**COMSATS University Islamabad,**

**Abbottabad Campus**

**SOFTWARE REQUIREMENTS SPECIFICATION   
(SRS DOCUMENT)**

**for**

**AI-Assisted Learning Management System**  
Version 0.1

***By***

**Hanzla Nouman CIIT/FA21-BSE-015/ATD**

**Laiba Binta Tahir CIIT/FA21-BSE-019/ATD**

**Arfah Ali CIIT/FA21-BSE-080/ATD**

***Supervisor*Mr. Mukhtiar Zamin**

***Bachelor of Science in Computer Science (2021-2025)***

**Table of Contents**

**Revision History 3**

**1.** **Introduction 5**

1.1 Purpose 5

1.2 Scope 5

**2.** **Overall description 5**

2.1 Product perspective 5

2.2 Operating environment 5

2.3 Design and implementation constraints 5

**3.** **Requirement identifying technique 6**

3.1 Use Case Diagram 6

3.2 Use Case Description 6

**4.** **Specific Requirements 8**

4.1 System feature X 8

**5.** **Quality attributes**

5.1 Usability 9

5.2 Performance 9

**6.** **External interface requirements 35**

6.1 User interfaces 35

6.2 Software interfaces 35

6.3 Hardware interfaces 35

6.4 Communications interfaces 35

**7.** **Project Gantt chart 35**

**8.** **References 10**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

**Application Evaluation History**

|  |  |
| --- | --- |
| **Comments (by committee)**  **\*include the ones given at scope time both in doc and presentation** | **Action Taken** |
|  |  |
|  |  |

**Supervised by**

**<Supervisor’s Name>**

Signature\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Introduction**

This document presents the Software Requirements Specification (SRS) for the AI-Assisted Learning Management System (LMS). It provides an organized and structured overview of the system, its purpose, scope, and features. The document serves as a foundation for all stakeholders, including students, teachers, administrators, and developers, to understand the system requirements and design goals.

The SRS will guide the development, implementation, and validation processes to ensure the system meets user needs and expectations.

**Purpose**

The purpose of this document is to outline the functional and non-functional requirements for the AI-Assisted LMS. This software will be developed to assist educational institutions in managing teaching, learning, and evaluation processes efficiently. The primary users of this system include students, teachers, and administrators.

By leveraging AI tools, the system aims to enhance content delivery, automate administrative tasks, personalize learning experiences, and streamline evaluations. This document ensures that the requirements are clear, measurable, and feasible for implementation.

**Scope**

The AI-Assisted LMS is an advanced learning platform designed to improve the educational process by integrating artificial intelligence into content generation, evaluation, and administrative functions. It supports the following key objectives:

* Personalized Learning: Providing AI-driven content recommendations and interactive materials tailored to individual students’ needs.
* Automated Evaluations: Enabling self-evaluation tools and standard assessment processes with instant feedback.
* Performance Analytics: Generating performance insights for teachers and administrators to make data-driven decisions.
* Attendance Management: Automating attendance tracking through AI-powered recognition tools.
* Content Generation: Assisting teachers with AI-based tools to create high-quality learning materials efficiently.

The software will be accessible via web browsers and mobile devices, ensuring flexibility for users. It connects with external services such as authentication servers and AI API servers for facial recognition and secure user management. The AI-Assisted LMS will serve educational institutions ranging from small schools to large universities, offering an efficient and intelligent solution for managing learning and teaching processes.

**Overall description**

**Product perspective**

The AI-Assisted Learning Management System (LMS) is a product designed to address the evolving needs of educational institutions by integrating artificial intelligence into traditional LMS platforms. Unlike existing systems, this product incorporates advanced AI-driven capabilities such as content generation, personalized learning recommendations, automated evaluations, and performance analytics.

The system is developed as a standalone solution but can also integrate with existing educational tools, databases, and infrastructure. It serves as a replacement for outdated or manual systems and enhances existing LMS solutions by offering innovative AI-powered features.

**Key Context**

**Origin:** The system was conceptualized to modernize educational processes, providing a scalable, intelligent, and efficient platform that addresses challenges faced by educators and learners.

**New Capabilities:** The AI-Assisted LMS offers functionalities such as:

* AI-powered content generation and recommendations.
* Automated evaluations and real-time feedback.
* AI-driven attendance management through facial recognition.
* Comprehensive performance analytics for decision-making.

**Operating environment**

Describe the environment in which the software will operate, which might include the hardware platform; operating systems and versions; geographical locations of users, servers, and databases; and organizations that host the related databases, servers, and websites.

Example:

*OE-1: The System shall operate correctly with the following web browsers: Windows Internet Explorer versions 7, 8, and 9; Firefox versions 12 through 26; Google Chrome (all versions); and Apple Safari versions 4.0 through 8.0.*

**Design and implementation constraints**

There are times when a certain programming language must be used, a code library that has already had time invested to develop it needs to be used, and so forth. Describe any factors that will restrict the options available to the developers and the rationale for each constraint.

Example:

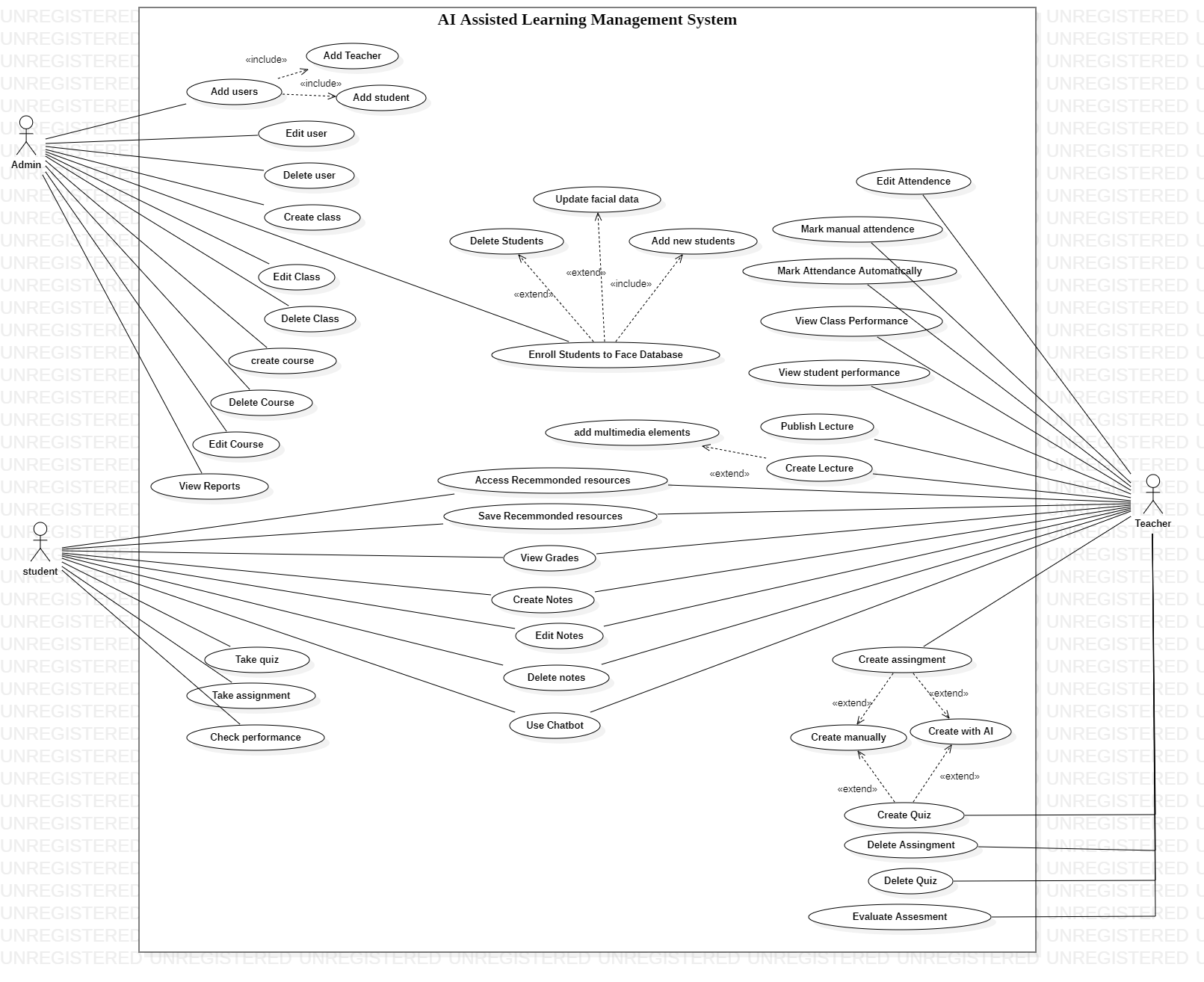
*CO-1: The system shall use the current corporate standard Oracle database engine*

**Requirement identifying technique**

This section describes the requirements identifying technique(s) which further help to derive functional requirements specification. The selection of the technique(s) will depend on the type of project. For instance,

* **Use case** is an effective technique for interactive end-user applications
* **Event- response tables** is for real time system and
* **Story boarding** for graphically intensive applications.

In addition to above, the projects involving data warehouses, batch processes, hardware devices with embedded control software, and computationally intensive applications required to follow other suitable techniques. Such techniques are described further in Chapter 12, “A picture is worth 1024 words.” For documenting this section let consider identifying requirements through use case as an example.

**Use case diagram**

**Use case description**

The table below provides a detailed example of a use case for an AI-Assisted Learning Management System (LMS). This structured template helps document and present user interactions and system functionalities in a clear and organized manner. The use case describes how specific system features are designed to meet user needs and outlines the steps, conditions, and outcomes associated with those interactions.

**Table 1 Show the detail use case template**

**Use Case 1: Access Recommended Resources**

|  |  |
| --- | --- |
| **Use Case ID** | **UC-1** |
| **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** | **1.0 Access Recommended Resources** 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., books, videos, quizzes, assignments). 5. The student reviews the recommendations, which may include explanations, additional reading materials, or practice exercises. 6. The student selects a resource to access. 7. The system directs the student to the selected resource or content. |
| **Alternative Flows** | **1.1 No Recommendations Available** 1. If the system cannot generate recommendations due to insufficient data, it displays: "No recommendations available at this time. Please complete more assessments." **1.2 No Internet Connection** 1. If the student experiences an internet 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. 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. |

**Use Case 2: Save Recommended Resources**

|  |  |
| --- | --- |
| **Use Case ID** | **UC-2** |
| **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** | **1.0 Save Recommended Resources** 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 Flows** | **2.1 Resource Already Saved** 1. If the student tries to save a resource already saved, the system displays: "This resource is already saved in your list." **2.2 Insufficient Data to Save** 1. If the system fails to save the resource, it displays: "Unable to save resource at this time. Please try again later." |
| **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." |
| **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. |

**Use Case 3: Mark Attendance**

|  |  |
| --- | --- |
| **Use Case ID** | **UC-3** |
| **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** | **1.0 Mark Attendance by Teacher** 1. The teacher selects the "Mark Attendance" option. 2. The teacher clicks on the "Capture Attendance" button to open the phone's camera. 3. The system detects students' faces and matches them with the stored data. 4. The system marks attendance automatically for recognized students. 5. The system displays a confirmation message: "Attendance successfully recorded for X students." |
| **Alternative Flows** | **1.1 Face Not Recognized** 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. **1.2 Poor Photo Quality** 1. If the teacher cannot take a clear photo due to low lighting or obstructions, the system prompts them to retake the picture. |
| **Exceptions** | **1.0.E1 Face Recognition Failure** 1. 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** 1. If the phone camera cannot be accessed, the system prompts the teacher to check the camera. |
| **Business Rules** | **BR-1**: Students must be enrolled in the system with their face data registered beforehand. **BR-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. |

**Use Case 4: Enroll Students in Database**

|  |  |
| --- | --- |
| **Use Case ID** | **UC-4** |
| **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** | **1.0 Enroll Students** 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 Flows** | **1.1 Missing Photo** 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** 1. If the admin fails to provide required information, the system displays: "Please provide all required details." **1.0.E2 Image Upload Error** 1. If the image upload fails, the system prompts the admin to retry. |
| **Business Rules** | **BR-1**: The admin must have sufficient privileges to enroll students. **BR-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. |

**Use Case 5: Create notes**

|  |  |
| --- | --- |
| **Use Case ID** | **UC-5** |
| **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 and displays a confirmation message. |
| **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. |

**Use Case 6: Create lecture**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Use Case ID** | UC-6 |
| **Use Case Name** | Create Lecture |
| **Actors** | **Primary Actor:** Teacher  **Secondary Actor:** Content Generation System |
| **Description** | The teacher creates lecture content using the AI-assisted content generation feature. Optionally, the teacher can manually edit AI-generated content. |
| **Trigger** | The teacher clicks the **"Create Lecture"** button in the course module and selects "Create with AI." |
| **Preconditions** | 1. The teacher must be logged into the LMS.  2. The teacher must have access to the specific course within the LMS. |
| **Postconditions** | The lecture is saved as a draft under the "Saved" tab.  The lecture is published under the "Published" tab for student access. |
| **Normal Flow** | 1. The teacher navigates to the "Courses" tab in the dashboard.  2. The teacher selects a course and clicks the "Create Lecture" button.  3. The teacher selects the "Create with AI" option.  4. The system prompts the teacher to enter inputs such as Title, Description, duration, Chapter, Academic Level, Learning Outcomes, Preferred Formats, Attachments. Objective, and Reference.  5. The AI generates lecture content based on the inputs provided and displays it for review.  6. The teacher optionally edits the AI-generated content using the built-in editor.  7. The teacher clicks "Save" to save as a draft or "Save and Publish" to make it available to students. |
| **Alternative Flows** | **AF-1:** Teacher selects "Create Manually":  1. Instead of "Create with AI," the teacher selects "Create Manually."  2. The system displays a form. |
|  | 3. The teacher manually enters all lecture details and content.  4. The teacher clicks "Save" or "Save and Publish" to complete the process. |
| **Exceptions** | **EX-1:** AI Generation Error:  1. The AI fails to generate content due to a technical issue.  2. The system suggests retrying or switching to manual. |
|  | **EX-2:** Save/Publish Error:  1. The system encounters an error while saving or publishing the lecture.  2. The system advises retrying. |
| **Business Rules** | 1. AI-generated content must align with inputs and academic standards.  2. Edited content must pass validation before saving or publishing. |
| **Assumptions** | 1. Teachers have reliable internet access during the creation process.  2. The AI system is functional and accessible. |

**Use Case 7: Create Assignment**

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Use Case ID** | UC-7 |
| **Use Case Name** | Create Assignment |
| **Actors** | Primary Actor: Teacher Secondary Actor: System |
| **Description** | The teacher creates and schedules an assignment by entering details, uploading resources, and applying rubrics. |
| **Trigger** | The teacher clicks the "Create Assignment" button. |
| **Preconditions** | 1. The teacher must be logged into the system. 2. The teacher must have appropriate permissions to create assignments. |
| **Postconditions** | 1. The assignment is saved in the system. 2. The assignment is scheduled for a specified date and time. 3. A confirmation message is displayed. |
| **Normal Flow** | 1. The teacher logs into the system and navigates to the "Assignments" tab. 2. The teacher clicks "Create Assignment." 3. The system displays the "Create Assignment" form. 4. The teacher fills out the form with details such as Assignment Name, Rubrics, References, Due Date, etc. 5. The teacher uploads additional resources or enters assignment content. 6. The teacher customizes or applies a predefined grading rubric. 7. The teacher reviews the details and selects "Save and Publish" or "Save." |
| **Alternative Flows** | **AF-1: Missing Mandatory Fields** 1. The system highlights incomplete fields and prompts the teacher to fill them in. 2. The teacher enters the missing information and continues. **AF-2: Invalid Due Date** 1. The system shows an error if the due date is invalid. 2. The teacher corrects the date and proceeds. |
| **Exceptions** | **EX-1: System Error** 1. The system fails to save the assignment. 2. The system displays an error message and prompts a retry. |
| **Business Rules** | BR-1: Assignments must include either instructions or attached resources. BR-2: Assignments must have a title and a valid due date. |
| **Assumptions** | 1. Teachers will create assignments within course timelines. 2. Stable internet connectivity during assignment creation. |

**Use Case 8: Create Quiz**

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Use Case ID** | UC-8 |
| **Use Case Name** | Create Quiz |
| **Actors** | Primary Actor: Teacher Secondary Actor: System |
| **Description** | The teacher creates and schedules a quiz by entering questions, defining correct answers, and assigning marks. |
| **Trigger** | The teacher clicks the "Create Quiz" button. |
| **Preconditions** | 1. The teacher must be logged into the system. 2. The teacher must have appropriate permissions to create quizzes. |
| **Postconditions** | 1. The quiz is saved in the system. 2. The quiz is scheduled for a specified date and time. 3. A confirmation message is displayed. |
| **Normal Flow** | 1. The teacher logs into the system and navigates to the "Quizzes" tab. 2. The teacher clicks "Create Quiz." 3. The system displays the "Create Quiz" form. 4. The teacher fills out details such as Quiz Title, Rubrics, References, Due Date, etc. 5. The teacher adds questions (e.g., MCQs, Short Questions, Fill-in-the-Blanks) using the question editor. 6. The teacher defines correct answers and assigns marks per question. 7. The teacher reviews the quiz structure and selects "Save and Publish" or "Save." |
| **Alternative Flows** | **AF-1: Missing Mandatory Fields** 1. The system highlights incomplete fields and prompts the teacher to fill them in. 2. The teacher enters the missing information and continues. **AF-2: Invalid Date or Time** 1. The system shows an error if the scheduled date is invalid. 2. The teacher corrects the date and proceeds. |
| **Exceptions** | **EX-1: System Error** 1. The system fails to save the quiz. 2. The system displays an error message and prompts a retry. |
| **Business Rules** | BR-1: Quizzes must include at least one question. BR-2: Quizzes must have a valid scheduled date and time. |
| **Assumptions** | 1. Teachers will create quizzes within course timelines. 2. Stable internet connectivity during quiz creation. |

**Functional Requirements**

This section outlines the functional requirements of the AI-Assisted Learning Management System (LMS), presented in a natural language style. The functional requirements are organized by key features such as Personalized Learning, Automated Evaluations, Performance Analytics, Attendance Management, and Content Generation. Each feature contains specific functional requirements that define the system's capabilities necessary to support the respective user actions or use cases. These requirements describe how the system will behave, including the responses to user inputs and anticipated error conditions, such as invalid input handling or failure scenarios.

**Functional Requirement**

For each feature, individual functional requirements have been labeled and organized with unique identifiers for clarity. Additionally, each requirement includes attributes like rationale, source, dependencies, and priority, which provide further context and reasoning behind the system's design. This approach helps to ensure that the AI-Assisted LMS aligns with both user needs and organizational goals. The following template has been used to document these requirements in a clear and consistent manner.

**Feature 1: Personalized Learning**

**Table: Functional Requirement 1.1**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-PL-001 |
| **Title** | Personalized Content Recommendations |
| **Requirement** | The system shall provide personalized learning material recommendations to each student based on their learning history, preferences, and performance metrics. |
| **Source** | Design Team |
| **Rationale** | To enhance student engagement and learning outcomes by providing relevant content. |
| **Business Rule** | Recommendations must be aligned with the curriculum. |
| **Dependencies** | FR-PL-002, FR-PA-001 |
| **Priority** | High |

**Table: Functional Requirement 1.2**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-PL-002 |
| **Title** | Interactive Material Generation |
| **Requirement** | The system should allow teachers to generate interactive content such as quizzes and flashcards tailored to students' needs. |
| **Source** | Teacher Feedback |
| **Rationale** | To facilitate active learning through engaging materials. |
| **Business Rule** | Materials must adhere to institutional guidelines. |
| **Dependencies** | FR-CG-001 |
| **Priority** | Medium |

**Feature 2: Automated Evaluations**

**Table: Functional Requirement 2.1**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-AE-001 |
| **Title** | Instant Feedback Mechanism |
| **Requirement** | The system shall provide instant feedback to students after quiz submissions, highlighting areas of improvement. |
| **Source** | Assessment Tools Research |
| **Rationale** | To support continuous learning through real-time feedback. |
| **Business Rule** | Feedback must include references to relevant study materials. |
| **Dependencies** | FR-PL-001, FR-PA-002 |
| **Priority** | High |

**Table: Functional Requirement 2.2**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-AE-002 |
| **Title** | Self-Evaluation Tools |
| **Requirement** | The system shall enable students to perform self-assessments using AI-driven evaluations of written and multiple-choice responses. |
| **Source** | Student Feedback Surveys |
| **Rationale** | To empower students with self-directed learning capabilities. |
| **Business Rule** | Self-assessments must not count toward final grades unless explicitly permitted. |
| **Dependencies** | None |
| **Priority** | Medium |

**Feature 3: Performance Analytics**

**Table: Functional Requirement 3.1**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-PA-001 |
| **Title** | Analytics Dashboard for Teachers |
| **Requirement** | The system shall provide a dashboard with visualized analytics on student performance trends, attendance, and engagement. |
| **Source** | Requirements Analysis Document |
| **Rationale** | To assist teachers in making data-driven decisions. |
| **Business Rule** | Data must be anonymized when aggregated. |
| **Dependencies** | FR-AM-001 |
| **Priority** | High |

**Table: Functional Requirement 3.2**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-PA-002 |
| **Title** | Predictive Performance Insights |
| **Requirement** | The system shall use AI to predict students’ future performance based on their learning patterns and attendance records. |
| **Source** | Predictive Modeling Research |
| **Rationale** | To enable proactive intervention strategies. |
| **Business Rule** | Predictions must be flagged as estimates and not definitive results. |
| **Dependencies** | FR-PA-001 |
| **Priority** | High |

**Feature 4: Attendance Management**

**Table: Functional Requirement 4.1**

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Identifier** | FR-AM-001 |
| **Title** | Automated Attendance Tracking |
| **Requirement** | The system shall track student attendance using AI-powered facial recognition during in-person sessions. |
| **Source** | Institutional Requirements Document |
| **Rationale** | To reduce manual errors and save time. |
| **Business Rule** | Facial recognition data must be encrypted and comply with privacy regulations. |
| **Dependencies** | External AI API Server Integration |
| **Priority** | High |

**Feature 5: Content Generation**

**Table: Functional Requirement 5.1**

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Identifier** | CG-01 |
| **Title** | AI-Assisted Lecture Creation |
| **Requirement** | The system must allow teachers to generate lecture content automatically by providing inputs such as learning objectives, topics, and references. |
| **Source** | Design Team |
| **Rationale** | Automating lecture content creation helps reduce the workload and improves efficiency for teachers. |
| **Business Rule** | The generated content must align with provided inputs and adhere to academic standards. |
| **Dependencies** | Requires a stable connection to the AI model and teacher access to the LMS. |
| **Priority** | High |

**Table: Functional Requirement 5.2**

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Identifier** | CG-02 |
| **Title** | Manual Editing of AI-Generated Content |
| **Requirement** | The system must allow teachers to edit and modify the AI-generated lecture content before saving or publishing. |
| **Source** | Teacher Feedback |
| **Rationale** | Teachers may want to customize AI-generated lectures to better fit their teaching style and student needs. |
| **Business Rule** | Edited content should be validated for required fields before saving or publishing. |
| **Dependencies** | Requires successful generation of lecture content using the AI feature. |
| **Priority** | High |

**Table: Functional Requirement 5.3**

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Identifier** | CG-03 |
| **Title** | Save and Publish Options |
| **Requirement** | The system must provide options to save the lecture as a draft or publish it directly for student access. |
| **Source** | Teacher Feedback |
| **Rationale** | Teachers may want flexibility to refine content progressively or make it immediately available to students. |
| **Business Rule** | Drafts are saved under the "Saved" tab and visible only to the teacher; published lectures appear under the "Published" tab for students. |
| **Dependencies** | Edited or AI-generated content must be validated before saving or publishing. |
| **Priority** | Medium |

**Non-Functional Requirements**

This section outlines the quality requirements that ensure the system meets performance, usability, and reliability standards.

**Usability**

*USABILITY-1: The system shall allow users to access the Content Recommendation module and receive recommendations within 2 clicks of the main dashboard.*

*USABILITY-2: The Standard Evaluation module shall provide clear, step-by-step instructions for completing evaluations to minimize user confusion.*

*USABILITY-3: The system interface shall support accessibility features compliant with WCAG 2.1 Level AA standards to accommodate users with disabilities.*

*USABILITY-4: The mobile version of the LMS shall provide consistent and intuitive user experience compared to the web version.*

*USABILITY-5: Self-Evaluation and Attendance modules shall include tooltips to guide users on their functionality.*

**Performance**

*PERFORMANCE-1: 95% of webpages generated by the LMS shall load within 3 seconds over a 20 Mbps or faster internet connection.*

*PERFORMANCE-2: The Content Generation module shall process and generate learning content within 10 seconds for standard text inputs.*

*PERFORMANCE-3: The AI Face Recognition Service (AI API Server) for attendance tracking shall process and verify a student’s face within 2 seconds of submission.*

*PERFORMANCE-4: The system shall support a minimum of 500 concurrent users accessing modules like Analytics and Content Recommendation without performance degradation.*

*PERFORMANCE-5: Database queries in the Analytics module shall return results within 5 seconds for a dataset of up to 100,000 records.*

**Reliability**

*RELIABILITY-1: The system uptime shall be 99.9% over a 24/7 operational period.*

*RELIABILITY-2: The Database Server shall recover automatically from failures within 30 seconds.*

*RELIABILITY-3: The system shall support data backup every 12 hours to prevent data loss.*

**Security**

*SECURITY-1: The system require multi-factor authentication (MFA) for users accessing sensitive modules like Analytics and Attendance.*

*SECURITY-2: Data transferred between Client Devices and servers shall be encrypted using HTTPS/TLS 1.3 protocols.*

*SECURITY-3: All user data stored in the Database Server shall be encrypted using AES-256 encryption standards.*

*SECURITY-4: The system shall log all user activities, including logins, access, and modifications, for auditing purposes.*

**Scalability**

*SCALABILITY-1: The system shall be able to scale horizontally to support up to 10,000 concurrent users by adding additional Application Servers.*

*SCALABILITY-2: The AI Face Recognition Service shall be optimized to scale up to process 5,000 attendance records per hour.*

*SCALABILITY-3: The system shall ensure seamless performance even as the database size grows up to 1 million records.*

**Maintainability**

*MAINTAINABILITY-1: The system shall allow new AI components to be integrated into the Application Server with minimal changes to existing modules.*

*MAINTAINABILITY-2: Code shall follow industry-standard design patterns and include detailed documentation to ensure maintainability.*

*MAINTAINABILITY-3: System logs and error messages shall be stored to facilitate troubleshooting and future system updates.*

**References**

*IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology.*

*IEEE Std 730-1998, IEEE Standard for Software Quality Assurance Plans.*

*IEEE Std 730.1-1995, IEEE Guide for Software Quality Assurance Planning.*

*IEEE Std 828-1998, IEEE Standard for Software Configuration Management Plans.*

*IEEE Std 1028-1997, IEEE Standard for Software Reviews.*