

Date	18 November 2022
Team ID	PNT2022TMID33711
Project Name	Project- skill and job Recommender
Team Leader Team Members	S.Nivetha P.Sangavi C.Priya R.Roshini

In partial fulfilment for the award of the degree

Of

BACHELOR OF ENGINEERING

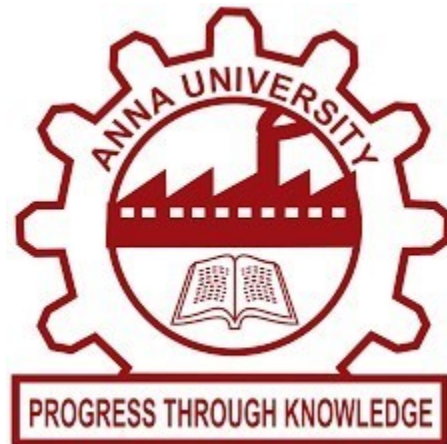
In

DEPARTMENT OF INFORMATION TECHNOLOGY



V.S.B. ENGINEERING COLLEGE, KARUR

V.S.B ENGINEERING COLLEGE, KARUR
(Approved by AICTE & Affiliated by Anna University,
Chennai)



BONAFIDE CERTIFICATE

Certified that this mini project report titled “Fertilizer Recommendation System for Disease Prediction” is the bonafide record work by **NIVETHA S (922519205073)**, **PRIYA C (922519205082)**, **ROSHINI R(922519205090)** and **SANGAVI P (922519205092)** for **IBM-NALAIYATHIRAN** in **VII semester of B.E., degree course in Computer Science and Engineering** branch during the academic year of 2022-2023

Staff-In Charge

Mr. Sathyanarayanan M

Evaluator

Dr.Parthiban M

Head of the Department

Mr. MANIVANNAN K

ACKNOWLEDGEMENT

First and foremost, we express my thanks to our parents for providing us a very nice environment for doing this mini project. We wish to express our sincere thanks to our founder and Chairman **Shri.V.S.BALSAMY** for his endeavour in educating us in this premier institution.

We wish to express our appreciation and gratefulness to our principal, **Dr.V.NIRMAL KANNAN** and vice principal **Mr.T.S.KIRUBASANKAR** for their encouragement and sincere guidance.

We are grateful to our head of the department **Mr. MANIVANNAN K** and our Nalaiyathiran project coordinator **Mrs. Nandhini I** Department of Information Technology for their valuable support.

We express our indebtedness to the supervisor of our Nalaiyathiran project, **Mrs. Nandhini I** head of the department, Department of Information Technology , for guidance throughout the course of our project.

Our sincere thanks to all the teaching staff of V.S.B Engineering College and our friends for their help in the successful completion of this IBM Nalaiyathiran project work. Finally, we bow before God, the Almighty who always had a better plan for us. We give our praise and glory to Almighty God for successful completion of this IBM Nalaiyathiran.

Project Report

1. INTRODUCTION

1.1 Project Overview

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

1.2 Purpose

The main aim of the project is to building a model with a skill recommender solution through which the fresher or the skilled person can log in and find the jobs by using the search option or they can directly interact with the chatbot and get their dream job.

2. LITERATURE SURVEY

1.3 Existing problem

Skill -based recommendation services platforms wants to turn into a leading source of the job recommender platform. The job and skill recommender system is expected to reduce unemployment and improve the skills of job seekers to boost the country's economy. Job recommender is giving guidance for job seekers to get quick contact for those jobs. Immediate announcement of the latest job openings, and walk-in

interviews with different locations and offers you full details of recruitment such as the number of job vacancies, eligibility criteria, interview date, required qualifications etc.

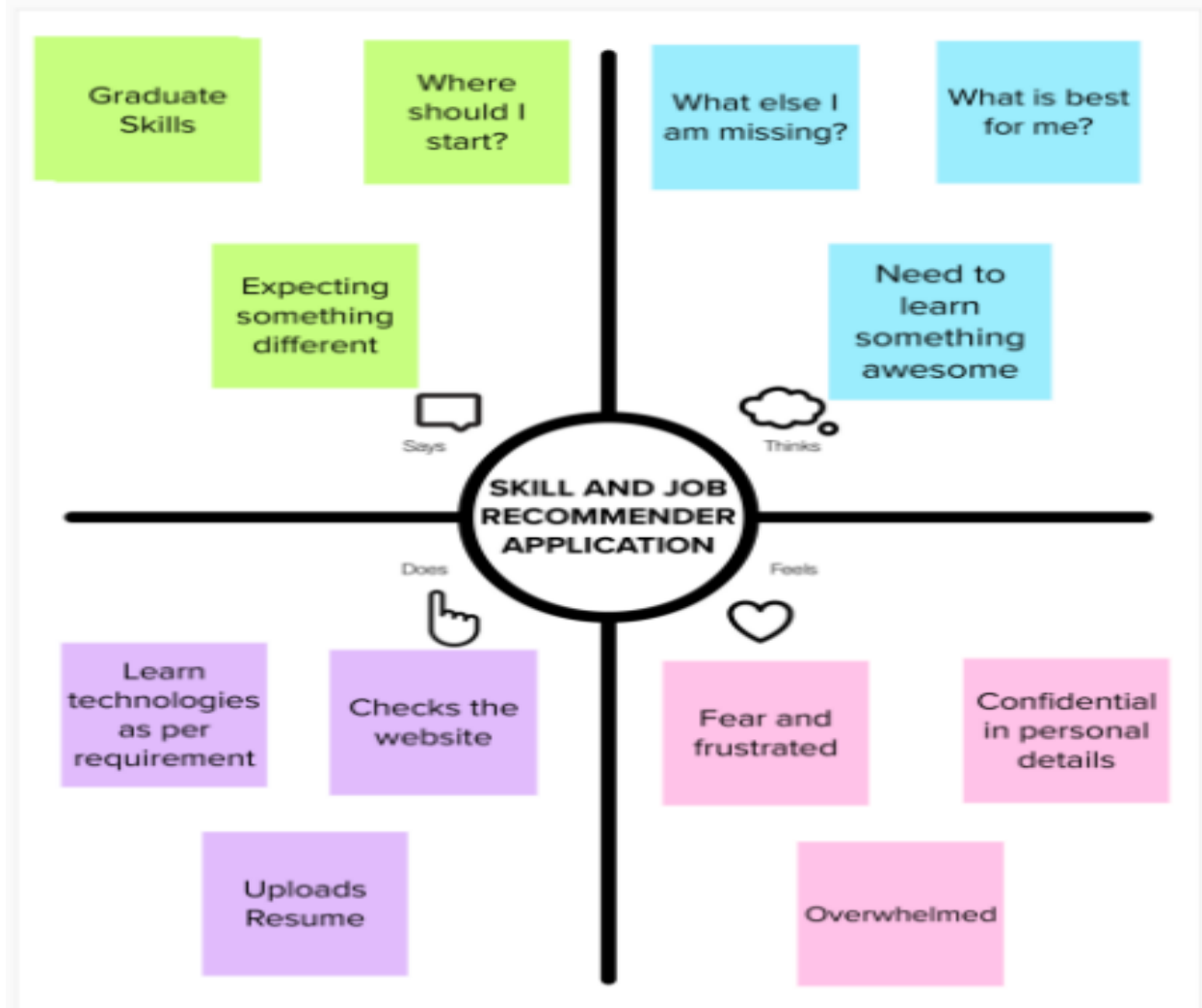
1.4 Problem Statement Definition

The main aim of the project is to develop an end-to-end web application capable of displaying the current job openings based on the user skill set. Users will interact with the chat-bot and can get recommendations based on their skills. This Application uniquely identifies the user's skills recommend the job according to the user's interest.

2. IDEATION & PROPOSED SOLUTION

2.1 Empathy Map Canvas

● Skill and Job Recommender Application Empathy Map



2.2 Ideation & Brainstorming



2.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Managing the huge measure of enrolling data on the Web, a task searcher generally goes through hours to see as helpful ones. Commonly, individuals who need industry information are hazy about what precisely they need to learn to get an appropriate occupation for them. We address the issue of prescribing reasonable responsibilities to individuals who are looking for a new work. We plan this suggestion issues administered AI issue.

2.	Idea / Solution description	<p>Work proposal is a significant undertaking for the cutting edge enrollment industry. A magnificent occupation recommender framework not just empowers to suggest a more lucrative work which is maximally lined up with the range of abilities of the present place of employment, yet in addition proposes to procure not many extra abilities which are expected to accept the new position. In this work, we made three sorts of data net-works from the authentic work information: (i) work change organization, (ii) work ability organization, and (iii) expertise co-event organization.</p>
3.	Novelty / Uniqueness	<p>From the most recent twenty years web based selecting stages have turned into an essential direct in many organizations for enlisting gifts. Such entries decline the ad cost, yet they experience the ill effects of data over-burden issue. Work entryways utilizing customary data recovery procedures, for example, Boolean hunt strategies are commonly utilizing basic word matching calculations. The main pressing concern of these entryways is their powerlessness to figure out the intricacy of matching between applicants' longings and associations' prerequisites.</p>

4.	Social Impact / Customer Satisfaction	<p>Social capability is especially striking for understudies who are arranged into one of the great frequency handicap gatherings like explicit learning disabilities, mental impediment, close to home unsettling influence, or consideration shortfall/hyperactivity jumble. Among the most famous of the educational methodologies for these understudies has been interactive abilities preparing (SST). Different meta-examinations of the writing recommend that SST has not delivered enormous, socially significant, long haul, or summed up changes in friendly capability of understudies with high-rate handicaps. Likely clarifications for the powerless impacts in some meta-examinations are talked about and explicit proposals are presented for planning and creating more successful SST mediations.</p>
5.	Business Model (Revenue Model)	<p>Various postings for various work jobs and occupation positions are posted at various sources in the enrollment business. Thusly, this is a difficult and tedious errand to gather the data and figure out most important client work association planning as indicated by the abilities and inclinations of a client. This examination work has been finished to conceal this equivalent issue and endeavors have been made to give a doable and productive answer for something very similar. We propose a substance based suggestion motor, which consequently gives best ideas to clients by coordinating their inclinations and abilities with the elements of a task posting. To create an expected suggestion, the proposed motor applies different text channels and element similitude estimations</p>

6.	Scalability of the Solution	<p>The present Recommender framework is a somewhat new area of examination in AI. The recommender framework's fundamental thought is to fabricate connection between the items, clients and go with the choice to choose the most proper item to a particular client.</p> <p>There are four principal ways that recommender frameworks produce a rundown of proposals for a client - content-based, Cooperative, Segment and half breeds sifting. In satisfied based sifting the model purposes details of a thing to suggest extra things with comparative properties.</p>
----	-----------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.4 Problem Solution fit

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns

Purpose:

- Solve complex problems in a way that fits the state of your customers.
- Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.

- Sharpen your communication and marketing strategy with the right triggers and messaging.
- Increase touch-points with your company by finding the right problem-behavior fit and building trust by solving frequent annoyances, or urgent or costly problems.

Problem-Solution fit canvas 2.0		Purpose / Vision	
Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS Who is your customer? The one who is interested in acquiring new skills and the one who is need of a job.	6. CUSTOMER CONSTRAINTS CC What constraints prevent your customers from taking action or limit their choices of solutions? Inadequate training, incorrect instruction, lack of necessary information about what to do or how to do it, poor equipment or supplies, lack of equipment or supplies.	5. AVAILABLE SOLUTIONS AS Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? Searching in online is better than noticing advertisement in newspapers.
	2. JOBS-TO-BE-DONE / PROBLEMS J&P Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one opportunity; jobs in various domains.	9. PROBLEM ROOT CAUSE RC What is the real reason that this problem exists? What is the back story behind the need to do this job? Financial crises, un employment in the society.	7. BEHAVIOUR BE What does your customer do to address the problem and get the job done? i.e. directly related: search for jobs related to their skills; indirectly associated: learn new skills
Identify strong TR & EM	3. TRIGGERS TR What triggers customers to act? Facing jobless situation in society, trying to be self dependent.	10. YOUR SOLUTION SL Dealing with the tremendous proportion of selecting information Online, an errand searcher for the most part goes through hours to see as supportive ones. Regularly, people who need industry data are foggy about what unequivocally they need to figure out how to get a proper occupation for them. We address the issue of recommending sensible obligations to people who are searching for another work.	8. CHANNELS of BEHAVIOUR CH 8.1 ONLINE What kind of actions do customers take online? Searching for new skills to upgrade ourselves to get a new job.
	4. EMOTIONS: BEFORE / AFTER EM How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.		8.2 OFFLINE What kind of actions do customers take offline? Refer books , journals, newspaper etc..

3. REQUIREMENT ANALYSIS

3.1 Functional Requirement

S. FUNCTIONAL REQUIREMENT

No (Epic)

SUB REQUIREMENT (Story)

1. Sign In / Login Register with username, password
2. Profile Registration Register with username, password, email, qualification, skills.
This data will be stored in a database.
3. Job profile display Display job profiles based on availability, location ,skills
4. ChatbotA chat on the webpage to solve user queries and issues
5. Job registration A copy of the company the user applied for with its registration/description details will be sent to the registered email id.
6. Logout Logging out of the webpage.

3.2 Non-Functional requirements

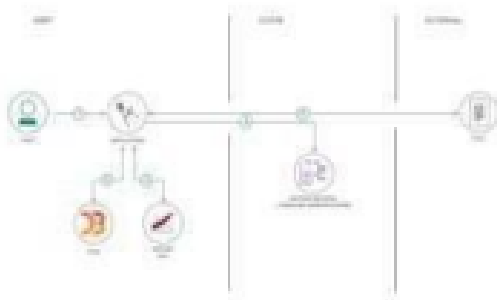
S. No	NON-FUNTIONAL REQUIREMENT	DESCRIPTION
1)	Usability	The webpage will be designed in such a way that any non-technical user can easily navigate through it and complete the job registration work.
2)	Security	Using of SSL certificate (Python Flask to Cloud connect) will provide security to the project. Database will be safely stored in DB2.
3)	Reliability	To make sure the webpage doesn't go down due to network traffic.

4)	Performance	Focus on loading the webpage as quickly as possible irrespective of the number of user/integrator traffic.
5)	Availability	This webpage will be available to all users at any given point of time.
6)	Scalability	Increasing the storage space of database can increase the number of users.

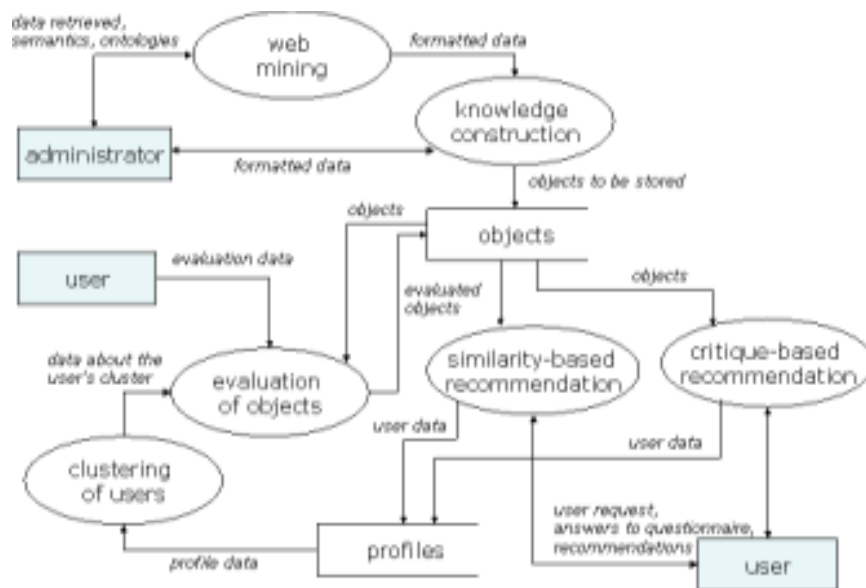
4. PROJECT DESIGN

4.1 Data Flow Diagram

Flow



1. User configures credentials for the Watson Natural Language Understanding service and starts the app.
2. User selects data file to process and load.
3. Apache Tika extracts text from the data file.
4. Extracted text is passed to Watson NLU for enrichment.
5. Enriched data is visualized in the UI using the D3.js library.



4.2 Solution & Technical Architecture

S.No.	Parameter	Description
1.	Is the System Robust?	Yes, it is partially buildable platform as the budget required will be more as cloud is a pay per use model and time taken will be quite.
2.	Is it highly modifiable?	Indeed, the framework is modifiable and it can own up to the progressions by recognizing blunders that requirements to be fixed and new functionalities. It is exceptionally receptive to the progressions.

3.	Is it Scalable?	Indeed, the framework proposed is exceptionally versatile as it can deal with the developing responsibility where great execution is likewise expected to effectively work. Organization of the stage has been finished utilizing different OS virtualization stage it will deal with the responsibility genuinely.
----	-----------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

4.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application.	I can receive confirmation email & click confirm	High	Sprint-1
	Login	USN-3	As a user, I can register for the application through Facebook.	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail.		Medium	Sprint-1
Customer (Web user)	Access	USN-5	As a user, I can log into the application by entering email & password.		High	Sprint-1
	Dashboard	USN-5	As a user, I can access my dashboard after signing in.	I can access my account / dashboard	High	Sprint-1
		USN-6	As a user, I can setup a profile, and basic details by signing in.			
Customer Care Executive	Chatbot	USN-7	As a user, I will upload my resume, certificates, and other requirements.	I can perform several task in the application	Medium	Sprint-1
		USN-8	As a user, I can seek guidance from the customer care executive.		High	Sprint-1
Administrator	DBMS	USN-9	As a administrator, I can keep the applications of your organization relies on running.	I can perform various modifications in the applications.	High	Sprint-1

5. PROJECT PLANNING & SCHEDULING

5.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Acceptance criteria	Team Members
Sprint-1	UI Design	USN-1	As a job seeker experience an awesome user interface in the website	Medium	Better Impression about a website	Dinesh Raj
Sprint-1	Registration	USN-2	As a job seeker register for the application by entering my email, password, and confirming my password.	High	I can access my account / dashboard	Dasvanth
Sprint-1		USN-3	As a job seeker receive confirmation email once I have registered for the application	High	I can receive confirmation email & click confirm	Hari Haran
Sprint-1		USN-4	As a job seeker register for the application through Facebook	Low	I can register & access the dashboard with Facebook Login	Dharshan
Sprint-1		USN-5	As a job seeker register for the application through Gmail	Medium	I can receive confirmation email & click confirm	Dinesh Raj
Sprint-1	Login	USN-6	As a job seeker log into the application by entering email & password	High	I can access my account / dashboard	Dasvanth, Dharshan

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Acceptance criteria	Team Members
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	Hari Haran
Sprint-2	User Profile	USN-9	As a user, I can view and update my details	Medium	I can modify my details/data	Dinesh Raj
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	Dasvanth
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	Dharshan
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	Hari Haran
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	Dinesh Raj

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Acceptance criteria	Team Members
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.	Dasvanth
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	Dharshan
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	Dinesh Raj
Sprint-3	Kubernetes	USN-17	As a user, I can access the website in any device	High	I can access my account in any device	Dasvanth
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	Hari Haran
Sprint-3	Technical support	USN-19	As a user, I can get a customer care support from the website which will solve my queries.	Medium	I can tackle my problem & queries.	Dharshan
Sprint-4	Unit Testing	USN-15	As a user, I can access the website without any interruption	High	I can access the website without any interruption	Dharshan
Sprint-4	Integration testing	USN-16	As a user, I can access the website without any interruption	High	I can access the website without any interruption	Hari Haran

5.2 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	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

```

1 import ibm_db
2 from flask import Flask, flash, redirect, render_template, request, url_for
3 from flask_cors import CORS, cross_origin
4 from flask import session
5 import sqlite3
6 import os
7
8 conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=b1bc1829-6f45-4cd4-bef4-10cf081900bf.clogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32304;")
9
10 con=sqlite3.connect("myimage.db")
11 con.execute("create table if not exists image(pid integer primary key,img TEXT)")
12 con.close()
13
14 app = Flask(__name__)
15 app.secret_key = '//sd_5#y2L"F4Q8z\n\xec]/'
16
17 app.config['UPLOAD_FOLDER']="static/images"
18
19 EMAIL=''
20
21 @app.route('/')
22 def index():
23     return render_template('signup.html')
24
25 @app.route('/home')
26 def home():
27     login=False
28     if 'usernameid' and 'passwordid' in session:
29         login=True
30     return render_template('index.html', login=login)
31
32 @app.route('/signup', methods=['POST','GET'])
33 @cross_origin()
34 def signup():

```

```

72     EMAIL=email
73     sql="SELECT * FROM authentication WHERE email=? AND password=?"
74     stmt=ibm_db.prepare(conn,sql)
75     ibm_db.bind_param(stmt,1,email)
76     ibm_db.bind_param(stmt,2,password)
77     ibm_db.execute(stmt)
78     account=ibm_db.fetch_assoc(stmt)
79     if account:
80         #session['loggedin'] = True
81         #session['email'] = email
82         #return render_template('home.html')
83         return redirect(url_for('home'))
84     else:
85         error = "Invalid email/password"
86         return redirect(url_for('login'))
87     elif request.method=='GET':
88         return render_template('login.html')
89
90 @app.route('/forgot', methods=['POST','GET'])
91 def forgot():
92     if request.method=='POST':
93         email=request.form['email']
94         remail=email
95         secret=request.form['secret']
96         sql="SELECT * FROM authentication WHERE email=? AND secret=?"
97         stmt=ibm_db.prepare(conn,sql)
98         ibm_db.bind_param(stmt,1,email)
99         ibm_db.bind_param(stmt,2,secret)
100        ibm_db.execute(stmt)
101        account=ibm_db.fetch_assoc(stmt)
102        if account:
103            return redirect(url_for('reset'))
104        else:
105            return redirect(url_for('forgot'))

```

```
app.py — D:\IBM PROJECT\sprint1 x app.py — C:\IBM project - final x
def post():
    con = sqlite3.connect("myimage.db")
    con.row_factory = sqlite3.Row
    cur = con.cursor()
    cur.execute("select * from image")
    data = cur.fetchall()
    con.close()

    if request.method == 'POST':
        upload_image = request.files['upload_image']

        if upload_image.filename != '':
            filepath = os.path.join(app.config['UPLOAD_FOLDER'], upload_image.filename)
            upload_image.save(filepath)
            con = sqlite3.connect("myimage.db")
            cur = con.cursor()
            cur.execute("insert into image(img) values(?)", (upload_image.filename,))
            con.commit()
            flash("File Upload Successfully", "success")

            con = sqlite3.connect("myimage.db")
            con.row_factory = sqlite3.Row
            cur = con.cursor()
            cur.execute("select * from image")
            data = cur.fetchall()
            con.close()
            return render_template("post.html", data=data)
    return render_template("post.html", data=data)

@app.route('/delete_record/<string:id>')
def delete_record(id):
    try:
        con = sqlite3.connect("myimage.db")
        cur = con.cursor()
        cur.execute("delete from image where nid=?" % (id))
```

```
index.html — Assignments\assignment 3\templates x index.html — github\...\templates x sample.html x app.py x
1 {% extends 'base.html' %}
2
3 {% block head %}
4 <title>Home - Instant job</title>
5 <link rel="stylesheet" href="static/css/style.css">
6 <link rel="stylesheet" href="static/css/index.css">
7 {% endblock %}
8
9
10 {% block content %}
11
12
13
14 <div class="topbar">
15
16     <div class="topbarLeft">
17         <h2 class="topbarHead">INSTANT JOB</h2>
18     </div>
19
20     <div class="topbarCenter">
21         <ul class="topbarList">
22             <li class="topbarListItem"><a class="topbarListItem" href="contacts">Contacts</a></li>
23             <li class="topbarListItem"><a class="topbarListItem" href="login">Log in</a></li>
24             <li class="topbarListItem"><a class="topbarListItem" href="signup">Sign up</a></li>
25         </ul>
26     </div>
27 </div>
28
29 <div class="header">
30
31     <div class="headerImg">
32         <h3 class="headerTitle">WELCOME TO INSTANT JOB</h3>
33     </div>
34
35
```

```
app.py — D:\IBM PROJECT\sprint1 x app.py — C:\_ibm project - final x
1 import ibm_db
2 from flask import Flask, flash, redirect, render_template, request, url_for
3 from flask_cors import CORS, cross_origin
4 from flask import session
5 import sqlite3
6 import os
7
8 conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=b1bc1829-6f45-4cd4-bef4-10cf081900bf.clogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32304;")
9
10 con=sqlite3.connect("myimage.db")
11 con.execute("create table if not exists image(pid integer primary key,img TEXT)")
12 con.close()
13
14 app = Flask(__name__)
15 app.secret_key = '"/sd_5#y2L"F4Q8z\n\xec]/'
16
17 app.config['UPLOAD_FOLDER']="static\images"
18
19 EMAIL=""
20
21 @app.route('/')
22 def index():
23     return render_template('signup.html')
24
25 @app.route('/home')
26 def home():
27     login=False
28     if 'usernameid' and 'passwordid' in session:
29         login=True
30     return render_template('index.html', login=login)
31
32 @app.route('/signup', methods=['POST', 'GET'])
33 @cross_origin()
34 def signup():
```

7.2 Feature 2

```
app.py — D:\IBM PROJECT\sprint1 x app.py — C:\_ibm project - final x forgot.html x jobApplication.html x
118 <input class="shadow form-control" type="text" id="curcityid" name="curcity" required><br><br>
119 <label for="curstateid">State:</label>
120 <input class="shadow form-control" type="text" id="curstateid" name="curstate" required><br><br>
121 <label for="curcntryid">Country:</label>
122 <input class="shadow form-control" type="text" id="curcntryid" name="curcntryid" required><br><br>
123 </fieldset><br>
124
125 <fieldset>
126 <legend>QUALIFICATIONS</legend><br>
127 <table>
128 <tr>
129 <td align="center"><b>Sl.No.</b></td>
130 <td align="center"><b>Degree</b></td>
131 <td align="center"><b>Board</b></td>
132 <td align="center"><b>Percentage</b></td>
133 <td align="center"><b>Year of Passing</b></td>
134 </tr>
135
136 <tr>
137 <td>1</td>
138 <td>Class X</td>
139 <td><input type="text" name="Xboard" required></td>
140 <td><input type="text" name="XPercent" required></td>
141 <td><input type="number" name="XYOP" required></td>
142 </tr>
143
144 <tr>
145 <td>2</td>
146 <td>Class XII</td>
147 <td><input type="text" name="XIIboard" required></td>
148 <td><input type="text" name="XIIPercent" required></td>
149 <td><input type="number" name="XIYOP" required></td>
150 </tr>
151
152 <tr>
```

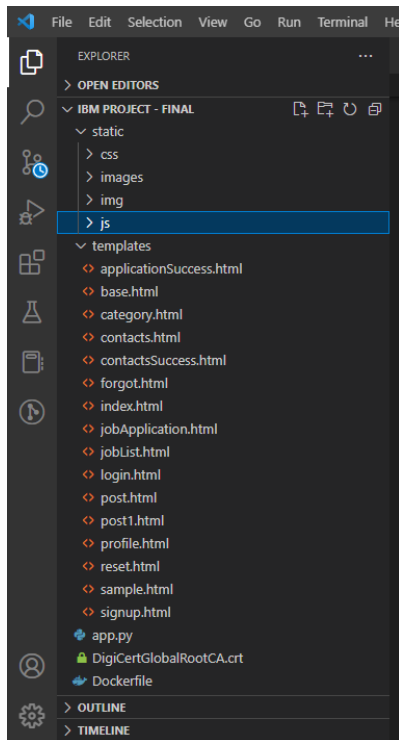
Line 133, Column 83 Tab Size: 4 HTM

```
app.py — D:\BM PROJECT\sprint1 x app.py — C:\bm project - final x forgot.html x jobApplication.html x JobList.html x
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="utf-8">
6   <title>Instant job</title>
7   <meta content="width=device-width, initial-scale=1.0" name="viewport">
8   <meta content="" name="keywords">
9   <meta content="" name="description">
10
11   <!-- Favicon -->
12   <link href="img/favicon.ico" rel="icon">
13
14   <!-- Google Web Fonts -->
15   <link rel="preconnect" href="https://fonts.googleapis.com">
16   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
17   <link href="https://fonts.googleapis.com/css2?family=Heebo:wght@400;500;600&family=Inter:wght@700;800&display=swap" rel="stylesheet">
18
19   <!-- Icon Font Stylesheet -->
20   <link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.4.1/font/bootstrap-icons.css" rel="stylesheet">
21
22   <!-- Libraries Stylesheet -->
23   <link href="lib/animate/animate.min.css" rel="stylesheet">
24   <link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">
25
26   <!-- Customized Bootstrap Stylesheet -->
27   <link href="css/bootstrap.min.css" rel="stylesheet">
28
29
30
31
32   <link rel="stylesheet" href="static/css/style.css">
33   <link rel="stylesheet" href="static/css/index.css">
34 </head>
```

```
app.py — D:\BM PROJECT\sprint1 x app.py — C:\bm project - final x forgot.html x jobApplication.html x JobList.html x reset.html x
1 {% extends 'base.html' %}
2
3 {% block head %}
4   <title>Forgot - Instant job</title>
5   <link rel="stylesheet" href="static/css/style.css">
6   <link rel="stylesheet" href="static/css/index.css">
7 {% endblock %}
8
9
10 {% block content %}
11
12   <div class="topbar">
13
14     <div class="topbarLeft">
15       <h2 class="topbarHead">INSTANT JOB</h2>
16     </div>
17
18     <div class="topbarCenter">
19       <ul class="topbarList">
20         <li class="topbarListItem"><a class="topbarListItem" href="signup">Sign up</a></li>
21         <li class="topbarListItem"><a class="topbarListItem" href="login">Log in</a></li>
22       </ul>
23     </div>
24   </div>
25
26   <section class="py-4 py-md-5 mt-5">
27     <div class="container py-md-5">
28       <div class="row d-flex align-items-center">
29         <div class="col-md-5 col-xl-4 text-center text-md-start">
30           <h2 class="display-6 mb-4">Reset your password</h2>
31           <p class="text-muted">Enter email and new password.</p>
32           <form method="post">
33             <div class="mb-3"><input class="shadow form-control" type="email" name="email" placeholder="Email"></div>
34             <div class="mb-3"><input class="shadow form-control" type="password" name="password" placeholder="Password"></div>
35           </form>
36         </div>
37       </div>
38     </div>
39   </section>
40 {% endblock %}
```

8. TESTING

8.1 Test Cases



8.2 User Acceptance

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

Find schemas or tables Refresh

SQL Schemas

Tables New table

Name	Schema	Properties
<input type="checkbox"/> SIGNUP	CWY91974	...

Total: 1, selected: 0

Table definition

SIGNUP
Approximate 3 rows (32.0 KB)
Updated on 2022-11-11 06:32:36

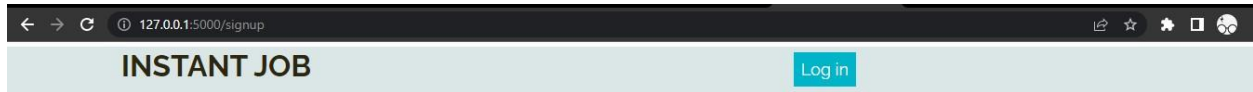
Name	Data type	Nullable	Length	Scale
NAME	VARCHAR	Y	32	0
EMAIL	VARCHAR	Y	32	0
PASSWORD	VARCHAR	Y	32	0

View data

9.1 Performance Metrics

9.1 Performance Metrics

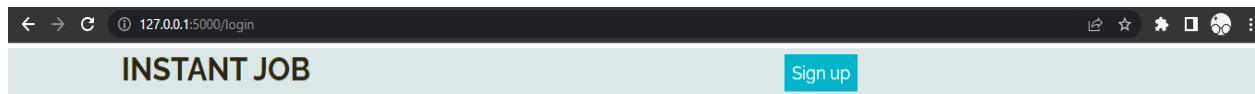
9.2 Output



Sign Up

[Create account](#)

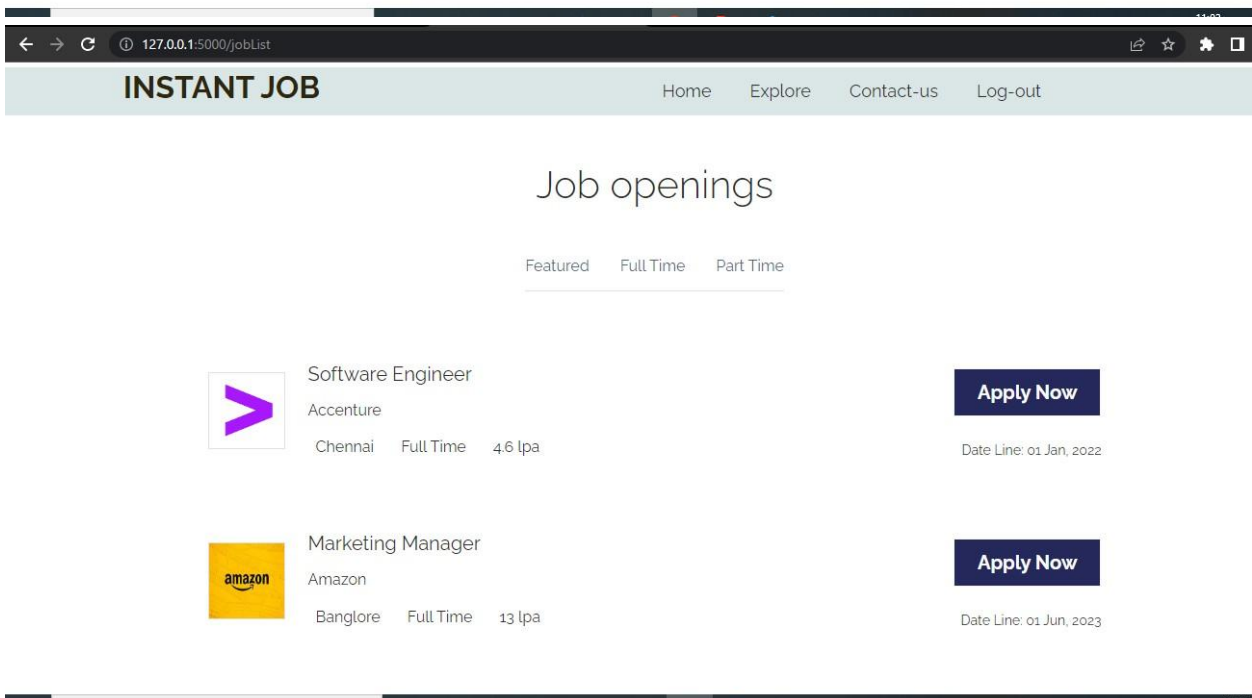
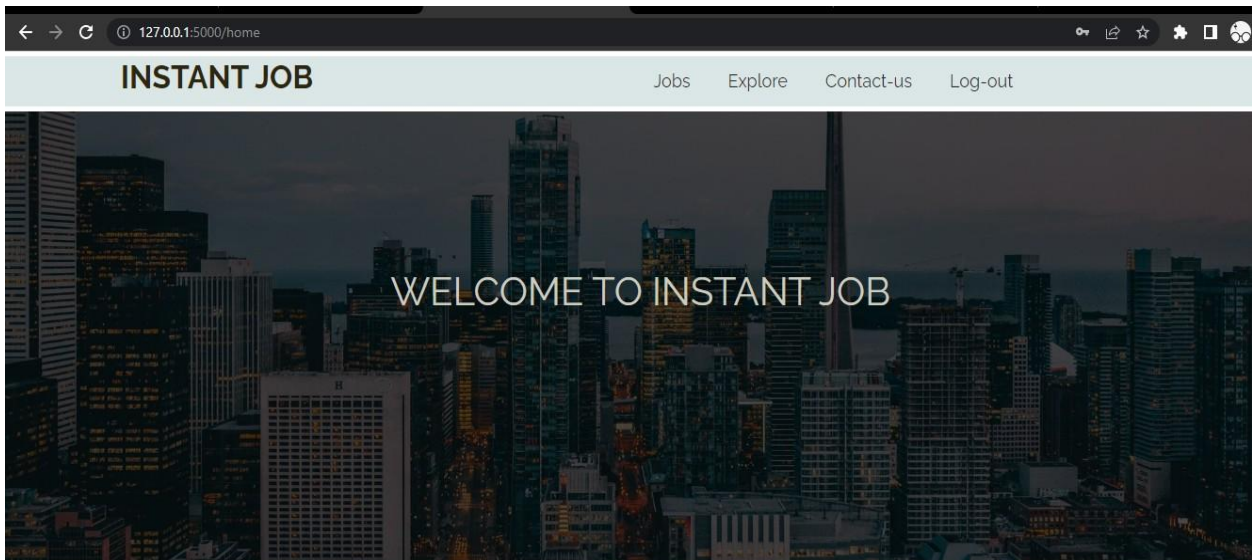
Already have an account? [Log in](#) →

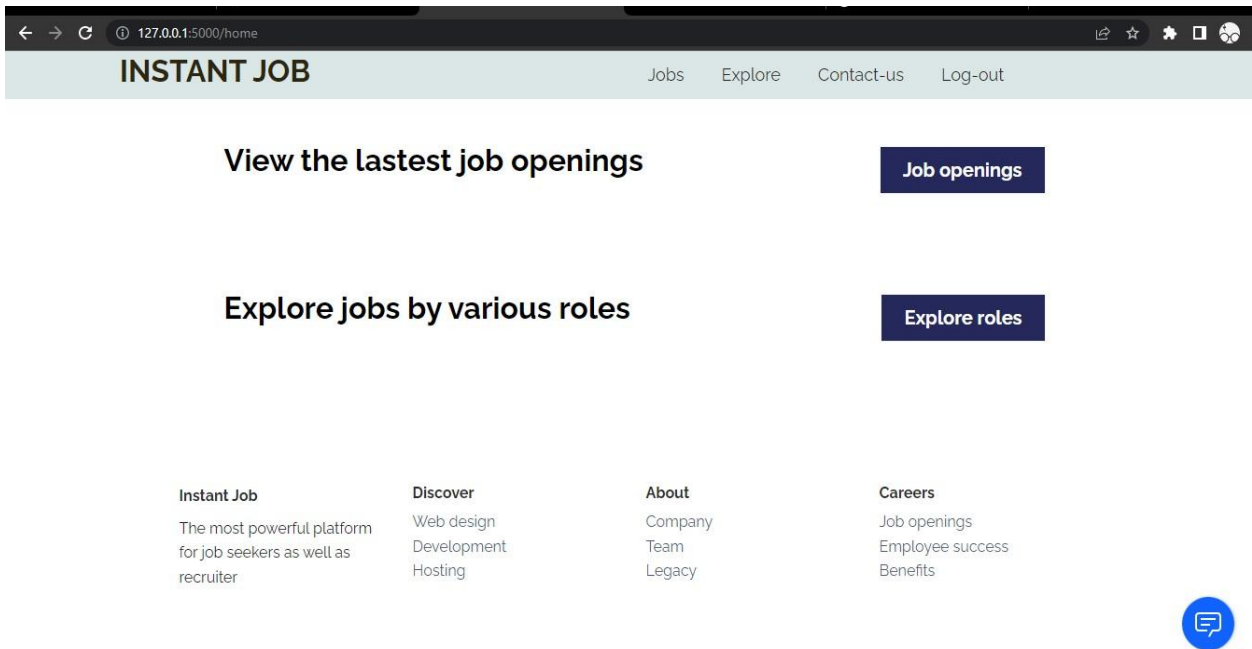


Login

[Log in](#)

[Forgot your password?](#)




A screenshot of the 'INSTANT JOB' application form. The browser's address bar shows '127.0.0.1:5000/jobApplication'. The header includes the logo 'INSTANT JOB' and links for 'Contact', 'Log in', and 'Sign up'. The form is titled 'APPLICATION FORM' in red, followed by 'PERSONAL DETAILS' in black. It contains four input fields: 'First Name:', 'Last Name:', 'Email (Enter same as account email id):', and 'Phone number:'. The phone number field is pre-filled with '+91 xxxxxxxxxxxx'.

IBM IBM-P How t In: X App P Using Tella- Vime- ibm c Goog Goog googl Perso My D Proj+ 127.0.0.1:5000/category

INSTANT JOB Home Jobs Contact-us Log-out

Get the job you really want

Discover your options with your personalized career search

 Software Engineer
Accenture
Java , C programming
Chennai Full Time 4.6 lpa

Apply Now

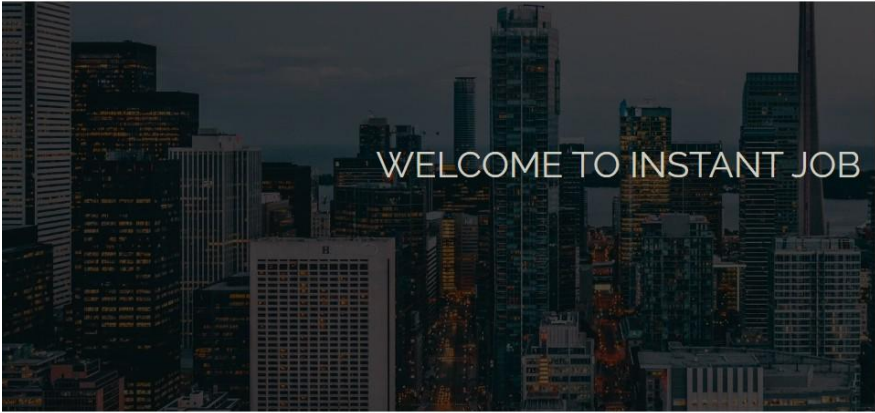
Date Line: 01 Jan, 2022

Marketing Manager

Type here to search

IBM IBM-P How t In: X App P Using Tella- Vime- ibm c Goog Goog googl Perso My D Proj+ 127.0.0.1:5000/home

INSTANT JOB Jobs Explore Contact-us Profile Log-out



WELCOME TO INSTANT JOB

Chase your career

Watson Assistant

Job

I'm afraid I don't understand. Please rephrase your question.

hi

Welcome, how can I assist you?

Built with IBM Watson®

Type here to search

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

CWY91974.APPLICATION Back

Export to CSV

UFXNAME	ULNAME	UEMAIL	UPHONE	UDOB	UGENDER	CURADD	CURZIPID	CURCITY	CURSTATE	CURCNTRYID	XE
User	One	user1@gmail.com	9876543210	2022-11-14	Male	Main road	636005	Salem	Tamilnadu	India	St
User	S	user4@gmail.com	6847	2022-11-15	Female	assa	636005	Salem	Tamilnadu	India	St
User	four	user4@gmail.com	868978798798797	2022-11-15	Others	Main road	698	Salem	Tamilnadu	India	St
User	three	user3@gmail.com	9876543210	2022-11-15	Others	Main road	-1	Salem	Tamilnadu	India	sd
User	two	user2@gmail.com	0123456789	2022-11-15	Female	Main road	636363	Madurai	Tamilnadu	India	St

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- When recruiting externally, hiring teams find candidates, evaluate them and, if all goes well, persuade them to join their company. All of which takes time.
- Everyone needs some time to adjust to a new role, but internal hires are quicker to onboard than external hires.
- May be familiar with people in their new team, especially in smaller businesses.
- Know how your company operates and most of your policies and practices.

DISADVANTAGES

- Employees who were considered for a role could feel resentful if a colleague or external

candidate is eventually hired.

- While your company may have a lot of qualified candidates for specific positions, this isn't necessarily true for every open role.

11. CONCLUSION

By the end of this project we will

- know fundamental concepts and techniques of recommender system.
- gain a broad understanding of databases and cloud.
- know how to build a web application using the Flask framework.
- know how to build chatbot.
- know how to containerize the application.

12. FUTURE SCOPE

- AI is revolutionizing the recommender systems.
- The popularity of LinkedIn and Google for jobs has proved that there is a future for job boards if effectively managed to provide solutions
- Right pricing strategies for online recruitment advertising are essential to get an effective response.
- Recruiters and job seekers are experiencing an entirely automated process of searching and connecting. All job boards should be perfectly indexed, highly responsive, and exhaustive in job descriptions to establish their credibility and reliability.

13. APPENDIX

Ref: <https://github.com/IBM-EPBL/IBM-Project-40069-1660622812>