

Use Case:	Upload project folders
Primary Actor:	Professor/TA
Goal in Context:	To upload project folders on which system needs to run plagiarism test
Preconditions:	Appropriate user id and password must be used to login to the portal
Trigger:	Professor/TA suspects two projects to be similar and would like to run plagiarism detection on those projects
Scenario:	<ol style="list-style-type: none"> 1. Professor/TA opens the portal 2. Professor/TA enters his username and password 3. The portal displays homepage of the portal on successful validation of the user 4. Professor/TA clicks the Browse button on the left side and gives the path of the directory of project1 in the local machine 5. Professor/TA clicks the Browse button on the right side and gives the path of the directory of project2 in the local machine 6. Once both directories path is provided, "Upload" button is enabled 7. Professor/TA clicks the upload button 8. "Upload successful" message is displayed
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect 2. The directories are empty
Priority:	High priority, needs to be implemented as a basic function as other use cases are highly dependent on this
When Available:	First release

Use Case:	Perform plagiarism detection
Primary Actor:	Professor/Teaching Assistant
Goal in Context:	To detect plagiarism on the uploaded files
Preconditions:	Non-empty project folders must be uploaded to both source and target
Trigger:	Professor/TA has uploaded two suspicious project folders and would like to run plagiarism test on them
Scenario:	<ol style="list-style-type: none"> 1. Professor/TA opens the portal 2. Professor/TA enters his username and password 3. The portal displays homepage of the portal on successful validation of the user 4. Professor/TA clicks the Browse button on the left side and

	<p>gives the path of the directory in the local machine</p> <ol style="list-style-type: none"> Professor/TA clicks the Browse button on the right side and gives the path of the directory in the local machine Once both directories path is provided, "Upload" button is enabled Professor/TA clicks the "Upload" button Once both directories are uploaded, Professor/TA can click "Check for Plagiarism" button
Exceptions:	<ol style="list-style-type: none"> Login credentials are incorrect The directories are empty The uploaded files are written in language other than desired language
Priority:	High priority, needs to be implemented as a basic function
When Available:	First release

Use Case:	View plagiarism detection result
Primary Actor:	Professor/Teaching Assistant
Goal in Context:	To view detailed plagiarism report
Preconditions:	Plagiarism test should be performed on two projects
Trigger:	Professor/TA has run plagiarism test on two suspicious project folders and would like to view the detailed report of the same
Scenario:	<ol style="list-style-type: none"> Professor/TA opens the portal Professor/TA enters his username and password The portal displays homepage of the portal on successful validation of the user Professor/TA clicks the Browse button on the left side and gives the path of the directory in the local machine Professor/TA clicks the Browse button on the right side and gives the path of the directory in the local machine Once both directories path is provided, "Check for plagiarism" button is enabled Professor/TA clicks the "Check for Plagiarism" button Professor/TA clicks "View Report" button A full report indicating the similarity score and similar files with similar portion highlighted is shown to the user
Exceptions:	<ol style="list-style-type: none"> Login credentials are incorrect The directories are empty The uploaded files are written in language other than desired language
Priority:	High priority, needs to be implemented as a basic function

When Available:	First release
------------------------	---------------

Use Case:	Download plagiarism detection result
Primary Actor:	Professor/Teaching Assistant
Goal in Context:	To download detailed plagiarism report
Preconditions:	Plagiarism test should be performed on two projects
Trigger:	Professor/TA has run plagiarism test on two suspicious project folders and would like to download the detailed report of the same
Scenario:	<ol style="list-style-type: none"> 1. Professor/TA opens the portal 2. Professor/TA enters his username and password 3. The portal displays homepage of the portal on successful validation of the user 4. Professor/TA clicks the Browse button on the left side and gives the path of the directory in the local machine 5. Professor/TA clicks the Browse button on the right side and gives the path of the directory in the local machine 6. Once both directories path is provided, "Check for plagiarism" button is enabled 7. Professor/TA clicks the "Check for Plagiarism" button 8. Professor/TA clicks "Download Report" button 9. A full report indicating the similarity score and similar files with similar portion highlighted is downloaded as pdf to users' local disk
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect 2. The directories are empty 3. The uploaded files are written in language other than desired language
Priority:	Medium priority
When Available:	Second release

Use Case:	Mail plagiarism detection report to professor
Primary Actor:	Teaching Assistant
Goal in Context:	To mail detailed plagiarism report to professor once TA has found similarity more than threshold value
Preconditions:	Plagiarism test should be performed on two projects
Trigger:	TA has run plagiarism test on two suspicious project folders, found similarity more than threshold and would like to send the detailed report of the same to the professor

Scenario:	<ol style="list-style-type: none"> 1. TA opens the portal 2. TA enters his username and password 3. The portal displays homepage of the portal on successful validation of the user 4. TA clicks the Browse button on the left side and gives the path of the directory in the local machine 5. TA clicks the Browse button on the right side and gives the path of the directory in the local machine 6. Once both directories path is provided, "Check for plagiarism" button is enabled 7. TA clicks the "Check for Plagiarism" button 8. TA finds the similarity score greater than threshold 9. TA clicks "Report to Professor" button
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect 2. The directories are empty 3. The uploaded files are written in language other than desired language
Priority:	Medium priority, can be implemented after basic functions
When Available:	Second release

Use Case:	Report to Disciplinary Board
Primary Actor:	Professor
Goal in Context:	To mail detailed plagiarism report to OSCCR once Professor has found similarity more than threshold value
Preconditions:	Plagiarism test should be performed on two projects
Trigger:	Professor has run plagiarism test on two suspicious project folders, found similarity more than threshold and would like to send the detailed report to OSCCR
Scenario:	<ol style="list-style-type: none"> 1. Professor opens the portal 2. Professor enters his username and password 3. The portal displays homepage of the portal on successful validation of the user 4. Professor clicks the Browse button on the left side and gives the path of the directory in the local machine 5. Professor clicks the Browse button on the right side and gives the path of the directory in the local machine 6. Once both directories path is provided, "Check for plagiarism" button is enabled 7. Professor clicks the "Check for Plagiarism" button 8. Professor verifies the similarity score greater than threshold 9. Professor clicks "Report to Disciplinary Board" button

	10. "Successfully reported to Disciplinary Board" message is displayed
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect 2. The directories are empty 3. The uploaded files are written in language other than desired language
Priority:	Medium priority, can be implemented after basic functions
When Available:	Second release

Use Case:	Add user
Primary Actor:	Admin
Goal in Context:	To add users who can access this application
Preconditions:	User should login as admin with right credentials
Trigger:	At the start of a semester, new TAs should be added so that they can access the application
Scenario:	<ol style="list-style-type: none"> 1. Admin opens the portal 2. Admin enters his username and password 3. The portal displays homepage of admin on successful validation of the user 4. Admin clicks the Add User button 5. A popup asks for new user's official mail id 6. Once email id given is validated, an email is sent to the user with a link to sign up to this portal 7. Once the user signs up, he/she is added to the list of users
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect
Priority:	Medium priority, can be implemented after basic functions
When Available:	Second release

Use Case:	Delete user
Primary Actor:	Admin
Goal in Context:	To delete users existing in the database
Preconditions:	User should login as admin with right credentials

Trigger:	At the end of the semester, TAs might not continue for the next semester
Scenario:	<ol style="list-style-type: none"> 1. Admin opens the portal 2. Admin enters his username and password 3. The portal displays homepage of admin on successful validation of the user 4. Admin can see the list of users who already uses this application 5. Admin clicks the delete button right next to the user 6. User is removed from the database 7. "User successfully removed" message is displayed
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect
Priority:	Medium priority, can be implemented after basic functions
When Available:	Second release

Use Case:	Grant access to the user
Primary Actor:	Admin
Goal in Context:	To grant access to the user who has signed up
Preconditions:	User should login as admin with right credentials
Trigger:	Admin wishes to grant access to user for using this application
Scenario:	<ol style="list-style-type: none"> 1. Admin opens the portal 2. Admin enters his username and password 3. The portal displays homepage of admin on successful validation of the user 4. Admin can see the list of users who have signed up for the application and are waiting for approval 5. Admin clicks the "Grant Access" button right next to the user 6. User is added to the database 7. "User successfully added" message is displayed
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect
Priority:	Medium priority, can be implemented after basic functions
When Available:	Second release

Use Case:	Upload project folders directly from github
------------------	---

Primary Actor:	Professor/TA
Goal in Context:	To download project folders directly from github or any other repository to which student uploads assignment and upload it to the portal
Preconditions:	Appropriate user id and password must be used to login to the portal
Trigger:	Student has given the url of the public repository to which he has uploaded the project and professor/TA would like to run plagiarism test on it
Scenario:	<ol style="list-style-type: none"> 1. Professor/TA opens the portal 2. Professor/TA enters his username and password 3. The portal displays homepage of the portal on successful validation of the user 4. Professor/TA enters the url of the repository to which student has uploaded the assignment 5. "Upload successful" message is displayed once the files downloaded from repository are uploaded to the plagiarism detection tool
Exceptions:	<ol style="list-style-type: none"> 1. Login credentials are incorrect 2. The url is incorrect
Priority:	Low priority, needs to be implemented after all the basic functionalities are implemented
When Available:	Second release