Feature 1: Login

Status: Completed

Use Case:	User login
Primary Actor:	Professor or TA
Goal in Context:	To login successfully as Professor or TA the password should match the login credentials.
Preconditions:	The application has been initiated.
Trigger:	The actor decides to check for plagiarism.
Scenario:	1. Login as professor 2. Login as TA
Exceptions:	1. Display login failure if the login credentials are invalid: User inputs the correct credential.

Feature 2: System configurations

Status: Completed

Use Case:	Configure the plagiarism threshold and type
Primary Actor:	Professor
Goal in Context:	Allow the professor to set the Plagiarism percentage, above which the code is considered plagiarized.
Preconditions:	The system has been programmed to accept the Plagiarism percentage and decide if the files are plagiarized
Trigger:	The professor decides to set the plagiarism percentage
Scenario:	1. The person interacting with the system selects the type of user as professor. 2. Professor sets the plagiarism percentage.
Exceptions:	1. Plagiarism Percentage is less than 0 or greater than 100: Professor sets the correct percentage between 0 and 100. 2. The website does not respond: Professor waits for the website to be up and running. 3. Plagiarism limit does not set correctly: Professor retries and sets the limit again.

Feature 3: Upload submissions

Status: Completed

Use Case:	Upload the files for plagiarism detection
Primary Actor:	Professor / TA
Goal in Context:	Allow the professor/ TA to upload files that needs to be checked for plagiarism.
Preconditions:	The primary actor is successfully logged in and the plagiarism limit has been set.
Trigger:	The professor/ TA decides to upload the files
Scenario:	1. The person interacting with the system selects the type of user as professor/TA 2. The Actor uploads all the files that he needs to check for plagiarism.
Exceptions:	1. The Actor uploads the incorrect files (other than something.py): Actor uploads the correct files. 2. The Actor gives a wrong path for the files upload: Actor checks and gives a correct path.

Use Case:	Upload the files with which previously uploaded files have to be compared with.
Primary Actor:	Professor / TA
Goal in Context:	Allow the professor/ TA to upload files that needs to be checked for plagiarism.
Preconditions:	The files that are to be checked for plagiarism have been uploaded.
Trigger:	The professor/ TA decides to upload the files
Scenario:	1. The person interacting with the system selects the type of user as professor/TA 2. The Actor uploads all the files that he needs to compare the files with for plagiarism detection
Exceptions:	1. The Actor uploads the incorrect files (other than something.py): Actor uploads the correct files. 2. The Actor gives a wrong path for the files upload: Actor checks and gives a correct path.

Feature 4: Plagiarism report

Status: Completed

Use Case:	Check for plagiarism
Primary Actor:	Professor / TA
Goal in Context:	The actor can check for plagiarism percentage of the files he uploaded.
Preconditions:	The primary actor is successfully logged in, the plagiarism limit has been set and the files to be checked for plagiarism have been uploaded successfully.
Trigger:	The professor/ TA decides to check for plagiarism.
Scenario:	1. The person interacting with the system selects the type of user as professor/TA 2. The Actor uploads all the files that he needs to check for plagiarism. 3. The Actor clicks on check for plagiarism button.
Exceptions:	1. The website becomes unresponsive: The Actor refreshes the page and starts again.

Use Case:	Verify the plagiarism report of every pair provided.
Primary Actor:	Professor / TA
Goal in Context:	The actor can check for the exact lines in the code that matched and contributed towards the plagiarism percentage generated.
Preconditions:	Successful generation of the plagiarism report.
Trigger:	The professor/ TA decides to verify the report generated.
Scenario:	1. The report and matching files are generated, and the actor decides to verify and check the code for plagiarism.
Exceptions:	1. The verification step is not ready: Actor checks at the uploaded files and tries again.

Status: Pending

Use Case:	Save the plagiarism report
Primary Actor:	Professor / TA
Goal in Context:	The actor saves the generated plagiarism report to a local drive
Preconditions:	Successful generation of the plagiarism report.
Trigger:	The professor/ TA decides to save the report generated.
Scenario:	1. The report and matching files are generated 2. The actor decides to save the report 3. The actor clicks on the "Save" button 4. The actor is prompted to select the destination folder and add the file name. 5. File is saved at the selected destination
Exceptions:	1. The Actor gives a wrong path for the files save location: Actor checks and gives a correct path. 2. The filename given by the Actor already exists: Actor renames the file and tries to save the file again.