AI-Powered Resume-Job Description Matcher - Interview Project Statement

Notes:

- Bring your laptop with a working prototype of the application you've developed so far.

- You are welcome to explore your architectural approach — feel free to go beyond the suggested styles if it better solves the problem. We're more interested in *how you think* than whether you follow a specific template.

- While we’re not expecting a full-fledged application, your prototype should demonstrate **SOME** of the core features that address the main goals of the project.

- Focus on what makes your work unique — whether it’s your approach, optimization, design, or any creative feature. This is your opportunity to highlight your strengths and problem-solving skills.

Problem Statement

The Challenge: Modern Job Application Struggles

In today's competitive job market, job seekers face several critical challenges:

The Numbers Game: A typical software engineer position receives numerous applications, while job seekers apply to a few positions monthly, creating a massive mismatch between effort and success.

Skill Gap Blindness: Candidates often apply to positions without understanding exactly why they're rejected. Common scenarios include:

- A frontend developer applying for a full-stack role, missing backend skills
- A data analyst applying to ML engineer positions lacking specific algorithms knowledge
- Recent graduates applying for senior roles without understanding the experience gaps
- Career changers do not know which skills to prioritize for their target roles

ATS (Applicant Tracking System) Black Box: Most resumes are filtered out by automated systems before human review, but candidates have no visibility into why their applications fail.

Inefficient Preparation: Job seekers waste time preparing generic applications instead of focusing on the most relevant skill gaps for their target positions.

Real-World Impact

For Job Seekers: Hours spent crafting applications for unsuitable roles, missed opportunities due to poor resume optimization, and frustration from a lack of constructive feedback.

For Recruiters: Time wasted reviewing mismatched applications, difficulty identifying genuinely qualified candidates, and increased hiring timelines.

The Solution Vision

Build an **Intelligent Resume-Job Description Matching System** that provides instant, actionable feedback to job seekers by analyzing resume-job fit, identifying skill gaps, and offering specific improvement recommendations.

Project Overview

Develop an AI-powered matching system that analyzes resumes against job descriptions using natural language processing and machine learning to provide percentage match scores, identify missing skills, and offer actionable career development insights.

Technical Requirements

Core Features

1. Document Processing & Analysis

- **Resume Upload**: Support PDF, DOC, DOCX formats with text extraction
- **Job Description Input**: Text area for pasting JD content with formatting preservation
- **Content Parsing**: Extract key information from both documents (skills, experience, education, requirements)
- **Entity Recognition**: Identify technologies, frameworks, certifications, and soft skills

2. Intelligent Matching Engine

- **Semantic Analysis**: Use sentence embeddings to understand context beyond keyword matching
- **Skill Alignment**: Compare technical and soft skills between the resume and the JD
- **Experience Matching**: Analyze years of experience, role levels, and career progression
- **Education Alignment**: Match degree requirements, certifications, and specialized training

3. Comprehensive Reporting

- **Match Percentage**: Overall compatibility score with breakdown by categories
- **Skill Gap Analysis**: Detailed list of missing technical and soft skills
- **Strength Identification**: Highlight areas where the candidate exceeds requirements
- **Improvement Recommendations**: Specific, actionable suggestions for resume enhancement

4. Interactive Skill Comparison

- **Side-by-side Table**: Visual comparison of required vs. present skills
- **Skill Categories**: Organized by Technical, Soft Skills, Experience, Education
- **Priority Ranking**: Importance-weighted missing skills based on JD emphasis
- **Progress Tracking**: Save analyses for monitoring improvement over time

Technology Stack

Machine Learning & NLP Component

- **Sentence Embeddings**: Sentence-BERT or Universal Sentence Encoder for semantic similarity
- **Entity Extraction**: spaCy NER models + custom skill recognition
- **Text Processing**: NLTK/spaCy for preprocessing and tokenization
- **Similarity Computation**: Cosine similarity for semantic matching
- **Skill Database**: Curated technology and skill taxonomy for accurate matching

Frontend Development

- Drag-and-drop file upload interface
- Rich text editor for job description input
- Interactive skill comparison tables
- Progress bars and match visualization
- Responsive design for mobile and desktop
- A mobile application with having camera feature
- A rich-featured dashboard

Backend Infrastructure

- **Primary**: FastAPI for a production-ready API with automatic documentation
- **Document Processing**: PyPDF2/pdfplumber for PDF extraction
- **File Storage**: Local storage with optional cloud integration
- **Caching**: Redis for storing analysis results and improving response times

Bonus Features

- **Skill Gap Prioritization**: "You are missing X, Y, Z skills" with learning resources
- **Industry-Specific Analysis**: Tailored matching for different tech domains
- **Resume Optimization Suggestions**: Specific wording and formatting improvements
- **Salary Expectation Analysis**: Match compensation expectations with market rates
- **Multiple Job Comparison**: Compare one resume against multiple job descriptions

System Architecture

User Journey

1. **Document Upload**:
   - User uploads resume (PDF/DOC) via drag-and-drop interface
   - The system extracts and processes text content
   - User pastes job description in the text area
2. **Processing & Analysis**:
   - Extract entities and skills from both documents
   - Generate sentence embeddings for semantic analysis
   - Calculate similarity scores across multiple dimensions
   - Identify gaps and strengths
3. **Results & Insights**:
   - Display overall match percentage with category breakdown
   - Show a detailed skill comparison table
   - Highlight missing skills with priority ranking
   - Provide specific improvement recommendations
4. **Action Planning**:
   - Export results as a PDF report
   - Save the analysis for future reference
   - Get learning resource recommendations for skill gaps

API Design

```
POST /api/analyze/match
- Input: {"resume_file": file, "job_description": "text"}
- Output: {
   "overall_match": 78.5,
   "category_scores": {
     "technical_skills": 85.2,
     "soft_skills": 72.1,
     "experience": 80.0,
     "education": 90.0
   },
   "missing_skills": ["Docker", "Kubernetes", "GraphQL"],
   "strengths": ["React", "Python", "Machine Learning"],
   "recommendations": [...]
  }

GET /api/skills/extract
- Input: {"text": "resume or job description text"}
- Output: {"skills": [...], "entities": {...}}

POST /api/reports/generate
- Input: {"analysis_id": "uuid"}
- Output: {"pdf_url": "generated_report.pdf"}
```

ML Pipeline Architecture

Input Documents → Text Extraction → Entity Recognition →
Skill Extraction → Embedding Generation → Similarity Calculation →
Gap Analysis → Recommendation Engine → Results

Sample Use Cases & Test Scenarios

Test Case 1: Frontend Developer → Full-Stack Role

- **Resume**: React, JavaScript, CSS, HTML, Git
- **Job Description**: React, Node.js, MongoDB, Docker, AWS
- **Expected Output**: 65% match, missing backend skills (Node.js, MongoDB, Docker)

Test Case 2: Data Analyst → ML Engineer Role

- **Resume**: Python, SQL, Pandas, Tableau, Statistics
- **Job Description**: Python, TensorFlow, PyTorch, MLOps, Deep Learning
- **Expected Output**: 40% match, missing ML frameworks and deployment skills

Test Case 3: Recent Graduate → Senior Developer Role

- **Resume**: Java, Spring Boot, MySQL (1 year experience)
- **Job Description**: Java, Spring, Microservices, Leadership (5+ years)
- **Expected Output**: 55% match, missing senior-level experience and architecture skills

Deliverables

1. **Working Application**: Deployed system with core matching functionality
2. **ML Pipeline**: Pre-Trained models and skill extraction system
3. **Sample Dataset**: Curated test cases demonstrating various scenarios
4. **Technical Report**: Architecture decisions, algorithm choices and system design approach that were chosen to develop the solution
5. **User Guide**: Instructions for optimal resume and JD formatting

Real-World Applications

This system addresses genuine pain points in the job market, helping:

- **Job Seekers**: Focus preparation efforts on the most relevant skills
- **Career Counselors**: Provide data-driven guidance to clients
- **Recruiting Agencies**: Pre-screen candidates more effectively
- **Educational Institutions**: Align curriculum with industry demands

The project demonstrates practical AI application in HR technology, a rapidly growing field with significant commercial potential.