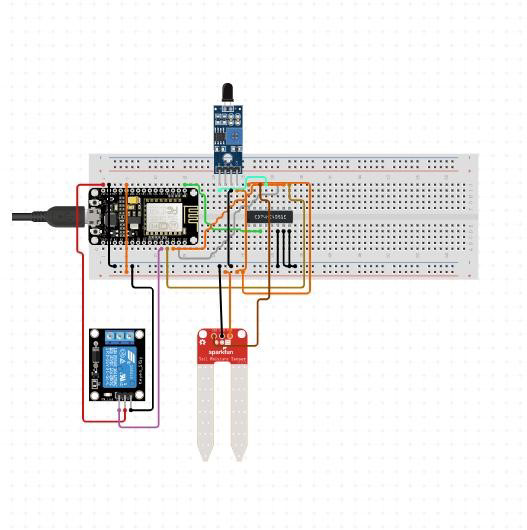
Smart Water Management

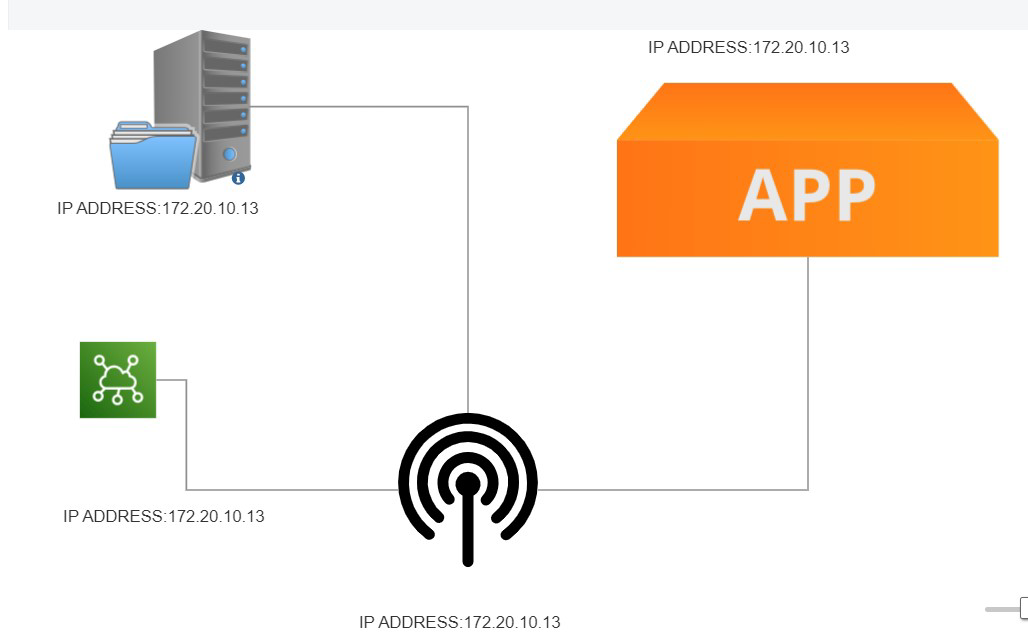
Phase 3 IoT Report

**1.Circuit Diagram**



# Fig: Circuit Diagram

**2. Network Structure Diagram**



# Fig: Network Diagram

**3.Hardware Components Selection**

The components required for this problem are listed.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Si No** | **Description** | **Type** |  | **Qty** |
| 1 | NodeMCU | V1.0E | 1 |  |
| 2 | Bread Board | Big Size | 1 |  |
| 3 | Moisture Sensor | Sensor | 1 |  |
| 4 | Relay module | connecting | 1 |  |
| 5 | Wires | - | 1 Set |  |
| 6 | Water pump | - | 1 |  |
| 7 | Laptop and internet | Intel(R) Core(TM) i3-10110U CPU @ 2.10GHz 2.59 GHz (8.00 GB RAM) and wifi or mobile data. | 1 |  |

**Table:** Manufactured with this hardware specification

**4.Use Case Diagram**

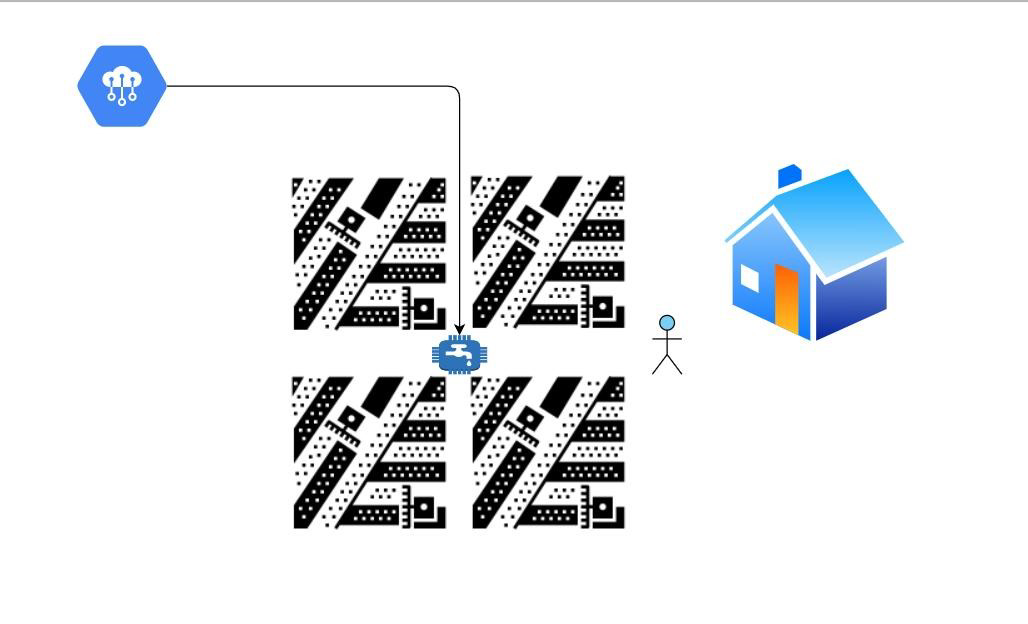
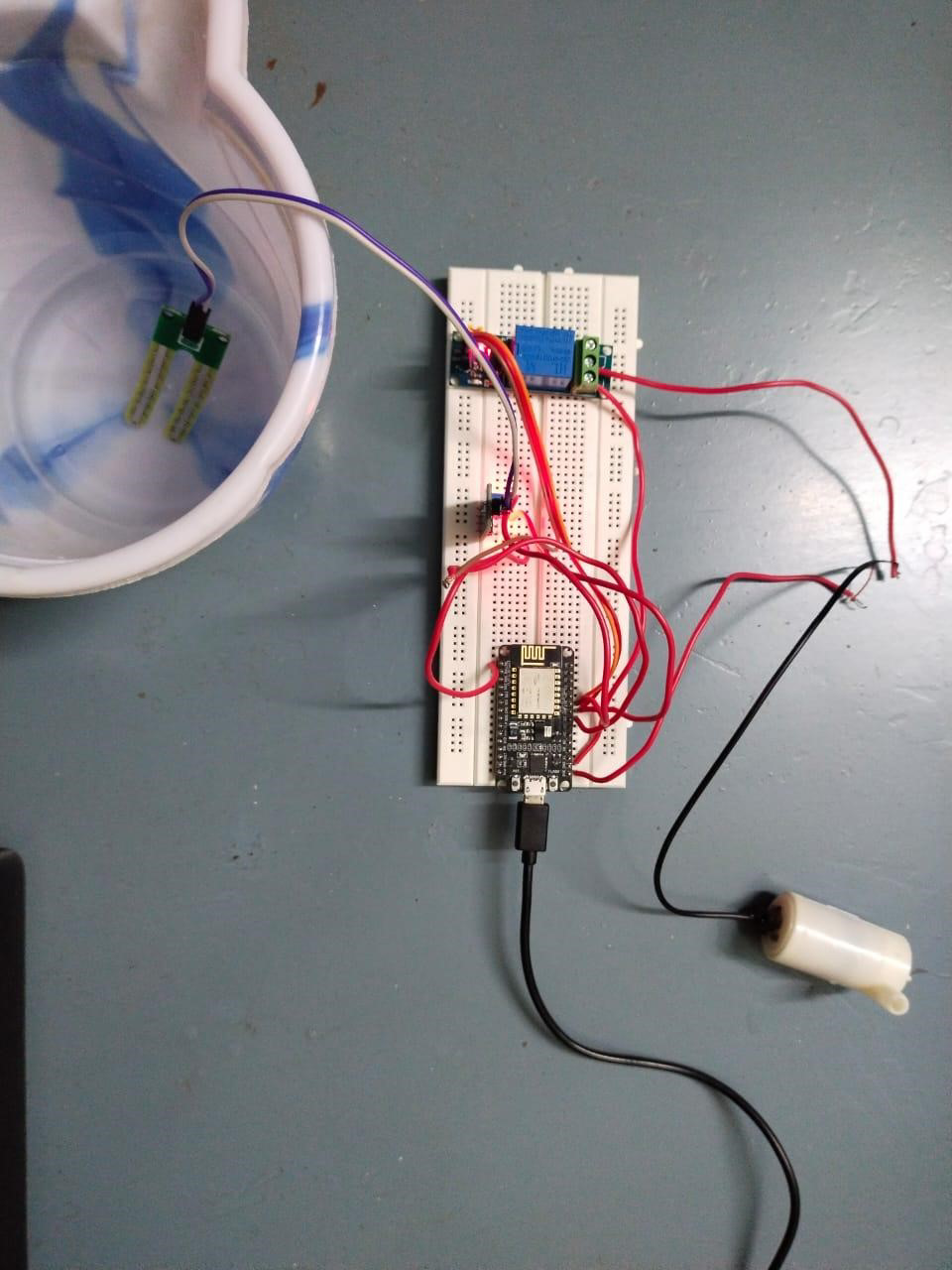


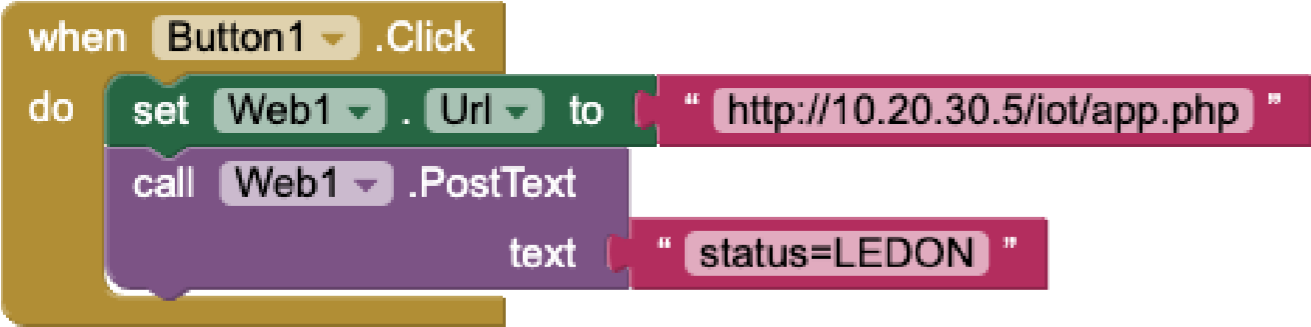
Fig: Use Case Diagram

**Model Working :**





**Sending request to the server**



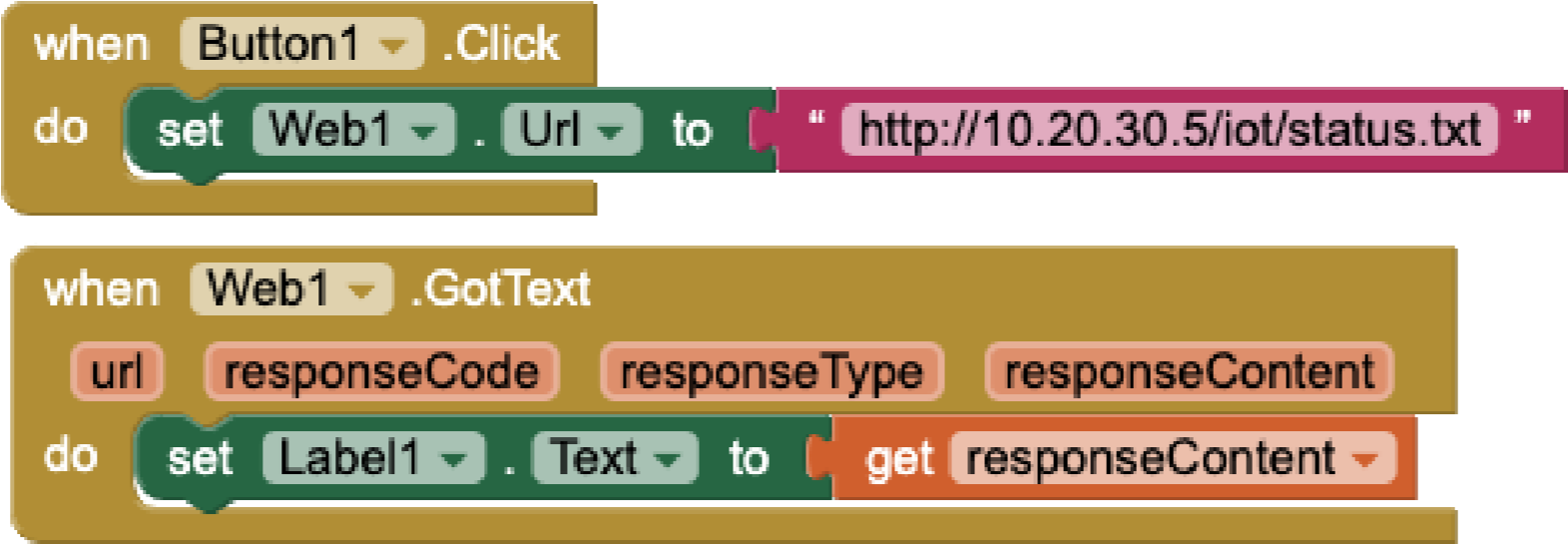
**Fig:** APP Data Send to Server (APP Blocks)

**PHP Code: (to send response)**

<?php

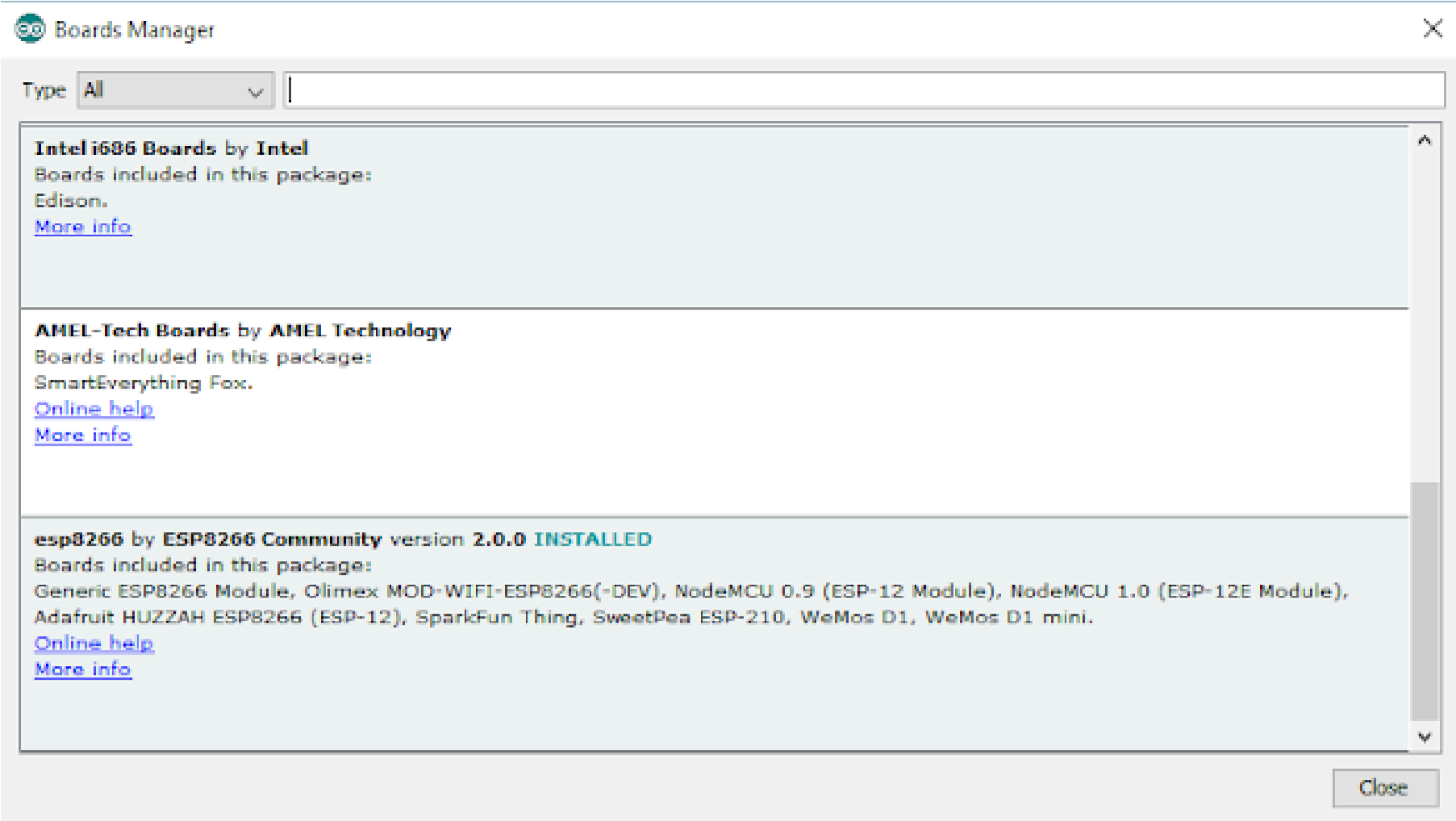
$status = $\_POST['status']; $name=fopen('status.txt','w'); fwrite($name, $status); fclose($name);

?>



**Fig:** APP Blocks(APP Data Fetch From Server)

**Working of arduino**



**Pin Connection**

|  |  |
| --- | --- |
| **NodeMCU PIN** | **LED** |
| D0 | Anode |
| GND | Cathode |

**Program for the project**

import time

import pyrebase

WIFI\_SSID = "your\_wifi\_ssid"

WIFI\_PASSWORD = "your\_wifi\_password"

FIREBASE\_HOST = "your\_firebase\_host"

FIREBASE\_AUTH = "your\_firebase\_auth\_token"

SOIL\_MOISTURE\_PIN = 0

WATER\_PUMP\_PIN = 5

soil\_moisture\_threshold = 500

config = {

"apiKey": FIREBASE\_AUTH,

"authDomain": FIREBASE\_HOST,

"databaseURL": FIREBASE\_HOST,

"storageBucket": FIREBASE\_HOST

}

firebase = pyrebase.initialize\_app(config)

db = firebase.database()

def setup():

print("Connecting to WiFi...")

while True:

try:

wifi = pywifi.PyWiFi()

iface = wifi.interfaces()[0]

iface.disconnect()

time.sleep(1)

profile = pywifi.Profile()

profile.ssid = WIFI\_SSID

profile.auth = const.AUTH\_ALG\_OPEN

profile.akm.append(const.AKM\_TYPE\_WPA2PSK)

profile.cipher = const.CIPHER\_TYPE\_CCMP

profile.key = WIFI\_PASSWORD

iface.remove\_all\_network\_profiles()

tmp\_profile = iface.add\_network\_profile(profile)

iface.connect(tmp\_profile)

time.sleep(5)

if iface.status() == const.IFACE\_CONNECTED:

print("Connected to WiFi!")

break

except Exception as e:

print(e)

continue

def loop():

soil\_moisture\_level = analogRead(SOIL\_MOISTURE\_PIN)

if soil\_moisture\_level < soil\_moisture\_threshold:

digitalWrite(WATER\_PUMP\_PIN, HIGH) # Turn on the water pump

time.sleep(5) # Water for 5 seconds

digitalWrite(WATER\_PUMP\_PIN, LOW) # Turn off the water pump

db.child("soilMoistureLevel").set(soil\_moisture\_level)

db.child("waterPumpStatus").set(1)

else:

digitalWrite(WATER\_PUMP\_PIN, LOW) # Turn off the water pump

db.child("soilMoistureLevel").set(soil\_moisture\_level)

db.child("waterPumpStatus").set(0)

time.sleep(60) # Wait for a minute before checking again

setup()

while True:

loop(