Contents

[Use Case Description: 1](#_Toc185358960)

[1. Create Notes by Students 1](#_Toc185358961)

[2. Edit Notes by student 2](#_Toc185358962)

[3. Delete Notes by Student 3](#_Toc185358963)

[4. Create lectures by teacher 5](#_Toc185358964)

[5. Create Assessments 7](#_Toc185358965)

[6. Evaluate Assessment 8](#_Toc185358966)

[7. Access Recommended resources 10](#_Toc185358967)

[8. Save Recommended Resources 11](#_Toc185358968)

[9. Mark Attendance by Teacher 13](#_Toc185358969)

[10. Enroll Students in the Database by Admin 14](#_Toc185358970)

# Use Case Description:

## 1. Create Notes by Students

|  |  |
| --- | --- |
| **Use Case ID** | **UC-1** |
| **Use Case Name** | Create Notes |
| **Actors** | **Primary Actor**: Student  **Secondary Actors**: LMS System |
| **Description** | A student logs into the LMS, selects a specific lecture, and creates notes using the note-taking tool available on the lecture screen. The system saves the notes for future reference. |
| **Trigger** | The student clicks the "Create Notes" button on a specific lecture screen. |
| **Preconditions** | 1. The student must be logged into the LMS.  2. The student must be enrolled in the relevant course. |
| **Postconditions** | 1. Notes are saved successfully in the LMS.  2. A confirmation message is displayed. |
| **Normal Flow** | **1.0 Create Notes**  1. The student logs into the LMS.  2. The student clicks on the "Courses" tab.  3. The student selects a specific lecture.  4. The student clicks on the "Create Notes" button.  5. The system displays the notes editor.  6. The student inputs their notes into the editor.  7. The student clicks the "Save Notes" button.  8. The system saves the notes and displays a confirmation message: "Notes saved successfully." |
| **Alternative Flows** | **1.1 Lecture Not Selected** |
|  | 1. If the student does not select a lecture, they are prompted to select one before proceeding. |
|  | Return to step 3 of the normal flow. |
| **Exceptions** | **1.0.E1 System Error When Saving Notes** |
|  | 1. The system displays an error message: "Unable to save notes. Please try again." |
|  | 2. The student is prompted to retry saving the notes. |
|  | **1.0.E2 Invalid Session** |
|  | 1. If the student’s session expires, the system prompts the student to log in again. |
|  | Return to step 1 of the normal flow. |
| **Business Rules** | BR-1: Notes must not exceed 5,000 characters. |
|  | BR-2: Only enrolled students can access the note-taking feature. |
| **Assumptions** | 1. Students will use this feature regularly for study purposes. |
|  | 2. The system will provide reliable auto-save functionality in future iterations. |

## 2. Edit Notes by student

|  |  |
| --- | --- |
| **Use Case ID** | **UC-2** |
| **Use Case Name** | Edit Notes |
| **Actors** | **Primary Actor:** Student |
|  | **Secondary Actor:** LMS System |
| **Description** | This Use case describes how a student edits previously created notes within the Learning Management System (LMS). The student can modify the content of notes to correct errors, update information, or enhance details. The system ensures all changes are saved properly and reflected immediately. |
| **Trigger** | The student selects the "Edit Notes" option from their notes list or directly from the lecture screen. |
| **Preconditions** | 1. The student is logged into the LMS.  2. The student has existing notes saved. |
| **Postconditions** | 1. The edited notes are saved and updated in the system.  2. A confirmation message is displayed indicating successful saving. |
| **Normal Flow** | 1. Student logs into the LMS.  2. Student navigates to the notes section or a specific lecture.  3. Student selects the note to be edited.  4. The system displays the note in an editable text editor.  5. Student makes changes and clicks the "Save" button.  6. System saves the updated notes and displays a confirmation message. |
| **Alternative Flows** | **AF-1: Notes Not Found**  1. If the selected note is missing or deleted, the system displays an error message: "Note not found."  2. Return to the notes list. |
| **Exceptions** | **E-1: Connection Error**  1. If there is a network issue during saving, the system shows an error: "Unable to save. Please check your connection."  2. The system prompts the student to retry saving. |
| **Business Rules** | \_ |
| **Assumptions** | 1. Students frequently update notes to refine understanding.  2. All notes are auto-saved periodically to prevent data loss. |

## 3. Delete Notes by Student

|  |  |
| --- | --- |
| **Use Case ID** | **UC-3** |
| **Use Case Name** | Delete Notes |
| **Actors** | **Primary Actor:** Student |
|  | **Secondary Actor:** LMS System |
| **Description** | This Use case describes how a student deletes previously created notes within the Learning Management System (LMS). The student can remove unnecessary or outdated notes. The system ensures that once deleted, the note is completely removed from the system and no longer visible to the student. |
| **Trigger** | The student selects the "Delete Notes" option for a specific note from their notes list or lecture screen. |
| **Preconditions** | 1. The student is logged into the LMS.  2. The student has existing notes to delete. |
| **Postconditions** | 1. The selected note is permanently deleted from the system.  2. The note is no longer visible to the students in their notes section. |
| **Normal Flow** | 1. Student logs into the LMS.  2. Student navigates to the notes section or a specific lecture.  3. Student selects the note they wish to delete.  4. The system asks for confirmation: "Are you sure you want to delete this note?"  5. Student confirms deletion.  6. The system permanently deletes the note and displays a confirmation message: "Note deleted successfully." |
| **Alternative Flows** | **AF-1: Cancel Deletion**  1. If the student chooses to cancel the deletion, the system returns to the notes list without making any changes. |
| **Exceptions** | **E-1: Note Not Found**  1. If the selected note does not exist or has already been deleted, the system shows an error: "Note not found."  2. Return to the notes list. |
|  | **E-2: Connection Error**  1. If there is a network issue during deletion, the system shows an error: "Unable to delete. Please check your connection."  2. The system prompts the student to retry deletion. |
| **Business Rules** | \_ |
| **Assumptions** | 1. Students delete notes when they no longer find them useful or relevant.  2. The LMS does not allow recovery of deleted notes once confirmed. |

## 4. Create lectures by teacher

**Use Case: Create a Lecture Manually**

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | **Use Case ID** | | |  | | --- | | UC-4 | |
| |  | | --- | | **Use Case Name** | | |  | | --- | | Create Lecture Manually | |
| |  |  | | --- | --- | | **Actors** |  | | |  | | --- | | Teacher | |
| |  |  | | --- | --- | | **Description** |  | | |  | | --- | | A teacher creates a lecture manually by uploading or entering lecture content in an AI-assisted LMS. | |
| |  |  | | --- | --- | | **Trigger** |  | | |  |  | | --- | --- | |  | The teacher clicks on the "Create Manually" button after navigating to the "Create Lecture" section of a course. | |
| |  | | --- | | **Preconditions** | | |  |  | | --- | --- | |  | The teacher must be logged into the LMS. The teacher must have access to the Courses tab in the dashboard. | |
| |  | | --- | | **Postconditions** | | |  |  | | --- | --- | |  | - POST-1: If the teacher presses "Save," the lecture appears under the "Saved" tab and is only visible to the teacher. - POST-2: If the teacher presses "Save and Publish," the lecture appears under the "Published" tab and is visible. | |
| |  | | --- | | **Normal Flow** | | |  |  | | --- | --- | |  | 1. The teacher logs into the LMS. 2. The teacher navigates to the "Courses" tab on the dashboard. 3. The LMS displays all available courses. 4. The teacher selects a course by clicking "Go to Course." 5. The teacher clicks "Create Lecture."  6. A form appears, and the teacher fills out fields like **Title**, **Description**, **duration, Chapter, Academic Level, Learning Outcomes, Preferred Formats, Attachments. Objective**, and **Reference**. 7. The teacher adds modules and enters Generate Content. 8. The teacher selects either "Save" (to appear in the "Saved" tab) or "Save and Publish" (to appear in the "Published" tab). 9. The system confirms the action. | |
| |  | | --- | | Alternative Flows | | |  |  | | --- | --- | |  | **AF-1: Field Validation Error** 1. The teacher submits the form with invalid or incomplete data. 2. The system highlights the errors and prompts corrections. | |
| |  |  | | --- | --- | | Exceptions |  | | |  |  | | --- | --- | |  | **EX-1: System Error** 1. The system fails to save or publish due to a technical issue. 2. The system displays an error message and advises retrying. | |

**Use Case: Create a Lecture Using AI**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  | | --- | | **Use Case ID** | | |  |  | | --- | --- | |  | UC-4 | |
| |  |  | | --- | --- | | **Use Case Name** |  | | |  | | --- | | Create Lecture with AI | |
| |  | | --- | | **Actors** | | |  |  | | --- | --- | |  | Teacher | |
| |  | | --- | | **Description** | | |  |  | | --- | --- | |  | A teacher generates lecture content using the AI feature of the LMS and decides whether to save it for later or publish it immediately. | |
| |  | | --- | | **Trigger** | | |  |  | | --- | --- | |  | The teacher clicks on the "Create with AI" button after navigating to the "Create Lecture" section of a course. | |
| |  | | --- | | **Preconditions** | | |  |  | | --- | --- | |  | 1. The teacher must be logged into the LMS. 2. The teacher must have access to the Courses tab in the dashboard. | |
| |  | | --- | | **Postconditions** | | |  | | --- | | 1. If the teacher presses "Save," the lecture appears under the "Saved" tab and is only visible to the teacher. 2. If the teacher presses "Save and Publish," the lecture appears under the "Published" tab and is visible. | |
| |  | | --- | | **Normal Flow** | | |  |  | | --- | --- | |  | 1. The teacher logs into the LMS. 2. The teacher navigates to the "Courses" tab on the dashboard. 3. The LMS displays all available courses. 4. The teacher selects a course by clicking "Go to Course." 5. The teacher clicks "Create Lecture." 6. A generic form appears, and the teacher fills out fields like **Title**, **Description**, **Learning Objective**, and **Reference**. 7. The teacher presses "Create with AI," and the system sends a request to an AI model. 8. The system generates and displays lecture content for review. 9. The teacher can save it in the "Saved" tab or save and publish it to the "Published" tab. 10. The system confirms the action. | |
| |  | | --- | | Alternative Flows | | |  | | --- | | **AF-1: AI Content Adjustment** 1. The teacher reviews the AI-generated content and edits it if necessary before saving or publishing. | |
| |  | | --- | | Exceptions | | |  | | --- | | **EX-1: AI Generation Error** 1. The AI fails to generate content due to connectivity or technical issues. 2. The system displays an error message and suggests retrying or switching to manual mode. | |

## 5. Create Assessments

|  |  |
| --- | --- |
| **Use Case ID** | **UC-5** |
| **Use Case Name** | Create Assessments (Assignments + Quizzes) |
| **Actors** | **Primary Actor:** Teacher  **Secondary Actor:** Standard Evaluation Module (System) |
| **Description** | The teacher creates and schedules assessments (assignments or quizzes) using the Standard Evaluation Module. The system saves the assessment, applies predefined rubrics, and schedules it for a specified date and time. |
| **Trigger** | The teacher clicks on the "Create Assessment" button in the system. |
| **Preconditions** | 1. The teacher must be logged into the system.  2. The teacher must have appropriate permissions to create assessments. |
| **Postconditions** | 1. The assessment is saved in the system.  2. The scheduled date and time for the assessment are recorded.  3. A confirmation message is displayed to the teacher. |
| **Normal Flow** | 1. The teacher logs into the system and navigates to the "Assessments" tab. 2. The teacher clicks the "Create Assessment" button. 3. The system prompts the teacher to select the assessment type: **Assignment** or **Quiz**. 4. The teacher enters the following assessment details: Assessment title, Description/instructions, Assessment type (Assignment/Quiz), Total marks, Scheduled date and time 5. If **Quiz** is selected:    1. The teacher creates quiz questions (e.g., MCQs, short answers) using the question editor.    2. The teacher defines correct answers and assigns marks per question.    3. The teacher reviews the quiz structure. 6. If **Assignment** is selected:    1. The teacher uploads assignment instructions (PDF, DOCX, etc.) or types instructions in the editor.    2. The teacher attaches any additional resources or templates if needed. 7. The teacher applies or customizes the grading rubric. 8. The teacher reviews all entered details and clicks the "Save and Schedule" button. 9. The system saves the assessment and displays a confirmation message: **"Assessment scheduled successfully."** |
| **Alternative Flow** | **1.1 Invalid Date or Time Selected**   1. If the teacher enters an invalid date or time (e.g., past date), the system displays an error message: **"Invalid date. Please select a future date and time."** 2. The teacher re-enters a valid date/time and resumes the flow from Step 4.   **1.2 Missing Mandatory Fields**   1. If any mandatory details (e.g., title, type) are missing, the system highlights the incomplete fields and prompts the teacher to fill them in. |
| **Exceptions** | **1.0.E1 System Error While Saving Assessment**   1. If the system encounters an error while saving, an error message is displayed: **"Unable to save assessment. Please try again."** 2. The teacher retries saving the assessment. |
| **Business Rules** | * BR-1: All assessments must have a title, type, and scheduled date/time. * BR-2: Quizzes must include at least one question. * BR-3: Assignments must include instructions or attached resources. |
| **Assumptions** | * Teachers will create assessments within the defined course timelines. * Internet connectivity is stable during the creation process. |

## 6. Evaluate Assessment

|  |  |
| --- | --- |
| **Use Case ID** | **UC-6** |
| **Use Case Name** | Evaluate assessment |
| **Actors** | **Primary Actor:** Teacher  **Secondary Actor:** Standard Evaluation Module (System) |
| **Description** | The teacher evaluates student submissions for assignments or quizzes. The system supports AI-assisted grading and allows manual adjustments to scores and feedback before finalizing and publishing results. |
| **Trigger** | The teacher accesses submitted assessments for evaluation. |
| **Preconditions** | 1. Students have submitted their assignments/quizzes.  2. The teacher is logged into the system and has access to the evaluation module. |
| **Postconditions** | 1. Scores and feedback are recorded and saved.  2. The results are available for publishing to students. |
| **Normal Flow** | 1. The teacher logs into the system and navigates to the "Pending Evaluations" section. 2. The system displays a list of submitted assignments/quizzes. 3. The teacher selects a specific assessment to evaluate. 4. The system performs AI-assisted grading (if enabled) based on predefined rubrics and displays the scores. 5. The teacher reviews the AI-assigned scores and feedback. 6. The teacher performs any of the following actions: Adjust scores manually, Edit or add detailed feedback for each student. 7. The teacher clicks the **"Save Evaluation"** button. 8. The system saves updated scores and feedback. 9. The teacher clicks **"Publish Results"** to make scores and feedback visible to students. 10. The system displays a confirmation message: **"Results published successfully."** |
| **Alternative Flow** | **2.1 No Submissions Available**   1. If no submissions are available, the system displays a message: **"No submissions found for evaluation."**   **2.2 Incomplete AI Grading**   1. If AI grading fails for any question or response, the system notifies the teacher: **"AI grading incomplete. Manual evaluation required."** 2. The teacher proceeds to manually grade the affected questions.   **2.3 Partial Evaluation**   1. If the teacher saves the evaluation without publishing, the system marks the submissions as **"Partially Evaluated.** |
| **Exceptions** | **2.0.E1 System Error During Grading**   1. If an error occurs during AI-assisted grading, the system displays: **"Error occurred during grading. Please grade manually."** |
| **Business Rules** | BR-1: Scores must be saved before publishing results.  BR-2: Teachers can override AI-assigned scores and feedback.  BR-3: Final scores and feedback cannot be edited after publishing. |
| **Assumptions** | 1. Teachers will cross-check AI-assigned scores for accuracy. 2. Students expect timely results after submission deadlines. |

## 7. Access Recommended resources

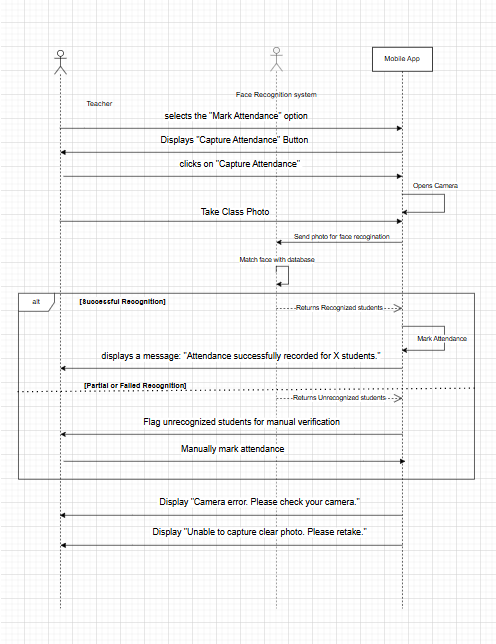
|  |  |
| --- | --- |
| **Use Case ID** | **UC-7** |
| **Use Case Name** | Access Recommended Resources |
| **Actors** | **Primary Actor:** Student  **Secondary Actor:** Content Recommendation Module (System) |
| **Description** | The student accesses the system to view personalized recommendations for study materials based on their individual performance, preferences, and progress in the course. |
| **Trigger** | The student logs into the system and navigates to the "Recommended Resources" section. |
| **Preconditions** | 1. The student must be logged into the system.  2. The student must have completed some assessments or have progress tracked in the system. |
| **Postconditions** | 1. The student can view a list of personalized study materials.  2. The student can access the recommended resources. |
| **Normal Flow** | 1. The student logs into the system. 2. The student navigates to the "Recommended Resources" section, either from the dashboard or the course-specific page. 3. The system analyzes the student's performance, preferences, and progress to generate a list of personalized recommendations. 4. The system presents the recommended resources (e.g., book, videos, quizzes, assignments) based on the student's current learning progress and weak areas. 5. The student reviews the recommendations, which may include explanations, additional reading materials, or practice exercises. 6. The student selects a resource to access (e.g., clicks on a video link, opens a practice quiz). 7. The system directs the student to the selected resource or content. |
| **Alternative Flow** | **1.1 No Recommendations Available**   1. If the system cannot generate recommendations due to insufficient data or performance tracking, it displays a message: **"No recommendations available at this time. Please complete more assessments."**   **1.2 No Internet Connection**   1. If the student experiences an internet connectivity issue, the system prompts: **"Unable to access resources due to connection issues. Please try again later."** |
| **Exceptions** | **1.0.E1 System Error While Accessing Resources**   1. If there is an error accessing the content, the system displays an error message: **"Unable to access the requested resource. Please try again."** 2. The student may retry accessing the resource. |
| **Business Rules** | BR-1: Recommended resources must align with the student's current course syllabus and performance data.  BR-2: The system may only recommend resources that are part of the officially approved content library. |
| **Assumptions** | 1. The system continuously tracks and updates the student's performance data. 2. Recommended resources will always be available to students as long as they have internet access. |

## 8. Save Recommended Resources

|  |  |
| --- | --- |
| **Use Case ID** | **UC-8** |
| **Use Case Name** | Save Recommended Resources |
| **Actors** | **Primary Actor:** Student  **Secondary Actor:** Content Recommendation Module (System) |
| **Description** | The student saves recommended study resources for later review, allowing them to bookmark content they find helpful or wish to return to. |
| **Trigger** | The student clicks on the "Save for Later" button next to a recommended resource. |
| **Preconditions** | 1. The student must be logged into the system.  2. The student must have accessed recommended resources. |
| **Postconditions** | 1. The resource is saved in the student's "Saved Resources" section.  2. The student can access the saved content at any time. |
| **Normal Flow** | 1. The student logs into the system and accesses the recommended resources section. 2. The student reviews the available recommendations and finds a resource they wish to save. 3. The student clicks on the "Save for Later" button next to the resource. 4. The system saves the resource to the student's personalized "Saved Resources" section. 5. The system displays a confirmation message: **"Resource saved successfully."** 6. The student can later navigate to their "Saved Resources" section to view and access all saved content. |
| **Alternative Flow** | **2.1 Resource Already Saved**   1. If the student tries to save a resource that has already been saved, the system displays a message: **"This resource is already saved in your list."** 2. The student may choose to either open the saved resource or continue browsing new recommendations.   **2.2 Insufficient Data to Save**   1. If the system fails to save the resource due to a technical issue, it displays: **"Unable to save resource at this time. Please try again later."** 2. The student can retry saving the resource after addressing the issue. |
| **Exceptions** | **2.0.E1 System Error During Save Operation**   1. If an error occurs while saving the resource, an error message is displayed: **"Error occurred while saving. Please try again later."**   The student retries saving the resource or contacts support for assistance |
| **Business Rules** | BR-1: Resources can only be saved if they are part of the official recommended list.  BR-2: Saved resources are accessible until the student marks them as "Completed" or removes them manually. |
| **Assumptions** | 1. The student may save multiple resources for later review. 2. Saved resources will not expire unless the course or content is deleted by the administrator. |

## 9. Mark Attendance by Teacher

|  |  |
| --- | --- |
| **Use Case ID** | **UC-9** |
| **Use Case Name** | Mark Attendance by Teacher |
| **Actors** | **Primary Actor:** Teacher  **Secondary Actor:** Face Recognition System (System) |
| **Description** | The teacher uses the mobile app to mark attendance by taking a photo of the class, utilizing face recognition technology to match students' faces to the database and automatically mark their attendance. |
| **Trigger** | The teacher opens the attendance marking feature on the mobile app and clicks on the "Capture Attendance" button. |
| **Preconditions** | 1. The teacher must be logged into the system.  2. The class roster must be populated in the system.  3. The students' faces must be registered in the system's database. |
| **Postconditions** | 1. Attendance is automatically marked for students whose faces are recognized.  2. A confirmation message is displayed to the teacher indicating the attendance status. |
| **Normal Flow** | 1. The teacher logs into the system.  2. The teacher selects the "Mark Attendance" option.  3. The teacher clicks on "Capture Attendance" to open the phone's camera.  4. The system detects students' faces and matches them with the stored data.  5. The system marks attendance automatically for recognized students.  6. The system displays a confirmation message: "Attendance successfully recorded for X students." |
| **Alternative Flow 1** | 1. If the system fails to recognize a student's face, it flags the student for manual verification.  2. The teacher manually marks attendance for the flagged student. |
| **Alternative Flow 2** | 1. If the teacher cannot take a clear photo (due to low lighting, angle, or obstruction), the system prompts them to retake the picture. |
| **Exceptions** | 1.0.E1 **Face Recognition Failure:** If the system cannot match a student's face, an error message is displayed: "Unable to recognize face. Please try again."  1.0.E2 **Camera Error:** If the phone camera cannot be accessed, the system prompts the teacher to check the camera. |
| **Business Rules** | 1. Students must be enrolled in the system with their face data registered beforehand.  2. The system must provide an option for manual attendance adjustment in case of face recognition failure. |
| **Assumptions** | 1. The face recognition database is regularly updated to reflect new student enrollments or changes.  2. Students must be in view of the camera for proper recognition. |



## 10. Enroll Students in the Database by Admin

|  |  |
| --- | --- |
| **Use Case ID** | **UC-10** |
| **Use Case Name** | Enroll Students in Database by Admin |
| **Actors** | **Primary Actor:** Admin  **Secondary Actor:** Student Database (System) |
| **Description** | The admin enrolls students into the system by entering their information and uploading face images for future face recognition-based attendance. |
| **Trigger** | The admin logs into the admin panel and selects the "Enroll Student" option. |
| **Preconditions** | 1. The admin must be logged into the system with appropriate permissions.  2. The student data (name, email, course, etc.) and face images are available for enrollment. |
| **Postconditions** | 1. The student’s information is added to the system.  2. The student’s face image is stored in the database for future face recognition. |
| **Normal Flow** | 1. The admin logs into the system.  2. The admin selects the "Enroll Student" option.  3. The admin enters the student’s personal details (e.g., name, email, course).  4. The admin uploads a photo of the student’s face for future attendance marking.  5. The system confirms that the student has been successfully enrolled.  6. The system stores the student’s face data and personal information. |
| **Alternative Flow** | 1. If the admin fails to upload the face photo, the system prompts the admin to upload a clear image. |
| **Exceptions** | 1.0.E1 **Incomplete Data:** If the admin fails to provide required student information (e.g., missing name or course), the system displays an error: "Please provide all required details."  1.0.E2 **Image Upload Error:** If the image upload fails, the system prompts the admin to retry. |
| **Business Rules** | 1. The admin must have sufficient privileges to enroll students.  2. The student’s face image should meet quality standards (e.g., clear and frontal view). |
| **Assumptions** | 1. The student’s face image is used only for attendance purposes and is stored securely.  2. Students will be enrolled at the beginning of each academic term or course. |