EXPERIMENT 4

AIM: To write Use case description for the case study “University Registration System”.

DESCRIPTION:

Use case description is the detailed explanation of all the use cases defined in the use case diagram. It defines the flow and function of the use case: when will the use case be active, when will it end, what are the necessary conditions to be fulfilled for its functioning.

The following is the template for use case description:

i. **Introduction**: Briefly describe the use case.

ii. **Actors**: List actors that participate and interact with this use case.

iii. **Pre Conditions:** List pre-conditions here. List the system state/conditions which must be true before this Use Case can be executed.

iv. **Post-Conditions:** Post-conditions on a use case lists possible states that the system can be in at the end of the use case execution. The system must be in one of those states. A post-condition also states actions that the system performs at the end of the use case, regardless of what occurred in the use case.

v. **Flow of events:**

Basic flow: primary events on use case execution

Alternate flow: Any other possible flow.

vi. **Special Requirements:** Enter any special requirements such as Performance requirements, Security requirements, and user interface requirements.

vii. **Related use cases:** List the related use cases if any.

The use case description should be in a box.

STEPS TO BE FOLLOWED:

i. Determine all the use cases.
ii. Take each use case one by one and determine their functionalities and constraints.
iii. Determine all the requirements for each use case.
iv. Write the description for each use case in the given standard format.
v. Build a box around all the use case descriptions.

RESULT:

1. LOGIN

Introduction: The use case documents the steps that must be followed in order to login into the system.

Actors: Administrator, Student, Faculty, DEO

Pre-Condition: The administrator/ student/ faculty/DEO must be registered onto the system before the use case begins.

Post Conditions: If the use case is successful, the administrator/ student/ faculty is logged into the system else the system state remains unchanged.

Event Flow

Basic Flow

1. The system requests that the actor enter his username, password. The role of the user is determined based on the username.
2. The Administrator/Student/Faculty enters his/her username, password.
3. The system validates the entered credentials and logs the actor into the system.

Alternative Flow

Alternative Flow 1: Invalid Login Details

If in the basic flow, the actor enters an invalid username, password the system displays an error message. The user can choose to either return to the beginning of the basic flow or cancel the login at which point the Use Case ends.

Alternative Flow 2: User Exits

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

Special Requirement: None

Associated Use Case: None

2. MAINTAIN STUDENT DETAILS

Introduction: The use case documents the steps that the administrator must follow in order to maintain student details and add, update, view and delete student information.

Actors: Administrator, DEO

Pre-Condition: The Administrator must be logged into the system before this use case begins.

Post Conditions: If the use case is successful, then the student information is added, updated, deleted or viewed else the system state remains unchanged.

Event Flow

Basic Flow:

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

1. The system requests that the administrator specify the function he/she would like to perform (add a student, update a student, view a student or delete a student).
2. Once the administrator provides the requested information, one of the sub flows is executed:
   a. If the administrator selects "Add a Student", the **Add a Student** subflow is executed.
   b. If the administrator selects "Update a Student", the **Update a Student** subflow is executed.
   c. If the administrator selects "Delete a Student", the **Delete a Student** subflow is executed.
   d. If the administrator selects "View a Student", the **View a Student** subflow is executed.

Basic Flow 1: Add a Student

The system request that the administrator enter the student information. This includes:

- Registration Number
- Roll Number
- Student name (First Name, Last Name)
- Photograph
- Date of Birth
- Mobile Number
- Email ID
- Branch
- Address
- Category
- Guardian Name
- Guardian Mobile Number

Once the administrator provides the requested information, the student is added to the system.

Basic Flow 2: Update a Student

1. The system requests that the administrator enter Registration Number.
2. The administrator enters the Registration Number.
3. The system retrieves and displays the student information.
4. The administrator makes the desired changes to the student information.

5. Once the administrator updates the necessary information, the system updates the student information with the updated information.

Basic Flow 3: Delete a Student

1. The system requests that the administrator enter the Registration Number.
2. The administrator enters the Registration Number.
3. The system retrieves and displays the student information.
4. The system prompts the administrator to confirm the deletion of the student record.
5. The administrator verifies the deletion.
6. The system deletes the record.

Basic Flow 4: View a Student

1. The system requests that the administrator to enter the Registration Number.
2. The system retrieves and displays the faculty information.

Alternative Flow

Alternate Flow 1: Invalid Entry
If in the **Add a Student** or **Update a Student** flow, the administrator enters invalid Registration Number/ Roll Number/ Student Name/ Date of Birth/ Mobile Number/ Email ID/ Address/ Branch/ Category/ Guardian Details or leaves them empty, the system displays an appropriate error message. The administrator returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: Student Already Exists
If in the **Add a Student** flow, a student with a specified student ID already exists, the system displays an error message. The administrator returns to the basic flow and may re-enter the Registration Number.

Alternate Flow 3: Student not Found
If in the **Update a Student** or **Delete a Student** or **View a Student** flow, the student information with the specified code does not exist, the system displays an error message. The administrator returns to the basic flow and may re-enter the Registration Number.

Alternate Flow 4: Update Cancelled
If in the **Update a Student** flow, the administrator decides not to update the student, the update is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 5: Delete Cancelled
If in the **Delete a Student** flow, the administrator decides not to delete the student, the delete is cancelled and the basic flow is re-started at the beginning.

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

Special Requirement: None

Associated Use Case: Login

3. MAINTAIN FACULTY DETAILS

Introduction: The use case documents the steps that the administrator must follow in order to maintain faculty details and add, update, view and delete faculty information.

Actors: Administrator

Pre-Condition: The Administrator must be logged into the system before this use case begins.

Post Conditions: If the use case is successful, then the faculty information is added, updated, deleted or viewed else the system state remains unchanged.

Event Flow

Basic Flow:

This use case starts when the administrator wishes to add/update/delete/view faculty information.

1. The system requests that the administrator specify the function he/she would like to perform (add a faculty, update a faculty, view a faculty or delete a faculty).
2. Once the administrator provides the requested information, one of the sub flows is executed:
   a. If the administrator selects "Add a Faculty", the **Add a Faculty** subflow is executed.
   b. If the administrator selects "Update a Faculty", the **Update a Faculty** subflow is executed.
   c. If the administrator selects "Delete a Faculty", the **Delete a Faculty** subflow is executed.
   d. If the administrator selects "View a Faculty", the **View a Faculty** subflow is executed.

Basic Flow 1: Add a Faculty

The system request that the administrator enter the faculty information. This includes:

- Faculty ID
- Faculty name (First Name, Last Name)
- Photograph
- Date of Birth
- Mobile Number
- Email ID
- Address
- Joining Date
- Education Qualification
- Associated Department

Once the administrator provides the requested information, the faculty is added to the system.

Basic Flow 2: Update a Faculty

1. The system requests that the administrator enter the faculty ID.
2. The administrator enters the faculty ID.
3. The system retrieves and displays the faculty information.
4. The administrator makes the desired changes to the faculty information.
5. Once the administrator updates the necessary information, the system updates the faculty information with the updated information.

Basic Flow 3: Delete a Faculty

1. The system requests that the administrator enter the faculty ID.
2. The administrator enters the faculty ID.

3. The system retrieves and displays the faculty information.
4. The system prompts the administrator to confirm the deletion of the faculty record.
5. The administrator verifies the deletion.
6. The system deletes the record.

Basic Flow 4: View a Faculty

1. The system requests that the administrator to enter the faculty ID.
2. The system retrieves and displays the faculty information.

Alternative Flow

Alternate Flow 1: Invalid Entry
If in the **Add a Faculty** or **Update a Faculty** flow, the administrator enters invalid Faculty ID/ Name/ Department/ Education Qualification/ Date of Birth/ Mobile Number/ Email ID/ Address/ Joining Date or leaves them empty, the system displays an appropriate error message. The administrator returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: Faculty Already Exists
If in the **Add a Faculty** flow, a faculty with a specified faculty ID already exists, the system displays an error message. The administrator returns to the basic flow and may re-enter the faculty code.

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

Alternate Flow 4: Update Cancelled
If in the **Update a Faculty** flow, the administrator decides not to update the faculty, the update is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 5: Delete Cancelled
If in the **Delete a Faculty** flow, the administrator decides not to delete the faculty, the delete is cancelled and the basic flow is re-started at the beginning.

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

Special Requirement: None

Associated Use Case: Login

4. MAINTAIN DEPARTMENT DETAILS

Introduction: The use case documents the steps that the administrator must follow in order to maintain department details and add, update, view and delete department information.

Actors: Administrator, DEO

Pre-Condition: The Administrator must be logged into the system before this use case begins.

Post Conditions: If the use case is successful, then the department information is added, updated, deleted or viewed else the system state remains unchanged.

Event Flow

Basic Flow:

This use case starts when the administrator wishes to add/update/delete/view department information.

1. The system requests that the administrator specify the function he/she would like to perform (add a department , update a department , view a department or delete a department).
2. Once the administrator provides the requested information, one of the sub flows is executed:
    a. If the administrator selects "Add a Department", the **Add a Department** subflow is executed.
    b. If the administrator selects "Update a Department", the **Update a Department** subflow is executed.
    c. If the administrator selects "Delete a Department", the **Delete a Department** subflow is executed.
    d. If the administrator selects "View a Department", the **View a Department** subflow is executed.

Basic Flow 1: Add a Department

The system request that the administrator enter the department information. This includes:

- Department ID
- Department name
- Department Head

Once the administrator provides the requested information, the department is added to the system.

Basic Flow 2: Update a Department

1. The system requests that the administrator enter the Department ID.
2. The administrator enters the Department ID.
3. The system retrieves and displays the department information.
4. The administrator makes the desired changes to the department information.
5. Once the administrator updates the necessary information, the system updates the department information with the updated information.

Basic Flow 3: Delete a Department

1. The system requests that the administrator enter the Department ID.
2. The administrator enters the Department ID.
3. The system retrieves and displays the department information.
4. The system prompts the administrator to confirm the deletion of the department record.

5. The administrator verifies the deletion.
6. The system deletes the record.

Basic Flow 4: View a Department

1. The system requests that the administrator to enter the Department ID.
2. The system retrieves and displays the department information.

Alternative Flow

Alternate Flow 1: Invalid Entry
If in the **Add a Department** or **Update a Department** flow, the administrator enters invalid Department ID/ Department Name/ Department Head or leaves them empty, the system displays an appropriate error message. The administrator returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: Department Already Exists
If in the **Add a Department** flow, a department with a specified Department ID already exists, the system displays an error message. The administrator returns to the basic flow and may re-enter the Department ID.

Alternate Flow 3: Department not Found
If in the **Update a Department** or **Delete a Department** or **View a Department** flow, the department information with the specified code does not exist, the system displays an error message. The administrator returns to the basic flow and may re-enter the Department ID.

Alternate Flow 4: Update Cancelled
If in the **Update a Department** flow, the administrator decides not to update the department , the update is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 5: Delete Cancelled
If in the **Delete a Department** flow, the administrator decides not to delete the department , the delete is cancelled and the basic flow is re-started at the beginning.

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

Special Requirement: None

Associated Use Case: Login

5. MAINTAIN COURSE DETAILS

Introduction: The use case documents the steps that the administrator must follow in order to maintain course details and add, update, view and delete course information.

Actors: Administrator, DEO

Pre-Condition: The Administrator must be logged into the system before this use case begins.

Post Conditions: If the use case is successful, then the course information is added, updated, deleted or viewed else the system state remains unchanged.

Event Flow

Basic Flow:

This use case starts when the administrator wishes to add/update/delete/view course information.

1. The system requests that the administrator specify the function he/she would like to perform (add a course , update a course , view a course or delete a course).
2. Once the administrator provides the requested information, one of the sub flows is executed:
    a. If the administrator selects “Add a Course ”, the **Add a Course** subflow is executed.
    b. If the administrator selects “Update a Course ”, the **Update a Course** subflow is executed.
    c. If the administrator selects “Delete a Course ”, the **Delete a Course** subflow is executed.
    d. If the administrator selects “View a Course ”, the **View a Course** subflow is executed.

Basic Flow 1: Add a Course

The system request that the administrator enter the course information. This includes:

- Course Code
- Course Name
- Syllabus
- Associated Department

Once the administrator provides the requested information, the course is added to the system.

Basic Flow 2: Update a Course

1. The system requests that the administrator enter the Course Code.
2. The administrator enters the Course Code.
3. The system retrieves and displays the course information.
4. The administrator makes the desired changes to the course information.
5. Once the administrator updates the necessary information, the system updates the course information with the updated information.

Basic Flow 3: Delete a Course

1. The system requests that the administrator enter the Course Code.
2. The administrator enters the Course Code.
3. The system retrieves and displays the course information.
4. The system prompts the administrator to confirm the deletion of the course record.
5. The administrator verifies the deletion.

6. The system deletes the record.

Basic Flow 4: View a Course

1. The system requests that the administrator to enter the Course Code.
2. The system retrieves and displays the course information.

Alternative Flow

Alternate Flow 1: Invalid Entry

If in the **Add a Course** or **Update a Course** flow, the administrator enters invalid Course Code/ Course Name/ Associated Department or leaves them empty, the system displays an appropriate error message. The administrator returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: Course Already Exists

If in the **Add a Course** flow, a course with a specified Course Code already exists, the system displays an error message. The administrator returns to the basic flow and may re-enter the Course Code.

Alternate Flow 3: Course not Found

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

Alternate Flow 4: Update Cancelled

If in the **Update a Course** flow, the administrator decides not to update the course, the update is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 5: Delete Cancelled

If in the **Delete a Course** flow, the administrator decides not to delete the course, the delete is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 6: User Exits

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

Special Requirement: None

Associated Use Case: Login

6. MAINTAIN SCHEME DETAILS

Introduction: The use case documents the steps that the administrator must follow in order to maintain scheme details and add, update, view and delete scheme information.

Actors: Administrator, DEO

Pre-Condition: The Administrator must be logged into the system before this use case begins.

Post Conditions: If the use case is successful, then the scheme information is added, updated, deleted or viewed else the system state remains unchanged.

Event Flow

Basic Flow:

This use case starts when the administrator wishes to add/update/delete/view scheme information.

1. The system requests that the administrator specify the function he/she would like to perform (add a scheme, update a scheme, view a scheme or delete a scheme).
2. Once the administrator provides the requested information, one of the sub flows is executed:
    a. If the administrator selects “Add a Scheme”, the **Add a Scheme** subflow is executed.
    b. If the administrator selects “Update a Scheme”, the **Update a Scheme** subflow is executed.
    c. If the administrator selects “Delete a Scheme”, the **Delete a Scheme** subflow is executed.
    d. If the administrator selects “View a Scheme”, the **View a Scheme** subflow is executed.

Basic Flow 1: Add a Scheme

The system request that the administrator enter the scheme information. This includes:

- Credits
- Associated Course Code
- Contact Hours / Week Details (Lecture, Theory, Internals)
- Relative Weights (Theory, Practical, Internals)

Once the administrator provides the requested information, the scheme is added to the system.

Basic Flow 2: Update a Scheme

1. The system requests that the administrator enter Course Code
2. The administrator enters the Course Code.
3. The system retrieves and displays the scheme information.
4. The administrator makes the desired changes to the scheme information.
5. Once the administrator updates the necessary information, the system updates the scheme information with the updated information.

Basic Flow 3: Delete a Scheme

1. The system requests that the administrator enter the Course Code.
2. The administrator enters the Course Code.
3. The system retrieves and displays the scheme information.
4. The system prompts the administrator to confirm the deletion of the scheme record.
5. The administrator verifies the deletion.

6. The system deletes the record.

Basic Flow 4: View a Scheme

1. The system requests that the administrator to enter the Course Code.
2. The system retrieves and displays the scheme information.

Alternative Flow

Alternate Flow 1: Invalid Entry

If in the **Add a Scheme** or **Update a Scheme** flow, the administrator enters invalid Credits/ Associated Course Code/ Contact Hours/ Relative Weights or leaves them empty, the system displays an appropriate error message. The administrator returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: Scheme Already Exists

If in the **Add a Scheme** flow, a scheme with a specified Course Code already exists, the system displays an error message. The administrator returns to the basic flow and may re-enter the Course Code.

Alternate Flow 3: Scheme not Found

If in the **Update a Scheme** or **Delete a Scheme** or **View a Scheme** flow, the scheme information with the specified code does not exist, the system displays an error message. The administrator returns to the basic flow and may re-enter the Course Code.

Alternate Flow 4: Update Cancelled

If in the **Update a Scheme** flow, the administrator decides not to update the scheme, the update is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 5: Delete Cancelled

If in the **Delete a Scheme** flow, the administrator decides not to delete the scheme, the delete is cancelled and the basic flow is re-started at the beginning.

Alternate Flow 6: User Exits

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

Special Requirement: None

Associated Use Case: Login

7. SEARCH DEPARTMENT

Introduction: This use case documents the steps that must be followed to search department.
Actors: Administrator, Student, Faculty
Pre-Condition: The Administrator/Student/Faculty must be logged into the system before this use case begins.
Post Conditions: If the use case is successful, the department details are displayed else an error message is displayed.
Event Flow **Basic Flow:** This use case starts when the Administrator/Student/Faculty wishes to view details of a department. 1. The Administrator/Student/Faculty selects a department from the list. 2. The Department details are displayed.
Alternative Flow **Alternative Flow 1: Department Details are not available** If the Department details are not available, an error message is shown. The user can either choose to return to the basic flow of the use case or the use case ends. **Alternative Flow 2: User Exits** If at any point of time, the Administrator/Student/Faculty decides not to view the department details, the use case ends.
Special Requirement: None
Associated Use Case: Login

8. VIEW TIMETABLE

Introduction: This use case documents the steps that must be followed to view time table.

Actors: Administrator, Student, Faculty

Pre-Condition: The Administrator/Student/Faculty must be logged into the system before this use case begins..

Post Conditions: If the use case is successful, the department details are displayed else an error message is displayed.

Event Flow

Basic Flow:

This use case starts when the Administrator/Student/Faculty wishes to view the timetable.

1. The Administrator/Student/Faculty selects a department and section from the list and the time table is displayed.
2. The Faculty can directly view his/her time table.

Alternative Flow

Alternative Flow 1: Time Table is not available

If the Time Table of the selected department is not available, an error message is shown. The user can either choose to return to the basic flow of the use case or the use case ends.

Alternative Flow 2: User Exits

If at any point of time, the Administrator/Student/Faculty decides not to view the timetable, the use case ends.

Special Requirement: None

Associated Use Case: Login

9. COURSE SELECTION

Introduction: This use case documents the steps that must be followed for a student to select course.

Actors: Administrator, Student

Pre-condition: The Administrator/Student must be logged into the system.

Post-condition: If the use case is successful, the student course details are updated in the system else the system state remains unchanged.

Basic flow:

1. The student selects the courses allotted to the department of the student.
2. Submit button is clicked and the courses are updated in the system.

Alternate flow 1: No Seats Available

1. If the seats corresponding to any course selected is full, a popup message is shown containing the name of the course having no seats. The user returns to the basic flow of the use case.

Alternate flow 2: User Exits

1. The user decides to cancel the course selection procedure and exits from the system. The use case ends.

Special Requirements: None

Associated use cases: Login

10. PAY REGISTRATION FEES

Brief Description: This use case documents the steps that must be followed to pay the registration fees.
Actors: Administrator, Student
Pre-Conditions : The Administrator/Student must be logged into the system
Post Conditions: If the use case is successful, the Administrator/Student is able to pay the registration fees, if any.
Flow of Events **Basic Flow** 1. The Student/Administrator selects the option to pay the registration fees. 2. The system displays the registration fees of the student inclusive of all taxes as the case may be. 3. The Student pays the bill using online modes of payment- Credit Card/Debit Card/Net Banking
Alternative Flow 1: Invalid Details If in the basic flow, the Student/Administrator enters an invalid information the system displays an error message. The user can choose to either return to the beginning of the basic flow or cancel the payment process at which point the Use Case ends. **Alternative Flow 2: User Exits** This allows the user to exit at any time during the use case. The use case ends
Special Requirements: None
Associated Use Case: Login, Receipt Generation

11. RECEIPT GENERATION

Introduction: This use case generates a receipt after successful payment by the user.
Actors: Administrator, Student
Pre-condition: The customer must be logged into the system. A successful payment should be made
Post-condition: A receipt is generated and sent to the student.
Basic flow: 1. After successful payment by the user an invoice consisting of details such as payment reference id, date, time, bank details is generated. 2. The receipt is emailed to the student.
Alternate flow 1: User Exits The user decides to cancel the post and exists from the system. The use case ends.
Special Requirements: None
Associated use cases: Login

12. PRINT REGISTRATION FORM

Introduction: This use case generates a registration after successful payment of registration fees and course selection.

Actors: Administrator, Student

Pre-condition: The customer must be logged into the system.
A successful payment should be made.
Courses should have been selected.

Post-condition: A registration form is downloaded onto the user's system.

Basic flow:

1. A registration form is generated containing student details and courses selected.

Alternate Flow 1: Courses not have been selected

If the user has not selected the courses, an error message is shown to the user and the use case ends.

Alternate Flow 2: Registration Fees not paid

If the user has not yet paid the registration fees, an error message is shown to the user and the use case ends.

Alternate flow 3: User Exits

The user decides to cancel the post and exists from the system. The use case ends.

Special Requirements: None

Associated use cases: Login , Pay Registration Fees