

1. INTRODUCTION

The main reason for choosing this project is to make life easier of human beings. As we all live a busy lifestyle some may feel turning off the lights, fan, etc., before going out is a time-consuming process. But no so long as this technology helps us in monitoring everything single handily by monitoring it with the help of our phone(s). As you become aged, simple tasks can become difficult. With features like voice command, home automation gives freedom and independence to the elderly. Additionally, the idea of home automation system will improve the normal living status at houses.

- **Advantages:**

1. **Safety:** User will be notified when any fire or gas leakage is there.
2. **Real time monitoring:** User will get to know electricity usage statistics, water tank level, plant moisture level.
3. **Centralized system:** User will get all features of home automation at one platform.
4. **Saves Time:** Everyone is living in busy environment, user don't even have to worry about the home.
5. **Ease for Disabled & Senior Citizens:** For Elderly & disabled people, simple tasks can become difficult. With features like voice command, home automation gives freedom and independence to the elderly.
6. **Monitor from anywhere:** User can monitor their home from anywhere they just have to login.

- **Disadvantages:**

1. **Dependency on Internet:** The system requires active strong internet connection (24×7)
2. **Dependency on power supply:** The system requires power supply (24×7)
3. **Electronic Circuit Failure:** Electronic Kit and Circuit may fail in future can cause malfunctioning to the system, short-circuit to the system.

2. PROPOSED SYSTEM

1. **Electrical port switching (on/off):** User can switch ON the electrical ports and switch OFF the electrical ports. There will be total 4 ports.
2. **Automatic on/off the electrical ports:** User will select Days, Start Time, Stop Time for the particular electrical port. At selected days & time the electrical port will automatically turned ON & turned OFF.
3. **Electricity consumption monitoring:** User will get the real time consumption information in the mobile. User will get the information such as energy consumption, power consumption, current consumption, bill (in rs). User can see the graph of the consumption.
4. **Water tank measuring:** User can see the current water level of the tank on mobile. User will be notified when the water tank will be full.
5. **Air quality monitoring:** User can see the current indoor Temperature, Humidity and the AQI (Air Quality Index) on mobile.
6. **Plant monitoring:** User can see the current moisture level of the plant on mobile. User will be notified when the plant needs water.
7. **Gas detection:** User will be notified when any gas leakage is detected.
8. **Fire detection:** User will be notified when any fire is detected.
9. **Google assistant:** User will voice command to the Google Assistant and particular action will be performed.
10. **Technical assistant bot:** User will ask command questions to the bot and bot will process it.
11. **Telegram Bot:** : User will ask command questions to the telegram bot and telegram bot will process it.

3. HARDWARE AND SOFTWARE REQUIREMENTS

- **Hardware Requirement:** -

- **Laptop or PC:**

1. i5 Processor Based Computer or higher
2. 8GB RAM
3. 25 GB Hard Disk

- **Electrical Components:**

1. 220-240v AC supply
2. Screws & nuts (20-25)
3. Electrical Wires (5-6 meters)
4. 4 buttons of 2 way switch
5. 4 switches
6. 1 electrical board of 4×4 size
7. 1 electrical board of 3×3 size
8. 1 electrical board of 1×4 size
9. Electrical tape

- **Electronic Components:**

1. Esp32 board
2. 4 set of Relays
3. Jumper Wires
4. DHT11 sensor
5. Flame sensor
6. Ultrasonic Sensor HC-SR04 sensor
7. MQ135 Air quality Sensor
8. 1 full size breadboard
9. 1 half size breadboard

10. Soil moisture sensor with probes
11. PZEM – 004T 80-260V 100A module
12. 12v DC supply
13. 5v DC supply

- **Android:**

1. 1.2 Quad core Processor or higher
2. 1GB RAM

o **Software Requirement:** -

- **Laptop or PC:**

1. Windows 7 or higher
2. Java
3. Android Studio
4. Node JS

- **Android:**

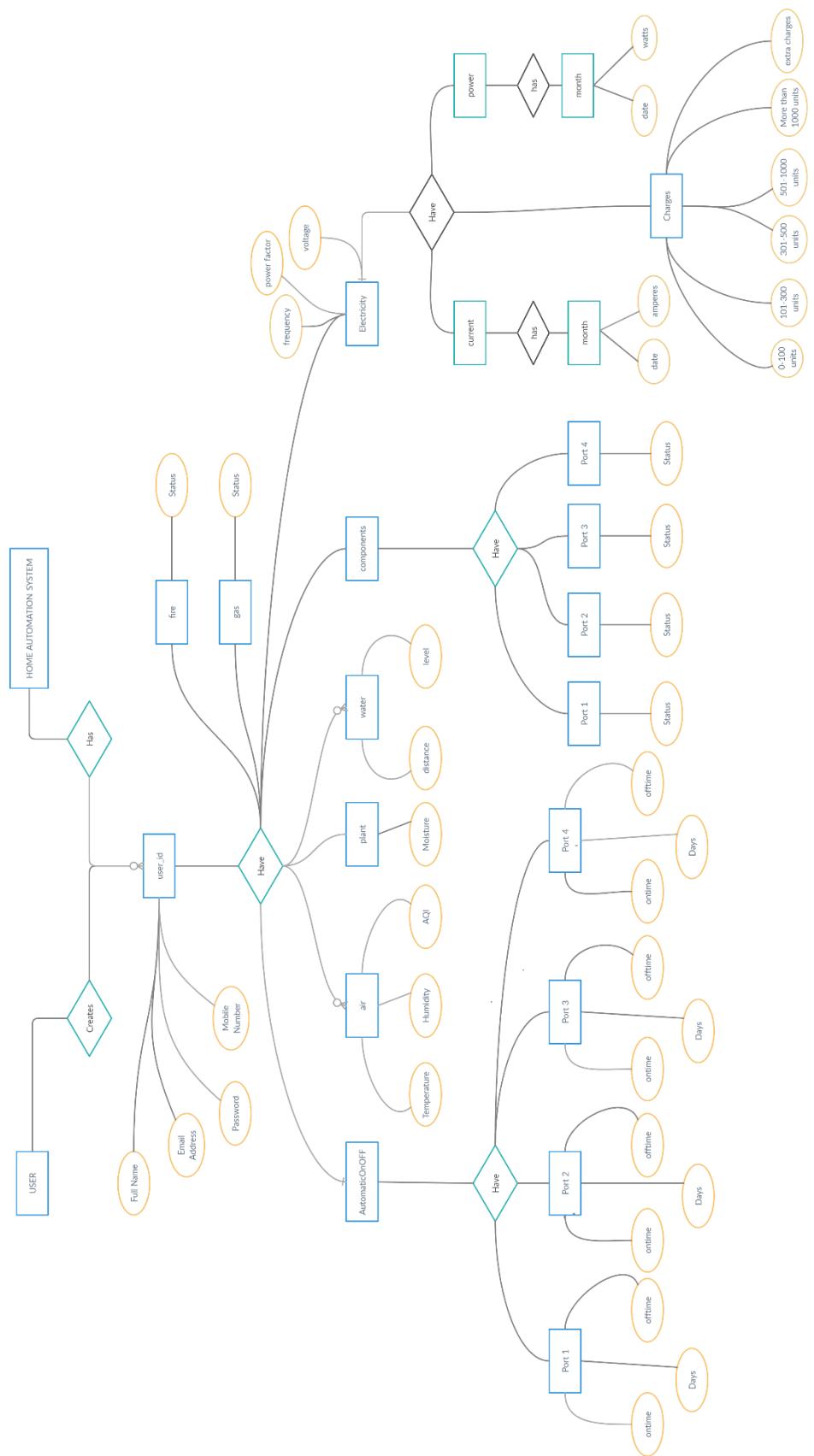
1. Android v5.0 or higher

4.EVENT TABLE

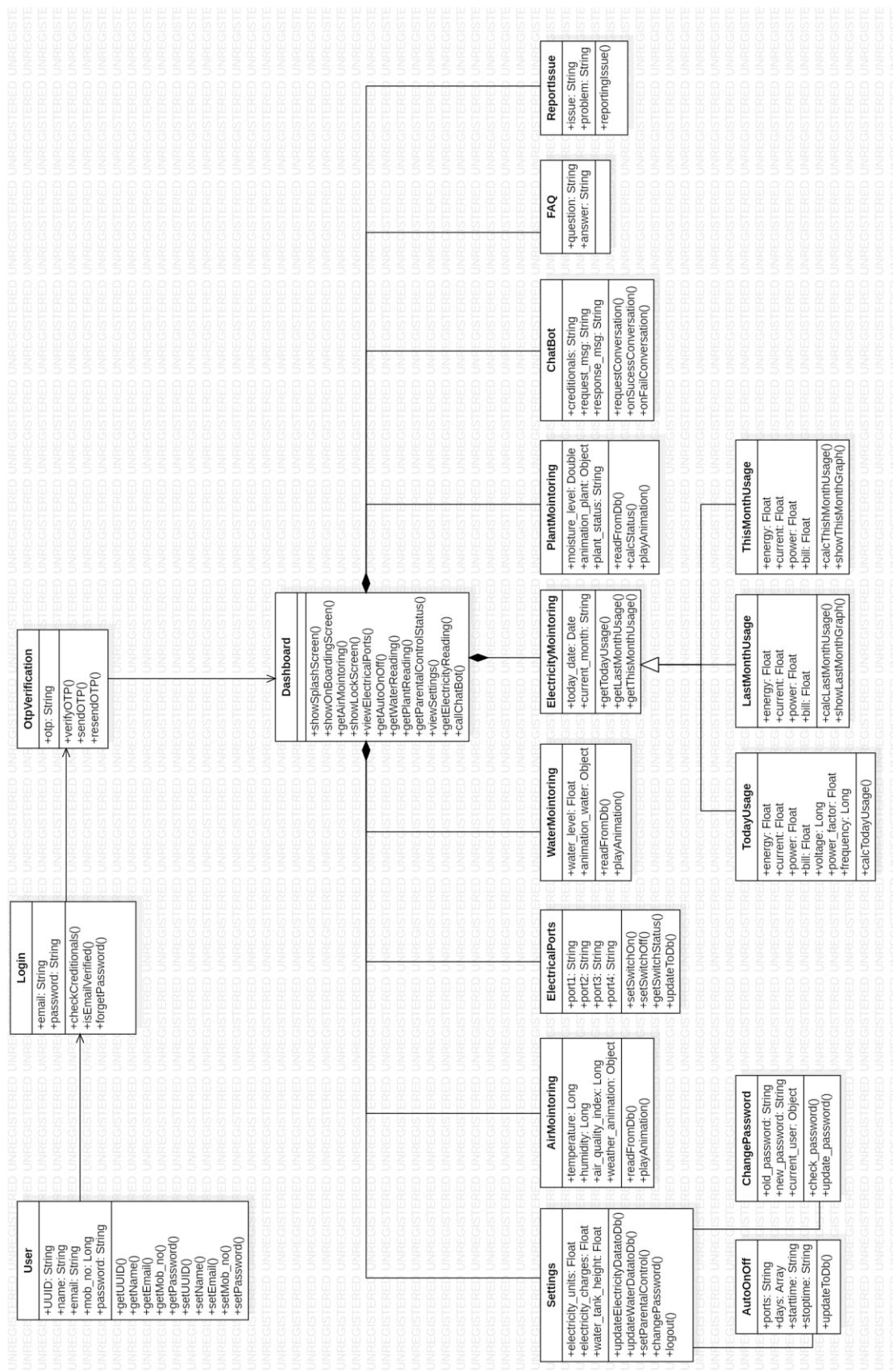
SR.NO.	EVENT	TRIGGER	SOURCE	ACTIVITY	RESPONSE	DESTINATION
1.	User Login	Login	User	Login	OTP will be send to email & Redirected to OTP Screen.	User
2.	2 factor verification	OTP	User	User will enter OTP	OTP will be verified & redirected to the dashboard.	User
3.	User Forget Password	Forget Password	User	Reset link will be sent to user	New Password will be updated.	User
4.	User Edit Profile	Edit Profile	User	Editing the Profile	Profile will be updated.	User
5.	Electrical Port Switching ON	Electrical Port ON	User	Setting port “ON” to the system	Port will be switched ON.	User
6.	Electrical Port Switching OFF	Electrical Port OFF	User	Setting port “OFF” to the system	Port will be switched OFF.	User
7.	Auto Port On/Off	Port On/Off	User	Setting days, start time & stop time.	Port will be auto switched ON/OFF.	User
8.	Water Tank	Water Level	System	Getting current water level.	User will be notified when tank overflow	User
9.	Indoor Air Monitoring	Detect the indoor temperature	System	Getting air temperature details.	Temperature, Humidity & AQI	User
10.	Electricity Usage	Check Electricity Usage	User	Getting Electricity usage from the system.	Electricity Usage & its statistics	User

11.	Fire Detection	Detect the fire	System	Detecting the fire.	User will be notified when fire is detected.	User
12.	Gas Leakage Detection	Detect the Gas Leakage	System	Detecting the gas.	User will be notified when gas is detected.	User
13.	Plant Status	Checking moisture level	User	Getting moisture level from the system.	User will be notified when moisture will be low	User
14.	Settings Tab	User updates Settings	User	Settings	Settings will be updated	User
15.	Google Assistant	User will Voice command.	User	Command will be processed.	Action of Command will be performed.	User
16.	Telegram	User will command.	User	Command will be processed.	Action of Command will be performed.	User
17.	Chatbot	Chatbot	User	Command will be processed.	Action of Command will be performed.	User
18.	FAQ	FAQ	User	User will open FAQ Screen.	User can view all FAQ's.	User
19.	Report Issue	Report Issue	User	User will Report Issue.	Issue will be send to Customer Care.	System
20.	User Change Password	Change Password	User	Change Password	New Password will be updated.	User
21.	User Logout	Logout	User	Logout	Logs out the user.	User

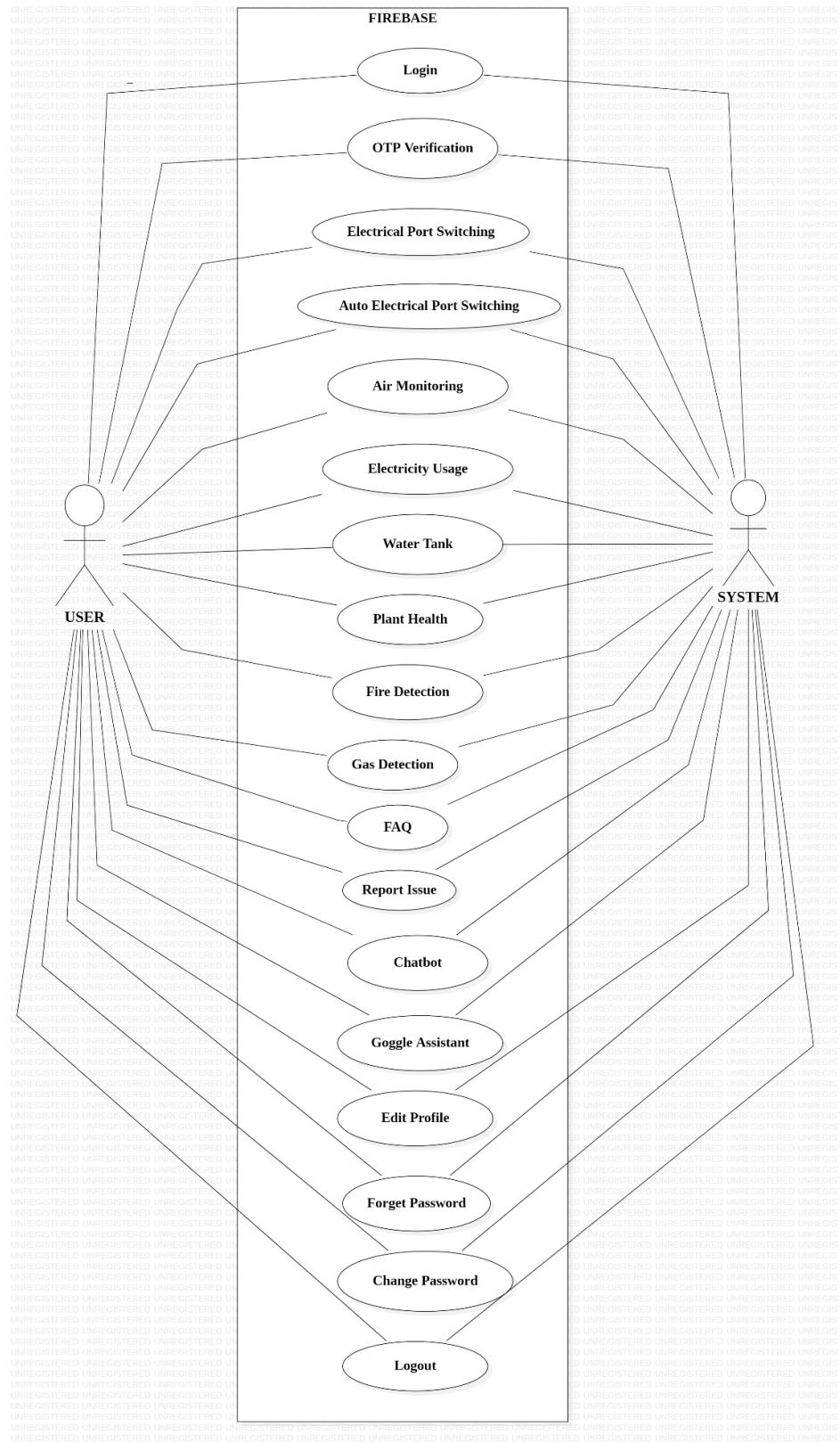
5. ENTITY RELATION DIAGRAM



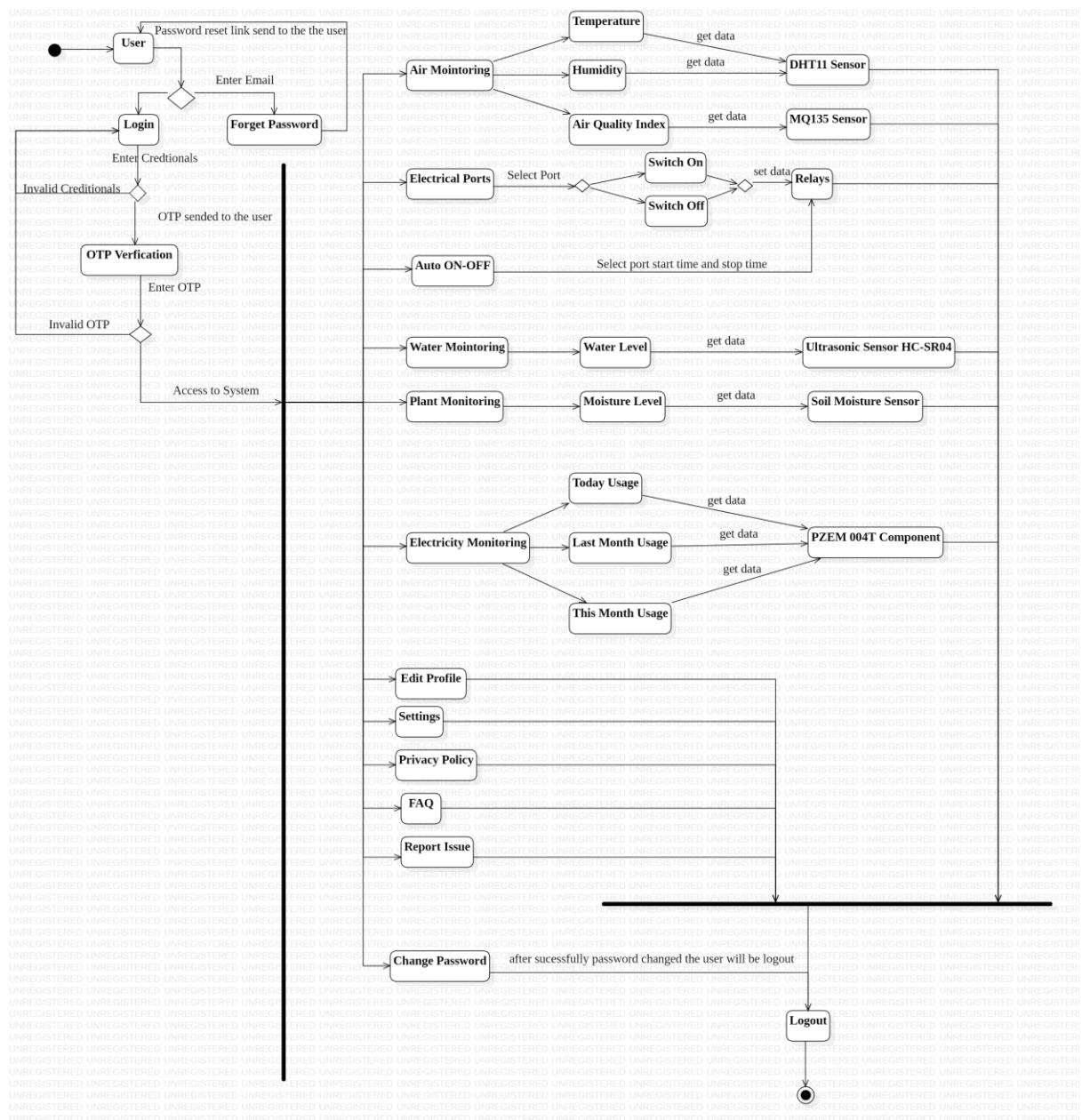
6. CLASS DIAGRAM



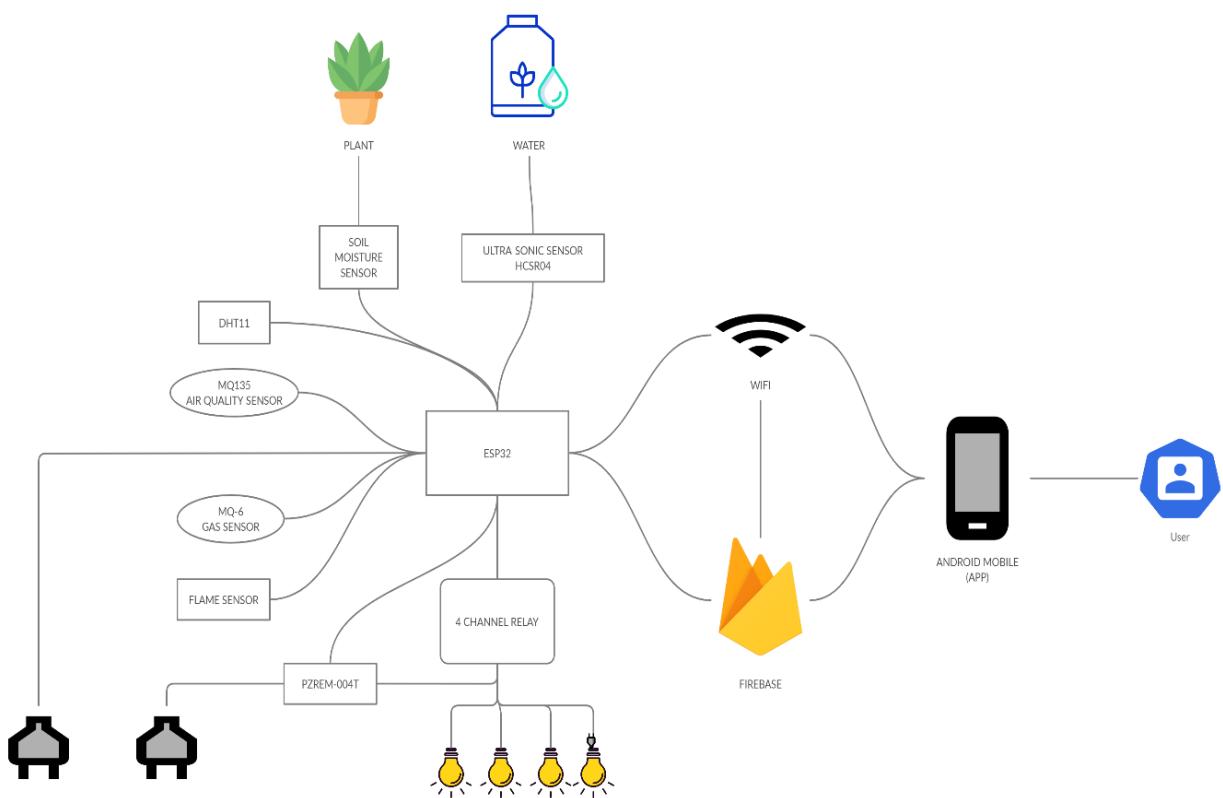
7. USE CASE DIAGRAM



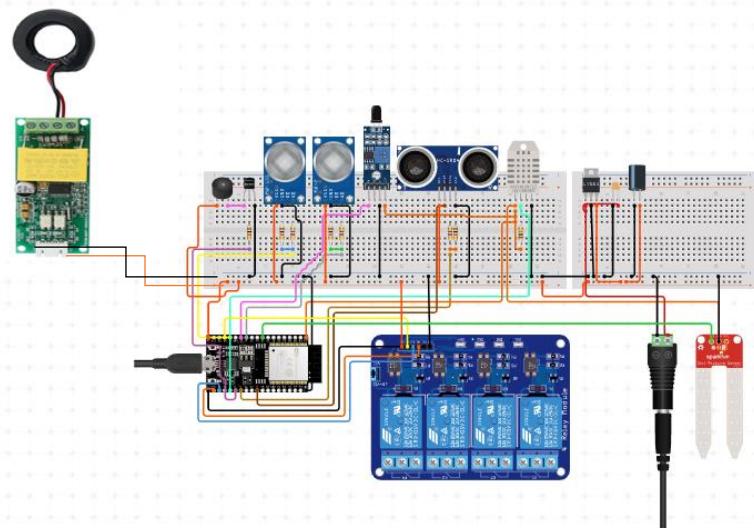
8. ACTIVITY DIAGRAM



9. BLOCK DIAGRAM



10. CIRCUIT DIAGRAM



11. CODE

IOT:

Myhomeiot.ino:

```
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <PZEM004Tv30.h>
#include <analogWrite.h>
#include "DHT.h"
#include <NTPClient.h>
#include <TimeLib.h>
#include <SPI.h>

#define FIREBASE_HOST "-----"
#define FIREBASE_AUTH "-----"

#define WIFI_SSID "-----"
#define WIFI_PASSWORD "-----"

#define DHTTYPE DHT11

char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday",
"Saturday"};
char monthsOfYear[12][12] = {"January", "February", "March", "April", "May", "June", "July",
"August", "September", "October", "November", "December"};

WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP);

String formattedDate, formattedTime, dayStamp, timeStamp, autoday, withoutsec, months,
finalMonth;
int monthint, monthdiff;
int hi, ti, flame_detected, mq6 ,mq135 ,soil, dbdistance,level;
float h ,t;
uint8_t DHTPin = 27;

DHT dht(DHTPin, DHTTYPE);
PZEM004Tv30 pzem(&Serial2);

int in1 = 4;
int in2 = 0;
int in3 = 2;
int in4 = 15;
int fire = 5;
int buzzer = 26;
int trigPin = 18;
```

```
int echoPin = 19;
long duration;
int distance;
int voltagei ,frequencyi ;
FirebaseData firebaseData;
WiFiClient client;

void setup()
{
    Serial.begin(115200);

    pinMode(in1, OUTPUT);
    pinMode(in2, OUTPUT);
    pinMode(in3, OUTPUT);
    pinMode(in4, OUTPUT);
    pinMode(fire, INPUT);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(buzzer, OUTPUT);

    pzem.resetEnergy();
    pzem.setAddress(0x42);

    dht.begin();

    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    Serial.print("Connecting to Wi-Fi");
    while (WiFi.status() != WL_CONNECTED)
    {
        Serial.print(".");
        delay(300);
    }
    Serial.print("Connected with IP: ");
    Serial.println(WiFi.localIP());
    Serial.println();

    timeClient.begin();
    timeClient.setTimeOffset(19800);

    while(!timeClient.update()) {
        timeClient.forceUpdate();
    }

    Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
    Firebase.reconnectWiFi(true);
}

void loop() {
    float current = pzem.current();
    float power = pzem.power();
    float energy = pzem.energy();
```

```

float voltage = pzem.voltage();
float frequency = pzem.frequency();
float pf = pzem(pf());
voltagei = (int) voltage;
frequencyi = (int) frequency;
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = duration*0.034/2;
flame_detected = digitalRead(fire);
mq6 = digitalRead(12);
mq135 = analogRead(35);
soil = analogRead(36);
h = dht.readHumidity();
t = dht.readTemperature();
hi = (int) h;
ti = (int) t;

while(!timeClient.update()) {
    timeClient.forceUpdate();
}

formattedDate = timeClient.getFormattedDate();
int splitT = formattedDate.indexOf("T");
dayStamp = formattedDate.substring(0, splitT);
months=dayStamp.substring(5,7);
monthint=months.toInt();
monthdiff=monthint-1;
finalMonth = String(monthsOfYear[monthdiff]);
timeStamp = formattedDate.substring(splitT+1, formattedDate.length()-1);
withoutsec=timeStamp.substring(0,5);
autoday=String(daysOfTheWeek[timeClient.getDay()]);
if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port1/days"))
{
String p1days=firebaseData.stringValue();
p1days.replace('[', ' ');
p1days.replace(']', ' ');
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port1/ontime");
String p1ontime=firebaseData.stringValue();
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port1/offtime");
String p1offtime=firebaseData.stringValue();
String myarrayp1[] = {p1days} ;
for ( int i = 0; i < sizeof(myarrayp1); i++ )
{
if(myarrayp1[i]==autoday)
}
}

```

```

{
  if(p1otime.equals(withoutsec))
  {
    Firebase.setString(firebaseData,
    "users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port1","ON");
  }

  if(p1oftime.equals(withoutsec))
  {
    Firebase.setString(firebaseData,
    "users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port1","OFF");
  }
}
}

if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port2/days"))
{
String p2days=firebaseData.stringData();
p2days.replace('[', ' ');
p2days.replace(']', ' ');
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port2/ontime");
String p2otime=firebaseData.stringData();
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port2/offtime");
String p2oftime=firebaseData.stringData();
String myarrayp2[] = {p2days} ;
for ( int i = 0; i < sizeof(myarrayp2); i++)
{
if(myarrayp2[i]==autoday)
{
  if(p2otime.equals(withoutsec))
  {
    Firebase.setString(firebaseData,
    "users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port2","ON");
  }

  if(p2oftime.equals(withoutsec))
  {
    Firebase.setString(firebaseData,
    "users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port2","OFF");
  }
}
}

if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port3/days"))
{

```

```
String p3days=firebaseData.stringValue();
p3days.replace('[', ' ');
p3days.replace(']', ' ');
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port3/ontime");
String p3otime=firebaseData.stringValue();
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port3/offtime");
String p3oftime=firebaseData.stringValue();
String myarrayp3[] = {p3days} ;
if(p3days.indexOf(autoday) > 0)
{
    if(p3otime.equals(withoutsec))
    {
        Firebase.setString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port3","ON");
    }

    if(p3oftime.equals(withoutsec))
    {
        Firebase.setString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port3","OFF");
    }
}

if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port4/days"))
{
String p4days=firebaseData.stringValue();
p4days.replace('[', ' ');
p4days.replace(']', ' ');
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port4/ontime");
String p4otime=firebaseData.stringValue();
Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/AutomaticOnOff/port4/offtime");
String p4oftime=firebaseData.stringValue();
String myarrayp4[] = {p4days} ;
for ( int i = 0; i < sizeof(myarrayp4); i++ )
{
    if(myarrayp4[i]==autoday)
    {
        if(p4otime.equals(withoutsec))
        {
            Firebase.setString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port4","ON");
        }

        if(p4oftime.equals(withoutsec))
        {
```

```

Firebase.setString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port4","OFF");
}
}
}
}
if( !isnan(current )){
    if(Firebase.getFloat(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/current/"+finalMonth+"/"+dayStamp))
    {
        float oldVal=firebaseData.floatData();
        float newVal=oldVal+(current/1000.000f);
        Firebase.setFloat(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/current/"+finalMonth+"/"+dayStamp,newVal
);
    }
    else
    {
        float newVal=current/1000.000f;
        Firebase.setFloat(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/current/"+finalMonth+"/"+dayStamp,newVal
);
    }
}

if( !isnan(power )){
    if(Firebase.getFloat(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/power/"+finalMonth+"/"+dayStamp))
    {
        float oldVal=firebaseData.floatData();
        float newVal=oldVal+(power/10.0f);
        Firebase.setFloat(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/power/"+finalMonth+"/"+dayStamp,newVal
);
    }
    else
    {
        float newVal=power/10.0f;
        Firebase.setFloat(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/power/"+finalMonth+"/"+dayStamp,newVal
);
    }
}

if( !isnan(voltage )){
    Firebase.setInt(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/voltage",voltagei);
}

if( !isnan(frequency )){

```

```
    Firebase.setInt(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/frequency",frequency);
}

if( !isnan(pf)){
    Firebase.setFloat(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/electricity/pf",pf);
}

if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port1"))
{
String val=firebaseData.stringData();
if(val=="ON")
{
    digitalWrite(in1,LOW);
}
else
{
    digitalWrite(in1,HIGH);
}
}

if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port2"))
{
String val=firebaseData.stringData();
if(val=="ON")
{
    digitalWrite(in2,LOW);
}
else
{
    digitalWrite(in2,HIGH);
}
}

if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port3"))
{
String val=firebaseData.stringData();
if(val=="ON")
{
    digitalWrite(in3,LOW);
}
else
{
    digitalWrite(in3,HIGH);
}
}
```

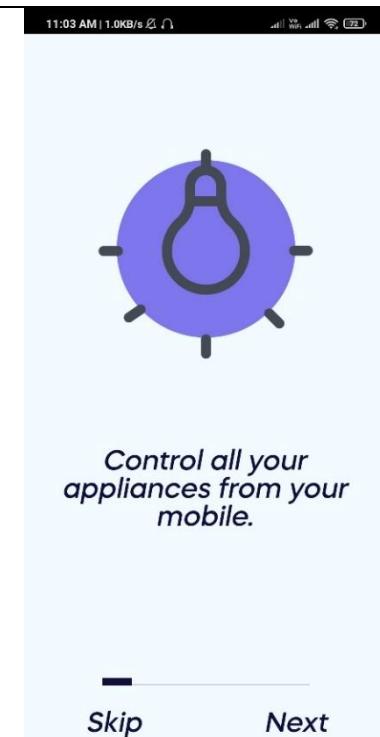
```
if(Firebase.getString(firebaseData,
"users/3Oou95cKpPX6OPBduFNTOHx6sO32/components/port4"))
{
String val=firebaseData.stringValue();
if(val=="ON")
{
  digitalWrite(in4,LOW);
}
else
{
  digitalWrite(in4,HIGH);
}
}

Firebase.getInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/water/distance");
dbdistance = firebaseData.intValue();
level = dbdistance - distance;
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/water/level",level);
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/air/temp", ti);
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/air/humidity", hi);
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/gas/gasleak", mq6);
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/air/mq135q", mq135);
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/plant/moisture", soil);
Firebase.setInt(firebaseData, "users/3Oou95cKpPX6OPBduFNTOHx6sO32/fire/fireleak",
flame_detected);
if(flame_detected==0)
{
  digitalWrite(buzzer, HIGH);
}
else
{
  digitalWrite(buzzer, LOW);
}
delay(1000);
```

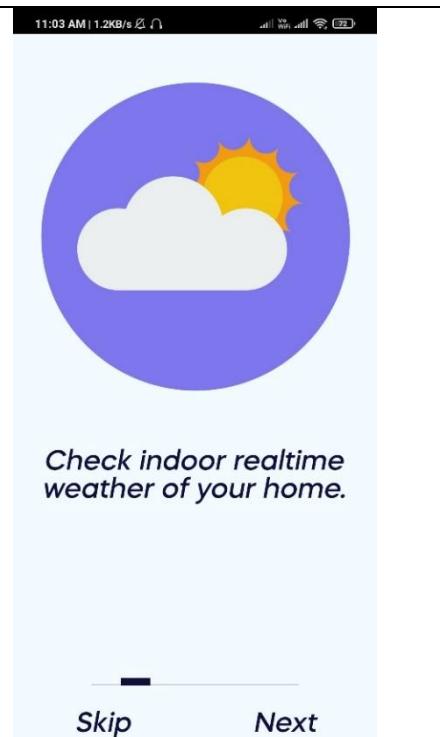
12. USER MANUAL WITH SCREEN SHOTS



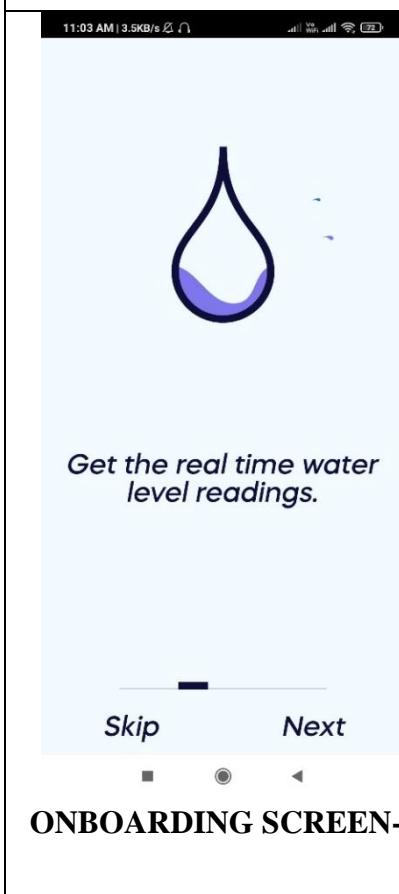
SPLASH SCREEN



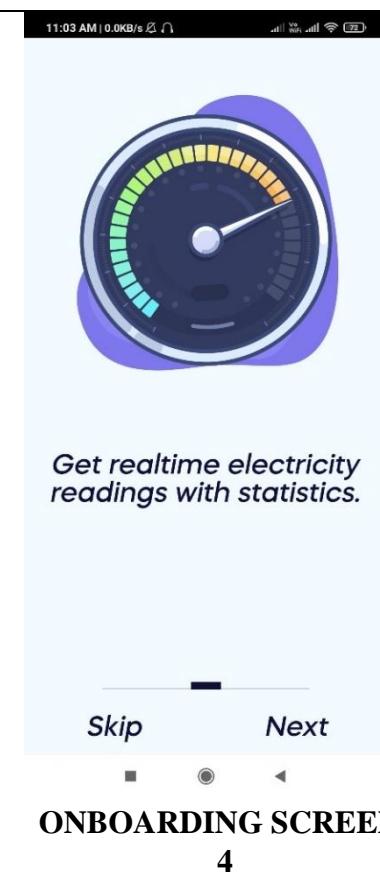
ONBOARDING SCREEN-1



ONBOARDING SCREEN-2



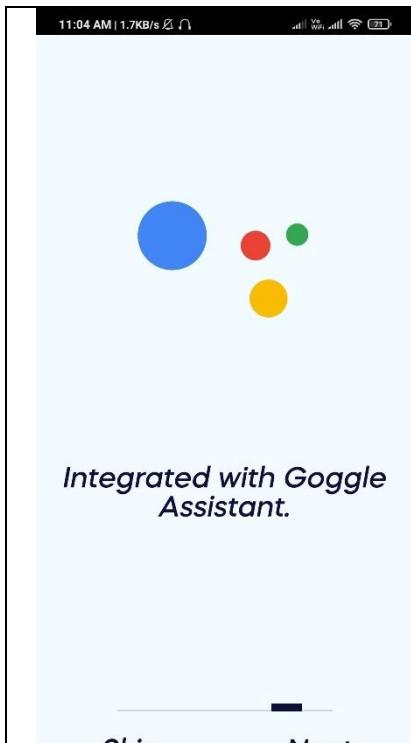
ONBOARDING SCREEN-3



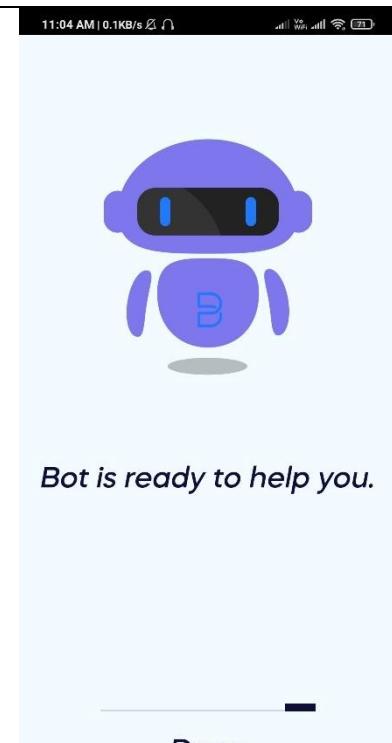
ONBOARDING SCREEN-4



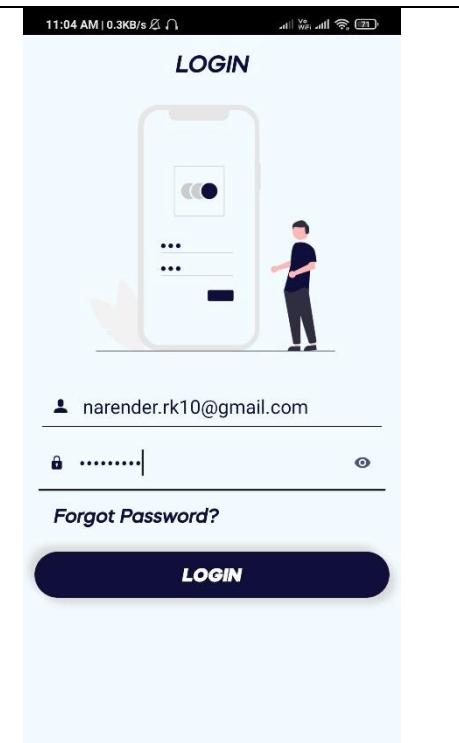
ONBOARDING SCREEN-5



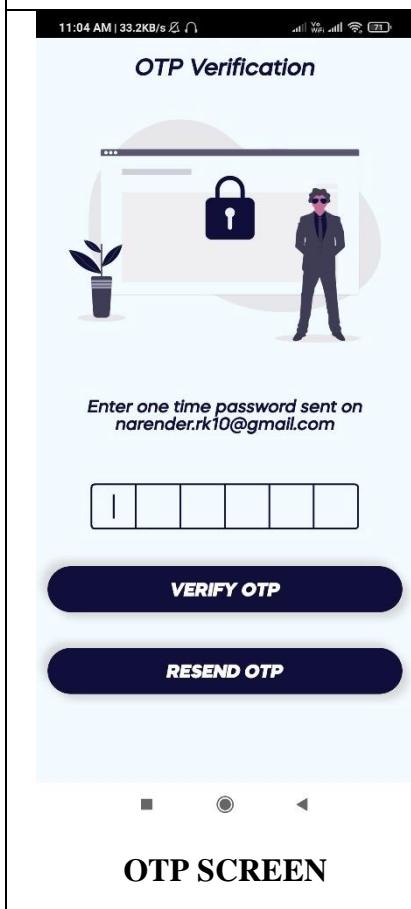
ONBOARDING SCREEN-6



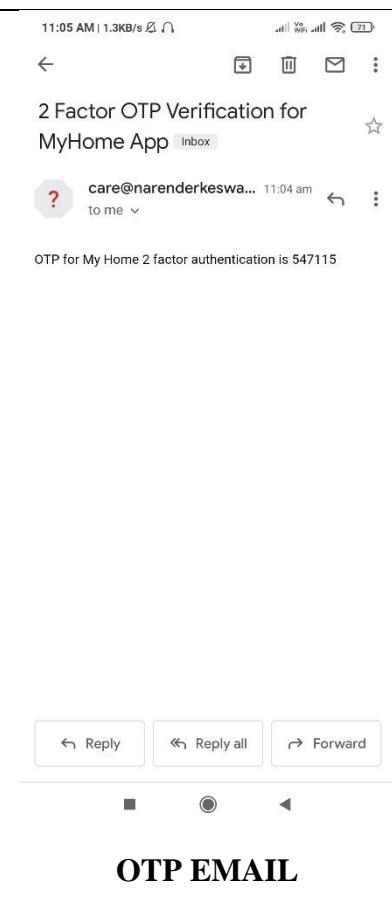
ONBOARDING SCREEN-7



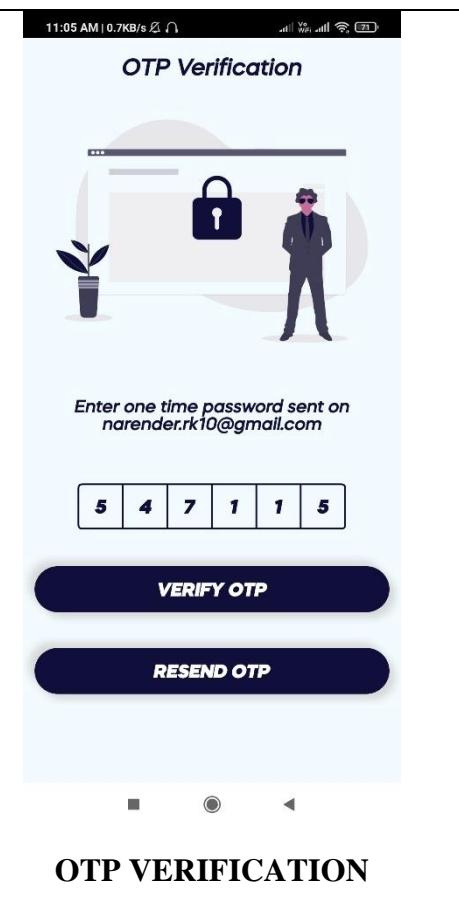
LOGIN SCREEN



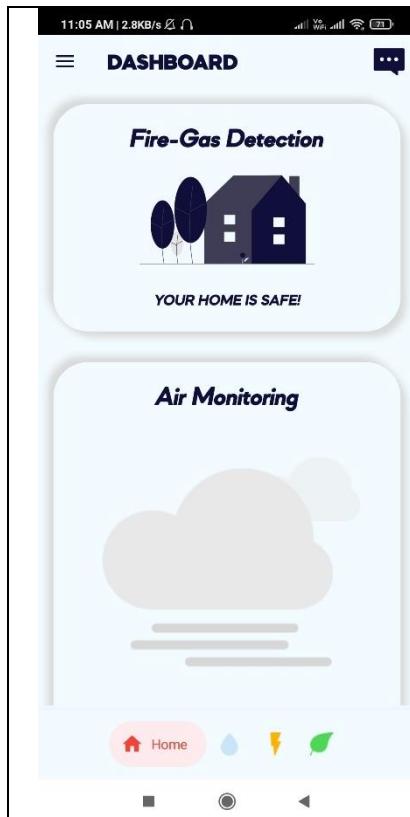
OTP SCREEN



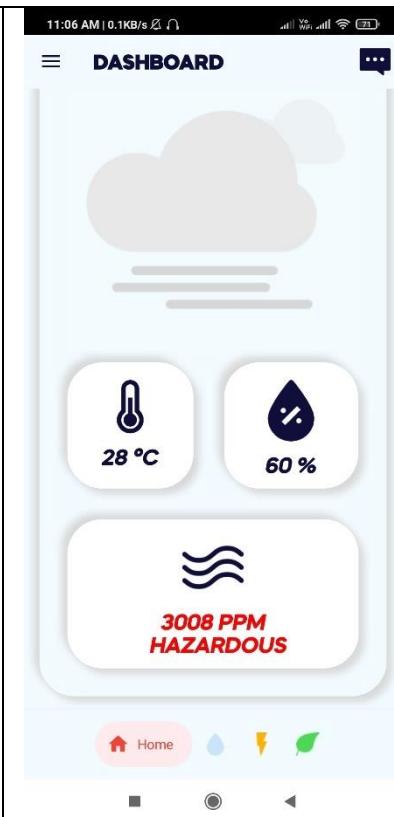
OTP EMAIL



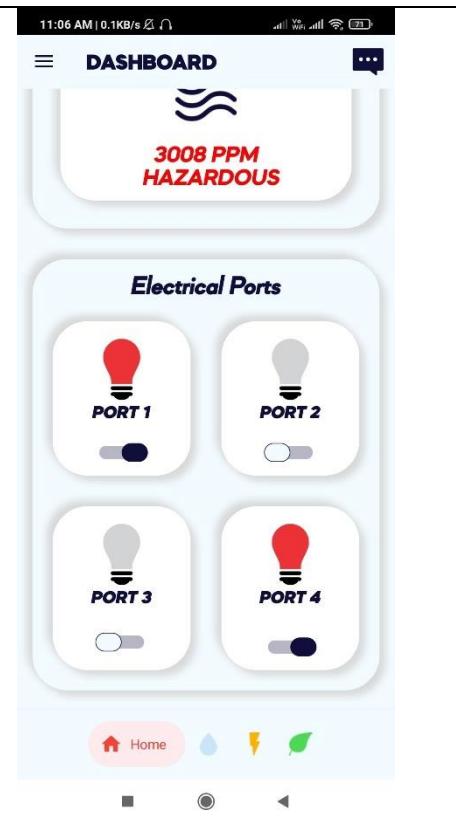
OTP VERIFICATION



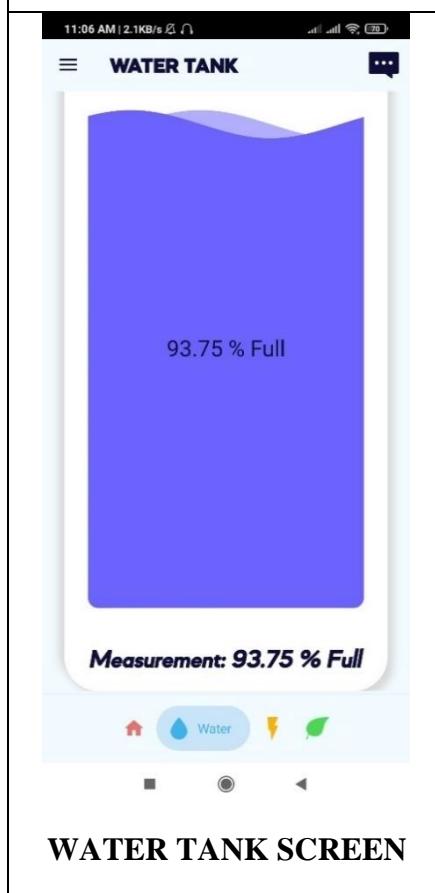
DASHBOARD SCREEN-1



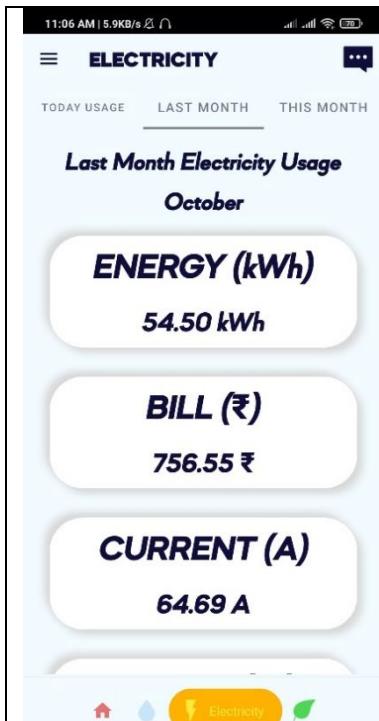
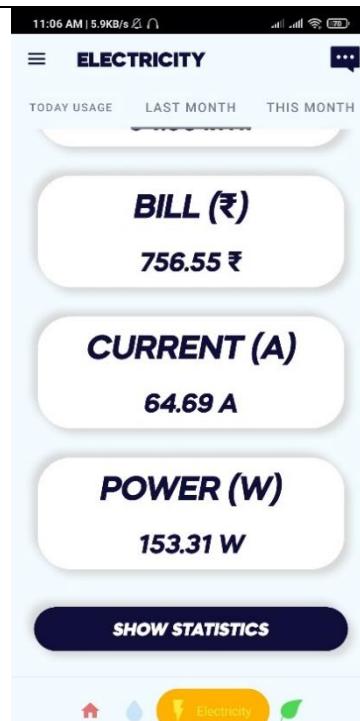
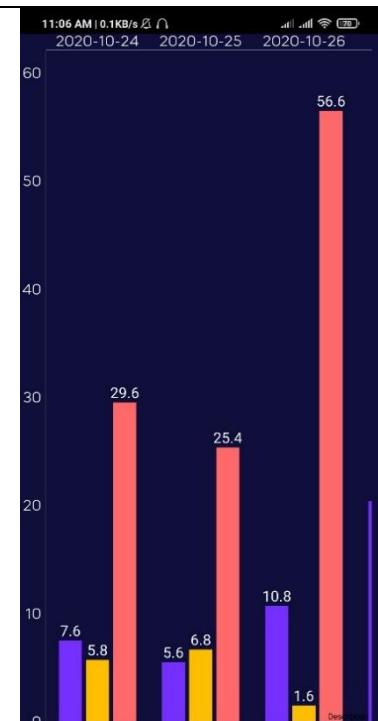
DASHBOARD SCREEN-2



DASHBOARD SCREEN-3



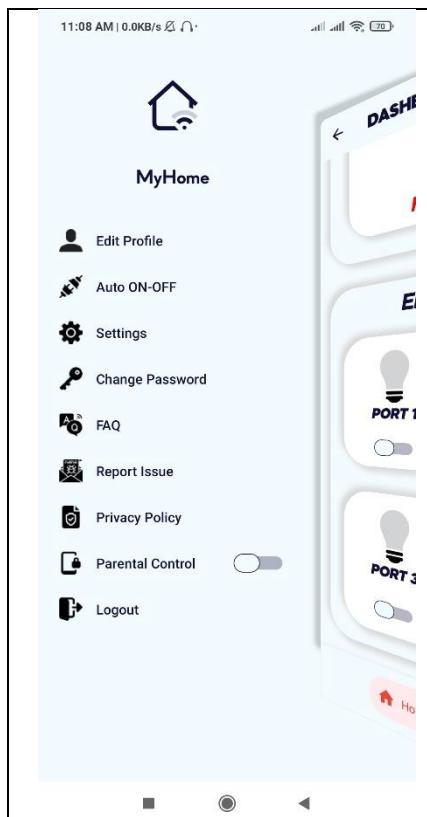
WATER TANK SCREEN

LAST MONTH
ELECTRICITY -1LAST MONTH
ELECTRICITY -2LAST MONTH
ELECTRICITY GRAPH

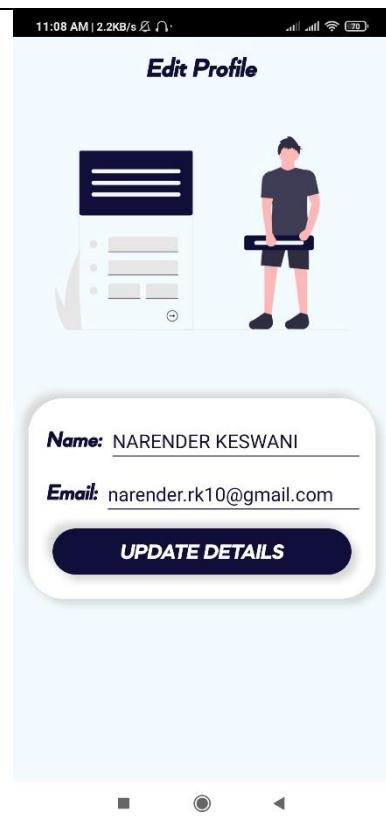
PLANT SCREEN-1



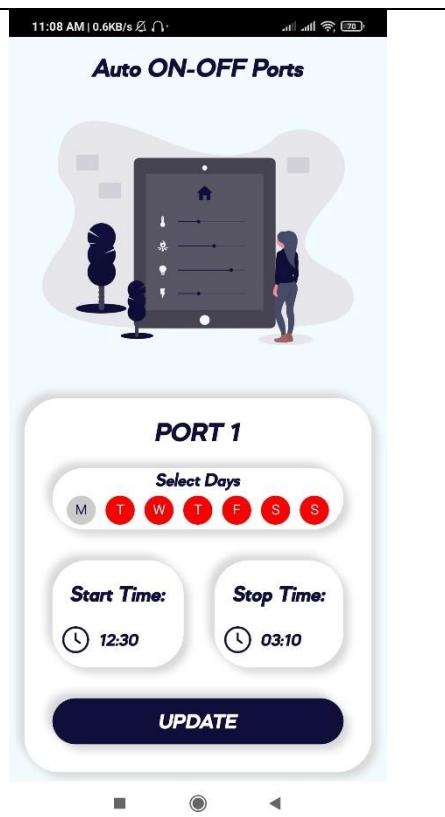
CHATBOT SCREEN



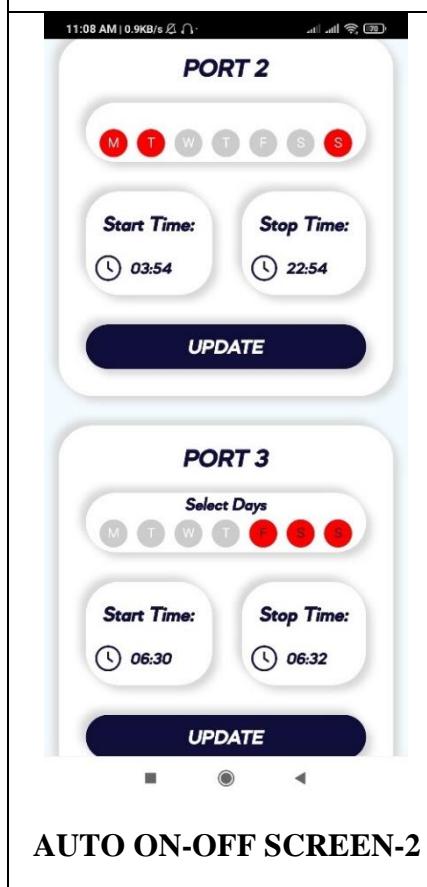
NAVIGATION DRAWER SCREEN



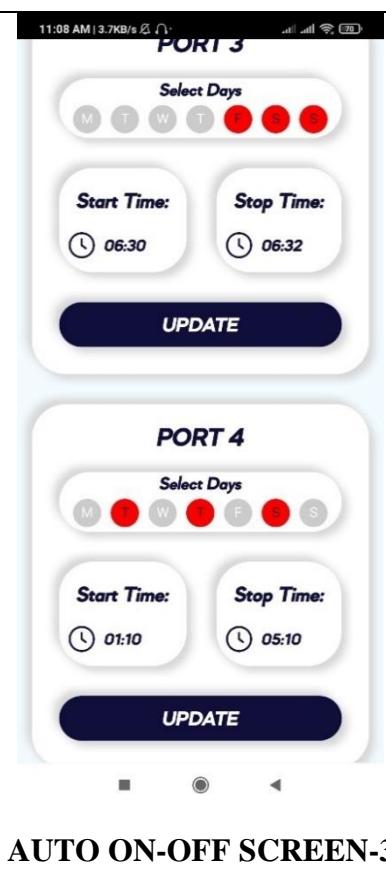
EDIT PROFILE SCREEN



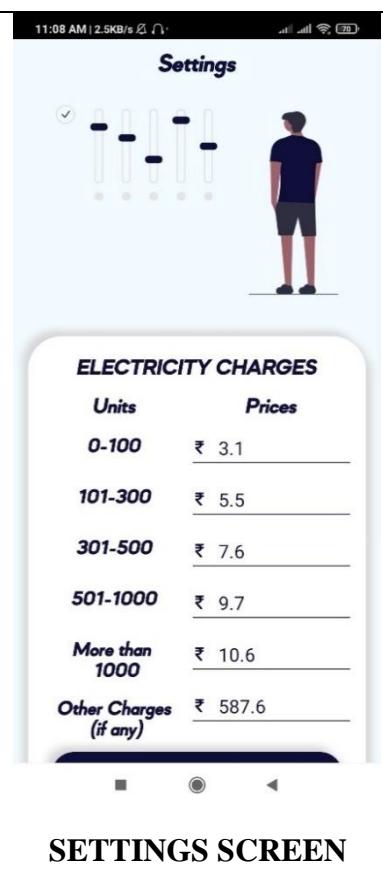
AUTO ON-OFF SCREEN-1



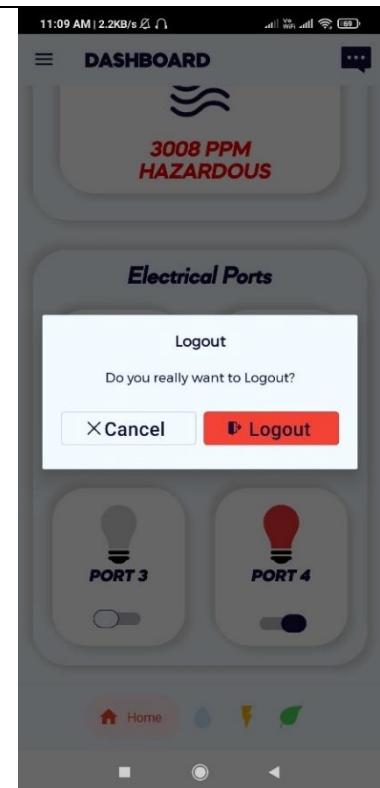
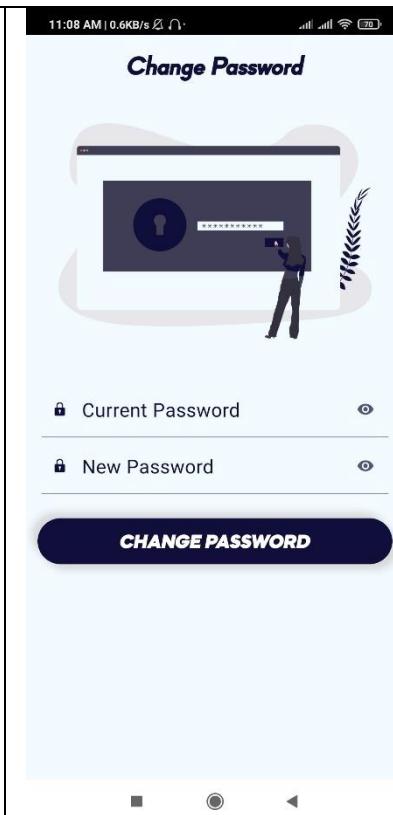
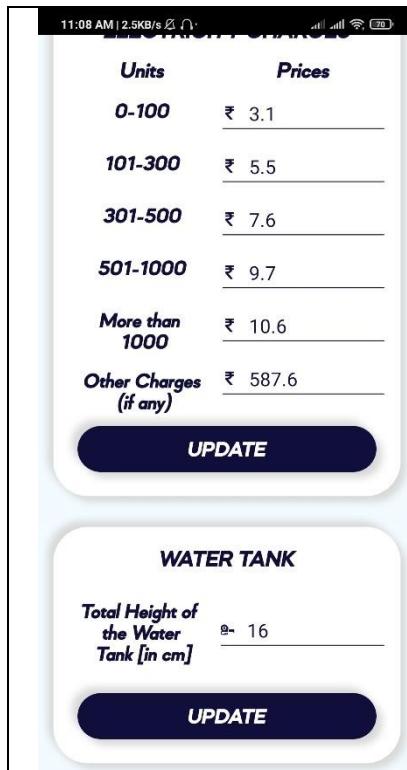
AUTO ON-OFF SCREEN-2



AUTO ON-OFF SCREEN-3



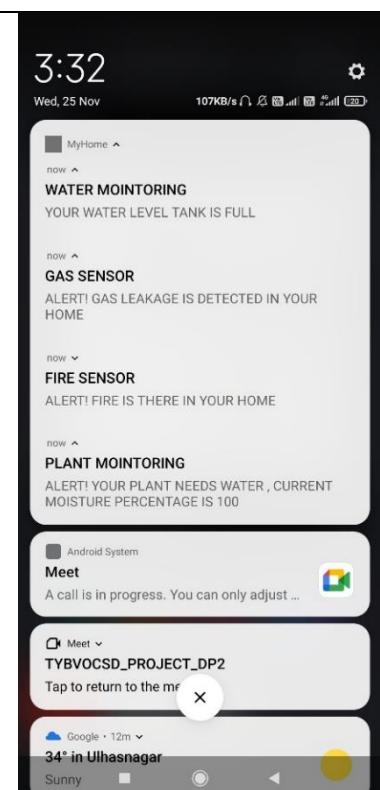
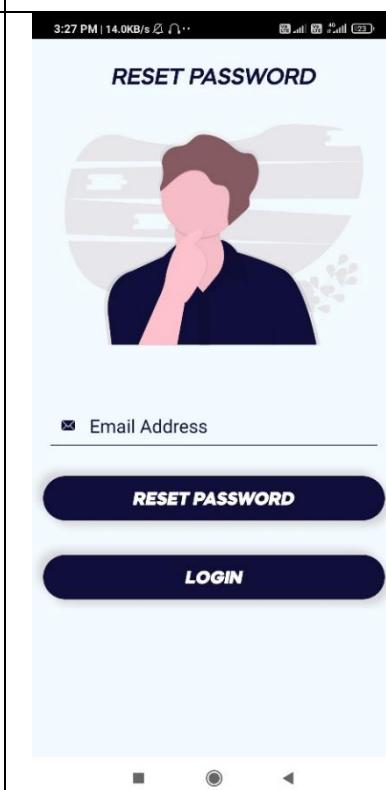
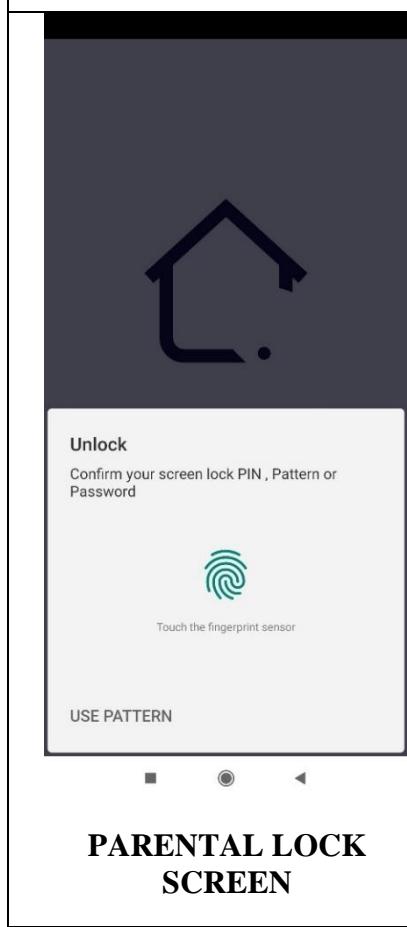
SETTINGS SCREEN



SETTINGS SCREEN-2

CHANGE PASSWORD SCREEN

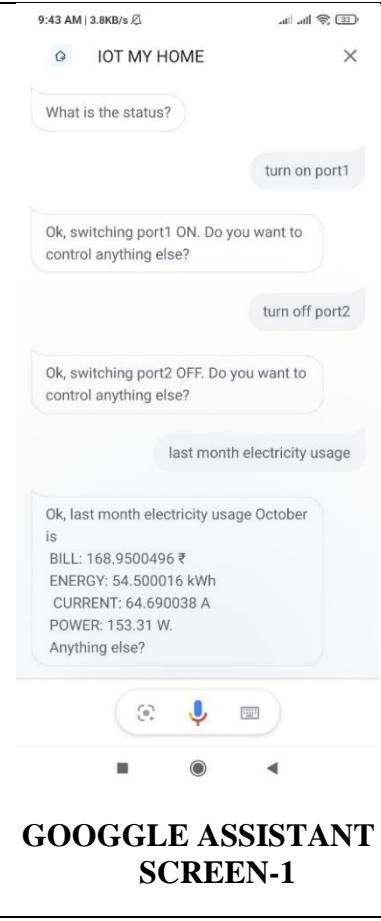
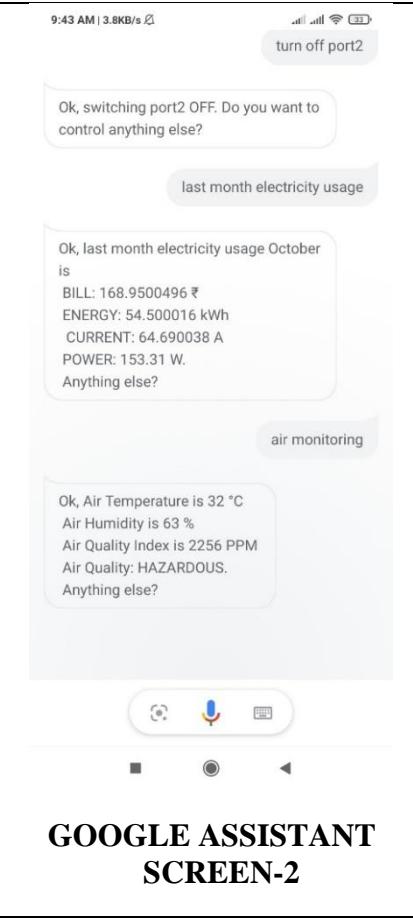
LOGOUT SCREEN



PARENTAL LOCK SCREEN

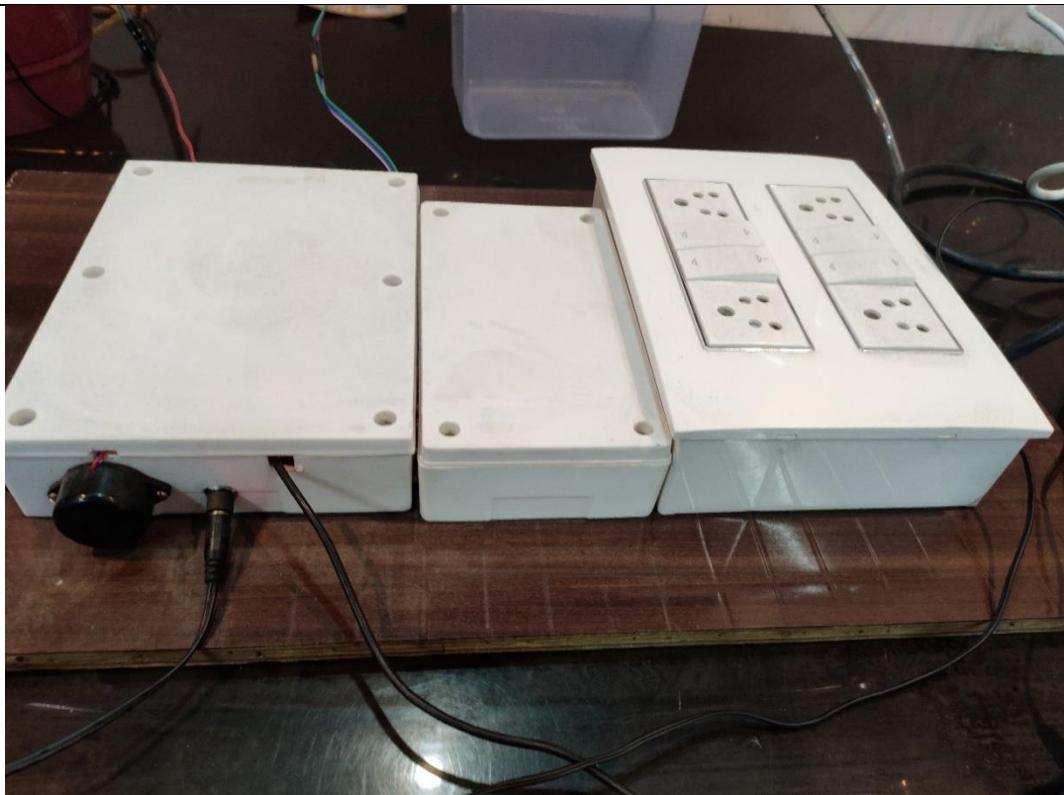
FORGOT / RESET PASSWORD SCREEN

NOTIFICATIONS SCREEN

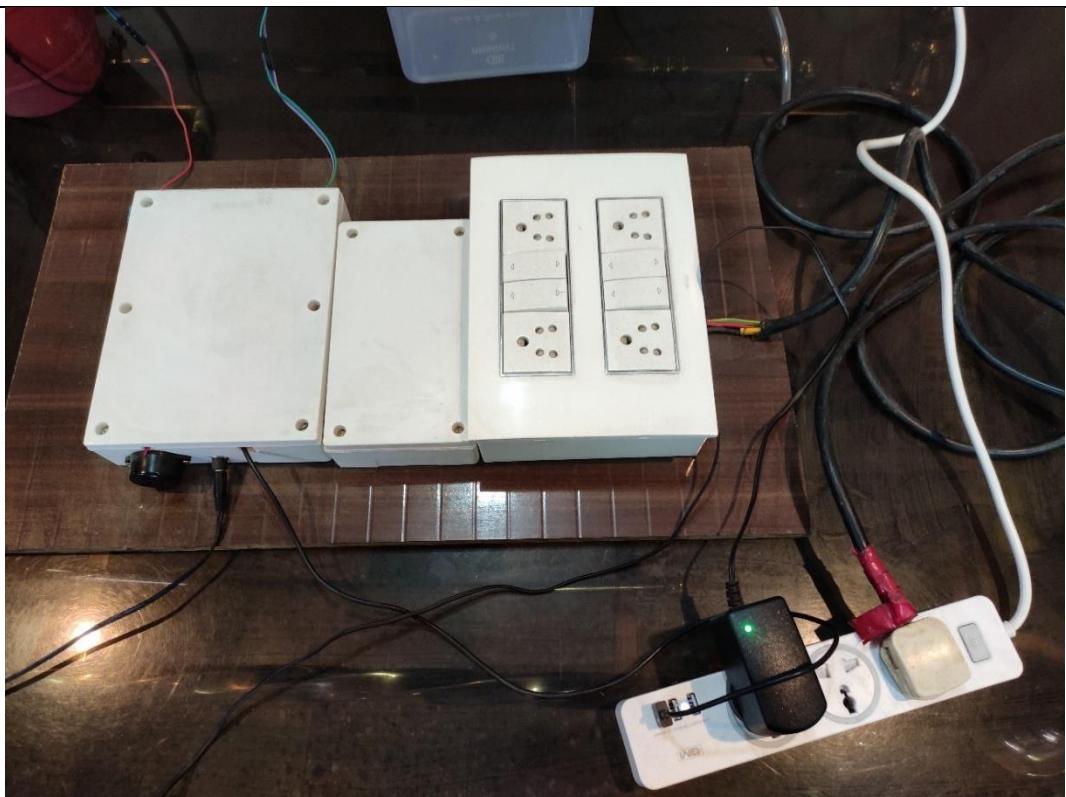
 <p>TELEGRAM BOT SCREEN</p>	 <p>GOOGLE ASSISTANT SCREEN-1</p>	 <p>GOOGLE ASSISTANT SCREEN-2</p>
IOT IMPLEMENTATION		



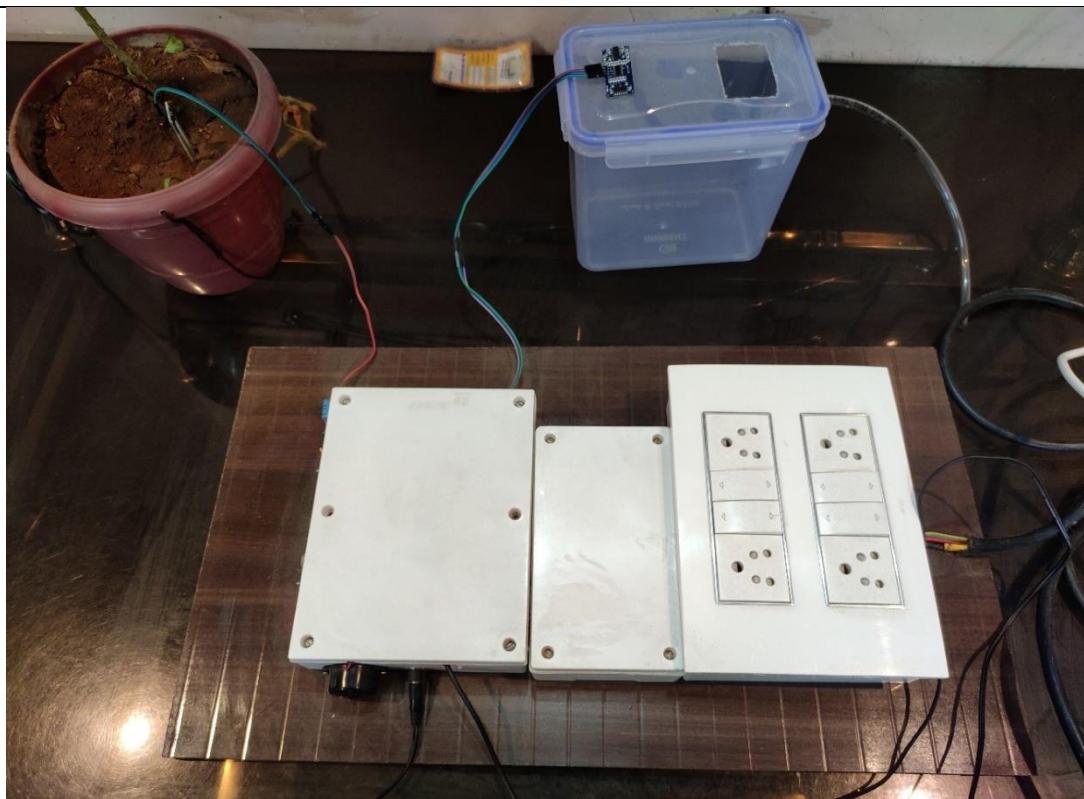
IOT DEVICE IMPLEMENTATION -1



IOT DEVICE IMPLEMENTATION-2



IOT DEVICE IMPLEMENTATION-3



IOT DEVICE IMPLEMENTATION-4

13. FUTURE ENHANCEMENT

- CCTV footage feature on the android phone.
- Adding Burglar detection system.
- Making smart colourful led bulb, which allows users to select colour, and particular colour will glow.
- Creating a smart locker and smart door. [password & fingerprint based]
- Create same app for the IOS users.

14. BIBLIOGRAPHY

- <https://medium.com/topic/programming/>
- <https://firebase.google.com/>
- <https://developer.android.com/>
- <https://randomnerdtutorials.com/>
- <https://www.instructables.com/>
- <https://www.arduino.cc/>
- <https://stackoverflow.com/>