Team – 104
Plagiarism Detector
Plan Document

Language for detection:
Python

Objective:

A plagiarism detecting web-based system which takes the source codes of multiple students as input, detect the similarity of these codes and decide whether plagiarism exists, and provides detail report about the similarity for users to review and print.

Users/Personas of the project:

Professors and TAs in universities who want to detect the code-plagiarism among students. Professors and TAs are different type of users. Professors can configure the detecting details of the system (like threshold of plagiarism, the types of plagiarism to detect…), while TAs can only upload the files of two students, detecting the similarity between codes and reviewing the report.

Solution Concept:

1) Types of plagiarism to detect: moving functions or methods to another location in the same file, breaking functions into separate files, renaming variables, extracting sequences of statements into methods, sharing same comments.
2) Algorithm we might use: AST, token
3) Back-end: Java
4) Front-end: HTML, CSS, Angular.js, Bootstrap
5) Version control: git

Project plan:

We will deliver the project in span of 8 sprints in the intervals of 1 week.

Sprint 1:	Analysis of the requirements, get to know the mock UI's.
Sprint 2:	Design the UML diagram and decide on the structure.
Sprint 3:	Design the UI of the system and accept the inputs.
Sprint 4:	Build the classes and start with the backend work.
Sprint 5:	Put the algorithm into work and calculate the plagiarism percent.
Sprint 6:	Integrate with the front end and start writing test cases.
Sprint 7:	Pre- deployment checks and carry out deployment process.
Sprint 8:	Take feedback and improve on the final product.