Enhancing Job Screening

Introduction

- **Title:** Enhancing Job Screening with AI-Powered Multi-Agent System

- **Subtitle:** Revolutionizing Recruitment Through Automation

- **Team Name:** <u>The Infinite Loopers</u>

- Title: Goal of the Project:-

  - Build a Multi-Agent AI System to:

  - Read and summarize Job Descriptions (JDs).

  - Match candidate qualifications with JD requirements.

  - Automatically shortlist candidates.

  - Send interview invites to selected candidates.

Problem Overview / PS ID: R1-05

HACKRONYX

THE ISSUES

- Manual screening of resumes and job descriptions is **time-consuming.**

- **Human errors** and biases often affect **candidate shortlisting.**

- **Difficulty in matching** relevant skills and experience with the JD
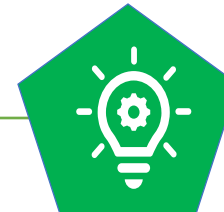
THE RESOLUTIONS

- **Faster** and more accurate candidate screening.

- Reduced bias and **improved hiring quality.**

- **End-to-end** automation of initial recruitment stages.

THE OUTCOMES

- Traditional job screening is **time-consuming**

- **High risk** of bias and manual error

- Recruiters overwhelmed by volume of applicants
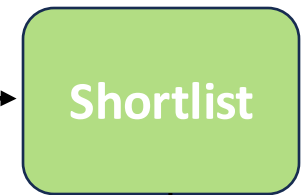
Solution Overview

OUR AI HIRING ASSISTANT

" We offer an AI-powered platform that automates job-CV matching and streamlines candidate screening."

JD and Resume Analyzer

AI-Based Candidate matching

Match Scoring & Ranking

Automated Interview scheduling

```
JD → NLP Agent → Matching → Shortlist → Interview Boc
```

- Built using multi-agent architecture

- Easily pluggable into existing HR systems

- Make requirement faster, smarter, and bias-free for modern companies
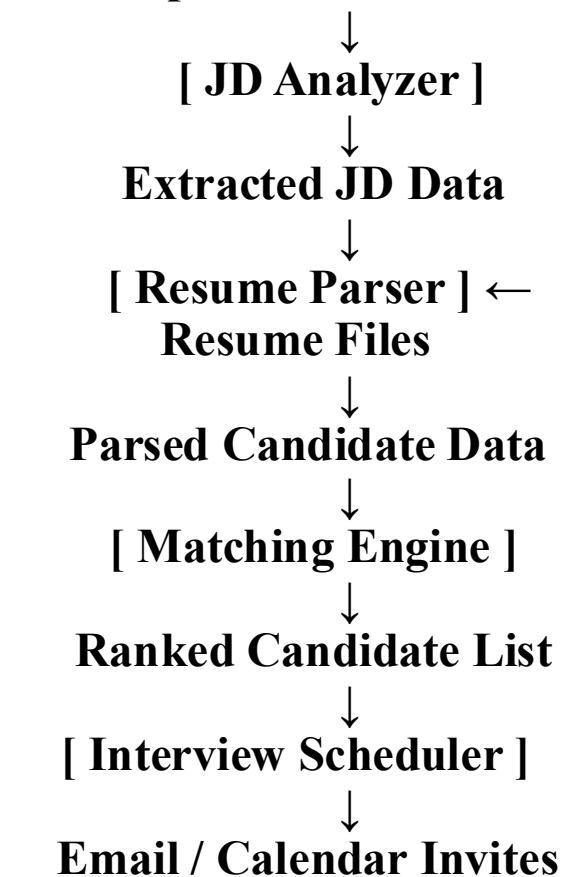
Technical Deep Dive

- **Architecture:**
- Agent 1: JD Summarizer (NLP model)
- Agent 2: Resume Extractor (NER + Parser)
- Agent 3: Matching Engine (ML-based)
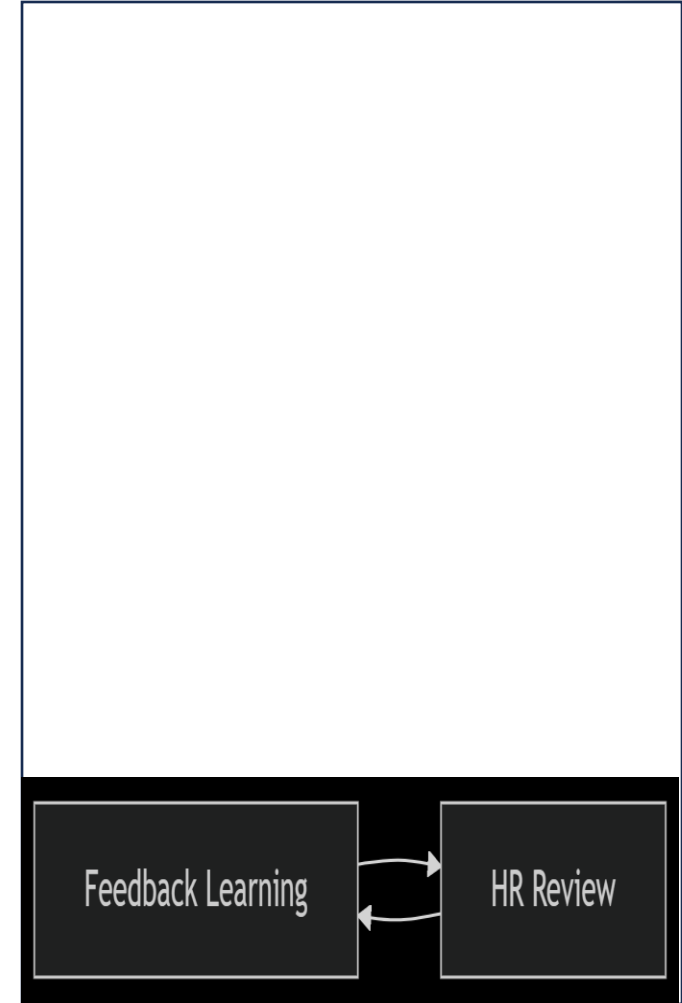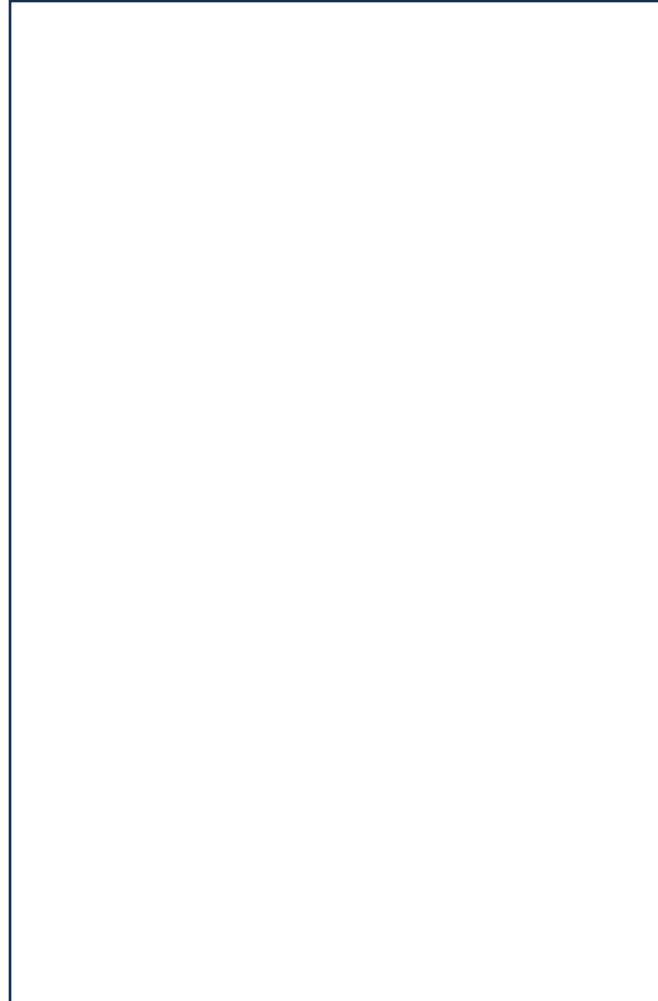- Agent 4: Interview Scheduler (Rule-based + Email APIs)

Tech Stack:
- Python, FastAPI/Flask
- NLP: spaCy, NLTK, or LLM (e.g., GPT or BERT)
- DB: MongoDB / PostgreSQL
- Frontend: React/HTML
- Scheduler: Google Calendar API, SMTP

User Uploads JD/Resumes
↓
[JD Analyzer]
↓
Extracted JD Data
↓
[Resume Parser] ←
Resume Files
↓
Parsed Candidate Data
↓
[Matching Engine]
↓
Ranked Candidate List
↓
[Interview Scheduler]
↓
Email / Calendar Invites

Implementation

Feedback Learning ⇄ HR Review

✓ Implementation Details

•**FRONTEND:**
HTML / CSS / TAILWIND / BOOTSTRAP / JAVASCRIPT / REACTJS

•**BACKEND:**
 NODEJS / PYTHON / DJANGO / EXPRESSJS

•**DATABASE:**
 MYSQL / MONGODB

•**AGENTS:**
 CREWAI / LANGCHAIN AGENTS / CUSTOM PYTHON CLASSES

•**EMAIL INTEGRATION:**
 SENDGRID API

•**NLP / ML:**
 SPACY / SBERT / SCIKIT-LEARN / HUGGINGFACE TRANSFORMERS

1.FEASIBILITY

➢ **Technical Feasibility**:
➢ NLP and LLMs can effectively parse JDs and CVs.
➢ Existing tools (Hugging Face, spaCy) support text extraction and matching.
➢ Multi-agent frameworks (AutoGen, LangChain) simplify orchestration.

➢ **Operational Feasibility**:
➢ Can integrate with existing ATS (Applicant Tracking Systems).
➢ Reduces HR workload, allowing focus on high-value tasks.

2. IMAPCT

➢ **80% Faster Screening Cycles**
➢ Automated processing significantly reduces the time .spent on reviewing JDs and CVs.

➢ **Reduced Bias**
➢ Standardized AI-driven scoring ensures fairer and more consistent candidate evaluation.

➢ **Enhanced Candidate Experience**
➢ Timely interview updates and communication improve engagement and satisfaction

➢ **Higher Hiring Accuracy**
 Intelligent matching improves the quality of shortlisted candidates.

Team Members

1. Chandni – Backend Developer | LinkedIn: [https://www.linkedin.com/in/chandni-suri-2639052ba/]

2. Deepak Kumar Sah – Backend Developer | LinkedIn: [https://www.linkedin.com/in/deepak-gupta-1017122a8]

3. Vikash Kumar – Frontend Developer | LinkedIn: [https://www.linkedin.com/in/vikash-kumar-038417366]

4. Rohit Kumar – Frontend Developer | LinkedIn: [https://www.linkedin.com/in/rohit-kumar-89770b2a8]

5. Sujay Kumar – Frontend Developer | LinkedIn: [https://www.linkedin.com/in/sujay-kumar-bb301a2a5]