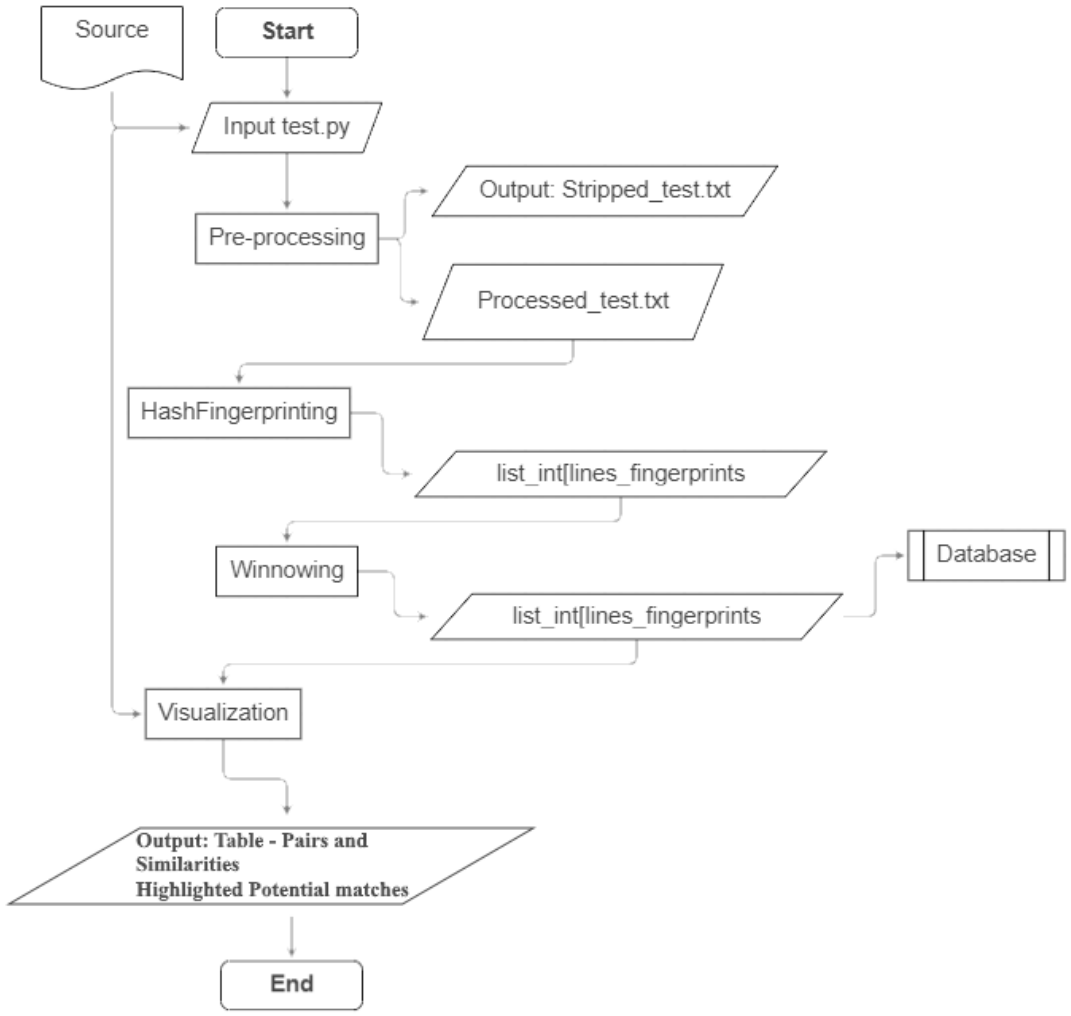
Preprocessing - Chase Jamieson
Hashing/Fingerprinting & Visual - Tracy Hotchkiss & Vinh Duong
Winnowing - Trevor Holland

Pipeline of Plagiarism Detector

Pipeline Interface



Sample of Table: Pairs and Similarities

doc_id Pair	File Similarity
doc6 - inputFile.py	97.6
doc1 - inputFile.py	97.6
doc6 - databaseFile2.py	74.79
doc2 - databaseFile1.py	74.79
doc1 - databaseFile2.py	74.79
doc2 - inputFile.py	71.43
doc3 - inputFile.py	53.6
doc6 - databaseFile3.py	50.79
doc3 - databaseFile1.py	50.79

Sample of Highlighted Potential Match's

```
File1 source code:
def numberFunction():
    a = 1 + 2
    b = a + a
    c = b + 1
    return c

def stringFunction(input1, input2):
    string1 = "abcd"
    string2 = "cda"
    string3 = string1 + string2
    return string2

def doesNothing():
    a = 1
    b = "b"
    c = True

def arrayAddition(a):
    a = a + [1]


numberFunction()
stringFunction("a","b")
doesNothing()
arrayAddition([4,3,2])
Process finished with exit code 0
```

Contracts

Preprocessing	
Input: source file	Example1 (‘databaseFile1.py’)
Output: stripped source file	Example2 (databaseFile1.py_Stripped)
Output: processed source file	Example3 (Str[processed])

Example1

```
def numberFunction():
    a = 1 + 2
    b = a + a
    c = b + 1
    return c

def stringFunction(input1, input2):
    string1 = "abcd"
    string2 = "cda"
    string3 = string1 + string2
    return string2

def doesNothing():
    a = 1
    b = "b"
    c = True

def arrayAddition(a):
    a = a + [1]

numberFunction()
stringFunction("a","b")
doesNothing()
arrayAddition([4,3,2])
```

Example2

```
def numberFunction():
 a = 1 + 2
 b = a + a
 c = b + 1
 return c
def stringFunction(input1, input2):
 string1 = "abcd"
 string2 = "cda"
 string3 = string1 + string2
 return string2
def doesNothing():
 a = 1
 b = "b"
 c = True
def arrayAddition(a):
 a = a + [1]
numberFunction()
stringFunction("a","b")
doesNothing()
arrayAddition([4,3,2])
```

Example3

```
def fun1():
 var1 = 1 + 2
 var2 = var1 + var1
 var3 = var2 + 1
 return var3
def fun2(var4, input2):
 var5 = "abcd"
 var6 = "cda"
 var7 = var5 + var6
 return var6
def fun3():
 var1 = 1
 var2 = "var2"
 var3 = True
def fun4(var1):
 var1 = var1 + [1]
fun1()
fun2("var1","var2")
fun3()
fun4([4,3,2])
```

Contracts

HashFingerprinting	
Input: processed source file	Example3 ('str[processed]')
Output: fingerprints source file	Example4 (list_int[lines_fingerprints])

Example4

```
[([1], 1440), ([1], 1382), ([1], 1265), ([1], 1008), ([1], 1553), ([1], 1514),
```

Contracts

Winnowing	
Input: fingerprints source file	Example4 (list_int[lines_fingerprints])
Output: list lines and fingerprints	Example5 (list_int[lines_fingerprints])

Example5

```
[([1], 1440), ([1], 1382), ([1], 1265), ([1], 1008), ([1], 692), ([1, 2], 632),
```

Build Inverted Index For Each File in Repo

Sample of inverted index

```
{12272: [1, 15], 11793: [1], 10698: [1], 8381: [1], 5997: [1, 2], 5836: [1, 2, 11, 12], 6601: [1, 2, 11, 12, 15, 16],
```

Build Into a Corpus

Sample of Corpus

```
{12272: {'doc1': [1, 15], 'doc2': [1], 'doc3': [1, 15], 'doc5': [1], 'doc6': [1, 15], 'doc7': [1], 'doc8': [1]},
```

Query a File

Sample of Masterlist

```
{'doc1': Counter({1: 4, 5: 4, 4: 3, 2: 1, 6: 1, 7: 1}), 'doc2': Counter({1: 7, 2: 5, 5: 3, 4: 2, 3: 1, 6: 1, 8: 1}), 'doc3': Counter({1: 4, 4: 3, 5: 3, 6: 1}),
```

Output: doc_id Pair | File Similarity

doc_id Pair	File Similarity
doc6 - inputFile.py	97.6
doc1 - inputFile.py	97.6
doc6 - databaseFile2.py	74.79
doc2 - databaseFile1.py	74.79
doc1 - databaseFile2.py	74.79
doc2 - inputFile.py	71.43
doc3 - inputFile.py	53.6
doc6 - databaseFile3.py	50.79
doc3 - databaseFile1.py	50.79
doc1 - databaseFile3.py	50.79
doc3 - databaseFile2.py	34.45
doc2 - databaseFile3.py	34.45
doc7 - databaseFile2.py	32.39
doc8 - databaseFile2.py	29.67
doc7 - inputFile.py	29.58
doc7 - databaseFile1.py	29.58
doc8 - inputFile.py	28.57
doc8 - databaseFile1.py	28.57
doc7 - databaseFile3.py	28.17
doc8 - databaseFile3.py	25.27
doc5 - databaseFile4.py	17.07
doc4 - databaseFile5.py	17.07
doc5 - databaseFile2.py	14.29
doc2 - databaseFile5.py	14.29
doc6 - databaseFile5.py	10.83
doc5 - inputFile.py	10.83
doc5 - databaseFile1.py	10.83
doc1 - databaseFile5.py	10.83
doc7 - databaseFile5.py	8.45
doc5 - databaseFile3.py	8.33
doc3 - databaseFile5.py	8.33
doc8 - databaseFile5.py	6.59
doc6 - databaseFile4.py	1.22
doc4 - inputFile.py	1.22
doc4 - databaseFile3.py	1.22
doc4 - databaseFile2.py	1.22
doc4 - databaseFile1.py	1.22
doc3 - databaseFile4.py	1.22
doc2 - databaseFile4.py	1.22
doc1 - databaseFile4.py	1.22
doc8 - databaseFile4.py	0.0
doc7 - databaseFile4.py	0.0

Interface Plans:

- Codemirror
- Flask
- Dynamic HTML

Testing Plans:

- Changing Winnowing Window Size, N-Gram, Highlight Counter Parameters

- Block Detection (cosine similarity for line blocks)

- More thorough check of all hashes after winnowing