# **FOREST FIRE DETECTION USING IOT**





INTRODUCTION :

When forest fires burn, they emit large volumes of carbon dioxide gas (CO2); you can use a network of IoT CO2 and temperature sensors for forest fire detection. IoT sensors can operate alongside satellite and optical detection systems or form a standalone network of sensors near key strategic Forest fires (wildfires) are common hazards in forests, particularly in remote or unmanaged areas. It is possible to detect forest fires, elevated CO2, and temperature levels using Internet of Things (IoT) sensors. You can deploy IoT, satellite and solar sensors in remote areas without the need for internet, cellular or mains power .



PURPOSE :

The main purpose of forest fire detection is to forecast the daily danger of fire and to obtain data for the burned areas compared with the traditional techniques of forest fire detection ,a wireless technique to detect the fire is far better.

LITERATURE SURVEY:

A.Existing problem :

Forest fires represent a real threat to human lives, ecological systems, and infrastructure. Many commercial fire detection sensor systems exist, but all of them are difficult to apply at large open spaces like forests because of their response delay, necessary maintenance needed, high cost, and other problems.

B.Proposed solution :

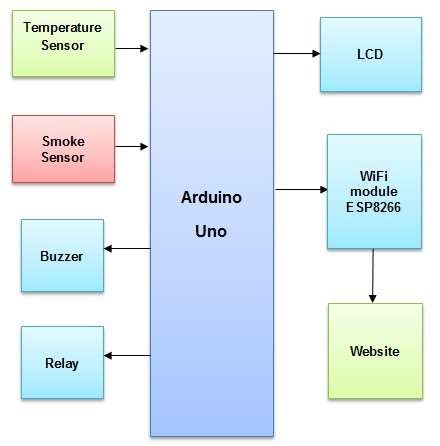
Now there are many methods used for providing fire detection to forest. They are remote monitoring, surveillance system and ZIGBEE technology etc. ... These systems are efficient for detecting fire in very short duration.

Satellite system. Earth-orbiting satellites and even air-floating devices have been employed for the observation and detection of forest fires.

Hence in this following ways forest fire can be detected and not only this there are many more modern technologies coming up in which we can detect the fire and save the forest.Hence these are a few proposed solutions with ,many more to come.

FLOW CHART:

Block diagram :



Hardware/software designing :

As we are designing using internet of things and then we use the coding technique of python and cloud services like IBM WATSON IOT platform and check the status of things such as

temperature,humidity,pressure by running the simulation .and then we use the node-red for both set the state in the homekit alarm system or then have another one set up to only fire after 30 seconds.And then we use web UI where we permit the nodeMcu to interface with the web and then the user can see it.

RESULT:

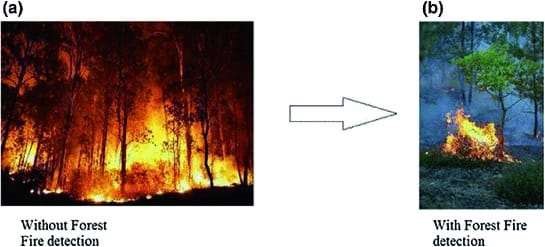
We successfully can detect forest fire using advanced technologies and internet of things and save a lot of forest and living beings inside of it.Hence our is that we can learn a lot about internet of things and apply it in many places to save our nature.

ADVANTAGES AND DISADVANTAGES:

The proposed system(IOT) detects the forest fire at a faster rate compared to existing system. It has enhanced data collection feature. The major aspect is that it reduces false alarm and also has accuracy due to various sensors present.

Monitoring of the potential risk areas and an early detection of fire can significantly shorten the reaction time and also reduce the potential damage as well as the cost of fire fighting.

It(existing system) can detect only when the fire is widely affected the forest. This cause many damages. But the cost of these devices are very high. These are not making use of the IOT that is main disadvantage of these system.



**APPLICATIONS:**

IoT Enabled Forest fire detection and online monitoring system . The objective of this project was to detect the forest fire as early as possible by measuring the level of temperature and CO2 level. They have used Temperature and smoke sensor to detect the ignition alarming temperature and the level of carbon dioxide gas (CO2).

Fire detection and management plays a very crucial part in terms of safety. Therefore this proposed system can be implemented in malls, offices, data centers etc..,.

With connections to low power wide area (LPWA) or even cellular networks, fire safety IoT is present for enhancing fire preventions, speeding up response times, and also keeping the first responders safe when they encounter fires.

**CONCLUSIONS:**

This type of system is the first of its kind to ensure no further damage is then to forests when there is fire breakout and immediately a message is sent to the user through the App. Immediate response or early warning to a fire breakout is mostly the only ways to avoid losses and environmental, cultural heritage damages to a great extent. Therefore the most important goals in fire surveillance are quick and reliable detection of fire. It is so much easier to suppress fire while it is in its early stages. Information about progress of fire is highly valuable for managing fire during all its stages. Based on this information the firefighting staff can be guided on target to block fire before it reaches cultural heritage sites and to suppress it quickly by utilizing required firefighting equipment and vehicles. With further research and innovation, this project can be implemented in various forest areas so that we can save our forests and maintain great environment.

**Future scope:**

The proposed system detects the forest fire at a faster rate compared to existing system. It has enhanced data collection feature. The major aspect is that it reduces false alarm and also has accuracy due to various sensors present. It minimizes the human effort as it works automatically. This is very affordable due to which can be easily accessed. The main objective of our project is to receive an alert message through an app to the respective user.

BIBLIOGRAPHY:

>>Install Python IDLE.

>>Create IBM account.

>>Create Node Red Apllication.

>>Create IBM Watson to IOT Platform.

>>Create the MIT App Inventor account, to build the mobile App.

**APPENDIX:**

A)SOURCE CODE:

import wiotp.sdk.device

import time

import random

myConfig = {

"identity": {

"orgId": "1pqadn",

"typeId": "ESP32",

"deviceId":"12345"

},

"auth": {

"token": "12345678"

}

}

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()

while True:

temp=random.randint(-20,125)

hum=random.randint(0,100)

smoke=random.randint(0,100)

fire=random.randint(0,100)

myData={'temperature':temp, 'humidity':hum,'smoke':smoke,'fire':fire}

client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)

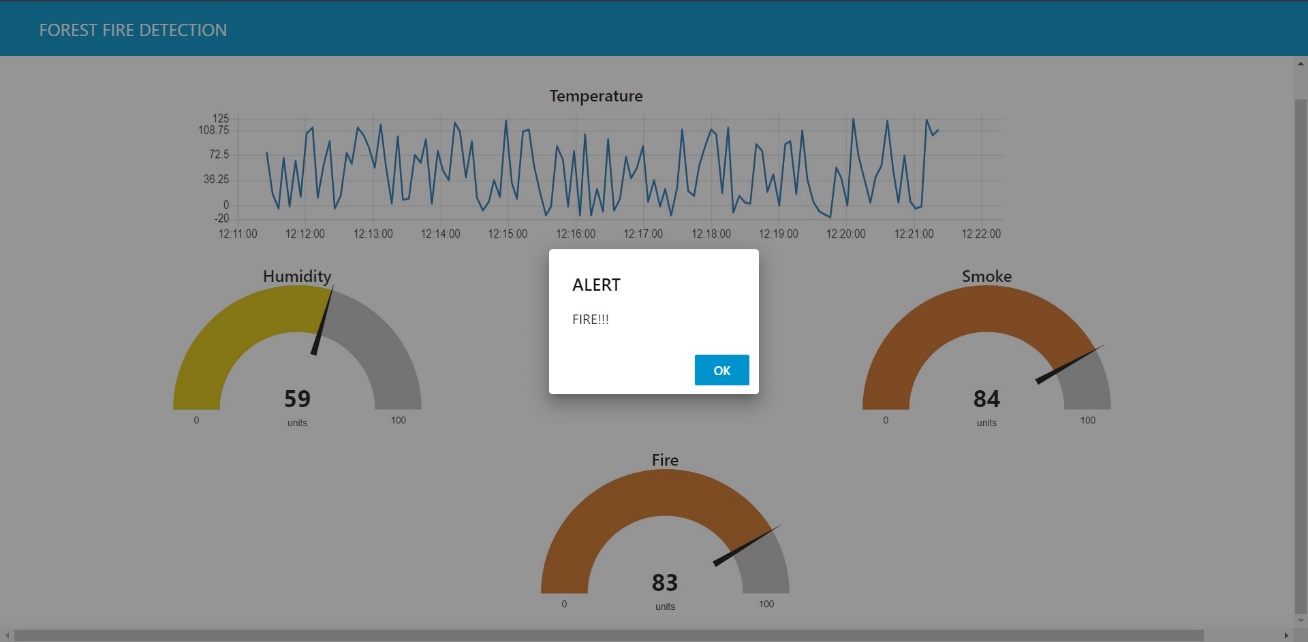
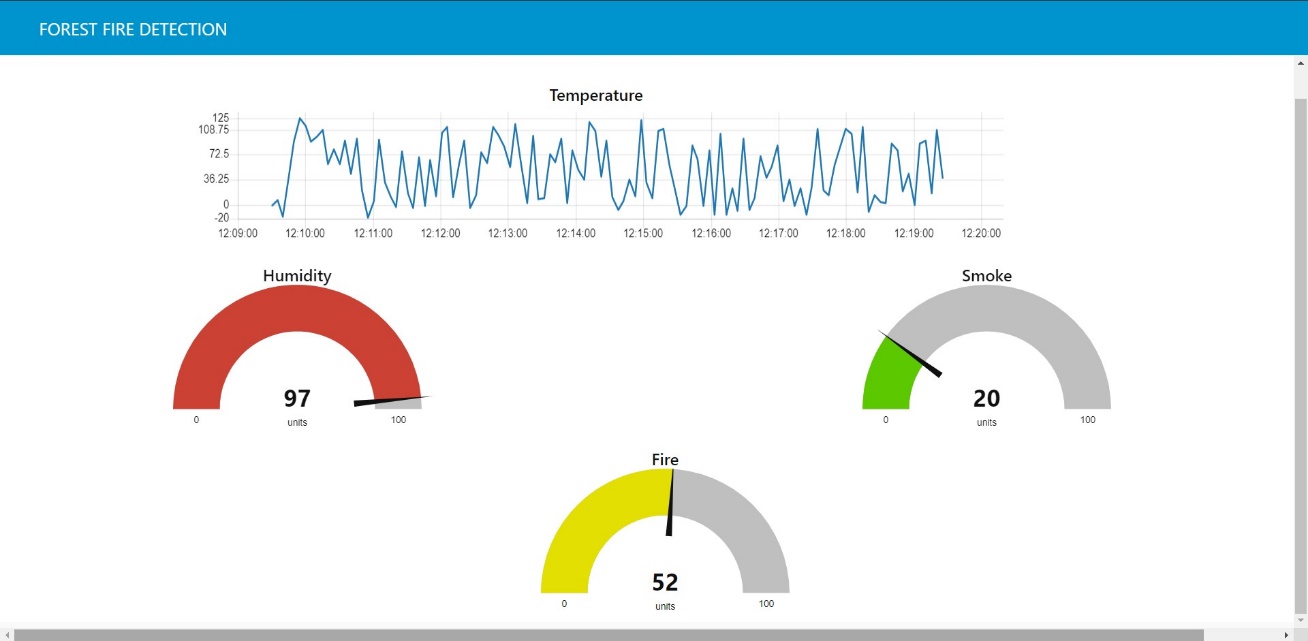
print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

time.sleep(5)

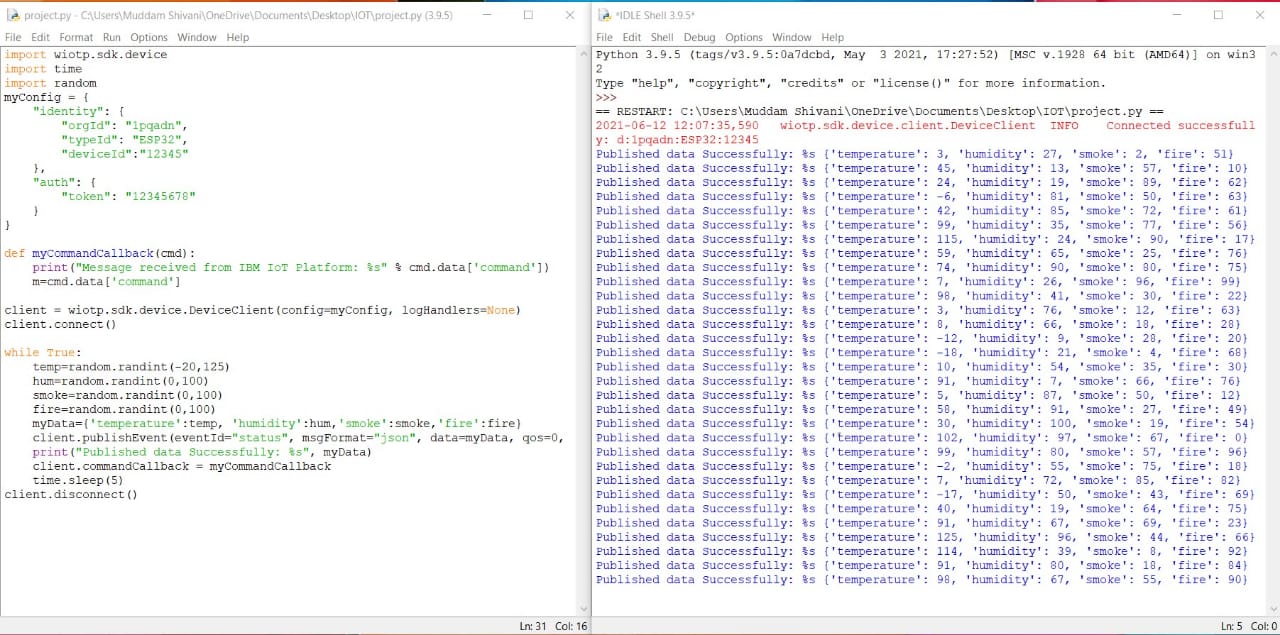
client.disconnect()

B) UI OUTPUT SCREENSHOT:

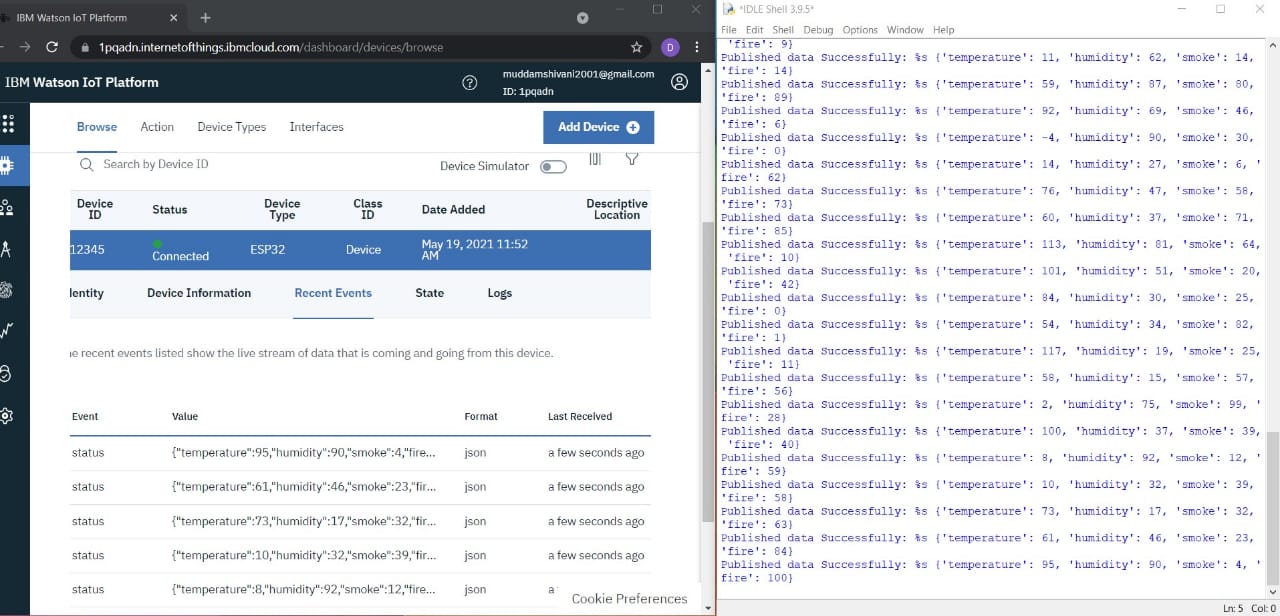


**SCREENSHOTS OF PROJECT**

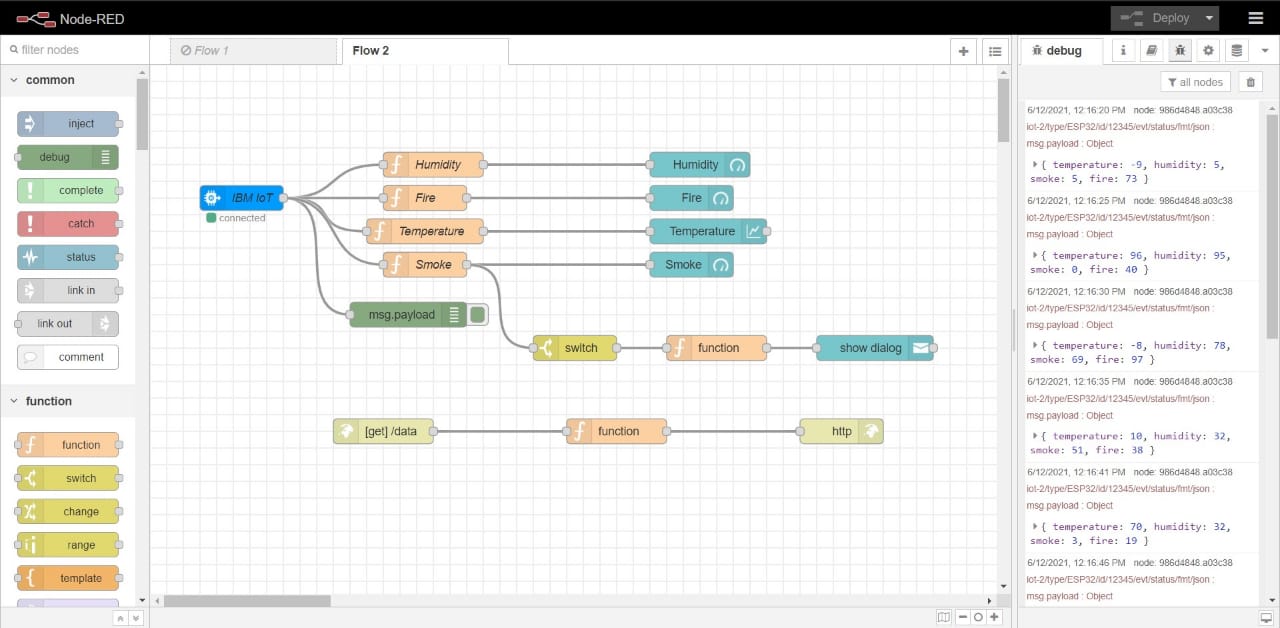
Python:



IBM Watson:

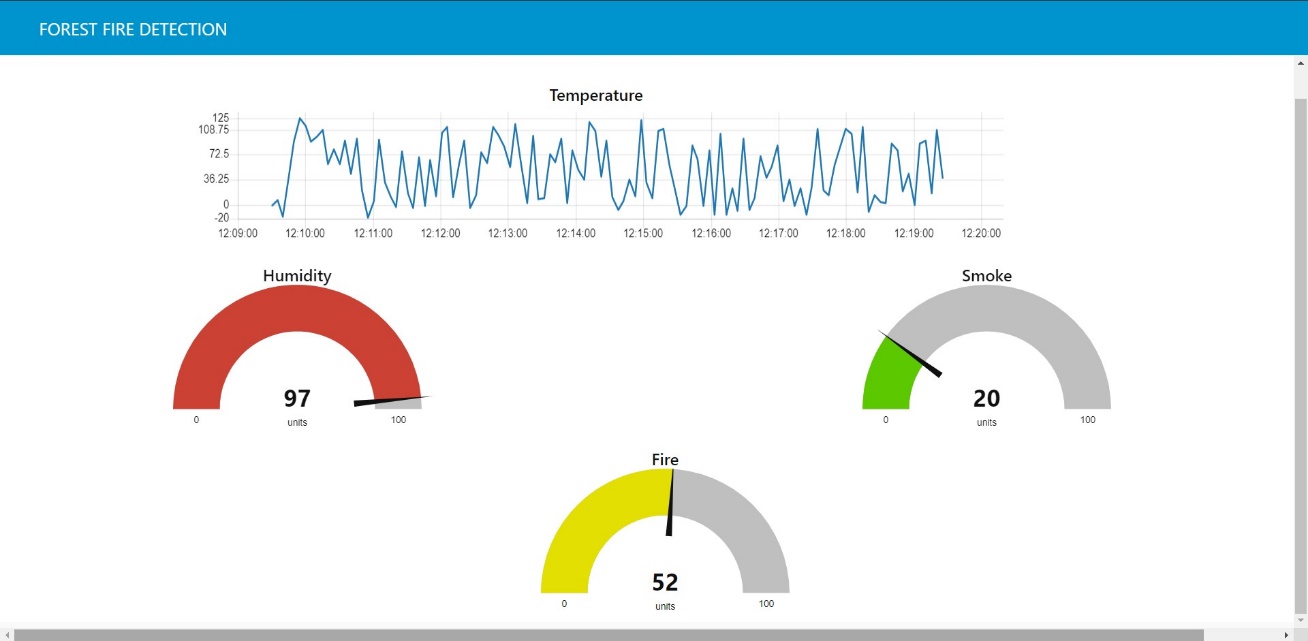


NodeRed:

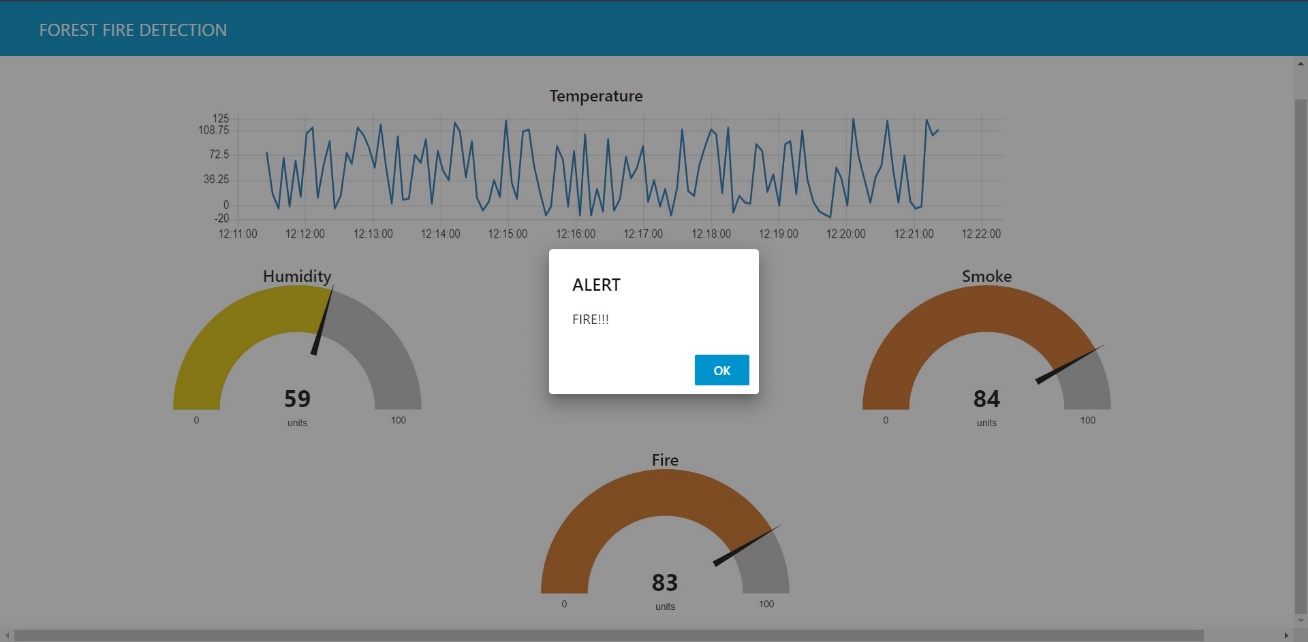


Web UI:

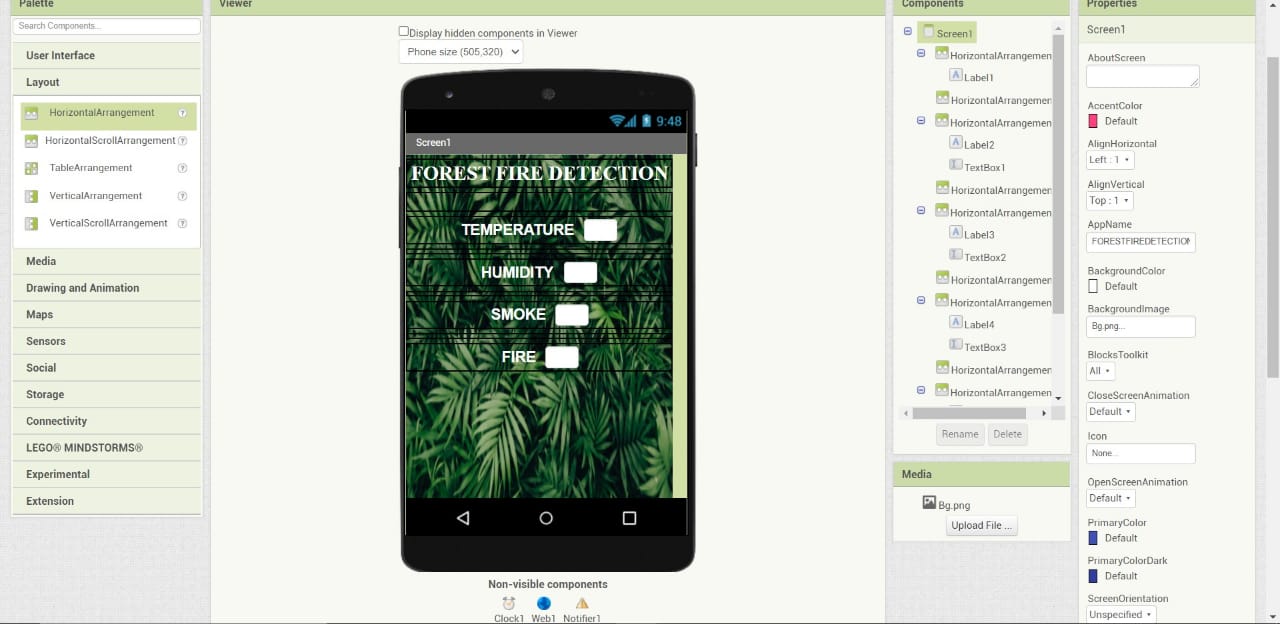
Without Alert:



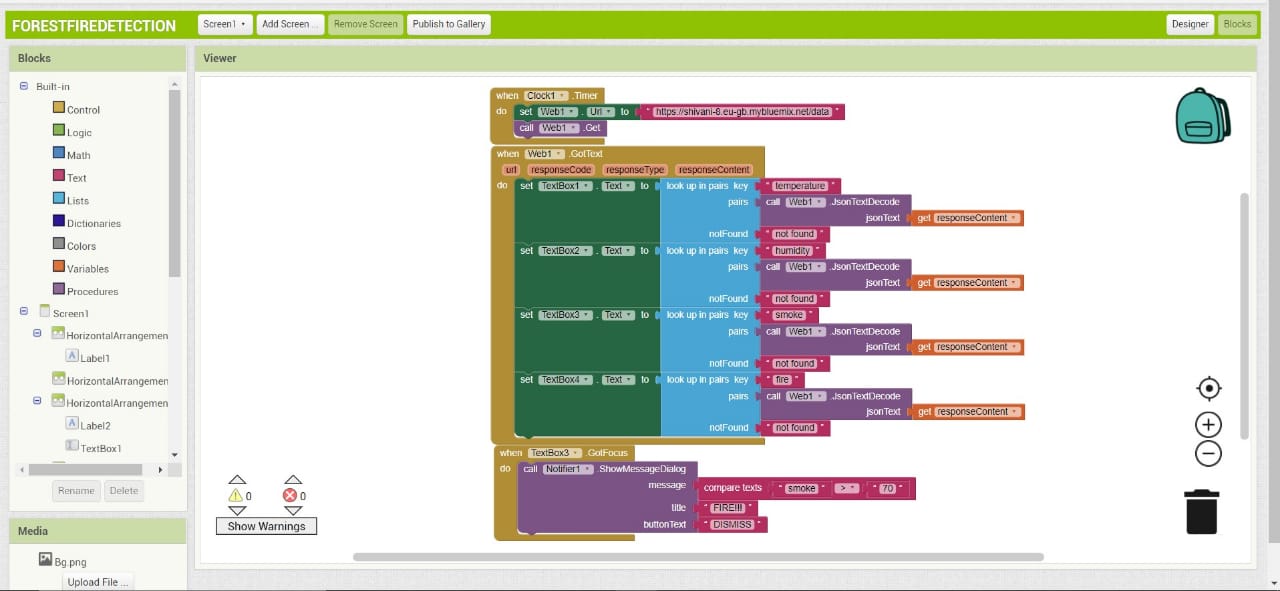
With Alert:



MIT App inventor:

Design:  


Blocks:



Mobile Application:

Without Alert: With Alert:

