Use Case 1:

Use Case:	Create a new account
Primary Actor:	User (Administrator, Course Staff, Student)
Goal in Context:	To facilitate any user to sign up for a new account by providing full name, email, new password and confirming the new password.
Preconditions:	System has been programmed for any user to create a new account providing credentials which can be used later for logging in again.
Trigger:	A user decides to create a new account.

Scenario:	1. User clicks on "SignUp" button
	2. User then fills the following fields:
	    • Full Name
	    • Email address (a valid one)
	    • Password (As per requirement)
	    • Confirm Password (Same as above)
	3. User then submits the form.

Exceptions:	If user tries to submit the form with any of the below issues, new user won't be created: 1. Any of the fields are left unfilled 2. Incorrect email address 3. The password doesn't match the given password requirements 4. The password typed twice do not match Remedy: Avoiding the above issues would allow a user to create a new account successfully.

Use Case 2:

Use Case:	Set new password using forgot password
Primary Actor:	User (Administrator, Course Staff, Student)

Goal in Context:	To allow an alternative for a legitimate user to decide a new password in the scenario of forgetting the current password.
Preconditions:	The user should already have a verified account.
Trigger:	A user forgets his/her password and decides to reset it using the "Forgot Password" facility.
Scenario:	1. User clicks on "Forgot Password" button on Login Page 2. They enter email address in the form. 3. The user then submits the form 4. The user receives an email with the credentials 5. The user clicks on the reset link in the email. 6. User enters new password as per requirements and confirms password.
Exceptions:	If user tries to submit the forms with any of the below issues, they would not be able to login unless they login using correct current password: 1. User tries to submit the forms keeping any of the required fields empty. 2. The user does not have an account. 3. Password for the user does not match. Remedy: Avoiding the above issues would allow a user to set a new password successfully.

Use Case 3:

Use Case:	Review Plagiarism Report
Primary Actor:	Course Staff
Goal in Context:	Allow course staff to review different submissions' similarity by analyzing the plagiarism report.
Preconditions:	A plagiarism report is required for the analysis and review.
Trigger:	A course staff member wishes to check if any two or more users have submitted plagiarized assignment having similarity greater than a threshold.

Scenario:	1. Select the plagiarism report of the student whose submission is needed to be checked. 2. Compare the similarity score to the pre-decided threshold and come to a decision if the submission is plagiarized or not.
Exceptions:	If the plagiarism report is not already generated, review is not possible. <u>Remedy</u>: The course staff must generate a plagiarism report before reviewing results.

Use Case 4:

Use Case:	**Provide feedback**
Primary Actor:	Course Staff
Goal in Context:	To allow a course staff to give a feedback regarding any issues faced while using the software, or request for some functionality to be added.
Preconditions:	The user logged in should be a course staff.
Trigger:	A course staff faces some issue or wishes to give some feedback regarding the software.
Scenario:	1. Course staff clicks on "Provide Feedback" link 2. They enter the feedback. 3. They submit the feedback.
Exceptions:	If the course staff tries to submit the form without mentioning any feedback, the feedback won't be sent. <u>Remedy</u>: Typing proper feedback will lead to provide feedback successfully.

Use Case 5:

Use Case:	**Ad hoc run on files of two users**
Primary Actor:	Course Staff

Goal in Context:	To facilitate a course staff to compare all the files of two users for plagiarism detection and score calculation..
Preconditions:	1. The two users should have at least one file to be compared with. 2. The files should be related to the same assignment of the same course. 3. Each file should be a Python file (extension of .py).
Trigger:	A course staff selects two students' submissions and wishes to detect plagiarism in them.
Scenario:	1. Course staff clicks on "Check user to user" link 2. They select assignment number, and two students' ID numbers. 3. Course staff clicks on "Calculate" button.
Exceptions:	If the course staff tries to submit the form without mentioning which assignment or students, or if any of the student has not submitted the assignment, the ad-hoc run won't be possible. Remedy: Avoiding above scenarios can help to do an ad-hoc run on two users' submissions successfully.

Use Case 6:

Use Case:	**Adding assignments to the course in the system**
Primary Actor:	Course staff
Goal in Context:	Course staff being able to add more assignments to the course
Preconditions:	Course staff should be handling the course for which they want to add the assignments
Trigger:	Course staff selecting to add a new assignment
Scenario:	1. A new assignment released in the course. 2. Course staff selects to add a new assignment to the course in the system.
Exceptions:	1. The course staff is not registered in the system.

	Remedy: They do not get access to the system.
	2. The course staff does not handle the course for which they want to add the assignment.
	Remedy: Course staff cannot view the course on the system.

Use Case 7:

Use Case:	**Verification of new sign up**
Primary Actor:	Administrator
Goal in Context:	Administrator verifies the new registrations and authorizes each new registration accordingly.
Preconditions:	User should not be registered in the system
Trigger:	User fills up and submits the registration form
Scenario:	1. User completes the registration form with all the required fields.
	2. User submits the form.
	3. Administrator receives the details of the form filled by the user.
	4. Administrator reviews the form.
	5. Admin decides to grant the user access to the system.
	6. An email consisting of user account credentials is sent to the user email.
Exceptions:	1. If the registration form is incomplete.
	Remedy: The user receives a warning in a window asking s/he to complete the form before submission.
	2. If the administrator denies the user's registration application.
	Remedy: The user receives an email informing them about their application being denied access.

Use Case 8:

Use Case:	**Change the roles of users**

Primary Actor:	Administrator
Goal in Context:	Change the role of a registered user from one type of actor to another
Preconditions:	A user must be registered into the system
Trigger:	User requests to change their role
Scenario:	1. User registered as a student in the system. 2. Student becomes a teaching assistant for the same course in the later semester, and needs access privileges as a member of the course staff. 3. User requests to change the role from that of student to course staff. 4. Admin verifies the request and grants/denies access accordingly.
Exceptions:	1. When the user is not registered into the system. Remedy: the user is not able to access this feature. 2. If the user does not belong to the institution where the course is being offered. Remedy: the user cannot request change of role.

Use Case 9:

Use Case:	**Student should be able to upload code file**
Primary Actor:	Student
Goal in Context:	Upload assignment
Preconditions:	Student should be registered in the system
Trigger:	Student wants to upload assignment
Scenario:	1. Student Logs In 2. Selects Assignment no. 3. Uploads the File to the system.
Exceptions:	Student is not able to upload file, if the file extension is not python file extension. Remedy : File extension should be a proper python extension.

Use Case 10:

Use Case:	**Student should be able to upload code github link**
Primary Actor:	Student
Goal in Context:	Students wants to submit Assignment code github URL
Preconditions:	Student should be registered in the system
Trigger:	Student wants to submit assignment
Scenario:	1. Student logs in the system. 2. Student selects the assignment no. to upload. 3. Student shares link to github for assignment
Exceptions:	Student is not able to submit github url. If the URL is not a valid github repo url ending with a .git extension. <u>Remedy</u>: Add proper Github url.