## Lab File

### Software Engineering

DEPARTMENT OF

### COMPUTER SCIENCE AND ENGINEERING



Submitted To: Submitted By:

Dr. Sumit Kumar Abhimanyu Bhatia

Associate Professor A2305219079

CSE Department, ASET 6CSE – 2X

## CASE STUDY: RESULT MANAGEMENT SYSTEM

**AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY AMITY UNIVERSITY UTTAR PRADESH**

## NOIDA-201301

**INDEX**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Exper iment No.** | **Category**  **of Assignment** | **Code** | **Name of Experiment** | **Date of Allotment of experiment** | **Date of Evaluation** | **Max**  **Mar ks** | **Marks obtai ned** | **Sign.**  **of Faculty** |
| **1.** | **Mandatory Experiment** | **LR (0)** | **FORMULATING THE PROBLEM STATEMENT FOR LIBRARY MANAGEMENT SYSTEM.** | **6/01/2022** | **13/01/2022** | **1** |  |  |
| **2.** | **Mandatory Experiment** |  | **USE CASE DIAGRAM DESIGN FOR LIBRARY MANAGEMENT SYSTEM** | **13/01/2022** | **20/01/2022** | **1** |  |  |
| **3.** | **Mandatory Experiment** |  | **CREATING LEVEL-0 AND LEVEL- 1 DATA FLOW DIAGRAM FOR LIBRARY MANAGEMENT SYSTEM** | **20/01/2022** | **27/01/2022** | **1** |  |  |
| **4.** | **Mandatory Experiment** |  | **Creating E-R Diagram for Library Management System** | **27/01/2022** | **3/02/2022** | **1** |  |  |
| **5.** | **Mandatory Experiment** |  | **Creating Use Case Templates for Library Management System** | **3/02/2022** | **17/02/2022** | **1** |  |  |
| **6.** | **Mandatory Experiment** |  | **Creating Sequence Diagrams for Library Management System** | **17/02/2022** | **03/03/2022** | **1** |  |  |
| **7.** | **Mandatory Experiment** |  | **Creating A Class Diagram for Library Management System** | **03/03/2022** | **10/03/2022** | **1** |  |  |
| **8.** | **Mandatory Experiment** |  | **Creating An Activity Diagram for Library Management System** | **10/03/2022** | **17/03/2022** | **1** |  |  |
| **9.** | **Mandatory Experiment** |  | **Creating A State Chart Diagram for Library Management System** | **17/03/2022** | **24/03/2022** | **1** |  |  |
| **10.** | **Mandatory Experiment** |  | **Creating A Component Diagram for Library Management System** | **24/03/2022** | **31/03/2022** | **1** |  |  |
| **11.** | **Mandatory Experiment** |  | **Creating A Collaborative Diagram for Library Management System** | **31.03/2022** | **7/04/2022** | **1** |  |  |
| **12.** | **Mandatory Experiment** |  | **Creating A Deployment Diagram for Library Management System** | **7/04/2022** | **21/04/2022** | **1** |  |  |

# Problem Statement

A school is organized in teaching different classes and in each class the students are offered different theory and lab subjects. The evaluation of each subject is done out of 100 marks. 100 marks are broken into 30 marks for internal evaluation and 70 marks for external evaluation. If the total marks of the student are >40 in a subject, he is considered 'Pass' in that subject. Otherwise, the student is considered "Fail" in that subject. The admin is responsible for adding the final marks of the students and the following are prepared and maintained manually:

* Information about various students.
* list of students which are registered.
* List of marks of each student.
* Personal details of students.

The school decides to automate the manual process for result generation in order to improve the existing system. The proposed system should perform the following functions:

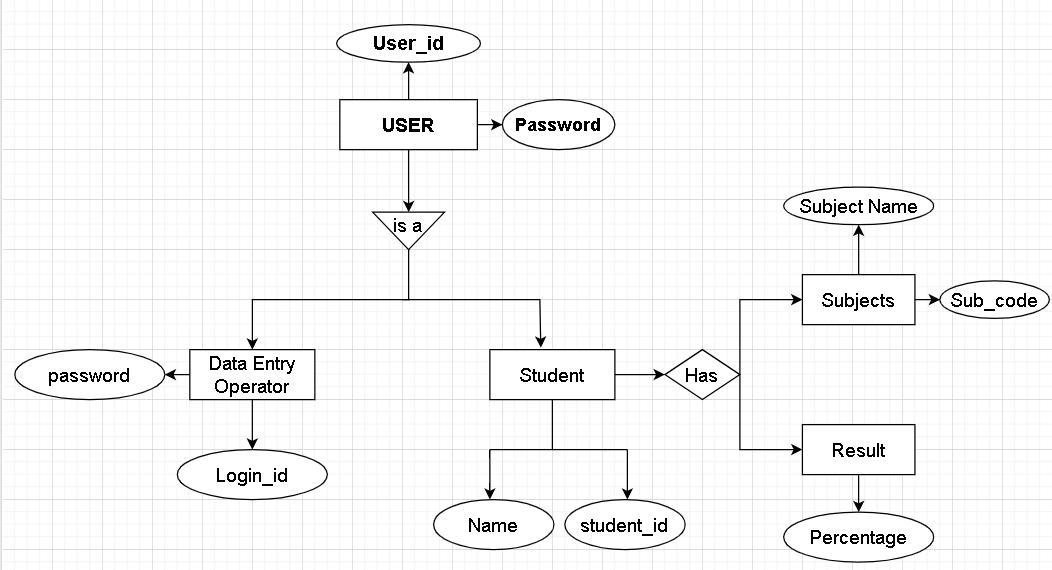
* Maintain personal details of student.
* Maintain the list of students.
* Issue of marks to students.
* Update the marks of students whenever needed.
* Manage the list of students.
* Add new students to existing lists.
* Maintain and update existing classes.
* Result generation.
* Result should be available to download in pdf format.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

## Diagram Description automatically generatedUSE-CASE DIAGRAM

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

**E-R DIAGRAM**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

**USE CASE TEMPLATES**

1. **LOGIN**

#### Introduction

This use case describes how a user logs into the result management system.

#### Actors:

1. data entry operator
2. administrator/deputy registrar

#### Pre-conditions: none

* 1. **Post conditions:**

if the use case is successful, the actor is Logged into the system. If not, the system state is unchanged.

#### Basic flow:

this use case starts when the actor wishes to login to the result management system.

1. System requests that the actor enter his/her name and password.
2. The actor enters his/her name & password.
3. System validates name & password, and if finds correct allow the actor to logs into the system.

#### Alternate Flows

* + 1. **Invalid name & password**

If in the basic flow, the actor enters an invalid name and/or password, the system displays an error message. The actor can choose to either return to the beginning of the basic flow or cancel the login, at that point, the use case ends.

#### Special Requirements: None

* 1. **Use case Relationships:** None

1. **MAINTAIN STUDENT DETAILS**
   1. **Introduction:** Allow DEO to maintain student details. This includes adding, changing and deleting student information
   2. **Actors :** DEO
   3. **Pre-Conditions:** DEO must be logged onto the system before this use case begins.
   4. **Post-conditions:** If use case is successful, student information is added/updated/deleted from the system. Otherwise, the system state is unchanged.
   5. **Basic Flow:** Starts when DEO wishes to add/modify/update/delete Student information.
2. The system requests the DEO to specify the function, he/she would like to perform (Add/update/delete)
3. One of the sub flow will execute after getting the information.

If DEO selects "Add a student", "Add a student" sub flow will be executed. If DEO selects "update a student", "update a student" sub flow will be executed.

If DEO selects "delete a student", "delete a student" sub flow will be executed.

#### Add a student

1. The system requests the DEO to enter: Name Address Roll No Phone No Date of admission
2. System generates unique id

#### Update a student

1. System requires the DEO to enter student-id.
2. DEO enters the student\_id. The system retrieves and display the student information.
3. DEO makes the desired changes to the student information.
4. After changes, the system updates the student record with changed information.

#### Delete a student

1. The system requests the DEO to specify the student-id.
2. DEO enters the student-id. The system retrieves and displays the student information.
3. The system prompts the DEO to confirm the deletion of the student.
4. The DEO confirms the deletion. (v) The system marks the student record for deletion.

#### Alternative flows

* + 1. **Student not found**

If in the update a student or delete a student sub flows, a student with specified\_id does not exist, the system displays an error message .The DEO may enter a different id or cancel the operation. At this point ,Use case ends.

#### Update Cancelled

If in the update a student sub-flow, the data entry operator decides not to update the student information, the update is cancelled and the basic flow is restarted at the begin.

Use Cases

#### Delete cancelled

If in the delete a student sub flows, DEO decides not to delete student record

,the delete is cancelled and the basic flow is re-started at the beginning.

#### Special requirements

None

#### Use case relationships

None

## Maintain Subject Details

* 1. **Introduction:** The DEO to maintain subject information. This includes adding, changing, deleting subject information from the system.
  2. **Actors :** DEO
  3. **Preconditions :** DEO must be logged onto the system before the use case begins.
  4. **Post conditions:** If the use case is successful, the subject information is added, updated, or deleted from the system, otherwise the system state is unchanged.
  5. **Basic flows:** The use case starts when DEO wishes to add, change, and/or delete subject information from the system.

1. The system requests DEO to specify the function he/she would like to perform i.e.

* Add a subject
* Update a subject
* Delete a subject.

1. Once the DEO provides the required information, one of the sub flows is executed.

* If DEO selected “Add a subject” the “Add-a subject sub flow is executed.
* If DEO selected “Update-a subject” the “update-a- subject” sub flow is executed.
* If DEO selected “Delete- a- subject”, the “Delete-a-subject” sub flow is executed.

#### Add a Subject

1. The System requests the DEO to enter the subject information. This includes:

* Name of the subject
* Subject Code
* Semester
* Credit points

1. Once DEO provides the requested information,

the system generates and assigns a unique subject-id to the subject. The subject is added to the system.

1. The system provides the DEO with new subject-id.

#### Update a Subject

1. The system requests the DEO to enter subject\_id.
2. DEO enters the subject\_id. The system retrieves and displays the subject information.
3. DEO makes the changes.
4. Record is updated.

#### Delete a Subject

1. Entry of subject\_id.
2. After this, system retrieves & displays subject information.

* System prompts the DEO to confirm the deletion.
* DEO verifies the deletion.
* The system marks the subject record for deletion.3.6 Alternative Flow

#### 3.6.1 Subject not found

If in any sub flows, subject-id not found, error message is displayed. The DEO may enter a different id or cancel the case ends here.

## Maintain Result Details

* 1. **Introduction:** This use case allows the DEO to maintain subject & marks information of each student. This includes adding

and/or deleting subject and marks information from the system.

* 1. **Actor:** DEO
  2. **Preconditions:** must be logged onto the system.
  3. **Post Conditions:** If use case is successful ,marks information is added or deleted from the system. Otherwise,

the system state is unchanged.

* 1. **Basic Flow:** This use case starts, when the DEO wishes to add, update and/or delete marks from the system.

1. DEO to specify the function
2. Once DEO provides the information one of the sub flow is executed.

* If DEO selected “Add Marks “, the Add marks sub flow is executed.
* If DEO selected “Update Marks”, the update marks sub flow is executed.
* If DEO selected “Delete Marks”, the delete marks sub flow is executed.
  + 1. **Add Marks Records:** Add marks information. This includes:

1. Selecting a subject code.
2. Selecting the student enrollment number.
3. Entering internal/external marks for that subject code & enrollment number.
4. If DEO tries to enter marks for the same combination of subject and enrollment number,the system gives a message that the marks have already been entered.
5. Each record is assigned a unique result\_id.

#### Delete Marks records

1. DEO makes the following entries:
   1. Selecting subject for which marks have to be deleted.
   2. Selecting student enrollment number.
   3. Displays the record with id number.
   4. Verify the deletion.
   5. Delete the record.

#### Update Marks records

1. The System requests DEO to enter the record\_id.
2. DEO enters record\_id. The system retrieves & displays the information.
3. DEO makes changes.
4. Record is updated.

#### Compute Result

1. Once the marks are entered, result is computed for each student.
2. If a student has scored more than 50% in asubject, the associated credit points are allotted to that student.
3. The result is displayed with subject-code, marks & credit points.

#### Alternative Flow

* + 1. **Record not found:**If in update or delete marks sub flows, marks with specified id number do not exist, the system displays an error message. DEO can enter another id or cancel the operation.

#### Delete Cancelled

If in Delete Marks, DEO decides not to delete marks, the delete is cancelled and basic flow is re-started at the beginning.

#### Special Requirements: None

* 1. **Use case relationships:** None

## View/Display result

* 1. **Introduction:** This use case allows the student/Teacher or anyone to view the result. The result can be viewed on the basis of course code and/or enrollment number.
  2. **Actors:** Administrator/DR, Teacher/Student

#### Pre-Conditions: None

* 1. **Post Conditions:** If use case is successful, the marks information is displayed by the system. Otherwise, state is unchanged.
  2. **Basic Flow:** Use case begins when student, teacher or any other person wish to view the result.

Two ways

-- Enrollment no.

-- Course code

(ii) After selection, one of the sub flow is executed.

#### View result enrollment number wise:

1. User to enter enrollment number
2. System retrieves the marks of all subjects with credit points
3. Result is displayed.

#### Alternative Flow

* + 1. **Record not found:** Error message should be displayed.

#### Special Requirements: None

* 1. **Use Case relationships:** None

## Generate Report

* 1. **Introduction:** This use case allows the Administrator, Student or Teacher to generate result reports.

#### Actors: DR

* 1. **Pre-Conditions:** Administrator, Student or Teacher must logged on to the system
  2. **Post conditions:** If use case is successful, desired report is generated. Otherwise, the system state is unchanged.
  3. **Basic Flow:** The use case starts, when Administrator, Student or Teacher wish to generate reports.

#### Alternative Flows

* + 1. **Record not found:** If not found, system generates appropriate message. The Administrator, Student or Teacher can select another option or cancel the operation. At this point, the use case ends.

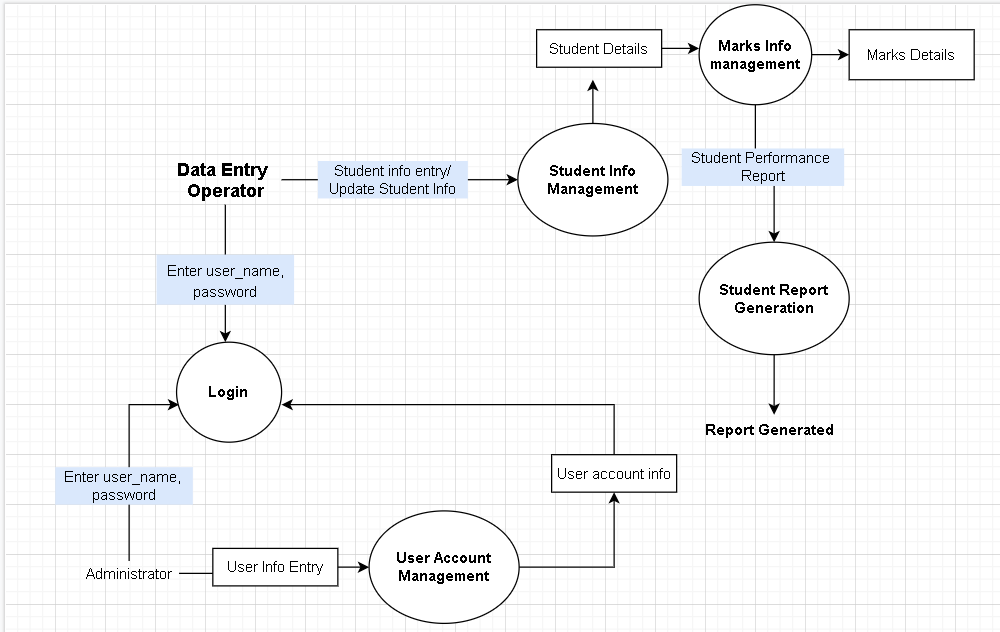
#### Special Requirements: None

* 1. **Use case relationships:** None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

## Diagram Description automatically generatedLEVEL DFD and 1-LEVEL DFD

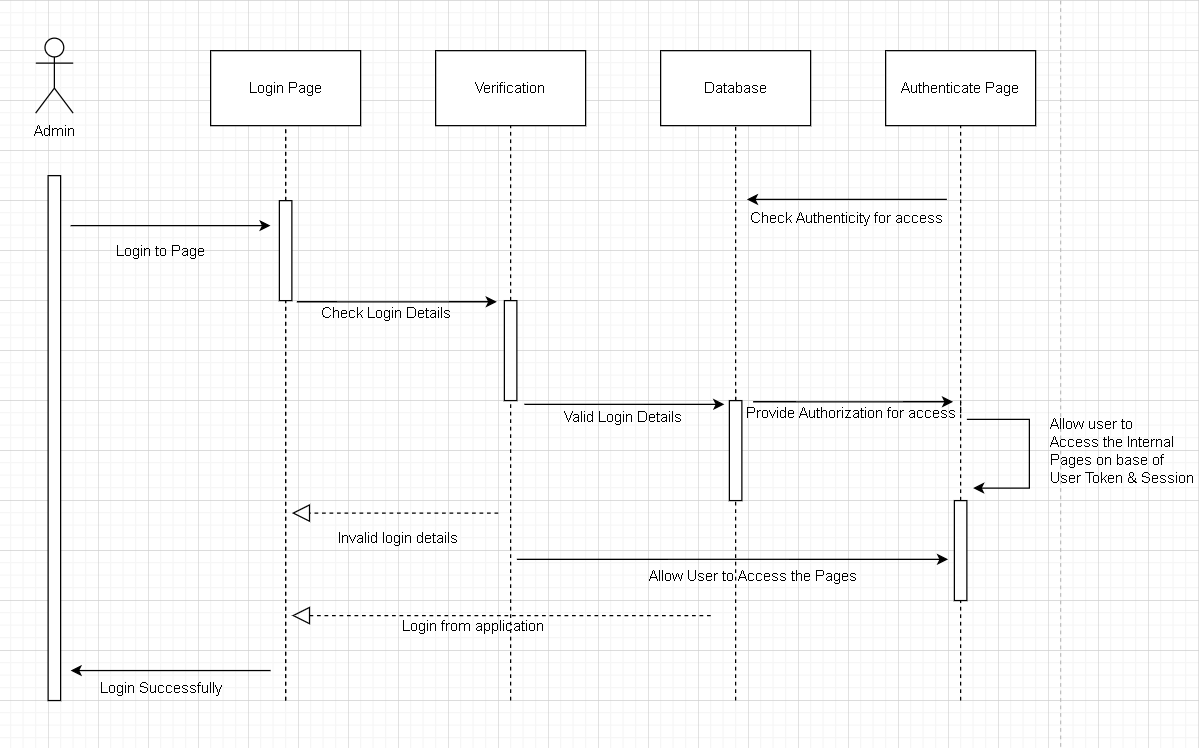
### level DFD



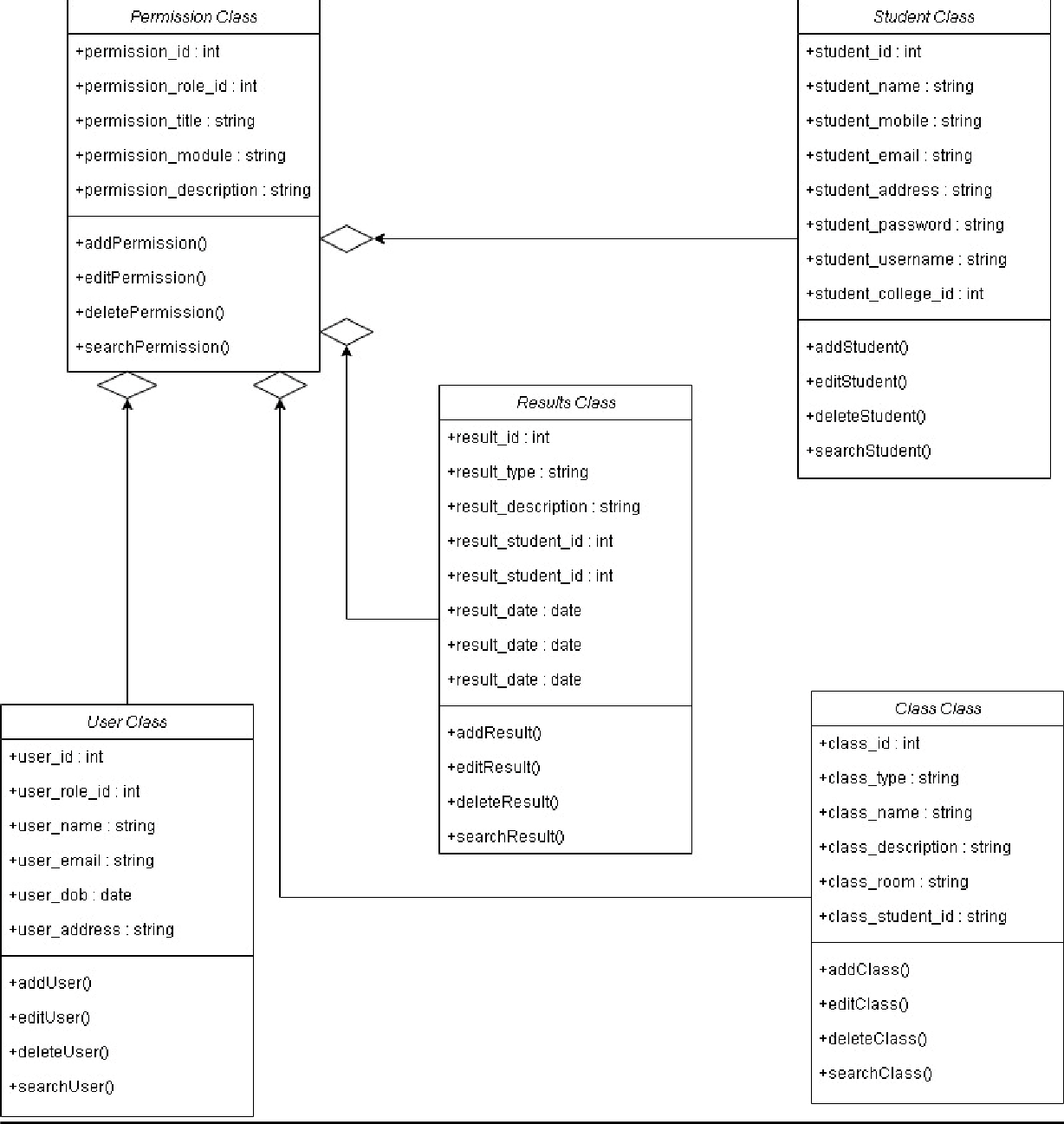
* 1. level DFD

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

# Sequence Diagram



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

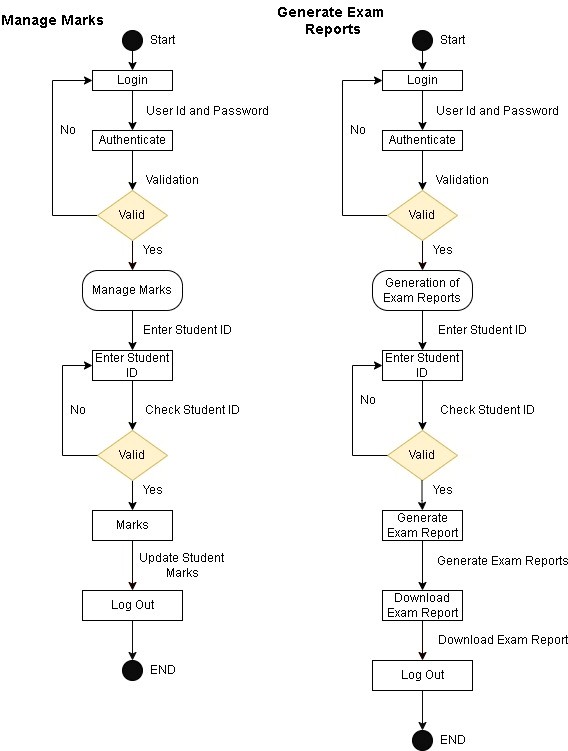
**Class Diagram**

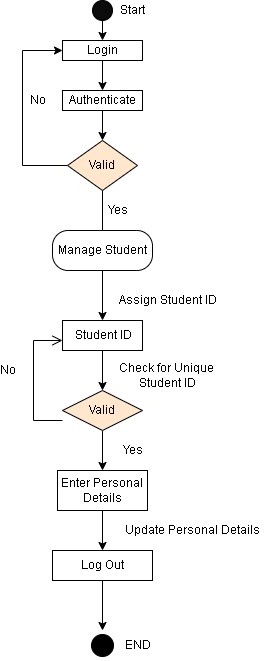
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

## Activity Diagram

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

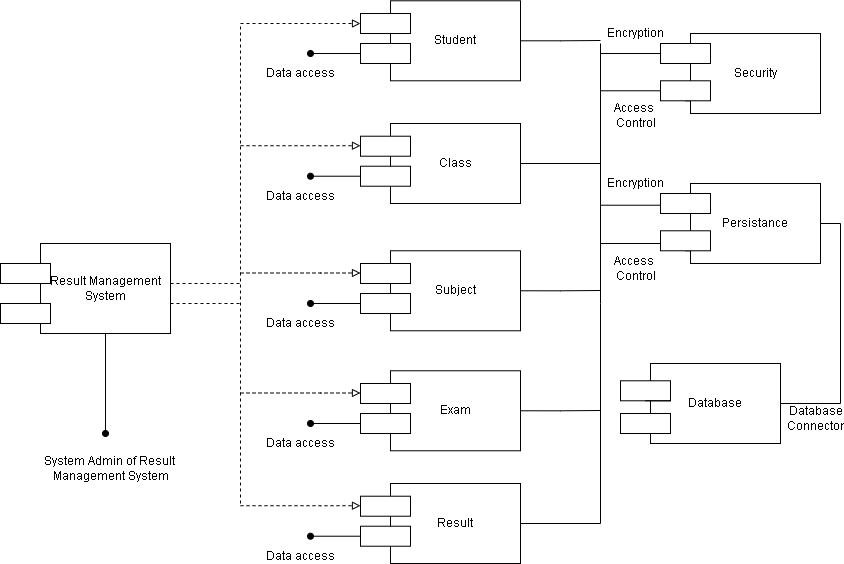
**State Chart**

**Student Details**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |

**Component Diagram**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Internal Assessment (Mandatory Experiment) Sheet for Lab Experiment Department of Computer Science & Engineering  Amity University, Noida (UP) | | | | |
| Programme | B. Tech CSE | | Course Name | Software Engineering |
| Course Code | [IT301] | | Semester | 6 |
| Student Name | Abhimanyu Bhatia | | Enrollment No. | A2305219079 |
| Marking Criteria | | | | |
| Criteria | Total Marks | Marks Obtained | | Comments |
| Concept (A) | 2 |  | |  |
| Implementation (B) | 2 |  | |  |
| Performance (C) | 2 |  | |  |
| Total | 6 |  | |  |