Project Report

on

Intelligent Customer Helpdesk with Smart Document Understanding

in

Machine Learning

by

Rajvi Dilipbhai Kapadia
rajvikapadia40@yahoo.com

Intelligent Customer Help Desk with Smart Document Understanding

1. Introduction

1.1 Overview

We will build a chatbot that uses various Watson AI Services (Watson Discovery, Watson Assistant, Watson Cloud Functions and Node-Red) to deliver an effective Web based UI through which we can chat with the assistant.

We will integrate the Watson Discovery service with Watson Assistant using webhooks.

- Project Requirements : Node-RED, IBM Cloud, IBM Watson, Node JS
- Functional Requirements : IBM Cloud
- Technical Requirements : ML, Watson AI, Node JS
- Software Requirements : Watson Assistant, Watson Discovery, Watson Cloud Functions, Node-RED
- Project Deliverables : Intelligent Chatbot with Smart Document Understanding
- Project Team : Kapadia Rajvi Dilipbhai
- Project Duration : 30 Days

1.2 Purpose

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.

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 cust omer representative?" we can return relevant sections of the owners manual to help solve our customers' problems. So unless and untill customer specifically asks for a customer representative the bot will try to solve all your queries.

To take it a step further, the project shall use the Smart Document Understanding feature of Watson Discovery to train it on what text in the owners manual is important and what is not. This will improve the answers returned from the queries. Then using Watson actions as webhook, Watson Discovery can be integrated with Watson assistant. Finally using Node-Red, Watson assistant can be integrated with a web UI. This UI can then be used to connect with Watson assistant and chat with it.

1.2.1 Scope of Work

- Create a customer care dialog skill in Watson Assistant
- Use Smart Document Understanding to build an enhanced Watson Discovery collection
- Create an IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery
- Build a web application with integration to all these services & deploy the same on IBM Cloud Platform

2. Literature Survey

2.1 Existing Problem

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.

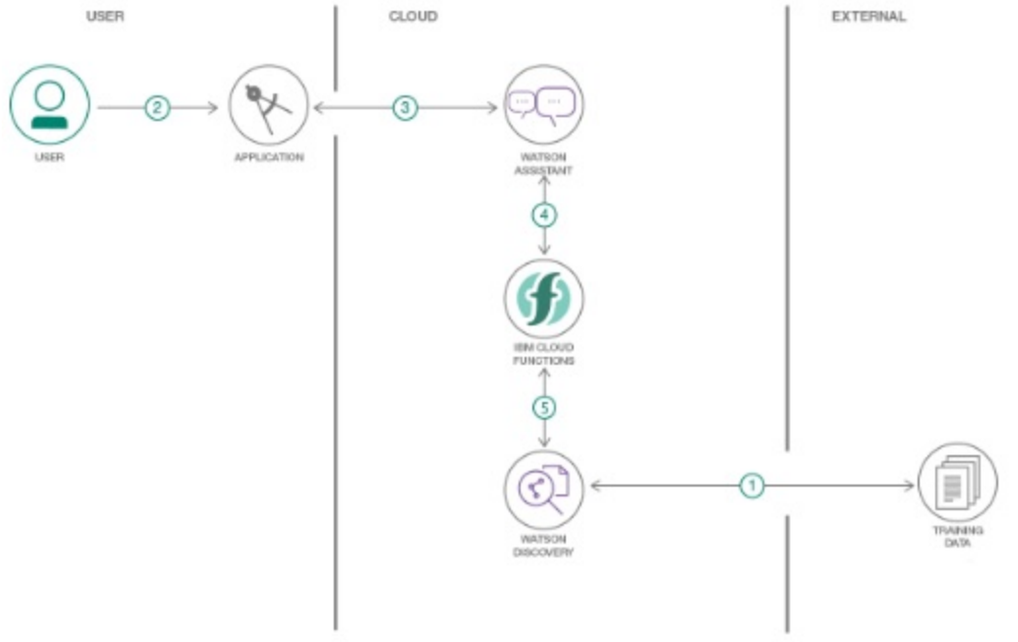
2.2 Proposed Solution

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 de vice's owners manual. So now, instead of "Would you like to speak to a cust omer representative?" we can return relevant sections of the owners manual to help solve our customers' problems. So unless and untill cust omer specifically asks for a customer representative the bot will try to solve all your queries.

To take it a step further, the project shall use the Smart Document Understanding feature of Watson Discovery to train it on what text in the owners manual is important and what is not. This will improve the answers returned from the queries. Then using Watson actions as webhook, Watson Discovery can be integrated with Watson assistant. Finally using Node-Red, Watson assistant can be integrated with a web UI. This UI can then be used to connect with Watson assistant and chat with it.

3. Theoretical Analysis

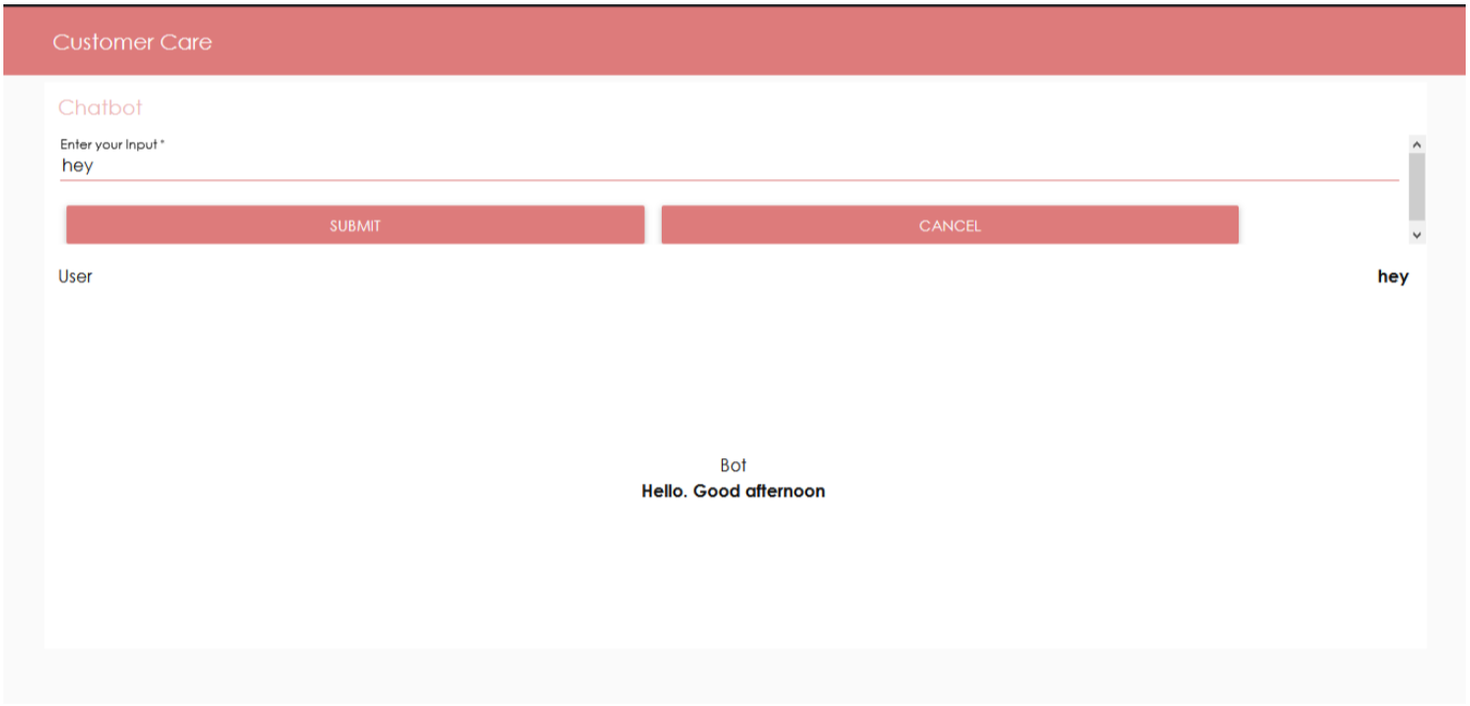
Block / Flow Diagram

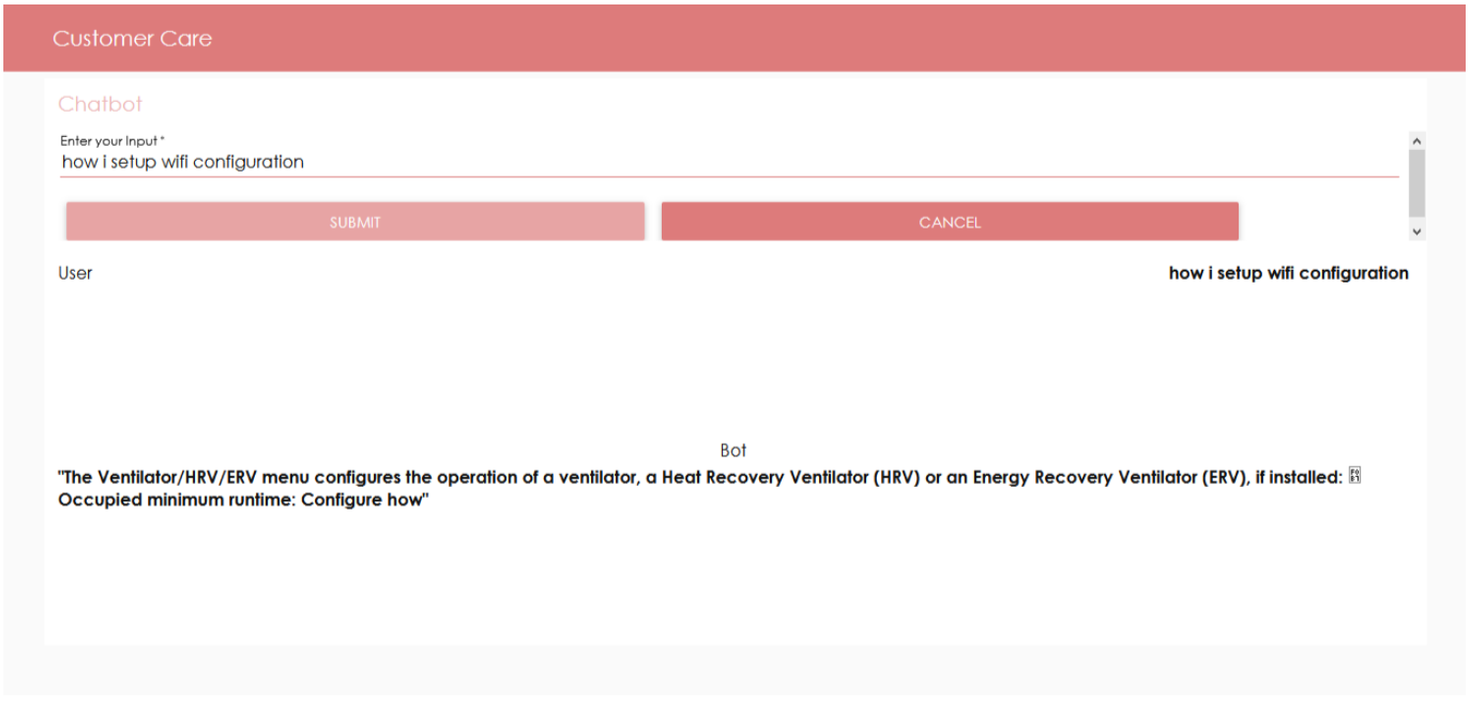


Hardware / Software Designing

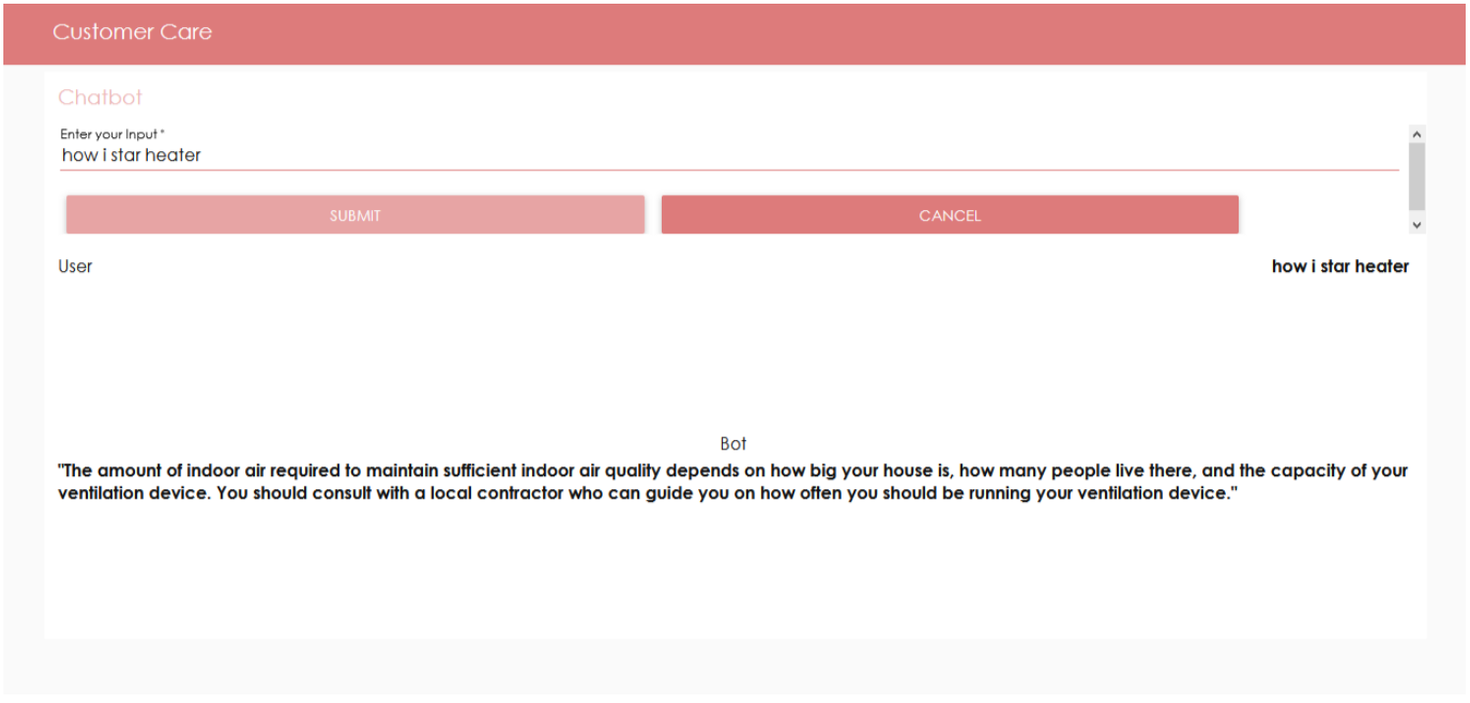
1. Create necessary Watson Services.
2. Configure Watson Discovery.
3. Create Watson Cloud Functions Action.
4. Configure Watson Assistant.
5. Integrate Watson Discovery with Watson Assistant using webhook.
6. Build Node-RED flow to integrate Watson Assistant and Web Dashboard.

4. Experimental Investigation





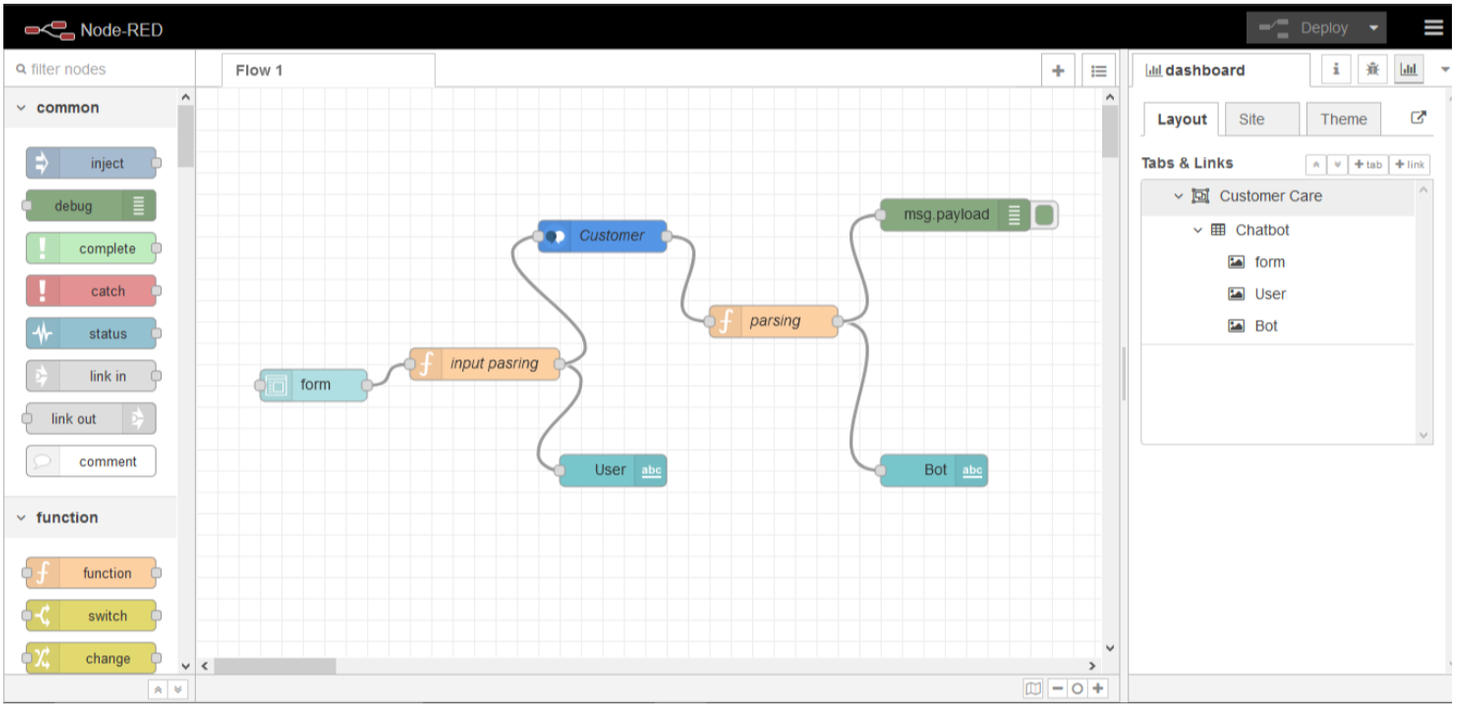

Customer Care
Chatbot
Enter your Input *
how i setup wifi configuration
SUBMIT
CANCEL
User
how i setup wifi configuration
Bot
"The Ventilator/HRV/ERV menu configures the operation of a ventilator, a Heat Recovery Ventilator (HRV) or an Energy Recovery Ventilator (ERV), if installed: Occupied minimum runtime: Configure how"


Customer Care
Chatbot
Enter your Input *
how i star heater
SUBMIT
CANCEL
User
how i star heater
Bot
"The amount of indoor air required to maintain sufficient indoor air quality depends on how big your house is, how many people live there, and the capacity of your ventilation device. You should consult with a local contractor who can guide you on how often you should be running your ventilation device."

5. Flowchart

Insert the following nodes into the flow in Node-RED.

- Inject
- Debug
- ui_Form
- ui_Text
- Function
- Assistant



6. Results

Web based UI was developed by integrating all the services using Node-RED.

URL for UI Dashboard : https://node-red-bdlzo.eu-gb.mybluemix.net/ui/

7. Advantages & Disadvantages

Advantages

1. Reduces Man Power
2. Cost Efficient
3. Less and less calls will be diverted to Customer Representatives.

Disadvantages

1. Sometimes it can mislead cutomers as it tries to search irrelevant information in the manual.
2. It may also give same answers to different queries.

8. Applications

1. This chatbot can be deployed to various websites as it can solve a lot of basic questions.
2. It can be used to deploy as Customer Helpdesk for small scale products as their manual usually has the solution for the user's problems.

9. Conclusion

An Intelligent Customer Helpdesk Chatbot was created using various Watson services like Watson Discovery, Watson Assistant, Watson Cloud Functions and Node-RED.

10. Future Scope

In the future, various other Watson services like Text-To-Speech and Speech-To-Text can be integrated in the chatbot. This can make the chatbot Hands-free.

11. Bibliography

1. Node-RED Starter Application :
https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-applicat ion/

2. Build your oen AI assistant : https://www.youtube.com/watch?v=hitUOFNne14

3. How to use Watson Assistant with Webhooks :
https://www.youtube.com/embed/5z3i5IsBVnk

4. Watson Discovery : https://developer.ibm.com/articles/introduction-watson-discovery/

Appendix
Source Code

Node-RED Flow code

```
[
   {
      "id":"9f300b89.fd354",
      "type":"tab",
      "label":"Flow 1",
      "disabled":false,
      "info":""
      },
   {
      "id":"9f1f4949.12df38",
      "type":"debug",
      "z":"9f300b89.fd354",
      "name":"",
      "active":true,
      "tosidebar":true,
      "console":false,
      "tostatus":false,
      "complete":"payload",
```

```
    "targetType":"msg",
    "x":710,
    "y":120,
    "wires":[]
},
{
    "id":"bf8808da.7a22d",
    "type":"function",
    "z":"9f300b89.fd354",
    "name":"parsing",
    "func":"msg.payload=msg.payload.output.text[0];\nreturn msg;",
    "outputs":1,
    "noerr":0,
    "x":540,
    "y":220,
    "wires":[
        ["159bf5a1.635132","9f1f4949.12df38"]
        ]
},
{
    "id":"2f8aeb5f.26dfd4",
    "type":"ui_form",
    "z":"9f300b89.fd354",
    "name":"",
    "label":"",
    "group":"b675edaf.03e88",
    "order":1,
    "width":24,
    "height":2,
    "options":
    [
        {
            "label":"Enter your Input",
            "value":"text",
            "type":"text",
            "required":true,
```

```
            "rows":null
        }
    ],
    "formValue":{"text":""},
    "payload":"",
    "submit":"submit",
    "cancel":"cancel",
    "topic":"",
    "x":110,
    "y":280,
    "wires":
    [
        ["2518e04f.1ff7b8"]
    ]
},
{
    "id":"2518e04f.1ff7b8",
    "type":"function",
    "z":"9f300b89.fd354",
    "name":"input pasring",
    "func":"msg.payload=msg.payload.text;\nreturn msg;",
    "outputs":1,
    "noerr":0,
    "x":270,
    "y":260,
    "wires":
    [
        ["d1acf3a2.5c82e","e25557c.f7bfda8"]
    ]
},
{
    "id":"d1acf3a2.5c82e",
    "type":"ui_text",
    "z":"9f300b89.fd354",
    "group":"b675edaf.03e88",
    "order":2,
```

```
    "width":24,
    "height":1,
    "name":"",
    "label":"User",
    "format":"{{msg.payload}}",
    "layout":"row-spread",
    "x":390,
    "y":360,
    "wires":[]
},
{
    "id":"159bf5a1.635132",
    "type":"ui_text",
    "z":"9f300b89.fd354",
    "group":"b675edaf.03e88",
    "order":3,
    "width":24,
    "height":6,
    "name":"",
    "label":"Bot",
    "format":"{{msg.payload}}",
    "layout":"col-center",
    "x":690,
    "y":360,
    "wires":[]
},
{
    "id":"e25557c.f7bfda8",
    "type":"watson-conversation-v1",
    "z":"9f300b89.fd354",
    "name":"Customer",
    "workspaceid":"d76120c5-0862-4cde-980b-c9f4c81567e8",
    "multiuser":false,
    "context":true,
    "empty-payload":false,
    "service-endpoint":"",
```

```
        "timeout":"",
        "optout-learning":false,
        "x":380,
        "y":140,
        "wires":[
            ["bf8808da.7a22d"]
            ]
    },
    {
        "id":"b675edaf.03e88",
        "type":"ui_group",
        "z":"",
        "name":"Chatbot",
        "tab":"ccdfc93c.d66e28",
        "order":1,
        "disp":true,
        "width":24,
        "collapse":false
    },
    {
        "id":"ccdfc93c.d66e28",
        "type":"ui_tab",
        "z":"",
        "name":"Customer Care",
        "icon":"",
        "disabled":false,
        "hidden":false
    }
]
```

Watson Cloud Function Action Code

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

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);
        });
    });
}
```