Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

PROJECT KICKOFF TEMPLATE

Project Scope:

- Create a customer care dialog skill in Watson Assistant
- Use Smart Document Understanding to build a 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

Deliveries:

In this project, there will be another option unlike the typical chatbots. If the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owners manual.

Teams:

Individual

Sumedha Rana

Category: Artificial Intelligent

Internship @ smartinternz.com (2020)

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

1.INTRODUCTION

1.1 Overview:

The main objective of the project is to create an Intelligent Customer Help Desk with smart document understanding by using multiple Watson AI Services (Discovery , Assistant, Cloud function and Node Red). By the top of the project, we'll learn best practices of building interactive information retrieval systems by combining various Watson Services.

Project Requirements: Python, IBMCloud

Software Requirements: Watson Assistant, Watson Discovery, Cloud Function, Node-Red.

Project Team: Sumedha Rana(Individual)

Project Duration: 19days

1.2 Purpose:

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

The purpose of this project is to build a customer helping chatbot such that if the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owners manual. So now, instead of "Would you like to speak to a customer representative?" we can return relevant sections of the owners manual to help solve our customers' problems.

2.LITERATURE SURVEY

2.1 Why the need arises? :

Generally Chatbots means getting input from users and getting only response questions and for some questions the output from bot will be like "try again", "I don't understand", "will you repeat again", and so on. and directs customer to customer agent but a good customer Chatbot should minimize involvement of customer agent to chat with customer to clarify his/her doubts.

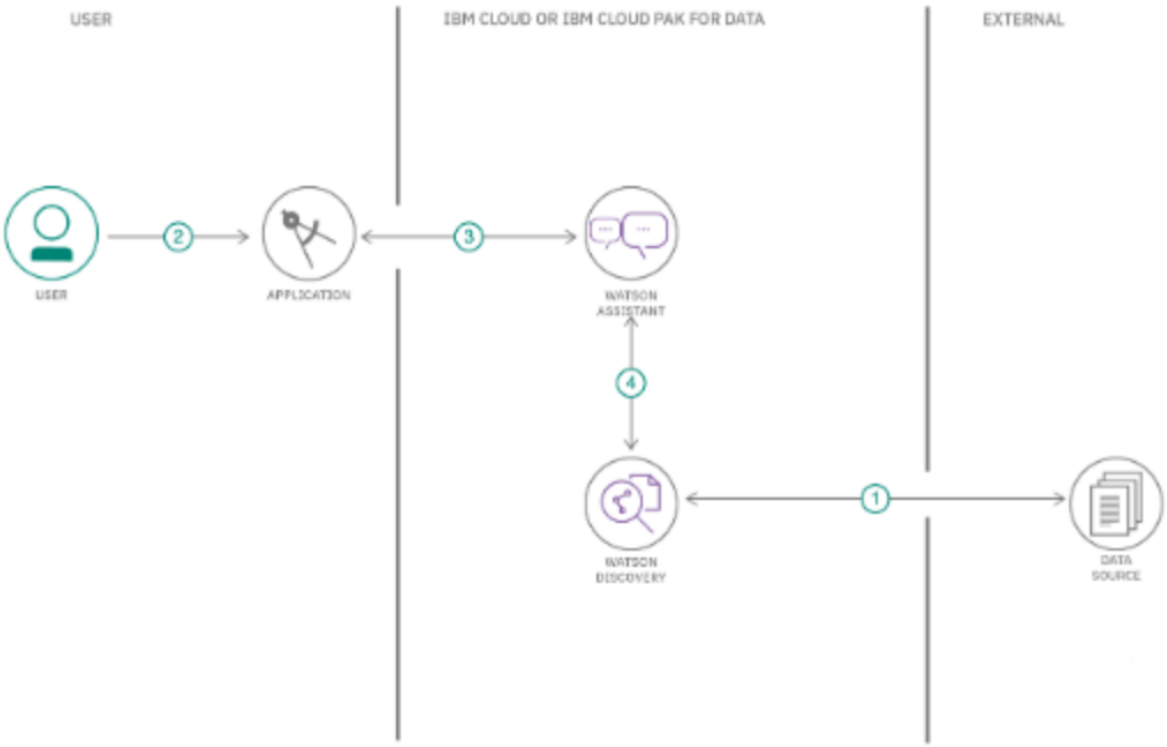
2.2 Proposed solution:

For the above problem we are able to put a virtual agent in chatbot so it can understand the queries that are posted by customers. The virtual agent should train from some insight records based company background so it can answer queries supported the merchandise or associated with company. In other words, some style of manual will be accustomed train the bot using AI. Here i'm using Watson Discovery as a

tool for implementing AI and getting trained by the owners manual of heater.

3.THEORITICAL ANALYSIS

3.1 Block/Flow Diagram



Step-1: the info source from external source is annotated by using Watson Discovery Smart Document Understanding.
Step-2: The user interacts with the back-end server through the appliance program. The front-end app program may be a chatbot that engages the user in a very conversation.
Step-3: Dialog between the user and back-end server is coordinated employing a Watson Assistant dialog skill.
Step-4: If the user asks an issue that falls outside of the scope of the pre-determined question set, then a research query is issued to the Watson Discovery service through a Watson Assistant search skill.

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

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. Create flow and configure node
6. Deploy and run Node Red app.

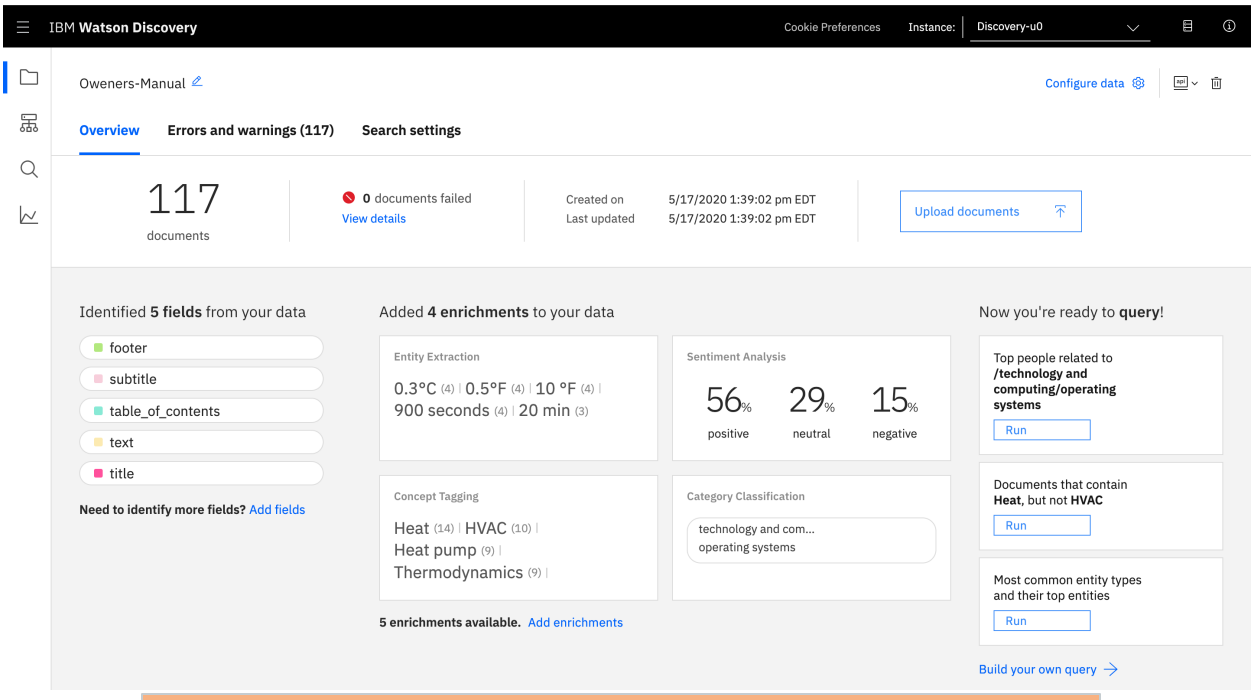
4.EXPERIMENTAL INVESTIGATIONS

4.1.Create IBM Cloud services

Create the following services:

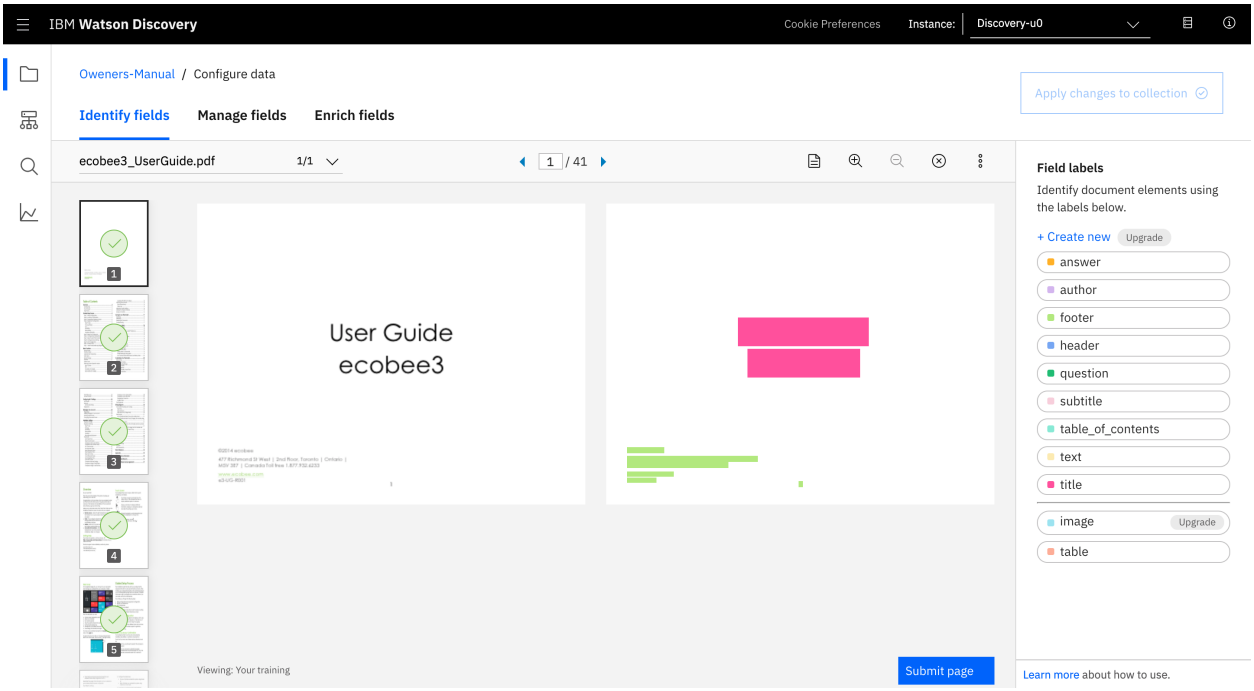
- Watson Discovery
- Watson Assistant
- Cloud Function
- Node Red

Images and Results:

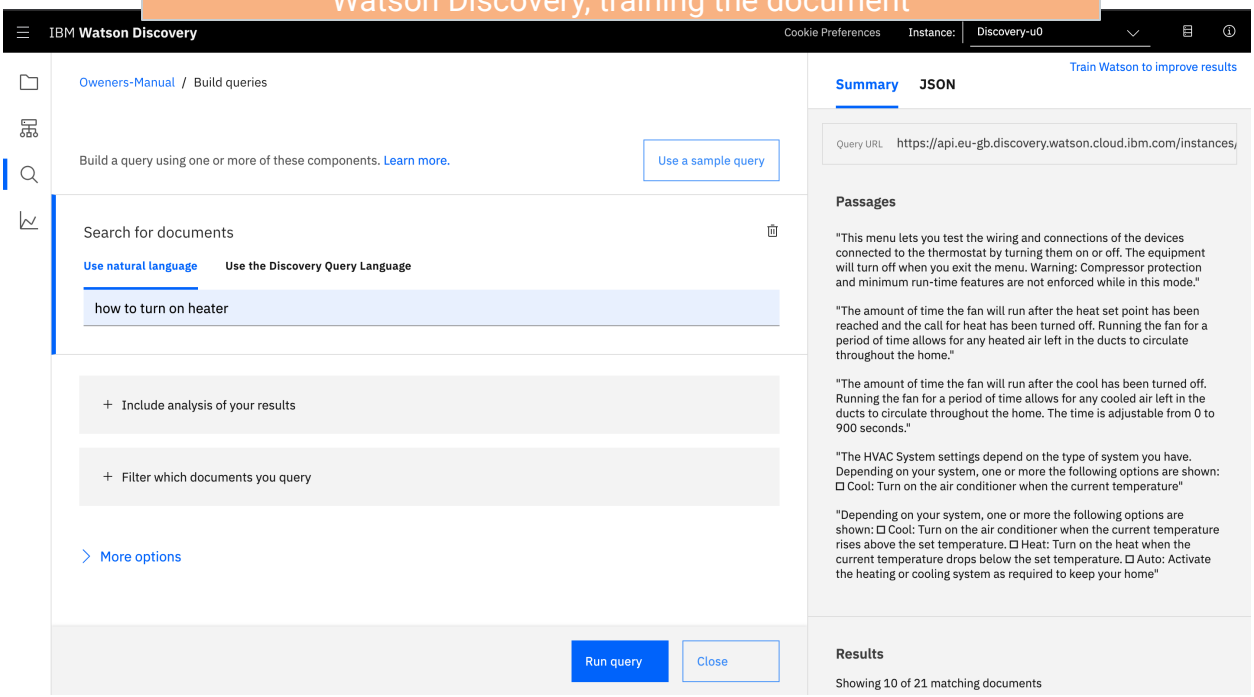


Watson Discovery, after loading the document

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

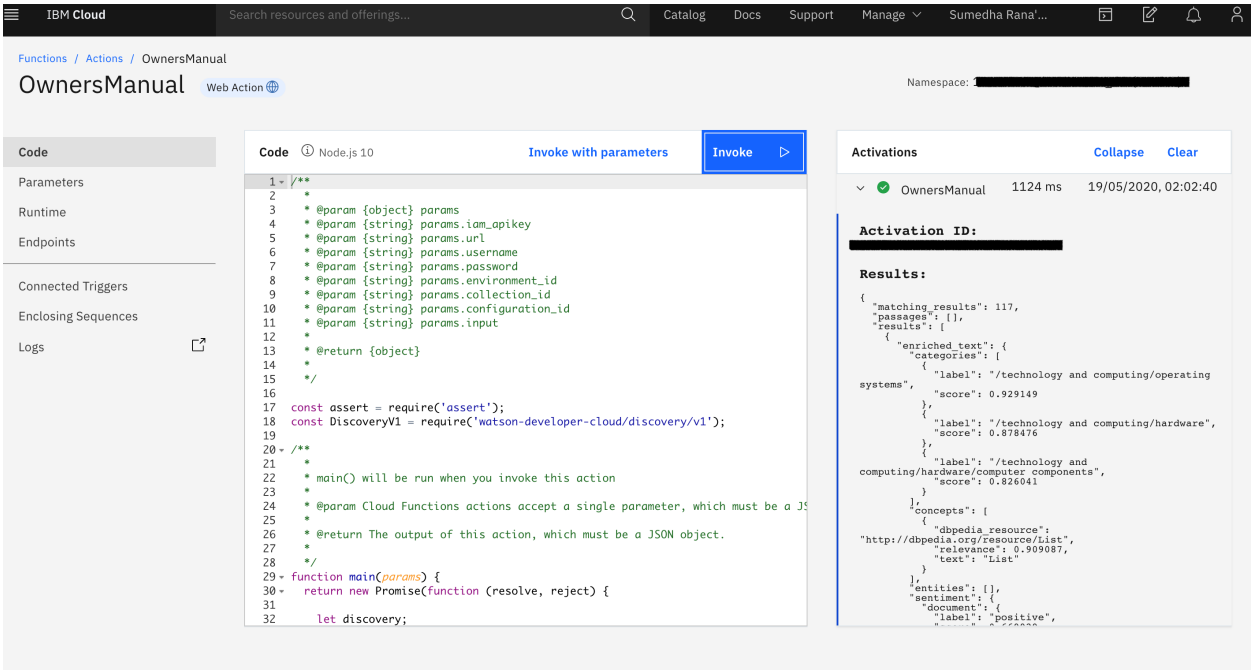


Watson Discovery, training the document



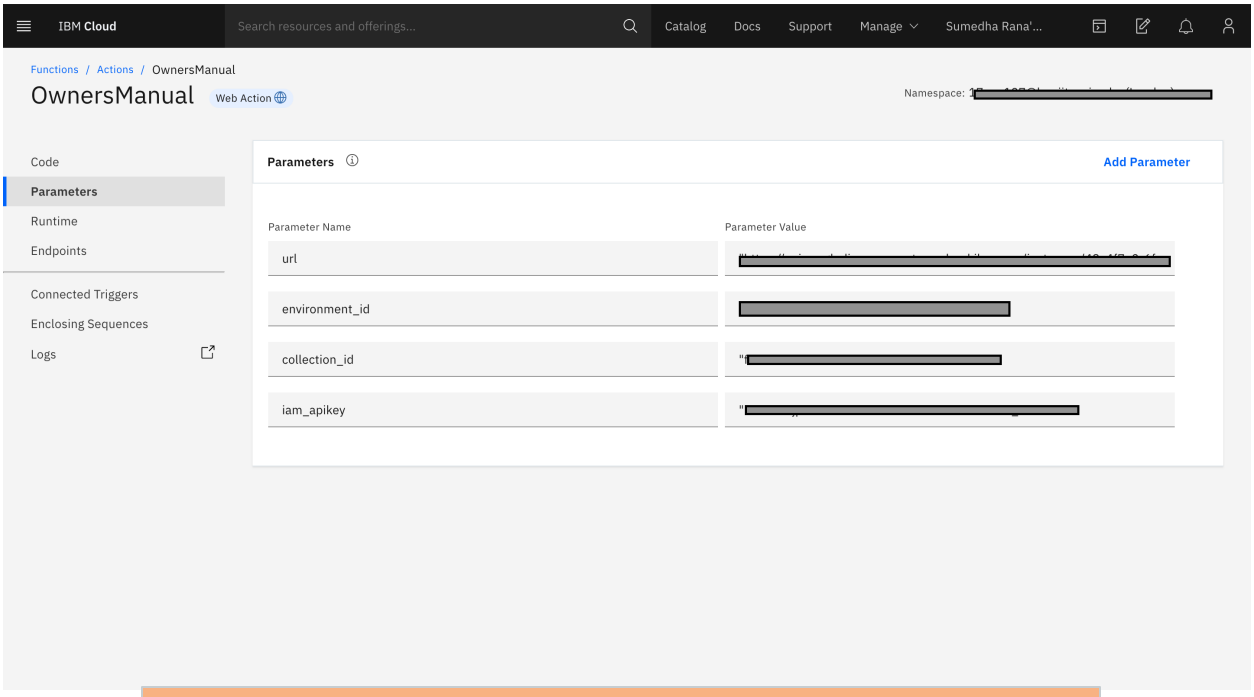
Watson Discovery, Testing the trained document

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

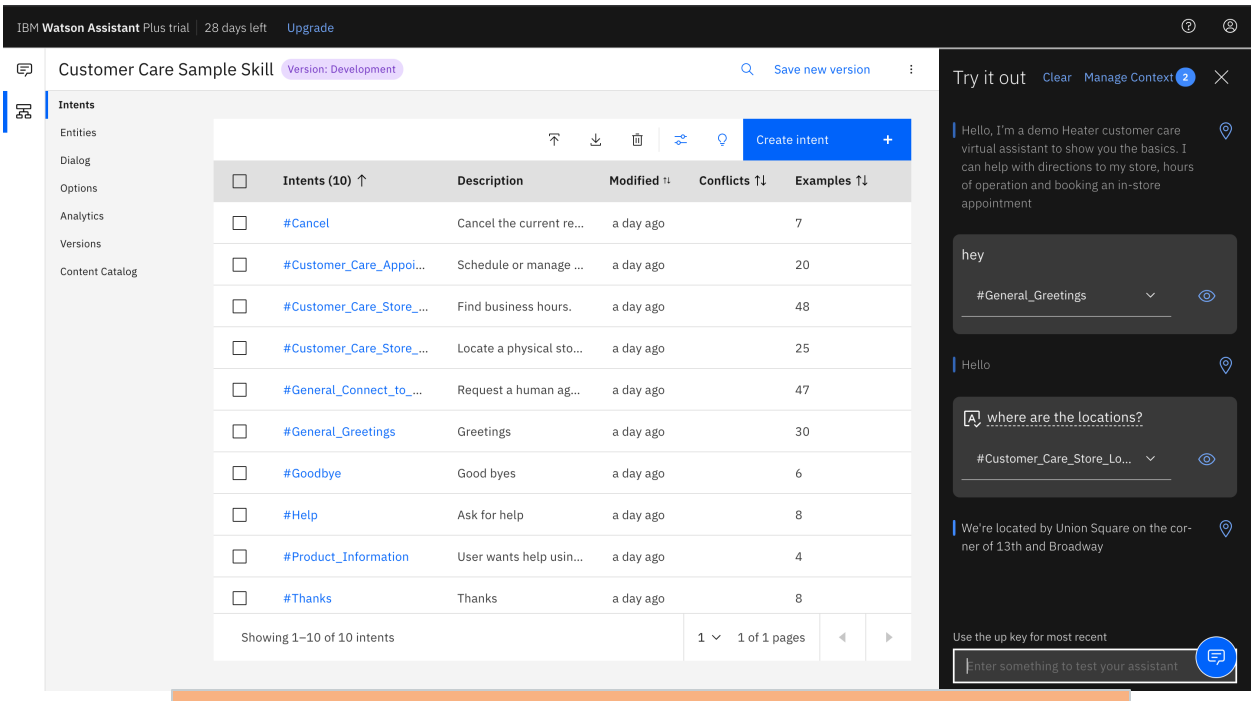


IBM Cloud Functions, code to connect Watson Assistant and Watson Discovery

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

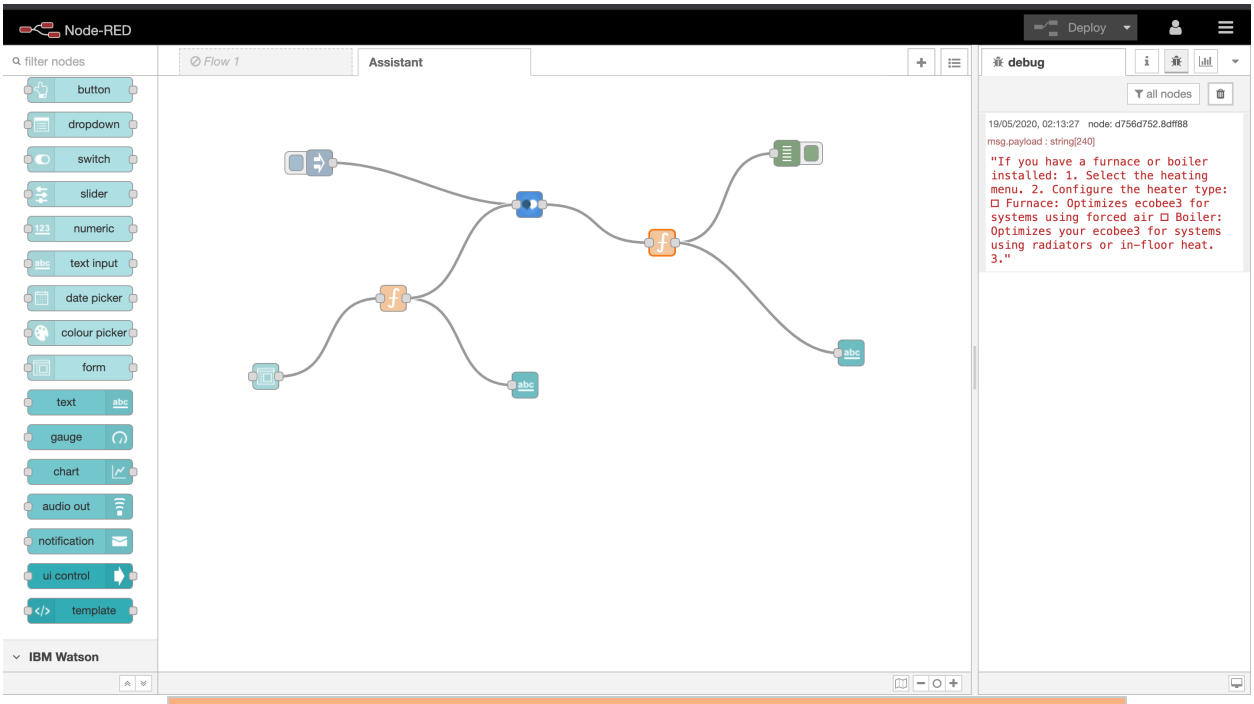


IBM Cloud Functions, Parameters to connect Watson Assistant and Watson Discovery

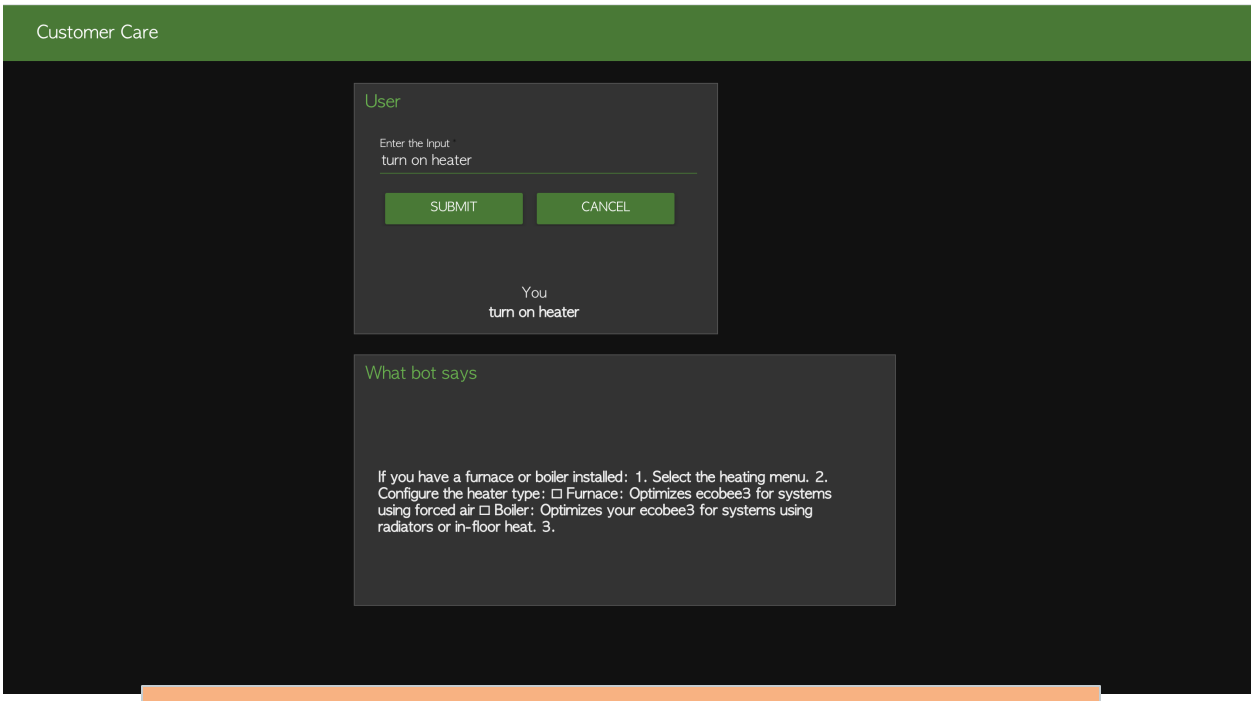


Watson Assistant, checking the output given by the chatbot in try section provided by Watson Assistant.

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)



Node-red, Testing the trained chatbot.



Testing the trained chatbot in user interacttive mode.

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

5.FLOWCHART

1 Create an account on IBM Cloud with Lite plan, then create the following:.

- Watson Discovery
- Watson Assistant
- Node Red

2. Configure Watson Discovery

Load the Document:

Launch the Watson Discovery tool and build a replacement data collection by selecting the Upload your own data option. Give the information collection a novel name. When prompted, select and upload the ecobee3_UserGuide.pdf file located within the data directory of your local repo. The Ecobee could be a popular residential thermostat that contains a wifi interface and multiple configuration options.

Annotate with SDU :

Now let's apply SDU to our document to determine if we are able to generate some better query responses.From the invention collection panel, click the Configure data button (located within the top right corner) to begin the SDU process.

As you go though the annotations one page at a time, Discovery is learning and may start automatically updating the upcoming pages. Once you get to a page that's already correctly annotated, you'll be able to stop, or just click tolerate acknowledge it's correct. The more pages you annotate, the higher the model are going to be trained.

For this specific owner's manual, at a minimum, it's suggested to mark the following: the most page as title

The table of contents (shown within the first few pages) as table_of_contents

All headers and sub-headers (typed in light green text) as a subtitle

All page numbers as footers

All warranty and licensing information (located within the previous couple of pages) as a footer

All other text should be marked as text.

Once you click the Apply changes to collection button , you'll be asked to reload

the document. Choose the identical owner's manual .pdf document as before.

3.Create IBM Cloud Functions action

Now let's create the net action that may make queries against our Discovery collection.

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

step-1: On the Create Action panel, provide a singular Action Name, keep the default package, and choose the Node.js 10 runtime. Click the Create button to make the action.

step-2: Once your action is made, click on the Code tab

within the code editor window, cut and paste within the code from the action.js file found within the actions directory of your local repo. The code is pretty straight-forward - it simply connects to the invention service, makes a question against the gathering, then returns the response.

If you press the Invoke button, it'll fail because of credentials not being defined yet. We'll do that next.

step-3: Select the Parameters tab and add the following keys from Watson Assistant:

url
environment_id collection_id
iam_apikey

step-4 :

In the endpoint section select the Enable as Web action and make note of the url below it.

4.Configure Watson Assistant

Launch the Watson Assistant tool and make a brand new dialog skill. Select the utilization sample skill option as your place to begin.This dialog skill contains all of the nodes needed to own a typical center conversation with a user.

Add new intent

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

The default customer care dialog doesn't have some way to cope with any questions involving outside resources, so we'll must add this.
Create a brand new intent which will detect when the user is asking about operating the Ecobee thermostat.

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

2. Click the Create intent button.

Name the intent #Product_Information, and at a minimum, enter the subsequent example inquiries to be related to it.

3. Create new dialog node

Now we'd like to feature a node to handle our intent. Click on the Dialog tab, then click on the sink menu for the tiny Talk node, and choose 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 like "how do I set the time?", it'll direct the conversation to the present node.
Enable webhook from Assistant
Set up access to our WebHook for the IBM Cloud Functions action you created in Step #4 of IBM cloud Functions.

4.Select the choices tab

Return to the Dialog tab, and click on on the Ask about product node. From the main points panel for the node, click on Customize, and enable Webhooks for this nod and click on Apply.
5. Test in Try it box.

5.Create flow and configure node:

Integration of Watson assistant in Node-RED

- Double-click on the Watson assistant node, provides a name to your node and enter the username, password and workspace id of your Watson assistant service
- After entering all the data click on Done
- Drag inject node on to the result the Input section
- Drag Debug on to the result the output section
- Double-click on the inject node
- Select the payload as a string
  Enter a sample input to be sent to the assistant service and click on on done

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

- Connect the nodes as shown below and click on on Deploy
- Click on the button to send input text to the assistant node
- Observe the output from the assistant service node
- The Bot output is found inside “output.text"
- Drag the function node to parse the JSON data and find the bot response
- Double click on the function node and enter the JSON parsing code and click on on done
- Connect the nodes as shown below and click on on Deploy
- Re-inject the flow and observe the parsed output
- For creating an online application UI we want “dashboard“ nodes which should be installed manually attend navigation pane and click on on manage palette
- Click on install
  Search for “node-red-dashboard” and click on on install and again click on install on the prompt the subsequent message indicates dashboard nodes are installed, close the manage palette
- Search for “Form” node and drag on to the flow
- Doube click on the “form” node to configure
- Click on the edit button to feature the “Group” name and “Tab” name
- Click on the edit button to feature tab name to web application
- Give sample tab name and click on on add do the identical thing for the group
- Give the label as “Enter your input”, Name as “text” and click on on Done
- Drag a function node, double-click on that and enter the input parsing code
- Click on done
- Connect the shape output to the input of the function node and output of the function to input of assistant node
- Search for “text” node from the “dashboard” section
- Drag two “text” nodes on to the flow
- Double click on the primary text node, change the label as “You” and click on on Done
- Double click on the second text node, change the label as “Bot” and click on on Done
- Connect the output of “input parsing” function node to “ You” text node and output of “Parsing” function node to the input of “Bot” text node
- Click on Deploy

7.ADVANTAGES & DISADVANTAGES

Advantages:

- Campanies can deploy chatbots to for easy and general human queries .
- Reduces man power.
- Cost efficient and straightforward to use.
- No must divert calls to buyer for straightforward queries and buyer can look on other works .
- It can group similar reasonably queries of customer and that we can make it learn from them by training it in future.

Disadvantages:

- Some times chatbot can mislead customers by giving wrong answers.
- Giving same account different queries.
- Some times cannot hook up with customer sentiments and intentions.
- If the chatbot doesn't understand the question it's helpless.

8.APPLICATIONS

- It will be deployed in popular social media applications like facebook,slack,telegram.
- Chatbot will be deployed on any website to clarify basic doubts of viewers making the work of consumers agents easy and efficient.

9.CONCLUSION

We are successful in creating an Intelligent help desk smart chatbot using Watson assistant, Watson discovery, Node-RED and cloud-functions.This project was very intresting and I learned a lot during this project.

10.FUTURE SCOPE

We can include Watson studio text to speech and speech to text services to access the chatbot hands free. This is one of the future scope of this project. This makes the bot user interactive and user friendly.

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

APPENDIX

Source Code

All the codes can be found on the Github page :
https://github.com/SmartPracticeschool/llSPS-INT-705-Intelligent-Customer-Help-Desk-with-Smart-Document-Understanding.git

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)

Sumedha Rana
Category: Artificial Intelligent
Internship @ smartinternz.com (2020)