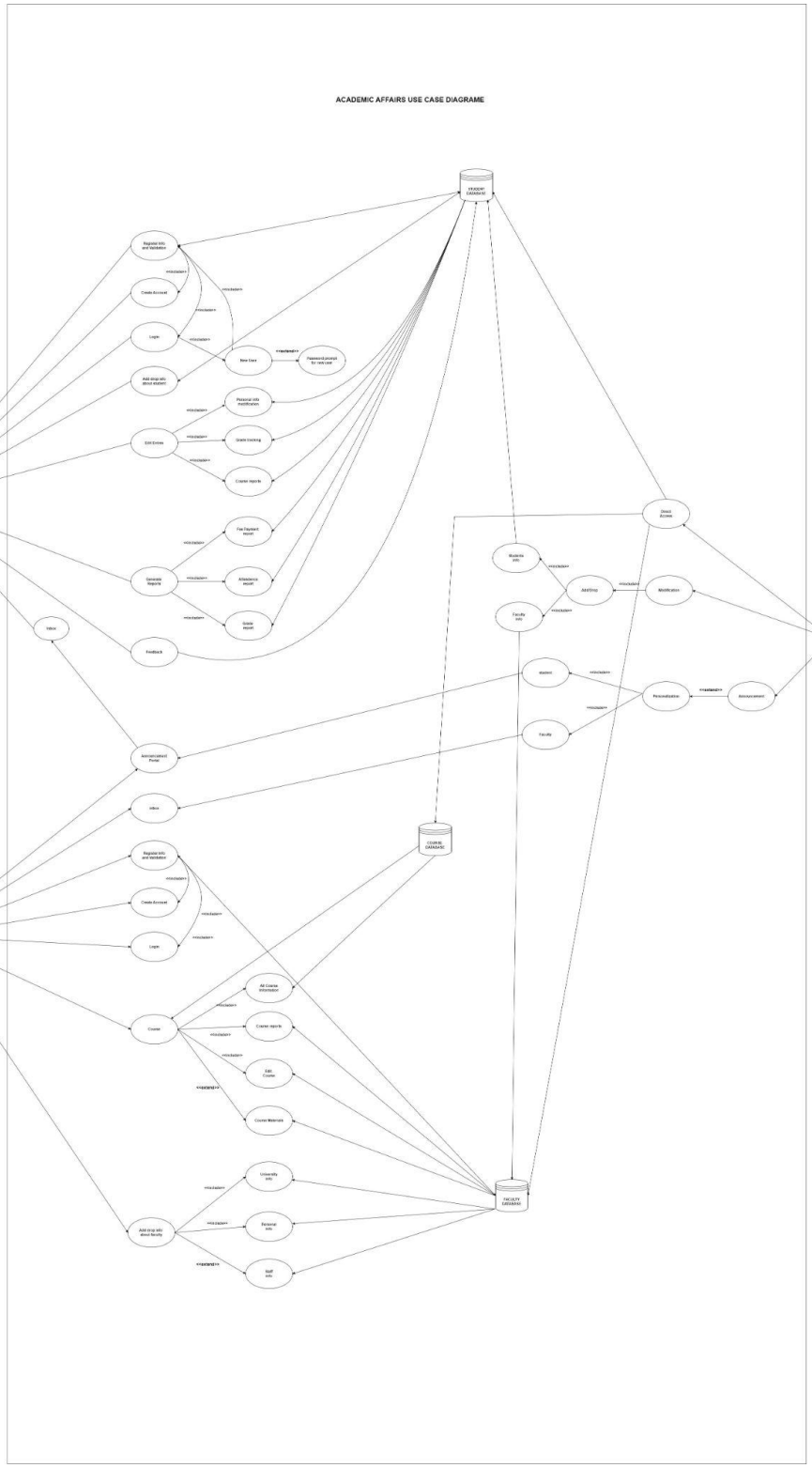
Academic Affairs System - Lab 3

Group Members

1. Abhimanyu Negi - 202001080
2. Sumit Vaniya - 202001085
3. Jayraj rathwa - 202001102
4. Darsh Gopani - 202001065
5. Yash Chauhan - 202001082
6. Shubh Golus - 202001077
7. Nandini Chaudhary - 202001090
8. Kush Shah - 202001104
9. Isha Popat - 202001095

Link for use case diagram :
https://github.com/Abhimanyu013/Group_11_Academic_Affairs/blob/8077b186423cf71f68c5d6b17002fa0c9e935296/use%20case%20diagram.jpg

1.


ACADEMIC AFFAIRS USE CASE DIAGRAME
STUDENT
FACULTY
ADMIN
Register Info and Validation
Create Account
Login
Add drop info about student
New User
Password proved for new user
Personal info modification
Edit Entries
Grade tracking
Course reports
Fee Payment report
Generate Reports
Attendance report
Grade report
Inbox
Feedback
Announcement Portal
Inbox
Register Info and Validation
Create Account
Login
Course
Add Course information
Course reports
Edit Course
Course Materials
Add drop info about faculty
University info
Personal info
Staff info
STUDENT DATABASE
COURSE DATABASE
FACULTY DATABASE
Grant Access
Student info
Add/Drop
Modification
Faculty info
student
Personalization
Announcement
Faculty
«include»
«extend»

2.

Name: Login Page for Student [“Includes” Login]

Actors: Student ,Student DB

Goal: Login and access the Web application

Preconditions:

1. A database of all the users is already existing.
2. The user is already on the login page of website

Basic Flow:

1. The web application displays a button to enter the faculty login/student login/admin login
2. The user chooses the student login option.
3. The web application displays the student login page
    a. Field for username and password is displayed
    b. A Login button is displayed
    c. A button for new users is displayed
4. The user enters the details in the appropriate fields and click on login button
5. The system fetches the information and sends it to the Student Database
6. Student Database validates the information and prompts the system of successful match
7. The system takes the user to the home page of the web application.

Post Conditions: The user is on the homepage of the web application.

Exceptions:

6a. Wrong username or password

6a1. The student database validates the information and prompts the system of unsuccessful match

6a2. The system displays a message to the user about wrong username/password

6a3. The system reverts back to step 3

Name: Login Page for New student [“extends” Login for Student]

Actors: Student ,Student DB

Goal: Login and access the Web application

Preconditions:

1. A database of all the users is already existing.
2. The user is already on the login page of website

Basic Flow:

1. The web application displays a button to enter the login/student login/admin login
2. The user chooses the student login option
3. The web application displays the faculty login page
    a. Field for username and password is displayed
    b. A Login button is displayed
    c. A button for new users is displayed

4. The user clicks the new users button.
5. The webpage redirects the user to the new user login page
    a. It displays the field to enter name, date of birth and registration no.
    b. It displays the field for enter new password and confirm password
    c. It displays the set password button
6. The user enters the details in the fields and clicks on set password
7. The system fetches the information and sends it to the student DB.
8. The student database validates the information of name, date of birth and registration no. and prompts the system of successful match.
9. The student database then modifies the password for the user in the database.
10. The system prompts the user with a message “Password Changed”
11. The system redirects the user to the flow as described in [Login for student]

Postconditions:

1. The password for the user is changed in the student database

Exceptions:

8a. Wrong details

8a1. The faculty database validates the information and prompts the system of unsuccessful match

8a2. The system displays a message to the user about wrong details

8a3. Step 5.

Name: Login Page for Faculty [“Includes” Login]

Actors: Faculty, Faculty DB

Goal: Login and access the Web application

Preconditions:

1. A database of all the users is already existing.
2. The user is already on the login page of website

Basic Flow:

1. The web application displays a button to enter the faculty login/student login/admin login
2. The user chooses the faculty login option
3. The web application displays the faculty login page
    a. Field for username and password is displayed
    b. A Login button is displayed
4. The user enters the details in the appropriate fields and clicks on the login button.
5. The system fetches the information and sends it to the Faculty Database
6. Faculty Database validates the information and prompts the system of successful match
7. The system takes the user to the home page of the web application.

Post Conditions: The user is on the homepage of the web application.

Exceptions:
6a. Wrong username or password

6a1. The faculty database validates the information and prompts the system of unsuccessful match

6a2. The system displays a message to the user about wrong username/password

6a3. The system reverts back to step 3

Name: Login Page for Admin [“Includes” Login]
Actors: Admin, Faculty DB
Goal: Login and access the Web application
Preconditions:

1. A database of all the users is already existing.
2. The user is already on the login page of website
3. Basic Flow:
4. The web application displays a button to enter the login/student login/admin login
5. The user chooses the admin login option.
6. The web application displays the admin login page
    a. Field for username and password is displayed
    b. A Login button is displayed
    c. The user enters the details in the appropriate fields and click on login button
7. The system fetches the information and sends it to the faculty Database
8. Faculty Database validates the information and prompts the system of successful match
9. The system takes the user to the home page of the web application.

Post Conditions: The user is on the homepage of the web application.

Exceptions:
8a. Wrong username or password

8a1. The faculty database validates the information and prompts the system of unsuccessful match

8a2. The system displays a message to the user about wrong username/password

8a3. The system reverts back to step 5

Name: Grade Tracking and Transcript Report

Actors: Faculty, Student

Description: The Grade Tracking and Transcript Report use case involves the process of recording and managing student grades and generating transcript reports. Faculty members

input student grades for each course they teach. Students can access their transcript report, which shows their grades and CGPA.

Preconditions:

1. The faculty member and student must have an active account in the academic affairs system.
2. The student must have completed at least one course.

Basic Flow:

1. The faculty member inputs student grades for each course they teach.
2. The academic affairs system validates the grades and updates the student's academic record.
3. The student accesses their transcript report through the academic affairs system.
4. The academic affairs system generates the transcript report based on the student's academic record.
5. The transcript report is displayed to the student, showing their grades for each course, cumulative GPA, and any academic standing (e.g., honors, probation, etc.).

Alternative Flow:

1a. If the faculty member inputs incorrect grades, the academic affairs system prompts them to review and correct the grades.
3a. If the student identifies any discrepancies or errors in their transcript report, they can contact the registrar to request a review and correction of their academic record.

Postconditions:

1. The student's grades are recorded and updated in the academic affairs system.
2. The student can access their transcript report to review their grades and academic standing.

Use Case: Fees Receipt Report

Actors: StudentDb, Student

Description: The Fees Receipt Report use case involves the process of recording and managing student fees payments and generating fees receipt reports. Admin input fees payment details for each student in the Student database, and the system maintains a record of all payments for each student. Students can access their fees receipt report, which shows their payment details and any outstanding fees.

Preconditions:

1. The student must have an active account in the academic affairs system.
2. The student must have made at least one fee payment.

Basic Flow:

1. The student makes a payment for fees.
2. The academic affairs system validates the payment details and updates the student's financial record.
3. The student accesses their fees receipt report through the academic affairs system.
4. The academic affairs system generates the fees receipt report based on the student's financial record.
5. The fees receipt report is displayed to the student, showing their payment details, outstanding fees, and any applicable late fees or penalties.

Alternative Flow:

5a. If the student identifies any discrepancies or errors in their fees receipt report, they can contact the admin to request a review and correction of their financial record.

Postconditions:

1. The student's fees payments are recorded and updated in the academic affairs system.
2. The student can access their fees receipt report to review their payment details and outstanding fees.

Use Case: Course Feedback
Description: The user wants to comment on the service of the software
Primary Actor: User
Triggering Event: Receive feedbacks from the user to improve the quality of the system
Basic Flow:

1. User logs in the system.

2. System authenticates the user.
3. User selects the course to give feedback.
4. User selects the option “Give feedbacks”
5. The system displays a blank text box to enter the message.
6. The user enters his message
7. System acknowledges that the feedback is sent

Alternate Path:

1. The authentication fails
    a. `System informs the user and does not allow the user to proceed
2. The system fails to submit the feedback
    a. System informs the user

Preconditions:

1. Users are registered.

Postconditions:

The feedback is submitted

Use Case: Edit student profile

Description: The user wants to update/modify his personal information

Primary Actor: User

Basic Flow:

1. The user selects the option “Modify” under his profile.
2. The system displays the user’s profile information and prompts him to update it.
3. The user modifies or updates the required information
4. The system validates the information
5. System prompts the user for a confirmation
6. User confirms the same
7. The system updates the profile in the data repository
8. System informs the user that the data has been updated successfully.

Preconditions:

1. User is registered

Postcondition:

1. The user details are changed in the database

Use case: Attendance Tracker

Description: The staff needs to track the attendance of the students

Actor: Student

Basic Flow:

1. The user logs in the system
2. System authenticates the user
3. The user opens his student profile
4. The system displays the student profile along with the courses registered
5. The student selects a course to check his attendance
6. The system displays his attendance along with the other course logistics

Alternate Path:

1.The authentication fails

a. `System informs the user and does not allow the user to proceed

2. The system fails to submit the feedback

b. System informs the user

Preconditions:

1. Users are registered.

Use Case Name: Broadcast Message to Students

Actors: Admin

Preconditions:

1)There are students enrolled in the system

Description:

1.Admin selects the "Broadcast Message" option from the dashboard.
2.System displays the "Broadcast Message" form, which includes:
3.A text area to enter the message
4..Admin enters the message in the text area.
5.Admin clicks the "Send" button.
6.System validates that the message is not empty..
7.System displays a confirmation message to the admin indicating that the message has been sent.

Alternate Flow:
6a. If the message is empty, the system displays an error message and prompts the admin to enter a message.

Postconditions:

All students in the system receive the broadcasted message via the selected delivery method.

Use Case Name: Course Reports(Faculty)

Actors: Faculty

Preconditions:

Faculty is authenticated and authorized to view course reports
At least one course has been taught by the faculty in the system

Description:
1.Faculty selects the "Course Reports" option from the dashboard.
2.System displays a list of courses taught by the faculty.
3.Faculty selects a course from the list.
4.System displays the course report for the selected course, which includes:
5.A summary of the course, such as course code, title, and semester.
6.A list of students enrolled in the course, along with their grades and any feedback provided.
7.Faculty reviews the course report.
8.Faculty selects the option to download or print the course report, if desired.
9.Faculty closes the course report and returns to the list of courses.

Alternate Flow:
4a. If the selected course does not have a course report generated, the system displays a message indicating that no report is available.

Postconditions:

Faculty has reviewed the course report for the selected course and can download or print it if desired.

Use Case Name: Maintain Course Information and Content

Actors: Faculty

Preconditions:

Faculty is authenticated and authorized to maintain course information and content
At least one course has been created in the system

Description:

1.Faculty selects a course from the list of courses in the dashboard.
2.System displays the course information and content.
3.Faculty selects the option to edit course content.
4.System displays the course content editor.
5.Faculty updates the course content
6.Faculty saves the changes and reviews the updated course content.
7.Faculty selects the option to publish the updated course content, if desired.
8.Faculty closes the course content editor and returns to the homepage.

Alternate Flow:
5a. If the faculty cancels the editing process, the system discards any changes made and returns the faculty to the course information and content display.

Postconditions:

Course information and content has been updated and published, if desired, and can be accessed by students in the course.

Use Case Name: Course Report (student)

Actors: Student

Preconditions:

Student is authenticated and authorized to view course reports
Student is enrolled in at least one course in the system

Description:

1.Students select the "Course Reports" option from the dashboard.
2.System displays a list of courses in which the student is currently enrolled.
3.Student selects a course from the list.
4.System displays the course report for the selected course, which includes:
5.A summary of the course, such as course code, title, and semester.
6.The student's grades in the course and the average grade
7.His attendance is also displayed
8.Student reviews the course report.
9.Student selects the option to download or print the course report, if desired.

10.Student closes the course report and returns to the list of courses.

Alternate Flow:
4a. If the selected course does not have a course report generated, the system displays a message indicating that no report is available.

Postconditions:

Students have reviewed the course report for the selected course and can download or print it if desired.

Name: Add/Drop Information about Students into the database

Actor: student, admin

Goal: To add or drop information about students.

Precondition: The user type must be an admin to add a new student in the system or add & drop information of the existing student. For a new student, the student must be eligible for university admission and must fulfill the university's requirements to get access.

Flow:
1. Check whether the student is eligible to get admission to college.
2. Provide a form to the student to get the information about a student.
3. Check whether the student fills out the form correctly.
4. Enter the data into academic affairs through an interface.
5. Save the information of the student and the student who is registered.
6. Edit the existing information and update if any change is made.
7. Admin can Delete the existing records.

Postcondition: updated the information of the student.

Alternative flow:
3a. If the information provided by the student is not completed or not relevant to the requirement then the student cannot submit the form and asks for correct information.

Name: Add/Drop Information about Faculty into the database.

Actor: faculty, admin

Goal: to add or drop faculty.

Precondition: The user type must be an admin to add or drop Information about Faculty into the database. For a new faculty, the faculty must be eligible for university and must fulfill the university's requirements to get access.

Flow:

1. Check whether the faculty is eligible for the university.
2. Provide a form to the faculty to get the information about the faculty.
3. Check whether the faulty fill out the form correctly.
4. Enter the data into academic affairs through an interface.
5. Save the information of the faculty and the faculty who are registered.
6. Edit the existing information and update if any change is made.
7. Admin can Delete the existing records.

Postcondition: updated the information of the faculty.

Alternative flow:
3a. If the information provided by the faculty is not complete or not relevant to the requirement then the faculty cannot submit the form and asks for correct information.

3.

Non-Functional Requirements

Accessibility and Ease of Use

- A Web application is required for the system with simple and interactive user experience for better usability.

Performance

- The system should be able to perform all the desired functions in a very short period of time.
- The system should be able to handle a large number of simultaneous users.
- It should be able to handle and process multiple requests simultaneously.

Reliability

- In case of a server failure, the system should recover in a short span of time without losing the previously held data.
- The System should perform all the operations as expected from them with great accuracy.

Security

- The system should validate if the user is a member of the system or not as it contains personal as well as academic information.

Scalability

- The system should be able to handle a large number of data and it should be able to maintain a lossless database.

Maintainability

- It should be easy for the administrator to maintain the system as it will contain a lot of database entries.