Project Report Format

1. **INTRODUCTION**
    a. Project Overview
    b. Purpose
2. **LITERATURE SURVEY**
    a. Existing problem
    b. References
    c. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
    a. Empathy Map Canvas
    b. Ideation & Brainstorming
    c. Proposed Solution
    d. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
    a. Functional requirement
    b. Non-Functional requirements
5. **PROJECT DESIGN**
    a. Data Flow Diagrams
    b. Solution & Technical Architecture
    c. User Stories
6. **PROJECT PLANNING & SCHEDULING**
    a. Sprint Planning & Estimation
    b. Sprint Delivery Schedule
    c. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    a. Feature 1
    b. Feature 2
    c. Database Schema (if Applicable)
8. **TESTING**
    a. Test Cases
    b. User Acceptance Testing
9. **RESULTS**
    a. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**

12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

1.INTRODUCTION:

1.1 Project Overview:

The recommender systems are being used in every possible system, for example, clothes recommendation, book recommendation, etc. However, the type of recommendations provided may be different according to the domain of its use. In the case of job recommendation system the case is a little bit different. Here, it will be favorable to provide mostly personalized and profile-based job recommendations. In job recommendation systems, there are varieties of students, having different education levels and skills. Based on students' respective background details, each one of them expects to get only those job recommendations that are highly relevant for that particular student.

1.2 Purpose:

- To make all the users employed according to their suggestions and their skills and job suggestions which are suitable to their opportunities.
- To test the skills and knowledge of the users who needs job.

2.LITERATURE SURVEY:

TITLE	Cost-Effective and Interpretable Job Skill Recommendation with Deep Reinforcement Learning
AUTHORS	Ying Sun, Fuzhen Zhuang, Hengshu Zhu, Qing He, Hui Xiong

YEAR OF PUBLICATION	April 2021
ABSTRACT	Nowadays, as organizations operate in very fast-paced and competitive environments, workforce has to be agile and adaptable to regularly learning new job skills. However, it is nontrivial for talents to know which skills to develop at each working stage. To this end, in this paper, we aim to develop a cost-effective recommendation system based on deep reinforcement learning, which can provide personalized and interpretable job skill recommendation for each talent. Specifically, we first design an environment to estimate the utilities of skill learning by mining the massive job advertisement data, which includes a skill-matching-based salary estimator and a frequent itemset-based learning difficulty estimator. Based on the environment, we design a Skill Recommendation Deep Q-Network (SRDQN) with multi-task structure to estimate the long-term skill learning utilities. In particular, SRDQN recommends job skills in a personalized and cost-effective manner; that is, the talents will only learn the recommended necessary skills for achieving their career goals. Finally, extensive experiments on a real-world dataset clearly validate the effectiveness and interpretability of our approach.
METHODOLOGY	Data Mining and Deep reinforcement learning
MERITS	Cost effective
DEMERITS	Must improve the performance for potential application such as curriculum recommendation
OVERCOME DEMERITS	Using CNN for comparable profile to be more potential
LINK	https://doi.org/10.1145/3442381.3449985

TITLE	Prediction of recommendations for employment utilizing machine learning procedures and geo-area based recommender framework
AUTHORS	Binny Parida, Prashanta Kumar Patra, Sthitapragyan Mohanty
YEAR OF PUBLICATION	19 November 2021
ABSTRACT	With increment in the utilization of Internet, the pace of increment of social networks is getting ubiquitous in recent years. This paper focuses on the job portal websites. The research objective of this paper is that the recommender framework takes the abilities from the website and makes suggestion to the candidates with the jobs whose descriptions are coordinating with their profiles the most. This paper additionally presents a short presentation on recommender framework and talks about different categories of this framework. From the start, information is cleaned by expelling the filthy information as extra space and duplicates. Then the job recommendations are made to the target applicants on the basis of their preferences. It utilizes different Machine Learning procedures which results show that Random Forest Classifier (RFC) gives the most noteworthy expectation accuracy when contrasted with different procedures. Finally, the optimization technique is utilized to get the most exact outcome. The advantage of recommender framework in career orientation is expressed. Geo-area based recommendation framework is utilized to find the organization's position which can assist the ideal applicants with reaching their destination. This examination shows that the utilization of job recommender system can assist with improving the recommendation of appropriate

	employment for work searchers
METHODOLOGY	Machine Learning
MERITS	Comparing multiple algorithms.
DEMERITS	Doesn’t find for all the close by location organization
OVERCOME DEMERITS	Using Nearest Neighbor algorithm can suggest a close by location organization.

LINK	https://doi.org/10.1016/j.susoc.2021.11.001

TITLE	***A Personalized Brand Proposal Based on User's Satisfaction and Curriculum Supported by an Intelligent Job Recommender System***
AUTHORS	Patricia Rayón Villela , Nelly Rigaud Téllez
YEAR OF PUBLICATION	25 July 2021

ABSTRACT	One of the main challenges' universities are confronted is the personalization of education services to improve quality mechanisms and strategies for supporting and assisting students when entering the workforce. Although many universities try to narrow the gap between academic life and job market, it is a highly challenging task to identify the right job for the right graduate. Market strives to find the most talented people and universities attempt to enrich students' personal brands, but these do not always align. Pitfalls are found in obtaining proper information that harmonize employment offers, course content and graduate's profile. This research places a transversal analysis of job mismatch in Latin American (LATAM) countries, builds a personalized brand based on satisfaction and course content and offers descriptions for an intelligent job recommender system. Proposal considers that providing a targeted job match implies by picking quantitatively relevant technical knowledge and transversal competencies of individual graduates and matching them to knowledge, skills and attitudes of employment offers and course content, in an efficient manner. Competencies from employment offers obtained with text mining are related to those from a current curriculum to help graduates bring about a personal brand for an appropriate job. Contribution of this research is the construction of a framework to construct match patterns that benefits graduates to meet professional success and to achieve personalization and optimization of the universities' offered services that represents an incremental improvement.

METHODOLOGY	Data Mining
MERITS	High Accuracy
DEMERITS	Doesn’t have the support for the geo based search
OVERCOME DEMERITS	Using machine learning and deep learning
LINK	10.1007/978-981-16-3941-8_12

TITLE	***Embedding-based Recommender System for Job to Candidate Matching on Scale***
AUTHORS	Jing Zhao, Jingya Wang, Madhav Sigdel, Bopeng Zhang, Phuong Hoang, Mengshu Liu and Mohammed Korayem
YEAR OF PUBLICATION	1 July 2021

ABSTRACT	The online recruitment matching system has been the core technology and service platform in CareerBuilder. One of the major challenges in an online recruitment scenario is to provide good matches between job posts and candidates using a recommender system on the scale. In this paper, we discussed the techniques for applying an embedding-based recommender system for the large scale of job to candidates matching. To learn the comprehensive and effective embedding for job posts and candidates, we have constructed a fused-embedding via different levels of representation learning from raw text, semantic entities and location information. The clusters of fused-embedding of job and candidates are then used to build and train the Faiss index that supports runtime approximate nearest neighbor search for candidate retrieval. After the first stage of candidate retrieval, a second stage reranking model that utilizes other contextual information was used to generate the final matching result. Both offline and online evaluation results indicate a significant improvement of our proposed two-staged embedding based system in terms of click-through rate (CTR), quality and normalized discounted accumulated gain (nDCG), compared to those obtained from our baseline system. We further described the deployment of the system that supports the million-scale job and candidate matching process at CareerBuilder. The overall improvement of our job to candidate matching system has demonstrated its feasibility and scalability at a major online recruitment site.
METHODOLOGY	Deep learning

MERITS	High Accuracy
DEMERITS	Didn’t compared with many algorithms.
OVERCOME DEMERITS	Compare more models with the same data.
LINK	https://doi.org/10.48550/arXiv.2107.00221

TITLE	***Job Recommendation based on Job Profile Clustering and Job Seeker behaviour.***
AUTHORS	D. Mhamdi*, R. Moulouki, M. Y. El Ghoumari, M. Azzouazi, L. Moussaid
YEAR OF PUBLICATION	August 2020
ABSTRACT	This article presents a recommender system that aims to help job seekers to find suitable jobs. First, job offers are collected from job search websites then they are prepared to extract meaningful attributes such as job titles and technical skills. Job offers with common features are grouped into clusters. As job seeker like one job belonging to a cluster, he will probably find other jobs in that cluster that he will like as well. A list of top n recommendations is suggested after matching data from job clusters and job seeker behavior, which consists on user interactions such as applications,

	likes and rating
METHODOLOGY	Clustering and Artificial Intelligence
MERITS	Good Accuracy
DEMERITS	Doesn't find perfect job with the required user skill
OVERCOME DEMERITS	using Word2vec method and k-means clustering algorithms used to capture and represent the context of job profiles
LINK	https://doi.org/10.1016/j.procs.2020.07.102

2 LITERATURE SURVEY

2.1 EXISTING PROBLEM

1. Confusing Application Process. Each job advertisement will have its own guidelines for filling out the application.
2. Staying Up to Date.
3. Having a Limited Professional Network.
4. Not Having the Right Degree.
5. No Feedback.

2.2 REFERENCES

1. [PDF] Recruitment and Job Search Application (researchgate.net)
2. (99+) Mobile Application System for Online Job seeker | vishakha Nagrale - Academia.edu
3. https://core.ac.uk/download/pdf/77979433.pdf

2.3 PROBLEM STATEMENT:

Date	19 September 2022
Team ID	PNT2022TMID27298
Project Name	Project – SKILL AND JOB RECOMMENDER
Maximum Marks	2 Marks

Customer Problem Statement Template:

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

Problem Statement (PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
PS-1	Freshers	Get a suitable job according to their skills	They are suffering to get a suitable job.	The freshers could not get a job according to their relevant skills.	Anxious

PS-2	Experienced persons	Get a job based on their skills and get a better salary	They are not getting better salaries according to their skills.	Experienced people could not get better salaries according to their skills.	Anxious

3. IDEATION & PROPOSED SOLUTION:

3.1 Empathy Map canvas:-

3.2 Ideation & Brain Storming:-

Date	19 September 2022
Team ID	PNT2022TMID27298
Project Name	Skill and Job Recommender
Maximum Marks	4 Marks

Brainstorm & Idea Prioritization Template:

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

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

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

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

Step-2: Brainstorm, Idea Listing and Grouping

Step-3: Idea Prioritization

3.3 Proposed Solution :-

Date	September 2022
Team ID	PNT2022TMID27298
Project Name	Skill and Job Recommender
Maximum Marks	2 Marks

Proposed Solution Template:

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

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The web application doesn't have all the relevant jobs that user seeks.
2.	Idea / Solution description	Based on the person-job fit premise, we propose a framework for job recommendation based on professional skills of job seekers. We automatically extracted the skills from the job seeker profiles using a variety of text processing techniques
3.	Novelty / Uniqueness	The skills and the job recommendation system is basic way of job recommendations by matching with the manually entered skills in the existing research projects. In this project the skills and the jobs are identified using the data taken from the student and the jobs are recommended based on the eligibility criteria with the specified skills.
4.	Social Impact / Customer Satisfaction	All skilled employees and fresher's are got employed using the recommender systems. Suitable jobs are suggested according the skills. The end users can choose their own interested jobs. For the experienced persons it's an easy way to get jobs instead of searching outside.
5.	Business Model (Revenue Model)	For 6 months, The user can access each and every feature in the web application for free and after that they have to subscribe for premium in order to continue the services.

6.	Scalability of the Solution	The recommender system will be adaptable based on the persons who are applying for the jobs. The infrastructure of the application should be maintained without the bugs. The ideas of the recommendation systems will be upgraded to the next level. The performance of the application will be effective and suitable to the hardware systems.

3.4 Problem Solution Fit:-

REQUIREMENT

4.REQUIREMENT ANALYSIS(FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS):-

Date	03 October 2022
Team ID	PNT2022TMID27298
Project Name	Skill and Job Recommender
Maximum Marks	4 Marks

4.1 Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN

FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Organisation Details	Details about organisation Details about vacancy of job
FR-4	Searching Job	Job details in clear manner. Variety of domains
FR-5	Optimised Details	Details of organisation or job in clear manner Optimised results for searching job.

4.2 Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It's an invaluable tool for job searches and recruitment.
NFR-2	**Security**	The security will in form of passwords, otp or the question is asked.
NFR-3	**Reliability**	It can be reliable because the user will be posting about their education and also certifications
NFR-4	**Performance**	Performance can be calculated based how many users are benefitted through site and also how user feel about the site.
NFR-5	**Availability**	The services like searching for job and applying for job always available
NFR-6	**Scalability**	This recommender system provides job to all people who are in need of job

5 PROJECT DESIGN :-

5.1 DATA FLOW DIAGRAMS:-

Date	20 October 2022
Team ID	PNT2022TMID27298
Project Name	Skill and Job Recommender

Maximum Marks	4 Marks

Data Flow Diagrams:-

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

Flow Diagram :-

1)User will login into website and search for job.

2)User fills form for applying job.

3)The list of users will be stored in container registry.

4)Data will be stored in IBM DB.

5)Chat Bot will display the results of job according to user skills.

6)For job recommendation it will fetch details from database.

7)Organization will provide vaccancy details.

8) Vacancy details stored in IBM Database

5.2 SOLUTION & TECHNICAL ARCHITECTURE :-

Date	19 September 2022
Team ID	PNT2022TMID27298
Project Name	Skill and Job Recommender
Maximum Marks	4 Marks

Solution Architecture:

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

1. Find the best tech solution to solve existing business problems.
2. Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
3. Define features, development phases, and solution requirements.
4. Provide specifications according to which the solution is defined, managed, and delivered.

5.3 USER STORIES:-

User Stories :-

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
User 1 (Fresher)	Registration	USN-1	As a user, I can register f by entering my email, password, and confirming my password, skills required for job, hobbies, languages known, experiences.	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	s a user, I can register r the application rough LinkedIn	I can register & access the dashboard with LinkedIn	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register & access the dashboard with Gmail.	Medium	Sprint-1
	Applying Job	USN-5	As a user, I can apply for the job using the input form which is registered during the registration process.	I can apply for job by matching with the form I registered and the job applying criteria given by organization	High	Sprint-1
	Changing Domain	USN-6	As a user, I can also change my domain which is different my course.	I can apply for job in different domains by filtering the domains option	High	Sprint-1
User 2 (Experience)	Applying Job	USN-1	As an experienced person, I can apply for job using the experience which I have been included in the form (resume).	I can apply for the higher positions by comparing thee experience I have with company job criteria	High	Sprint-1
	Selecting based on roles	USN-2	As an experienced person, I can apply for job using the experience which I have been included in the form (resume).	I can change the roles using the job filter drop down menu.	Medium	Sprint-2
	Selecting based on salary package	USN-3	As an experienced person, I can expect the desired salary package.	I can expect the desired salary package based on the experience or previous job salary package	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer Care Executive	Issuing offer letter	USN-1	As a customer executive, I can rectify the user issue if the offer letter is not issued or any technical issues.	Offer letter is not received on the correct date	High	Sprint-1
	Acknowledgement mail	USN-2	As a customer executive, if the customer issue has been rectified acknowledgement should be sent	Acknowledgement can sent using mail or message	High	Sprint-2
Administrator	Technical Issues	USN-1	As a admin, I have to solve any technical glitch happened like if the user data has been leaked or any malware has been there in server.	Issues can be rectified by scanning the entire system	High	Sprint-1

6. PROJECT PLANNING AND SCHEDULING :-

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-1		USN-2	As a user, I will receive a confirmation email once I have registered for the application.	1	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.

Sprint-1		USN-3	As a user, I can register for the application through Facebook.	2	Low	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-1		USN-4	As a user, I can register for the application through Gmail.	2	Medium	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-1	Login	USN-5	As a user, I can log into the application by entering my email & password.	1	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-1	Dashboard	USN-7	As a user, I can access the website in a second.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-1	Dashboard	USN-8	As a user, If I Log in correctly, I can view my dashboard and I can navigate to any pages which are already listed there.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar
						R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-2	User Profile	USN-9	As a user, I can view and update my details.	2	Medium	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-2	Database	USN-10	As a user, I can store my details and data in IBM Database.	2	Medium	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-2	Cloud Storage	USN-11	As a user, I can upload my photo, resume and much more in the website.	1	Medium	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.

Sprint-2	Chatbot	USN-12	As a user, I can ask the Chatbot about the latest job openings, which will help meand show the recent job openings based on my profile.		High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-2	Identity-Aware	USN-13	As a User, I can access my account by entering the correct login credentials and myuser credentials are only displayed to me.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-3	SendGrid service	USN-14	As a user, I can get a notification or mail about a job opening with the help of the SendGrid service.	1	Medium	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-3	Learning Resource	USN-15	As a user, I can learn the course and Iwill attain the skills which will be useful for developing my technical skills.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-3	Docker	USN-16	As a user, I can access the website in any device.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-3	Kubernetes	USN-17	As a user, I can access the website in any device.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**

Sprint-4	Integration testing	USN-16	As a user, I can access the website without any interruption.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.
Sprint-4	System testing	USN-17	As a user, I can access the website without any interruption.	2	High	Shyam Sundar S, Sham Kumar J, Nithish Kumar R, Ashish M, Pelleti Hima Teja Reddy.

Project Tracker, Velocity & Burndown Chart: (4 Marks) **Burndown Chart:**

7 CODING AND SOLUTION

7.1 FEATURE 1

```
<html lang="en">
 <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Skill Login</title>
    <style>
        *{
    margin: 0;
    padding: 0;

    box-sizing: border-box;
 }

 body{
      background-color: cornflowerblue;
    width: 100%;
    height: 100vh;
    display: flex;
    justify-content: center;
```

```
   align-items: center;
}
.container{
   width: 50%;
   height: 60vh;
   padding: 20px;
   border-radius: 20px;
   box-shadow:  0px 5px 25px rgba(0,0,0,0.2);
   display: flex;
   justify-content: space-evenly;
   align-items: center;
   flex-direction: column;
  background-color: rgb(211,211,211);
}

.container form{
   width: 100%;
   height: 100%;
   display: flex;
   justify-content: space-evenly;
   align-items: center;
   flex-direction: column;
}

.inputBox{
   width: 100%;
   margin: 10px 0;
   border-radius: 5px;
   overflow: hidden;
   height:50px;

}

.inputBox input[type="text"], .inputBox input[type="email"]{
   width: 100%;
   height: 50px;
   border-radius: 5px;
   border: none;
   outline: none;
   overflow: hidden;
   border: 1px solid rgba(0,0,0,0.2);
   padding: 0px 10px;
   font-size: 16px;
   color: #444;
}
```

```
.inputBox textarea{
   width: 100%;
   height: 120px;
   border-radius: 5px;
   border: none;
   outline: none;
   overflow: hidden;
   border: 1px solid rgba(0,0,0,0.2);
   padding: 0px 10px;
   font-size: 16px;
   color: #444;
}

.inputBox button{
   width: 100%;
   padding: 10px 20px;
   border: none;
   outline: none;
   background: rgb(0, 119, 255);
   color: #FFF;
   font-size: 20px;
   font-weight: bold;
   cursor: pointer;

}

.inputBox button:hover{
   background: rgb(0, 17, 255);
   transition: all 0.3s ease;
}

::placeholder{
   font-size: 16px;
}

.alert{
   width: 100%;
   background: rgb(0, 255, 106);
   padding: 10px 20px;
   border-radius: 5px;
   text-align: center;
   font-size: 18px;
   font-weight: 900;
```

```
    display: none;
 }
 h1{
 color:cornflowerblue;
 }
    </style>
 </head>
 <body>
    <div class="container">

      <b><h1>IBM</h1></b>
        <form action="/login" method="POST">

            <div class="inputBox">
                <input type="text" id="emailid" name="fn" placeholder="Enter
Your Name" />
            </div>
            <div class="inputBox">
                <input  class="inputBox" type="password" id="name"
name="password" placeholder="Enter Password...." />
            </div>



            <div class="inputBox">
                <button type="submit">Submit</button>
            </div>
            <center><h4>Return to <a
href="homepage.html">Homepage</a></h4></center>
        </form>
    </div>

   <script>
  window.watsonAssistantChatOptions = {
    integrationID: "a2445f85-b7a3-4c58-980c-72a414ea407b", // The ID of this
integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "1c2b79b4-a6d1-4c4d-bf2f-852aa415e243", // 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>
```

1. Above code is for the login feature
2. It is to ensure that the correct user user login into the portal.

```
3. from flask import Flask, render_template, request

   import ibm_db



   app = Flask(__name__)



   conn = ibm_db.connect(

       "DATABASE=bludb;HOSTNAME=1bbf73c5-d84a-4bb0-85b9-
   ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=32286;S
   ECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;
   UID=dtk93338;PWD=honkzW80jUXH3UJN;",

       "", "")

   num=[]

   num1=[]

   num2=[]



   @app.route('/')

   def index():
```

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


@app.route('/login.html')
def login1():
    return render_template('login.html')


@app.route('/register.html')
def register():
    return render_template('register.html')


@app.route('/apply.html')
def applyjob():
    return render_template('apply.html')


@app.route('/application.html')
def application():
    return render_template('application.html')


@app.route('/login',methods=["GET","POST"])
def login():
     i = int(0)
     if request.method=="POST":
```

```
        fn= request.form.get('fn')
        sql = "select * from userdetails"
        stmt = ibm_db.exec_immediate(conn, sql)
        while ibm_db.fetch_row(stmt) != False:
            em=ibm_db.result(stmt,0)
            num1.append(em)
            i=i+1

        for y in range(i):
            ele2=str(fn)
            if ele2==num1[y]:
                return render_template("apply.html")









@app.route('/register', methods=["GET", "POST"])
def insert():
    i=int(0)
    if request.method == "POST":
        fn = request.form.get('firstname')
        mn = request.form.get('middlename')
```

```
        ln = request.form.get('lastname')
        course = request.form.get('course')
        gender = request.form.get('gender')
        #skill = request.form.get('skillset')
        email = request.form.get('email')
        pw = request.form.get('password')
        sql = "select * from userdetails"
        stmt = ibm_db.exec_immediate(conn, sql)
        while ibm_db.fetch_row(stmt) != False:
           em=ibm_db.result(stmt,0)
           num.append(em)
           i=i+1
        for y in range(i):
           ele2=str(fn)
           if ele2==num[y]:
              print("already registered mail")
              return render_template("alertwrong.html")



        insert_sql = "Insert INTO userdetails Values(?,?,?,?,?,?,?)"
        prep_stmt = ibm_db.prepare(conn, insert_sql)
        ibm_db.bind_param(prep_stmt, 1, fn)
```

```
        ibm_db.bind_param(prep_stmt, 2, mn)
        ibm_db.bind_param(prep_stmt, 3, ln)
        ibm_db.bind_param(prep_stmt, 4, course)
        ibm_db.bind_param(prep_stmt, 5,  gender)
        ibm_db.bind_param(prep_stmt, 6, email)
        ibm_db.bind_param(prep_stmt, 7, pw)
        ibm_db.execute(prep_stmt)
        print("inserted")
        return render_template('register.html')


@app.route('/apply',methods=["GET","POST"])
def applyforjob():
    fn = request.form.get('fn')
    ln = request.form.get('ln')
    citizenship = request.form.get('Citizenship')
    city = request.form.get('city')
    twelve = request.form.get('twelth')
    tenth = request.form.get('tenth')
    insert_sql = "insert INTO newapply values(?,?,?,?,?,?)"
    prep_stmt = ibm_db.prepare(conn, insert_sql)
    ibm_db.bind_param(prep_stmt, 1, fn)
    ibm_db.bind_param(prep_stmt, 2, ln)
    ibm_db.bind_param(prep_stmt, 3, citizenship)
```

```
    ibm_db.bind_param(prep_stmt, 4, city)
    ibm_db.bind_param(prep_stmt, 5, twelve)
    ibm_db.bind_param(prep_stmt, 6, tenth)
    ibm_db.execute(prep_stmt)
    return render_template("apply.html")


if __name__ == '__main__':
    app.run(host='0.0.0.0')
```

The above code is implemented using python language

1. The above code contains all the features like login to ensure privacy
2. It also have the code for registration
3. It also connects to database which ibm database(db2)
4. It has the database of companies which contains job offers ,salary package.

8 TESTING

8.1TEST CASE

TEST MODULE	TEST CASE	EXPECTED RESULT	TEST RESULT
ADMIN	Provide validlogin credentials	User successfully logged in and directed to the admindashboard page	PASS
ADMIN	Enters invalid logincredentials	Displays Error message	PASS
ADMIN	Upon successful login, click on the 'List of Employers' tab.	Displays thedetails of list of active employers registered withthe application	PASS
ADMIN	Click on 'Active/Deactivate' tabunder status of the employer	The statusof the employer will be changed to active/deactivate.	PASS
EMPLOYER	Provide details forregistration	Employer successfully registeredwith theapplication	PASS
EMPLOYER	Upon successful login, click on 'Post New Job'tab	Employer posts jobs with the required details	PASS
EMPLOYER	Employer tryingto postjobwith insufficient details	Prompts to fill in all the necessarydetails of thejob	PASS
EMPLOYER	Employer clickson the'List Posted Jobs' tab	All the jobs posted by the employer willbe	PASS

		displayed.	

EMPLOYER	Employer clicks on ‘Active/deactivate’ under Status	The status of the job postingwillchanged to active/deactivated.	PASS
EMPLOYER	Employerclicks on the ‘view’ tab under candidates column	The list of the details of applicants for a particular jobposting are displayed.	PASS

JOB BSEEKER	Provide details forregistration	Jobseeker successfully registeredwiththe application	PASS
JOBSEEKER	Enters invalid logincredentials	Error message displayed	PASS
JOBSEEKER	Upon successful login, clickon ‘My Profile’ tab	List detailsof jobseeker	PASS
JOBSEEKER	Upon successful login, clickon ‘Search Jobs’tab	Details of the activejob postingsaredisplayed.	PASS
JOBSEEKER	Upon successful login, clickon ‘Applied Jobs’tab	Details of the jobsthat are appliedby the jobseeker are displayed	PASS
JOBSEEKER	Click on ‘Add Review’tab	Displays a form to fill in the reviewdetails of thecompany	PASS
JOBSEEKER	Logout	Redirects to the Home page of theapplication	

8.2 USER ACCEPTANCE TESTING

9 RESULTS

9.1 PERFORMANCE METRICS

Performance testing is performed to determine how well the system can perform in terms of responsiveness under all kinds of load. The web application is tested to see if it can sustain huge amount of requests providing higher throughput under different loads. I have simulated multiple hits on various pages of the application to evaluate the overall performance.

Operating System	Windows 10 (64 bit)
RAM	8 GB
Processor	Intel core i7
Processor Speed	3.40 GHz

10. ADVANTAGES AND DISADVANTAGES:-

ADVANTAGES:-

- Career growth
- Nervousness will be reduced
- Increase in speed of work
- Hardwork
- Confidence

DISADVANTAGES:-

- Not doing the task within deadline
- Not having the necessary skill
- Making mistakes in task
- Stage fear

11. CONCLUSION

- Content-Based Filtering and Collaborative Filtering of recommendations have been compared. Additionally, an aggregation plus recommender system has been devised. Content-Based Filtering recommends the results based on matching the personal preferences of the user with the given document whereas collaborative filtering recommends based on the preferences of fellow users.
- Along with this, testing and collecting more user data for better performance of the collaborative filtering module is required. Lastly, improving the cleansing process of the job description and using natural language processing are required. While using collaborative filtering, this work can be improved by giving different weights to different users based on their LinkedIn skills.

12.FUTURE SCOPE:-

- There is a lot of scope of enhancement to the existing recommendation system. An approach naturally solves the candidate and job cold-start problem in the absence of interaction data. The existing system has a job recommendation based on their skills and working field mentioned as a requirement.
- As part of the future work, we can make a good UI for this system with some features like the portal can send email notifications to candidates about certain job availabilities, there can be a feedback or review section for the application.

13.APPENDIX:-

13.1 SOURCE CODE:-

Login.html

```
<html lang="en">
 <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Skill Login</title>
    <style>
        *{
    margin: 0;
    padding: 0;

    box-sizing: border-box;
 }

 body{
      background-color: cornflowerblue;
    width: 100%;
    height: 100vh;
    display: flex;
    justify-content: center;
    align-items: center;
 }
 .container{
    width: 50%;
    height: 60vh;
    padding: 20px;
    border-radius: 20px;
    box-shadow:  0px 5px 25px rgba(0,0,0,0.2);
    display: flex;
    justify-content: space-evenly;
    align-items: center;
    flex-direction: column;
   background-color: rgb(211,211,211);
 }

 .container form{
```

```
   width: 100%;
   height: 100%;
   display: flex;
   justify-content: space-evenly;
   align-items: center;
   flex-direction: column;
}

.inputBox{
   width: 100%;
   margin: 10px 0;
   border-radius: 5px;
   overflow: hidden;
   height:50px;

}

.inputBox input[type="text"], .inputBox input[type="email"]{
   width: 100%;
   height: 50px;
   border-radius: 5px;
   border: none;
   outline: none;
   overflow: hidden;
   border: 1px solid rgba(0,0,0,0.2);
   padding: 0px 10px;
   font-size: 16px;
   color: #444;
}

.inputBox textarea{
   width: 100%;
   height: 120px;
   border-radius: 5px;
   border: none;
   outline: none;
   overflow: hidden;
   border: 1px solid rgba(0,0,0,0.2);
   padding: 0px 10px;
   font-size: 16px;
   color: #444;
}

.inputBox button{
```

```
   width: 100%;
   padding: 10px 20px;
   border: none;
   outline: none;
   background: rgb(0, 119, 255);
   color: #FFF;
   font-size: 20px;
   font-weight: bold;
   cursor: pointer;

}

.inputBox button:hover{
   background: rgb(0, 17, 255);
   transition: all 0.3s ease;
}

::placeholder{
   font-size: 16px;
}

.alert{
   width: 100%;
   background: rgb(0, 255, 106);
   padding: 10px 20px;
   border-radius: 5px;
   text-align: center;
   font-size: 18px;
   font-weight: 900;
   display: none;
}
h1{
color:cornflowerblue;
}
   </style>
</head>
<body>
   <div class="container">

     <b><h1>IBM</h1></b>
       <form action="/login" method="POST">

           <div class="inputBox">
               <input type="text" id="emailid" name="fn" placeholder="Enter
```

```
Your Name" />
            </div>
            <div class="inputBox">
                <input  class="inputBox" type="password" id="name"
name="password" placeholder="Enter Password...." />
            </div>



            <div class="inputBox">
                <button type="submit">Submit</button>
            </div>
            <center><h4>Return to <a
href="homepage.html">Homepage</a></h4></center>
        </form>
    </div>

   <script>
  window.watsonAssistantChatOptions = {
    integrationID: "a2445f85-b7a3-4c58-980c-72a414ea407b", // The ID of this
integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "1c2b79b4-a6d1-4c4d-bf2f-852aa415e243", // 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>
```

Register.html

```
<!DOCTYPE html>
 <html lang="en">
 <head>
    <meta charset="UTF-8">
    <title>Registration</title>
```

```
 <style>
    @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swa
p');
 *{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: "Poppins", sans-serif;
 }
 body{
  background:rgba(0,174,239,0.8);
  display:flex;

 }
 .wrapper{
  width: 500px;
  padding: 2rem 0 1rem 0;
  margin: 50px auto;
  background: #fff;
  border-radius: 10px;
  text-align: center;
  box-shadow: 0 20px 35px rgba(0, 0, 0, 0.1);
 }
 h1{
  font-size: 2rem;
  background: linear-gradient(to right,cornflowerblue,skyblue,blue);
            -webkit-background-clip: text;
            color: transparent;

 }
 p{
  margin-bottom: 1.7rem;
 }
 .inputs{
  width: 85%;
  outline: none;
  border: none;
  background: #dfe9f5;
  padding: 12px 14px;
  margin-bottom: 10px;
  border-radius: 10px;
 }

 button{
```

```
 font-size: 1.1rem;
 margin-top: 1rem;
 padding: 8px 0;
 border-radius: 5px;
 outline: none;
 border: none;
 width: 85%;
 background: cornflowerblue;
 color: #fff;
 cursor: pointer;
}
button:hover{
 background: rgba(100,149,237,0.767);
}
  .radio{
     align-items: center;
  }
  textarea{
     resize: none;
  }
  .email{
  padding-left:45px;
  }
</style>
 </head>
 <body>
  <div class="wrapper">
    <h1>Sign up</h1> <br>
    <form action="/register" method="POST">

 <label> Firstname </label>
 <input class="inputs" type="text" name="firstname" size="15"/> <br> <br>
 <label> Middlename: </label>
 <input class="inputs" type="text" name="middlename" size="15"/> <br> <br>
 <label> Lastname: </label>
 <input class="inputs" type="text" name="lastname" size="15"/> <br> <br>
      <label>
Course :
</label>
 <select>
 <option name="course" value="Course">Course</option>
 <option name="course" value="BCA">BCA</option>
 <option name="course" value="BBA">BBA</option>
 <option name="course" value="B.Tech">B.Tech</option>
 <option name="course" value="MBA">MBA</option>
```

```
 <option name="course" value="MCA">MCA</option>
 <option name="course" value="M.Tech">M.Tech</option>
 </select>
 <br>
 <br>
 <label>
Gender :
</label>
 <select>
 <option name="gender" value="Course">Male</option>
 <option name="gender" value="BCA">Female</option>
 <option name="gender" value="BBA">Others</option>
 </select>
 <br>
 <br>
      <label>Skillset</label><br>
      <textarea cols="50" rows="5" name="skillset"></textarea><br>

      <label> Password </label>
      <input type="password" name="pn"/><br><br>
      <div class="email">
      <label>Email</label>
      <input type="email" name="email">
      </div>
     <button>Submit</button>
    </form>
      </div>
 <script>
  window.watsonAssistantChatOptions = {
    integrationID: "a2445f85-b7a3-4c58-980c-72a414ea407b", // The ID of this
integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "1c2b79b4-a6d1-4c4d-bf2f-852aa415e243", // 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>
```

Company detai;ls.html

```
<html>
   <head><title>List of Jobs</title>
   <style>
      body {
  width: 100%;
  font-family: sans-serif;
  background-color: antiquewhite;
 }

 .projects-section {
  margin: 3em;
 }

 h2 {
  font-size: 2.4em;
  margin-bottom: 32px;
  font-family: sans-serif;
  color: maroon;
 }

 .card-container {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
  gap: 30px;
 }

 .card {
  display: flex;
  flex-direction: column;
  gap: 12px;
  max-width: 300px;
  overlow: hidden;
 }

 .card:focus-visible, .card:hover {
  scale: 1.02;
  transition: 400ms ease-in-out;
  outline-offset: 6px;
  border-radius: 6px;
 }
```

```
img {
 aspect-ratio: 1/1.2;
 border-radius: .4em;
}

h3 {
 font-size: 1.4rem;
 font-weight: bold;
 font-family: cursive;
}

p {
 font-size: 1rem;
 color: #888;
 line-height: 145%;
}

.link {
 display: inline-flex;
 width: fit-content;
 align-items: center;
}

.link::after {
 display:block;
 content: '';
 border-bottom: solid 3px #019fb6;
 transform: scaleX(0);
 transition: transform 250ms ease-in-out;
}

a {
 color: darkred;
 text-decoration: none;
 font-weight: 500;
}

.arrow {
 padding-left: 4px;
 color: darkred;
}

@media screen and (min-width: 320px) {
```

```
  html {
    font-size: calc(16px + 6 * ((100vw - 320px) / 680));
  }
}
@media screen and (min-width: 1000px) {
  html {
    font-size: 18px;
  }
}
      </style>
   </head>
   <body>
<div class="projects-section">
  <h2>List of Jobs</h2>
  <div class="card-container">

    <div tabindex=0 class="card">
      <img
```


src="
AVDxAVDw8VDxUVFRAPFRUWFhUXFRgYHSggGBolHRUVITEhJSktLi4uFx8zODMsNygtLisBCgoKDg0O
GhAQFy0dHR8tLS0tLS0tLSstLS0tLS0tLS0tLS0tLS0tLS0tLS0tListLS0tLS0tLS0tLS0tLf
/AABEIALcBEwMBIgACEQEDEQH/xAAcAAABBQEBAQAAAAAAAAAAAAACAAEDBQYEBwj/xABCEAACAQID
BAgDBQYFAwUAAAABAgADEQQSIQUGMVETMkFhcYGRoSJSsTNCYsHRFCNygpLCB1NjouEWQ7IVJHPw8f
/EABoBAAIDAQEAAAAAAAAAAAAAAAABAgQFAwb/xAA3EQACAQIDBQQIBQUBAAAAAAAAAQIDEQQhMQUS
UZGhYXGx0SIyM0FCgeHwBhRiwdIVNIKSokP/2gAMAwEAAhEDEQA/AOeCYdorTdRlkcYyS0EiMADAMl
IgEQAEwTJCIxEYiMwTJCIJEYiIxoZEa0YAmNCjWgIGCYZEG0QgDGMkIgkRjZGYJEkIgWjsFwDBIkhE
EiIZHGhkRiIAARBMkIjWgBGY9odogpPCIZHBMmWkxNgCT6zpp7NY9YhfDU+0r18zo7fE+q1zi4O3r1
L2s1HuzfkubBKens6NcV6HvljhCjv4+gnTT+3JD8XDN8z52A+eSMZR1jvYXRx+pIdaNgZNTw1MHhYN
ZVFfCp6GBj4fq1XSGgwLNF/Oi5uKFQRrfrnrW1G96gx/3ED6vD5Wl4nR2Wkp+Z55fCeE39QGxwLG2g/
kb1Ra7XtNmo6V6bkm5+Rk4JXWvH7KPG+6JgZlp0NXt2FwmqJkOmpAZMG7UvRfMOgOWD87/a/zxpdU2/
WT5yq+5NBuBqsWKJpnrkD3GH1uTRzVn5jmMN8yidhpNnbv8S5vWBtdLoiJbKuUwvlZuRNRjbsF3ZlZH
Q4ItRrzm3a8jmPwwrwLYJcbvBG7hdAWSCGh6n1T72D5wvBGhQibVVtUMDd38AaprkF1Ja8oP26hX3i7
r8DLAtTc6tbXnVN2Wxs4S7NUDvC/g6OuS/kfZVwnfXPwZqxaOp03rMVqUMA+XhP7HINCHbUOtgitVMm
JY3p5QL+I7HIEKmJPpoqlPqBs5kmPrbzqx/sMQzpkBL/zhZPgTlKA6ne3/62oh99MbTyXGF6T5nJU+t
2q8dKFX8ed69DFS38dswAYt7cZHA3Nx1Vck53dUnWpbEHzt6QMq7qwzbDGLdFKCfljK8D2iQwrgpYbw
MxjiIWAjLlOpi3eJcqyPrigTAAhlRSmBHAg1fjL+xSQgG1i4znRuD/rVNVHmc+Y3Z9cQGbsLnW6fw/tJ
hdFJctSL0hcdnImRAJKB+iGu5C5DqBJ6DfAmkoMwKoq1mPXcu8VHLmcwidqddLRvK54Cxd2tMN5wyGvJ
kYi1h9WkNVK7rvWfxdB0OCFUbdkBcyNDVVN5xtxmCu0MxpF9xkW/56UuhyeAlFCukhpVe3w8i0rQVn2C
9yHSVJKtXDZ8dQC1IxygbgOtfFx1ZRMJmYHBrKh/nrHUSQCjc3VC6/a9HEQtxDDkhSmX7sWy5+ixZ8uK
6ZPTnadw6zG3P4H5yqbSMpx2n8RiIf59Yji1gRn6DeWeUKgPFhcmfNLk2/cLDLNMtxgd4FvsUVgT/MM6
J6Ze/Exbjr8YSQFs8NLgLoodrsqokXEnqJt1WvP5FhHEPKYLvCOQNAvQUt7iNJbSztDjUojjxqGuXe
RIQn4X6yWXATTi2VuCSLmV3qUM3VMpW0sCoiztQT/mtP2nU1cx2OEtg9vgqBSgcZYrW0KiJTr7+XNnWb
1mtuXfwoSC16/C8sJxm/BOalEfT8hcgBeqyJ/UhaKqRw/lzOLwtJ6Yp8NWUNldSd0u6Pc/0a0+uU7N
R6JbEqSfpt+7lJ4eOkcfDo9U/DUHtEmSC/Vfvzi3EiDgdyuiqNQFdOvzB9CH9ucQDhR+S4kf7gcUGtYM
0FUbDp/JH+K/iG0I/HvnpD2MkIHJvRnR2qHRCF1vMLRjBl/a5Cp/W0iIUq5fsuKd7JhMQ+y08gbJQXBO
2HG5DF5kfCJt6eqBvHytGE18vrYOVTQWYvtUPVtmWOYdSyLeUKhRw8kKdhL5/RgYwzlphdXM8QI/Bbr
7lENSSoCOW3bJMgvxp6DI5EfBRVaXcBlFiBq2XNBVRAJRW36jNFyEb2jfvCJoxsgkcXMKHnqHNGGsvA
+VQSqTrz/cUtR0mnKrQVcjnB5LQS0ElMqTqFsyFjwZzK3RsIRZ8RJ/C6OZ5BEq2irbHjNYfhGsR5X2EtEEsB
4oOjROq+Cmi9BpSJ3lSKhEgJkOoqVu6sHgALLd+Bwrp/VbIeXVSbPmQqeZSZoHhRrBZiwzJO5bgLW3c
5cQFDhv3/Okjom/Z8aKpsyvc5rRyNu4ySvPrRNVK0gKfjTmHKgQg4ZjMnf+6/Y4vdg0uFGKfd9DGQC3+
5aggeTnafnrOtLbmkTFmfZLlZdvnnO94pDHOKHgcfXRuNNJoMinxjeU7sr+gEeTIJb2rPoHyQt2lw2fX
6+Rxnb9pG9zmnUJ07DxrlNqhUZ0GiJr5j7o5nb9rpi/fXxpodF+EvCqmvWQRxnV1aVXbimcLHH3zEe7
WXH3CV4j+3v2aQuUqPUWB4M3iPK5iBUZ8lqEcEZxWsxzw0zb7BkDutkpPYvXcVQDuBmD1ToAPIaDh/C
4NDvUk5HlU+XkdtgaRRCc5bl/CvfdRZcV9jzSRTcZK0gXNvNRNyjG3HZ7ydk+V0ixjnGJuIbSEadbVy
8VU44MVqvG/b75M/gtZoqXFrpLsmHTVlqWNE6mkWIugNCcRRpROhFGgYdtpu51sGefsgvVaw16X3eU
VA84atyOyfgWuqOKm3Mg3ndtwJ1D7H3qCdGc+dfsOmW2JdlL80rXr7z2tKHugH2dqfXzlojntjdffWoWj
iodkFGP6ZvkRHFcGd1o1YGAg3qDu5m08i7ZgDC1lGMG+k8Pd2PFhkS6fgG7BWgjXOYZfW7g86rmtsP8
+yIQo7/nl93kZvwoAJg1+t4ujjPRxWdJXY2G2jPy02eLs9tGlTXr5+i3Rs3AFSsjnzOybcqOAWR6TKn
cbdJUAoXFnNfDTHRBL5wvPO76/3Njlh7hNFwf2zuzE5yqsjD2Wn2WzWnI9bgEdqcTjh6Ml7+4Pn3cgB
bX1PjPYpZkPJeDnZqGndPTfA+tOM4ARl5+Dc7XdsRMc+UKzAJ+R/C+HSXR3OxUrjQd58aDpQcB+I6cHc
ktWW7/Un15a9ULdURyZeB3OP8APS4eADNemWE1iY/o2yCrJEKedjK7E4W96d3wlvJHnyXZ2GSabfRY
UnBP0u75R6p2TMxyLr1lCGTdLq7YNnMsxZrcDOvzh9ESFybCRxEO8wMS3u/oyI+vw/6Anf+SnrtOyYd
gCNYBgkdRZq71ugLnw+8eL1R7jcVHoajgegOCQbGhcU1ZXNTMkfnJ3H5Y5rXY2jTYNsB/YvO8RyvuxNCSY
uK04v9r5J9x7SYT0Gc2+Q5hkdAPsPwUaZYVGN24VE8GV+OTqpvaPiX7enlPQWl9T2w8ddshsD0GY11L5RHd
j2gUkUbhBm/7M6/8ICNPUu6fhTwx5uGSA3JgtvA2G5PAjNR0D/KJNzuuDhKcvuuX52fFZcz2kdkSK5GqQ
YJmODcsq4T3dlB1HfQkdSyrq1aTThERl0y5zDlr7Owz2ZyobD2LmVuxlYOU4L2UaHCrgFeb59FW6tPPD
ax5tW9BU+Ixpv28fOWY1+1z/Xjgns/3iHKK+7xxQYXbBm7hHmmB1MOeWGGlqwDivq4X6N2uVwYfHGl9
YmxiJMcR1AYEMjp4hM2ZBrt2YH09ewFGdQ31oIL5p7NYyyAIyTOBXbDnLZaLiuG5FEUN9crMGF/5E8CR
uTrufcedhHeUGVUPTG7/vmKC+P1fqtTKsHZb5lx3a/2lVoeKmg4GpGCuXJ5p7ySg3fZXKC2KeMyJYVpbmc
QG6UVpIG53GVUmpD4mkcS4qTk6zpLXE3qeuqXVUe+OKCY7VnzbYqD6mPNqH1KEVkvkVxcX21lqtrUAp
kp6+rbxVAmadE5Z5Nwer6ZBvYGDnwCNMWt76F1pZDRKwyprQ5DUqqhsQZI+nZdzxPlXJkqqQSrT09G
CzfBnUGS9WjqqJVkwfMz7mm1R5Bpt5Q1lFiuEYTEfKmd6bQvjl3GaCDQ9yrNdamOxZxpeLYWoZWuVHfV+
0ZA+rB9UxmB/2FD1/oD4RqqAb1dkdBpsPnRQ6ehbnAdvkvN/8tQ+KsVrBIfVU8NmaEBjzTv87yyBgrKnV
02yAP2bvfZHnCRabF4RnqxcmeWtH9qvtHdJ/+nd/P1+4whFOaw2FfzaxUIbVIp8tyaN/eRUg63lX4
eq5hp9aqHUbVBFx9+Fsr3XJFyTadXNK81EYoGdYfAGQfVkS2Tw8aOiwH+YmvDSn8uLYJgsaO51XpqB
QjXTYRK7pgFyT6h1J4vQSyPSqNHwPdYU/Vt/BN/hqg5Tk8kQvB4p4e1zbNDx75VMYLR9kYM/5UWwR
8pIxTSlkHHBgaTkWUiqSaKlBtwc55dxr14jkP2lBn1dlxL2hVT8VhiMXxMB9lBzETsfDkM0H1wE7cOdp8
yeB69VCqA6YaVHRtwjIcLWRkVAfZu2rwtNcNTi81EvVZ8AfeCoQtoeLT3HdQOdOSi9mYOoNS5B2Zx0
MvvZaqt+1obpbdh5SQERzpJ4uRp8vd3RCZ/ZvbNbiwBiRxlNSXhIbsDdHcmxl2A2H8C23iTthX4vnb8h
xBlD5n7oExpOXHgR/tH/4f6c8I+Ld9Jqe9Tj1Oo4hhsQIO62j49g+Mm1a7RvxbfJhW9Bf+7ccwQCw0f
7N58v3oyUOyVWNqzJxjy4ID6qsdffkzN1ioTCx3OaZNGymEjkQn3n0lEeKR3j2l8VmHsrFUU1RF0UWuD
0VHiSVXQaPofjhNyLi+RjN39rMpQupkQhjU0bS2z3Q4MV6bX1lkx1tm8EjA0zDf77sBvn2V/EKbjsVvjL
bNLZb4zvEgQm94G/EKfIHprM5qy3/JDa8yEaEnpyiIBR7A0OXJtK7f/84svoVF/lpp3c4m4TKa/vIb7S
7Ys7y/pnw4+8BvPrzsXDWWhgrS3JMu/J5kWOj2Kdz5wfswTMODFq6C/cvHqbfQAvW+P/SW+QQ3SK
tF2CEPFE4Pe4rVQnoyW+F2G1Iu5x+t4DE4w5Zq4DM6PQMZ7XeYPp7sWnGEnQ2zvSUwDFuXXOlEyvqCid6f
ACh6CWefBp5JlNchR33S+NsJDNGm2tO7U0Dtec8ML/7ihKUT9y+eddMn4HKlv6Nm5p9dY1kdkH9ewqLGp
ndnxQCdtjTmQjXsySEFsdm/Zw3tLC+iZL/8Ag5Ltp+4QRftU/WT5Dst7gNKYK6z/ALtfdNFQ+4+2X
u0flF3kcy0+GgSubvbXgjPLzHBi+XEyUMKZ/s29VxT9fWOIbnoXpq8v0EW6r3qnGXF6vq2QQEVGD2+vVZ
5cc7eGd+/EXu+GdKzdwjeFfnG9/u2tzj7Yux1IuF0T/CLCtBK0JS7Fun9FxxJX4h6B2JpTJ53kqH9Xnb
RJA9ZFa0ZGMHaQ2QHIOwHJKhXo7/+/p0hQhxPTG23tJk6YRSOsYwx7+mbmcKdsz8v8vSuJ39fCa6qOm5
xuvxOkxmRY6C8ScL5RRfOJHrmQ9sXqk8N5sXPJhFAqvqe6eBb4yxcGpjzNWv3vbx0YZfdUq0Nqht/SvCc
dkCm4HI3dGBcRZuYk8KVX/4c+u0t70knMYoqSoRe4i7LHqX4H3xCndJ4xgA+RNYrQz0Fn6uRnfCqVPVb
4ZHZ7GYpfXEVmWqJ7/ejOsOq+vQWzsyrs7oOZSw2fxSgOsT5HTZmvO6w7DM25mjzkKL+eRAEhdtdaO3X
2yoe5WuxqjJtZgDU47cKZPbVFGyDmSAEjnx6u8zEp0GYHoWTdd9a/DLGBLRWnGxuG+DSb8ZVCEdCuE7b
k+TNeoxFZ6C1A3g6GkVdLrj7L/pVHeuvMBP+jAyDyFmpk3GDJlSZ/6wLX3t8Ls3jAW9TDkmCzCWeLUbnFg
kNmVlF5vc+bSCw4hGT/N8h5tC9+s5Q8C5iZ7NadZGEUzkBKGqEo+LjO/93KEw/GGuHjqv6gXXtMIaCHX
4yCzF6tXwVStJ+h/CB6aDqqZ6DI+uWrsnm6tiTyYt0kHB9nVpRTFJlkXBH8ssqJHjSLbJiBDqWqmp+M+D
4c/J1vMYBWTRfaZgh6fVtGrmCu4Ti3hWjwqZB3vsbnXWeLTcHTEWqf7C8pzYNhLgbMyQFoPqDXmNGOa
6MjOfE0cyN5M7k4Xr2xSb47/gv6WuCY5eqj5x+ZI6NpZrV6D7dqLo7ogXV7Fjq8lc5I33kYHfCZomz
9AlxI/17fvyCBktoTDGc0kGBCDVsIDSkllV+YtPb3uNx7hP1IUcIjO4vJ4R2z38DqAEfkglu2q7O75+3tv4X
JvqxHuUrSI10hwHVsYvnOk4a5Wa14Ca3wz3ioKZuy/CPq7wTXLxa5jlxjv7dOfD9GpmOZDZXXnsbE7Dtx
vQbQ7RWm5cyyO0REktGtGmBHaMRJLRWjAhtGIk0Ik0RIhtZiiLKBWMREVgkSYIEzg8jzEUQB3H1NTN
HwjIBLERQ4q8PdB0y8wNC+uV5PrEzNh1HrQ1y5iCUdF1XfXwOKR5XLuMhESNDatLLD3DcdRNtpq6EI7
glQHLp26gvdhRzaRQjVMw1XjtVOMxU9nwWNW0T0b7Bq9lHfyXowJiKp7H2RMvY7DBSzdsVJWSRkvTAdcf
5RuUwe/pQDZdw7DcJrVvTiH/5BmuuwW0dv1/6uVhMQtuzHbFGcVZlVkjMKbtHwu4JBrFNHkt66Ex4fTpx
N/ThyHK2Nf8hNqqR8kHCnvOcM3Whg1QKwzj3dg8Pn4bumUY9hcS0pO47Hx97VXgGyA0f6+W5Szd9P2N
dEOxIDE4R4sZU4yT+SXS4B6aZXxFaCEXWt9NMnnrdkzmrrfkq6Vmvs2nJZpRZVT7OfUnXVuXFWMvnmmrmL
FEbdwQa9y5DxzHmqALMaxTQNHCZ4Zt2bOEVMQZtLnGGebzuSWMVb8LCvTbf28rZZxNPLvmNGL9PnQOm8ZZLn
4fTGLyOcE8vMBrNmw6QgX5Ak3MCkPO3pDFhmDWi0hXUjS4QZFE7ZIGKZJxVo3nb/X0hnKH/ADg2Hm/Db
6pBNUHZhT7lkQtHqUwKuoXoMg5VwXh0RkZRG8PClyI4UCfbVnthKtPsyRgnDdVc/5sDFaxtsErNGN1
jWf7i9s8lyEmqO1GtQE9Sk4bRaRzvR0BGZFpSHM8a+66kT/hGIQzwSOmgwJBRKQaAbXsJPtDUCnVQ9
YPa1y7jSKxmVtaZCbSbQdwSjTrOcbeX1QRiJo7lE3x9L0+9mTRvq4d9JQmczDVsQbu13Zcg7AfrEeVMK3
Oa4ExHZ7b9GUSRN4ha3XhB0V7mYMLpTl5MwYQpBaSuPXfQKQyCoqWAdD6vc3snqFeKT8RfBjxoZE0J6
qbRvt9iTBvrWyHNprOPfwVRw+Th4HV/pRqqqxEKVuV6yJRGdpqpkMt04HMLhSmAZXcvKu4PBZ9hs9ln
2aDG+bRqEcnwdoGTFjgpeWazXSh7k/pWuqKVddvDZIy46Wx43bzhkb/mPONZDEKBJzt/6IWUfqHyGQ5O
gtZR9jUkcdDXxRBy+A+hVTpq1Xw1LfuZabHmhnhT+P9YAL9Wxe1JeNjFh6Z2lzHNmGhUDJcqB1eJC0
pG1U4lK4Um7bx2VtuoF/RnkyOT4vOvT1Y2e2SgqlqlyFQO9n6YtqQaG6u2mfAHF1OI6iTbF0Hf0L3J4
T9YlE0qdBeSHvEKXSdSw/ByKV4xQXrAFIW5fLGWDaVDKTdS/2eVK8r8uPuEZfRxeVQsc+QHrRYj76VEN
M5lZ8W9Lz5Dc+XH6HNlN9a8XJc9lFmYNvTT/nuK5nOdzeZjE9lWWfdjQK5d3eWR5mnP8t52VZ3BlVJv

PFgPeTLhFHf4zKxG3cFR+Pff6c+uUepoUdkYqprHdX6sumvQq1QngCfKTJgmPWIEtKVF20pqT3Kt7eN
uEsKGwq7dbKg7zc+i/rMue3cZX/taNlxefV7sV1NGOycLR9vVu+Cy6K78CgTBoOIJ7+EnVbaAAeHbN
RQ3epDrsz918o9tfeWOHwtOn9mir4AAnxPEylUwWNxX9zXy4e7kt2K77FiGLwuH9hS+enV3bMlQ2Xi
H4U2A5t8I9Dr7Syw+7Z/7tQfwoP7m/SaKNLFLZOGp6x3u/yyXQ41No156Pd7vPU4MPsfDpqKYJ+Zvi
N+6+g8p3WjxTQjFRVoqy7MilKTk7yd2DFHjRkRoxhRjABoJhSn3j3lwmz0D4qplJvkpKM1Spb5V/M2
HfAC1jTyzF/wCMyBv3OBdl5vXCk+Sqbes0W6X+ImDx7iiQ1GuerTcgrUPJHHE9xA7rx2A15EExNUUa
Ei/K8RiAYwTHMUQAEQTJDAMBgxR4oAZS0VpJaNabZmAWiyyQLfTt5Trp7Nqn7uUc209uMjOpGCvNpd
+Q4QlN2irlfaNaXKbKUdZie5dPcyRMOgNkT4uQBZvTUiZ1Xa1GOUE5vs+/BMvU9m1ZZyaiu37/AHKZ
MOzcB5yQYL5iPAS+GzaznqhRzZradwFz62nTS2In33Zu4fCPzPuJSqY7H1cqcVTXbm+v8eZbhhsHTz
nJzfT7+Zm+gReFvEzqo4Gq/VptbmRlHq1r+U01DCU01RFB+a129TrJ5Tls91nvYiq5v7436WLCxypq
1Gmor74WKChsBj9o6juUXPqbfSWFDZFBfuZjzc5vbh7TvilulhKFN3jBX46vm/2K1TFVZ5Sk/DwGVQ
BYCw7AOAjxo8snAaKKPEA0ec+MxtGiL1qtKmOb1FT6mLZ+OoYhOkoVadRLkZ0YMLjiNOB7oAdEUqN6
d4qGz6BrVrkk5aVIH4qr24DkOZ7J4dvBvjj8cx6SsyUze1CkxRAORtq/83tJKLYNn0KGBNha44jlHn
ysjFGDISjg3V1JVlPcRqJ6Fun/AIn4imOgxhFXh0ddtGHc5HW8Trzg4MVz2YzxnfH/ABMx6YqvhsN0
VJKVapSD9HnqEoxUtdiV1INtJuaG+JI+KkD4Pb6ieGb0Vc+OxLgWDYvEMByzVGP5xqHEW8jQbu7/AO
NpYsVsViq1SiQempNYhlAJAReCte2ot6TMbb2tWxld8RXa7ufJF+6q8lA0nEZZbN2Oaq9I5KoerbrN
368BG7LMkVBh03KkMpIIIIIOoI4ETo2nhOifKCSpF1J7ROOID0LYu2DWpKzm76hzzYdvnpLqjiGGqs
R4Ej6TCbtP8LD8QPqP+JrsI1wJ1hJtHOSSNru7t1y4pVmzBtEc8Q3YCe0GaqeX0WIII4ggjxE9MoVM
6K3zKp9RecqsUsyUHcMwDDMYzkSAiiigM4KewKv3yiDvNz6L+ZnTS2TRXjmc/wBI9Br7zRjBLxdi3c
NB+Z95KuVeqoHfbX1kp1sRP4t1dn3cI06EPh3u8pqOCcD93TCjnYLceJ4yRdmt95x4AE/W1veWVSpe
Rkyv+Wg3eV5PtOzxE7Wjkuw5lwFIcQW/iP1AsD6SZVAFlAA5AWHpHJjEzvGKjklY4tuWbdxiYBMKAZ
IQ0KZ3aG+WBoVOjZ2ZgbMUTMFPee3yvLfZ20aOITpKFRXS9iR2HkwOoOvAx2Yr3Ouc+Mx1GiM1arTp
rzd1Ue5me333o/YkFOlY4hxdb6iknDMR2nkO48tfIcdi6lVy9Z2dzxdjcn17O6SjC4nKx73gNqYavc
UK9GoRxCVFYjxAM7bT5uSsyMHRmVwbqykgg9xHCbjYG/GNqUgz1bspKsCi6lTa/DtFoShZgmSbxf4m
YlK1WjhqNFRTq1KfSPmdmyMVJABAGoPOZDaO+O0q98+LqgfLTIpD/ZY+s6v+msTia5ZWoqtWq7dIzm
wzsTqAt+JnNvbsBcBUWj0wqVSmapZMopg6KNTqTYn05yWSdhXdrmfqOS2YkljxYm5J7zxmz/w53hqY
WjXWmE+KuDmYE6hANNZiaxOgXVmIVBzJ0l6lHoMPkB1Cm55sdCfUxSd2kNEO9W3a2Or9JVfMFBWkLA
BUvrYDmdfTlOfYGyHxT1MtQU6dFA1WqVzWZjZEVb6sxv5AnslfUNppt2quXB5RxqYmpUbvCgIg8v3n
9UU3ZWQRKjbmzqdHL0ZYjg2YgnNz0A4/lKVxNdt2j/7Yv29KmvdqPzmUaNX94jYbv7QLUEN/iQlG14
5eF/IiZTbBviKp51qh9WMtt130qLyZD/UCP7ZV7ZW1eoP9RvfWCYPU4GHZNjlsqovAAKPLSZA9h7x9
Zt8PR6pJFyeA7PGJrMlfIot7aWXoh3VP7ZnZrN+EsKJ76v8AZ+kzC0iY3qEdC53WF+k/k/umtwI0mV
3VFmqDuT6tNds4aeZhAjI7aaz0PYjXw9I/6YHpp+UwSTc7ttfDJ3Fx/uMKugR1LGCYZgGcSYMUeKAF
8XgFpHeNeSAcmcO0NrUKH2rgG18oBLW8Bwgbax/QUS46x+FB+I/pqZ5pXx3TVnS7M6JmrOWFl+VSWO
rH5Rc2EklxItnpmz9tYauctKqpaxOQ3VrDibHUjh6yu3k3gOGZURFZmXMSxNgL2Gg48DPJdv4hqZoO
pIb9qQD+FgwYeYlquILIpOunae8xqKvYTeRbbR3mxdZFSoaacS3RFgGU6oGub3AtccL35SsG28VSz5
MQ+RqRQ0yS12Yg5lufgIAYXHG48RU4CtUqr07Eim71Bh0C6FKZALlu25zCw4WnBt4VKgSjTDFqtTLZ
eJQC7W/+8401u3DO4dXmOHOBsnaDU69SnmYIyU2ZbmxcEgG3C9jAwy2QACwHAd0j2dgTUbE1r2FJsM
lrdY1A549lsvvHKWSYorOx17XxN3cli2W4B1N1XQWB115d8ssHscYbC0a9XXFYkO7gqb06J0RNT1SC
Da3Ht0jYbaAoKdLsQLDh5mcW2NtVcQytWdbgBKaiygAdg7SfWKUXK2eQJpFTU46Tu2DQZcOj3+Go9Y
j+V8p+k4aksNg4ovhaFO1sgqi/zFqrNc8uweUJvNDjoanCYhaeHNRzooY//kw21MdUxFZ69U3d2LN3
cgO4CwHhLfbuOui0F4dZ/wC0fn6Sjo4R69VMPTNnqvlB7EQau57lUE+Um2kriSzLfdXA2R8Y462ajh
PDhWqDyIQH8Tcoe2VtRY24lFB5G+b+2WuNq0xlpURalTVadEduRe0/iJux72M5d5QBhrD7rqWPfYj8
5ygveycnZWRjnGkv92KZbDqB2tU8viMyWJxhOi6Dtm33MCnCA5lGVqgNyON7/QxSkmwSJd5UVcFYf5
iC/PUzEETYb01x+zIna1W4H4VDX92EyJH5+g1M6ES33WQ/vj2XpDzs0rNr61qh/GZpN38KaeFVze9V
nqAW+4CUH/gT5zN7S+1f+NvrIRzJM4KvD0m2w/3fEflMVWGhm1oDqHwj+IXuObfEXSl/FU+izLVGM1O

+nUpfxv8AQfpMpn7I5IIvIt91G/eVP4B9f+ZrMAdD/FMhuwbV2HOkfZlmv2f2+MUQkWKNNvum98P4V
G+gMxCCbHc8/unH+p9VH6SVTQUdS/gGOYxlc6AxRRQAtrxXjmDJiK7bOy1xKqGdlCknS1ted/CYnGY
TD4Si+Gw9ZqhqYl6+IqGxzMQAFBGlhYcOUtN+dr5b0Q1kVc1Y89LgHuA185kaFCocKmMrMU6Y3wuGA
F+g/wAyqx1uewCwse3skksrkW9bFexptjcKlXL0fSVqjZrW/d02IvfvtHxlTJQJ+Wkx8wpMq8XmqYx
Qqk9Hh3drC+UFgpJ5CTbx1CtCp3oR5HT85K/pNkXokWzUgmEwSCwAwFNyPxVWZzOPZrX2jhz2JSxlR
vKi1ved+8hCVKdK/Uo4WkP5aVz9ZWYBFOMqBlBybNqsLi9naoqAjkdRIf8AmS+MiqIBoOwSfYNhgsQ
54NtJVPeKVIk/+XvAq0ySbAxYEZdmLbjUx2LqH+Xo0kpfChR95xbVxFlerYDiQo4Dko7uAl4MImCww
SwOMqUM+LrkXZM65loofuKARe3H2me2nRNSgyL1iBbvIINvad+0NomqXZgc1RswF+ot72NuJ4DyMc0
3JL3Ci7JlRXFlP8J+k79m4c0aFIkqc1AVNDwUk6a9unvOcrfsnPs+sWptTuSqNlRvwXuB5QlqhxeTC
quWJJ4k3MHD4hqTM9NirtTKM1gT0ZIJAvwvbs/OBiHC6fePBRqT4AawauFrOpAWxI0JYD84SlFahFN
6HNS3irJUuwVwCdCLG3iJrqGLoYzCVb5lY0X6JbXviFsygkdhsNe+YHE7OrU9WQ2+biPUcJo91gei/
nYjv4Tmpk3Ex5l/u5tpMOr06mbKTmUgXOa1j6i3pOjb2wHua1BSysbsgHxKe0gdombsQbEa9oIkdB6
lxjMe2JqZ9QigrTXkL39THwOBqYismHpdeo1s3YicXdvwhbk9wg0aByjo6bEnusL87nslxsnBdDdib
1GFmI4BflHdJOWVhWNHtZqd1p0PskRKVHmaaAKCe82v5zCbWW1aoOVRvrN7hMKfhZuNtBOR9zGxOIul
T7Rrlcmo5634eUnFWRFvM8+rcDNvRX4E8ppn/wAJMOFObEYhjY9VUFyOwaGUaYV7KMtrHgey0ipKTy
JNWRU77D9zTP8Aqn3U/pMdeerthVdctRFZTxVgCPeXO52x8OKzWoUtKX+Wp1zDmJOTsrkYv3Hku7d+
n4H7NuzsuJsMCdT5T0Lefd9sRSVaORSr5rWC5hYi2gmU/wCnMRSOtKoe8DMP9t5CDuORHSM125rfDU
H4kPsf0lBR2NiTwo1PMW+s027Oz6tEOagtmy2FwTpfl4zpUa3SK1LyCY8AyuTFFFFAZa3iBkDV7G0lE
lcR5RvgxrDFBRd26UIL27SAPQWg4/F51pIFstPD0Kapxy5KYGXTlrNDtnYOIFZzTpF0ZmZWUrcZjex
BN76zC7f2VjRVJrLUpgKOhpK2awJDZmZbjMSBoOFpLeV0yO67WKnDbTKYyoLfbUzRYdqhaiPc/wBJH
nId4K/S1ehJIu1JCLE9Z1N7Dtljs3dnFsTWqKxqtfKO02GgANrACWmxdwnxFRKlSs+ZXRntb4VQ6An
nFfUlY4N9dpKcZnU3U4h7W7VVVQWi2cy9NiWB/wCxhqYP8VXpCP8AZO/aG7hrYg/tFNw3S1Gt8QJV2
vpbjwGs1C7i0/2dThlyVWZXrZixaoFDBAMxstg3dFnZBle5jTVy3N+F5DgQ/wCwYVWy2KV3UAEEB6z
8dderPQdibodG6mpSVhnUszqhsB2ASz21ulTr1BUWwsgUU7lVCgk6W4ceElJ5oilkzy5cDoCbm/VAU
6+HOVW1MNUZujCNTAt0ha4a3Gyg666az3LY+wzQZTdcqghVBJ4i3b4mFtnd/DYo5qtMZwLCoLZrcjz
HjFKb4jjFHiD4SvXtSoqBmNnqX6qdvnO3/wBEdLUEDAWsrAeOtyLXnqOG3SRNFqALyFOx+ssq2zejp
dHQS+Y/vCxFzbhxnOrU3IuVnJr3LVnWnHelu5K/HQ80w+6wSmFpAZjYvVY61D3njOmju0FF2ZC3nab
pNiVG1d0XuALfpKfGZFcojFgvFsthm5DXXxmBiMXj4p1HBRS7m/HwSNnD4fCSahvOT+aKzYm76Yh3Q
1VU08udFUlsrcCL6WPnNHT3OwYFirnvzW+gE8/xmLqDFVejZlAWkrWYi/w37OPbOzAbSrqbrWqA/wA
ZsfI6GbOEqSrUoylk2kZmLpRpVZRjojV4zctONCoR+F9RfxGo9DLLYmxxSohKq02fMxvYNa/DUjukG
7O16lcslSxKqCGAsTrbW2kv7zvK+jKq4mc2ruXhsQS4zU3PEpbKTzK/paU53Gemf3eR++9j6HQes3g
ivBOw3mZDC7qVjbOUUeOY+QH6zR7O2XToLZB8R6znif0HdO28RMbk3qJRSAKzmxOBpVPtKaN3lRf14
zqMaRGV6bGww4UU8xf6zopYdE0RFXnlUC/pJzBMd2ABEEiGYJgAxgxzFAATGMcyMmACvFBvFACYVQT
czqWvKwGSU3iAtle8hxWHSoLOoI7OY8D2R6TAiKoRaAGf2vgeidK1BsrqCFuSUdTq1OoOR4g8QfMGC
pvEqOP3QAYXqLcZw40Pcwtax+nCXOIphwVPA+3fMftPCNTfJUFxxU9luYPZMuvWqYWpv23qcteKfZ3
6q/PjqYejDFQ3L7s1p2r6fXu2eHrq6h0IKngROuk0wWzMW9A3RtD1kbVW9OB75rsBtKm44hT8pI9uc
t0sZQq+rNdzyfLyKtbCVqXrRfes10/exbWitI6ZvwIPnHY2lrMqhwHkT4qmOLr63PtOSvtemvVXN7D
9ZXq4qjS9eaXzz5a9DvTw9Wp6sW/vtLBVMhxWNp0usbt8g1PnylFiNsVX0vlHJdPfjKPaG1lQ5EGer
8gOid7n7o7uMyqm19+W5h4bz4vy87GjT2burerSsuzzLPa+3qvBCVdvhoomrM54eP0ke1qIprTDBRV
6ENiCNAX7dBoODcJUYPaa4e9QDPimFmruNKY+Skn3V7zqZzbdxdR0CFiatX4SSdQv3vLsnGtSqqluT
m5TqNdytnlyRaoTpOpvQilGmm7+93y77d+Z07m1cEabviAnTVq7upqJcBCQFAJ0HA8ec4Mfl/aKuQAL
0rhQBYAA20EbB7FLK9QXShTQnNxzMBoqjtJNoVCkDx8/Geiw8FFJL3ZGDXqb8nJ+80+5NL7R+z4FHu
T+U1MrtgYQUsOo7Wux8+HtaWMc3dnJDiKNeNeIYQMRMEGMTAByY8C8cGACMExzGvGAxgmOYJMAEZHE
TBvEA5gGOTBJjAUaNeKAEQkqmRLDEQE6VSIbVbiQiOIErEgMjxWGSquVxcdh7VPMGEIYMjKKkrNXTJ
Rk4u6dmjL47Y9SkbqM6cwNQO8SChUE2E56+CpvqyKTztY+omJiNjqWdOVux+Zr0drO1qsb9q8vKxSI

```
9hHbEWh7Q2RXGuGdP/jqA+zA/UeczO0Km0qfXoAD5hTLD1BIlD+kYlZLd5/S5YW0aGrb5fVmnWuDOD
H7Vo09GcFvkX4m9Bw85jq+KxL6M7W7VByjzAteQUsO3PyEtUthzftJZcF5vyOE9pwXqR515idqVKmg
PRp3G9Q/zcF8vWciVABlQWH1PMntMkweyK9TqUqh78pt6nSaHZ+59U61WVByHxN+gmzQwdKgrRVvEz
q2KnVfpO/hyM/QU5hpc3FhxueU1Wyd2GdumxNxfhT4G3I/KPfwl/s3ZFCh9mnxfOdW9ezylhO0qcHJ
StmtH3nJVZqLjfJ6mZ3sOSgtJFsrMBYCwCLrbu1t6So2Ds41qgU9RbGofw8vEzdsoIsQCDxBFwZFhs
NTpgimiqCbkAWuZ1jLdVkcZK7Jo8UaQAeBHjRgKNHjQAYmNeOYMACJgXiMaABXgtEDHMYERgmE0AxA
IwDHMGACijRQAjWSrFFAkgxHEUUACEO8UUAHBhXiiiAUe0aKFgAekrcVB8RCSgg4Ko8FAiiiQEsQEe
KMTCijRRhcKKKKIQo0UUAFGiijAUGKKADGIxRQGCY0UUQgYrxooAAYJiijAEwTFFAAYooohH//2Q==
">
      <h3>AMAZON</h3>
      <p>FULL STACK DEVELOPER</p>
      <div class="link">
        <a href="#">Salary:8 LPA</a>
      </div>
      <a href="application.html">APPLY NOW</a>
    </div>

    <div tabindex=0 class="card">
      <img src="https://www.smallbusinesscomputing.com/wp-
content/uploads/2018/07/Freshworks_Logo.jpeg">
      <h3>FRESH WORKS</h3>
      <p>SOFTWARE ENGINEER</p>
      <div class="link">
        <a href="#">SALARY:5 LPA</a>
      </div>
      <a href="application.html">APPLY NOW</a>
    </div>

    <div tabindex=0 class="card">
      <img src="https://www.the-sun.com/wp-content/uploads/sites/6/2022/01/kc-
google-watching-comp.jpg">
      <h3>GOOGLE</h3>
      <p>TESTER</p>
      <div class="link">
        <a href="#">SALARY:7 LPA</a>

      </div>
      <a href="application.html">APPLY NOW</a>
    </div>
     <div tabindex=0 class="card">
      <img src="https://1000logos.net/wp-content/uploads/2020/09/Maersk-
Logo.png">
      <h3>MAERSK</h3>
      <p>FULL STACK DEVELOPER</p>
      <div class="link">
```

```
        <a href="#">SALARY:7 LPA</a>
      </div>
       <a href="application.html">APPLY NOW</a>
    </div>

    <div tabindex=0 class="card">
      <img
src="https://download.logo.wine/logo/TVS_Motor_Company/TVS_Motor_Company-
Logo.wine.png">
      <h3>TVS</h3>
      <p>MECHANIC</p>
      <div class="link">
        <a href="#">SALARY:3 LPA</a><br>

      </div>
        <a href="apply.html">APPLY NOW</a>

    </div>
 <div tabindex=0 class="card">
      <img src="https://mir-s3-cdn-
cf.behance.net/project_modules/max_1200/0af17216851061.562b230c40530.png">
      <h3>ZOHO</h3>
      <p>DEVELOPER</p>
      <div class="link">
        <a href="#">SALARY:4 LPA</a>
      </div>
  <a href="application.html">APPLY NOW</a>
    </div>

  </div>
 </div>
    <script>
  window.watsonAssistantChatOptions = {
    integrationID: "a2445f85-b7a3-4c58-980c-72a414ea407b", // The ID of this
integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "1c2b79b4-a6d1-4c4d-bf2f-852aa415e243", // 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>
```

Main.py

```
from flask import Flask, render_template, request
import ibm_db

 app = Flask(__name__)

 conn = ibm_db.connect(
    "DATABASE=bludb;HOSTNAME=1bbf73c5-d84a-4bb0-85b9-
ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=32286;SECURIT
Y=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=dtk93338
;PWD=honkzW80jUXH3UJN;",
    "", "")
 num=[]
 num1=[]
 num2=[]

@app.route('/')
def index():

    return render_template('homepage.html')

@app.route('/login.html')
def login1():
    return render_template('login.html')

@app.route('/register.html')
def register():
    return render_template('register.html')

@app.route('/apply.html')
def applyjob():
    return render_template('apply.html')

@app.route('/application.html')
def application():
```

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

@app.route('/login',methods=["GET","POST"])
def login():
     i = int(0)
     if request.method=="POST":
        fn= request.form.get('fn')
        sql = "select * from userdetails"
        stmt = ibm_db.exec_immediate(conn, sql)
        while ibm_db.fetch_row(stmt) != False:
           em=ibm_db.result(stmt,0)
           num1.append(em)
           i=i+1

        for y in range(i):
           ele2=str(fn)
           if ele2==num1[y]:
              return render_template("apply.html")




@app.route('/register', methods=["GET", "POST"])
def insert():
    i=int(0)
    if request.method == "POST":
        fn = request.form.get('firstname')
        mn = request.form.get('middlename')
        ln = request.form.get('lastname')
        course = request.form.get('course')
        gender = request.form.get('gender')
        #skill = request.form.get('skillset')
        email = request.form.get('email')
        pw = request.form.get('password')
        sql = "select * from userdetails"
        stmt = ibm_db.exec_immediate(conn, sql)
        while ibm_db.fetch_row(stmt) != False:
           em=ibm_db.result(stmt,0)
           num.append(em)
           i=i+1
        for y in range(i):
           ele2=str(fn)
           if ele2==num[y]:
```

```
            print("already registered mail")
            return render_template("alertwrong.html")


        insert_sql = "Insert INTO userdetails Values(?,?,?,?,?,?,?)"
        prep_stmt = ibm_db.prepare(conn, insert_sql)
        ibm_db.bind_param(prep_stmt, 1, fn)
        ibm_db.bind_param(prep_stmt, 2, mn)
        ibm_db.bind_param(prep_stmt, 3, ln)
        ibm_db.bind_param(prep_stmt, 4, course)
        ibm_db.bind_param(prep_stmt, 5,  gender)
        ibm_db.bind_param(prep_stmt, 6, email)
        ibm_db.bind_param(prep_stmt, 7, pw)
        ibm_db.execute(prep_stmt)
        print("inserted")
        return render_template('register.html')

@app.route('/apply',methods=["GET","POST"])
def applyforjob():
    fn = request.form.get('fn')
    ln = request.form.get('ln')
    citizenship = request.form.get('Citizenship')
    city = request.form.get('city')
    twelve = request.form.get('twelth')
    tenth = request.form.get('tenth')
    insert_sql = "insert INTO newapply values(?,?,?,?,?,?)"
    prep_stmt = ibm_db.prepare(conn, insert_sql)
    ibm_db.bind_param(prep_stmt, 1, fn)
    ibm_db.bind_param(prep_stmt, 2, ln)
    ibm_db.bind_param(prep_stmt, 3, citizenship)
    ibm_db.bind_param(prep_stmt, 4, city)
    ibm_db.bind_param(prep_stmt, 5, twelve)
    ibm_db.bind_param(prep_stmt, 6, tenth)
    ibm_db.execute(prep_stmt)
    return render_template("apply.html")

if __name__ == '__main__':
    app.run(host='0.0.0.0')
```

13.2 GITHUB & PROJECT DEMO LINK:-

- https://github.com/IBM-EPBL/IBM-Project-21179-1659774640
-