

Project Report on Intelligent Customer Help Desk with Smart Document Understanding (Category: Artificial Intelligence)



**Submitted By:- Priyanshi Agrawal
(pagrawal1_be17@thapar.edu)**

TABLE OF CONTENTS

1. INTRODUCTION

- i. Overview 3
- ii. Purpose 3

2. LITERATURE SURVEY

- i. Existing Problem 4
- ii. Proposed Solution 4

3. THEORITICAL ANALYSIS

- i. Block Diagram 4
- ii. Hardware / Software Designing 5

4. EXPERIMENTAL INVESTIGATION 5

5. FLOWCHART 6

6. RESULT 7

7. ADVANTAGES & DISADVANTAGES 7

8. APPLICATIONS 8

9. CONCLUSION 8

10. FUTURE SCOPE 8

11. BIBILOGRAPHY 8

12. APPENDIX 9

A. Source code

1.INTRODUCTION

1.1 Overview

The project involves building up of a chatbot with the help of IBM Watson Services like Watson Discovery, Watson Assistant, Watson Cloud Functions and Node-Red for making a user interface through which customers can interact with company's representatives and solve their issues.

- Skills Required : Node-RED, IBM Cloud, IBM Watson Services(Watson Assistant, Watson Discovery), Cloud functions
- Project Deliverables : Intelligent Chatbot with Smart Document Understanding
- Project Duration : 19 Days

1.2 Purpose

Making a kind of chatbot that can handle every question be it the one from pre-determined question set or if it falls outside this scope then analyzing if they need a talk with real agent .One interesting thing integrated in the bot is that if a 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 owner's manual. So now, instead of "Would you like to speak to a customer representative?" we can return relevant sections of the owner's manual to help solve our customers' problems.

So unless and until customer specifically asks for a customer representative the bot will try to solve all your queries.Using the Watson Discovery Smart Document Understanding (SDU) feature ,you annotate fields within your documents to train custom conversion models. As you annotate, Watson is learning and will start predicting annotations. Using Watson actions as webhook, Watson Discovery can then be integrated with Watson assistant and finally using Node-Red App, Watson assistant can be integrated with a web UI for customers. This UI can then be used to connect with Watson assistant and chat with it.

1.2.1 Scope of Work:

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

2. LITERATURE SURVEY

2.1 Existing problem

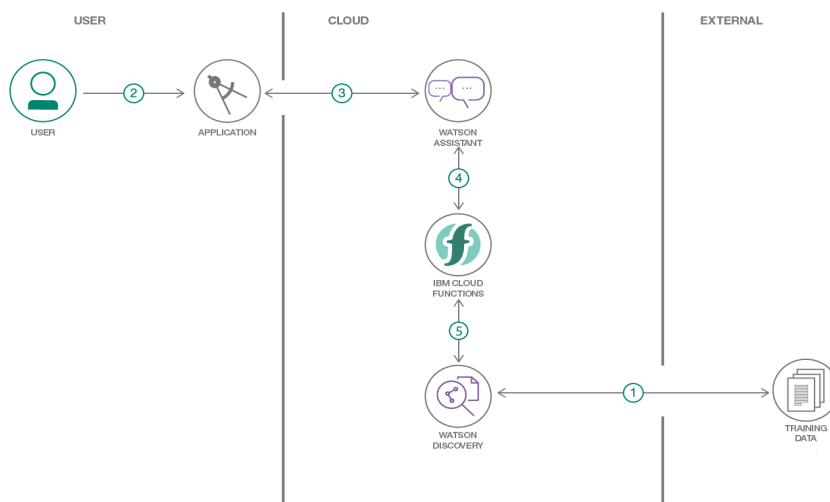
Existing chatbots can handle only some predefined questions and for other it may either not understand those or direct the customer to talk to a real customer care agent. Technologies like Artificial Intelligence have known to decrease user involvement as it can handle a lot of stuff all by itself. So in view of that there is a need of less interference of real agents to handle user's problems by managing them on their own as much as it can.

2.2 Proposed solution

Solution to the above problem can be resolved if the bot can be trained in such a way so that if a question is asked related to the company's product or about the company or the services it provides then it can answer those queries by itself. This project thus made use of Watson Discovery Smart Document Understanding (SDU) feature and later integrated Assistant and Discovery to Node-RED.

3. THEORETICAL ANALYSIS

3.1 Block/Flow Diagram



3.2 Hardware / Software designing

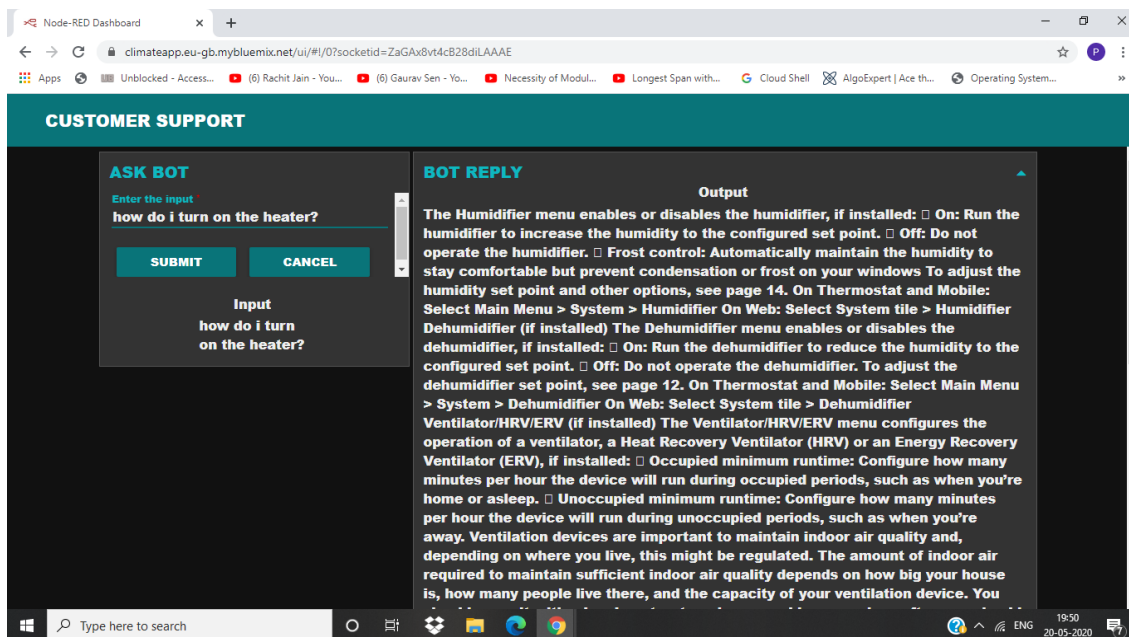
- Create necessary Watson services.
- Create a customer care dialog skill in Watson Assistant
- Use Smart Document Understanding to build an enhanced Watson Discovery collection
- Create an IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery
- Build a web application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

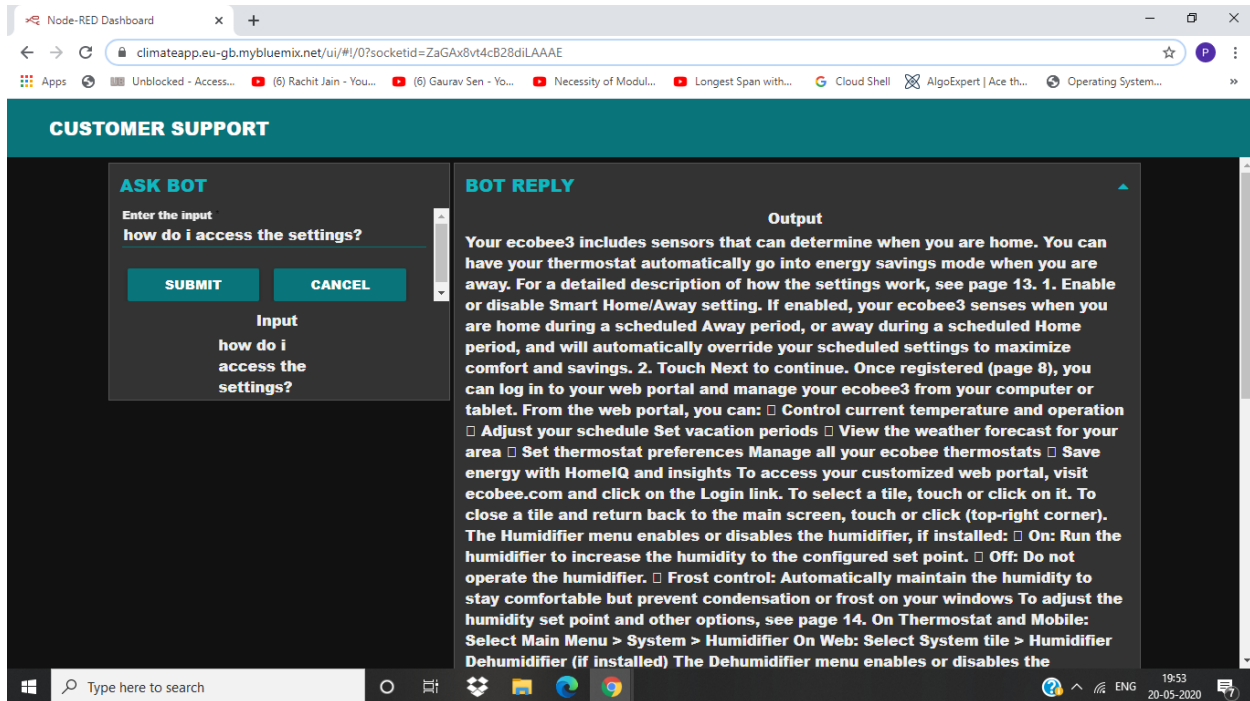
4. EXPERIMENTAL INVESTIGATIONS

Create the following services:

- Watson Discovery
- Watson Assistant
- Node Red
- IBM cloud function

After the creation of all these services we finally integrate it in Node-Red App and obtain the following results:

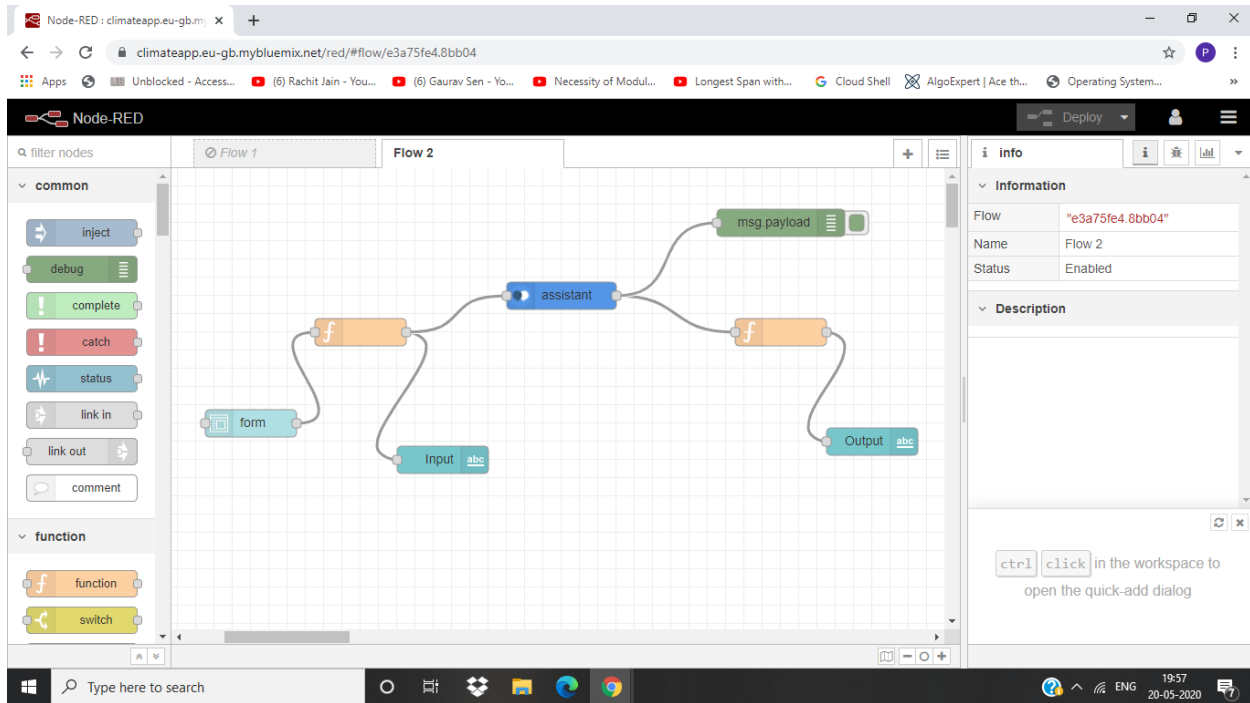




5. FLOWCHART

Insert the following nodes into the flow of Node-red:

- 1) form (input from the user)
- 2) function node (for passing the input to assistant)
- 3) ui_text (output the question asked by the user)
- 4) assistant node (takes the question as input and provides the answer via Watson assistant and Watson Discovery)
- 5) function node (for processing the output given by the assistant node)
- 6) msg.payload node(for debugging)
- 7) ui_text (for printing the processed output or answer provided by assistant)



6. RESULTS

Node-Red Dashboard can be accessed build above and results shown can be tested using the provided URL - <https://climateapp.eu-gb.mybluemix.net/ui>

7. ADVANTAGES & DISADVANTAGES

Advantages:

- Are available 24x7
- Reduces workforce for this purpose
- Save Money
- Efficient use of trendy technology like Artificial Intelligence and NLP.

Disadvantages:

- Chatbots have no emotions so sometime may not understand the state of customer.
- May give same answer for different queries.
- Might sometime provide inadequate results if not trained properly.

8.APPLICATIONS

- It can be deployed in various popular social media applications like facebook, slack, telegram.
- Solving users problems 24x7

9.CONCLUSION

Successfully created Intelligent helpdesk smart chatbot using Watson assistant, Watson discovery, Node-RED and cloud-functions.

10. Future Scope

Including more documents in the Discovery and including text to speech and speech to text services to access the chatbot handsfree.

11. Bibliography

1.Node-RED Starter Application:

<https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/>

2. Build your own AI assistant :

<https://www.youtube.com/watch?v=TrVmuM6AhMU&feature=youtu.be>

3. How to use Watson Assistant with Webhooks :

<https://www.youtube.com/embed/5z3i5lsBVnk>

4. Watson Discovery :

<https://developer.ibm.com/articles/introduction-watson-discovery/>

12. Appendix

Source Code

1) Node-RED Flow code

```
[
  {
    "id": "e3a75fe4.8bb04",
    "type": "tab",
    "label": "Flow 2",
    "disabled": false,
    "info": ""
  },
  {
    "id": "ce39e9c6.d35358",
    "type": "ui_form",
    "z": "e3a75fe4.8bb04",
    "name": "",
    "label": "",
    "group": "c2256707.4f1478",
    "order": 1,
    "width": 7,
    "height": 2,
    "options": [
      {
        "label": "Enter the input",
        "value": "input",
        "type": "text",
        "required": true,
        "rows": null
      }
    ],
    "formValue": {
      "input": ""
    },
    "payload": "",
    "submit": "submit",
  }
]
```

```

    "cancel": "cancel",
    "topic": "",
    "x": 340,
    "y": 280,
    "wires": [
      [
        "5887ecec.dbe2d4"
      ]
    ]
  },
  {
    "id": "5887ecec.dbe2d4",
    "type": "function",
    "z": "e3a75fe4.8bb04",
    "name": "",
    "func": "msg.payload=msg.payload.input\\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 460,
    "y": 180,
    "wires": [
      [
        "f767fc94.58ef6",
        "8a3a8bc5.d48a18"
      ]
    ]
  },
  {
    "id": "f767fc94.58ef6",
    "type": "ui_text",
    "z": "e3a75fe4.8bb04",
    "group": "c2256707.4f1478",
    "order": 3,
    "width": 3,
    "height": 2,
    "name": "",
    "label": "Input",

```

```

    "format": "{{msg.payload}}",
    "layout": "col-center",
    "x": 550,
    "y": 320,
    "wires": []
  },
  {
    "id": "8a3a8bc5.d48a18",
    "type": "watson-conversation-v1",
    "z": "e3a75fe4.8bb04",
    "name": "",
    "workspaceid": "6c3a31ce-0de9-4c3f-b0b2-27a4542e3769",
    "multiuser": false,
    "context": false,
    "empty-payload": false,
    "service-endpoint":
    "https://api.eu-gb.assistant.watson.cloud.ibm.com/instances/17c8f24c-7612-4732-87b
    e-495d3f2eb86e",
    "timeout": "",
    "optout-learning": false,
    "x": 680,
    "y": 140,
    "wires": [
      [
        "1676787f.0983e8",
        "a0281250.8e127"
      ]
    ]
  },
  {
    "id": "1676787f.0983e8",
    "type": "function",
    "z": "e3a75fe4.8bb04",
    "name": "",
    "func": "msg.payload.text = \"\\n\";nif(msg.payload.context.webhook_result_1){\n
    for(var i in msg.payload.context.webhook_result_1.results){\n      msg.payload.text =
    msg.payload.text+ \"\\n\\n\" + msg.payload.context.webhook_result_1.results[i].text;\n

```

```

\n } \n msg.payload = msg.payload.text;\n}\nelse\nmsg.payload =
msg.payload.output.text[0]\nreturn msg;";
  "outputs": 1,
  "noerr": 0,
  "x": 920,
  "y": 180,
  "wires": [
    [
      "1cc47ebc.d48f41"
    ]
  ]
},
{
  "id": "a0281250.8e127",
  "type": "debug",
  "z": "e3a75fe4.8bb04",
  "name": "",
  "active": true,
  "tosidebar": true,
  "console": false,
  "tostatus": false,
  "complete": "false",
  "x": 920,
  "y": 60,
  "wires": []
},
{
  "id": "1cc47ebc.d48f41",
  "type": "ui_text",
  "z": "e3a75fe4.8bb04",
  "group": "29083e83.1059a2",
  "order": 2,
  "width": 14,
  "height": 21,
  "name": "",
  "label": "Output",
  "format": "{{msg.payload}}",

```

```

    "layout": "col-center",
    "x": 1020,
    "y": 300,
    "wires": []
  },
  {
    "id": "c2256707.4f1478",
    "type": "ui_group",
    "z": "",
    "name": "ASK BOT",
    "tab": "b700dd00.69db3",
    "order": 1,
    "disp": true,
    "width": 7,
    "collapse": false
  },
  {
    "id": "29083e83.1059a2",
    "type": "ui_group",
    "z": "",
    "name": "BOT REPLY",
    "tab": "b700dd00.69db3",
    "order": 2,
    "disp": true,
    "width": 14,
    "collapse": true
  },
  {
    "id": "b700dd00.69db3",
    "type": "ui_tab",
    "z": "",
    "name": "CUSTOMER SUPPORT",
    "icon": "dashboard",
    "disabled": false,
    "hidden": false
  }
]

```

Cloud function Node.js 10 code for discovery integration with assistant:

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