Flow Overview

Step-by-step flow from start to end:

Step 1: User Registration and Login

- **Frontend**: Users register and log in through the user interface. The system gathers basic information such as name, email, educational background, and career goals.
- **Backend**: Once the user submits the registration, **Spring Boot** handles the request, storing user information in the **MySQL** database through **Hibernate**.

Step 2: Skill Gap Analysis and Learning Plan

- **User Action**: After logging in, users provide details about their current skill set, desired career path, and the job they are interested in.
- **Backend Process**: The **Skill Gap Analysis** feature processes this data:
  1. The system compares the user's current skills with the requirements of the target job.
  2. It assigns weightages to both the existing and required skills, highlighting gaps.
  3. A personalized learning plan is generated, suggesting courses or certifications to bridge the gap.
- **Frontend**: The user sees a detailed breakdown of their skill gaps and a list of recommended courses that they can pursue.
- **Technologies**:
  - **Java 17**, **Spring Boot 3.0** handle the skill gap calculations and communicate with the front-end.
  - **MySQL 8.0** stores the user’s profile and skill information.
  - The learning plan is updated dynamically based on the user’s progress and market trends.

Step 3: Course Recommendation System

- **Backend Process**:
  1. The system regularly fetches and updates available courses from various learning platforms.
  2. Based on the user’s progress and the industry demand for specific skills, new courses are recommended.
  3. The user receives a dynamic list of courses tailored to fill their skill gaps.
- **Frontend**: A list of courses is displayed, where users can select and enroll in those that best match their career goals.
- **Technologies**:
  - **Spring Boot 3.0** fetches course data.
  - **JS** dynamically updates the course list based on user interaction.

Step 4: Job Matching and Application Assistance

- **User Action**: The user can access the **Job Matching** feature to find jobs relevant to their skills.
- **Backend Process**:
    1. The system uses **Open NLP** (Natural Language Processing) to scan job listings and match them with the user's profile.
    2. The job listings are presented to the user.
- **Frontend**: Users can view job matches, apply directly
- **Technologies:**
    - **Open NLP** processes job descriptions and tailors them to the user profile.
    - **MySQL** stores user resumes and their job search preferences.

Step 5: Feedback System:

- **User Action**: Once users complete a course, they can provide feedback.
- **Backend Process**:
    1. The feedback is collected and stored in the database.
    2. Feedback is analyzed to understand user satisfaction, learning effectiveness, and the relevancy of the course to the job market.
    3. This analysis is used to refine future course recommendations and improve the system's learning resources.
- **Frontend**: The feedback form is displayed to users at the end of every course/module, allowing them to rate their experience and suggest improvements.
- **Technologies**:
    - **Spring Boot 3.0** for collecting and processing feedback.
    - **MySQL 8.0** for storing feedback and course interaction data.

Step 6: Conclusion and User Benefits

- Once the users have used all features (skill gap analysis, job matching, mentorship, etc.), they can seamlessly navigate their career path, improve skills, and connect with job opportunities.
- The system continuously provides personalized recommendations, and through regular feedback and course completion, it ensures that users remain up-to-date with the job market's evolving demand