MSIS 549 Project Proposal: JoraFlow

Name: Nazim Zerrouki
Email: nazimz@uw.edu

1. Elevator Pitch

A lot of job seekers struggle with manual application tracking and data fragmentation, which costs them hundreds of hours in lost productivity and significant 'job search fatigue.' So I want to build JoraFlow, an automated recruitment analytics dashboard that helps them visualize their entire pipeline and optimize their strategy by using AI agents to autonomously parse emails and job boards for status updates. Unlike manual spreadsheets or basic Chrome extensions, my solution removes the manual entry burden entirely by proactively syncing with user communication channels.

2. Project Description

Problem Statement: The modern job search is a numbers game. Candidates often apply to hundreds of roles across multiple platforms (LinkedIn, Indeed, Greenhouse). Currently, there is no automated way to aggregate these data points. Candidates are forced to manually update spreadsheets, leading to "ghosting" going unnoticed and a lack of clear data on which versions of their resume or which job boards are actually yielding interviews.

How AI Technologies Will Address This: I will address this problem through Agentic Orchestration. Specialized AI agents will use **Gemini Pro** to read and categorize unstructured data from email notifications (confirmations, interview invites, rejections).

- **LLM Parsing:** Instead of rigid regex, LLMs will "read" emails like a human to extract company names, job titles, and specific status updates (e.g., "moving to next round" vs. "keep your resume on file").
- **Automated UI Generation:** Using tools like **Lovable**, the system will instantly translate this parsed data into a visual "Recruitment Funnel" (Sankey Diagram).

Target Users: Early-career professionals, career changers, and tech job seekers who apply to high volumes of roles and participate in online communities like r/JobSearch.

Data Sources:

- **Primary:** User Gmail/Outlook APIs (via OAuth).
- **Secondary:** Browser-based metadata from job board confirmation pages.

Anticipated Challenges:

- **Security/Privacy:** Navigating the "restricted scope" requirements for Google/Microsoft APIs.

- **Email Variance:** Designing prompts that handle thousands of different email templates from various Applicant Tracking Systems (ATS).
- **Token Efficiency:** Managing the cost of LLM calls when a user has 500+ job-related emails.

3. Implementation Plan

Type of Solution: A Web-based Dashboard with an underlying Multi-Agent Backend
.
Technologies: Gemini 3 Pro (for its massive context window, which is perfect for parsing long email threads).
- **Development Agents:** Lovable (UI/Frontend), Codex/GitHub Copilot (Backend logic).
- **Automation:** Python-based agents connected via GitHub for CI/CD.

Technical Approach:
- **Ingestion:** Agents pull the last 90 days of "Job" or "Career" filtered emails.
- **Analysis:** A "Parser Agent" identifies key status triggers in each thread.
- **Visualization:** Data is pushed to a Supabase/PostgreSQL database and rendered in a real-time dashboard.

Demo Concept: I plan to showcase a live "Recruiting Funnel" generated in real-time. I will input a sample dataset of 100 varied job emails and show how the AI agents instantly categorize them into a visual Sankey diagram showing "Applied → Interviewed → Rejected → Offered."

4. Selected Project Strengths

- **Business Value & Real-World Applicability:** This tool addresses the massive mental health and productivity drain the recruiting process has on a candidate's psyche. By providing transparency, it allows users to pivot their strategy (e.g., "I'm getting rejections at the screen stage; I need to change my interview prep, not my resume").
- **Technical Innovation (Agentic Workflow):** Unlike simple scrapers, this uses autonomous agents to handle the high variance of human language in emails, making the "manual spreadsheet" obsolete.