**AI-Powered Resume Screening Tool**

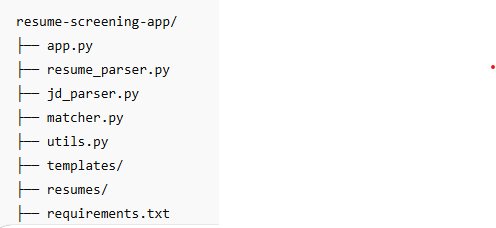
**Problem Statement:**

Recruiters spend excessive time manually screening resumes. This tool automates the process by parsing resumes, extracting key information, and matching candidate skills with a job description using NLP.

**Tech Stack :**

* **Language**: Python
* **Libraries**: spaCy / BERT, Scikit-learn, PyPDF2 / pdfminer, Pandas
* **Interface**: Streamlit (for UI) or Flask (for API deployment)
* **Key Features**
* **Resume Parsing**
  + Extract text from uploaded PDF resumes using PyPDF2 or pdfminer.six.
  + Clean and tokenize the text using NLP (spaCy/BERT).
* **Job Description Analysis**
  + User inputs or uploads a job description.
  + Extract key requirements like skills, technologies, experience, and qualifications.
* **Skill Matching & Candidate Scoring**
  + Compare each resume’s extracted content with the job description.
  + Use **cosine similarity**, **TF-IDF**, or **BERT embeddings** to score the match.
  + Generate a match percentage for each candidate.
* **Candidate Ranking**
  + Rank all candidates based on similarity scores.
  + Display rankings in a sortable and interactive table.
* **User Interface**
  + Streamlit dashboard to upload resumes and job descriptions.
  + Display parsed resumes, extracted skills, match score, and final ranking.
* **Optional Enhancements**
  + Use Named Entity Recognition (NER) to extract education, experience, and skills.
  + Integrate a feedback system to improve ranking logic over time.

**Folder Structure**



**How to Run**

1. Clone the repo
2. Install dependencies: pip install -r requirements.txt
3. Run Streamlit: streamlit run app.py
4. Upload job description and multiple resumes
5. View scores and download top-ranked resumes

**ML/NLP Concepts Used**

* Text preprocessing (tokenization, stop word removal)
* TF-IDF vectorization or BERT embeddings
* Cosine similarity for scoring
* Named Entity Recognition (spaCy)