Project Report

Name:	Badal
Title:	Intelligent Customer Help Desk With Smart Document Understanding.
Category:	Artificial Intelligence.

Github Link:

https://github.com/manas01kumar/IISPS-INT-459-Intelligent-Customer-Help-Desk-with-Smart-Document-Understanding

Youtube Link:

https://www.youtube.com/watch?v=yPGmslBgBG0&t=237s

INDEX

Introduction

This project is use to create an AI chatbot for smart document understanding. This project is created with the help of IBM services like watson assistant and IBM watson discovery. For front-end we use node-red.

i) IBM assistant:

Watson Assistant is a conversation AI platform that helps you provide customers fast, straightforward and accurate answers to their questions, across any application, device or channel. By addressing common customer inquiries, Watson Assistant reduces the cost of customer interactions, helping your agents focus on complex use cases – not repetitive responses.

ii) IBM Discovery:

Watson Discovery is an award-winning enterprise search and AI search technology that breaks open data silos and retrieves specific answers to your questions while analyzing trends and relationships buried in enterprise data. Watson Discovery applies the latest breakthroughs in machine learning, including natural language processing capabilities, and is easily trained on the language of your domain. Unlike competitors, Watson Discovery can be deployed on any cloud or on-premises environment.

iii) Node-Red:

Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON.

Technical requirement:

- Artificial Intelligence,
- Machine Learning,
- Python,
- JSON,
- JavaScript,
- IBM watson services.

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 Scope

- 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

LITERATURE SURVEY

Existing problem

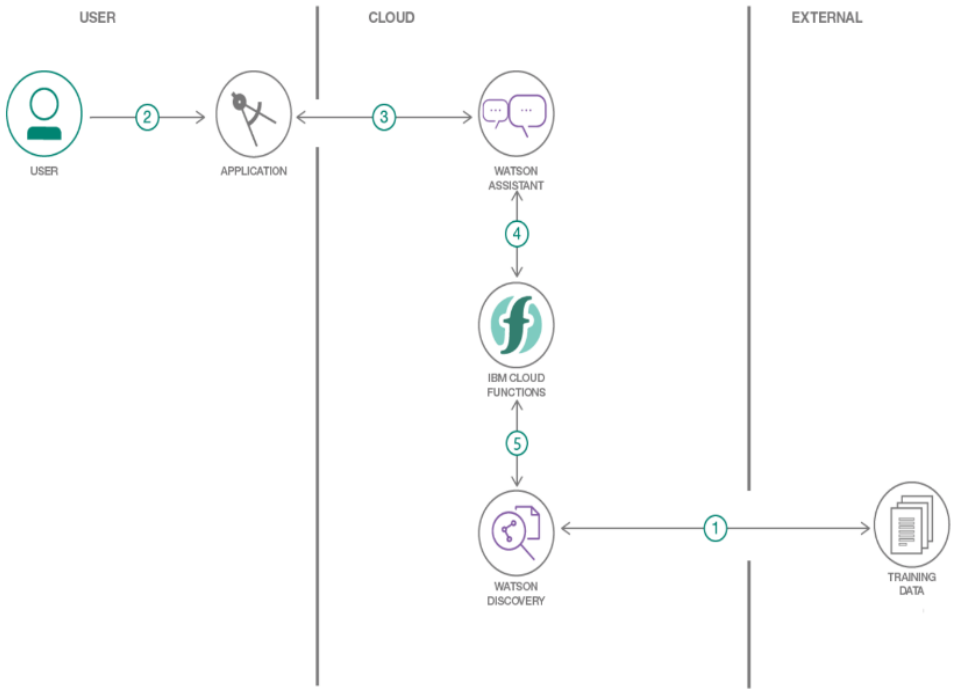
Basically, Chatbot are loaded with a certain set of questions that is more like if and else flow, the question or the user input which lies out of the scope of the chatbot is not answered and rather a message like “Try Again “is displayed so that it directs the customer to the customer agent. Generally, the chat bot needs to provide an efficient result with least traffic reaching and problems. So, to achieve certain problems we need SMART chat bot so that it can answer the queries of the customer.

Proposed Solution:

For the above-mentioned problem, we have to put a virtual agent in chatbot so it can understand the queries that are posted by customers. The virtual agent should be trained from some insight based on company backgrounds, working hours, store locations and product related information. In this project I used Watson Discovery to achieve the above solution.

Theoritical Analysis

1. Flow- Chart:



Step 1: The document in this step the owner's manual is indexed and understand by the smart document understanding feature of watson discovery.

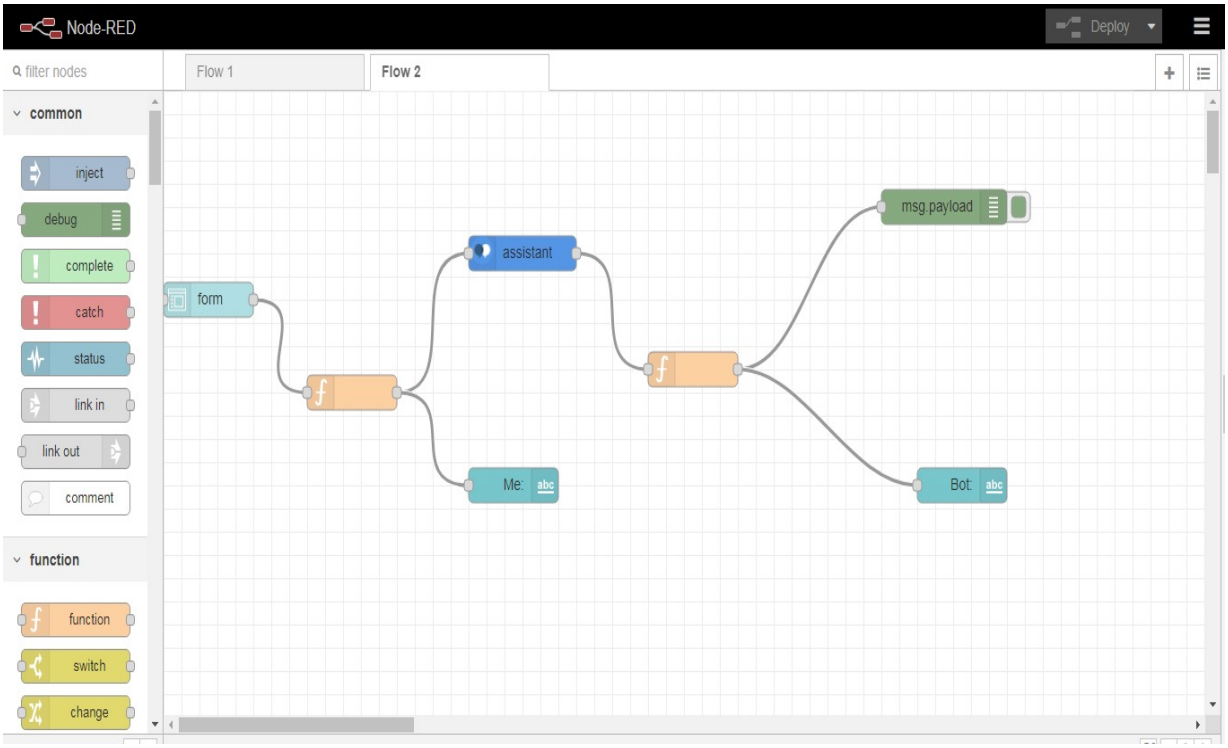
Step 2: In this step the user interacts with the user iterface/front-end which is created in Node-red.

Step 3: In step 3 it interacts with watson services.

Step 4: When the query comes, then assistant invokes the cloud function action

Step 5: It returns the result of an query as an output.

Node red flow chart :



It Contains:

a. Form

b. Function

c. Assistant

d. Output

Technical Specification :

i. Hardware Specification

Hardware : Pentium

RAM : 1GB and more

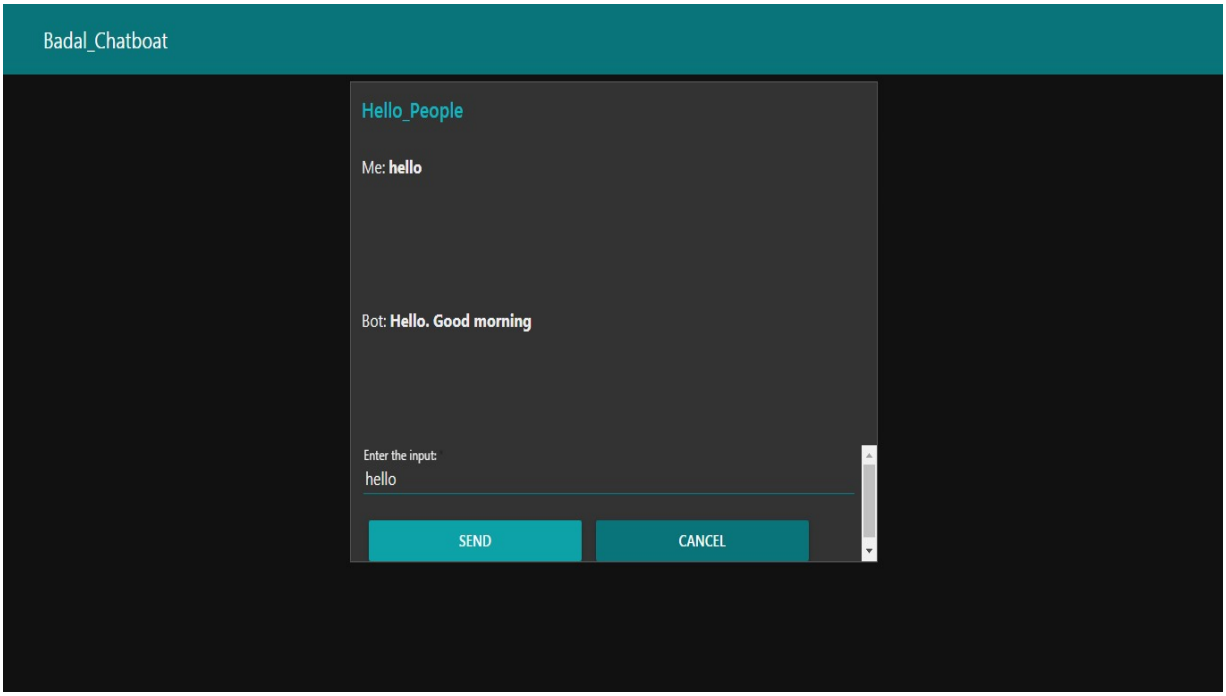
ii. Software Specification

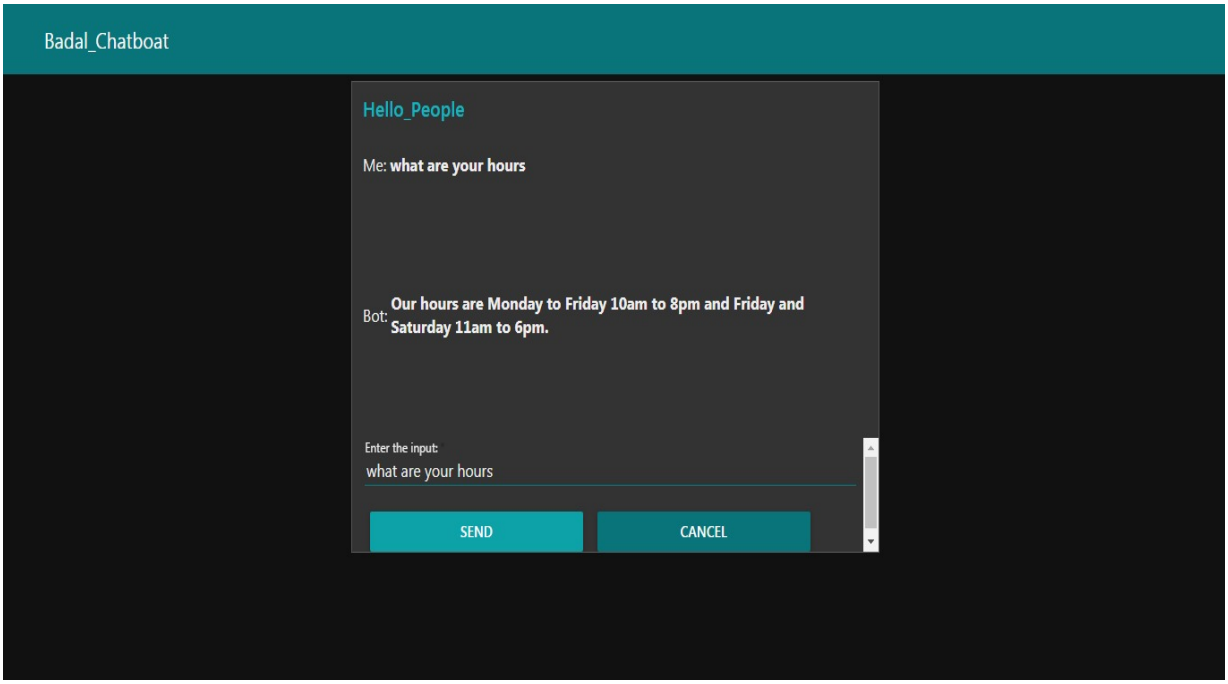
Operating System :	Windows, Linux
Technology :	Python, IBM Watson Discovery
User Interface tool :	Node Red Application
Assistant :	IBM Watson Assistant
Database :	IBM Cloud

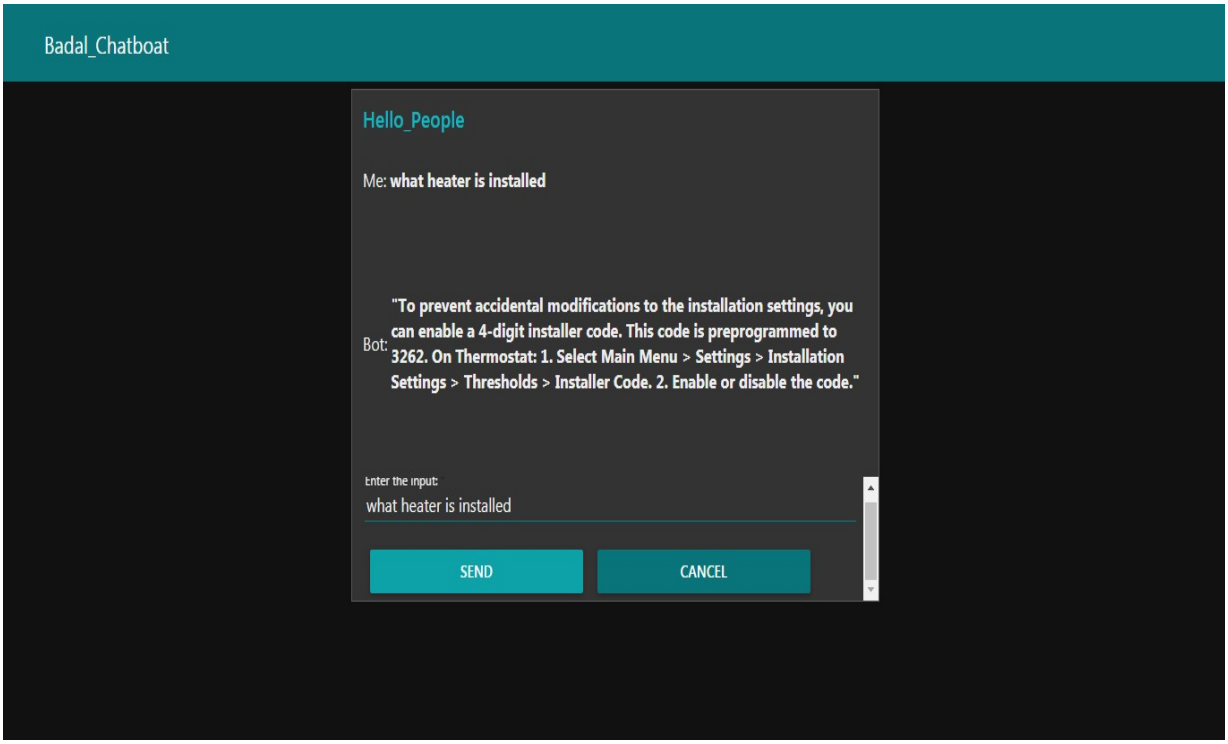
iii. Software Designing

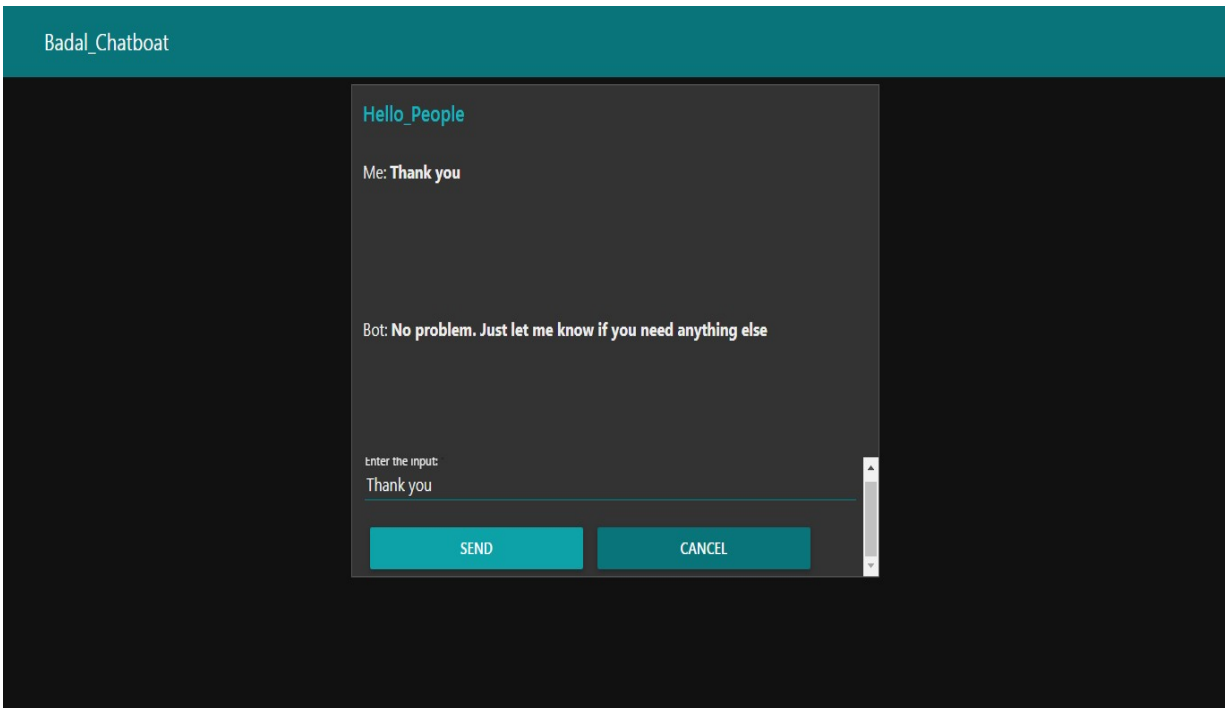
- Create an IBM Cloud services.
- Configure Watson Discovery.
- Create IBM Cloud Function action
- Configure Watson Assistant
- Create flow and configure node
- Deploy and run node red app

Experiments:






Badal_Chatboat
Hello_People
Me: what heater is installed
Bot: "To prevent accidental modifications to the installation settings, you can enable a 4-digit installer code. This code is preprogrammed to 3262. On Thermostat: 1. Select Main Menu > Settings > Installation Settings > Thresholds > Installer Code. 2. Enable or disable the code."
Enter the input:
what heater is installed
SEND
CANCEL


Badal_Chatboat
Hello_People
Me: Thank you
Bot: No problem. Just let me know if you need anything else
Enter the input:
Thank you
SEND
CANCEL

Results :

Web based User interface was developed by integrating all services on Node red.

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

Applications

This chatbot can be deployed to various websites as it can solve a lot of basic questions. It can be used to deploy as Customer Helpdesk for small scale products as their manual usually has the solution for the user's problems.

Conclusion

An Intelligent Customer Helpdesk Chatbot was created using various Watson services like Watson Discovery, Watson Assistant, Watson Cloud Functions and Node-RED.

Future Enhancement

In the future, various other Watson services like Text-To-Speech and Speech-To-Text can be integrated in the chatbot. This can make the chatbot Hands-free.

Bibliography

1. Node-RED Starter Application:
https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/
2. Build your own AI assistant :
https://www.youtube.com/watch?v=hitUOFNne14
3. How to use Watson Assistant with Webhooks :
https://www.youtube.com/embed/5z3i5IsBVnk
4. Watson Discovery :
https://developer.ibm.com/articles/introduction-watson-discovery/

Appendix

Source Code

Watson Cloud Function Code

```
/**
 *
 * @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);
  });
 });
}
```

Flow Json File


```
[{"id":"449b476c.8aa988","type":"tab","label":"Flow
4","disabled":false,"info":""},{"id":"5a56a0d.925846","type":"ui_form","z":"449b
476c.8aa988","name":"","label":"","group":"95b15e92.91d39","order":3,"width"
:0,"height":0,"options":[{"label":"Enter the
input","value":"input","type":"text","required":true,"rows":null}],"formValue":{"in
```

put":""},"payload":"","submit":"submit","cancel":"cancel","topic":"","x":90,"y":280
,"wires":[["17437ec7.ca2dd1"]]},{"id":"17437ec7.ca2dd1","type":"function","z"
:"449b476c.8aa988","name":"","func":"msg.payload =
msg.payload.input;\nreturn
msg;","outputs":1,"noerr":0,"x":250,"y":180,"wires":[["1a1e3ab0.106155","8bfa
ad3c.56dd2"]]},{"id":"1a1e3ab0.106155","type":"watson-conversation-v1","z"
:"449b476c.8aa988","name":"AssistantBot","workspaceid":"d11e07bf-68ce-
4116-aa78-3488296f0e0a","multiuser":false,"context":true,"empty-payload":
false,"service-endpoint":"https://api.eu-gb.assistant.watson.cloud.ibm.co
m/instances/5ac74d84-bd69-43a2-a188-3db046b6d238","timeout":"","opto
ut-learning":false,"x":430,"y":100,"wires":[["c87d7d3e.a9987","bd1e1474.480
068"]]},{"id":"c87d7d3e.a9987","type":"debug","z":"449b476c.8aa988","name
":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"f
alse","x":670,"y":60,"wires":[]},{"id":"bd1e1474.480068","type":"function","z":"4
49b476c.8aa988","name":"","func":"msg.payload.text =
\"\";\nif(msg.payload.context.webhook_result_1){\n   for(var i in
msg.payload.context.webhook_result_1.results){\n   msg.payload.text
= msg.payload.text+\"\\n\"+
msg.payload.context.webhook_result_1.results[i].text;\n   }\n
msg.payload = msg.payload.text;\n   \n}\nelse{\n   msg.payload =
msg.payload.output.text[0];\n}\nreturn
msg;","outputs":1,"noerr":0,"x":560,"y":240,"wires":[["f413de95.dd8d1"]]},{"id":
"f413de95.dd8d1","type":"ui_text","z":"449b476c.8aa988","group":"95b15e9
2.91d39","order":4,"width":0,"height":0,"name":"","label":"text","format":"{{msg
.payload}}","layout":"row-spread","x":720,"y":320,"wires":[]},{"id":"8bfaad3c.56
dd2","type":"ui_text","z":"449b476c.8aa988","group":"95b15e92.91d39","orde

r":2,"width":0,"height":0,"name":"","label":"input","format":"{{msg.payload}}","layout":"row-spread","x":350,"y":320,"wires":[]},{"id":"95b15e92.91d39","type":"ui_group","z":"","name":"GroupsDetails","tab":"463675ef.0eee1c","order":1,"disp":true,"width":"6","collapse":false},{"id":"463675ef.0eee1c","type":"ui_tab","z":"","name":"MyAssistant","icon":"dashboard","disabled":false,"hidden":false}]