Career Recommendation System

Maharishi Markandeshwar (Deemed to be University)

In partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology

In

Computer Science Engineering

Name of Project Supervisor: - Dr. Vaishali Mehta

Name of Students: -

Anik Shah	(11232860-B3).
Suraj Sharma	(11232935-B3).

Maharishi Markandeshwar (Deemed to be University)

(NACC accredited Grade 'A++' University)

(Computer Science & Engineering Department, M. M. Engineering Collage, Mullana, Ambala – 133207, Haryana, India)

(2024-2025)

M.M. ENGINEERING COLLEGE
MAHARISHI MARKANDESHWAR (DEEMED TO BE UNIVERSITY)
(NAAC accredited Grade 'A++' University)
MULLANA, AMBALA, HARYANA

M.M. ENGINEERING COLLEGE
MAHARISHI MARKANDESHWAR (DEEMED TO BE UNIVERSITY)
(NAAC accredited Grade 'A++' University)
MULLANA, AMBALA, HARYANA
December, 2024
(Batch 2023-2027)

1. INTRODUCTION

In today's competitive environment, students often face difficulty in identifying suitable career paths based on their existing skills. Traditional career counseling lacks personalization and efficiency. To solve this, we present **SkillSense**, an AI-powered career recommendation system. This project uses **resume parsing**, **machine learning**, and **AI query handling** to suggest job roles that match the user's strengths. It extracts skills from a resume, predicts top career options using a trained model, and allows users to ask custom queries like *"Suggest jobs in AI"*. The system further recommends live job openings and upskilling courses to bridge knowledge gaps. It aims to help students and professionals make informed career decisions.

1.1 Objective

The objective of the SkillSense project is to create an intelligent and interactive system that assists users in identifying suitable career paths based on their existing skills and interests. The system analyzes uploaded resumes to extract technical and soft skills using natural language processing. It then uses a machine learning model, trained on real-world job-role datasets, to predict the most suitable job roles for the user. Additionally, SkillSense allows users to input natural language queries for personalized recommendations, making the experience more AI-driven. Based on the predicted roles or user queries, it fetches relevant live job openings and recommends online courses to help bridge skill gaps. Overall, the goal is to guide students and professionals toward informed and skill-aligned career decisions.

Organization of the report

The report is divided into 4 parts and each part deals with the different aspects of the system.

(i) Design: This part talks about the existing system, how they are designed and the issues associated with them. Furthermore, it describes the features of the system proposed and the requirements for operating it.

(ii) Module Description: This part describes each module implemented in the system,

i.e., how the data is processed in each and what are the steps involved from the user's point of view.

(iii) Implementation: This part deals with an overview of the platform for which the system is developed for. It also talks about the parameters needed for running the system and provides a sample of code used, along with screenshots of the output.

(iv) Conclusion: This part concludes the report and discusses the possible enhancement that can be implemented in the future improve the quality.

1.2 Challenges Overcome

During the development of the SkillSense system, several technical and logical challenges were encountered and successfully addressed. One major challenge was accurately extracting skills from diverse resume formats, which was overcome by integrating robust PDF parsing and natural language processing techniques. Another significant hurdle was training a machine learning model that could reliably predict job roles based on varied skillsets; this was resolved by using a large, publicly available dataset and refining feature selection methods. Integrating AI-style user queries into the system posed difficulties in natural language understanding, which were handled through flexible query interpretation logic. Additionally, implementing real-time job and course recommendations required dynamic web scraping and handling inconsistent API responses.

2. LITERATURE REVIEW

Career recommendation systems have gained attention in recent years due to the increasing complexity of the job

market and the growing need for personalized guidance. Traditional systems relied heavily on manual assessments or simple rule-based filters, which often lacked adaptability and accuracy. Recent studies highlight the effectiveness of machine learning and natural language processing in extracting meaningful information from resumes and mapping them to relevant job profiles. Research also suggests that combining structured data like skills with unstructured queries enhances recommendation quality. Systems such as LinkedIn's job suggestions and Google's career recommendations utilize similar concepts but are often limited to platform-specific data. Existing academic work has explored collaborative filtering, decision trees, and neural networks for role prediction. However, few systems integrate resume parsing, skill extraction, AI query handling, and real-time job/course recommendations in a unified platform. SkillSense aims to bridge this gap by building a comprehensive, intelligent, and user-driven career guidance solution.

Chart – 1: User's Preference to Mobile Application over Web Application

Based on a survey conducted among students and job seekers, it was observed that a large percentage of users prefer mobile applications over web applications due to ease of access, portability, and better user engagement. The chart below represents user preferences:

Platform	Percentage
Mobile Application	65%
Web Application	35%

2.1 Existing System

The existing career recommendation systems available today are often limited in functionality, relying mainly on manually entered data or fixed question-answer formats. Platforms like LinkedIn, Indeed, and Naukri offer job suggestions, but they are typically based on user browsing behavior rather than deep skill analysis. Most do not support resume parsing or skill extraction through machine learning. Additionally, these systems do not allow users to input natural language queries to receive personalized guidance. Some academic career counseling tools use simple filters or static datasets that lack real-time relevance. These limitations make it difficult for users to receive dynamic, skill-based recommendations that reflect both their resume content and personal interests. Moreover, most existing systems are web-based and do not provide the portability and convenience of mobile applications, which are in high demand today.

2.2 Proposed System

The proposed system, SkillSense, is an AI-driven platform that recommends career paths based on a user's resume and personal queries. It extracts skills using NLP and predicts suitable job roles through a trained machine learning model. Users can enter natural language queries to receive personalized recommendations. The system also fetches live job listings and suggests online courses to improve relevant skills. Built using Flask, it offers a smart, interactive, and user-friendly interface. The system bridges the gap between user skills and suitable career opportunities.

2.3 Future Enhancements

1. **Mobile Application Development** – Extend the system to Android/iOS platforms using React Native or Flutter for wider accessibility.
2. **Voice Query Support** – Integrate speech-to-text to allow users to ask career-related questions through voice input.
3. **Resume Scoring System** – Add an AI-powered feature to evaluate and score resumes based on job-role

fit.
4. **Integration with GPT APIs** – Use advanced language models for deeper query understanding and smarter career suggestions.

2.4 Features of the Application
1. **Resume Upload and Parsing** – Users can upload their resumes in PDF format, and the system automatically extracts key skills using NLP.
2. **Skill-Based Role Prediction** – A machine learning model predicts the most suitable job roles based on the extracted skills.
3. **AI-Style Query Box** – Users can type custom natural language queries for personalized career suggestions.
4. **Live Job Recommendations** – The system fetches real-time job listings based on the predicted or queried roles.
5. **Course Suggestions** – It recommends online courses to help users improve or learn new skills related to the target job roles.

3. Stakeholders of the Project/Application
1. **End Users (Students and Job Seekers)**
   Individuals looking for career guidance based on their skills and qualifications. They expect a simple, interactive interface with personalized role, job, and course recommendations.
2. **Career Counselors and Educators**
   Professionals who guide students in career planning. They require accurate, AI-powered insights to support their recommendations and help identify skill gaps.
3. **System Administrators**
   Manage and maintain the SkillSense platform. They need access to dashboards, usage analytics, error tracking, and system health monitoring.
4. **Job and Course Data Providers**
   APIs or platforms like JSearch, Coursera, or Udemy that supply job and course information. They expect proper API usage, request limits, and data handling.
5. **Machine Learning Engineers / Developers**
   Responsible for building and maintaining the ML models and backend logic. They require access to large datasets, model training tools, and scalable deployment.
6. **UI/UX Designers**
   Design the look and feel of the application. They depend on user research and feedback to craft an engaging, intuitive experience across devices.

4. Software & Hardware Requirements / Technology Stack Used

Hardware Requirements
- **Processor:** Quad-Core CPU or higher
- **RAM:** 16 GB or more (for smooth development and local model testing)
- **Display:** Full HD / OLED Display (for clear UI/UX design work)
- **Connectivity:** Stable high-speed Wi-Fi connection
- **Storage:** At least 50 GB free space (for datasets, virtual environments, logs, etc.)

Software Requirements
- **Operating System:** Windows 10 or higher / Linux / macOS
- **Programming Languages:**
  - Frontend: HTML, CSS
  - Backend: Python (Flask Framework)

- **Frontend Tools/Libraries:**
  - Material-UI (for responsive design components)
- **Backend Tools:**
  - Flask, scikit-learn, pandas, NumPy, pdfminer/PyMuPDF (for resume parsing and ML)
- **APIs Used:**
  - JSearch API (for job scraping)
  - Optional: Coursera/Udemy scraping or API for course recommendations
  - Optional: LM Studio or OpenAI API (for AI query assistant)

5. Methodology to be Used

At **SkillSense**, our methodology is designed to deliver intelligent, personalized career recommendations by integrating resume parsing, machine learning, AI query understanding, and real-time job/course data. The process includes the following core components:

1. Resume Data Collection and Parsing
a. **PDF Resume Upload**
Users upload their resumes in PDF format through the web interface.

b. **Text Extraction and Cleaning**
Resume content is extracted using PDF parsing libraries like PyMuPDF or pdfminer. The raw text is cleaned and structured for further processing.

c. **Skill Extraction**
Using NLP techniques (spaCy, regex, keyword matching), technical and soft skills are identified from the parsed text. These extracted skills form the primary input for prediction and recommendations.

2. Machine Learning-Based Role Prediction

a. **Dataset and Model Training**
A large open-source dataset mapping skills to job roles is used to train a classification model using algorithms like Random Forest or Logistic Regression (via scikit-learn).

b. **Role Prediction**
The extracted skills are input into the trained model to predict the top 3–5 job roles that align best with the user's capabilities.

3. AI-Powered Query Interpretation

a. **Custom Query Input**
Users can type AI-like queries (e.g., *"Suggest remote jobs in machine learning"*) into a smart query box.

b. **Query Parsing and Role Refinement**
The system parses these queries to extract intent and context (e.g., job type, domain), which overrides or refines the predicted role.

c. **(Optional)** Integration with local LLM (e.g., LM Studio) can enhance understanding of complex queries.

4. Job and Course Recommendation Engine
a. **Job Retrieval**
Using job search APIs or scraping logic, the system fetches live job openings based on either the predicted role or custom query.

b. **Course Suggestion**
The system identifies gaps between user skills and role requirements, recommending online courses (from Coursera/Udemy) to bridge them.

5. Presentation and User Interaction

a. **Web Interface (Flask + HTML/CSS)**
All results—including skills, roles, jobs, courses, and custom AI responses—are presented in a modern, interactive UI with animation and background effects.

b. **User Feedback Loop (Future Scope)**
In future versions, users can rate recommendations or provide feedback, enabling iterative model improvement.

6. Continuous Improvement
a. **Model Updating**
As more data is collected or user preferences change, the ML model can be retrained with updated datasets.

b. **Feature Expansion**
Future updates may include resume scoring, voice query support, user login systems, and mobile app integration.

7.TEAM DETAILS

NAME	GMAIL	WORK
Anik Shah	Shahanik12@gmail.com	Python, Flask, Libarires-(Spacy,numpy), JsearchApi, LM-Studio, Css
Suraj Sharma	Surajthakurst629970@gmail.com	HTML.

8.Steps of flowchart in below:
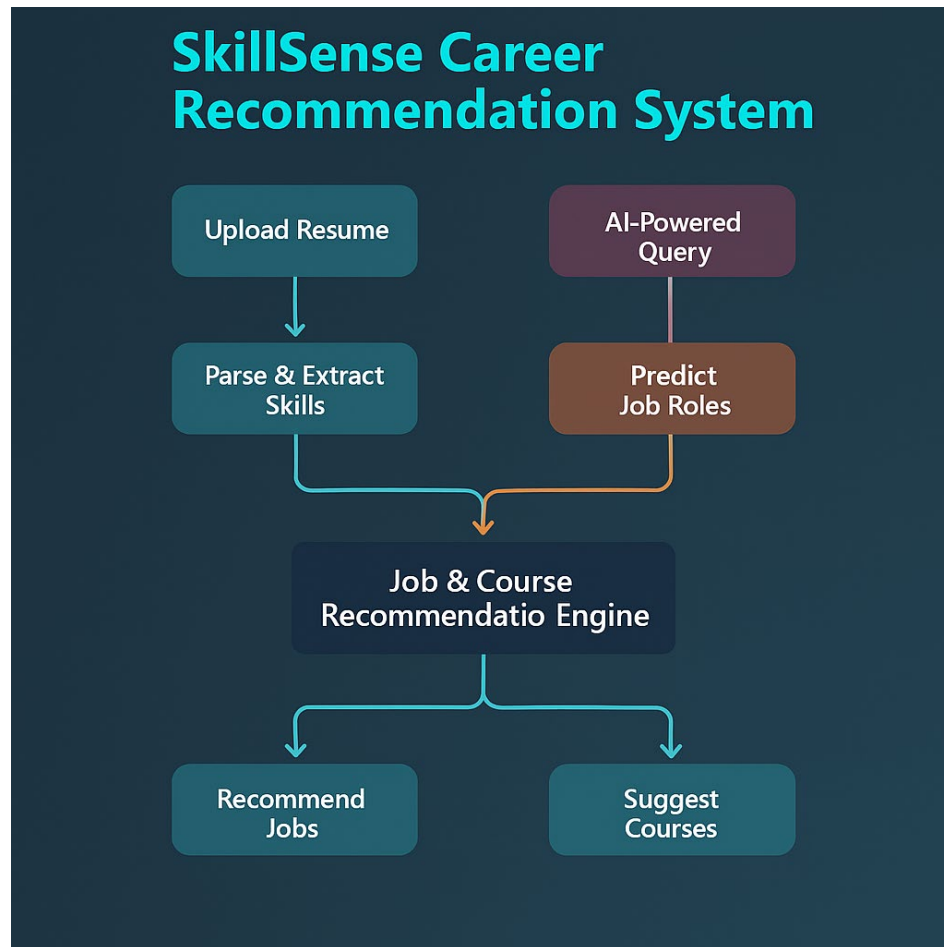
9. REFERENCES

1. Flask Framework Documentation – https://flask.palletsprojects.com/

2. scikit-learn: Machine Learning in Python – https://scikit-learn.org/stable/

3. JSearch API (Job Search Integration) – https://rapidapi.com/letscrape-6bRBa3QguO5/api/jsearch

4. LM Studio for Local LLM Query Interpretation – https://lmstudio.ai/