SPRINT 4

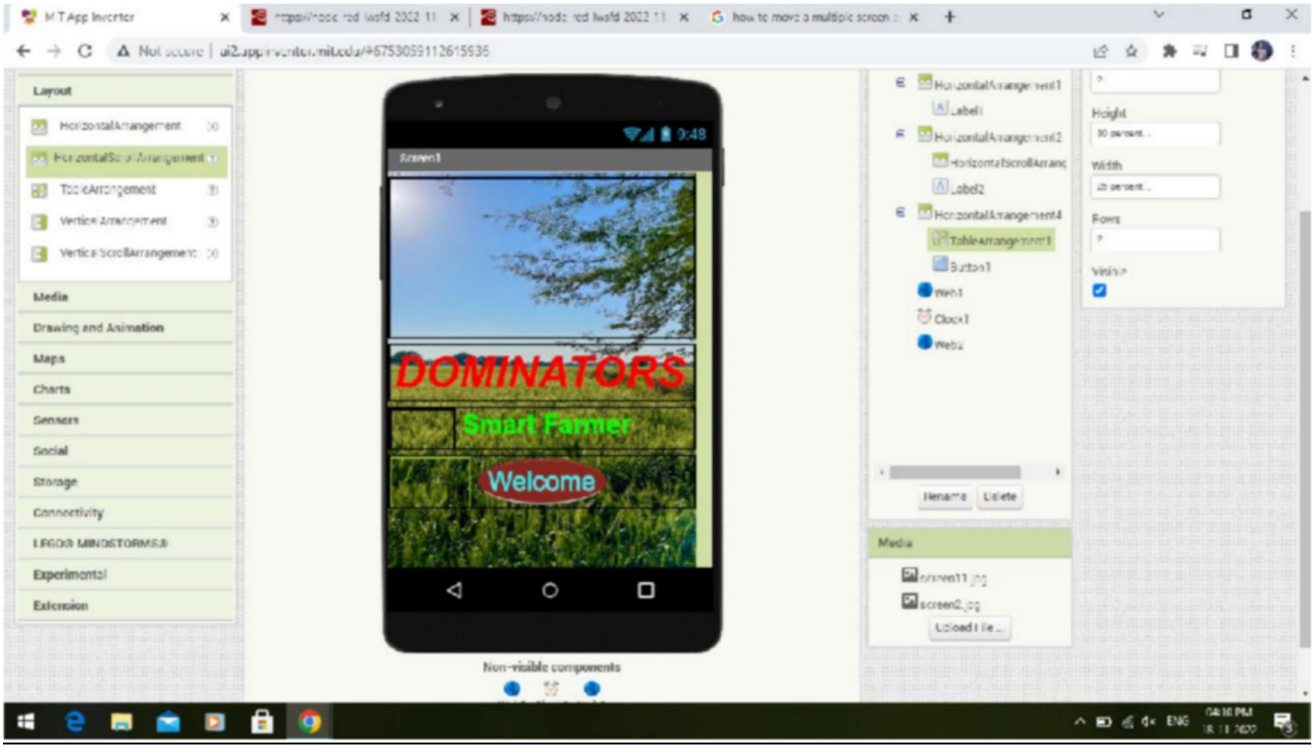
Team ID	PNT2022TMID08036
Project Name	SmartFarmer - IoT Enabled Smart Farming Application

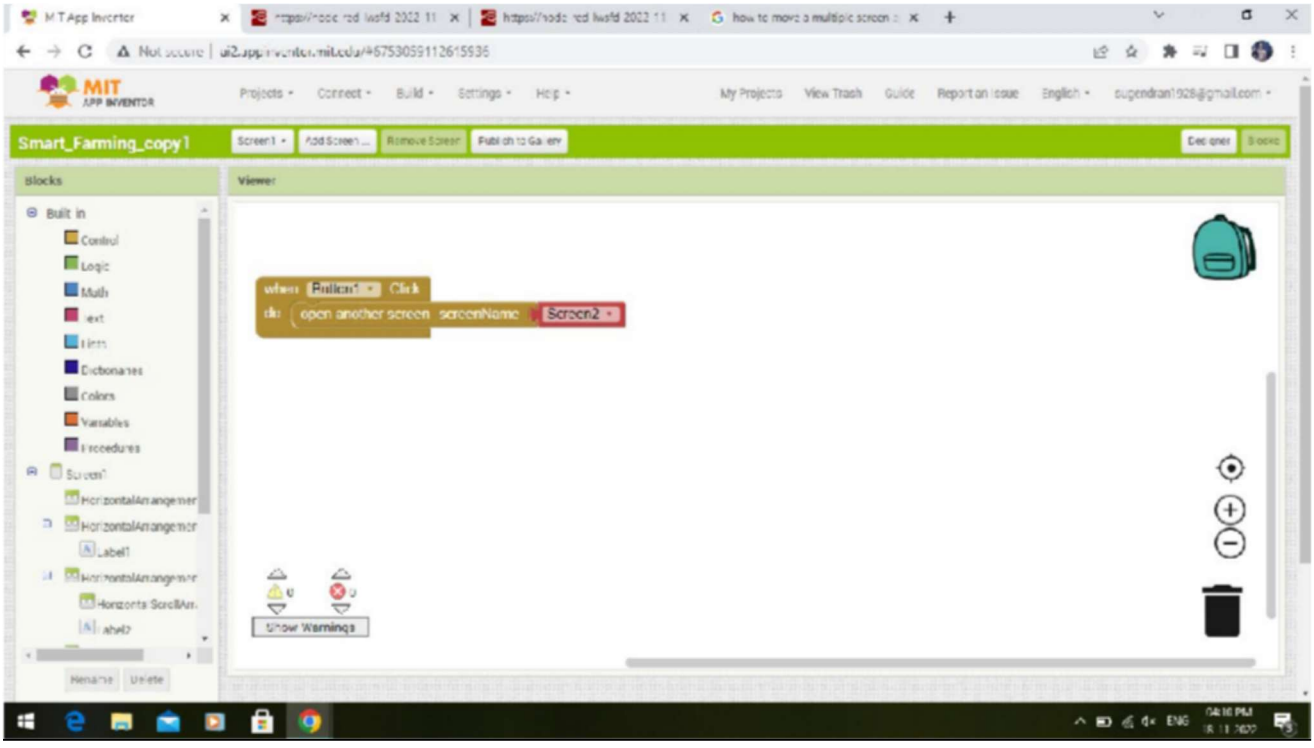
Python Script

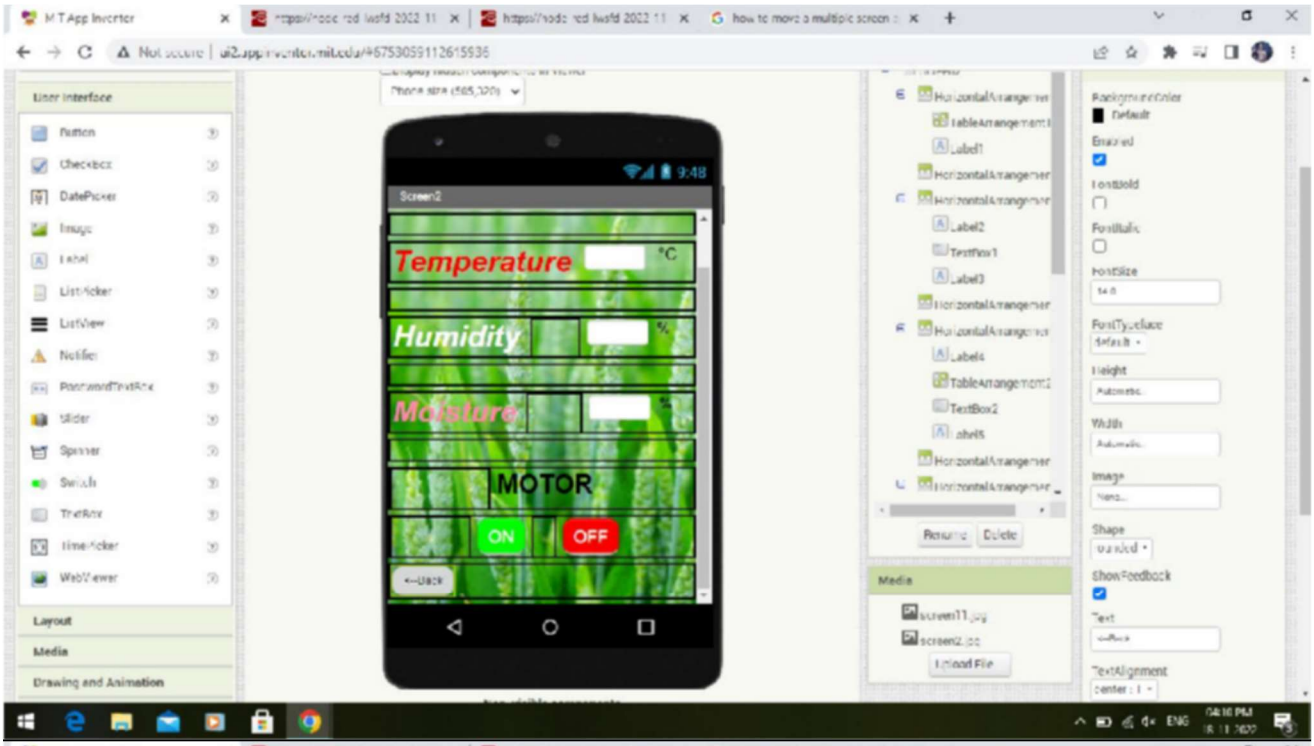
```
import time
import sys
import ibmiotf.application
import ibmiotf.device import random

organization = "1nuzg6"
deviceType = "srimathi"
deviceId = "94868"
authMethod = "token"
authToken = "987654321"

def myCommandCallback(cmd):
print("Command received: %s" % cmd.data)
for key in cmd.data.keys():
if key == 'motor':
if cmd.data['motor'] == 'ON':
print("MOTOR is turned ON")
elif cmd.data['motor'] == 'OFF':
```

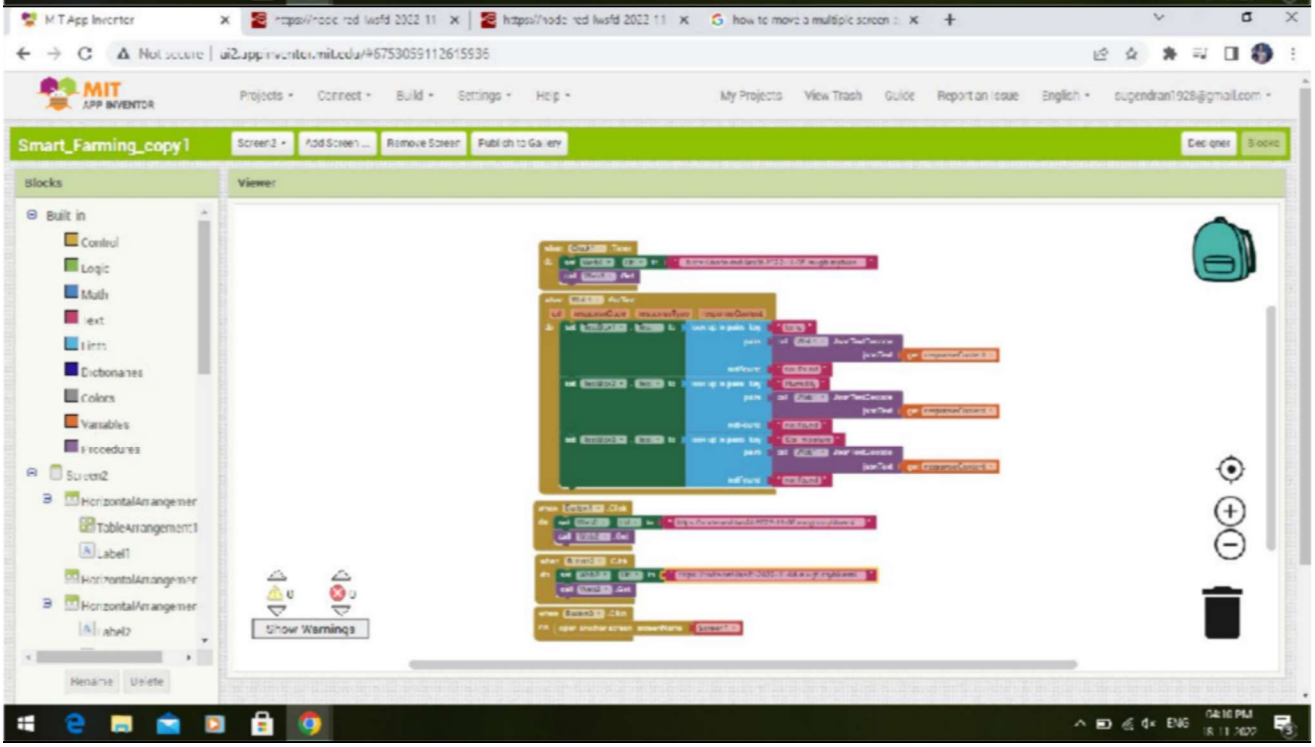
```
print("MOTOR is turned OFF")
 try:
deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
print("Caught exception connecting device: %s" % str(e))
sys.exit()
deviceCli.connect()
while True:
temp=random.randint(0,40)
Humid=random.randint(0,100)
moist=random.randint(0,40)
data = { 'temperature' : temp, 'humidity': Humid, 'soil_moisture':moist
}
def myOnPublishCallback():
print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "soil moisture =%s" % moist,"to IBM Watson")
success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
if not success:
print("Not connected to IoTF")
time.sleep(10)
deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()
```

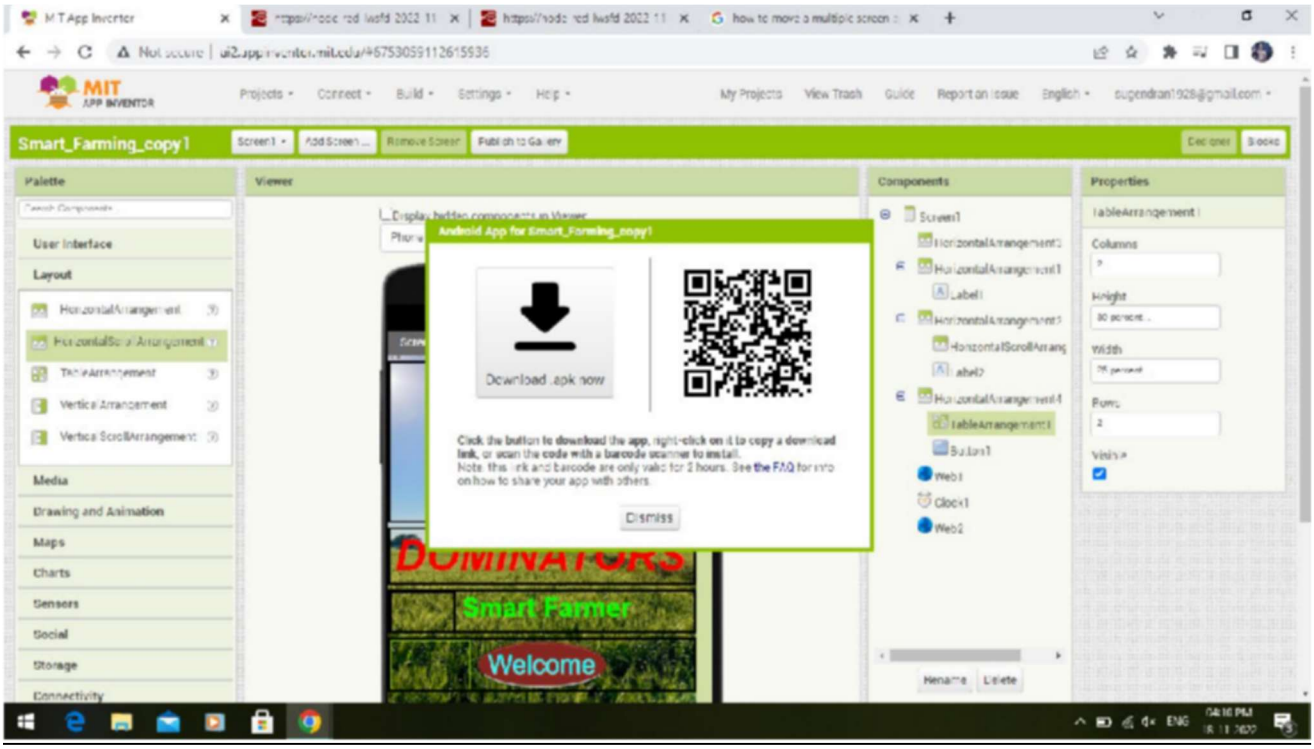
User Application or Mobile Application



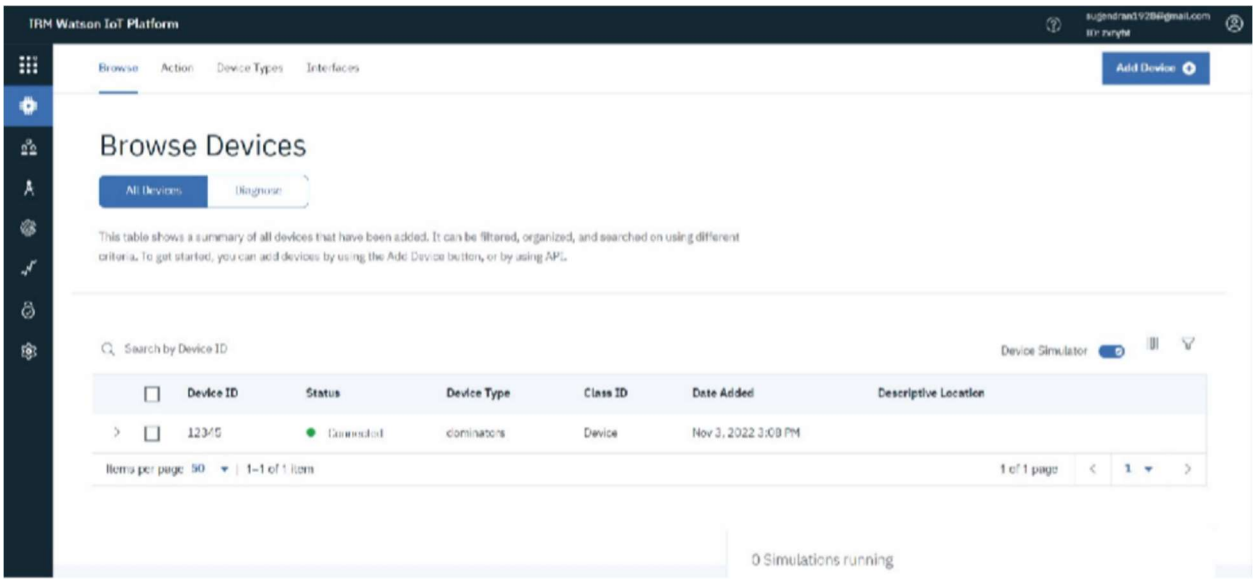



Screen2
Temperature
°C
Humidity
%
Moisture
%
MOTOR
ON
OFF
←Back


Smart_Farming_copy1
Blocks
Viewer
Show Warnings



Python Script configured to IBM Watson IoT platform



The Sensors data in the python script will be received by IBM Watson IoT platform