

IBM NALAIYA THIRAN PROJECT REPORT

SKILL / JOB RECOMMENDER

TEAM ID:PNT2022TMID47412

SUBMITTED BY

ARUN KUMAR K(TL)	910019104005
KAVIYARASAN R	910019104020
NITHESHWARAN T	910019104027
SANJAY KUMAR V	910019104037

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

ANNA UNIVERSITY REGIONAL CMAPUS MADURAI

ANNA UNIVERSITY :CHENNAI 600 025

TABLE OF CONTENT	PAGE NO
1. INTRODUCTION	4
1.1 Project Overview	4
1.2 Purpose	4
2. LITERATURE SURVEY	5
2.1 Existing problem	5
2.2 References	5
2.3 Problem Statement Definition	8
3. IDEATION & PROPOSED SOLUTION	9
3.1 Empathy Map Canvas	9
3.2 Ideation & Brainstorming	11
3.3 Proposed Solution	12
3.4 Problem Solution fit	14
4. REQUIREMENT ANALYSIS	17
4.1 Functional requirement	17
4.2 Non-Functional requirements	20
5. PROJECT DESIGN	22
5.1 Data Flow Diagrams	22
5.2 Solution & Technical Architecture	23
5.3 User Stories	26
6. PROJECT PLANNING & SCHEDULING	28
6.1 Sprint Planning & Estimation	28
6.2 Sprint Delivery Schedule	32
6.3 Reports from JIRA	32
7. CODING & SOLUTIONING	33
7.1 Feature 1	33
7.2 Feature 2	38

7.3 Database Schema	50
8. TESTING	51
8.1 Test Cases	51
8.2 User Acceptance Testing	51
9. RESULTS	52
9.1 Performance Metrics	52
10.ADVANTAGES & DISADVANTAGES	53
11.CONCLUSION	54
12.FUTURE SCOPE	54
13.APPENDIX	54
Source Code	86
GitHub & Project Demo Link	86

1.INTRODUCTION:

1.1 Project Overview:

According to the preferences of the job recommender system creator, this architecture both makes it easier to grasp how the job suggestion process works and permits the use of a variety of text processing and recommendation methods. Additionally, we help to making new employers' profiles of job seekers and open positions public. Future areas of our research will centre on conducting a more thorough review taking into account a larger number of methodologies and data as well as a thorough evaluation of the impact of each job seeker's professional skill on the recommendation they obtain for a position.

A Web application that can show current job positions based on users skill set. Abstract Job recommender systems have gained popularity in recent years because they efficiently alleviate information overload by producing personalised job suggestions.

1.2 Purpose:

To create a complete online application that can show available positions depending on the users' skill sets. The database contains both the users' data and that of their users. Based on the user's skill set, an alert is delivered when a position becomes available. The user will converse with the chatbot and receive suggestions depending on his abilities. To obtain the most recent job vacancies on the market, we may utilise a job search API, which will retrieve the information straight from the website.

2.LITERATURE SURVEY:

2.1 Existing Problem:

Recommendation engines, like other AI-based technologies, depend on data; if you don't have high-quality data or can't crunch and analyse it effectively, you won't be able to maximise the recommendation engine's potential. The danger of unintentional disclosure of personal information is increased by the information that recommenders collect. Additionally, malevolent individuals have the ability to skew or undermine the suggestions made to other users.

2.2 References:

Adomavicius G, Tuzhilin A (2005). Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Trans. Knowl. Data Eng.* 17(6):734-749. Barbieri N, Costa G, Manco G, Ortale R (2011). Modeling Item Selection and Relevance for Accurate Recommendations: A Bayesian Approach. In *Proceedings of the fifth ACM conference on Recommender systems (RecSys '11)*, Chicago, Illinois, USA, ACM pp. 21-28. Belkhirat A, Belkhir A, Bouras A (2011). A New Similarity Measure for the Profiles Management. *UKSim 13th International Conference on Modelling and Simulation*. Cambridge, United Kingdom: IEEE. Bhargava HK, Sridhar SC (1999). *Beyond Spreadsheets: Tools for Building Decision Support Systems*. *IEEE Computer* 32(3):31-39. Breaugh JA, Starke M (2000). Research on Employee Recruitment: So Many Studies, So Many Remaining Questions. *J. Manag.* 26(3):405-434. Breese JS, Heckerman D, Kadie C (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, WI,

ACM, pp. 43-52. Brusilovsky P (2001). Adaptive hypermedia. User Model. User Adapt. Interact. 11(1-2):87-110. Burke R (1999). Integrating Knowledge-Based and Collaborative-Filtering Recommender Systems. In Proceedings of the AAAI Workshop on AI in Electronic Commerce, Orlando, Florida, USA pp. 69-72. Burke R (2000). Knowledge-Based Recommender Systems. Encyclopedia of Library and Information Systems 69:32. Burke R (2002). Hybrid Recommender Systems: Survey and Experiments. User Model. User-Adapt. Interact 12(4):331-370. Burke R (2007). Hybrid Web Recommender Systems. The Adaptive Web: Methods and Strategies of Web Personalization 4321:377-407. Carroll M, Marchington M, Earnshaw J, Taylor S (1999). Recruitment in Small Firms: Processes, Methods and Problems. Employee Relations 21(3):236-250. Chen PC (2009). A Fuzzy Multiple Criteria Decision Making Model in Employee Recruitment. IJCSNS Int. J. Comput. Sci. Netw. Secur. 9(7):113-117. Das A, Datar M, Garg A, Rajaram S (2007). Google News Personalization Scalable Online Collaborative Filtering. In Proceedings of the 16th International Conference on World Wide Web, WWW '07, Banff, Alberta, CANADA, ACM, pp. 271-280. Färber F, Weitzel T, Keim T (2003). An Automated Recommendation Approach to Selection in Personnel Recruitment. In Proceedings of AMCIS. Tampa, FL, USA, AISeL. Fazel-Zarandi M, Fox MS (2010). Semantic Matchmaking for Job Recruitment: An Ontology Based Hybrid Approach. In Proceedings of the 3rd International Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web at the 8th International Semantic Web Conference. Washington D.C., USA, CEUR-WS.org. Felfernig A (2005). Koba4MS: selling complex products and services using knowledge-based recommender technologies. In Proceedings of 7th IEEE

International Conference on E-Commerce Technol. Munich, Germany, IEEE pp. 92-100. Felfernig A, Schubert M, Mandl M (2010). Recommendation and Decision Technologies For Requirements Engineering. RSSE '10. Cape Town, South Africa, ACM pp. 11-15. Ghazanfar MA, Prügell-Bennett A (2010).

Hannon J, Bennett M, Smyth B (2010). Recommending Twitter Users to Follow Using Content and Collaborative Filtering Approaches. In Proceedings of the fourth ACM conference on Recommender systems (RecSys '10), New York, NY, USA, ACM pp. 199-206. Herlocker JL, Konstan JA, Borchers A, Riedl J (1999). An Algorithmic Framework for Performing Collaborative Filtering. In Proceedings of the 22nd ACM SIGIR Conference on Research Development in Information Retrieval. Berkeley, CA, USA, ACM. Hien NT, Haddawy P (2007). A Decision Support System for Evaluating International Student Applications. 37th ASEE/IEEE Frontiers in Education Conference. Milwaukee, WI, IEEE pp. 1-4. Hofmann T, Puzicha J (1999). Latent Class Models for Collaborative Filtering. In Proceedings of the 16th International Joint Conference on Artificial Intelligence, Stockholm, Sweden pp. 688-693. Hu R, Pu P (2011). Enhancing Collaborative Filtering Systems with Personality Information. In Proceedings of RecSys'11, Chicago, Illinois, USA, ACM pp.197-204. Huang Z, Zeng D, Chen H (2007). A Comparative Study of Recommendation Algorithms in ECommerce. IEEE Intell. Syst. 22:68-78. Keim T (2007). Extending the Applicability of Recommender Systems: A Multilayer Framework for Matching Human Resources. In Proceedings of the 40th Annual Hawaii International Conference on System Sci. (HICSS'07). Hawaii, USA, IEEE. Koren Y, Sill J (2011). OrdRec: an Ordinal Model for Predicting Personalized Item Rating Distributions. In Proceedings of the fifth ACM conference on Recommender

systems (RecSys '11), New York, NY, USA, ACM pp. 117-124. Lang S, Laumer S, Maier C, Eckhardt A (2011). Drivers, Challenges and Consequences of E-Recruiting – A Literature Review. SIGMIS-CPR'11. San Antonio, Texas, USA, ACM pp. 26-35. Laumer S, Eckhardt A (2009). Help to Find the Needle in a Haystack: Integrating Recommender Systems in an IT Supported Staff Recruitment System. In proceedings of the special interest group on management information system's 47th annual conference on Computer personnel Research, Limerick, Ireland, ACM pp. 7-12. Laumer S, Eckhardt A (2010). Analyzing IT Personnel's Perception of Job-Related Factors in Good and Bad Times. In Proceedings of the 2010 Special Interest Group on Management Information System's 48th annual conference on Computer personnel research on Computer personnel research, Vancouver, BC, Canada, ACM pp. 95-99. Laumer S, Eckhardt A, Weitzel T (2010). Electronic Human Resources Management.

In An E-business Environment. J. Electron. Commer. Res. 11(4):240-250. Lee DH, Brusilovsky.

2.3 Problem Statement Definition:

A candidate should have five to ten years of expertise in infrastructure operations, application development, as well as two or more years of practical knowledge with CI/CD development models, in order to be employed in this expanding field of cloud computing. Industry credentials are not universally necessary for this employment; instead, each company has different requirements. An effective cloud automation engineer has practical knowledge of cloud platforms and tools. It is essential for the engineer's expertise and experience to be represented in her work because this is often a senior job.

3.IDEATION AND PROPOSED SOLUTION:

3.1 Empathy map canvas:

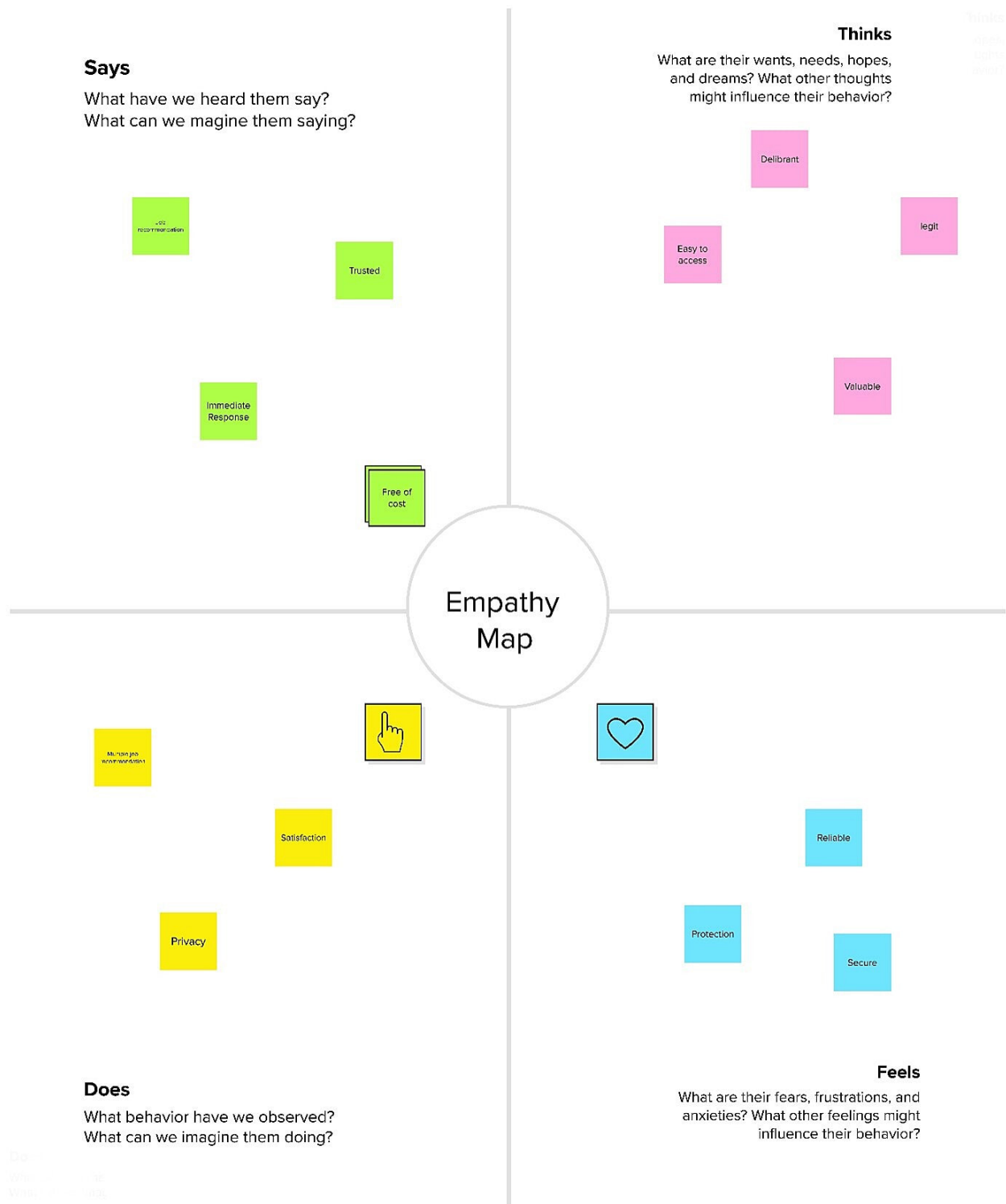
An empathy map is a straightforward, simple-to-understand picture that summarises information about a user's actions and views. It is a helpful tool that enables teams to comprehend their users more fully. It's important to comprehend both the actual issue and the individual who is experiencing it in order to develop a workable solution. Participants learn to think about situations from the user's perspective, including goals and problems, via the process of constructing the map.

3.1.1 Pain of Empathy map

- Lack of experience
- Job insecurity
- Lack of communication
- Environmental issues

3.1.2 Gain of Empathy map

- Simple to attend interview
- Reduce cost to hire
- Can know the different opportunities



3.2 IDEATION AND BRAINSTROMING

It provide recommendations based on recorded information on the users' preferences framework facilitates the understanding of job recommendation process as well as it allows the use of a variety of text processing and recommendation methods according to the preferences of the job recommender system designer. web application capable of displaying the current job openings based on the skillset of the users.

Framework for job recommendation based on professional. skills of job seekers. 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 Future directions of our work will focus on performing a more exhaustive evaluation considering a greater amount of methods and data as well as a comprehensive evaluation of the impact of each professional skill of a job seeker on the received job recommendation.

It make job recommendations by matching skills from resume to posted jobs. put forward the proposal of a framework for job recommendation based on professional skills of job seekers; Data Collection- >Data Preparation->Recommendation.

Jobs from Indeed and throws the best jobs available for you ... Personalized job recommendation sites implemented a variety of types of recommender systems, such a content based filtering, collaborative filtering, knowledge-based and hybrid approaches:

- a. finding jobs where candidates similar to a given candidate have applied .

- b. finding similar jobs to the jobs where a candidate has applied.

3.3 PROPOSED SOLUTION

S.NO	Parameter	
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none">■ The dataset used for this research are sourced from stack overflow survey data which is modelled as the user data for this research.■ Another dataset was created by web scraping . the job board using R programming language to fulfill the road map of this dissertation.■ The research question proposed by this research is “can an efficient recommended system be modelled for the job seekers which recommended jobs with the users skillset and job domain and also addresses the issues of cold start?■ To answer the research question,below are the objectives that need to be satisfied with going forward.

2.	Idea / Solution description	<ul style="list-style-type: none"> ■ Personal finance applications will ask users to add their expenses and based on their expenses, wallet balance will be updated which will be visible to the user. Also, users can get an analysis of their expenditure in graphical forms. They have an option to set a limit for the amount to be used for that particular month if the limit is exceeded the user will be notified with an email alert.
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> ■ It will be a web application ,easy and lightweight accessible from any device. ■ Expense Tracker can give you a clear picture of how much you are earning and spending and You can evaluate if you are making more expenses and where you need to cut. ■ Expense tracking will help you identify, limit, or eliminate your wasteful spending habits. It will further help you put your money where you need to.
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> ■ It will help the people to track their expenses and also alerts when you exceed the limit of your budget.

5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> ■ The Expense tracker app will solve crucial problems for our users, the best way and simplest way to make money is through Launching our app as a paid one. ■ For free or lite version of application , advertisements are the best means to earn regular income .
6.	Scalability of the Solution	<ul style="list-style-type: none"> ■ The web application will be developed in a micro services framework which is flexible and different services can use different resources. . ■ The data will be stored and the application will run on cloud services which has the ability to expand when needed.

3.4 PROBLEM SOLUTION FIT

1.Customer Segments

- Persons who are seeking employment
- Persons that recruit job candidates

2.Jobs to be done /Problems

- Create a platform to facilitate job searching

- Profile with safe personal data.
- Make the job-filtering process simpler

3. Triggers

- Financial Problem.
- Sociality Pressure.
- Dissatisfaction of job.
- Finds a better way to improve her knowledge as well as career growth.

4. Emotions

Before

- Sad, depressed and low confidence.
- Fear of Rejection before attending any hiring process.

After

- Highly Motivated.
- Gained confidence to do any task.

5.Available Solutions

Pros:

- Promotion of People's skillset.
- Marketing of company infrastructure.

Cons:

- Delivery false information.
- Intense competition.

6. Customer Constraints

- It is a Potential Scam and Time Consuming.

7. Behaviour

- When users apply for fraudulent jobs,they get unhappy due to wasted time.
- Users were not satisfied when platforms allowed hirers to psot job that were not real.
- Cheating during online recruitment process.

8.Channels of Behaviour

Online:

- Apply and maintain a connection with recuriters.

- Also search about job openings.

Offline:

- Learn and gain required skill in open source platform as well as in our job website.

9.Problem Root Cause

- Jobs that are listed on unreliable platforms may be fraudulent.
- Companies fail to disclose their true infrastructure.
- Some job portals want payment in advance of the starting.

10.Solution

- A Fake job offer is detected and removed automatically.
- Recommended and skill to job seeker for a particular job openings.
- A Notification will be sent via Email regarding job openings.

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

A functional requirement is a technical feature of software that helps systems behave and operate. Engineers typically program these features directly into a system's software. It's essential that software has functional requirements to help a

system perform tasks properly.

Six functional requirements:

Here are several common functional requirements that you might find in a software system:

1. Business requirements

A common functional requirement involves the requirements that a company needs to operate. Each company may have different business requirements that help them to function properly. For example, a grocery store may include a functional requirement that allows users to enter their phone number into a rewards system to receive discounts on products.

2. Administrative protocols

Administrative protocols allow software to perform administrative protocols, which are routine operations for a system. These protocols can include system reporting and testing to ensure the software runs smoothly. For example, a software system may run a monthly scan that identifies areas of improvement, which the system can include in a report. You may view the report to better understand the functions and quality of your system.

3. User preferences

User preference is a functional requirement that helps individuals operate software easily. It may involve specific features that enhance usability, like functions that help you navigate to certain webpages. Depending on the software,

you may set specific preferences based on your technology needs.

4. System requirements

System requirements involve specifications for software and hardware. This can include the specific actions that a system takes to complete a task. For example, if software archives data according to the date that a user saved the data, it may run through all data to find the oldest files before moving data to the system's archives. It also includes how a system responds under special circumstances. For instance, if software detects a security breach, it may deny all access to users temporarily.

5. Authentication

Authentication ensures that users validate their identity before performing certain software functions. It may involve requiring users to enter a password, username or company information. You might use this requirement in your software to protect information. Companies often use requirement to ensure that the proper individuals have access to sensitive data, like specific members of management. A business may require all employees to use authentication, or they may only require management to enter their information when accessing valuable information.

6. Legal requirements

Some industries require companies to follow regulations and meet requirements for their software systems. A company's software may have functional requirements that help systems meet these regulations and requirements. The software may perform a routine scan to ensure a company follows proper

protocols, or it may update systems regularly to stay up-to-date with new regulations.

4.2 NON FUNCTIONAL REQUIREMENT

A nonfunctional requirement is a function that helps software operate efficiently. These requirements are not mandatory for a system to have, though they typically increase a software's overall quality, speed and storage capacity. Nonfunctional requirements allow users to specific software features that enhance their usability.

Six nonfunctional requirements:

Here are some common nonfunctional requirements:

1. Usability

A common nonfunctional requirement involves specific features that help users operate software. While usability is sometimes a functional requirement, it can also be a nonfunctional requirement if a user adds extra protocols and features to further enhance usability. If software has a high level of usability, it means that users can handle a software's interface easily while understanding the different features and functions within a system.

2. Reliability

A common nonfunctional requirement involves features that analyze and increase the system's reliability. It's important for users to have reliable software so that their information is secure from potential security threats or data loss. To

determine your system's reliability, you can consider the number of failures your system has during reliability testing. A failure may include an unexpected system shut-down, a loss of information, connection issues or data that downloaded incorrectly.

3. Scalability

Scalability is a feature that companies may add when receiving large amounts of data that needs to be saved in their system. Having software that's scalable involves expanding storage space to allow for more saved information. Companies may employ a nonfunctional requirement that gives them the option of expanding their storage space beyond the software's initial capability.

4. Security

Nonfunctional security features involve adding protocols to protect valuable data. While most software involves features that authenticate users, a company may add extra security measures to minimize the risk of a potential data breach. For example, a college may add extra security protocols, like a firewall, for students accessing their financial information to ensure their data remains safe.

5. Localization

Software may include a nonfunctional requirement that involves localizing the data within software to the region of the user. The software adapts to the location so that the user can better understand the information found within the system. It may change the language of the system, the monetary currency or the time zone. For example, if a user travels to a different country, they may have a software feature on their system that changes the time zone automatically.

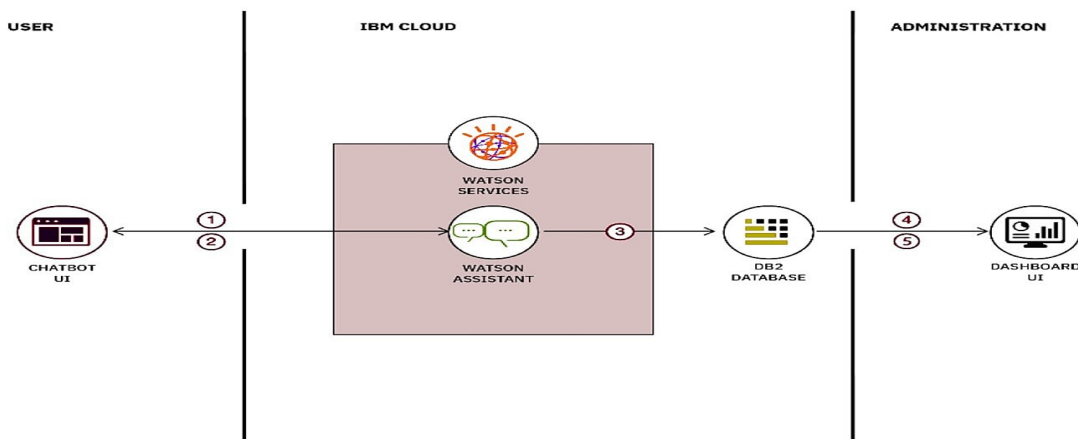
6. Performance

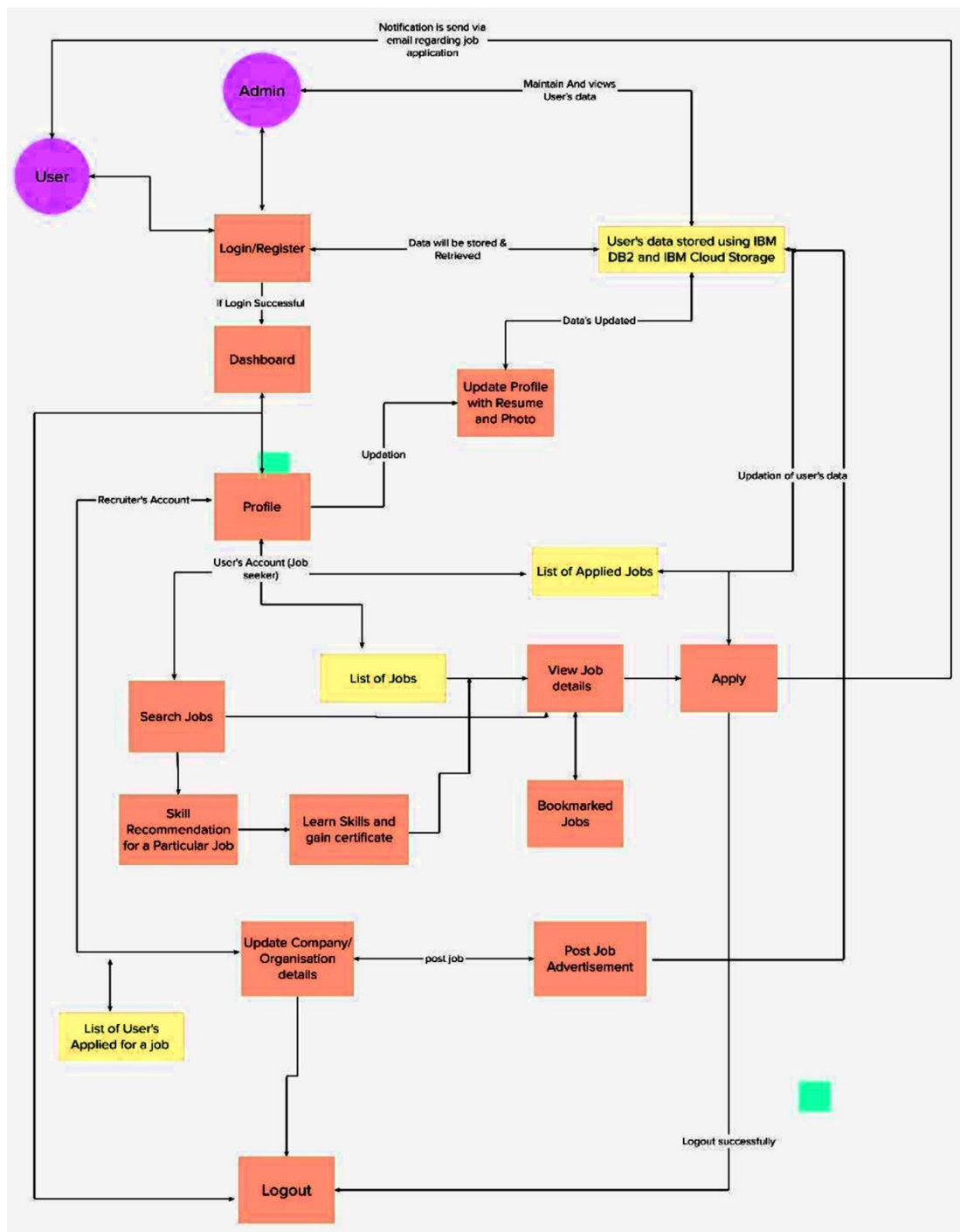
Users may install nonfunctional requirements that enhance the overall performance of their system. Typically, the level of speed that a system has is important for users, since they may want a system that works quickly. Nonfunctional requirements may boost a computer's speed and efficiency. For example, if an employee submits a project, having high speed may help the project submit quickly.

5.PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.





5.2 SOLUTION AND TECHNICAL ARCHITECTURE

Solution architecture is a complex process – with many sub-processes

– that bridges the gap between business problems and technology solutions.

Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.

Provide specifications according to which the solution is defined, managed, and delivered.

Job Seeker

- First, We register in the application and after registration login to the platform and upload the CV.

Job Recruiter

- Recruiter login the application then upload job advertisement.
- Analysis the fake job offer detection and eliminates it.

Feature Extraction

- The Extraction of user details from resume is done using TF-IDF Technique.
- The Extracted user educational documents and details are verified properly.
- Dynamic features extraction are done from user activity.
- Feature Extracted from job post including job profile, job roles and

responsibilities and much more using TF-IDF technique.

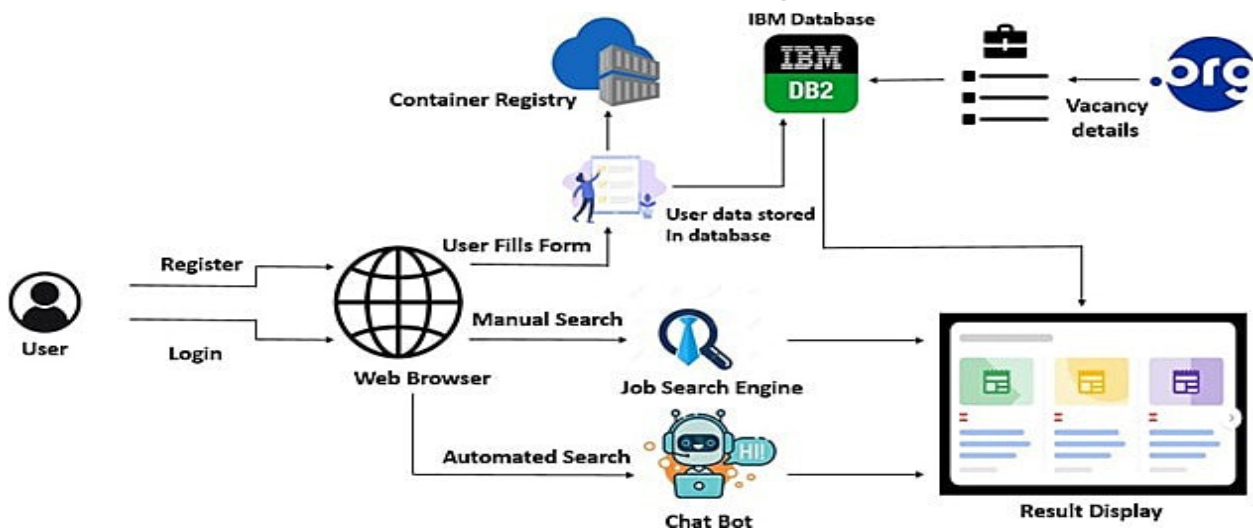
Job Recommendation

- Jobs will be recommended based on user Skillset.
- A notification via email regarding a job post.
- 'N' No.of.jobs will be recommended to user(job seeker).

Skill Recommendation

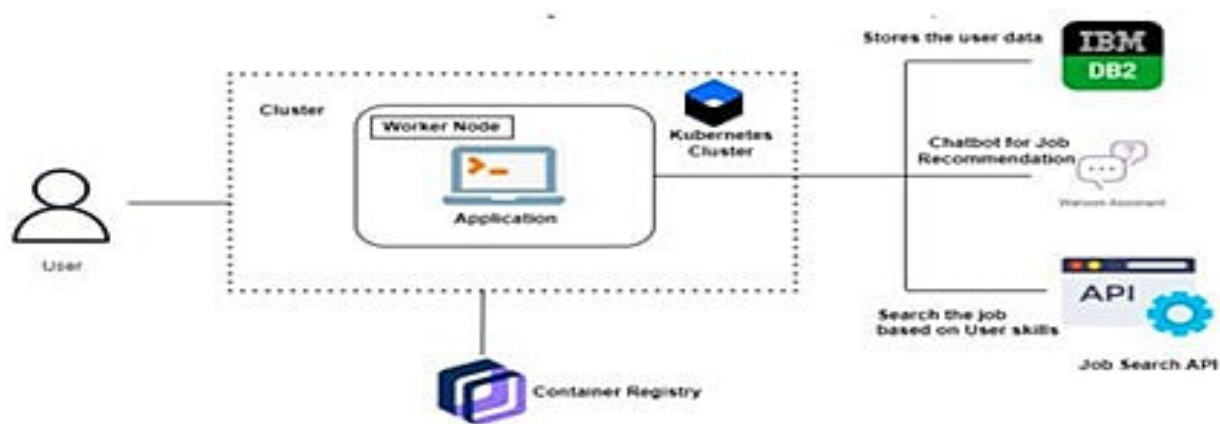
- Skill recommendation will be given for a particular job.
- Learning Resources will be provided and instruction will be given to utilize open source.
- Filtering the skills from the extracted feature set which the job seeker doesn't already possess.

Solution Architecture Diagram



TECHNICAL

They can also register with the help of chatbot. The user data will be stored and retrieved with the help of this database. The user documents like photos, resumes and much more will be stored in cloud bucket, etc., With the help of API, the user can search the job based on their Skillset. Application Deployment on Local System / Cloud



5.3 USER STORIES

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
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 access my account /dashboard	High	Sprint-1

		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through online websites	I can register & access the dashboard with online website Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can receive confirmation Gmail & click confirm	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can receive confirmation email & click confirm	High	Sprint-1
	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 skill sets I can able to get personalised job recommendations.	I can receive job recommendations	High	Sprint-1
Customer Care Executive		USN-7	As a customer care executive	24/7 chatbot support	High	Sprint-

			e, we provide 24/7 chatbot support.			1
Administrator		USN-8	As an administrator, I can able to view the progress and make required changes in the project	Deploy user specific and personalised job recommendations	High	Sprint-1

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	UI Design & Frontend Development	USN-1	As a user I can expect to experience a cool user interface and smooth user experience	8	High	Arun Kumar K
Sprint-1	Home	USN-2	As a user, I will land on the landing page of the website	1	High	Kaviyarasan R
Sprint-1	Database	USN-3	As a user my data will be stored in database for further use	5	High	Sanjay Kumar V
Sprint-1	Registration	USN-4	As a user, I can register for the application as a Job seeker or Recruiter.	2	High	Nitheshwaran T
Sprint-2		USN-5	As a user, I will receive verification email once I have registered for the application	2	Medium	Kaviyarasan R
Sprint-3		USN-6	As a user, I can register for the application through Google Signup	3	Low	Sanjay Kumar V
Sprint-3		USN-7	As a user, I can register for the application through Signin with LinkedIn	3	Low	Arun Kumar K

Sprint-1	Login	USN-8	As a user, I can log into the application as Jobseeker or Recruiter by entering registered email & correct password	2	High	Nitheshwaran T
Sprint-3		USN-9	As a user, I can log into the application using google sign in option	3	Low	Arun Kumar K

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	UI Design & Frontend Development	USN-1	As a user I can expect to experience a cool user interface and smooth user experience	8	High	Nitheshwaran T
Sprint-1	Home	USN-2	As a user, I will land on the landing page of the website	1	High	Sanjay Kumar V
Sprint-1	Database	USN-3	As a user my data will be stored in database for further use	5	High	Arun Kumar K
Sprint-3		USN-10	As a user, I can log into the application using LinkedIn Login	3	Low	Arun Kumar K
Sprint-2	Profile Setup	USN-11	As a fresh user I need to setup my profile initially by filling required details which can be modified later	3	High	Kaviyarasan R
Sprint-2		USN-12	As a fresh recruiter I need to setup profile for my company by filling required details which can be modified later	3	High	Kaviyarasan R

Sprint-2	Cloud Storage	USN-13	As a user I can upload my Image, Resume and much more in the website	3	Medium	Nitheshwaran T
Sprint-2	Posting	USN-14	As a Recruiter I can post various job openings	5	High	Kaviyarasan R
Sprint-2	Job Listing	USN-15	As a user I can access jobs posted by recruiters and Google Job Search API	5	High	Sanjay Kumar V
Sprint-2	Applying	USN-16	As a Job Seeker I can view all Job openings in the home page and also, I can search for specific jobs and apply for the same	3	High	Nitheshwaran T
Sprint-2	Shortlisting	USN-17	As a Recruiter I can view applied candidates and shortlist few among them.	3	High	Arun Kumar K , Nitheshwaran T
Sprint-3	Chatbot	USN-18	As a User I can access chatbot to avail any kind of guidance in the website	5	High	Sanjay Kumar V, Kaviyarasan R
Sprint-3	Notification (SendGrid)	USN-19	As a User, I can get notification on new Job openings via email using SendGrid service	3	Medium	Arun Kumar K

Sprint-3	Courses & Webinars	USN-20	As an administrator I can suggest users various courses from famous websites like Udemy, Coursera based on their skillset to improve their skills	3	Low	Kaviyarasan R
----------	--------------------	--------	---	---	-----	---------------

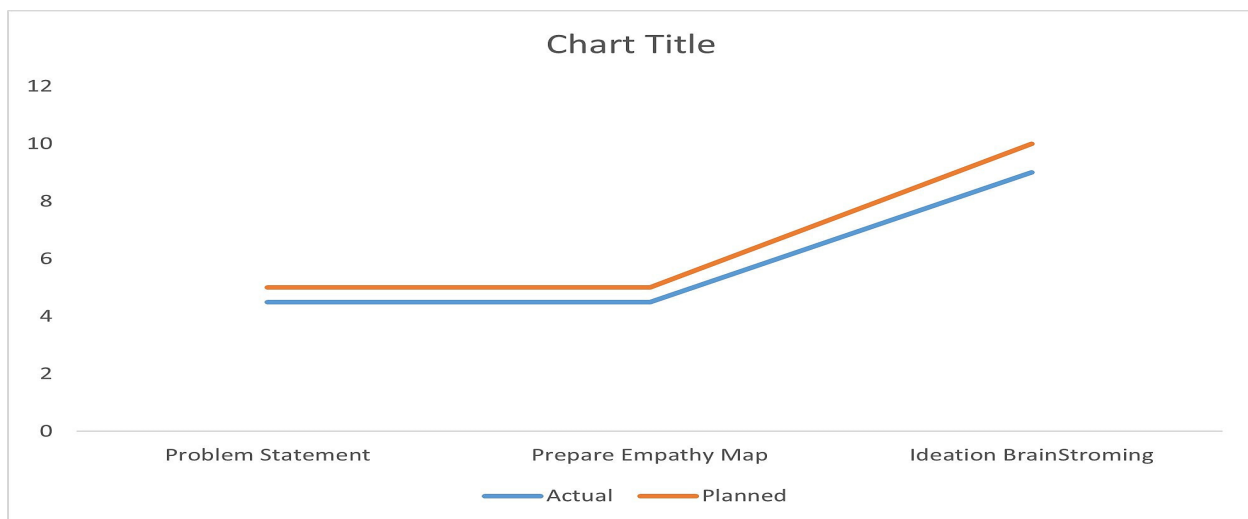
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	UI Design & Frontend Development	USN-1	As a user I can expect to experience a cool user interface and smooth user experience	8	High	Arun Kumar K
Sprint-1	Home	USN-2	As a user, I will land on the landing page of the website	1	High	Nitheshwaran T
Sprint-1	Database	USN-3	As a user my data will be stored in database for further use	5	High	Arun Kumar K,
Sprint-3	Interviews	USN-21	As a recruiter I can schedule face to face interview with shortlisted candidates using WebRTC framework	3	Low	Sanjay Kumar V, Nitheshwaran T
Sprint-3		USN-22	As a shortlisted candidate I can join the scheduled interview using the meeting link	3	Low	Arun Kumar K, Sanjay Kumar V
Sprint-4	System testing	USN-23	As a user I can access my website without any fault or malfunction	5	Medium	Nitheshwaran T
Sprint-4	Docker	USN-24	As a user I can access my containerized application in any device	3	High	Kaviyarasan R, Sanjay Kumar V

Sprint-4	Kubernetes	USN-25	As a user I can access my containerized application in any device with greater security	5	High	Sanjay Kumar V, Arun kumar K
Sprint-4	Deployment in the Cloud	USN-26	As a user I can access the website from anywhere in the world	3	High	Nitheshwaran T, Kaviyarasan R

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	26 Oct 2022	31 Oct 2022	20	18 Oct 2022
Sprint-2	25	6 Days	02 Nov 2022	07 Nov 2022	25	18 Nov 2022
Sprint-3	30	6 Days	08 Nov 2022	13 Nov 2022	30	18 Nov 2022
Sprint-4	15	6 Days	14 Nov 2022	19 Nov 2022	15	18 Nov 2022

6.3 REPORTS FROM JIRA



7. CODING AND SOLUTIONING

7.1 FEATURE 1

```
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=fbd88901-ebdb-4a4f-
a32e-
9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32731;S
ECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=xbd1166
6;PWD=sl50jj06TEZWtPSQ",",")

@app.route('/')

def homer():
    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'] = True
    session['id'] = account['USERNAME']
    userid= account['USERNAME']
    session['username'] = account['USERNAME']
    msg = 'Logged in successfully !'

    msg = 'Logged in successfully !'
    return render_template('dashboard.html', msg = msg)
else:
    msg = 'Incorrect username / password !'
return render_template('login.html', msg = msg)

```

```

@app.route('/register', methods=['GET', 'POST'])

```

```

def registet():

```

```

    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 exists !'
    elif not re.match(r'^@]+@^[^@]+\.[^@]+', email):
        msg = 'Invalid email address !'
    elif not re.match(r'[A-Za-z0-9]+', username):
        msg = 'name must contain only characters and numbers !'
    else:
        insert_sql = "INSERT INTO users VALUES (?, ?, ?)"
        prep_stmt = ibm_db.prepare(conn, insert_sql)
        ibm_db.bind_param(prepare_stmt, 1, username)
        ibm_db.bind_param(prepare_stmt, 2, email)
        ibm_db.bind_param(prepare_stmt, 3, password)
        ibm_db.execute(prepare_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 =?"
```

```
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 = 'there is only 1 job position! for you'
```

```
    return render_template('apply.html', msg = msg)
```

```
insert_sql = "INSERT INTO job VALUES (?, ?, ?, ?, ?)"
```

```
prep_stmt = ibm_db.prepare(conn, insert_sql)
```

```
ibm_db.bind_param(prepare_stmt, 1, username)
```

```
ibm_db.bind_param(prepare_stmt, 2, email)
```

```
ibm_db.bind_param(prepare_stmt, 3, qualification)
```

```
ibm_db.bind_param(prepare_stmt, 4, skills)
```

```
ibm_db.bind_param(prepare_stmt, 5, jobs)
```

```
ibm_db.execute(prepare_stmt)
```

```
msg = 'You have successfully applied for job !'
```

```
session['loggedin'] = True
```

```
TEXT = "Hello sandeep,a new appliaction for job position" +jobs+"is  
requested"
```

```
#sendmail(TEXT,"sandeep@thesmartbridge.com")
```

```
sendgridmail("sandeep@thesmartbridge.com",TEXT)
```

```
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)
```

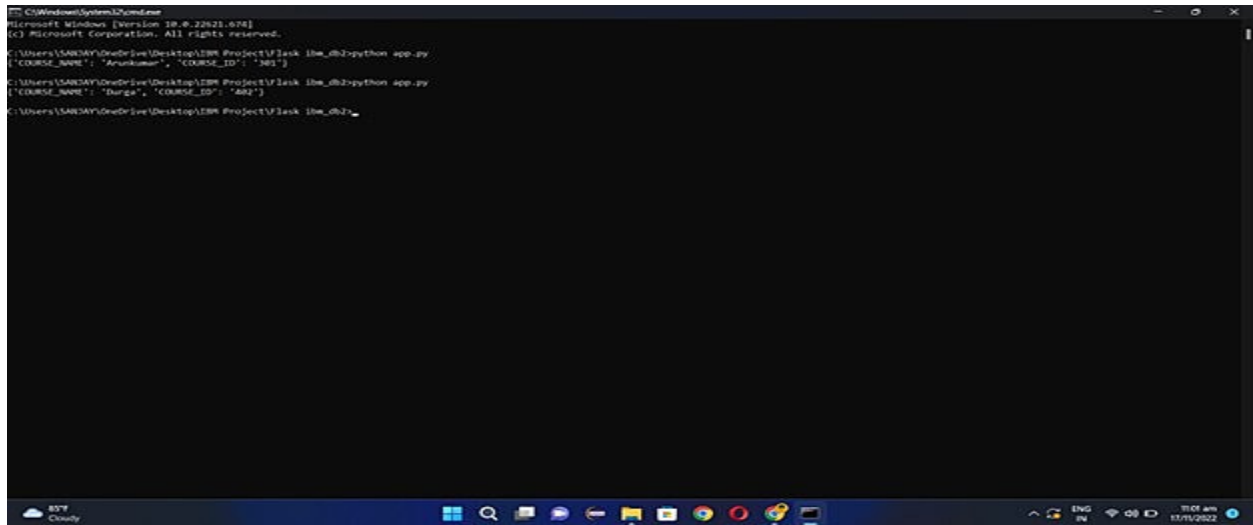
```
    session.pop('id', None)
```

```
    session.pop('username', None)
```

```
    return render_template('home.html')
```

```
if __name__ == '__main__':
```

```
    app.run(host='0.0.0.0')
```



7.2 FEATURE CODE 2

```
import re
```

```
from flask import Flask, render_template, request, session, redirect
```

```
from flask_session import Session
```

```
import ibm_db
```

```
import os
```

```
app=Flask(__name__)
```

sessions

```
app.config["SESSION_PERMANENT"] = False
```

```
app.config["SESSION_TYPE"] = "filesystem"
```

Session(app)

##-----Database-----

```
app.secret_key = 'a'
```

```
app.config['database'] = 'bludb'
```

```
app.config['hostname'] = 'fbd88901-ebdb-4a4f-a32e-  
9822b9fb237b.c1ogj3sd0tgu0lqde00.databases.appdomain.cloud'
```

```
app.config['port'] = '32731'
```

```
app.config['protocol'] = 'tcpip'
```

```
app.config['uid'] = 'xbd11666'
```

```
app.config['pwd'] = 'sl50jj06TEZWtPSQ'
```

```
app.config['security'] = 'SSL'
```

```
conn_str = 'database=bludb;hostname=fbd88901-ebdb-4a4f-a32e-  
9822b9fb237b.c1ogj3sd0tgu0lqde00.databases.appdomain.cloud;port=32731;prot  
ocol=tcpip;\
```

```
uid=xbd11666;pwd=sl50jj06TEZWtPSQ;security=SSL'
```

```
conn = ibm_db.connect(conn_str, "", "")
```

```
#-----routing-----
```

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

```
def home():
```

```
    session['name'] = "
```

```
    session['email'] = "
```

```
    session['apply'] = False
```

```
    session['title'] = "
```

```
    session['skills'] = "
```

```
    session['company'] = "
```

```
    session['msg'] = "
```

```
session['title'] = "  
session['skills'] = "  
session['regmsg'] = "  
return render_template('login.html')
```

```
@app.route('/apply',methods=['POST'])
```

```
def apply():
```

```
    session['apply']= True  
    session['title'] = request.form['title']  
    session['skills'] = request.form['skill']  
    session['company'] = request.form['company']  
    if(session['name']== ""):  
        session['msg'] = "Please login before applying job."  
        return render_template('login.html')  
    return render_template('apply.html')
```

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

```
def viewjobs():
```

```
    session['title'] = "  
    session['skills'] = "  
    session['msg'] = "  
    session['regmsg'] = "  
    sql = "select * from jobs"  
    stmt = ibm_db.prepare(conn, sql)
```



```

ibm_db.execute(stmt)

values = ibm_db.fetch_assoc(stmt)

result = ""

while values != False:

    result += '<div class="card" style="width:20rem">'

    result += '<form action="apply" class="card-body" method="post">'

    result += '<input class="card-title title" name="title" value="" +
values['ROLE'] + "">'

    result += '<input class="card-text company" name="company" value="" +
values['COMPANY'] + ""><br>'

    result += '<input class="card-text skills" name="skill" value="" +
values['SKILLS'] + ""><br>'

    result += '<input type="submit" value="Apply" class="btn btn-
primary"></form></div> '

    values = ibm_db.fetch_assoc(stmt)

session['result'] = result

if session['name'] == "":

    return render_template('viewJob.html')

return render_template('viewafterlogin.html')


@app.route('/login')

def login():

    session['apply']= False

    return render_template('login.html')

```

```

@app.route('/register',methods=['POST','GET'])
def register():
    if request.method == 'GET':
        return render_template('register.html')
    name = request.form['name']
    email = request.form['email']
    password = request.form['password']
    # ----- tablename = jobregister-----
    sql="SELECT * FROM jobregister WHERE name=? "
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,name)
    ibm_db.execute(stmt)
    account =ibm_db.fetch_assoc(stmt)
    if account:
        msg='Account already exists !'
        return render_template('register.html',msg=msg)
    elif not re.match(r'^[@]+@[^@]+\.[^@]+',email):
        msg='Invalid email Address'
        return render_template('register.html',msg=msg)
    elif not re.match(r'[A-Za-z0-9]+',name):
        msg='Name must contain atleast one character and Number'
        return render_template('register.html',msg=msg)
    else:
        # ----- tablename = jobregister-----

```

```

sql = 'INSERT INTO jobregister values(?,?,?)'
prepare_stmt = ibm_db.prepare(conn,sql)
ibm_db.bind_param(prepare_stmt,1,name)
ibm_db.bind_param(prepare_stmt,2,email)
ibm_db.bind_param(prepare_stmt,3,password)
ibm_db.execute(prepare_stmt)
session['msg']='Registration successful'
return render_template('login.html')

```

```

@app.route('/welcome',methods=['POST','GET'])

```

```

def welcome():

```

```

    if request.method == 'GET':

```

```

        return render_template('welcome.html')

```

```

    email = request.form['email']

```

```

    password = request.form['password']

```

```

    #-----table name = jobregister -----

```

```

    sql = "select name from jobregister where email=? AND password=?"

```

```

    prep_stmt = ibm_db.prepare(conn, sql)

```

```

    ibm_db.bind_param(prepare_stmt, 1, email)

```

```

    ibm_db.bind_param(prepare_stmt, 2, password)

```

```

    ibm_db.execute(prepare_stmt)

```

```

    values = ibm_db.fetch_assoc(prepare_stmt)

```

```

    if values:

```

```

        session['name'] = values['NAME']

```

```
session['email'] = email

if session['apply']:
    return render_template('apply.html')
return render_template("welcome.html")
else:
    session['msg'] = "Incorrect email or password"
    return redirect("/login")
```

```
@app.route('/')
def hello():
    return render_template('index.html')
```

```
@app.route('/index')
def index():
    return render_template('index.html')
```

```
@app.route('/job_listing')
def job_listing():
    return render_template('job_listing.html')
```

```
@app.route('/job_details')
def job_details():
    return render_template('job_details.html')
```

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

```
def about():
```

```
    return render_template('about.html')
```

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

```
def contact():
```

```
    return render_template('contact.html')
```

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

```
def register():
```

```
    return render_template('register.html')
```

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

```
def login():
```

```
    if request.method == 'POST':
```

```
        username = request.form['email']
```

```
        password = request.form['password']
```

```
        sql = "SELECT * FROM users WHERE user_mail =? AND  
user_password=?"
```

```
        stmt = ibm_db.prepare(con, 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)
```

```
        session['loggedin'] = True
```

```
        session['username'] = account['USERNAME']
```

```
        session['email'] = account['USER_MAIL']
```

```
        return redirect('/')
```

```
else:  
    return render_template('login.html')
```

```
@app.route('/logout')  
def logout():  
    session.pop('loggedin', None)  
    session.pop('username', None)  
    session.pop('email', None)  
    return redirect('/')
```

```
@app.route('/single-blog')  
def single_blog():  
    return render_template('single-blog.html')
```

```
@app.route('/apply')  
def apply():  
    return render_template('apply.html')
```

```
@app.route('/uploader')  
def uploader():  
    return render_template('uploader.php')
```

```
if __name__ == "__main__":  
    app.run(debug=True)
```

Skills and Job Recommender x +

127.0.0.1:5000/contact

Get in Touch

Enter Message

Enter your name

Enter Subject

SEND

Anna University Regional Campus,
Madurai, 625019

+91 9876543210

aurcm@gmail.com
Send us your query anytime!

28°C Haze 11:28 19-11-2022

Job board HTML-5 Template x +

127.0.0.1:5000/job_listing

Filter Jobs

9 Jobs found

Sort by None

Job Category

All Category

Job Type

☐ Full Time


☒ Part Time


☐ Remote


☐ Freelance

Job Location

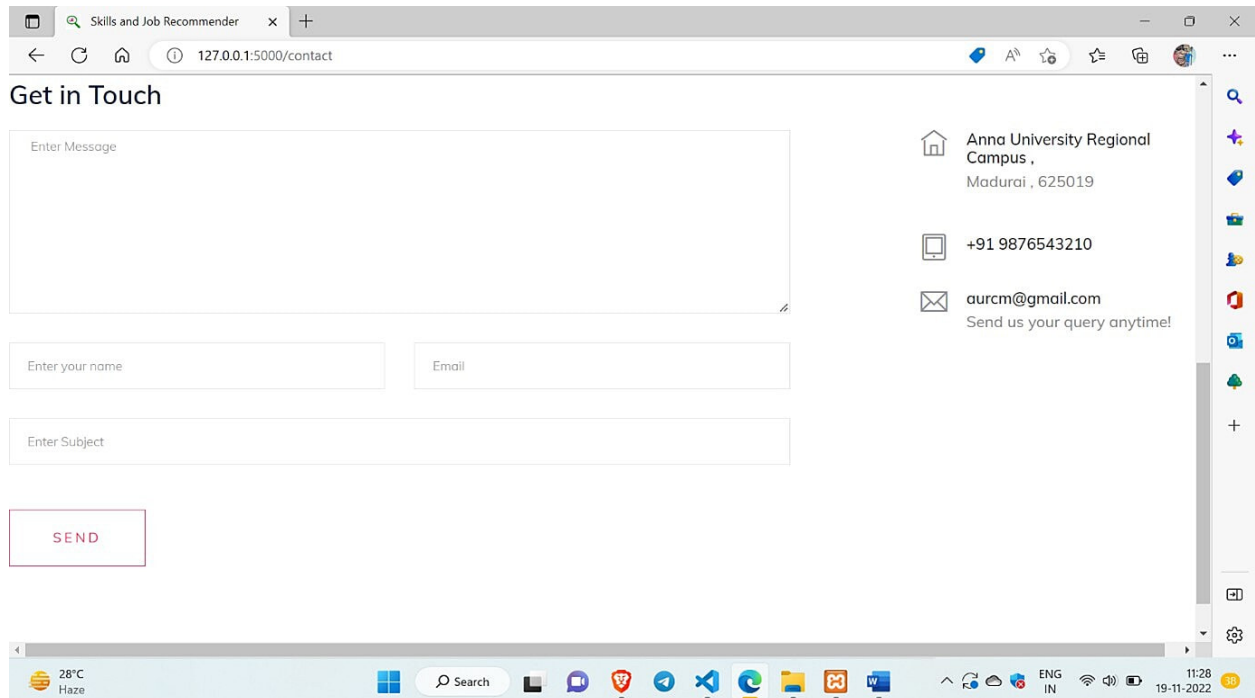
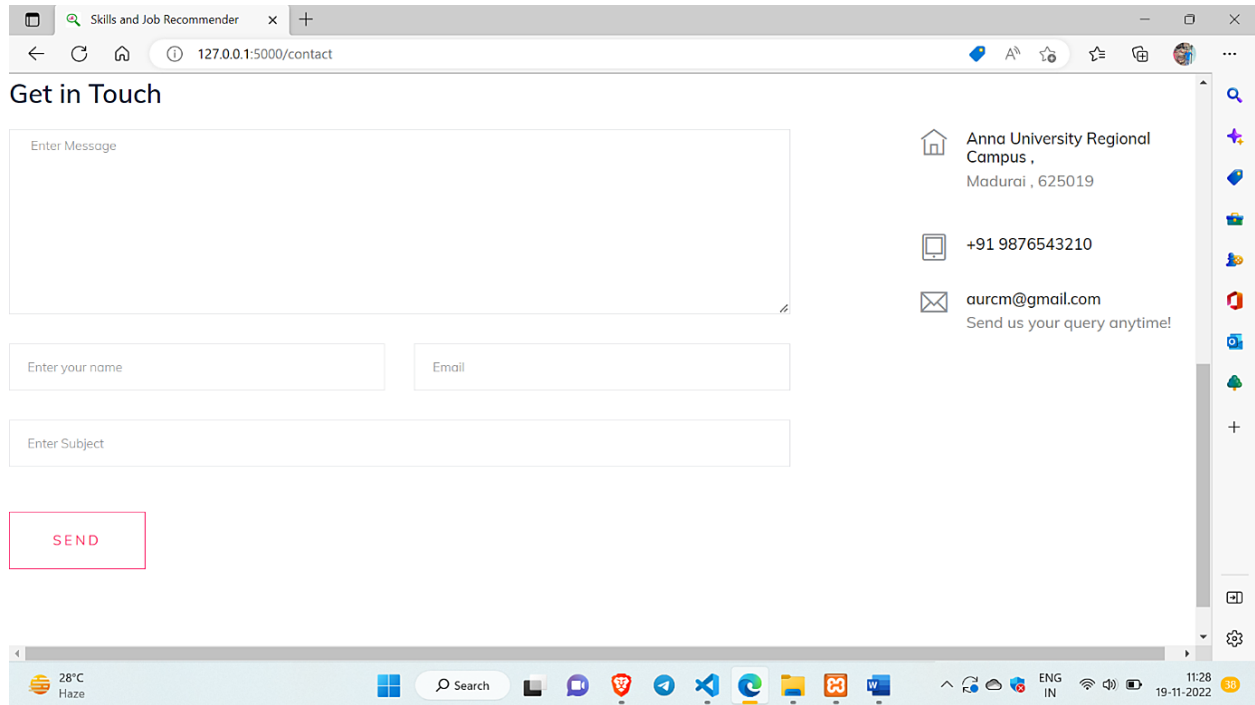
Anywhere

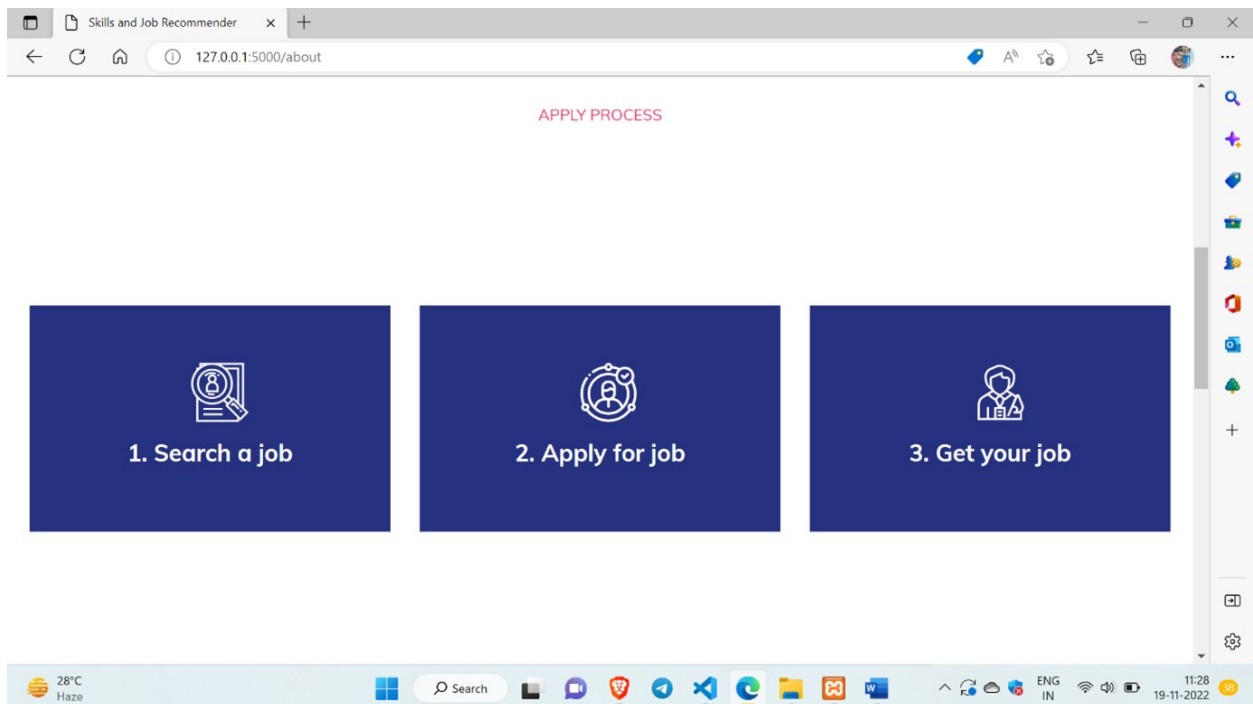
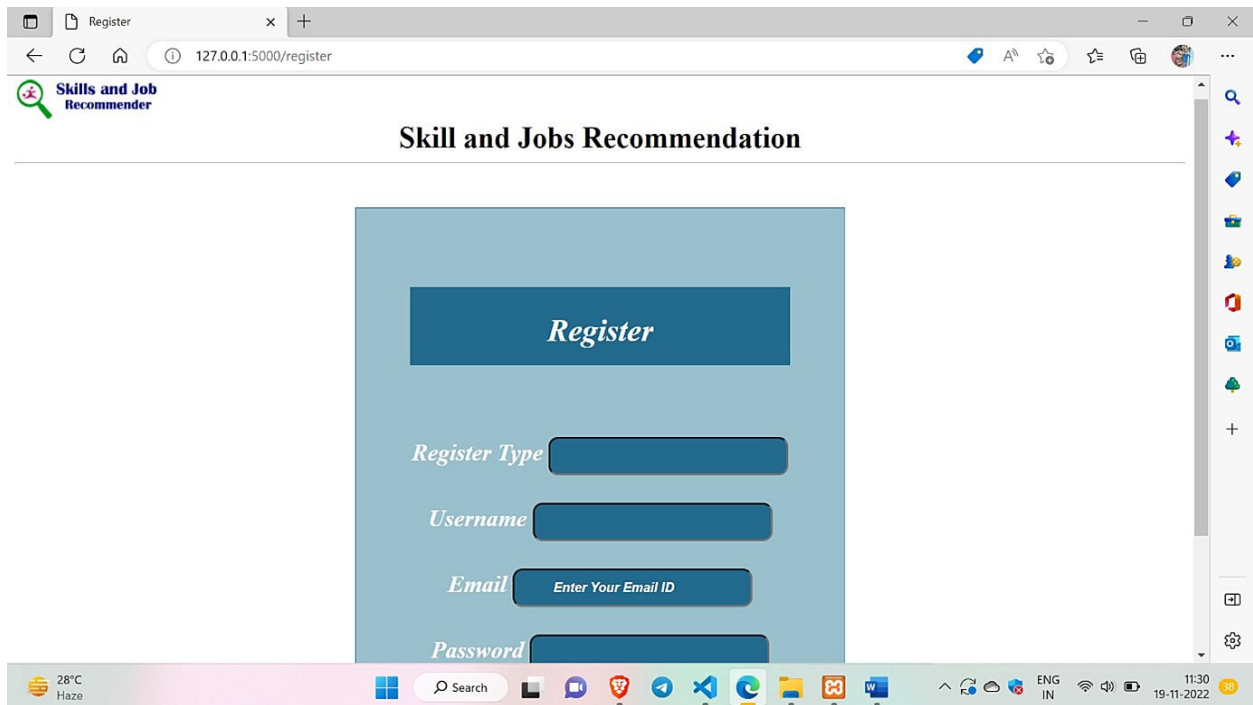
 **Digital Marketer**
Coimbatore 5LPA
3 hours ago

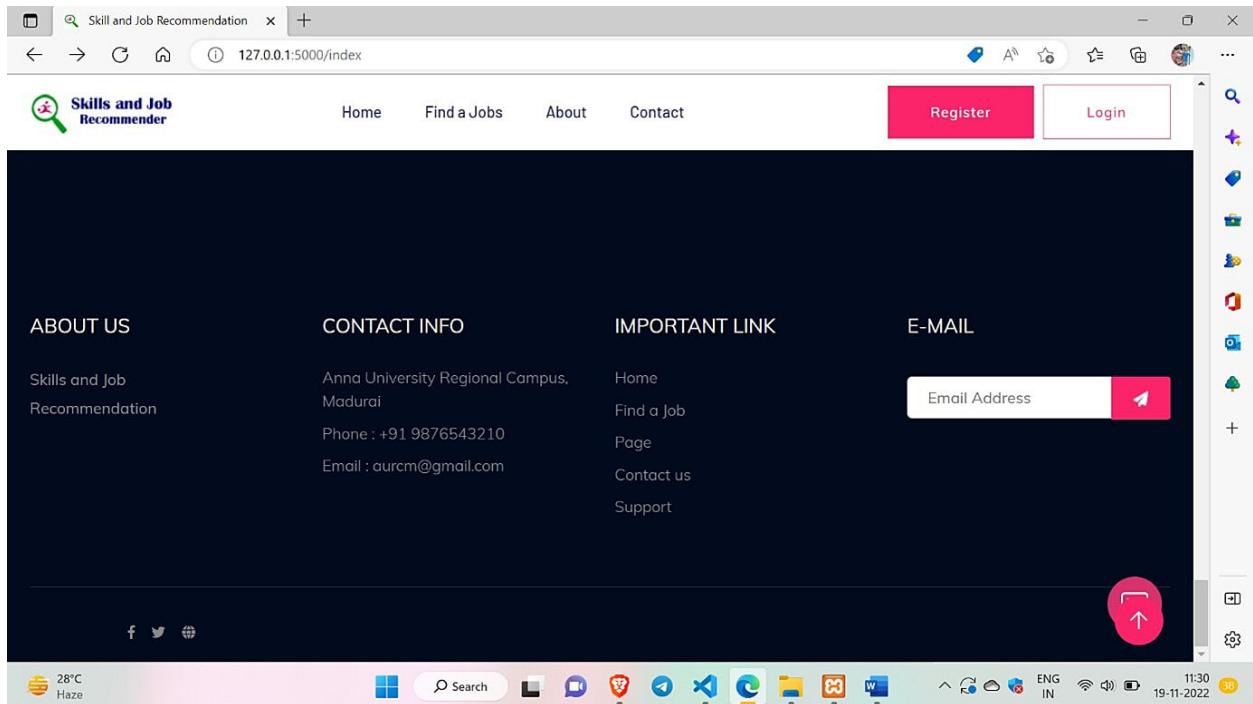
 **System Engineer**
Bangalore 8LPA
3 hours ago

 **Data Analyst**
Chennai 16LPA
4 hours ago

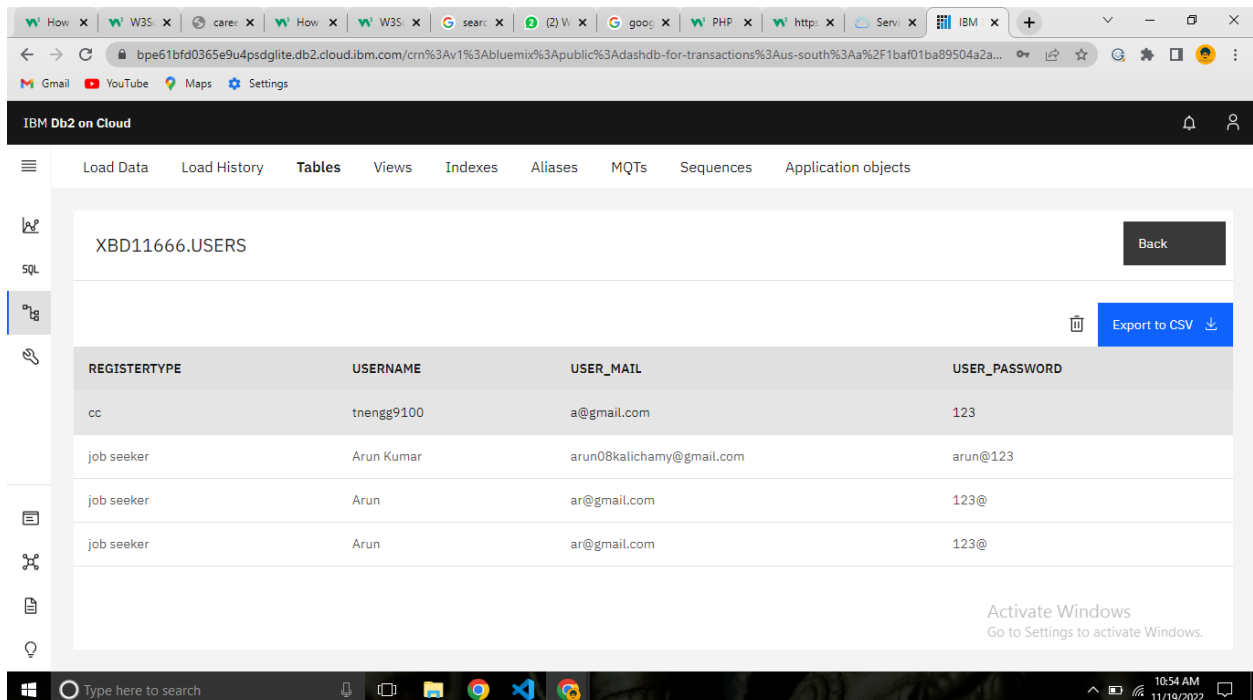
28°C Haze 11:26 19-11-2022







7.3 DATABASE SCHEMAS



8.TESTING

8.1 TEST CASES

S.No	Parameter	Values	Screen Shot
1	Model Summary	-	
2	Client Application	Training Accuracy-98%	
3	Confidence Score(Only Yolo Projects)	Confidence Score-85-90	

8.2 USER ACCEPTANCE TESTING

1.Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [CourseName] project at the time of the release to User Acceptance Testing (UAT).

2.Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37

Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Registration	7	0	0	7
Client Application	50	0	0	50
Login	2	0	0	2
Security	3	0	1	3
Database Maintenance	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

9.1 PERFORMANCE METRICS

S.No	Parameter	values	Screenshot
1	Mode summary	Parameter	
2	Client Application	Training Accuracy=98%	

3	Confidence Score(Only Yolo Project)	Confidence Score=85.90%	
---	---	----------------------------	--

10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Better Candidate Quality. ...
- Faster Hiring Time. ...
- Cost-Efficient. ...
- Better Retention Rate. ...
- Boosts Morale. ...
- Biased Recommendations. ...
- Tension in the Workplace. ...
- Decreased Workforce Diversity.

DISADVANTAGES

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

11. CONCLUSION

In this paper, we have considered the job recommender system (JRS) literature from several perspectives.

These include the influence of data science competitions, the effect of data availability on the choice of method and validation, and ethical considerations in job recommender systems. Furthermore, we branched the large class of hybrid recommender systems to obtain a better view on how these hybrid recommender systems differ. Both this multi-perspective view, and the new taxonomy of hybrid job recommender systems has not been discussed by previous reviews on job recommender systems.

12. FUTURE SCOPE

The objective of recommender systems is to provide recommendations based on recorded information on the users' preferences. These systems use information filtering techniques to process information and provide the user with potentially more relevant items. Here's where recommendation systems come in handy: they shorten the distance between the customer's need and satisfaction. Not only do they help find the search product; they also discover the needs that customers are not even aware they have.

13. APPENDIX

SOURCE CODE:

Resiter.html

```
<html>
<head>
<meta charset="UTF-8">
<title> Register </title>
<link rel="stylesheet" href="static/css/reg&log.css">
</head>
```

```
<body>
<div class="logo">
<a href="#"></a>
<div class="heading">Skill and Jobs Recommendation</div>
</div>
<hr>
</br></br>
<div align="center">
<div align="center" class="border">
<div class="header">
<h1 class="word">Register</h1>
</div></br></br></br>
<h2 class="word">
<form action="{{ url_for('register') }}" method="post">

<label for="registype"> <b>Register Type</b></label>
<input id="registertype" name="registertype" type="text" class="textbox" required
style="color: white;"><br><br>
<label for="uname"> <b>Username</b> </label>
<input id="username" name="username" type="text" class="textbox" required
style="color: white;"></br></br>
<label for="email"> <b>Email</b> </label>
<input id="email" name="email" type="email" placeholder="Enter Your Email
ID" class="textbox" required style="color: white;"></br></br>
<label for="pass"> <b>Password</b> </label>
<input id="password" name="password" type="password" class="textbox"
required style="color: white;"></br></br>

<input type="submit" class="btn" value="Sign Up"></br>
```

```
</form>
</h2>
<p class="bottom">Already have an account? <a class="bottom"
href="{ {url_for('login')}} "> Sign In here</a></p>
</div>
</div>
</body>
</html>
```

Index.html

```
<!doctype html>
<html class="no-js" lang="zxx">
<head>
<meta charset="utf-8">
<meta http-equiv="x-ua-compatible" content="ie=edge">
<title>Skill and Job Recommendation</title>
<meta name="description" content="">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="manifest" href="site.webmanifest">
<link rel="shortcut icon" type="image/x-icon" href="static/img/favicon.jpg">

<!-- CSS here -->
<link rel="stylesheet" href="static/css/bootstrap.min.css">
<link rel="stylesheet" href="static/css/owl.carousel.min.css">
<link rel="stylesheet" href="static/css/flaticon.css">
<link rel="stylesheet" href="static/css/price_rangs.css">
<link rel="stylesheet" href="static/css/slicknav.css">
<link rel="stylesheet" href="static/css/animate.min.css">
<link rel="stylesheet" href="static/css/magnific-popup.css">
```



```
<link rel="stylesheet" href="static/css/fontawesome-all.min.css">
<link rel="stylesheet" href="static/css/themify-icons.css">
<link rel="stylesheet" href="static/css/slick.css">
<link rel="stylesheet" href="static/css/nice-select.css">
<link rel="stylesheet" href="static/css/style.css">
</head>

<body>
<!-- Preloader Start -->
<!-- <div id="preloader-active">
<div class="preloader d-flex align-items-center justify-content-center">
<div class="preloader-inner position-relative">
<div class="preloader-circle"></div>
<div class="preloader-img pere-text">

</div>
</div>
</div>
</div>
</div> -->
<!-- Preloader Start -->
<header>
<!-- Header Start -->
<div class="header-area header-transparent">
<div class="header-top header-sticky">
<div class="container">
<div class="row align-items-center">
<div class="col-lg-3 col-md-2">
<!-- Logo -->
<div class="logo">
```

```
<a href="#"></a>
</div>
</div>
<div class="col-lg-9 col-md-9">
<div class="menu-wrapper">
<!-- Main-menu -->
<div class="main-menu">
<nav class="d-none d-lg-block">
<ul id="navigation">
<li><a href="index">Home</a></li>
<li><a href="job_listing">Find a Jobs </a></li>
<li><a href="about">About</a></li>
<li><a href="contact">Contact</a></li>
</ul>
</nav>
</div>
<!-- Header-btn -->
<div class="header-btn d-none f-right d-lg-block">
<a href="register" class="btn head-btn1">Register</a>
<a href="login" class="btn head-btn2">Login</a>
</div>
</div>
</div>
<!-- Mobile Menu -->
<div class="col-12">
<div class="mobile_menu d-block d-lg-none"></div>
</div>
</div>
</div>
```

```
</div>
</div>
<!-- Header End -->
</header>
<main>

<!-- slider Area Start-->
<div class="slider-area ">
<!-- Mobile Menu -->
<div class="slider-active">
<div class="single-slider slider-height d-flex align-items-center" data-
background="static/img/hero/h1_hero.jpg">
<div class="container">
<div class="row">
<div class="col-xl-6 col-lg-9 col-md-10">
<div class="hero__caption">
<h1>Find the most exciting startup jobs</h1>
</div>
</div>
</div>
<!-- Search Box -->
<div class="row">
<div class="col-xl-8">
<!-- form -->
<form action="#" class="search-box">
<div class="input-form">
<input type="text" onkeyup="showResult(this.value)" placeholder="Job Tittle or
keyword">
<div id="livesearch"></div>
```

```
</div>
<div class="select-form">
<div class="select-itms">
<select name="select" id="select1">
<option value="">COIMBATORE</option>
<option value="">CHENNAI</option>
<option value="">MADURAI</option>
<option value="">BANGLORE</option>
</select>
</div>
</div>
<div class="search-form">
<a href="#">Find job</a>
</div>
</form>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- slider Area End-->
```

```
<!-- Our Services Start -->
<div class="our-services section-pad-t30">
<div class="container">
<!-- Section Tittle -->
<div class="row">
<div class="col-lg-12">
```

```
<div class="section-tittle text-center">
<h2>Browse Top Categories </h2>
</div>
</div>
</div>
<div class="row d-flex justify-content-center">
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-tour"></span>
</div>
<div class="services-cap">
<h5>Design & Creative</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-cms"></span>
</div>
<div class="services-cap">
<h5>Design & Development</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
```

```
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-report"></span>
</div>
<div class="services-cap">
<h5>Sales & Marketing</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-app"></span>
</div>
<div class="services-cap">
<h5>Mobile Application</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-helmet"></span>
</div>
<div class="services-cap">
<h5>Construction</h5>
<a href="apply"><button class="btn">Apply</button></a>
```

```
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-high-tech"></span>
</div>
<div class="services-cap">
<h5>Information Technology</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-real-estate"></span>
</div>
<div class="services-cap">
<h5>Real Estate</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-services text-center mb-30">
<div class="services-ion">
<span class="flaticon-content"></span>
```

```
</div>
<div class="services-cap">
<h5>Content Writer</h5>
<a href="apply"><button class="btn">Apply</button></a>
</div>
</div>
</div>
</div>
<!-- More Btn -->
<!-- Section Button -->
<div class="row">
<div class="col-lg-12">
<div class="browse-btn2 text-center mt-50">
<a href="job_listing" class="border-btn2">Browse All Sectors</a>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- Our Services End -->

<!-- Online CV Area Start -->
<div class="online-cv cv-bg section-overly pt-90 pb-120" data-
background="static/img/gallery/cv_bg.jpg">
<div class="container">
<div class="row justify-content-center">
<div class="col-xl-10">
<div class="cv-caption text-center">
<!-- <p class="pera1">FEATURED TOURS Packages</p> -->
```



```
<p class="pera2"> Make a Difference with Your Online Resume!</p>
<a href="uploader" class="border-btn2 border-btn4" type="file">Upload your
cv</a>
<!-- <form action="" method="POST" enctype="multipart/form-data">
<input type="file" name="image" class="border-btn2 border-btn4" />
<input type="submit"/>
</form> -->

</div>
</div>
</div>
</div>
</div>
<!-- Online CV Area End-->
<!-- Featured_job_start -->
<section class="featured-job-area feature-padding">
<div class="container">
<!-- Section Tittle -->
<div class="row">
<div class="col-lg-12">
<div class="section-tittle text-center">
<span>Recent Job</span>
<h2>Featured Jobs</h2>
</div>
</div>
</div>
<div class="row justify-content-center">
<div class="col-xl-10">
<!-- single-job-content 1-->
```

```
<div class="single-job-items mb-30">
<div class="job-items">
<div class="company-img">
<a href="apply"></a>
</div>
<div class="job-tittle">
<a href="apply"><h4>Digital Marketer</h4></a>
<ul>
<li><i class="fas fa-map-marker-alt"></i>Coimbatore</li>
<li>5LPA</li>
</ul>
</div>
</div>
<div class="items-link f-right">
<a href="apply">Apply</a>
<span>3 hours ago</span>
</div>
</div>
<!-- single-job-content 2-->
<div class="single-job-items mb-30">
<div class="job-items">
<div class="company-img">
<a href="apply"></a>
</div>
<div class="job-tittle">
<a href="apply"><h4>System Engineer</h4></a>
<ul>
<li><i class="fas fa-map-marker-alt"></i>Banglore</li>
<li>8LPA</li>
```

```
</ul>
</div>
</div>
<div class="items-link f-right">
<a href="apply">Apply</a>
<span>3 hours ago</span>
</div>
</div>
<!-- single-job-content 3-->
<div class="single-job-items mb-30">
<div class="job-items">
<div class="company-img">
<a href="apply"></a>
</div>
<div class="job-tittle">
<a href="apply"><h4>Data Analyst</h4></a>
<ul>
<li><i class="fas fa-map-marker-alt"></i>Chennai</li>
<li>16LPA</li>
</ul>
</div>
</div>
<div class="items-link f-right">
<a href="apply">Apply</a>
<span>4 hours ago</span>
</div>
</div>
<!-- single-job-content 4-->
<div class="single-job-items mb-30">
```

```
<div class="job-items">
<div class="company-img">
<a href="apply"></a>
</div>
<div class="job-tittle">
<a href="apply"><h4>Junior Software Engineer</h4></a>
<ul>
<li><i class="fas fa-map-marker-alt"></i>Madurai</li>
<li>3LPA</li>
</ul>
</div>
</div>
<div class="items-link f-right">
<a href="apply">Apply</a>
<span>5 hours ago</span>
</div>
</div>
</div>
</div>
</div>
</section>
<!-- Featured_job_end -->
<!-- How Apply Process Start-->
<div class="apply-process-area apply-bg pt-150 pb-150" data-
background="static/img/gallery/how-applybg.png">
<div class="container">
<!-- Section Tittle -->
<div class="row">
<div class="col-lg-12">
```

```
<div class="section-tittle white-text text-center">
<span>Apply process</span>
<h2> How it works</h2>
</div>
</div>
</div>
<!-- Apply Process Caption -->
<div class="row">
<div class="col-lg-4 col-md-6">
<div class="single-process text-center mb-30">
<div class="process-ion">
<span class="flaticon-search"></span>
</div>
<div class="process-cap">
<h5>1. Search a job</h5>
<!-- <p>Sorem spsum dolor sit amsectetur adipisclit, seddo eiusmod tempor
incididunt ut laborea.</p> -->
</div>
</div>
</div>
<div class="col-lg-4 col-md-6">
<div class="single-process text-center mb-30">
<div class="process-ion">
<span class="flaticon-curriculum-vitae"></span>
</div>
<div class="process-cap">
<h5>2. Apply for job</h5>
<!-- <p>Sorem spsum dolor sit amsectetur adipisclit, seddo eiusmod tempor
incididunt ut laborea.</p> -->
```

```
</div>
</div>
</div>
<div class="col-lg-4 col-md-6">
<div class="single-process text-center mb-30">
<div class="process-ion">
<span class="flaticon-tour"></span>
</div>
<div class="process-cap">
<h5>3. Get your job</h5>
<!-- <p>Sorem spsum dolor sit amsectetur adipisclit, seddo eiusmod tempor
incididunt ut laborea.</p> -->
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- How Apply Process End-->

</main>
<footer>
<!-- Footer Start-->
<div class="footer-area footer-bg footer-padding">
<div class="container">
<div class="row d-flex justify-content-between">
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-footer-caption mb-50">
<div class="single-footer-caption mb-30">
```

<div class="footer-tittle">

<h4>About Us</h4>

<div class="footer-pera">

<p>Skills and Job Recommendation</p>

</div>

</div>

</div>

</div>

</div>

<div class="col-xl-3 col-lg-3 col-md-4 col-sm-5">

<div class="single-footer-caption mb-50">

<div class="footer-tittle">

<h4>Contact Info</h4>

Anna University Regional Campus,

Madurai

Phone : +91 9876543210

Email : aurcm@gmail.com

</div>

</div>

</div>

<div class="col-xl-3 col-lg-3 col-md-4 col-sm-5">

```
<div class="single-footer-caption mb-50">
```

```
<div class="footer-tittle">
```

```
<h4>Important Link</h4>
```

```
<ul>
```

```
<li><a href="#">Home </a></li>
```

```
<li><a href="#">Find a Job</a></li>
```

```
<li><a href="#">Page</a></li>
```

```
<li><a href="#">Contact us</a></li>
```

```
<li><a href="#">Support</a></li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-5">
```

```
<div class="single-footer-caption mb-50">
```

```
<div class="footer-tittle">
```

```
<h4>E-mail</h4>
```

```
<!-- Form -->
```

```
<div class="footer-form" >
```

```
<div id="mc_embed_signup">
```

```
<form target="_blank" action="https://spondonit.us12.list-
```

```
manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&id=92a4423d0
```

```
1"
```

```
method="get" class="subscribe_form relative mail_part">
```

```
<input type="email" name="email" id="newsletter-form-email"
```

```
placeholder="Email Address"
```

```
class="placeholder hide-on-focus" onfocus="this.placeholder = ""
```

```
onblur="this.placeholder = ' Email Address '">
```

```
<div class="form-icon">
```



```
<button type="submit" name="submit" id="newsletter-submit"
class="email_icon newsletter-submit button-contactForm"></button>
</div>
<div class="mt-10 info"></div>
</form>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- footer-bottom area -->
<div class="footer-bottom-area footer-bg">
<div class="container">
<div class="footer-border">
<div class="row d-flex justify-content-between align-items-center">
<div class="col-xl-2 col-lg-2">
<div class="footer-social f-right">
<a href="#"><i class="fab fa-facebook-f"></i></a>
<a href="#"><i class="fab fa-twitter"></i></a>
<a href="#"><i class="fas fa-globe"></i></a>
</div>
</div>
</div>
</div>
</div>
</div>
```

</div>

<!-- Footer End-->

</footer>

<!-- JS here -->

<!-- All JS Custom Plugins Link Here here -->

<script src="./static/js/vendor/modernizr-3.5.0.min.js"></script>

<!-- JQuery, Popper, Bootstrap -->

<script src="./static/js/vendor/jquery-1.12.4.min.js"></script>

<script src="./static/js/popper.min.js"></script>

<script src="./static/js/bootstrap.min.js"></script>

<!-- JQuery Mobile Menu -->

<script src="./static/js/jquery.slicknav.min.js"></script>

<!-- JQuery Slick , Owl-Carousel Plugins -->

<script src="./static/js/owl.carousel.min.js"></script>

<script src="./static/js/slick.min.js"></script>

<script src="./static/js/price_rangs.js"></script>

<!-- One Page, Animated-HeadLin -->

<script src="./static/js/wow.min.js"></script>

<script src="./static/js/animated.headline.js"></script>

<script src="./static/js/jquery.magnific-popup.js"></script>

<!-- Scrollup, nice-select, sticky -->

<script src="./static/js/jquery.scrollUp.min.js"></script>

<script src="./static/js/jquery.nice-select.min.js"></script>

<script src="./static/js/jquery.sticky.js"></script>

```
<!-- contact js -->
<script src="/static/js/contact.js"></script>
<script src="/static/js/jquery.form.js"></script>
<script src="/static/js/jquery.validate.min.js"></script>
<script src="/static/js/mail-script.js"></script>
<script src="/static/js/jquery.ajaxchimp.min.js"></script>
```

```
<!-- JQuery Plugins, main JQuery -->
<script src="/static/js/plugins.js"></script>
<script src="/static/js/main.js"></script>
<script>
window.watsonAssistantChatOptions = {
integrationID: "f98e678a-f49c-440c-813b-ee5e4d022946", // The ID of this
integration.
region: "au-syd", // The region your integration is hosted in.
serviceInstanceID: "83b70d1c-2dec-4ecb-b3da-c8b57383facf", // The ID of your
service instance.
onLoad: function(instance) { instance.render(); }
};
setTimeout(function(){
const t=document.createElement('script');
t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" +
(window.watsonAssistantChatOptions.clientVersion || 'latest') +
"/WatsonAssistantChatEntry.js";
document.head.appendChild(t);
});
</script>
<script>
```

```
function showResult(str) {  
    if (str.length==0) {  
        document.getElementById("livesearch").innerHTML="";  
        document.getElementById("livesearch").style.border="0px";  
        return;  
    }  
    var xmlhttp=new XMLHttpRequest();  
    xmlhttp.onreadystatechange=function() {  
        if (this.readyState==4 && this.status==200) {  
            document.getElementById("livesearch").innerHTML=this.responseText;  
            document.getElementById("livesearch").style.border="1px solid #A5ACB2";  
        }  
    }  
    xmlhttp.open("GET","livesearch.php?q="+str,true);  
    xmlhttp.send();  
}  
</script>
```

</body>

</html>

About .html

<!doctype html>

<html class="no-js" lang="zxx">

<head>

<meta charset="utf-8">

<meta http-equiv="x-ua-compatible" content="ie=edge">

<title>Skills and Job Recommender </title>

<meta name="description" content="">

<meta name="viewport" content="width=device-width, initial-scale=1">

```
<link rel="manifest" href="site.webmanifest">
<link rel="shortcut icon" type="image/x-icon" href="static/img/favicon.ico">
```

```
<!-- CSS here -->
```

```
<link rel="stylesheet" href="static/css/bootstrap.min.css">
<link rel="stylesheet" href="static/css/owl.carousel.min.css">
<link rel="stylesheet" href="static/css/flaticon.css">
<link rel="stylesheet" href="static/css/price_rangs.css">
<link rel="stylesheet" href="static/css/slicknav.css">
<link rel="stylesheet" href="static/css/animate.min.css">
<link rel="stylesheet" href="static/css/magnific-popup.css">
<link rel="stylesheet" href="static/css/fontawesome-all.min.css">
<link rel="stylesheet" href="static/css/themify-icons.css">
<link rel="stylesheet" href="static/css/slick.css">
<link rel="stylesheet" href="static/css/nice-select.css">
<link rel="stylesheet" href="static/css/style.css">
</head>
```

```
<body>
```

```
<header>
```

```
<!-- Header Start -->
```

```
<div class="header-area header-transparent">
```

```
<div class="header-top header-sticky">
```

```
<div class="container">
```

```
<div class="row align-items-center">
```

```
<div class="col-lg-3 col-md-2">
```

```
<!-- Logo -->
```

```
<div class="logo">
```

```
<a href="index"></a>
```

```
</div>
</div>
<div class="col-lg-9 col-md-9">
<div class="menu-wrapper">
<!-- Main-menu -->
<div class="main-menu">
<nav class="d-none d-lg-block">
<ul id="navigation">
<li><a href="index">Home</a></li>
<li><a href="job_listing">Find a Jobs </a></li>
<li><a href="about">About</a></li>
<li><a href="contact">Contact</a></li>
</ul>
</nav>
</div>
<!-- Header-btn -->
<div class="header-btn d-none f-right d-lg-block">
<a href="#" class="btn head-btn1">Register</a>
<a href="#" class="btn head-btn2">Login</a>
</div>
</div>
</div>
<!-- Mobile Menu -->
<div class="col-12">
<div class="mobile_menu d-block d-lg-none"></div>
</div>
</div>
</div>
</div>
```

```
</div>
<!-- Header End -->
</header>
<main>

<!-- Hero Area Start-->
<div class="slider-area ">
  <div class="single-slider section-overly slider-height2 d-flex align-items-center"
    data-background="assets/img/hero/about.jpg">
    <div class="container">
      <div class="row">
        <div class="col-xl-12">
          <div class="hero-cap text-center">
            <h2>About us</h2>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
<!-- Hero Area End -->

<!-- How Apply Process Start-->
<div class="apply-process-area apply-bg pt-150 pb-150" data-
  background="assets/img/gallery/how-applybg.png">
  <div class="container">
    <!-- Section Tittle -->
    <div class="row">
      <div class="col-lg-12">
        <div class="section-tittle white-text text-center">
```

```
<span>Apply process</span>
<h2> How it works</h2>
</div>
</div>
</div>
<!-- Apply Process Caption -->
<div class="row">
<div class="col-lg-4 col-md-6">
<div class="single-process text-center mb-30">
<div class="process-ion">
<span class="flaticon-search"></span>
</div>
<div class="process-cap">
<h5>1. Search a job</h5>
</div>
</div>
</div>
<div class="col-lg-4 col-md-6">
<div class="single-process text-center mb-30">
<div class="process-ion">
<span class="flaticon-curriculum-vitae"></span>
</div>
<div class="process-cap">
<h5>2. Apply for job</h5>
</div>
</div>
</div>
<div class="col-lg-4 col-md-6">
<div class="single-process text-center mb-30">
```



```
<div class="process-ion">
<span class="flaticon-tour"></span>
</div>
```

```
<div class="process-cap">
<h5>3. Get your job</h5>
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- How Apply Process End-->
```

```
<!-- Online CV Area Start -->
```

```
<div class="online-cv cv-bg section-overly pt-90 pb-120" data-
background="assets/img/gallery/cv_bg.jpg">
```

```
<div class="container">
```

```
<div class="row justify-content-center">
```

```
<div class="col-xl-10">
```

```
<div class="cv-caption text-center">
```

```
<p class="pera1">FEATURED TOURS Packages</p>
```

```
<p class="pera2"> Make a Difference with Your Online Resume!</p>
```

```
<a href="#" class="border-btn2 border-btn4">Upload your cv</a>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- Online CV Area End-->
```

```
</main>
<footer>
<!-- Footer Start-->
<div class="footer-area footer-bg footer-padding">
<div class="container">
<div class="row d-flex justify-content-between">
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-6">
<div class="single-footer-caption mb-50">
<div class="single-footer-caption mb-30">
<div class="footer-tittle">
<h4>About Us</h4>
<div class="footer-pera">
<p>Skills and Job Recommendation</p>
</div>
</div>
</div>
</div>

</div>
</div>
<div class="col-xl-3 col-lg-3 col-md-4 col-sm-5">
<div class="single-footer-caption mb-50">
<div class="footer-tittle">
<h4>Contact Info</h4>
<ul>
<li>
<a href="">
```

Anna University Regional Campus,

Madurai

Phone : +91 9876543210

Email : aurcm@gmail.com

</div>

</div>

</div>

<div class="col-xl-3 col-lg-3 col-md-4 col-sm-5">

<div class="single-footer-caption mb-50">

<div class="footer-tittle">

<h4>Important Link</h4>

Home

Find a Job

Page

Contact us

Support

</div>

</div>

</div>

<div class="col-xl-3 col-lg-3 col-md-4 col-sm-5">

<div class="single-footer-caption mb-50">

<div class="footer-tittle">

<h4>E-mail</h4>

<!-- Form -->

```
<div class="footer-form" >
<div id="mc_embed_signup">
<form target="_blank" action="https://spondonit.us12.list-
manage.com/subscribe/post?u=1462626880ade1ac87bd9c93a&id=92a4423d0
1"
method="get" class="subscribe_form relative mail_part">
<input type="email" name="email" id="newsletter-form-email"
placeholder="Email Address"
class="placeholder hide-on-focus" onfocus="this.placeholder = ""
onblur="this.placeholder = ' Email Address '">
<div class="form-icon">
<button type="submit" name="submit" id="newsletter-submit"
class="email_icon newsletter-submit button-contactForm"></button>
</div>
<div class="mt-10 info"></div>
</form>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- footer-bottom area -->
<div class="footer-bottom-area footer-bg">
<div class="container">
<div class="footer-border">
```

```
<div class="row d-flex justify-content-between align-items-center">
<div class="col-xl-2 col-lg-2">
<div class="footer-social f-right">
<a href="#"><i class="fab fa-facebook-f"></i></a>
<a href="#"><i class="fab fa-twitter"></i></a>
<a href="#"><i class="fas fa-globe"></i></a>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- Footer End-->
</footer>
```

```
<!-- JS here -->
```

```
<!-- All JS Custom Plugins Link Here here -->
<script src="./assets/js/vendor/modernizr-3.5.0.min.js"></script>
<!-- JQuery, Popper, Bootstrap -->
<script src="./assets/js/vendor/jquery-1.12.4.min.js"></script>
<script src="./assets/js/popper.min.js"></script>
<script src="./assets/js/bootstrap.min.js"></script>
<!-- JQuery Mobile Menu -->
<script src="./assets/js/jquery.slicknav.min.js"></script>

<!-- JQuery Slick , Owl-Carousel Plugins -->
<script src="./assets/js/owl.carousel.min.js"></script>
<script src="./assets/js/slick.min.js"></script>
```

```
<script src="./assets/js/price_rangs.js"></script>
<!-- One Page, Animated-HeadLin -->
<script src="./assets/js/wow.min.js"></script>
<script src="./assets/js/animated.headline.js"></script>
<script src="./assets/js/jquery.magnific-popup.js"></script>

<!-- Scrollup, nice-select, sticky -->
<script src="./assets/js/jquery.scrollUp.min.js"></script>
<script src="./assets/js/jquery.nice-select.min.js"></script>
<script src="./assets/js/jquery.sticky.js"></script>

<!-- contact js -->
<script src="./assets/js/contact.js"></script>
<script src="./assets/js/jquery.form.js"></script>
<script src="./assets/js/jquery.validate.min.js"></script>
<script src="./assets/js/mail-script.js"></script>
<script src="./assets/js/jquery.ajaxchimp.min.js"></script>

<!-- Jquery Plugins, main Jquery -->
<script src="./assets/js/plugins.js"></script>
<script src="./assets/js/main.js"></script>

</body>
</html>
```

Github Link:<https://BM-EPBL/IBM-Project-6621-1658833815>

Project Demo Link:

<https://www.dropbox.com/s/8y8z8flc4f9j0pt/DemoLINK.mp4?dl=0>

