PROJECT REPORT

RECRUITIO - A Smart Chatbot enabled recruiter

Team name: **The_Alchemists**
Team members: **Sufyan Parkar, Chaitanya Dandekar, Manish Pawar**
E-mail: **sufyanparkar@gmail.com**
Category: **AI Recruiter – Shortlist a Suitable candidate for specific Job Role**
Application ID: **SPS_CH_APL_20200006684**
Project ID: **SPS_PRO_844**

DEMO VIDEO LINK:

[redacted]

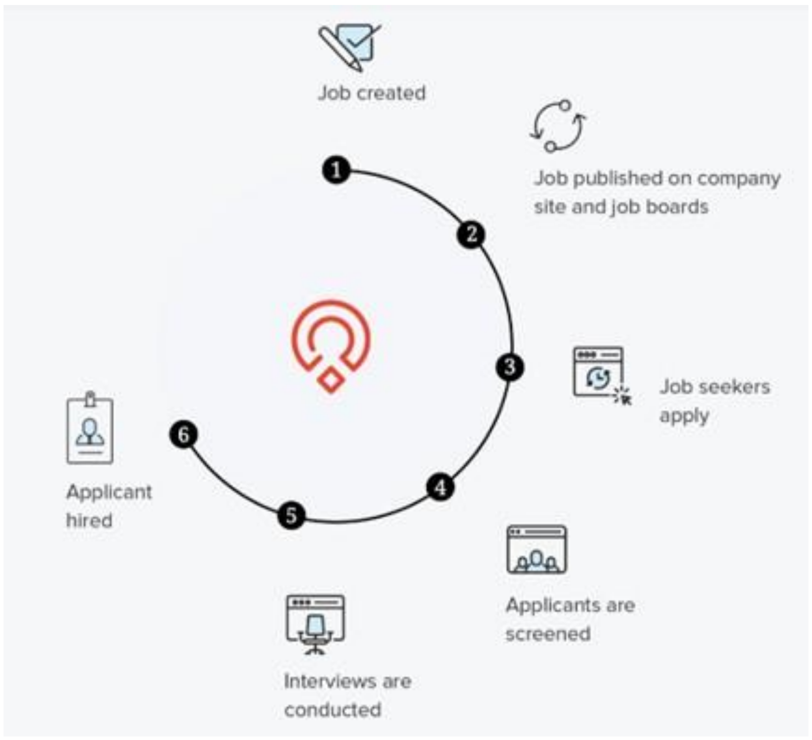
[redacted]

INDEX

1. INTRODUCTION

1.1 Overview

The holy grail of recruitment is finding a quick, easy, and accurate way to automate candidate shortlisting. As organizations start to embrace the idea that recruiting and retaining talented employees represents a competitive advantage.



- Project needs:
  IBM Cloud, Watson-Assistant, Node-Red
- Technical needs:
  Artificial Intelligence, Machine Learning, Watson AI, NodeJS, Javascript, CSS, HTML5.
- Project Deliverables:
  A smart Chatbot for recruitment.

- Project Team:
  The_Alchemists

1.2 Purpose

The future of recruitment through technology has a direct & positive impact on the top & bottom line of any organization. But as things change more, the more they stay the same. Despite these demands for change, the classic challenges of recruitment remain including how to find, attract, screen, and shortlist candidates. Shortlisting is often the most challenging and time-consuming step in the recruitment process. In the world of recruiting, shortlisting is defined as the process of finding the best possible candidate from your pool of applicants for an open position and advancing them to the next stage of the interview which is typically an in-person meeting. While it seems straightforward, it’s not. In fact, many leaders in hiring say that shortlisting candidates is the most difficult part of their job. According to industry stats, 75% of applicants are unqualified and 88% are not strong enough to move forward to an interview.

1.2.1 Scope of work

- An appealing home-page website with details and introduction of the company.
- A login-registration page and connecting with database like cloudant.
- A page to upload resume & parse that resume according to skills.
- Build a Watson assistant chatbot which will guide in recruitment process.
- Build a comparision chart for candidate's scores against company requirements and shortlisting based on some criteria

2. LITERATURE SURVEY

2.1 Existing Problem

52% of talent acquisition leaders say that the most difficult part of their job is to shortlist the right candidate and 3% of candidates never hear back from a company after one touchpoint. On the flip side, it’s a challenge for employers to communicate well with all their candidates. For high volume recruiting, this would require communicating with thousands of candidates, in addition to a recruiter’s normal screening functions and other duties. A survey of talent acquisition leaders found that while 46% struggle with attracting strong candidates in the current candidate-driven talent market, 52% said the most difficult part of recruitment was identifying the right candidates from a large applicant pool.

2.2 Proposed Solution

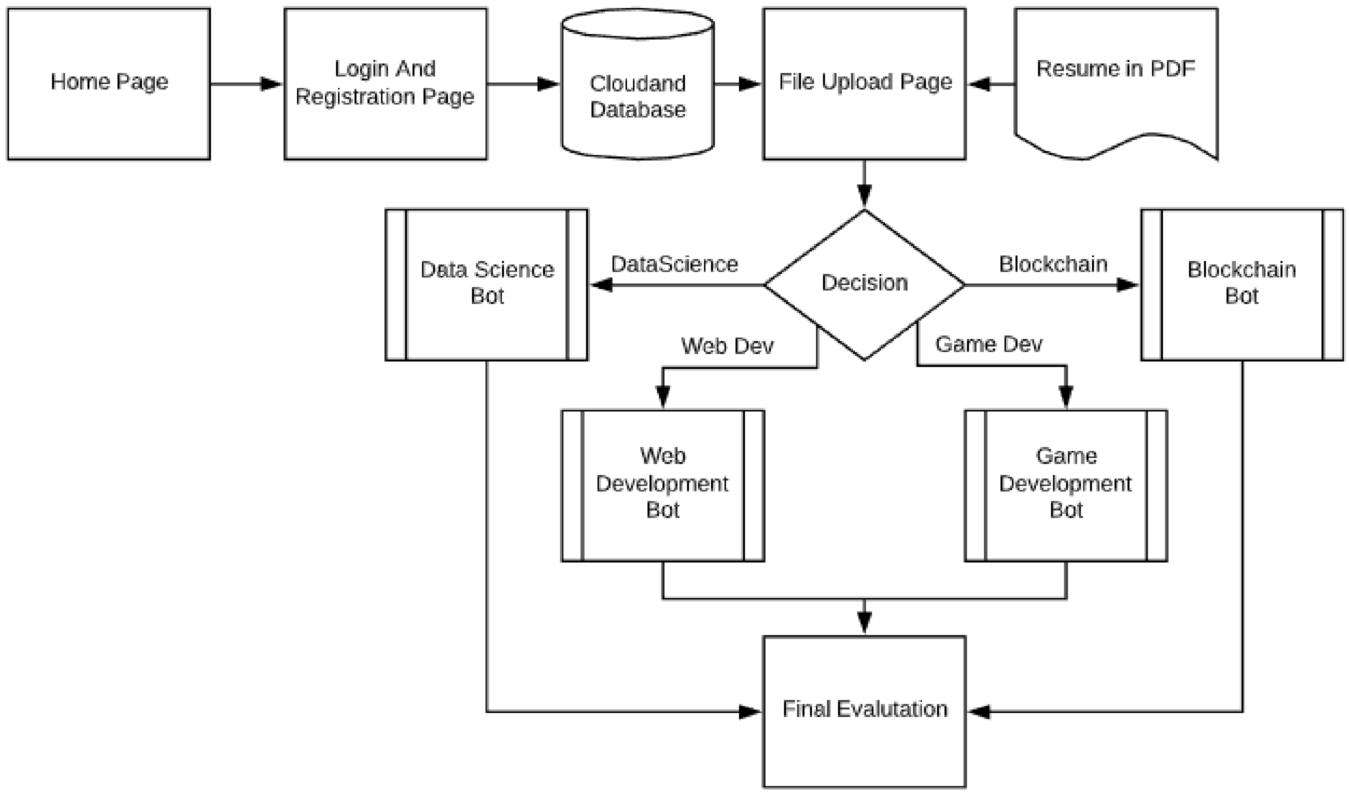
Competition for talent is fiercer than ever, and it’s never been more difficult for recruiters to find the right talent. Times like these, call for hiring managers to adapt to a solution that completely streamlines the entire hiring process. We thus present a smart way to shortlist candidates applying for a job or an internship in a particular firm or a company, thus bridging the gap between the HR and the applicants, also aiding in better assessment of talented individuals, through sophisticated procedural steps that selects the strongest candidates through quality and skill check using a Virtual Assistant.

This has been made possible by leveraging the use of IBM Cloud functionalities like IBM Watson Assistant enabled Chatbot, Cloudant and DB2 for database, Watson Natural Language Understanding (NLU) for information extraction and Node-Red for UI and flow integration. The webpage has been made by basic web development languages like HTML and CSS, and with proper Javascript integration.

3. THEORETICAL ANALYSIS

3.1 Architecture/Flow

The following flow is the basic working flow of the project.



The above flow describes our recrutio project. From the top left we have the home page. This page will contain the Information about the project. Further in the home page we have the login and the registration page. These will be the two seperate pages. The registration page will store the user data in the cloudant database whereas, the login page will give access to the user. Further, file upload page will ask the user to upload their resume after that according to the filtering process it will decide which bot it will go to. Now when the user arrives at a bot, the bot will ask the user relevant question related to the specific field. It will take the user skills with the rating process. After it has calculated them it will take the user to the final evaluation page. At this page it will calculate and show whether the user is shortlisted or not for further process.

3.2 Hardware/Software Designing

- Create IBM cloud services
- Configure Watson Assistant as per user skills
- Connecting with cloudant
- Create flows in node red for home, login-reg, file-upload, Assistant bot page, Summary page
- Configure all flows, deploy and run the app.
- Software skills: Nodejs, Javascript, Bootstrap, CSS, HTML5.

4. EXPERIMENTAL INVESTIGATIONS

4.1 Create IBM cloud services configuring each of them.

- We begin with creating IBM cloud account and initiating required cloud services, like Watson Assistant, Node-Red, and Cloudant.
- These services are then connected with their API keys, and for deployment automation required toolchains were integrated.
- A Cloud Foundary application with 512MB is used for the Cloudant SQL Database.

- UI with Node-Red
- Candidate's resume PDF parsing
- Connecting with Cloudant
- Watson Assistant
- Summarize results

4.2 UI Webpage using Node-Red.

A webpage for the applicants user interface was designed using simple HTML and javascript, and styled using CSS and Bootstrap.

This webpage illustrates the company information, with an application portal. The user then has the option to 'Login' or 'Register' in order to use the portal. In the following steps the applicant is asked to upload his/her resume in PDF format, which is then parsed for domain based skills.

With this, we get applicants skillsets and expertise, and he/she is then redirected to the particular HR that is replaced by a Bot. For this use case we have designed 4 chatbots corresponding to each domain viz. Data Science, Blockchain, Web Development and Game Development.

We made the node red flow for assistants of respective domains configured to work with user uploaded resume, adding some node red dashboad ui elements to enhance the look.

4.3 Candidates resume PDF parsing

The candidates resume is then examined and the algorithm searches for words that relates to technical skills, these extracted keywords let us determine the expertise of the candidate, using that, particular Chatbots are trigerred, with which the applicant later interacts.

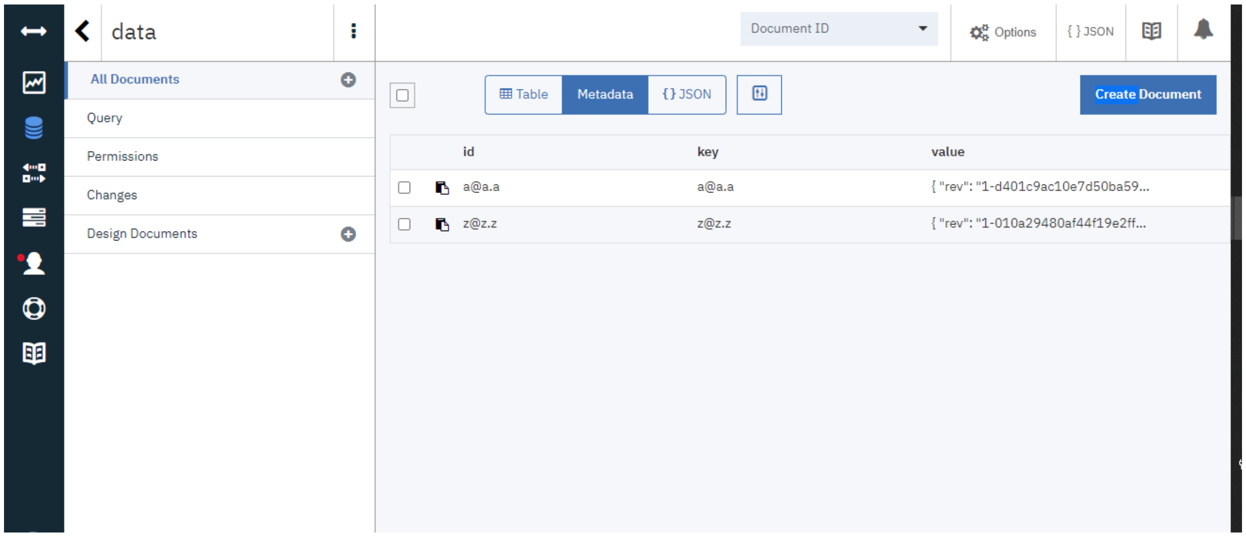
Code:

```
function searchword(text, word)
{
    var x = 0, y=0;
    var w = word;
    for (i=0;i< text.length;i++)
        {
        if(text[i] == word[0])
            {
                for(j=i;j< i+word.length;j++)
                {
                    if(text[j]==word[j-i])
                    {
                        y++;
                    }
                    if (y==word.length)
                    {
                        x++;
                    }
                }
            y=0;
            }
        }

    if(x > 0)
    {
        return w
    }
    else
    {
        return 0
    }
}

function multicall2(list)
{
    var ultlist = [];
    var size = [];
    for(let i = 0;i<list.length;i++)
    {
        var semiult = [];
        var big = list[i];
        var take;
        for(let j=0;j<big.length;j++)
        {
            take =searchword(msg.payload.pages[0].text,
big[j]);
            if (take == 0)
            {
                continue;
            }
            else
            {
                semiult.push(big[j]);
            }
        }

        ultlist.push(semiult);
        size.push(semiult.length);
    }
    return [ultlist,size];
}
var ds = ['Deep
Learning','Tensorflow','Pytorch','Keras','Caffe','CNN','RNN'
,'Theaono','Sklearn','OpenCV','Django','Flask','Azure','Goog
le Cloud','AWS','IBM'];
var wb =
['HTML','JavaScript','Apache','Spark','Django','Flask','Azur
e','Google
Cloud','AWS','IBM','Ruby','Typescript','Swing','Db2','MySQL'
];
var gd = ['C','C++','C#','C
sharp','Javascript','Python','Java','Unity','Frostbite','Unr
eal Engine','GameMaker'];
var bc =
['Cryptography','Ethereum','Tether','Bitcoin','Mining','NEO'
,'EOS','Swift','Javascript','Python','Solidity','Hyperledger
'];
```

```
var main = [ds,wb,gd,bc];
var f,elements,sizes;
f = multicall2(main);
elements = f[0];
sizes = f[1]
var stringg = ['DS','WB','GD','BC'];
var maxn = sizes.indexOf(Math.max.apply(null,sizes));

a = {payload : elements[maxn]};
var b = {payload : stringg[maxn]};
return [a,b];
```

4.4 Cloudant for Database.

Another IBM Cloud service named Cloudant - an SQL Database is used in this project to store the applicant information, which can be later retrieved. As this is a structured database, all the information is stored in tables. The HR person has access to this table, thus filtering out only the eligible candidates for future selection process.



id	key	value
a@a.a	a@a.a	{ "rev": "1-d401c9ac10e7d50ba59...
z@z.z	z@z.z	{ "rev": "1-010a29480af44f19e2ff...

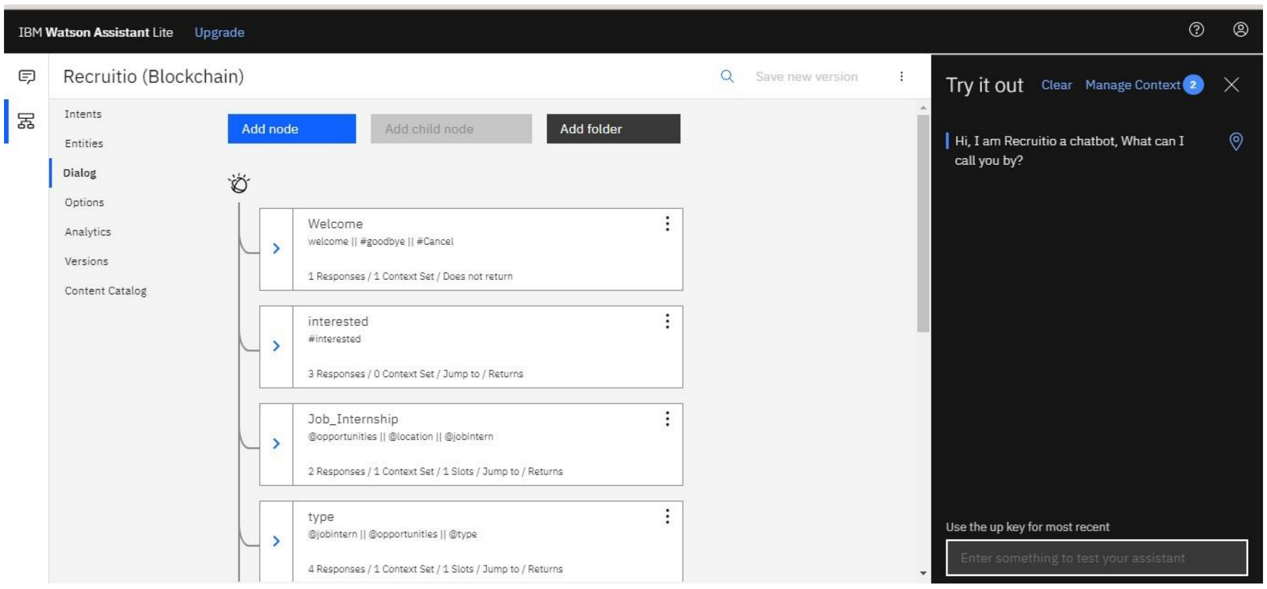
Above is the cloudant database to store the user login details. It's already encrypted by IBM so that there's no data leak if deployed in production.

4.5 Chatbot using Watson Assistant

We created a Chatbot using the famous IBM Watson Assistant service, this chatbot is used on the frontend for interacting with the applicant, that seeks job/internship on our website.

Various bot parameters like Intents, Entities and Dialog flow was effectively utilised in order to achieve best possible result, we can claim that the bot can handle any type of recruitment conversation whether it be with a sophisticated user or someone messing around.
Context variable – a Watson Assistant parameter has enabled to gain a lot of insights on the user entered answers, this has allowed to extract certain information that user enters while chatting with the bot, like rating him/her for technical skills. Thus, getting the applicant interests like job/internship or part-time/full-time/work from home.

We also managed to get the details of the location that the applicant is interested to work at. Along with that the applicants soft skills are also looked at.



Welcome
welcome || #goodbye || #Cancel
1 Responses / 1 Context Set / Does not return

interested
#interested
3 Responses / 0 Context Set / Jump to / Returns

Job_Internship
@opportunities || @location || @jobintern
2 Responses / 1 Context Set / 1 Slots / Jump to / Returns

type
@jobintern || @opportunities || @type
4 Responses / 1 Context Set / 1 Slots / Jump to / Returns

Hi, I am Recruitio a chatbot, What can I call you by?

Above is the dialog section of watson assistant which is universal for all domain skillsets under a specific company.

4.6 Summarize results

Keeping in mind, that we need to reject the weaker ones and shortlist the stronger candidates, the final UI page is designed.This Final page has the results comparing the applicants rated scores with that of the company's requirement. Based on that, the applicant is shortlisted or rejected. The HR person has access to all the candidate's details, thus He/She can use that to connect with the selected candidates.

Code:

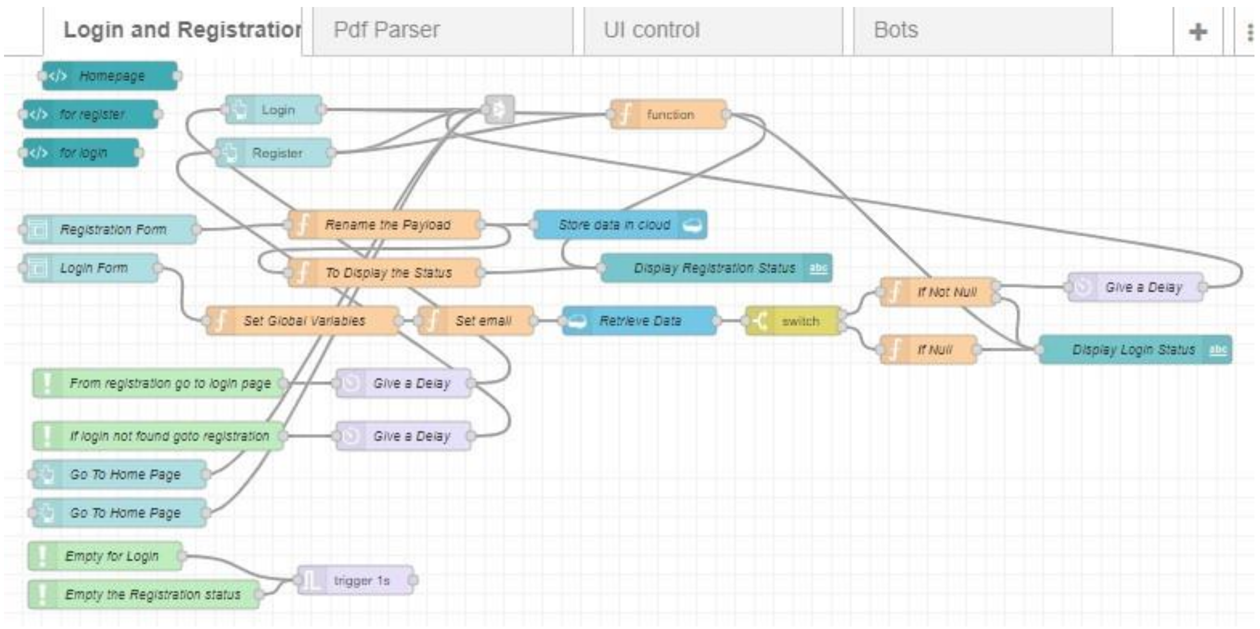
```
if(msg.payload.context.ratings === true)
{
    var companyreq = [];
    var usereq =
[msg.payload.context.basicrate,msg.payload.context.framerate,msg.payloa
d.context.dbrate,msg.payload.context.cloudrate];
    var howmany = 0;
    var n = msg.payload.context.name;
    var msg2;
    var clearconvo = {payload:"You will be directed to the Results Page
shortly!"};
    for(let i = 0;i<4;i++)
    {
        companyreq.push(Math.floor(Math.random() * 3) + 3);
    }
    for(let i=0;i<4;i++)
    {
        if(usereq[i]>=companyreq[i])
        {
            howmany += 1;
        }
    }
    if(howmany >= 2)
    {
        msg2 ={payload: "Congrats! " +n+ ". You Have Been 
Shortlisted."};
    }
    else
    {
        msg2 ={payload:"Sorry, " +n+ ". Your skillset doesn't match the 
requirements."};
    }
    var m={
        "series":["Your Ratings","Company Requirements"],

"data":[[usereq[0],usereq[1],usereq[2],usereq[3]],[companyreq[0],compan
yreq[1],companyreq[2],companyreq[3]]],
        "labels":["BasicRate","FrameRate","DbRate","CloudRate"]
    };
    return 
[{payload:[m]},msg2,clearconvo,{payload:{"group":{"hide":["Recrutio_Hom
e_page","Recrutio_Login","Recrutio_Registration","Recrutio_File_upload"
,"Recrutio_DS","Recrutio_WB","Recrutio_GD","Recrutio_BC"],"show":["Recr
utio_Final"]}}}];
}
```

5. FLOWCHART

We have built 4 node red flows. Those are:

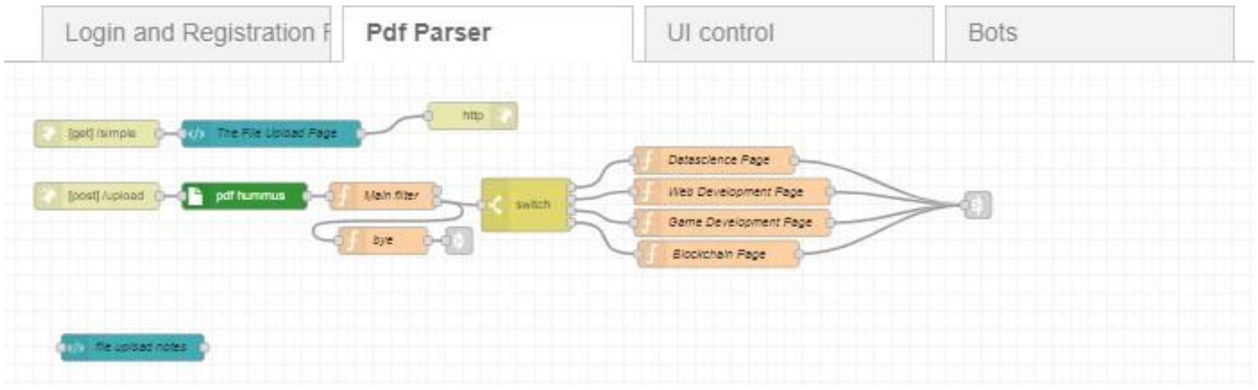
- Home page and login-registration UI
- PDF upload page UI
- UI Control
- Assistant page UI

Home page and login-registraion UI



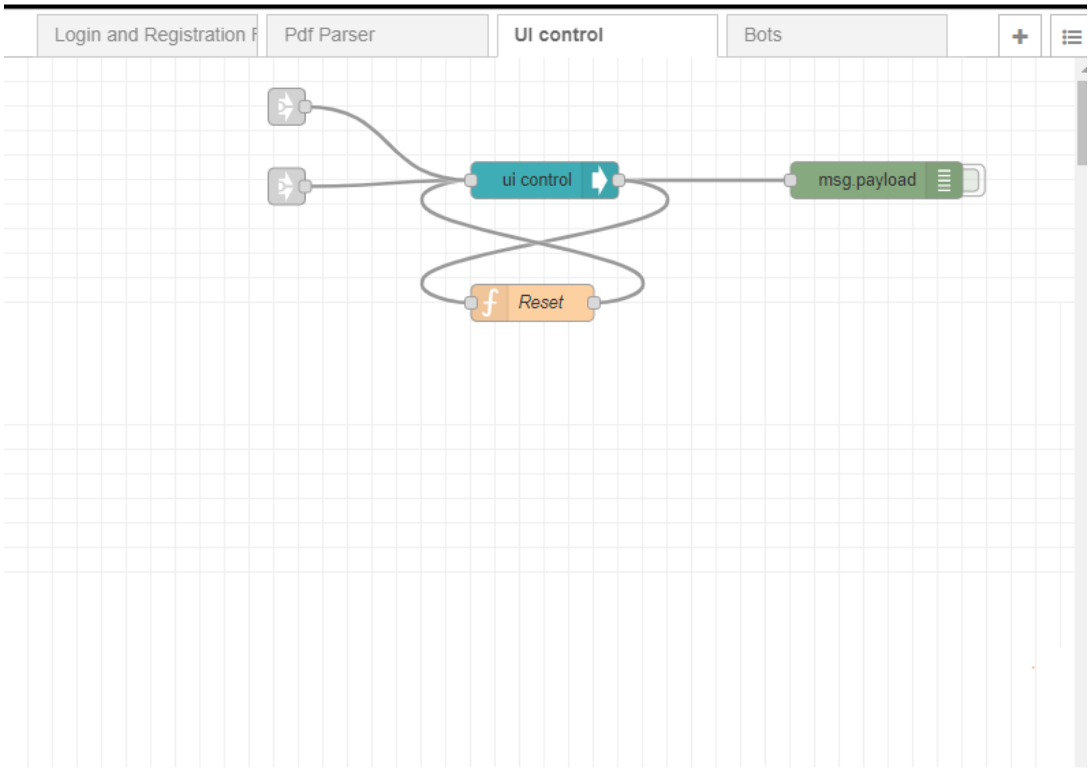
An appealing home-page with login registration page made with node red nodes like button, delay, function, debug, input and other things.

PDF upload page UI



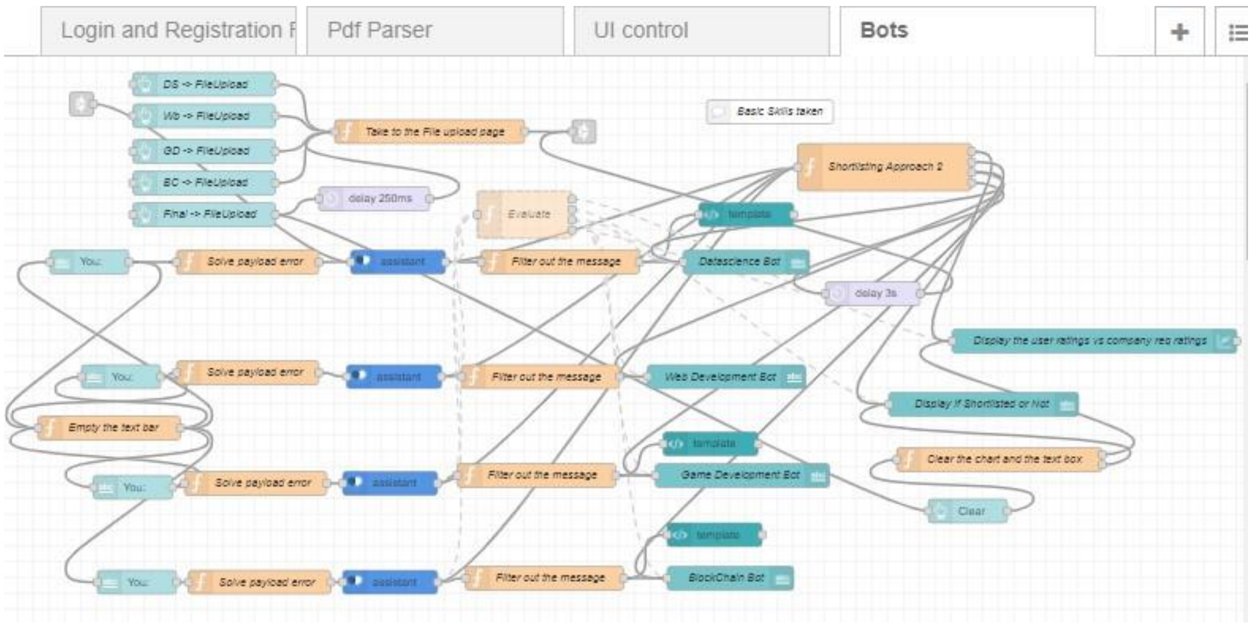
We have used hummus to parse user uploaded pdf and then hard coded javascript to extract user skills.

UI control flow



The above flow lets us jump to and fro from login-reg back to home and from last summarize page to file upload home page and so on.

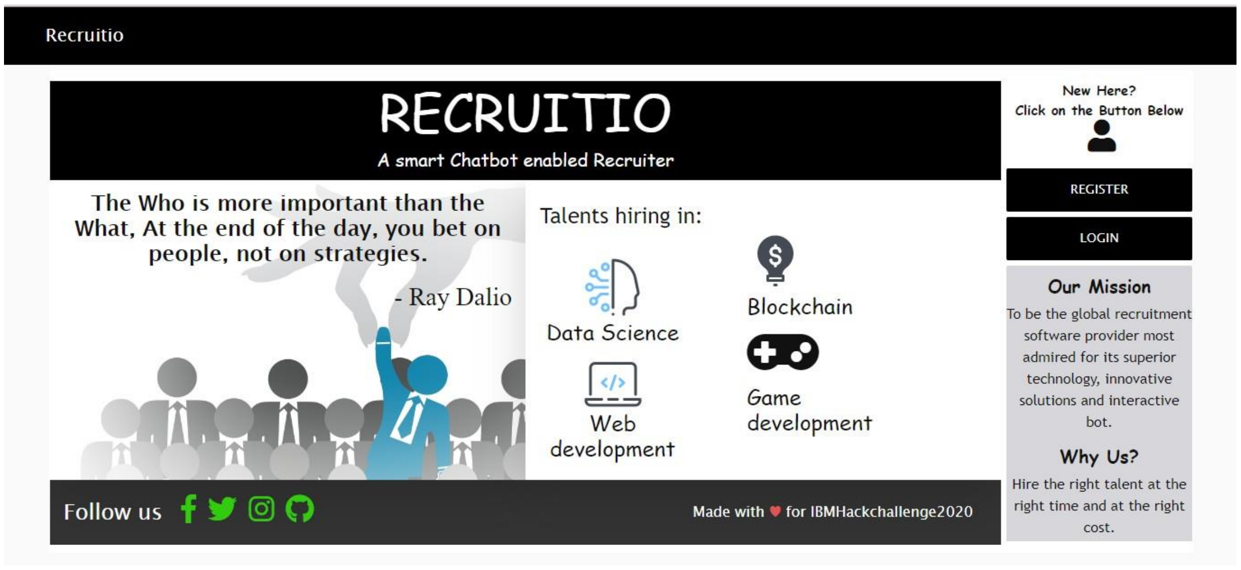
Bots/Assistant UI



We can see above the node red flow for assistants of respective domains configured to work with user uploaded resume, adding some node red dashboad ui elements to enhance the look.

6. Results

This is the main Homepage of the website, with login and register buttons and job/internship opportunities available in the firm.


Recruitio

RECRUITIO

A smart Chatbot enabled Recruiter

The Who is more important than the What, At the end of the day, you bet on people, not on strategies.

- Ray Dalio

Talents hiring in:

Data Science

Blockchain

Web development

Game development

New Here?
Click on the Button Below

REGISTER

LOGIN

Our Mission

To be the global recruitment software provider most admired for its superior technology, innovative solutions and interactive bot.

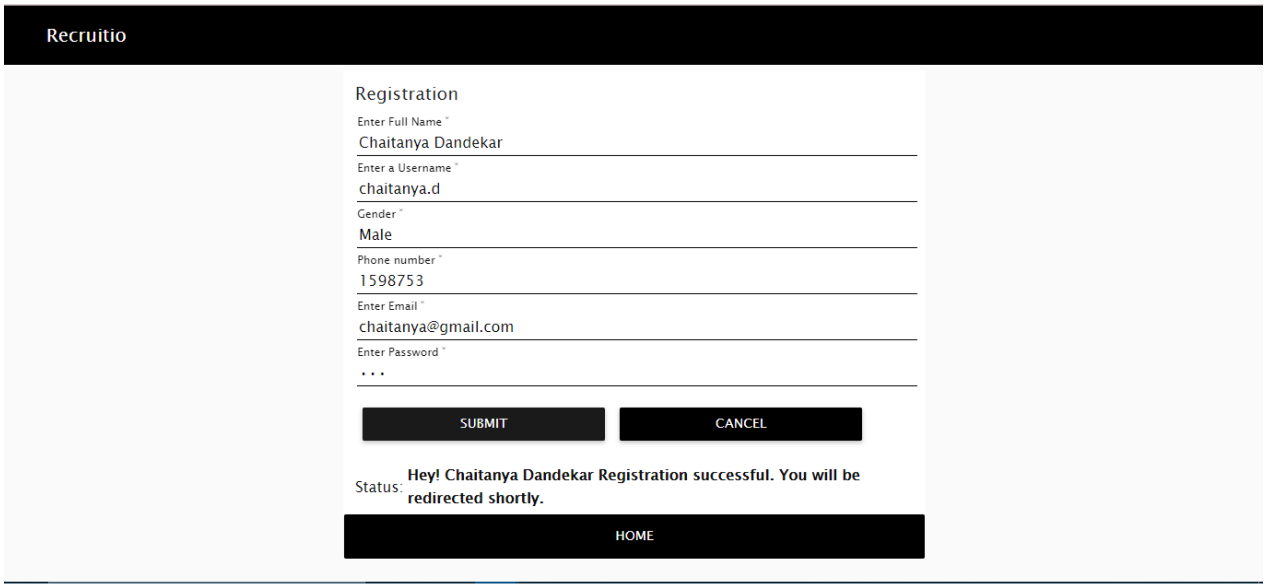
Why Us?

Hire the right talent at the right time and at the right cost.

Follow us

Made with ♥ for IBMHackchallenge2020

This snip is of the New user Registration page that intakes personal information, that can be used for login later on.

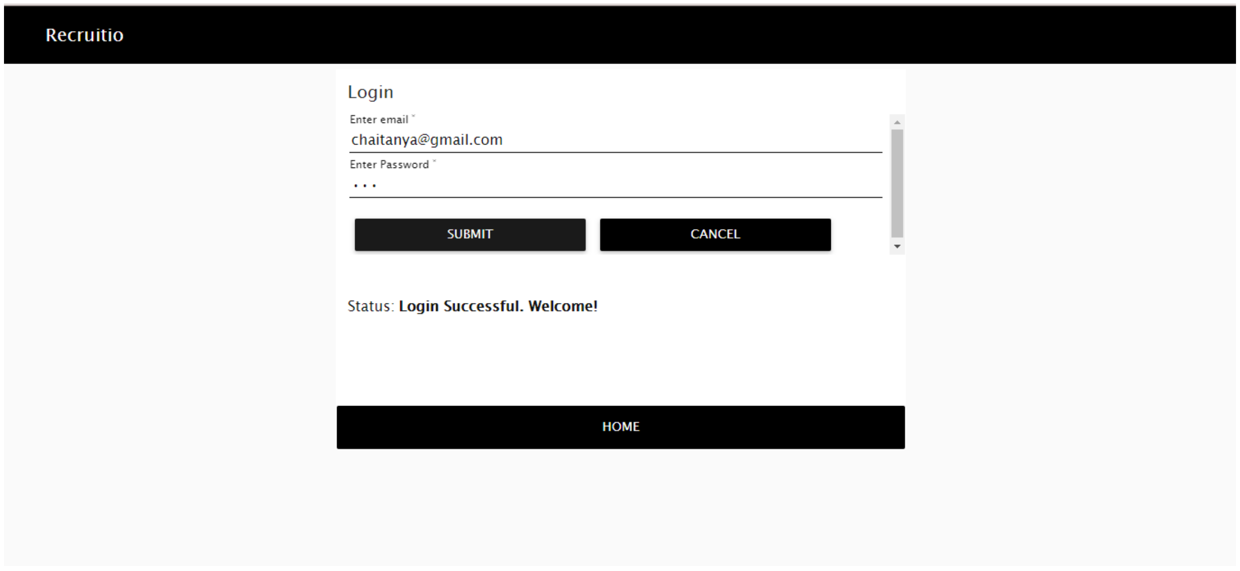


Enter Full Name *
Chaitanya Dandekar
Enter a Username *
chaitanya.d
Gender *
Male
Phone number *
1598753
Enter Email *
chaitanya@gmail.com
Enter Password *
...

Status: **Hey! Chaitanya Dandekar Registration successful. You will be redirected shortly.**

When clicked on Submit it takes the user to the login page, as shown below.

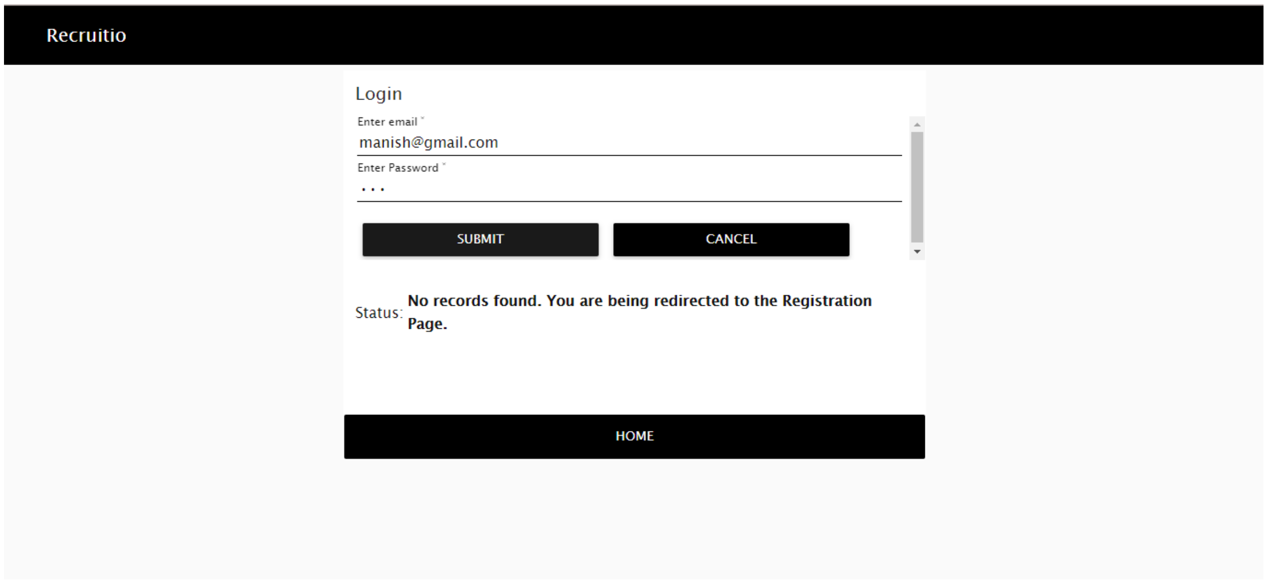
This is the Login page that acknowledges if the login was successful, and throws an error message, if wrong password or email is detected,
In such cases, the user is then redirected to the registration page, since such ID was not detected in the database.
The following 2 snips illustrates that.



Enter email *
chaitanya@gmail.com
Enter Password *
...

Status: **Login Successful. Welcome!**

Here, Manish(an unregistered user) tried a login attempt, is later on redirected to the Registration page.


Recruitio

Login

Enter email *
manish@gmail.com

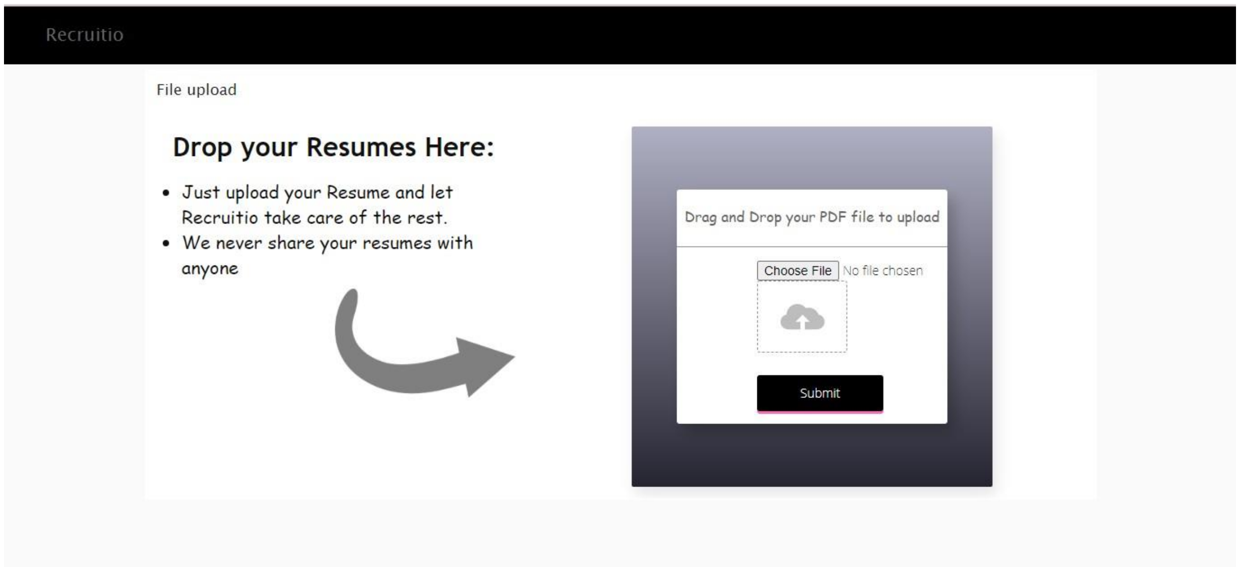
Enter Password *
···

Status: **No records found. You are being redirected to the Registration Page.**

Once a user is succesfully logged in he/she is then expected to upload their resume on the following page, as shown below.

This page has both upload options:

1. Drag and Drop.
2. Click to choose file.


Recruitio

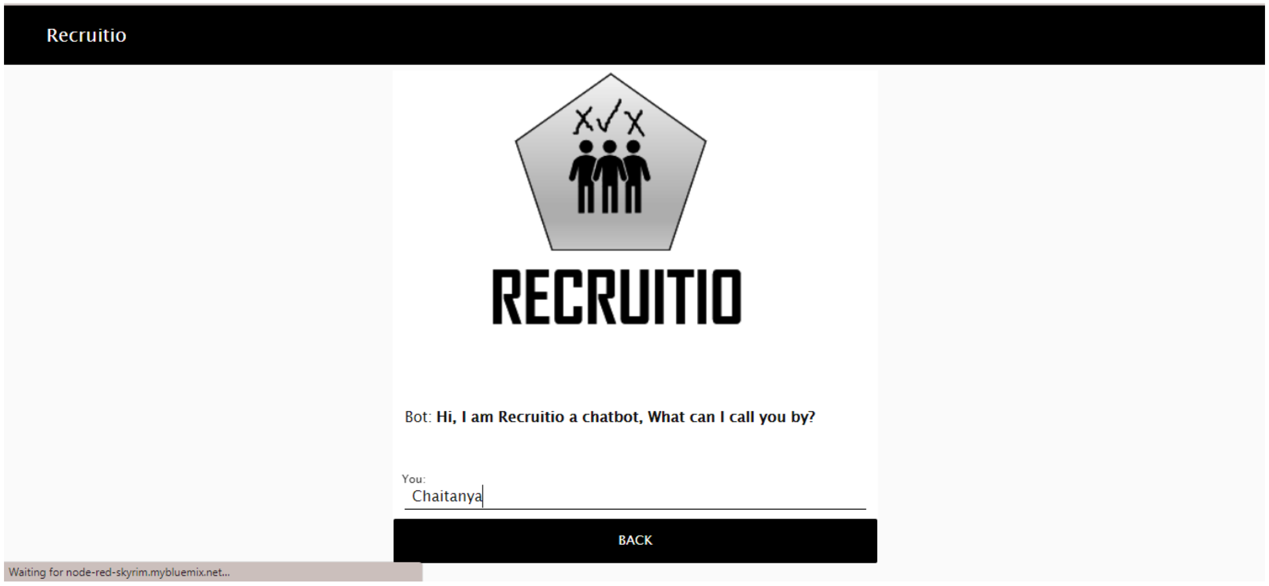
File upload

Drop your Resumes Here:

- Just upload your Resume and let Recruitio take care of the rest.
- We never share your resumes with anyone

Drag and Drop your PDF file to upload

Once the Submit button is clicked the PDF file of the Applicant's Resume is parsed through the PDF parser, which was explained above, and the user is then directed to the particular chatbot of the corresponding domain.



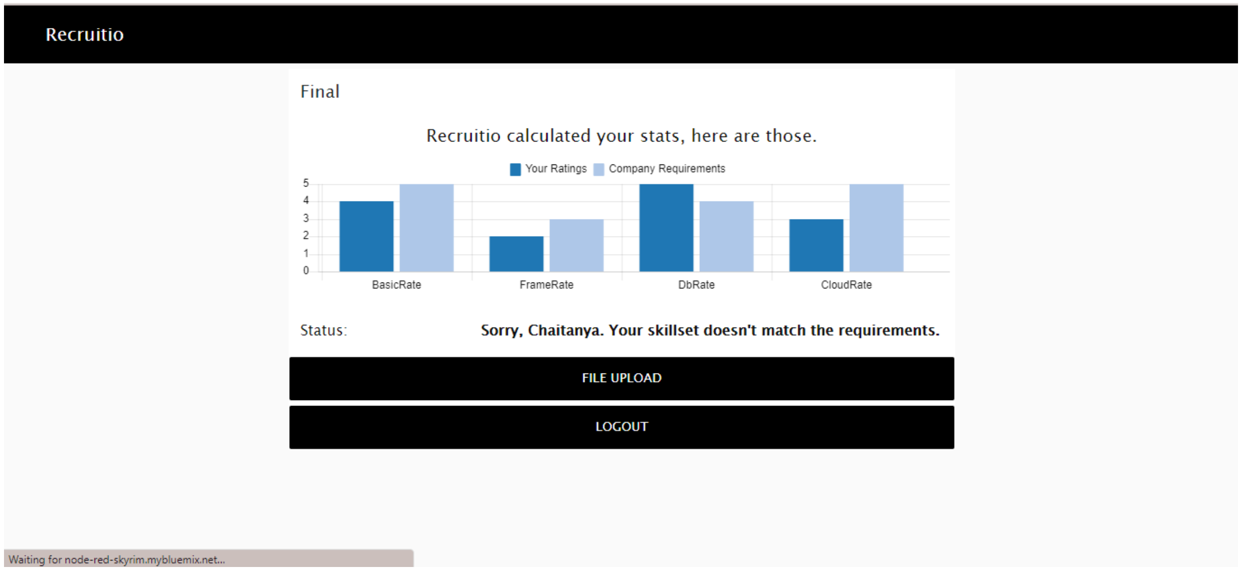
Bot: **Hi, I am Recruitio a chatbot, What can I call you by?**

You:
Chaitanya

Waiting for node-red-skyrim.mybluemix.net...

Here onwards the applicant interacts with the Chatbot, and the chatbot extracts critical information from the applicant by asking him/her to rate themselves on technical skills.

This is Final page of the UI that shows the chart with comparison, of the applicants skill and the company's requirement.



Recruitio calculated your stats, here are those.

Status: **Sorry, Chaitanya. Your skillset doesn't match the requirements.**

Waiting for node-red-skyrim.mybluemix.net...

7. ADVANTAGES & DRAWBACKS

- **Advantages:**
  - Companies can deploy chatbots to enchance their recruiting process.
  - Reduces man power and much effective.
  - Cost efficient.
  - Works well with multiple production systems
  - Easy file uploads
  - Much interactive user friendly chat
  - Applicable in vast variety of domains

- **Disadvantages:**
  - May tamper performance at extreme large scale environments
  - UI/UX may tamper too in such cases.

8. APPLICATION

Hiring becomes more productive when a personal reference is attached. Recruiting softwares let us track all the employee referrals and identify the social quotient of employees in any organization. With the help of bulk upload mechanism, recruiters save a lot of time to source multiple profiles. It lets you organize candidate database from all sources, effective resume parsing and a lot more for sleek and user friendly interaction of candidates with the HR of a company.

9. CONCLUSION

The future of recruitment through technology has a direct & positive impact on any organization. But as things change more, the classic challenges of recruitment remain including how to find, attract, screen, and shortlist candidates often the most challenging and time-consuming step in the recruitment process. As a result Recruitment chatbots were invented to tackle the problem. But the early chatbots created wouldnt do any complex tasks. Therefore to solve the issue AI chatbots were created, this projects aims to solve that issue. The project was created through Node -Red and with the help of cloudant for storage and watson Assistant for the chatbot we evaluated the user's skills and shortlisted them based on requirements.

10. FUTURE SCOPE

Things we can add :

- Replace manual sourcing with an extension
- A quick feedback/reminder mechanism to alert candidates as well as hiring peeps so as not to lose deserving candidates.
- More interactive UI/UX
- Enhancing for multiple production enviroments.

11. REFERENCES

- https://github.com/ibmets/node-red-bluemix-starter
- https://github.com/garyrwilson/Watson-Assistant-Labs
- https://github.com/garyrwilson/Watson-Cognitive-Labs
- https://github.com/iyzico/recruitment
- https://github.com/opencats/OpenCATS
- https://beamery.com/blog/sourcing-developers-on-github