

CS5500 -
Foundations of Software Engineering -
Project Phase B - UI Mock Up

Khan, Hassan
Raphael, Samuel
Ma, Yichuan P.
Mousset, Pierre-Alexandre

I - Overview

The name of our plagiarism detection software is called “Plagus”.

Plagus is a code plagiarism detector for Python and TypeScript.

We divide our project into five different sections :

- User Interface
- Architecture
- Plagiarism detection algorithm
- Unit, integration and system testing
- Documentation

	UI/UX	Architecture	Algorithm	Testin g	Documentation
Samuel		X	X		X
Hassan	X	X		X	X
Yichuan			X	X	X
Pierre	X		X		X

II - User Interface

Landing Page

To landing page

- A navbar with two different options: Home (current) and Contact. The contact page is going to be a basic contact page mostly for technical support.
- In order to run the plagiarism detector 3 steps :
 - 1) Upload two sets files either in Python or TypeScript .
 - 2) Select the programming language between TypeScript and Python
 - 3) Run the detector.

The detector won't run if :

- The files are not in the correct format
- The files are too large
- Less than two files have been submitted

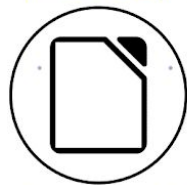
Welcome to Plagus!

Please upload two sets of files in either Python or TypeScript and then we will check for plagiarism!



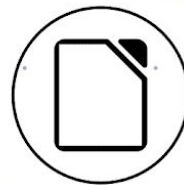
Steep 1/3 : Upload Your 2 Files

Files Set A



UPLOAD

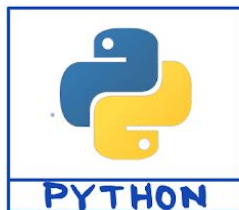
Files Set B



UPLOAD



Steep 2/3 : Select The Language



Steep 3/3 : Get Your Plagiarism Score



RUN DETECTOR

I - Result Page (Python assignment example)

PLAGUS

HOME

CONTACT

John Doe

- Search.py
- view.py
- add.py

```
import requests as rq
from bs4 import BeautifulSoup as bs
song = input("Enter a song")
# Making the parameters for the url
url_param = {"q": song.lower()}
# Making a get request
analytic_search = rq.get("http://www.google.com", param=url_param)
Soup = bs(analytic_search.text, "html")
lyric_url = search_results[0].a.get("href")
lyric_page = rq.get(lyric_url)
```

Lucia Doe

- main.py
- check.py

```
import requests as rq
for x in rq:
    if x == "test":
        new = "a"
    else:
        new = "b"
# Making a get request
analytic_search = rq.get("http://www.google.com", param=url_param)
Soup = bs(analytic_search.text, "html")
lyric_url = search_results[0].a.get("href")
lyric_page = rq.get(lyric_url)
```

PREV

54%

NEXT



report
submissions



validate
submissions