PROJECT REPORT

TITLE: *Intelligent Customer Help Desk With Smart Document Understanding*

Submitted as an Internship Project Report

To: SmartInternz (https://smartinternz.com)

By: Dasa Gangadhar Seshadri (seshasatyasaieesha@gmail.com)

Category: Career Basic

Role: Artificial Intelligence Developer

1. Introduction

1.1 Project Description:

The typical customer care chatbot can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. When a question falls outside of the scope of the pre-determined question set, the option is typically to tell the customer the question isn't valid or offer to speak to a real person. In this project, there will be another option. If the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owners manual. So now, instead of "Would you like to speak to a customer representative?" we can return relevant sections of the owners 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 owners manual is important and what is not. This will improve the answers returned from the queries.

Project Requirements: Python, IBM Cloud, IBM Watson

Project Domain: AI,ML,IBM Watson

Softwares Needed: Watson Assistant, Watson Discovery

1.2 Scope of Work:

- Create a customer care dialog skill in Watson Assistant
- Use Smart Document Understanding to build an enhanced Watson Discovery collection
- Create an IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery

- Build a web application with integration to all these services & deploy the same on IBM Cloud Platform

2. Literature Survey

2.1 Existing Problem

The typical customer care chatbot can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. When a question falls outside of the scope of the pre determined question set, the option is typically to tell the customer the question isn't valid or offer to speak to a real person i.e. customer care representative. So the problem that our project is addressing to is to utilise the Smart Document Understanding feature of IBM Watson Discovery and thus return the relevant sections from the manual for the customer's reference.

2.2 Proposed Solution

Steps:

1. Create IBM Cloudservices
2. Configure WatsonDiscovery
3. Create IBM Cloud Functionsaction.
4. Create a Node red flow to connect all the services together.
5. Configure WatsonAssistant.
6. Create flow and configurenode
7. Deploy and run Node Redapp.

We shall see each of these steps below:

1. Create IBM Cloudservices:

The following services are to be created:

Watson Discovery

Watson Assistant

Node Red

2. Configure Watson Discovery:

Import the document. Launch Watson Discovery tool and create a new data collection by selecting "Upload your data" option. Give the collection a unique name. When prompted, select the "ecobee3_userguide.pdf" manual file thats been provided.

Annotating with Smart Document Understanding:

Now, by applying Discovery's Smart Document Understanding to our document, we can generate better query responses. To do this, click the "Configure data" button from the Discovery collection panel which is located in the top right corner to start the Smart Document Understanding process. Here, our goal is to annotate all pages of the document so that Discovery can learn which text is important and which can be ignored.

3. Create IBM Cloud functions Action:

Next step now is to create web actions which will query the Discovery collection.
To do this, Go to IBM Functions section and from the main panel, click on Actions tab.
Then click on "Create" and from there select "Create Action" option. Provide a unique name there and click on "Create" button to create our action.
Once the action gets created, go to Code tab and copy paste the code from "cloud functions actions" file. This code connects to the Discovery Service and makes a query to the collection and returns the response.
Next, select "Parameters" tab and add the following keys: url, environment_id, collection_id, iam_apikey. The values for these are to be given from the Discovery service created in previous step. Next, click on "Invoke" and this will show the results returned from dicovery service.
Now, go to Endpoints panel and check the "Enable as web action" box which will then generate a pulic endpoint URL. Copy this URL and store it somewhere since this url is needed in assistant configuration in next step.

4. Configure Waston Assistant:

Launch Watson Assistant tooling and create a new dialog skill. Select "Use sample skill" option which contains all the nodes needed to have a typical customer care conversation with user.

Add new intent:
We have to add a new intent which can handle user queries related to product since this is not provided by the default dialog. Select Intents tab from Customer Care Sample Skill panel and click "create intent" button. Name it as "#product_info" and give it a few example questions that are commonly asked.

Create a new Dialog node:
To add a node which handles our intent, click on "Dialog" tab and select the "Add node" from the drop down. Give it a name "Ask about product" and assign this to our new intent. This is mainly to facilitate the Bot to redirect to this node whenever it recognises a question related to the product.

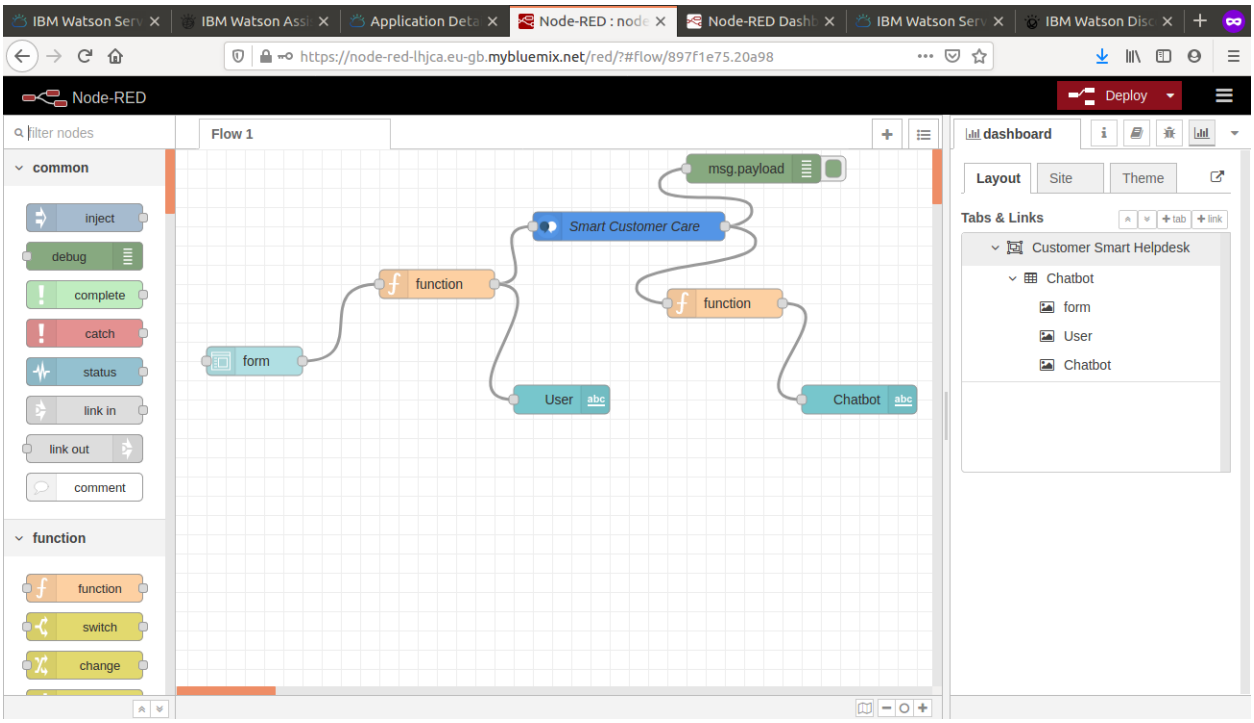
Enable Webhook from Assistant:
Here, we need to provide access to the webhook for IBM Cloud function which we created in previous step. This can be done by entering public URL endpoint to the action and enable webhook for our "Ask about product" node in Dialog panel and click on Apply.

Testing:
Now its time to test our Bot. Go to Dialog panel in Assistant tool and click on "Try it" button on the top right side. Give some user input and see the response from the bot. Whenever you give a query related to the product, say for example, "How to turn on my heater", this will then trigger our "Ask about product" node and is indicated by the "#product_info" response.

5. Create the flow and configure the node:
Open your IBM Nodered app from the catalog and initially go to Manage palette and install Dashboard from there. This will install most of the commonly used nodes. Clear the screen by deleting the default nodes shown and then configure your app by creating the flow in below manner:
Inject
Assistant
Debug
Function
Ui_Form
UI_text

Link the nodes according to the flow and configure them necessarily.

The screenshots below depict the above process:



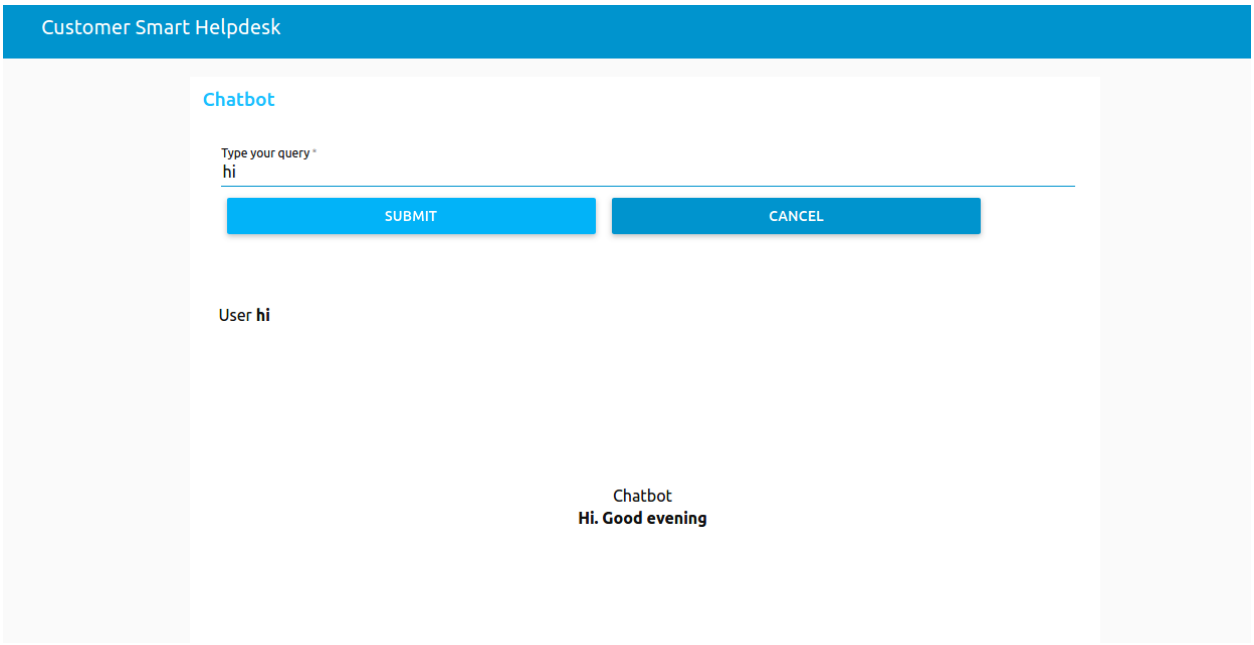
6. Deploy and run the Node Red App:

Deploy the configured node red app by clicking on "Deploy" button in red color.

Then copy the url that you have noted previously and paste in a new tab upto "/ui/" as shown below to get the dashboard from which you can interact with the Bot as an user.

https://node-red-lhjca.eu-gb.mybluemix.net/ui/

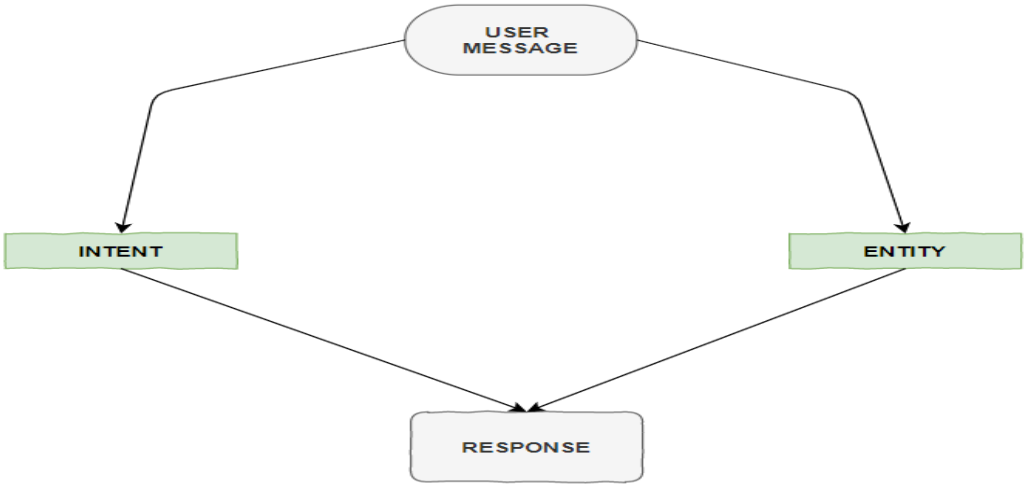
This can be seen from screenshot below which shows the dashboard that opens up when we paste the url in new tab and then we can interact with the Bot through any message and here I have started with Greeting saying Hi and the bot has responded back correctly giving a greeting reply back.



3. Theoretical Analysis

3.1 Block Diagram

The below diagram depicts how the interaction is happening among the various components of our system.



3.2 Hardware and Software Specifications:

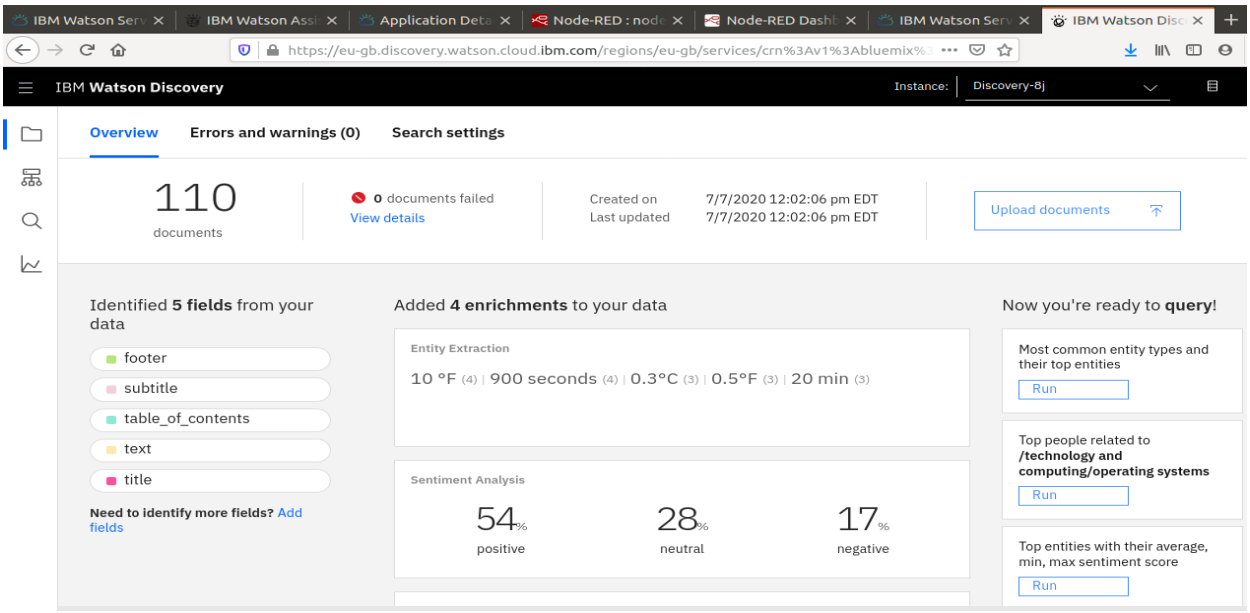
Hardware: PC or Laptop with good configuration, Internet Connectivity with good speed

Software: IBM Cloud services, IBM Watson, Python, AI, ML

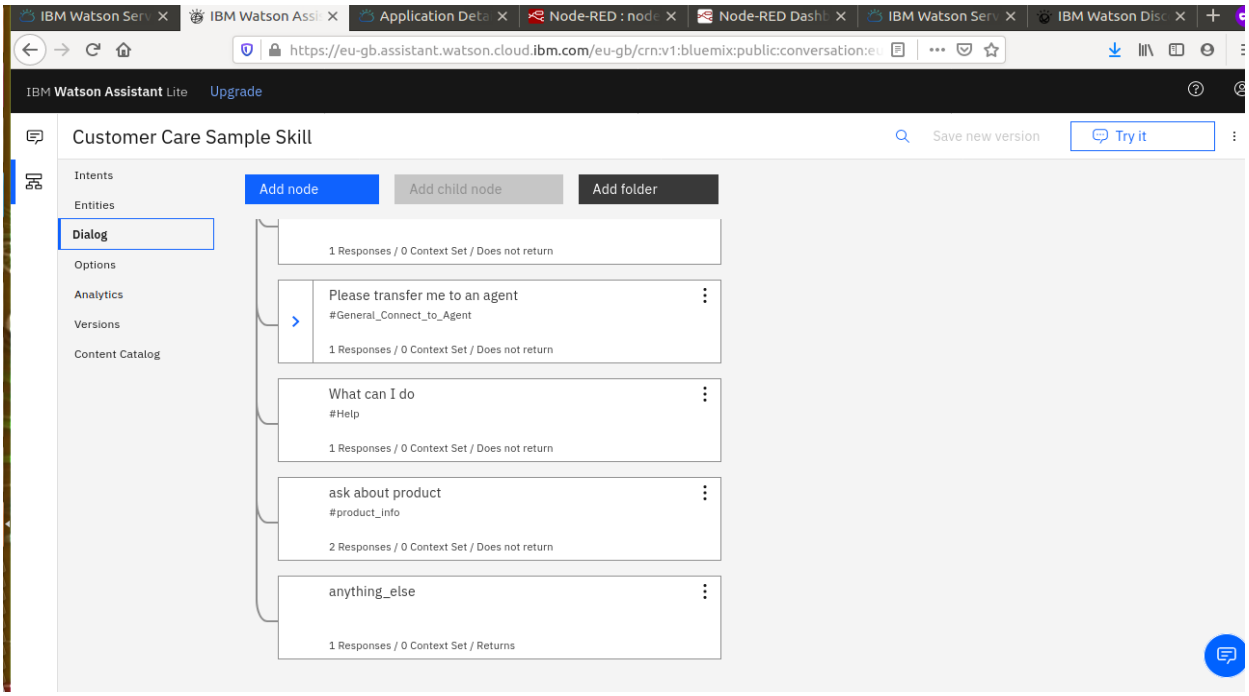
4. Experimental Investigations.

The screenshots inserted below provide a depiction to our work on this project with various configurations and tests done.

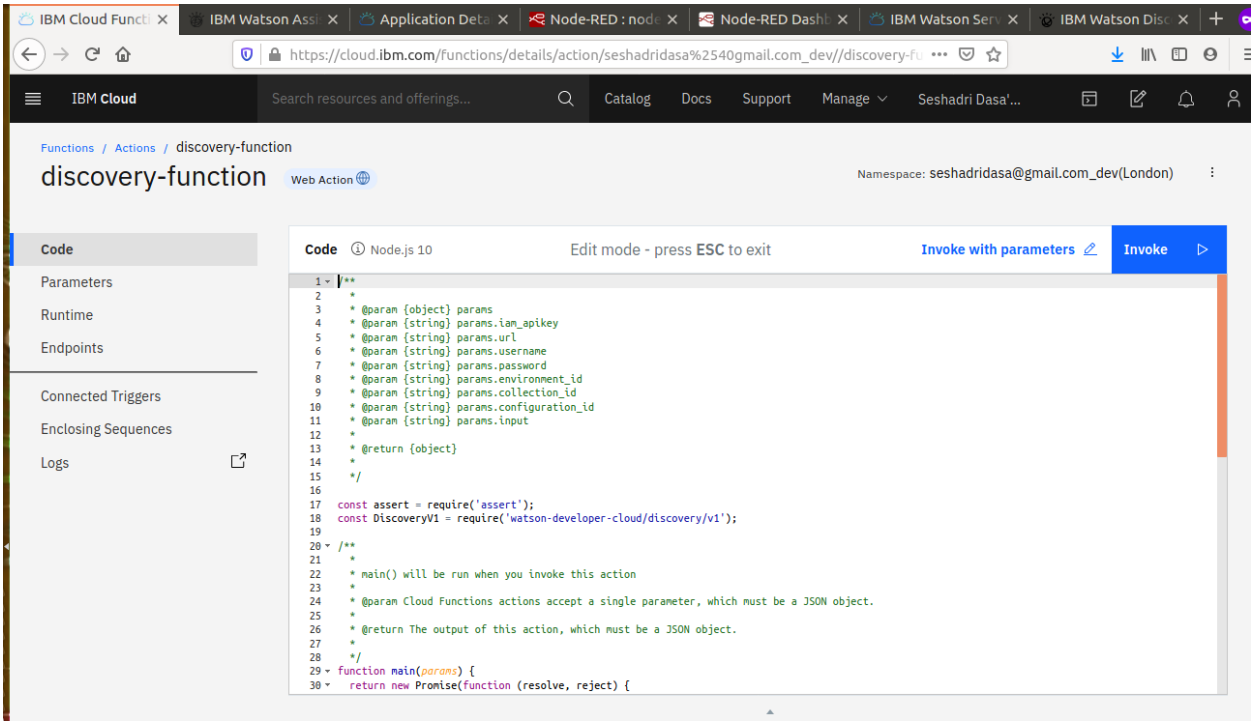
Watson Discovery:



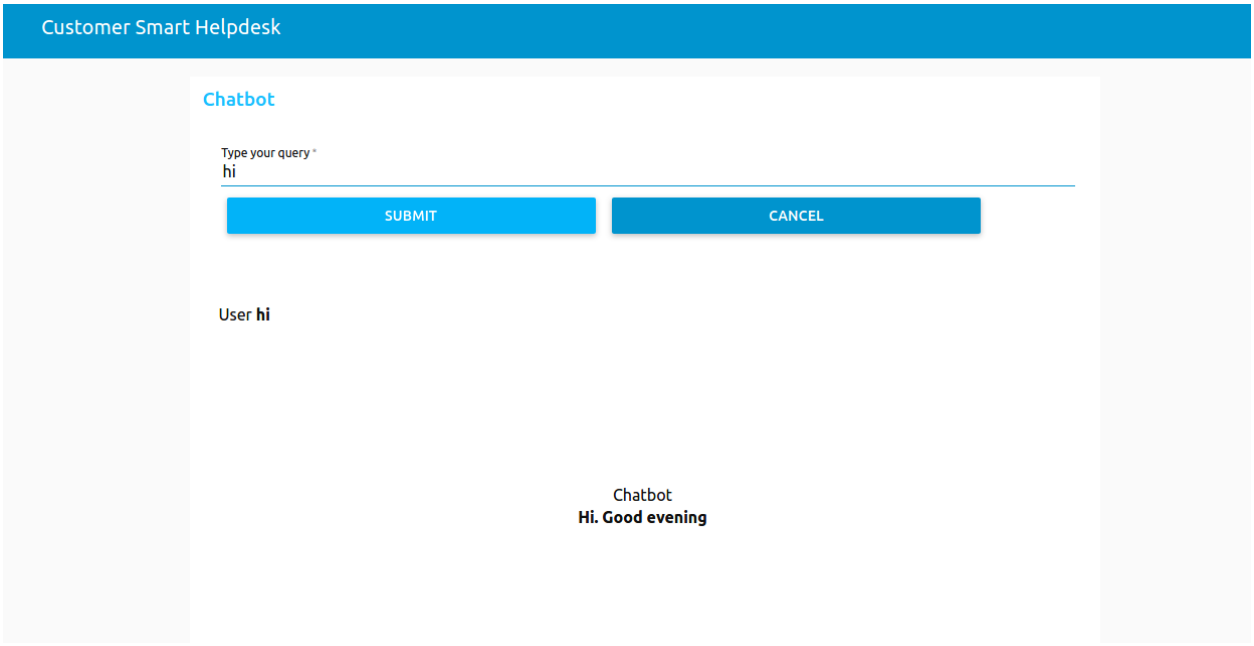
Watson Assistant:



Cloud Function:

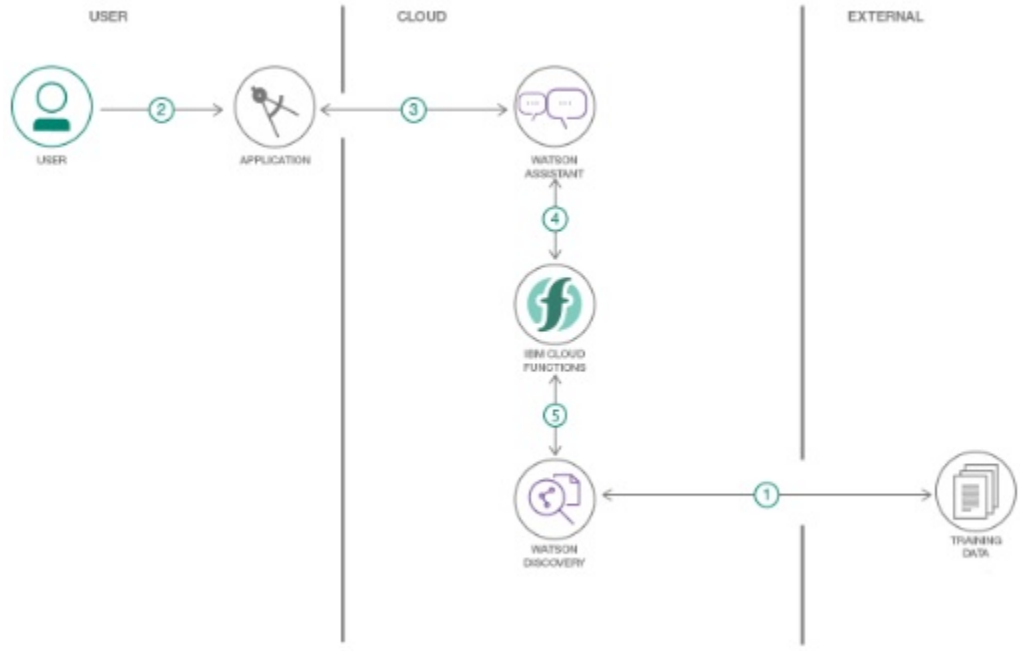


Dashboard:



5. Flowchart:

The figure below shows the flow of tasks for our project:



We can brief it as below:

1. Annotate the document using Watson Discovery Smart Document Understanding.

2. User interaction with backend is facilitated through a frontend dashboard app UI which is a bot which engages user in a conversation and answers his/her queries.

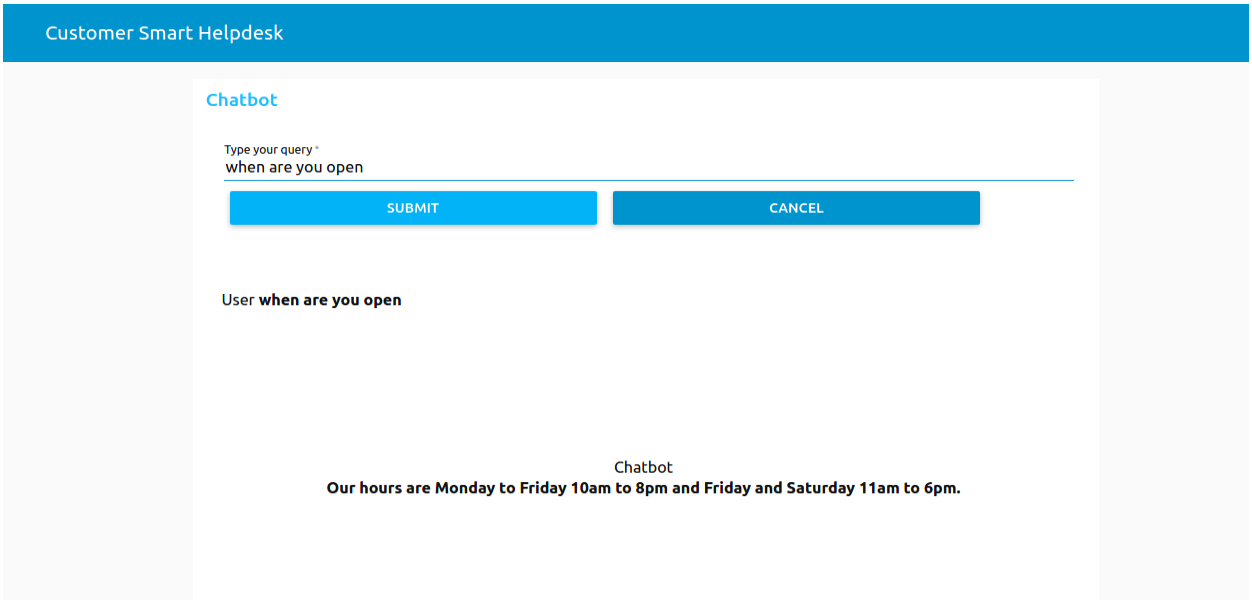
3. Dialog flow between user and backend is coordinated using Watson Assistant dialog skill.

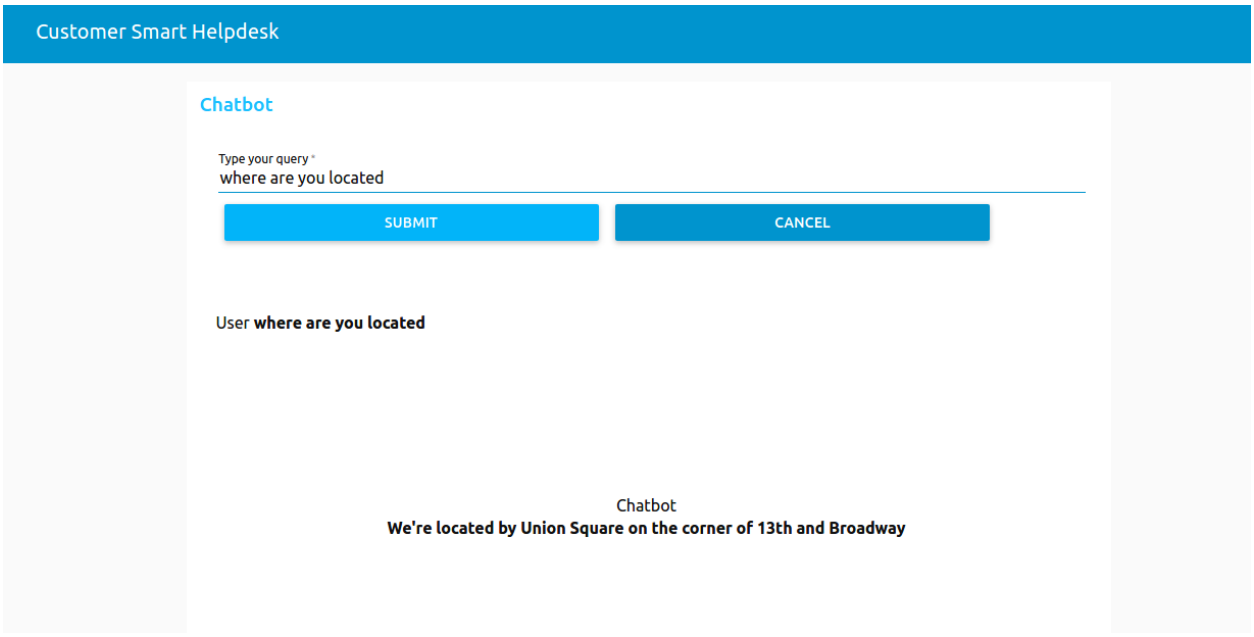
4. If the user's query is related to product info, a search query is then passed to the predefined IBM Cloud functions Action.

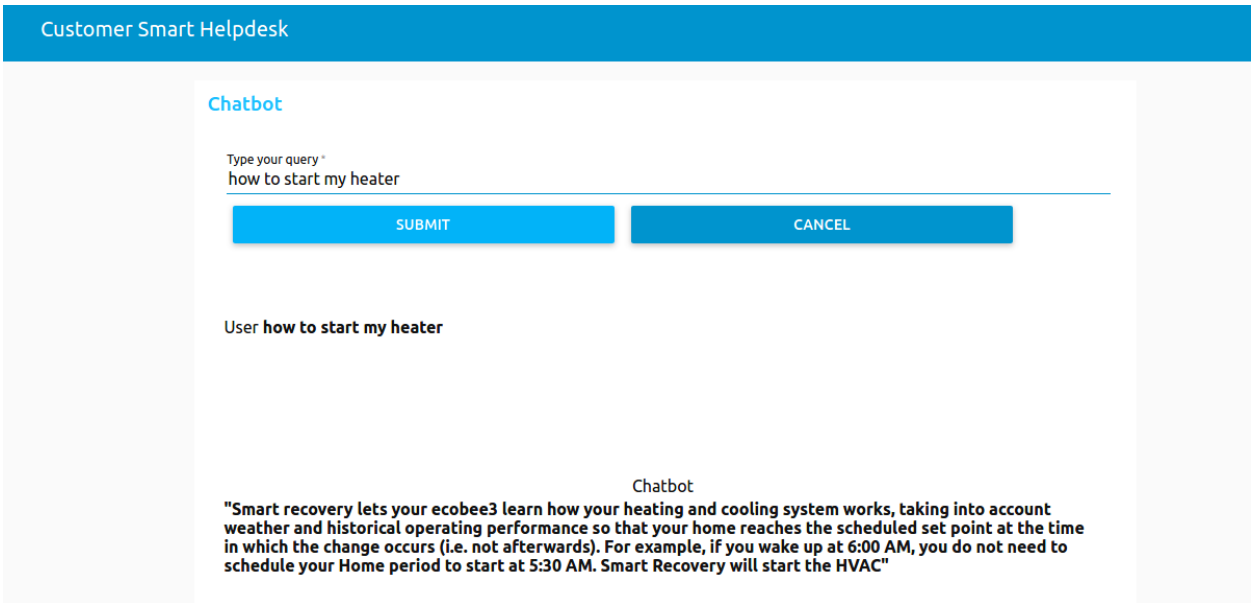
5. Cloud Functions Action will query the Watson Discovery Service and return the results.

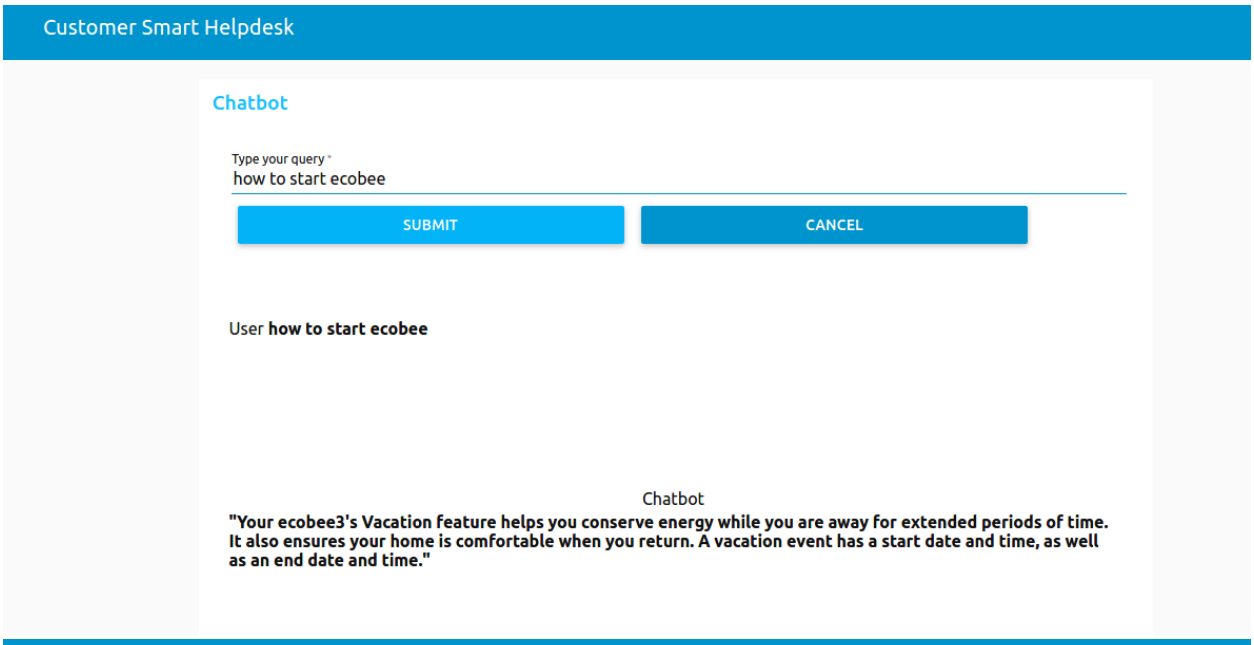
6. Results:

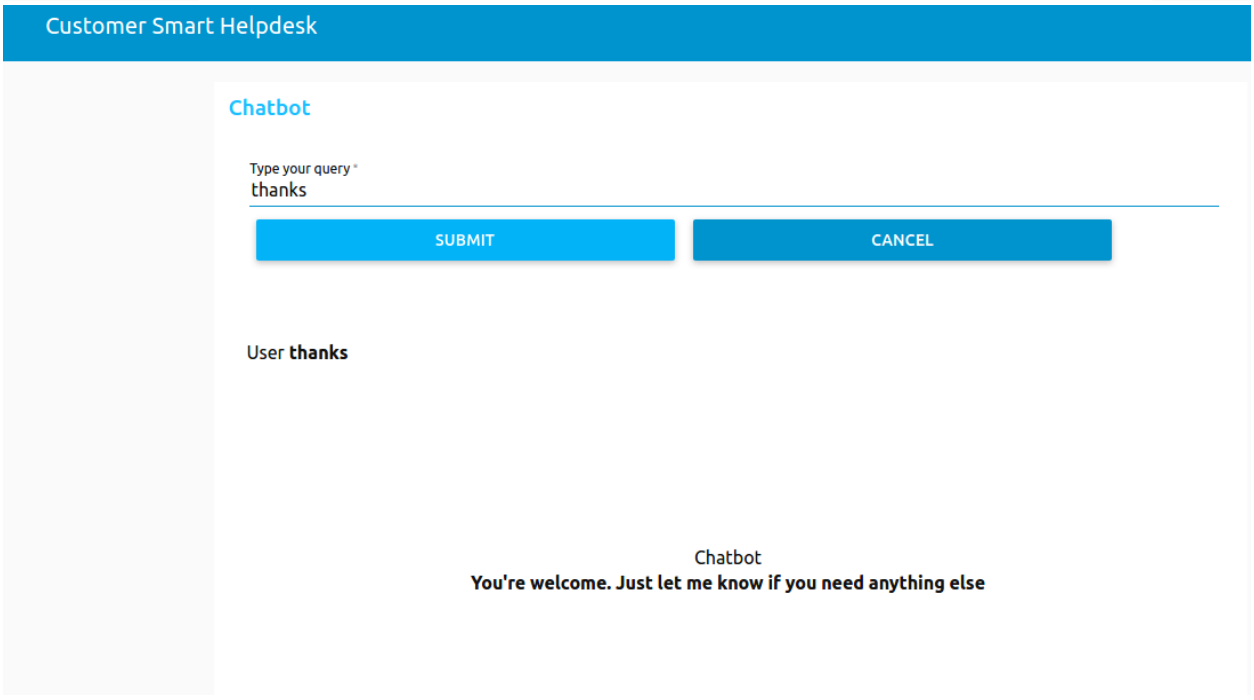
The screenshots below show the various results for several queries given to the chatbot that was successfully built.

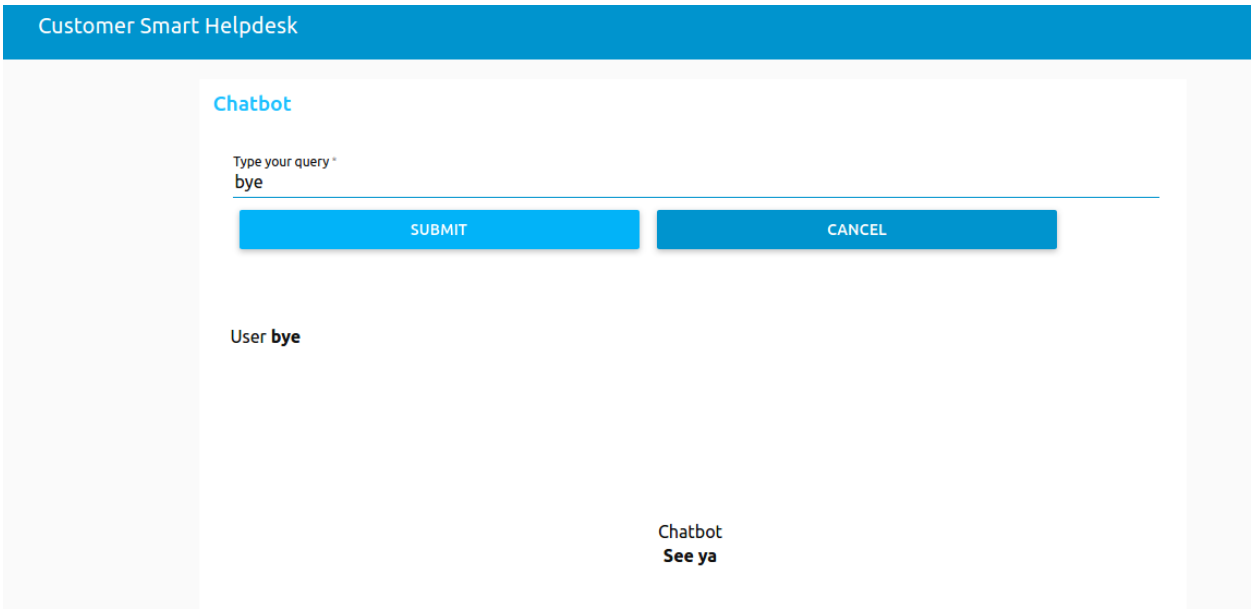



Customer Smart Helpdesk
Chatbot
Type your query *
where are you located
SUBMIT
CANCEL
User where are you located
Chatbot
We're located by Union Square on the corner of 13th and Broadway


Customer Smart Helpdesk
Chatbot
Type your query *
how to start my heater
SUBMIT
CANCEL
User how to start my heater
Chatbot
"Smart recovery lets your ecobee3 learn how your heating and cooling system works, taking into account weather and historical operating performance so that your home reaches the scheduled set point at the time in which the change occurs (i.e. not afterwards). For example, if you wake up at 6:00 AM, you do not need to schedule your Home period to start at 5:30 AM. Smart Recovery will start the HVAC"


Customer Smart Helpdesk
Chatbot
Type your query *
how to start ecobee
SUBMIT
CANCEL
User how to start ecobee
Chatbot
"Your ecobee3's Vacation feature helps you conserve energy while you are away for extended periods of time. It also ensures your home is comfortable when you return. A vacation event has a start date and time, as well as an end date and time."


Customer Smart Helpdesk
Chatbot
Type your query *
thanks
SUBMIT
CANCEL
User thanks
Chatbot
You're welcome. Just let me know if you need anything else



7. Advantages and Disadvantages:

Advantages:

The main advantage of building and using chatbots is that they eliminate the need for labor during online interaction with customers and this helps a lot as many organisations receive several queries at once and thus chatbot saves costs to the company and thus by aligning chatbot with their objectives, organisations can use it as an effective means to enhance their customer conversation.

Another advantage is that chatbots can be available 24/7 to their customers unlike human representatives who have fixed functioning timings and thus customer need not wait for someone to approach them and also Chatbot never gets tired unlike humans.

Next, AI based chatbots are special due to their learning from interactions capabilities and thus get updated regularly.

Human representatives can attend only a limited number of customers at a time whereas this is not the case with Bots as they can manage and serve multiple clients at a time.

Disadvantages:

Like any other system, chatbots too are bound to have some disadvantages which include:

Complex Interface

Sophisticated Environment

Bad memory etc.

8. Applications

A typical chatbot can have applications such as:

User Assistance

Content Delivery

Companionship

Customer Support

E-Commerce

Healthcare etc.

9. Conclusion

The work carried out in this project is very helpful as it helps to build a chatbot which can answer user queries about a product in a smart manner unlike typical systems. They provide enhanced User Guidance. 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 approvals. Use of conversational AI chatbots itself means better engagement and relentless need for customer satisfaction in the near future.

10. Future Work/Scope

The future scope of this project lies in enhancing the functionality of the chatbot by providing it with features such as:

Nature Language Processing and Understanding

Integration with IOT Devices

Voice Control etc.

11. Bibliography

1. https://www.ibm.com/cloud/architecture/tutorials/cognitive_discovery
2. https://cloud.ibm.com/docs/assistant?topic=assistant-getting-started
3. https://developer.ibm.com/recipes/tutorials/how-to-create-a-watson-chatbot-on-nodered/
4. http://www.iotgyan.com/learning-resource/integration-of-watson-assistant-to-node-red
5. https://github.com/IBM/watson-discovery-sdu-with-assistant
6. https://www.youtube.com/watch?v=Jpr3wVH3FVA

12. Acknowledgments

I would hereby like to Acknowledge and sincerely thank SmartInternz and SmartBridge team for providing me with this opportunity of Internship through the amazing platform to work in the domain of my interest and thus helped me in enhancing my skills by learning IBM Cloud Services for AI.