PNT2022TMID12203

IBM – NALAIYA THIRAN PROJECT

SKILL JOB RECOMMENDER APPLICATION

ABSTRACT

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

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 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 issimplified. The proposed recommendation system is tested on an array of test cases with a fullyfunctioning 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

PROJECT REPORT FORMAT

1.INTRODUCTION

1.1 PROJECT OVERVIEW

Nowadays, job search is a task commonly done on the Internet using job search engine sites like LinkedIn1, Indeed2 , and others. Commonly, a job seeker has two ways to search a job using these sites: 1) doing a query based on keywords related to the job vacancy that he/she is looking for, or 2) creating and/or updating a professional profile containing data related to his/her education, professional experience, professional skills and other, and receive personalized job recommendations based on this data. Sites providing support to the former case are more popular and have a simpler structure; however, their recommendations are less accurate than those of the sites using profile data.

1.2 PURPOSE

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

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

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

1. Students/Job seekers find their desired job based ontheir skillset.
2. Integrating Intelligent CHATBOT for job recommendation application.
3. A study of LinkedIn as an Employment Tool for Jobseeker & Recruiter.
4. Cloud storage and sharing services.

2.2 REFERENCES

1. https://www.researchgate.net/publication/325697854_Job_Recommendation_based_on_Job_Seeker_Skills_An_Empirical_Study

2. https://smartinternz.com/saas-guided-project/1/skill-job-recommender

3. https://www.semanticscholar.org/paper/Job-Recommendation-Based-on-Job-Seeker-Skills%3A-An-Valverde-Rebaza-Puma/3d49cb87fde48591a1c1a2ef088d885c4480adbf

2.3 PROBLEM STATEMENT DEFINITION

Dealing with the enormous amount of recruitinginformation 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.

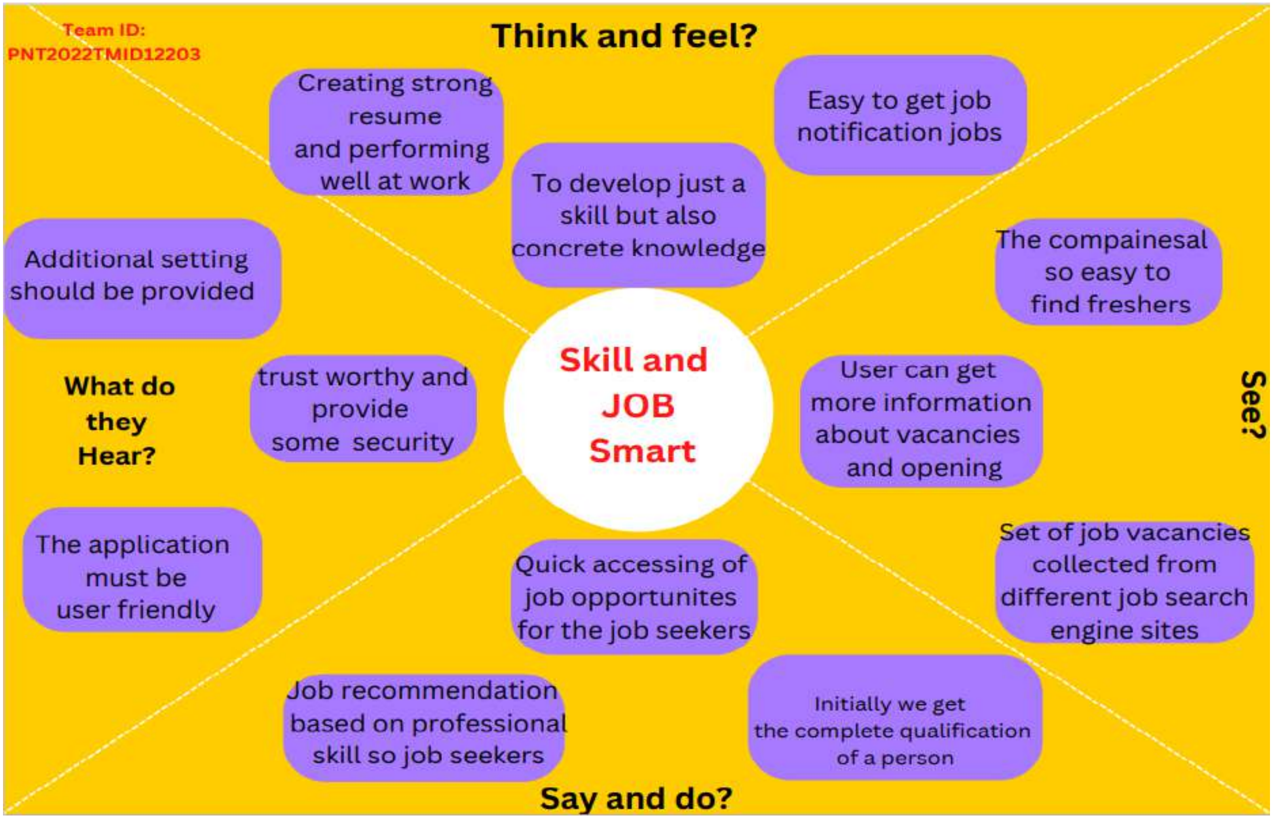
Job recommender technology aims to help job seekers infinding jobs that match their skills.

The internet caused a substantial impact on the recruitment process through the creation of e-recruiting platforms that become a primary recruitment channel in most companies. While companies established job positions on these portals, job-seeker uses them to publishtheir profiles. E-Recruitment platforms accomplished clear advantages for both recruiters and job-seekers by reducing the recruitment time and advertisement cost

.Recommender system technology aims to help users in finding items that match their preferences; it has a successful usage in a wide-range of applications to deal with problems related to information overload efficiently.In order to improve the e-recruiting functionality, many recommender system approaches have been proposed. This paper will analyze e-recruiting process and related issues for building personalized recommender system of candidates.

3.IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



Team ID: PNT2022TMID12203

Think and feel?

Creating strong resume and performing well at work

To develop just a skill but also concrete knowledge

Easy to get job notification jobs

What do they Hear?

Additional setting should be provided

trust worthy and provide some security

The application must be user friendly

Skill and JOB Smart

See?

The compainesal so easy to find freshers

User can get more information about vacancies and opening

Set of job vacancies collected from different job search engine sites

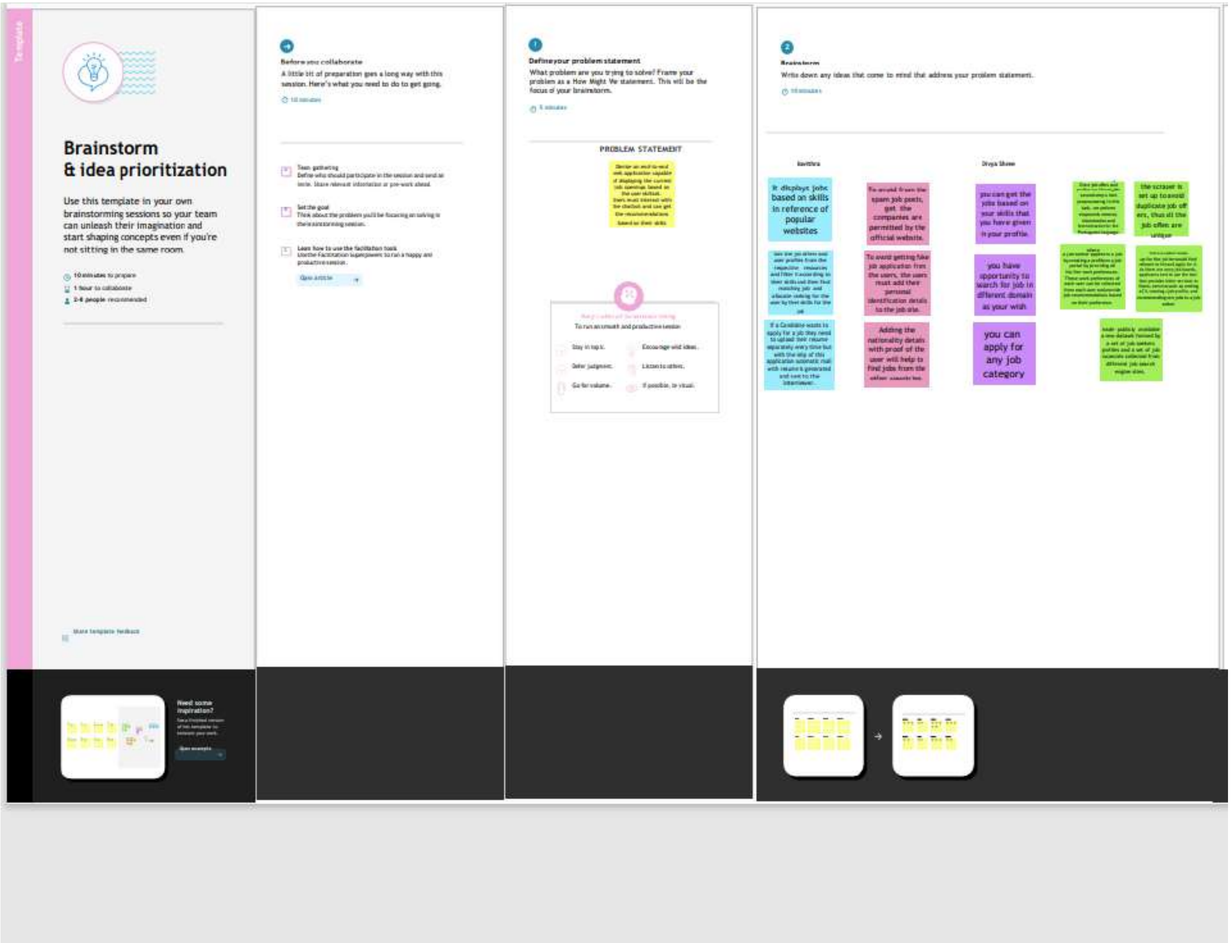
Say and do?

Quick accessing of job opportunites for the job seekers

Job recommendation based on professional skill so job seekers

Initially we get the complete qualification of a person

3.2 IDEATION AND BRAINSTORMING



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

Share template feedback

Need some inspiration?
See a finished version of this template to kickstart your work.
Open example

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

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

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

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 STATEMENT

Devise an end-to-end web application capable of displaying the current job openings based on the user skillset. Users must interact with the chatbot and can get the recommendations based on their skills

[illegible]
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.

2
Brainstorm
Write down any ideas that come to mind that address your problem statement.
10 minutes

Kavithra

It displays jobs based on skills in reference of popular websites

[illegible] user profiles from the respective resources and filter it according to their skills and then find matching job and allocate ranking for the user by their skills for the job

If a Candidate wants to apply for a job they need to upload their resume separately every time but with the help of this application automatic mail with resume is generated and sent to the interviewer.

To avoid from the spam job posts, get the companies are permitted by the official website.

To avoid getting fake job application from the users, the users must add their personal identification details to the job site.

Adding the nationality details with proof of the user will help to find jobs from the other countries too.

Divya Shree

you can get the jobs based on your skills that you have given in your profile.

you have opportunity to search for job in different domain as your wish

you can apply for any job category

[illegible]

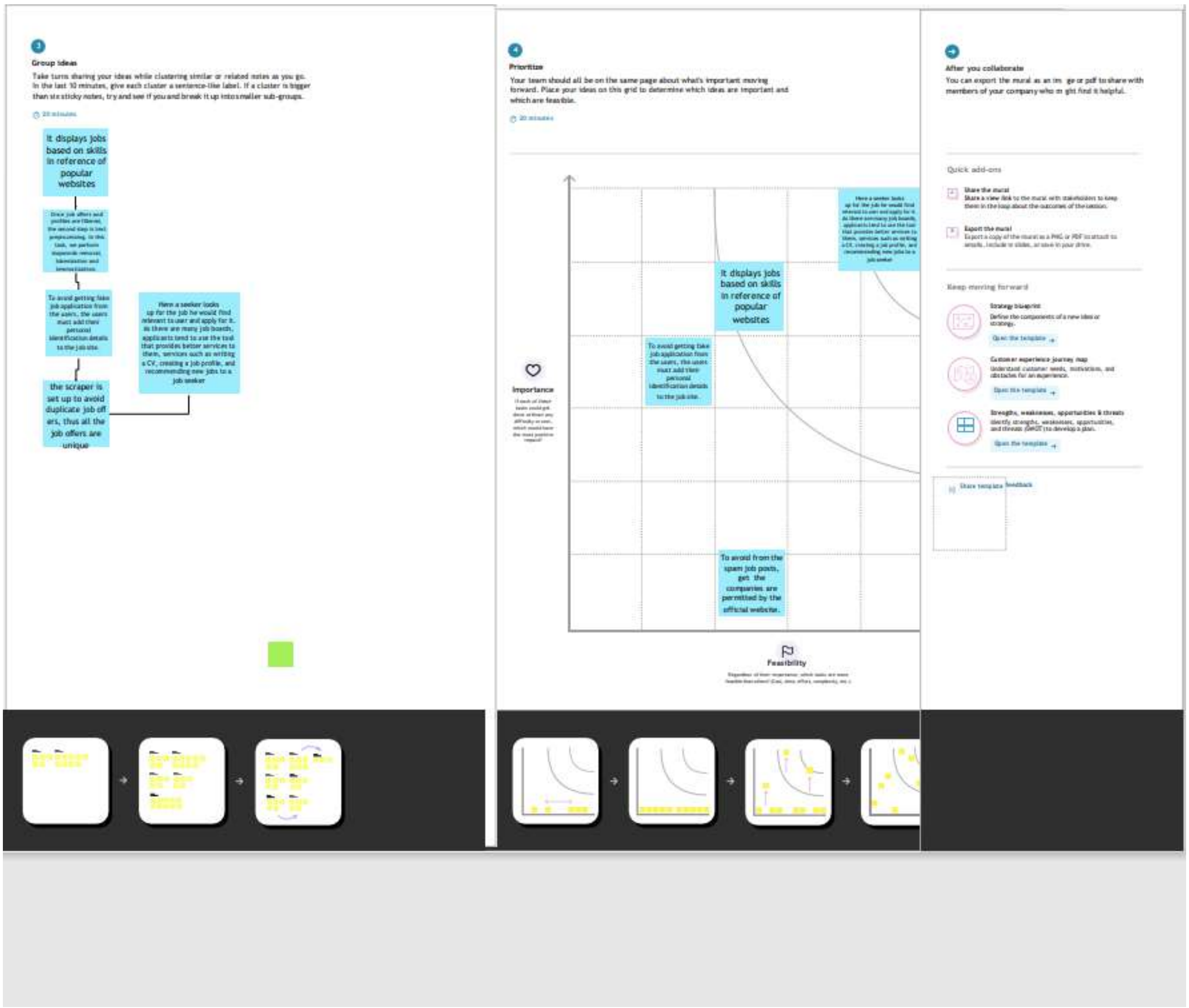
the scraper is set up to avoid duplicate job offers, thus all the job offers are unique

[illegible]

[illegible]

[illegible] publicly available a new dataset formed by a set of job seekers profiles and a set of job vacancies collected from different job search engine sites.

PNT2022TMID12203



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, 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

It displays jobs based on skills in reference of popular websites

Once job offers and profiles are filtered, the second step is text preprocessing. In this task, we perform stopwords removal, tokenization and lemmatization.

To avoid getting fake job application from the users, the users must add their personal identification details to the job site.

Here a seeker looks up for the job he would find relevant to user and apply for it. As there are many job boards, applicants tend to use the tool that provides better services to them, services such as writing a CV, creating a job profile, and recommending new jobs to a job seeker

the scraper is set up to avoid duplicate job off ers, thus all the job offers are unique

4

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

Here a seeker looks up for the job he would find relevant to user and apply for it. As there are many job boards, applicants tend to use the tool that provides better services to them, services such as writing a CV, creating a job profile, and recommending new jobs to a job seeker

It displays jobs based on skills in reference of popular websites

To avoid getting fake job application from the users, the users must add their personal identification details to the job site.

To avoid from the spam job posts, get the companies are permitted by the official website.

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an im ge or pdf to share with members of your company who m ght find it helpful.

Quick add-ons

Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

Export the mural
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

Share template feedback

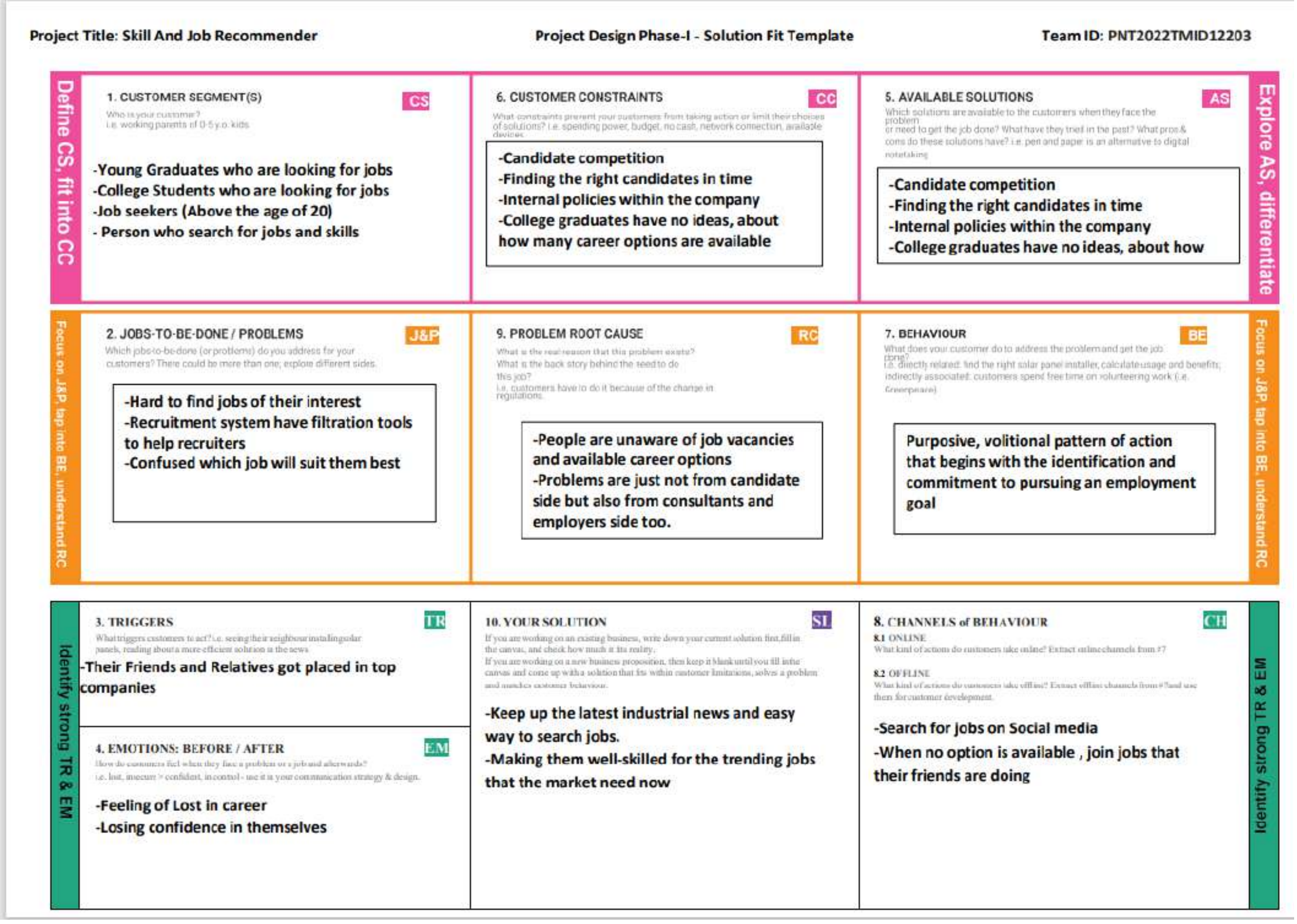
3.3 PROPOSED SOLUTON

Team ID	PNT2022TMID12203
Project Name	SKILL/JOB RECOMMENDER APPLICATION

S.No.	**Parameter**	**Description**
1.	Problem Statement (Problem to be solved)	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 chat bot 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 chat bot 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.
2.	Idea / Solution description	The contributions of this wok are threefold, we: i) made publicly available a new dataset formed by a set of job seekers profiles and a set of job vacancies collected from different job search engine sites ii) put forward the proposal of a framework for job recommendation based on professional skills of job seekers iii) Called out an evaluation to quantify empirically the recommendation abilities of two state-of-the-art methods, considering different configurations, within the proposed framework. We thus present a general panorama of job recommendation task aiming to facilitate research and ideal- world application design regarding this important issue.

3.	Novelty / Uniqueness	The best positions are suggested to any person according to her skills. While the position of known profiles is assumed to be correct, it should be noted that there are usually multiple advisable positions corresponding to a set of skills. A recommendation system should return a set of most likely positions and all of them can be equally valid. The recommendation method we use is simply based on representing both positions and profiles as comparable vectors and seeking for each profile the positions with the most similar vectors.
4.	Social Impact / Customer Satisfaction	Students will be benefited as they will get to know which job suits them based on their skill set and therefore Lack of Unemployment can Be reduced.
5.	Business Model (Revenue Model)	We can provide the application for job seekers in a subscription based and we can share the profiles with companies and generate the revenue by providing them best profiles.
6.	Scalability of the Solution	Data can be scaled up and scaled down according to number of current job openings available.

3.4 PROBLEM STATEMENT FIT



Project Title: Skill And Job Recommender

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID12203

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

-Young Graduates who are looking for jobs
-College Students who are looking for jobs
-Job seekers (Above the age of 20)
- Person who search for jobs and skills

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

-Candidate competition
-Finding the right candidates in time
-Internal policies within the company
-College graduates have no ideas, about how many career options are available

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem
or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

-Candidate competition
-Finding the right candidates in time
-Internal policies within the company
-College graduates have no ideas, about how

Explore AS, differentiate

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

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

-Hard to find jobs of their interest
-Recruitment system have filtration tools to help recruiters
-Confused which job will suit them best

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

-People are unaware of job vacancies and available career options
-Problems are just not from candidate side but also from consultants and employers side too.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Purposive, volitional pattern of action that begins with the identification and commitment to pursuing an employment goal

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

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news

-Their Friends and Relatives got placed in top companies

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

-Feeling of Lost in career
-Losing confidence in themselves

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

-Keep up the latest industrial news and easy way to search jobs.
-Making them well-skilled for the trending jobs that the market need now

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

-Search for jobs on Social media
-When no option is available , join jobs that their friends are doing

Identify strong TR & EM

4. REQUIREMENT ANALYSIS

4.1 Function Requirement

Software Required:

Python, Flask, Docker.

4.2 Non-Function Requirement

System Required:

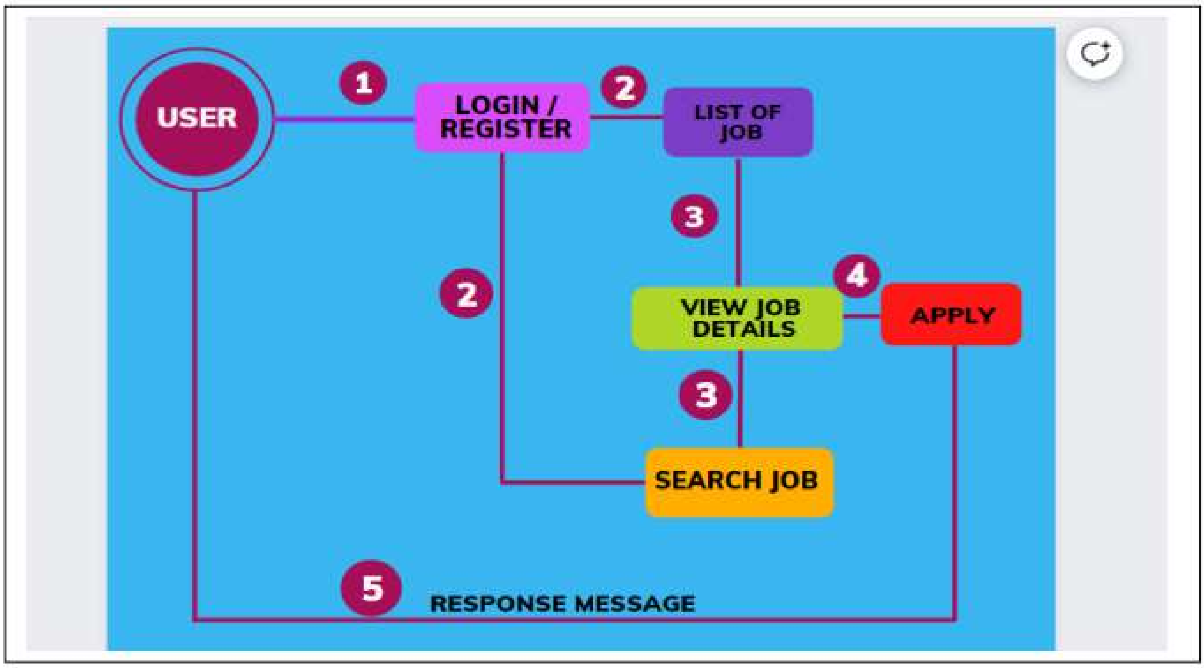
8GB RAM, Intel Core i3, OS-Windows/Linux/MAC, Laptop or Desktop.

5.PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

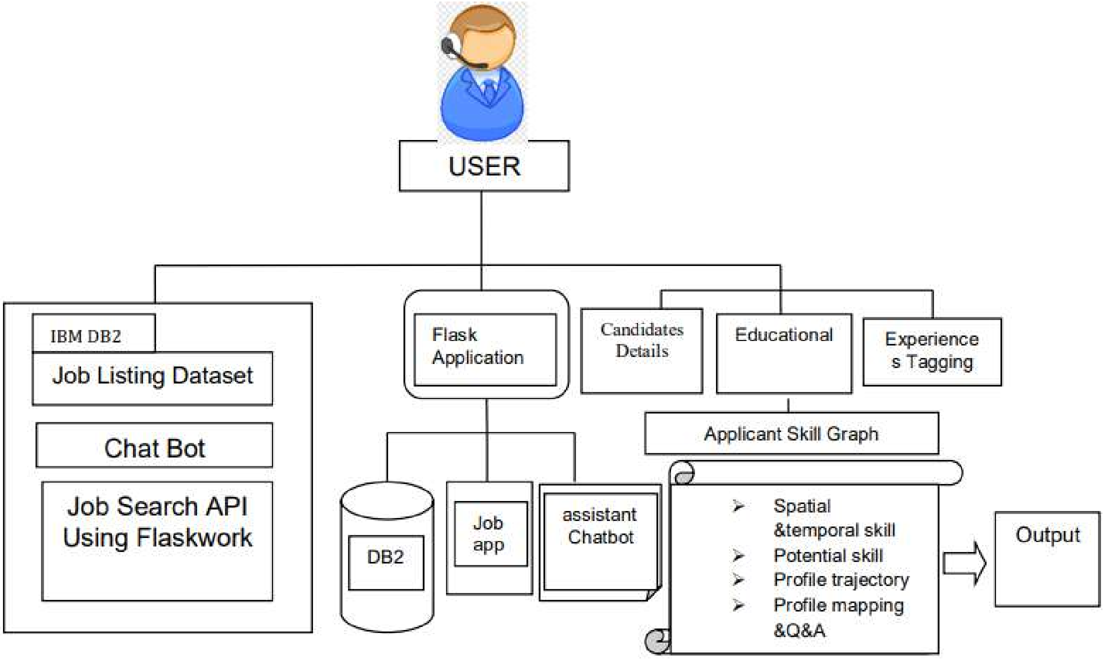
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, What changes the information, and where data is stored.

DATA FLOW DIAGRAM FOR JOBS RECOMMENDED APPLICATION

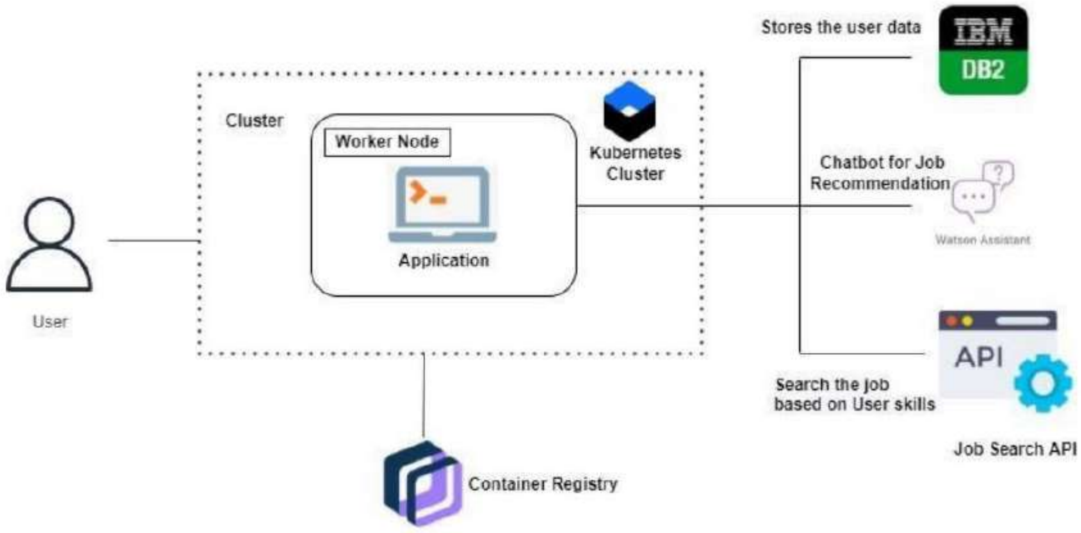


5.2 SOLUTION AND TECHINAL ARCHITECTURE

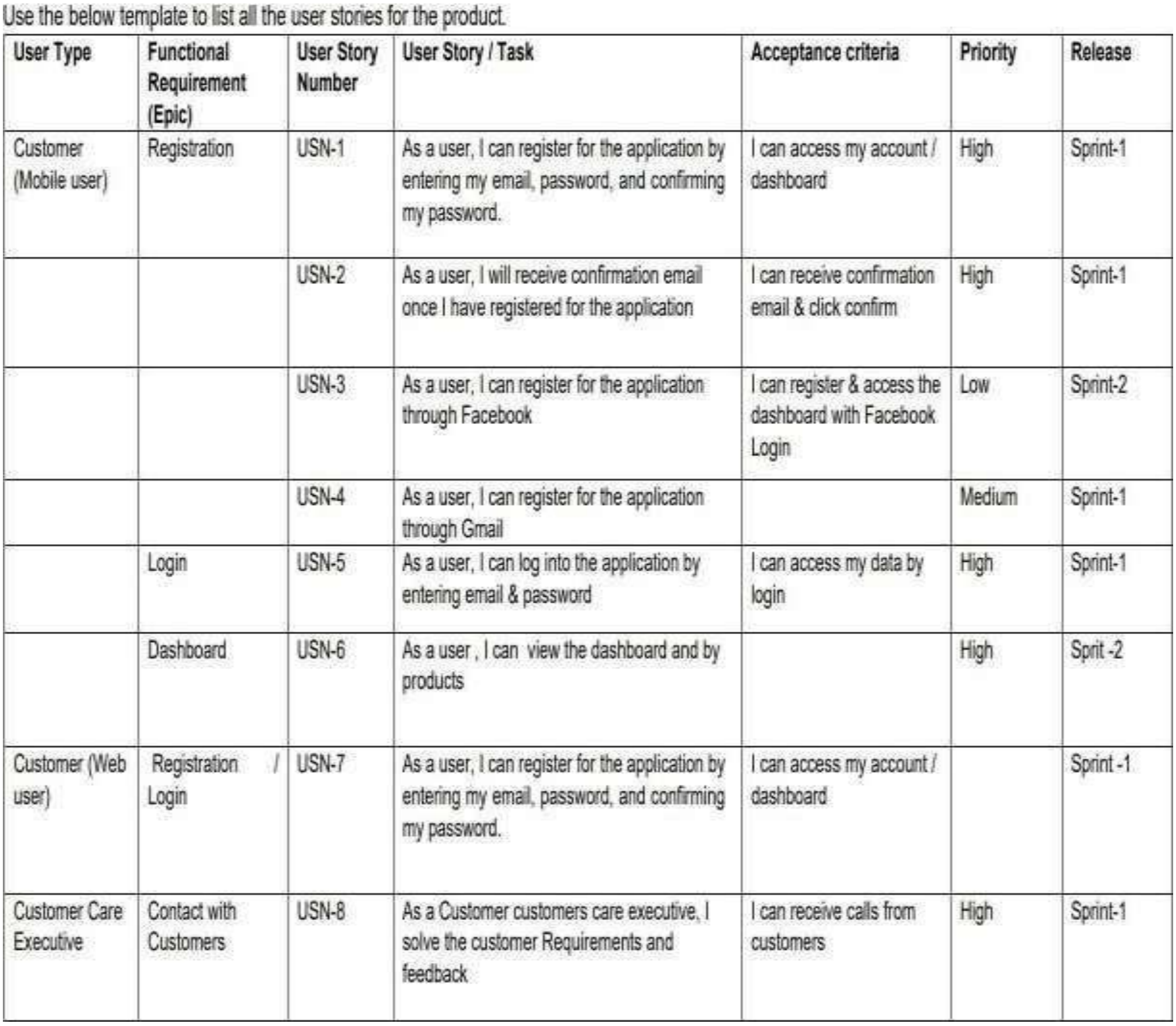
SOLUTION ARCHITECTURE



TECHNICAL ARCHITECTURE

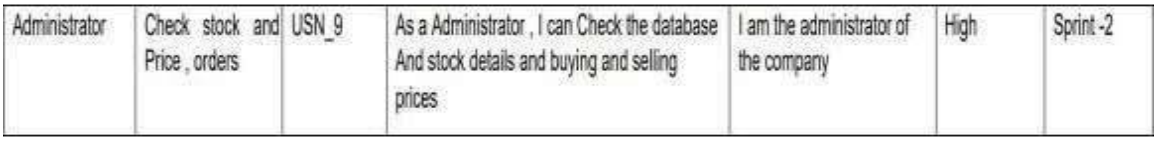


5.3 USER STORIES



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

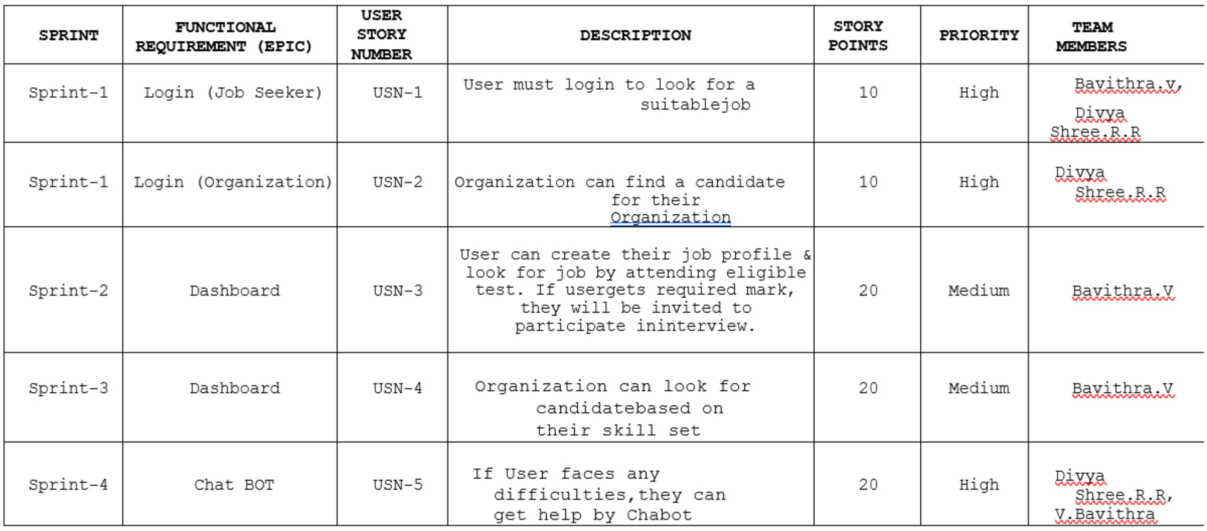
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	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 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
	Login	USN-5	As a user, I can log into the application by entering email & password	I can access my data by login	High	Sprint-1
	Dashboard	USN-6	As a user , I can view the dashboard and by products		High	Sprit -2
Customer (Web user)	Registration / Login	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard		Sprint -1
Customer Care Executive	Contact with Customers	USN-8	As a Customer customers care executive, I solve the customer Requirements and feedback	I can receive calls from customers	High	Sprint-1



Administrator	Check stock and Price , orders	USN_9	As a Administrator , I can Check the database And stock details and buying and selling prices	I am the administrator of the company	High	Sprint -2

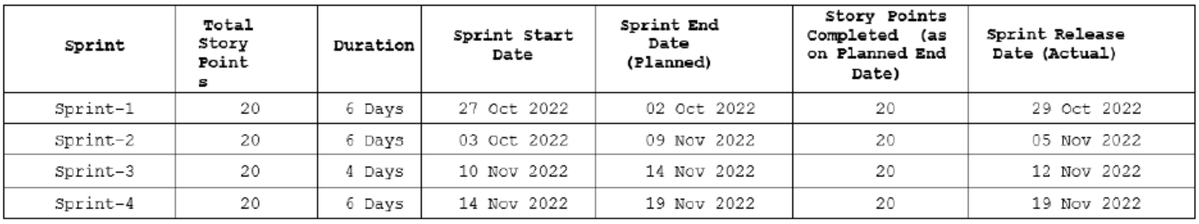
6.PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION



SPRINT	FUNCTIONAL REQUIREMENT (EPIC)	USER STORY NUMBER	DESCRIPTION	STORY POINTS	PRIORITY	TEAM MEMBERS
Sprint-1	Login (Job Seeker)	USN-1	User must login to look for a suitablejob	10	High	Bavithra.v, Divya Shree.R.R
Sprint-1	Login (Organization)	USN-2	Organization can find a candidate for their Organization	10	High	Divya Shree.R.R
Sprint-2	Dashboard	USN-3	User can create their job profile & look for job by attending eligible test. If usergets required mark, they will be invited to participate ininterview.	20	Medium	Bavithra.V
Sprint-3	Dashboard	USN-4	Organization can look for candidatebased on their skill set	20	Medium	Bavithra.V
Sprint-4	Chat BOT	USN-5	If User faces any difficulties,they can get help by Chabot	20	High	Divya Shree.R.R, V.Bavithra

6.2 SPRINT DELIVERABLES 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	27 Oct 2022	02 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	03 Oct 2022	09 Nov 2022	20	05 Nov 2022
Sprint-3	20	4 Days	10 Nov 2022	14 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'scalculate the team's averagevelocity (AV) per iteration unit (story points per day)

6.3 Report from JIRA

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

PNT2022TMID12203

7.CODING AND SOLUTIONING

7.1 FEATURE 1

JOB SEEKER

```
      <!DOCTYPE html>


<html lang="en">
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

<style>
.rus{

border-radius: 0%
};

.black{ color:black;
}
</style>

</head>
<body>

<!-------========================================================-->
<nav class="navbar navbar-inverse rus">
<div class="container-fluid">
<div class="navbar-header">
<a class="navbar-brand" href="#">Skill and Job Recommender</a>
</div>

<ul class="nav navbar-nav navbar-right">
<li><a href="#"><span class="glyphicon glyphicon-log-in"></span> Log out</a></li>
</ul>
</div>
</nav>
<!-------========================================================-->

<div class="container">
<ul class="nav nav-tabs">
<li class="active"><a data-toggle="tab" href="#home">Profile</a></li>
<li><a data-toggle="tab" href="#menu1">Eligibility Test</a></li>
<li><a data-toggle="tab" href="#menu2">Find Job</a></li>

</ul>
```

```
<div class="tab-content">
<div id="home" class="tab-pane fade in active">
<h3>PROFILE</h3>
<hr>

<form>
<div class="alert alert-danger">
<div class="form-group">
<label for="name"><span clss="black">Name: </span></label>
<input type="text" class="form-control" id="name" value="Mahadevan">
</div>
<div class="form-group">
<label for="email"><span class="black">Email Address </span></label>
<input type="text" class="form-control" id="email" value="Mahadevan@gmaiil.com">
</div>

<div class="form-group">
<label for="email"><span class="black">Mobile No </span></label>
<input type="text" class="form-control" id="email" value="9003824096">
</div>

<div class="form-group">
<label for="quali"><span class="black">What is your Nationality
</span></label>
<input type="text" class="form-control" id="quali" value="India">
</div>



<div class="form-group">
<label for="quali">What is your Qualification</label>
<input type="text" class="form-control" id="quali" value="BE">
</div>
 <div class="form-group">
<label for="quali">What year did you complete your bachelor Degree?</label>
<input type="text" class="form-control" id="quali" value="2023">
</div>

?</label>

<div class="form-group">
<label for="experience">How many years of work experience do you have

<input type="text" class="form-control" id="experience" value="0">
</div>


<div class="form-group">
<label for="place">Where have you graduated high school degree ?</label>
<input type="text" class="form-control" id="place" value="Government high school Ariyalur">
</div>
```

```
<div class="form-group">
<label for="SSLC">What is your SSLC Percentage :</label>
<input type="text" class="form-control" id="SSLC" value="90">
</div>

<div class="form-group">
<label for="HSC">What is your HSC Percentage :</label>
<input type="text" class="form-control" id="HSC" value="90">
</div>

<div class="form-group">
<label for="degree">What is your Percentage in bachelor Degree:</label>
<input type="text" class="form-control" id="degree" value="90.5">
</div>

<textarea class="form-control" id="aboutyou" rows="3" placeholder="Write few words about
yourself"></textarea> <br> <br>

<button type="submit" class="btn btn-primary btn-success">Save changes</button> <br> <br>
</div>
</form>
</div>

<div id="menu1" class="tab-pane fade">

<h3>Eligibility Test </h3> <hr>

<div class="alert alert-info">
<label>What function is used to display output on the screen?</label>
<div class="radio">
<label><input type="radio" name="print">printf()</label>
</div>
<div class="radio">
<label><input type="radio" name="scanf">scanf()</label>
</div>
<div class="radio">
<label><input type="radio" name="switch">switch</label>
</div>
</div> <hr>

<div class="alert alert-info">
<label>What function is used to display output on the screen?</label>
<div class="radio">
<label><input type="radio" name="print">printf()</label>
</div>
<div class="radio">
<label><input type="radio" name="scanf">scanf()</label>
</div>
<div class="radio">
<label><input type="radio" name="switch">switch</label>
</div>
</div> <hr>
```

```
<div class="alert alert-info">
<label>What function is used to display output on the screen?</label>
<div class="radio">
<label><input type="radio" name="print">printf()</label>
</div>
<div class="radio">
<label><input type="radio" name="scanf">scanf()</label>
</div>
<div class="radio">
<label><input type="radio" name="switch">switch</label>
</div>
</div> <hr>

<div class="alert alert-info">

<label>What function is used to display output on the screen?</label>
<div class="radio">
<label><input type="radio" name="print">printf()</label>
</div>
<div class="radio">
<label><input type="radio" name="scanf">scanf()</label>
</div>
<div class="radio">
<label><input type="radio" name="switch">switch</label>
</div>
</div> <hr>

<div class="alert alert-info">
<label>What function is used to display output on the screen?</label>
<div class="radio">
<label><input type="radio" name="print">printf()</label>
</div>
<div class="radio">
<label><input type="radio" name="scanf">scanf()</label>
</div>
<div class="radio">
<label><input type="radio" name="switch">switch</label>
</div>
</div><hr>

<div class="alert alert-info">
<label>What function is used to display output on the screen?</label>
<div class="radio">
<label><input type="radio" name="print">printf()</label>
</div>
<div class="radio">
<label><input type="radio" name="scanf">scanf()</label>
</div>
<div class="radio">
<label><input type="radio" name="switch">switch</label>
</div>
</div>
```

```
<button type="submit" class="btn btn-primary btn-success">Submit the test</button> <br>
<br>
</div>


<div id="menu2" class="tab-pane fade">
<h3>Find the job</h3> <hr>
<br>
 job">

<div class="container">
<!  >
<form>
<div class="input-group">
<input type="text" class="form-control" id="myInput" placeholder="Look for a

<div class="input-group-btn">
<button class="btn btn-default" type="submit">
<i class="glyphicon glyphicon-search"></i>
</button>
</div>
</div>
</form> <br> <br>


<!  >
<div style="width: 600px;">
<!-- Left-aligned -->
<div class="media">
<div class="media-left">
<img src="images/img_avatar1.png" class="media-object" style="width:60px">
</div>
<div class="media-body">
<h4 class="media-heading">Java Developer</h4>
<h5 class="media-heading"> Company Name : JK Software Solution</h4>
<p>Job Location : Chennai</p>
<p> Salary : 45000</p>
<button type="submit" class="btn btn-primary btn-success">Apply </button>
<br> <hr>
</div>
</div>
</div>

<br>
```

```
<div style="width: 600px;">
<!-- Left-aligned -->
<div class="media">
<div class="media-left">
<img src="images/img_avatar1.png" class="media-object" style="width:60px">
</div>
<div class="media-body">
<h4 class="media-heading">.NET Developer</h4>
<h5 class="media-heading"> Company Name : JK Software Solution</h4>

<br> <hr>
 <p>Job Location : Chennai</p>
<p> Salary : 45000</p>
<button type="submit" class="btn btn-primary btn-success">Apply </button>
</div>
 </div>
</div>
<div style="width: 600px;">
<!-- Left-aligned -->
<div class="media">
<div class="media-left">
<img src="images/img_avatar1.png" class="media-object" style="width:60px">
</div>
<div class="media-body">
<h4 class="media-heading">PHP Developer</h4>
<h5 class="media-heading"> Company Name : JK Software Solution</h4>
<p>Job Location : Chennai</p>
<p> Salary : 45000</p>
<button type="submit" class="btn btn-primary btn-success">Apply </button>
<br> <hr>
</div>
</div>
</div>
<div style="width: 600px;">
<!-- Left-aligned -->
<div class="media">
<div class="media-left">
<img src="images/img_avatar1.png" class="media-object" style="width:60px">
</div>
<div class="media-body">
<h4 class="media-heading">UX /UI Designer</h4>
<h5 class="media-heading"> Company Name : JK Software Solution</h4>
<p>Job Location : Chennai</p>
<p> Salary : 45000</p>
<button type="submit" class="btn btn-primary btn-success">Apply </button>
<br> <hr>
</div>
</div>
</div>
</div>
```

```
<script>
    $(document).ready(function(){
    $("#myInput").on("keyup", function() { var value = $(this).val().toLowerCase();
    $("#myDIV *").filter(function() {
    $(this).toggle($(this).text().toLowerCase().indexOf(value) > -1)
    });
    });
    });
    </script>

    </body>

    </html>
```

PNT2022TMID12203

BACK END FEATURE 1

```
from flask import *#importing flask (Install it using python -m pip install flask) import
os
import ibm_db import bcrypt
#import send_from_directory
app = Flask( name ) #initialising flask app.secret_key = ''
PEOPLE_FOLDER = os.path.join('static', 'people_photo')

import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=54a2f15b-5c0f-46df-8954-
7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;SECURITY=SSL;S
SLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=phb43134;PWD=mTTnIE45
Raj9A0rr",'','')


@app.route("/") #defining the routes for the home() funtion (Multiple routes can be used as
seen here)
@app.route("/home")

def dashboard():
return render_template("dashboard.html") #rendering our home.html contained within
/templates







def createjobseeker_profile(): if request.method == 'POST':
name = request.form['name'] email = request.form['email']
mobileno=request.form['mobileno'] nationality=request.form['nationality']
qualification=request.form['qualification'] passout=request.form['passout']
experience=request.form['experience"]; degree=request.form['degree'];
per_sslc=request.form['per_sslc']; per_hsc=request.form['per_hsc'];
per_degree=request.form['per_degree'];

tum = request.files['resume']
tum.save(os.path.join(app.config['UPLOAD_DIR'],tum.filename))



hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())

query = "SELECT EMAIL FROM JOBSEEKER WHERE EMAIL=?"
stmt = ibm_db.prepare(conn, query) ibm_db.bind_param(stmt,1,email) ibm_db.execute(stmt)
isUser = ibm_db.fetch_assoc(stmt)
```

```
if not isUser:
insert_sql = "INSERT INTO JOBSEEKER (name, email, mobileno, nationality, qualification,
passout, experience, degree,per_sslc,per_hsc,per_degree) 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, mobileno)
ibm_db.bind_param(prep_stmt, 4, nationality)
ibm_db.bind_param(prep_stmt, 5, qualification)
ibm_db.bind_param(prep_stmt, 6, passout)
ibm_db.bind_param(prep_stmt, 7, experience)
ibm_db.bind_param(prep_stmt, 8, degree)
ibm_db.bind_param(prep_stmt, 9, per_sslc)
ibm_db.bind_param(prep_stmt, 10, per_hsc)
ibm_db.bind_param(prep_stmt, 11, per_degree) ibm_db.execute(prep_stmt)
return render_template('dashboard.html')


return render_template('dashboard.html',name='Home')





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

Feature 2

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
  <style>
    .rus{
        border-radius: 0%
    };
    .black{
      color:black;
    }
</style>
</head>
<body>
<nav class="navbar navbar-inverse rus">
    <div class="container-fluid">
      <div class="navbar-header">
```

```
        <a class="navbar-brand" href="#">Skill and Job Recommender</a>
      </div>
      <ul class="nav navbar-nav navbar-right">
        <li><a href="#"><span class="glyphicon glyphicon-log-in"></span> Log out</a></li>
      </ul>
    </div>
  </nav>
<div class="container">
  <ul class="nav nav-tabs">
    <li class="active"><a data-toggle="tab" href="#home">Company Profile</a></li>
    <li><a data-toggle="tab" href="#menu2">Find candidate</a></li>
  </ul>
  <div class="tab-content">
    <div id="home" class="tab-pane fade in active">
      <h3>COMPANY PROFILE</h3>
      <hr>
      <form>
        <div class="alert alert-danger">
        <div class="form-group">
            <label for="name"><span clss="black">Company Name: </span></label>
            <input type="text" class="form-control" id="name" value="Mahadevan">
          </div>
          <div class="form-group">
```

```
            <label for="email"><span class="black">Company Email
Address </span></label>

            <input type="text" class="form-control" id="email"
value="Mahadevan@gmaiil.com">

          </div>

          <div class="form-group">

            <label for="email"><span class="black">Mobile No
</span></label>

            <input type="text" class="form-control" id="email"
value="9003824096">

          </div>

          <div class="form-group">

            <label for="quali"><span
class="black">Location</span></label>

            <input type="text" class="form-control" id="quali"
value="India">

          </div>

          <textarea class="form-control" id="aboutyou" rows="3"
placeholder="Write few words about company"></textarea> <br>
<br>



          <button type="submit" class="btn btn-primary btn-
success">Save changes</button> <br> <br>

          </div>

        </form>

    </div>




   <div id="menu2" class="tab-pane fade">

      <h3>Find the Condidate</h3> <hr>

      <br>
      <div class="container">
      <!.........................................................................................................
>
      <form>
      <div class="input-group">
       <input type="text" class="form-control" id="myInput"
placeholder="Look for a candidate">
<div class="input-group-btn">
                           <button class="btn btn-default"
type="submit">
                              <i class="glyphicon glyphicon-
search"></i>
                           </button>
                         </div>
                      </div>
                    </form> <br> <br>
      <!.....................................................................>
      <div style="width: 600px;">
                         <!-- Left-aligned -->
                         <div class="media">
                           <div class="media-left">
                              <img src="images/img_avatar1.png"
class="media-object" style="width:60px">
                           </div>
                           <div class="media-body">
                              <h4 class="media-heading">Name :
Mahadevan</h4>
          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>
```

```
          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>

                         </div>

                       </div>

                  </div>

      <br>

      <div style="width: 600px;">

                       <!-- Left-aligned -->

                       <div class="media">

                         <div class="media-left">

                            <img src="images/img_avatar1.png"
class="media-object" style="width:60px">

                         </div>

                         <div class="media-body">

                            <h4 class="media-heading">Name :
Manikandan</h4>

          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>

          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>

                         </div>

                       </div>

                  </div>  <br>

      <div style="width: 600px;">

                       <!-- Left-aligned -->

                       <div class="media">

                         <div class="media-left">

                            <img src="images/img_avatar1.png"
class="media-object" style="width:60px">

                         </div>
```

```
                           <div class="media-body">

                              <h4 class="media-heading">Name :
Kavi</h4>

          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>

          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>

                           </div>

                         </div>

                    </div>  <br>


      <div style="width: 600px;">

                         <!-- Left-aligned -->

                         <div class="media">

                           <div class="media-left">

                              <img src="images/img_avatar1.png"
class="media-object" style="width:60px">

                           </div>

                           <div class="media-body">

                              <h4 class="media-heading">Name :
Ram</h4>

          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>

          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>

                           </div>

                         </div>

                    </div>  <br>

    </div>

  </div>
```

```
</div>

<script>

  $(document).ready(function(){

    $("#myInput").on("keyup", function() {

    var value = $(this).val().toLowerCase();

    $("#myDIV *").filter(function() {

      $(this).toggle($(this).text().toLowerCase().indexOf(value)
> -1)

    });

    });

  });

  </script>

</body>

</html>
```

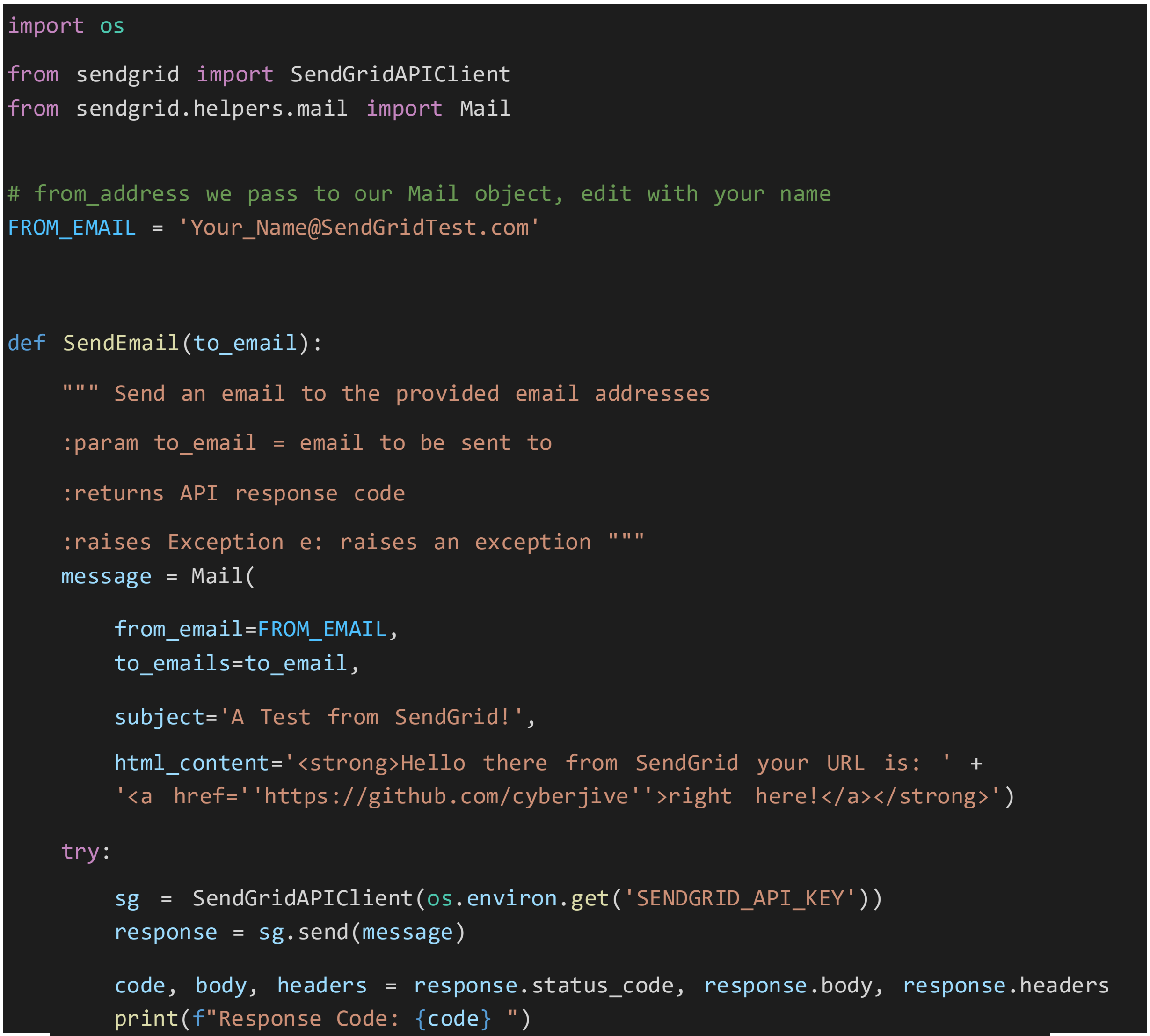
CHATBOT :

Chatbot has been implemented to provide assistance .

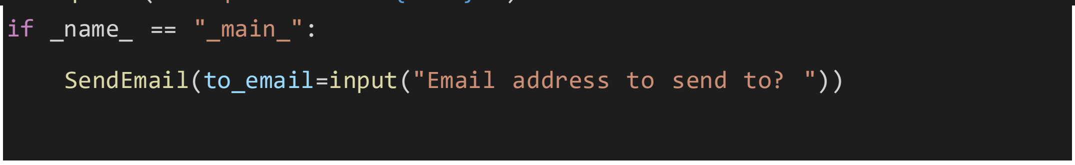
```
window.watsonAssistantC
hatOptions = {
                          integrationID: "d73273d3-3f44-4304-
                          84ee-8fd243016d1d", // The ID of this
                          integration.
                              region: "jp-tok", // The region
                          your integration is hosted in.
                          serviceInstanceID: "81229104-ee6b-46da-
                          ac1c-67ede110663a", // 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.clie
ntVersion || 'latest') +
"/WatsonAssistantChatEntry.js";
    document.head.appendChild(t);
  });
```

7.3 Database Schema(if Applicable) :



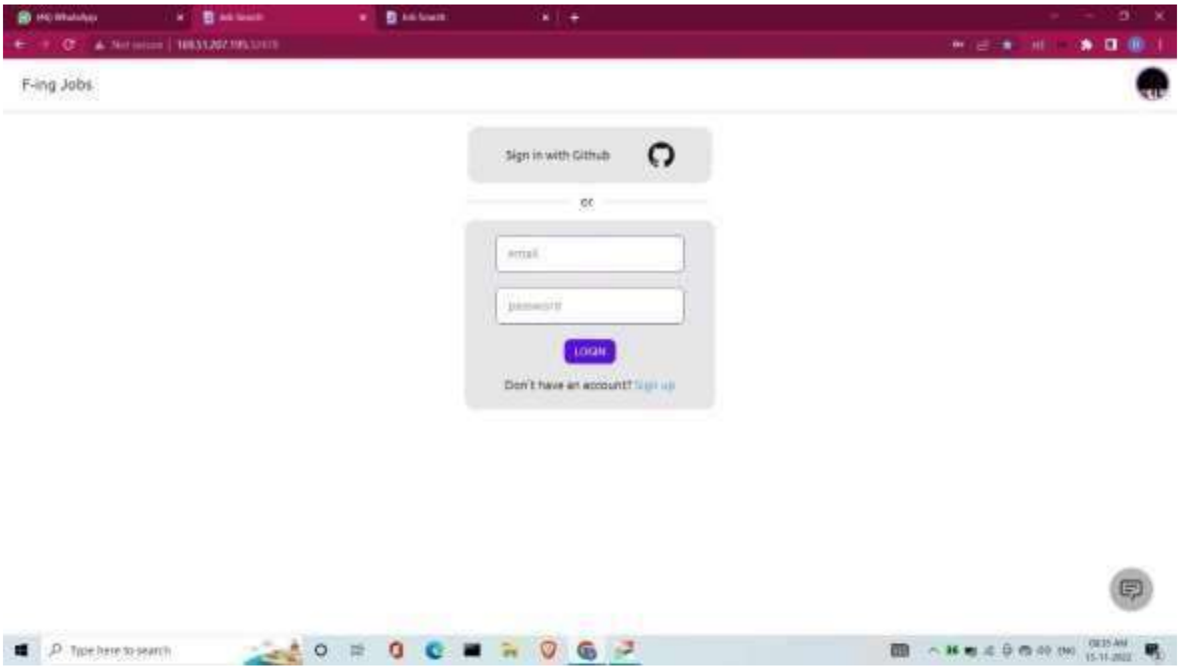
```
import os

from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail


# from_address we pass to our Mail object, edit with your name
FROM_EMAIL = 'Your_Name@SendGridTest.com'



def SendEmail(to_email):

    """ Send an email to the provided email addresses

    :param to_email = email to be sent to

    :returns API response code

    :raises Exception e: raises an exception """
    message = Mail(

        from_email=FROM_EMAIL,
        to_emails=to_email,

        subject='A Test from SendGrid!',

        html_content='<strong>Hello there from SendGrid your URL is: ' +
        '<a href=''https://github.com/cyberjive''>right here!</a></strong>')

    try:

        sg = SendGridAPIClient(os.environ.get('SENDGRID_API_KEY'))
        response = sg.send(message)

        code, body, headers = response.status_code, response.body, response.headers
        print(f"Response Code: {code} ")
```

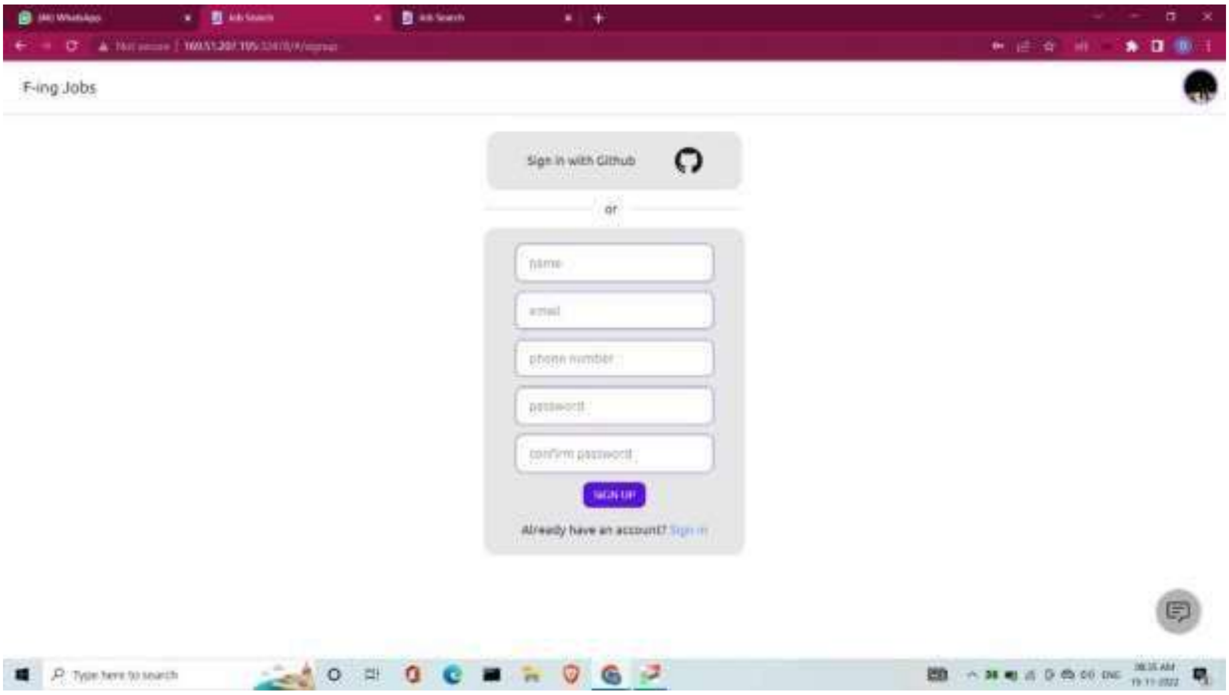


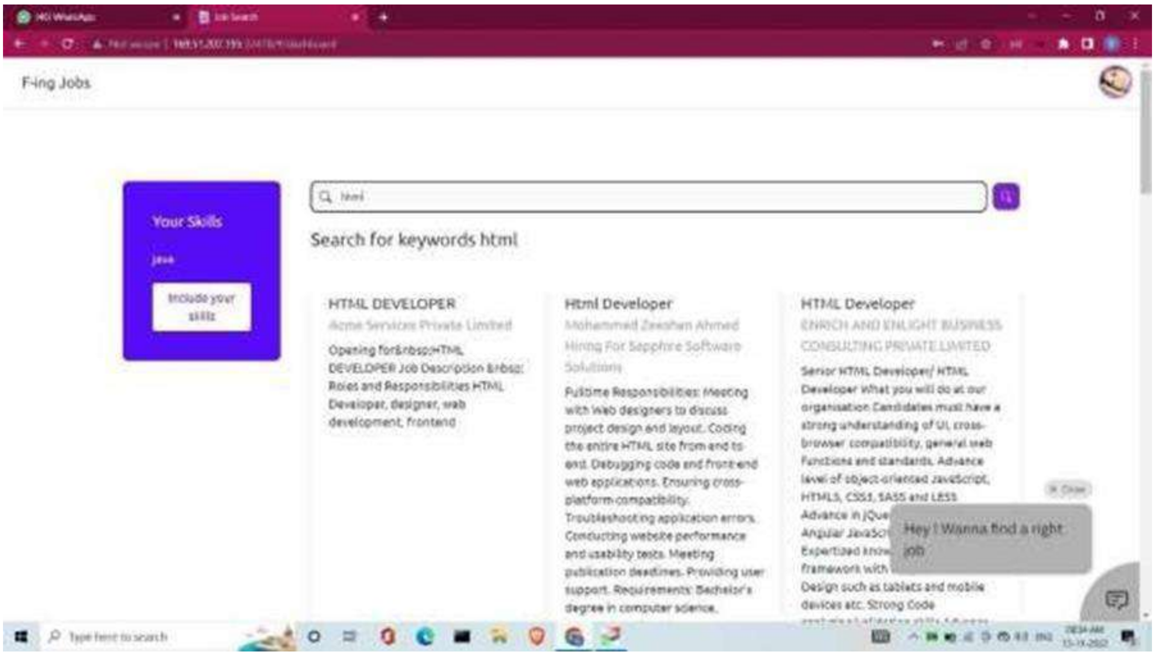
```
if _name_ == "_main_":

    SendEmail(to_email=input("Email address to send to? "))
```

8. TESTING

8.1 TEST CASES:







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 JavaScr Expertized know framework with Design such as tablets and mobile devices etc. Strong Code

Hey ! Wanna find a right job

9 RESULTS:

The project has been completed as we expected.

We ensured that Database was designed and well connectedto our project.

The Expected results were gotten.

10 ADVANTAGES & DISADVANTAGES:

ADVANTAGES

- Person who looks for a job can easily find a suitable jobbased on their skill set.
- Person can check their eligibility by attending eligibilitytest.
- Most of the Recruiters find the suitable person based onthe scores they have gotten in the eligibility.

DISADVANTAGES

- Person Job May get technical difficulty while taking the eligibility.
- Job seeker may have trouble to contact recruiters directly.

11 CONCLUSION

The application has been developed to make job searcheasier.

The application that we have developed is user friendly.

User can find a job based on their skillset in the short periodof time. The jobseeker certainly get benefit by using this application.

In the addition, Chatbot Has been implemented with the helpof IBM what's on. The chatbot helps jobseeker and organization when they experience the difficulties**.**

12 FUTURE SCOPE

The linked in the well known application to find a job and stay

connected with professional and organization.

The job seekers and organization use linked in to find a job.

In the future, There are lots of possibilities to enhance our project similar to linked in.

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 theimpact 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 APPENDIX

SOURCE CODE

Sprint 1:

Index.html

```
<!DOCTYPEhtml>
                <html lang="en">
                  <head>
                    <meta charset="utf-8">
                     <meta name="viewport"
                content="width=device-width, initial-scale=1">
                    <title>Log-in Page </title>
                     <link href="css/bootstrap.min.css"
                rel="stylesheet">
                     <script src="chat.js"></script>
                  </head>
                  <style>
                     .pad{
                          padding-top: 40%;
                     }
                     .bold{
                          font-weight: bolder;
                     }
                     .backgroundcolour{
                          background-color: beige;
                     }
                     .font{
                          font-weight: bolder;
                          color:whitesmoke;
                     }
                  </style>
                     <body>

                          <div class="container">
                               <div class="row">
                                    <div class="col-sm-6">

                                    </div>
                                    <div class="col-sm-6">
```

```
                         <div class="pad ">



                                    <center><img
src="images/logo.png" class="img-responsive"
width="20%" height="20%"> </center> <HR>
                                    <form>
                                         <div
class="input-group">
                                           <span
class="input-group-addon"><i  class="glyphicon
glyphicon-user"></i></span>
                                           <input
id="email" type="text" class="form-control"
name="email" placeholder="Enter Your Registered
Email ID" required>
                                         </div>
                                         <br>
                                         <div
class="input-group">
                                           <span
class="input-group-addon"><i  class="glyphicon
glyphicon-lock"></i></span>
                                           <input
id="password" type="password" class="form-
control" name="password" placeholder="Enter Your
Password" required>
                    </div>

                <hr>

<div class="row">
<div class="col-sm-6">
<button type="submit" id="" class="btn btn-
success btn-block"> <span class="font">LOG IN
</span></button>

     </div>

     <div class="col-sm-6">

      <button id="" class="btn btn-danger btn-
block"><a href="forget_password.html"><span
class="font">FORGET PASSWORD
```

```
</span></a></button>

     </div>
                                        </div>
<br>


                                   </form>
                                   <a
href="signup.html"> <button type="submit" id=""
class="btn btn-primary btn-block"><span
class="font">CREATE AN ACCOUNT </span> </button>
</a>
                         </div>


                    </div>

               </div>
          </div>

          <script
src="js/jquery.min.js"></script>
          <script
src="js/bootstrap.min.js"></script>
          <script src="chat.js"></script>
     </body>
</html>
```

FORGET-PASSWORD.html

```
<!DOCTYPEhtml>

                <html lang="en">
                  <head>
                    <meta charset="utf-8">
                     <meta name="viewport"
                content="width=device-width, initial-scale=1">
                    <title>Responsive Sample</title>
                     <link href="css/bootstrap.min.css"
                rel="stylesheet">
                  </head>
                  <style>
                     .pad{
                          padding-top: 20%;
                          padding-left: 20%;
```

```
          padding-right: 20%;
     }
     .bold{
          font-weight: bolder;
     }
  </style>
     <body>
          <div class="container">

               <div class="pad">
                <form>
                    <center> <H3
class="bold">FORGET PASSWORD</H3></center> <hr>
                    <div class="form-group">
                      <label for="email">Email
address:</label>
                      <input type="email"
class="form-control" id="email"
placeholder="Enter Your Registered Email
Addtress">
                    </div>
                    <div class="form-group">
                      <label
for="pwd">OTP</label>
                      <input type="password"
class="form-control" id="pwd" placeholder="Enter
your OTP ">
                    </div>

                    <button type="submit"
class="btn btn-default">Recover
Password</button>
                  </form>
               </div>
          </div>

          <script
src="js/jquery.min.js"></script>
          <script
src="js/bootstrap.min.js"></script>
     </body>
</html>
```

SIGN UP.html

```
<!DOCTYPE
html>
           <html lang="en">
             <head>
               <meta charset="utf-8">
                <meta name="viewport"
           content="width=device-width,
           initial-scale=1">
               <title>Creating New Account
           </title>
                <link
           href="css/bootstrap.min.css"
           rel="stylesheet">
             </head>
             <style>
                .pad{
                     padding-top: 10%;
                   padding-bottom: 10%;
                }
                .bold{
                     font-weight: bolder;
                }
             </style>
                <body>

                     <div class="container">

                                    <div
           class="pad">
                                         <h3
           class="bold">CREATE AN ACCOUNT</h3>
                                         <HR>

                <form>

                                       <div
           class="form-group">


                <label for="name">First
           Name</label>

                <input type="name"
           class="form-control" id="name"
           placeholder="Enter Your First
           Name">
```

```
     </div>
                             <div
class="form-group">

     <label for="name">Last
Name</label>


     <input type="name"
class="form-control" id="name"
placeholder="Enter Your Last Name">

     </div>

     <div class="form-group">

     <label for="email">Email
address:</label>

     <input type="email"
class="form-control" id="email"
placeholder="Enter Your Email ID">

     </div>

     <div class="form-group">

     <label
for="pwd">Password:</label>

     <input type="password"
class="form-control" id="pwd"
placeholder="Enter Your Password">
     </div>
                             <div
class="form-group">


     <label for="repwd">Reenter
Password:</label>

     <input type="password"
class="form-control" id="repwd"
placeholder="Enter Your Password
Again">
```

```
     </div>



     <button type="button" id=""
class="btn btn-primary btn-block
btn-danger">CREATE AN
ACCOUNT</button>

     <hr>

     <button type="submit"
id="login_already" class="btn btn-
primary btn-block btn-success">Do
you have an Account Already
?</button>


     </div>

     </form>
                         </div>

          </div>


          <script
src="js/jquery.min.js"></script>
          <script
src="js/bootstrap.min.js"></script>
          <script
src="chat.js"></script>
     </body>
</html>
```

BACK END :

```
import os
import ibm_db
import bcrypt
#import send_from_directory
app = Flask(__name__) #initialising flask
app.secret_key = ''
PEOPLE_FOLDER = os.path.join('static', 'people_photo')
import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=54a2f15b-5c0f-
46df-8954-
7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;P
ORT=32733;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRoo
tCA.crt;PROTOCOL=TCPIP;UID=phb43134;PWD=mTTnIE45Raj9A0rr",'',
'')
@app.route("/") #defining the routes for the home() funtion
(Multiple routes can be used as seen here)
@app.route("/home")
def home():
    return render_template("home.html") #rendering our
home.html contained within /templates
@app.route("/login")
def login():
    if request.method == 'POST':
        email = request.form['email']
        password = request.form['password']
        if not email or not password:
            return render_template('login.html',error='Please
fill all fields')

        query = "SELECT * FROM USER WHERE email=?"
        stmt = ibm_db.prepare(conn, query)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        isUser = ibm_db.fetch_assoc(stmt)
        print(isUser,password)
        if not isUser:
            return
render_template('login.html',error='Invalid Credentials')
```

```
        isPasswordMatch =
bcrypt.checkpw(password.encode('utf-
8'),isUser['PASSWORD'].encode('utf-8'))
        if not isPasswordMatch:
            return
render_template('login.html',error='Invalid Credentials')
        session['email'] = isUser['EMAIL']
        return redirect(url_for('user_dashboard'))
    return render_template('login.html',name='Home')
app.config["UPLOAD_DIR"] = "uploads"
@app.route("/register",methods=['GET','POST'])
def register():
  if request.method == 'POST':
    email = request.form['email']
    password = request.form['password']
    firstname=request.form['firstname']
    middlename=request.form['middlename']
    lastname=request.form['lastname']
    dob=request.form['dob']
    age=request.form['age']
    drno=request.form['drno']
    landmark1=request.form['lnmk1']
    landmark2=request.form['lnmk2']
    village =request.form['village']
    district=request.form['district']
    pincode=request.form['pincode']
    country=request.form['country']
    country_code=request.form['countryCode']
    phone=request.form['phone']
    schlname10=request.form['schlname10']
    maxmark10=request.form['maxmark10']
    actmark10=request.form['actmark10']
    schlname212=request.form['schlname212']
    maxmark212=request.form['maxmark212']
    actmark212=request.form['actmark212']
    clgname=request.form['clgname']
    clgmaxmark=request.form['clgmaxmark']
    clgactmark=request.form['clgactmark']
    stream=request.form['stream']
    graduation=request.form['graduation']
    grayear=request.form['grayear']
    branch=request.form['branch']
    skills=request.form['form[]']
    q1=request.form['Q1']
    q2=request.form['Q2']
    q3=request.form['Q3']
    q4=request.form['Q4']
    q5=request.form['Q5']
    position=request.form['position']
    experience=request.form['expyears']
    designation=request.form['designation']
    tum = request.files['resume']

tum.save(os.path.join(app.config['UPLOAD_DIR'],tum.filename))

    hash=bcrypt.hashpw(password.encode('utf-
8'),bcrypt.gensalt())
    query = "SELECT EMAIL FROM JOBSEEKER WHERE EMAIL=?"
    stmt = ibm_db.prepare(conn, query)
    ibm_db.bind_param(stmt,1,email)
    ibm_db.execute(stmt)
    isUser = ibm_db.fetch_assoc(stmt)

    if not is User:
      insert_sql = "INSERT INTO JOBSEEKER (EMAIL, PASSWORD,
FIRSTNAME, MIDDLENAME, LASTNAME, DOB, AGE, DOOR_NO,
LANDMARK1, LANDMARK2,VILLAGE, DISTRICT, PINCODE, COUNTRY,
COUNTRY_CODE, PHONE, SCHOOL_NAME, SC_MAX_MARK,
SC_ACT_MARK,HSC_SCHOOL_NAME,HSC_MAX_MARK,
HSC_ACT_MARK,COLLEGE_NAME,COLLEGE_MAX,COLLEGE_ACT, STREAM,
GRADUATION, GRADUATION_YEAR, BRANCH, SKILLS, PRESENTATION,
HACKATHON, CODING, WON, PRIZE_WON, POSITION,
EXPYEARS,DESIGNATION, RESUME) VALUES
(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
?,?,?,?,?,?,?,?,?)"
      prep_stmt = ibm_db.prepare(conn, insert_sql)
      ibm_db.bind_param(prep_stmt, 1, email)
      ibm_db.bind_param(prep_stmt, 2, password)
      ibm_db.bind_param(prep_stmt, 3, firstname)
      ibm_db.bind_param(prep_stmt, 4, middlename)
      ibm_db.bind_param(prep_stmt, 5, lastname)
      ibm_db.bind_param(prep_stmt, 6, dob)
      ibm_db.bind_param(prep_stmt, 7, age)
      ibm_db.bind_param(prep_stmt, 8, drno)
      ibm_db.bind_param(prep_stmt, 9, landmark1)
      ibm_db.bind_param(prep_stmt, 10, landmark2)
      ibm_db.bind_param(prep_stmt, 11, village)
      ibm_db.bind_param(prep_stmt, 12, district)
      ibm_db.bind_param(prep_stmt, 13, pincode)
      ibm_db.bind_param(prep_stmt, 14, country)
      ibm_db.bind_param(prep_stmt, 15, country_code)
      ibm_db.bind_param(prep_stmt, 16, phone)
      ibm_db.bind_param(prep_stmt, 17, schlname10)
      ibm_db.bind_param(prep_stmt, 18, maxmark10)
      ibm_db.bind_param(prep_stmt, 19, actmark10)
      ibm_db.bind_param(prep_stmt, 20, schlname212)
      ibm_db.bind_param(prep_stmt, 21, maxmark212)
      ibm_db.bind_param(prep_stmt, 22, actmark212)
      ibm_db.bind_param(prep_stmt, 23, clgname)
      ibm_db.bind_param(prep_stmt, 24, clgmaxmark)
      ibm_db.bind_param(prep_stmt, 25, clgactmark)
      ibm_db.bind_param(prep_stmt, 26, stream)
      ibm_db.bind_param(prep_stmt, 27, graduation)
      ibm_db.bind_param(prep_stmt, 28, grayear)
      ibm_db.bind_param(prep_stmt, 29, branch)
      ibm_db.bind_param(prep_stmt, 30, skills)
      ibm_db.bind_param(prep_stmt, 31, q1)
      ibm_db.bind_param(prep_stmt, 32, q2)
      ibm_db.bind_param(prep_stmt, 33, q3)
      ibm_db.bind_param(prep_stmt, 34, q4)
      ibm_db.bind_param(prep_stmt, 35, q5)
      ibm_db.bind_param(prep_stmt, 36, position)
      ibm_db.bind_param(prep_stmt, 37, experience)
      ibm_db.bind_param(prep_stmt, 38, designation)
      ibm_db.bind_param(prep_stmt, 39, tum)
      ibm_db.execute(prep_stmt)
      return render_template('register.html')
    else:
      return render_template('register.html',error='Invalid
Credentials')
  return render_template('register.html',name='Home')
@app.route("/orgregister")
def orgregister():
    return render_template("orgregister.html") #rendering our
register.html contained within /templates
@app.route("/user_dashboard")
def user_dashboard():
    return render_template("user_dashboard.html")
@app.route('/logout')
def logout():
    session.pop('email',   None)
    return redirect(url_for('login'))
if __name__== "__main__": #checking if__name__'s value is
'__main__'.__name__is an python environment variable who's
value will always be '__main__' till this is the first
```

```
instatnce of app.py running
    app.run(debug=True,port=8080) #running flask (Initalised
on line 4)
```

SPRINT 2

Index.html

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8">

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

    <title>Log-in Page </title>

     <link href="css/bootstrap.min.css" rel="stylesheet">

     <script src="chat.js"></script>

  </head>

  <style>

     .pad{

          padding-top: 40%;

     }

     .bold{

          font-weight: bolder;
    }

    .backgroundcolour{
          background-color: beige;
    }

    .font{

          font-weight: bolder;
          color:whitesmoke;
    }

</style>

    <body>
          <br>

          <div class="container">



                <div class="row">
                      <div class="col-sm-6">

                            <div class="pad ">
```

```
                              <center><img
src="images/ORGANIZATION_LOGO.png"  class="img-responsive"
width="20%" height="20%"> </center> <BR>

                              <form>

                                  <div class="input-group">

                                    <span class="input-group-
addon"><i class="glyphicon glyphicon-user"></i></span>

                                    <input id="email"
type="text" class="form-control" name="email" placeholder="Enter
Your Organization Registered Email ID" >

                                  </div>



                                  <br>



                                  <div class="input-group">

                                    <span class="input-group-
addon"><i class="glyphicon glyphicon-lock"></i></span>

                                    <input id="password"
type="password" class="form-control" name="password"
placeholder="Enter Your Password">

                                  </div>



                                  <hr>



                                  <div class="row">

                                  <div class="col-sm-6">

    <button type="submit" id="" class="btn btn-success btn-
block"> <span class="font">LOG IN </span></button>

    </div>
```

```
     <div class="col-sm-6">

      <button  id="" class="btn btn-danger btn-block">
<a href=""><span class="font">FORGET PASSWORD
</span></a></button>

     </div>

                              </div> <br>



                         </form>



<ahref="Organization/Create_Account/org_create_account.html">
<button type="submit" id="" class="btn btn-primary btn-
block"><span class="font">CREATE AN ACCOUNT </span> </button>
</a>

                    </div>




                    </div>

                    <div class="col-sm-6">



                         <div class="pad ">
```

```
                                   <center><img
src="images/logo.png" class="img-responsive" width="20%"
height="20%"> </center> <br>

                                   <form>

                                        <div class="input-
group">

                                          <span class="input-
group-addon"><i class="glyphicon glyphicon-user"></i></span>

                                          <input id="email"
type="text" class="form-control" name="email" placeholder="Enter
Your Registered Email ID" required>

                                        </div>



                                        <br>



                                        <div class="input-
group">

                                          <span class="input-
group-addon"><i class="glyphicon glyphicon-lock"></i></span>

                                          <input id="password"
type="password" class="form-control" name="password"
placeholder="Enter Your Password" required>

                                        </div>



                                        <hr>



                                        <div class="row">

                                             <div class="col-sm-
6">
```

```
                                                              <button
type="submit" id="" class="btn btn-success btn-block"> <span
class="font">LOG IN </span></button>

                                                         </div>

                                                         <div class="col-sm-
6">

           <button  id="" class="btn btn-danger btn-block">

<a href="forget_password.html">

<span class="font">FORGET PASSWORD </span></a></button>

                                                         </div>

                                                    </div> <br>



                                          </form>

                                          <a href="signup.html">
<button type="submit" id="" class="btn btn-primary btn-block">

<span class="font">CREATE AN ACCOUNT </span> </button>

</a></div>

</div>

</div>

</div>

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

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

          <script src="chat.js"></script>

     </body>

</html>
```

FORGET-PASSWORD.html

```
<!DOCTYPEhtml>
                 <html lang="en">
                 <head>
                     <meta charset="utf-8">
                      <meta name="viewport"
                 content="width=device-width, initial-scale=1">
                     <title>Responsive Sample</title>
                      <link href="css/bootstrap.min.css"
                 rel="stylesheet">
                   </head>
                   <style>
                      .pad{
                           padding-top: 20%;
                           padding-left: 20%;
                           padding-right: 20%;
                      }
                      .bold{
                           font-weight: bolder;
                      }
                   </style>
                      <body>
                           <div class="container">

                                <div class="pad">
                                 <form>
                                     <center> <H3
                 class="bold">FORGET PASSWORD</H3></center> <hr>
                                     <div class="form-group">
                                       <label for="email">Email
                 address:</label>
                                       <input type="email"
                 class="form-control" id="email"
                 placeholder="Enter Your Registered Email
                 Addtress">
                                     </div>
                                     <div class="form-group">
                                       <label
                 for="pwd">OTP</label>
                                       <input type="password"
                 class="form-control" id="pwd" placeholder="Enter
                 your OTP ">
```

```
                    </div>

                    <button type="submit"
class="btn btn-default">Recover
Password</button>
                 </form>
               </div>
          </div>

          <script
src="js/jquery.min.js"></script>
          <script
src="js/bootstrap.min.js"></script>
     </body>
</html>
```

SIGNUP.html

```
<!DOCTYPE
html>
```

```
<html lang="en">

  <head>

    <meta charset="utf-8">

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

    <title>Creating New Account
</title>

     <link
href="css/bootstrap.min.css"
rel="stylesheet">

  </head>

  <style>

     .pad{

          padding-top: 10%;
```

```
            padding-bottom: 10%;
        }
        .bold{
             font-weight: bolder;
        }
    </style>
        <body>

             <div class="container">

                                 <div
  class="pad">
                                       <h3
 class="bold">CREATE AN ACCOUNT</h3>
                                       <HR>

        <form>
                                     <div
 class="form-group">


        <label for="name">First
 Name</label>

        <input type="name"
 class="form-control" id="name"
 placeholder="Enter Your First
 Name">

        </div>
                                     <div
 class="form-group">
```

```
     <label for="name">Last
Name</label>

     <input type="name"
class="form-control" id="name"
placeholder="Enter Your Last Name">

     </div>

     <div class="form-group">

     <label for="email">Email
address:</label>

     <input type="email"
class="form-control" id="email"
placeholder="Enter Your Email ID">

     </div>

     <div class="form-group">

     <label
for="pwd">Password:</label>

     <input type="password"
class="form-control" id="pwd"
placeholder="Enter Your Password">

     </div>

                             <div
class="form-group">

     <label for="repwd">Reenter
```

```
Password:</label>

     <input type="password"
class="form-control" id="repwd"
placeholder="Enter Your Password
Again">

     </div>




     <button type="button" id=""
class="btn btn-primary btn-block
btn-danger">CREATE AN
ACCOUNT</button>

     <hr>

     <button type="submit"
id="login_already" class="btn btn-
primary btn-block btn-success">Do
you have an Account Already
?</button>



     </div>

     </form>
                    </div>


          </div>
          <script
src="js/jquery.min.js"></script>

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

          <script
src="chat.js"></script>

     </body>

</html>
```

HOME PAGE :

Forgot password.html

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8">

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

    <title>Responsive Sample</title>

              <link href="css/bootstrap.min.css"
rel="stylesheet">

  </head>

  <style>

              .pad{

               padding-top: 20%;

               padding-left: 20%;

               padding-right: 20%;

              }

              .bold{

               font-weight: bolder;

              }

  </style>

              <body>

               <div class="container">


                   <div class="pad">
```

```
                    <form>

                        <center> <H3 class="bold">FORGET
PASSWORD</H3></center> <hr>

                        <div class="form-group">

                          <label for="email">Email
address:</label>

                          <input type="email" class="form-
control" id="email" placeholder="Enter Your Registered Email
Addtress">

                        </div>

                        <div class="form-group">

                          <label for="pwd">OTP</label>

                          <input type="password" class="form-
control" id="pwd" placeholder="Enter your OTP ">

                        </div>



                        <button type="submit" class="btn btn-
default">Recover Password</button>

                     </form>

                   </div>



              </div>



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

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

             </body>

</html>
```

Signup.html

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8">

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

    <title>Creating New Account </title>

              <link href="css/bootstrap.min.css"
rel="stylesheet">

  </head>

  <style>

              .pad{

               padding-top: 10%;

        padding-bottom: 10%;

              }


              .bold{

               font-weight: bolder;

              }


  </style>


              <body>


               <div class="container">
```

```
                              <div class="pad">
                                   <h3 class="bold">CREATE AN
ACCOUNT</h3>
                                   <HR>
                                   <form>
                            <div class="form-group">
                                             <label
for="name">First Name</label>
                                             <input type="name"
class="form-control" id="name" placeholder="Enter Your First
Name">
                                        </div>


                            <div class="form-group">
                                             <label
for="name">Last Name</label>
                                             <input type="name"
class="form-control" id="name" placeholder="Enter Your Last
Name">
                                        </div>


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

```
                                               <label
for="email">Email address:</label>

                                               <input type="email"
class="form-control" id="email" placeholder="Enter Your Email
ID">

                                          </div>

                                          <div class="form-group">

                                               <label
for="pwd">Password:</label>

                                               <input
type="password" class="form-control" id="pwd" placeholder="Enter
Your Password">

                                          </div>



                             <div class="form-group">

                                               <label
for="repwd">Reenter Password:</label>

                                               <input
type="password" class="form-control" id="repwd"
placeholder="Enter Your Password Again">

                                          </div>








                                               <button
type="button" id="" class="btn btn-primary btn-block btn-
danger">CREATE AN ACCOUNT</button>

                                               <hr>
```

```
                                                <button
type="submit" id="login_already" class="btn btn-primary btn-
block btn-success">Do you have an Account Already ?</button>



                                    </div>

                                    </form>

                               </div>


                  </div>



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

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

                  <script src="chat.js"></script>

                </body>

</html>
```

BACK ENDAPP.PY

```
from flask import *#importing flask (Install it using python -m pip install flask)

import os

import ibm_db

import bcrypt

#import send_from_directory

app = Flask(__name__) #initialising flask

app.secret_key = ''

PEOPLE_FOLDER = os.path.join('static', 'people_photo')


import ibm_dbconn = ibm_db.connect("DATABASE=bludb;HOSTNAME=54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=phb43134;PWD=mTTnIE45Raj9A0rr",'','')



@app.route("/") #defining the routes for the home() funtion (Multiple routes can be used as seen here)

@app.route("/home")


def home():

    return render_template("index/INDEX.html") #rendering our home.html contained within /templates
```

```
@app.route("/login")
def login():
    if request.method == 'POST':
        email = request.form['email']
        password = request.form['password']
        if not email or not password:
            return  render_template('login.html',error='Please fill all fields')

        query = "SELECT * FROM USER WHERE email=?"
        stmt = ibm_db.prepare(conn, query)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        isUser = ibm_db.fetch_assoc(stmt)
        print(isUser,password)

        if not isUser:
            return  render_template('login.html',error='Invalid Credentials')

        isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))

        if not isPasswordMatch:
            return  render_template('login.html',error='Invalid Credentials')
        session['email'] = isUser['EMAIL']
        return  redirect(url_for('user_dashboard'))

    return  render_template('login.html',name='Home')




app.config["UPLOAD_DIR"] = "uploads"
@app.route("/register",methods=['GET','POST'])
def register():
  if request.method == 'POST':
    email = request.form['email']
    password = request.form['password']
    firstname=request.form['firstname']
    middlename=request.form['middlename']
    lastname=request.form['lastname']
    dob=request.form['dob']
    age=request.form['age']
    drno=request.form['drno']
    landmark1=request.form['lnmk1']
    landmark2=request.form['lnmk2']
    village =request.form['village']
    district=request.form['district']
    pincode=request.form['pincode']
```

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

country_code=request.form['countryCode']

phone=request.form['phone']

schlname10=request.form['schlname10']

maxmark10=request.form['maxmark10']

actmark10=request.form['actmark10']

schlname212=request.form['schlname212']

maxmark212=request.form['maxmark212']

actmark212=request.form['actmark212']

clgname=request.form['clgname']

clgmaxmark=request.form['clgmaxmark']

clgactmark=request.form['clgactmark']

stream=request.form['stream']

graduation=request.form['graduation']

grayear=request.form['grayear']

branch=request.form['branch']

skills=request.form['form[]']

q1=request.form['Q1']

q2=request.form['Q2']

q3=request.form['Q3']

q4=request.form['Q4']

q5=request.form['Q5']

position=request.form['position']

experience=request.form['expyears']

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

```
    tum = request.files['resume']

tum.save(os.path.join(app.config['UPLOAD_DIR'],tum.filename))




    hash=bcrypt.hashpw(password.encode('utf-
8'),bcrypt.gensalt())


    query = "SELECT EMAIL FROM ORGANIZATION WHERE EMAIL=?"

    stmt = ibm_db.prepare(conn, query)

    ibm_db.bind_param(stmt,1,email)

    ibm_db.execute(stmt)

    isUser = ibm_db.fetch_assoc(stmt)


    if not isUser:

      insert_sql = "INSERT INTO ORGANIZATION (ORGANIZATION_NAME,
ORGANIZATION_ID, ORGANIZATION_ADDRESS,SET_PASSWORD,
REENTERPASSWORD) VALUES (?,?,?,?,?)"

      prep_stmt = ibm_db.prepare(conn, insert_sql)

      ibm_db.bind_param(prep_stmt, 1, ORGANIZATION_NAME)

      ibm_db.bind_param(prep_stmt, 2, ORGANIZATION_ID)

      ibm_db.bind_param(prep_stmt, 3, ORGANIZATION_ADDRESS)

      ibm_db.bind_param(prep_stmt, 4, SET_PASSWORD)

      ibm_db.bind_param(prep_stmt, 5, REENTERPASSWORD)

      ibm_db.execute(prep_stmt)
      return
render_template('Organization\Create_Account\org_create_account.
html')

    else:

      return
render_template('Organization\Create_Account\org_create_account.
html',error='Invalid Credentials')


  return  render_template('create_account.html',name='Home')




@app.route("/orgregister")

def orgregister():

    return  render_template("org_create_account.html")  #rendering
our register.html contained within /templates




@app.route("/user_dashboard")

def user_dashboard():


    return  render_template("user_dashboard.html")


@app.route('/logout')

def logout():

    session.pop('email',   None)

    return  redirect(url_for('login'))
```

```
if___name___== "__main___": #checking if___name__'s value is
'__main__'.___name___is an python environment variable who's
value will always be '__main__' till this is the first instatnce
of app.py running

    app.run(debug=True,port=8080) #running flask (Initalised on
line 4)




    hash=bcrypt.hashpw(password.encode('utf-
8'),bcrypt.gensalt())


    query = "SELECT EMAIL FROM JOBSEEKER WHERE EMAIL=?"

    stmt = ibm_db.prepare(conn, query)

    ibm_db.bind_param(stmt,1,email)

    ibm_db.execute(stmt)

    isUser = ibm_db.fetch_assoc(stmt)


    if not isUser:

      insert_sql = "INSERT INTO JOBSEEKER (FIRSTNAME,
LASTNAME,EMAIL_ADDRESSS,PWD,REPWD=?"

      prep_stmt = ibm_db.prepare(conn, insert_sql)

      ibm_db.bind_param(prep_stmt, 1, firstname)

      ibm_db.bind_param(prep_stmt, 2, lastname)

      ibm_db.bind_param(prep_stmt, 3, email_address)

      ibm_db.bind_param(prep_stmt, 4, pwd)

      ibm_db.bind_param(prep_stmt, 5, repwd)
```

```
        ibm_db.execute(prep_stmt)

        return render_template('Job Seeker\signup.html')

      else:

        return render_template('Job
  Seeker\signup.html',error='Invalid Credentials')


    return render_template('signup.html',name='Home')
```

JAVASCRIPT:

```
window.watsonAssistantCh
atOptions= {
                          integrationID: "d73273d3-3f44-
                      4304-84ee-8fd243016d1d", // The ID of
                      this integration.
                          region: "jp-tok", // The region
                      your integration is hosted in.
                          serviceInstanceID: "81229104-ee6b-
                      46da-ac1c-67ede110663a", // 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.cli
                      entVersion || 'latest') +
                      "/WatsonAssistantChatEntry.js";
                          document.head.appendChild(t);
                        });
```

Verify.py

```
from flask import request

from flask_restful  import  Resource

from utils.tokener import confirm_token

from utils.dbQuery import insertQuery

class Verify(Resource):

data=request.json

token=data["token"]

email=confirm_token(token)

if(not email):

return {"status":"Couldn't verify"},400

res=insertQuery("UPDATE user set verified=True where

email=?",(email,))

if(not res):

return {"status":"Couldn't  Update"},400

return  {"status":"verified  successfully"},200
```

SPRINT :3

FRONT END

DASHBROAD.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>


  <style>
    .rus{

        border-radius: 0%
    };


    .black{
      color:black;

    }

</style>


</head>

<body>


<!--
=======================================================-->

<nav class="navbar navbar-inverse rus">

    <div class="container-fluid">

      <div class="navbar-header">

        <a class="navbar-brand" href="#">Skill and Job
Recommender</a>

      </div>


      <ul class="nav navbar-nav navbar-right">

        <li><a href="#"><span class="glyphicon glyphicon-log-
in"></span> Log out</a></li>

      </ul>

    </div>

  </nav>

<div class="container">

  <ul class="nav nav-tabs">

    <li class="active"><a data-toggle="tab"
href="#home">Profile</a></li>
```

```
  <li><a data-toggle="tab" href="#menu1">Eligibility
Test</a></li>

  <li><a data-toggle="tab" href="#menu2">Find Job</a></li>


 </ul>


 <div class="tab-content">

  <div id="home" class="tab-pane fade in active">

   <h3>PROFILE</h3>

   <hr>


   <form>

    <div class="alert alert-danger">

    <div class="form-group">

      <label for="name"><span clss="black">Name:
</span></label>

      <input type="text" class="form-control" id="name"
value="Mahadevan">

     </div>

     <div class="form-group">

      <label for="email"><span class="black">Email Address
</span></label>

      <input type="text" class="form-control" id="email"
value="Mahadevan@gmaiil.com">

     </div>


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

```
            <label for="email"><span class="black">Mobile No
</span></label>

            <input type="text" class="form-control" id="email"
value="9003824096">

          </div>


          <div class="form-group">

            <label for="quali"><span class="black">What is your
Nationality </span></label>

            <input type="text" class="form-control" id="quali"
value="India">

          </div>

 <div class="form-group">

            <label for="quali">What is your
Qualification</label>

            <input type="text" class="form-control" id="quali"
value="BE">

          </div>


          <div class="form-group">

            <label for="quali">What year did you complete your
bachelor Degree?</label>

            <input type="text" class="form-control" id="quali"
value="2023">

           </div>
          <div class="form-group">

            <label for="experience">How many years of work
experience do you have ?</label>

            <input type="text" class="form-control"
id="experience" value="0">

          </div>


          <div class="form-group">

            <label for="place">Where have you graduated high
school degree ?</label>

            <input type="text" class="form-control" id="place"
value="Government high school Ariyalur">

          </div>


          <div class="form-group">

            <label for="SSLC">What is your SSLC Percentage
:</label>

            <input type="text" class="form-control" id="SSLC"
value="90">

          </div>


          <div class="form-group">

            <label for="HSC">What is your HSC Percentage
:</label>

            <input type="text" class="form-control" id="HSC"
value="90">

          </div>
          <div class="form-group">
            <label for="degree">What is your Percentage in
bachelor Degree:</label>
            <input type="text" class="form-control" id="degree"
value="90.5">
          </div>

          <textarea class="form-control" id="aboutyou" rows="3"
placeholder="Write few words about yourself"></textarea> <br>
<br>

          <button type="submit" class="btn btn-primary btn-
success">Save changes</button> <br> <br>
          </div>
        </form>
    </div>
    <div id="menu1" class="tab-pane fade">
      <h3>Eligibility Test </h3> <hr>

      <div class="alert alert-info">
        <label>What function is used to display output on the
screen?</label>
              <div class="radio">
                      <label><input type="radio"
name="print">printf()</label>
                      </div>
                      <div class="radio">
```

```
                   <label><input type="radio"
name="scanf">scanf()</label>
                   </div>
                   <div class="radio">
                   <label><input type="radio"
name="switch">switch</label>
                   </div>
     </div> <hr>


     <div class="alert alert-info">
       <label>What function is used to display output on the
screen?</label>
            <div class="radio">
                   <label><input type="radio"
name="print">printf()</label>
                   </div>
                   <div class="radio">
                   <label><input type="radio"
name="scanf">scanf()</label>
                   </div>
                   <div class="radio">
                   <label><input type="radio"
name="switch">switch</label>
                   </div>
     </div> <hr>


     <div class="alert alert-info">
```

```
        <label>What function is used to display output on the
screen?</label>

              <div class="radio">

                       <label><input type="radio"
name="print">printf()</label>

                       </div>

                       <div class="radio">

                       <label><input type="radio"
name="scanf">scanf()</label>

                       </div>

                       <div class="radio">

                       <label><input type="radio"
name="switch">switch</label>

                       </div>

      </div> <hr>


      <div class="alert alert-info">

        <label>What function is used to display output on the
screen?</label>

              <div class="radio">

                       <label><input type="radio"
name="print">printf()</label>

                       </div>

                       <div class="radio">

                       <label><input type="radio"
name="scanf">scanf()</label>

                       </div>

                       <div class="radio">
```

```
                      <label><input type="radio"
name="switch">switch</label>

                      </div>

      </div> <hr>


      <div class="alert alert-info">

        <label>What function is used to display output on the
screen?</label>

             <div class="radio">

                      <label><input type="radio"
name="print">printf()</label>

                      </div>

                      <div class="radio">

                      <label><input type="radio"
name="scanf">scanf()</label>

                      </div>

                      <div class="radio">

                      <label><input type="radio"
name="switch">switch</label>

                      </div>

      </div><hr>


      <div class="alert alert-info">

        <label>What function is used to display output on the
screen?</label>

             <div class="radio">

                      <label><input type="radio"
name="print">printf()</label>
```

```
                      </div>

                      <div class="radio">

                      <label><input type="radio"
name="scanf">scanf()</label>

                      </div>

                      <div class="radio">

                      <label><input type="radio"
name="switch">switch</label>

                      </div>

      </div>



                <button type="submit" class="btn btn-primary
btn-success">Submit the test</button> <br> <br>

  </div>




    <div id="menu2" class="tab-pane fade">

      <h3>Find the job</h3> <hr>

      <br>


      <div class="container">


      <form>

                      <div class="input-group">

                         <input type="text" class="form-control"
id="myInput" placeholder="Look for a job">

                         <div class="input-group-btn">
```

```
                           <button class="btn btn-default"
type="submit">
                              <i class="glyphicon glyphicon-
search"></i>
                           </button>
                         </div>
                      </div>
                    </form> <br> <br>




      <div style="width: 600px;">
                         <!-- Left-aligned -->
                         <div class="media">
                           <div class="media-left">
                              <img src="images/img_avatar1.png"
class="media-object" style="width:60px">
                           </div>
                           <div class="media-body">
                              <h4 class="media-heading">Java
Developer</h4>
          <h5 class="media-heading"> Company Name : JK Software
Solution</h4>
                              <p>Job Location : Chennai</p>
          <p> Salary : 45000</p>
          <button type="submit" class="btn btn-primary btn-
success">Apply </button> <br> <hr>
                           </div>
```

```
</div>

                         </div>



           <br>

           <div style="width: 600px;">

                              <!-- Left-aligned -->

                              <div class="media">

                                <div class="media-left">

                                   <img src="images/img_avatar1.png" class="media-object" style="width:60px">

                                </div>

                                <div class="media-body">

                                   <h4 class="media-heading">.NET Developer</h4>

               <h5 class="media-heading"> Company Name : JK Software Solution</h4>

                                   <p>Job Location : Chennai</p>

               <p> Salary : 45000</p>

               <button type="submit" class="btn btn-primary btn-success">Apply </button> <br> <hr>

                                </div>

                              </div>

                         </div>



           <div style="width: 600px;">

                              <!-- Left-aligned -->

                              <div class="media">
                                <div class="media-left">
                                   <img src="images/img_avatar1.png"
  class="media-object" style="width:60px">
                                </div>
                                <div class="media-body">
                                   <h4 class="media-heading">PHP
  Developer</h4>
               <h5 class="media-heading"> Company Name : JK Software
  Solution</h4>
                                   <p>Job Location : Chennai</p>
               <p> Salary : 45000</p>
               <button type="submit" class="btn btn-primary btn-
  success">Apply </button> <br> <hr>
                                </div>
                              </div>
                         </div>
  <div style="width: 600px;">
                              <!-- Left-aligned -->
                              <div class="media">
                                <div class="media-left">
                                   <img src="images/img_avatar1.png"
  class="media-object" style="width:60px">
                                </div>
                                <div class="media-body">
                                   <h4 class="media-heading">UX /UI
  Designer</h4>
               <h5 class="media-heading"> Company Name : JK Software
  Solution</h4>
```

```
                              <p>Job Location : Chennai</p>
          <p> Salary : 45000</p>
          <button type="submit" class="btn btn-primary btn-success">Apply </button> <br> <hr>
                           </div>
                         </div>
                    </div>
</div>
</div>
</div>
<script>
  $(document).ready(function(){
    $("#myInput").on("keyup", function() {
    var value = $(this).val().toLowerCase();
    $("#myDIV *").filter(function() {
      $(this).toggle($(this).text().toLowerCase().indexOf(value) > -1)
    });
    });
  });
  </script>

</body>

</html>
```

BACK END :

```
from flask import *#importing flask (Install it using python -m
pip install flask)

import os

import ibm_db

import bcrypt

#import send_from_directory

app = Flask(__name__) #initialising flask

app.secret_key = ''

PEOPLE_FOLDER = os.path.join('static', 'people_photo')

import ibm_db

conn = ibm_db.connect("DATABASE=bludb;

HOSTNAME=54a2f15b-5c0f-46df-8954-
7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT
=32733;SECURITY=SSL;

SSLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID
=phb43134;PWD=mTTnIE45Raj9A0rr",'','')

@app.route("/") #defining the routes for the home() funtion
(Multiple routes can be used as seen here)

@app.route("/home")

def dashboard():

    return render_template("dashboard.html")

def createjobseeker_profile():

  if request.method == 'POST':

    name = request.form['name']

    email = request.form['email']

    mobileno=request.form['mobileno']
    nationality=request.form['nationality']

    qualification=request.form['qualification']

    passout=request.form['passout']

    experience=request.form['experience"];

    degree=request.form['degree'];

    per_sslc=request.form['per_sslc'];

    per_hsc=request.form['per_hsc'];

    per_degree=request.form['per_degree'];

    tum = request.files['resume']

tum.save(os.path.join(app.config['UPLOAD_DIR'],tum.filename))

    hash=bcrypt.hashpw(password.encode('utf-
8'),bcrypt.gensalt())

    query = "SELECT EMAIL FROM JOBSEEKER WHERE EMAIL=?"

    stmt = ibm_db.prepare(conn, query)

    ibm_db.bind_param(stmt,1,email)

    ibm_db.execute(stmt)

    isUser = ibm_db.fetch_assoc(stmt)

    if not isUser:

      insert_sql = "INSERT INTO JOBSEEKER (name, email,
mobileno, nationality,

 qualification, passout, experience,
degree,per_sslc,per_hsc,per_degree) 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, mobileno)
      ibm_db.bind_param(prep_stmt, 4, nationality)
      ibm_db.bind_param(prep_stmt, 5, qualification)
      ibm_db.bind_param(prep_stmt, 6, passout)
      ibm_db.bind_param(prep_stmt, 7, experience)
      ibm_db.bind_param(prep_stmt, 8, degree)
      ibm_db.bind_param(prep_stmt, 9, per_sslc)
      ibm_db.bind_param(prep_stmt, 10, per_hsc)
      ibm_db.bind_param(prep_stmt, 11, per_degree)
      ibm_db.execute(prep_stmt)
      return render_template('dashboard.html')
  return render_template('dashboard.html',name='Home')
@app.route('/logout')
def logout():
    session.pop('email',   None)
    return redirect(url_for('login'))
```

SPRINT:4

```
<!DOCTYPE html>

<html lang="en">

<head>

  <title>Bootstrap Example</title>

  <meta charset="utf-8">

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

  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>


  <style>

    .rus{


        border-radius: 0%

    };


    .black{

      color:black;

    }

</style>
```

```
</head>

<body>


<!
===========================================================-->

<nav class="navbar navbar-inverse rus">

    <div class="container-fluid">

      <div class="navbar-header">

        <a class="navbar-brand" href="#">Skill and Job
Recommender</a>

      </div>


      <ul class="nav navbar-nav navbar-right">

        <li><a href="#"><span class="glyphicon glyphicon-log-
in"></span> Log out</a></li>

      </ul>

    </div>

  </nav>

<!
===========================================================-->


<div class="container">

  <ul class="nav nav-tabs">

    <li class="active"><a data-toggle="tab" href="#home">Company
Profile</a></li>

    <li><a data-toggle="tab" href="#menu2">Find
candidate</a></li>
```

```
  </ul>


  <div class="tab-content">

    <div id="home" class="tab-pane fade in active">

      <h3>COMPANY PROFILE</h3>

      <hr>


      <form>

        <div class="alert alert-danger">

        <div class="form-group">

            <label for="name"><span clss="black">Company Name:
</span></label>

            <input type="text" class="form-control" id="name"
value="Mahadevan">

          </div>

          <div class="form-group">

            <label for="email"><span class="black">Company Email
Address </span></label>

            <input type="text" class="form-control" id="email"
value="Mahadevan@gmaiil.com">

          </div>


          <div class="form-group">

            <label for="email"><span class="black">Mobile No
</span></label>

            <input type="text" class="form-control" id="email"
value="9003824096">
```

```
          </div>


          <div class="form-group">

            <label for="quali"><span
class="black">Location</span></label>

            <input type="text" class="form-control" id="quali"
value="India">

          </div>




          <textarea class="form-control" id="aboutyou" rows="3"
placeholder="Write few words about company"></textarea> <br>
<br>


          <button type="submit" class="btn btn-primary btn-
success">Save changes</button> <br> <br>

          </div>

        </form>

    </div>




    <div id="menu2" class="tab-pane fade">
```

```
      <h3>Find the Condidate</h3> <hr>

      <br>


      <div class="container">

      <!.......................................................................................
   >

      <form>

                        <div class="input-group">

                           <input type="text" class="form-control"
   id="myInput" placeholder="Look for a candidate">

                           <div class="input-group-btn">

                             <button class="btn btn-default"
   type="submit">

                                <i class="glyphicon glyphicon-
   search"></i>

                             </button>

                           </div>

                        </div>

                      </form> <br> <br>


      <!..............................................>



      <div style="width: 600px;">

                           <!-- Left-aligned -->

                           <div class="media">

                             <div class="media-left">
```

```
                              <img src="images/img_avatar1.png"
class="media-object" style="width:60px">
                           </div>
                           <div class="media-body">
                              <h4 class="media-heading">Name :
Mahadevan</h4>
          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>
          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>
                           </div>
                         </div>
                    </div> <br>
<div style="width: 600px;">
                         <!-- Left-aligned -->
                         <div class="media">
                           <div class="media-left">
                              <img src="images/img_avatar1.png"
class="media-object" style="width:60px">
                           </div>
                           <div class="media-body">
                              <h4 class="media-heading">Name :
Manikandan</h4>
          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>
          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>
                           </div>
```

```
                              </div>

                         </div>  <br>


        <div style="width: 600px;">

                              <!-- Left-aligned -->

                              <div class="media">

                                <div class="media-left">

                                   <img src="images/img_avatar1.png"
class="media-object" style="width:60px">

                                </div>

                                <div class="media-body">

                                   <h4 class="media-heading">Name :
Kavi</h4>

          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>

          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>

                                </div>

                              </div>

                         </div>  <br>


        <div style="width: 600px;">

                              <!-- Left-aligned -->

                              <div class="media">

                                <div class="media-left">

                                   <img src="images/img_avatar1.png"
class="media-object" style="width:60px">
```

```
                              </div>

                              <div class="media-body">

                                <h4 class="media-heading">Name :
Ram</h4>

          <h5 class="media-heading"> Qualification : B.E
(Computer Science Engineering )</h4>

          <button type="submit" class="btn btn-primary btn-
success">Know More </button> <br> <hr>

                              </div>

                            </div>

                       </div>  <br>

 </div>

</div>

</div>

<script>

  $(document).ready(function(){

    $("#myInput").on("keyup", function() {

    var value = $(this).val().toLowerCase();

    $("#myDIV *").filter(function() {

$(this).toggle($(this).text().toLowerCase().indexOf(value) > -1)

    });

    });

  });

  </script>

</body>

</html>
```

GITHUB & PROJECT DEMO LINK:

All the tasks of developing the application were uploadedon the github.

The github has been uploaded below.

https://github.com/IBM-EPBL/IBM-Project-25197-1659954891