Plagiarism Detection Tool
Team 206

Team Members
Mohd Asim Khan
Sanket Mathur
Jayanth Gangadhar
Jiaxin Yang

Problem Statement:

Plagiarism is a statement that someone copied code deliberately without attribution. In academia, programming assignments are used to evaluate students in programming courses. Therefore, it is a necessity to ensure a student's submission is novel. However , the task of manually checking individual submissions is very tedious for the instructors and hence the task of plagiarism detection must be automated.

Project Description

Plagiarism Detection Tool is a web-based single-page application that will be used by instructors to find out similarities between the python source codes submitted by students.

- The items to be compared can be a single program or a whole set of directories containing multiple source-code files.
- The instructor will also have a feature to compare projects submitted by two students or he can directly generate a report for similarity found among all the projects submitted by all students in the class in one go.
- The application will be capable of inspecting Python source-code only.

Technologies To be Used:

FrontEnd Development	HTML, CSS, ReactJS/AngularJS 1.x, Bootstrap
BackEnd Development	J2EE, Spring Framework
Database	MongoDB
Version Control System	Git
Continuous Integration Tool	Jenkins
Project Build Tool	Maven
Deployment Tool	AWS

Issue Tracking Tool	Git
Architecture	MVC

Project Plan:

Week 1	● Phase A documentation ● Writing Use Cases ● Creating mockups and wireframes for the User Interface
Week 2	● Designing UML diagrams ● Design Flow Diagrams ● Design Interaction Diagrams
Week 3	● Setting up environment for development ● Committing first set of configuration files and build scripts ● Integrating the application with AWS to get it deployed ○ Creating EC2 instance ○ Setting up DB on AWS DB or mlabs ● Basic home Page deployed to check if application is deployed successfully ● Installing continuous integration tool (JENKINS) to pull commited code directly from the repository
Week 4	Code Build ● Authentication - building sign in, sign up and logout part of the application.
Week 5	Code Build and unit testing ● Authorization of the users. Functionality to upload the files/folders and all the other options where this application can get input files and folders. ● Apply the plagiarism algorithm ● Write unit test cases
Week 6	Code build and unit testing ● Apply the plagiarism algorithm ● Building the comparison report ● Writing test cases

	● Write unit test cases
Week 7	Code build and unit testing Writing test cases ● Building the comparison report ● Building reports that admin can run on the system which are specific to the admin roles ● Writing test cases
Week 8	● Integration testing ● System testing