Summer Fellowship Report

On

Employer Recommendation System

Submitted by

Harshit Jain

Under the guidance of

Nancy Rachel Varkey

Senior Project Manager

Spoken Tutorial, IIT BOMBAY

23rd April 2020-3rd June 2020

Acknowledgement

This report has been prepared for the internship that has been done virtually in order to study the practical aspects of the Recommendation System and its implementation in the real field with the purpose of learning and exploring for professional development.

The aim of this internship is to understand and implement the work flow of recommending jobs to students and deserving students to employers and to familiar with Django and basic recommender systems algorithms.

I would like to express my sincere gratitude to my project head Nancy Rachel Varkey who has given me this valuable opportunity to work in such a learning environment to get industrial-based experience. I am also grateful to the enthusiastic team of Spoken Tutorial especially the Technical team. The Technical team comprising of two great mentors, Sir Abhijit Bonik and Mam Kirti Ambre for their constant supervision and their support was truly helpful in completing my tasks and my internship.

Harshit Jain.

Content

1.Introduction

Spoken Recommendation System

It is a multi-user system that recommends jobs to students on the basis of their skillset and deserving students to employers on similarity basis of their requirements.

It consists of two roles – Employer and student. It takes employer and company details from the registered employer. Employer can post his jobs in the system and can review the students by seeing their profile. Students can apply for the jobs by giving their details which include personal and academic details and can get the most suited jobs as per their skills.

Technology used: Django, Html, CSS, java-script, basic recommender algorithms (content filtering).

Scope:

In recent years, the volume of data has grown exponentially. The evaluation of such data or extraction of information is very difficult due to its high volume. Hiring the deserving students is the challenge faced by all companies.

Outsourcing companies in India received more than 1 million jobs application. Interested students fill their details in form of resumes etc. Once the applications are received, they are subjected to careful scrutiny by screeners.

A Recommendation System is expected to provide recommendations in 2 ways:

1.firstly recommending most eligible students for the specified job to the employers.

2.recommending jobs to the aspiring candidates according to their matching profile.

2. System description:

This system will take details of employers, job type they want to offer, skills they are looking for in the applicants from employee with all relevant details and from student it will take their academic details, cv, skills.

1.Information-Extraction:

Students skills, performances, interest in job-type and Employers organisation details and skills they required are extracted and put in a single frame for feature engineering.

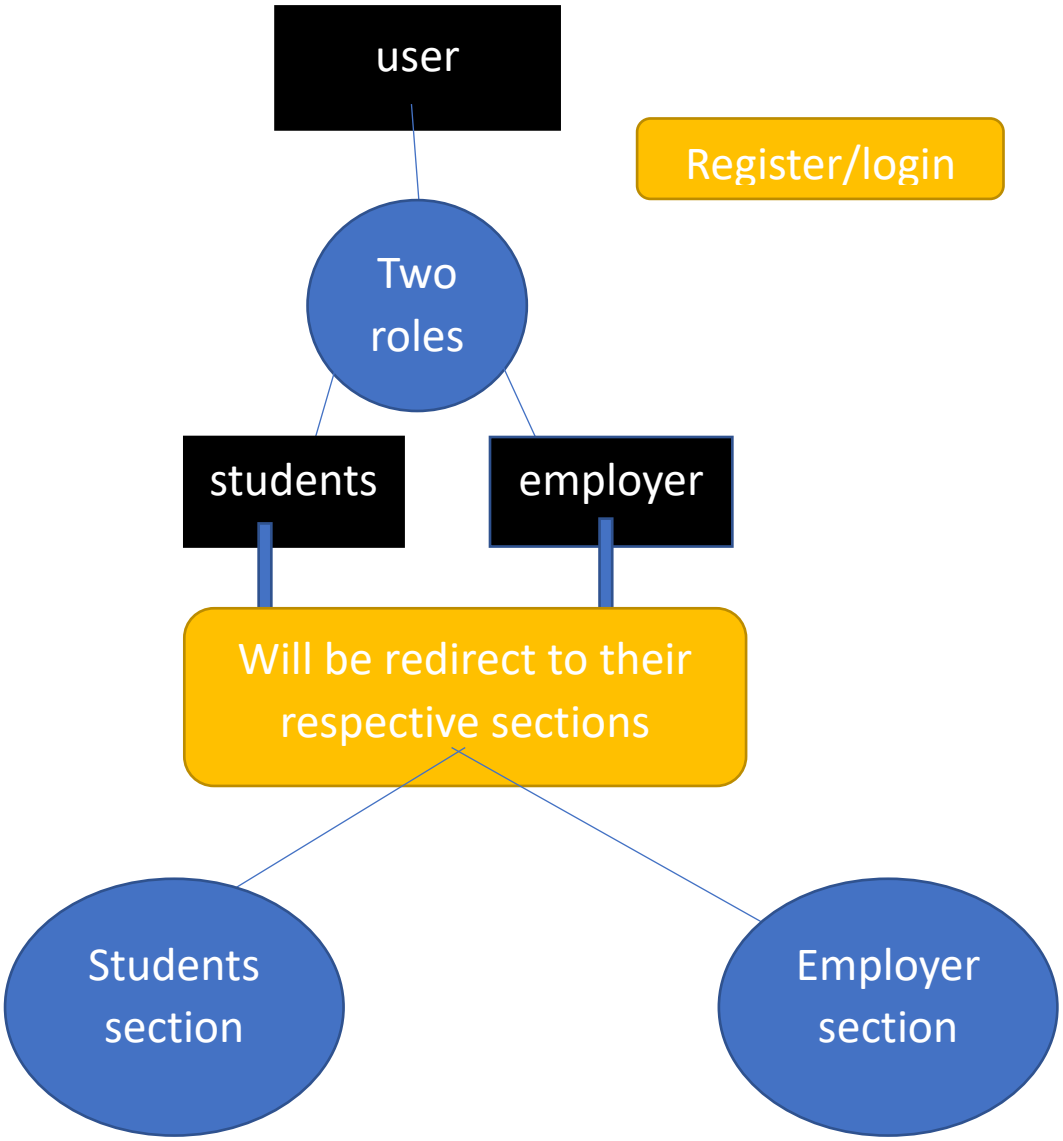
2.Similarity:

Used Jaccard similarity between employer's requirements and student skills.

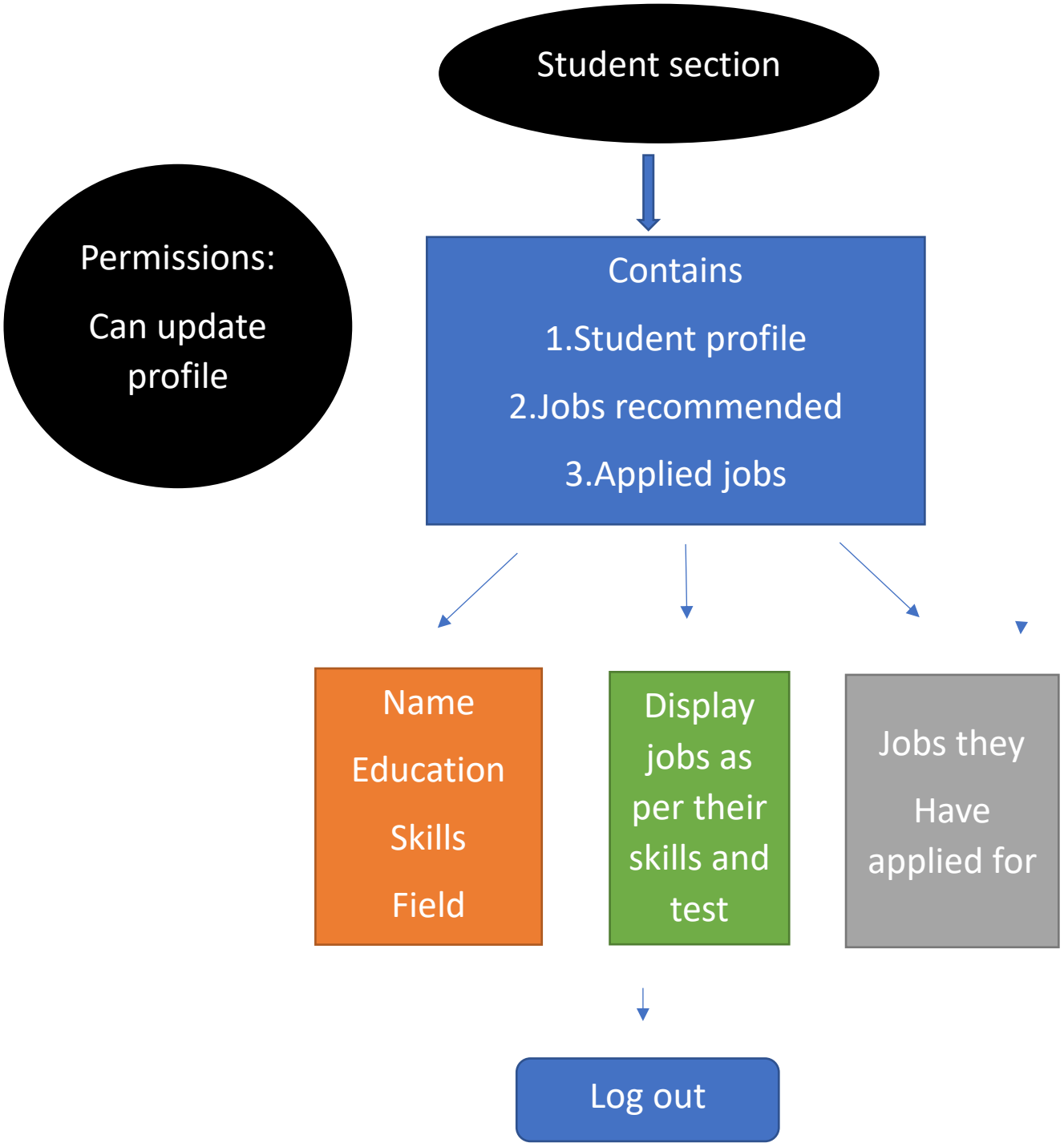
3.Techniques:

Content based filtering which is done on the basis of features. It is based on similarity of attributes whereas collaborative filtering based on similarity of interactions.

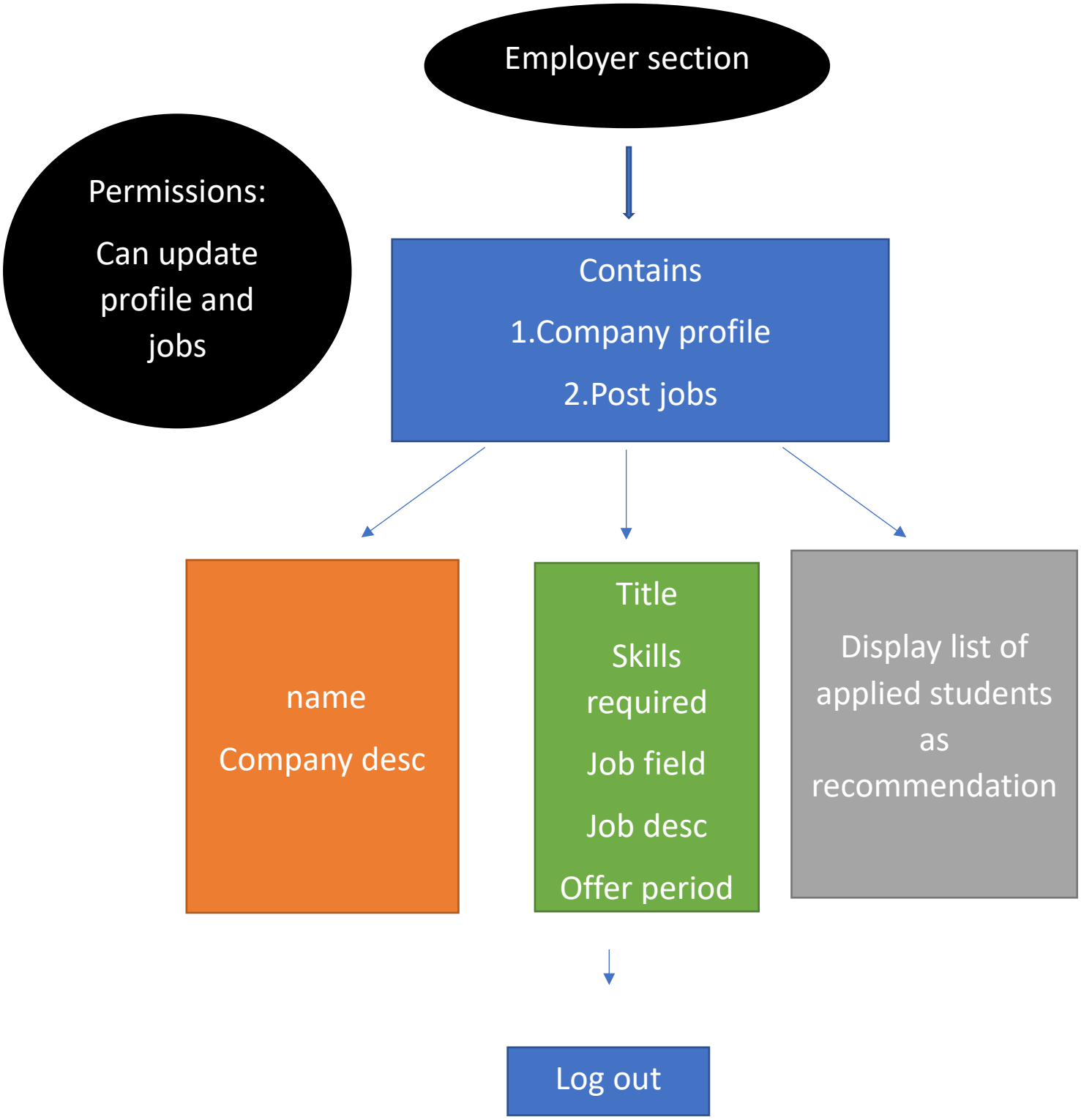
3. Diagrammatic flow of System



a) Student Section



b) Employer section:



Problems:

1. Selection of recommender algorithms for better recommendation.
2. Flow of multiuser system.
3. Database creation.
4. Role privacy.

Solution:

I have researched about the algorithms and found that content filtering is suitable as per the features. I learnt about databases and their relation from Django doc. The main thing in any website is privacy. I have created decorators to filter the roles while authentication. Visibility of student's profile to employer was a challenge for me. I have used reverse engineer technique to learn the concepts of accessing and showcasing data. The solution is to search, understand and implement .

Recommendation approach and Computation methods:

1.Content based approach:

The document representations of employers have to be matched with users. The dataset of required skills(features) will be represented as matrix form. The textual similarity will be used by system to recommend job-types to Students as per their skills. The applicants of specific job-type and excellent academic performance will be shown to employers.

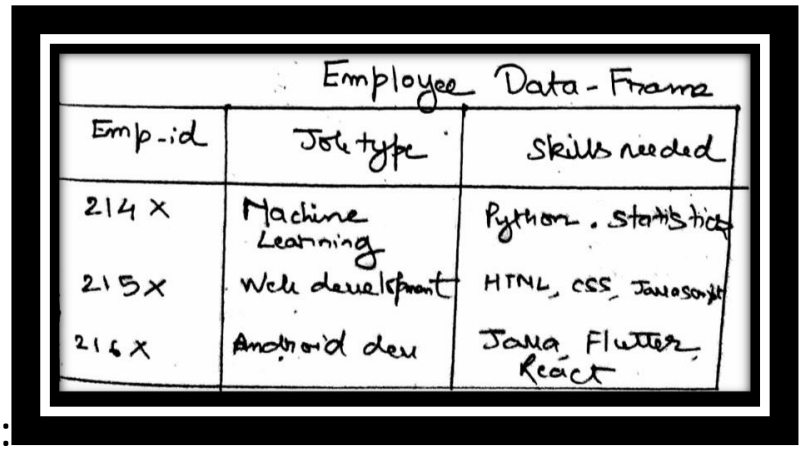
2.Jaccard Similarity:

This similarity is used to find the matching between student's skills and employer requirements.

```
def get_jaccard_sim(str1, str2):
    a = set(str1.split())
    b = set(str2.split())
    c = a.intersection(b)
    return float(len(c)) /
(len(a) + len(b) - len(c))
```

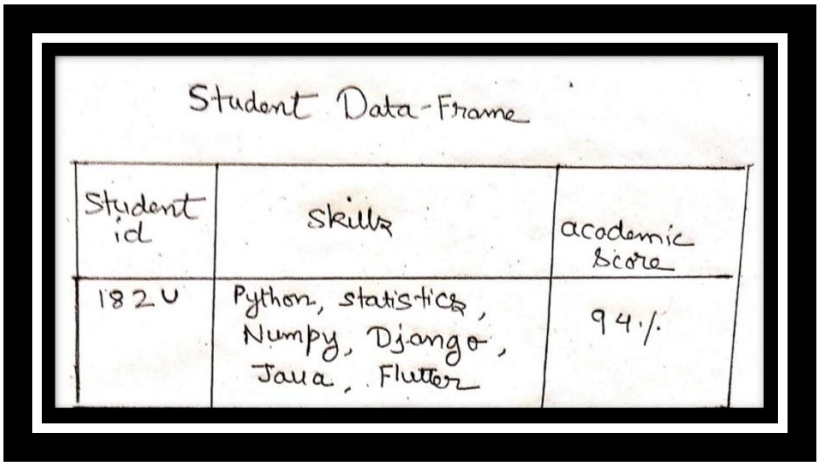
Mechanism:

:



Employee Data-Frame

Emp-id	Job type	Skills needed
214 X	Machine Learning	Python . statistics
215 X	Web development	HTML, CSS, Javascript
216 X	Android dev	Java, Flutter, React



Student Data-Frame

Student id	skills	academic score
182U	Python, statistics, Numpy, Django, Java, Flutter	94%

Implementing Jacard similarity:

Total unique skills= Python, statistics, numpy , Django, Java, Flutter, React, HTML, CSS, Java-script

no of unique skills=10

for Machine Learning jaccard sim=intersection/total skills

1. student will be recommended Machine Learning as skills having=skills required.

2. student will not be recommended for Web development as sim=0

3. 2/3 sim for Android development which is more than 50 per so it will be recommended to him.

Architecture Overview

The proposed system will be a Django based application which will allow employers and students to register their details. Students can apply for the job online. Employees can browse through the posted resume and select suitable candidates.

Features are:

Filter, blog, search facility for students according to their jobs, job recommendation. Privacy level for different companies, recent jobs will be displaying on homepage. Employer can post jobs, select students and get student recommendation

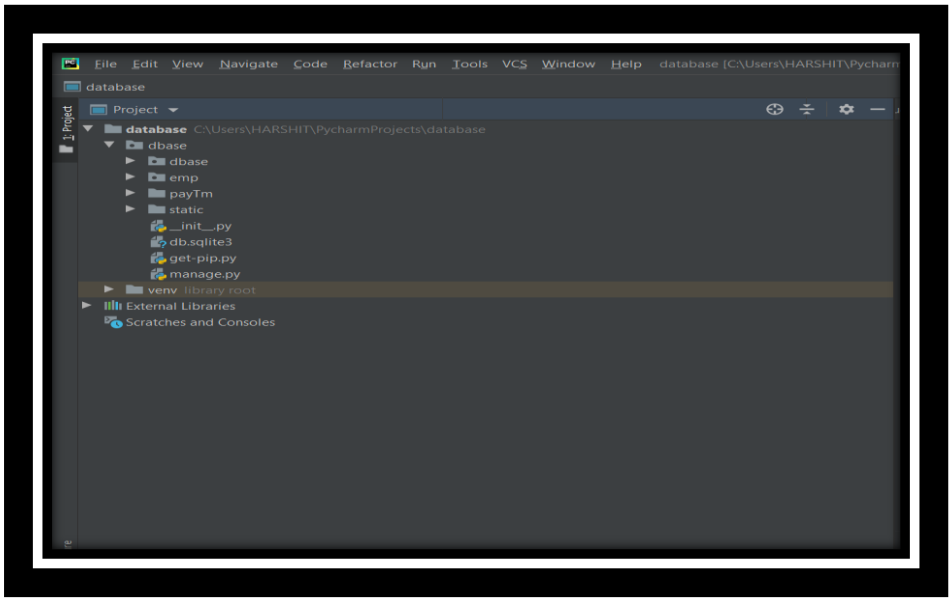
Methods:

Used content based filtering to recommend the user.
1.recommend jobs depending on skill-set matching with past user.

2.finding similarity of students skill-set with job details.

3.sort the similarity value and recommend top n jobs to them.

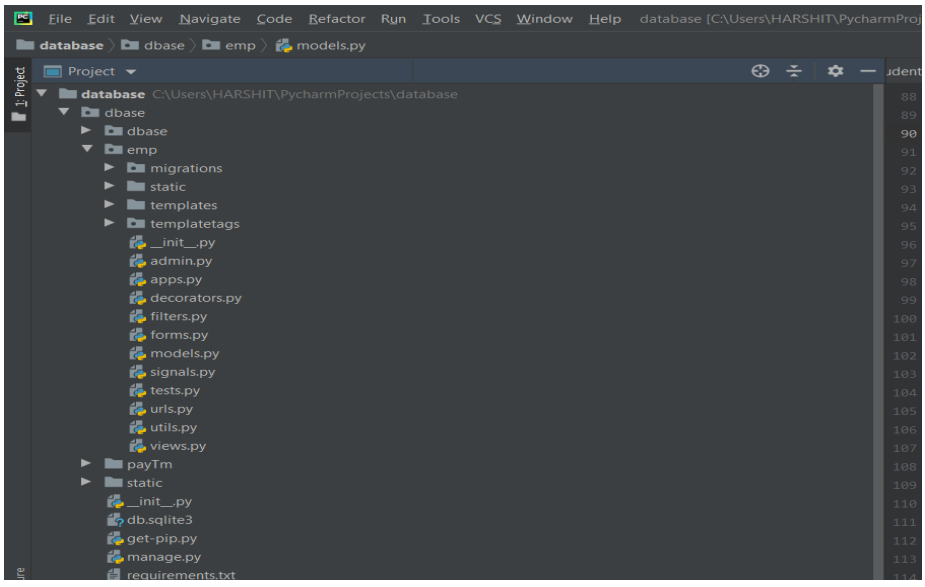
Project Structure:



Inside database folder [Employer recommendation System] has project name: dbase

App name: emp

Directories: static[images, js, CSS] and Paytm [payment methods] .



In side emp[app-name] are migrations, static, templates and python files.

Setup an Existing Django Project.

Step 1: Grab a copy of the project.

```
git clone git@github.com:Spoken-tutorial/Spoken-Reccomendation-System.git
```

Step 2: Create a virtual environment and install dependencies.

```
mkvirtualenv Employer recommendation System
venv\Scripts\activate
pip install -r requirements.txt
```

Step 3： Initialize database.

```
cd dbase
python ./manage.py syncdb
python ./manage.py migrate
```

Step 4： create a new superuser for the admin.

```
python ./manage.py createsuperuser
```

Step 5: Run the development server to verify everything is working.

```
python ./manage.py runserver
```

Future scope:

In this field of job recommendation system there is a large scope for future work such as:

- By using different similarity measure we can see which gives the most accurate answer when compared with the other similarity measures.
- We can consider large number of parameters by giving them associated weights for more accurate Content Based recommendation.

We can perform natural language processing to extract information from job-seekers resume and then recommending him the jobs

Integrate with spoken tutorial and add monthly subscription for users.

References:

1.Django documentation.

https://docs.djangoproject.com/en/3.0/

2.Spoken tutorial

3. Recommendation system articles.

https://developers.google.com/machine-learning/recommendation

Conclusion:

This portal will provide an efficient search for online information on job vacancies for students.

The main goal of this project is to attempt to select the right graduates based on industry needs.

Project focused on improving the online job portals and tried to reduce problems that are encountered in existing systems by developing a knowledge system.

The advantages of the new portal are as follow:

1.Achieve the main targets of the project.

2.Standard content services

3.Good management level and Flexibility.