SKILL / JOB RECOMMENDER APPLICATION

TEAM ID : PNT2022TMID08534

TEAM MEMBERS

MAHENDIRAN R (810419205025)

ELAVARASAN V (810419205016)

PRAKASH P (810419205030)

SEENIVASAN G (810419205041)

VIKRAM D (810419205053)

TEAM MENTOR:

CONTENTS

SKILLS/ JOB RECOMMENDER APPLICATION

ABSTRACT

Most businesses now use Internet-based recruiting portals as their main hiring method. Such platforms save the time and expense associated with hiring new employees, but they have problems with outdated information retrieval strategies based search approaches. As a result, a sizable number of applicants passed up the chance to be hired. The recommender system technology is used successfully in e-commerce applications to address issues linked to information overload. It helps customers identify products that fit their personal preferences. Numerous recommender system ideas have been put out in an effort to enhance the functionality of e-recruiting. This proejct will provide a survey of the electronic hiring process and current suggestion building methodologies for matching individuals to job. The user and their information are stored in the Database. An alert is sent when there is an opening based on the user skill set. Users will interact with the Chabot 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. INTRODUCTION

1.1 PROJECT OVERVIEW

Finding jobs that best suits the interests and skill set is quite a challenging task for the job seekers. The difficulties arise from not having proper knowledge on the organization's objective, their work culture and current job openings. In addition, finding the right candidate with desired qualifications to fill their current job openings is an important task for the recruiters of any organization. Job recommendation system has certainly made job seeking convenient to job seekers. This is the solution where recruiter as well as the job seeker meets aiming at fulfilling their individual requirement. To develop end-to-end web users are the cheapest as well as the fastest source of communication reaching wide range of audience on just a single click irrespective of their geographical distance. As simplified recruitment process which makes way convenience for job seekers to access job portal on the go, as our app script is built to support numerous business models as per the industry requirements. In this decade people are using their smart phone rather than web portals for job seeking online. So it is the right way for career portal which balances the gap between recruitment board and job searcher candidates. On the job seeker panel, the user can register their profile using system. Here, a job seeker searches for the positions that interest him and submits an application. Due to the abundance of job boards, candidates typically choose the one that offers the best services to them, including skills, creating a job profile, and suggesting new positions to job seekers.

1.2 PURPOSE

Job recommendation is primarily aimed at supporting the discovery of jobs that may interest the user. It should be dynamic in order to cater to the changing preferences of the user. The proposed system will help the user to overcome these difficulties by matching their skills and other details with appropriate companies suitable for respective user. The proposed system consists of user dataset with various attributes and company dataset with company details.

2. LITERATURE REVIEW

2.1 EXISTING PROBLEM

2.1.1 TITLE: Implementation of an Intelligent Online Job Portal Using Machine Learning Algorithms

AUTHOR: F. M. Javed Mehedi Shamrat

Business intelligence and analytics are data management techniques used in organisations to gather historical and current data utilising software and statistics. To provide insights for enhanced decision-making by analysing unprocessed data. In the current financial environment, it is necessary to be analytical and seek for the simplest method or intelligent business model in order to survive and develop one's own firm. The main goal is to assess how well different machine learning algorithms work with the system of an online job portal. This proposed module includes three phases, including the Clusters similar kind of job search phase (CSK), which creates a visual graph displaying clusters of similar types of jobs that job seekers have searched for on the website of the job portal, the email notifications send phase (ENS), which is in charge of sending email notifications to job seekers when a job circular is posted on the website, and the extract the job circular phase (EJC), which is the method for finding relevant job postings. The outcome demonstrates the effective grouping of related job searches, sending of email notifications to particular individuals, and information extraction from the web. The primary goal of the essay is to group together verses from the Holy Qur'an. The K-means technique was used by the authors of the research to mine the text from the Holy Qur'an and count the number of steam- and unsteam-steamed words in each cluster. The final illustration displays the various densities within each cluster. The authors suggested text document vectorization. After that, in order to provide the optimum results, the initial seed points should be chosen as widely apart as feasible.

2.1.2 **TITLE:** Designing And Implementation Of A Graduate Job Portal System

AUTHOR: Zamiwe Tembo

Better digital abilities may be crucial for successful job searching as hiring procedures increasingly move online. Digital inequality, on the other hand, raises concerns about who is most likely to be able to seek for jobs online, especially on social media, given that it implies that people utilise the Internet in different ways and to varying degrees of skill. This essay investigates online job searching, covering the function of digital job-search abilities. The findings indicate that online job-seeking activities are influenced by sociodemographic traits (such as age, race, education, and income) as well as online experiences, using social media, and having better digital job-search skills. These results demonstrate the existence of digital disparities in online job searching, including variations due to social media usage. Additionally, the majority of these research INFORMATION, COMMUNICATION & SOCIETY 1827 did not take into account the importance of digital job-search abilities. However, Puckett and Hargittai (2012) showed that people with higher-level Internet abilities were more likely to use the Internet for finding information about jobs, indicating that this is a domain worth additional investigation. The study focused on the job-searching experiences of college students. They lacked assessments that were primarily focused on job-search abilities, a gap that this paper fills. The literature on the possible use of social media for job searching and the importance of digital job-search abilities is reviewed in the sections that follow.

2.1.3 **TITLE:** Recruitment And Selection Process With Reference Using Job Portal Framework

AUTHOR: Ankit Bhatnagar1 ,Nitish Kajla2 , Mahesh Kumar Gupta3

The project's goal is to create an online search gateway for the college's or company's placement department. With the proper login information, this system, which is a web application, may be accessed both inside and outside the company. To manage the scholar data related to placement, the placement department frequently uses these technologies as a web employment site. Students submit their information as a resume. The application manages numerous modules and the reports that go along with them. This means that a poor distribution of or lack of information about employment chances prevents people from learning about new job prospects, which is one of the causes of the lack of jobs. This indicates that although there are additional positions available, job searchers are not aware of them. Here, our website aids job searchers in their search for employment. The Internet has altered many parts of our lives today, including how we hunt for jobs. The goal of creating this website was to save both the candidate and the employer time. We offer two alternatives on our website: the first is to search for employment, and the second is to search for employees. Assume that if a person is seeking for work, they must select the option to search for positions, provide their contact information, and upload their resume[2]. On the other hand, if a business is seeking for an employee, it should select the option to search for employees and fill out the necessary information.

2.1.4 **TITLE:** Shared Values of E-Recruitment Portal: Determinant Factors of Job-Seekers' Intention to use Job Portals

AUTHOR: Aradhana Patra,Munjarin Rahman.

In Malaysia, job portals are a popular tool for locating employment. in Malaysia, a job. Due to technology advancements, traditional job searching is no longer used. advancements. Consequently, this study has been carried out to concentrate on the need and behaviour various ages, genders, educational backgrounds, and jobseekers from Malaysia through a survey questionnaire on the internet. an analytical or quantitative strategy was adopted. 104 job searchers took part in the poll, shared their opinions, and comments regarding their use of Malaysian online job hunting resources. When there a few open-ended inquiries for a more thorough and thorough research investigation. this academic undertaking addressed five important factors: Usability, User Experience, Performance Expectancy, and Performance Quality. Subjective Norm and Credibility as Important Influences on Behavioural Intention to utilise a job site jobseekers. This study immediately advances the Job Portals' effectiveness and user-friendliness. With lower job search expenses and a more secure network, job hopefuls now have more opportunities to explore employment. By accelerating two-way interaction, it makes it easier for job applicants to comprehend the hiring process and gives more information about the company. E-recruitment, a subset of e-HRM, is a company's e-business initiative that uses web-based electronic technology to carry out human resources operations and procedures. According to this report, Job Street is the most popular job board in Malaysia, followed by LinkedIn. Both of these job sites place a strong emphasis on the needs of their customers and strive to effectively address the majority of the problems encountered by job seekers.

2.1.5 **TITLE:** Inequality in online job searching in the age of social media

AUTHOR: Gökçe Karaoglu, Eszter Hargittai & Minh Hao Nguyen

Better digital abilities may be crucial for successful job searching as hiring procedures increasingly move online. Digital inequality, on the other hand, raises concerns about who is most likely to be able to seek for jobs online, especially on social media, given that it implies that people utilise the Internet in different ways and to varying degrees of skill. This essay investigates online job searching, covering the function of digital job-search abilities. The findings indicate that online job-seeking activities are influenced by sociodemographic traits (such as age, race, education, and income) as well as online experiences, using social media, and having better digital job-search skills. These results demonstrate the existence of digital disparities in online job searching, including variations due to social media usage. Additionally, the majority of these research INFORMATION, COMMUNICATION & SOCIETY 1827 did not take into account the importance of digital job-search abilities. However, Puckett and Hargittai (2012) showed that people with higher-level Internet abilities were more likely to use the Internet for finding information about jobs, indicating that this is a domain worth additional investigation. The study focused on the job-searching experiences of college students. They lacked assessments that were primarily focused on job-search abilities, a gap that this paper fills. The literature on the possible use of social media for job searching and the importance of digital job-search abilities is reviewed in the sections that follow.

2.2 REFERENCES

1. F. M. Javed Mehedi Shamrat, Implementation of an Intelligent Online Job Portal Using Machine Learning Algorithms, 2020.
2. Zamiwe Tembo, Designing And Implementation Of A Graduate Job Portal System, 2019.
3. Ankit Bhatnagar1 ,Nitish Kajla2 , Mahesh Kumar Gupta3,Recruitment And Selection Process With Reference Using Job Portal Framework, 2021.
4. Aradhana Patra,Munjarin Rahman,Shared Values of E-Recruitment Portal: Determinant Factors of Job-Seekers' Intention to use Job Portals, 2020.
5. Gökçe Karaoglu, Eszter Hargittai & Minh Hao Nguyen,Inequality in online job searching in the age of social media, 2021.

2.3 PROBLEM STATEMENT DEFINITION

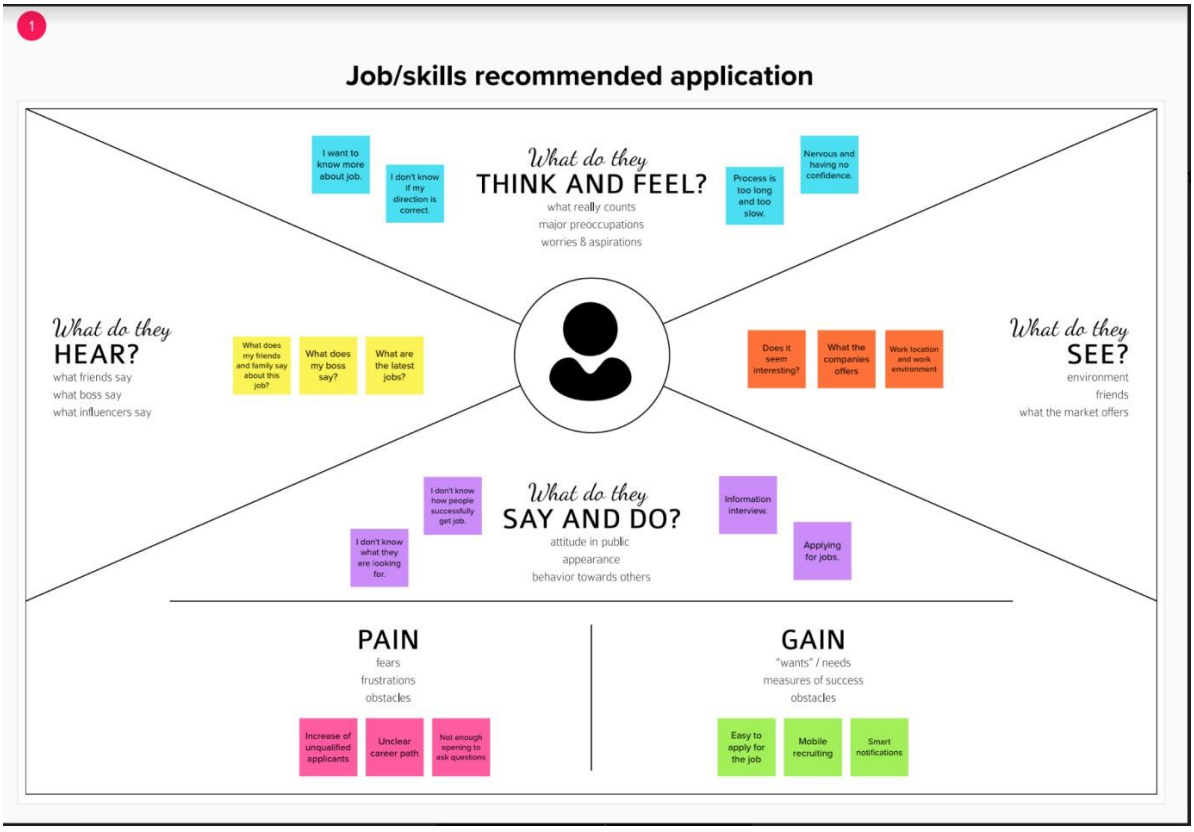
The existing system is handled manually. The system follows large number of paper work for maintaining job details and user can be difficult to search the part time jobs in manual process. In current system the student or user don't know about part time jobs details or company/office details and location. In this existing system takes lots of time for searching particular jobs information.

3. IDEATION & PROPOSED SOLUTION

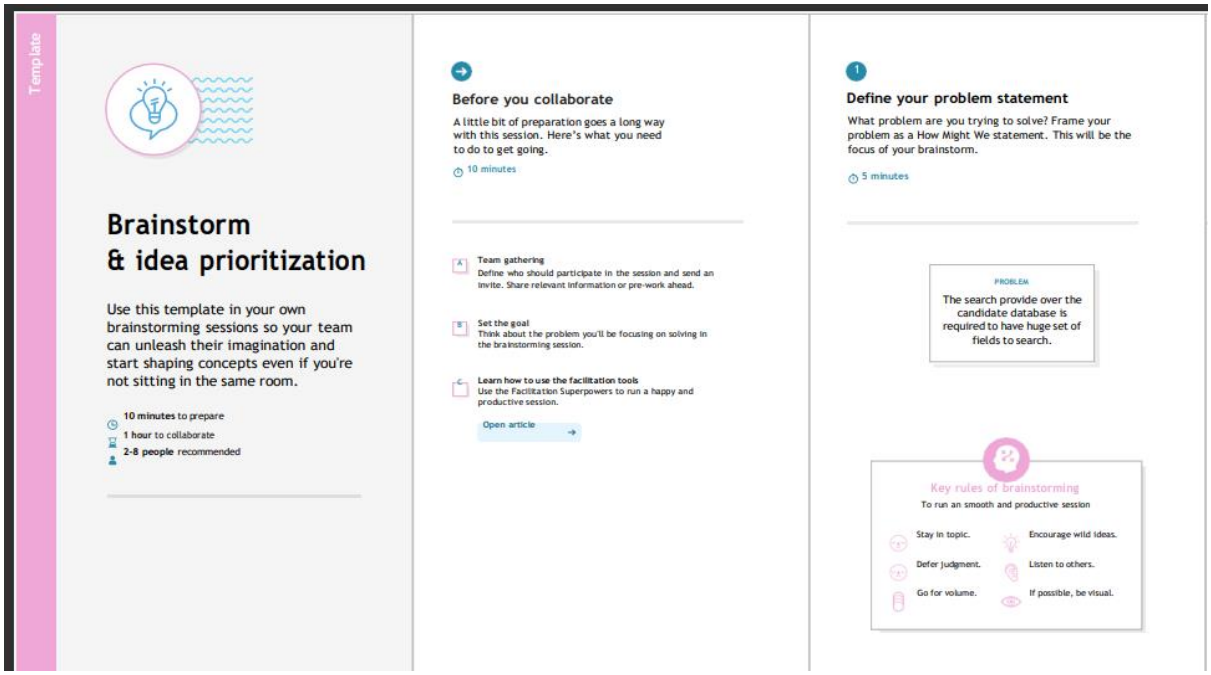
3.1 EMPATHY MAP CANVAS

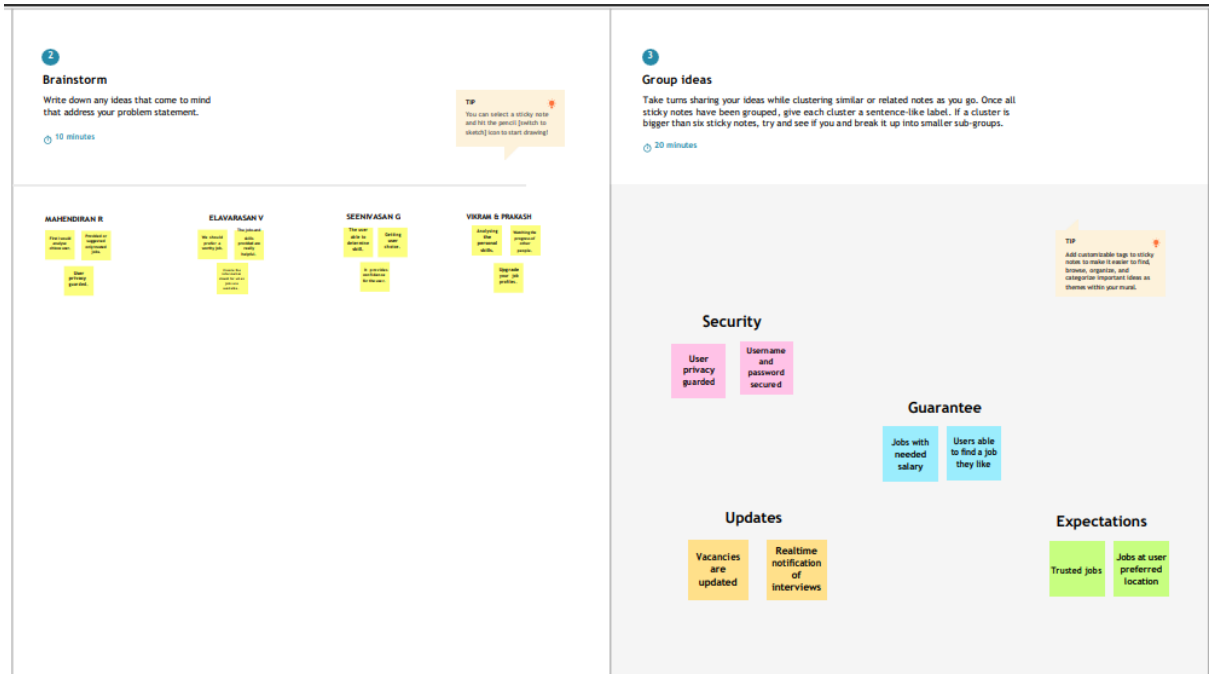
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users.Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



3.2 IDEATION & BRAINSTORMING

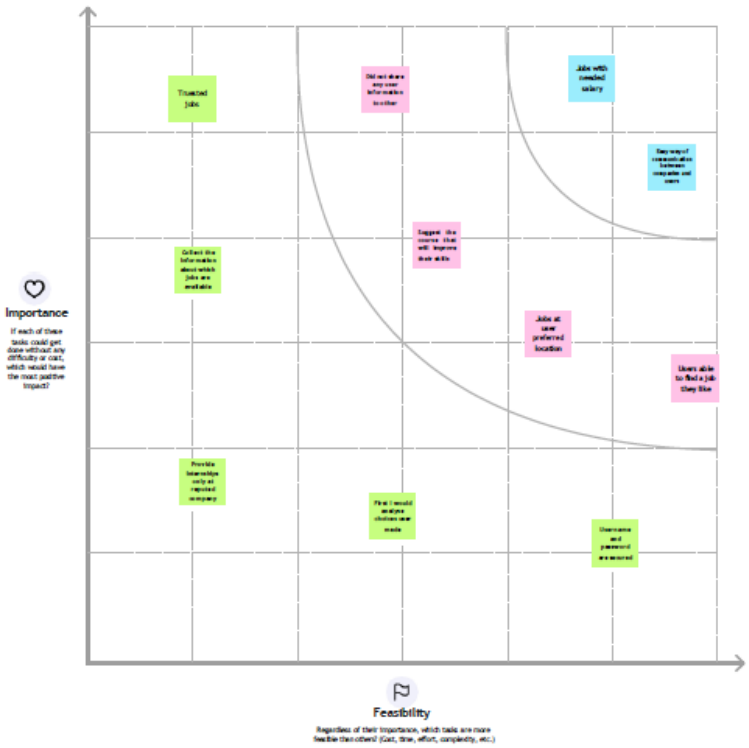




Prioritize

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

20 minutes



After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

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

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

Keep moving forward

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

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

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

Share template feedback

3.3 PROPOSED SOLUTION

The proposed system is developed after a detailed study about the requirements requested by the user. Proposed system is a computerized one, where all the limitations of manual system are compensated. jobs details of web application for skill based Job application system have simplified the working information and make a user friendly environment, where the user is provided with much flexibility to manage effectively. It helps the admin to generate desirable interface more quickly and also to produce better results.

3.4 PROBLEM SOLUTION FIT

Dealing with the enormous amount of recruiting information on the Internet, a job seeker always spends hours to find useful ones. Many times, people who lack industry knowledge are unclear about what exactly they need to learn in order to get a suitable job for them. We address the problem of recommending suitable jobs to people who are seeking a new job.

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

- Create interface

  This module offered a framework for job platform application to the user, to get answers without any human assistance. Admin can train keywords with answers for future processing. Chatbots are such kind of computer programs that interact with users using natural languages.

- Registration

  There is registration form available where new user can create their account by providing required information to the system. The registration form details are like name, email, gender, mobile number, address, and etc. These details are stored in the database. And then can getting to the username and password in the system. After the login process, the user can login in the system using username and password.

- Update job details

  The company can register to this application, the registered details like company name, id, email address; mobile number etc. after the registration process, the company can update the job details.

- Update skills

  The user can upload the skill details to this application. And the user will interact with the Chabot and can get the recommendations based on their skills.

- Recommend job with alert

  After updating the skills details, the system will recommend the job openings based on the user skills.

- Apply job

  After get the job alert, the user can apply the job through this application.

4.2 NON FUNCTIONAL REQUIREMENTS

Usability

The system shall allow the users to access the system with pc using web application. The system uses a web application as an interface. The system is user friendly which makes the system easy

Availability

The system is available 100% for the user and is used 24 hrs a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

Scalability

Scalability is the measure of a system's ability to increase or decrease in performance and cost in response to changes in application and system processing demands.

Security

A security requirement is a statement of needed security functionality that ensures one of many different security properties of software is being satisfied.

Performance

The information is refreshed depending upon whether some updates have occurred or not in the application. The system shall respond to the member in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when doing large processing jobs. Responses to view information shall take no longer than 5 seconds to appear on the screen.

Reliability

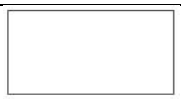
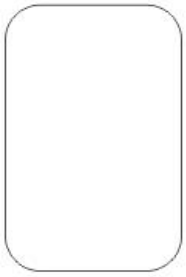
The system has to be 100% reliable due to the importance of data and the damages that can be caused by incorrect or incomplete data. The system will run 7 days a week. 24 hours a day.

5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

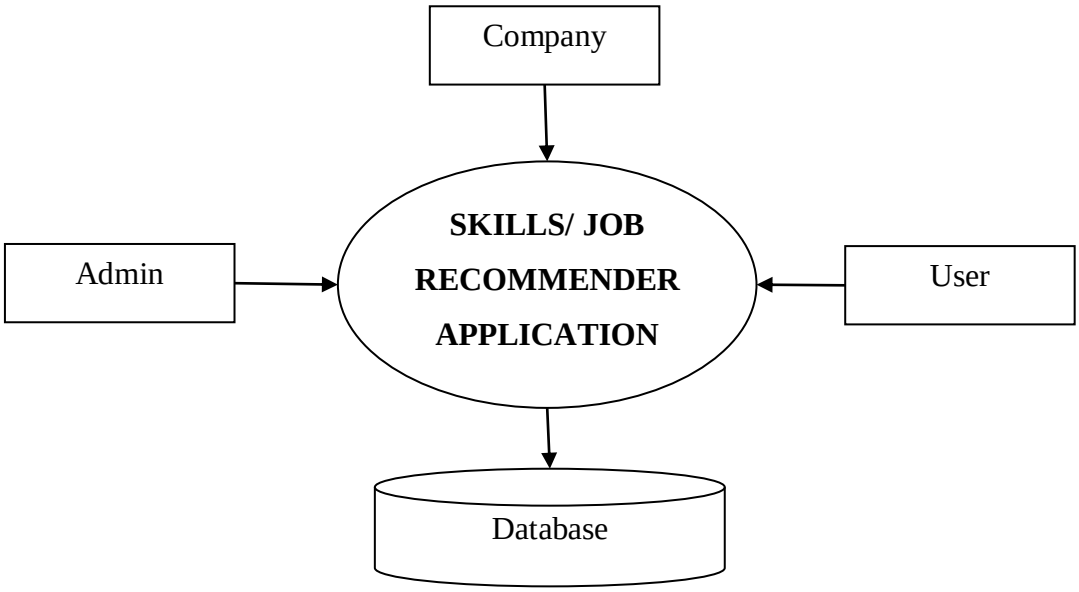
A two-dimensional diagram explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output. Individuals seeking to draft a data flow diagram must identify external inputs and outputs, determine how the inputs and outputs relate to each other, and explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects.

Data flow Symbols:

Symbol	Description
	An **entity**. A source of data or a destination for data.
	A **process** or task that is performed by the system.
	A **data store**, a place where data is held between processes.
	A **data flow**.

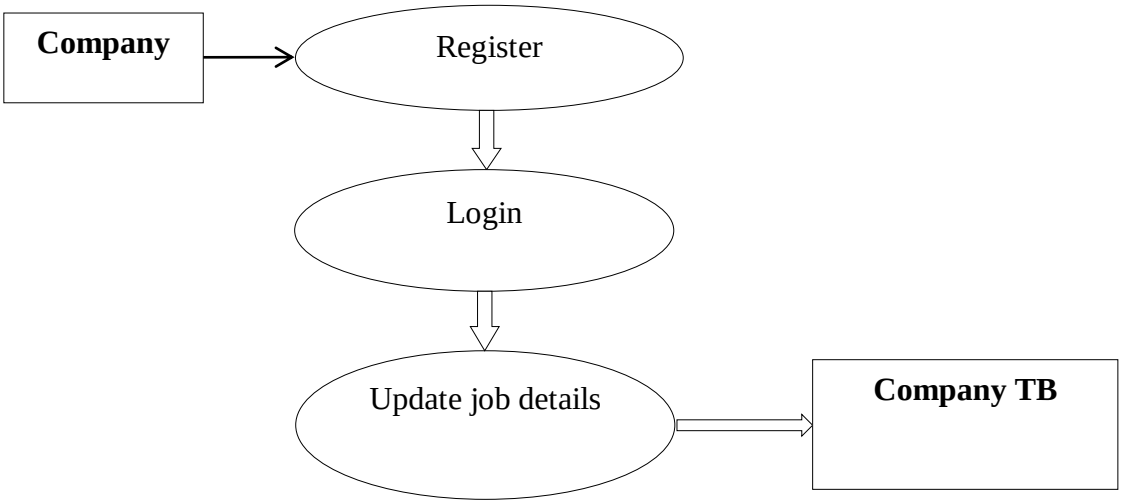
LEVEL 0

The Level 0 DFD shows how the system is divided into 'sub-systems' (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job, and shows the flow of data between the various parts of the system.



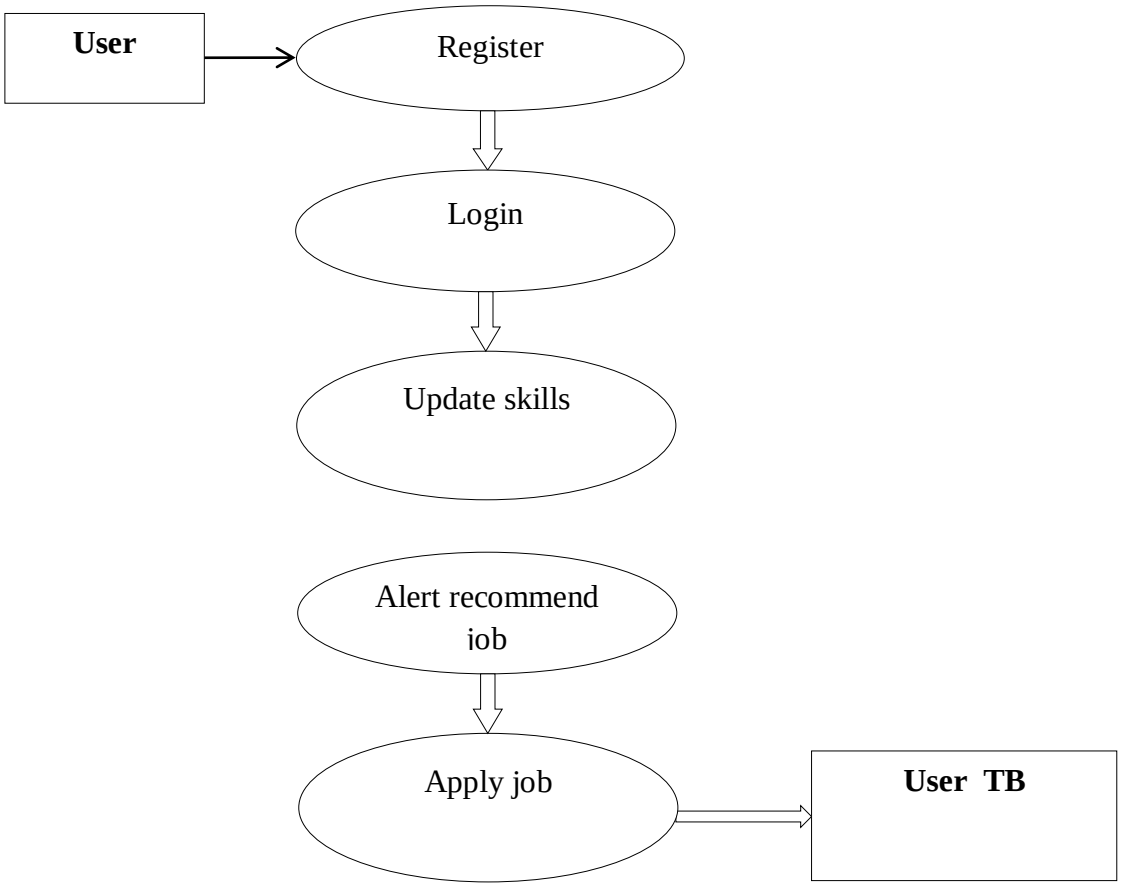
LEVEL 1

The next stage is to create the Level 1 Data Flow Diagram. This highlights the main functions carried out by the system. As a rule, to describe the system was using between two and seven functions - two being a simple system and seven being a complicated system. This enables us to keep the model manageable on screen or paper.

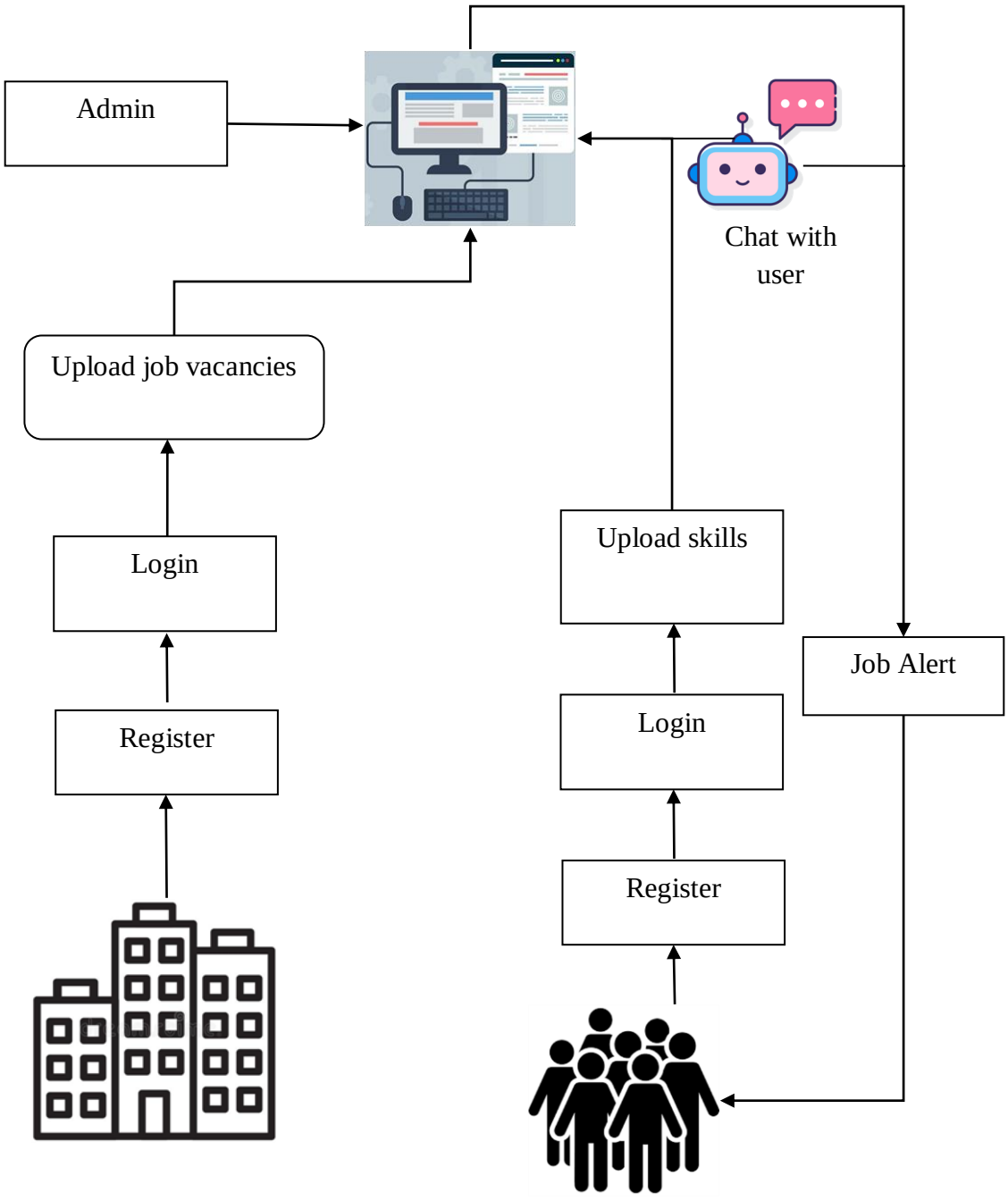


LEVEL 2

A Data Flow Diagram (DFD) tracks processes and their data paths within the business or system boundary under investigation. A DFD defines each domain boundary and illustrates the logical movement and transformation of data within the defined boundary. The diagram shows 'what' input data enters the domain, 'what' logical processes the domain applies to that data, and 'what' output data leaves the domain. Essentially, a DFD is a tool for process modelling and one of the oldest.



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES

User type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Releas
Customer (mobile user)	Registration	USN - 1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint -1
		USN – 2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint -1
		USN -3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint – 2
		USN - 4	As a user, I can register for the application through Gmail	I can receive confirmation email & click confirm	Medium	Sprint -1
	Login	USN - 5	As a user, I can log into the application by entering email & password	I can access my account / dashboard	High	Sprint -1
	Dashboard	USN - 6	Create a model set that contains those models, then assign it to a role.	Assign that group to the appropriate roles on the Roles page	High	Sprint -1
Customer (web user)	Identity-Aware	USN - 7	Open, public access, User-authenticated access,	Company public	High	Sprint -1

			Employee-restricted access.	website. App running on the company intranet. App with access to customer private information.		
Customer care executive	Communication	USN - 8	A customer care executive is a professional responsible for communicating the how's and why's regarding service expectations within a company.	For how to tackle customer queries.	Medium	Sprint -1
administrator	Device management	USN - 9	You can Delete/Disable/Enable devices in Azure Active Directory but you cannot Add/Remove Users in the directory.	Ease of use.	Medium	Sprint -1

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional requirements (epic)	User story number	User story / task	priority	Acceptance criteria	Team members
Sprint - 1	UI Desin	USN -1	As a user, I can see and experience an awesome user interface in the website	Medium	Better Impression about a website	Mahendiran R
Sprint – 1	Registration	USN -2	As a user, I can register for the application by entering my email, password, and confirming my password.	High	I can access my account I dashboard	Mahendiran R
Sprint – 1		USN -3	As a user, I will receive confirmation email once I have registered for the application	High	I can receive confirmation email & click confirm	Mahendiran R
Sprint – 1		USN -4	As a user, I can register for the application through Facebook	Low	I can register & access the dashboard with Facebook Login	Mahendiran R
Sprint – 1		USN -5	As a user, I can register for the application through Gmail	Medium	I can receive confirmation email & click confirm	Mahendiran R

Sprint – 1	Login	USN -6	As a user, I can log into the application by entering email & password	High	I can access my account I dashboard	Mahendiran R
Sprint – 1	Flask	USN -7	As a user, I can access the website in a second	High	I can access my account I dashboard	Mahendiran R
Sprint - 1	Dashboard	USN - 8	As a user, If I Logged in correctly, I can view my dashboard and I can navigate to any pages which are already listed there.	High	I can access all the pages/ dashboard	Mahendiran R
			SUBMISSION OF SPRINT – 1			
Sprint - 2	User profile	USN – 9	As a user, I can view and update my details	Medium	I can modify my details/data	Elavarasan V
Sprint - 2	Database	USN – 10	As a user, I can store my details and data in the website w	Medium	I can store my data	Elavarasan V
Sprint - 2	Cloud storage	USN – 11	As a user, I can upload my photo, resume and much more in the website.	Medium	I can Upload my documents and details	Elavarasan V
Sprint - 2	Chatbot	USN – 12	As a user, I can ask the Chatbot about latest job openings, which will help me and show the recent job openings based on my profile	High	I can know the recent job openings	Elavarasan V

Sprint - 2	Identity - aware	USN - 13	As a User, I can access my account by entering by correct login credentials. My user credentials is only displayed to me.	high	I can have my account safely	Elavarasan V
			SUBMISSION OF SPRINT-2			
Sprint - 3	Sendgrid service	USN – 14	As a user, I can get a notification or mail about a job opening with the help of sendgrid service.	Medium	I can get a notification in a second.	Prakash P
Sprint - 3	Learning resource	USN – 15	As a user, I can learn the course and I will attain the skills which will be useful for developing my technical skills.	High	I can gain the knowledge and skills	Prakash P
Sprint - 3	Docker	USN -16	As a user, I can access the website in any device	High	I can access my account in any device	Prakash P
Sprint - 3	Kubernets	USN -17	As a user, I can access the website in any device	High	I can access my account in any device	Prakash P
Sprint - 3	Deployment in cloud	USN -18	As a user, I can access the website in any device	High	I can access my account in any device	Prakash P
Sprint - 3	Technical support	USN -19	As a user, I can get a customer care support from the website which will	Medium	I can tackle my problem & queries.	Prakash P

			solve my queries.			
			SUBMISSION OF SPRINT - 3			
Sprint - 4	Unit testing	USN – 20	As a user, I can access the website without any interruption	Hgh	I can access the website without any interruption	Seenivasan G Vikram D
Sprint - 4	Integration testing	USN – 21	As a user, I can access the website without any interruption	Hgh	I can access the website without any interruption	Seenivasan G Vikram D
Sprint - 4	System testing	USN – 22	As a user, I can access the website without any interruption	Hgh	I can access the website without any interruption	Seenivasan G Vikram D
Sprint - 4	correction	USN – 23	As a user, I can access the website without any interruption	Hgh	I can access the website without any interruption	Seenivasan G Vikram D
Sprint - 4	Acceptance testing	USN - 24	As a user, I can access the website without any interruption	Hgh	I can access the website without any interruption	Seenivasan G Vikram D
			SUBMISSION OF SPRINT 4			

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint start date	Sprint end date(planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint – 1	20	6 days	24 oct 2022	29 oct 2022	20	17 Nov 2002
Sprint – 2	20	6 days	31 oct 2022	5 Nov 2022	20	17 Nov 2022
Sprint – 3	20	6 days	7 Nov 2022	12 Nov 2022	20	18 Nov 2022
Sprint - 4	20	6 days	14 Nov 2022	19 Nov 2022	20	18 Nov 2022

Velocity:

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

AV sprint duration 20 2 velocity 10

6.3 REPORTS FROM JIRA

7. CODING & SOLUTIONING

7.1 FEATURE 1

```
    #send grid

defsendmsg(Mailid,message):
    importsmtplib
    fromemail.mime.multipartimportMIMEMultipart
    fromemail.mime.textimportMIMEText
    fromemail.mime.baseimportMIMEBase
    fromemailimportencoders

    fromaddr = "sampletest685@gmail.com"
    toaddr = Mailid

    # instance of MIMEMultipart
    msg = MIMEMultipart()

    # storing the senders email address
    msg['From'] = fromaddr

    # storing the receivers email address
    msg['To'] = toaddr

    # storing the subject
    msg['Subject'] = "Alert"

    # string to store the body of the mail
    body = message

    # attach the body with the msg instance
    msg.attach(MIMEText(body, 'plain'))
```

```
    # creates SMTP session
    s = smtplib.SMTP('smtp.gmail.com', 587)

    # start TLS for security
    s.starttls()

    # Authentication
    s.login(fromaddr, "hneucvnontsuwgpj")

    # Converts the Multipart msg into a string
    text = msg.as_string()

    # sending the mail
    s.sendmail(fromaddr, toaddr, text)

    # terminating the session



if__name__ == '__main__':
    app.run(host='0.0.0.0', debug='TRUE')
```

7.2 FEATURE 2

Integrating the Watson Assistant.

Associating the script with html pages.

7.3 DATABASE SCHEMA

A table is a data structure that organizes information into rows and columns. It can be used to both store and display data in a structured format. For example, databases store data in tables so that information can be quickly accessed from specific rows. Websites often use tables to display multiple rows of data on page. Spreadsheets combine both purposes of a table by storing and displaying data in a structured format.

Databases often contain multiple tables, with each one designed for a specific purpose. For example, a company database may contain separate tables for employees, clients, and suppliers. Each table may include its own set of fields, based on what data the table needs to store. In database tables, each field is considered a column, while each entry (or record), is considered a row. A specific value can be accessed from the table by requesting data from an individual column and row.

Company table

Field	**Type**
companyname	nvarchar(50)
regno	nvarchar(50)
mobile	nvarchar(50)
email	nvarchar(50)
website	nvarchar(50)
address	nvarchar(50)
username	nvarchar(50)
password	nvarchar(50)

Job table

Field	**Type**
Company name	nvarchar(50)
Contact no	nvarchar(50)
address	nvarchar(50)

location	nvarchar(50)
vacancy	nvarchar(50)
job	nvarchar(50)
department	nvarchar(50)
Website	nvarchar(50)
cname	nvarchar(50)

Registration table

Field	**Type**
Name	nvarchar(50)
gender	nvarchar(50)
age	nvarchar(50)
Email	nvarchar(50)
Phone	nvarchar(50)
Address	nvarchar(50)
degree	nvarchar(50)
department	nvarchar(50)
Username	nvarchar(50)
password	nvarchar(50)

8. TESTING

8.1 TEST CASES

A test case has components that describe input, action and an expected response, in order to determine if a feature of an application is working correctly. A test case is a set of instructions on "HOW" to validate a particular test objective/target, which when followed will tell us if the expected behavior of the system is satisfied or not.

Characteristics of a good test case:

- Accurate: Exacts the purpose.
- Economical: No unnecessary steps or words.
- Traceable: Capable of being traced to requirements.
- Repeatable: Can be used to perform the test over and over.
- Reusable: Can be reused if necessary.

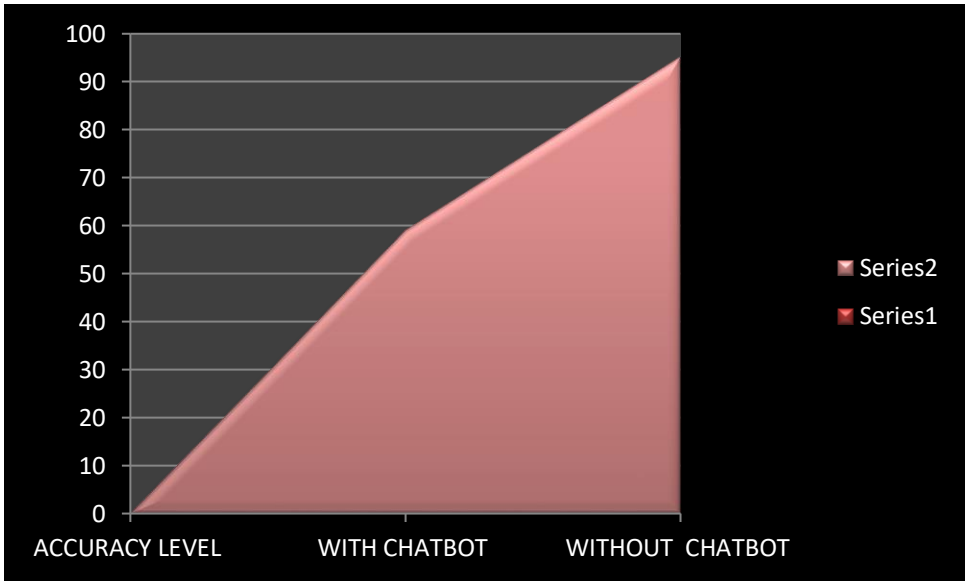
S.NO	**Scenario**	**Input**	**Excepted output**	**Actual output**
1	Admin Login Form	User name and password	Login	Login success.
2	User Registration Page	User Basic Details	Registered successfully	User registration details are stored in database.
3	User Login Form	User name and password	Login	Login success.
4	Update Skills Details	Skills Details	Updated successfully	Skills details are stored in database.

8.2 USER ACCEPTANCE TESTING

This is a type of testing done by users, customers, or other authorised entities to determine application/software needs and business processes. Acceptance testing is the most important phase of testing as this decides whether the client approves the application/software or not. It may involve functionality, usability, performance, and U.I of the application. It is also known as user acceptance testing (UAT), operational acceptance testing (OAT), and end-user testing.

9. RESULTS

9.1 PERFORMANCE METRICS



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- User can easily know about the company details.
- Automation of existing manual information systems.
- Reduction of manual processing
- Users will interact with the Chabot and can get the recommendations based on their skills.
- Keep track of daily information exchange at the server by the administrator.
- Increase in processing and transfer speeds of information over the network.
- Decrease in processing time

DISADVANTAGES

- Poor communication between user and company officer, so here intimating about new job is a hard task.
- Know the company job vacancy information is very difficult
- Immediate response to the queries is difficult.
- More stationary use so they are expensive.
- Manual system is takes more time.
- Existing system is manually, so it increases the chances of errors.

11. CONCLUSION

In this essay, we suggested a structure for the duty of job recommendations. The use of a variety of text processing and recommendation methods in accordance with the preferences of the job recommender system creator is permitted by this framework, which also makes it easier to comprehend the job suggestion process. Furthermore, we make a new dataset with profiles of job seekers and open positions publicly accessible. The coding is done in a simplified and easy to understandable manner so that other team trying to enhance the project can do so without facing much difficulty. The documentation will also assist in the process as it has also been carried out in a simplified and concise way.

12. FUTURE SCOPE

In future we can develop this project in android application. This system is developed such a way that additional enhancement can be done without much difficulty. The renovation of the project would increase the flexibility of the system. Also the features are provided in such a way that the system can also be made better and efficient functionality

- Try to all user contact with online.
- Add more features in site future.

13. APPENDIX

SOURCE CODE

App.py

```
from flask import Flask, render_template, flash, request, session
from flask import render_template, redirect, url_for, request

importjson
from json2html import *
importrequests

import ibm_db
importpandas
import ibm_db_dbi
fromsqlalchemyimportcreate_engine

engine = create_engine('sqlite://',
                       echo = False)

dsn_hostname = "98538591-7217-4024-b027-
8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud"
dsn_uid = "tvd24047"
dsn_pwd = "C0fhAXeLsuoQuvel"

dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "BLUDB"
dsn_port = "30875"
dsn_protocol = "TCPIP"
dsn_security = "SSL"

dsn = (
```

```
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
        "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port, dsn_protocol, dsn_uid, dsn_pwd,dsn_security)



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

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

app = Flask(__name__)
app.config['DEBUG']
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'



@app.route("/")
defhomepage():

    return render_template('index.html')
```

```
@app.route("/Home")
defHome():
    return render_template('index.html')

@app.route("/AdminLogin")
defAdminLogin():
    return render_template('AdminLogin.html')

@app.route("/NewUser")
defNewUser():
    return render_template('NewUser.html')

@app.route("/NewCompany")
defNewCompany():
    return render_template('NewCompany.html')

@app.route("/UserLogin")
defStudentLogin():
    return render_template('UserLogin.html')

@app.route("/CompanyLogin")
defCompanyLogin():
    return render_template('CompanyLogin.html')


@app.route("/Search")
defSearch():
    return render_template('Search.html')



@app.route("/AdminHome")
```

```
defAdminHome():

    conn = ibm_db.connect(dsn, "", "")
    pd_conn = ibm_db_dbi.Connection(conn)
    selectQuery = "SELECT * from regtb "
    dataframe = pandas.read_sql(selectQuery, pd_conn)

    dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
    data = engine.execute("SELECT * FROM Employee_Data").fetchall()
    return render_template('AdminHome.html', data=data)




@app.route("/ACompanyInfo")
defACompanyInfo():

    conn = ibm_db.connect(dsn, "", "")
    pd_conn = ibm_db_dbi.Connection(conn)
    selectQuery = "SELECT * from companytb "
    dataframe = pandas.read_sql(selectQuery, pd_conn)

    dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
    data = engine.execute("SELECT * FROM Employee_Data").fetchall()


    return render_template('ACompanyInfo.html', data=data)



@app.route("/AjobInfo")
defAjobInfo():
```

```
    conn = ibm_db.connect(dsn, "", "")
    pd_conn = ibm_db_dbi.Connection(conn)
    selectQuery = "SELECT * from jobtb "
    dataframe = pandas.read_sql(selectQuery, pd_conn)

    dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
    data = engine.execute("SELECT * FROM Employee_Data").fetchall()

    return render_template('AjobInfo.html', data=data)


@app.route("/SCompanyInfo")
defSCompanyInfo():


    conn = ibm_db.connect(dsn, "", "")
    pd_conn = ibm_db_dbi.Connection(conn)
    selectQuery = "SELECT * from jobtb "
    dataframe = pandas.read_sql(selectQuery, pd_conn)

    dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
    data = engine.execute("SELECT * FROM Employee_Data").fetchall()

    return render_template('SCompanyInfo.html', data=data)

@app.route("/CompanyHome")
defCompanyHome():
    return render_template('CompanyHome.html')


@app.route("/UserHome")
defUserHome():

    uname=  session['uname']
```

```
    conn = ibm_db.connect(dsn, "", "")
    pd_conn = ibm_db_dbi.Connection(conn)
    selectQuery = "SELECT * FROM regtb where Username="'+ uname +"' "
    dataframe = pandas.read_sql(selectQuery, pd_conn)
    dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
    data = engine.execute("SELECT * FROM Employee_Data").fetchall()

    return render_template('UserHome.html', data=data)




@app.route("/CJobInfo")
defCJobInfo():


    cname=  session['cname']


    conn = ibm_db.connect(dsn, "", "")
    pd_conn = ibm_db_dbi.Connection(conn)
    selectQuery = "SELECT * FROM jobtb where Cname="'+ cname +"' "
    dataframe = pandas.read_sql(selectQuery, pd_conn)
    dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
    data = engine.execute("SELECT * FROM Employee_Data").fetchall()



    return render_template('CJobInfo.html', data=data)


@app.route("/adminlogin", methods=['GET', 'POST'])
defadminlogin():
    error = None
    if request.method == 'POST':
      if request.form['uname'] == 'admin'or request.form['password'] == 'admin':
```

```
        conn = ibm_db.connect(dsn, "", "")
        pd_conn = ibm_db_dbi.Connection(conn)
        selectQuery = "SELECT * FROM regtb  "
        dataframe = pandas.read_sql(selectQuery, pd_conn)
        dataframe.to_sql('Employee_Data', con=engine, if_exists='append')
        data = engine.execute("SELECT * FROM Employee_Data").fetchall()

        return render_template('AdminHome.html', data=data)

    else:
     return render_template('index.html', error=error)



@app.route("/userlogin", methods=['GET', 'POST'])
defuserlogin():
   error = None
   if request.method == 'POST':

      username = request.form['uname']
      password = request.form['password']


      conn = ibm_db.connect(dsn, "", "")

      pd_conn = ibm_db_dbi.Connection(conn)

        selectQuery = "SELECT * from regtb where UserName='" + username + "' and
password='" + password + "'"
      dataframe = pandas.read_sql(selectQuery, pd_conn)
```

```
        ifdataframe.empty:
            data1 = 'Username or Password is wrong'
            return render_template('goback.html', data=data1)
        else:
            print("Login")
                selectQuery = "SELECT * from regtb where UserName='" + username + "' and password='" + password + "'"
            dataframe = pandas.read_sql(selectQuery, pd_conn)

            dataframe.to_sql('Employee_Data',
                             con=engine,
                             if_exists='append')

            # run a sql query
            print(engine.execute("SELECT * FROM Employee_Data").fetchall())

                return render_template('UserHome.html', data=engine.execute("SELECT * FROM Employee_Data").fetchall())



@app.route("/companylogin", methods=['GET', 'POST'])
defcompanylogin():
    error = None
    if request.method == 'POST':

        uname = request.form['uname']
        password = request.form['password']
        session['cname'] = uname


        conn = ibm_db.connect(dsn, "", "")

        pd_conn = ibm_db_dbi.Connection(conn)
```

```
        selectQuery = "SELECT * from companytb where UserName='" + uname + "' and password='" + password + "'"
    dataframe = pandas.read_sql(selectQuery, pd_conn)

    ifdataframe.empty:
        data1 = 'Username or Password is wrong'
        return render_template('goback.html', data=data1)
    else:
        print("Login")
         selectQuery = "SELECT * from companytb where UserName='" + uname + "' and password='" + password + "'"
        dataframe = pandas.read_sql(selectQuery, pd_conn)

        dataframe.to_sql('Employee_Data',
                         con=engine,
                         if_exists='append')

        # run a sql query
        print(engine.execute("SELECT * FROM Employee_Data").fetchall())

            return  render_template('CompanyHome.html',  data=engine.execute("SELECT * FROM Employee_Data").fetchall())

@app.route("/NewStudent1", methods=['GET', 'POST'])
defNewStudent1():
    if request.method == 'POST':

        name = request.form['name']
        gender = request.form['gender']
        Age = request.form['Age']
        email = request.form['email']
        pnumber = request.form['pnumber']
        address = request.form['address']
```

```
        Degree = request.form['Degree']
        depat = request.form['depat']
        uname = request.form['uname']
        passw = request.form['passw']

    conn = ibm_db.connect(dsn, "", "")

     insertQuery = "insert into regtb values('" + name + "','" + gender + "','" + Age + "','" + email + "','" + pnumber + "','" + address + "','" + Degree + "','" + depat + "','" + uname + "','" + passw + "')"
    insert_table = ibm_db.exec_immediate(conn, insertQuery)

    sendmsg(email, "Successfully registered this website")

    data1 = 'Record Saved!'
    return render_template('goback.html', data=data1)




@app.route("/newcompany", methods=['GET', 'POST'])
defnewcompany():
    if request.method == 'POST':

        cname = request.form['cname']
        regno = request.form['regno']
        mobile = request.form['mobile']

        email = request.form['email']
        Website = request.form['Website']
        address = request.form['address']
        uname = request.form['uname']
        passw = request.form['passw']
```

```
        conn = ibm_db.connect(dsn, "", "")

                                    insertQuery = "insert into companytb
values('"+cname+"','"+regno+"','"+mobile+"','"+email+"','"+Website+"','"+address+"','"+una
me +"','"+passw+"')"
        insert_table = ibm_db.exec_immediate(conn, insertQuery)

        data1 = 'Record Saved!'
        return render_template('goback.html', data=data1)




@app.route("/newjob", methods=['GET', 'POST'])
defnewjob():
    if request.method == 'POST':
        cnn = session['cname']
        cname = request.form['cname']
        cno = request.form['cno']
        Address = request.form['Address']
        JobLocation = request.form['JobLocation']
        Vacancy = request.form['Vacancy']
        Job = request.form['Job']
        Department = request.form['depat']
        website = request.form['website']

        conn = ibm_db.connect(dsn, "", "")

        insertQuery = "insert into jobtb values('" + cname + "','" + cno + "','" + Address + "','" +
JobLocation + "','" + Vacancy + "','" + Job + "','" + Department + "','" + website +
"','"+cnn+"')"
        insert_table = ibm_db.exec_immediate(conn, insertQuery)
```

```
        conn = ibm_db.connect(dsn, "", "")
        pd_conn = ibm_db_dbi.Connection(conn)
        selectQuery1 = "SELECT  *  FROM  regtb where  Department='" + Department + "'"
        dataframe = pandas.read_sql(selectQuery1, pd_conn)

        dataframe.to_sql('regtb', con=engine, if_exists='append')
        data1 = engine.execute("SELECT * FROM regtb").fetchall()

        foritem1indata1:
            Mobile = item1[5]
            Email = item1[4]
            sendmsg(Email,"Jop Title"+Job + " More Info Visit Website")




    data = 'Record Saved!'
    return render_template("goback.html", data=data)




@app.route("/jobsearch", methods=['GET', 'POST'])
defjobsearch():
    if request.method == 'POST':
        jobname = request.form['name']


        url = "https://linkedin-jobs-search.p.rapidapi.com/"

        payload = {
```

```
    "search_terms": jobname,
    "location": "india",
    "page": "1"
}
headers = {
    "content-type": "application/json",
    "X-RapidAPI-Key": "b045b9af95msha8d7c3160785729p1674cdjsnbdf4adbf9868",
    "X-RapidAPI-Host": "linkedin-jobs-search.p.rapidapi.com"
}

response = requests.request("POST", url, json=payload, headers=headers)

print(response.text)

infoFromJson = json.loads(response.text)

df = pandas.json_normalize(infoFromJson)

df.to_sql('regtb', con=engine, if_exists='append')
data1 = engine.execute("SELECT * FROM regtb").fetchall()



return render_template('Search.html',data=data1)
```

#send grid

```
defsendmsg(Mailid,message):
    importsmtplib
    fromemail.mime.multipartimportMIMEMultipart
    fromemail.mime.textimportMIMEText
    fromemail.mime.baseimportMIMEBase
    fromemailimportencoders

    fromaddr = "sampletest685@gmail.com"
    toaddr = Mailid

    # instance of MIMEMultipart
    msg = MIMEMultipart()

    # storing the senders email address
    msg['From'] = fromaddr

    # storing the receivers email address
    msg['To'] = toaddr

    # storing the subject
    msg['Subject'] = "Alert"

    # string to store the body of the mail
    body = message

    # attach the body with the msg instance
    msg.attach(MIMEText(body, 'plain'))

    # creates SMTP session
    s = smtplib.SMTP('smtp.gmail.com', 587)

    # start TLS for security
    s.starttls()
```

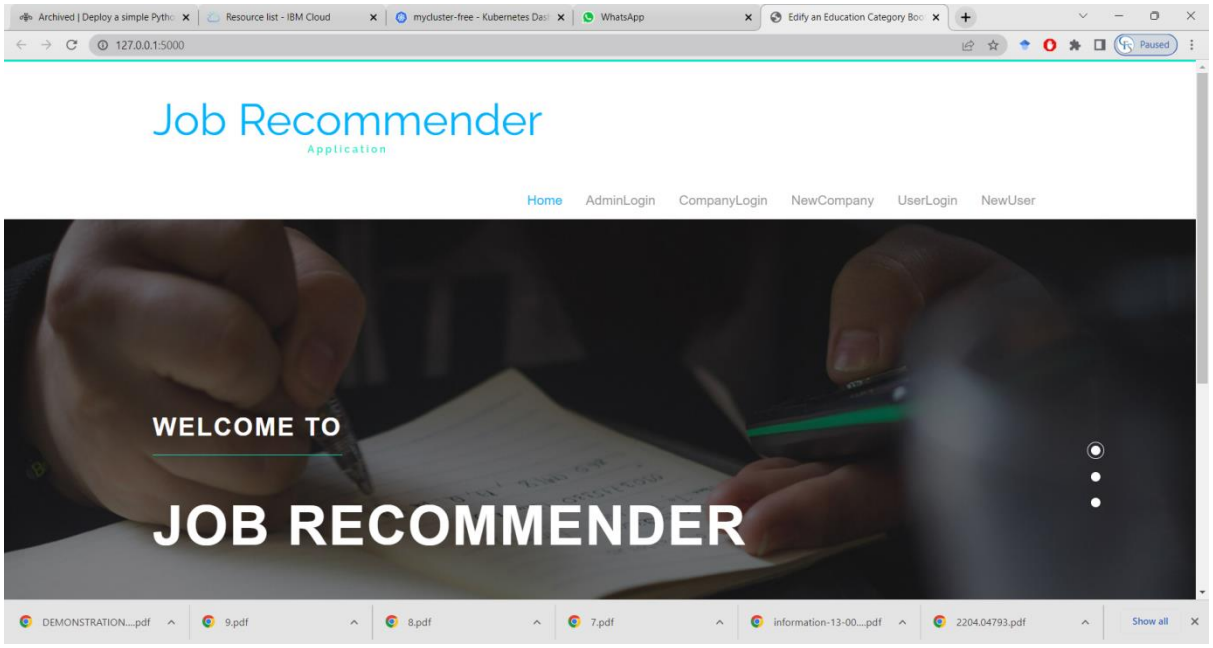
```
    # Authentication
    s.login(fromaddr, "hneucvnontsuwgpj")

    # Converts the Multipart msg into a string
    text = msg.as_string()

    # sending the mail
    s.sendmail(fromaddr, toaddr, text)

    # terminating the session




if__name__ == '__main__':
    app.run(host='0.0.0.0', debug='TRUE')
```

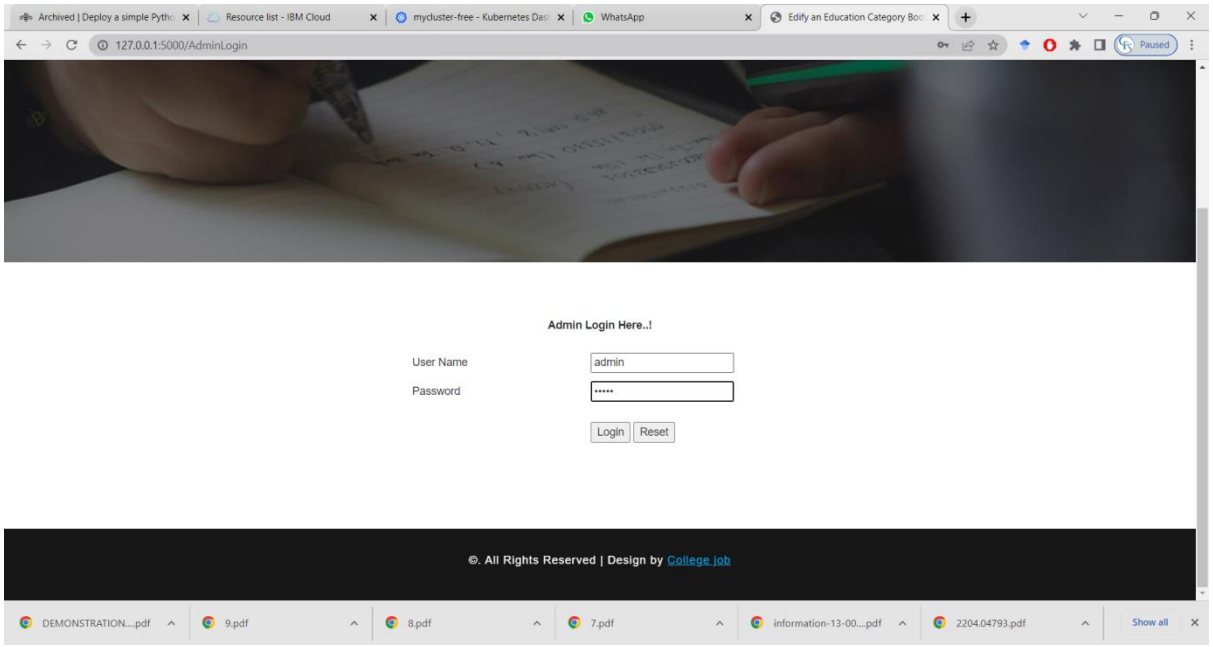
job.py

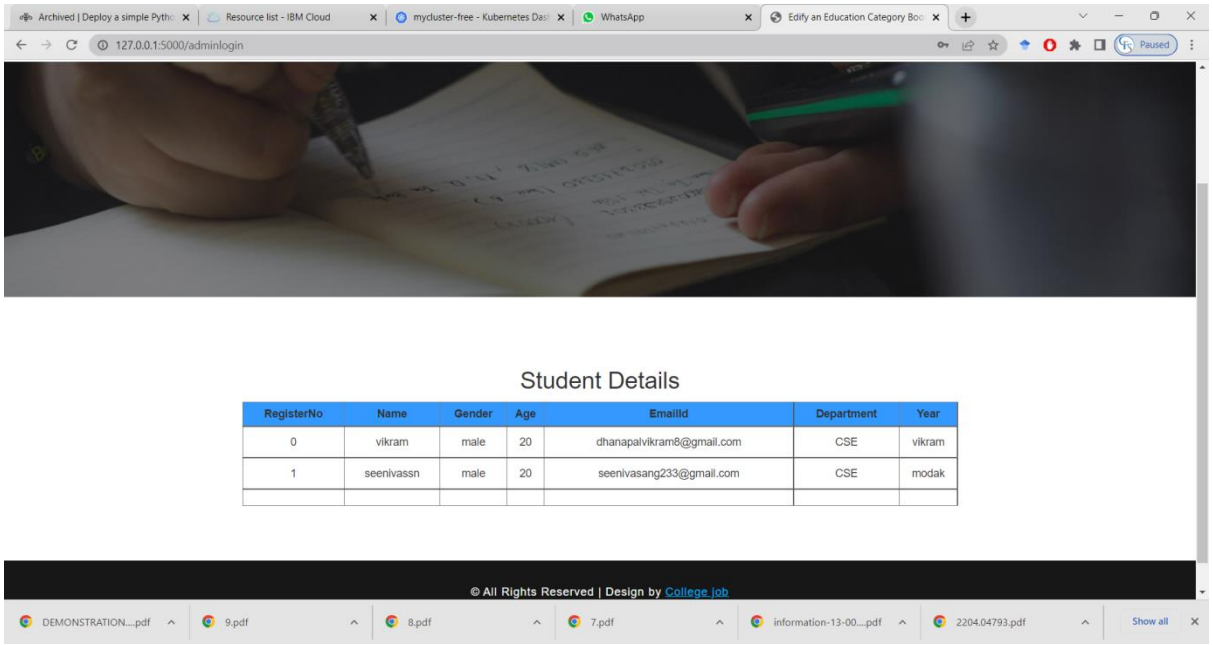
```
importrequests
importjson
importpandasaspd
from json2html import *
url = "https://linkedin-jobs-search.p.rapidapi.com/"

payload = {
    "search_terms": "python programmer",
    "location": "india",
    "page": "1"
}
headers = {
    "content-type": "application/json",
    "X-RapidAPI-Key": "b045b9af95msha8d7c3160785729p1674cdjsnbdf4adbf9868",
    "X-RapidAPI-Host": "linkedin-jobs-search.p.rapidapi.com"
```

```
}

response = requests.request("POST", url, json=payload, headers=headers)

print(response.text)

infoFromJson = json.loads(response.text)
print(json2html.convert(json = infoFromJson))
#data = json.loads(elevations)
df = pd.json_normalize(infoFromJson)
print(df)
```

SCREENSHOTS

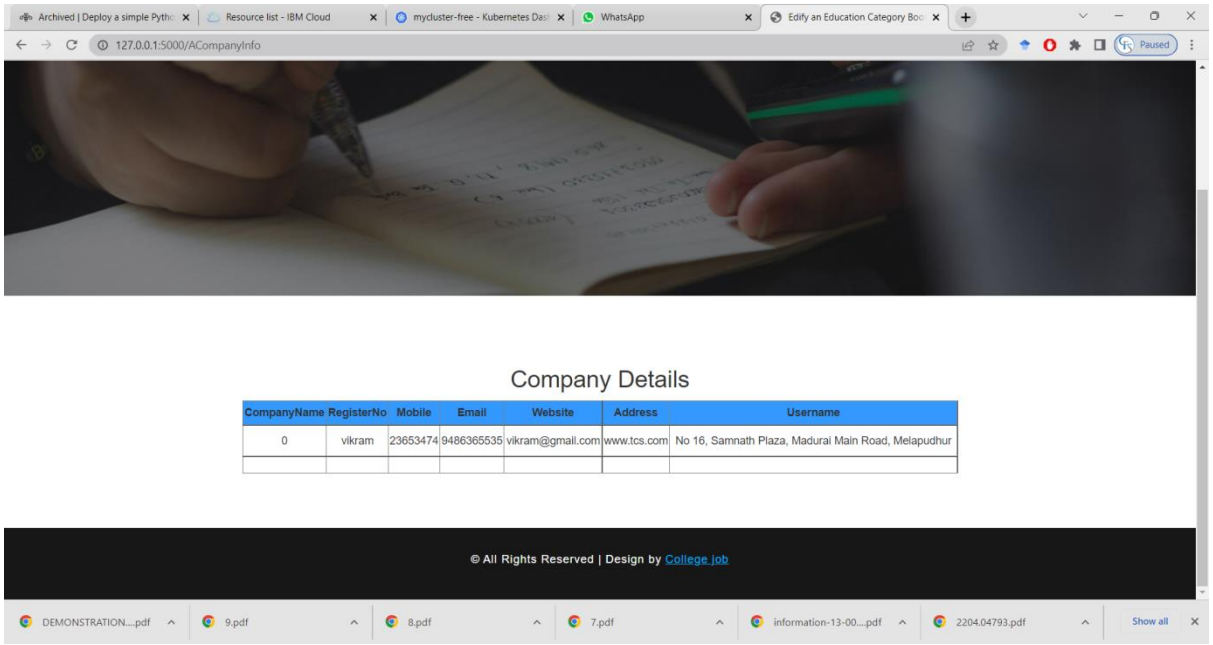






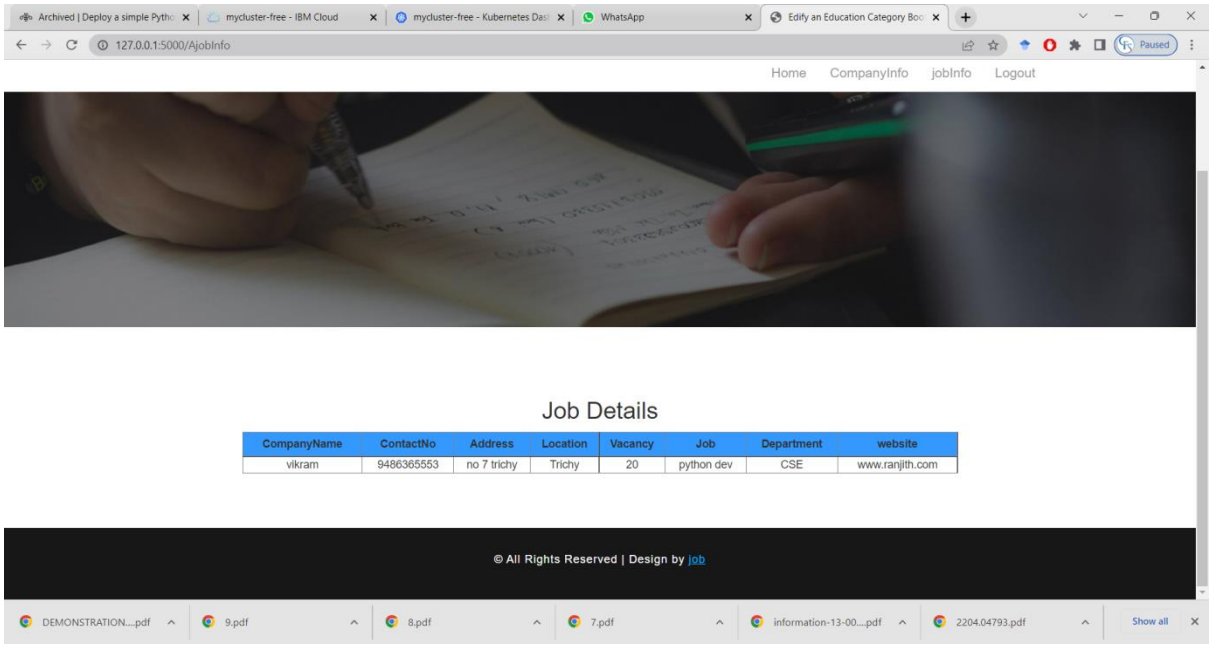
Student Details

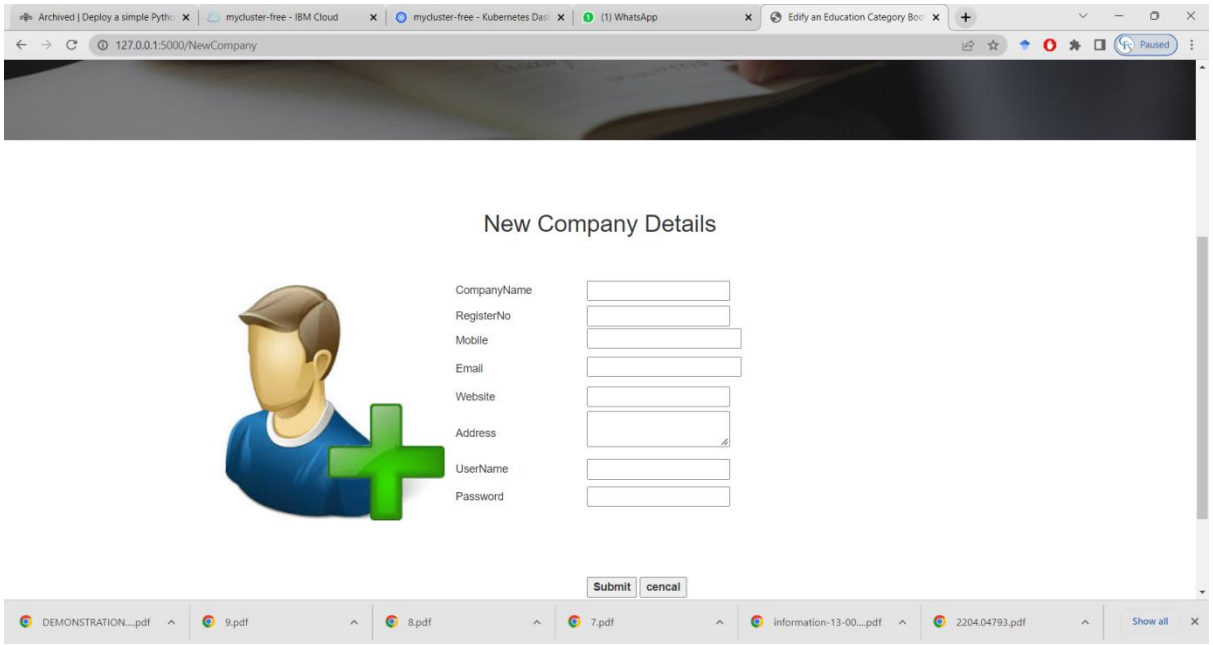
RegisterNo	Name	Gender	Age	EmailId	Department	Year
0	vikram	male	20	dhanapalvikram8@gmail.com	CSE	vikram
1	seenivassn	male	20	seenivasang233@gmail.com	CSE	modak

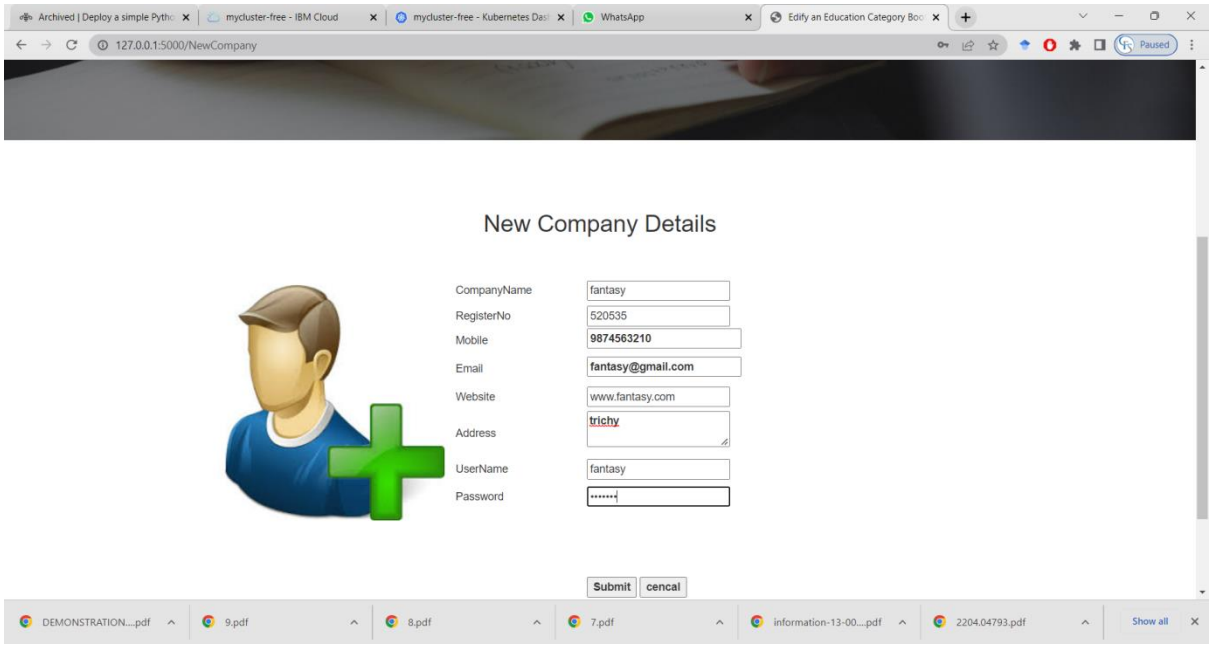


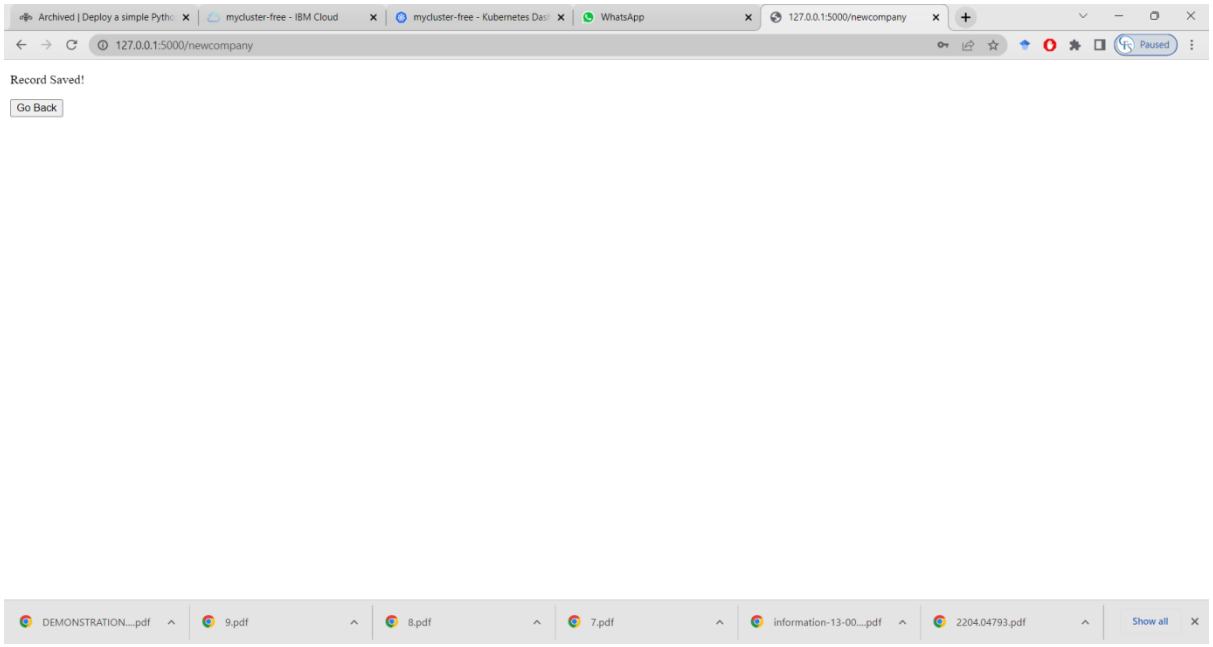
Company Details

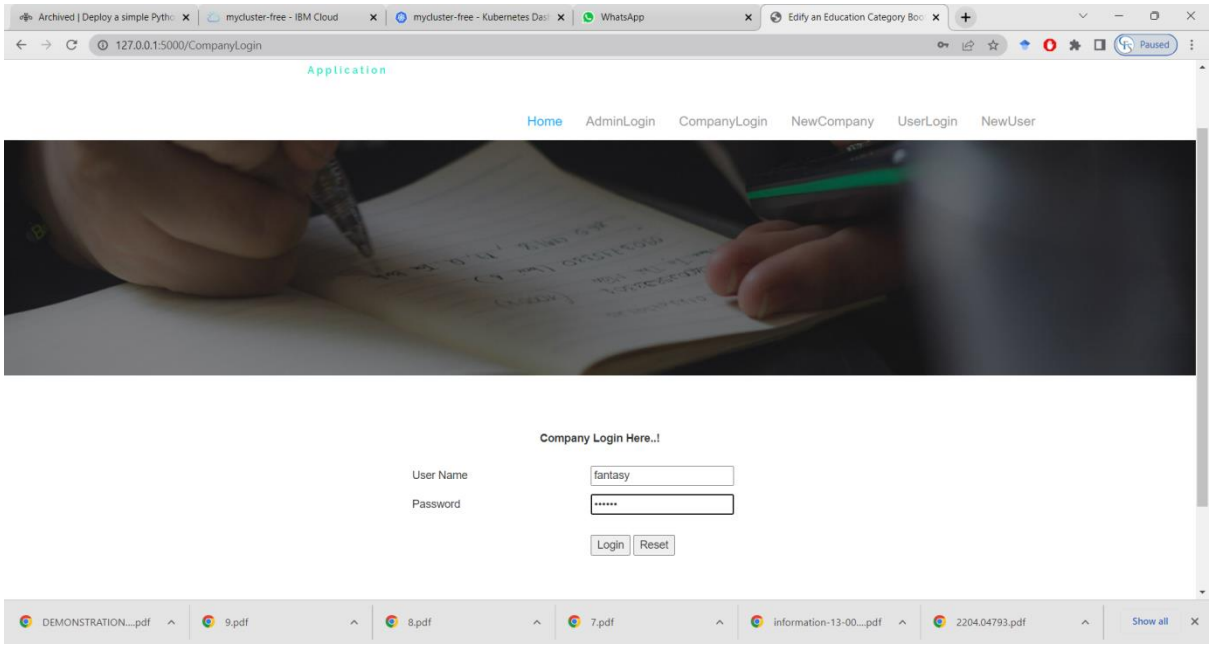
CompanyName	RegisterNo	Mobile	Email	Website	Address	Username
0	vikram	23653474	9486365535	vikram@gmail.com	www.tcs.com	No 16, Samnath Plaza, Madurai Main Road, Melapudhur

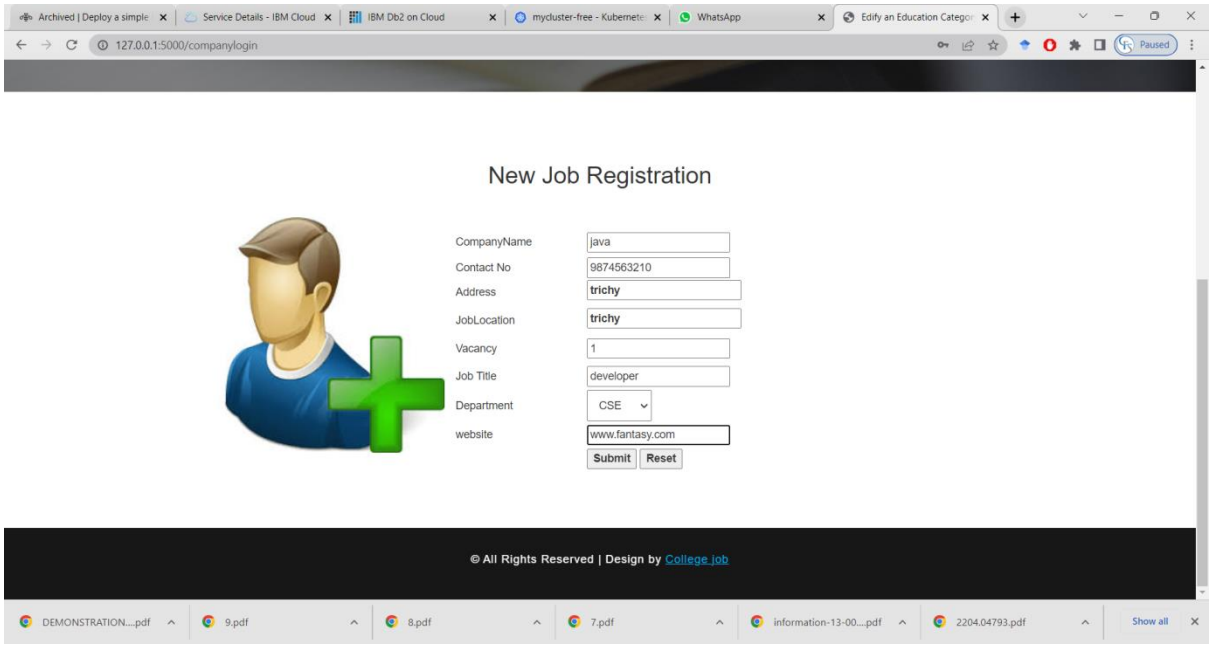

Archived | Deploy a simple Pytho
mycluster-free - IBM Cloud
mycluster-free - Kubernetes Das
WhatsApp
Edify an Education Category Boo
127.0.0.1:5000/AjobInfo
Paused
Home
CompanyInfo
jobInfo
Logout
Job Details
CompanyName
ContactNo
Address
Location
Vacancy
Job
Department
website
vikram
9486365553
no 7 trichy
Trichy
20
python dev
CSE
www.ranjith.com
© All Rights Reserved | Design by job
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all

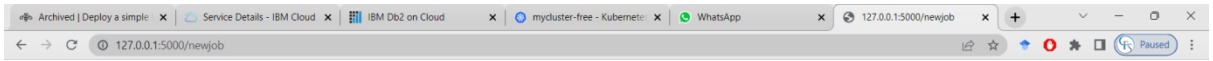

Archived | Deploy a simple Pytho
mycluster-free - IBM Cloud
mycluster-free - Kubernetes Das
(1) WhatsApp
Edify an Education Category Boo
127.0.0.1:5000/NewCompany
Paused
New Company Details
CompanyName
RegisterNo
Mobile
Email
Website
Address
UserName
Password
Submit
cencal
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all


Archived | Deploy a simple Pytho
mycluster-free - IBM Cloud
mycluster-free - Kubernetes Das
WhatsApp
Edify an Education Category Boo
127.0.0.1:5000/NewCompany
Paused
New Company Details
CompanyName
fantasy
RegisterNo
520535
Mobile
9874563210
Email
fantasy@gmail.com
Website
www.fantasy.com
Address
trichy
UserName
fantasy
Password
Submit
cencal
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all


Archived | Deploy a simple Pytho
mycluster-free - IBM Cloud
mycluster-free - Kubernetes Das
WhatsApp
127.0.0.1:5000/newcompany
127.0.0.1:5000/newcompany
Paused
Record Saved!
Go Back
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all

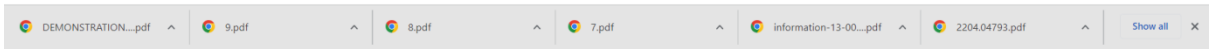

Application
Home
AdminLogin
CompanyLogin
NewCompany
UserLogin
NewUser
Company Login Here..!
User Name
fantasy
Password
Login
Reset

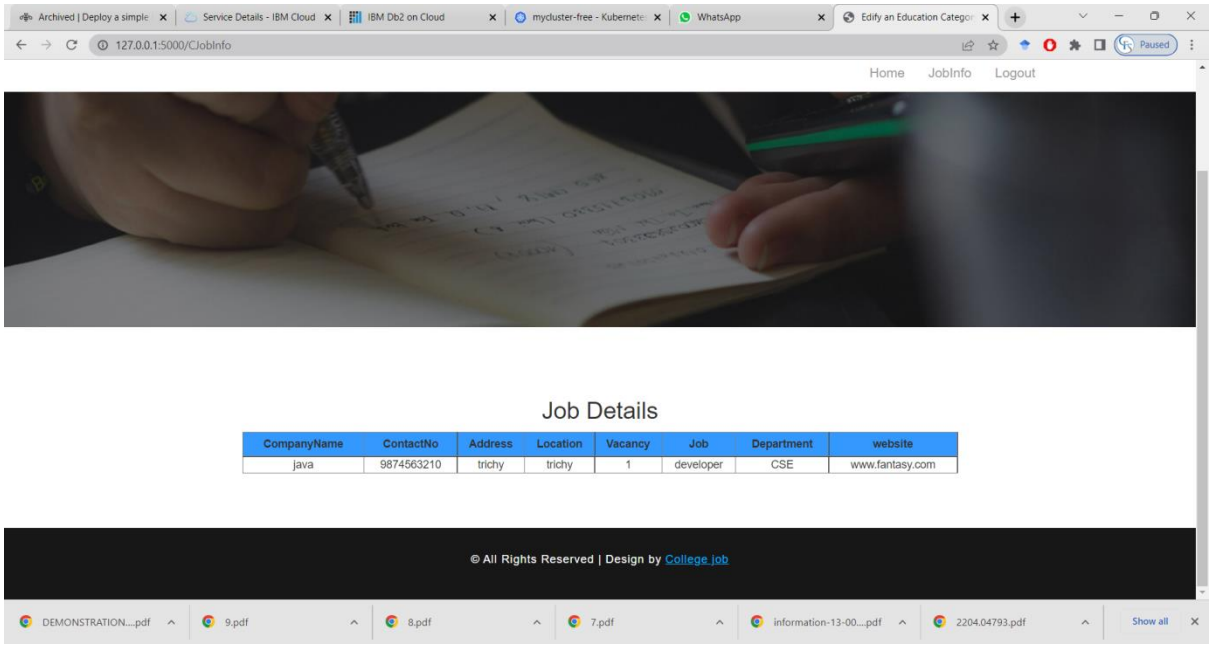

New Job Registration
CompanyName
java
Contact No
9874563210
Address
trichy
JobLocation
trichy
Vacancy
1
Job Title
developer
Department
CSE
website
www.fantasy.com
Submit
Reset
© All Rights Reserved | Design by College job



Record Saved!

Go Back

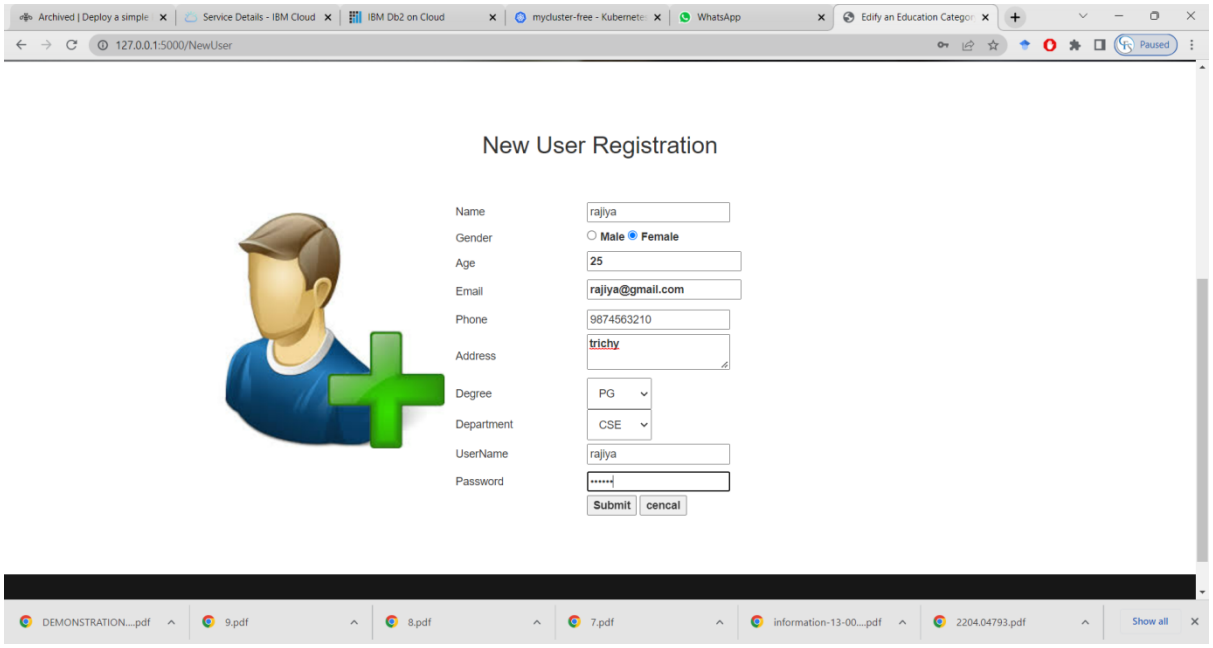


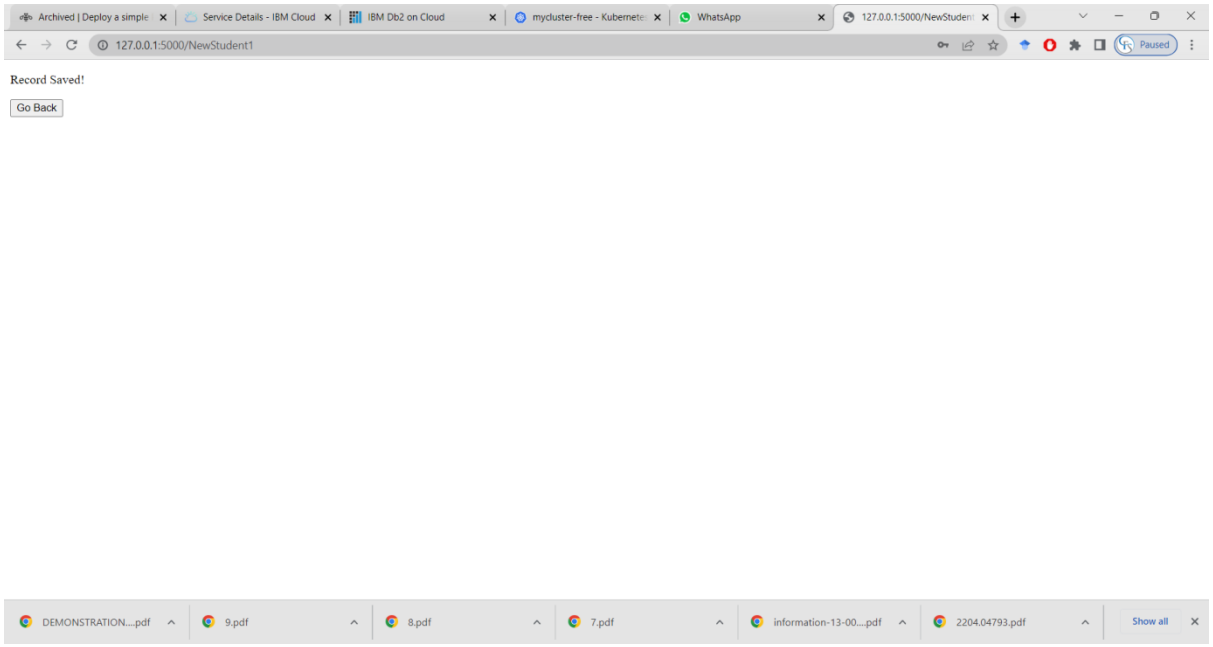


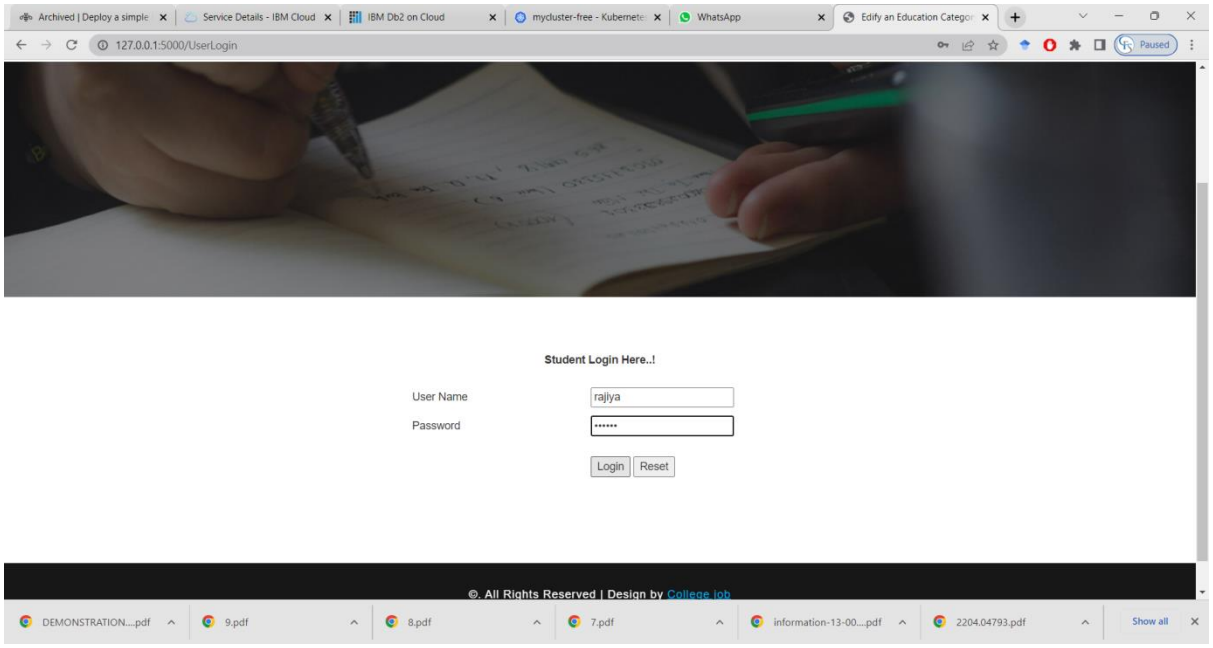
Job Details

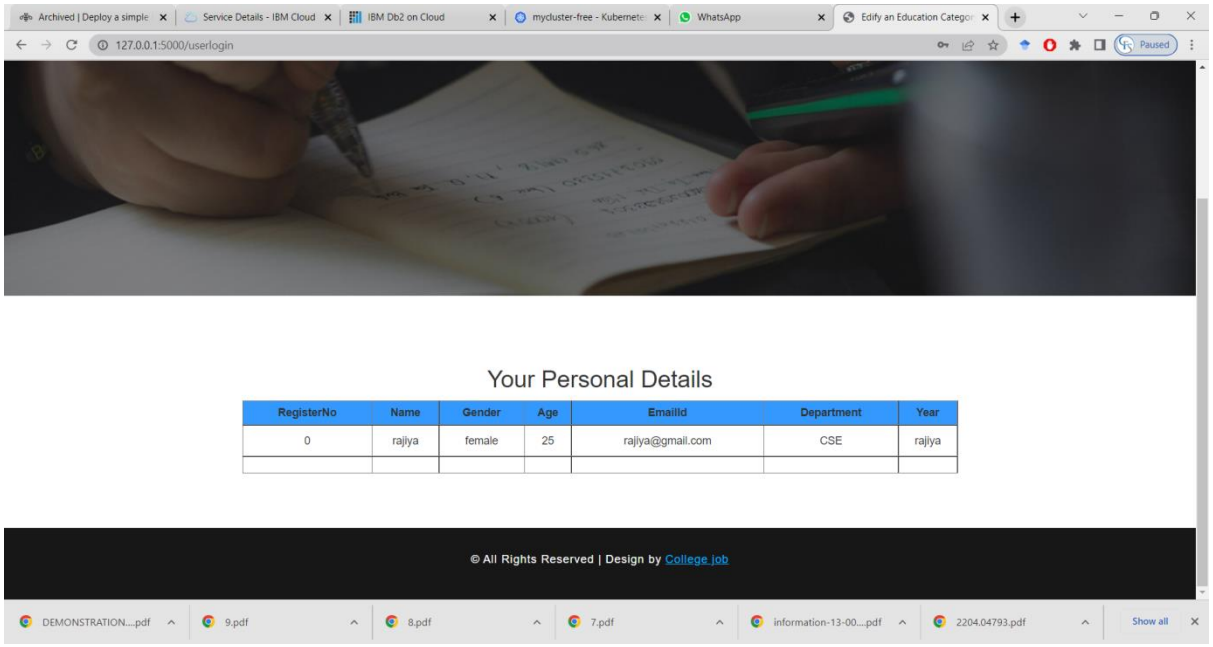
CompanyName	ContactNo	Address	Location	Vacancy	Job	Department	website
java	9874563210	trichy	trichy	1	developer	CSE	www.fantasy.com

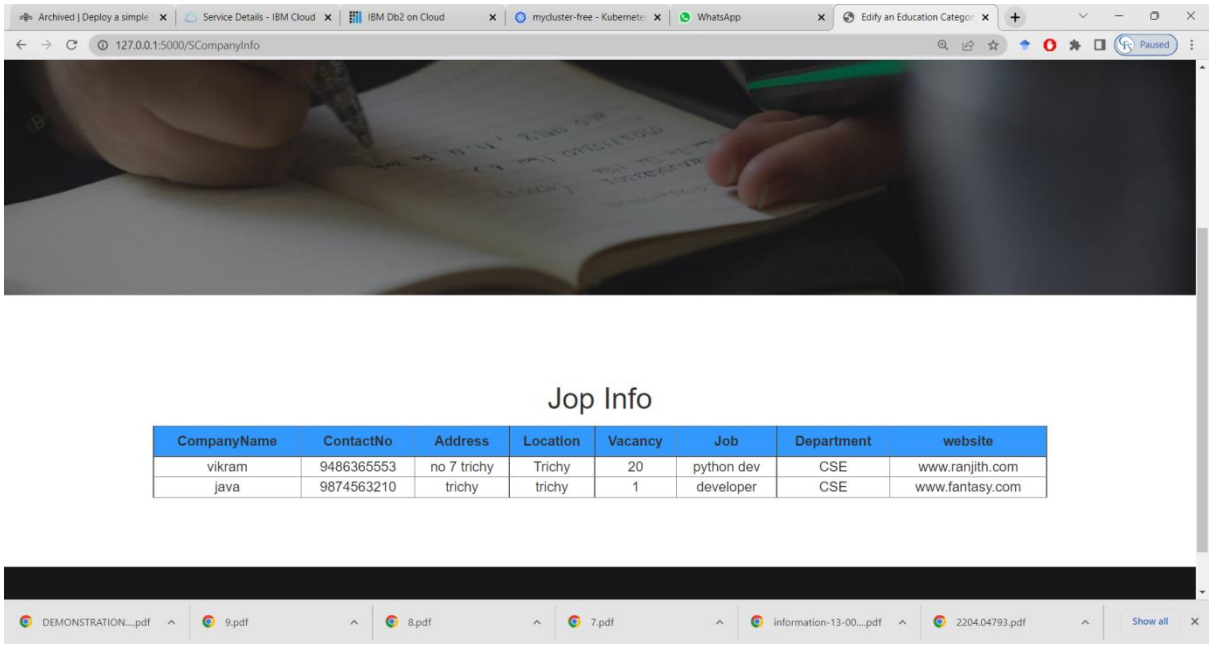
© All Rights Reserved | Design by College job


Archived | Deploy a simple
Service Details - IBM Cloud
IBM Db2 on Cloud
mycluster-free - Kubernete
WhatsApp
Edify an Education Categor
127.0.0.1:5000/NewUser
Paused
New User Registration
Name
rajiya
Gender
Male
Female
Age
25
Email
rajiya@gmail.com
Phone
9874563210
Address
trichy
Degree
PG
Department
CSE
UserName
rajiya
Password
Submit
cencal
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all


Archived | Deploy a simple
Service Details - IBM Cloud
IBM Db2 on Cloud
mycluster-free - Kubernete
WhatsApp
127.0.0.1:5000/NewStudent1
Paused
Record Saved!
Go Back
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all

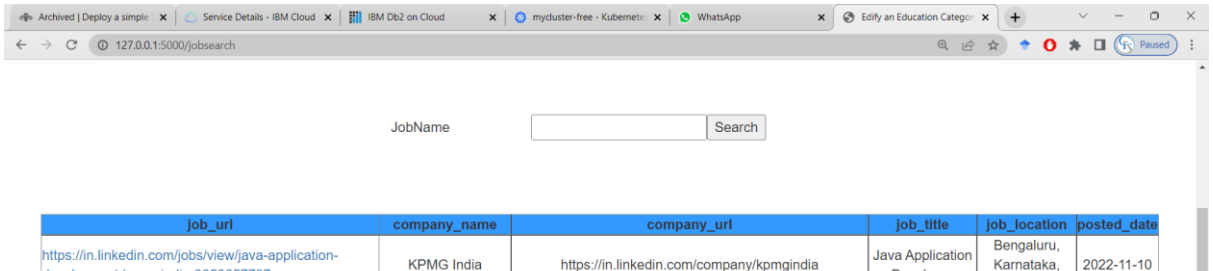

Archived | Deploy a simple
Service Details - IBM Cloud
IBM Db2 on Cloud
mycluster-free - Kubernete
WhatsApp
Edify an Education Categor
127.0.0.1:5000/UserLogin
Paused
Student Login Here..!
User Name
rajiya
Password
Login
Reset
©. All Rights Reserved | Design by College job
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all


Archived | Deploy a simple
Service Details - IBM Cloud
IBM Db2 on Cloud
mycluster-free - Kubernete
WhatsApp
Edify an Education Categor
127.0.0.1:5000/userlogin
Paused
Your Personal Details
RegisterNo
Name
Gender
Age
EmailId
Department
Year
0
rajiya
female
25
rajiya@gmail.com
CSE
rajiya
© All Rights Reserved | Design by College job
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all



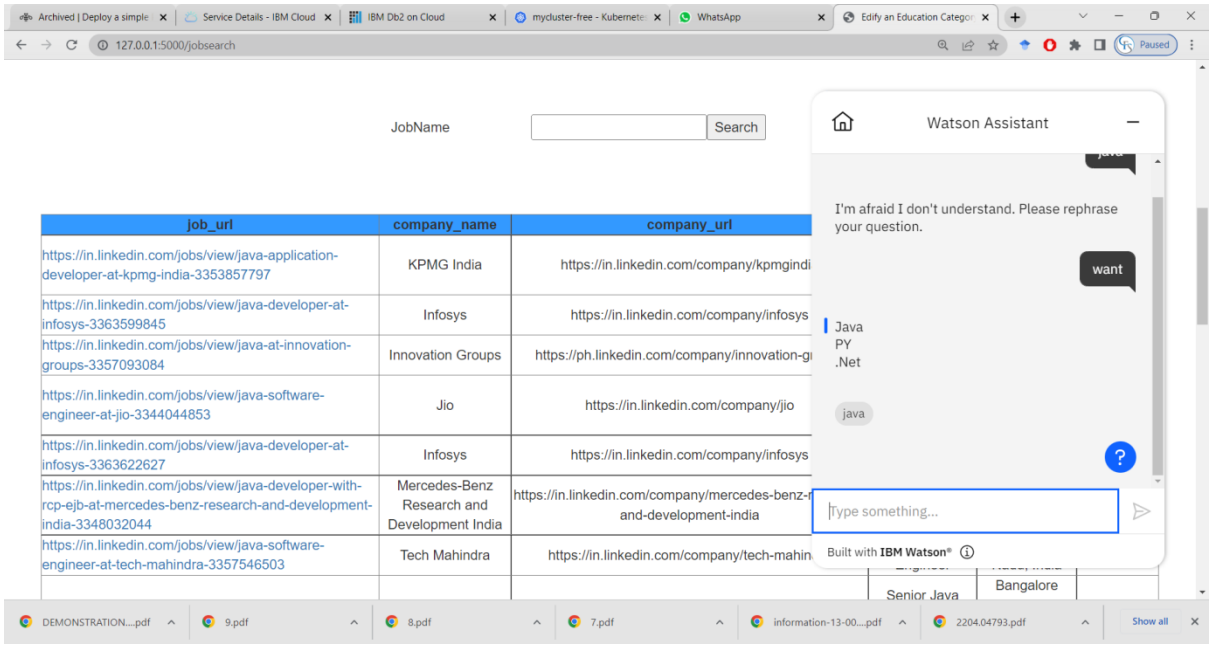
Jop Info

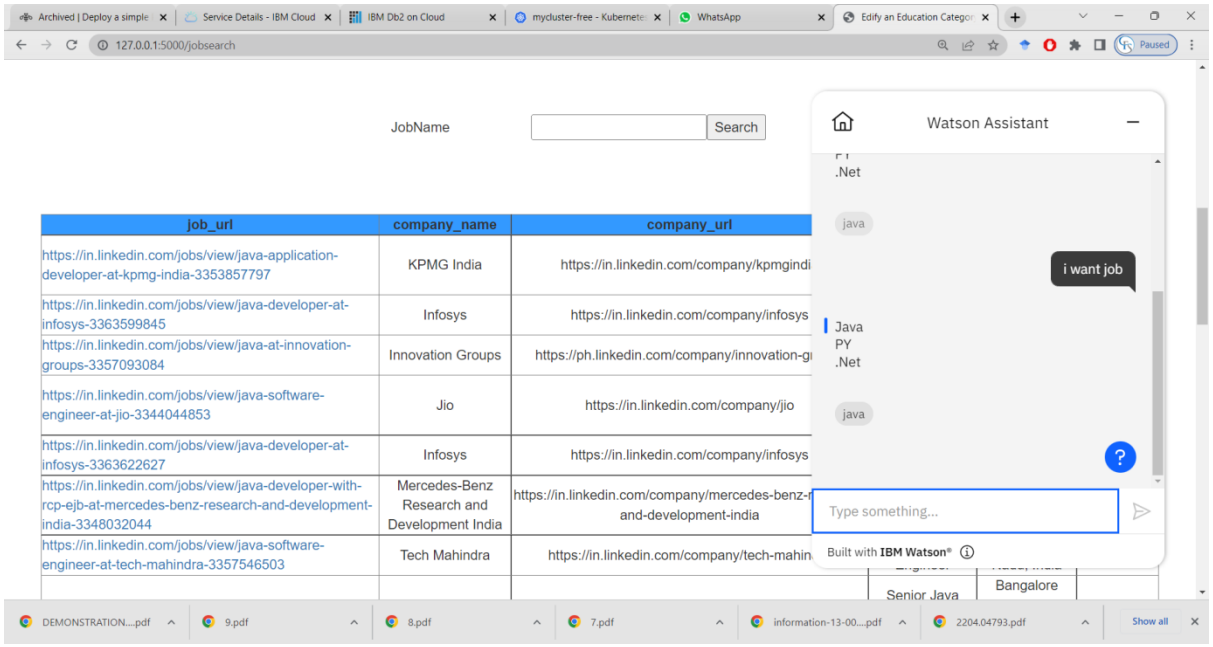
CompanyName	ContactNo	Address	Location	Vacancy	Job	Department	website
vikram	9486365553	no 7 trichy	Trichy	20	python dev	CSE	www.ranjith.com
java	9874563210	trichy	trichy	1	developer	CSE	www.fantasy.com

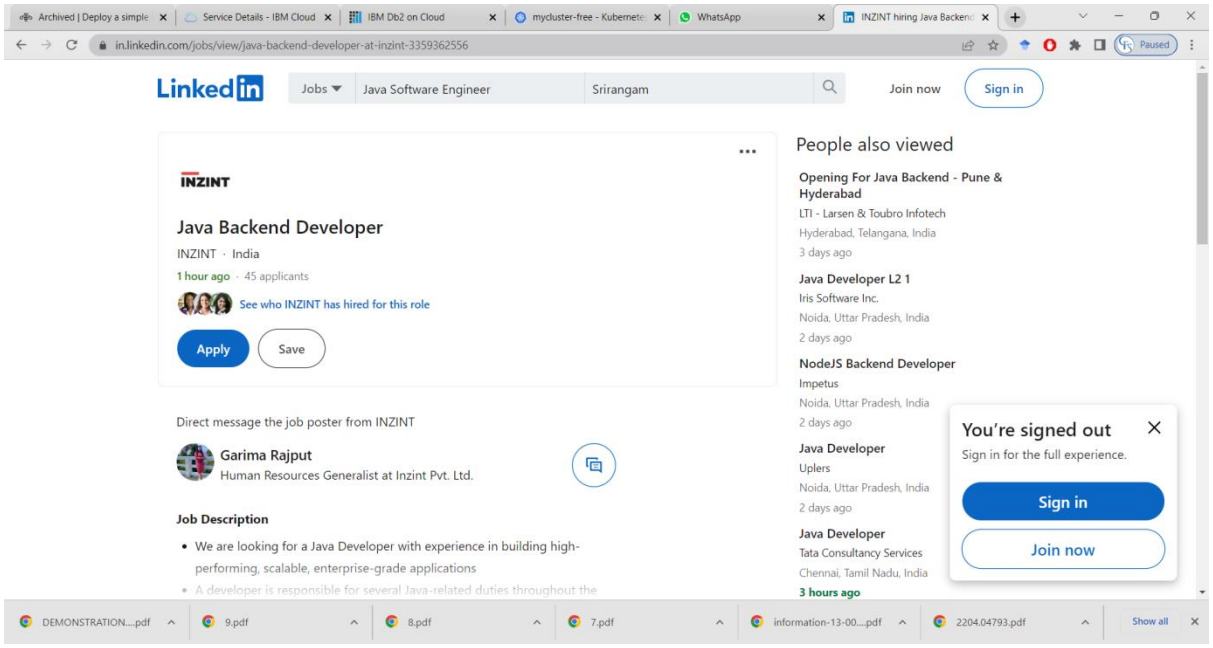


JobName Search

job_url	company_name	company_url	job_title	job_location	posted_date
https://in.linkedin.com/jobs/view/java-application-developer-at-kpmg-india-3353857797	KPMG India	https://in.linkedin.com/company/kpmgindia	Java Application Developer	Bengaluru, Karnataka, India	2022-11-10
https://in.linkedin.com/jobs/view/java-developer-at-infosys-3363599845	Infosys	https://in.linkedin.com/company/infosys	Java Developer	India	2022-11-17
https://in.linkedin.com/jobs/view/java-at-innovation-groups-3357093084	Innovation Groups	https://ph.linkedin.com/company/innovation-groups	JAVA	New Delhi, Delhi, India	2022-11-17
https://in.linkedin.com/jobs/view/java-software-engineer-at-jio-3344044853	Jio	https://in.linkedin.com/company/jio	Java Software Engineer	Mumbai, Maharashtra, India	2022-11-09
https://in.linkedin.com/jobs/view/java-developer-at-infosys-3363622627	Infosys	https://in.linkedin.com/company/infosys	Java Developer	India	2022-11-17
https://in.linkedin.com/jobs/view/java-developer-with-rcp-ejb-at-mercedes-benz-research-and-development-india-3348032044	Mercedes-Benz Research and Development India	https://in.linkedin.com/company/mercedes-benz-research-and-development-india	Java Developer with RCP/EJB	Bengaluru, Karnataka, India	2022-11-11
https://in.linkedin.com/jobs/view/java-software-engineer-at-tech-mahindra-3357546503	Tech Mahindra	https://in.linkedin.com/company/tech-mahindra	Java Software Engineer	Chennai, Tamil Nadu, India	2022
			Senior Java	Bangalore	


Archived | Deploy a simple
Service Details - IBM Cloud
IBM Db2 on Cloud
mycluster-free - Kubernete
WhatsApp
Edify an Education Categor
127.0.0.1:5000/jobsearch
Paused
JobName
Search
Watson Assistant
I'm afraid I don't understand. Please rephrase your question.
want
Java
PY
.Net
java
Type something...
Built with IBM Watson®
job_url
company_name
company_url
https://in.linkedin.com/jobs/view/java-application-developer-at-kpmg-india-3353857797
KPMG India
https://in.linkedin.com/company/kpmgindi
https://in.linkedin.com/jobs/view/java-developer-at-infosys-3363599845
Infosys
https://in.linkedin.com/company/infosys
https://in.linkedin.com/jobs/view/java-at-innovation-groups-3357093084
Innovation Groups
https://ph.linkedin.com/company/innovation-gr
https://in.linkedin.com/jobs/view/java-software-engineer-at-jio-3344044853
Jio
https://in.linkedin.com/company/jio
https://in.linkedin.com/jobs/view/java-developer-at-infosys-3363622627
Infosys
https://in.linkedin.com/company/infosys
https://in.linkedin.com/jobs/view/java-developer-with-rcp-ejb-at-mercedes-benz-research-and-development-india-3348032044
Mercedes-Benz Research and Development India
https://in.linkedin.com/company/mercedes-benz-r
and-development-india
https://in.linkedin.com/jobs/view/java-software-engineer-at-tech-mahindra-3357546503
Tech Mahindra
https://in.linkedin.com/company/tech-mahin
Senior Java
Bangalore
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all


Archived | Deploy a simple
Service Details - IBM Cloud
IBM Db2 on Cloud
mycluster-free - Kubernete
WhatsApp
Edify an Education Categor
127.0.0.1:5000/jobsearch
Paused
JobName
Search
Watson Assistant
.Net
java
i want job
Java
PY
.Net
java
Type something...
Built with IBM Watson®
job_url
company_name
company_url
https://in.linkedin.com/jobs/view/java-application-developer-at-kpmg-india-3353857797
KPMG India
https://in.linkedin.com/company/kpmgindi
https://in.linkedin.com/jobs/view/java-developer-at-infosys-3363599845
Infosys
https://in.linkedin.com/company/infosys
https://in.linkedin.com/jobs/view/java-at-innovation-groups-3357093084
Innovation Groups
https://ph.linkedin.com/company/innovation-gr
https://in.linkedin.com/jobs/view/java-software-engineer-at-jio-3344044853
Jio
https://in.linkedin.com/company/jio
https://in.linkedin.com/jobs/view/java-developer-at-infosys-3363622627
Infosys
https://in.linkedin.com/company/infosys
https://in.linkedin.com/jobs/view/java-developer-with-rcp-ejb-at-mercedes-benz-research-and-development-india-3348032044
Mercedes-Benz Research and Development India
https://in.linkedin.com/company/mercedes-benz-r
and-development-india
https://in.linkedin.com/jobs/view/java-software-engineer-at-tech-mahindra-3357546503
Tech Mahindra
https://in.linkedin.com/company/tech-mahin
Senior Java
Bangalore
DEMONSTRATION....pdf
9.pdf
8.pdf
7.pdf
information-13-00....pdf
2204.04793.pdf
Show all



GITHUB & PROJECT DEMO LINK

- **https://github.com/IBM-EPBL/IBM-Project-13266-1659515508**
- **https://youtu.be/DQ92J8lBa-w**