**Tech-Tribe**

**Group Name: Tech Tribe**

**Phase III**

**Project Title: Employee Skills and Career Development Platform**

**System Architecture Overview:**

* **Frontend (Client Side):**
  + The frontend is the part of the system that users interact with directly through their web browsers.
  + It includes components for the user interface, such as profile management forms, job search interfaces, skill assessment tools, and feedback submission forms.
  + Frontend technologies like HTML, CSS, JavaScript, and Angular framework will be used to create dynamic and interactive web pages.
* **Backend (Server Side):**
  + The backend handles server-side logic, database interactions, and business processes.
  + It consists of web servers, application servers, and databases. Backend technologies like Node.js with Express framework will be used for server-side scripting and API development.
  + The backend communicates with the frontend via Restful APIs, handling requests for data retrieval, updates, authentication, and authorization.
* Database:
  + MongoDB, a NoSQL database, will be used to store and manage structured data related to user profiles, skills, job listings, feedback, and analytics.
  + MongoDB's document-based storage allows for flexibility and scalability, suitable for a web-based application with evolving data needs.
* **Security Layer:**
  + The security layer ensures the confidentiality, integrity, and availability of data.
  + It includes components for user authentication, authorization, data encryption (TLS/SSL for data in transit, AES for data at rest), input validation, and security monitoring.
  + Security measures are implemented both in the frontend (client-side validations) and backend (server-side validations and access controls).

**Component diagram for Website:**

User’s Browser

Explanation of Components:

Database

MongoDB

Backend:

Components:

* Web Server(Node.js +Express)
* Application Logic
* Restful APIs
* Database Interaction (MongoDB)
* Security Layer

Frontend:

Components:

* Profile Management Interface
* Job Search Interface
* Skill Assessment Tools
* Feedback Submission Forms
* User's Browser:
  + This represents the user interface where users interact with the platform using their web browsers.
* Frontend Components:
  + Profile Management Interface: Allows users to create, edit, and manage their profiles.
  + Job Search Interface: Enables users to search for job vacancies within the organization.
  + Skill Assessment Tools: Provides tools for users to assess their skills and identify skill gaps.
  + Feedback Submission Forms: Allows users to submit feedback and set career development goals.
* Backend Components:
  + Web Server (Node.js + Express): Handles HTTP requests from the frontend and routes them to appropriate backend services.
  + Application Logic: Implements business logic such as profile management, job recommendation, skill assessment algorithms, and feedback processing.
  + RESTful APIs: Exposes endpoints for frontend-backend communication, including user authentication, profile management, job listings, and skill assessment.
  + Database Interaction (MongoDB): Manages data storage and retrieval, storing user profiles, job data, skill assessments, feedback, and analytics.
  + Security Layer: Implements security measures such as user authentication, data encryption, input validation, and access controls to ensure data security and privacy.

**Detailed diagram**

Employee

* Name: string
* Email: string
* Skills: list of string
* Certifications: list of string
* CareerGoals: list of string
* UpdateProfile()
* addSkill (skill:Skill)
* removeSkill(skill:Skill)
* setCareerGoals(goals: list of string)
* Name: string
* Description: string
* Skills: list of string
* ProficiencyLevel: list of string
* UpdateDescription(desc:string)
* UpdateProficiency(level:enum)

Skill

* Employee Class:
* Attributes:
  + Name: Represents the name of the employee.
  + Email: Represents the email address of the employee.
  + Skills: Represents a list of skills associated with the employee.
  + Certifications: Represents a list of certifications achieved by the employee.
  + CareerGoals: Represents a list of career goals set by the employee.
* Methods:
  + updateProfile(): Allows the employee to update their profile information.
  + addSkill(skill: Skill): Allows the employee to add a new skill to their profile.
  + removeSkill(skill: Skill): Allows the employee to remove a skill from their profile.
  + setCareerGoals(goals: list of string): Allows the employee to set or update their career goals.
* Skill Class:
* Attributes:
  + Name: Represents the name of the skill.
  + Description: Represents a brief description of the skill.
  + ProficiencyLevel: Represents the proficiency level of the skill (e.g., beginner, intermediate, advanced).
* Methods:
  + updateDescription(desc: string): Allows updating the description of the skill.
  + updateProficiency(level: enum): Allows updating the proficiency level of the skill.

This class diagram illustrates the basic structure of the Employee Skills and Career Development Platform, including the Employee and Skill classes along with their attributes and methods. It shows how employees can manage their profiles, skills, certifications, and career goals within the platform.

**Sequence Diagram for Performing Skill Assessment:**

Employee

1. Cliks

“Update Profile

1. Sends

updateProfile()

3. Updates Employee’s

Profile

Steps:

1. The employee clicks on the "Update Profile" option in the frontend interface.
2. The frontend sends a request to the backend's updateProfile() method.
3. The backend processes the request and updates the employee's profile in the database.

**Sequence Diagram for Updating Employee Profile:**

External

Backend

Frontend

Employee

1. Selects

“Skill Assessment”

1. Sends

“assessSkill()

1. Fetches

Skill Assessment

Tool’s

Questions

6. Calculates Skill Assessment Results

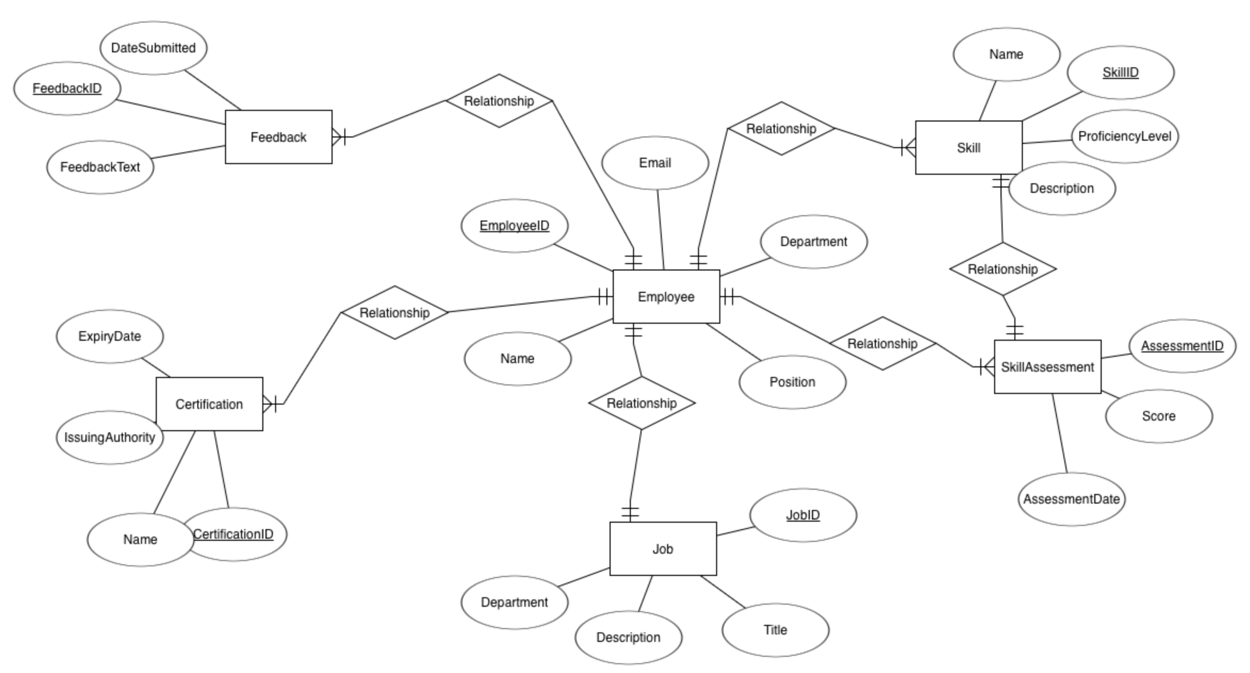
5. Answers Questions

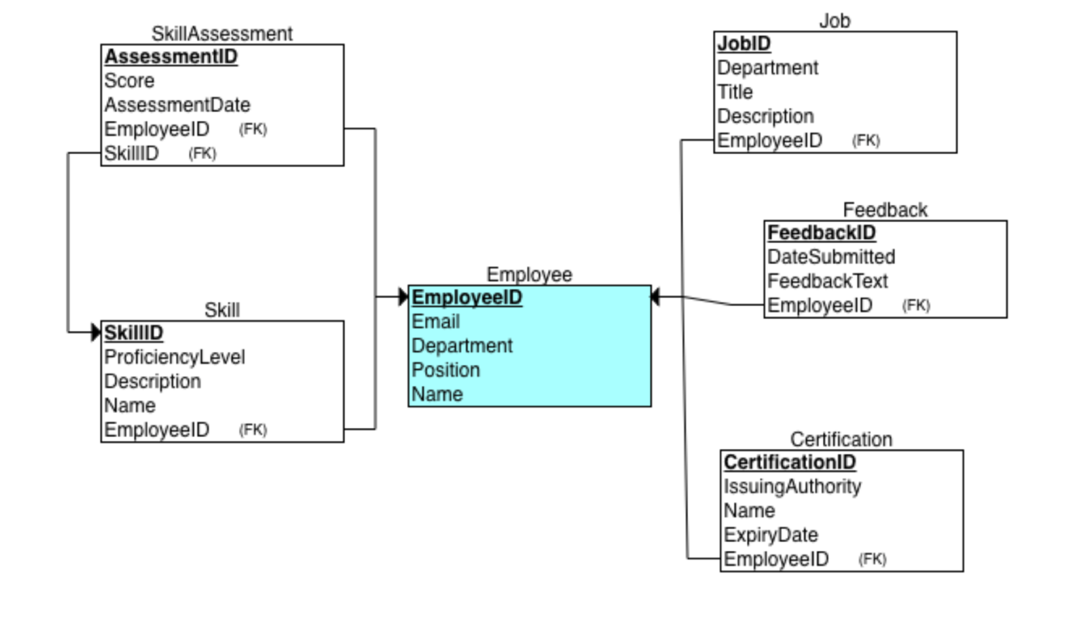
1. Presents Questions to Employee

Steps:

1. The employee selects the "Skill Assessment" option in the frontend interface.
2. The frontend sends a request to the backend's assessSkill() method.
3. The backend fetches the skill assessment questions from an external tool.
4. The backend presents the questions to the employee in the frontend.
5. The employee answers the questions.
6. The backend calculates the skill assessment results based on the employee's answers.

**Database Design:**

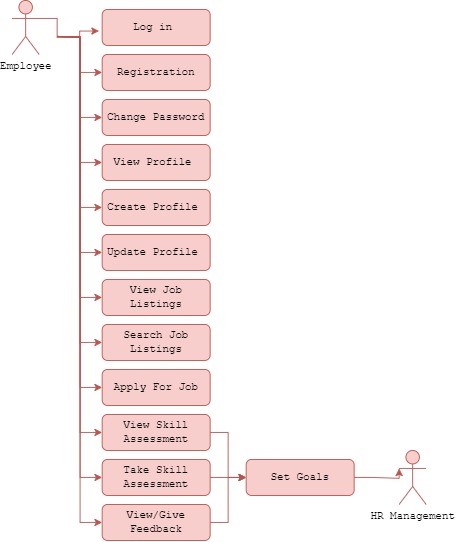




Relationships Between Tables:

* Employee - Skill Relationship:
  + Many-to-Many relationship: An employee can have multiple skills, and a skill can be associated with multiple employees.
  + This relationship is implemented using an intermediate table (e.g., EmployeeSkills) that maps employee IDs to skill IDs.
* Employee – Certification Relationship:
  + One-to-Many relationship: An employee can have multiple certifications, but each certification is associated with only one employee.
  + The Employee table includes a foreign key (EmployeeID) in the Certification table to establish this relationship.
* Employee - Feedback Relationship:
  + One-to-Many relationship: An employee can provide multiple feedback entries, but each feedback entry is associated with only one employee.
  + The Employee table includes a foreign key (EmployeeID) in the Feedback table to establish this relationship.
* Employee - SkillAssessment Relationship:
  + One-to-Many relationship: An employee can have multiple skill assessment records, but each assessment record is associated with only one employee.
  + The Employee table includes a foreign key (EmployeeID) in the SkillAssessment table to establish this relationship.

Modeling   
Use Case Diagram:



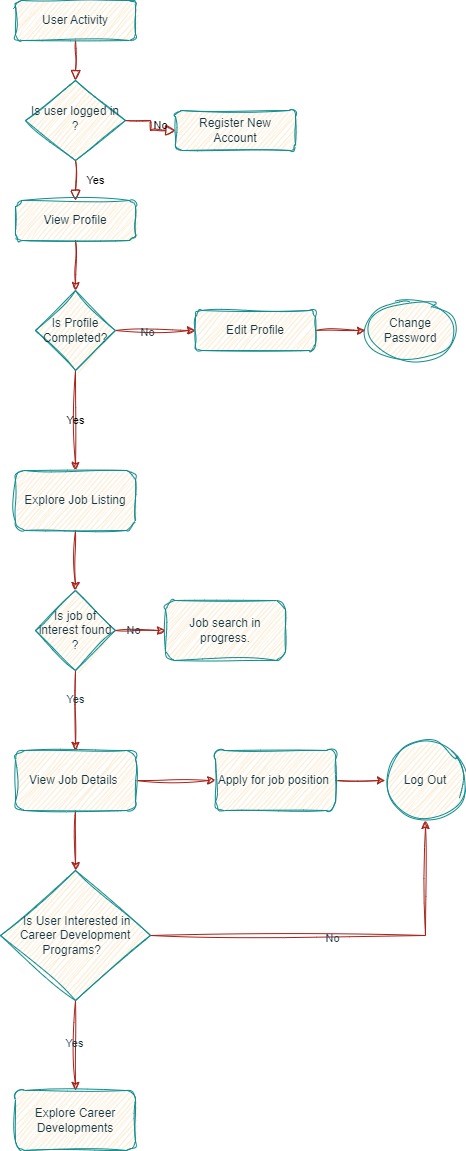
The Employee Skills and Career Development Platform Use Case Diagram shows how users engage with the system's operations, which include HR administration capabilities. The website is primarily designed for two types of users: "employees" and human resources professionals.

Employees may log in, register, and manage their profiles. This includes reading, creating, and updating profiles, as well as accessing job postings, completing skill assessments, and receiving comments. Employees may also define professional development objectives using the site.

Furthermore, HR administration features are included, allowing HR staff to access analytics and data, manage job postings (including addition, removal, and updating), and monitor employee comments.

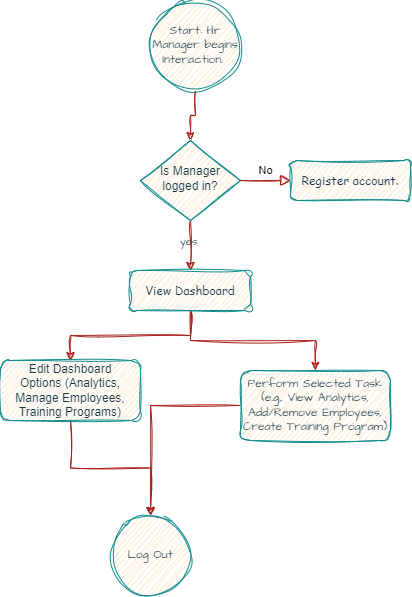
This thorough graphic provides a clear understanding of how users interact with the platform. It includes both employee-centric

Activity Diagrams:   
Employee:



The activity diagram illustrates the sequential steps involved when an employee engages with the Employee Skills and Career Development Platform. It begins with the user's initiation of interaction with the platform. The system first checks if the user is logged in; if not, the user is prompted to log in before proceeding. Upon successful login, the user views their profile, and the system verifies if the profile is complete. If the profile is incomplete, the user is directed to complete it before exploring further. Once the profile is complete, the user proceeds to explore job listings within the organization. If a job of interest is found, the user can view the job details. Additionally, the system prompts the user to consider exploring career development programs. If the user expresses interest, they can explore available opportunities. The activity concludes once the user completes their interaction with the platform, ensuring a seamless and guided experience for employees seeking skill development and career advancement. 

Hr Manager: 



The Employee Skills and Career Development Platform offers HR managers a comprehensive toolset to efficiently manage workforce capabilities and career advancement initiatives. At the outset of their interaction, HR managers are prompted to log in securely to access the platform's array of administrative functionalities. Once authenticated, they are greeted with a dynamic dashboard presenting a spectrum of options tailored to HR management tasks, ranging from workforce analytics to employee profile management and training program oversight.

HR managers navigate through the platform to address specific needs and objectives. Whether delving into detailed analytics reports to glean insights into workforce trends, fine-tuning employee profiles to reflect evolving skill sets, or orchestrating training programs to nurture talent and foster career growth, the platform provides intuitive interfaces and robust tools to streamline administrative workflows. Through seamless execution of tasks and prompt feedback mechanisms, HR managers can effectively leverage the platform to optimize talent management strategies and drive organizational success.

