Project Report

**Name:**Digisha Kaneria

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

**Category**: *Artificial Intelligence*

*Internship at smartinternz.com@2020*

1. **Introduction**
   1. **Project Overview:**

Conversations play an important role in everyday life. Conversation can be general which are used to generate fun or they can be used to solve queries. For any conversation in general at least two people are required. Conversation can also occur between a computer and a human. Such conversations can be achieved through chatbots.

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 ownersmanual. So now, instead of “Would you like to speak to a customer representative?” we can return relevant sections of the owners manualto 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.

What is a Chatbot?

Chatbot is made up of two words “Chat” representing conversation and “Bot” representing a robot. Hence a chatbot is enabling conversations with a robot.

■ Generally speaking a bot is any software that performs an automated task, however, weare interested in the class of bots that live online in chat platforms or on social mediacalled chatbots.

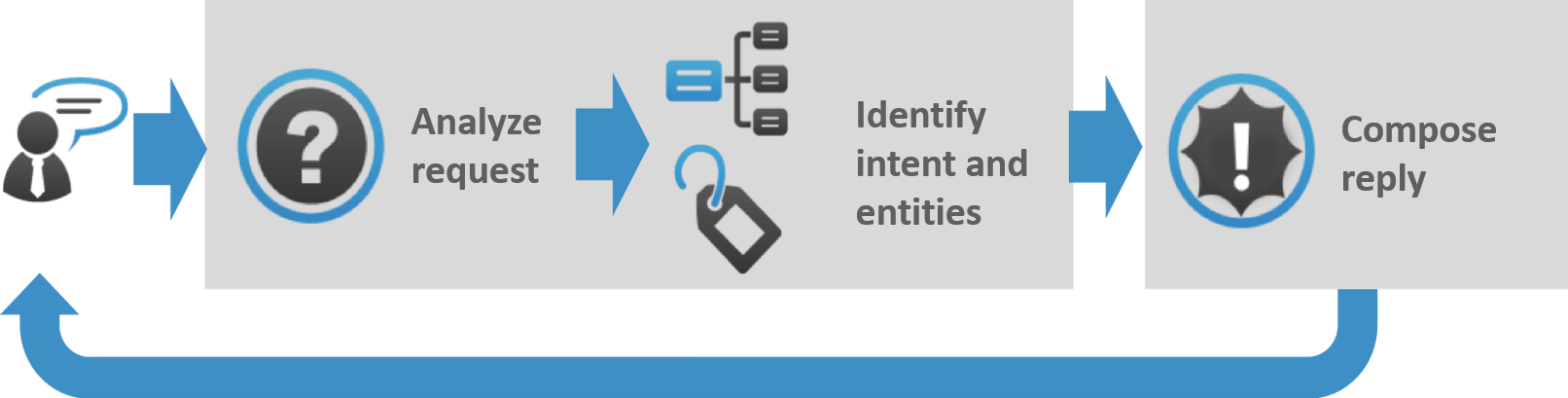
■ The most intuitive definition is that a bot is software that can have a conversation with ahuman. For example, a user could ask the bot a question or give it an instruction and thebot could respond or perform an action as appropriate.

How a chatbot works ?

A chatbot performs two main tasks :

1. User request analysis

2. Returning response



The ability to identify the user’s intent and extract data and relevant entities contained in the user’s request is the first condition and the most relevant step at the core of a chatbot: If you are not able to correctly understand the user’s request, you won’t be able to provide the correct answer. Returning the response: once the user’s intent has been identified, the chatbot must provide the most appropriate response for the user’s request. The answer may be: • A generic and predefined text • A text retrieved from a knowledge base that contains different answers • A contextualized piece of information based on data the user has provided • Data stored in enterprise systems • The result of an action that the chatbot performed by interacting with one or more backend application • A disambiguating question that helps the chatbot to correctly understand the user’s request

**PURPOSE**

Chatbot applications streamline interactions between people and services, enhancing customerexperience. At the same time, they offer companies new opportunities to improve the customerengagement process and operational efficiency by reducing the typical cost of customer service.To be successful, a chatbot solution should be able to effectively perform both of these tasks.

**Project Requirements**: Python, IBM Cloud, IBMWatson

**Functional Requirements**: IBMcloud

**Technical Requirements**: AI, ML, WATSONAI, PYTHON

**Software Requirements**: Watson assistant, Watsondiscovery.

**1.2 Scope of Work**

* Create a customer care dialog skill in WatsonAssistant
* 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 WatsonDiscovery.
* Build a web application with integration to all these services &deploy the same on IBM CloudPlatform

1. **Literature Survey**
   1. **Proposed 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 preddetermined question set, the option is typically to tell the customer the question isn’t valid or offer to speak to a real person.

**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. This human intervention should be reduced by making the assisstant more capable of guiding the user by refering the appropriate section in the manual which is stored in the backend.

* 1. **Proposed Solution**

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.

# 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.

# Create IBM Cloudservices

Create the following services:

* 1. WatsonDiscovery
  2. WatsonAssistant
  3. NodeRed

# 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. Fromthe Discovery collection panel, click the Configure data button (located in the top right corner) to start the SDUprocess. 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.

# 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 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’sdirectory 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.

# 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

Step#4. 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.

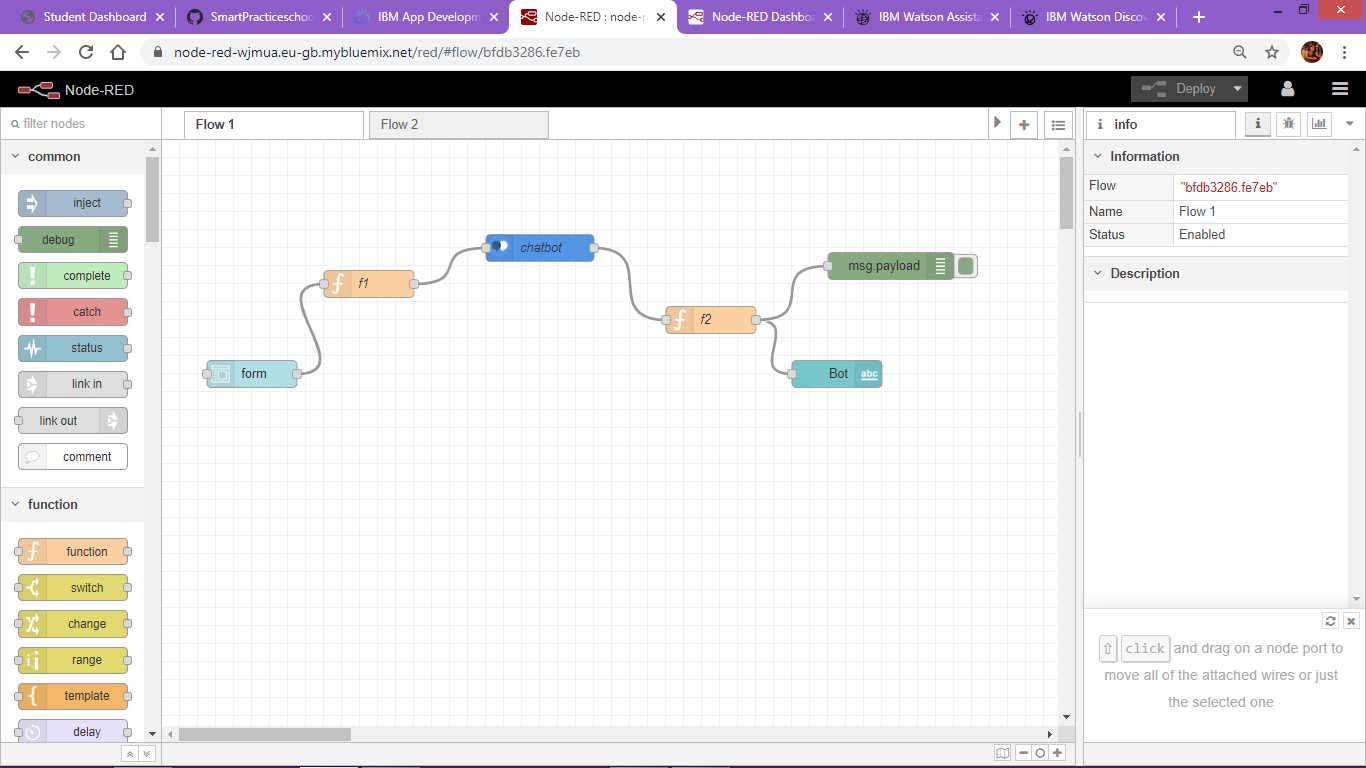
Test in Assistant Tooling

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.

# Create flow and configure node:

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

* Inject
* Assistant
* Debug
* Function
* Ui\_Form
* Ui\_Text

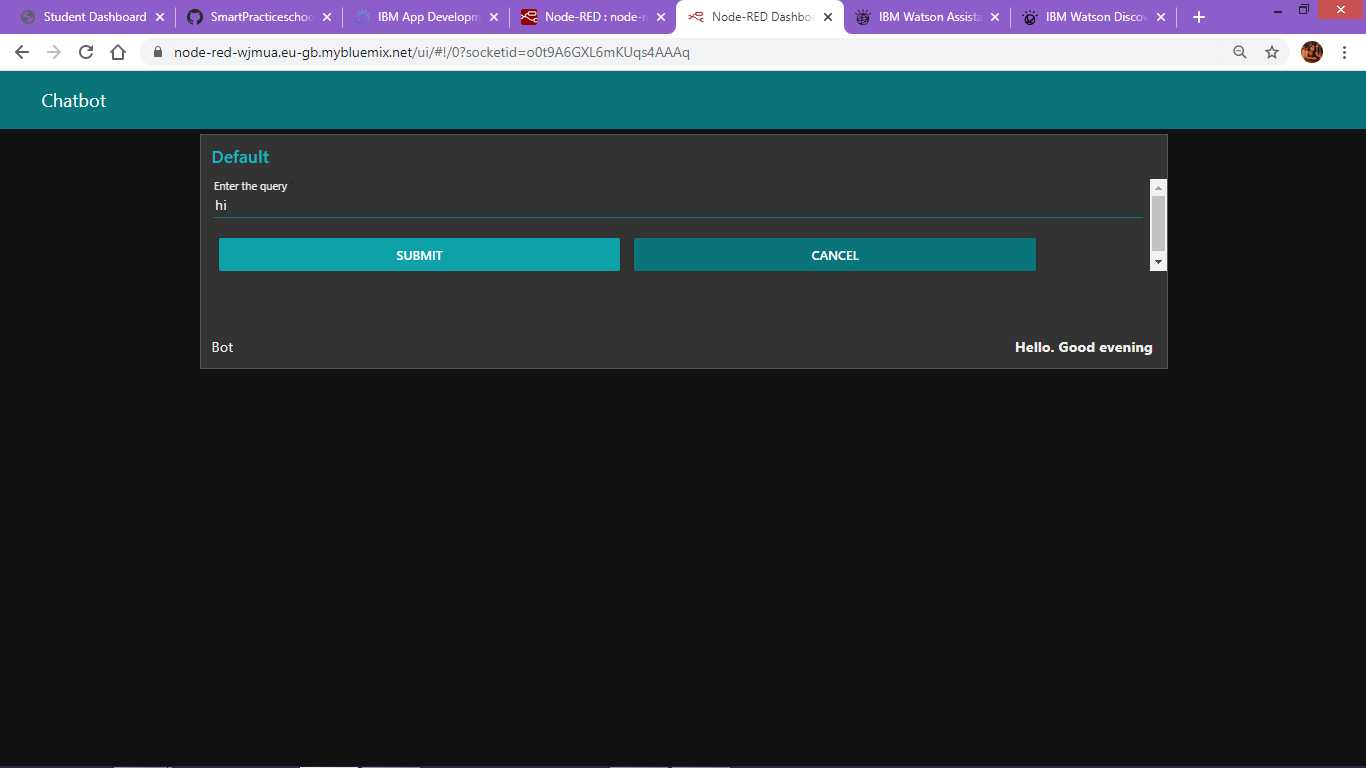


1. **Deploy and run Node Red app.**

Deploy the Node Red flow.

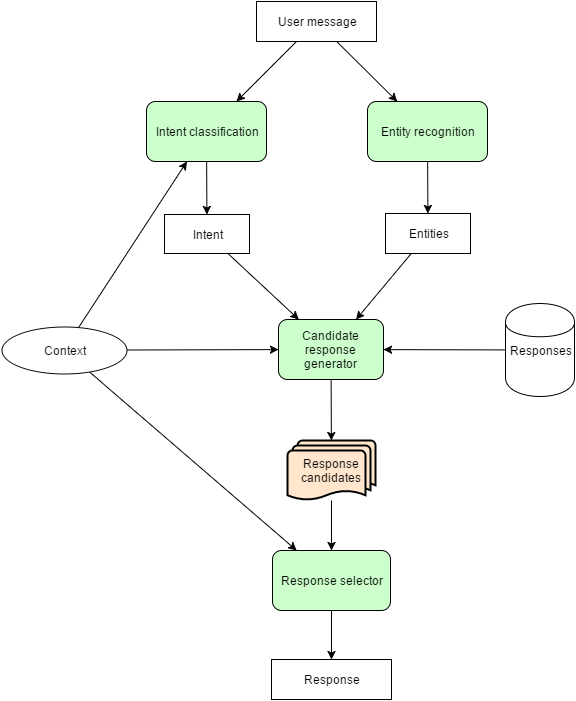
Then copy the link url upto .net/ and paste at anew tab by ui at the end of the url, like this,

[https://node-red-xvpnn.mybluemix.net/ui](https://node-red-xvpnn.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, Watsondiscovery.

**4.Experimental Investigations**

IBM Watson Assistant is trained on the entities and intents created by the developer.Whenever the backend data dialog node is triggered with user query,IBM Watson Assisstant uses cloud functions to connect to IBM Watson discovery service which is already trained with the user manual which is used to answer the customers’ queries.

The web app utilizes multiple Watson services to create a better customer care experience.

Using the Watson Discovery Smart Document Understanding (SDU) feature, we will enhance the Discovery model so that queries will be better focused to only search the most relevant information found in a typical owner's manual.

Using Watson Assistant, we will use a standard customer care dialog to handle a typical conversation between a custmomer and a company representitive. When a customer question involves operation of a product, the Assistant dialog will communicate with the Discovery service using a webhook.

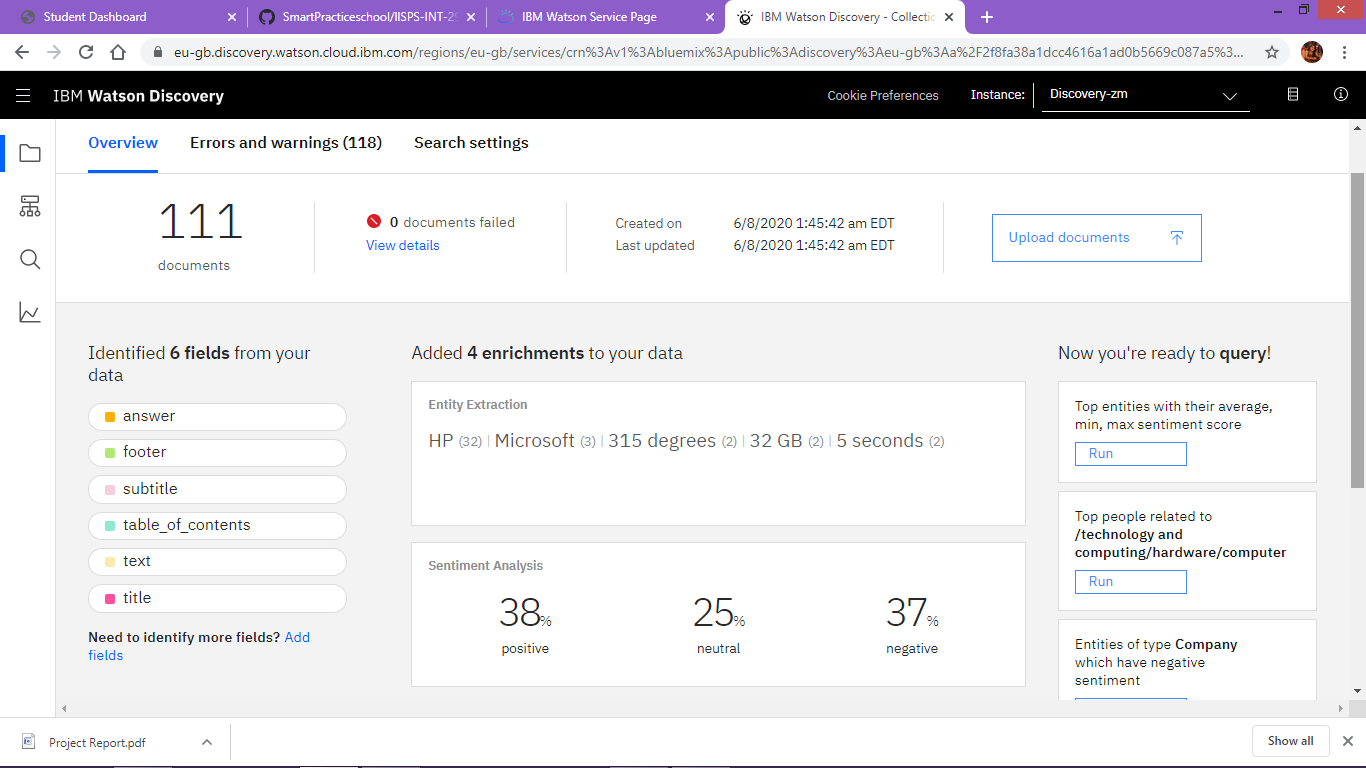
SDU trains Watson Discovery to extract custom fields in your documents. Customizing how your documents are indexed into Discovery will improve the answers returned from queries.

With SDU, you annotate fields within your documents to train custom conversion models. As you annotate, Watson is learning and will start predicting annotations. SDU models can also be exported and used on other collections.

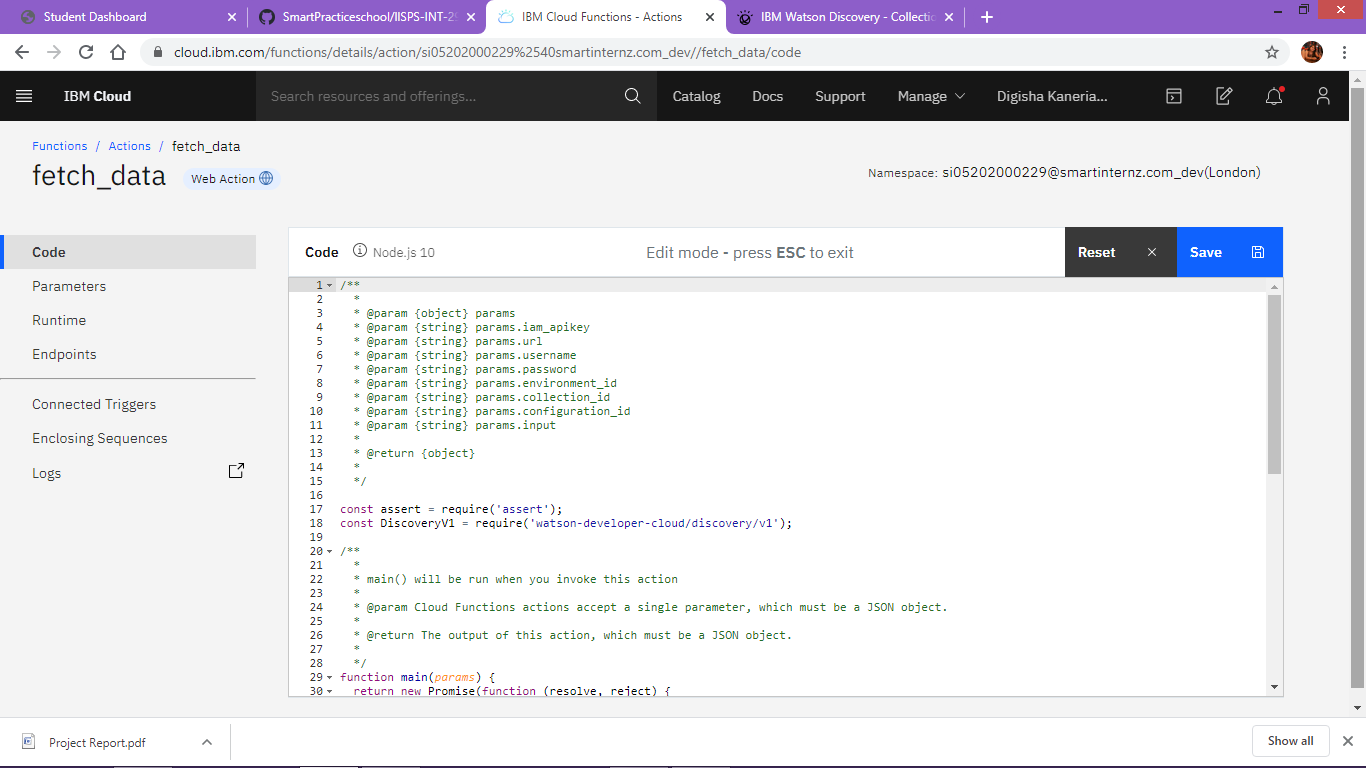
A webhook is a mechanism that allows you to call out to an external program based on something happening in your program. When used in a Watson Assistant dialog skill, a webhook is triggered when the Assistant processes a node that has a webhook enabled. The webhook collects data that you specify or that you collect from the user during the conversation and save in context variables, and sends the data to the Webhook request URL as an HTTP POST request. The URL that receives the webhook is the listener. It performs a predefined action using the information that is provided by the webhook as specified in the webhook definition, and can optionally return a response.

In our example, the webhook will communicate with an IBM Cloud Functions web action, which is connected to the Watson Discovery service.

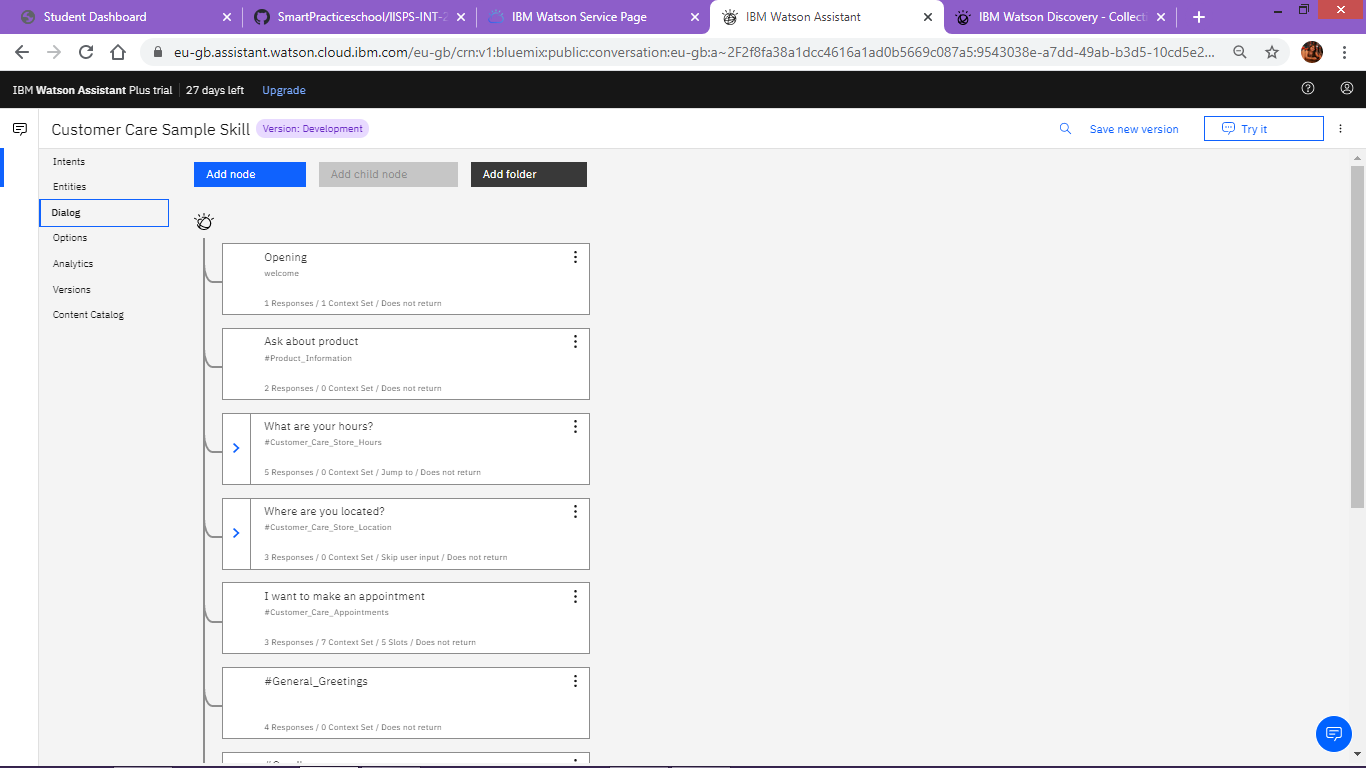
Watson Discovery:



Cloud Function:

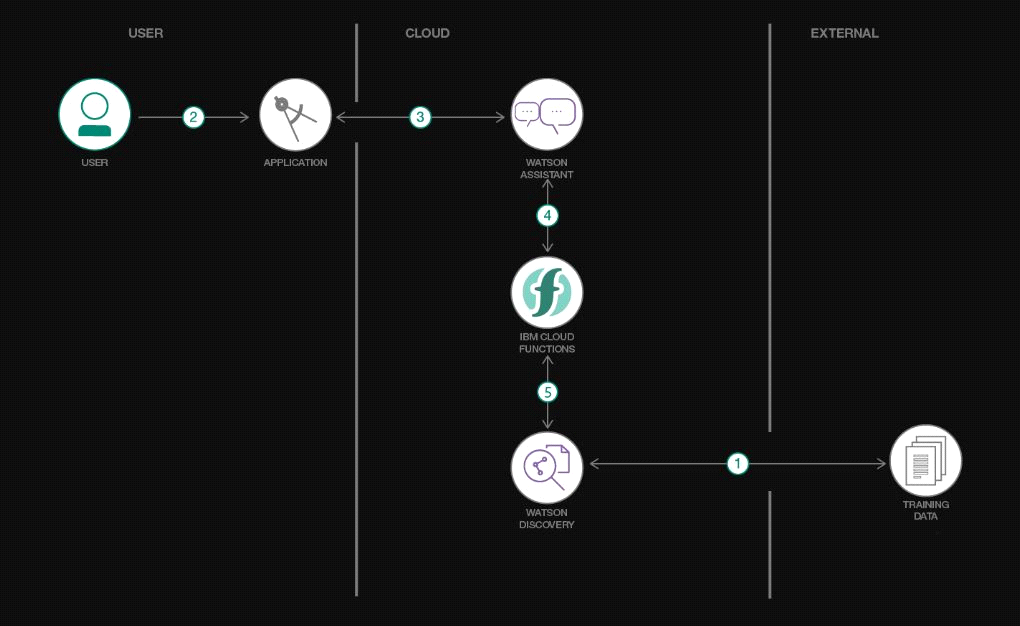
****

Watson Assistant:

****

1. **Flowchart**

This is the flow how we are going to do the tasks for the proposed problems.

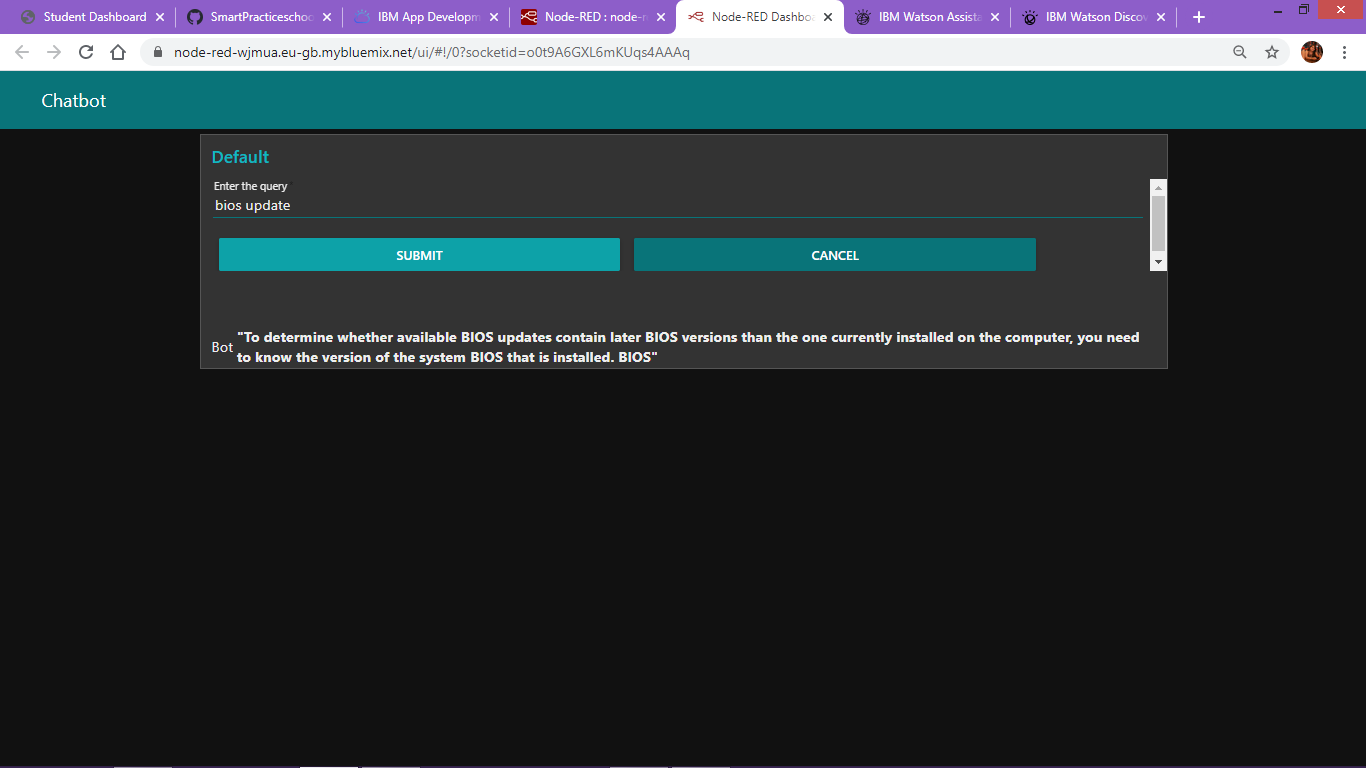


The document is annotated using Watson DiscoverySDU

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

1. 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.
2. 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.
3. 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).
4. 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 youare solving different problems at the same time. There are chatbots companies already working on developing voice chatbot services.

**Disadvantages:**

1. 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.
2. 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.
3. Lack emotions
4. Made to handle first level questions only
5. Requires maintenance
6. Zero decision-making
7. You need to train them to converse with your customers in the right way.
8. You also need to structure and optimize your knowledge base in a bot-friendly way

**8.Applications**

Some applications can be: -

1. Order Food: Various fast food giants like KFC and Pizza Hut have invested in Chatbots that enable customers to place their orders through conversations. Taco Bell went a step further to improve the conversational experience by giving their Chatbot named TacoBot some personality.
2. Book Flights: Icelandair’s chatbot gives their customers the ability to search for and book flights in a text-based conversational manner. Instead of drop-down menus, customers enter the information themselves.
3. 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.
4. Health Care: Chatbots have also made their way into health care by easing the burden on medical professionals by facilitating faster medical diagnosis, answering health-related questions, booking appointments and lots more. A Chatbot like Super Izzy can track menstrual cycles, dates and fertile windows.
5. Content delivery: Media Publishers have realized that chatbots are a powerful way to engage with theiraudiences 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.
6. 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.
7. **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.

1. **Future Scope**

The chatbot can be integrated with several messaging apps such as facebook,slack.There can also be stand alone integrations with webchat,voice telephony.The chatbot can also deal with images such as menu card of restaurant.The code for it is already been configured in the nodeflow.But we are not dealing with the images present in user manual of this project.It can also be integrated with databases(db2) to store the credentials of customers using it. For example, in case of appointments,the customer data is made to flow to backend where details is stored according to the schema of the database.The stored data can also be used for market research and with suitable tools and techniques,insights can also be derived from the stored data.In this project, the query and reply from assistant is replaced everytime when a new query arrives.This can be improved with a continuous chat flow as in other chat applications so that the information asked in previous query if forgotten need not be asked again as simple scrolling will fetch the results.

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.

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