

Final Capstone Project – Data Science & AI Internship

Project Title: AI Powered – Multi-Agent Resume Screener and Recommender

Objective

Build a **production-quality, enterprise-grade web application** that intelligently screens, analyzes, and recommends the **top 3 resumes per job role** using an AI-powered multi-agent system. Interns will work independently to create a clean, modular, and client-ready product that simulates real-world hiring automation using LLMs, LangGraph, and a web interface.

Key Features & Deliverables

Core Functionality:

- Resume Upload (PDF/DOCX) *–at least 50 resumes*
- Job Description Selector (Designation, Role, Experience, Skill-set and Project similarity)
- Multi-agent orchestration (LangGraph)
- Top 3 candidate recommendations with detailed explanations (make it available for download)
- Responsive and attractive enterprise-style web UI
- Fast response time with optimized agent execution

Agents to Implement:

| Agent | Description |
|------------------------|--|
| Recruiter Agent | Extracts education, skills, tools, and experience from resumes |
| Analyst Agent | Matches extracted features to JD using a scoring algorithm |
| HR Agent | Evaluates tone, soft skills, and flags any red flags |
| Recommender Agent | Ranks resumes and suggests top 3 for the selected role |
| Feedback Agent (Bonus) | Learns from rejected candidates and improves logic (optional) |

Timeline – 10-Day Plan

Day Task

- 1 Understand LangChain, LangGraph, agent orchestration
 - 2 Build the resume parsing utility
 - 3 Create Recruiter Agent
 - 4 Create Analyst Agent with scoring model
 - 5 Build HR Agent to assess tone & communication
 - 6 Implement Recommender Agent with ranking
 - 7 Orchestrate agents using LangGraph
 - 8 Build a user-friendly, responsive web interface
 - 9 Optimize response time and polish the product
 - 10 Final testing, submit live working link, and documentation
-







Public Access Requirement

The final deployed application should be **accessible online** through a temporary public URL (via any free hosting/tunneling tool). Ensure:

- The app is live and stable when submitted
 - Load time and agent response time are minimized
 - The web UI is clean, enterprise-ready, and intuitive
-

Submission Checklist

Each candidate must submit:

-  A working **website link** to their deployed capstone app
 -  Top 3 resume suggestions clearly shown with explanations
 -  A well-structured and modular codebase (including project structure)
 -  README.md with clear setup instructions and tech overview
 -  UI that is responsive, clean, and client-presentable
 -  Bonus: Include a feedback agent or resume scoring charts (optional)
-

Evaluation Criteria

Category

- Code Structure & Modularity
- Multi-Agent Logic & LangGraph Workflow
- Response Time & Efficiency
- User Interface (Professional & Responsive)
- Technical Depth (Use of AI, vector search, etc.)
- Documentation & Delivery (README, live link, presentation)