### **Smart Home Assistant With IBM Cloud**

**INTRODUCTION**

**a)OVERVIEW**

The Smart home assistant is your personal assistant which performs the tasks of a person. It acts according to the person's commands.

**b)PURPOSE**

A home assistant helps the user in controlling appliances in the home easily by giving voice commands. This gives the temperature and humidity of the home and also we can set reminders. Users can also send messages to the desired person just by giving voice inputs.

**2 LITERATURE SURVEY**

**a)EXISTING PROBLEM**

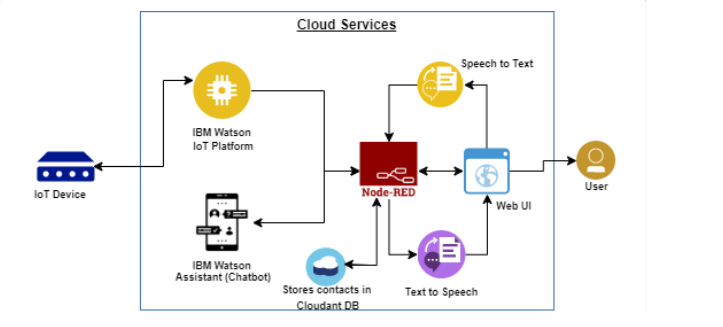
It is inconvenient to use different devices to check the temperature, humidity, set reminders, and send messages

**b)PROPOSED SOLUTION**

The smart home assistant can perform all the required functions from informing about the temperature, humidity,set reminders and send messages

**3 THEORETICAL ANALYSIS**

1. **BLOCK DIAGRAM**

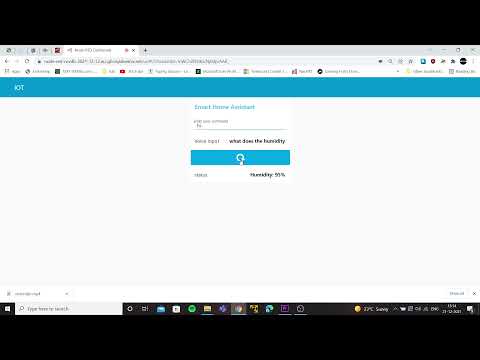


1. **Software designing**

* Developed a python code which sends the temperature and humidity values and switches on/off the appliances by subscribing to the commands from the IBM Cloud platform. We configured IBM cloud services, which acts as the backbone of our project.
* In order to create a Web UI for our project, we used and configured the Node-Red services. API keys are used to interconnect between Node-Red and IBM IOT platform.
* IBM Watson Assistant is used to create skills for the assistant by using intents, entities and dialogs.
* IBM Cloudant Database is used to store numbers w.r.t names.
* Voice commands are converted to text. The converted text is sent to Watson Assistant. The reply from the Assistant is then converted to voice commands using Text to Speech API.
* Commands are sent from the Web UI to control the appliances such as -
  + Switch On/Off Fans & Lights.
  + Gets temperature & Humidity values from IBM IoT Platform.
  + To send a message to another person.
  + Set reminders.

**EXPERIMENTAL INVESTIGATIONS**

The video in the link below shows the experiments done with the application



<https://youtu.be/vkXzpFDf_2c>

ADVANTAGES

* Control and Monitoring.
* The IoT significantly improves the way you can control and monitor all the processes taking place at your home
* Cost and Energy Savings.
* Environment Impact.
* Better Security.
* Comfort.

DISADVANTAGES

* Significant installation costs.
* Reliable internet connection is crucial.
* Security issues.
* Technological problems in connected homes.
* You may lock yourself out of your own house.
* Helplessness if technology fails.
* Some people may not like smart technologies.
* Maintenance and repair issues.

APPLICATIONS

Since we came clear about smart home technology, let us go through some of the smart home devices that make a perfect connected home.

* **Smart security locks**– Smart locks offer a variety of features including door and window sensors, motion detectors, video cameras, and recording mechanism. Accessible via cloud, users get real-time information on the security status of the home.
* **Smart thermostat**– Smart thermostats typically allow users to schedule their desirable temperature settings and incorporate it into the home automation systems. With the smart device, users can monitor their HVAC from wherever they are.
* **Smart Health Sensors**: Having smart sensors throughout the home that will keep tabs of resident’s health and look for signs of illness. Further in case of an emergency, it can remotely connect with medical practitioners and alert caretakers.

While connecting the smart home appliances to a smartphone or any other connected control unit is a huge convenience, there are many other advantages of smart home technology. It helps people have a greater awareness of where their resources are being used and for what.

CONCLUSION

The IoT device market has undergone radical changes in only a few short years. Starting with disparate devices and no ecosystems to speak of, the market has now grown to encompass enterprise players working together to create ecosystems, tailored for mobile technology, which allows IoT devices to become interconnected.

Automation of the home may have once seemed like a peculiar and unlikely concept, but as our devices become smarter and more investment is poured into the development of ioT consumer products, we are likely to see increased competition spur on further innovation in the field.

FUTURE SCOPE

Future scope for the home automation systems involves making homes even smarter.

Homes can be interfaced with sensors including motion sensors, light sensors and temperature sensors and provide automated

toggling of devices based on conditions.

That is exactly what sums up the smart home as it can be referred to as a residence

that uses internet-connected devices to enable the remote monitoring and management of appliances and systems like lighting and heating.

BIBLIOGRAPHY

<https://dzone.com/articles/iot-smart-home-automation-and-its-future-predictio>

<https://intuitive-design.foleon.com/knect365/smart-home/conclusion/>

<http://smarthomes-technology.blogspot.com/p/conclusion-and-additional-work.html>

<https://hkrtrainings.com/future-scope-of-iot>

APPENDIX

a)Source code

import wiotp.sdk.device

import os

import time

import datetime

import random

myConfig = {

"identity": {

"orgId": "e9psgv",

"typeId": "RIFT",

"deviceId": "2001"

},

"auth": {

"token": "12345678"

}

}

def myCommandCallback(cmd):

print("Message recieved from IBM IoT Platform: %s" % cmd.data)

m=cmd.data['command']

if (m == "lighton"):

print("Light is Switched ON")

elif (m== "lightoff"):

print("Light is Switched OFF")

elif (m== "fanoff"):

print("Fan is Switched OFF")

elif (m== "fanon"):

print("Fan is Switched OFF")

print(" ")

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

client.connect()

while True:

temp=random.randint(-20,125)

hum=random.randint(0,100)

myData={'temperature':temp, 'humidity': hum}

client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0 , onPublish=None)

print("Published Data Successfully: %s",myData)

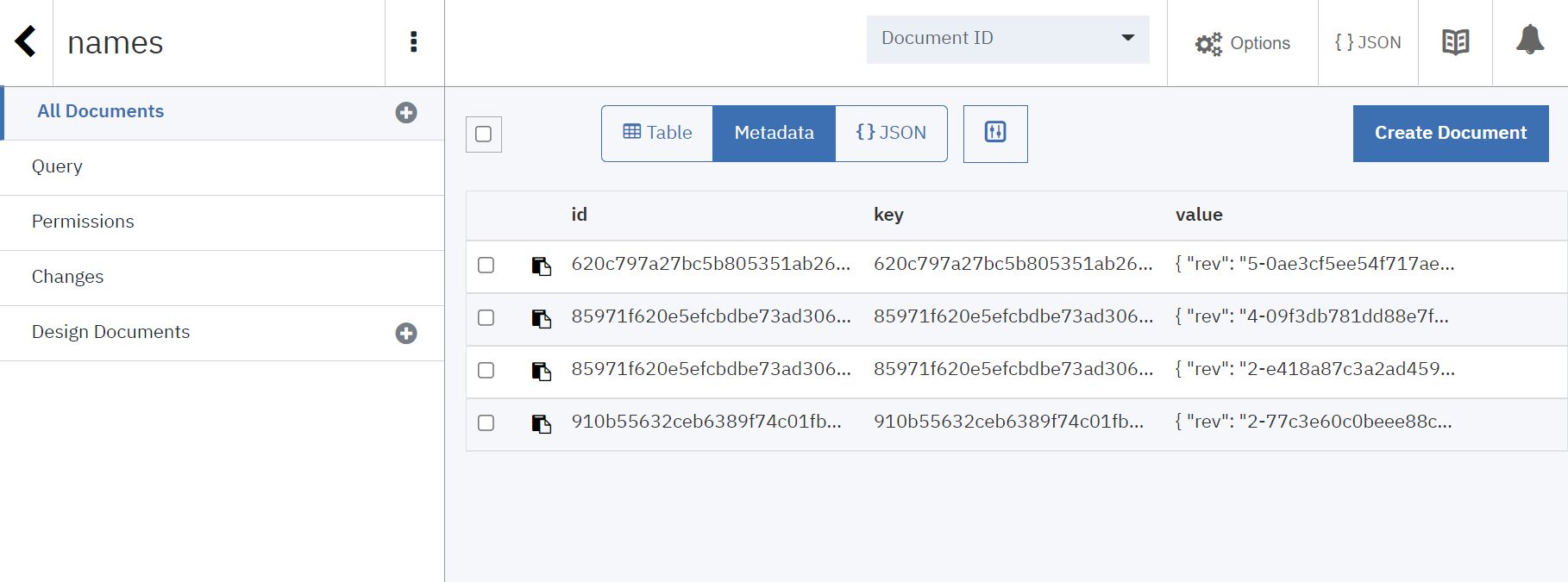
client.commandCallback = myCommandCallback

time.sleep(2)

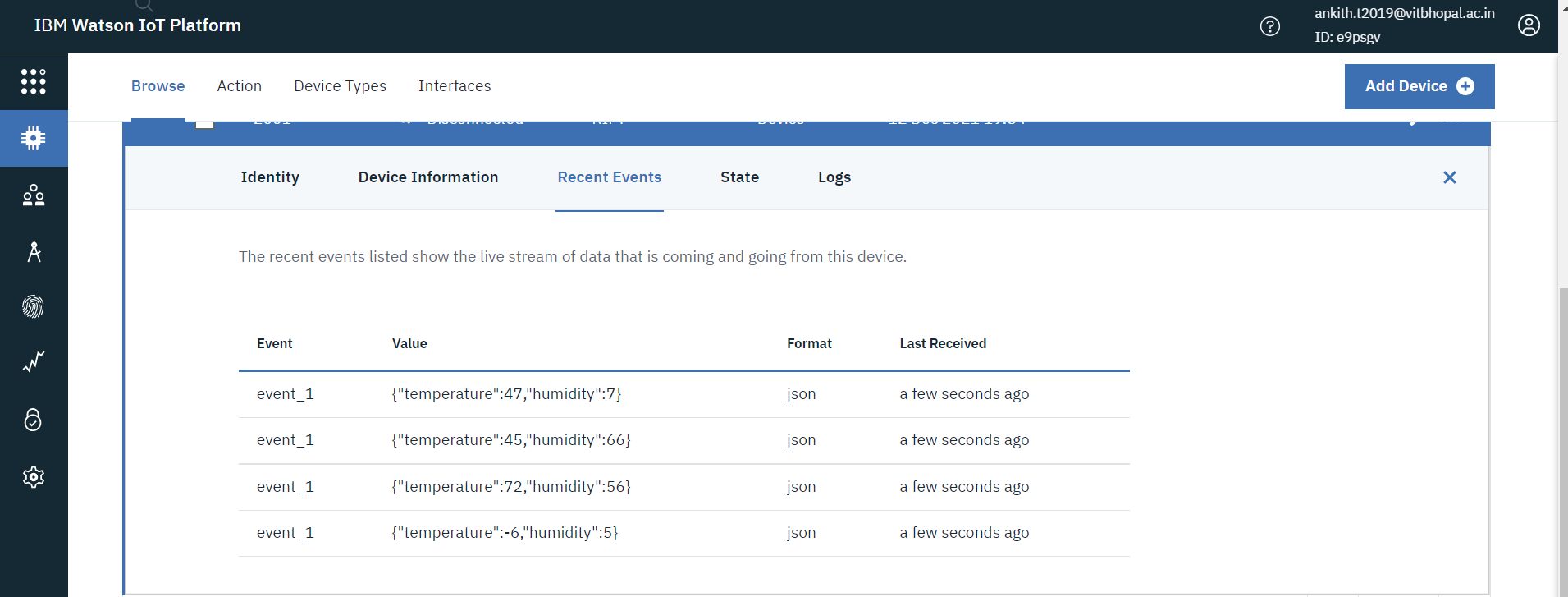
client.disconnect()

SCREENSHOT

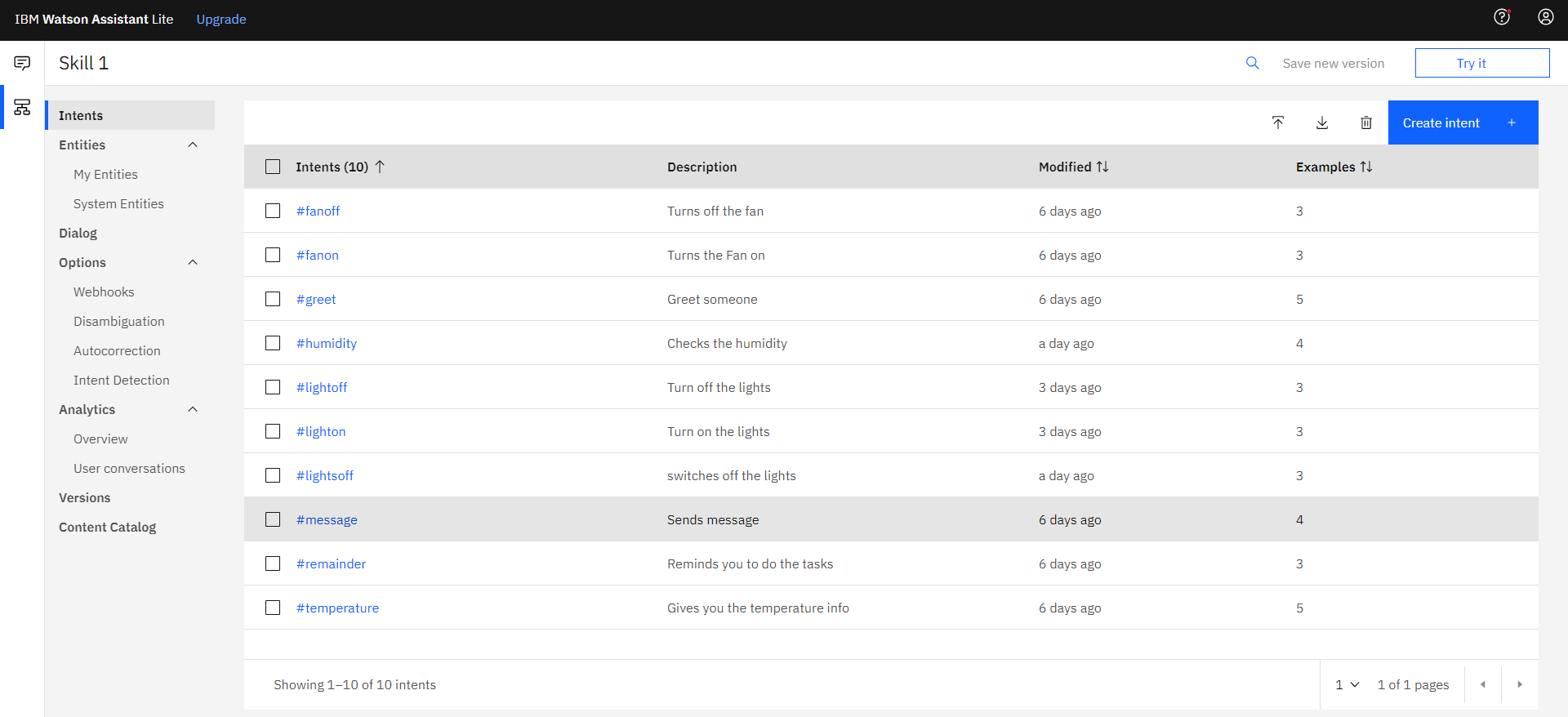
IBM CLOUDANT

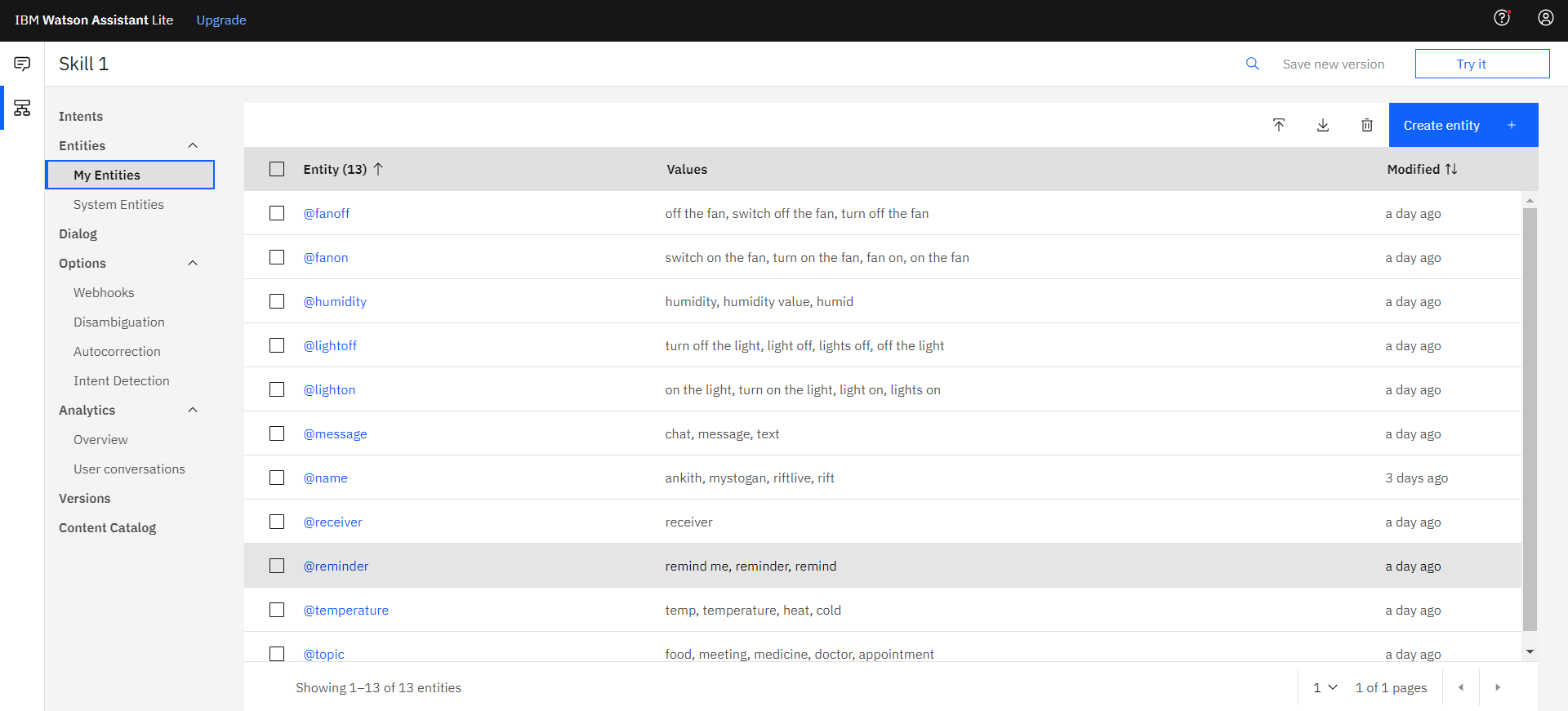


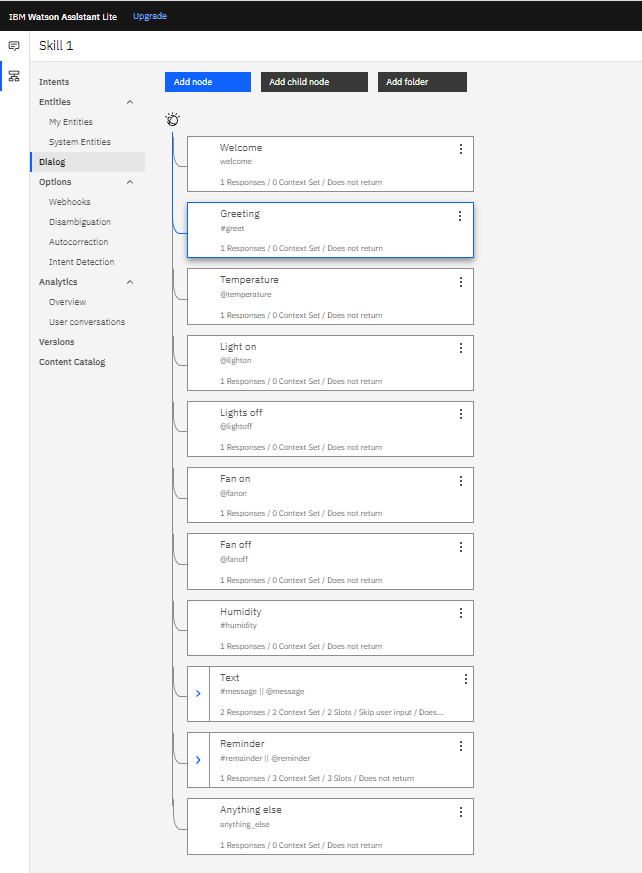
IBM IOT PLATFORM



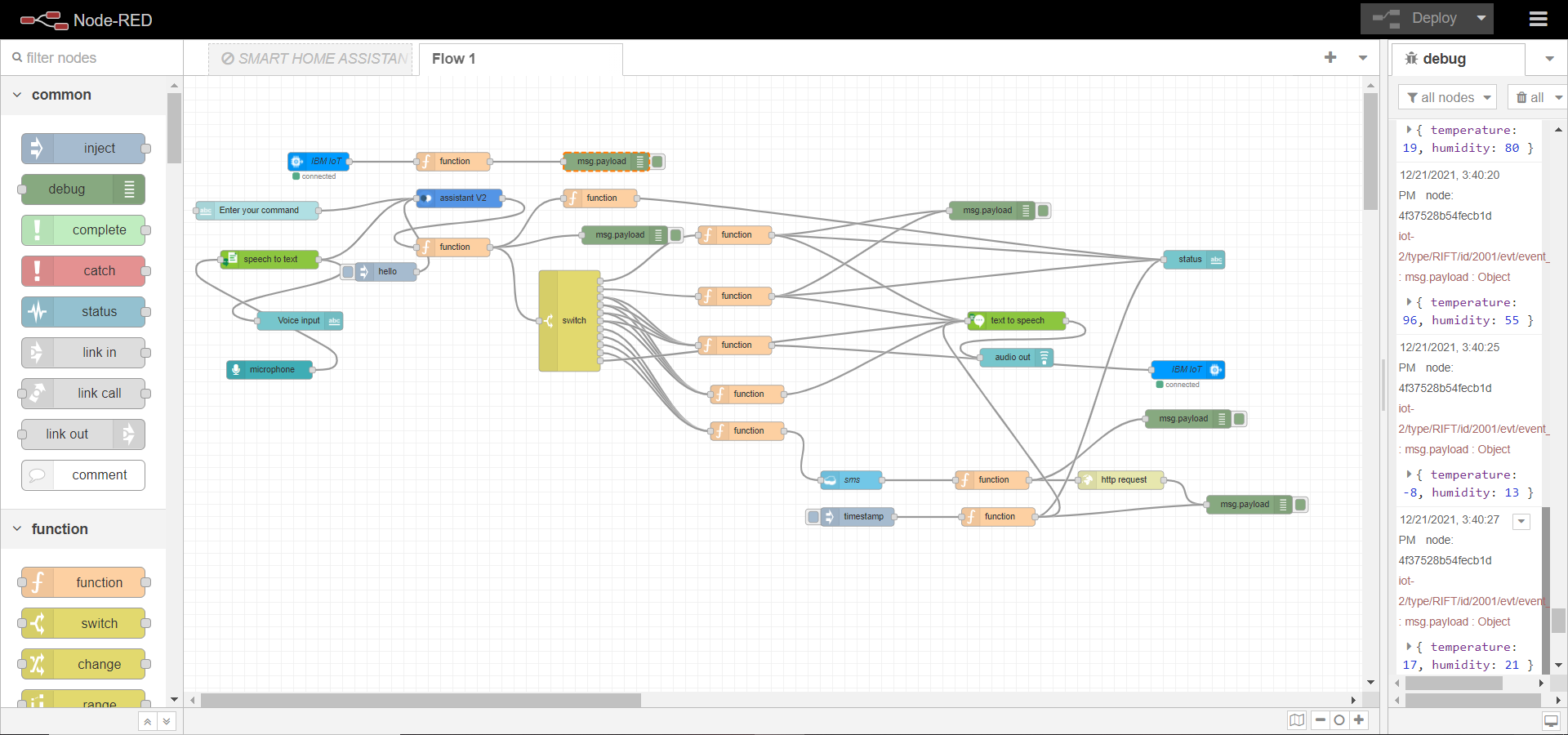
IBM WATSON SKILLS



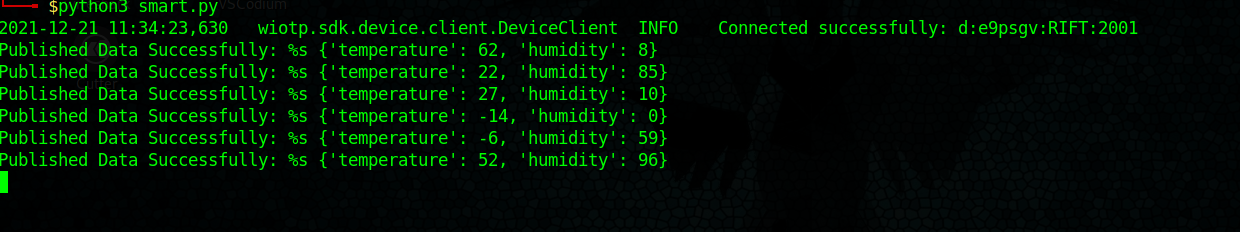




NODE RED FLOW



PYTHON EXECUTION



UI

