recommendation system architecture

https://lucid.app/documents/embedded/9b6d3187-f2fc-490b-a958-1230637f5871

AI Recommendation System Architecture

System Overview

This architecture diagram represents a 4-tier recommendation system for your LinkedIn-style job platform. The system is designed to efficiently match job seekers with relevant job listings based on skills, experience, and preferences.

Key Components

1. **Client Application Layer**
    - Job Seeker Interface: Where users search for jobs and receive recommendations
    - Recruiter Interface: Where employers post jobs and find candidate matches
2. **API Layer (FastAPI)**
    - `/recommendations/jobs`: Endpoint for job recommendations to users
    - `/recommendations/candidates`: Endpoint for candidate recommendations to employers
    - Handles authentication, request validation, and response formatting
3. **Recommendation Engine Layer**
    - Feature Extraction: Processes raw user and job data into comparable feature vectors
    - Scoring Algorithm: Calculates match scores based on multiple factors

- Explanation Generator: Creates human-readable explanations for recommendations

4. **Data Storage Layer**
    - User Profiles: Database storing user skills, experience, and preferences
    - Job Listings: Database of current job postings and requirements
    - Skill Taxonomy: Reference database for standardizing and relating skills
    - Redis Cache: In-memory storage for quick access to computed recommendations

Data Flow

1. **Recommendation Request Flow:**
    - User requests recommendations through the client interface
    - Request is sent to the API layer
    - API calls the recommendation engine
    - Engine first checks Redis cache for existing results
    - If not cached, engine:
        - Retrieves relevant data from databases
        - Extracts features from profile and job data
        - Calculates scores and generates explanations
        - Stores results in Redis cache
        - Returns formatted recommendations to client
2. **Data Update Flow:**
    - When user profiles or job listings are updated
    - Affected cache entries are invalidated
    - Recommendations are recalculated on next request