PROJECT REPORT

Team ID : PNT2022TMID33644

SKILL/JOB RECOMMENDER

1. INTRODUCTION

Having lots of skills but wondering which job will best suit you ? Don't need to worry! we have come up with a skill recommender solution through which the fresher or the skilled person can login and find the jobs by using search option or they can directly interact with the chatbot and get their dream job.

1.1Project Overview

To develop an end to end web application capable of displaying the current job openings based on the skillset of the users. The users and their information are stored in the Database. An alert is sent when there is an opening based on the user skillset. User will interact with the chatbot and can get the recommendations based on his skills. We can use job search API to get the current job openings in the market which will fetch the data directly from the webpage.

1.2Purpose

- To recommend suitable jobs to the candidates
- To make the recruitment system more secure and easier.
- To suggest skills to the users so that they can acquire them and get a suitable job.
- To show the deserving candidates to the companies who may fit into their working culture.
- To make the process of job hunt easier for the fresher as well as experienced candidates.

2. LITERATURE SURVEY

The JRS has been studied from many aspects.Summarized the categories of existing online recruiting platforms and listed the advantages and disadvantages of technical approaches in different JRSs. For example, bidirectional recommendation is accomplished but only binary

representation is allowed in the probabilistic hybrid approach. We also had done some research on feature extraction, resume mining, recommendation approach, ranking, and explanation for the JRS explained that user profiling and calculating similarity are presented as the prevailing process of a JRS.

2.1Existing Problem

Recently, job recommendation has attracted a lot of research attention and has played an important role on the online recruiting website. Different from traditional recommendation systems which recommend items to users, job recommender systems (JRSs) recommend one type of users (e.g., job applicants) to another type of users (e.g., recruiters). In particular, job recommender system is designed to retrieve a list of job descriptions to a job applicant

2.2References

- Anika Gupta, Dr. Deepak Garg "Applying Data Mining Techniques in Job Recommender System for Considering Candidate Job Preferences "International al Conference on Advances in Computing, Communications and Informatics (ICACCI) 2014.
- Xiangpei Hu, Lirong Wu, Chao Li "SMS-based Mobile Recommendation System for Campus Recruitment in China", 10th International Conference on Mobile Business 2011
- Ronak V Patil, Sneha R Gadekar, Prashant P Chavan, Vikas G Aher, "Desktop based recommendation system for campus recruitment using MAHOUT", Multidisciplinary Journal of Research in Engineering and Technology, Volume 2,sue 2, Pg.480-485
- R. Munger, ‘‘Technical communicators beware: The next generation of high-tech recruiting methods.’’ IEEE Trans. Professional Communication, vol 45, pp. 276-290, 2002.

- Anika,"Applting data mining for job recommendation by exploring job preferences",computer science and engineering department, Thapar university, Patiala-147004

2.3Problem Statement Definition

Dealing with the enormous amount of recruiting information on the Internet, a job seeker always spends hours to find useful ones. Many times, people who lack industry knowledge are unclear about what exactly they need to learn in order to get a suitable job for them. We address the problem of recommending suitable jobs to people who are seeking a new job. We formulate this recommendation problem as a supervised machine learning problem

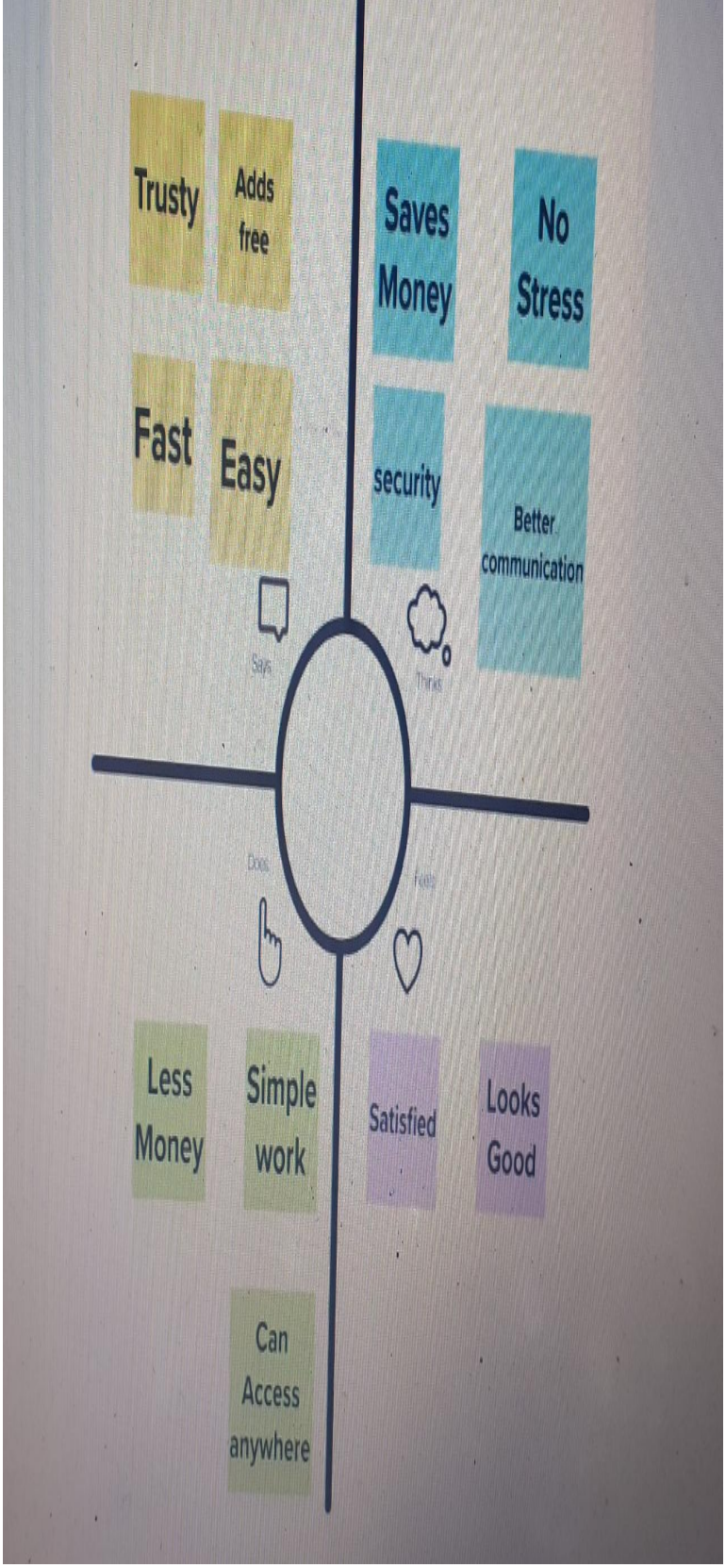
3. IDEATION & PROPOSED SOLUTION

After a survey performed in literature we have studied the different recommendation systems results and choose the matching method that contains two types of matching which provides the better matching result than existing one.

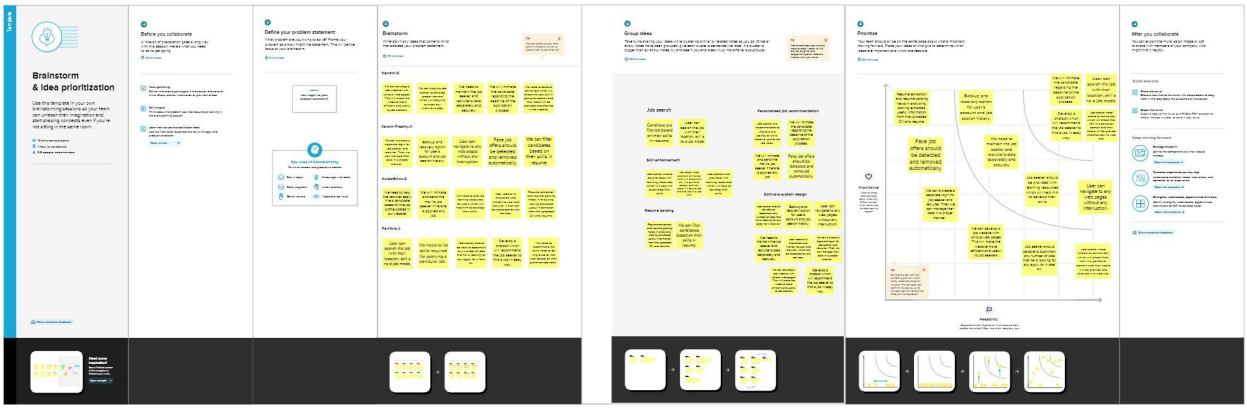
Due to existing systems data handling techniques, effective recommendation results are not upto that extends. And with these all reasons we got motivated to developed the job recommendation system using profile matching using web crawling for TPO achieve the following:

- Two types of matching provides the better results for job recommendation.
- Android application facility
- Key word based job searching using web crawling. Therefore, the proposed paper presents the structure of the developed system, and the better job recommendation than the existing systems.

3.1 Empathy Map Canvas


Trusty
Adds free
Saves Money
No Stress
Fast
Easy
security
Better communication
Says
Thinks
Does
Feels
Less Money
Simple work
Satisfied
Looks Good
Can Access anywhere

3.2Ideation & Brainstroming



3.3Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	• Dealing with the enormous amountof recruiting information on the Internet, a job seeker always spends hours to find useful ones. Many times, people who lack industry knowledge are unclear about what exactly they need to learn in order to get a

		suitablejob for them. We address the problem of recommending suitable jobs to people who are seeking a new job. We formulate this recommendation problem as

		a supervised machinelearning problem.
2.	Idea / Solution description	• The skills are extracted from the job seeker's resumeusing the TF-IDFtechnique. The job seeker's profile may get outdated sometimes as they fail to update the resume regularly. • The dynamic behaviour of the job seeker is noted by observing the jobs he applied for. So, the dynamic features are extracted, whichare an updatedversion of basicfeatures, by makinga statistic at regular intervals. • The dynamic recommendation engine works as follows: A collaborative user-based filtering algorithm is used initially to overcome the

		cold-start problem. It takes the features extracted from the jobseeker's profile and the features extracted from the job description, computes the similarity between the two using Euclidean distance, and recommends the top k similar jobs appliedto generate the initial recommendation jobs.

		• The system provides the initial recommendation to the job seeker and records his behaviour. Thus, we will be able to arrive at a set of jobs in which the job seeker is interested and a set of jobs in
		which he is notinterested. The extended new basic features help in updating the job seeker's profile. • Thus, the job applicant is provided withnew recommendations. Similarly, the same recommendation system helps provide job

		applicant recommendations to the job recruiters to find the most eligible candidates for theirfirm. Training programmes and certification

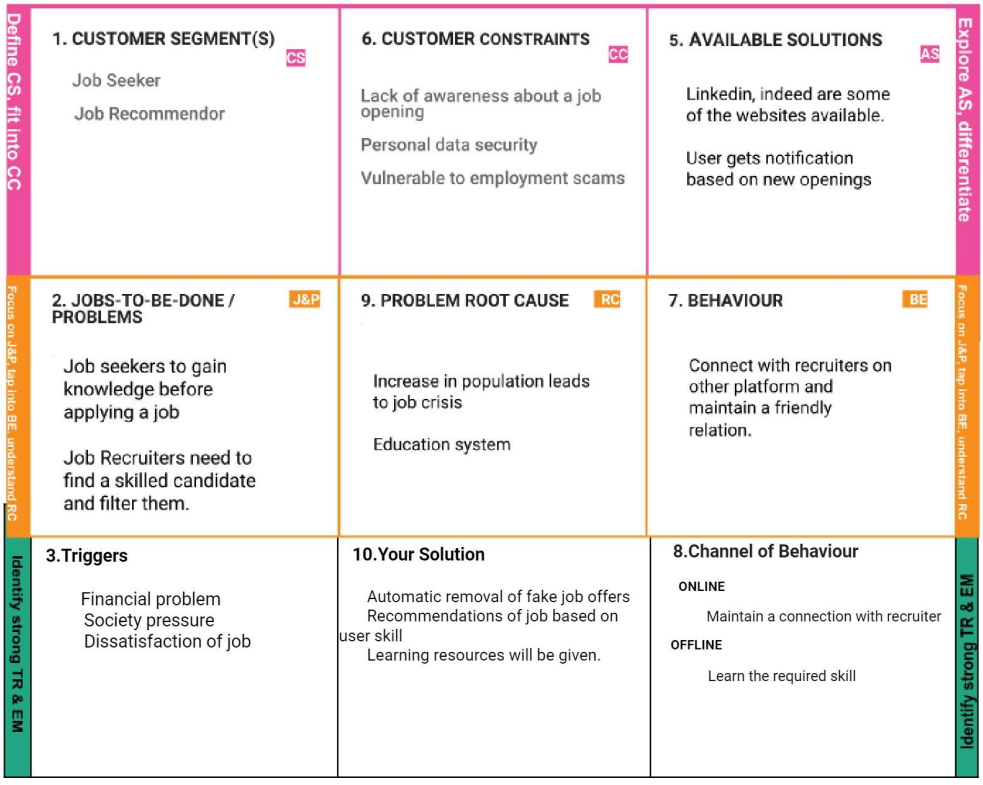
		coursesare also recommended to job seekers based on their job interests to grow their skills.
3.	Novelty / Uniqueness	• A fake job detection ML model whichverifies the job postings and removes the fraudulent ones before getting listed on the platform is integrated with the recommendation engine to bring down the employment scams. • This will preventthe job seekerfrom getting trapped with fraud one.
4.	Social Impact / CustomerSatisfaction	• The job &skill recommender systemwill minimize the unemployment and improve the skillsof job seekersto boost thecountry's economy. • The customer satisfaction can be measured by customer loyalty and customer

		reviews after deployment of the project.
5.	Business Model (Revenue Model)	• A subscription model will be provided for both employees and employers with additional costsfor features alongwith recurring monthly or yearly costs.

6.	Scalability of the Solution	• In order to provide the best scalability, cloud computing is utilised. • The cloud is capable of increasing or decreasing IT resources as needed to meet the changing demand and workload.

3.4Problem Solution fit

Project Title: Skill/Job Recommander | **Project Design Phase-I - Solution Fit Template** | **Team ID:** PNT2022TMID50561



4. REQUIREMENT ANALYSIS

4.1Functional Requirement

System requirements for candidates/job recommendation

There are major requirements presented in literatures that should be derived when recommending candidates for a specific job (Malinowski et al., 2006, 2008; Keim, 2007).

1. The matching of individuals to job depends on skills and abilities that individuals should have.

2. Recommending people is a bidirectional process thatneeds to take into account thepreferences not only of the recruiter but also of the candidate.

3. Recommendations should be based on the candidate attributes, as well as the relationalaspects that determine the fit between the person and the team members with whom the person will be collaborated.

4. Individual is considered to be unique; we cannot choose a single person several times suchas a movie or book.

4.2Non-Functional requirements

Job recommendation problem is bidirectional recommendation between job-seeker and job. The recommendation process can be divided into two parts: job

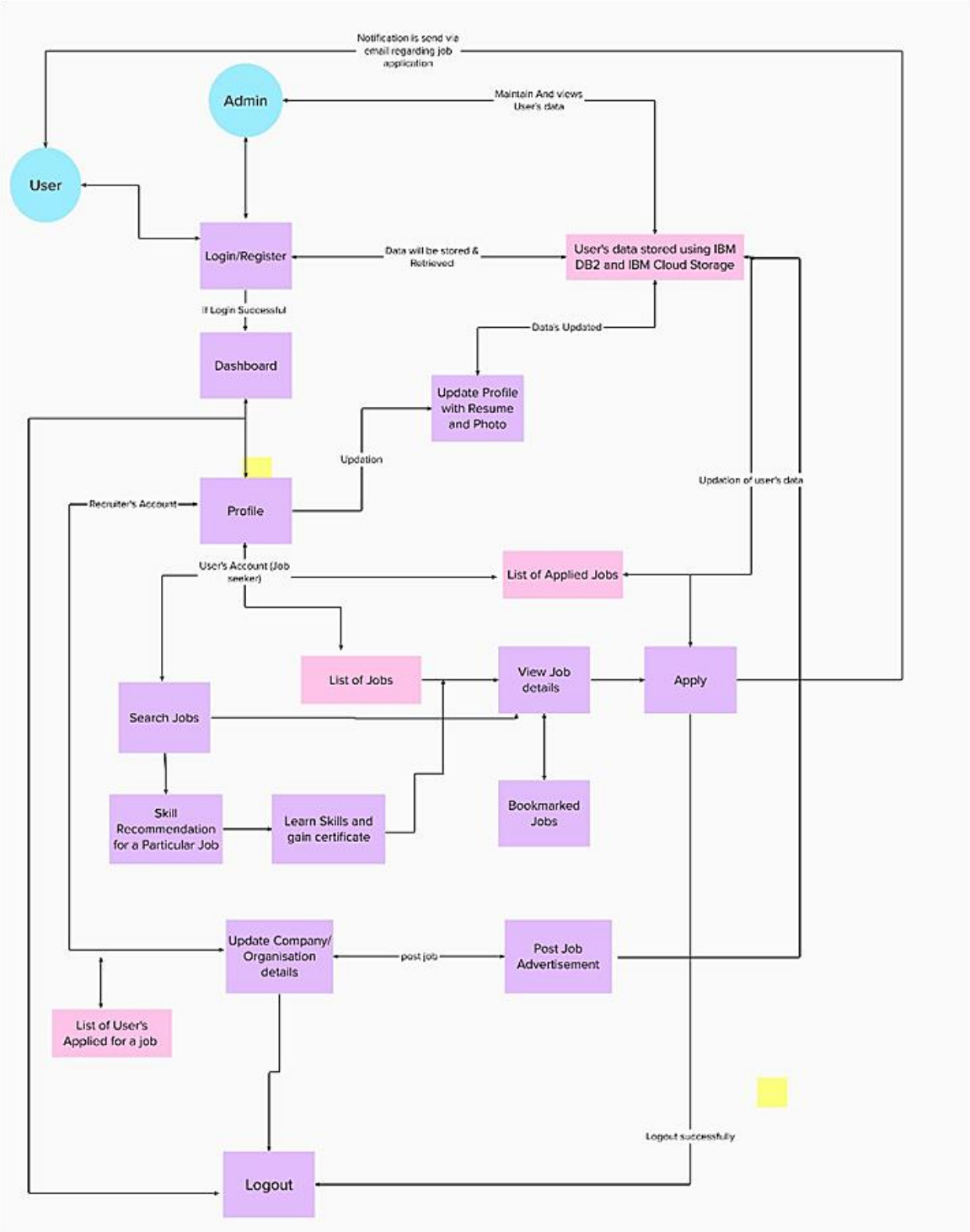
recommendation and job- seeker recommendation. The design idea of these two parts is the same roughly (Yu et al.,2011,Malinowski et al., 2006). For a job-seeker, the job with higher matching degree should be recommended to him.

Similarly, for a job, the job-seeker with higher matching degree should be recommended to it (Yu et al., 2011). In general, the ranking items either are the top n candidates that best fit the job inconsideration or the top n job profiles that best fit the candidates" preferences.

Additionally, Fazel-Zarandi and Fox (2010) mentioned that skills requirements matching need todistinguish between must-have and nice-to-have requirements in the matching process. Must- have requirements are constraints that should be possessed by the applicant, whereas nice-to have requirements are preferences that are taken into consideration when ranking applicants.

5. Project Design

5.1Data Flow Diagram



5.2Solution and Technical Architecture

Functional Requirements:

Following are the functional requirements of the proposedsolution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form And through Gmail
FR-2	User Confirmation	Confirmation via Email That is via OTP
FR-3	Chat Bot	A Chat Bot will be there in website to solve user queriesand problems related to applying a job, searchfor a job and muchmore.
FR-4	User Login	Login through Form Login throughGmail
FR-5	User Search	Exploration of Jobsbased on job filters and skill recommendations.
FR-6	User Profile	Updation of the user profile through the login credentials
FR-7	User Acceptance	Confirmation of the Job.

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	This application can be used by thejob seekersto login andsearch for the job based on her Skills set.
NFR-2	**Security**	This application is secure with separate loginfor Job Seekers as well as Job Recruiters.

NFR-3	**Reliability**	This application is open-source and feel free to use,without need to pay anything. The enormous job openings will be provided to all the job seekers without any limitation.
NFR-4	**Performance**	The performance of this application is quicker response and takes lessertime to do any process.
NFR-5	**Availability**	This application provides job offers and recommends Skills for a Particular Job openings.
NFR-6	**Scalability**	The Response timeof the application is quite faster compared to any otherapplication.

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	The user can interacts with our application with the helpof chatbot, etc.	HTML, CSS, JavaScript / Angular Js /React Js etc.
2.	Application Logic-1	The User can login with application, by previously he should register in our webapp.	Javascript
3.	Application Logic-2	They can also register with the helpof chatbot.	IBM Watson Assistant
4.	Cloud Database	The user data will be stored and retrieved with thehelp of this database.	IBM DB2, IBM Cloudant etc.
5.	File Storage	The user documents like photos, resumes andmuch more willbe stored in cloud bucket, etc.,	IBM Block Storage or Other StorageService or LocalFilesystem

6.	External API	With the help of API, the user can search the job basedon their Skillset.	IBM API, etc.
7.	Infrastructure (Server / Cloud)	Application Deployment on Local System/ Cloud	Local, CloudFoundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description
1.	Is it Scalable?	It follows highly scalable technologies that allows application to handle increase in large userdata's, workload and perform any operation without any problem.
2.	Is it Modifyable?	It is highlyModifyable and Maintenance requires low cost, compared to other application.
3.	Is the System Robust?	It does not disturb the performance of the computer by not affecting the operating system. It works in minimal hardware systems.

5.3User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story/ Task	Acceptance criteria		
Customer (Mobile user)	Registration	USN-1	As a user,I can register for the application by entering my email, password, and confirming my password.	I can accessmy account / dashboard		
		USN-2	As a user, I will receive confirmation emailonceI have registered for the application	I can receive confirmation email & clickconfirm		
		USN-3	As a user,I can register for the applicationthrough online websites	I can register &access the dashboard with online websiteLogin		
		USN-4	As a user,I can register for the applicationthrough Gmail	I can receive confirmation Gmail & clickconfirm		

	Login	USN-5	As a user, I can log into the application by entering email& password	I can receive confirmation email & clickconfirm		
	Dashboard					
Customer (Webuser)		USN-6	As a user, I can able to take up the skill assessment and view the appropriate test score. Based on the skillsets I canable to getpersonalised job recommendations.	I can receive jobrecommendatio ns		
Customer Care Executive		USN-7	As a customer care executive, we provide 24/7 chatbot support.	24/7 chatbot support		
Administrat or		USN-8	As an administrator, I can able to view the progress and make required changesin the project	Deploy user specific and personalised job recommendations		

6. Project Planning And Scheduling

6.1Sprint Planning ,Estimation and Delivery Schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story/ Task	Priority	Acceptance criteria	Team Members
Sprint-1	UIDesign	USN-1	Asa user, I can see and experiencean awesome user interface in the website	Medium	Better Impression about a website	Yuvashree K

Sprint-1	Registration	USN-2	As a user, I can register for the application by entering my email, password, and confirming my password.	High	I can access my account / dashboard	Yuvashree K
Sprint-1		USN-3	As a user, I willreceive confirmation email once I have registered for the application	High	I can receive confirmation email& click confirm	Sherbia S Sriparadeepa K
Sprint-1		USN-4	As a user, I can register for the applicati on through Facebook	Low	I can register & access the dashboard with Facebook Login	Vairabharani S Yuvashree K
Sprint-1		USN-5	As a user, I can register for the applicati on through Gmail	Medium	I can receive confirmation email& click confirm	Sherbia S Yuvashree K
Sprint-1	Login	USN-6	As a user,I can log into the applicationby entering email& password	High	I can access my account / dashboard	Vairabharani S
Sprint-!	Flask	USN-7	As a user, Ican access the website in asecond	High	I can access my account / dashboard	Sherbia S

Sprint	Functional Requirement (Epic)	User StoryNumber	User Story / Task	Priority	Acceptance criteria	Team Members
Sprint-1	Dashboard	USN-8	As a user,If I Logged in correctly, I can view my dashboard and I can navigateto any pages which are already listed there.	High	I can accessall the pages/dashboard	Yuvashree K Sherbia S
			Submission Of Sprint-1			
Sprint-2	User Profile	USN-9	Asa user, I can view and updatemy details	Medium	I can modify my details/data	Sripradeepa S
Sprint-2	Database	USN-10	Asa user, I can store my detailsanddata in the websitew	Medium	I can store my data	Sherbia S Sripradeepa S
Sprint-2	Cloud Storage	USN-11	As a user, I can upload my photo,resume and much more in the website.	Medium	I can Uploadmy documents anddetails	Vairabharani S
Sprint-2	Chatbot	USN-12	Asa user, I can ask the Chatbotabout latest job openings, whichwillhelp me and show the recentjob openings based on my profile	High	I can knowthe recent jobopenings	Vairabharani S Yuvashree K

Sprint-2	Identity-Aware	USN-13	Asa User, I can accessmy accountby entering by correct logincredentials. My user credentials is only displayed to me.	High	I can have my account safely	Yuvashree K
			Submission of Sprint-2			

Sprint	**Functional Requirement (Epic)**	**User StoryNumber**	**User Story / Task**	**Priority**	**Acceptance criteria**	**Team Members**
Sprint-3	Sendgrid service	USN-14	Asa user, I can get a notification or mail abouta job opening with the helpof sendgrid service.	Medium	I can get a notification in a second.	Vairabhrani S Sherbia S
Sprint-3	Learning Resource	USN-15	Asa user, I can learn the courseandI will attain the skills which will be useful for developing my technical skills.	High	I can gainthe knowledge andskills	Sripradeepa K Sherbiya S
Sprint-3	Docker	USN-16	Asa user, I can accessthe website inany device	High	I can accessmy account in anydevice	Sherbia S
Sprint-3	Kubernates	USN-17	As a user, I can access the website in any device	High	I can accessmy account in anydevice	Sripradeepa K

Sprint-3	Deployment in cloud	USN-18	As a user, I can access the website in any device	High	I can accessmy account in anydevice	Sherbia S
Sprint-3	Technical support	USN-19	As a user, I can get a customer caresupport from the website whichwill solve my queries.	Medium	I can tackle my problem &queries.	Yuvashree K Sherbia S
			Submission of Sprint-3			
Sprint-4	Unit Testing	USN-15	As a user, I can access the website withoutany interruption	High	I can access thewebsite withoutany interruption	Yuvashree K Sripradeepa K
Sprint-4	Integration testing	USN-16	As a user, I can access the website withoutany interruption	High	I can accessthe website without any interruption	Sherbia S Sripradeepa K

Sprint	**Functional Requirement (Epic)**	**User StoryNumber**	**User Story / Task**	**Priority**	**Acceptance criteria**	**Team Members**
Sprint-4	System testing	USN-17	As a user, I can access the website withoutany interruption	High	I can accessthe website without any interruption	Vairabharani S Sripradeepa K Sherbia S Yuvashree K

Sprint-4	Correction	USN-18	As a user, I can access the website withoutany interruption	High	I can accessthe website without any interruption	Yuvashree K
Sprint-4	Acceptancetesting	USN-19	As a user, I can access the website withoutany interruption	High	I can accessthe website without any interruption	Yuvashree K Sherbia S
			Submission of Sprint-4			

7. Coding And Solutioning

- Style.css
- apply.html
- dashboard.html
- display.html
- home.html
- login.html
- register.html

8. Testing

1. The app should allow the user to login and the login credentials should be stored in the database using back-end.
2. User should apply to respective jobs and the details must be

maintained in the data base.

3. Chat-box must be interactive and used to clarify the users dought.
4. Application should be user-friendly.

9. RESULTS

9.1 Performance Metrics

We present extensive empirical experiments focused on evaluating the quality of job recommendations. For these experiments, we take the case of recommending a set of job offers given a specific professionalprofile. Our data set is composed by 50 professional profiles from LinkedIn and 3877 job offers from Catho. Both profiles and job offers correspond to Brazilian professionals and companies from the IT field. Due to the extensive of the IT field, professional profiles can also differ a little bit among them. The distribution of subfields within our sample of 50 professional profiles which reflects the greater number of developers and BI consultants

First, we use our framework to generate 10 job offer recommendations for 50 different profiles.Thus, for each evaluated technique, we obtained a total of 500 recommendations. Second, a group of 5 Resource Human professionals evaluated manually these recommendations and allocate a score ranging from 1 to 10. The more accurate or suitable the recommendation, the greater the score. In order to make the results more understandable, we standardize these scores dividing them by the maximum score. Third, once these scores are obtained, we averaged them and also calculated Precision and Minimum Effectiveness (ME). Precision for a single profile by dividing the number of relevant documents (recommendations with a score greater than 0.5) by all the retrieved documents (total of recommendations); then, we average this precision over all the profiles. On the other hand, the Minimum Effectiveness (ME) allocates a score of 1 if at least one out of the 10 recommendations for a profile has a score greater or equal to 0.5, otherwise it allocates 0. Thus, we average this value to have an estimator of the global effectiveness of the system of 10 job recommendations per profile.

10.ADVANTAGES & DISADVANTAGES

Advantages

The main aim of any recommendation engine is to stimulate demand and actively engage users. Primarily a component of an eCommerce personalization strategy, recommendation engines dynamically populate various products onto websites, apps, or emails, thus enhancing thecustomer experience.

Dis-advantages

- Significant investments required.
- Too many choices.
- The complex onboarding process.
- Lack of data analytics capability.
- The 'cold start' problem.
- Inability to capture changes in user behavior.
- Privacy concerns.

11.CONCLUSION

We used a literature analysis of many journals and proceedings related to the recruiting process and the job recommendation researches. We have seen from our literature review and from the challenges that faced the holistic e-recruiting platforms, an increased need for enhancing the quality of candidates/job matching. The recommender system technologies accomplished significant success in a broad range of applications and potentially a powerful searching and recommending techniques. Consequently, there is a great opportunity for applying these technologies in recruitment environment to improve the matching quality. This survey shows that several approaches for job recommendation have been proposed, and many techniques combined in order to produce the best fit between jobs and candidates. We presented state of the art of job recommendation as well as, a comparative study for its approaches that proposed by literatures. Additionally, we reviewed typical recommender system techniques and the recruiting process related issues. We conclude that the field of job recommendations is still unripe and require further improvements. As part of our ongoing research, we aim to build a new recommendation approach and test with real data for employee and staffi ng data from large companies.

12.FUTURE SCOPE

In the last years, job recommender systems have become popular since they successfully reduce information overload by generating personal-ized job suggestions. Although in the literature exists a variety of techniques and strategies used as part of job recommender systems, most of them fail to recommending job vacancies that fit properly to the job seekers profiles. Thus, the contributions of this work are threefold, we: i) made publicly available a new

dataset formed by a set of job seekers profiles and a set of job vacancies collected from different job search engine sites; ii) put forward the proposal of a framework for job recommendation based on professional skills of job seekers; and iii) carried out an evaluation to quantify empirically the recommendation abilities of two state-of-the-art methods, considering different configurations, within the proposed framework. We thus present a general panorama of job recommendation task aiming to facilitate research and real-world application design regarding this important issue.

13.APPENDIX

13.1 Source Code

apply.html

```
<!DOCTYPE
html>
          <html lang="en">
          <head>
          <meta charset="UTF-8">
          <meta http-equiv="X-UA-Compatible" content="IE=edge">
          <meta name="viewport" content="width=device-width, initial-scale=1.0">
          <title>JOBPORTAL | APPLY</title>
          <!-- favicon -->
          <!-- <link rel="shortcut icon" href="/assets/img/favicon.ico" type="image/x-
          icon"> -->
          <!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
          <link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-
          32x32.png">
          <!-- bootstrap css cdn -->
          <link rel="stylesheet"
          href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
          integrity="sha384-
          JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
          crossorigin="anonymous">
          <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
```

```
a
w
e
s
o
m
e
/
4
.
7
.
0
/
c
s
s
/
f
o
n
t
-
a
w
e
s
o
m
e
.
c
s
s
"
>

<!-- css
styleshee
t -->

<link
rel="styl
esheet"
href="css
/style.cs
s">

<!-- font
styles
cdn -->
```

```
<link rel="preconnect" href="https://fonts.gstatic.com">
<link href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap" rel="stylesheet">
<link href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swap" rel="stylesheet">
</head>
<body>
<!-- bootstrap navbar -->
<div class="logo mt-3 text-center">
<a class="main-logo-img mt-5" href=".png"><img src="" alt="" height="50px" width="180px">
<!-- <a class="navbar-brand" href="index.html">JobPortal</a> -->
</a>
</div>
<!-- navbar ends -->
<!-- Login form -->
<div class="login text-center mt-5">
<h2>Apply Now</h2>
<div class="msg">{{ msg }}</div>
<form action="/apply" method="post" class="mt-3">
<!-- <input type="text" placeholder="fullname" id="fullname"> </br></br> -->
<input type="text" name="username" placeholder="Enter Your Username" id="username" required></br></br>
<input type="email" name="email" placeholder="Enter Your email" id="email" required></br></br>
<input type="text" name="qualification" placeholder="Enter Your Qualification" id="qualification" required></br></br>
<input type="text" name="skills" placeholder="Enter Your skills" id="skills" required></br></br>


<select name ="s">
<option value ="PYTHON"> Python</option>
<option value ="ML"> ML</option>
<option value ="AI"> AI</option>
</select>
</br></br>
<button type="submit" id="button" class="btn btn-primary"> Submit</button>
</form>
</div>
```

```
<div class="note mt-3 text-center">
<p> click here to go to dashboard <a href="dashboard">Dashboard! </a> </p>


</div>

</body>
</html>
```

dashboard.html

```
<!DOCTYPE
html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>JOBPORTAL | HOME</title>
<meta charset="UTF-8">
<!-- favicon -->
<!-- <link rel="shortcut icon" href="/assets/img/favicon.ico" type="image/x-
icon"> -->
<!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
<link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-
32x32.png">
<!-- bootstrap css cdn -->
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">
<!-- css stylesheet -->
<link rel="stylesheet" href="css/style.css">
<!-- font styles cdn -->
<link rel="preconnect" href="https://fonts.gstatic.com">
<link href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap"
rel="stylesheet">
```

```
<link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swap"
rel="stylesheet">
</head>
<body>
<!-- bootstrap navbar -->
<nav class="navbar sticky-top navbar-expand-lg navbar-light">
<div class="container-fluid">
<a class="main-logo-img mt-3" href="#"><img src="static/img/smartinternz.png"
alt="sheep-logo" height="50px" width="180px">
<!-- <a class="navbar-brand" href="index.html">JobPortal</a> -->
</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>


<div class="row donate-sponsor">
<a type="button" class="btn btn-success mr-1" id="donate"
href="/logout">LOGOUT</a>
<a type="button" class="btn btn-warning mr-1" id="sponsor"
href="register">REGISTER</a>
<a type="button" class="btn btn-primary mr-1" id="sponsor" href="display">MY
JOBS</a>
</div>
</div>
</nav>
<!-- navbar ends -->


<!-- what we focus on -->
<section class="our-focus">
<div class="container">
<h2 class="text-center mt-3">Available Jobs</h2>
<div class="row ml-3 mt-3">
<div class="col-lg-3 mr-5" id="focus-first">
<div class="card" style="width: 19rem;">
<!-- <img src="assets/img/home kids.jpg" class="card-img-top" alt="..."> -->
<div class="card-body">
<h5 class="card-title">Python</h5>
```

```
<img src="https://s3.jp-tok.cloud-object-storage.appdomain.cloud/bu-123/chris-ried-ieic5Tq8YMk-unsplash.jpg", width="200px", height="120px" />
<p class="card-text">Skills for python
</p>
<a href="apply" class="btn btn-primary">Apply Now</a>
</div>
</div>
</div>
<div class="col-lg-3 mr-5" id="focus-second">
<div class="card" style="width: 20rem;">
<!-- <img src="assets/img/friendship day.JPG" class="card-img-top" alt="..."> -->
<div class="card-body">
<h5 class="card-title">Data Scientist</h5>
<img src="https://s3.jp-tok.cloud-object-storage.appdomain.cloud/bu-123/stephen-dawson-qwtCeJ5cLYs-unsplash.jpg", width="200px", height="120px"/>
<p class="card-text">Skills for datascientist</p>
<a href="apply" class="btn btn-primary">Apply Now</a>
</div>
</div>
</div>
<div class="col-lg-3 ml-5" id="focus-third">
<div class="card" style="width: 20rem;">
<!-- <img src="assets/img/health camp.jpg" class="card-img-top" alt="..."> -->
<div class="card-body">
<h5 class="card-title">HR Manager</h5>
<img src="https://s3.jp-tok.cloud-object-storage.appdomain.cloud/bu-123/campaign-creators-gMsnXqILjp4-unsplash.jpg", width="200px", height="120px"/>
<p class="card-text">Skills for hr manager</p>
<a href="apply" class="btn btn-primary">Apply Now</a>
</div>
</div>
</div>


</div>
</div>
</section>
<!-- focus section ends -->
```

```
<!-- footer starts -->
<!-- Site footer -->

</body>
</html>
```

display.html

```
<!DOCTYPE
html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>JOBPORTAL | HOME</title>
<meta charset="UTF-8">
<!-- favicon -->
<!-- <link rel="shortcut icon" href="/assets/img/favicon.ico" type="image/x-
icon"> -->
<!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
<link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-
32x32.png">
<!-- bootstrap css cdn -->
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">
<!-- css stylesheet -->
<link rel="stylesheet" href="css/style.css">
<!-- font styles cdn -->
<link rel="preconnect" href="https://fonts.gstatic.com">
<link href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap"
rel="stylesheet">
<link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swap"
rel="stylesheet">
</head>
<body>
```

```
<!-- bootstrap navbar -->
<nav class="navbar sticky-top navbar-expand-lg navbar-light">
<div class="container-fluid">
<a class="main-logo-img mt-3" href="#"><img src="static/img/smartinternz.png"
alt="sheep-logo" height="50px" width="180px">
<!-- <a class="navbar-brand" href="index.html">JobPortal</a> -->
</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>

<div class="row donate-sponsor">
<a type="button" class="btn btn-success mr-1" id="donate"
href="/logout">LOGOUT</a>
<a type="button" class="btn btn-warning mr-1" id="sponsor"
href="register">REGISTER</a>
<a type="button" class="btn btn-primary mr-1" id="sponsor" href="display">MY
JOBS</a>
</div>
</div>
</nav>
<!-- navbar ends -->

<!-- what we focus on -->
<section class="our-focus">
<div class="container">
<h2 class="text-center mt-3">Available Jobs</h2>
<div class="row ml-3 mt-3">
<div class="col-lg-3 mr-5" id="focus-first">
<div class="card" style="width: 19rem;">
<!-- <img src="assets/img/home kids.jpg" class="card-img-top" alt="..."> -->
<div class="card-body">
<h5 class="card-title">Python</h5>
<img src="https://s3.jp-tok.cloud-object-storage.appdomain.cloud/bu-123/chris-
ried-ieic5Tq8YMk-unsplash.jpg", width="200px", height="120px" />
<p class="card-text">Skills for python
</p>
<a href="apply" class="btn btn-primary">Apply Now</a>
```

```
</div>
</div>
</div>
<div class="col-lg-3 mr-5" id="focus-second">
<div class="card" style="width: 20rem;">
<!-- <img src="assets/img/friendship day.JPG" class="card-img-top" alt="..."> -
->
<div class="card-body">
<h5 class="card-title">Data Scientist</h5>
<img src="https://s3.jp-tok.cloud-object-storage.appdomain.cloud/bu-
123/stephen-dawson-qwtCeJ5cLYs-unsplash.jpg", width="200px", height="120px"/>
<p class="card-text">Skills for datascientist</p>
<a href="apply" class="btn btn-primary">Apply Now</a>
</div>
</div>
</div>
<div class="col-lg-3 ml-5" id="focus-third">
<div class="card" style="width: 20rem;">
<!-- <img src="assets/img/health camp.jpg" class="card-img-top" alt="..."> -->
<div class="card-body">
<h5 class="card-title">HR Manager</h5>
<img src="https://s3.jp-tok.cloud-object-storage.appdomain.cloud/bu-
123/campaign-creators-gMsnXqILjp4-unsplash.jpg", width="200px",
height="120px"/>
<p class="card-text">Skills for hr manager</p>
<a href="apply" class="btn btn-primary">Apply Now</a>
</div>
</div>
</div>

</div>
</div>
</section>
<!-- focus section ends -->

<!-- footer starts -->
<!-- Site footer -->

</body>
</html>
```

register.html

```
<!DOCTYPE
html>
          <html lang="en">
          <head>
          <meta charset="UTF-8">
          <meta http-equiv="X-UA-Compatible" content="IE=edge">
          <meta name="viewport" content="width=device-width, initial-scale=1.0"
          <title>JOBPORTAL | LOGIN</title>
          <!-- favicon -->
          <!-- <link rel="shortcut icon" href="/assets/img/favicon.ico" type="i
          icon"> -->
          <!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-ic
          <link rel="icon" type="image/png" sizes="16x16" href="/assets/img/fav
          32x32.png">
          <!-- bootstrap css cdn -->
          <link rel="stylesheet"
          href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstra
          integrity="sha384-
          JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
          crossorigin="anonymous">
          <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/f
          awesome/4.7.0/css/font-awesome.css">
          <!-- css stylesheet -->
          <link rel="stylesheet" href="css/style.css">
          <!-- font styles cdn -->
          <link rel="preconnect" href="https://fonts.gstatic.com">
          <link href="https://fonts.googleapis.com/css2?family=Alegreya&display
          rel="stylesheet">
          <link
          href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&disp
          rel="stylesheet">
          </head>
          <body>
          <!-- bootstrap navbar -->
          <div class="logo mt-3 text-center">
          <a class="main-logo-img mt-5" href="#"><img src="static/img/smartinte
          alt="sheep-logo" height="50px" width="180px">
          <!-- <a class="navbar-brand" href="index.html">JobPortal</a> -->
          </a>
```

```
</div>
<!-- navbar ends -->
<!-- Login form -->
<div class="login text-center mt-5">
<h2> Register Form </h2>
<form action="/register" method="post">
<div class="msg">{{ msg }}</div>
<!-- <input type="text" placeholder="fullname" id="fullname"> </br></

<input type="text" name="username" placeholder="Enter Your Username"
id="username" required></br></br>
<input type="email" name="email" placeholder="Enter Your Email ID" id
required></br></br>
<input type="password" name="password" placeholder="Enter Your Passwo
id="password" required></br></br>


</br>
</br>
<button type="submit" id="button" class="btn btn-primary" > Register
</form>
</div>

<div class="note mt-3 text-center"> <!--Register form -->
<p> already have an account ? please login <a href="/login">login! </


</div>


</body>
</html>
```

style.css

```
* {
    font-family: 'Alegreya', serif !important;

    }
```

```
/* // Small devices (landscape phones, 576px and up) */
@media (min-width: 576px) {

}

/* // Medium devices (tablets, 768px and up) */
@media (min-width: 768px) {

}

/* // Large devices (desktops, 992px and up) */
@media (min-width: 992px) {
.navbar {
padding-top:15px;
padding-bottom:15px;
background-color: white;
}
.navbar-
brand{ padding-
left: 5px;
}
.navbar-nav
{ margin-left:
30px;
}
.nav-item
{ padding-
left:5px;
}
.donate-
sponsor{ margin-
right:10px;
}
#donate,#sponsor{
margin:5px;
padding: 5px 15px 5px 15px;
}
.homepage-header{
background: url('../img/group.JPG');
background-size: cover;
```

```
background-position: center top;
padding: 0;
position: relative;
width: 100%;
overflow: hidden;
display: -webkit-flex;
display: -ms-flexbox;
display: flex;
height: 85vh;
}
.home-for-children
{ background-color:
#ffeeba;margin-top:0px;
padding-top:10px;
padding-bottom:30px;
}
.home-for-children h2 {
line-height: 2.5rem !important;
letter-spacing: 3px;
font-weight: 600;
}
.home-for-children h5
{line-height: 1.8rem;
}
.home-for-children .btn-success
{padding: 8px 25px;
font-size: large;
}
/* our focus */
.our-focus .container
{ margin-
top:3rem !important;
}

.our-focus .row{
margin-top:1.5rem !important;
}
.our-focus .card {
```

```
border: none !important;
}
.our-focus #focus-first
{margin-right:80px;
}
.our-focus #focus-second
{margin-right:90px;
}
/* media */
.media .container{
margin-top: 3rem !important;
}
.media .row{
margin-top: 1.5rem !important;
}
/* footer */
.row-initiative
{margin-top:10px
}
.site-footer
{ margin-top:30px;
line-height:24px;
}
.footer-links
{ padding-left:0;
list-style: none;
}
.footer-links li
{display: block;
}
.footer-links.inline li
{display: inline-block;
}
.footer-links li
a{color: black;
}
.footer-logo img{
```

```
width: 100px;
}
.social-icons
{ text-align:
right;margin-
left:50px;
}
.social-icons
li{list-style:
none;
display: inline-block;
}
.social-icons li a
{border-radius: 50%;
margin-left:10px;
}
input
{ padding:10px
20px;
}
#button
{ border:none;
padding:10px 20px;
border-radius:10px;
animation:pulse 3s infinite ease-out;
}
}

/* // X-Large devices (large desktops, 1200px and up)
*/
@media (min-width: 1200px) {
.navbar-
brand{ padding-left:
10px;
}
}

/* // XX-Large devices (larger desktops, 1400px and up)
*/
@media (min-width: 1400px) {
.navbar-
brand{ padding-left:
```

```
15px;
```

```
        }
        }
```

app.py

```
from
collectio
ns import
UserDict
        from flask import Flask,render_template,request,redirect,url_for,session
        import ibm_db
        import re

        app =Flask(__name__)
        app.secret_key='a'

        conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-
        9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;SECURITY=SSL;SSLSer
        Mcd7x6",'','')
        @app.route('/')
        def home():
        return render_template('home.html')

        @app.route('/login',methods =['GET','POST'])
        def login():
        global userid
        msg=''

        if request.method=='POST':
        username =request.form['username']
        password=request.form['password']
        sql="SELECT * FROM users WHERE username =? AND password=?"
        stmt =ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)
```

```
print(account)
if account:
session['loggedin']=account['USERNAME']
session['id']=account['USERNAME']
userid=account['USERNAME']
session['username']=account['USERNAME']
msg= 'logged in succesfully !!!'

return render_template('dashboard.html',msg=msg)

else:
msg='incorrect'

return render_template('login.html',msg=msg)
@app.route('/register',methods=['GET','POST'])
def register():
msg=''
if request.method =='POST':
username =request.form['username']
email=request.form['email']
password=request.form['password']
sql="SELECT * FROM users WHERE username =?"
stmt=ibm_db.prepare(conn,sql)
ibm_db.bind_param(stmt,1,username)
ibm_db.execute(stmt)
account=ibm_db.fetch_assoc(stmt)
print(account)
if account:
msg='Account already exist'
elif not re.match(r'[^@]+@[^@]+\.[^@]+',email):
msg='invalid email address !'
elif not re.match(r'[A-Za-z0-9]+',username):
msg='Name must contain char and number'
else:
insert_sql="INSERT INTO users VALUES (?,?,?)"
prep_stmt=ibm_db.prepare(conn,insert_sql)
ibm_db.bind_param(prep_stmt,1,username)
```

```
ibm_db.bind_param(prep_stmt,2,email)
ibm_db.bind_param(prep_stmt,3,password)
ibm_db.execute(prep_stmt)
msg='you have successfully registered '
elif request.method =='POST':
msg="Please fill out the form"
return render_template('register.html',msg=msg)

@app.route('/dashboard')
def dash():
return render_template('dashboard.html')

@app.route('/apply',methods=['GET','POST'])
def apply():
msg=''
if request.method =='POST':
username =request.form['username']
email=request.form['email']
qualification=request.form['qualification']
skills =request.form['skills']
jobs=request.form['s']
sql='SELECT * FROM users WHERE username =?'
insert_sql="INSERT INTO job VALUES (?,?,?,?,?)"
prep_stmt=ibm_db.prepare(conn,insert_sql)
ibm_db.bind_param(prep_stmt,1,username)
ibm_db.bind_param(prep_stmt,2,email)
ibm_db.bind_param(prep_stmt,3,qualification)
ibm_db.bind_param(prep_stmt,4,skills)
ibm_db.bind_param(prep_stmt,5,jobs)
ibm_db.execute(prep_stmt)
msg='you have successfully applied for job '
session['loggedin']=True
elif request.method =='POST':
msg="Please fill out the form"
return render_template('apply.html',msg=msg)

@app.route('/display')
```

```
def display():
print(session["username"],session["id"])
cursor=mysql.connection.cursor()
cursor.execute('SELECT * FROM job WHERE userid = %s',(session['id']))
account=cursor.fetchone()
print("accountdisplay",account)

return render_template('display.html',account=account)

@app.route('/logout')
def logout():
session.pop('loggedin',None)
session.pop('id',None)
session.pop('username',None)

return render_template('home.html')

if__name__=='__main__':
app.run(host='0.0.0.0')
```

deployment.yaml

```
apiVersion:
apps/v1
          kind: Deployment
          metadata:
          name: job
          spec:
          replicas: 3
          selector:
          matchLabels:
          app: job
          template:
          metadata:
          labels:
          app: job
          spec:
```

```
        containers:
        - name: job
        image: karolinpreethy/jobportal11
        imagePullPolicy: Always
        ports:
        - containerPort: 5000
```

service.yaml

```
ap
iV
er
si
o
n:
v1
   kind: Service
   metadata:
   name: job
   spec:
   ports:
   - port: 5000
   targetPort: 5000
   selector:
   app: job
```

13.2 Github Link

github.com/IBM-EPBL/IBM-Project-39436-1660437806