**Project Report**

**Intelligent Customer Help Desk with Smart Document Understanding**

Created By: Zeel Prajapati

Email: prajapatizeel75@gmail.com

**1. Introduction:**

**1.1 Overview**

We will build a chatbot that uses various Watson AI Services (Watson Discovery, Watson Assistant, Watson Cloud Functions and Node-Red) to deliver an effective Web based UI through which we can chat with the assistant. We will integrate the Watson Discovery service with Watson Assistant using webhooks.

● Project Requirements: Node-RED, IBM Cloud, IBM Watson, Node JS

● Functional Requirements: IBM Cloud

● Technical Requirements: AI, ML, Watson AI, Node JS

● Software Requirements: Watson Assistant, Watson Discovery, Watson Cloud Functions, Node-RED

**1.2. Purpose**

The aim of the internship was to explore, understand and utilize the different services provided by the IBM and combining them to create a useful application to tackle real life problems. The project is to develop a smart customer care chatbot which can not only answer the normal questions but can also resolve some complex queries of customers with some proper training instead of returning some hard-coded answer like “I didn’t understand the question, please rephrase it.” This can save a lot of time of customer care representatives and the customers. The chatbot can be trained with the manual of a particular product and then can answer the queries about device operations

**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. And the question remains unanswered in most cases

**2.2 PROPOSED SOLUTION**

To solve the above-mentioned problem, we will introduce a Watson Discovery feature that is Smart Document Understanding. 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 owner’s manual. So now, instead of “Would you like to speak to a customer representative?” we can return relevant sections of the owner’s manual to help solve our customers’ problems.

**3. Theoretical Analysis Block / Flow Diagram**



Hardware / Software Designing

1. Create necessary Watson Services.

2. Configure Watson Discovery.

3. Create Watson Cloud Functions Action.

4. Configure Watson Assistant.

5. Integrate Watson Discovery with Watson Assistant using webhook.

6. Build Node-RED flow to integrate Watson Assistant and Web Dashboard.

3.2. Hardware/Software Designing

i. Create IBM Cloud services

ii. Configure Watson Discovery

iii. Create IBM Cloud Functions action

iv. Configure Watson Assistant

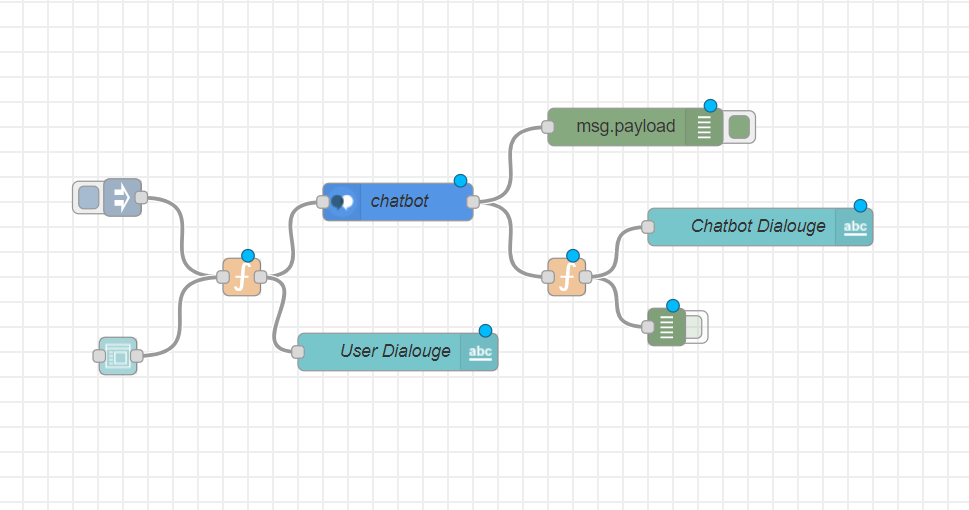
v. Create flow and configure node

vi. Deploy and run Node Red web-app

**4. Experimental Investigation**

1. Configure Watson Discovery Upload the required document and annotate all the text with different field like title, subtitle, text, footer etc. Index the document by Title, subtitle and text field and divide the document with the subtitle fields Run the test queries to see if the SDU works properly
2. Setting up IBM Cloud Function IBM cloud function will be used to make queries against our Discovery collection. The function will be given some input and it will pass this input to the Watson discovery and fetch the output.

**5. Flowchart**



**6. Results**

Web based UI was developed by integrating all the services using Node-RED.

URL for UI Dashboard: <https://node-red-hprbw.eu-gb.mybluemix.net/ui/>

**7. PRONS AND CONS**

**PROS:**

1. Companies can deploy chatbots to rectify simple and general human queries.
2. Reduces man power
3. Cost efficient
4. No need to divert calls to customer agent and customer agent can look on other works.

**CONS:**

1. Some times chatbot can mislead customers
2. Giving same answer for different sentiments
3. Sometimes cannot connect to customer sentiments and intentions.

**8. Applications**

The chatbot can be deployed on almost any website or application to resolve the customers’ queries and also the bot can be deployed on many social media platforms like Facebook, Slack, Telegram etc

**9. CONCLUSION**

By doing the above procedure and all we successfully created Intelligent helpdesk smart chatbot using Watson assistant, Watson discovery, Node-RED and cloud-functions.

**10. Future Scope**

1. Accuracy of smart document understanding can be increased by using some additional training and involvement of some websites can also be done using Watson Assistant learn skill
2. With use of text-to-speech and speech-to-text service the chatbot can be accessible handsfree.

**11. Bibliography**

**a) Source Code**

[

{

"id": "6421788e.1597b8",

"type": "tab",

"label": "Watson Dashboard",

"disabled": false,

"info": ""

},

{

"id": "6e63dcee.784f44",

"type": "function",

"z": "6421788e.1597b8",

"name": "",

"func": "msg.payload=msg.payload.input;\nreturn msg;",

"outputs": 1,

"noerr": 0,

"x": 357,

"y": 370.20001220703125,

"wires": [

[

"178dabf1.43ad14",

"4905008d.e9c8f"

]

],

"l": false

},

{

"id": "c8939de1.5f783",

"type": "function",

"z": "6421788e.1597b8",

"name": "bot",

"func": "var message = \"\";\n\nif(msg.payload.intents[0].intent == \"stack\_device\_operations\"){\n for(var i in msg.payload.context.webhook\_result\_1.passages){\n message = message + msg.payload.context.webhook\_result\_1.passages[i].passage\_text + \"\\n\";\n }\n \n msg.payload = message;\n}\n\nelse\nmsg.payload=msg.payload.output.text[0];\n\nreturn msg;",

"outputs": 1,

"noerr": 0,

"x": 575,

"y": 360,

"wires": [

[

"bba12c3a.e6dbc",

"e042557d.b56048"

]

],

"l": false

},

{

"id": "bba12c3a.e6dbc",

"type": "debug",

"z": "6421788e.1597b8",

"name": "",

"active": false,

"tosidebar": true,

"console": false,

"tostatus": false,

"complete": "true",

"targetType": "full",

"x": 672.600058555603,

"y": 439.20001888275146,

"wires": [],

"l": false

},

{

"id": "ca968d73.7d73a",

"type": "inject",

"z": "6421788e.1597b8",

"name": "",

"topic": "hi",

"payload": "how to setup wifi?",

"payloadType": "str",

"repeat": "",

"crontab": "",

"once": false,

"onceDelay": 0.1,

"x": 199.60000610351562,

"y": 256.4000015258789,

"wires": [

[

"6e63dcee.784f44"

]

],

"l": false

},

{

"id": "f7b18b0d.147448",

"type": "ui\_form",

"z": "6421788e.1597b8",

"name": "",

"label": "",

"group": "4ef4d1dc.cd5c1",

"order": 1,

"width": 16,

"height": 2,

"options": [

{

"label": "enter text here",

"value": "input",

"type": "text",

"required": true,

"rows": null

}

],

"formValue": {

"input": ""

},

"payload": "",

"submit": "submit",

"cancel": "cancel",

"topic": "",

"x": 196,

"y": 383.20001220703125,

"wires": [

[

"6e63dcee.784f44"

]

],

"l": false

},

{

"id": "1811ca4b.559366",

"type": "ui\_text",

"z": "6421788e.1597b8",

"order": 0,

"width": 0,

"height": 0,

"name": "",

"label": "text",

"format": "{{msg.payload}}",

"layout": "row-spread",

"x": -13,

"y": 397,

"wires": [],

"l": false

},

{

"id": "178dabf1.43ad14",

"type": "watson-conversation-v1",

"z": "6421788e.1597b8",

"name": "chatbot",

"workspaceid": "668f7c6b-7a8c-4e34-9f71-c757589e998e",

"multiuser": false,

"context": true,

"empty-payload": true,

"service-endpoint": "https://api.eu-gb.assistant.watson.cloud.ibm.com/instances/60d8ac52-2958-4f53-9bf0-cd2b2728dc08",

"timeout": "",

"optout-learning": false,

"x": 460,

"y": 280,

"wires": [

[

"c8939de1.5f783",

"f8f9a6a3.8455e8"

]

]

},

{

"id": "4905008d.e9c8f",

"type": "ui\_text",

"z": "6421788e.1597b8",

"group": "4ef4d1dc.cd5c1",

"order": 2,

"width": 16,

"height": 1,

"name": "User Dialouge",

"label": "You",

"format": "{{msg.payload}}",

"layout": "row-right",

"x": 430,

"y": 480,

"wires": []

},

{

"id": "e042557d.b56048",

"type": "ui\_text",

"z": "6421788e.1597b8",

"group": "4ef4d1dc.cd5c1",

"order": 4,

"width": 0,

"height": 0,

"name": "Chatbot Dialouge",

"label": "Chatbot",

"format": "{{msg.payload}}",

"layout": "row-left",

"x": 690,

"y": 220,

"wires": []

},

{

"id": "f8f9a6a3.8455e8",

"type": "debug",

"z": "6421788e.1597b8",

"name": "",

"active": true,

"tosidebar": true,

"console": false,

"tostatus": false,

"complete": "false",

"x": 420,

"y": 200,

"wires": []

},

{

"id": "4ef4d1dc.cd5c1",

"type": "ui\_group",

"z": "",

"name": "Default",

"tab": "566458fa.0d6548",

"order": 1,

"disp": true,

"width": "16",

"collapse": false

},

{

"id": "566458fa.0d6548",

"type": "ui\_tab",

"z": "",

"name": "Chatbot",

"icon": "dashboard",

"disabled": false,

"hidden": false

}

]

**b) References**

* 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-chatboton-nodered/
  4. http://www.iotgyan.com/learning-resource/integration-of-watson-assistant-tonode-red/