**g**

**Software Requirements Specification**

**for**

Course Management

**Prepared by**

**Devansh Khandelwal(20BCS071)**

**Raj Tayde(20BCS174)**

**Jan 2023**

**Table of Contents (Just for your reference - Do not include in the module)**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Product Scope 1

1.3 References 1

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Use case diagram 2

2.5 Assumptions and Dependencies 2

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.3 Software Interfaces 3

**4. System Features (Functional Requirements) 4**

4.1 FR 1 4

4.2 FR 2 (and so on…) 4

**5. List Of Open Issues 5**

(**\*\*USE THE SAME FONT STYLE AND SIZE FOR UNIFORMITY**)

# **Introduction**

## **Purpose**

The main objective of the application is to offer a platform for lecturers to upload course-related documents, give assignments, and Announcements. The application also evaluates the assignment, and projects allotted to students and recommends optimal grades for each student. Students are able to see their performance, course content, submit assignments and ask questions.

## **Product Scope**

This software will be generic for all educational institutions, intuitive and user friendly. Lecturers can provide various assignments and they also see the student participation both individually and also for the complete course.

## **References**

http://www.uml-diagrams.org/use-case-include.html

http://pmblog.accompa.com/2009/10/08/use-case-template-example-requirementsmanagement basics/

http://www.batimes.com/articles/generalization-and-use-case-models-part-2.html http://aakashtechsupportdocs.readthedocs.org/en/latest/index.html

http://www.math-cs.gordon.edu/courses/cs211/ATMExample/

# **Overall Description**

## **Product Perspective**

The product will be a platform to upload Course Content like lectures PDFs, slides and various assignments. The lecturer will be advised with overall class performances and also individual performances so that he/she will get an idea of what he/she should change in his course accordingly. Students can form groups among themselves so that they can be assigned a project and evaluated accordingly.

## **Product Functions**

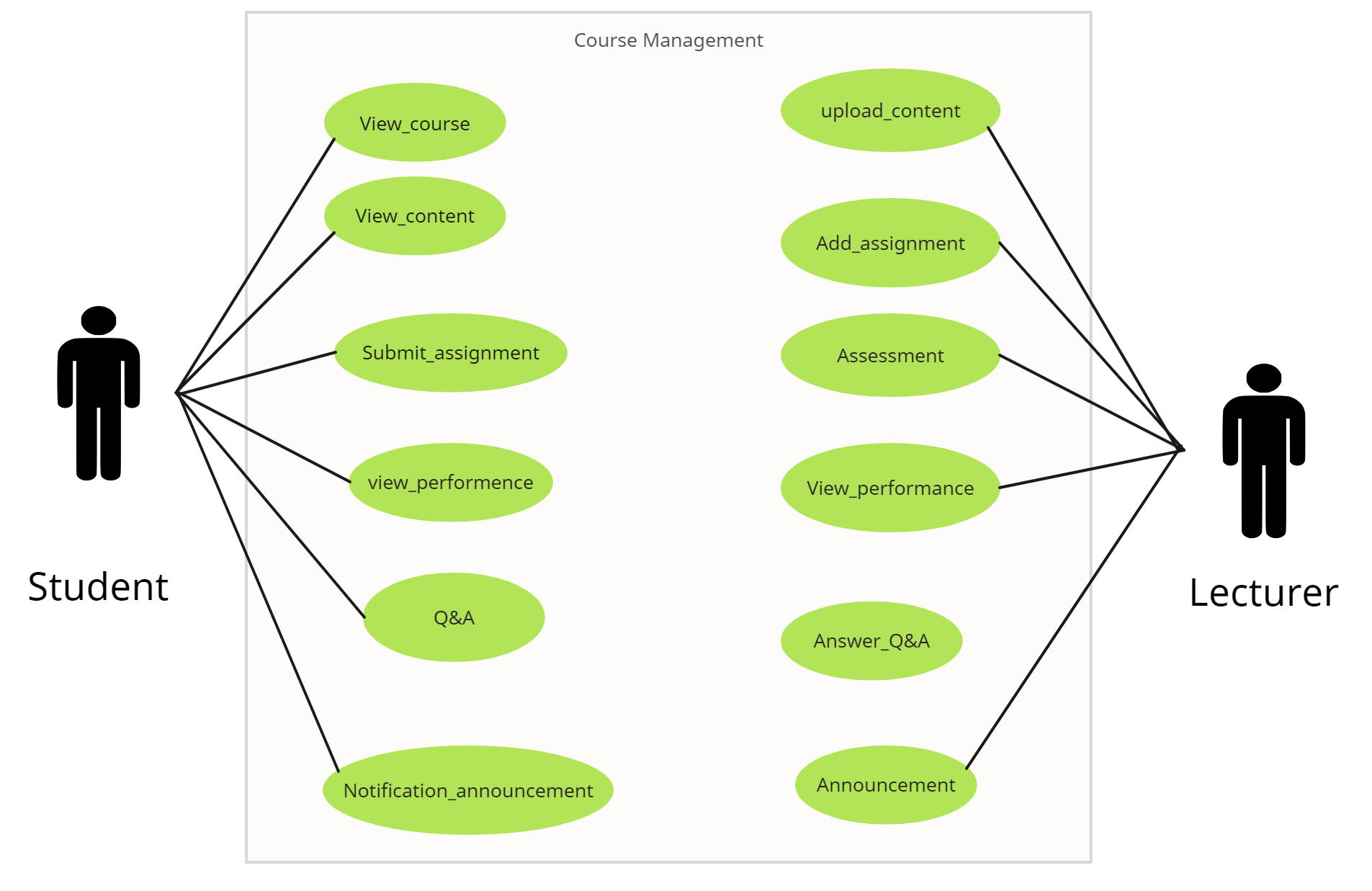
The main functions involved with the product are to organize quizzes, provide assignments and helpful documents of courses taught in the institution, the product also suggests the grade to be given on the quizzes conducted to each student and solve assignments. Students will do the peer assessment for the assignments. Each student will be assessing at least 3 of the assignment solutions submitted by their peers. The quiz will be time limited exams, while assignments have no time limit. Lecturers can also view his uploads and questions, student(s) performances (group and individual).

## **User Classes and Characteristics**

A student will be logging in with the password and email-id he has with him at the home page and proceeding further to his home page (Assignments,Quizzing and course documentation Module) where he will be shown his profile by default. He will be shown all the courses he has already registered for. When he clicks on a particular course user has 4 options in form of pages available to him/her, See Course Content and details, download assignment, Participate in a Quiz, See previous quiz performance, Form Groups/See group Details.

The lecturer will be logging in the same ways as the student does, and his home page will be his profile and from there he can go to the following four pages, Upload Content (like videos, documents, Assignments). Organize a quiz, Evaluate quizzes and Group Projects, View performances.

## User Case diagram.

****

# **External Interface Requirements**

## **User Interfaces**

A student will be logging in with the password and email-id he has with him at the home page and proceeding further to his home page (Assignments,Quizzing and course documentation Module) where he will be shown his profile by default. He will be shown all the courses he has already registered for. When he clicks on a particular course user has 4 options in form of pages available to him/her, See Course Content and details, download assignment, Participate in a Quiz, See previous quiz performance, Form Groups/See group Details.

The lecturer will be logging in the same ways as the student does, and his home page will be his profile and from there he can go to the following four pages, Upload Content (like videos, documents, Assignments). Organize a quiz, Evaluate quizzes and Group Projects, View performances.

## **Software Interfaces**

The system will be interacting with a database written in Django and will be working on web browsers like google chrome, Mozilla Firefox etc.

**3.3** [**UI/UX Design**](https://www.figma.com/file/I71r5x3sUlVq2OQEFLJRPR/FUSIONIIIT---AC3?node-id=246%3A262&t=oacBFgySYGISQgHf-0)

**3.4** [**DataBase Schema**](https://miro.com/app/board/uXjVPpiX7Es=/)

# **System Features (Functional Requirements)\**

**4.1.1 view\_content**

| **UC ID** | UC#1 | |
| --- | --- | --- |
| **Use case**  **Name** | view\_content | |
| **Description** | Student can see the content of various courses he is enrolled. (Use UC#) | |
| **Actor** | Student | |
| **Precondition** | 1 | The student must be logged in. |
| 2 | Student should have registered in that particular course. |
| **Main Flow** | 1 | Firstly student have to enter in the desired course of his choice |
|  | 2 | Student clicks on the option “View Content” |
| 3 | The system displays all the content related to the course(like slides and notes). |
| 4 | Student can view the displayed information and download it but cannot change anything. |
| 5 | The system returns to the employee ‘Dashboard’ – initial screen. |
| **Post**  **conditions** | Student performs the desired action and the system allows him to do them. | |
| **Alternate**  **Flow** | NIL | |
| **Sub Flow** | NIL | |
| **Global**  **Alternate**  **Flow** | NIL | |

**4.1.2 view\_course**

| **UC ID** | UC#2 | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Use case Name** | view\_course | | | | | |
| **Description** | Student can see the various details about the desired course. (Use UC#) | | | | | |
| **Actor** | Student | | | | | |
| **Precondition** | 1 | | | The student must be logged-in . | | |
| 2 | | | Student should have registered in that particular course. | | |
| **Main Flow** | 1 | | | | Student clicks on the “course name” | |
| 2 | | | | The system displays all the course details of the selected course. | |
| 3 | | | | Student can go to the assignment, the content of course, notification, etc. | |
| 4 | | | | Student can view the displayed information but cannot change anything. | |
| 5 | | | | The system returns to the employee ‘Dashboard’ – initial screen. | |
| **Post conditions** | Student performs the desired action and the system allows him to do them. | | | | | |
| **Alternate Flow** |  | | NIL | | | |
| **Sub Flow** |  | | NIL | | | |
| **Global Alternate Flow** |  | | NIL | | | |

**4.1.3 submit\_assignment**

| **UC ID** | UC#3 | |
| --- | --- | --- |
| **Use case Name** | submit\_assignment | |
| **Description** | Student can solve the assignment that is given, submit it, and can check previous assignments. | |
| **Actor** | Student | |
| **Precondition** | 1 | The student must be logged-in . |
| 2 | Student should have registered in that particular course. |
| 3 | The assignment should be open. |
| **Main Flow** | 1 | Firstly student have to enter in the desired course of his choice |
|  | 2 | Student clicks on the option “Assignment” |
| 3 | The system displays all assignments with deadlines. |
| 4 | The student can view the assignment which was previously given. |
| 5 | Student solves and submits the current assignment. |
| 6 | Submissions are taken in the form of PDF or DOCS. |
| 7 | After adding the PDF or DOCS click on the “Submit” button. |
| 8 | The system will display that the assignment is submitted. |
| **Post conditions** | Student performs the desired action and the system allows him to do them. | |
| **Alternate Flow** |  | NIL |
| **Sub Flow** |  | NIL |
| **Global Alternate Flow** | GA1 | Student can ‘cancel’ the Assignment at any time by exercising such an option |
|  | Post-condition – The system returns to the “Solve Assignment Dashboard” – initial screen. |

**4.1.4 view\_performance**

| **UC ID** | UC#4 | |
| --- | --- | --- |
| **Use case Name** | view\_performance | |
| **Description** | Student can view the Marks of Various Exams of Various Courses he is enrolled in. | |
| **Actor** | Student | |
| **Precondition** | 1 | The student must be logged-in. |
| 2 | Student should have registered in that particular course. |
| 3 | The Lecturer must have uploaded the Marks he is wishing to view. |
| **Main Flow** | 1 | Firstly student have to enter in the desired course of his choice |
|  | 2 | The system displays all the courses he is enrolled in on the Dashboard. |
| 3 | Student can select any course from the displayed courses. |
| 4 | The system displays all the options associated with the particular course. |
| 5 | Student selects the “Marks” Option. |
| 6 | The system displays him his score for the various exams he has given until now. |
| **Post conditions** | Student can view his Performance for the desired course he’s wished to see. | |
| **Alternate Flow** |  | NIL |
| **Sub Flow** |  | NIL |
| **Global Alternate Flow** |  | NIL |

**4.1.5 Q&A**

| **UC ID** | UC#5 | |
| --- | --- | --- |
| **Use case Name** | Q&A | |
| **Description** | Student can ask any question related to the Courses he’s enrolled in (Use UC#) | |
| **Actor** | Student | |
| **Precondition** | 1 | The student must be logged in. |
| 2 | Student should have registered in that particular course. |
| **Main Flow** | 1 | Firstly student have to enter in the desired course of his choice. |
|  | 2 | Student clicks on the option “Q&A” |
| 3 | Then the system displays him the Text Box . |
| 4 | Students can post their valid doubts along with description.[A1] |
| 5 | Student can also view the Question and answers of other Students and also the Lecturer response. |
| 6 | The system shows the acknowledgement.about his asked question. |
| 7 | Student can also delete his Question at any point of time if it was unaddressed by the Lecturer. |
| **Post conditions** | Student can post the Doubts on any course he is enrolled in | |
| **Alternate Flow** |  | NIL |

| **Sub Flow** |  | NIL |
| --- | --- | --- |
| **Global Alternate Flow** |  | NIL |

**4.1.6 notification\_announcement**

| **UC ID** | UC#6 | | |
| --- | --- | --- | --- |
| **Use case Name** | notification\_announcement | | |
| **Description** | Student can see the announcement of course he is enrolled. (Use UC#) | | |
| **Actor** | Student | | |
| **Precondition** | 1 | | The student must be logged in. |
| 2 | | Student should have registered in that particular course. |
| **Main Flow** | 1 | | Firstly student have to enter in the desired course of his choice |
|  | 2 | | Student clicks on the icon “Notification” |
| 3 | | Student can view the displayed notification but cannot change anything. |
| **Post conditions** | Student performs the desired action and the system allows him to do them. | | |
| **Alternate Flow** |  | NIL | |
| **Sub Flow** |  | NIL | |
| **Global Alternate Flow** |  | NIL | |

**Use Case Description**

**4.2.1 upload\_content**

| **UC ID** | UC#8 | |
| --- | --- | --- |
| **Use case**  **Name** | upload\_content | |
| **Description** | Lecturer can upload course-related documents. | |
| **Actor** | Lecturer | |
| **Precondition** | 1 | Lecturer should have been added by the admin. |
| 2 | Lecturer should have added a course to upload documents in it. |
| **Main Flow** | 1 | Lecturer clicks on the Upload Documents Button. |
| 2 | Lecturer selects the course |
| 3 | Lecturer uploads the file(s). |
| 4 | Lecturer clicks on submit button.[A1] |
| 5 | Includes Authentication Use Case. |
| **Post**  **conditions** | Lecturer will be able to upload the documents. | |
| **Alternate**  **Flow** | A1 | Lecturer cancels the documents. |
|  | Post-condition – Document will not upload and lecturer will be reverted back to the main page. |
| **Sub Flow** |  | NIL |
| **Global**  **Alternate**  **Flow** |  | NIL |

**4.2.2 add\_assignment**

| **UC ID** | UC#9 | |
| --- | --- | --- |
| **Use case**  **Name** | add\_assignment | |
| **Description** | Lecturer will add questions for assignment. | |
| **Actor** | Lecturer | |
| **Precondition** | 1 | Lecturer should have been added by the admin. |
| 2 | Lecturer should have added a course to add an assignment in it. |
| **Main Flow** | 1 | Lecturer selects the course |
| 2 | Lecturer clicks on the Add Assignment Button. |
| 3 | Lecturer will be redirected to add assignment page, where system will ask for question statement and answer options. |
| 4 | Lecturer clicks on submit button. [A1] |
| 5 | Includes Authentication Use Case. |
| **Post**  **conditions** | Lecturer will be adding questions. | |
| **Alternate**  **Flow** | A1 | Lecturer cancels the addition of assignment in between before its completion. |
|  | Post-condition – Assignment will not upload and lecturer will be reverted back to the main page. |
| **Sub Flow** |  | NIL |
| **Global**  **Alternate**  **Flow** |  | NIL |

**4.2.3 assessment**

| **UC ID** | UC#12 | |
| --- | --- | --- |
| **Use case**  **Name** | assessment | |
| **Description** | Lecturer needs to evaluate the assignments submitted by students for their specific courses. | |
| **Actor** | Lecturer | |
| **Precondition** | 1 | Lecturer should have been added by the admin. |
| 2 | Lecturer should have added a course to assest an assignment in it. |
| **Main Flow** | 1 | Lecturer clicks on the desired course. |
| 2 | Lecturer clicks on Assessment. |
| 3 | Depending upon the deadline of the assignment. the lecturer will see the list of names of student and their respective files ( For now in .zip or .pdf format). |
| 4 | Lecturer now evaluates all the submitted assignments.. |
| 5 | Includes Authentication Use Case. |

|  |  |  |
| --- | --- | --- |
| **Post**  **conditions** | Students will be submitting the assignment and get evaluated. | |
| **Alternate**  **Flow** | A1 | After the allotted time for the assignment expires the student will not be able to submit their their assignment. |
|  | Post-condition Student will not be able to submit their assignment. |
| **Sub Flow** |  | NIL |
| **Global**  **Alternate**  **Flow** |  | NIL |

**4.2.4 view\_performance**

| **UC ID** | UC#11 | |
| --- | --- | --- |
| **Use case**  **Name** | view\_performance | |
| **Description** | 1 | Lecturer can see the students’ performances in the quizzes he uploaded. |
| 2 | Lecturer can also see the individual set of answers attempted by a particular student. |
| **Actor** | Lecturer | |
| **Precondition** | 1 | Lecturer should have been added by the admin. |
| 2 | Lecturer should have created the quiz to which he wants to see the results. |
| **Main Flow** | 1 | Lecturer will login on to his account |
| 2 | Selects the specific course out of the ones he/she teaches. |
| 3 | Lecturer will be redirected to quiz performances page, where system asks which quiz result he/she wants to see. |
| 4 | Lecturer looks at the results and analyze performance. |
| **Post**  **conditions** | Lecturer will have and provide answer key of various quizzes. | |
| **Alternate**  **Flow** |  | NIL |
| **Sub Flow** |  | NIL |
| **Global**  **Alternate**  **Flow** |  | NIL |

**4.2.5 answer\_Q&A**

| **UC ID** | UC#7 | |
| --- | --- | --- |
| **Use case**  **Name** | answer\_Q&A | |
| **Description** | Lecturer can clarify the doubt on any course he is offering. | |
| **Actor** | Lecturer | |
| **Precondition** | He must be enrolled in the course he wishes to comment on. | |
| **Main Flow** | 1 | Lecturer will login. |
| 2 | Lecturer selects a course. |
| 3 | Lecturer clicks on the Q&A button. |
| 4 | Lecturer will be redirected to Q&A page, where he/she can perform the describe action. |
| 5 | Lecturer clicks on any question and answer that particular question. |
| 6 | A lecturer can delete his answer or question at any point of time. |
| **Post**  **conditions** | NIL | |
| **Alternate**  **Flow** | NIL | |
| **Sub Flow** |  | NIL |
| **Global**  **Alternate**  **Flow** |  | NIL |

**4.2.6 announcement**

| **UC ID** | UC#7 | |
| --- | --- | --- |
| **Use case**  **Name** | announcement | |
| **Description** | Lecturer can post any announcement related to the course.(Like: class timing, quiz date, etc.) | |
| **Actor** | Lecturer | |
| **Precondition** | He must be enrolled in the course he wishes to comment on. | |
| **Main Flow** | 1 | Lecturer will login. |
| 2 | Lecturer selects a course. |
| 3 | Lecturer clicks on the Q&A button. |
| 4 | Lecturer will be redirected to Q&A page, where he/she can perform the describe action. |
| 5 | Lecturer clicks on the Post button to successfully post the comment. |
| 6 | A lecturer can delete his comment and also any student’s comment at any point of time. |
| **Post**  **conditions** | NIL | |
| **Alternate**  **Flow** | NIL | |
| **Sub Flow** |  | NIL |
| **Global**  **Alternate**  **Flow** |  | NIL |

# Appendix

| Actor | The person who is interacting with the software. Here, in this case, students and instructors. |
| --- | --- |
| Admin | The admin is the person who takes care of the software and its state of working. |
| Assignments | Assignments are optional questions that can be solved by the student. |
| Courses | Course is a unit of teaching that typically lasts one academic term. It is offered by one or more instructors. |
| Course Documents | All helpful files including presentations and PDFs for the course. |
| Lecturer | The person who offers the course. |
| Student | The person who takes the course. |
| Quiz | Compulsory question paper that should be solved by the student in a limited time. |

# List of Open Issues

| **S. No** | **Issue Details** | **Category** | **Solution** | **Comments** |
| --- | --- | --- | --- | --- |
| 1 |  |  |  |  |
| 2 |  |  |  |  |

## 