TABLE OF CONTENTS

1. INTRODUCTION **3**
1.1 PURPOSE 4
1.2 INTENDED AUDIENCE 4
1.3 CONCLUSION 5

2. INCEPTION OF COURSE MANAGEMENT SYSTEM **5**
2.1 INTRODUCTION 6
2.2 EXISTING STORY 6
2.3 INCEPTION PROCEDURE 7
2.3.1 IDENTIFY THE CLIENT OF OUR PROJECT 7
2.3.2 ICEBREAKING 7
2.3.3 IDENTIFYING THE STAKEHOLDERS OF THE COURSE MANAGEMENT SYSTEM 8
2.3.4 IDENTIFYING THE MULTIPLE VIEWPOINTS OF THE STAKEHOLDER 8
2.4 CONCLUSION 9

3. ELICITATION OF CMS **10**
3.1 INTRODUCTION 10
3.2 ELICITING REQUIREMENTS 10
3.2.1 COLLABORATIVE REQUIREMENTS GATHERING 11
3.2.2 QUALITY FUNCTION DEPLOYMENT 11
3.2.3 USAGE SCENARIO 14

4. SCENARIO BASED MODELING **21**
4.1 INTRODUCTION 21
4.2 DEFINITION OF USE CASE 21
4.3 USE CASE DIAGRAMS 23
4.3.1 LEVEL-0 23
4.3.2 LEVEL-1 24
4.3.3 LEVEL-1.1 25
4.3.4 LEVEL-1.1.2 27
4.3.5 LEVEL-1.2 29
4.3.6 LEVEL-1.3 32
4.3.7 LEVEL 1.3.2 34
4.3.8 LEVEL 1.3.4 36
4.3.9 LEVEL 1.4 38

4.3.10 LEVEL 1.5 39
4.3.11 LEVEL 1.6 41
4.3.12 LEVEL 1.7 43
4.3.13 LEVEL 1.8 45

5. ACTIVITY DIAGRAM OF COURSE MANAGEMENT SYSTEM 47

6. SWIMLANE DIAGRAM OF COURSE MANAGEMENT SYSTEM 61

5. DATA MODELLING OF COURSE MANAGEMENT SYSTEM 76

5.1 Data Modelling Concept 76
5.1.1 Data Objects 77
5.1.1.1 NOUN IDENTIFICATION 77
5.1.1.2 Potential Data objects: 80
5.1.1.3 Analysis for finalizing Data Objects 81
5.1.1.4 Final Data Objects 82
5.2 DATA OBJECT RELATIONSHIPS 83
5.3 ENTITY RELATIONSHIP DIAGRAM 84
5.4 SCHEMA DIAGRAM 86

6. CLASS-BASED MODELING FOR CMS 91

6.1 CLASS BASED MODELING CONCEPT 91
6.2 Noun List for CMS 91
6.3 Verb List of CMS 93
6.4 GENERAL CLASSIFICATION 94
Table:1 General Classification 95
6.5 SELECTION CRITERIA 98
Table:2 Selection Criteria 99
6.6 ATTRIBUTE SELECTION 100
Table:3 Attribute Selection 100
6.7 METHOD IDENTIFICATION 104
Table:4 Method Attributes 104
6.8 Analyzing Classes 112
6.9 Class Card 113
Table:5 Class Card for User 113
Table: 6 Class Card for Student 114
Table:7 Class Card for Teacher 115
Table:8 Class Card for System 116
Table: 9 Class Card for Database 117
Table: 10 Class Card for course Creation 118

Table:11 Class Card for section 118
Table:12 Class Card for Exam 119
Table:13 Class Card for course Plan 119
Table:14 Class Card for Result 120
Table:15 Class Card for Re-Examine 121
Table:16 Class Card for Feedback 122
Table:17 Class Card for Assignment 122
Table:18 Class Card for Dashboard 123
Table:19 Class Card for course content 123
6.10 CRC Diagram 125

7. BEHAVIORAL MODELING OF CMS 126
7.1 State Diagrams 130
7.2 SEQUENCE DIAGRAM: 136

8. DFD 136
LEVEL 0: 138
LEVEL 1: 139
LEVEL 1.1: 140
LEVEL 1.1.2: 141
LEVEL 1.2: 142
LEVEL 1.3: 143
LEVEL 1.4: 144
LEVEL 1.5: 145
LEVEL 1.6: 146
LEVEL 1.7: 147
LEVEL 1.8: 148

1. INTRODUCTION

This chapter is a part of our software requirement specification for the project “Course Management System”. In this chapter we will focus on the intended audience for this project.

1.1 PURPOSE

This document briefly describes the Software Requirement Analysis of Course Management System. It contains the functional, non-functional and the supporting requirements and establishes a requirement’s baseline for the development of the system. The requirements contained in the SRS are independent, uniquely numbered and organized by topics. The SRS serves as an official means of communicating user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

1.2 INTENDED AUDIENCE

This SRS report is intended for several audiences including the customers as well as the project managers, designers, developers, and testers. The customer will use this SRS to verify that the developer team has created a product that is acceptable to the customer. The project managers of the developer team will use this SRS to plan milestones and a delivery date, and ensure that the developing team is on track during development of the system. The designers will use this SRS as a basis for creating the system’s design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer’s needs. The developers will use this SRS as a basis for developing the

system's functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created a software that will fulfill all of the customer's documented requirements. The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

1.3 CONCLUSION

This analysis of the audience helped us to focus on the users who will be using our analysis. This overall document will help each and every person related to this project to have a better idea about the project.

2. INCEPTION OF COURSE MANAGEMENT SYSTEM

In this chapter, the Inception part of the SRS will be discussed briefly.

2.1 INTRODUCTION

A **Course Management System** (CMS) is a collection of software tools providing an online environment for **course** interactions. A CMS typically includes a variety of online tools and environments, such as: An area for faculty posting of class materials such as **course** syllabus and handouts (Source: Vanderbilt). Moreover, it enables the instructor to create online course content and post it on the Web without having to handle HTML or other programming languages (Source: Technopedia). This document will act as a demand list of what the stakeholders of their respective wanted for their anticipated software to be. Like the various features and interfaces of the software which will help them to ease the way of their work.

2.2 EXISTING STORY

Currently, a course is fully managed by a teacher. The tasks performed by a teacher are:

1. Make Course Plan
2. Make Marks Distribution
3. Create Exam Question
4. Evaluate Exam Script
5. Take Attendance
6. Check Assignments
7. Calculate Total Marks
8. Re-Examine if needed
9. Take Classes

2.3 INCEPTION PROCEDURE

At the beginning of our project, we entered the inception stage. This stage includes, how the project will be started and their ***scope*** and ***limitations***. The main goal of this phase is to ***identify the requirements***, ***demand*** and establish some sort of ***mutual understanding*** between the software team and the stakeholders of the Course Management System(CMS). In order to make this phase effective followed some steps namely:

a. Identifying the client of our project
b. Icebreaking
c. Identifying the stakeholders of the Course Management System(CMS)
d. Identifying the multiple viewpoints of stakeholders

2.3.1 IDENTIFY THE CLIENT OF OUR PROJECT

At first, we identified the location from where we will start our expedition. The most probable location was universities and we could not find any better than the University of Dhaka. The most important stakeholder of our project is the Teachers. We found out that the students are also equally important as the teachers. These two kinds of people are our main client. We have analyzed our requirements with the consent of both of them.

2.3.2 ICEBREAKING

Icebreaking refers to diminishing the communication barrier between you and the other person. It is a crucial part since it decides the acceptance of our proposal. We started this phase by talking with them with context free languages. Their

behavior, response to our question or willing to accept the new course management system solely depends on this phase.

2.3.3 IDENTIFYING THE STAKEHOLDERS OF THE COURSE MANAGEMENT SYSTEM

Stakeholder refers to any person or group who will be affected directly or indirectly by the system. Stakeholders include end-users who interact with the system and everyone else in an organization who may be affected by its installation. Course Management System has a limited number of stakeholders. Identification of the stakeholders were done from the information provided by the teachers and the students. It turned out that we don't have any stakeholders other than the clients.

2.3.4 IDENTIFYING THE MULTIPLE VIEWPOINTS OF THE STAKEHOLDER

Different stakeholders expect different benefits from the system as every person has his own point of view. So, we have to recognize the requirements from multiple viewpoints. Different viewpoints of the stakeholders about the expected software are given below:

Teacher's Viewpoint:

1. Software should manage the whole course.
2. Teacher should be able to plan the course prior to the start of the course.
3. Teacher should be able to announce notices to the students.
4. Teacher should be able to select his/her marking criteria for the whole course.
5. Teacher should be able to allocate the weightage of each criteria.
6. Teacher should be able to upload any related study contents to help the students.
7. Teacher should be notified if any event is nearby.

8. Teacher should be able to plan an event.
9. Teachers should be able to view the assignments through the software.
10. Attendance management should be done by the software.
11. Teacher should be able to design a mark scheme for any sort of exams.
12. The software will evaluate marks and GP of the students

Student's Viewpoint:

1. Students want to get notices through software.
2. Students want to submit assignments through software.
3. Students believe that automation of Course Management will ensure impartial marking and errors will be less.
4. Students want a user friendly interface.
5. Students want to see their current marks through the software.
6. Students tend to be more excited about the automation of the Course Management System than the teachers.

2.4 CONCLUSION

The primary goal of this project is to model and design a software for those people who are related to a course. For these reasons, the software will be designed in such a way that it will bring comfort to the client who will use it. The software will be simple and user friendly. Otherwise, it will not be appreciated by the clients. The software will be designed in such a way that it takes very little time to manage. To make this software project successful, collaboration with stakeholders is the main priority. To sum up, what they want, how the software will work and how it can be more efficient than the earlier times is the main concern in this phase.

3. ELICITATION OF CMS

After discussing the Inception phase, we need to focus on the Elicitation phase. So this chapter specifies the Elicitation phase.

3.1 INTRODUCTION

Requirements Elicitation is a part of requirements engineering that is the practice of gathering requirements from the stakeholders. We have faced many difficulties, like understanding the problems, making questions for the stakeholders, limited communication with stakeholders due to shortage of time and volatility of the stakeholders. Though it is not easy to gather requirements within a very short time, we have surpassed these problems in an organized and systematic manner.

3.2 ELICITING REQUIREMENTS

We have seen Question and Answer (Q&A) approach in the previous chapter, where the inception phase of requirement engineering has been described. The main task of this phase is to combine the elements of problem solving, elaboration, negotiation and specification. The collaborative working approach of the stakeholders is required to elicit the requirements. We have finished the following tasks for eliciting requirements-

- Collaborative Requirements Gathering
- Quality Function Deployment
- Usage Scenarios
- Elicitation of Work Products

3.2.1 COLLABORATIVE REQUIREMENTS GATHERING

We have met with 12 stakeholders in the Inception phase. To be more specific we have met 6 teachers and 6 students. These meetings created an indecisive state for us to elicit the requirements. To solve this problem, we have met with some stakeholders (who are acting a vital role in the whole process) more than once to elicit the requirements.

3.2.2 QUALITY FUNCTION DEPLOYMENT

❑ Normal

1. **Course Creation by the Teacher:**
   An authorized teacher will be able to create a course.

2. **Course Plan:**
   The teacher will be able to create a course plan for the course. The course plan will include number of classes, estimated mid term date, estimated final date, probable assignments, probable lab works and others.

3. **Marks Distribution of Course:**
   Teacher should be able to make the marks distribution. He will be able to select and add fields and also allocate the marks percentage for every field.

4. **Exam Section:**

Teacher will be able to create question, make a marking model for the question, check exam script and the numbers will be added to the database automatically.

5. **Uploading Content Files:**
   The course teacher will upload the necessary content files. The content files include documents, reference videos.

6. **Announce Notice and Exam Schedules:**
   Course Teacher will announce exam schedules and notices earlier in the software which will reach every concerned student's dashboard and email.

7. **Submission of Assignment:**
   Assignments can be submitted to the teacher through the software. The teacher can also return feedback to the students through the software.

8. **Attendance Management:**
   Attendance will be taken automatically in every class and all records will be kept and integrated with the grade sheet later on.

9. **Evaluate Mark and GP:**
   Marks from all the fields will be compiled as per their respective weights and corresponding GPA will be calculated automatically.

10. **Privacy of Grade Sheet:**
    The teacher will decide whether the total grade sheet will be accessible by everyone or students can view their individual results only

11. **Rechecking the exam paper:**
    If any student is not satisfied with the marks and results he made, he may ask the teacher to re-evaluate his exam scripts and the result will be updated after the re-evaluation. The student may give feedback too.

12. Dashboard for Teacher and Student:
There will be separate interfaces for teachers and students in the software for a certain course.

❑ Expected

- **Send Invitation to students:**
  Enrolled Students will be invited to join the particular course in the software through email.

- **Students will get results & important notices through email:**
  The final marks, GP and important notices will be sent to the enrolled students through email.

- **Grade Sheet will be prepared for teacher:**
  The results will be compiled together and a grade sheet will be prepared by the software which will be sent to the teacher after all the course completes.

- **Students will be able to give feedback on Re-examined result:**
  The student will be able to give feedback on the script which he asked the teacher to re-examine.

- **Concurrency:**
  Parallel use of all the users of a course will be ensured.

- **Scalability:**
  Efficiency will be maintained while all the users use the software.

❑ Exciting

- **Make a mobile app for user:**
  A mobile app will be created corresponding to the software to provide comfort to the users. The users will be able to access the app from anywhere and anytime.

- **Attendance through facial recognition:**
  Attendance will be taken through facial recognition. The photo will be taken with the help of the CCTV camera in the class.

- **Video and Live Classes:**
  The classes will go live if the teacher permits and videos of the classes will be uploaded automatically by the software taking permission from the teacher.

- **Warning the student:**
  If marks or attendance of a student is very low, a warning message will appear in the dashboard of the student.

3.2.3 USAGE SCENARIO

A user can be a teacher or a student. But a user can not be a teacher and student of the same course simultaneously.

Authentication and Authorization:

- At the very beginning, a user will be allowed to create an account or log in.

- If the user has no account, he can create account. He has to provide full name, a photo, user ID, university, department, institutional mail, phone number, photo, username and password.
- The system will verify the information from the university database and notify the user through email. The system will also check the email and username. If the email or the username already exists, then the system will request user to re-enter email or username .The user will be given 5 minutes to respond to the confirmation mail. If s/he fails, s/he needs to start fresh again.
- The account creation will be finalized by clicking the link on the confirmation mail.
- If s/he is an authorized user then s/he can directly login providing username and password. If the user forget the necessary information to log in, then he will get chance to recover information via email.

Dashboard 1:

- After authorization, the user will get a dashboard where s/he will have the courses he registered to and the courses he/she is a teacher of.

DASHBOARD
Teaching: - SRS(405) - BUS(410) - *Create a new course*
Learning: - CSE(801) - SE(805)

Course Creation:

- If a user wants to create a new course, his/her identity will be verified from the university database whether he is a teacher or not.
- When a teacher creates a course with the course name and course code, the course will be available in the teaching section of the dashboard of the user(teacher). In the meantime, an invitation email will be sent to the students who are selected by the teacher. Clicking a link from the invitation mail will automatically register the invited student to a certain course and the courses will be available in the student section dashboard of the user(student).

Setting Course Plan:

- The teacher can set the course plan. The course plan includes number of classes, estimated midterm date, estimated final date, probable assignments and probable lab works.

Category	**Sub-Category/ Number**	**Date**
Number of Classes	20 classes	1/1/20 to 31/4/20
Mid Term	1	1/3/20
Add Categories	*Add Sub-Categories*	*Add Date*

Fields Allocation and Marks Distribution:

- Teacher can create templates for marks distribution from some selected criteria(he can also add some fields if needed).
- Teachers can allocate marks for different fields of the template created.

Fields	**Marks Percentage**
Final Exam	33%
Mid Exam	20%
Lab	33%
Add Fields	*Add Marks Percentage*

Exam System:

- A course teacher can create question paper, answer sheet for question, check script and records marks in database. Then the system calculate total marks and GP. The course teacher can send the result to the student via email or upload to dashboard or do both.
- Teachers will get a grade sheet automatically which will be calculated by the software itself.

Question	**Marks Distribution**	**Marks**	**Result**
1.What is Software Engineering	a.Understanding	1	
	b.Definition	1	
	Add more		
2.Describe Scrum Method	a.Understanding	3	
	b.Definition	1	

	Add more
Add more	*Add more*

Fig: ANSWER SHEET

Upload:

- The teacher can upload class contents, study materials, books, video contents (Of the previous live classes and referenced tutorial), results and notices. Notices will be sent to all enrolled students via email for every upload.
- Students can upload assignments.

Download:

- Teacher can download assignments, grade sheet.
- Students can download class contents, study materials, books, video contents (Of the previous live classes and referenced tutorial) and grade sheet.

View:

- Teacher and Students both can view class contents, results and notices.

Attendance:

- Teacher can take attendance through the software. A photo of the class can be taken with the CCTV camera. Through facial recognition technology, individuals will be identified and attendance will be given.

Online Classes:

- Teacher can go live in classes and video will be uploaded by the software later with the permission of the teacher.
- Students can attend classes online through video streaming. In that case, the student will miss the attendance.

Warning System:

- Students below the attendance 60% cannot participate in exam. Students with critical attendance level will be sent an email as a warning and the students with low marks will also be warned via this system.

Dashboard 2:

- There will be different dashboards for teachers and students.
- **Student's Dashboard:** Students can see their previously uploaded assignments and the upcoming and pending assignments. The dashboard will show if there is any notice. Marks of particular fields will be uploaded in the dashboard. Course Contents will be shown in dashboard.

- **Teacher's Dashboard:** Teachers will see the upcoming events, the submitted assignments, a marksheet of all the students where s/he can add marks of some fields(marks of other fields will be updated by the software like attendance, final exams etc.)

Re-examining the Result:

- A student can request for re-examining his result to the course teacher. Then the teacher re-check his result and send the re-examined result . The student can give his feedback on the re-examined result.

Feedback:

- Student can give feedback at the end of the course for re-examined scripts.

4. SCENARIO BASED MODELING

This chapter describes the Scenario Based Model for the "**Course Management System**"

4.1 INTRODUCTION

Although the success of a computer-based system or product is measured in many ways, user satisfaction resides at the top of the list. If we understand how end users (and other actors) want to interact with a system, our software team will be better able to properly characterize requirements and build meaningful analysis and design models. Hence, requirements modeling begins with the creation of scenarios in the form of Use Cases, activity diagrams and swim lane diagrams.

4.2 DEFINITION OF USE CASE

A Use Case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a Use Case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A Use Case diagram simply describes a story using corresponding actors who perform important roles in the story and makes the story understandable for the users. The first step in writing a Use Case is to define that set of "actors" that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be described. Actors represent the roles that people play as the system operators. Every user has one or more goals when using system. Actors either consume/ produce/ modify information.

Primary Actor

Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software. The actors who do more than one job(consume/produce/ manipulate information) are our primary actors.

Secondary Actor

Secondary actors support the system so that primary actors can do their work. They either produce or consume information.

4.3 USE CASE DIAGRAMS

Use Case diagrams give the non-technical view of overall system.

4.3.1 LEVEL-0

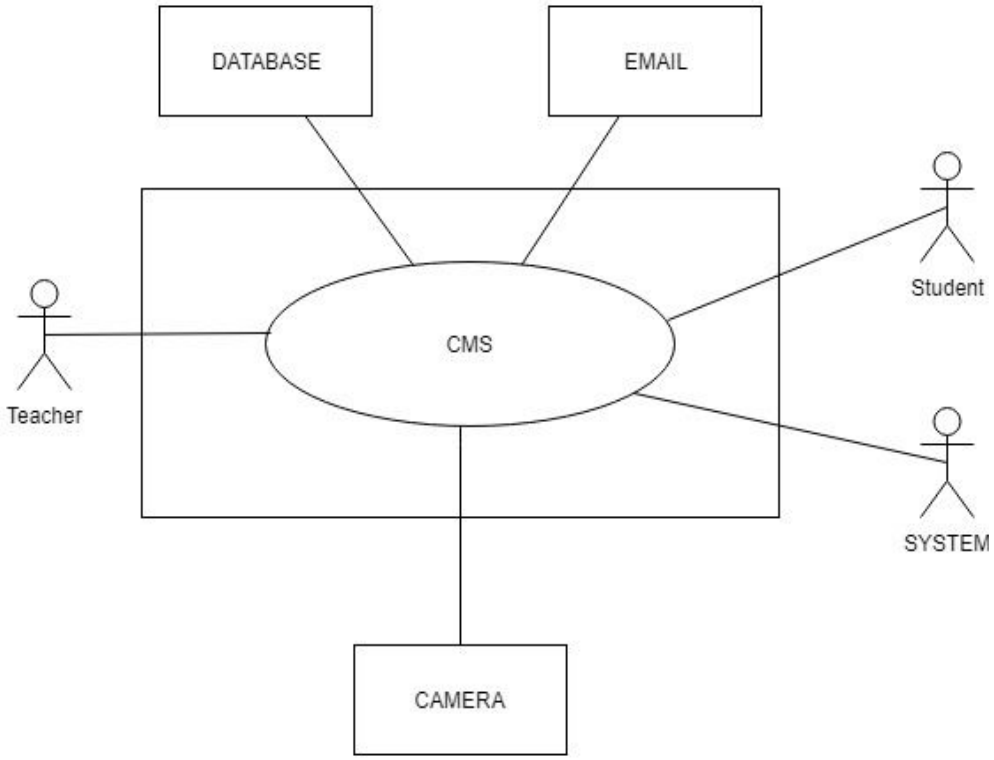


Figure 1: CMS_cms

Description of Use case diagram level-0:

After analyzing the user's story, we found 5 actors that directly or indirectly interacts with the system. Primary actors are those who will play action and get a

reply from the system whereas secondary actors only produce or consume information.

Primary Actors: User,Student,Teacher,Database

Secondary Actors: E-mail ,Camera

4.3.2 LEVEL-1

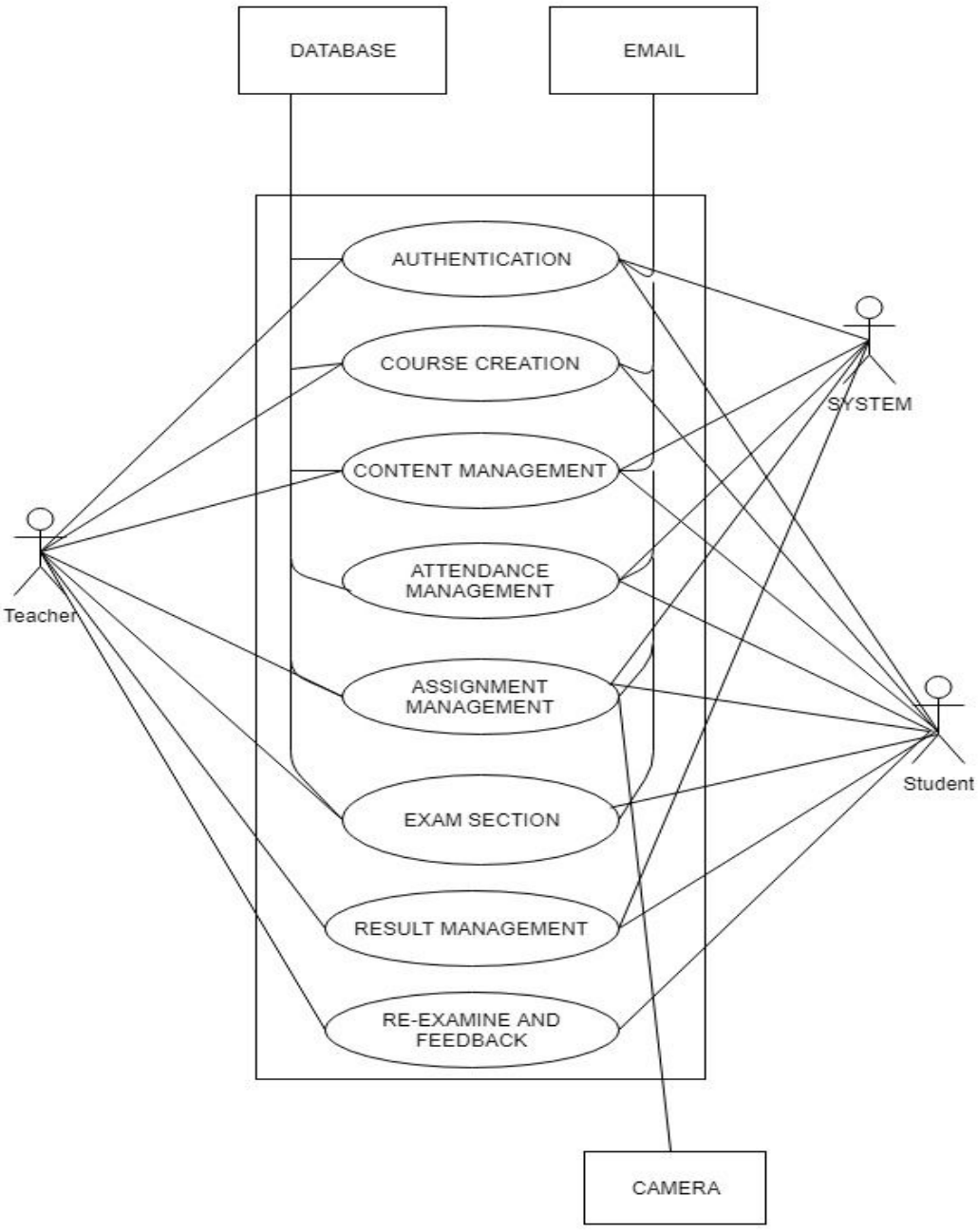


Figure 2: CMS_Sub-Systems

Description of Use case diagram level-1:

There are 8 subsystems in Course Management System. They are as follows:

1. Authentication

2. Course Creation

3. Content Management

4. Attendance Management

5.Assignment Management

6.Exam Section

7.Result Management and

8.Re-examine and Feedback

The subsystems are further decomposed, in level 1.1, 1.2, 1.3 , 1.4, 1.5, 1.6, 1.7, 1.8 respectively.

Primary Actors: User,Student,Teacher,Database

Secondary Actors: E-mail ,Camera

4.3.3 LEVEL-1.1

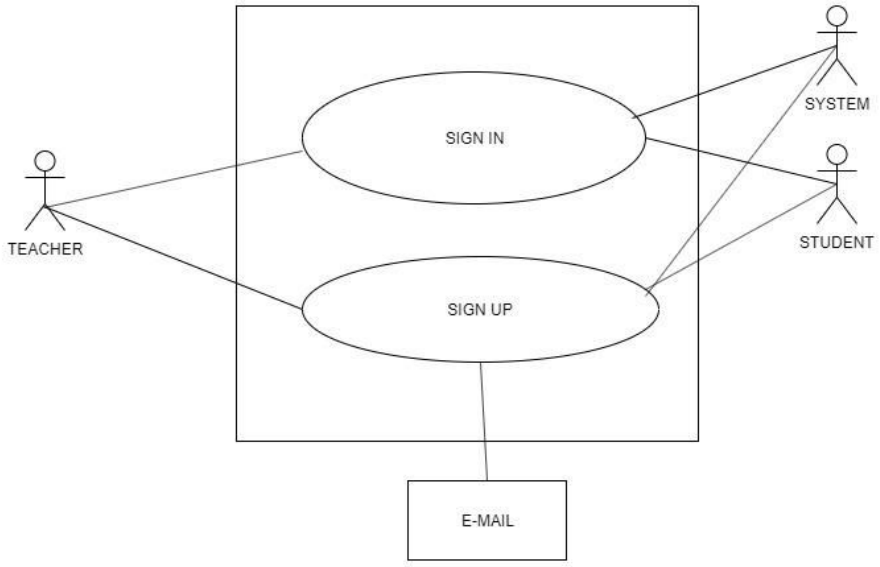


Figure 3: CMS_Authentication

Description of Use case diagram level-1.1:

Authentication is a process in which credentials provided are compared to those on files in a database of authorized user's information. The authentication subsystem can be divided into two parts. They are as follows:

1. Sign In

2. Sign Up

The Sign Up subsystem are further decomposed in level 1.1.2.

Primary Actors: User,Student,Teacher,Database ,System

Secondary Actors: E-mail

Action-Reply

A1: User provides credentials.

R1: System will check the validity of the given credentials. For valid information system will allow user(Student or Teacher) to log into the account.

A2: User provides invalid credentials.

R2: System will show error message and allows to try again.

Exception:

A3: User fails to log into the account for third time.

R3: System will be blocked for two minutes.

4.3.4 LEVEL-1.1.2

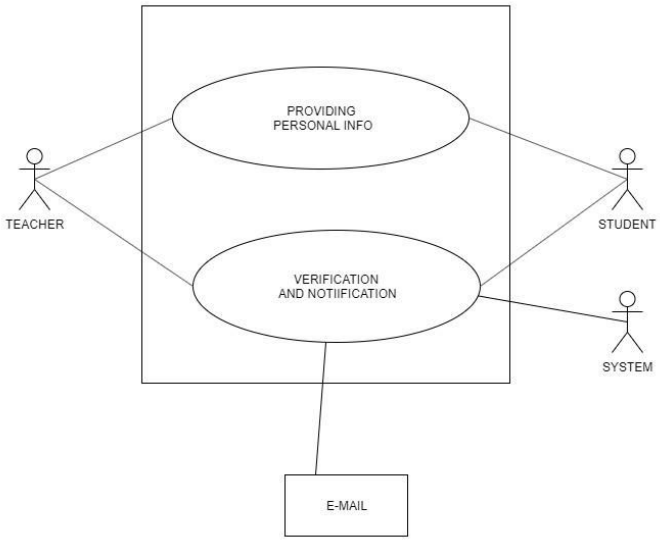


Figure 4: CMS_SignUp

Description of Use case diagram level-1.1.2:

At the time of signing up a user has to provide his/her personal information(full name, a photo, user ID, university, department, institutional mail, phone number, photo, username and password).This information will be stored in database.

Primary Actors: Student,Teacher,Database,System

Secondary Actors: E-mail

Action Reply

A1: Student/Teacher creates an account filled with valid information.

R1:If the information is valid , the System creates an account and the account information is stored. The user will be notified through email.

A2: User creates an account filling with invalid information.

R2: System allows the user to try again for providing information.

Note: Password should be at least 8 characters at most 32 characters

4.3.5 LEVEL-1.2

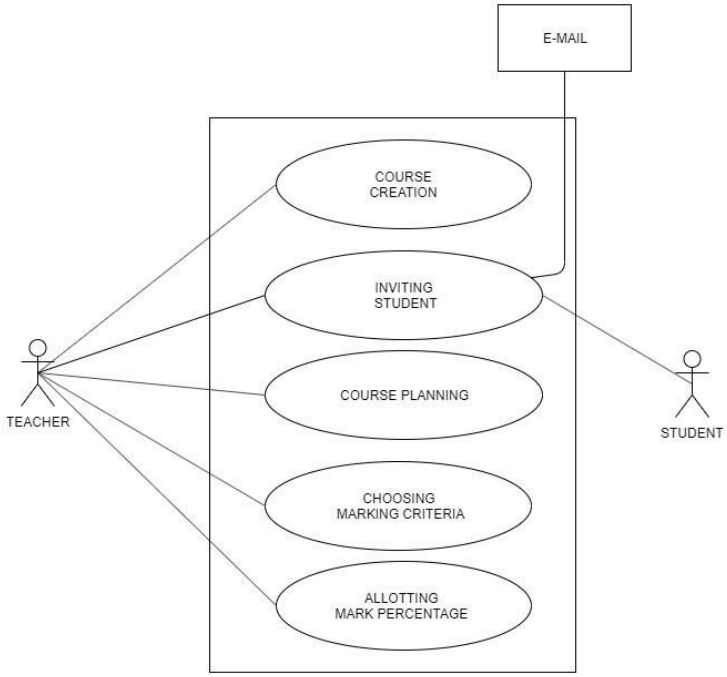


Figure 5: CMS_Course_Creation

Description of Use case diagram level-1.2:

Anything related to course creation will be handled by this subsystem. This subsystem is divided into five parts.

1.Course creation

2.Inviting Students

3.Course Planning

4.Choosing Marking Criteria

5.Allotting Mark Percentage

When a teacher wants to create a course with the course name and course code, the system will allow him to create a course. In the meantime, if s/he wants to add students in the course ,s/he will select an invitation email will be sent to the student.

The teacher can set the course plan. The course plan includes number of classes, estimated midterm date, estimated final date, probable assignments and probable lab works.

Teacher can create templates for marks distribution from some selected criteria(he can also add some fields if needed) as well as he can allocate mark percentage to the selected criteria.

Primary Actors: Student,Teacher,System

Secondary Actors: E-mail

Action Reply

Teacher:

A1: Teacher will give course name and course code to create a course

R1: System will justify whether he is a teacher or not. If he is a teacher, system will allow him to create a course.

A2: After creating a course s/he will invite students

R2: Invitation will be sent via email

A3: Will select course plan

R3: This plan will be recorded and will be shown in dashboard

A4: Teacher will choose marking criteria and will allocate mark percentage

R4: This actions will be recorded and will be shown on dashboard

Student:

A1: Student will receive invitation from teacher via email with a OTP

R1: Student will response to the invitation and will join the course using that OTP.

A2:Student will receive invitation from teacher via email with a OTP

R2: Student will not response to the invitation

4.3.6 LEVEL-1.3

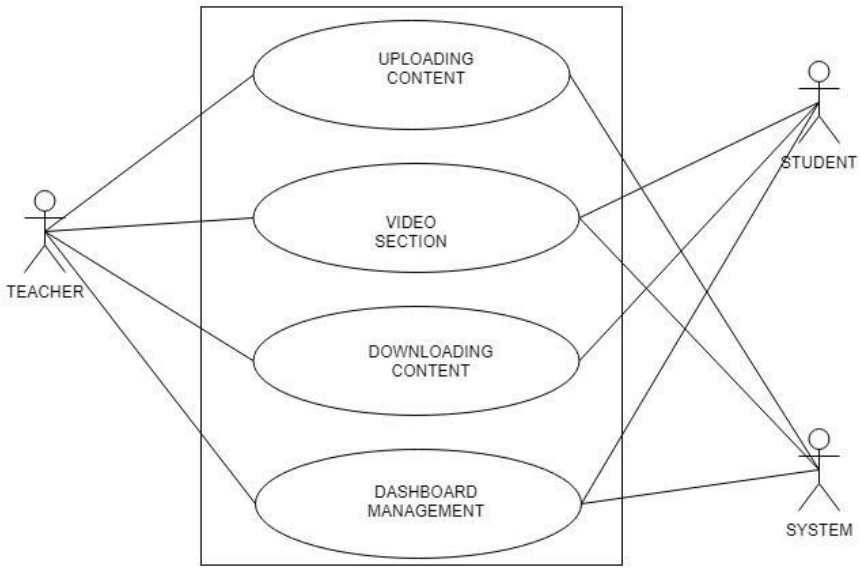


Figure 6: CMS_Content_Management

Description of Use case diagram level-1.3:

Anything related with content management will be handled by this subsystem. This subsystem is divided into four parts. These are as following:

1.Uploading content

2.Video Section

3.Downloading Content

4.Dashboard Management

Video Section and dashboard management will be further decomposed in level 1.3.2 and 1.3.4 respectively.

The teacher will upload class contents, study materials, books, video contents (Of the previous live classes and referenced tutorial), results and notices. Notices will be sent to all enrolled students via email for every upload. Students can upload assignments. The uploaded content will be stored in database.

Teacher can go live in classes and video will be uploaded by the software later with the permission of the teacher.

Students can download class contents, study materials, books, video contents (Of the previous live classes and referenced tutorial) and grade sheet. Teacher can download assignments.

Primary Actors: Student,Teacher,System,Database

Secondary Actors: E-mail

Action Reply

Teacher:

A1: Teacher wants to upload contents.

R1: System will allow to upload contents, the contents will be stored in database and will be shown in dashboard.

A2: Teacher wants to download something from dashboard.

R2: System will allow to download content from database.

Student:

A1: Student wants to upload assignment.

R1: Assignment will be uploaded.

A2: Student wants to download.

R2: contents will be downloaded from database.

4.3.7 LEVEL 1.3.2

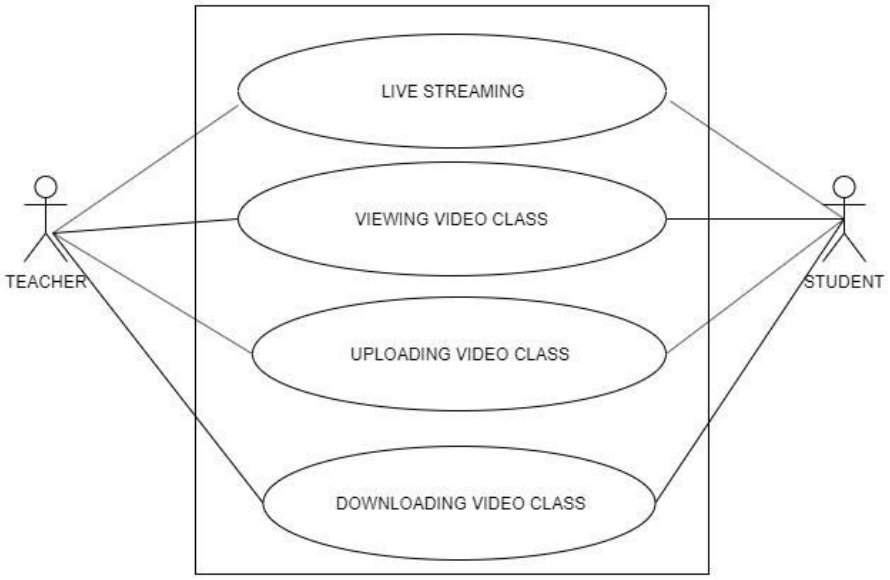


Figure 7: CMS_Video_Section

Description of Use case diagram level-1.3.2:

This subsection will describe every thing about video section. This subsection is divided into four parts. Such as:

1. Live Streaming

2.Video Class

3.Uploading Video Class and

4. Downloading video class

Teacher can go live in classes and video will be uploaded by the software later with the permission of the teacher.

Students can attend classes online through video streaming. In that case, the student will miss the attendance.

After the live class if teacher wants s/he can upload video class. Then everyone in this course can download video class.

Primary Actors: Student,Teacher,Database

Secondary Actors:

Action Reply

Teacher:

A1: Teacher wants to take live class

R1: System will allow him/her to take live class

A2: Teacher wants to upload video class

R2: Video content will be stored in database and will be shown on the dashboard.

A3: Teacher wants to download video class

R3: Video content will be downloaded from database.

Student:

A1: Students want to join live class.

R1: System will allow them to join live class.

A2: Students want to download video content.

R2: Video content will be downloaded from database.

4.3.8 LEVEL 1.3.4

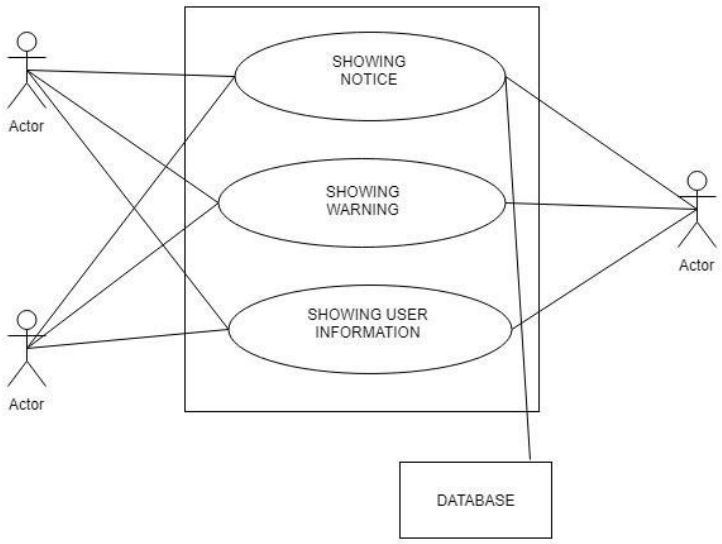


Figure 8: CMS_Dashboard_Management

Description of Use case diagram level-1.3.2:

This subsection will describe every thing about Dashboard . This subsection is divided into three parts. Such as:

1.Showing Notice

2.Showing Warning

3. Showing User Info

There will be different dashboards for teachers and students.

> **Student's Dashboard:** Students can see their previously uploaded assignments and the upcoming and pending assignments. The dashboard will show if there is any notice. Marks of particular fields will be uploaded in the dashboard. Course Contents will be shown in dashboard.
>
> **Teacher's Dashboard:** Teachers will see the upcoming events, the submitted assignments, a mark sheet of all the students where s/he can add marks of some fields.

Students below the attendance 60% cannot participate in exam. Students with critical attendance level will be able to see the warning .

Primary Actors: Student,Teacher,System

Secondary Actors:

4.3.9 LEVEL 1.4

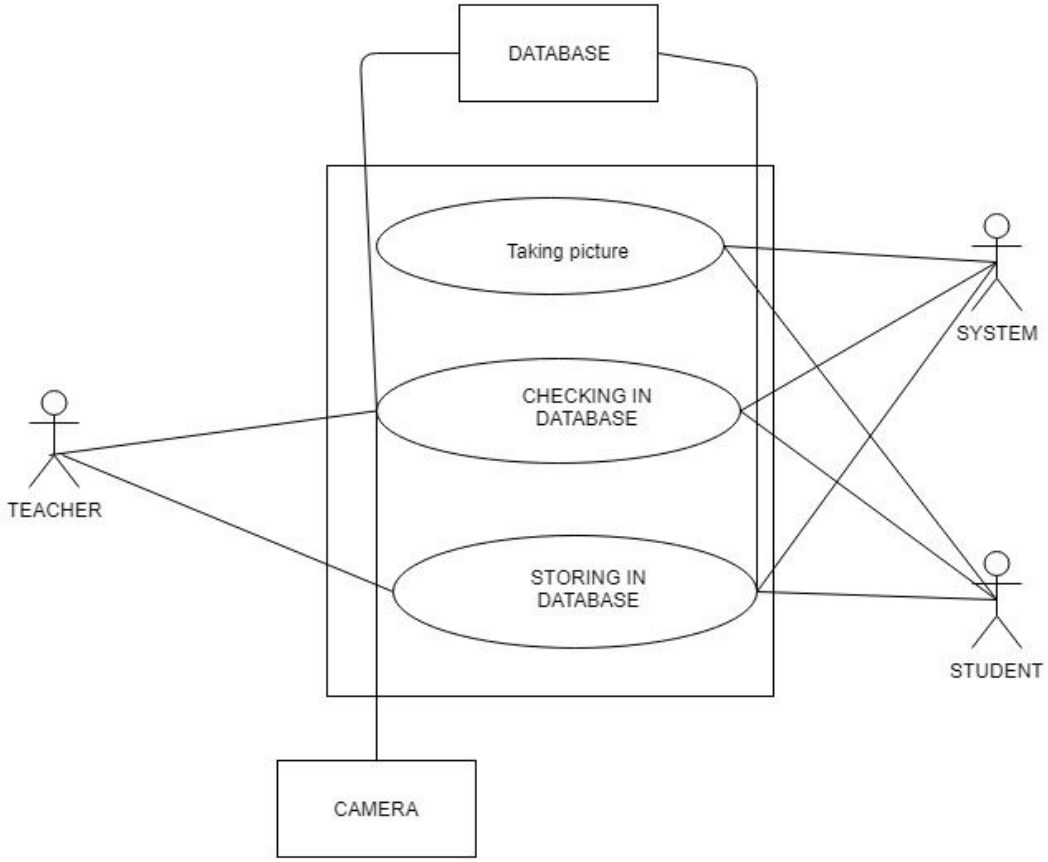


Figure 9: CMS_Attendance_Management

Description of Use case diagram level-1.4:

Anything related with attendance management will be handled by this subsystem. This subsystem is divided into three parts. These are as following:

1.Taking Picture

2.Checking in Database and

3.Storing in Database

Teacher will take a clear picture of the students or system will collect picture from CCTV footage . After that system will identify the present students through facial recognition. And the attendance will be stored in Database.

Primary Actors: Student,Teacher,Database,System

Secondary Actors: Camera

Action Reply

A1: System will collect picture from teacher or CCTV camera

R1: Attendance will be recorded in database through facial recognition.

4.3.10 LEVEL 1.5

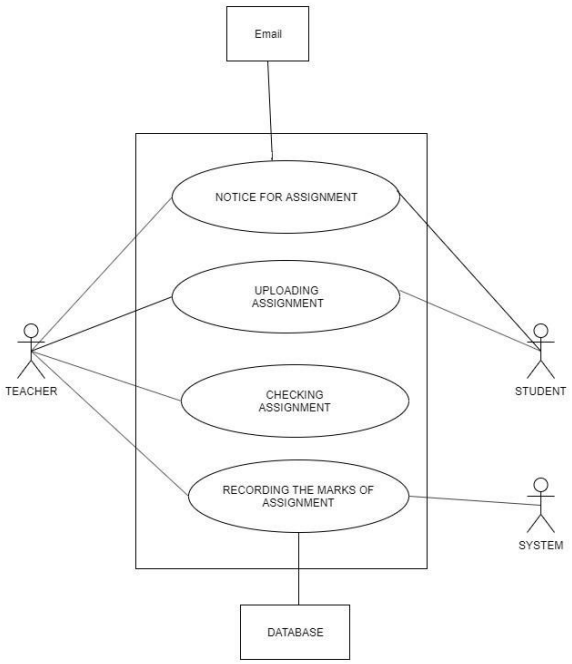


Figure 10: CMS_Assignment_Management

Description of Use case diagram level-1.5:

Anything related with assignment management will be handled by this subsystem. This subsystem is divided into four parts. These are as following:

1.Notice for Assignment

2.Uploading Assignment

3.Checking Assignment and

4.Recording marks for assignments

Teacher will declare the topic of the assignment and the last date for submitting assignments. Students will see the notice in their dashboard. Students will upload their assignments. Teacher will download their assignments and will check it manually. Then the marks of the assignment will be recorded in database.

Primary Actors: Student,Teacher,Database,System

Secondary Actors: Email

Action Reply

Teacher:

A1: Teacher will declare the date and topic of the assignment.

R1: System will notify the students through email and dashboard.

A2: Teacher wants to download the assignments.

R2: Assignments will be downloaded from database.

A3: Teacher will give marks for the assignments on given field

R3: Marks will be stored in the database.

Student:

A1: Student wants to upload assignment.

R1: System will allow him to upload assignment and the assignment will be stored in database.

4.3.11 LEVEL 1.6

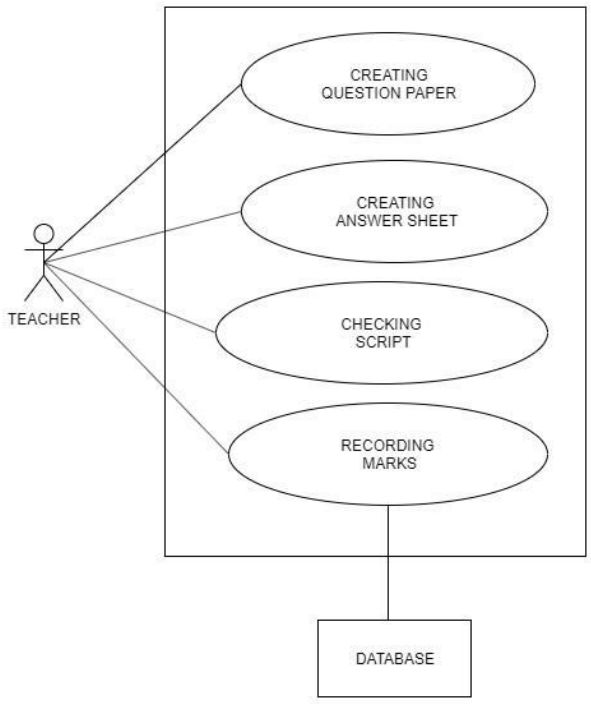


Figure 11: CMS_Exam_Section

Description of Use case diagram level-1.6:

Anything related with examination will be handled by this subsystem. This subsystem is divided into four parts. These are as following:

1.Creating Question Paper

2.Creating Answer Sheet

3. Checking Script and

4. Recording marks

Teacher will declare a date for the exam . S/he will create a question paper as well as an answer sheet with accurate answers. After checking the answer sheets teacher will upload the marks and the marks will be recorded in database.

Primary Actors: Teacher,Database

Secondary Actors:

Action Reply

A1: Teacher wants to create question paper and answer sheet

R1: System will allow to do so.

A2: Teacher wants to record marks of the final exam

R2: Marks will be recorded in database.

4.3.12 LEVEL 1.7

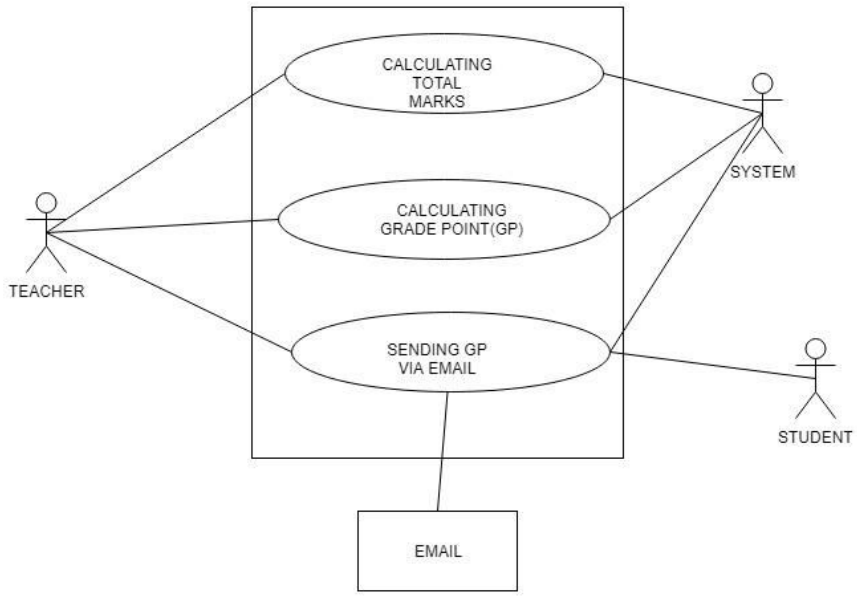


Figure 12: CMS_Result_Management

Description of Use case diagram level-1.7:

Anything related with result management will be handled by this subsystem. This subsystem is divided into three parts. These are as following:

1. Calculating Total Marks

2. Calculating Grade Point and

3. Sending GP via email

Teacher will enter the marks of a student. System will calculate total marks as well as will calculate grade point. After calculating grade point this grade point will be sent to the students via email.

Primary Actors: Student,Teacher,System

Secondary Actors: Email

Action Reply

A1: Teacher will enter marks of the exam.

R1: System will calculate total marks as well as grade point . Then the system will send this grade point to the students via email.

4.3.13 LEVEL 1.8

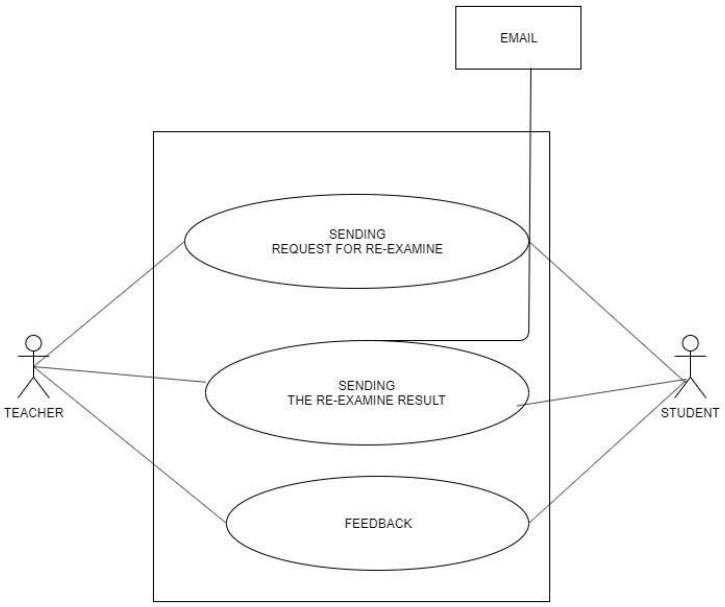


Figure 13: CMS_Re-examine_Feedback

Description of Use case diagram level-1.8:

Anything related with feedback and re-examine will be handled by this subsystem. This subsystem is divided into three parts. These are as following:

1. Sending Request for re-examine

2. Sending the re-examine result

3. Feedback

A student can request for re-examining his result to the course teacher. Then the teacher re-check his result and send the re-examined result . The student can give his feedback on the re-examined result. Student can also give anonymous feedback at the end of the course.

Primary Actors: Student,Teacher,System

Secondary Actors: Email

Action Reply

Student:

A1: Request for re-examine the result

R1: Request will send to the course teacher.

A2: Students wants to give anonymous feedback.

R2: System sends anonymous feedback to the course teacher.

Teacher:

A1: Receives request for re-examine

R1: Re-checks the result

A2: Teacher wants to upload re-examined marks

R2: Marks uploaded and stored in database. And this marks will be sent to the student via email.

5. ACTIVITY DIAGRAM OF COURSE MANAGEMENT SYSTEM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

ID: CMS_SU_AD
Name: Sign Up

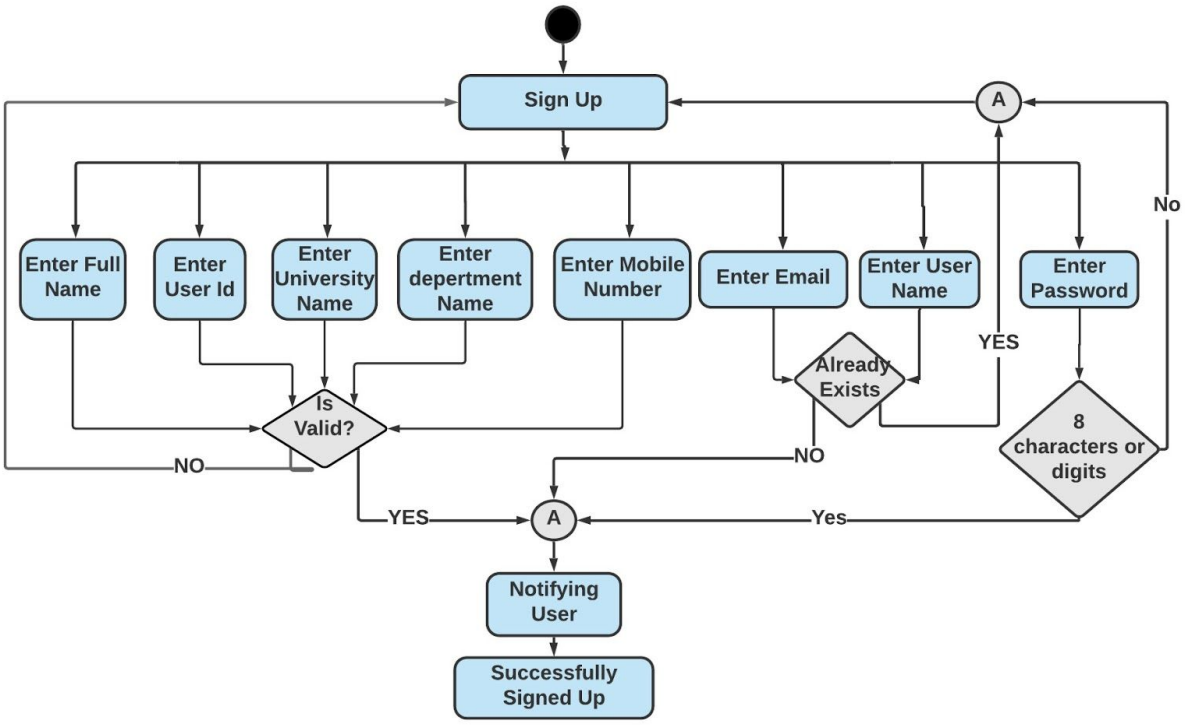


Fig: Sign_Up_AD

Description: From figure:Sign_Up_AD it is visible that, Full Name, User ID, University Name, Department Name, Email, Mobile Number, Username and password has to be entered to sign up. If the username already exists in the database, then the user will be asked to enter a new username. If any of the above information is invalid, s/he will be asked to re enter the information. After all the information is verified, the user will be notified. With this, the sign up will be complete.

ID: CMS_SI_AD
Name: Sign In

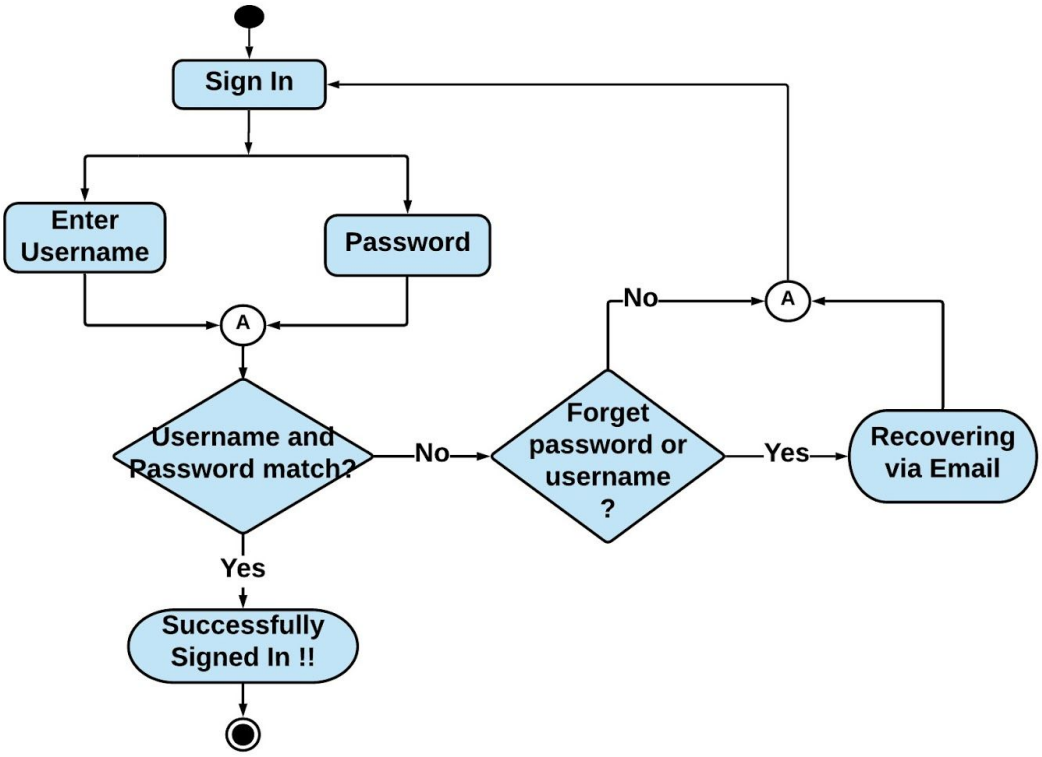


Fig: Sign_In_AD

Description: From Fig: Sign_In_AD it is visible that username and password has to be entered to sign in. If the username and password matches and is found in the database, then the user will successfully sign in. On the contrary, the user will be asked if s/he forgot the password. If the user confirms that s/he forgot the password, an email will be sent to him/her to recover his/her account. If the user has not forgotten his password, s/he will be able to re enter the username and password.

ID: CMS_whole_system_AD
Name: Whole System Briefing

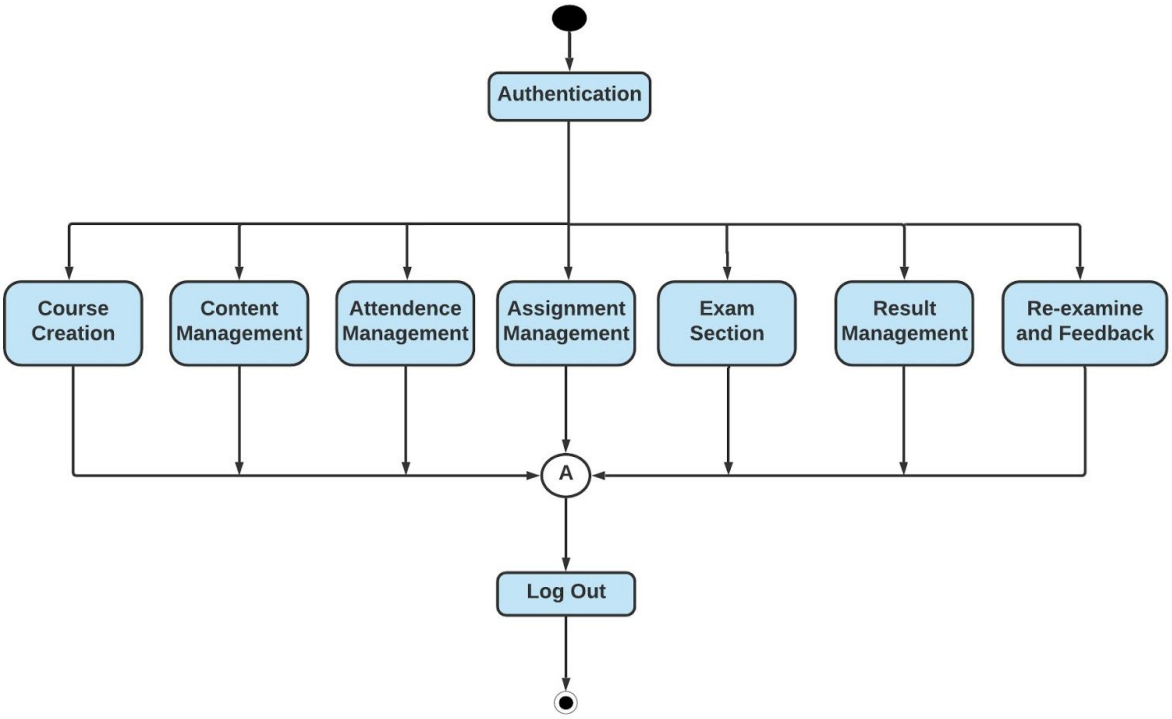


Fig: Whole_System

Description: From fig: Whole_system it is visible that the following activities can be done after authentication.

1. Course Creation
2. Content Management
3. Attendance Management
4. Assignment Management
5. Exam Selection
6. Result Management
7. Re-examine and Feedback

One can log out after the aforementioned activities.

ID: CMS_CC_AD

Name: Course Creation

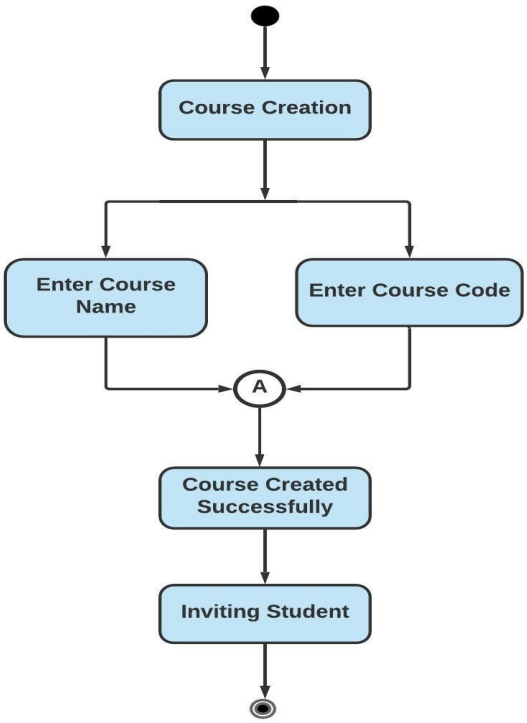


Fig: Course_Creation_AD

Description: From Figure:Course Creation, it is visible that Course Name and Course Code has to be entered to create a course. After the course creation is successful, the desired students will be invited via email by the teacher with the help of the system.

ID: CMS_CP_AD
Name: Course Planning

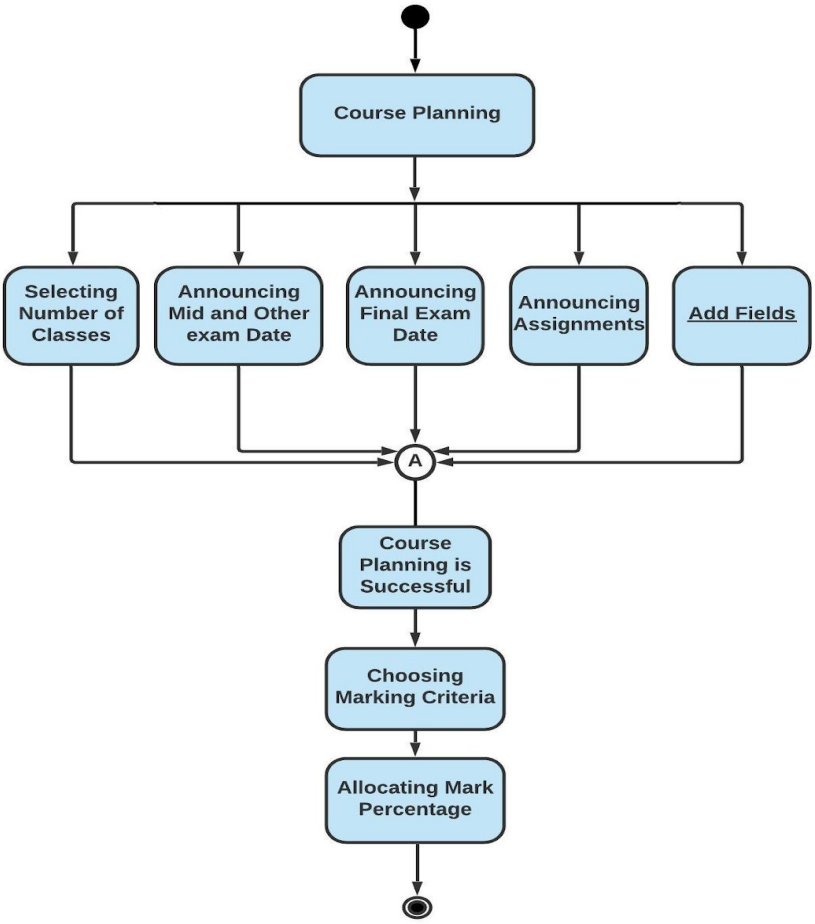


Fig: Course_Planning_AD

Description: From Figure: Course_Planning_AD, it is visible that the following activities can be done in Course Planning.

a. Selecting Number of Classes
b. Announcing Mid and Other Exam date
c. Announcing Final Exam date
d. Announcing Assignments
e. Announcing Lab Works

One can log out after the aforementioned activities.

ID: CMS_CM_AD
Name: Content Management

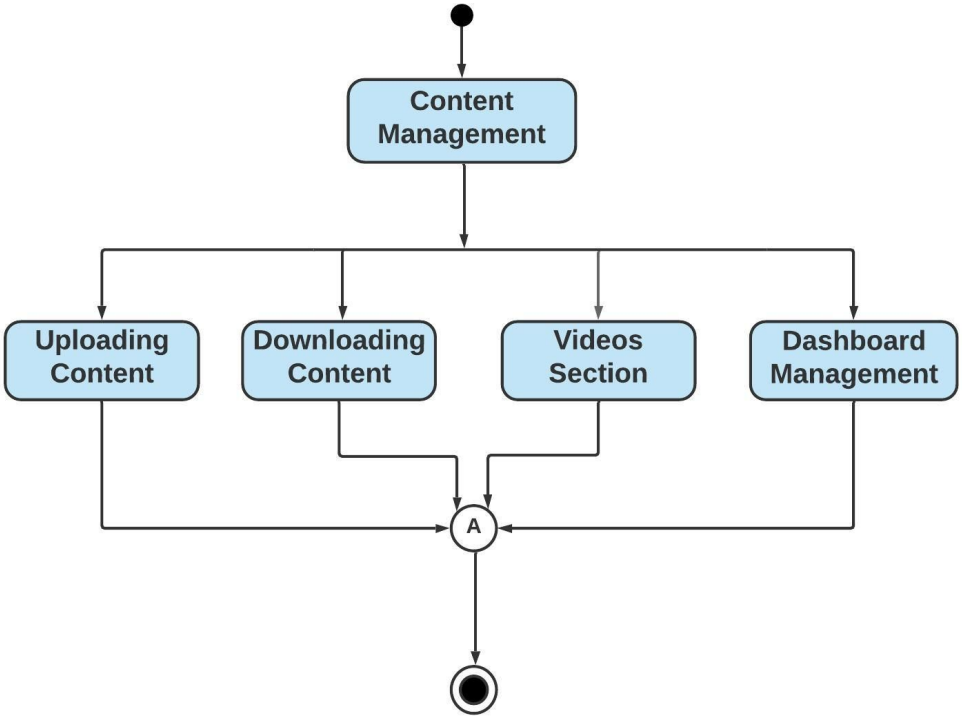


Fig: Content_Management

Description: From Figure: Content_Management it is visible that contents can be uploaded and downloaded. Besides, Videos Section and Dashboard Management will be under this sub system.

ID: CMS_VS_AD
NAME: Video Section

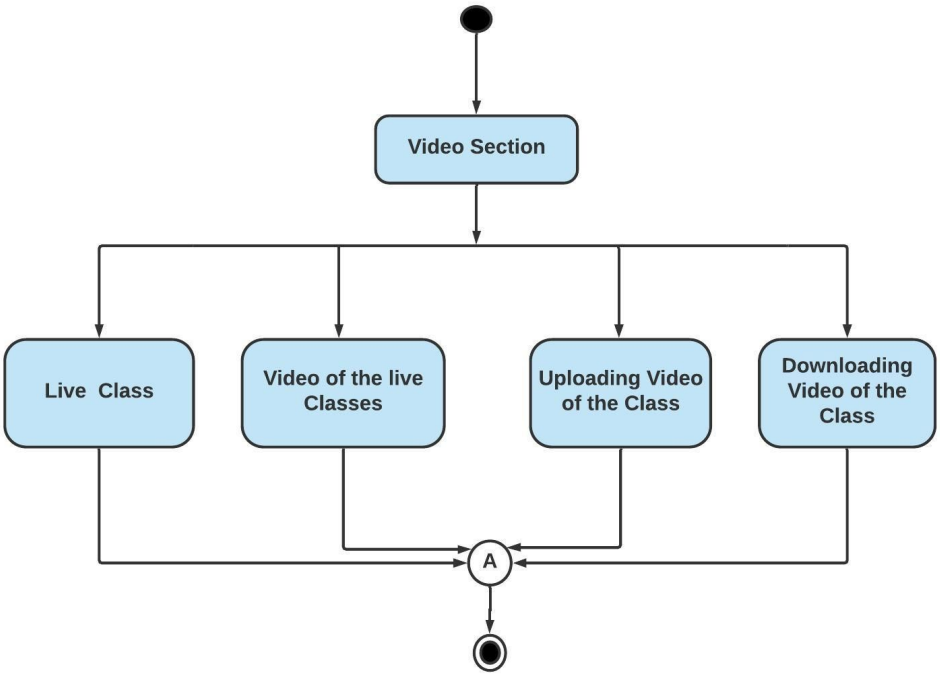


Fig: Video_Section

Description: In the video section, classes can be streamed live. The live class can be attended. The video of the class can be uploaded in the system. The video of the class can be downloaded.

ID: CMS_DM_AD
Name: Dashboard Management

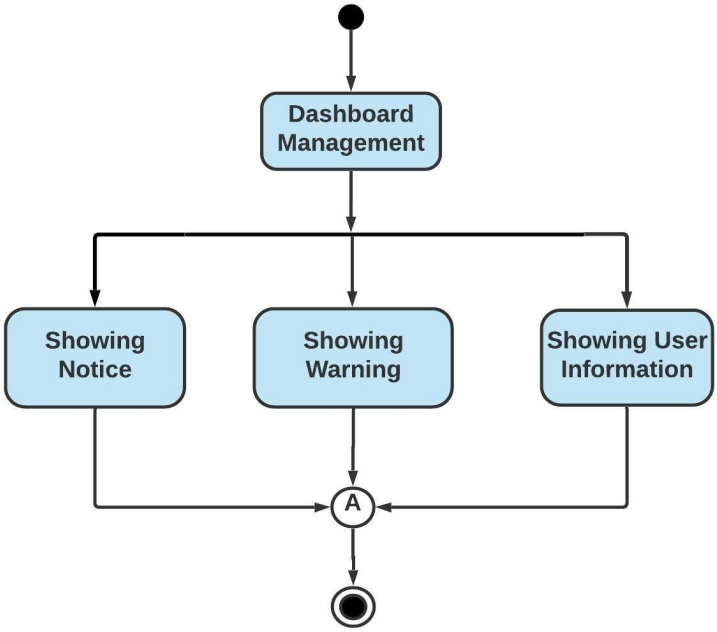


Fig: Dashboard_Management

Description: From figure: Dashboard_Management it is visible that notices, warning and User information will be shown in the dashboard.

ID: CMS_AtM_AD
Name: Attendance Management

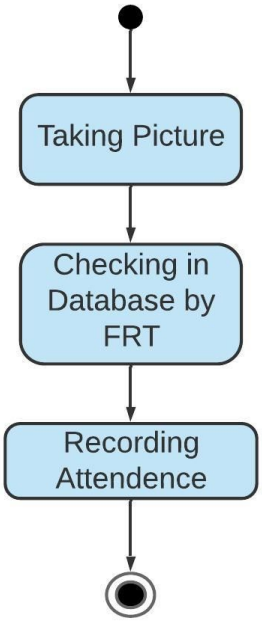


Fig: Attendance_Management

Description: From Figure: Attendance_Management it is visible that Picture of the classroom will be taken in the attendance management system. With Facial Recognition Technology(FRT), the photo will be matched in the database and the attendance of the corresponding individual will be recorded.

ID: CMS_AsM_AD
Name: Assignment Management

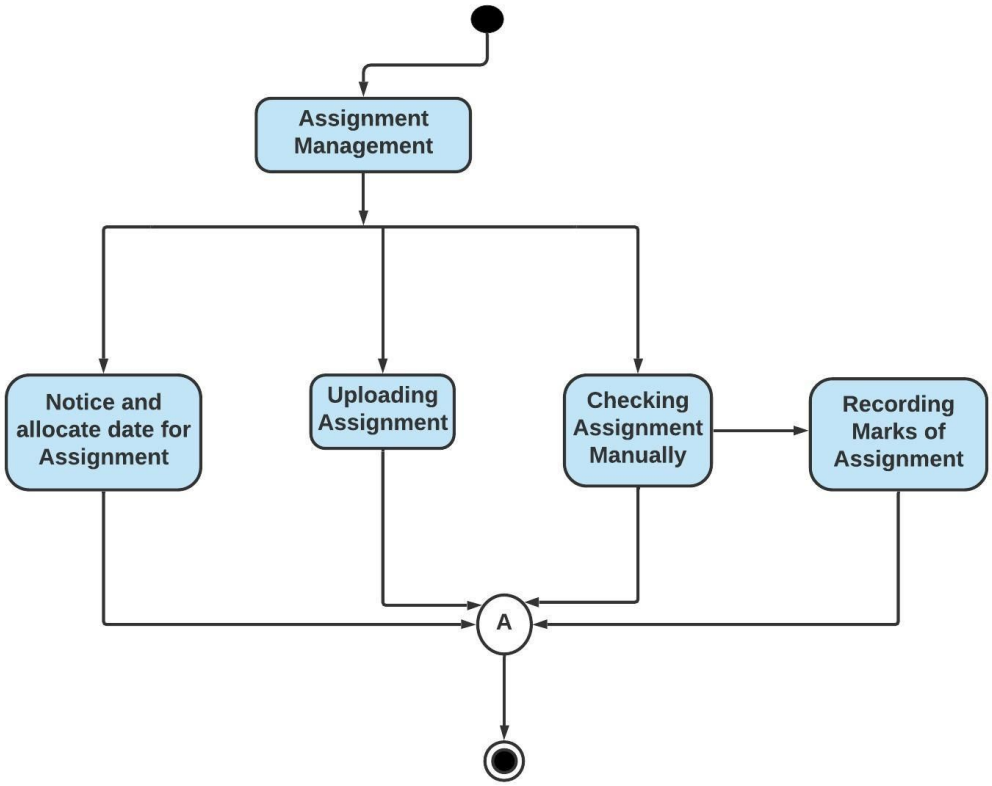


Fig: Assignment_Management

Description: From Figure: Assignment_Management it is visible that notice and date will be provided for assignments. The assignments can be uploaded in the system. The assignments can be checked from the system manually. The marks of the assignment can be recorded in the system after checking the assignments.

ID: CMS_ES_AD
Name: Exam Section

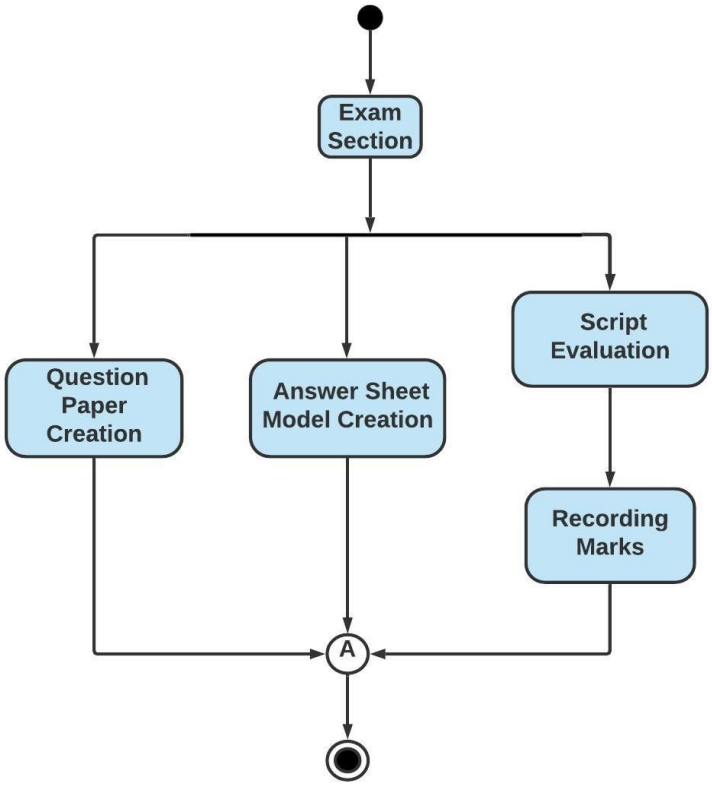


Fig: Exam_Section

Description: From the figure: Exam_Section it is visible that question paper can be created based on readymade templates, answer sheet can be created based on readymade templates, script checking will be done manually but marks will be entered on answer sheet and marks will be calculated from the answer sheet and recorded automatically.

ID: CMS_MC_AD
Name: Marks Compilation

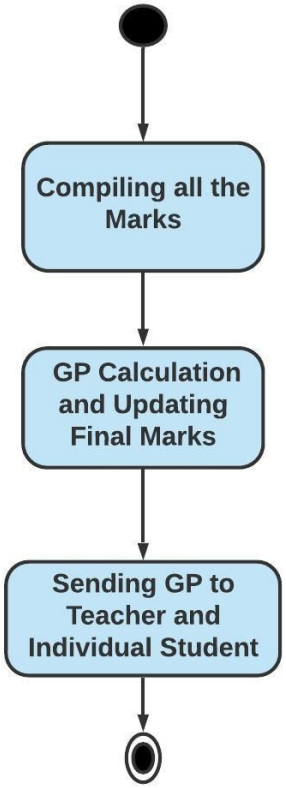


Fig: Total_Marks_Calculation

Description: From Figure: Total_Marks_Calculation it is visible that all the marks will be compiled here and Grade point will be calculated. The final GP will be sent to Teacher and individual Student.

ID: CMS_REF_AD

Name: Re-Examine and Feedback

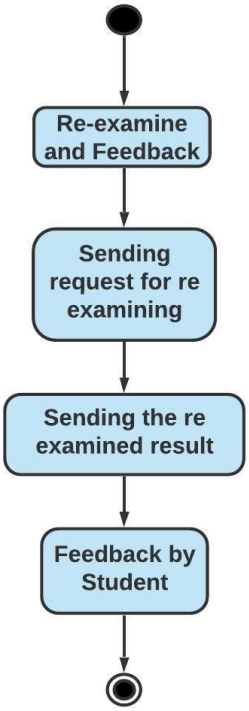


Fig: Re-Examining Result

Description: From Figure: Re-Examining Result it is visible that A request for re-examining can be sent. Re-examining will be done after the request and the new results will be sent to the individual with appropriate comments. Feedback can be given against the re-examined result.

6. SWIMLANE DIAGRAM OF COURSE MANAGEMENT SYSTEM

A **swimlane** (or **swimlane diagram**) is used in process flow **diagrams**, or flowcharts, that visually distinguishes job sharing and responsibilities for sub-processes of a business process.

Note: The interface role defines a common role which is played by User.

ID: CMS_SU_SL
Name: Sign Up

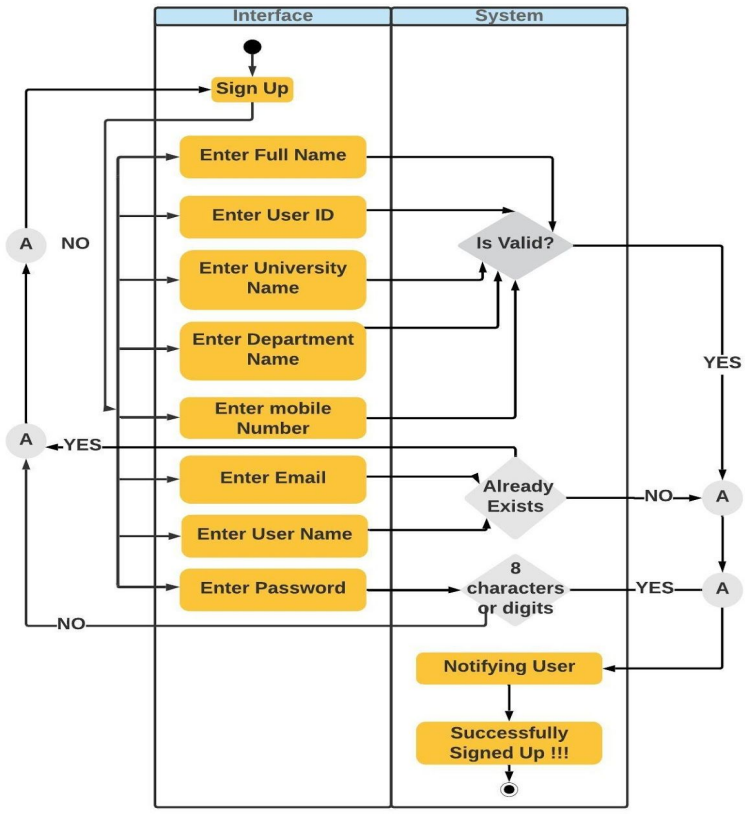


Fig: Sign Up

Responsibility:

From figure: Sign Up, the responsibilities of Interface and System in Sign up can be identified which are mentioned below.

Interface:

- Enter Full Name, User ID, University Name, Department Name, Mobile Number, Email, Username, Password.

System:

- Verifying the entered information.
- Notifying the user after successful sign up.

ID: CMS_SI_SL

Name: Sign In

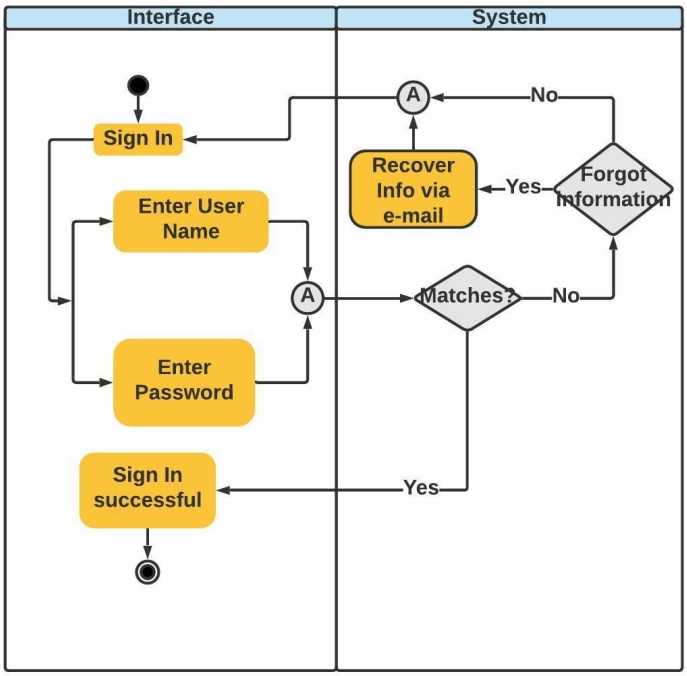


Fig: Sign In

Responsibility:

From figure:Sign In, the responsibilities of Interface and System in Sign up can be identified which are mentioned below.

Interface:

- A user can sign in entering Username and Password.

System:

- Account recovery through email.
- Check whether Username and password matches.
- Check if anyone forgot his/her information.

ID: CMS_CC_SL

Name: Course Creation

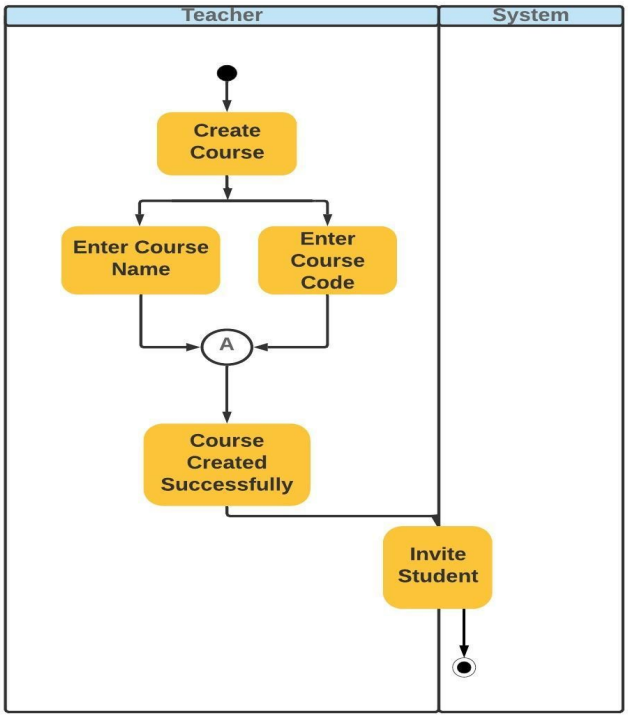


Fig: Course Creation

Responsibility:

From figure: Course Creation, the responsibilities of Teacher and System in creating a course can be identified which are mentioned below.

Teacher:

- Create Course entering Course Name and Course Code

System:

- Invite Student through Email.

ID: CMS_CP_SL

Name: Course Planning

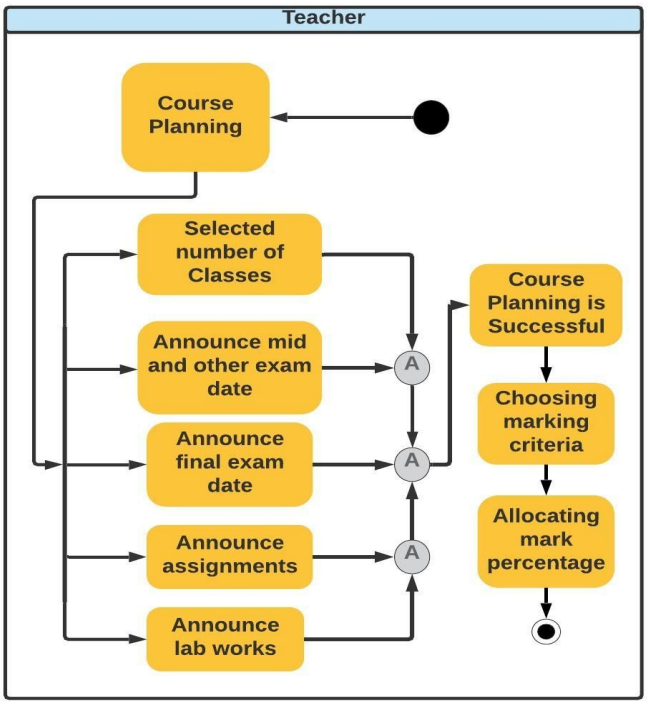


Fig: Course Planning

Responsibility:

From figure: Course Planning, the responsibilities of Teacher in planning a course can be identified which are mentioned below.

Teacher:

- Make Course Plan
- Select Number of Classes
- Announce Mid and other exams
- Announce Final Exam
- Announce Assignments
- Announce Lab works
- Choose Mark Criteria
- Allocate Mark Percentage to selected criteria

ID: CMS_CM_SL
Name: Content Management

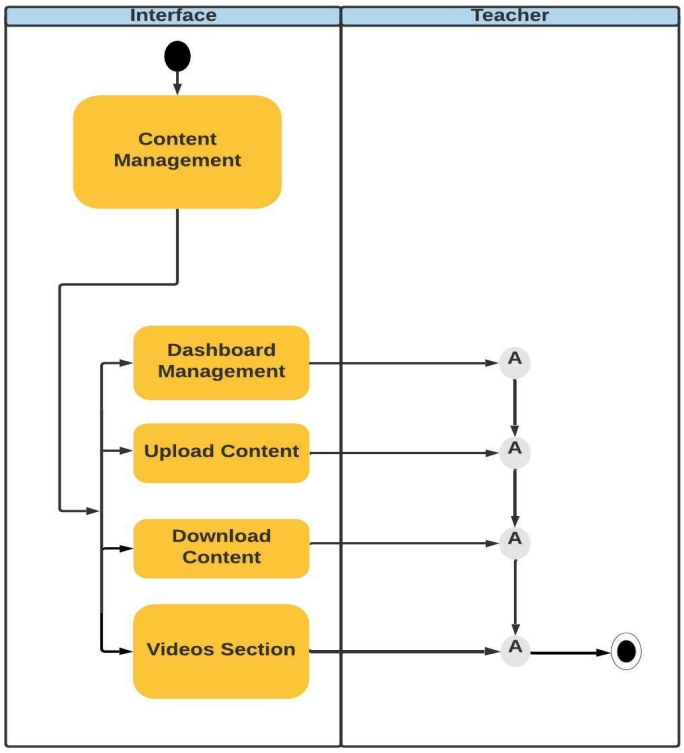


Fig: Content Management

Responsibility:
From figure: Content Management, the responsibilities of Teacher and Interface in managing contents can be identified which are mentioned below.
Interface:

- Uploading any Content
- Downloading any Content
- Video Section
- Dashboard Management

ID: CMS_VS_SL
NAME: Video Section

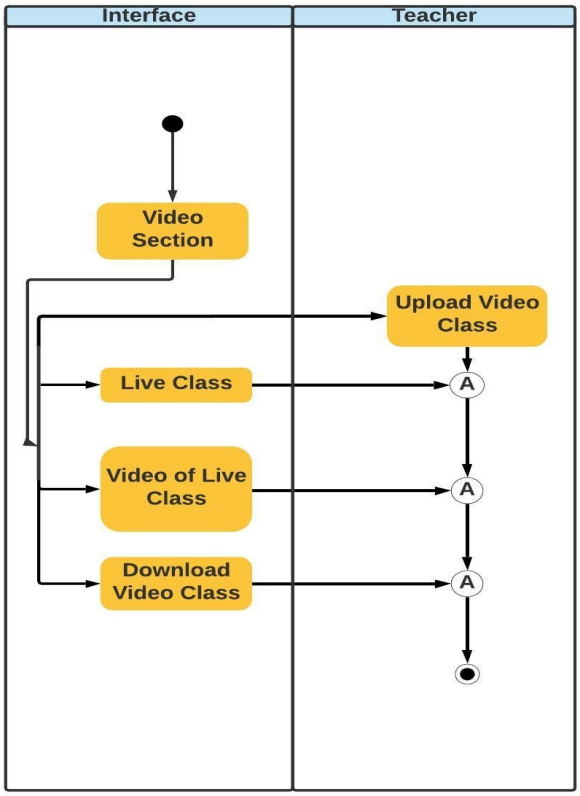


Fig: Video Section

Responsibility:

From figure: Content Management, the responsibilities of Teacher and Interface in managing contents can be identified which are mentioned below.

Interface:

- Attend live class
- Watch videos of the classes
- Download videos of the classes

Teacher:

- Upload video of classes

ID: CMS_DM_SL
Name: Dashboard Management

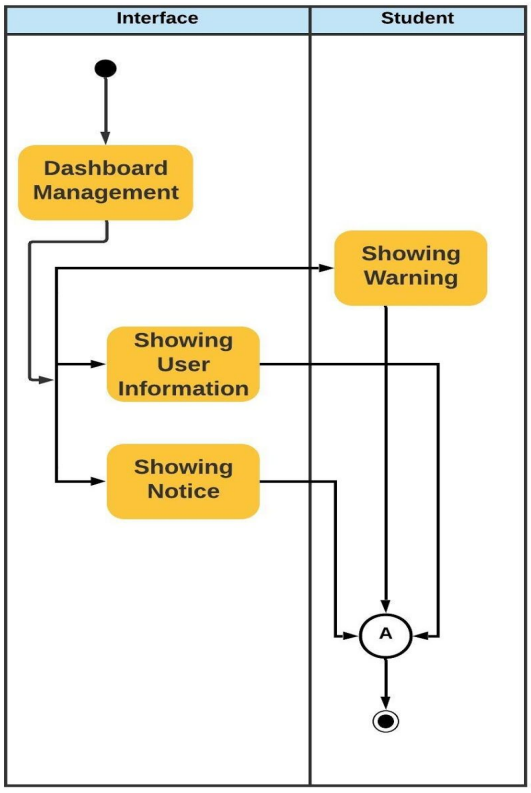


Fig: Dashboard Management

Responsibility:

From figure: Dashboard Management, the responsibilities of Interface and Student in managing contents can be identified which are mentioned below.

Interface:

- Will have a dashboard
- Dashboard will show user information
- Dashboard will show notice

Student:

- Warnings will be shown in dashboard

ID: CMS_AtM_SL
Name: Attendance Management

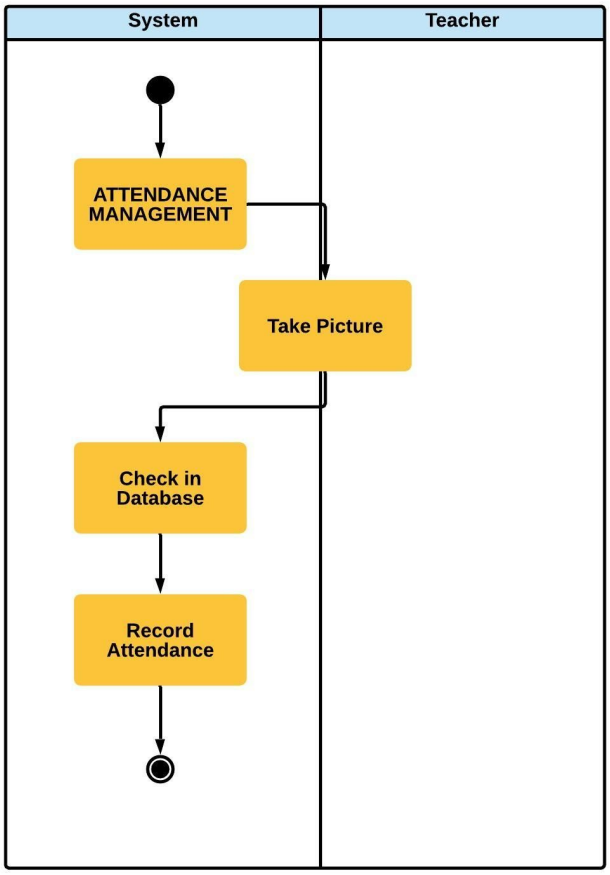


Fig: Attendance Management

Responsibility:

From figure: Attendance Management, the responsibilities of System and Teacher in managing contents can be identified which are mentioned below.

System:

- May take a picture of the class
- Check in database with Facial Recognition Technology (FRT)
- Record Attendance

Teacher:

- May take a picture of the class

ID: CMS_AsM_SL
Name: Assignment Management

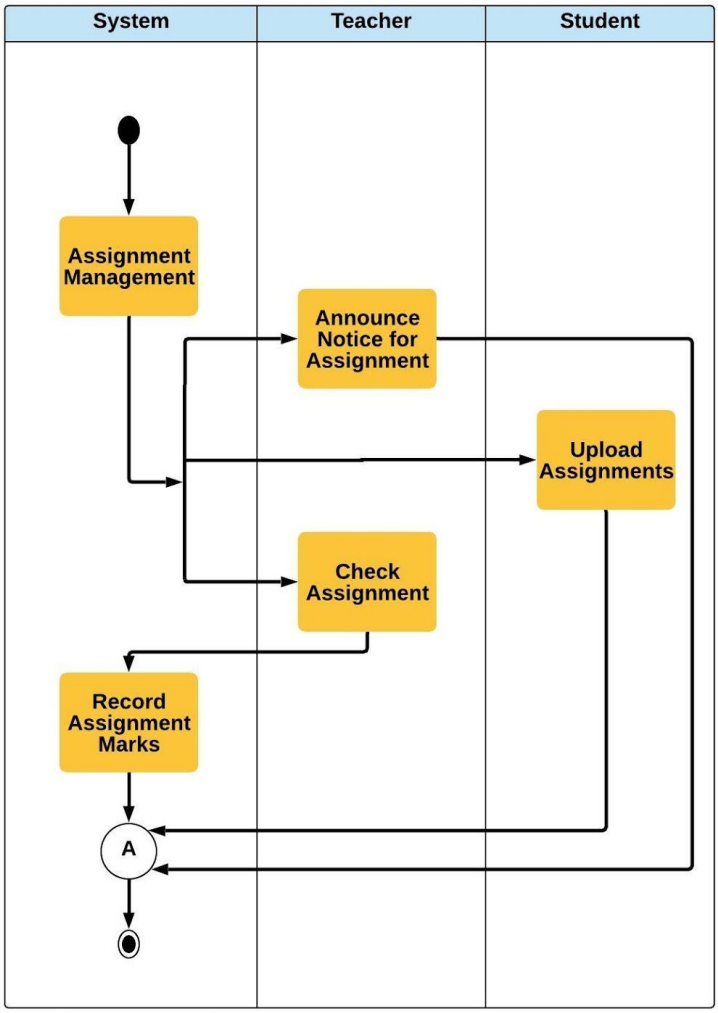


Fig: Assignment Management

Responsibility:

From figure: Assignment Management, the responsibilities of system, teacher and student in managing assignments can be identified which are mentioned below.

System:

- Record marks of assignments

Teacher:

- Announce assignments
- Check assignments

Student:

- Upload assignments

ID: CMS_ES_SL
Name: Exam Section

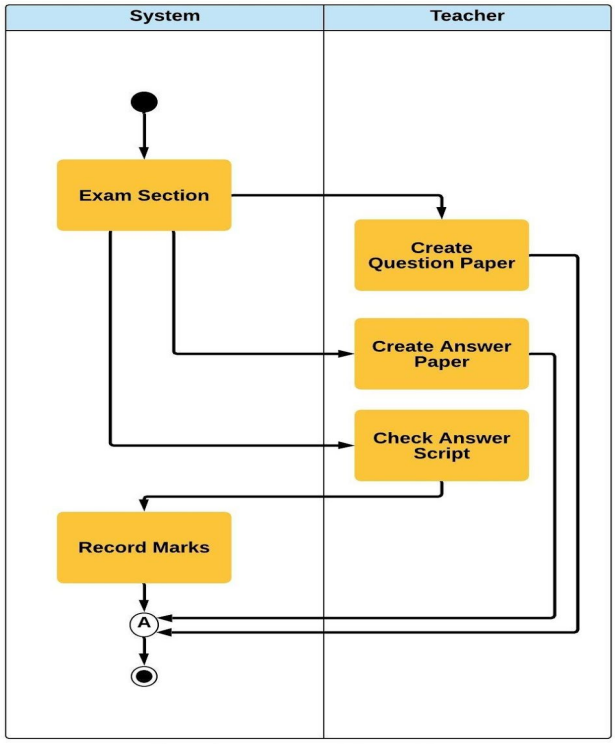


Fig: Exam Section

Responsibility:

From figure: Exam Section, the responsibilities of system and teacher in managing exams can be identified which are mentioned below.

Teacher:

- Create question paper based on readymade templates
- Create answer paper based on readymade templates
- Check answer script and enter marks on answer sheet

System:

- Record marks by calculating from the answer sheet

ID: CMS_MC_SL
Name: Marks Compilation

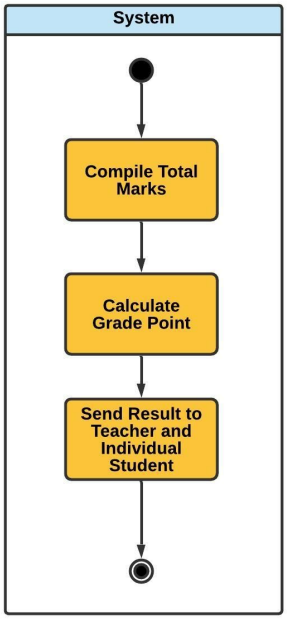


Fig: Total Marks Calculation

Responsibility:
From figure: Total Marks Calculation, the responsibilities of system and teacher in marks calculation can be identified which are mentioned below.
System:

- Compile all the marks
- Calculate grade point
- Send result to teacher and individual student

ID: CMS_REF_SL
Name: Re-Examine and Feedback

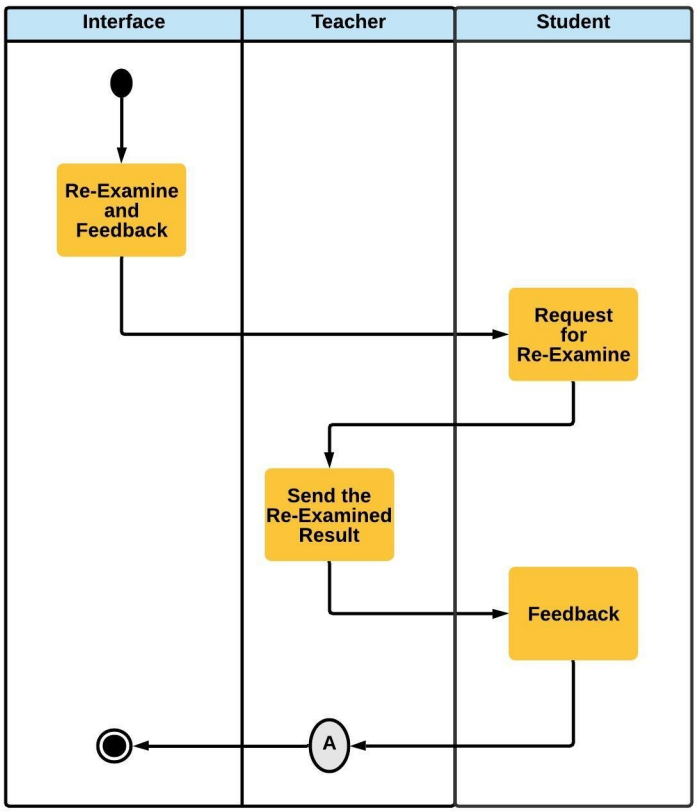


Fig: Re-Examination

Responsibility:

Teacher:

- Re-Examine and send the re-examined result with appropriate comments

Student:

- Request for re-examining
- Send Feedback

5. DATA MODELLING OF COURSE MANAGEMENT SYSTEM

5.1 Data Modelling Concept

If software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated, then the software team chooses to create data models as part of overall requirements modeling. The entity-relationship diagram (ERD) identifies all data objects that are processed within the system, the relationships
between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

5.1.1 Data Objects

A data object is a representation of composite information that must be understood by the software. Here, composite information means information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

5.1.1.1 NOUN IDENTIFICATION

We identified all the nouns whether they are in problem space or in solution space from our story:

Serial no	Noun	Problem/solution space	Attributes
1	CMS	p	

2	Teacher	s	14, 19, 25, 26, 28, 29,30, 14, 32, 34
3	Course content	s	
4	Marks distribution	p	
5	Tabulation	p	
6	Student	s	14, 19, 25, 26, 28, 29,30, 14, 32, 34, 67
7	Assignment	s	30, 55
8	System	p	
9	Result	s	40, 41, 43, 27, 56, 60
10	Feature	p	
11	Authorization	p	
12	Authentication	s	
13	User	s	14, 19, 25, 26, 27, 28, 29, 14, 32
14	User-name	s	
15	Course code	s	
16	Information	s	
17	Account	s	18, 19, 29, 30, 28, 14, 31
18	Name	s	
19	User-id	s	

20	Level	p	
21	Course name	s	
22	Course plan	s	43, 54, 21, 15, 46, 3
23	Lab works	s	

24	Notification	s	47, 32, 12, 19
25	University	p	
26	Department	p	
27	Institutional mail	s	
28	Phone number	s	
29	Photo	s	
30	Assignment no	s	
31	Password	s	
32	Confirmation	s	
33	Profile	p	
34	Designation	s	
35	Dashboard	s	16
36	Notice	s	60-65
37	Attendance	s	19, 67, 29
38	Face detection	p	
39	Virtual exam	p	

40	Marksheet	s	
41	Evaluation of grade	s	
42	Exam	s	40, 41, 39, 49, 23, 7, 19, 61
43	Criteria	s	
44	Procedure	p	
45	Accurate Answer	p	
46	Template	s	
47	Warning System	s	

48	Dashboard	p	
49	Video Streaming	p	
50	Application	p	
51	Material	p	
52	Book	p	
53	Content	p	
54	Live Class	s	
55	No of Assignments	s	
56	GP	s	
57	Feedback	p	
58	Manual checking	p	
59	Script	p	

60	Re-examine result	s	
61	Re-examine	s	
62	Particular Fields	p	
63	ID	p	
64	Final Exam Date	s	
65	Midterm date	s	
66	Own choice	p	
67	Critical attendance level	s	
68	Warning	p	
69	Video Section	s	70, 71
70	Video Class	s	
71	Marks Allocation	s	

5.1.1.2 Potential Data objects:

No	Data Object	Attribute
1	Teacher	14, 19, 25, 26, 28, 29,30, 14, 32, 34
2	Student	14, 19, 25, 26, 28, 29,30, 14, 32, 34, 67
3	Video Section	70, 54
4	Exam	39-41, 23, 56, 61, 19, 61
5	Attendance	19, 67, 29
6	Notice	60-65
7	Notification	47, 32, 12, 19
8	Course plan	2, 6, 43, 54, 21, 15, ,46, 3
9	User	14, 19, 27, 28, 29, 32
10	Result	40, 41, 43, 27, 56, 60
11	Assignment	30, 55
12	Account	18, 19, 29, 30, 28, 14, 31

5.1.1.3 Analysis for finalizing Data Objects

- ❏ Both teacher and student have some common attributes. So their common attributes can be stored as **User.**

- ❏ In order to have access to the system, the user has to identify himself as a valid user. Then the user will be verified and will receive a **notification.**

- Teacher and student have some individual responsibilities. So those attributes are stored separately in **Teacher** and **Student**.

- The **Attendance** of the students will be stored and evaluated later.

- The teachers will use the system and gives assignments to the students. The students then submit the **Assignments** within due dates.

- For the whole course, the teacher plans a timeline. The timeline contains the activities of the teacher for the whole semester. The thing in whole is called **Course Plan**.

- During the course, the teacher will take several exams of his own choice. These information are stored in **Exam**.

- After the semester, the student will get their **Result**. The result will contain their grade point of the course.

- The teachers can upload reference videos according to the course contents to the video section. The students can download those videos from the **Video Section**.

5.1.1.4 Final Data Objects

In the following table we finalize the data objects with their attributes Most of the attributes of the data objects are selected from the usage scenario and some of the attributes are selected to complete to the system which are not in the usage scenario but important for the data objects .

Table 2: Final data object

1	User-Account	User ID, user name, Institutional mail, phone number
2	Student	User ID, user name, Institutional mail, phone number
3	Teacher	User ID, user name, Institutional mail, phone number
4	Assignment	Assignment no, User id ,Course code
5	Attendance	User ID, critical attendance level , course code
6	Notification	User ID, warning message , notice
7	Exam	User ID, Course code, Script number, Answer sheet number
8	Course Plan	Course code, User ID, Number of class, Exam date, Number of lab work , Number of assignment
9	Result	Course code, User ID, GP, Institutional mail, Re-examine gp

10	Video Class	Name, user id , course code
11	Mark sheet	Gp , user id , course code , Assignment marks , Final exam marks, Lab work Marks, Teacher's choice Marks
12	Template	Course code, User ID, Number of fields
13	Course content	Course code, User ID

5.2 DATA OBJECT RELATIONSHIPS

Data objects are connected to one another in different ways.

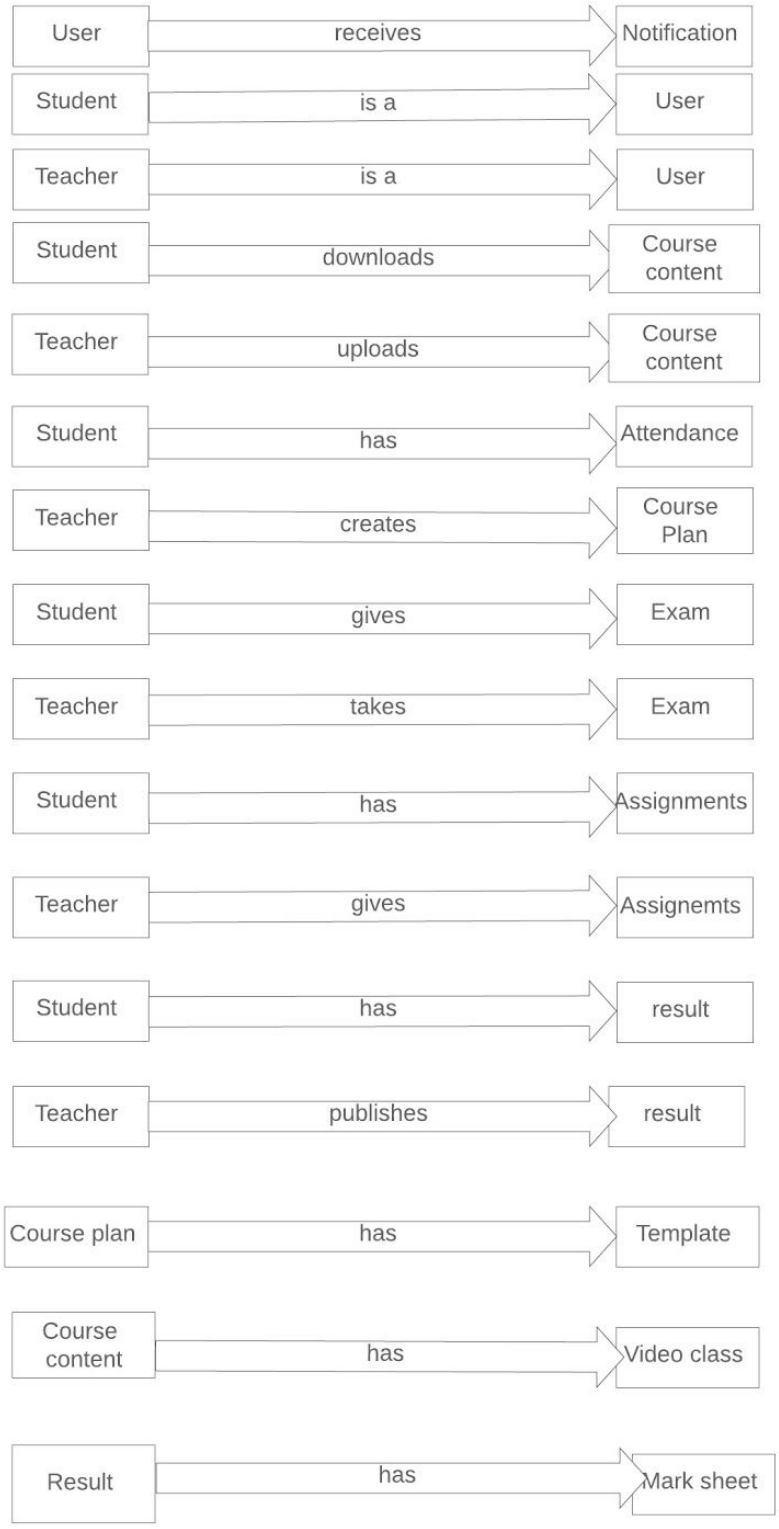

User
receives
Notification
Student
is a
User
Teacher
is a
User
Student
downloads
Course content
Teacher
uploads
Course content
Student
has
Attendance
Teacher
creates
Course Plan
Student
gives
Exam
Teacher
takes
Exam
Student
has
Assignments
Teacher
gives
Assignemts
Student
has
result
Teacher
publishes
result
Course plan
has
Template
Course content
has
Video class
Result
has
Mark sheet

Fig: Data Object Relationship Diagram

5.3 ENTITY RELATIONSHIP DIAGRAM

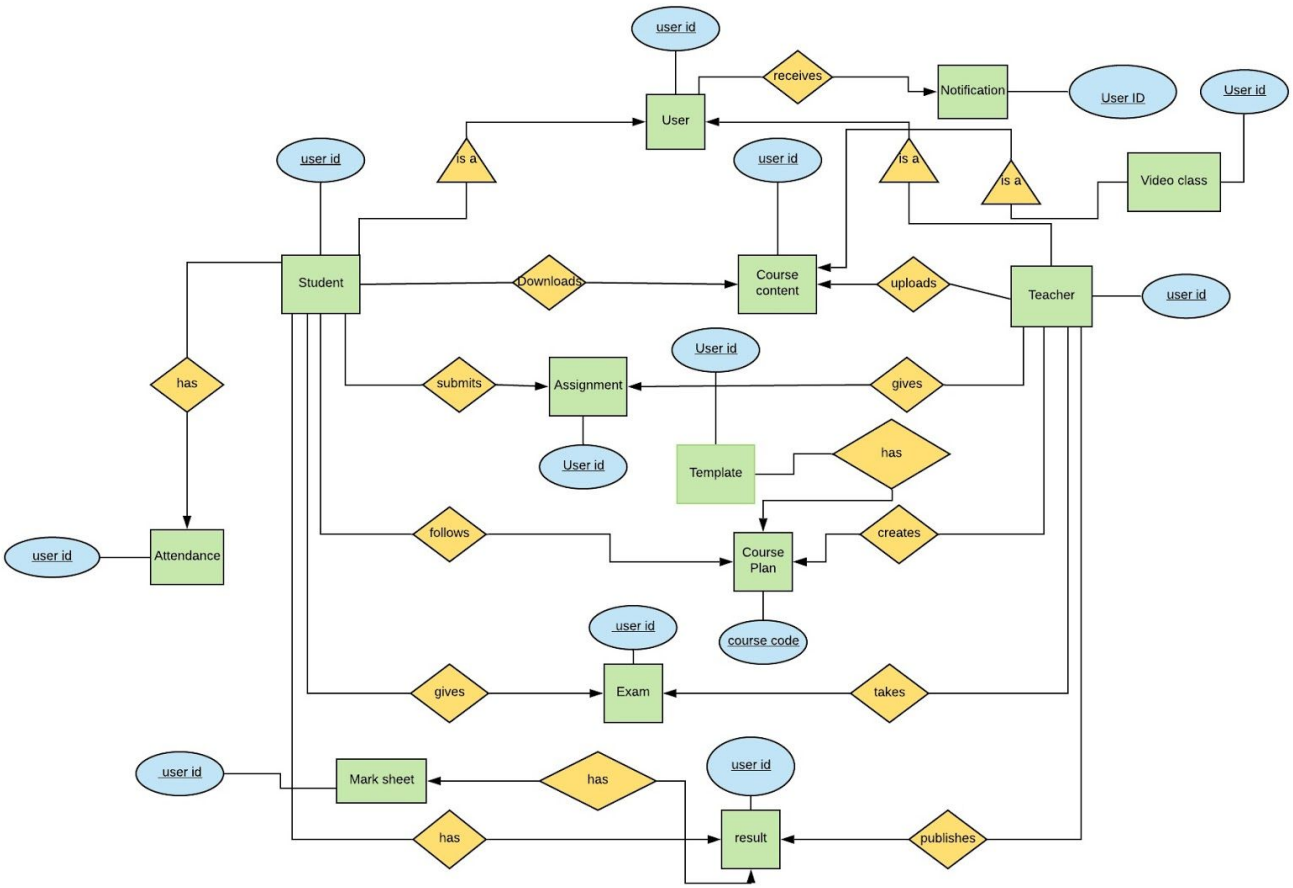


Fig: Entity Relationship Diagram

5.4 SCHEMA DIAGRAM

Table 3: Schema for user

USER

Attributes	Type	Size
User ID	Varchar	50
User name	Varchar	50
Institutional mail	Varchar	50
Phone number	Varchar	20

Table 4: Schema for student

STUDENT

Attributes	Type	Size
User ID	Varchar	50
User name	Varchar	50
Institutional mail	Varchar	50

Phone number	Varchar	20

Table 5: Schema for teacher

TEACHER

Attributes	Type	Size
User ID	Varchar	50
User name	Varchar	50
Institutional mail	Varchar	50
Phone number	Varchar	20

Table 5: Schema for assignment

ASSIGNMENT

Attributes	Type	Size
Assignment no	Varchar	50
User id	Varchar	50
Course code	Varchar	50

Table 6: Schema for attendance

Attendance

Attributes	Type	Size
User ID	Varchar	50
Critical Attendance level	Number	20
Course Code	Varchar	50

Table 5: Schema for notification

Notification

Attributes	Type	Size
User ID	Varchar	50
Course code	Varchar	50
Notice	Varchar	50

Warning message	Varchar	50

Table 6: Schema for exam

Exam

Attributes	Type	Size
User ID	Varchar	50
Course code	Varchar	20
Script number	Varchar	30

Answer sheet number	Varchar	20

Table 7: Schema for course plan

COURSE PLAN

Attributes	Type	Size
User ID	Varchar	50
Course Code	Varchar	50
Number of class	Number	10
Exam date	Varchar	10
Number of lab work	number	10
Number of assignment	number	10

Table 5: Schema for result

RESULT

Attributes	Type	Size
User ID	Varchar	50
Course code	Varchar	50
GP	Number	10
Institutional mail	Varchar	50
Re examine gp	number	10

Table 5: Schema for video section

VIDEO CLASS

Attributes	Type	Size
Name	Varchar	50
User ID	Varchar	50
Course Code	Varchar	50

Table 6: Schema for mark sheet

MARK SHEET

Attributes	Type	Size
GP	Number	10
User ID	Varchar	50
Course Code	Varchar	20
Assignment Marks	Number	10
Final Exam Marks	Number	10
Lab work Marks	Number	10
Teacher’s choice Marks	Number	10

Table 5: Schema for course content

COURSE CONTENT

Attributes	Type	Size
Course Code	Varchar	50
User ID	Varchar	50

Table 5: Schema for template

TEMPLATE

Attributes	Type	Size
User ID	Varchar	50
Course Code	Varchar	10
Number of fields	Number	10

6. CLASS-BASED MODELING FOR CMS

This Chapter is intended to describe class based modeling of “Course Management System”.

6.1 CLASS BASED MODELING CONCEPT

Class-based modeling represents the objects that the system will manipulate, the operations that will be applied to the objects, relationships between the objects and the collaborations that occur between the classes that are defined.

6.2 Noun List for CMS

No	Noun	No	Noun
1	CMS	11	University
2	Teacher	12	Department
3	Course-content	13	institutional mail
4	Marks-distribution	14	Phone Number
5	Tabulation	15	Photo
6	Student	16	User-Name
7	Assignment	17	Password
8	System	18	Confirmation
9	Result	19	Profile
10	feature	20	Designation

21	Authorization	49	Dashboard
22	Authentication	50	Notice
23	User	51	attendance
24	User-Name	52	face detection
25	Course-Code	53	virtual class
26	Information	54	mark sheet
27	Account	55	criteria
28	Full-Name	56	evaluation of grade
29	User-Id	57	exam
30	level	58	criteria
31	Course-name	59	procedure
32	course plan	60	accurate answer
33	Lab-Works	61	template
34	invitation	62	Warning System
35	Dashboard	63	video streaming
36	application	64	id
37	material	65	final exam date
38	book	66	midterm date
39	content	67	own choice
40	assignment	68	critical attendance level
41	database		
42	CGPA		
43	feedback		
44	manual checking		
45	script		
46	re-examine result		
47	re-examine		
48	particular fields		

6.3 Verb List of CMS

NO	Verb	No	Verb
1	Allocate	30	Response
2	Create	31	Give
3	Provide	32	Start
4	Verify	33	Fail
5	Request	34	Finalize
6	Finalize	35	Click
7	Authorize	36	Login
8	Get	37	Forget
9	Recover	38	Recover
10	Teach	39	Register
11	Wants	40	Send
12	Create	41	Invite
13	Invite	42	Include
14	Available	43	Estimate
15	Set	44	Add
16	Includes	45	Do
17	Estimate	46	Reference
18	Select	47	Attend
19	Upload	48	Take
20	calculate	49	Stream
21	Understand	50	Submit
22	Reference	51	log in
23	Need	52	Re-check

24	Send	53	Warn
25	Like	54	Miss
26	Notify	55	Go
27	Check	56	
28	Request	57	
29	Re-enter	58	

6.4 GENERAL CLASSIFICATION

To identify the potential classes, we have first selected the nouns from the solution space of the story. These were then characterized in seven general classification. The seven general characteristics are as follows:

1. External entities

2. Things

3. Events

4. Roles

5. Organizational units

6. Places

7. Structures

Following are the specifications of the nouns according to the general Classifications:

Table:1 General Classification

No.	Noun	General Classification
1	CMS	
2	Teacher	4,5,7
3	Course content	2,7
4	Marks distribution	2
5	Tabulation	2
6	Student	4,5,7
7	Assignment	2
8	System	2,4,7
9	Result	2
10	Feature	
11	Authorization	3
12	Authentication	3
13	User	4,5,7
14	User Name	
15	Course code	

16	Information	2
17	Account	2,4,7
18	Full Name	
19	User ID	
20	Level	
21	Course Name	
22	Course Plan	2,3,7
23	Lab works	2
24	Invitation	3
25	University	6
26	Department	5
27	E-mail	1
28	Phone Number	2
29	Photo	
30	User Name	
31	Password	
32	Confirmation	3
33	Profile	2
34	Designation	
35	Notice	2

36	Attendance	
37	Virtual class	3,5
38	Mark Sheet	
39	Criteria	
40	Evaluation of Grade	3
41	Examination	3
42	Procedure	3
43	Accurate Answer	2
44	Template	2
45	Warning System	3
46	Video Section	2,3
47	ID	
48	Dashboard	2
49	Application	
50	Course Material	2
51	Book	2
52	Content	2
53	Database	2,4,7
54	GP	
55	Feedback	2,3

56	Manual checking	3
57	Script	2
58	Result	2
59	Re-examine	2,3
60	Particular fields	
61	Midterm Exam Date	
62	Final Exam Date	
63	Own choice	
64	Warning	3
65	Critical Attendance Level	

6.5 SELECTION CRITERIA

The potential classes were then selected as classes by six Selection Criteria. A potential class becomes a class when it fulfills all six characteristics.

1.Retain information

2.Needed services

3.Multiple attributes

4.Common attributes

5.Common operations

6.Essential requirements

Table:2 Selection Criteria

No	**Noun**	Selection Criteria
1	Teacher	1-5
2	Student	1-5
3	User	1-5
4	Account	1-5
5	System	6
6	Database	6
7	Exam	2,6
9	Course plan	1,2
10	Video Section	1,2,3
11	Feedback	1,2,3
12	Course creation	2,6
13	Reexamine	2,6
14	Course Content	2,3,6

6.6 ATTRIBUTE SELECTION

After identifying the classes, we have specified their attributes and methods.

Table:3 Attribute Selection

No	Name	Attribute
1	User	full name user ID university department e-mail phone number photo username password

2	Teacher	full name user ID university department e-mail phone number photo username password Designation
3	Student	full name user ID university department e-mail phone number photo username password attendance critical attendance level

4	Account	full name user ID university department e-mail phone number photo username password
5	System	confirmation profile warning photo attendance
6	Video Section	Video class Live class
7	Course Plan	Marks distribution template Criteria Marks Allocation Number of classes provable assignments

		provable Lab works exam date notice
8	Database	User Information University Information
9	Exam	Question paper Answer sheet Marks
10	course Content	assignment course plan materials
11	Re-examine	manual checking course code course name tabulation sheet accurate answer GP
12	Feedback	Feedback
13	Course Creation	CourseCode

6.7 METHOD IDENTIFICATION

After identifying the classes, we have specified their methods.

Table:4 Method Identification

No	Noun	Method
1	User	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber() getPhoto() getUsername() getPassword() setFullName() setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber()

		setPhoto() setUsername() setPassword() SearchAndView()
2	Teacher	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber() getPhoto() getUsername() getPassword() getDesignation() setDesignation() setFullName() setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber()

		setPhoto() setUsername() setPassword() createCourse() takeAttencende() invitation() SearchAndView() upload() download() shareNotice()
3	Student	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber() getPhoto() getUsername() getPassword() getAttendance() setFullName()

		setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber() setPhoto() setUsername() setPassword() setAttendance() SearchAndView() uploadAssignment() download() joinVideoClass() giveFeedback() challengeForReexamine()
4	Account	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber()

		getPhoto() getUsername() getPassword() setFullName() setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber() setPhoto() setUsername() setPassword() modifyInformation()
5	System	VerifyingUser() notifyingUser() warning()
6	Video Section	uploadingVideoclass() liveVideoClass() downloadingVideoClass() sharingVideo()

7	Course Plan	setMarksDistributionTemplate() setCriteria() setMarksAllocation () getMarksDistributionTemplate() getCriteria() getmarksAllocation () setCoursePlan() getCoursePlane() setProvableAssignments() getProvableAssignments() setProvableLabWorks() getProvableLabWorks() setNumberOfClass() getNumberOfClass() setExamDate() getExamDate()
8	Database	getInformation() savingInformation() getAndStoreContent() searchingInformation() ViewingInformation()

		downloadingInformation() updatingInformation() removingInformation()
9	Exam	setQuestionPaper() setAnswerSheet() getQuestionPaper() getAnswerSheet() checkingScript() recordingMarks()
10	Content	getAssignDate() setAssignDate() setCoursePlan() collectAssignments() checkAssignments() uploadCourseMaterials()

11	Re-examine	getCourseName() setCourseName() getCourseCode() setCourseCode() makeTabulation() selectAccurateAnswer() makeGP() modifyResult()
12	Feedback	sendingFeedback() RespondingToFeedback()
13	Course Creation	creatingCourse() getCourseCode() setCourseCode() invitingStudent() verification()[1]

Note:

1. In verification () method, the System will check whether a user is valid or not to create a course according to user's university database.

6.8 Analyzing Classes

- User has common methods and attributes with account. So, we merge them as User and add the common and unique attributes and methods to User.
- Here, User is super class for student and teacher class.
- System class will be used for recording attendance, sending notice and warning to the students and verifying the users.
- Database class is for saving, searching, updating and removing information into the system.
- Course creation class is for creating course and sending invitation to the students.
- Video class is for uploading, downloading, sharing video content and live streaming.
- Exam class will be used for exam and checking script.
- Course plan is for designing the course.
- Here, Result class is not selected in general classification and selection criteria as a class. But this class is important to complete the system.For this reason, we select Result class as an environment class for this class. Result class is for making result and sending result to students.
- Reexamine class will be used to send request for reexamine.
- Feedback class will be used to send feedback.
- Here, assignment class is not selected in general classification and selection criteria as a class. But this class is important to complete the system. For this reason, we select assignment class as an environment class for this class. Assignment class is for submitting assignments and checking assignments.
- Dashboard class will be used as a notice board.

- Course Content class is for uploading the contents and downloading contents of this course.

6.9 Class Card

After identifying our final classes we have generated the following class cards.

Table:5 Class Card for User

User	
Attribute	Method
full name user ID university department e-mail phone number photo username password	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber() getPhoto() getUsername() getPassword() setFullName() setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber() setPhoto() setUsername() setPassword() Searching() Viewing()

Responsibilities	Collaboration
1. Creating new account 2. Involving with system operations (example : uploading study materials)	System Database Dashboard Video section Course content Assignment

Table: 6 Class Card for Student

Student	
Attribute	Method
full name user ID university department e-mail phone number photo username password	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber() getPhoto() getUsername() getPassword() setFullName() setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber() setPhoto() setUsername() setPassword() Searching() Viewing()

	isStudentOfAnyCourse() getNameOfCourses()
Responsibilities	Collaboration
1. Creating new account 2. Involving with system operations(example : downloading study materials)	System Database Dashboard Video section Course content Assignment Reexamine Feedback

Table:7 Class Card for Teacher

Teacher	
Attribute	Method
full name user ID university department e-mail phone number photo username password	getFullName() getUserID() getUniversity() getDepartment() getEmail() getPhoneNumber() getPhoto() getUsername() getPassword() setFullName() setUserID() setUniversity() setDepartment() setEmail() setPhoneNumber() setPhoto() setUsername() setPassword()

	Searching() Viewing() isTeacherOfAnyCourse() getNameOfCourses()
Responsibilities	Collaboration
1. Creating new account 2. Involving with system operations(example : uploading study materials)	System Database Dashboard Video section Course content Assignment Course creation Course plan Result Course content

Table:8 Class Card for System

System	
Attribute	Method
profile warning photo attendance	recordingProfile() sendingWarning() takePhoto() recordingAttendance() verification() sendingNotice()
Responsibilities	Collaboration

1. Handling database 2. Verifying user 3. Recording attendance 4. Sending notice and warning	User Database Dashboard

Table: 9 Class Card for Database

Database	
Attribute	Method
User Information University Information	getInformation() savingInformation() getAndStoreContent() searchingInformation() ViewingInformation() downloadingInformation() updatingInformation() removingInformation()
Responsibilities	Collaboration
1. Saving information 2. Searching Information 3. Updating and removing information	User System Result

Table: 10 Class Card for course Creation

Course Creation	
Attribute	Method
Course code	creatingCourse() verification() getCourseCode() setCourseCode() invitingStudent()
Responsibilities	Collaboration
1. Creating course 2. Sending invitation to students	Teacher Student System Database

Table:11 Class Card for section

Video Section	
Attribute	Method
Video class Live class	uploadingVideo() liveVideoClass() downloadingVideoClass() sharingVideo()
Responsibilities	Collaboration
1. Uploading video 2. Live streaming 3. Sharing video class 4. Downloading video contents	User Teacher system

Table:12 Class Card for Exam

Exam	
Attribute	Method
Question paper Answer sheet Marks	setQuestionPaper() setAnswerSheet() getQuestionPaper() getAnswerSheet() checkingScript() recordingMarks()
Responsibilities	Collaboration
1. Creating question and answer sheet. 2. Checking script and recording marks.	Student Teacher Database

Table:13 Class Card for course Plan

Course Plan	
Attribute	Method
Marks distribution template Criteria Marks Allocation Number of classes provable assignments provable Lab works exam date notice	setMarksDistributionTemplate() setCriteria() setMarksAllocation () getMarksDistributionTemplate() getCriteria() getmarksAllocation () setNotice() givingNotice() setProvableAssignments()

	getProvableAssignments() setProvableLabWorks() getProvableLabWorks() setNumberOfClass() getNumberOfClass() setExamDate() getExamDate()
Responsibilities	Collaboration
1. Choosing number criteria. 2. Allocating marks. 3. Setting number of classes, exam date, provable lab works, provable assignments.	Teacher System

Table:14 Class Card for Result

Result	
Attribute	Method
mark	calculateTotalMarks() getGradeSheet() sendResult()
Responsibilities	Collaboration

1. Calculating total marks and grade point. 2. Sending result to the students.	System Student Teacher

Table:15 Class Card for Re-Examine

Re-Examine	
Attribute	Methodo
	sendingRequestToReExamine() reexamine() sendingReExamineResult()
Responsibilities	Collaboration
1. Sending request to reexamine. 2. Reexamining. 3. Sending re-examined result.	Teacher Student

Table:16 Class Card for Feedback

Feedback	
Attribute	Method

feedback	setFeedback() getFeedback() sendingFeedback() RespondingToFeedback()
Responsibilities	Collaboration
1. Sending feedback 2. Responding to the feedback	System Student Teacher

Table:17 Class Card for Assignment

Assignment	
Attribute	Method
assignment	assignmentSubmission() checkAssignment()
Responsibilities	Collaboration
1. Submitting and checking assignments.	Teacher Student

Table:18 Class Card for Dashboard

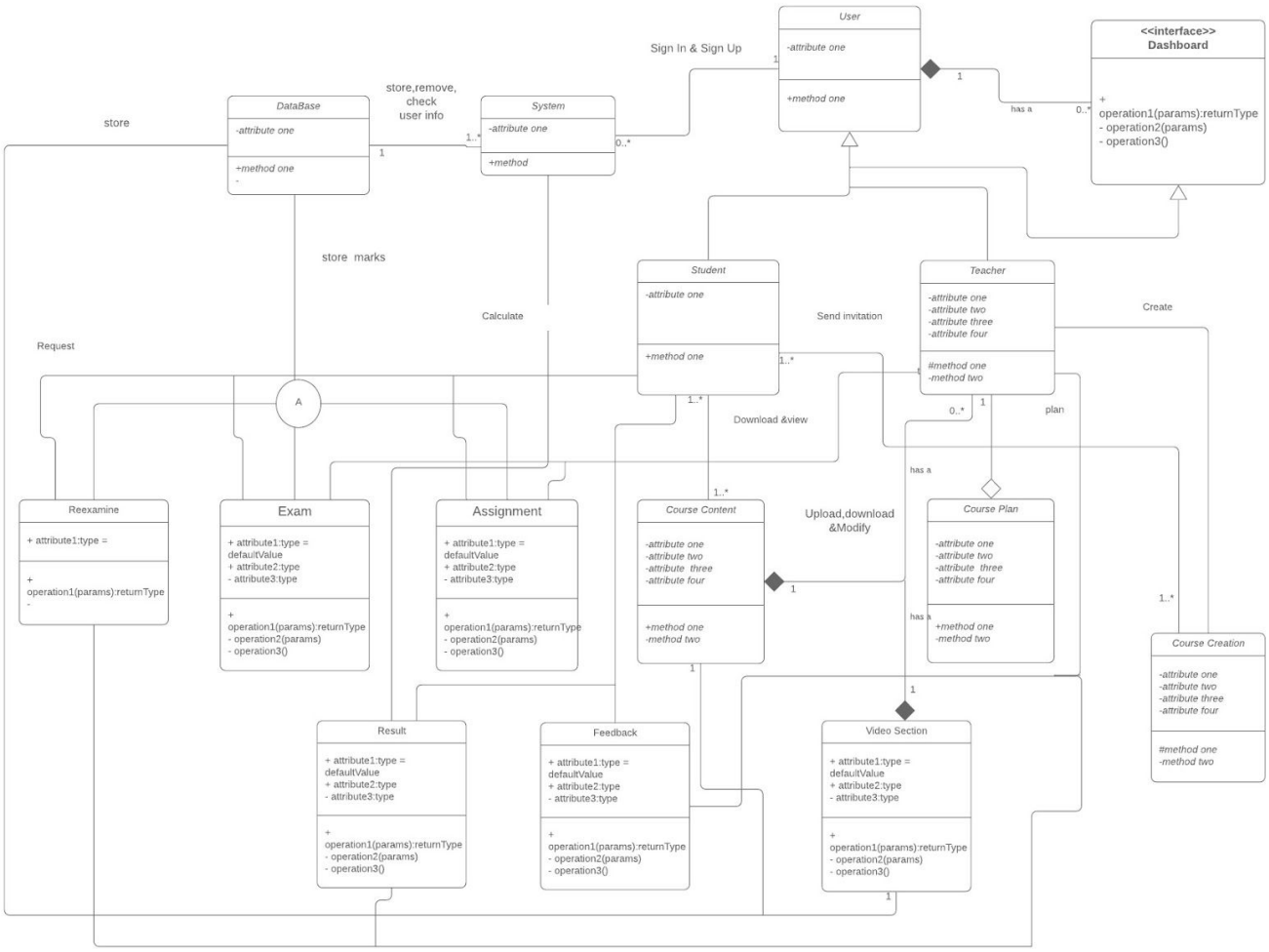
Dashboard	
Attribute	Method

Upcoming event Notice Upcoming assignment pending assignment role	showingUpcomingEvent() showingNotice() showingUpcomingAssignment() showingpendingAssignment() showingRole()
Responsibilities	Collaboration
1. Showing notice, upcoming event and details of assignments.	User System

Table:19 Class Card for course content

Course Content	
Attribute	Method
Materials Course plan	uploadingMaterials() uploadingCoursePlan() downloadingMaterials() downloadingCoursePlan()
Responsibilities	Collaboration
1. Uploading and downloading materials.	Student Teacher

6.10 CRC Diagram



7. BEHAVIORAL MODELING OF CMS

Serial NO	Event	Event Name	Initiator	Collaborator
1	Entering into in CMS	entry	user	System

2	Creating account	Sign up	user	System
3	Providing information[1]	Providing_info	User	System
4	Verifying information	Verification	System	
5	Re-entering email and username[2]	Reentering_email _username	User	
6	Storing information in database	Storing	System	Database
7	Notifying user	Notification	System	User
8	Responding to notification	Confirmation	User	
9	Providing credentials	Sign In	User	System
10	Recovering information via e-mail	Recovering	User	
11	Creating new course	Course_creation	Teacher	Student, Course creation, System
12	Identity verification[3]	Identification	System	Teacher
13	Selecting students	Student_selection	Teacher	Student
14	Inviting students	invitation	Teacher	Student
15	Creating course plan	Course_planning	Teacher	Course plan
16	Selecting criteria[4]	Criteria_selection	Teacher	System, Course plan
17	Allocating marks	Marks_allocation	Teacher	System, Course plan
18	Uploading course contents	Uploading_content	Teacher	System, Course content, Video section, Database

19	Downloading contents	Downloading_contents	User	Database, System, Course content, Video section
20	Viewing course contents	Viewing_content	user	Database, System, Course content, Video section
21	Attending online class	Attending_online_class	User	System,Video section
22	Sending notice	Sending_notice	Teacher	System
23	Sending warning via email[5]	Sending_warning	Teacher	System Student
24	Showing warning in dashboard	Showing_warning	System	Student, Dashboard
25	Showing notice in dashboard	Showing_notice	System	Student, Teacher, Dashboard
26	Showing user info in dashboard	Showing_info	System	User, Dashboard
27	Showing upcoming and pending events	Showing_events	system	User, Dashboard
28	Sending notice for upcoming assignments	Sending_notice	Teacher	Student
29	Uploading assignments	Uploading_assignments	Student	System, Database, Assignment
30	Checking assignments	Checking_assignments	Teacher	
31	Recording assignment's marks	Recording_ass_marks	Database	System, Assignment
32	Taking photo	Taking_photo	System	Student

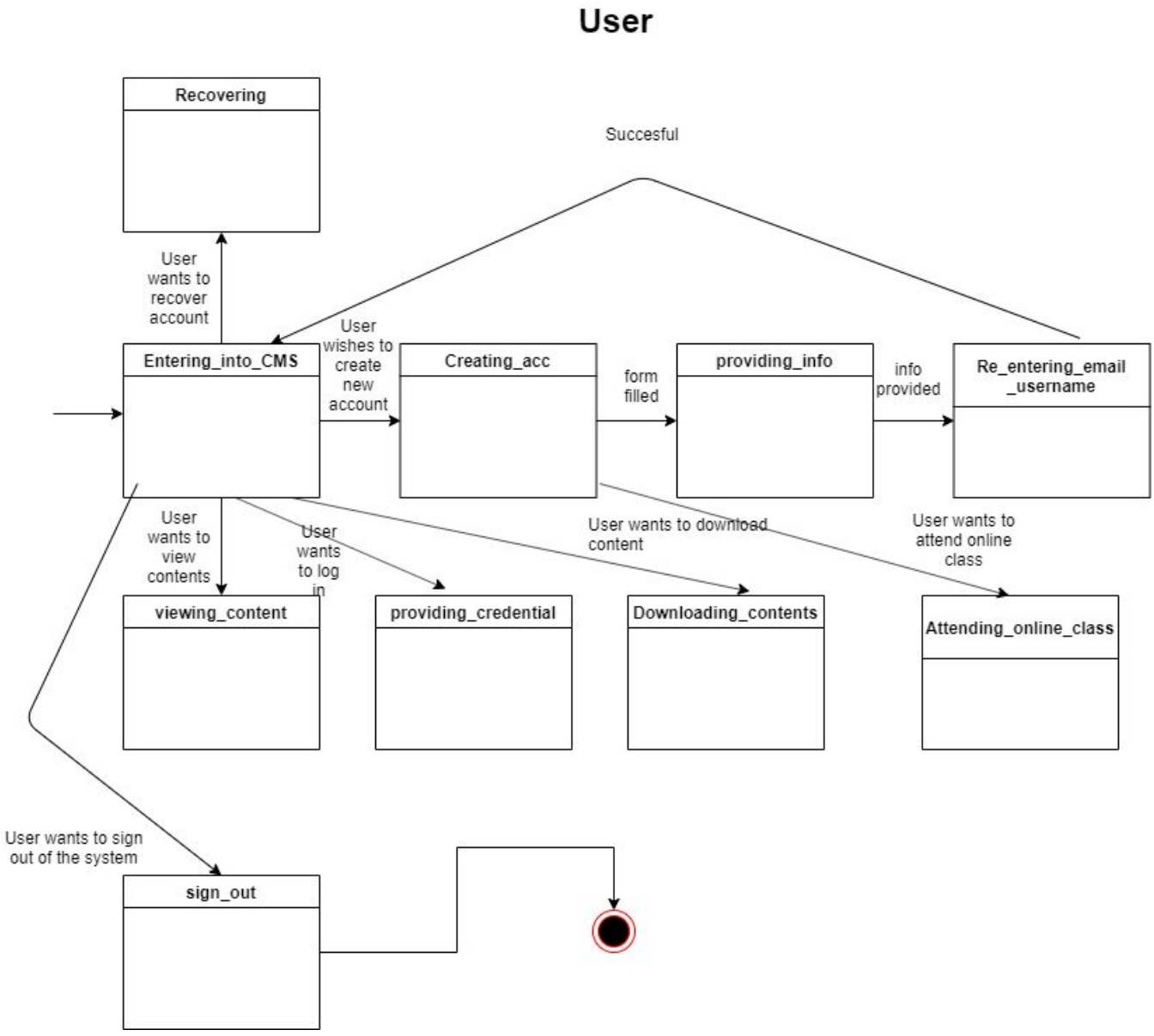
33	Recording attendance via facial recognition	Recording_attendence	System	Database
34	Announcing exam schedules	Announcing_exam_dates	teacher	Student, Exam
35	Creating question paper	questionPaper_creation	Teacher	System, Exam
36	Creating answer sheet	answerSheet_creation	Teacher	System, Exam
37	Recording exam marks	Recording_marks	Database	System, Exam
38	Calculating total marks	Marks calculation	system	Result
39	Calculating grade point	Gp calculation	System	Result
40	Sending result via email	Sending_result	Teacher	Student, System, Result
41	Uploading result into dashboard	Uploading_result	Teacher	System, Result, Course Plan
42	Updating mark	Updating_marks	System	Database
43	Requesting for reexamine	Requesting_reexamine	Student	Teacher, Reexamine
44	Rechecking script	Rechecking_script	Teacher	Reexamine
45	Sending reexamined result	Sending_result	System	Student, Reexamine
46	Giving feedback on reexamined result	Giving_feedback	Student	Teacher,Feedback
47	Giving anonymous feedback about course	Giving_anonymous_feedback	Student	Teacher, Feedback

48	Maintaining privacy of grade sheet	Maintaining_privacy	System	Result
49	Signing out	Sign out	User	System

NOTE

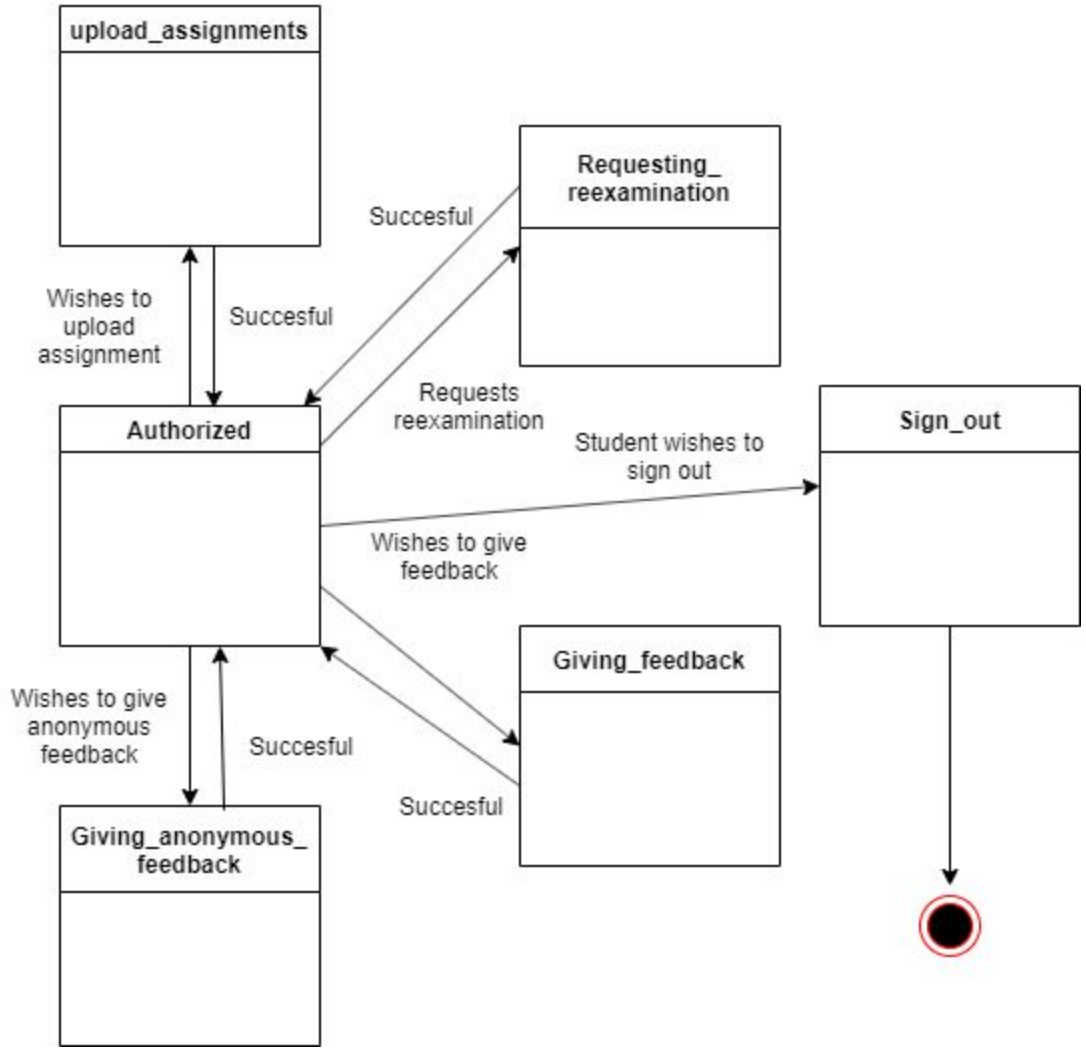
1. To create an account user has to provide information.
2. If the email or username already exists then the system will request user to re-enter email username.
3. If a user wants to create a new course, his or her identity will be verified from the university database whether he is a teacher or not.
4. After creating a course teacher will able to choose criteria.

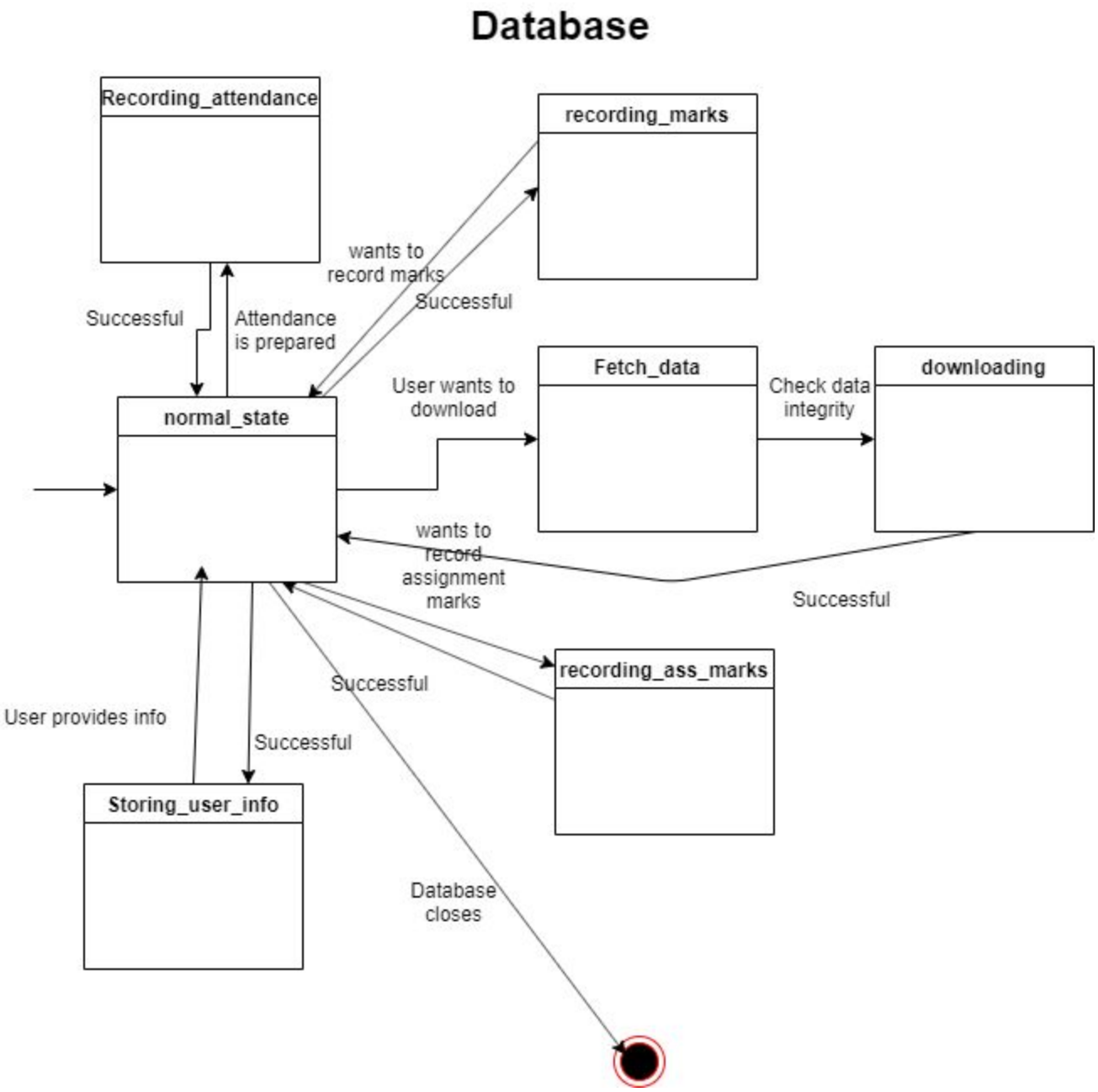
7.1 State Diagrams

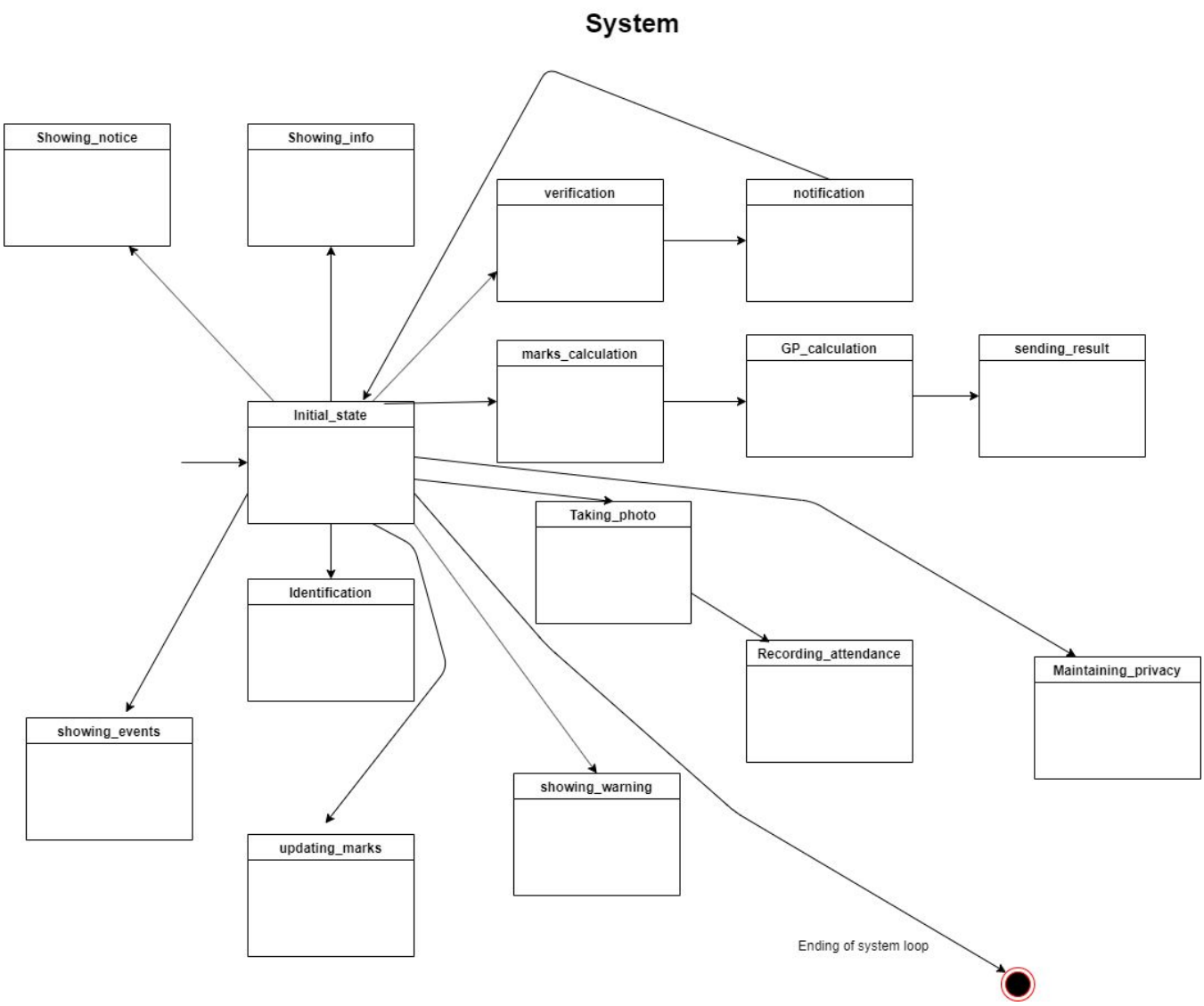


After successful completion of all states except Creating_acc and its consecutives, the state returns to Entering_into_CMS.

Student

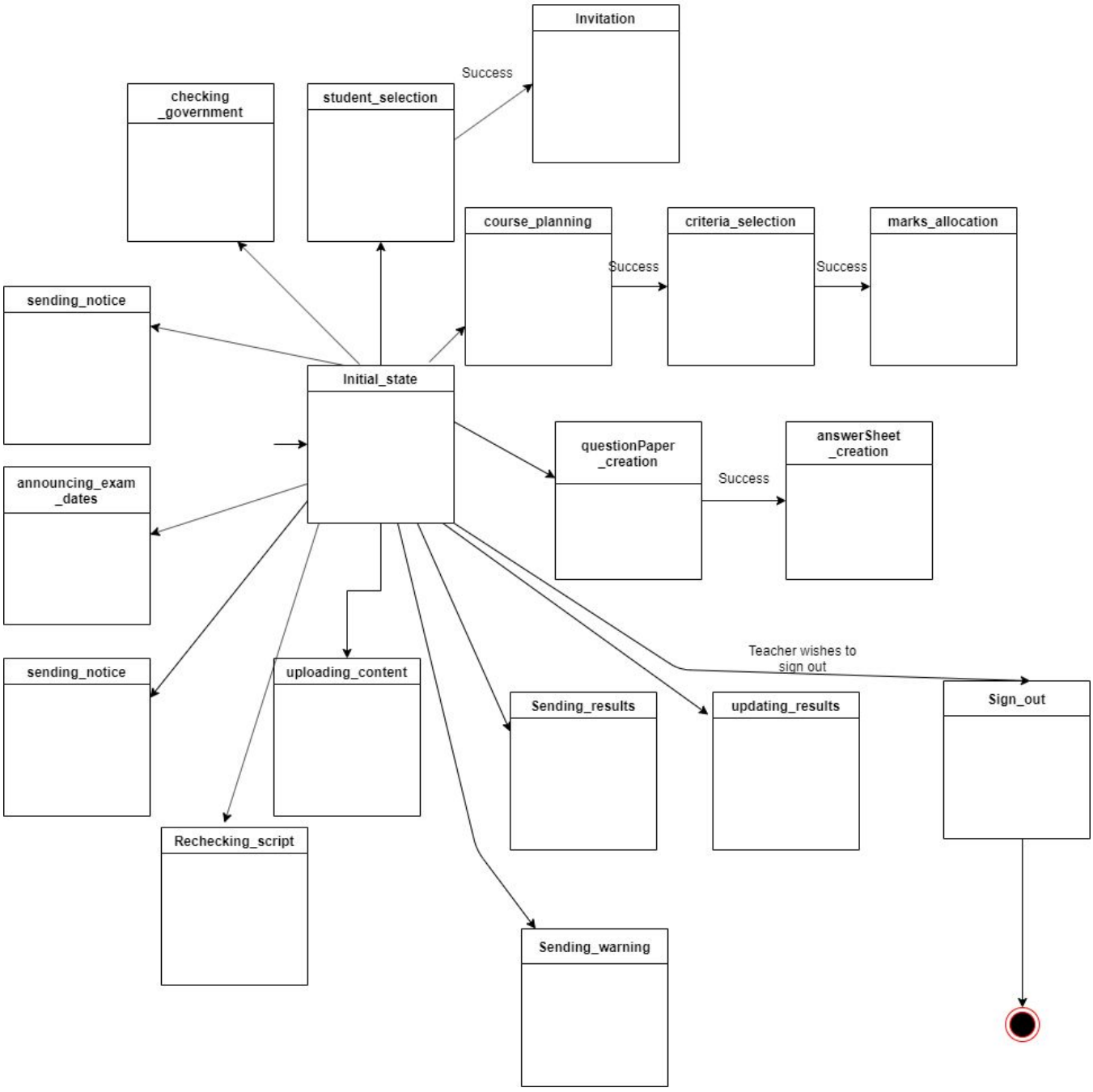

upload_assignments
Requesting_ reexamination
Succesful
Wishes to upload assignment
Succesful
Requests reexamination
Authorized
Sign_out
Student wishes to sign out
Wishes to give feedback
Giving_feedback
Wishes to give anonymous feedback
Succesful
Succesful
Giving_anonymous_ feedback


Database
Recording_attendance
recording_marks
wants to
record marks
Successful
Successful
Attendance
is prepared
normal_state
User wants to
download
Fetch_data
Check data
integrity
downloading
wants to
record
assignment
marks
Successful
recording_ass_marks
Successful
User provides info
Successful
Storing_user_info
Database
closes

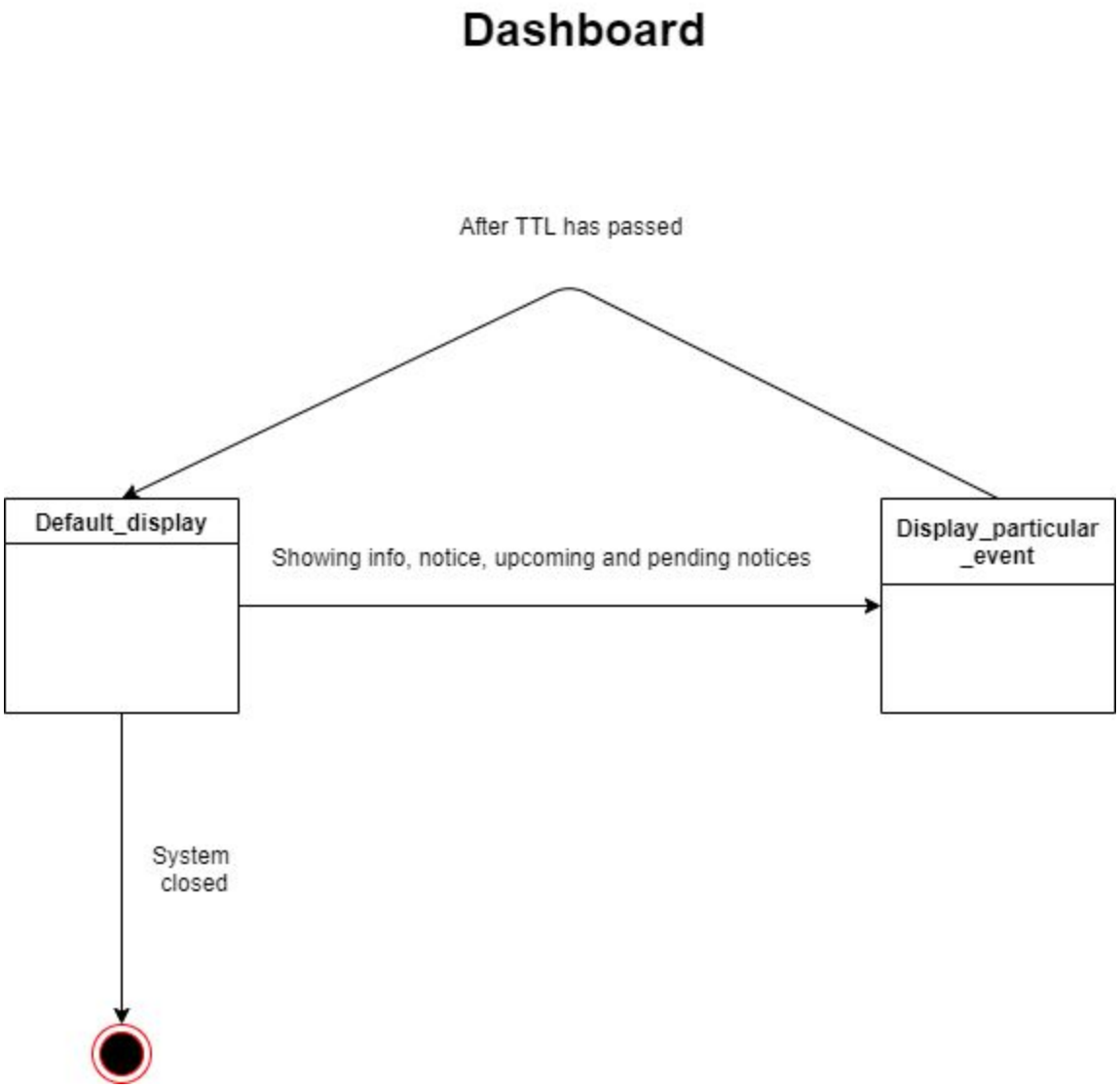


All outward arrows mean system wishes for particular action. After every dead ending state there is an inward arrow to initial state which is omitted for redundancy.

Teacher


Invitation
Success
checking _government
student_selection
course_planning
criteria_selection
marks_allocation
Success
Success
sending_notice
Initial_state
questionPaper _creation
answerSheet _creation
Success
announcing_exam _dates
Teacher wishes to sign out
sending_notice
uploading_content
Sending_results
updating_results
Sign_out
Rechecking_script
Sending_warning

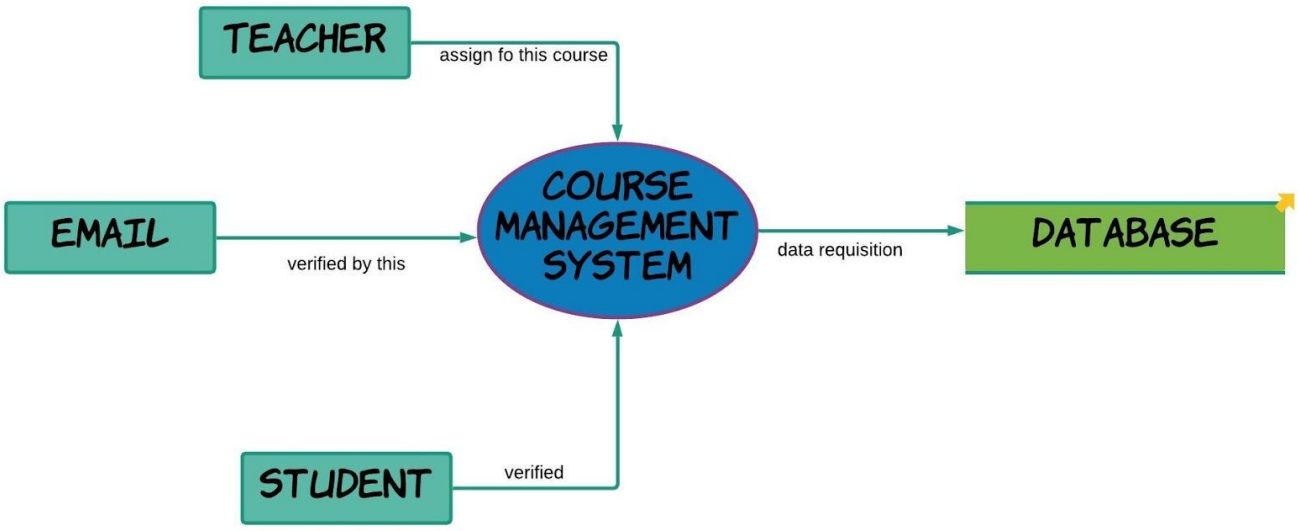
Note: All outward arrows mean system wishes for particular action. After every dead ending state there is an inward arrow to initial state which is omitted for redundancy.



7.2 SEQUENCE DIAGRAM:

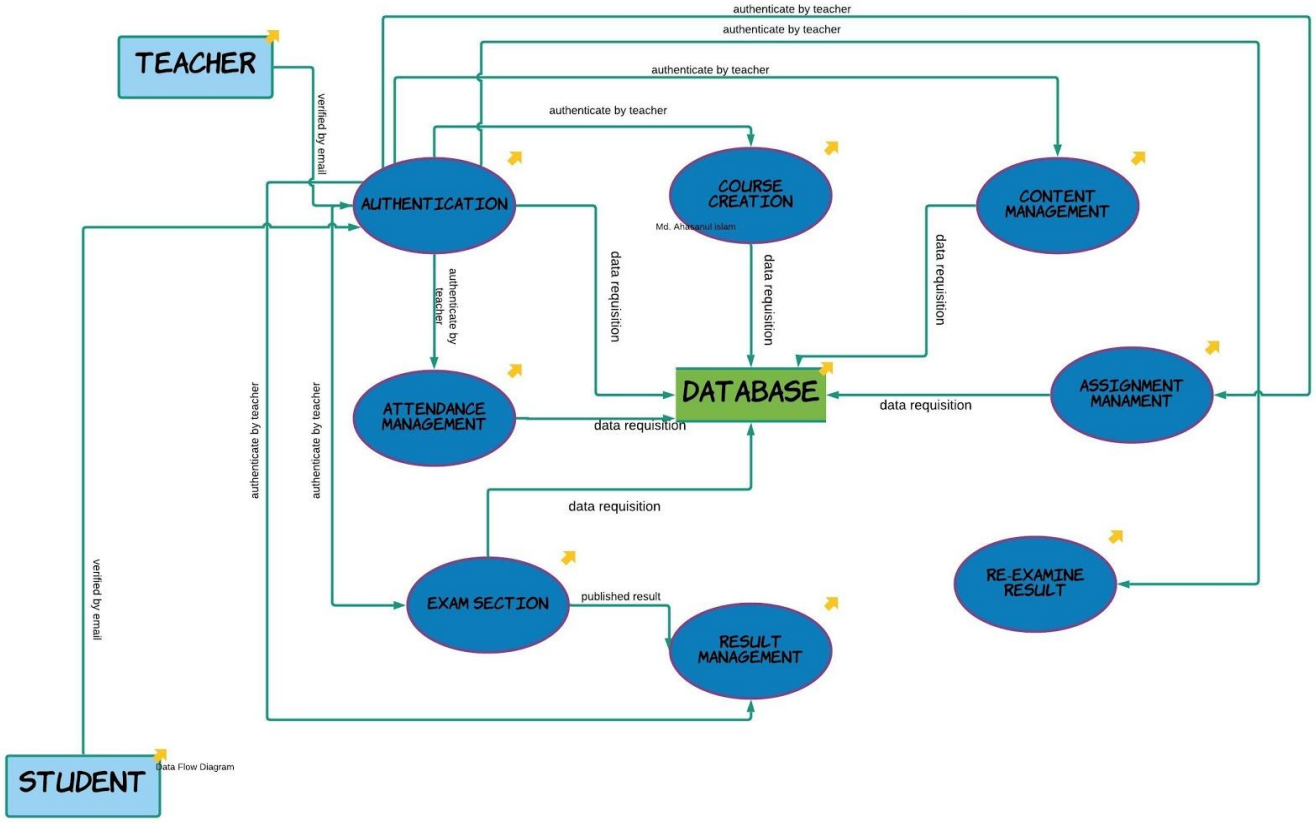
8. DFD

LEVEL 0:



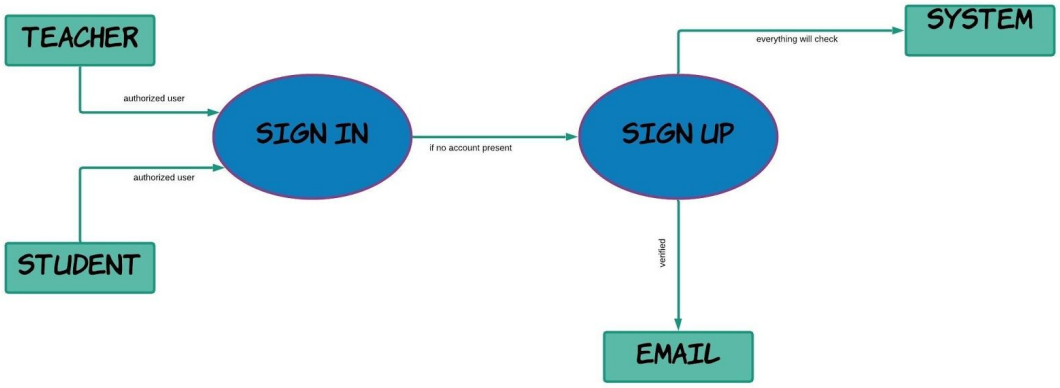
Level : 0

LEVEL 1:



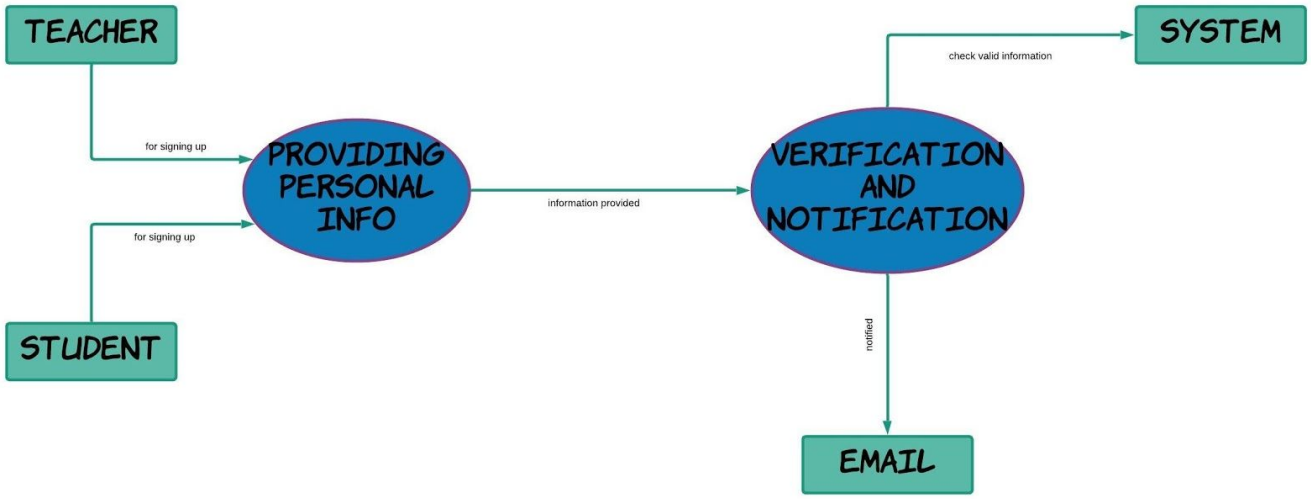
Level : 1

LEVEL 1.1:



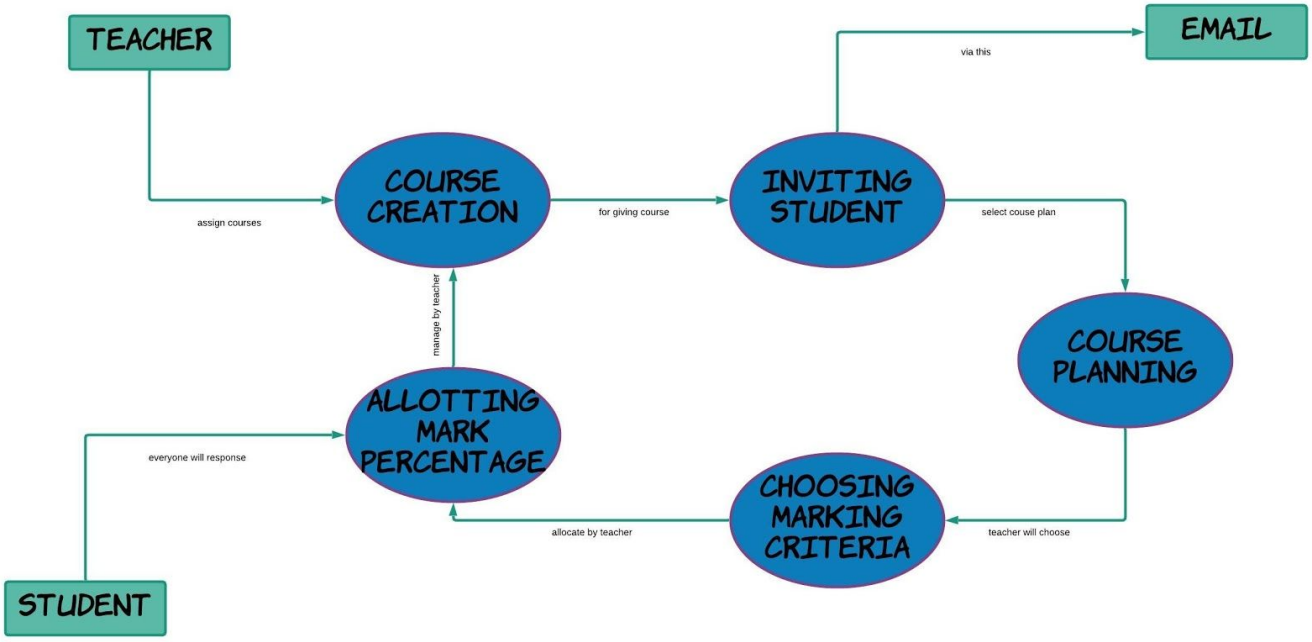
Level : 1.1(Authentication)

LEVEL 1.1.2:



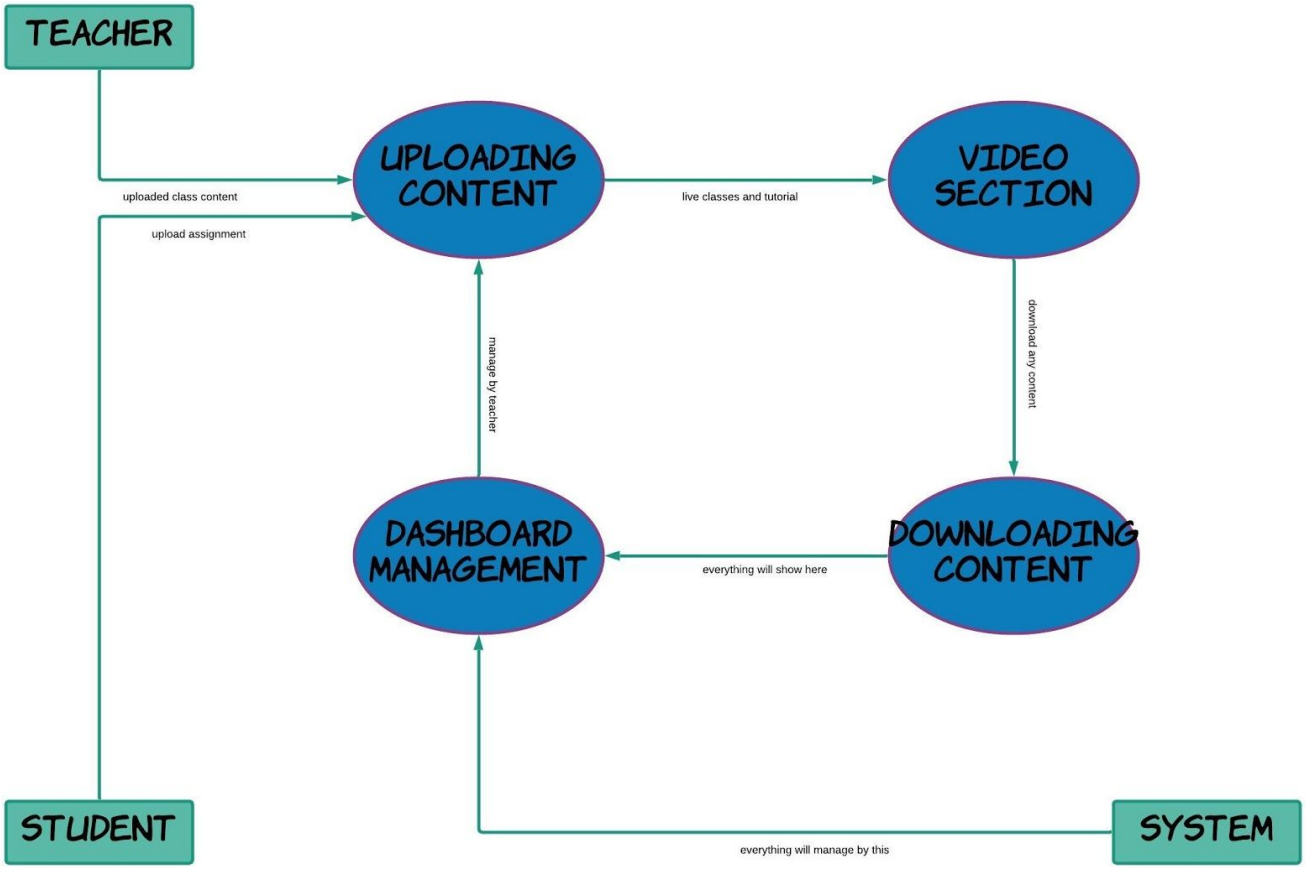
Level : 1.1.2(Sign Up)

LEVEL 1.2:



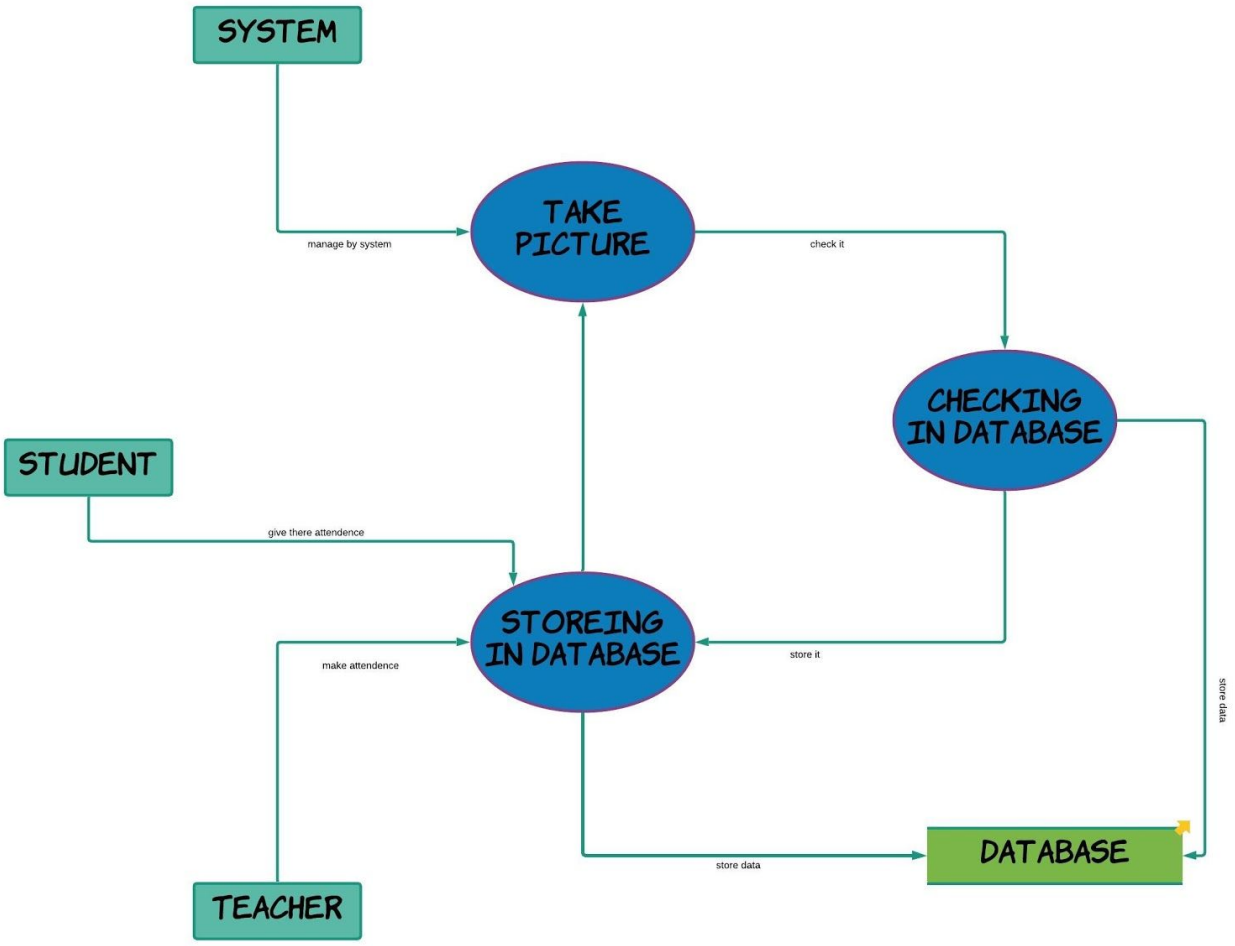
Level : 1.2(Course creation)

LEVEL 1.3:



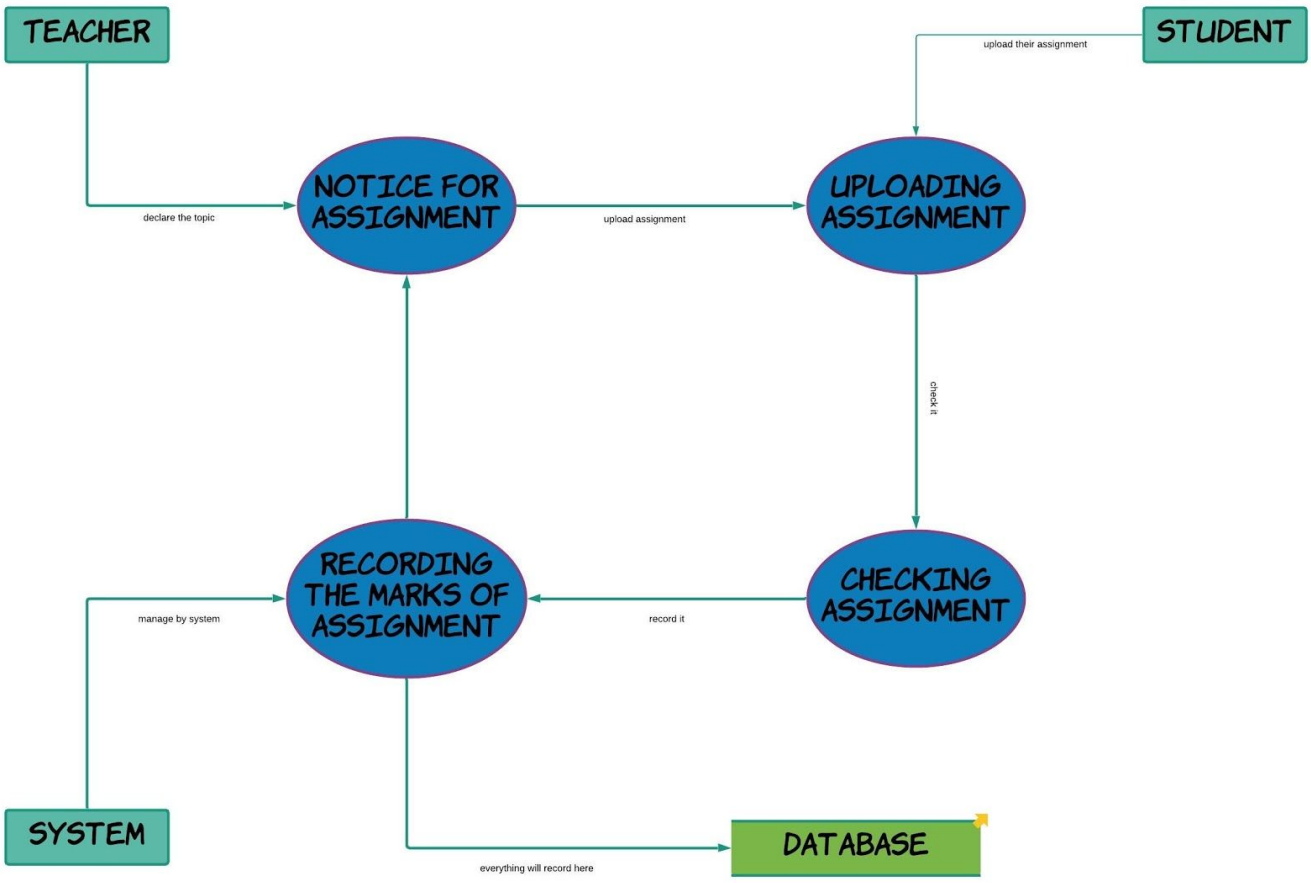
Level : 1.3(Content Management)

LEVEL 1.4:



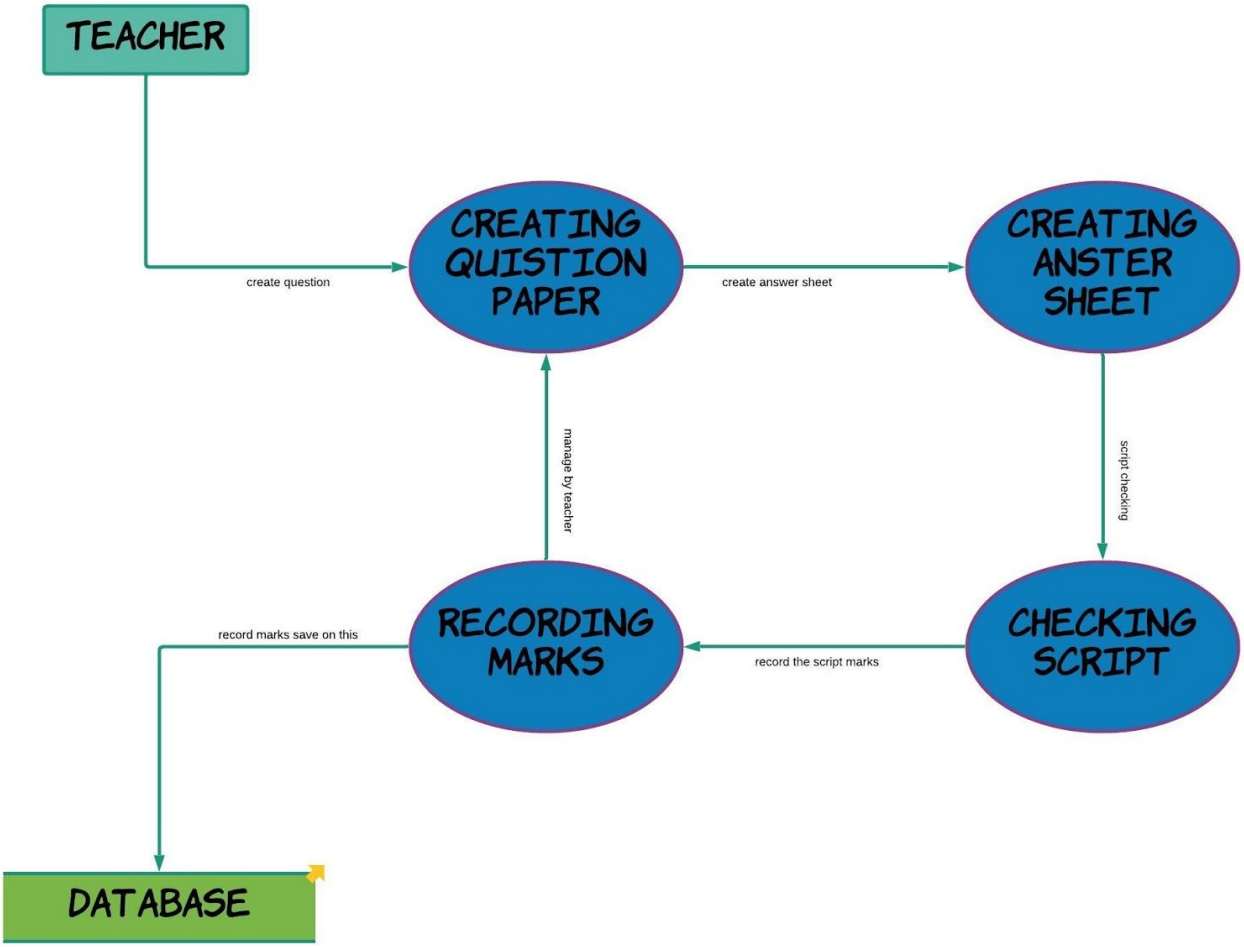
Level : 1.4(Attendence Management)

LEVEL 1.5:



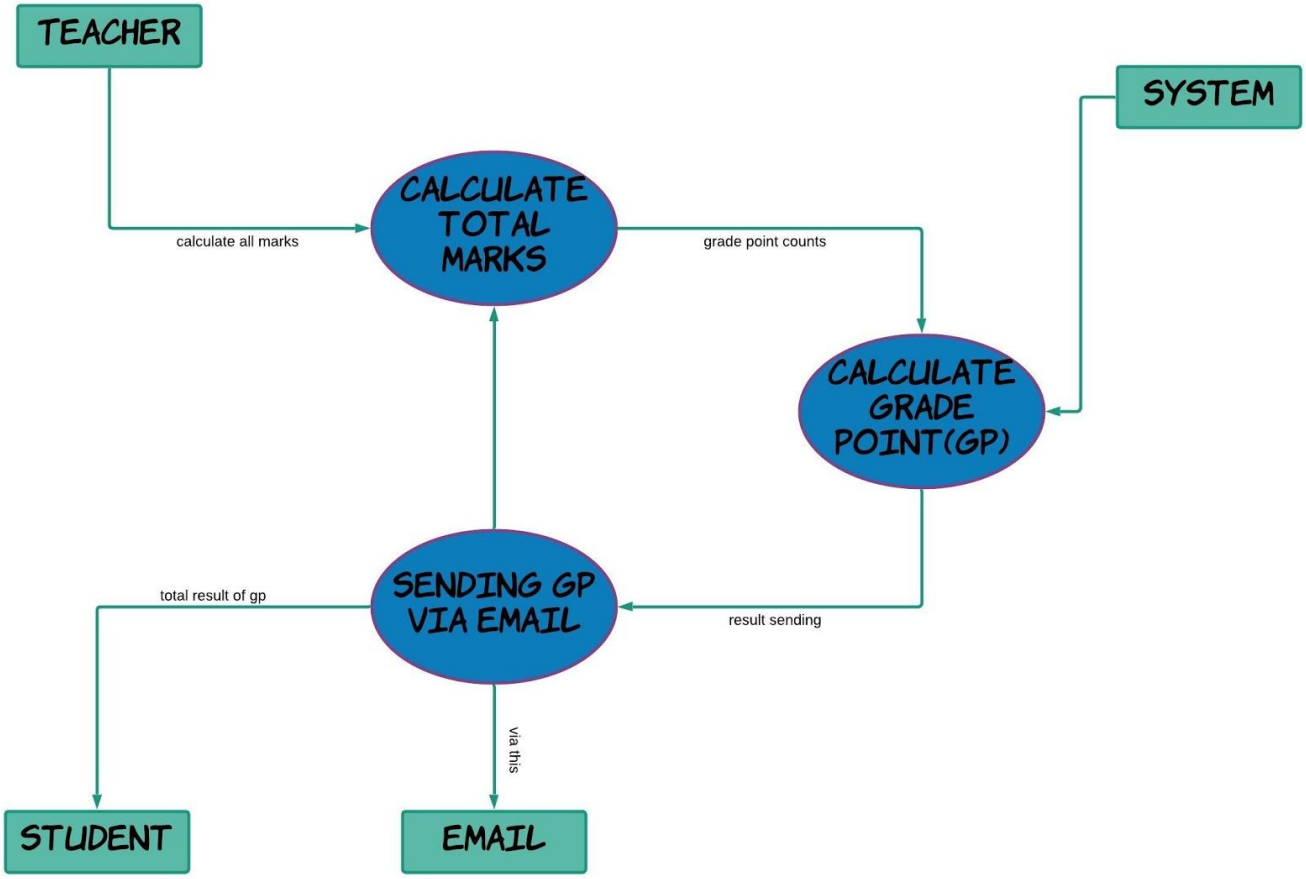
Level : 1.5(Attendence Management)

LEVEL 1.6:



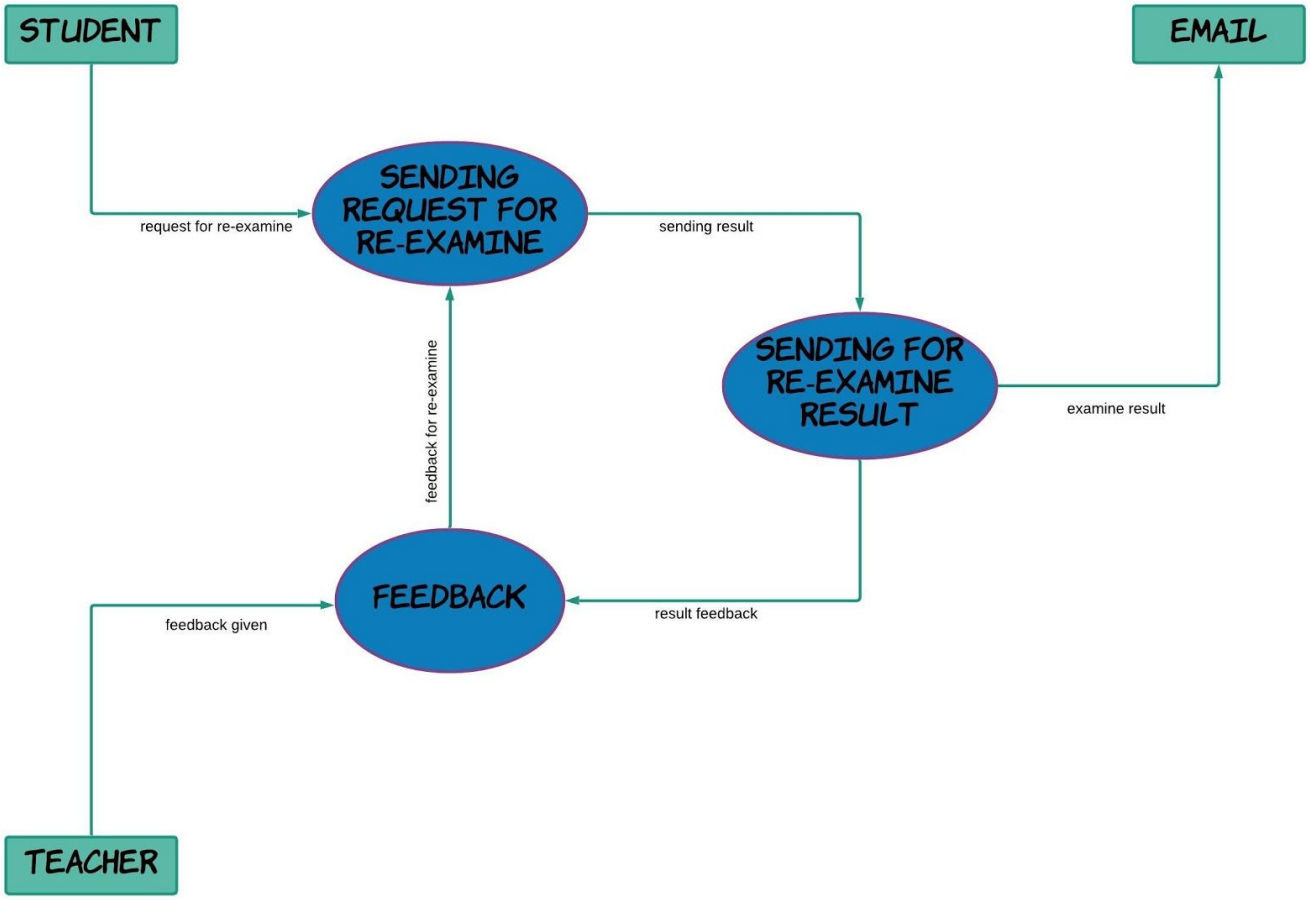
Level : 1.6(Exam Section)

LEVEL 1.7:



Level : 1.7(Result Management)

LEVEL 1.8:



Level : 1.8(Re-Examine and Feedback)

Full Report link :
https://docs.google.com/document/d/1VIyBLyLpfJ_Zsc2E_Cd4jA6uxvOnxzbDKALGYU6mUsc/edit?ts=5dcbf69f#

Sequence Diagram:
https://www.lucidchart.com/invitations/accept/76b27785-4921-4858-8521-17b36b7bef8a