

Emertxe

Emertxe Information technologies (p) ltd

IOT Based Home Automation Project

Presented By : TIMMAREDDY DEEKSHITHA

Indian Institute Of Information Technology(IIIT),Surat, Gujarat

Aim of this project-

- **Home Automation using Arduino (Picsimlab) and Blynk application.**
- **Controlling Garden Lights.**
- **Controlling fans and heater** based on whether it is hot or cold and also displaying the notifications on lcd.
- **Controlling the water level of the tank** based on its capacity and also sending respective notifications to lcd and applications.

What we learnt in this internship-

➤ WEEK-1:

- C-programming and its Data Types.
- Different types of operations and Conditional statements.
- Pointers and Functions.
- Storage classes and Strings manipulation.

c programming-

- It is a language which is the foundation for many popular programming languages.
- Syntax contains basic keywords and structures like functions and variables.
- Compiler translates c code into machine readable instructions.
- Debugging: identify and fix errors in the code.
- Int , short , long , float , double , char are the data types and variables.
- Arithmetic operators , relational operators , logical operators , bitwise operators are the operators and expressions in C.
- There are conditional statements like (if , else , else if) , loops (for , while , do-while) , switch statements to handle multiple cases efficiently.

What we learnt in this internship-

➤WEEK-2:

- Introduction to C++ classes such as object creation.
- Constructors and Destructors.
- Introduction to IOT and its applications.
- Installation of Picsimlab , Null modem emulator , Arduino IDE and all the required libraries.

Week-3 and Week-4 , we started building our Home automation Project.

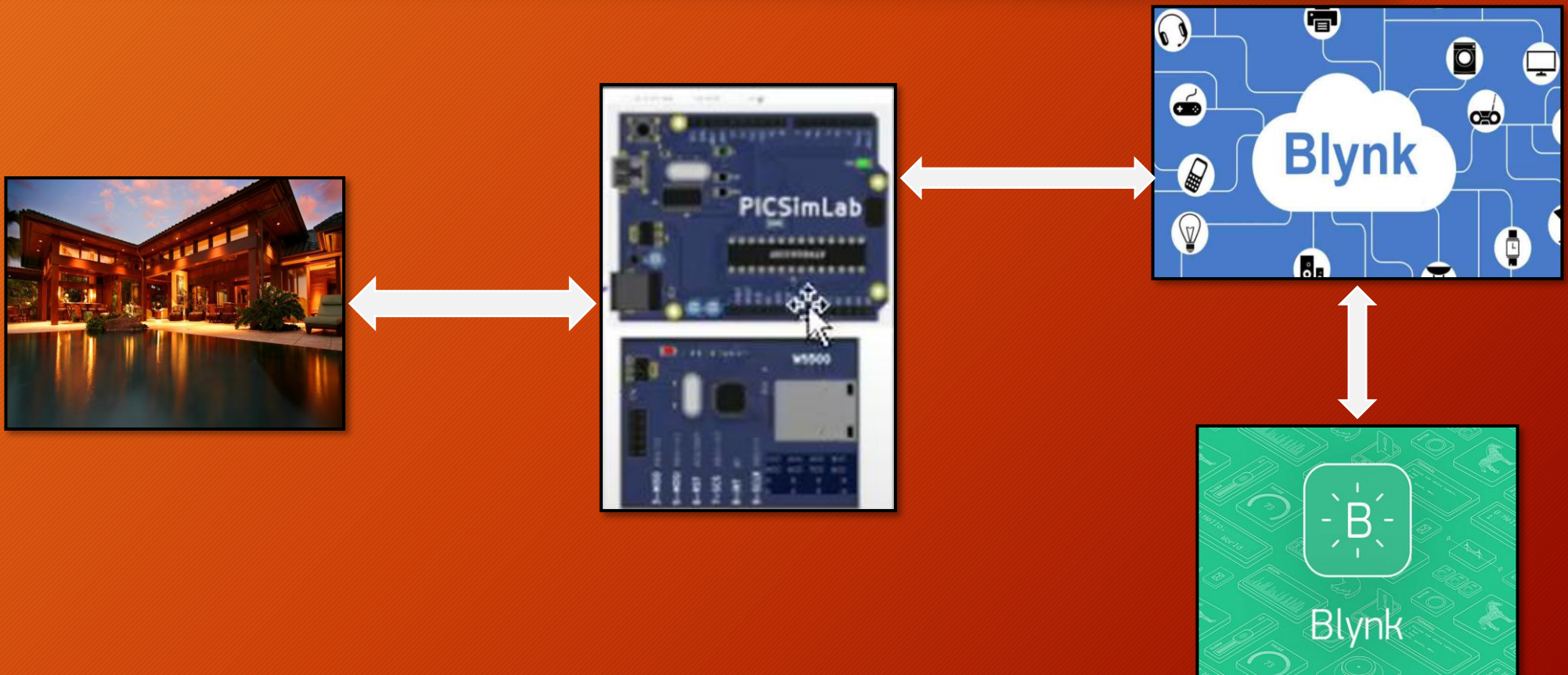
C++ PROGRAMMING

- It is a powerful , general purpose programming language . It's widely used in various applications , from game development to operating systems.
- It is an object oriented programming language.
- It contains complex syntax and also more features.
- It is more resource intensive and is used for application development.
- It supports classes and also contains inheritance.
- It also contains advanced data types.

Overview of this project:-

- All the peripherals are simulations only and no real time objects are used.
- We are using ***Blynk Mobile app*** to control devices.
- The main purpose of this project is to bring up an IOT based Home Automation Solution for controlling appliances remotely to make our busy life more convenient.
- The home automation is simulated using ***picsimlab simulator***. In this project we should be able to control the lights, temperature of the home, inflow and outflow of water in the water tank using Blynk mobile app.

Overview:

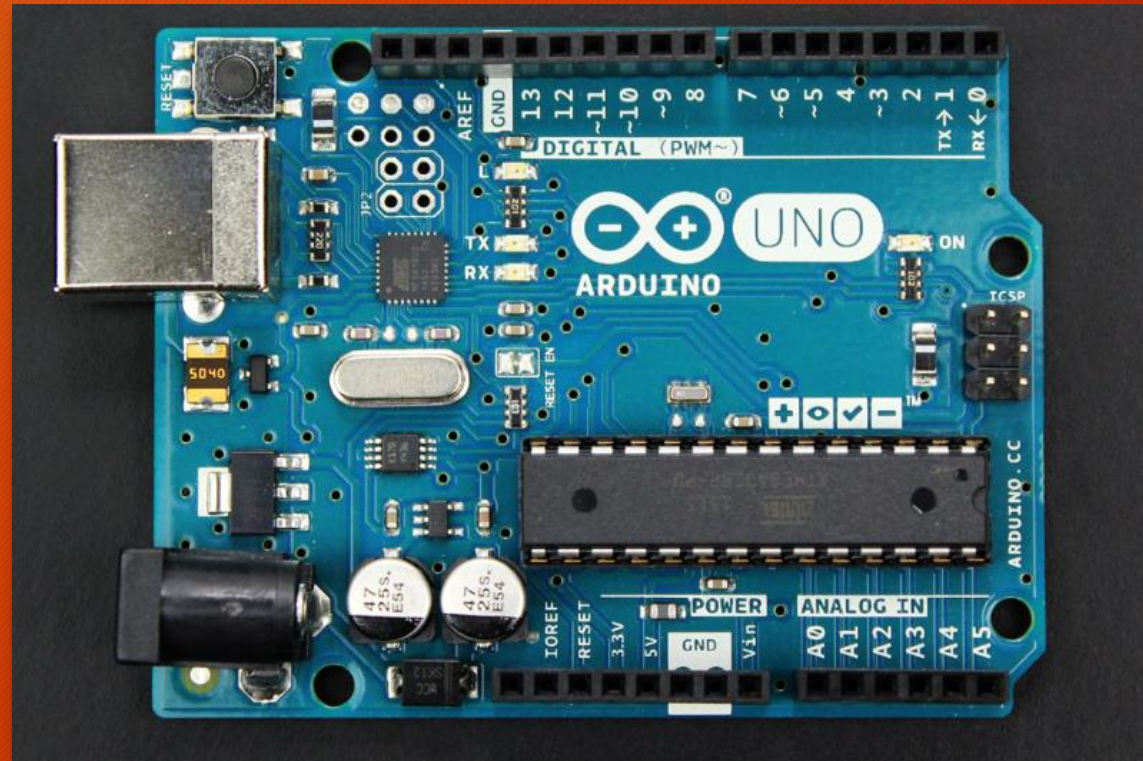


Let's get started



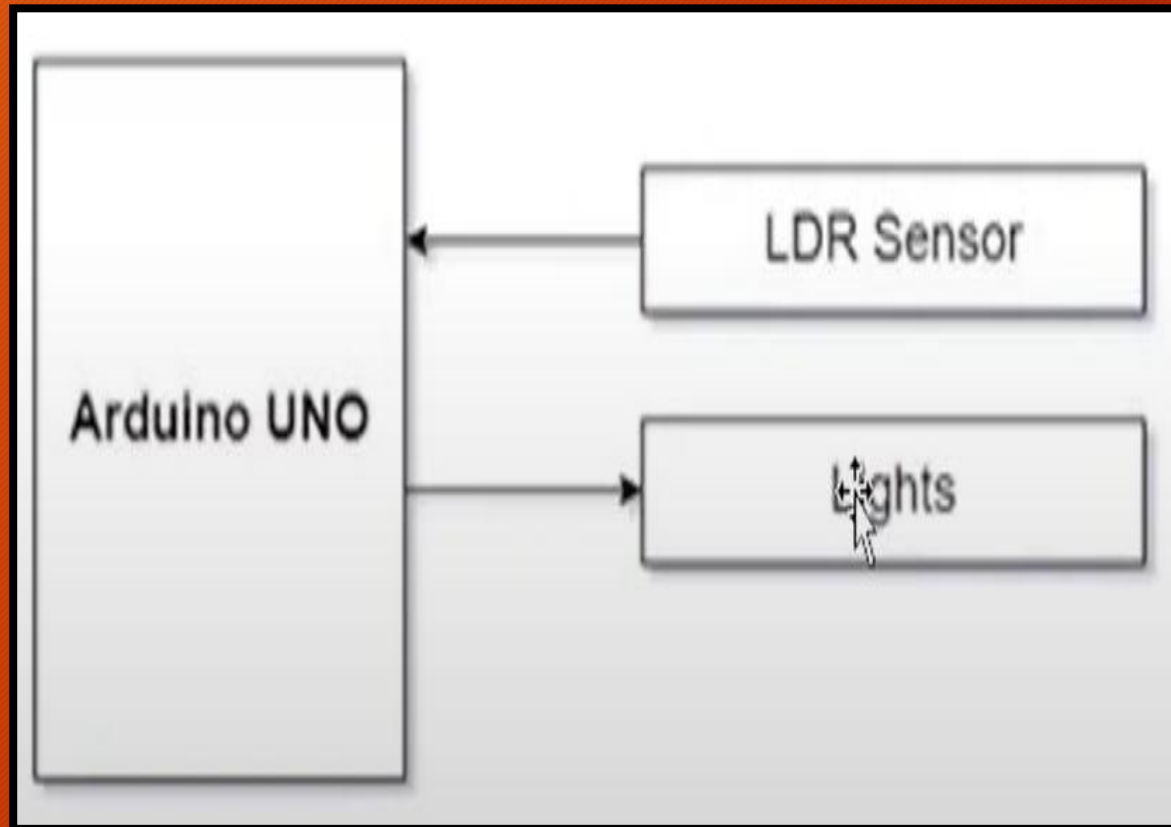
Arduino -

- Arduino IDE is the development environment for writing and uploading code to the Arduino board.
- Arduino boards comes in various models , each designed for specific applications.
- They feature a microcontroller, input/output pins and other components.



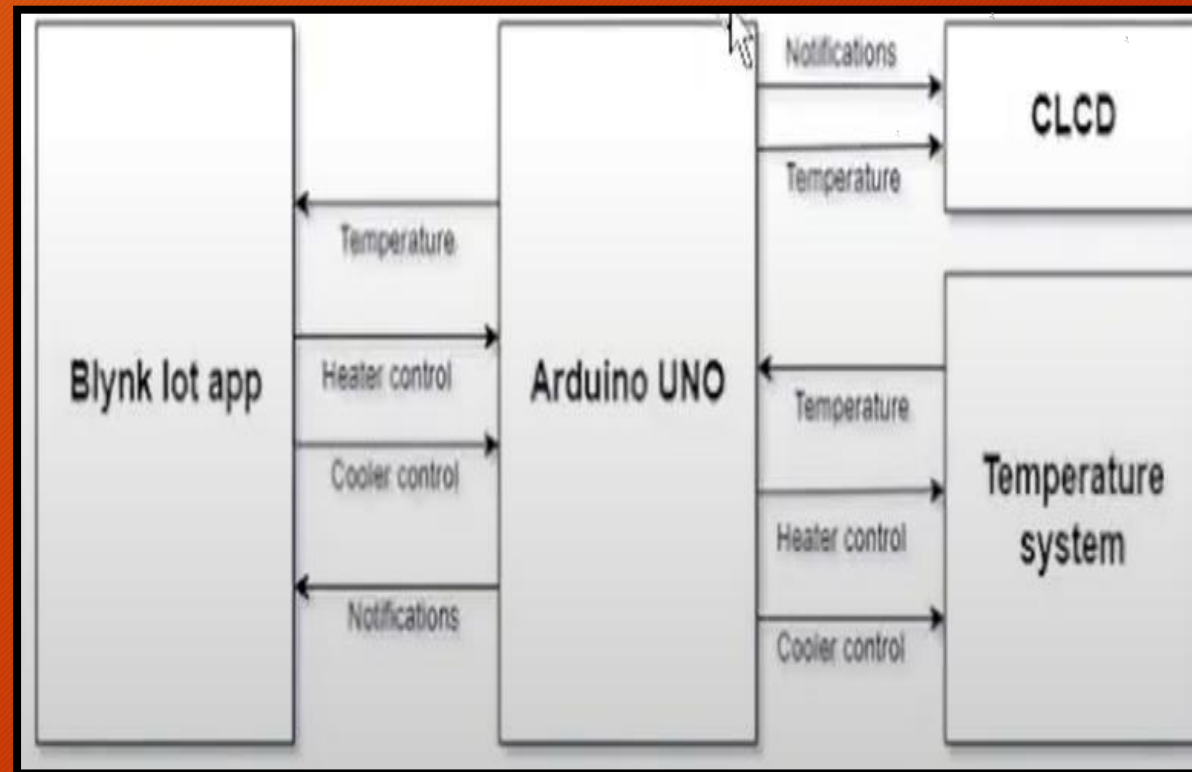
Garden lights control -

- Read the ldr sensor value, based on the reading from ldr , vary the brightness of the led , which resembles controlling garden lights.



Temperature control system

- It consists of heating resistor , an LM35 temperature sensor , and a cooler which resembles the temperature control system at home.
- The temperature from the temperature sensor LM35 and display it on the CLCD.
- Control the temperature of the turning ON/OFF the heater and cooler through the Blynk IOT mobile app.

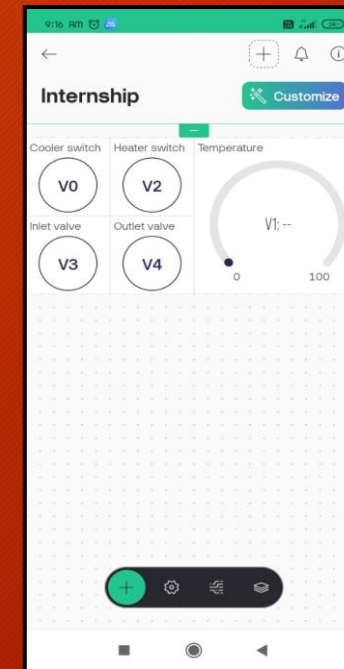
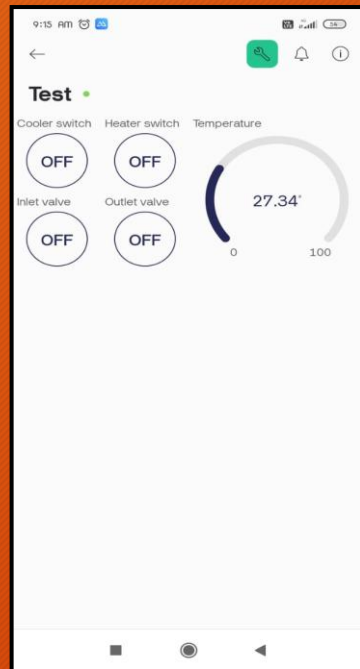


Water tank inlet and outlet valve control

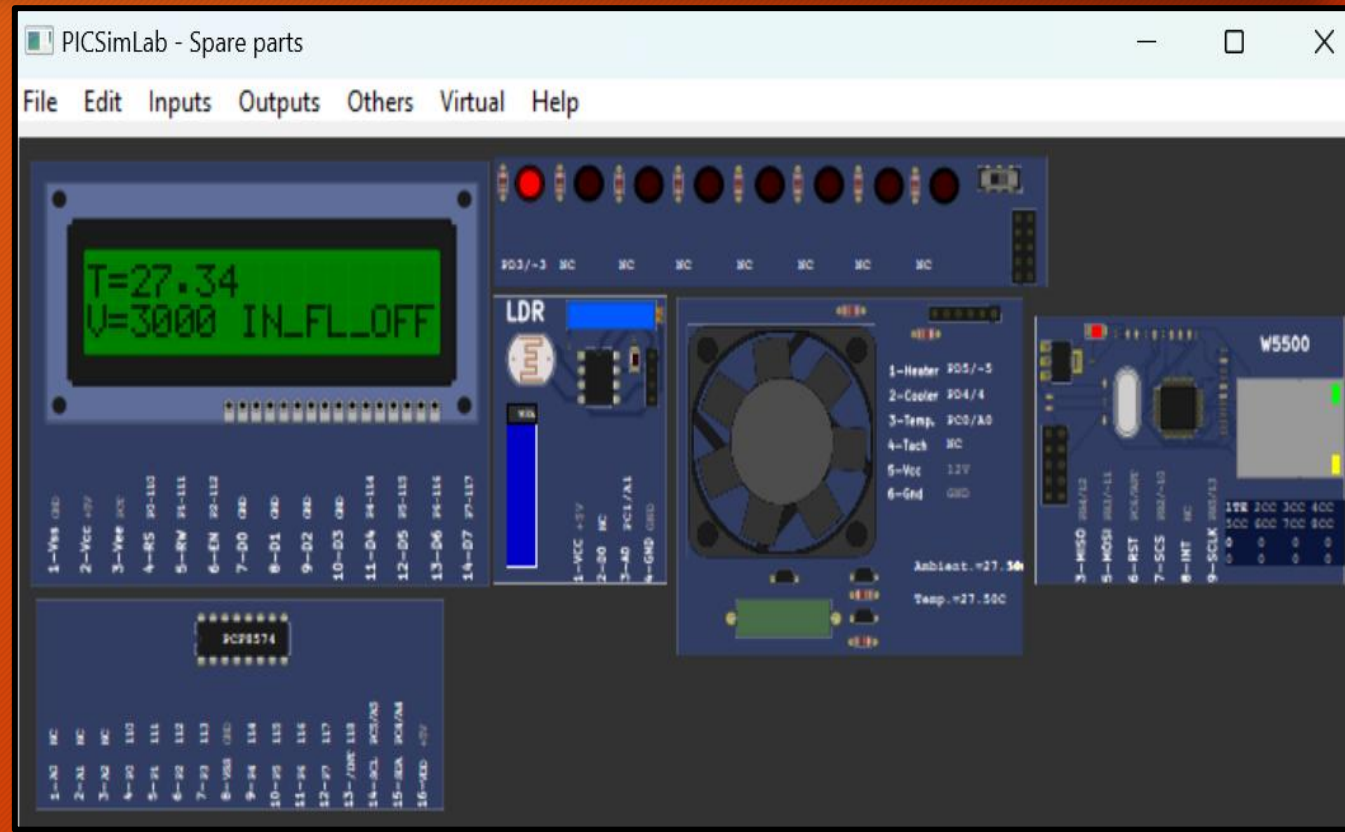
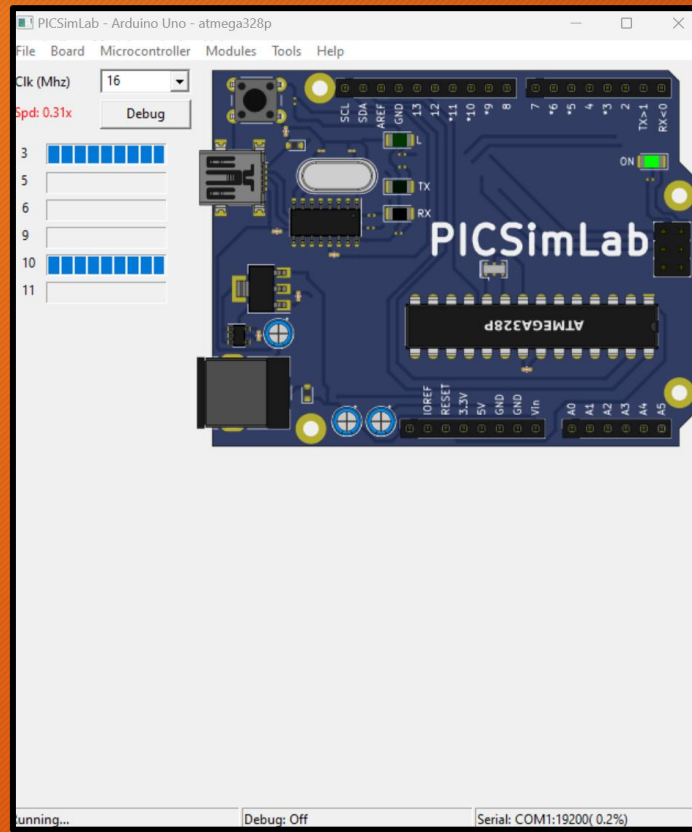
- Read the volume of the water in the tank through Serial Communication and display it on the CLCD , control the volume of the water in the tank by controlling the inlet and outlet valve . By sending commands through serial communication.
- Display the volume of the water in the tank on the CLCD.

Blynk iot app:-

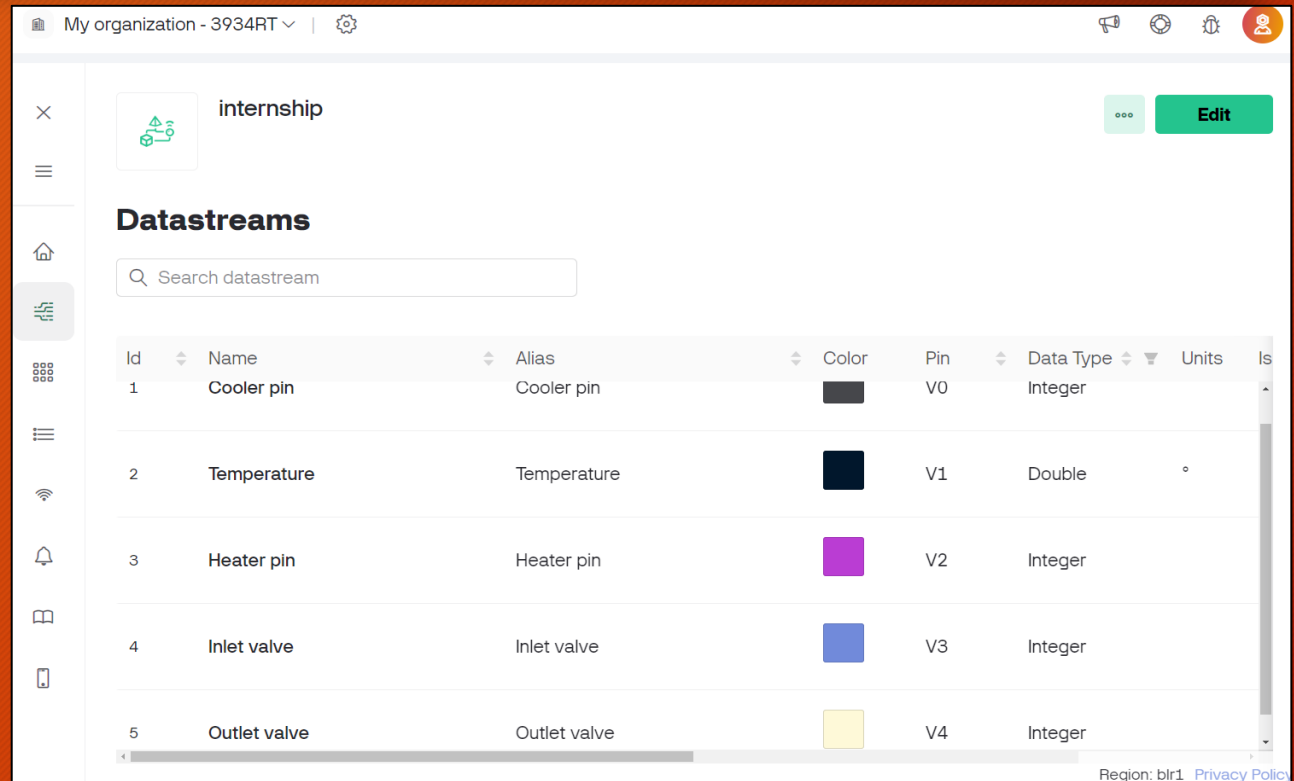
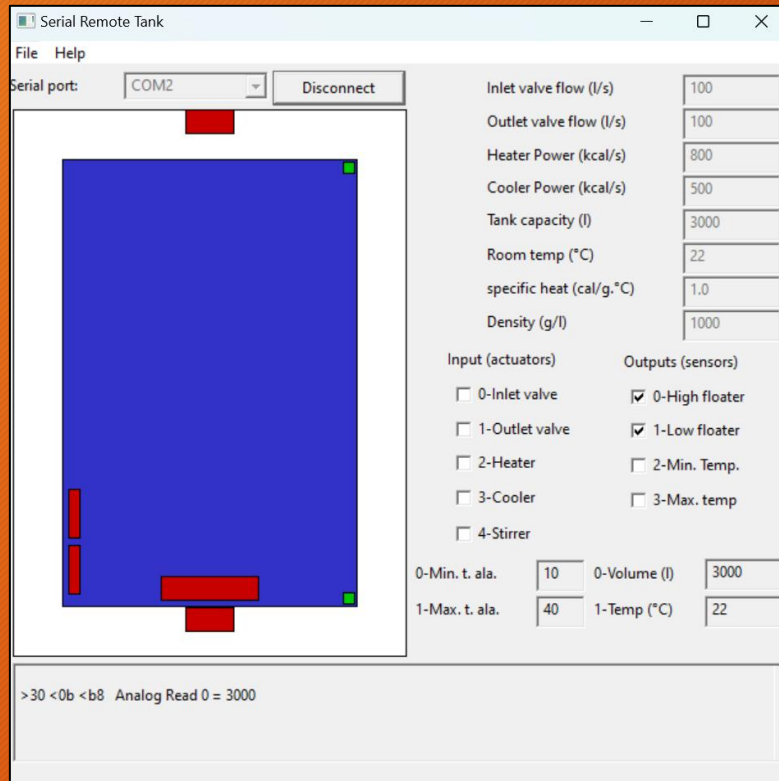
- Button widgets to control the heater, cooler and gauge widget to display the temperature.
- Button widgets to control the inlet valve, outlet valve and gauge to display volume of water in tank.



Project image :-



Project image:-



PROJECT IMAGE-CODE

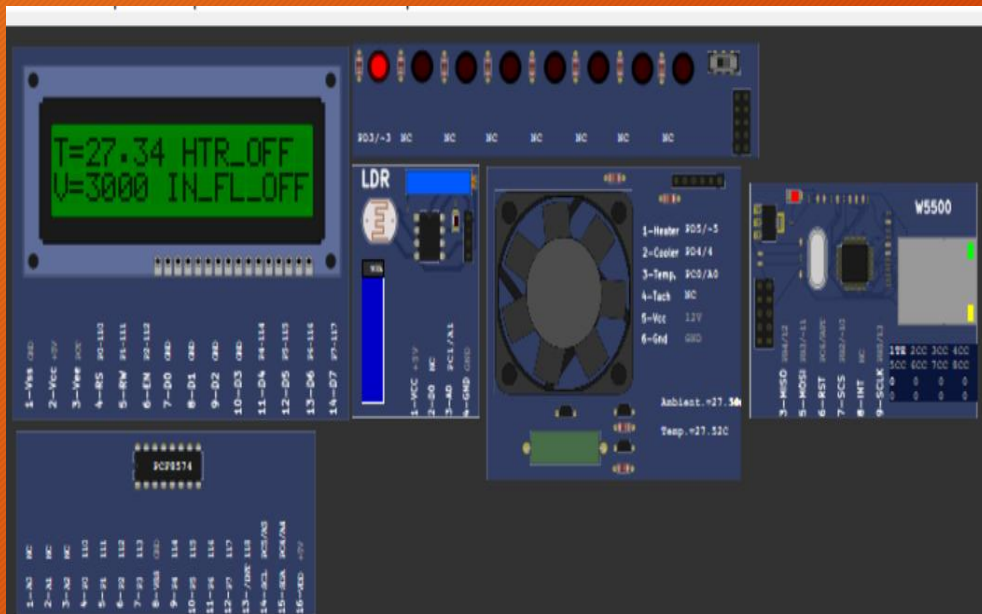
```
home_automation_blynk_controlled.ino ldr.cpp ldr.h main.h serial_tank.cpp serial_tank.h temperature_system.cpp temperature_system.h
17 #include <SPI.h>
18 #include <Ethernet.h>
19 #include <BlynkSimpleEthernet.h>
20 #include <Wire.h>
21 #include <LiquidCrystal_I2C.h>
22
23 #include "main.h"
24 #include "temperature_system.h"
25 #include "ldr.h"
26 #include "serial_tank.h"
27
28 char auth[] = BLYNK_AUTH_TOKEN;
29 bool heater_sw,inlet_sw,outlet_sw, cooler_sw;
30 unsigned int tank_volume;
31
32 BlynkTimer timer;
33
34 LiquidCrystal_I2C lcd(0x27, 16, 2); // set the LCD address to 0x27 for a 16 chars and 2 line display
35
36 // This function is called every time the Virtual Pin 0 state changes
37 /*To turn ON and OFF cooler based virtual PIN value*/
38 BLYNK_WRITE(COOLER_V_PIN)
39 {
40     cooler_sw = param.asInt();
41     if (cooler_sw){
42         cooler_control(ON);
43         lcd.setCursor(8,0);
44         lcd.print("CLR_ON ");
45     }
46 }
```

```
home_automation_blynk_controlled.ino ldr.cpp ldr.h main.h serial_tank.cpp serial_tank.h
47     else{
48         cooler_control(OFF);
49         lcd.setCursor(8,0);
50         lcd.print("CLR_OFF");
51     }
52 }
53 }
54 /*To turn ON and OFF heater based virtual PIN value*/
55 BLYNK_WRITE(HEATER_V_PIN )
56 {
57     heater_sw = param.asInt();
58     if(heater_sw){
59         heater_control(ON);
60         lcd.setCursor(8,0);
61         lcd.print("HTR_ON ");
62     }
63 }
64 else{
65     heater_control(OFF);
66     lcd.setCursor(8,0);
67     lcd.print("HTR_OFF");
68 }
69 }
70 /*To turn ON and OFF inlet vale based virtual PIN value*/
71 BLYNK_WRITE(INLET_V_PIN)
72 { /*to read the status of inlet button*/
73     inlet_sw = param.asInt();
74     if(inlet_sw){
75         /*turn on the inlet valve*/
```

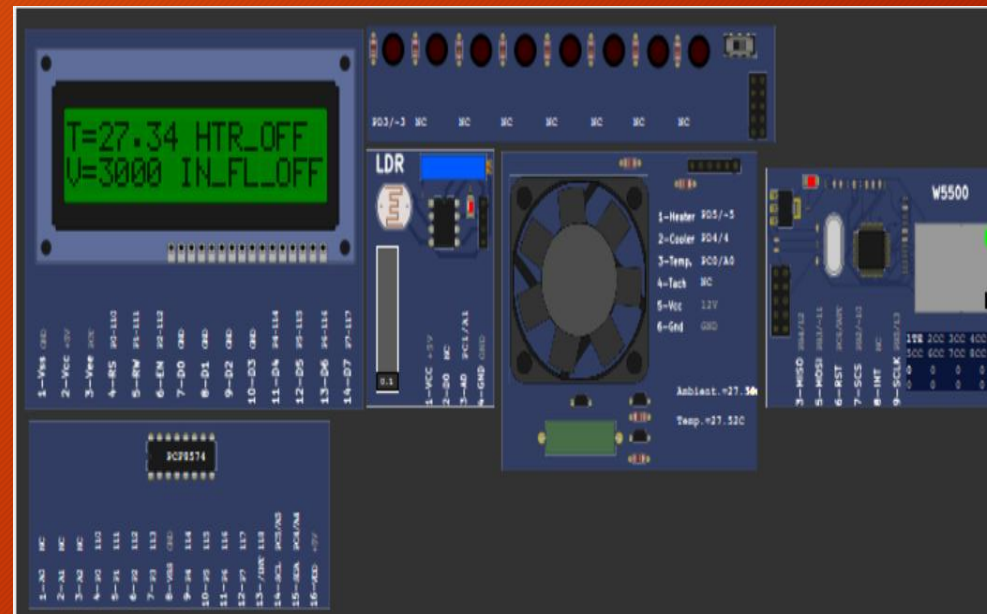
```
home_automation_blynk_controlled.ino ldr.cpp ldr.h main.h serial_tank.cpp serial_tank.h
166     Blynk.begin(BLYNK_AUTH_TOKEN);
167     init_ldr();
168     init_temperature_system();
169     init_serial_tank();
170     /*update temperature to the blynk app every one second*/
171     timer.setInterval(1000,update_temperature_reading);
172 }
173 }
174 String temp;
175 void loop(void)
176 {
177     /*read the value from ldr and control led brightness*/
178     brightness_control();
179     /*read the temp and convert into string and display it on clcd*/
180     temp = String(read_temperature(), 2); //reading 2 fields
181     lcd.setCursor(2,0);
182     lcd.print(temp);
183     /*READ THE VOLUME AND DISPLAY IT ON THE CLCD*/
184     tank_volume = volume();
185     lcd.setCursor(2,1);
186     lcd.print(tank_volume);
187     /*to connect the device continuously to the cloud*/
188     Blynk.run();
189     /*turn on the timer*/
190     timer.run();
191     handle_temp();
192     handle_tank();
193 }
```

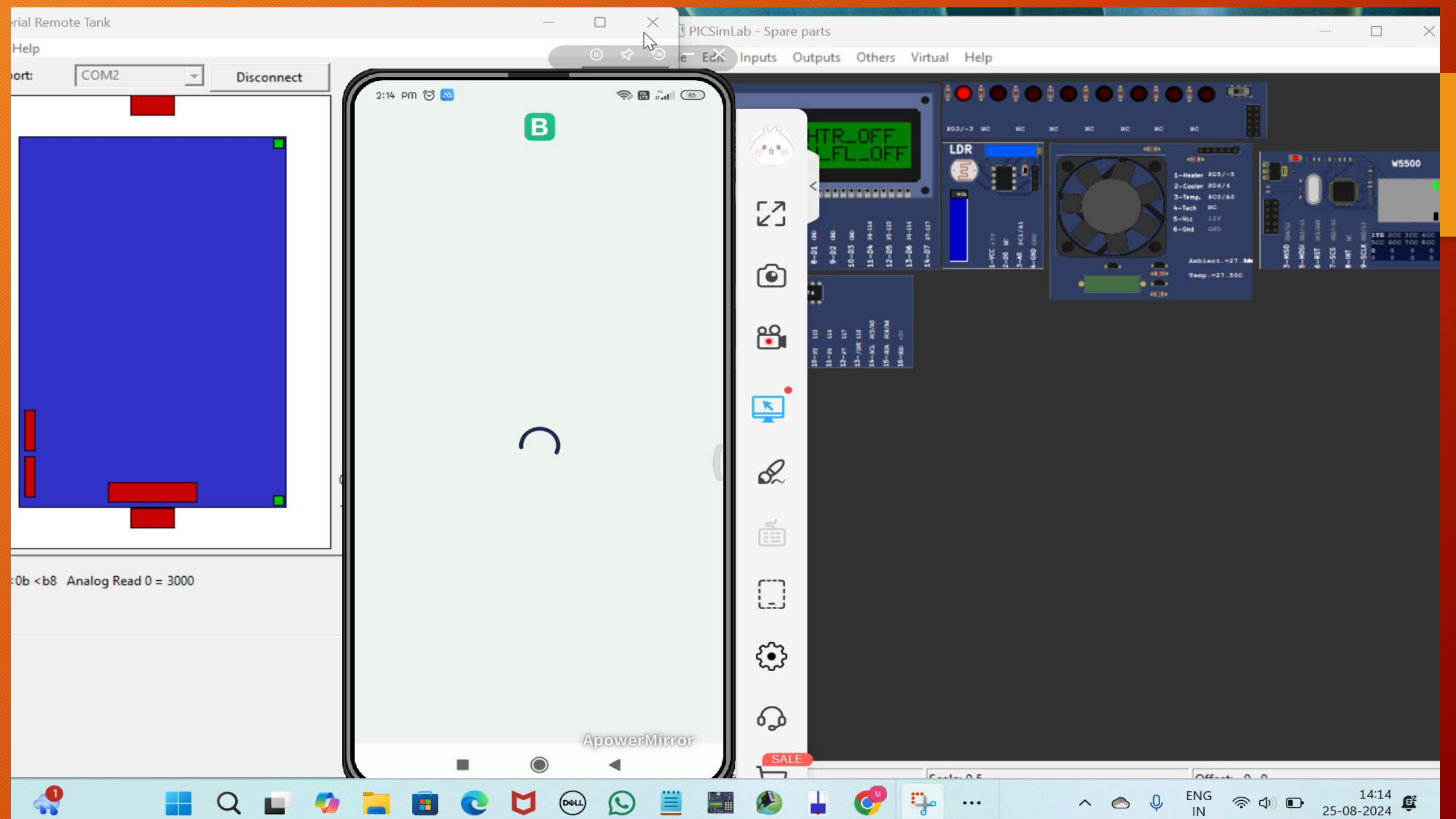

LDR-

When ldr value is high



When ldr value is low





Thank
you

