Software Requirement Specification Document for Career Compass

Mayar Mohamed, Habiba Shawkat, Sarah Mustafa, Shahd Elsayed
Supervised by: Dr. Fatma Helmy, Eng. Sarah Hatem

May 1, 2024

Table 1: Document version history

Version	Date	Reason for Change
1.0	15-November-2023	Proposal First Version Specifications are added
1.2	Dec-2023	Implemented GUI for the mobile application
2.0	14-Jan-2024	SRS First version's specifications are defined.

GitHub Backend repo: `https://github.com/MayarMohamed7/Career-Compass-GP`

GitHub Application repo: `https://github.com/MayarMohamed7/Career-Compass-frontend`

Contents

Abstract

In today's ever-changing job market, individuals, including students, often face challenges in identifying and pursuing a job path aligned with their interests and skills. The objective of this project is to develop a job coach recommender system designed to guide each person on their journey. Our AI Job Coach recommender system will initiate text-based conversations with users through a mobile application. Users will share their skills and interests with the bot, and the system will intelligently analyze and extract pertinent details from the user's responses. It will then provide a tailored strategy to steer individuals toward fulfilling and enjoyable job options by leveraging natural language processing, user-provided data, and machine learning.

1 Introduction

1.1 Purpose of this document

This document is intended to define the document details. Furthermore, the documentation is used to guide the developers and to be a product approval record for the needed functions. This document will explain the software implementation. The software implementation covers the algorithms and methods used in our project. The career coach app is intended to give the user the specific job recommendation needed based on the user's entered skills. The application will contain dataset extracted from wuzzuf using web Scraping technique.

1.2 Scope of this document

Firstly, this document explores some similar systems to Career Compass and provides a detailed illustration of the system's problem statement, overview, context, and scope, It also includes objectives and user characteristics for the mobile application. Furthermore, it focuses on the system's detailed functional specifications and non-functional requirements, design constraints, data design, and the system's initial class diagram. Lastly, it covers operational scenarios and the project plan.

1.3 Business Context

The project focuses on cutting-edge technologies that perfectly align with industry trends and positions the system as the go-to app for each individual in the market. By offering a personalized experience through the bot, the project works on increasing the user's engagement. This would definitely benefit the business as it offers different kinds of career guidance services and engaged users are more likely to receive valuable recommendations.

It also improves career guidance services as it can leverage this technology to provide more accurate and tailored advice to individuals seeking professional development. Businesses offering leveraged career guidance services with integrated technologies like NLP and ML can gain a competitive edge.

2 Similar Systems

2.1 Academic

Survey three academic similar systems with references.

Extracted Skill Embeddings [1]:

An excessive number of job vacancies are available on Internet recruiting platforms due to the digitization of the hiring process. Job seekers must manually go through numerous listings, comprehend the requirements, and match their talents to the available positions. While businesses aim to position people quickly, this process is time-consuming. An automated job recommendation system that efficiently matches job seekers with relevant employment is needed to address this. By expediting the hiring process, it benefits both job seekers and professionals looking to switch careers.

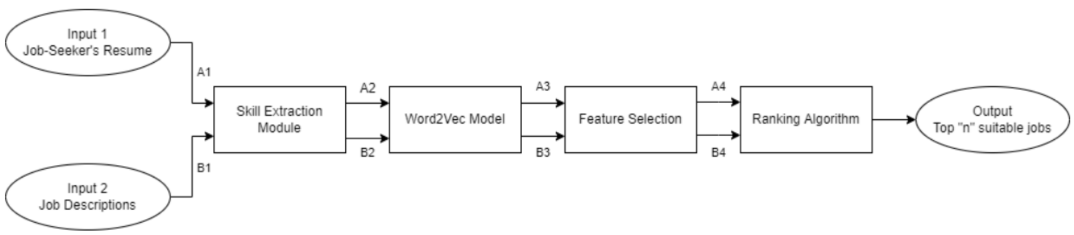

Figure 1: System

The researchers developed a framework that utilizes abilities from unstructured CVs and JDs to overcome issues with job recommendation algorithms. For job recommendations, they employed Word Mover's Distance (WMD) and cosine similarity, comparing them to TF-IDF vectors. The system eliminates the need for parsing CVs, as demonstrated by testing it on a dataset from Kariyer.net. This new method selects a representative subset of skills and turns them into dense vectors, aiding effective job matching, particularly in the IT sector. Similarly, skills are retrieved for every JD between steps B1 and B4, and sets of vectors are created using a chosen subset of the extracted skills. A ranking algorithm determines which n JDs are most suitable for a specific job seeker in the last stage, R1. The relevant sections provide details on how the two approaches differ in the final stage, which are listed below, and whose workflow is: – Skill Extraction Module – Word2Vec Model – Feature Selection – Ranking

Datasets: They gathered 500,000 IT-related JDs in Turkish and English published on Kariyer.net. These are split into two sections: The remaining ones are used for testing and assessment, and the remaining 450,000 are used to train a Word2Vec model.

Results: The proposed systems were evaluated on real datasets from Kariyer.net, which holds 25 million CVs and over 39 thousand active job positions. Text preprocessing involved replacing HTML tags, punctuation, and stop words with white spaces to capture skill keywords. Feature selection aimed to remove falsely extracted keywords, using Cosine Distances between extracted word embeddings and the mean vector. The model removed "customer support" with a Cosine

Distance of 0.41 from the document's skill set. The evaluation involved labeling each CV-JD pair as one or zero based on whether it is an appropriate recommendation considering both employer and job seeker skills. The results are presented in the Table below.

	P@1	P@3	P@5	P@10
TF-IDF with Cosine Similarity	0.6	0.48	0.41	0.35
Skill Embeddings with Cosine Similarity	0.79	0.74	0.74	0.73
Skill Embeddings with WMD	**0.95**	0.77	0.79	0.75

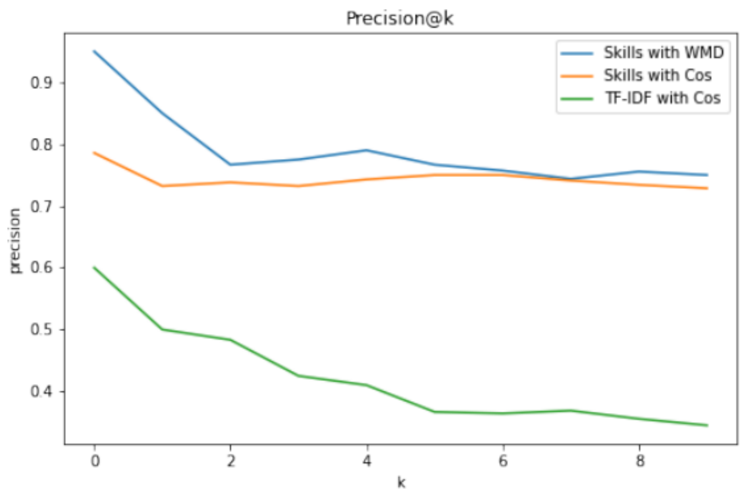

Figure 2: Precision@k values for the three models.

Criticizing the paper: Comparison with Current Methods: The main comparison made in this research is between the suggested methods and a TF-IDF baseline. A deeper examination that includes contrasts with other cutting-edge job recommendation systems or methodologies would offer a better picture of the uniqueness and potency of the suggested strategies in a larger context.

Generalization of Results: Using a particular dataset from Kariyer.net, the research shows how effective the suggested procedures are. It would be beneficial to discuss how the findings apply to datasets from other platforms or sectors, though. Evaluating the practical utility of the offered ideas requires an understanding of how well they work in different scenarios.

The paper addresses the weakness of CNN and focuses primarily on the fact that when classifying the test data, it gives a false positive, ignores the relative spatial orientation of the function, and gives a false negative due to the lack of rotational invariance. A different type of neural network - Capsule - was used to solve certain problems faced by CNN. The aim is for this new neural network to be both invariant and spatially conscious of rotations. Since tested using the MNIST dataset, the capsule has already proven to be superior to CNN. To test the capsule on higher-dimensional results, the researchers continued conducting experiments. So they applied a number of model adjustments to find the right set of configurations. Some of the model adjustments were: stacking more capsule layers, increasing the number of layers of convolution before the capsule layer, averaging the measurement of the ensemble. Bear in mind that the dataset used was the CIFAR-10 dataset, a subset of 80 million small photos. Instead of MNIST, this dataset was chosen because its images are sufficiently complex in comparison. Finally, a comparison is achieved between different models, and it was noted that the addition of a convolution layer enhances validation accuracy by 0.41 percent and that using a 4-model ensemble increases validation accuracy by 1.85 percent. This paper is based on the pre-judgment that, without making any clear distinction between them, Capsule has better accuracy than CNN.

Career recommendation using content-based filtering[2]

The main problem this paper addresses is: Frameworks have been used as a solution to the ubiquitous problem of managing an excessive amount of information in a variety of domains. By offering an organized method for managing information, these frameworks are especially made to tackle the problem of data overload. Enabling people to concentrate on information that matters in their own areas of interest is the fundamental objective. Now, frameworks that are in place are essential for reducing the negative impacts of information overload in a variety of domains by enabling users to focus their attention on relevant information that is in line with their personal interests. To ensure that these frameworks remain relevant and effective in the ever-changing field of information management, a thorough examination of their effectiveness and adaptability is necessary. This problem statement seeks to explore and understand the current state of framework implementation, identifying potential areas for improvement and optimization to enhance the user experience and facilitate more targeted information access in diverse domains. How the researchers contributed to solving the problem: Used Hybrid filtering: It is the mix of the possibility of both collaborative filtering and content-based filtering in order to reach a very high precision. even though the actual reason is mainly due to the absence of data about the domain dependencies in collaborative filtering, and the people's choices in content-based systems. Hence, the amalgamation of both leads to increasing common data and also enhances the recommendations. The proposed recommendation system has the following steps: Data preparation, Similarity coefficient using Cosine similarity, Clustering using K-means algorithm, and Web page recommendation.
The NLP used in the recommendation system is performed on the feedback that has been given by the users. It has different steps included, like sentence segmentation, tokenization, normalizing, data cleaning, and sentimental analysis. Sentence segmentation: a bit of content is divided into segments.
Tokenization:responsible for finding the word limits also considered as a base step for stemming and lemmatization in (normalizing)
Normalizing:incorporate stemming and lemmatization to lessen the quantity of in general terms to certain"root"terms.
Data Cleaning: spelling and grammar revision.
Sentiment Analysis: characterization of feelings, determining from users' words whether their feedback is positive or negative.

The main result researchers reached: the collaborative filtering approach has major 3 problems cold start, trust, and privacy, Content-based filtering's major drawback is Content analysis is important to describe the characteristics of an item. The similarity estimation is inadequate for the product description.
Criticizing the paper: the authors mentioned the drawbacks of both collaborative filtering and content-based filtering without proof. While the datasets are mentioned there's no explanation for the relativity of the dataset's attributes. The dataset used in this paper wasn't directly mentioned

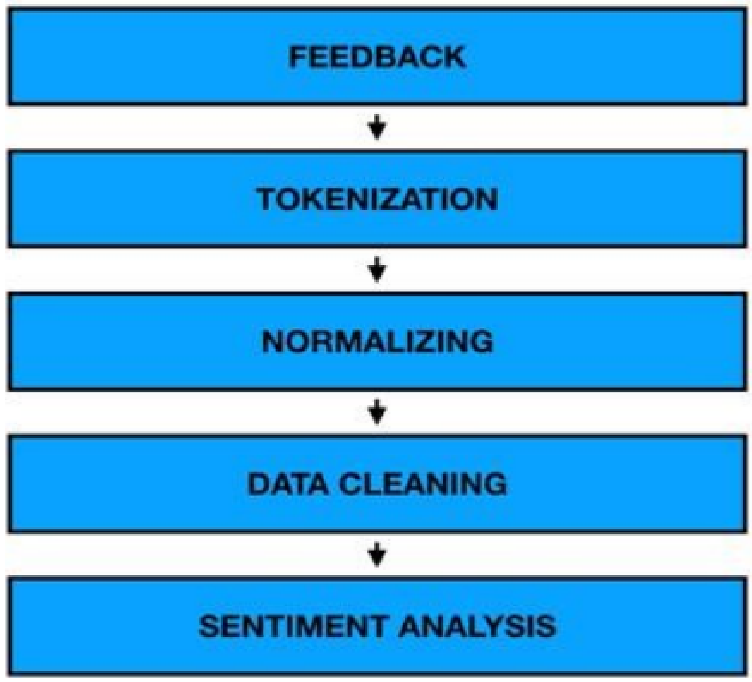

Figure 3: Flow diagram of NLP [2]

Machine Learning Approach for Determining the Relevant Skills from Job Description [3]:
In this paper, we present Skill2Vec, a novel machine learning approach inspired by Word2Vec, aimed at improving the efficiency of job recommendation systems in the context of the digitized hiring process. Addressing the challenges faced by recruiters in the information technology domain, Skill2Vec focuses on determining skill relatedness by leveraging a Skip-gram model within the Word2Vec architecture. Our methodology involves data collection, utilizing skills dictionaries for parsing and training, and extracting skills from Dice.com job descriptions. The experimental evaluation demonstrates Skill2Vec's effectiveness, with a noteworthy 78 percent relevance in the top 5 "nearest" skills for randomly selected queries. We conclude by highlighting Skill2Vec's potential in building relationship networks between skills in recruitment, emphasizing its simplicity, lower computational cost, and accuracy in generating high-dimensional word vectors. Future research directions include incorporating domain-specific information for enhanced relevance in various sectors.

Additionally, the paper critiques weaknesses in Convolutional Neural Networks (CNN) for classifying test data and introduces the Capsule network as a solution, demonstrating its superiority using the CIFAR-10 dataset. While highlighting model adjustments and improvements, the paper suggests a need for a more comprehensive comparison with other cutting-edge job recommendation systems for a broader context. Furthermore, the generalization of results is discussed, suggesting the importance of evaluating the applicability of findings to datasets from diverse platforms and sectors for a more holistic understanding.

2.2 Business Applications

Career Hub [4] :

A one-stop shop for your career is CareerHub AI. With the help of cutting-edge GPT technology, we provide tailored career advice at every stage of your path. With the help of AI, choose the perfect career path, get personalized advice, calculate your potential pay, create a cover letter, and be ready for job interviews. Users can find possible job pathways based on their abilities and interests, get individualized career advancement guidance, and learn about their market value.In addition, the platform offers access to interview questions and answers and a personal cover letter writer. Visit this link: topai.tools/t/careerub-ai-com

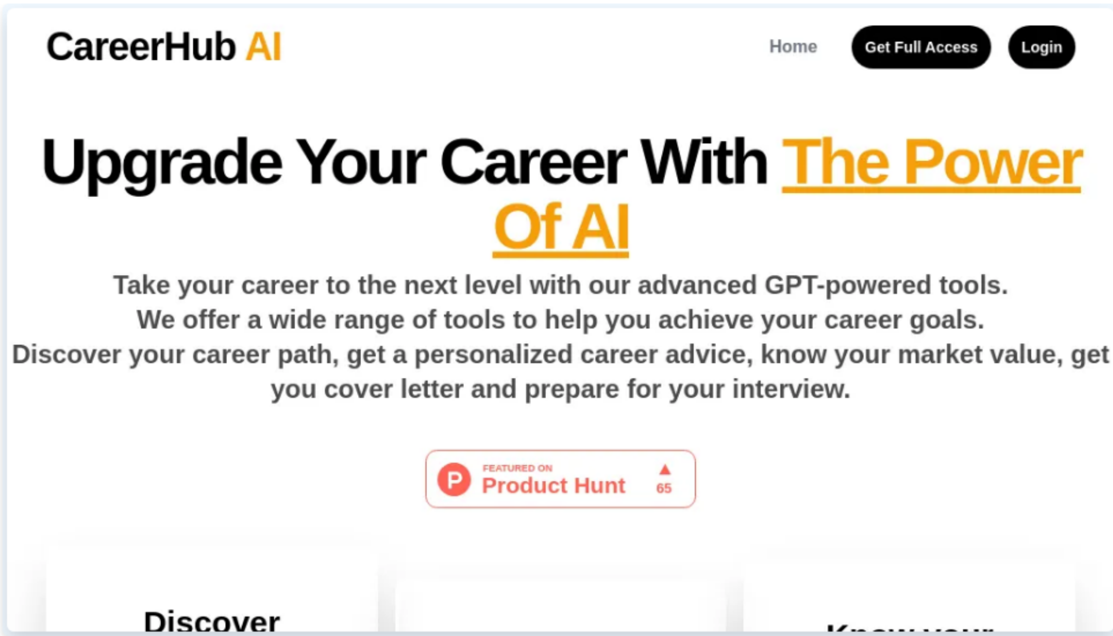

Figure 4: Career Hub

Career GPT:[5]

Career GPT is an AI platform that assists job searchers and students in high school and college in exploring career alternatives they might not have previously thought about. With choices for both realistic and creative career pathways, it leverages a chatbot driven by an API to deliver useful and realistic outcomes. .

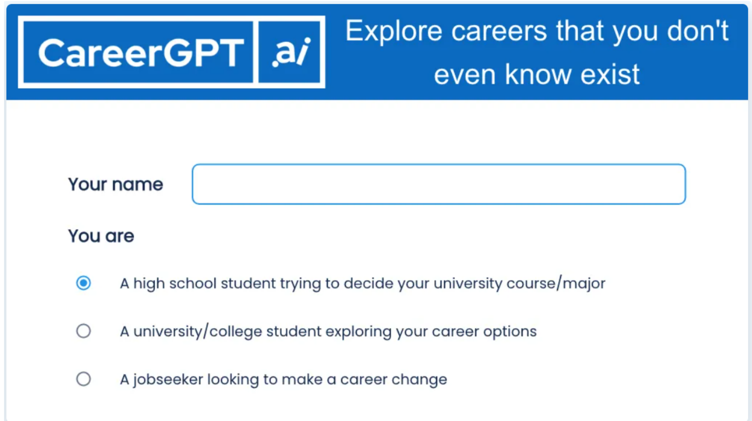

Figure 5: Career GPT

3 System Description

3.1 Problem Statement

In Today's dynamic job market, students or individuals in general have difficulties in determining and pursuing a career path that is in line with their interests and skills. The goal of this project is to create a career coach recommender system that can put each individual on track. Our AI Career Coach recommender system will engage in text-based conversation with any individual via the mobile application. The individual will provide the bot with his skills and interests and the system intelligently analyzes and extracts relevant details from the user's responses and offers a customized strategy to direct people toward meaningful and enjoyable career options by integrating the power of natural language processing, user-provided data, and machine learning.

3.2 System Overview

Our Career Compass helps users figure out what career works best for them by taking the user input which contains the user's skills and in return recommending a suitable job for the user. The system overview shown in figure 7, shows the detailed that will happen in the system:

1. We collected different datasets and then merged them into one dataset.
2. The merged dataset then undergoes some pre-processing.
3. The output of the pre-processing step will be subjected to some feature engineering.
4. We split the dataset into train and test.
5. Then the train dataset undergoes machine and deep learning algorithms to choose the best one according to the evaluation metrics.

3.2.1

User will conduct a conversation with the chatbot in which he will provide his skills manually and then his skills will be extracted. After that the extracted skills as well as the dataset will undergo some NLP pre-processing: removing unimportant characters, Lowercasing, Removing stopwords, removing numeric values, removing irrelevent columns, encoding categories.

3.2.2

Skill Extraction Module: In our skill extraction module, we extract skills from user's input with the help of a skill dictionary, where the model would search if any of the skills in the sentence match already present ones in the skill dictionary dataset. We have four main steps in this part:

1. PhraseMatcher: is a tool provided by the spaCy library that allow us to look for multi-word skills available in our skill dictionary. This way we can ensure that "Machine learning" is captured as a single skill rather than as separate entities "machine" and "learning".
2. NER: Named Entity Recognition, it checks for named entities which can include company names, technologies that could be skills.
3. POS tagger: it looks for nouns and verbs that could represent skills.
4. N-grams: it looks for bigrams in the text that are present in your skill dictionary.

Using all these components together gives us a robust skill extraction system that can identify both single-word and multi-word skills from natural language text.

3.2.3

Feature engineering which contains: feature selection, in which we select the most important features from our dataset and then feature vectorization using Term Frequency-Inverse Document Frequency. TF-IDF is a technique used to determine the frequency of words within a document (the user input) relative to a collection of documents (Job descriptions). TF-IDF will calculates the importance of skills based on their frequency in the user's input compared to the job descriptions dataset.

3.2.4

We then tried some machine and deep learning models in order to predict the correct job.

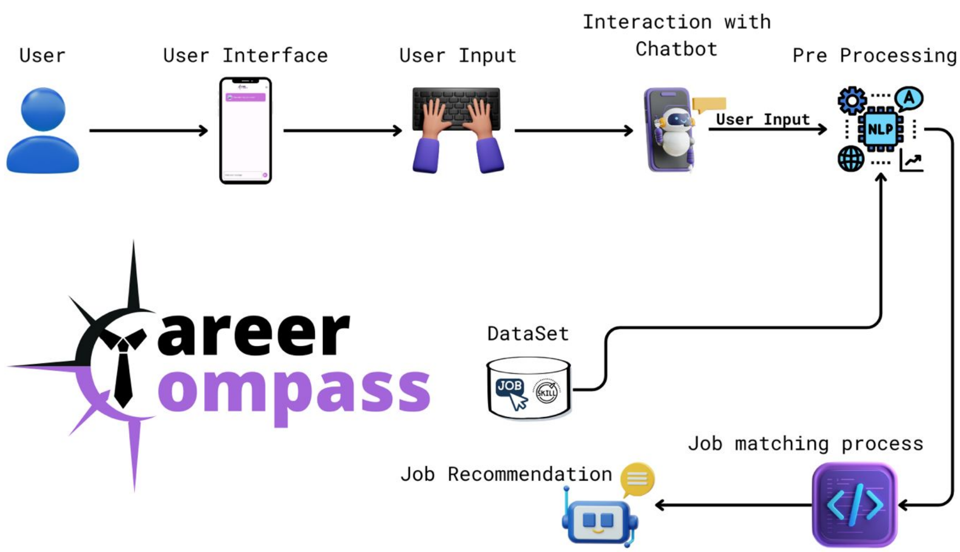

Figure 6: System Overview

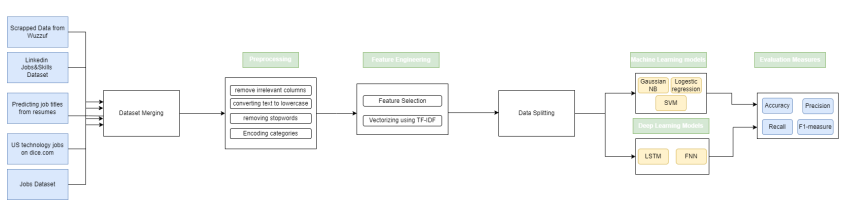

Figure 7: Detailed System Overview

3.3 System Context

The user enters his skills into the chat, Career Compass can ask more questions to gain additional info and will then recommend a job that matches the user's entered skills.

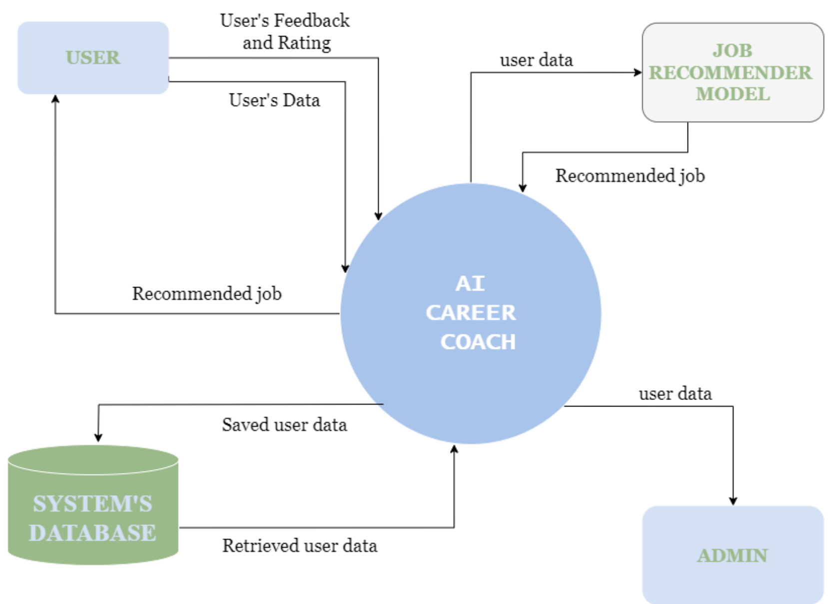

Figure 8: Context Diagram

3.4 System Scope

- User Profile Creation
- User will interact with the chatbot.
- Using Natural Language Processing (NLP) to extract the user's skills from the user's input.
- Collecting a Job description dataset using scrapping for jobs on Wuzzuf.
- Merging other datasets together to enhance our dataset.
- Extracting skills out of user's raw input.
- Using TF-IDF for our feature vectorization step.
- Using extracted skills to match a job using machine and deep learning models.

3.5 Objectives

Our main objective is to develop a user-friendly mobile application that would help individuals know what the best career path for them is. Our career coach AI would perfectly analyze the user's skills and provide a personalized career choice that would be suitable for the user saving them money considering that a normal career coaching session costs quite a lot as well as saving time as the mobile application would be easy to access simplifying the process of making career decisions on the go.

3.6 User Characteristics

- Users should be able to use mobile phones.
- Users should be able to converse in English with the chatbot.
- Students who can't seem to choose a job.
- University students who are still not certain about their major choice.
- Employers intending to switch jobs.

4 Functional Requirements

4.1 System Functions

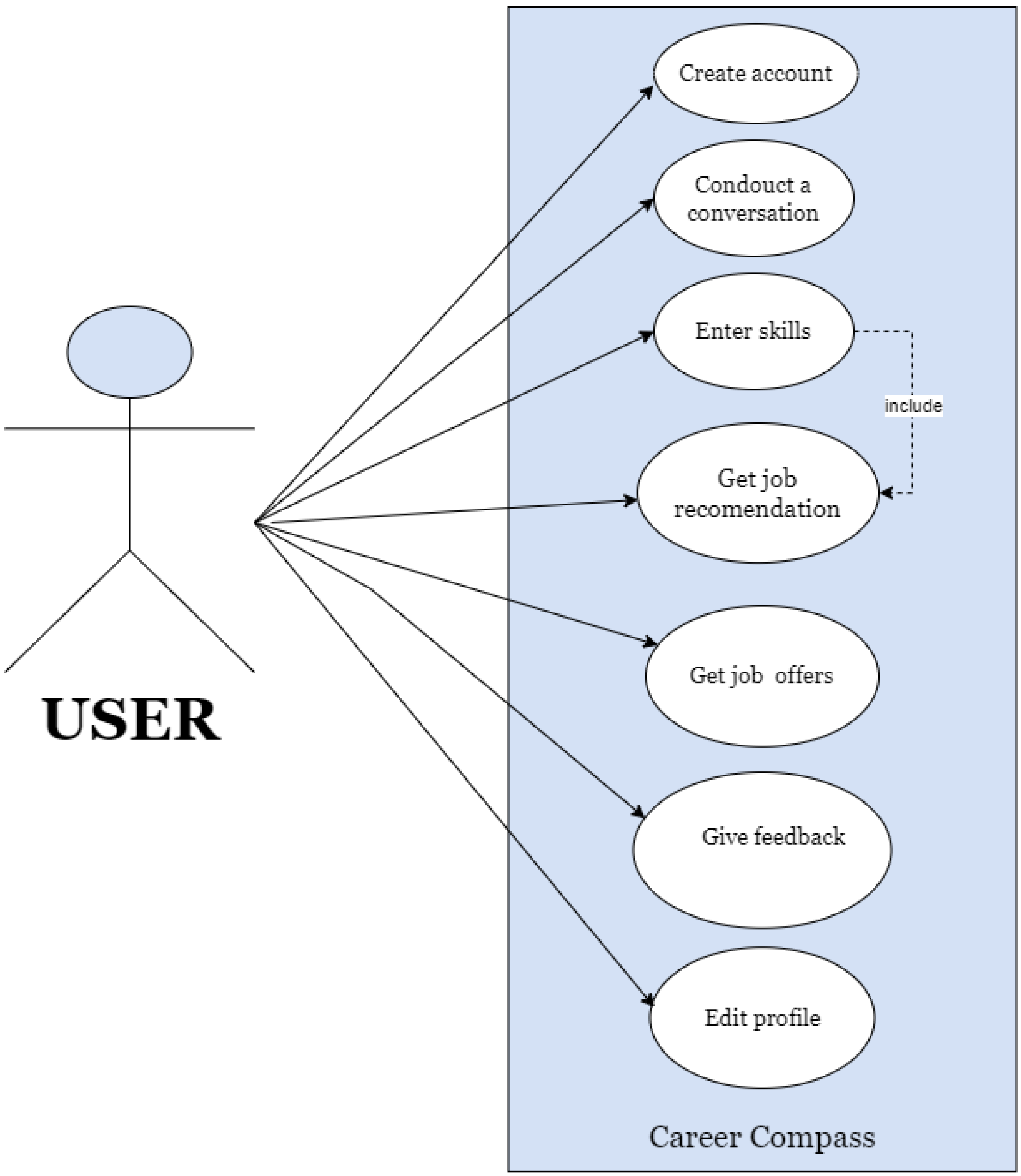

Figure 9: User Use case

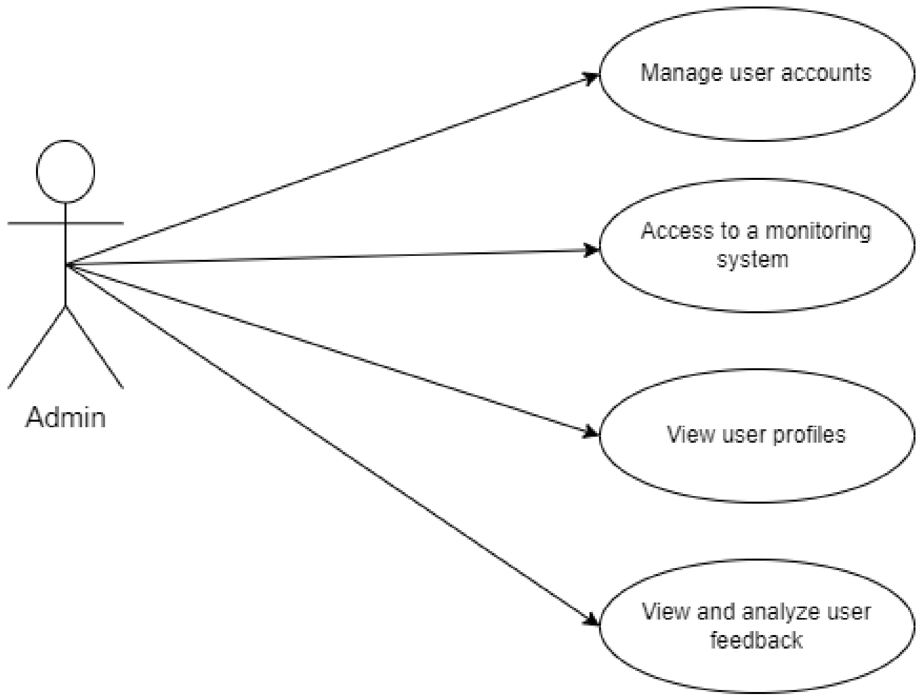

Figure 10: Admin Use Case

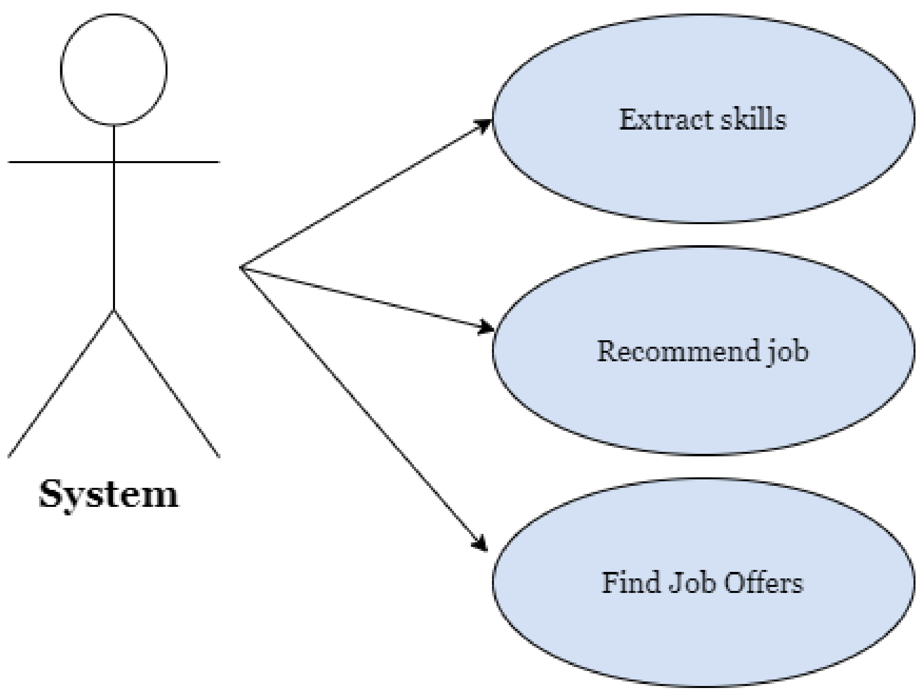

Figure 11: System Use Case

ID:01 User should be able to register using email and password.
ID:02 User should be able to conduct a conversation with the chatbot.
ID:03 User enters his skills into the chat.
ID:04 System could ask questions to gain more info about user's skills and strengths.
ID:05 User can edit his profile.
ID:06 Job recommender model has to recognize and extract skills from the user input.
ID:07 Job recommender model will match the user skills with a specific job.
ID:08 User can give feedback after the job recommender model provides him with a suitable job.
ID:09 User gets job offers that are popular in today's job market.
ID:10 Admin should be able to view and manage user accounts, including user registration information.
ID:11 Admin should have access to a monitoring system that displays user-chatbot interactions in real-time. This includes chat history, user inputs, and chatbot responses.
ID:12 Admin should have the ability to view user profiles. This includes skills, interests, and any other relevant information provided by the user.
ID:15 Admin should be able to view and analyze user feedback received after the AI suggests suitable jobs. This includes reviewing feedback comments and analyzing trends for system improvement.

4.2 Detailed Functional Specification

Table 2: Enter Skills Function Description

Name	Enter Skills
Code	ID:03
Priority	High
Critical	8/10.
Description	The system should allow users to enter their skills into the chat interface.
Input	Users input skills in natural language format.
Output	System processes and records the user-entered skills.
Pre-condition	User is logged into the system and engaged in a chat session.
Post-condition	User's skills are successfully captured and stored for further analysis.
Dependency	Dependent on the chatbot interface being active and responsive.
Risk	Risk of misinterpretation of skills if the natural language processing (NLP) is not robust.

Table 3: Asks Questions Function Description

Name	Asks Questions
Code	ID:04
Priority	High
Critical	9/10
Description	The system should ask follow-up questions to gain more information about users' skills.
Input	User's initial skills input from the chat.
Output	System generates relevant follow-up questions based on the initial input.
Pre-condition	User has entered skills into the chat.
Post-condition	System receives additional details about the user's skills and strengths.
Dependency	Relies on a well-designed natural language processing (NLP) algorithm.
Risk	Risk of user frustration if the questions are too frequent or not contextually relevant.

Table 4: Match Skills with Jobs Function Description

Name	Match Skills with Job
Code	ID:05
Priority	High
Critical	9/10
Description	Algorithm that matches users' skills with recommended jobs.
Input	User's paragraph, user's extracted skills.
Output	Number of recommended jobs.
Pre-condition	User entered Email, Full Name, Password, and user's paragraph input.
Post-condition	Recommends jobs based on users' entered skills, user rates the system, and gives feedback.
Dependency	User's input paragraph, skills extraction.
Risk	Unstructured user's input may match users' skills with jobs they are not interested in.

Table 5: Get Job Offers Function Description

Name	Get Job Offers
Code	ID:09
Priority	Medium
Critical	7/10
Description	System provides users with available job offers for the job title matched with their skills.
Input	Job title approved by the user.
Output	Links to job offers that satisfy the job title.
Pre-condition	User approves the recommended job.
Post-condition	User explores the provided job offers and may apply if interested.
Dependency	The availability and accuracy of job offers depend on external job databases or APIs.
Risk	The system might not find suitable job offers based on the job recommendation.

Table 6: Extract Skills from User Input Function Description

Name	Extract Skills from User Input
Code	ID:06
Priority	High
Critical	8/10
Description	Function extracts skills from the user's input to enter the processing steps.
Input	User's paragraph.
Output	Skills of users separated by "|".
Pre-condition	User registers in the system and conducts a chat with the chatbot.
Post-condition	System provides the user with a suitable job recommendation.
Dependency	The function ID:06 depends on ID:03
Risk	Connection interruption, the data is lost and user needs to re-fill the form

5 Design Constraints

5.1 Hardware Limitations

- The program was created specifically with the Android operating system in mind.
- A stable internet connection is needed for the system to work perfectly.

6 Non-functional Requirements

6.1 Security

- The User's chat history can only be accessible through his account and will not be visible to other users.
- During registration, the user password will be hashed to ensure the overall security of the user's data.

6.2 Accuracy

The system gives an accurate job recommendation that is relevant to the user's entered skills.

6.3 Usability

The system interface should be designed for seamless user interaction.

6.4 Response time

The system doesn't let the user wait a long time for the output and replies to the user's input as fast as possible.

7 Data Design

We have 2 datasets, The First one is a skill dictionary dataset.The skill dictionary dataset is a comprehensive collection encompassing various skills, not limited to technical expertise. This dataset serves as a foundation for training the system to recognize and extract skills from user inputs and it works as a skill dictionary. With multiple columns representing diverse skills, it aids in enhancing the natural language processing (NLP) capabilities of the system. The second dataset is derived from Wazzuf through web scraping. It captures the extracted skills from user inputs, aligning them with the skill dictionary dataset. This dataset plays a crucial role in the job recommendation process. By matching user-expressed skills with the skill dictionary, the system can identify suitable job recommendations. The integration of this dataset ensures that the career coaching application offers personalized and relevant suggestions to users based on their skills and interests.

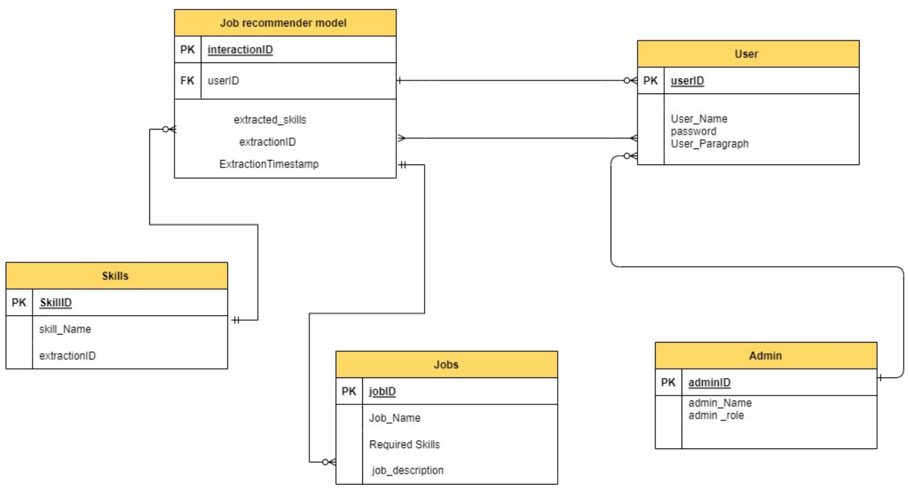

Figure 12: ER Diagram

8 Preliminary Object-Oriented Domain Analysis

Initial Class Diagram

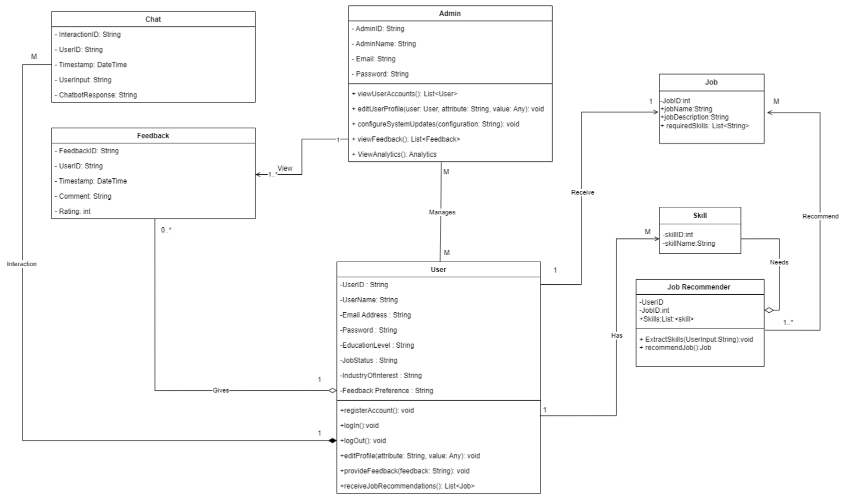

Figure 13: Class Diagram

9 Operational Scenarios

1. **Scenario 1:** The user as shown in figure 6, can start with opening the application and registering. Following that, the user can conduct a conversation with the chatbot where he will provide the bot with his skills and then will receive a suitable job based on his provided skills.

2. **Scenario 2:** The pre-processing as shown in figure 7, occur on the user raw input text and the JDs dataset. The input will first undergoes all the pre-processing techineqes mentioned starting from removing unimportant characters, lowercasing, removing stopwords, lemmatization.

3. **Scenario 3:** The user as shown in figure 13, can ask the chatbot to provide links to job offers for the matched job the bot previously provided.

4. **Scenario 4:** The database, as shown in figure12, is used to save all user related data his account information, skills he provided the chat with and lastly the matched job depending on his skills that the bot replied with.

10 Project Plan

Detailed plan from Proposal to SDD.

Task	Start date	End Date	Duaration	Member
Supervisor	20/8/2023	30/8/2023	11 days	All team members
Idea	1/11/2023	9/11/2023	9 days	All team members
Information collection and researches	2/11/2023	14/11/2023	13 days	All team members
Survey and Pre-Proposal	9/11/2023	15/11/2023	7 days	All team members
Document Submission	10/11/2023	15/11/2023	6 days	All team members
Proposal Presentation 10%	16/11/2023	19/11/2023	4 days	All team members
Meetings with Career Coach	2/12/2023	10/12/2023	9 days	All team members
Information collection and researches	9/12/2023	25/12/2023	17 days	All team members
Working on enhancing the skills dictionary database	25/12/2023	10/1/2024	17 days	All team members
Scrapping Dataset From Wazzuf	25/12/2023	13/1/2024	20 days	All team members
SRS Preperation	2/12/2023	10/1/2023	40 days	All team members
SRS Presentation 30%	15/1/2024	17/1/2024	3 days	All team members
Using TF-IDF to determine the freq	20/1/2024	4/2/2024	16 days	All team members
Using cosine similarity to increase accuracy for job matching	20/1/2024	4/2/2024	16 days	All team members
SSD Prepration	25/1/2024	21/3/2024	16 days	All team members
SSD Presentation 65%	1/3/2024	25/3/2024	57 days	All team members

Figure 14: project plan

11 Appendices

11.1 Definitions, Acronyms, Abbreviations

Term	Stands For
TF-IDF	Term Frequency-Inverse Document Frequency
TF	Term Frequency
CNN	Convolution neural network
GUI	Graphical User Interface
SVM	Support Vector Machine
NLP	Natural Language Processing
IT	Information Technology
API	Application programming interface

References

[1] Atakan Kara, F Serhan Daniş, Günce K Orman, et al. "Job Recommendation Based on Extracted Skill Embeddings". In: *Proceedings of SAI Intelligent Systems Conference*. Springer. 2022, pp. 497–507.

[2] Tanya V Yadalam, Vaishnavi M Gowda, Vanditha Shiva Kumar, et al. "Career recommendation systems using content based filtering". In: *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. IEEE. 2020, pp. 660–665.

[3] Le Van-Duyet, Vo Minh Quan, and Dang Quang An. "Skill2vec: Machine learning approach for determining the relevant skills from job description". In: *arXiv preprint arXiv:1707.09751* (2017).

[4] *CareerHub AI*. `https://careerhub-ai.com/`. Accessed: May 1, 2024.

[5] *Career GPT*. `https://careergpt.ai/?via=topaitools`. Accessed: May 1, 2024.