Use Case:	Login
Primary Actor:	Faculty, TA, Admin, Committee Member
Goal in Context:	To login to the application to perform plagiarism check or/and view reports or/and download reports.
Preconditions:	The user is registered and has a valid user ID and password.
Trigger:	The user wants to use the application to perform plagiarism check or/and view reports or/and download reports.
Scenario:	1. The user opens the welcome page for the application. 2. The user enters his/her user ID and password. 3. Application verifies user id and password. 4. Grants access if the user is valid. 5. Returns error for incorrect user id and/or password.
Exceptions:	1. User ID and password is invalid. The user should request registration if unregistered. See 'Request Registration/Enrollment' use case. 2. User is unable to recall the correct user ID and/or password. The user should send an access reset request to the Admin. See 'Forgot Password' use case.

The following fields are optional	
Priority:	High priority since authentication is crucial to prevent unwanted access to the software and is the portal to the software.
When available:	Sprint 4/1st increment
Channel toactor:	Via PC-based or mobile based browser with internet connection.
Secondary Actor:	Admin
Channels to Secondary Actors:	Via PC-based or mobile based browser with internet connection.
Open Issues:	Mechanisms to prevent unauthorized entry.

Use Case:	Request Registration/Enrollment
Primary Actor:	Faculty, Committee Member
Goal in Context:	To allows users(super) to file a new user request upon which it will be put into Admin's queue for processing.
Preconditions:	User has correct User ID/Password to access the system, corresponding details about the user to be added.
Trigger:	The user wants to add another user to the application to perform plagiarism check or/and view reports or/and download reports.
Scenario:	1. The user opens the Request Registration page. 2. The user enters all the details like First Name, Last Name, Email id, chooses a username and password. 3. The user clicks register. 4. The request is sent to the Admin for approval.
Exceptions:	1. User ID or password are not recognized – see use case for 'Request Registration' or 'Forgot Password'.

The following fields are optional	
Priority:	High priority as it enables users to gain access to the application.
When available:	Sprint 4/1st increment
Channel toactor:	Via PC-based or mobile based browser with internet connection.
Secondary Actor:	Admin.
Channels to Secondary Actors:	Via PC-based or mobile based browser with internet connection.
Open Issues:	Automated Authentication

Use Case:	Approve Registration/Enrollment Request
Primary Actor:	Admin
Goal in Context:	To approve the enrollment request made by a faculty/committee so that other users can use the application.
Preconditions:	The Admin receives the request for enrollment; appropriate details are provided; System is fully functional.
Trigger:	Faculty/Committee members file a register request for other users so that they can use the application.
Scenario:	1. The admin logs on to Plagiarism Detection application. 2. Admin enters admin login details. 3. All system details are displayed upon login. 4. The request notification appears on request dashboard. 5. The admin selects the request. 6. Admin confirms the identity and details 7. The admin will click the accept button.
Exceptions:	1. The request received has incorrect details- see use case – Reject Enrollment Request. 2. The request is redundant – the database throws error and the request is ignored.

The following fields are optional	
Priority:	High priority, to help the users use the system functionalities.
When available:	Sprint 4/1st increment
Channel toactor:	Via PC-based or mobile based browser with internet connection.
Secondary Actor:	Faculty/Committee members
Channels to Secondary Actors:	Via PC-based or mobile based browser with internet connection.
Open Issues:	1. Protect the admin credentials 2. What mechanism the admin uses to verify the user provided details.

Use Case:	Compare source code
Primary Actor:	Faculty, TA
Goal in Context:	Find similarities and differences by comparing between two pieces of code
Preconditions:	The faculty has the files/folders/source code to compare.
Trigger:	Faculty decides to check for plagiarism
Scenario:	1. The user logs into the Plagiarism Detection Application. 2. Then enters his/her user ID and password. 3. The user uploads either files or folders or piece of code to compare 4. The user runs a differences check. Source codes are analyzed. 5. Percentage of the code match is returned for the compared source codes along with a descriptive diff check.
Exceptions:	1. Uploaded files are corrupt, invalid. 2. Uploaded files/source codes are not compatible in terms of the source code language in which case an error will be thrown asking user to provide correct set of inputs.

The following fields are optional	
Priority:	High priority as it is the brain of the application
When available:	Sprint 5/2nd increment
Channel toactor:	Via PC-based or mobile based browser with internet connection.
Secondary Actor:	-
Channels to Secondary Actors:	-
Open Issues:	Algorithm to use for comparing the 2 pieces of code

Use Case:	Download report
Primary Actor:	Faculty, Committee Member
Goal in Context:	To allow users to download plagiarism report
Preconditions:	The source codes have been compared successfully and the plagiarism percentage is already calculated.
Trigger:	The user wants to download the report for the latest source code comparison.
Scenario:	1. The user logs into the Plagiarism Detection Application. 2. Then enters his/her user ID and password. 3. User uploads the files to be checked. 4. User runs source code comparison. 5. Plagiarism report is generated. 6. User downloads the report.
Exceptions:	Download fails – ask user to retry

The following fields are optional	
Priority:	Low Priority, to be implemented as nice-to-have functions
When available:	Sprint 5/2nd increment
Channel toactor:	Via PC-based or mobile based browser with internet connection.
Secondary Actor:	-
Channels to Secondary Actors:	-
Open Issues:	To work on the formats supported – PDF, Excel, docx

Use Case:	Send Email Notifications
Primary Actor:	Faculty
Goal in Context:	Send a notification to committee member and/or student with plagiarism report
Preconditions:	Plagiarism report has been generated
Trigger:	The user decides to notify Committee member and/or student when high match is found
Scenario:	1. The user logs into the Plagiarism Detection application. 2. Then enters his/her user ID and password. 3. The user runs comparison for source code 4. Report has been generated. 5. User has an option to notify student without report via email. 6. User has an option to notify committee member with report via email.
Exceptions:	1. User receives error while sending an email and can manually send the report.

The following fields are optional	
Priority:	Low Priority, to be implemented as nice-to-have functions
When available:	Sprint 6/3rd increment
Channel toactor:	Via PC-based or mobile based browser with internet connection.
Secondary Actor:	-
Channels to Secondary Actors:	-
Open Issues:	Mechanism to send email notifications

Use Case:	Update user
Primary Actor:	Faculty/Admin/Committee Members
Goal in Context:	To update existing user’s details like username, password, name etc.
Preconditions:	Appropriate user ID and password are present for a user
Trigger:	The user decides to update his/her details.
Scenario:	1. The user logs into the application. 2. Then enters his/her user ID and password. 3. Upon successful login, the system displays user menu options 4. The user selects ‘Profile’ from the menu options. 5. The system displays existing profile details to the user. 6. The user selects ‘Update my Profile’ option from the page. 7. The system displays editable fields which user can edit. 8. User updates the fields and clicks on ‘Save’ button. 9. System asks user to confirm the action. 10. Upon selecting ‘Yes’, user’s profile is updated, and success message is displayed.
Exceptions:	1. User ID or password are not recognized – see use case for ‘Request Registration’ or ‘Forgot Password’. 2. User clicks ‘No’ on update confirmation upon which user profile is not updated

The following fields are optional	
Priority:	Low Priority, to be implemented as nice-to-have functions
When available:	Sprint 6/3rd increment
Channel toactor:	Via PC/Mobile based browser and Internet connection
Secondary Actor:	-
Channels to Secondary Actors:	-
Open Issues:	Is there any mechanism to protect unauthorized access to the system which prevents malicious updates?

Use Case:	Delete user
Primary Actor:	Admin
Goal in Context:	To delete existing user in the system.
Preconditions:	Admin must be logged in, user to be deleted must be present in the system
Trigger:	Admin receives a delete user request in his/her queue.
Scenario:	1. The admin logs into the application. 2. Then enters his/her user ID and password. 3. Upon successful login, the system displays admin menu options 4. The admin selects 'Delete' from the menu options. 5. A dropdown is displayed with existing users in the system. 6. The admin selects relevant user from the dropdown. 7. The admin clicks on 'Delete' button to proceed. 8. System asks admin to confirm the action. 9. Upon selecting 'Yes', user is deleted, and success message is displayed.
Exceptions:	1. Admin clicks 'No' on delete confirmation upon which user is not deleted.

The following fields are optional	
Priority:	Low Priority, to be implemented as nice-to-have functions
When available:	Sprint 6/3rd increment
Channel toactor:	Via PC/Mobile based browser and Internet connection
Secondary Actor:	Faculty
Channels to Secondary Actors:	Via PC/Mobile based browser and Internet connection
Open Issues:	-

Use Case:	Request to delete a user
Primary Actor:	Faculty/Committee Members
Goal in Context:	To allows users(super) to file a terminate user request upon which it will be put in Admin’s queue for processing
Preconditions:	User has correct User ID/Password to access the system, corresponding details about the user to be deleted
Trigger:	The user decides to file a request to terminate another user
Scenario:	1. The user logs into the application. 2. Then enters his/her user ID and password. 3. Upon successful login, the system displays user menu options 4. The user selects ‘Admin Request’ option from the menu options. 5. The system displays existing set of users assigned under the logged in user along with the option to terminate them 6. The user selects one or more of the users from the page to delete. 7. User and clicks on ‘Delete’ button. 8. System asks user to confirm the action. 9. Upon selecting ‘Yes’, delete request is put into admin’s queue and success message is displayed. See ‘Delete User’ use case.
Exceptions:	2. User ID or password are not recognized – see use case for ‘Request Registration’ or ‘Forgot Password’. 3. User clicks ‘No’ on delete confirmation upon which delete request is cancelled.

The following fields are optional	
Priority:	Low Priority, to be implemented as nice-to-have functions
When available:	Sprint 6/3rd increment
Channel toactor:	Via PC/Mobile based browser and Internet connection
Secondary Actor:	Admin
Channels to Secondary Actors:	Via PC/Mobile based browser and Internet connection
Open Issues:	How to handle delete user requests in the form of access control? e.g. Some hierarchy which will control the request flow like Committee -> Professor -> TA

Use Case:	Forgot Password.
Primary Actor:	Faculty, TA, Committee Member.
Goal in Context:	The user has forgotten the password and wants to reset the password.
Preconditions:	The user is registered and has a valid user id and password.
Trigger:	The user does not recollect his/her password.
Scenario:	1. The user opens the application welcome page 2. The user clicks the button for Forgot password. 3. The user is then sent a link to his/her email to reset the password.
Exceptions:	1. The user does not receive the link to reset the password, the user can then contact the admin to reset the password.

The following fields are optional	
Priority:	Low Priority, to be implemented as nice-to-have functions.
When available:	Sprint 6/3rd increment
Channel toactor:	Via PC/Mobile based browser and Internet connection
Secondary Actor:	Admin
Channels to Secondary Actors:	Via PC/Mobile based browser and Internet connection
Open Issues:	1. Using security questions for recovering password