SMART KITCHEN USING IBM CLOUD

Category: Internet Of Things

by:-

1.Athish vp (athishvp1999@gmail.com)
2.Vedika Rashmi (vedikarashmi@gmail.com)
3.B.Sai JIthendra Reddy (banajithendra2@gmail.com)
4.Sucharita naha (sucharitanaha@gmail.com)

1.INTRODUCTION

The global smart kitchen appliances market size was valued at USD 9.87 billion in 2019 and is anticipated to expand at a CAGR of 19.1% over the forecast period. Rising consumer disposable income and increasing technological trends such Internet of Things enabled devices are spurring the market growth. Increasing adoption of smart kitchen appliances in residential as well as commercial sectors across the world is also one of the prominent factors driving the market growth.

2.Project description:-

In this project we have created a model for a smart kitchen
The main features of this project are
1. Measuring the cylinder weight
By measuring the cylinder weight we can identify the status of the cylinder, is the amount of lpg gas pressent in the cylinder is high ,medium,and low . And if the cylinder becomes empty we will get an sms message stating that 'The cylinder is empty'.
2. Detecting the gas leakage in the cylinder
Once the gas leakage is detected we will get an sms stating that 'There is a gas leakage in the cylinder' .
3.Measuring the jar weight(smart jar)
By measuring the jar weight we can identify whether the jar is full ,medium,or low.And if the jar is empty we will get an sms stating that 'The jar is empty'.

3. Features:

1.We can replace all the regular storage jars with the smart jars, which sends an alert when the jar gets empty or the measured sensor value is below the threshold.

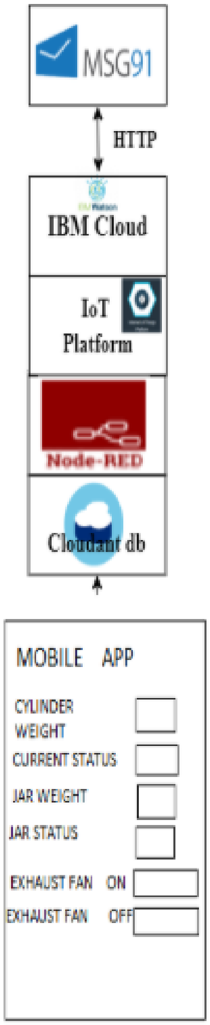
2.These jars communicate with the controller through Nrf communication.

3.The cylinder is attached with a leakage sensor that detects the leakage from the cylinder and sends a notification if any leakage is detected.

4.If any leakage is detected the exhaust fans are automatically switched ON.

5.Cylinder weight is also measured and sends an alert when it is empty, based on the empty cylinder weight.
6.All these parameters can be monitored by both Mobile App and Web App.

4.BLOCK DAIGRAM:-



5.THE PYTHON IDLE CODE-

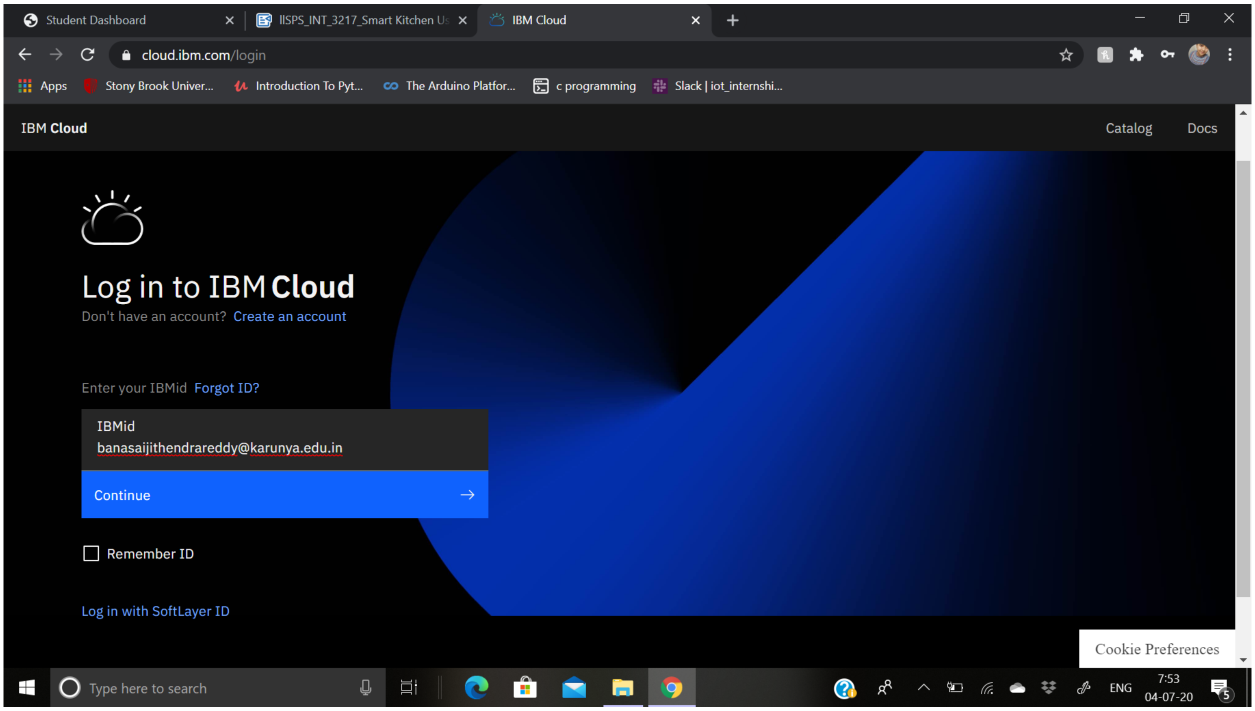
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import requests
from cloudant.client import Cloudant
from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
url = "https://www.fast2sms.com/dev/bulk"
#Provide your IBM Watson Device Credentials
organization = "kg0p6r"
deviceType = "raspberrypi"
deviceId = "project-15"
authMethod = "token"
authToken = "123456789"
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data)#Commands
    if cmd.data['command']=='exhauston':
        print("EXHAUST ON IS RECEIVED")
    elif cmd.data['command']=='exhaustoff':
        print("EXHAUST OFF IS RECEIVED
try:
deviceOptions = {"org": organization, "type": deviceType, "id":

deviceId, "auth-method": authMethod, "auth-token": authToken}
      deviceCli = ibmiotf.device.Client(deviceOptions)
      except Exception as e:
      print("Caught exception connecting device: %s" % str(e))
      sys.exit()
deviceCli.connect()
cylinder_weight=10
jar_weight=1000
cyl_empty=0
jar_empty=0
while True:
    cylinder_weight=cylinder_weight-1
    jar_weight=jar_weight-14
    if(cylinder_weight>0 and cylinder_weight<=3):
        current_status="low"
    elif(cylinder_weight>3 and cylinder_weight<=7):
        current_status="medium"
    elif(cylinder_weight>7 and cylinder_weight<=10):
        current_status="high"
    else:
        cylinder_weight=0
        status="empty"
        if(cyl_empty==0):
            querystring =
{"authorization":"CEBM7ZYzfkWlyPrtnmJNoH3xAGFLjhsX06ceg5UuiqQp2dS4RVMpPyG
0vdxQOq5N1ktTiLhrERac6wWU","sender_id":"FSTSMS","message":"The cylinder is
empty","language":"english","route":"p","numbers":"7541057457,9952162247,7530015671
,9500377352"}
            headers = {'cache-control': "no-cache"}
            response = requests.request("GET", url, headers=headers,
params=querystring)
         print(response.text)
            cyl_empty=1
    if(jar_weight>0 and jar_weight<=300):
        jar_status="low"
    elif(jar_weight>300 and jar_weight<=700):
        jar_status="medium"
    elif(jar_weight>700 and jar_weight<=1000):
        jar_status="high"
    else:
        jar_weight=0
        jar_status="empty"
        if(jar_empty==0):
            querystring =
{"authorization":"CEBM7ZYzfkWlyPrtnmJNoH3xAGFLjhsX06ceg5UuiqQp2dS4RVMpPyG
0vdxQOq5N1ktTiLhrERac6wWU","sender_id":"FSTSMS","message":"The jar is
empty","language":"english","route":"p","numbers":"7541057457,9952162247,7530015671
,9500377352"}
            headers = {'cache-control': "no-cache"}
            response = requests.request("GET", url, headers=headers,
params=querystring)
            print(response.text)
            jar_empty=1
    gasleak=0
    gasleak=gasleak+1
    if(gasleak==40):
        print('gas leak is detected')
        querystring =
{"authorization":"CEBM7ZYzfkWlyPrtnmJNoH3xAGFLjhsX06ceg5UuiqQp2dS4RVMpPyG
0vdxQOq5N1ktTiLhrERac6wWU","sender_id":"FSTSMS","message":"There is a gas
leak","language":"english","route":"p","numbers":"7541057457,9952162247,7530015671,9
500377352"}
        headers = {'cache-control': "no-cache"}
        response = requests.request("GET", url, headers=headers, params=querystring)
        print(response.text)
        jar_empty=1


 else:
        print('no gas leak')

    data =
{'cylinder_weight':cylinder_weight,'jar_weight':jar_weight,'current_status':current_status,'ja
r_status':jar_status}
    #print (data)
    def myOnPublishCallback():
      print ("Published cylinder_weight = %s " % cylinder_weight, "jar_weight = %s " %
jar_weight ,"current_status = %s" % current_status,"jar_status = %s" % jar_status, "to IBM
Watson")
    success = deviceCli.publishEvent("smart_kitchen", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
      print("Not connected to IoTF")
    time.sleep(2)
    deviceCli.commandCallback = myCommandCallback
    client = Cloudant("fcf055c3-9b00-418e-a955-a88fd70d32d0-bluemix", "a10a022499db0fcd47dce369b941bb895d640c859923fae1fd2806ea5cbe4721",url="https://fcf055c3-9b00-418e-a955-a88fd70d32d0-bluemix:a10a022499db0fcd47dce369b941bb895d640c859923fae1fd2806ea5cbe4721@fcf055c3-9b00-418e-a955-a88fd70d32d0-bluemix.cloudantnosqldb.appdomain.cloud")
    client.connect()
    database_name = "project15"
    my_database = client.create_database(database_name)
    if my_database.exists():
        print(f"'{database_name}' successfully created.")

    record_data={'cylinder_weight':cylinder_weight,'jar_weight':jar_weight,'current_status':current_status,'jar_status':jar_status}
    new_document = my_database.create_document(record_data)
    if new_document.exists():
        print(f"Document  successfully created.")
    result_collection = Result(my_database.all_docs,include_docs=True)
    print(f"Retrieved minimal document:\n{result_collection[0]}\n")


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

THE LINK TO NODE RED-

https://node-red-uoedf.eu-gb.mybluemix.net/?_ga=2.152525369.141832829.1593542549-94931584.1593542549

6.WEB APPLICATION

CREATING IBM CLOUD SERVICES:

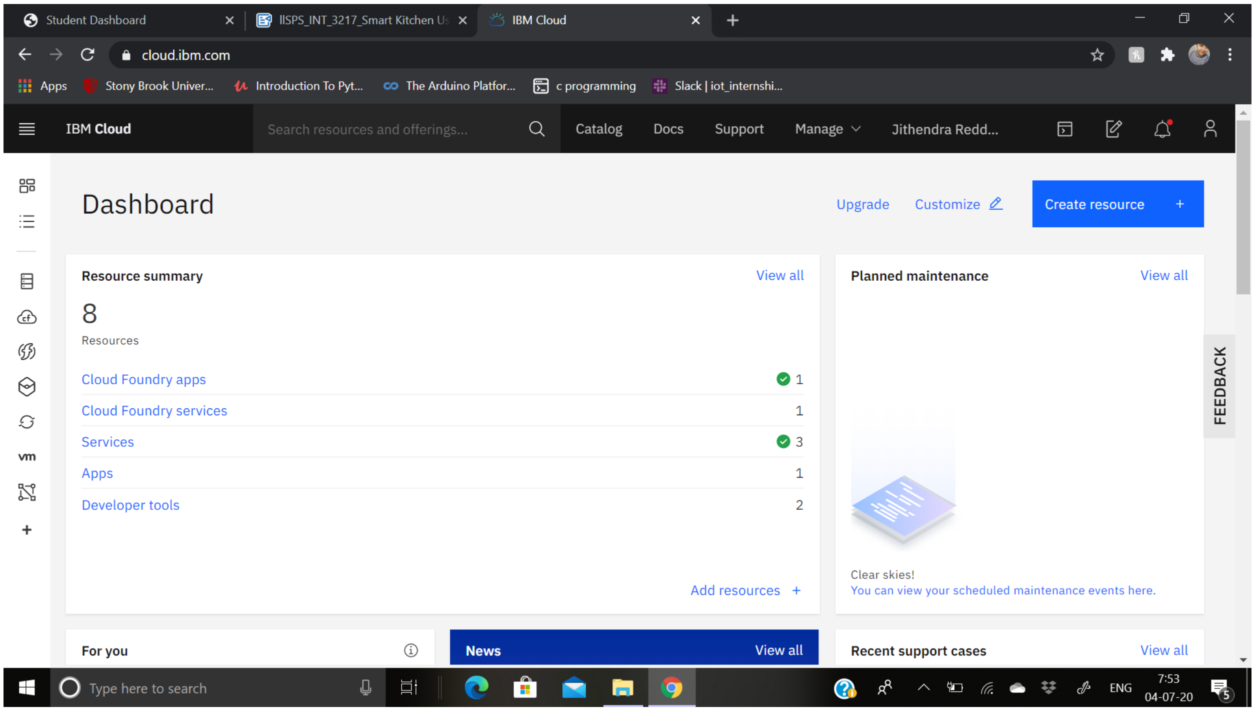


Log in to IBM **Cloud**

Don't have an account? Create an account

Enter your IBMid Forgot ID?

IBMid
banasaijithendrareddy@karunya.edu.in



Dashboard

Upgrade Customize Create resource +

Resource summary View all

8
Resources

Cloud Foundry apps	✓ 1
Cloud Foundry services	1
Services	✓ 3
Apps	1
Developer tools	2

Add resources +

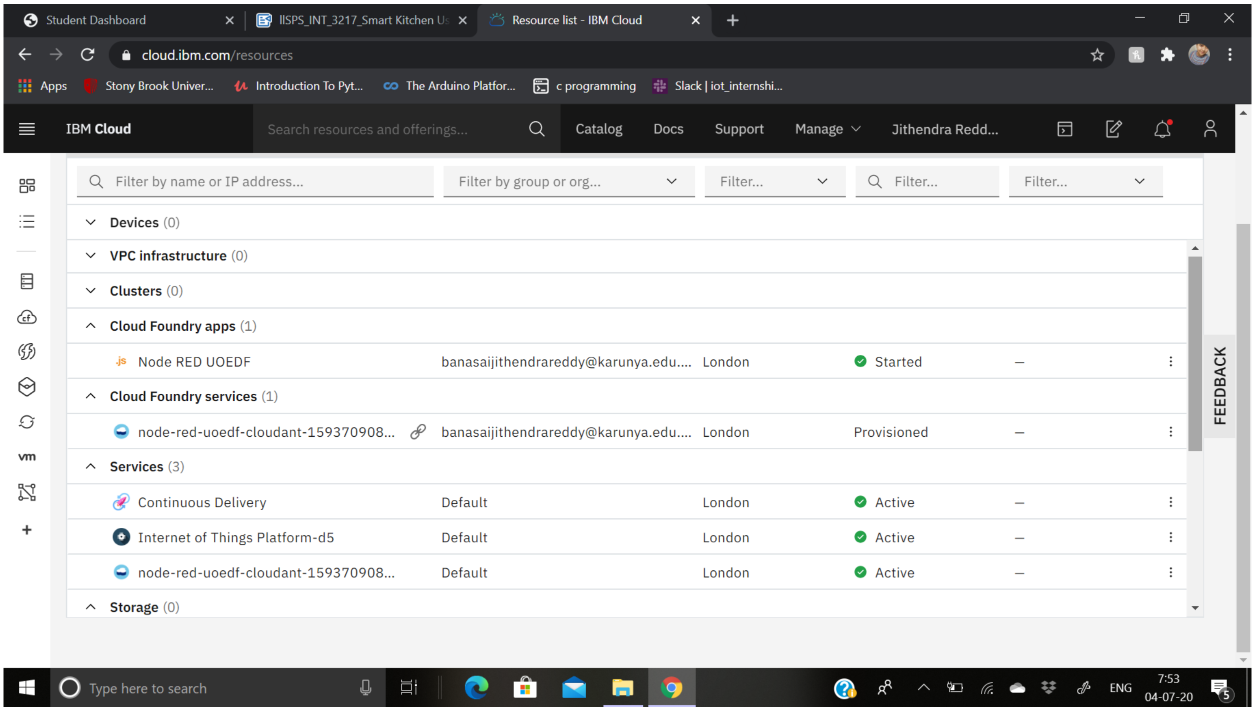
Planned maintenance View all

Clear skies!
You can view your scheduled maintenance events here.

For you

News View all

Recent support cases View all



Devices (0)
VPC infrastructure (0)
Clusters (0)
Cloud Foundry apps (1)

Node RED UOEDF	banasaijithendrareddy@karunya.edu....	London	✓ Started	–

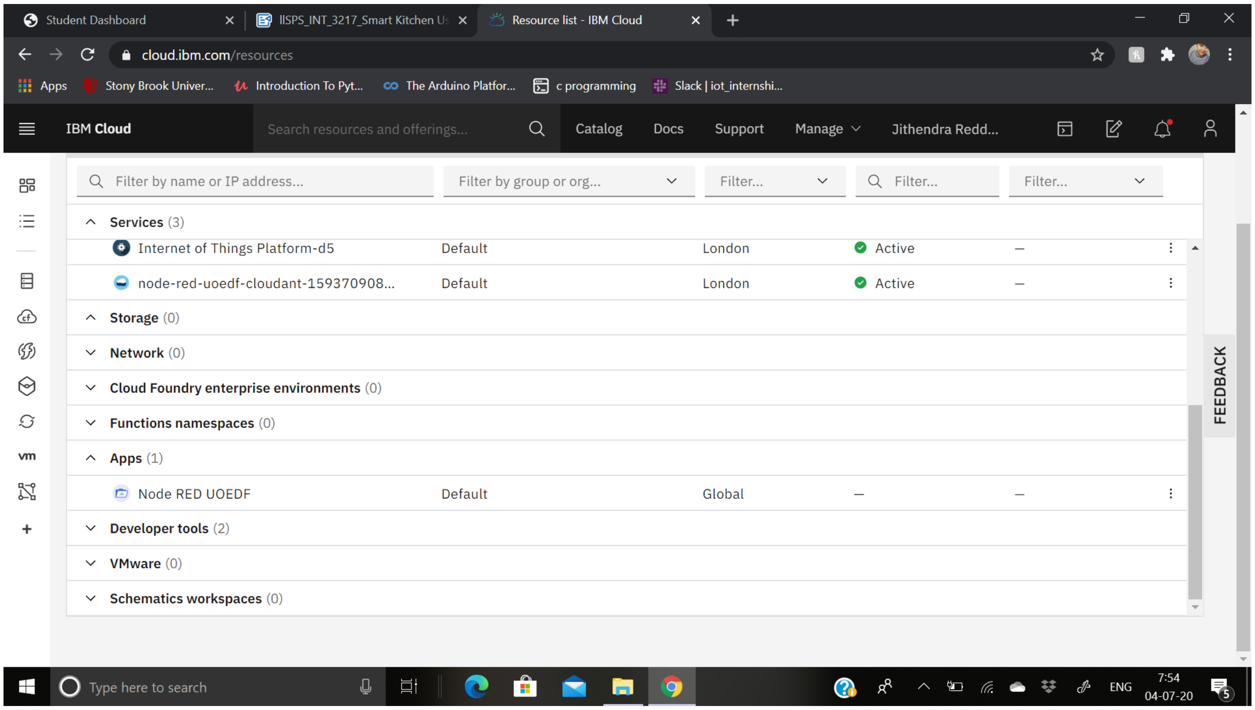
Cloud Foundry services (1)

node-red-uoedf-cloudant-159370908...	banasaijithendrareddy@karunya.edu....	London	Provisioned	–

Services (3)

Continuous Delivery	Default	London	✓ Active	–
Internet of Things Platform-d5	Default	London	✓ Active	–
node-red-uoedf-cloudant-159370908...	Default	London	✓ Active	–

Storage (0)



Services (3)

Internet of Things Platform-d5	Default	London	Active	–
node-red-uoedf-cloudant-159370908...	Default	London	Active	–

Storage (0)

Network (0)

Cloud Foundry enterprise environments (0)

Functions namespaces (0)

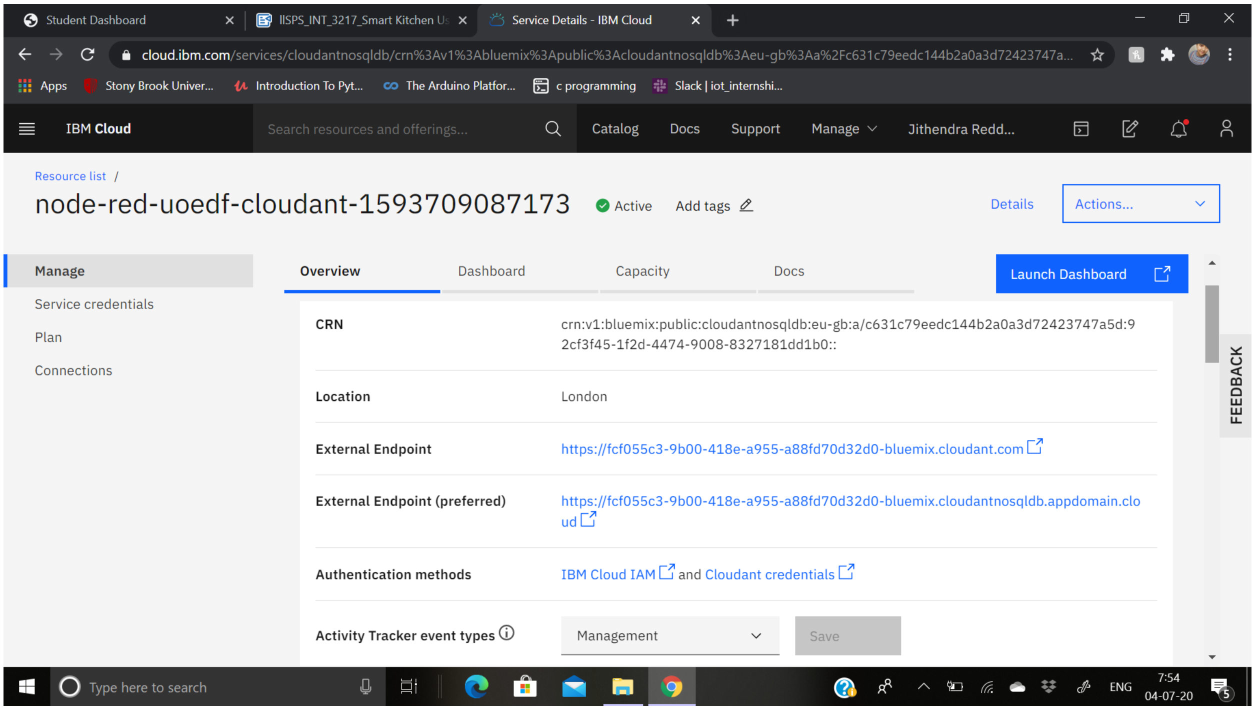
Apps (1)

Node RED UOEDF	Default	Global	–	–

Developer tools (2)

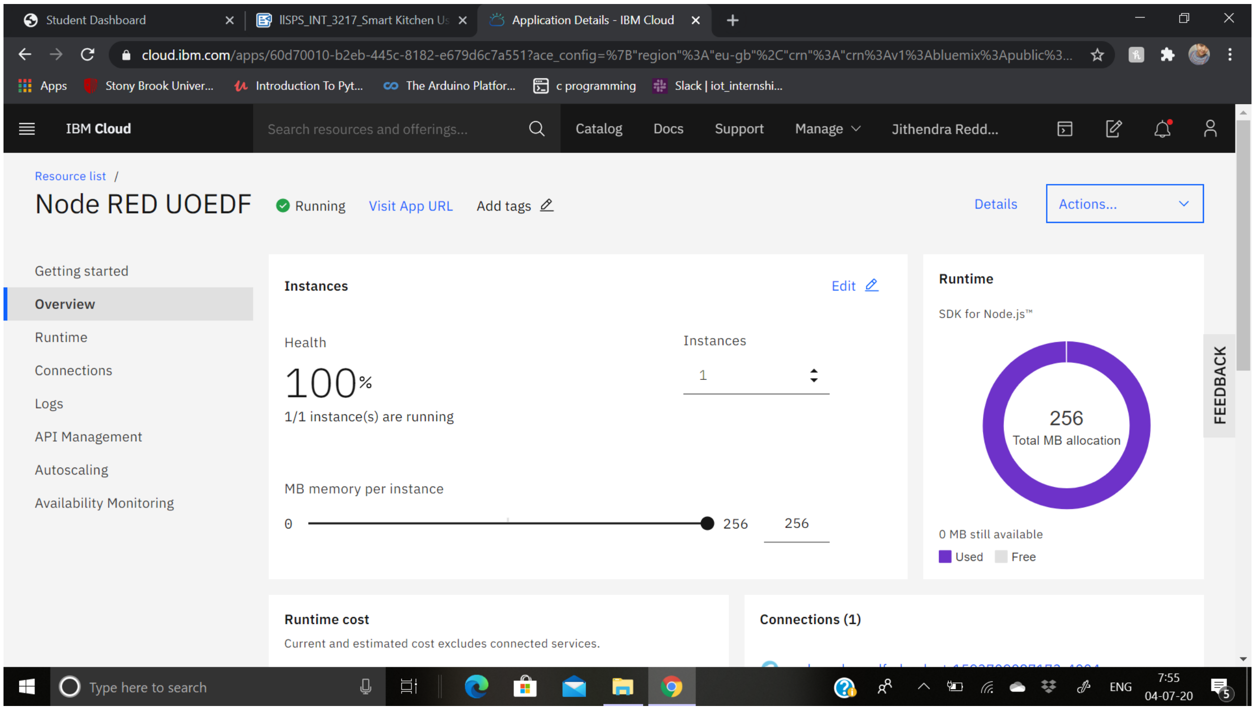
VMware (0)

Schematics workspaces (0)



node-red-uoedf-cloudant-1593709087173 Active

CRN	crn:v1:bluemix:public:cloudantnosqldb:eu-gb:a/c631c79eedc144b2a0a3d72423747a5d:92cf3f45-1f2d-4474-9008-8327181dd1b0::
Location	London
External Endpoint	https://fcf055c3-9b00-418e-a955-a88fd70d32d0-bluemix.cloudant.com
External Endpoint (preferred)	https://fcf055c3-9b00-418e-a955-a88fd70d32d0-bluemix.cloudantnosqldb.appdomain.cloud
Authentication methods	IBM Cloud IAM and Cloudant credentials
Activity Tracker event types	Management



cloud.ibm.com/apps/60d70010-b2eb-445c-8182-e679d6c7a551?ace_config=%7B"region"%3A"eu-gb"%2C"crn"%3A"crn%3Av1%3Abluemix%3Apublic%3...

Node RED UOEDF ● Running Visit App URL Add tags

Instances

Health

100%

1/1 instance(s) are running

Instances

1

MB memory per instance

0 256 256

Runtime

SDK for Node.js™

256 Total MB allocation

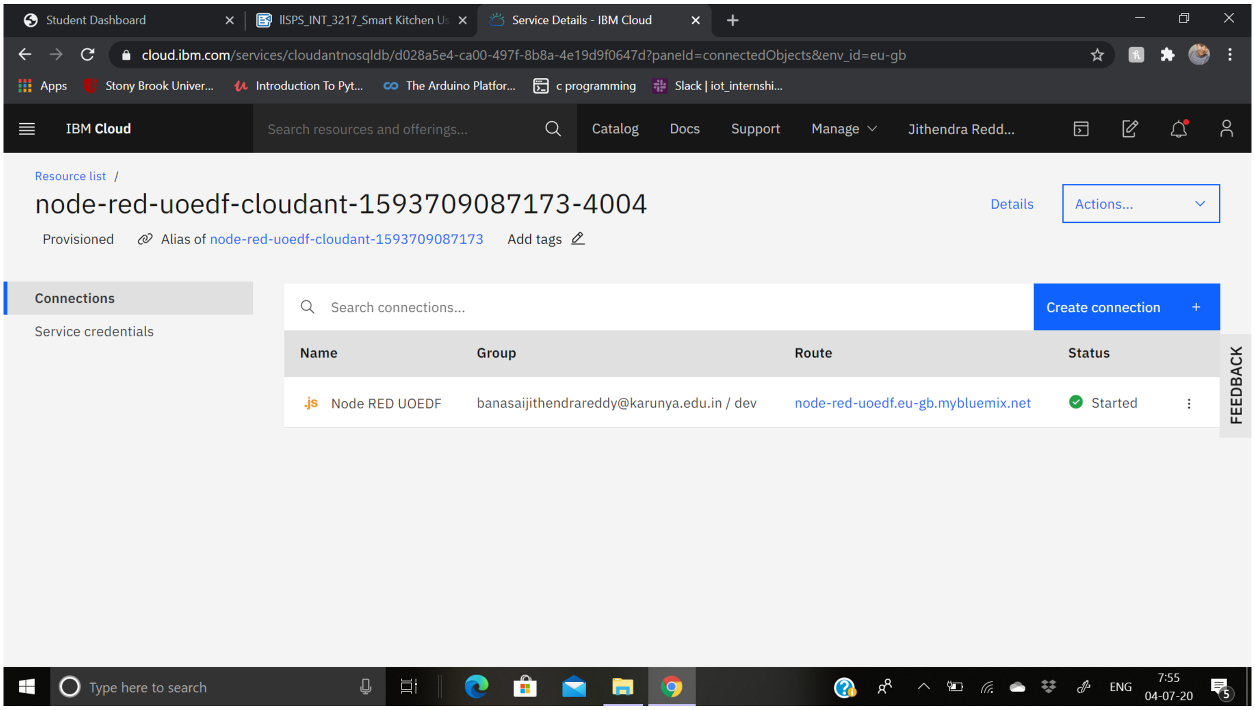
0 MB still available

■ Used ■ Free

Runtime cost

Current and estimated cost excludes connected services.

Connections (1)

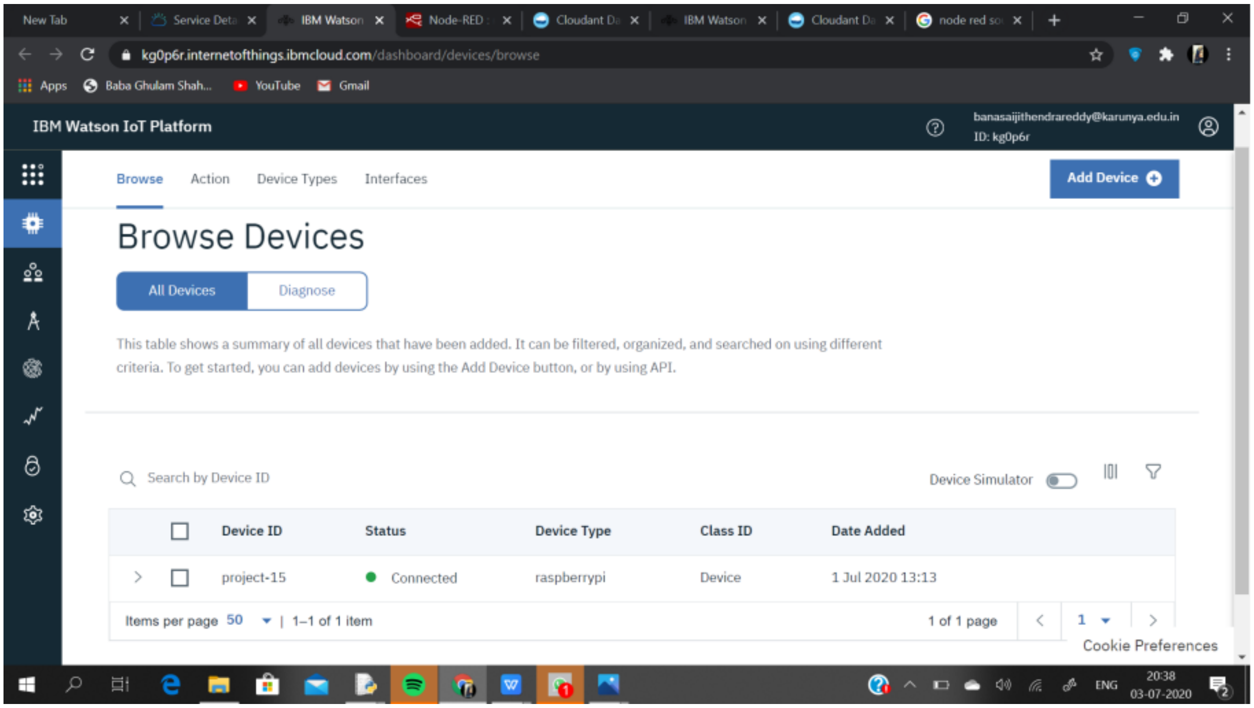


cloud.ibm.com/services/cloudantnosqldb/d028a5e4-ca00-497f-8b8a-4e19d9f0647d?paneld=connectedObjects&env_id=eu-gb

node-red-uoedf-cloudant-1593709087173-4004

Provisioned Alias of node-red-uoedf-cloudant-1593709087173 Add tags

Name	Group	Route	Status
.js Node RED UOEDF	banasaijithendrareddy@karunya.edu.in / dev	node-red-uoedf.eu-gb.mybluemix.net	● Started


kg0p6r.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
banasaijithendrareddy@karunya.edu.in
ID: kg0p6r
Browse
Action
Device Types
Interfaces
Add Device

Browse Devices

All Devices | Diagnose

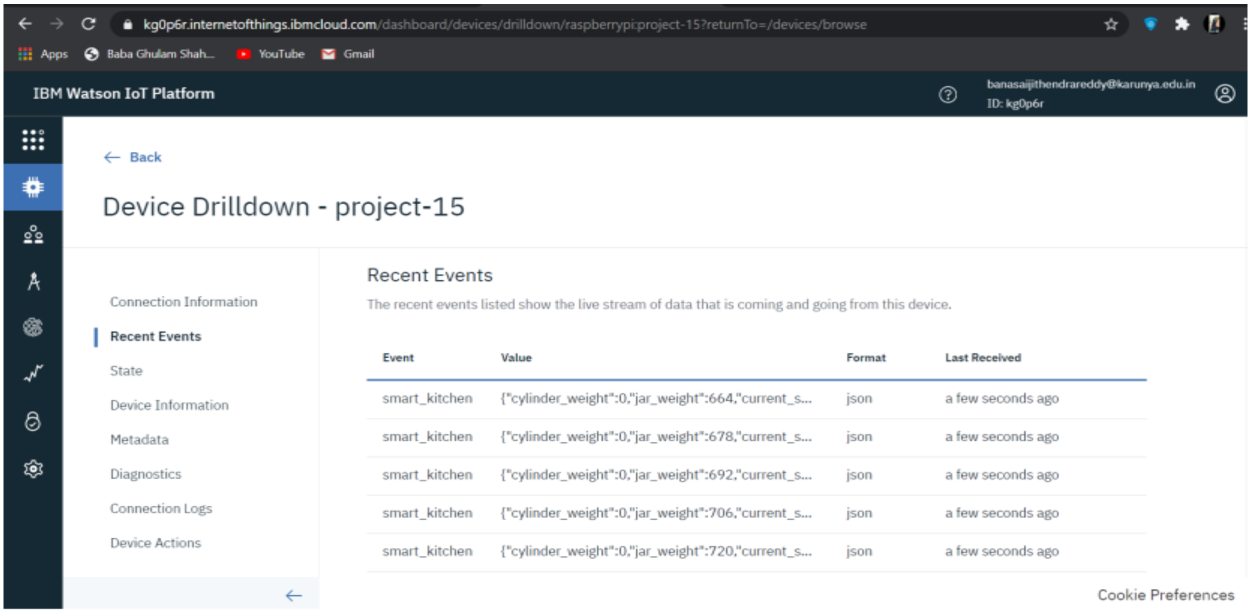
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Search by Device ID   Device Simulator

	Device ID	Status	Device Type	Class ID	Date Added
□	project-15	● Connected	raspberrypi	Device	1 Jul 2020 13:13

Items per page 50 | 1–1 of 1 item   1 of 1 page

FOR STORING THE DATAS IN THE CLOUD :

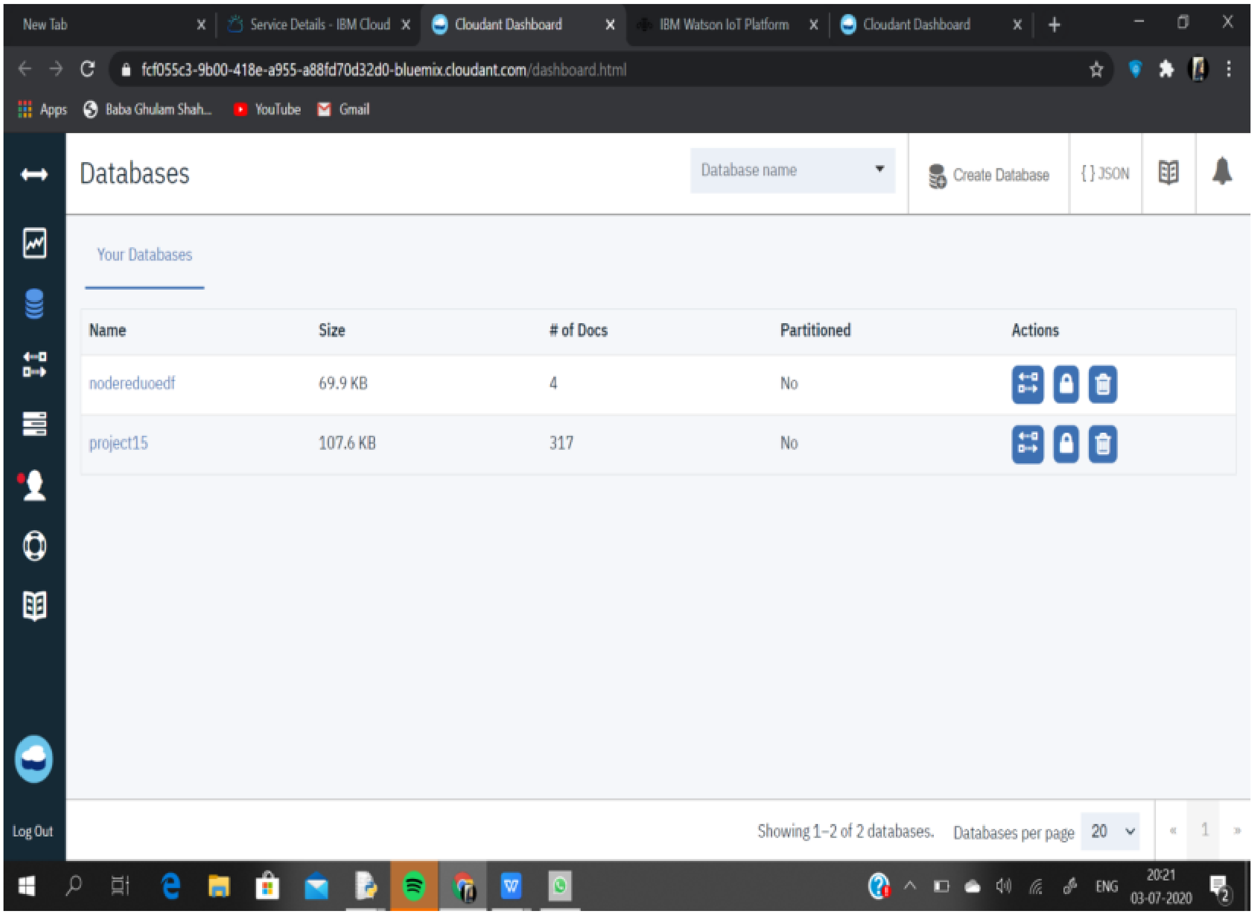


Device Drilldown - project-15

Recent Events

The recent events listed show the live stream of data that is coming and going from this device.

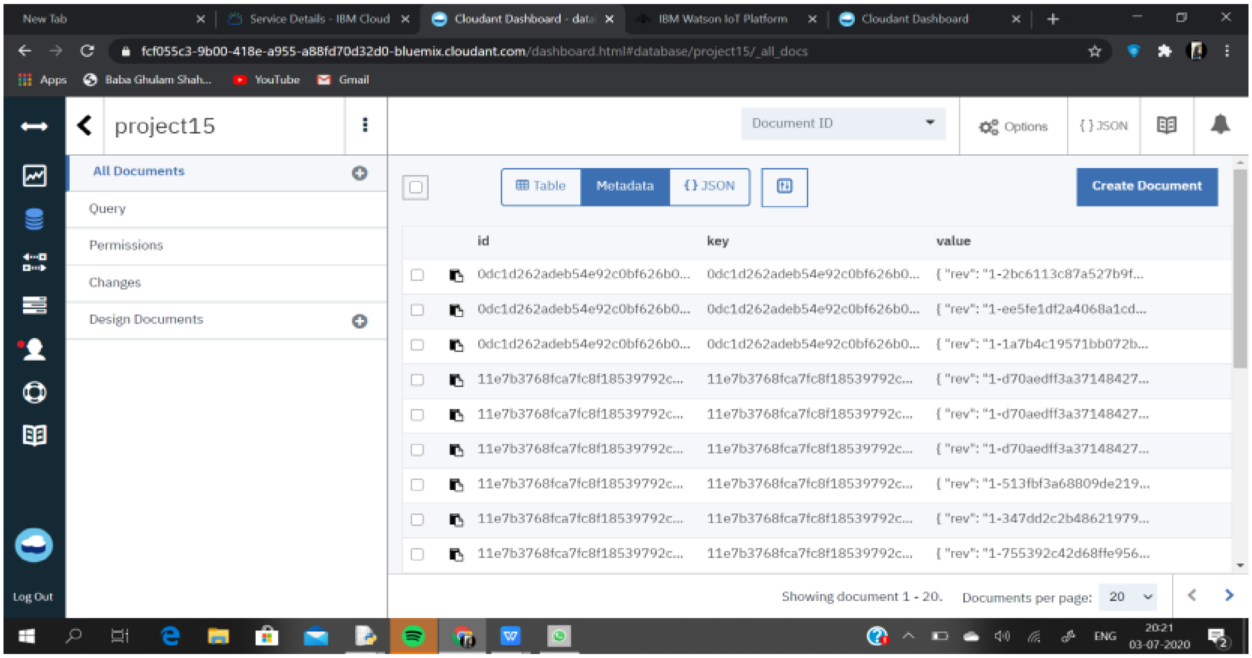
Event	Value	Format	Last Received
smart_kitchen	{"cylinder_weight":0,"jar_weight":664,"current_s...	json	a few seconds ago
smart_kitchen	{"cylinder_weight":0,"jar_weight":678,"current_s...	json	a few seconds ago
smart_kitchen	{"cylinder_weight":0,"jar_weight":692,"current_s...	json	a few seconds ago
smart_kitchen	{"cylinder_weight":0,"jar_weight":706,"current_s...	json	a few seconds ago
smart_kitchen	{"cylinder_weight":0,"jar_weight":720,"current_s...	json	a few seconds ago



Databases

Name	Size	# of Docs	Partitioned	Actions
nodereduoedf	69.9 KB	4	No	
project15	107.6 KB	317	No	

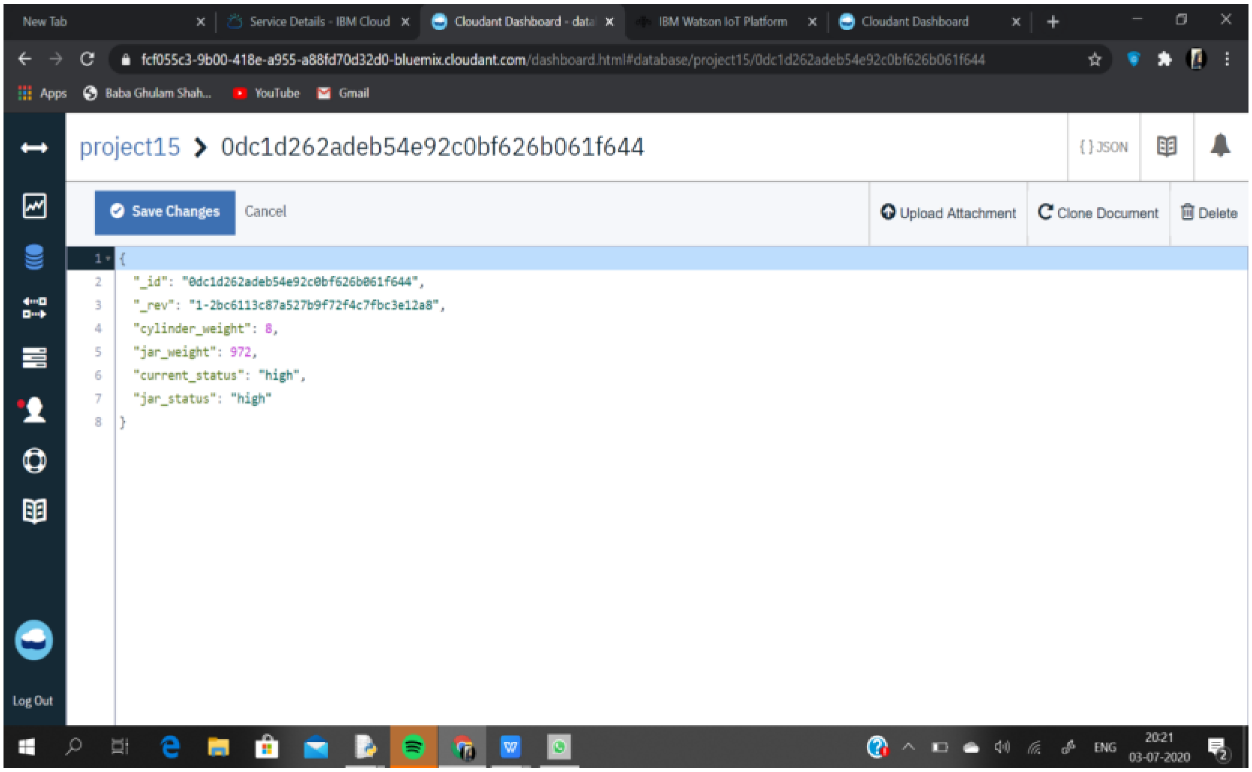
Showing 1–2 of 2 databases. Databases per page 20



id	key	value
0dc1d262adeb54e92c0bf626b0...	0dc1d262adeb54e92c0bf626b0...	{ "rev": "1-2bc6113c87a527b9f...
0dc1d262adeb54e92c0bf626b0...	0dc1d262adeb54e92c0bf626b0...	{ "rev": "1-ee5fe1df2a4068a1cd...
0dc1d262adeb54e92c0bf626b0...	0dc1d262adeb54e92c0bf626b0...	{ "rev": "1-1a7b4c19571bb072b...
11e7b3768fca7fc8f18539792c...	11e7b3768fca7fc8f18539792c...	{ "rev": "1-d70aedff3a37148427...
11e7b3768fca7fc8f18539792c...	11e7b3768fca7fc8f18539792c...	{ "rev": "1-d70aedff3a37148427...
11e7b3768fca7fc8f18539792c...	11e7b3768fca7fc8f18539792c...	{ "rev": "1-d70aedff3a37148427...
11e7b3768fca7fc8f18539792c...	11e7b3768fca7fc8f18539792c...	{ "rev": "1-513fbf3a68809de219...
11e7b3768fca7fc8f18539792c...	11e7b3768fca7fc8f18539792c...	{ "rev": "1-347dd2c2b48621979...
11e7b3768fca7fc8f18539792c...	11e7b3768fca7fc8f18539792c...	{ "rev": "1-755392c42d68ffe956...

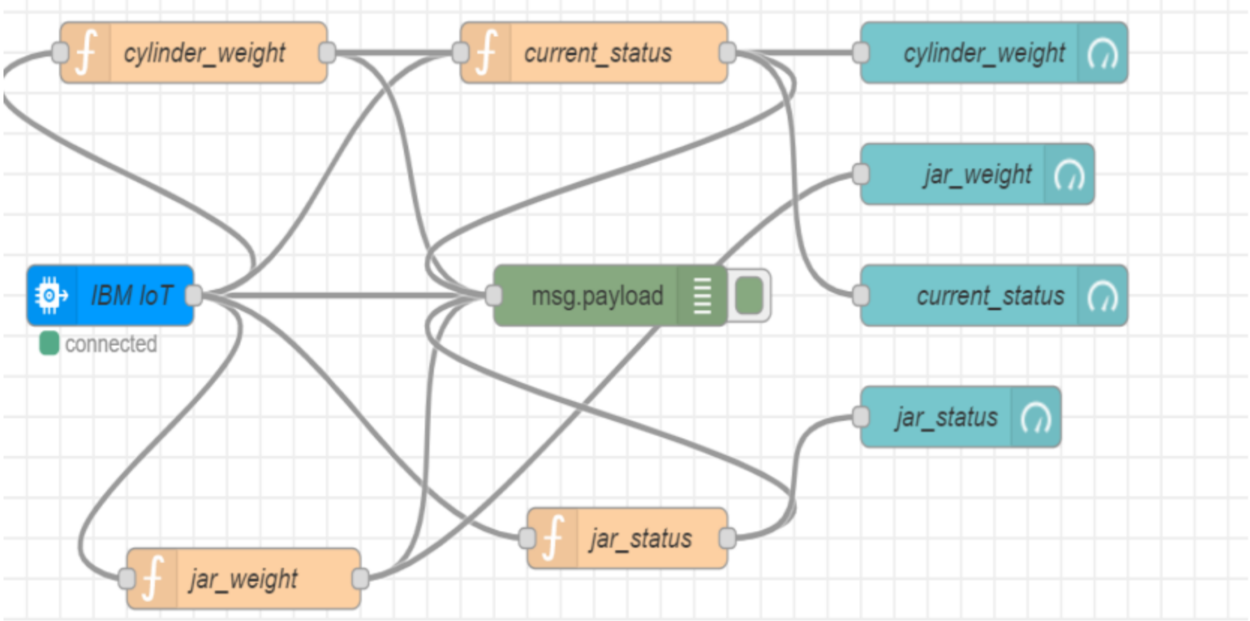
Showing document 1 - 20. Documents per page: 20

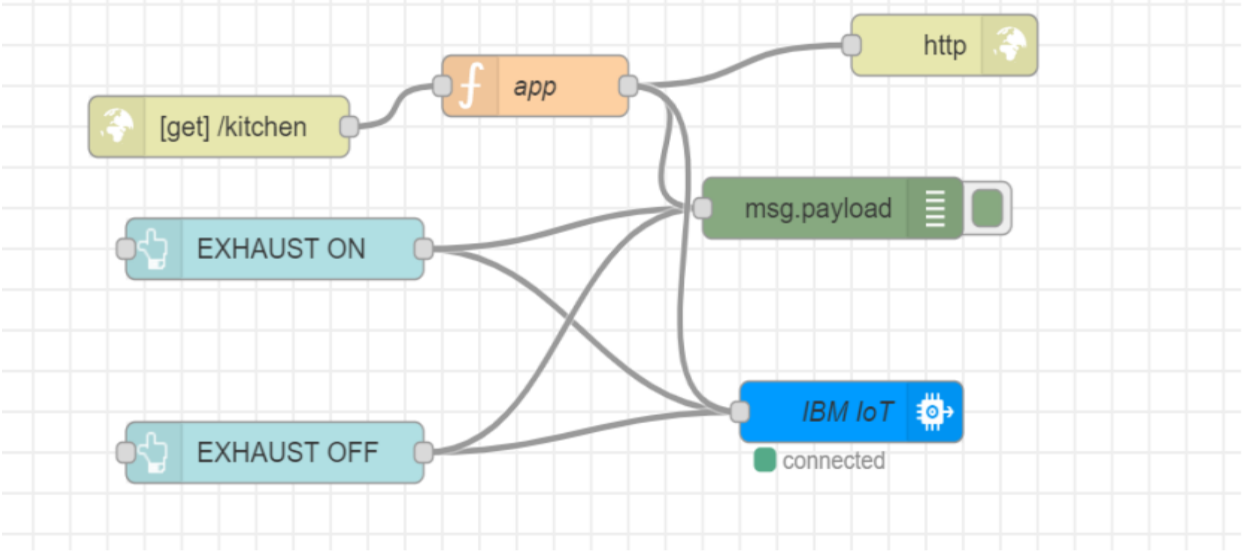
1

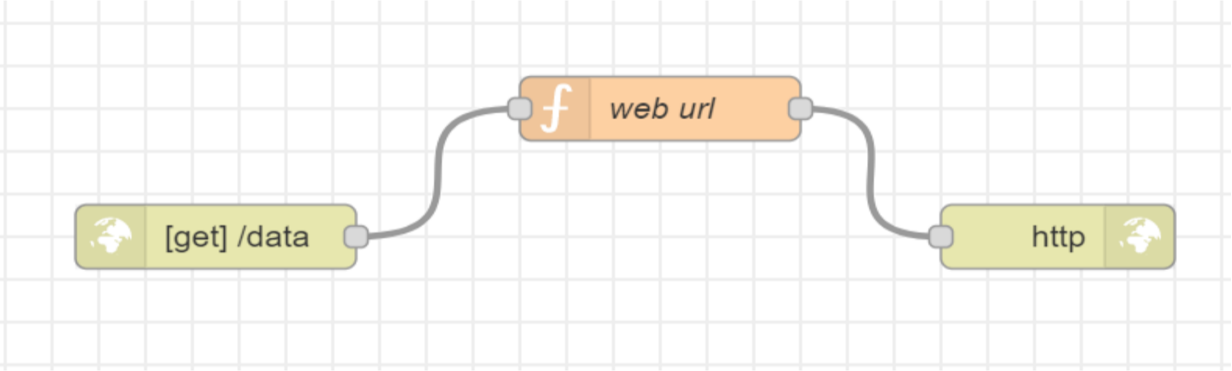


```
{
  "_id": "0dc1d262adeb54e92c0bf626b061f644",
  "_rev": "1-2bc6113c87a527b9f72f4c7fbc3e12a8",
  "cylinder_weight": 8,
  "jar_weight": 972,
  "current_status": "high",
  "jar_status": "high"
}
```

7. .NODE RED FLOW-







8.THE NODE RED SOURCE CODE-

[{"id":"ef8b0c47.89464","type":"tab","label":"Flow
1","disabled":false,"info":""},{"id":"d6ba6dcc.080b4","type":"ibmiot
in","z":"ef8b0c47.89464","authentication":"apiKey","apiKey":"a0757bc6.3ac788","inputType"
:"evt","logicalInterface":"","ruleId":"","deviceId":"project-15","applicationId":"","deviceType":"ra
spberrypi","eventType":"smart_kitchen","commandType":"","format":"json","name":"IBM
IoT","service":"registered","allDevices":"","allApplications":"","allDeviceTypes":"","allLogicalIn
terfaces":"","allEvents":"","allCommands":"","allFormats":"","qos":0,"x":70,"y":160,"wires":[["a6
6564a4.8f1fa8","a91f2b53.663ca8","39c79dcb.457bd2","38ccb347.3acd8c","8c332a56.df
1628"]]},{"id":"a66564a4.8f1fa8","type":"debug","z":"ef8b0c47.89464","name":"","active":true
,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","s
tatusVal":"","statusType":"auto","x":370,"y":160,"wires":[]},{"id":"a91f2b53.663ca8","type":"fu
nction","z":"ef8b0c47.89464","name":"cylinder_weight","func":"global.set('cylinder_weight',
msg.payload.cylinder_weight)\nmsg.payload=msg.payload.cylinder_weight\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":120,"y":40,"wires":[["a66564a4.8f1fa8
","e5cd5746.d40e78"]]},{"id":"39c79dcb.457bd2","type":"function","z":"ef8b0c47.89464","na
me":"jar_weight","func":"global.set('jar_weight',msg.payload.jar_weight)\nmsg.payload=
msg.payload.jar_weight\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":150,"y":300,"wires":[["a66564a4.8f1fa
8","36d098ac.05e038"]]},{"id":"8c332a56.df1628","type":"function","z":"ef8b0c47.89464","n
ame":"jar_status","func":"global.set('jar_status',msg.payload.jar_status)\nmsg.payload=
msg.payload.jar_status\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":380,"y":280,"wires":[["a66564a4.8f1fa
8","29e4490.d62d3b8"]]},{"id":"38ccb347.3acd8c","type":"function","z":"ef8b0c47.89464","n
ame":"current_status","func":"global.set('current_status',msg.payload.current_status)\nm
sg.payload=msg.payload.current_status\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":360,"y":40,"wires":[["a66564a4.8f1fa8
","73a29b42.0ad7b4"]]},{"id":"e5cd5746.d40e78","type":"ui_gauge","z":"ef8b0c47.89464","n
ame":"cylinder_weight","group":"19324344.bd6a0d","order":0,"width":0,"height":0,"gtype":"d
onut","title":"cylinder_weight","label":"kilograms","format":"{{value}}","min":0,"max":"100","co
lors":["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":600,"y":40,"wires":[]},{"id":"36d0
98ac.05e038","type":"ui_gauge","z":"ef8b0c47.89464","name":"jar_weight","group":"3e78b3
30.9f7a4c","order":1,"width":0,"height":0,"gtype":"gage","title":"jar_weight","label":"grams","fo
rmat":"{{value}}","min":0,"max":"1000","colors":["#00b500","#e6e600","#ca3838"],"seg1":"","s
eg2":"","x":590,"y":100,"wires":[]},{"id":"73a29b42.0ad7b4","type":"ui_gauge","z":"ef8b0c47.8
9464","name":"current_status","group":"4e84342a.77d52c","order":0,"width":0,"height":0,"gt

ype":"gage","title":"current_status","label":"kilograms","format":"{{value}}","min":0,"max":"10
0","colors":["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":600,"y":160,"wires":[]},{"id
":"29e4490.d62d3b8","type":"ui_gauge","z":"ef8b0c47.89464","name":"jar_status","group":"
33a94f0e.61524","order":0,"width":0,"height":0,"gtype":"gage","title":"jar_status","label":"gra
ms","format":"{{value}}","min":0,"max":"1000","colors":["#00b500","#e6e600","#ca3838"],"se
g1":"","seg2":"","x":580,"y":220,"wires":[]},{"id":"8e29eb85.d67248","type":"ui_button","z":"ef8b
0c47.89464","name":"","group":"bd555a02.900608","order":0,"width":0,"height":0,"passthru":
false,"label":"EXHAUST
ON","tooltip":"","color":"","bgcolor":"","icon":"","payload":"{\"command\":\"exhauston\"}","payl
oadType":"str","topic":"","x":210,"y":440,"wires":[["58c1fd64.2c9984","1a95e5e9.f66bca"]]},{"
id":"bd6a9ffb.3c311","type":"ui_button","z":"ef8b0c47.89464","name":"","group":"84947f66.
e35c9","order":0,"width":0,"height":0,"passthru":false,"label":"EXHAUST
OFF","tooltip":"","color":"","bgcolor":"","icon":"","payload":"{\"command\":\"exhauston\"}","payl
oadType":"str","topic":"","x":210,"y":540,"wires":[["58c1fd64.2c9984","1a95e5e9.f66bca"]]},{"
id":"58c1fd64.2c9984","type":"ibmiot
out","z":"ef8b0c47.89464","authentication":"apiKey","apiKey":"a0757bc6.3ac788","outputTy
pe":"evt","deviceId":"project-15","deviceType":"raspberrypi","eventCommandType":"smart_k
itchen","format":"json","data":"blink","qos":0,"name":"IBM
IoT","service":"registered","x":520,"y":520,"wires":[]},{"id":"1a95e5e9.f66bca","type":"debug","
z":"ef8b0c47.89464","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,
"complete":"false","statusVal":"","statusType":"auto","x":510,"y":420,"wires":[]},{"id":"b68d548
7.a08c48","type":"http
in","z":"ef8b0c47.89464","name":"","url":"/data","method":"get","upload":false,"swaggerDoc":
"","x":210,"y":680,"wires":[["319d53be.99b6bc"]]},{"id":"fc447542.5ea888","type":"http
response","z":"ef8b0c47.89464","name":"","statusCode":"","headers":{},"x":570,"y":680,"wires
":[]},{"id":"319d53be.99b6bc","type":"function","z":"ef8b0c47.89464","name":"web
url","func":"msg.payload={\"cylinder_weight\":global.get(\"cylinder_weight\"),\"current_st
atus\":global.get(\"current_status\"),\"jar_weight\":global.get(\"jar_weight\"),\"jar_status
\":global.get(\"jar_status\")}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":400,"y":620,"wires":[["fc447542.5ea8
88"]]},{"id":"b862bd99.4f53c","type":"function","z":"ef8b0c47.89464","name":"app","func":"m
sg.payload=msg.payload.command\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":350,"y":360,"wires":[["b06c19a0.6bcd
38","1a95e5e9.f66bca","58c1fd64.2c9984"]]},{"id":"47cea2d.e33135c","type":"http
in","z":"ef8b0c47.89464","name":"","url":"/kitchen","method":"get","upload":false,"swaggerDo
c":"","x":180,"y":380,"wires":[["b862bd99.4f53c"]]},{"id":"b06c19a0.6bcd38","type":"http

response","z":"ef8b0c47.89464","name":"","statusCode":"","headers":{},"x":570,"y":340,"wires
":[]},{"id":"a0757bc6.3ac788","type":"ibmiot","z":"","name":"API","keepalive":"60","serverNam
e":"kg0p6r.messaging.internetofthings.ibmcloud.com","cleansession":true,"appId":"","shar
ed":false},{"id":"19324344.bd6a0d","type":"ui_group","z":"","name":"cylinder_weight","tab":"d
a2bb438.c2b718","order":1,"disp":true,"width":"6","collapse":false},{"id":"3e78b330.9f7a4c",
"type":"ui_group","z":"","name":"jar-weight","tab":"da2bb438.c2b718","order":3,"disp":true,"wi
dth":"6","collapse":false},{"id":"4e84342a.77d52c","type":"ui_group","z":"","name":"current_st
atus","tab":"da2bb438.c2b718","order":4,"disp":true,"width":"6","collapse":false},{"id":"33a94
f0e.61524","type":"ui_group","z":"","name":"jar_status","tab":"da2bb438.c2b718","order":5,"di
sp":true,"width":"6","collapse":false},{"id":"bd555a02.900608","type":"ui_group","z":"","name"
:"EXHAUST
ON","tab":"da2bb438.c2b718","order":5,"disp":true,"width":"6","collapse":false},{"id":"84947f
66.e35c9","type":"ui_group","z":"","name":"EXHAUST
OFF","tab":"da2bb438.c2b718","order":6,"disp":true,"width":"6","collapse":false},{"id":"da2bb
438.c2b718","type":"ui_tab","z":"","name":"smart_kitchen
","icon":"dashboard","disabled":false,"hidden":false}]

THE OUTPUT-

IN NODE RED-

9.Web Application User Interface:-

User Interface (UI) Design focuses on anticipating what users might need to do and ensuring that the interface has elements that are easy to access, understand, and use to facilitate those actions. UI brings together concepts from interaction design, visual design, and information architecture.

Choosing Interface Elements

Users have become familiar with interface elements acting in a certain way, so try to be consistent and predictable in your choices and their layout. Doing so will help with task completion, efficiency, and satisfaction.

Interface elements include but are not limited to:

- **Input Controls**: buttons, text fields, checkboxes, radio buttons, dropdown lists, list boxes, toggles, date field
- **Navigational Components**: breadcrumb, slider, search field, pagination, slider, tags, icons
- **Informational Components**: tooltips, icons, progress bar, notifications, message boxes, modal windows
- **Containers**: accordion

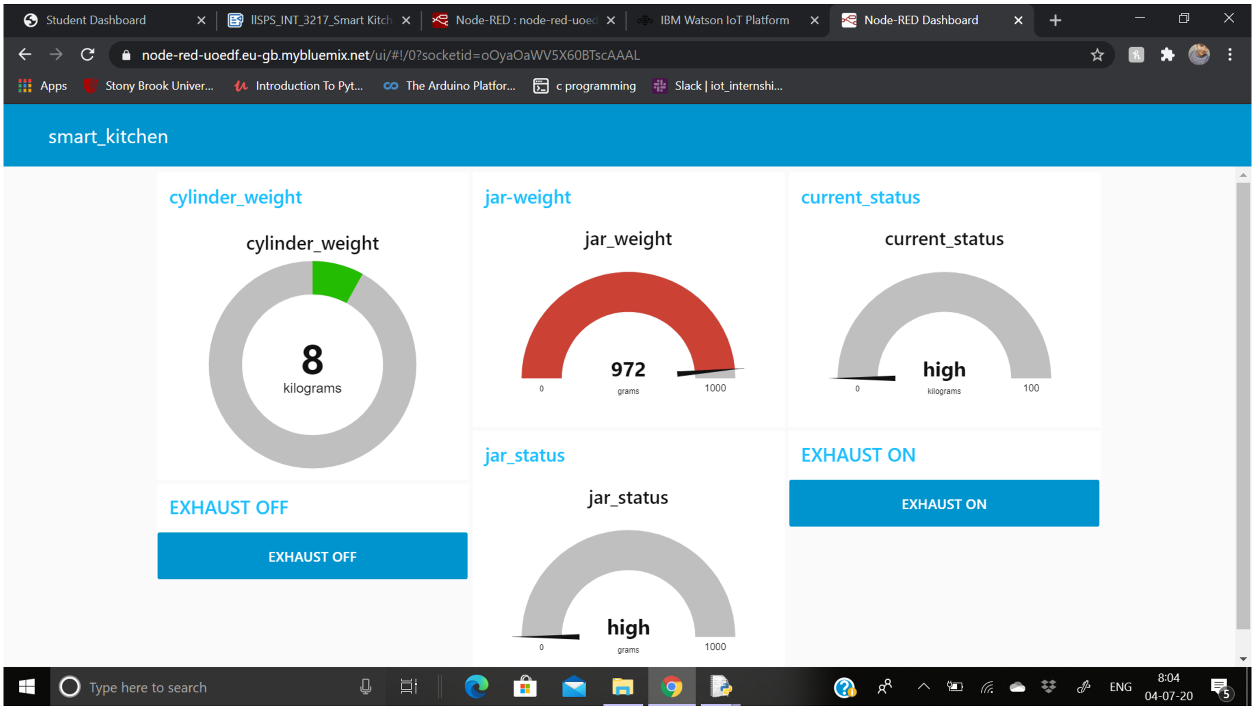
There are times when multiple elements might be appropriate for displaying content. When this happens, it’s important to consider the trade-offs. For example, sometimes elements that can help save you space, put more of a burden on the user mentally by forcing them to guess what is within the dropdown or what the element might be.

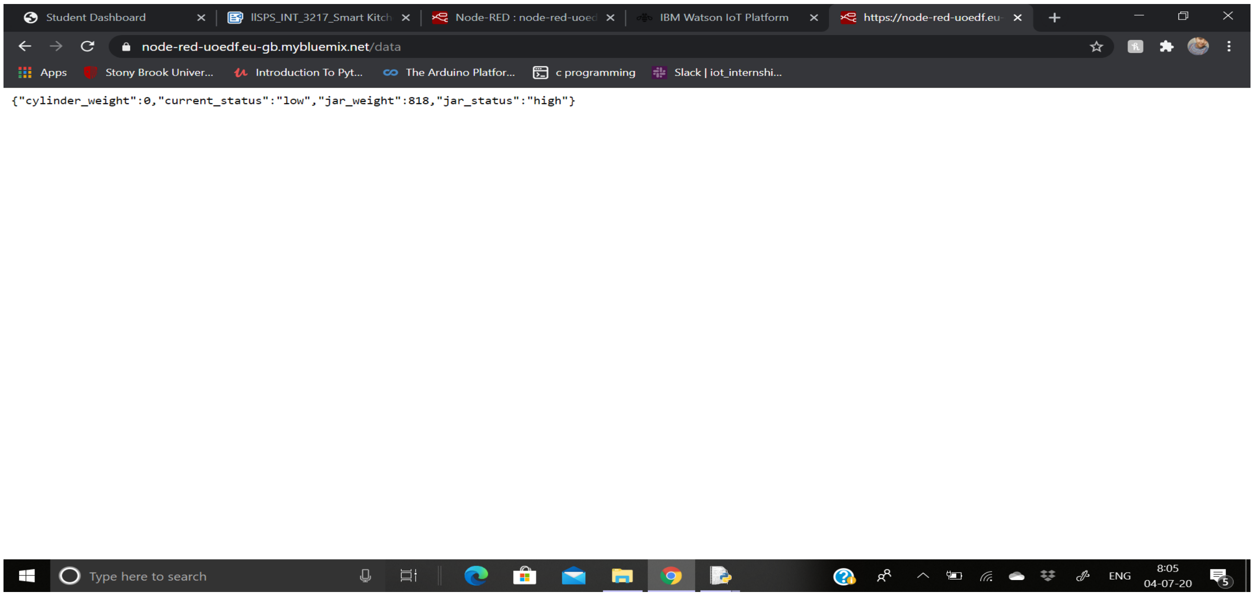
Best Practices for Designing an Interface

Everything stems from knowing your users, including understanding their goals, skills, preferences, and tendencies. Once you know about your user, make sure to consider the following when designing your interface:

- **Keep the interface simple.** The best interfaces are almost invisible to the user. They avoid unnecessary elements and are clear in the language they use on labels and in messaging.
- **Create consistency and use common UI elements.** By using common elements in your UI, users feel more comfortable and are able to get things done more quickly. It is also important to create patterns in language, layout and design throughout the site to help facilitate efficiency. Once a user learns how to do something, they should be able to transfer that skill to other parts of the site.
- **Be purposeful in page layout.** Consider the spatial relationships between items on the page and structure the page based on importance. Careful placement of items can help draw attention to the most important pieces of information and can aid scanning and readability.
- **Strategically use color and texture.** You can direct attention toward or redirect attention away from items using color, light, contrast, and texture to your advantage.
- **Use typography to create hierarchy and clarity.** Carefully consider how you use typeface. Different sizes, fonts, and arrangement of the text to help increase scanability, legibility and readability.
- **Make sure that the system communicates what's happening.** Always inform your users of location, actions, changes in state, or errors. The use of various UI elements to communicate status and, if necessary, next steps can reduce frustration for your user.
- **Think about the defaults.** By carefully thinking about and anticipating the goals people bring to your site, you can create defaults that reduce the burden on the

- user. This becomes particularly important when it comes to form design where you might have an opportunity to have some fields pre-chosen or filled out.





```
{"cylinder_weight":0,"current_status":"low","jar_weight":818,"jar_status":"high"}
```

10.Mit App Inventor :-

MIT App Inventor is a web application integrated development environment originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). It allows newcomers to computer programming to create application software(apps) for two operating systems (OS): Android, and iOS, which, as of 8 July 2019, is in final beta testing. It is free and open-source software released under dual licensing: a Creative Commons Attribution ShareAlike 3.0 Unported license, and an Apache License 2.0 for the source code.

It uses a graphical user interface (GUI) very similar to the programming languages Scratch (programming language) and the StarLogo, which allows users to drag and drop visual objects to create an application that can run on android devices, while a App-Inventor Companion (The program that allows the app to run and debug on) that works on iOS running devices are still under development. In creating App Inventor, Google drew upon significant prior research in educational computing, and work done within Google on online development environments.

App Inventor and the other projects are based on and informed by constructionist learning theories, which emphasize that programming can be a vehicle for engaging powerful ideas through active learning. As such, it is part of an ongoing movement in computers and education that began with the work of Seymour Papert and the MIT Logo Group in the 1960s, and has also manifested itself with Mitchel Resnick's work on Lego Mindstorms and StarLogo.

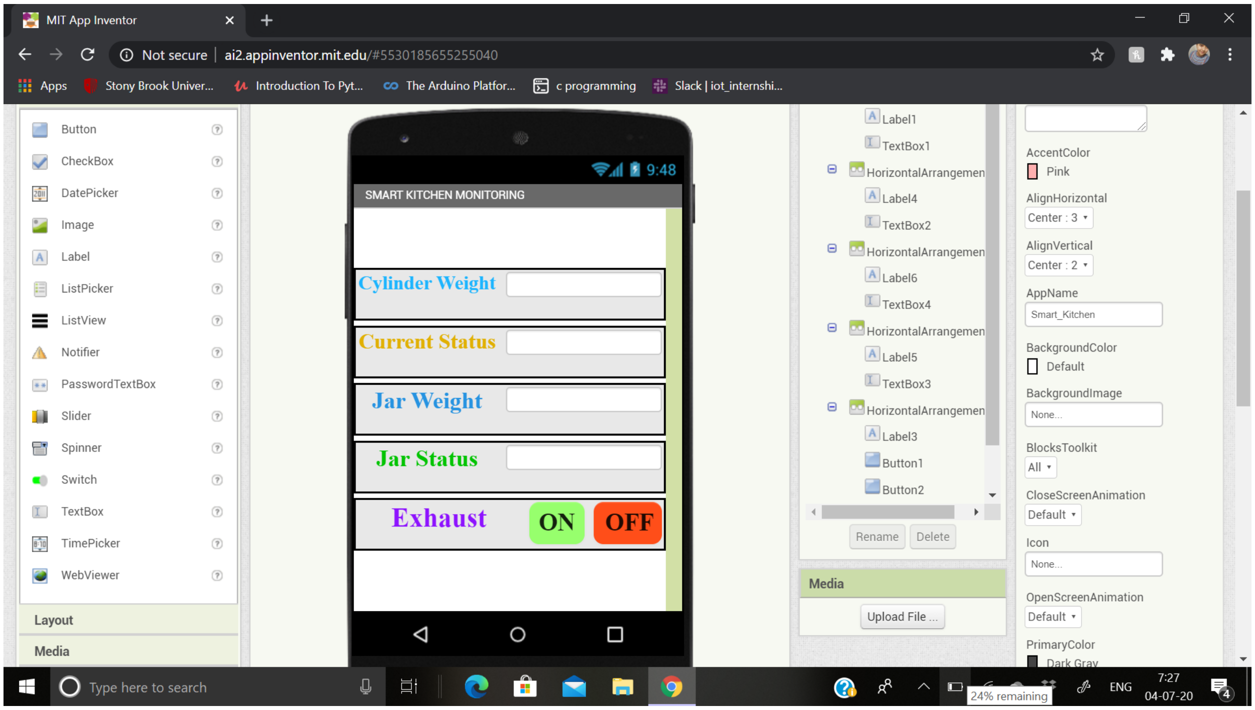
App Inventor also supports the use of cloud data via an experimental Firebase#Firebase Realtime Database component.

So in our project we have mad a mobile app to monitor the status of the components present in the kitchen. The following are the option which will be vivible in the mobile app:

1. Cylinder weight
2. Current status
3. Jar weight
4. Jar status
5. Exhaust - ON / OFF

So it is not necessary to use the web application to access the smart kitchen everytime.It can be used through the phone also.

The screenshots of the app is given below.



SMART KITCHEN MONITORING

Cylinder Weight

Current Status

Jar Weight

Jar Status

Exhaust ON OFF

AccentColor: Pink

AlignHorizontal: Center : 3

AlignVertical: Center : 2

AppName: Smart_Kitchen

BackgroundColor: Default

BackgroundImage: None...

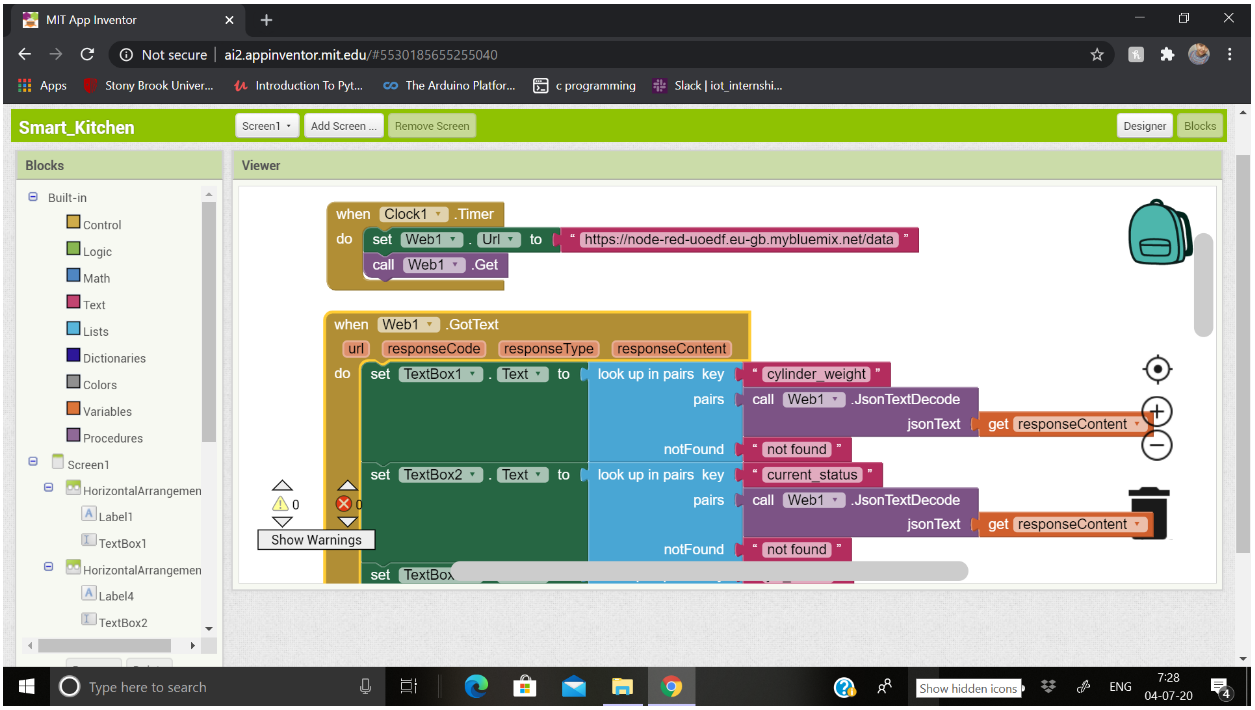
BlocksToolkit: All

CloseScreenAnimation: Default

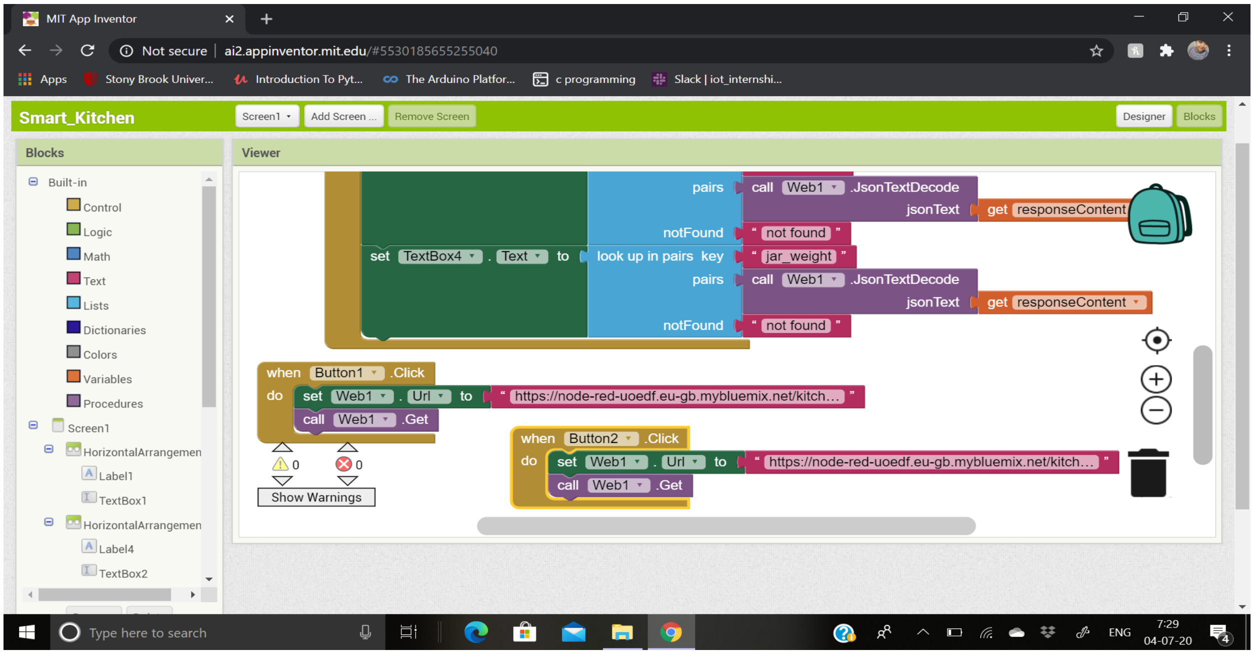
Icon: None...

OpenScreenAnimation: Default

PrimaryColor: Dark Gray



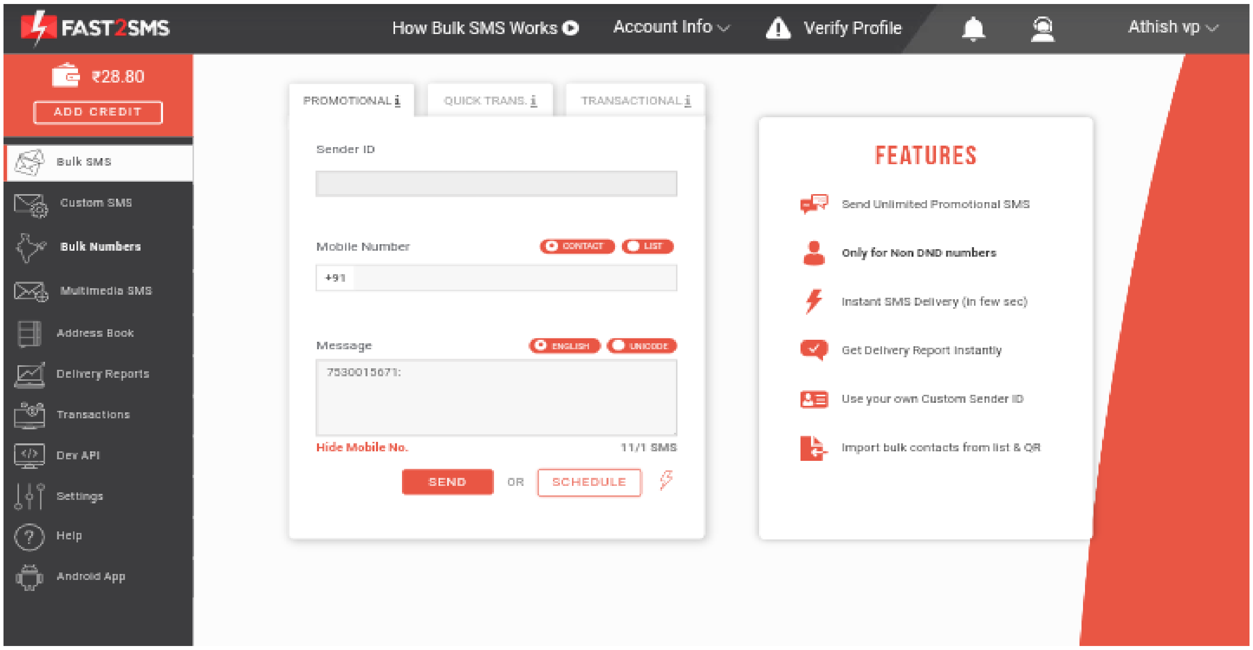
```
when Clock1 .Timer
do  set Web1 . Url to " https://node-red-uoedf.eu-gb.mybluemix.net/data "
    call Web1 .Get

when Web1 .GotText
  url  responseCode  responseType  responseContent
do  set TextBox1 . Text to  look up in pairs  key " cylinder_weight "
                                              pairs  call Web1 .JsonTextDecode
                                                          jsonText  get responseContent
                                              notFound " not found "
    set TextBox2 . Text to  look up in pairs  key " current_status "
                                              pairs  call Web1 .JsonTextDecode
                                                          jsonText  get responseContent
                                              notFound " not found "
    set TextBox
```



```
set TextBox4 . Text to  look up in pairs  key  " jar_weight "
                                          pairs     call Web1 .JsonTextDecode
                                                                jsonText  get responseContent
                                          notFound  " not found "
        pairs     call Web1 .JsonTextDecode
                            jsonText  get responseContent
        notFound  " not found "

when Button1 .Click
do  set Web1 . Url to  " https://node-red-uoedf.eu-gb.mybluemix.net/kitch... "
    call Web1 .Get

when Button2 .Click
do  set Web1 . Url to  " https://node-red-uoedf.eu-gb.mybluemix.net/kitch... "
    call Web1 .Get
```

11.Sending sms to mobile:

FAST2SMS is a service that helps to send sms to your mobile.We have integrated that to the application.



Sender ID

Mobile Number +91

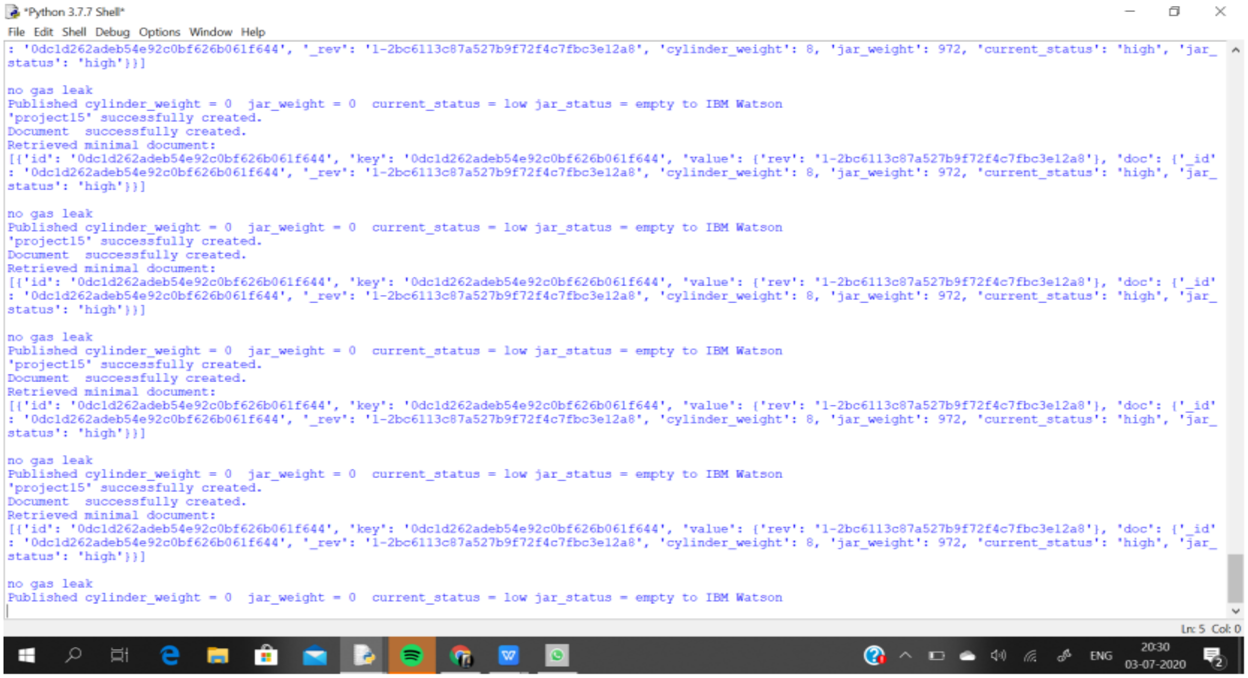
Message 7530015671:

Hide Mobile No. 11/1 SMS

FEATURES

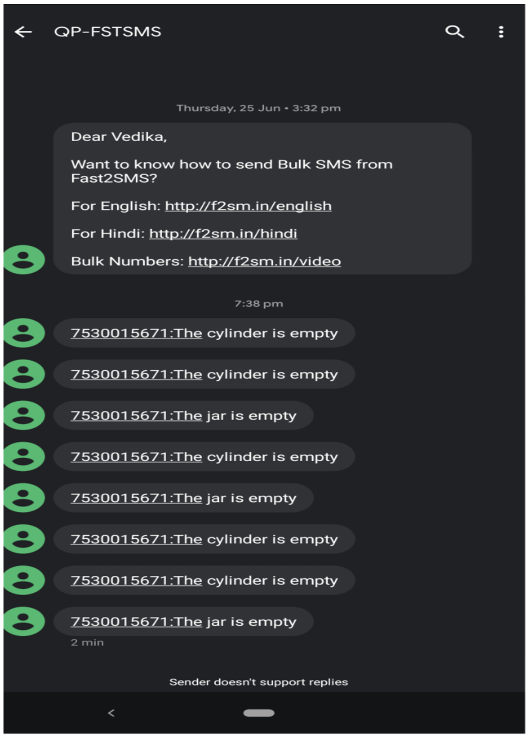
- Send Unlimited Promotional SMS
- Only for Non DND numbers
- Instant SMS Delivery (in few sec)
- Get Delivery Report Instantly
- Use your own Custom Sender ID
- Import bulk contacts from list & QR

12.THE PYTHON IDE OUTPUT:-



```
: '0dc1d262adeb54e92c0bf626b061f644', '_rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8', 'cylinder_weight': 8, 'jar_weight': 972, 'current_status': 'high', 'jar_
status': 'high'}}]

no gas leak
Published cylinder_weight = 0  jar_weight = 0  current_status = low jar_status = empty to IBM Watson
'project15' successfully created.
Document  successfully created.
Retrieved minimal document:
[{'id': '0dc1d262adeb54e92c0bf626b061f644', 'key': '0dc1d262adeb54e92c0bf626b061f644', 'value': {'rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8'}, 'doc': {'_id'
: '0dc1d262adeb54e92c0bf626b061f644', '_rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8', 'cylinder_weight': 8, 'jar_weight': 972, 'current_status': 'high', 'jar_
status': 'high'}}]

no gas leak
Published cylinder_weight = 0  jar_weight = 0  current_status = low jar_status = empty to IBM Watson
'project15' successfully created.
Document  successfully created.
Retrieved minimal document:
[{'id': '0dc1d262adeb54e92c0bf626b061f644', 'key': '0dc1d262adeb54e92c0bf626b061f644', 'value': {'rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8'}, 'doc': {'_id'
: '0dc1d262adeb54e92c0bf626b061f644', '_rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8', 'cylinder_weight': 8, 'jar_weight': 972, 'current_status': 'high', 'jar_
status': 'high'}}]

no gas leak
Published cylinder_weight = 0  jar_weight = 0  current_status = low jar_status = empty to IBM Watson
'project15' successfully created.
Document  successfully created.
Retrieved minimal document:
[{'id': '0dc1d262adeb54e92c0bf626b061f644', 'key': '0dc1d262adeb54e92c0bf626b061f644', 'value': {'rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8'}, 'doc': {'_id'
: '0dc1d262adeb54e92c0bf626b061f644', '_rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8', 'cylinder_weight': 8, 'jar_weight': 972, 'current_status': 'high', 'jar_
status': 'high'}}]

no gas leak
Published cylinder_weight = 0  jar_weight = 0  current_status = low jar_status = empty to IBM Watson
'project15' successfully created.
Document  successfully created.
Retrieved minimal document:
[{'id': '0dc1d262adeb54e92c0bf626b061f644', 'key': '0dc1d262adeb54e92c0bf626b061f644', 'value': {'rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8'}, 'doc': {'_id'
: '0dc1d262adeb54e92c0bf626b061f644', '_rev': '1-2bc6113c87a527b9f72f4c7fbc3e12a8', 'cylinder_weight': 8, 'jar_weight': 972, 'current_status': 'high', 'jar_
status': 'high'}}]

no gas leak
Published cylinder_weight = 0  jar_weight = 0  current_status = low jar_status = empty to IBM Watson
```

THE MESSAGE OUTPUT:-



Thursday, 25 Jun • 3:32 pm

Dear Vedika,

Want to know how to send Bulk SMS from Fast2SMS?

For English: http://f2sm.in/english

For Hindi: http://f2sm.in/hindi

Bulk Numbers: http://f2sm.in/video

7:38 pm

7530015671:The cylinder is empty

7530015671:The cylinder is empty

7530015671:The jar is empty

7530015671:The cylinder is empty

7530015671:The jar is empty

7530015671:The cylinder is empty

7530015671:The cylinder is empty

7530015671:The jar is empty

2 min

Sender doesn't support replies

13.Advantages and dissadvantages:-

Advantages :-

1. Smart appliances can make life easier
2. They look cool
3. Convenience
4. Less worry when your in a vacation
5. cooking becomes easier
6. Saftey

Disadvantages :-

1. Smart appliances cost more
2. Utilizes resource to build
3. More technical security threats

14. Conclusion:-

Our system will detect the leakage of the gas, incase there is any leakage it will send a sms to the owner and it will turn on exhaust fan on. The system will continuously monitor the weight of the lpg gas and the weight of the jar .