Software Requirements Specification

For

Course Management

Prepared by

Aditya Gupta(21BCS007)

Adarsh Kumar(21BCS006)

Advait Manke(21BCS010)

K.L.Praseeda Keerthi(21BCS119)

Mahanthi Pranitha(21BCS124)

Student Mentor:

Vedant Bande(21BCS238)

Table of Contents

Table of contents

1. Introduction

    1.1 Introduction about the Fusion

    1.2 Purpose of the module

    1.3 Scope of the module

2. User/Actor Description(characteristics)

3. Functional Requirements

    3.1 Use case Diagram

    3.2 Use case description

    3.3 Other Functional Requirements

    3.4 Other Constraints

        3.4.1 User Interfaces

        3.4.2 Software (Tech) Stack Used

4. **Non-Functional Requirements**

5. **Module dependencies with other fusion modules**

    5.1. UI Level

    5.2 DB Level Dependencies

    5.3. Module Level Dependencies

1. Introduction

1.1 Introduction about the Fusion – A brief description:

FusionIIIT is a **student-driven initiative** that aims to provide a **holistic solution** for the **seamless integration and automation of diverse functions** within PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur . It is crafted with precision using **Flutter** and powered by the **Django Web framework** .

FusionIIIT encompasses everything from efficient administration management to academic prowess and miscellaneous departmental tasks. It is like a digital wizard that takes care of everything, from organizing the administrative stuff to making academics smoother. It jumps into various departments and sections, making sure every corner of campus life runs smoothly.

In the admin side, it handles the complicated paperwork and processes. For academics, it brings a digital touch, making learning and managing courses easier. But it doesn't stop there; FusionIIIT is like a friendly companion for all the different parts of the campus, making sure everything works well.

In simpler terms, FusionIIIT is not just a tool – it's a helpful friend, making life at PDPM IIITDM Jabalpur more organized and enjoyable for everyone

1.2 Purpose of the module:

The purpose of the module course management is to provide details for instructors to upload course-related materials, assign tasks, and make announcements. The application also assesses the assignments and projects assigned to students and recommends optimal grades for each student. Students can view their performance, course content, submit assignments, and ask questions.

1.3 Scope of the module:

The product scope of this software is to provide a user-friendly and intuitive platform for educational institutions. Lecturers can assign various tasks and view student participation both individually and for the entire course.

2.User/Actor Description(characteristics)

2.1 Student:

The student actor in the course management module represents individuals who are enrolled in a course and intend to participate in the learning process by engaging with the course content, completing assignments, and taking quizzes. The course management module provides a digital platform for students to access course materials and interact with their instructors and peers.

Role: The student's role is to start learning by logging in, accessing course materials, submitting assignments, taking quizzes, and engaging with course content in the digital realm.

Specific Functionalities:

• Access course materials.

• Submit assignments.

• View performance in previous Assignments.

• View course content and details.

• Download content.

2.2 Instructor:

The instructor in the course management module represents individuals who are responsible for teaching a course and intend to manage the learning process by engaging with the course content, assigning assignments, and creating quizzes. The course management module provides a digital platform for lecturers to upload course materials, interact with students, and evaluate their performance.

Role: The instructor, upon logging in, manages courses by uploading materials, creating quizzes, evaluating student work, and reviewing performance data.

Specific Functionalities:

• Upload Content

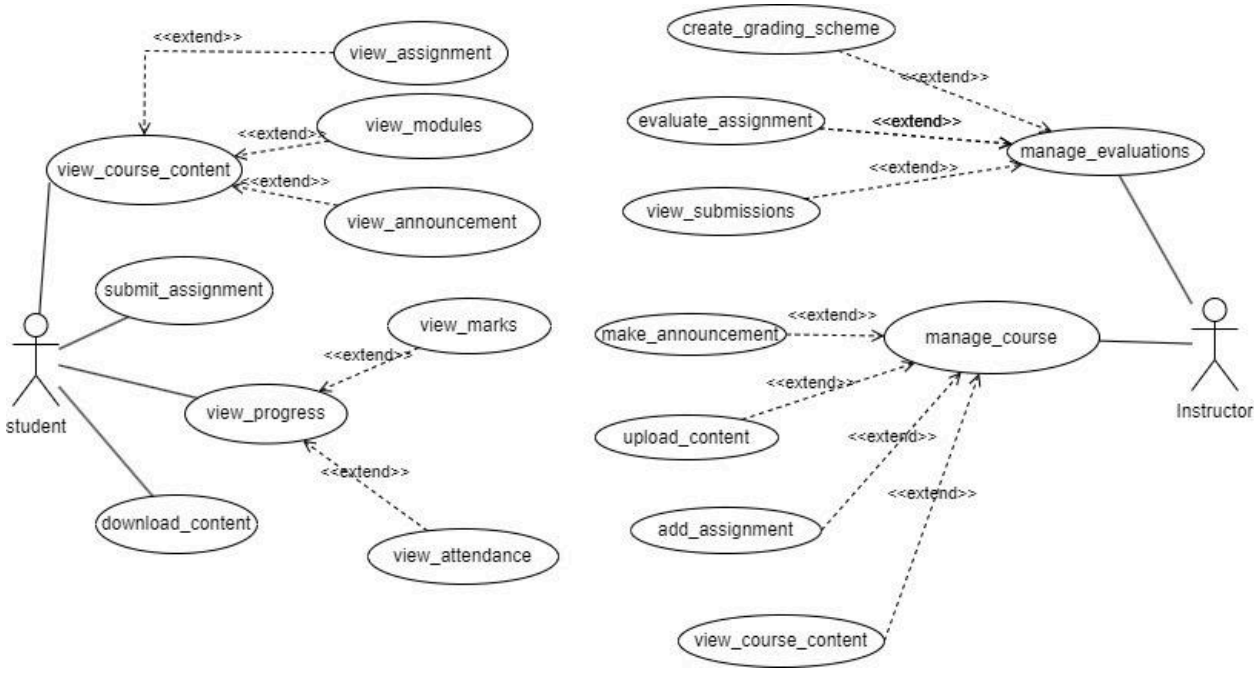
• Add Assignments

• Evaluate Assignments

• View performances

3. Functional Requirements

3.1 Use Case Diagram



3.2 Use Case Description

3.2.1 View Course Content

UC ID	UC#1	
Use case Name	View Course Content	
Description	The "View Course Content" use case allows both instructors and students to view the content of a course, including videos, lectures, readings, and assignments.	
Actor	Student	
Precondition	1	The student must be logged-in .
	2	Student should have registered in that particular course.

Main Flow	1	The user logs in to the system.
	2	The user navigates to the "Courses" tab.
	3	The user selects the desired course from the list of available courses.
	4	The system displays the course content, including: Videos Lectures Readings Assignments Any other relevant course materials
	5	The user can view and interact with the course content as permitted by their role (e.g., students may be able to download materials, while instructors may have additional editing or management capabilities).
Post conditions	The user has successfully viewed the course content. The system has updated any necessary logs or records (e.g., tracking user activity, recording content access).	
Alternate Flow	AF1	User is not registered for the course: If the user attempts to view content for a course they are not registered in, the system displays an error message indicating that they are not enrolled.
	AF2	Technical error: If a technical error occurs during content retrieval or display, the system displays an appropriate error message and logs the incident for troubleshooting.
Sub Flow		NIL
Global Alternate Flow	GA1	User cancels the procedure: The user can cancel the content viewing process at any time and return to the previous screen or dashboard.
	GA2	System error: If a general system error occurs (e.g., database failure, server issues), the system displays an error message and logs the incident for further investigation.

3.2.2 Submit Assignment

UC ID	UC#2
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.

3.2.3 View Progress

UC ID	UC#3	
Use case Name	View Porgess	
Description	The "View Progress" use case allows a student to view their progress in a course, including their marks and attendance.	
Actor	Student	
Precondition	1	The student must be logged-in .

	2	Student should have registered in that particular course.
	3	Marks and attendance data for the course should be uploaded by the instructor.
Main Flow	1	The student navigates to the "Courses" tab.
	2	The system displays a list of courses the student is registered in.
	3	The student selects the desired course.
	4	The student clicks on the "View Progress" option.
	5	The system retrieves and displays the student's progress information, including: Marks for each assignment and exam Overall course grade (if calculated) Attendance record (dates present, absent, or excused)
	6	The student can view and analyze their progress information.
Post conditions	The student has successfully viewed their progress in the course. The system has logged the student's access to their progress information.	
Alternate Flow	A1	Marks or attendance data not available: If marks or attendance data have not yet been uploaded by the instructor, the system displays a message indicating that the information is not yet available.
	A2	Technical error: If a technical error occurs while retrieving or displaying progress data, the system displays an error message and logs the incident.
Sub Flow	S1	View detailed marks: The student can click on a specific assignment or exam to view more detailed information about their marks, such as: Breakdown of marks by criteria Instructor feedback (if available)
	S2	View attendance details: The student can click on an attendance record to view more specific details, such as: Reason for absence (if provided) Approval status for excused absences
Global Alternate Flow	GA1	Student cancels the procedure: The student can cancel the progress viewing process at any time and return to the previous screen.
	GA2	System error: If a general system error occurs, the system displays an error message and logs the incident. The student's work is preserved, and they can retry viewing their progress later.

3.2.4 Download Content

UC ID	UC#4	
Use case Name	download_content	
Description	The "Download Content" use case allows a student to download course content materials for a registered course.	
Actor	Student	
Precondition	1	The student must be logged-in .
	2	The course content files must have been uploaded by the instructor.
Main Flow	1	The student navigates to the desired course.
	2	The student clicks on the "Content" tab.
	3	The system displays a list of available content files for the course.
	4	The student selects the content file(s) they want to download.
	5	The student clicks the "Download" button.
	6	The system initiates the download process. The student's browser or download manager handles the file transfer. The system displays a progress bar or notification indicating the download status.
	7	Upon completion, the downloaded files are saved to the student's designated download location.
Post conditions	The selected content files have been successfully downloaded to the student's device.	
Alternate Flow	A1	File not found: If the selected content file is not found or has been removed, the system displays an error message indicating that the file is unavailable.
	A2	File corrupted: If the system detects that a file is corrupted or invalid during the download process, it displays an error message and prevents the download.
	A3	Insufficient storage: If the student's device does not have enough storage space to accommodate the download, the system displays a warning message and prompts them to free up space or cancel the download.
Sub Flow		NIL
Global Alternate	GA1	Student cancels the download: The student can cancel the download process at any time before completion. The system interrupts the download and returns to the

Flow		content list.
	GA2	System error: If a general system error occurs during the download, the system displays an error message and logs the incident. The student can retry the download later.
	GA3	Network interruption: If the network connection is lost or interrupted during the download, the system pauses the download and prompts the student to resume once connectivity is restored.

3.2.5 Manage Courses

UC ID	UC#5	
Use case Name	manage_courses	
Description	The "Manage Course" use case empowers instructors to administer various aspects of their courses, including adding assignments, managing content, making announcements, and more.	
Actor	Instructor	
Precondition	1	The instructor must be logged in to the system.
	2	The course to be managed must exist within the system. If not, the instructor must create it first.
Main Flow	1	The instructor navigates to the "Courses" tab.
	2	The instructor selects the "Manage Course" option.
	3	The system presents a list of available courses assigned to the instructor.
	4	The instructor selects the desired course to manage.
	5	The system displays the course management interface, offering options such as: Add assignments View content Upload content Make announcements
	6	The instructor interacts with the interface to perform the desired course management tasks.
Post conditions	The instructor has successfully executed the intended course management actions. The system has updated its records to reflect the changes made.	

Alternate Flow	A1	Course not found: If the instructor attempts to manage a course that does not exist, the system presents a message indicating the course's absence and prompts the instructor to create it.
Sub Flow		NIL
Global Alternate Flow	GA1	Instructor cancels the procedure: The instructor can terminate the course management process at any point before completing actions. The system returns to the course management interface or the previous screen.

3.2.6 Manage Evaluation

UC ID	UC#6	
Use case Name	manage _evaluation	
Description	The "Manage Evaluations" use case allows instructors to fully administer the assessment process for their courses. They can create and customize grading schemes, evaluate student assignment and manage submissions.	
Actor	Instructor	
Precondition	1	The instructor must be logged-in .
	2	The course for which evaluations are to be managed must exist within the system.
	3	Students must have submitted assignments or completed assessments relevant to the evaluations.
Main Flow	1	The instructor navigates to the "Courses" tab.
	2	The instructor selects the desired course and clicks on the "Manage Evaluations" option.
	3	The system displays the course evaluation interface, offering functionalities such as: Create grading schemes Evaluate assignments View submissions
Post conditions	The instructor has successfully completed the desired evaluation tasks for their course.	
Alternate Flow		NIL
Sub Flow		NIL

Global Alternate Flow		NIL

3.4.1 User Interfaces

Flutter framework: UI development must adhere to Flutter's guidelines and best practices.

Cross-platform compatibility: UI elements must render consistently across Android, iOS, and potentially web.

State management: Efficient management of UI state and data flow using appropriate Flutter techniques (e.g., Provider, BLoC, MobX).

Navigation: Clear and intuitive navigation patterns tailored for mobile experiences.

Device-specific features: Leverage device features like cameras, GPS, or sensors as needed for specific use cases.

3.4.2 Software (Tech) Stack Used

Frontend: Flutter (Dart programming language)

Backend: Django (Python), PostgreSQL

API communication: RESTful APIs for data exchange between frontend and backend.

Libraries and frameworks: Potential use of additional libraries for:

Authentication (e.g., Django REST framework authentication)

3.4.3 Business Rules

Enrollment deadlines (enforced within Django backend logic)

Assignment deadlines (enforced within Django backend logic)

Grading policies (implemented within grading and feedback functionalities)

Attendance tracking (stored and managed in the PostgreSQL database)

Communication channels (integrated into app or provided as external links)

User roles and permissions (handled by Django's authentication and authorization system)

4. Non-Functional Requirements

4.1 Performance

The system ensures fast page loading, rapid response to user interactions, and scalability to accommodate increased users and courses without performance slowdown.

4.2 Scalability

The system should scale effortlessly to handle an increasing number of users and courses without compromising performance.

4.3 Availability

The system ensures uninterrupted accessibility 24/7, with regular backups for student data and materials, along with established disaster recovery plans for major failures.

4.4 Security

The system prioritizes robust user authentication, strong password policies, encrypted sensitive data, frequent security updates, and strict adherence to data privacy regulations.

4.5 Usability

The system boasts an intuitive interface for easy learning, provides clear instructions with helpful guidance, and offers accessible online help, tutorials, and support resources for both students and instructors.

5. Module dependencies with other fusion modules

5.1. UI Level

-Not dependent on other modules

5.2 DB Level Dependencies

-Dependent on course list, student list.

5.3. Module Level Dependencies

-Dependent on The Program and Curriculum module for providing the approved list of courses