

INTERNSHIP PROJECT REPORT

ON

**INTELLIGENT CUSTOMER HELP DESK WITH
SMART DOCUMENT UNDERSTANDING**

BY

ARNAV GAUR

(Artificial Intelligence Intern at SmartBridge)

3rd year Student at

Manipal Institute of Technology, Manipal

Pursuing

B.Tech in Computer Science and Engineering

(arnavgaur8@gmail.com)

Internship Title : IISPS-INT-1772-Intelligent-Customer-Help-Desk-with-Smart-Document-Understanding - SB24937

Project ID : SPS_PRO_99

SBID : SB20200024937

Application ID : SPS_APL_20200003545

Table of contents

1	INTRODUCTION
	1.1 Overview
	1.2 Purpose
2	LITERATURE SURVEY
	2.1 Existing problem
	2.2 Proposed solution
3	THEORITICAL ANALYSIS
	3.1 Architecture
	3.2 Software designing
4	EXPERIMENTAL INVESTIGATIONS
5	FLOWCHART
6	RESULT
7	ADVANTAGES AND DISADVANTAGES
8	APPLICATIONS
9	CONCLUSION
10	FUTURE SCOPE
11	REFERENCES

INTRODUCTION

In this project a chatbot is created which offers a complete and easy way to answer different sets of questions asked by the customers. With the help of Watson discovery channel it can also answer some typical questions about the operation of a device because we have feeds the owners manual to the watson discovery channel. The benefits of this kind of chatbot is that it is superior than the typical chatbot which can answers simple questions like store location and hours. The chatbot is upgraded with the help of watson discovery collection which is build using smart document understanding.

It's main objective is

- To solve customer's queries as early as possible to save the time of the customer.
- We will use the IBM cloud function that allows watson assistant to post queries to Watson discovery.
- The goal is to set up a remote connection between the customer and the company.

By this chatbot anyone can have their problem solved by posting queries to chatbot via being at home or without calling an employee.

1.1 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 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, a chatbot is created which offers a complete and easy way to answer different sets of questions asked by the customers. With the help of Watson discovery channel it can also answer some typical questions about the operation of a device because we have feeds the owners manual to the watson discovery channel.

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.

1.2 Purpose

- To solve customer's queries as early as possible to save the time of the customer.
- We will use the IBM cloud function that allows watson assistant to post queries to Watson discovery.
- The goal is to set up a remote connection between the customer and the company.

2. LITERATURE SURVEY

The literature review method is an examination of information on specific subject. It is reviewing what is known and not what is assumed. It aims to create the final, precise representation of the knowledge and research-based theory available topic

2.1 Existing problems

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

So, the solution is 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.

3. THEORITICAL ANALYSIS

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.

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, we are interested in the class of bots that live online in chat platforms or on social media called chatbots.
- In this context, there are many possible definitions and some confusion about what a bot is. This is partly because there are so many varied use cases for bots and these influence what people perceive a chatbot to be.
- The most intuitive definition is that a bot is software that can have a conversation with a human. For example, a user could ask the bot a question or give it an instruction and the bot could respond or perform an action as appropriate.

Types of Chatbots

To understand the nature of chatbot conversations it is important to understand that there are three types of chatbots:

- ***Scripted ChatBot*** : These are chatbots whose behaviour is determined by rules. Conversations with this type of chatbot can only follow predetermined paths. At each step in the conversation the user will need to pick from explicit options to determine the next step in the conversation. How the options are presented to the user at each step in the conversation, i.e. whether they need a text, voice or touch response will depend on the features of the chat platform and how the bot is programmed that the user is on and the design of the bot.
- ***Intelligent ChatBot*** : Intelligent chatbots are built with artificial intelligence techniques. Artificial intelligence allows them to be more flexible in terms of the user input they can accept. They can accept free form input in the form of text or voice statements (but of course they are not limited to other forms of input if that makes sense). AI also allows them to improve the more that they are used. It should be noted however that although AI works very well in very limited knowledge domains, or for one off instructions, the actual intelligence of the bot is limited. It is extremely difficult to get a bot to “understand” context or ambiguity or to have a useful memory that influences the conversation.
- ***Application ChatBot*** : Both scripted and intelligent chatbots can have graphical user interfaces. As mentioned, both scripted and intelligent chatbots can have graphical user interfaces. Application bots is therefore not a separate category of bots per say. The fact that the bots can be interacted with using a graphical user interface is an important concept for chatbot developers. If a user can do the job they need to do more efficiently via a graphical interface then the bot needs to show a graphical interface at that point in the conversation.

Why chatbots are important?

Chatbot applications streamline interactions between people and services, enhancing customer experience. At the same time, they offer companies new opportunities to improve the customer engagement 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. Human support plays a key role here: Regardless of the kind of approach and the platform, human intervention is crucial in configuring, training and optimizing the chatbot system.

How a chatbot works ?

A chatbot performs two main tasks

1. User request analysis
2. Returning response

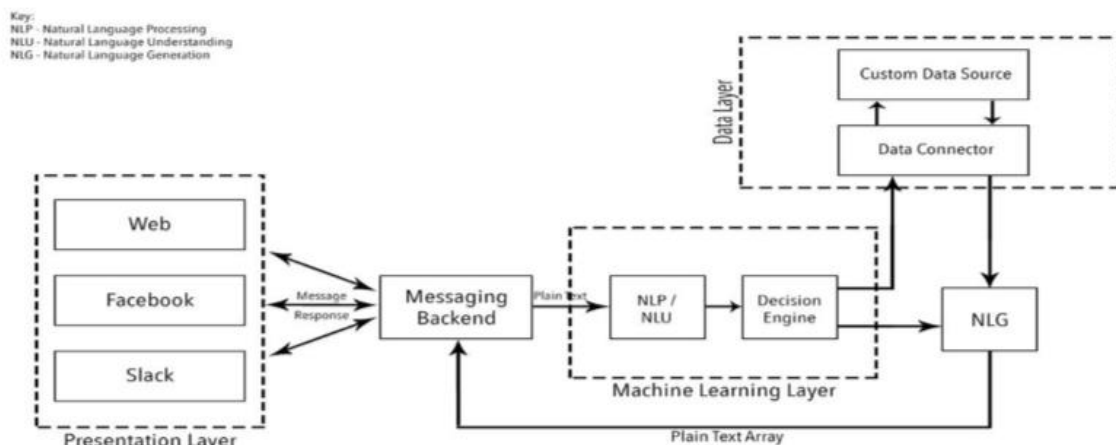


The ability to identify the user's intent and extract data and relevant entities contained in the user's request is the first condition and the most relevant step at the core of a chatbot: If you are not able to correctly understand the user's request, you won't be able to provide the correct answer. 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

ARCHITECTURE OF A CHATBOT



SOFTWARE DESIGNING

The software which are required to build the chatbot are:

1. IBM watson services

2. IBM Assistant
3. IBM cloud
4. Github
5. Node red
6. User interface
7. JSON Editor

Creating a chatbot on IBM cloud

Process for creating a Restaurant Chatbot using IBM cloud is shown below. To build a chatbot 3 important things are to be built, they are

- Intents : An intent is a collection of user statements that have the same meaning. By creating intents, you train your assistant to understand the variety of ways users express a goal. (represented using #)
- Entities : Entities are like nouns or keywords. By building out your business terms in entities your assistant can provide targeted responses to queries. (represented using @)
- Dialog : Bot responses to the user queries are mentioned in Dialogue. Dialog contains two pre defined nodes. One is Welcome node and the other is Anything else node. The Welcome contains the text to be displayed to start the conversation. Anything else node triggers when no node conditions are satisfied. All the other node lie between these two nodes only.

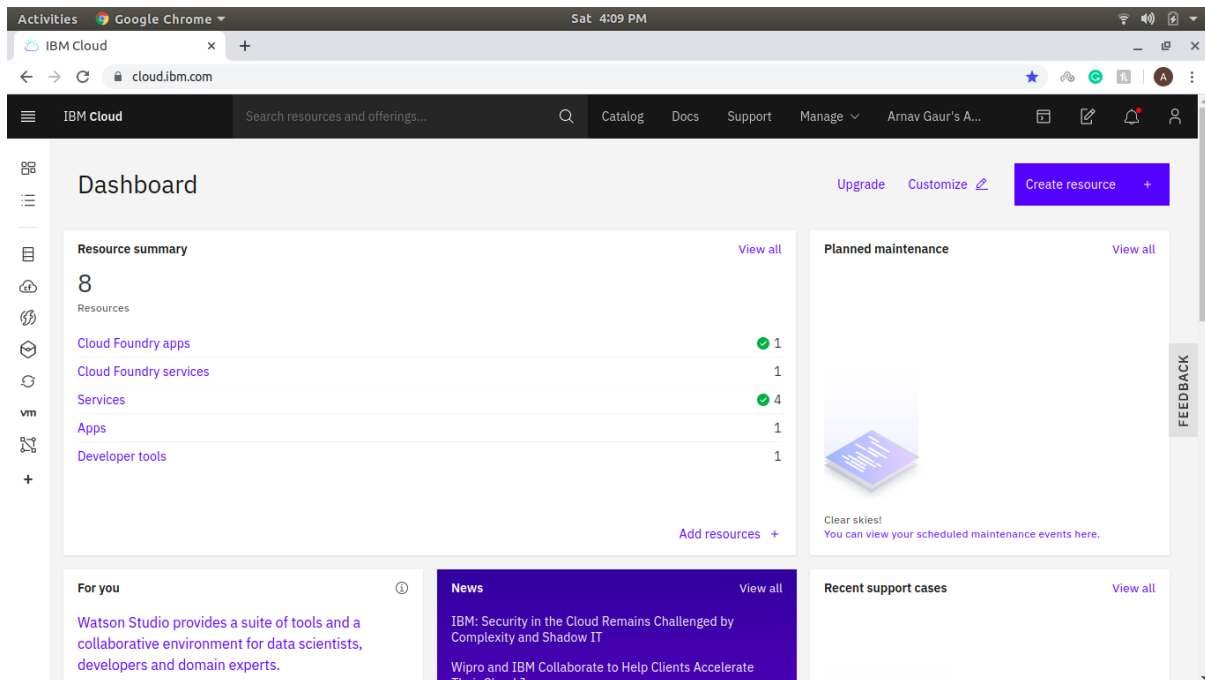
Apart from these three, context variables are used to store user mentioned content which bot have to remember for future conversation like the person's name, email-id or any other details.

In the given example 2 context variables are used one for storing the item name for which the order has to be placed and other for storing the number i.e., the quantity of the food_item for which the order has to be placed. These contexts are stored in the bots memory and once the processing of the information is done these contexts can be deleted using simple JSON code.

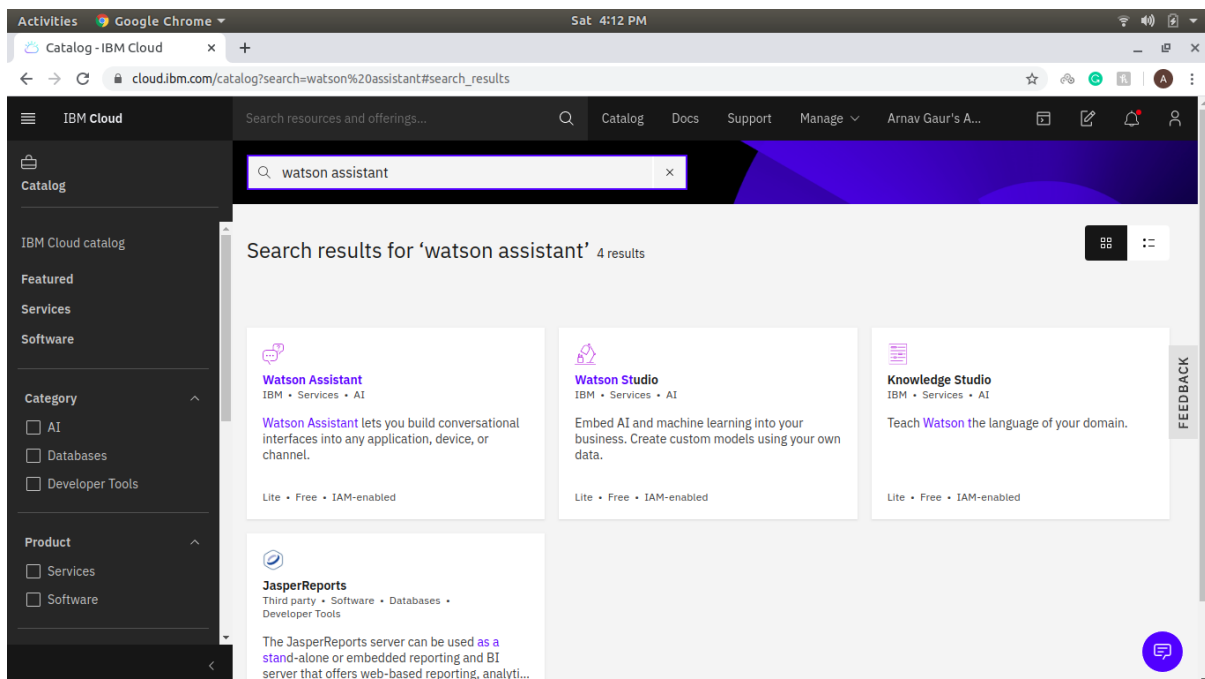
At the end of the process the bot is integrated to a sample page provided by IBM. Apart from the page, the bot can also be integrated to Third Party Integrations like Slack, Facebook page, Intercom or Standalone Integrations like Web Chats or Preview link or as a Wordpress plugin.

STEPS:

1. Create an IBM account by logging in into <https://cloud.ibm.com/> and clicking on Create an IBM account.
2. Fill the details and create an IBM account
3. Verify your email id and then login into IBM account by clicking on Login.
4. After the login dashboard is launched. On dashboard all the created resources will be displayed.



5. To create a chatbot on IBM cloud we require Watson Assistant service. All the available resources on the cloud are found under the catalog section. Open Catalog. Under categories select AI and in that Watson Assistant service is found.



- Click open the Watson Assistant service. Select the region, pricing plan and give a name to the service and click on Create. It's showing me a warning because I have already created this.

Catalog / Services /

Watson Assistant

Author: IBM • Date of last update: 06/05/2020 • Docs • API docs

Create About

Select a region

Select a region

Dallas

Select a pricing plan

Displayed prices do not include tax. Monthly prices shown are for country or region: [United States](#)

Warning notification
You can have only one instance of a Lite plan per service. To create a new instance, [delete](#) your existing Lite plan instance.

Plan	Features	Pricing
Lite	10,000 Messages/Month AI-Based Intent and Entity Recognition Entity Synonym Recommendations Visual Dialog Edit with Simple Response Types (Text, Options, Images, etc...) Prebuilt Content Available	Free

Create

Add to estimate

[View terms](#)

FEEDBACK

- On launch page click on Launch Watson Assistant

Resource list /

Watson Assistant-fq

Active Add tags

Details Actions...

Manage

Service credentials

Plan

Connections

Start by launching the tool

Launch Watson Assistant

Getting started tutorial

API reference

Plan

Lite

Upgrade

Credentials

Download Show credentials

API key:

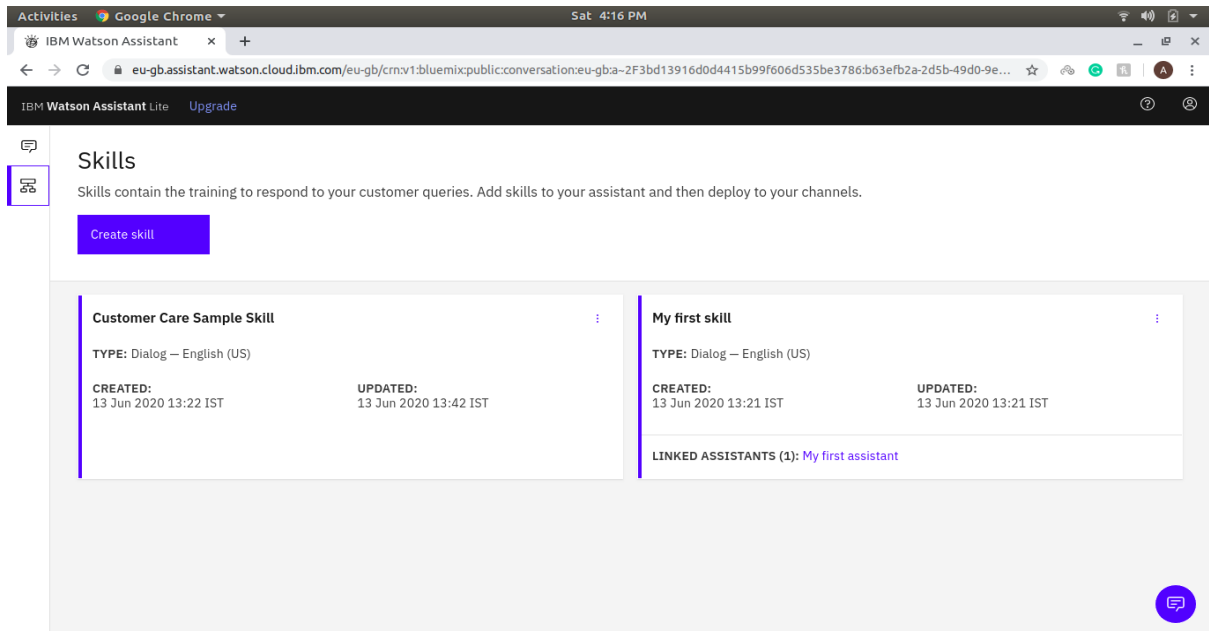
.....

URL:

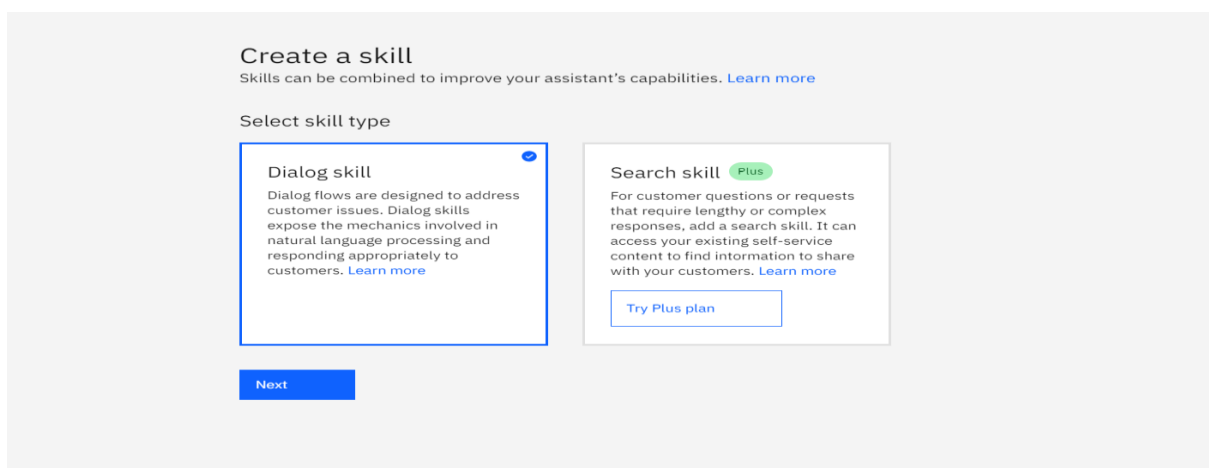
https://api.us-south.assistant.watson.cloud.ibm.com/instances/9ebd35a1-1e6f-4050-bb78-0418d...

FEEDBACK

8. Assistant page is launched. On the left pane click on skills.



9. On the launch page click on Create Skill. Select the skill type to be Dialog Skill and click on Next.



10. Skill can be created in 3 different ways. Select Create Skill tab and give a name to your skill (also optionally description) of the skill and click on Create a Dialog Skill.

The screenshot shows the 'Create dialog skill' form in the IBM Watson Assistant interface. The form has three tabs: 'Create skill' (selected), 'Use sample skill', and 'Import skill'. The 'Create skill' tab contains the following fields:

- Name:** A text input field with the value 'my first bot'.
- Description (optional):** A text area with the placeholder text 'Add a description for this skill'.
- Language:** A dropdown menu with 'English (US)' selected.
- Create dialog skill:** A blue button at the bottom.

The top of the interface shows the 'IBM Watson Assistant Lite' header with an 'Upgrade' link and a close button (X) in the top right corner.

11. You can create Intents, Entities and Dialog. and Click on Try it on the right side of the screen and check try giving the greetings and check the response.

The screenshot shows the 'Customer Care Sample Skill' interface in the IBM Watson Assistant. The left sidebar contains a navigation menu with the following items: 'Intents' (selected), 'Entities', 'Dialog', 'Options', 'Analytics', 'Versions', and 'Content Catalog'. The main area displays a table of intents.

<input type="checkbox"/>	Intents (10) ↑	Description	Modified T1	Examples T1
<input type="checkbox"/>	#Cancel	Cancel the current request	3 hours ago	7
<input type="checkbox"/>	#Customer_Care_Appointments	Schedule or manage an in-store appointment.	3 hours ago	20
<input type="checkbox"/>	#Customer_Care_Store_Hours	Find business hours.	3 hours ago	48
<input type="checkbox"/>	#Customer_Care_Store_Location	Locate a physical store location or an address.	3 hours ago	25
<input type="checkbox"/>	#General_Connect_to_Agent	Request a human agent.	3 hours ago	47
<input type="checkbox"/>	#General_Greetings	Greetings	3 hours ago	30
<input type="checkbox"/>	#Goodbye	Good byes	3 hours ago	6

At the bottom of the table, it says 'Showing 1–10 of 10 intents'. On the right side of the table, there are controls for '1' (selected), '1 of 1 pages', and a 'Try it' button. The top of the interface shows the 'IBM Watson Assistant Lite' header with an 'Upgrade' link and a 'Try it' button.

Activities Google Chrome Sat 4:19 PM

IBM Watson Assistant x +

eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn:v1:bluemix:public:conversation:eu-gb:a-2F3bd13916d0d4415b99f606d535be3786:b63efb2a-2d5b-49d0-9e...

IBM Watson Assistant Lite Upgrade

Customer Care Sample Skill

Save new version Try it

Intents

Entities

My entities

System entities

Dialog

Options

Analytics

Versions

Content Catalog

Entity (6) ↑	Values	Modified ↑↓
@holiday	new years eve, valentine's day, independence day, labor day, christmas eve, hallow...	3 hours ago
@landmark	empire state building, times square, grand central	3 hours ago
@phone	US Phone pattern	3 hours ago
@reply	no, yes	3 hours ago
@specialist	Maria, Derrik, Nicholas, Barbara, Brenda, Robert	3 hours ago
@zip_code	US Zip	3 hours ago

Showing 1–6 of 6 entities

1 1 of 1 pages

Create entity +

Activities Google Chrome Sat 4:19 PM

IBM Watson Assistant x +

eu-gb.assistant.watson.cloud.ibm.com/eu-gb/crn:v1:bluemix:public:conversation:eu-gb:a-2F3bd13916d0d4415b99f606d535be3786:b63efb2a-2d5b-49d0-9e...

IBM Watson Assistant Lite Upgrade

Customer Care Sample Skill

Save new version Try it

Intents

Entities

Dialog

Options

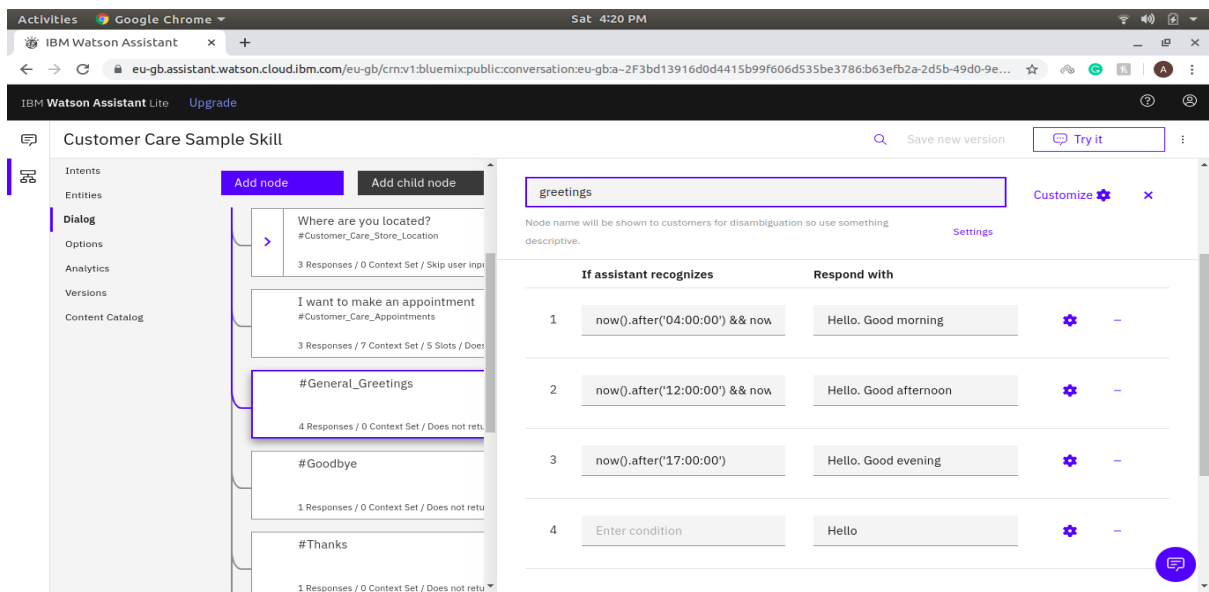
Analytics

Versions

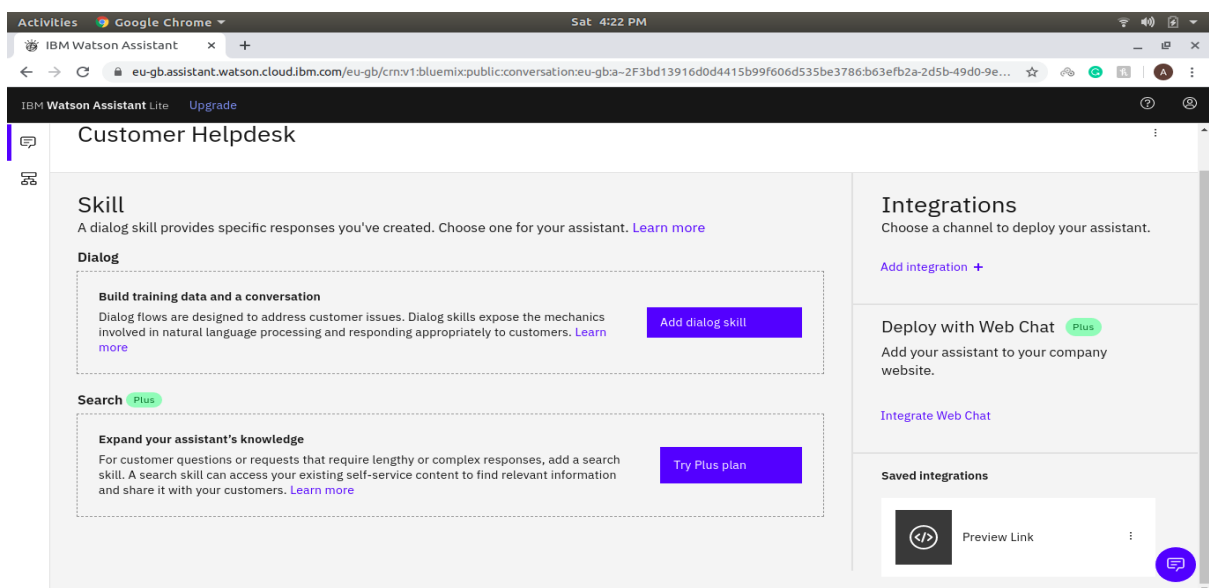
Content Catalog

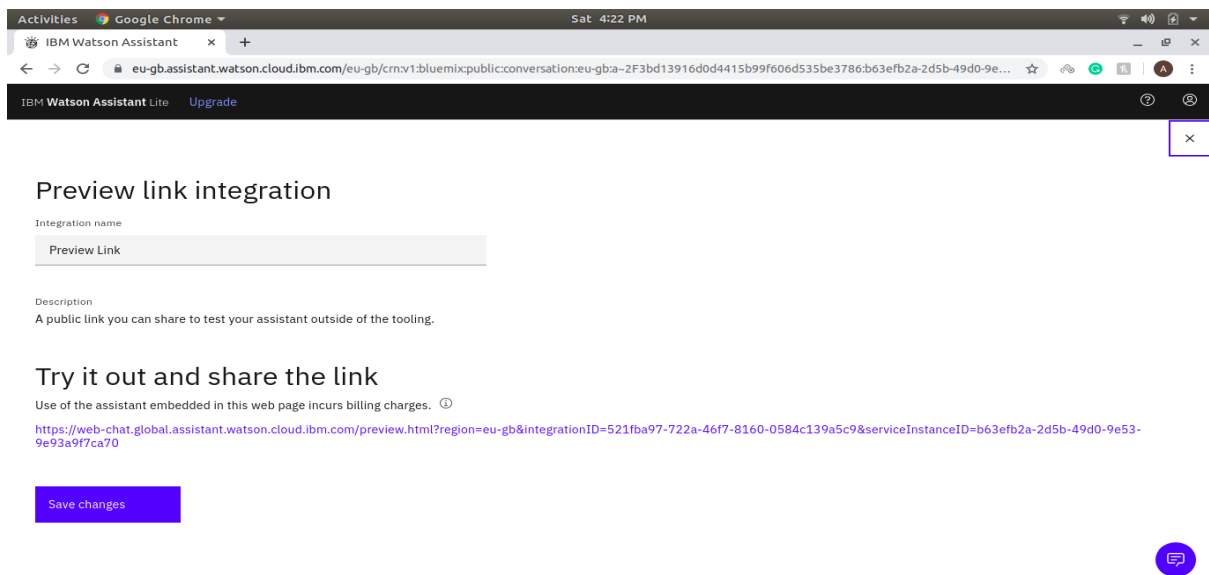
Add node Add child node Add folder

- Opening
welcome
1 Responses / 1 Context Set / Does not return
- > What are your hours?
#Customer_Care_Store_Hours
5 Responses / 0 Context Set / Jump to / Does not return
- > Where are you located?
#Customer_Care_Store_Location
3 Responses / 0 Context Set / Skip user input / Does not return
- I want to make an appointment
#Customer_Care_Appointments
3 Responses / 7 Context Set / 5 Slots / Does not return
- #General_Greetings
4 Responses / 0 Context Set / Does not return



12. Under saved integrations click Preview Link. Click on the link on the launch page, give a name to the integration and save the changes.



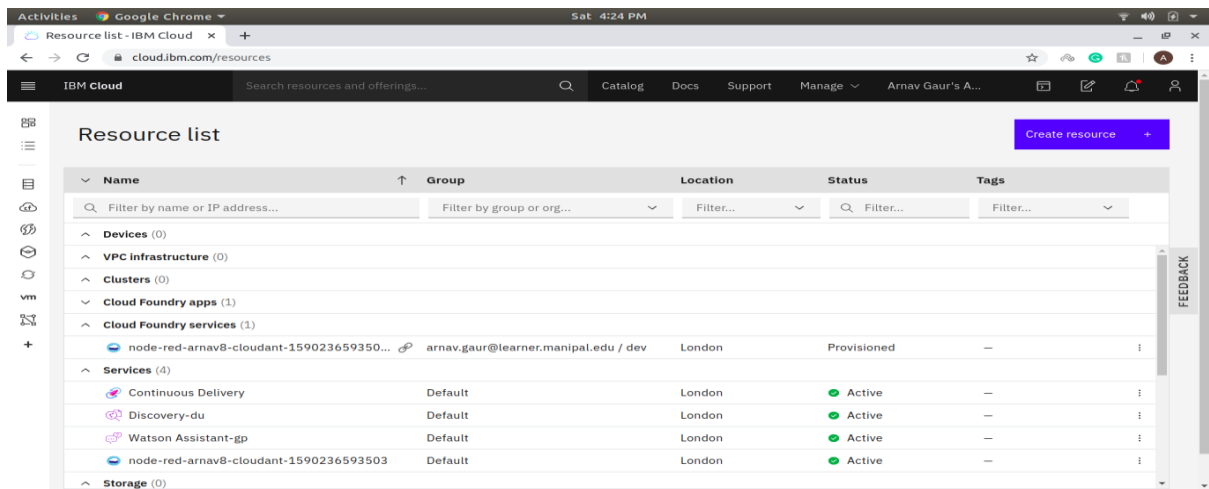


13. Test the bot with the preview given by IBM.

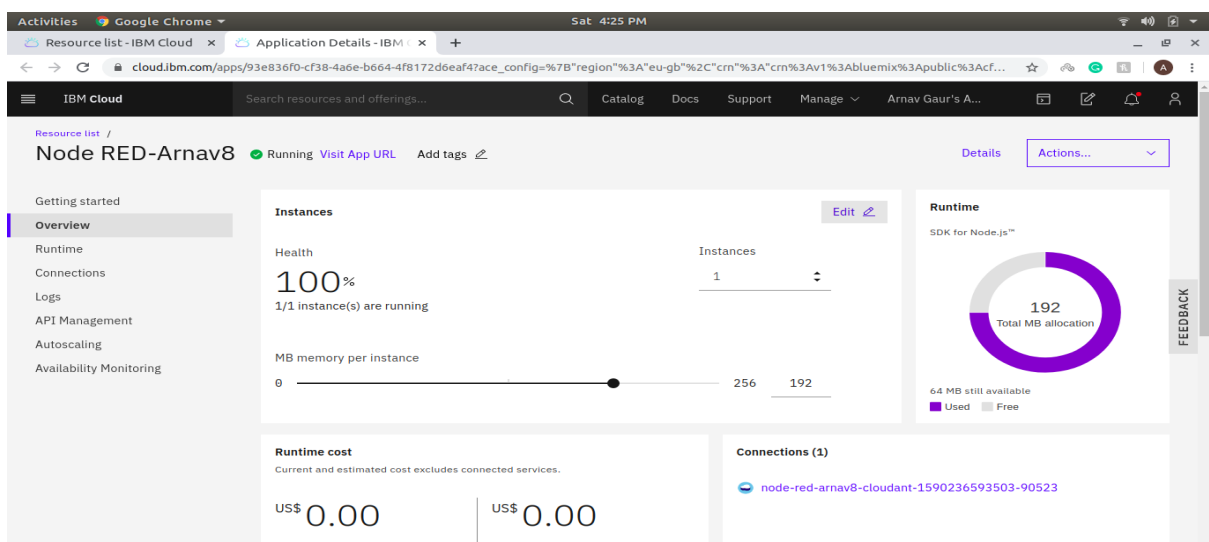
You can integrate the bot with any other third party services or html pages.

Creating UI of Watson Assistant with Node red flow (Chatbot).

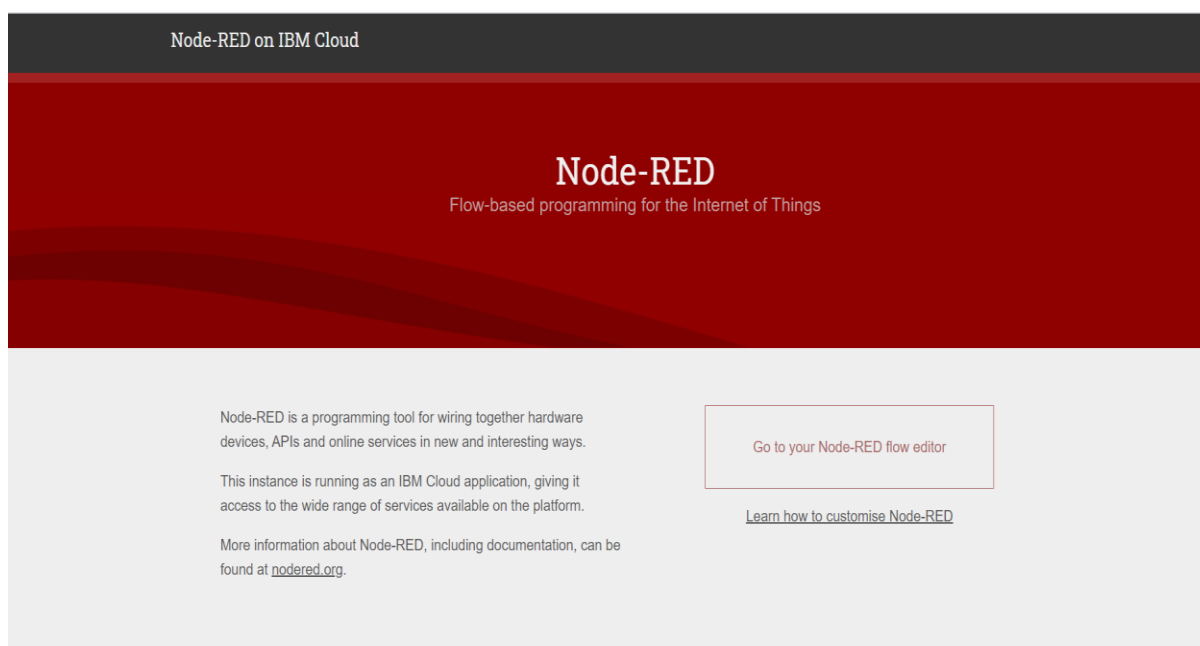
1. Click on the services in the dashboard. Go to the Cloud Foundry apps. And click on Node red app. (Here I'm assuming that you have already created the Node-red services).



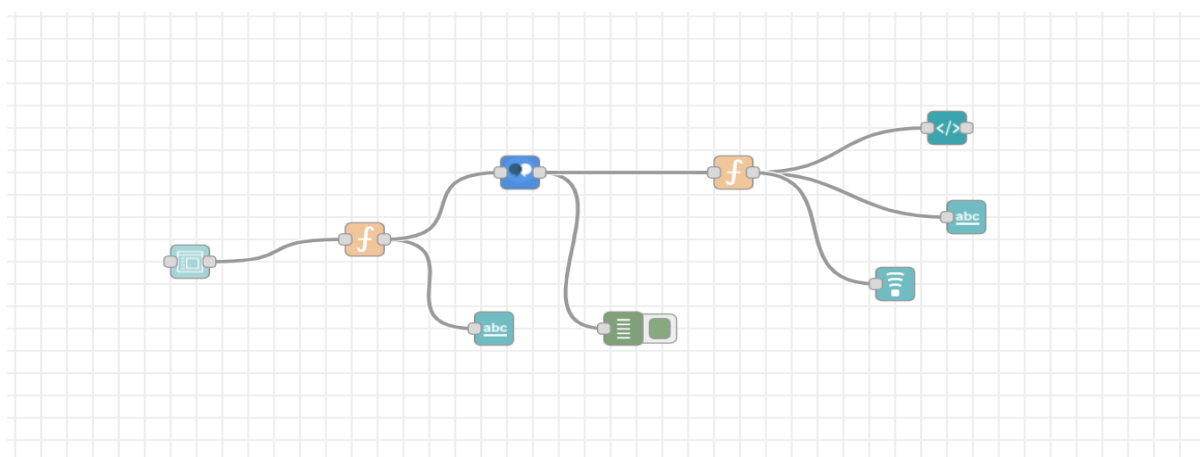
2. A node red dashboard appears as shown below:



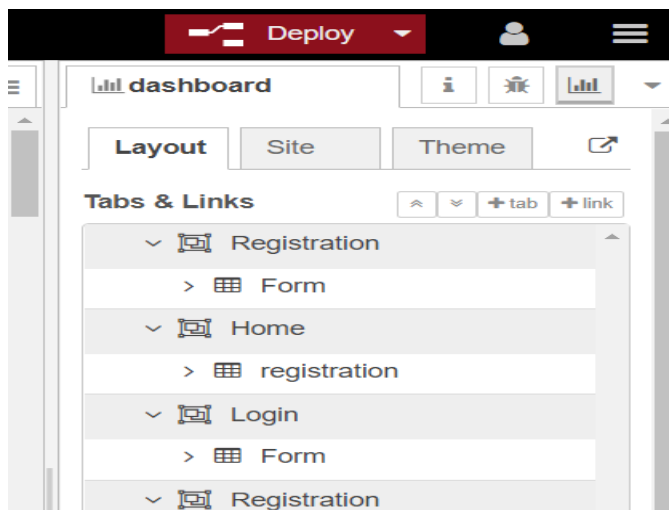
- Click on Visit app URL. A page appears. Click on Go to your node red flow Editor.



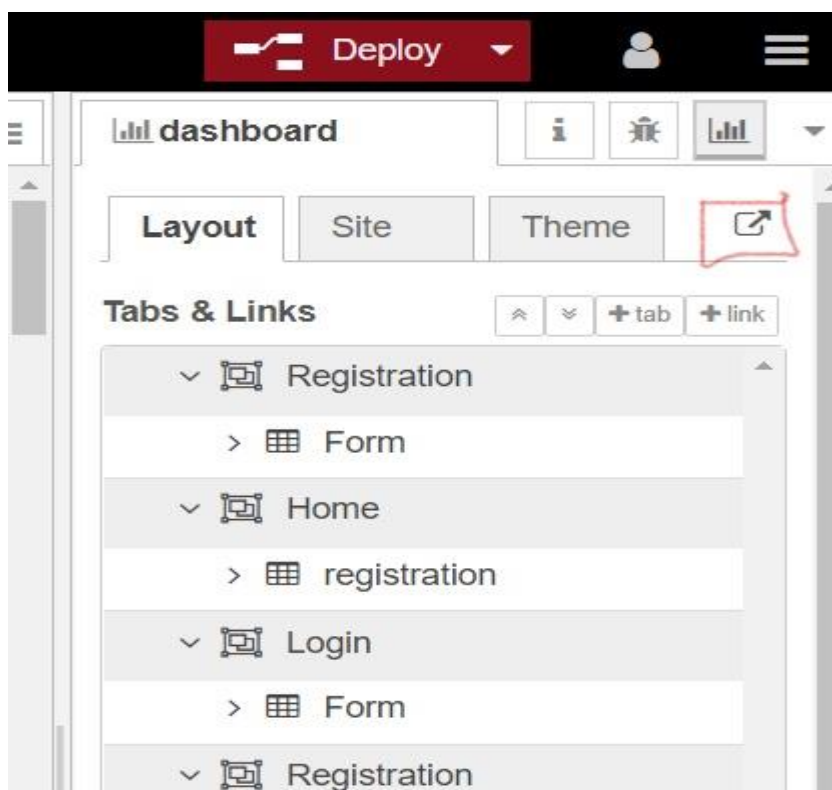
4. Here's the flow:



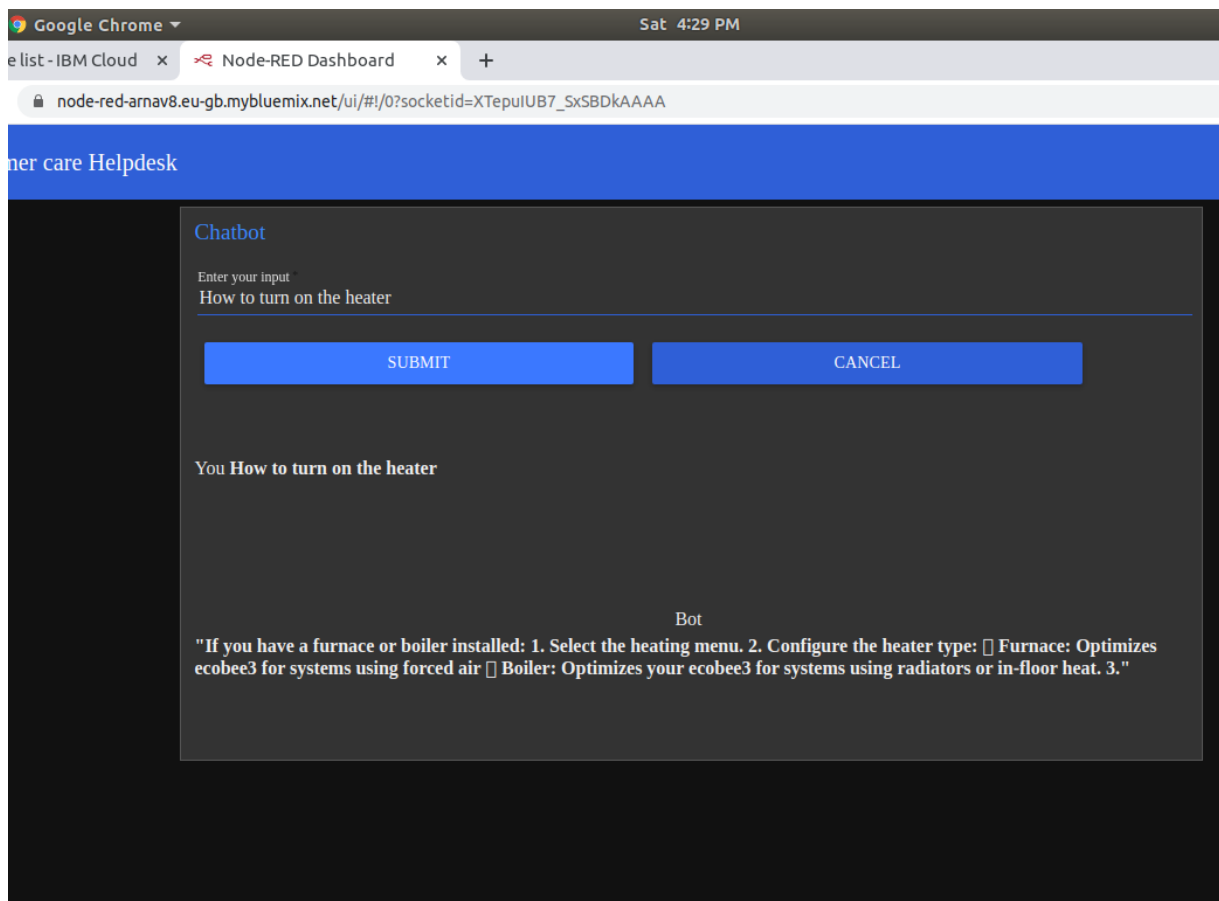
5. Deploy your Flow. Click on the dashboard. Also add the workspace Id and Service endpoint.



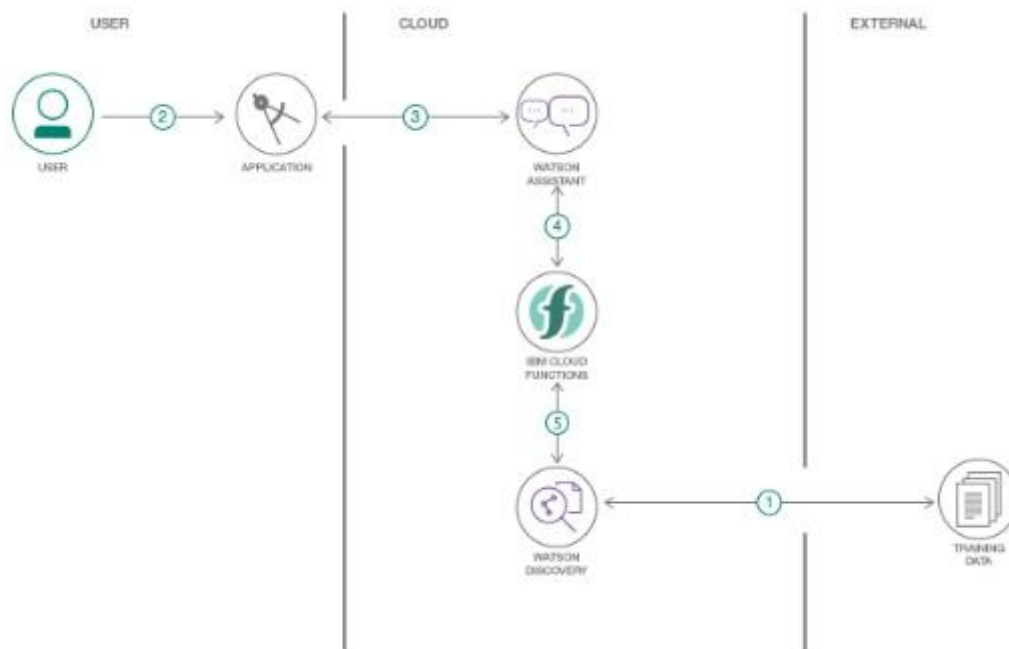
6. UI is created successfully. Click on that arrow button present near the themes.



7. The final output is shown as below. This is how the deployed chatbot looks and responds to our query.



FLOWCHART



1. The document is annotated using Watson Discovery Smart Document Understanding.
2. The user interacts with the back-end server via the app UI. The front-end app UI is a chatbot that engages the user in a conversation.
3. Dialog between the user and back-end server is coordinated using a Watson Assistant dialog skill.
4. If the user asks a product operation question, a search query is passed to a predefined IBM Cloud Functions action.
5. The IBM Cloud Functions action will query the Watson Discovery Service and return the results.

RESULTS

- We have Created a Chatbot which is able to answer queries.
- The model created i.e. a chatbot would be able to identify any operational question posted by the user and using IBM Watson discovery will redirect the user to the section of the owner's manual where the answer to the question lies.

ADVANTAGES

Chatbots have been on the rise since a couple of years and have already faced a wide adoption. They are bringing a new way for businesses to communicate with the world and most importantly with their customers by the help of exploding popularity of messaging apps, the accelerated development of all kinds of sensors and wearables and of course with the rise of emerging technologies and Artificial Intelligence (AI).

- Keeping Up with the Trends: Being Present on Messaging Platforms
- Improved Customer Service.
- Always-Available Customer Support
- Proactive Customer Interaction
- Increased Customer Engagement

DISADVANTAGES

This definition however often leads to two potential misconceptions.

1. The biggest misconception that arises is that a chatbot is a bot that converses with a human in the way that another human would converse with a human. Software or even a robot (the digital part of the robot is of course software) that communicates with a human in natural language is not difficult to imagine. Science fiction is full of examples. While this may be the end goal, this is simply not possible using the current technology. Not only is it not possible, it often leads to unrealistic expectations regarding the chatbots capabilities and inevitable frustrations when those expectations are not met.
2. The second misconception is that a chatbot communicates using only text or voice. Actually chatbots allow users to interact with them via graphical

interfaces or graphical widgets, and the trend is in this direction. Many chat platforms including WeChat, Facebook Messenger and Kik allow web views on which developers can create a completely customized graphical interfaces.

APPLICATIONS

A Chatbot is a program that can have a conversation with a person using rules and Artificial Intelligence (AI) in a way that mimics human-like conversations and interactions. Chatbots have become popular in the past few years as businesses discover innovative ways to put them to use. Having a Chatbot today has numerous benefits for businesses – they make life easier for customers, are available 24/7, save time (no more long waits to talk to a service rep) and they are easy to use.

- **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.
- **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.
- **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.
- **Companionship:** Russian technology company Endurance developed its companion chatbot for Senior People and Patients with Alzheimer's Disease. 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

- **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.

CONCLUSION

There is more to building chatbots and conversational UI than just plugging tools, services, and data together. It takes practice and a deeper understanding of underlying concepts to get the design right and build bots that give users a great experience. The user should be able to get the job done by having a conversation with the bot without having to think too much and with a smile on their face.

From my perspective, chatbots or smart assistants with artificial intelligence are dramatically changing businesses. There is a wide range of chatbot building platforms that are available for various enterprises, such as e-commerce, retail, banking, leisure, travel, healthcare, and so on.

Chatbots can reach out to a large audience on messaging apps and be more effective than humans. They may develop into a capable information-gathering tool in the near future.

FUTURE SCOPE

Chatbots are hot software in the enterprise, but to maintain longevity and relevance, developers need to take a look at the barriers to entry, interface options and NLP issues.

From gauging purchase intent to answering questions about IT issues, chatbots are on track to play a major role in the contemporary enterprise. Chatbots are fully functioning, semi-autonomous systems that can assist customer service experiences and response time.

The clearest use of chatbots right now is in customer service and online ordering, where it can automate (and in some cases solve) customer issues or complete orders without human interaction.

- Adding Natural Language Processing in the Bot to understand the User Statements.
- Adding Sentiment Analysis to predict User Sentiment during the Chat.
- Use Voice Capabilities of the Bot.
- Use Voice Recognition with Bot.

11. BIBLIOGRAPHY

- **Node-RED Starter Application:-**

<https://developer.ibm.com/tutorials/how-tocreate-a-node-red-starter-application/>

- **Watson Assistant Guide:-**

<https://www.ibm.com/cloud/watson-assistant/>

<https://developer.ibm.com/components/watson-assistant/series/learningpath-watson-assistant>

- **Watson Discovery Guide:-**

<https://developer.ibm.com/articles/introductionwatson-discovery/>

- **Watson Assistant with Webhooks:-**

<https://www.youtube.com/embed/5z3i5IsBVnk>

- **Cloud Functions Action:-**

<https://www.youtube.com/embed/G3bqRndQtQg>

12. APPENDIX

For source code visit:-

<https://github.com/SmartPracticeschool/IISPS-INT-1772-Intelligent-Customer-Help-Desk-with-Smart-Document-Understanding>

Code for **cloudfunction.json**:-

```
/**
 *
 * @param {object} params
 * @param {string} params.iam_apikey
 * @param {string} params.url
 * @param {string} params.username
 * @param {string} params.password
 * @param {string} params.environment_id
 * @param {string} params.collection_id
 * @param {string} params.configuration_id
 * @param {string} params.input
 *
 * @return {object}
 */

const assert = require('assert');
const DiscoveryV1 = require('watson-developer-cloud/discovery/v1');

/**
 *
 * main() will be run when you invoke this action
 *
 * @param Cloud Functions actions accept a single parameter, which must be
a JSON object.
 *
 * @return The output of this action, which must be a JSON object.
 */
function main(params) {
  return new Promise(function (resolve, reject) {

    let discovery;

    if (params.iam_apikey) {
      discovery = new DiscoveryV1({
        'iam_apikey': params.iam_apikey,
        'url': params.url,
        'version': '2019-03-25'
      });
    }
    else {
      discovery = new DiscoveryV1({
        'username': params.username,
        'password': params.password,
        'url': params.url,
        'version': '2019-03-25'
      });
    }

    discovery.query({
      'environment_id': params.environment_id,
      'collection_id': params.collection_id,
      'natural_language_query': params.input,
```

```

    'passages': true,
    'count': 3,
    'passages_count': 3
  }, function(err, data) {
    if (err) {
      return reject(err);
    }
    return resolve(data);
  });
});
}

```

Code for **flow.json**:-[

```

{
  "id": "4fcc0436.303d0c",
  "type": "tab",
  "label": "Flow 1",
  "disabled": false,
  "info": ""
},
{
  "id": "186026e0.7bf459",
  "type": "ui_form",
  "z": "4fcc0436.303d0c",
  "name": "",
  "label": "",
  "group": "d8cdca57.ac2728",
  "order": 1,
  "width": 12,
  "height": 2,
  "options": [
    {
      "label": "Enter your input",
      "value": "text",
      "type": "text",
      "required": true,
      "rows": null
    }
  ],
  "formValue": {
    "text": ""
  },
  "payload": "",
  "submit": "submit",
  "cancel": "cancel",
  "topic": "",
  "x": 80,
  "y": 260,
  "wires": [
    [
      "324e744.9ef758c"
    ]
  ]
}

```

```

    ]
  ],
  {
    "id": "324e744.9ef758c",
    "type": "function",
    "z": "4fcc0436.303d0c",
    "name": "",
    "func": "msg.payload=msg.payload.text;\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 190,
    "y": 180,
    "wires": [
      [
        "286aa77d.d4aed8",
        "e3ad0814.7e0ca8"
      ]
    ]
  },
  {
    "id": "1bdfdfbc.5cdd4",
    "type": "function",
    "z": "4fcc0436.303d0c",
    "name": "",
    "func": "msg.payload=msg.payload.output.text[0];\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 620,
    "y": 220,
    "wires": [
      [
        "3be5a015.0e12e"
      ]
    ]
  },
  {
    {
      "id": "286aa77d.d4aed8",
      "type": "watson-conversation-v1",
      "z": "4fcc0436.303d0c",
      "name": "Customer Care",
      "workspaceid": "91144cdd-1b8f-4c3c-909d-f2a1fd0c5ef0",
      "multiuser": false,
      "context": true,
      "empty-payload": false,
      "service-endpoint": "https://api.eu-
gb.assistant.watson.cloud.ibm.com/instances/b63efb2a-2d5b-49d0-9e53-
9e93a9f7ca70",
      "timeout": "",
      "optout-learning": false,
      "x": 400,
      "y": 140,
      "wires": [
        [
          "78fad62e.32c228",
          "1bdfdfbc.5cdd4"
        ]
      ]
    }
  }
}

```

```

    ]
  },
  {
    "id": "e3ad0814.7e0ca8",
    "type": "ui_text",
    "z": "4fcc0436.303d0c",
    "group": "d8cdca57.ac2728",
    "order": 2,
    "width": 12,
    "height": 1,
    "name": "",
    "label": "You",
    "format": "{{msg.payload}}",
    "layout": "row-left",
    "x": 300,
    "y": 300,
    "wires": []
  },
  {
    "id": "78fad62e.32c228",
    "type": "debug",
    "z": "4fcc0436.303d0c",
    "name": "",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "false",
    "x": 620,
    "y": 80,
    "wires": []
  },
  {
    "id": "3be5a015.0e12e",
    "type": "ui_text",
    "z": "4fcc0436.303d0c",
    "group": "d8cdca57.ac2728",
    "order": 3,
    "width": 12,
    "height": 3,
    "name": "",
    "label": "Bot",
    "format": "{{msg.payload}}",
    "layout": "col-center",
    "x": 770,
    "y": 300,
    "wires": []
  },
  {
    "id": "d8cdca57.ac2728",
    "type": "ui_group",
    "z": "",
    "name": "Chatbot",
    "tab": "462d23d2.b39c8c",
    "order": 1,
    "disp": true,
    "width": 12,

```

```
        "collapse": false
    },
    {
        "id": "462d23d2.b39c8c",
        "type": "ui_tab",
        "z": "",
        "name": "Customer care Helpdesk",
        "icon": "dashboard",
        "disabled": false,
        "hidden": false
    }
]
```

-----**END**-----