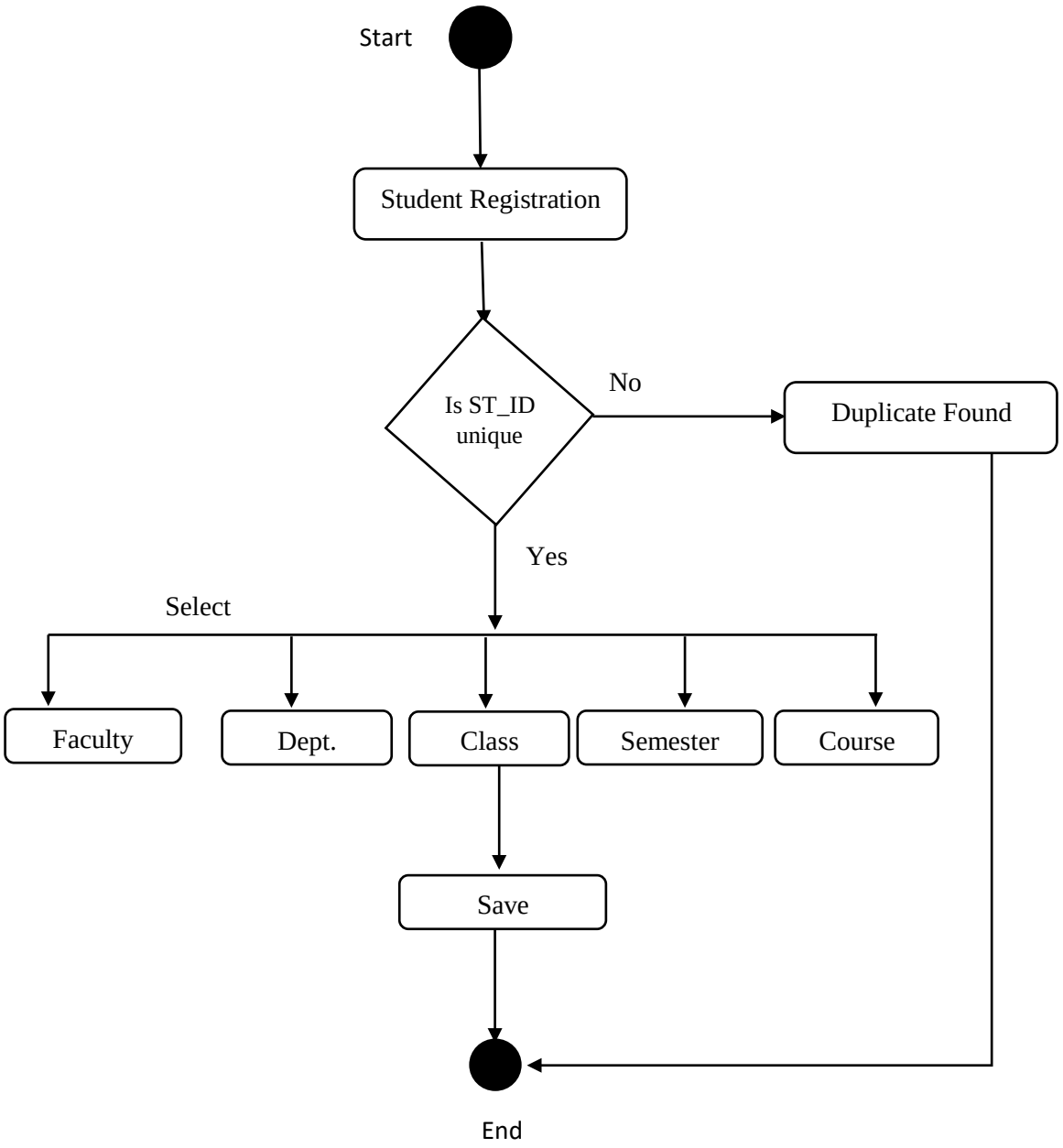


Figure 3.9: Students Registration

3.2.9 Exam Schedule

Exam schedule is to set examination time so that student and teachers will be aware the date of examination is simple information

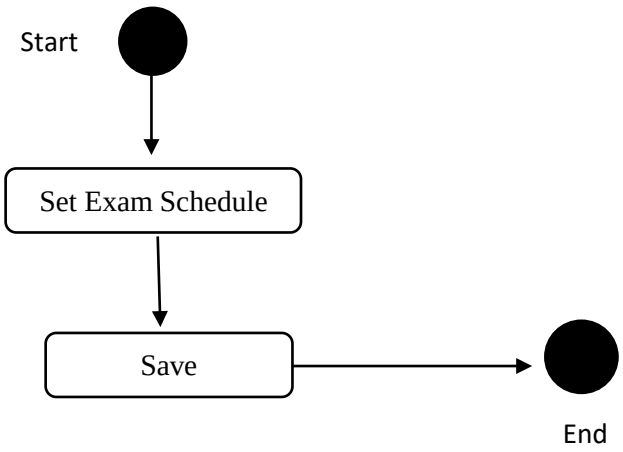


Figure 3.10: Exam Schedules

3.2.10 Teacher Submit Results

One of the main activities of the teacher is to submit the result of exam of students, when teacher login into his Dashboard he can see only his courses and student whom registered to this course and then he will submit the Exam results as soon as possible.

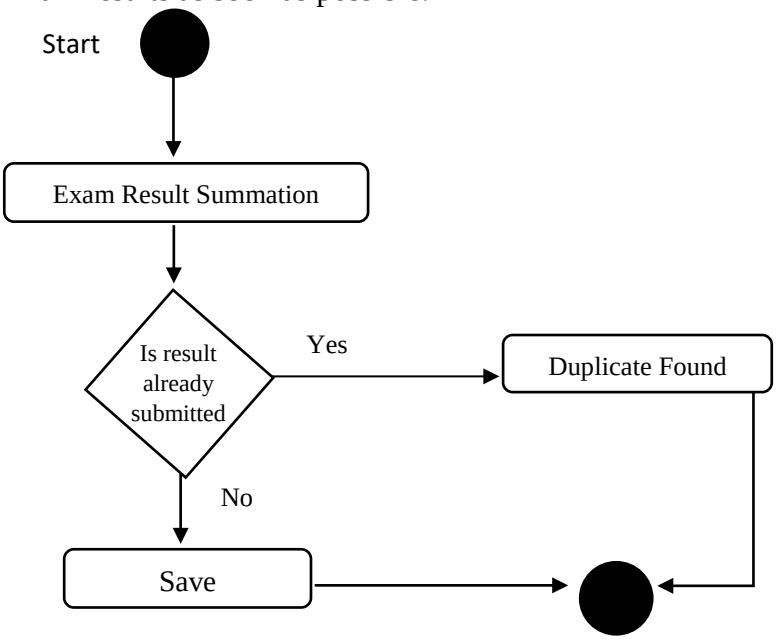


Figure 3.11: Teacher Submit Results

3.2.11 Teacher Take Class Attendance

teachers will take the Class attendance of the students in every class they attend, when the teacher login into his Dashboard he takes Attendance option he selects his course and see list of students and take attendance and tick the present student with **Present** and mark absent students as **Absents**.

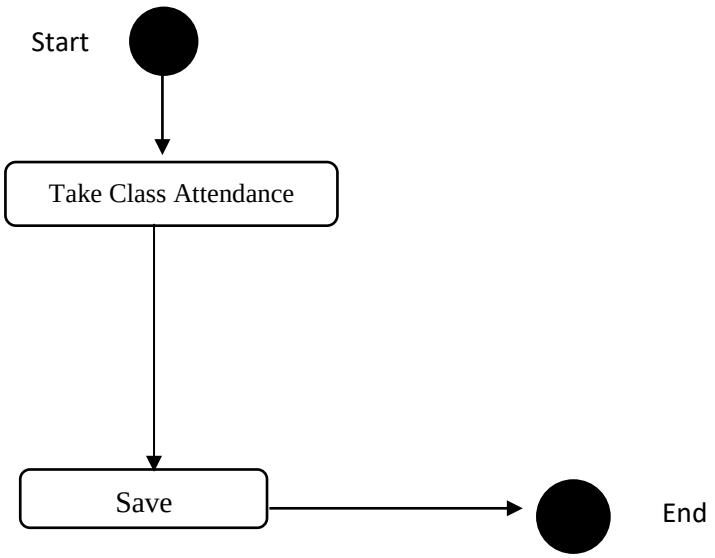


Figure 3.12: Teacher Take Class Attendance

3.2.12 Teacher view Student Attendance

Teacher can see their course class attendance report of students so that they can know which student normally absent.

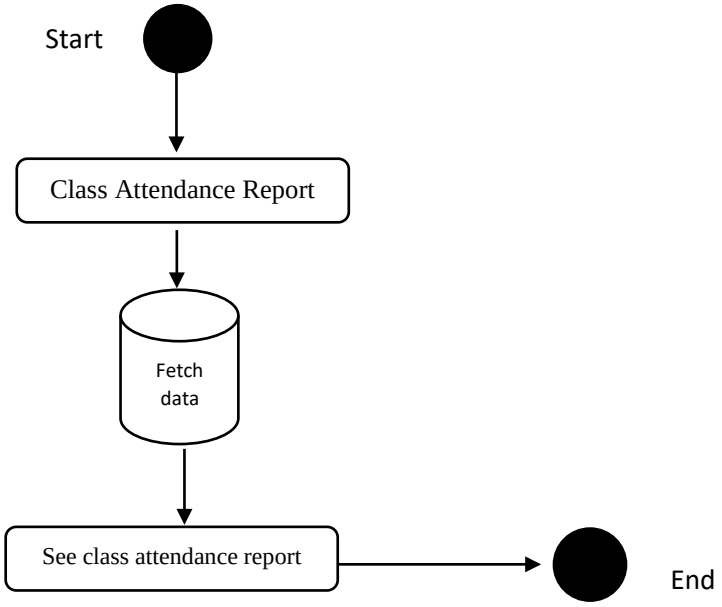


Figure 3.13: Teacher view Student Attendance

3.2.13 Student Login to Portal and See Results, Registered Course and Exam Schedule

Every student has unique ID No and password 12345 that he can login to Student Portal, after successful login he will see his exams Result, Registered Courses and next Exam schedule if posted.

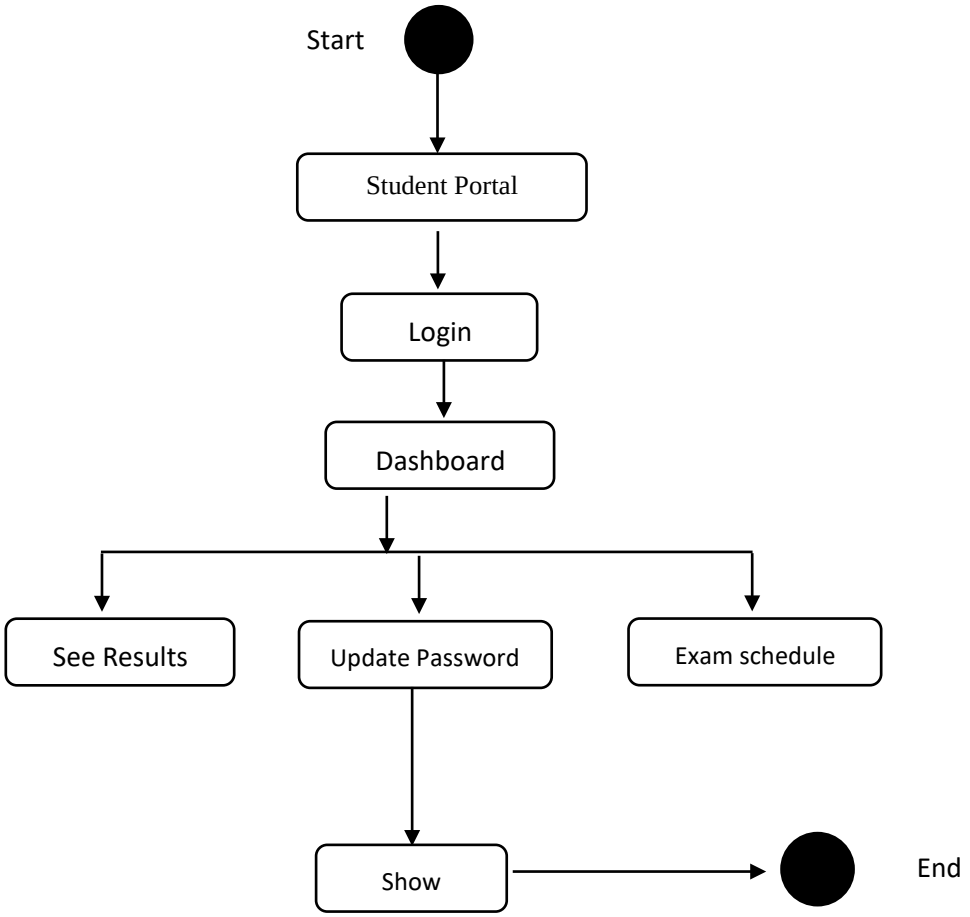


Figure 3.14: Student Login to Portal

3.3 Sequence Diagrams

Mainly sequence diagrams understand us how the data will be followed in any application. Now we are going to show some sequence diagrams.

3.3.1 User Registration

New user will be added in the system.

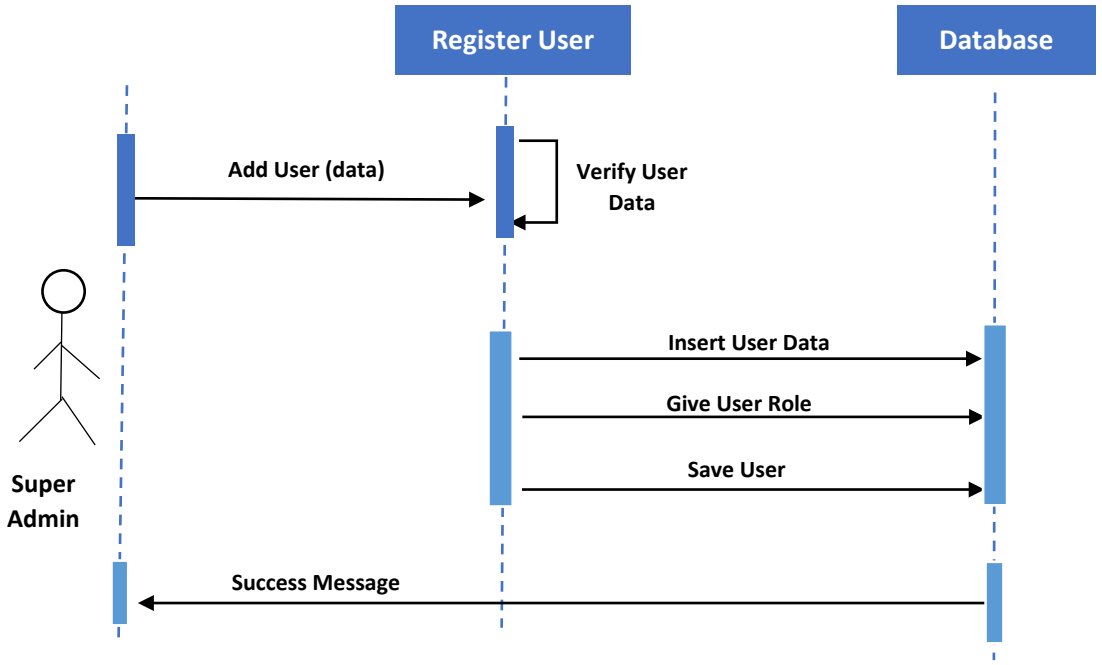


Figure 3.15: Sequence Diagram for User Registration

3.3.2 Faculty Registration

New Faculty will be added in the system.

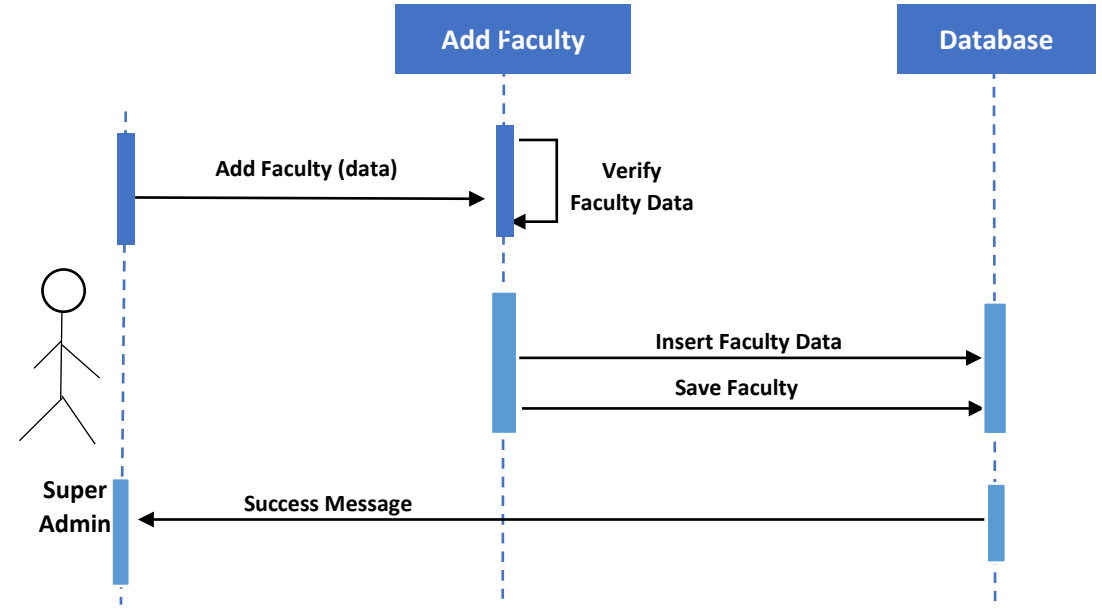


Figure 3.16: Sequence Diagram for Faculty Registration

3.3.3 Department Registration

New Department will be added in the system.

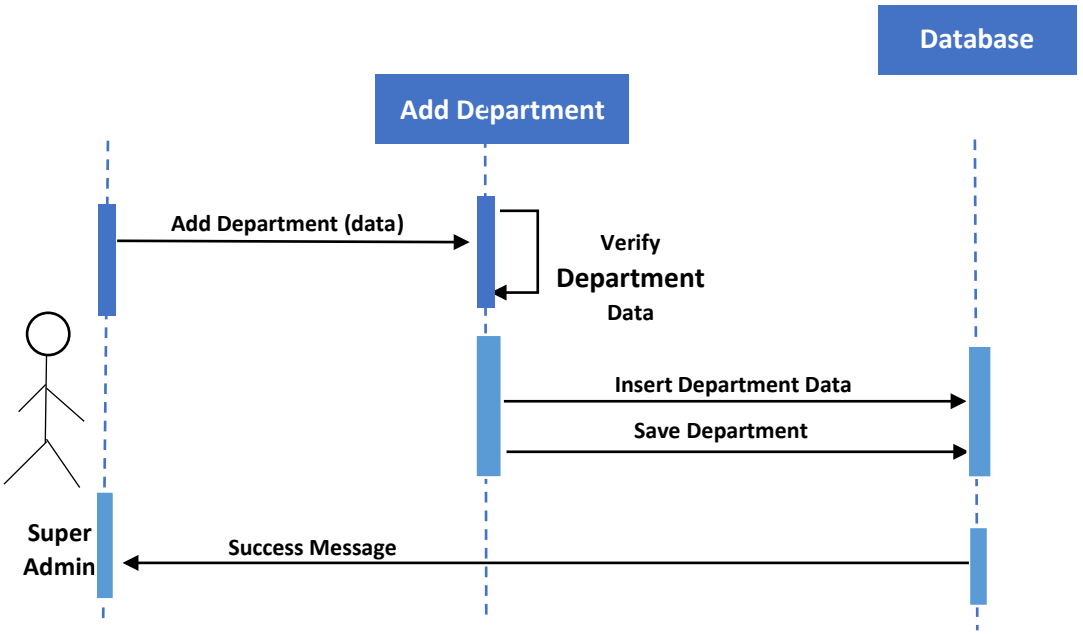


Figure 3.17: Sequence Diagram for Dept. Registration

3.3.4 Class Registration

New Class will be added in the system.

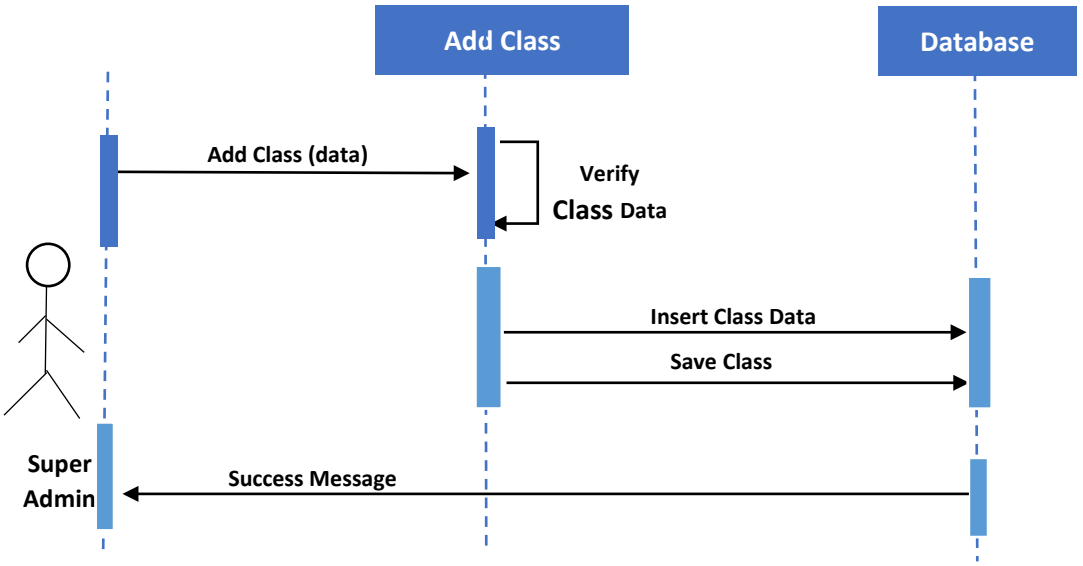


Figure 3.18: Sequence Diagram for Class Registration

3.3.5 Semester Registration

New Semester will be added in the system.

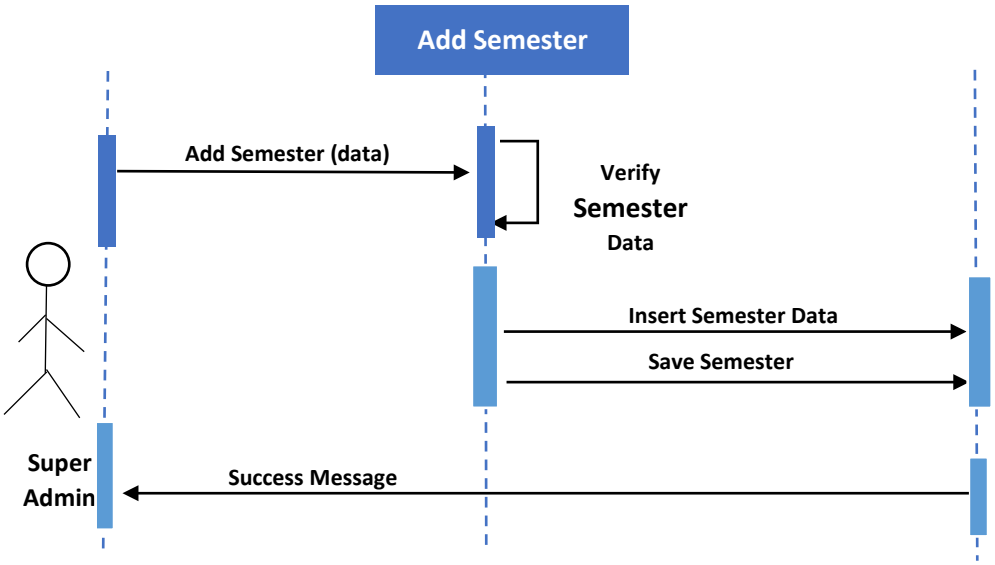


Figure 3.19: Sequence Diagram for Semester Registration

3.3.6 Course Registration

University will be added in the system.

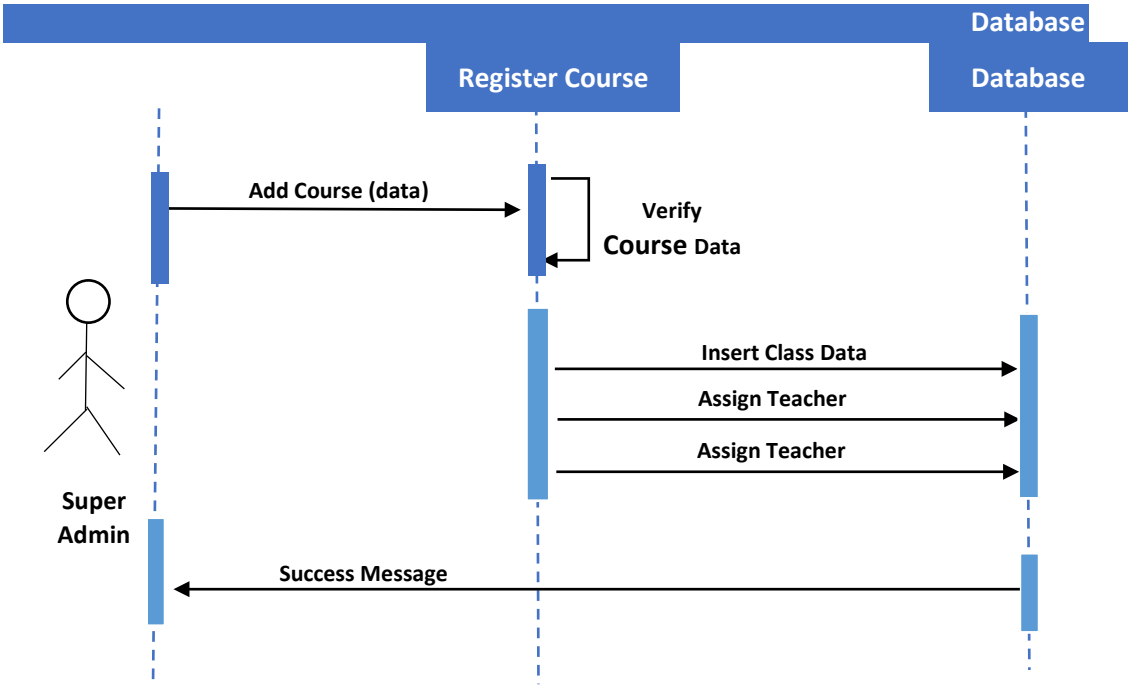


Figure 3.20 : Sequence Diagram for Course Registration

3.3.7 Teacher Registration

New Teacher will be added in the system.

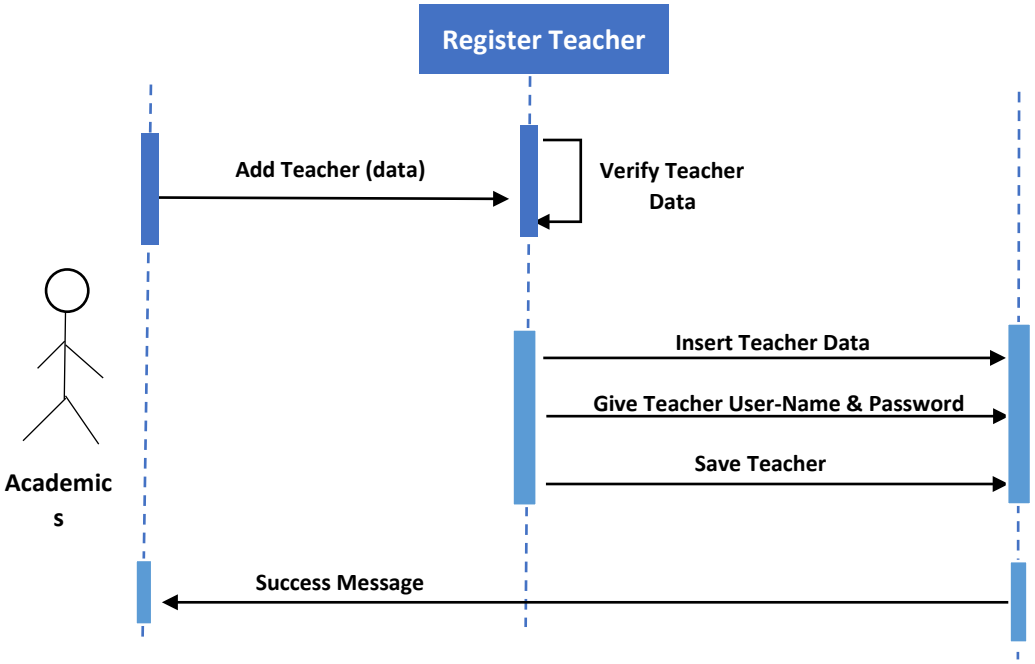


Figure 3.21: Sequence Diagram for Teacher Registration

3.3.8 Exam Schedule

Exam Schedule will be added in the system.

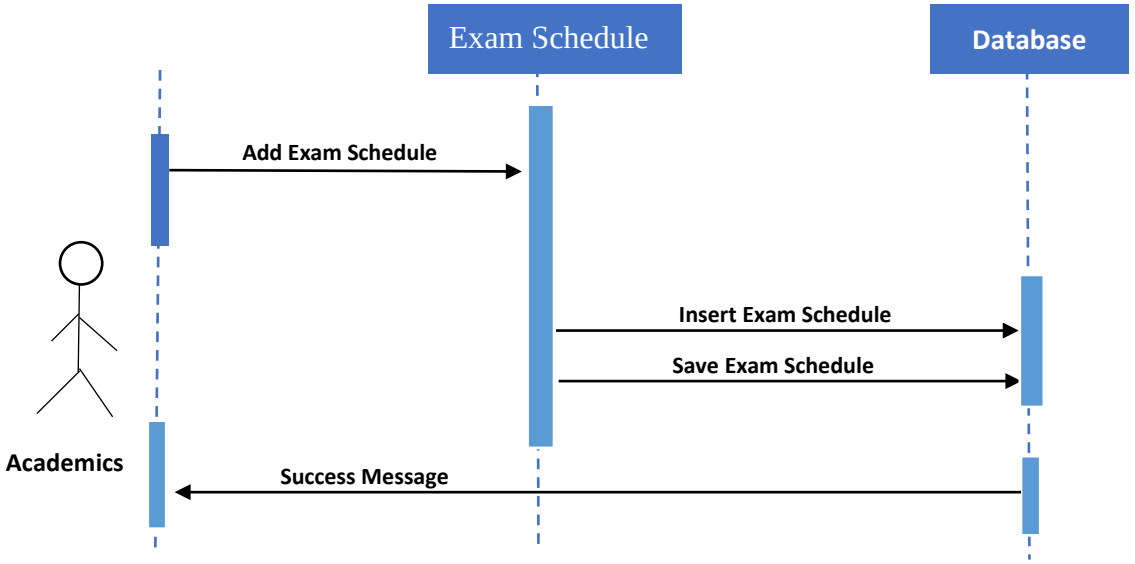


Figure 3.22: Sequence Diagram for Exam Schedule

3.3.9 Student Registration

New Student will be added in the system.

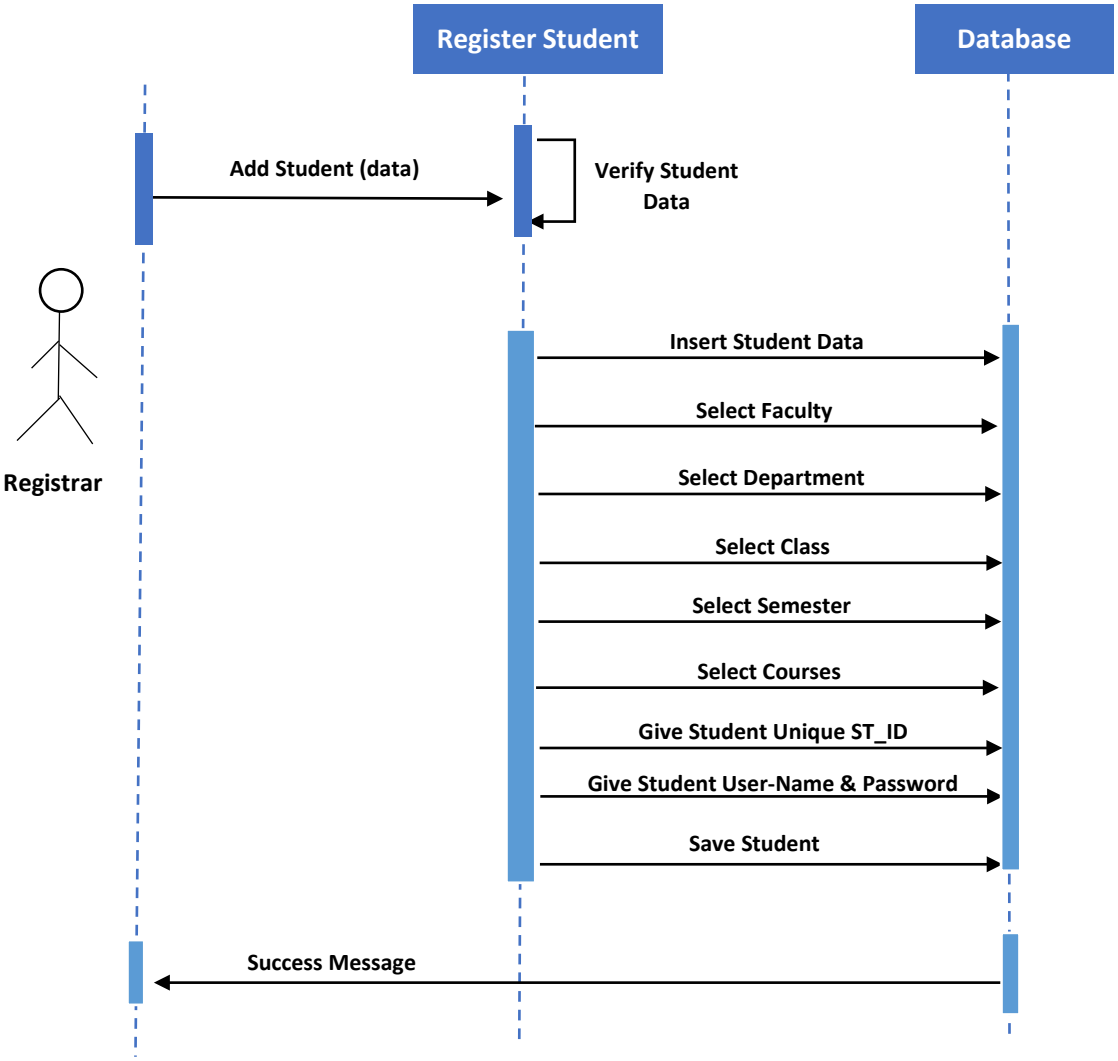


Figure 3.23: Sequence Diagram for Student Registration

3.3.10 Teacher Submit Exam Result

Teachers will submit Students Exam Results in the system.

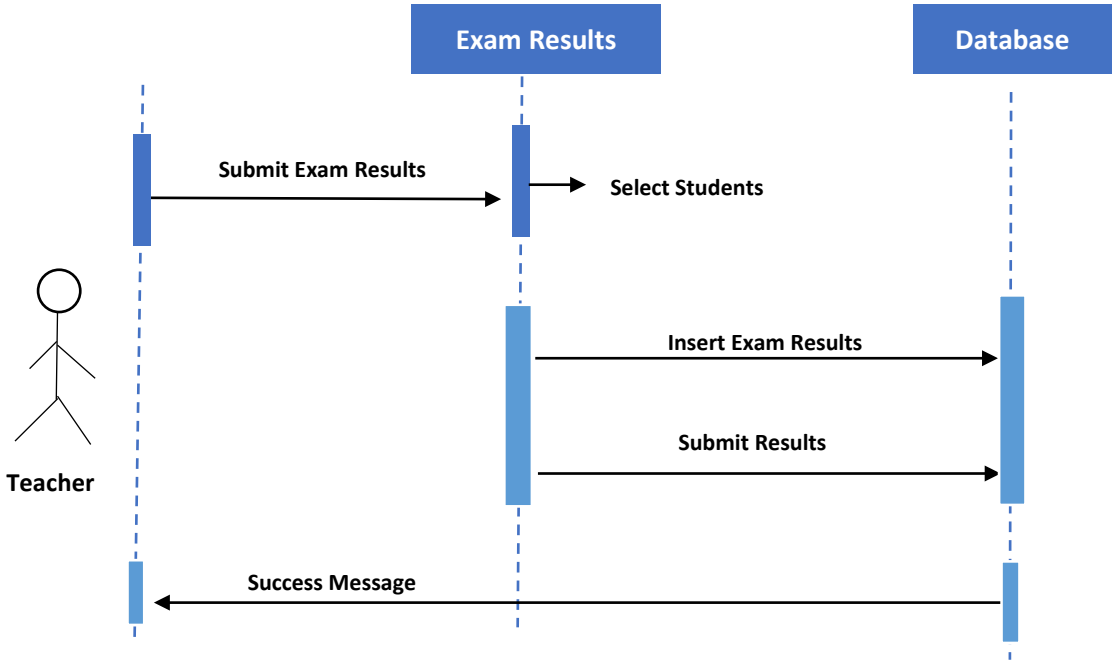


Figure 3.24: Sequence Diagram for Teacher Submit Exam Result

3.3.11 Teachers Take Class Attendance

Teachers will take class attendance in every class they had during the semester

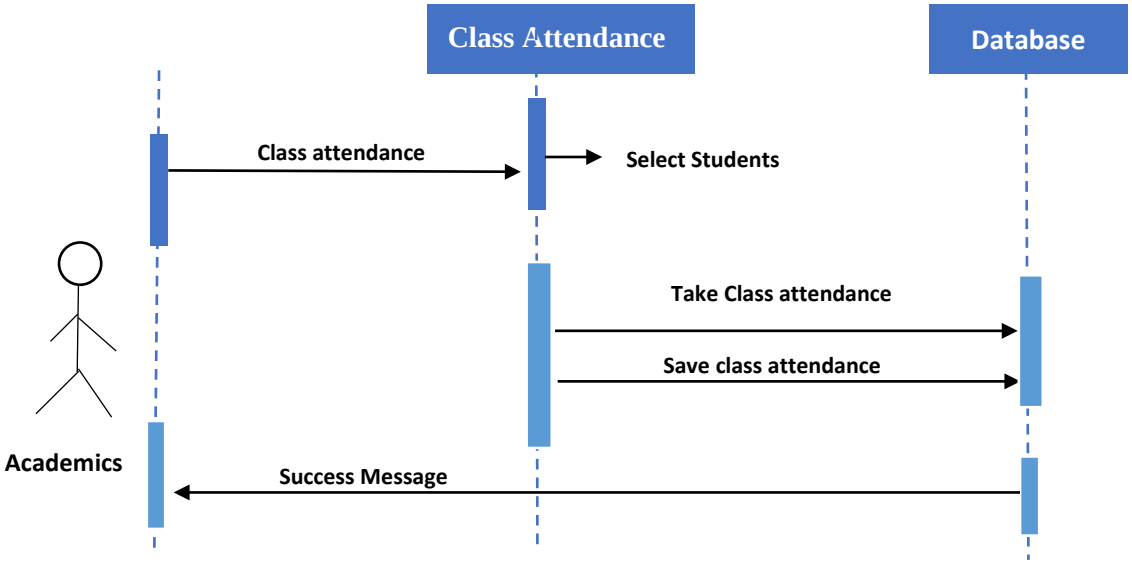


Figure 3.25: Sequence Diagram for Teachers Take Class Attendance

3.3.12 Student Portal Login

Students have portal so that they will login to see exam results and exam schedule.

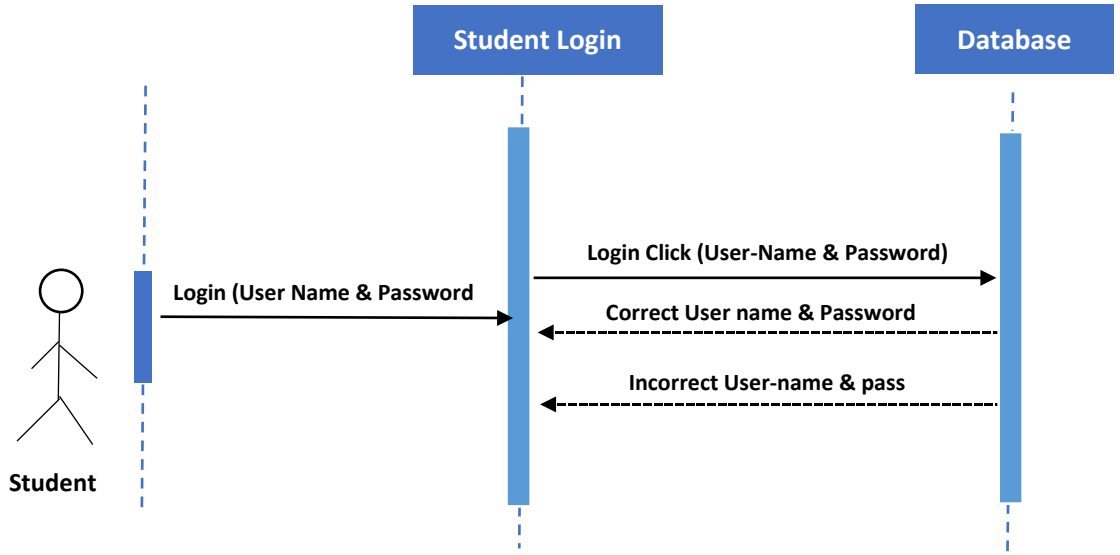


Figure 3.26: Sequence Diagram for Student Portal Login

3.3.13 Student See Exam Results and Exam schedule

When student login into his portal he will see the examination results also he can see the next examination schedule.

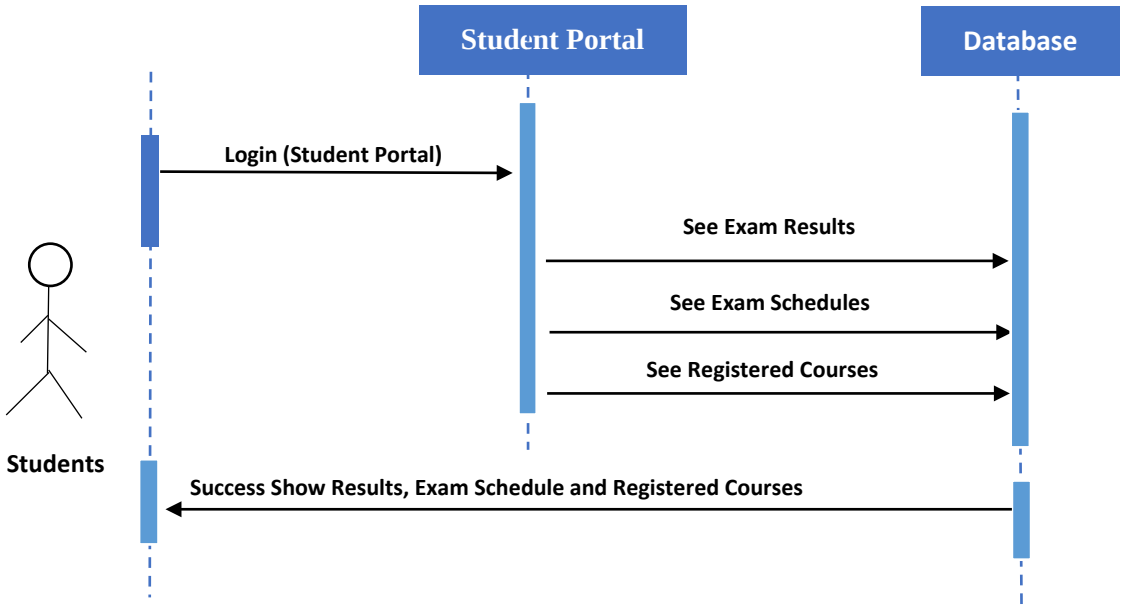


Figure 3.27: Sequence Diagram for Student See Exam Results and Exam schedule

Chapter 4
System Design Specification

4.1 Development tools and technology

Software development tools are used to develop software. There are different kinds of tools for software developers to make the process of software development very smooth. But now I will mention some of tools that I am using to develop my project.

4.1.1 User Interface Technology

First of all, after landing any application, interface of that software is appeared to the user. So, the importance of user interface is very high. For success of any software application, a good looking user interface plays a vital role. User interface includes using good image, graphics, typography, stylesheets, scripting etc.

4.1.1.1 CSS framework or Twitter Bootstrap

After completing markup, cascading style sheets are designed. It generally explains how HTML elements will display. There are three ways to write stylesheet. One internal CSS, another is external CSS. And last one is inline CSS. But most of time external CSS is used. Because, by using external CSS, all CSS data can be kept in different files Bootstrap also provides media query which features responsive layout for different devices with different screen size. Bootstrap provides a set of some files which contains stylesheets which gives basic definition.

Bootstrap also provides some JavaScript components also. There are some built in components like JQuery UI. By using Bootstrap framework, we get both CSS and JavaScript facilities with a single platform. But before start designing an application interface with Bootstrap, one may have some basic knowledge about this framework. It will increase the efficiency.

4.1.1.2 JQuery UI

JQuery UI is a name of library which uses JavaScript as core programming language. It simplifies codes of a programming language named JavaScript. Generally, it interacts with Graphical User Interface (GUI). It also provides visual effects with animation. It can also override cascading style sheets. It also provides AJAX functionality which stands for Asynchronous JavaScript and XML. It is also a subset of JavaScript. By using AJAX no page reloading is required. It is also compatible with any browser like Google Chrome, Mozilla Firefox, Opera, Safari, Internet Explorer etc.

4.1.1.3 Programming Language

For developing any application system minimum one programming language is essential. In my application, there are two different programming language is used. One is for front-end side. And another is for server side. The front-end language that I have used to my application is HTML, CSS and JavaScript. And the server-side programming language name is PHP Frame-work Called **Laravel** which is very popular framework, Both of them are open source general purpose scripting language.

4.1.2 Implemented tools and platform

As I have said before, there are some tools and technologies that need to be used for developing software. It is very important to determine which tools and platforms are the best match of my requirements. After making a proper decision, one need to start using them.

4.1.2.1 Integrated Development Environment

IDE stands for Integrated Development Environment. Programmers write code on IDE. After that IDE provide the feature to execute the source code. For developing my web application, I have used phpStromes which is powered by JetBrains company. It is a commercial IDE for cross platform environment. It is able to suggest code to the programmers also.

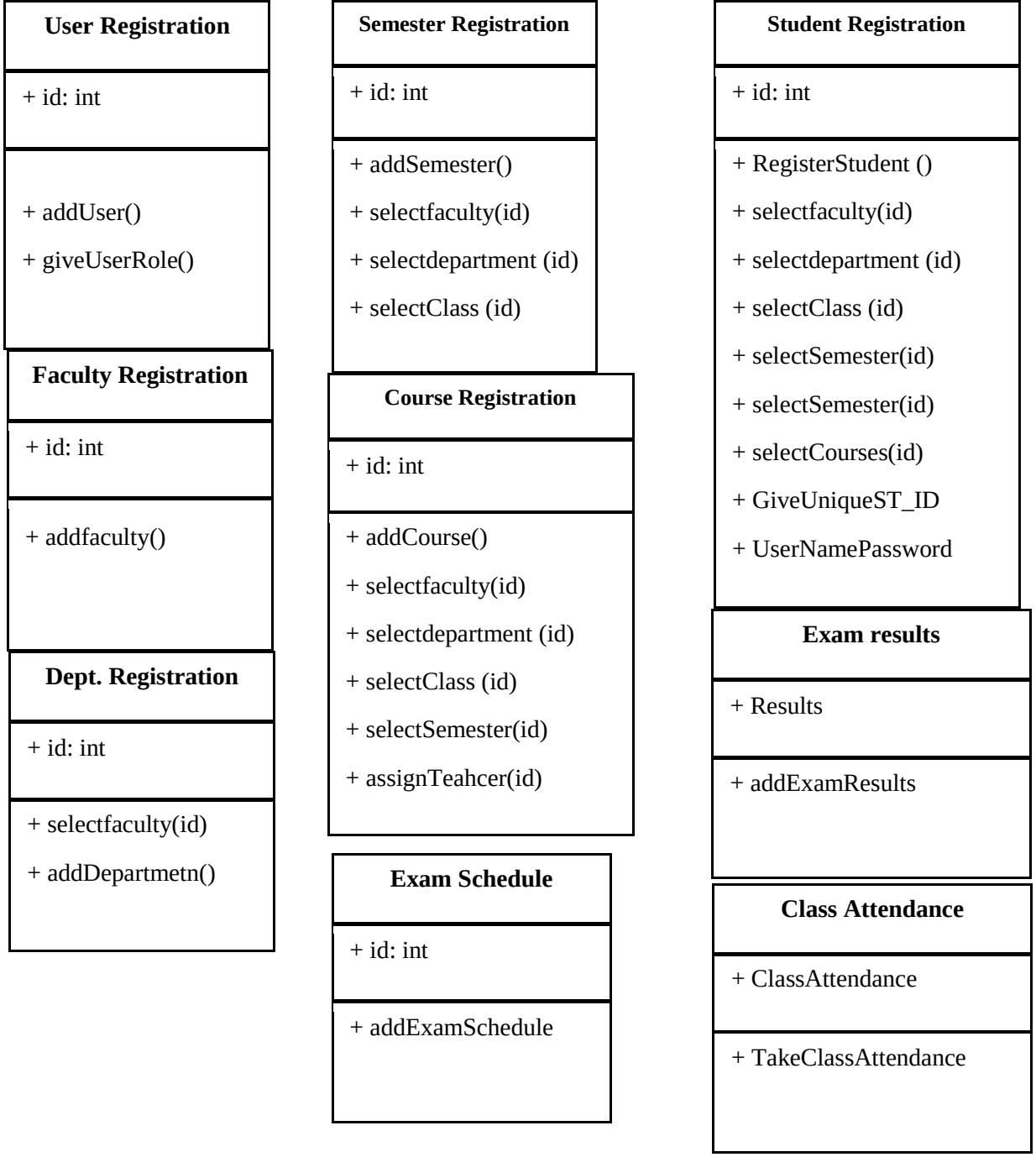
4.1.2.2 Web Server

We have used apache server. It is a free and open source software to use. It can be used on cross platform. It supports a wide range of features and most of them are already implemented as compiled modules. This module can extend the main features or core functionality.

4.1.2.3 Database Server

For developing our whole project, we have followed Relational Database Management System or RDBMS. And we find that MySql provides the feature of RDBMS. So we should not have any issue to use MySql database. It is also very easy to use. It can also ensure the security, scalability, high performance and many things.

4.2 Class Diagram


User Registration
+ id: int
+ addUser()
+ giveUserRole()
Semester Registration
+ id: int
+ addSemester()
+ selectfaculty(id)
+ selectdepartment (id)
+ selectClass (id)
Student Registration
+ id: int
+ RegisterStudent ()
+ selectfaculty(id)
+ selectdepartment (id)
+ selectClass (id)
+ selectSemester(id)
+ selectSemester(id)
+ selectCourses(id)
+ GiveUniqueST_ID
+ UserNamePassword
Faculty Registration
+ id: int
+ addfaculty()
Course Registration
+ id: int
+ addCourse()
+ selectfaculty(id)
+ selectdepartment (id)
+ selectClass (id)
+ selectSemester(id)
+ assignTeahcer(id)
Exam results
+ Results
+ addExamResults
Dept. Registration
+ id: int
+ selectfaculty(id)
+ addDepartmetn()
Exam Schedule
+ id: int
+ addExamSchedule
Class Attendance
+ ClassAttendance
+ TakeClassAttendance

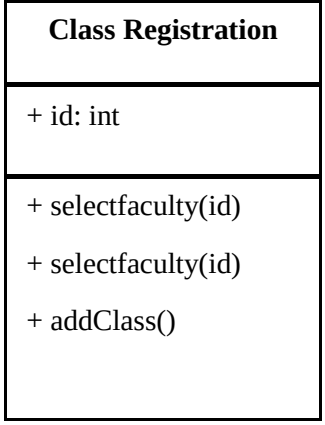


Figure 4.1 Class Diagram

4.3 Database Design Diagram

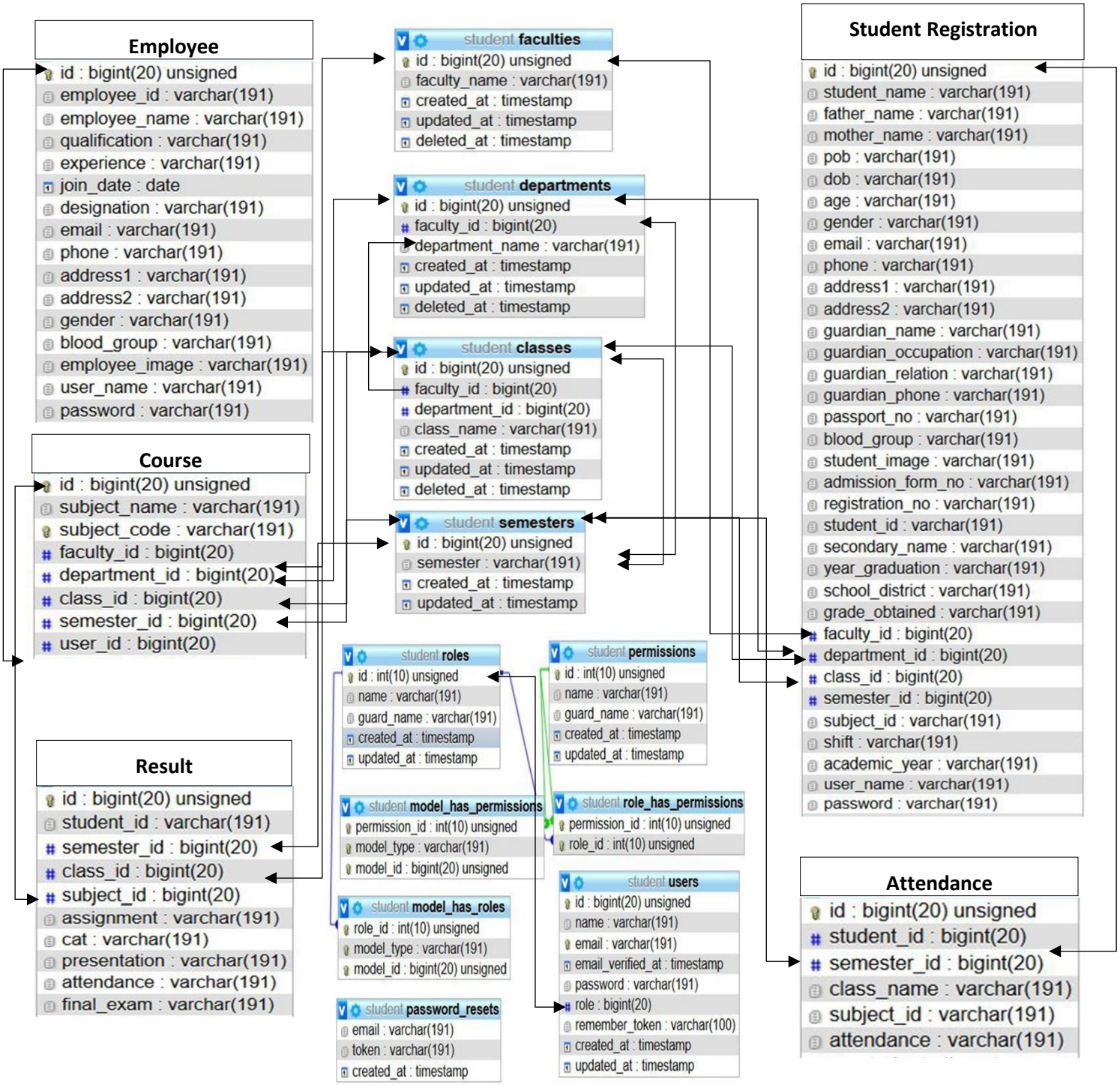


Figure 4.2: Database Diagram

4.4 ER Diagram

This ER (Entity Relationship) Diagram represents the model of University Management System Entity. The entity-relationship diagram of University Management System shows all the visual instrument of database tables and the relations between Students, Employees, Courses, Results, and Attendance. It used structure data and define the relationships between structured data group of University Management System functionalities. The main entities of the University Management System are Students, Employees, Courses, Results, and Attendance. **University Management System entities and their attributes**

- **Student Entity:** Attributes of Students are: Student_id, Student_name, Faculty_id, Depatment_id, Class_id, Semester_id, Email, Phone, student_image, ……ets, user_name, and password.
- **Employee Entity:** Attributes of Employee are: employee_id, employee_name, qualification, experience, join_date, designation, email, phone, addresses, gender, blood_group employee_image, user_name, password, role.
- **Course Entity:** Attributes of Course are: course_id, course_name, course_code, faculty_id, department_id, class_id, semester_id, teacher_id.
- **Attendance Entity:** Attributes of Attendance are: student_id, semester_id, class_name, subject_id, attdance.
- **Result Entity:** Attributes of Result are: student_id, semester_id, class_id, subject_id, assignment, Cat, presentation, attendance, final_exam.

ER Diagram

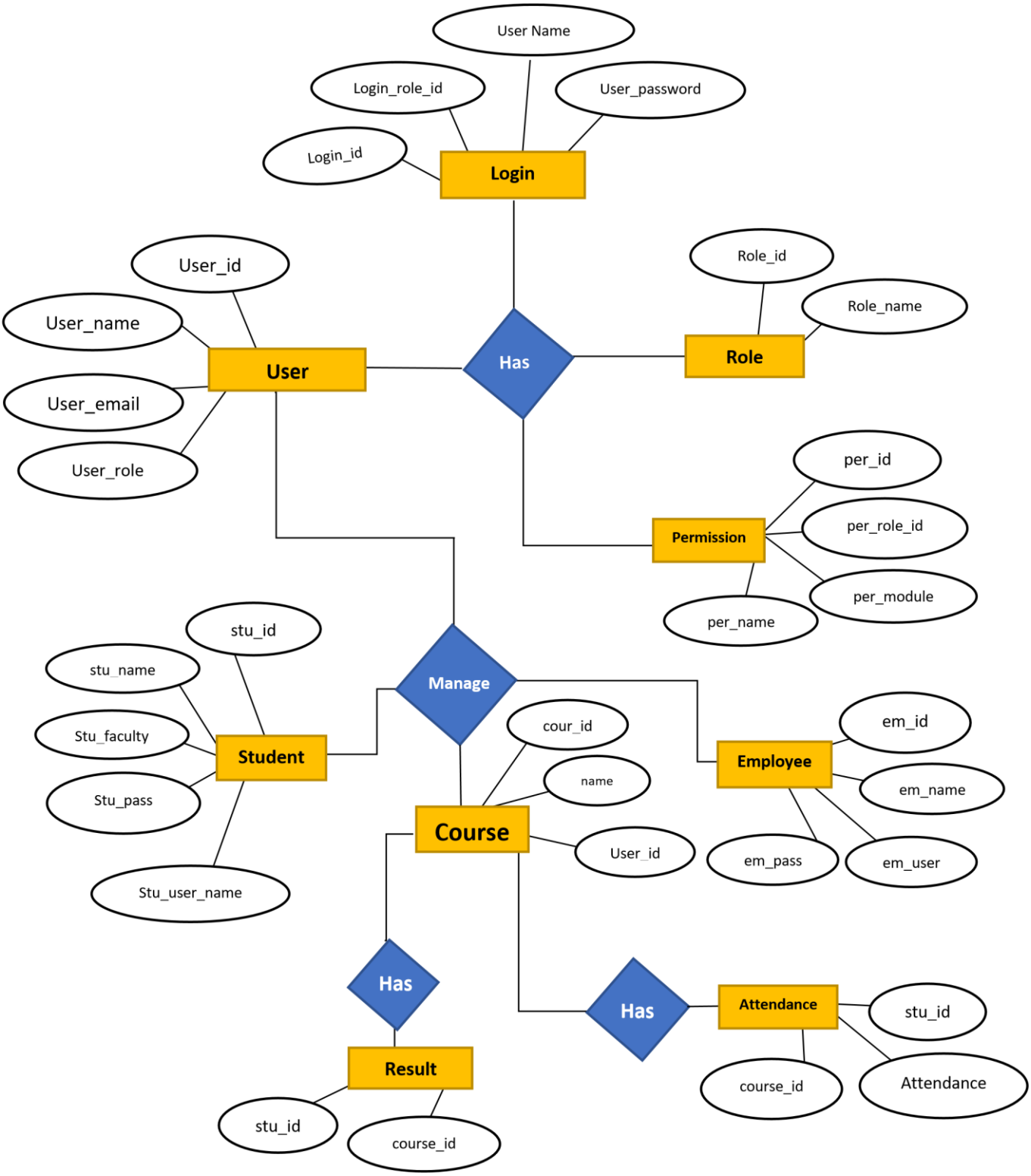

User Name
Login_role_id
User_password
Login_id
Login
User_id
User_name
User_email
User_role
User
Has
Role_id
Role_name
Role
per_id
per_role_id
per_module
per_name
Permission
stu_id
stu_name
Stu_faculty
Stu_pass
Stu_user_name
Student
Manage
cour_id
name
Course
User_id
em_id
em_name
em_pass
em_user
Employee
Has
Result
stu_id
course_id
Has
Attendance
stu_id
course_id
Attendance

Figure 4.3: ER Diagram

Chapter 5
System Test

5.1 Testing Features

Feature testing can be considered as making change to add or modify the new functionality to the existing project. To test the features and functionality, a new test set is to be written for testing purpose. Almost every feature and functionality have different characteristics. Those are designed to make the application more useful, intuitive, reliable, secured, scalable, effective and efficient.

5.1.1 Features to be tested

Features	**Priority**	**Description**
Login	1	User must be authenticated by login
Logout	1	Session must be destroyed after logout
Add Faculty	1	Faculty Data must be inserted Properly
Add Department	1	Department Data must be inserted Properly
Add Class	1	Class must be inserted Properly
Add Semester	1	Semester must be inserted Properly
Register Teacher	2	Teacher must be registered give them User-name and Password
Register Course & Assign Teacher for the Course	2	Course must be registered and assign into one teacher
Register Students	3	Student Registration data must be inserted successfully
Add Student Exam Result	3	Teacher will add student's exam results
Take Student Class Attendance	3	Teacher take student class attendance
Students See their Exams Results	3	Every student has to see his exam results from is portal
Technological Features		
Database	1	Database will be used at almost every operation. So, this is why, this part must be controlled tightly.

Here, 1 = Low Priority, 2 = Medium Priority, 3 = High Prior

5.2 Testing Strategy

Testing strategy is to be considered as a general purpose of testing process. Testing object, testing function methods, total available resources are included to the testing strategy. It is also the indicator of test levels that are to be performed on the whole software development life cycle. Those strategies that are prepared by the quality assurance team should be reviewed by the developers of the application. After that it should be reviewed by the test team leads. Different kinds of testing strategies can be performed according to the type of application system that need to be tested.

5.2.1 Pass / Fail Criteria

Pass or fail criteria will be set by the test engineers. They will prepare the pass / fail criteria on the basis of which input data are worked and which are not works well. Those data that are worked well will considered as pass criteria. And rest of inputs data will be considered as fail criteria.

Now I will give the pass / fail criteria below.

- System crash will not be considered as pass case.
- If any criteria pass 100% times, then it will be considered as pass criteria only.
- If data can't be displayed to the application properly, then it is also to be considered as fail criteria.

5.3 Testing Schedule

Test Phase	**Time**
Testing plan create	1 week
Test specification	2 week
Unit testing	During development time
Component test	1 week
Test Phase	**Time**
Integration testing	1 week
Validating use cases	1 week
Testing user interfaces	1 week
Load testing	1 week
Performance testing	1 week
Release to production	1 week

5.4 Trace Ability Matrix

Project Manager			**Business Analyst Lead**		
QA Lead			**Target Implementation Date**		
BR#	**Category / Functionality / Activity**	**Requirement Description**	**Use Case Reference**	**Test Case Reference**	**Comments**
BR-1	Functional	Add User	Use case 3.1.1	Test case 5.6.2	
BR-2	Functional	Add Faculty	Use case 3.1.2	Test case 5.6.3	
BR-3	Functional	Add Department	Use case 3.1.3	Test case 5.6.4	
BR-4	Functional	Add Class	Use case 3.1.3	Test case 5.6.5	
BR-5	Functional	Add Semester	Use case 3.1.5	Test case 5.6.6	
BR-6	Functional	Add Teacher	Use case 3.1.7	Test case 5.6.7	
BR-7	Functional	Register Course	Use case 3.1.6	Test case 5.6.8	
BR-8	Functional	Student Registration	Use case 3.1.9	Test case 5.6.9	
BR-10	Functional	Add Exam Results	Use case 3.1.10	Test case 5.6.10	
BR-11	Functional	Take Class Attendance	Use case 3.1.11	Test case 5.6.11	
BR-12	Functional	View Exam Results	Use case 3.1.14	Test case 5.6.11	

5.5 Testing Environment

Testing environment means to prepare the environment with hardware and software so that test engineers can be able to execute test cases as required. Besides hardware and software usage, network configuration might be needed to execute test plans.

For making the environment for testing, some key area needs to setup. Those are:

- Test data
- Database server
- Client's operating system
- Front end running environment
- Browser
- System and application
- Network
- Documentation is also required. Like: user manuals, installation guides, configuration guides, documents etc.

5.6 Test Cases

A test case refers to some rules and regulations or conditions by which it can be determined whether a system can be able to meet the works or requirements under test cases properly. We know, there is a chance to have some fault or break holes in any application. This is very common scenario. And those issues are solved usually by software testing approaches. But if we don't care of those issues, then the full application development might be ruined. So proper testing must be needed. For testing our application, I have prepared some test cases. Now, I am going to provide them.

5.6.1 Log In

Test case #1			**Test case name:** Log in		
System: University Management System					
Designed by: Abdikani Mohamed Ai			**Designed date:** 11-Oct-2019		
Executed by:			**Executed date:**		
Short description: The registered users and students need to login to the application. And before that our application will check the authentication and authorization.					
Pre-conditions: • Users are always redirected to the login page whether they have not authenticated by our application while entering into the dashboard. • Assume that, the user-name is 'admin@admin.com and password is '12345'					
Step	**Phone**	**Password**	**Expected result**	**Pass/Fail**	**Comment**
1	Admin@hotamil.com	421	Invalid		

2	admin		Password can’t be blank		
3		432	Mobile can’t be blank		
4	Register Course	12345	Successfully login to the application and redirected to the dashboard		
Post-conditions: Users including Admin, Teacher and Students will successfully login to the application.					

Add

5.6.2 User

Test case #2	**Test case name:** Add User			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: System Super Admin can create the other admins and manage the users.				
Pre-conditions: • System Super Admin he must create the other users • It will give them user name and password • It will assign for them role				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	Insert user data is not provided	Application will Denay if some filed are missing or duplicate		
2	Insert user data provided	User will be created successfully		
Post-conditions: After creation the user, these users can able to use the system directly with the permission that you gave him.				

5.6.3 Faculty

Test case #3	**Test case name:** Add Faculty			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: one of the first activities is to create all the available faculties in the university				
Pre-conditions: • User must be authenticated by our application. • Insert full name of the faculty				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**

Add

1	All data is not provided	If the faculties data is not filled the system will return please fill.		
2	Insert Existing data	If you insert existing faculty data then the application will deny the duplicate data		
2	Data provided properly	Faculty saved successfully		
Post-conditions: After adding faculty, the faculty will appear in all other forms needs to be fill				

5.6.4 Department

Test case #4	**Test case name:** Add Department			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: is to create all the available departments of the university				
Pre-conditions: • User must be authenticated by our application when he creating the dept. • Insert full name of the Department. • Choose the department's Faculty.				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	All data is not provided	If the Dept. data is not filled the system will return please fill		
2	Insert Existing data	If you insert existing Department data then the application		

Add

		will deny the duplicate data		
2	Data provided properly	Faculty saved successfully		
Post-conditions: After inserted departments, the departments will appear in all other forms needs to be fill				

5.6.5 Class

Test case #5		**Test case name:** Add Class		
System: University Management System				
Designed by: Abdikani Mohamed Ai		**Designed date:** 11-Oct-2019		
Executed by:		**Executed date:**		
Short description: classes are the university classes that teacher uses normally and students of each class they will have their own class				
Pre-conditions: • User must be authenticated by our application when he creating the class. • Insert Class name as Short with No • Choose the department and faculty of this class.				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	All data is not provided	If the class. data is not filled the system will return please fill		

Add

2	Insert Existing data	If you insert existing Class data then the application will deny the duplicate data		
2	Data provided properly	Faculty saved successfully		
Post-conditions: After inserted classes, the classes will appear in all other forms needs to be fill				

5.6.6 Semester

Test case #6		**Test case name:** Add Class		
System: University Management System				
Designed by: Abdikani Mohamed Ai		**Designed date:** 11-Oct-2019		
Executed by:		**Executed date:**		
Short description: as normally university works semesterly so that in every you need to add the new semester into system so that all classes and courser and student will register again with new semester.				
Pre-conditions: • User must be authenticated by our application when he creating the class. • Insert Semester name as Short with year				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**

Add

1	Semester data added	System will save the semester		
2	Semester date did not add	The system will not save any semester		
Post-conditions: After creating semester, it will be whole identification of the system like if you want recognize course you check course semester and also if you do student you check by semester and results will be submitted as semesterly.				

5.6.7 Register Teacher

Test case #7	**Test case name:** Register Teacher			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: classes are the university classes that teacher uses normally and students of each class they will have their own class				
Pre-conditions: • User must be authenticated by our application when he creating the class. • Register new teacher to the system • Give teacher user-name and password				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	All data is not provided	If the teacher. data is not filled fully the system will return please fill		
2	Insert Existing data	If you insert existing teacher email data then the system will deny the duplicate data		
2	Data provided properly	Teacher registered successfully		
Post-conditions: After teacher registration, the teacher will be responsible to add exam results and take class attendance for his own courses.				

5.6.8 Add Course

Test case #8	**Test case name:** Add Course			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: classes are the university classes that teacher uses normally and students of each class they will have their own class				
Pre-conditions: • User must be authenticated by our application when he creating the class. • Add new course • Assign course to specified teacher				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	All data is not provided	If the course. data is not filled fully the system will return please fill		
2	Insert Existing data	If you insert existing course with same course code data then the system will deny the duplicate data		
2	Data provided properly	Course saved successfully		
Post-conditions: After you register course, the registered course will be later assigned by the students so that when the teacher is going to take his own course attendance, he will see student who registered to this course, same as when he will submit exam results.				

5.6.9 Student Registration

Test case #9	**Test case name:** Student Registration			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: Student registration Is the main activity of this system when you register the students is to select their Faculty, Department, Class, Semester and Choose course that they will register on				
Pre-conditions: • User must be authenticated by our application when he creating the class. • Register new students • Select Faculty, Department, Class, Semester and select courses that they registered • Give student user-name and password				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	All data is not provided	registered. data is not filled fully the system will return please fill the remaining or this filled is required		
2	Insert Existing data (Duplicate)	If you insert existing Student ID data then the system will deny the duplicate Student Data		
2	Data provided properly and selected all the options	Student Registered successfully		
Post-conditions: After student registration, the registered students can see their exam results from their own portal, and also teacher will see the students when they are taking class attendance or submiting the exam results.				

5.6.10 Add Exam results

Test case #10	**Test case name:** Add Exam results			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: Teacher will submit the exam results at the end of each semester and upload it into the system so that students can see their results.				
Pre-conditions: • User must be authenticated by our application. • Teachers have to submit student exam results by course s				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	Students registered for the Course	Are students registered for that course available		
2	Teacher submit student results data	Data will be saved into data base successfully		
Post-conditions: After teachers upload the results, students are able to see their results from their portal Online				

5.6.11 Take class attendance

Test case #11	**Test case name:** Take class attendance			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: Teacher will take student class attendance in every class he will take				
Pre-conditions: • User must be authenticated by our application. • Teachers will take the student class attendance for his course registered student				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	Students registered for the Course	Are students registered for that course available		

2	Teacher Update student attendance	Teacher will update student class attendance every day		
Post-conditions: after teachers takes the class attendance, the attendance will convert into marks so that the teacher will generate these marks into student results mark as attendance marks				

5.6.12 Students see Exam results

Test case #12	**Test case name:** Students see exam results			
System: University Management System				
Designed by: Abdikani Mohamed Ai	**Designed date:** 11-Oct-2019			
Executed by:	**Executed date:**			
Short description: students when they login to their Portal they are able to see exam results for the semesters				
Pre-conditions: • User must be authenticated by our application. • Students see Exam results from the portal				
Step	**Action**	**Response**	**Pass/Fail**	**Comment**
1	Exam results for the last semester are available	Students can see exam results for last semester from their portal		

2	Exam results for the last semester are not available	Students can't see exam results for last semester from their portal that means teacher not uploaded yet		
Post-conditions: After student login into their portal, students see their results this one of the main objectives of this system, this will make student happy and every student see his results individually, that means they cannot see any other student's results and also students will get their results as soon as possible after every semester ends.				

Chapter 6
User Manual

6.1 First Page

User when he hit the URL of out University Management System (UMS) he will this screen so that if he staffs of the university, he will choose staff login and also if he is Student then he will choose student Portal to login to our Application. Now I will provide the screenshot of login page below.

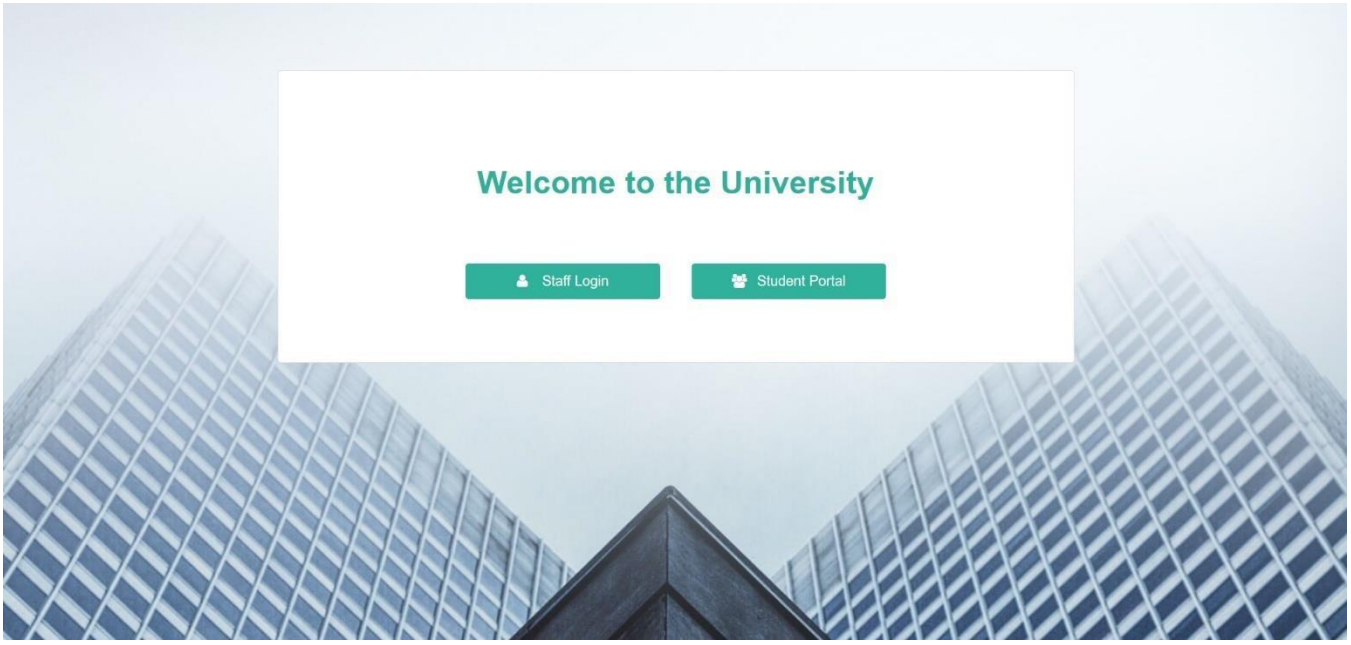
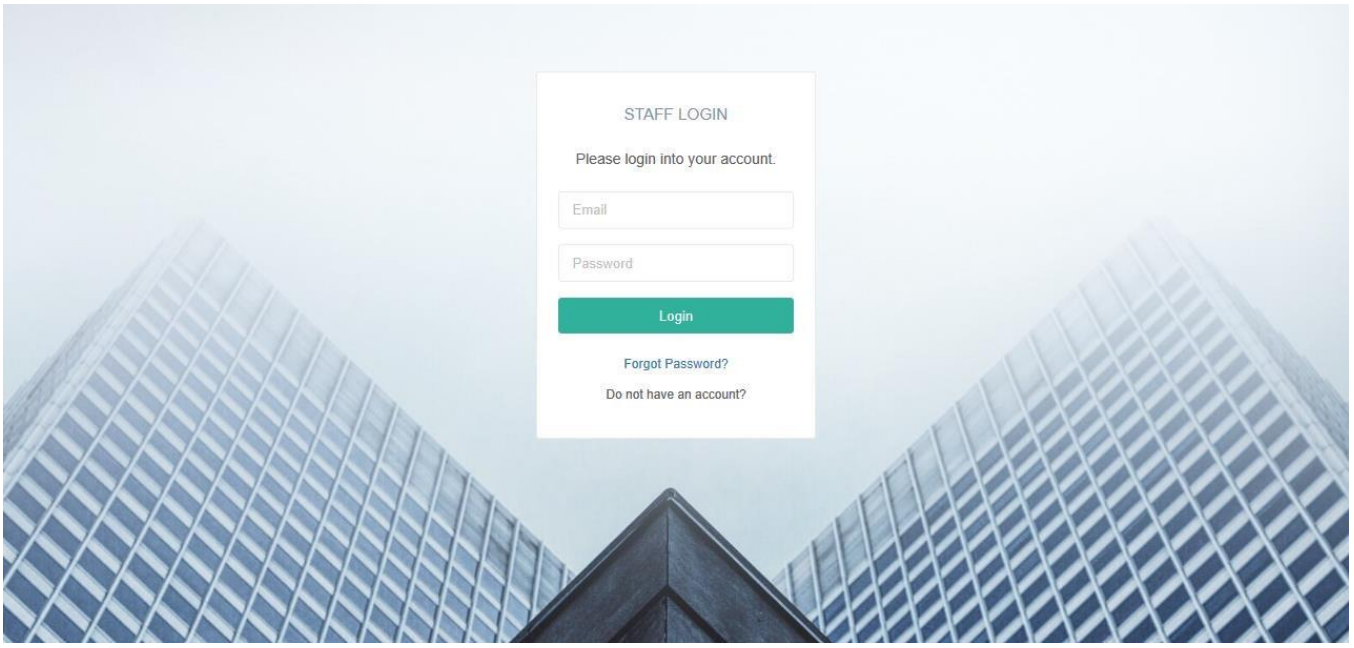


Figure 6.1: First Page

6.2 Staff Login Page

If you are staff User you need to login to our application for using this application. But before that they need to be authenticated by our application. First of all, they need to enter into the login page and fill up the required information. After that they will get access. Now I will provide the screenshot of staff login page below.



6.2 Staff Login

6.3 Register Staff Page They staff of the University it can be Academic, Registers…. Teacher so that they need to be registered by the Admin so that they can access the system and do what they shall do. Now I will provide the screenshot of staff Registration page below.

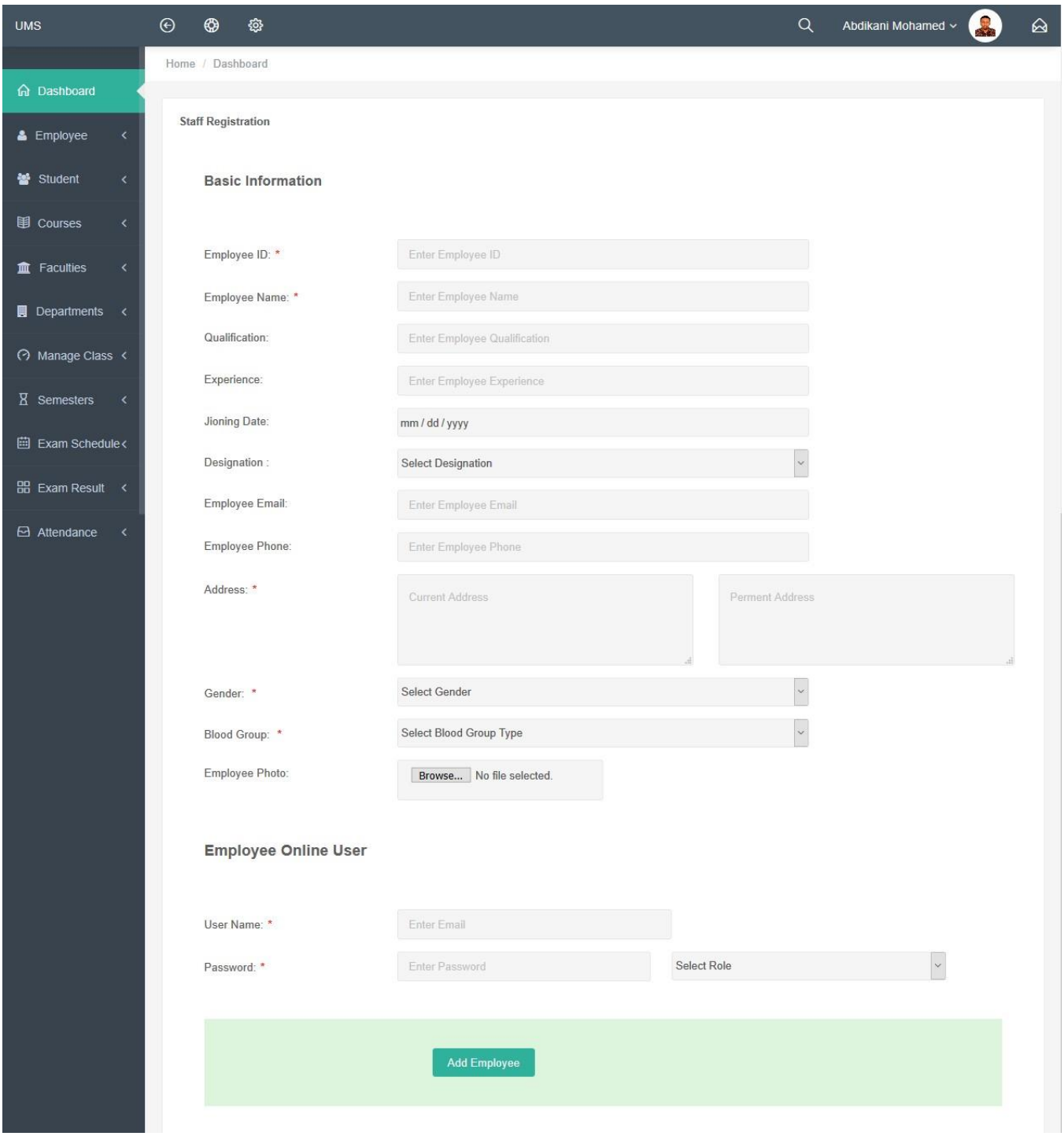


Figure 6.1: Register Staff

6.4 View Registered Staff

When you create university staff then you can manage them by going to staff list page and view, update or edit any staff you need. Now I will provide the screenshot of List of staff Registration page below.

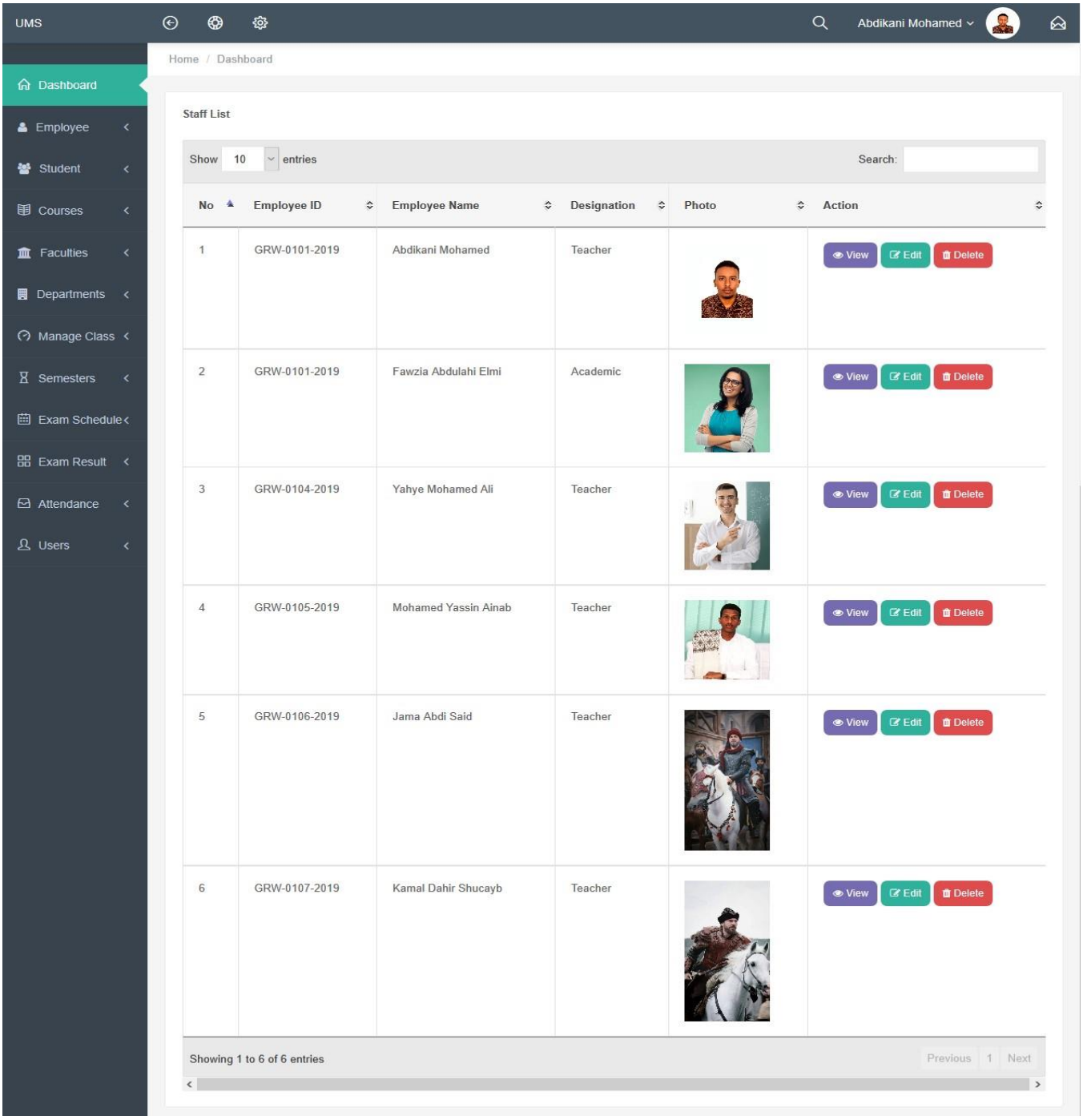


Figure 6.4: Registered Staff View

6.5 Add Faculty

After you login as system admin then you can create university faculties and update previous one, so that I will provide Screenshot for Registering new faculty.

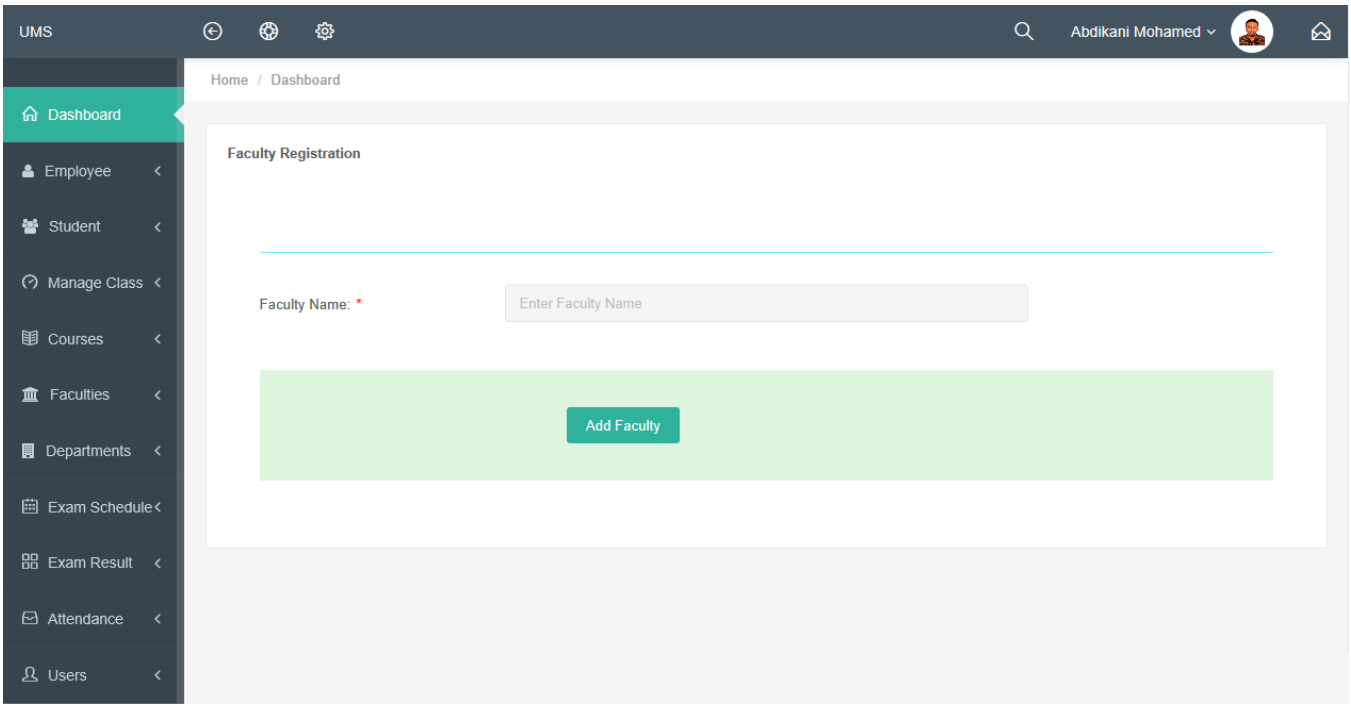


Figure 6.5: Add Faculty

6.6 View Registered Faculties

After you create faculties you can see list of available faculties in your system then you can delete them or Update them, Now I will provide the screenshot of Registered faculties List page below.

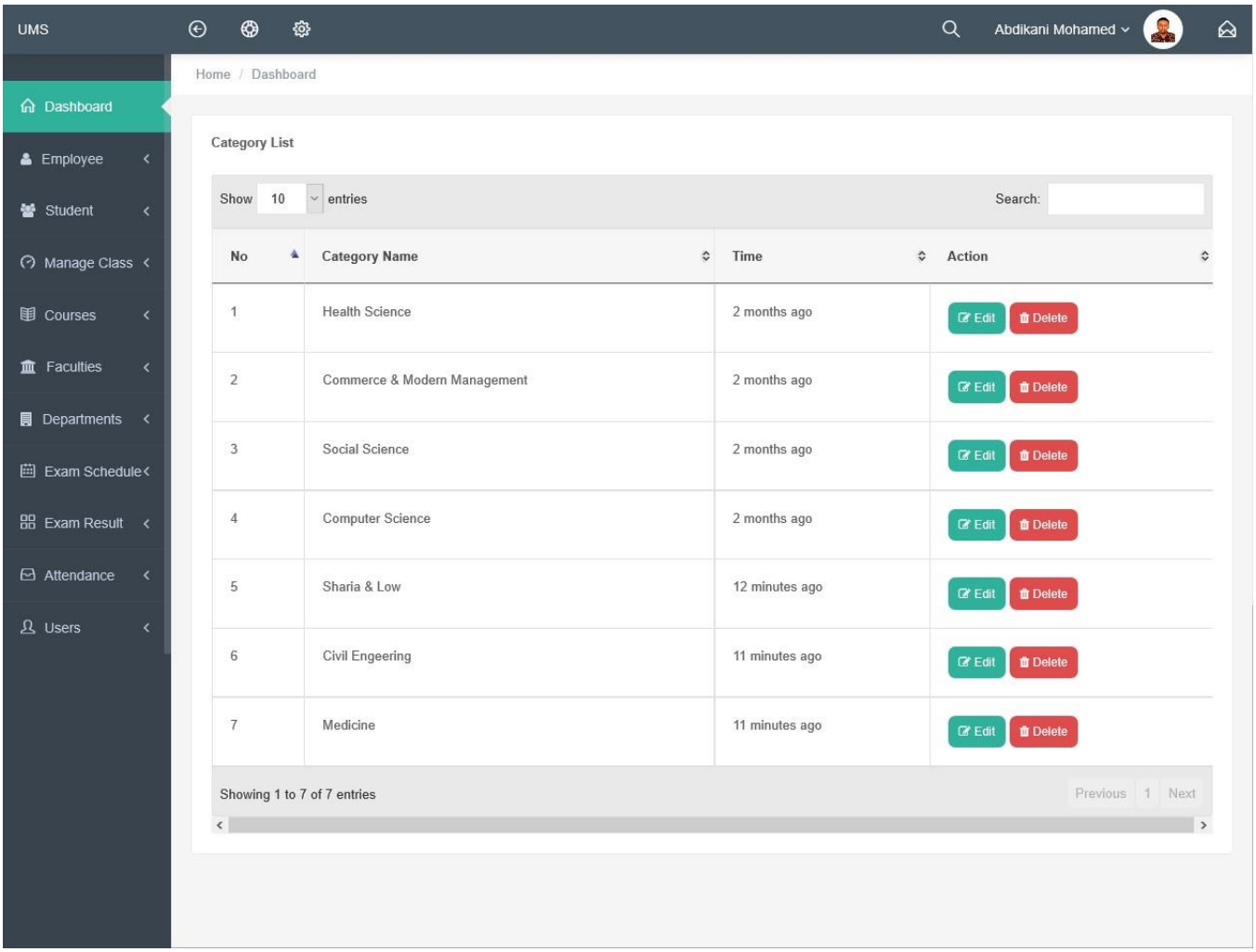


Figure 6.6: View Registered Faculties

6.7 Add Department

When you have created faculties now you need to create faculties Department and select every department her faculty, so that I will provide Screenshot for adding new Department.

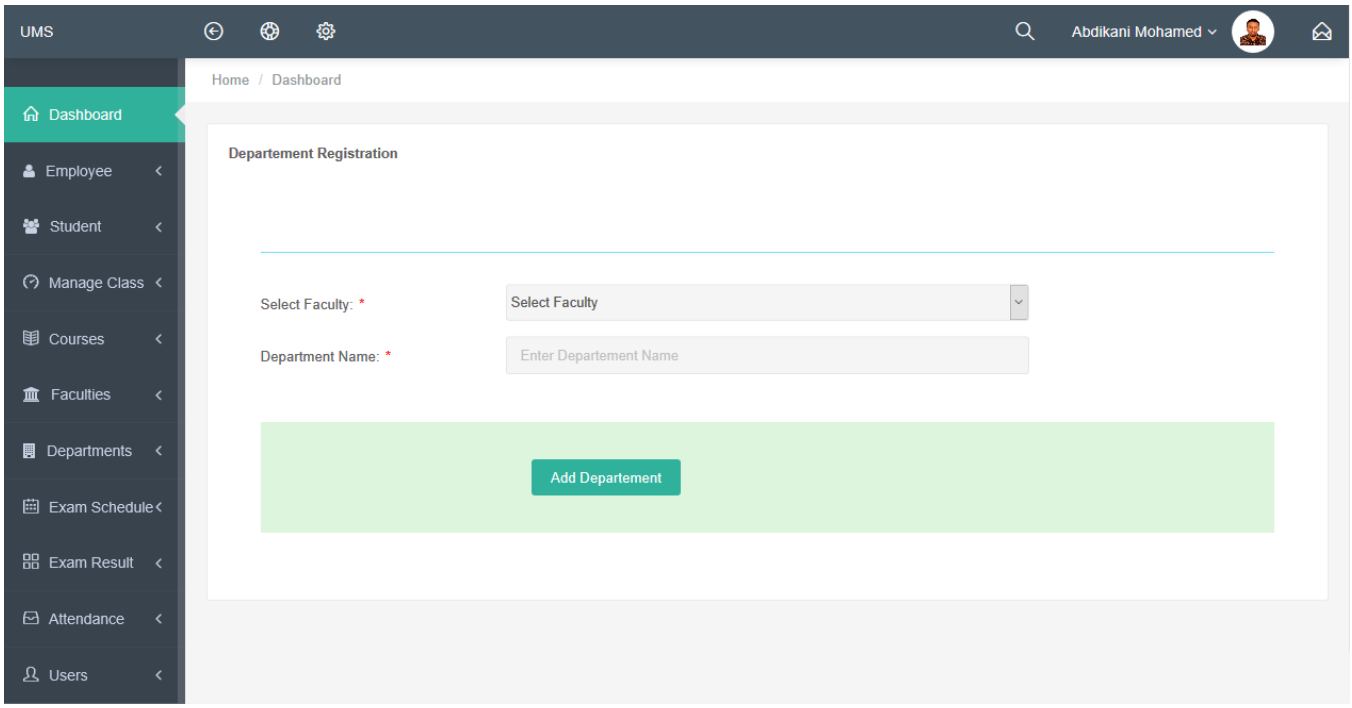


Figure 6.7: Add Department

6.8 View Departments

After you create department you can see list of available Department's in your system then you can delete them or Update them, Now I will provide the screenshot of Department List page below.

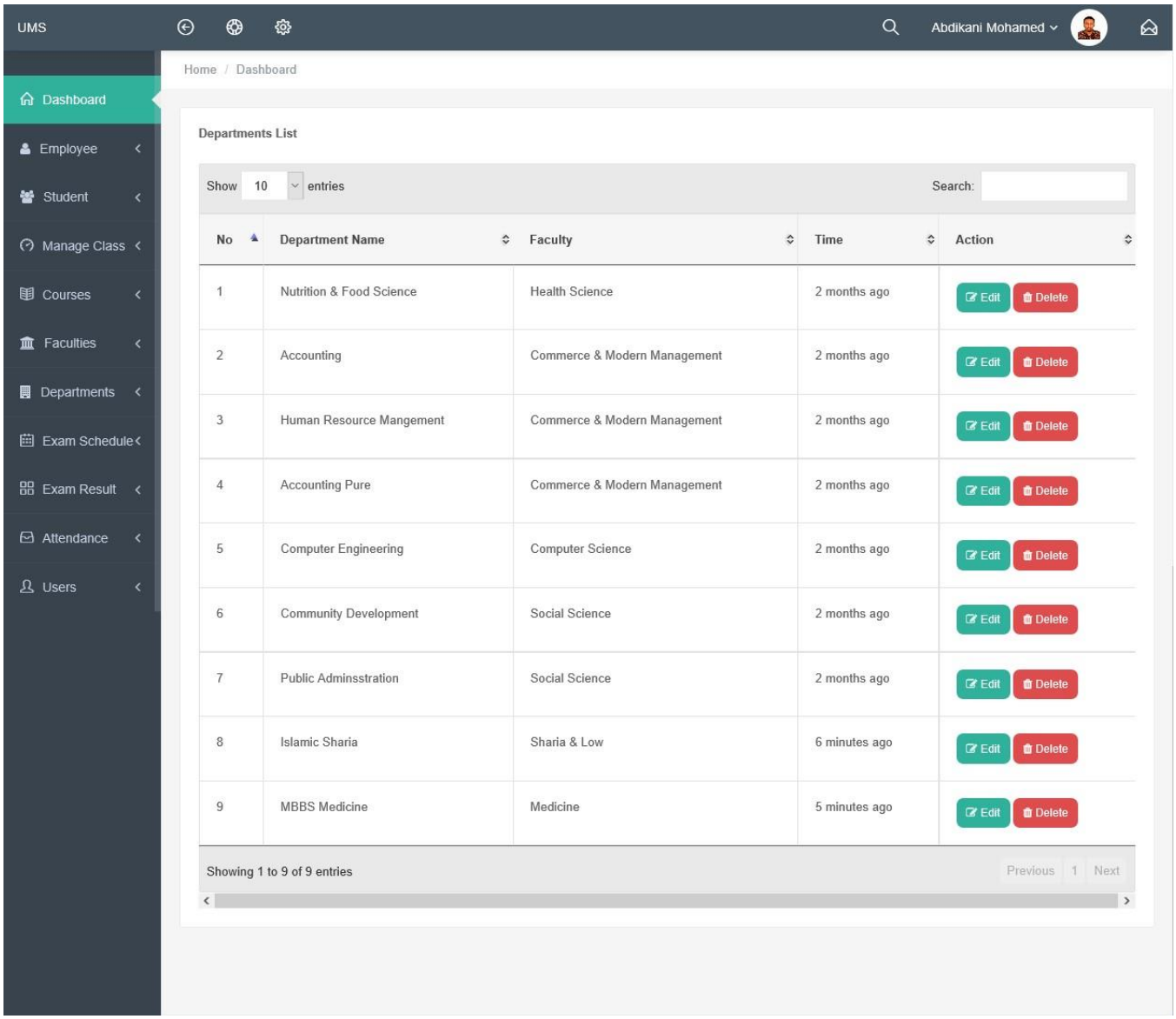


Figure 6.8 : View Registered Departments

6.9 Add Class

When you created faculties and Departments then you can Add University Classes, and select each class his department and Faculties, Now I will provide the screenshot of Add Class page below.

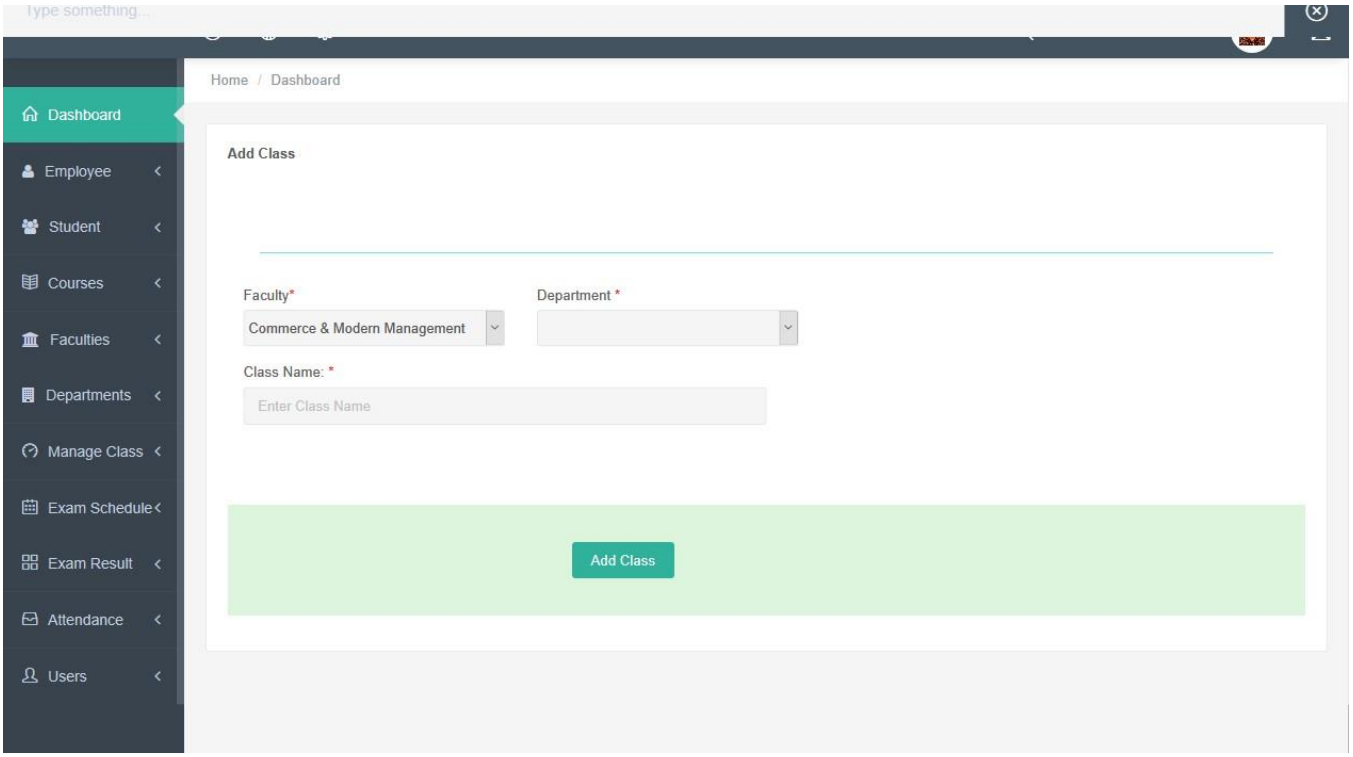


Figure 6.9: Add Class

6.10 View Classes

After you create Class you can see list of available Classes in your system then you can delete them or Update them, Now I will provide the screenshot of Class list below.

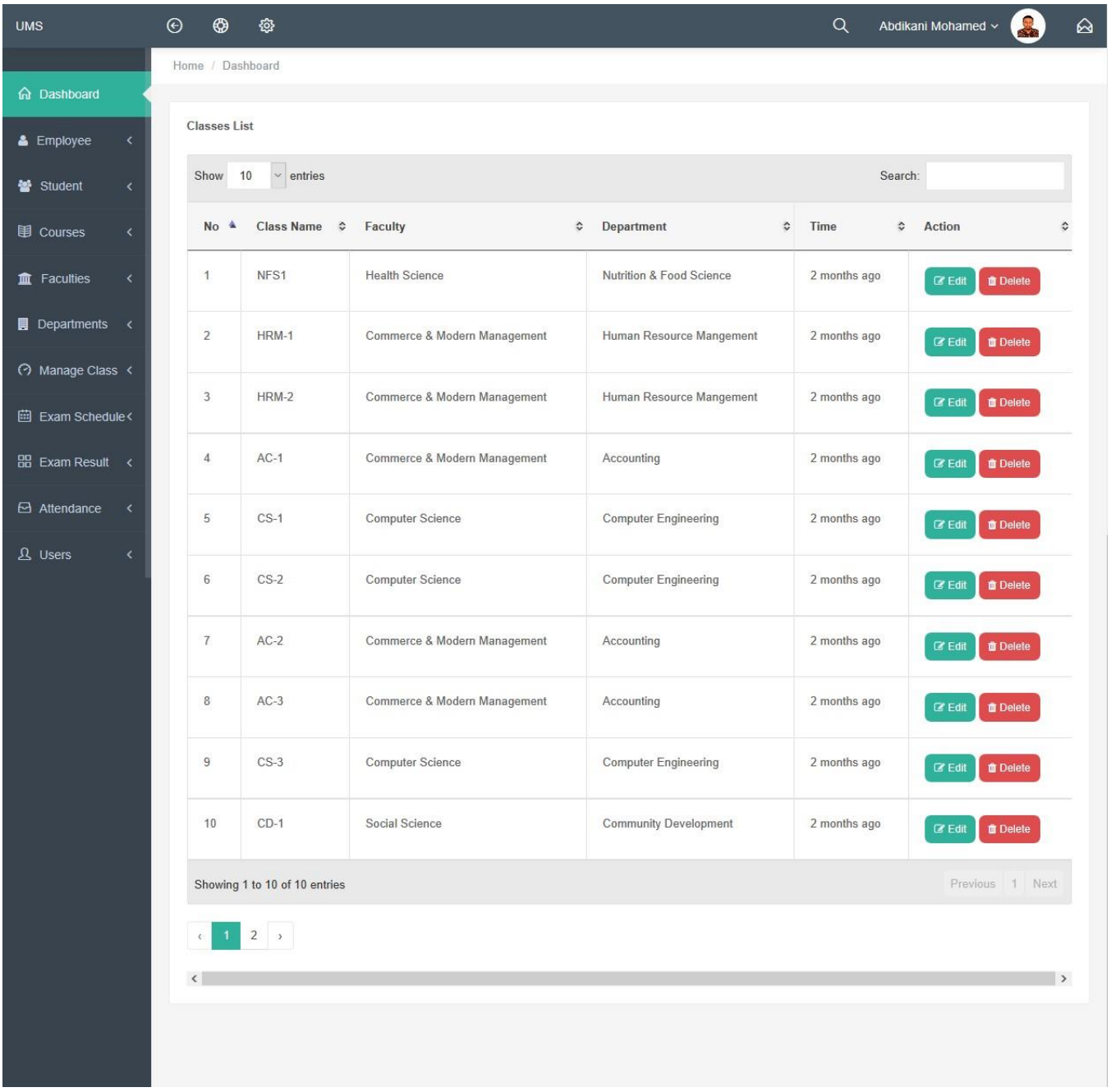


Figure 6.10: View Classes

6.11 Create Semester

Once at the semester you need to create the current semester so that will available in application, Now I will provide the screenshot of creating Semester below.

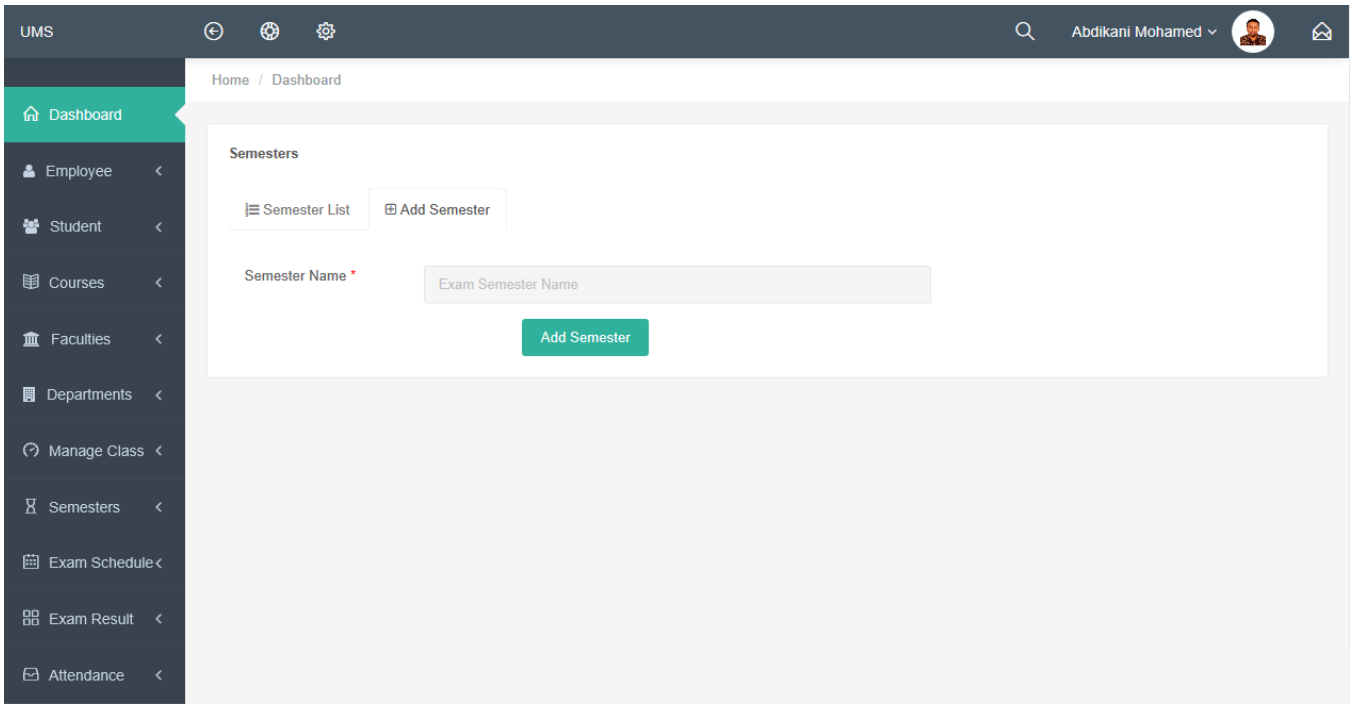


Figure 6.11: Create Semester

6.12 Register Courses

System admins can register course to the system you will add course name plus code and you will select all these (Faculty, Department, Class, Semester) of the course and after your assign teacher to the course, Now I will provide the screenshot of creating Course below.

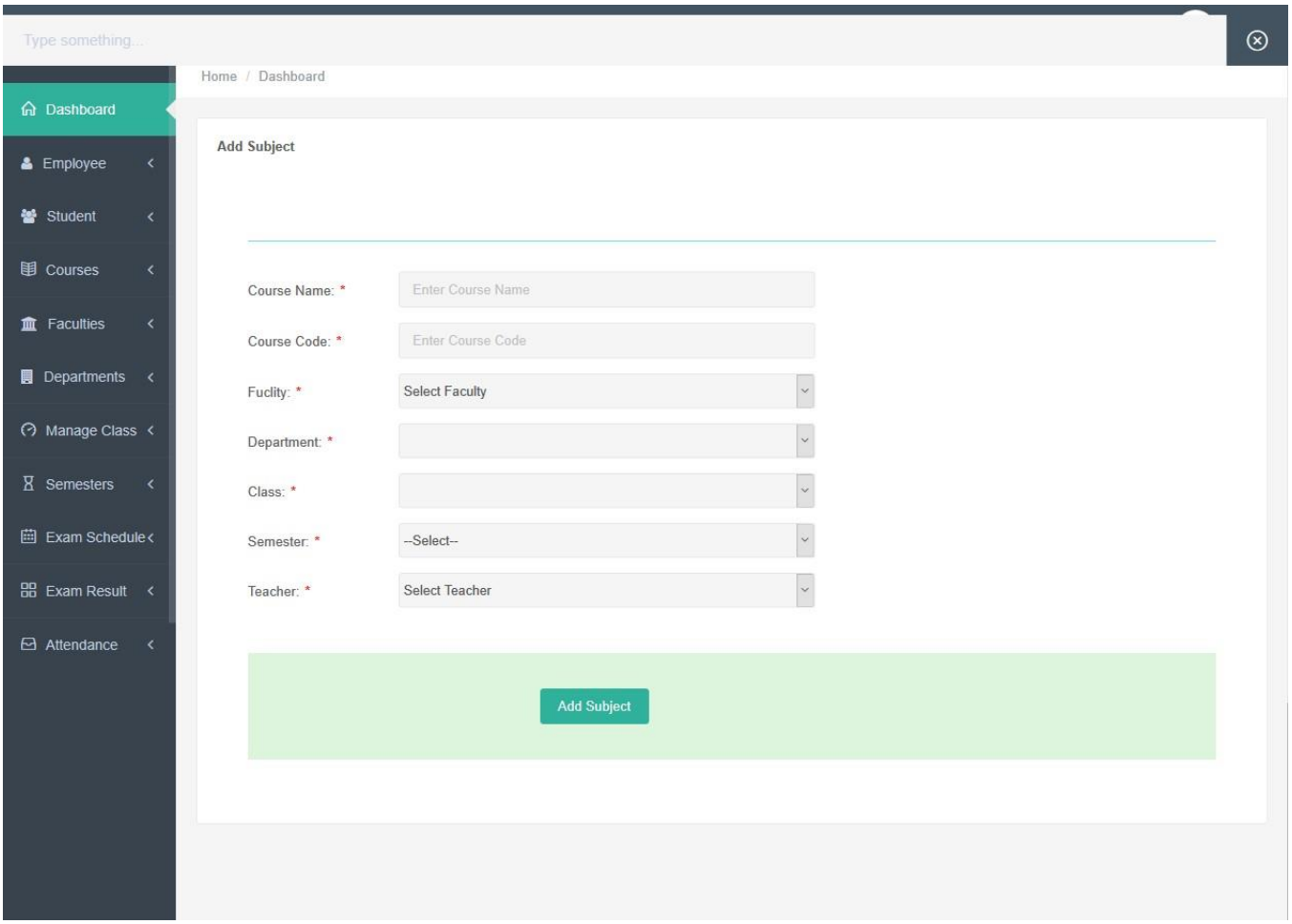


Figure 6.12: Register Courses

6.13 Manage Registered Courses

When you create the course, you can also manage it like you can delete the course or update all these (Faculty, Department, Class, Semester) of the course and you can re-assign teacher to the course,

Now I will provide the screenshot of Courses List below.

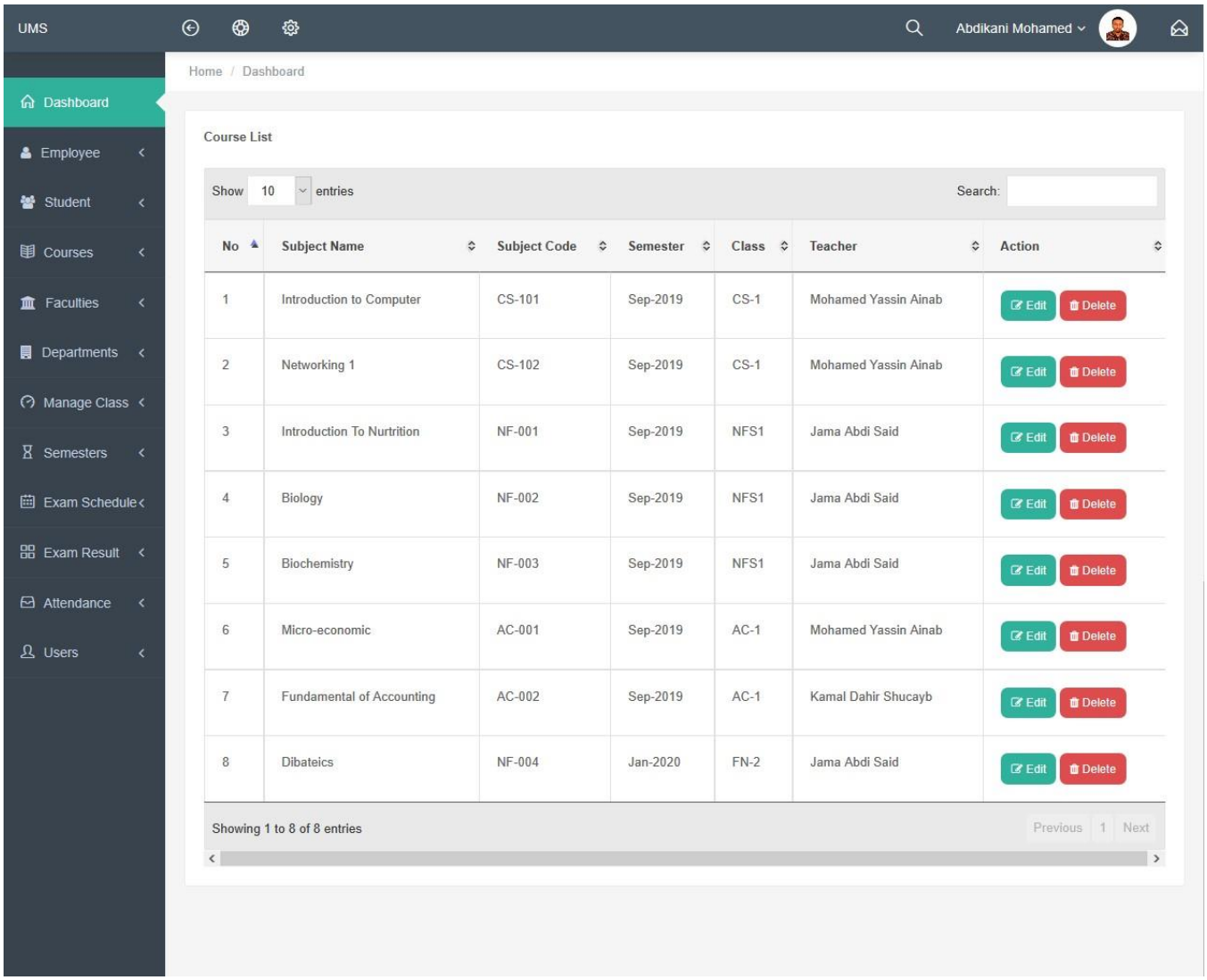


Figure 6.13: Manage Registered Courses

6.14 Student Registration

Student registration is one of the main activities of this application, so that when we are going to register every student should have unique student ID select all these (Faculty, Department, Class, Semester, Courses) of the students, after that you will give student user-name and password, Now I will provide the screenshot of registration of the student Form below.

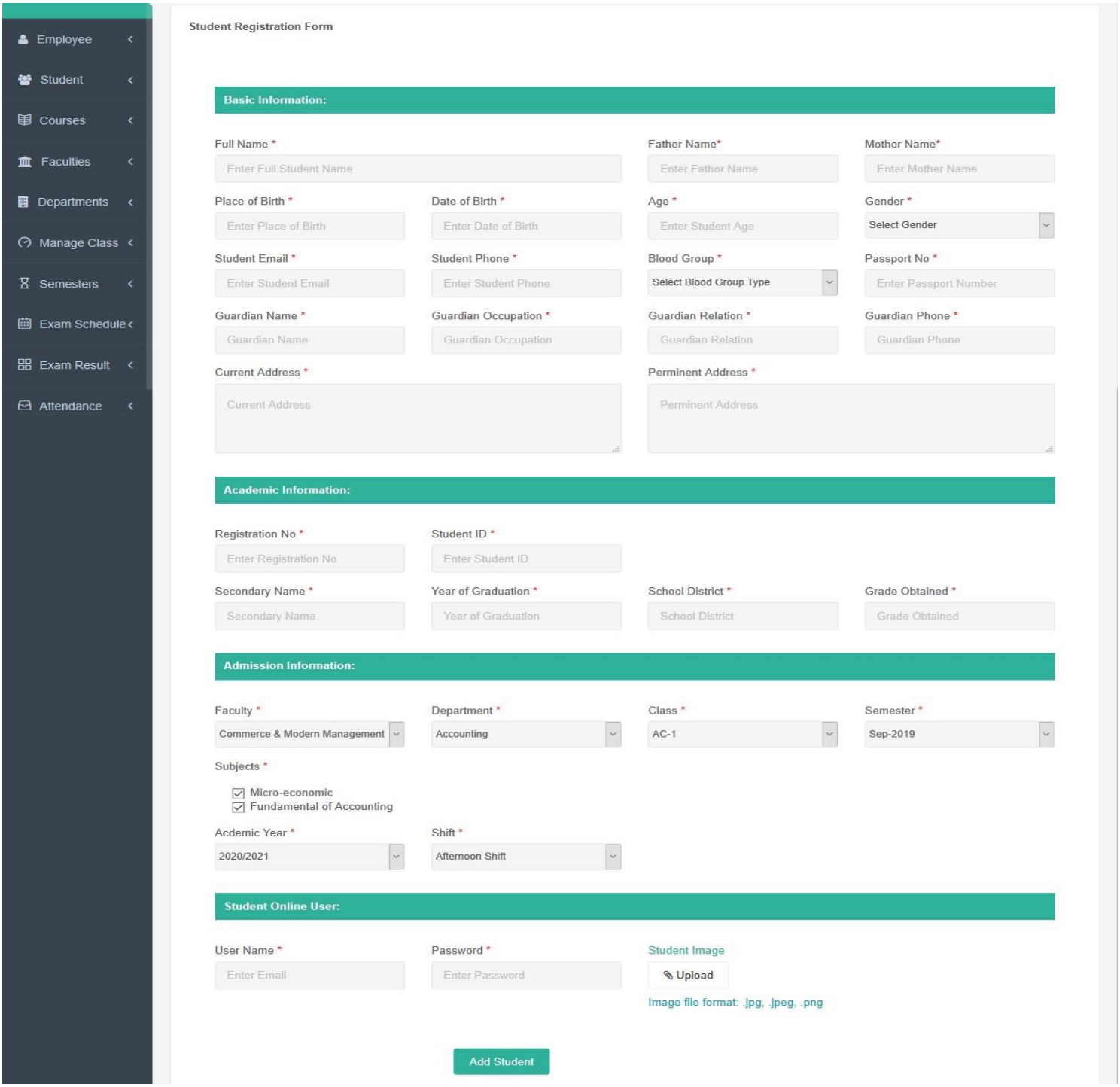


Figure 6.14: Student Registration

6.15 Manage Registered Students

Now you can see list of the registered Students and you can update or delete students,

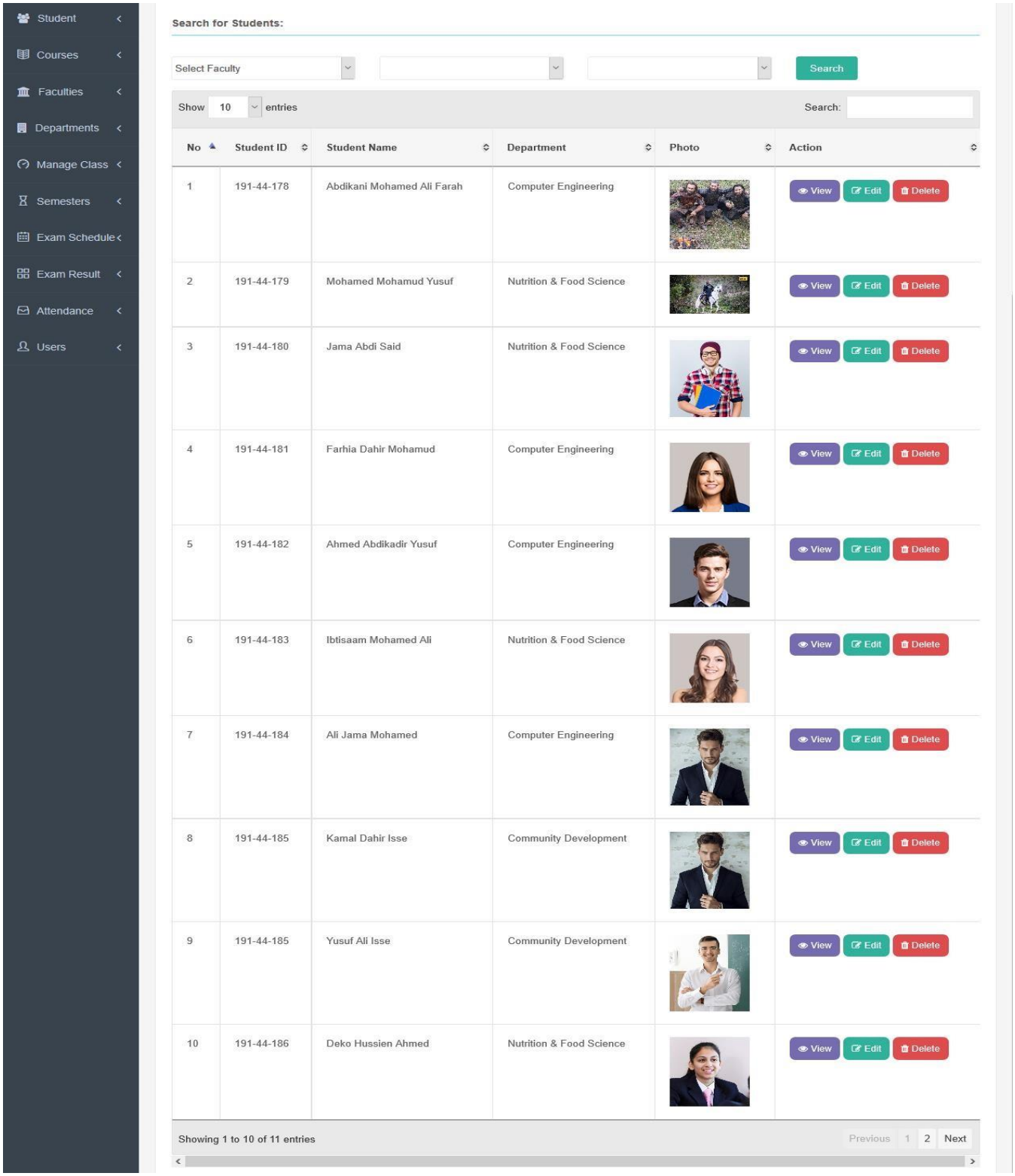


Figure 6.15: Manage Registered Students

6.16 Exam Schedules

Is to add exams schedules for the current semester so that student and teacher know the exact date of the examinations, Now I will provide the screenshot of Exam Schedule below.

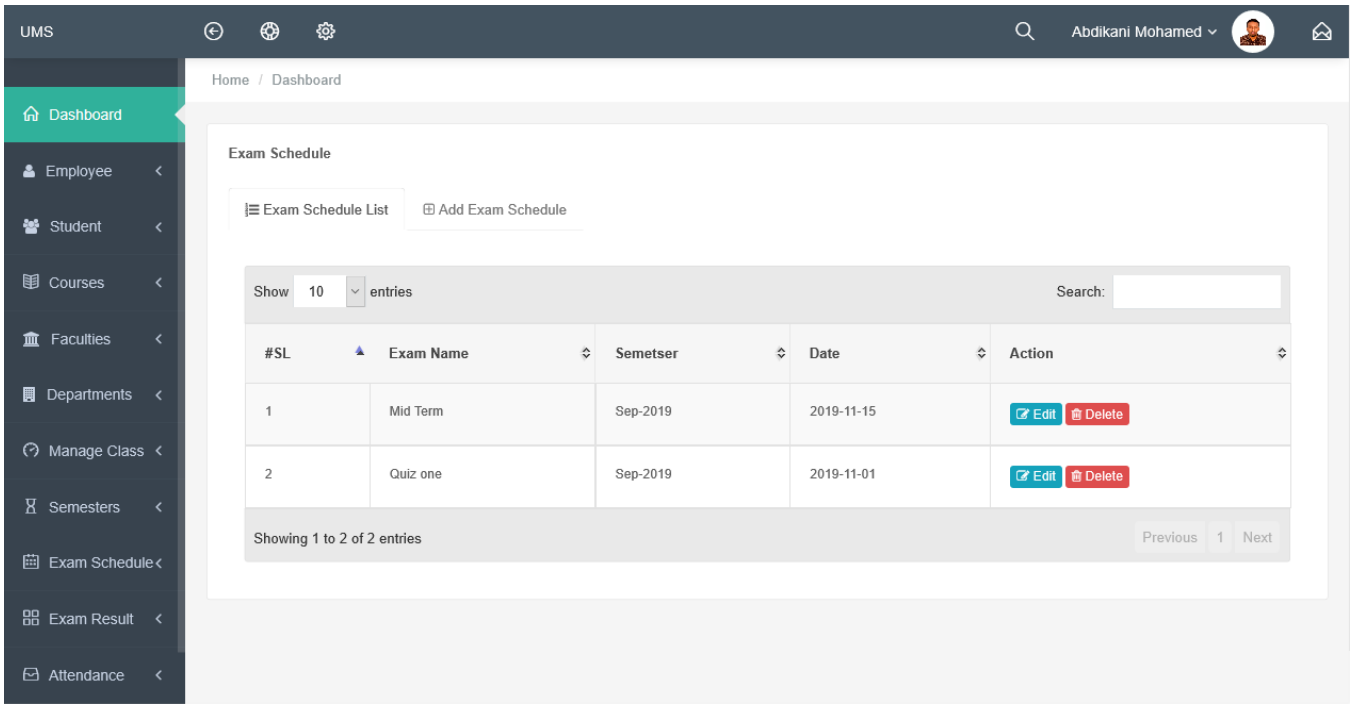


Figure 6.16: Exam Schedules

6.17 Teacher Insert Exam Results

Teacher when they login in to their dashboard they can insert student's exam results for the courses that are assigned for that teacher, Now I will provide the screenshot of teacher insert student exam result below.

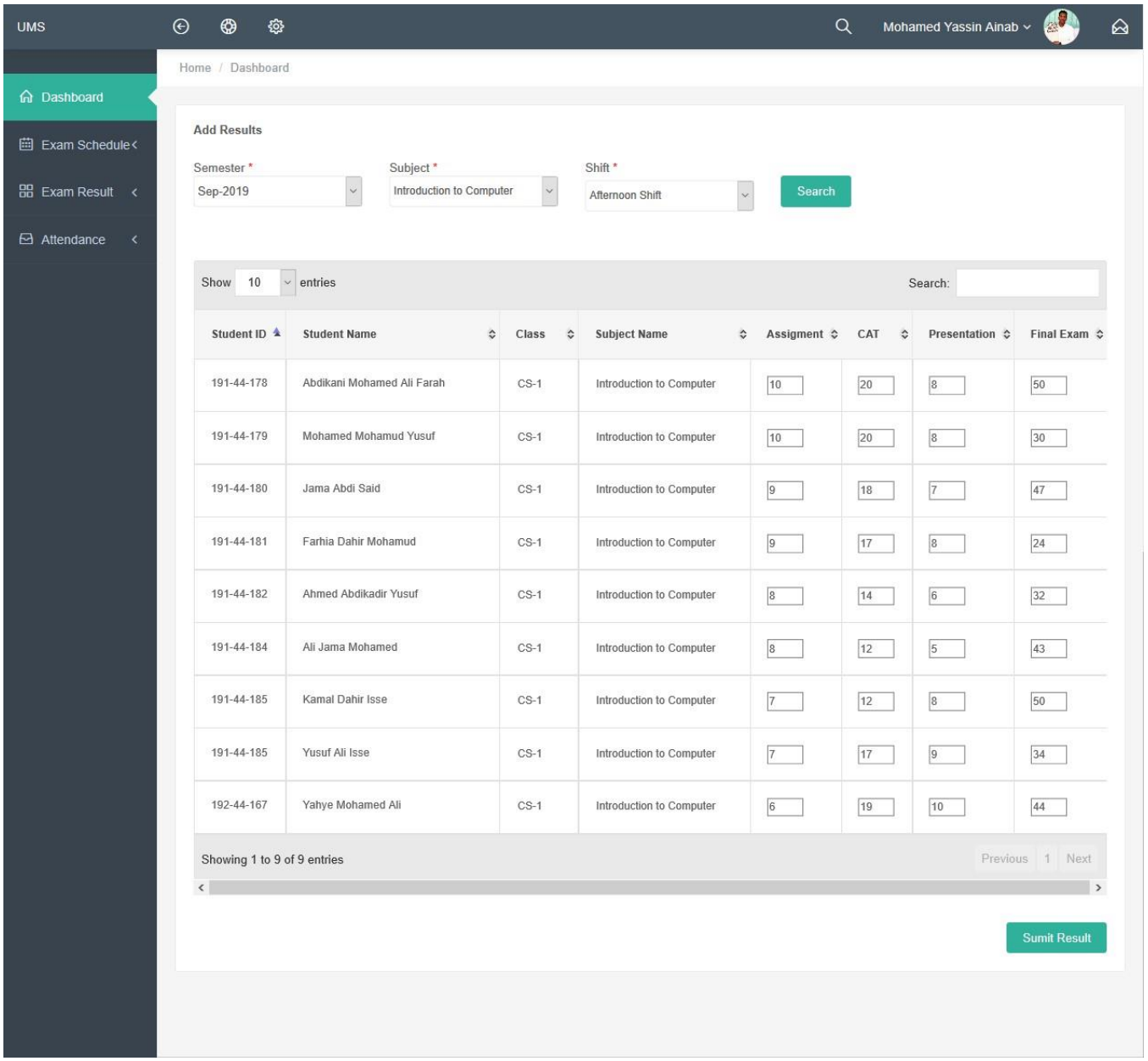


Figure 6.17: Teacher Insert Student Exam Results

6.18 Find Results

When teacher insert the results teachers can see the results and if there is any mistake he can update and submit final results, Now I will provide the screenshot of teacher find exam result below.

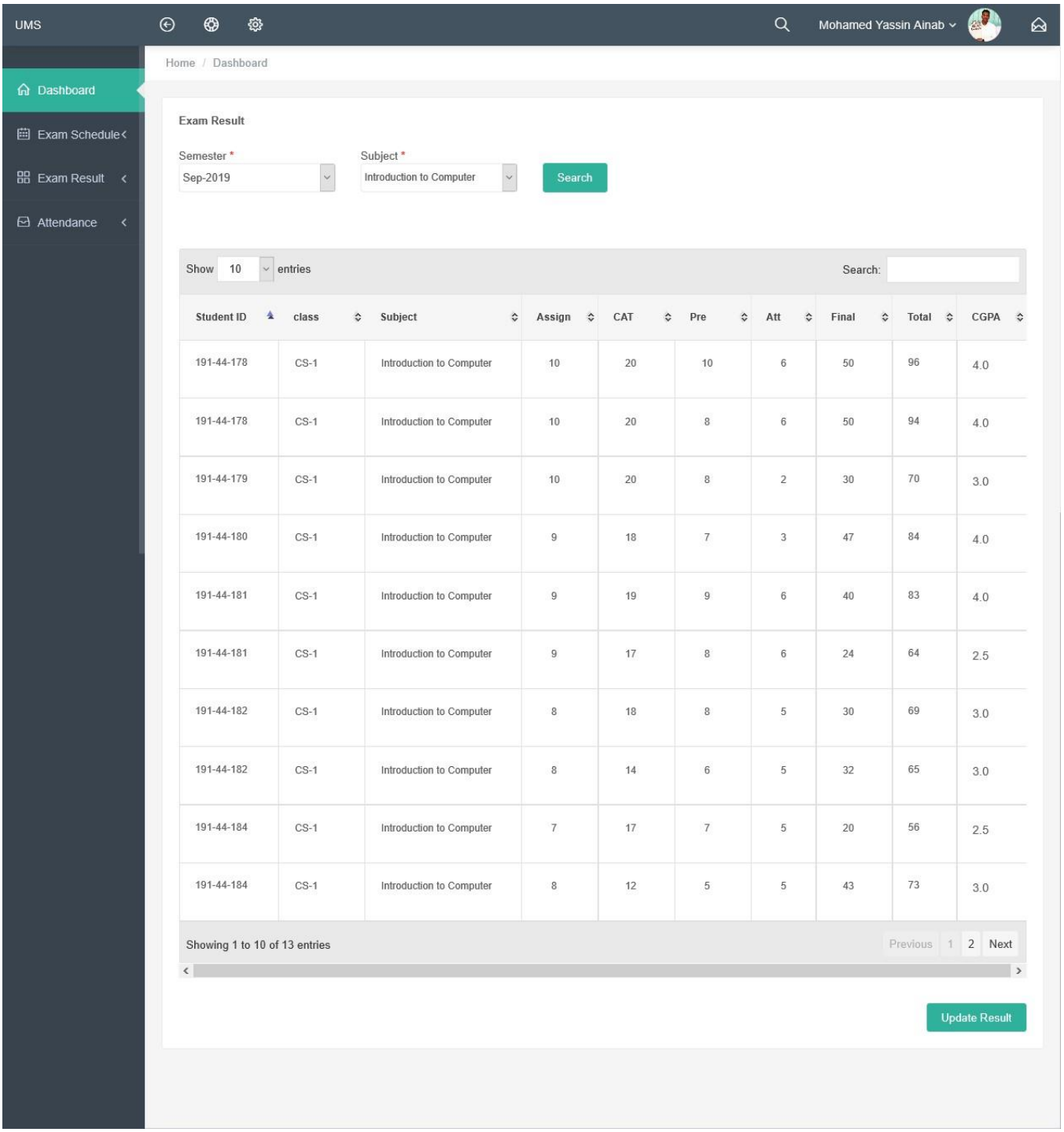


Figure 6.18: Find Student Exam Results

6.19 Teacher Take Class Attendance

Taking class attendance is responsibility of the teacher at end of each he will take class attendance for his courses, Now I will provide the screenshot of teacher take class attendance below.

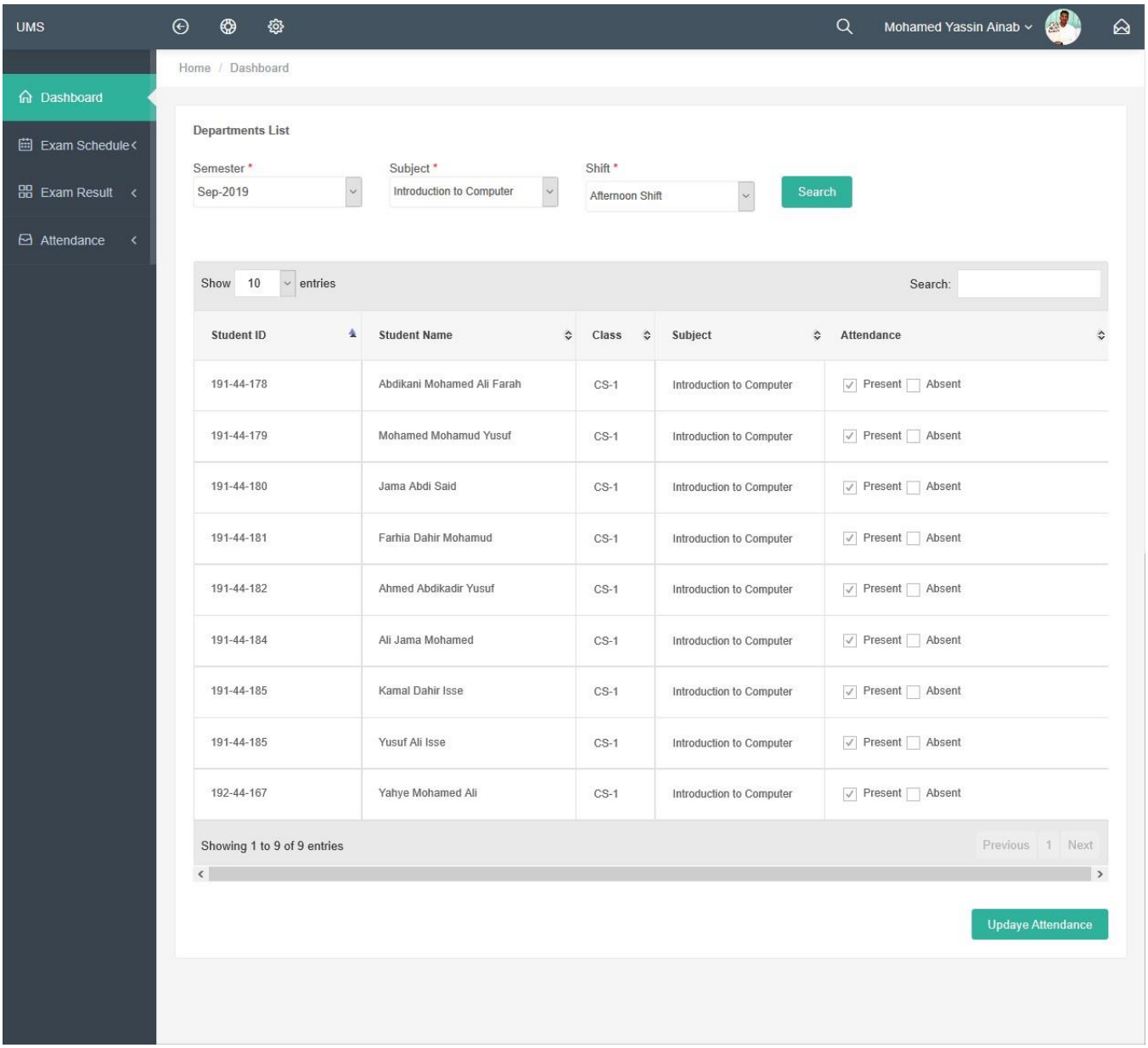


Figure 6.19: Teacher Take Class Attendance

6.20 Generate Attendance Marks

In this system when semester ends the teacher can generate the attendance that he was taking whole the semester into marks for student the system will calculate and generate marks from student attendance days, Now I will provide the screenshot of Attendance summary below.

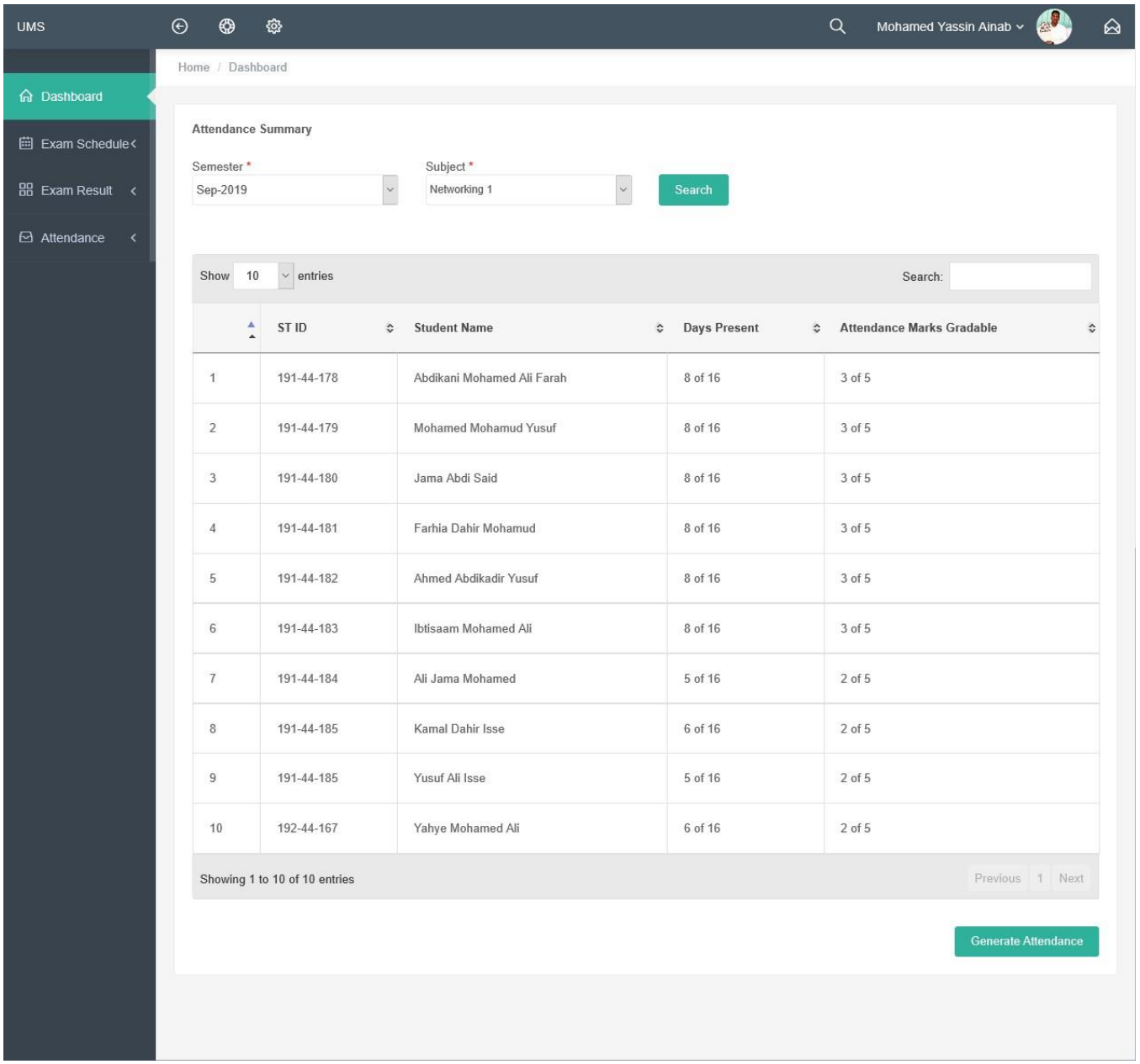


Figure 6.20: Generate Attendance Marks

6.21 Student Portal Login Page

If you are Student you need to login to our application, before that they need to be authenticated by our application. First of all, they need to enter your Student ID and password:12345. Now I will provide the screenshot of Student login page below.

Figure 6.21: Student Portal Login

6.22 Student Portal Dashboard

After student successfully login this is the screen he will see as his dashboard, in this dashboard has three menu students can see his exam results also he can see exam schedules and he can see his profile and change the password of his user. Now I will provide the screenshot of students Portal Dashboard below.

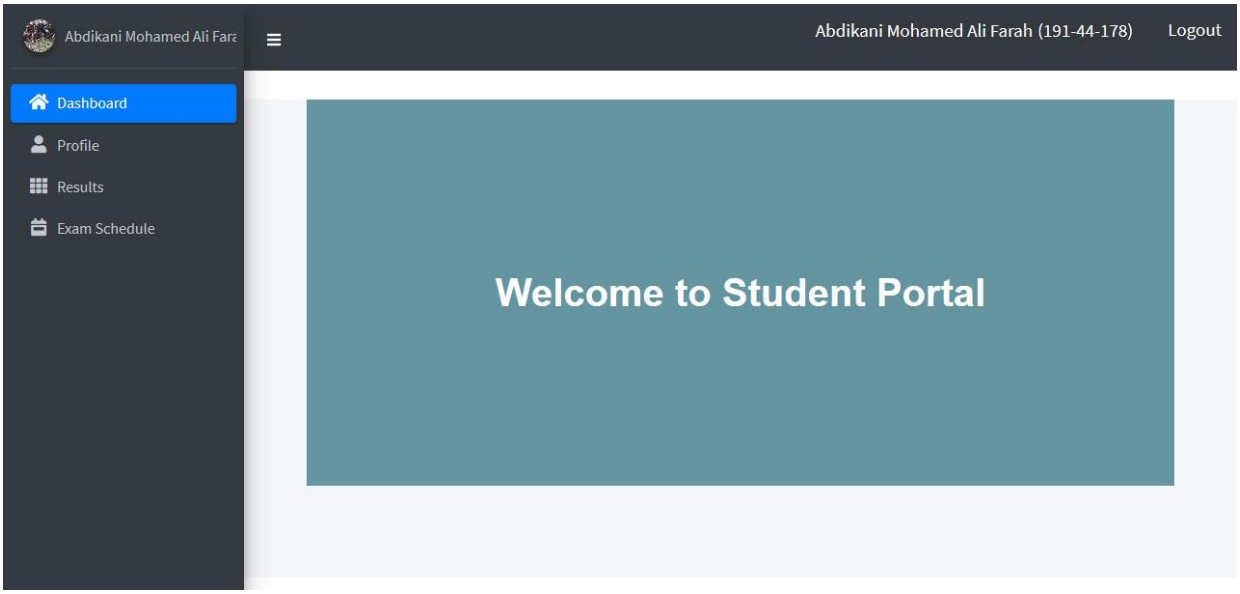


Figure 6.22: Student Portal Dashboard

6.23 Student See Exam Results

After student successfully login he can see his exam result from his portal by choosing the semester and then see the result. Now I will provide the screenshot of students see exam results below.

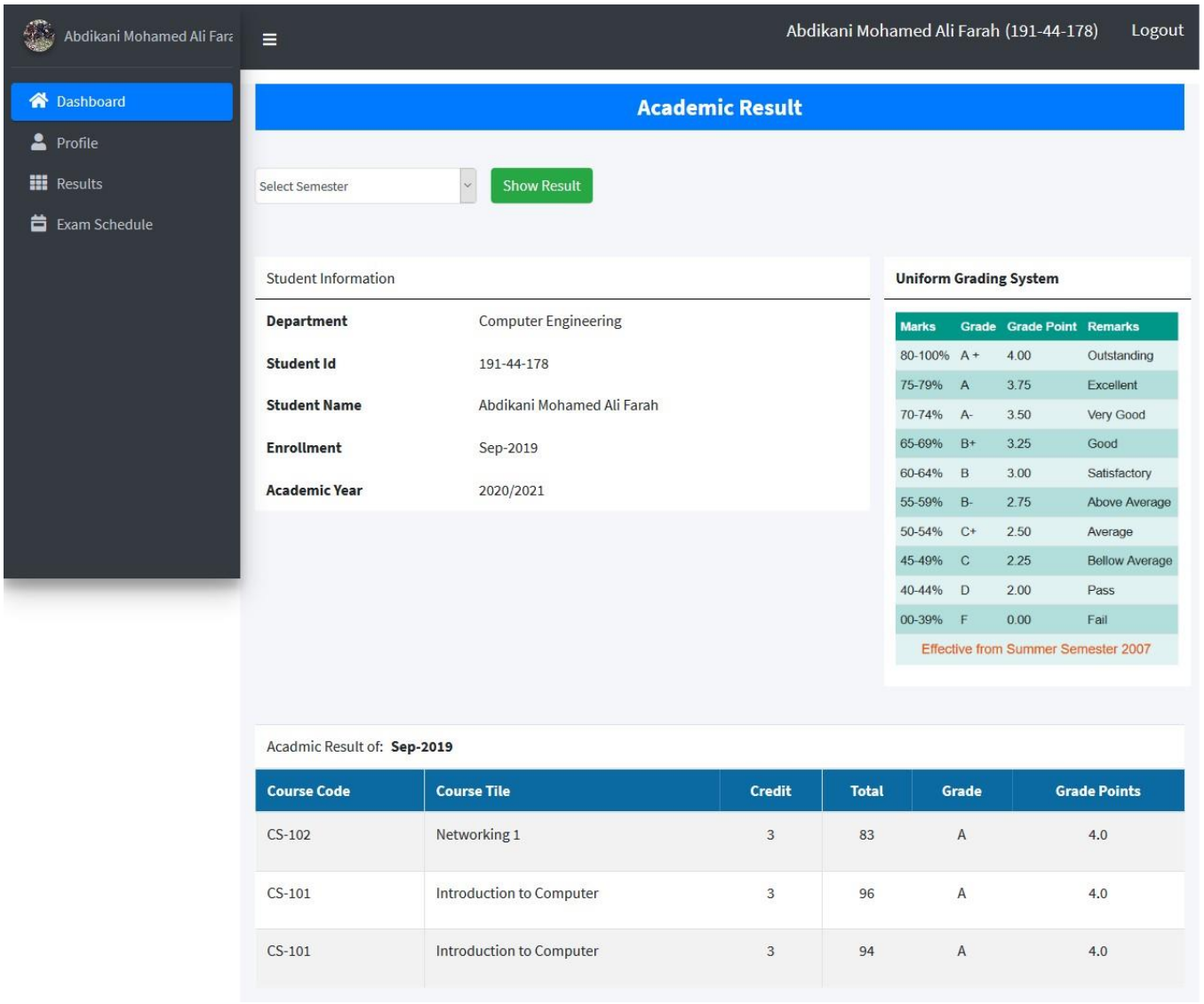


Figure 6.23: Student See Exam Results

6.24 Student see his profile and change password

Students can see his information and he can change the default password of 12345 into whatever he wants. Now I will provide the screenshot of students Profile below.

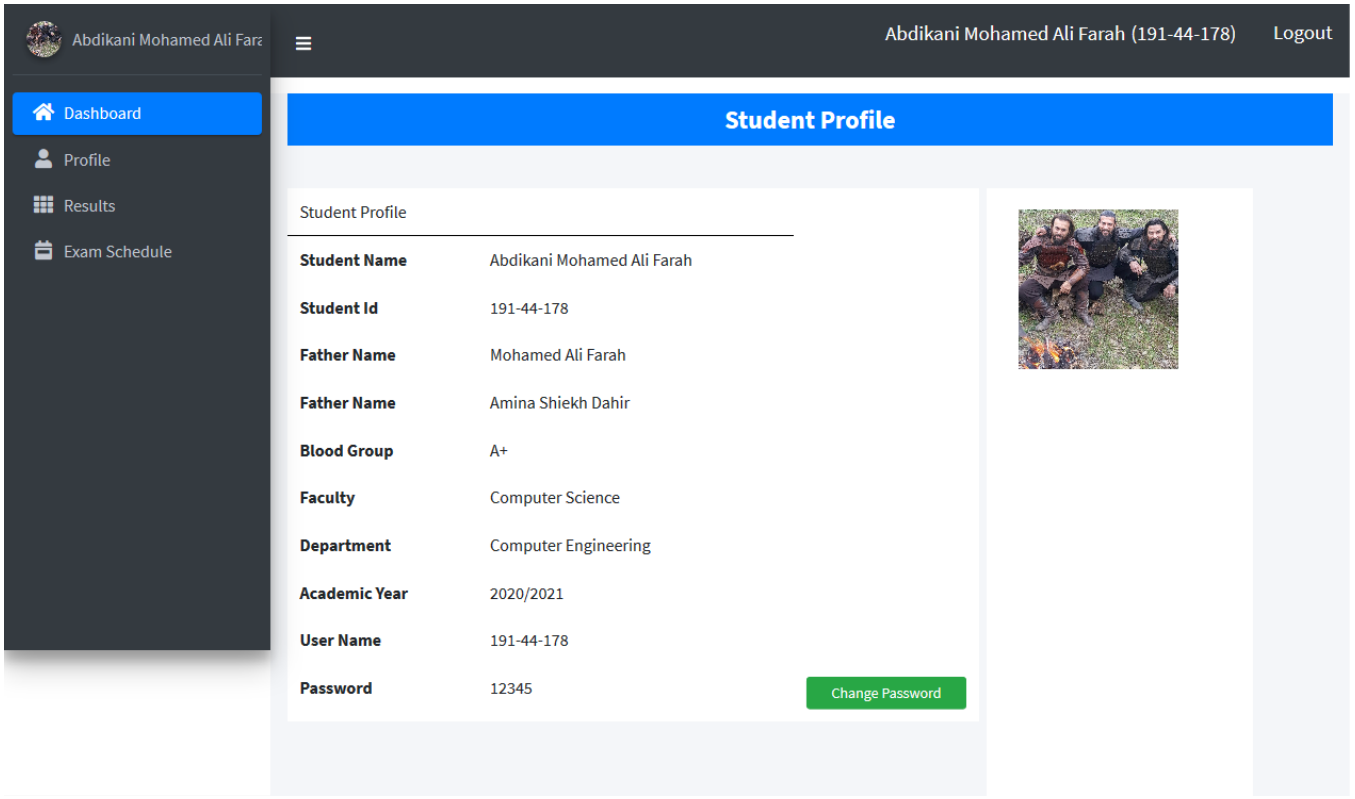


Figure 6.24: Student Change Password

6.25 Student see Exam schedules

It's better that student know examination schedules so that he will be prepare for it like quiz one, two and midterms as also final Exam schedules Now I will provide the screenshot of Exam schedule below.

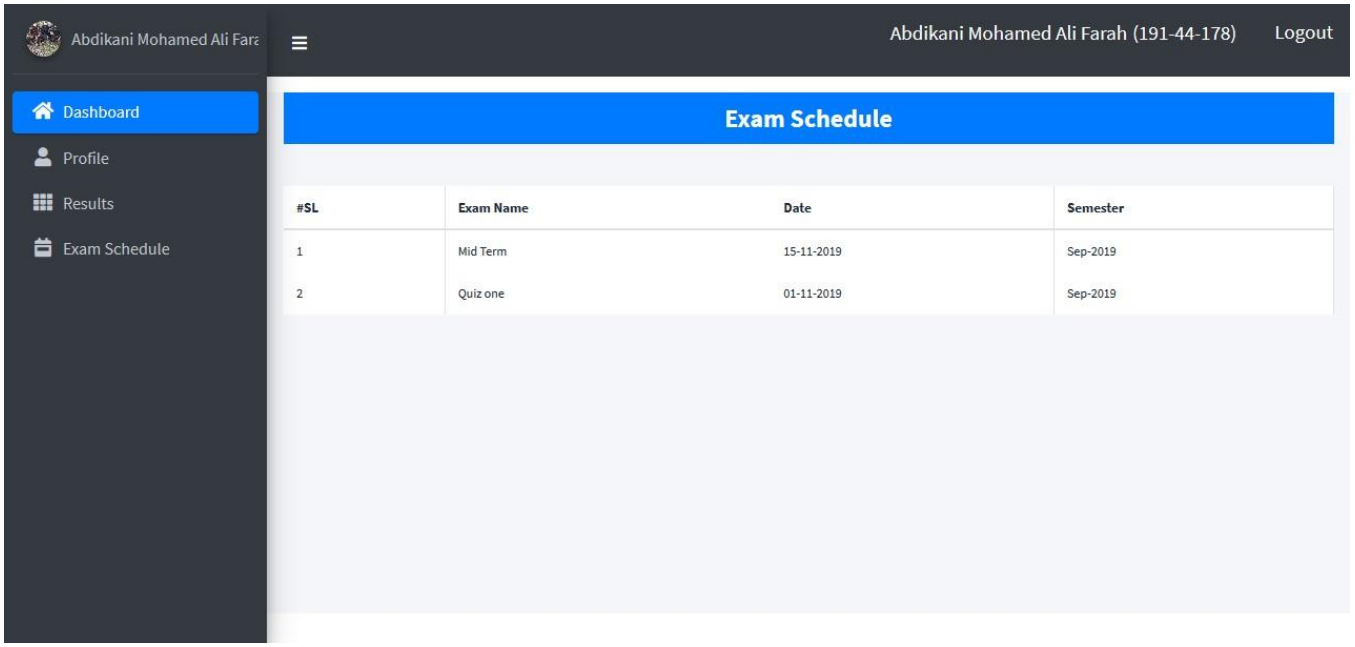


Figure 6.25: Student see Exam schedules

Chapter 7
Conclusion

7.1 GitHub Link

https://github.com/xudayfe112/university-management-system

7.2 Project Summary

I have started to develop this project from July. From the very starting of development of this project, it demands hard working, patients, persistency to meet the requirements of stakeholders. After that I have proposed the design. And then started to work.

Database plays a vital role for any application software. And so this is why, I have designed the database diagram having tables with proper relationship. After that, I have prepared the user interface and continue to the next part. It is to be said that, the interface of my application is very simple and easy to understand. After completing that, I have started to write the core functionality of the project.

Developing project is not end of all tasks actually. There are some other important tasks to perform. And that is testing. It is also known as quality assurance also. Almost at every software company there are a quality assurance team. Their main responsibility is to find the loop holes or vulnerability of software. If there any bug remains before handover to the stakeholders, there is a change to ruin the whole project. So testing plan is very important. And after developing the project, I have assured the quality of this project.

7.3 Limitations

For developing this project, I have faced some limitations. Now I will describe those in brief.

- **Student Payment:** In our application, there is not having any finance functions that can keep students' fees payment when students pay and students see so far what they paid and what is remaining payments is one limitation of our system.
- **Teacher Evaluation:** in this application there is no any teacher evaluation function that student can evaluate their teacher and management see the results of evaluation survey.
- **Online admission:** in our application the student they cannot request admission online unless they have to go to the office and get admission manually.

7.4 Obstacles and Achievements

I believe that if there are not any obstacles to develop a project, then there doesn't have any challenges. Because we know, challenge give us the opportunity to prove ourselves. Obstacles, challenges and achievements are like a path to the success.

Before starting this project, I didn't know the actual flow of software development life cycle. By developing this project, I have learnt to know how to have a row requirement from clients.

After that I have learnt system analysis, database design and many things. My supervisor helps me a lot from the very beginning of the development of this project.

There are some other obstacles and achievements also that I will describe below.

- **Lack of Stakeholder's Engagement:** There are different types of stakeholders in our application. And each stakeholder uses different functionality. And almost they are busy with their day to day activities so, this is why, I didn't get all of them in proper time.
- **Scope Change:** Sometimes, some features need to be changed or modified. Then I need to follow reverse engineering process. And again, designed to meet the new requirements. It also made me frustrated sometimes.

7.5 Future Scope

I have learnt a lot throughout the whole development stage of this project. For making this project developed. I got extra skills from every challenge that i have face, I am very much thankful to all my colleagues as they share their idea and discussion gave me some opportunities to make my system complete. It will help me to work with similar type project in future also.

7.6 References

I have gained some knowledge from some platforms. Obviously, I will mention those references. For making my project successful those resources help me a lot. Not I will mention the names below.

- Aditi Sarkar, 16-aug-2016, Online School Management System Project Paper, Brac University.
- Song, Shaping, "A book management system library" (2004). Theses Digitization Project. 31
- https://www.freeprojectz.com/uml-diagram/university-management-system-uml-diagram
- www.google.com
- www.youtube.com
- www.github.com
- www.wikipedia.com
- www.php.net
- www.mysql.com
- www.themeforest.net

- www.getbootstrap.com
- www.w3schools.com
- www.jquery.com
- www.stakeoverflow.com