

Project Plan

Brief:

We will be building a software plagiarism detection tool for Python3. That does individual file as well as multiple file analysis. This will be a web-based application, mainly using HTML, CSS and JavaScript. There will be a maximum of 3 roles, the teacher, the student and admin. The users role will influence their interaction with the system and access to services.

1. The first page will be a login page. The user may enter a username and password here (or do a Guest Login).
2. Depending on the kind of role associated with the account the next page will vary for all users.

We will build the system in multiple phases. Starting with UI and iteratively adding new features with unit and integration testing in each phase.

The python ast library will be used to generate Abstract Syntax Trees. ASTs for both documents will then be compared using software metrics and comparison algorithms. The system will be robust to coding styles, naming conventions and abstractions. In general, it will be able to recognize program characteristics that are inherently difficult to change.

Post comparison the probability of plagiarism will be reflected on the screen. The user will be given the option to download the report and view the areas of similarity. Code fragments that are similar will be mentioned for the user to review.

Roadmap:

Iteration 1: Requirement gathering, figuring out Use cases and deciding on Technology.

Iteration 2: UML, high-level architecture and starting with UI (Login/Main page). Deciding on algorithm for plagiarism detection.

Iteration 3: Develop algorithm to generate AST and compare files. Unit Test for UI. Checking ways to temporarily save and retrieve files for comparison.

Iteration 4: Scale out algorithm to accommodate projects. Unit testing for single-file comparisons. Working on adding features to UI.

Iteration 5: Integrate front-end and back-end. Black-box testing.