

OVERVIEW

Design, develop, test and implement an application to facilitate the instructors to detect plagiarism among students submitting assignment. The submitted assignments is compared based on textual similarities, moving similar functions or methods to another location in the same file, renaming variables, classes, and methods, extracting sequences of statements into methods, etc.

PLAN

Week	To Do
Week 1 Planning	<ol style="list-style-type: none">1. Create an outline plan for the entire project2. Decide the programming language to detect for plagiarism3. Create a mockup UI
Week 2 Environment Setup and Research	<ol style="list-style-type: none">1. Research on the algorithms that may be implemented for detecting plagiarism2. Research on the various libraries that may be used to parse the file3. Decide the technologies to implement4. Setup local environment (Java, Database)
Week 3 & 4 UML Design and Interfaces	<ol style="list-style-type: none">1. Design the UML class diagrams2. Decide and define the interfaces to implement
Week 5 & 6 Backend and Frontend Implementation	<ol style="list-style-type: none">1. Add JUnit test cases2. Implement parser library3. Implement a plagiarism detection algorithm4. Complete necessary backend methods5. Implement basic frontend features
Week 7 Testing and delivery of the prototype	<ol style="list-style-type: none">1. Test JUnit test cases and fix bugs2. Test the prototype and fix bugs3. Deploy the prototype and deliver
Week 8 & 9 Backend and Frontend enhancement	<ol style="list-style-type: none">1. Optimize algorithm, methods, features of the system2. Enhance the UI
Week 10 Testing and final delivery of product	<ol style="list-style-type: none">1. Perform tests and fix bugs2. Deploy and deliver the project3. Present the project to professor and TAs

Targeted programming language: C