Final Report

Team Id : PNT2022TMID33615
Team Leader : Srikanth K
Team Members: Saran Kumar R
Siva Ganesh G
Thirughanamuthuselvan S

1 Introduction

1.1 Project Overview

This Project will work on website to give the service to the job seeker . This is implemented using the python flask , HTML , CSS and IBM Products like DB2 to store the users related information, Object Storage to store the big data like images , videos, Chat Bot to interact with the user in effective manner.

First, job offers are collected from job search websites then they are prepared to extract meaningful attributes such as job titles and technical skills. Job offers with common features are grouped into clusters. As job seeker like one job belonging to a cluster, he will probably find other jobs in that cluster that he will like as well. A list of top n recommendations is suggested after matching data from job clusters and job seeker behavior, which consists on user interactions such as applications, likes and rating.

1.2 Purpose

Now-a- days more number of students are lacking out of job due to high competition. We guide them through our website so that they can easily get the job for their specific set of skills and we improve their knowledge through learning platform , so that they can improve for various other job roles.

2 Literature Survey

2.1 Existing Problem

i) You’re pressed for time

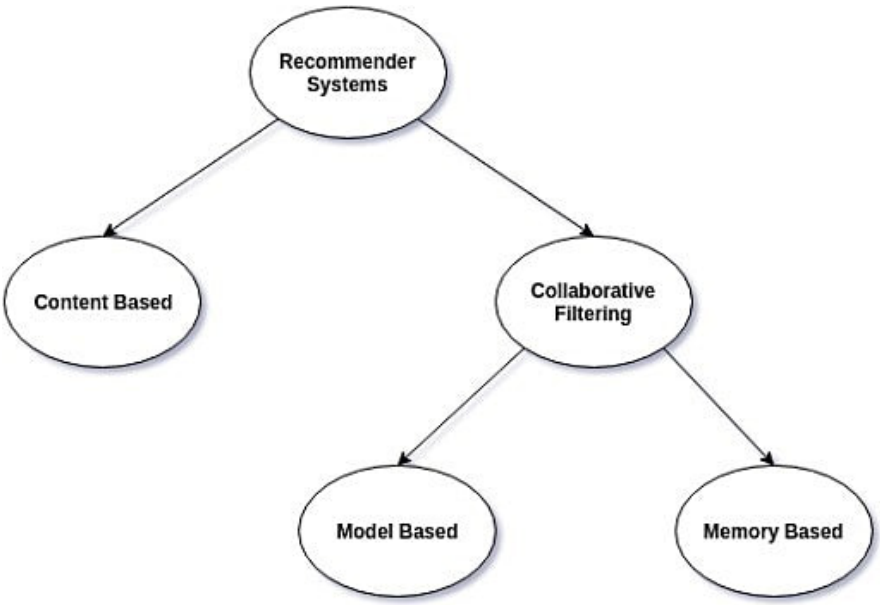
As we all deal with the ever-increasing complexity of modern life, there seems to be no end to the demands on our time – and looking for a job is just one more. So, how do you squeeze a job search into your already overcrowded schedule? How much time should you allot to this activity?

Well, there’s an old expression that says looking for a job is a full-time job. Now, this may be a bit of an exaggeration, but the fact of the matter is that the employment process can be extremely complicated and time-consuming.

ii) You lack a strong online presence

This issue seems to be more prevalent among certain groups. For example, individuals re-entering the job market after being out of work for some time for whatever reason (like raising

a child or taking care of a loved one), people who have held the same job or worked at the same company for a long time, and older individuals who may not be tech-savvy. Individuals in these groups, and others, have likely never developed a professional online profile.



References

Lee I (2007). An Architecture for a Next-GenerationHolistic E-Recruiting System. Commun.ACM 50(7):81-85.

2. Färber F, Weitzel T, Keim T (2003). An Automated Recommendation Approach to Selection in Personnel

Recruitment. In Proceedings of AMCIS. Tampa, FL, USA, AISeL.

3. Lang S, Laumer S, Maier C, Eckhardt A (2011). Drivers, Challenges and Consequences of E-Recruiting–A Literature Review. SIGMISCPR'11. San Antonio, Texas, USA, ACM pp. 26-35.

4. Brusilovsky P (2001). Adaptive hypermedia. User Model. User Adapt. Interact. 11(1- 2):87-110.

2.3 Problem Statement Definition

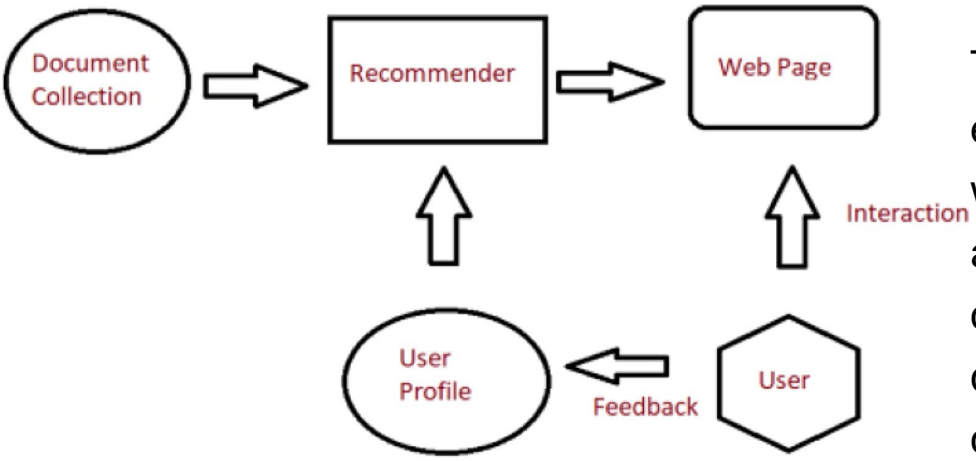


Fig: Recommender System

To develop an end-to-end web application capable of displaying the current job openings based on the user skill set. Users will interact with the chat-bot and Can get 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 web-page.

Recommend a job who are seeking for jobs through online platform.

How this is achieve ?

By the above figure , we collect the user releated information and company related information and provide a job for their skills.

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map

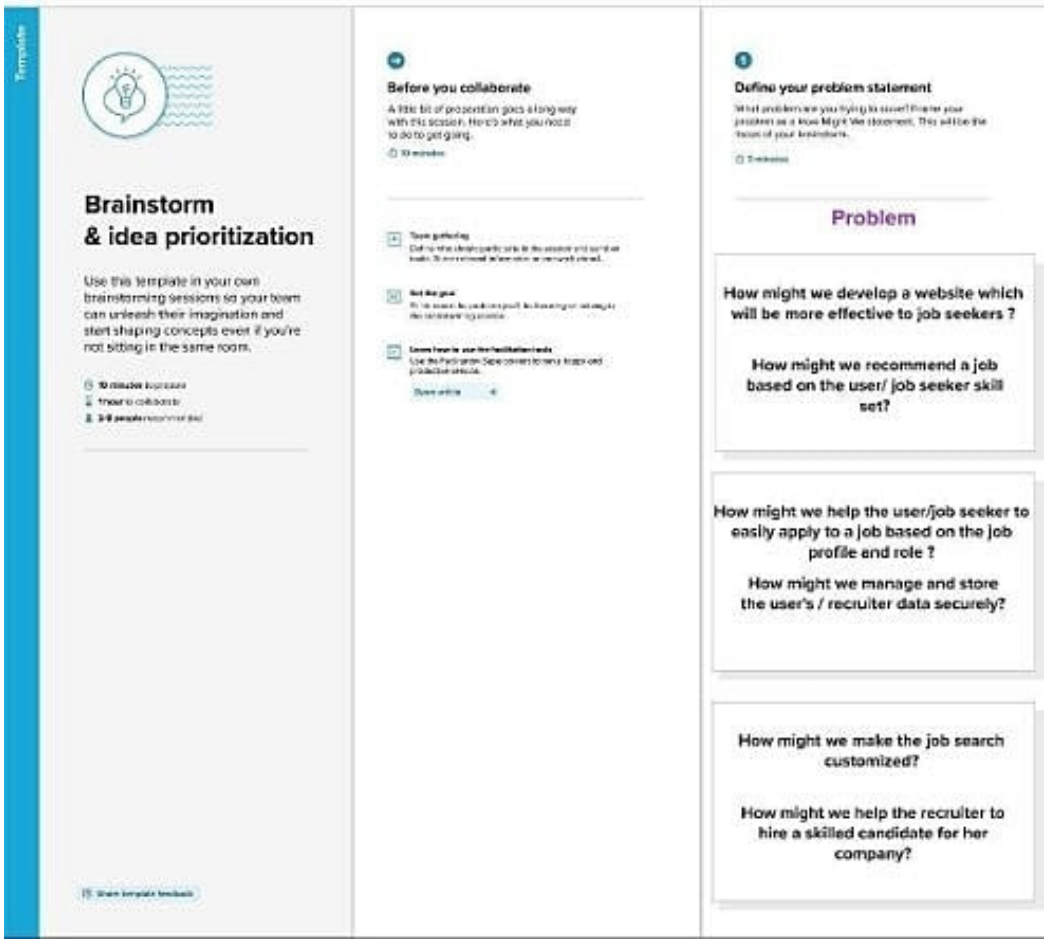
3.2 Ideation & Brainstromin

Brainstorming provides a free and open environment that encourages everyone within a teamtoparticipate in the creative thinking process that leads toproblem solving. Prioritizing volume over value, out-of- the-box ideas are welcome and built upon, and all participants areencouraged to collaborate, helping eachother develop a rich amount of creative solutions.

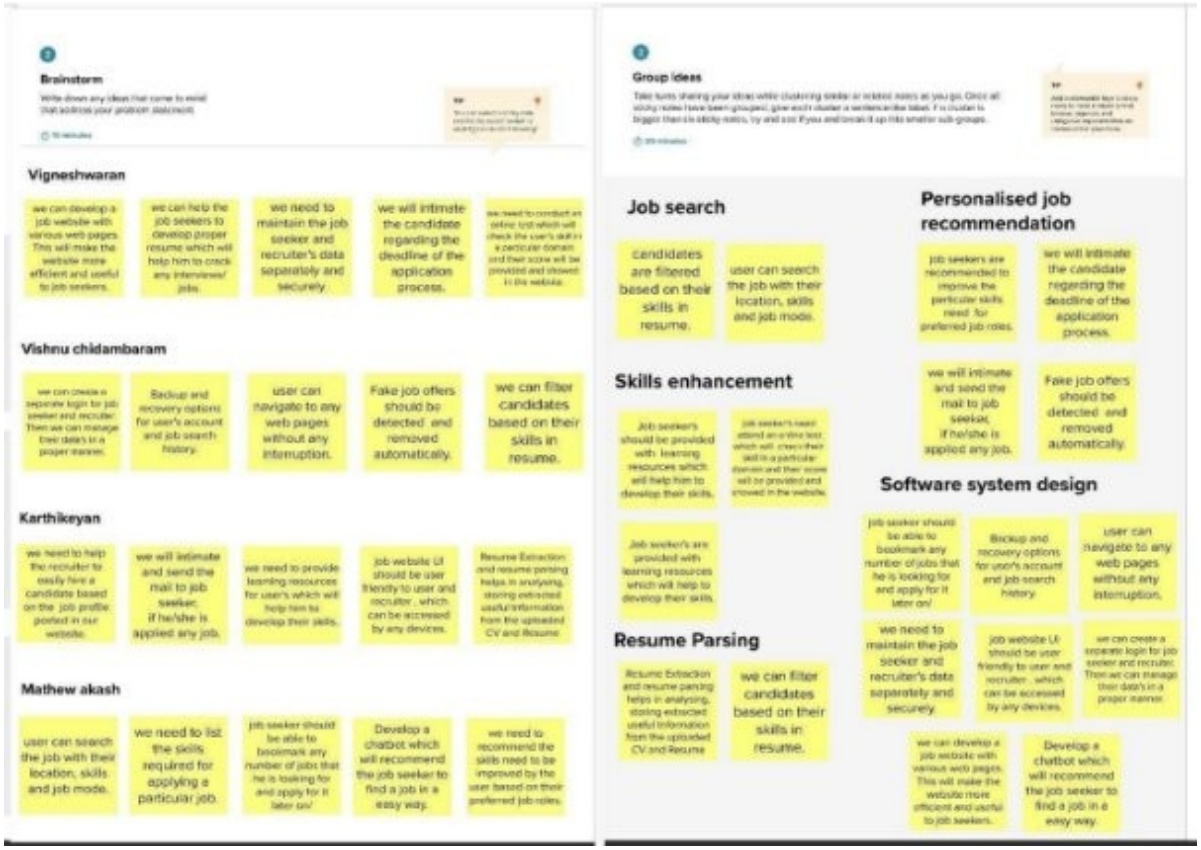
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference: https:/ www.mural.co/templates/empathy-map-canvas

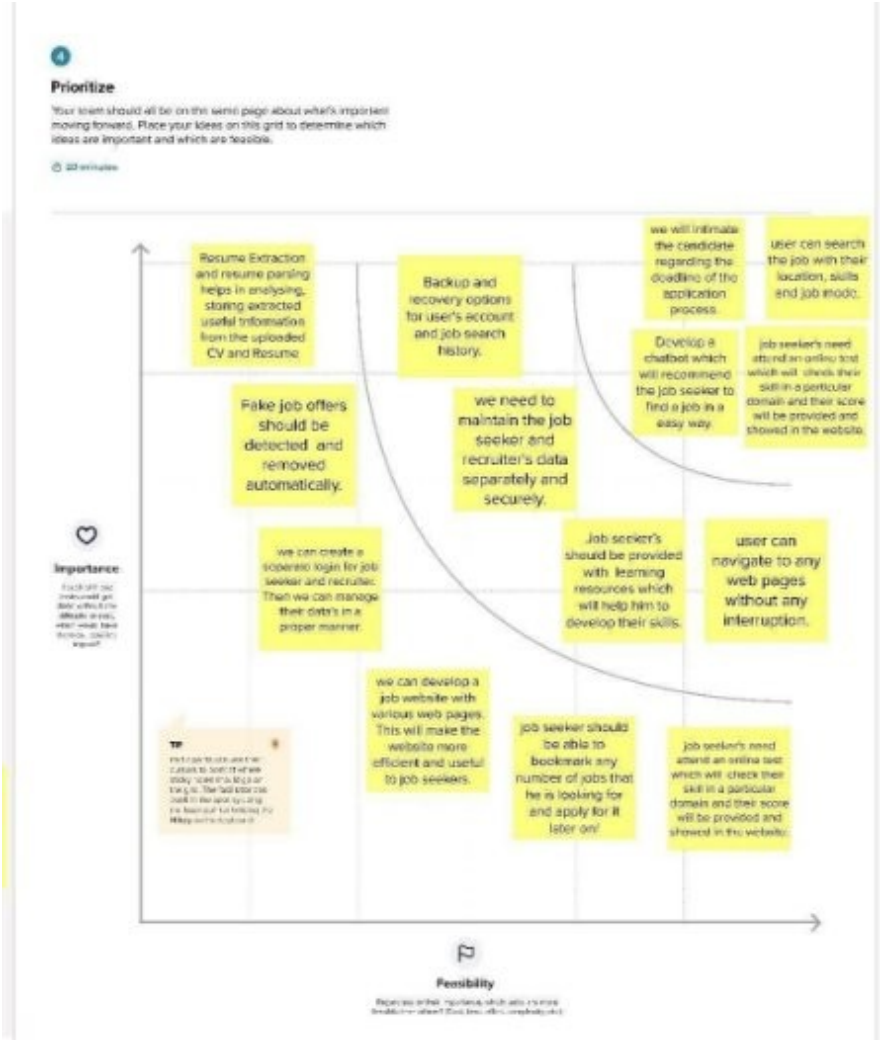
Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



3.3 Proposed solution

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	To develop an end-to-end web application capable of displaying the current job openings based on the user skill set. Users will interact with the chat-bot and can get recommendationsbased on their skills. We can use a job search APIto get the current job openings in the market which will fetchthe data directly from the web-page.
2.	Idea / Solution description	We have an idea to recommend many jobs frommany companies for which role you want and also it shows number of vacancies available in companies. Surely it is an application for all domain graduates.
3.	Novelty / Uniqueness	This Application uniquely identifies the user's skills recommend the job according to the user'sinterest.
4.	Social Impact / Customer Satisfaction	This Application will help user to gain more knowledge about the company, their job role,company vacancies and the company expectations.
5.	Business Model (Revenue Model)	Cost effective, User friendly.

6.	Scalability of the Solution	It is lifelong recommender app. Once the user has login to this application, he will be notifiedabout the job up to date.

3.4 Problem solution fit

1.Customer Segments: Graduates easily find job opportunities throughonline in everywhere.	**6.Customer constrains:** Constraint that customer will face about data security which they provided to the site.	**5.Available solutions:** Alert through notifications in which areathey are interested.

2.Jobs to be done : Notify them through message and emails.	**9.Problem route cause:** Finding job opportunities in newspaperand reference through friends is difficult task.	**7.Behavior:** Providing the information with real time.

3.Triggers: Make a refferal opttion so advertise andshow advertisement in websites. **4.Emotions:** Now days many graduates are in unemployment situation.By using this application they get a job means graduates feelhappy.	**10.Solution:** Skill and job application recommend many jobs from many companies for which role you want and also is shownumber of vacancies available in companies.	**8.Channels of behavior:** **8.18.1 online:** Using chat bot we can clarify the customer queries **8.28.2 offline:** Make an option of cart so that they

		can view about the job details even in offline.

4 Requirement Analysis

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.

4.1 Functional Requirements

Requirements analysis is a set of operations that helps define users' expectations of the application you are building or modifying. Software engineering professionals sometimes call it requirement engineering, requirements capturing or requirement gathering.

The process involves analyzing, documenting, validating and managing system or software requirements. Requirements analysis involves various tasks that help engineers understand stakeholder demands and explain them in simple and visual ways.It is essential to a software or system project's success.

For a project to be successful, its requirements must be:

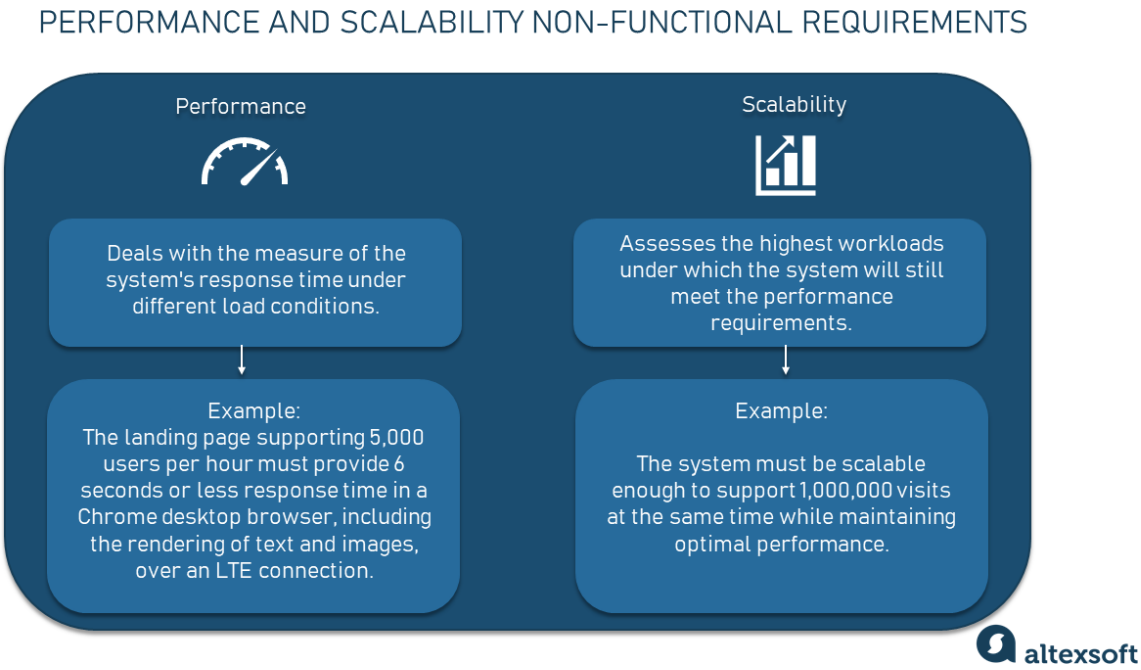
> Testable --> Actionable

> Documented --> Measurable

> Tracable

Requirements analysis involves various stakeholders, such asproject sponsors, throughout the project as well as end users whose inputs are most important. The best results typically occur when all parties work together to develop a high-quality requirements document.

4.2 Non - Functional Requirements



Nonfunctional Requirements (NFRs) define system attributes such as **security, reliability, performance, maintainability, scalability, and usability**. They serve as constraints or restrictions on the design of the system across thedifferent backlogs.

Difference Between Functional and non- functional Requirements

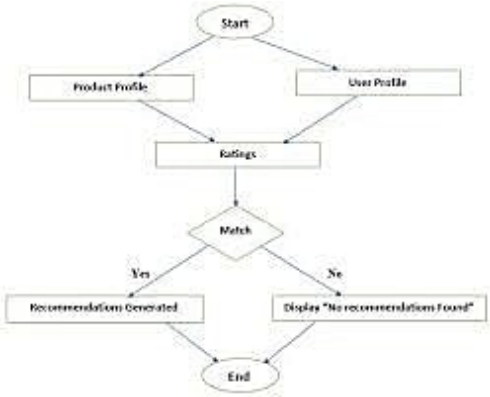
FUNCTIONAL vs NONFUNCTIONAL REQUIREMENTS

	Functional requirements	Nonfunctional requirements
Objective	Describe what the product does	Describe how the product works
End result	Define product features	Define product properties
Focus	Focus on user requirements	Focus on user expectations
Documentation	Captured in use case	Captured as a quality attribute
Essentiality	They are mandatory	They are not mandatory, but desirable
Origin type	Usually defined by user	Usually defined by developers or other tech experts
Testing	Component, API, UI testing, etc. Tested before nonfunctional testing	Performance, usability, security testing, etc. Tested after functional testing
Types	External interface, authentication, authorization levels, business rules, etc.	Usability, reliability, scalability, performance, etc.

5 Project Design

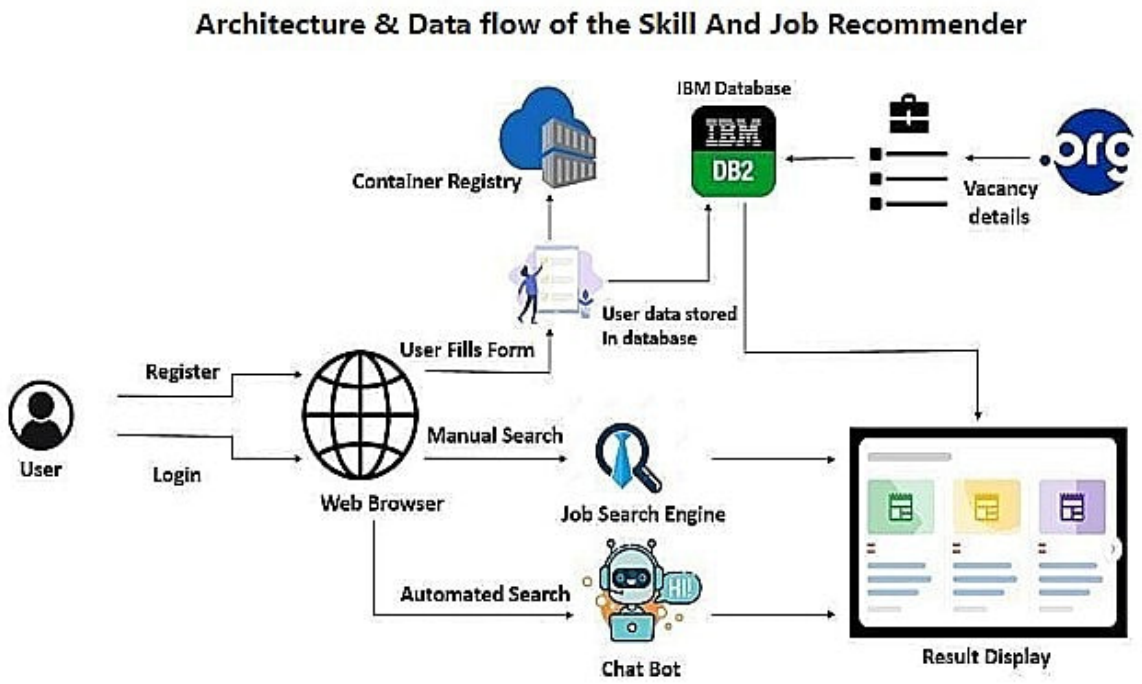
5.1 Data flow Diagrams

The below diagram shows the data flow of our application



Solution Architecture:

Solution Architecture Diagram:



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

- Final the best tech solutions to solve existing business problems
- Describe the structure , characteristics,behaviour and other aspects of the software to project stakeholders.
- Define features , developement process and solution requirements.
- Provide specifications according to which the solution is defined , managed and delivered.

5.3 Users Stories

Our End User:

Our end users are the persons who are seeking for jobs for their attained skills.

Their needs:

They need a faster job recommending system to choose their jobs and they need some educational content support.

What we delivery to them(Benefit):

We provide all their aspects through our application so that they can fulfill their needs and they attend th company process easily and we grantee.

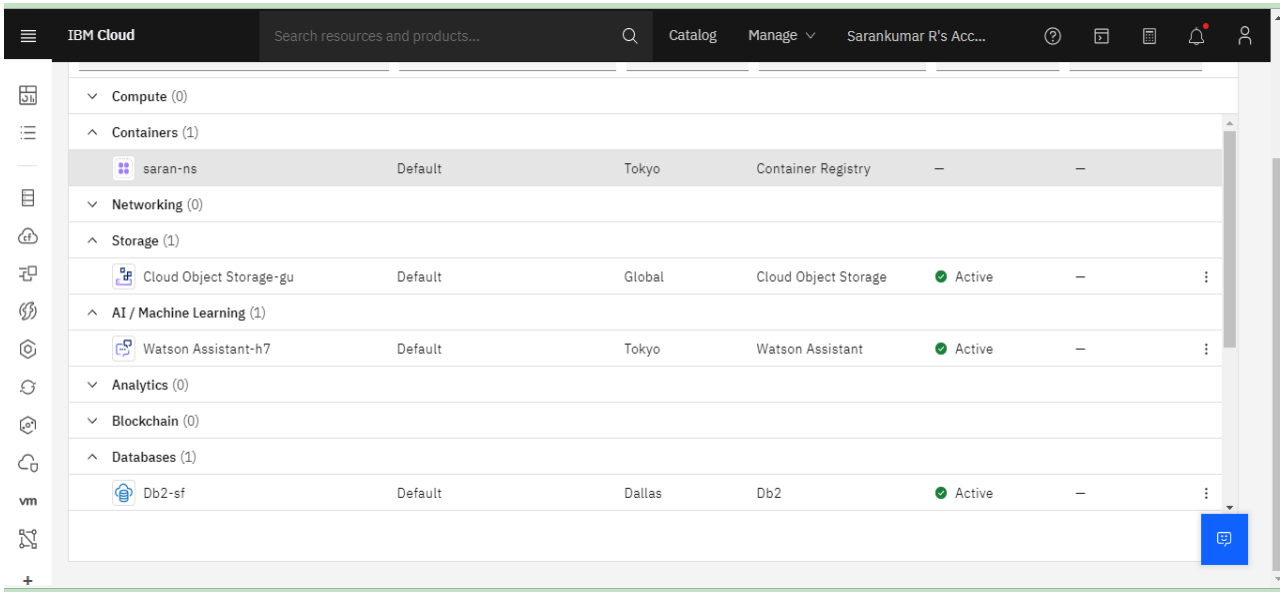
6 Project Planning & Scheduling

Sprint 1

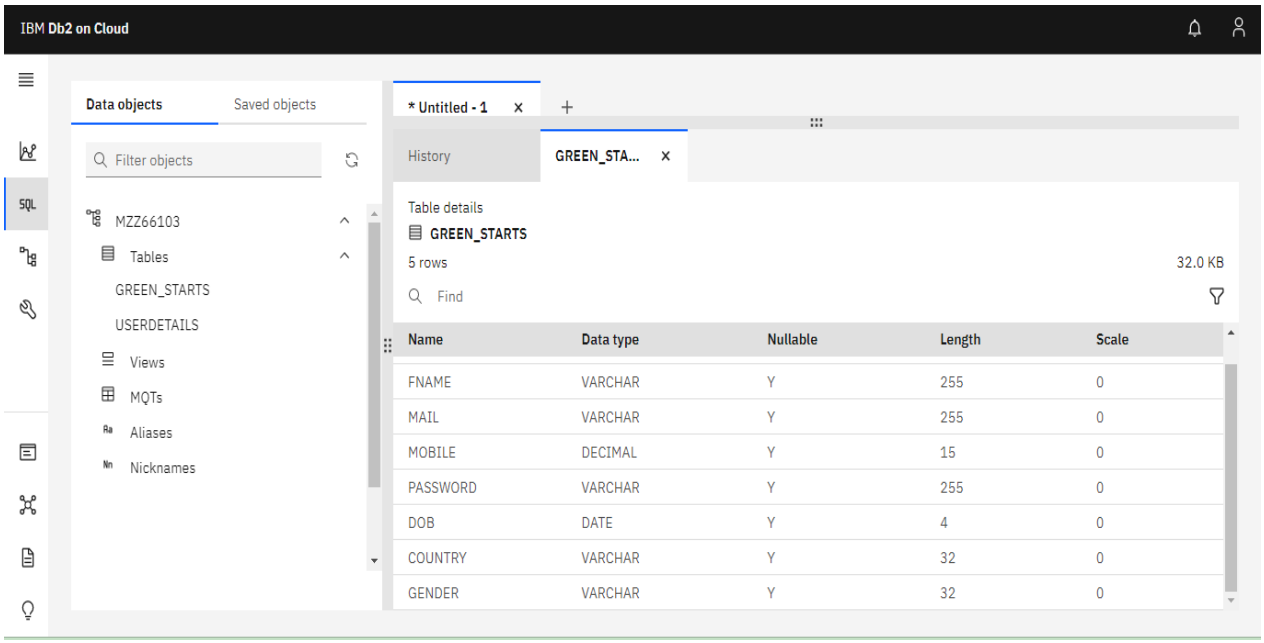
REQUIRED SERVICES:

1. *IBM DB2*
2. *IBM OBJECT STORAGE*
3. *IBM WATSON ASSISTANT*
4. *IBM CONTAINERS*
5. *SENDGRID INTEGARATION (TWILIO)*

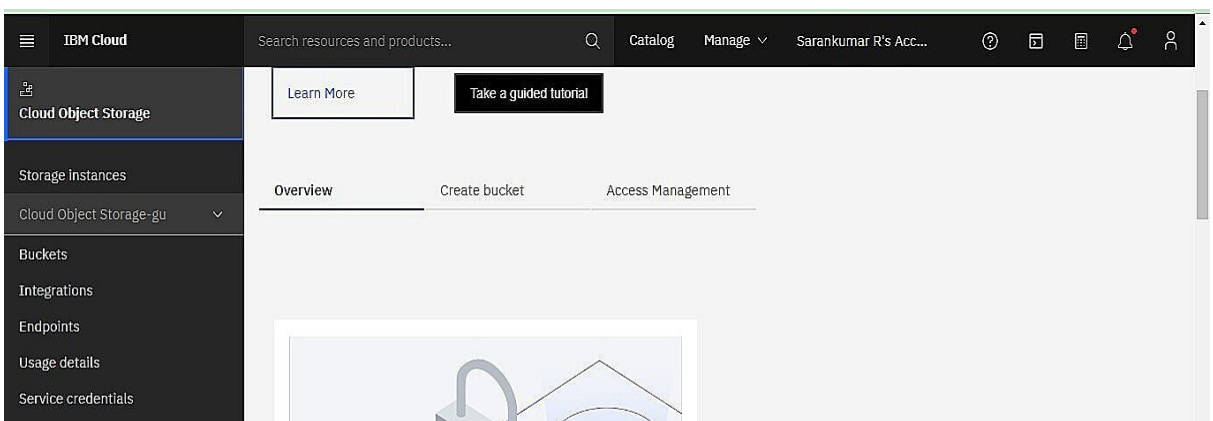
CREATING CLOUD ACCOUNT:



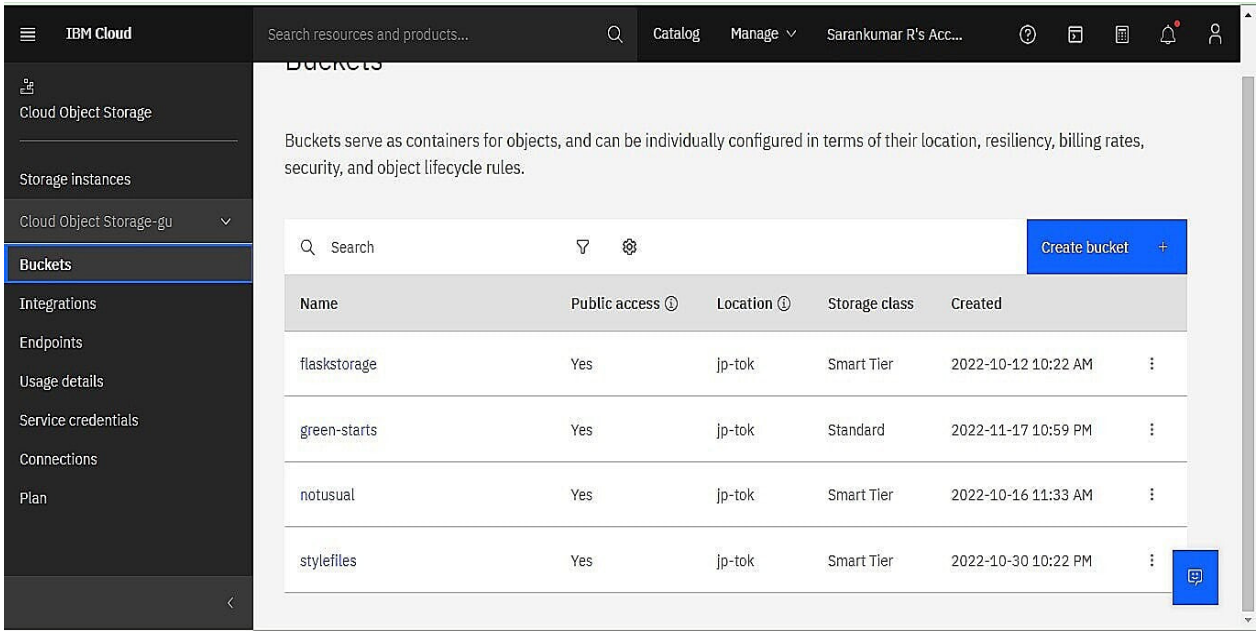
CREATE DATABASE AND TABLES IN IBM DB2:



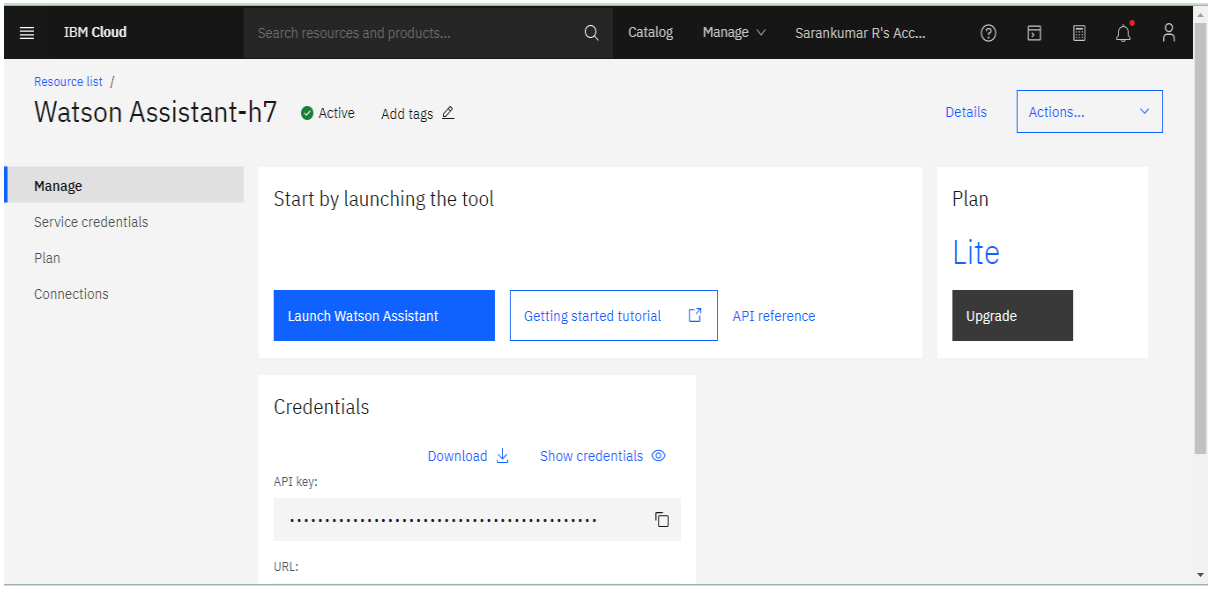
CREATION OF OBJECT STORAGE IN IBMCLOUD:



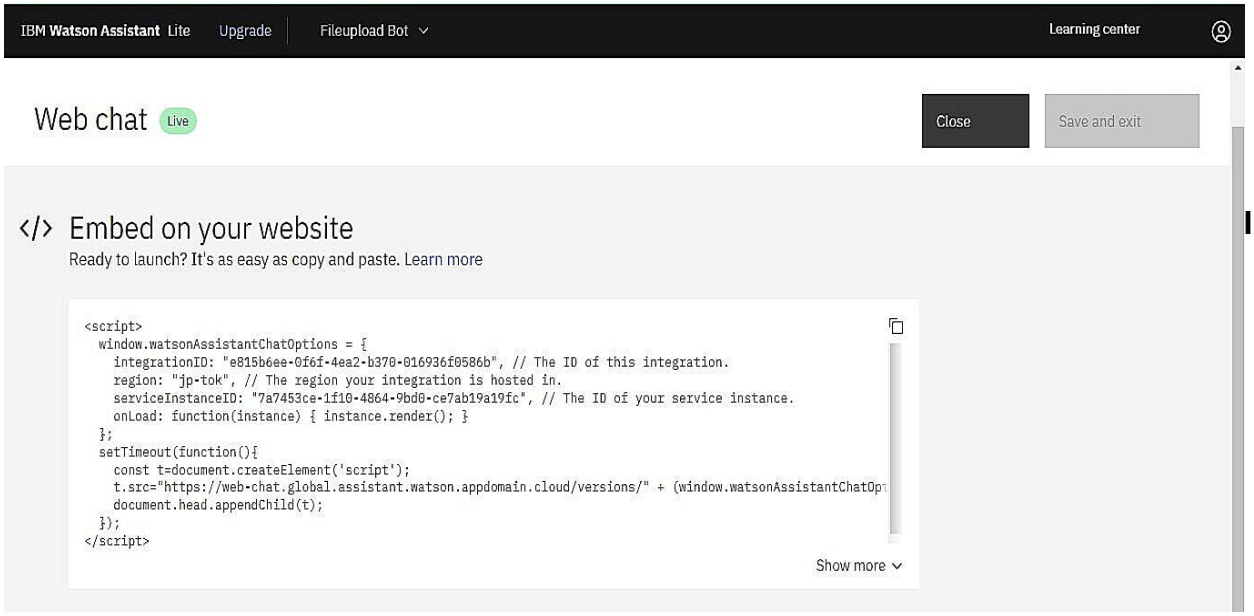
CLOUD OBJECT STORAGE BUCKETS:



CREATING WATSON ASSISTANT:

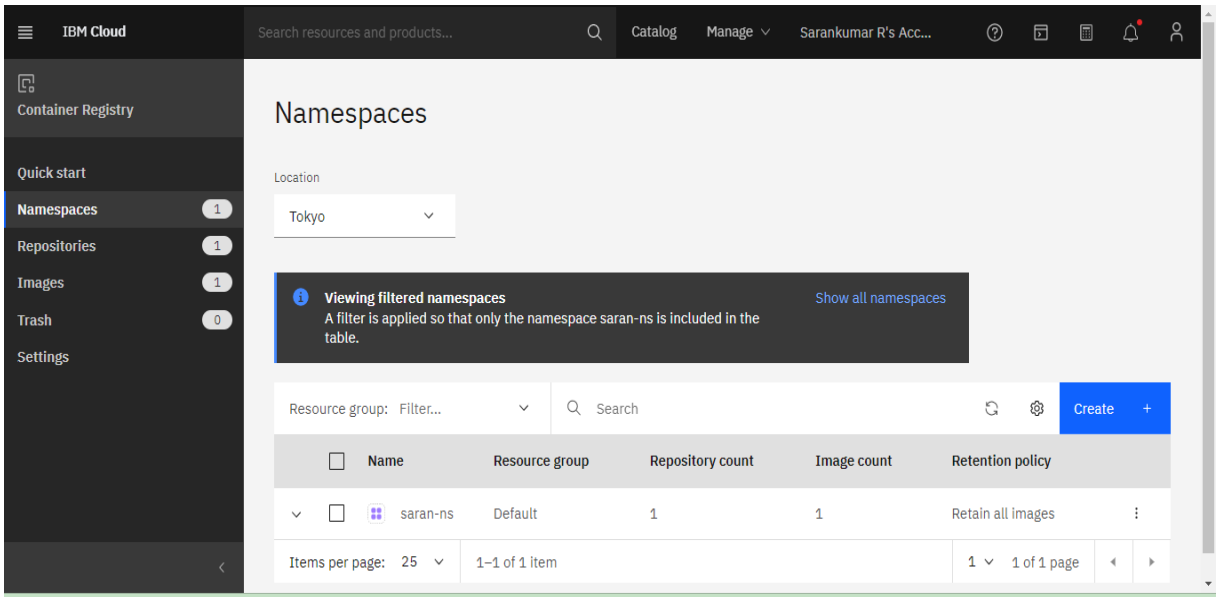


EMBEDD WATSON ASSISTANT ON TO THE WEBSITE:

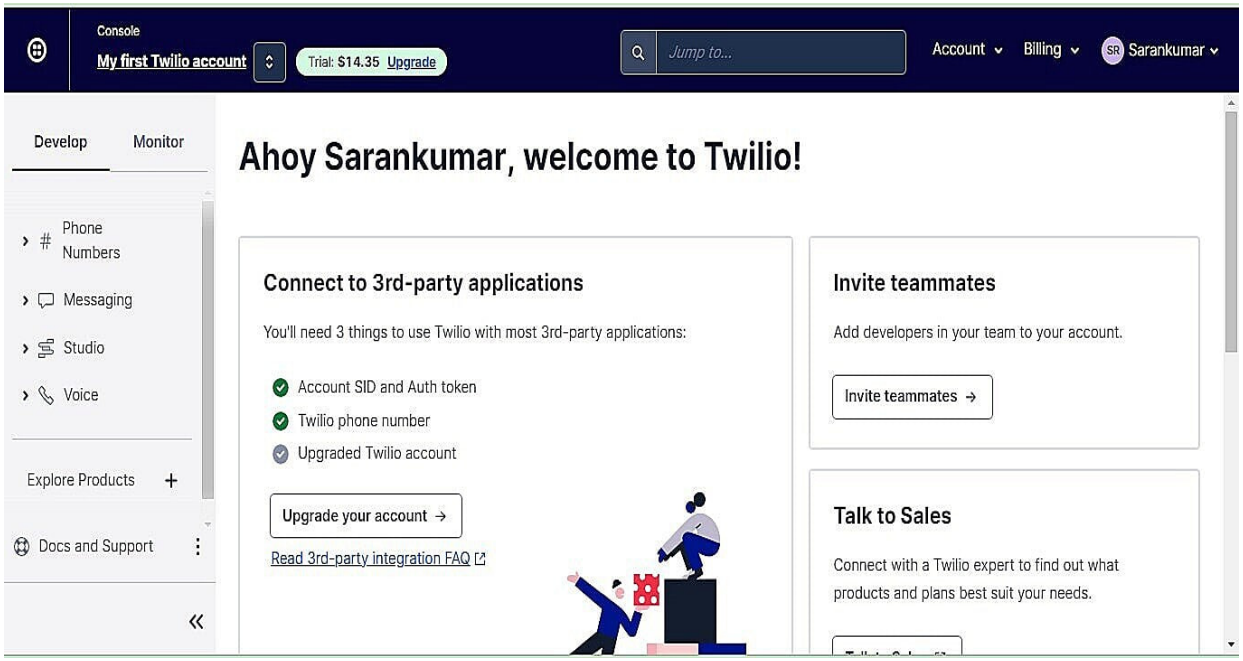


IBM

CONTAINER REGISTRY:

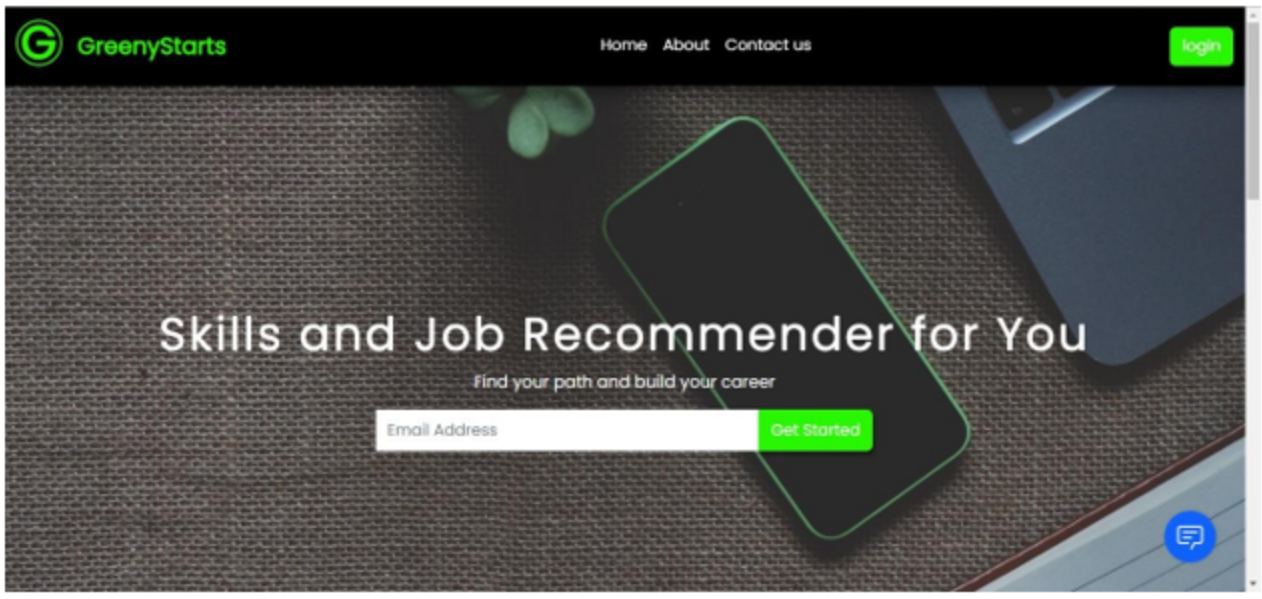


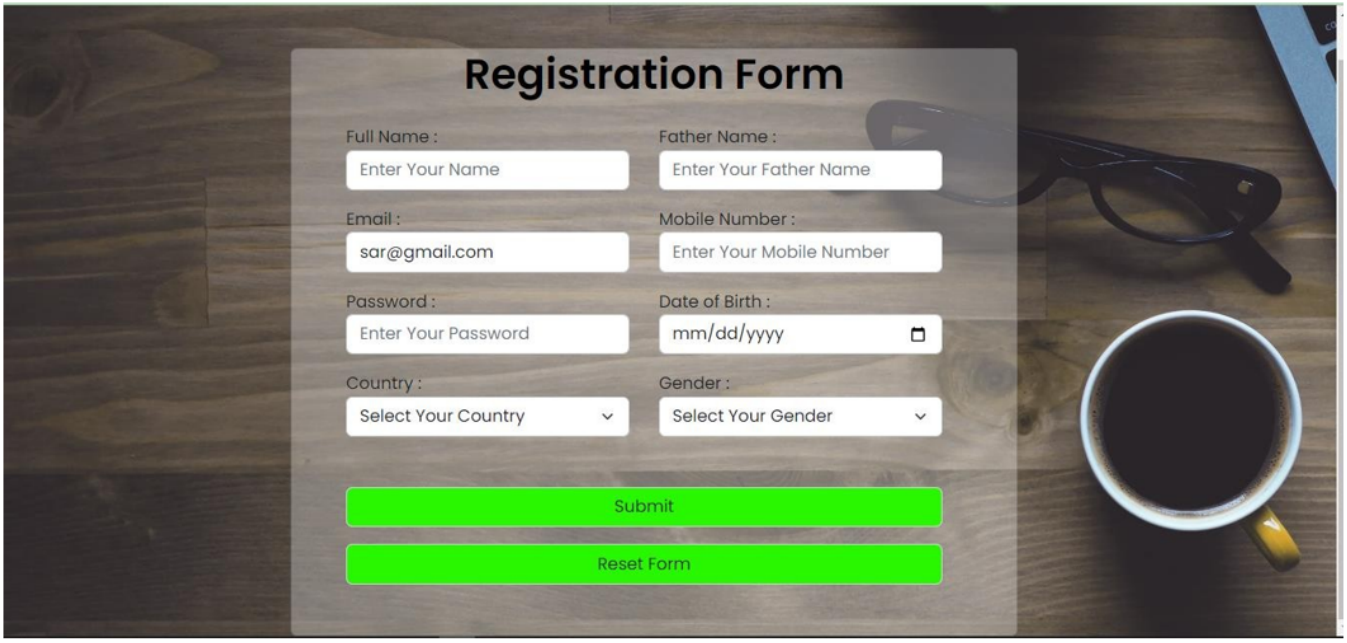
SENDGRID INTEGRATION:



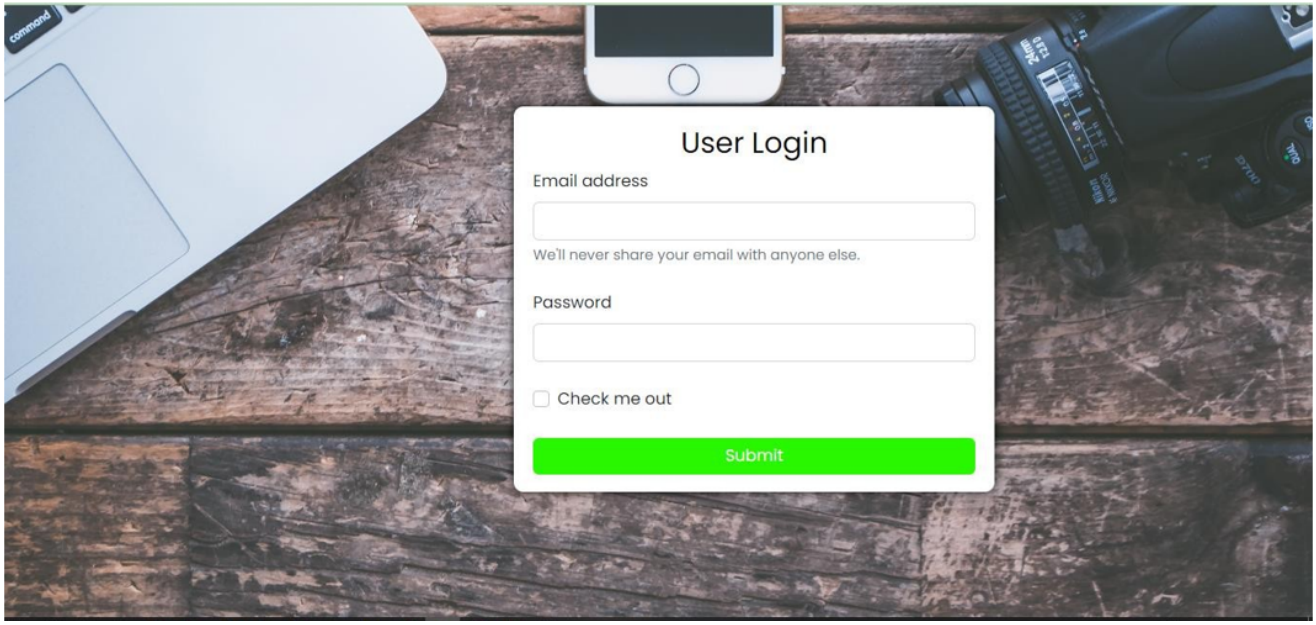
Application interface

HOME PAGE:




Registration Form
Full Name :
Enter Your Name
Father Name :
Enter Your Father Name
Email :
sar@gmail.com
Mobile Number :
Enter Your Mobile Number
Password :
Enter Your Password
Date of Birth :
mm/dd/yyyy
Country :
Select Your Country
Gender :
Select Your Gender
Submit
Reset Form

LOGIN PAGE:


User Login
Email address
We'll never share your email with anyone else.
Password
Check me out
Submit

7. coding

CREATING PYTHON FLASK AND INTEGRATING ALL THE SERVICES SPECIFIED IN SPRINT-1:

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

ibm_db

import ibm_boto3                                                                    ")
from ibm_botocore.client import Config, ClientError

COS_ENDPOINT="https://s3.jp-tok.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID="8ir_OK_kdjY8SXvQXnvcYD34CrCrX7TIELEbWpzBQCFH"
COS_INSTANCE_CRN="crn:v1:bluemix:public:cloud-object-
storage:global:a/cac77355b8e64285ab5824053ea1e90d:66b5a2a3-b654-458e-b4af-
b3c551fc5b0f::"


cos = ibm_boto3.resource("s3", ibm_api_key_id=COS_API_KEY_ID,
      ibm_service_instance_id=COS_INSTANCE_CRN,
      config=Config(signature_version="oauth"),
      endpoint_url=COS_ENDPOINT
)


app=Flask(__name__)

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

                                                                                    ")
@app.route("/login")def
login():
      return render_template("login.html")
```

```
      mail=request.form['email'] mobile=request.form['mbno']
      password=request.form['pass'] dob=request.form['dob']
      country=request.form['country']
      gender=request.form['gender']

      sql="INSERT INTO green_starts values(?,?,?,?,?,?,?,?)"
      stmt=ibm_db.prepare(conn,sql) ibm_db.bind_param(stmt,1,name)
      ibm_db.bind_param(stmt,2,fname) ibm_db.bind_param(stmt,3,mail)
      ibm_db.bind_param(stmt,4,mobile) ibm_db.bind_param(stmt,5,password)
      ibm_db.bind_param(stmt,6,dob) ibm_db.bind_param(stmt,7,country)
      ibm_db.bind_param(stmt,8,gender)
```

```
        ibm_db.execute(stmt)

        print("Account is Created !!")
        print(name+" "+fname+" "+mail+" "+mobile+" "+password+" "+dob+" "+country+""+gender)

        image="/static/images/tick1.gif"msg1="Congratulation!"
        msg2="Your Account has been Successfully Created!!"
        return render_template("notification.html",image=image,msg1=msg1,msg2=msg2)

@app.route("/checkuser",methods=['post'])def
checkuser():
        conn = ibm_db.connect("DATABASE=bludb ; HOSTNAME=6667d8e9-9d4d-4ccb-ba32-
21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=S
SL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mzz66103;PWD=EljmVVQ9CytfCyNc;", '', '')
        print("Cloud Connected Successfully!!")

        mail=request.form['email'] password=request.form['pass']

        if mail=="Admin@gmail.com" and password=="Admin":
                print("Admin Login")
                return render_template("admin.html")

        print(mail+" "+password)

        sql = "select *from green_starts WHERE mail=?"stmt =
        ibm_db.prepare(conn, sql) ibm_db.bind_param(stmt, 1,mail )
        ibm_db.execute(stmt)
        lis = ibm_db.fetch_assoc(stmt)

        image="/static/images/image2.gif"msg1="Ooops!"
        msg2="This E-Mail Doesn't Exist!!"

        if lis:
                sql="SELECT password from green_starts WHERE mail=?"
                stmt=ibm_db.prepare(conn,sql) ibm_db.bind_param(stmt,1,mail)
                ibm_db.execute(stmt)

                lis=ibm_db.fetch_assoc(stmt)


                e=lis.get('PASSWORD')


                if e==password:
                        print("Logged in successfully!!")
                        return   redirect(url_for('.user1',mail=mail))

                else:
```

```
                print("****Error****")return
render_template("notification.html",image=image,msg1=msg1,msg2="You Have EnteredWrong
Password!!")

    else:
        return
render_template("notification.html",image=image,msg1=msg1,msg2=msg2)



#                                ***********        Object Storage              ***********

def get_bucket_contents(bucket_name):
    print("Retrieving bucket contents from: {0}".format(bucket_name))try:
        files = cos.Bucket(bucket_name).objects.all()files_names = []
        for file in files: files_names.append(file.key)
            print("Item: {0} ({1} bytes).".format(file.key, file.size))return files_names
    except ClientError as be:
        print("CLIENT ERROR: {0}\n".format(be))

    except Exception as e:
        print("Unable to retrieve bucket contents: {0}".format(e))

@app.route('/user1',methods=['post','get'])def user1():
    mail=request.args['mail']

    conn = ibm_db.connect("DATABASE=bludb ; HOSTNAME=6667d8e9-9d4d-4ccb-ba32-
21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=S
SL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mzz66103;PWD=EIjmVVQ9CytfCyNc;", '', '')
    print("Cloud Connected Successfully!!")

    sql="SELECT * from green_starts WHERE mail=?"
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,mail) ibm_db.execute(stmt)

    lis=ibm_db.fetch_assoc(stmt)

    a=lis.get('UNAME') b=lis.get('FNAME')
    c=lis.get('MAIL') d=lis.get('MOBILE')
    e=lis.get('PASSWORD')f=lis.get('DOB')
    g=lis.get('COUNTRY') h=lis.get('GENDER')

    files = get_bucket_contents('green-starts') return
    render_template('user1.html', files =
files,mail=mail,username=a,fname=b,usermail=c,mobile=d,country=g,gender=h)
```

```
if_name_=="_main__":
    app.run(host="0.0.0.0",port="8020",debug=True)
```

8 .Testing

8.1 Test cases

It is shown in Project video how the test cases work

9. Results

It is shown in Project video how the test cases work

10.Advantages and disadvantages

Advantages

1.Used at anywhere any time.

2.And it is user friendly by sending notifications to them to manage their time.

3.Able to crack interview and able to attend multiple companies.

4. Able to learn multiple course in which they interset.

Disadvantages

1. Can be used only in online

Programming Languages used

1 HTML

Hypertext Markup Language, or HTML, is a programming language used to describe the structure of information on a webpage. Together, HTML, CSS, and JavaScript make up the essential building blocks of websites worldwide, with CSS controlling a page's appearance and JavaScript programming its functionality.

HTML TAGS

<!DOCTYPE>

- All HTML documents must start with a <!DOCTYPE> declaration.
- The declaration is not an HTML tag. It is an "information" to the browser about what document type to expect.
- In HTML 5, the declaration is simple:
- <!DOCTYPE html>

<a> Tag

The <a> tag defines a hyperlink, which is used to link from one page to another.

- The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

<body> Tag

- The <body> tag defines the document's body.
- The <body> element contains all the contents of an HTML document, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

 Tag

- The
 tag inserts a single line break.
- The
 tag is useful for writing addresses or poems.
- The
 tag is an empty tag which means that it has no end tag.

<button> Tag

- The <button> tag defines a clickable button.
  - Inside a <button> element you can put text (and tags like <i>, <b>, ,
, <img>, etc.). That is not possible with a button created with the <input> element!

Cascading Style Sheets

CSS stands for Cascading Style Sheets

CSS describes how HTML elements are to be displayed on screen, paper, or in other media

CSS saves a lot of work. It can control the layout of multiple web pages all at once

External stylesheets are stored in CSS files

Cascading Style Sheet(CSS) is used to set the style in web pages that contain HTML elements. It sets the background color, font-size, font-family, color, … etc property of elements on a web page.
There are three types of CSS which are given below:

Inline CSS
Internal or Embedded CSS
External CSS

Inline CSS:

Inline CSS contains the CSS property in the body section attached with element is known as inline CSS.
This kind of style is specified within an HTML tag using the style attribute.

```
<p style = "color:#009900; font-size:50px;
    font-style:italic; text-align:center;">
  GeeksForGeeks
</p>
```

Internal or Embedded CSS:

This can be used when a single HTML document must be styled uniquely.
The CSS rule set should be within the HTML file in the head section
i.e the CSS is embedded within the HTML file.

```
<div class ="geeks">
        A computer science portal for geeks
 </div>
```

External CSS:

External CSS contains separate CSS file which contains only style property

with the help of tag attributes
(For example class, id, heading, … etc). CSS property written in a separate file with .css extension and should be linked to the HTML document using ***link*** tag.
This means that for each element, style can be set only once and that will be applied across web pages.
Example: The file given below contains CSS property.
This file save with .css extension.

```
body {
   background-color:powderblue;
}
```

Properties of CSS:

Inline CSS has the highest priority, then comes Internal/Embedded followed by External CSS which has the least priority.

Multiple style sheets can be defined on one page.

If for an HTML tag, styles are defined in multiple style sheets then the below order will be followed.
As Inline has the highest priority, any styles that are defined in the internal and external style sheets are overridden by Inline styles.

Internal or Embedded stands second in the priority list and overrides the styles in the external style sheet.

External style sheets have the least priority. If there are no styles defined either in inline or internal style sheet then external style sheet rules are applied for the HTML tags.

JavaScript

JavaScript is the world's most popular programming language.

JavaScript is the programming language of the Web.

JavaScript is easy to learn.

JavaScript Statements

JavaScript statements are composed of:

Values, Operators, Expressions, Keywords, and Comments.

This statement tells the browser to write "Hello Dolly." inside an HTML element with id="demo":

```
document.getElementById("demo").innerHTML = "Hello Dolly.";
```

JavaScript Values

The JavaScript syntax defines two types of values:

- Fixed values
- Variable values

Fixed values are called **Literals**.

Variable values are called **Variables**.

Variables are containers for storing data (storing data values).

In this example, x, y, and z, are variables, declared with the var keyword:

```
var x = 5;
var y = 6;
var z = x + y;
```

LET Keyword

The let keyword was introduced in ES6 (2015).

Variables defined with let cannot be Redeclared.

Variables defined with let must be Declared before use.

Variables defined with let have Block Scope.

Operators

There are different types of JavaScript operators:

- Arithmetic Operators
- Assignment Operators
- Comparison Operators
- Logical Operators
- Conditional Operators
- Type Operators

JavaScript Strings

JavaScript strings are for storing and manipulating text.

```
let text = "John Doe";
```

String Length

To find the length of a string, use the built-in `length` property:String Methods

String length
String slice()
String substring()
String substr()
String replace()
String replaceAll()
String toUpperCase()
String trim()
String trimStart()
String trimEnd()
String padStart()
String padEnd()
String charAt()
String charCodeAt()
String split()

11. Conclusion

Thus by using this application users can able to find their jobs in online at anywhere anyplace . Through notifications users can gain the updates from the company interview process which date the selection process is start when to reach the location , place where to attend the test. By the learning platform they can able to enhance their already existing knowledge and improve their knowledge in new fields.

So these can be helpful in managing their time and works . It also helps to manage their family and it provide time to spend their time with their family instead searching jobs at many locations.

By using this the users get high benefit and easily find their jobs for their known skills.

12. Future Scope

This application can be used by many people in future so they this will get daily basis for job seekers to find jobs.

And by designing the application through high level technologies can able to give better customer support.

By adding Machine learning algorithms and artificial intelligence , we can able to predict the industry market at which time the demand get increased so that we can recommend more number of updates in a particular time so that the users can able to prepare for the interview process earlier.

Git Up code link

https://github.com/IBM-EPBL/IBM-Project-33254-1660217126.git