Project Report

Name: Nihar P. Asare(niharasare@gmail.com)

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

Category: *Artificial Intelligence*

Internship: SmartBridge 2020

1. Introduction

1.1 Project Overview:

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 predetermined question set, the option is typically to tell the customer the question isn't valid or offer to speak to a realperson.

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 preloaded with the device's owners manual. So now, instead of passing the questions to customer representives, we can return relevant sections of the owners manual to help solve our customers'problems.

For the above mentioned solution, 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 accuracy returned from the queries.

Project Requirements: Python, IBM Cloud, IBMWatson

Functional Requirements: IBMcloud

Technical Requirements: AI, ML, WATSON AI, PYTHON

Software Requirements: Watson assistant, Watsondiscovery.

1.2 Project Purpose:

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

2.2 Proposed Solution

Steps:

a. Create IBM Cloudservices

b. Configure WatsonDiscovery

c. Create IBM Cloud Functionsaction.

d. Create a Node red flow to connect all the services together.

e. Configure Watson Assistant

f. Create flow and configure node

g. Deploy and run Node Red app.

1. Create IBM Cloudservices

Create the following services:

a. WatsonDiscovery
b. WatsonAssistant
c. NodeRed

2.Configure WatsonDiscovery

- *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 redmi_note_5_pro_manual.pdf file located in the data directory of your local repo.

- *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 Functionsaction

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 packageand select the Node.js 10runtime.

Click the Create button to create the action.

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

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 Discoveryservice:

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 parametervalues.

4. Configure WatsonAssistant

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.

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

Select the Options tab:

Enter the public URL endpoint for your action. Return to the Dialog tab, and click on the Askabout 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

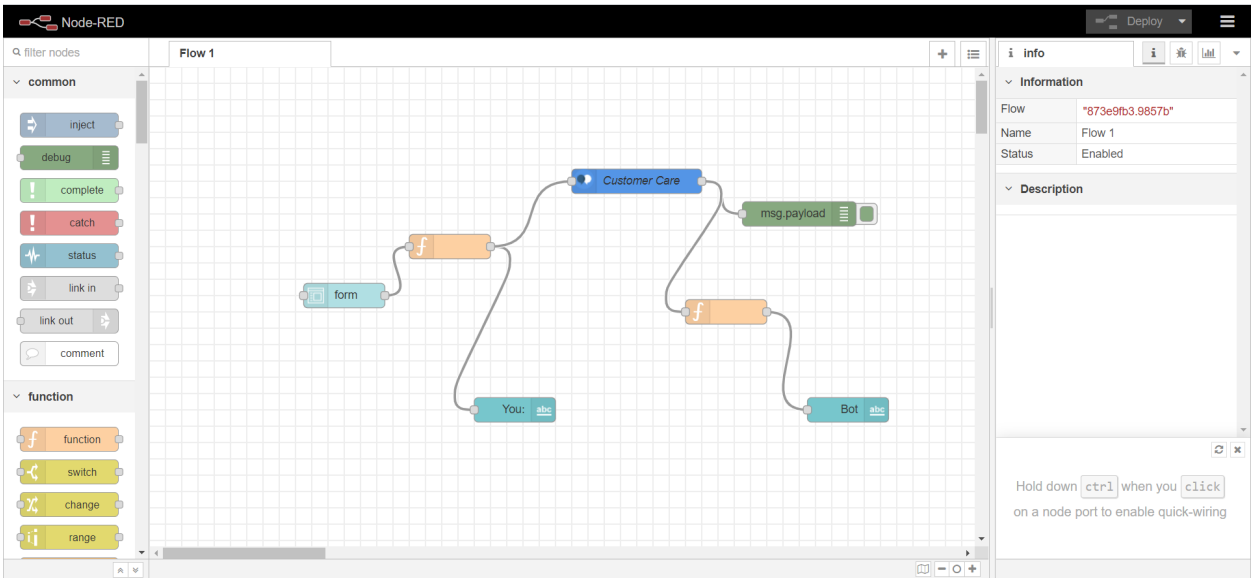
From the Dialog panel, click the Try it button located at the top right side of the panel.Enter some user input: Note that the input "how do I turn on the heater?" has triggered our Ask about product dialog node, which is indicated by the #Product_Information response.And because we specified that $webhook_result_1.passages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

5. Create flow and configurenode:

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

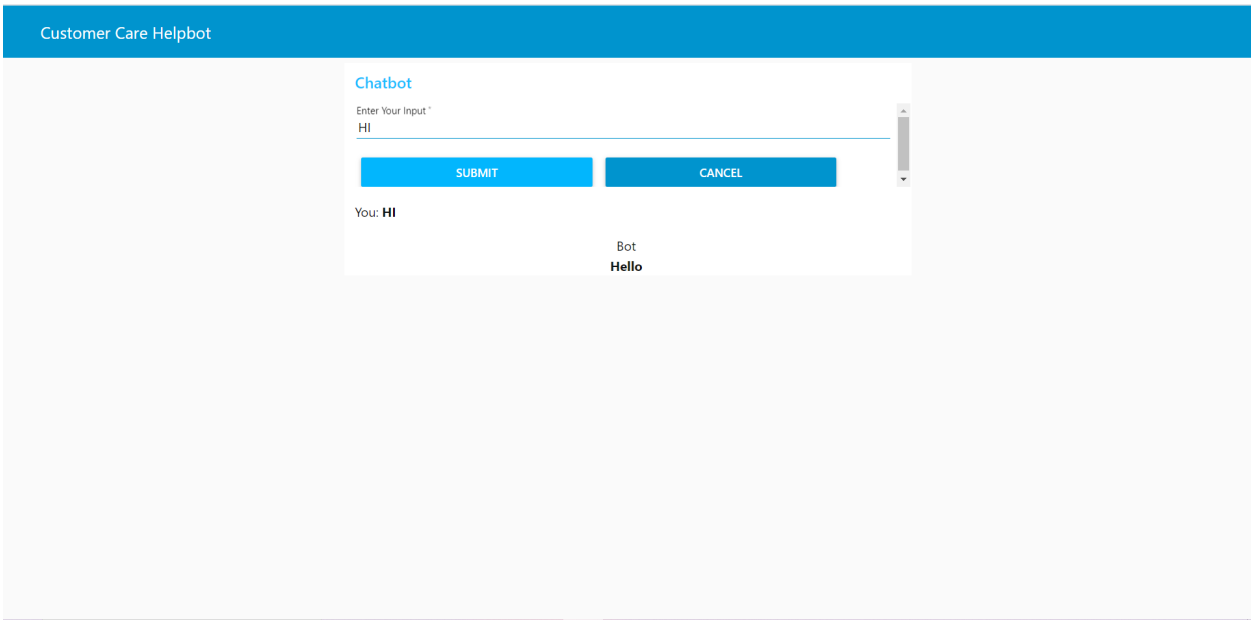


6. Deploy and run Node Red app.

Deploy the Node Red flow.

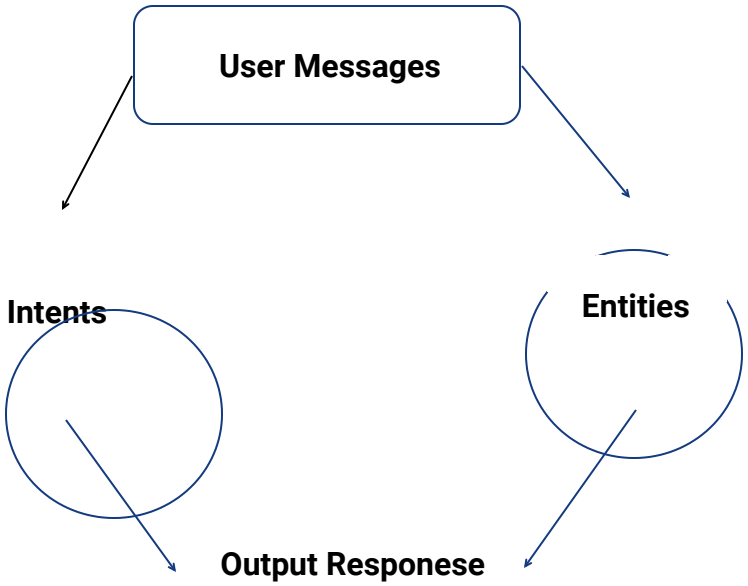
Then copy the link url upto .net/ and paste at a new tab by ui at the end of url:

https://node-red-ijakh.mybluemix.net/ui



3.Theoretical Analysis

3.1 Block Diagram



3.2 Hardware and Software Designing

Project Requirements: Python, IBM Cloud, IBMWatson

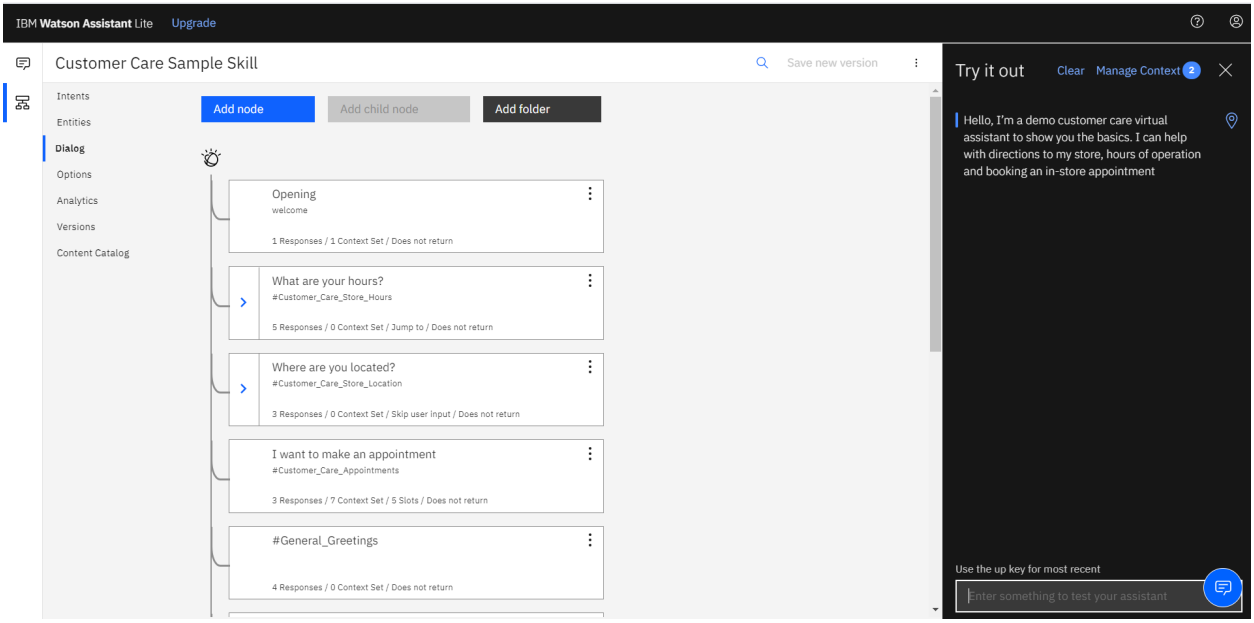
Functional Requirements: IBMcloud

Technical Requirements: AI, ML, WATSONAI, PYTHON

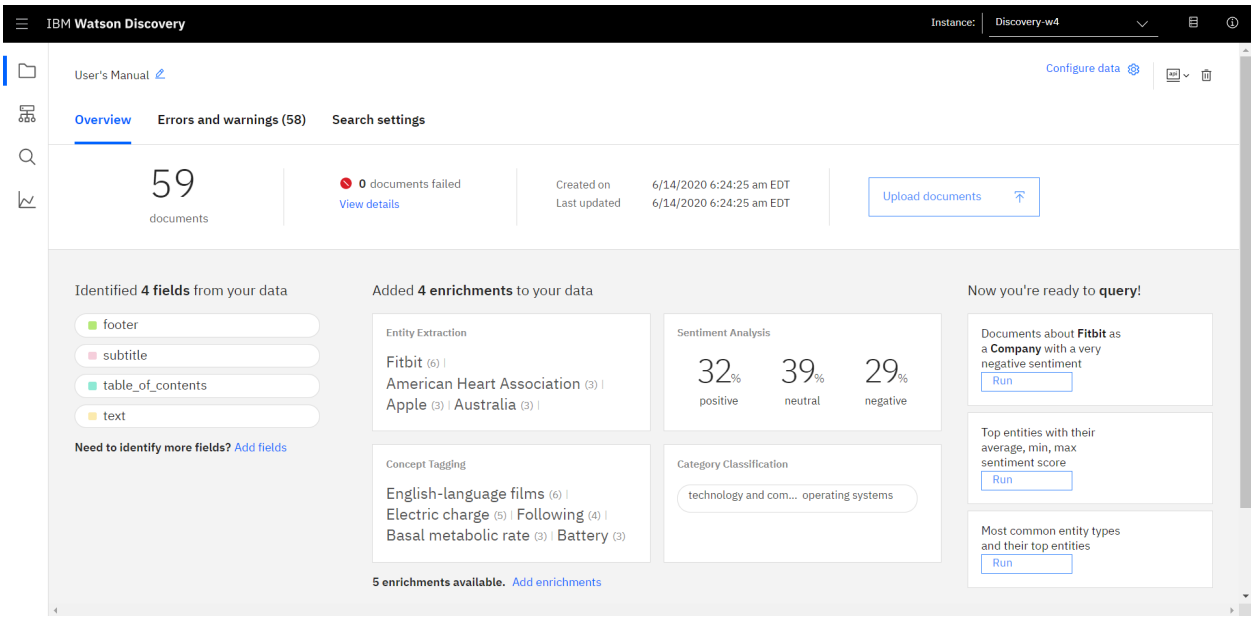
Software Requirements: Watson assistant, Watson discovery.

4.Experimental Investigations:

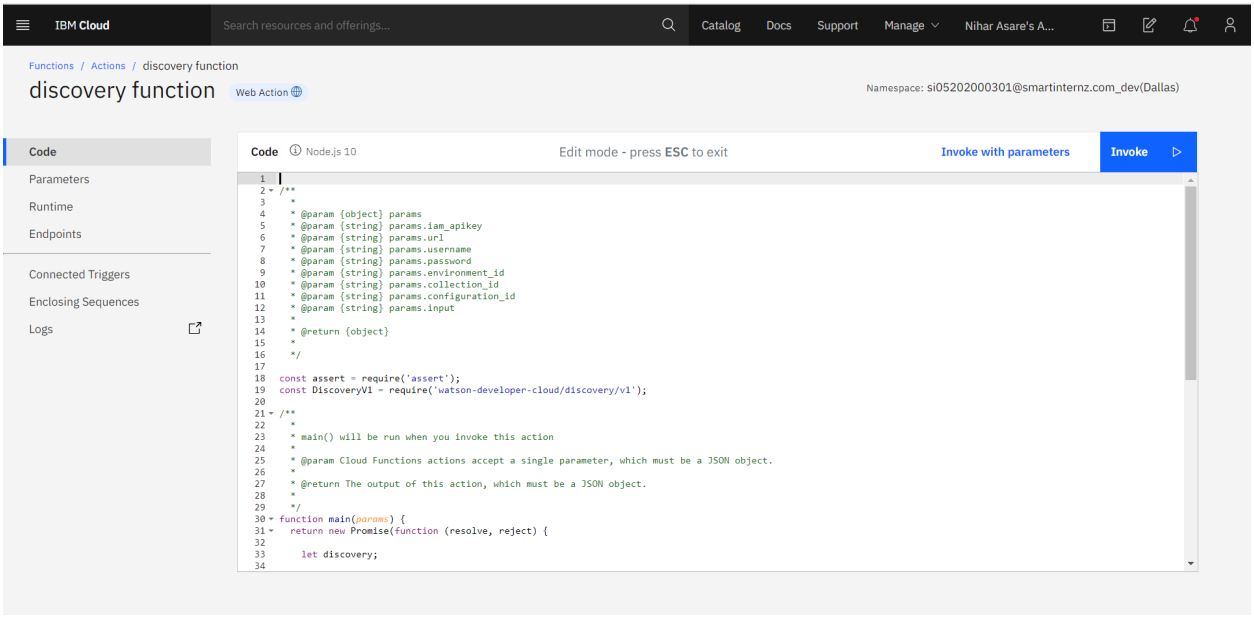
Watson Assistant:



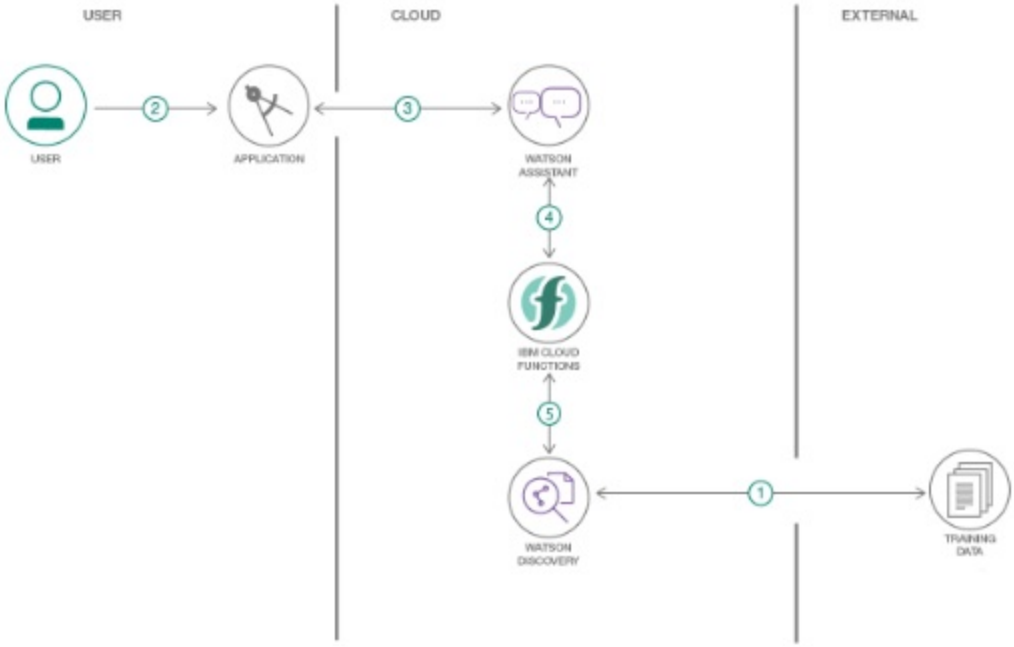
Watson Discovery:



Cloud Function:



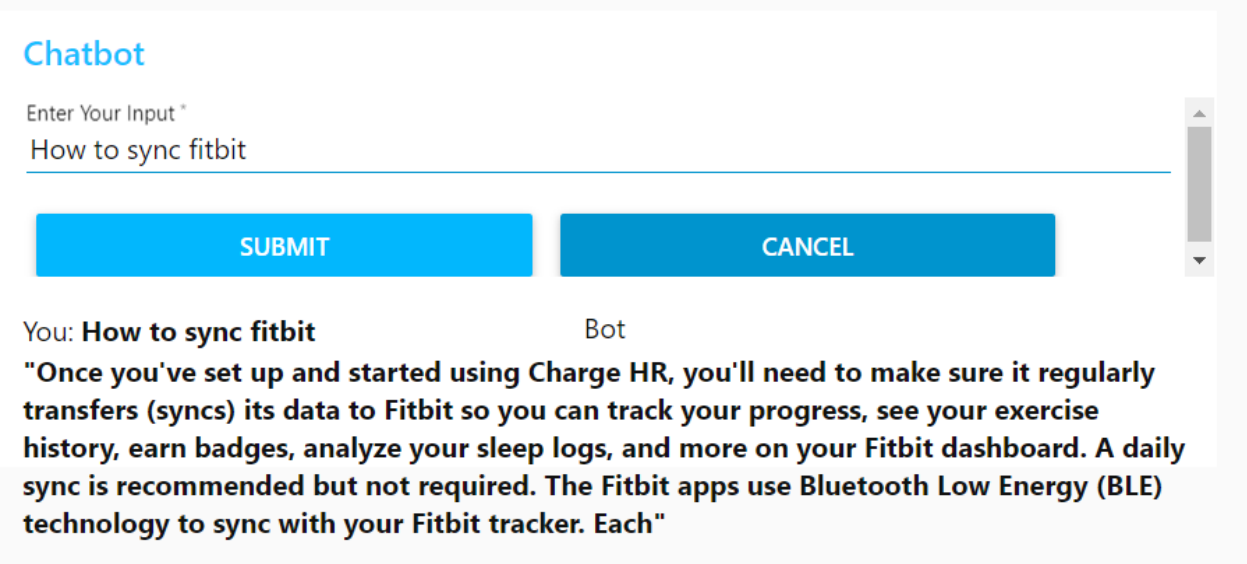
5. Flowchart



1. The document is annotated using Watson DiscoverySDU
2. The user interacts with the backend server via the app UI. The frontend app UI is a chatbot that engages the user in aconversation.
3. Dialog between the user and backend server is coordinated using a Watson Assistant dialogskill.
4. If the user asks a product operation question, a search query is passed to a predefined IBM Cloud Functionsaction.
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.



7.Advantages and Disadvantages

Advantages:

A. Reduced costs: Chatbots eliminate the need for labor duringonline 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 » duringnon-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).

Disadvantages:

A. Complexity of 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.

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.
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.

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: We can make them smarter using NLP. ML is getting advanced day-by-day, so it will affect the performance of chatbot for future purpose as well.

2] Integration with IoT Devices: The smart home devices, 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.

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

1. https://www.ibm.com/cloud/get-started
2. https://github.com/watson-developer-cloud/node-red-labs
3. https://www.w3schools.com/howto/howto_make_a_website.asp
4. https://www.youtube.com/embed/_UgRPaxipgI
5. https://www.youtube.com/embed/Jpr3wVH3FVA
6. https://www.youtube.com/embed/G3bqRndQtQg