IBM – NALAIYA THIRAN PROJECT

SKILL JOB RECOMMENDER APPLICATION

INDUSTRY MENTOR	**:**	**KRISHNA CHAITANYA**
FACULTY MENTOR	**:**	**ROHINI M**

TEAM ID	**:**	PNT2022TMID02696
TEAM LEAD	**:**	DHIKSHA S
TEAM MEMBER	**:**	ELAMMATHI M
TEAM MEMBER	**:**	ADAIKALA PITCHAI THABITHA J I
TEAM MEMBER	**:**	EMMA THOMAS

ABSTRACT

In the last years, job recommender systemshave become popularsince they successfully reduce information overload by generating personalized job suggestions. Although in the literature exists a variety of techniques and strategies used as part of job recommender systems, most of them fail to recommending job vacancies that fit properly to the job seekers profiles. Thus, the contributions of this work are threefold, we: i) made publicly available a new dataset formed by a set of job seekers profilesand a set of job vacancies collectedfrom different job search enginesites; ii) put forward the proposal of a framework for job recommendation based on professional skills of job seekers; and iii) carriedout an evaluation to quantifyempirically the recommendation abilities of two state-of-the-art methods, considering different configurations, within the proposed framework. We thus present a general panorama of job recommendation task aiming to facilitate research and real-world application design regarding this important issue.

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

TABLE OF CONTENT

1. INTRODUCTION

1.1 PROJECT OVERVIEW:

There has been a sudden boom in the technical industry and an increase in the number of good startups. Keeping track of various appropriate job openings in top industry names has become increasingly troublesome. This leads to deadlines and hence important opportunities being missed. Through this research paper, the aim is to automate this process to eliminate this problem. To achieve this, IBM cloud services like db2, Watson assistant , cluster, kubernetes have been used. A hybrid system of Content-Based Filtering and Collaborative Filtering is 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 qualityover quantity

1.2 PURPOSE:

With an increasing numberof cash-rich, stable,and promising technicalcompanies/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 millionsof jobs which are generally not 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 quality of the data than the quantity.

2.LITERATURE SURVEY

2.1EXISTING PROBLEM:

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

They tend to miss out on these postings because there is an ocean of existing systems that list millions of jobs which are generally not relevant at all to the users. There is an abundance of choices and not much streamlining. On the basis of the actual skillsor 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 quality of the data than the quantity.

2.2 REFERENCES:

i. Shaha T Al-Otaibi and Mourad Ykhlef. "A survey of job recommender systems". In: International Journal of the Physical Sciences7.29 (2012), pp. 5127–5142. issn: 19921950. doi: 10.5897/IJPS12. 482

ii. N Deniz, A Noyan, and O G Ertosun. "Linking Person-job Fit to

Job Stress: The Mediating Effect of Perceived Person-organization Fit". In: Procedia - Social and Behavioral Sciences 207 (2015), pp. 369– 376.

iii. M Diaby, E Viennet, and T Launay. "Toward the next generation of recruitment tools: An online social network-based job recommender system". In: Proc. of the 2013 IEEE/ACMInt. Conf. on Advances in Social NetworksAnalysis and Mining,ASONAM 2013 (2013), pp. 821–828. doi: 10. 1145/2492517.2500266.

iv. M Diaby and E Viennet. "Taxonomy-based job recommender systemson Facebook and LinkedIn profiles". In: Proc. of Int. Conf. on Research Challenges in Information Science(2014), pp. 1–6. issn: 21511357. doi: 10.1109/RCIS.2014.6861048.

v. M Kusner et al. "From word embeddings to document distances". In: Proc. of the 32nd Int. Conf. on Machine Learning, ICML'15. 2015, pp. 957–966.

vi. T Mikolovet al. "Distributed Representations of Words and Phrases and Their Compositionality". In: Proc. of the 26th Int. Conf. on Neural Information Processing Systems - Volume 2. NIPS'13. Lake Tahoe, Nevada,2013, pp. 3111–3119. url: http://dl.acm.org/citation.cfm?id=2999792. 2999959.

vii. T Mikolov et al. "Efficient estimation of word representations in vector space".In: arXivpreprintarXiv:1301.3781 (2013).

viii. G Salton and C Buckley. "Term-weighting approaches in automatic text retrieval". In: Information Processing and

Management 24.5 (1988), pp. 513–523. issn: 0306-4573. doi: https://doi.org/10. 1016/0306- 4573(88)90021- 0. url: http://www.sciencedirect.com/science/article/pii/ 030645738890021

2.3 PROBLEM STATEMENT DEFINITION:

"Can an efficient recommender system be modeled for the Job seekers which recommend Jobswith the user’s skill setand job domain and also addresses the issue of cold start?".

In current situation recruitment s done manually for lakhs of students in which many talented students may lose their opportunities due to different reasons since it is done manually, and company also need the highly talented people from the mass group for their growth. So we have build a cloud application to do this process in a efficient manner.

3. IDEATION & PROPOSED SOLUTION

In this project you will be working on two modules :

1. Admin and
2. User

ADMIN:

The role of the admin is to check out the database about the stock and have a track of all thethings that theusers are purchasing.

USER :

The user will login into the website and gothrough the products available on the website. Instead of navigating to several screens the user can directly talk to Chatbot. Getthe recommendations based on information provided by the user.

FEATURES OF CHATBOT :

- Using chatbot we can manageuser's choices and orders.
- The chatbot can give recommendations to the users based ontheir interests.
- It can promotethe best deals and offers on that day.
- It will storethe customer's details and orders in the database.
- The chatbot will send a notification to customers if the order is confirmed.
- Chatbots can also help in collecting customerfeedback.

3.1 EMPATHY MAP CANVAS:

An empathy map is a collaborative visualization used to articulate what we know about aparticular type of user. It externalizes knowledge about users in order to

1. create a shared understanding of user needs, and
2. aid in decision making

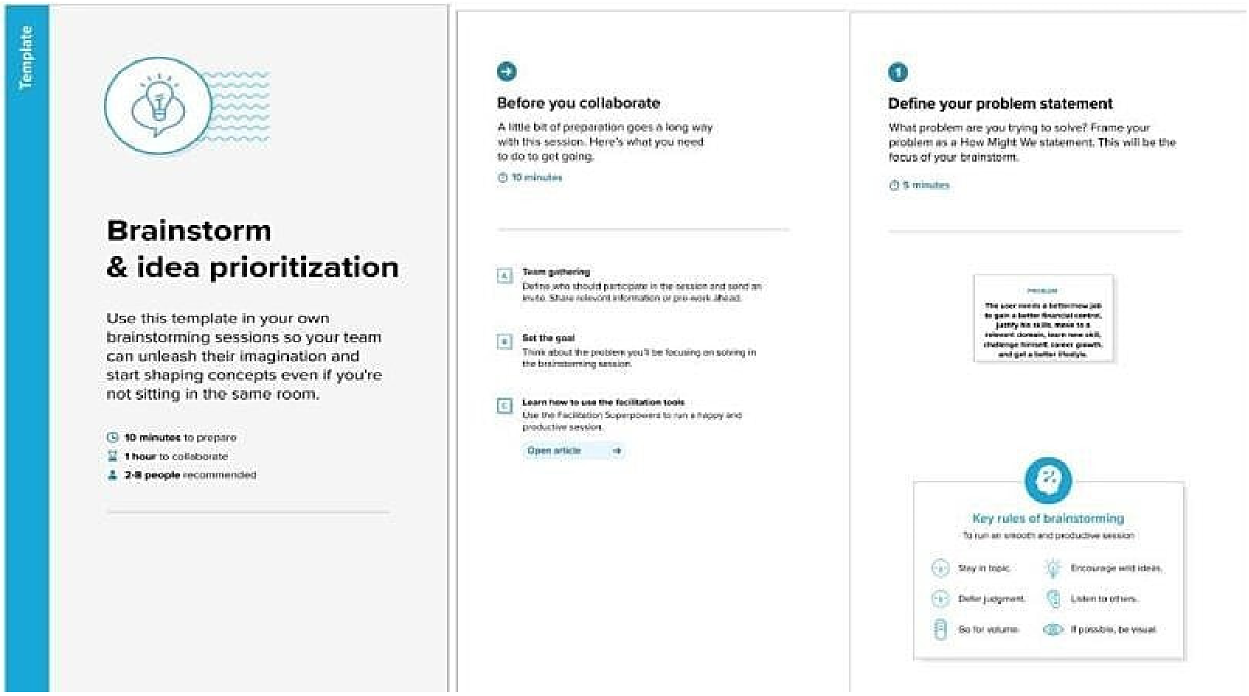
3.2 IDEATION & BRAINSTROMING:

Brainstorm & Idea Prioritization Template:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions. Use this template in yourown brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in thesame room.

STEP 1:

Team Gathering, Collaboration and Select the Problem Statement



Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- **10 minutes** to prepare
- **1 hour** to collaborate
- **2-8 people** recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1 **Define your problem statement**

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

The user needs a better/new job to gain a better financial control, justify his skills, move to a relevant domain, learn new skill, challenge himself, career growth, and get a better lifestyle.

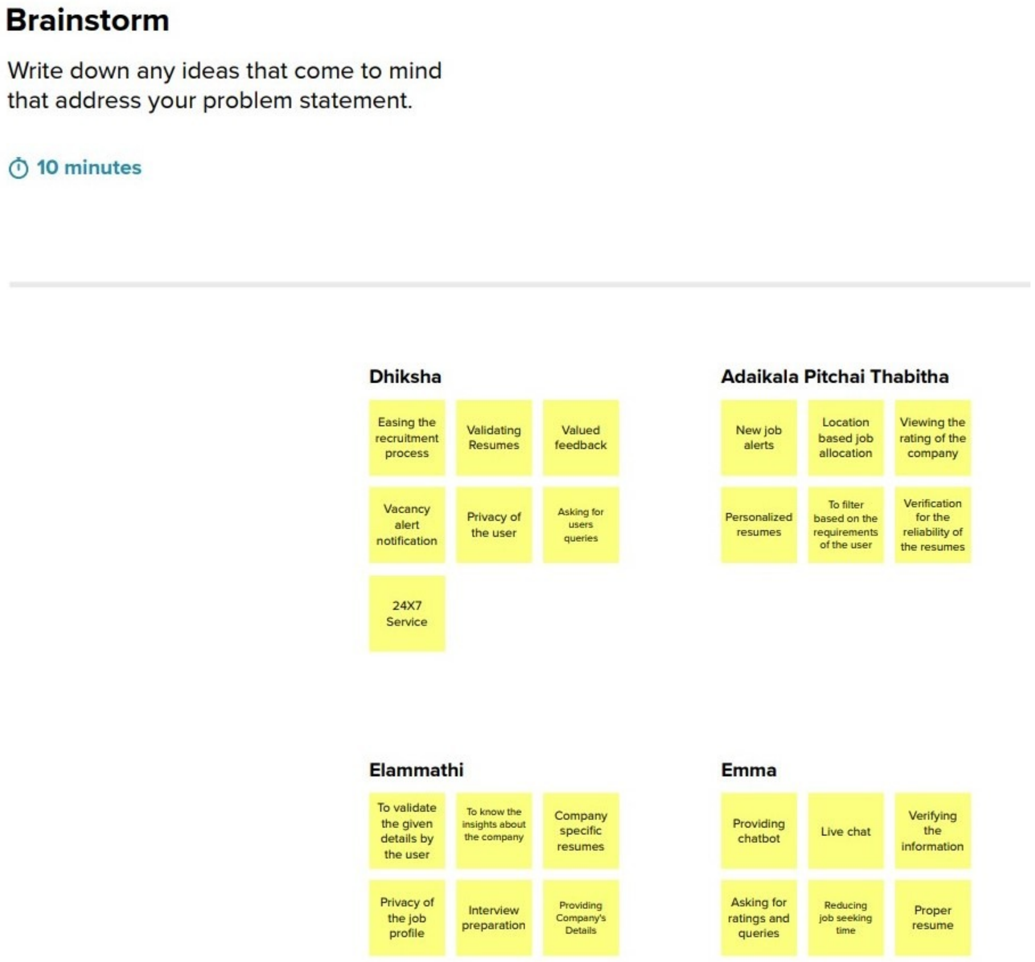
Key rules of brainstorming

To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

STEP 2:

Brainstorm, Idea Listingand Grouping



Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Dhiksha

- Easing the recruitment process
- Validating Resumes
- Valued feedback
- Vacancy alert notification
- Privacy of the user
- Asking for users queries
- 24X7 Service

Adaikala Pitchai Thabitha

- New job alerts
- Location based job allocation
- Viewing the rating of the company
- Personalized resumes
- To filter based on the requirements of the user
- Verification for the reliability of the resumes

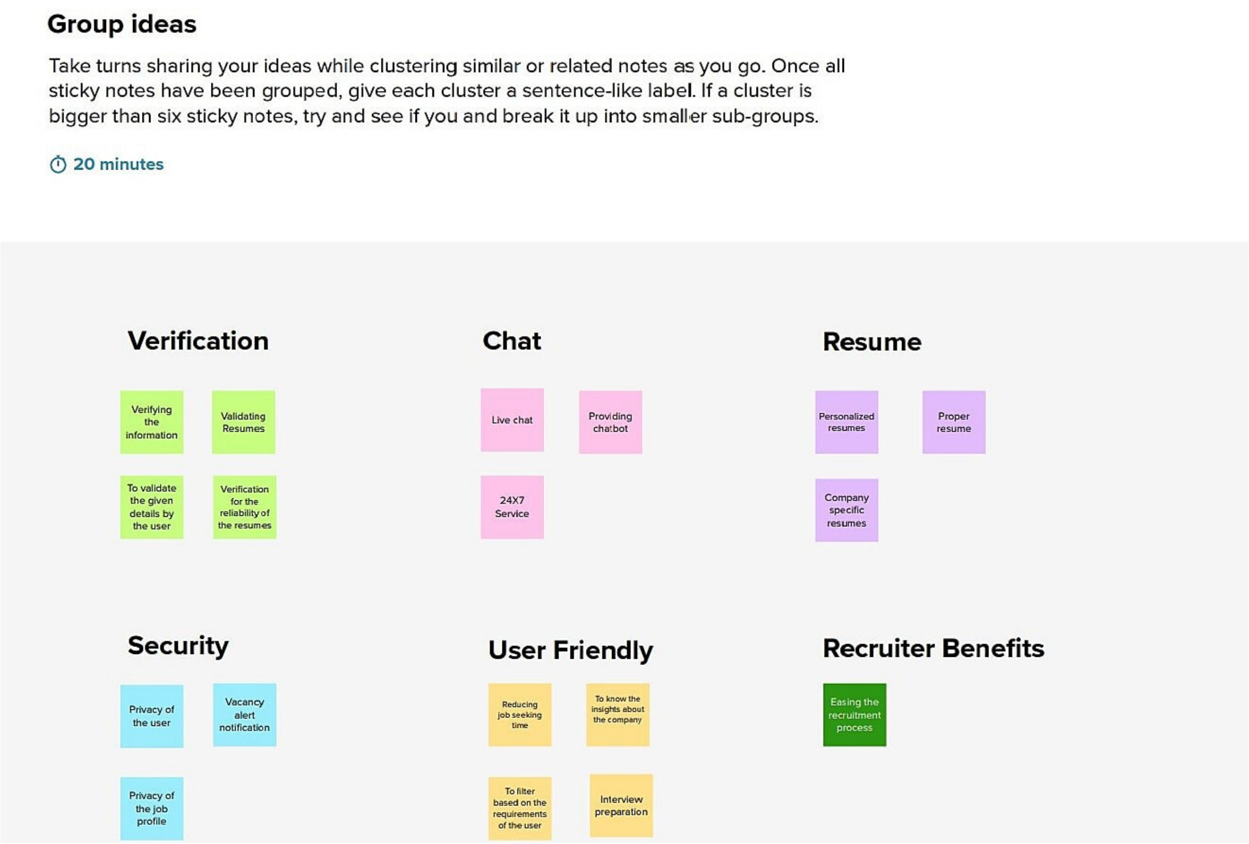
Elammathi

- To validate the given details by the user
- To know the insights about the company
- Company specific resumes
- Privacy of the job profile
- Interview preparation
- Providing Company's Details

Emma

- Providing chatbot
- Live chat
- Verifying the information
- Asking for ratings and queries
- Reducing job seeking time
- Proper resume

PNT2022TMID02696



Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Verification

- Verifying the information
- Validating Resumes
- To validate the given details by the user
- Verification for the reliability of the resumes

Chat

- Live chat
- Providing chatbot
- 24X7 Service

Resume

- Personalized resumes
- Proper resume
- Company specific resumes

Security

- Privacy of the user
- Vacancy alert notification
- Privacy of the job profile

User Friendly

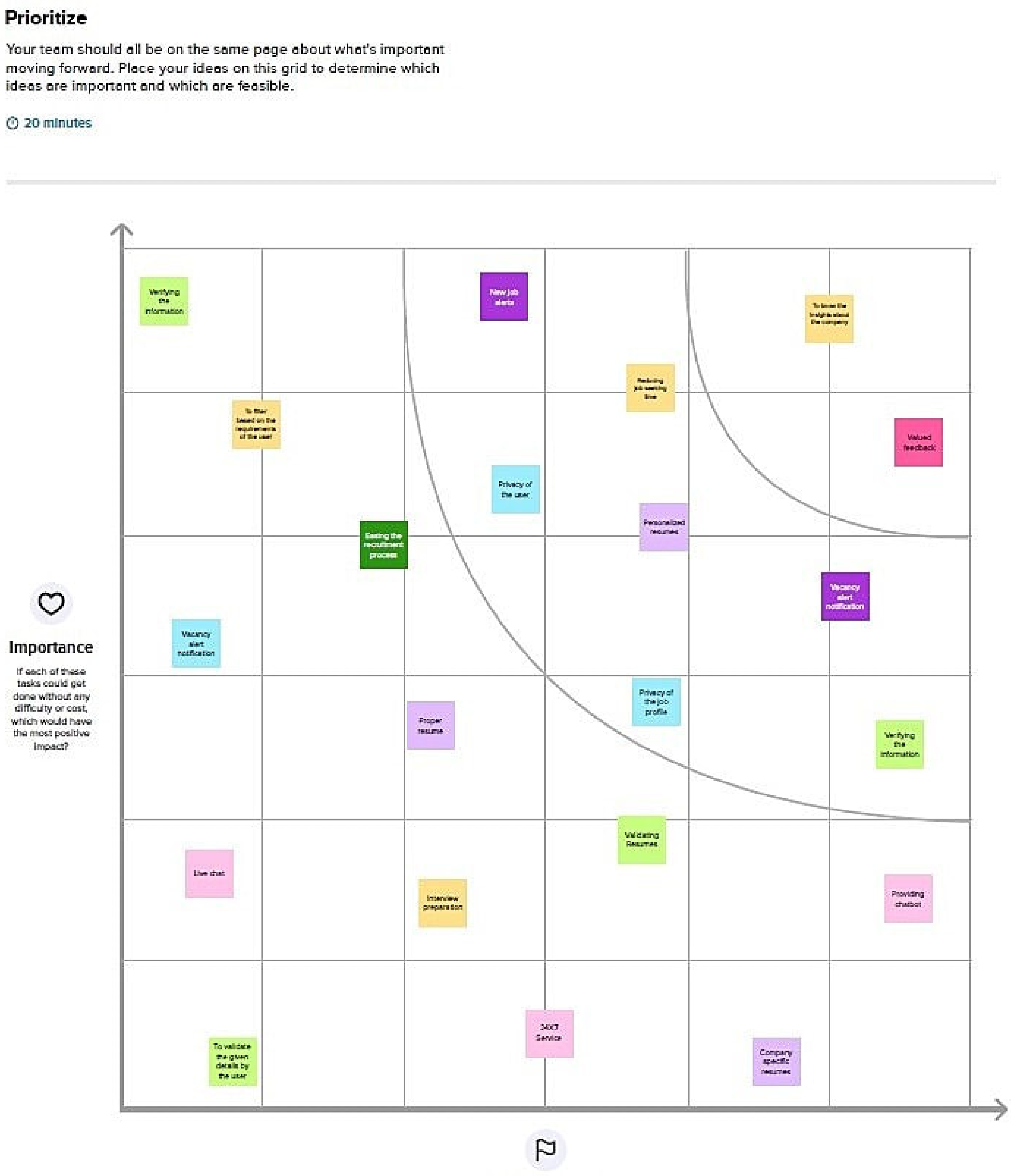
- Reducing job seeking time
- To know the insights about the company
- To filter based on the requirements of the user
- Interview preparation

Recruiter Benefits

- Easing the recruitment process

STEP 3:

Idea Prioritization



Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Verifying the information
New job alerts
To know the insights about the company
Reducing job seeking time
To filter based on the requirements of the user
Valued feedback
Privacy of the user
Easing the recruitment process
Personalized resumes
Vacancy alert notification
Vacancy alert notification
Proper resume
Privacy of the job profile
Verifying the information
Validating Resumes
Live chat
Interview preparation
Providing chatbot
24X7 Service
To validate the given details by the user
Company specific resumes

3.3 PROPOSED SOLUTION:

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

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

3.4 PROBLEM SOLUTION FIT:

Problem-Solution

Project Title: Skill / Job Recommender Application **Team ID:** PNT2022TMID02696

Define CS, fit into CC

1.CUSTOMER SEGMENT(S)

- Job finder who are looking for a suitable job for themselves

6. CUSTOMER CONSTRAINTS

- Not knowing the awareness of job vacancy
- Afraid of falling into the scam

5. AVAILABLE SOLUTIONS

- There are already application like LinkedIn. Naukri etc
- They often get notified by the job openings

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS

Job Seeker:

- They should be ready with the communication skills
- They need to gain knowledge before applying a particular job.

Job Recruiter:

- They have to filter the candidate who is best suited for the company

9. PROBLEM ROOT CAUSE

- As the time goes by, the competition keeps on growing day by day
- The education system failed to create qualified engineers

7. BEHAVIOUR

- Learn a particular skillset for a particular job openings.
- Approach multiple people and gain knowledge from them

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS

- Family crisis
- Social Pressure
- Hating the job

4. EMOTIONS: BEFORE / AFTER BEFORE

- Depression and lack of confidence
- Fear of rejection

AFTER

- Highly excited
- Gained confidence to do any task

10. YOUR SOLUTION

- A Scam offer about any job openings are detected
- The Client will be notified immediately
- Proper resources will be provided

8. CHANNELS OF BEHAVIOUR ONLINE

- Google search the opportunities
- Make connection in LinkedIn

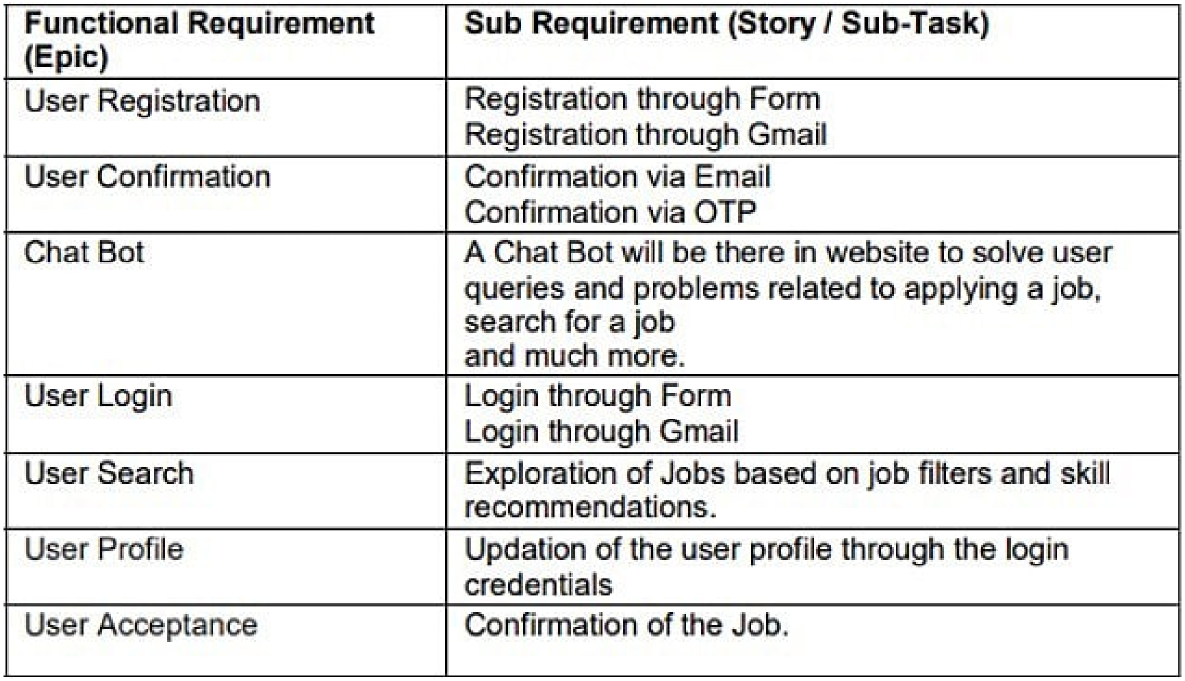
OFFLINE

- Try to get more professional training from the trainers

Extract online & offline CH of BE

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT:



Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
User Registration	Registration through Form Registration through Gmail
User Confirmation	Confirmation via Email Confirmation via OTP
Chat Bot	A Chat Bot will be there in website to solve user queries and problems related to applying a job, search for a job and much more.
User Login	Login through Form Login through Gmail
User Search	Exploration of Jobs based on job filters and skill recommendations.
User Profile	Updation of the user profile through the login credentials
User Acceptance	Confirmation of the Job.

4.2 NON-FUNCTIONAL REQUIREMENTS:

Non-functional Requirements:

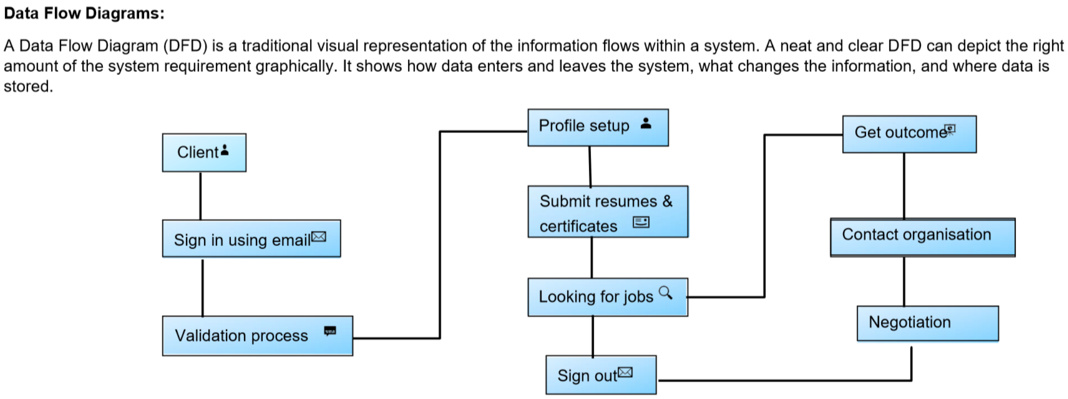
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Filtering based on the skillset of the user
NFR-2	**Security**	Two step authentication
NFR-3	**Reliability**	User data is preserved
NFR-4	**Performance**	Load balancing
NFR-5	**Availability**	Irrespective of the time, either company can put job opening or the seeker can apply for jobs.
NFR-6	**Scalability**	User friendly application

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 entersand leaves the system, what changes the information, andwhere data is stored.



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 & TECHNICAL ARCHITECTURE:

Solution architecture is a complex process – with many sub-processes – that bridges the gapbetween business problems and technology solutions. Its goals are to:

i. Find the best tech solution to solve existing business problems.

ii. Describe the structure, characteristics, behaviour, and other aspects of the softwaretoproject stakeholders.

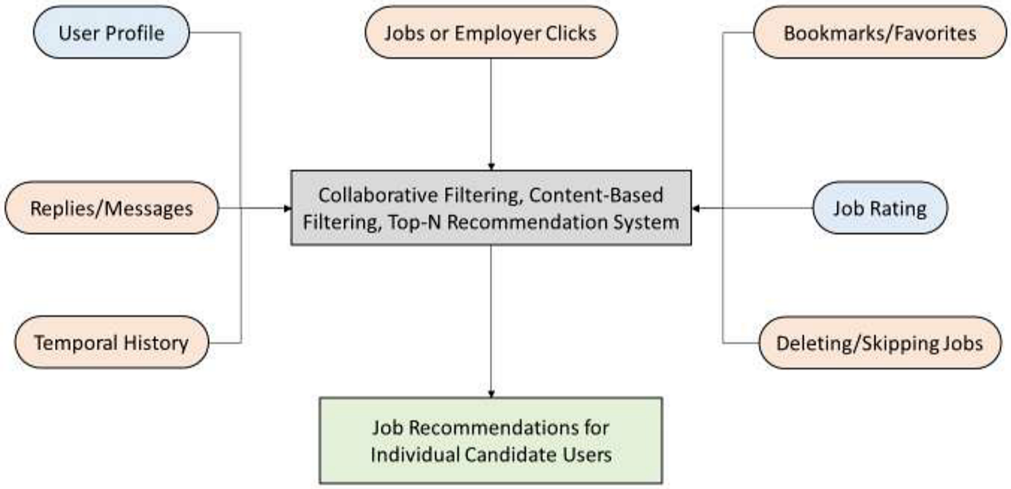
iii. Define features,development phases, and solution requirements.

iv. Provide specifications according to which the solutionis defined, managedanddelivered.

v. Provide the best businessrequire recommend by using the optimised and efficientalgorithm

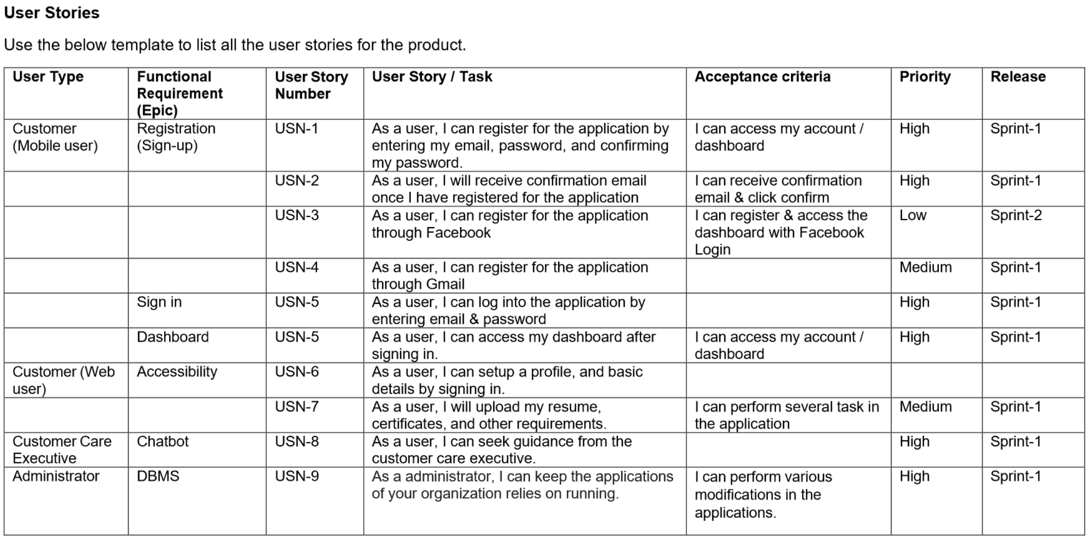
vi. Differentiate the fake job recommend by fake sites and be aware from the Scammers

JOB RECOMMENDED APPLICATION : (SOLUTION ARCHITECTURE)


User Profile
Jobs or Employer Clicks
Bookmarks/Favorites
Replies/Messages
Collaborative Filtering, Content-Based Filtering, Top-N Recommendation System
Job Rating
Temporal History
Deleting/Skipping Jobs
Job Recommendations for Individual Candidate Users

PNT2022TMID02696

5.3 USER STORIES:



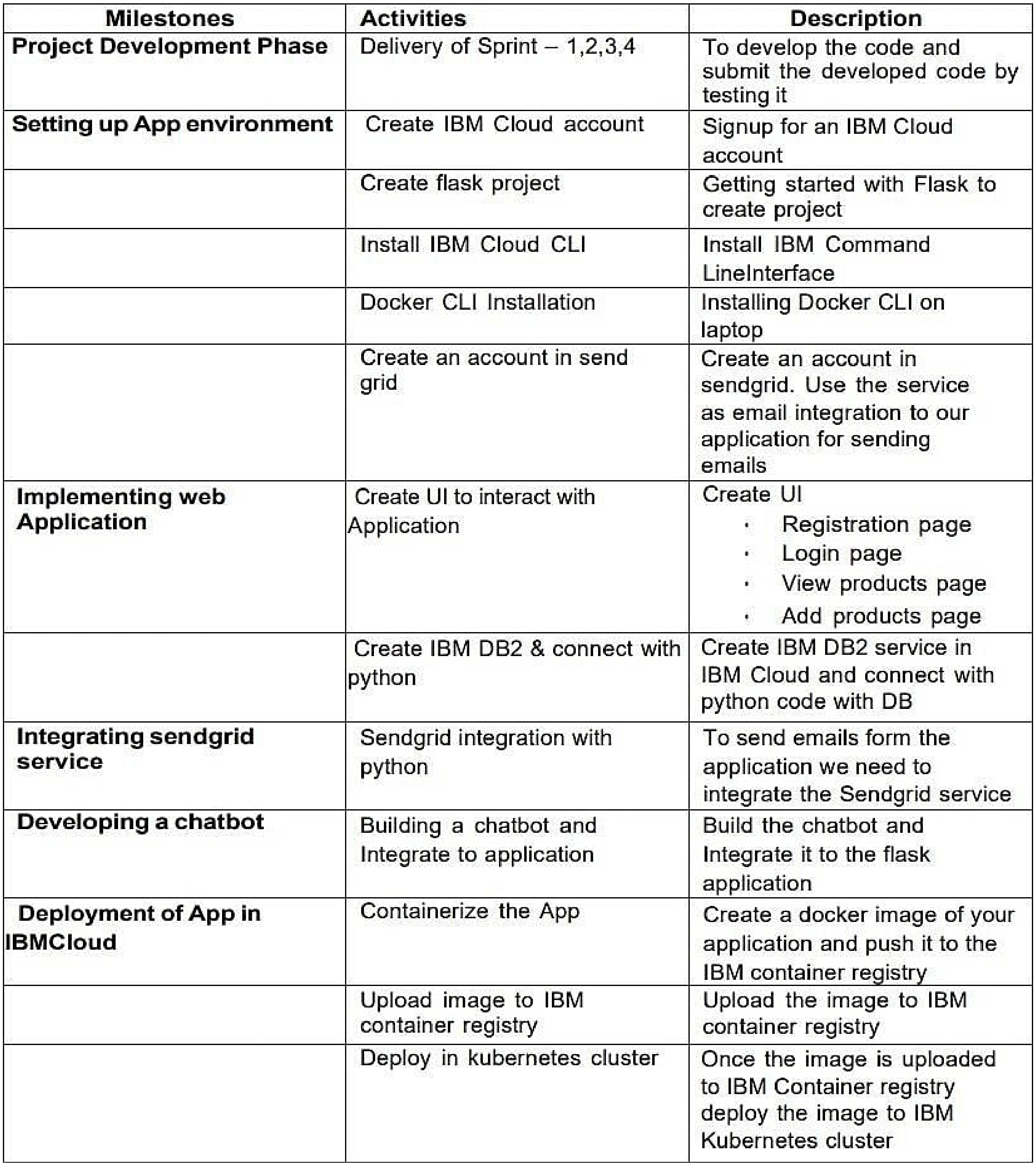
User Stories

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

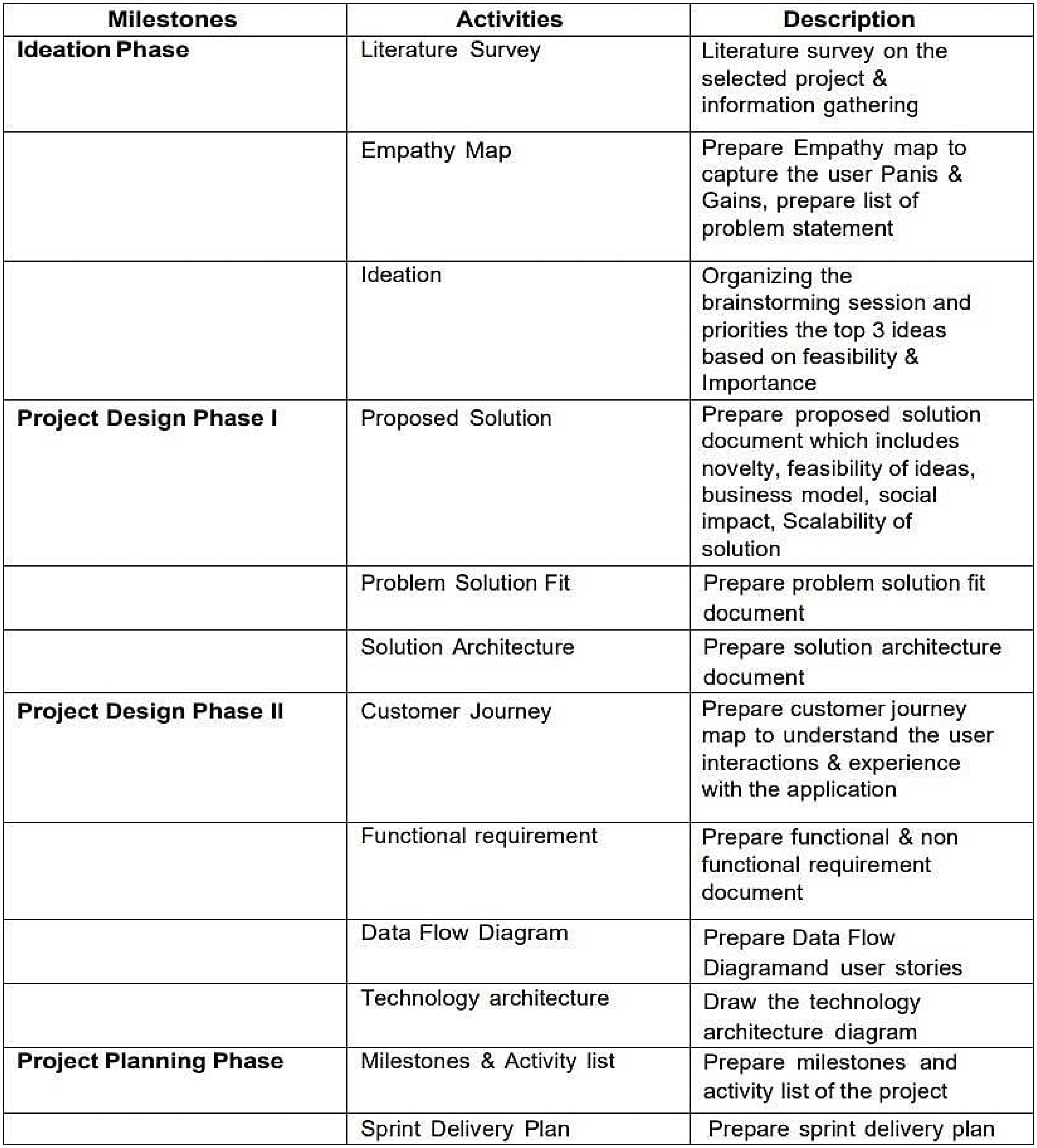
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration (Sign-up)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	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 Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Sign in	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-5	As a user, I can access my dashboard after signing in.	I can access my account / dashboard	High	Sprint-1
Customer (Web user)	Accessibility	USN-6	As a user, I can setup a profile, and basic details by signing in.			
		USN-7	As a user, I will upload my resume, certificates, and other requirements.	I can perform several task in the application	Medium	Sprint-1
Customer Care Executive	Chatbot	USN-8	As a user, I can seek guidance from the customer care executive.		High	Sprint-1
Administrator	DBMS	USN-9	As a administrator, I can keep the applications of your organization relies on running.	I can perform various modifications in the applications.	High	Sprint-1

6. PROJECT PLANNING &SCHEDULE

6.1 SPRINT PLANNING & ESTIMATION:

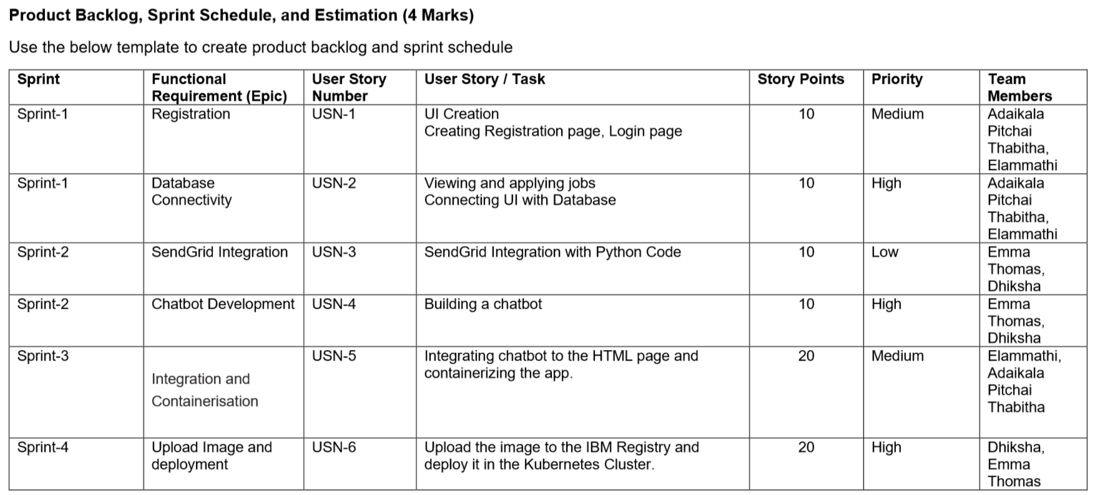


Milestones	Activities	Description
Project Development Phase	Delivery of Sprint – 1,2,3,4	To develop the code and submit the developed code by testing it
Setting up App environment	Create IBM Cloud account	Signup for an IBM Cloud account
	Create flask project	Getting started with Flask to create project
	Install IBM Cloud CLI	Install IBM Command LineInterface
	Docker CLI Installation	Installing Docker CLI on laptop
	Create an account in send grid	Create an account in sendgrid. Use the service as email integration to our application for sending emails
Implementing web Application	Create UI to interact with Application	Create UI • Registration page • Login page • View products page • Add products page
	Create IBM DB2 & connect with python	Create IBM DB2 service in IBM Cloud and connect with python code with DB
Integrating sendgrid service	Sendgrid integration with python	To send emails form the application we need to integrate the Sendgrid service
Developing a chatbot	Building a chatbot and Integrate to application	Build the chatbot and Integrate it to the flask application
Deployment of App in IBMCloud	Containerize the App	Create a docker image of your application and push it to the IBM container registry
	Upload image to IBM container registry	Upload the image to IBM container registry
	Deploy in kubernetes cluster	Once the image is uploaded to IBM Container registry deploy the image to IBM Kubernetes cluster



Milestones	Activities	Description
Ideation Phase	Literature Survey	Literature survey on the selected project & information gathering
	Empathy Map	Prepare Empathy map to capture the user Panis & Gains, prepare list of problem statement
	Ideation	Organizing the brainstorming session and priorities the top 3 ideas based on feasibility & Importance
Project Design Phase I	Proposed Solution	Prepare proposed solution document which includes novelty, feasibility of ideas, business model, social impact, Scalability of solution
	Problem Solution Fit	Prepare problem solution fit document
	Solution Architecture	Prepare solution architecture document
Project Design Phase II	Customer Journey	Prepare customer journey map to understand the user interactions & experience with the application
	Functional requirement	Prepare functional & non functional requirement document
	Data Flow Diagram	Prepare Data Flow Diagramand user stories
	Technology architecture	Draw the technology architecture diagram
Project Planning Phase	Milestones & Activity list	Prepare milestones and activity list of the project
	Sprint Delivery Plan	Prepare sprint delivery plan

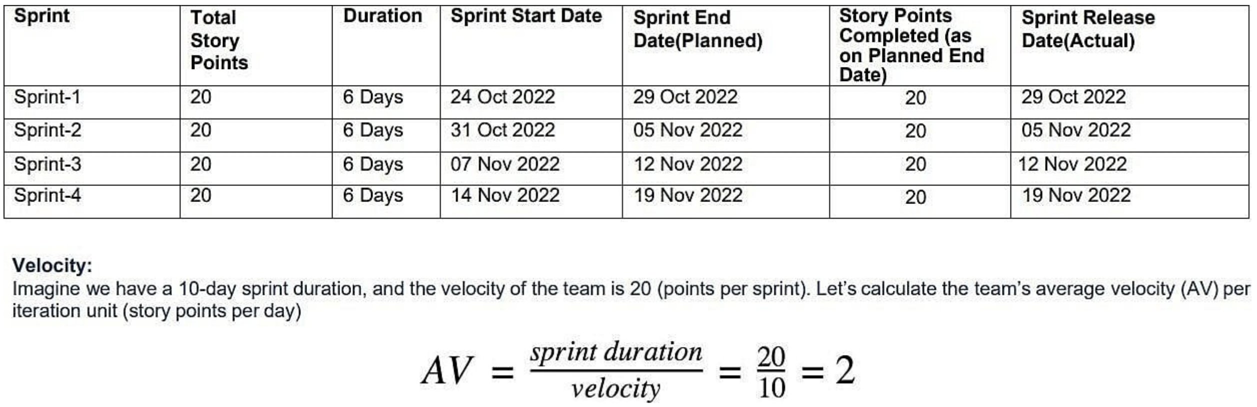
6.2 SPRINT DELIVERY SCHEDULE:



Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	UI Creation Creating Registration page, Login page	10	Medium	Adaikala Pitchai Thabitha, Elammathi
Sprint-1	Database Connectivity	USN-2	Viewing and applying jobs Connecting UI with Database	10	High	Adaikala Pitchai Thabitha, Elammathi
Sprint-2	SendGrid Integration	USN-3	SendGrid Integration with Python Code	10	Low	Emma Thomas, Dhiksha
Sprint-2	Chatbot Development	USN-4	Building a chatbot	10	High	Emma Thomas, Dhiksha
Sprint-3	Integration and Containerisation	USN-5	Integrating chatbot to the HTML page and containerizing the app.	20	Medium	Elammathi, Adaikala Pitchai Thabitha
Sprint-4	Upload Image and deployment	USN-6	Upload the image to the IBM Registry and deploy it in the Kubernetes Cluster.	20	High	Dhiksha, Emma Thomas



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	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

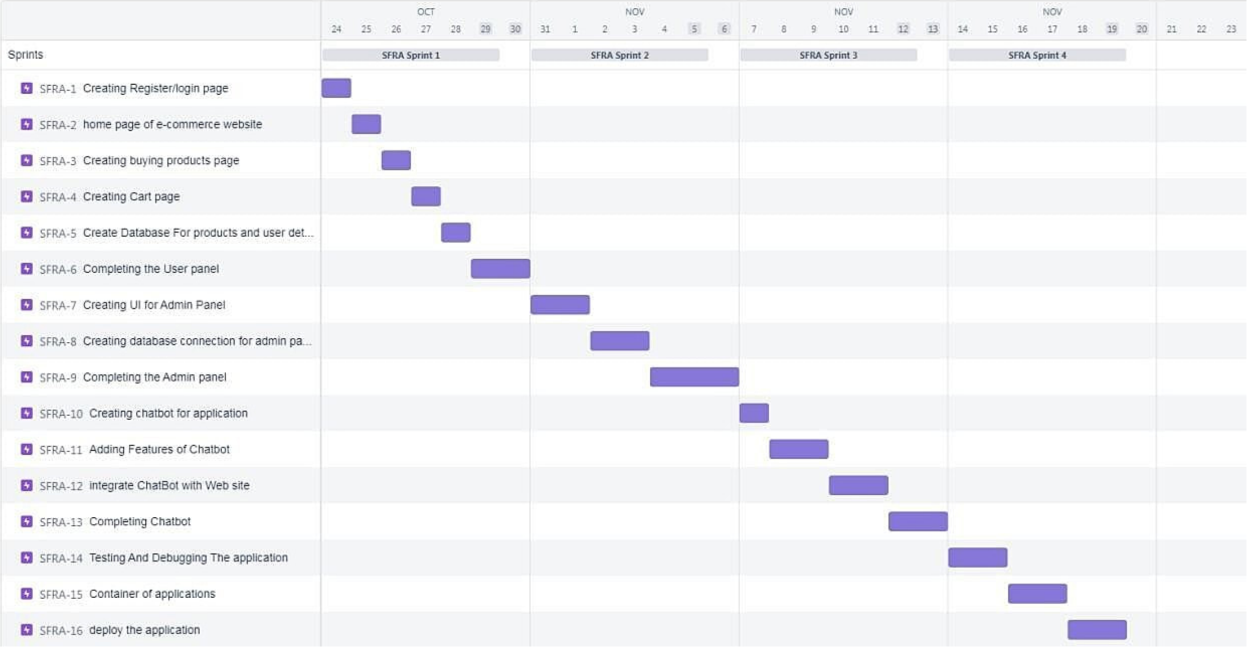
Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.3 REPORTS FROM JIRA:

Burndown Chart:



Sprints

SFRA-1 Creating Register/login page

SFRA-2 home page of e-commerce website

SFRA-3 Creating buying products page

SFRA-4 Creating Cart page

SFRA-5 Create Database For products and user det...

SFRA-6 Completing the User panel

SFRA-7 Creating UI for Admin Panel

SFRA-8 Creating database connection for admin pa...

SFRA-9 Completing the Admin panel

SFRA-10 Creating chatbot for application

SFRA-11 Adding Features of Chatbot

SFRA-12 integrate ChatBot with Web site

SFRA-13 Completing Chatbot

SFRA-14 Testing And Debugging The application

SFRA-15 Container of applications

SFRA-16 deploy the application

7. CODING & SOLUTIONING

7.1 FEATURE-1:

index.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Form | CodingLab</title>
    <link rel="stylesheet" href="style.css">
    <script src="https://kit.fontawesome.com/a076d05399.js"></script>
  </head>
  <body>
    <div class="container">
<div class="wrapper">
        <div class="title"><span>Login Form</span></div>
        <form action="#">
          <div class="row">
            <i class="fas fa-user"></i>
            <input type="text" placeholder="Email or Phone" required>
          </div>
```

```
<div class="row">
  <i class="fas fa-lock"></i>
  <input type="password" placeholder="Password" require
```

```
</div>
      <div class="pass"><a href="#">Forgot password?</a></div>
      <div class="row button">
       <input type="submit" value="Login">
      </div>
      <div class="signup-link">Not a member? <a href="#">Signup now</a></div>
     </form>
    </div>
   </div>
  </body>
 </html>
```

Style.css

```
@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600;700&display=
s wap'); *{ margin: 0; padding: 0; box-sizing: border-box; font-family: 'Poppins',sans-
serif;
}
body{    background:
#1abc9c;    overflow:
hidden;
}
::selection{ background:
rgba(26,188,156,0.3);
}
.container{ max-
width: 440px;
padding: 0 20px;
margin: 170px auto;
```

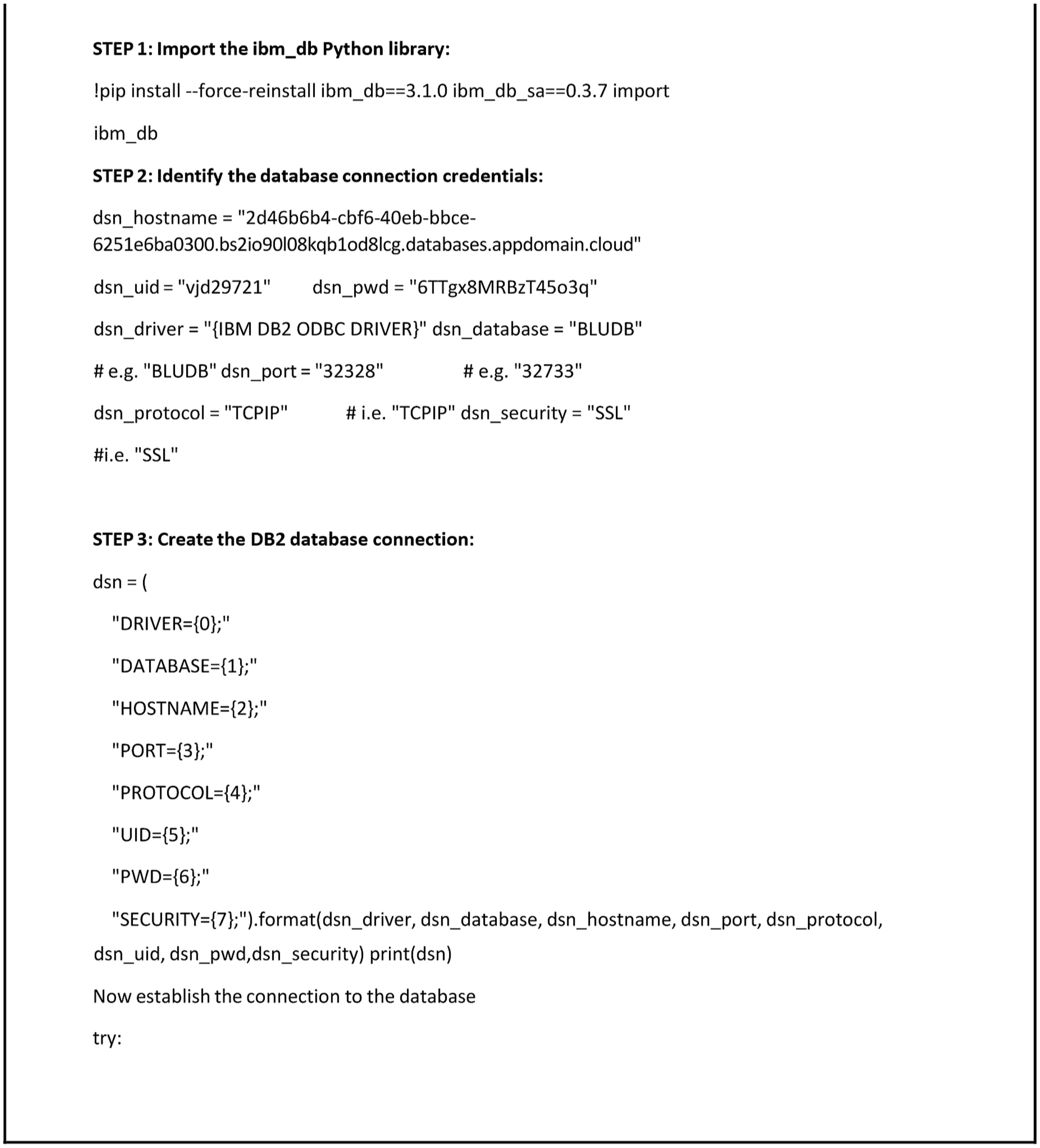
```
}
.wrapper{ width: 100%; background: #fff;
border-radius: 5px; box-shadow: 0px 4px 10px
1px rgba(0,0,0,0.1);
}
.wrapper .title{ height:
90px; background:
#16a085; border-radius:
5px 5px 0 0;
  color: #fff; font-
size: 30px; font-
weight: 600;
  display: flex; align-
items: center; justify-
content: center;
}
.wrapper form{ padding: 30px
25px 25px 25px;
}
.wrapper form .row{
  height: 45px;
  margin-bottom: 15px;
  position: relative;
}
.wrapper form .row input{
height: 100%; width:
100%; outline: none;
padding-left: 60px;
border-radius: 5px;
```

```
border: 1px solid lightgrey;
font-size: 16px;  transition:
all 0.3s ease;
}
form .row input:focus{ border-color: #16a085; box-
shadow: inset 0px 0px 2px 2px rgba(26,188,156,0.25);
}
form .row input::placeholder{
color: #999;
}
.wrapper form .row i{
position: absolute;
width: 47px; height:
100%;
  color: #fff; font-
size: 18px;
  background: #16a085;
border: 1px solid #16a085;
border-radius: 5px 0 0 5px;
display: flex; align-items:
center; justify-content:
center;
}
.wrapper form .pass{
margin: -8px 0 20px 0;
}
.wrapper form .pass a{
color: #16a085;  font-
```

INTEGRATING CHATBOT TO HTML PAGE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1>My Chatbot</h1>
    <blockquote>Click the bottom right corner to chat</blockquote>
<script>
  window.watsonAssistantChatOptions = {
    integrationID: "01ca5fe5-3f42-4a97-8965-332afedd97be", // The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "5683f375-e95c-4fa1-8471-5b76177675c2", // 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>
</body>
</html>
```

7.3 DATABASE SCHEMA :



STEP 1: Import the ibm_db Python library:

```
!pip install --force-reinstall ibm_db==3.1.0 ibm_db_sa==0.3.7 import

ibm_db
```

STEP 2: Identify the database connection credentials:

```
dsn_hostname = "2d46b6b4-cbf6-40eb-bbce-
6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"

dsn_uid = "vjd29721"        dsn_pwd = "6TTgx8MRBzT45o3q"

dsn_driver = "{IBM DB2 ODBC DRIVER}" dsn_database = "BLUDB"

# e.g. "BLUDB" dsn_port = "32328"              # e.g. "32733"

dsn_protocol = "TCPIP"          # i.e. "TCPIP" dsn_security = "SSL"

#i.e. "SSL"
```

STEP 3: Create the DB2 database connection:

```
dsn = (

    "DRIVER={0};"

    "DATABASE={1};"

    "HOSTNAME={2};"

    "PORT={3};"

    "PROTOCOL={4};"

    "UID={5};"

    "PWD={6};"

    "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port, dsn_protocol,
dsn_uid, dsn_pwd,dsn_security) print(dsn)
```

Now establish the connection to the database

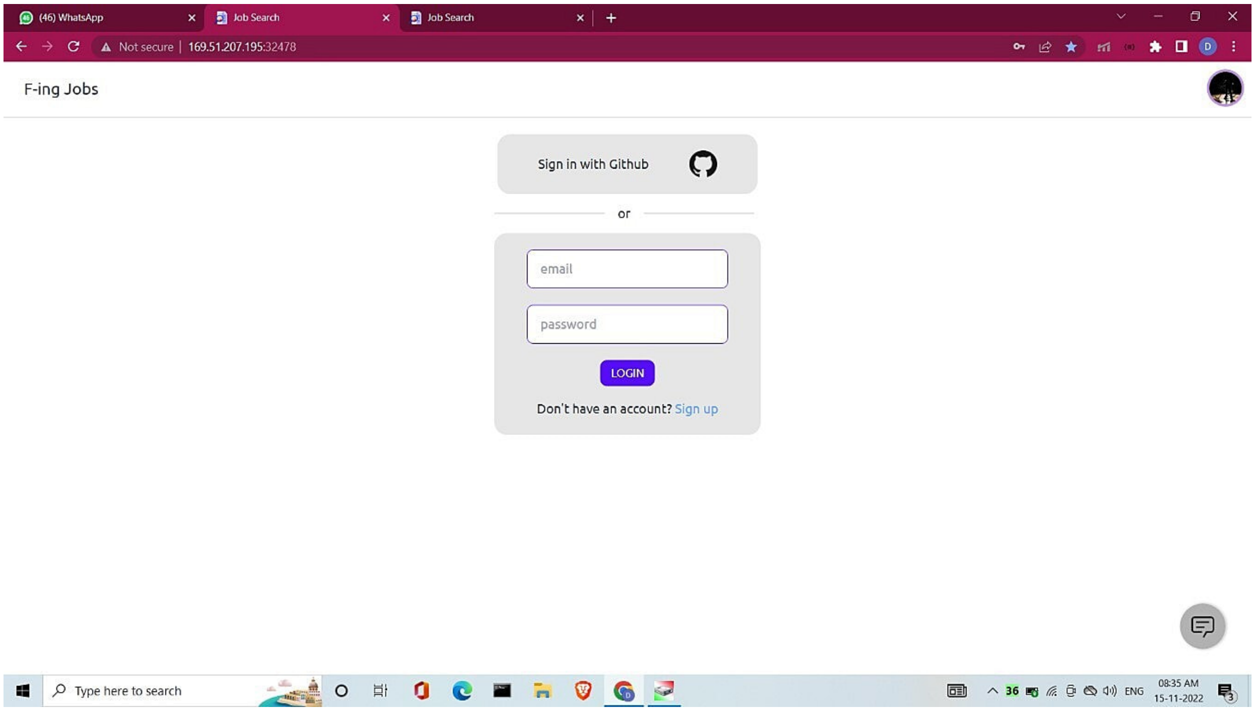
```
try:
   conn = ibm_db.connect(dsn, "", "")      print ("Connected to database: ", dsn_database, "as user:
", dsn_uid, "on host: ", dsn_hostname)

except:
   print ("Unable to connect: ", ibm_db.conn_errormsg() )

server = ibm_db.server_info(conn)

print ("DBMS_NAME: ", server.DBMS_NAME)
print    ("DBMS_VER:    ",     server.DBMS_VER)
print ("DB_NAME:  ", server.DB_NAME)

client = ibm_db.client_info(conn)

print  ("DRIVER_NAME:                  ",   client.DRIVER_NAME)   print
("DRIVER_VER:                                ",    client.DRIVER_VER)   print
("DATA_SOURCE_NAME:   ",   client.DATA_SOURCE_NAME)   print
("DRIVER_ODBC_VER:            ",   client.DRIVER_ODBC_VER)   print
("ODBC_VER:            ", client.ODBC_VER)
print ("ODBC_SQL_CONFORMANCE: ", client.ODBC_SQL_CONFORMANCE)
print ("APPL_CODEPAGE:        ", client.APPL_CODEPAGE) print
("CONN_CODEPAGE:       ", client.CONN_CODEPAGE)
```

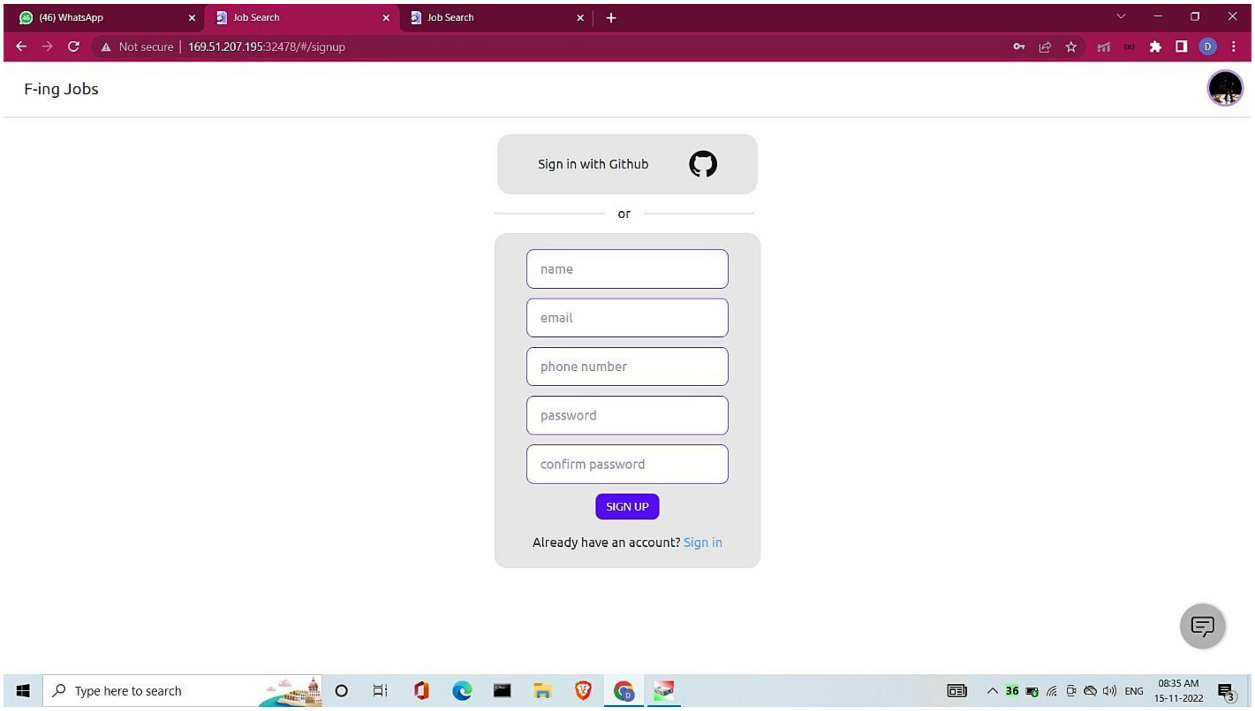
STEP 4: Close the Connection:

```
ibm_db.close(conn)
```

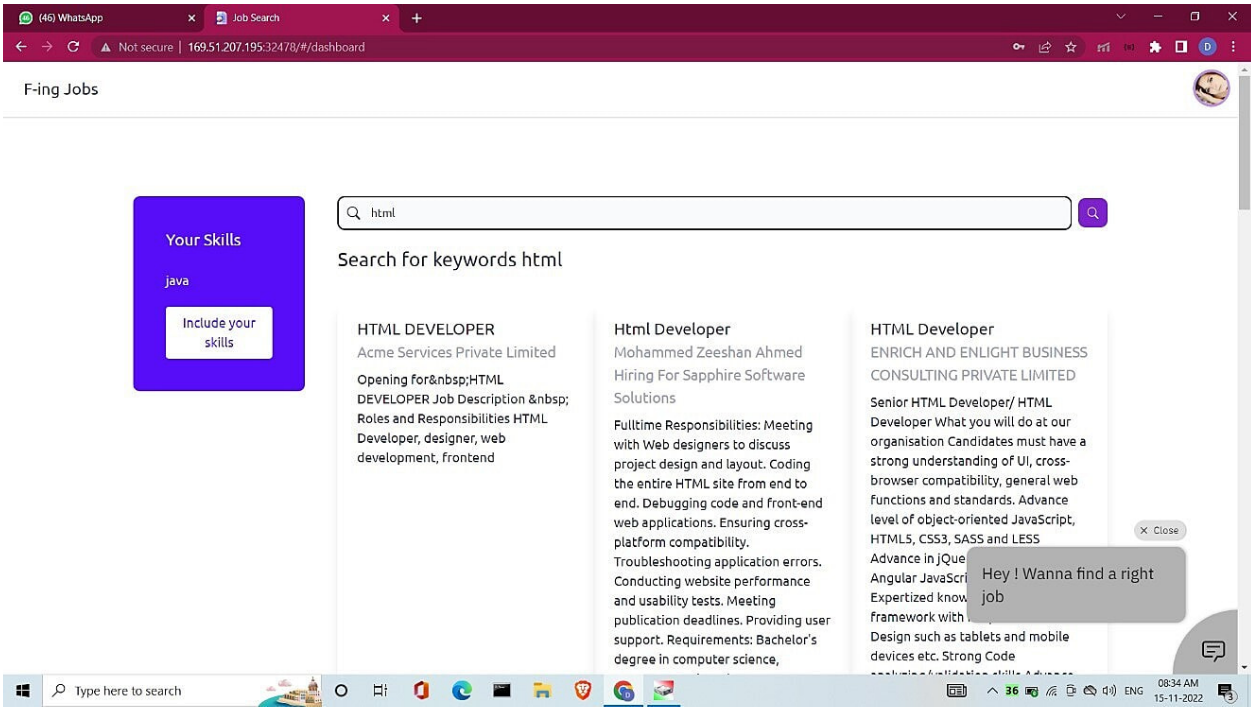
8. TESTING

8.1 TEST CASES:





PNT2022TMID02696



Search for keywords html

HTML DEVELOPER
Acme Services Private Limited
Opening for HTML DEVELOPER Job Description   Roles and Responsibilities HTML Developer, designer, web development, frontend

Html Developer
Mohammed Zeeshan Ahmed Hiring For Sapphire Software Solutions
Fulltime Responsibilities: Meeting with Web designers to discuss project design and layout. Coding the entire HTML site from end to end. Debugging code and front-end web applications. Ensuring cross-platform compatibility. Troubleshooting application errors. Conducting website performance and usability tests. Meeting publication deadlines. Providing user support. Requirements: Bachelor's degree in computer science,

HTML Developer
ENRICH AND ENLIGHT BUSINESS CONSULTING PRIVATE LIMITED
Senior HTML Developer/ HTML Developer What you will do at our organisation Candidates must have a strong understanding of UI, cross-browser compatibility, general web functions and standards. Advance level of object-oriented JavaScript, HTML5, CSS3, SASS and LESS Advance in jQue Angular JavaScri Expertized know framework with Design such as tablets and mobile devices etc. Strong Code

Hey ! Wanna find a right job

9. RESULTS

9.1 PERFORMANCE METRICS:

The performance of a recommendation algorithm is evaluated by using some specific metrics that indicate the accuracy of the system. The type of metric used depends on the type of filtering technique. Root Mean Square Error (RMSE), Receiver Operating Characteristics (ROC), Area Under Cover (AUC), Precision, Recall and F1 score is generally used to evaluate the performance or accuracy of the recommendation algorithms.

Root-mean square error (RMSE). RMSE is widely used in evaluating and comparing the performance of a recommendation system model compared to other models. A lower RMSE value indicates higher performance by the recommendation model. RMSE, as mentioned by [61], can be as represented as follows:

$$RMSE = \sqrt{\frac{1}{N_p} \sum_{u,i} (p_{ui} - r_{ui})^2} \tag{1}$$

where, N_p is the total number of predictions, p_{ui} is the predicted rating that a user u will select an item i and r_{ui} is the real rating.

Precision. Precision can be defined as the fraction of correct recommendations or predictions (known as True Positive) to the total number of recommendations provided, which can be as represented as follows:

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\,(TP) + False\ Positive\ (FP)} \tag{2}$$

It is also defined as the ratio of the number of relevant recommended items to the number of recommended items expressed as percentages.

Recall. Recall can be defined as the fraction of correct recommendations or predictions (known as True Positive) to the total number of correct relevant recommendations provided, which can be as represented as follows:

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\,(TP) + False\ Negative\ (FN)} \tag{3}$$

It is also defined as the ratio of the number of relevant recommended items to the total number of relevant items expressed as percentages.

F1 Score. F1 score is an indicator of the accuracy of the model and ranges from 0 to 1, where a value close to 1 represents higher recommendation or prediction accuracy. It represents precision and recall as a single metric and can be as represented as follows:

$$F1\ score = 2 \times \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

Coverage. Coverage is used to measure the percentage of items which are recommended by the algorithm among all of the items.

Accuracy. Accuracy can be defined as the ratio of the number of total correct recommendations to the total recommendations provided, which can be as represented as follows:

$$Accuracy = \frac{TP + FN}{TP + FN + TN + FP} \tag{5}$$

Intersection over union (IoU). It represents the accuracy of an object detector used on a specific dataset.

$$IoU = \frac{TP}{TP + FN + FP} \tag{6}$$

ROC. ROC curve is used to conduct a comprehensive assessment of the algorithm's performance [57].

AUC. AUC measures the performance of recommendation and its baselines as well as the quality of the ranking based on pairwise comparisons [5].

Rank aware top-N metrics. The rank aware top-N recommendation metric finds some of the interesting and unknown items that are presumed to be most attractive to a user [63].

Mean reciprocal rank (MRR), mean average precision (MAP) and normalized discounted cumulative gain (NDCG) are three most popular rank aware metrics.

MRR: is calculated as a mean of the reciprocal of the position or rank of first relevant recommendation [64][65]. MRR as mentioned by [64][65] can be expressed as follows:

$$MRR = \frac{1}{N_u}\sum_{u \in N_u}\frac{1}{L_u^n\,[k] \in R_u} \tag{7}$$

where u, N_u and R_u indicate specific user, total number of users and the set of items rated by the user, respectively. L indicates list of ranking length (n) for user (u) and k represents the position of the item found in the he lists L.

MAP: MAP is calculated by determining the mean of average precision at the points where relevant products or items are found. MAP as mentioned by [65] can be expressed as follows.

$$MAP = \frac{1}{N_u\,|R_u|} \sum_{k=1}^{n} \mathbb{1}\left(L_u^n\,[k] \in R_u\right) P_u@k \tag{8}$$

where P_u represents precision in selecting relevant item for the user.

NDCG: NDCG is calculated by determining the graded relevance and positional information of the recommended items, which can be expressed as follows [65].

$$NDCG_u = \frac{\sum_{k=1}^{n} G\,(u, n, k) D\,(k)}{\sum_{k=1}^{n} G^*\,(u, n, k) D\,(k)} \tag{9}$$

where *D* (*k*) is a discounting function, *G* (*u*, *n*, *k*) is the gain obtained recommending an item found at *k*-th position from the list L and *G** (*u*, *n*, *k*) is the gain related to *k*-th item in the ideal ranking of *n* size for u user.

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- It helps candidates to search the job which perfectly suites them and make them aware of all the job openings.
- It help recruiters of the company to choose the right candidates for their organisations with appropriate skills.
- Since it is cloud application , it does require any installation of softwares and is portable

DISADVANTAGES:

- Privacy concerns.
- Too many choices.
- Cold-start problem.
- It is costly.
- Uninterrupted internet connection is required for smooth functioning of application.

11. CONCLUSION

This application helps in finding job opportunities relevant to the applicant's skill set. We have built this application with flask, ibm cloud services along with chatbot integration.

12. FUTURE SCOPE

Future directions of our work will focus on performing more 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. We can use machine learning technicques to recommend data in a efficient way.

13. PROJECT LINKS

REPO LINK:

https://github.com/IBM-EPBL/IBM-Project-14891-1659591618

PROJECT DEMO LINK:

https://youtu.be/vlsgCkzxNE8

DEPLOYED WEB APP URL:

http://169.51.207.195:32478/