# Information Retrieval(CS F441)- Assignment 1

# Plagiarism Checker

**Team Members:**

1) **Venkateshwar Dhari Singh(2018A7PS0246H)**
2) **Dhruv Maheshwari(2018A7PS0170H)**
3) **Jatin Arora(2018A7PS0551H)**

## Abstract:

Plagiarism is considered a form of intellectual theft and fraud. It involves using someone else's words or ideas and passing them off as your own by not providing credit, either deliberately or accidentally.

Nowadays, it is very common that students and other people copy others works without even using citation.

This application uses the dataset(a corpus of documents) which will be used to detect plagiarism by using Containment measure technique/.

## Brief Overview of the steps followed for building this application:

1) **Modules Imported:** Natural language Toolkit(NLTK)

NLTK is an open source tool which provides various tokenizers, lemmatizers, stemmers, taggers etc.

**2)Reading the original and suspicious documents:**

After importing the modules we read the original document set and the suspicious document.

**3)Pre Processing for all documents:**

In the pre processing step:

- Words are tokenized using word tokenizer
- After tokenizing, all words are converted to lowercase letters
- Then all stop words are removed from the set of lowercase letters.

**4) Calculating the no of overlapping trigrams in the two texts:**

The plagiarism content between the two texts is found by calculating the Jaccard similarity coefficient,

J(A,B)= (Intersection of S(A) and S(B)) / (Union of S(A) and S(B) )

S(A)- set of trigrams in suspicious trigrams
S(B)- set of trigrams in original document

The containment measure C is a better metric for or document pairs with varied document lengths. Here, we normalize by the trigrams in the suspicious document only. C is given by,

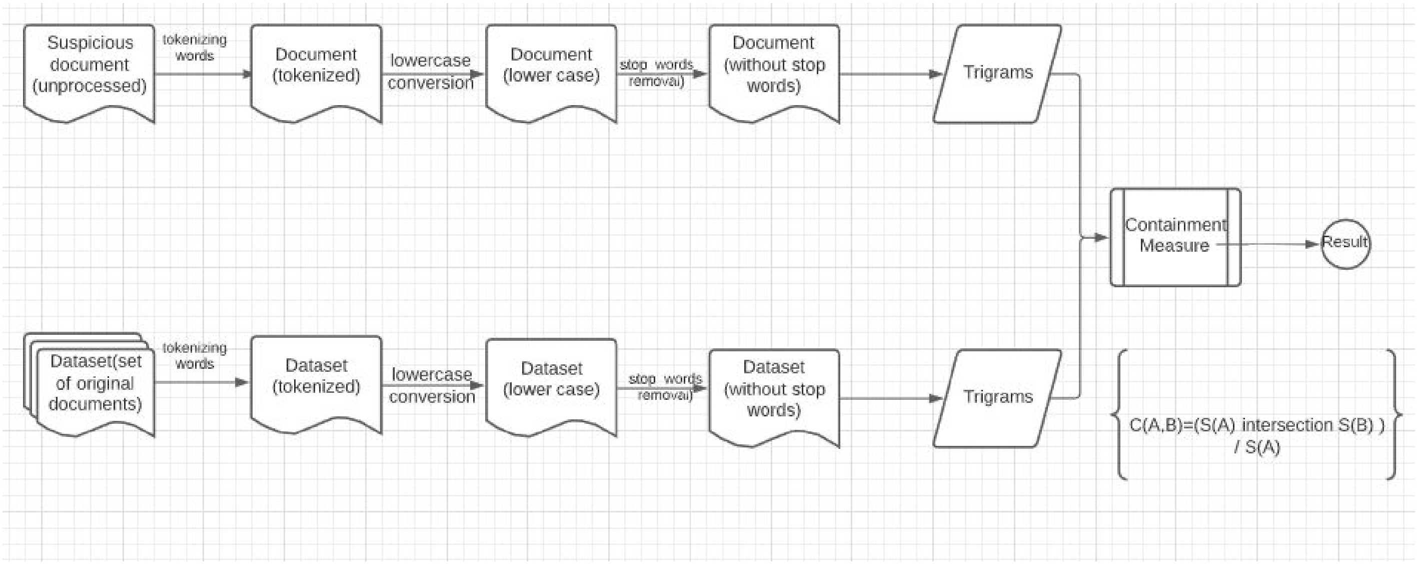
**C(A,B)= (Intersection of S(A) and S(B)) / S(A)**

S(A)- set of trigrams in suspicious trigrams
S(B)- set of trigrams in original document

*We used containment measure to detect amount of similarity between the documents.*

## Design and architecture:

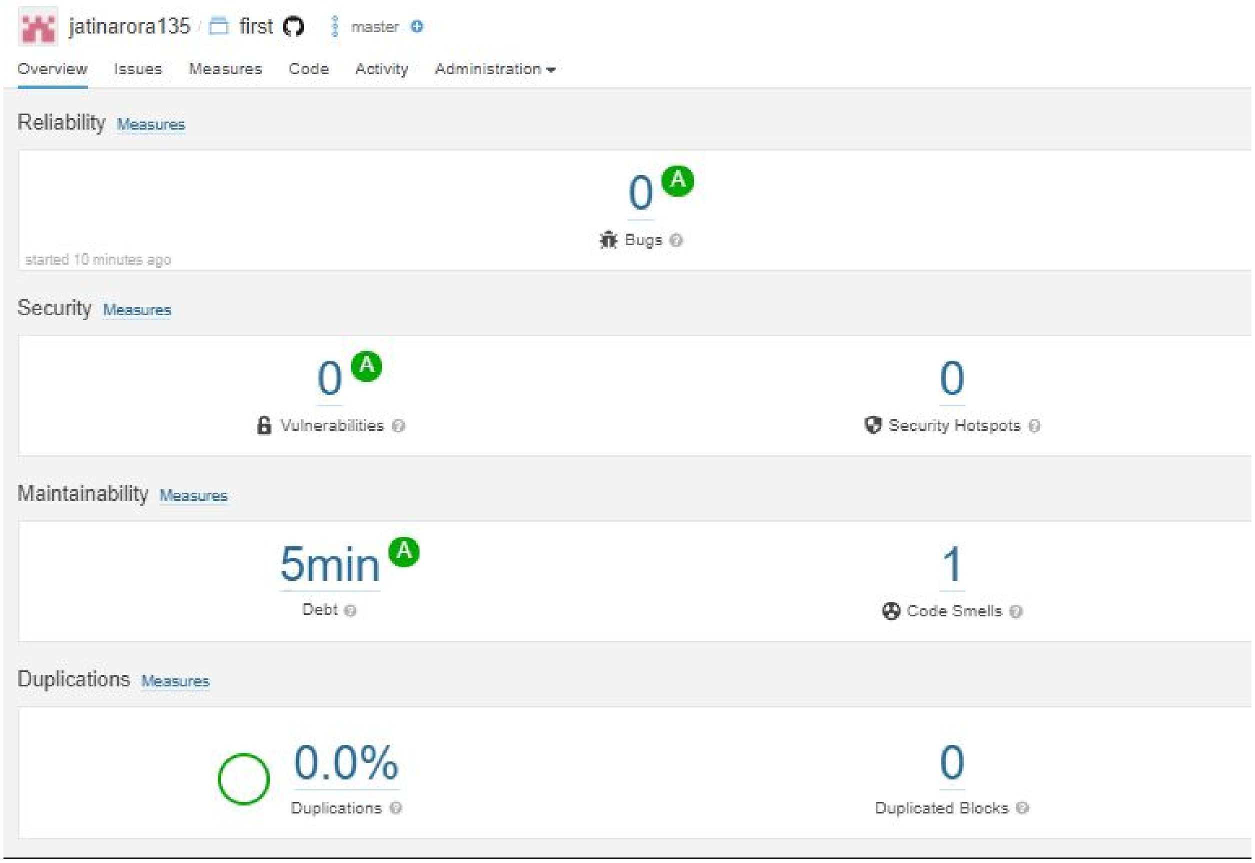
Below is the pictorial view of the design document of our project:



# Code Quality Reports:

We have also generated code quality reports using Sonarcloud to test the quality of our project

**All the files passed with ‘A’ grade during testing the code quality**



Reliability Measures

0 A
Bugs

started 10 minutes ago

Security Measures

0 A
Vulnerabilities

0
Security Hotspots

Maintainability Measures

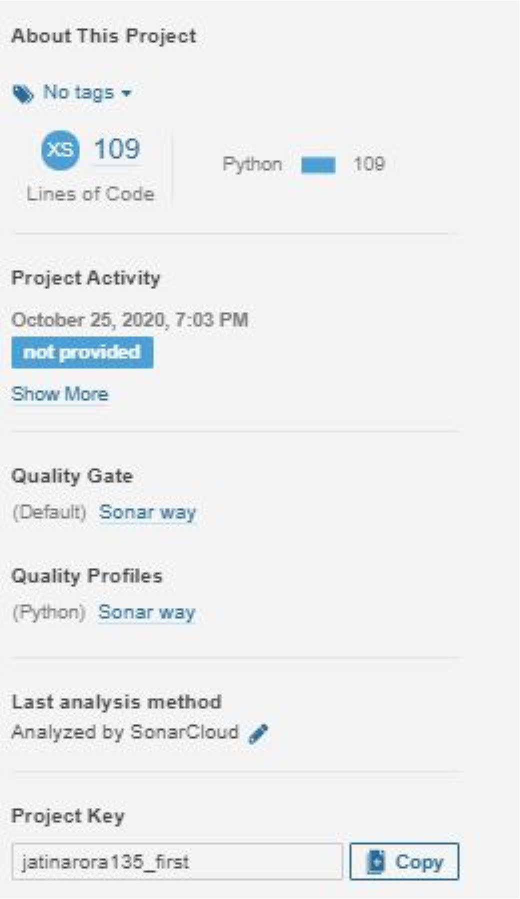
5min A
Debt

1
Code Smells

Duplications Measures

0.0%
Duplications

0
Duplicated Blocks



About This Project

No tags ▾

XS 109

Lines of Code

Python 🟦 109

Project Activity

October 25, 2020, 7:03 PM

not provided

Show More

Quality Gate

(Default) Sonar way

Quality Profiles

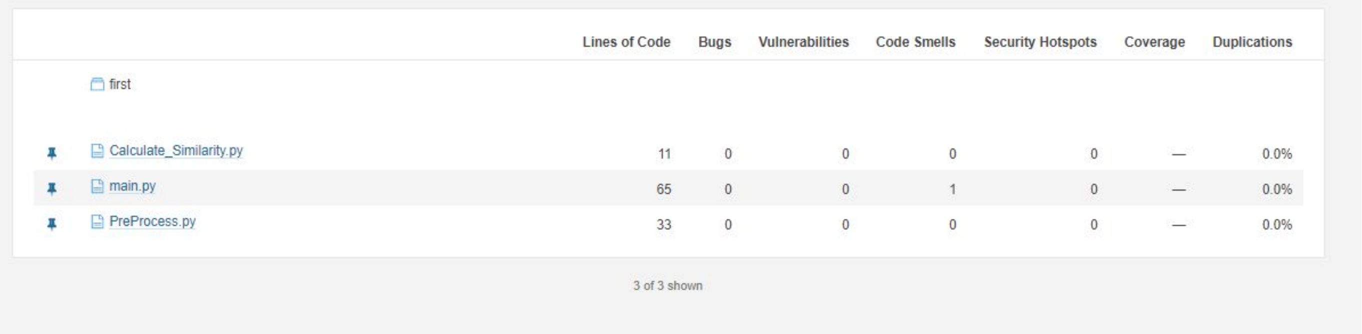
(Python) Sonar way

Last analysis method

Analyzed by SonarCloud

Project Key

jatinarora135_first



| | Lines of Code | Bugs | Vulnerabilities | Code Smells | Security Hotspots | Coverage | Duplications |
|---|---|---|---|---|---|---|---|
| first | | | | | | | |
| Calculate_Similarity.py | 11 | 0 | 0 | 0 | 0 | — | 0.0% |
| main.py | 65 | 0 | 0 | 1 | 0 | — | 0.0% |
| PreProcess.py | 33 | 0 | 0 | 0 | 0 | — | 0.0% |

3 of 3 shown