PROJECT REPO RT

TITLE: Skill/Job Recommender Application

TEAM LEADER : VENGATESHWARAN.H

TEAM MEMBERS:

SARAN.M

SATHISHKUMAR.R

SASIKUMAR.S

TEAM ID: PNT2022TMID33634

TABLE OF CONTENTS	PAGE NO
1.INTRODUCTION	
1.1 PROJECT OVERVIEW 1.2 PURPOSE	1-3

1.INTRODUCTION

1.1 PROJECT OVERVIEW

Job recommendation is an important task for the modern recruitment industry. An excellent job recommender system not only enables to recommend a higher paying job which is maximally aligned with the skill-set of the current job, but also suggests to acquire few additional skills which are required to assume the new position.

In this work, we created three types of information networks from the historical job data: (i) job transition network, (ii) job-skill network, and (iii) skill co-occurrence network. We provide a representation learning model which can utilize the information from all three networks to jointly learn the representation of the jobs and skills in the shared k-dimensional latent space.

In our experiments, we show that by jointly learning the representation for the jobs and skills, our model provides better recommendation for both jobs and skills. Additionally, we also show some case studies which validate our claim.

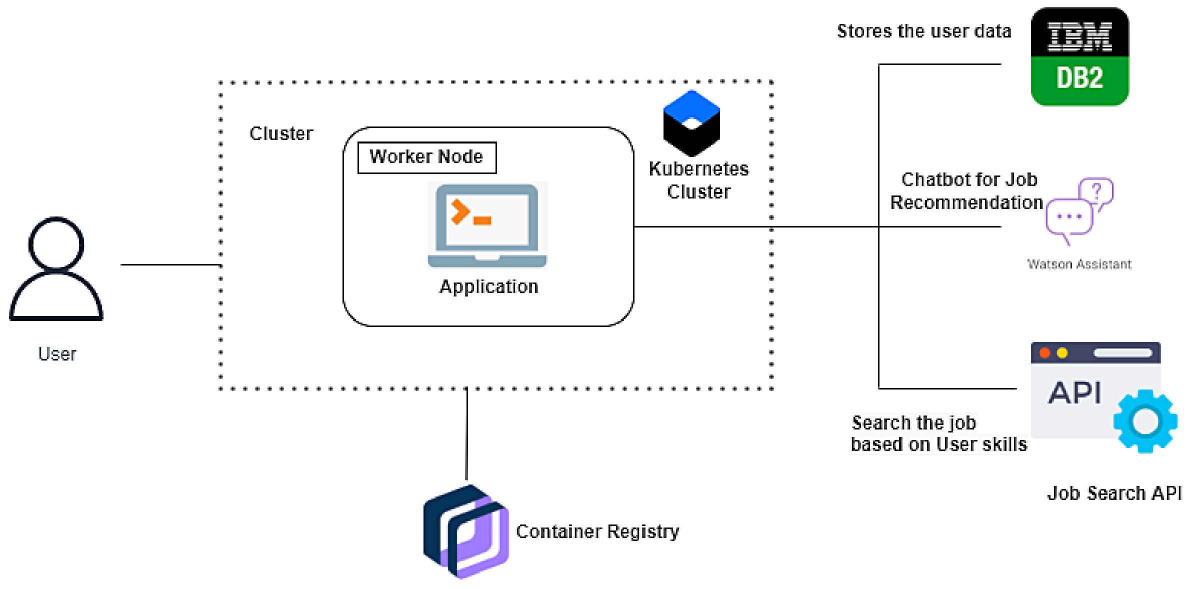
To develop an end-to-end web application capable of displaying the current job openings based on the user skillset.

The user and their information are stored in the Database. An alert is sent when there is an opening based on the user skillset.

Users will interact with the chatbot and can get the recommendations based on their skills. We can use a job search API to get the current job openings in the market which will fetch the data directly from the webpage

1.2 PURPOSE

An excellent job recommender system not only enables to recommend a higher paying job which is maximally aligned with the skill-set of the current job, but also suggests to acquire few additional skills which are required to assume the new position. candidates who are seeking for the job.Then that filtered job is recommended for that candidates based on their skillset.



2.IDEATION PHASE

2.1 LITERATURE SURVEY

ABSTRACT

This paper presents a job recommender system to match resumes to job descriptions (JD),both of which are non-standard and unstructured/semi-structured in form. First, the paper proposes a combination of natural language processing (NLP) techniques for the task of skill extraction. The performance of the combinedtechniques on an industrial scale dataset yieldeda precision and recall of 0.78 and 0.88 respectiv ely. The paper then introduces the concept of extracting implicitskill s – the skills which are not explicitly mentioned in a JD but may be implicit in the context of geography, industry or role. To mineand infer implicit skills for a JD, we find the other JDs similar to this JD. This similarity match is done in the semanticspace. A Doc2Vec model is trained on 1.1 Million JDs covering several domains crawled from the web, and all the JDs are projected onto this semantic space. The skills absent in the JD but present in similar JDs are obtained, and the obtained skills are weighted using several techniques to obtain the set of final implicit skills.Finally, severalsimilarity measures are explored to match the skills extractedfrom a candidate’s resume to explicit and implicit skills of JDs. Empirical results for matching resumes and JDs demonstrate that the proposed approach gives a mean reciprocal rank of 0.88, an improvement of 29.4% when compared to the perform ance of a baseline methodthat uses only explicit skills.

INTRODUCTION

Formal job search and application typically involves matching one’s profile or curriculum vitae (CV) with the available job descriptions (JD), a nd then applying for those job opportunities whose JDs are the closest match to one's CV, and also considering his/her needs, constraints, and aspirations.A few of the things that a person may consider while doing this matching are: a) required skills mentioned in the JDs and skills possessed by self, b) current salary versus salary offered in the new job, c) future prospects after joining the new job, etc. Some of the entities are easy to extract from a JD, for example, the salary offered in a job. However, some other entities, for example, skill extraction (are Python and Java an animal and an island in Indonesia, respectively, or two object-oriented programming languages) and future prospects of a company (it is subjective as well as dependent upon marketconditions), need serious consideration.Though tremendous progress has been made in general purpose searchengines, job searchengines have ma de only

LITERATURE SURVEY

A candidate acquires skills through formal education, vocation, internships, and/or previous jobs' experience. In due course of time, the candidate may start identifying (new) relevant jobs based on the basis of these acquired skills. The key function of ajob search engine is to help the candidate by recommending those jobs which are the closest match to the candidate's existing skill set. This recommendation can be providedby matching skills of the candidate with the skills m entioned in the available JDs. A common approach while doing a skill match is to use standard keyword matching or information retrieval framework as explained in Salton and Buckley (1988). A few challenges of this kind of approaches are: a) The skill may be mentioned in different forms or in terms of synonyms (e.g. cp lusplus, c++; programming, scripting, etc.) in CVs and JDs, b) There could be skills that may not be specified in a candidate's profile or a JD, but can be easily determined by businessknowledge (for example,'java' being an object-oriented programming (OOP) language, its experience also indicates experience of OOP), and c) A skill could be an out of dictionary skill, that is, a not-so-common skill-term missing in the dictionary or from a new unseen domain for which the system may not have skills. A framework for skill extraction and normalization was proposed in Zhao et al. (2015). In this paper, a taxonomy of skill was built and Wikipedia was utiliz ed for skill normalization. In Kivimaki et al. (2013),authors prop osed a system for skill extraction from documents primarilytarg eting towards hiring and capacitymanagement in an organizatio n. The system first computes similarities between an input document andthe texts of Wikipedia pages and then uses a biased, hub-avoiding version of the Spreading Activation algorithm on the Wikipedia graph to as sociatethe input documentwith skills. Colucciet al. (2003) intro duced the concept of implicit skills.

Inspired by their work we have explored a new method in this paper to mine implicitskillsusing word and document embeddings. In Lau and Sure (2002),authors described a methodology for application-driven development of ontologies, with a sample instantiation of the methodology for skills ontology development. In Bastian et al. (2014),the team at LinkedIn built a large-scale topic extraction pipeline that included constructing a folksonomy of skills and expertise and implementing an inference and recommender system for skills. The main idea of a job recommendation system is to provide a set of (job) recommendations in response to a user’s current profile. In these systems, the users typically can upload their skills or resume or their job search criterion; similarly, the employers or their agents can upload JDs or skills set needed etc along with information such as location, position and other job specific details.

We mined the web to extract a heterogeneous mixture of JDs from various open-source websites. The entire dataset consists of 1.1 Million mined JDs. It has a substantial mix from multiple domains like IT/Software, Health-care, Recruiting, Education and 48 other such domains. This data is used to train our Word2Vec and Doc2Vec models which are explainedfurther in Section 4. Since no standard large open source dataset exists for the task of CV toJD matching, we approached a research team (Maheshwary and Misra (2018)) who had worked on this problem using deep Siamese Network. The dataset borrowed from them consists of 1314 resumes which came in as a part of summer research intern application at their company and a set of 3809 JDs from various domains. We have used this dataset forour full job recommender system evaluation so that we can compare our results with some existing published results.

REFERENCES

1. Al-
Otaibi, S. T., and Ykhlef,M. 2012. A survey of jobrecommendersy stems. International Journal of Physical Sciences7(29):5127–5142.

2.Paparrizos, I. K.; Cambazoglu, B. B.; and Gionis, A. 2011. Machine learned jobrecommendation. In RecSys.

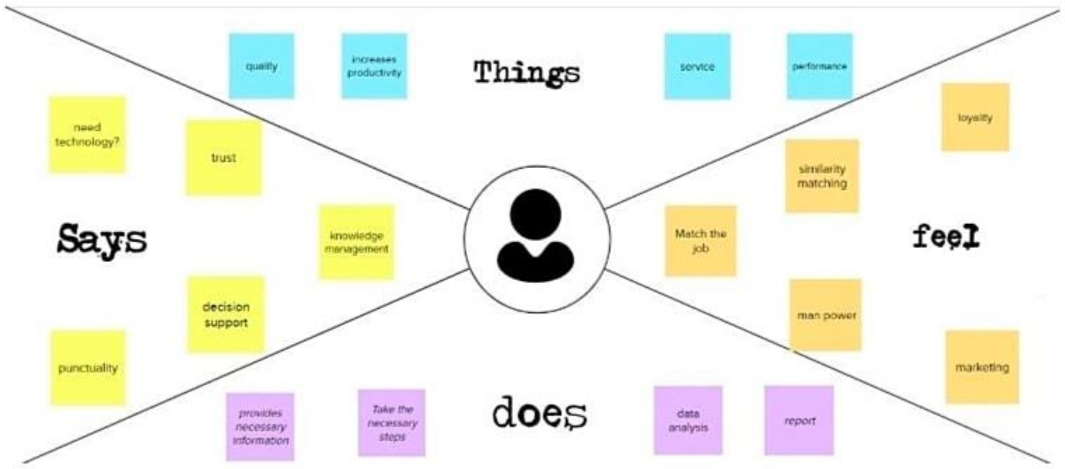
3.Lau, T., and Sure, Y. 2002. Introducing ontology-
based skills management at alargeinsurance company. In In Pr oceedings of the Modellierung 2002, 123−134.

4 .Maheshwary, S., and Misra, H. 2018. Matching resumes to jobs via
deep siamese network. In Companion of the The Web Conferen ce 2018 on The Web Conference2018, 87 88. International Worl dWide Web Conferences Steering Committee.

2.2 EMPATHIZE

PROBLEM STATEMENT:

A lot of people who have skills searching for a job. Sometimes,findinga Job that
matchestheir skills is very complicated. Thus, they need to id entify their job openings by asking company persons and traveling to new places. For that problem, our project helps people to create a profile and portfolios to mention their skills and qualities and based on that our
project recommends the available job openings which are ma tchedto your profile.It helps to solve the complexity of finding a job for an Individual.



2.3 IDEATION

PROBLEM STATEMENT:

A lot of people who have skills searching for a job. Sometimes, finding a Job that matches their skills is very complicated. Thus, they need to identify theirjob openings by asking company persons and traveling to new places. For that problem, our project helps people to create a profile and p ortfolios to mentiontheir skills and qualities and based on that our project recommends the available job openings which are matched to your profile. It helps to solve the complexity of finding a job foran Individual

IDEAS:

Vengateshwaran.H:

- Identifying the people who are having skills through the app byverification
- Recommend the people to the companies

Saran.M:

- Advertise the companies through the app
- Make a priority list for companies and job offerby date

Sathishkumar.R:

- Analysing the company’s data and creating a filtering-based search forthe people
- Increase the scalability and ensure the right recommendation for theuser

Sasikumar.S:

- Connect industry peopleand users throughthe application
- User-friendly application

Best Ideas:

Connect industry peopleand users throughthe application
User-friendly application
Increase the scalability and ensure the right recommendation for the user
Make a priority list for companies and job offers by date
Identifying the people who are having skills through the app by verification

3.PROJECT DESIGN PHASE 1

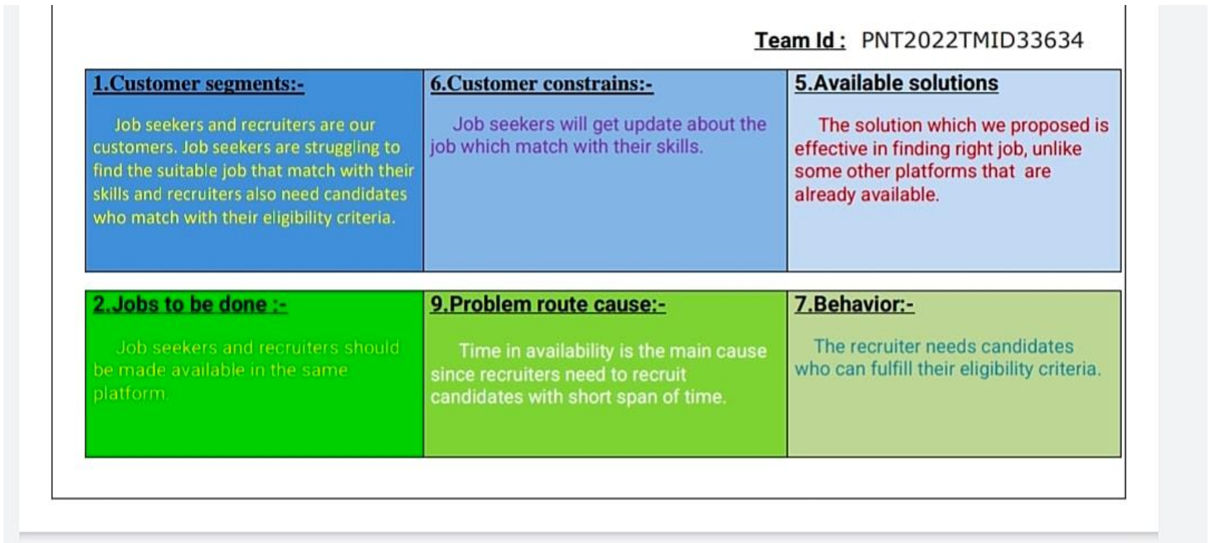
3.1Proposed Solution Template:

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

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	A lot of people who have skills searching for a job. Sometimes, finding a Job that matches their skills is very complicated. Thus, they needto identify their job openings by asking company personsand trave ling to newplaces.
2.	Idea / Solution description	Creating a web application for recommending job openi ngs to the users who have th e

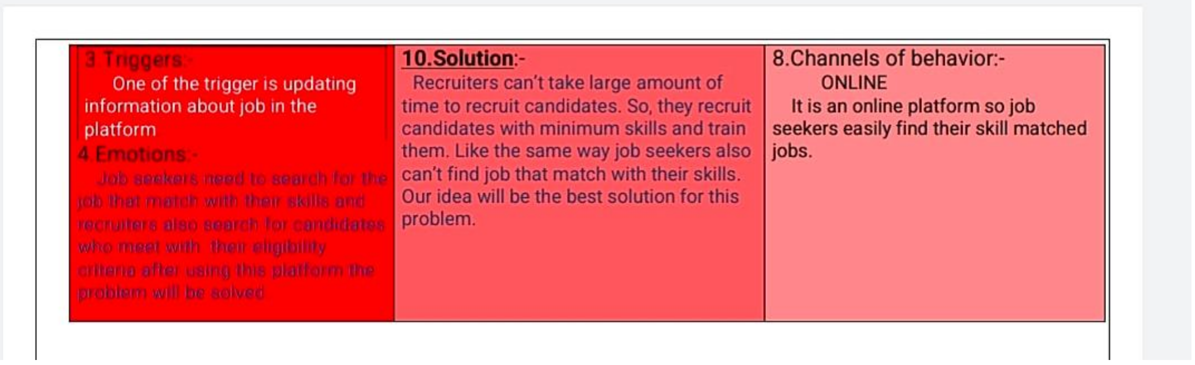
		required skills
3.	Novelty / Uniqueness	Users can findthe right job recommendations and organizations also get the right employeesfor the valuable job.
4.	Social Impact/ Customer Satisfaction	It decreases the chaos in finding the required job. It connects organizations and people whoneed jobs around the world.
5.	Business Model(Revenue Model)	Assigning a small penny for users and organizations if the users get the job or organizations get the employees and showing the recommend ed advertisements in application
6.	Scalability of the Solution	We can change the scalability of the applicationby maintaining data in servers efficiently

3.2 PROBLEM SOLUTION FIT



Team Id : PNT2022TMID33634

1.Customer segments:- Job seekers and recruiters are our customers. Job seekers are struggling to find the suitable job that match with their skills and recruiters also need candidates who match with their eligibility criteria.	**6.Customer constrains:-** Job seekers will get update about the job which match with their skills.	**5.Available solutions** The solution which we proposed is effective in finding right job, unlike some other platforms that are already available.
2.Jobs to be done :- Job seekers and recruiters should be made available in the same platform.	**9.Problem route cause:-** Time in availability is the main cause since recruiters need to recruit candidates with short span of time.	**7.Behavior:-** The recruiter needs candidates who can fulfill their eligibility criteria.



3 Triggers:- One of the trigger is updating information about job in the platform 4 Emotions:- Job seekers need to search for the job that match with their skills and recruiters also search for candidates who meet with their eligibility criteria after using this platform the problem will be solved	**10.Solution**:- Recruiters can't take large amount of time to recruit candidates. So, they recruit candidates with minimum skills and train them. Like the same way job seekers also can't find job that match with their skills. Our idea will be the best solution for this problem.	8.Channels of behavior:- ONLINE It is an online platform so job seekers easily find their skill matched jobs.

3.3 SOLUTION ARCHITECTURE

Example - Solution Architecture Diagram:

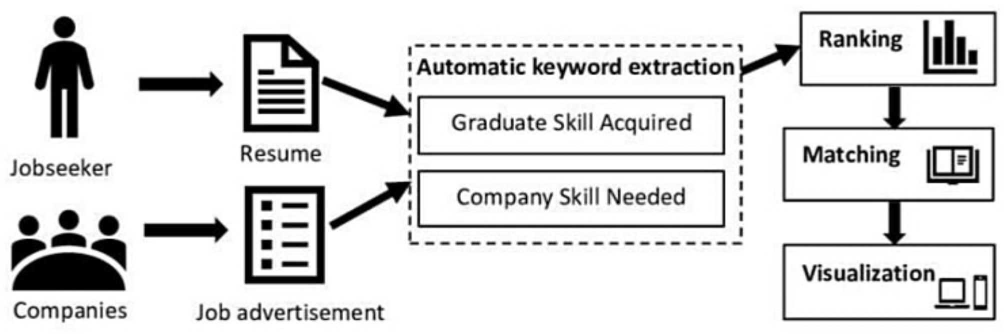


Figure 1: Architecture of skill and job recommender application

Reference: https://www.researchgate.net/figure/The-proposed-job-recommender-systemworkflow_fig2_332140232

4.PROJECT DESIGN PHASE 2

4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solu tion.

FR No.	Functional Requi rement (Epic)	Sub Requirement (Story/ Sub-Task)
FR-1	User Registration	Registration through Form. Registration through Gmail. Registration thr ough Application.
FR-2	User Confirmatio n	Confirmation via Email. Confirmati on via OTP.
FR-3	User Login	Login using credentials.
FR-4	User Application	Search for desiredcompany.
FR-5	User Profile	Complete user profileby providing pers onal details.

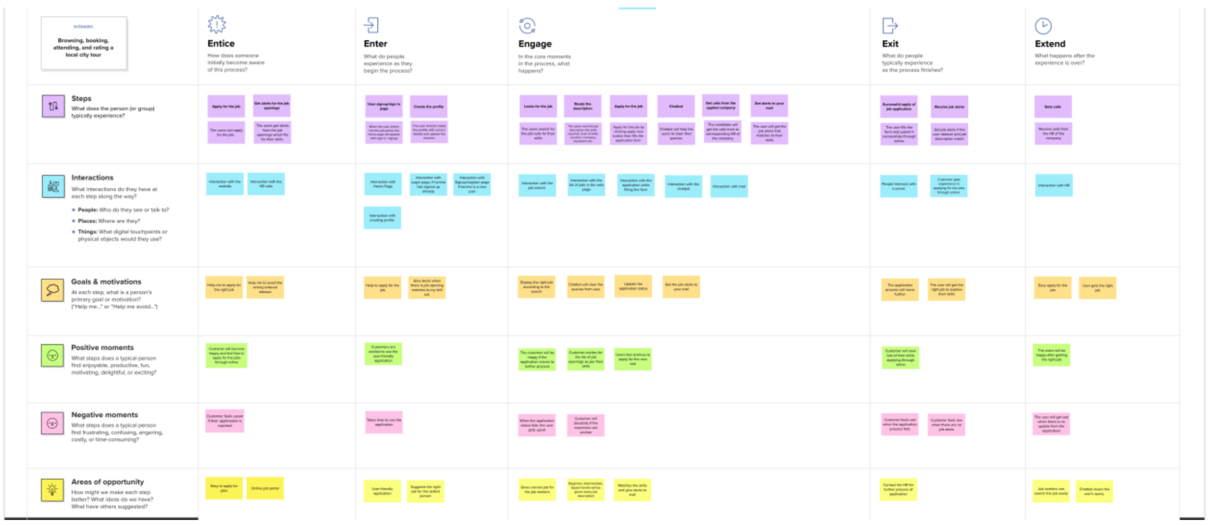
FR-6	User Application	User applies for the desired company.

Non-functional Requirements:

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

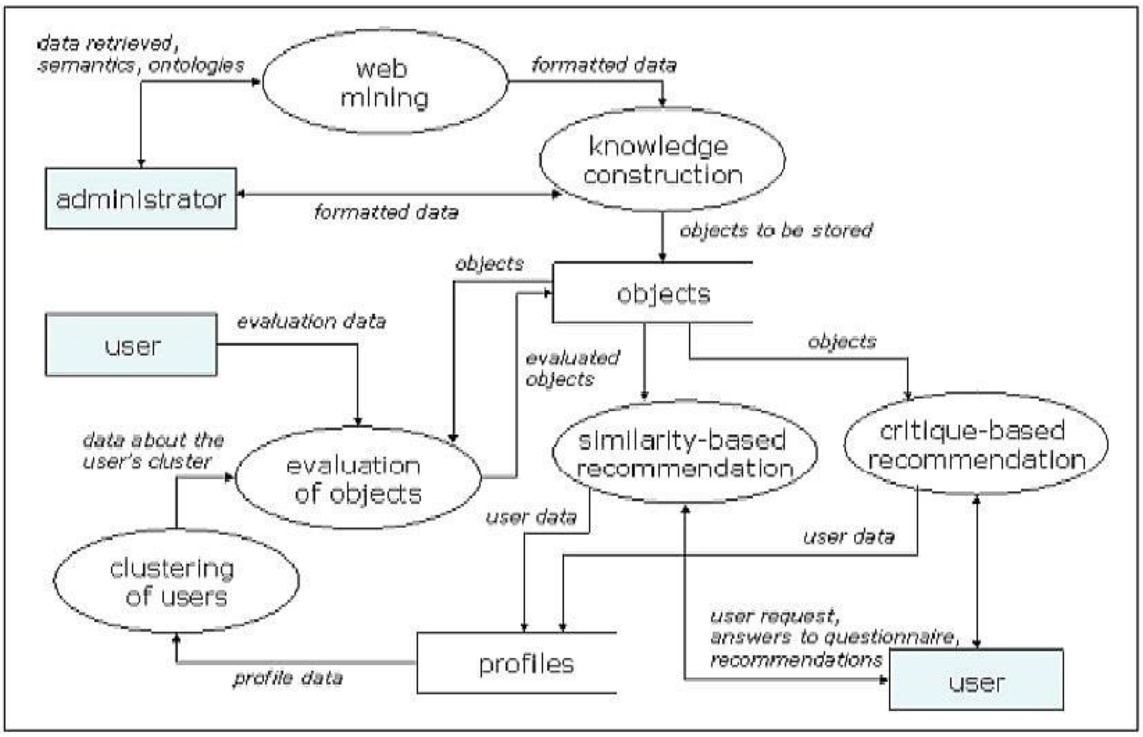
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	1. User-Friendly Application.
NFR-2	**Security**	1. End-to-End Encryption.
NFR-3	**Reliability**	1. Based on personalised skillsets.
NFR-4	**Performance**	1. Analysing the skillsets of the user to ensure ourrecommendations reach them better.
NFR-5	**Availability**	1. 24/7 chatbot support✓ 24/7 chatbot support.
NFR-6	**Scalability**	1. Reaching the on-scale requirement of the user.

4.2 CUSTOMER JOURNEY


SCENARIO
Browsing, booking, attending, and rating a local city tour
Entice
How does someone initially become aware of this process?
Enter
What do people experience as they begin the process?
Engage
In the core moments in the process, what happens?
Exit
What do people typically experience as the process finishes?
Extend
What happens after the experience is over?
Steps
What does the person (or group) typically experience?
Interactions
What interactions do they have at each step along the way?
People: Who do they see or talk to?
Places: Where are they?
Things: What digital touchpoints or physical objects would they use?
Goals & motivations
At each step, what is a person's primary goal or motivation? ("Help me..." or "Help me avoid...")
Positive moments
What steps does a typical person find enjoyable, productive, fun, motivating, delightful, or exciting?
Negative moments
What steps does a typical person find frustrating, confusing, angering, costly, or time-consuming?
Areas of opportunity
How might we make each step better? What ideas do we have? What have others suggested?
Apply for the job
Get alerts for the job openings
User signup/sign in page
Create the profile
Easy to apply for jobs
Online job portal
User-friendly application

4.3 DATA FLOW DIAGRAM

A data flow diagram is a traditional visual representation of the information flows within the 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.



4.4 TECHNOLOGY ARCHITECTURE

Technical Architecture:

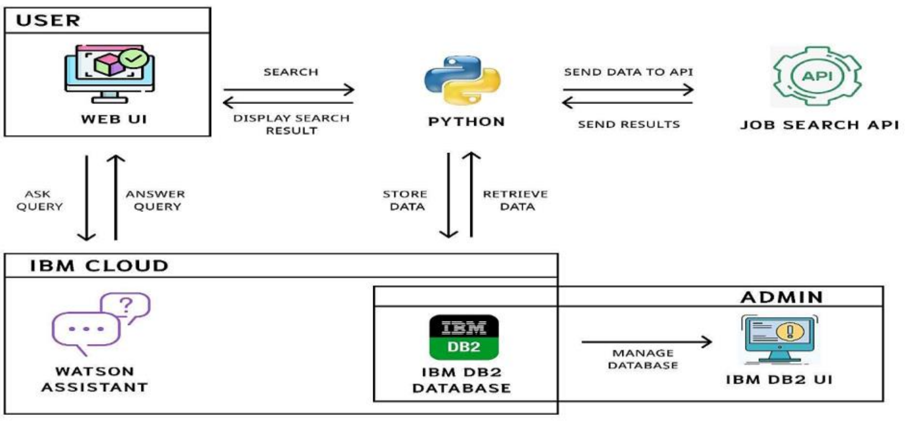


Table-1: Components & Technologies:

S.No	Component	Description	Technology
	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
	Application Logic-1	Logic for a process in the application	Java / Python
	Application Logic-2	Logic for a process in the application	IBM Watson STT service
	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
	Database	Data Type, Configurations etc.	MySQL, NoSQL ,etc.
	Cloud Database	Database Serviceon Cloud	IBM DB2, IBM Cloudant etc.
	File Storage	File storage requirements	IBM Block Storage or Other StorageService or LocalFilesystem
	External API-1	Purpose of External API used in the application	IBM Weather API,etc.
	External API-2	Purpose of External API used in the application	Aadhar API, etc.

10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System/ Cloud LocalServer Configuration: Cloud Server Configuration:	Local, CloudFoundry, Kubernetes, etc.

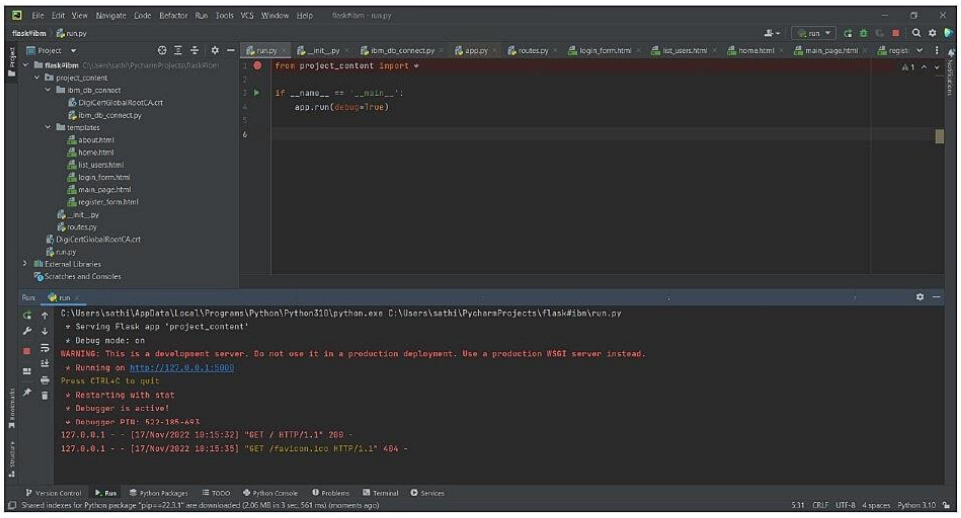
Table-2: Application Characteristics:

S.No	**Characteristics**	**Description**	**Technology**
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List allthe security / accesscontrols implemented, use of firewalls etc.	e.g., SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier,Micro-services)	Technology used
4.	Availability	Justify the availability of application (e.g.,use of load balancers, distributed servers etc.)	Technology used

5.	Performance	Design consideration for the performance of the application(number of requests per sec, use of Cache,use of CDN's) etc.	Technology used

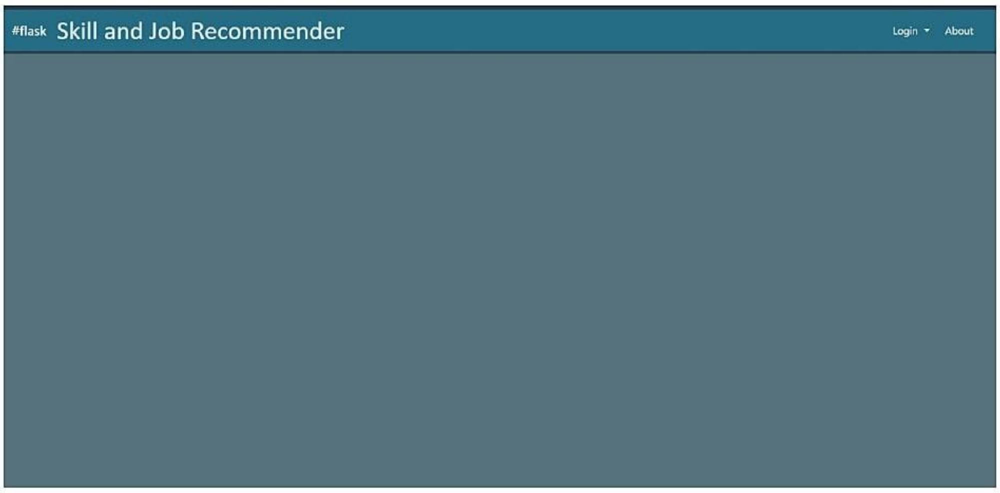
5.SETTING UP APPLICATION ENVIRONMENT

5 .1 CREATE FLASK PROJECT

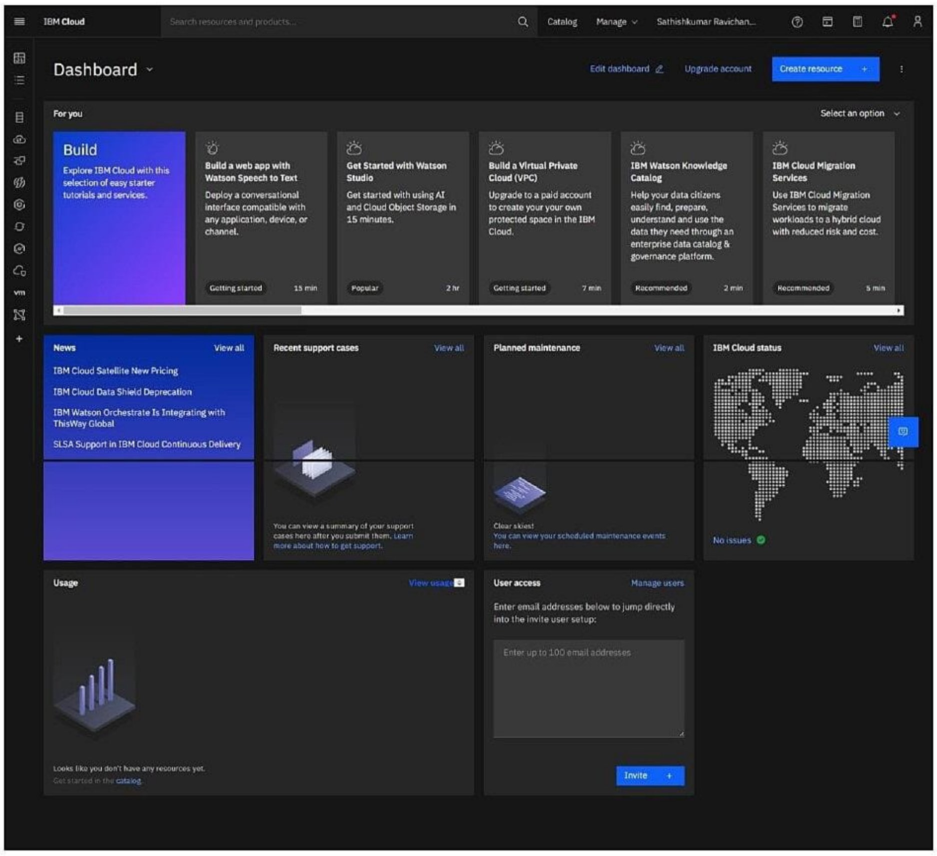


```python
from project_content import *

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

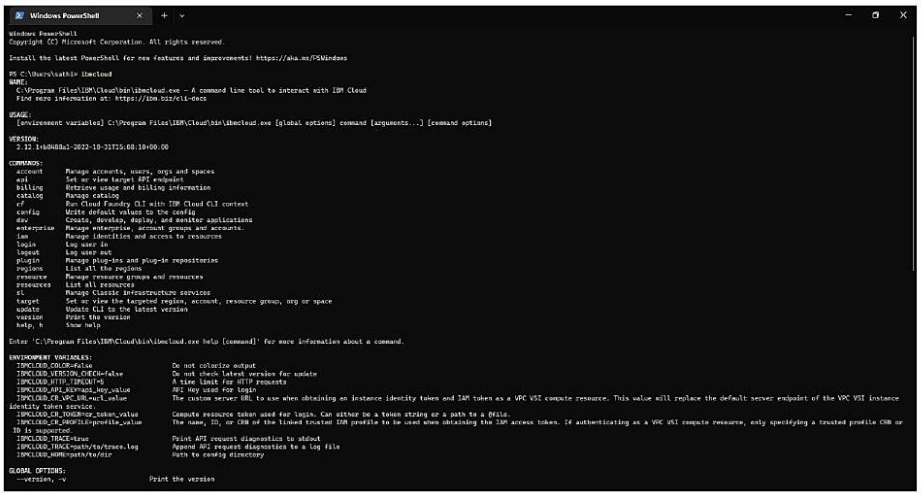
```
C:\Users\sathi\AppData\Local\Programs\Python\Python310\python.exe C:\Users\sathi\PycharmProjects\flask#ibm\run.py
 * Serving Flask app 'project_content'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 522-185-693
127.0.0.1 - - [17/Nov/2022 10:15:32] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [17/Nov/2022 10:15:35] "GET /favicon.ico HTTP/1.1" 404 -
```



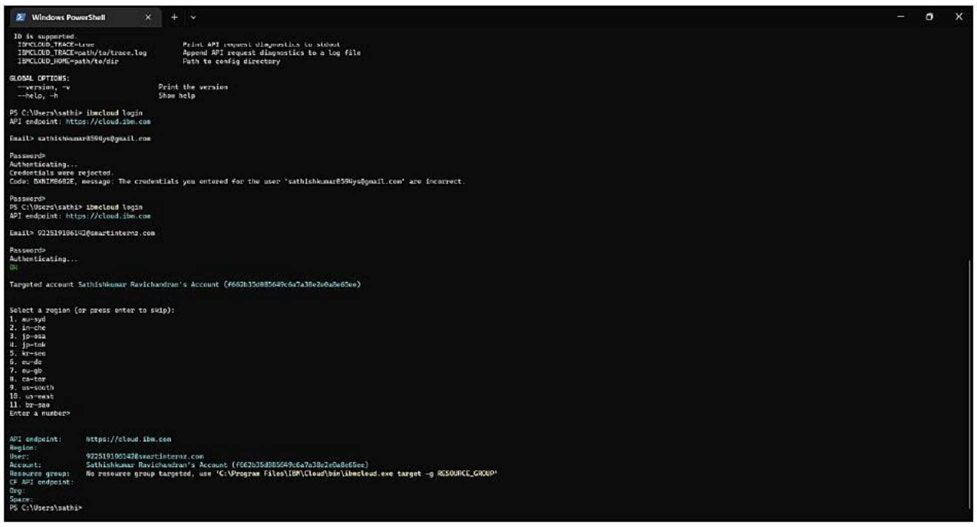
5.2 CREATE IBM CLOUD ACCOUNT


IBM Cloud
Search resources and products...
Catalog
Manage
Sathishkumar Ravichan...
Dashboard
Edit dashboard
Upgrade account
Create resource
For you
Select an option
Build
Explore IBM Cloud with this selection of easy starter tutorials and services.
Build a web app with Watson Speech to Text
Deploy a conversational interface compatible with any application, device, or channel.
Getting started
15 min
Get Started with Watson Studio
Get started with using AI and Cloud Object Storage in 15 minutes.
Popular
2 hr
Build a Virtual Private Cloud (VPC)
Upgrade to a paid account to create your your own protected space in the IBM Cloud.
Getting started
7 min
IBM Watson Knowledge Catalog
Help your data citizens easily find, prepare, understand and use the data they need through an enterprise data catalog & governance platform.
Recommended
2 min
IBM Cloud Migration Services
Use IBM Cloud Migration Services to migrate workloads to a hybrid cloud with reduced risk and cost.
Recommended
5 min
News
View all
IBM Cloud Satellite New Pricing
IBM Cloud Data Shield Deprecation
IBM Watson Orchestrate Is Integrating with ThisWay Global
SLSA Support in IBM Cloud Continuous Delivery
Recent support cases
View all
You can view a summary of your support cases here after you submit them. Learn more about how to get support.
Planned maintenance
View all
Clear skies!
You can view your scheduled maintenance events here.
IBM Cloud status
View all
No issues
Usage
View usage
Looks like you don't have any resources yet.
Get started in the catalog.
User access
Manage users
Enter email addresses below to jump directly into the invite user setup:
Enter up to 100 email addresses
Invite

5.3 INSTALL IBM CLOUD CLI

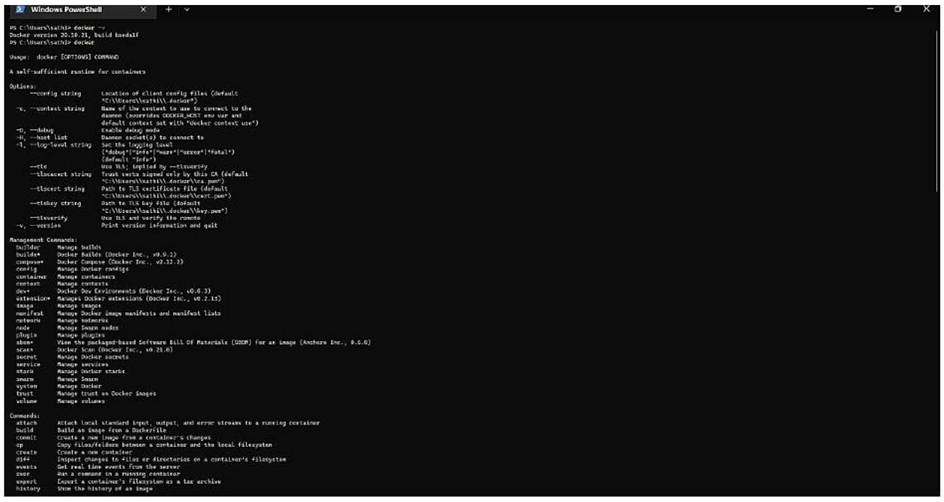


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\sathi> ibmcloud
NAME:
  C:\Program Files\IBM\Cloud\bin\ibmcloud.exe - A command line tool to interact with IBM Cloud
  Find more information at: https://ibm.biz/cli-docs

USAGE:
  [environment variables] C:\Program Files\IBM\Cloud\bin\ibmcloud.exe [global options] command [arguments...] [command options]

VERSION:
  2.12.1+b0488a1-2022-10-31T15:08:10+00:00

COMMANDS:
  account      Manage accounts, users, orgs and spaces
  api          Set or view target API endpoint
  billing      Retrieve usage and billing information
  catalog      Manage catalog
  cf           Run Cloud Foundry CLI with IBM Cloud CLI context
  config       Write default values to the config
  dev          Create, develop, deploy, and monitor applications
  enterprise   Manage enterprise, account groups and accounts.
  iam          Manage identities and access to resources
  login        Log user in
  logout       Log user out
  plugin       Manage plug-ins and plug-in repositories
  regions      List all the regions
  resource     Manage resource groups and resources
  resources    List all resources
  sl           Manage Classic infrastructure services
  target       Set or view the targeted region, account, resource group, org or space
  update       Update CLI to the latest version
  version      Print the version
  help, h      Show help

Enter 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe help [command]' for more information about a command.

ENVIRONMENT VARIABLES:
  IBMCLOUD_COLOR=false                     Do not colorize output
  IBMCLOUD_VERSION_CHECK=false             Do not check latest version for update
  IBMCLOUD_HTTP_TIMEOUT=5                  A time limit for HTTP requests
  IBMCLOUD_API_KEY=api_key_value           API Key used for login
  IBMCLOUD_CR_VPC_URL=url_value            The custom server URL to use when obtaining an instance identity token and IAM token as a VPC VSI compute resource. This value will replace the default server endpoint of the VPC VSI instance
identity token service.
  IBMCLOUD_CR_TOKEN=cr_token_value         Compute resource token used for login. Can either be a token string or a path to a @file.
  IBMCLOUD_CR_PROFILE=profile_value        The name, ID, or CRN of the linked trusted IAM profile to be used when obtaining the IAM access token. If authenticating as a VPC VSI compute resource, only specifying a trusted profile CRN or
 ID is supported.
  IBMCLOUD_TRACE=true                      Print API request diagnostics to stdout
  IBMCLOUD_TRACE=path/to/trace.log         Append API request diagnostics to a log file
  IBMCLOUD_HOME=path/to/dir                Path to config directory

GLOBAL OPTIONS:
  --version, -v                  Print the version
```

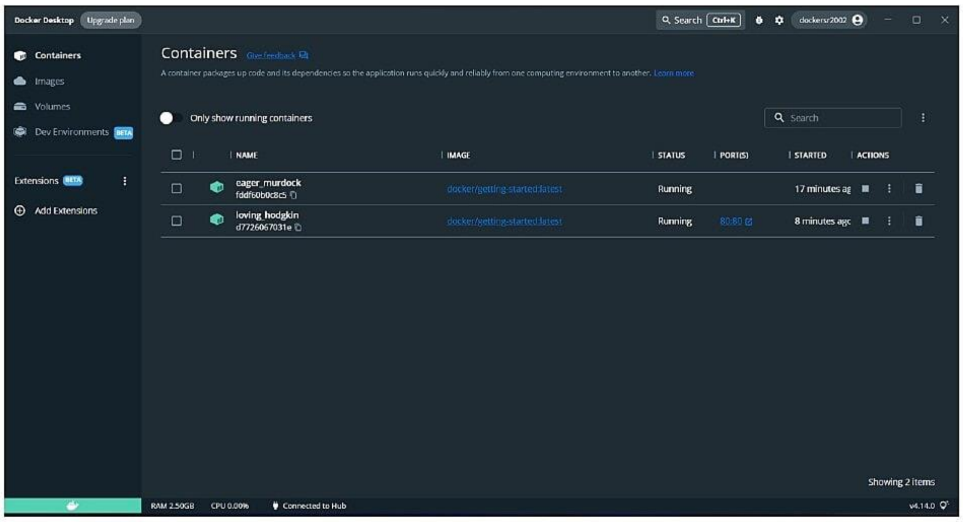


```
ID is supported.
  IBMCLOUD_TRACE=true                       Print API request diagnostics to stdout
  IBMCLOUD_TRACE=path/to/trace.log          Append API request diagnostics to a log file
  IBMCLOUD_HOME=path/to/dir                 Path to config directory

GLOBAL OPTIONS:
  --version, -v                       Print the version
  --help, -h                          Show help

PS C:\Users\sathi> ibmcloud login
API endpoint: https://cloud.ibm.com

Email> sathishkumar8594ys@gmail.com

Password>
Authenticating...
Credentials were rejected.
Code: BXNIM0602E, message: The credentials you entered for the user 'sathishkumar8594ys@gmail.com' are incorrect.

Password>
PS C:\Users\sathi> ibmcloud login
API endpoint: https://cloud.ibm.com

Email> 922519106142@smartinternz.com

Password>
Authenticating...
OK

Targeted account Sathishkumar Ravichandran's Account (f662b35d085649c6a7a38e2e0a8e65ee)


Select a region (or press enter to skip):
1. au-syd
2. in-che
3. jp-osa
4. jp-tok
5. kr-seo
6. eu-de
7. eu-gb
8. ca-tor
9. us-south
10. us-east
11. br-sao
Enter a number>


API endpoint:      https://cloud.ibm.com
Region:
User:              922519106142@smartinternz.com
Account:           Sathishkumar Ravichandran's Account (f662b35d085649c6a7a38e2e0a8e65ee)
Resource group:    No resource group targeted, use 'C:\Program Files\IBM\Cloud\bin\ibmcloud.exe target -g RESOURCE_GROUP'
CF API endpoint:
Org:
Space:
PS C:\Users\sathi>
```

5.4 DOCKER CLI INSTALLATION



```
PS C:\Users\sathi> docker -v
Docker version 20.10.21, build baeda1f
PS C:\Users\sathi> docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
      --config string      Location of client config files (default
                           "C:\\Users\\sathi\\.docker")
  -c, --context string     Name of the context to use to connect to the
                           daemon (overrides DOCKER_HOST env var and
                           default context set with "docker context use")
  -D, --debug              Enable debug mode
  -H, --host list          Daemon socket(s) to connect to
  -l, --log-level string   Set the logging level
                           ("debug"|"info"|"warn"|"error"|"fatal")
                           (default "info")
      --tls                Use TLS; implied by --tlsverify
      --tlscacert string   Trust certs signed only by this CA (default
                           "C:\\Users\\sathi\\.docker\\ca.pem")
      --tlscert string     Path to TLS certificate file (default
                           "C:\\Users\\sathi\\.docker\\cert.pem")
      --tlskey string      Path to TLS key file (default
                           "C:\\Users\\sathi\\.docker\\key.pem")
      --tlsverify          Use TLS and verify the remote
  -v, --version            Print version information and quit

Management Commands:
  builder     Manage builds
  buildx*     Docker Buildx (Docker Inc., v0.9.1)
  compose*    Docker Compose (Docker Inc., v2.12.2)
  config      Manage Docker configs
  container   Manage containers
  context     Manage contexts
  dev*        Docker Dev Environments (Docker Inc., v0.0.3)
  extension*  Manages Docker extensions (Docker Inc., v0.2.13)
  image       Manage images
  manifest    Manage Docker image manifests and manifest lists
  network     Manage networks
  node        Manage Swarm nodes
  plugin      Manage plugins
  sbom*       View the packaged-based Software Bill Of Materials (SBOM) for an image (Anchore Inc., 0.6.0)
  scan*       Docker Scan (Docker Inc., v0.21.0)
  secret      Manage Docker secrets
  service     Manage services
  stack       Manage Docker stacks
  swarm       Manage Swarm
  system      Manage Docker
  trust       Manage trust on Docker images
  volume      Manage volumes

Commands:
  attach      Attach local standard input, output, and error streams to a running container
  build       Build an image from a Dockerfile
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  events      Get real time events from the server
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  history     Show the history of an image
```



Containers Give feedback

A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. Learn more

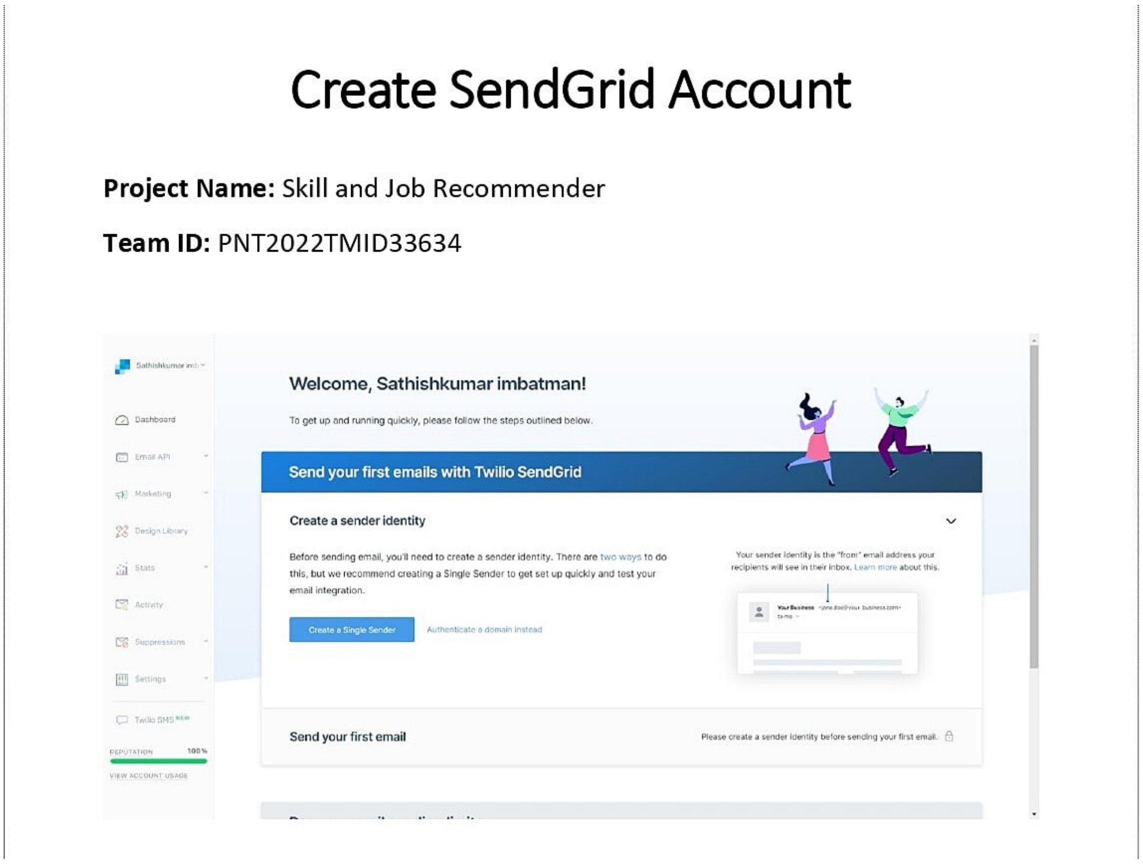
Only show running containers

NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
eager_murdock fddf60b0c8c5	docker/getting-started:latest	Running		17 minutes ag	
loving_hodgkin d7726067031e	docker/getting-started:latest	Running	80:80	8 minutes agc	

Showing 2 items

RAM 2.50GB CPU 0.00% Connected to Hub v4.14.0

5.5 CREATE AN ACCOUNT IN SENDGRID



Create SendGrid Account

Project Name: Skill and Job Recommender

Team ID: PNT2022TMID33634

Welcome, Sathishkumar imbatman!

To get up and running quickly, please follow the steps outlined below.

Send your first emails with Twilio SendGrid

Create a sender identity

Before sending email, you'll need to create a sender identity. There are two ways to do this, but we recommend creating a Single Sender to get set up quickly and test your email integration.

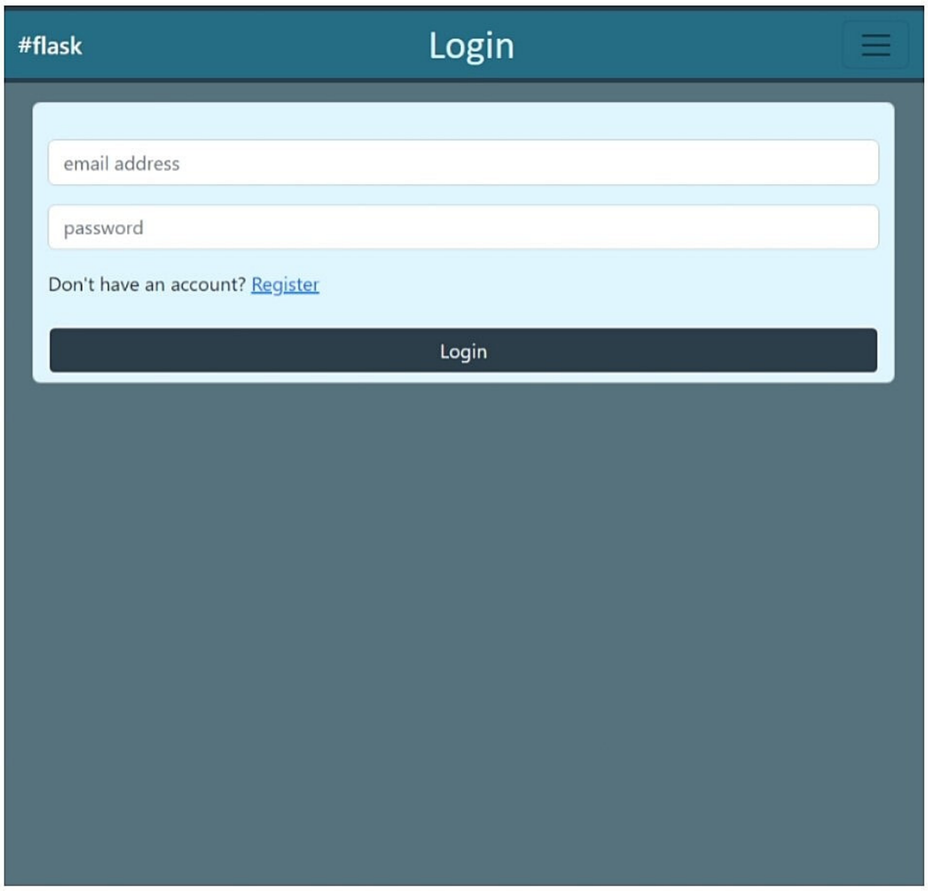
Your sender identity is the "from" email address your recipients will see in their inbox. Learn more about this.

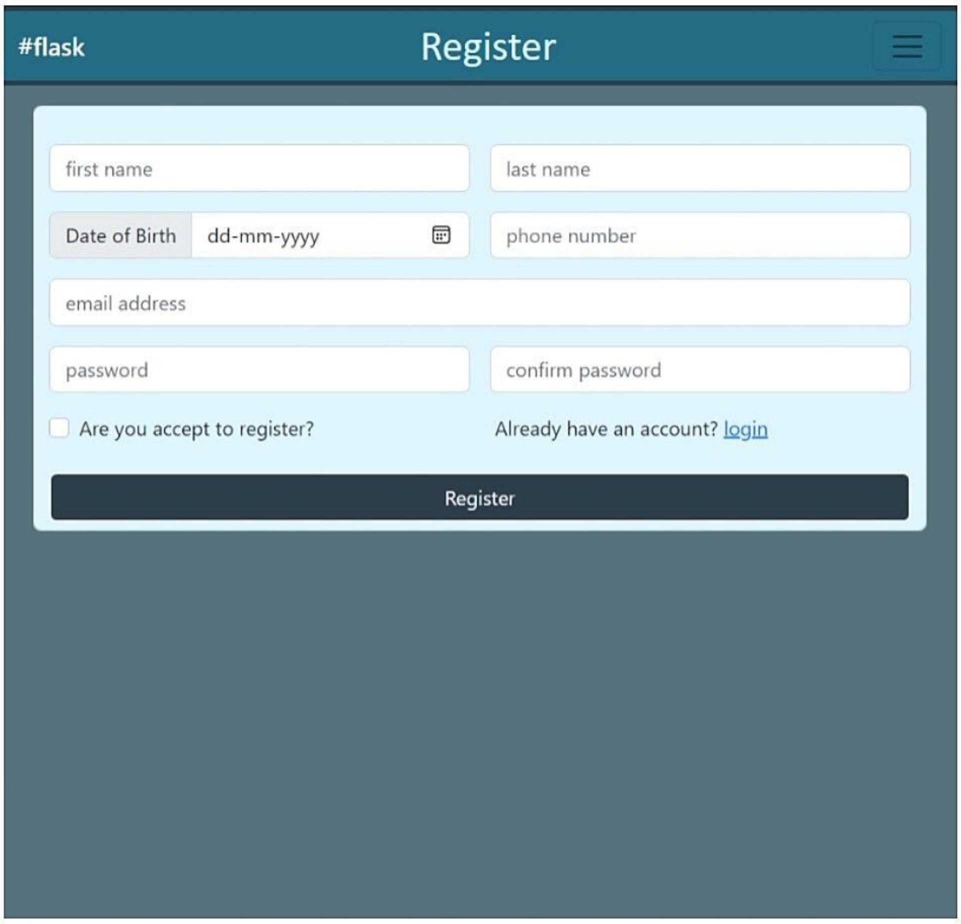
Send your first email

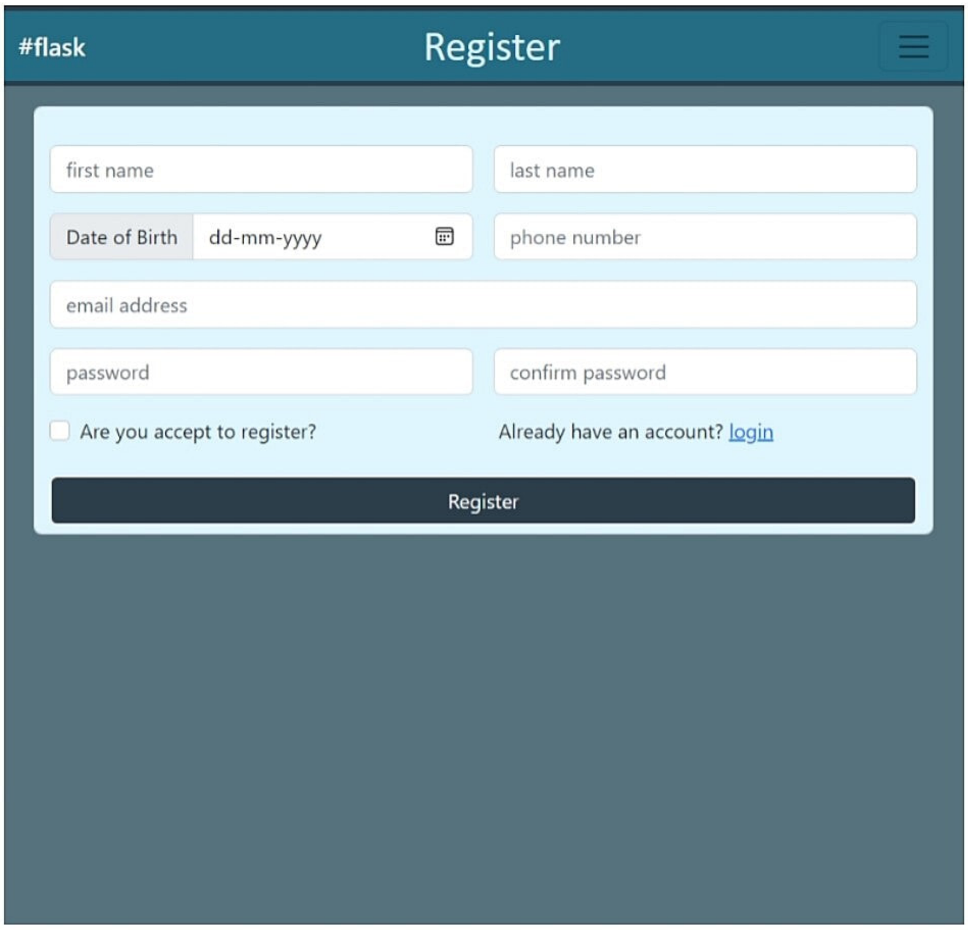
Please create a sender identity before sending your first email.

6 IMPLEMENTING WEB APPLICATION

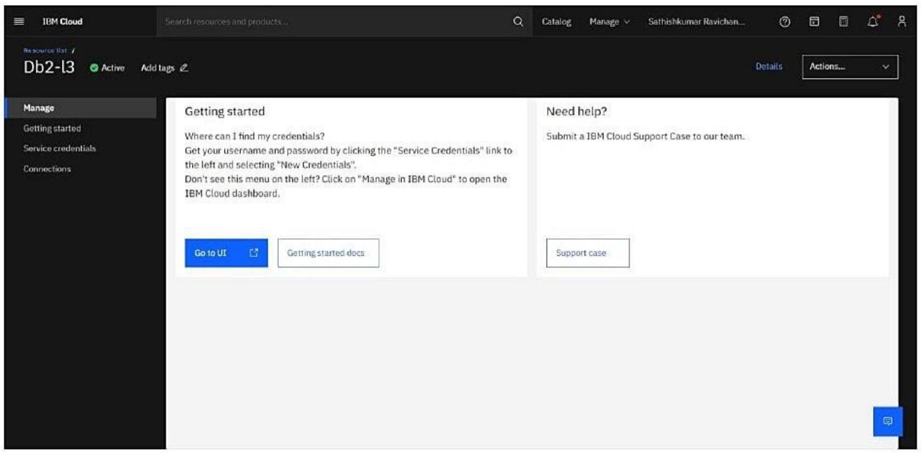
6.1 CREATE UI TO INTERACT WITH APPLICATION


#flask
Login
email address
password
Don't have an account? Register
Login


#flask
Register
first name
last name
Date of Birth
dd-mm-yyyy
phone number
email address
password
confirm password
Are you accept to register?
Already have an account? login
Register



6.3 CREATE IBM DB2 AND CONNECT WITH PYTHON

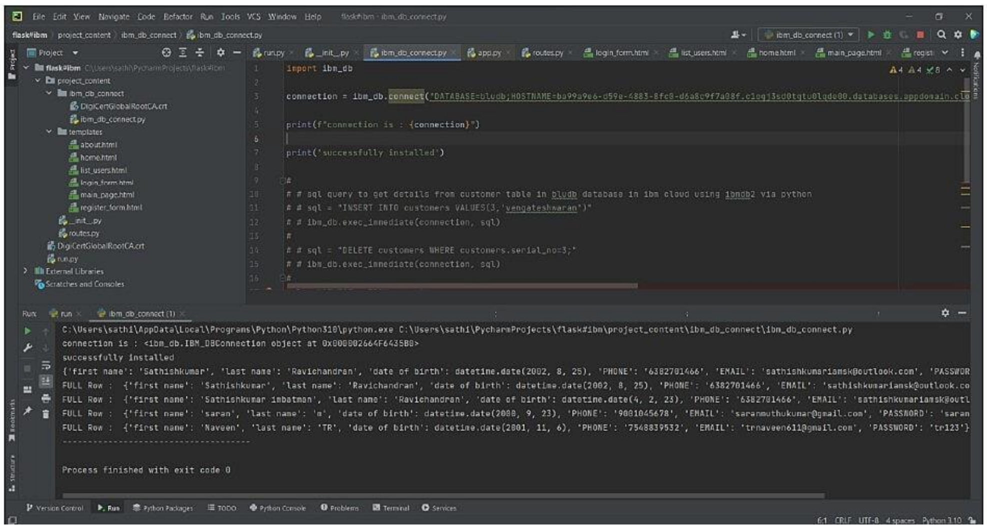


Getting started

Where can I find my credentials?
Get your username and password by clicking the "Service Credentials" link to the left and selecting "New Credentials".
Don't see this menu on the left? Click on "Manage in IBM Cloud" to open the IBM Cloud dashboard.

Need help?

Submit a IBM Cloud Support Case to our team.



```
import ibm_db

connection = ibm_db.connect('DATABASE=bludb;HOSTNAME=ba99a9e6-d59e-4883-8fc8-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.clo

print(f"connection is : {connection}")

print('successfully installed')

#
# # sql query to get details from customer table in bludb database in ibm cloud using ibmdb2 via python
# # sql = "INSERT INTO customers VALUES(3,'vengateshwaran')"
# # ibm_db.exec_immediate(connection, sql)
#
# # sql = "DELETE customers WHERE customers.serial_no=3;"
# # ibm_db.exec_immediate(connection, sql)
#
```

```
C:\Users\sathi\AppData\Local\Programs\Python\Python310\python.exe C:\Users\sathi\PycharmProjects\flask#ibm\project_content\ibm_db_connect\ibm_db_connect.py
connection is : <ibm_db.IBM_DBConnection object at 0x000002664F6435B0>
successfully installed
{'first name': 'Sathishkumar', 'last name': 'Ravichandran', 'date of birth': datetime.date(2002, 8, 25), 'PHONE': '6382701466', 'EMAIL': 'sathishkumariamsk@outlook.com', 'PASSWOR
FULL Row :  {'first name': 'Sathishkumar', 'last name': 'Ravichandran', 'date of birth': datetime.date(2002, 8, 25), 'PHONE': '6382701466', 'EMAIL': 'sathishkumariamsk@outlook.co
FULL Row :  {'first name': 'Sathishkumar imbatman', 'last name': 'Ravichandran', 'date of birth': datetime.date(4, 2, 23), 'PHONE': '6382701466', 'EMAIL': 'sathishkumariamsk@outl
FULL Row :  {'first name': 'saran', 'last name': 'm', 'date of birth': datetime.date(2000, 9, 23), 'PHONE': '9001045678', 'EMAIL': 'saranmuthukumar@gmail.com', 'PASSWORD': 'saran
FULL Row :  {'first name': 'Naveen', 'last name': 'TR', 'date of birth': datetime.date(2001, 11, 6), 'PHONE': '7548839532', 'EMAIL': 'trnaveen611@gmail.com', 'PASSWORD': 'tr123'}
-----------------------------------

Process finished with exit code 0
```

7 INTEGRATING SENDGRID SERVICE

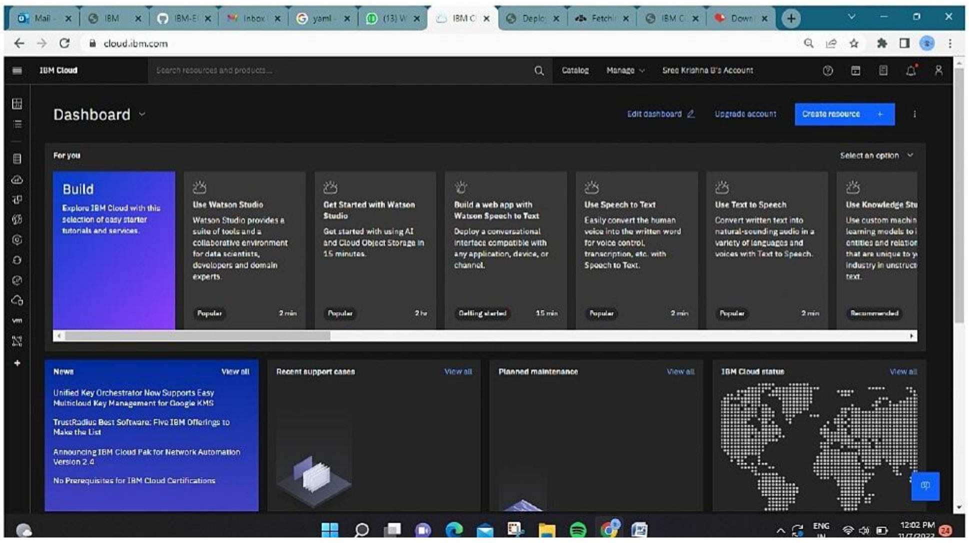
```
# using SendGrid's Python Library
# https://github.com/sendgrid/sendgrid-python
import os
from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail

# 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} ")
        print(f"Response Body: {body} ")
        print(f"Response Headers: {headers} ")
        print("Message Sent!")
    except Exception as e:
        print("Error: {0}".format(e))
    return str(response.status_code)

if __name__ == "__main__":
    SendEmail(to_email=input("Email address to send to? "))
```

8 DEPLOYMENT OF APP IN IBM CLOUD

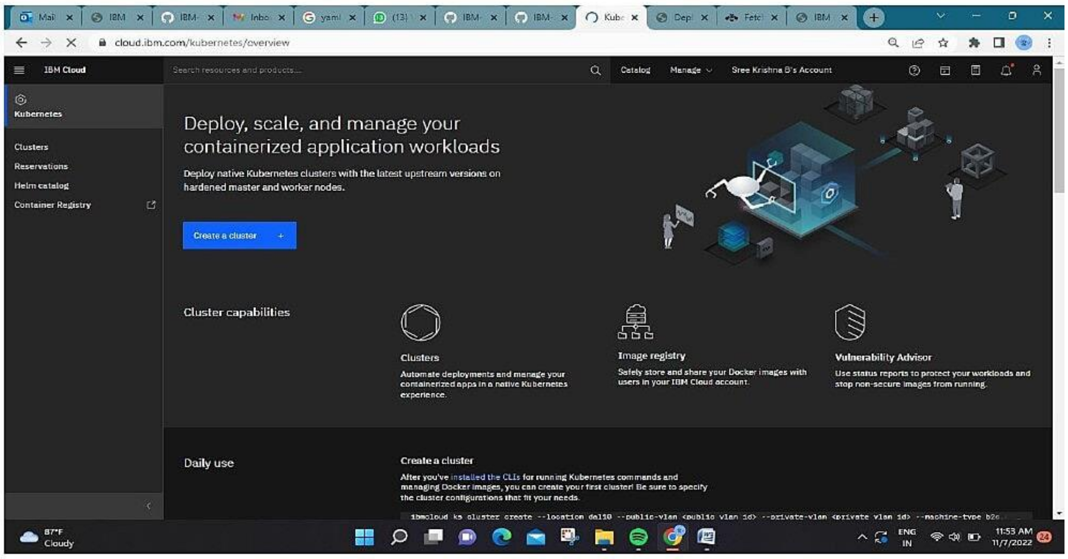
DEPLOY ON KUBERNETES

Create a Kubernetes cluster

- Sign in to your IBM CloudDashboard.
- Open **IBM Kubernetes Service**.

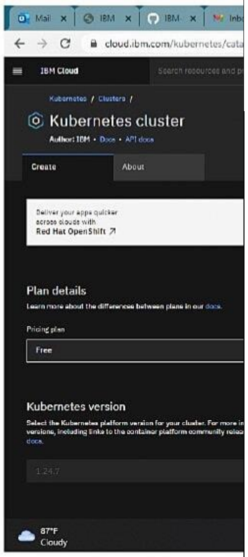


- Click **Create Cluster**.



- Selectthe **Region**where you want
  to deploy the cluster, type in a **name** for your cluster,t
  hen click **Create Cluster**.
- Select the appropriate cluster type dependingon your acco
  unt.

- It takes some time for
  the cluster to get ready (around 30 minutes).



cloud.ibm.com/kubernetes/cata

IBM Cloud

Kubernetes / Clusters /

Kubernetes cluster

Author: IBM • Docs • API docs

Create | About

Deliver your apps quicker across clouds with Red Hat OpenShift

Plan details

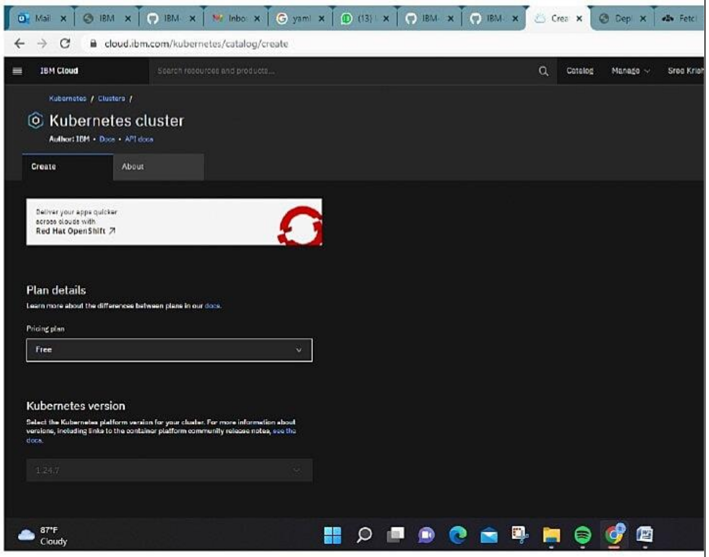
Learn more about the differences between plans in our docs.

Pricing plan: Free

Kubernetes version

Select the Kubernetes platform version for your cluster. For more information about versions, including links to the container platform community release notes, see the docs.

1.24.7



cloud.ibm.com/kubernetes/catalog/create

IBM Cloud

Kubernetes / Clusters /

Kubernetes cluster

Author: IBM • Docs • API docs

Create | About

Deliver your apps quicker across clouds with Red Hat OpenShift

Plan details

Learn more about the differences between plans in our docs.

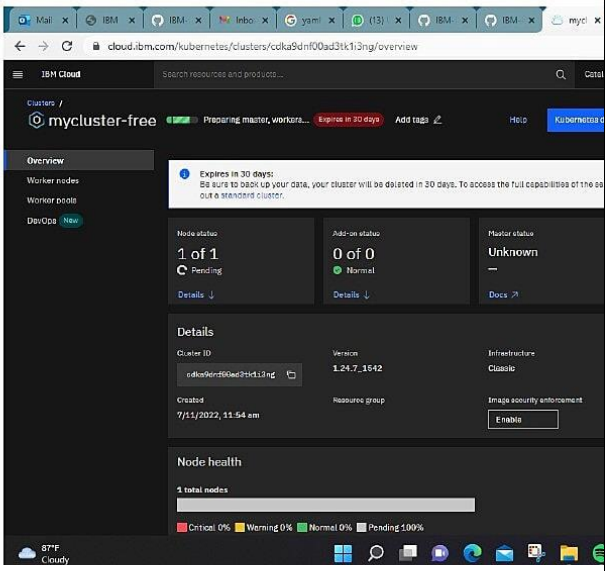
Pricing plan: Free

Kubernetes version

Select the Kubernetes platform version for your cluster. For more information about versions, including links to the container platform community release notes, see the docs.

1.24.7

- Once the cluster is ready,click on your cluster's name and you will b e redirected to a new page with inf ormation about your clusterand worker node.



cloud.ibm.com/kubernetes/clusters/cdka9dnf00ad3tk1i3ng/overview

IBM Cloud

Clusters /

mycluster-free — Preparing master, workers... — Expires in 30 days — Add tags

Overview | Worker nodes | Worker pools | DevOps

Expires in 30 days:
Be sure to back up your data, your cluster will be deleted in 30 days. To access the full capabilities of the se... out a standard cluster.

Node status: 1 of 1 — Pending

Add-on status: 0 of 0 — Normal

Master status: Unknown

Details

Cluster ID: cdka9drdf00ad3tk1i3ng

Version: 1.24.7_1542

Infrastructure: Classic

Created: 7/11/2022, 11:54 am

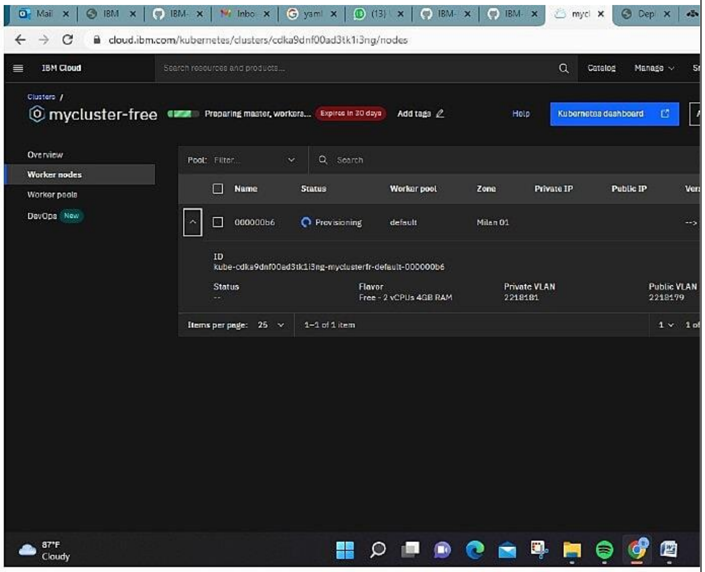
Resource group:

Image security enforcement: Enable

Node health

1 total nodes

🟥 Critical 0% 🟨 Warning 0% 🟩 Normal 0% ⬜ Pending 100%

	• Click on the **Worker Nodes** tab to note the cluster's Public IP. 

cloud.ibm.com/kubernetes/clusters/cdka9dnf00ad3tk1i3ng/nodes

Clusters /
mycluster-free — Preparing master, workers... — Expires in 30 days

Name	Status	Worker pool	Zone	Private IP	Public IP
000000b6	Provisioning	default	Milan 01		

ID
kube-cdka9dnf00ad3tk1i3ng-myclusterfr-default-000000b6

Status	Flavor	Private VLAN	Public VLAN
--	Free - 2 vCPUs 4GB RAM	2218181	2218179

Items per page: 25 — 1–1 of 1 item

9 PROJECT PLANNING PHASE

9.1 MILESTONE AND ACTIVITY LIST

PREREQUISITES:

1. Python IDLE
2. Flask
3. IBM Cloud
4. Docker

1. **IDEATION PHASE**
   a. Literature Survey

1. Empathize
2. Defining ProblemStatement

2. Create IBM DB2 and connect with Python

    a. IBM DB2 with Python

6. **INTEGRATING SENDGRIDSERVICE**

1. Sendgrid integration with Python code

7. **DEPLOYMENT OF APPIN IBM CLOUD**

1. Containerize the app

    a. Docker image creation

1. Creating docker image for flask app

2. Upload imageto IBM container registry

3. Deploy in Kubernetes cluster

8. **PROJECT PLANNINGPHASE**

1. Prepare Milestone and Activity list

2. Sprint DeliveryPlan

9. **PROJECT DEVELOPMENT PHASE**

1. Project development-Delivery of sprint-1

2. Project development-Delivery of sprint-2

3. Project development-Delivery of sprint-3

4. Project development-Delivery of sprint-4

10. **DEVELOPING A CHATBOT**

1. Building chatbotand integrate to app

9.2 SPRINT DELIVERY PLAN

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

Sprint	Functional Requirement (Epic)	User Story Number	User Story/ Task	Story Points	Priority	Team Members
Sprint-1	Registration	SRA-1	As a user, I can register for the application by entering my Name, email, password, confirming my password, Age, Blood Group.	3	High	Vengateshwaran H Sathishkumar R
Sprint-3		SRA-2	As a user, I will receiveconfirmation email once I have registered for the application	3	Medium	Saran M Sasikumar S Sathishkumar R
Sprint-2		SRA-3	As a user,I can register for the application through Gmail	5	Medium	Vengateshwaran H Sathishkumar R

Sprint-1	Login	SRA-4	As a user, I can log into the application by entering emailand password	1	High	Vengateshwaran H Sasikumar S
Sprint-3		SRA-5	As a user,I can reset my password using Forgot Password option	4	Medium	Vengateshwaran H Sathish kumar R
Sprint-4		SRA-6	As a user,I can view my pastrequests for jobs	2	Low	Saran M
Sprint-4		SRA-8	As a user,I can closepast requests I made for jobs	2	Low	Sasikumar S
			As a user,I can viewthe homepage of the website	2		Vengateshwaran H

Sprint-1	Home Page	SRA-10			Medium	Sasikumar S
Sprint - 1	About Page	SRA-12	As a user, I can viewthe about pageon the websiteand get information related to jobs	2	Medium	Sathish kumar R
Sprint - 2	Register	SRA-13	As a user,I can register.	3	High	Sasikumar S

Sprint	**Functional Requirement (Epic)**	**User Sto**	**User Story/ Task**	**Story Points**	**Priority**	**Team Me**

		ryNumber				mbers
Sprint-2	Send Request	SRA-14	As a user, I can raisea request for job with specific requirements through the request page.	2	High	Vengateshwaran H
Sprint-3	View Requests	SRA-15	As a user,I can view requests forjob verified by admin	4	Medium	Sathishkumar R Vengateshwaran H Sasi

						kumar S
Sprint-4	Maintenance	SRA-16	As an admin,I can maintain the databases involved	2	Medium	Vengateshwaran H
Sprint-2	Handle Requests	SRA-17	As an admin,I can viewall requests for job	1	High	Sathish kumar R
Sprint-4		SRA-18	As an admin, I can deleterequests that arepast some timeperiod or havebeen closed	3	Low	SasikumarS SaranM

Sprint-2	Solving User Queries	SRA-19	Creating an ChatBotthat helps to solve thequeries of the user.	2	High	Vengateshwaran H

Project Tracker,Velocity & BurndownChart: (4 Marks

Sprint	Total StoryPoints	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned EndDate)	Sprint ReleaseDate (Actual)
Sprint-1	8	5 Days	27 Oct 2022	31 Oct 2022	20	31 Oct 2022

Sprint-2	13	6 Days	1 Nov 2022	06 Nov 2022	20	06 Nov 2022
Sprint-3	11	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	9	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

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

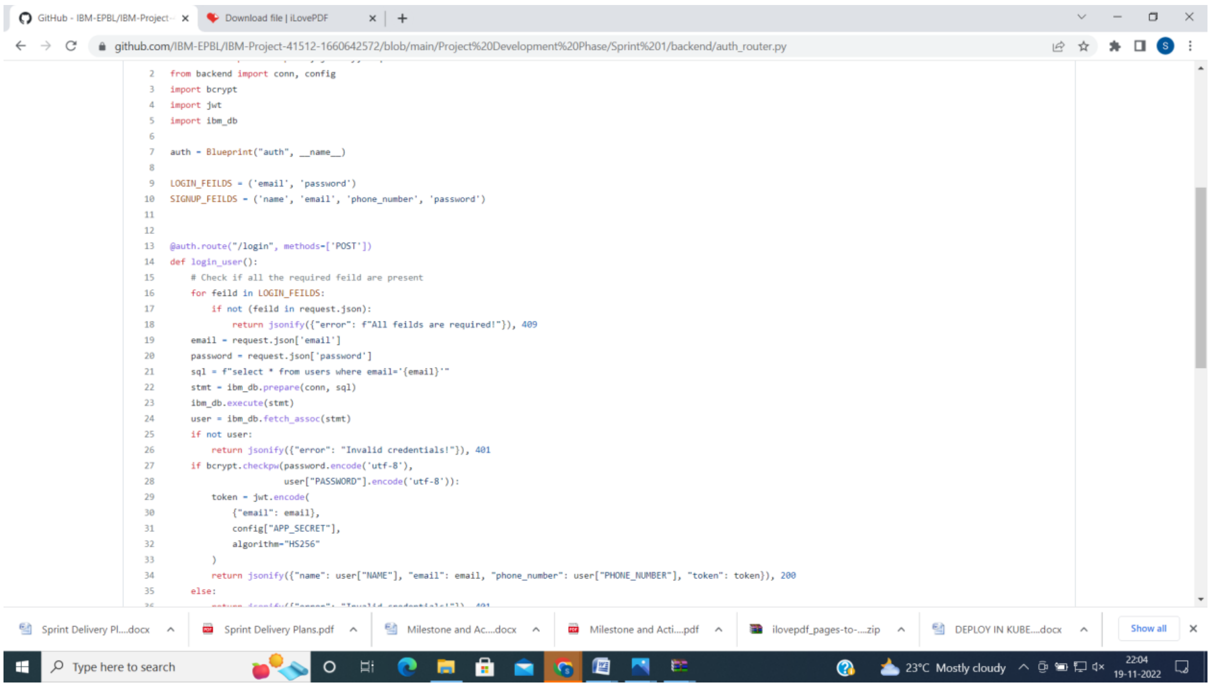
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

Burndown Chart: [illegible] project

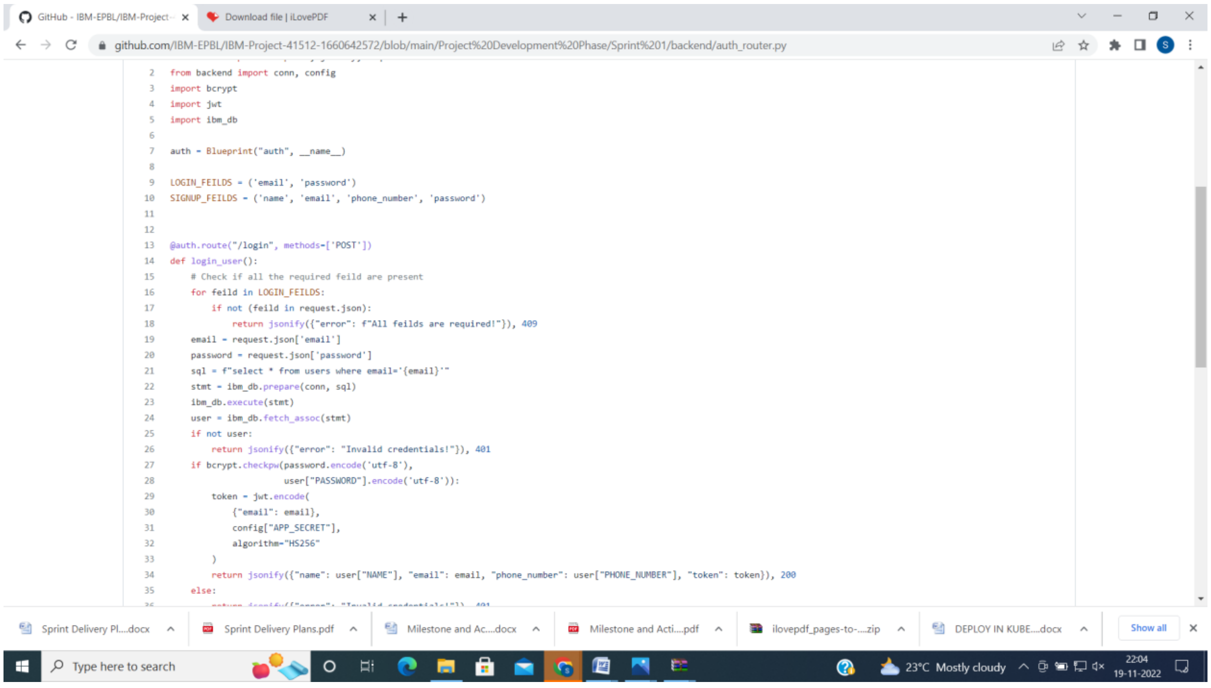
https://www.visual-paradigm.com/scrum/scrum-burndown-chart/

10 PROJECT DEVELOPMENT PHASE

10.1 SPRINT 1

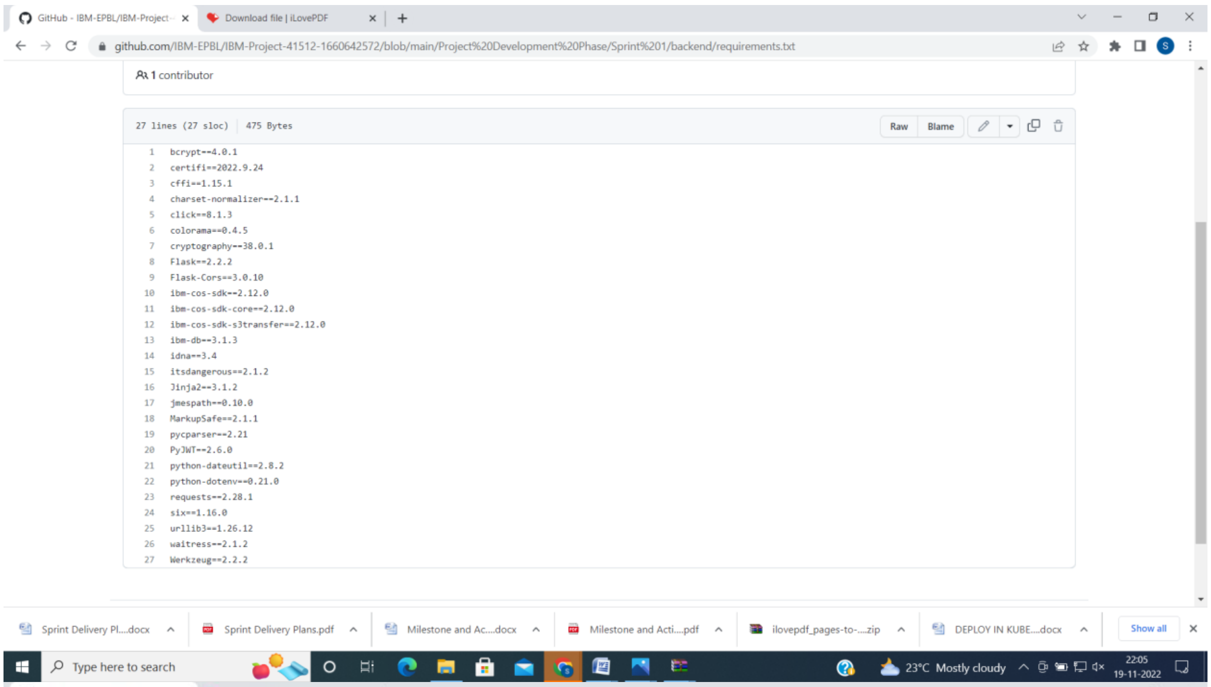

github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/backend/auth_router.py

```python
from backend import conn, config
import bcrypt
import jwt
import ibm_db

auth = Blueprint("auth", __name__)

LOGIN_FEILDS = ('email', 'password')
SIGNUP_FEILDS = ('name', 'email', 'phone_number', 'password')


@auth.route("/login", methods=['POST'])
def login_user():
    # Check if all the required feild are present
    for feild in LOGIN_FEILDS:
        if not (feild in request.json):
            return jsonify({"error": f"All feilds are required!"}), 409
    email = request.json['email']
    password = request.json['password']
    sql = f"select * from users where email='{email}'"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.execute(stmt)
    user = ibm_db.fetch_assoc(stmt)
    if not user:
        return jsonify({"error": "Invalid credentials!"}), 401
    if bcrypt.checkpw(password.encode('utf-8'),
                      user["PASSWORD"].encode('utf-8')):
        token = jwt.encode(
            {"email": email},
            config["APP_SECRET"],
            algorithm="HS256"
        )
        return jsonify({"name": user["NAME"], "email": email, "phone_number": user["PHONE_NUMBER"], "token": token}), 200
    else:
```



github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/backend/auth_router.py

```python
from backend import conn, config
import bcrypt
import jwt
import ibm_db

auth = Blueprint("auth", __name__)

LOGIN_FEILDS = ('email', 'password')
SIGNUP_FEILDS = ('name', 'email', 'phone_number', 'password')


@auth.route("/login", methods=['POST'])
def login_user():
    # Check if all the required feild are present
    for feild in LOGIN_FEILDS:
        if not (feild in request.json):
            return jsonify({"error": f"All feilds are required!"}), 409
    email = request.json['email']
    password = request.json['password']
    sql = f"select * from users where email='{email}'"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.execute(stmt)
    user = ibm_db.fetch_assoc(stmt)
    if not user:
        return jsonify({"error": "Invalid credentials!"}), 401
    if bcrypt.checkpw(password.encode('utf-8'),
                      user["PASSWORD"].encode('utf-8')):
        token = jwt.encode(
            {"email": email},
            config["APP_SECRET"],
            algorithm="HS256"
        )
        return jsonify({"name": user["NAME"], "email": email, "phone_number": user["PHONE_NUMBER"], "token": token}), 200
    else:
```

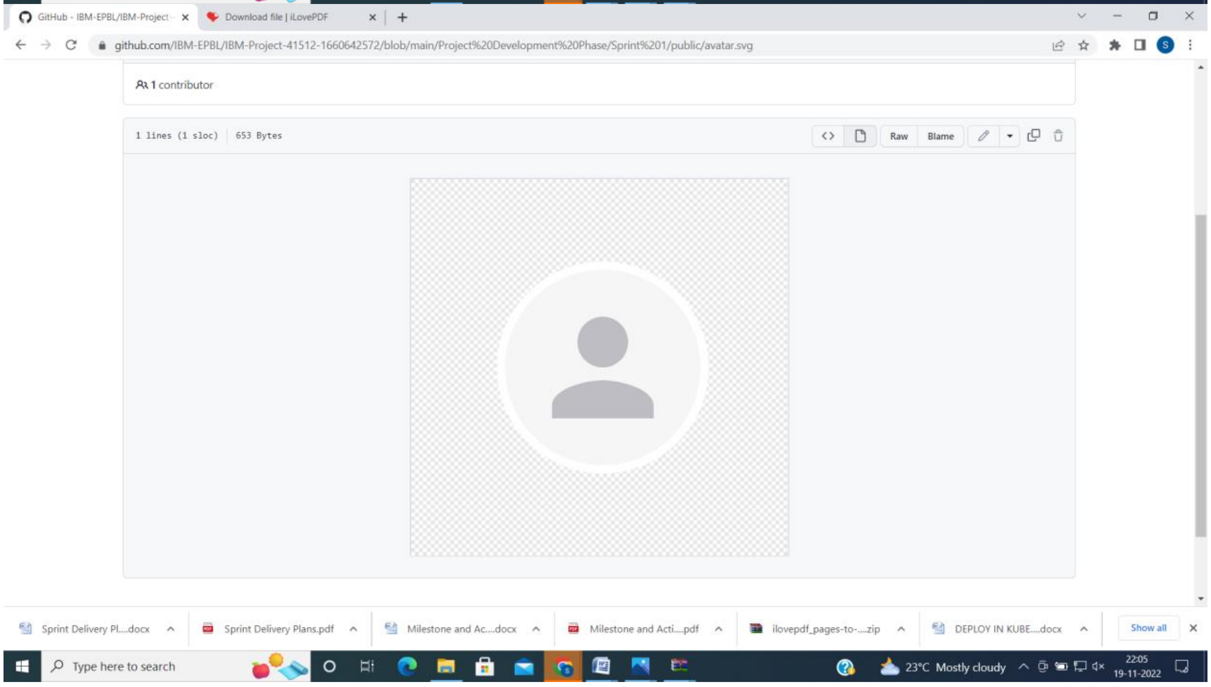

GitHub - IBM-EPBL/IBM-Project- | Download file | iLovePDF

github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/backend/requirements.txt

1 contributor

27 lines (27 sloc) | 475 Bytes

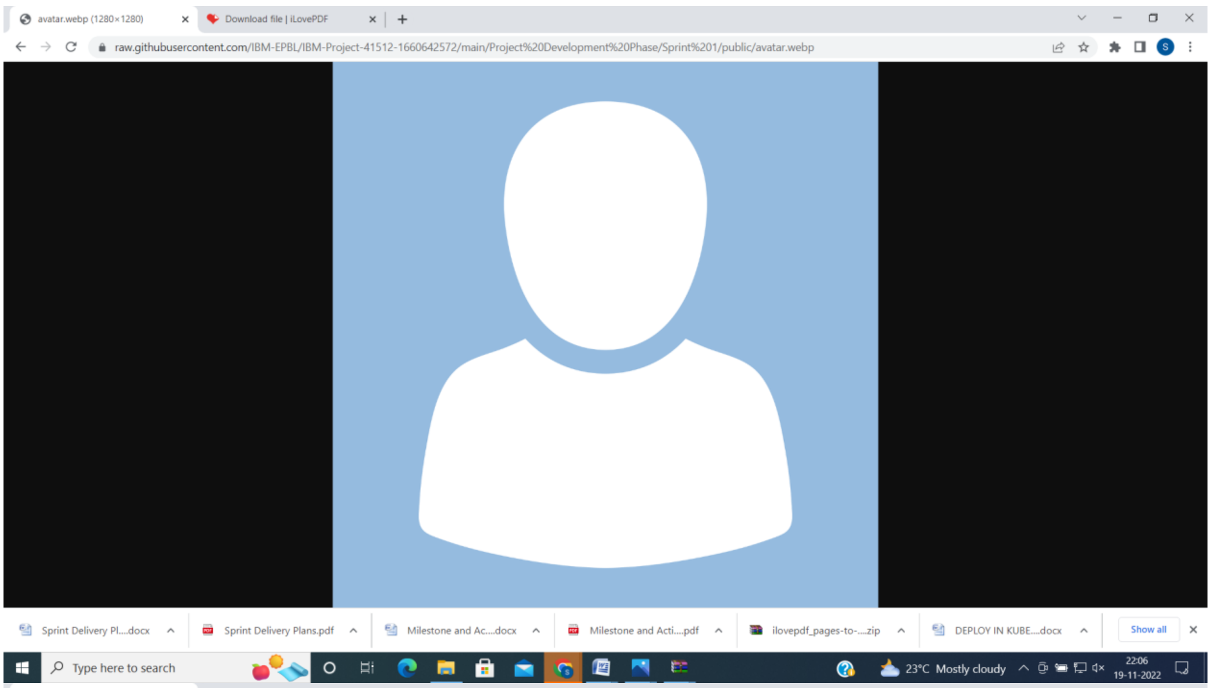
```
bcrypt==4.0.1
certifi==2022.9.24
cffi==1.15.1
charset-normalizer==2.1.1
click==8.1.3
colorama==0.4.5
cryptography==38.0.1
Flask==2.2.2
Flask-Cors==3.0.10
ibm-cos-sdk==2.12.0
ibm-cos-sdk-core==2.12.0
ibm-cos-sdk-s3transfer==2.12.0
ibm-db==3.1.3
idna==3.4
itsdangerous==2.1.2
Jinja2==3.1.2
jmespath==0.10.0
MarkupSafe==2.1.1
pycparser==2.21
PyJWT==2.6.0
python-dateutil==2.8.2
python-dotenv==0.21.0
requests==2.28.1
six==1.16.0
urllib3==1.26.12
waitress==2.1.2
Werkzeug==2.2.2
```


GitHub - IBM-EPBL/IBM-Project- | Download file | iLovePDF

github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/public/avatar.svg

1 contributor

1 lines (1 sloc) | 653 Bytes

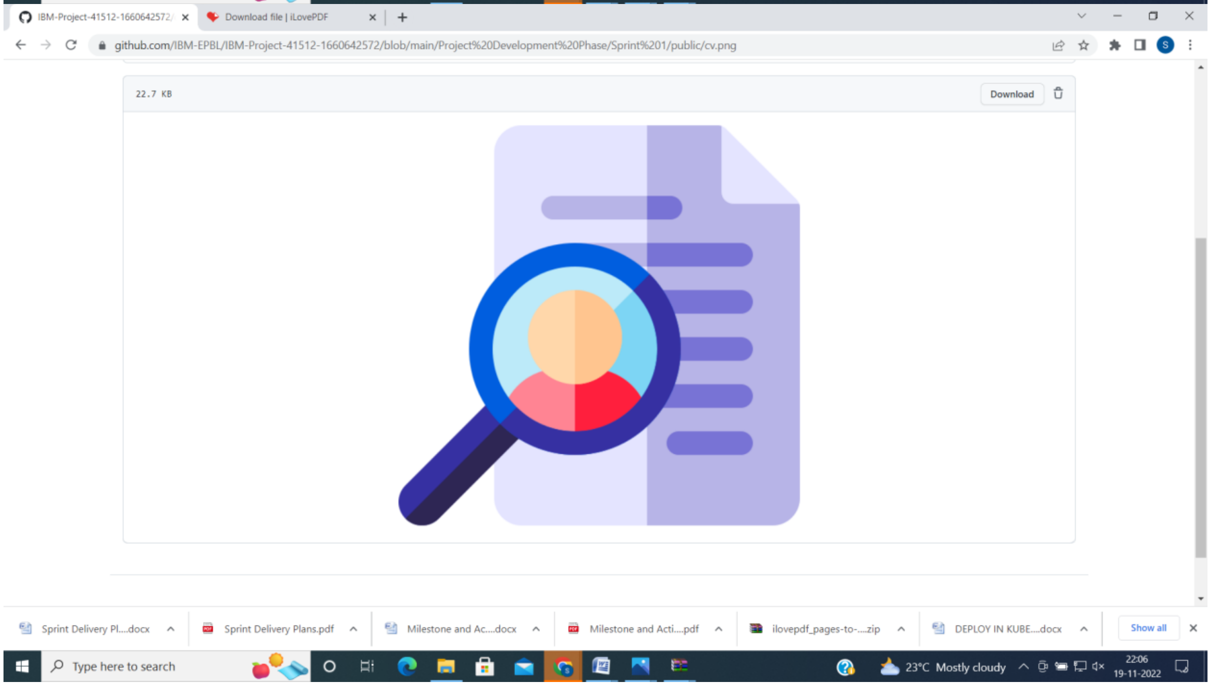

avatar.webp (1280×1280)

Download file | iLovePDF

raw.githubusercontent.com/IBM-EPBL/IBM-Project-41512-1660642572/main/Project%20Development%20Phase/Sprint%201/public/avatar.webp

Sprint Delivery Pl....docx | Sprint Delivery Plans.pdf | Milestone and Ac....docx | Milestone and Acti....pdf | ilovepdf_pages-to-....zip | DEPLOY IN KUBE....docx | Show all

Type here to search | 23°C Mostly cloudy | 22:06 19-11-2022


IBM-Project-41512-1660642572/

Download file | iLovePDF

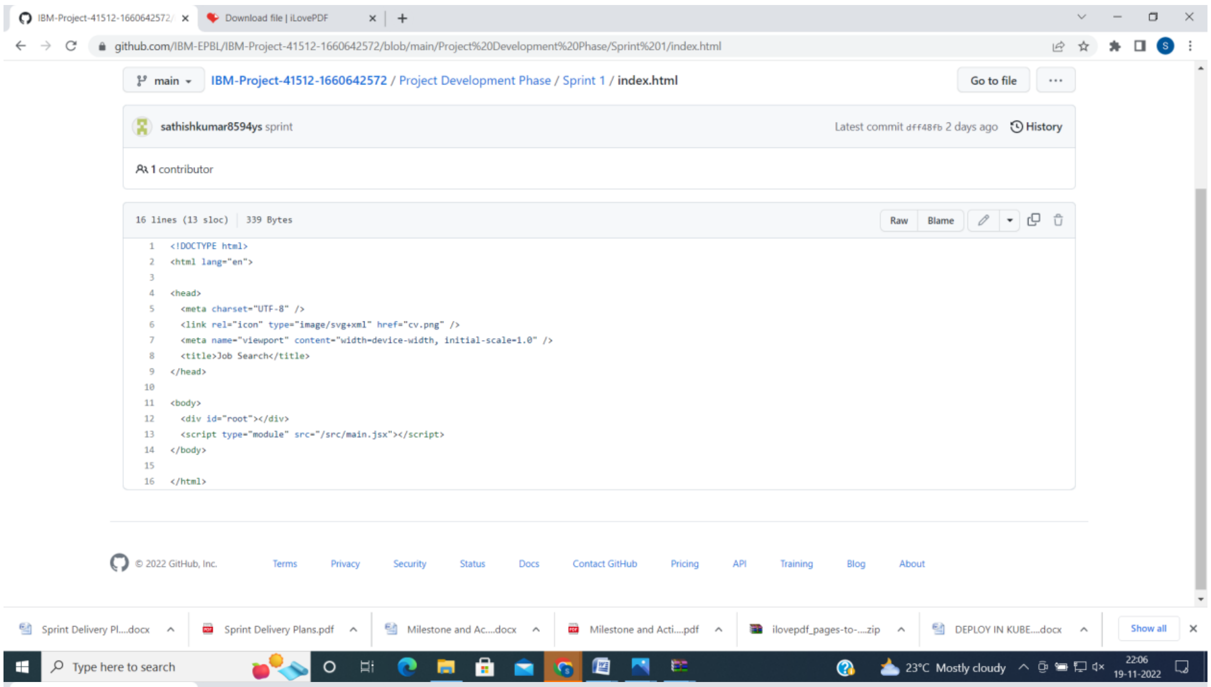
github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/public/cv.png

22.7 KB

Download

Sprint Delivery Pl....docx | Sprint Delivery Plans.pdf | Milestone and Ac....docx | Milestone and Acti....pdf | ilovepdf_pages-to-....zip | DEPLOY IN KUBE....docx | Show all

Type here to search | 23°C Mostly cloudy | 22:06 19-11-2022



github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/index.html

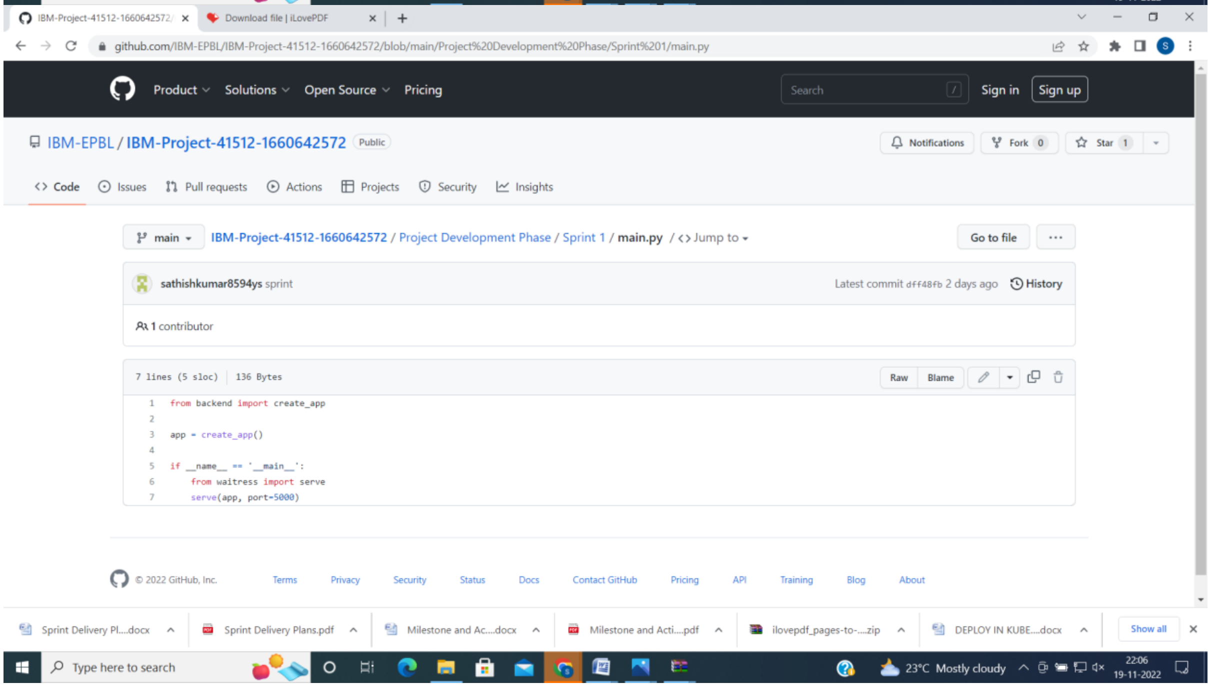
IBM-Project-41512-1660642572 / Project Development Phase / Sprint 1 / index.html

sathishkumar8594ys sprint — Latest commit dff48fb 2 days ago — History

1 contributor

16 lines (13 sloc) | 339 Bytes

```html
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <link rel="icon" type="image/svg+xml" href="cv.png" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Job Search</title>
</head>

<body>
  <div id="root"></div>
  <script type="module" src="/src/main.jsx"></script>
</body>

</html>
```



github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/main.py

IBM-EPBL / IBM-Project-41512-1660642572 Public

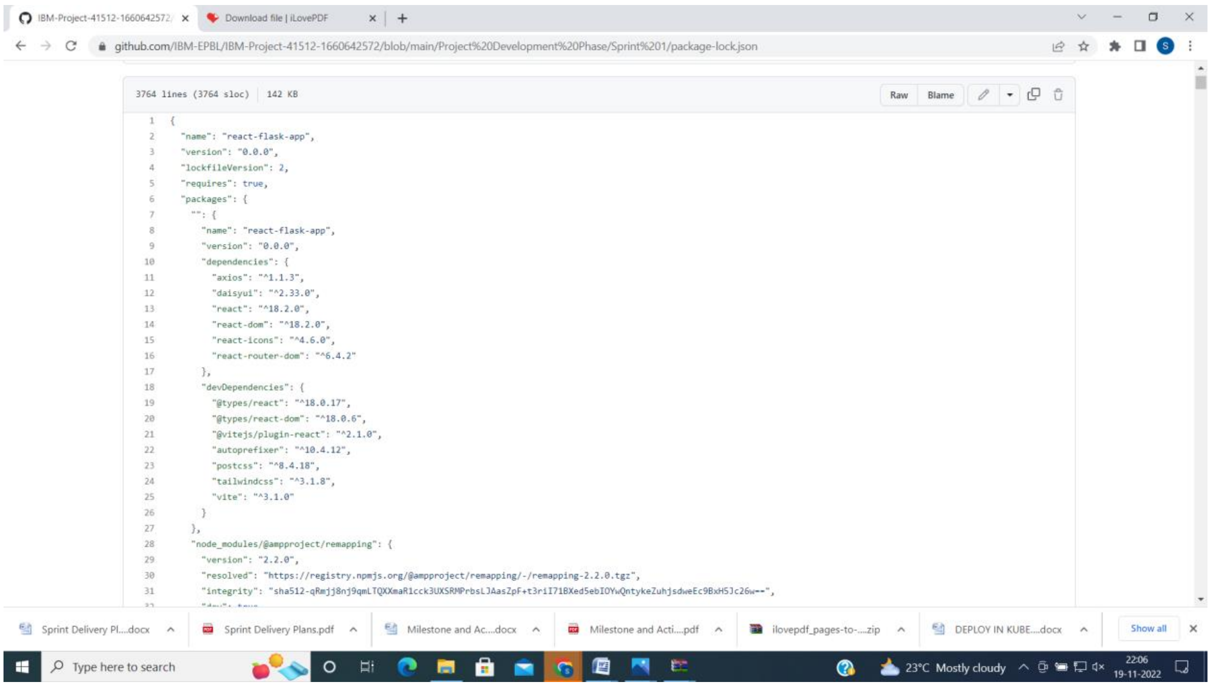
IBM-Project-41512-1660642572 / Project Development Phase / Sprint 1 / main.py

sathishkumar8594ys sprint — Latest commit dff48fb 2 days ago — History

1 contributor

7 lines (5 sloc) | 136 Bytes

```python
from backend import create_app

app = create_app()

if __name__ == '__main__':
    from waitress import serve
    serve(app, port=5000)
```

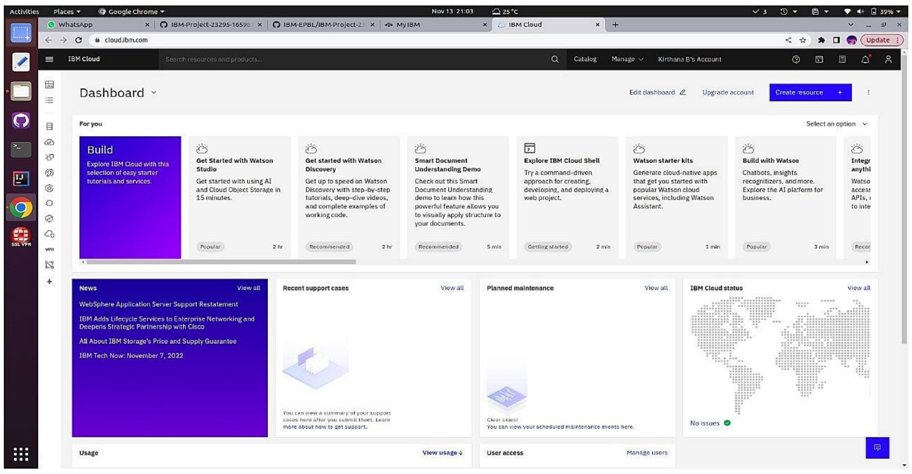


```
3764 lines (3764 sloc)  142 KB
{
  "name": "react-flask-app",
  "version": "0.0.0",
  "lockfileVersion": 2,
  "requires": true,
  "packages": {
    "": {
      "name": "react-flask-app",
      "version": "0.0.0",
      "dependencies": {
        "axios": "^1.1.3",
        "daisyui": "^2.33.0",
        "react": "^18.2.0",
        "react-dom": "^18.2.0",
        "react-icons": "^4.6.0",
        "react-router-dom": "^6.4.2"
      },
      "devDependencies": {
        "@types/react": "^18.0.17",
        "@types/react-dom": "^18.0.6",
        "@vitejs/plugin-react": "^2.1.0",
        "autoprefixer": "^10.4.12",
        "postcss": "^8.4.18",
        "tailwindcss": "^3.1.8",
        "vite": "^3.1.0"
      }
    },
    "node_modules/@ampproject/remapping": {
      "version": "2.2.0",
      "resolved": "https://registry.npmjs.org/@ampproject/remapping/-/remapping-2.2.0.tgz",
      "integrity": "sha512-qRmjj8nj9qmLTQXXmaR1cck3UXSRMPrbsLJAasZpF+t3riI71BXed5ebIOYwQntykeZuhjsdweEc9BxH5Jc26w==",
```

IBM-Project-41512-1660642572
Download file | iLovePDF
github.com/IBM-EPBL/IBM-Project-41512-1660642572/blob/main/Project%20Development%20Phase/Sprint%201/package-lock.json
Raw
Blame
Sprint Delivery Pl....docx
Sprint Delivery Plans.pdf
Milestone and Ac....docx
Milestone and Acti....pdf
ilovepdf_pages-to-....zip
DEPLOY IN KUBE....docx
Show all
Type here to search
23°C Mostly cloudy
22:06
19-11-2022

11.DEVELOPING A CHATBOT

1. **Navigateto cloud.ibm.com**



Dashboard

For you

Build
Explore IBM Cloud with this selection of easy starter tutorials and services.

Get Started with Watson Studio
Get started with using AI and Cloud Object Storage in 15 minutes.
Popular 2 hr

Get started with Watson Discovery
Get up to speed on Watson Discovery with step-by-step tutorials, deep-dive videos, and complete examples of working code.
Recommended 2 hr

Smart Document Understanding Demo
Check out this Smart Document Understanding demo to learn how this powerful feature allows you to visually apply structure to your documents.
Recommended 5 min

Explore IBM Cloud Shell
Try a command-driven approach for creating, developing, and deploying a web project.
Getting started 2 min

Watson starter kits
Generate cloud-native apps that get you started with popular Watson cloud services, including Watson Assistant.
Popular 1 min

Build with Watson
Chatbots, insights, recognitizers, and more. Explore the AI platform for business.
Popular 3 min

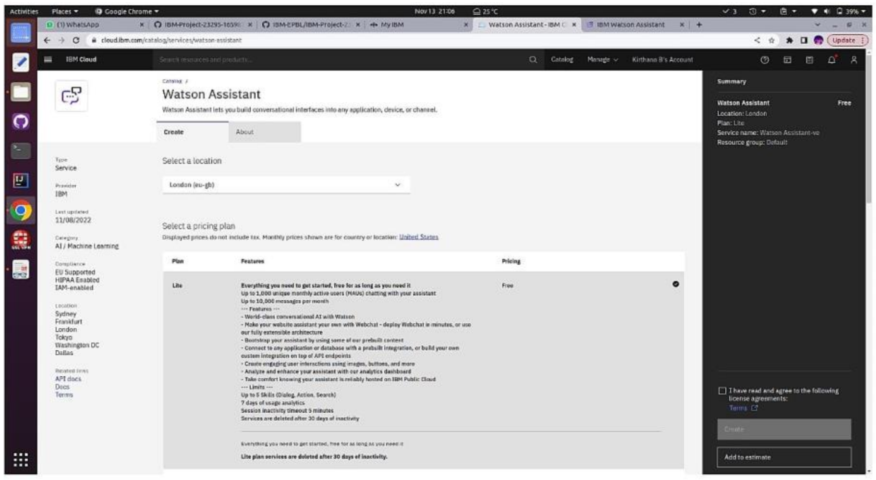
News
WebSphere Application Server Support Restatement
IBM Adds Lifecycle Services to Enterprise Networking and Deepens Strategic Partnership with Cisco
All About IBM Storage's Price and Supply Guarantee
IBM Tech Now: November 7, 2022

Recent support cases
You can view a summary of your support cases here after you submit them. Learn more about how to get support.

Planned maintenance
Clear skies!
You can view your scheduled maintenance events here.

IBM Cloud status
No issues

2. **In the catalog , search for Watson assistant service and click it**



Catalog /

Watson Assistant

Watson Assistant lets you build conversational interfaces into any application, device, or channel.

Create | About

Type: Service
Provider: IBM
Last updated: 11/08/2022
Category: AI / Machine Learning
Compliance: EU Supported, HIPAA Enabled, IAM-enabled
Location: Sydney, Frankfurt, London, Tokyo, Washington DC, Dallas
Related links: API docs, Docs, Terms

Select a location

London (eu-gb)

Select a pricing plan

Displayed prices do not include tax. Monthly prices shown are for country or location: United States

Plan	Features	Pricing
Lite	Everything you need to get started, free for as long as you need it Up to 1,000 unique monthly active users (MAUs) chatting with your assistant Up to 10,000 messages per month --- Features --- - World-class conversational AI with Watson - Make your website assistant your own with Webchat - deploy Webchat in minutes, or use our fully extensible architecture - Bootstrap your assistant by using some of our prebuilt content - Connect to any application or database with a prebuilt integration, or build your own custom integration on top of API endpoints - Create engaging user interactions using images, buttons, and more - Analyze and enhance your assistant with our analytics dashboard - Take comfort knowing your assistant is reliably hosted on IBM Public Cloud --- Limits --- Up to 5 Skills (Dialog, Action, Search) 7 days of usage analytics Session inactivity timeout 5 minutes Services are deleted after 30 days of inactivity Everything you need to get started, free for as long as you need it Lite plan services are deleted after 30 days of inactivity.	Free

Summary

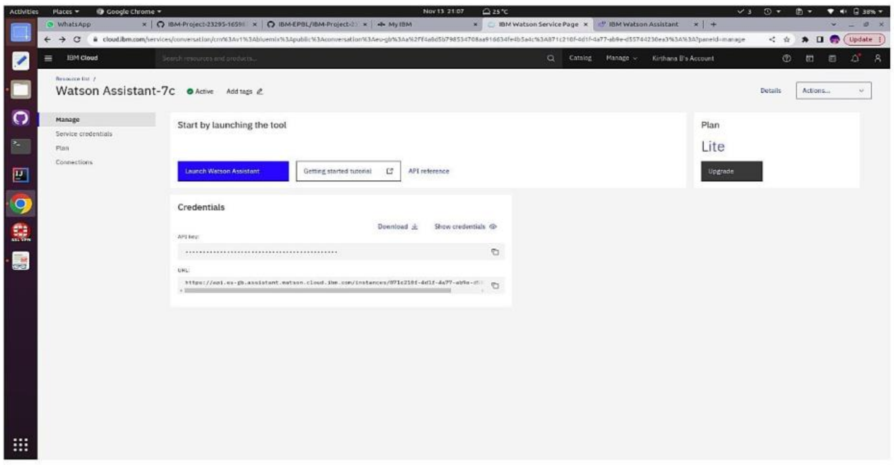
Watson Assistant — Free
Location: London
Plan: Lite
Service name: Watson Assistant-ve
Resource group: Default

I have read and agree to the following license agreements: Terms

Create

Add to estimate

3. **Create the Watson assistant by agreeing to the license agreement. After clicking create button the following page will be displayed**



Resource list /

Watson Assistant-7c — Active — Add tags

Details | Actions...

Manage
Service credentials
Plan
Connections

Start by launching the tool

Launch Watson Assistant | Getting started tutorial | API reference

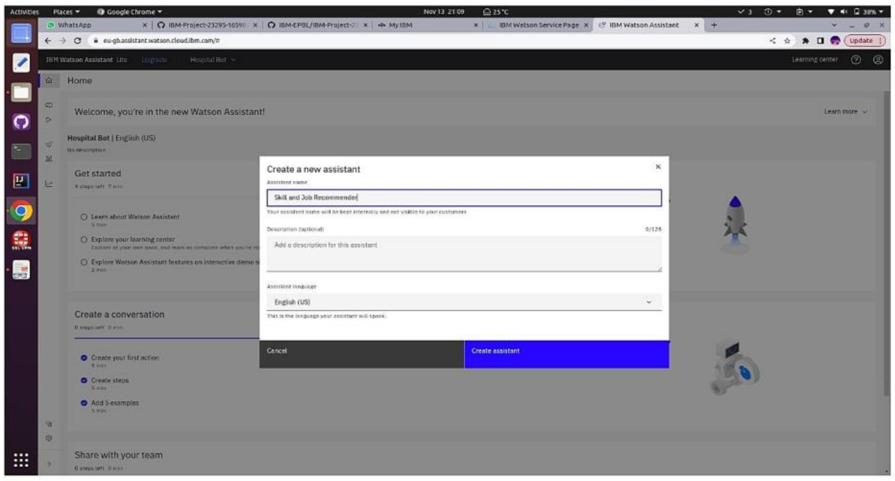
Plan: Lite — Upgrade

Credentials

Download | Show credentials

API key:
```
............................................
```
URL:
```
https://api.eu-gb.assistant.watson.cloud.ibm.com/instances/071c210f-4d1f-4a77-ab9e-d5
```

4. **Click Launch Watson Assistant , and create the Assistant by giving assistant name and click create**


Create a new assistant

Assistant name: Skill and Job Recommender

Your assistant name will be kept internally and not visible to your customers

Description (optional) 0/128

Add a description for this assistant

Assistant language: English (US)

This is the language your assistant will speak.

5. **Develop the Actions needed for our webpage. For our webpage we need to create the chatbot for skill/Job recommender Application , so we provided the steps for our application**

integrationI
D:
"5013d6c0-
675c-43bc-
8ba8-
4bde8a8bf6
14", // The
ID of
thisintegrat
ion.region: "
eu-
gb", // The
region
your integra
tion is host
ed in.
serviceInstanceID: "871c210f-
4d1f-4a77-ab9e-
d55744230ea3", // The ID of you

```
  r service instance.onLoad: functi
  on(instance) { instance.render();
   }
 };
 setTimeout(function(){
  const t=document.createElement('script');
  t.src="https://web-
chat.global.assistant.watson.appdomain.cloud/versions/"
+ (window.watsonAssistantChatOptions.clientVersion ||
'latest') +"/WatsonAssistantChatEntry.js";
  document.head.appendChild(t);
 });
</script>
```

Our Chatbot Link

https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Feu- gb.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-871c210f-4d1f-4a77-ab9e-d55744230ea3%3A%3A0aafa119-340b-4b86- ba45-ae875209ce29&integrationI

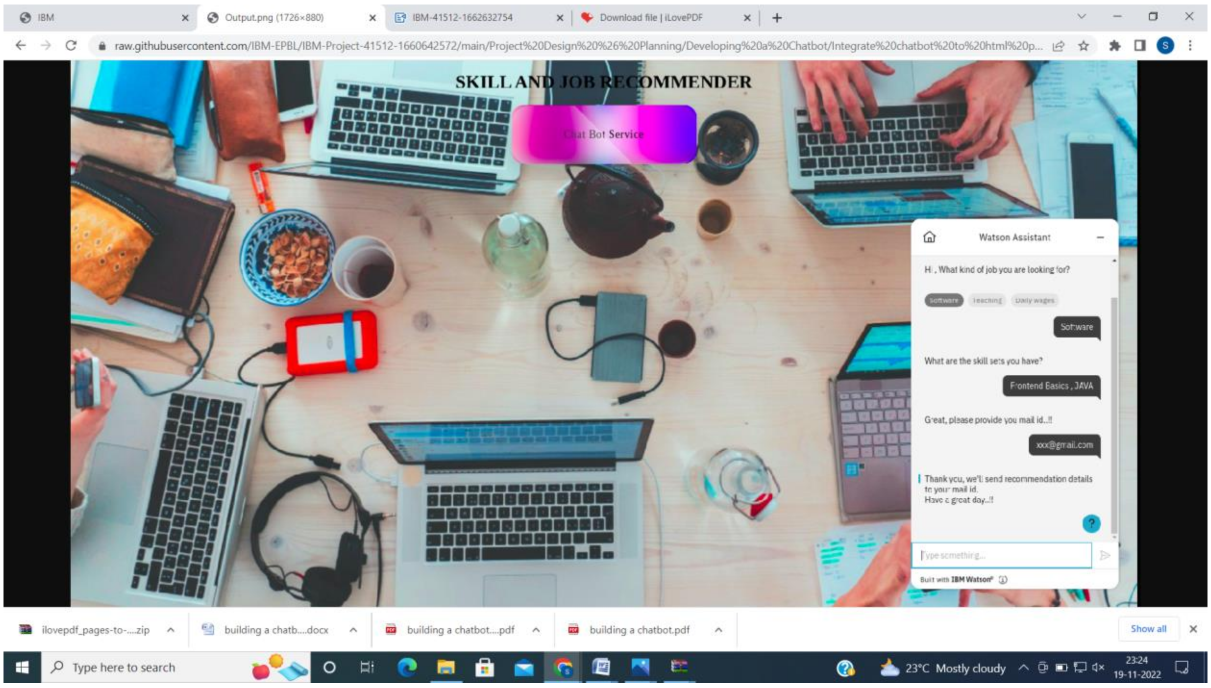
```
button{
   position: relative;
   width: 320px;
   height:55px;
   margin:3% auto;
   color: white;
   font-size: 15px;
   background: rgba(255,255,255,0.06);
   padding: 10px 30px;
```

```
  border-radius: 20px;
  border: none;
}
button a
{
  position: absolute;
  top:0;
  left:0;
  width:100%;
  height:100%;
  display: flex;
  justify-content: center;
  align-items: center;
  background:rgba(255,255,255,0.05);
  box-shadow: 0 15px 35px rgba(0,0,0,.2);
  border-top: 1px solid rgba(0, 0, 0, 0.1);
  border-bottom: 1px solid rgba(255,255,255,.1);
  border-radius: 30px;
  color: rgb(0, 0, 0);
  z-index:1;
  font-weight:400;
  letter-spacing:1px;
  text-decoration:none;
  overflow:hidden;
  transition: 0.2s;
  backdrop-filter: blur(15px);
}
button:hover a{
  letter-spacing:3px;
```

```
 }
 button a::before
 {
   content: '';
   position: absolute;
   top:0;
   left:0;
   width: 50%;
   height: 100%;
   background: linear-gradient(to
left,rgba(255,255,255,0.3),transparent);
   transform: skew(45deg) translateX(0);
   transition: 0.2s;
 }
 button:hover a::before
 {
   transform: skew(45deg) translateX(200%);
 }
 button::before
 {
   content: '';
   position: absolute;
   left:50%;
   transform: translateX(-50%);
   bottom: -3px;
   width:30px;
   height: 8px;
   background: linear-gradient(to right,rgb(92, 255, 252),rgb(29,
59, 230));
```

```
  border-radius: 10px;
  transition: 0.2s;
  transition-delay: 0;
 }

 button:hover::before
 {
  bottom: 0;
  height: 50%;
  width: 80%;
  border-radius: 30px;
  transition-delay: 0.5s;
 }
 button::after
 {
  content: '';
  position: absolute;
  left:50%;
  transform: translateX(-50%);
  top: -1px;
  width:30px;
  height: 8px;
  background: linear-gradient(to right,rgb(92, 255, 252),rgb(72,
12, 240));
  border-radius: 20px;
  transition: .5s;
  transition-delay: 0;
 }
 button:hover::after
 {
  bottom: 0;
  height: 50%;
  width: 80%;
  border-radius: 30px;
  transition-delay: 0.5s;
 }
<!DOCTYPE html>
<html>
  <head>
     <title>Skill and Job Recommender</title>
     <link rel="stylesheet" href="./Chat.css">
     <style>
        body {
             background:
url(https://miro.medium.com/max/1200/1*qjuZnAsQiGFIuE_-
X-wGPQ.jpeg) no-repeat center center fixed;
             -webkit-background-size: cover;
             -moz-background-size: cover;
             -o-background-size: cover;
             background-size: cover;
             text-align: center;
           align-content: center;
                 }
        button{
           background:linear-gradient(to
left,blue,violet,lavender,purple,pink);
           animation-name: example;

           margin: auto;
```

```
              text-align: center;
              align-content: center;
              height: 100px;
              font-size:large;
              font-weight: bolder;
              font-family: Bodoni Mt;
            }

            h1
            {
         align-content: center;
            }
        </style>
    </head>
    <body >
        <h1 style="color:rgb(0, 0, 0);">SKILL AND JOB
RECOMMENDER</h1>
        <div></div>
        <button><a href="https://web-
chat.global.assistant.watson.appdomain.cloud/preview.html?b
ackgroundImageURL=https%3A%2F%2Feu-
gb.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fu
px-871c210f-4d1f-4a77-ab9e-
d55744230ea3%3A%3A0aafa119-340b-4b86-ba45-
ae875209ce29&integrationID=5013d6c0-675c-43bc-8ba8-
4bde8a8bf614&region=eu-gb&serviceInstanceID=871c210f-
4d1f-4a77-ab9e-d55744230ea3">Chat Bot
Service</a></button>
        <!-- Embed-->
<script>
  window.watsonAssistantChatOptions = {
    integrationID: "5013d6c0-675c-43bc-8ba8-4bde8a8bf614", // The ID of this integration.
    region: "eu-gb", // The region your integration is hosted in.
    serviceInstanceID: "871c210f-4d1f-4a77-ab9e-d55744230ea3", // The ID of your service instance.
    onLoad: function(instance) { instance.render(); }
  };
  setTimeout(function(){
    const t=document.createElement('script');
    t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
    document.head.appendChild(t);
  });
</script>

    </body>
</html>
```



Hi , What kind of job you are looking for?

Software

What are the skill sets you have?

Frontend Basics , JAVA

Great, please provide you mail id..!!

xxx@gmail.com

Thank you, we'll send recommendation details to your mail id.
Have a great day..!!