

Project Report

Topic: Intelligent Customer Helpdesk with Smart Document Understanding

Category: Machine Learning / Artificial Intelligence

Application ID: SPS_APL_20200000656

Project ID: SPS_PRO_99

Internship at: SMARTINTERNZ

Completed by: Sayak Banerjee

e-mail id: sayak.banerjee.ece21@heritageit.edu.in

Contents:

1. INTRODUCTION

1.1. Overview

1.2. Purpose

2. Analysis Report

2.1. Present Problem

2.2. Proposed Solution

3. THEORETICAL ANALYSIS

3.1. Blockdiagram

3.2. Hardware/Software requirements

4. PROCEDURAL ANALYSIS

5. FLOWCHART

6. RESULTS

7. ADVANTAGES AND DISADVANTAGES

8. APPLICATION

9. CONCLUSION

10. FUTURESCOPE

11. BIBLIOGRAPHY

APPENDIX

SourceCode

1. INTRODUCTION

1.1.Overview

A customer helpdesk, chatbot, is designed for Ecobee3. Ecobee3 is a smart thermostat which is a smart home device. The chatbot helps the user to get information by asking a question. This chatbot accesses an unstructured document (user manual) of Ecobee3, by smart understanding of the document it replies to any query related to Ecobee3.

1.2.Purpose

The main aim is to enhance the efficiency of the chatbot (CustomerHelpdesk) by incorporating Smart Document Understanding. So that all questions get an answer.

2. Analysis Report

2.1.PresentProblem

The general chatbots are able to help customers with specific queries ,like "What is the store location?" , "What is the operation time?" , "Book me an appointment." and soon. They are unable to give details about a particular device and its operations .When the question is outside the scope it is unable to reply.

2.2.ProposedSolution

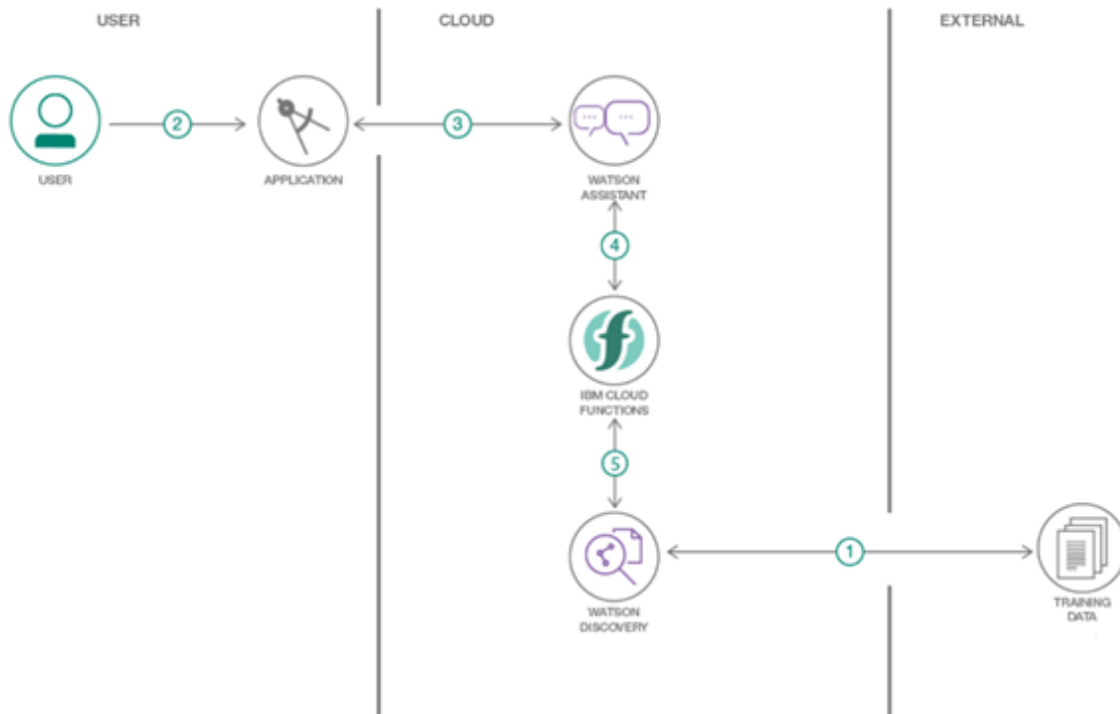
This project deals with those queries which are out of the set of predefined queries. One of the major part of these kind of queries are about a device specification and handling assistance of the device.

Usually, facts about operating a device is available in the user manual. With the help of Watson Discovery (an IBM cloud service), we are able to smartly analyze the user manual to obtain the important parts of the manual and divide it subtitle wise.

Therefore, any query regarding the operations of the device gets a reply from the Helpdesk after it has accessed the user manual using Smart Document Understanding.

3. THEORITICAL ANALYSIS

3.1. Block Diagram



3.2. Hardware/Software Requirements

We have to build an IBM Cloud account. By creating IBM Cloud account we are eligible to create various IBM Services. For this project, we are required to build Watson Discovery Service, IBM Cloud function action, Watson Assistant Service and Node-Red App. Also, a document of the user manual of ecobee3 is needed, which is fed to Watson Discovery Service.

4. PROCEDURAL ANALYSIS

1. Create an IBM Cloud account.

To create IBM Cloud account, go to <https://www.ibm.com/cloud>

Click on [Create an account](#).

If an account exists then log in with the IBMid and password.

After logging in, check Dashboard for present Services.

2. Create Watson Discovery.

Open the catalog option in IBM Cloud account and select [Services](#).

Under the Artificial Intelligence(AI) section, Discovery Service can be found.

Select a region and plan and configure the service. Then click on [Create](#).

Discovery Service will be created successfully.

Copy the "url" and "api key" value, these will be required in cloud function action.

Then click on [Launch Watson Discovery](#) to launch the Discovery Service.

1. Create Watson Assistant.

Open the catalog option in IBM Cloud account and select [Services](#).

Under the Artificial Intelligence(AI) section, Watson Assistant Service can be found.

Select a region and plan and configure the service. Then click on [Create](#).

Assistant Service will be created successfully.

Copy the "url" and "api key" values, these will be required in Node-Red flow.

Then click on [Launch Watson Assistant](#) to launch the Assistant Service.

3. Configure Watson Discovery

IBM Watson Discovery when launched.

Click on [Upload your own data](#) to upload a new dataset (any unstructured file).

Go to [Configure data](#) after opening the dataset.

In [Identify labels](#), label the data on the basis of fields like "title", "subtitle", "text", "footer" and so on.

Then the document is split in the [Manage Fields](#) on the basis of "subtitle" in this case, by selecting [split document on each occurrence of "subtitle"](#).

Then click [Apply changes to collection](#) and upload the original document to save the changes.

The [Overview](#) page of the document after field identification and splitting.

Now we see that the original document is split into 121 documents. Also, Sentiment Analysis is seen. [Build your own query](#) helps to test the accuracy.

Copy the api values of "collection ID" and "environment ID", these will be required in the cloud function action (parameter).

1. Create IBM Cloud function action.

The web action that will make queries against the Discovery collection (data) is built. Go to IBM Cloud Dashboard and click on [Create Resources](#) then select functions.

Click on [Actions](#) on the left panel. Then select [Create](#) and provide a unique action name.

Click on Code tab. A certain code is written which, connects function to the Discovery Service, makes the query against the collection and returns the result.

The code is provided under the name of disco-action.js in Source Codes (Appendix).

Click on [Parameters](#) tab.

The values of following parameters, "url", "environment_id", "collection_id" and "iam_apikey" can be accessed from the Discovery Service. This helps the action to connect to the Discovery Service.

Click on [Endpoints](#) tab.

Click on [Enable as Web Action](#). This generates an URL of cloud function action. Copy this URL, this will be required in Watson Assistant as webhooks.

4. Configure Watson Assistant.

IBM Watson Assistant when launched.

Then go to [Skills](#) tab and select [Create Skills](#).

Two types of skills can be made, Dialog and Search (only for Plus plan). Select [Dialog Skill](#).

Select [Create skill](#) and provide a name and description of the skill. Skills can also be imported using [Import skills](#) and default predefined skill can be used using [Use sample skill](#).

[Intents](#) is a goal or purpose of user's questions. [Entities](#) is a specific detail of questions and statements. [Dialogue](#) puts intent and entity together to provide interaction.

We create respective Intents and Entity. Also, a dialogue flow is created.

The dialogue box "Product_details" is for customer interaction using the user manual. In [Skills](#) tab, click on [View API details](#) of the skill. Copy the "Skill ID", this will be required in Node-RED flow.

Open the [Assistants](#) tab and click on [Create Assistant](#).

Give a name to the assistant. Add the created skill to the assistant.

So, the assistant acquires the skill created and is able to answer the queries.

To turn on Webhooks, click on [Options](#).

Paste the url of cloud function action into the Webhooks' url section. It is a must to end the url in webhook with ".json". Therefore, now the assistant is linked to the cloud

function using Webhooks. Whereas the cloud function action is connected to the Watson Discovery Service. Thus, the assistant is indirectly connected to Discovery Service using Cloud function.

Customize the "Product_details" dialogue and turn on the Webhooks option.

The query of the user is passed through the key "input" to Discovery Service. The response from Discovery service is captured in variable "\$webhook_result_1".

5. Build Node-RED flow to integrate all services together.

Go to [Create Resource](#) and search for [Node-RED App](#). Open the [Node-RED App](#).

Provide a unique name for the app or accept the default name. Provide IBM Cloud API key or generate a new one. Also provide memory per instance (in this case 256Mb). Select region to create the DevOps toolchain and select [Create](#).

Initially, the [Status](#) in Delivery Pipelines will show "in progress". The Deploy stage will take time to get completed. Once the Deploy stage is passed, the status will turn to "success".

The details of the application page after deploy stage is passed.

The newly created Node-RED App will be listed under the [App Section](#) in [Resource list](#). Also a corresponding entry under the [Cloud Foundry App](#) section can be seen.

Click to open the [App URL](#). The Node-RED editor will open.

The Node-RED editor shows a basic flow (Flow 2), where the output can be seen in the debug window (on the right). On the left is the palette.

The nodes present in this flow are "inject" and "debug".

6. Build a web dashboard using Node-Red flow.

Go to [Settings](#) and then [Install Tab](#) and search for [Node-red-dashboard](#). Install the [Node-red-dashboard](#).

By installing node-red-dashboard, "form" node becomes available.

The "assistant" node is used to link the flow with the assistant (Customer Help Desk).

The Workspace ID, Service endpoint and API key are the Skill ID, URL and API key respectively of the Watson Assistant.

Taking various other nodes from pallette, the desired web dashboard is built.

"form" node helps to built the query section for the user.

"text" node is named as "QUESTION" which displays the query.

"template" node is named as "ANSWER" which displays the answer of the query.

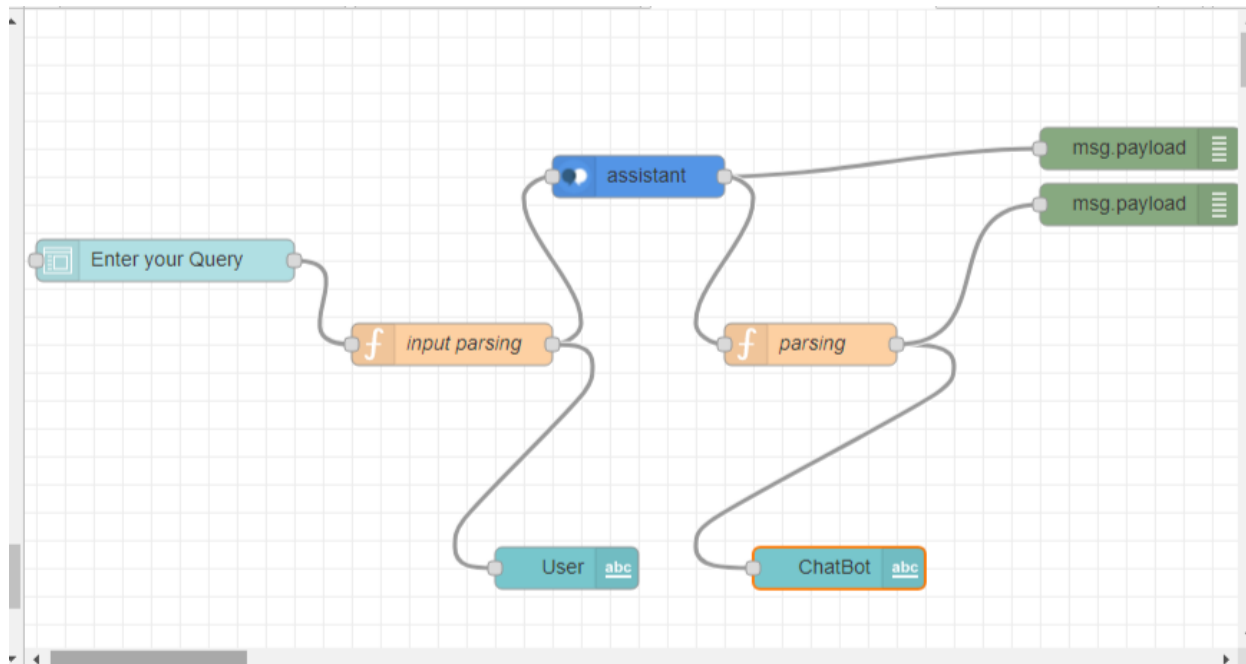
"assistant" node links the flow with the Watson Assistant (Customer Help Desk) to feed questions and obtain answer from it.

There are two "function" nodes, one named as "input parsing" for feeding the question text to the assistant and the other named as "parsing" which converts output of assistant from json to text format.

"debug" node is used to help the designer debug the outputs.

The two "function" nodes have to be coded in order to extract text from json. The code is given in Source Code (Appendix)

5. FLOWCHART



6. RESULTS

Response of Customer Help Desk to few demo queries are as follows-

Customer Help Desk

Enter your Query

Hi

SUBMIT

CANCEL

User

Hi

ChatBot

Hello. Good morning

Customer Help Desk

Enter your Query

How to turn on heater?

SUBMIT

CANCEL

User

How to turn on heater?

ChatBot

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 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 rises above the set temperature. ☐ Heat: Turn on the heat when the current temperature drops below the set temperature. ☐ Auto: Activate the heating or cooling system as required to keep your home within the configured range of set temperatures. ☐ Aux: Only use the auxiliary or backup heat source to maintain the heat set point temperature. This option only appears if auxiliary heat is configured in the Equipment menu. ☐ Off: Turn the system off. When the system is off, only the current temperature will be displayed on the Home screen. On Thermostat and Mobile: Select Main Menu > System > HVAC System On Web: Select System tile > HVAC

Customer Help Desk

Enter your Query

How to clean thermostat?

SUBMIT

CANCEL

User

How to clean thermostat?

ChatBot

You can clean your ecobee3 screen by spraying water on to a clean, microfiber cloth. Wipe the surface of the screen with the dampened cloth. Note: Never spray any liquids directly onto your ecobee3. Generates an alert that indicates it is time to clean or replace the filter on the furnace, if installed. You can set the Last Filter Change date, turn the Reminder On or Off, and set the Frequency of the maintenance interval. If the HVAC system uses a UV lamp for air purification, your ecobee3 can generate an alert indicating that it is time to clean or replace the lamp. You can set the Last Lamp Change date, turn the Reminder On or Off, and set the Frequency of the maintenance interval.

Activate Windows

Customer Help Desk

How to change brightness?

SUBMIT

CANCEL

User

How to change brightness?

ChatBot

You can customize the brightness of your ecobee3's screen. The brightness for both the active and standby screens can be configured independently. You can also configure the screen to automatically sleep (i.e. turn off) whenever your ecobee3 enters the Sleep activity period. For example, if your thermostat is located in a bedroom, you may want to blank the screen when you are sleeping, whereas if the thermostat is in a hallway, you may want the screen displayed all the time. On Thermostat: 1. Select Main Menu > Settings > Preferences 2. Select Screen brightness. 3. Adjust the values of the Active and Standby screen brightness. 4. Select Screen sleeps when I sleep if you want to make the screen blank during the Sleep activity period. You can configure how long a manual change to the set point will remain in effect. On Thermostat: 1. Select Main Menu > Settings > Preferences 2. Select Hold action. 3. Select the hold action from the list: ☐ 2 hours ☐ 4 hours ☐ Until the next scheduled activity ☐ Until you change it (default value) Decide at time of change On Web: 1. Select Settings tile. 2. Select Preferences. 3. Select Hold Action. 4. Select the hold action from the list: ☐ 2 hours ☐ 4 hours ☐ Until the next scheduled activity ☐ Until you change it (default value) ☐ Decide at time of change The standby screen activates whenever the thermostat is not in use. It shows the current indoor temperature and outdoor weather conditions. 1 Current indoor temperature 2 Current outdoor weather conditions The standby screen is configurable. You can adjust: ☐ Standby screen activation time (page 21) Standby screen brightness (page 21)

Activate Windows

7. ADVANTAGES AND DISADVANTAGES

Advantages:

1. Fast Response
2. Multiple queries can be addressed
3. Reduction in manual labour
4. 24*7 availability.

Disadvantages:

1. Accuracy is compromised at times
2. Limited reply.

8. APPLICATION

This chatbot, Customer Helpdesk, is mainly applicable for any device operation query. Instead of searching for details about the operations of a device from its user manual or waiting for reply from customer care representatives, the chatbot helps to find answers by just raising a query about any specific topic of the device.

9. CONCLUSION

The intelligence of Customer Helpdesk is increased using Smart Document Understanding. To incorporate the various operations in the Chatbot, functions like Watson Assistant, Watson Discovery, Cloud function action and Node-RED Application have been used.

Each of the services helps to build different sections of the chatbot. Integration of different sections is possible by the use of Node-RED. Also, it provides an UI for the customer interaction.

The results obtained show that the Customer Helpdesk is able to communicate with the user like a normal chatbot. In addition to this, Customer Helpdesk is able to give details about the device operations and configurations.

10. FUTURE SCOPE

Device operation related answers are obtained but there is limited field of reply. Higher number of documents (data) feeding can lead to more accuracy.

A personalized or more user friendly Chatbot is seen as the future scope to this system.

11. BIBLIOGRAPHY

Reference links:

1. <https://github.com/IBM/watson-discovery-sdu-with-assistant>
2. <https://www.youtube.com/watch?v=-yniuX-Poyw&feature=youtu.be>
3. <https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/>
4. <https://drive.google.com/file/d/1pKM2lt793hv9RIBAWl4-VGihMXCTF4vI/view>
5. <http://www.iotgyan.com/learning-resource/integration-of-watson-assistant-to-node-red>
6. <https://drive.google.com/file/d/15s07ymOgBMInOf7mabqla5mLiAtIVJ31/view>
7. <https://www.youtube.com/watch?v=Jpr3wVH3FVA&feature=youtu.be>
8. <https://cloud.ibm.com/>

Source Code

Flows.json

```
[
  {
    "id": "25ac7c62.445054",
    "type": "ui_text",
    "z": "1bdf6859.c812c8",
    "group": "75fd3683.4b7c58",
    "order": 3,
    "width": 27,
    "height": 8,
    "name": "",
    "label": "ChatBot",
    "format": "{{msg.payload}}",
    "layout": "col-center",
    "x": 580,
    "y": 400,
    "wires": []
  },
  {
    "id": "75fd3683.4b7c58",
    "type": "ui_group",
    "z": "",
    "name": "",
    "tab": "f04d2422.808bc8",
    "order": 1,
    "disp": true,
    "width": "27",
    "collapse": false
  },
  {
    "id": "f04d2422.808bc8",
    "type": "ui_tab",
    "z": "",
    "name": "Customer Help Desk",
    "icon": "dashboard",
    "disabled": false,
    "hidden": false
  }
]
```

skill-basic.json (Watson Assistant Skill Code)

```
{
  "intents": [
    {
      "intent": "Thanks",
      "examples": [
        {
          "text": "thanks for the help"
        },
        {
          "text": "Was of great help"
        },
        {
          "text": "thanks"
        },
        {
          "text": "was helpful"
        },
        {
          "text": "thank you"
        },
        {
          "text": "Thanks a ton"
        },
        {
          "text": "very helpful"
        }
      ],
      "description": ""
    },
    {
      "intent": "product_information",
      "examples": [
        {
          "text": "wiring configuration?"
        },

```

```
{
  "text": "Accessory confirmation?"
},
{
  "text": "Equipment configuration"
},
{
  "text": "how to reboot"
},
{
  "text": "What is smart recovery?"
},
{
  "text": "Diagnostics"
},
{
  "text": "Wi-fi radio"
},
{
  "text": "How do i turn on the heater?"
},
{
  "text": "How do i set the time?"
},
{
  "text": "How to adjust temperature?"
},
{
  "text": "Overview of the device"
},
{
  "text": "Network"
},
{
  "text": "What mode if on vacation?"
},
{
```

```
    "text": "How to access the settings?"
  },
  {
    "text": "Moved to new house"
  },
  {
    "text": "Changing owner"
  },
  {
    "text": "thresholds"
  },
  {
    "text": "How do I access the settings"
  },
  {
    "text": "Guided Setup process"
  }
]
},
{
  "intent": "Goodbye",
  "examples": [
    {
      "text": "Bye"
    },
    {
      "text": "adieu"
    },
    {
      "text": "Goodbye"
    }
  ],
  "description": ""
},
{
  "intent": "Greeting",
  "examples": [
```

```
{
  "text": "Hello agent"
},
{
  "text": "Hey you"
},
{
  "text": "You there"
},
{
  "text": "Who is this?"
},
{
  "text": "What's up?"
},
{
  "text": "Help"
},
{
  "text": "Help me"
},
{
  "text": "Can you help"
},
{
  "text": "Assist me"
},
{
  "text": "Hello"
},
{
  "text": "Hey"
},
{
  "text": "yo"
},
{
```

```
"text": "Hola"
},
{
  "text": "I had a question"
},
{
  "text": "I am facing trouble"
},
{
  "text": "I am facing difficulty"
},
{
  "text": "can you help me"
},
{
  "text": "need help"
},
{
  "text": "Good morning"
},
{
  "text": "Hi there"
},
{
  "text": "Hey there"
},
{
  "text": "Hey advisor"
},
{
  "text": "Hi"
},
{
  "text": "how are you"
},
{
  "text": "Hello I am looking for some help"
```



```
    }
  ],
  "description": ""
}
],
"entities": [
  {
    "entity": "Help",
    "values": [
      {
        "type": "synonyms",
        "value": "Help",
        "synonyms": [
          "Help",
          "support",
          "need"
        ]
      }
    ]
  },
  {
    "fuzzy_match": true
  }
],
"metadata": {
  "api_version": {
    "major_version": "v2",
    "minor_version": "2018-11-08"
  }
},
"webhooks": [
  {
    "url":
"https://eu-gb.functions.cloud.ibm.com/api/v1/web/rikat137%40gmail.com_dev/default
/disco-action.json",
    "name": "main_webhook",
    "headers": []
  }
],
```

```
"dialog_nodes": [  
  {  
    "type": "response_condition",  
    "parent": "Welcome",  
    "conditions": "anything_else",  
    "dialog_node": "response_2_1590237977828",  
    "previous_sibling": "response_10_1590237975624"  
  },  
  {  
    "type": "response_condition",  
    "output": {  
      "generic": [  
        {  
          "values": [  
            {  
              "text": "Hey there. I am Ecobee3 ChatBot. "  
            }  
          ],  
          "response_type": "text",  
          "selection_policy": "sequential"  
        }  
      ]  
    },  
    "parent": "Welcome",  
    "dialog_node": "response_10_1590237975624"  
  },  
  {  
    "type": "response_condition",  
    "output": {  
      "text": {  
        "values": [  
          "Try rephrasing"  
        ],  
        "selection_policy": "sequential"  
      }  
    },  
    "parent": "node_8_1589837059252",
```

```
"conditions": "anything_else",
"dialog_node": "response_5_1589837095700",
"previous_sibling": "response_5_1589837092740"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "$webhook_result_1"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_8_1589837059252",
  "conditions": "$webhook_result_1",
  "dialog_node": "response_5_1589837092740"
},
{
  "type": "standard",
  "title": "Goodbye",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Hope to see you again."
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  }
}
```

```

    ]
  },
  "conditions": "#Goodbye",
  "dialog_node": "node_4_1590434812528",
  "previous_sibling": "Welcome"
},
{
  "type": "standard",
  "title": "Anything else",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "I didn't understand. You can try rephrasing."
          },
          {
            "text": "Can you reword your statement? I'm not understanding."
          },
          {
            "text": "I didn't get you."
          }
        ]
      },
      "response_type": "text",
      "selection_policy": "sequential"
    ]
  }
],
  "conditions": "anything_else",
  "dialog_node": "Anything else",
  "previous_sibling": "node_4_1590245252192",
  "disambiguation_opt_out": true
},
{
  "type": "standard",
  "title": "Greeting",
  "output": {

```

```
"generic": [
  {
    "values": [
      {
        "text": "How may i help you?"
      },
      {
        "text": "Hey there! What do you want to know?"
      }
    ],
    "response_type": "text",
    "selection_policy": "sequential"
  }
],
"conditions": "#Greeting",
"dialog_node": "node_6_1590249431012",
"previous_sibling": "node_8_1589837059252"
},
{
  "type": "standard",
  "title": "Thanks",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Happy to Help!"
          },
          {
            "text": "Most welcome!"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  }
}
```

```

    },
    "conditions": "#Thanks",
    "dialog_node": "node_4_1590245252192",
    "previous_sibling": "node_6_1590249431012"
  },
  {
    "type": "standard",
    "title": "Product_details",
    "actions": [
      {
        "name": "main_webhook",
        "type": "webhook",
        "parameters": {
          "input": "<?input.text?>"
        },
        "result_variable": "webhook_result_1"
      }
    ],
    "metadata": {
      "_customization": {
        "mcr": true
      }
    }
  },
  "conditions": "#product_information",
  "digress_in": "does_not_return",
  "dialog_node": "node_8_1589837059252",
  "previous_sibling": "node_4_1590434812528"
},
{
  "type": "standard",
  "title": "Welcome",
  "metadata": {
    "callout": {
      "name": "main_webhook",
      "type": "webhook",
      "parameters": {
        "input": "<?input.text?>"
      }
    }
  }
}

```

```

    },
    "result_variable": "webhook_result_1"
  },
  "_customization": {
    "mcr": true
  }
},
"conditions": "welcome",
"dialog_node": "Welcome"
}
],
"counterexamples": [],
"system_settings": {
  "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
  },
  "human_agent_assist": {
    "prompt": "Did you mean:"
  },
  "spelling_auto_correct": true
},
"learning_opt_out": false,
"name": "Basic",
"language": "en",
"description": "Basic functions like greets and replies to appreciation."
}

```

