Feature: Login system with session management

Use Case: 1

Use Case:	Users registering on the portal.
Primary Actor:	Users
Goal in Context:	To register/enroll on the plagiarism detection portal.
Preconditions:	1. Must have a valid and unique email ID (should not be already present in the system).
Trigger:	The user decides to enroll himself for the plagiarism detection portal so that he can use it.
Scenario:	1. The user selects "Register as a new user" from the portal/website. 2. The user fills out the required details in the registration portal. 3. The user enters a valid password (at least 8 characters in length) and verifies it. 4. The user then submits the form to complete the registration. 5. If successful, the user is taken back to the index (login) page. 6. If unsuccessful, the user is shown a popup message stating that he has entered some invalid information.
Exceptions:	1. Invalid email id format. 2. Password entered does not comply with the password requirements. 3. User with the same email address already exists in the system. 4. Form validation for all the fields(User submits an empty or incomplete form).
Priority:	Essential and must be implemented.
When available:	First increment.
Channel to actor:	Via the registration form.
Status:	Done.

Use Case: 2

Use Case:	Users logging into the portal
Primary Actor:	Users
Goal in Context:	To login into plagiarism portal
Preconditions:	1. User must be registered into the system with valid email.
Trigger:	The user decides to login
Scenario:	1. The user fills login form with valid username and password 2. User clicks login button 3. User should be able to see the dashboard.
Exceptions:	1. User not registered in the system 2. Invalid credentials
Priority:	Essential and must be implemented.
When available:	First increment.
Channel to actor:	Via the login form.
Status:	Done.

Use Case: 3

Use Case:	Users wants session to be persisted on browser
Primary Actor:	Users
Goal in Context:	To persist user session
Preconditions:	1. User must be registered into the system with valid email.
Trigger:	User wants start using app with previous login session
Scenario:	1. The user fills login form with valid username and password 2. User clicks login button 3. User should be able to see the dashboard. 4. User closes the tab 5. User opens the system in another tab. 6. User should be able to see dashboard instead on login page
Exceptions:	1. Session is cleared in the browser
Priority:	Essential and must be implemented.
When available:	First increment.
Channel to actor:	Browser
Status:	Done.

Feature: Admin operations

Use Case: 4

Use Case:	Adding admin users.
Primary Actor:	Admin
Goal in Context:	To add new admin user to the plagiarism detection portal.
Preconditions:	1. System must have initial admin credentials which is created when the system is started.
Trigger:	Admin decides to add another admin.
Scenario:	1. Login using valid admin credentials. 2. Click on add Account Manage. 3. Fill in the valid details for add Admin user. 4. Add admin user is displayed. 5. Click on the add admin. 6. Admin successfully added message will be displayed.
Exceptions:	1. Invalid email id format. 2. Password entered does not comply with the password requirements. 3. User with the same email address already exists in the system. 4. Form validation for all the fields(User submits an empty or incomplete form).
Priority:	Essential and must be implemented.
When available:	Third Increment.
Channel to actor:	Via the registration form.
Status:	Done.

Use Case: 5

Use Case:	Removing users.
Primary Actor:	Admin
Goal in Context:	To remove the user from plagiarism portal.
Preconditions:	1. Must have valid registered user.
Trigger:	Admin decides to remove a user from the system.
Scenario:	1. Login using valid admin credentials. 2. Click on add Account Manage. 3. Fill in the valid details for remove user. 4. Click on search. 5. If user present in the system, user details will be displayed in the search result. 6. If user is removed successfully ‘user successfully removed from the system’ message will be displayed.
Exceptions:	1. User which is not present in the system.
Priority:	Essential and must be implemented.
When available:	Third Increment.
Channel to actor:	Via user removal form.
Status:	Done.

Feature: Plagiarism run

Use Case 6:

Use Case:	Users perform plagiarism check.
Primary Actor:	Users
Goal in Context:	Upload students solutions/files to the system.
Preconditions:	1. The user must be logged into the system. 2. The users must have valid github urls. 3. The permission to github repository must be set to pull and run plagiarism check.
Trigger:	The user wants to runs plagiarism check by providing github urls.
Scenario:	1. User logs into the system. 2. Click on Add run. 3. For each student the user will provide a git repository by clicking on add student and provide the student name and respective git repository. 4. Provide description for the run. 5. Click on Run.
Exceptions:	1. User inputs github url of invalid format. 2. User does not provide a run description. 3. User does not add student and github urls. 4. User uploads a file greater than the size limit.
Priority:	Essential and must be implemented.
When available:	Second increment.
Channel to actor:	Run interface to accept students and git information.
Open Issues:	1. Should there be another way to upload solutions?
Status:	Done.

Use Case 7:

Use Case:	Advanced plagiarism run.
Primary Actor:	Users
Goal in Context:	Run plagiarism with advanced settings.
Preconditions:	1. The user must be logged into the system. 2. The users must have valid github urls. 3. The permission to github repository must be set to pull and run plagiarism check.
Trigger:	The user wants to run advanced plagiarism check by providing github urls.
Scenario:	1. User logs into the system. 2. Click on Add run. 3. For each student the user will provide a git repository by clicking on add student and provide the student name and respective git repository. 4. Provide description for the run. 5. Click on advanced button. 6. Adjust weight strategies as required. 7. Click on Run.
Exceptions:	1. User inputs github url of invalid format. 2. User does not provide a run description. 3. User does not add student and github urls. 4. User uploads a file greater than the size limit.
Priority:	Essential and must be implemented.
When available:	Second increment.
Channel to actor:	Run interface to accept students and git information.
Open Issues:	
Status:	Done.

Feature: Plagiarism Report

Use Case 8:

Use Case:	View the plagiarism run report.
Primary Actor:	Users
Goal in Context:	View the report of the plagiarism run that is performed
Preconditions:	1. The user must be logged into the system. 2. The users must have valid github urls. 3. The permission to github repository must be set to pull and run plagiarism check.
Trigger:	The user wants to check the results of the performed runs.
Scenario:	1. User logs into the system. 2. Click on Add run. 3. For each student the user will provide a git repository by clicking on add student and provide the student name and respective git repository. 4. Provide description for the run. 5. Click on Run or go through the run history. 6. The report is generated which displays detailed report about plagiarism among the source provided. 7. Difference between the files can also be seen in the git diff tab
Exceptions:	1. Report no longer present in the back end.
Priority:	Essential and must be implemented.
When available:	Third increment.
Channel to actor:	User dashboard.
Open Issues:	
Status:	Done.

Use Case 9:

Use Case:	Check history of plagiarism runs.
Primary Actor:	Users
Goal in Context:	Check previously run plagiarism checks.
Preconditions:	1. The user must be logged into the system.
Trigger:	The user wants to check results of previously run plagiarism.
Scenario:	1. User logs into the system. 2. Click on various respective plagiarism report from server.
Exceptions:	1. Report no longer present on the back end.
Priority:	Essential and must be implemented.
When available:	Second increment.
Channel to actor:	User dashboard.
Open Issues:	
Status:	Done.

Feature: System Statistics

Use Case: 8

Use Case:	System Information.
Primary Actor:	Admin
Goal in Context:	Display system information to admin.
Preconditions:	1. Must have valid admin credentials
Trigger:	Admin wants to view complete system information.
Scenario:	1. Login using valid admin credentials. 2. Click on usage info. 3. System and application details will be displayed.
Exceptions:	1. System or application is down.
Priority:	Essential and must be implemented.
When available:	Third Increment.
Channel to actor:	Via user removal form.
Status:	Done