**Intelligent Virtual Career Advisor - Technical Flow Document**

### **Project Overview**

**Title:** Intelligent Virtual Career Advisor  
 **Hackathon Duration:** 48 hours  
 **Team Name:** T3 Coders

**Team Leader:** Milan Bhadarka

**Objective:** To guide individuals through their career paths using an AI-driven system that combines skills assessment, job market analysis, personalized recommendations, and verifiable learning paths.

## **Technical Architecture Overview**

### **1. Skills Assessment Module**

**Input:**

* The user provides the skills they have.
* Resume upload (PDF/DOCX).

**Process:**

* **NLP Skill Extraction (Resume):**
  + Libraries: nltk, transformers, sentence transformer (sentence-transformers/all-MiniLM-L6-v2)
  + Preprocess and tokenize text, extract skills as the weighted entities from the resume.
  + Using Cosine Similarity for Weighted Skills
* **Quiz to check for Prerequisites to learn new skills:**
  + Quiz questionnaires generated by Mistral which are divided into difficulty levels (Beginner, Intermediate, Advanced, Expert) and assign the level of the User.
* **User’s Analytical Dashboard containing the skill and resume analysis.**
* **Clustering of the skills according to the different Tech Stacks from the resume:**  
  + Using K-Means Clustering to segment the clusters according to the skills.
* **Job Recommendation System**  
  + Using K-Means Clustering along with NLP(TF-IDF) and vectorizer to recommend appropriate jobs according to skills and the clusters and show top5 eligible jobs.

### **2. Job Market Analysis Engine**

**Data Sources:**

* Public APIs (LinkedIn Jobs, Kaggle Datasets,)
* **Exploratory Data Analysis**
  + Extract trending skills and hidden insights using Plotly
* **Personalized AI Assitant**
  + A RAG-based system fed through the job analytics data that provides insights on the job queries.
* **Prediction and Classification Models(Yet to be Integrated)**
  + A prediction model to predict the salary of the user based on the user metadata and the Job data.
  + Classification model to predict that the job will be affected by Automation or not(on the scale of High, Medium and Low) multiclass classification.
  + Classification model to predict whether the user will get the growth or not.

Model: XGBoost Regressor and Classifier (89% accuracy on regression)

**A RAG-based system fed through the job analytics data that provides insights on the job queries.**

**Output:**

* Answers to any technical Job query for career enhancement.
* Understanding of any tech stack

### **3. Career Path Recommender System**

**Approach:** Hybrid Recommender

* **Content-Based Filtering:**
  + Match the user’s skill vector with job requirement vectors.
  + Cosine similarity for closeness scoring.
* **Collaborative Filtering (if user data is available):**
  + Using nltk and TF-IDF and sentence-transformers/all-MiniLM-L6-v2 along with Cosine Similarity
* **Alternative:**
  + Frame as a classification problem: skill set as features, career role as target.
  + Model: XGBoost / Random Forest

**Output:**

* Top 5 recommended career roles.

### **4. Resume & Interview Tips Generator**

**Input:** Resume or extracted skills.

**Approach:**

* Using Gemini LLM to get the ATS score and the enhancements for resume.
* Provides:  
  + Resume improvement suggestions.
  + Interview prep tips based on targeted role.

**Output:**

* Tailored resume suggestions.
* Interview strategies per role.

### 

### **5. Skill-Based RAG Chatbot**

**Purpose:** Real-time skill/job-related querying using your CSV/FAISS + Groq LLaMA setup.

**Components:**

* FAISS Index (context retrieval from text chunks).
* Sentence Transformer (MiniLM) for embedding.
* Groq LLaMA (via RAG) for generation.

**Use Case:**

* Ask questions for any technical skills and Jobs.

### **6. Course Recommendation & Certification Verification(Image Processing) (REMAINING WORK)**

**Input:** Recommended Career path by the Career Path Recommender System

**Process:**

* Course dataset (Coursera, Udemy, edx, and skillshare).
* Match required skills → fetch relevant courses.

**Certificate Verification:**

* User uploads certificate (Image or PDF).
* Image Processing:  
  + OCR with Tesseract.
  + Logo/Seal detection for authenticity.
  + Template matching if applicable.

**Output:**

* Verified skill acquisition.

### **7. NFT-Based Badge & Blockchain Integration (REMANING WORK)**

**Input:** Verified course completion.

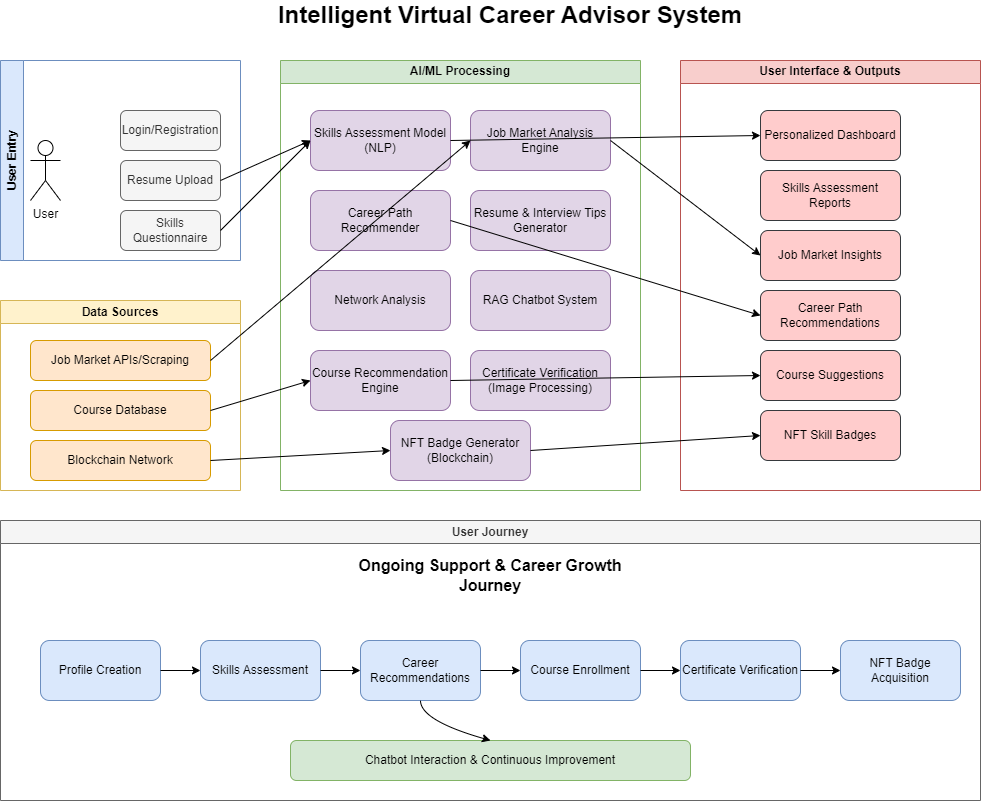
**Process:**

* Generate a digital badge NFT.
* Use smart contracts (e.g., Solidity) to mint NFT.
* Store badge on blockchain (Polygon / Ethereum testnet).

**Tools:**

* IPFS for file storage.
* Web3.py / Hardhat / Alchemy API.

**Output:**

* Immutable, verifiable badge owned by the user.
* 

Use Case Type Flow Diagram

### **Overall Tech Stack Summary**

* **NLP & ML:** spaCy, HuggingFace Transformers, sklearn, LDA, XGBoost, NLTK,
* **Data Processing:** Pandas, NumPy, Plotly, LabelEncoder, Standard Scalar
* **Model Serving & Backend:** FastAPI, MongoDB, Clerk, Node.js
* **Web Technology:** Next.js, Tailwind CSS, three.js, Spline, Cloudinary
* **Image Processing:** CV2, PaddleOCR, PIL
* **LLM Integration:** Groq API with LLaMA + RAG, Gemini, Mistral
* **Blockchain:** Solidity, Web3.py, Ethereum, sepoliam, remix ide

**FUTURE GOALS:**

* **Real-time Job Analytics with LinkedIn data**
* **To use OCR for Resume enhancements**
* **A recommender System which is not using Hugging Face Transformer.**