20CSL62 INTERNET OF THINGS AND CLOUD LABORATORY

IOT BASED HOME AUTOMATION USING TELEGRAM BOT

A MINI PROJECT REPORT

Submitted by

PAVITHRA J
20CSR145

PRIYA R
20CSR154

PRIYADHARSHINI S
20CSR157

in partial fulfillment of the requirements

for the award of the degree

of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KONGU ENGINEERING COLLEGE

(Autonomous)

PERUNDURAI ERODE – 638 060

ABSTRACT

Home automation has gained significant popularity in recent years, enabling homeowners to control and monitor various devices and systems within their homes remotely. The proposed system allows users to interact with their home automation system using the Telegram messaging app, providing a user-friendly and accessible interface. By leveraging the Telegram bot API and integrating it with the home automation infrastructure, users can remotely control and monitor devices such as lights, thermostats, security systems, and appliances with ease. The Telegram bot acts as a bridge between the user and the home automation system, enabling seamless communication and control. Users can issue commands through simple text messages or use predefined buttons and menus within the bot interface to perform various automation tasks. The system also supports real-time status updates, allowing users to receive notifications and alerts regarding their home's state and activities.Furthermore, the integration of Telegram's security features ensures a robust and secure communication channel between the user and the home automation system. End-to-end encryption and user authentication mechanisms provide an additional layer of privacy and protection for the user's commands and data.The effectiveness of the proposed system was evaluated through a series of experiments, demonstrating its reliability, responsiveness, and ease of use. The results indicate that the integration of home automation with a Telegram bot offers a convenient and efficient solution for controlling and monitoring smart devices within a home remotely.

COMPONENTS REQUIRED

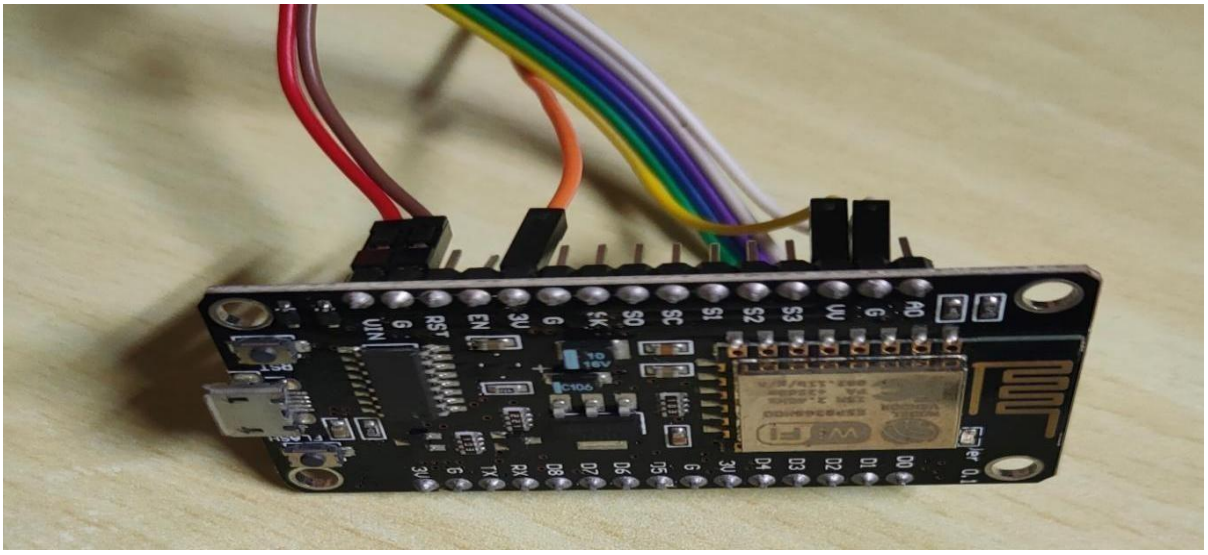
- **NODE MCU** – Node MCU is an open-source platform based on ESP8266 which can connect objects and let data transfer using the Wi-Fi protocol.

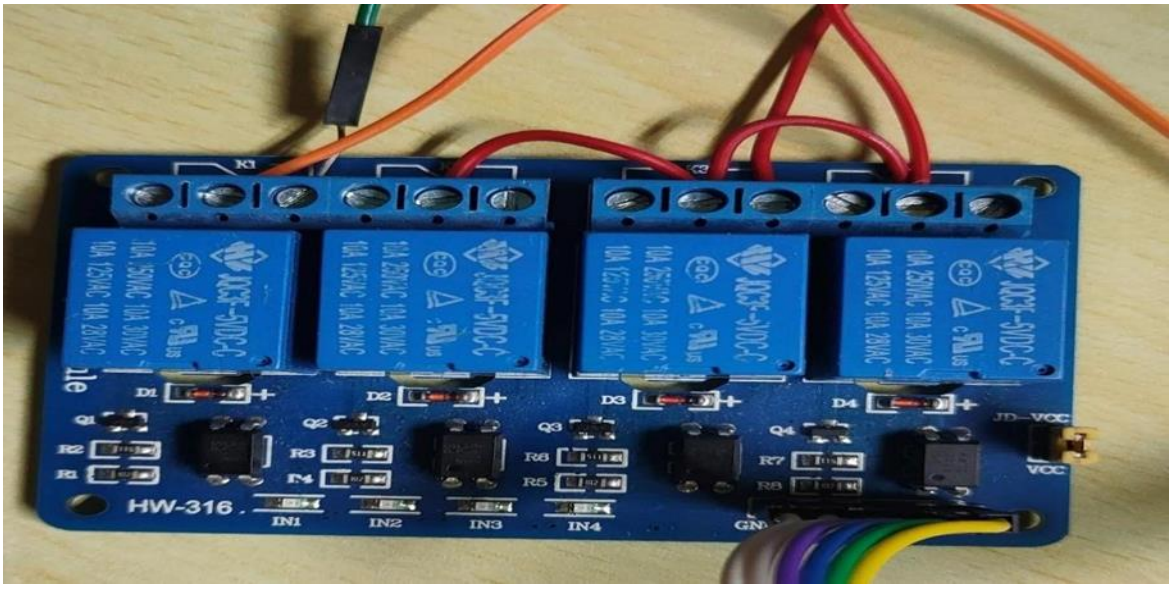
- **FOUR CHANNEL RELAY** – A 4-channel relay is an electromechanical device that allows you to control four different circuits using a single device.

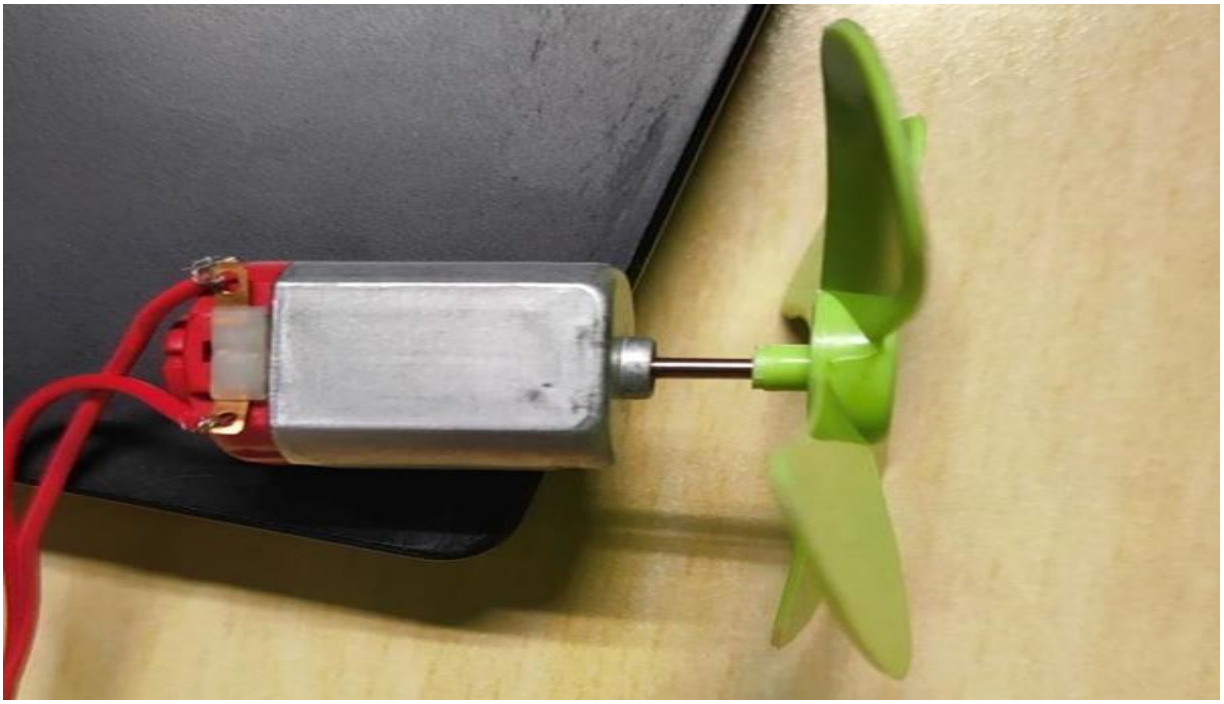
- **JUMPER WIRES** – A jumper wire is an electric wire that connects remote electric circuits used for printed circuit boards.

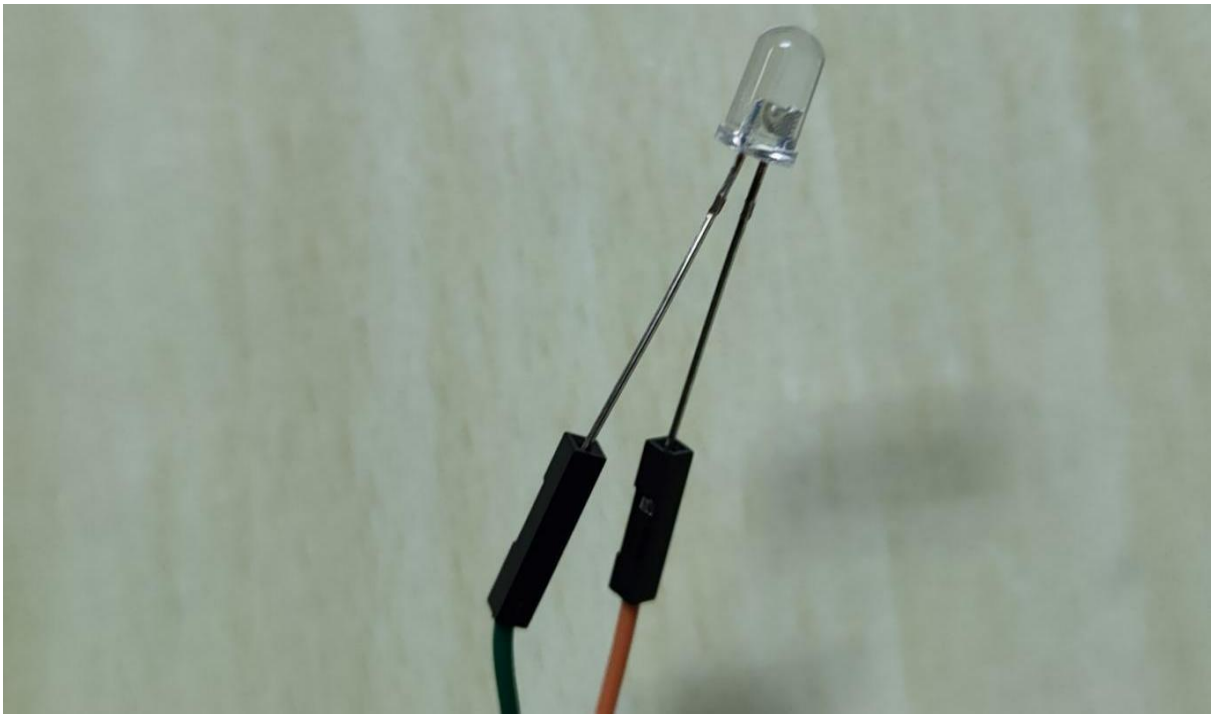
- **LIGHT EMITTING DIODE (LED)** – semiconductor device, which can emit light when an electrical current passes through it.

HARDWARE SETUP









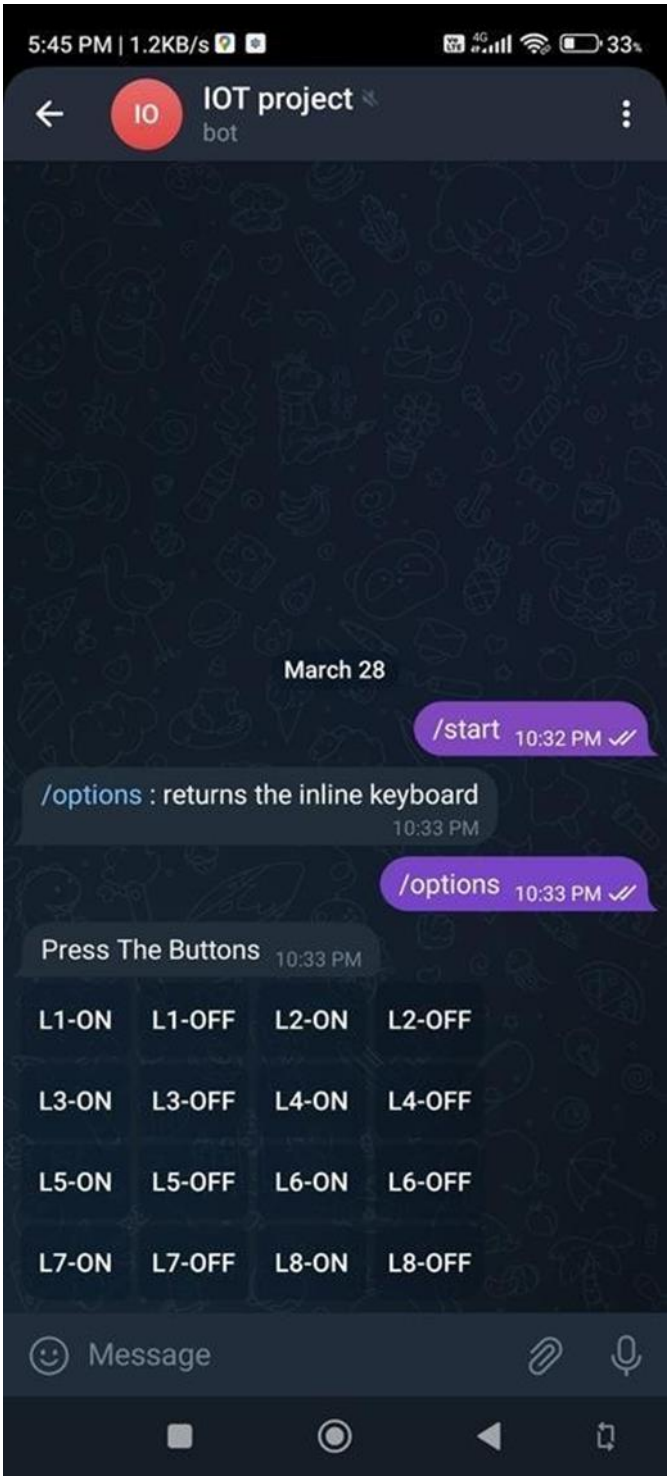
CODING

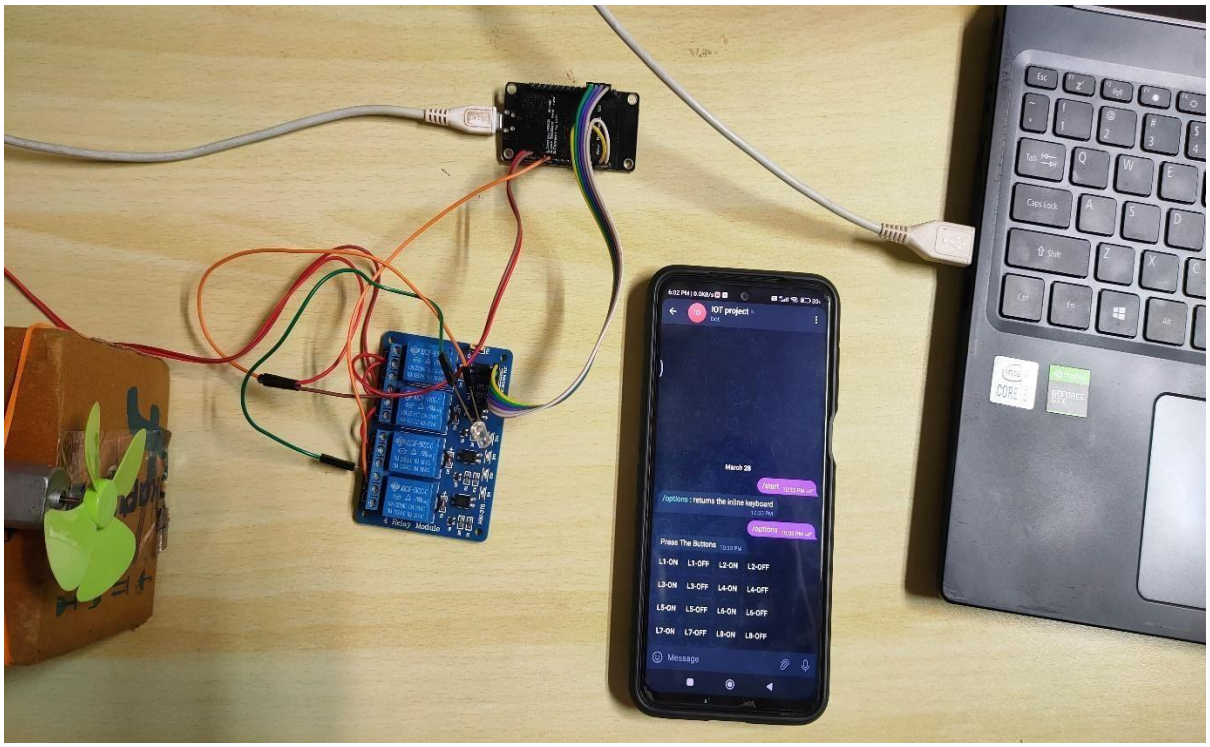
```
#include <ESP8266WiFi.h> // for ESP8266 Board
#include<WiFiClientSecure.h>
#include<UniversalTelegramBot.h> #include <ArduinoJson.h>
char ssid[] = "priyadharshini";           // network SSID (name) char password[] = "9025575833";
// network password

#define TELEGRAM_BOT_TOKEN
"5336543587:AAHyIWGEyjna9RwB7iv_QulC4TgzBJmMrhA" // YourTelegram Bot Token


WiFiClientSecure client;
UniversalTelegramBot bot(TELEGRAM_BOT_TOKEN, client);#define LED_PIN1 D1 //
Digital pins of Esp8266
#define LED_PIN2 D2
#define LED_PIN3 D3
#define LED_PIN4 D4
#define LED_PIN5 D5
#define LED_PIN6 D6
#define LED_PIN7 D7
#define LED_PIN8 D8

int delayBetweenChecks = 1000;
unsigned long lastTimeChecked;       //last time messages' scan has been done unsigned long
lightTimerExpires;
boolean lightTimerActive = false;


void setup()
{
Serial.begin(115200);
WiFi.mode(WIFI_STA);
WiFi.disconnect();
 delay(100);

pinMode(D1, OUTPUT);
digitalWrite(D1, HIGH);
 pinMode(D2, OUTPUT);
 digitalWrite(D2, HIGH);
 pinMode(D3, OUTPUT);
digitalWrite(D3, HIGH);
 pinMode(D4, OUTPUT);
 digitalWrite(D4, HIGH);
 pinMode(D5, OUTPUT);

digitalWrite(D5, HIGH);
 pinMode(D6, OUTPUT);
 digitalWrite(D6, HIGH);
 pinMode(D7, OUTPUT);
```

```
digitalWrite(D7, HIGH);
 pinMode(D8, OUTPUT);
 digitalWrite(D8, HIGH);

// attempt to connect to Wifi network:
 Serial.print("Connecting Wifi: ");
Serial.println(ssid);
WiFi.begin(ssid, password);


while (WiFi.status() != WL_CONNECTED) { Serial.print(".");
delay(500);
}


Serial.println("");
 Serial.println("WiFi connected");
Serial.print("IP address: ");
 Serial.println(WiFi.localIP());


client.setInsecure();
 bot.longPoll = 60;
}

void handleNewMessages(int numNewMessages)
{for (int i = 0; i < numNewMessages; i++)
{
if (bot.messages[i].type ==  F("callback_query"))
{
String text = bot.messages[i].text; Serial.print("Call back button pressed with text: ");
Serial.println(text);

if (text == F("ON1")) {digitalWrite(LED_PIN1, LOW);}
else if (text == F("OFF1")) { digitalWrite(LED_PIN1, HIGH);}if (text == F("ON2")) {
digitalWrite(LED_PIN2, LOW);}
else if (text == F("OFF2")) { digitalWrite(LED_PIN2, HIGH);}if (text == F("ON3"))
{digitalWrite(LED_PIN3, LOW);}
else if (text == F("OFF3")) { digitalWrite(LED_PIN3, HIGH);}if (text == F("ON4"))
{digitalWrite(LED_PIN4, LOW);}
else if (text == F("OFF4")) {digitalWrite(LED_PIN4, HIGH);}if (text == F("ON5"))
{digitalWrite(LED_PIN5, LOW);}
else if (text == F("OFF5")) { digitalWrite(LED_PIN5, HIGH);}if (text == F("ON6")) {
digitalWrite(LED_PIN6, LOW);}
else if (text == F("OFF6")) { digitalWrite(LED_PIN6, HIGH);}if (text == F("ON7"))
{digitalWrite(LED_PIN7, LOW);}
else if (text == F("OFF7")) { digitalWrite(LED_PIN7, HIGH);}if (text == F("ON8"))
{digitalWrite(LED_PIN8, LOW);}
else if (text == F("OFF8")) {digitalWrite(LED_PIN8, HIGH);}


} else {
```

```
String chat_id = String(bot.messages[i].chat_id);

String text = bot.messages[i].text;if (text == F("/options"))
{ String keyboardJson = F("[[{ \"text\" : \"L1-ON\", \"callback_data\" :
\"ON1\" },{ \"text\" : \"L1-OFF\", \"callback_data\" : \"OFF1\" },{ \"text\" :
\"L2-ON\", \"callback_data\" : \"ON2\" },{\"text\" : \"L2-OFF\",
\"callback_data\" : \"OFF2\"}],");
keyboardJson += F("[{ \"text\" : \"L3-ON\", \"callback_data\" : \"ON3\" },
{ \"text\" : \"L3-OFF\", \"callback_data\" : \"OFF3\" },{ \"text\" : \"L4-ON\",
\"callback_data\" : \"ON4\" },{ \"text\" : \"L4-OFF\", \"callback_data\" :
\"OFF4\" }],");
keyboardJson += F("[{ \"text\" : \"L5-ON\", \"callback_data\" : \"ON5\" },
{ \"text\" : \"L5-OFF\", \"callback_data\" : \"OFF5\" },{ \"text\" : \"L6-ON\",
\"callback_data\" : \"ON6\" },{ \"text\" : \"L6-OFF\", \"callback_data\" :
\"OFF6\" }],");
keyboardJson += F("[{ \"text\" : \"L7-ON\", \"callback_data\" : \"ON7\" },
{ \"text\" : \"L7-OFF\", \"callback_data\" : \"OFF7\" },{ \"text\" : \"L8-ON\",
\"callback_data\" : \"ON8\" },{ \"text\" : \"L8-OFF\", \"callback_data\" :
\"OFF8\" }]]");
bot.sendMessageWithInlineKeyboard(chat_id, "Press The Buttons", "", keyboardJson);
}
if (text == F("/start"))
{ bot.sendMessage(chat_id, "/options : returns the inline keyboard\n", "Markdown");}
}
}
}
void loop()

{ if (millis() > lastTimeChecked + delayBetweenChecks)
{ // getUpdates returns 1 if there is a new message from Telegram
int numNewMessages = bot.getUpdates(bot.last_message_received + 1);

if (numNewMessages) {
Serial.println("got response"); handleNewMessages(numNewMessages);
}
lastTimeChecked = millis();
if (lightTimerActive && millis() > lightTimerExpires)
 {
        lightTimerActive = false;
       digitalWrite(LED_PIN1, LOW);
       digitalWrite(LED_PIN2, LOW);
      digitalWrite(LED_PIN3, LOW);
       digitalWrite(LED_PIN4, LOW);
      digitalWrite(LED_PIN5, LOW);
       digitalWrite(LED_PIN6, LOW);
      digitalWrite(LED_PIN7, LOW);
       digitalWrite(LED_PIN8, LOW);


}
}
}
```

OUTPUT


5:45 PM | 1.2KB/s
33%
IO
IOT project
bot
March 28
/start 10:32 PM
/options : returns the inline keyboard
10:33 PM
/options 10:33 PM
Press The Buttons 10:33 PM
L1-ON
L1-OFF
L2-ON
L2-OFF
L3-ON
L3-OFF
L4-ON
L4-OFF
L5-ON
L5-OFF
L6-ON
L6-OFF
L7-ON
L7-OFF
L8-ON
L8-OFF
Message



CONCLUSION

With this project, we achieved the automation of devices and accessibility of devices in a place from anywhere in the world.