# Use Cases:

1. **Admin Creates User Account**

* **Admin Creates a new user account with user Email and ID**
* **It Auto Generates a random password for the account**

1. **User Logins**

* User clicks on the "LogIn" button.
* User enters their registered College ID/email and password.
* System validates the entered credentials.
* If valid:
  + Users gain access to their account and the platform.
* If invalid:
  + System displays an error message.
  + User has the option to reset the password or contact support.

1. **User Resets Password**

* User clicks on the "Forgot Password" link.
* User is prompted to enter their registered email address.
* System verifies the email address's existence in the database.
* If the email is valid:
  + System sends a password reset link to the user's email.
  + User receives the email and clicks on the reset link.
  + User is redirected to a page where they can create a new password.
  + User enters a new password and confirms it.
  + The system updates the user's password.
  + User can now log in with the new password.
* If the email is not valid:
  + System displays an error message.
  + User is prompted to re-enter their email address.

1. **Teacher Adds Assignment**

* Teacher logs into their account.
* Teacher accesses the specific course where they are the instructor.
* Teacher navigates to the assignment creation section.
* Teacher provides assignment details, including title, description, and due date.
* Teacher sets assignment parameters, such as maximum points.
* Teacher creates the assignment.
* The system notifies enrolled students about the new assignment.

1. **Student Submits Assignment**

* Students log into their account.
* Students access the course where they have an assignment.
* Student navigates to the assignment submission section.
* Students upload their assignment file or enters text.
* Student submits the assignment.
* The system records the submission and notifies the Teacher.

1. **Student Checks Course Grades**

* Student logs into their account.
* Student accesses the course.
* The system displays the quizzes and assignments’ grades for this specific courses

1. **Student Labels and Categorization/Platform Organization**

* System must organize content with clear labels, categories, nested folders, and efficient tagging systems for easy resource discovery.

1. **Student Self-Enrolls with Codes**
   * Student receive the enrollment code and logs into their account.
   * Student navigates to the "Enroll in a Course" section.
   * Student enters the enrollment code provided by the teacher.
   * System verifies the code's validity.
   * If the code is valid:
     + Student is successfully enrolled in the course.
     + System adds the course to the student's enrolled courses.
       1. If the course is not full:
          1. Student can now access the course content.
       2. If the course is full (maximum enrollment reached):
          1. System displays a message indicating that the course is full.
          2. Student can join a waiting list if available or choose another course.
   * If the code is invalid or expired:
     + System displays an error message.
     + Student has the option to re-enter a valid code.
2. **Admin Creates Courses**

* Adminlogs into their account.
* Adminaccesses the course creation section.
* Adminprovides course details, including title, image, description, code, enrollment code, and assigned teachers.
* Admincreates the course.
* The system makes the course available to enrolled students.

1. **Teacher Adds/Reuses Material**

* Teacher logs into their account.
* Teacher navigates a course he previously given and goes to a specific post
* Teacher can reuse the post details by clicking a button.
* Teacher will choose another course and go to their post creation page with all the data prefilled with the older post data

1. **Teacher Adds Quiz**

* Teacher logs into their account.
* Teacher accesses the course where they want to create a quiz.
* Teacher creates a new quiz with questions, including text, multiple-choice.
* Teacher configures the Grade of each question.
* Teacher configures grading options (automatic or manual).
* Teacher configures the time for the quiz.
* Teacher publishes the quiz for students.
* The system notifies enrolled students about the new quiz.

1. **Student Submits Quiz**

* Student logs into their account.
* Student accesses the course with a quiz assignment.
* Student takes the quiz, answering questions.
* Student submits the quiz.
* System grades the quiz automatically (if configured) or awaits manual grading by the instructor.
* The system records the submission and notifies the Teacher.

1. Teacher Grades Quiz

* Teacher can access a certain Quiz by going to the course home page and checking the quiz menu
* Upon selecting the desired quiz and accessing its page, he will find a tab where a list of all students in this course will be provided with their submission status
* He can select a submission where he will go through each question and that students answer manually and select whether it is correct or not

1. Student Checks Pending deadlines:

* Student can either Enter a course home page and check the quizzes and assignment sections
* Alternatively the student can access the calendar page where all upcoming dates for pending tasks will be shown

1. Teacher Creates Post

* Teacher can Enter any course home page where he has privilege to post and simply access the post creation area
* He must add a title and description for the post
* He can Add a schedule date, a poll, and materials to the post

1. **Teacher Integrates YouTube Video**

* Teacher logs into their account.
* Teacher accesses the course where they want to include a YouTube video.
* When teacher creates a post they can paste a YouTube link inside the post description and it will get pasted into an embed automatically
* System validates the URL and fetches the video details.
* System adds the YouTube video in the course content.

1. **User Accesses Settings**

* Users can access their account settings by clicking on their profile picture or username.
* Within the user settings section, users can customize various aspects of their platform experience, including:
* **Profile Information:**
* Users can update their profile picture.
* Users can edit their name, bio, or other personal details.
* **Privacy Settings:**
* Users can configure privacy settings for their profile, such as who can view their profile and contact them.
* **Notification Preferences:**
* Users can manage their notification preferences, including email notifications, course updates, and announcements.
* **Language Preferences:**
* Users can select their preferred language for the platform's interface.
* **Theme Selection:**
* Users can choose between different themes (e.g., light mode, dark mode) for the platform's appearance.
* **Accessibility Settings:**
* Users can enable or customize accessibility features, such as text-to-speech or high-contrast mode.
* **Security Settings:**
* Users can update their password or enable two-factor authentication for added security.
* **Email Preferences:**
* Users can specify their email communication preferences, such as newsletter subscriptions.
* **Data Management:**
* Users can access tools to manage their data, including downloading their course materials and personal information.
* Users can save their settings, and the platform will apply the changes accordingly.

1. **Student Feedback Mechanisms**

* System must allow learners to provide feedback on courses, content, and the platform itself through surveys, ratings, and open forums.

1. **Assessment Tools/Teacher Dashboard**

* System must offer assessment tools for instructors to design and administer quizzes, assignments, and assessments.
* System must provide automated grading, feedback mechanisms, course modifications, and course analytics to teachers.

1. **Student Downloads Materials**

* Students logs into their account.
* Students accesses the course for which they want to download materials.
* Students navigates to the "Course Materials" or "Resources" section of the course.
* Students locates the specific material they want to download (e.g., a document, video, presentation).
* Students clicks on the material to download it.
* Students selects the download location and confirms the download.
* The system initiates the download process.

1. **Admin Manages User**

* The Admin can view a list of all users registered on the platform.
* The Admin can search for specific users based on criteria such as name, email, or role (Teacher, Student, etc.).
* The Admin can access individual user profiles to view and edit their information.
* The Admin can reset passwords for users or force password changes.
* The Admin can deactivate or suspend user accounts if necessary.
* The Admin can reactivate previously suspended accounts.

1. **Admin Manages Courses**

* The Admin can view a list of all courses offered on the platform.
* The Admin can modify or delete courses.
* The Admin can assign or change instructors for courses.
* The Admin can monitor course enrollment and view enrollment statistics.
* The Admin can set course enrollment limits.
* The Admin can archive or temporarily hide courses.
* The Admin can review and approve/disapprove course content created by teachers.

1. **Teacher Assigns Roles and Permissions**

* The Teachercan define and manage user roles and permissions.
* The Teachercan create custom user roles with specific permissions.
* The Teachercan assign roles to users or groups of users.
* The Teachercan revoke or modify permissions for specific users or roles.
* The Teachercan track changes to roles and permissions.

1. **Admin Reviews Reports and Analytics**

* The Admin can access comprehensive reports and analytics on user activity, course engagement, and platform usage.
* The Admin can use analytics to identify trends, areas for improvement, and potential issues.

1. **Admin Supports User**

* The Admin can provide support to users, including responding to inquiries and troubleshooting issues.
* The Admin can access a support ticketing system to manage and resolve user-reported problems.

1. **Admin Manages Platform Maintenance and Updates**

* The Admin can schedule and perform routine maintenance tasks, including updates and backups.
* The Admin can ensure the platform remains up-to-date with the latest security patches and features.

# 

# Use Case Diagrams:A diagram of a user Description automatically generated

# 



# Use Case Description:

## 1: User Register:

| Use case ID | Case 1 | |
| --- | --- | --- |
| Use Case Name | User Registration | |
| Actors | Student, teacher, Admin | |
| Pre-condition | None | |
| Post-condition | User is registered and can log in. | |
| Flow of events | User Action | System Action |
| 1. Users enter their username, email, and password. |  |
|  | 1.1 system check the validation of the entered email, ensuring it is in the correct format and not already registered. |
| Exception scenario | Invalid email format or email already registered. | |

## 2: User Login

| Use case ID | Case 2 | |
| --- | --- | --- |
| Use Case Name | User Login | |
| Actors | Student, teacher, Admin | |
| Pre-condition | User is registered. | |
| Post-condition | User is logged into the platform. | |
| Flow of events | User Action | System Action |
| 1. Users enter their registered email and password. |  |
|  | 1.2. System validate the entered credentials. |
| Exception scenario | Invalid login credentials. | |

## 3: User Reset Password:

| Use case ID | Case 3 | |
| --- | --- | --- |
| Use Case Name | Password Reset | |
| Actors | Student, teacher, Admin | |
| Pre-condition | User has a registered account. | |
| Post-condition | User can log in with a new password. | |
| Flow of Events | User Actions | System Actions |
| 1. User enter their email and click on the "Forgot Password" link. |  |
|  | 1.1. System send a password reset link to the user's email. |
| 1. User receive the email and click on the reset link. |  |
|  | 2.1. User is redirected to a page where they can create a new password. |
| 1. User enter a new password and confirms it. |  |
|  | 3.1. The system update the user's password. |
| Exception scenario | Invalid or unregistered email address. | |

## 4: Student Enroll in Course:

| Use case ID | Case 4 | |
| --- | --- | --- |
| Use Case Name | Course Enrollment | |
| Actors | Student | |
| Pre-condition | Student is logged into their account. | |
| Post-condition | Student is enrolled in the course and can access course content. | |
| Flow of Events | User Actions | System Actions |
| 1. Student navigate to the course catalog or search feature. |  |
|  | 1.1. System display the course to the student |
| 1. Student enter enrollment code or invitation link |  |
|  | 2.1. System enroll the student in the course |
| Exception scenario | Course code or invitation link is not valid Or Course is at maximum enrollment capacity. | |

## 5: Teacher Add Assignment:

| Use case ID | Case 5 | |
| --- | --- | --- |
| Use Case Name | Assignment Creation | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and accesses the course. | |
| Post-condition | Assignment is created and students are notified. | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher provide assignment details, including title, description, and due date. |  |
|  | 1.1. The system notify enrolled students about the new assignment. |
| 1. Teacher create the assignment. |  |
| Exception scenario | None | |

## 6: Student Submit Assignment:

| Use case ID | Case 6 | |
| --- | --- | --- |
| Use Case Name | Assignment Submission | |
| Actors | Student | |
| Pre-condition | Student is logged into their account and access the course with an assignment. | |
| Post-condition | Assignment submission is recorded. | |
| Flow of Events | User Actions | System Actions |
| 1. Student navigate to the assignment submission section. |  |
|  | 1.1.. System redirect him to the submission page |
| 1. Student upload their assignment file or enter text and submit it |  |
|  | 2.1.. The system record the submission and notify the teacher. |
| Exception scenario | The file size is too big or file format not supported | |

## 7: Student Check Progress:

| Use case ID | Case 7 | |
| --- | --- | --- |
| Use Case Name | Progress Tracking | |
| Actors | Student | |
| Pre-condition | Student is logged into their account and access the course. | |
| Post-condition | Student view their course progress. | |
| Flow of Events | User Actions | System Actions |
| 1. Student view the course progress dashboard. |  |
|  | 1.1.. The system display completed and pending lectures and tasks. |
| 1. Student can track their progress and navigate to different course sections. |  |
| Exception scenario | None | |

## 8: Student Label and Categorization/Platform Organization:

| Use case ID | Case 8 | |
| --- | --- | --- |
| Use Case Name | Content Organization | |
| Actors | Student | |
| Pre-condition | Student is logged into their account and have at least one enrolled course. | |
| Post-condition | Content is organized with clear labels and categories. | |
| Flow of Events | User Actions | System Actions |
| 1. Student access Organization tools |  |
|  | 1.1.. The system provides labeling, categorization, and tagging features. |
| 1. User interacts with the content organization features to classify resources. |  |
| Exception scenario | None | |

## 9: Admin Creates Enrollment Option:

| Use case ID | Case 9 | |
| --- | --- | --- |
| Use Case Name | Enrollment Creation (Access Code, Invitation Link) | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their account and accesses the course management section. | |
| Post-condition | Enrollment Option is generated and can be shared with potential students. | |
| Flow of Events | User Actions | System Actions |
| 1. The Admin choose an enrollment options for the course. |  |
|  | 1.1. System generates enrollment Option |
| 1. Admin share the enrollment option with potential students through email, messages, or the platform. |  |
| Exception scenario | None | |

## **10: Teacher Adds/Reuses Material**:

| Use case ID | Case 10 | |
| --- | --- | --- |
| Use Case Name | Material Management | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and accesses the course materials section. | |
| Post-condition | Course materials are added or reused for the course. | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher upload course materials (readings, videos, assignments) or reuses previously uploaded materials. |  |
|  | 1.1. System apply the changes and save it in the database |
| Exception scenario | None | |

## 11: Teacher Add Quiz:

| Use case ID | Case 11 | |
| --- | --- | --- |
| Use Case Name | Quiz Creation | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and accesses the course where they want to create a quiz. | |
| Post-condition | The quiz is published for students in the course. | |
| Flow of Events | User Actions | System Actions |
|
| 1. Teacher create a new quiz |  |
|
|  | 1.1. The system notifies enrolled students about the new quiz. |
|
| Exception Scenarios | None | |

## **12: Student Submit Quiz:**

| Use case ID | Case 12 | |
| --- | --- | --- |
| Use Case Name | Quiz Submission | |
| Actors | Student | |
| Pre-condition | Student is logged into their account and accesses the course with a quiz assignment. | |
| Post-condition | Quiz submission is recorded and awaits grading. | |
| Flow of Events | User Actions | System Actions |
| 1. Student enter the quiz |  |
|  | 1.1. System redirect him to the quiz screen |
| 1. Student answer and submit the quiz. |  |
|  |  | 2.1. The system record the submission and grades it. |
| Exception scenario | None | |

## 

## 13: Teacher Integrate YouTube Video

| Use case ID | Case 13 | |
| --- | --- | --- |
| Use Case Name | YouTube Video Integration | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and accesses the course where they want to include a YouTube video. | |
| Post-condition | The YouTube video is added to the course content. | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher paste the YouTube video link (URL) into the provided field. |  |
|  | 1.1. System validate the URL and fetches the video details. |
|  | 1.2. System add the YouTube video in the course content. |
| Exception scenario | Invalid or inaccessible YouTube video URL. | |

## 14: User Access Settings

| Use case ID | Case 14 | |
| --- | --- | --- |
| Use Case Name | Accessing User Settings | |
| Actors | Student, Teacher, Admin | |
| Pre-condition | User is logged into their account. | |
| Post-condition | User's settings are updated as per their preferences. | |
| Flow of Events | User Actions | System Actions |
| 1. User access account settings. |  |
|  | 1.1. System display list of customization tools to the user |
| 1. Users customize aspects including profile information, privacy settings, notification preferences, language preferences, theme selection, accessibility settings, security settings, email preferences, and data management then save it. |  |
|  | 2.1. System applies changes |
| Exception scenario | None | |

## 15: Feedback Mechanisms

| Use case ID | Case 15 | |
| --- | --- | --- |
| Use Case Name | Providing Feedback | |
| Actors | Student, Teacher | |
| Pre-condition | User is logged into their account. | |
| Post-condition | User feedback is collected for improvement. | |
| Flow of Events | User Actions | System Actions |
| 1. Users provide feedback through surveys, ratings, and open forums. |  |
|  | 1.1. System record the provided feedback |
| Exception scenario | None | |

## 16: Assessment Tools/Teacher Dashboard

| Use case ID | Case 16 | |
| --- | --- | --- |
| Use Case Name | Assessment and Dashboard | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and has access to the course management section. | |
| Post-condition | Teachers can design, administer assessments and manage courses efficiently. | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher Access Assesment Tools |  |
|  | 1.1. System provide assessment tools, automated grading, feedback mechanisms, course modifications, and course analytics to teachers. |
| 1. Teacher design and administers quizzes, assignments, and assessments. |  |
|  | 2.1. System will apply changes and save it in the database |
| Exception scenario | None | |

## 17: Student Download Materials

| Use case ID | Case 17 | |
| --- | --- | --- |
| Use Case Name | Downloading Course Materials | |
| Actors | Student | |
| Pre-condition | Student is logged into their account and accesses the course materials section. | |
| Post-condition | The selected course material is downloaded to the student's device. | |
| Flow of Events | User Actions | System Actions |
| 1. Student navigate to the "Course Materials" or "Resources" section of the course. |  |
|  | 1.1. System will display the material list |
| 1. Within the material view, the student finds and clicks on the "Download" button/icon. |  |
|  | 2.1. The system generate the downloaded file and save it in the user’s device. |
| Exception scenario | The user device has insufficient space | |

## 18: Admin Add User

| Use case ID | Case 18 | |
| --- | --- | --- |
| Use Case Name | User Creation | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | User account is created | |
| Flow of Events | User Actions | System Actions |
| 1. Admin create a new user account |  |
|  |  | 1.1. System save it in the database |
| Exception scenario | None | |

## 19: Admin Update User Data

| use case ID | Case 19 | |
| --- | --- | --- |
| Use Case Name | User Update | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | User accounts are updated according to Admin's actions. | |
| Flow of Events | User Actions | System Actions |
| 1. Admin view the platform users |  |
|  | 1.1. System display the users in the platform |
| 1. Admin access individual user profiles to view and Update their information if necessary. |  |
|  | 2.1. System apply the changes and save it in the database |
| Exception scenario | None | |

## 

## 20: Admin Delete User

| Use case ID | Case 20 | |
| --- | --- | --- |
| Use Case Name | User Deletion | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | User accounts are deleted according to Admin's actions. | |
| Flow of Events | User Actions | System Actions |
| 1. Admin view the platform users |  |
|  | 1.1. System display the users in the platform |
|  | 1. Admin access individual user profiles to view and Delete their information if necessary. |  |
|  |  | 2.1. System apply the changes and save it in the database |
| Exception scenario | None | |

## 21: Admin Add Courses

| Use case ID | Case 21 | |
| --- | --- | --- |
| Use Case Name | Course Creation | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | Course is Created | |
| Flow of Events | User Actions | System Actions |
| 1. Admin create the course and set it’s limitation |  |
|  | 1.1. System will save course to the database |
| Exception scenario | None | |

## 

## 22: Admin Update Courses

| Use case ID | Case 22 | |
| --- | --- | --- |
| Use Case Name | Course Update | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | Course-related actions are executed according to the Admin's decisions. | |
| Flow of Events | User Actions | System Actions |
| 1. Admin Enter to the courses section |  |
|  | 1.1. System will display list of all courses offered on the platform. |
| 1. Admin modify specific course. |  |
|  | 2.1. System will apply the changes and save it |
| Exception scenario | None | |

## 23: Admin Delete Courses

|  |  |  |
| --- | --- | --- |
| Use case ID | Case 23 | |
| Use Case Name | Course Delete | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | Course-related actions are executed according to the Admin's decisions. | |
| Flow of Events | User Actions | System Actions |
| 1. Admin Enter to the courses section |  |
|  | 1.1. System will display list of all courses offered on the platform. |
|  | 1. Admin select courses to delete it. |  |
|  |  | 2.1. The system will delete it from database |
| Exception scenario | None | |

## 24: Admin Collect Reports and Analytics

| Use case ID | Case 24 | |
| --- | --- | --- |
| Use Case Name | Reports and Analytics Gathering | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | Admins can make informed decisions based on the insights from reports and analytics. | |
| Flow of Events | User Actions | System Actions |
| 1. Admin Access Reports And Analytics |  |
|  | 1.1. System collect comprehensive reports and analytics data on user activity, course engagement, and platform usage. |
| 1. Admin use analytics to identify trends, areas for improvement, and potential issues. |  |
| Exception scenario | None | |

## 

## 25: Admin Monitor System Configuration

| Use case ID | Case 25 | |
| --- | --- | --- |
| Use Case Name | System Configuration Monitor | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | Platform settings and configurations are adjusted based on Admin's actions. | |
| Flow of Events | User Actions | System Actions |
| 1. Admin configure system settings and parameters according to specific needs. |  |
|  | 1.1. System apply the changes to the platform and save it |
|
|
|
| Exception scenario | None | |

## 26: Admin Review Ticket

| Use case ID | Case 26 | |
| --- | --- | --- |
| Use Case Name | Review Ticket | |
| Actors | Admin | |
| Pre-condition | Admin is logged into their Admin account. | |
| Post-condition | User ticket has been reviewed | |
| Flow of Events | User Actions | System Actions |
| 1. Admin access a ticketing system. |  |
|  | 1.1. System offer the list of tickets to the admin |
| 1. Admin review the tickets and take an action. |  |
| Exception scenario | None | |

## 27: User Open Ticket

| Use case ID | Case 27 | |
| --- | --- | --- |
| Use Case Name | User Ticket | |
| Actors | Student, Teacher | |
| Pre-condition | User is logged into their Admin account. | |
| Post-condition | Admin receive the ticket and be suspended until he review it | |
| Flow of Events | User Actions | System Actions |
| 1. User Create a new ticket and add it’s details. |  |
|  | 1.1. System will send the created ticket to the Admin to review it |
| Exception scenario | None | |

## **28: Teacher Update Material**:

| Use case ID | Case 28 | |
| --- | --- | --- |
| Use Case Name | Material Update | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and accesses the course materials section. | |
| Post-condition | Course materials are Updated for the course. | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher Update course materials (readings, videos, assignments). |  |
|  |  | 1.1. System apply the changes and save in the database |
| Exception scenario | None | |

## **29: Teacher Delete Material**:

| Use case ID | Case 29 | |
| --- | --- | --- |
| Use Case Name | Material Deletion | |
| Actors | Teacher | |
| Pre-condition | Teacher is logged into their account and accesses the course materials section. | |
| Post-condition | Course materials are Deleted for the course. | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher enter to the selected course |  |
|  |  | 1.1. System display all the content of this course |
|  | 1. Teacher will select some materials to delete from the course |  |
|  |  | 2.1. System will delete the it from the course and apply changes |
| Exception scenario | None | |

## 30: Teacher Assign Rules

| Use case ID | Case 30 | |
| --- | --- | --- |
| Use Case Name | Rules Assignment | |
| Actors | Teacher | |
| Pre-condition | Teacher is assigned to the course and Student is enrolled into the course | |
| Post-condition | Student rules updated | |
| Flow of Events | User Actions | System Actions |
| 1. Teacher Select specific student and assign rules to him |  |
|  |  | 1.1. System save the updated rules |
| Exception scenario | None | |