Skill / Job Recommender Application

Team ID	PNT2022TMID31116
Team member	BRINTHA X BALAJI C INDHRA P MAHENDIRAN K

Skill / Job Recommender Application:

1. INTRODUCTION
    a. Project Overview
    b. Purpose
2. LITERATURE SURVEY
    a. Existing problem
    b. References
    c. Problem StatementDefinition
3. IDEATION & PROPOSED SOLUTION
    a. Empathy Map Canvas
    a. Ideation & Brainstorming
    b. Proposed Solution
    c. Problem SolutionFit
1. REQUIREMENT ANALYSIS
    a. Functional Requirements
    b. Non-Functional Requirements

2. PROJECT DESIGN

    a. Data Flow Diagrams

    b. Solution & Technical Architecture

    c. User Stories

3. PROJECT PLANNING & SCHEDULING

    a. Sprint planning& Estimation

    b. Sprint DeliverySchedule

    c. Reports from JIRA

4. CODING & SOLUTIONING

    a. Feature 1

    b. Feature 2

    c. Database Schema(if Applicable)

5. TESTING

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 fresheror the skilled person can log in and find the jobs by using the search option or they can directly interact with the chatbot and get their dreamjob.

a. Project Overview

There has been a sudden boom in the technical industry and an increase in the number of good start-ups. Keeping track of various appropriate job openings in top industry names has become increasingly troublesome. This leads to deadlines and hence importantopportunities being missed.Through this research paper, the aim is to automate this process to eliminate this problem. To achievethis, IBM cloud services like db2, Watson assistant, cluster,Kubernetes have been used. A hybrid system of Content-Based Filtering and Collaborative Filteringis implemented to recommend these jobs. The intention is to aggregateand recommend appropriate jobs to job seekers, especially in

the engineering domain. The entire process of accessing numerous company websites hoping to find a relevant job opening listed on their career portals is simplified. The proposed recommendation system is tested on an array of test cases with a fully functioning user interface in the form of a web application. It has shown satisfactory results, outperforming the existing systems. It thus testifies to the agenda of quality over quantity.

b. Purpose

Withan increasing number of cash-rich, stable, and promisingtechnical companies/startups on the web which are in much demand right now, many candidates want to apply and work for these companies. They tend to miss out on these postings because there is an ocean of existing systems that list millions of jobs which are generallynot relevant at all to the users. There is an abundance of choices and not much streamlining. On the basis of the actual skills or interests of an

individual, job seekers often find themselves unable to find the appropriate employment for themselves. This system, therefore, approaches the idea from a data point of view, emphasizing more on the qualityof the data than the quantity.

2. LITERATURE SURVEY

a. Existing Problem

Existing system is not very efficient, it does not benefit the user in maximum way, so the proposed system uses ibm cloud services like db2, Watson virtual assistant, cluster, kubernetes and docker for containerization of the application.

i. In this section, we describe our framework for job recommendation. We narrow down the scope and focus on recommendation of job vacancies for Information Technology (IT) professionals acting in the Brazilian market.The proposed frameworkis composed by three stages:data collection, data preparation and recommendation.

ii. we select a group of the nearest job offers based on the distance to that profile (job matching). In the case of TF-IDFrepresentation, we use the cosinedistance while for word embeddings, we use the relatively new Word Mover'sDistance (WMD) [Kus15].Once retrieved the top "k" job offersfor the profile, we sort them in descending order based on the inverse of this distance (ranking)To perform job offers scraping, we created a list of keywords from the IT industry and used them as search terms. For each keyword, we search all the related job

offers using Catho's search engine and save the retrieved results in our database; thus, the content's quality is highly related to the quality of the Catho's search engine.

iii. we retrieved data from job search sites using only IT keywords, there were still some job offers that do not correspond to this field, then, the first step in this phase is filtering out job offers that do not belong to the IT field. To achieve this, we use a dictionary of weighted IT terms to match each job offer in its document-like format.

iv. Once job offers and profiles are filtered, the second step is text preprocessing. In this task, we perform stop words removal, tokenization and lemmatization for the Portuguese language.

v. The third step, feature representation, aims to represent these documents (job offers and profiles) as vector space models. For this purpose, we adopted two approaches: word embeddings and TF-IDF. The latter technique does not require so much effort to be implemented unlike the former.

b. References

1] Fabian Abel, András Benczúr, Daniel Kohlsdorf, Martha Larson, and Róbert

Pálovics. RecSys challenge 2016: Job recommendations. In Proceedings of the 10th ACM conference on Recommender Systems, pages 425–426, 2016.

[2] Fabian Abel, Yashar Deldjoo, Mehdi Elahi, and Daniel Kohlsdorf. RecSys challenge 2017: Offline and online evaluation. In Proceedings of the eleventh ACM Conference on Recommender Systems, pages 372–373, 2017.

[3] Charu C. Aggarwal. Recommender systems. Springer, 2016.

[4] Shibbir Ahmed, Mahamudul Hasan, Md. Nazmul Hoq, and Muhammad Abdullah Adnan. User interaction analysis to recommend suitable jobs in career-oriented social networking sites. In 2016 International Conference on Data and Software Engineering (ICoDSE), pages 1–6. IEEE, 2016.

[5] Shaha T. Al-Otaibi and Mourad Ykhlef. A survey of job recommender systems. International Journal of Physical Sciences, 7(29):5127–5142, 2012.

[6] Nikolaos D Almalis, George A Tsihrintzis, and Evangelos Kyritsis. A constraint-based job recommender system integrating FoDRA. International Journal of Computational Intelligence Studies, 7(2):103–123,2018.

[7] Dhruv Arya, Viet Ha-Thuc, and Shakti Sinha. Personalized federated search at linkedin. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pages 1699–1702, 2015.

[8] Jack Bandy. Problematic machine behavior: A systematic literature review of algorithm audits. Proceedings of the ACM on Human-Computer Interaction, 5(CSCW1):1–34, 2021.

[9] Shivam Bansal, Aman Srivastava, and Anuja Arora. Topic modeling driven content based jobs recommendation engine for recruitment industry. Procedia computer science, 122:865–872, 2017.

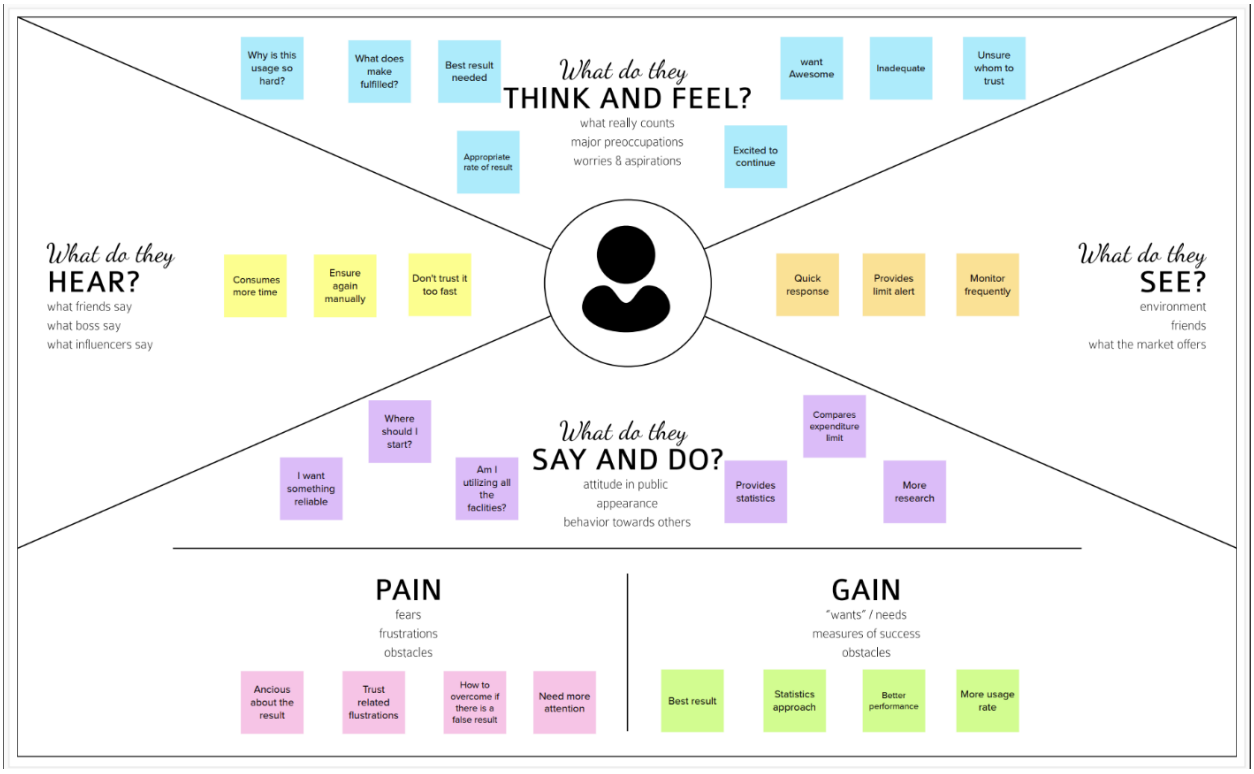
[10] Mathieu Bastian, Matthew Hayes, William Vaughan, Sam Shah, Peter Skomoroch, Hyungjin Kim,Sal Uryasev, and Christopher Lloyd. LinkedIn skills: large-scale topic extraction and inference. In
Proceedings of the 8th ACM Conference on Recommender systems, pages 1–8, 2014.

1.5 Problem Statement Definition

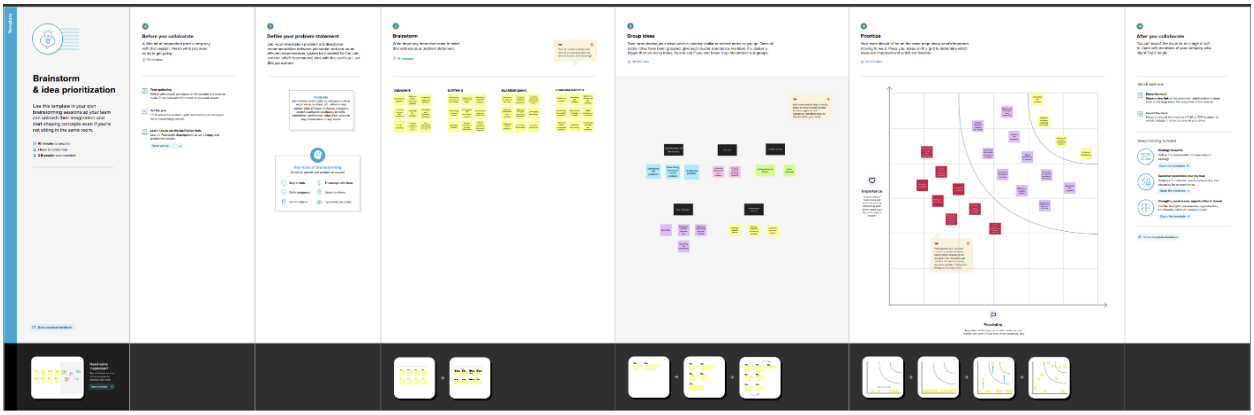
Problem Statement (PS)	I am (User)	I'm tryingto	But	Because	Which makesme feel
PS-1	A user who is unemployed	Get a job in the mentioned domain with a reasonable package	Many jobs are not matching with my skillset	Not sure what the companies are looking for in a candidate	Not really updated or qualified
PS-2	A user who is looking for a career change or field change	Explore various domains and find a good job	I am not fully aware of the various domains available.	There are not many platforms to get enough knowledge about the domains	Far back in my knowledge of the opportunities available with no exposure
PS-3	An employed person aiming for a higher position or post	Aiming for a better job or a job with higher package in a higher position.	I am over qualified for few jobs and under qualified for few jobs	The job opportunities available are not all of the same position or capacity	There is a very little chances for me to get promoted in a particular domain or place

2. IDEATION AND PROPOSEDSOLUTION

2.1 Empathy Map Canvas



3.1 Ideation &Brainstorming



3.2 Proposed Solution

1. PROBLEMSTATEMENT:

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.

2. IDEA/SOLUTIONDESCRIPTION:

The goal is to compile and propose suitable employment to job searchers, particularly in the engineering field. It is no longer necessary to visit various firm websites in search of an appropriate job listing on each one's career portal

3. NOVELTY/UNIQUENESS:

Based on the user's previous queries, it cannot provide anything unexpected. The system offers a wide range of applications that can be utilised to strengthen and perfect it. In other words, it is possible to do away with the onetime configuration step or process required to fetch employment from a specific new organisation.

4. SOCIAL IMAPACT/CUSTOMER SATISFACTIO:

Social networking sites are regarded by human resources professionals as a key tool for successfully and efficiently disseminating information about job searchers and posting job listings. In the information era, social networking sites are the most widely used and significant online communication platform where individuals may exchange private information and establish professional connections. Additionally, job seekers are perusing their social networking accounts in search of simple, affordable job postings

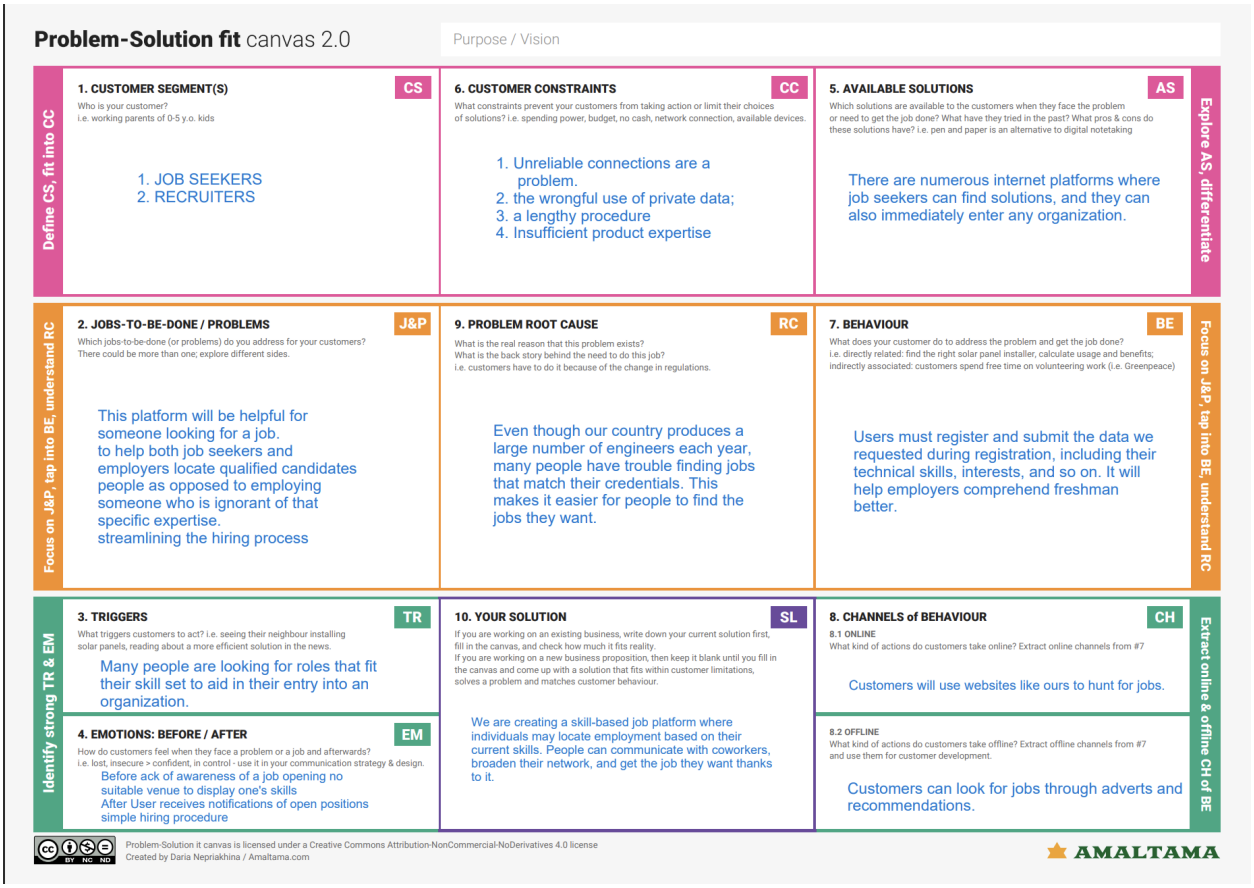
5. BUSINESS MODEL:

Networking sites are simple to use, provide access to a large number of job postings, and allow job seekers to avoid paying for printed newspapers, saving them money. The interview process is cut-short into efficient and effective method because of the online flow of the recruitment that saves the business entities of both recruiters and the candidate.

6. SCALABILITY OF SOLUTION:

Because of the objective of this application is to provide a platform to suggest offers so that the demand of this application will be flexible and scalable to the increasing social and business impact.

2.2 Problem Solution Fit



4. REQUIREMENT ANALYSIS

4.1Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration throughForm. Registration through Gmail. Registration through Application.
FR-2	User Confirmation	Confirmation via Email. Confirmation via OTP.
FR-3	User Login	Login usingcredentials.
FR-4	User Application	Search for desired company.
FR-5	User Profile	Complete user profile by providing personal details.
FR-6	User Application	User applies for the desiredcompany.

4.2Non-functional requirements:

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

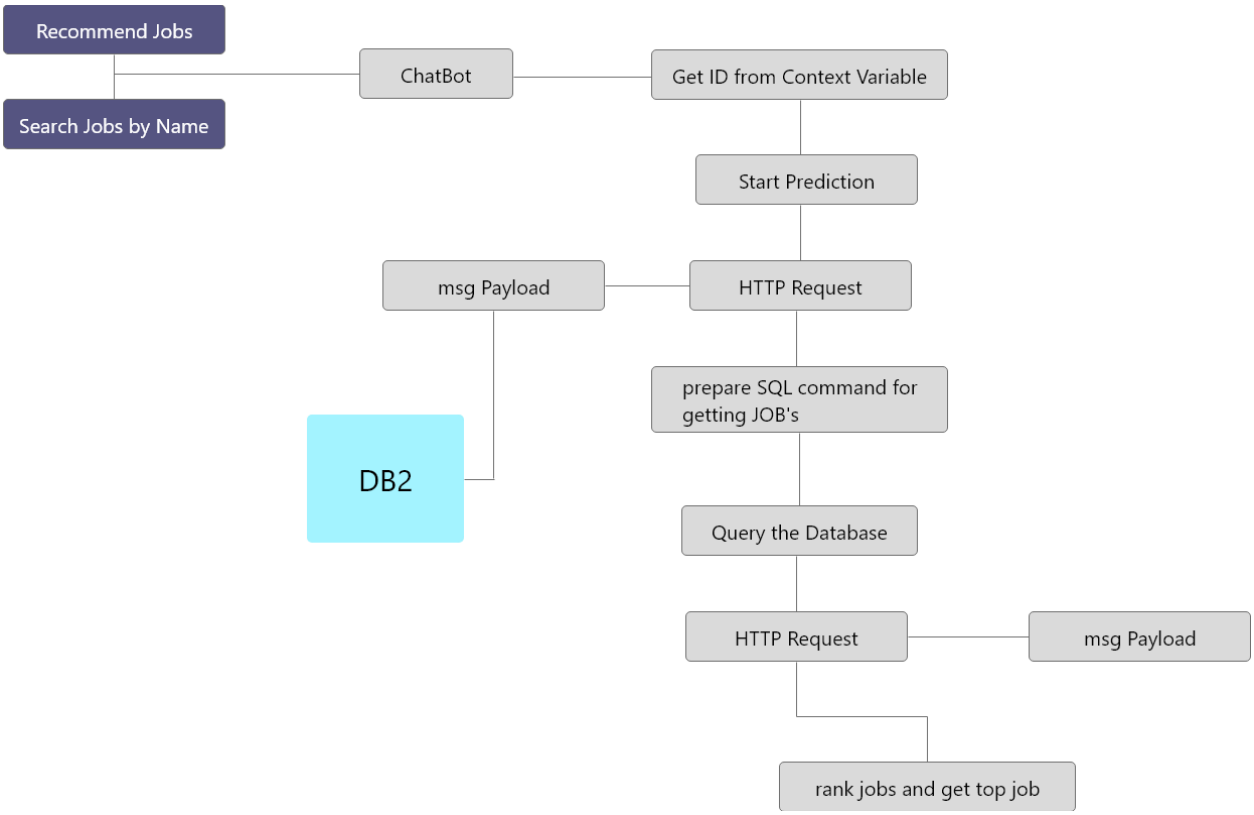
FR No.	**Non-Functional Requirement**	**Description**

NFR-1	Usability	1. User-Friendly Application.
NFR-2	Security	1. End-to-End Encryption.
NFR-3	Reliability	1. Based on personalised skillsets.
NFR-4	Performance	1. Analysing the skill sets of the user to ensureourrecommendations reachthem better.
NFR-5	Availability	1. 24/7 chatbot support ✓ 24/7 chatbot support.
NFR-6	Scalability	1. Reaching the on-scale requirement of the user.

5.PROJECT DESIGN

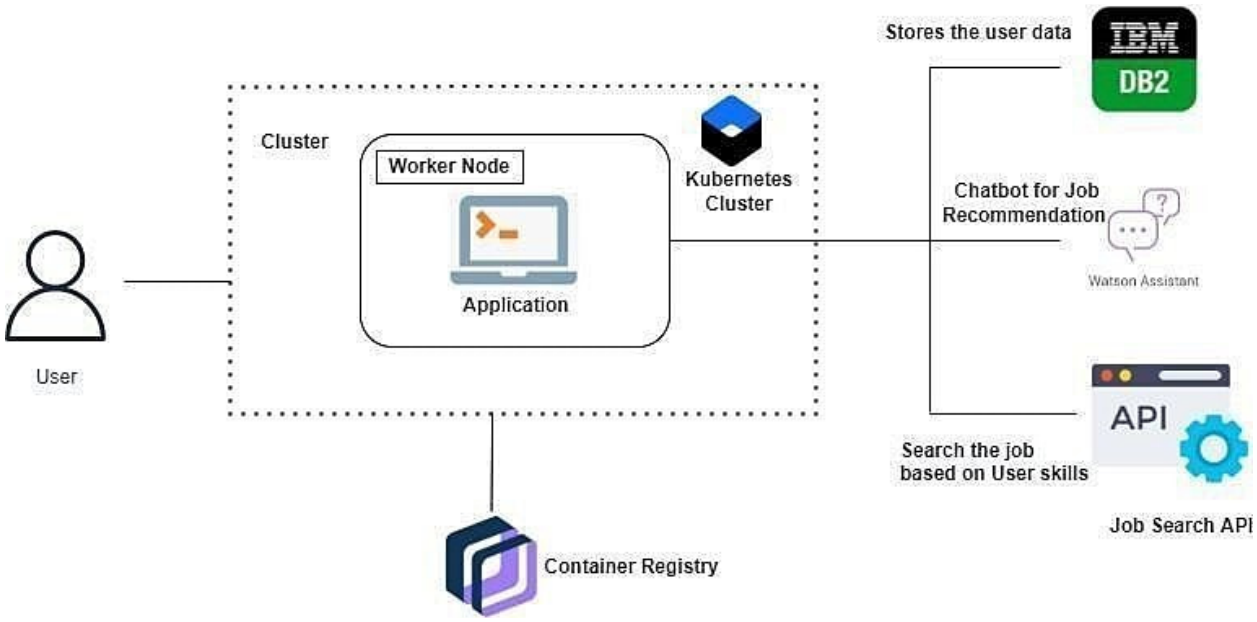
5.1Data flow diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation ofthe 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 Technology Architecture:

The deliverable shall include the architectural diagramas below



S. No	COMPONENT	DESCRIPTION	TECHNOLOGY
1.	User Interface	How user interacts with application e.g. Web UI,Mobile App, Chatbot etc.	HTML, CSS, JavaScript /Angular Js /ReactJ setc.
2.	Developing Interface	Developing application for thetask	Java /Python
3.	Voice Assistance	Voice commands instead of typing.	IBM Watson STT service
4.	Chatbot Assistance	Conversational Interface	IBM Watson Assistant
5.	Database	Data Type,Configurations etc.	MySQL, NoSQL,etc.

6.	Cloud Database	Database Serviceon Cloud	IBMDB2, IBM Cloudantetc.
7.	File Storage	File storage requirements	IBM Block Storage orOther Storage Service or LocalFile system
8.	Machine Learning Model	Purpose of Machine LearningModel	Object Recognition Model,etc.
9.	Infrastructure (Server/Cloud)	Application Deployment on LocalSystem /Cloud LocalServer Configuration: Cloud Server Configuration:	Local, Cloud Foundry,Kubernetes, etc.

5.3User Stories

Use the below templateto list all the user stories for the product.

User Type	Functional Requirement	User Story Number	User Story/ Task	Acceptancecriteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register forthe application by entering my email,password, andconfirming my password.	I can accessmy account / dashboard	High	Sprint1

		USN-2	As a user, I will receive confirmation email onceI have registered forthe application	I can receive confirmation email & click confirm	High	Sprint1
		USN-3	As a user, I can register for the application through Facebook	I can register & accessthe dashboard with Facebook Login	Low	Sprint2
		USN-4	As a user,I can register for the application through Gmail	I can receive confirmation email & click confirm	Mediu m	Sprint1
	Login	USN-5	As a user,I can log into the application by entering email & password	I can access my account / dashboard	High	Sprint1

	Dashboard	USN-6	Create a modelset that contains those models, then assign it to a role.	Assign that groupto the appropriate roles on the Roles page.	High	Sprint1

Customer(Web user)	Identity-Aware	USN-7	Open, public access, User-authenticated access, Employeerestricted access.	Company public website. App running on thecompany intranet. App with access tocustomer private information.	High	Sprint1
CustomerCare Executive	Communication	USN-8	A customer careexecutiveis a professionalresponsible for communicating the how's and why's regarding serviceexpectations within a company.	For how totackle customer queries.	Medium	Sprint1

Administrator	Device management	USN-9	You can Delete/Disable/Enable devices in Azure Active Directory but you cannot Add/Remove Usersin the directory.	Ease of use.	Medium	Sprint1

6. PROJECT PLANNINGAND SCHEDULING

6.1 Sprint Planning and Examination

Title	Description
Literature Survey and Information Gathering	Gathering Information by referring the technical papers, research publications etc
Prepare Empathy Map	To capture user pain and gainsPrepare List of Problem Statement
Ideation	Prioritise a top 3 ideas basedon feasibility and Importance
Proposed Solution	Solution includenovelty, feasibility, business model, social impact and scalability of solution
Problem Solution Fit	Solution fit document
Solution Architecture	Solution Architecture
Customer Journey	To Understand User
	Interactions and experiences withapplication

Functional Requirement	Prepare functional Requirement
Data flow Diagrams	Data flowdiagram
Technology Architecture	Technology Architecture diagram
Milestone &sprint delivery plan	Activity what we done&further plans
Project Development Delivery of sprint 1,2,3 & 4	Develop and submitthe developed codeby testing it

6.2 Sprint deliveryschedule

SPRINT	TASK	
SPRINT 1	Create Registration page , login page , Jobsearch portal, job apply portal in flask	
SPRINT 2	Connect application to ibm db2	
SPRINT 3	Integrate ibm Watson assisstant	
SPRINT 4	Containerize the app and deploytheapplication in ibm cloud	

6.3 Reports from JIRA:

Average Age Report.
Created vs Resolved IssuesReport
Pie Chart Report.
Recently CreatedIssues Report
Resolution Time
Report.
Single Level
Group by
Report.Time
Since Issues
Report.
Time Tracking Report.

7.CODING & SOLUTIONING

```
from flask import Flask , render_template , request, session
import ibm_db
import re

app = Flask(_name_)

app.secret_key = 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=6667d8e9-9d4d-4ccb-ba32-
21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=SSL;SSL
ServiceCertificate=DigiCertGlobalRootCA.crt;UID=wmx93883;PWD=uQM2V5K7w8G0j4IK",'','')
@app.route('/login',methods=['GET','POST'])
def login():
    global userid
    msg="  "


    if request.method == 'POST' :
        name = request.form['name']
```

```
        password = request.form['password']
        sql = "SELECT * FROM USER WHERE name =? AND password=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,name)
        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['name']
            userid=  account['name']
            session['name'] = account['name']
            msg = 'Logged in successfully !'
            return render_template('welcome.html', msg = msg)
        else:
            msg = 'Incorrect name / password !'
    return render_template('login.html', msg = msg)




@app.route('/', methods =['GET', 'POST'])
def register():
    msg ="" "
    if request.method == 'POST' :
        name = request.form['name']
        email = request.form['email']
        password = request.form['password']
        rollno = request.form['rollno']
        sql = "SELECT * FROM USER 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)
        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]+', name):
```

```
            msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO  USER VALUES (?, ?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, name)
            ibm_db.bind_param(prep_stmt, 2, email)
            ibm_db.bind_param(prep_stmt, 3, rollno)
            ibm_db.bind_param(prep_stmt, 4, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
            return render_template('login.html',msg=msg)
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg = msg)


if _name_ == '_main_':
    app.run()
```

IBM Watson(ChatBot Service):

```
<script>
    window.watsonAssistantChatOptions = {
      integrationID: "ff946d70-3717-4f9b-ab79-a23fe39326b9", // The ID of this integration.
      region: "us-south", // The region your integration is hosted in.
      serviceInstanceID: "ae4661b5-f345-47b1-a3cc-89cc1561a5bf", // 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>
```

8. TESTING

8.1 Test Cases:

We tested for various validations. Tested all the features with using all the functionalities. Tested the data base storage and retrieval feature too.

Testing was done in phase 1 and phase 2, where issues found in phase1 were fixed andthen tested again in phase2.

8.2 User Acceptance Testing:

Real world testing was also done, by giving to remote users and asking them to use the application. Their difficulties were fixed and tested again until all the issueswere

.

9. ADVANTAGES AND DISAVANTAGES

ADVANTAGE :

1. Ithelps candidates to search the job which perfectly suitesthem and makethem aware of all the job openings.
2. Ithelp recruiters of the company to choose the right candidates for theirorganizations with appropriate skills.
3. Since itis cloud application, it does requireany installation of software andis portable.

DISADVANTAGE:

4. It is costly.
5. Uninterrupted internetconnection is requiredfor smooth functioning ofapplication

11.CONCLUSION

we have used ibm cloud services like db2, cloud registry, kubernetes , Watson assistant to create this application , which will be very useful for candidates who are searching for job and as well as for the

company to select the right candidate for their organization.

12.FUTURE SCOPE

Future directions of our work will focus on performing a more exhaustive evaluation considering a greater amount of methodsand data as well as a comprehensive evaluation of the impact of each professional skill of a job seeker on the received job recommendation. We can use machine learning techniques to recommend data in a efficientway.