PLAGIARISM DETECTOR

DARSHAN PANSE, SAMANJATE SOOD, SHAIL SHAH, AND VAIBHAV DAVE

1. Project Statement

Our goal is to create a prototype that performs the function of plagiarism detection, by comparing the source code of two C programs. The Plagiarism Detector is a desktop based application, where the front-end is implemented using JavaFx, and the core system logic is written in Java. We are planning on using MySQL to persist user related data if needed.

The software program so created would have the functionality to compare the two software programs, whose source code is written in C, and inform the user about the similarities that may exist between the program structures. It will also specify the degree of similarity that would exist, and it will be expressed as a form of coloured labels for each pair of student.

The two primary actors within the software are given below:
- Instructor
- Teaching Assistant

The interaction between the actors and the system would follow a simple procedure. To start using the software, the user will have to run the desktop application. This could either be done via double-clicking a GUI based icon, or through a cmd/terminal based command, depending on the Operating System.

Once the application is running, the two actors will be able to upload multiple directories corresponding to the assignment submission of the students in class; the instructors wishes to check for plagiarism. The tool will then filter out the programs written in C in each directory, and associate them with the student and the assignment number. The association will be based on the directory name and the drop-down list selection by the user. The user will then be able to run the software on these directories and display the result, along with the level of similarities between two submissions; and those will be marked.

The users can then generate a report for the program output. The two users are differentiated by the following functionalities.
- Only the instructor will be able to reveal names of the suspicious students. This is done to prevent any bias that the Teaching Assistant might have towards the students.
- The TAs will only be able to send an email to the instructor, whereas the instructor will have additional functionality to send emails to the alleged students as well as OSCCR.

We have provided further detailed steps for interacting with the system and they are mentioned in the use-case document, and along with, they are pictorially represented the corresponding mock-up UIs for easy reference.

We have decided to divide the eight-week time-line provided for implementing the project into 4, two-week sprints. Each sprint will have their own minimum working products to be met, and each team member would work on predefined sub-tasks during the particular sprints.