ATTENDANCE MANAGEMENT SYSTEM

Aarthi.V.S (185001003)
Adithya Vikram N (185001009)
Anusha Chandrasekaran (185001020)

Problem Statement

In our current educational scenario, maintaining a good attendance record is a necessity for every student. Not only does it ensure that they can follow the curriculum but it is also often an eligibility criteria to attempt examinations. Thus, it is important for students to always be on top of the status of their attendance. But in the current system, it is difficult for the students to access their attendance records since it is maintained in paper. The faculty also have a tough time in sharing the information in real-time. Furthermore, it is time consuming for the faculty to organize and maintain attendance records for multiple classes and courses. Also, since everything is done manually, there are bound to be a few human errors in the records. Since the attendance records are only shared periodically to the students, these errors are usually spotted very late, at which point it is very difficult or even impossible to rectify them. On top of these issues, keeping track of OD's is a nightmare for both the students and the faculty.
These problems can be prevented if there exists a real-time, transparent software system which is accessible to both the students and the faculty.

Proposed Solution:

Our goal is to build a software system that eliminates the problems in our current attendance management system. The main idea is to introduce a common interface for both the faculty and the students. In order to meet our goal, our system will comprise of the following:

A student platform which allows the students to freely access their attendance records. Through this, the students can keep track of the status of their attendance and quickly report any discrepancies. This also helps them make decisions regarding their attendance percentage.
A faculty platform which lets the faculty take and store attendance with ease. It consolidates and summarizes all the records for efficient tracking and management. It also minimizes the errors and saves a lot of time for the faculty.
Regular statistics reports for each student and overall reports for the faculty. It also includes other practical statistical functionalities.
A Notification system that alerts the students when their attendance goes below the cutoff percentage.
A Feature to manage the requests and approvals of ODs.

Conclusion:

Our system acts as a bridge between the students and the faculty to tackle the existing lack of a 2 way platform and the transparency that comes with it, with the added benefits of reduced errors, and increased efficiency. Overall, it is a hassle free way to manage attendance.

Attendance Management System
Software Requirements Specification

Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

1.Introduction

Currently, the attendance process is done manually wherein the faculty take and record the attendance of a class on paper and the students have no easy way of viewing their attendance unless they approach the faculty in charge even after which there might be some delay because of consolidation of records kept on paper. This is extremely time consuming for the faculty and prone to some human errors. A lot of times, identifying and notifying the faculty of any discrepancies in the attendance records can be done only after the attendance has been finalized. The process of requesting and approving ODs is also prone to all the problems mentioned above.

We have decided to investigate the use of an AMS. This system would be used by students to track their attendance and by faculty to manage attendance of their courses. The purpose of this document is to analyze and elaborate on the high-level needs and features of the AMS. It focuses on the capabilities and facilities provided by a Library.

1.1 Purpose

The purpose of Software Requirements Specification (SRS) document is to describe the external behavior of the Online Library System. Requirements Specification defines and describes the operations, interfaces, performance, and quality assurance requirements of the Online Library System. The document also describes the nonfunctional requirements such as the user interfaces. It also describes the design constraints that are to be considered when the system is to be designed, and other factors necessary to provide a complete and comprehensive description of the requirements for the software. The Software Requirements Specification (SRS) captures the complete software requirements for the system, or a portion of the system. Requirements described in this document are derived from the Vision Document prepared for the Online Library System.

1.2 Scope

The Software Requirements Specification captures all the requirements in a single document. The AMS that is to be developed provides the faculty and students with many attendance management facilities. The AMS is supposed to have the following features.

The system allows the faculty/Administrator to create a class and add students to the same.
The system provides a login facility to the users.
The system provides the users with the option to check their account details and/or change their options like password of the account whenever needed all through the day.
The system provides the faculty with capabilities to record and store attendance of their courses and the AMS is up and running all day.
The system lets the faculty check the attendance of all the students in their class.
The system allows the students to access their attendance details 24 hours a day and all through the semester.

1.3 Definitions, Acronyms and Abbreviations

AMS – Attendance Management System

1.4 Overview

The SRS will provide a detailed description of the Online Library System. This document will provide the outline of the requirements, overview of the characteristics and constraints of the system.

1.4.1 Section 2:

This section of the SRS will provide the general factors that affect the product and its requirements. It provides the background for those requirements. The items such as product perspective, product function, user characteristics, constraints, assumptions and dependencies and requirements subsets are described in this section.

1.4.2 Section 3:

This section of SRS contains all the software requirements mentioned in section 2 in detail sufficient enough to enable designers to design the system to satisfy the requirements and testers to test if the system satisfies those requirements.

2. Overall Description

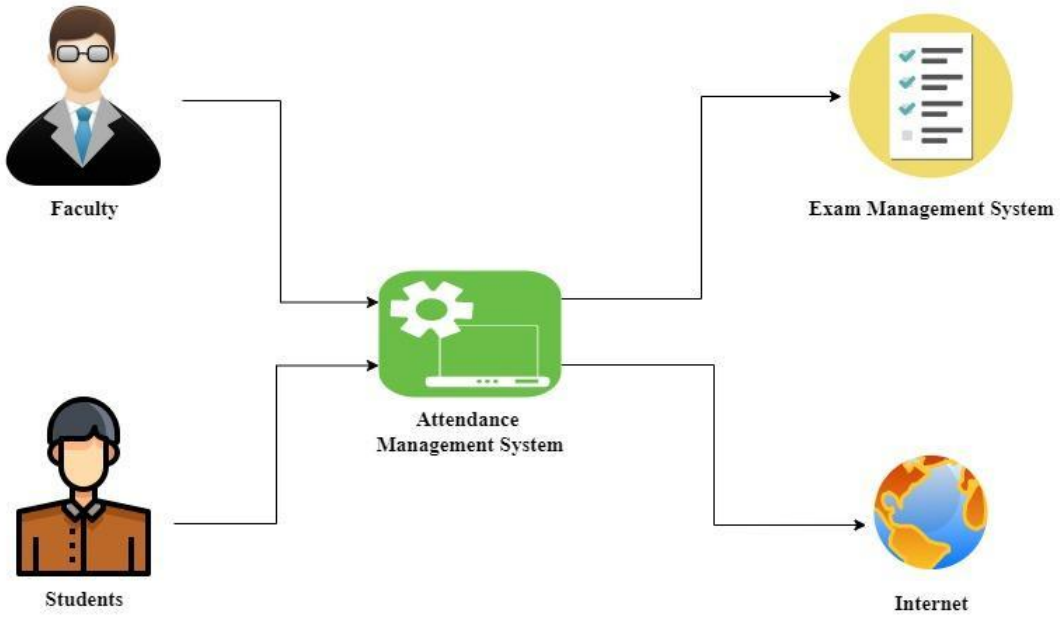
2.1 Product Perspective

The Attendance Management System (AMS) is a software that can be used by educational institutions to improve the efficiency of the attendance management process. Both the students and faculty will be able to save a lot of time and effort using the AMS. Our system helps the faculty record,store and manage the attendance of their students and also provides insightful

statistics regarding the overall attendance of the class. The students can track their attendance in real-time while also being able to report any discrepancies.

The complete overview of the system is as shown in the overview diagram below:
The product to be developed has interactions with the users: Faculty & Students.

The product has to interact with other systems like: Internet and the Institution's Exam management System.



2.2 Product Functions

The AMS provides online real time information about the attendance to all users. The Product functions are more or less the same as described in the product perspective. The system provides different types of services based on the type of users [Faculty/Student].

The students should be provided real time information about their attendance.
The students are given a provision to request ODs and attach relevant information for the same.
The students are provided with a feature to report any discrepancies in their attendance.
The faculty are provided with an interface to take and record attendance for each of the courses under them.
The faculty can get the information about their students' attendance.
The faculty are given a provision to approve ODs and resolve the reported discrepancies.
A notification functionality should be provided to alert students when their attendance reduces below the cut-off percentage.

2.3 User characteristics

The users of the system are students and the faculty and the administrators who maintain the system. The members and the librarian are assumed to have basic knowledge of the computers and Internet browsing. The administrators of the system must have more knowledge of the internals of the system and must be able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, users manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

2.4 Constraints

The information of all the users must be stored in a database that is accessible by the AMS.
The AMS is running all 24 hours a day.
The users access the AMS from any computer that has Internet browsing capabilities and an Internet connection.
The users must have their correct usernames and passwords to enter into the AMS.

2.5 Assumptions and dependencies

The users have sufficient knowledge of computers.
The classrooms should have Internet connection so that faculty can record attendance in the AMS.
The users know the English language, as the user interface will be provided in English.
The product can access the university student database.

3. Specific Requirements

This section describes in detail all the functional requirements.

3.1 Functionality

3.1.1 Login Capabilities

The system shall provide the users with login capabilities.

3.1.3 Alerts

The system can alert the administrator in case of any problems. It can also send alerts to students in case of attendance shortage.

3.2 Usability

The system shall allow the users to access the system from the Internet using HTML or it's derivative technologies. The system uses a web browser as an interface.
Since all users are familiar with the general usage of browsers, no specific training is required.
The system is user friendly and self-explanatory.

3.3 Reliability

The system has to be very reliable due to the importance of data and the damages incorrect or incomplete data can do.

3.3.1 Availability

The system is available 100% for the user and is used 24 hrs a day and 365 days a year.
The system shall be operational 24 hours a day and 7 days a week.

3.3.2 Mean Time Between Failures (MTBF)

The system will be developed in such a way that it may fail once in a year.

3.3.3 Mean Time to Repair (MTTR)

Even if the system fails, the system will be recovered back up within an hour or less.

3.3.4 Accuracy

The accuracy of the system is limited by the accuracy of the speed at which the employees of the library and users of the library use the system.

3.3.5 Maximum Bugs or Defect Rate

Not specified.

3.3.6 Access Reliability

The system shall provide 100% access reliability.

3.4 Performance

3.4.1 Response Time

The Login page should be able to be loaded within 3 seconds. The information is refreshed every 30 seconds. The system shall respond to the member in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when doing large processing jobs.

3.4.2 Administrator Response

The system shall take as less time as possible to provide service to the administrator.

3.4.3 Throughput

The number of transactions is directly dependent on the number of users, the users may be the faculty or students who use the system for entering attendance and checking their attendance.

3.4.4 Capacity

The system is capable of handling 250 users at a time.

3.4.5 Resource Utilization

The resources are modified according to the user requirements.

3.5 Supportability

The system designers shall take into consideration the following supportability and technical limitations.

3.5.1 Internet Protocols

The system shall comply with the TCP/IP protocol standards and shall be designed accordingly.

3.5.2 Maintenance

The maintenance of the system shall be done as per the maintenance contract.

3.5.3 Standards

The coding standards and naming conventions will be as per the ISO standards.

3.6 Design Constraints

3.6.1 Software Language Used

The languages that shall be used for coding the AMS are Active Server Pages (ASP), Java Servlets, Java Server Pages (JSP), HTML, JavaScript, and VBScript. For working on the coding phase of the AMS, the Internet Information Services (IIS) Server needs to be installed.

3.6.2 Development Tools

Will make use of the available Java Development Tool kits for working with JavaBeans and Java Server Pages. Also will make use of the online references available for developing programs in ASP, HTML and the two scripting languages, JavaScript and VBScript.

3.6.3 Class Libraries

Will make use of the existing Java libraries available for JSP and Servlets.

3.7 On-line User Documentation and Help System Requirements

Online help is provided for each of the features available with the AMS. All the applications provide an on-line help system to assist the user. The nature of these systems is unique to application development as they combine aspects of programming (hyperlinks, etc) with aspects of technical writing (organization, presentation).

Also, a Read Me file is typically included as a standard component. The Read Me includes a "What's New With This Release" section, and a discussion of compatibility issues with earlier releases. Most users also appreciate documentation defining any known bugs and workarounds in the Read Me file.

3.8 Purchased Components

The System Administrator will need to purchase the license for IIS Server. Mostly it is available with Windows Environment. So the system doesn't need to purchase any licensing products.

3.9 Interfaces

3.9.1 User Interfaces

The AMS will make use of the existing Web Browsers. The user interface of the system shall be designed with the following pages included.

Login Page
Dashboard for Students
Dashboard for Faculty
Attendance Entry page for Faculty
Detailed Course Attendance Information for Faculty
Detailed Course Attendance Information for Students

3.9.2 Hardware Interfaces

The existing Local Area Network (LAN) will be used for collecting data from the users.

3.9.3 Software Interfaces

A firewall will be used with the server to prevent unauthorized access to the system.

3.9.4 Communications Interfaces

The AMS will be connected to the World Wide Web.

3.10 Licensing Requirements

The usage is restricted to only SSN Institutions who are purchasing the AMS and signs the maintenance contract.

3.11 Legal, Copyright, and Other Notices

AMS is a trademark of SSN Institutions and cannot be used without its consent.

3.12 Applicable Standards

The ISO/IEC 27001:2019 guidelines for the documentation of computer based application systems will be followed.

4. Supporting Information

The use-case storyboards or the user-interface prototypes are not available. The appendices are not to be considered as part of the requirements.

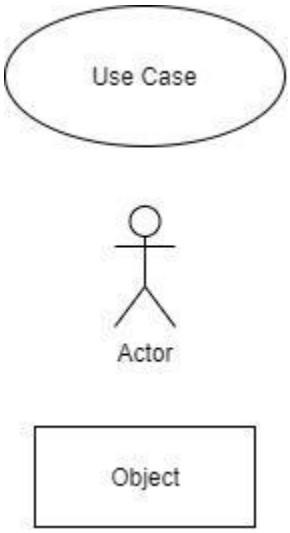
Attendance Management System
Use Case Model

Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

Aim:

To create a UML use-case diagram for the Attendance Management System.

Notations:



Identification of Actors:

Student
Faculty
Administrator
Attendance Management System
Credentials Database
User (Student/Faculty)

Identification of Scenarios:

The user (student/faculty) logs in to the system.
The faculty record attendance for a session.
The faculty modify attendance in case of discrepancies or OD approval.
The faculty view the overall course attendance report.
The students view their attendance history.
The students get their attendance percentage.
The students are notified about their absence in a class.

The students report a discrepancy in case of an error.
The students apply for OD.
The administrator adds and manages students in the system.
The administrator adds and manages faculty in the system.
The administrator deletes students from the system.
The administrator deletes faculty from the system.
The administrator adds and manages a course in the system by registering and removing students and assigning and removing faculty.
The administrator exports the attendance data of a course.
The administrator finishes a course which deletes the course attendance details from the system.

Relating Use Cases:

Generalization:

In the generalization process, the common characteristics of classes are combined to form a class in a higher level of hierarchy, i.e., subclasses are combined to form a generalized super-class. It represents an “is – a – kind – of” relationship.

Association:

Includes:

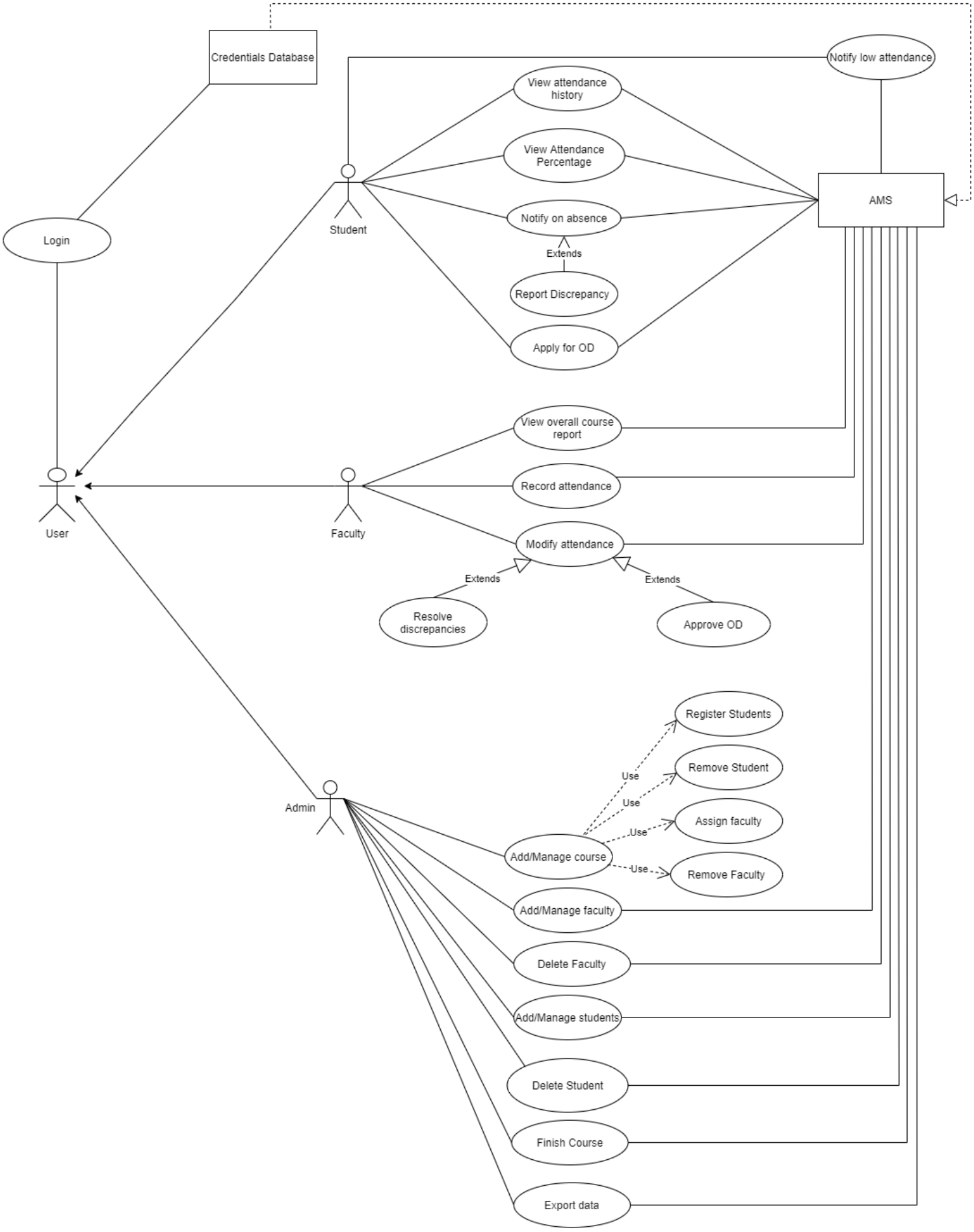
An include relationship between use cases means that the base use case explicitly incorporates the behavior of another use case at a location specified in the base.

Extends:

An extends relationship between use cases means that the base use case implicitly incorporates the behavior of another use case at a location specified indirectly by the extending use case.

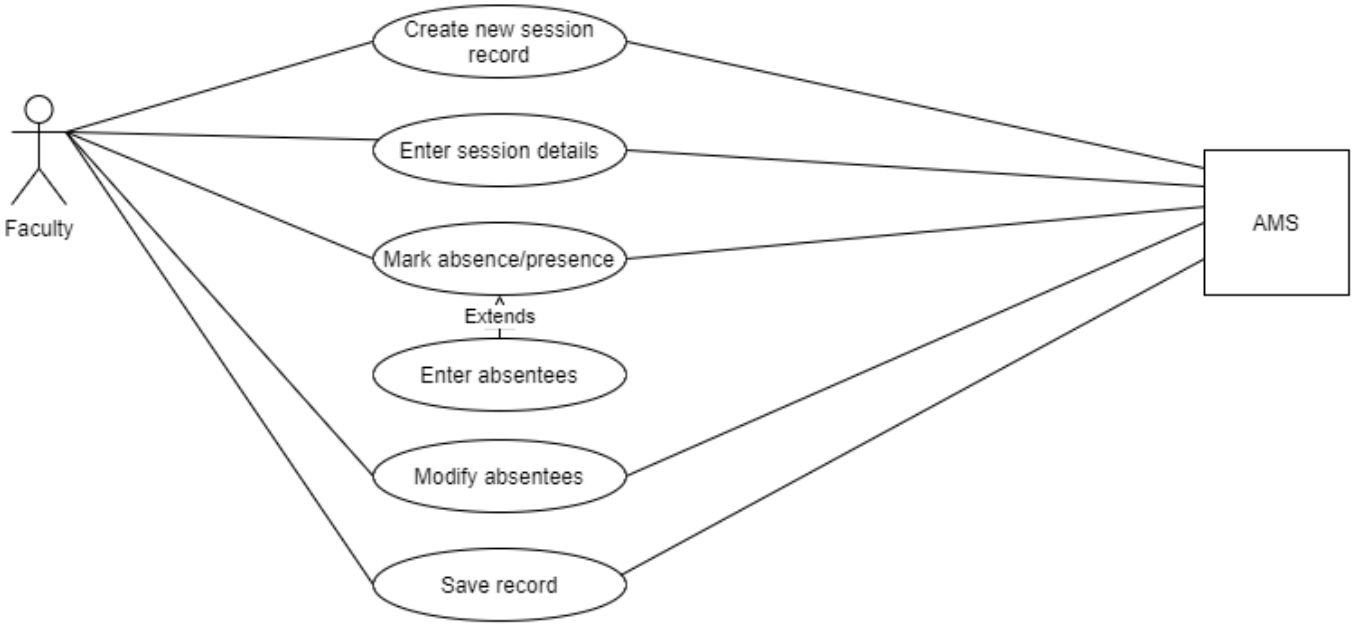
Use Cases:

Main Success Scenario

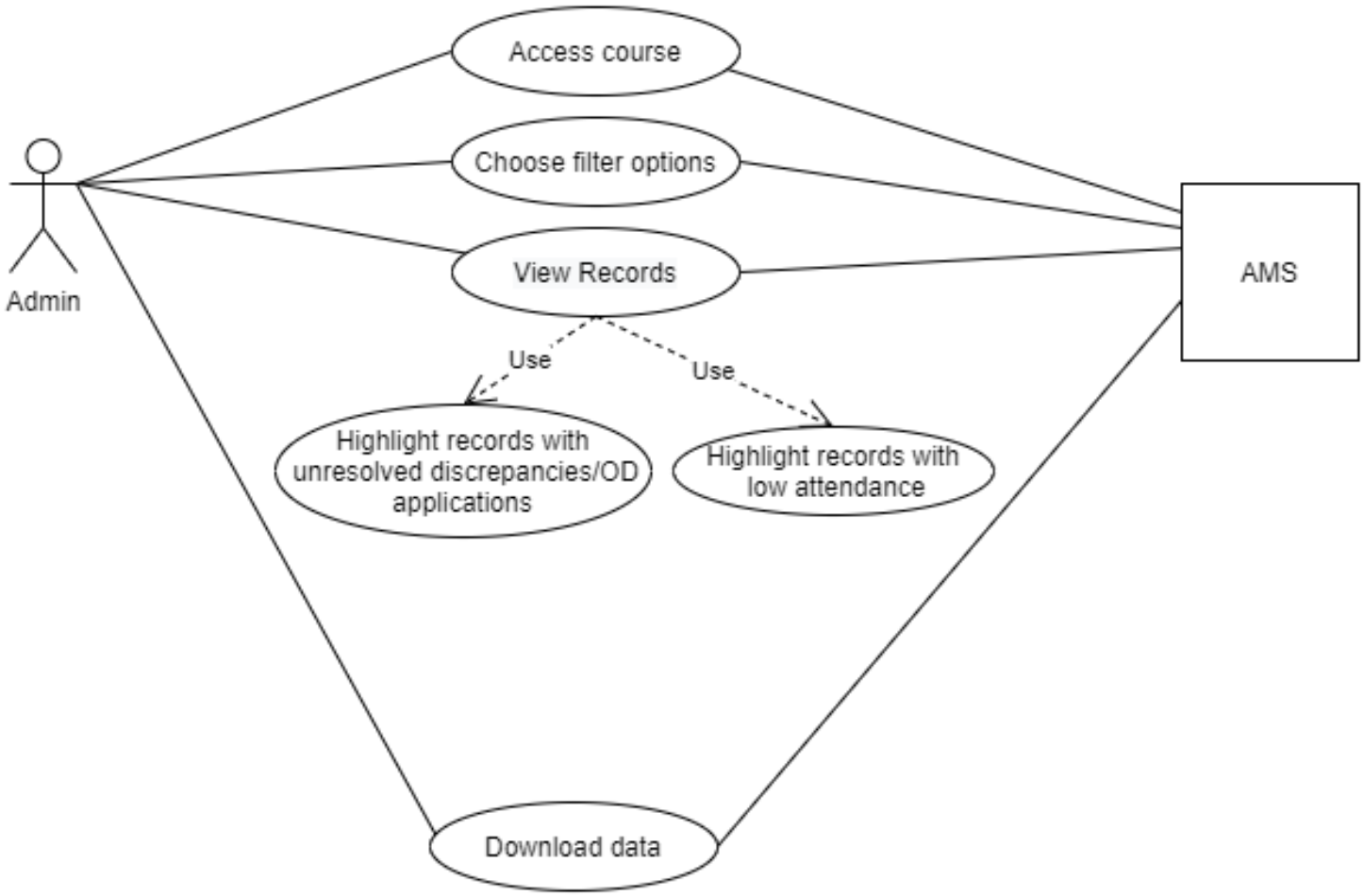

Credentials Database
Notify low attendance
View attendance history
View Attendance Percentage
Notify on absence
Extends
Report Discrepancy
Apply for OD
Student
AMS
Login
View overall course report
Record attendance
Modify attendance
Extends
Extends
Resolve discrepancies
Approve OD
User
Faculty
Admin
Register Students
Remove Student
Assign faculty
Remove Faculty
Use
Use
Use
Use
Add/Manage course
Add/Manage faculty
Delete Faculty
Add/Manage students
Delete Student
Finish Course
Export data

Sub-functions

Record Attendance

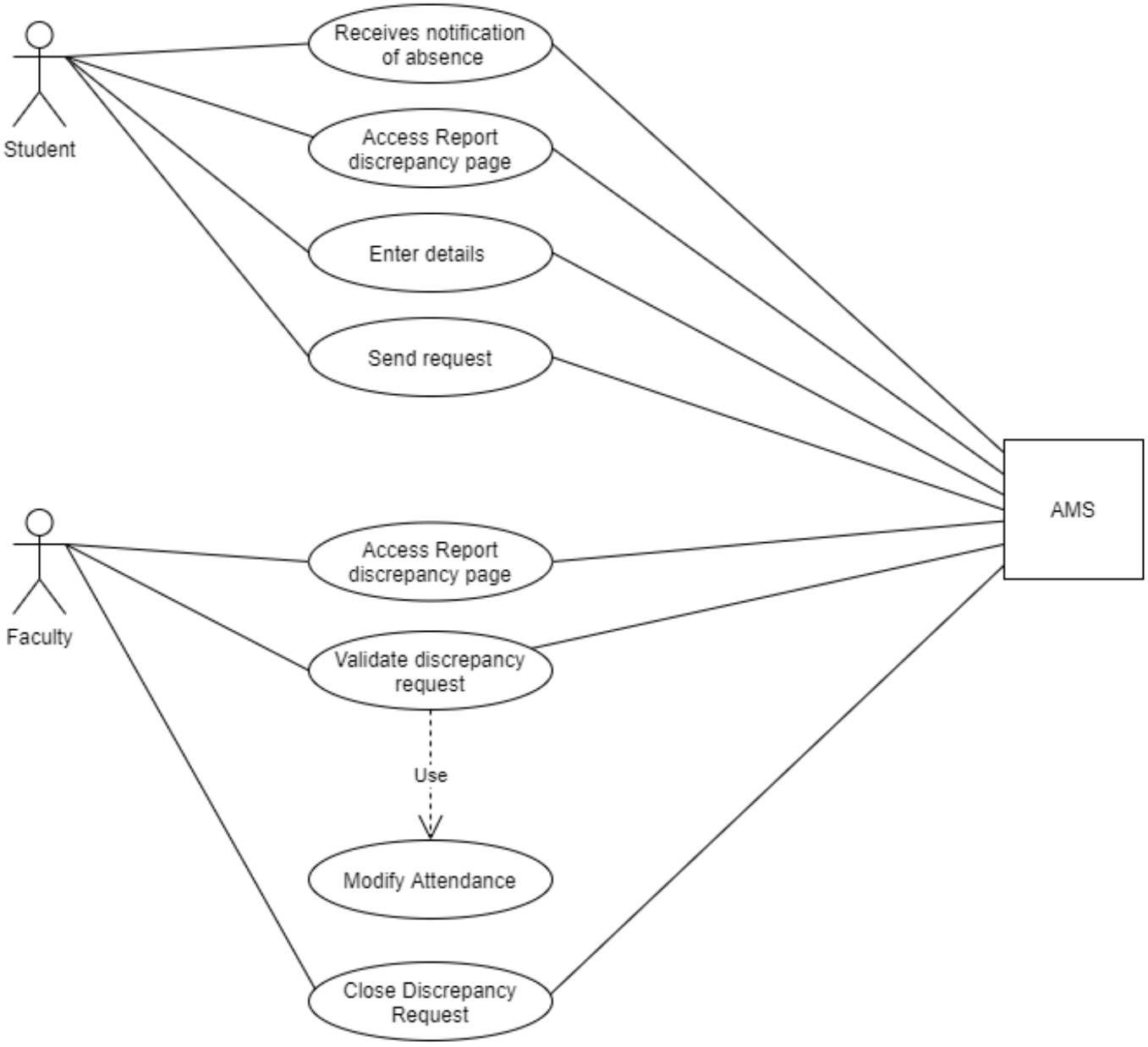


Export data

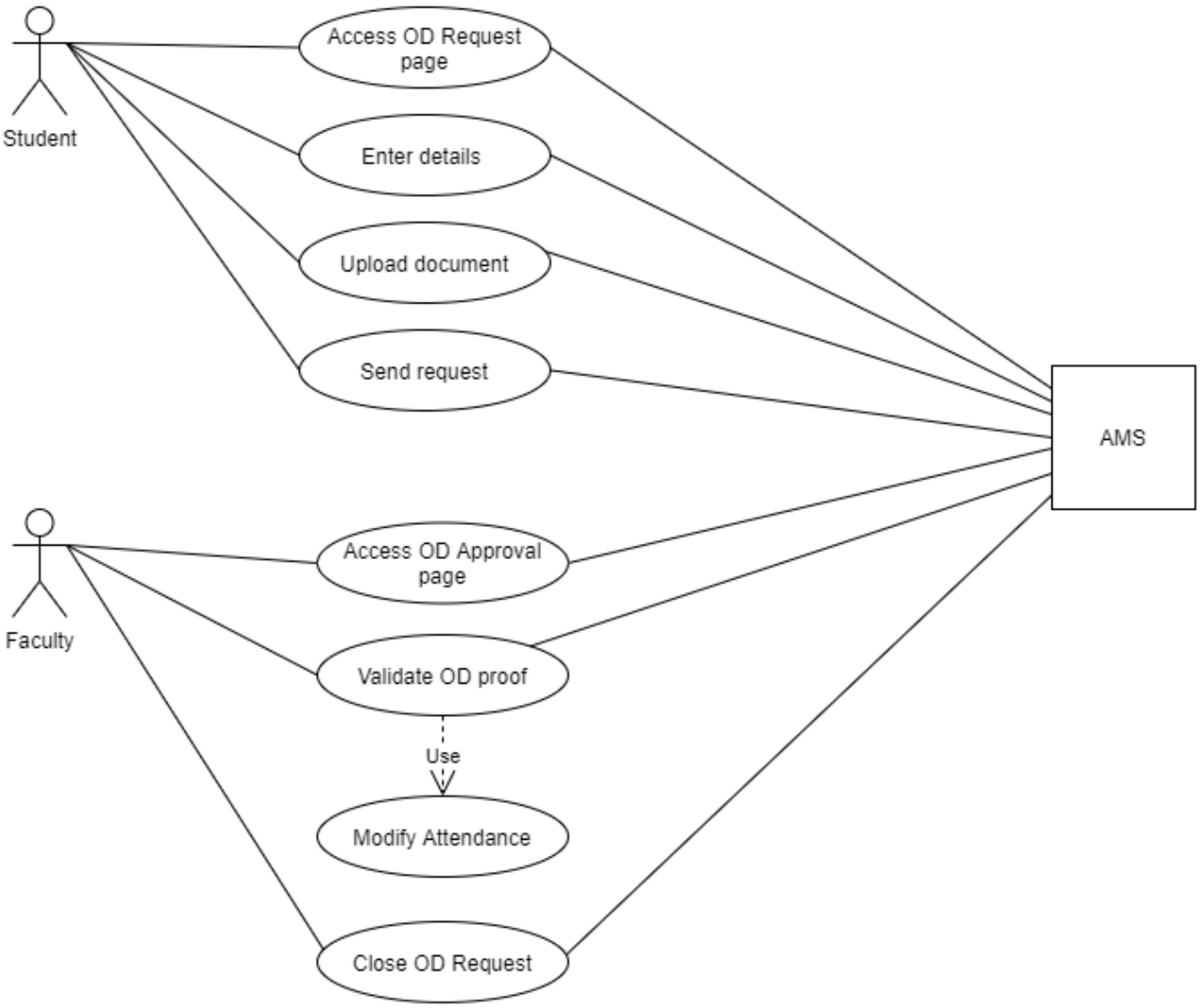

Access course
Choose filter options
View Records
Admin
AMS
Use
Use
Highlight records with unresolved discrepancies/OD applications
Highlight records with low attendance
Download data

Alternate Scenarios:

Discrepancy- Failure scenario



Manage OD


Student
Access OD Request page
Enter details
Upload document
Send request
AMS
Faculty
Access OD Approval page
Validate OD proof
Use
Modify Attendance
Close OD Request

Fully Dressed Use Cases :

1) *Use Case Name* : Manage Attendance

Scope : Attendance Management System

Level : User Goal

Primary Actor : Faculty

Stakeholders and Interests :

Faculty: Wants an user-friendly digital interface to record, save and manage attendance with no errors that will reduce their administrative workload.

Student : Wants an interface that displays their attendance in real-time and provides statistics for the same along with provisions for applying for OD.

Administrator : Wants a system to easily and reliably manage student, faculty and course information while also being able to export relevant attendance data when required.

Institution : Wants a fault tolerant attendance system with punctual,automatic reporting which will reduce the number of conflicts.

Preconditions **:**

Faculty and students are logged into the system using their usernames and passwords.

*Success Guarantee***:**

Attendance is recorded. Attendance Percentage is calculated. Students are notified of their absence.

Main Success Scenario:

Faculty create a new session record.
Faculty enter session details.
Faculty take attendance.
Faculty verify the absentees list and save the record.
Students are notified of their absence in a session.
Students view their attendance record and attendance percentage.
Faculty gets the overall course attendance report including statistics.
Students apply for OD by providing all relevant details and can view the status of their application.

Administrator exports the required attendance data of a course to the institution.

Extensions:

Session record not saved

By human error

Faculty can create a new session record and retake the attendance.

System Fails

Faculty can login again.

System tries to recover to its previous state.

If the record is present,save it again.

If not, Faculty can create a new session record and retake the attendance.

Session Absentees List is incorrect

i. Faculty can modify the attendance of the wrongly marked student(s).

Student not found in course

i. Faculty inform the Administrator about the issue.

ii. Admin verifies the student details.

iii. Admin adds the student to the course.

Administrator is unable to export attendance data

i. Admin logs in to the system again.

ii. Admin enters the correct filters and exports the data.

Special Requirements:

Digital Device with Internet Connectivity.

Robust Recovery when access to the database fails.

Real-time updation for students within 30 seconds 90% of the time.

Technology and Data Variations List:

Faculty can enter attendance either by marking individual students as present/absent or by entering the list of absentees.

Frequency of Occurrence:

Every time the faculty has a class.

Open Issues:

How would substitute teachers take attendance?

What happens when the faculty forgets to take attendance?

2) *Use Case Name***:** Manage Discrepancy

Scope : Attendance Management System

Level : User Goal

Primary Actor : Student

Stakeholders and Interests :

Faculty: Wants an user-friendly digital interface to record, save and manage attendance with no errors that will reduce their administrative workload. Wants an easy way to resolve discrepancies raised by students.

Student : Wants an interface that displays their attendance in real-time and provides statistics for the same. Wants a system that makes it easy to report discrepancies in their attendance.

Administrator: Wants information about unresolved discrepancies.

Preconditions :

Attendance has been recorded and students are notified that they have been marked absent for a session.

Success Guarantee :

Discrepancy is resolved. If a legitimate claim, the attendance record of the student is corrected and the attendance percentage is updated.

Main Success Scenario :

Student accesses the report discrepancy page.
Student enters the relevant details.
Student sends the request.
Faculty access the discrepancy details.
Faculty validates the details of the request.
Faculty modifies the attendance if legitimate claim.
Faculty closes the discrepancy request.

Extensions :

Students are unable to send request
i. Student logs in again.
ii. Student creates a new request.
The modification made by faculty is not reflected to the student
Database Error

Admin is contacted.

Database recovery mechanisms are initiated.

Slow Updation

Student rechecks at a later time.

Faculty are unable to view the discrepancy request

Human Error

Student creates a new request and resubmits the data.

Database Error

Admin is contacted.

Database recovery mechanisms are initiated.

Special Requirements:

1.Digital Device with Internet Connectivity.

2. Robust Recovery when access to the database fails.

3. Students can track the status of their request in real time within 30 seconds of any update 90% of the time.

Frequency of Occurrence:

Sometimes / Rarely

*Open Issues**:***

Students abuse this feature by spamming the faculty with requests.

Faculty ignore the requests.

3) *Use Case Name :* Export Data

Scope : Attendance Management System

Level : Sub Function

Primary Actor : Administrator

Stakeholders and Interests :

Administrator : Wants a system to easily and reliably manage student, faculty and course information while also being able to export relevant attendance data when required.

Institution : Wants a fault tolerant attendance system with punctual,automatic reporting which will reduce the number of conflicts. Wants to receive exported data from the system.

Preconditions:

Administrator is verified and logged in.

Success Guarantee:

Administrator is able to download the required attendance data.

Main Success Scenario:

Administrator accesses the required course.
Administrator enters the required filter options to get relevant data.
Administrator views the retrieved records with the low attendance and unresolved discrepancies/OD requests highlighted.
Administrator downloads the data.

Extensions:

The downloaded data is corrupted
i. The administrator downloads the required data again.
The retrieved data has irrelevant/missing columns.
i. The administrator enters the filters again.
ii. The administrator downloads the required data again.

Special Requirements:

Digital Device with Internet Connectivity.
Robust Recovery when access to the database fails.
System should be able to export the data in csv format and the administrator should have the required software (Microsoft Excel, Open Office Calc) to access the data.

Frequency of Occurrence:

Periodically

Open Issues:

What if the download time is too long because of the large size of the data?
What happens if the administrator wants a consolidated attendance report for a group of courses?

Documentation:

Developing the use case models helped us identify the individual student and faculty functionalities. It also helped us develop an overall structure for the system to satisfy the pre-conditions with respect to the database of students and faculty. This process also helped us build the format in which a lot of the common functionality will be executed.

Attendance Management System
Domain Model Diagram And Class Diagram

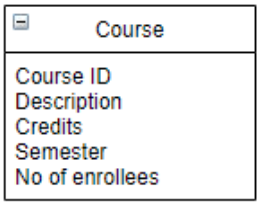
Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

Aim:

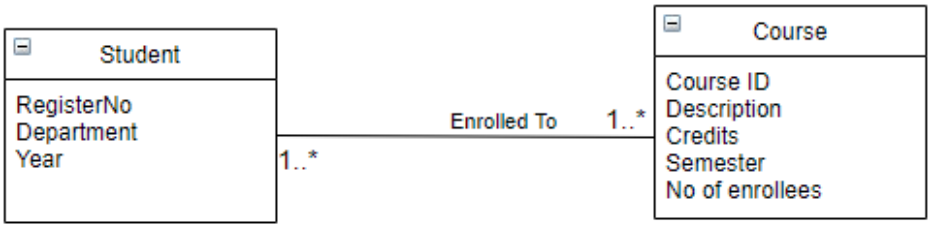
To create a domain model diagram and class diagram of the Attendance Management System.

UML Notations for Domain Model Diagram:

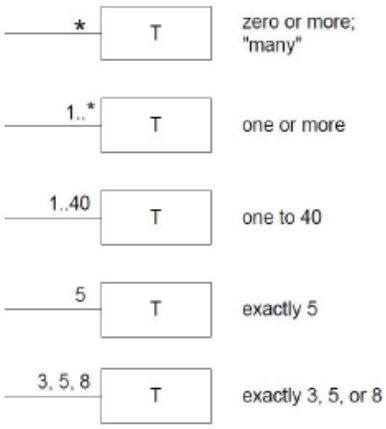
Description Class



Association Line

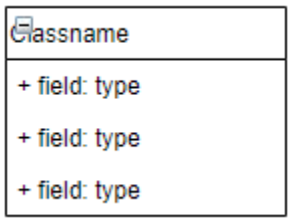


Association Line Multiplicity

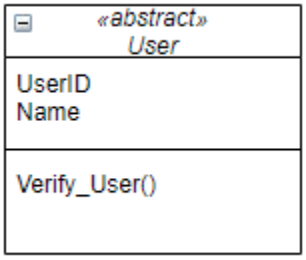


UML Notations for Class Diagrams:

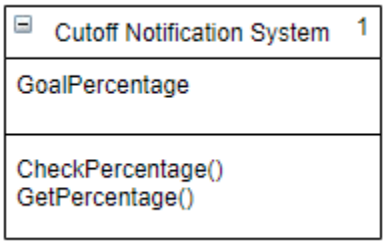
Class



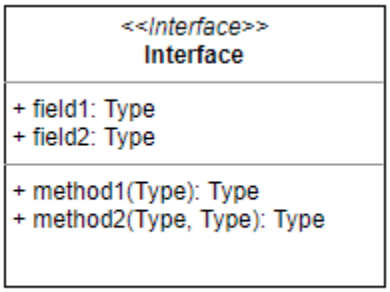
Abstract Class



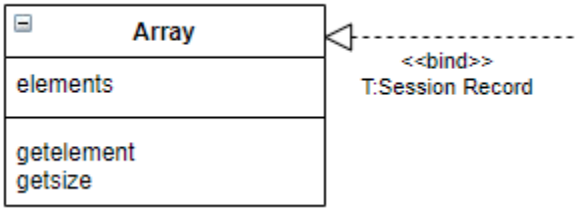
Singleton Class



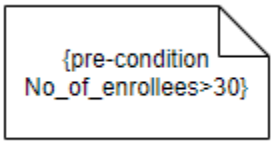
Interface



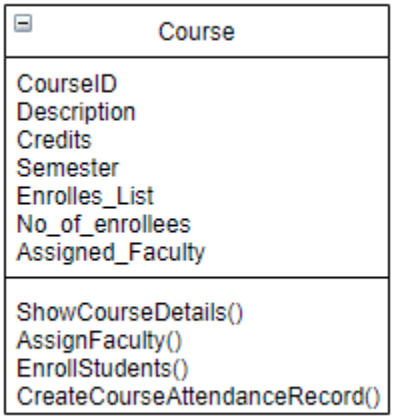
Template

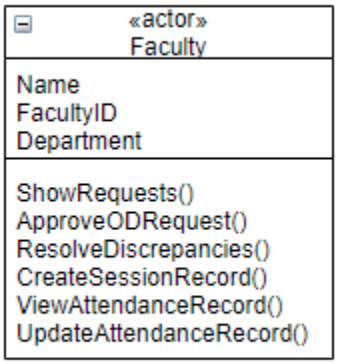


Constraints



Association Text

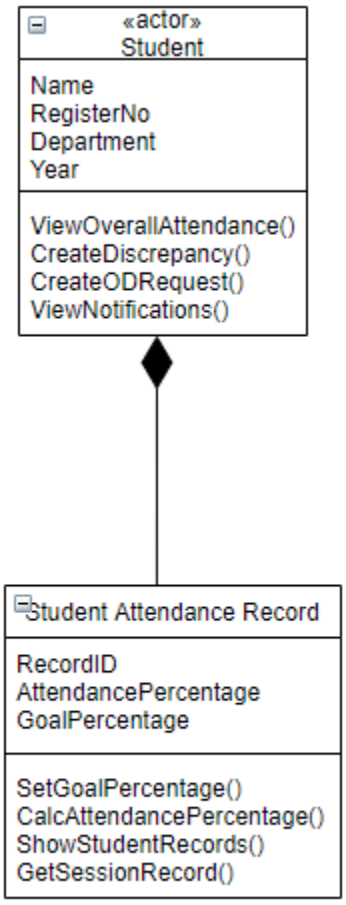




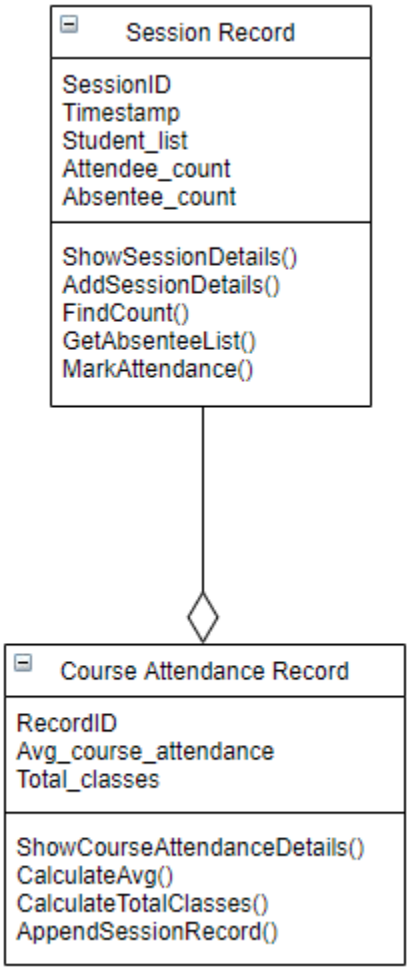
Association Line

Both:

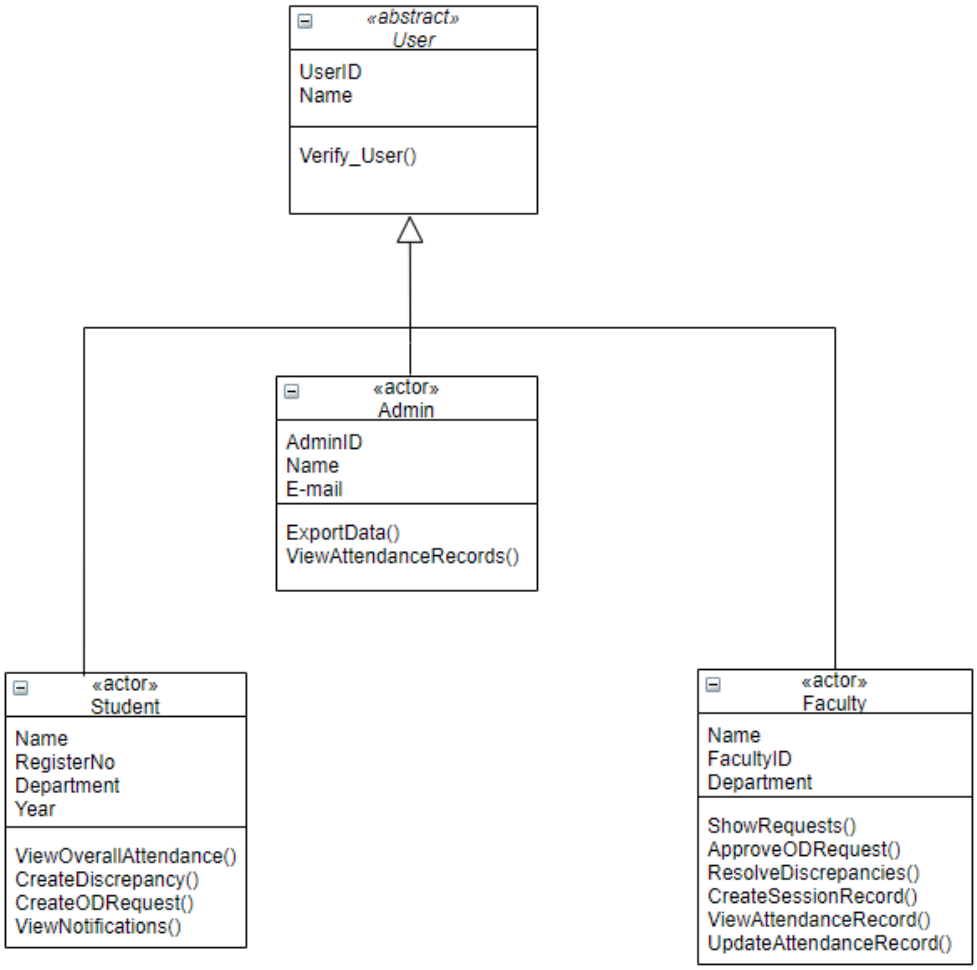
Composition



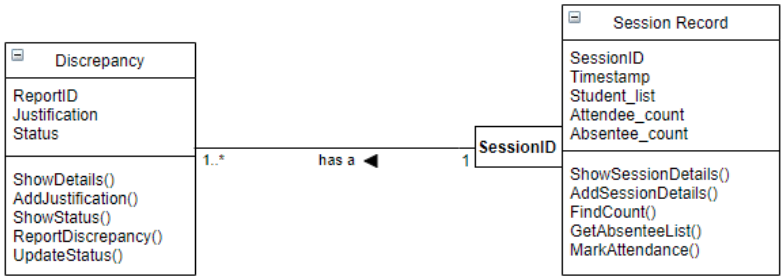
Aggregation



Generalization



Qualified Association



Identification of Classes:

Conceptual Class Category List

Conceptual Class Category	Examples
Transactions	*Notification System*
Roles of People	*Student, Faculty, Admin*
Specification, designs or description of things	*Student Attendance Record, OD*
Events	*Discrepancy*
Record of Transaction, Process, Service	*Attendance Database*

Noun Phrases

A student platform which allows the **students** to freely access their **attendance records**. Through this, the students can keep track of the **status** of their attendance and quickly report any **discrepancies**. This also helps them make decisions regarding their **attendance percentage**. A faculty platform which lets the **faculty** take and store **attendance** with ease.It also minimizes the **errors** and saves a lot of time for the faculty. A **Notification system** that alerts the students when their attendance goes below the **cutoff percentage**. A Feature to manage the requests and approvals of **ODs**.

Conceptual Class List:

Student
Attendance Record
Status - Rejected (Attribute of class)
Discrepancies
Attendance Percentage - Rejected (Attribute of class)
Faculty
Attendance - Rejected (Vague)
Errors - Rejected (Too general)
Notification System
OD

Identification of Associations:

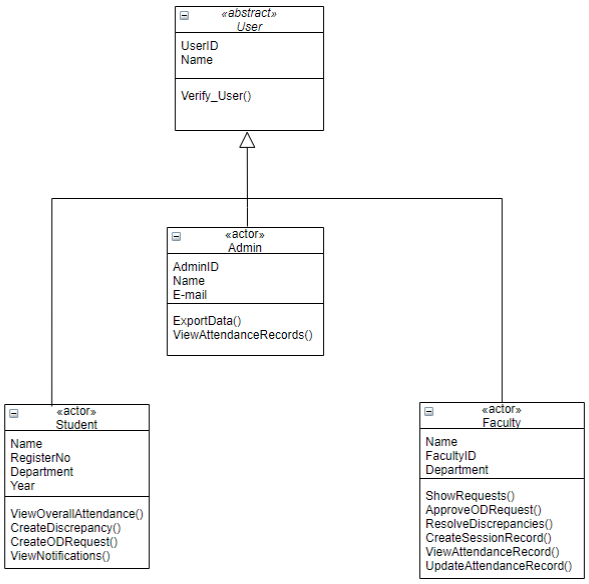
Association Category List

Category	**Examples**
A is a logical part of B	*StudentAttendanceRecord—CourseAttendanceRecord*
A is logically contained in B	*SessionRecord-CourseAttendanceRecord*
A is a member of B	*Student-Course*
A is an organizational subunit of B	*CutoffNotifcationSystem-NotificationSystem*
A uses or manages B	*Admin-AttendanceDatabase*
A communicates with B	*Student-Faculty*
A is related to a transaction B	*Student-Notification*
A is owned by B	*CourseAttendanceRecord-Course*

Definition of Associations and Notations:

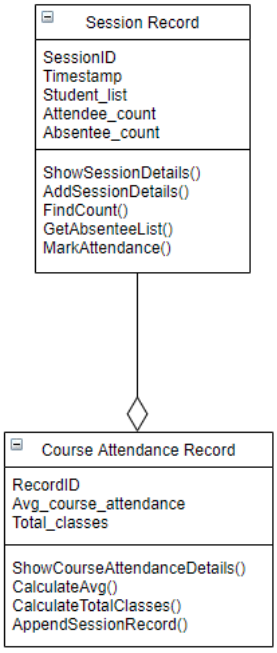
a. Generalization

In the generalization process, the common characteristics of classes are combined to form a class in a higher level of hierarchy, i.e., subclasses are combined to form a generalized super-class. It represents an “is – a – kind – of” relationship.



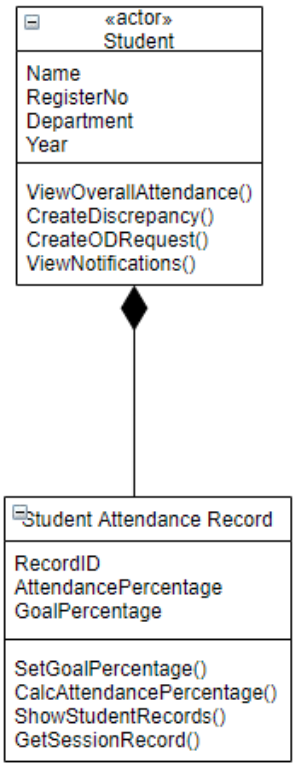
b. Aggregation

Aggregation is referred to as a "part–of" or "has–a" relationship, with the ability to navigate from the whole to its parts. An aggregate object is an object that is composed of one or more other objects.



c. Composition

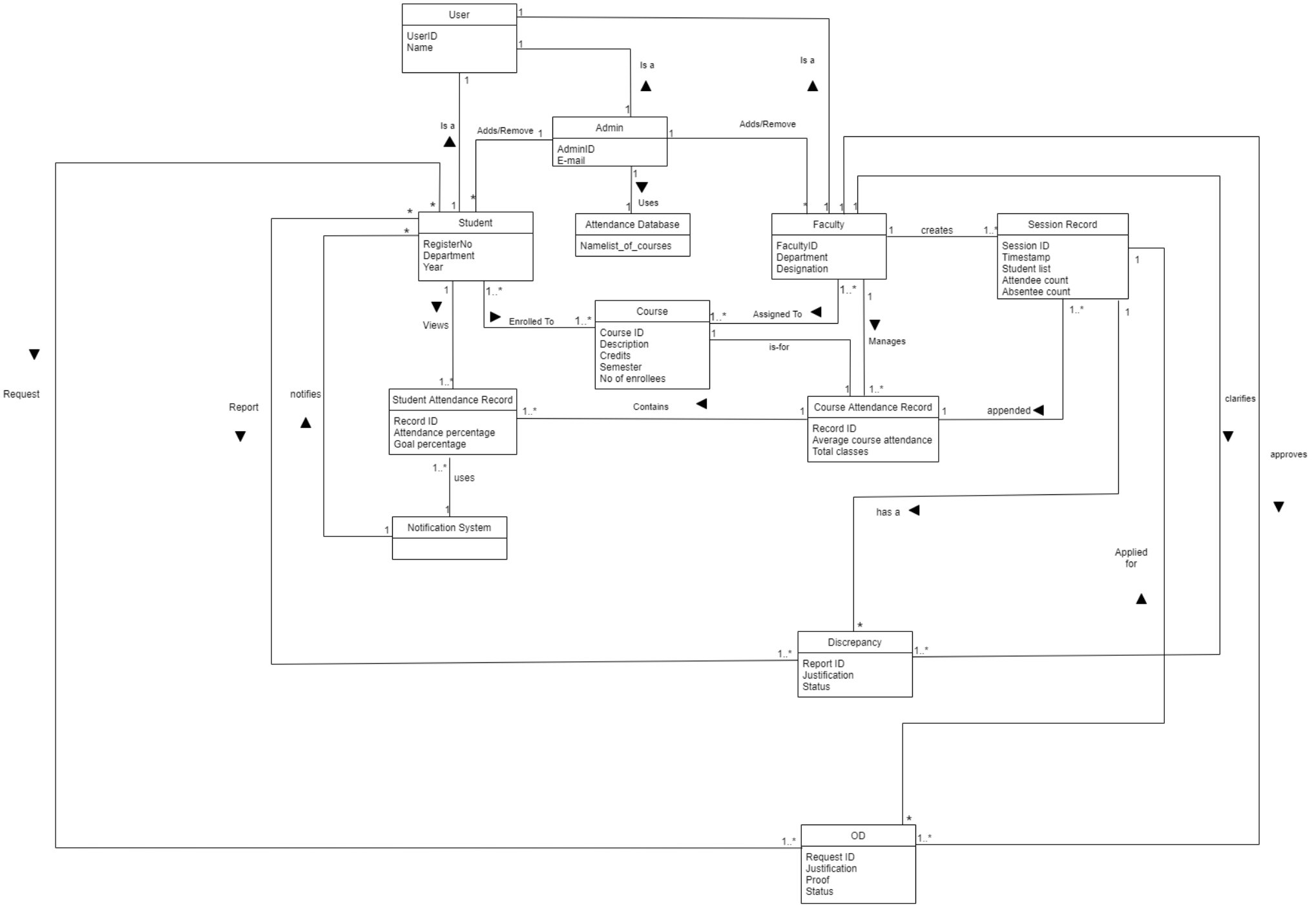
Composition is a special case of aggregation. In a more specific manner, a restricted aggregation is called composition. When an object contains the other object, if the contained object cannot exist without the existence of a container object, then it is called composition.



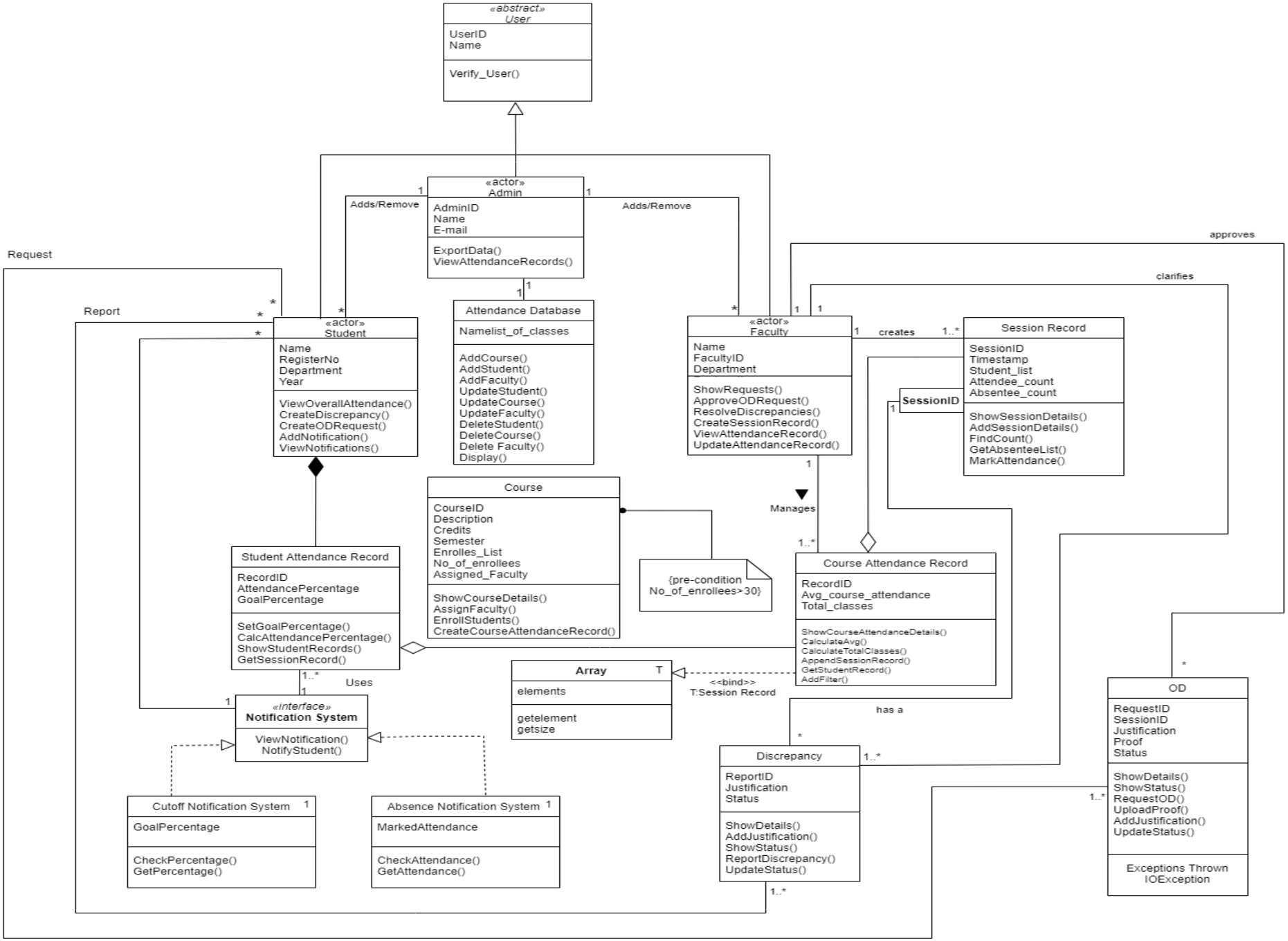
Multiplicity based Associations

a. Notification System 1 ----- * Student
b. Discrepancy 1..* ----- * Student
c. CourseAttendanceRecord 1 ----- 1..* StudentAttendanceRecord
d. Admin 1 ----- 1 Database
e. Student 1..*---- 1..* Course

Domain Model:


User
UserID
Name
Is a
Is a
Is a
Admin
AdminID
E-mail
Adds/Remove
Adds/Remove
Uses
Attendance Database
Namelist_of_courses
Student
RegisterNo
Department
Year
Faculty
FacultyID
Department
Designation
creates
Session Record
Session ID
Timestamp
Student list
Attendee count
Absentee count
Views
Enrolled To
Course
Course ID
Description
Credits
Semester
No of enrollees
Assigned To
is-for
Manages
Request
Report
notifies
Student Attendance Record
Record ID
Attendance percentage
Goal percentage
Contains
Course Attendance Record
Record ID
Average course attendance
Total classes
appended
clarifies
approves
uses
Notification System
has a
Applied for
Discrepancy
Report ID
Justification
Status
OD
Request ID
Justification
Proof
Status

Class Diagram:



Documentation:

Developing the domain models and class diagram helped us identify the individual classes and the hierarchy of classes. It helped us understand the associations between the various classes. It helped us identify the main attributes and methods of the classes. This process has brought us one step closer to the actual implementation of our system.

Attendance Management System
Interaction Diagram

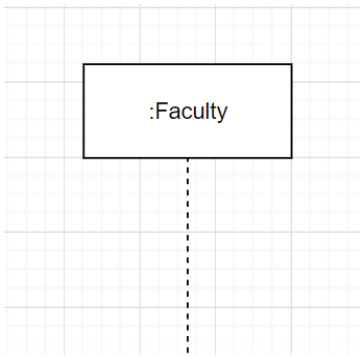
Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

Aim:

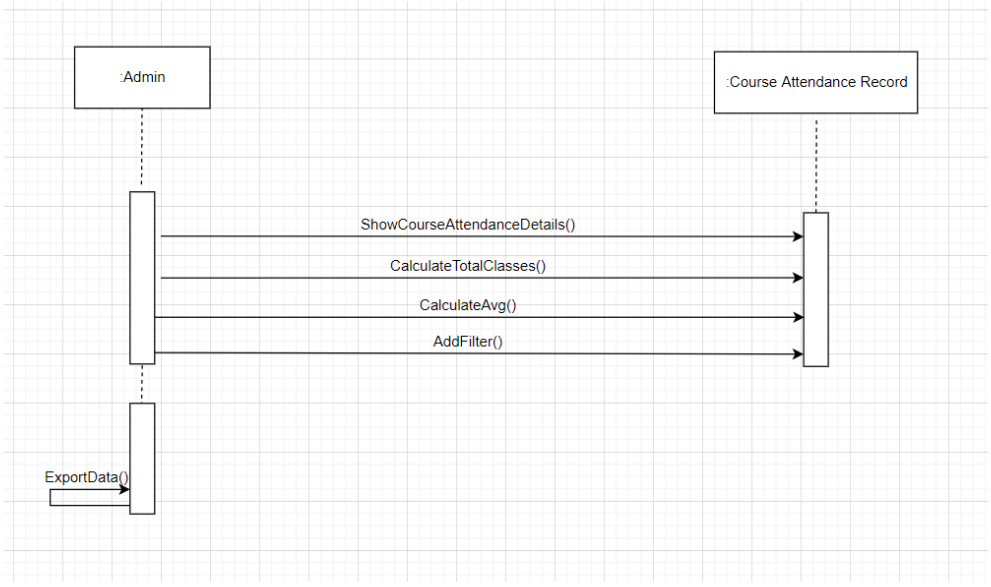
To create interaction diagrams- sequence and communication diagrams of the Attendance Management System.

UML Notations for Sequence Diagram:

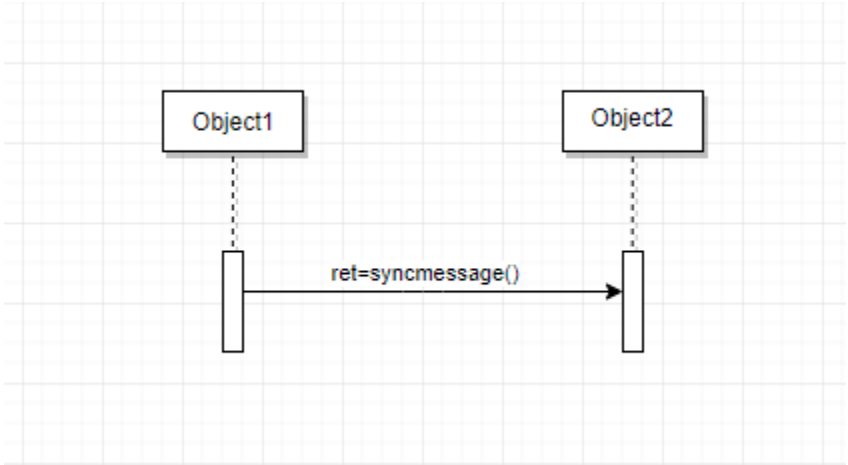
LifeLine Notations:



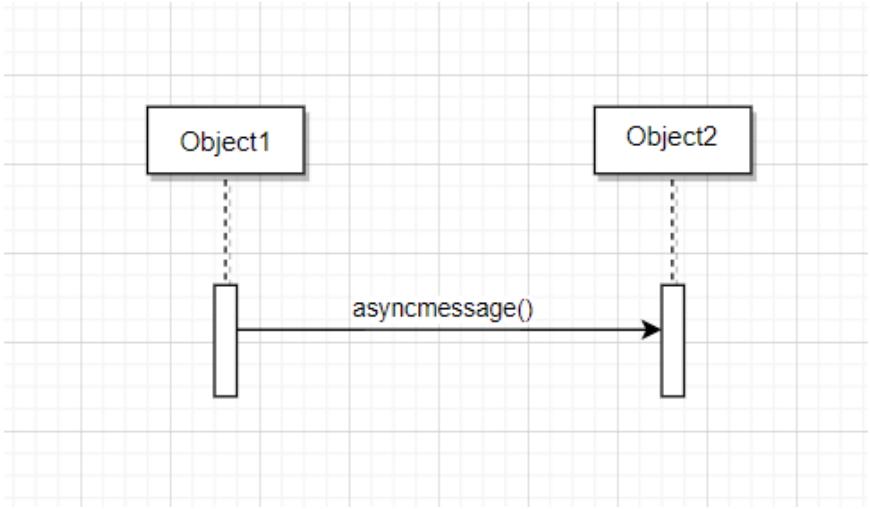
Activation Bars:



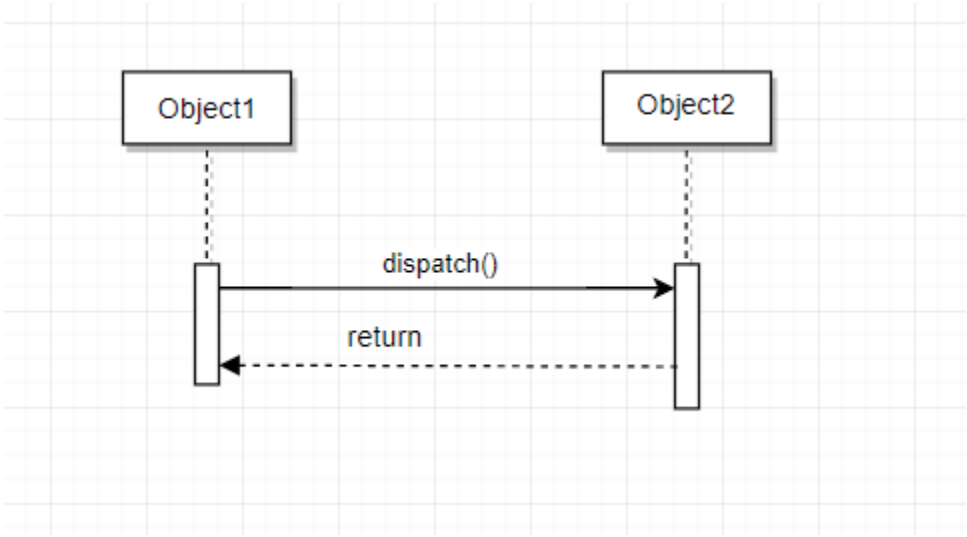
Synchronous Message:



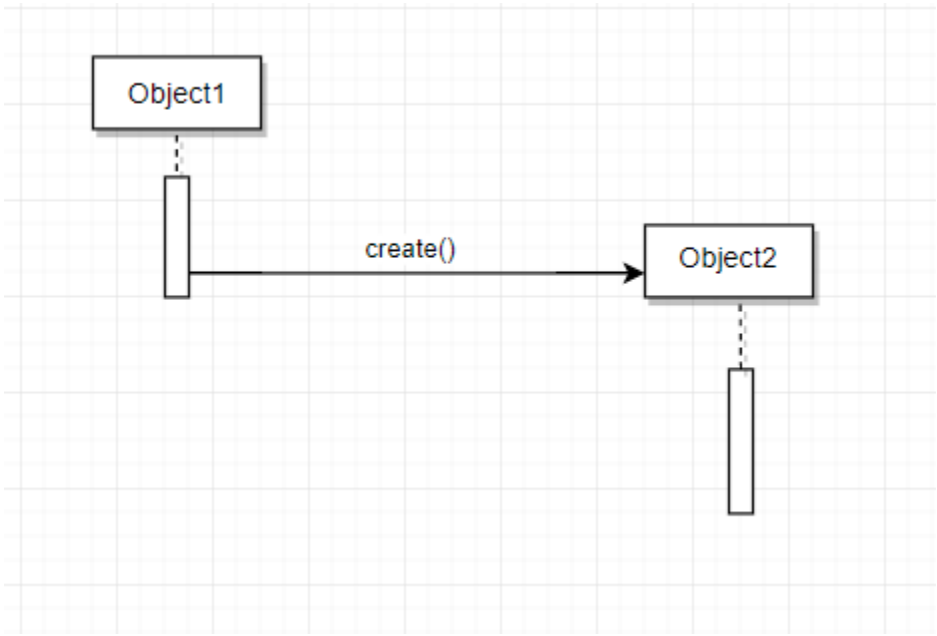
Asynchronous Message:



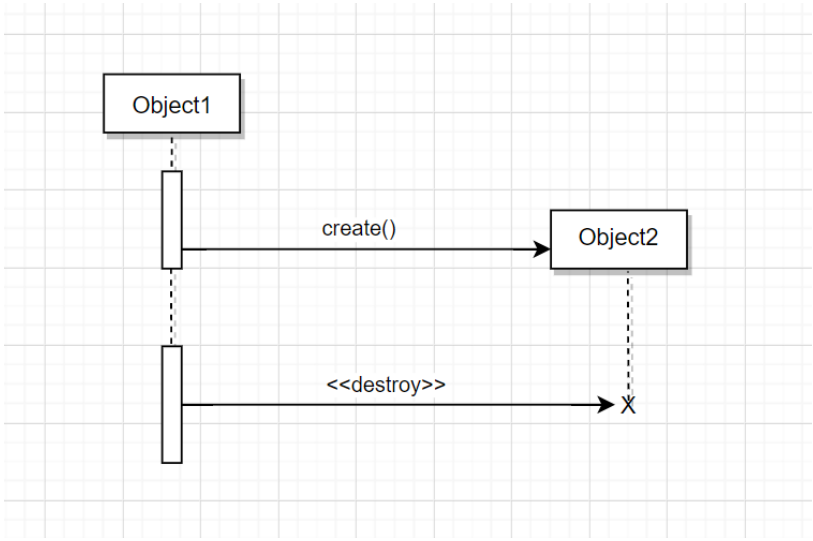
Return Message:



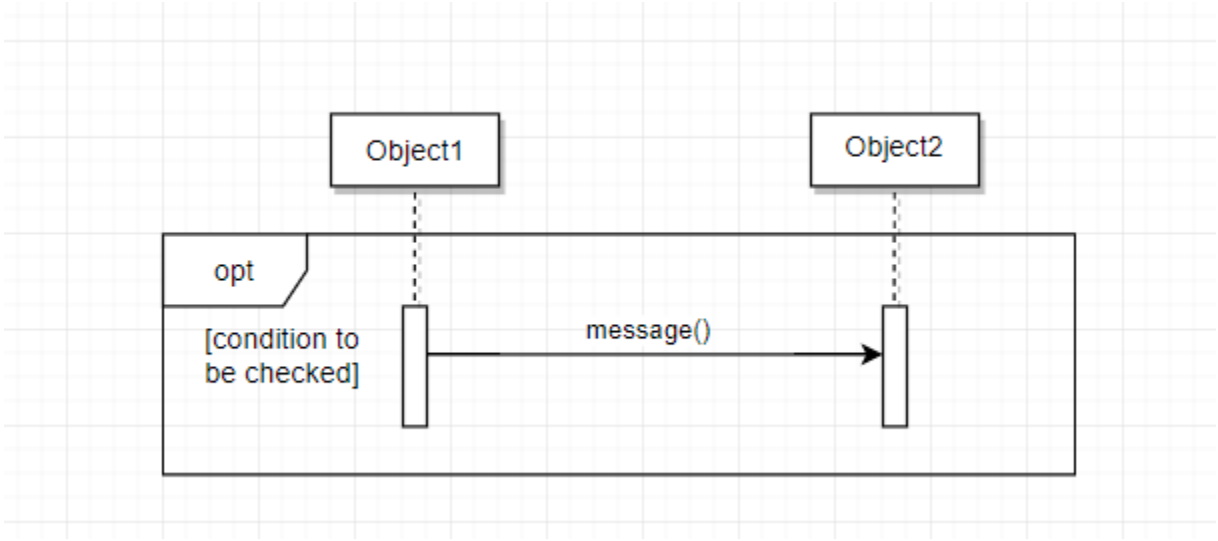
Create:



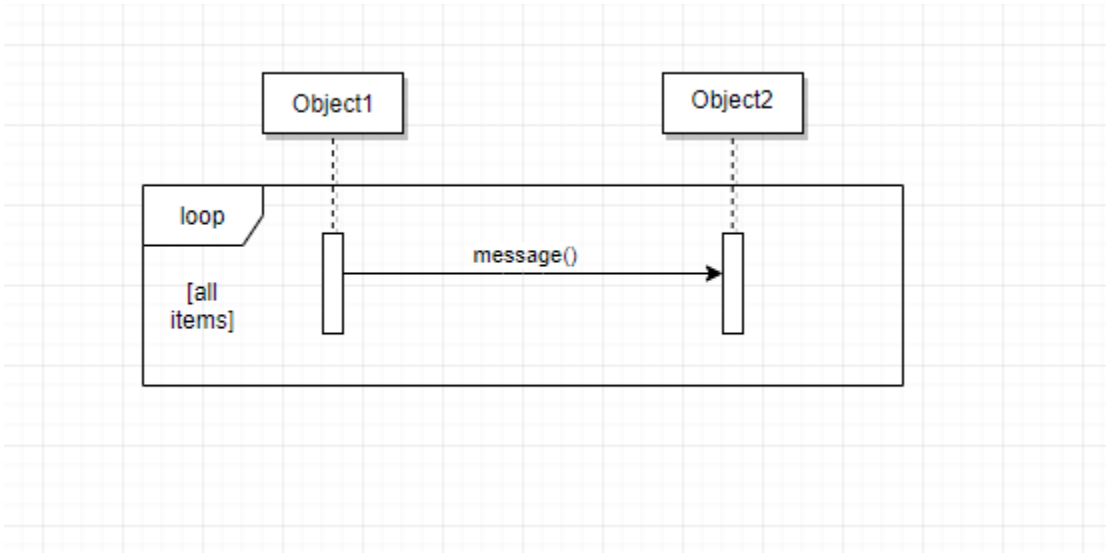
Destroy:



Option:

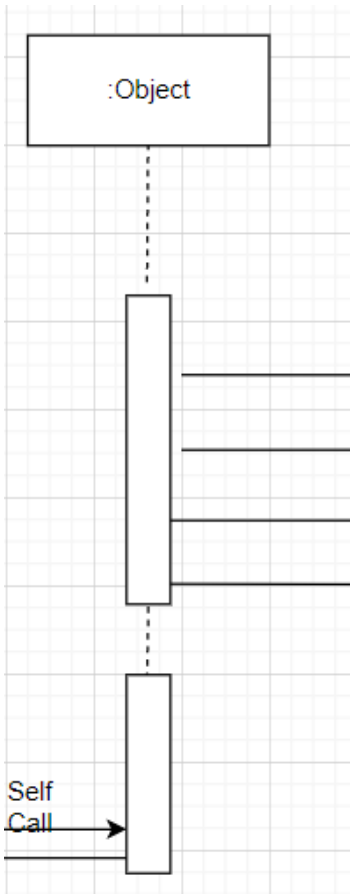


Loop:

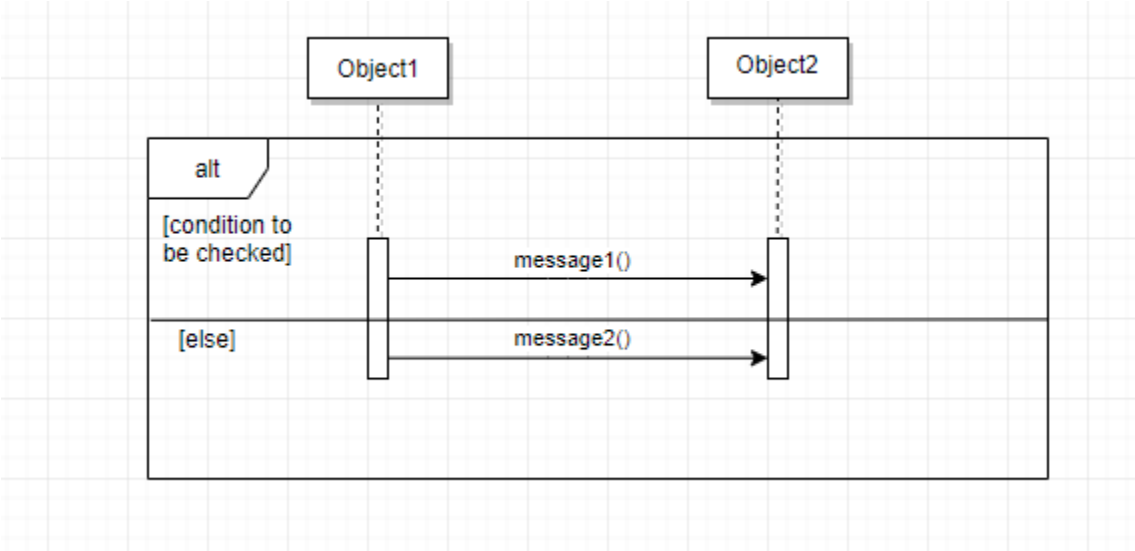


\

Reflexive Message:

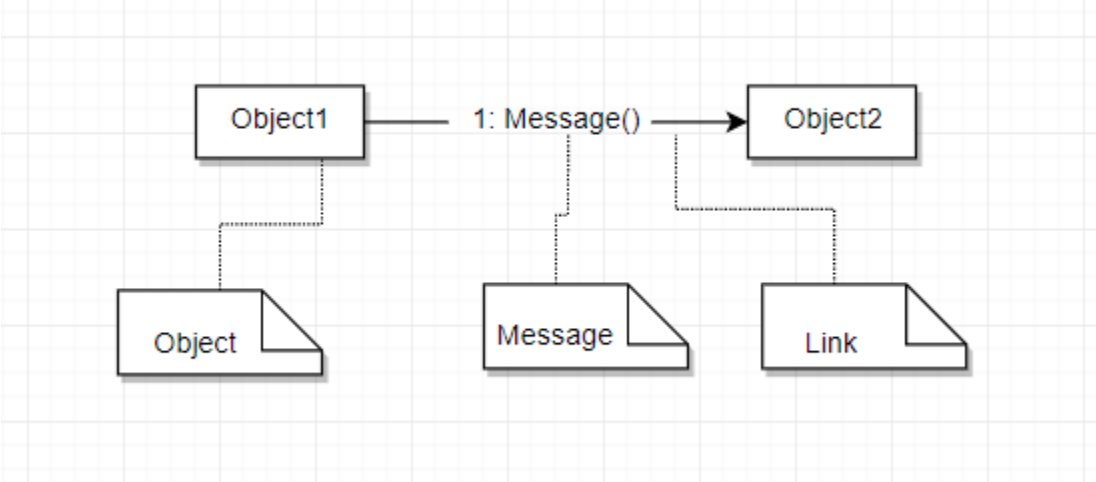


Alternative:

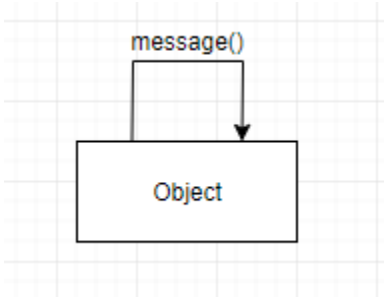


Communication Diagram Notations:

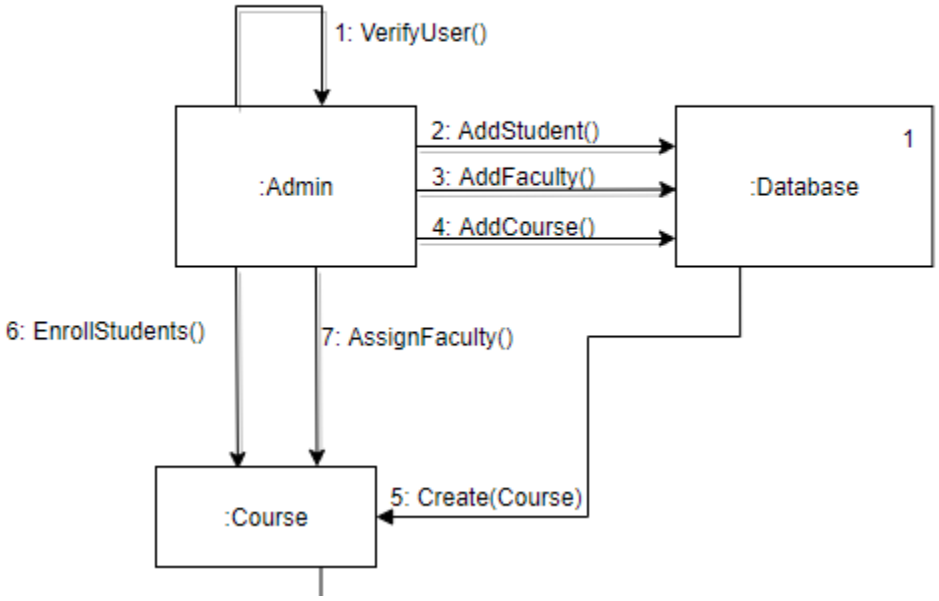
Messages, Link and Objects:



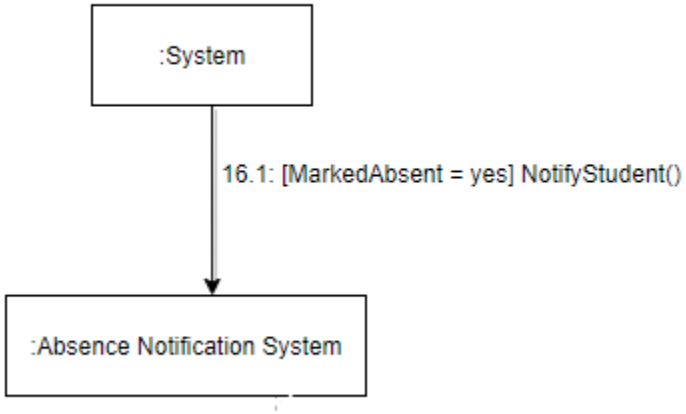
Message to this/self:



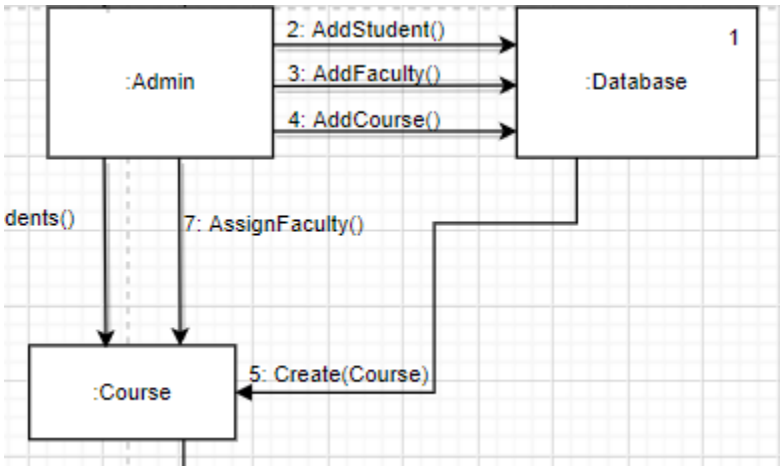
Message Number Sequencing:



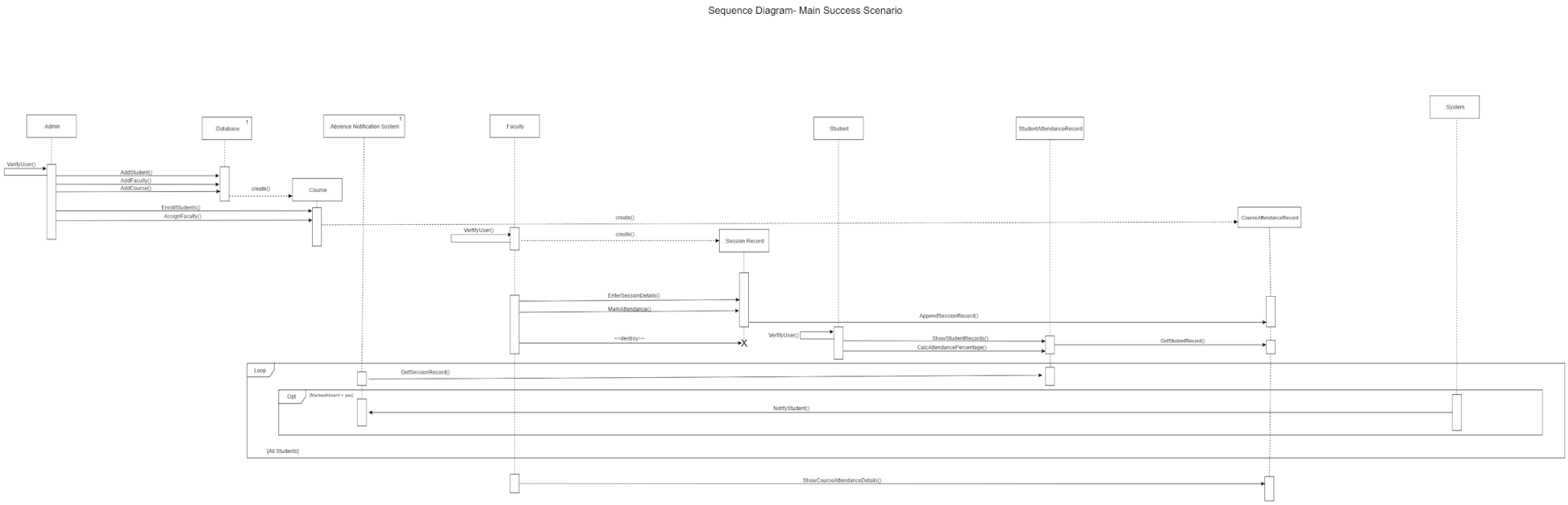
Conditional Message:



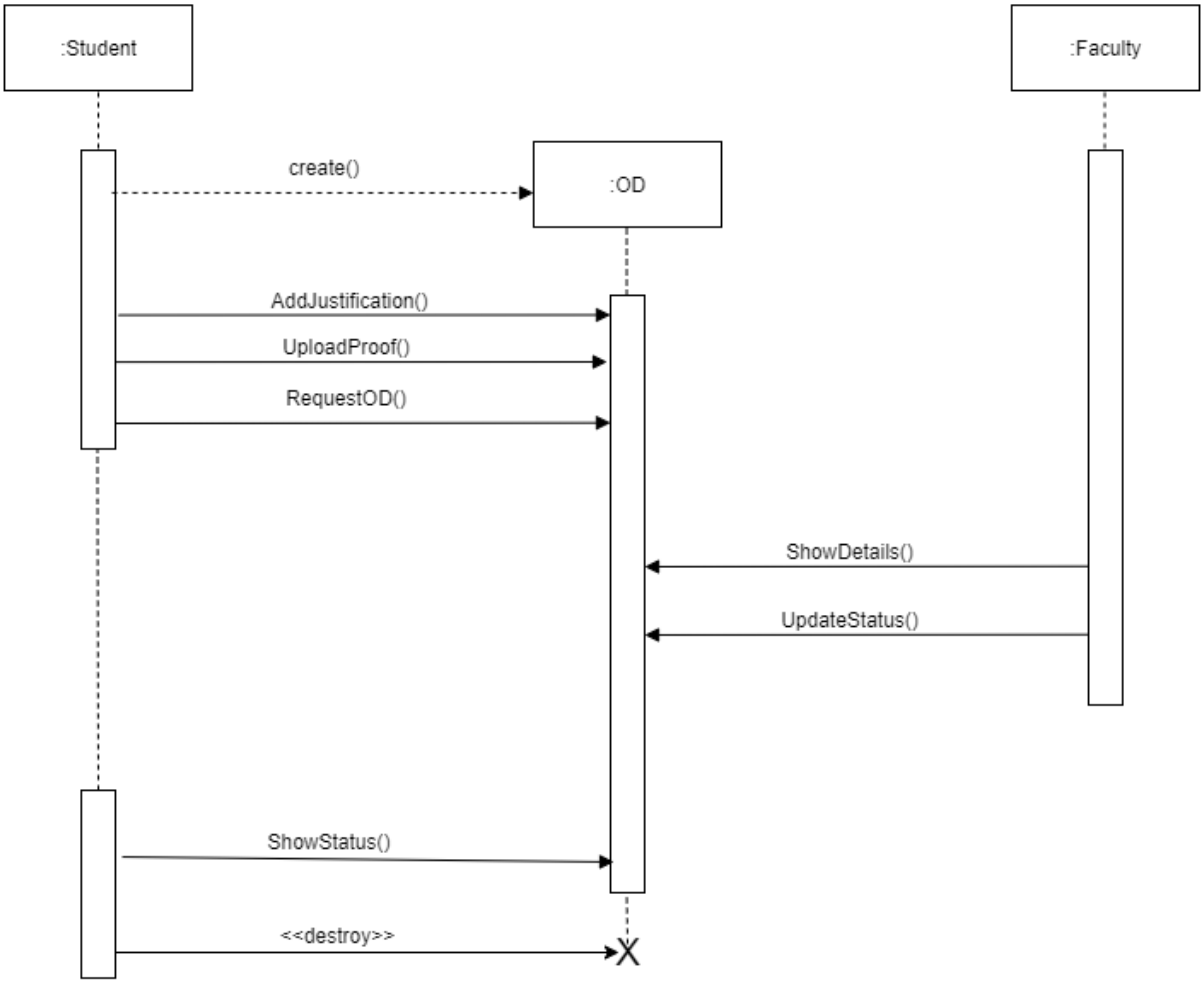
Create:



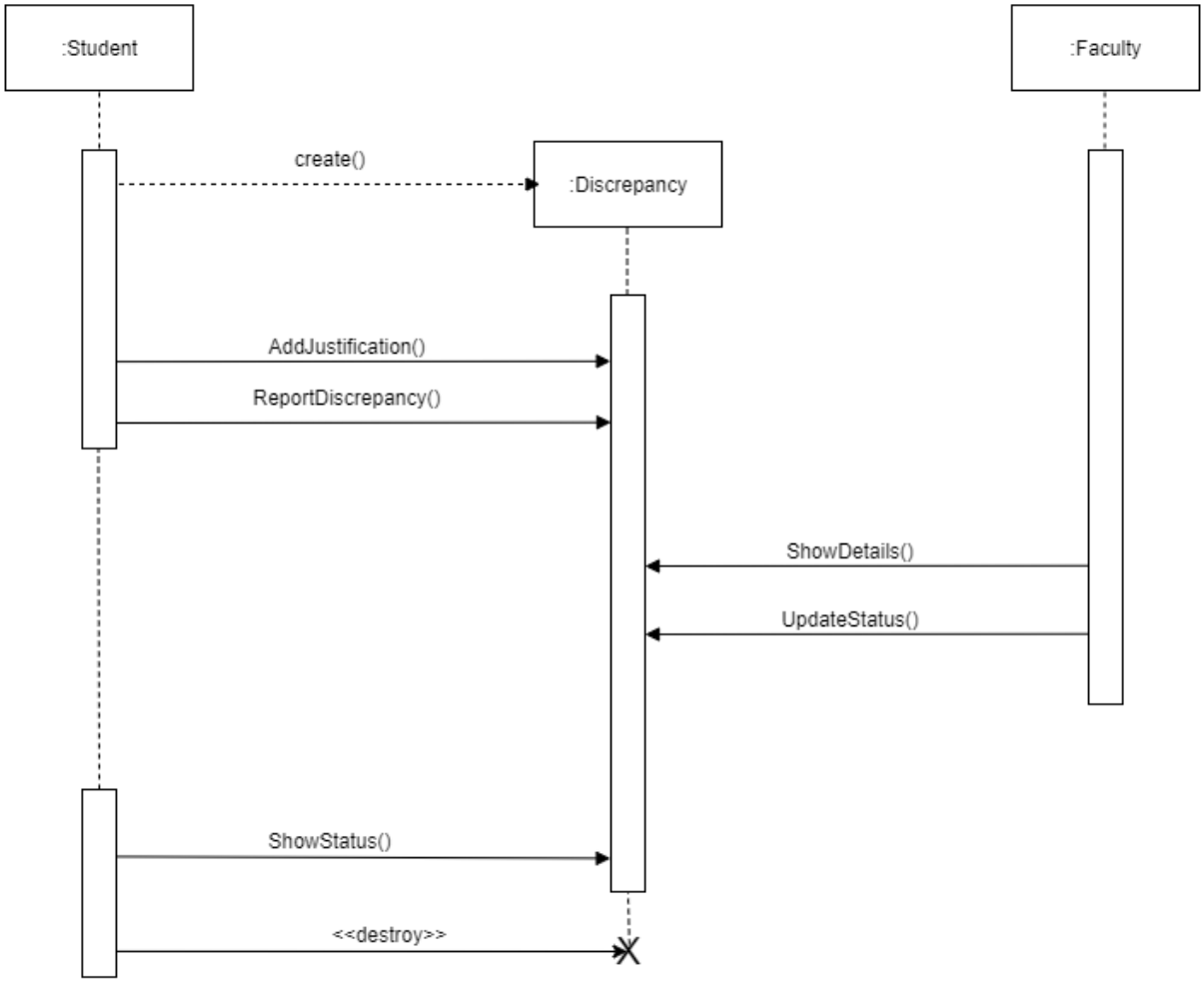
Sequence Diagrams:



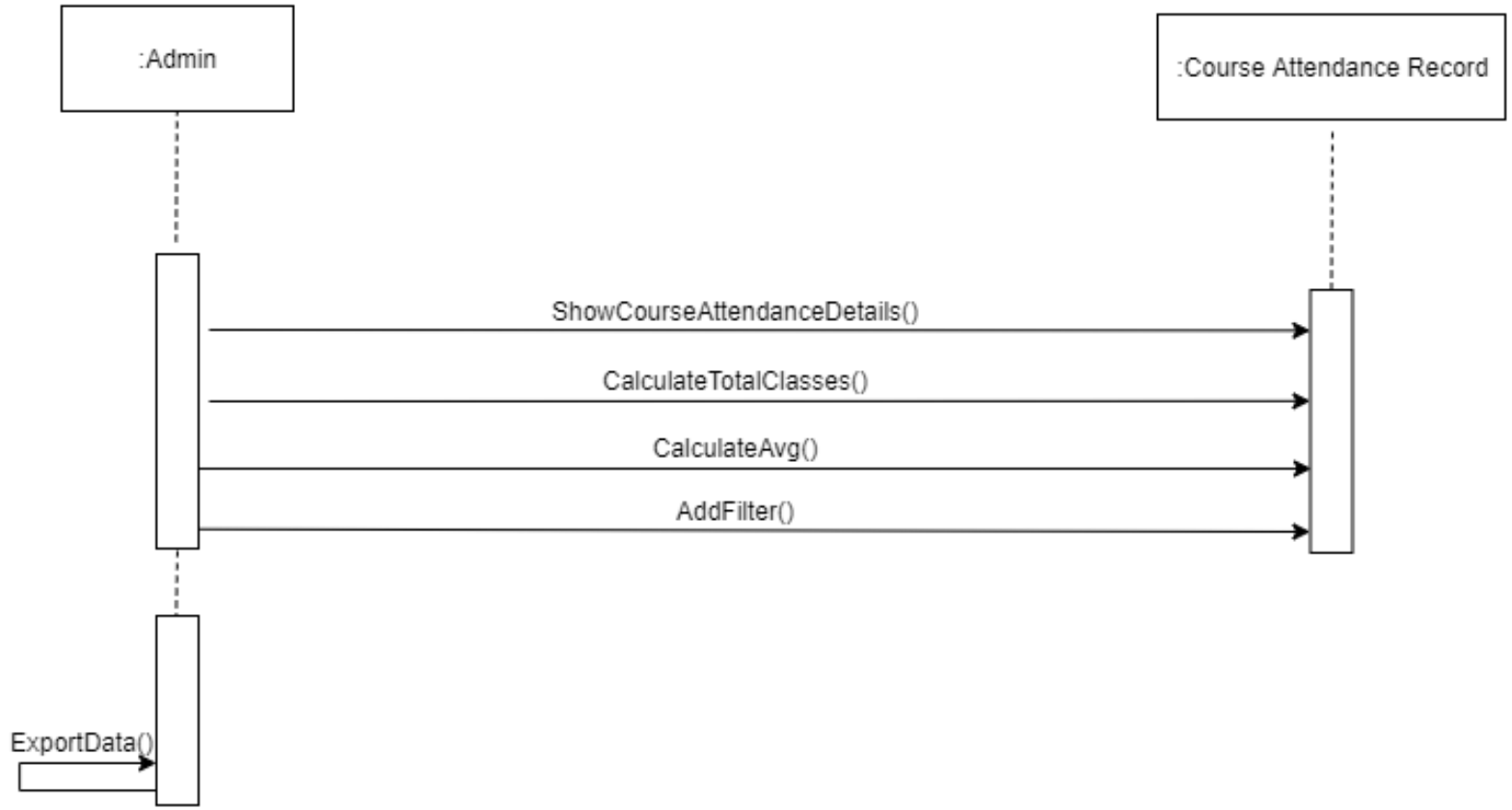
Sequence Diagram- Alternate Scenario: Request OD



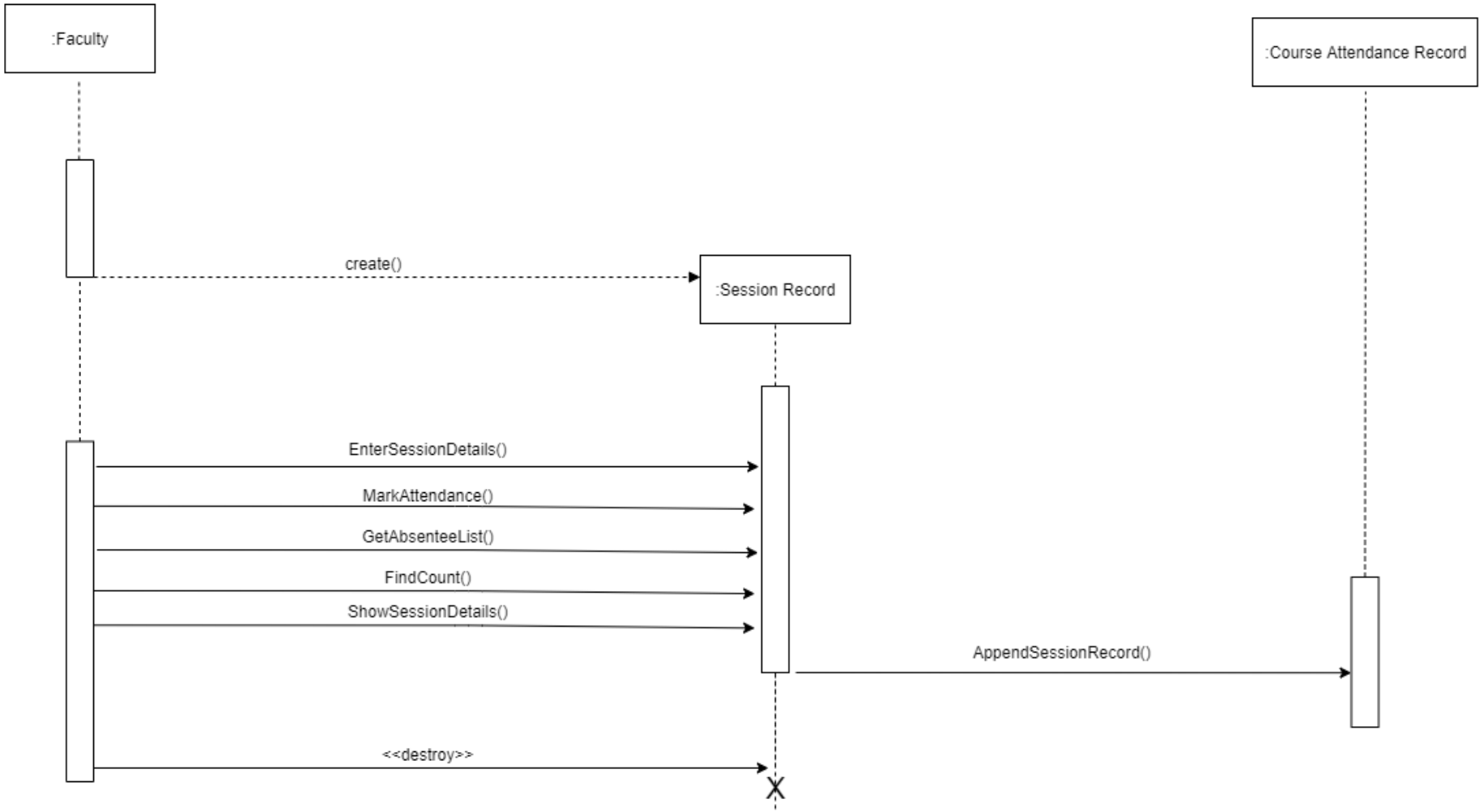
Sequence Diagram- Alternate Scenario: Report Discrepancy



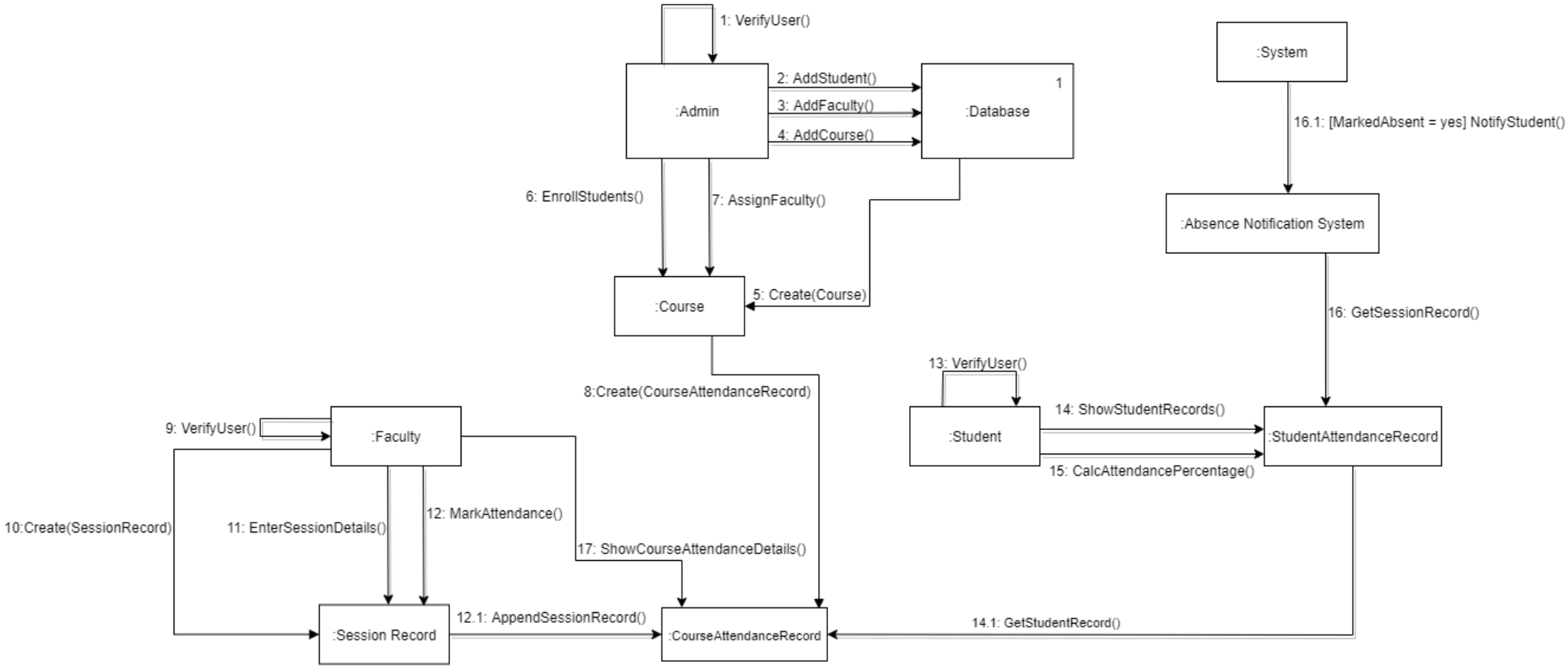
Sequence Diagram- Subfunction: Export Data



Sequence Diagram- Subfunction: Record Attendance



Communication Diagram - Main Success Scenario



Documentation:

Developing the sequence diagrams and communication diagrams helped us identify how the objects interact with each other. It helped us understand the timeline of processes that will occur in our system and the message passing within the various objects. This process has brought us one step closer to the actual implementation of our system.

Attendance Management System
State Machine Diagram And Activity Diagram

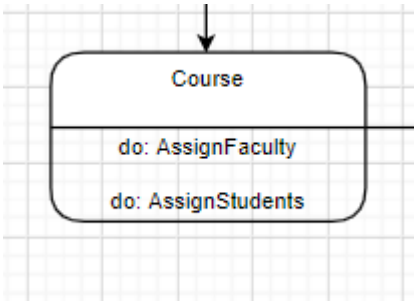
Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

Aim:

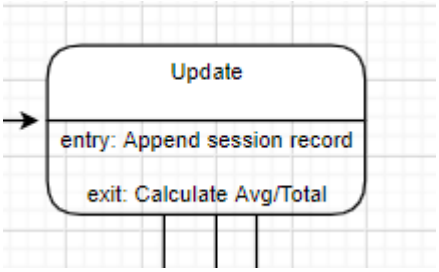
To create a state machine diagram and activity diagram of the Attendance Management System.

UML Notations for State Machine Diagram:

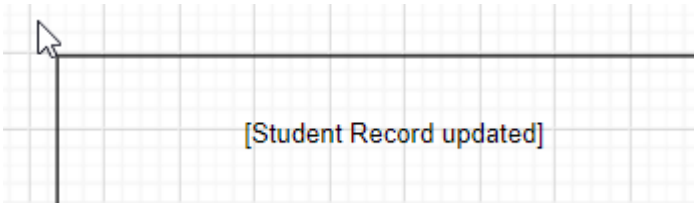
State



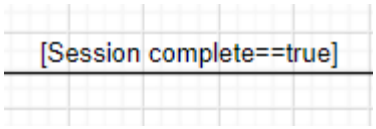
Entry and Exit



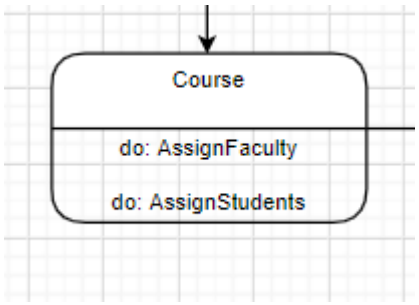
Activity



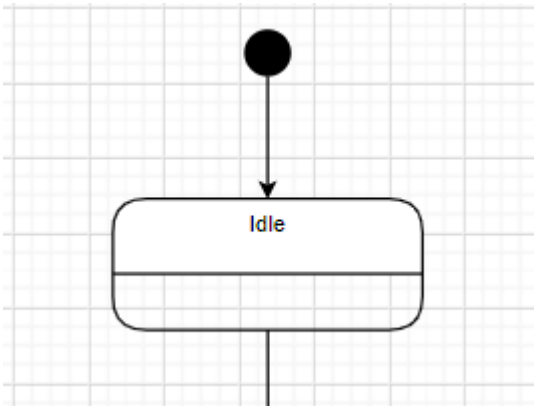
Guard Condition



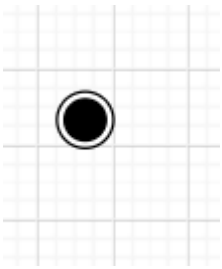
Action



Start State

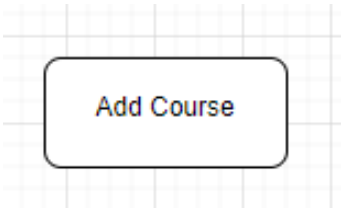


End State

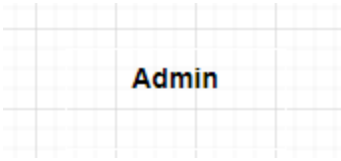


UML Notations for Activity Diagram:

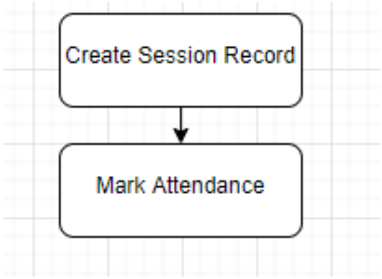
Activity



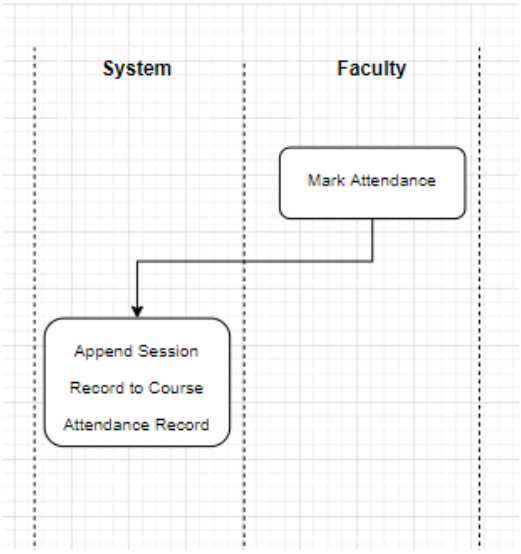
Object Node



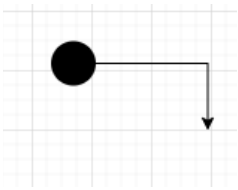
Control Flow/Transition



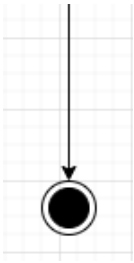
Object Flow



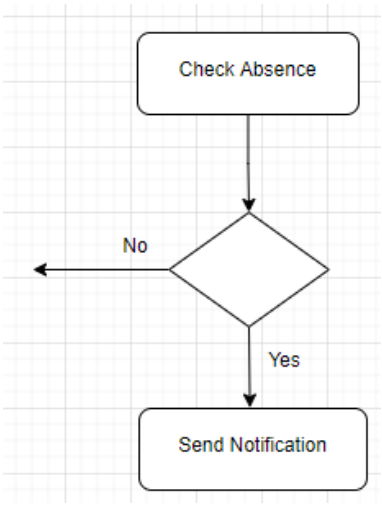
Initial Node



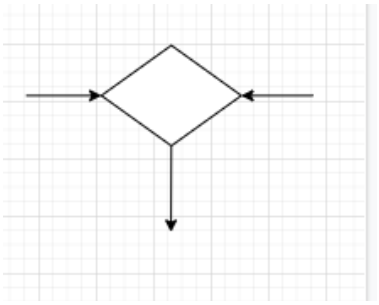
Final Node



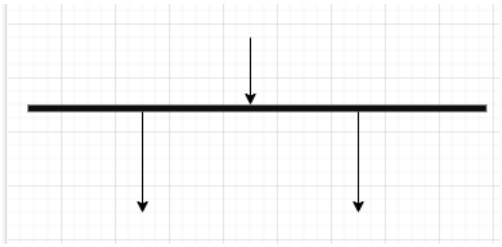
Decision Node



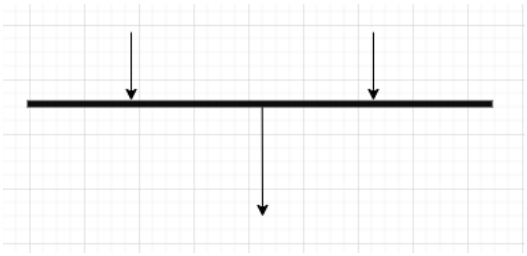
Merge Node



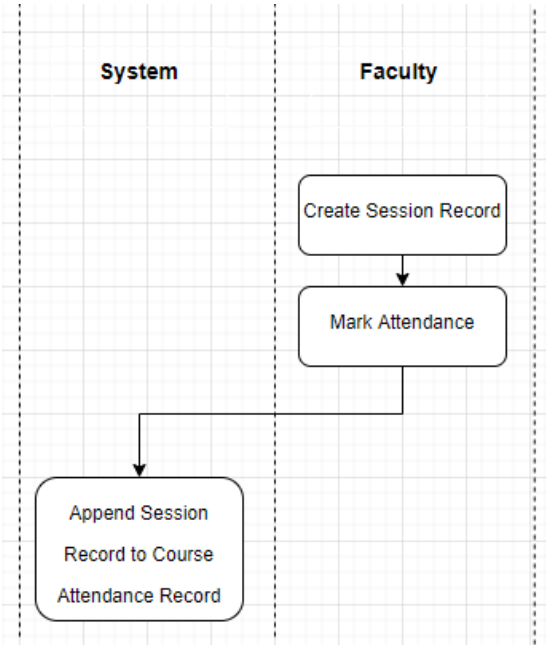
Fork Node



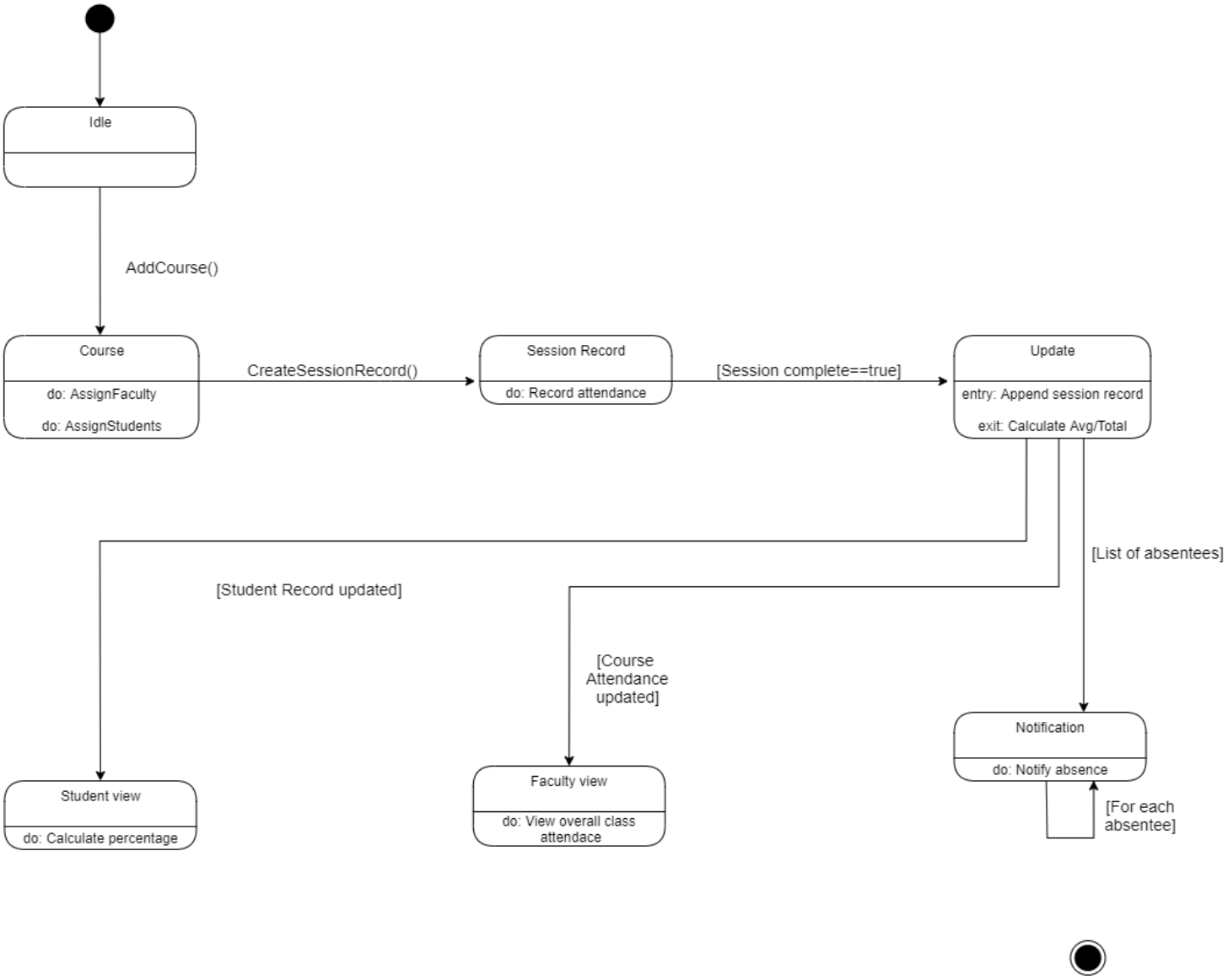
Join Node



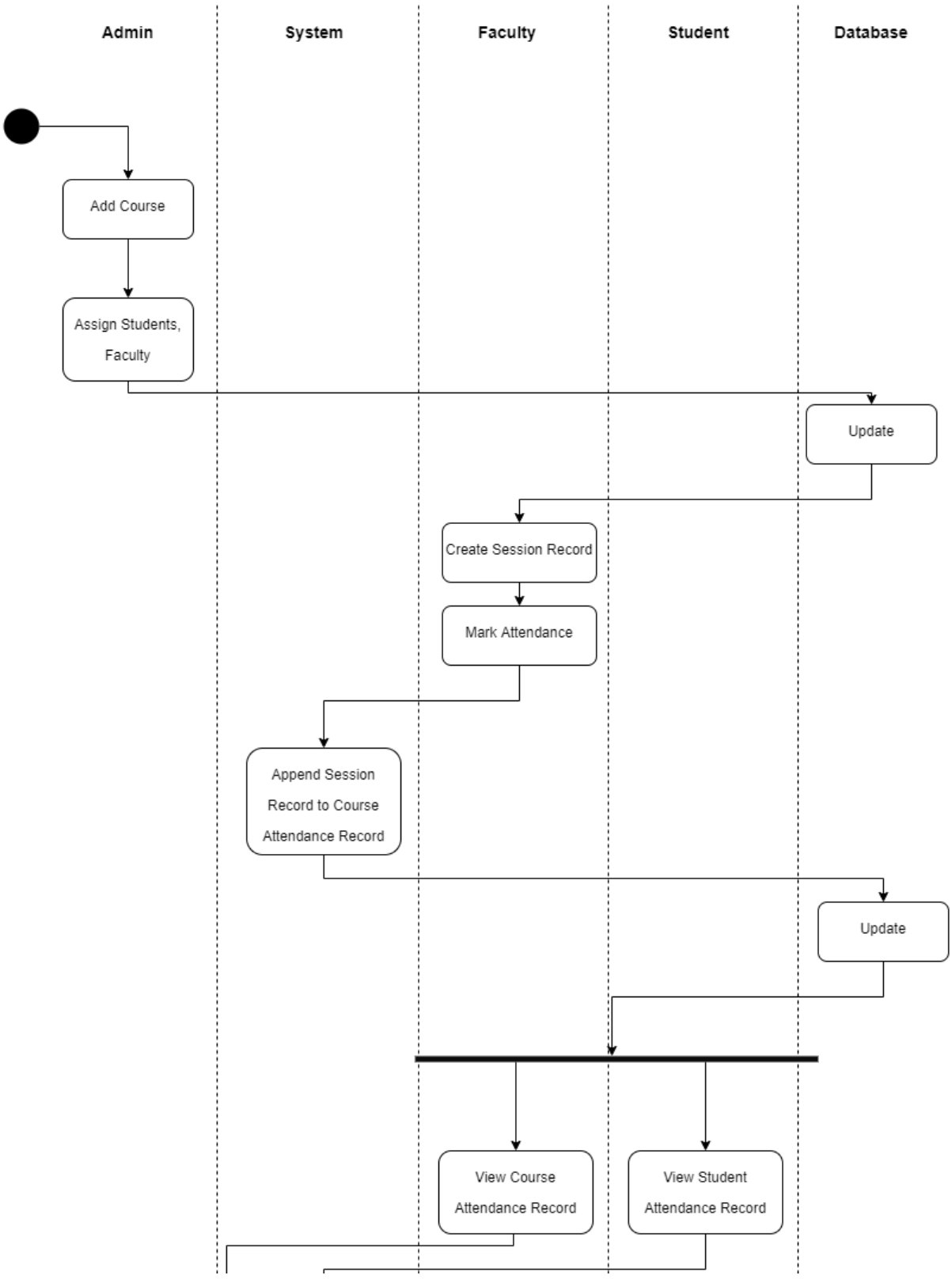
Swimlane and Partition

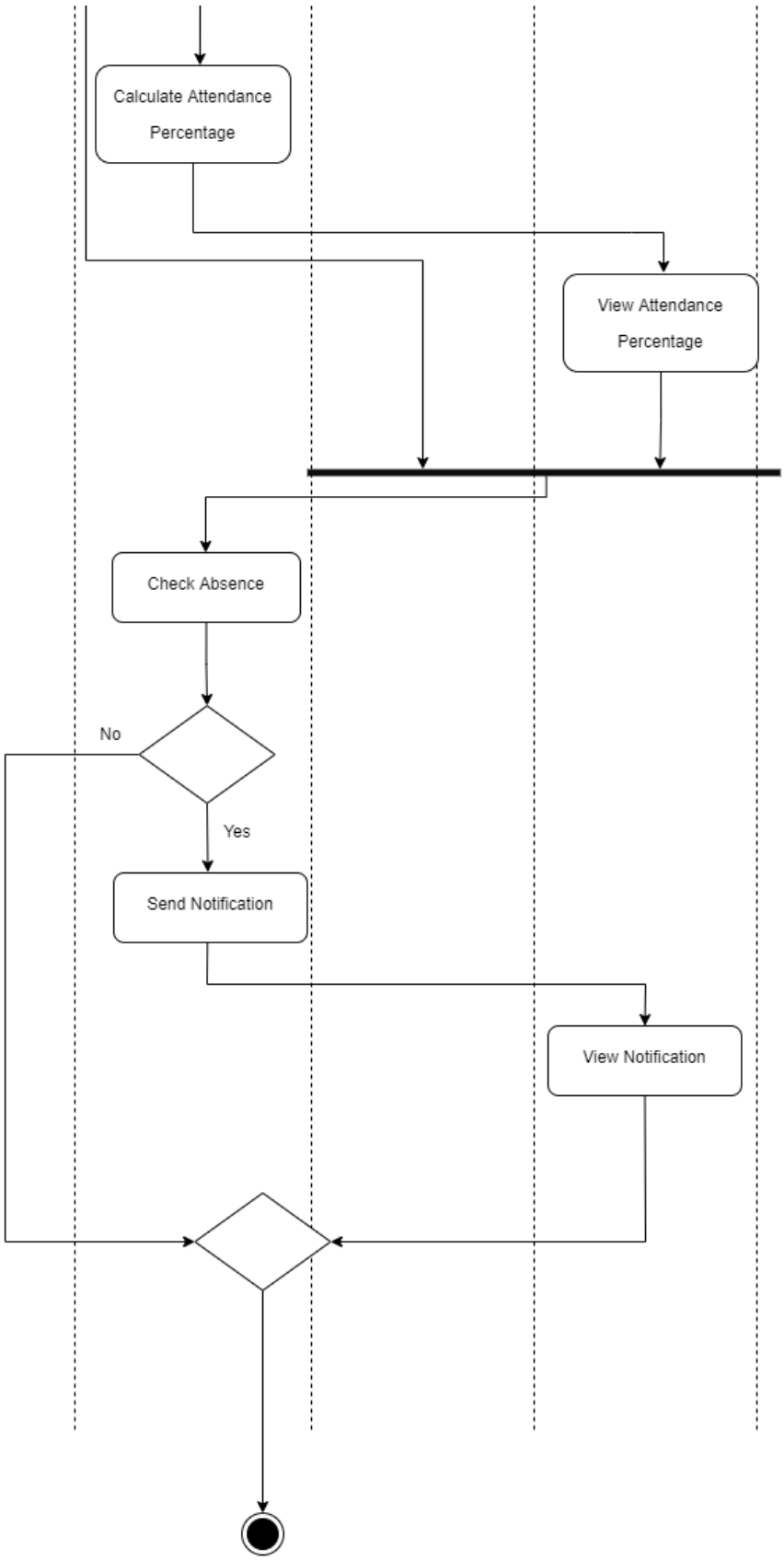


State Diagram



Activity Diagram




Calculate Attendance Percentage
View Attendance Percentage
Check Absence
No
Yes
Send Notification
View Notification

Documentation:

Developing the State Machine Diagram and Activity Diagram helped us identify the tasks of each user (object) involved. It gave us clarity on how to classify the different states that our system will be in throughout its working. It also helped us further our understanding of the timeline of the system's dynamic behavior. This process has brought us one step closer to the actual implementation of our system.

Attendance Management System
Package Diagram

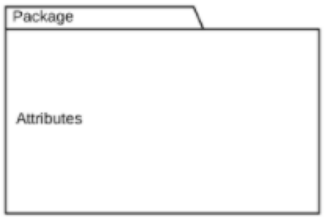
Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

Aim:

To create a package diagram of the Attendance Management System.

UML Notations for Package Diagram:

Package:



Dependency:

Access:

Import:

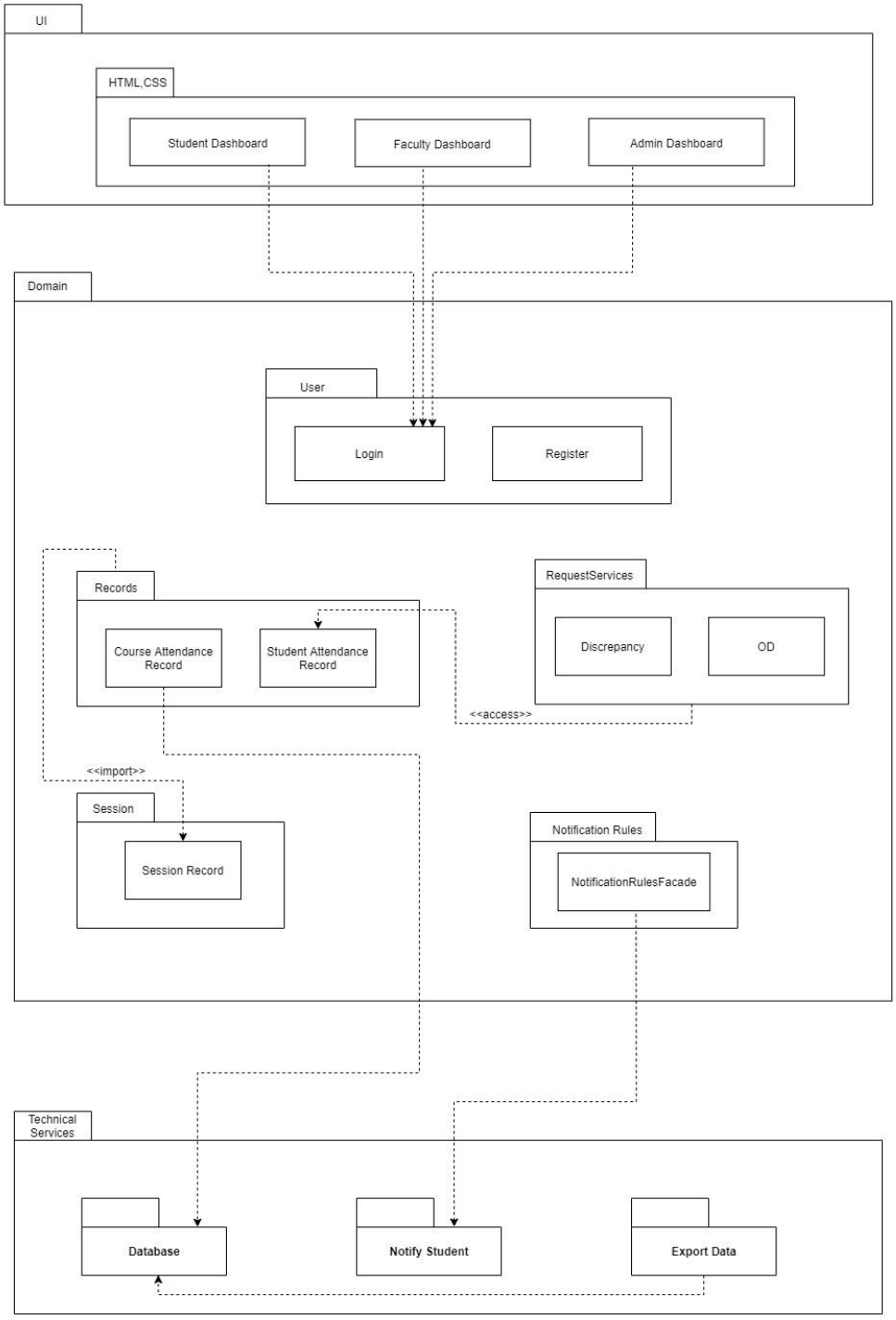
List of Packages:

1. **UI**
    a. HTML,CSS

2. **Domain**
    a. User
    b. Records
    c. RequestServices
    d. Session
    e. NotificationRules

3. **TechnicalServices**
    a. Database
    b. NotifyStudent
    c. ExportData

Package Diagram



Documentation:

Developing the package diagram helped us identify how the individual classes can be grouped together. It helped us understand the dependencies between the various packages. It helped us identify how the packages can be separated into UI,Domain and Technical Service. This process has brought us one step closer to the actual implementation of our system.

Attendance Management System
Component and Deployment Diagram

Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

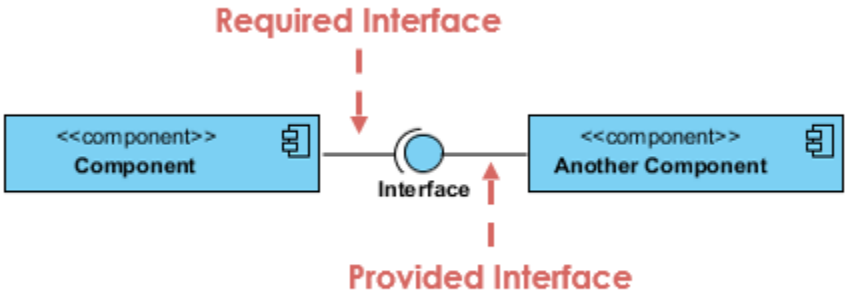
Aim:

To create a component and deployment diagram of the Attendance Management System.

UML Notations for Component Diagram:

1. Component

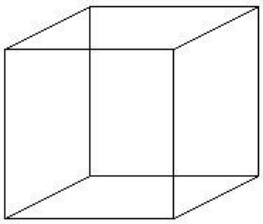
2. Interface



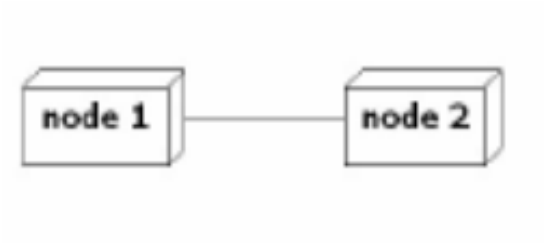
3. Port

UML Notations for Deployment Diagram

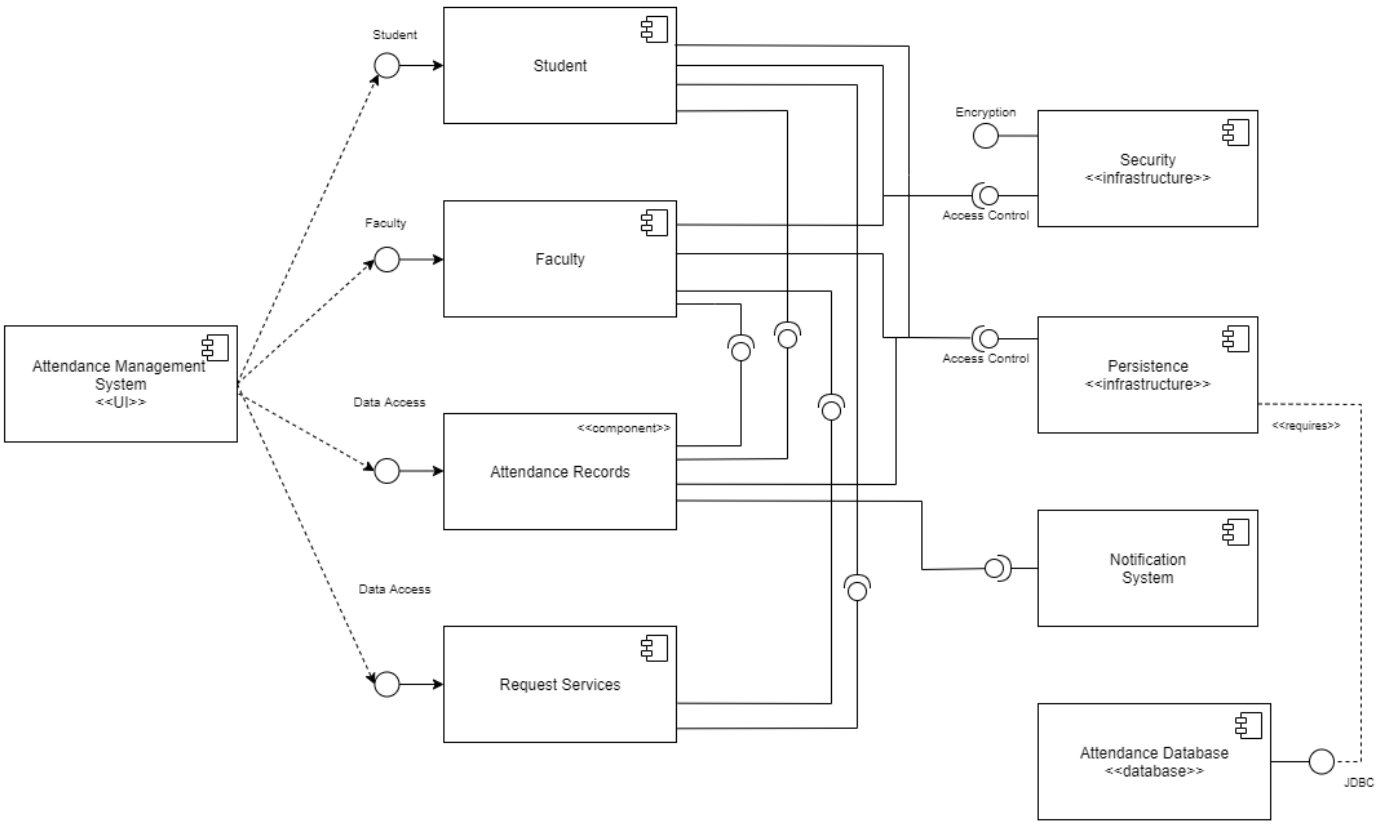
1. Node



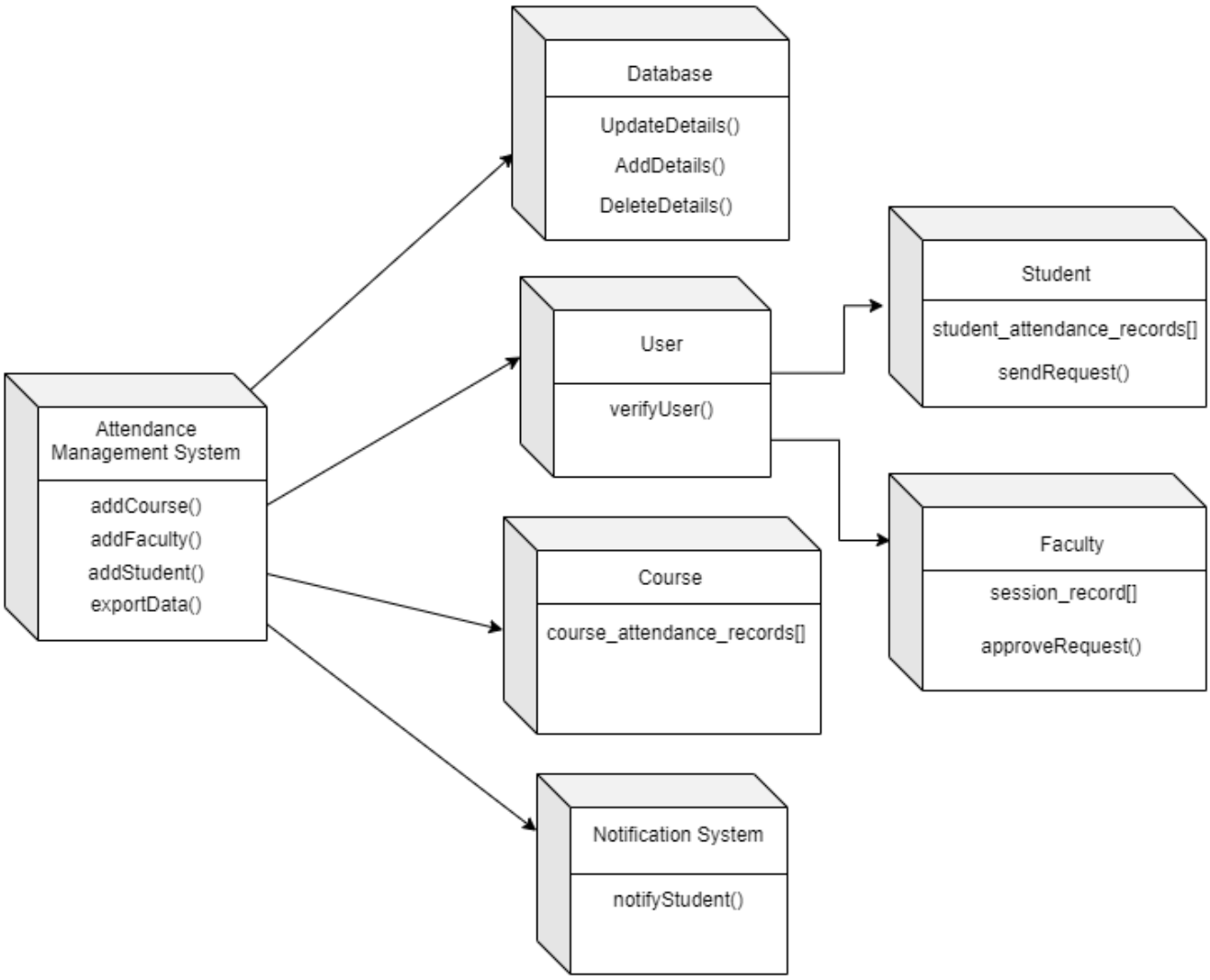
2. Communication Path



Component Diagram



Deployment Diagram



Documentation:

Developing the component and deployment diagram helped us identify the actual implementation details of our project. It helped us understand the various components and how they relate to the functionalities of our system. The deployment diagram helped us identify how the software will be deployed in a hardware system. It also helped us identify and categorize the different layers in our system. This process has brought us one step closer to the actual implementation of our system.

Attendance Management System
Software Implementation

Aarthi.V.S
Adithya Vikram.N
Anusha Chandrasekaran

Aim: To implement the Attendance Management System using HTML and MySQL.

Hardware and Software Requirements:

Software Requirements:

1. Apache Tomcat
2. MySQL Database Service
3. MySQLConnectector/J
4. Java Environment
5. Browser

Hardware Requirements:

1. A server to host the application

Code:

HTML files:

C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminAddCourse.html<!DOCTYPE html>

```
<html>
<head>
	<title>Attendance Managament System</title>
<style>
	body{
		font-family:'Garamond';
		font-size:130%;
		color:#D79D30;
		background-color:Black;
		padding-left: 8px;
	}
	.topnav{
		width: 100%;
		overflow: hidden;
		background-color:#D79D30;
	}
	.topnav a{
```

```
        float: right;
        color: Black;
        text-align: center;
        padding: 14px 16px;
        text-decoration: none;
        font-size: 17px;
        width:12%;
}

.topnav a:hover{
        color:#fbfcd4;
}
.topnav a.active {
        color: #fbfcd4;
}
input[type=text], input[type=password], input[type=email], input[type=number]{
width: 170px;
height: 7px;
padding: 12px;
margin: 7px 4px 22px 0;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
button,input[type=button]{
        border: 2px solid #D79D30;
        color: #fbfcd4;
        background-color: #151515;
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
button:hover,input[type=button]:hover{
        border: 2px solid black;
        color: black;
        background-color: #D79D30;
}

form{
        background-color: #151515;
        align-self: center;
        box-sizing: border-box;
        padding: 50px;
```

```
            padding-left: 140px;
            margin-left: auto;
            margin-right: auto;
            width: 590px;
            height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;

}

.dropdown:hover .dropbtn {
  color: #fbfcd4;
}

.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
```

```
        }

        .dropdown-content a {
          float: none;
          color: #D79D30;
          background-color:#151515;
          padding: 12px 16px;
          text-decoration: none;
          display: block;
          text-align: left;
          width:78%;
        }

        .dropdown-content a:hover {
          background-color: black;
          color:#D79D30;
        }

        .dropdown:hover .dropdown-content {
          display: block;
        }

</style>
</head>
<body>

<div class="container">
   <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
        <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
        <div class="topnav">
                <a href="AMSLogout">Logout</a>
                <div class="dropdown">
                        <button class="dropbtn">Student
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddStudent.html">Add Student</a>
                                <a href="adminDeleteStudent.html">Delete Student</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="dropbtn">Course
                                <i class="fa fa-caret-down"></i>
```

```
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddCourse.html">Add Course</a>
                                <a href="adminDeleteCourse.html">Delete Course</a>
                                <a href="adminAssignCourse.html">Assign Course</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="active dropbtn">Faculty
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddFaculty.html">Add Faculty</a>
                                <a href="adminDeleteFaculty.html">Delete Faculty</a>
                        </div>
                </div>
                <a href="adminExport.html">Export</a>
        </div>
</div>

<form name="admin_addstudent_form" action="AdminAddCourseServlet" method="POST">
        <h3>Add Course</h3>
        <label for="acid" >Course ID:</label>
        <input type="text" name="course_id" id="acid" style="margin-left: 105px;"><br>
        <label for="c_name" >Course Description:</label>
        <input type="text" name="description" id="c_name" style="margin-left: 33px;"><br>
        <label for="dept">Department:</label>
        <input type="text" name="department" id="dept" style="margin-left: 92px;"><br>
        <label for="credits">Credits:</label>
        <input type="number" name="credits" id="credits" style="margin-left: 133px;"><br>
        <button type="submit" name="submit" value="Submit" style="margin-left:
25%;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminAddFaculty.html<!DOCTYPE html>
<html>
<head>
        <title>Attendance Managament System</title>
<style>
        body{
                font-family:'Garamond';
```

```
        font-size:130%;
        color:#D79D30;
        background-color:Black;
        padding-left: 8px;
}
.topnav{
        width: 100%;
        overflow: hidden;
        background-color:#D79D30;
}
.topnav a{
        float: right;
        color: Black;
        text-align: center;
        padding: 14px 16px;
        text-decoration: none;
        font-size: 17px;
        width:12%;
}

.topnav a:hover{
        color:#fbfcd4;
}
.topnav a.active {
        color: #fbfcd4;
}
input[type=text], input[type=password], input[type=email], input[type=number]{
width: 170px;
height: 7px;
padding: 12px;
margin: 7px 4px 22px 0;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
button,input[type=button]{
        border: 2px solid #D79D30;
        color: #fbfcd4;
        background-color: #151515;
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
```

```
button:hover,input[type=button]:hover{
	border: 2px solid black;
	color: black;
	background-color: #D79D30;
}

form{
	background-color: #151515;
	align-self: center;
	box-sizing: border-box;
	padding: 50px;
	padding-left: 140px;
	margin-left: auto;
	margin-right: auto;
	width: 590px;
	height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;

}
```

```
.dropdown:hover .dropbtn {
  color: #fbfcd4;
}

.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}

.dropdown-content a {
  float: none;
  color: #D79D30;
  background-color:#151515;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
  width:78%;
}

.dropdown-content a:hover {
  background-color: black;
  color:#D79D30;
}

.dropdown:hover .dropdown-content {
  display: block;
}

</style>
</head>
<body>

<div class="container">
        <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
        <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
        <div class="topnav">
                <a href="AMSLogout">Logout</a>
                <div class="dropdown">
```

```
                        <button class="dropbtn">Student
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddStudent.html">Add Student</a>
                                <a href="adminDeleteStudent.html">Delete Student</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="dropbtn">Course
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddCourse.html">Add Course</a>
                                <a href="adminDeleteCourse.html">Delete Course</a>
                                <a href="adminAssignCourse.html">Assign Course</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="active dropbtn">Faculty
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddFaculty.html">Add Faculty</a>
                                <a href="adminDeleteFaculty.html">Delete Faculty</a>
                        </div>
                </div>
                <a href="adminExport.html">Export</a>
        </div>
</div>


<form name="admin_addstudent_form" action="AdminAddFacultyServlet" method="POST">
        <h3>Add Faculty</h3>
        <label for="faculty_id" >Faculty ID:</label>
        <input type="text" name="faculty_id" id="faculty_id" style="margin-left: 58px;"><br>
        <label for="f_name">Name:</label>
        <input type="text" name="f_name" id="f_name" style="margin-left: 96px;"><br>
        <label for="dept">Department:</label>
        <input type="text" name="dept" id="dept" style="margin-left: 47px;"><br>
        <label for="email">E-mail:</label>
        <input type="email" name="email" id="email" style="margin-left: 90px;"><br>
        <label for="desig">Designation:</label>
        <input type="text" name="desig" id="desig" style="margin-left: 47px;"><br>
```

```
	<button type="submit" name="submit" value="Submit" style="margin-left:
25%;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminAddStudent.html<!DOCTYPE html>
<html>
<head>
	<title>Attendance Managament System</title>
<style>
	body{
		font-family:'Garamond';
		font-size:130%;
		color:#D79D30;
		background-color:Black;
		padding-left: 8px;
	}
	.topnav{
		width: 100%;
		overflow: hidden;
		background-color:#D79D30;
	}
	.topnav a{
		float: right;
		color: Black;
		text-align: center;
		padding: 14px 16px;
		text-decoration: none;
		font-size: 17px;
		width:12%;
	}

	.topnav a:hover{
		color:#fbfcd4;
	}
	.topnav a.active {
		color: #fbfcd4;
	}
	input[type=text], input[type=password], input[type=email], input[type=number]{
	width: 170px;
	height: 7px;
	padding: 12px;
```

```
margin: 7px 4px 22px 0;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
button,input[type=button]{
        border: 2px solid #D79D30;
        color: #fbfcd4;
        background-color: #151515;
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
button:hover,input[type=button]:hover{
        border: 2px solid black;
        color: black;
        background-color: #D79D30;
}

form{
        background-color: #151515;
        align-self: center;
        box-sizing: border-box;
        padding: 50px;
        padding-left: 140px;
        margin-left: auto;
        margin-right: auto;
        width: 590px;
        height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
```

```
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;

}

.dropdown:hover .dropbtn {
  color: #fbfcd4;
}

.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}

.dropdown-content a {
  float: none;
  color: #D79D30;
  background-color:#151515;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
  width:78%;
}

.dropdown-content a:hover {
  background-color: black;
  color:#D79D30;
}

.dropdown:hover .dropdown-content {
```

```
            display: block;
        }

</style>
</head>
<body>

<div class="container">
        <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
        <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
        <div class="topnav">
                <a href="AMSLogout">Logout</a>
                <div class="dropdown">
                        <button class="dropbtn">Student
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddStudent.html">Add Student</a>
                                <a href="adminDeleteStudent.html">Delete Student</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="dropbtn">Course
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddCourse.html">Add Course</a>
                                <a href="adminDeleteCourse.html">Delete Course</a>
                                <a href="adminAssignCourse.html">Assign Course</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="active dropbtn">Faculty
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddFaculty.html">Add Faculty</a>
                                <a href="adminDeleteFaculty.html">Delete Faculty</a>
                        </div>
                </div>
                <a href="adminExport.html">Export</a>
        </div>
</div>
```

```
<form name="admin_addstudent_form" action="AdminAddStudentServlet" method="POST">
	<h3>Add Student</h3>
	<label for="studentid" >Student ID:</label>
	<input type="text" name="student_id" id="studentid" style="margin-left: 58px;"><br>
	<label for="s_name">Name:</label>
	<input type="text" name="s_name" id="s_name" style="margin-left: 100px;"><br>
	<label for="dept">Department:</label>
	<input type="text" name="dept" id="dept" style="margin-left: 50px;"><br>
	<label for="email">E-mail:</label>
	<input type="email" name="email" id="email" style="margin-left: 95px;"><br>
	<label for="joining_year">Year of Joining:</label>
	<input type="number" name="joining_year" id="joining_year" value= "2020" style="margin-left:
25px;"><br>
	<button type="submit" name="submit" value="Submit" style="margin-left:
25%;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminAssignCourse.html<!DOCTYPE html>
<html>
<head>
	<title>Attendance Mangament System</title>
<style>
	body{
		font-family:'Garamond';
		font-size:130%;
		color:#D79D30;
		background-color:Black;
		padding-left: 8px;
	}
	.topnav{
		width: 100%;
		overflow: hidden;
		background-color:#D79D30;
	}
	.topnav a{
		float: right;
		color: Black;
		text-align: center;
```

```
                padding: 14px 16px;
                text-decoration: none;
                font-size: 17px;
                width:12%;
        }

        .topnav a:hover{
                color:#fbfcd4;
        }
        .topnav a.active {
                color: #fbfcd4;
        }
        input[type=text], input[type=password], input[type=email], input[type=number],input{
        width: 170px;
        height: 7px;
        padding: 12px;
        margin-right: 15px;
margin-left: 0px;
margin-top:22px;
margin-bottom: 4px;
        border: none;
        background: #fbfcd4;
        box-sizing: border-box;

        }
        button,input[type=button]{
                border: 2px solid #D79D30;
                color: #fbfcd4;
                background-color: #151515;
                border-radius: 4px;
                width: 120px;
                height: 25px;
        }
        button:hover,input[type=button]:hover{
                border: 2px solid black;
                color: black;
                background-color: #D79D30;
        }

        form{
                background-color: #151515;
                align-self: center;
                box-sizing: border-box;
                padding: 50px;
```

```
            padding-left: 140px;
            margin-left: auto;
            margin-right: auto;
            width: 590px;
            height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;

}

.dropdown:hover .dropbtn {
  color: #fbfcd4;
}

.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
```

```
        }

        .dropdown-content a {
          float: none;
          color: #D79D30;
          background-color:#151515;
          padding: 12px 16px;
          text-decoration: none;
          display: block;
          text-align: left;
          width:78%;
        }

        .dropdown-content a:hover {
          background-color: black;
          color:#D79D30;
        }

        .dropdown:hover .dropdown-content {
          display: block;
        }

</style>


<script>
        var request;
        function sendInfo(value)
        {
                if(window.XMLHttpRequest){
                request=new XMLHttpRequest();
                }
                else if(window.ActiveXObject){
                request=new ActiveXObject("Microsoft.XMLHTTP");
                }

                try
                {
                        request.onreadystatechange=getInfo;
                        request.open("GET", "GetCourseIDS", true);
                        request.setRequestHeader("Content-type",
"application/x-www-form-urlencoded");
                        request.send();
                }
```

```
		catch(e)
		{
		alert("Unable to connect to server");
		}
	}

	function getInfo(){
	if(request.readyState==4 && request.status==200){
		var val=request.responseText;
		document.getElementById('course_ids').innerHTML=val;
	}
	}

	</script>

</head>
<body onload="sendInfo()">

<div class="container">
   <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
	<h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
	<div class="topnav">
		<a href="AMSLogout">Logout</a>
		<div class="dropdown">
			<button class="dropbtn">Student
				<i class="fa fa-caret-down"></i>
			</button>
			<div class="dropdown-content">
				<a href="adminAddStudent.html">Add Student</a>
				<a href="adminDeleteStudent.html">Delete Student</a>
			</div>
		</div>
		<div class="dropdown">
			<button class="dropbtn">Course
				<i class="fa fa-caret-down"></i>
			</button>
			<div class="dropdown-content">
				<a href="adminAddCourse.html">Add Course</a>
				<a href="adminDeleteCourse.html">Delete Course</a>
				<a href="adminAssignCourse.html">Assign Course</a>
			</div>
		</div>
		<div class="dropdown">
```

```
                        <button class="active dropbtn">Faculty
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddFaculty.html">Add Faculty</a>
                                <a href="adminDeleteFaculty.html">Delete Faculty</a>
                        </div>
                </div>
                <a href="adminExport.html">Export</a>
        </div>
</div>


<form name="admin_addstudent_form" action="AdminAssignCourseServlet" method="POST">
        <h3>Assign Course</h3>
        <label for="acid">Active Course ID:</label>
        <input type="text" name="acid" id="acid" style="margin-left: 50px;"><br>
        <label for="course_id">Course ID:</label>
        <input list="course_ids" id="courseid" style="margin-left: 105px;" name="course_id"/>
        <datalist id="course_ids">
        </datalist>
        <label for="facultyid" >Faculty ID:</label>
        <input type="text" name="facultyid" id="facultyid" style="margin-left:
105px;margin-bottom:12px;"><br>
        <label for="studentid">Student ID (Enter ID's separated by ','):</label>
        <input type="text" name="student_id" id="studentid" style="width:
100%;margin-top:20px;"><br>
        <button type="submit" name="submit" value="Submit" style="margin-left: 25%;margin-top:
10px;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminDeleteCourse.html<!DOCTYPE html>
<html>
<head>
        <title>Attendance Managament System</title>
<style>
        body{
                font-family:'Garamond';
                font-size:130%;
                color:#D79D30;
                background-color:Black;
```

```
        padding-left: 8px;
}
.topnav{
        width: 100%;
        overflow: hidden;
        background-color:#D79D30;
}
.topnav a{
        float: right;
        color: Black;
        text-align: center;
        padding: 14px 16px;
        text-decoration: none;
        font-size: 17px;
        width:12%;
}

.topnav a:hover{
        color:#fbfcd4;
}
.topnav a.active {
        color: #fbfcd4;
}
input[type=text], input[type=password], input[type=email], input[type=number]{
width: 170px;
height: 7px;
padding: 12px;
margin: 7px 4px 22px 0;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
button,input[type=button]{
        border: 2px solid #D79D30;
        color: #fbfcd4;
        background-color: #151515;
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
button:hover,input[type=button]:hover{
        border: 2px solid black;
        color: black;
```

```
        background-color: #D79D30;
}

form{
        background-color: #151515;
        align-self: center;
        box-sizing: border-box;
        padding: 50px;
        padding-left: 140px;
        margin-left: auto;
        margin-right: auto;
        width: 590px;
        height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;

}

.dropdown:hover .dropbtn {
  color: #fbfcd4;
}
```

```
.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}

.dropdown-content a {
  float: none;
  color: #D79D30;
  background-color:#151515;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
  width:78%;
}

.dropdown-content a:hover {
  background-color: black;
  color:#D79D30;
}

.dropdown:hover .dropdown-content {
  display: block;
}

</style>
</head>
<body>

<div class="container">
   <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
        <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
        <div class="topnav">
                <a href="AMSLogout">Logout</a>
                <div class="dropdown">
                        <button class="dropbtn">Student
                                <i class="fa fa-caret-down"></i>
                        </button>
```

```
                        <div class="dropdown-content">
                                <a href="adminAddStudent.html">Add Student</a>
                                <a href="adminDeleteStudent.html">Delete Student</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="dropbtn">Course
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddCourse.html">Add Course</a>
                                <a href="adminDeleteCourse.html">Delete Course</a>
                                <a href="adminAssignCourse.html">Assign Course</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="active dropbtn">Faculty
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddFaculty.html">Add Faculty</a>
                                <a href="adminDeleteFaculty.html">Delete Faculty</a>
                        </div>
                </div>
                <a href="adminExport.html">Export</a>
        </div>
</div>



<form name="admin_addstudent_form" action="AdminDeleteCourseServlet" method="POST">
        <h3>Delete Course</h3>
        <label for="acid" >Active Course ID:</label>
        <input type="text" name="acid" id="acid" style="margin-left: 58px;"><br>
        <button type="submit" name="submit" value="Submit" style="margin-left:
25%;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminDeleteFaculty.html<!DOCTYPE html>
<html>
<head>
```

```
	<title>Attendance Managament System</title>
<style>
	body{
		font-family:'Garamond';
		font-size:130%;
		color:#D79D30;
		background-color:Black;
		padding-left: 8px;
	}
	.topnav{
		width: 100%;
		overflow: hidden;
		background-color:#D79D30;
	}
	.topnav a{
		float: right;
		color: Black;
		text-align: center;
		padding: 14px 16px;
		text-decoration: none;
		font-size: 17px;
		width:12%;
	}

	.topnav a:hover{
		color:#fbfcd4;
	}
	.topnav a.active {
		color: #fbfcd4;
	}
	input[type=text], input[type=password], input[type=email], input[type=number]{
	width: 170px;
	height: 7px;
	padding: 12px;
	margin: 7px 4px 22px 0;
	border: none;
	background: #fbfcd4;
	box-sizing: border-box;

	}
	button,input[type=button]{
		border: 2px solid #D79D30;
		color: #fbfcd4;
		background-color: #151515;
```

```
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
button:hover,input[type=button]:hover{
        border: 2px solid black;
        color: black;
        background-color: #D79D30;
}

form{
        background-color: #151515;
        align-self: center;
        box-sizing: border-box;
        padding: 50px;
        padding-left: 140px;
        margin-left: auto;
        margin-right: auto;
        width: 590px;
        height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
```

```
  margin: 0;

 }

 .dropdown:hover .dropbtn {
  color: #fbfcd4;
 }

 .dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
 }

 .dropdown-content a {
  float: none;
  color: #D79D30;
  background-color:#151515;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
  width:78%;
 }

 .dropdown-content a:hover {
  background-color: black;
  color:#D79D30;
 }

 .dropdown:hover .dropdown-content {
  display: block;
 }

</style>
</head>
<body>

<div class="container">
        <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
```

```
	<h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
	<div class="topnav">
		<a href="AMSLogout">Logout</a>
		<div class="dropdown">
			<button class="dropbtn">Student
				<i class="fa fa-caret-down"></i>
			</button>
			<div class="dropdown-content">
				<a href="adminAddStudent.html">Add Student</a>
				<a href="adminDeleteStudent.html">Delete Student</a>
			</div>
		</div>
		<div class="dropdown">
			<button class="dropbtn">Course
				<i class="fa fa-caret-down"></i>
			</button>
			<div class="dropdown-content">
				<a href="adminAddCourse.html">Add Course</a>
				<a href="adminDeleteCourse.html">Delete Course</a>
				<a href="adminAssignCourse.html">Assign Course</a>
			</div>
		</div>
		<div class="dropdown">
			<button class="active dropbtn">Faculty
				<i class="fa fa-caret-down"></i>
			</button>
			<div class="dropdown-content">
				<a href="adminAddFaculty.html">Add Faculty</a>
				<a href="adminDeleteFaculty.html">Delete Faculty</a>
			</div>
		</div>
		<a href="adminExport.html">Export</a>
	</div>
</div>




<form name="admin_addstudent_form" action="AdminDeleteFacultyServlet" method="POST">
	<h3>Delete Faculty</h3>
	<label for="faculty_id" >Faculty ID:</label>
	<input type="text" name="faculty_id" id="faculty_id" style="margin-left: 58px;"><br>
```

```
	<button type="submit" name="submit" value="Submit" style="margin-left:
25%;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminDeleteStudent.html<!DOCTYPE html>
<html>
<head>
	<title>Attendance Managament System</title>
<style>
	body{
		font-family:'Garamond';
		font-size:130%;
		color:#D79D30;
		background-color:Black;
		padding-left: 8px;
	}
	.topnav{
		width: 100%;
		overflow: hidden;
		background-color:#D79D30;
	}
	.topnav a{
		float: right;
		color: Black;
		text-align: center;
		padding: 14px 16px;
		text-decoration: none;
		font-size: 17px;
		width:12%;
	}

	.topnav a:hover{
		color:#fbfcd4;
	}
	.topnav a.active {
		color: #fbfcd4;
	}
	input[type=text], input[type=password], input[type=email], input[type=number]{
	width: 170px;
	height: 7px;
	padding: 12px;
```

```
margin: 7px 4px 22px 0;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
button,input[type=button]{
        border: 2px solid #D79D30;
        color: #fbfcd4;
        background-color: #151515;
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
button:hover,input[type=button]:hover{
        border: 2px solid black;
        color: black;
        background-color: #D79D30;
}

form{
        background-color: #151515;
        align-self: center;
        box-sizing: border-box;
        padding: 50px;
        padding-left: 140px;
        margin-left: auto;
        margin-right: auto;
        width: 590px;
        height: 490px;
}

aside {
  width: 20%;
  padding-left: 15px;
  float: left;
  font-style: italic;
  background-color: #151515;
  color: #D79D30;
}

.dropdown {
  float: right;
  overflow: hidden;
```

```
}

.dropdown .dropbtn {
  font-size: 16px;
  border: none;
  outline: none;
  color: black;
  padding: 14px 16px;
  background-color: inherit;
  font-family: inherit;
  margin: 0;

}

.dropdown:hover .dropbtn {
  color: #fbfcd4;
}

.dropdown-content {
  display: none;
  position: absolute;
  background-color: #f9f9f9;

  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}

.dropdown-content a {
  float: none;
  color: #D79D30;
  background-color:#151515;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
  width:78%;
}

.dropdown-content a:hover {
  background-color: black;
  color:#D79D30;
}

.dropdown:hover .dropdown-content {
```

```
            display: block;
        }

</style>
</head>
<body>

<div class="container">
        <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
        <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
        <div class="topnav">
                <a href="AMSLogout">Logout</a>
                <div class="dropdown">
                        <button class="dropbtn">Student
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddStudent.html">Add Student</a>
                                <a href="adminDeleteStudent.html">Delete Student</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="dropbtn">Course
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddCourse.html">Add Course</a>
                                <a href="adminDeleteCourse.html">Delete Course</a>
                                <a href="adminAssignCourse.html">Assign Course</a>
                        </div>
                </div>
                <div class="dropdown">
                        <button class="active dropbtn">Faculty
                                <i class="fa fa-caret-down"></i>
                        </button>
                        <div class="dropdown-content">
                                <a href="adminAddFaculty.html">Add Faculty</a>
                                <a href="adminDeleteFaculty.html">Delete Faculty</a>
                        </div>
                </div>
                <a href="adminExport.html">Export</a>
        </div>
</div>
```

```
<form name="admin_addstudent_form" action="AdminDeleteStudentServlet" method="POST">
        <h3>Delete Student</h3>
        <label for="student_id" >Student ID:</label>
        <input type="text" name="student_id" id="student_id" style="margin-left: 58px;"><br>
        <button type="submit" name="submit" value="Submit" style="margin-left:
25%;">Submit</button>
</form>


</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\adminExport.html<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
    body{
      font-family:'Garamond';
      font-size:130%;
      color:#D79D30;
      background-color:Black;
      padding-left: 8px;
    }
    .topnav{
      width: 100%;
      overflow: hidden;
      background-color:#D79D30;
    }
    .topnav a{
      float: right;
       color: Black;
       text-align: center;
       padding: 14px 16px;
       text-decoration: none;
       font-size: 17px;
       width:12%;
    }
```

```
.topnav a:hover{
   color:#fbfcd4;
}
.topnav a.active {
     color: #fbfcd4;
  }
input[type=text], input[type=password], input[type=email], input[type=number]{
width: 170px;
height: 7px;
padding: 12px;
margin-right: 15px;
margin-left: 0px;
margin-top:22px;
margin-bottom: 4px;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
input[type=button]{
   border: 2px solid #D79D30;
   color: #fbfcd4;
   background-color: #151515;
   border-radius: 4px;
   width: 120px;
   height: 25px;
}
input[type=button]:hover{
   border: 2px solid black;
   color: black;
   background-color: #D79D30;
}

  form{
     background-color: #151515;
   align-self: center;
   box-sizing: border-box;
   padding: 50px;
   padding-left: 140px;
   margin-left: auto;
   margin-right: auto;
   width: 590px;
   height: 490px;
  }
```

```
 aside {
 width: 20%;
 padding-left: 15px;
 float: left;
 font-style: italic;
 background-color: #151515;
 color: #D79D30;
}

.dropdown {
 float: right;
 overflow: hidden;
}

.dropdown .dropbtn {
 font-size: 16px;
 border: none;
 outline: none;
 color: black;
 padding: 14px 16px;
 background-color: inherit;
 font-family: inherit;
 margin: 0;

}

.dropdown:hover .dropbtn {
 color: #fbfcd4;
}

.dropdown-content {
 display: none;
 position: absolute;
 background-color: #f9f9f9;

 box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
 z-index: 1;
}

.dropdown-content a {
 float: none;
 color: #D79D30;
 background-color:#151515;
```

```
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  text-align: left;
  width:78%;
 }

 .dropdown-content a:hover {
  background-color: black;
  color:#D79D30;
 }

 .dropdown:hover .dropdown-content {
  display: block;
 }

 #export{
    display: none;
 }




</style>
<script>

 var request;
 var request1;
 function sendInfo(value)
 {
    if(window.XMLHttpRequest){
    request=new XMLHttpRequest();
    }
    else if(window.ActiveXObject){
    request=new ActiveXObject("Microsoft.XMLHTTP");
    }

    try
    {
       request.onreadystatechange=getInfo;
       request.open("GET", "GetActiveCourseIDS", true);
       request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
       request.send();
    }
```

```
    catch(e)
    {
    alert("Unable to connect to server");
    }
}

function getInfo(){
if(request.readyState==4 && request.status==200){
    var val=request.responseText;
    document.getElementById('acids').innerHTML=val;
}
}

function getdetails(){
    var id=document.getElementById("acid").value;
    var url="acid="+id;
    if(window.XMLHttpRequest){
    request1=new XMLHttpRequest();
    }
    else if(window.ActiveXObject){
    request1=new ActiveXObject("Microsoft.XMLHTTP");
    }

    try
    {
        request1.onreadystatechange=exportdetails;
        request1.open("POST", "./AdminExportServlet", true);
        request1.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
        request1.send(url);
    }
    catch(e)
    {
    alert("Unable to connect to server");
    }
}

function exportdetails(){
    if(request1.readyState==4 && request1.status==200){
        var val=request1.responseText;
        document.getElementById("export").innerHTML=val;
        exportTableToCSV(document.getElementById("acid").value+"_Attendance.csv");
        document.getElementById("acid").value="";
    }
}
```

```
function downloadCSV(csv, filename) {
    var csvFile;
    var downloadLink;

    // CSV file
    csvFile = new Blob([csv], {type: "text/csv"});

    // Download link
    downloadLink = document.createElement("a");

    // File name
    downloadLink.download = filename;

    // Create a link to the file
    downloadLink.href = window.URL.createObjectURL(csvFile);

    // Hide download link
    downloadLink.style.display = "none";

    // Add the link to DOM
    document.body.appendChild(downloadLink);

    // Click download link
    downloadLink.click();
}


function exportTableToCSV(filename) {
    var csv = [];
    var rows = document.querySelectorAll("table tr");

    for (var i = 0; i < rows.length; i++) {
        var row = [], cols = rows[i].querySelectorAll("td, th");

        for (var j = 0; j < cols.length; j++)
            row.push(cols[j].innerText);

        csv.push(row.join(","));
    }

    // Download CSV file
    downloadCSV(csv.join("\n"), filename);
```

```
    }


    </script>

</head>



<body onload="sendInfo()">

    <div class="container">
        <img class="logo" src="logo.jpg" alt="logo" style="width:6%; height:6%; float:left;">
        <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
        <div class="topnav">
            <a href="AMSLogout">Logout</a>
            <div class="dropdown">
                <button class="dropbtn">Student
                    <i class="fa fa-caret-down"></i>
                </button>
                <div class="dropdown-content">
                    <a href="adminAddStudent.html">Add Student</a>
                    <a href="adminDeleteStudent.html">Delete Student</a>
                </div>
            </div>
            <div class="dropdown">
                <button class="dropbtn">Course
                    <i class="fa fa-caret-down"></i>
                </button>
                <div class="dropdown-content">
                    <a href="adminAddCourse.html">Add Course</a>
                    <a href="adminDeleteCourse.html">Delete Course</a>
                    <a href="adminAssignCourse.html">Assign Course</a>
                </div>
            </div>
            <div class="dropdown">
                <button class="active dropbtn">Faculty
                    <i class="fa fa-caret-down"></i>
                </button>
                <div class="dropdown-content">
                    <a href="adminAddFaculty.html">Add Faculty</a>
                    <a href="adminDeleteFaculty.html">Delete Faculty</a>
                </div>
```

```
        </div>
        <a href="adminExport.html">Export</a>
      </div>
    </div>
    <form>
      <h3>Export Data</h3>
      <label for="acid">Active Course ID:</label>
      <input list="acids" id="acid" required style="margin-right:10px" name="acid"/>
      <datalist id="acids">
      </datalist>
      <input type="button" name="submit" value="Export" style="margin-left: 25%;margin-top: 10px;"
onclick="getdetails()">
    </form>
    <div id="export"><p></p></div>

</body>
</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\facultydiscrepancy.html<!DOCTYPE
html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>

    <style>
                body{
                        font-family:'Garamond';
                        font-size:130%;
                        color:#D79D30;
                        background-color:Black;
                        padding-left: 8px;
                }
                .topnav{
                        width: 100%;
                        overflow: hidden;
                        background-color:#D79D30;
                }
                .topnav a{
                        float: right;
                          color: Black;
                          text-align: center;
                          padding: 14px 16px;
                          text-decoration: none;
```

```
                font-size: 17px;
                width:12%;
        }

        .topnav a:hover{
                color:#fbfcd4;
        }
        .topnav a.active {
                color: #fbfcd4;
         }
        input[type=text], input[type=password]{
border: none;
display: inline;
font-family: inherit;
font-size: inherit;
padding: none;
width: auto;
background-color: #151515;
color:#D79D30;

        }
        button,input[type=button]{
                border: 2px solid #D79D30;
                color: #fbfcd4;
                background-color: #151515;
                border-radius: 4px;
                width: 120px;
                height: 25px;
        }
        button:hover,input[type=button]:hover{
                border: 2px solid black;
                color: black;
                background-color: #D79D30;
        }

         form{
                 background-color: #151515;
                align-self: center;
                box-sizing: border-box;
                padding: 70px;
                padding-left: 140px;
                margin-left: auto;
                margin-right: auto;
                width: 560px;
```

```
                    height: 490px;
                }

                aside {
                width: 20%;
                padding-left: 15px;
                float: left;
                font-style: italic;
                background-color: #151515;
                color: #D79D30;
                }

    table {
        margin-left: auto;
        margin-right: auto;
        width: 50%;
        padding: 10px;
    }

    th,td{
        height: 35px;
        vertical-align: middle;
        padding: 15px;
        text-align: left;
        background-color: #151515;
    }
        </style>

<script>
    var request;
    function sendInfo(value)
    {
        if(window.XMLHttpRequest){
        request=new XMLHttpRequest();
        }
        else if(window.ActiveXObject){
        request=new ActiveXObject("Microsoft.XMLHTTP");
        }

        try
        {
            request.onreadystatechange=getInfo;
            request.open("GET", "./FacultyDiscrepancy", true);
            request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
```

```
        request.send();
    }
    catch(e)
    {
    alert("Unable to connect to server");
    }
}

function getInfo(){
if(request.readyState==4 && request.status==200){
    var val=request.responseText;
    document.getElementById('disclist').innerHTML=val;
}
}

function discprocess(stat,id){
    var res = id.split(";");
    var stuid=res[0];
    var acid=res[1];
    var sessid=res[2];
    var url="stu_id="+stuid+"&acid="+acid+"&sess_id="+sessid+"&status="+stat;
    if(window.XMLHttpRequest){
    request=new XMLHttpRequest();
    }
    else if(window.ActiveXObject){
    request=new ActiveXObject("Microsoft.XMLHTTP");
    }

    try
    {
        request.onreadystatechange=function() {
            if(request.readyState==4 && request.status==200){
                var val=request.responseText;
                var btid=id+"approved";
                var bt1id=id+"denied";
                var tble=id+"table";
                var form = document.getElementById(btid);
                var form1 = document.getElementById(bt1id);
                form.remove();
                form1.remove();
                var row=document.getElementById(tble);
                var tdata=row.cells[4];
                tdata.innerHTML=val;
            }
```

```
            };
            request.open("POST", "./FacultyProcessDiscrepancy", true);
            request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
            request.send(url);
          }
          catch(e)
          {
          alert("Unable to connect to server");
          }
        }


    </script>



</head>


<body onload="sendInfo()">

    <div class="container">
                    <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
              <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
                    <div class="topnav">
                            <a href="AMSLogout">Logout</a>
                            <a class="active" href="./facultydiscrepancy.html">Discrepancy</a>
                            <a href="./facultyod.html">OD</a>
                            <a href="./FacultyHomePage.html">Home</a>
                    </div>
           </div>

    <div id="disclist"></div>

</body>
</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\FacultyHomePage.html<!DOCTYPE
html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
```

```
<style>
   body{
      font-family:'Garamond';
      font-size:110%;
      color:#D79D30;
      background-color:Black;
      padding-left: 8px;
   }
   .topnav{
      width: 100%;
      overflow: hidden;
      background-color:#D79D30;
   }
   .topnav a{
      float: right;
        color: Black;
        text-align: center;
        padding: 14px 16px;
        text-decoration: none;
        font-size: 17px;
        width:12%;
   }

   .topnav a:hover{
      color:#fbfcd4;
   }
   .topnav a.active {
       color: #fbfcd4;
    }

    table {
       width: 50%;
       padding: 10px;
       margin: 0 auto;
    }

    th,td{
       height: 35px;
       vertical-align: middle;
       padding: 15px;
       text-align: left;
       background-color: #151515;
```

```
 }

 a{
    text-decoration: none;
    color:#D79D30;

 }
 .courseref{ background-color: #151515;font-style: italic;}

 .courseref:hover{
    background-color: #0A0A0A;
 }

 .section {
    margin-left: 450px;
 }
 li{
    padding: 10px;
 }
 aside {
 position: fixed;
 width: 400px;
 padding-left: 10px;
 float: left;
 font-style: italic;
 background-color: #151515;
 color: #D79D30;
}

h1{
   font-size: 25px;
   padding-left: 26%;
}

.tdbutton{
   width: 100%;
   height: 100%;
   background-color: #151515;
   color:#D79D30;
   border: 1px #D79D30 solid;
   border-radius: 2px;
   text-decoration: none;
   cursor: pointer;
   margin:0;
```

```
        padding:10px;
    }

    .bttd{
        padding-top: 0px;
        padding-bottom: 0px;
    }

</style>

<script>
    var request;
    function sendInfo(value)
    {
        if(window.XMLHttpRequest){
        request=new XMLHttpRequest();
        }
        else if(window.ActiveXObject){
        request=new ActiveXObject("Microsoft.XMLHTTP");
        }

        try
        {
            request.onreadystatechange=getInfo;
            request.open("GET", "./Faculty", true);
            request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
            request.send();
        }
        catch(e)
        {
        alert("Unable to connect to server");
        }
    }

    function getInfo(){
    if(request.readyState==4 && request.status==200){
        var val=request.responseText;
        document.getElementById('classes').innerHTML=val;
    }
    }

</script>

</head>
```

```
<body onload="sendInfo()" id="body">
   <div class="container">
      <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
            <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
      <div class="topnav">
         <a href="./AMSLogout">Logout</a>
         <a href="./facultydiscrepancy.html">Discrepancy</a>
         <a href="./facultyod.html">OD</a>
         <a class="active" href="./FacultyHomePage.html">Home</a>
      </div>
   </div>

   <div id="classes" ></div>
</body>
</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\facultyod.html<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>

   <style>
                  body{
                        font-family:'Garamond';
                        font-size:130%;
                        color:#D79D30;
                        background-color:Black;
                        padding-left: 8px;
                  }
                  .topnav{
                        width: 100%;
                        overflow: hidden;
                        background-color:#D79D30;
                  }
                  .topnav a{
                        float: right;
                          color: Black;
                          text-align: center;
                          padding: 14px 16px;
                          text-decoration: none;
                          font-size: 17px;
```

```
                width:12%;
        }

        .topnav a:hover{
                color:#fbfcd4;
        }
        .topnav a.active {
                color: #fbfcd4;
         }
        input[type=text], input[type=password]{
border: none;
display: inline;
font-family: inherit;
font-size: inherit;
padding: none;
width: auto;
background-color: #151515;
color:#D79D30;

        }
        button,input[type=button]{
                border: 2px solid #D79D30;
                color: #fbfcd4;
                background-color: #151515;
                border-radius: 4px;
                width: 120px;
                height: 25px;
        }
        button:hover,input[type=button]:hover{
                border: 2px solid black;
                color: black;
                background-color: #D79D30;
        }

         form{
                background-color: #151515;
                align-self: center;
                box-sizing: border-box;
                padding: 70px;
                padding-left: 140px;
                margin-left: auto;
                margin-right: auto;
                width: 560px;
                height: 490px;
```

```
                }

                aside {
                width: 20%;
                padding-left: 15px;
                float: left;
                font-style: italic;
                background-color: #151515;
                color: #D79D30;
               }

        table {
      margin-left: auto;
      margin-right: auto;
      width: 50%;
      padding: 10px;
   }

   th,td{
      height: 35px;
      vertical-align: middle;
      padding: 15px;
      text-align: left;
      background-color: #151515;
   }
       </style>

<script>
   var request;
   function sendInfo(value)
   {
      if(window.XMLHttpRequest){
      request=new XMLHttpRequest();
      }
      else if(window.ActiveXObject){
      request=new ActiveXObject("Microsoft.XMLHTTP");
      }

      try
      {
         request.onreadystatechange=getInfo;
         request.open("GET", "./FacultyOD", true);
         request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
         request.send();
```

```
    }
    catch(e)
    {
    alert("Unable to connect to server");
    }
}

function getInfo(){
if(request.readyState==4 && request.status==200){
    var val=request.responseText;
    document.getElementById('odlist').innerHTML=val;
}
}

function odprocess(stat,id){
    var res = id.split(";");
    var stuid=res[0];
    var acid=res[1];
    var sessid=res[2];
    var url="stu_id="+stuid+"&acid="+acid+"&sess_id="+sessid+"&status="+stat;
    if(window.XMLHttpRequest){
    request=new XMLHttpRequest();
    }
    else if(window.ActiveXObject){
    request=new ActiveXObject("Microsoft.XMLHTTP");
    }

    try
    {
       request.onreadystatechange=function() {
          if(request.readyState==4 && request.status==200){
             var val=request.responseText;
             var btid=id+"approved";
             var bt1id=id+"denied";
             var tble=id+"table";
             var form = document.getElementById(btid);
             var form1 = document.getElementById(bt1id);
             form.remove();
             form1.remove();
             var row=document.getElementById(tble);
             var tdata=row.cells[5];
             tdata.innerHTML=stat;
          }
       };
```

```
            request.open("POST", "./FacultyProcessOD", true);
            request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
            request.send(url);
          }
          catch(e)
          {
          alert("Unable to connect to server");
          }
        }


    </script>



</head>


<body onload="sendInfo()">

    <div class="container">
                    <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
              <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
                    <div class="topnav">
                            <a href="AMSLogout">Logout</a>
                            <a href="./facultydiscrepancy.html">Discrepancy</a>
                            <a class="active" href="./facultyod.html">OD</a>
                            <a href="./FacultyHomePage.html">Home</a>
                    </div>
           </div>

    <div id="odlist"></div>

</body>
</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\index.html<!DOCTYPE html>
<html>
<head>
<title>Login</title>
<style>
            body{
                    font-family:'Garamond';
                    font-size:130%;
                    color:#D79D30;
```

```
                background-color:Black;
                padding-left: 8px;
        }

        input[type=text], input[type=password]{
          width: 170px;
          height: 7px;
          padding: 12px;
          margin: 7px 4px 22px 0;
          border: none;
          background: #fbfcd4;
          box-sizing: border-box;

        }

        form{
                background-color: #151515;
                align-self: center;
                box-sizing: border-box;
                padding: 70px;
                padding-left: 140px;
                margin-left: auto;
                margin-right: auto;
                width: 540px;
                height: 440px;
        }

   button{
                border-radius: 4px;
                width: 120px;
                height: 25px;
      border: 2px solid #D79D30;
      background-color: transparent;
      color:#D79D30;
        }
        button:hover,input[type=button]:hover{
                border: 2px solid black;
                color: black;
                background-color: #D79D30;
        }

</style>
</head>
```

```
<body>
   <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
         <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>

   <div>
   <form class="loginform" name="login" id="login" action="AMSLogin" method="POST">
      <h3>Login</h3>
      <label for="username">Username: </label>
      <input type="text" name="username" id="username" style=" margin-left: 10px; "><br>
      <label for="pwd">Password:</label>
      <input type="Password" name="password" id="pwd" style="margin-left: 15px;"><br>
      <button type="submit" name="submit" value="Submit" >Submit</button>
   </form>
   </div>
   </body>
   </html>


C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\StudentDiscrepancy.html<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>

   <style>
      body{
         font-family:'Garamond';
         font-size:130%;
         color:#D79D30;
         background-color:Black;
         padding-left: 8px;
      }
      .topnav{
         width: 100%;
         overflow: hidden;
         background-color:#D79D30;
      }
      .topnav a{
         float: right;
           color: Black;
           text-align: center;
```

```
        padding: 14px 16px;
        text-decoration: none;
        font-size: 17px;
        width:12%;
    }

    .topnav a:hover{
       color:#fbfcd4;
    }
    .topnav a.active {
         color: #fbfcd4;
     }


     table {
       margin-left: auto;
       margin-right: auto;
       width: 50%;
       padding: 10px;
    }

    th,td{
       height: 35px;
       vertical-align: middle;
       padding: 15px;
       text-align: left;
       background-color: #151515;
    }

</style>

<script>
    var request;
    function sendInfo(value)
    {
       if(window.XMLHttpRequest){
       request=new XMLHttpRequest();
       }
       else if(window.ActiveXObject){
       request=new ActiveXObject("Microsoft.XMLHTTP");
       }

       try
       {
```

```
            request.onreadystatechange=getInfo;
            request.open("GET", "./StudentDiscrepancyViewServlet", true);
            request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
            request.send();
          }
          catch(e)
          {
          alert("Unable to connect to server");
          }
        }

        function getInfo(){
        if(request.readyState==4 && request.status==200){
          var val=request.responseText;
          document.getElementById('disclist').innerHTML=val;
        }
        }



    </script>



</head>


<body onload="sendInfo()">

    <div class="container">
                    <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
              <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
                    <div class="topnav">
                            <a href="AMSLogout">Logout</a>
                            <a class="active" href="./StudentDiscrepancyHome.html">Discrepancy</a>
                            <a  href="./StudentODHome.html">OD</a>
                            <a href="./StudentHomePage.html">Home</a>
                    </div>
           </div>
    <h4>Your Discrepancy Reports</h4>
    <div id="disclist"></div>

</body>
```

```
</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\StudentDiscrepancyHome.html<!DOCT
YPE html>
<html>
<head>
	<title>Attendance Mangament System</title>
	<style>
		body{
			font-family:'Garamond';
			font-size:130%;
			color:#D79D30;
			background-color:Black;
			padding-left: 8px;
		}
		.topnav{
			width: 100%;
			overflow: hidden;
			background-color:#D79D30;
		}
		.topnav a{
			float: right;
			  color: Black;
			  text-align: center;
			  padding: 14px 16px;
			  text-decoration: none;
			  font-size: 17px;
			  width:12%;
		}

		.topnav a:hover{
			color:#fbfcd4;
		}
		.topnav a.active {
			  color: #fbfcd4;
		  }
		input[type=text], input[type=password]{
		width: 170px;
		height: 7px;
		padding: 12px;
		margin: 7px 4px 22px 0;
		border: none;
		background: #fbfcd4;
		box-sizing: border-box;

		}
```

```
                button,input[type=button]{
                        border: 2px solid #D79D30;
                        color: #fbfcd4;
                        background-color: #151515;
                        border-radius: 4px;
                        width: 120px;
                        height: 25px;
                }
                button:hover,input[type=button]:hover{
                        border: 2px solid black;
                        color: black;
                        background-color: #D79D30;
                }

                 form{
                          background-color: #151515;
                        align-self: center;
                        box-sizing: border-box;
                        padding: 70px;
                        padding-left: 140px;
                        margin-left: auto;
                        margin-right: auto;
                        width: 560px;
                        height: 490px;
                 }

                 aside {
                 width: 20%;
                 padding-left: 15px;
                 float: left;
                 font-style: italic;
                 background-color: #151515;
                 color: #D79D30;
                }
        </style>
</head>
<body>


        <div class="container">
                <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
                <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
                <div class="topnav">
```

```
                        <a href="AMSLogout">Logout</a>
                        <a class="active" href="./StudentDiscrepancyHome.html">Discrepancy</a>
                        <a href="./StudentODHome.html">OD</a>
                        <a href="./StudentHomePage.html">Home</a>
                    </div>
            </div>

            <form name="s_discform" action="./StudentDiscrepancySubmitServlet" method="POST">
                    <h3>Request for Discrepancy</h3>
                    <label for="acid" >Course ID:</label>
                    <input type="text" name="acid" id="acid" style="margin-left: 58px;"><br>
                    <label for="sessionID">Session ID:</label>
                    <input type="text" name="session_id" id="sessionID" style="margin-left: 55px;"><br>
                    <label for="justification">Justification:</label>
                    <input type="text" name="justification" id="justification" style="margin-left:
48px;"><br>
                    <button type="submit" name="submit" value="Submit" style="margin-left:
30px;">Submit</button>
                    <button type="submit" formaction="./StudentDiscrepancy.html" style="margin-left:
10px;">View History</button>
            </form>

</body>
</html>
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\StudentHomePage.html<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>

    <style>
        body{
            font-family:'Garamond';
            font-size:110%;
            color:#D79D30;
            background-color:Black;
            padding-left: 8px;
        }
        .topnav{
            width: 100%;
            overflow: hidden;
            background-color:#D79D30;
```

```
}
.topnav a{
   float: right;
     color: Black;
     text-align: center;
     padding: 14px 16px;
     text-decoration: none;
     font-size: 17px;
     width:12%;
}

.topnav a:hover{
   color:#fbfcd4;
}
.topnav a.active {
     color: #fbfcd4;
 }

 table {
     width: 50%;
     padding: 10px;
 }

 th,td{
     height: 35px;
     vertical-align: middle;
     padding: 15px;
     text-align: left;
     background-color: #151515;
 }

 a{
     text-decoration: none;
     color:#D79D30;

 }
 .courseref{ background-color: #151515;font-style: italic;}

 .courseref:hover{
     background-color: #0A0A0A;
 }

 .section {
     margin-left: 450px;
```

```
    }
    li{
        padding: 10px;
    }
    aside {
    position: fixed;
    width: 400px;
    padding-left: 10px;
    float: left;
    font-style: italic;
    background-color: #151515;
    color: #D79D30;
   }

   h1{
      font-size: 25px;
      padding-left: 1.2%;
   }

</style>



<script>
   var request;
   var request1;
   function sendInfo(value)
   {
      if(window.XMLHttpRequest){
      request=new XMLHttpRequest();
      }
      else if(window.ActiveXObject){
      request=new ActiveXObject("Microsoft.XMLHTTP");
      }

      try
      {
         request.onreadystatechange=getInfo;
         request.open("GET", "./StudentHomePageServlet", true);
         request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
         request.send();
      }
      catch(e)
      {
```

```
        alert("Unable to connect to server");
        }
    }

    function getInfo(){
    if(request.readyState==4 && request.status==200){
        var val=request.responseText;
        document.getElementById('classes').innerHTML=val;
    }
    }



    function getNotifs(value)
    {
        if(window.XMLHttpRequest){
        request1=new XMLHttpRequest();
        }
        else if(window.ActiveXObject){
        request1=new ActiveXObject("Microsoft.XMLHTTP");
        }

        try
        {
            request1.onreadystatechange=printNotifs;
            request1.open("GET", "./StudentNotificationServlet", true);
            request1.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
            request1.send();
        }
        catch(e)
        {
        alert("Unable to connect to server");
        }
    }

    function printNotifs(){
    if(request1.readyState==4 && request1.status==200){
        var val1=request1.responseText;

document.getElementById('notifs').innerHTML=document.getElementById('notifs').innerHTML+val1;
    }
    }

  </script>
```

```
</head>

<body onload="sendInfo();getNotifs()">
<div class="container">
	<img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
	<h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
	<div class="topnav">
		<a href="AMSLogout">Logout</a>
		<a href="./StudentDiscrepancyHome.html">Discrepancy</a>
		<a href="./StudentODHome.html">OD</a>
		<a class="active" href="./StudentHomePage.html">Home</a>
	</div>
</div>

<aside id="notifs">
	<p> Notification panel</p>
</aside>

<div class="section">
	<form action="StudentCourseAttendanceServlet" method="POST">
		<div class="courses" id="classes">
		</div>
	</form>
</div>

</body>

</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\StudentOD.html<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>

   <style>
      body{
         font-family:'Garamond';
         font-size:130%;
         color:#D79D30;
         background-color:Black;
```

```
        padding-left: 8px;
    }
    .topnav{
        width: 100%;
        overflow: hidden;
        background-color:#D79D30;
    }
    .topnav a{
        float: right;
          color: Black;
          text-align: center;
          padding: 14px 16px;
          text-decoration: none;
          font-size: 17px;
          width:12%;
    }

    .topnav a:hover{
        color:#fbfcd4;
    }
    .topnav a.active {
          color: #fbfcd4;
      }


      table {
        margin-left: auto;
        margin-right: auto;
        width: 50%;
        padding: 10px;
    }

    th,td{
        height: 35px;
        vertical-align: middle;
        padding: 15px;
        text-align: left;
        background-color: #151515;
    }

</style>

<script>
    var request;
```

```
    function sendInfo(value)
    {
       if(window.XMLHttpRequest){
       request=new XMLHttpRequest();
       }
       else if(window.ActiveXObject){
       request=new ActiveXObject("Microsoft.XMLHTTP");
       }

       try
       {
          request.onreadystatechange=getInfo;
          request.open("GET", "./StudentODViewServlet", true);
          request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
          request.send();
       }
       catch(e)
       {
       alert("Unable to connect to server");
       }
    }

    function getInfo(){
    if(request.readyState==4 && request.status==200){
       var val=request.responseText;
       document.getElementById('odlist').innerHTML=val;
    }
    }



  </script>



</head>


<body onload="sendInfo()">

  <div class="container">
                <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo">
                <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
```

```
		<div class="topnav">
			<a href="AMSLogout">Logout</a>
			<a href="./StudentDiscrepancyHome.html">Discrepancy</a>
			<a class="active" href="./StudentODHome.html">OD</a>
			<a href="./StudentHomePage.html">Home</a>
		</div>
	</div>
  <h4>Your OD Applications</h4>
  <div id="odlist"></div>

</body>
</html>C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\StudentODHome.html<!DOCTYPE
html>
<html>
<head>
	<title>Attendance Mangament System</title>
	<style>
		body{
			font-family:'Garamond';
			font-size:130%;
			color:#D79D30;
			background-color:Black;
			padding-left: 8px;
		}
		.topnav{
			width: 100%;
			overflow: hidden;
			background-color:#D79D30;
		}
		.topnav a{
			float: right;
			  color: Black;
			  text-align: center;
			  padding: 14px 16px;
			  text-decoration: none;
			  font-size: 17px;
			  width:12%;
		}

		.topnav a:hover{
			color:#fbfcd4;
		}
		.topnav a.active {
			  color: #fbfcd4;
```

```
  }
input[type=text], input[type=password]{
width: 170px;
height: 7px;
padding: 12px;
margin: 7px 4px 22px 0;
border: none;
background: #fbfcd4;
box-sizing: border-box;

}
button,input[type=button]{
        border: 2px solid #D79D30;
        color: #fbfcd4;
        background-color: #151515;
        border-radius: 4px;
        width: 120px;
        height: 25px;
}
button:hover,input[type=button]:hover{
        border: 2px solid black;
        color: black;
        background-color: #D79D30;
}

 form{
          background-color: #151515;
        align-self: center;
        box-sizing: border-box;
        padding: 70px;
        padding-left: 140px;
        margin-left: auto;
        margin-right: auto;
        width: 560px;
        height: 490px;
 }

 aside {
 width: 20%;
 padding-left: 15px;
 float: left;
 font-style: italic;
 background-color: #151515;
 color: #D79D30;
```

```
                }
        </style>
</head>
<body>


        <div class="container">
                <img class="logo" src="logo.jpg" style="width:6%;height:16%;float:left" alt="logo" >
                <h2 style="align-self:left;color:#D79D30;padding-top: 20px;"> Attendance Management
System</h2>
                <div class="topnav">
                        <a href="AMSLogout">Logout</a>
                        <a href="./StudentDiscrepancyHome.html">Discrepancy</a>
                        <a class="active" href="./StudentODHome.html">OD</a>
                        <a href="./StudentHomePage.html">Home</a>
                </div>
        </div>

        <form name="s_odform" action="./StudentODSubmitServlet" method="POST">
                <h3>Request for OD</h3>
                <label for="acid" >Course ID:</label>
                <input type="text" name="acid" id="acid" style="margin-left: 58px;"><br>
                <label for="sessionID">Session ID:</label>
                <input type="text" name="session_id" id="sessionID" style="margin-left: 55px;"><br>
                <label for="justification">Justification:</label>
                <input type="text" name="justification" id="justification" style="margin-left:
48px;"><br>
                <label for="proof">Proof(drive link):</label>
                <input type="text" name="proof" id="proof" style="margin-left: 10px;"><br>
                <button type="submit" name="submit" value="Submit" style="margin-left:
30px;">Submit</button>
                <button type="submit" formaction="./StudentOD.html" style="margin-left: 10px;">View
History</button>
        </form>

</body>
</html>
```

JAVA SERVLETS:
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminAddCourseServlet.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
```

```
import java.sql.*;
import java.time.*;


public class AdminAddCourseServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";

         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            String query="INSERT into course VALUES
("+"'"+request.getParameter("course_id")+"',"+"'"+request.getParameter("description")+"',"+"'"+request.g
etParameter("department")+"',"
                                    +""+Integer.parseInt(request.getParameter("credits"))+")";
            cstmt=conn.prepareCall(query);
            cstmt.executeUpdate();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
```

```
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
           try{
              if(stmt!=null)
                 stmt.close();
           }
           catch(SQLException se2){
           }
           try{
              if(conn!=null)
                 conn.close();
           }
           catch(SQLException se){
              se.printStackTrace();
           }
        }
        RequestDispatcher rd = request.getRequestDispatcher("./adminExport.html");
                              rd.forward(request, res);

     }
  }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminAddFacultyServlet.jav
aimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class AdminAddFacultyServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
```

```
  static final String DB_URL = "jdbc:mysql://localhost/AMS";
  static final String USER = "root";
  static final String PASS = "aarthi00*";


 public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
 {

     HttpSession session = request.getSession(false);
     Connection conn=null;
     PreparedStatement stmt=null;
     CallableStatement cstmt=null;
     String code="";
     if(session == null){
        res.sendRedirect("./index.html");
     }
     else{
        res.setContentType("text/html") ;
        PrintWriter out = res.getWriter();
        String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
        try{
           Class.forName("com.mysql.jdbc.Driver");
           conn = DriverManager.getConnection(DB_URL,USER,PASS);
           stmt= conn.prepareStatement("INSERT INTO faculty VALUES (?,?,?,?,?)");
           stmt.setString(1,request.getParameter("faculty_id"));
                                    stmt.setString(2,request.getParameter("f_name"));
           stmt.setString(3,request.getParameter("dept"));
           stmt.setString(4,request.getParameter("email"));
           stmt.setString(5,request.getParameter("desig"));
           stmt.executeUpdate();

           stmt.close();
           conn.close();
        }
        catch(SQLException se){
           se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
           try{
              if(stmt!=null)
```

```
                stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }
         RequestDispatcher rd = request.getRequestDispatcher("./adminExport.html");
                              rd.forward(request, res);

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminAddStudentServlet.ja
vaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class AdminAddStudentServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
```

```
{

    HttpSession session = request.getSession(false);
    Connection conn=null;
    PreparedStatement stmt=null;
    CallableStatement cstmt=null;
    String code="";
    if(session == null){
       res.sendRedirect("./index.html");
    }
    else{
       res.setContentType("text/html") ;
       PrintWriter out = res.getWriter();
       String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
       try{
          Class.forName("com.mysql.jdbc.Driver");
          conn = DriverManager.getConnection(DB_URL,USER,PASS);
          stmt=conn.prepareStatement("INSERT INTO student VALUES (?,?,?,?,?);");
          stmt.setString(1,request.getParameter("student_id"));
                                    stmt.setString(2,request.getParameter("s_name"));
          stmt.setString(3,request.getParameter("dept"));
          stmt.setString(4,request.getParameter("email"));
                                    int joining_year=Integer.parseInt(request.getParameter("joining_year"));
                                    stmt.setInt(5,joining_year);
          stmt.executeUpdate();

          stmt.close();
          conn.close();
       }
       catch(SQLException se){
          se.printStackTrace();
       }
       catch(Exception e){
       e.printStackTrace();
       }
       finally{
          try{
             if(stmt!=null)
                stmt.close();
          }
          catch(SQLException se2){
          }
          try{
             if(conn!=null)
```

```
                conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }
         RequestDispatcher rd = request.getRequestDispatcher("./adminExport.html");
                        rd.forward(request, res);

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminAssignCourseServlet.j
avaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class AdminAssignCourseServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
```

```
String code="";
if(session == null){
   res.sendRedirect("./index.html");
}
else{
   res.setContentType("text/html") ;
   PrintWriter out = res.getWriter();
   String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
   try{
      Class.forName("com.mysql.jdbc.Driver");
      conn = DriverManager.getConnection(DB_URL,USER,PASS);
      String acid=request.getParameter("acid");
      String course_id=request.getParameter("course_id");
      String faculty_id=request.getParameter("facultyid");
      stmt=conn.prepareStatement("INSERT INTO active_course VALUES (?,?,?);");
      stmt.setString(1,acid);
                                    stmt.setString(2,course_id);
      stmt.setString(3,faculty_id);
      stmt.executeUpdate();
      String sid=request.getParameter("student_id");
      String[] sid_list=sid.split(",");
      float initial_percentage=0;
      for(int i=0;i<sid_list.length;i++){
         stmt=conn.prepareStatement("insert into enrollment values (?,?,?)");
         stmt.setString(1,acid);
         stmt.setString(2,sid_list[i]);
         stmt.setFloat(3,initial_percentage);
         stmt.executeUpdate();
      }


      stmt.close();
      conn.close();
   }
   catch(SQLException se){
      se.printStackTrace();
   }
   catch(Exception e){
   e.printStackTrace();
   }
   finally{
      try{
         if(stmt!=null)
            stmt.close();
```

```
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }
         RequestDispatcher rd = request.getRequestDispatcher("./adminExport.html");
                              rd.forward(request, res);

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminDeleteCourseServlet.j
avaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class AdminDeleteCourseServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {
```

```
HttpSession session = request.getSession(false);
Connection conn=null;
PreparedStatement stmt=null;
CallableStatement cstmt=null;
String code="";
if(session == null){
   res.sendRedirect("./index.html");
}
else{
   res.setContentType("text/html") ;
   PrintWriter out = res.getWriter();
   String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
   try{
      Class.forName("com.mysql.jdbc.Driver");
      conn = DriverManager.getConnection(DB_URL,USER,PASS);
      stmt=conn.prepareStatement("DELETE FROM active_course WHERE acid=?;");
      stmt.setString(1,request.getParameter("acid"));
      stmt.executeUpdate();

      stmt.close();
      conn.close();
   }
   catch(SQLException se){
      se.printStackTrace();
   }
   catch(Exception e){
   e.printStackTrace();
   }
   finally{
      try{
         if(stmt!=null)
            stmt.close();
      }
      catch(SQLException se2){
      }
      try{
         if(conn!=null)
            conn.close();
      }
      catch(SQLException se){
         se.printStackTrace();
      }
   }
```

```
            RequestDispatcher rd = request.getRequestDispatcher("./adminExport.html");
                                rd.forward(request, res);

        }
    }

    public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
    {
        doGet(request,res);
    }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminDeleteFacultyServlet.j
avaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class AdminDeleteFacultyServlet extends HttpServlet
{

    static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/AMS";
    static final String USER = "root";
    static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

        HttpSession session = request.getSession(false);
        Connection conn=null;
        PreparedStatement stmt=null;
        CallableStatement cstmt=null;
        String code="";
        if(session == null){
            res.sendRedirect("./index.html");
        }
        else{
            res.setContentType("text/html") ;
```

```
        PrintWriter out = res.getWriter();
        String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
        try{
           Class.forName("com.mysql.jdbc.Driver");
           conn = DriverManager.getConnection(DB_URL,USER,PASS);
           stmt=conn.prepareStatement("DELETE FROM faculty WHERE faculty_id=?;");
           stmt.setString(1,request.getParameter("faculty_id"));
           stmt.executeUpdate();

           stmt.close();
           conn.close();
        }
        catch(SQLException se){
           se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
           try{
              if(stmt!=null)
                 stmt.close();
           }
           catch(SQLException se2){
           }
           try{
              if(conn!=null)
                 conn.close();
           }
           catch(SQLException se){
              se.printStackTrace();
           }
        }
        RequestDispatcher rd = request.getRequestDispatcher("./AdminHomePage.html");
                              rd.forward(request, res);

     }
  }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {
     doGet(request,res);
  }
```

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminDeleteStudentServlet.j
avaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;

public class AdminDeleteStudentServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";

  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            stmt=conn.prepareStatement("DELETE FROM student WHERE student_id=?;");
            stmt.setString(1,request.getParameter("student_id"));
            stmt.executeUpdate();

            stmt.close();
            conn.close();

```
        }
        catch(SQLException se){
           se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
           try{
              if(stmt!=null)
                 stmt.close();
           }
           catch(SQLException se2){
           }
           try{
              if(conn!=null)
                 conn.close();
           }
           catch(SQLException se){
              se.printStackTrace();
           }
        }
        RequestDispatcher rd = request.getRequestDispatcher("./adminExport.html");
                              rd.forward(request, res);

     }
  }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AdminExportServlet.javaimp
ort java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.*;
import java.util.Date;

public class AdminExportServlet extends HttpServlet
```

```
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

       HttpSession session = request.getSession(false);
       Connection conn=null;
       PreparedStatement stmt=null;
       CallableStatement cstmt=null;
       ResultSet rs;
       String code="";
       String code1="";
       String query;
       ArrayList<String> ids = new ArrayList<String>();
       ArrayList<String> names = new ArrayList<String>();
       ArrayList<Float> percentages = new ArrayList<Float>();
       ArrayList<Integer> sess_ids = new ArrayList<Integer>();
       if(session == null){
          res.sendRedirect("./index.html");
       }
       else{
          res.setContentType("text/html") ;
          PrintWriter out = res.getWriter();
          String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
          String acid=(String)request.getParameter("acid");
          int tot_classes=0;
          float avg_percentage=0;;
          int nob75=0;
          try{
             Class.forName("com.mysql.jdbc.Driver");
             conn = DriverManager.getConnection(DB_URL,USER,PASS);
             stmt=conn.prepareStatement("select course.description from course,active_course where
course.course_id=active_course.course_id && active_course.acid=?");
             stmt.setString(1,acid);
             rs=stmt.executeQuery();
             String coursename="";
```

```
            while(rs.next()){
               coursename=rs.getString("description");
            }
            code1="<h2>"+acid+"-"+coursename+"</h2>";
            query="{call proc_get_students(?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,acid);
            rs = cstmt.executeQuery();
            String thead="<table><tr><th>Register Number</th><th>Name</th><th>Percentage</th>";
            while(rs.next()){
               String id=rs.getString("student_id");
               String name=rs.getString("name");
               Float p=rs.getFloat("percentage");
               ids.add(id);
               names.add(name);
               percentages.add(p);
            }
            stmt=conn.prepareStatement("select * from sessions where acid=? order by session_id desc");
            stmt.setString(1,acid);
            rs=stmt.executeQuery();
            String timestamps="<tr><th></th><th></th><th></th>";
            while(rs.next()){
               int sess=rs.getInt("session_id");
               sess_ids.add(sess);
               java.sql.Timestamp ts= rs.getTimestamp("timestamp");
               SimpleDateFormat sdf1 = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss.S");
               String valueFromDB = ts.toString();
               Date d1 = sdf1.parse(valueFromDB);
               SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-YY");
               String dateWithoutTime = sdf.format(d1);
               thead=thead+"<th>"+sess+"</th>";
               timestamps=timestamps+"<th>"+dateWithoutTime+"</th>";
               tot_classes++;
            }
            thead=thead+"</tr>"+timestamps+"</tr>";
            code=code+thead;
            for(int i=0;i<ids.size();i++){
               avg_percentage+=percentages.get(i);
               if(percentages.get(i)<75){
                  nob75++;
               }
               String
trow="<tr>"+"<td>"+ids.get(i)+"</td>"+"<td>"+names.get(i)+"</td>"+"<td>"+percentages.get(i)+"</td>
";
```

```
                query="{call proc_student_attendance(?,?)}";
                cstmt=conn.prepareCall(query);
                cstmt.setString(1,ids.get(i));
                cstmt.setString(2,acid);
                rs = cstmt.executeQuery();
                while(rs.next()){
                   int isabsent=rs.getInt("absent");
                   String a;
                   if(isabsent==1){
                      a="Absent";
                   }
                   else{
                      a="Present";
                   }
                   trow = trow  +"<td>"+a+"</td>";
                }
                code=code+trow+"</tr>";
             }
             code=code+"<tr><td>Absentees</td><td></td><td></td>";
             avg_percentage=avg_percentage/ids.size();
             for(int i=0;i<sess_ids.size();i++){
                String abslis="<td><ul>";
                stmt=conn.prepareStatement("select name from absentee_list,student where acid=? &&
session_id=? && student.student_id=absentee_list.student_id;");
                stmt.setString(1,acid);
                stmt.setInt(2,sess_ids.get(i));
                rs=stmt.executeQuery();
                while(rs.next()){
                   abslis=abslis+"<li>"+rs.getString("name")+";</li>";
                }
                abslis=abslis+"</ul></td>";
                code=code+abslis;
             }
             code=code+"</tr></table>";
             out.print(code);
             rs.close();
             cstmt.close();
             stmt.close();
             conn.close();
          }
          catch(SQLException se){
             se.printStackTrace();
          }
          catch(Exception e){
```

```
            e.printStackTrace();
            }
            finally{
               try{
                  if(stmt!=null)
                     stmt.close();
               }
               catch(SQLException se2){
               }
               try{
                  if(conn!=null)
                     conn.close();
               }
               catch(SQLException se){
                  se.printStackTrace();
               }
            }

         }
      }

      public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
      {
         doGet(request,res);
      }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AMSLoginServlet.javaimpor
t java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.util.Date;


public class AMSLoginServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";
```

```
  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

                HttpSession session = request.getSession();
                Date createTime = new Date(session.getCreationTime());
                String userIDKey = new String("username");
                String userID = request.getParameter("username");
                String pwd = request.getParameter("password");
                PrintWriter out = res.getWriter();
                String user_type="";
                int flag=0;
                 // Check if this is new comer on your web page.

                 Connection conn = null;
                 PreparedStatement stmt = null;
                 ResultSet rs=null;

                try {
                        Class.forName(JDBC_DRIVER);
                        conn = DriverManager.getConnection(DB_URL, USER, PASS);
        String query="{call proc_login(?,?,?)}";
        CallableStatement cstmt=conn.prepareCall(query);
        cstmt.setString(1,userID);
        cstmt.setString(2,pwd);
        cstmt.registerOutParameter(3,Types.VARCHAR);
        cstmt.executeUpdate();
        user_type=cstmt.getString(3);
                        cstmt.close();

                if(user_type.equals("student"))
                 {
                                RequestDispatcher rd =
request.getRequestDispatcher("./StudentHomePage.html");
            session.setAttribute(userIDKey, userID);
                                stmt=conn.prepareStatement("select name from student where
student_id=?");
                                stmt.setString(1,userID);
                                rs=stmt.executeQuery();
                                while(rs.next())
                                        session.setAttribute("Name", rs.getString("name"));
                                rd.forward(request, res);
```

```
                }

                else if(user_type.equals("faculty"))
                {
                                RequestDispatcher rd =
request.getRequestDispatcher("./FacultyHomePage.html");
            session.setAttribute(userIDKey, userID);
                                stmt=conn.prepareStatement("select name from faculty where
faculty_id=?");
                                stmt.setString(1,userID);
                                rs=stmt.executeQuery();
                                while(rs.next())
                                        session.setAttribute("Name", rs.getString("name"));
                                rd.forward(request, res);
                }

                else if(user_type.equals("admin"))
                {
                                RequestDispatcher rd =
request.getRequestDispatcher("./adminExport.html");
            session.setAttribute(userIDKey, userID);
                                rd.forward(request, res);
                }

        else{
          out.println("Sorry UserName or Password Error!");
          RequestDispatcher rd=request.getRequestDispatcher("./index.html");
          rd.include(request, res);
        }
                 rs.close();
                stmt.close();
                conn.close();
                }
                catch(SQLException se) {
                        //Handle errors for JDBC
                        se.printStackTrace();
                        } catch(Exception e) {
                        //Handle errors for Class.forName
                        e.printStackTrace();
                        } finally {
                                //finally block used to close resources
                                try {
                                        if(stmt != null)
                                        stmt.close();
```

```
                                    }
                                    catch(SQLException se2) {
                                    } // nothing we can do
                                    try {
                                    if(conn!=null)
                                    conn.close();
                                    }
                                    catch(SQLException se) {
                                    se.printStackTrace();
                                  } //end finally try
                                }
  }

    public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
    {
       doGet(request,res);
    }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\AMSLogoutServlet.javaimp
ort java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;

public class AMSLogoutServlet extends HttpServlet
{

  static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
  static final String DB_URL = "jdbc:mysql://localhost/AMS";
  static final String USER = "root";
  static final String PASS = "aarthi00*";

  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {
    CallableStatement cstmt=null;
    Connection conn=null;
    res.setContentType("text/html") ;
    PrintWriter out = res.getWriter();
    HttpSession session = request.getSession(true);
    String id=(String)session.getAttribute("username");
    try{
```

```
      Class.forName("com.mysql.jdbc.Driver");
      conn = DriverManager.getConnection(DB_URL,USER,PASS);
      cstmt=conn.prepareCall("call proc_remove_notifications(?)");
      cstmt.setString(1,id);
      cstmt.executeUpdate();
      cstmt.close();
      conn.close();
   }
   catch(SQLException se){
        se.printStackTrace();
   }
   catch(Exception e){
     e.printStackTrace();
   }
   finally{
        try{
           if(cstmt!=null)
              cstmt.close();
        }
        catch(SQLException se2){
        }
        try{
           if(conn!=null)
              conn.close();
        }
        catch(SQLException se){
           se.printStackTrace();
        }
     }
   session.invalidate();
   RequestDispatcher rd=request.getRequestDispatcher("./index.html");
   rd.forward(request, res);
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}
C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyAddAbsenteeServlet.ja
vaimport java.io.*;
import javax.servlet.*;
```

```
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.time.LocalDateTime;

public class FacultyAddAbsenteeServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String query;
      ArrayList<String> ids = new ArrayList<String>();
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         String fid=(String)session.getAttribute("username");
         String acid=(String)request.getParameter("acid");
         String stu_id=(String)request.getParameter("student_id");
         Integer sess_id=Integer.parseInt(request.getParameter("session_id"));
         String text="";
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            query="{call proc_add_absentee(?,?,?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,stu_id);
            cstmt.setString(2,acid);
            cstmt.setInt(3,sess_id);
```

```
            cstmt.executeUpdate();
            query="{call proc_student_percentage(?,?,?,?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,stu_id);
            cstmt.setString(2,acid);
            cstmt.registerOutParameter(3,Types.INTEGER);
            cstmt.registerOutParameter(4,Types.INTEGER);
            cstmt.executeUpdate();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }
         request.setAttribute("acid", acid);
         RequestDispatcher rd=request.getRequestDispatcher("./CourseAttendance");
         rd.forward(request, res);
      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }
```

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyCourseAttendanceServlet.javaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.*;
import java.util.Date;

public class FacultyCourseAttendanceServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";

  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException, IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      ResultSet rs;
      String code="";
      String code1="";
      String query;
      ArrayList<String> ids = new ArrayList<String>();
      ArrayList<String> names = new ArrayList<String>();
      ArrayList<Float> percentages = new ArrayList<Float>();
      ArrayList<Integer> sess_ids = new ArrayList<Integer>();
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String fid=(String)session.getAttribute("username");
         String acid=request.getParameter("acid");

```
        int tot_classes=0;
        float avg_percentage=0;;
        int nob75=0;
        try{
           Class.forName("com.mysql.jdbc.Driver");
           conn = DriverManager.getConnection(DB_URL,USER,PASS);
           stmt=conn.prepareStatement("select course.description from course,active_course where
course.course_id=active_course.course_id && active_course.acid=?");
           stmt.setString(1,acid);
           rs=stmt.executeQuery();
           String coursename="";
           while(rs.next()){
              coursename=rs.getString("description");
           }
           code1="<h2>"+acid+"-"+coursename+"</h2>";
           query="{call proc_get_students(?)}";
           cstmt=conn.prepareCall(query);
           cstmt.setString(1,acid);
           rs = cstmt.executeQuery();
           String thead="<table><tr><th>Register Number</th><th>Name</th><th>Percentage</th>";
           while(rs.next()){
              String id=rs.getString("student_id");
              String name=rs.getString("name");
              Float p=rs.getFloat("percentage");
              ids.add(id);
              names.add(name);
              percentages.add(p);
           }
           stmt=conn.prepareStatement("select * from sessions where acid=? order by session_id desc");
           stmt.setString(1,acid);
           rs=stmt.executeQuery();
           String timestamps="<tr><th></th><th></th><th></th>";
           while(rs.next()){
              int sess=rs.getInt("session_id");
              sess_ids.add(sess);
              java.sql.Timestamp ts= rs.getTimestamp("timestamp");
              SimpleDateFormat sdf1 = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss.S");
              String valueFromDB = ts.toString();
              Date d1 = sdf1.parse(valueFromDB);
              SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-YY");
              String dateWithoutTime = sdf.format(d1);
              thead=thead+"<th>"+sess+"</th>";
              timestamps=timestamps+"<th>"+dateWithoutTime+"</th>";
              tot_classes++;
```

```
            }
            thead=thead+"</tr>"+timestamps+"</tr>";
            code=code+thead;
            for(int i=0;i<ids.size();i++){
               avg_percentage+=percentages.get(i);
               if(percentages.get(i)<75){
                  nob75++;
               }
               String
trow="<tr>"+"<td>"+ids.get(i)+"</td>"+"<td>"+names.get(i)+"</td>"+"<td>"+percentages.get(i)+"</td>
";
               query="{call proc_student_attendance(?,?)}";
               cstmt=conn.prepareCall(query);
               cstmt.setString(1,ids.get(i));
               cstmt.setString(2,acid);
               rs = cstmt.executeQuery();
               while(rs.next()){
                  int isabsent=rs.getInt("absent");
                  String a;
                  if(isabsent==1){
                     a="Absent";
                  }
                  else{
                     a="Present";
                  }
                  trow = trow  +"<td>"+a+"</td>";
               }
               code=code+trow+"</tr>";
            }
            code=code+"<tr><td>Absentees</td><td></td><td></td><td></td>";
            avg_percentage=avg_percentage/ids.size();
            for(int i=0;i<sess_ids.size();i++){
               String abslis="<td><ul>";
               stmt=conn.prepareStatement("select name from absentee_list,student where acid=? &&
session_id=? && student.student_id=absentee_list.student_id;");
               stmt.setString(1,acid);
               stmt.setInt(2,sess_ids.get(i));
               rs=stmt.executeQuery();
               while(rs.next()){
                  abslis=abslis+"<li>"+rs.getString("name")+"</li>";
               }
               abslis=abslis+"</ul></td>";
               code=code+abslis;
            }
```

```
            code1=code1+"<p class=fp>Total Classes:"+tot_classes+"</p>"+
            "<p class=fp>Average Percentage:"+avg_percentage+"</p>"+
            "<p class=fp>No. of students below 75%:"+nob75+"</p>";
            code1=code1+"<div id='stuform'></div>";
            code=code+"</tr>";
            code=code+"<tr><td>Update Absentee List</td><td></td><td></td>";
            for(int i=0;i<sess_ids.size();i++){
               String abslis="<td><input type='button' value='Add'
onclick=\"addabsentee('"+sess_ids.get(i)+"')\">" + "<input type='button' value='Remove'
onclick=\"removeabsentee('"+sess_ids.get(i)+"')\"></td>";
               code=code+abslis;
            }
            code=code+"</tr></table>";
            out.print(docType +
          "<html>\n" +
            "<head><title>" + "Faculty" + "</title>\n" +
            "<LINK REL='stylesheet' href='./style.css' type='text/css'> "+
            "<script>    function addabsentee(id) {var down = document.getElementById(\"stuform\");var
br = document.createElement('br');var form = document.createElement('form');form.setAttribute('method',
'post');form.setAttribute('action', 'AddAbsentee');var x=document.createElement('LABEL');var t =
document.createTextNode('Add');x.setAttribute('for', 'Add');x.appendChild(t);var sess =
document.createElement('input');sess.setAttribute('type', 'hidden');sess.setAttribute('name',
'session_id');sess.setAttribute('value',id);var acid =
document.createElement('input');acid.setAttribute('type', 'hidden');acid.setAttribute('name',
'acid');acid.setAttribute('value','"+acid+"');var stu =
document.createElement('input');stu.setAttribute('type', 'text');stu.setAttribute('name',
'student_id');stu.setAttribute('placeholder', 'Student ID');var s =
document.createElement('input');s.setAttribute('type', 'submit');s.setAttribute('value',
'Submit');form.appendChild(sess);form.appendChild(acid);
form.appendChild(br.cloneNode());form.appendChild(x);form.appendChild(stu);
form.appendChild(br.cloneNode()); form.appendChild(s); down.appendChild(form);} function
removeabsentee(id) {var down = document.getElementById(\"stuform\");var br =
document.createElement('br'); var form = document.createElement('form');form.setAttribute('method',
'post');form.setAttribute('action', 'RemoveAbsentee');var x=document.createElement('LABEL');var t =
document.createTextNode('Remove');x.setAttribute('for', 'Remove');x.appendChild(t);var sess =
document.createElement('input');sess.setAttribute('type', 'hidden');sess.setAttribute('name',
'session_id');sess.setAttribute('value',id);var acid =
document.createElement('input');acid.setAttribute('type', 'hidden');acid.setAttribute('name',
'acid');acid.setAttribute('value','"+acid+"');var stu =
document.createElement('input');stu.setAttribute('type', 'text');stu.setAttribute('name',
'student_id');stu.setAttribute('placeholder', 'Student ID');var s =
document.createElement('input');s.setAttribute('type', 'submit');s.setAttribute('value',
'Submit');form.appendChild(sess);form.appendChild(acid);
```

```
form.appendChild(br.cloneNode());form.appendChild(x); form.appendChild(stu);
form.appendChild(br.cloneNode()); form.appendChild(s); down.appendChild(form);}</script>"+
            "</head>\n"+
            "<body>\n" +
               code1+
               code+
            "</body>"+
         "</html>");
            rs.close();
            cstmt.close();
            stmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }
```

```
}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyDiscrepancyPageServlet.javaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;

public class FacultyDiscrepancyPageServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException, IOException
  {

       HttpSession session = request.getSession(false);
       Connection conn=null;
       PreparedStatement stmt=null;
       CallableStatement cstmt=null;
       String code="";
       if(session == null){
          res.sendRedirect("./index.html");
       }
       else{
          res.setContentType("text/html") ;
          PrintWriter out = res.getWriter();
          String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
          String fid=(String)session.getAttribute("username");
          try{
             Class.forName("com.mysql.jdbc.Driver");
             conn = DriverManager.getConnection(DB_URL,USER,PASS);
             code="<table> <tr> <th>Student ID</th> <th>Active Course ID</th> <th>Session ID</th> <th>Justification</th> <th>Status</th></tr>";
             String query="{call proc_get_faculty_discrepancy(?)}";
             cstmt=conn.prepareCall(query);
             cstmt.setString(1,fid);
             ResultSet rs = cstmt.executeQuery();
             String app="";
```

```
            String done="";
            while(rs.next()){
               int sess_id=rs.getInt("session_id");
               String acid=rs.getString("acid");
               String stu_id=rs.getString("student_id");
               String just=rs.getString("justification");
               String stat=rs.getString("status");
               if(stat.equals("applied")){
                  String id=stu_id+";"+acid+";"+sess_id;
                  app = app + "<form id='" + id +"'> <tr id='" + id+"table"+"'> <td><input type='text'
name='stuid' value='" + stu_id +"' readonly></td>" + "<td><input type='text' name='acid' value='" + acid
+"' readonly></td>" + "<td><input type='text' name="sessid' value='" + sess_id +"' readonly></td>" +
"<td><input type='text' name='just' value='" + just +"' readonly></td>" + "<td><input type='button'
name='approve' value='Approve' id='"+ id+"approved"+ "'onclick=\"discprocess('approved','"+ id+"')\">
<input type='button' value='Deny' name='deny' id='"+ id+"denied"+"'onclick=\"discprocess('denied','"+
id+"')\"></td></tr></form>";
               }
               else{
                  done=done
+"<tr><td>"+stu_id+"</td>"+"<td>"+acid+"</td>"+"<td>"+sess_id+"</td><td>"+just+"</td><td>"+stat
+"</td></tr>";
               }
            }
            code=code+app+done+"</table>";
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
```

```
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }
         out.print(docType +
         "<html>\n" +
            "<head><title>" + "Faculty" + "</title>\n" +
            "<style> table, th, td {border: 1px solid black;}
table{width:80%;margin-top:20px;margin-bottom:20px;}</style>"+
            "</head>\n"+
            "<body>\n" +
               code+
            "</body>"+
         "</html>");

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyHomePageServlet.jav
aimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;

public class FacultyHomePageServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
```

```
 {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="<h1>Hello,"+(String)session.getAttribute("Name")+"</h1>";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String fid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            String query="{call proc_faculty_course_list(?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,fid);
            ResultSet rs = cstmt.executeQuery();
            String thead="<table><tr><th>Course ID</th><th>Course Name</th><th>Enter
Attendance</th></tr>";
            code=code+thead;
            while(rs.next()){
               String acid=rs.getString("acid");
               code=code +"<tr><td class='courseref'><a
href='CourseAttendance?acid="+acid+"'>"+acid+"</a></td>"+"<td>"+rs.getString("description")+"</td>
"+"<td class='bttd'><form method='POST' action='./Attendance'>"+"<input type='hidden' name='acid'
value='"+acid+"'><input type='submit' class='tdbutton' value='Take Attendance'></form>"+"</td><tr>";
            }
            code=code+"</table";
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
```

```
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }
         out.print(code);

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyODPageServlet.javai
mport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;

public class FacultyODPageServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
```

```
 {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="ods";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String fid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            code="<table> <tr> <th>Student ID</th> <th>Active Course ID</th> <th>Session ID</th>
<th>Justification</th> <th>Proof</th> <th>Status</th></tr>";
            String query="{call proc_get_faculty_od(?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,fid);
            ResultSet rs = cstmt.executeQuery();
            String app="";
            String done="";
            while(rs.next()){
               int sess_id=rs.getInt("session_id");
               String acid=rs.getString("acid");
               String stu_id=rs.getString("student_id");
               String just=rs.getString("justification");
               String prf=rs.getString("proof");
               String stat=rs.getString("status");
               if(stat.equals("applied")){
                  String id=stu_id+";"+acid+";"+sess_id;
                  app = app + "<form id='" + id +"'> <tr id='" + id+"table"+"'> <td><input type='text'
name='stuid' value='" + stu_id +"' readonly></td>" + "<td><input type='text' name='acid' value='" + acid
+"' readonly></td>" + "<td><input type='text' name="sessid' value='" + sess_id +"' readonly></td>" +
"<td><input type='text' name='just' value='" + just +"' readonly></td>" + "<td><input type='text'
name='proof' value='" + prf +"' readonly></td>" +"<td><input type='button' name='approve'
value='Approve' id='"+ id+"approved"+ "'onclick=\"odprocess('approved','"+ id+"')\"> <input
type='button' value='Deny' name='deny' id='"+ id+"denied"+"'onclick=\"odprocess('denied','"+
id+"')\"></td></tr></form>";
               }
```

```
                else{
                    done=done
+"<tr><td>"+stu_id+"</td>"+"<td>"+acid+"</td>"+"<td>"+sess_id+"</td><td>"+just+"</td><td>"+prf
+"</td><td>"+stat+"</td></tr>";
                }
            }
            code=code+app+done+"</table>";
            rs.close();
            cstmt.close();
            conn.close();
        }
        catch(SQLException se){
            se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
            try{
                if(stmt!=null)
                    stmt.close();
            }
            catch(SQLException se2){
            }
            try{
                if(conn!=null)
                    conn.close();
            }
            catch(SQLException se){
                se.printStackTrace();
            }
        }
        out.print(docType +
        "<html>\n" +
            "<head><title>" + "Faculty" + "</title>\n" +
            "<style> table, th, td {border: 1px solid black;}
table{width:80%;margin-top:20px;margin-bottom:20px;}</style>"+
            "</head>\n"+
            "<body>\n" +
                code+
            "</body>"+
        "</html>");

    }
```

```
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyProcessDiscrepancy.j
avaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.time.LocalDateTime;

public class FacultyProcessDiscrepancy extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String query;
      ArrayList<String> ids = new ArrayList<String>();
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         String fid=(String)session.getAttribute("username");
         String acid=(String)request.getParameter("acid");
```

```
        String stu_id=(String)request.getParameter("stu_id");
        Integer sess_id=Integer.parseInt(request.getParameter("sess_id"));
        String stat=(String)request.getParameter("status");
        String text="";
        try{
           Class.forName("com.mysql.jdbc.Driver");
           conn = DriverManager.getConnection(DB_URL,USER,PASS);
           query="{call proc_process_discrepancy(?,?,?,?)}";
           cstmt=conn.prepareCall(query);
           cstmt.setString(1,stu_id);
           cstmt.setString(2,acid);
           cstmt.setInt(3,sess_id);
           cstmt.setString(4,stat);
           cstmt.executeUpdate();
           cstmt.close();
           conn.close();
        }
        catch(SQLException se){
           se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
           try{
              if(stmt!=null)
                 stmt.close();
           }
           catch(SQLException se2){
           }
           try{
              if(conn!=null)
                 conn.close();
           }
           catch(SQLException se){
              se.printStackTrace();
           }
        }
        PrintWriter out = res.getWriter();
        out.print(stat);
     }
  }
```

```
   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyProcessOD.javaimpor
t java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.time.LocalDateTime;

public class FacultyProcessOD extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String query;
      ArrayList<String> ids = new ArrayList<String>();
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         String fid=(String)session.getAttribute("username");
         String acid=(String)request.getParameter("acid");
         String stu_id=(String)request.getParameter("stu_id");
         Integer sess_id=Integer.parseInt(request.getParameter("sess_id"));
```

```
        String stat=(String)request.getParameter("status");
        String text="";
        try{
           Class.forName("com.mysql.jdbc.Driver");
           conn = DriverManager.getConnection(DB_URL,USER,PASS);
           query="{call proc_process_od(?,?,?,?)}";
           cstmt=conn.prepareCall(query);
           cstmt.setString(1,stu_id);
           cstmt.setString(2,acid);
           cstmt.setInt(3,sess_id);
           cstmt.setString(4,stat);
           cstmt.executeUpdate();
           cstmt.close();
           conn.close();
        }
        catch(SQLException se){
           se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
           try{
              if(stmt!=null)
                 stmt.close();
           }
           catch(SQLException se2){
           }
           try{
              if(conn!=null)
                 conn.close();
           }
           catch(SQLException se){
              se.printStackTrace();
           }
        }
        PrintWriter out = res.getWriter();
        out.print(stat);
     }
  }

  public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {
```

```
        doGet(request,res);
    }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyRemoveAbsenteeSer
vlet.javaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.time.LocalDateTime;

public class FacultyRemoveAbsenteeServlet extends HttpServlet
{

    static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/AMS";

    static final String USER = "root";
    static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

        HttpSession session = request.getSession(false);
        Connection conn=null;
        PreparedStatement stmt=null;
        CallableStatement cstmt=null;
        String query;
        ArrayList<String> ids = new ArrayList<String>();
        if(session == null){
            res.sendRedirect("./index.html");
        }
        else{
            res.setContentType("text/html") ;
            String fid=(String)session.getAttribute("username");
            String acid=(String)request.getParameter("acid");
            String stu_id=(String)request.getParameter("student_id");
            Integer sess_id=Integer.parseInt(request.getParameter("session_id"));
            String text="";
            try{
                Class.forName("com.mysql.jdbc.Driver");
```

```
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            query="{call proc_remove_absentee(?,?,?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,stu_id);
            cstmt.setString(2,acid);
            cstmt.setInt(3,sess_id);
            cstmt.executeUpdate();
            query="{call proc_student_percentage(?,?,?,?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,stu_id);
            cstmt.setString(2,acid);
            cstmt.registerOutParameter(3,Types.INTEGER);
            cstmt.registerOutParameter(4,Types.INTEGER);
            cstmt.executeUpdate();
            cstmt.close();
            conn.close();
        }
        catch(SQLException se){
            se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
            try{
                if(stmt!=null)
                    stmt.close();
            }
            catch(SQLException se2){
            }
            try{
                if(conn!=null)
                    conn.close();
            }
            catch(SQLException se){
                se.printStackTrace();
            }
        }
        request.setAttribute("acid", acid);
        RequestDispatcher rd=request.getRequestDispatcher("./CourseAttendance");
        rd.forward(request, res);
    }
}
```

```
   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultySubmitAttendance.ja
vaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.text.SimpleDateFormat;
import java.time.LocalDateTime;

public class FacultySubmitAttendance extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String query;
      ArrayList<String> ids = new ArrayList<String>();
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         String fid=(String)session.getAttribute("username");
         String acid=(String)request.getParameter("acid");
         String text="";
         try{
```

```
    Class.forName("com.mysql.jdbc.Driver");
    conn = DriverManager.getConnection(DB_URL,USER,PASS);
    query="{call proc_get_students(?)}";
    cstmt=conn.prepareCall(query);
    cstmt.setString(1,acid);
    ResultSet rs = cstmt.executeQuery();
    while(rs.next()){
       String stu_id=rs.getString("student_id");
       ids.add(stu_id);
    }
    query="{call proc_new_session(?,?,?)}";
    cstmt=conn.prepareCall(query);
    cstmt.setString(1,acid);
    text = request.getParameter("timestamp");
    text=text.replace("T"," ");
    text=text + ":00";
    Timestamp ts = Timestamp.valueOf(text);
    cstmt.setTimestamp(2,ts);
    cstmt.registerOutParameter(3,Types.INTEGER);
    cstmt.executeUpdate();
    int sess_id=cstmt.getInt(3);
    for(int i=0;i<ids.size();i++){
       if(request.getParameter(ids.get(i))!=null){
          query="{call proc_add_absentee(?,?,?)}";
          cstmt=conn.prepareCall(query);
          cstmt.setString(1,ids.get(i));
          cstmt.setString(2,acid);
          cstmt.setInt(3,sess_id);
          cstmt.executeUpdate();
       }
    }
    query="{call proc_acid_percentage(?)}";
    cstmt=conn.prepareCall(query);
    cstmt.setString(1,acid);
    cstmt.executeUpdate();
    rs.close();
    cstmt.close();
    conn.close();
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
e.printStackTrace();
```

```
        }
        finally{
           try{
              if(stmt!=null)
                 stmt.close();
           }
           catch(SQLException se2){
           }
           try{
              if(conn!=null)
                 conn.close();
           }
           catch(SQLException se){
              se.printStackTrace();
           }
        }
        RequestDispatcher rd = request.getRequestDispatcher("./FacultyHomePage.html");
                              rd.forward(request, res);
     }
  }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\FacultyTakeAttendance.javai
mport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;

public class FacultyTakeAttendance extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";

   static final String USER = "root";
   static final String PASS = "aarthi00*";
```

```
  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

        HttpSession session = request.getSession(false);
        Connection conn=null;
        PreparedStatement stmt=null;
        CallableStatement cstmt=null;
        String code="";
        String code1="";
        if(session == null){
           res.sendRedirect("./index.html");
        }
        else{
           res.setContentType("text/html") ;
           PrintWriter out = res.getWriter();
           String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
           String fid=(String)session.getAttribute("username");
           String acid=(String)request.getParameter("acid");
           code="<form method='POST' action='./SubmitAttendance'><input type='hidden' name='acid'
value='"+acid+"'><label id='takelabel' for='timestamp'>Date and Time of Class</label><input
type='datetime-local' required id='timestamp' name='timestamp'>";
           try{
              Class.forName("com.mysql.jdbc.Driver");
              conn = DriverManager.getConnection(DB_URL,USER,PASS);
              stmt=conn.prepareStatement("select course.description from course,active_course where
course.course_id=active_course.course_id && active_course.acid=?");
              stmt.setString(1,acid);
              ResultSet rs=stmt.executeQuery();
              String coursename="";
              while(rs.next()){
                 coursename=rs.getString("description");
              }
              code1="<h2>"+acid+"-"+coursename+"</h2><h2>Enter Attendance</h2>";
              String query="{call proc_get_students(?)}";
              cstmt=conn.prepareCall(query);
              cstmt.setString(1,acid);
              rs = cstmt.executeQuery();
              String thead="<table><tr> <th>Register Number</th><th>Student
Name</th><th>Attendance(Mark if absent)</th></tr>";
              code=code+thead;
              while(rs.next()){
                 String stu_id=rs.getString("student_id");
                 String stu_name=rs.getString("name");
```

```
            code=code +"<tr> <td>"+stu_id+"</td>"+"<td>"+stu_name+"</td><td>"+"<input
type='checkbox' name='"+stu_id+"'>"+"</td><tr>";
          }
          code=code+"</table><input type='submit' id='takeenter' value='Enter'></form>";
          rs.close();
          cstmt.close();
          conn.close();
        }
        catch(SQLException se){
          se.printStackTrace();
        }
        catch(Exception e){
        e.printStackTrace();
        }
        finally{
          try{
            if(stmt!=null)
              stmt.close();
          }
          catch(SQLException se2){
          }
          try{
            if(conn!=null)
              conn.close();
          }
          catch(SQLException se){
            se.printStackTrace();
          }
        }
        out.print(docType +
        "<html>\n" +
          "<head><title>" + "Faculty" + "</title>\n" +
          "<LINK REL='stylesheet' href='./style.css' type='text/css'> "+
          "</head>\n"+
          "<body>\n" +
            code1+
            code+
          "</body>"+
        "</html>");

    }
  }
```

```
   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\GetActiveCourseIDS.javaim
port java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class GetActiveCourseIDS extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            stmt=conn.prepareStatement("select acid from active_course;");
```

```
            ResultSet rs=stmt.executeQuery();
            while(rs.next()){
               String cid=rs.getString("acid");
               out.print("<option value='"+cid+"'>"+cid+"</option>");
            }
            rs.close();
            stmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\GetCourseIDS.javaimport
java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
```

```
import java.util.*;
import java.sql.*;
import java.time.*;


public class GetCourseIDS extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            stmt=conn.prepareStatement("select course_id from course;");
            ResultSet rs=stmt.executeQuery();
            while(rs.next()){
               String cid=rs.getString("course_id");
               out.print("<option value='"+cid+"'>"+cid+"</option>");
            }
            rs.close();
            stmt.close();
            conn.close();
         }
         catch(SQLException se){
```

```
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }

      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentCourseAttendanceSer
vlet.javaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class StudentCourseAttendanceServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
```

```
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

       HttpSession session = request.getSession(false);
       Connection conn=null;
       PreparedStatement stmt=null;
       CallableStatement cstmt=null;
       String code="";
       String table="";
                    int total_sessions=0;
                    int total_absent=0;
                    String temp="";
       if(session == null){
          res.sendRedirect("./index.html");
       }
       else{
          res.setContentType("text/html") ;
          PrintWriter out = res.getWriter();
          String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
          out.println(docType +
          "<html>\n" +
             "<head>" +
             "<LINK REL='stylesheet' href='./style.css' type='text/css'> "+
             "</head>\n"+
             "<body>\n");
          String sid=(String)session.getAttribute("username");
                              String acid=request.getParameter("acid");
          try{
             Class.forName("com.mysql.jdbc.Driver");
             conn = DriverManager.getConnection(DB_URL,USER,PASS);
             stmt=conn.prepareStatement("select course.description from course,active_course where
course.course_id=active_course.course_id && active_course.acid=?");
             stmt.setString(1,acid);
             ResultSet  rs=stmt.executeQuery();
             String coursename="";
             while(rs.next()){
                coursename=rs.getString("description");
             }
             out.print("<h2>"+acid+"-"+coursename+"</h2>");
```

```
        String query="{call proc_student_attendance(?,?)}";
        cstmt=conn.prepareCall(query);
        cstmt.setString(1,sid);
                                cstmt.setString(2,acid);
        rs = cstmt.executeQuery();
        String thead="<table><tr><th>Session ID</th><th>Time
Stamp</th><th>Present/Absent</th></tr>";
                                table=thead;
        while(rs.next()){
                                        total_sessions=total_sessions+1;
                                        if(rs.getInt("absent")==1)
                                        {
                                                total_absent=total_absent+1;
                                                temp="Absent";
                                        }
                                        else
                                                temp="Present";
                                        java.sql.Timestamp ts= rs.getTimestamp("timestamp");

table=table+"<tr><td>"+rs.getInt("session_id")+"</td>"+"<td>"+ts.toString()+"</td>"+"<td>"+temp+"</
td></tr>";
        }
                                code="";
                                code="<p>Total classes: "+total_sessions+"</p>";
                                code=code+"<p>Total absent: "+total_absent+"</p>";
        int total_present=total_sessions-total_absent;
                                float percentage=((float)total_present/(float)total_sessions)*100;
                                code=code+"<p>Attendance Percentage: "+percentage+"</p>";
                                float
next_percentage=((float)total_present/((float)total_sessions+1))*100;
                                code=code+"<p>Attendance Percentage If You Miss Next Class:
"+next_percentage+"</p>";
                                int can_miss=0;
                                int need_present=0;
                                float calc=0;
                                int temp_sessions=total_sessions;
                                if(percentage<75.0)
                                {
                                        code=code+"<p>Classes that you can afford to miss:
"+can_miss+"</p>";
            need_present=3*total_sessions-4*total_present;
            need_present=need_present+1;
                                        code=code+"<p>Classes you need to attend to stay above
cut-off: "+need_present+"</p>";
```

```
                                        }
                                        else
                                        {
                                                can_miss=(-3*total_sessions+4*total_present)/3;
            can_miss=can_miss+1;
                                                code=code+"<p>Classes that you can afford to miss:
"+can_miss+"</p>";


                                        }
                                        out.print(table + code);
                                        out.print("</body></html>");
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }

      }
   }
```

public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException, IOException

```
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentDiscrepancySubmitS
ervlet.javaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class StudentDiscrepancySubmitServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String sid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            String query="{call proc_apply_discrepancy(?,?,?,?)}";
            cstmt=conn.prepareCall(query);
```

```
            cstmt.setString(1,sid);
                                    cstmt.setString(2,request.getParameter("acid"));
                                    int session_id=Integer.parseInt(request.getParameter("session_id"));
                                    cstmt.setInt(3,session_id);
                                    cstmt.setString(4,request.getParameter("justification"));
            cstmt.executeUpdate();

            cstmt.close();
            conn.close();
          }
          catch(SQLException se){
            se.printStackTrace();
          }
          catch(Exception e){
          e.printStackTrace();
          }
          finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
          }
          RequestDispatcher rd = request.getRequestDispatcher("./StudentDiscrepancyHome.html");
                                rd.forward(request, res);

       }
    }

    public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
    {
       doGet(request,res);
    }
```

```
}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentDiscrepancyViewSer
vlet.javaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class StudentDiscrepancyViewServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String sid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            String query="{call proc_get_student_discrepancy(?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,sid);
            ResultSet rs = cstmt.executeQuery();
            String thead="<table><tr><th>Course ID</th><th>Session
ID</th><th>Description</th><th>Status</th></tr>";
```

```
                                        out.println(thead);
            while(rs.next()){

code="<tr><td>"+rs.getString("acid")+"</td>"+"<td>"+rs.getInt("session_id")+"</td>"+"<td>"+rs.getStr
ing("justification")+"</td>"+"<td>"+rs.getString("status")+"</td></tr>";
                                                    out.println(code);
            }
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }


      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }
```

```
}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentHomePageServlet.jav
aimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;

public class StudentHomePageServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="<h1>Hello,"+(String)session.getAttribute("Name")+"</h1>";;
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         out.println(code);
         String sid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            String query="{call proc_student_course_list(?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,sid);
            ResultSet rs = cstmt.executeQuery();
            String thead="<table><tr><th>Enrolled CID</th><th>Course
Name</th><th>Percentage</th></tr>";
                                    out.println(thead);
```

```
            while(rs.next()){
               String acid=rs.getString("acid");
               code="<tr><td class='courseref'><a
href='StudentAttendance?acid="+acid+"'>"+acid+"</a></td>"+"<td>"+rs.getString("description")+"</td
>"+"<td>"+rs.getFloat("percentage")+"</td></tr>";
                                                  out.println(code);
            }
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }


      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }
```

```
}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentNotificationServlet.ja
vaimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class StudentNotificationServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String sid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
            String query="{call proc_get_notifications(?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,sid);
                                    String ulhead="<ul>";
            ResultSet rs = cstmt.executeQuery();
            ArrayList<String> acids=new ArrayList<String>();
```

```
            while(rs.next()){
               String acid=rs.getString("acid");
               String type=rs.getString("type");
                                                if((type.equals("percentage")) && !acids.contains(acid)){
                                                        ulhead=ulhead+"<li>Your attendance in "+acid+" has
fallen below the cut-off.</li>";
                  acids.add(acid);
               }
                                                else if(type.equals("absence"))
                                                        ulhead=ulhead+"<li>Your were marked absent in
"+acid+" for the session "+rs.getInt("session_id")+".</li>";
            }
                                        ulhead=ulhead+"</ul>";
                                        out.println(ulhead);
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }


      }
   }
```

```
   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
   }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentODSubmitServlet.jav
aimport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class StudentODSubmitServlet extends HttpServlet
{

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
   static final String DB_URL = "jdbc:mysql://localhost/AMS";
   static final String USER = "root";
   static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

      HttpSession session = request.getSession(false);
      Connection conn=null;
      PreparedStatement stmt=null;
      CallableStatement cstmt=null;
      String code="";
      if(session == null){
         res.sendRedirect("./index.html");
      }
      else{
         res.setContentType("text/html") ;
         PrintWriter out = res.getWriter();
         String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
         String sid=(String)session.getAttribute("username");
         try{
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(DB_URL,USER,PASS);
```

```
            String query="{call proc_apply_od(?,?,?,?,?)}";
            cstmt=conn.prepareCall(query);
            cstmt.setString(1,sid);
                                    cstmt.setString(2,request.getParameter("acid"));
                                    int session_id=Integer.parseInt(request.getParameter("session_id"));
                                    cstmt.setInt(3,session_id);
                                    cstmt.setString(4,request.getParameter("justification"));
                                    cstmt.setString(5,request.getParameter("proof"));
            cstmt.executeUpdate();

            cstmt.close();
            conn.close();
          }
          catch(SQLException se){
            se.printStackTrace();
          }
          catch(Exception e){
          e.printStackTrace();
          }
          finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
          }
          RequestDispatcher rd = request.getRequestDispatcher("./StudentODHome.html");
                                rd.forward(request, res);

       }
    }

    public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
    {
       doGet(request,res);
```

```
    }

}C:\Users\aarth\Desktop\College\tomcat8\webapps\AMS\WEB-INF\classes\StudentODViewServlet.javai
mport java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import java.sql.*;
import java.time.*;


public class StudentODViewServlet extends HttpServlet
{

    static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/AMS";
    static final String USER = "root";
    static final String PASS = "aarthi00*";


  public void doGet(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
  {

        HttpSession session = request.getSession(false);
        Connection conn=null;
        PreparedStatement stmt=null;
        CallableStatement cstmt=null;
        String code="";
        if(session == null){
            res.sendRedirect("./index.html");
        }
        else{
            res.setContentType("text/html") ;
            PrintWriter out = res.getWriter();
            String docType ="<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";
            String sid=(String)session.getAttribute("username");
            try{
                Class.forName("com.mysql.jdbc.Driver");
                conn = DriverManager.getConnection(DB_URL,USER,PASS);
                String query="{call proc_get_student_od(?)}";
                cstmt=conn.prepareCall(query);
                cstmt.setString(1,sid);
                ResultSet rs = cstmt.executeQuery();
```

```
            String thead="<table><tr><th>Course ID</th><th>Session
ID</th><th>Justification</th><th>Proof</th><th>Status</th></tr>";
                                        out.println(thead);
            while(rs.next()){

code="<tr><td>"+rs.getString("acid")+"</td>"+"<td>"+rs.getInt("session_id")+"</td>"+"<td>"+rs.getStr
ing("justification")+"</td>"+"<td>"+rs.getString("proof")+"</td>"+"<td>"+rs.getString("status")+"</td>
</tr>";
                                                    out.println(code);
            }
            rs.close();
            cstmt.close();
            conn.close();
         }
         catch(SQLException se){
            se.printStackTrace();
         }
         catch(Exception e){
         e.printStackTrace();
         }
         finally{
            try{
               if(stmt!=null)
                  stmt.close();
            }
            catch(SQLException se2){
            }
            try{
               if(conn!=null)
                  conn.close();
            }
            catch(SQLException se){
               se.printStackTrace();
            }
         }


      }
   }

   public void doPost(HttpServletRequest request , HttpServletResponse res) throws ServletException,
IOException
   {
      doGet(request,res);
```

}

Database:

Schema diagram of the tables:

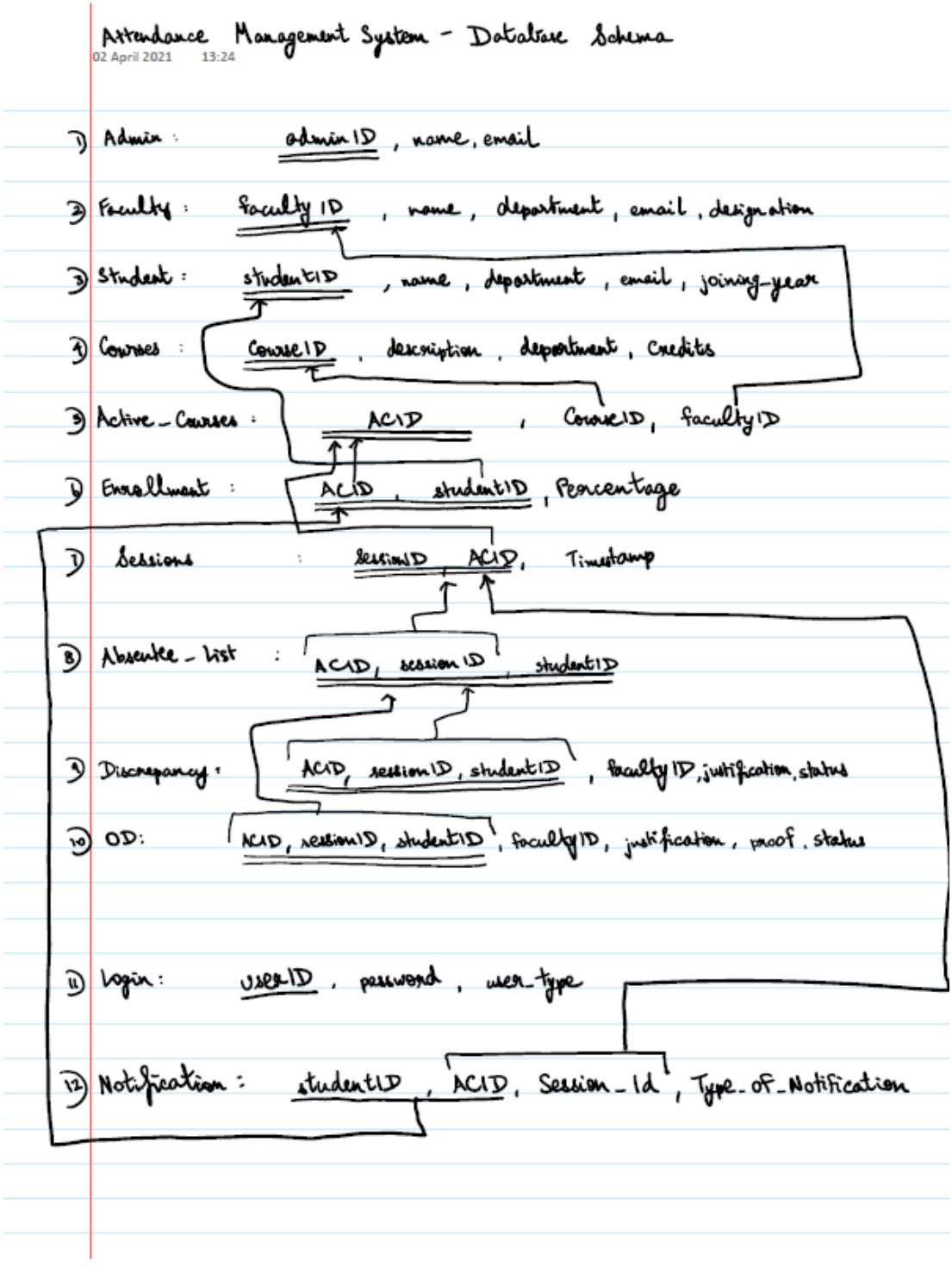


Table creation statements:

```
create database AMS;
use AMS;

drop procedure if exists proc_get_notifications;
drop procedure if exists proc_remove_notifications;
drop procedure if exists proc_acid_percentage;
drop procedure if exists proc_get_students;
drop procedure if exists proc_student_percentage;
drop procedure if exists proc_student_course_list;
drop procedure if exists proc_get_faculty_discrepancy ;
drop procedure if exists proc_get_faculty_od ;
drop procedure if exists proc_get_student_od ;
drop procedure if exists proc_get_student_discrepancy ;
drop procedure if exists proc_apply_discrepancy;
drop procedure if exists proc_apply_OD;
drop procedure if exists proc_process_discrepancy;
drop procedure if exists proc_process_OD;
drop procedure if exists proc_new_session;
DROP PROCEDURE IF EXISTS proc_remove_absentee;
DROP PROCEDURE IF EXISTS proc_add_absentee;
DROP PROCEDURE IF EXISTS proc_student_info;
DROP PROCEDURE IF EXISTS proc_student_attendance;
DROP PROCEDURE  IF EXISTS  proc_faculty_course_list;
DROP PROCEDURE  IF EXISTS  proc_login;
drop table if exists notification;
drop table if exists login;
drop table if exists admin;
drop table if exists od;
drop table if exists discrepancy;
drop table if exists absentee_list;
drop table if exists sessions;
drop table if exists enrollment;
drop table if exists active_course;
drop table if exists faculty;
drop table if exists course;
drop table if exists student;



create table IF NOT EXISTS student(student_id varchar(10) not null,
name varchar(50) not null,
```

```
department varchar(5) not null CHECK (department IN
('CSE','IT','ECE','EEE','MECH','BME','CIV','CHEM')),
email varchar(50) not null,
joining_year year,
PRIMARY KEY ( student_id )
);


create table IF NOT EXISTS faculty(faculty_id varchar(10) not null,
name varchar(50) not null,
department varchar(5) not null CHECK (department IN
('CSE','IT','ECE','EEE','MECH','BME','CIV','CHEM')),
email varchar(50) not null,
designation varchar(30) not null,
PRIMARY KEY ( faculty_id )
);


create table IF NOT EXISTS course(course_id varchar(10) not null,
description varchar(50) not null,
department varchar(5) not null CHECK (department IN
('CSE','IT','ECE','EEE','MECH','BME','CIV','CHEM')),
credits int not null,
PRIMARY KEY ( course_id )
);


create table IF NOT EXISTS active_course(acid varchar(20) not null,
course_id varchar(10) not null,
faculty_id varchar(10) not null,
PRIMARY KEY (acid),
CONSTRAINT FK_CID FOREIGN KEY (course_id) REFERENCES course(course_id) ON DELETE
CASCADE,
CONSTRAINT FK_FID FOREIGN KEY (faculty_id) REFERENCES faculty(faculty_id)  ON DELETE
CASCADE
);

create table IF NOT EXISTS enrollment(acid varchar(20) not null,
student_id varchar(10) not null,
percentage float not null,
PRIMARY KEY (acid,student_id),
CONSTRAINT FK_ACID FOREIGN KEY (acid) REFERENCES active_course(acid)  ON DELETE
CASCADE,
```

```
CONSTRAINT FK_SID FOREIGN KEY (student_id) REFERENCES student(student_id)  ON DELETE
CASCADE
);

create table IF NOT EXISTS sessions(session_id INT not null,
acid varchar(20) not null,
timestamp TIMESTAMP,
PRIMARY KEY(session_id,acid),
CONSTRAINT FK1_ACID FOREIGN KEY (acid) REFERENCES active_course(acid) ON DELETE
CASCADE);

create table IF NOT EXISTS absentee_list(session_id INT not null,
acid varchar(20) not null,
student_id varchar(10) not null,
PRIMARY KEY(session_id,acid,student_id),
CONSTRAINT FK1_SID FOREIGN KEY (student_id) REFERENCES student(student_id)  ON
DELETE CASCADE,
CONSTRAINT FK_SESS FOREIGN KEY (session_id,acid) REFERENCES sessions(session_id,acid)
ON DELETE CASCADE,
CONSTRAINT FK_ENR FOREIGN KEY (student_id,acid) REFERENCES enrollment(student_id,acid)
ON DELETE CASCADE
);

create table IF NOT EXISTS discrepancy(session_id INT not null,
acid varchar(20) not null,
student_id varchar(10) not null,
faculty_id varchar(10) not null,
justification varchar(100) not null,
status varchar(15) not null CHECK (status IN ('applied','approved','denied')),
PRIMARY KEY(session_id,acid,student_id),
CONSTRAINT FK_ABS FOREIGN KEY (session_id,acid,student_id) REFERENCES
absentee_list(session_id,acid,student_id)  ON DELETE CASCADE,
CONSTRAINT FK1_FID FOREIGN KEY (faculty_id) REFERENCES faculty(faculty_id)  ON
DELETE CASCADE
);

create table IF NOT EXISTS od(session_id INT not null,
acid varchar(20) not null,
student_id varchar(10) not null,
faculty_id varchar(10) not null,
justification varchar(100) not null,
proof varchar(25) not null,
status varchar(15) not null CHECK (status IN ('applied','approved','denied')),
PRIMARY KEY(session_id,acid,student_id),
```

```
CONSTRAINT FK1_ABS FOREIGN KEY (session_id,acid,student_id) REFERENCES
absentee_list(session_id,acid,student_id)  ON DELETE CASCADE,
CONSTRAINT FK2_FID FOREIGN KEY (faculty_id) REFERENCES faculty(faculty_id)  ON
DELETE CASCADE
);

create table IF NOT EXISTS admin(admin_id varchar(10) not null,
name varchar(20) not null,
email varchar(25) not null,
PRIMARY KEY(admin_id)
);

create table IF NOT EXISTS login(user_id varchar(10) not null,
password varchar(20) not null,
user_type varchar(10) not null CHECK (user_type IN ('admin','student','faculty')),
PRIMARY KEY(user_id)
);

create table IF NOT EXISTS notification(student_id varchar(10) not null,
acid varchar(20) not null,
session_id INT not null,
type varchar(20) not null CHECK (type IN ('percentage','absence')),
CONSTRAINT FK3_ENR FOREIGN KEY (student_id,acid) REFERENCES
enrollment(student_id,acid)  ON DELETE CASCADE,
CONSTRAINT FK1_SESS FOREIGN KEY (session_id,acid) REFERENCES sessions(session_id,acid)
ON DELETE CASCADE
);
```

Procedure code

```
DELIMITER //
CREATE PROCEDURE proc_login (IN u_id varchar(10),IN pswd varchar(20),OUT user_type
varchar(10))
BEGIN
   SELECT login.user_type into user_type FROM login WHERE user_id=u_id && password=pswd;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_faculty_course_list (IN f_id varchar(10))
BEGIN
   SELECT active_course.acid,active_course.course_id,course.description FROM active_course,course
WHERE active_course.faculty_id=f_id && active_course.course_id=course.course_id;
END //
```

```
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_course_info (IN c_id varchar(10))
BEGIN
   SELECT * FROM course WHERE course_id=c_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_student_attendance (IN s_id varchar(10),IN aid varchar(20))
BEGIN
   select sessions.session_id as session_id,timestamp,(select count(*) from absentee_list where
session_id=sessions.session_id && acid=aid && student_id=s_id) as absent from sessions where
sessions.acid=aid order by sessions.session_id desc;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_student_info (IN s_id varchar(10))
BEGIN
   SELECT * FROM student WHERE student_id=s_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_new_session (IN aid varchar(20),IN time timestamp,OUT sess_id int)
BEGIN
   select coalesce(max(session_id)+1,1) into sess_id from sessions where acid=aid;
   insert into sessions values (sess_id,aid,time);
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_add_absentee (IN s_id varchar(10),IN aid varchar(20),IN sess_id int)
BEGIN
   insert into absentee_list values (sess_id,aid,s_id);
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_remove_absentee (IN s_id varchar(10),IN aid varchar(20),IN sess_id int)
BEGIN
   SET foreign_key_checks = 0;
```

```
   delete from absentee_list where session_id=sess_id && acid=aid && student_id=s_id;
   SET foreign_key_checks = 1;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_process_OD (IN s_id varchar(10),IN aid varchar(20),IN sess_id int,IN stat
varchar(15))
BEGIN
   declare same INT default 1;
   update od set status=stat where session_id=sess_id && acid=aid && student_id=s_id;
   if strcmp(stat,'approved')=0 then
      call proc_remove_absentee(s_id,aid,sess_id);
   end if;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_process_discrepancy (IN s_id varchar(10),IN aid varchar(20),IN sess_id
int,IN stat varchar(15))
BEGIN
   declare same INT default 1;
   update discrepancy set status=stat where session_id=sess_id && acid=aid && student_id=s_id;
   if strcmp(stat,'approved')=0 then
      call proc_remove_absentee(s_id,aid,sess_id);
   end if;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_apply_discrepancy (IN s_id varchar(10),IN aid varchar(20),IN sess_id
int,IN just varchar(100))
BEGIN
   declare fid varchar(10);
   select faculty_id into fid from active_course where acid=aid;
   insert into discrepancy values(sess_id,aid,s_id,fid,just,'applied');
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_apply_od (IN s_id varchar(10),IN aid varchar(20),IN sess_id int,IN just
varchar(100),IN prf varchar(25))
BEGIN
   declare fid varchar(10);
```

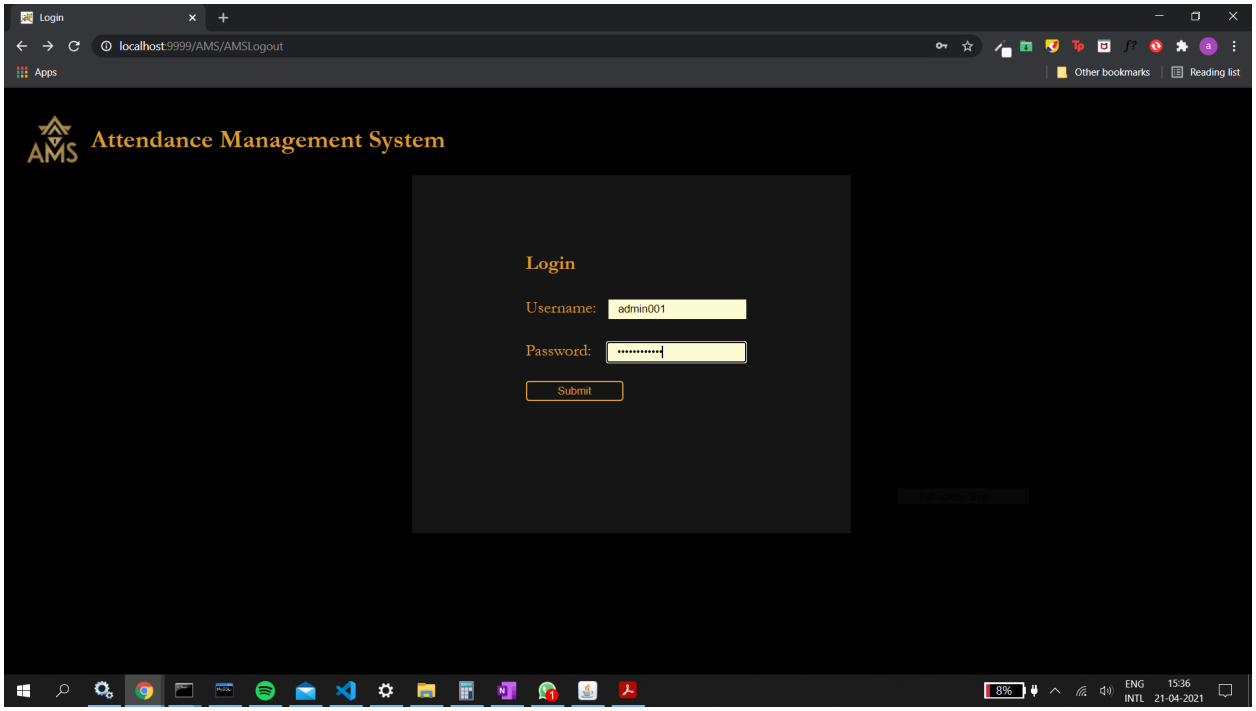
```
   select faculty_id into fid from active_course where acid=aid;
   insert into od values(sess_id,aid,s_id,fid,just,prf,'applied');
END //
DELIMITER ;


DELIMITER //
CREATE PROCEDURE proc_get_faculty_od (IN f_id varchar(10))
BEGIN
   select * from od where faculty_id=f_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_get_student_od (IN s_id varchar(10))
BEGIN
   select * from od where student_id=s_id;
END //
DELIMITER ;


DELIMITER //
CREATE PROCEDURE proc_get_faculty_discrepancy (IN f_id varchar(10))
BEGIN
   select * from discrepancy where faculty_id=f_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_get_student_discrepancy (IN s_id varchar(10))
BEGIN
   select * from discrepancy where student_id=s_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_student_course_list (IN s_id varchar(10))
BEGIN
   SELECT enrollment.acid,enrollment.percentage,active_course.course_id,course.description from
enrollment,active_course,course  where enrollment.student_id=s_id &&
enrollment.acid=active_course.acid && active_course.course_id=course.course_id;
END //
DELIMITER ;
```

```
DELIMITER //
CREATE PROCEDURE proc_student_percentage (IN s_id varchar(10),IN aid varchar(20),OUT tot_sess
INT, OUT noabsent INT)
BEGIN
   SELECT max(sessions.session_id) ,sum((select count(*) from absentee_list where
session_id=sessions.session_id && acid=aid && student_id=s_id)) into tot_sess,noabsent from sessions
where sessions.acid=aid;
   update enrollment set percentage=((tot_sess-noabsent)/tot_sess)*100 where student_id=s_id &&
acid=aid;
END //
DELIMITER ;


DELIMITER //
CREATE PROCEDURE proc_get_students (IN aid varchar(20))
BEGIN
   SELECT enrollment.student_id,student.name,enrollment.percentage from enrollment,student where
enrollment.acid=aid && student.student_id=enrollment.student_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_acid_percentage (IN aid varchar(20))
BEGIN
   DECLARE b,tot,noa INT;
   DECLARE a varchar(10);
   DECLARE cur_1 CURSOR FOR
   SELECT student_id from enrollment where acid=aid;
   DECLARE CONTINUE HANDLER FOR NOT FOUND SET b = 1;
   OPEN cur_1;
         update_perc: LOOP
                  FETCH cur_1 INTO a;
      IF b = 1 THEN
                        LEAVE update_perc;
      END IF;
                  call proc_student_percentage(a,aid,@tot,@noa);
         END LOOP update_perc;
         CLOSE cur_1;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_remove_notifications (IN s_id varchar(10))
```

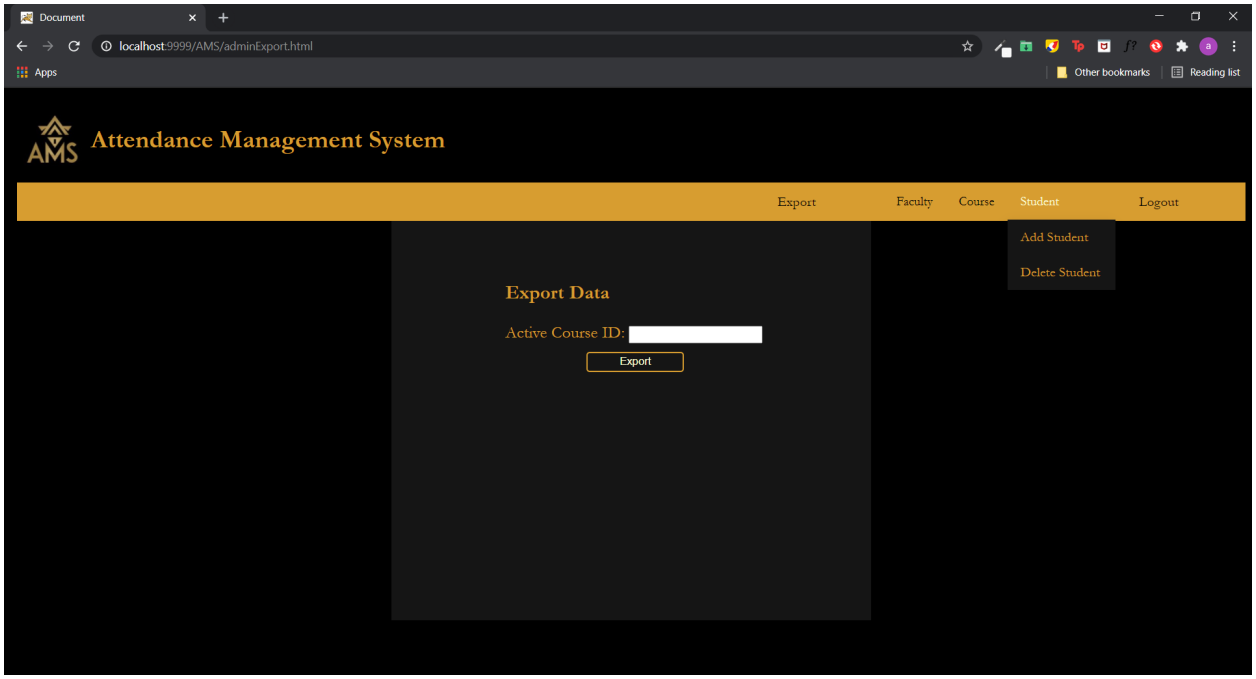
```
BEGIN
   delete from notification where student_id=s_id;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE proc_get_notifications (IN s_id varchar(10))
BEGIN
   select * from notification where student_id=s_id;
END //
DELIMITER ;


DELIMITER //
CREATE TRIGGER trigger_absence_notification
   AFTER INSERT
   ON absentee_list FOR EACH ROW
   BEGIN
   insert into notification values(new.student_id,new.acid,new.session_id,'absence');
END //
DELIMITER ;

DELIMITER //
CREATE TRIGGER trigger_percentage_notification
   AFTER UPDATE
   ON enrollment FOR EACH ROW
   BEGIN
   DECLARE s_id INT;
   select max(sessions.session_id) into s_id from sessions where acid=new.acid;
   IF new.percentage<75 THEN
      insert into notification values(new.student_id,new.acid,s_id,'percentage');
   END IF;
END //
DELIMITER ;

DELIMITER //
CREATE TRIGGER trigger_percentage_update
   AFTER DELETE
   ON absentee_list FOR EACH ROW
   BEGIN
   DECLARE tot,noa INT;
   call proc_student_percentage(old.student_id,old.acid,@tot,@noa);
END //
DELIMITER ;
```

Output : ADMIN

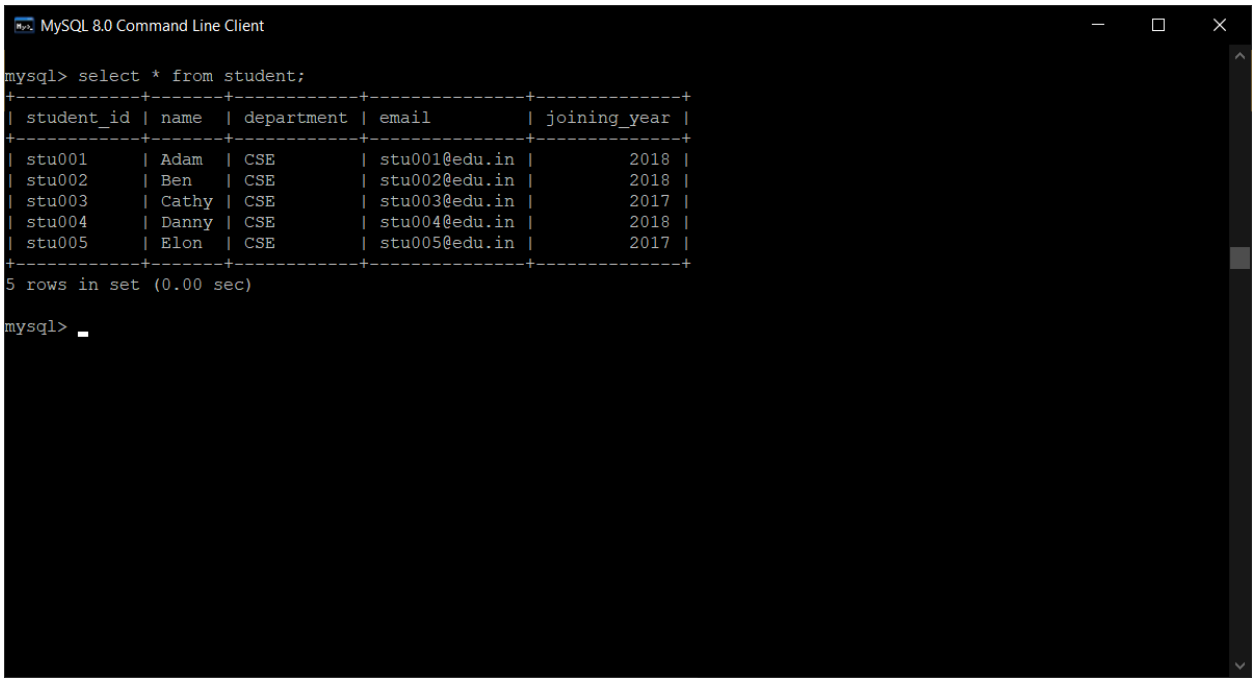
Admin Login:



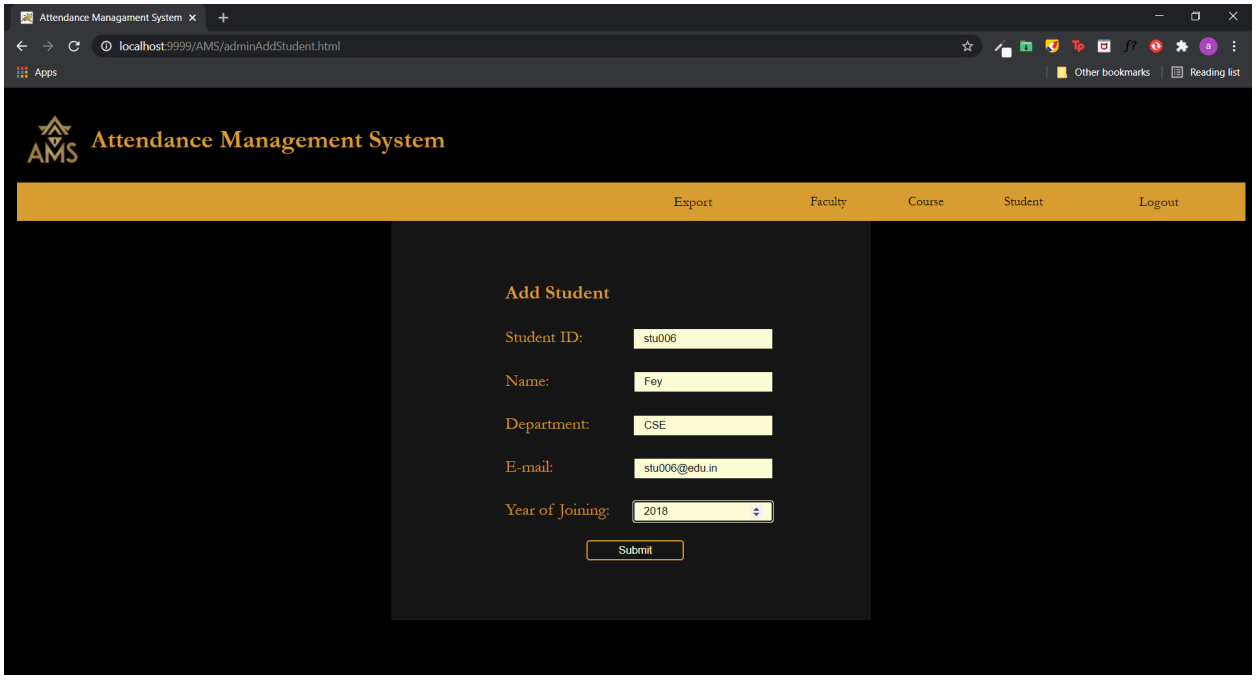
Admin Homepage - Student Hover:



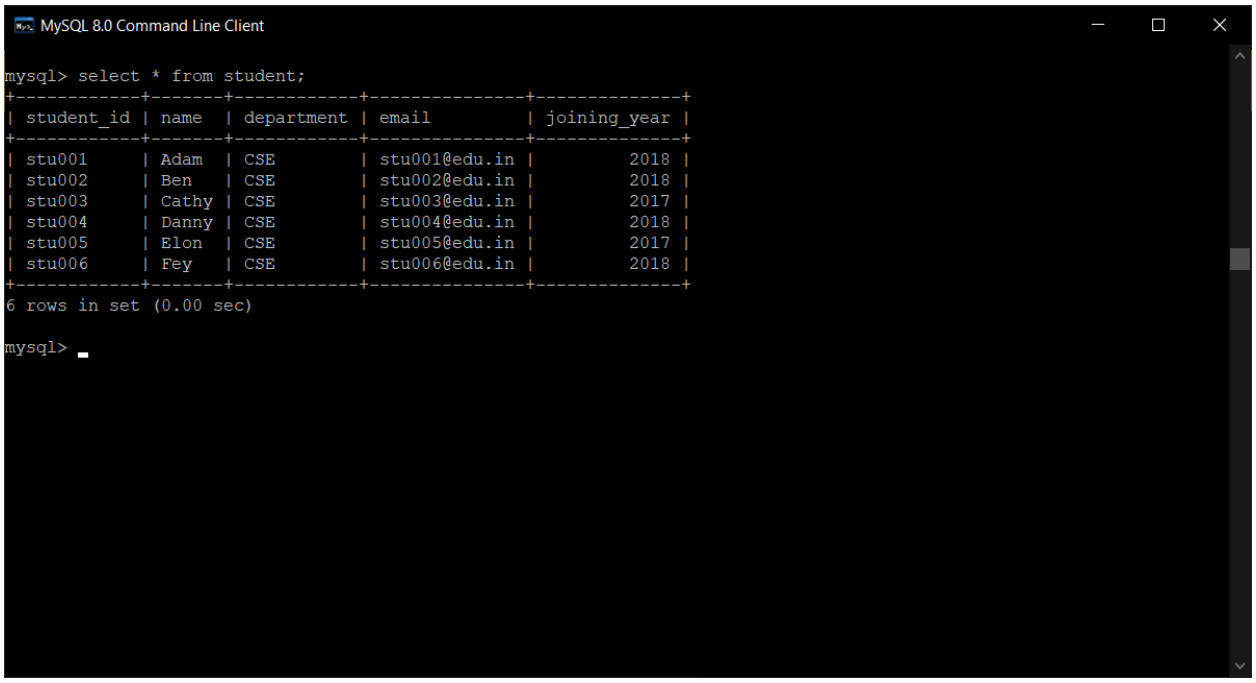
Before adding student:



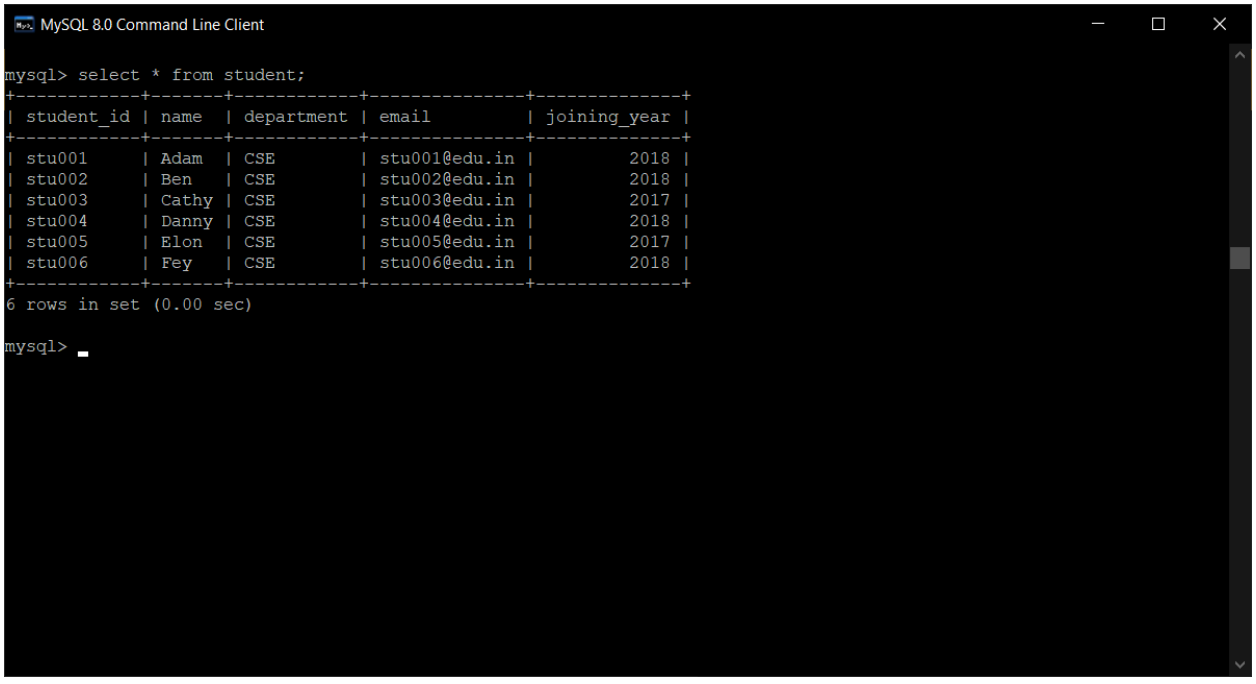
Add Student:



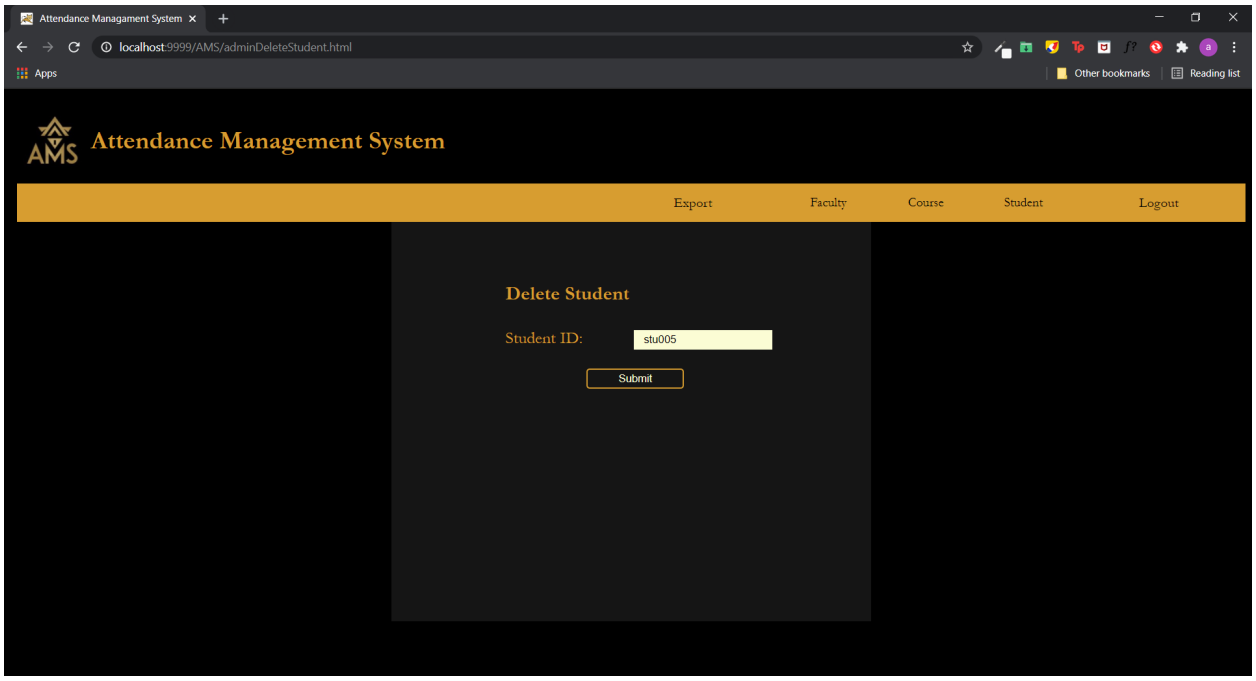
After Adding Student:



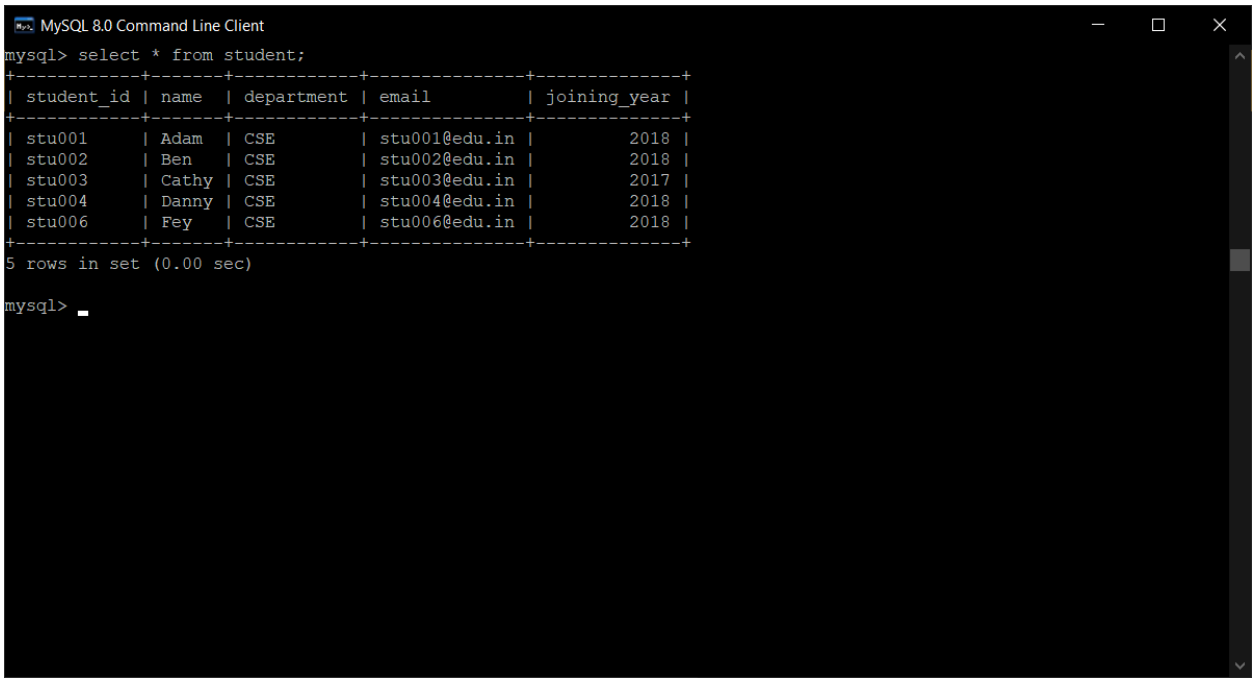
Before Deleting Student:



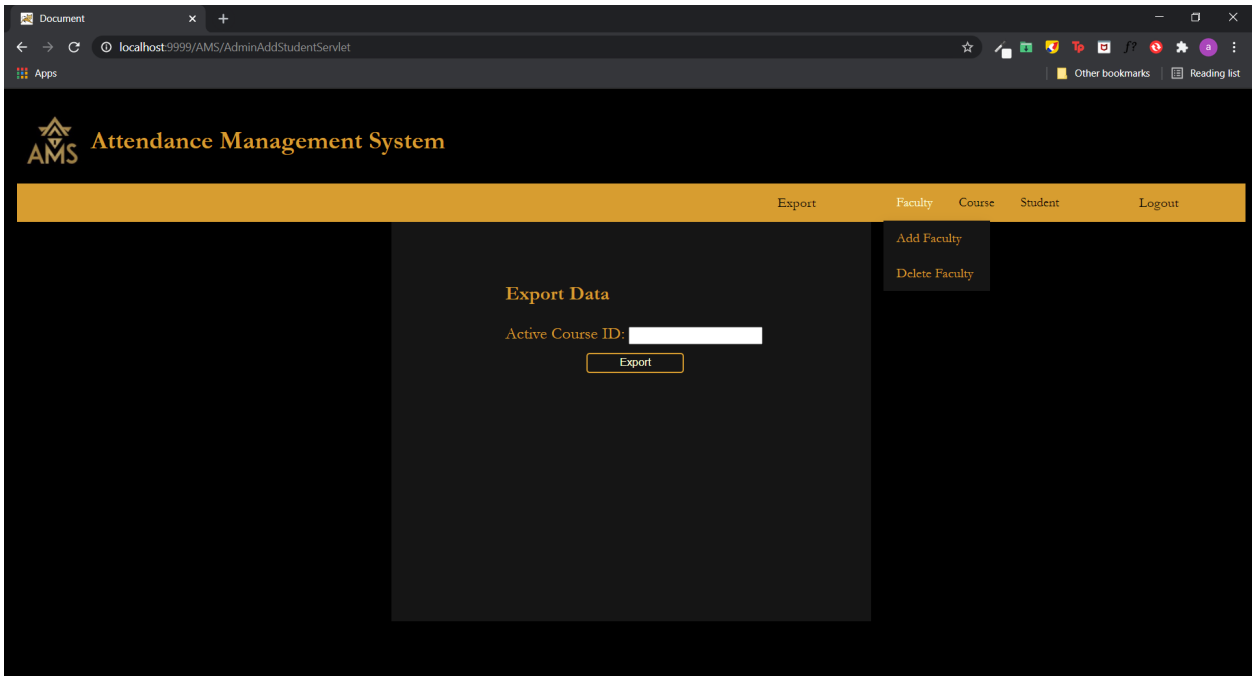
Delete Student:



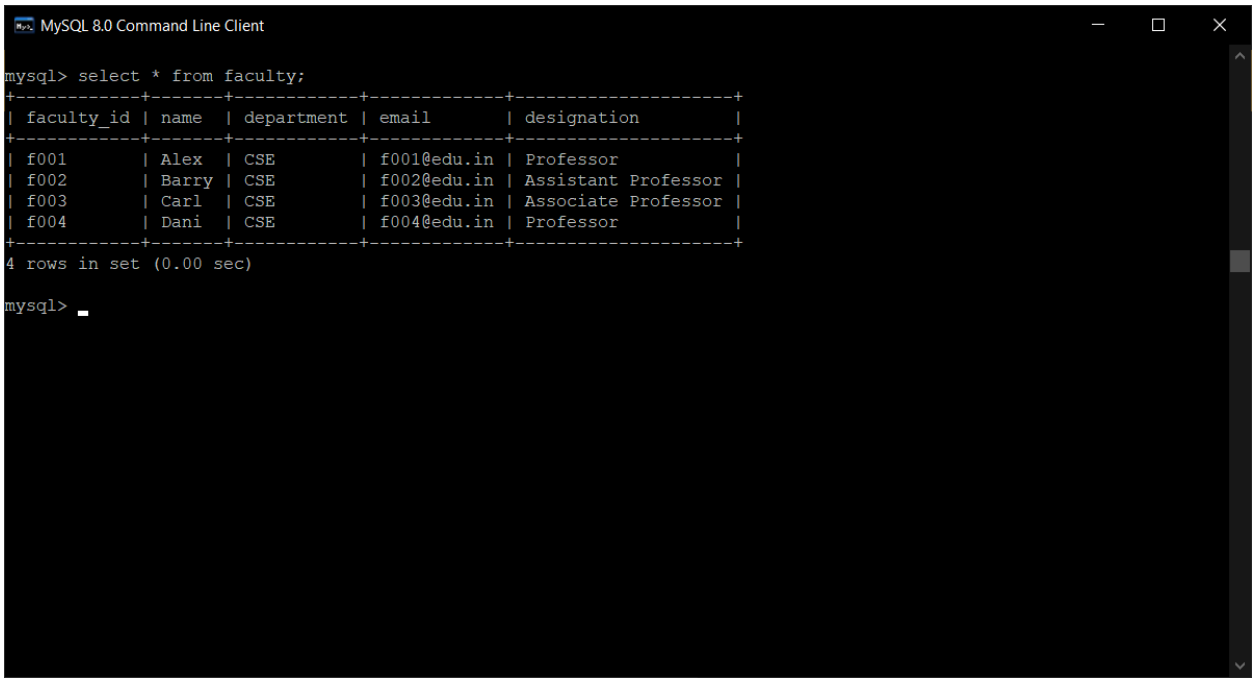
After Deleting Student:



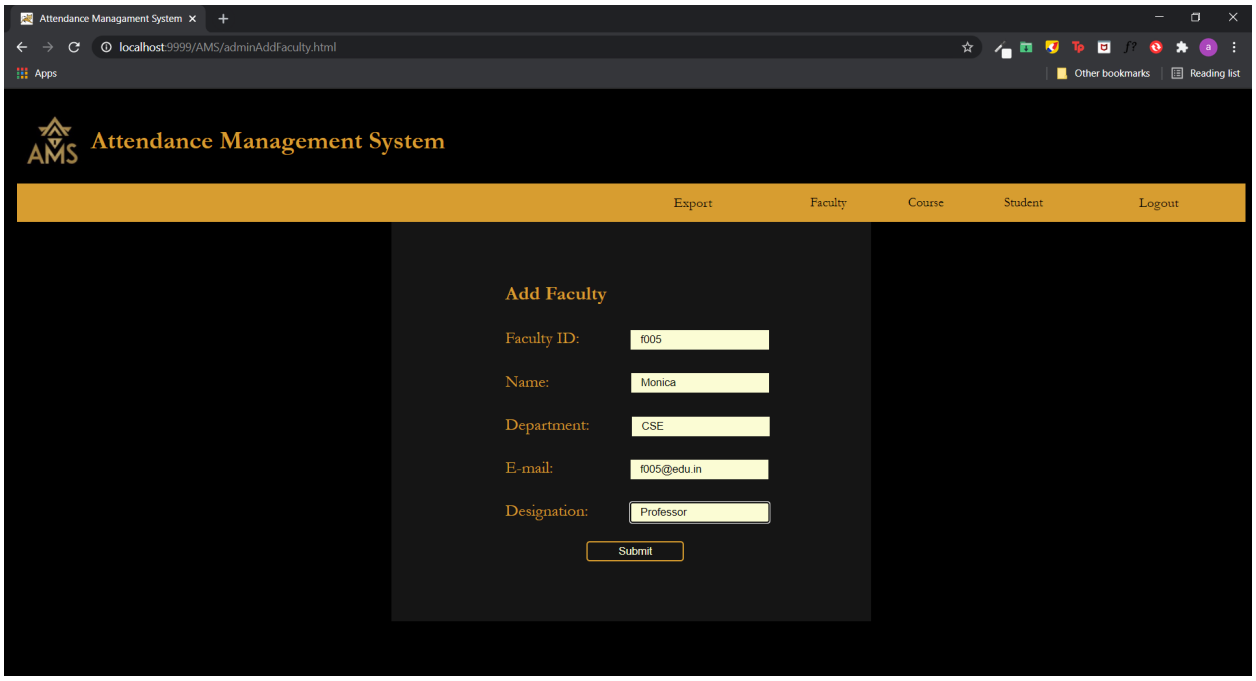
Admin Homepage - Faculty Hover:



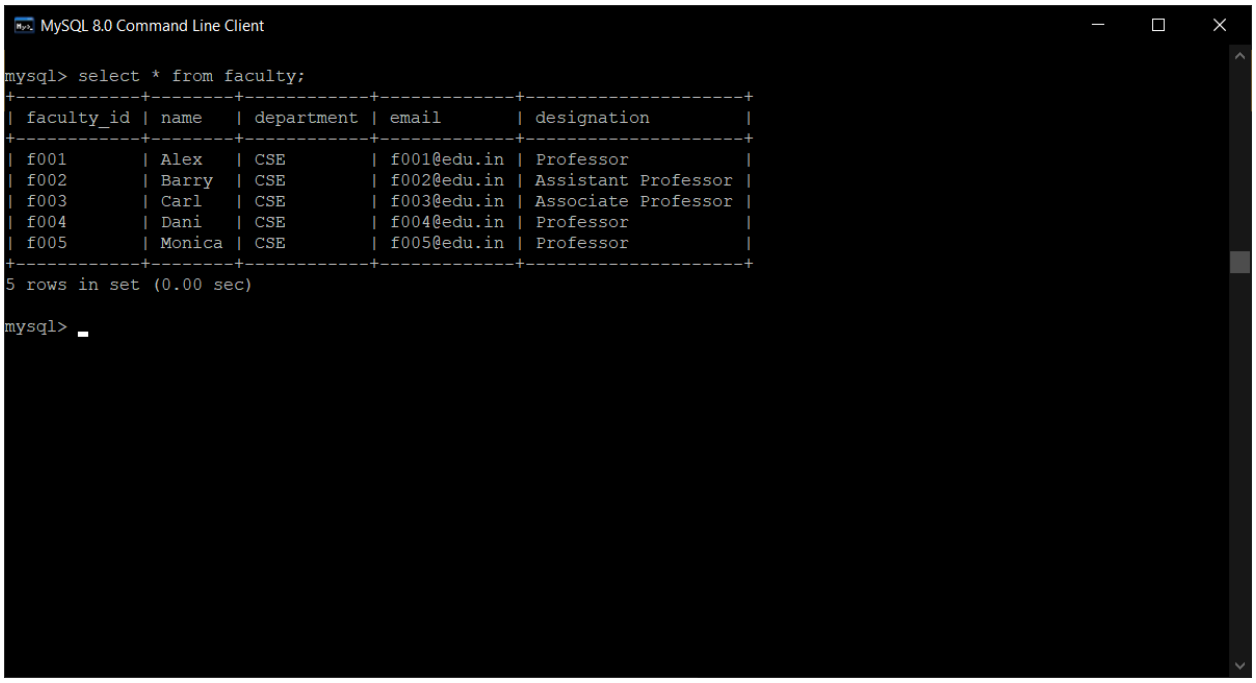
Before Adding Faculty:



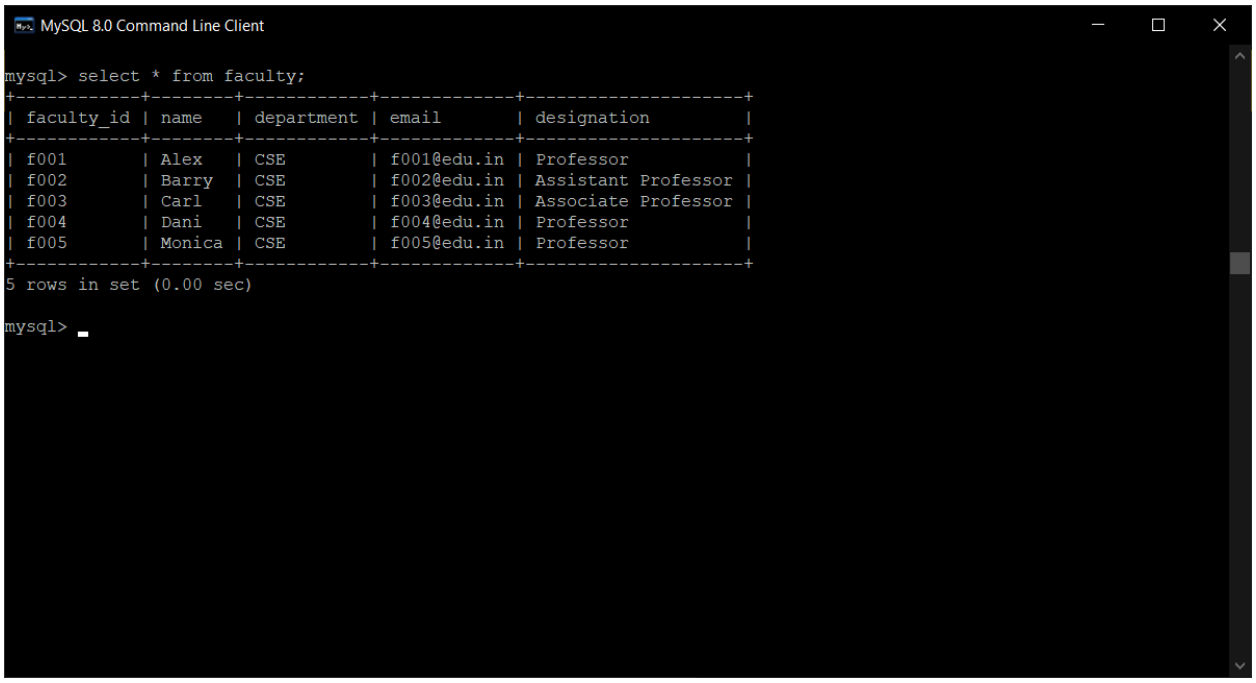
Add Faculty:



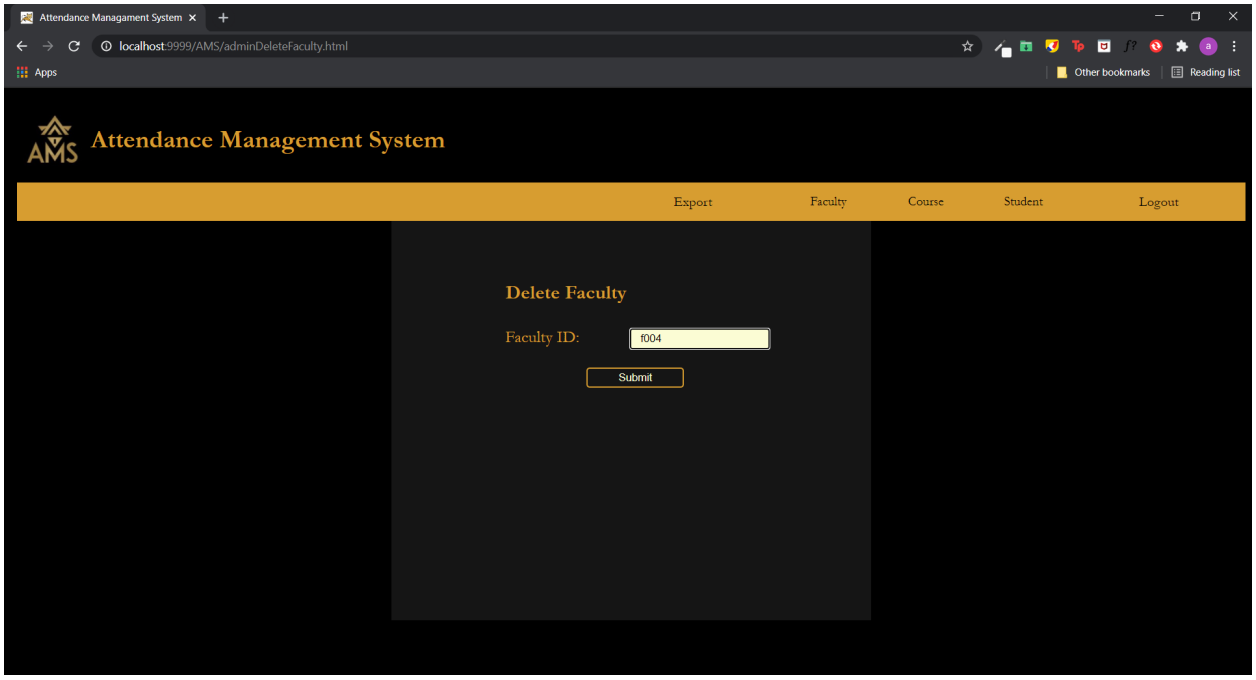
After Adding Faculty:



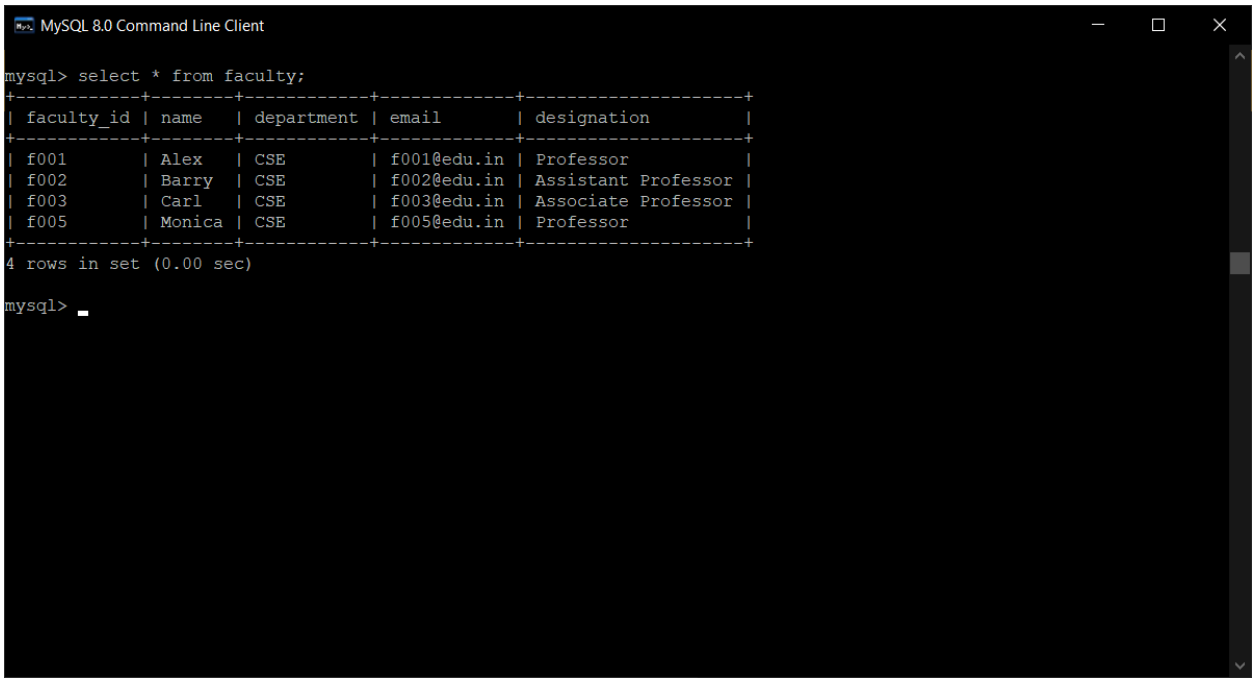
Before Deleting Faculty:



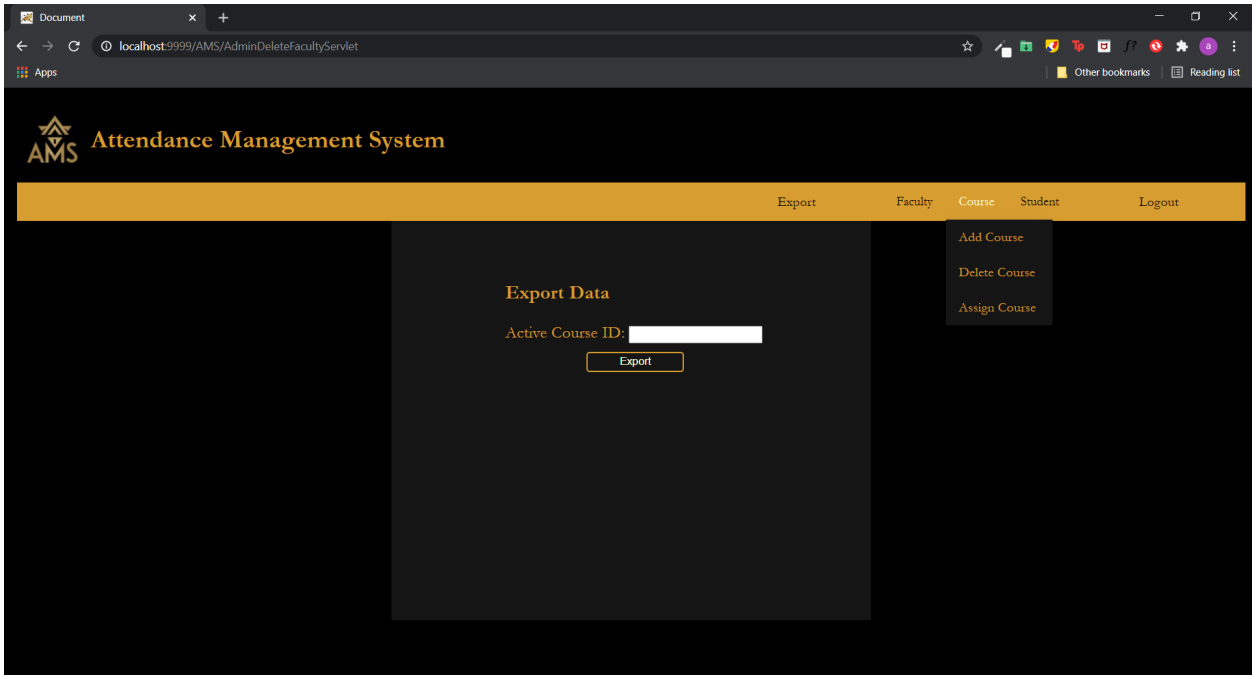
Delete Faculty:



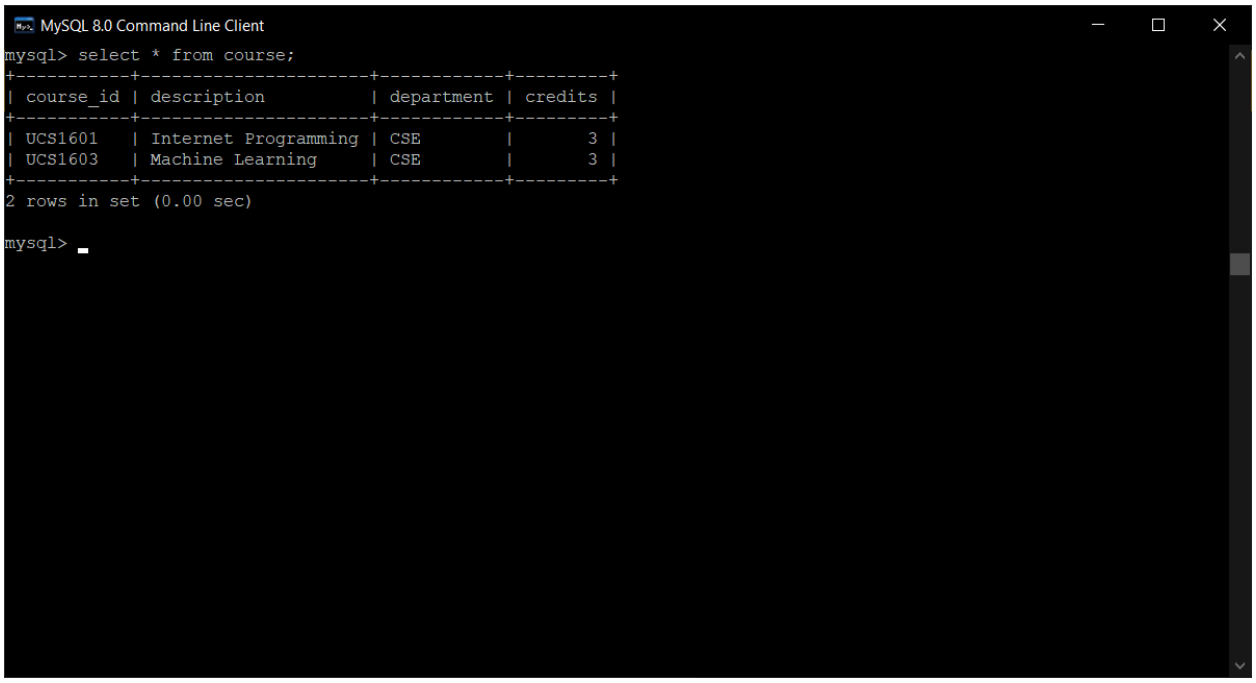
After Deleting Faculty:



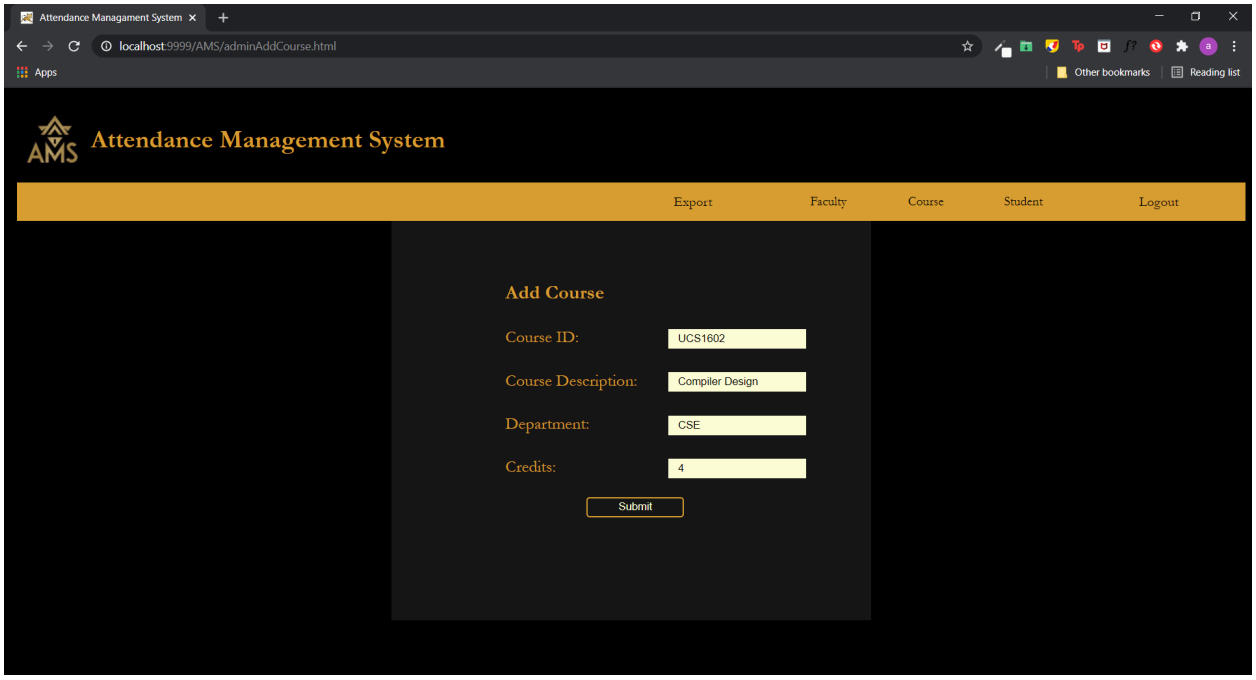
Admin Homepage - Course Hover:



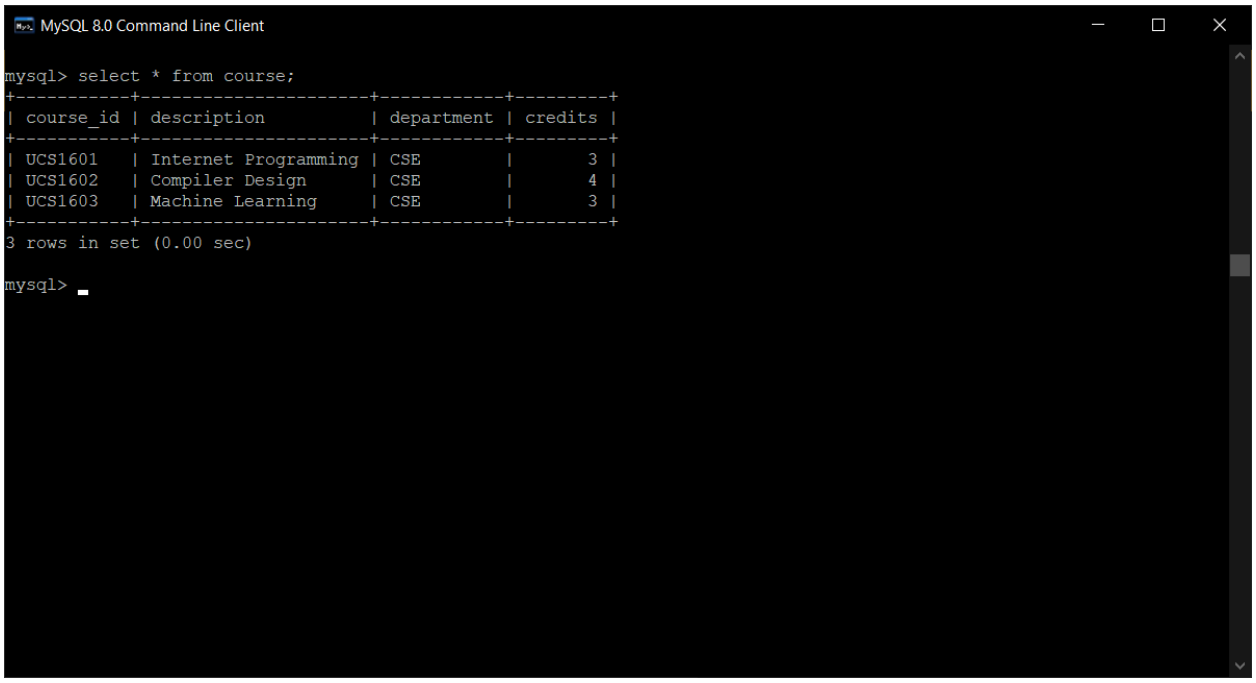
Before Adding Course:



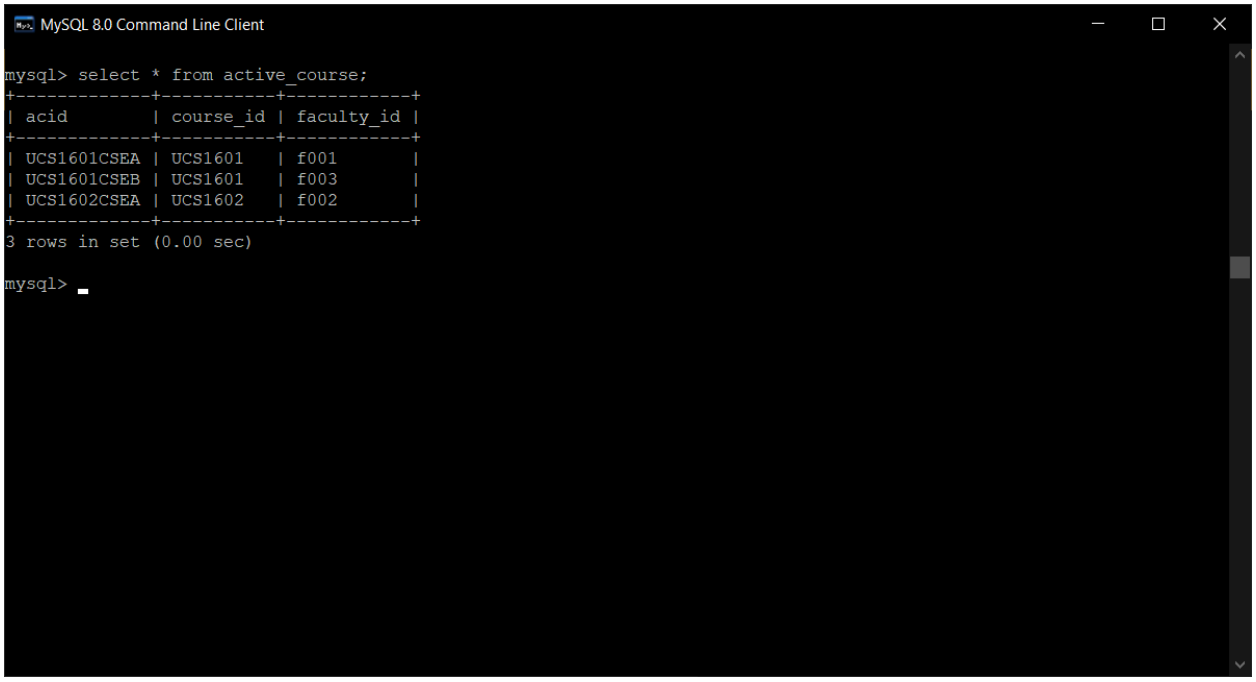
Add Course:



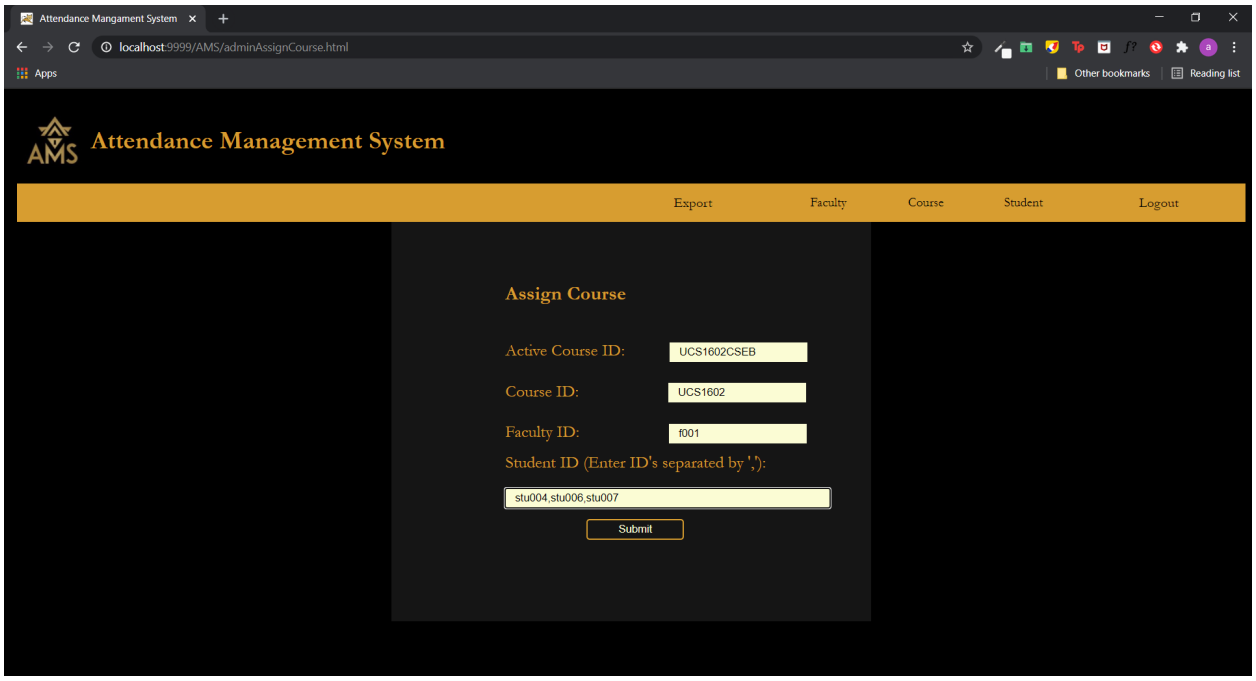
After Adding Course:



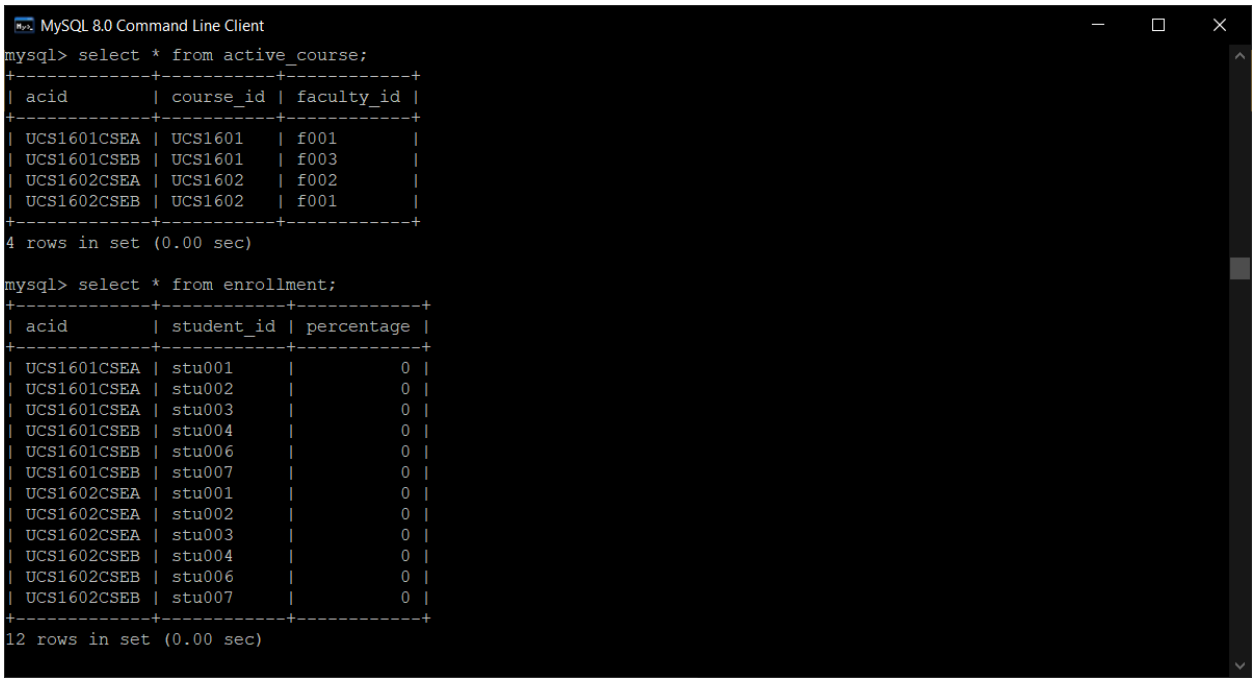
Before Assigning Course:



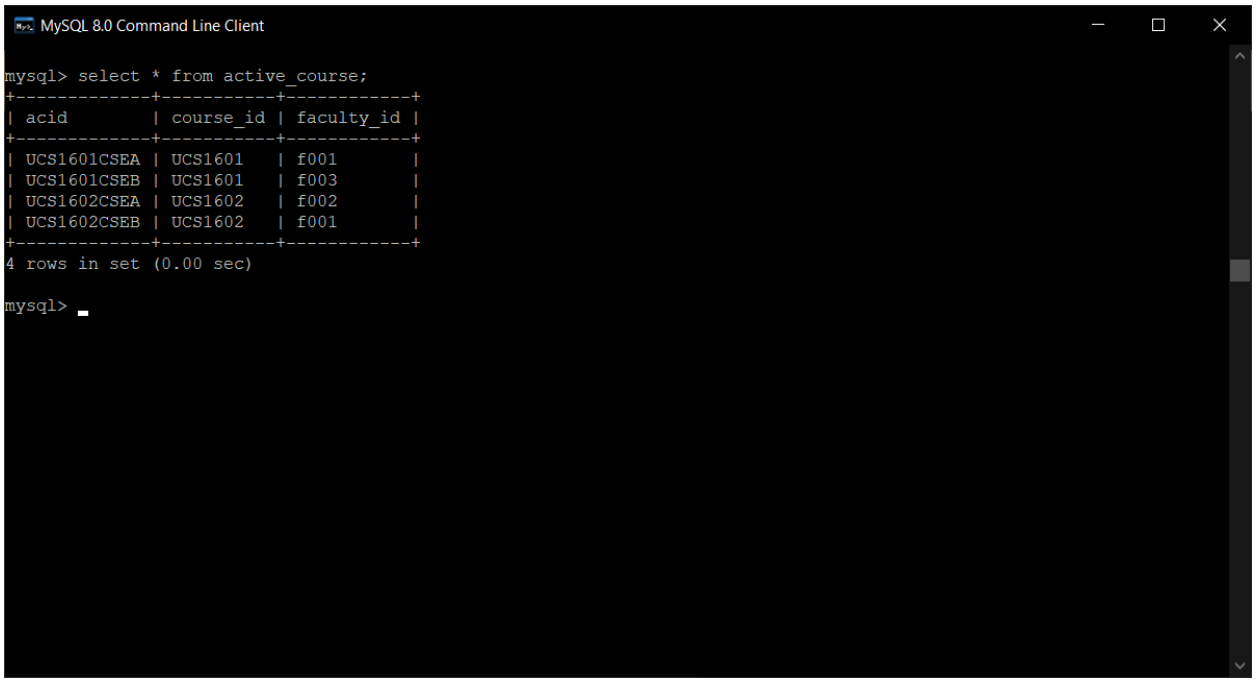
Assign Course:



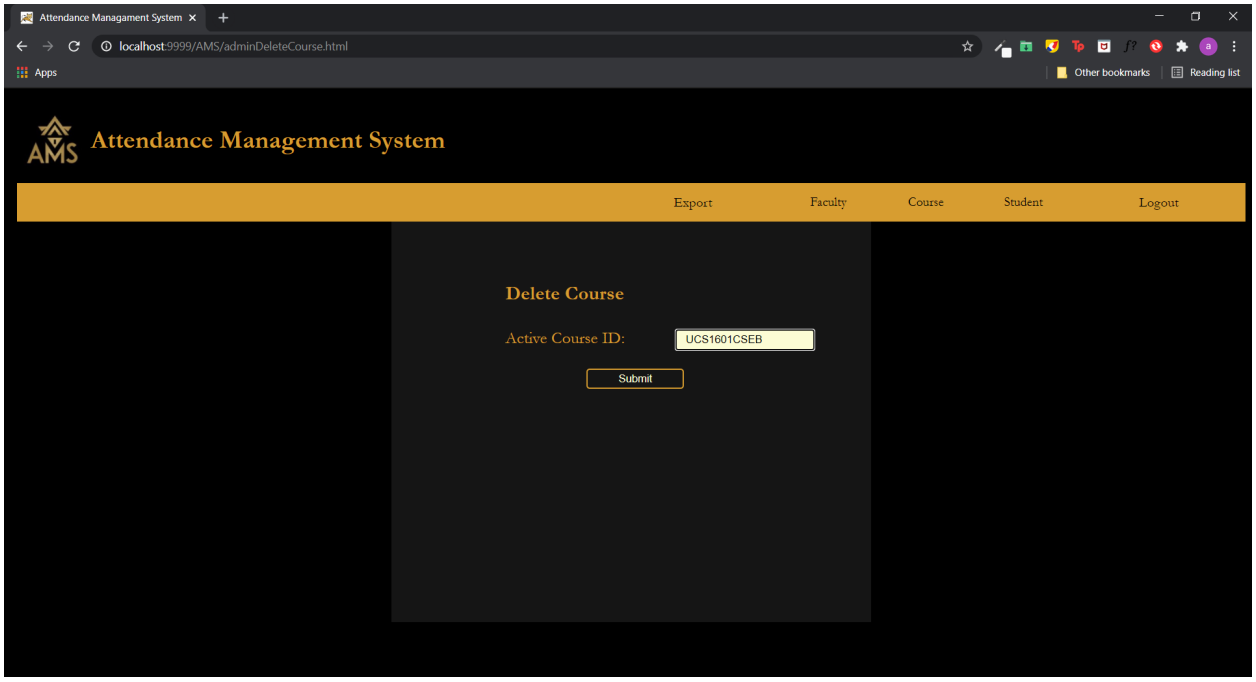
After Assigning Course:



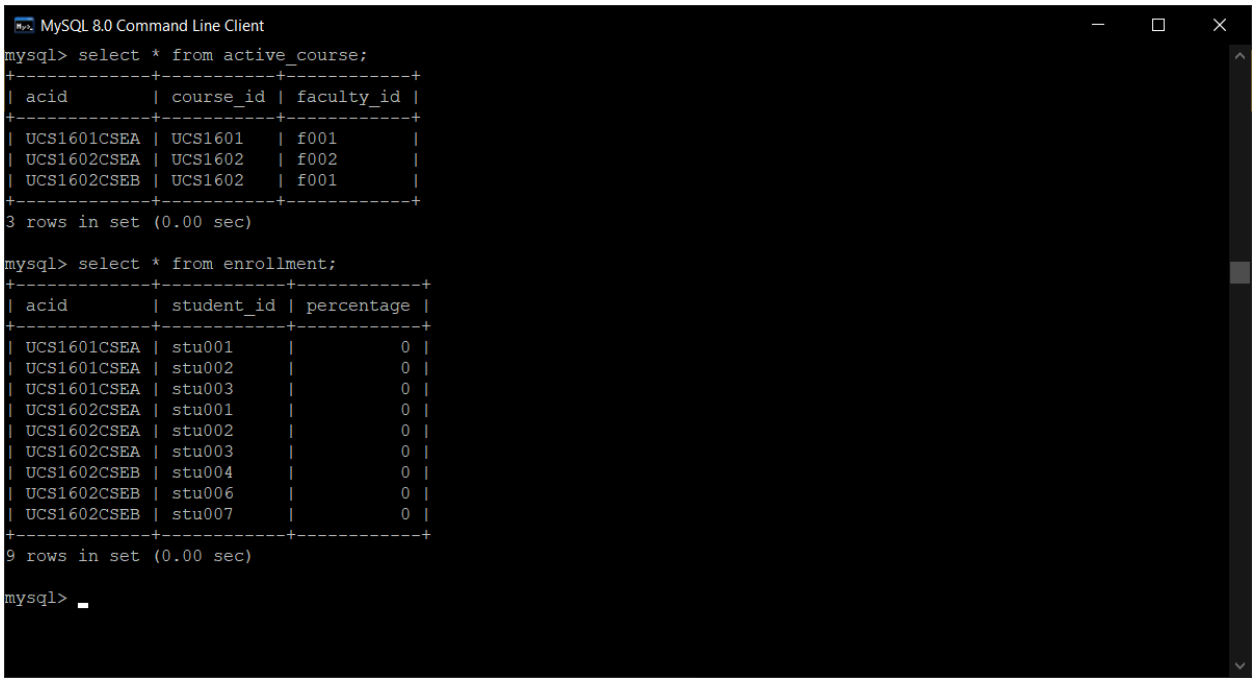
Before Deleting Course:



Delete Course:

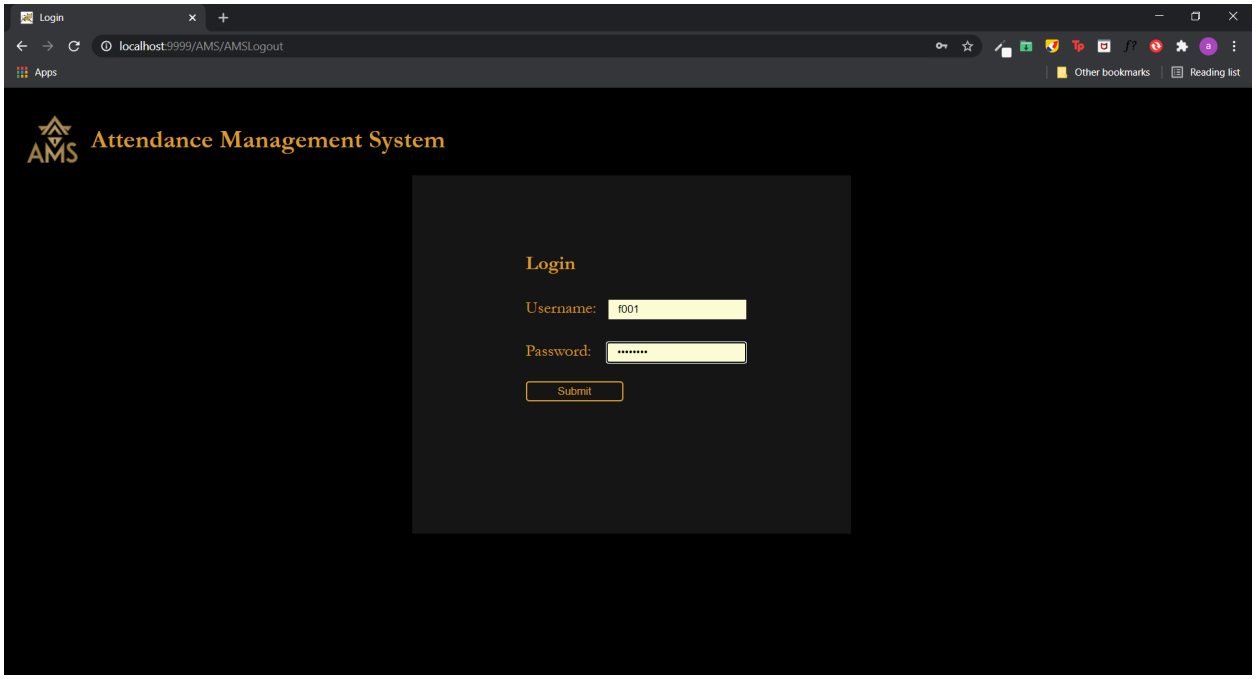


After Deleting Course:

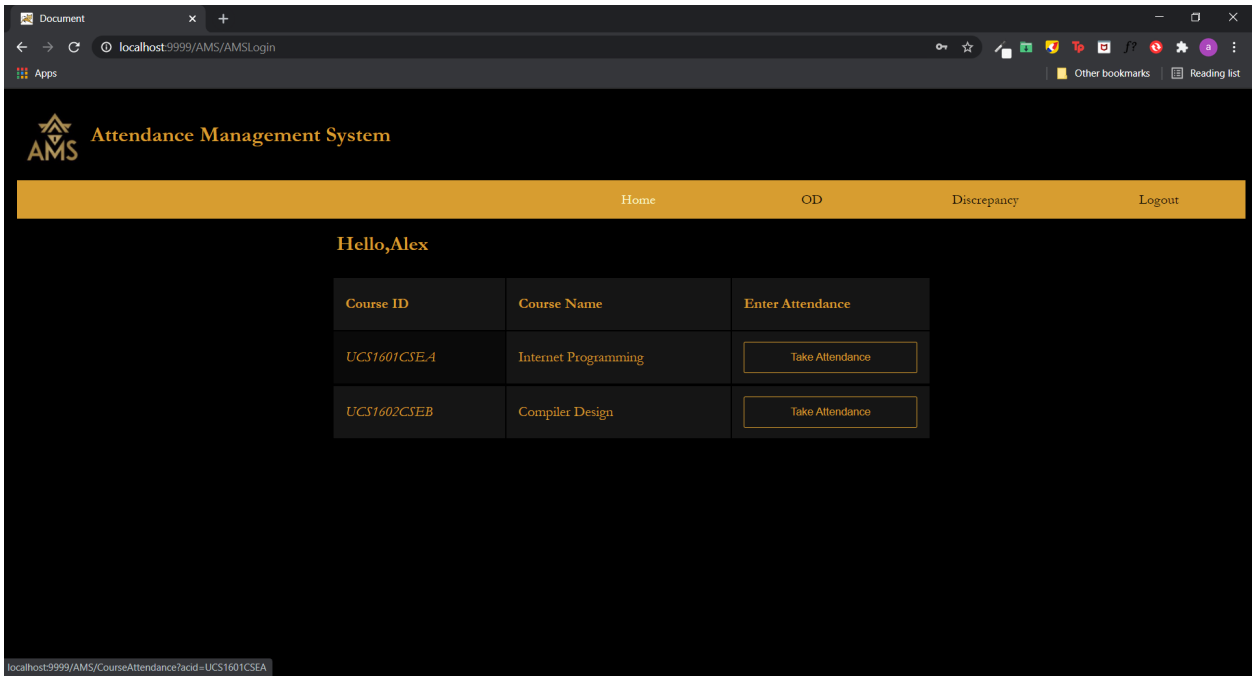


Output : Faculty

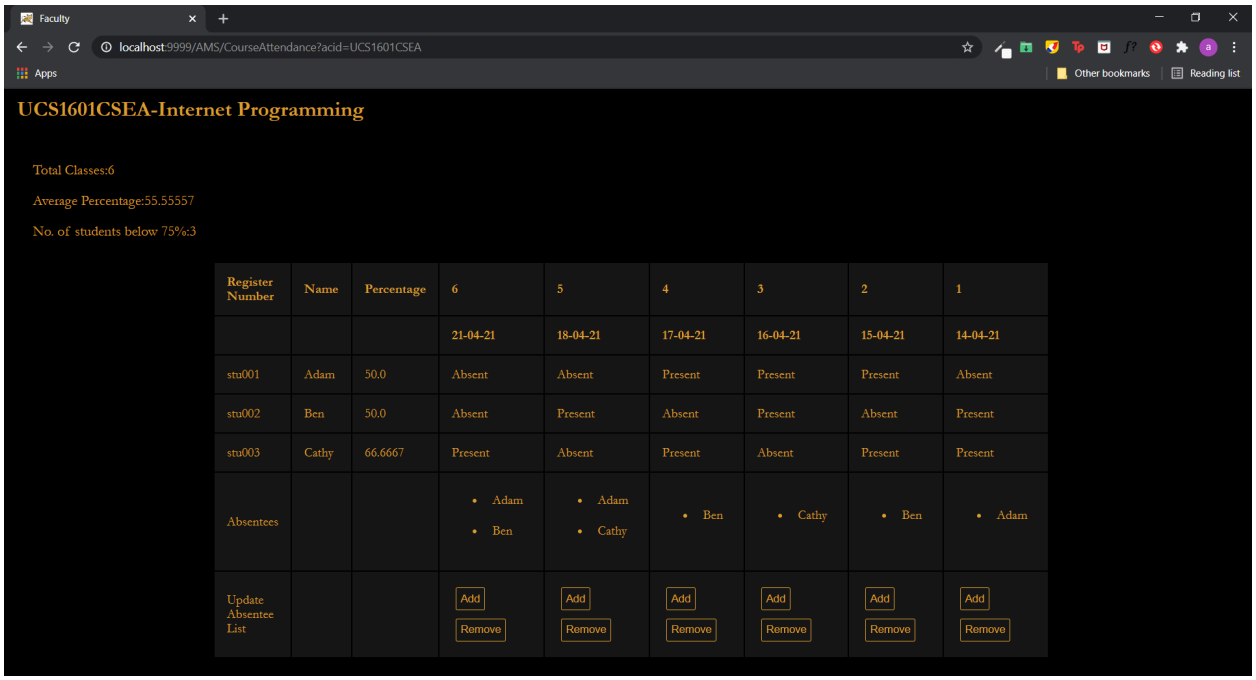
Faculty Login:



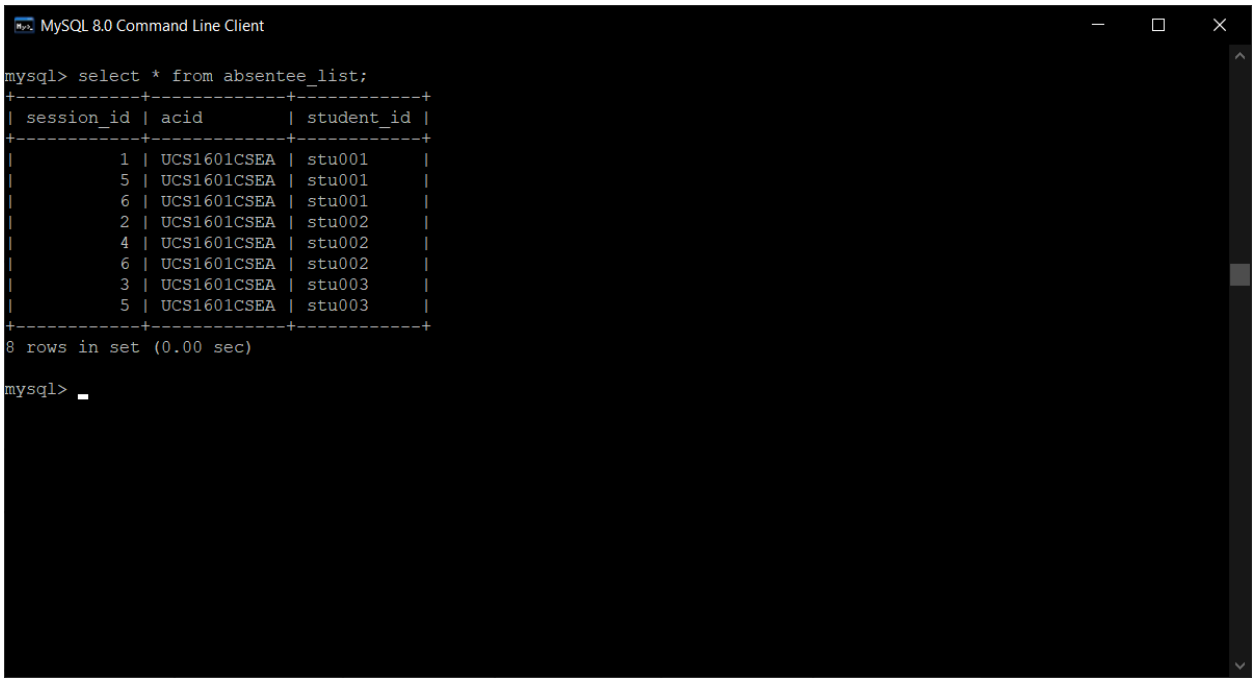
Faculty Homepage:



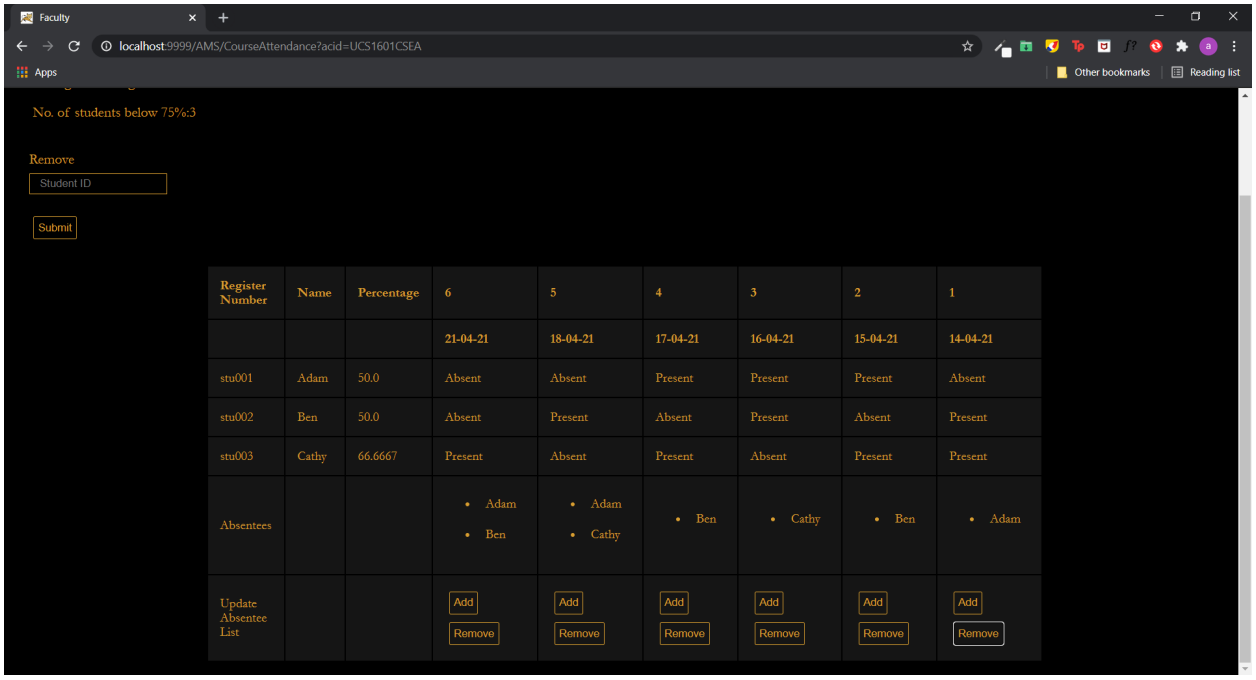
Course Attendance Details:

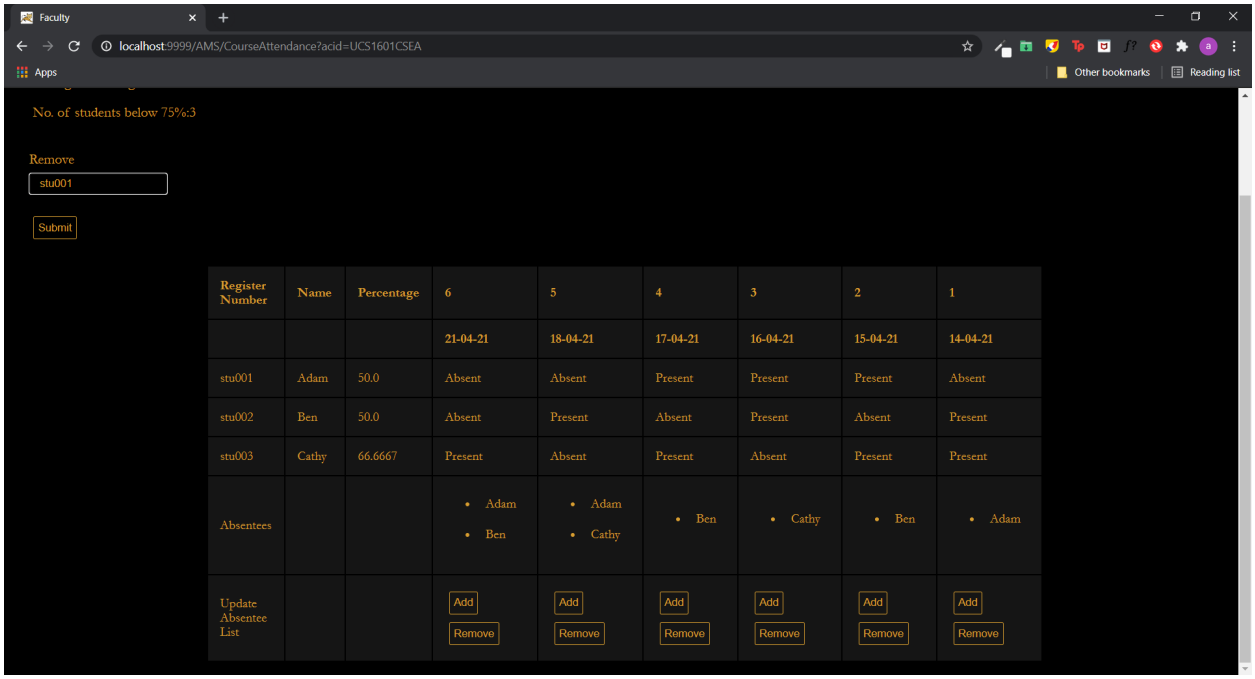


Before Removing Absentee:

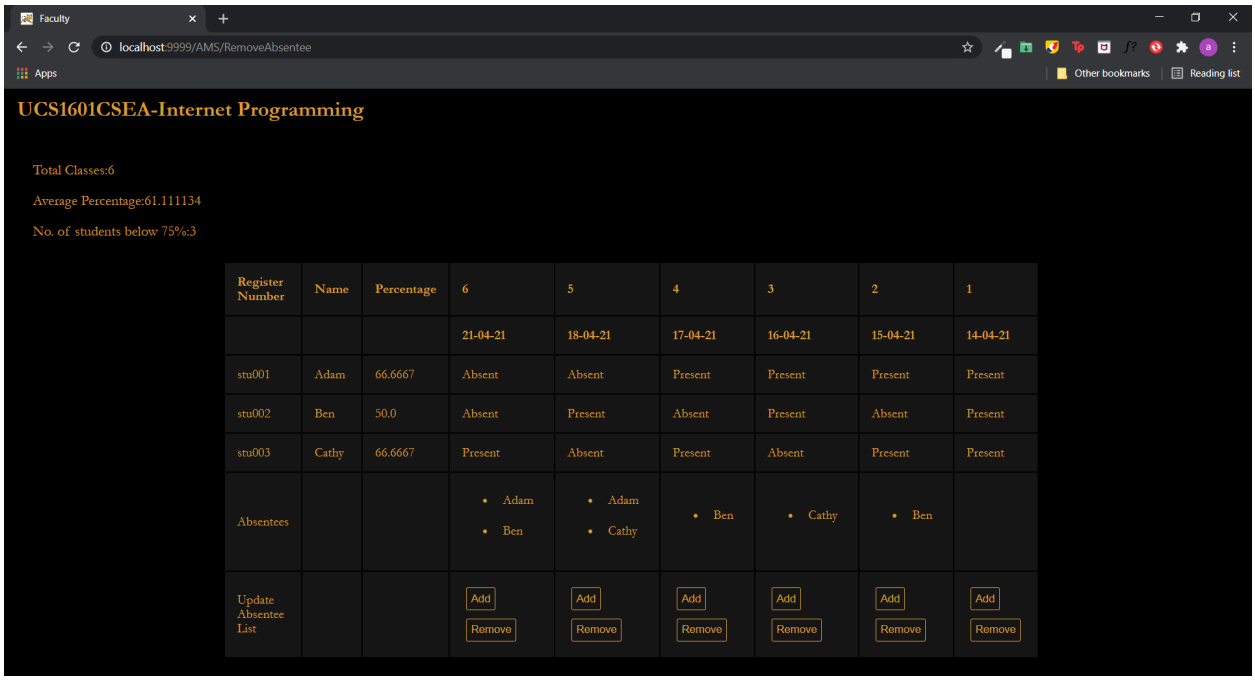


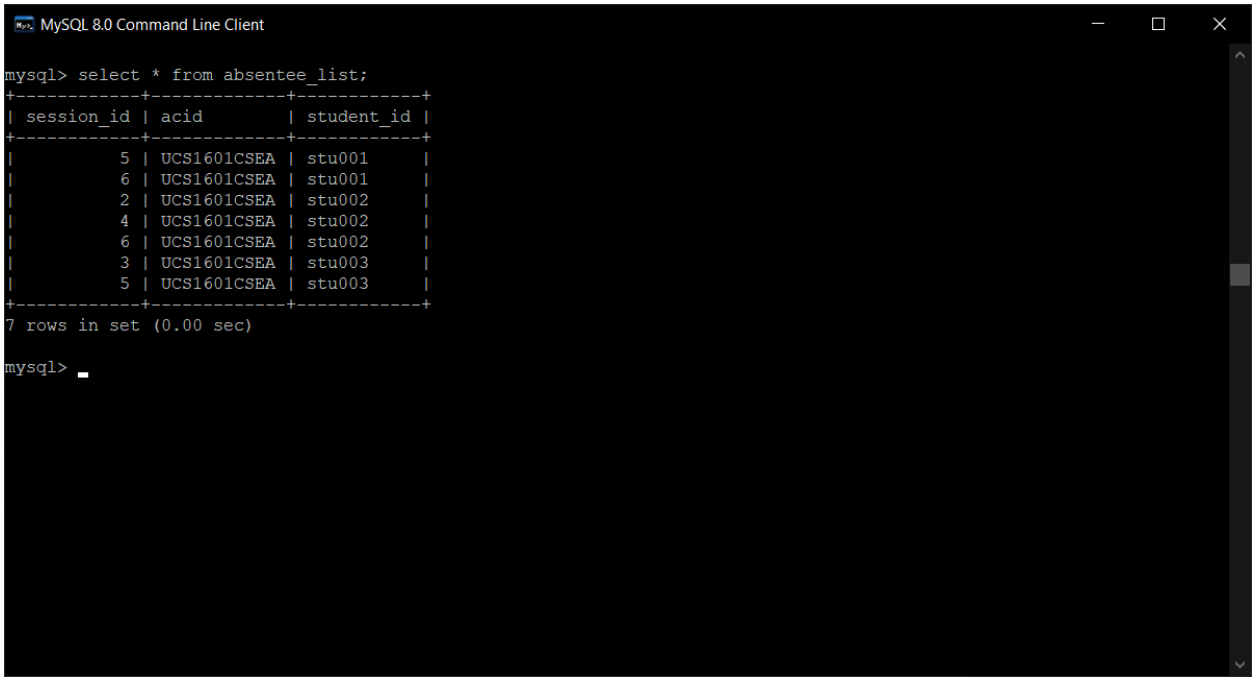
Remove Absentee:



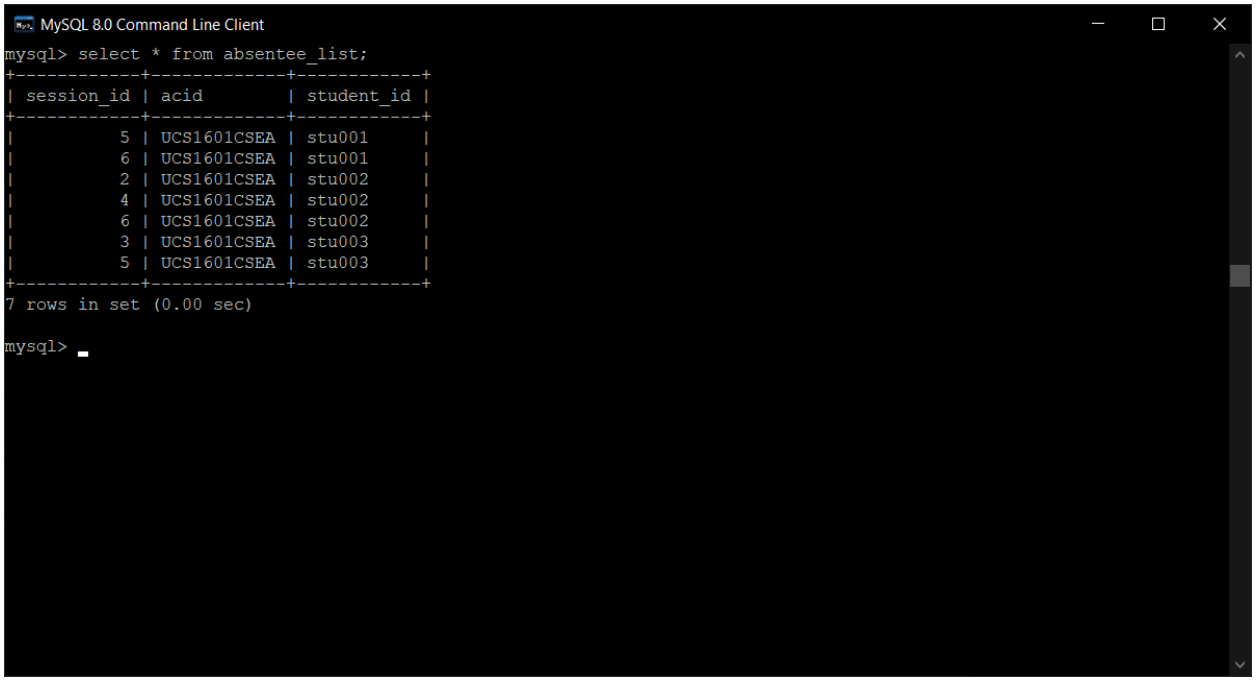


After Removing Absentee:

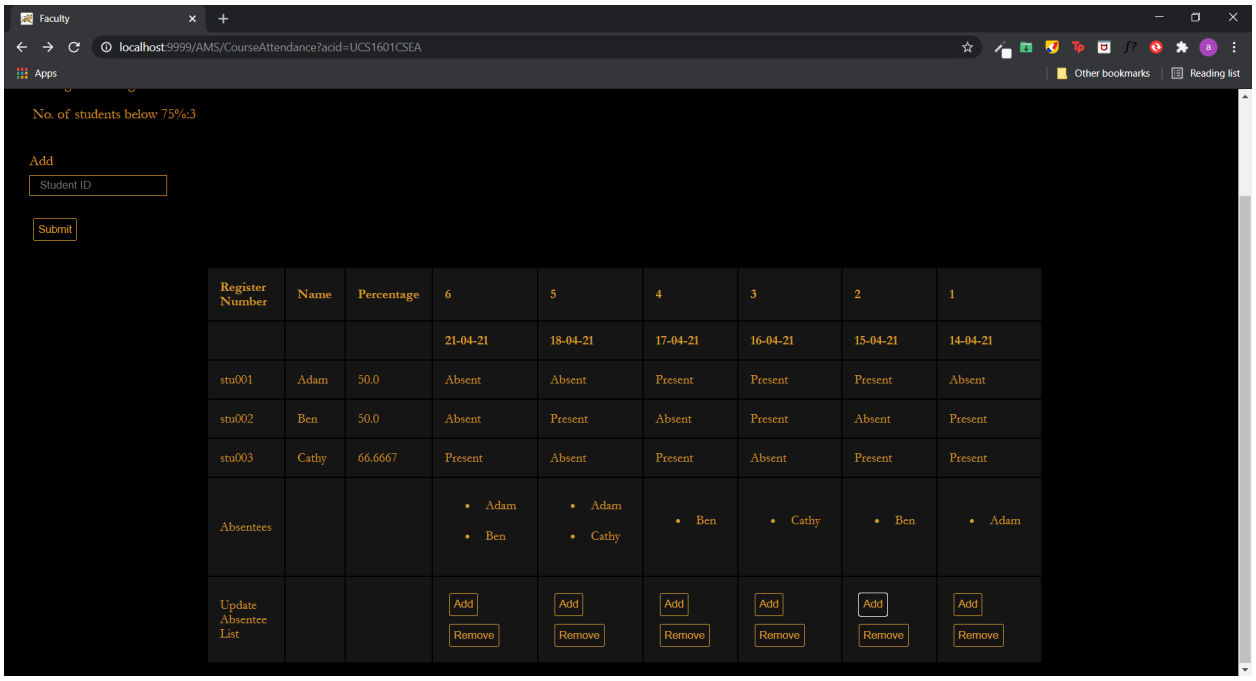


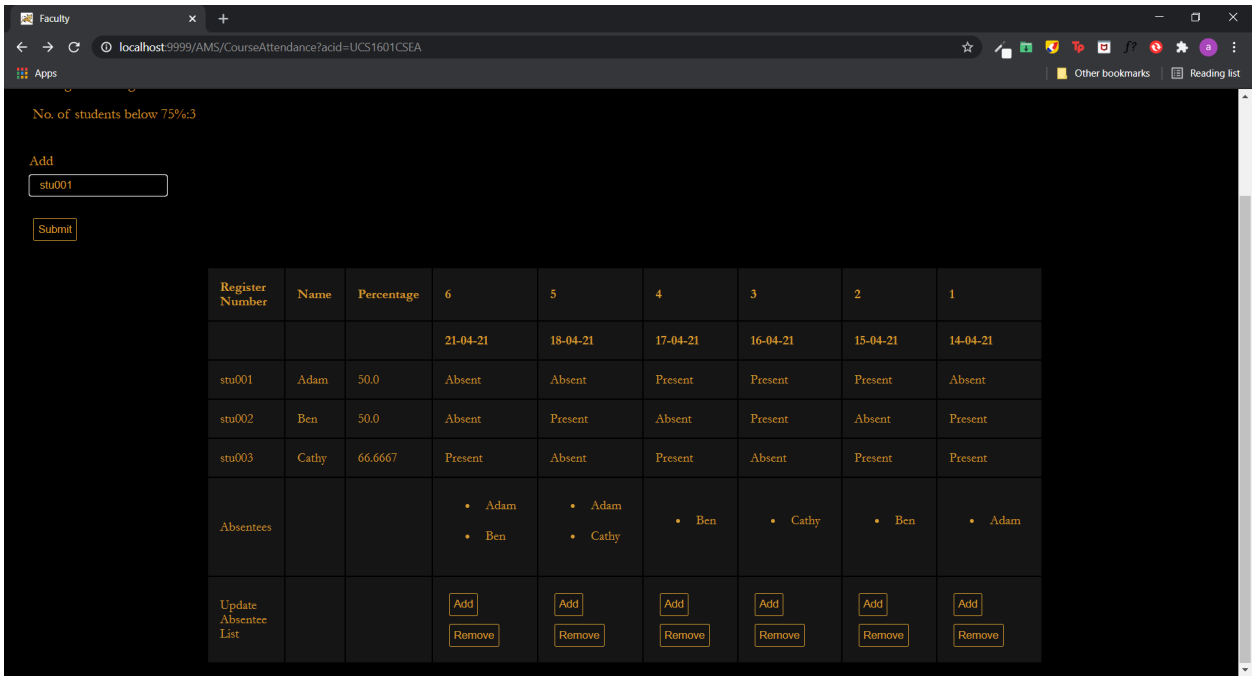


Before Adding Absentee:

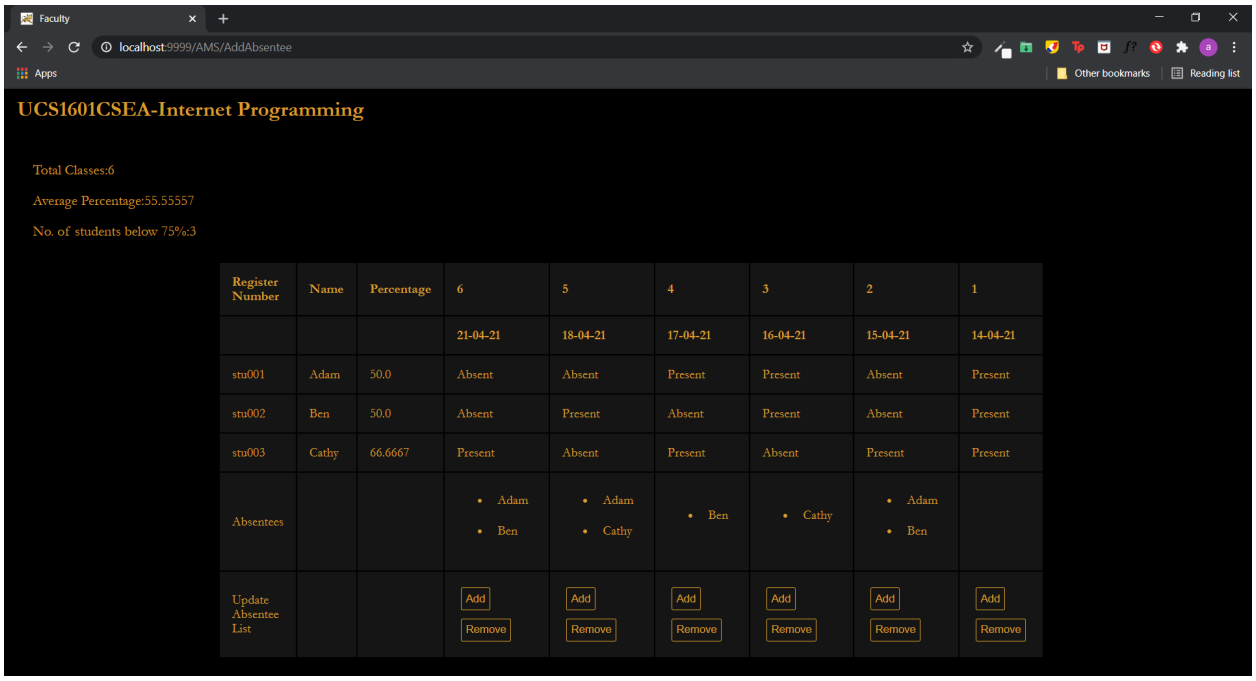


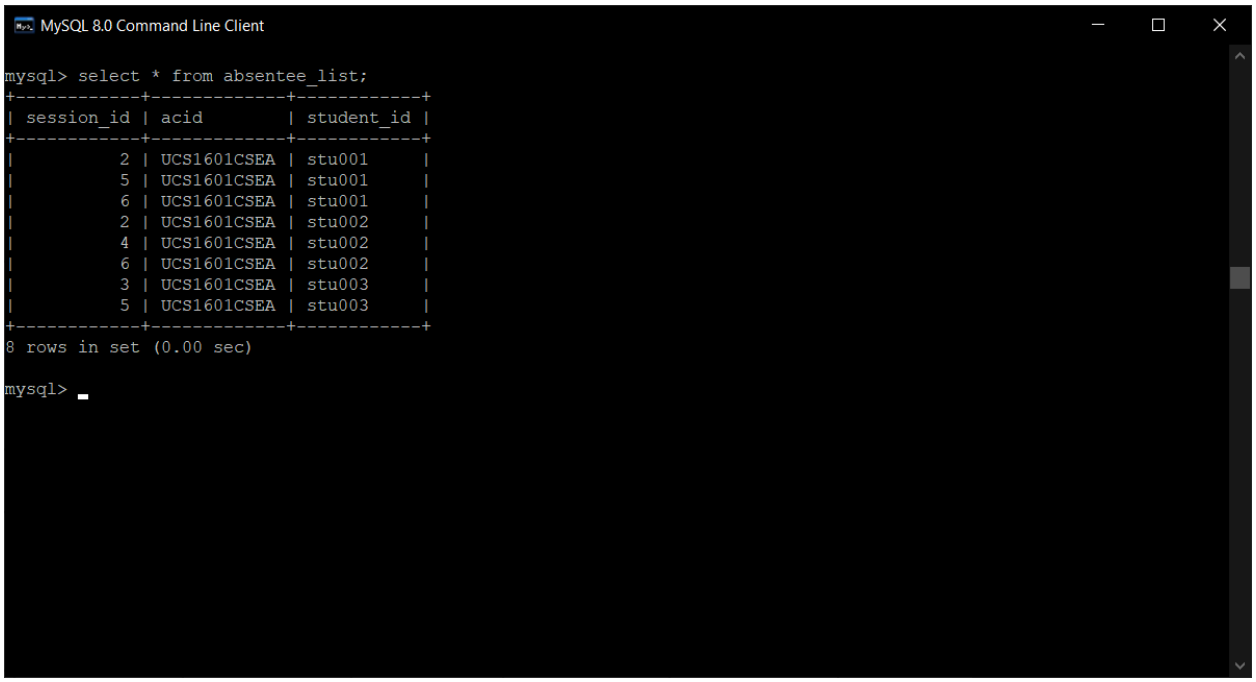
Add Absentee:



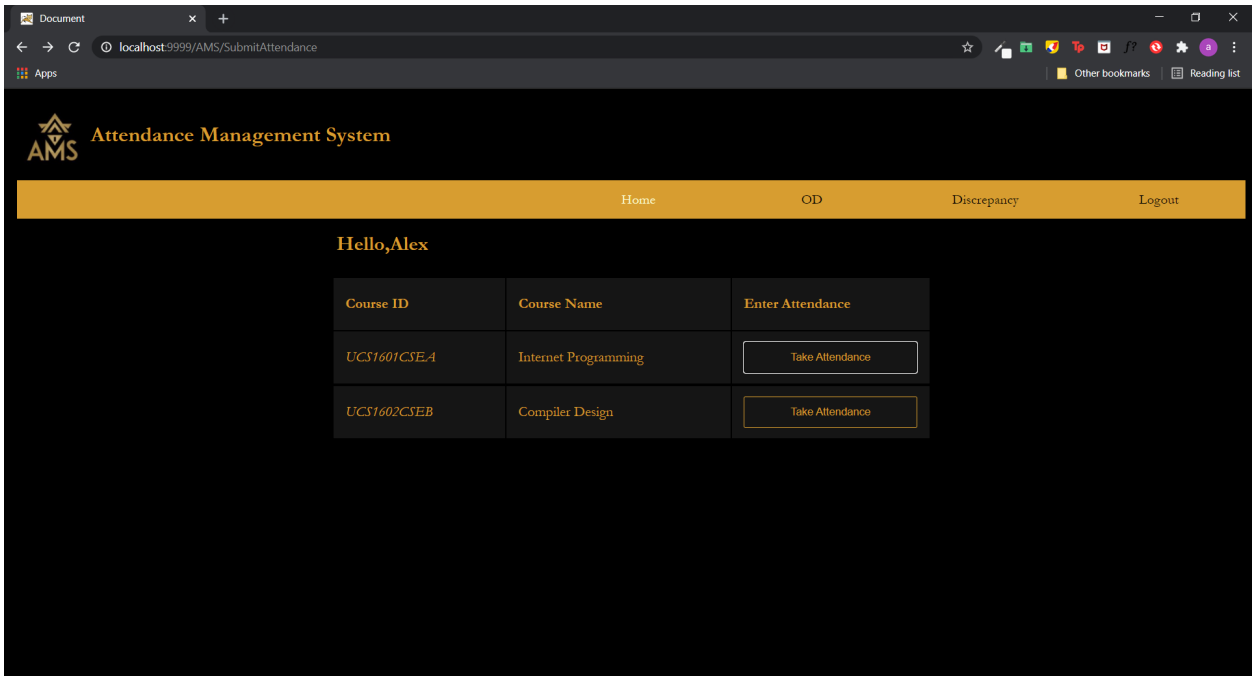


After Adding Absentee:

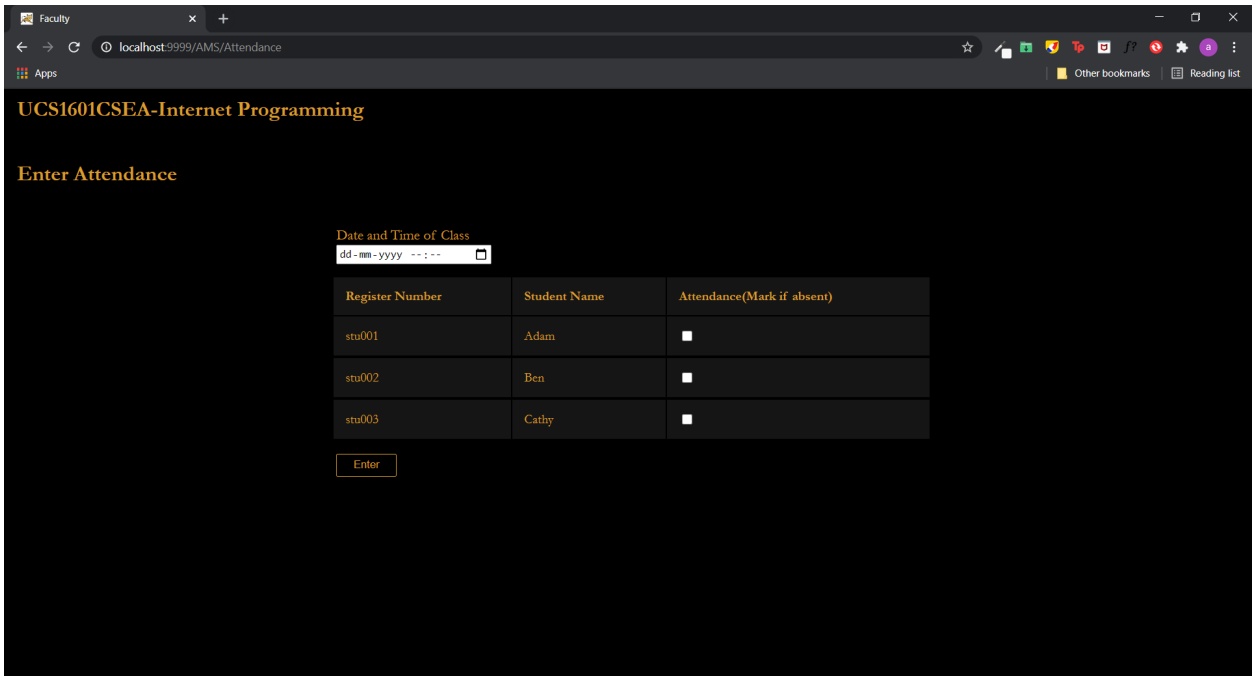


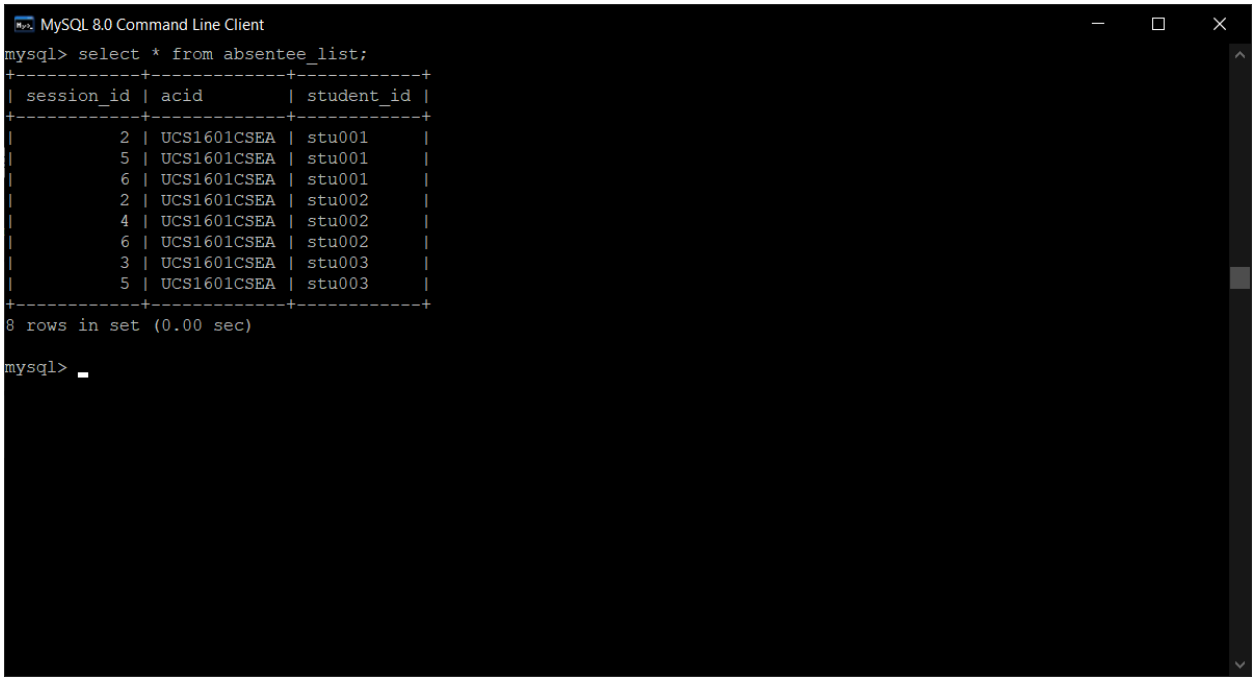


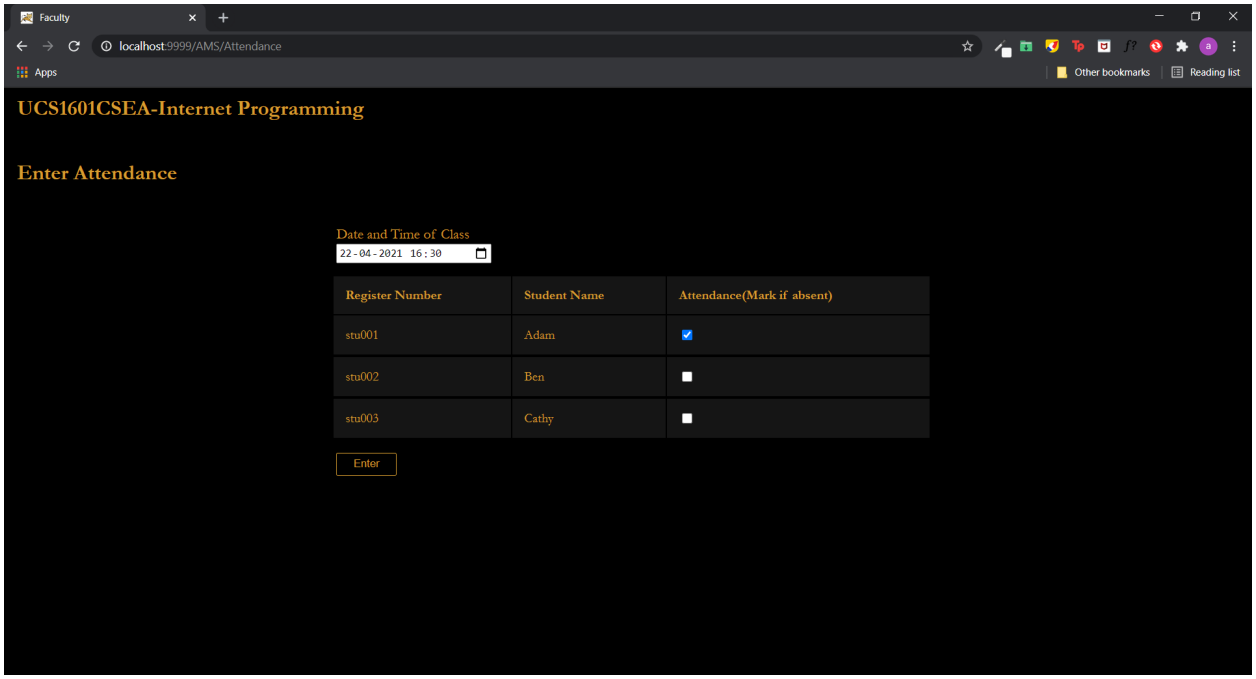
Faculty Homepage - Take Attendance Hover



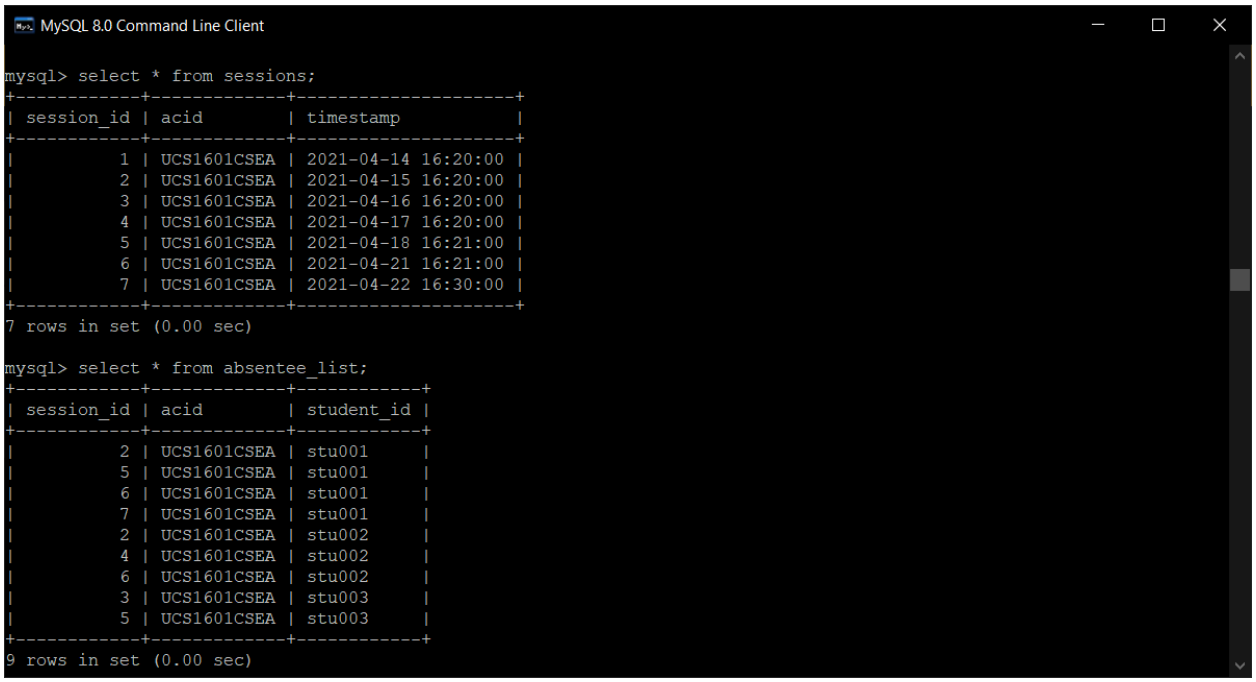
Take Attendance:

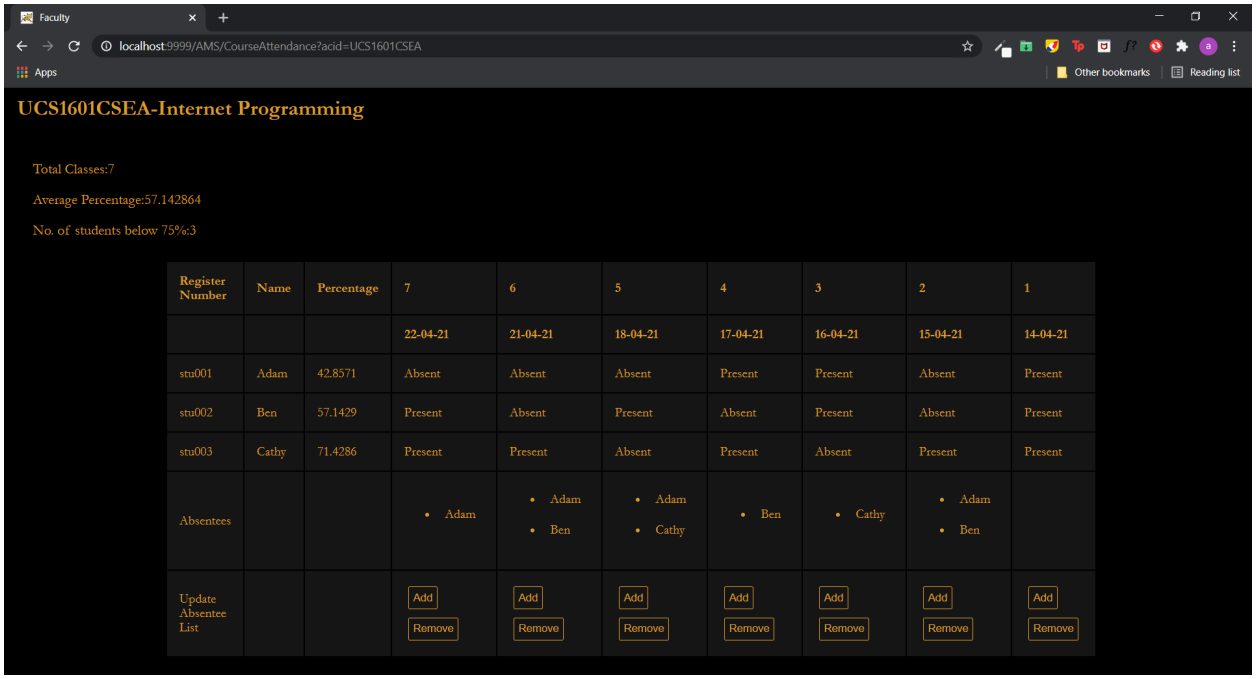



MySQL 8.0 Command Line Client
mysql> select * from absentee_list;
+------------+-------------+------------+
| session_id | acid | student_id |
+------------+-------------+------------+
2	UCS1601CSEA	stu001
5	UCS1601CSEA	stu001
6	UCS1601CSEA	stu001
2	UCS1601CSEA	stu002
4	UCS1601CSEA	stu002
6	UCS1601CSEA	stu002
3	UCS1601CSEA	stu003
5	UCS1601CSEA	stu003
+------------+-------------+------------+
8 rows in set (0.00 sec)
mysql>


Faculty
localhost:9999/AMS/Attendance
Apps
Other bookmarks
Reading list
UCS1601CSEA-Internet Programming
Enter Attendance
Date and Time of Class
22-04-2021 16:30
Register Number
Student Name
Attendance(Mark if absent)
stu001
Adam
stu002
Ben
stu003
Cathy
Enter

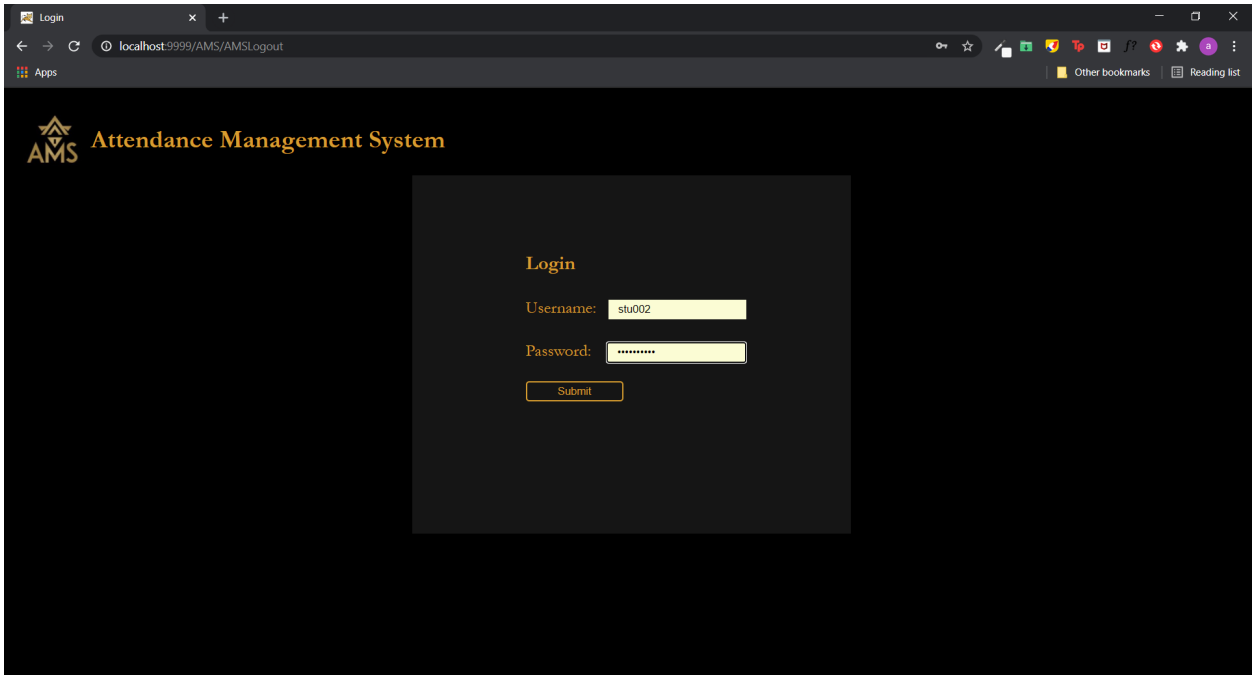
After Taking Attendance:



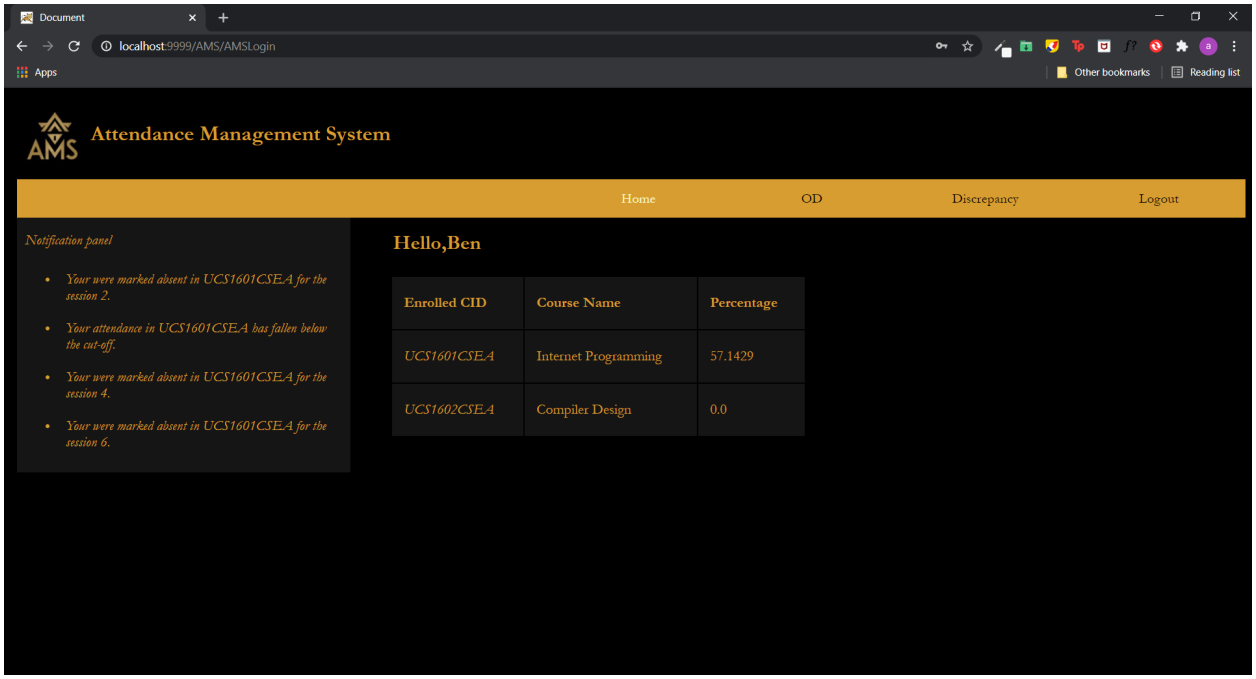


Output : STUDENT

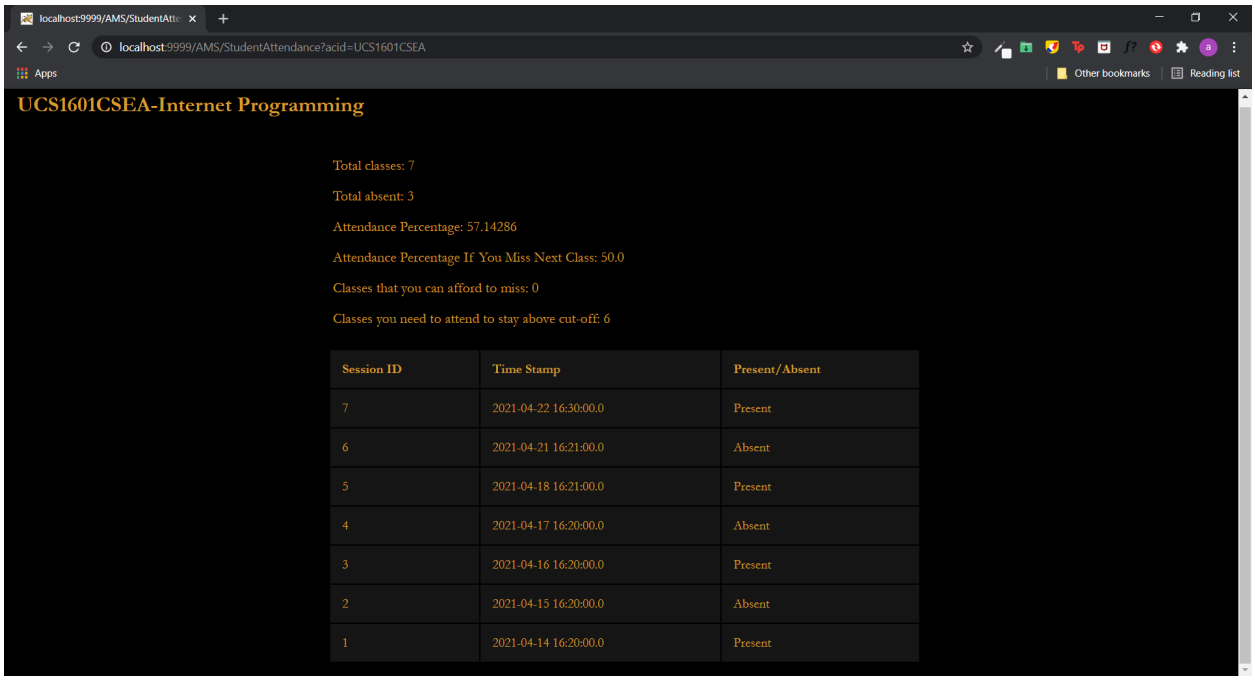
Student Login:



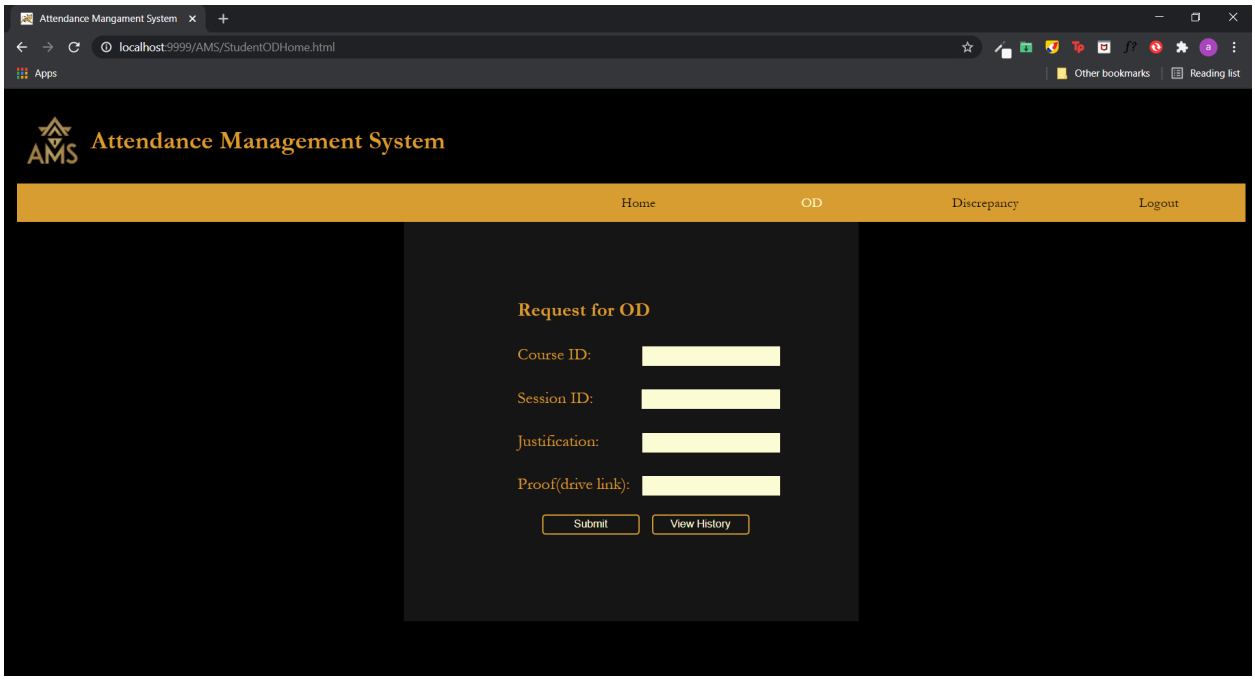
Student Homepage:



Student Attendance Details for selected Course:

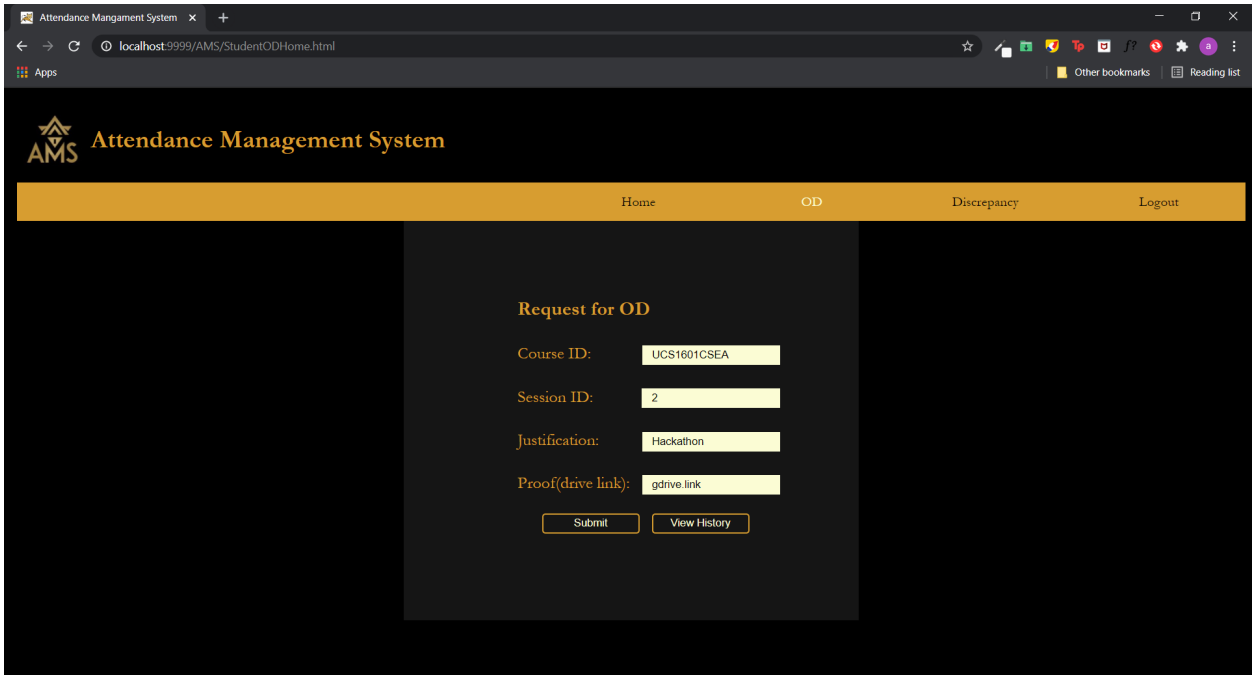


Student - Request for OD

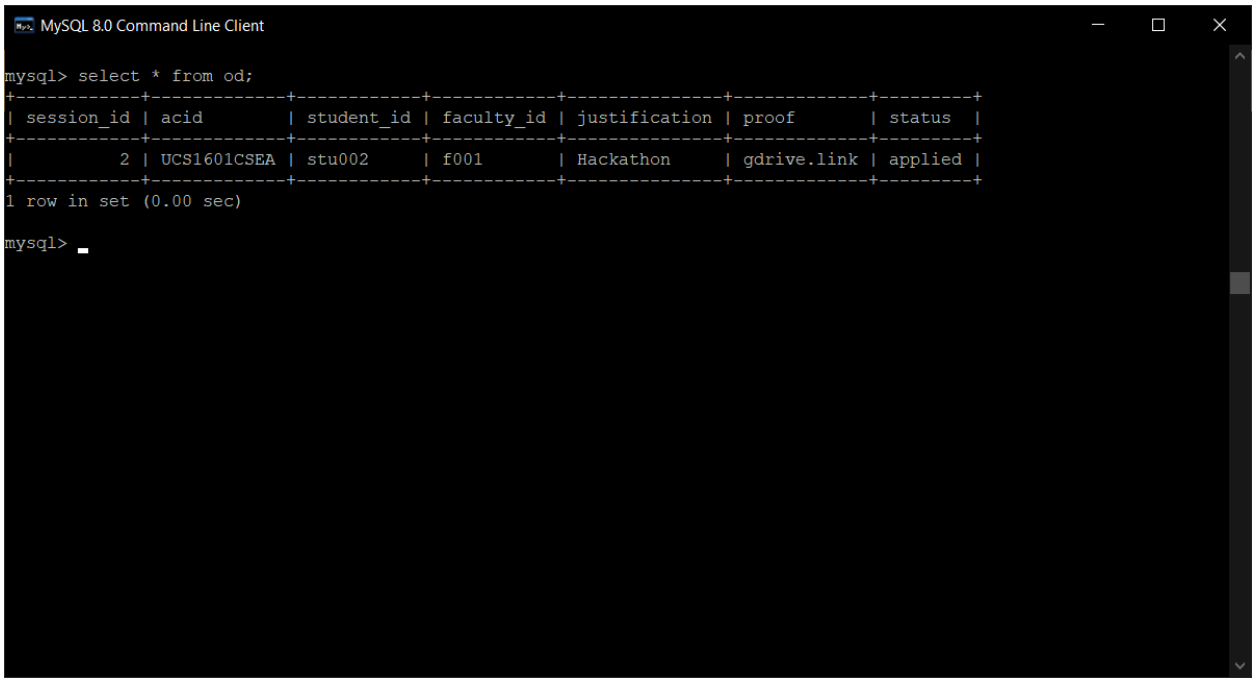


Before OD Request:

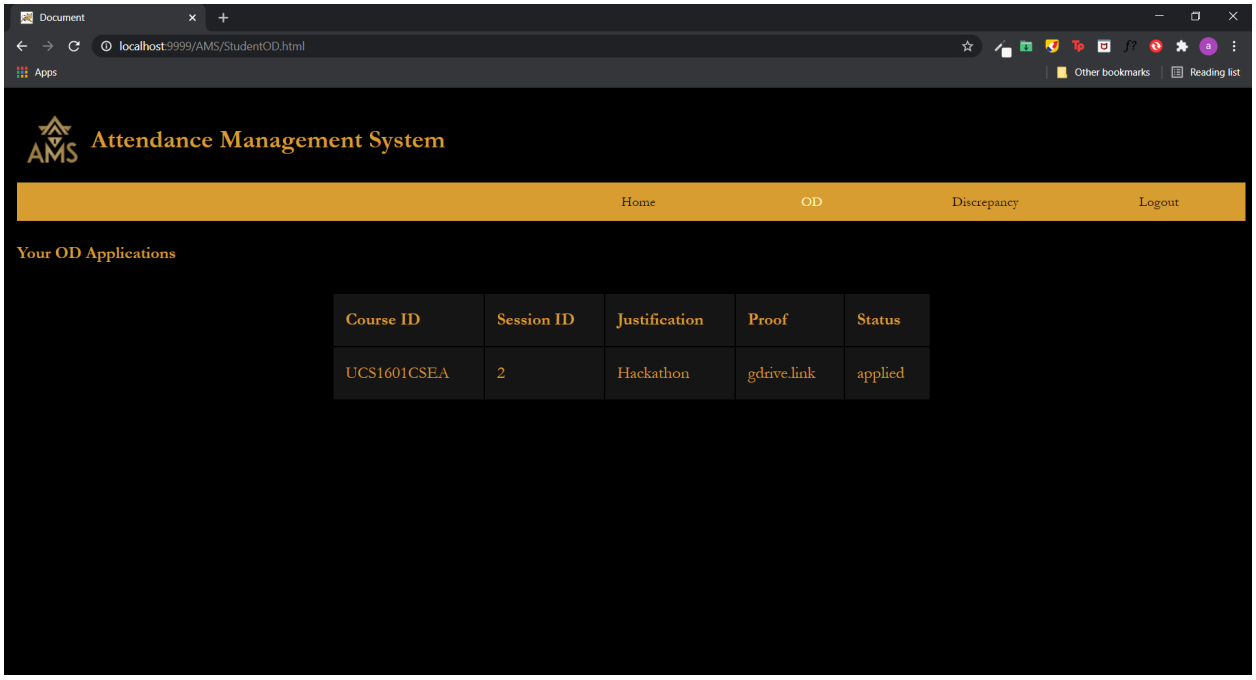
Request OD:



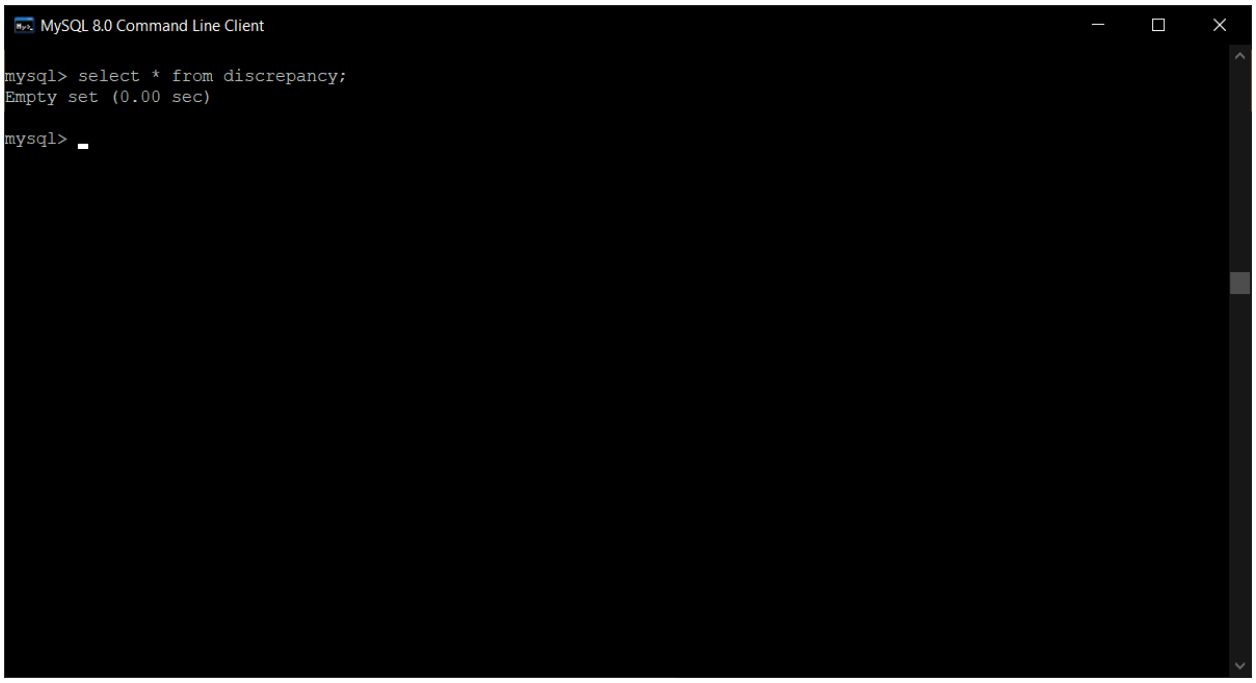
After OD Request:



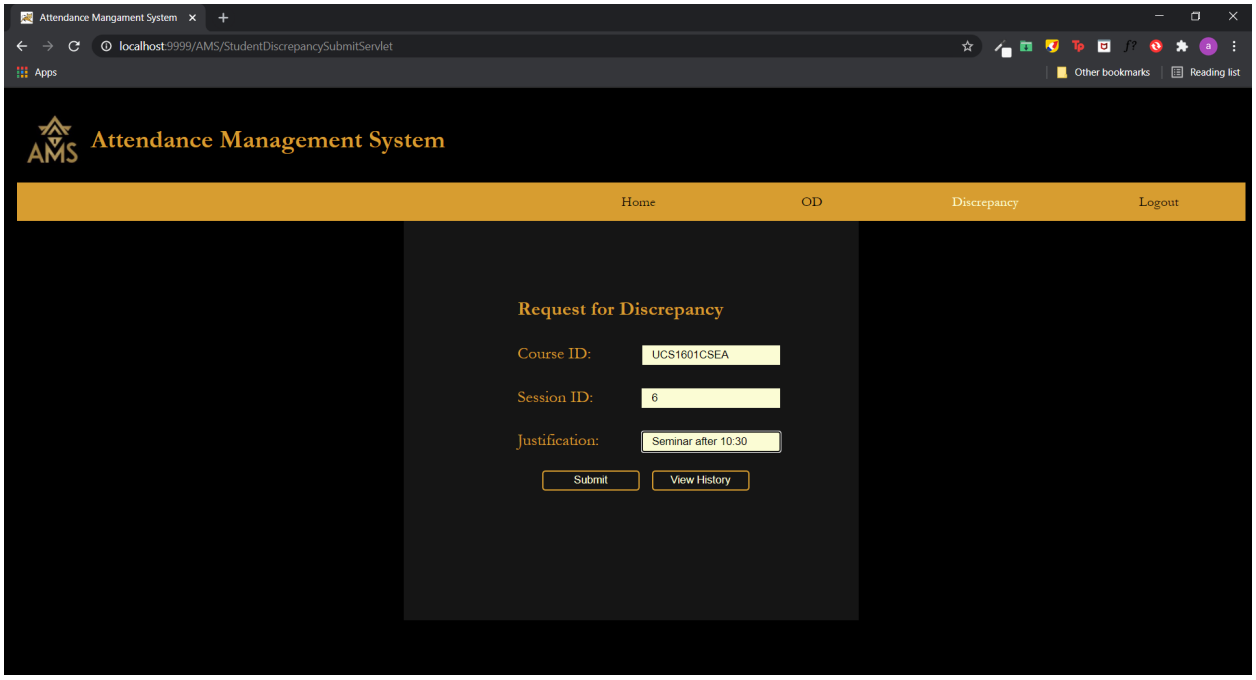
OD Applications:



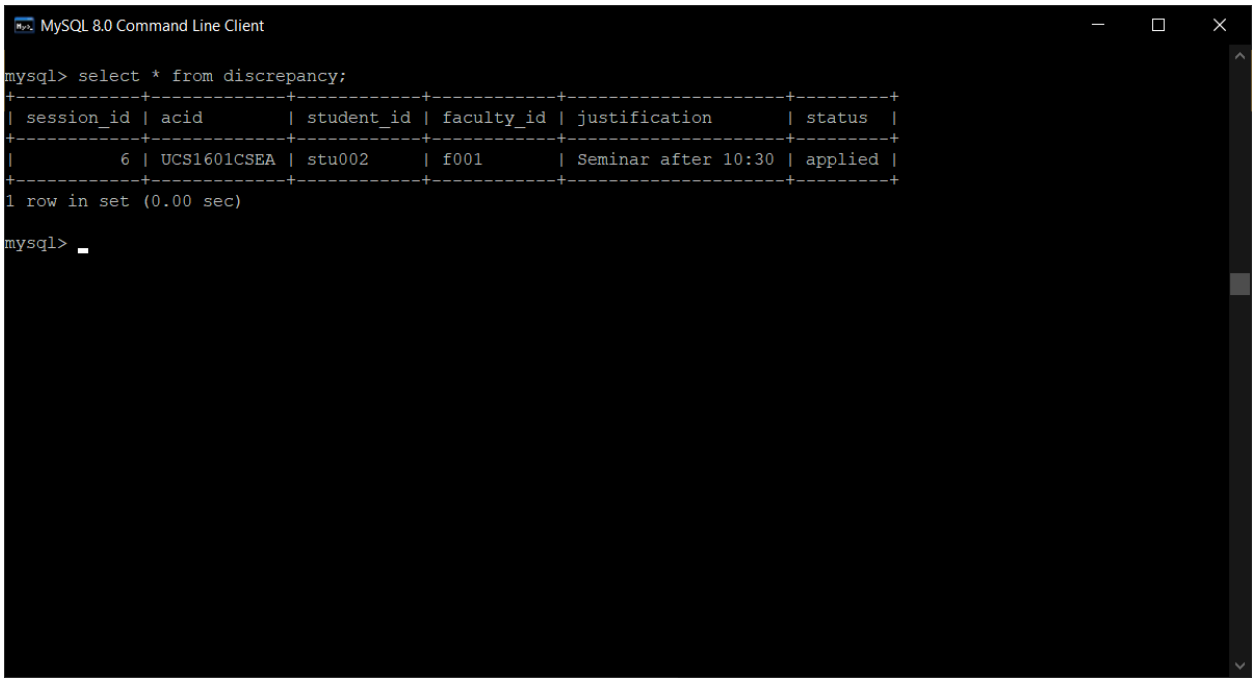
Before Reporting Discrepancy:



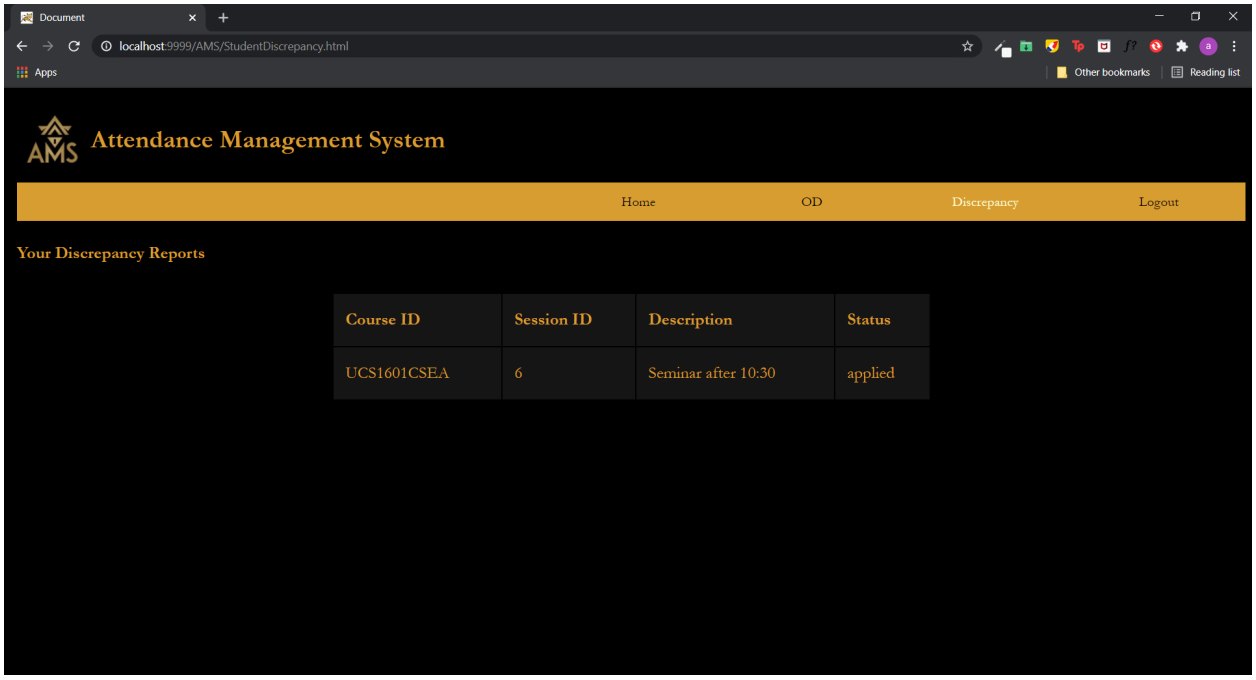
Report Discrepancy:



After Reporting Discrepancy:

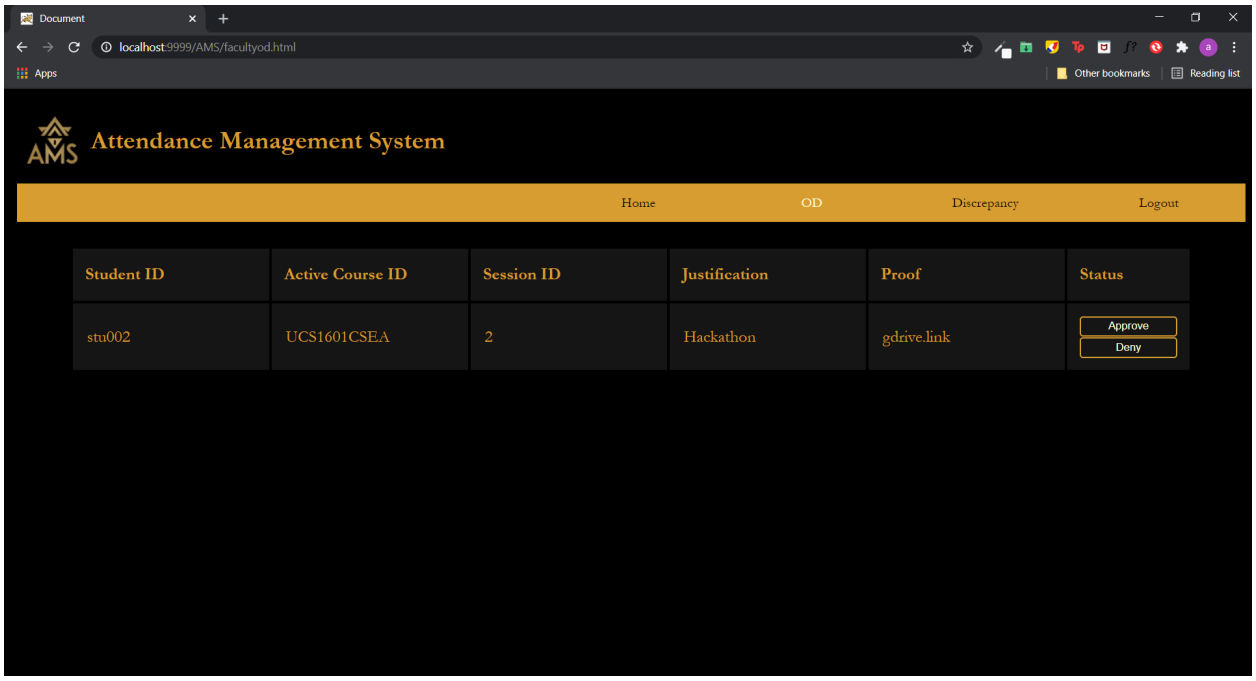


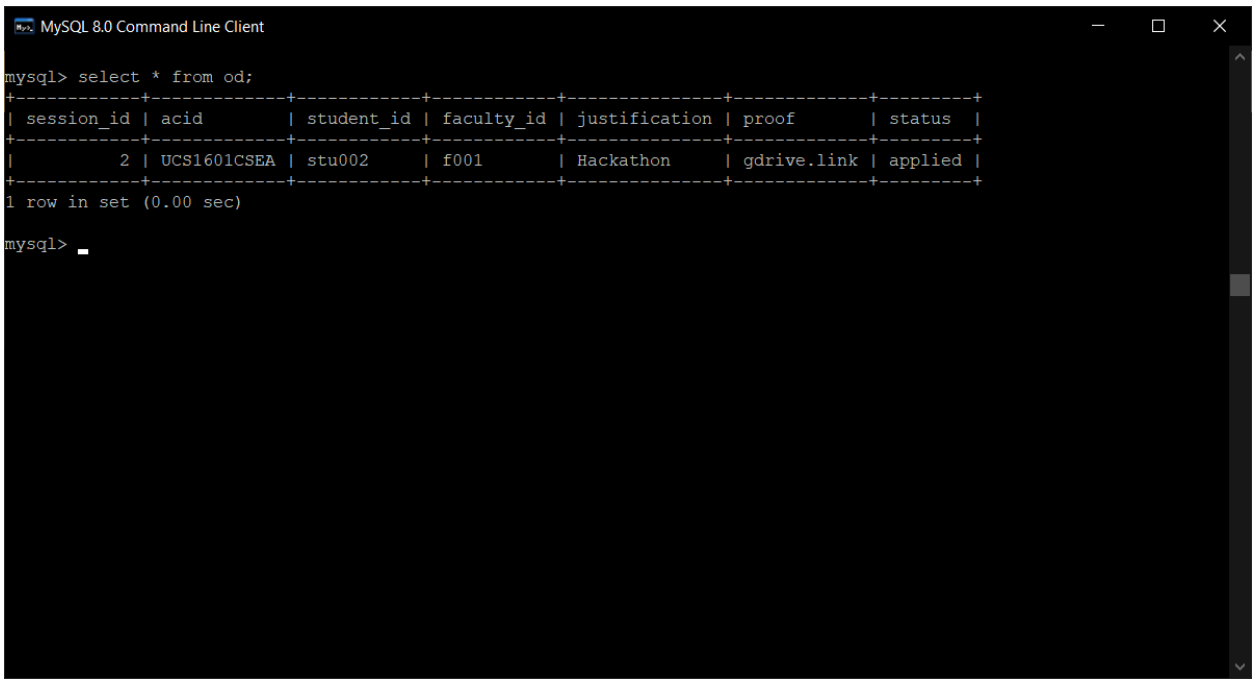
Discrepancy Reports:



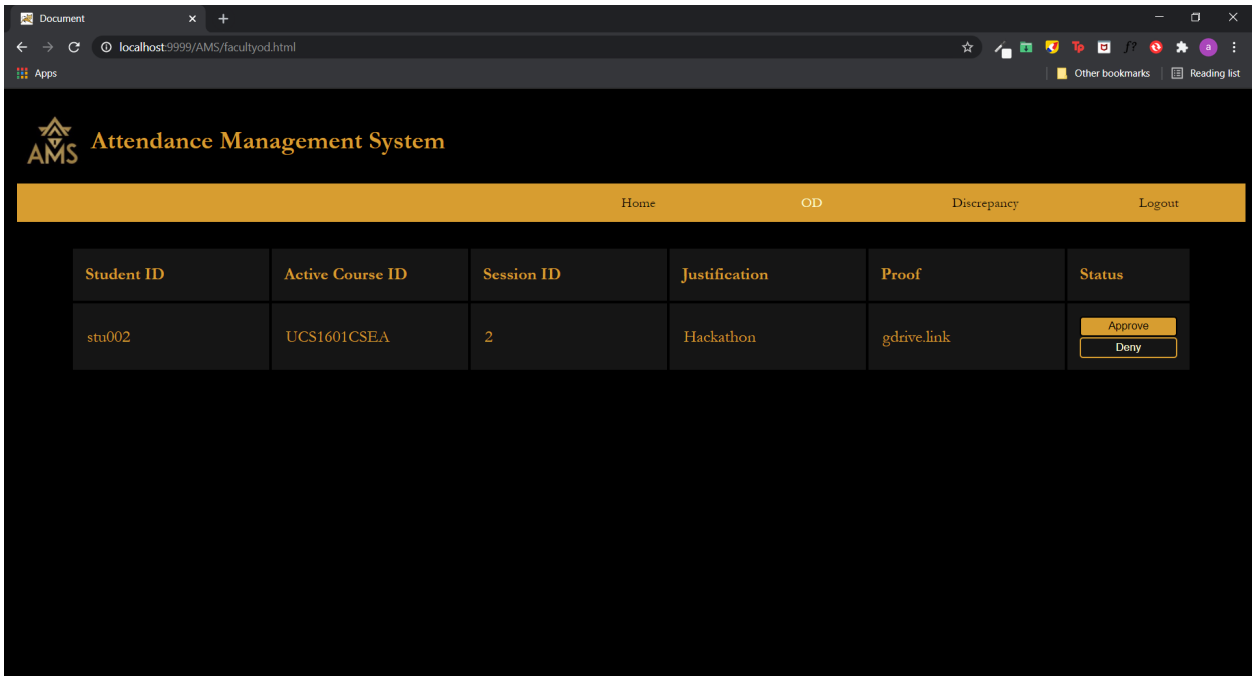
Output : OD Processing

Faculty- OD Applications:

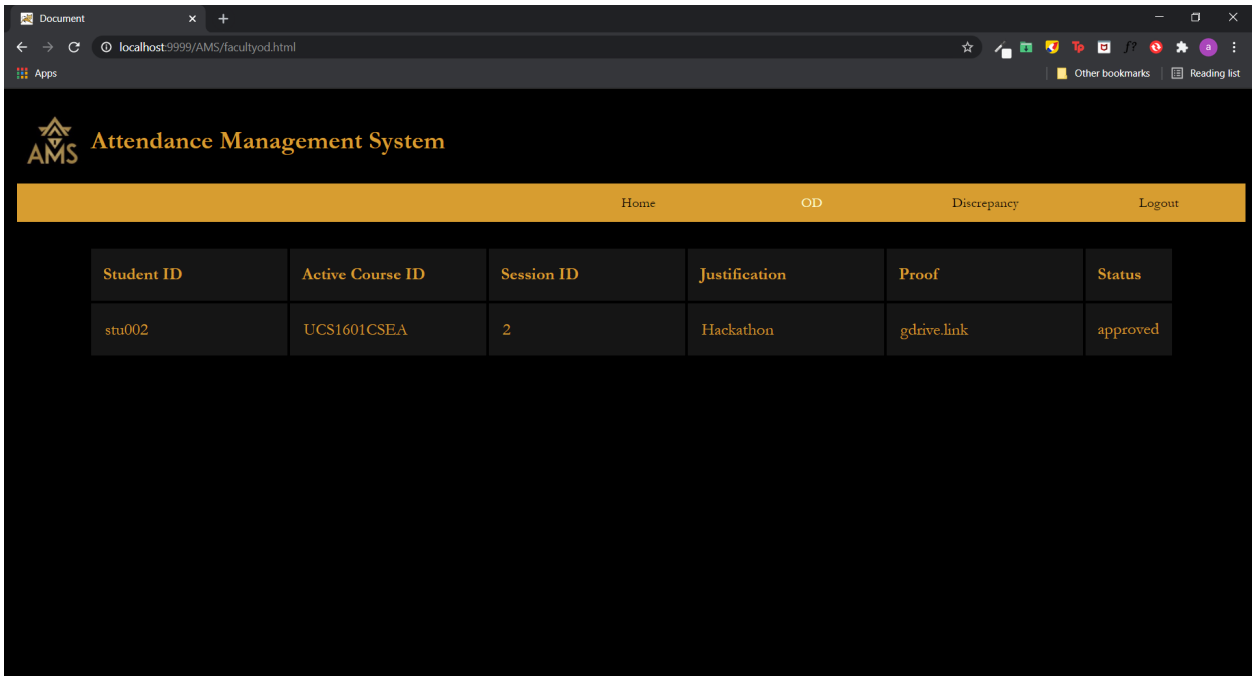




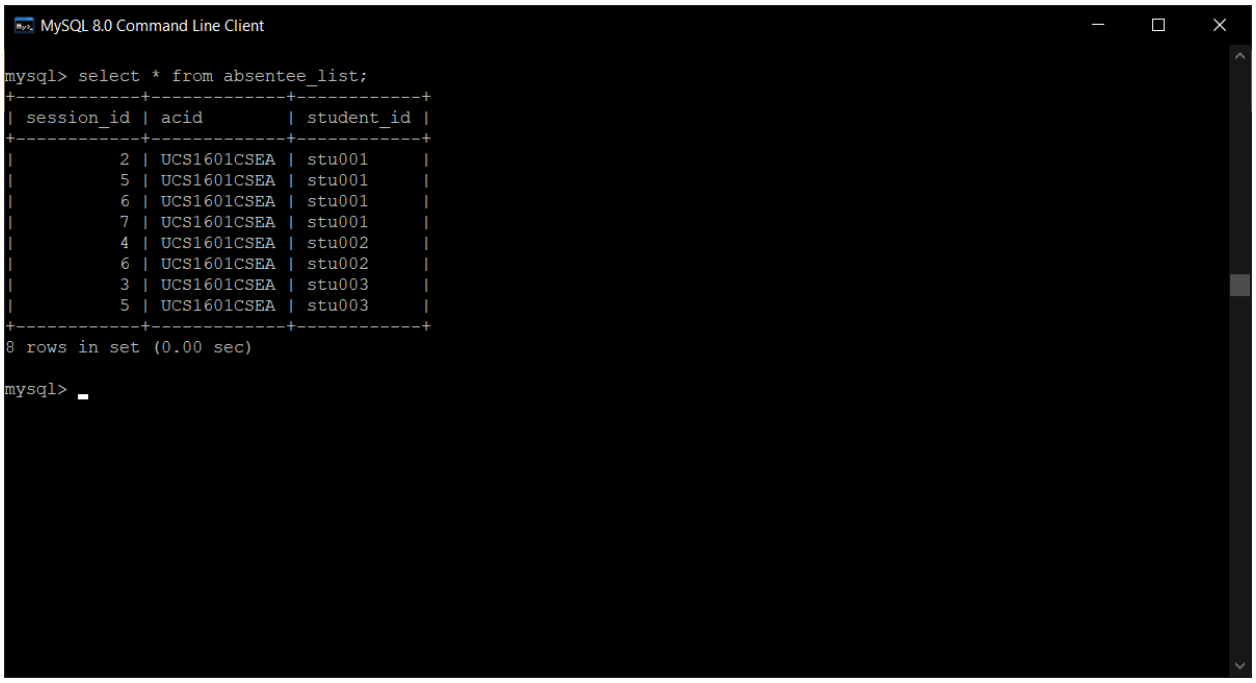
Response to OD Request:



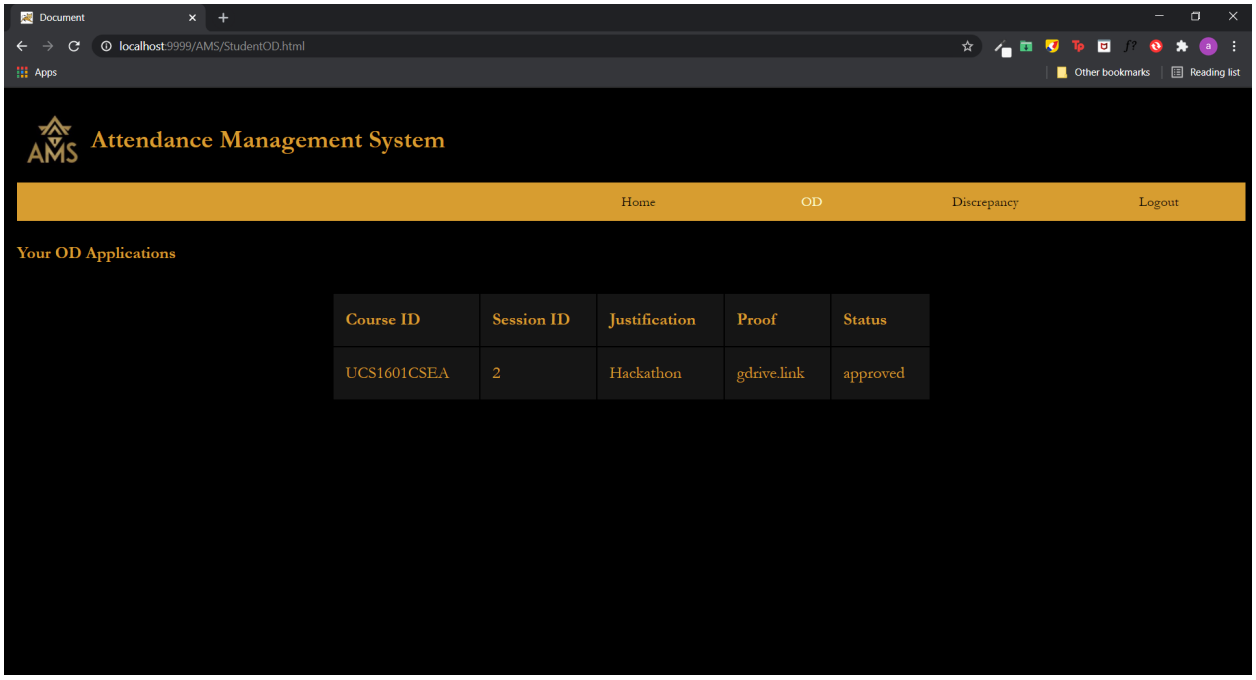
After responding to OD Request:



Updated Attendance after Approval of OD:

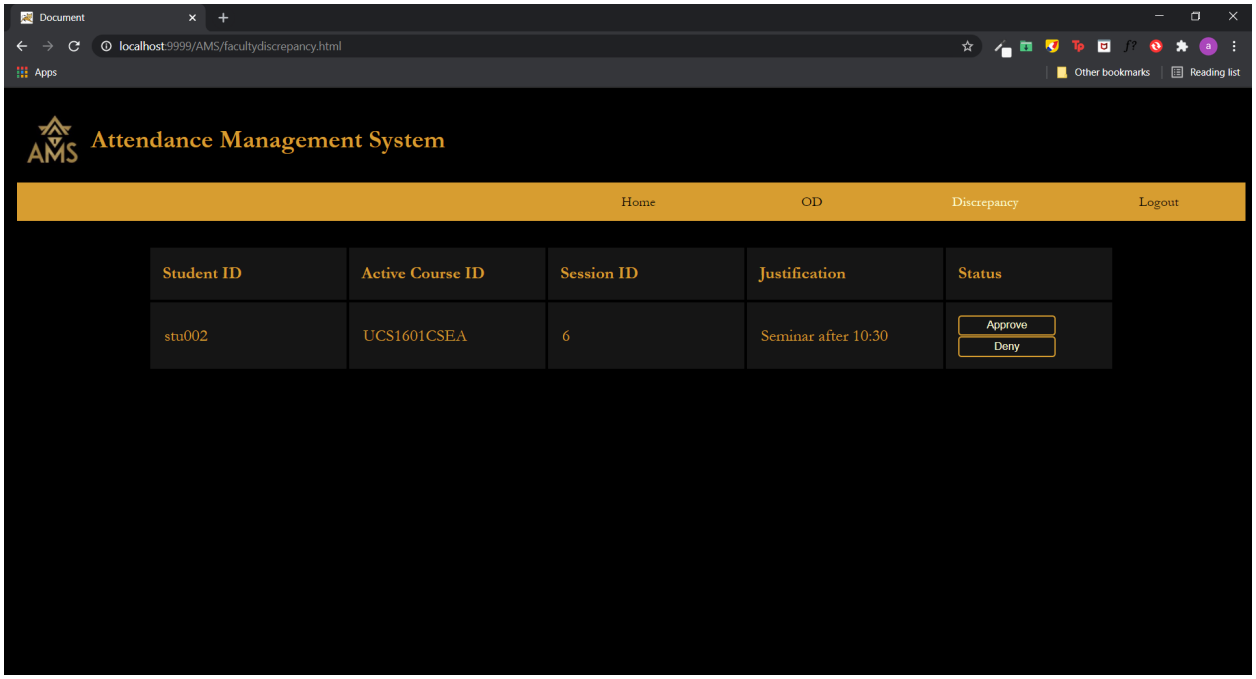


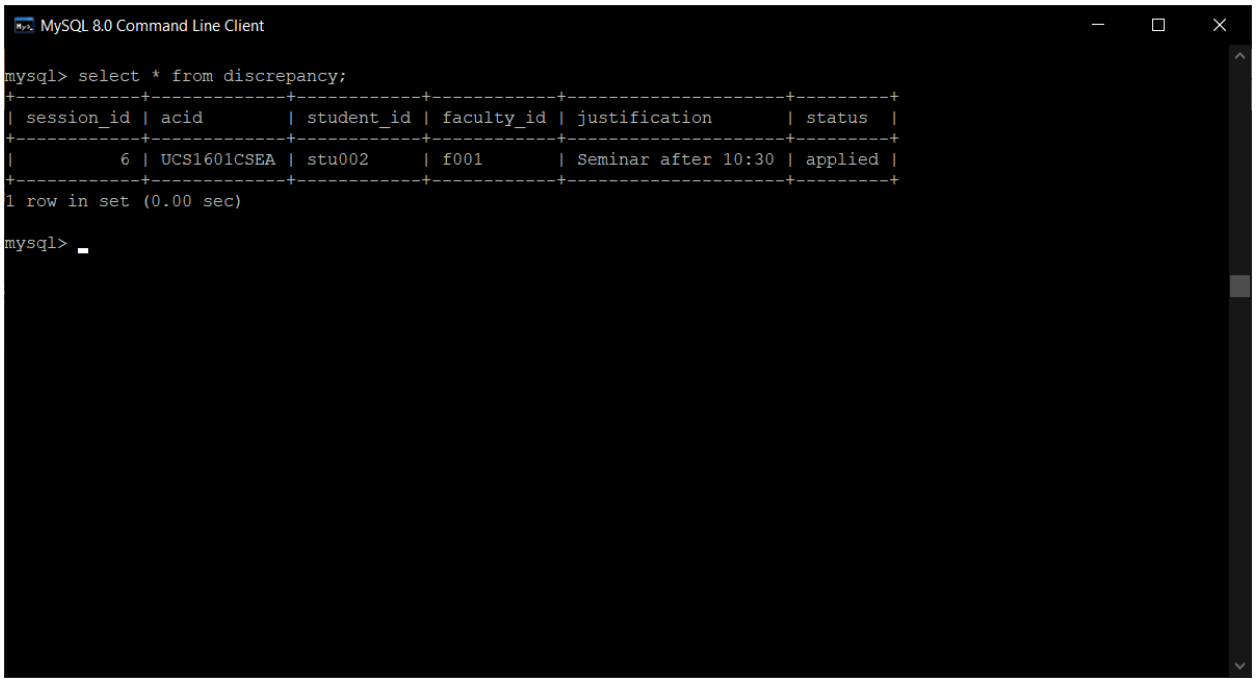
Student - OD Approved Status Change:



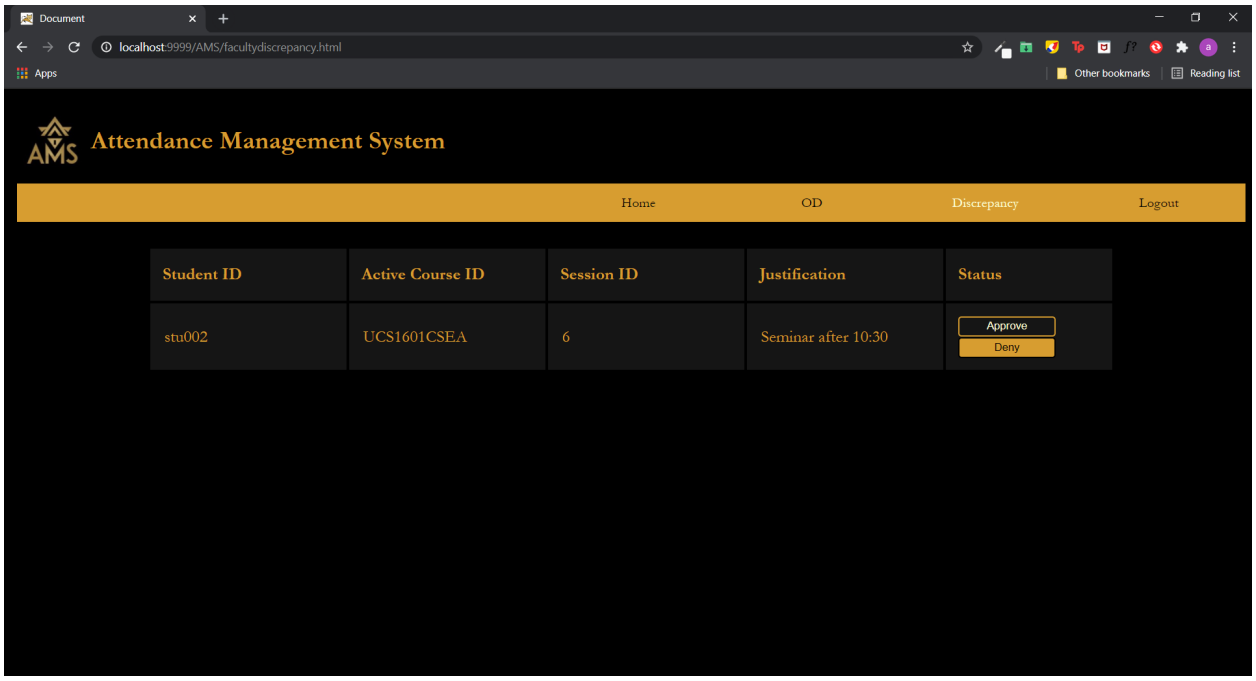
Output : DISCREPANCY Processing

Faculty - Discrepancy Reports:

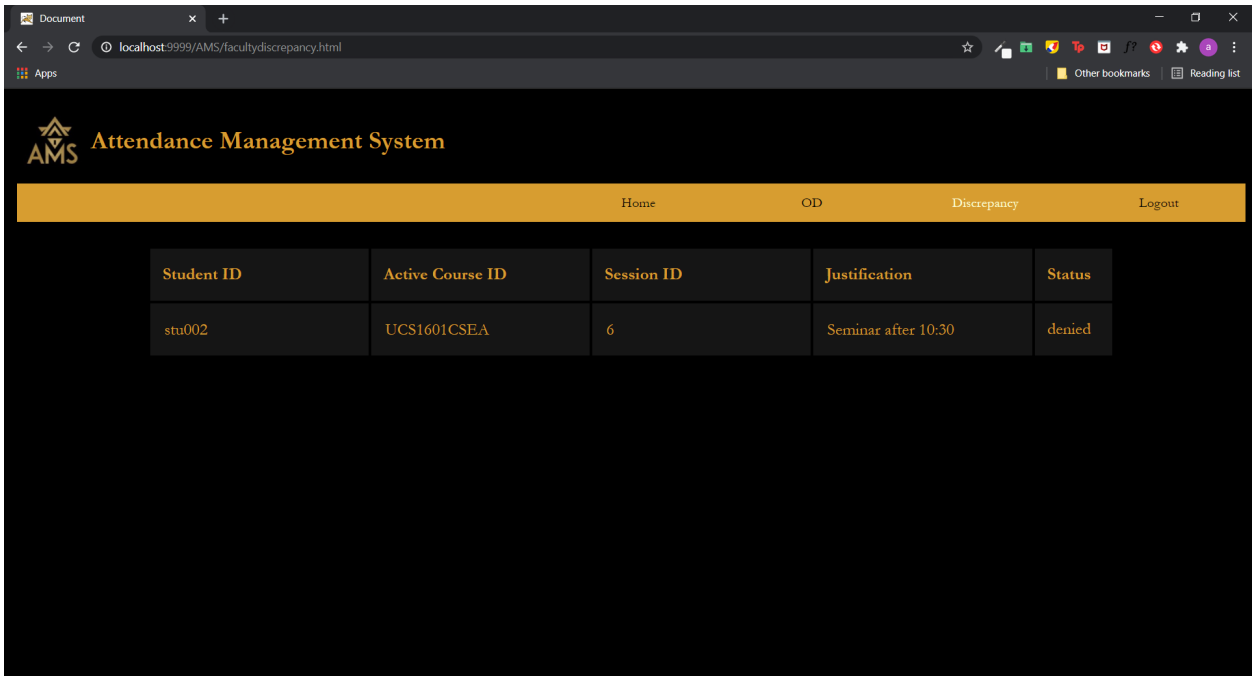




Response to Discrepancy Report:



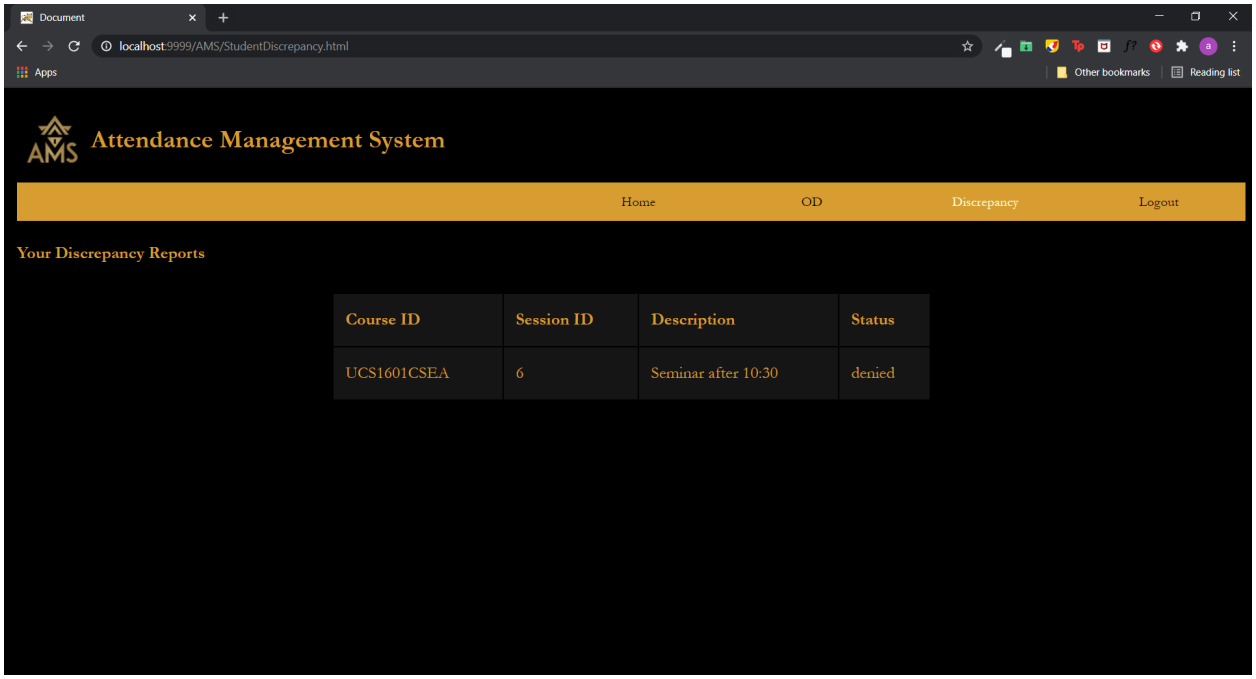
After Responding to Discrepancy Report:



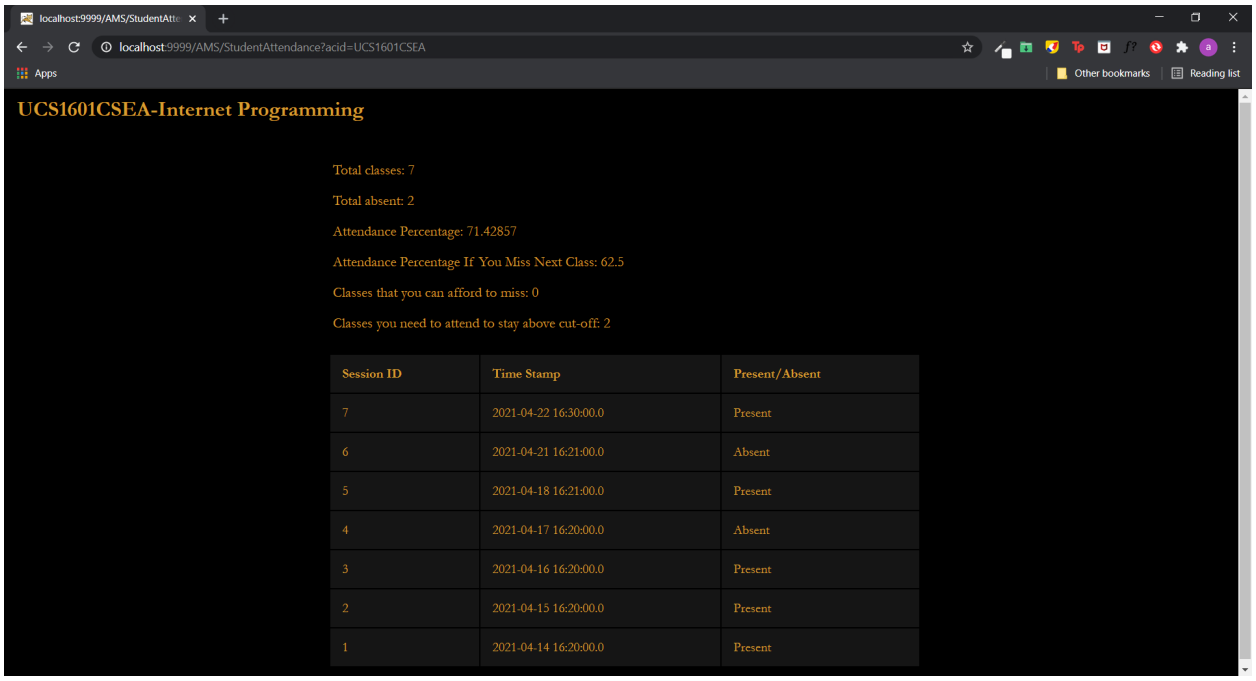
MySQL 8.0 Command Line Client

```
mysql> select * from discrepancy;
+------------+-------------+------------+------------+--------------------+--------+
| session_id | acid        | student_id | faculty_id | justification      | status |
+------------+-------------+------------+------------+--------------------+--------+
|          6 | UCS1601CSEA | stu002     | f001       | Seminar after 10:30 | denied |
+------------+-------------+------------+------------+--------------------+--------+
1 row in set (0.00 sec)

mysql> select * from absentee_list;
+------------+-------------+------------+
| session_id | acid        | student_id |
+------------+-------------+------------+
|          2 | UCS1601CSEA | stu001     |
|          5 | UCS1601CSEA | stu001     |
|          6 | UCS1601CSEA | stu001     |
|          7 | UCS1601CSEA | stu001     |
|          4 | UCS1601CSEA | stu002     |
|          6 | UCS1601CSEA | stu002     |
|          3 | UCS1601CSEA | stu003     |
|          5 | UCS1601CSEA | stu003     |
+------------+-------------+------------+
8 rows in set (0.00 sec)

mysql>
```

Student - Discrepancy Report Status Change

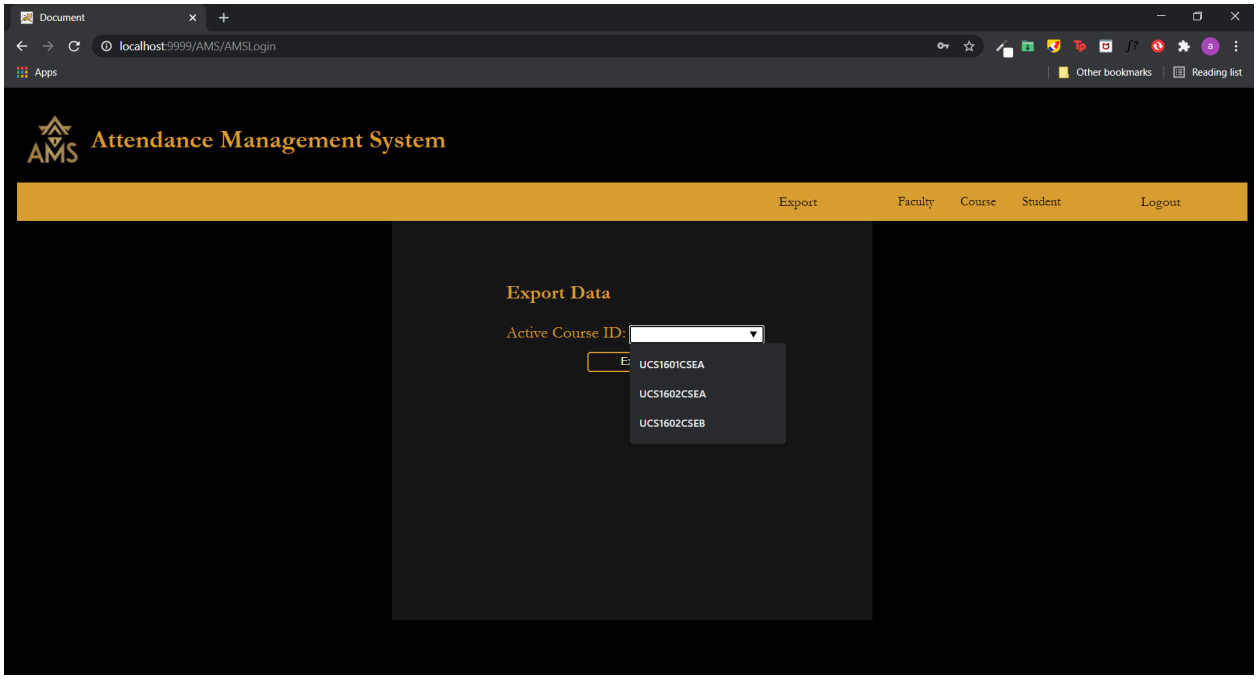


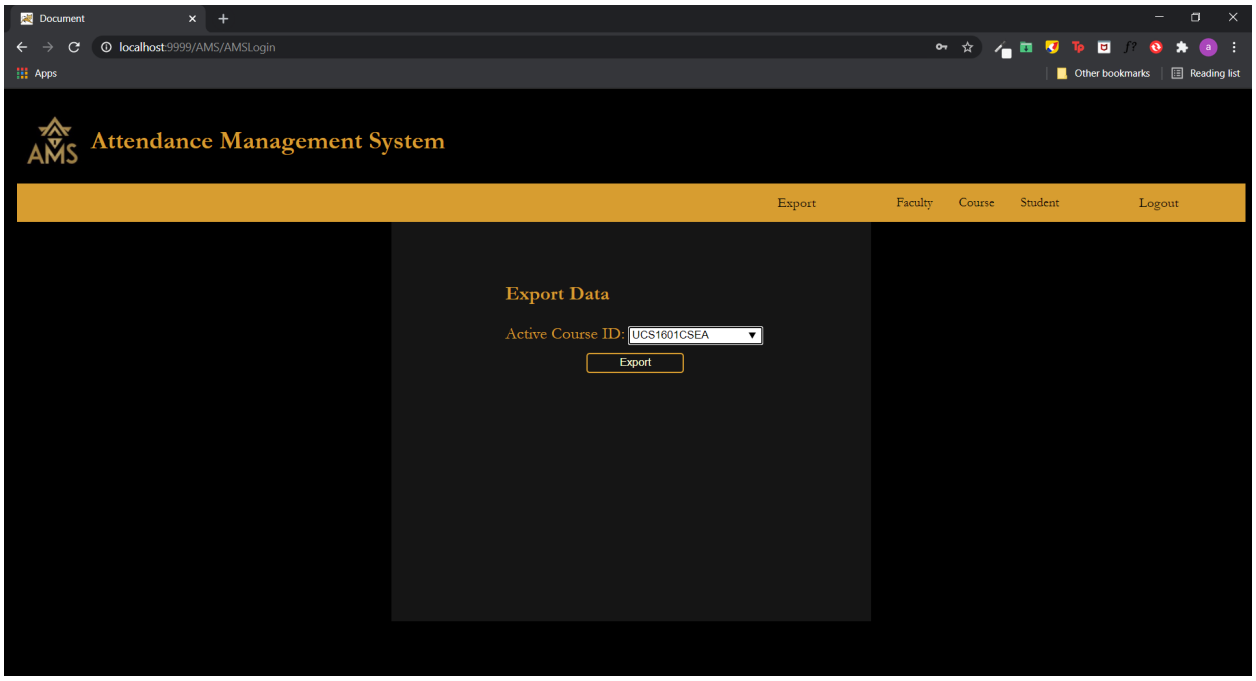
Updated Attendance after denial of Discrepancy:



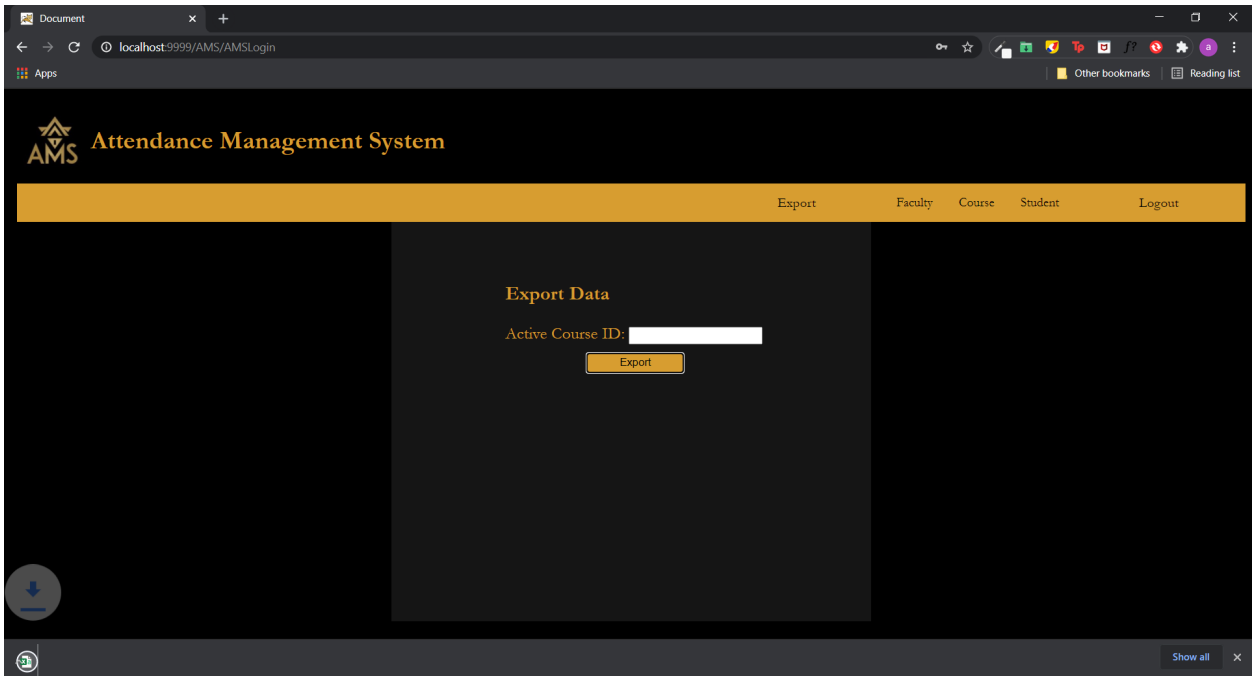
Output : Export Attendance by Admin

Admin Homepage - Export Data:

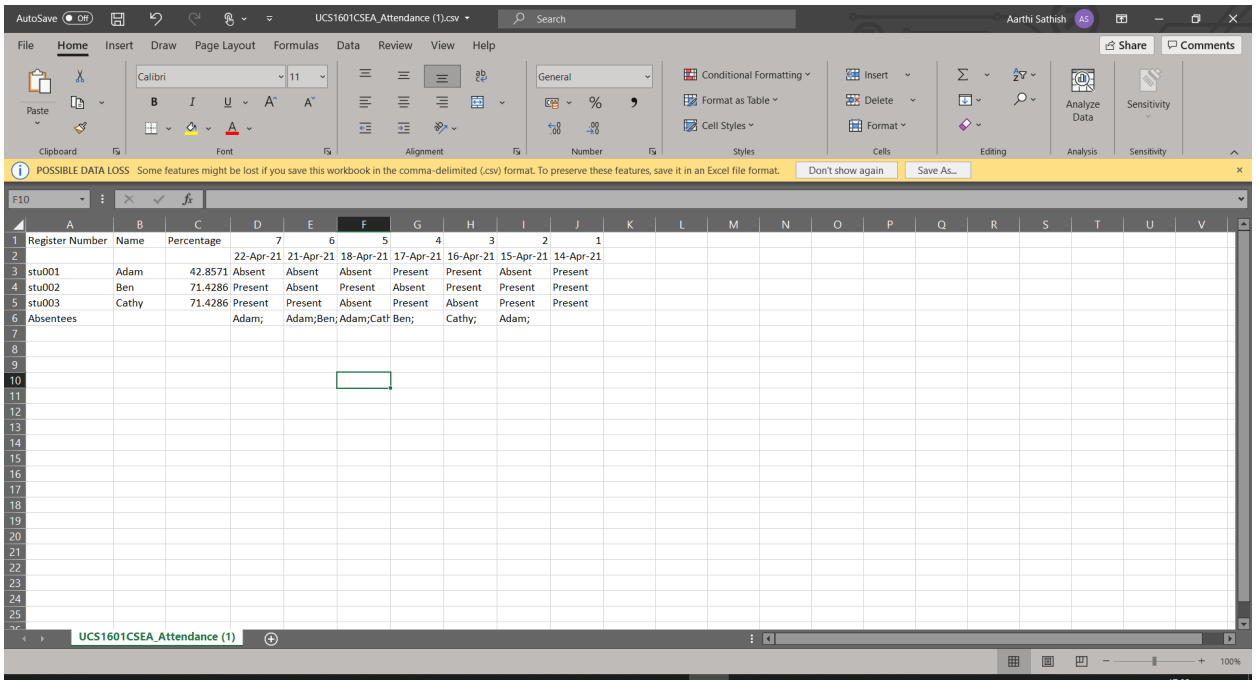




Downloading Course Attendance Data:



Excel Sheet with Course Attendance Details:



Register Number	Name	Percentage	7	6	5	4	3	2	1
			22-Apr-21	21-Apr-21	18-Apr-21	17-Apr-21	16-Apr-21	15-Apr-21	14-Apr-21
stu001	Adam	42.8571	Absent	Absent	Absent	Present	Present	Absent	Present
stu002	Ben	71.4286	Present	Absent	Present	Absent	Present	Present	Present
stu003	Cathy	71.4286	Present	Present	Absent	Present	Absent	Present	Present
Absentees			Adam;	Adam;Ben;	Adam;Cath	Ben;	Cathy;	Adam;	

Documentation:

During the implementation of the AMS, we learnt to use different software elements together and gained an understanding on the different layers that make up the system. We also reflected on the importance of the models that were developed at the earlier stages as they gave us a significant headstart as we ventured into the implementation phase. This phase helped us in understanding the needs of the end user which further helped us create a user-friendly application that has functionalities which always keep the user in mind.

An idea that we wished to implement in our system but were unable to, was to include a calendar in the UI where the schedule of the user (student/faculty) could be updated. One other feature we wanted to experiment with was the automation of the attendance entry. To make the lives of the faculty members even easier, a unique code generated for each session could be shared by the faculty to the students during class. The students would have to enter this code to be marked present for that session. This decreases the possibility of students who weren't present receiving attendance for that session. A small but useful feature that we also wished to add was to give the option of attaching a file as proof while applying for an OD.

Mini Project - Attendance Management System

Test Cases

Aarthi.V.S
Adithya Vikram
Anusha Chandrasekran

Aim:

To develop test cases and perform testing of the Attendance Management System.

Identification of Test Scenarios:

1) Login
2) Take Attendance
3) Update Absentee List
4) View Course Attendance Record
5) Apply for OD
6) Report a Discrepancy
7) Export Attendance
8) Delete a Student Record
9) Add a new Faculty Record
10) Assign Students and Faculty to a Course

Test Cases:

ID	Scenario	Steps	Input	Expected output	Actual output	Pass/fail
1.1	Login	Enter username and password	Valid Credentials	Successful login to homepage	Successful login to homepage	Pass
1.2			Invalid Credentials	Does not login and retry login	Does not login and retry login	Pass
2.1	Take attendance	Enter the timestamp details	Valid format	Accepts input and submits attendance	Accepts input and submits attendance	Pass
2.2			Invalid format	Does not accept attendance	Does not accept attendance	Pass
3.1	Update Absentee List	Enter Absentee Details to be added	Enter valid participant	Absentee is recorded in the database	Absentee is recorded in the database	Pass
3.2			Enter invalid participant(Not part of course)	Absentee is not recorded in the database	Absentee is not recorded in the database	Pass

4.1	View course attendance record	Enter the URL for the course attendance page	Course handled by the faculty	Able to view the course attendance	Able to view the course attendance	Pass
4.2			Course that is not handled by the faculty	Unable to view the course attendance	Unable to view the course attendance	Fail
5.1	Apply for OD	Fill OD Request Form	Enter valid course ID	OD request should be submitted	OD request should be submitted	Pass
5.2			Enter invalid course ID	OD request should not be submitted	OD request should not be submitted	Pass
6.1	Report a discrepancy	Fill Discrepancy Request Form	Fill all required fields	Request is submitted	Request is submitted	Pass
6.2			No justification given	Request is not submitted	Request is not submitted	Pass
7.1	Export attendance	Enter Active Course ID	A valid course ID	Data is exported successfully	Data is exported successfully	Pass
7.2			An invalid course ID	Error displayed	An empty file is created	Fail
8.1	Delete a Student Record	Enter student ID to of student to be deleted	Valid, existing student ID	Student Record is removed from Database	Student Record is removed from Database	Pass
8.2			Non existent student ID	No changes in the database	No changes in the database	Pass
9.1	Add a new Faculty Record	Enter all the required personal details and the their department	Existing Department	Faculty record is added successfully	Faculty record is added successfully	Pass
9.2			Invalid Department	Faculty record is not added	Faculty record is not added	Pass

10.1	Assign students and faculty to a Course	Enter Student IDs and faculty ID of those who have to be assigned to the course.	Student IDs entered with proper formatting (comma separated)	The students are enrolled to the course	The students are enrolled to the course	Pass
10.2			Invalid format for Student IDs entry (commas missing)	No changes in the database	No changes in the database	Pass

Documentation:

The testing process exposed the vulnerabilities of our system. This will be a lesson to take home for the future and to make use of in our future endeavours. Testing has broadened our perspective in considering all possible scenarios during the development of a software system.This process has brought us to the end of the actual implementation of our system.