<u>Instructions To Access Resume Ranking Application:</u>

- 1) The application works with python 3.7, install a <u>virtual environment</u> for the project.
- 2) Run "pip install requirements.txt" to install all the dependencies required for the project
- 3) Once the dependencies are installed, in the project run the "resume rating main.py"
- 4) Once the file is running, it should look something like this. (Please feel free to reach out if there is trouble running the file due to any error)

```
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vivek\Desktop\resume_ranking>C:\Users\vivek\anaconda3\Scripts\activate

(base) C:\Users\vivek\Desktop\resume_ranking>c:\Users\vivek\anaconda3\Scripts\activate

(py37) C:\Users\vivek\Desktop\resume_ranking>C:\Users\vivek\anaconda3\Servs\py37\python.exe c:\Users\vivek\Desktop\resume_ranking\resume_rating_main.py

* Serving Flask app "resume_rating_main" (lazy loading)

* Environment: production

WARNING: This is a development server. Do not use it in a production deployment.

Use a production WSGI server instead.

* Debug mode: on

* Restarting with watchdog (windowsapi)

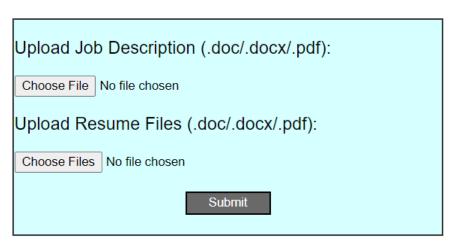
* Debugger PIN: 145-005-118

* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

- 5) Once the client is running locally, use the IP given in the last line in any local browser
- 6) Now, it should look something like this



Resume Parsing and Ranking



- 7) Choose files to upload
 - a) One for job description (works better when the file is divided into primary and secondary)
 - b) Any number for resumes (the more files, the longer to process)

8) Finally, click submit and it should give results in the following format:



Resume Parsing and Ranking

Upload Job Description (.doc/.docx/.pdf):

Choose File Java Job De..._080422.pdf

Upload Resume Files (.doc/.docx/.pdf):

Choose Files AravindhanK[12y_0m].pdf

Submit

9) The final results:

Resume Ranking according to the percentage match

1 Aravindhan Kolanjinathan Aravindhan K[12y_0m].pdf 79% Possible Match