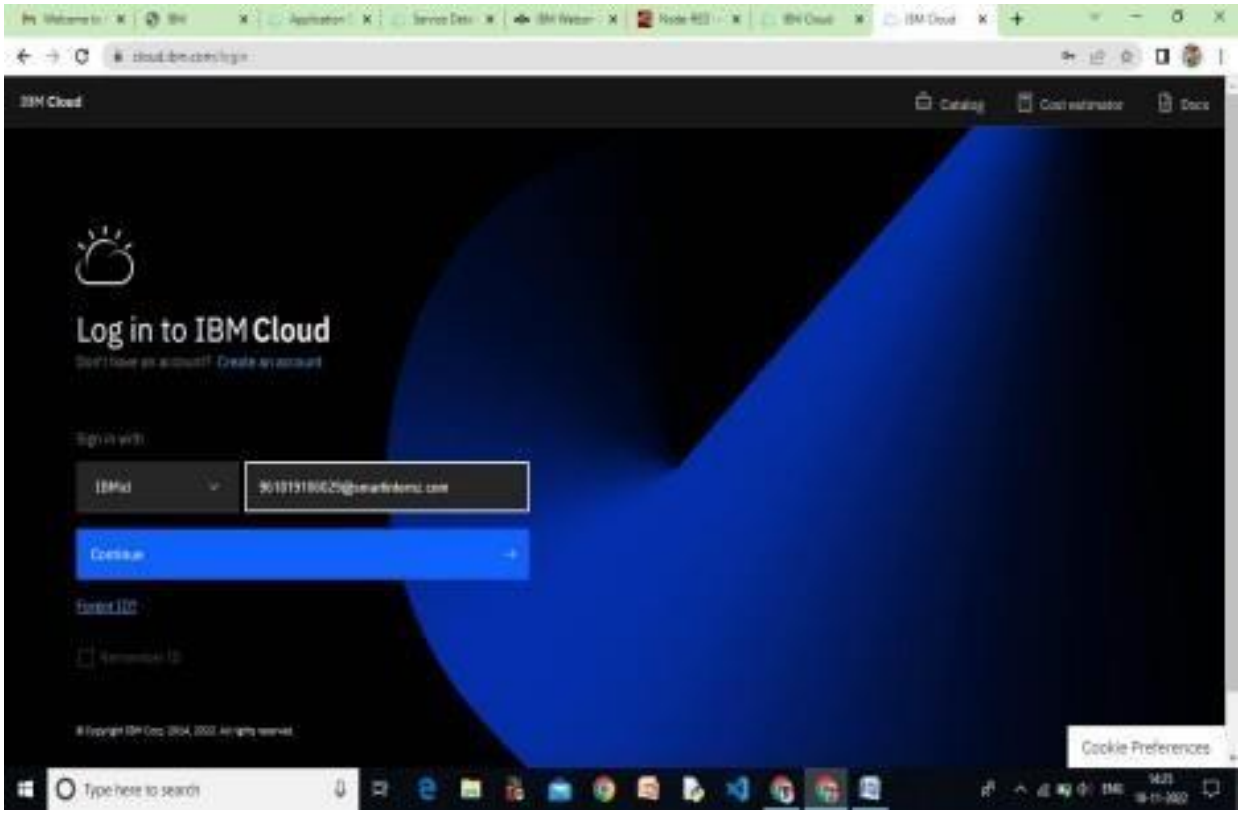
TEAM ID	**PNT2022TMID31677**
PROJECT NAME	**SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY**

RESOURCES PAGES

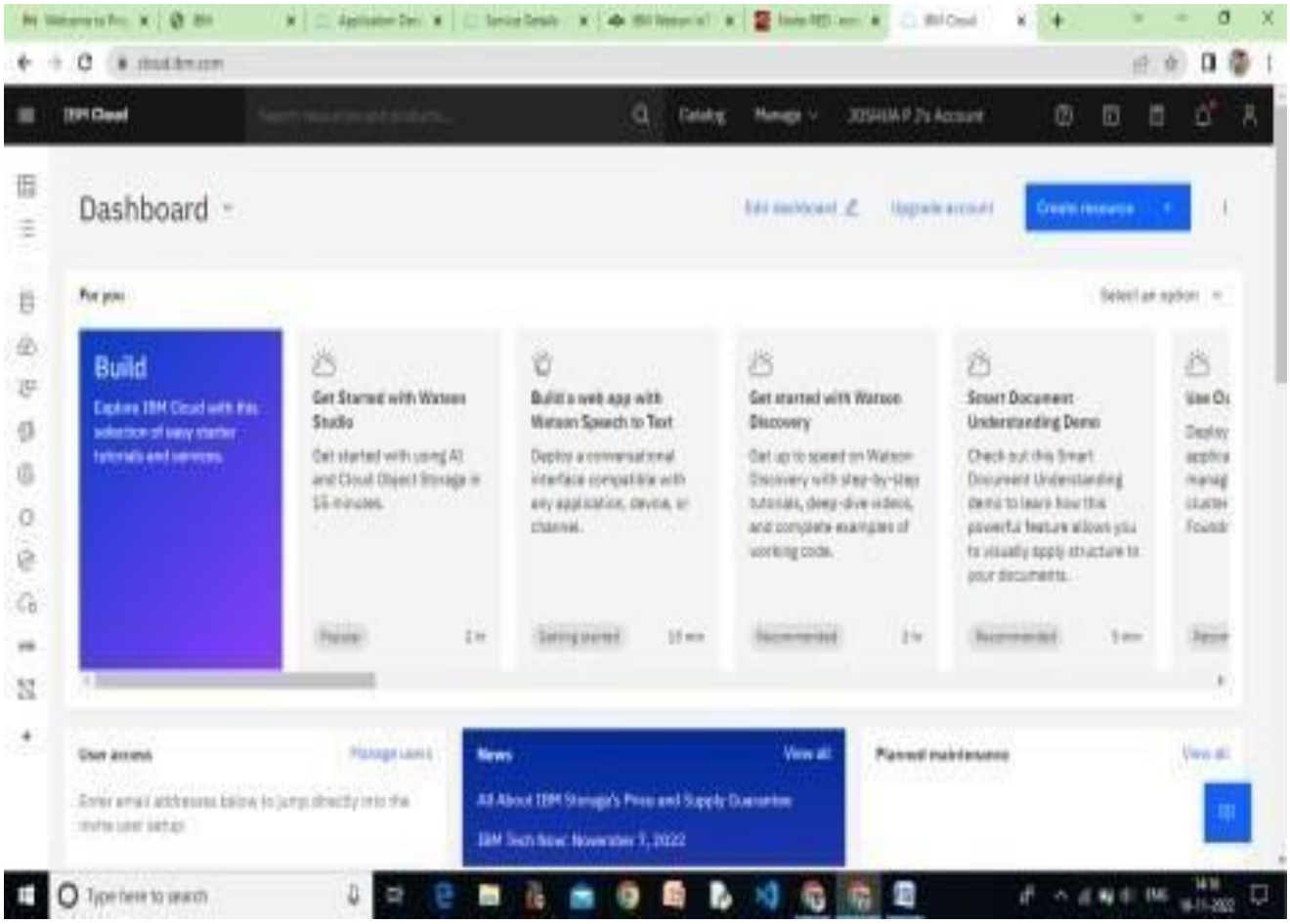
IBM CLOUD:

LOGIN TO THE WEBSITE: IBM CLOUD LOGIN

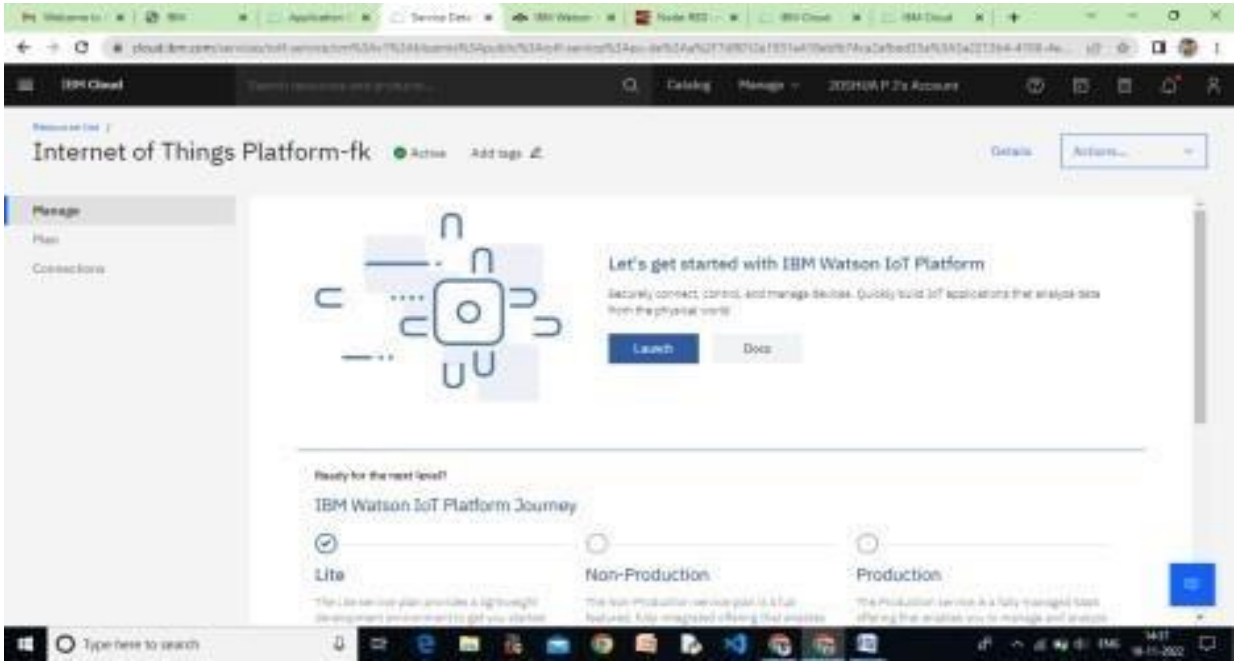
1. Enter the SI Mail id and create your password.



2. After Login page dashboard will be displayed and click the catalog and open internet of things.

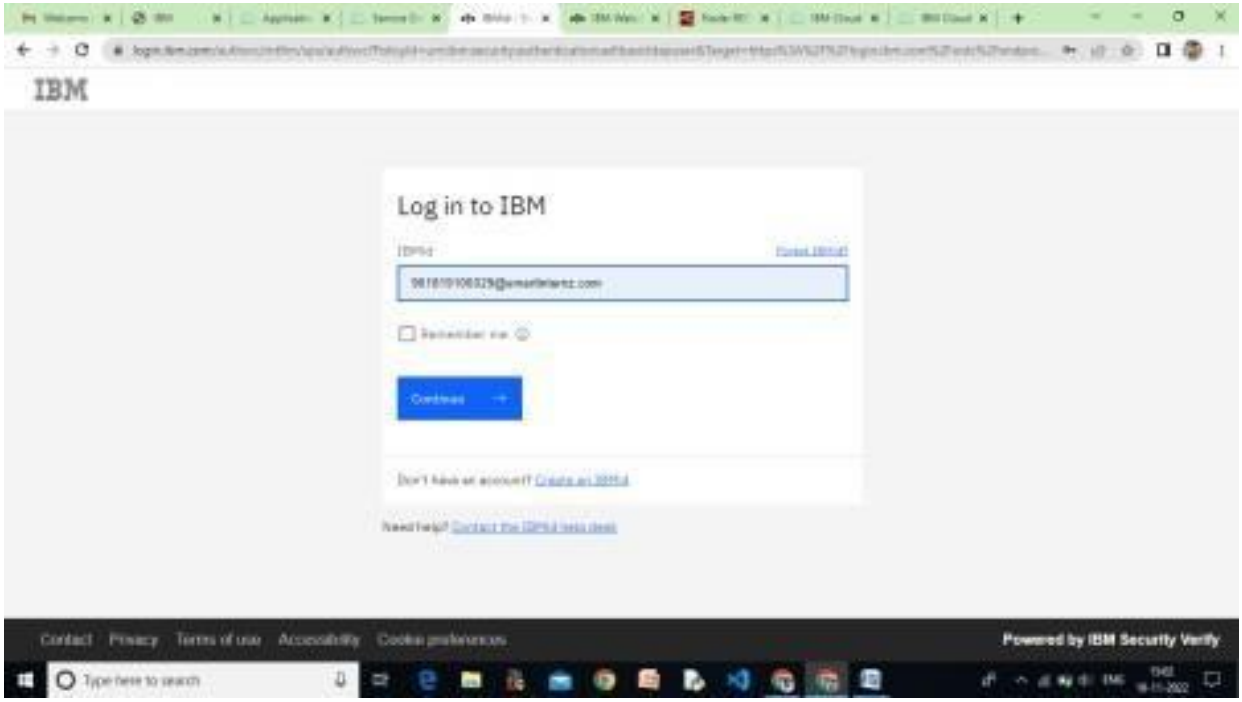


3. After that you will find this page and click the launch button:

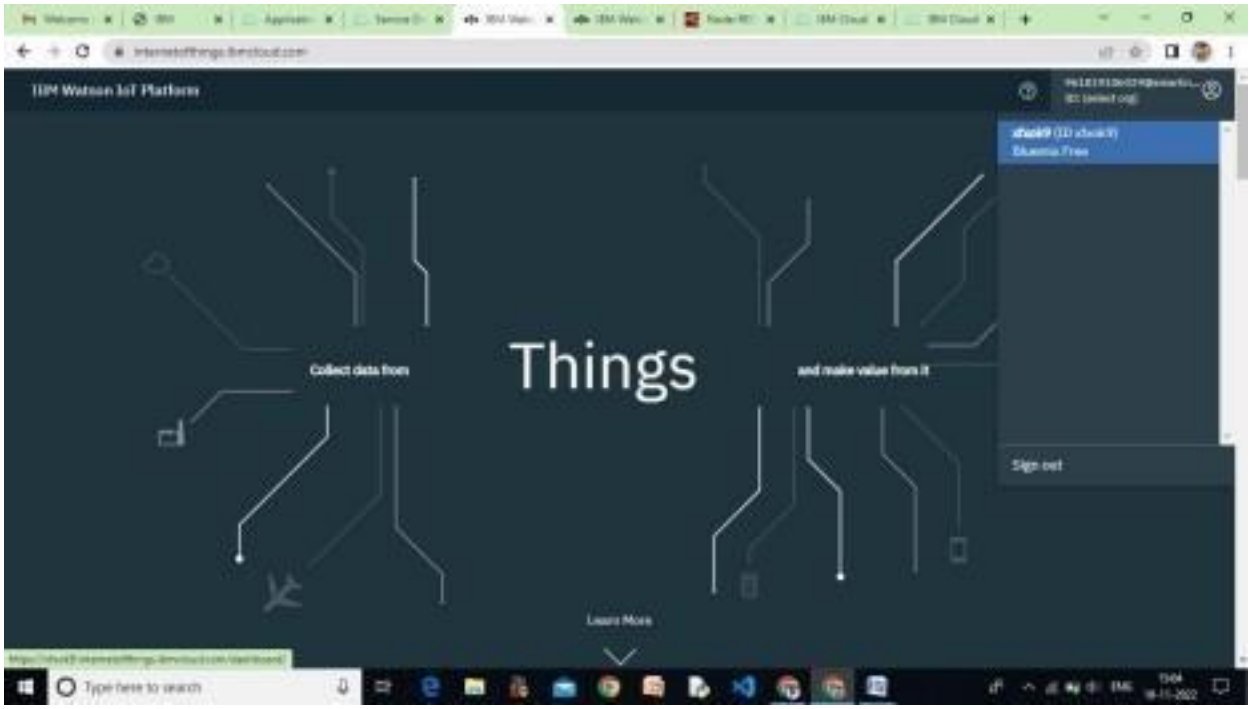


TO LOGIN TO IBM WATSON:

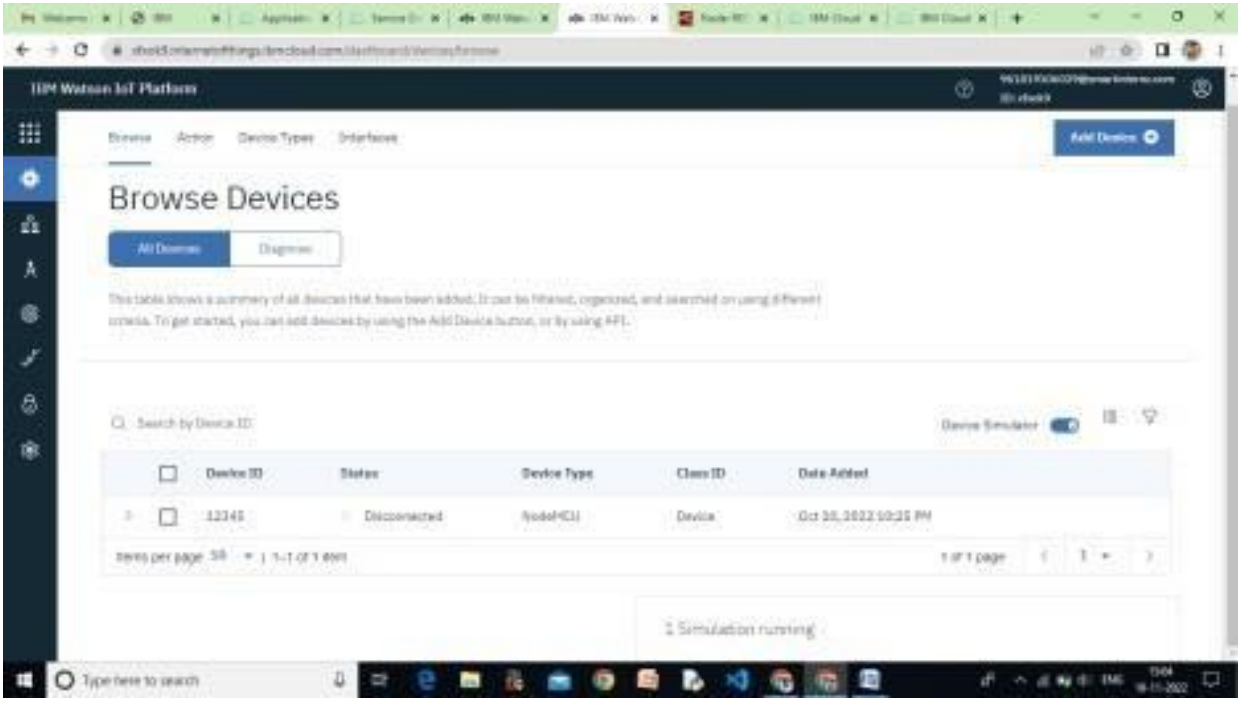
1. After that log in to the Page and give your SI Mail id and password



2. Then sign in the right-side corner id given in the IBM Watson:



3. After that you will find the Browse Device Page:



4. Create a Device by entering all the required details given below and click finish:

Organization ID: xfxok9

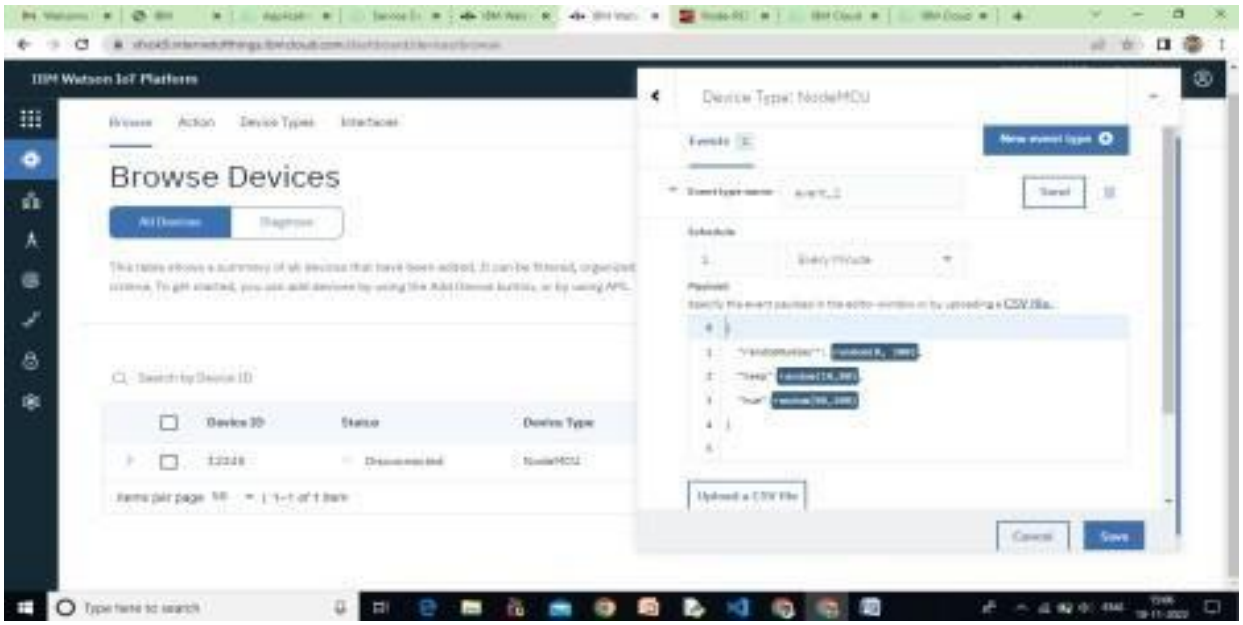
Device Type: NodeMCU

Device ID: 6385476358

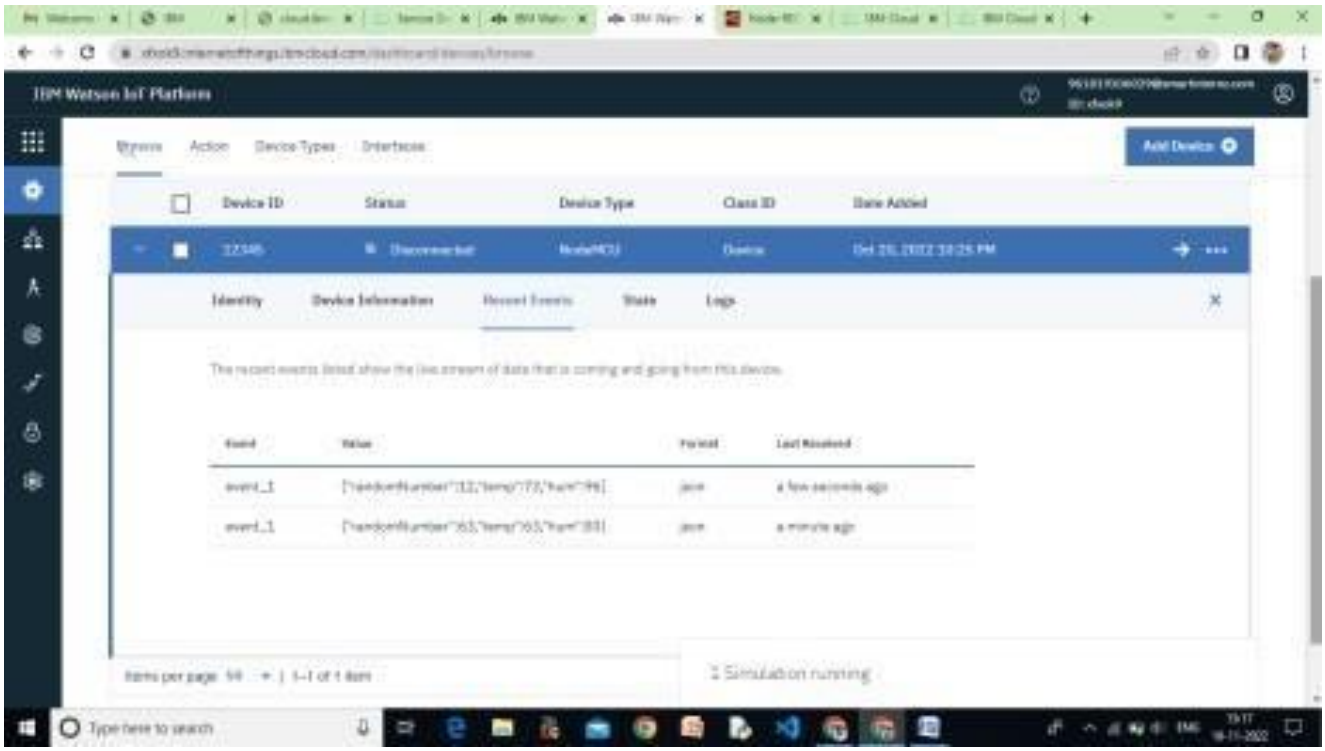
Authentication Method: use-token-auth

Authentication Token: 9384731286

5. And then you will receive this page by stimulating the created device:

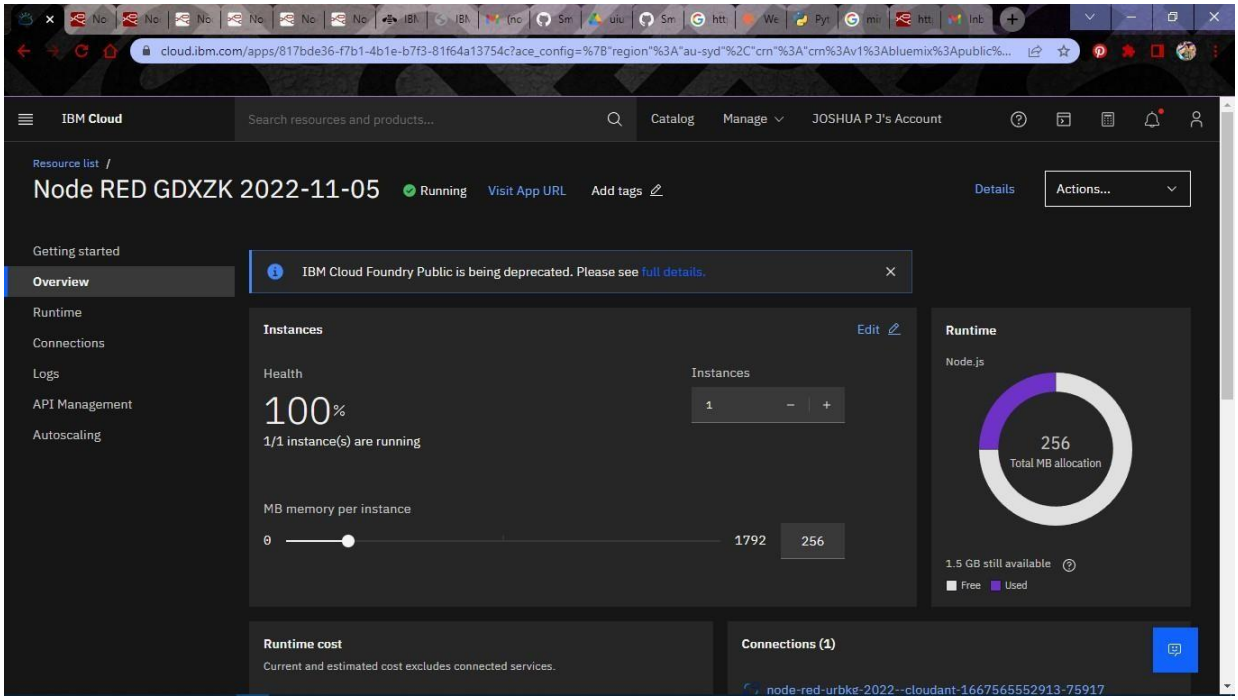


STATUS PAGE (RESULT):



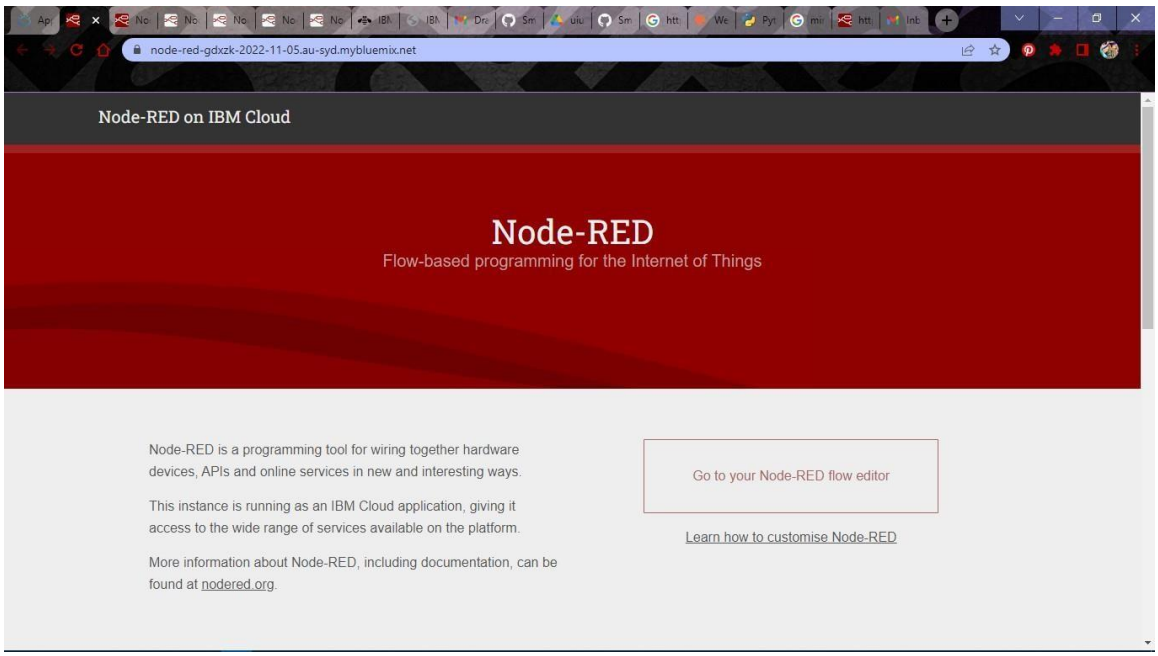
NODE RED :

- Here, click the Visit AppURL:

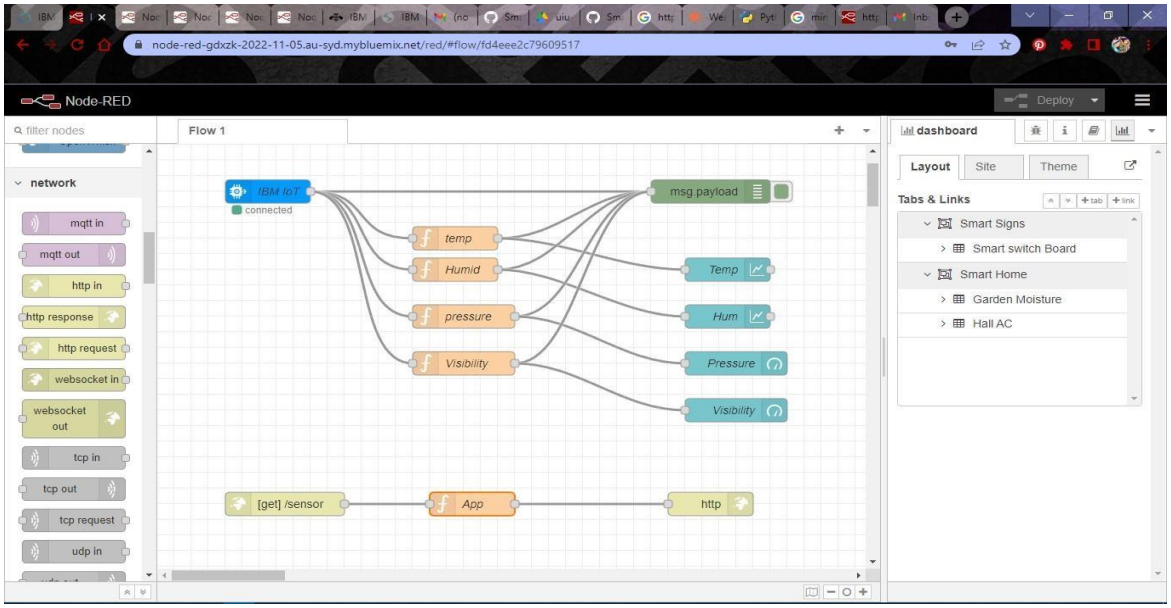


LOGIN PAGE OF NODE RED:

After logging in, we get a page like this.
Here click “Go to your NODE RED flow editor”

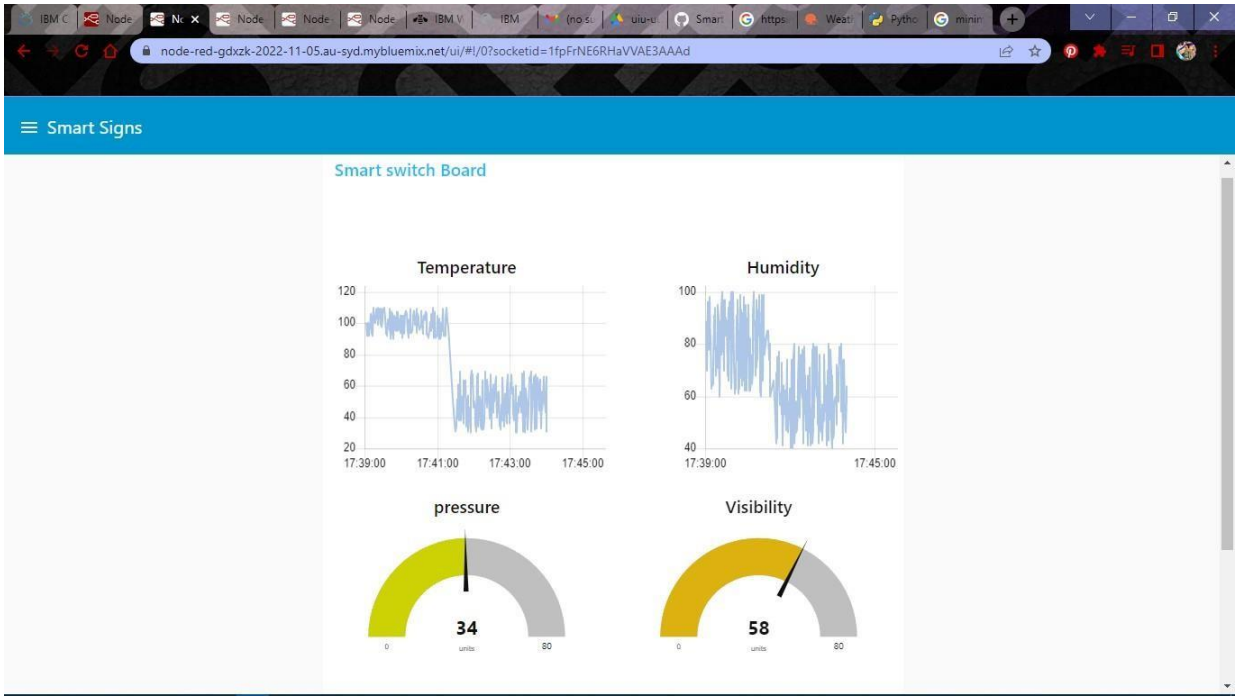


Developing Route based on the program :



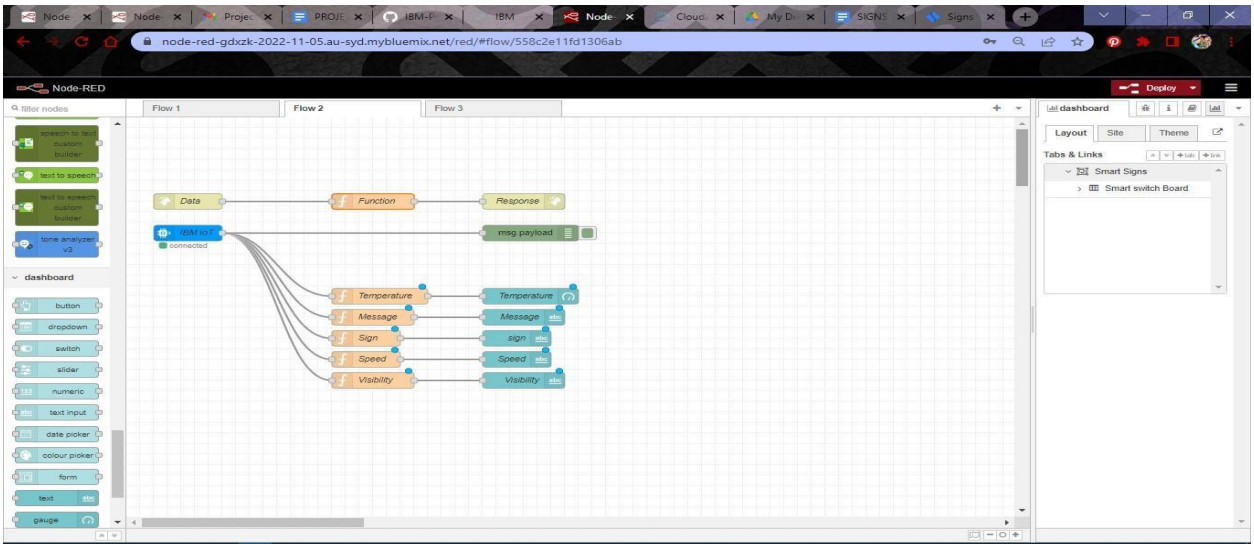
OUTPUT:

(The output will be displayed in graphical representation)



DEVELOPING ROUTE BASED ON THE PROGRAM:

Here based on the project the routing is developed by using appropriate nodes.



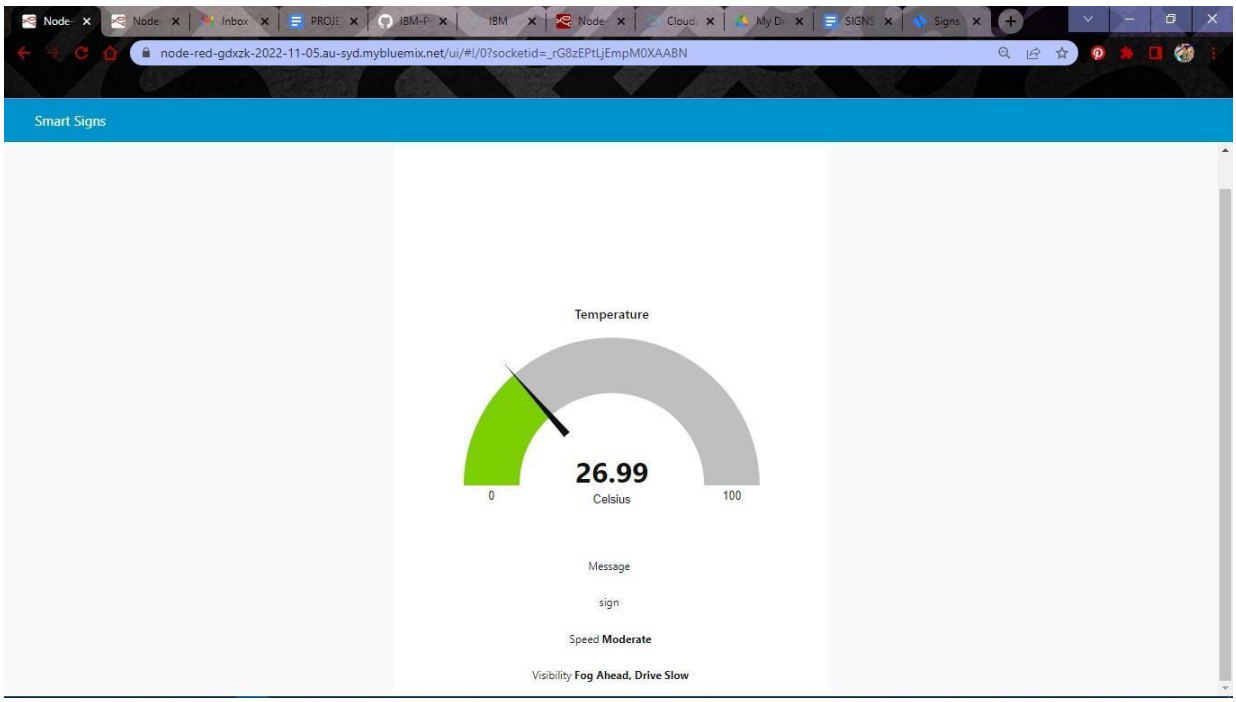
Reference link:
https://node-red-gdxzk-2022-11-05.au-syd.mybluemix.net/red/#flow/558c2e11fd1306ab

OUTPUT FOR NODE RED:

After making the proper connection between nodes the deploy button is enabled and the result is displayed on the node-red dashboard.

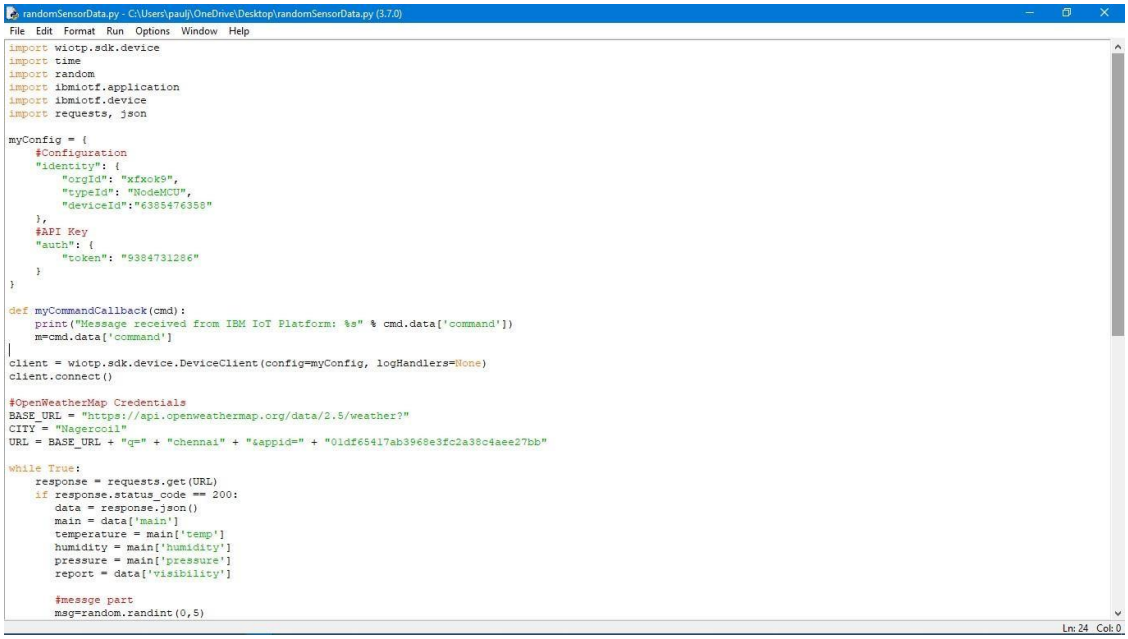
Reference link:
https://node-red-gdxzk-2022-11-05.au-syd.mybluemix.net/ui/#!/0?socketid=K3WuCimudoiryXYUAABD

It shows the result in a diagrammatic structure.

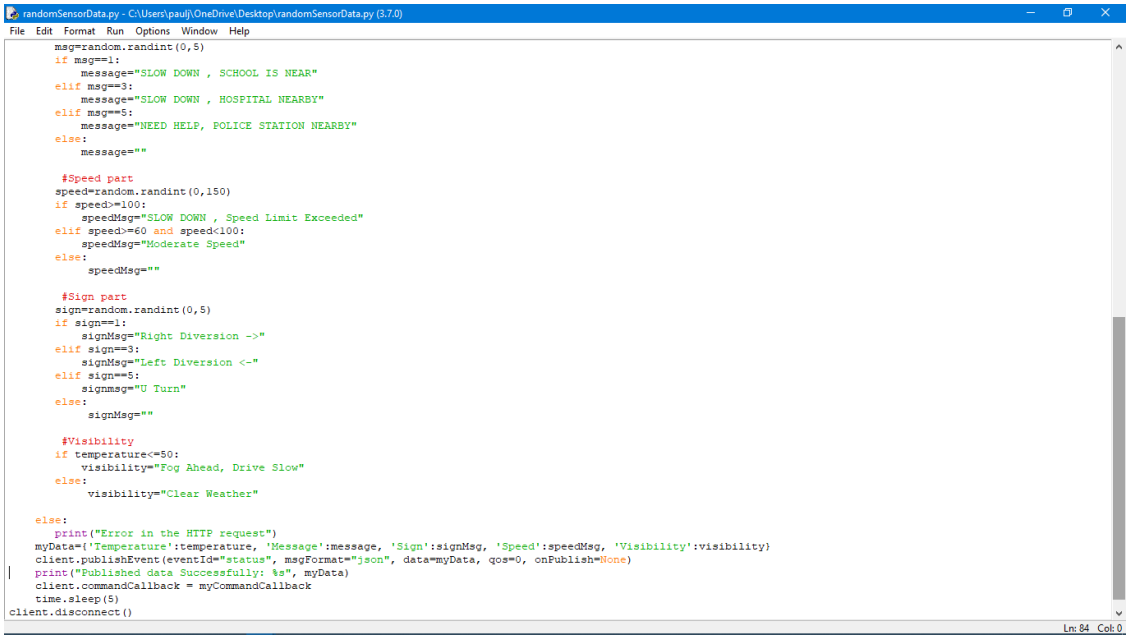


CODE IN PYTHON IDLE :

PROGRAM:



```
import wiotp.sdk.device
import time
import random
import ibmiotf.application
import ibmiotf.device
import requests, json

myConfig = {
    #Configuration
    "identity": {
        "orgId": "xfxok9",
        "typeId": "NodeMCU",
        "deviceId":"6385476358"
    },
    #API Key
    "auth": {
        "token": "9384731286"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

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

#OpenWeatherMap Credentials
BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"
CITY = "Nagercoil"
URL = BASE_URL + "q=" + "chennai" + "&appid=" + "01df65417ab3968e3fc2a38c4aee27bb"

while True:
    response = requests.get(URL)
    if response.status_code == 200:
       data = response.json()
       main = data['main']
       temperature = main['temp']
       humidity = main['humidity']
       pressure = main['pressure']
       report = data['visibility']

       #messge part
       msg=random.randint(0,5)
```



```
       msg=random.randint(0,5)
       if msg==1:
           message="SLOW DOWN , SCHOOL IS NEAR"
       elif msg==3:
           message="SLOW DOWN , HOSPITAL NEARBY"
       elif msg==5:
           message="NEED HELP, POLICE STATION NEARBY"
       else:
           message=""

        #Speed part
       speed=random.randint(0,150)
       if speed>=100:
           speedMsg="SLOW DOWN , Speed Limit Exceeded"
       elif speed>=60 and speed<100:
           speedMsg="Moderate Speed"
       else:
            speedMsg=""

        #Sign part
       sign=random.randint(0,5)
       if sign==1:
           signMsg="Right Diversion ->"
       elif sign==3:
           signMsg="Left Diversion <-"
       elif sign==5:
           signmsg="U Turn"
       else:
           signMsg=""

        #Visibility
       if temperature<=50:
           visibility="Fog Ahead, Drive Slow"
       else:
            visibility="Clear Weather"

    else:
       print("Error in the HTTP request")
    myData={'Temperature':temperature, 'Message':message, 'Sign':signMsg, 'Speed':speedMsg, 'Visibility':visibility}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(5)
client.disconnect()
```

Program used in the code:

```
import wiotp.sdk.device
import time
import random
import ibmiotf.application
import ibmiotf.device
import requests, json

myConfig = {
    #Configuration
    "identity": {
        "orgId": "xfxok9",
        "typeId": "NodeMCU",
        "deviceId":"6385476358"
    },
    #API Key
    "auth": {
        "token": "9384731286"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']

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

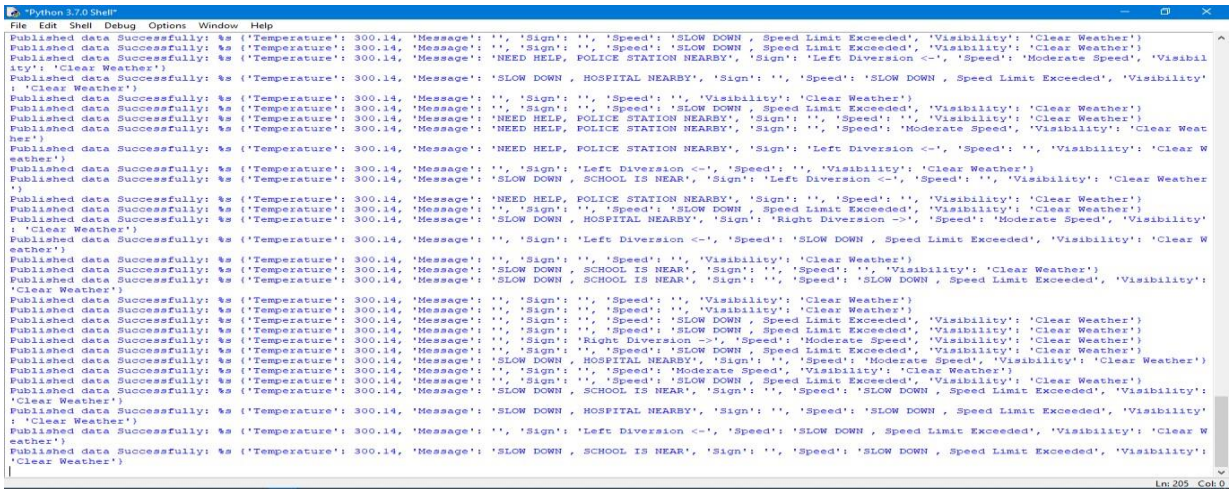
```
#OpenWeatherMap Credentials
BASE_URL =
"https://api.openweathermap.org/data/2.5/weather?"

CITY = "Nagercoil"
URL = BASE_URL + "q=" + "chennai" + "&appid=" +
"01df65417ab3968e3fc2a38c4aee27bb"

while True:
    response = requests.get(URL)
    if response.status_code == 200:
        data = response.json()
        main = data['main']
        temperature = main['temp']
        humidity = main['humidity']
        pressure = main['pressure']
        report = data['visibility']

        #messge part
        msg=random.randint(0,5)
        if msg==1:
            message="SLOW DOWN , SCHOOL IS NEAR"
        elif msg==3:
            message="SLOW DOWN , HOSPITAL NEARBY"
        elif msg==5:
            message="NEED HELP, POLICE STATION NEARBY"
        else:
            message=""

         #Speed part
        speed=random.randint(0,150)
        if speed>=100:
```

```
            speedMsg="SLOW DOWN , Speed Limit Exceeded"
        elif speed>=60 and speed<100:
            speedMsg="Moderate Speed"
        else:
             speedMsg=""
         #Sign part
        sign=random.randint(0,5)
        if sign==1:
            signMsg="Right Diversion ->"
        elif sign==3:
            signMsg="Left Diversion <-"
        elif sign==5:
             signmsg="U  Turn"
        else:
             signMsg=""

         #Visibility
        if temperature<=50:
            visibility="Fog Ahead, Drive Slow"
        else:
             visibility="Clear Weather"
    else:
        print("Error in the HTTP request")
    myData={'Temperature':temperature, 'Message':message,
'Sign':signMsg, 'Speed':speedMsg, 'Visibility':visibility}
    client.publishEvent(eventId="status", msgFormat="json",
data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(5)
```

client.disconnect()

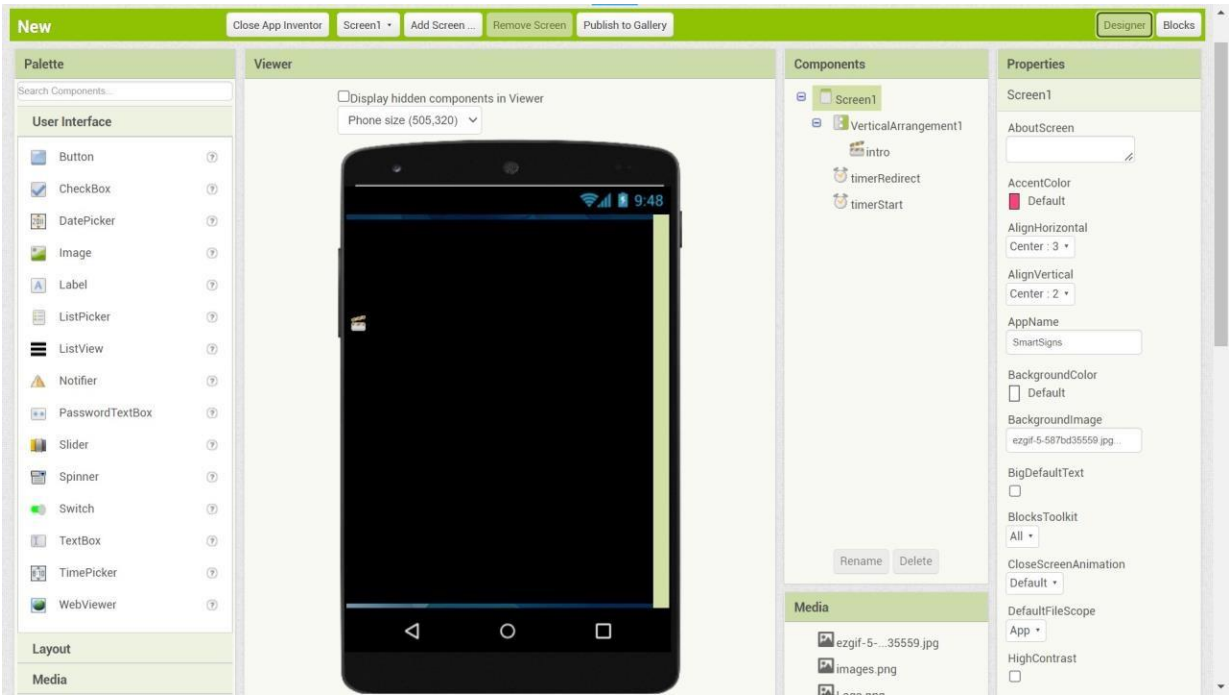
Output displayed in Python Idle:

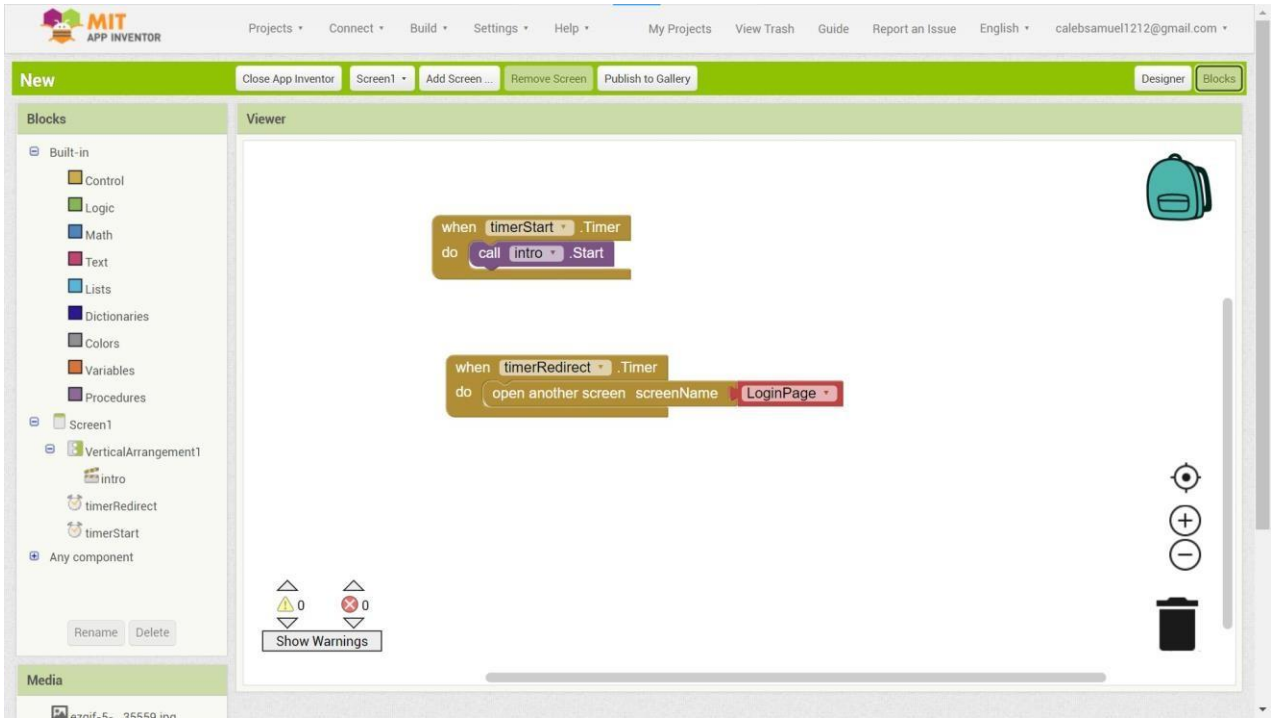
The output of the code was displayed in python idle shell mode.



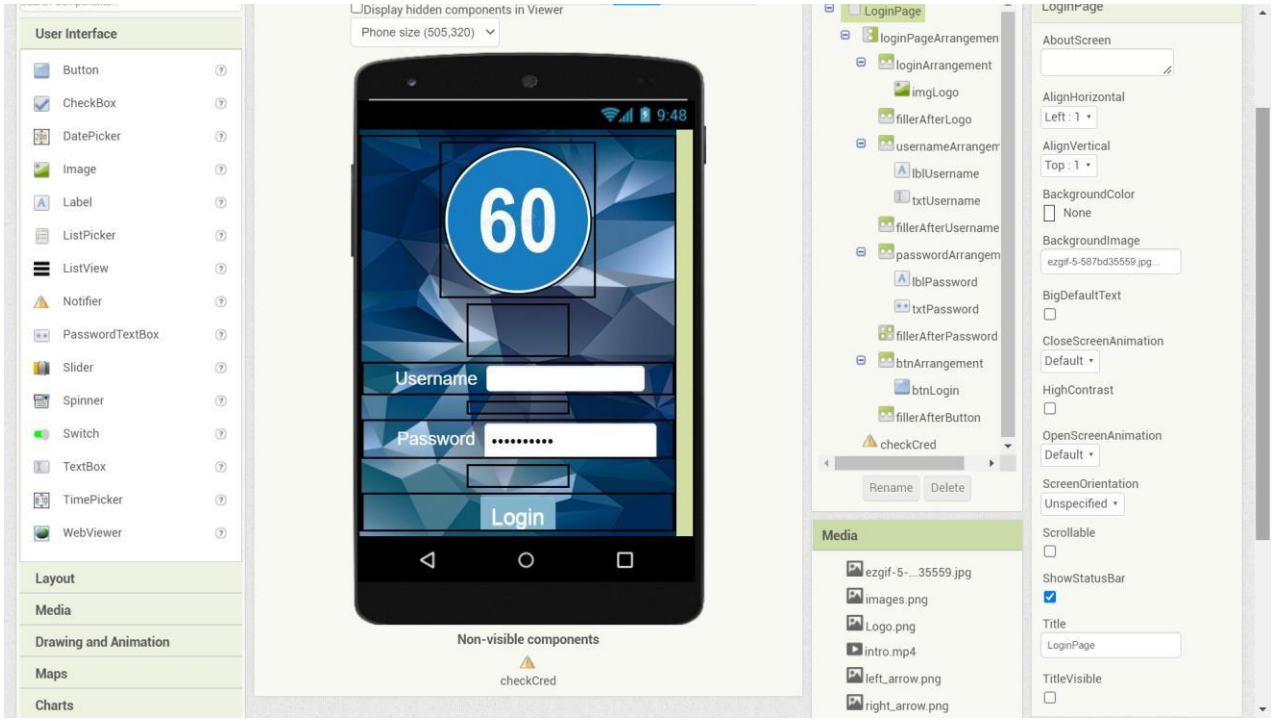
MIT APP INVENTOR:

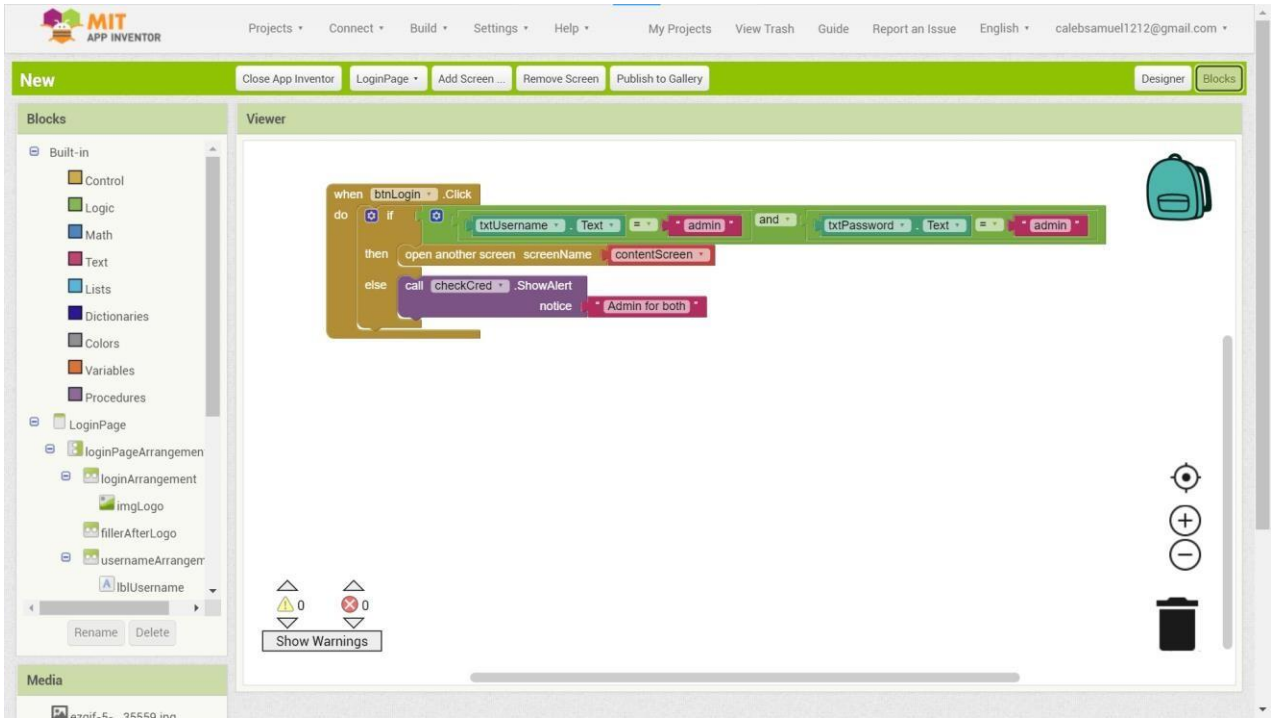
For Screen 1:



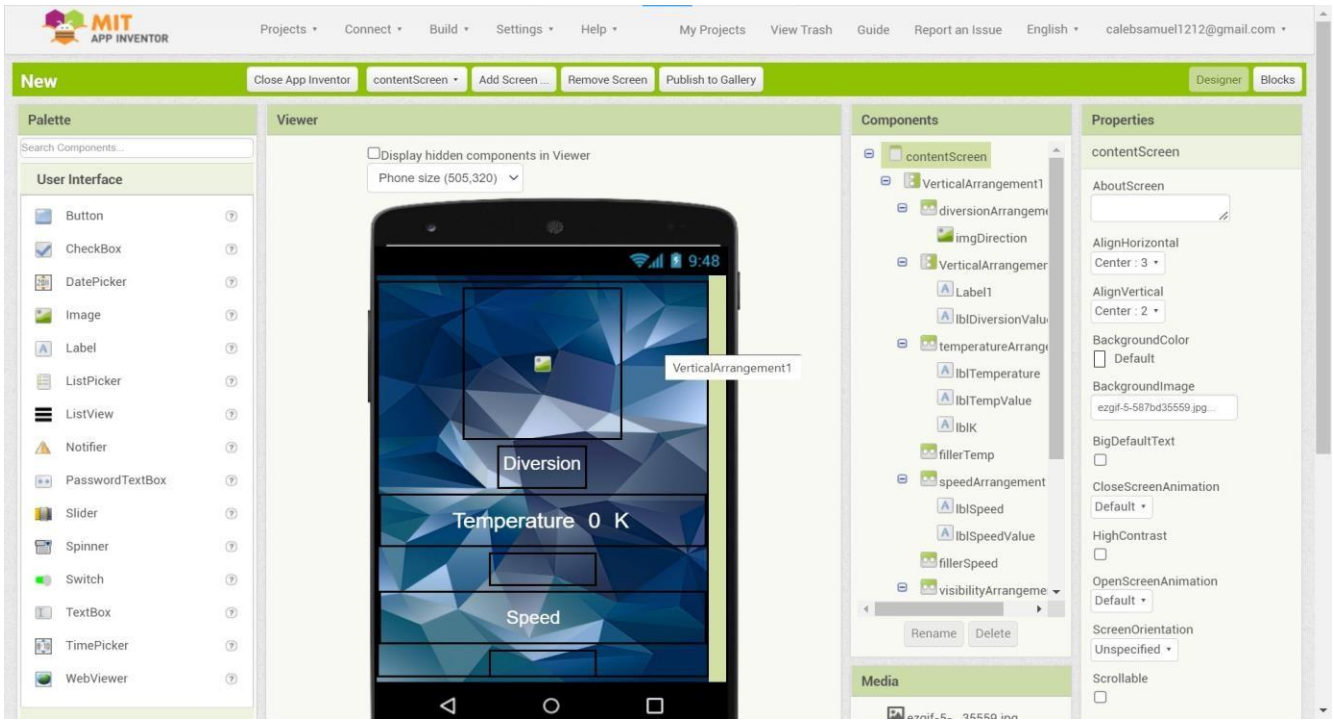

MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
calebsamuel1212@gmail.com
New
Close App Inventor
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
Screen1
VerticalArrangement1
intro
timerRedirect
timerStart
Any component
Rename
Delete
Media
Viewer
when timerStart .Timer
do call intro .Start
when timerRedirect .Timer
do open another screen screenName LoginPage
0
0
Show Warnings

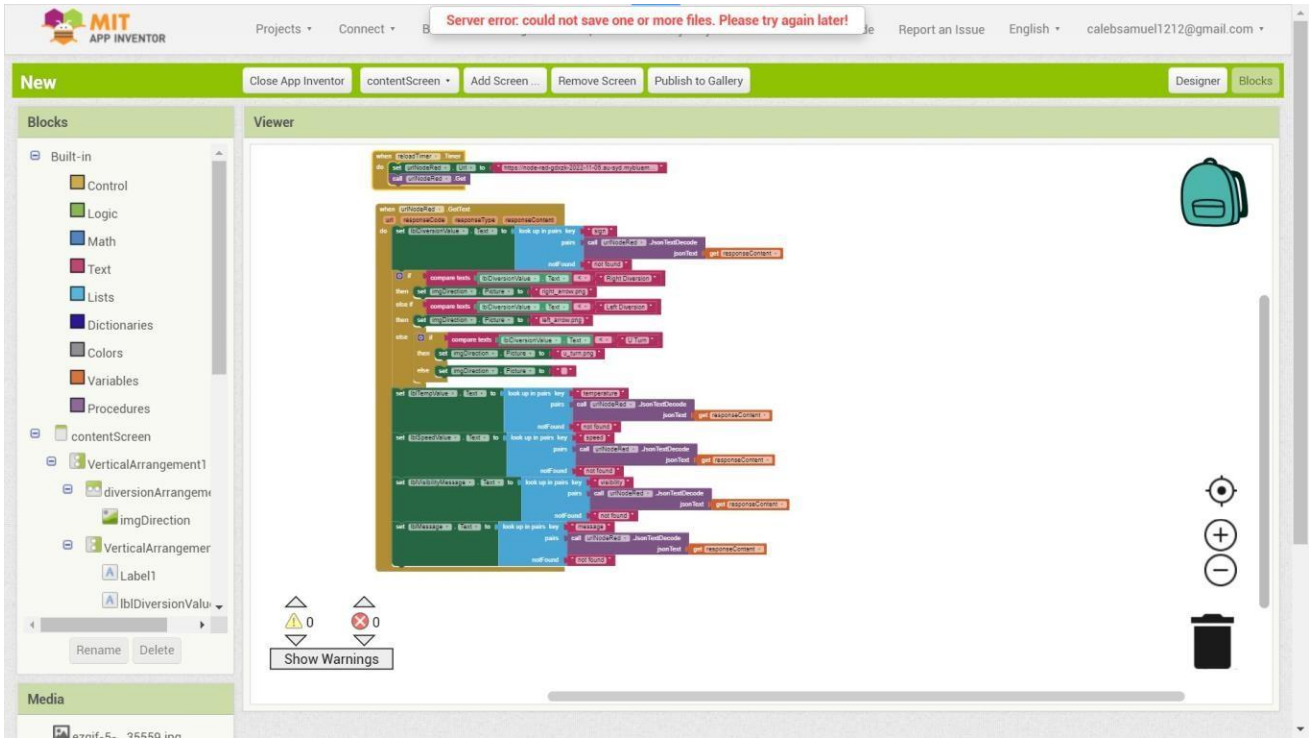
For Screen 2 :


User Interface
Button
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
Maps
Charts
Display hidden components in Viewer
Phone size (505,320)
9:48
60
Username
Password
Login
Non-visible components
checkCred
LoginPage
loginPageArrangemen
loginArrangement
imgLogo
fillerAfterLogo
usernameArrangem
lblUsername
txtUsername
fillerAfterUsername
passwordArrangem
lblPassword
txtPassword
fillerAfterPassword
btnArrangement
btnLogin
fillerAfterButton
checkCred
Rename
Delete
Media
ezgif-5-...35559.jpg
images.png
Logo.png
intro.mp4
left_arrow.png
right_arrow.png
LoginPage
AboutScreen
AlignHorizontal
Left : 1
AlignVertical
Top : 1
BackgroundColor
None
BackgroundImage
ezgif-5-587bd35559.jpg...
BigDefaultText
CloseScreenAnimation
Default
HighContrast
OpenScreenAnimation
Default
ScreenOrientation
Unspecified
Scrollable
ShowStatusBar
Title
LoginPage
TitleVisible


MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
calebsamuel1212@gmail.com
New
Close App Inventor
LoginPage
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
LoginPage
loginPageArrangemen
loginArrangement
imgLogo
fillerAfterLogo
usernameArrangem
lblUsername
Rename
Delete
Media
Viewer
when btnLogin .Click
do if
txtUsername . Text = admin and txtPassword . Text = admin
then open another screen screenName contentScreen
else call checkCred .ShowAlert
notice Admin for both
0
0
Show Warnings

For Screen 3 :


MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
calebsamuel1212@gmail.com
New
Close App Inventor
contentScreen
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Palette
Search Components...
User Interface
Button
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Viewer
Display hidden components in Viewer
Phone size (505,320)
9:48
VerticalArrangement1
Diversion
Temperature 0 K
Speed
Components
contentScreen
VerticalArrangement1
diversionArrangeme
imgDirection
VerticalArrangemer
Label1
lblDiversionValu
temperatureArrang
lblTemperature
lblTempValue
lblK
fillerTemp
speedArrangement
lblSpeed
lblSpeedValue
fillerSpeed
visibilityArrangeme
Rename
Delete
Media
Properties
contentScreen
AboutScreen
AlignHorizontal
Center : 3
AlignVertical
Center : 2
BackgroundColor
Default
BackgroundImage
ezgif-5-587bd35559.jpg...
BigDefaultText
CloseScreenAnimation
Default
HighContrast
OpenScreenAnimation
Default
ScreenOrientation
Unspecified
Scrollable


MIT APP INVENTOR
Projects
Connect
Server error: could not save one or more files. Please try again later!
Report an Issue
English
calebsamuel1212@gmail.com
New
Close App Inventor
contentScreen
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
contentScreen
VerticalArrangement1
diversionArrangeme
imgDirection
VerticalArrangemer
Label1
lblDiversionValu
Rename
Delete
Media
Viewer
0
0
Show Warnings