SOFTWARE ENGINEERING
INNOVATIVE PROJECT (MC-310)

ATTENDANCE TRACKING AND MANAGEMENT SYSTEM

Submitted to: Sh. Jankhongam Thouthang

SHIVAM SHAURYA 2K18/EE/194

For Codes and More, Scan the QR Code

INTRODUCTION AND PROBLEM STATEMENT

With the increasing number of students and the courses, the departments, faculties, and the students are facing issues like:

- The attendance is taken manually by the faculty.
- Data in the attendance list might be inaccurate due to deception.
- Faculty needs to manually analyze the number of absentees and the percentage.
- It is very time-consuming and the result of the calculation might be inaccurate.
- Data loss may happen and the whole attendance process is prone to human mistakes.

The proposed software would allow the following functionalities:

- It enhances the speed of performing attendance task easily.
- The entire student's attendance data will be stored and managed properly and effectively.
- The system enables the lecturer to add, view, delete or make changes accordingly.
- The system enhances the calculation process to be more accurate and fast.
- The system will help to analyze all the data inserted and then verified whether every student is following the university attendance policy or not.
- The attendance will be marked period wise and the reports of the students will be generated.

INITIAL REQUIREMENTS DOCUMENT

Title of the Project	Attendance Tracking Management System
Stakeholders Involved in Capturing Requirements	Administrator, Faculty and Students
Techniques Used for Requirement Capturing	Interviewing and Brainstorming
Name of the Persons along with Designations	Mr. Jankhongam Thouthang Shivam Shaurya 2K18/EE/194 Pulkit Aggarwal 2K18/EC/122
Date	14 May 2021
Version	1.0

Consolidated List of Initial Requirements:
- A system is to be implemented which can be run on the University's LAN.
- The system shall be able to generate login ID and password to the system operator.
- There are two types of members in the university to use the system – Faculty and Students.
- The Administrator shall be able to maintain details of the courses.
- The Administrator shall be able to maintain details of the students and the faculties.
- The Faculty and students shall be able to login using the generated ID and password.
- The Faculty shall be able to view information about courses.
- The Faculty shall be able to maintain (add/update/view) attendance in the courses.
- The system shall calculate the percentage of attendance of each student for every course he/she is enrolled in.
- Students shall be able to view their attendance.
- The system should also be able to generate reports like :
  a) Details of all the members.
  b) List of students enrolled course wise and their attendance percentage.
  c) List of students with attendance percentage less than 75%.
- The system shall be able to provide easy interface for the users.

Table of Contents

1. Introduction-

The Attendance Tracking Management System (ATMS) for students is to be developed. The system would allow the following functionalities:
- It enhances the speed of performing attendance task easily.
- The entire student's attendance data will be stored and managed properly and effectively.
- The system enables the faculty to add, view, delete or make changes accordingly.
- The system enhances the calculation process to be more accurate and faster.
- The system will help to analyze all the data inserted and then verified whether every student is following the attendance policy or not.
- The attendance will be marked date wise and the reports of the students will be generated.

1.1. Purpose
The Attendance Tracking Management System (ATMS) maintains the information about the attendance of the students in the primary school. The system also maintains the record of all the students and faculty in the school. The records are maintained to maintain the attendance of students.

1.2. Scope
- The product is titled Attendance Tracking Management System (ATMS). The system will be referred as ATMS in rest of the SRS.
- The proposed ATMS must be able to perform the following functions:
  Dos:
  - Issue of login username and password to system operators.
  - Maintain details of courses offered in the school.
  - Maintain details of the students in the school.
  - Maintain details of the faculty in the school.
  - Maintain attendance.
  - Generate Reports of the following:
    - List of courses
    - List of students
    - List of defaulters

  Don'ts
  - Students can't enroll for multiple courses and faculties can teach multiple courses.
  - Complaint and queries section is not available
  - Leave Application is not covered

Benefits

The ATMS provides the following benefits:

- Easy maintenance of attendance.
- Accurate and automated attendance percentage calculation.
- Printing of reports.

1.3. Definitions, Acronyms, and abbreviations

Define all terms, acronyms and abbreviations. This may help to make the SRS document more readable, understandable and clear to all the stake holders. The definitions and acronyms used in LMS are given as:

- **SRS:** Software Requirement Specification
- **ATMS:** Attendance Tracking Management System
- **System operator:** System administrator, faculty, data entry operator
- **RAM:** Random Access Memory
- **Student:** Any candidate admitted in a course offered by the school
- **System Administrator:** User having all the privileges to operate the ATMS.
- **Faculty/Teacher:** Teaching Staff

1.4. References

Provide list of all documents including books, research papers referenced anywhere in the SRS document. The referenced material used in ATMS is given as:

- Object Oriented Software Engineering' by Yogesh Singh & Ruchika Malhotra, New Age Publishing House, 3rd Edition, 2008.
- IEEE Recommended Practice for Software Requirements Specifications – IEEE Std. 830-1998.
- IEEE Standard for Software Test Documentation – IEEE Std. 829-1998.

1.5. Overview

- The SRS contains an analysis of the requirements necessary to help easy design.
- The overall description provides interface requirements necessary to help easy design system, product perspective, hardware interfaces, software interfaces, communication interface, memory constraints, product functions, user characteristics and other constraints.
- Succeeding pages illustrate the characteristics of typical naïve users accessing the system along with legal and functional constraints enforced that affect Attendance Tracking Management System in any fashion.

2. Overall Description

- The ATMS maintains records of attendance of students and details of students, faculty and courses in the school.

- It is assumed that the students have already admitted in the school.
- It is assumed that the students are already enrolled in the courses.
- The system administrator will receive the lists of the admitted students (course wise) from the academic section.
- The establishment section will provide the list of faculty appointed and courses offered in the school.
- The ATMS maintains and views the attendance of students enrolled in the course offered by the school.
- It allows the faculty to mark the attendance of the students in his class.
- It allows the student to view his attendance.
- The administrator/DEO will have to maintain the following information:
  - Attendance details (only admin)
  - Student details
  - Faculty details
  - Course details

2.1 Product Perspective

The Attendance Tracking Management System shall be developed using client/server architecture and will be compatible with any Operating System. The front end of the system will be developed using HTML and CSS, and the back-end will be developed using PHP and MySQL. It provides simple database rather than complex ones for high requirements and it provides good and easy graphical user interface to both new as well as experienced user of the computer.

2.1.1 System Interfaces
None

2.1.2 User Interfaces
The ATMS will have the user-friendly and menu-driven interfaces:
a) **Login:** to allow the entry of only authorized users through valid username and password.
b) **Maintain Login:** to maintain login details.
c) **Maintain Faculty:** to maintain faculty details.
d) **Maintain Student:** to maintain student details.
e) **Maintain Courses:** to maintain course details.
f) **Maintain Attendance**: to maintain attendance details.
g) **View Attendance:** to view/display the attendance of students according to courses.
h) **Report Generation:** to generate report of student's attendance.

2.1.3 Hardware Interfaces
a) Screen Resolution of at least 640 * 480 or above.
b) Support for printer.
c) Computer systems will be in the networked environment as it is a multi-user system.

2.1.4 Software Interfaces
a) Any Operating System web browser
b) HTML and CSS for designing front end.
c) PHP and MySQL for designing the back end.

2.1.5 Communication Interfaces
Communication is via Internet as it is a web-enabled system.

2.1.6 Memory Constraints
At least 2 GB RAM and 1 GB space of hard disk will be required to run the software.

2.1.7 Operations
None

2.1.8 Site Adaptation Requirements
The terminal at the client side will have to support the hardware and software interfaces specified in sections 2.1.3 and 2.1.4, respectively.

2.2 Product functions
The ATMS will allow access only to authorised users with specific roles (system administrator, data entry operator, faculty and student). Depending upon the user's role, he/she will be able to access only specific modules of the system.

A summary of major functions that the ATMS will perform is given as follows:
- A login facility for enabling only authorized users to login using the generated username and password.
- The system shall allow the Administrator/DEO to maintain the details (add, update, delete and view) of students and courses.
- The system shall allow the students to view the details of their attendance.
- The system shall generate reports for attendance of student.

2.3 User Characteristics
- **Qualifications:** At least matriculation and comfortable with English.
- **Experience:** The Administrator and the department staff should be well versed with the details of courses as well as the users.

- **Technical Experience:** Elementary knowledge of computers.

2.4 Constraints

- Password can't be updated once added.
- The delete, add and update operation is available to the administrator and data entry operator. To reduce the complexity of the system there is no check on delete operation. Hence the administrator or the DEO should be very careful before deletion of any record and he/she will be responsible for data consistency.
- The user will not be allowed to update the primary key.
- Students cannot enrol in multiple course

2.5 Assumptions and Dependencies

- The username and password must be created by system administrator and communicated to concerned user confidentially.
- Each department of the school should provide the list of all the courses to the administrator.
- The administrator should provide the list of registered students to the concerned department.

2.6 Apportioning of Requirements

Not required

3. Specific Requirements

This section contains the software requirements in detail along with the various forms to be developed.

3.1 External Interface Requirements

3.1.1 User Interfaces

The following interfaces will be provided by the system.

1. **Login**

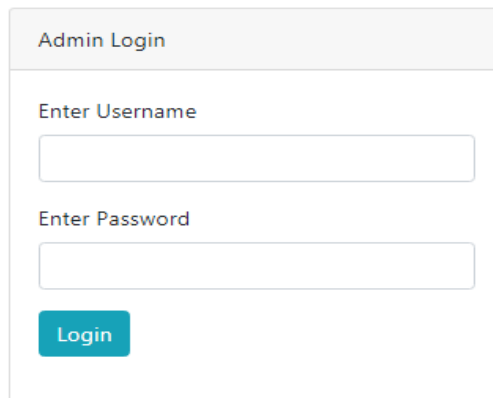This will be the first interface which will be provided by the system.

This has 4 options :
- Admin Login
- Date Entry Operator Login
- Faculty Login
- Student Login

Admin login
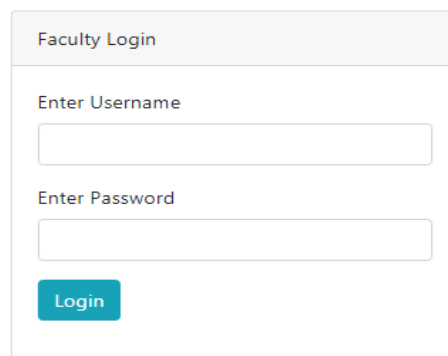

Attendance Tracking Management System

Admin Login

Enter Username

Enter Password

Login

Various fields available on this form will be:
- Username: Alphanumeric and can be of any number of characters. Blank spaces are not allowed.
- Password: Alphanumeric and can be of any number of characters. Blank spaces are not allowed.

Faculty login

Attendance Tracking Management System

Faculty Login

Enter Username

Enter Password

Login

Various fields available on this form will be:

- Username: Email-Id is their username. Blank spaces are not allowed.
- Password: Alphanumeric and can be of any number of characters. Blank spaces are not allowed.

Student Login

Attendance Tracking Management System

Student Login

Enter Username

Enter Password

Login

Various fields available on this form will be:

- Username: Email-ID is their username. Blank spaces are not allowed.
- Password: Alphanumeric and can be of any number of characters. Blank spaces are not allowed.

2. **View attendance**

Attendance Tracking Management System

Home Attendance Defaulter Course Faculty Student Admin DEO Logout

Attendance List   Add Chart Report

Show 10 ⬍ entries   Search:

Roll Number	Student Name	Course	Attendance Status	Attendance Date	Faculty	Edit	Delete
			No data available in table				

Showing 0 to 0 of 0 entries   Previous Next

This interface displays the list of the attendance of students with the following attributes

- Roll Number
- Student Name
- Course

- Attendance Status
- Attendance Date
- Faculty Name

3. Overall student attendance status

Attendance Tracking Management System

Home Attendance Defaulter Course Faculty Student Admin DEO Logout

Overall Student Attendance Status

Show 10 ⬍ entries

Search:

Student Name	Roll Number	Course	Faculty	Attendance Percentage	Report
		No data available in table			

Showing 0 to 0 of 0 entries

Previous Next

This interface displays the list of the attendance of students with the following attributes

- Roll Number
- Student Name
- Course
- Faculty Name
- Attendance Percentage

4. Student List

Attendance Tracking Management System

Home Attendance Defaulter Course Faculty Student Admin DEO Logout

Student List

Add

Show 10 ⬍ entries

Search:

ID	Roll No.	Student Name	Date of Birth	Course	Email ID	Edit	Delete
			No data available in table				

Showing 0 to 0 of 0 entries

Previous Next

This interface displays the list of the attendance of students with the following attributes

- Student ID
- Roll Number
- Student Name
- Date of Birth
- Course

- Email ID

5. Faculty List

This interface displays the list of the attendance of students with the following attributes
- Faculty ID
- Name
- Email Address
- Course

6. Generate Report

Various fields available on this form will be:
- From Date: Starting date from which the report should begin.
- To Date: Ending date to which the report should end.

3.1.2. Hardware Interfaces
a) Screen Resolution of at least 640 * 480 or above.

b) Support for printer.

c) Computer systems will be in the networked environment as it is a multi-user system.

3.1.3. Software Interfaces

a) MS-Windows Operating System (XP/Vista/8/10)

b) HTML and CSS for designing front end.

c) PHP and MySQL for designing the back end.

3.1.4. Communication Interfaces

Communication is via Internet as it is a web-enabled system.

3.2 Functions

<u>**USE CASE DESCRIPTION OF LOGIN**</u>

Introduction :   This use case documents the steps that must followed in order to get logged onto the system.
Actors :   Administrator, Data entry operator, Faculty, Student.
Pre – Condition :   This system must be connected to University LAN.
Post – Condition :   If the use case is successful, then the user is logged onto the system. Otherwise, the system state is unchanged.
Event Flow :   **Basic Flow :**   This use case starts when the user wishes to login to the Attendance Tracking Management system.   1. The system prompts the login screen to the user asking for username and password.   2. The user enters the username and password.   3. The system validates the username and password, and allows the user to log into the system.   **Alternate Flows:**   **Alternate Flow 1: Invalid Username or Password**   If the system does not validate the user credentials (due to incorrect username or password), then an error message is flagged and the use case returns to the beginning of basic flow.   **Alternate Flow 2: User exits**   This allows the user to exit at any time during the use case. The use case ends.
Special Requirements :   None

Associated Use Case :
None

USE CASE DESCRIPTION OF MAINTAIN LOGIN DETAILS

Introduction :   This use case documents the steps that must be followed in order to maintain the login details of the users of the system.
Actors :   Administrator, Data Entry Operator
Pre – Condition :   The user must be logged onto the system before this use case begins.
Post – Condition :   If the use case is successful, then user's login credentials are added, updated or deleted by the user. Otherwise, the system state is unchanged.

Event Flow :
 Basic Flow :
 This use case starts when the user wishes to add/delete/update user login credentials.
 1. The system requests that the user specify the function he/she would like to perform (either Add a user, Delete a user or Update a user) (new user to be added is Admin or Deo here)
 2. Once the user provides the requested information, one of the sub flows is executed.
     • If the user selects "Add a User", the **Add a User sub flow is executed**.
     • If the user selects "Delete a User", the **Delete a User sub flow is executed**.
     • If the user selects "Update a User", the **Update a User sub flow is executed**.
 	
 Basic flow 1: Add a User
 The system requests that the user enter the user's login credentials. This includes:
     • Username
     • Password
 Once the user provides the requested information the user's login credentials are added to the system.
 	
 Basic flow 2: Delete a User
 1. **The system requests the user to specify the username.**
 2. The user enters the username.
 3. The system retrieves the user's login credentials.
 4. The system prompts the user to confirm the deletion of user's login credentials.
 5. The user confirms the deletion.
 6. The system deletes the record.

Basic flow 3: Update a User
1. **The system requests the user to enter the username.**
2. The user enters the username.
3. The system retrieves the user's login credentials.
4. The user updates the username of the user as requested by the user.
5. Once the user updates the credentials, the system updates the user's login credentials with the updated credentials.

Alternate Flows:
Alternative Flow 1: Invalid Entry
If in the **Add a User or Update a User** flows, the user enters invalid data in any of the required fields or leaves any field empty, then the system displays an appropriate error message. The user returns to the beginning of the basic flow.

Alternative Flow 2: Username already exists
If in the **Add a User or Update a User** flow, a user with specified username already exists, the system displays an error message. The user returns to the beginning of the basic flow.

Alternative Flow 3: Username not found
If in the **Update a User or Delete a User** flow, the user login credentials with the username does not exist, the system displays an error message. The user returns to the beginning of the basic flow.

Alternative Flow 4: Update Cancelled
If in the **Update a User** flow, the user decides not to update the username, the update is cancelled and the basic flow is re-started from the beginning.

Alternative Flow 5: Delete Cancelled
If in the **Delete a User** flow, the user decides not to delete the user, the delete is cancelled and the basic flow is re-started from the beginning.

Alternate Flow 6: User exits
This allows the user to exit at any time during the use case. The use case ends.

Special Requirements :
None

Associated Use Case :
Login

USE CASE DESCRIPTION OF MAINTAIN STUDENTS

Introduction :

This use case documents the steps that the administrator/DEO must follow in order to maintain the student and add, update, delete and view student information.

Actors :

Administrator, Data entry operator

Pre – Condition :

The Administrator/DEO must be logged onto the system before this use case begins.

Post – Condition :

If the use case is successful, then student information is added, updated, deleted or viewed. Otherwise, the system state is unchanged.

Event Flow :

Basic Flow :

This use case starts when the Administrator/DEO wishes to add/ update/ view/ delete student information.

1. The system requests that the Administrator/DEO specify the function he/she would like to perform (either Add a student, Update a student, Delete a student or View a student).
2. Once the Administrator/DEO provides the requested information, one of the sub flows is executed.
   - If the Administrator/DEO selects "Add a Student", the **Add a Student sub flow is executed**.
   - If the Administrator/DEO selects "Update a Student", the **Update a Student sub flow is executed**.
   - If the Administrator/DEO selects "Delete a Student", the **Delete a Student sub flow is executed**.
   - If the Administrator/DEO selects "View a Student", the **View a Student sub flow is executed**.

Basic flow 1: Add a Student

The system requests the Administrator/DEO to enter the student information. This includes:
- Roll Number
- Name
- E-mail Id
- Date of Birth
- Course Code

Once the Administrator/DEO provides the requested information the student is added to the system and system generates unique Id.

Basic flow 2: Update a Student
1. **The system requests the Administrator/DEO to enter the Student ID.**

2. The Administrator/DEO enters the student id.
3. The system retrieves and displays the student information.
4. The Administrator/DEO makes the desired changes to the student information. This includes any of the information specified in the **Add a Student sub-flow**.
5. Once the Administrator/DEO updates the necessary information, the system updates the student information with the updated information.

Basic flow 3: Delete a Student
1. **The system requests the Administrator/DEO to specify the Student ID.**
2. The Administrator/DEO enters the student id. The system retrieves and displays the student information.
3. The system prompts the Administrator/DEO to confirm the deletion of the student record.
4. The Administrator/DEO confirms the deletion.
5. The system deletes the record.

Basic flow 4: View a Student
1. **The system requests the Administrator/DEO to specify the Student ID.**
2. The Administrator/DEO enters the student id.
3. The system retrieves and displays the student information.

Alternate Flows:
Alternative Flow 1: Invalid Entry
If in the **Add a Student or Update a Student** flows, the user enters invalid student roll number/name/course code/date of birth/email-id or leaves any field empty, the system displays an appropriate error message. The user returns to the basic flow.

Alternative Flow 2: Student already exists
If in the **Add a Student** flow, a Student with a specified student id already exists, the system displays an error message. The user returns to the basic flow and may re-enter the Student id.

Alternative Flow 3: Student not found
If in the **Update a Student or Delete a Student or View a Student** flow, the Student information with the specified student id does not exist, the system displays an error message. The user returns to the basic flow and may re-enter the Student id.

Alternative Flow 4: Course not found
If in the **Add a Student or Update a Student** flow, the Course information with the specified id does not exist, the system displays an error message. The user may verify and re-enter the course id.

Alternative Flow 5: Update Cancelled

If in the **Update a Student** flow, the user decides not to update the student, the update is cancelled and the basic flow is re-started at the beginning.

Alternative Flow 6: Delete Cancelled

If in the **Delete a Student** flow, the user decides not to delete the student, the delete is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 7: User exits

This allows the user to exit at any time during the use case. The use case ends.

Special Requirements :   None
Associated Use Case :   Login

USE CASE DESCRIPTION OF MAINTAIN FACULTY

Introduction :   This use case documents the steps that the administrator/DEO must follow in order to maintain the faculty and add, update, delete and view faculty information.
Actors :   Administrator, Data entry operator
Pre – Condition :   The Administrator/DEO must be logged onto the system before this use case begins.
Post – Condition :   If the use case is successful, then faculty information is added, updated, deleted or viewed. Otherwise, the system state is unchanged.
Event Flow :   **Basic Flow :**   This use case starts when the Administrator/DEO wishes to add/update/view/delete faculty information.   1. The system requests that the Administrator/DEO specify the function he/she would like to perform (either Add a Faculty, Update a Faculty, Delete a Faculty or View a Faculty).   2. Once the Administrator/DEO provides the requested information, one of the sub flows is executed.   • If the Administrator/DEO selects "Add a Faculty", the **Add a Faculty sub flow is executed**.   • If the Administrator/DEO selects "Update a Faculty", the **Update a Faculty sub flow is executed**.

- If the Administrator/DEO selects "Delete a Faculty", the **Delete a Faculty sub flow is executed**.
- If the Administrator/DEO selects "View a Faculty", the **View a Faculty sub flow is executed**.

Basic flow 1: Add a Faculty

The system requests that the Administrator/DEO enter the course information. This includes:
- Name
- Address
- Qualification
- E-mail Id
- Date of Joining
- Course Code

Once the Administrator/DEO provides the requested information the faculty is added to the system and system generates unique Id.

Basic flow 2: Update a Faculty
1. **The system requests the Administrator/DEO to enter the Faculty ID.**
2. The Administrator/DEO enters the faculty id.
3. The system retrieves and displays the faculty information.
4. The Administrator/DEO makes the desired changes to the faculty information. This includes any of the information specified in the **Add a Faculty sub-flow**.
5. Once the Administrator/DEO updates the necessary information, the system updates the faculty information with the updated information.

Basic flow 3: Delete a Faculty
1. **The system requests the Administrator/DEO to specify the Faculty ID.**
2. The Administrator/DEO enters the faculty id.
3. The system retrieves and displays the faculty information.
4. The system prompts the Administrator/DEO to confirm the deletion of the faculty record.
5. The Administrator/DEO confirms the deletion.
6. The system deletes the record.

Basic flow 4: View a Faculty
1. **The system requests the Administrator/DEO to specify the Faculty ID.**
2. The Administrator/DEO enters the faculty id.
3. The system retrieves and displays the faculty information.

Alternate Flows:
Alternative Flow 1: Invalid Entry

If in the **Add a Faculty or Update a Faculty** flows, the user enters invalid faculty name/address/email-id/course code/qualification/date of joining or leaves any field empty, the system displays an appropriate error message. The user returns to the basic flow.

Alternative Flow 2: Faculty already exists
If in the **Add a Faculty** flow, a faculty with a specified faculty id already exists, the system displays an error message. The user returns to the basic flow and may re-enter the faculty id.

Alternative Flow 3: Faculty not found
If in the **Update a Faculty or Delete a Faculty or View a Faculty** flow, the faculty information with the specified faculty id does not exist, the system displays an error message. The user returns to the basic flow and may re-enter the faculty id.

Alternative Flow 4: Course not found
If in the **Add a Faculty or Update a Faculty** flow, the course information with the specified code does not exist, the system displays an error message. The user may verify and re-enter the course code.

Alternative Flow 5: Update Cancelled
If in the **Update a Faculty** flow, the user decides not to update the faculty, the update is cancelled and the basic flow is re-started at the beginning.

Alternative Flow 6: Delete Cancelled
If in the **Delete a Faculty** flow, the user decides not to delete the faculty, the delete is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 7: User exits
This allows the user to exit at any time during the use case. The use case ends.

Special Requirements :   None
Associated Use Case :   Login

USE CASE DESCRIPTION OF MAINTAIN COURSES

Introduction :   This use case documents the steps that the administrator/DEO must follow in order to maintain the course details and add, update, delete and view course information.
Actors :   Administrator, Data entry operator

Pre – Condition :
The Administrator/DEO must be logged onto the system before this use case begins.
Post – Condition :
If the use case is successful, then course information is added, updated, deleted or viewed. Otherwise, the system state is unchanged.
Event Flow :   **Basic Flow :**   This use case starts when the Administrator/DEO wishes to add/update/view/delete course information.   1. The system requests that the Administrator/DEO specify the function he/she would like to perform (either Add a course, Update a course, Delete a course or View a course).   2. Once the Administrator/DEO provides the requested information, one of the sub flows is executed.   - If the Administrator/DEO selects "Add a Course", the **Add a Course sub flow is executed**.   - If the Administrator/DEO selects "Update a Course", the **Update a Course sub flow is executed**.   - If the Administrator/DEO selects "Delete a Course", the **Delete a Course sub flow is executed**.   - If the Administrator/DEO selects "View a Course", the **View a Course sub flow is executed**.   **Basic flow 1: Add a Course**   The system requests that the Administrator/DEO enter the course information. This includes: - Course Name   - Course Code   - Semester   - Credit points   Once the Administrator/DEO provides the requested information the course is added to the system and system generates unique Id.   **Basic flow 2: Update a Course**   1. **The system requests the Administrator/DEO to enter the Course ID.**   2. The Administrator/DEO enters the course id.   3. The system retrieves and displays the course information.   4. The Administrator/DEO makes the desired changes to the course information. This includes any of the information specified in the **Add a course sub-flow**.   5. Once the Administrator/DEO updates the necessary information, the system updates the course information with the updated information.

Basic flow 3: Delete a Course
1. **The system requests the Administrator/DEO to specify the Course ID.**
2. The Administrator/DEO enters the course id. The system retrieves and displays the course information.
3. The system prompts the Administrator/DEO to confirm the deletion of the course record.
4. The Administrator/DEO verifies the deletion.
5. The system deletes the record.

Basic flow 4: View a Course
1. **The system requests the Administrator/DEO to specify the Course ID.**
2. The Administrator/DEO enters the course id.
3. The system retrieves and displays the course information.

Alternate Flows:
Alternative Flow 1: Invalid Entry
If in the **Add a Course or Update a Course** flows, the user enters invalid course code/course name/semester/credit or leaves any field empty, the system displays an appropriate error message. The user returns to the basic flow and may re-enter the invalid entry.

Alternative Flow 2: Course already exists
If in the **Add a Course** flow, a course with a specified code already exists, the system displays an error message. The user returns to the basic flow and may re-enter the Course id.

Alternative Flow 3: Course not found
If in the **Update a Course or Delete a Course or View a Course** flow, the Course information with the specified code does not exist, the system displays an error message. The user returns to the basic flow and may re-enter the course code.

Alternative Flow 4: Update Cancelled
If in the **Update a Course** flow, the user decides not to update the course, the update is cancelled and the basic flow is re-started at the beginning.

Alternative Flow 5: Delete Cancelled
If in the **Delete a Course** flow, the user decides not to delete the course, the delete is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 6: User exits
This allows the user to exit at any time during the use case. The use case ends.

Special Requirements :

None
Associated Use Case : Login

USE CASE DESCRIPTION OF MAINTAIN ATTENDANCE

Introduction : This use case documents the steps that the administrator/faculty must follow in order to maintain the course details and add, update and delete attendance information.
Actors : Administrator, Faculty
Pre – Condition : The Administrator/Faculty must be logged onto the system.
Post – Condition : If the use case is successful, then course information is added, updated or deleted. Otherwise, the system state is unchanged.

Event Flow :
Basic Flow :
This use case starts when the Administrator/Faculty wishes to add/update/view/delete the attendance of the students.
  1. The system requests that the Administrator/ Faculty specify the function he/she would like to perform (either Add attendance, Update attendance, or Delete attendance).
  2. Once the Administrator/ Faculty provide the requested information, one of the sub flows is executed.
  3. If the Administrator/ Faculty select "Add Attendance", the **Add Attendance sub flow is executed**.
  4. If the Administrator/ Faculty select "Update Attendance", the **Update Attendance sub flow is executed**.
  5. If the Administrator/ Faculty select "Delete Attendance", the **Delete Attendance sub flow is executed**.
 	
Basic flow 1: Add Attendance
The system requests that the Administrator/Faculty enter the Attendance. This includes:
  1. Student ID
  2. Faculty ID
  3. Date
  4. Attendance status(Present or Absent)
Once the Administrator/Faculty provides the requested information and the Attendance is added to the system.
 	
Basic flow 2: Update Attendance

1. **The system requests the Administrator/Faculty to enter the Student ID and Date.**
2. The Administrator/Faculty enters the student id and date.
3. The system retrieves and displays the attendance information.
4. The system updates the attendance of the student, mark present if absent earlier or vice versa.
5. The system prompts the Administrator/Faculty to confirm the updation of the attendance record.
6. The Administrator/Faculty verifies the updation.
7. The system updates the attendance information with the updated information.

Basic flow 3: Delete Attendance
1. **The system requests the Administrator/Faculty to specify the Student ID and Date.**
2. The Administrator/Faculty enters the student id and date. The system retrieves and displays the attendance information.
3. The system prompts the Administrator/Faculty to confirm the deletion of the attendance record.
4. The Administrator/Faculty verifies the deletion.
5. The system deletes the record.

Alternate Flows:
Alternative Flow 1: Invalid Entry
If in the **Add Attendance or Update Attendance** flows, the user enters invalid student id/faculty id/date/status or leaves any field empty; the system displays an appropriate error message. The user returns to the basic flow and may re-enter the invalid entry.

Alternative Flow 2: Attendance already exists
If in the **Add Attendance** flow, attendance with a specified student id of same date already exists, the system displays an error message. The user returns to the basic flow and may re-enter the details.

Alternative Flow 3: Attendance not found
If in the **Update Attendance or Delete Attendance** flow, the attendance information with the specified id does not exist, the system displays an error message. The user returns to the basic flow and may re-enter the details.

Alternative Flow 4: Update Cancelled
If in the **Update Attendance** flow, the user decides not to update the attendance, the update is cancelled and the basic flow is re-started at the beginning.

Alternative Flow 5: Delete Cancelled   If in the **Delete Attendance** flow, the user decides not to delete the course, the delete is cancelled and the basic flow is re-started at the beginning.   **Alternate Flow 6: User exits**   This allows the user to exit at any time during the use case. The use case ends.
Special Requirements :   None
Associated Use Case :   Login

USE CASE DESCRIPTION OF VIEW ATTENDANCE

Introduction :   This use case documents the steps that must be followed to view attendance.
Actors :   Administrator, Student, Faculty
Pre – Condition :   The Administrator/Student/Faculty must be logged onto the system.
Post – Condition :   If the use case is successful, user will be able to view attendance.
Event Flow :   **Basic flow :**   1. The user enters the student Id for which he or she has to view the attendance.   2. Then the user enters the student Id.   3. The system retrieves the information of the student's attendance.   **Alternate Flows:**   **Alternate Flow 1: Student Id is invalid**   If the system shows that the student id is invalid, then an error message is flagged and the use case ends.   **Alternate Flow 2: User exits**   This allows the user to exit at any time during the use case. The use case ends.
Special Requirements :   None
Associated Use Case :   Login

USE CASE DESCRIPTION OF REPORT GENERATION

Introduction :

This use case documents the steps that must be followed to generate one of the following report:
    1. Course List
    2. Student List
    3. Low attendance Students list

Actors : Administrator, Faculty
Pre – Condition : The Administrator/Faculty must be logged onto the system.
Post – Condition : If the use case is successful, report will be generated.

Event Flow :
Basic flow :
    1. The use case starts when user wants to generate a report.
    2. User selects one of the options.
    3. Then user enters the required information if needed (Course Id or Student Id with date).
    4. System retrieves the information.
 	
Alternate Flows:
Alternate Flow 1: Record not found
If not found, the system generates an appropriate message. The user can select another option or cancel the operation. At this point, the use case ends.
 	
Alternate Flow 2: Course Id or the Student Id is invalid
In case of first and second option, if the system shows that the student id or the course id is invalid, then an error message is flagged and the user may re-enter the id or the use case ends.
 	
Alternate Flow 3: User exits
This allows the user to exit at any time during the use case. The use case ends.

Special Requirements : None
Associated Use Case : Login

3.3 Performance requirements

The minimum hardware requirements of Attendance Tracking Management System are a 1 GigahertzCPU and 2 Gigabytes of RAM. The system must interface with the standard output device, keyboard and mouse to interact with this software.

3.4 Safety Requirements

To ensure that no one of ATMS's users loses any data while using ATMS (due to a crash or a bug of some kind) the developer team updates ATMS regularly.

3.5 Logical Database Requirements

The following information will be stored in the database:

- Admin and DEO information: Username and Password
- Courses information: Code, Name, Semester and Credit
- Faculty information: Name, Address, Email-id, Qualification, Date of Joining and Course Code
- Student information: Roll number, Name, Date of Birth, Course Code and Email-id
- Attendance information: Student id, Faculty id, Date and Status

3.6 Security Requirements

ATMS has four levels of users each with different privileges i.e.

- Administrator who is the super user and can perform most of the functions in the system such as creating users, view students and faculty list, view attendance list etc.

- Data entry Operator who enters the entry into the system. They can maintain students, faculties and courses in the system.

- Faculties who are the employees of the School. They can mark the attendance of the students and view their list.

-Students who are enrolled in the school and their attendance are managed in the system. They can only view their attendance.

3.7 Software Quality Attributes

ATMS provides the users with very simple features. Due to its well designed and easyto use interface it can be used by any users. However, users must already have a basicknowledge of using a computer.

3.8 Software System Attributes

- **Reliability:**
  The application will be available to the students throughout the registration period and have a high degree of fault tolerance.
- **Usability:**
  The application will be user-friendly and easy to operate and the functions will be easy understandable.
- **Security:**
  The application will be password protected. Users will have to enter correct login ID and password to access the application.
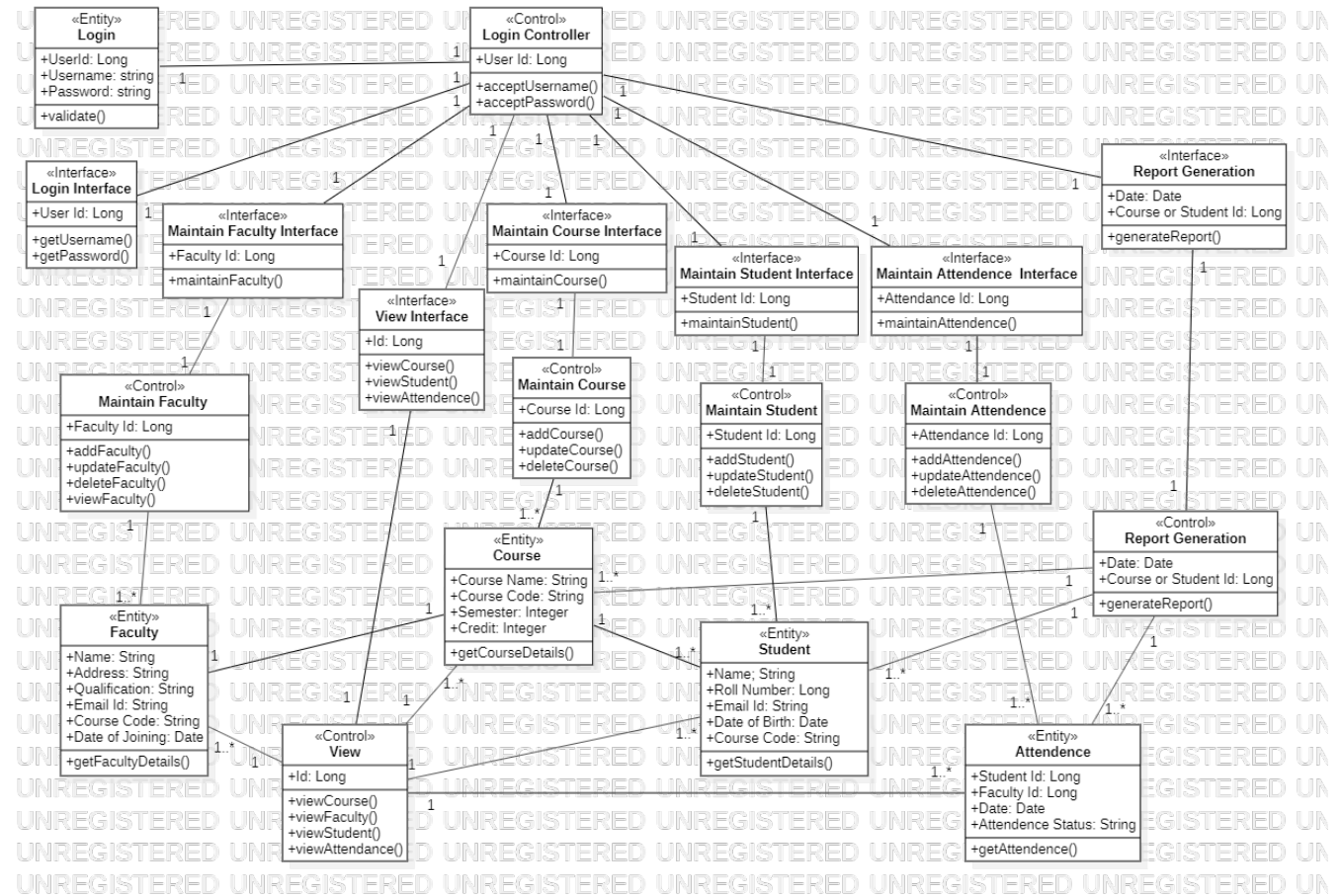- **Maintainability:**

The application will be designed in a maintainable manner. It will be easy to incorporate new requirements in individual modules.
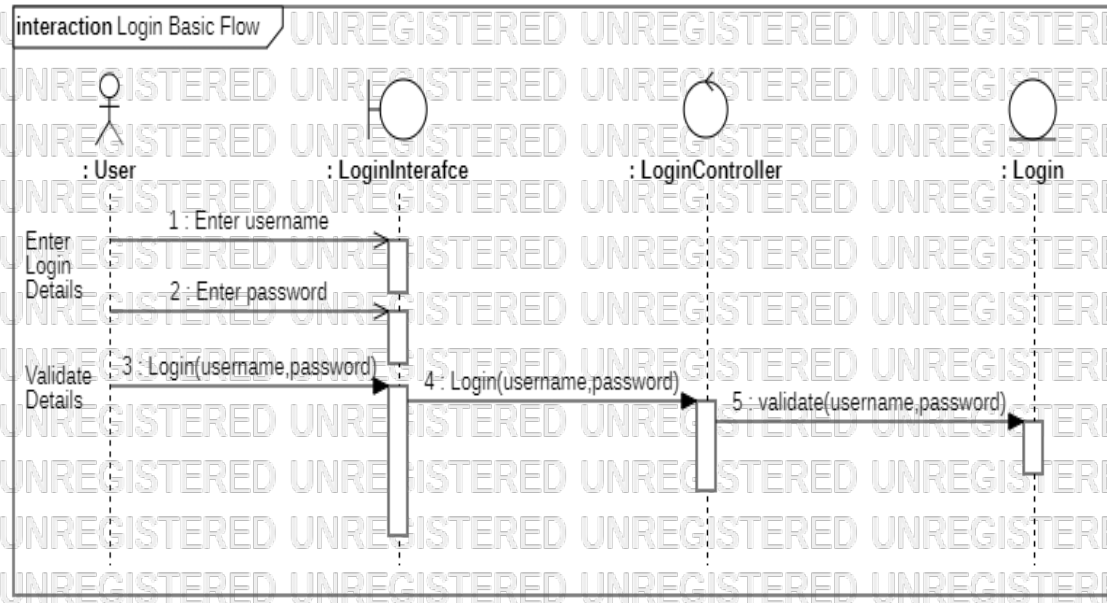
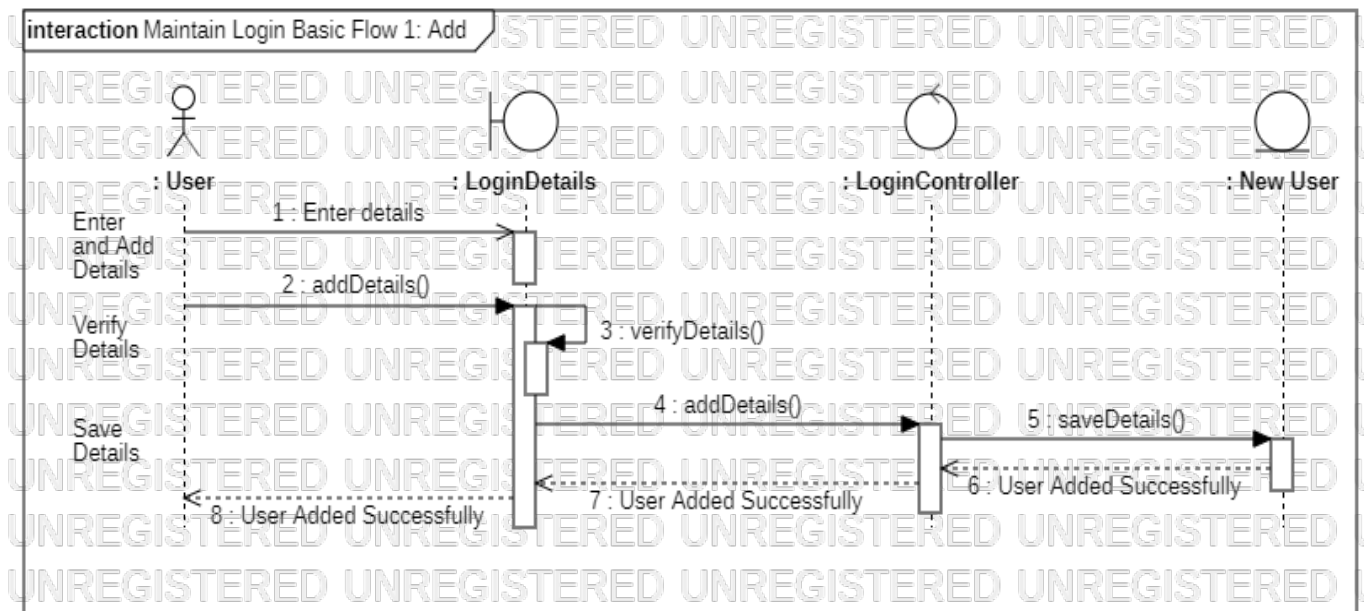## UML DIAGRAMS
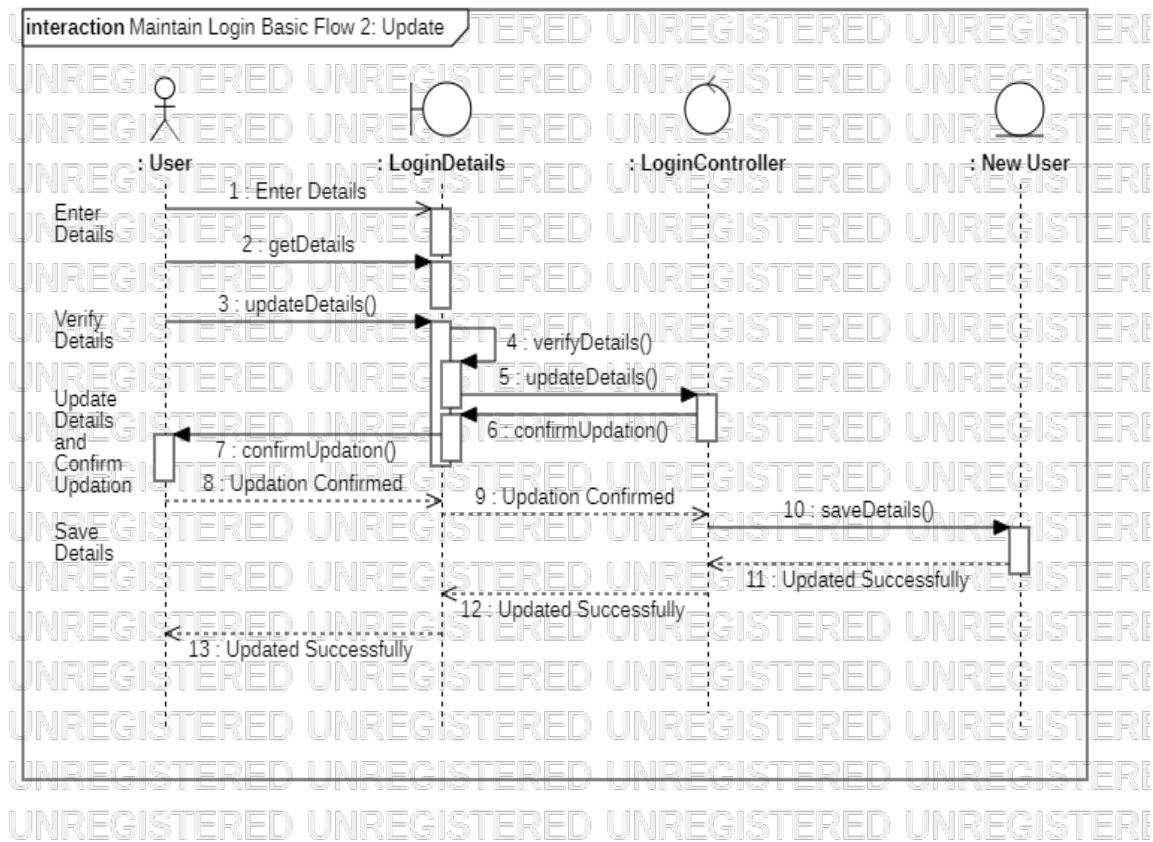
USE CASE
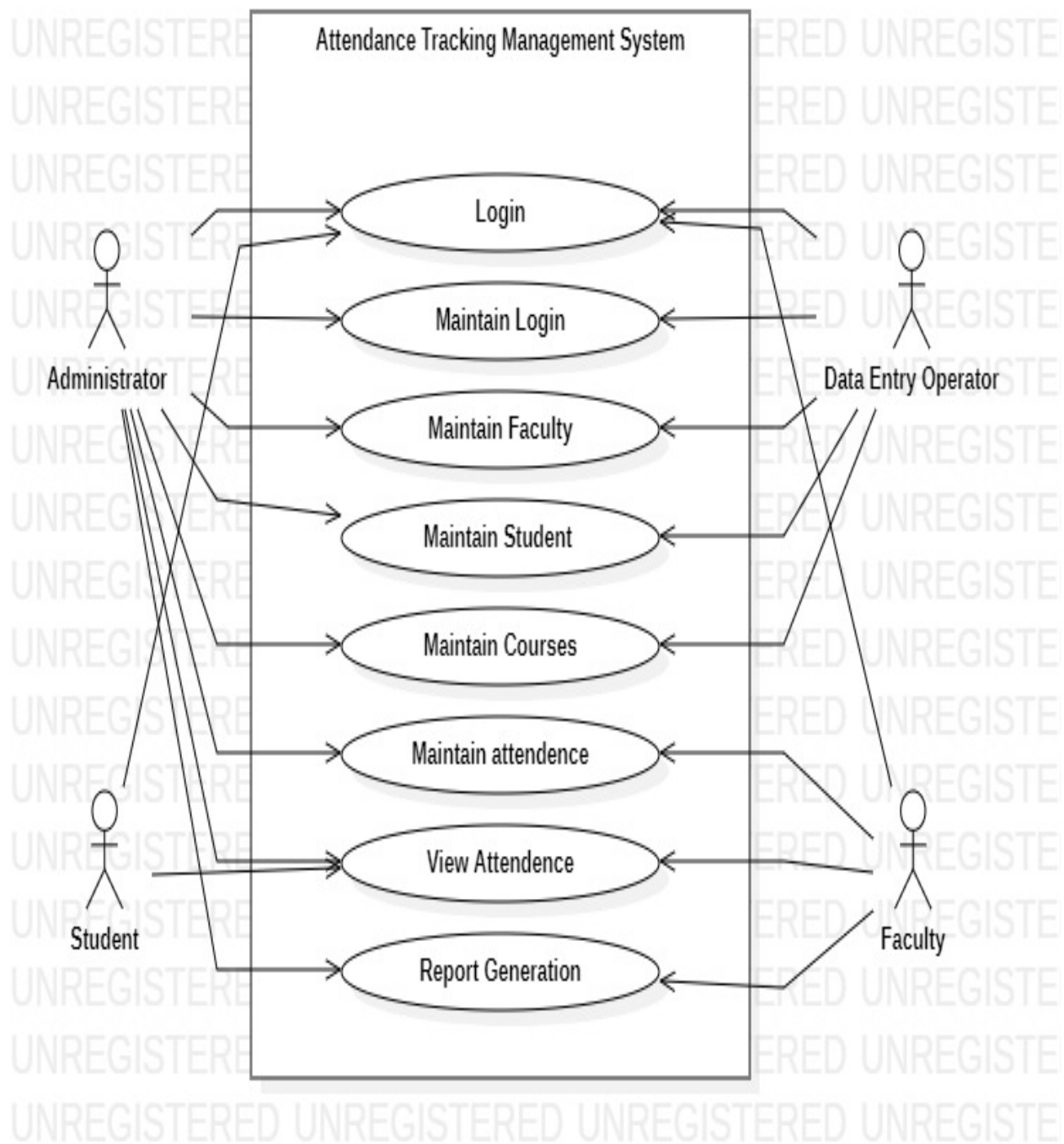
CLASS DIAGRAM

SEQUENCE DIAGRAMS

Login Basic flow

Maintain login
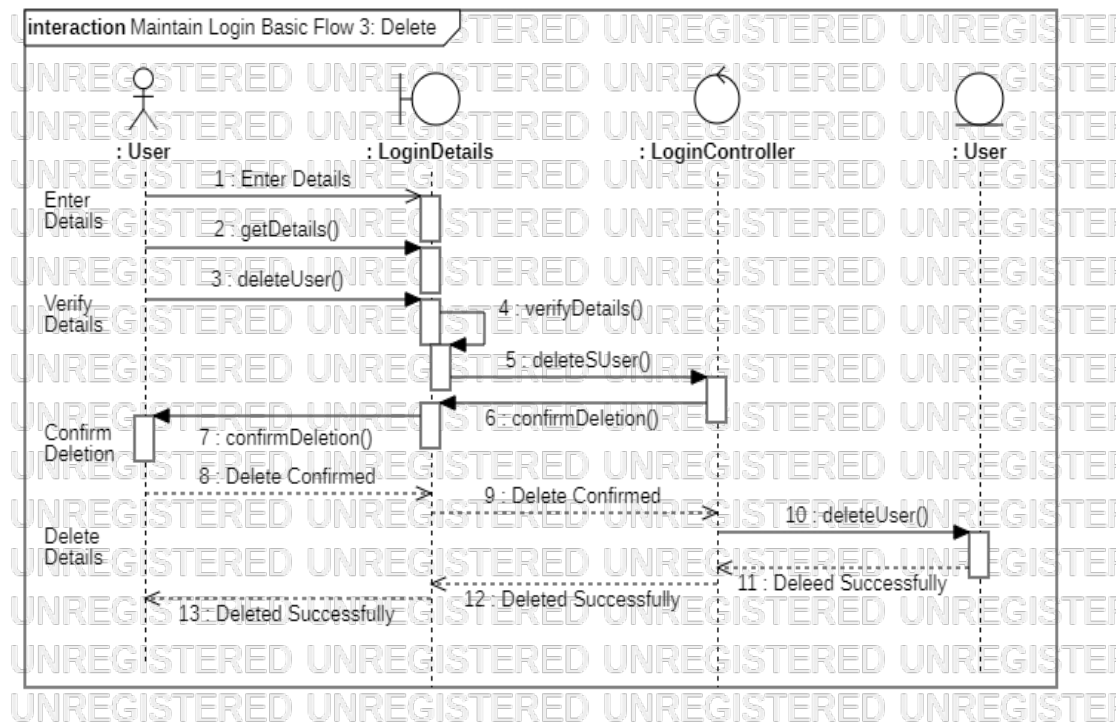### Basic flow – add

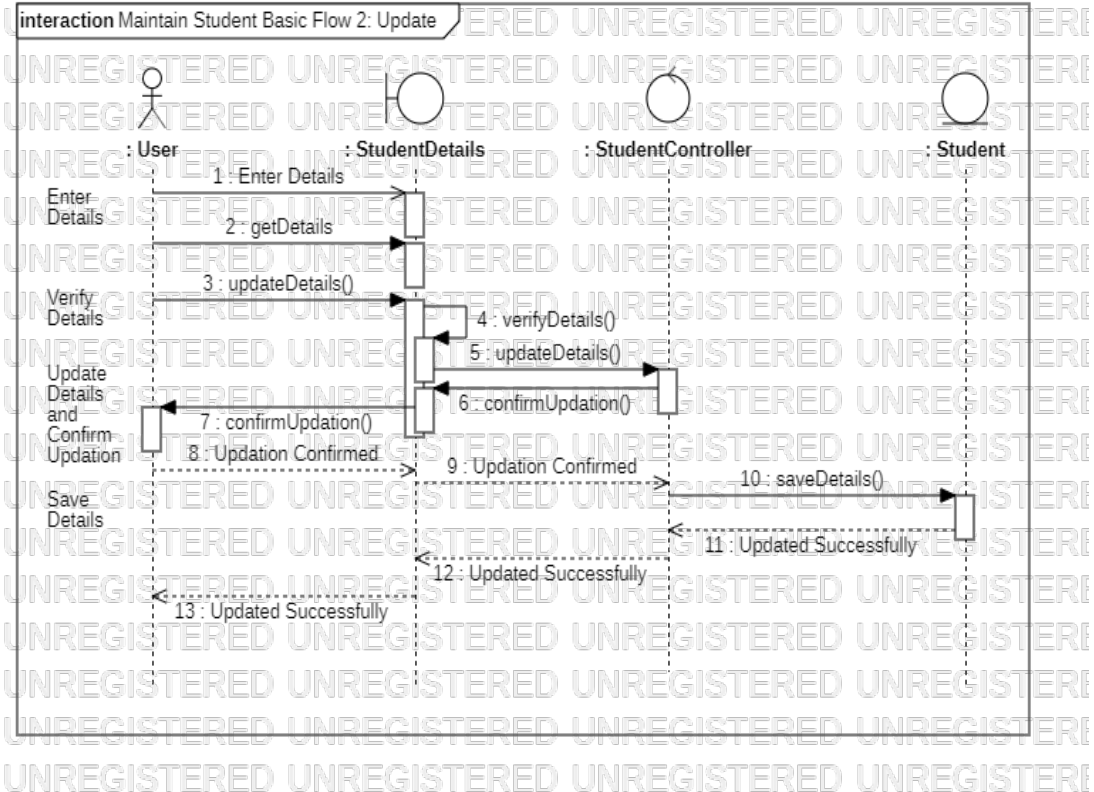# Basic flow- update

Basic flow – delete

interaction Maintain Login Alternate Flow 5: User Not Found
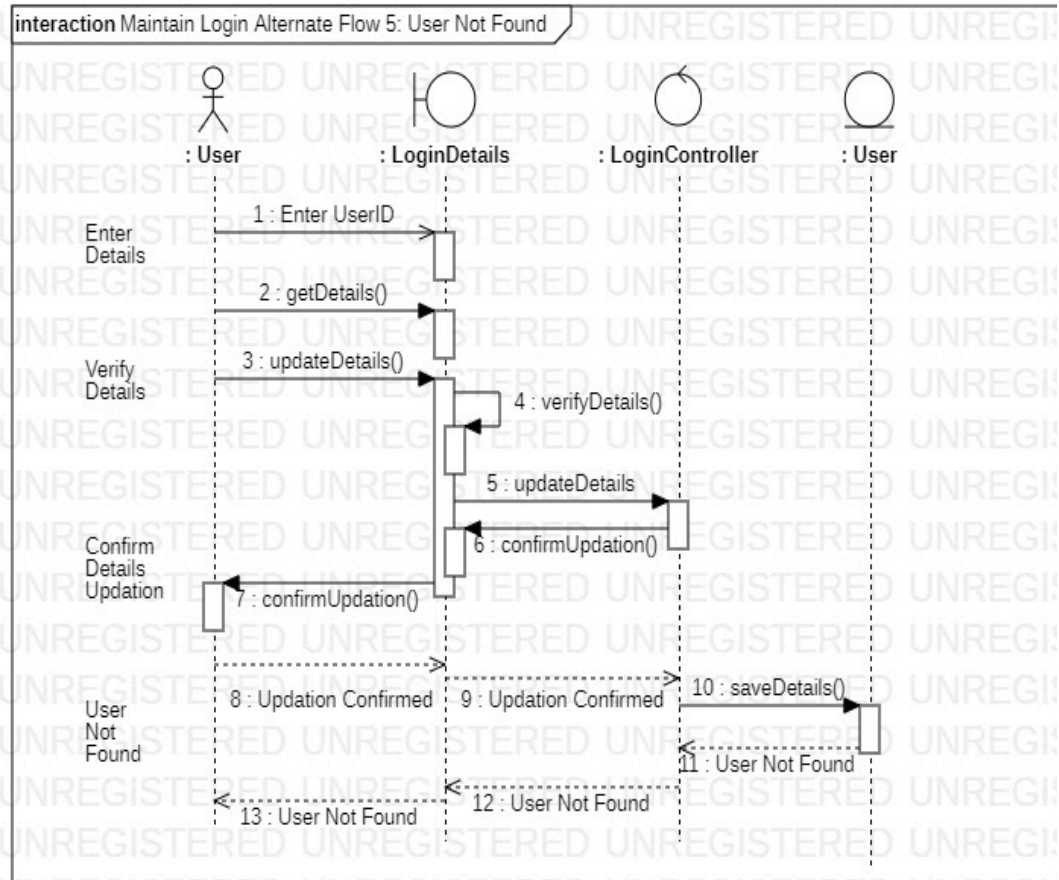
Maintain student
Basic flow – add
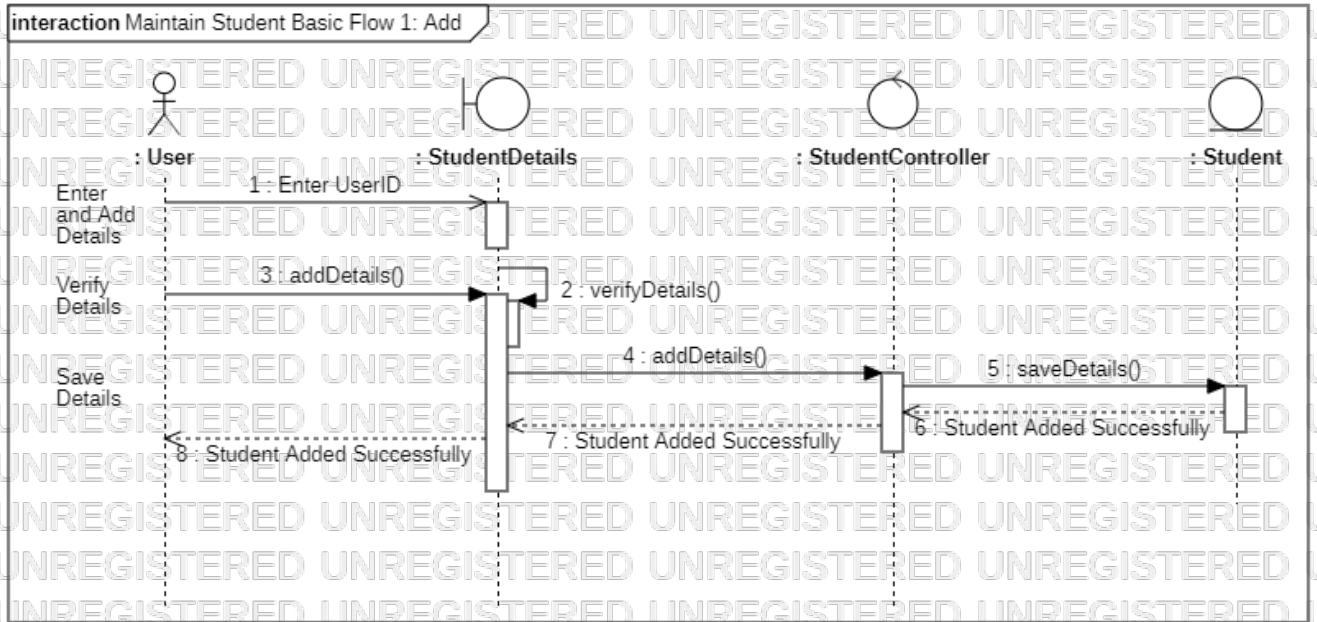
Basic flow- update
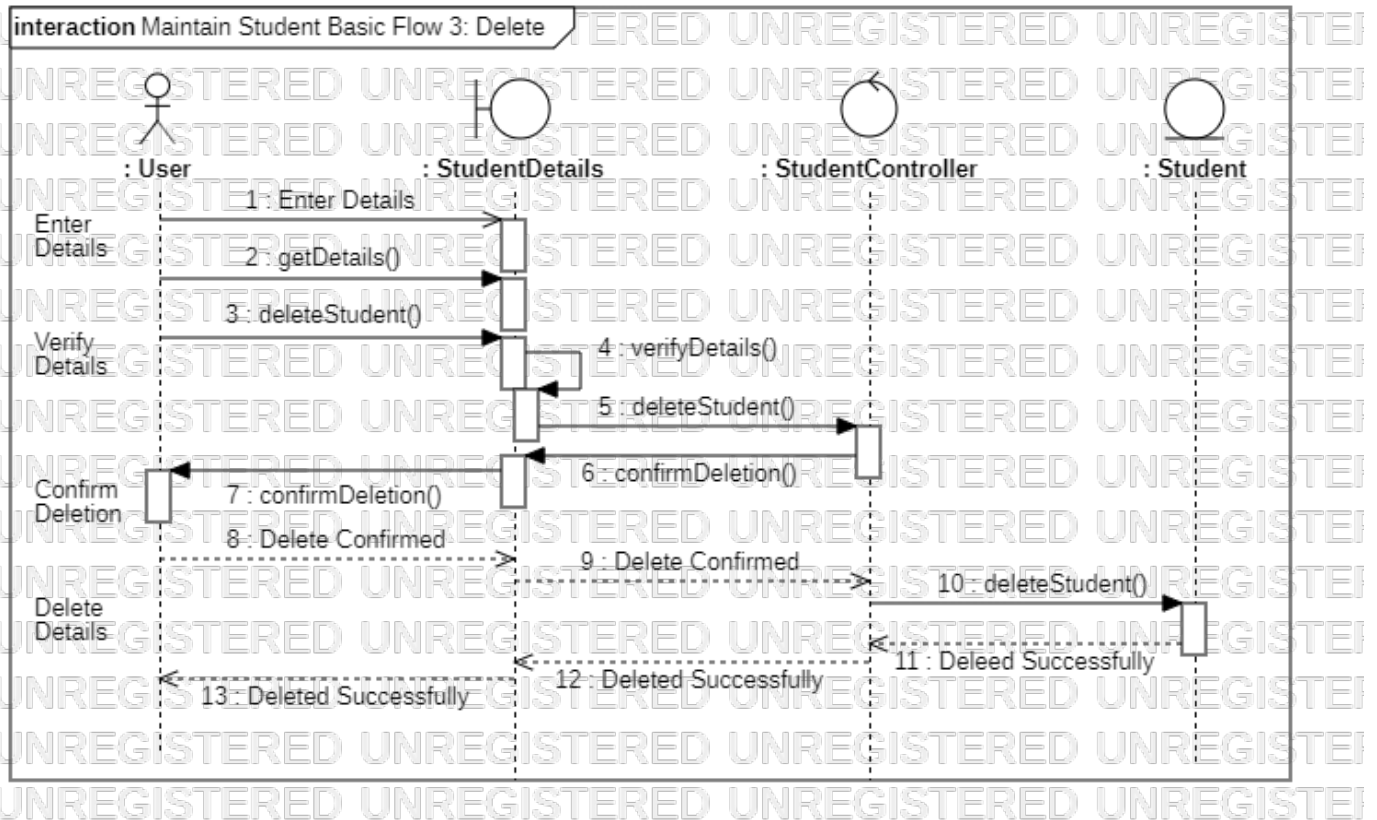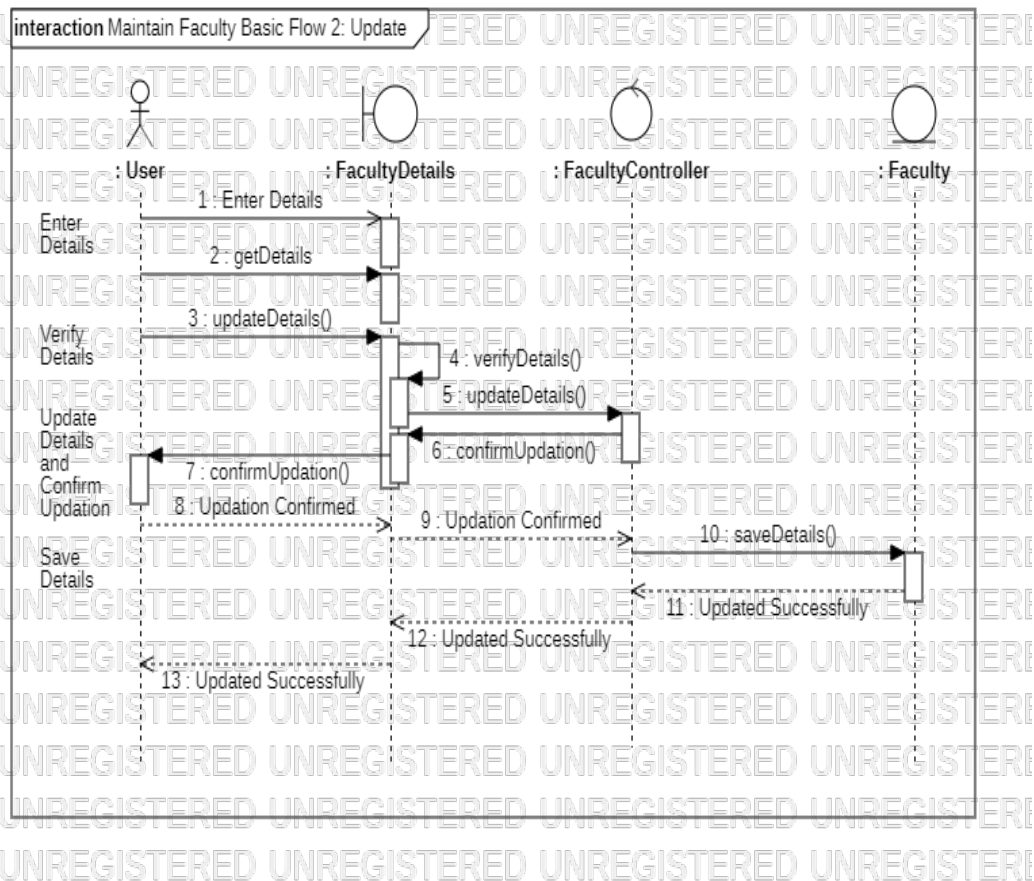

Basic flow – delete
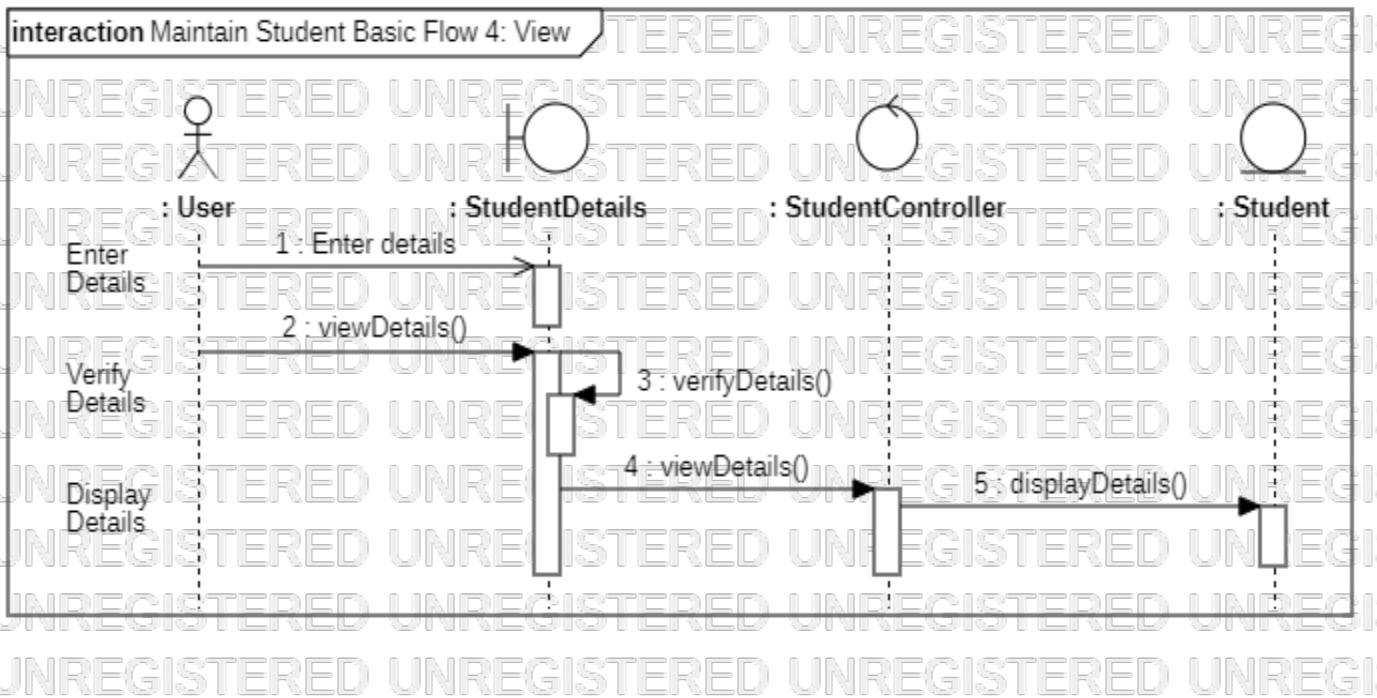
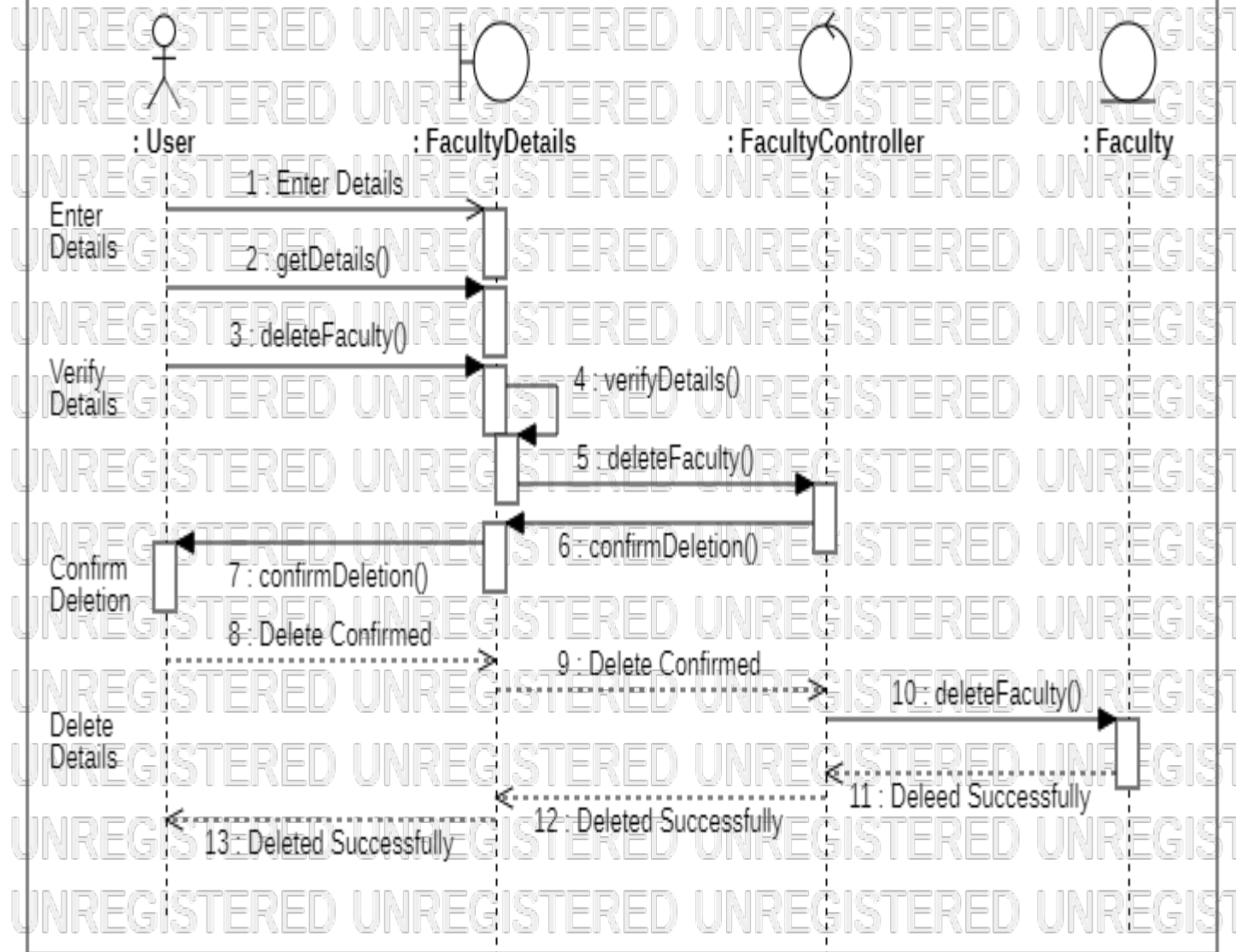Basic flow - view

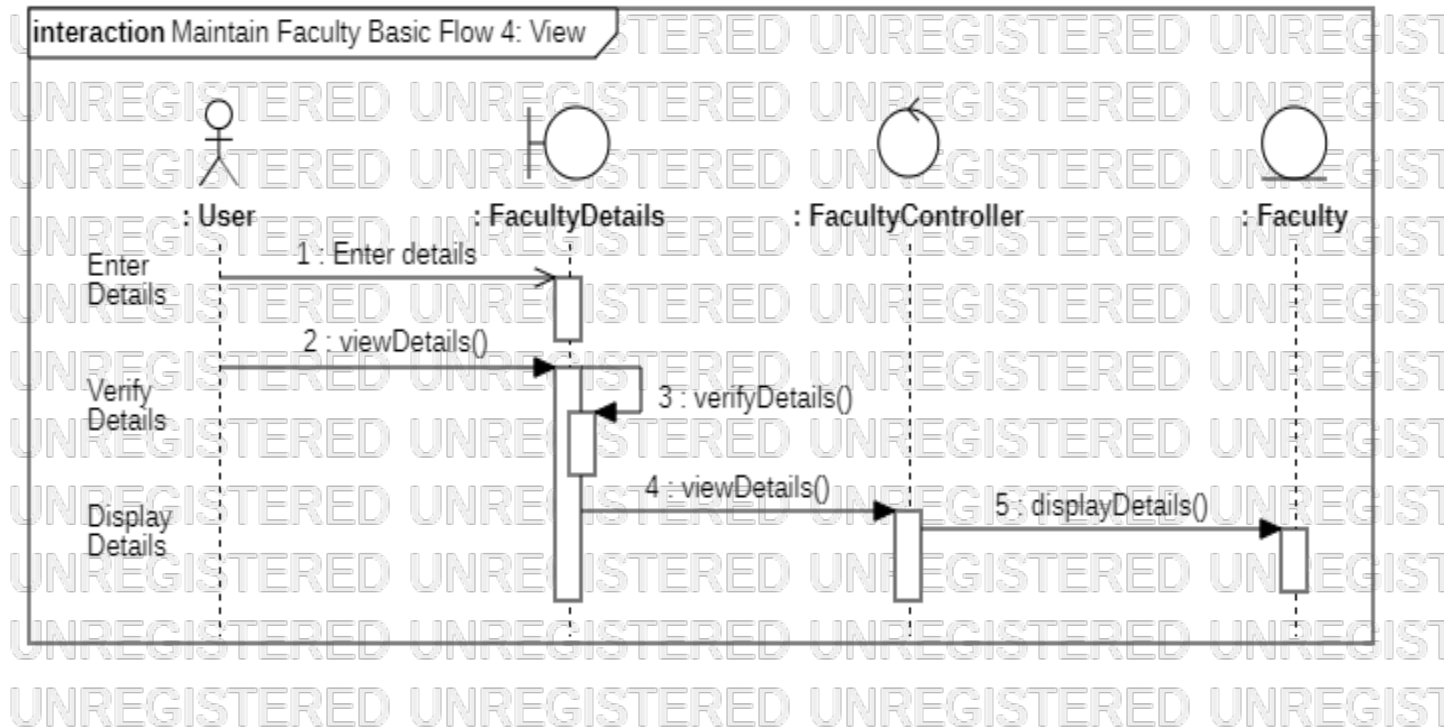**Maintain faculty**

Basic flow – add

Basic flow- update

Basic flow – delete

Basic flow - view

interaction Maintain Faculty Basic Flow 4: View

: User   : FacultyDetails   : FacultyController   : Faculty

Enter Details — 1 : Enter details

Verify Details — 2 : viewDetails()

3 : verifyDetails()

Display Details — 4 : viewDetails()   5 : displayDetails()

Maintain course

Basic flow – add

interaction Maintain Course Basic Flow 1: Add

: User   : CourseDetails   : CourseController   : Course

Enter and Add Details — 1 : Enter details

Verify Details — 2 : addDetails()

3 : verifyDetails()

Save Details — 4 : addDetails()   5 : saveDetails()

6 : Course Added Successfully

7 : Course Added Successfully

8 : Course Added Successfully

Basic flow- update

interaction Maintain Course Basic Flow 2: Update

: User   : CourseDetails   : CourseController   : Course

Enter Details
1 : Enter Details
2 : getDetails

Verify Details
3 : updateDetails()
4 : verifyDetails()
5 : updateDetails()

Update Details and Confirm Updation
6 : confirmUpdation()
7 : confirmUpdation()
8 : Updation Confirmed
9 : Updation Confirmed

Save Details
10 : saveDetails()
11 : Updated Successfully
12 : Updated Successfully
13 : Updated Successfully
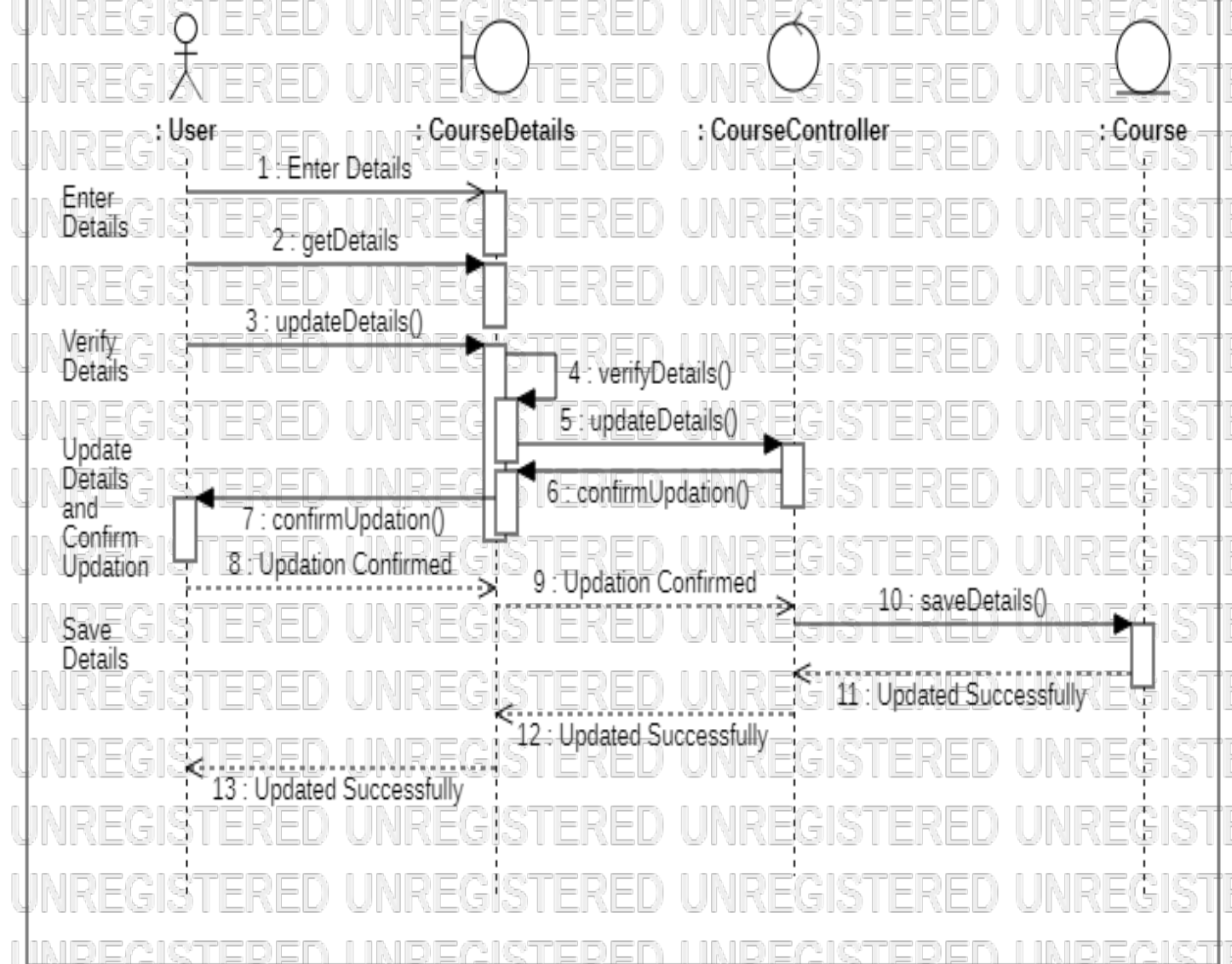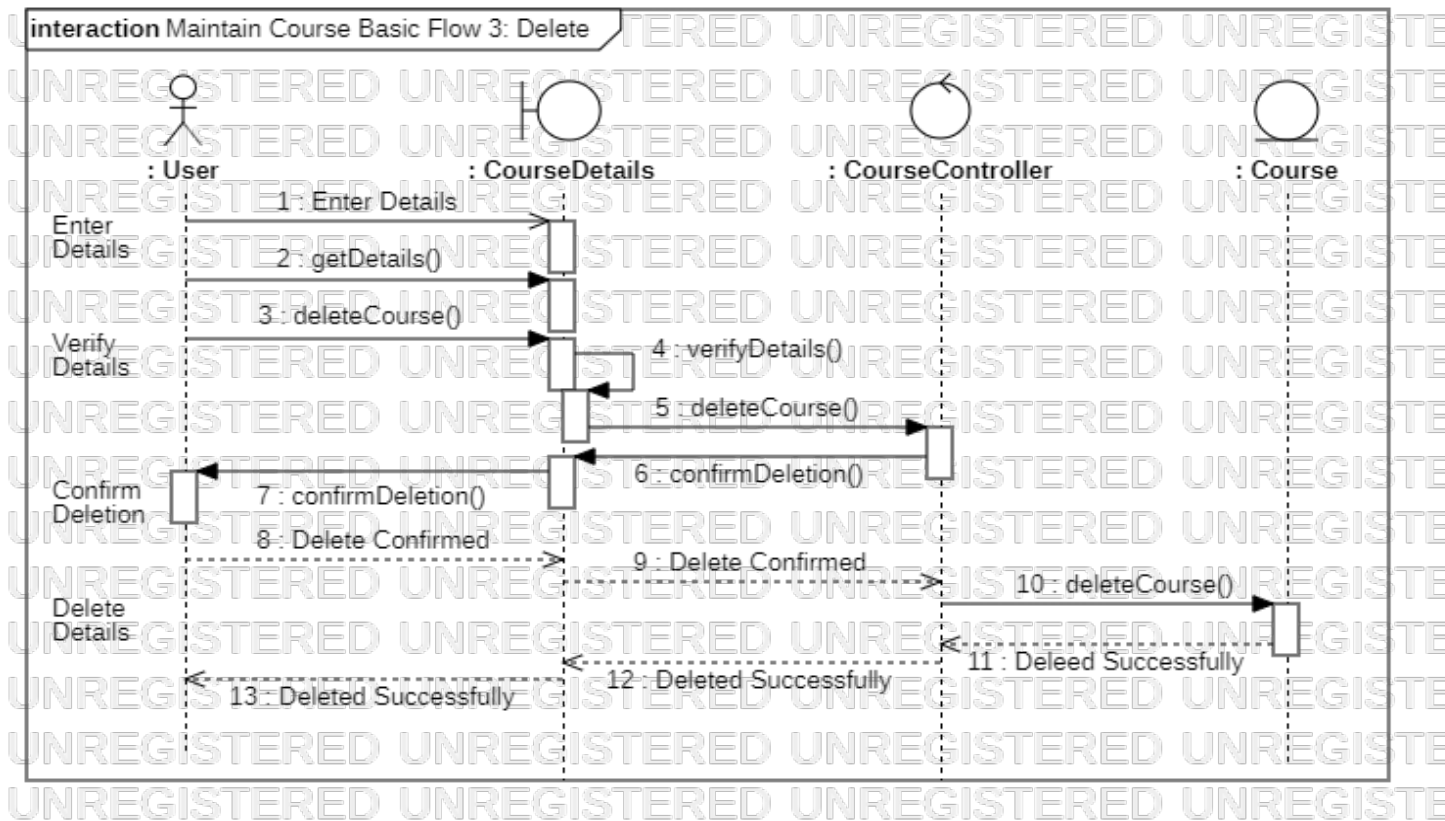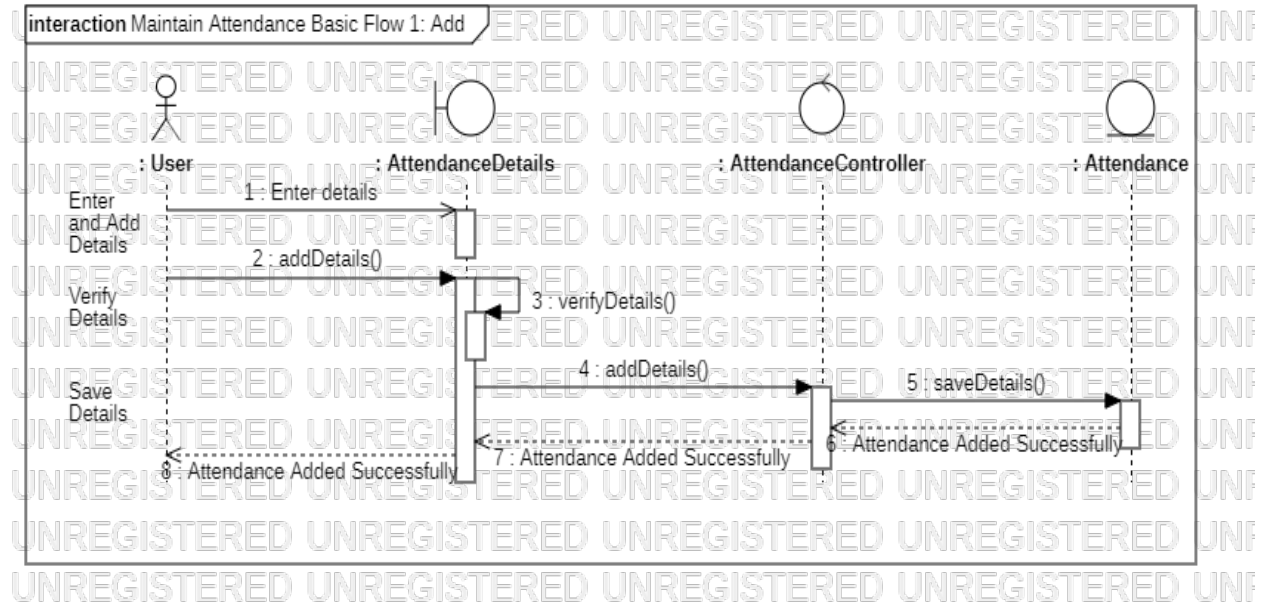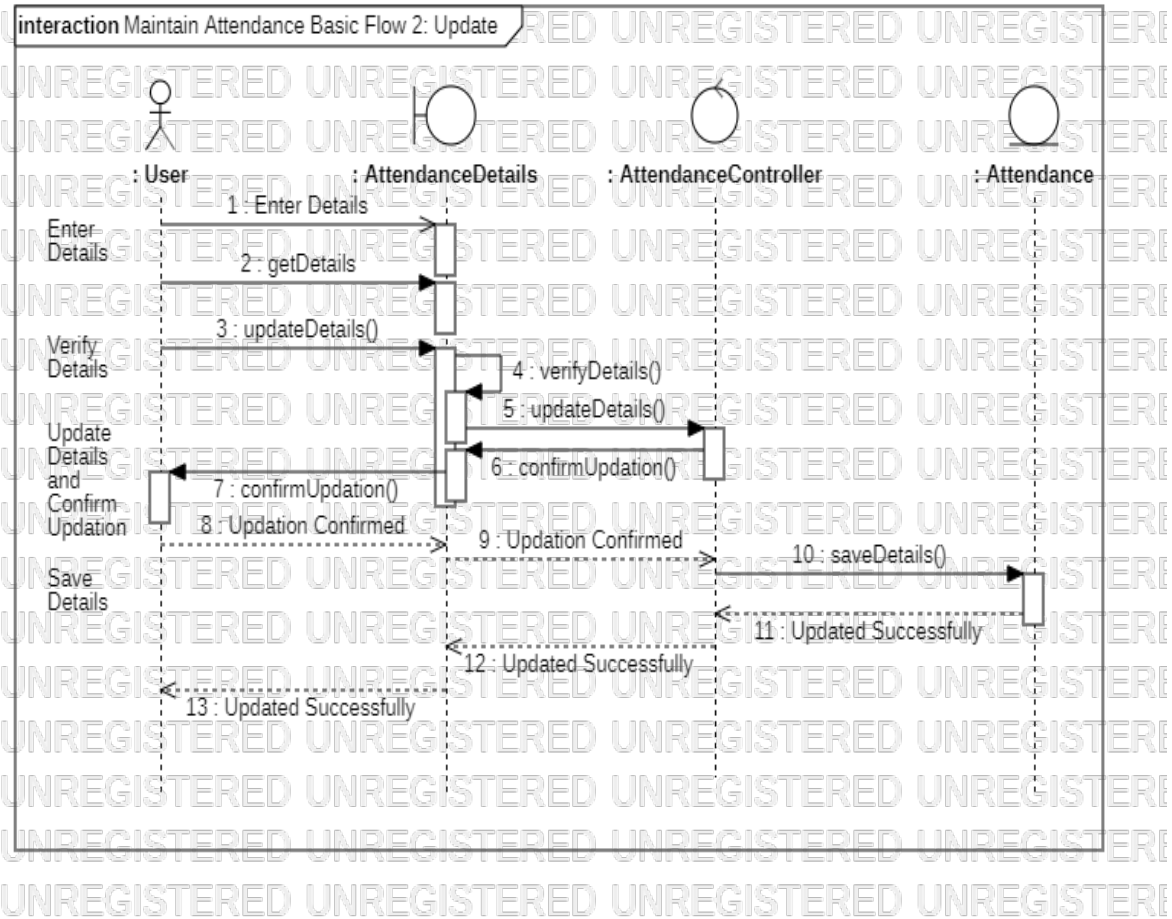
Basic flow – delete
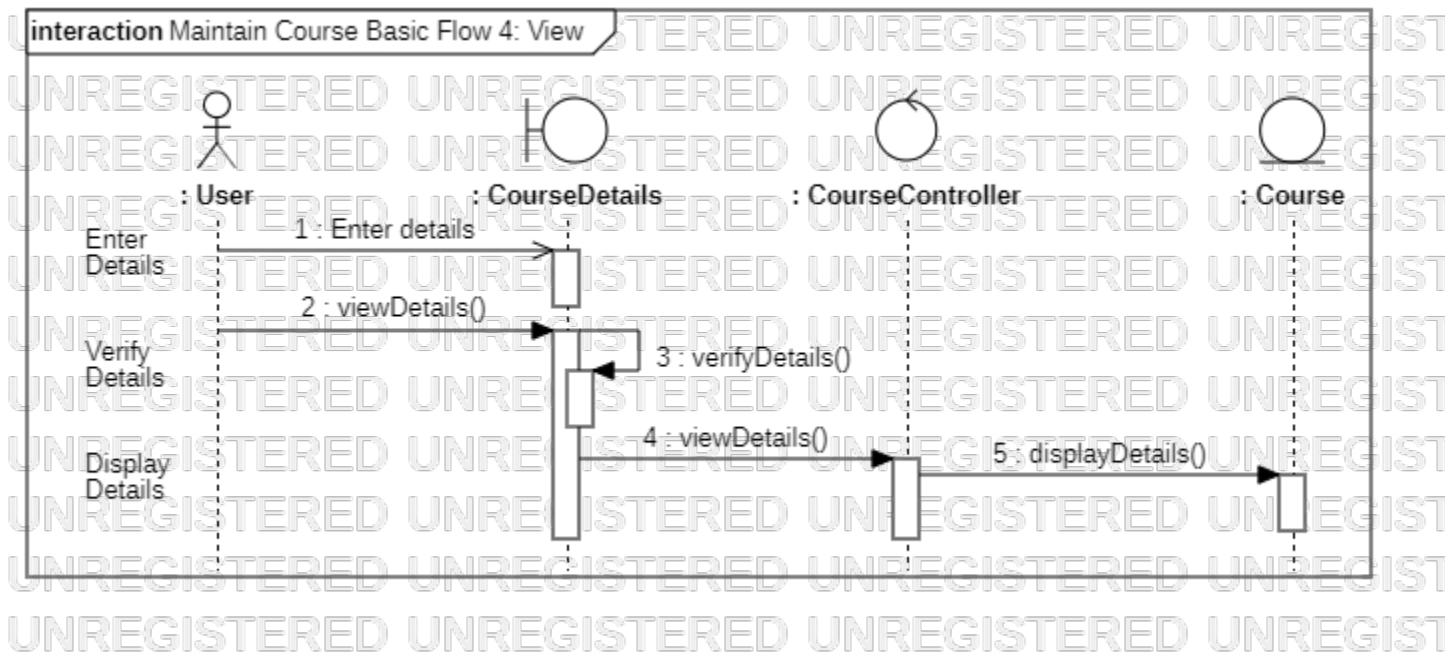
Basic flow - view
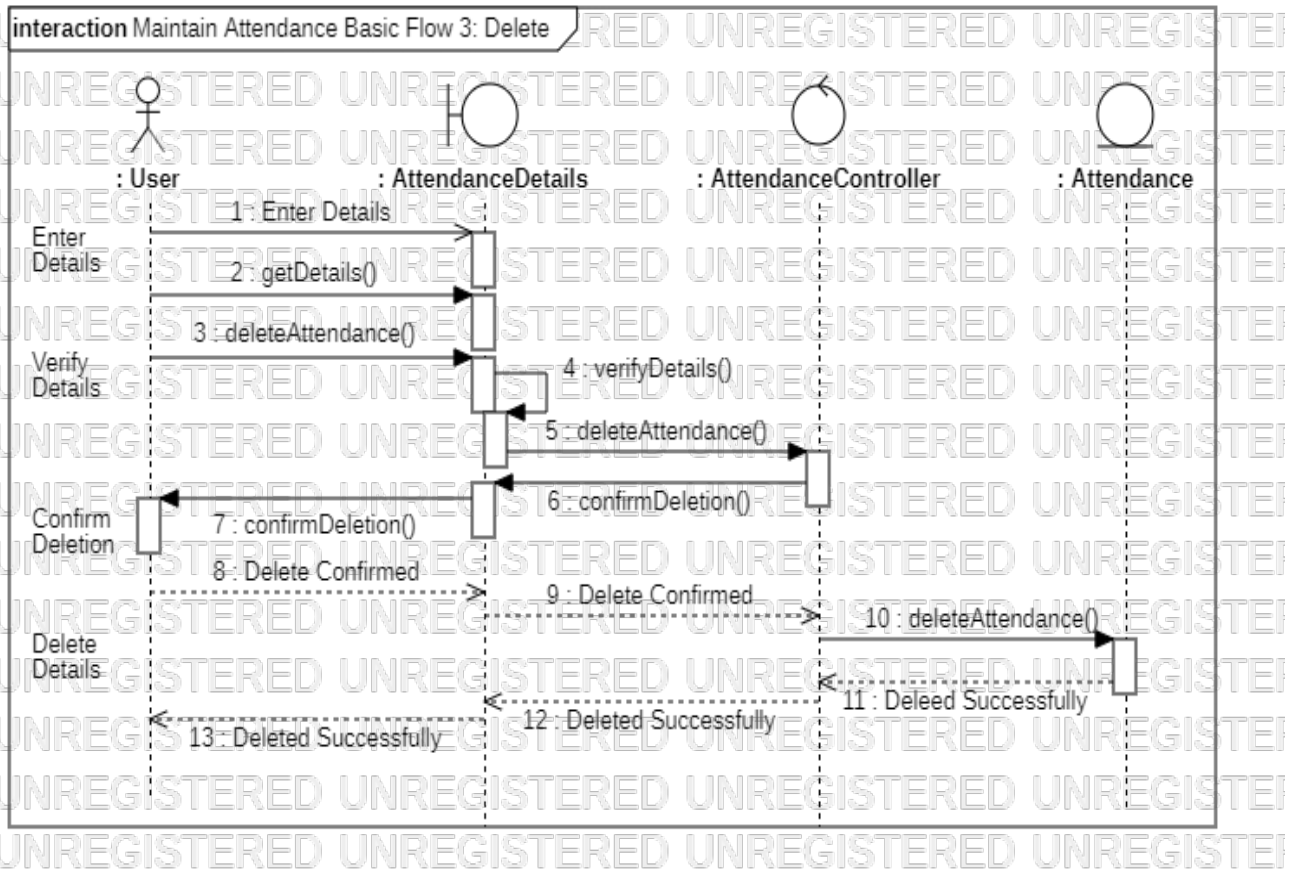
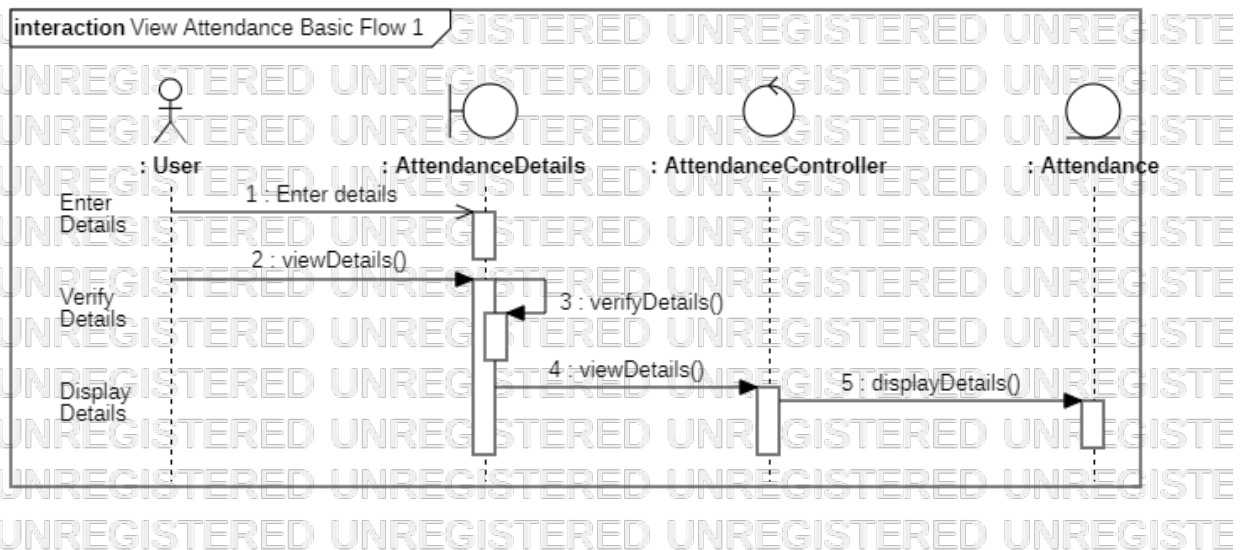# Maintain attendance
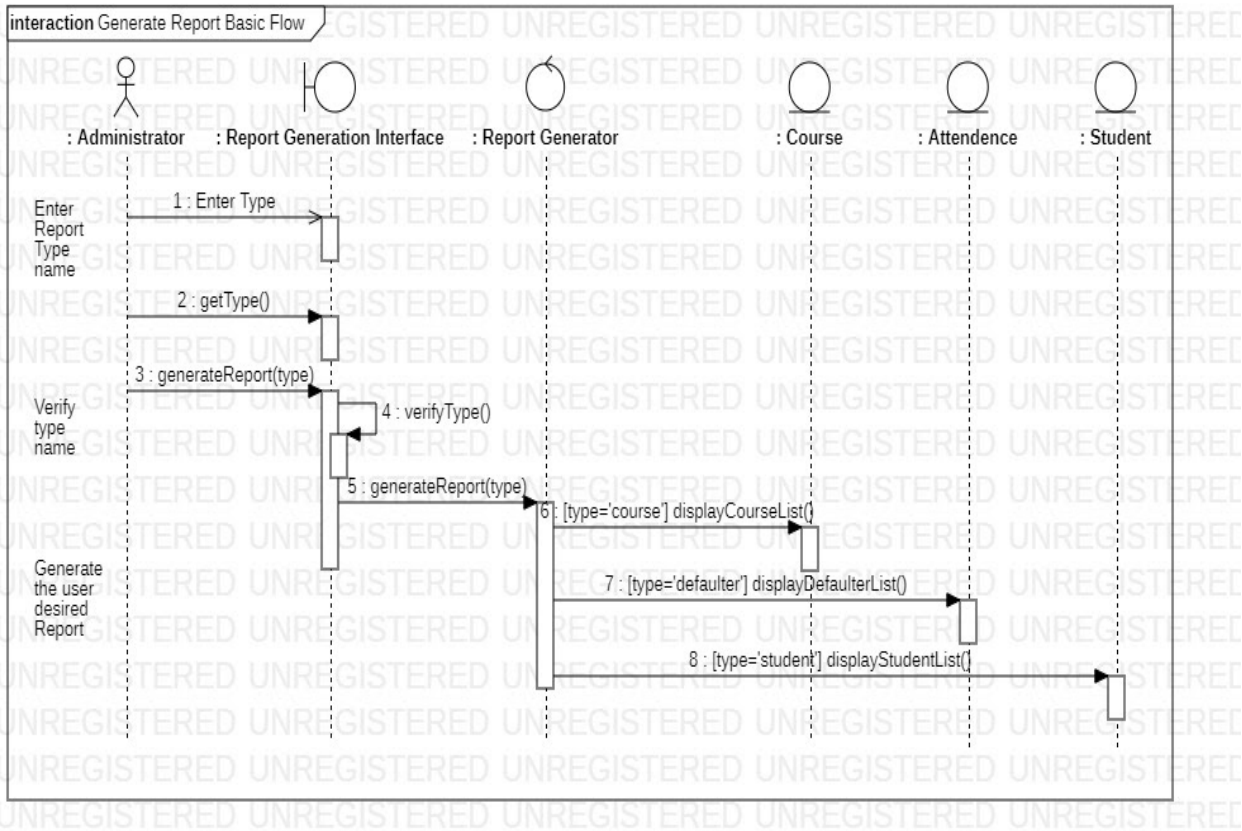## Basic flow – add

Basic flow- update

Basic flow – delete
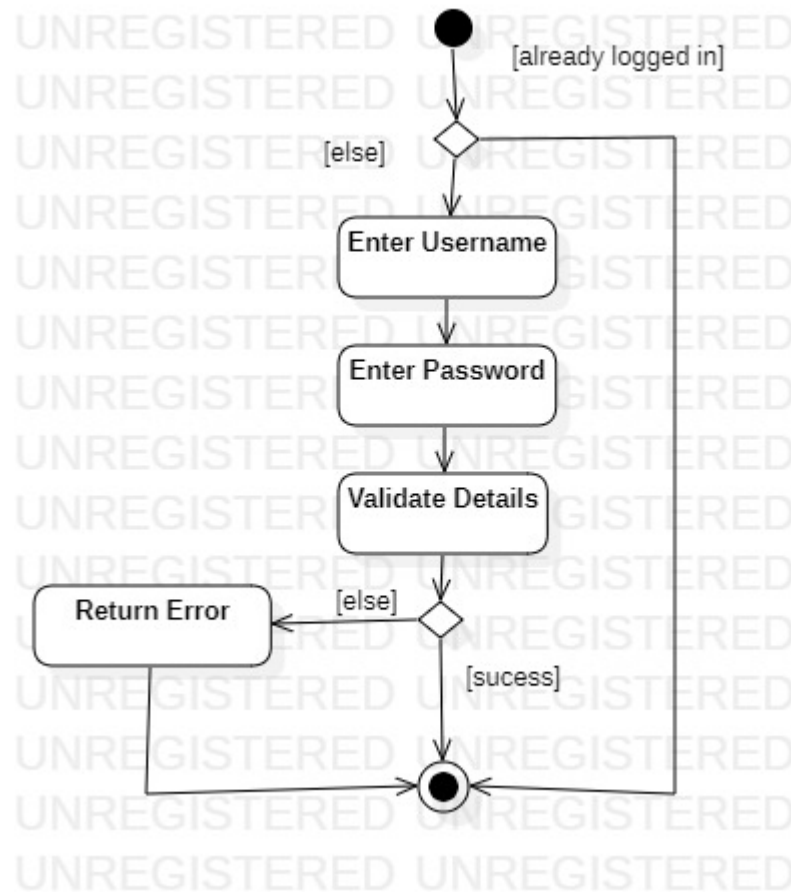

View attendance
Basic flow

Generate report
Basic flow

ACTIVITY DIAGRAM

Login

Maintain Attendance
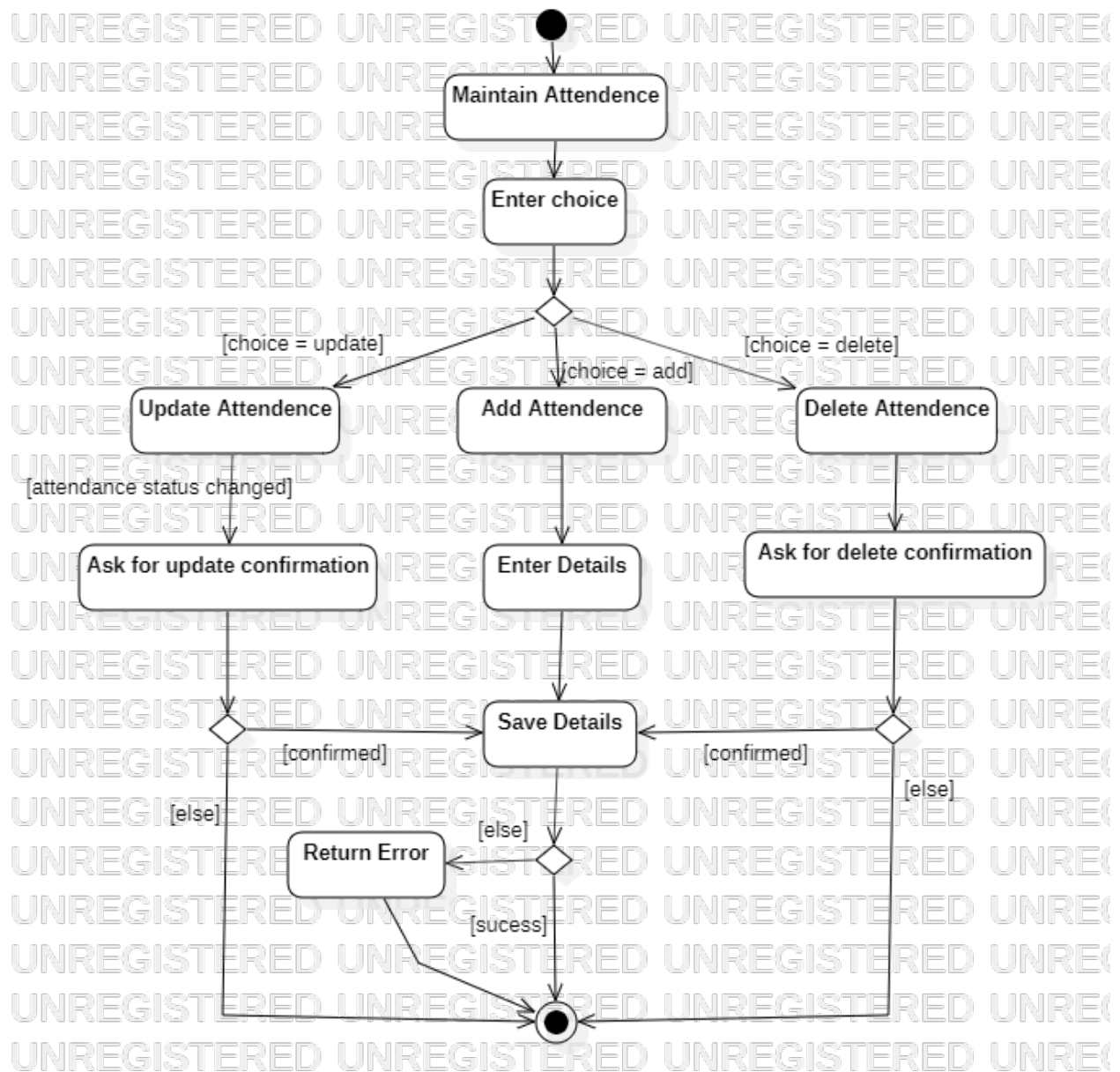
Maintain Faculty
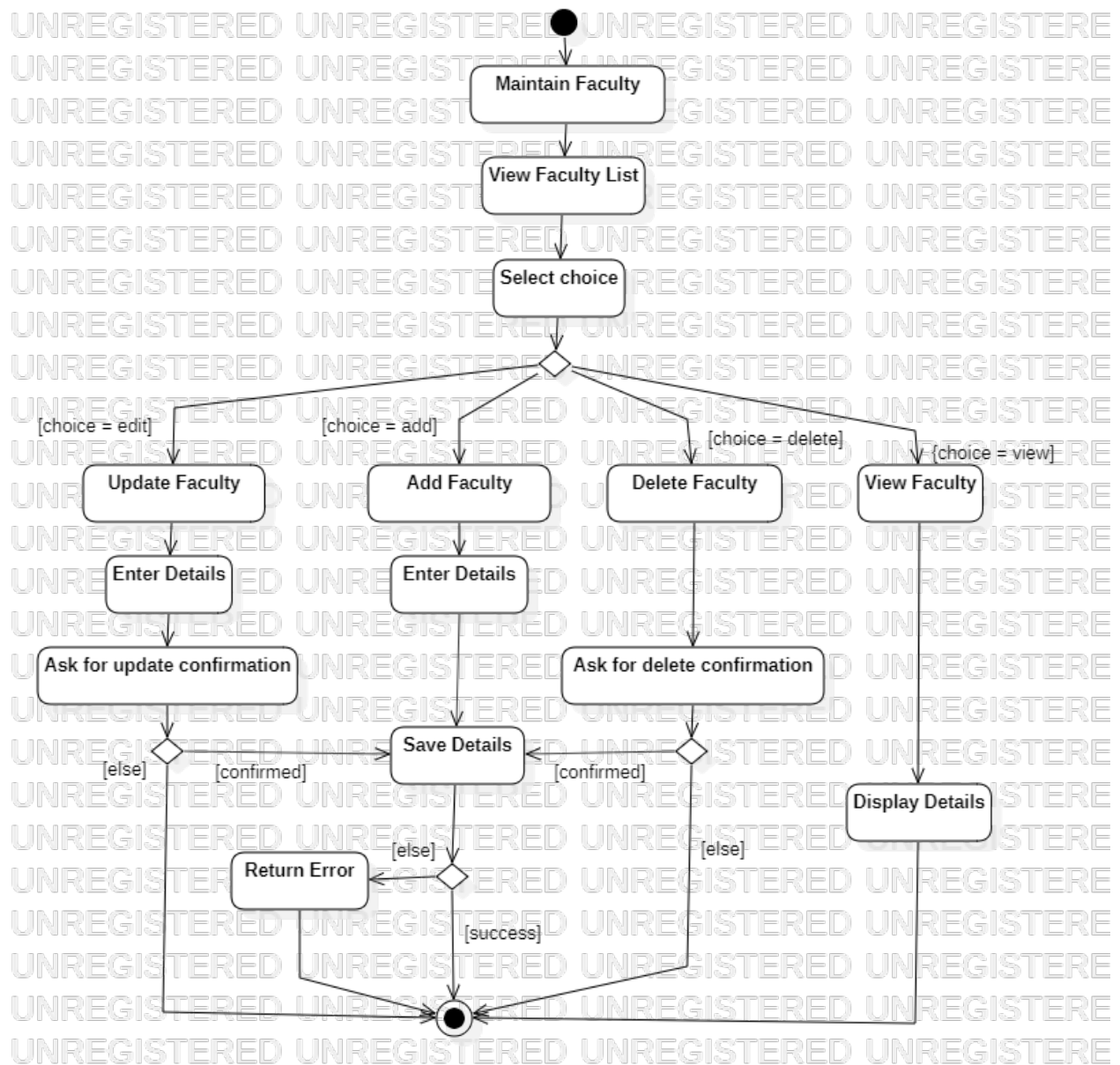
PROJECT SCREEN SHOTS AND EXECUTION DETAILS
Main page

Attendance Tracking Management System
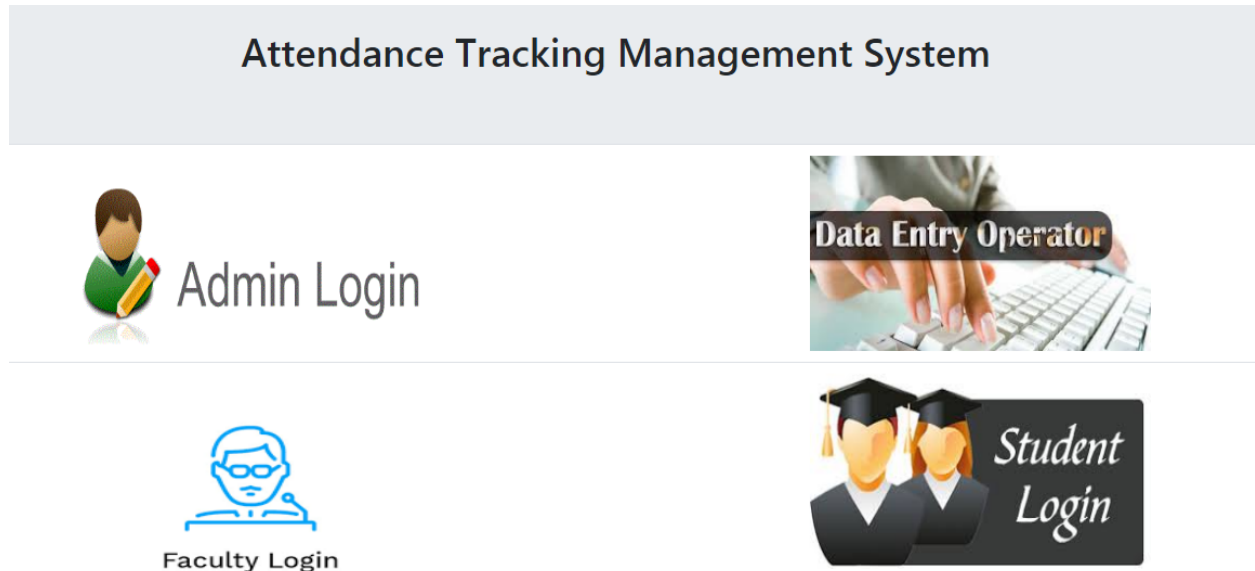
This is the main page of the system. It has four options to login into the system for four types of users – Admin, Data entry Operator, Faculty and Student.

Admin Login Page

Attendance Tracking Management System

Admin Login

Enter Username

Enter Password

Login

Admin has to enter username and password to login into the system. The system will check for empty or incorrect username or password.

Admin Home Page

This page shows the overall attendance status of all the students. It contains student's name, roll number, course, faculty name, attendance percentage. Also there is an option for pdf and chart report. PDF report asks for a from date and to date and creates a report for a particular student and attendance details of that student. Chart Report is similar to pdf report but is shows that in the form of pie chart also.

Admin Maintain Course Page

In this page, the admin can see the list of all the details of the courses and maintain the course – add, update and delete. Course has the following details – code, name, semester, and credit.

Maintain Faculty Page

Attendance Tracking Management System

Home   Course   Faculty   Student   Admin   DEO   Attendance   Defaulter   Logout

Faculty List   Add

Show 10 ⬍ entries   Search: [_____]

ID	Name	Email Address	Course	View	Edit	Delete
2	Bhupender Rana	bhupender.rana@gmail.com	SE 101	View	Edit	Delete
3	Sneha Sharma	sneha.sharma@gmail.com	SE 201	View	Edit	Delete
4	Vandana Gupta	vandana.gupta@gmail.com	SE 301	View	Edit	Delete
5	Rahul Kumar	rahul.kumar@gmail.com	SE 401	View	Edit	Delete

Showing 1 to 4 of 4 entries   Previous 1 Next

In this page, the admin can see the list of the faculty and maintain the faculty – add, view, update and delete. Faculty has the following details – faculty id, name, email id, course, qualification, address, and date of joining.

Maintain Student Page

Attendance Tracking Management System

Home   Course   Faculty   Student   Admin   DEO   Attendance   Defaulter   Logout

Student List   Add

Show 10 ⬍ entries   Search: [_____]

ID	Roll No.	Student Name	Date of Birth	Course	Email ID	Edit	Delete
1	1	Aman Sharma	2003-03-04	SE 101	aman@gmail.com	Edit	Delete
3	2	Rohit Kumar	2003-04-19	SE 101	rohit@gmail.com	Edit	Delete
4	3	Devesh Singh	2004-01-15	SE 101	devesh@gmail.com	Edit	Delete
5	4	Aakash Garg	2003-12-14	SE 101	aakash@gmail.com	Edit	Delete
6	5	Rohan Negi	2003-07-12	SE 101	rohan@gmail.com	Edit	Delete
7	1	Hardik Jain	2003-12-19	SE 201	hardik@gmail.com	Edit	Delete
8	2	Deepak Kumar	2002-12-19	SE 201	deepak@gmail.com	Edit	Delete

In this page, the admin can see the list of the student and maintain the student – add, update and delete. Student has the following details – student id, name, email id, course, and date of birth.

Maintain Admin Page

In this page, the admin can see the usernames of all the admin's present and maintain the admin – add, update, and delete. Admin has only username and password.

Maintain DEO Page

In this page, the admin can see the usernames of all the deo's present and maintain the deo – add, update, and delete. DEO has only username and password.

Maintain Attendance Page

Attendance Tracking Management System

Home   Course Faculty Student Admin DEO Attendance Defaulter Logout

Attendance List   Report Chart Add

Show 10 ⬍ entries   Search:

Roll Number ⇅	Student Name ⇅	Course ⇅	Attendance Status ⇅	Attendance Date ⇅	Faculty ⇅	Edit ⇅	Delete ⇅
1	Hardik Jain	SE 201	Present	2020-09-01	Sneha Sharma	Edit	Delete
2	Deepak Kumar	SE 201	Present	2020-09-01	Sneha Sharma	Edit	Delete
3	Mohit Garg	SE 201	Absent	2020-09-01	Sneha Sharma	Edit	Delete
4	Shreya Gupta	SE 201	Present	2020-09-01	Sneha Sharma	Edit	Delete
5	Palak Garg	SE 201	Present	2020-09-01	Sneha Sharma	Edit	Delete
1	Hardik Jain	SE 201	Absent	2020-09-02	Sneha Sharma	Edit	Delete
2	Deepak Kumar	SE 201	Present	2020-09-02	Sneha Sharma	Edit	Delete

In this page, the admin can see the details of the attendance of all the students. Admin can also make the report and chart for a particular course which contains the details of the attendance of the students enrolled in that course. Attendance List contains student roll number, name, course code, attendance status, attendance date, and faculty name. Admin can add, edit or delete the attendance.

Defaulter's Page

Attendance Tracking Management System

Home   Course Faculty Student Admin DEO Attendance Defaulter Logout

Defaulter's Student Attendance Status

Show 10 ⬍ entries   Search:

Student Name ⇅	Roll Number ⇅	Course ⇅	Faculty ⇅	Attendance Percentage ⇅
Abhishek Sethi	5	SE 401	Rahul Kumar	70%
Chetan Garg	2	SE 301	Vandana Gupta	60%
Devesh Singh	3	SE 101	Bhupender Rana	70%
Ishika Gupta	5	SE 301	Vandana Gupta	70%
Mohit Garg	3	SE 201	Sneha Sharma	58%

Showing 1 to 10 of 21 entries (filtered from 10 total entries)   Previous 1 2 3 Next

In this page, the list of students is shown whose attendance is less than 75%. It shows student name, roll number, course, and faculty name and attendance percentage.

DEO login page

DEO has to enter username and password to login into the system. The system will check for empty or incorrect username or password.

DEO home page

This page shows the options to maintain courses, facuty, students and deo. Nav bar also shows the same options. DEO can add, edit or delete theses details.
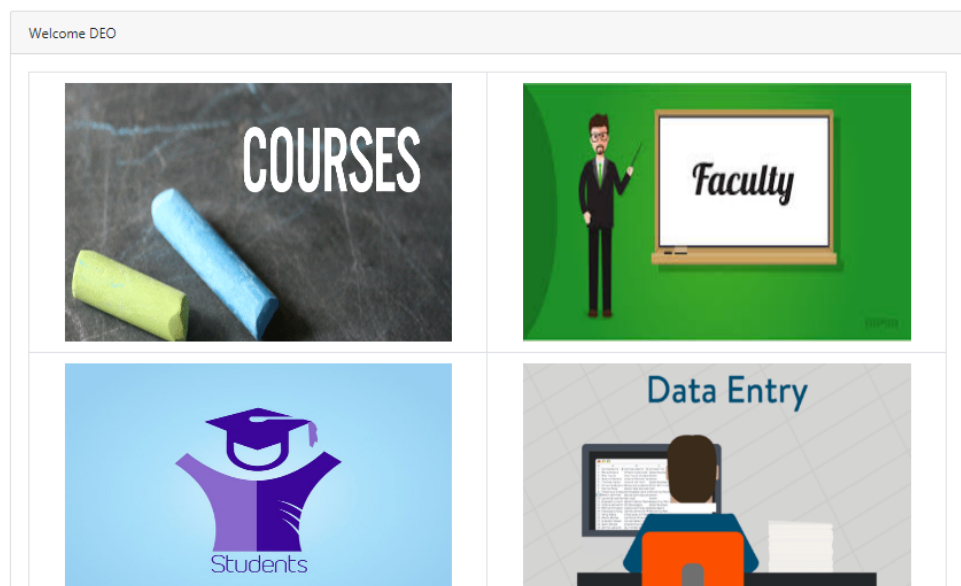
Faculty Login

Faculty has to enter username and password to login into the system. The system will check for empty or incorrect username or password.

Faculty Home Page

This page shows the overall attendance status of the students to whom that faculty teaches. It contains student's name, roll number, course, and attendance percentage. Also there is an option

for pdf report. PDF report asks for a from date and to date and creates a report for a particular student and attendance details of that student.

Faculty Profile Page

Attendance Tracking Management System

Home   Profile   Attendance   Defaulter   Logout

Profile	
Name *	Rahul Kumar
Address *	Rohini, Delhi
Email Address *	rahul.kumar@gmail.com
Qualification *	M.Sc
Course *	SE 401
Date of Joining *	2020-09-01

This page shows the profile of the faculty. The details of the faculty are displayed in read only form is - Name, Address, Email-id, qualification, course code, and date of joining.

Attendance List

Attendance Tracking Management System

Home   Profile   Attendance   Defaulter   Logout

Attendance List   Report   Add

Show 10 ▲▼ entries   Search: [   ]

Roll Number ↑↓	Student Name ↑↓	Course ↑↓	Attendance Status ↑↓	Attendance Date ↑↓	Edit ↑↓	Delete ↑↓
1	Ankit Yadav	Course 4	Present	2020-09-01	Edit	Delete
2	Sagar Garg	Course 4	Present	2020-09-01	Edit	Delete
3	Ojasvi Tayagi	Course 4	Present	2020-09-01	Edit	Delete
4	Ritwik Garg	Course 4	Absent	2020-09-01	Edit	Delete
5	Abhishek Sethi	Course 4	Absent	2020-09-01	Edit	Delete
1	Ankit Yadav	Course 4	Present	2020-09-02	Edit	Delete
2	Sagar Garg	Course 4	Present	2020-09-02	Edit	Delete

This page shows the list of the attendance of the students to whom the faculty teaches. It displays student roll number, name, course, attendance status, and date. Option to add, edit or delete is also present.

Defaulter's List

Attendance Tracking Management System

Home Profile Attendance Defaulter Logout

Defaulter's List

Show 10 ⬍ entries   Search: [_____]

Student Name	Roll Number	Course Code	Attendance Percentage
Ritwik Garg	4	Course 4	60%
Abhishek Sethi	5	Course 4	70%

Showing 1 to 10 of 21 entries (filtered from 5 total entries)   Previous 1 2 3 Next

This page displays the list of the defaulter's – students whose attendance is less than 75%.

Student Login Page

Attendance Tracking Management System

Student Login

Enter Username

[_____]

Enter Password

[_____]

Login

Student has to enter username and password to login into the system. The system will check for empty or incorrect username or password.

Student Home Page

Attendance Tracking Management System

Home   Profile   Attendance   Logout

Attendance Chart

Student Name	Rohit Kumar
Course	Course 1
Faculty	Bhupender Rana
Time Period	2020-01-01 to 2020-12-31

Overall Attendance Analytics

● Present
● Absent

20%
80%

Overall Attendance Analytics

● Present
● Absent

20%
80%
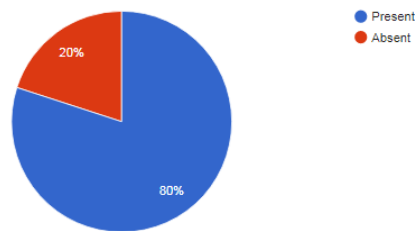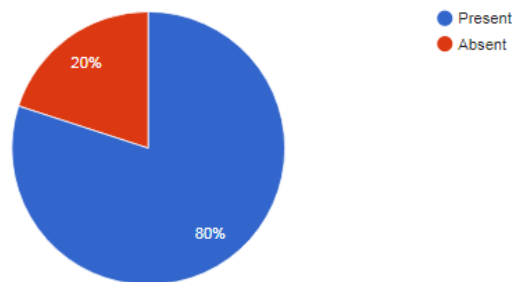
Date	Attendance Status
2020-09-01	Present
2020-09-02	Absent
2020-09-03	Present
2020-09-04	Present
2020-09-06	Present
2020-09-07	Present
2020-09-08	Absent

This page displays the overall attendance of the student in the form of table and pie chart. The list for details of attendance is also displayed.

Student Profile Page

Attendance Tracking Management System

Home   Profile   Attendance   Logout

Profile	
Student Name *	Rohit Kumar
Email Address *	rohit@gmail.com
Course *	SE 101
Date of Birth *	2003-04-19

This page displays all the information of student in read only form. The information contains name, email address, course, and date of birth.

Attendance Page

Attendance Tracking Management System

Home   Profile   Attendance   Logout

Attendance List

Show 10 ⬍ entries   Search:

Faculty Name ↑↓	Course Code ↑↓	Attendance Status ↑↓	Attendance Date ↑↓
Bhupender Rana	SE 101	Present	2020-09-01
Bhupender Rana	SE 101	Absent	2020-09-02
Bhupender Rana	SE 101	Present	2020-09-03
Bhupender Rana	SE 101	Present	2020-09-04
Bhupender Rana	SE 101	Present	2020-09-06
Bhupender Rana	SE 101	Present	2020-09-07
Bhupender Rana	SE 101	Present	2020-09-08

This page shows the list of the attendance details date wise. It shows faculty name, course, attendance status, and attendance date.

Make Report

This displays the form to make report. User has to enter the from date and to date and the report will be generated in the pdf format as shown below.

PDF Report of Student

Report of a particular student is formed in the pdf format. It displays student name, roll number, course, attendance date, and attendance status.

PDF report of a Course

<div align="center">

Attendance Report

</div>

Date - 2020-09-01

Student Name	Roll Number	Course	Faculty	Attendance Status
Aman Sharma	1	Course 1	Bhupender Rana	Present
Rohit Kumar	2	Course 1	Bhupender Rana	Present
Devesh Singh	3	Course 1	Bhupender Rana	Present
Aakash Garg	4	Course 1	Bhupender Rana	Present
Rohan Negi	5	Course 1	Bhupender Rana	Present

Date - 2020-09-02

Student Name	Roll Number	Course	Faculty	Attendance Status
Aman Sharma	1	Course 1	Bhupender Rana	Present
Rohit Kumar	2	Course 1	Bhupender Rana	Absent
Devesh Singh	3	Course 1	Bhupender Rana	Present
Aakash Garg	4	Course 1	Bhupender Rana	Absent
Rohan Negi	5	Course 1	Bhupender Rana	Present

Report of a particular course is formed in the pdf format. It displays student name, roll number, course, faculty name, attendance date, and attendance status.

Chart Report of a Course

Attendance Tracking Management System

Home   Course   Faculty   Student   Admin   DEO   Attendance   Defaulter   Logout

Attendance Chart

Course	Course 2
Date	2020-09-04

Overall Attendance Analytics

- Present
- Absent

40%

60%
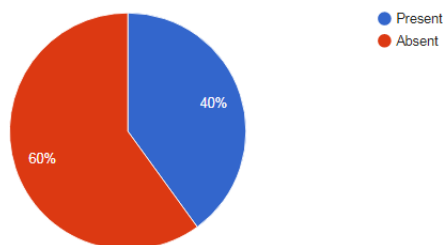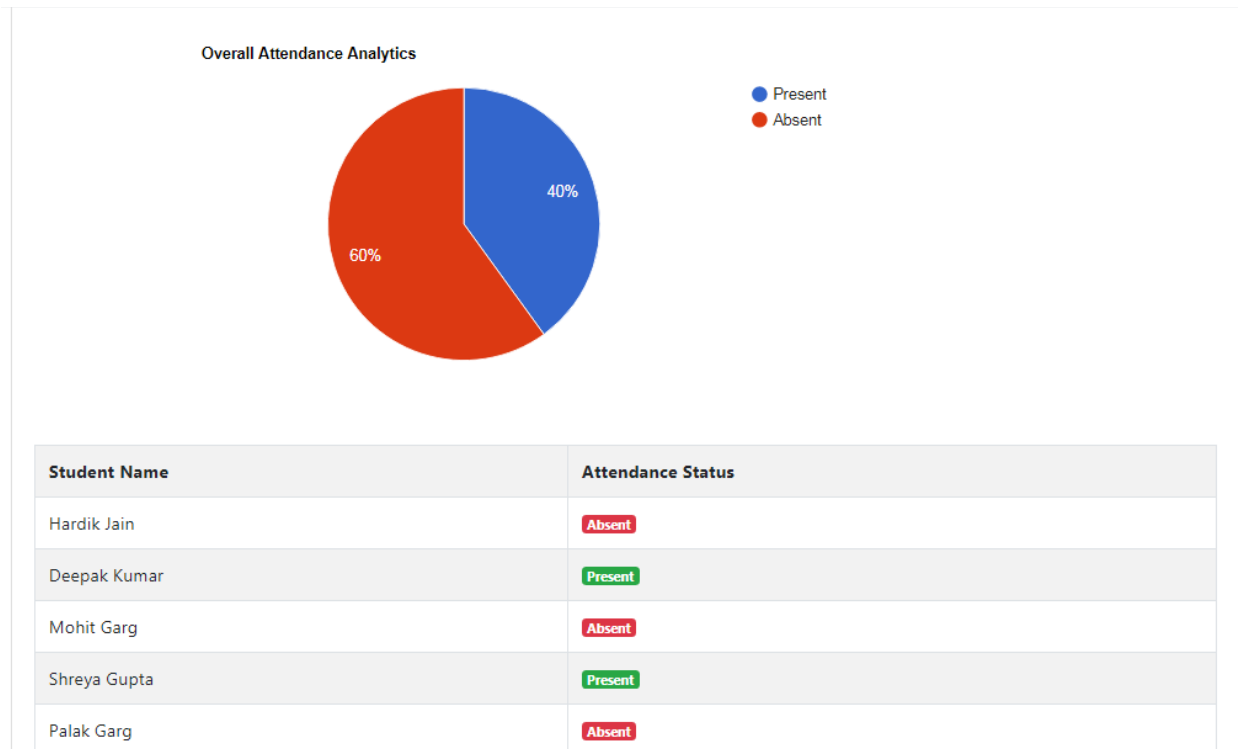
Pie chart of a particular course is displayed. It displays student name, course, attendance date, and attendance status.

Add Attendance for Faculty

This is the form to add attendance for the faculty. It displays course, date and list of students with their roll number, name and attendance status.

TEST CASES

LOGIN

Test Case ID	Scenario name and Description	Username	Password	Expected Output	Remarks (if any)
TC 1	Scenario 1 – Login Basic Flow	Abcuser	abc1122	Login successful	-
TC 2	Scenario 2 – Login Alternate Flow: Invalid Username	abc!	abc	Invalid Username	Username is invalid
TC 3		abc	Pass@	Invalid Password	Password is invalid
TC 4		Abcuser5	abc123	Username Invalid	Username does not exists in database
TC 5		Abcuser	password56*	Password Invalid	Password does not exists in database

MAINTAIN LOGIN

Test Case ID	Scenario name and Description	Username	Password	Update confirmed	Delete confirmed	Expected Output	Remarks (if any)
TC 1	Scenario 1 - Add User Basic Flow	abcuser	abc1122	N/A	N/A	User Added Successfully	-
TC 2	Scenario 2 - Add User Alternate Flow - Invalid Entry	us@	Abc	N/A	N/A	Username invalid	Username is in invalid format
TC 3		abcuser	Abc@!	N/A	N/A	Password invalid	Password is in invalid format
TC 4	Scenario 3 - Add User Alternate Flow - Username already exists	abcuser	abc1122	N/A	N/A	Username already exists	The user with the same username already exists

TC	Scenario						
TC 5	Scenario 4 - Add User Alternate Flow - User Exits	abc	Abc	N/A	N/A	User is allowed to exit any time	-
TC 6	Scenario 5 - Update User	abcuser	abc2223	Yes	N/A	User Updated successfully	-
TC 7	Scenario 6 - Update User Alternate Flow - Invalid Entry	a@	abc	N/A	N/A	Username invalid	Username is in invalid format
TC 8		Abcuser	@#	N/A	N/A	Password invalid	Password is in invalid format
TC 9	Scenario 7 - Update User Alternate Flow - Username already exists	Abcuser	abc2223	Yes	N/A	Username already exists	The user with the same username already exists
TC 10	Scenario 8 - Update User Alternate Flow - Username not found	Abcuser	abc	N/A	N/A	User not found	User with this username does not exists in database
TC 11	Scenario 9 - Update User Alternate Flow - Update cancelled	abcuser	abc2223	No	N/A	Update is cancelled	
TC 12	Scenario 10 - Update User Alternate Flow - User exits	Abc	abc	N/A	N/A	User is allowed to exit any time	-
TC 13	Scenario 11 - Delete User	abcuser	N/A	N/A	Yes	User Deleted	-
TC 14	Scenario 12 - Delete User Alternate Flow -	abcuser	N/A	N/A	N/A	User not found	User with this username does not exists in database

Test Case ID	Scenario name and Description	Username				Expected Output	Remarks
	Username not found						
TC 15	Scenario 13 - Delete User alternate Flow - Delete Cancelled	abcuser	N/A	N/A	No	Delete Cancelled	-
TC 16	Scenario 14 - Delete User Alternate Flow - User exits	Abc	N/A	N/A	N/A	User is allowed to exit any time	-
TC 17	Scenario 15 - View User	abcuser	N/A	N/A	N/A	User is displayed successfully	-
TC 18	Scenario 16 - View User Alternate Flow - Username not found	abcuser	N/A	N/A	N/A	User not found	User with this username does not exists in database
TC 19	Scenario 17 - View User Alternate Flow - User Exits	abc	N/A	N/A	N/A	User is allowed to exit any time	-

MAINTAIN STUDENT

Test Case ID	Scenario name and Description	Roll Number	Name	Email ID	Date of Birth	Course Code	Update confirmed	Delete confirmed	Expected Output	Remarks (if any)
TC 1	Scenario 1 - Add Student Basic Flow	001	abc	abc@gmail.com	2000-02-13	SE201	N/A	N/A	Student Added Successfully	-

TC	Scenario									
TC 2	Scenario 2 - Add Student Alternate Flow - Invalid Entry	a1	abc	abc	abc	abc	N/A	N/A	Roll Number invalid	Roll Number is in invalid format
TC 3		001	In+	abc	Abc	abc	N/A	N/A	Name invalid	Name is in invalid format
TC 4		001	Mohit	abc	abc	Abc	N/A	N/A	Email ID invalid	Email ID is in invalid format
TC 5		001	Mohit	mohi@gmail.com	abc	Abc	N/A	N/A	Date of Birth invalid	Date of Birth is in invalid format
TC 6		001	Mohit	mohit@gmail.com	2000-02-13	SE@	N/A	N/A	Course Code invalid	Course Code is in invalid format
TC 7	Scenario 3 - Add Student Alternate Flow - Course not found	001	Mohit	mohit@gmail.com	2000-02-13	SE701	N/A	N/A	Course not found	Course does not exists in database
TC 8	Scenario 4 - Add Student Alternate Flow - Student already exists	001	Mohit	mohit@gmail.com	2000-02-13	SE201	N/A	N/A	Student already exists	The student with the same roll number and course code alr

TC	Scenario								Expected Result	Comments
										eady exists
TC 9	Scenario 5 - Add Student Alternate Flow - User Exits	abc	abc	abc	abc	abc	N/A	N/A	User is allowed to exit any time	-
TC 10	Scenario 5 - Update Student	001	+	+	2000-02-13	SE201	Yes	N/A	User Updated successfully	-
TC 11	Scenario 6 - Update Student Alternate Flow - Invalid Entry	a1	*	*	*	*	N/A	N/A	Roll number invalid	Roll number is in invalid format
TC 12		001	In+	*	*	*	N/A	N/A	Name invalid	Name is in invalid format
TC 13		001	+	In+	*	*	N/A	N/A	Email ID invalid	Email ID is in invalid format
TC 14		001	+	+	13-02-2000	*	N/A	N/A	Date of Birth invalid	Date of Birth is in invalid format

TC 15		001	In+	+	2000-02-13	SE	N/A	N/A	Course Code invalid	Course Code is in invalid format
TC 16	Scenario 8 - Update Student Alternate Flow - Student already exists	001	+	+	2000-02-13	SE201	Yes	N/A	Student already exists	The Student with the same roll number and course code already exists
TC 17	Scenario 9 - Update Student Alternate Flow - Student not found	001	*	*	*	*	N/A	N/A	Student not found	Student with this roll number and course code does not exists in database
TC 18	Scenario 10 - Update Student Alternate Flow - Course not found	001	*	*	*	SE201	N/A	N/A	Course not found	Course does not exists in database
TC 19	Scenario 11 - Update Student Alternate Flow -	001	+	+	2000-02-13	SE201	No	N/A	Update cancelled	

	Update cancelled									
TC 20	Scenario 12 - Update Student Alternate Flow - User exits	*	*	*	*	*	N/A	N/A	User is allowed to exit any time	-
TC 21	Scenario 13 - Delete Student	001	N/A	N/A	N/A	N/A	N/A	Yes	Student Deleted	-
TC 22	Scenario 14 - Delete Student Alternate Flow - Student not found	001	*	*	*	*	N/A	N/A	User not found	The Student with the roll number and course code does not exists in the database
TC 23	Scenario 15 - Delete Student alternate Flow - Delete Cancelled	001	N/A	N/A	N/A	N/A	N/A	No	Delete Cancelled	-

TC 24	Scenario 16 - Delete Student Alternate Flow - User exits	*	N/A	N/A	N/A	N/A	N/A	N/A	User is allowed to exit any time	-
TC 25	Scenario 17 - View Student	001	N/A	N/A	N/A	N/A	N/A	N/A	Student is display ed success fully	-
TC 26	Scenario 18 - View Student Alternate Flow - Student not found	001	*	*	*	*	N/A	N/A	Student not found	\