

## **Team name and member details**

Team Name: Theme:

**CV Screening** Neptune

**Team Members: Mentor:** 

Pranav Vikhrankar Jishnu P. Nair Gaurav Rathi



## **Problem statement**

Recruiting good candidates is a very important aspect of any business. Manually selecting appropriate candidate from number of application is a tedious job for hiring process. Also, going through all the resume is a trouble some work. There is a possibility that good candidates are overlooked.



## **Objective**

- To go through all resumes.
- ▶ To match the keywords between resumes and job description.
- To shortlist appropriate candidates in an unbiased way.
- To rank the resumes on the basis of skills, education, experience.



## **Solution**

- Requirement: Job description and Resumes (in pdf or docx format)
- Converting these files data into text format.
- Data pre-processing for removal of stopwords, unnecessary symbols.
- Comparison using Cosin similarity for getting the similarity percentage.
- Storing all extracted data and similarity percentage in Mongo DB database.
- Then ranking the candidates on the basis of skills, education and experience.

Prototype submission for participants – Hackpions | EY GDS Hackathon

Display the ranking in the descending order with clean and simple UI



## Solution

#### How it helps to solve the problem?

- ✓ By considering every resume applied for job.
- ✓ Removing manual biased selection process.
- Selecting optimal candidate.

#### **Extent of scalability and usability:**

- ✓ Solution is scalable to screen thousands of resume at a time.
- ✓ Can use on any Cloud ,any OS, any browser.

#### What are the impact metrics?

Considering the previous solutions to select optimal candidate, our solution saves lot of time and also remove manual intervention.

#### How user experience will be improved:

✓ As previously, there was need to go through every resume, which is tedious job. Using our solution just upload the resumes and get the result.



## Solution

#### Limitations of the solution?

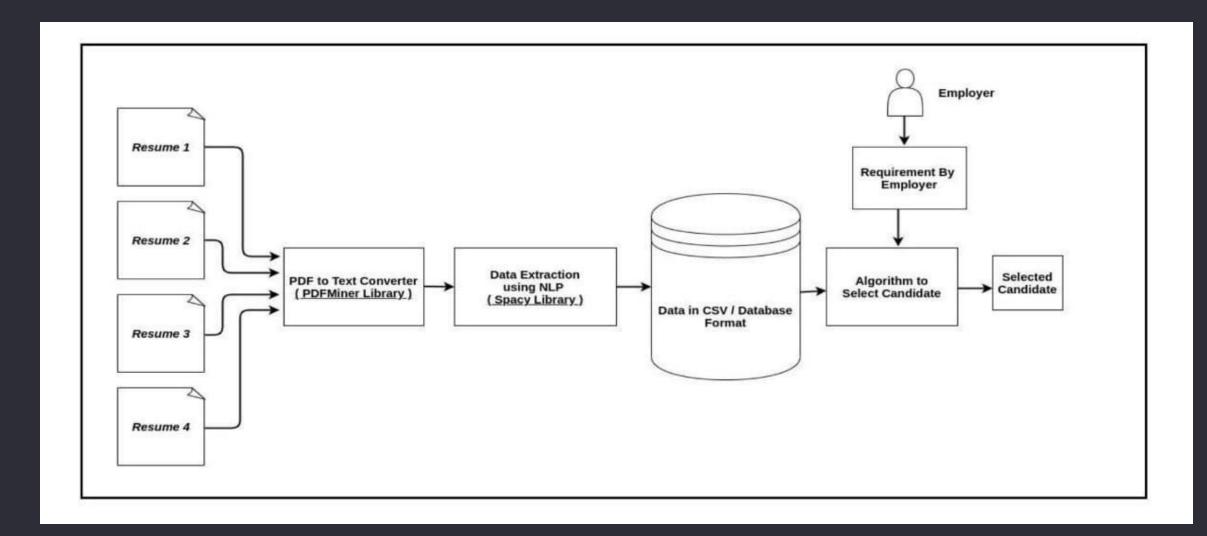
- ✓ Can extract only that skills which are present in mapping dataset.
- ✓ Currently can process only 100 resumes in 30 seconds (hardware limitation). Can be solved using GPU.

### Most challenging part of the solution?

- ✓ Data extraction from PDF or DOCX.
- ✓ Segregation of the data.

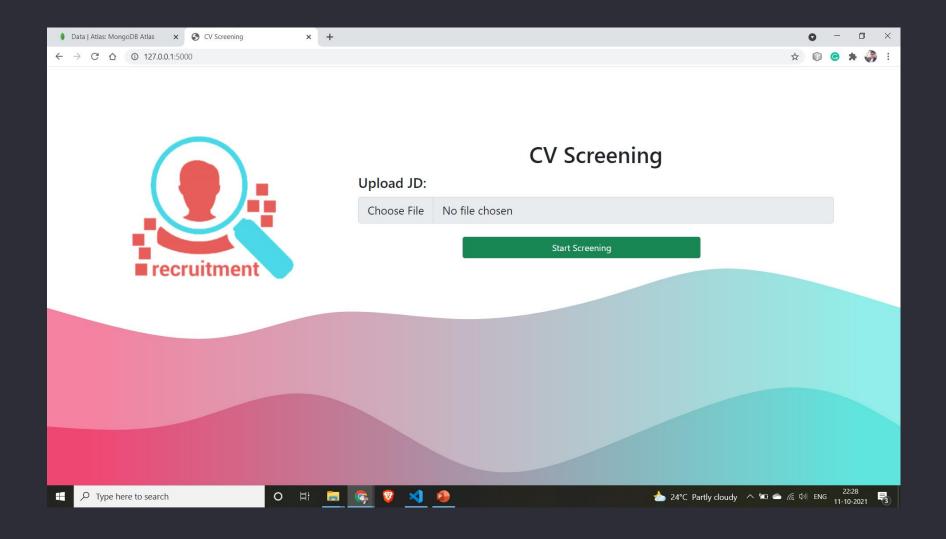


# **Block diagram**



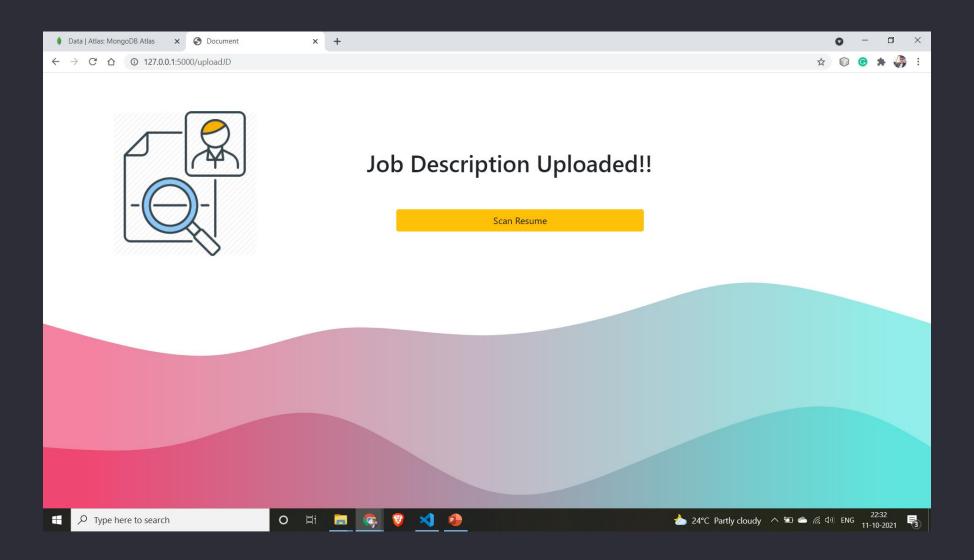


# Wire frames





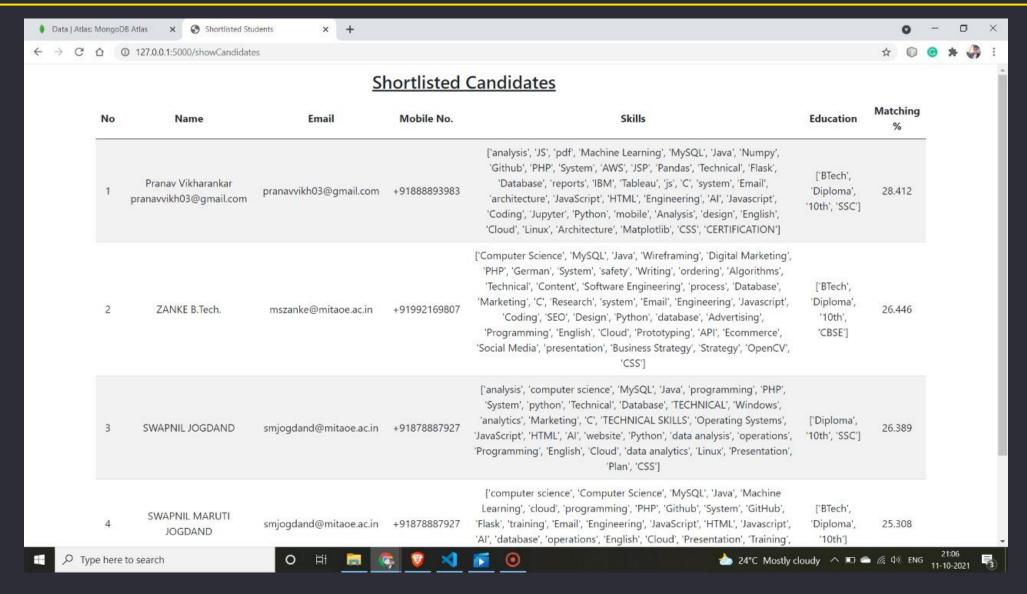
# Wire frames



Prototype submission for participants – Hackpions | EY GDS Hackathon

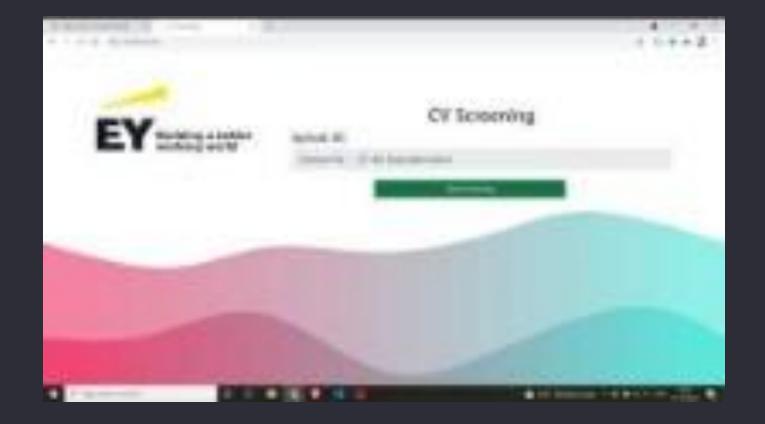


## Wire frames





# **Working Prototype**



Prototype submission for participants – Hackpions | EY GDS Hackathon

Video Link :- Link



# **Technology Stack**

- Python
- Flask framework
- MongoDB Atlas
- HTML, CSS, Bootstrap 5.0
- NLP Spacy library
- Scikit learn python library
- PDF Miner and docx2txt library



Prototype submission for participants – Hackpions | EY GDS Hackathon

# Attachment

Link for the paper draft :-



## **Future Scope**

- Extracting experience from resume.
- Increasing the skill mapping dataset.
- Filter out result on the basis of different combinations like only (skills and education, skills and experience, experience and education, etc.)
- ▶ To scan and retrieve result for thousands of resume in minimum time using big data techniques.

Prototype submission for participants – Hackpions | EY GDS Hackathon

Exporting of results into excel sheet



