

SMART KITCHEN USING I.B.M CLOUD - SMARTJAR

Overview :

This project will guide you in developing the "SMART KITCHEN" project using sensor, Fast2sms, IBM cloud services and MIT app inventor under the IoT Platform.

Purpose :

To give the characteristic data about certain features and their values related to the smart jar using sensor and thereby alerting the user through Fast2sms and the mobile application whenever required.

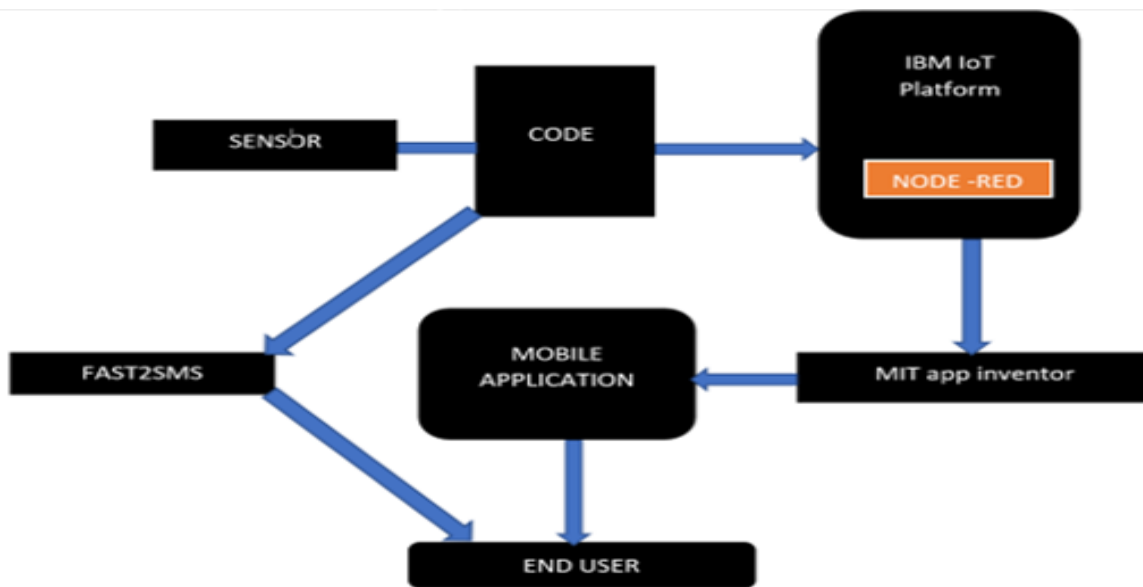
Existing Problem :

Leakage is the major problem when it comes to the Jar usage. Users are facing a problem as they do not have an easy and smart system which would help them in assessing this and take action.

Proposed Solution :

Using the fast2sms and the mobile application, we alert the user about the leakage (or) the low level of the jar and thereby avoiding the wastage of materials in kitchen (or) giving the knowledge of availability of materials in the kitchen.

Block Diagram:



Hardware/Software Required :

I.B.M :

Acts as a platform for using various services and softwares.
Helps us to store,collect and retrieve data when required.

Node-Red :

It is a software in I.B.M platform which has "Nodes" as the basic building blocks for showing the project related flows.It also has the "UI" option in it.

Python (IDLE) :

It is used for the running all the codes related to this project.

Sensor :

Used for taking the values which are required for us, which are further sent to cloud level.

Fast2Sms :

It helps us in sending the alerts/messages to multiple users at same instance of time.

MIT app inventor :

This helps us in building the model apk's/applications to simulate the data from IBM platform. Blocks act as a backbone in this platform.

Apk/Application created :

It displays the data to the users and thereby alerts and helps them in saving the wastage / leakage.

EXPERIMENTAL INVESTIGATIONS AND THE PROJECT :

IBM is a platform which is used to create several services and softwares. It helps us to store the data and retrieve it whenever required. Initially values (random-here) are taken from the device level through the python code .These details are sent to the IBM

IoT platform. These values can be observed in the "recent events" section in the device data of IBM IoT platform. Flows related to the project can be shown in the node-red platform. UI button nodes can also be used to create ui buttons which can be used to display data. Also http request can be used for creating an "**URL**" with the required data, which can be used later in MIT app inventor. MIT app inventor can be used to create apk/application related to this project. It has a designer section and a block section. Data can be visualized in the application created. Alerts/messages can be sent to the user regarding the jar status. Hence, ensuring the "avoiding of leakage and justified usage of groceries."

0 Simulations running

```
Python 3.8.5 Shell
File Edit Shell Debug Options Window Help
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:43:08) [MSC v.1926 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Shash/OneDrive/Desktop/Smart_Jar.py =====
2020-08-11 11:24:28,321 abmiotf.device.Client INFO Connected successfully: d:dlw8bs:Myjar:123456
Published Pressure = 29Pa Weight = 95g to IBM Watson
Published Pressure = 37Pa Weight = 42g to IBM Watson
Published Pressure = 24Pa Weight = 34g to IBM Watson
Published Pressure = 29Pa Weight = 80g to IBM Watson
Published Pressure = 27Pa Weight = 26g to IBM Watson
Published Pressure = 31Pa Weight = 1g to IBM Watson
{"return":true,"request_id":"Xd7e8wo2n0rp9mu","message":["Message sent successfully to NonDND numbers"]}
Published Pressure = 32Pa Weight = 71g to IBM Watson
Published Pressure = 24Pa Weight = 48g to IBM Watson
Published Pressure = 15Pa Weight = 37g to IBM Watson
Published Pressure = 38Pa Weight = 1g to IBM Watson
{"return":true,"request_id":"71lh9gducpctojf","message":["Message sent successfully to NonDND numbers"]}
Published Pressure = 35Pa Weight = 33g to IBM Watson
Published Pressure = 33Pa Weight = 69g to IBM Watson
Published Pressure = 38Pa Weight = 41g to IBM Watson
Published Pressure = 30Pa Weight = 32g to IBM Watson
|
Ln: 5 Col: 0
```

Student Dashboard | IISPS_INT_35 | SmartPractice | https://node- | Node-RED | MIT App Inventor | IBM Watson | IBM Cloud | +

Not secure | ai2.appinventor.mit.edu/#5284498690670592

Apps | YouTube | Maps | News | Translate | Other bookmarks

MIT APP INVENTOR

My Projects | Connect | Build | Settings | Help | My Projects | View Trash | Gallery | Guide | Report an Issue | English | smrm810@gmail.com

Sensordata | Screen1 | Add Screen | Remove Screen | Designer | Blocks

Blocks

- Built-in
 - Control
 - Logic
 - Math
 - Text
 - Lists
 - Dictionaries
 - Colors
 - Variables
 - Procedures
- Screen1
 - Label1
 - TextBox1
 - TextBox4
 - Label2
 - TextBox2
 - TextBox3
 - Web1

Media

Upload File ...

Viewer

when Clock1.Timer

do

- set Web1.Url to https://node-red-vitxl-2020-08-07.eu-gb.mybluemix.net
- call Web1.Get

when Web1.GotText

url responseCode responseType responseContent

do

- set TextBox1.Text to look up in pairs key Pressure pairs call Web1.JsonTextDecode jsonText get responseContent
- set TextBox2.Text to look up in pairs key Weight pairs call Web1.JsonTextDecode jsonText get responseContent

if TextBox2.Text ≤ 5

then set TextBox3.Text to Time to refill the container.

else set TextBox3.Text to The container is not empty yet.

if TextBox1.Text ≤ 15

then set TextBox4.Text to There is a leakage in the container.

else set TextBox4.Text to The container is intact.

Show Warnings

Download audio from this page

Privacy Policy and Terms of Use

3:50 PM 8/11/2020

Student Dashboard x IISPS_INT_35 x SmartPractice x https://node- x Node-RED: x MIT App Inv x IBM Watson x IBM Cloud x +

node-red-vitd-2020-08-07.eu-gb.mybluemix.net/red/#flow/ee4dc004.3ff7f

Apps YouTube Maps News Translate Other bookmarks

Node-RED Deploy

filter nodes

Flow 1

common

- inject
- debug
- complete
- catch
- status
- link in
- link out
- comment

function

- function
- switch
- change
- range
- template
- delay
- trigger
- OpenWhisk
- the

IBM IoT connected

Pressure

Weight

msg.payload

Sensordata

function

http

info

Search flows

Flows

- Flow 1
- Subflows
- Global Configuration Nodes

Flow 1

Flow "ee4dc004.3ff7f"

Import a flow by dragging its JSON into the editor, or with `ctrl-i`

3:50 PM 8/11/2020

Student Dashboard x IISPS_INT_35 x SmartPractice x https://node x Node-RED: x MIT App Inv x IBM Watson x IBM Cloud x + -

node-red-vitbd-2020-08-07.eu-gb.mybluemix.net/sensordata

Apps YouTube Maps News Translate Other bookmarks

```
{"Pressure":15,"Weight":82}
```

Windows taskbar showing search, task view, and application icons (Edge, File Explorer, VLC, Chrome, PowerPoint, Teams, Word, Recorder).

System tray: 3:51 PM, 8/11/2020, ENG, network, volume, and notification icons.

11:24

17%

Screen1

Pressure

37

The container is
intact

Weight

42

The container is not
empty yet.