

Safety Gadget for Child Safety Monitoring and Notification

IBM NALAIYATHIRAN

Project Development –Delivery of Sprint 2

Creating Node –Red service and connect with IBM cloud and Web UI

TITLE	IOT based child safety gadget for child safety monitoring and notification
DOMAIN NAME	INTERNET OF THINGS
TEAM ID	PNT2022TMID37747
TEAM LEADERNAME	PRIYADHARSHINI N
TEAM MEMBER NAME	DURGA P JANANI P PRABADEVI E

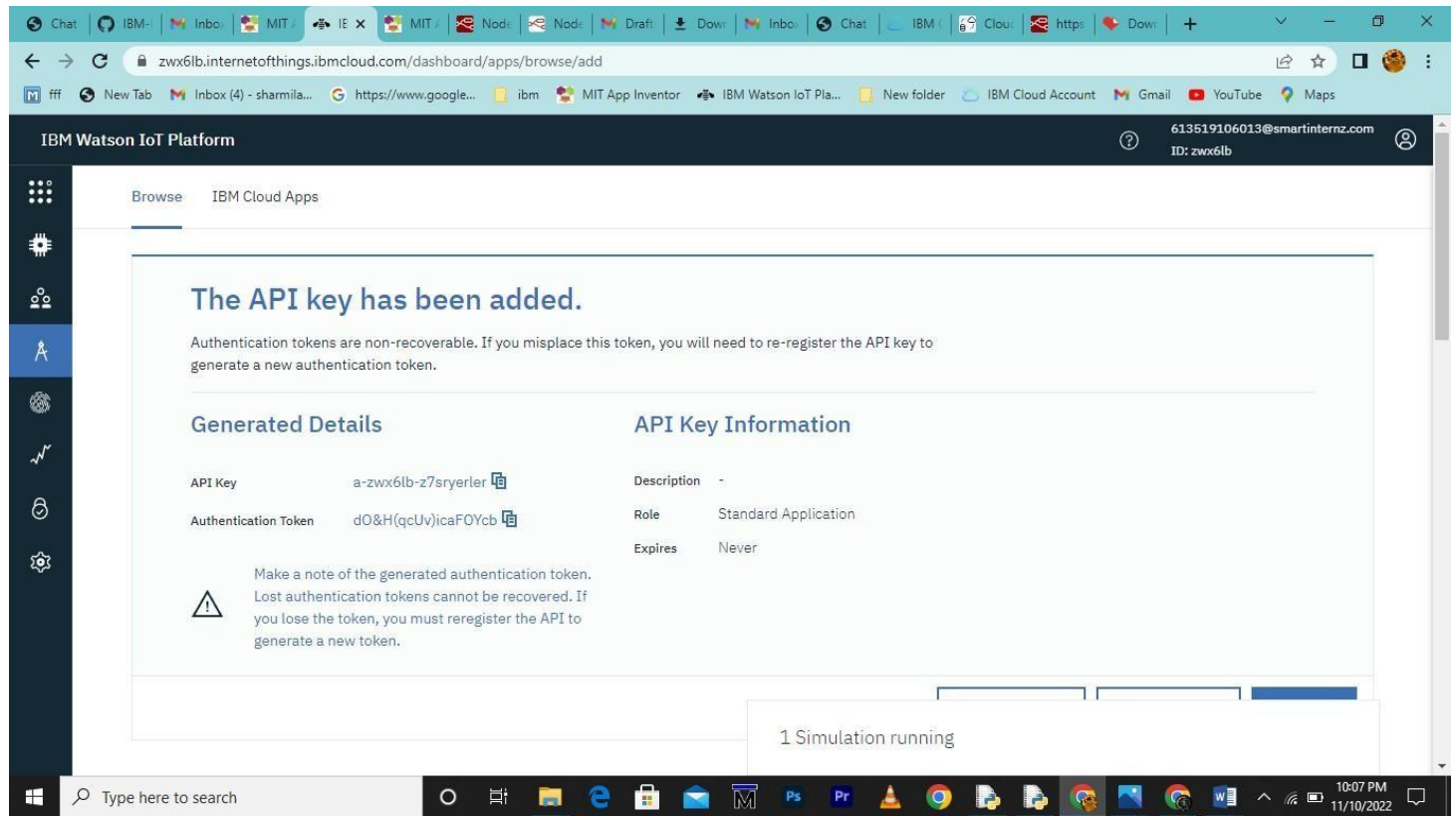
Creating Node-Red service:

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow named 'Flow 1'. The flow starts with an 'IBM IoT' node (connected) and a 'worldmap' node (connected 0). The 'IBM IoT' node is connected to several function nodes: 'function', 'temp', 'temp_status', 'latitude', 'your_child_zone', and 'longitude'. These function nodes are then connected to various output nodes: 'Child name abc', 'gauge', 'Temperature abc', 'temp_status abc', 'Latitude abc', 'Child Zone abc', 'Longitude abc', and 'Longitude' (with a checkmark icon). The 'worldmap' node is also connected to a 'Longitude' node (with a checkmark icon). The right sidebar shows the 'debug' console with a list of messages. The messages are JSON objects containing sensor data, including temperature, latitude, longitude, and child zone information. The messages are timestamped and include a node ID.

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow named 'Flow 1'. The flow starts with a '[get] /sensor' node connected to a function node, which is then connected to an 'http' node. The function node is also connected to several other function nodes: 'latitude', 'your_child_zone', and 'longitude'. These function nodes are then connected to various output nodes: 'Latitude abc', 'Child Zone abc', 'Longitude abc', and 'Longitude' (with a checkmark icon). The 'worldmap' node is also connected to a 'Longitude' node (with a checkmark icon). The right sidebar shows the 'debug' console with a list of messages. The messages are JSON objects containing sensor data, including temperature, latitude, longitude, and child zone information. The messages are timestamped and include a node ID.

Connecting with IBM Cloud:

Using IBM IOT node through API key

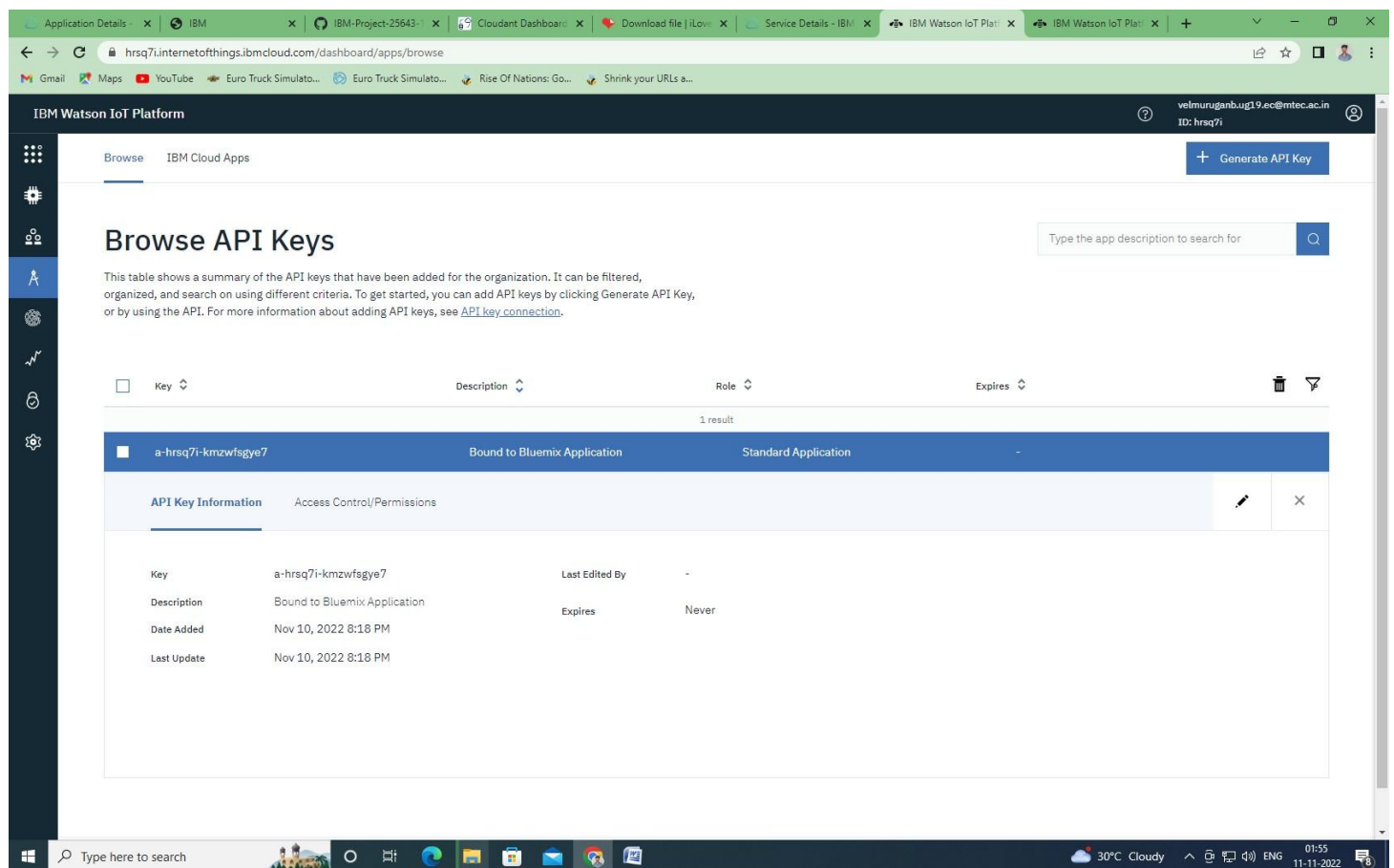


The screenshot shows the IBM Watson IoT Platform dashboard. The main heading is "The API key has been added." Below this, a message states: "Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the API key to generate a new authentication token." The dashboard is divided into two sections: "Generated Details" and "API Key Information".

Generated Details		API Key Information	
API Key	a-zwx6lb-z7sryerler	Description	-
Authentication Token	dO&H(qcUv)icaFOYcb	Role	Standard Application
		Expires	Never

Make a note of the generated authentication token. Lost authentication tokens cannot be recovered. If you lose the token, you must reregister the API to generate a new token.

1 Simulation running



The screenshot shows the IBM Watson IoT Platform dashboard with the "Browse API Keys" section. A "Generate API Key" button is visible in the top right. The main heading is "Browse API Keys". Below this, a message states: "This table shows a summary of the API keys that have been added for the organization. It can be filtered, organized, and search on using different criteria. To get started, you can add API keys by clicking Generate API Key, or by using the API. For more information about adding API keys, see [API key connection](#)."

1 result

Key	Description	Role	Expires
a-hrsq7i-kmzwfsgye7	Bound to Bluemix Application	Standard Application	-

API Key Information Access Control/Permissions

Key	Description	Last Edited By	Expires
a-hrsq7i-kmzwfsgye7	Bound to Bluemix Application	-	Never
Date Added	Nov 10, 2022 8:18 PM		
Last Update	Nov 10, 2022 8:18 PM		

Transferring values from Python Code:

```
Child Safety device.py - C:/Users/kutta/Desktop/IBM-Dr/Child Safety device.py (3.7.4)
File Edit Format Run Options Window Help

import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "illzal"
deviceType = "latlonstem"
deviceId = "613510"
authMethod = "token"
authToken = "1092837465"
#api key (a-illzal-mbdxqo6z0s)
#api token (zSYzISuAWFxF_7GkT)

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod}
    deviceCli = ibmiotf.device.Client(deviceOptions)
    #.....
except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type
print("power on ")
print("checking connection to waston iot...")
time.sleep(2)
deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")
print("i can provide your children live location and temperature ")
print()
name=str(input("enter your child name:"))
while True:

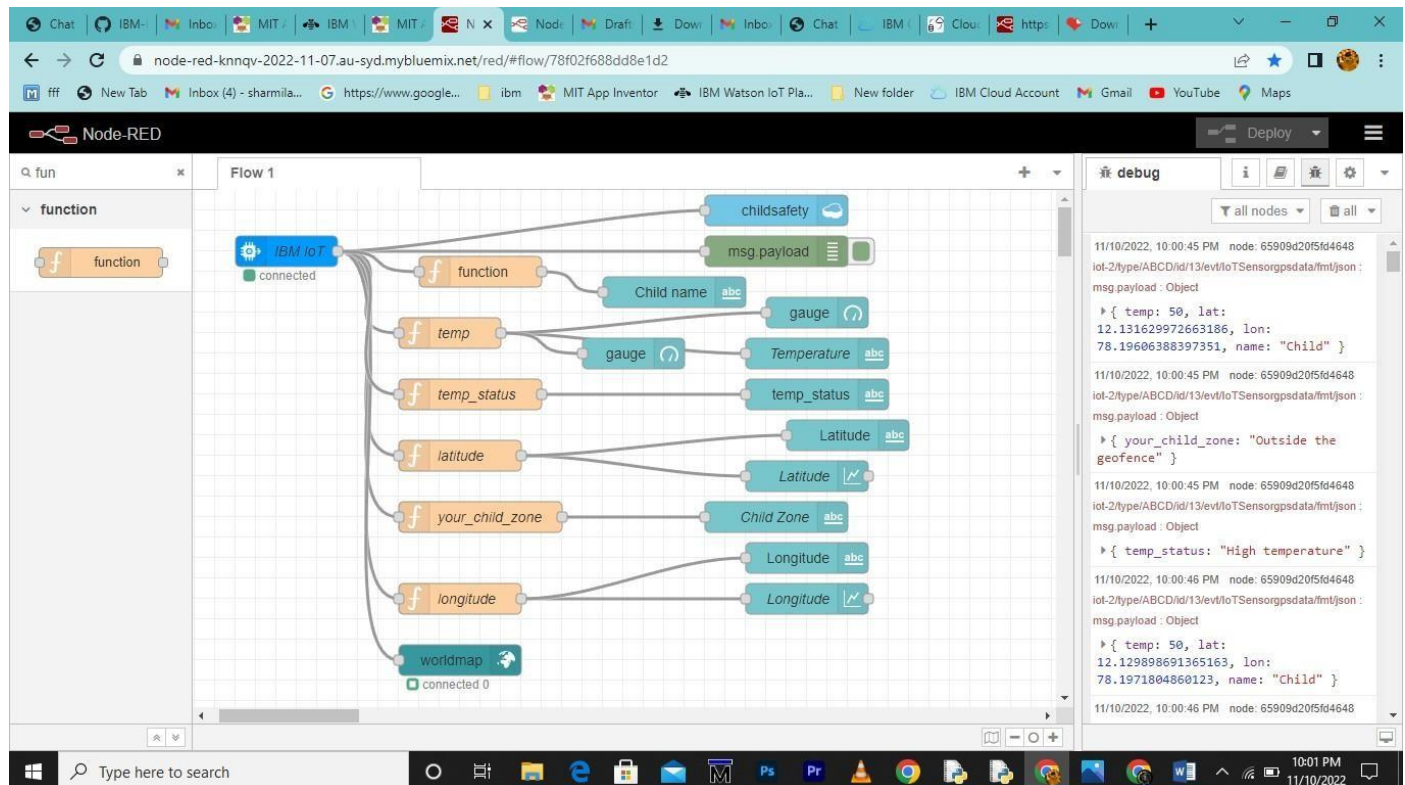
    temperature=random.randint(20,50)#random temperature for your child
    latitude=random.uniform(10.781377,10.78643)#random latitude for your child
    longitude=random.uniform(79.129113,79.134014)#random longitude for your child

    data = { 'temp' : temperature, 'lat' : latitude, 'lon':longitude, 'name':name }
    #print data
    def myOnPublishCallback():
        print("Published Temperature = %s C & Latitude = %s & Longitude = %s" % (temperature, latitude, longitude))

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/kutta/Desktop/IBM-Dr/Child Safety device.py =====
power on
checking connection to waston iot...
2022-11-10 22:14:21,799 ibmiotf.device.Client INFO Connected success
lly: d:illzal:latlonstem:613510
dear user ... welcome to IBM-IOT
i can provide your children live location and temperature

enter your child name:child
Published Temperature = 39 C latitude = 10.782749628132827 & longitude = 79.867253162 & to IBM Watson
Published Temperature = 39 C latitude = 10.782669248109656 & longitude = 79.1255540076 & to IBM Watson
Published Temperature = 43 C latitude = 10.781765104656792 & longitude = 79.077964707 & to IBM Watson
Published Temperature = 30 C latitude = 10.786083936690018 & longitude = 79.2366715787 & to IBM Watson
Published Temperature = 31 C latitude = 10.784810558975826 & longitude = 79.0117359415 & to IBM Watson
Published Temperature = 45 C latitude = 10.785949922923024 & longitude = 79.5563867668 & to IBM Watson
Published Temperature = 24 C latitude = 10.784168891438233 & longitude = 79.9528906442 & to IBM Watson
Published Temperature = 23 C latitude = 10.786248060883958 & longitude = 79.4368596464 & to IBM Watson
Published Temperature = 27 C latitude = 10.783808327214418 & longitude = 79.951933729 & to IBM Watson
Published Temperature = 43 C latitude = 10.786340416981865 & longitude = 79.7748803969 & to IBM Watson
Published Temperature = 49 C latitude = 10.786208956579015 & longitude = 79.2192551409 & to IBM Watson
Published Temperature = 45 C latitude = 10.783690544907325 & longitude = 79.504415061 & to IBM Watson
```

Node-Red:



Node-Red Dashboard:

