EXPERIMENT 5

Aim

To draft the Software Requirements Specification Document for the Course Management System

Description

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers (in market-driven projects, these roles may be played by the marketing and development divisions) on what the software product is to do as well as what it is not expected to do. Software requirements specification permits a rigorous assessment of requirements before design can begin and reduces later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules.

Draft a Software Requirement Specification Document for the Course Registration System

1. Introduction 1.1 Purpose 1.2 Scope 1.3 Definitions, Acronyms, and Abbreviations 1.4 References 1.5 Overview
2. Overall Description 2.1 Product Perspective 2.1.1 System Interfaces 2.1.2 User Interfaces 2.1.3 Hardware Interfaces 2.1.4 Software Interfaces 2.1.5 Communication Interfaces 2.1.6 Memory Constraints 2.1.7 Operations 2.1.8 Site Adaptation Requirements 2.2 Product Functions 2.3 User Characteristics 2.4 Constraints

2.5 Assumptions and Dependencies
2.6 Apportioning of Requirements

3. Specific Requirements
3.1 External Interface
3.2 Functions
3.3 Performance Requirements
3.4 Logical Database Requirements
3.5 Design Constraints
3.5.1 Standards Compliance

3.6 Software System Attributes
3.6.1 Reliability
3.6.2 Availability
3.6.2 Security
3.6.3 Maintainability
3.6.4 Portability
3.7 Organizing the specific requirements
3.7.1 System Mode
3.7.2 User Class
3.7.3 Objects
3.7.4 Feature
3.7.5 Stimulus
3.7.6 Response
3.7.7 Functional Hierarchy
3.8 Additional Components

4 Supporting Information

1.Introduction

This document aims at defining the overall software requirements for 'Course Management System'. Efforts have been made to define the requirements exhaustively and accurately. The final product will be having the only features/functionalities mentioned in this document and assumptions for any additional functionality/feature should not be made by any of the parties involved in developing/testing/implementing/using this product. In case it is required to have some additional features, a formal change request will need to be raised and consequently a new release of this document and/or product will be released.

1.1 Purpose

The Course Management System(CMS) maintains the information about courses, a student has opted for online. The students can view their attendance, download course material, check their grades and network with other classmates. It also maintains the class lectures, grades and online assignments uploaded by the faculty.

1.2 Scope

The name of the software is Course Management System(CMS), the system will be referred to as CMS in the rest of the SRS. The proposed CMS will be able to perform the following functions:

Dos

1. Maintain details of courses offered to the students in each semester.
2. Provide registration facility to students.
3. Forum Facility for Discussion**.**
4. Maintain Students details.
5. Faculty can Upload Course Material and Assignments.
6. Faculty can mark and view attendance of students.
7. Students can view their attendance.
8. Faculty can upload marks details of students.
9. Generate the following reports :
    a. Student's Mark Sheet
    b. Student's Attendance Report
    c. Student's Registration Form

Don'ts

1. Does not generate loginID and password for faculty and Students.
2. Does not allow to upload photos or files in forum.

Benefits

The CMS provides the following facilities :

1. Easy availability of assignments, lectures and course materials.
2. Easy marking of attendance.
3. Efficient calculation for percentage and therefore Marksheet Generation.
4. Printing of Reports.

1.3 Definitions, Acronyms and Abbreviations

SRS : Software Requirement Specification.

CMS: Course Management System.

RAM: Random Access Memory.

Student: Any candidate admitted in a programme offered by the University.

System Administrator/Administrator: User having all the privileges to maintain book, student and faculty/employee details.

Faculty: Teaching Staff of the University—Professor, Associate Professor and Assistant Professor.

1.4 References

- Software Engineering by K.K. Aggarwal & Yogesh Singh, New Age Publishing House, 3rd Edition, 2008.
- IEEE Recommended Practice for Software Requirements Specification—IEEE Std. 890-1998.
- IEEE Standard for Software Test Documentation – IEEE Std. 829-1998.

1.5 Overview

The rest of the SRS document describes various system requirements, interfaces, features and functionalities in detail.

2. Overall Description

The CMS provides access to the student and faculty members based on unique identification id and password possessed by the user.Thesystem allows the student to opt for courses at the beginning of the semester and to view his/her attendance and also to print the registration car by generating it. The system provides the faculty member to post the attendance each day of the student and to upload course material and assignments. The system should provide the student to a list of courses to opt for.

2.1 Product Perspective

The CMS shall be developed using client/server architecture and be compatible with Microsoft windows operating system.The front end of the system will be developed using visual basic 6.0 and the backend will be developed using MS SQL server 2005.

2.1.1 System Interfaces

None

2.1.2 User Interfaces

The application will have a user friendly and menu based interface. Following screens will be provided:

- A login screen asking registered user for entering username and password. Also the language selection option will be provided.
- .The students will be able to update his/her details like address or contact number through an interface.
- An interface will be provided to student to register for different courses at the beginning of the semester.
- Another interface for student to view his attendance in the past.
- The faculty will be able to mark attendance of the student day wise through maintain attendance interface.
- The faculty will be able to mark marks of the student course wise through maintain marks interface.
- All the members will be able to post in forum through post interface.

2.1.3 Hardware Interfaces

(i) Screen resolution of at least 500*500 is advised for proper and complete viewing of all the interfaces.

(ii) Support for printer(dot matrix, deskjet, laserjet).

(iii) Standalone system or network based- not a concern, as it will be possible to run the application on any of these.

2.1.4 Software Interfaces

- MS-windows Operating system.
- MS SQL server 2005 for backend.

2.1.5 Communication Interfaces

Communication is via local area network(LAN).

2.1.6 Memory Constraints

At least 512 MB RAM and 1 GB space on hard disk will be required for running the application.

2.1.7 Operation

None.

2.1.8 Site Adaptation Requirements

The terminals at the client side will have to support the hardware and software interfaces specified in above sections.

2.2 Product Functions

The CMS will allow access only to registered users, database administrator and system engineer. Depending on the authorization grant provided by the DBA only specific modules of the system can be accessed by a user.

A summary of major functions that the software will perform:

- The software should provide the student and faculty members to login to the system.
- The student should be provided by a list of courses in the starting of each semester from which they can choose and fill their Registration Form online.
- The system should also provide the ability to student to print the registration form of the personal details and the courses opted by the student.
- The students should be able to change their phone no, Address and email id.
- The students should be provided by the facility to view his/her attendance and download the attendance as well.
- The faculty should be provided with the ability to add courses, delete courses, view attendance, mark attendance of every student, enter marks of students, print report card, upload course material, lectures and assignments.
- The students should be able to download and print their report cards.
- There should be a forum facility for discussion.

2.3 User Characteristics

- Qualification: At least matriculation and comfortable with English
- Experience: Should be well versed/informed about the registration process of the university.
- Technical Experience: Elementary knowledge of computers.

2.4 Constraints

- The system may become slow with increase in number of records being stored.
- There will be only one administrator.
- User will not be allowed to update the primary key.
- The delete operation is available to the administrator.

2.5 Assumptions and Dependencies

- Each user would have only one username and password. In other words a user is not allowed to have multiple id.
- The academic section will provide the lists of the admitted students.
- The academic section will provide the list of the faculty members in the university.

2.6 Apportioning of Requirements

Not required.

3. Specific requirements

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

3.1 External Interface Requirements

3.1.1 User Interfaces

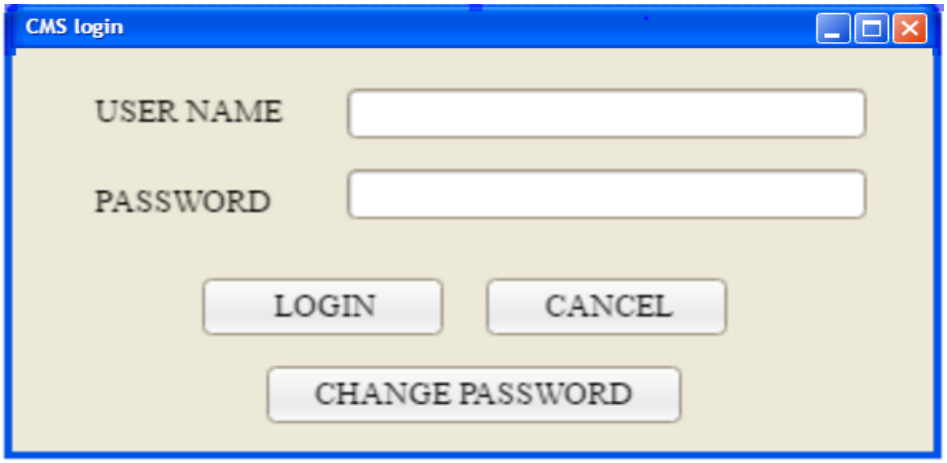
The following user interfaces will be provided by the system.

(i) Login Form

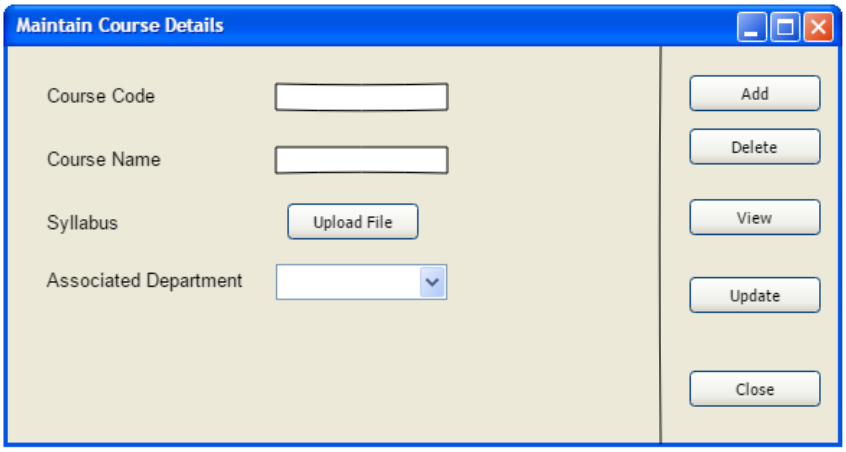
This will be the first form which will be displayed. It will allow the user to access the different forms based on his/her role.

Various fields available on this form will be:

- Login ID- Alphanumeric of length in the range of 4 to 15 characters. Special characters and blank spaces are not allowed.
- Password- Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters ae allowed.

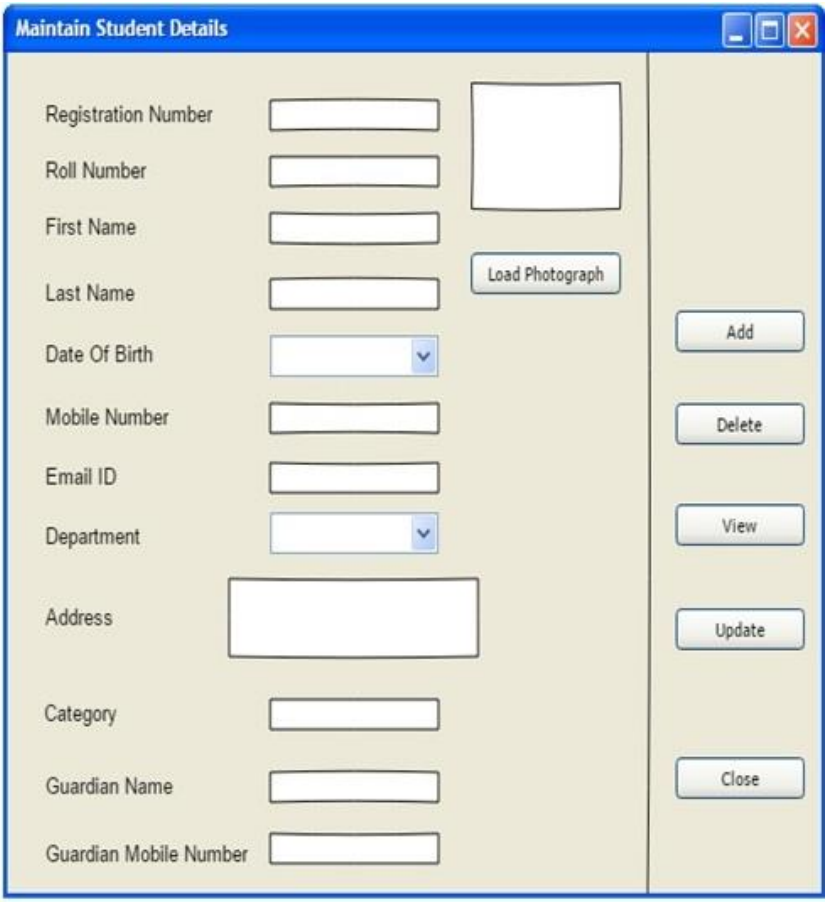


(ii) Maintain Course Details



1. **Course Id:** Numeric and will have value from 100 to 500.The system will not allow the user to enter non-numeric characters,special characters and out-of-range numbers.
2. **Course Name::** Alphanumeric of length 1 to 100. special characters are not allowed and spaces are allowed.Numeric data will not be allowed.The system should not allow the user to enter special characters and out of range characters.
3. **Syllabus:** A button through which user will be able to upload the syllabus in pdf format only.
4. **Associated department:** A drop down menu for selecting the appropriate department for the particular course.

(iii) Maintain Student Details



This form will be accessible to Administrator/DEO. It will allow him/her to add/delete/modify student details.

Various fields available on this form will be:

1. **Registration number:** Numeric and will have value from 100 to 19999.The system will not allow the user to enter non-numeric characters and out-of-range numbers.
2. **Roll number:** Alphanumeric of length 11. The system will not allow the user to enter special characters except '/', blank spaces, out of range characters.
3. **First name:** Alphanumeric of length 1 to 100. Special characters and spaces are not allowed. Numeric data will not be allowed. The system should not allow the user to enter special characters, spaces and out of range characters.
4. **Last name:** Alphanumeric of length 1 to 100. Special characters and spaces are not allowed. Numeric data will not be allowed. The system should not allow the user to enter special characters, spaces and out of range characters.
5. **Date of Birth:** Will be of format mm/dd/yyyy. It will have 10 alphanumeric characters.
6. **Mobile number:** Numeric and of length should be exactly equals to 10 digit system should not allow user to enter Alphabets and special characters and out of range or under range numbers.
7. **Email-id:** Alphanumeric of length 1 to 1000. Special characters are allowed and spaces

are allowed. Numeric data will be allowed. The system should not allow the user to enter spaces and out of length characters.

8. **Branch:** A drop down menu for choosing branch from the available options.
9. **Address:** Alphanumeric of length 1 to 1000. Special characters and spaces are allowed. The system should not allow the user to enter out of range characters.
10. **Category:** Alphanumeric of length 1 to 200. Special characters and spaces are not allowed. Numeric data not allowed. The system should not allow the user to enter numeric data and out of range characters.
11. **Guardian name**: Alphanumeric of length 1 to 200. Special characters and numeric data are not allowed spaces are allowed. System should not allow the user to enter numeric data and out of range characters.
12. **Guardian mobile number:** Numeric and of length should be exactly equals to 10 digit. System should not allow user to enter Alphabets and special characters and out of range or under range numbers.

(iv) Maintain Faculty Details

(v) **Faculty Id:** Numeric and will have value from 100 to 19999.The system will not allow the user to enter non-numeric characters and out-of-range numbers.

(vi) **First name:** Alphanumeric of length 1 to 100. special characters and spaces are not allowed.Numeric data will not be allowed.The system should not allow the user to enter special characters, spaces and out of range characters.

(vii) **Last name:** Alphanumeric of length 1 to 100. special characters and spaces are not allowed.Numeric data will not be allowed.The system should not allow the user to enter special characters, spaces and out of range characters.

(viii) **Date of Birth:** Will be of format mm/dd/yyyy. It will have 10 alphanumeric characters.

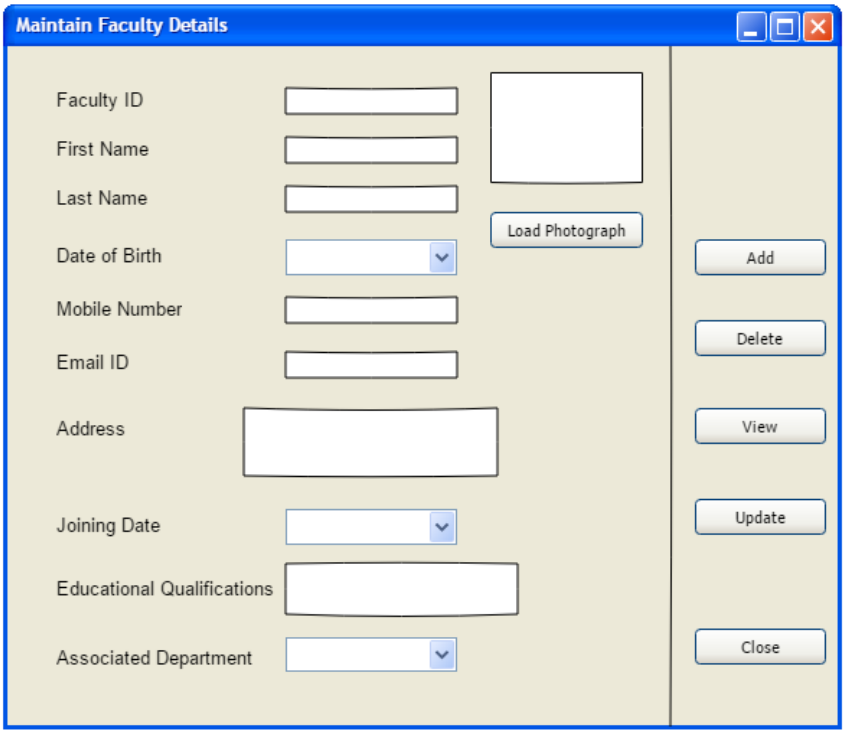
(ix) **Mobile number:** Numeric and of length should be exactly equals to 10 digit.system should not allow user to enter Alphabets and special characters and out of range or under range numbers.

(x) **Email-id:**Alphanumeric of length 1 to 1000. special characters are allowed and spaces are allowed.Numeric data will be allowed.The system should not allow the user to enter spaces and out of length characters.

(xi) **Address: :** Alphanumeric of length 1 to 1000. special characters and spaces are allowed.The system should not allow the user to enter out of range characters.

(xii) **Joining Date:** A drop down menu for choosing the date of joining should be present.

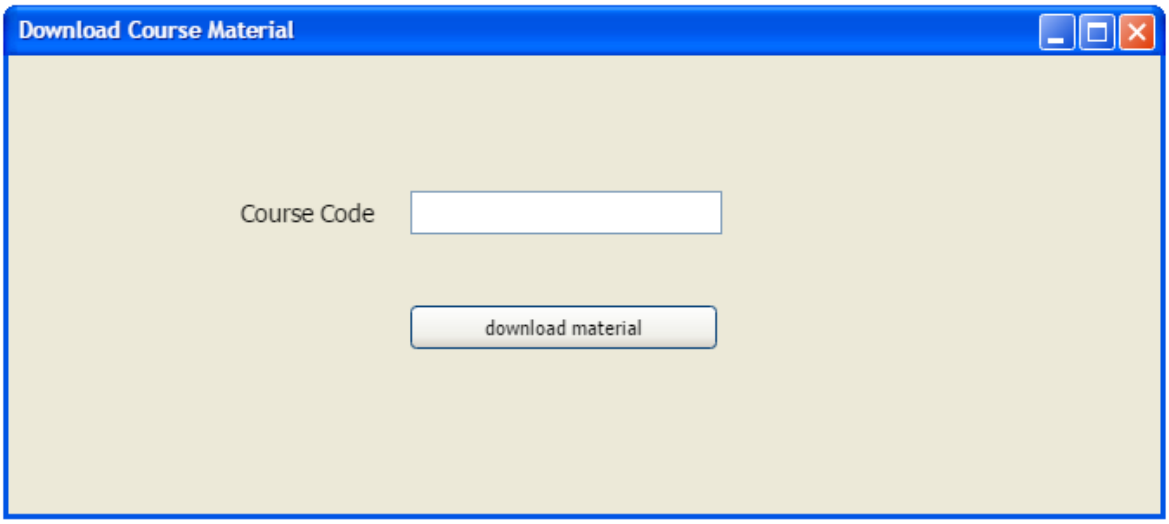
(xiii) **Associated Department:** Alphanumeric of length 1 to 1000. special characters and spaces are not allowed.Numeric data will not be allowed.The system should not allow the user to numeric data,special characters , spaces and out of length characters.



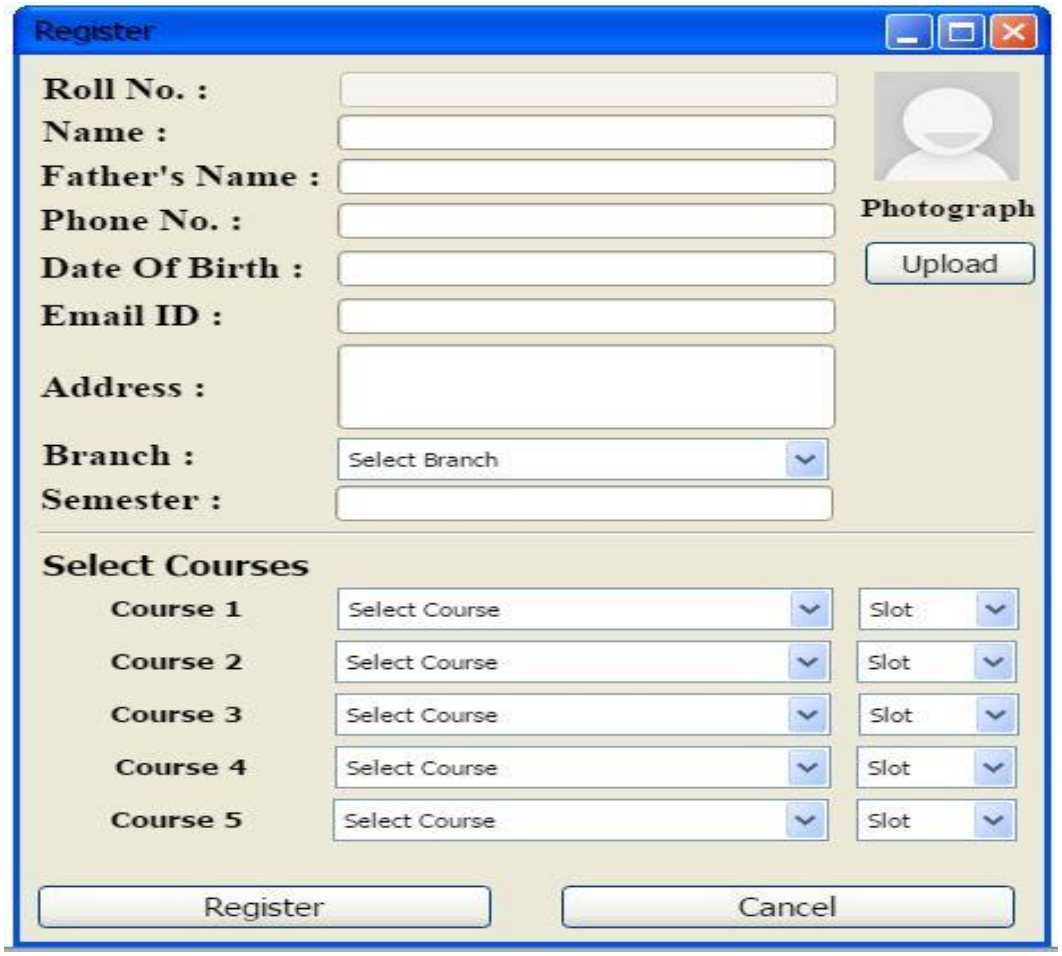
(v)Download Course material

This system allows the Student to download course material .

1. **Course Id:** Numeric and will have value from 100 to 500.The system will not allow the user to enter non-numeric characters,special characters and out-of-range numbers.



(vi) Register



This form will be accessible to students. This will allow him/her to register for a new semester.

Various fields available on this form will be:

1. **Roll number:** Alphanumeric of length 11.The system will not allow the user to enter special characters except '/', blank spaces.
2. **Name:** Alphanumeric of length 1 to 200. Special characters are not allowed and blank spaces are allowed. Numeric data will not be allowed. The system should not allow the user to enter special characters, spaces and out of range characters.
3. **Father's name**: Alphanumeric of length 1 to 200. Special characters and numeric data are not allowed, spaces are allowed. System should not allow the user to enter numeric data and out of range characters.
4. **Phone number:** Numeric and length should be exactly equals to 10 digit. System should not allow user to enter Alphabets and special characters and out of range or under range numbers.
5. **Date of Birth:** Will be of format mm/dd/yyyy. It will have 10 alphanumeric characters.

6. **Email-id: :** Alphanumeric of length 1 to 1000. Special characters are allowed and spaces are not allowed except '@' , '.' and '_'. Numeric data will be allowed. The system should not allow the user to enter spaces and out of length characters.
7. **Address: :** Alphanumeric of length 1 to 1000. special characters and spaces are allowed. The system should not allow the user to enter out of range characters.
8. **Branch:** A drop down menu for choosing branch from the available options.
9. **Semester :** Numeric and in range 1 to 8. Special character and blank spaces are not allowed. The system should not allow the user to enter out of range characters.
10. **Select Course :** A drop down menu for choosing courses from the available options.
11. **Slot :** A drop down menu for choosing slots from the available option.

(vii) Forum

This form will be accessible to Administrator/Students/Faculty. This will allow them to post in forum.

Various fields available on this form will be:

1. **Type a Message :** Alphanumeric of any length. Special characters, Numeric Data and blank spaces are allowed.

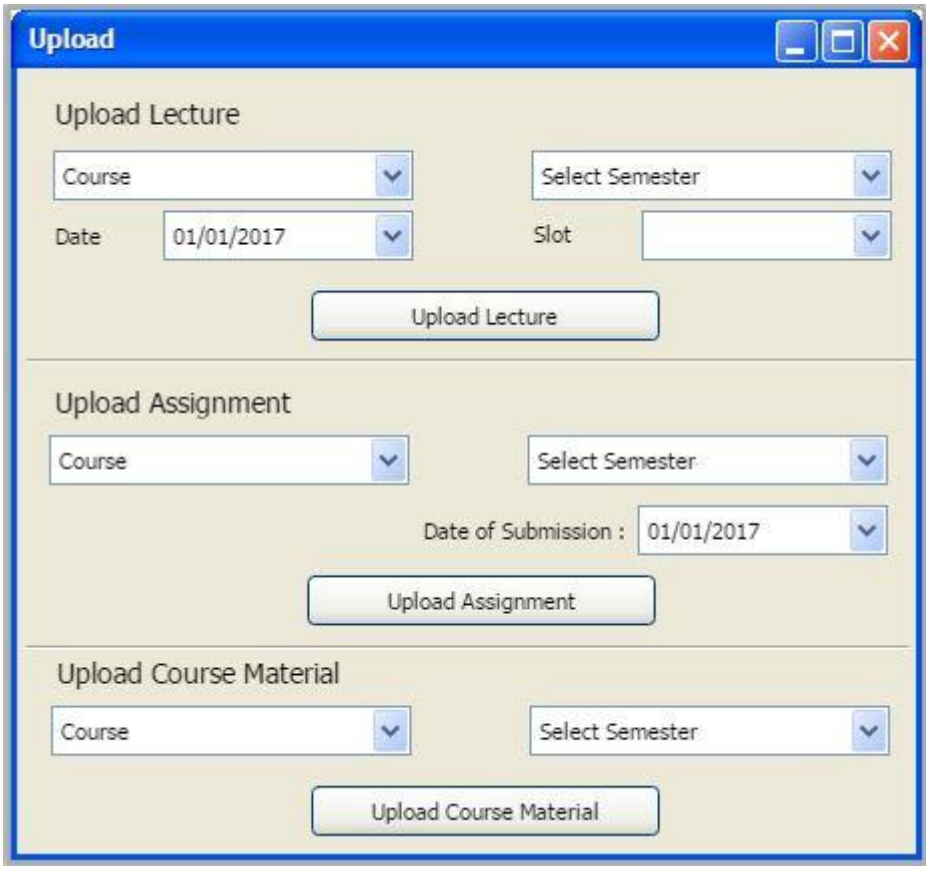
(viii) Maintain Marks

This form will be accessible to Faculty. This will allow the faculty to enter the marks of the student.

Various fields available on this form will be:

1. **Enter Roll Number :** Alphanumeric of length 11.The system will not allow the user to enter special characters except '/', blank spaces.
2. **Marks :** A dropdown menu for choosing the marks from 0 to 100 for a particular course.

(ix) Upload material

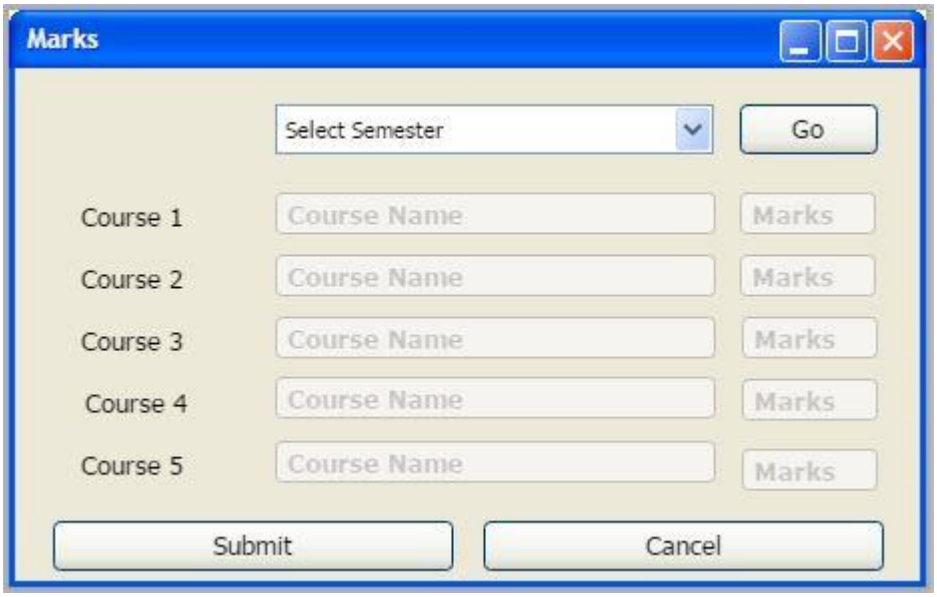


This form will be accessible to Administrator/Faculty. It will allow him/her to Upload Lecture/Assignment/Course Material.

Various fields available on this form will be:

1. **Course:** A drop down menu for choosing Course from the available options.
2. **Date:** A drop down menu for choosing date from the available options.
3. **Semester:** A drop down menu for choosing Semester from the available options.
4. **Slot:** A drop down menu for choosing Slot from the available options.

(x)View marks



This form will be accessible to Administrator/Student. It will allow him/her to View Marks.

Various fields available on this form will be:

1. **Semester:** A drop down menu for choosing Semester from the available options.

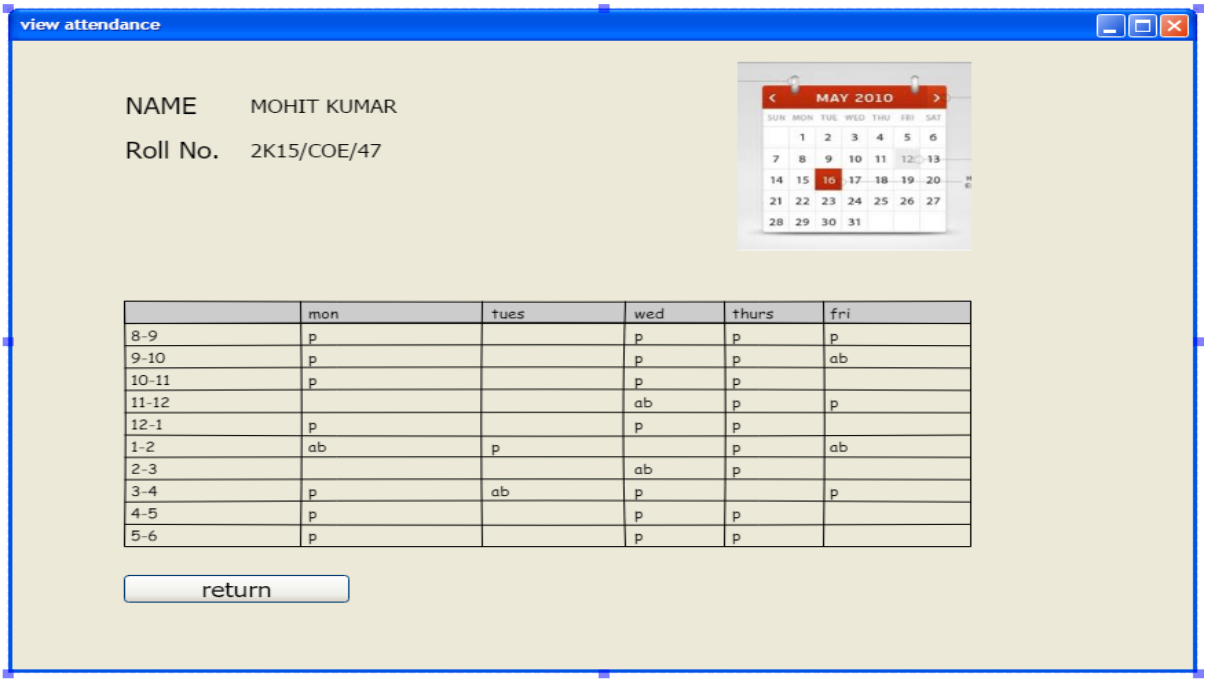
(xi)Print marks

This form will be accessible to Administrator/Student. It will allow him/her to print Marks.

Various fields available on this form will be:

None

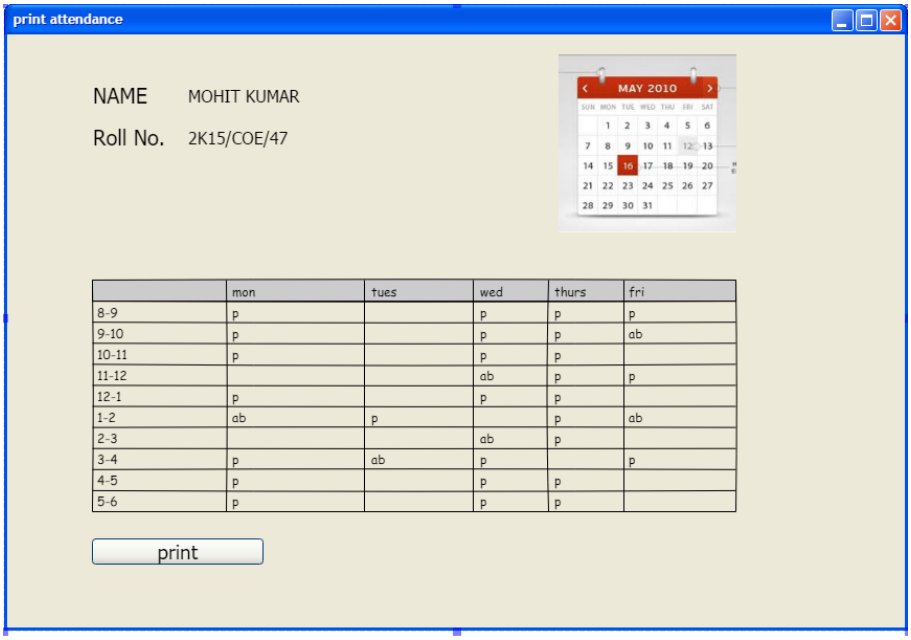
(xii) View Attendance



	mon	tues	wed	thurs	fri
8-9	p		p	p	p
9-10	p		p	p	ab
10-11	p		p	p	
11-12			ab	p	p
12-1	p		p	p	
1-2	ab	p		p	ab
2-3			ab	p	
3-4	p	ab	p		p
4-5	p		p	p	
5-6	p		p	p	

This form will be accessible to Administrator/Student. It will allow him/her to View Attendance.

(xiii) Print Attendance



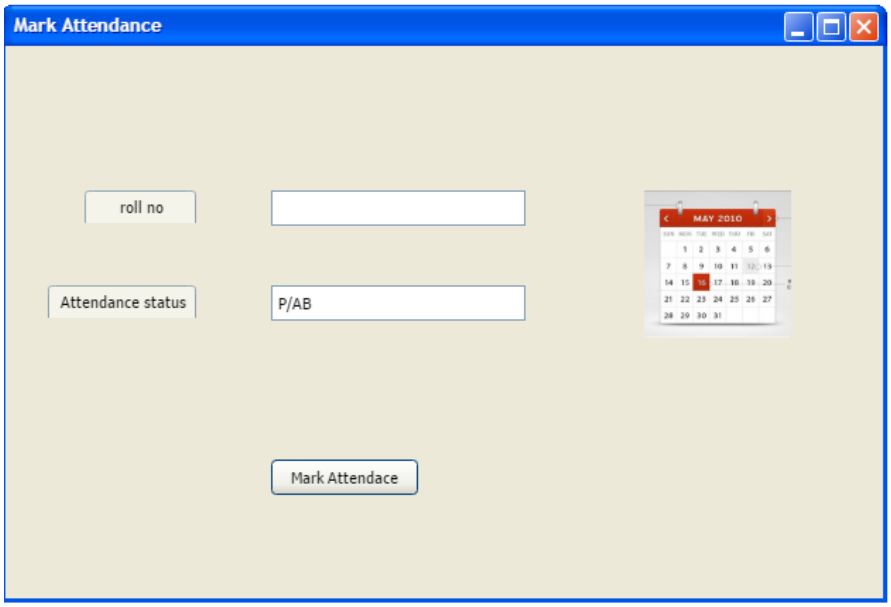
	mon	tues	wed	thurs	fri
8-9	p		p	p	p
9-10	p		p	p	ab
10-11	p		p	p	
11-12			ab	p	p
12-1	p		p	p	
1-2	ab	p		p	ab
2-3			ab	p	
3-4	p	ab	p		p
4-5	p		p	p	
5-6	p		p	p	

This form will be accessible to Administrator/Student. It will allow him/her to print Attendance.

Various fields available on this form will be:

None

(xiv) Mark Attendance



This form will be accessible to faculty/administartor. This will allow him/her to mark attendance for a particular date.

Various fields available on this form will be:

1. **Roll number:** Alphanumeric of length 11.The system will not allow the user to enter special characters except '/', blank spaces.
2. **Attendance status:** Alphanumeric which can either be P or AB.

3.1.2 Hardware Interfaces

As stated in section 2.1.3 .

3.1.3 Software Interfaces

As stated in section 2.1.4 .

3.1.4 Communication Interfaces

None

3.2 Functional Requirements

3.2.1 LOGIN

A. Use case diagram

INTRODUCTION: Login allows the user to be logged in to the system.
ACTORS: Student, Faculty member, Administrator.
PRE-CONDITION: none.
POST-CONDITION: If the use-case is successful, the actor is now logged into the system, if not the system state is unchanged.
FLOW OF EVENTS: **BASIC FLOW:** • The actor enters the password and the username • Authentication is done • The user gets logged in successfully. **ALTERNATE FLOW:** **1. Invalid username/password:** The system shows an error message and the system is redirected to the basic flow. 2. **Username/password incorrect:** The system shows an error message and the system is redirected to the basic flow.
SPECIFIC REQUIREMENT: none
ASSOSIATED USECASE: none

B. Validity Checks

1. Every user will have a unique login ID.
2. Login ID cannot be blank.
3. Login ID can only have 4 to 15 characters.
4. Login ID will not accept special character and blank spaces.
5. Password cannot be blank.
6. Length of password can only be 4 to 15 characters.
7. Alphabets, digits, hyphens and underscore characters are allowed in the password field.
8. Password will not accept blank spaces.

C. Sequencing Information

None

D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.2 Maintain Student Details :

Introduction: This use case documents the steps that the administrator must follow in order to maintain student details. This includes adding, updating, viewing and deleting student information.

Actors: Administrator, DEO

Pre-Condition: The Administrator/DEO 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/viewed from the system. Otherwise the system state remains unchanged.

Flow of Events:

Basic Flow:

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

1. The system requests that the Administrator/DEO 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/DEO provides the requested information, one of the sub flows is executed:
   a. If the Administrator/DEO selects “Add a Student”, the **Add a Student** subflow is executed.
   b. If the Administrator/DEO selects “Update a Student”, the **Update a Student** subflow is executed.
   c. If the Administrator/DEO selects “Delete a Student”, the **Delete a Student** subflow is executed.
   d. If the Administrator/DEO selects “View a Student”, the **View a Student** subflow is executed.

Basic Flow 1: Add a Student

The system request that the Administrator/DEO to 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/DEO provides the requested information, the student is added to the system.

Basic Flow 2: Update a Student

1. The system requests that the Administrator/DEO enter Registration Number.
2. The Administrator/DEO enters the Registration Number.
3. The system retrieves and displays the student information.
4. The Administrator/DEO makes the desired changes to the student information. This includes any of the information specified in the **Add a Student** subflow.
5. Once the Administrator/DEO updates the necessary information, the system updates the student information with the updated information.

Basic Flow 3: Delete a Student

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

Basic Flow 4: View a Student

1. The system requests that the Administrator/DEO to enter the Registration Number.
2. The system retrieves and displays the student information.

Alternative Flow:

Alternate Flow 1: Invalid Entry

If in the **Add a Student** or **Update a Student** flow, the Administrator/DEO 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/DEO 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/DEO 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/DEO 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/DEO 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/DEO 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

B. Validity Checks

1. Only the Administrator/DEO will be authorized to access the maintain student details module.
2. Every registered student will have a unique registration number and roll number.
3. Registration Number and roll number cannot be blank.
4. Name(First Name, Last Name) cannot be blank.
5. Date of Birth cannot be blank.
6. Mobile Number cannot be blank.
7. Email ID cannot be blank.
8. The field to choose Department cannot be blank.
9. Category cannot be blank.
10. Address cannot be blank.
11. Length of roll number can be at most 11.
12. Roll number cannot special characters except '/'.
13. Guardian name cannot include special characters and digits, blank spaces are allowed.
14. Name(Last Name, First Name) cannot include special characters and digits, blank spaces are allowed.
15. Address can have length up to 10 to 200 characters.
16. Phone number can have exactly 10 digits.
17. Email ID can have up to 50 characters.
18. Email ID should contain '@' and '.' characters.
19. Email ID cannot include blank spaces.
20. Date of birth would be in the format mm/dd/yyyy.

C. Sequencing Information: None

D. Error Handling/ Response to abnormal situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.3 Forum :

Introduction: This use case documents the steps that the administrator/student must follow in order to post a discussion in forum.
Actors: Administrator**,** Student.
Pre-condition The student must be logged in to the system.
Post-Condition: If the use-case is successful, the student can post in forum.
Flow of Events: **Basic Flow:** This use case starts when the Student wishes to post something in forum. 1. The student enters his/her thought in the forum and post it. **Alternate Flow:** None.
Specific Requirement: none
Associated Usecase: Login

B. Validity Checks

1. Only the Student/Administrator will be authorized to access the forum module.
2. Type Message field can contain alphabets, numbers, special characters.
3. Type message field can have any number of characters.

C. Sequencing Information:

None

D. Error Handling/Response to Abnormal Situations :

If any of the validation/sequencing flow does not hold true, an appropriate error message will be prompted to the administrator/student for doing the needful.

3.2.4 Semester Registration:

Introduction: This use case documents the steps that the administrator/student must follow in order to register for a new semester.

ACTORS: Administrator, Student.

PRE-CONDITION: The administrator/student must be logged in to the system before the use case begins.

POST-CONDITION: If the use-case is successful, the student gets registered for a new semester.

Flow of Events:

Basic Flow:

This use case starts when the Administrator/Student wishes to register a student for a new semester.

1. The system request that the Administrator/Student to enter the student information. This includes :

- Student name (First Name, Last Name)
- Photograph
- Date of Birth
- Mobile Number
- Email ID
- Branch
- Address
- Guardian Name
- Guardian Mobile Number
- Semester
- Course codes and respective slots.

Once the Administrator/Student provides the requested information, the student is registered for a new semester.

Alternative Flow:

Alternate Flow 1: Invalid Entry

If in the Administrator/Student enters invalid Student Name/Date of Birth/Mobile Number/Email ID/Address/Branch/Category/Guardian Details/Semester/Course Codes/Slots or leaves them empty, the system displays an appropriate error message. The Administrator/DEO returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: Registration Cancelled

If the Administrator/Student decides not to register the student, the registration 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.

SPECIFIC REQUIREMENT: none

ASSOSIATED USECASE: Login

B. Validity Checks

1. Only the Administrator/DEO will be authorized to access the maintain student details module.
2. Name(First Name, Last Name) cannot be blank.
3. Date of Birth cannot be blank.
4. Mobile Number cannot be blank.
5. Email ID cannot be blank.
6. The field to choose Department cannot be blank.
7. Address cannot be blank.
8. Guardian name cannot include special characters and digits, blank spaces are allowed.
9. Name(Last Name, First Name) cannot include special characters and digits, blank spaces are allowed.
10. Address can have length up to 10 to 200 characters.
11. Phone number can have exactly 10 digits.
12. Email ID can have up to 50 characters.
13. Email ID should contain '@' and '.' characters.
14. Email ID cannot include blank spaces.
15. Date of birth would be in the format mm/dd/yyyy.

C. Sequencing Information: None

D. Error Handling/ Response to abnormal situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.5 Maintain Marks :

Introduction: This use case documents the steps that the Administrator/Data Entry Operator(DEO) must follow in order to maintain student's marks details. This includes adding, updating and viewing student's marks information.

Actors: Administrator, DEO.

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

Post Conditions: If the use case is successful, then the student's marks information is added/updated/viewed from the system. Otherwise the system state remains unchanged.

Flow of Events:

Basic Flow:

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

1. The system requests that the Administrator/DEO specify the function he/she would like to perform (add marks, update marks, view marks).
2. Once the Administrator/DEO provides the requested information, one of the sub flows is executed:
   a. If the Administrator/DEO selects "Add Marks", the **Add Marks** sub flow is executed.
   b. If the Administrator/DEO selects "Update Marks", the **Update Marks** sub flow is executed.
   c. If the Administrator/DEO selects "View Marks", the **View Marks** sub flow is executed.

Basic Flow 1: Add Marks

1. The system requests that the Administrator/DEO enter Registration Number.
2. The Administrator/DEO enters the Registration Number.
3. The system retrieves and displays the student information.
4. The Administrator/DEO enters the marks of the student for various courses.
5. Once the Administrator/DEO enters the necessary information, the system stores the student information with the marks information.

Basic Flow 2: Update Marks

1. The system requests that the Administrator/DEO to enter Registration Number.
2. The Administrator/DEO enters the Registration Number.
3. The system retrieves and displays the student information.
4. Once the Administrator/DEO makes the necessary changes in information, the system stores the student information with the marks information.

Basic Flow 3: View Marks

1. The system requests that the Administrator/DEO to enter the Registration Number.
2. The system retrieves and displays the student information with marks information.

Alternative Flow:

Alternate Flow 1: Invalid Entry

If in the **Add Marks** or **Update Marks** flow, the Administrator/DEO enters invalid marks or leaves them empty, the system displays an appropriate error message. The Administrator/DEO returns to the basic flow and may re-enter the invalid entry. **Alternate Flow 2: Marks Already Entered** If in the **Add Marks** flow, marks have already been entered, the system displays an error message. The Administrator/DEO returns to the basic flow and may re-enter the Registration Number. **Alternate Flow 3: Student not Found** If in the **Update Marks** or **Add Marks** or **View Marks** flow, the student information with the specified code does not exist, the system displays an error message. The Administrator/DEO returns to the basic flow and may re-enter the Registration Number. **Alternate Flow 4: Update Cancelled** If in the **Update Marks** flow, the Administrator/DEO decides not to update the student's Marks, the update is cancelled and the basic flow is re-started at the beginning. **Alternate Flow 5: 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

B. Validation Checks

1. Only the Administrator/DEO will be authorized to access the maintain marks details module.
2. Roll number cannot be blank.
3. Length of roll number can be at most 11.
4. Roll number cannot special characters except '/'.
5. Marks field cannot be left blank.

C. Sequencing Information: None

D. Error Handling/ Response to abnormal situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.6 Upload Course material:

Introduction: This use case documents the steps that the Administrator/Faculty must follow in order to Upload Course Material/Assignments/Lectures.

Actors: Administrator, Faculty.

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

Post Conditions: If the use case is successful, then the assignment/lecture/Course Material is uploaded.

Flow of Events:

Basic Flow:

This use case starts when the Administrator/faculty wishes to Upload Course Material/Assignments/Lectures.

1. The system requests that the Administrator/faculty specify the function he/she would like to perform (Upload Course Material/Upload Assignments/ Upload Lectures.).
2. Once the Administrator/Faculty provides the requested information, one of the sub flows is executed:
   a. If the Administrator/ Faculty selects "Upload Lecture", the **Upload Lecture** sub flow is executed.
   b. If the Administrator/Faculty selects "Upload Assignment", the **Upload Assignment** sub flow is executed.
   c. If the Administrator/Faculty selects "Upload Course Material", the **Upload Course Material** sub flow is executed.

Basic Flow 1: Upload Lecture

The system request that the Administrator/Faculty to enter the lecture information. This includes:

- Course
- Semester
- Date of Submission

Once the Administrator/Faculty provides the requested information, the assignment can be uploaded.

Basic Flow 2: Upload Assignment

The system request that the Administrator/Faculty to enter the Assignment information. This includes:

- Course
- Semester
- Date
- Slot

Once the Administrator/Faculty provides the requested information, the lecture can be uploaded.

Basic Flow 3: Upload course Material

The system request that the Administrator/Faculty to enter the Course material information. This includes: ● Course ● Semester Once the Administrator/Faculty provides the requested information, the course material can be uploaded.
Alternative Flow: **Alternate Flow 1: Invalid Entry** If in the **Upload Lecture** or **Upload Assignment** or **Upload Course Material** flow, the Administrator/Faculty enters invalid Course ID/Date/Semester or leaves them empty, the system displays an appropriate error message. The Administrator/Faculty returns to the basic flow and may re-enter the invalid entry. **Alternate Flow 2: Lecture Already Exists** If in the **Upload Lecture** flow, a lecture with same semester, course ID, Date and slot already exists, the system displays an error message. The Administrator/Faculty returns to the basic flow. **Alternate Flow 3: 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

B. Validity Checks

1. Only the Administrator/Faculty will be authorized to access the Upload module.
2. Course cannot be blank.
3. Semester cannot be blank.
4. Date cannot be blank.
5. Slot cannot be blank.

C. Sequencing Information: None

D. Error Handling/ Response to abnormal situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.7 View Marks:

Introduction: This use-case allows the student/administrator to print his/her marks.
Actors: Administrator, Student.
Pre-Condition: The administrator/student must be logged in to the system.
Post-Condition: If the use case is successful the student can view his/her marks for any semester that is cleared by the student.
Flow of Events: **Basic Flow:** This use case starts when the Administrator/student wishes to view his/her marks. 1. The system requests that the Administrator/Student to specify the semester. 2. Once the Administrator/Student provides the requested information, the marks of the student are displayed. 3. A message is shown to print the mark-sheet of that particular semester. **Alternate Flow:** **Alternate Flow 1: Print Marks** If the student/administrator selects to print marks the print marks use case is called. **Alternate 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: Login

B. Validity Checks

1. Only the Student/Administrator will be authorized to access the View Marks module.

2. Semester can hold only those semesters cleared by the student.

C. Sequencing Information:

None

D. Error Handling/Response to Abnormal Situations :

If any of the validation/sequencing flow does not hold true, an appropriate error message will be prompted to the administrator/student for doing the needful.

3.2.8 Print Marks:

Introduction: This use-case allows the student/administrator to print student's marks.
ACTORS: Administrator, Student.
PRE-CONDITION: The administrator/student must be logged in to the system.
POST-CONDITION: If the use case is successful the student/administrator can print marks for any semester that is cleared by the student.
FLOW OF EVENTS: **BASIC FLOW:** This use case starts when the Administrator/student wishes to print marks. 1. The system requests that the Administrator/Student to confirm printing mark-sheet. 2. The mark-sheet is printed. **ALTERNATE FLOW:** **Alternate Flow 1: User Exits** This allows the user to exit at any time during the use case. The use case ends.
SPECIFIC REQUIREMENT: none
ASSOSIATED USECASE: Login

B. Validity Checks

1. Only the Student/Administrator will be authorized to access the Print Marks module.

C. Sequencing Information:

None

D. Error Handling/Response to Abnormal Situations :

If any of the validation/sequencing flow does not hold true, an appropriate error message will be prompted to the administrator/student for doing the needful.

3.2.9 Maintain Course details

INTRODUCTION: Maintain courses use-case allows the user to add, delete, update and view course.

ACTORS: Data Entry Operator, Administrator

PRE-CONDITION: The user must be logged in to the system.

POST-CONDITION: If the use-case is successful, a course is added, deleted , updated or viewed .

FLOW OF EVENTS:

BASIC FLOW:

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

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

2.Once the admin/Deo provides the requested information, one of the following subflows is Executed.

- If the admin/DEO selects "Add a course" then Add a course subflow is executed.
- If the admin/DEO selects "Delete a course" then delete a course subflow is executed.
- If the admin/DEO selects "Update a course" then update a course subflow is executed.
- If the admin/DEO selects "View a course" then view a course subflow is executed.

Basic Flow 1: Add a course :

- The system requests the admin/DEO to enter the course information. This includes
    - Name of the course.
    - code of the course.
    - Syllabus
    - Associated syllabus
- Once the admin/DEO provides the required information, The course is added to the system.

Basic Flow 2: Delete a course :

- The system requests the admin/DEO to enter the course code
- The admin/DEO enters the course code. The system retrieves and displays the required information
- The system prompts to confirm the deletion of the course
- The admin/DEO verifies the deletion.
- The System deletes the course.

Basic Flow 3: Update a course :

- The system requests the admin/DEO to enter the course code
- The admin/DEO enters the course code. The system retrieves and displays the required information
- The admin/DEO makes the desired changes to the course information.This include any of the information specified in add course use case
- Once the admin/DEO updates the information the system updates the course information with the updated information.

Basic Flow 4: View a course :

- The system requests the admin/DEO to enter the course code
- The admin/DEO enters the course code. The system retrieves and displays the required information

ALTERNATE FLOW:

Alternate Flow 1: Invalid Entry

If the Add a course or update a course flow, the actor enters invalid course code or leaves the course code, name empty, the system displays an appropriate error message. The actor returns to the basic flow and may re-enter the invalid entry. **Alternate Flow 2: Course already Exists** **I**f the course with a specified course code already exists, the system displays an error message. The actor returns to the basic flow and may re-enter the invalid entry. **Alternate Flow 3: Course not found** **I**f in the Update/delete or view course the course code entered does not exists, The system displays an error message. The actor returns to the basic flow and may re-enter the course code. **Alternate Flow 4: Deletion not allowed** If the course is already taken by some students then deletion will not be allowed. The actor returns to the basic flow and may re-enter the course code. **Alternate Flow 5: User Exits** This allows the user to exit at any time during the use case. The use case ends.
SPECIFIC REQUIREMENT: none
ASSOSIATED USECASE: Login

B. Validity Checks

1. Only the DEO/Administrator will be authorized to access the Maintain Course module.
2. Name of the course cannot be blank.
3. Code of the course cannot be blank.
4. Name of course cannot include special characters and digits, blank spaces are allowed.
5. Name of course can have a length from 3 to 50 characters.
6. Code of the course contains alphanumeric characters. Special characters and blank spaces are not allowed.
7. Code of the course should have a length of 5 characters.
8. Credits should not be blank.
9. In case of adding a course the same course name or code should not be used before.
10. In case of deleting a course the course code should be present in the system.
11. Credits should be a single digit number greater than 0.

C. Sequencing Information: None

D. Error Handling/Response to Abnormal Situations :

If any of the validation/sequencing flow does not hold true, an appropriate error message will be prompted to the administrator/Faculty member for doing the needful

3.2.10 View Attendance

INTRODUCTION : View Attendance use-case allows the faculty member and students to view attendance.
ACTORS: Students, Administartor.
PRE-CONDITION: The user must be logged in to the system.
POST-CONDITION: If the use-case is successful, the student would be able to see his/her attendance and the percentage of attendance.
FLOW OF EVENTS: **BASIC FLOW:** • The Student selects the month /week/day for which the student needs to see his/her attendance. • The system displays the required information. **ALTERNATE FLOW:** **Alternate Flow 1: User Exits** The use case ends and the system state remains unchanged.
SPECIFIC REQUIREMENT: none
ASSOSIATED USECASE: Maintain Attendance, Login.

B. Validity checks

1. Roll No: Alphanumeric of length 3 to 100 characters. Special characters are not allowed except underscore. Numeric data will be allowed.
2. Password cannot be blank.
3. Length of password can only be 4 to 15 characters.
4. Alphabets, digits, hyphens and underscore characters are allowed in the password field.
5. Password will not accept blank spaces.

C. Sequencing Information

None

D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.11 Maintain Attendance

INTRODUCTION: Maintain attendance use-case allows the faculty member and administrator to mark attendance of the student.
ACTORS: Administrator, Faculty member.
PRE-CONDITION: The user must be logged in to the system.
POST-CONDITION: If the use-case is successful, the attendance of a student is marked.
FLOW OF EVENTS: **BASIC FLOW:** • The user enters the roll number of the student. • The user enters selects the current date and time. • The system records the current date and the attendance of the student is marked for that date. **ALTERNATE FLOW:** **Alternate Flow 1: Invalid Roll Number** If the roll number entered by the user is incorrect or is not available then an error message is shown and the use-case restarts. **Alternate Flow 2: User Exits** The use case ends and the system state remains unchanged.
SPECIFIC REQUIREMENT: none
ASSOSIATED USECASE: Login

B. Validity checks

1. Roll No: Alphanumeric of length 3 to 100 characters. Special characters are not allowed except underscore. Numeric data will be allowed.
2. Attendance status can be P or AB .

C. Sequencing Information

None

D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.12 Maintain Faculty Details

INTRODUCTION: Maintain faculty use-case allows the user to add, delete, update and view faculty details.

ACTORS: Data Entry Operator, Administartor

PRE-CONDITION: The user must be logged in to the system.

POST-CONDITION: If the use-case is successful, a faculty member is added , deleted , updated or viewed .

FLOW OF EVENTS:

BASIC FLOW:

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

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

2.Once the admin/Deo provides the requested information, one of the following subflows is Executed.

- If the admin/DEO selects "Add a faculty member" then Add a faculty member subflow is executed.
- If the admin/DEO selects "Delete a faculty member" then delete a faculty member subflow is executed.
- If the admin/DEO selects "Update a faculty member" then update a faculty member subflow is executed.
- If the admin/DEO selects "View a faculty member" then view a faculty member subflow is executed.

Basic Flow 1: Add a faculty member:

- The system requests the admin/DEO to enter the faculty information. This includes
  - Name of the faculty member.
  - Faculty id
  - Address
  - Telephone
  - Email
  - Password
  - Photograph
- Once the admin/DEO provides the required information, The faculty member is added to the system.

Basic Flow 2: Delete a faculty member:

- The system requests the admin/DEO to enter the faculty member id.
- The admin/DEO enters the faculty member id. The system retrieves and displays the required information.
- The system prompts to confirm the deletion of the faculty member.
- The admin/DEO verifies the deletion.
- The System deletes the faculty member.

Basic Flow 3: Update a faculty member:

- The system requests the admin/DEO to enter the faculty member id.
- The admin/DEO enters the faculty member id. The system retrieves and displays the required information.

- The admin/DEO makes the desired changes to the faculty member information. This include any of the information specified in add faculty member use case
- Once the admin/DEO updates the information the system updates the faculty member information with the updated information.

Basic Flow 4: View a course :

- The system requests the admin/DEO to enter the faculty member id.
- The admin/DEO enters the faculty member id. The system retrieves and displays the required information

ALTERNATE FLOW:

Alternate Flow 1: Invalid Entry

If the Add a course or update a faculty member flow, the actor enters invalid faculty member id or leaves the faculty member id, name empty, the system displays an appropriate error message. The actor returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 2: faculty member already Exists

If the faculty member with a specified faculty member id already exists, the system displays an error message. The actor returns to the basic flow and may re-enter the invalid entry.

Alternate Flow 3: faculty member not found

If in the Update/delete or view course the course code entered does not exists, The system displays an error message. The actor returns to the basic flow and may re-enter the faculty member id.

Alternate Flow 4: User Exits

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

SPECIFIC REQUIREMENT: none

ASSOSIATED USECASE: Login

B. Validity Checks

2. Only the Administrator will be authorized to access the maintain faculty details module
3. Every registered faculty will have a unique faculty id.
4. Faculty ID cannot be blank.
5. Name(First Name, Last Name) cannot be blank.
6. Date of Birth cannot be blank.
7. Mobile Number cannot be blank.
8. Email ID cannot be blank.
9. Joining Date cannot be blank.
10. Educational Qualifications cannot be blank.
11. The field to choose Associated Department cannot be blank.

C. Sequencing Information: None

D. Error Handling/ Response to abnormal situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.2.13 Download Course Material

INTRODUCTION : Download Course Material use-case allows the Student to download the course material from the list of materials for various courses.

ACTORS: Students, Administartor.

PRE-CONDITION: The user must be logged in to the system.

POST-CONDITION: If the use-case is successful, the student would be able to download the course material for the particular course.

FLOW OF EVENTS:

BASIC FLOW:

This use case starts when the Administrator/faculty wishes to Download Course Material/Assignments/Lectures.

3. The system requests that the Administrator/faculty specify the function he/she would like to perform (Download Course Material/download Assignments/ download Lectures.).
4. Once the Administrator/Faculty provides the requested information, one of the sub flows is executed:
   d. If the Administrator/ Faculty selects "Download Lecture", the **download Lecture** sub flow is executed.
   e. If the Administrator/Faculty selects "Download Assignment", the **download Assignment** sub flow is executed.
   f. If the Administrator/Faculty selects "Download Course Material", the **download Course Material** sub flow is executed.

Basic Flow 1: Download Lecture

The system request that the Administrator/Faculty to enter the lecture information. This includes:

- Course
- Semester

Once the Administrator/Faculty provides the requested information, the assignment can be downloaded.

Basic Flow 2: Download Assignment

The system request that the Administrator/Faculty to enter the Assignment information. This includes:

- Course
- Semester
- Date
- Slot

Once the Administrator/Faculty provides the requested information, the lecture can be downloaded.

Basic Flow 3: Download course Material

The system request that the Administrator/Faculty to enter the Course material information. This includes:

- Course
- Semester

Once the Administrator/Faculty provides the requested information, the course material can be downloaded.

ALTERNATE FLOW:

Alternate Flow 1: Course not found

If in the Update/delete or view course the course code entered does not exists, The system displays an error message. The actor returns to the basic flow and may re-enter the course code.

Alternate Flow 2: User Exits

The use case ends and the system state remains unchanged.

SPECIFIC REQUIREMENT: none

ASSOSIATED USECASE: Login.

B. Validity Checks

1.Code of the course cannot be blank.

2.Code of the course contains alphanumeric characters. Special characters and blank spaces are not allowed.

3.Code of the course should have a length of 5 characters.

C. Sequencing Information: None

D. Error Handling/Response to Abnormal Situations :

If any of the validation/sequencing flow does not hold true, an appropriate error message will be prompted to the administrator/Faculty member for doing the needful

3.2.13 Print Attendance

Introduction: This use-case allows the student/administrator to print student's attendance.
ACTORS: Administrator, Student.
PRE-CONDITION: The administrator/student must be logged in to the system.
POST-CONDITION: If the use case is successful the student/administrator can print attendance for any month.
FLOW OF EVENTS: **BASIC FLOW:** This use case starts when the Administrator/student wishes to print attendance. 1. The system requests that the Administrator/Student to confirm printing attendace. 2. The attendance is printed. **ALTERNATE FLOW:** **Alternate Flow 1: User Exits** This allows the user to exit at any time during the use case. The use case ends.
SPECIFIC REQUIREMENT: none
ASSOSIATED USECASE: Login

B. Validity checks

1. Roll No: Alphanumeric of length 3 to 100 characters. Special characters are not allowed except underscore. Numeric data will be allowed.

C. Sequencing Information

None

D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

3.3 Performance Requirements

- Should run on 500 MHz, 512 MB RAM machine.
- Responses should be within 2 seconds.

3.4 Design Constraints

none

3.5 Software System Attributes

3.5.1 Usability

The application will be user friendly and easy to operate and the functions will be easily understandable.

3.5.2 Reliability

The application will be available to student throughout the semester and have a high degree of fault tolerance.

3.5.3 Security

The application will be password protected Users will have to enter correct user id and password to access the system..

3.5.4 Maintainability

Any updates or defect fixes shall be able to be made on server-side computers only, without any patches required by the user.

3.5.5 Portability

The application will be easily portable on any windows based system that has SQL server installed.

3.6 Logical Structure of the Data

The following information will be placed in a database:

Table name	Description
Login	Records the login details of the user
Student	Records the details of the students.
Faculty	Records the details of the faculty.
Course	Records the details of the courses.
Courses Selected	Records the details of the courses opted by the student.
Attendance Record	Record the details of the attendance of the student for each course opted by the student.
Marks Record	Records the marks of the student for each course opted by him/her.
Lectures	Records the lecture of the various courses by different faculties.
Assignments	Records the assignments uploaded by the faculties for various courses.
Course/Study Material	Records the course/study material uploaded by the faculties for various courses.

3.7 Other Requirements