Plagiarism Detection in Python Source Code

Managing Software Development
Team-202
Fibin Francis Assissi | Sidharth Thapar | Hao Wu

Objective:

Purpose of a Plagisrism Detector:
To help instructors detect situations where two or more students submit similar solutions to an assignment, in which one version derives from another version through one or more behavior-preserving transformations.

Target Language: Python
Python is our obvious choice being the easiest and most powerful Imperative Programming Languages. With a large pool of libraries and active development community, a lot of students choose python as their first preference to make projects. Its popularity in the Machine Learning, Data Science and Web Development communities also motivated us to choose it.

Implementation:

AST Method:
1) Generate an Abstract Syntax Tree(AST) from the two source code folders.
2) Split the AST into subtrees(subtree is a complete semantic structure).
3) Calculate the similarity between the 2 programs by comparing the subtrees of two programs, using **Longest Common Subsequence** Method.

Technology Stack:

Front-End	AngularJS/ ReactJS
Backend	Java (Spring Boot)
Database	MySql/MongoDB
Deployment	AWS

ROADMAP:

List of Tasks to be completed

1) Requirement gathering, research on the implementation and plan use cases for the application along with architecture
2) Create UML diagram, and plan the overall implementation
3) Setting up the environment, start with basic functionality implementation
4) Complete release 1, with all primary use cases
5) Finish all development work and do testing in parallel, implement lesser priority use cases
6) Finish integration testing and get feedback from other teams
7) Complete development and testing of the entire application
8) Final presentation

REFER to Roadmap.xls for complete ROADMAP