

PROJECT REPORT

Topic: Intelligent Customer Help Desk With Smart Document Understanding

Author: Vivek Mudaliar

Domain: Artificial Intelligence

Email-ID: vm4713@srmist.edu.in

Internship: Smartbridge(Remote Internship Program)

CONTENTS

1. INTRODUCTION

1.1 Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 Proposed solution

3. THEORITICAL ANALYSIS

3.1 Block diagram

3.2 Hardware / Software designing

4. EXPERIMENTAL INVESTIGATIONS

5. FLOWCHART

6. RESULT

7.ADVANTAGES & DISADVANTAGES

8.APPLICATIONS

9.CONCLUSION

10.FUTURE SCOPE

11.BIBILOGRAPHY

12.APPENDIX

12.1. Cloud Function Action code

12.2. Node-red Flow json

1. Introduction

1.1 Overview

A chatbot is an artificial intelligence (AI) software that can simulate a conversation (or a chat) with a user in natural language through messaging applications, websites, mobile apps or through the telephone. Why are chatbots important? A chatbot is often described as one of the most advanced and promising expressions of interaction between humans and machines. However, from a technological point of view, a chatbot only represents the natural evolution of a Question Answering system leveraging Natural Language Processing (NLP). Formulating responses to questions in natural language is one of the most typical Examples of Natural Language Processing applied in various enterprises' end-use applications.

Many high-tech banking organizations are looking to integrate automated AI-based solutions such as chatbots into their customer service in order to provide faster and cheaper assistance to their clients who are becoming increasingly comfortable with technology. In particular, chatbots can efficiently conduct a dialogue, usually replacing other communication tools such as email, phone, or sms. In banking, their major application is related to quick customer service answering common requests, as well as transactional support.

Several studies report significant reduction in the cost of customer services, expected to lead to billions of dollars of economic savings in the next 10 years. In 2019, Gartner predicted that by 2021, 15% of all customer service interactions globally will be handled completely by AI. A study by Juniper Research in 2019 estimates retail sales resulting from chatbot-based

interactions will reach \$112 billion by 2023

Since 2016 when Facebook allowed businesses to deliver automated customer support, e-commerce guidance, content and interactive experiences through chatbots, a large variety of chatbots were developed for the Facebook Messenger platform.

1.2 Purpose

Chatbot applications streamline interactions between people and services, enhancing customer experience. At the same time, they offer companies new opportunities to improve the customers engagement process and operational efficiency by reducing the typical cost of customer service.

To be successful, a chatbot solution should be able to effectively perform both of these tasks. Human support plays a key role here: Regardless of the kind of approach and the platform, human intervention is crucial in configuring, training and optimizing the chatbot system.

2. Literature Survey

2.1 Existing Problem

There's no point delivering an engaging, technologically advanced chatbot – if it doesn't actually do anything! Your chatbot needs to have a USP to make it attractive to users, and top of the list is a chatbot that makes their lives easier in some way. That could be supplying them with instant access to personal data, giving them useful information that enriches their experience (or even makes their day better) or helping them in their work.

The technical aspects of building a chatbot will depend on what type of chatbot you want to develop. A purely text-based bot will be much easier to develop than a

voice-activated version.

For any platform, there has to be a set process that the chatbot goes through, so it's important to develop an 'intelligent' platform that:

1. Finds out the goal of the user by asking a series of questions and responding to the answers in a coherent manner
2. Collects relevant user information
3. Processes the data and uses its analysis to respond to the user's goal
4. Stores the information in a database so that if the same query arises again, it can use the information to form a more accurate response (known as machine learning).

Building chatbots that integrate within a digital system (especially if it's an integrated system that uses both online portals and IVR telephony) is a task that requires a considerable amount of expertise.

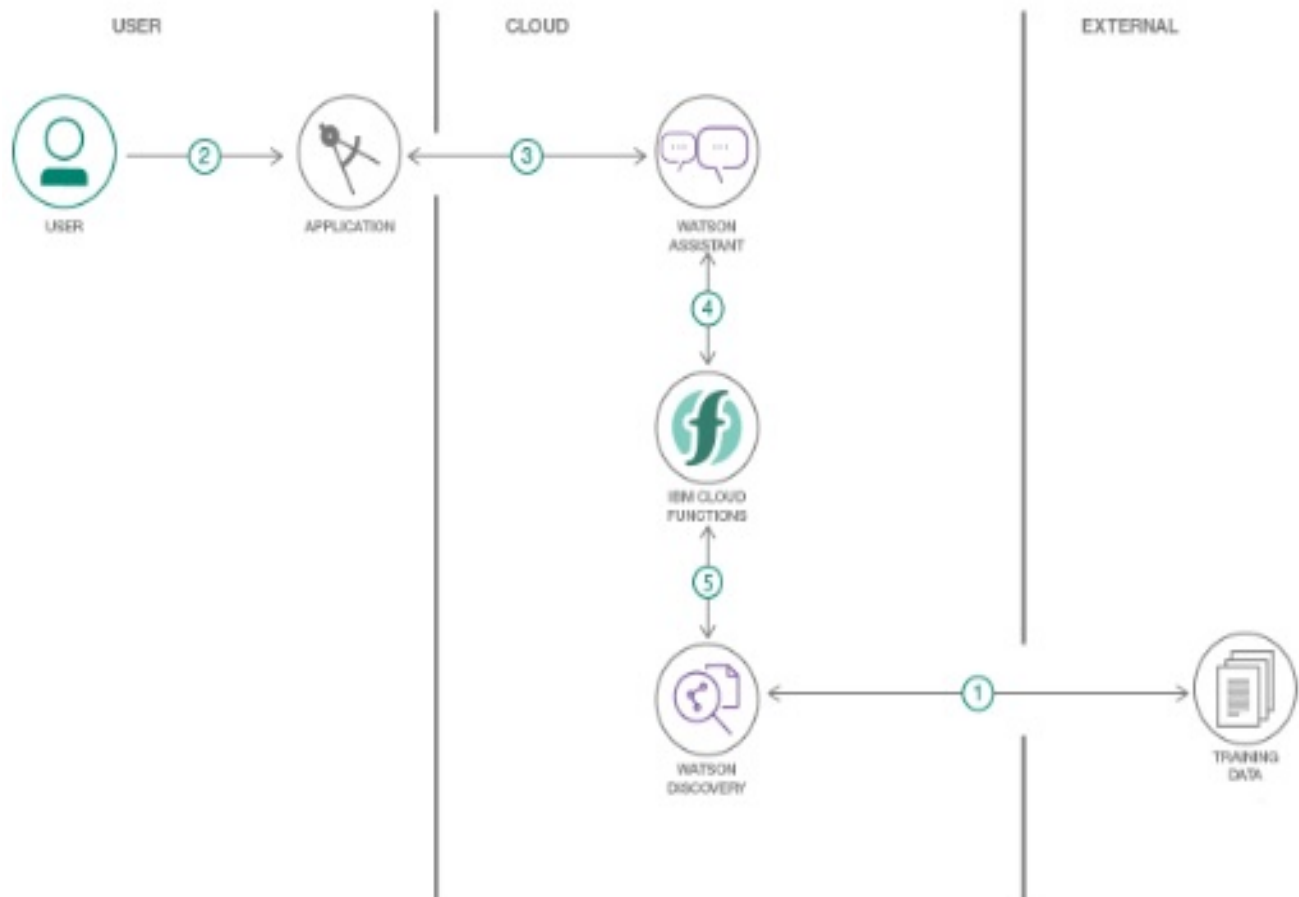
By far the best option is to work with a specialist to create a bespoke system specially adapted to your business. While there are platforms that allow you to build generic bots, they lack the nuance of tailored designs, are unlikely to provide your customers with a satisfactory experience and will be difficult to integrate into your wider channel shift strategy.

2.2 Proposed Solution

To solve the problems above we would use various IBM cloud services which would make our job easy. The chatbot would use smart document understanding with which it would get the relevant text and headings from the document which is uploaded in the IBM Discovery which would then in turn be connected with IBM Watson which would create the chatbot with necessary intents and dialogs. These both services are connected with the cloud functions which creates a url that is used by the Watson as webhooks. At last we would require a user interface so we use Node red which would display our chatbot as a website.

3. Theoretical Analysis

3.1 Block Diagram



3.2 Hardware/Software Requirements

- IBM Watson for creating the chatbot
- IBM Discovery to understand the document by the reference of which the chatbot would reply.
- IBM Cloud function to connect the the watson to the discovery using the url.
- We need to deploy the chatbot for that we would need a website which can be

built using Node-Red which would create a flow based visual.

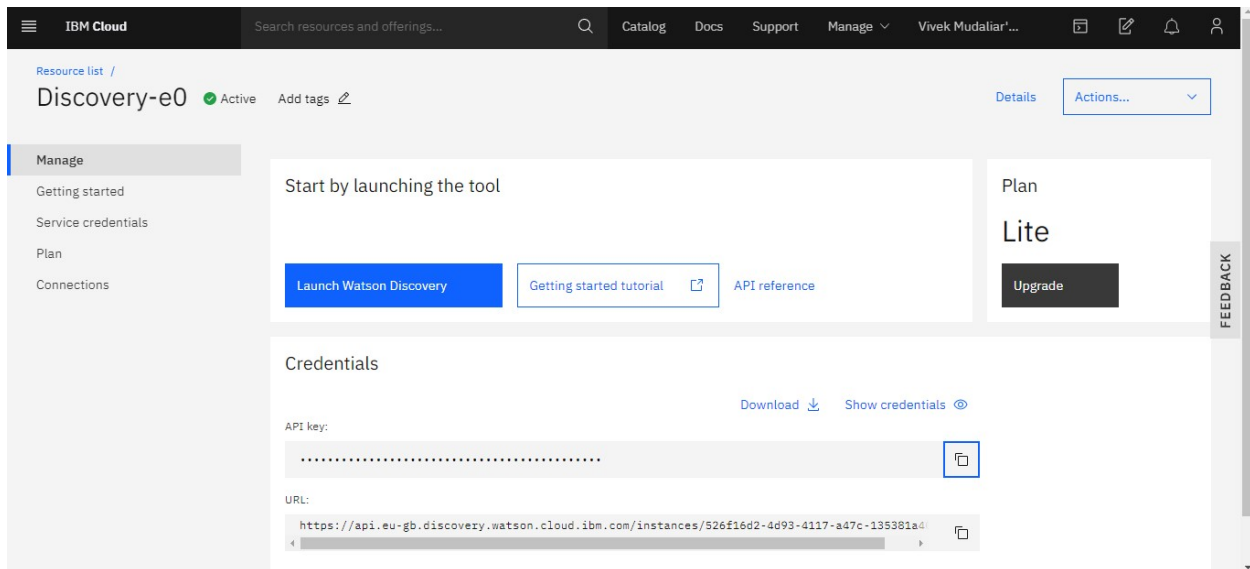
4. Experimental Investigation

1. Create an account in IBM Cloud

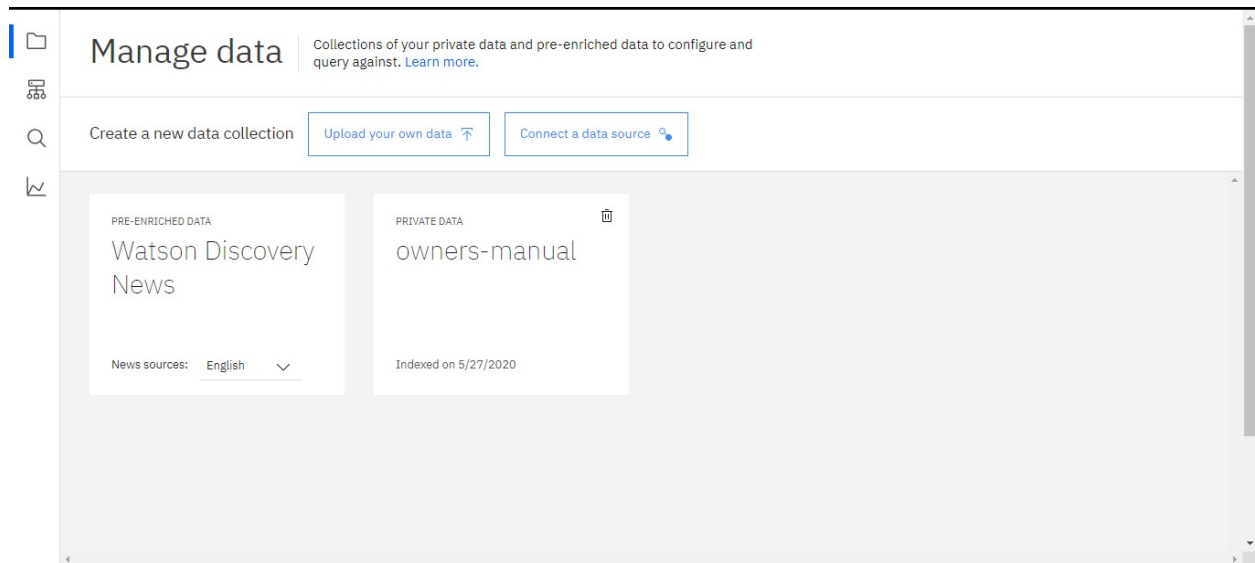
Use the services:

- IBM Watson Discovery
- IBM Watson Assistant
- IBM Cloud Function

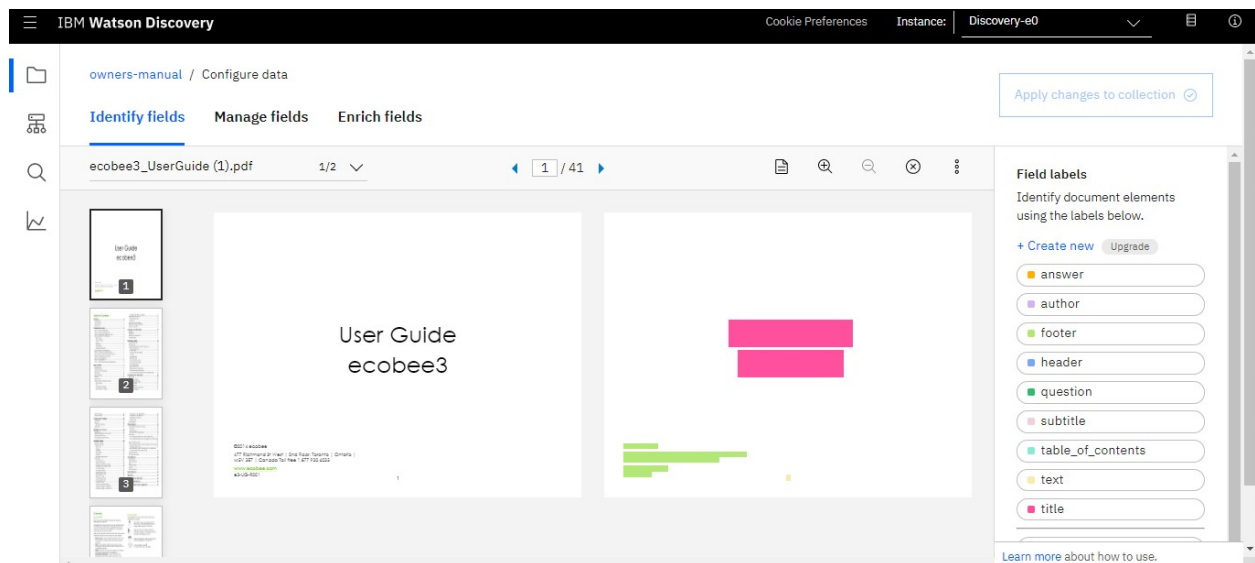
2. Create Watson Discovery



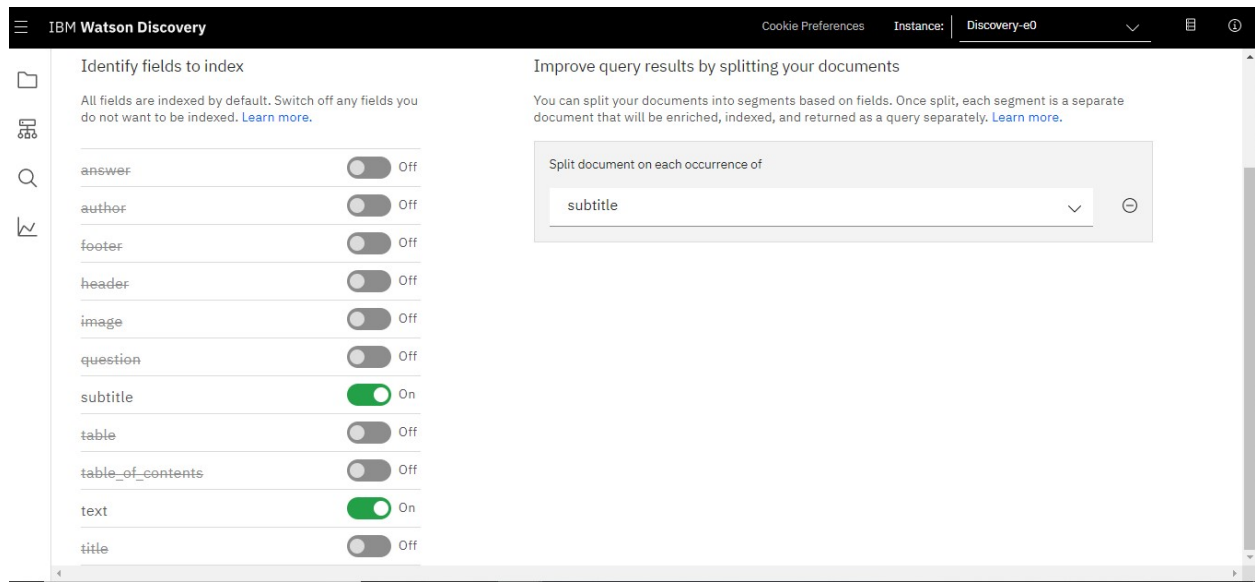
- a. Create an Private data. Here it is Owners Manual where I used upload your data button to get my discovery dataset.



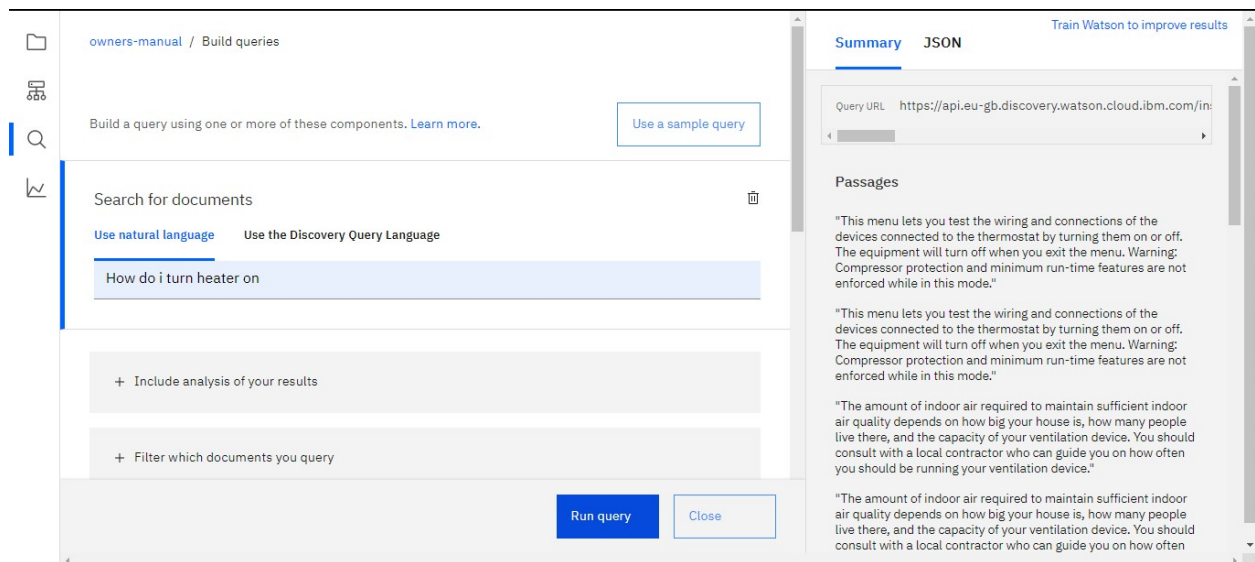
b. After uploading the User Guide for ecobee thermostat in discovery. Smart document understanding needed to be done here.]



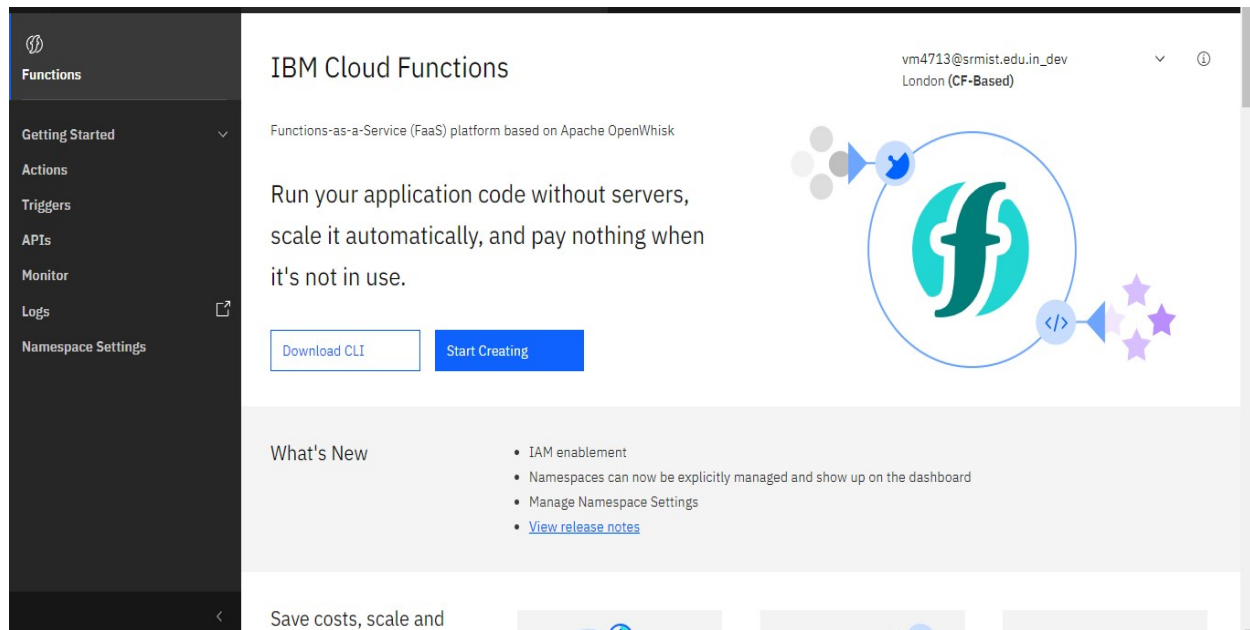
c. Managing Fields comes next where only subtitle and text are used rest are considered extra data which will not be useful in the chatbot.



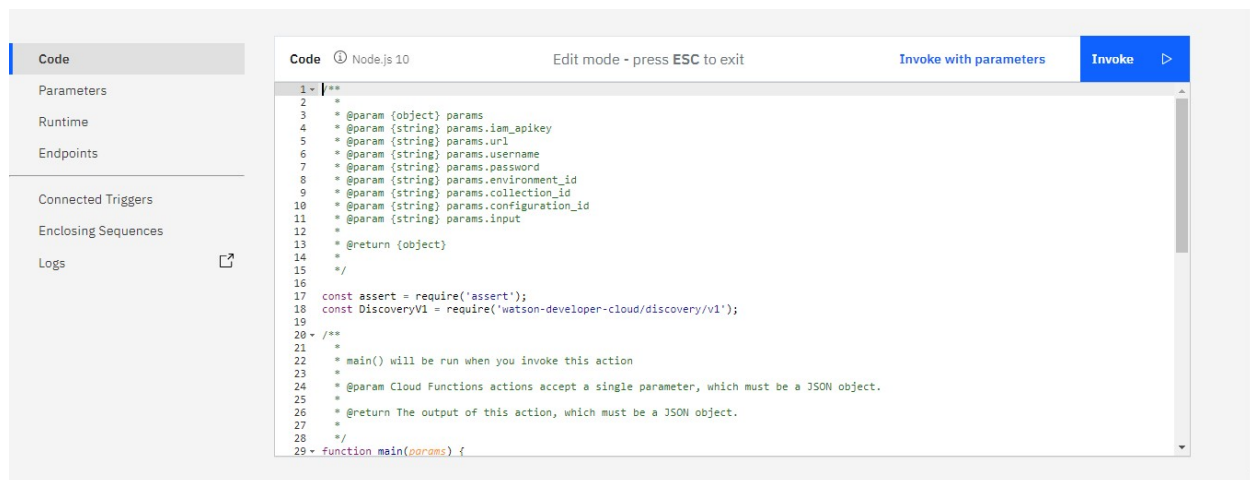
d. Run query in the discovery to check if its running correctly or not. And we can see in the screenshot that it is running



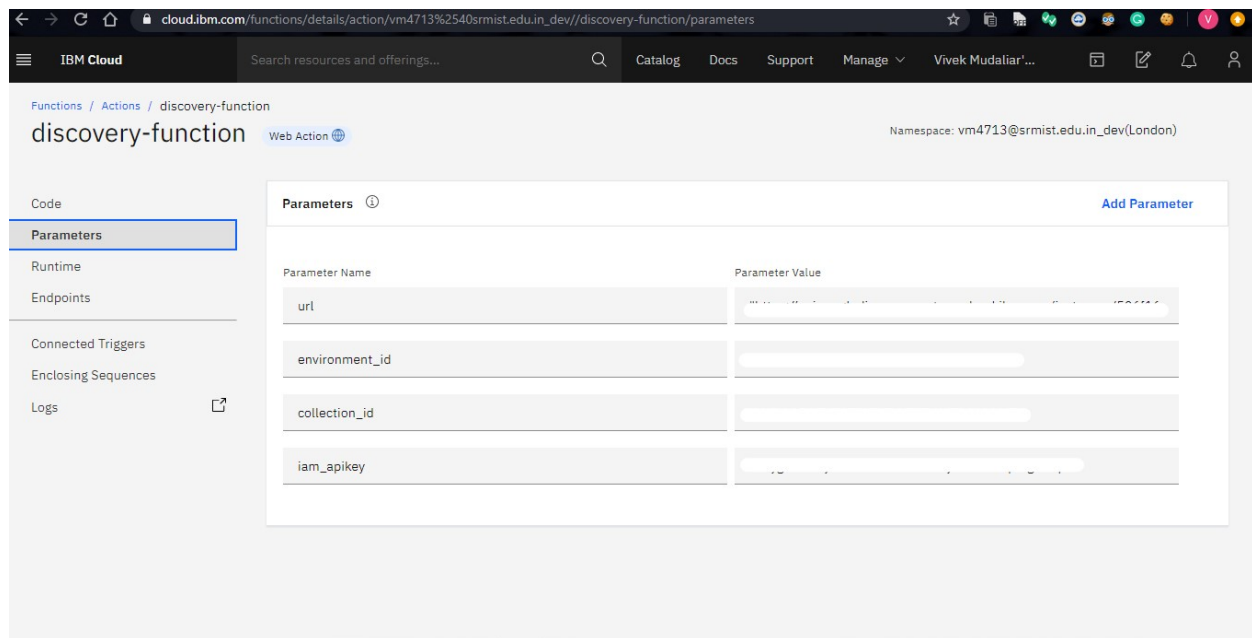
3. Create Cloud Function to connect discovery with Watson



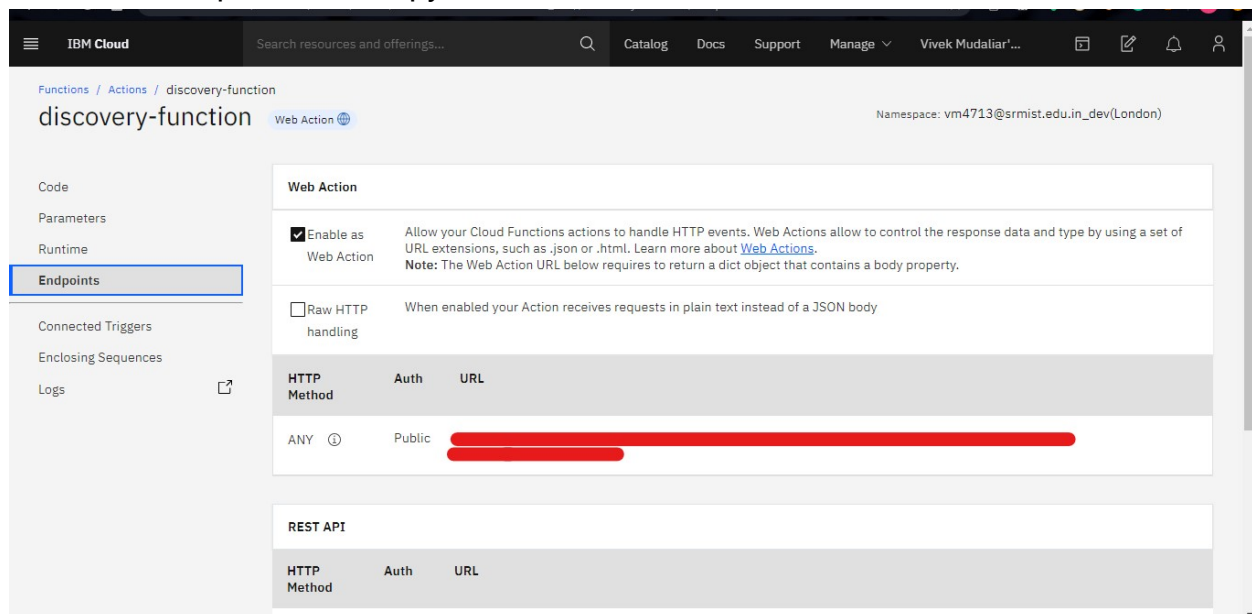
a. Write the code to activate the cloud function in the Action section.



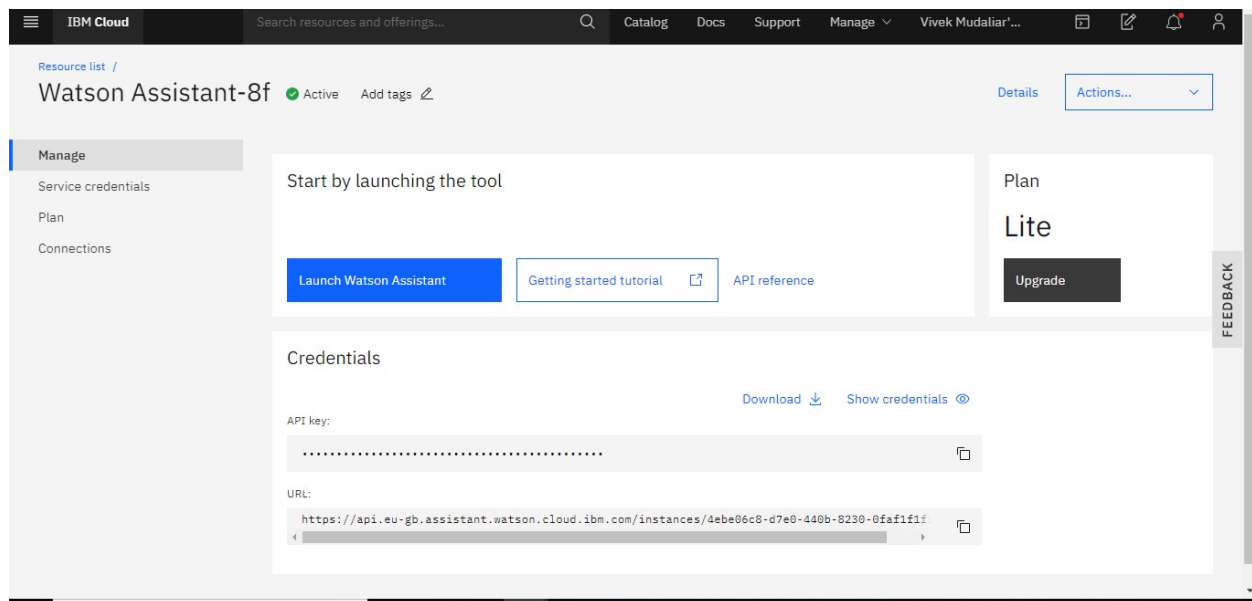
b. Give the Parameters values for web action from the discovery such as url, cloolection id environmental id and api key.



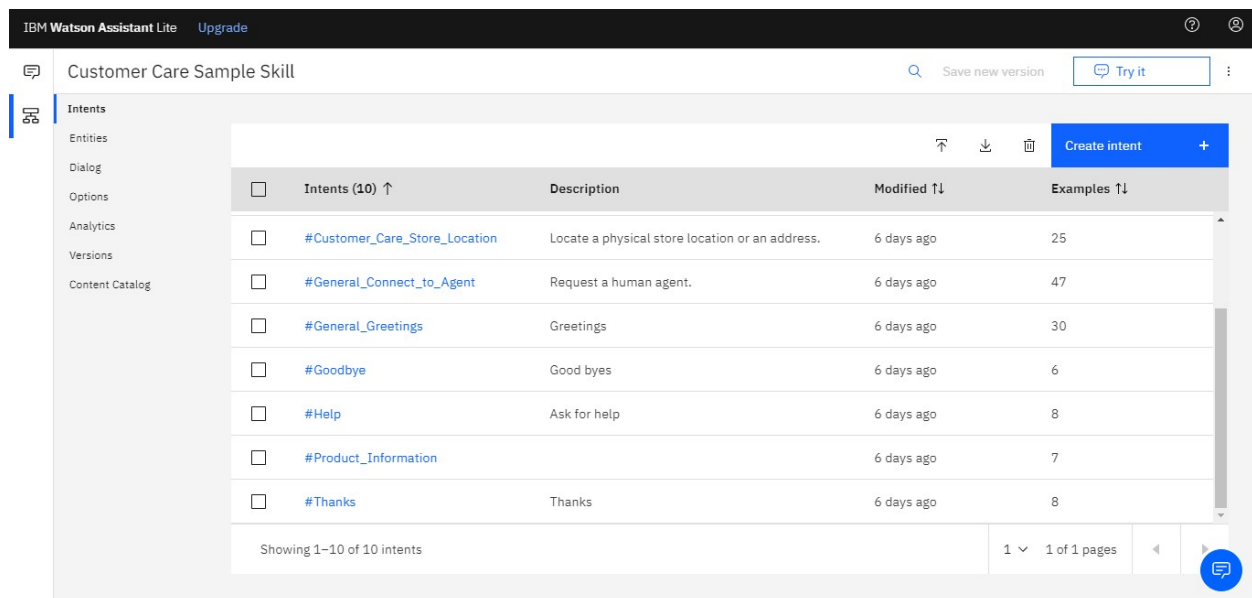
c. Click on Endpoints and copy the url from here which would be used in the Assistant.



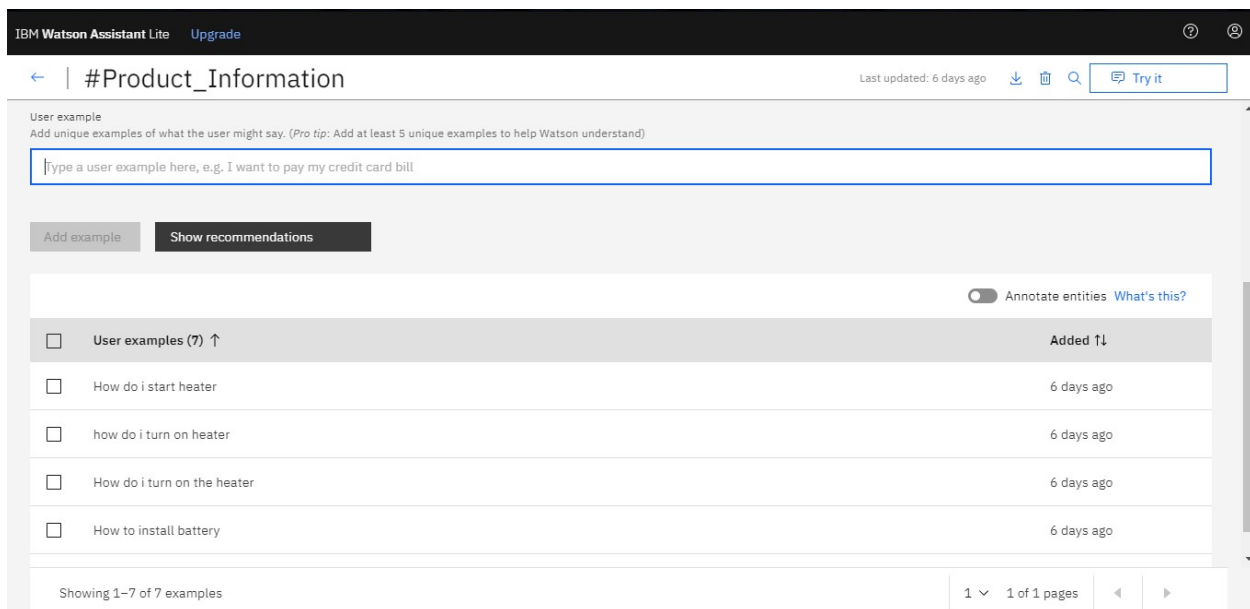
4.Launch Watson Assistant



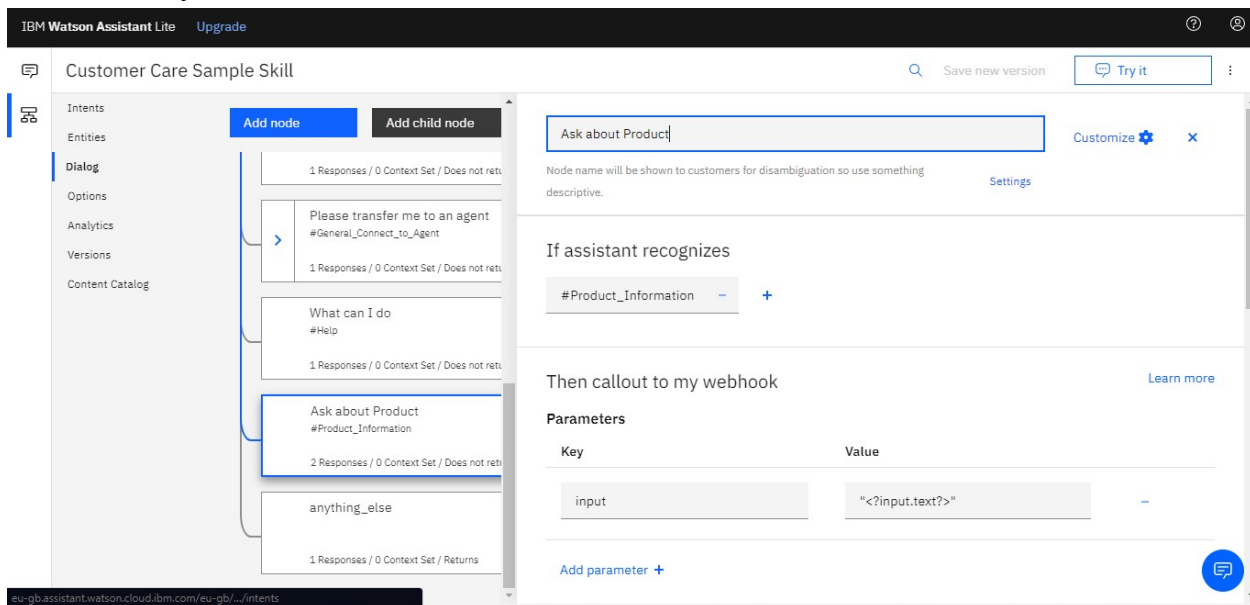
a.Create a new assistant named Customer Care Skill.



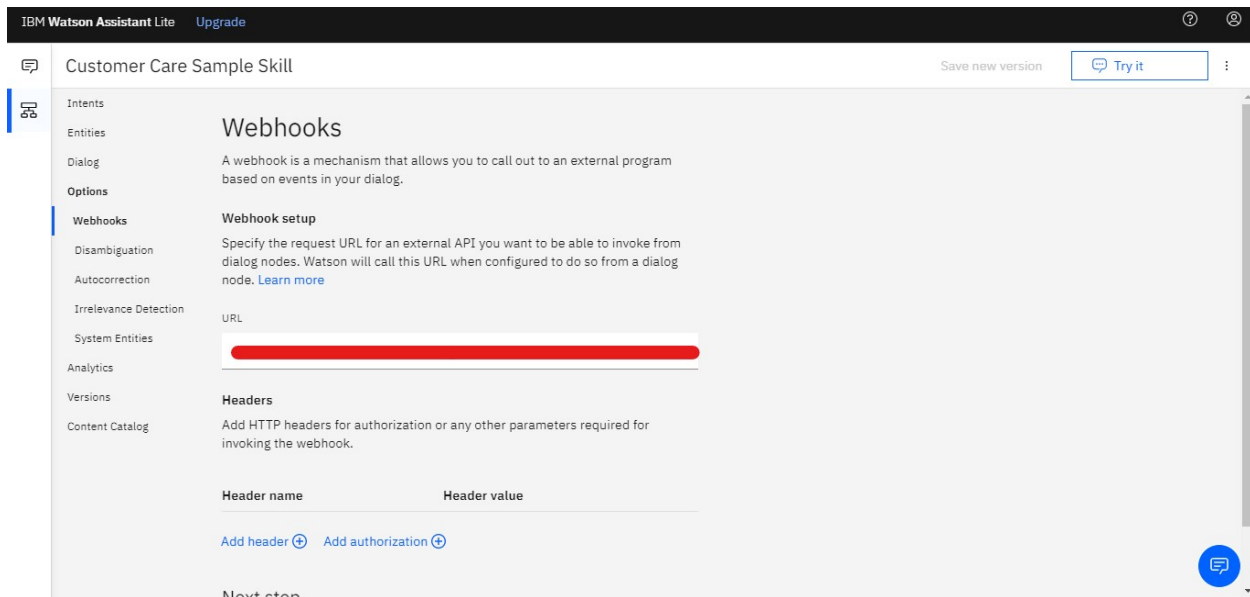
b.Create necessary Intent which here is Product_Information as per the discovery dataset such as how do I start heater, how to change settings, how to change time etc.



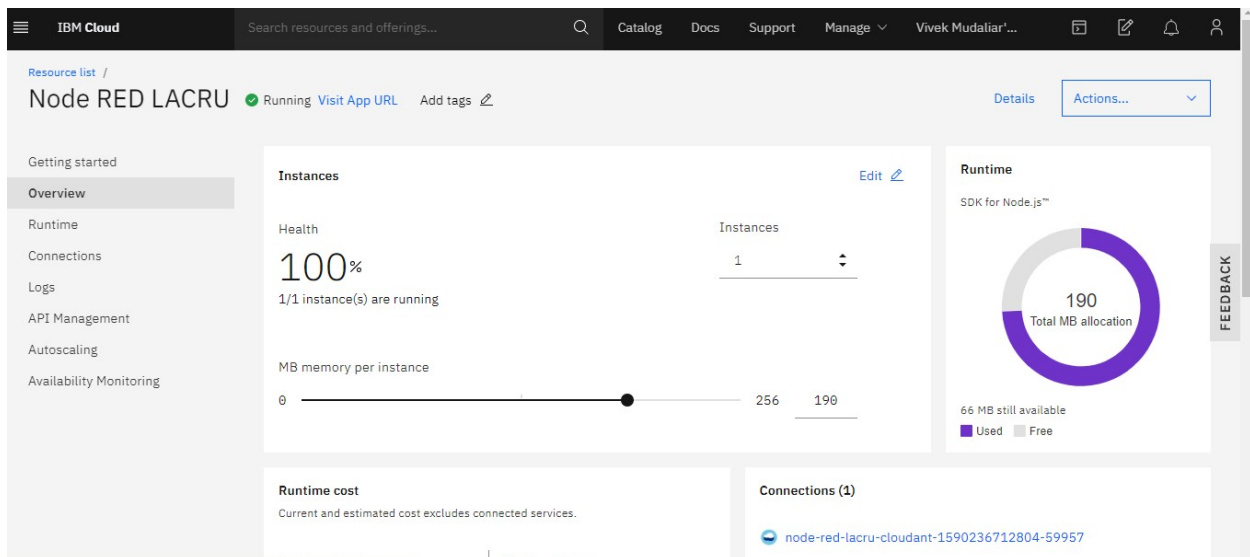
c. Create a Dialog for that Product_Information Intent which then should be connected to the discovery.



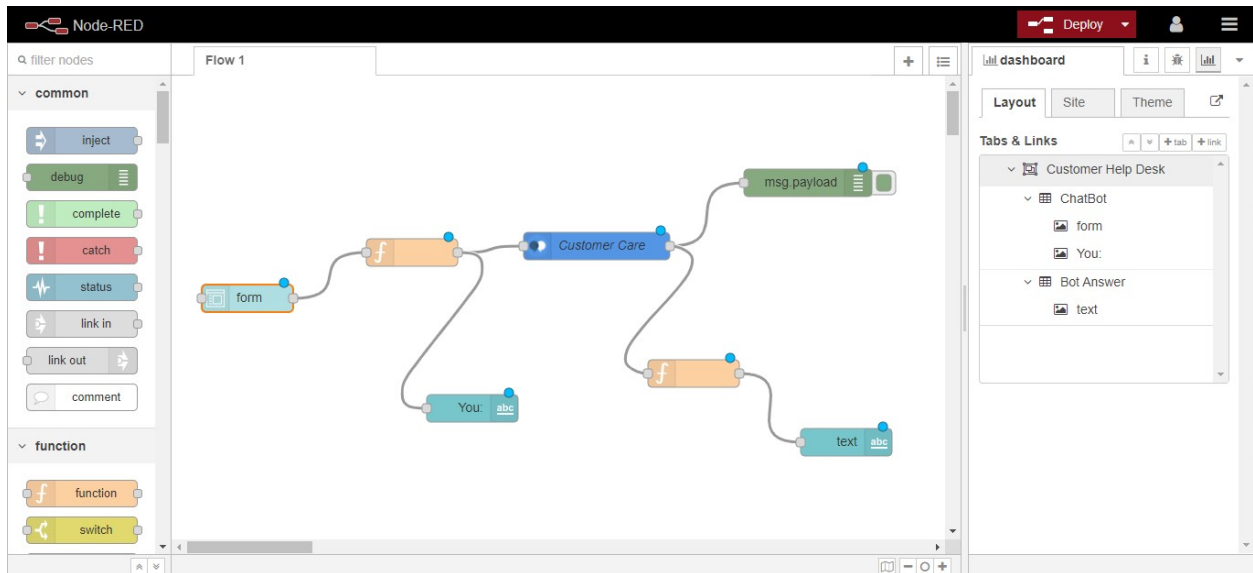
d. Turn on the Webhooks option in this dialog and paste the url from the cloud functions.



5. Create Node-Red Flow



5. Flowhart



6. Results

These are the results of the Chatbot.

Customer Help Desk

ChatBot

Enter your Input

Hi

SUBMIT

CANCEL

You: **Hi**

Bot Answer

Hello. Good afternoon

Customer Help Desk

ChatBot

Enter your Input

how to turn on heater

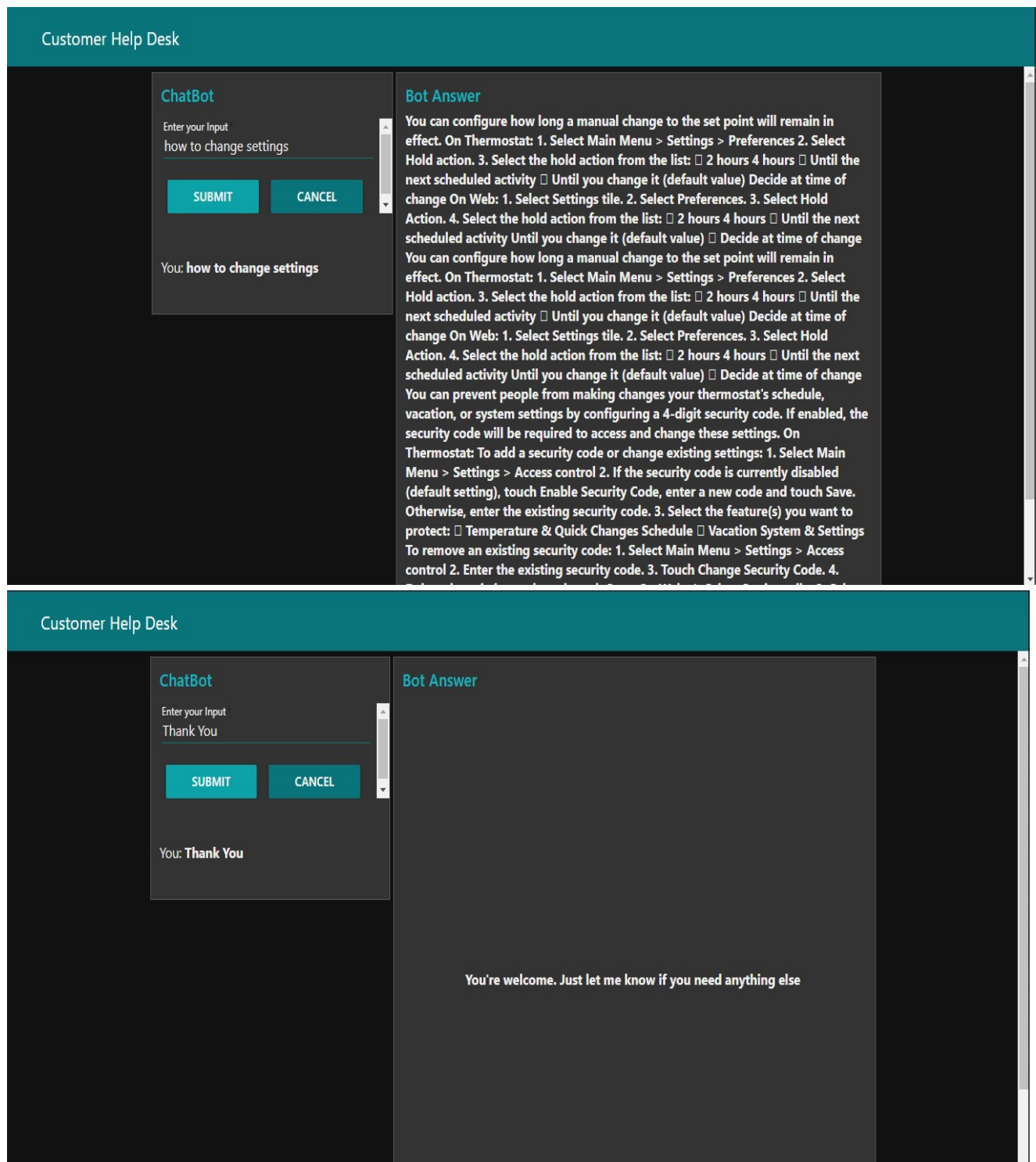
SUBMIT

CANCEL

You: **how to turn on heater**

Bot Answer

If you have a furnace or boiler installed: 1. Select the heating menu. 2. Configure the heater type: ☐ Furnace: Optimizes ecobee3 for systems using forced air ☐ Boiler: Optimizes your ecobee3 for systems using radiators or in-floor heat. 3. Touch Next. You will be returned to the Equipment configuration menu. If you have a furnace or boiler installed: 1. Select the heating menu. 2. Configure the heater type: ☐ Furnace: Optimizes ecobee3 for systems using forced air ☐ Boiler: Optimizes your ecobee3 for systems using radiators or in-floor heat. 3. Touch Next. You will be returned to the Equipment configuration menu. This menu lets you test the wiring and connections of the devices connected to the thermostat by turning them on or off. The equipment will turn off when you exit the menu. Warning: Compressor protection and minimum run-time features are not enforced while in this mode.



The link for this Node red application is as follows

https://node-red-lacru.eu-gb.mybluemix.net/ui/#!/0?socketid=Vs_Ov39x5wEd6RoTAA4

7. Merits and Demerits

Merits

- Good for small size questions and queries and requires a small team to develop
- Less Cost in development
- Maintenance of the chatbot is low and manageable.

Demerits

- Smart Documents Understanding should be done properly to get good results sometimes unwanted data is also displayed.
- Some questions might not be answered.
- Sometimes wrong answers are given if discovery dataset is not proper.

8. Applications

Unlike humans who can only communicate with one human at a time, chat bots can simultaneously have conversations with thousands of people. No matter what time of the day it is or how many people are contacting you, every single one of them will be answered immediately.

Imagine you own a restaurant, and you have a good reputation for your food of which most of your revenues come from delivery. As the demand keeps rising, you will have more customers to take orders from but very few staff to attend them all. Having a chatbot would eliminate such problem and cater to each and every person and ensure that no order is missed. Companies like Taco Bell and Dominos are already using chatbots to arrange delivery of parcels.

Chatbots have the benefit that it can quite easily be used in any industry. Unlike other products where you have to do a lot of development and testing to change platforms, chatbots are relatively easy to switch. One has to just train the bot by giving the right conversation structure and flow to switch its current field or industry.

Or if there is a lot of back and forth between two sections of the industry say customer support and sales, then you could have custom built presets which would already have the conversation flow and structure to carry out the interactions with the user.

Humans are bound to change of emotions. Chatbots, on the other hand, are bound by some rules and obey them as long as they're programmed to. They will always treat a

customer in the perfect way no matter how rough the person is or how foul language the person uses.

Not everyone orders the same food everyday, people's choices may change everyday. In this case, it can use your order history to make suggestions for the next order, learn your address details and much more.

9. Conclusion

The followings steps are the process of the creation of this chat bot which included the services such as IBM Watson, IBM Cloud Function, IBM Watson Discovery and Node-red. Thus a Chatbot is successfully created which gives us answer from the dataset that is uploaded in the discovery and trained using smart document understanding which is the ecobee user guide.

Node-Red Link:

https://node-red-lacru.eu-gb.mybluemix.net/ui/#!/0?socketid=Vs_Ov39x5wEd6RoTAA4

Github Link:

<https://github.com/VivekM20/Intelligent-Customer-Help-Desk-With-Smart-Documents-Understanding/tree/master>

10. Future Scope

- More documents can be trained increase the range of answers which the chatbot can provide
- Improve the UI
- Can be made commercial with appropriate Improvements.

11. Bibliography

- http://www.ibm.com/cloud/architecture/tutorials/cognitive_discovery
- <https://developer.ibm.com/recipes/tutorials/how-to-create-a-watson-chatbot-on-nodered/>
- <https://cloud.ibm.com/docs/assistant?topic=assistant-getting-started>

- <http://www.iotgyan.com/learning-resource/integration-of-watson-assistant-to-node-red>
- <https://github.com/IBM/watson-discovery-sdu-with-assistant>

12. Appendix

12.1 Action for Cloud Function Code

Code:

```
/**
 *
 * @param {object} params
 * @param {string} params.iam_apikey
 * @param {string} params.url
 * @param {string} params.username
 * @param {string} params.password
 * @param {string} params.environment_id
 * @param {string} params.collection_id
 * @param {string} params.configuration_id
 * @param {string} params.input
 *
 * @return {object}
 */

const assert = require('assert');
const DiscoveryV1 = require('watson-developer-cloud/discovery/v1');

/**
 *
 * main() will be run when you invoke this action
 *
 * @param Cloud Functions actions accept a single parameter, which must be a JSON
```

object.

*

* @return The output of this action, which must be a JSON object.

*

*/

```
function main(params) {  
  return new Promise(function (resolve, reject) {
```

```
    let discovery;
```

```
    if (params.iam_apikey){  
      discovery = new DiscoveryV1({  
        'iam_apikey': params.iam_apikey,  
        'url': params.url,  
        'version': '2019-03-25'  
      });  
    }
```

```
    else {  
      discovery = new DiscoveryV1({  
        'username': params.username,  
        'password': params.password,  
        'url': params.url,  
        'version': '2019-03-25'  
      });  
    }  
  }
```

```
    discovery.query({  
      'environment_id': params.environment_id,  
      'collection_id': params.collection_id,  
      'natural_language_query': params.input,  
      'passages': true,  
      'count': 3,  
      'passages_count': 3  
    }, function(err, data) {
```

```

    if (err) {
      return reject(err);
    }
    return resolve(data);
  });
});
}

```

12.2 Node-red Flow (json)

```

[{"id":"dbb4859a.a775d8","type":"ui_form","z":"9bd0d8a9.470f08","name":"","label":"","group":
:"2994ed89.963f52","order":1,"width":6,"height":2,"options":[{"label":"Enter your
Input","value":"input","type":"text","required":true,"rows":null}], "formValue":{"input":""},"paylo
ad":"","submit":"submit","cancel":"cancel","topic":"","x":83,"y":243,"wires":[["a7e32fad.db5ad"]
]}, {"id":"2994ed89.963f52","type":"ui_group","z":"","name":"ChatBot","tab":"6fc48c16.d4617
4","order":1,"disp":true,"width":6,"collapse":false}, {"id":"6fc48c16.d46174","type":"ui_tab","z":
":"","name":"Customer Help Desk","icon":"dashboard","disabled":false,"hidden":false}]

```

Github Link:

<https://github.com/VivekM20/Intelligent-Customer-Help-Desk-With-Smart-Documents-Understanding/tree/master>

Video Demo:

https://drive.google.com/file/d/1kjmy5jrMd-Wy_N1GQ74il-vZ84sK0yNG/view

Intelligent Customer Help Desk With Smart Document Understanding

Project Summary

In this project, there will be another option. If the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owners manual. So now, instead of "Would you like to speak to a customer representative?" we can return relevant sections of the owners manual to help solve our customers' problems.

The typical customer care chatbot can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. When a question falls outside of the scope of the pre-determined question set, the option is typically to tell the customer the question isn't valid or offer to speak to a real person.

Project Requirements

- IBM Watson for creating the chatbot
- IBM Discovery to understand the document by the reference of which the chatbot would reply.
- IBM Cloud function to connect the the watson to the discovery using the url.
- We need to deploy the chatbot for that we would need a website which can be built using Node-Red which would create a flow based visual.

Functional Requirements

The Chatbot needs to get query from the user. Then it should be able to categorize the

the query according to the document which is used in discovery. Appropriate actions should be taken when a query is asked to the chatbot. The chatbot must be able to interact with the customer and answer all the queries they have related to a product or service.

Technical Requirements

IBM Cloud Services, Node Red and Python programming are the technical requirements that would be required by the application. An account in IBM Cloud is required.

Software Requirements

Softwares required by the projects are IBM Watson, IBM discovery, IBM Cloud functions, Node-Red and active internet connection.

Project Deliverables

A Chatbot which uses smart document understanding of a user guide and answers queries of a customer.

Project Team

Vivek Mudaliar- Developer

Project Schedule

This project needs to be completed before 5/6/2020.

