PROJECT REPORT

Intelligent Customer Help Desk with Smart Document Understanding

Category: Artificial Intelligence Developer

Application ID:SPS_APL_20200003566

Project ID: SPS_PRO_99

Internship at SmartInternz

Kasimsetty Lakshmi Prasanna

prasannakasimsetty85@gmail.com

1. INTRODUCTION

1.1 Overview:

We use the typical customer care chatbot experience but instead of relying on predefined responses, our dialog will provide a hook that can call out to other IBM Watson services for additional sources of information. In our case, it will be an owner's manual that has been uploaded into Watson Discovery.

1.2 Purpose:

The purpose is to Enhance the customer helpdesks with Smart Document Understanding using webhooks in Watson Assistant.

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.

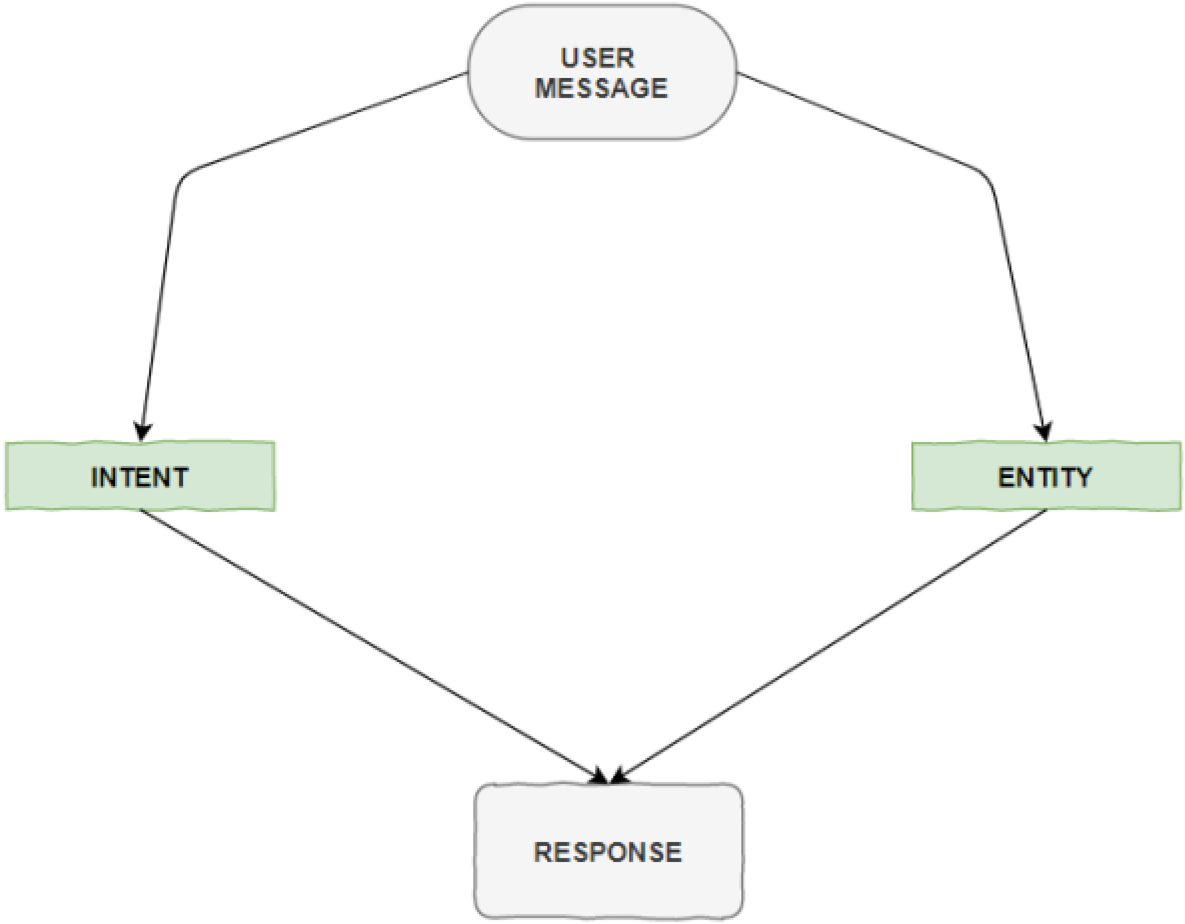
2.2 Proposed solution

In this project, 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 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.

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 owner’s manual is important and what is not. This will improve the answers returned from the queries.

3. THEORITICAL ANALYSIS

3.1 Block Diagram



3.2 Hardware / Software designing

1. Create IBM Cloud Services
2. Configure Watson Discovery
3. Create IBM Cloud Functions action
4. Configure Watson Assistant
5. Build Node-RED Flow to Integrate All Services
6. Configure the nodes and Build A Web Dashboard in Node-RED
7. Deploy and Run the application

4. EXPERIMENTAL INVESTIGATIONS

1.Create IBM Cloud services

Create the following services:

a. Watson Discovery

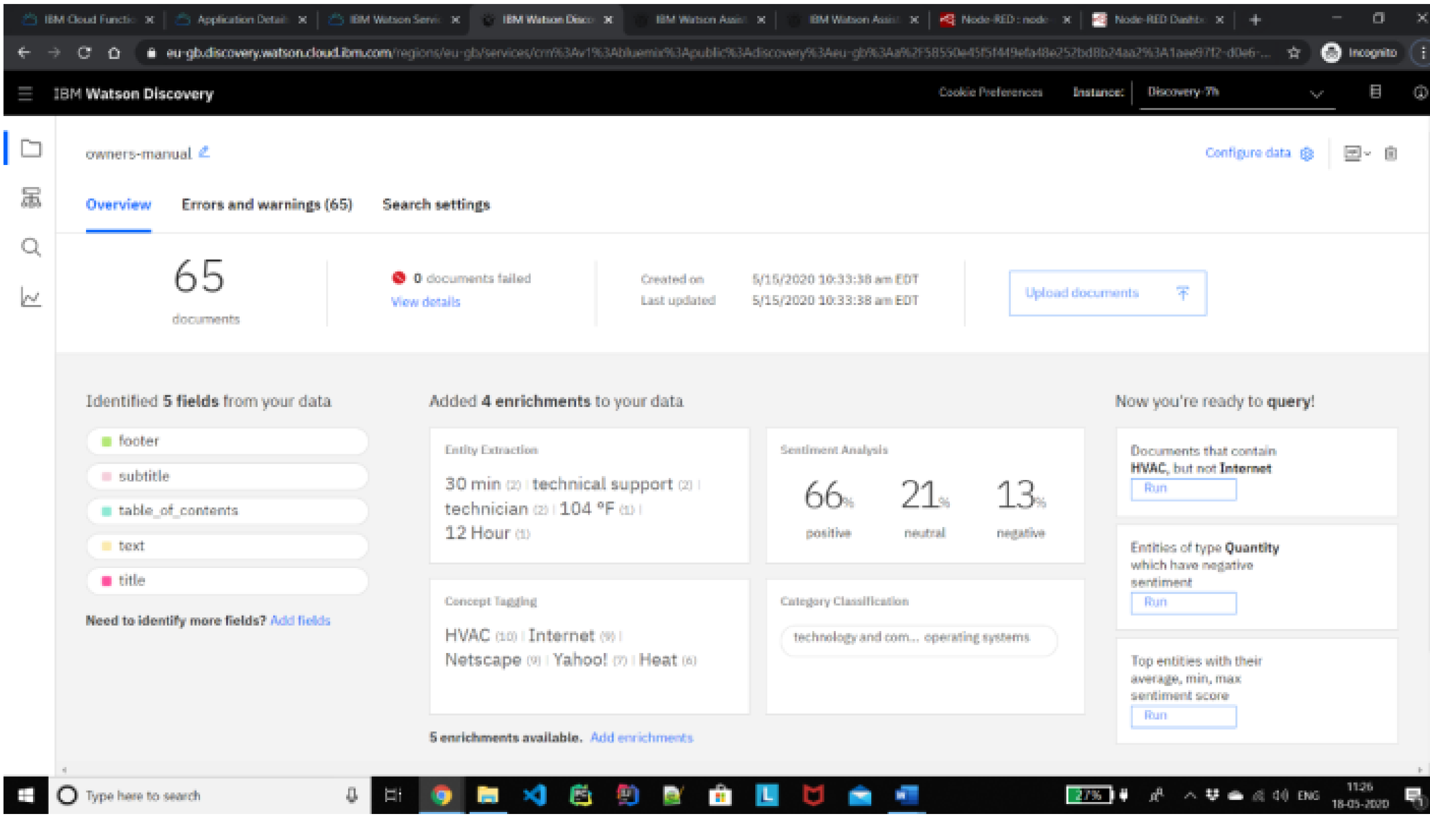
b. Watson Assistant

c. Node Red

2.Configure Watson Discovery

Import the document

Launch the Watson Discovery tool and create a new data collection by selecting the Upload your own data option. Give the data collection a unique name. When prompted, select and upload the ecobee3_userGuide.pdf file located in the data directory of your local repo.


IBM Watson Discovery
Cookie Preferences
Instance:
Discovery-7h
owners-manual
Configure data
Overview
Errors and warnings (65)
Search settings
65
documents
0 documents failed
View details
Created on 5/15/2020 10:33:38 am EDT
Last updated 5/15/2020 10:33:38 am EDT
Upload documents
Identified 5 fields from your data
footer
subtitle
table_of_contents
text
title
Need to identify more fields? Add fields
Added 4 enrichments to your data
Entity Extraction
30 min (2) | technical support (2) | technician (2) | 104 °F (1) | 12 Hour (1)
Sentiment Analysis
66% positive
21% neutral
13% negative
Concept Tagging
HVAC (10) | Internet (9) | Netscape (9) | Yahoo! (7) | Heat (6)
Category Classification
technology and com... operating systems
5 enrichments available. Add enrichments
Now you're ready to query!
Documents that contain HVAC, but not Internet
Run
Entities of type Quantity which have negative sentiment
Run
Top entities with their average, min, max sentiment score
Run
Type here to search

Annotate with SDU

Now let's apply SDU to our document to see if we can generate some better query responses. From the Discovery collection panel, click the Configure data button (located in the top right corner) to start the SDU process. The goal is to annotate all of the pages in the document so Discovery can learn what text is important, and what text can be ignored.

3.Create IBM Cloud Functions action

Now let's create the web action that will make queries against our Discovery collection.

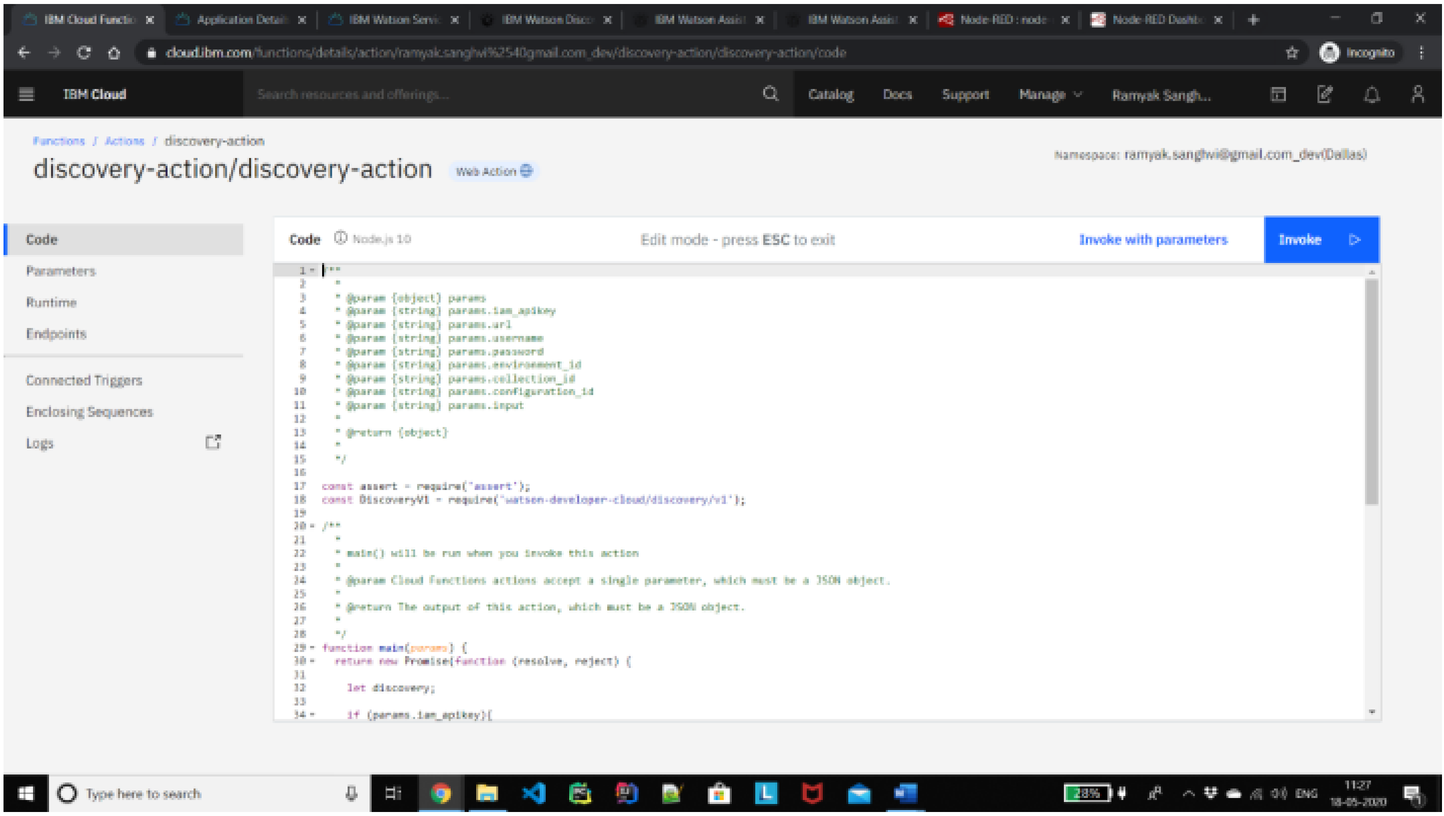
Start the IBM Cloud Functions service by selecting Create Resource from the IBM Cloud dashboard. Enter functions as the filter, then select the Functions card:

From the Functions main panel, click on the Actions tab. Then click on Create. From the Create panel, select the Create Action option.

On the Create Action panel, provide a unique Action Name, keep the default package and select the Node.js 10 runtime.

Click the Create button to create the action.

Once your action is created, click on the Code tab:



```
/**
  *
  * @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){
```

In the code editor window, cut and paste in the code from the disco-action.js file found in the action’s directory of your local repository. The code is pretty straight-forward - it simply connects to the Discovery service, makes a query against the collection, then returns the response.

If you press the Invoke button, it will fail due to credentials not being defined yet. We'll do this next. Select the Parameters tab:

Add the following keys:

url

environment_id

collection_id

iam_apikey

For values, please use the values associated with the Discovery service you created in the previous step. Now that the credentials are set, return to the Code panel and press the Invoke button again. Now you should see actual results returned from the Discovery service:

Next, go to the Endpoints panel:

Click the checkbox for Enable as Web Action. This will generate a public endpoint URL.

Take note of the URL value, as this will be needed by Watson Assistant in a future step.

To verify you have entered the correct Discovery parameters, execute the provided curl command. If it fails, re-check your parameter values.

4.Configure Watson Assistant

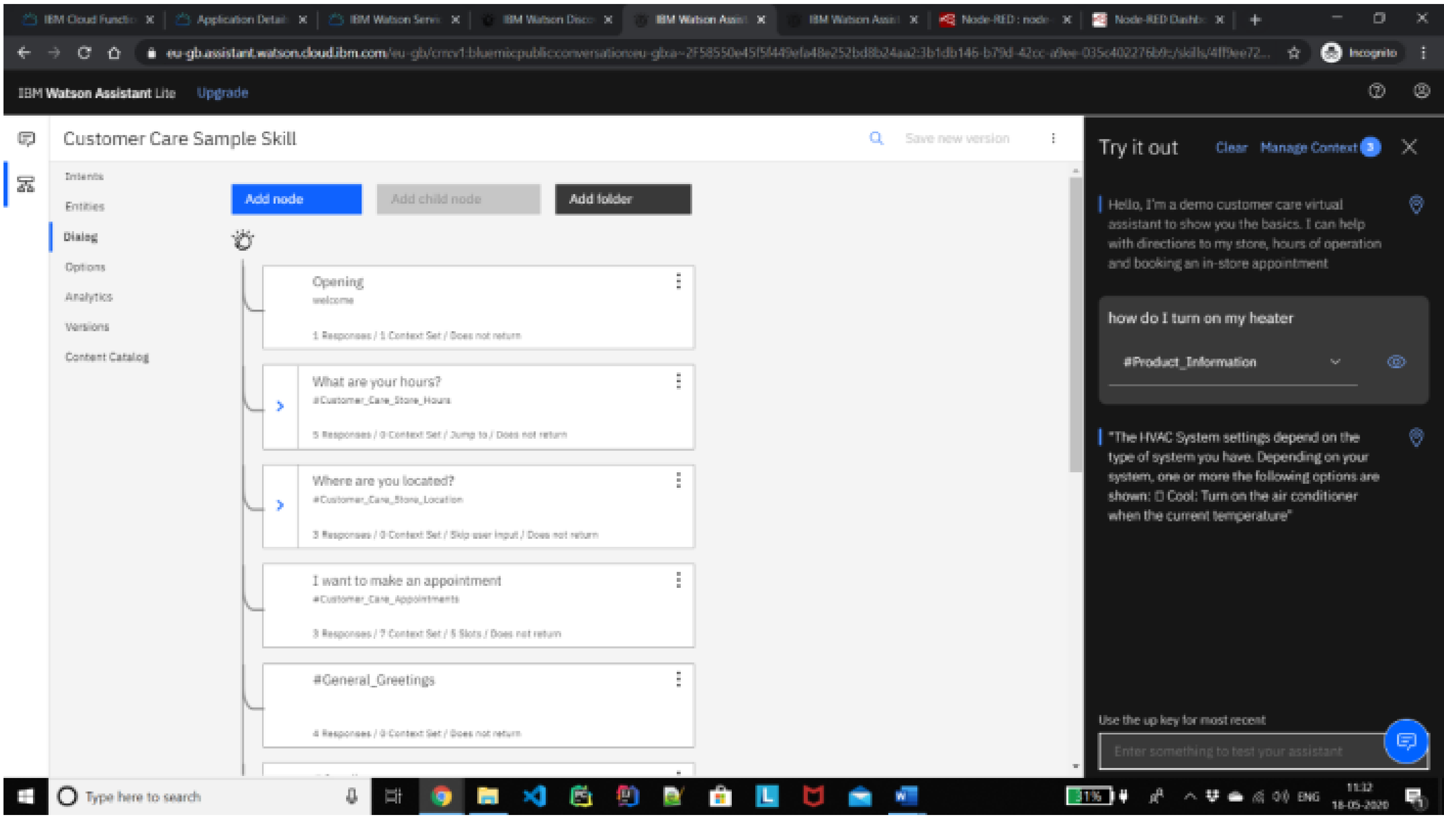
Launch the Watson Assistant tool and create a new dialog skill. Select the Use sample skill option as your starting point. This dialog skill contains all of the nodes needed to have a typical call center conversation with a user.

Add new intent

The default customer care dialog does not have a way to deal with any questions involving outside resources, so we will need to add this. Create a new intent that can detect when the user is asking about operating the

Product. From the Customer Care Sample Skill panel, select the Intents tab.

Click the Create intent button. Name the intent #Product_Information, and at a minimum, enter the following example questions to be associated with it.



Opening
welcome
1 Responses / 1 Context Set / Does not return

What are your hours?
#Customer_Care_Store_Hours
5 Responses / 0 Context Set / Jump to / Does not return

Where are you located?
#Customer_Care_Store_Location
3 Responses / 0 Context Set / Skip user input / Does not return

I want to make an appointment
#Customer_Care_Appointments
3 Responses / 7 Context Set / 5 Slots / Does not return

#General_Greetings
4 Responses / 0 Context Set / Does not return

Hello, I'm a demo customer care virtual assistant to show you the basics. I can help with directions to my store, hours of operation and booking an in-store appointment

how do I turn on my heater

#Product_Information

"The HVAC System settings depend on the type of system you have. Depending on your system, one or more the following options are shown: Cool: Turn on the air conditioner when the current temperature"

Create new dialog node

Now we need to add a node to handle our intent. Click on the Dialog tab, then click on the drop-down menu for the Small Talk node, and select the Add node below option.

Name the node "Ask about product" and assign it our new intent. This means that if Watson Assistant recognizes a user input such as "how do I set the time?", it will direct the conversation to this node.

Enable webhook from Assistant

Set up access to our Webhook for the IBM Cloud Functions action you created in

Step #4. Select the Options tab:

Enter the public URL endpoint for your action. Return to the Dialog tab, and click on the Ask about product node. From the details panel for the node, click on Customize, and enable Webhooks for this node: Click Apply.

The dialog node should have a Return variable set automatically to $webhook_result_1. This is the variable name you can use to access the result from the Discovery service query.

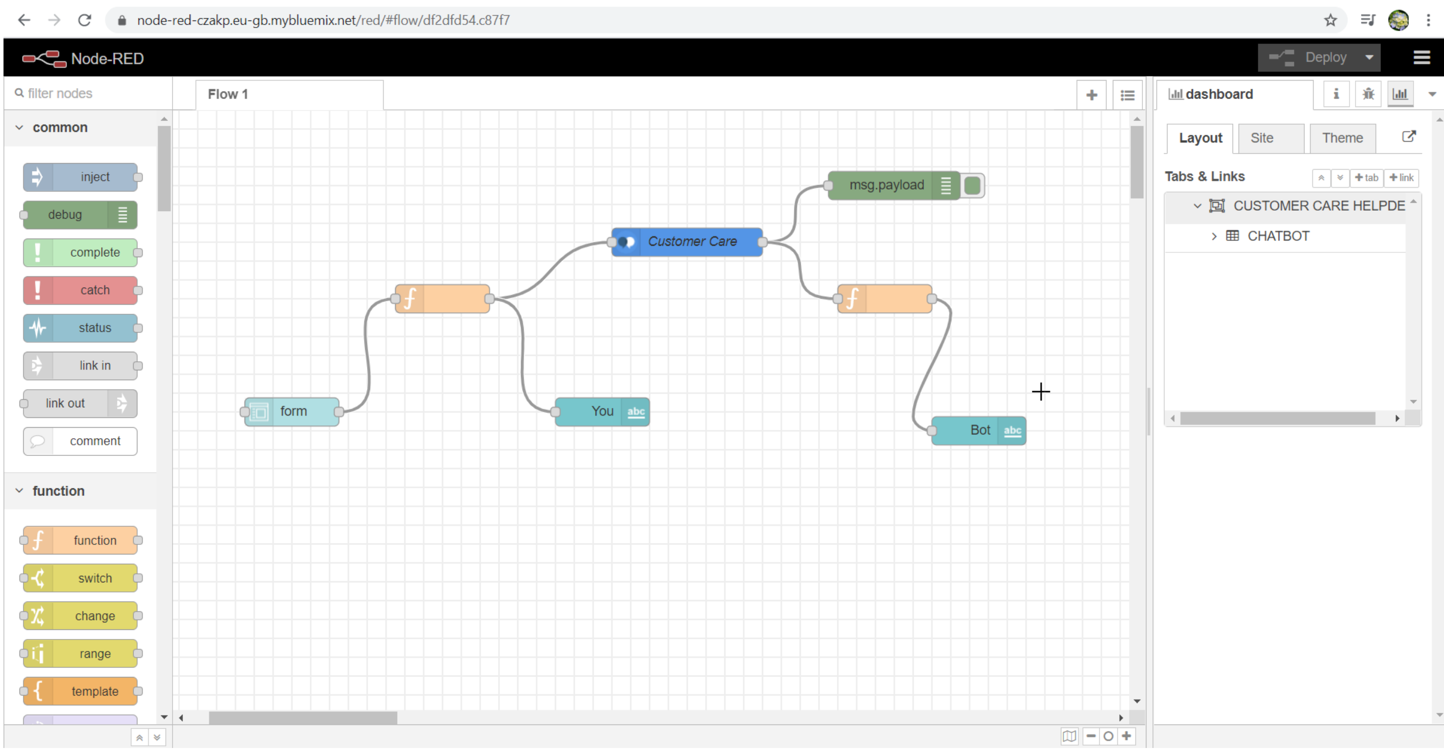
Test in Assistant Tooling

From the Dialog panel, click the Try it button located at the top right side of the panel. Enter some user input: Note that the input "how do I turn on the heater?" has triggered our Ask about product dialog node, which is indicated by the #Product_Information response. And because we specified that $webhook_result_1.passages be the response, that value is displayed also. You can also verify that the call was successfully completed by clicking on the Manage Context button at the top right. The response from the Discovery query will be stored in the $webhook_result_1 variable.

5.Create flow and configure node:

At first go to manage palette and install dashboard. Now, Create the flow with the help of following node:

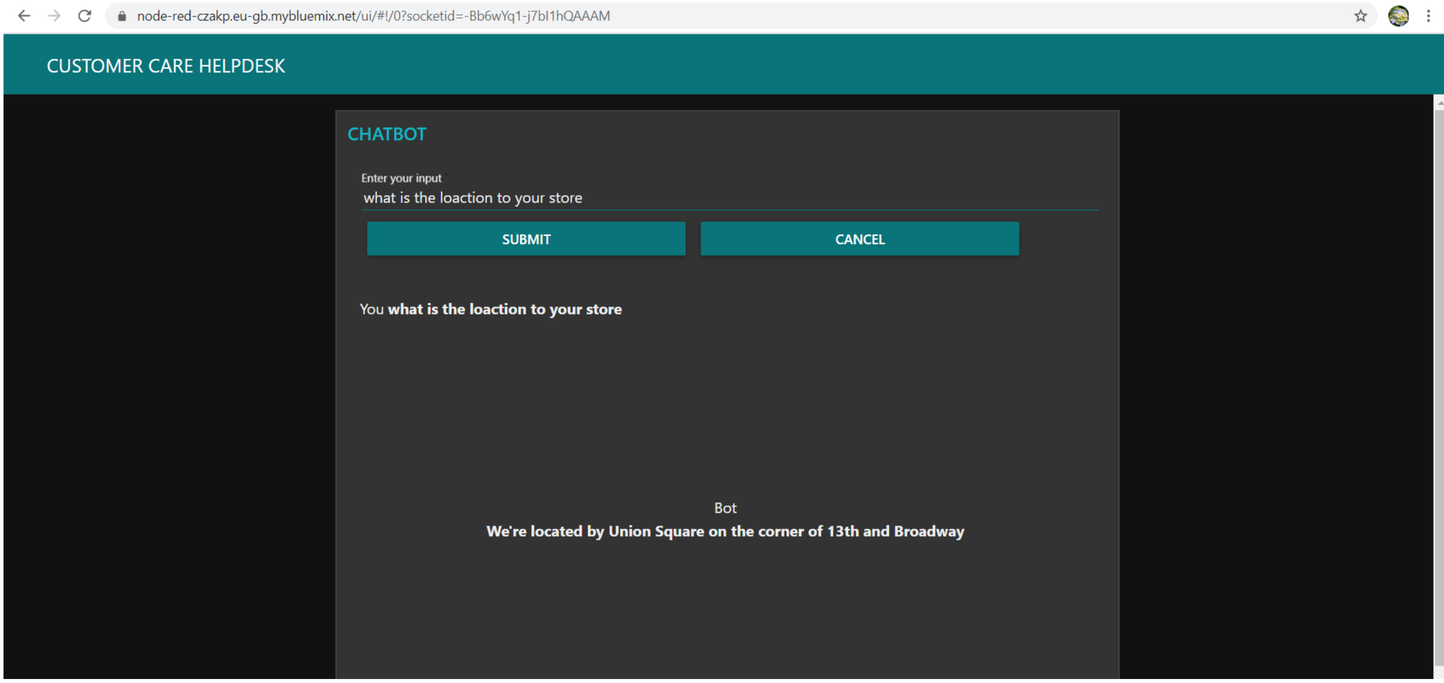
1. Inject
2. Assistant
3. Debug
4. Function
5. Ui_Form
6. Ui_Text



6.Deploy and run Node Red app.

Deploy the Node Red flow.

Then copy the link url upto .net/ and paste at anew tab by ui at the end of the url, like this, https://node-red-czakp.eu-gb.mybluemix.net/ui/



node-red-czakp.eu-gb.mybluemix.net/ui/#!/0?socketid=-Bb6wYq1-j7bI1hQAAAM
CUSTOMER CARE HELPDESK
CHATBOT
Enter your input
SUBMIT
CANCEL

what is the loaction to your store

You **what is the loaction to your store**

Bot
We're located by Union Square on the corner of 13th and Broadway

5. FLOWCHART

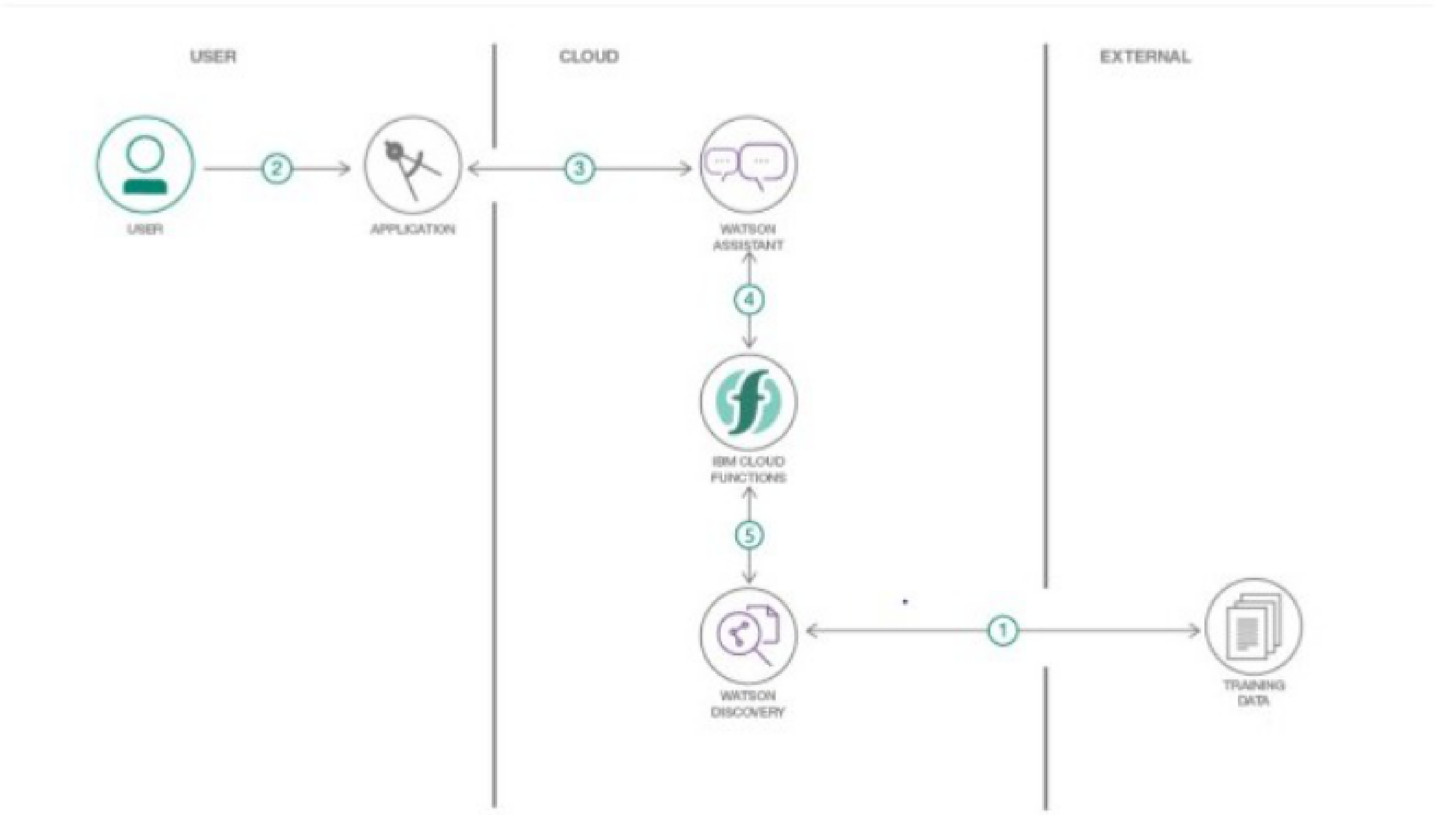
1. The document is annotated using Watson Discovery SDU

2.The user interacts with the backend server via the app UI. The frontend app UI is a chatbot that engages the user in a conversation.

3.Dialog between the user and backend server is coordinated using a Watson Assistant dialog skill.

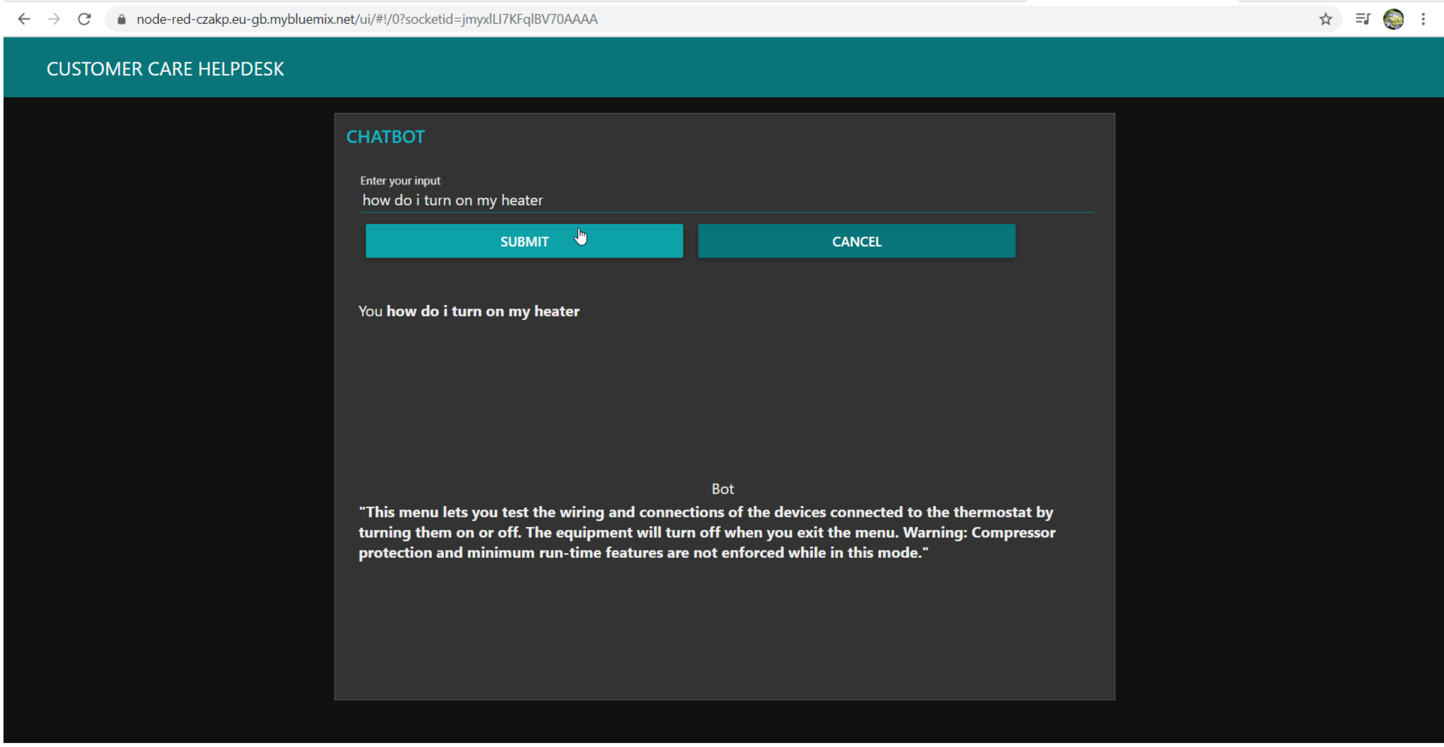
4.If the user asks a product operation question, a search query is passed to a predefined IBM Cloud Functions action.

5.The Cloud Functions action will query the Watson Discovery service and return the results.



6.RESULTS

The chatbot was successfully made using Watson assistant and using SDU. All the services were integrated using Node Red Application.


node-red-czakp.eu-gb.mybluemix.net/ui/#!/0?socketid=jmyxlLI7KFqlBV70AAAA
CUSTOMER CARE HELPDESK
CHATBOT
Enter your input
SUBMIT
CANCEL

how do i turn on my heater

You **how do i turn on my heater**

Bot

"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."

7.ADVANTAGES AND DISADVANTAGES

Advantages:

A. Reduced costs: Chatbots eliminate the need for labor during online interaction with customers. This is obviously a great advantage for companies that receive multiple queries at once. In addition to saving costs with them, companies can align the chatbot with their objectives, and use them as a means to enhance customer conversion.

B. 24/7 Availability: Unlike humans, once we install a chatbot, it can handle queries at any time of day. Thus, the customer does not have to wait for a commercial of the company to help him. This also allows companies to monitor customer « traffic » during non-working hours and contact them later.

C. Learning and updating: AI-based chatbots are able to learn from interactions and update independently. This is one of the main advantages. When you hire a new employee, you have to train them continuously. However, chatbots « form » themselves (with certain limitations, of course).

D. Management of multiple clients: Humans can serve a limited number of customers at the same time.

This restriction does not exist for chatbots, and they can manage all the necessary queries simultaneously. This is one of the main advantages of using chatbot, as no customer is left unattended and you are solving different problems at the same time. There are chatbots companies already working on developing voice chatbot services.

Disadvantages:

A. Complex interface: It is often considered that chatbots are complicated and need a lot of time to understand what you want in customer. Sometimes, it can also annoy the client about their slowness, or their difficulty in filtering responses. They don’t get you right: Fixed chatbots can get stuck easily. If a query doesn’t relate to something you’ve previously « taught » it, you won’t understand it. This can lead to a frustrated customer and the loss of the sale. Other times they do understand you, but they need double (or triple) as many messages as one person, which spoils the user experience.

B. Bad memory: The chatbots are not able to memorize a conversation already had, which forces the user to write the same thing over and over again. This can be cumbersome for the client and annoying for the effort required. Therefore, it is important to be careful when designing chatbots and make sure that the program is able to understand users’ queries and respond accordingly.

8.APPLICATIONS

Some applications can be: -

1. Help User: This chatbot will be useful for the user to ask the assistant the queries related to the Product and will give them clear guidance about the Product. If the Assistant doesn’t know about a certain query, it will redirect to the correct person for it.

2. Content delivery: Media Publishers have realized that chatbots are a powerful way to engage with their audiences and monitor engagement to gain valuable insights on reader interests. Chat with the CNN and

Wall Street Journal Chatbots on Facebook Messenger and receive the latest news directly in Messenger, without having to visit their websites.

3. Companionship: The primary function of the chatbot is to be a virtual companion – To speak with senior people on general topics like the weather, nature, hobbies, movies, music, news, etc. The chatbot asks questions, reacts to the answers, is able to speak on various topics, and share interesting news and facts from Google.

9. CONCLUSION

This chatbot will be useful for the user to ask the assistant the queries related to the Product and will give them clear guidance about the product. If the Assistant doesn't know about a certain query, it will redirect to the correct person for it. Chatbots are quickly making transformational changes and allowing businesses to thrive through customer interactions. The feedback and survey through chatbots strengthen the position of businesses as they analyze the reason behind different levels of customer approval. Use of conversational AI chatbots only means better engagement and relentless need for customer satisfaction in the near future.

10.FUTURE SCOPE

Future Scope of this chatbot can be by adding the following to make it more advance: -
1] Smarter Virtual Assistants: Much of what virtual assistants do now are basic skills, such as retrieving data and basic computation. As natural language processing (NLP) continues to mature, virtual assistants will improve their comprehension and response capabilities, allowing for their use to become more widespread and complex. Also, as machine learning progresses, we may see virtual assistants become smarter and begin to learn and predict customer needs.

2] Integration with IoT Devices: Car speakers, smart home devices, and wearables are just a few examples where the virtual assistant is departing from its original hardware and making its way to in-context devices.

These integrations ensure that virtual assistants can always be near their human and ready to support any need. It is expected that these integrations will continue at an accelerated pace throughout 2018.

3] Voice-control: Voice recognition can be added with the virtual assistant. Then the customer can control application by using his voice. Soon, we could be joining meetings with a voice command, instead of dialing in the long meeting ID and password.

11.BIBLIOGRAPHY

- https://cloud.ibm.com/docs

APPENDIX

A.Sourcecode :

Cloud Function 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);
    });
  });
}
```

Customer Care skill flow.json

```
{
     "intents": [
       {
         "intent": "Customer_Care_Appointments",
         "examples": [
           {
             "text": "I want to talk in person with someone about my case"
           },
           {
             "text": "I would like to discuss my situation face to face"
           },
        },
        {
          "text": "I would like to make an appointment to visit the nearest store to my
location."
        },
        {
          "text": "Could I speak to someone in the store next tuesday?"
        },
        {
          "text": "Can I book an in person session"
        },
        {
          "text": "are you available on tuesday"
        },
        {
          "text": "can i book for tonight"
        },
        {
          "text": "do you have availability next week"
        },
        {
          "text": "can i make an appointment"
        },
        {
          "text": "meet in store"
        },
        {
          "text": "can you make an appointment for me"
        },
        {
          "text": "i'd like to make an appointment"
        },
        {
          "text": "What time can I meet the staff?"
        },
        {
          "text": "When can I meet with one of your employees at your store?"
        },
        {
          "text": "Want to change my visit"
        },
        {
          "text": "Store appointment"
        },
        {
          "text": "Set up an appt"
        },
        {
          "text": "Make an appointment"
        },
        {
          "text": "i'd like to come in for an appointment"
        },
        {
          "text": "I prefer a face to face visit"
        }
      ],
      "description": "Schedule or manage an in-store appointment."
    },
    {
      "intent": "General_Greetings",
      "examples": [
        {
          "text": "How have you been?"
        },
        {
          "text": "hi"
        },
        {
          "text": "Hi there"
        },
        {
          "text": "Looking good eve"
        },
        {
```

```
  "text": "Hey you"
},
{
  "text": "How is it going?"
},
{
  "text": "You there"
},
{
  "text": "Who is this?"
},
{
  "text": "What's up?"
},
{
  "text": "What's new?"
},
{
  "text": "How r u?"
},
{
  "text": "Hey twin"
},
{
  "text": "Hi advisor"
},
{
  "text": "Ok take me back"
},
{
  "text": "Hey there"
},
{
  "text": "Hey there all"
},
{
  "text": "Hey how are you doing"
```

```
},
{
  "text": "Hello I am looking for some help here"
},
{
  "text": "Hello"
},
{
  "text": "Hello Agent"
},
{
  "text": "Have you been well?"
},
{
  "text": "Greetings"
},
{
  "text": "Good to see you"
},
{
  "text": "Good morning"
},
{
  "text": "Good evening"
},
{
  "text": "Good day"
},
{
  "text": "hiya"
},
{
  "text": "yo"
},
{
  "text": "How are things going?"
},
        {
          "text": "How are you today?"
        }
      ],
      "description": "Greetings"
    },
    {
      "intent": "Goodbye",
      "examples": [
        {
          "text": "ciao"
        },
        {
          "text": "arrivederci"
        },
        {
          "text": "see ya"
        },
        {
          "text": "good bye"
        },
        {
          "text": "so long"
        },
        {
          "text": "bye"
        }
      ],
      "description": "Good byes"
    },
    {
      "intent": "Customer_Care_Store_Location",
      "examples": [
        {
          "text": "what is the address"
        },
        {
```

```
  "text": "Looking for a location"
},
{
  "text": "What is the closest store to my address?"
},
{
  "text": "What is the nearest branch?"

},
{
  "text": "What is the store near my zip code?"
},
{
  "text": "location please"
},
{
  "text": "how do i find you"
},
{
  "text": "I want to know about a store"
},
{
  "text": "where are you"
},
{
  "text": "what's your location"
},
{
  "text": "give me directions"
},
{
  "text": "which cross streets are you on"
},
{
  "text": "how can i get to you from grand central"
},
{
```

```
             "text": "please suggest route from times square"
           },
           {
             "text": "Where is your office?"
           },
           {
             "text": "I need help with find a store"
           },
           {
             "text": "I'd like to go to a store"
           },
           {
             "text": "Go to your company"
           },
           {
             "text": "how do i get to your business"
           },
           {
             "text": "Where are you located?"
           },
           {
             "text": "Where is?"
           },
           {
             "text": "Find store"
           },
           {

"t
ex
t"
:
{
             "text": "how do i get to your place"
           },
           {
             "text": "where are you located"
           },
           {
```

```
      "text": "can you give me directions"
    }
  ],
  "description": "Locate a physical store location or an address."
},
{
  "intent": "Product_Information",
  "examples": [
    {
      "text": "How do I turn on thermostat"
    },
    {
      "text": "How to start the heater"
    },
    {
      "text": "how to increase brightness"
    },
    {
      "text": "how to adjust temperaure"
    },
    {
      "text": "how do I access the settings"
    },
    {
      "text": "How do I set the time"
    },
    {
      "text": "How do I turn on the heater"
    },
    {
      "text": "How to set the temperature"
    },
    {
      "text": "How to set temperature"
    },
    {
      "text": "how to install battery"
```

```
        }
      ],
      "description": ""
    },
    {
      "intent": "Customer_Care_Store_Hours",
      "examples": [
        {
          "text": "store open hours?"
        },
        {
          "text": "are stores open on sunday"
        },
        {
          "text": "how late are you open tonight"
        },
        {
          "text": "how late are you open"
        },
        {
          "text": "Are you open on Sunday"
        },
        {
          "text": "Hours of operation"
        },
        {
          "text": "What time do you close today"
        },
        {
          "text": "what time do you close on Sunday"
        },
        {
          "text": "What time do you open on Saturdays"
        },
        {
          "text": "What time do you close"
        },
{
  "text": "when do you close"
},
{
  "text": "will you be open Memorial day"
},
{
  "text": "will you open for christmas"
},
{
  "text": "what are your hours"
},
{
  "text": "What time is your store open on saturday?"
},
{
  "text": "What time do stores close?"
},
{
  "text": "What time does the central manchester store shut on a saturday?"
},
{
  "text": "What time are you closing today?"
},
{
  "text": "What are your hous?"
},
{
  "text": "What are ur opening hours?"
},
{
  "text": "What are the saturday opening times for the local store?"
},
{
  "text": "What are the hours of operation?"
},
{
  "text": "What are the business hours of the store nearest to me?"
},
{
  "text": "Is the branch open now?"
},
{
  "text": "How long are you open?"
},
{
  "text": "How early do you open?"
},
{
  "text": "Does the store in the city center opens till 8pm on weekends?"
},
{
  "text": "Can you tell me how late the stores are open till?"
},
{
  "text": "At what hour can I swing by?"
},
{
  "text": "Are you open on sundays, and if so what are the hours?"
},
{
  "text": "Are you open on bank holidays?"
},
{
  "text": "Are you open during thanksgiving?"
},
{
  "text": "Are the stores open early?"
},
{
  "text": "will you open on christmas"
},
{
  "text": "how late y'all stay up till"
```

```
},
{
  "text": "how late are you there"
},
{
  "text": "how early do you open on Saturdays"
},
{
  "text": "Are you closed new Year's eve"
},
{
  "text": "Are you closing early today"
},
{
  "text": "What is the opening time for the washington store?"
},
{
  "text": "store open"
},
{
  "text": "store hours"
},
{
  "text": "store hrs"
},
{
  "text": "when can i visit your store"
},
{
  "text": "when does the store close"
},
{
  "text": "when do your stores open"
},
{
  "text": "open hours store"
},
        {
          "text": "store open now"
        }
      ],
      "description": "Find business hours."
    },
    {
      "intent": "Cancel",
      "examples": [
        {
          "text": "i changed my mind"
        },
        {
          "text": "cancel that"
        },
        {
          "text": "never mind"
        },
        {
          "text": "forget it"
        },
        {
          "text": "nevermind"
        },
        {
          "text": "i don't want a table anymore anymore"
        },
        {
          "text": "cancel the request"
        }
      ],
      "description": "Cancel the current request"
    },
    {
      "intent": "Help",
      "examples": [
        {
```

```
      "text": "what can i say"
    },
    {
      "text": "what can i do"
    },
    {
      "text": "i need assistance"
    },
    {
      "text": "help me"
    },
    {
      "text": "help me decide"
    },
    {
      "text": "help"
    },
    {
      "text": "can you assist me"
    },
    {
      "text": "can you help"
    }
  ],
  "description": "Ask for help"
},
{
  "intent": "Thanks",
  "examples": [
    {
      "text": "much appreciated"
    },
    {
      "text": "many thanks"
    },
    {
      "text": "that's nice of you"
```

```
        },
        {
          "text": "thank you very much"
        },
        {
          "text": "thx"
        },
        {
          "text": "i appreciate it"
        },
        {
          "text": "thank you"
        },
        {
          "text": "thanks"
        }
      ],
      "description": "Thanks"
    },
    {
      "intent": "General_Connect_to_Agent",
      "examples": [
        {
          "text": "Contact person"
        },
        {
          "text": "Could you please transfer me to your master?"
        },
        {
          "text": "Customer service representative please."
        },
        {
          "text": "Do not want a robot?"
        },
        {
          "text": "Hi can you transfer me"
        },
{
  "text": "How can I skip the recorded menu and go straight to a live person?"
},
{
  "text": "I don't want to talk to a bot."
},
{
  "text": "I don't want to talk to you"
},
{
  "text": "Is there anyone there I can actually talk to for real?"
},
{
  "text": "I need to speak to a representative. How would I go about doing so?"
},
{
  "text": "I want agent"
},
{
  "text": "I want a manager"
},
{
  "text": "I want an agent to help me"
},
{
  "text": "I want to talk to a person"
},
{
  "text": "I would like to speak to someone"
},
{
  "text": "Need help from human"
},
{
  "text": "Please assist me to get to an agent"
},
{
  "text": "Operator please"
},
{
  "text": "Please connect me to a live agent"
},
{
  "text": "Please let me talk to a human being."
},
{
  "text": "Pls connect"
},
{
  "text": "Put me through to someone"
},
{
  "text": "representative"
},
{
  "text": "I want to speak to a person"
},
{
  "text": "I would like to speak to a human"
},
{
  "text": "I dont want to talk to a computer"
},
{
  "text": "call the manager"
},
{
  "text": "I want to speak to a human"
},
{
  "text": "I want to talk to the manager"
},
{
  "text": "A real agent, please."
```

```
},
{
  "text": "Call agent"
},
{
  "text": "talk to a human"
},
{
  "text": "Agent help"
},
{
  "text": "Yes, take me to a real person"
},
{
  "text": "Where is the closest agent?"
},
{
  "text": "Send me to an agent"
},
{
  "text": "I don't want to speak with a robot"
},
{
  "text": "get me a person"
},
{
  "text": "Can I connect to an agent?"
},
{
  "text": "Can I speak to a human please?"
},
{
  "text": "Can I speak to a live person?"
},
{
  "text": "Can I speak to an advisor?"
},
        {
          "text": "Can I speak with somebody?"
        },
        {
          "text": "Can I talk to someone?"
        },
        {
          "text": "Can you assist me to connect to an agent?"
        },
        {
          "text": "Can you connect me with a real person?"
        },
        {
          "text": "Connect me to a live operator please."
        }
      ],
      "description": "Request a human agent."
    }
  ],
  "entities": [
    {
      "entity": "zip_code",
      "values": [
        {
          "type": "patterns",
          "value": "US Zip",
          "patterns": [
            "(\\b|\\s)\\d{5}(\\b|\\s)"
          ]
        }
      ]
    },
    {
      "entity": "sys-time",
      "values": []
    },
    {
```

```
"entity": "specialist",
"values": [
  {
    "type": "synonyms",
    "value": "Barbara",
    "synonyms": [
      "barbra"
    ]
  },
  {
    "type": "synonyms",
    "value": "Derrik",
    "synonyms": [
      "derek",
      "derik",
      "derrik",
      "derrick"
    ]
  },
  {
    "type": "synonyms",
    "value": "Brenda",
    "synonyms": []
  },
  {
    "type": "synonyms",
    "value": "Maria",
    "synonyms": []
  },
  {
    "type": "synonyms",
    "value": "Robert",
    "synonyms": [
      "bob"
    ]
  },
  {
      "type": "synonyms",
      "value": "Nicholas",
      "synonyms": [
        "nick"
      ]
    }
  ]
},
{
  "entity": "phone",
  "values": [
    {
      "type": "patterns",
      "value": "US Phone pattern",
      "patterns": [
        "(\\d{3})-(\\d{3})-(\\d{4})"
      ]
    }
  ]
},
{
  "entity": "sys-number",
  "values": []
},
{
  "entity": "holiday",

  "values": [
    {
      "type": "synonyms",
      "value": "valentine's day",
      "synonyms": [
        "valentine day",
        "valentines day"
      ]
    },
    {
```

```
  "type": "synonyms",
  "value": "labor day",
  "synonyms": []
},
{
  "type": "synonyms",
  "value": "christmas eve",
  "synonyms": [
    "x mas eve",
    "x-mas eve",
    "xmas eve"
  ]
},
{

  "type": "synonyms",
  "value": "new years eve",
  "synonyms": [
    "12-31",
    "12/31",
    "dec 31",
    "dec 31st",
    "new year's eve"
  ]
},
{
  "type": "synonyms",
  "value": "new years",
  "synonyms": [
    "1/1",
    "jan 1",
    "jan 1st",
    "jan first",
    "january 1",
    "january 1st",
    "january first",
    "new year",
    "new year day",
    "new years day"
  ]
},
{
  "type": "synonyms",
  "value": "memorial day",
  "synonyms": []
},
{
  "type": "synonyms",
  "value": "thanksgiving",
  "synonyms": [
    "turkey day"
  ]
},
{
  "type": "synonyms",
  "value": "halloween",
  "synonyms": []
},
{
  "type": "synonyms",
  "value": "christmas",
  "synonyms": [
    "christmas day",
    "x man day",
    "xmas",
    "x mas",
    "x-mas",
    "x-mas day",
    "xmas day"
  ]
},
{
  "type": "synonyms",
  "value": "independence day",
      "synonyms": [
        "7/4",
        "fourth of july",
        "july 4",
        "july 4th",
        "july fourth"
      ]
    }
  ]
},
{
  "entity": "sys-date",
  "values": []
},
{
  "entity": "landmark",
  "values": [
    {
      "type": "synonyms",
      "value": "grand central",
      "synonyms": []
    },
    {
      "type": "synonyms",
      "value": "empire state building",
      "synonyms": [
        "empire state",
        "emprire state"
      ]
    },
    {
      "type": "synonyms",
      "value": "times square",
      "synonyms": [
        "time sqaure",
        "time square",
        "times sqaure"
```

```json
          ]
        }
      ],
      "fuzzy_match": true
    },
    {
      "entity": "reply",
      "values": [
        {
          "type": "synonyms",
          "value": "yes",
          "synonyms": [
            "definitely",
            "go for it",
            "let's do it",
            "ok",
            "please",
            "sure",
            "why not",
            "yeah",
            "yes",
            "you bet",
            "you betcha",
            "yep"
          ]
        },
        {
          "type": "synonyms",
          "value": "no",
          "synonyms": [
            "definitely not",
            "don't think so",
            "dont think so",
            "i think not",
            "nope",
            "not at this time",
            "not now"
          ]
        }
      ]
    }
  ],
  "metadata": {
    "api_version": {
      "major_version": "v2",
      "minor_version": "2018-11-08"
    }
  },
  "webhooks": [
    {
      "url":
"https://eu-gb.functions.cloud.ibm.com/api/v1/web/si05202000490%40smartinternz.com_dev/default
/discovery%20function.json",
      "name": "main_webhook",
      "headers": []
    }
  ],
  "dialog_nodes": [
    {
      "type": "event_handler",
      "output": {
        "text": {
          "values": [
            "We only accept appointments between 11am and 5pm"
          ]
        }
      },
      "parent": "slot_105_1498132552870",
      "metadata": {},
      "next_step": {
        "behavior": "reprompt"
      },
      "conditions": "$time.after('17:30:30') || $time.before('10:59:59')",
      "event_name": "filled",
      "dialog_node": "handler_1_1509694458589",
      "previous_sibling": "handler_106_1498132552870"
    },
    {
      "type": "event_handler",
      "output": {},
      "parent": "slot_105_1498132552870",
      "context": {
        "time": "@sys-time"
      },
      "metadata": {},
      "conditions": "@sys-time",
      "event_name": "input",
      "dialog_node": "handler_106_1498132552870",
      "previous_sibling": "handler_107_1498132552870"
    },
    {
      "type": "event_handler",
      "output": {
        "text": "What time on <? $date.reformatDateTime(\"EEEEE\") ?> do you want to come in?"
      },
      "parent": "slot_105_1498132552870",
      "metadata": {},
      "event_name": "focus",
      "dialog_node": "handler_107_1498132552870"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "We are open on @holiday regular hours"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Hours of Operation",
      "context": {},
      "metadata": {},
      "conditions": "@holiday",
      "dialog_node": "node_5_1482426503106",
      "previous_sibling": "node_1_1522387330204"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "We are open on <? @sys-date.reformatDateTime(\"EEEEE\") ?> from 10am until 8pm"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Hours of Operation",
      "context": {},
      "metadata": {},
      "conditions": "@sys-date.reformatDateTime(\"EEEEE\") == \"Monday\" ||
@sys-date.reformatDateTime(\"EEEEE\") == \"Tuesday\" || @sys-date.reformatDateTime(\"EEEEE\")
== \"Wednesday\" || @sys-date.reformatDateTime(\"EEEEE\") == \"Thursday\" ||
@sys-date.reformatDateTime(\"EEEEE\") == \"Friday\"",
      "dialog_node": "node_1_1522387330204",
      "previous_sibling": "node_4_1482425833988"
    },
    {
      "type": "response_condition",
      "output": {

        "text": {
          "values": [
            "Our hours on <? @sys-date.reformatDateTime(\"EEEEE\") ?> are 11am to 6pm."
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Hours of Operation",
      "context": {},
      "metadata": {},
      "conditions": "@sys-date.reformatDateTime(\"EEEEE\") == \"Saturday\" ||
@sys-date.reformatDateTime(\"EEEEE\") == \"Sunday\"",
      "dialog_node": "node_2_1482424204936",
      "previous_sibling": "node_5_1482426503106"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "Our hours are Monday to Friday 10am to 8pm and Friday and Saturday 11am to 6pm."
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Hours of Operation",
      "context": {},
      "metadata": {},
      "conditions": " true",
      "dialog_node": "node_6_1482426521282",
      "previous_sibling": "node_2_1482424204936"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "We are closed on @holiday"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Hours of Operation",
      "context": {},
      "metadata": {},
      "conditions": "@holiday:christmas || @holiday:thanksgiving || @holiday:(new years)",
      "dialog_node": "node_4_1482425833988"
```

```
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "please try again later"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_5_1591242858609",
  "conditions": "anything_else",
  "dialog_node": "response_4_1591242892865",
  "previous_sibling": "response_8_1591242891061"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "\"<?$webhook_result_1.passages[0].passage_text?>\""
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_5_1591242858609",
          "conditions": "$webhook_result_1",
          "dialog_node": "response_8_1591242891061"
        },
        {
          "type": "standard",
          "output": {
            "text": "OK. Let me know how I can help"
          },
          "parent": "node_22_1467833484410",
          "metadata": {},
          "conditions": "@reply:no",
          "dialog_node": "node_21_1468350173406",
          "previous_sibling": "node_19_1468350024009"
        },
        {
 "e
 nt
 it
 y"
 :
 "h
 ol
 id
 ay
 ",

          "type": "standard",
          "output": {
            "text": {
              "values": [
                "OK. Transferring... [Use IBM Cloud Functions to connect to backend systems]"
              ]
            }
          },
          "parent": "node_22_1467833484410",
          "metadata": {},
          "conditions": "@reply:yes",
          "dialog_node": "node_19_1468350024009"
        },
        {
```

```
  "type": "response_condition",
  "output": {
    "text": {
      "values": [
        "Hello. Good evening",
        "Hi. Good evening",
        "Hello. How can I help you?"
      ],
      "selection_policy": "sequential"
    }
  },
  "parent": "node_13_1502484041694",
  "metadata": {},
  "conditions": "now().after('17:00:00')",
  "dialog_node": "node_15_1488295465298",
  "previous_sibling": "node_1_1495022305143"
},
{
  "type": "response_condition",
  "output": {
    "text": {
      "values": [
        "Hello. Good afternoon",
        "Hi there. How can I help you",
        ""
      ],
      "selection_policy": "sequential"
    }
  },
  "parent": "node_13_1502484041694",
  "metadata": {},
  "conditions": "now().after('12:00:00') && now().before('17:00:00')",
  "dialog_node": "node_1_1495022305143",
  "previous_sibling": "node_16_1488295517679"
},
{
  "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "Hello",
            "Hi there",
            "Hi. How can I help"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "node_13_1502484041694",
      "metadata": {},
      "dialog_node": "node_28_1522448362216",
      "previous_sibling": "node_15_1488295465298"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "Hello. Good morning",
            "Hello.How can I help You",
            ""
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "node_13_1502484041694",
      "metadata": {},
      "conditions": "now().after('04:00:00') && now().before('12:00:00')",
      "dialog_node": "node_16_1488295517679"
    },
    {
      "type": "event_handler",
      "output": {
        "text": {
          "values": [
        "We'll do our best to book you with @specialist"
      ],
      "selection_policy": "sequential"
    }
  },
  "parent": "slot_12_1522596437268",
  "event_name": "filled",
  "dialog_node": "handler_15_1522596463593",
  "previous_sibling": "handler_13_1522596437268"
},
{
  "type": "event_handler",
  "output": {},
  "parent": "slot_12_1522596437268",
  "context": {
    "specialist": "@specialist"
  },
  "metadata": {},
  "conditions": "@specialist",
  "event_name": "input",
  "dialog_node": "handler_13_1522596437268",
  "previous_sibling": "handler_14_1522596437268"
},
{
  "type": "event_handler",
  "output": {},
  "parent": "slot_12_1522596437268",
  "event_name": "focus",
  "dialog_node": "handler_14_1522596437268"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
        "Sorry... let's try again"
      ]
    }
  },
  "parent": "slot_8_1509132875735",
  "context": {
    "date": null,
    "time": null,
    "confirm": null
  },
  "metadata": {},
  "conditions": "@reply:no",
  "event_name": "filled",
  "dialog_node": "handler_17_1509135162089",
  "previous_sibling": "handler_14_1509133469904"
},
{
  "type": "event_handler",
  "output": {},
  "parent": "slot_8_1509132875735",
  "context": {
    "confirm": "@reply && slot_in_focus"
  },
  "metadata": {},
  "conditions": "@reply && slot_in_focus",
  "event_name": "input",
  "dialog_node": "handler_9_1509132875735",
  "previous_sibling": "handler_10_1509132875735"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
        "Perfect!"
      ]
    }
  },
  "parent": "slot_8_1509132875735",
      "metadata": {},
      "conditions": "@reply:yes",
      "event_name": "filled",
      "dialog_node": "handler_14_1509133469904",
      "previous_sibling": "handler_9_1509132875735"
    },
    {
      "type": "event_handler",
      "output": {
        "text": "Let me confirm: You want an appointment for <?
$date.reformatDateTime(\"EEEEE\") ?> at <? $time.reformatDateTime(\"h a\") ?>. Is this
correct?"
      },
      "parent": "slot_8_1509132875735",
      "metadata": {},
      "event_name": "focus",
      "dialog_node": "handler_10_1509132875735"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "To get to our business from the Empire State Building, walk to Herald Square and
take the N train to Union Square"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Directions",
      "metadata": {},
      "conditions": "@landmark:(empire state building)",
      "dialog_node": "node_7_1482459200886",
      "previous_sibling": "node_3_1522439390442"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "To get to our business from Times Square, take the N train downtown to Union Square"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Directions",
      "metadata": {},
      "conditions": "@landmark:(times square)",
      "dialog_node": "node_8_1482459217052",
      "previous_sibling": "node_7_1482459200886"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "To get to our business from Grand Central, take the 4,5 or 6 train downtown to Union Square."
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Directions",
      "metadata": {},
      "conditions": "@landmark:(grand central)",
      "dialog_node": "node_4_1522439442155",
      "previous_sibling": "node_8_1482459217052"
    },
    {
      "type": "standard",
      "title": "Provide location",
      "output": {
        "text": {
          "values": [
            "We're located by Union Square on the corner of 13th and Broadway"
      ],
      "selection_policy": "sequential"
    }
  },
  "parent": "Directions",
  "metadata": {},
  "conditions": "true",
  "dialog_node": "node_3_1522439390442"
},
{
  "type": "response_condition",
  "output": {
    "text": {
      "values": [
        "[Use IBM Cloud Functions to connect to to backend systems]"
      ]
    }
  },
  "parent": "Reservation using slots",
  "metadata": {},
  "conditions": "$user_needs_help",
  "dialog_node": "node_25_1522598839584",
  "previous_sibling": "handler_7_1509696539866"
},
{
  "type": "slot",
  "output": {},
  "parent": "Reservation using slots",
  "metadata": {},
  "variable": "$specialist",
  "dialog_node": "slot_12_1522596437268",
  "previous_sibling": "slot_105_1498132552870"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": []
    }
  },
  "parent": "Reservation using slots",
  "disabled": true,
  "metadata": {},
  "event_name": "focus",
  "dialog_node": "handler_7_1509696539866",
  "previous_sibling": "handler_16_1509133697261"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
        "OK. Canceling your request..."
      ]
    }
  },
  "parent": "Reservation using slots",
  "context": {
    "date": null,
    "time": null,
    "phone": null,
    "confirm": null,
    "specialist": null,
    "user_cancelled": true
  },
  "metadata": {},
  "next_step": {
    "behavior": "skip_all_slots"
  },
  "conditions": "#Cancel",
  "event_name": "generic",
  "dialog_node": "handler_16_1509133697261",
  "previous_sibling": "handler_3_1501275087289"
},
{
  "type": "slot",
  "output": {},
  "parent": "Reservation using slots",
  "metadata": {
    "_customization": {
      "mcr": true
    }
  },
  "variable": "$date",
  "dialog_node": "slot_102_1498132501942",
  "previous_sibling": "node_3_1519173961259"
},
{
  "type": "slot",
  "output": {},
  "parent": "Reservation using slots",
  "metadata": {},
  "variable": "$confirm",
  "dialog_node": "slot_8_1509132875735",
  "previous_sibling": "slot_12_1522596437268"
},
{
  "type": "slot",
  "output": {},
  "parent": "Reservation using slots",
  "metadata": {
    "_customization": {}
  },
  "variable": "$phone",
  "dialog_node": "slot_22_1522444583114",
  "previous_sibling": "slot_8_1509132875735"
},
{
  "type": "response_condition",
  "output": {
    "text": {
          "values": [
            "Let me know how else I can help"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "Reservation using slots",
      "context": {},
      "metadata": {},
      "conditions": "$user_cancelled",
      "dialog_node": "node_10_1509697567474",
      "previous_sibling": "node_25_1522598839584"
    },
    {
      "type": "response_condition",
      "output": {
        "text": {
          "values": [
            "Let me check availability...  [Use IBM Cloud Functions to connect to backend systems]"
          ]
        }
      },
      "parent": "Reservation using slots",
      "context": {},
      "metadata": {},
      "conditions": "true",
      "dialog_node": "node_3_1519173961259",
      "previous_sibling": "node_10_1509697567474"
    },
    {
      "type": "slot",
      "output": {},
      "parent": "Reservation using slots",
      "metadata": {
        "_customization": {
          "mcr": true
        }
```

```
  },
  "variable": "$time",
  "dialog_node": "slot_105_1498132552870",
  "previous_sibling": "slot_102_1498132501942"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
        "I see you need help making an appointment. Let me transfer you to an agent..."
      ],
      "selection_policy": "sequential"
    }
  },
  "parent": "Reservation using slots",
  "context": {
    "date": null,
    "time": null,
    "phone": null,
    "confirm": null,
    "specialist": null,
    "user_needs_help": true
  },
  "metadata": {},
  "next_step": {
    "behavior": "skip_all_slots"
  },
  "conditions": "#Help",
  "event_name": "generic",
  "dialog_node": "handler_3_1501275087289"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
              "Looks like you're trying to make a reservation in the past. Try again."
            ]
          }
        },
        "parent": "slot_102_1498132501942",
        "metadata": {},
        "next_step": {
          "behavior": "reprompt"
        },
"text": "What is the nearest branch?"
        "conditions": "$date.before(now())",
        "event_name": "filled",
        "dialog_node": "handler_6_1509695999145",
        "previous_sibling": "handler_103_1498132501942"
      },
      {
        "type": "event_handler",
        "output": {},
        "parent": "slot_102_1498132501942",
        "context": {
          "date": "@sys-date"
        },
        "metadata": {},
        "conditions": "@sys-date",
      "event_name": "input",
      "dialog_node": "handler_103_1498132501942",
      "previous_sibling": "handler_104_1498132501942"
    },
    {
      "type": "event_handler",
      "output": {
        "text": "What day would you like to come in?"
      },
      "parent": "slot_102_1498132501942",
      "metadata": {},
      "event_name": "focus",
      "dialog_node": "handler_104_1498132501942"
    },
    {
      "type": "event_handler",
      "output": {
        "text": {
          "values": [
            "Thanks"
          ],
          "selection_policy": "sequential"
        }
      },
      "parent": "slot_22_1522444583114",
      "context": {},
      "metadata": {},
      "conditions": "true",
      "event_name": "filled",
      "dialog_node": "handler_22_1522598191131",
      "previous_sibling": "handler_23_1522444583114"
    },
    {
      "type": "event_handler",
      "output": {},
      "parent": "slot_22_1522444583114",
      "context": {
```

```
    "phone": "@phone"
  },
  "metadata": {},
  "conditions": "@phone",
  "event_name": "input",
  "dialog_node": "handler_23_1522444583114",
  "previous_sibling": "handler_24_1522444583114"
},
{
  "type": "event_handler",
  "output": {
    "text": "I'll just need a phone to hold your reservation"
  },
  "parent": "slot_22_1522444583114",
  "metadata": {},
  "event_name": "focus",
  "dialog_node": "handler_24_1522444583114"
},
{
  "type": "standard",
  "title": "Where are you located?",
  "output": {},
  "metadata": {},
  "next_step": {
    "behavior": "skip_user_input"
  },
  "conditions": "#Customer_Care_Store_Location",
  "digress_in": "does_not_return",
  "dialog_node": "Directions",
  "digress_out": "allow_all",
  "previous_sibling": "Hours of Operation"
},
{
  "type": "standard",
  "title": "Please transfer me to an agent",
  "output": {
    "text": {
          "values": [
            "Would you like me to transfer you to a representative?"
          ],
          "selection_policy": "sequential"
        }
      },
      "metadata": {},
      "conditions": "#General_Connect_to_Agent",
      "digress_in": "does_not_return",
      "dialog_node": "node_22_1467833484410",
      "digress_out": "allow_all_never_return",
      "previous_sibling": "node_2_1468243505617"
    },
    {
      "type": "standard",
      "output": {
        "text": {
          "values": [
            "I didn't understand can you try again"
          ],
          "selection_policy": "sequential"
        }
      },
      "metadata": {},
      "conditions": "anything_else",
      "digress_in": "returns",
      "dialog_node": "node_2_1467831978407",
      "digress_out": "allow_all",
      "previous_sibling": "node_5_1591242858609",
      "disambiguation_opt_out": true
    },
    {
      "type": "standard",
      "title": "What can I do",
      "output": {
        "generic": [
          {
            "values": [
              {
                "text": "I can tell you about our store locations and opening hours, or help you set up an appointment."
              },
              {
                "text": "You could also ask me to connect you to an agent."
              }
            ],
            "response_type": "text",
            "selection_policy": "sequential"
          }
        ]
      },
      "conditions": "#Help",
      "dialog_node": "node_4_1570050459690",
      "previous_sibling": "node_22_1467833484410"
    },
    {
      "type": "standard",
      "output": {
        "text": {
          "values": [
            "So long",
            "See ya",
            "Good bye"
          ],
          "selection_policy": "sequential"
        }
      },
      "metadata": {},
      "conditions": "#Goodbye",
      "digress_in": "does_not_return",
      "dialog_node": "node_12_1468329566917",
      "previous_sibling": "node_13_1502484041694"
    },
    {
      "type": "frame",
      "title": "I want to make an appointment",
      "output": {},
      "metadata": {
        "fallback": "leave",
        "_customization": {
          "mcr": true
        }
      },
      "conditions": "#Customer_Care_Appointments",
      "digress_in": "does_not_return",
      "dialog_node": "Reservation using slots",
      "digress_out": "allow_all",
      "previous_sibling": "Directions",
      "digress_out_slots": "allow_all"
    },
    {
      "type": "standard",
      "output": {
        "text": {
          "values": [
            "You're welcome. Just let me know if you need anything else",
            "No problem. Just let me know if you need anything else",
            "My pleasure. Just let me know if you need anything else"
          ],
          "selection_policy": "sequential"
        }
      },
      "metadata": {},
      "conditions": "#Thanks",
      "digress_in": "does_not_return",
      "dialog_node": "node_2_1468243505617",
      "previous_sibling": "node_12_1468329566917"
    },
    {
      "type": "standard",
      "output": {},
      "metadata": {},
      "conditions": "#General_Greetings",
      "digress_in": "does_not_return",
      "dialog_node": "node_13_1502484041694",
      "previous_sibling": "Reservation using slots"
    },
    {
      "type": "standard",
      "title": "Ask about product",
      "actions": [
        {
          "name": "main_webhook",
          "type": "webhook",
          "parameters": {
            "input": "<?input.text?>"
          },
          "result_variable": "webhook_result_1"
        }
      ],
      "metadata": {
        "_customization": {
          "mcr": true
        }
      },
      "conditions": "#Product_Information",
      "dialog_node": "node_5_1591242858609",
      "previous_sibling": "node_4_1570050459690"
    },
    {
      "type": "standard",
      "title": "What are your hours?",
      "output": {},
      "metadata": {},
      "next_step": {
        "behavior": "jump_to",
        "selector": "body",
        "dialog_node": "node_3_1522439390442"
      },
      "conditions": "#Customer_Care_Store_Hours",
      "digress_in": "does_not_return",
      "dialog_node": "Hours of Operation",
      "digress_out": "allow_all",
      "previous_sibling": "Opening"
    },
    {
      "type": "standard",
      "title": "Opening",
      "output": {
        "text": {
          "values": [
            "Hello, Welcome to the Customer Care HelpDesk.How may I help you?"
          ],
          "selection_policy": "sequential"
        }
      },
      "context": {
        "no_reservation": true
      },
      "metadata": {},
      "conditions": "welcome",
      "dialog_node": "Opening"
    }
  ],
  "counterexamples": [],
  "system_settings": {
    "tooling": {
      "store_generic_responses": true
    },
    "off_topic": {
      "enabled": true
    },
    "disambiguation": {
      "prompt": "Did you mean:",
      "enabled": true,
      "randomize": true,
      "max_suggestions": 5,
      "suggestion_text_policy": "title",
      "none_of_the_above_prompt": "None of the above."
    },
    "system_entities": {
      "enabled": true
    },
    "spelling_auto_correct": true
  },
  "learning_opt_out": false,
  "name": "Customer Care Sample Skill",
  "language": "en",
  "description": "Sample simple customer service skill to get you started."
}
```

NodeRedFlow.json

```
[
        {
            "id": "df2dfd54.c87f7",
            "type": "tab",
            "label": "Flow 1",
            "disabled": false,
            "info": ""
        },
        {
            "id": "dd950266.b9191",
            "type": "ui_form",
            "z": "df2dfd54.c87f7",
            "name": "",
            "label": "",
            "group": "fe4fd38a.9460a",
            "order": 1,
            "width": 0,
            "height": 0,
            "options": [
                {
```

```
                "label": "Enter your input",
                "value": "text",
                "type": "text",
                "required": true,
                "rows": null
            }
        ],
        "formValue": {
            "text": ""
        },
        "payload": "",
        "submit": "submit",
        "cancel": "cancel",
        "topic": "",
        "x": 240,
        "y": 320,
        "wires": [
            [
                "b5648aab.dd0ea8"
            ]
        ]
    },
    {
        "id": "b5648aab.dd0ea8",
        "type": "function",
        "z": "df2dfd54.c87f7",
        "name": "",
        "func": "msg.payload=msg.payload.text\nreturn msg;",
        "outputs": 1,
        "noerr": 0,
        "x": 400,
        "y": 200,
        "wires": [
            [
                "117723c4.e3aafc",
                "220ea9df.332416"
            ]
        ]
    },
    {
        "id": "29d2c28c.a9196e",
        "type": "function",
        "z": "df2dfd54.c87f7",
        "name": "",
        "func": "msg.payload=msg.payload.output.text[0];\nreturn msg;",
        "outputs": 1,
        "noerr": 0,
        "x": 870,
        "y": 200,
        "wires": [
            [
                "7f2579ea.cb5478"
            ]
        ]
    },
    {
        "id": "117723c4.e3aafc",
        "type": "watson-conversation-v1",
        "z": "df2dfd54.c87f7",
        "name": "Customer Care",
        "workspaceid": "f1f3bb87-d962-4c9b-87f7-b49d6cab539e",
        "multiuser": false,
        "context": true,
        "empty-payload": false,
        "service-endpoint": "https://api.eu-gb.assistant.watson.cloud.ibm.com/instances/eeb79120-927f-4b50-9d18-8391fcc0d0fd",
        "timeout": "",
        "optout-learning": false,
        "x": 660,
        "y": 140,
        "wires": [
            [
                "fa5a802.2ea968",
                "29d2c28c.a9196e"
        ]
    ]
},
{
    "id": "220ea9df.332416",
    "type": "ui_text",
    "z": "df2dfd54.c87f7",
    "group": "fe4fd38a.9460a",
    "order": 2,
    "width": 0,
    "height": 0,
    "name": "",
    "label": "You",
    "format": "{{msg.payload}}",
    "layout": "row-left",
    "x": 570,
    "y": 320,
    "wires": []
},
{
    "id": "7f2579ea.cb5478",
    "type": "ui_text",
    "z": "df2dfd54.c87f7",
    "group": "fe4fd38a.9460a",
    "order": 3,
    "width": 13,
    "height": 6,
    "name": "",
    "label": "Bot",
    "format": "{{msg.payload}}",
    "layout": "col-center",
    "x": 970,
    "y": 340,
    "wires": []
},
{
    "id": "fa5a802.2ea968",
        "type": "debug",
        "z": "df2dfd54.c87f7",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "false",
        "x": 880,
        "y": 80,
        "wires": []
    },
    {
        "id": "fe4fd38a.9460a",
        "type": "ui_group",
        "z": "",
        "name": "CHATBOT",
        "tab": "1bff366.44c80ca",
        "order": 1,
        "disp": true,
        "width": "13",
        "collapse": false
    },
    {
        "id": "1bff366.44c80ca",
        "type": "ui_tab",
        "z": "",
        "name": "CUSTOMER CARE HELPDESK",
        "icon": "dashboard",
        "disabled": false,
        "hidden": false
    }
]
```

THE END