Following are the interfaces for major modules in our plagiarism detector.

User:

It represents users in our system who will be using the plagiarism detector. The user could a Faculty or an Admin. They will have common functionality to initialize during login and clean up during logout.

Faculty:

Faculty represents users who will use the plagiarism detector for grading student's assignment. They will have access to the course and assignments submitted by the students and functionality to run plagiarism detector on the submissions.

Submission:

This represents submission made by the students. These submissions can be later used to generate source and use them in the plagiarism detector.

Source:

Source represents the project or submission source on which plagiarism should be run. There cloud be different kinds of sources like Github source and uploaded files as source. Source handles the responsibility to maintain the source files which are later used to generate Abstract Syntax Tree.

ASTNode:

This represents each node in the Abstract Syntax Tree. There could different kinds of nodes like Expresssion and Statement. Each file is parsed and the content is represented as an ASTNode.

Expression:

Expression is a type of ASTNode. There could be different kinds of expressions like integer expression, string expression and variable Expression.

Statement:

Statement is a type of ASTNode. There could be difference kinds of statements like declaration, sequence and assignment.

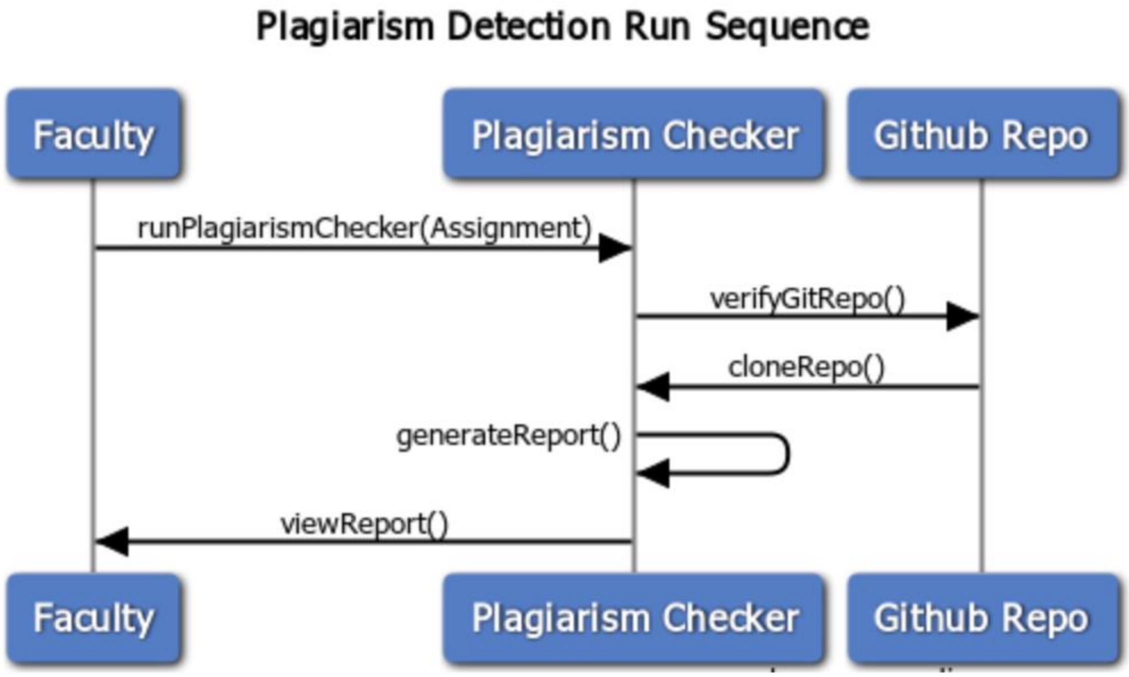
ASTComparator:

This is used to compare two different Abstract Syntax Trees. There could be different comparisons logic where one could be efficient than the other when used in different scenario.

ASTIterator:

This iterator is used to traverse the AST tree. There could be different iterators for AST trees which could be used in different scenarios.

Since we are using git for submission below is the sequence diagram of how the system will interact with git and user.


Plagiarism Detection Run Sequence
Faculty
Plagiarism Checker
Github Repo
runPlagiarismChecker(Assignment)
verifyGitRepo()
cloneRepo()
generateReport()
viewReport()
Faculty
Plagiarism Checker
Github Repo