IoT Based Weather Adaptive Street Lighting System

Naan Mudhalvan Project Report

TEAM ID: NM2023TMID15296

Submitted by

HEMAMALINI.P	**(713120205003)**
IYAPPAN.G	**(713120104006)**
SANGEETHA.M	**(713120106303)**
SATHISHKUMAR.S	**(713120104010)**

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING AND TECHNOLOGY

In

INFORMATION TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

ELECTRONICS AND COMMUNICATION ENGINEERING

SRI SAI RANGANATHAN ENGINEERING COLLEGE

COIMBATORE - 641 109

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

TABLE OF CONTENTS

‘

CHAPTER 1 INTRODUCTION

1.1 PROJECT OVERVIEW:

The Internet of Things is an emerging topic of technical, social, and economic significance. Consumer products, durable goods, cars and trucks, industrial and utility components, sensors, and other everyday objects are being combined with Internet connectivity and powerful data analytic capabilities that promise to transform the way we work, live, and play.

Internet of Things aim towards making life simpler by automating every small task around us. As much is IoT helping in automating tasks, the benefits of IoT can also be extended for street lighting system.

The objective of an IoT-based weather adaptive street lighting system is to intelligently adjust the brightness of street lights based on real-time weather conditions. By leveraging IoT technology, the system aims to optimize energy usage, reduce costs, enhance safety, and minimize environmental impact.

It achieves these objectives by automatically adjusting lighting levels according to weather data, improving visibility during adverse conditions, saving energy during favorable weather, enabling remote monitoring and management, and contributing to sustainable and smarter cities.

Overall, the system aims to create efficient and safe urban environments while promoting energy efficiency and environmental sustainability.

1.2 PURPOSE:

The purpose of an IoT-based weather adaptive street lighting system is to optimize energy usage, enhance safety, promote environmental sustainability, and achieve cost saving. The aim is to provide adaptive lighting, enable remote monitoring and management, efficiently utilize resources, improve road safety, and ensure cost-effectiveness in street lighting operations.

- Energy Efficiency: Optimize energy usage by adjusting brightness levels and intelligently controlling streetlights. This reduces energy consumption, saving costs and promoting environmental sustainability.
- Safety Enhancement: Increase visibility and improve road safety by dynamically adapting lighting to weather conditions such as fog, rain, or snow. This reduces the risk of accidents and ensures safer conditions for drivers and pedestrians.
- Cost Reduction: Lower operational costs through reduced energy consumption and remote monitoring and control capabilities. Maintenance efforts are minimized, leading to financial savings for municipalities or organizations responsible for street lighting.
- Environmental Impact: Contribute to a reduced carbon footprint by minimizing energy waste and light pollution. The system ensures that lighting is provided only when and where needed, reducing environmental impact.
- Data-driven Insights: Collect and analyze data from sensors to gain valuable insights for optimizing energy consumption, predicting maintenance needs, and informing future urban planning decisions.

CHAPTER 2 IDEATION AND PROPOSED SOLUTION

2.1 PROBLEM STATEMENT DFINITION:

The Problem statement Comprises set of questions which the project seeks to address. It identifies the current state and future state and any gaps between the two.

The existing problem in IoT-based weather adaptive street lighting systems is the need for an efficient and reliable solution that optimizes energy usage, enhances road safety, and ensures accurate lighting adjustments based on real-time weather conditions. The system should provide accurate lighting adjustments, ensure standardized and seamless integration, offer reliable connectivity, incorporate regular maintenance and sensor calibration, implement robust privacy and security measures, and offer cost-effective solutions for wide-scale implementation.

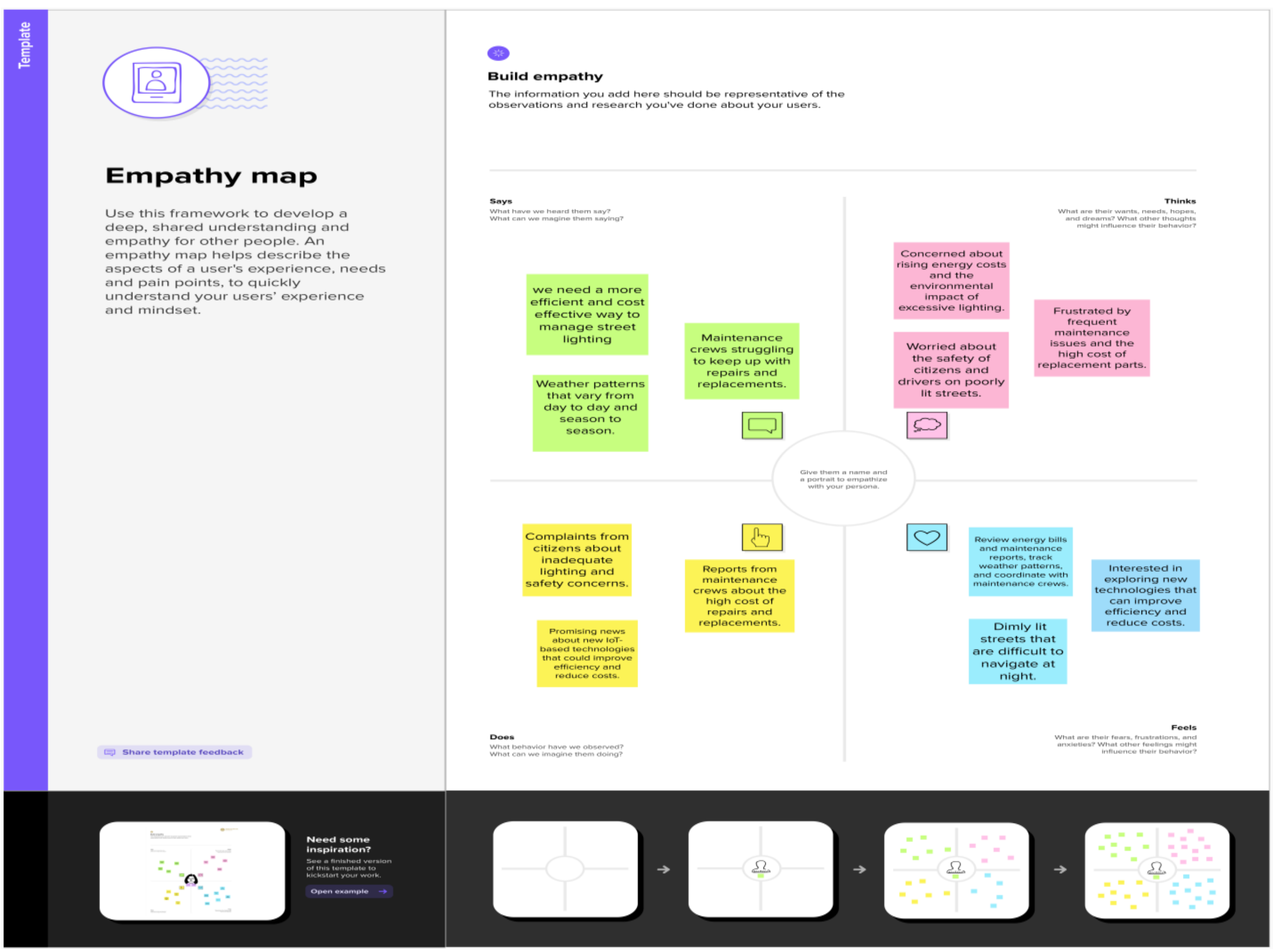
The Problem arises here in this project is:

PROBLEM 1:

PROBLEM 2:

2.2 EMPATHY MAP CANVAS:

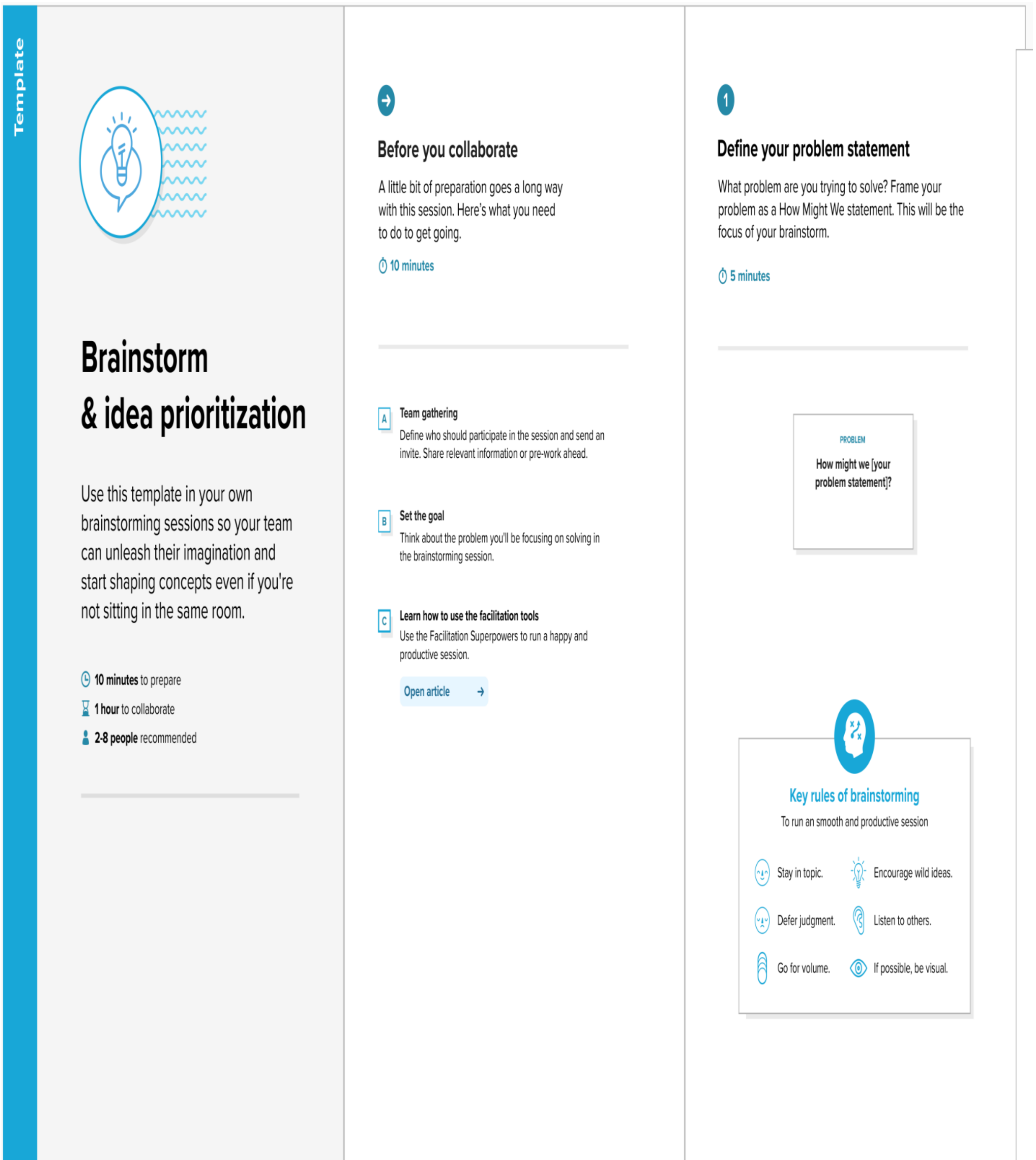
An empathy map is a visual tool that helps to understand and empathize with the feelings, needs, and experiences of a particular user or customer. It provides insights into their thoughts, behaviors, emotions, and goals, facilitating the development of more user-centered solutions.



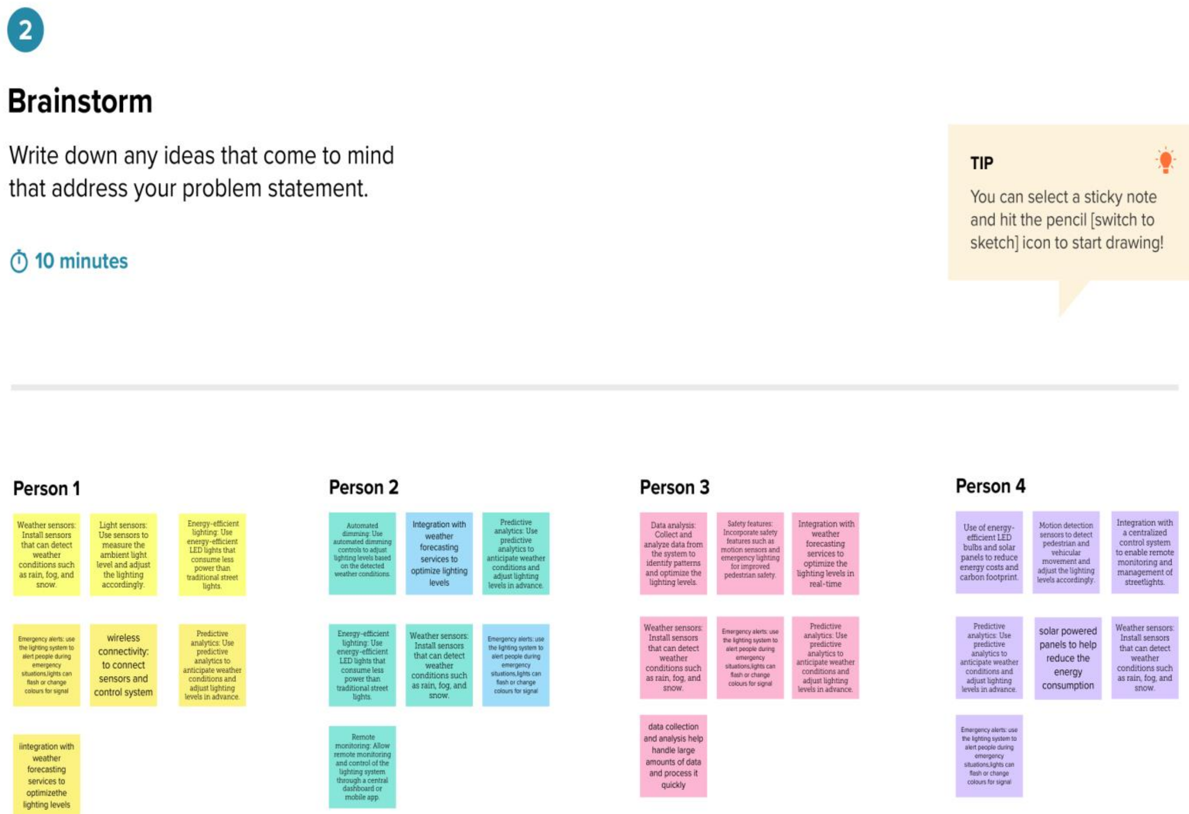
2.3 IDEATION AND BRAINSTORMING:

Brainstorm & Idea Prioritization:

Step-1: Team Gathering, Collaboration and Select the Problem Statement

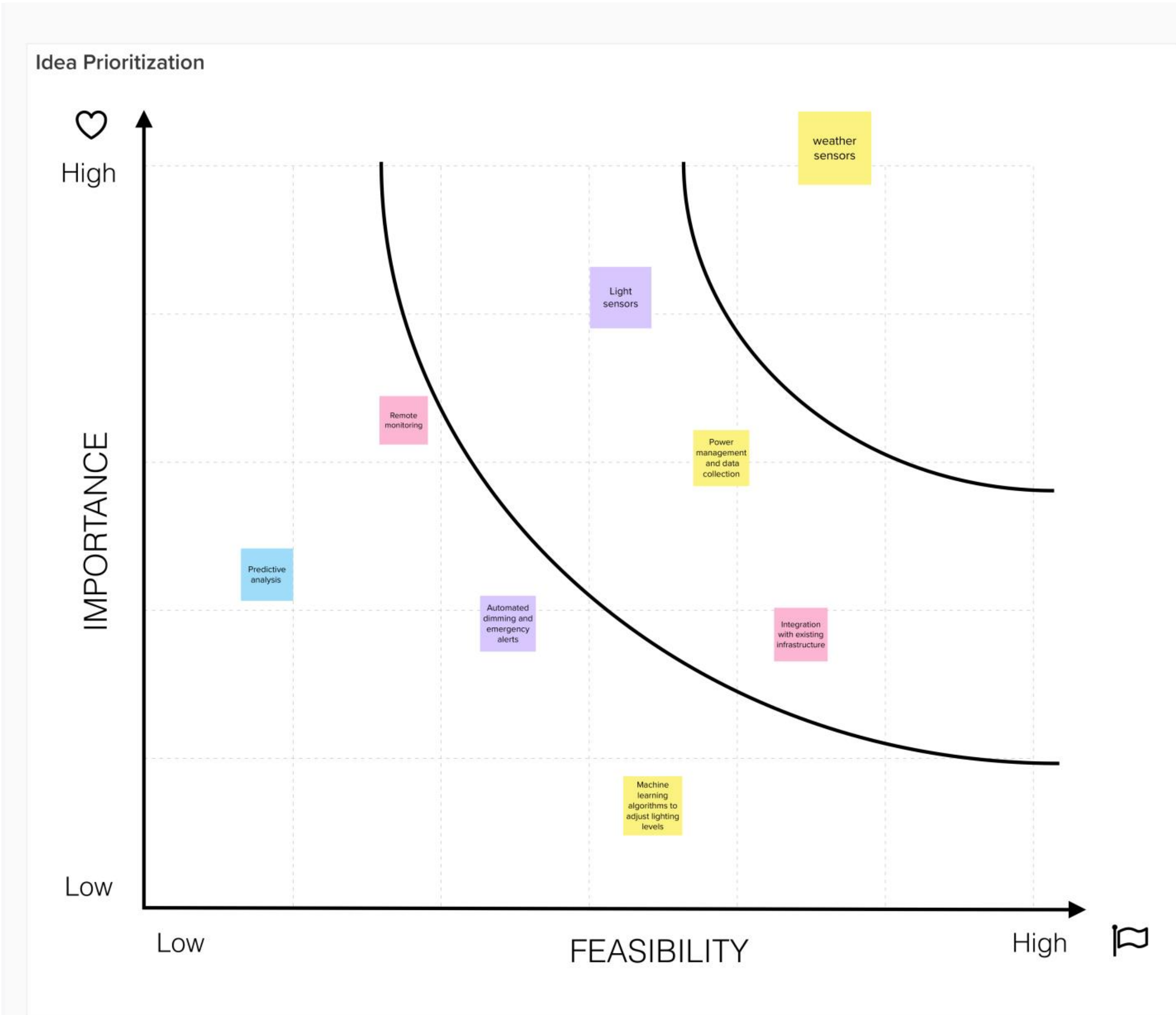


Step-2: Brainstorm, Idea Listing and Grouping



GROUP IDEAS:

Step-3: Idea Prioritization



2.4 PROPOSED SOLUTION:

The proposed solution should relate the current situation to a desired result and describe the benefits that is accurate when the desired result is achieved. So, begin your proposed solution by briefly describing this desired result. Our proposed solution involves integrating IoT technology with street lighting infrastructure to create a weather-adaptive system.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To reduce cost and consume energy by using sensors to detect the weather and produce light wherever it's highly needed.
2.	Idea / Solution description	Using smart techniques and sensors by utilizing advanced technology like sensor for monitoring the weather using IoT.
3.	Novelty / Uniqueness	Enabling the remote monitoring and control of street lights ,reducing the need for manual intervention and improving maintenance efficiency.
4.	Social Impact / Customer Satisfaction	The previous system lacked the ability to adapt to changing weather conditions, this will provide better sustainable environment.
5.	Business Model (Revenue Model)	A software that controls the lighting system and provides weather data analysis. Additionally the system can enhance the public safety by ensuring that streets as well-lit during inclement weather .As Productivity increases customer satisfaction also increases and hence need for implementation also increases which can raise the revenue of the business.
6.	Scalability of the Solution	It is definitely scalable we can also increase the constraints when problem arises.

CHAPTER 3 REQUIREMENT ANALYSIS:

3.1 FUNCTIONAL REQUIREMENTS:

The functional requirements of an IoT-based weather-adaptive street lighting system play a crucial role in ensuring its effectiveness and successful implementation. These requirements define the specific functionalities and capabilities that the system should possess to achieve its intended goals.

By addressing these requirements, the system can adapt its lighting operations based on real-time weather conditions, optimize energy usage, and enhance safety on the streets.

FR No.	Functional Requirement	Sub Requirement
FR-1	**User Registration**	Registration through Gmail
FR-2	**User Confirmation**	Confirmation via Email/ OTP
FR-3	**Login to system**	Check credentials Check the role of access
FR-4	**Manage Modules**	Manage system admins Manage role of user Manage user permission
FR-5	**Check Details**	Temperature details Weather condition details (Rain,Snow,Hot) Wind speed details
FR-6	**Log out**	Exit

3.2 NON FUNCTIONAL REQUIREMENTS:

In addition to the functional requirements, non-functional requirements play a crucial role in the successful implementation of an IoT-based weather-adaptive street lighting system.

These requirements address the qualities and constraints that the system should possess to ensure reliability, performance, security, and user satisfaction.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Fast response time,Easy to access Clears all the error messages Provides adequate help and support users including online services
NFR-2	Security	The system is designed to ensure the security of the data and devices in the system. The communication between the devices should be encrypted to prevent unauthorized access. The system can also be able to detect and prevent any attempts to hack or breach the system.
NFR-3	Reliability	This system is highly reliable and available at all times. The lighting system should not fail due to connectivity issues or hardware failures, and the software should be designed to handle unexpected situations such as power outages, network disruptions, and other failures.

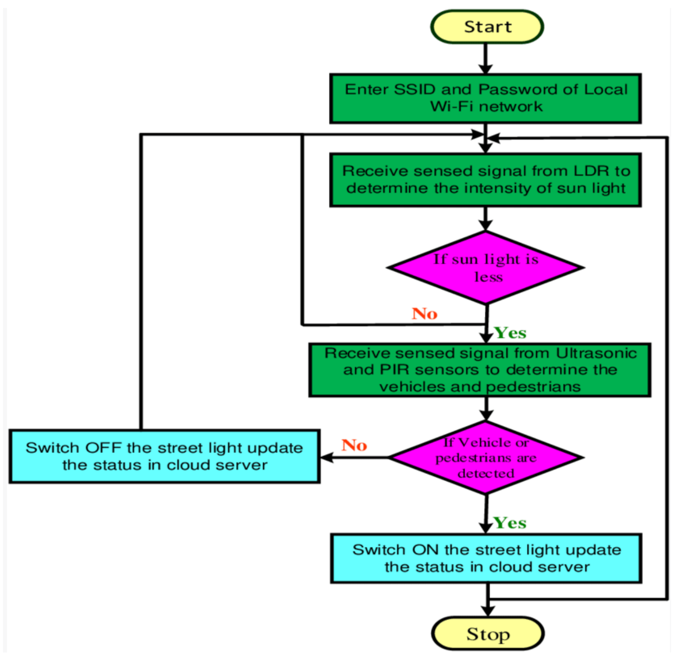
NFR-4	**Performance**	It will perform at a high level with minimal delay. This system is designed to process the data generated by the connected devices efficiently and quickly, to ensure the timely and accurate operation of the lighting system.
NFR-5	**Availability**	Available for 24/7, and it is designed to minimize any downtime. This system have mechanisms in place to detect and correct errors quickly to ensure that the lighting system remains operational at all times**.**
NFR-6	**Scalability**	It handles an increasing number of connected devices and be able to scale up or down based on the number of devices, the amount of data generated, and the processing requirements.

The functional requirements of an IoT-based weather adaptive street lighting system involve real-time weather monitoring, intelligent lighting control, remote management, energy optimization, and data analytics. The non-functional requirements include reliability, scalability, security, user-friendliness, and interoperability.

CHAPTER 4 PROJECT DESIGN

4.1 DATA FLOW DIAGRAM:

A data flow diagram provides a visual representation of how data is processed and flows within a system, helping to analyze and understand the data movement and relationships within a system or process.



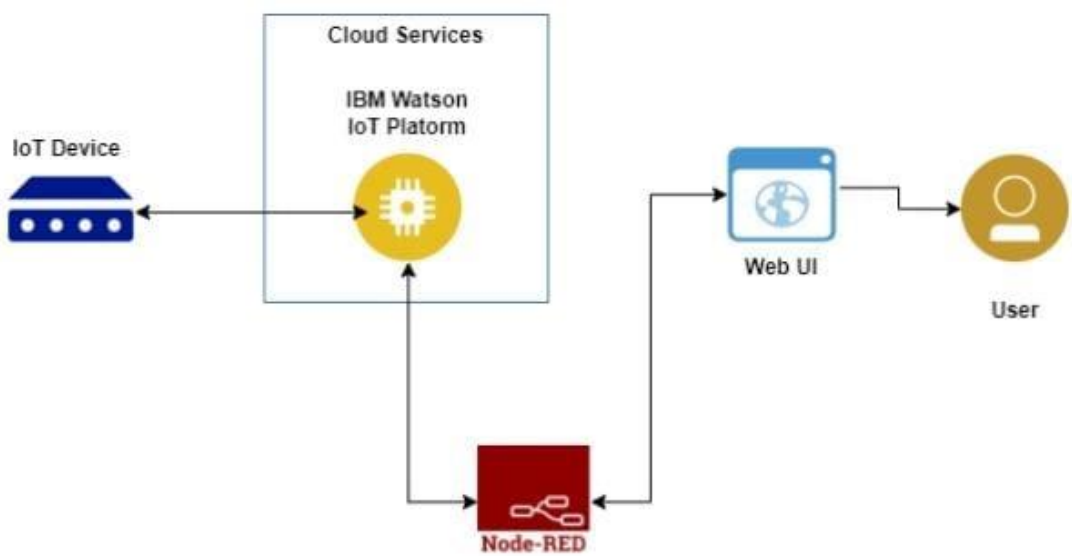
- Using various sensors, the various weather conditions, including temperature, moisture content, wind speed and humidity are measured. The results are then stored in the IBM cloud.

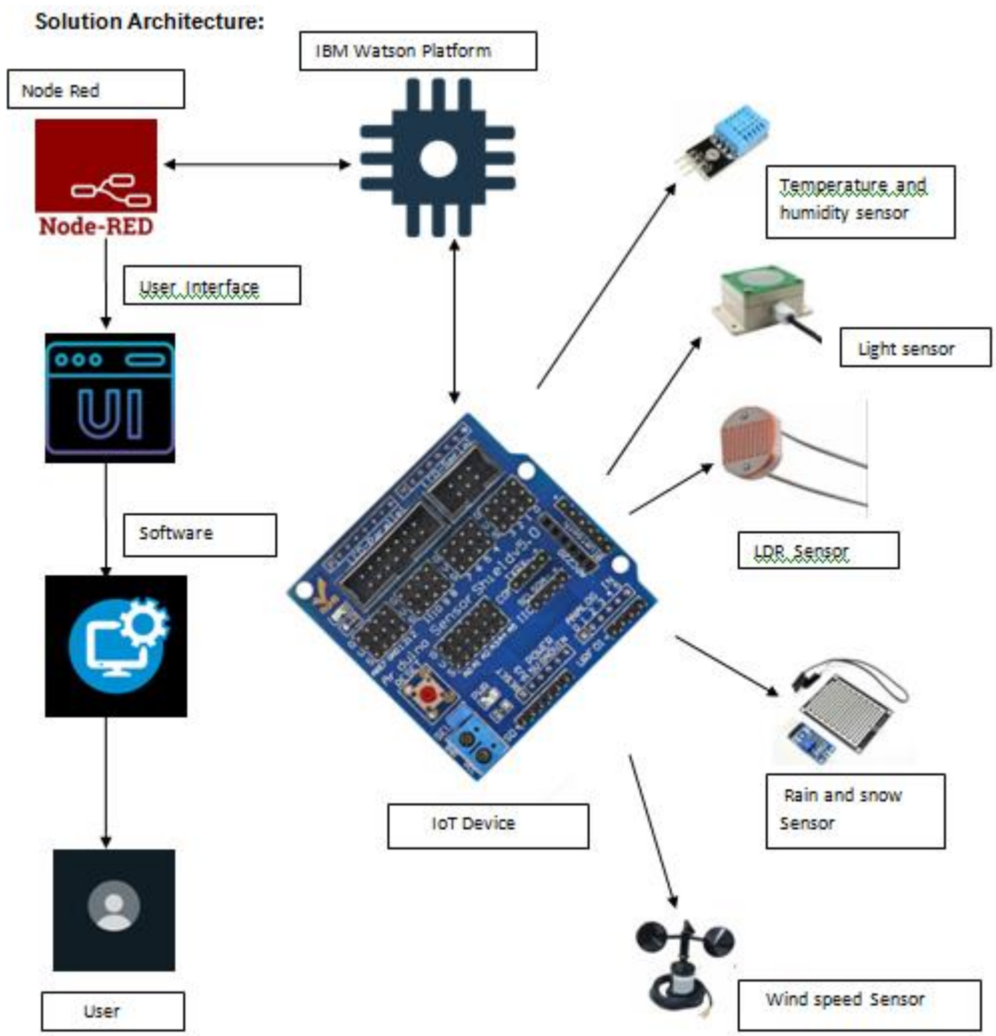
- The Arduino UNO is utilized as a processing unit to process the data from the sensors and weather API.

To write the hardware, software, and APIs. NODE-RED is employed as a programming tool. In order to communicate, the MQTT protocol is used.

4.2 SOLUTION AND TECHNICAL ARCHITECTURE:

The solution definition provides an overview of the proposed approach and its benefits, while the technical architecture definition details the technology infrastructure required for successful implementation.





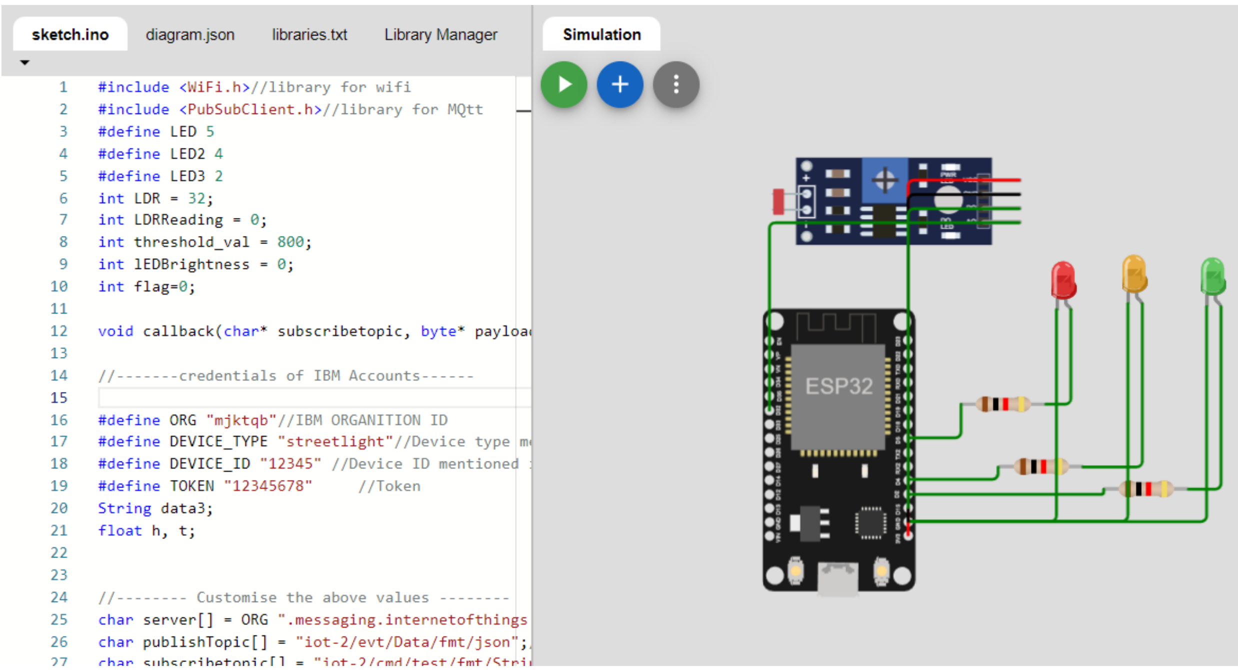
4.3 USER STORIES:

A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer. The purpose of a user story is to articulate how a piece of work will deliver a particular value back to the customer.

CHAPTER 5 CODING AND SOLUTIONING

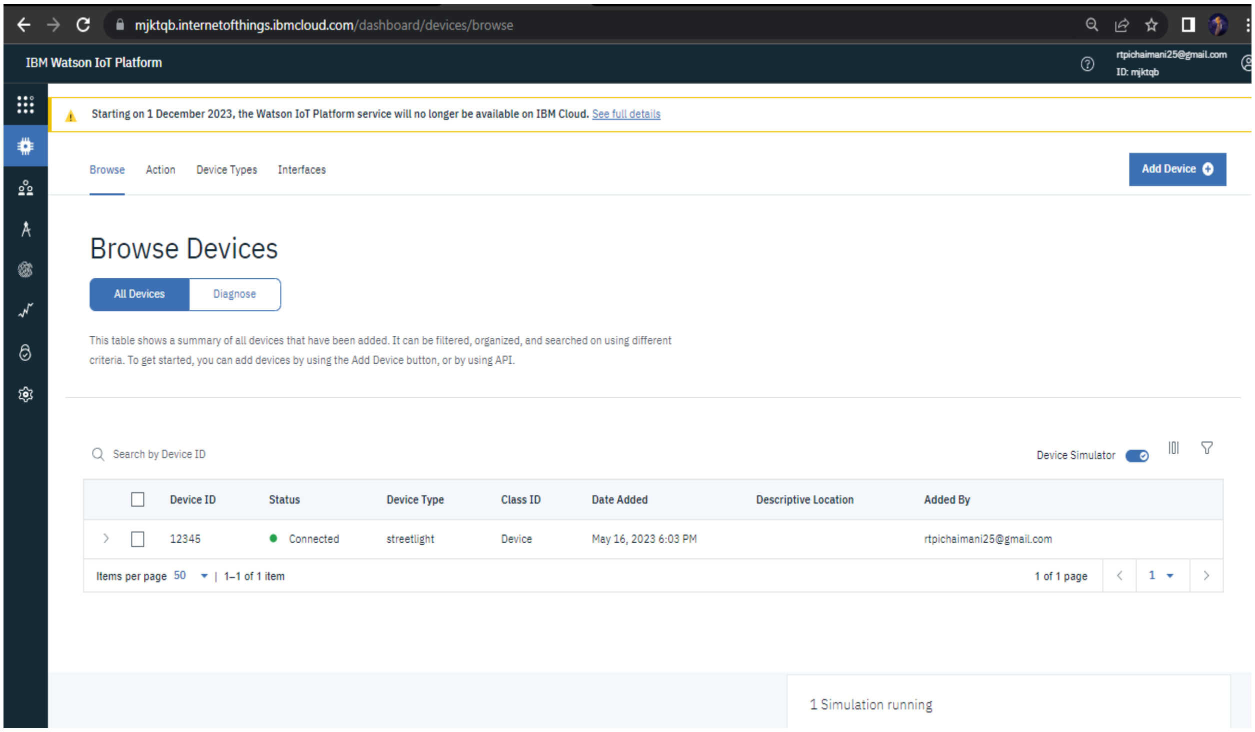
5.1 FEATURE-1:

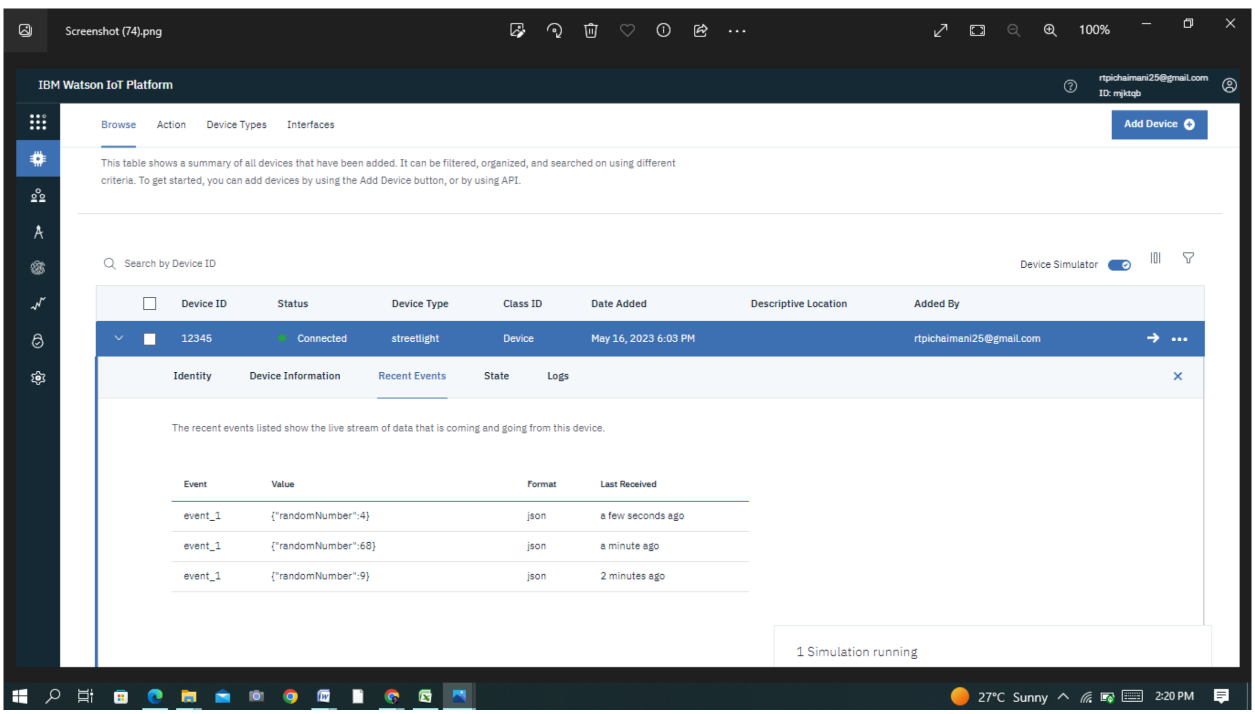
To adjust the lighting levels based on weather conditions, First connections are built using wokwi.com website using ESP32 , Photo resistor and LED.



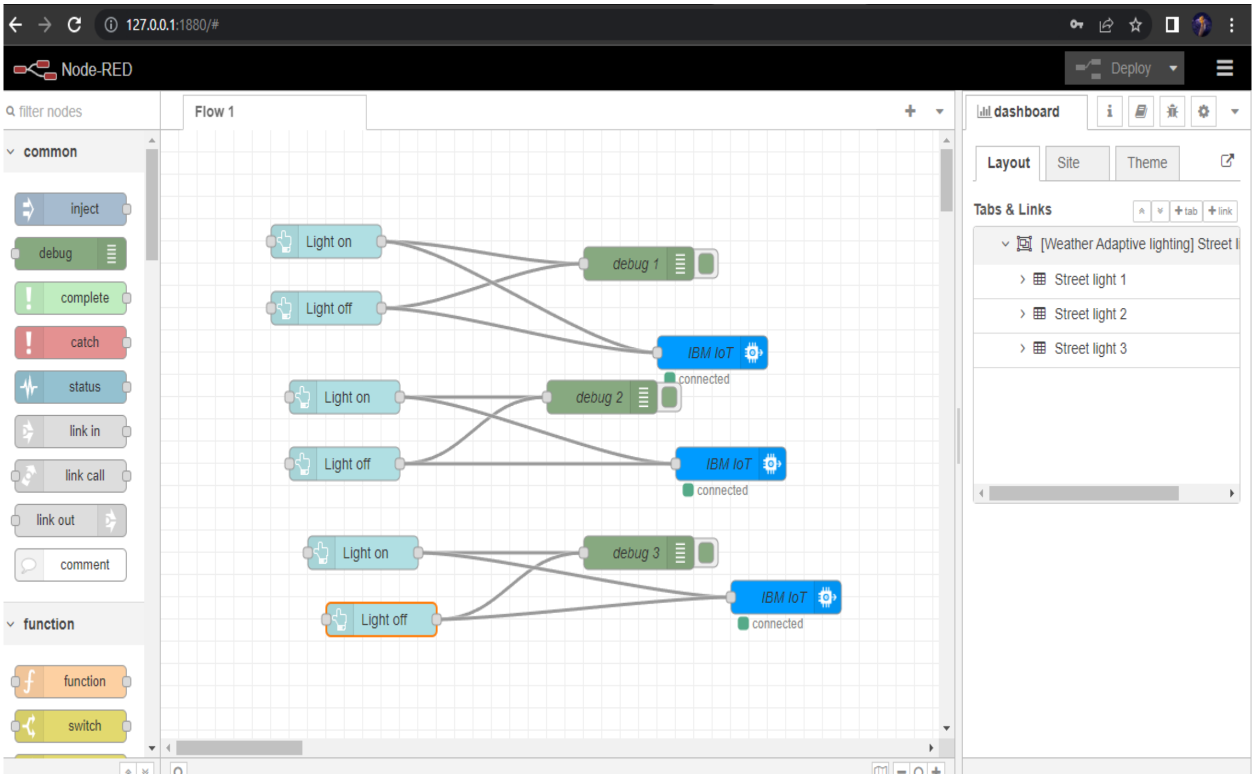
5.2 FEATURE-2:

After giving connections in wokwi, connect wokwi to IBMwatson Platform, Add device and create device type and device ID , copy those and place that in wokwi code then simulate running in IBMwatsonplatform using registered device.Start simulation. Make sure your device is connected to the ibm platform.

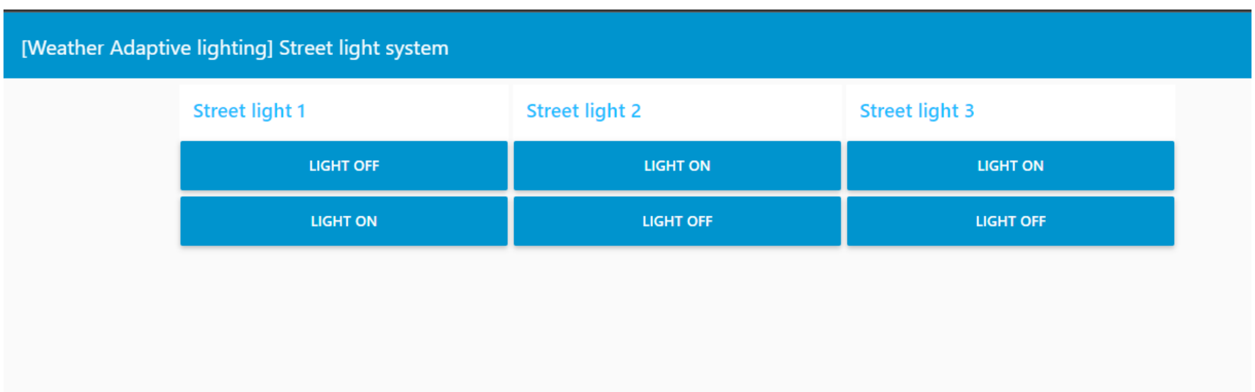

mjktqb.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
rtpichaimani25@gmail.com
ID: mjktqb
Starting on 1 December 2023, the Watson IoT Platform service will no longer be available on IBM Cloud. See full details
Browse
Action
Device Types
Interfaces
Add Device
Browse Devices
All Devices
Diagnose
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
Added By
12345
Connected
streetlight
Device
May 16, 2023 6:03 PM
rtpichaimani25@gmail.com
Items per page 50
1–1 of 1 item
1 of 1 page
1 Simulation running


Screenshot (74).png
100%
IBM Watson IoT Platform
rtpichaimani25@gmail.com
ID: mjktqb
Browse
Action
Device Types
Interfaces
Add Device
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
Added By
12345
Connected
streetlight
Device
May 16, 2023 6:03 PM
rtpichaimani25@gmail.com
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
event_1
{"randomNumber":4}
json
a few seconds ago
event_1
{"randomNumber":68}
json
a minute ago
event_1
{"randomNumber":9}
json
2 minutes ago
1 Simulation running
27°C Sunny
2:20 PM

After this, connect IBMwatsonplatform to Node red which would help to adjust the lighting levels based on weather conditions.

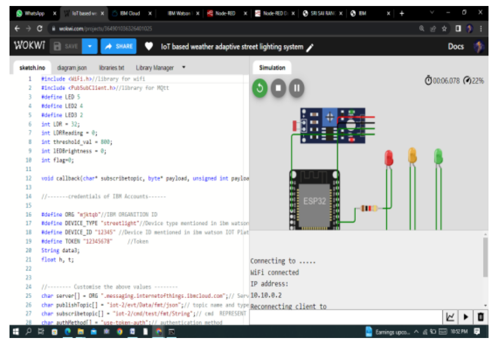
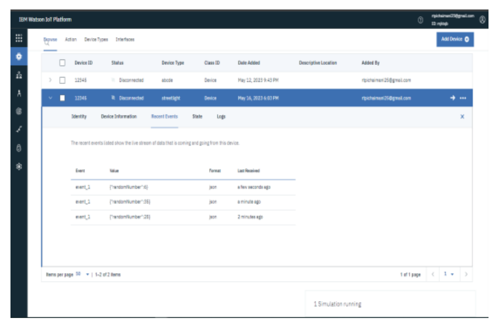


INTERFACE:



CHAPTER 6 RESULTS

6.1 PERFORMANCE METRICS:

Parameter	Values	Screenshot
Threshold LDR LDRReading Led brightness	800 32 0 0	 

			NFT - Risk Assessment						
S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	ad/Volume Chang	Risk Score	Justification
1	Temperature	New	Low	No Changes	Moderate	Moderate	No Changes	RED	As we have seen the changes
2	Wind	New	Low	No Changes	Moderate	Moderate	No changes	YELLOW	As we have seen the changes
3	Light	New	Low	No Changes	Moderate	Moderate	No Changes	GREEN	As we have seen the changes

NFT - Detailed Test Plan				
S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff
1	Temperature	Using wokwi and node red	Dependancy cloud client / Risk -Moderate	
2	Wind	Using wokwi and node red	Dependancy cloud client / Risk -Moderate	

S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects	Approvals/SignOff
1	Temperature	Using wokwi ad node	no	Expectation partially met	No-Go	Observed intermittent performance issue.Bug is open		
2	Wind	Using wokwi ad node	yes	Expectations met	Go	Observed response for weather		
3	Light	Using wokwi ad node	yes	Expectations met	Go	Observes reponse and adjusted		

CHAPTER 7 ADVANTAGES & DISADVANTAGES

7.1 ADVANTAGES:

- Energy efficiency
- Enhanced safety
- Cost reduction
- Environment sustainability
- Intelligent control and analytics
- Integration with smart city infrastructure
- Improved user experience
- Customizable lighting profiles
- Remote monitoring and maintenance

7.2 DISADVANTAGES:

- Initial Investment
- Technical complexity
- Connectivity and Reliability
- Privacy and security issues
- Compatibility issues
- Limited Accuracy
- Dependency on Weather data
- Reliance on internet connectivity
- Legal and Regulatory considerations

CHAPTER 8 CONCLUSION

Thus the objective of the project to implement an IoT system in order to help pedestrians and people by providing proper lighting based on weather conditions. They provide adaptive lighting that adjusts to changing weather conditions, resulting in cost savings, reduced environmental impact, and enhanced visibility for drivers and pedestrians. By leveraging the capabilities of IoT technology, weather adaptive street lighting systems can contribute to the creation of smarter, more sustainable, and safer cities.

CHAPTER 9 FUTURE SCOPE

IoT based weather adaptive street lighting system have the potential for several enhancements that can further improve their functionality and impact. The future advancements discussed in this report pave the way for the realization of these benefits, ultimately improving the quality of life for residents and fostering more efficient and environmentally conscious urban environments.

The integration with smart city initiatives, advancements in analytics and maintenance, focus on sustainability, dynamic lighting control, integration with autonomous vehicles, and data-driven urban planning are key areas that will shape the future development and implementation of these systems, leading to smarter, safer, and more sustainable cities.

CHAPTER 10 APPENDIX

WOKWI CODE:

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#define LED 5
#define LED2 4
#define LED3 2
int LDR = 32;
int LDRReading = 0;
int threshold_val = 800;
int lEDBrightness = 0;
int flag=0;

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "mjktqb"//IBM ORGANITION ID
#define DEVICE_TYPE "streetlight"//Device type mentioned in ibm watson IOT
Platform
#define DEVICE_ID "12345" //Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event
perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/test/fmt/String";// cmd  REPRESENT command
type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
```

```
//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined
client id by passing parameter like server id,portand wificredential
void setup()// configureing the ESP32
{
  Serial.begin(115200);

  pinMode(LED,OUTPUT);
  pinMode(LED2,OUTPUT);
  pinMode(LED3,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{
  //PublishData(t, h);
  //delay(1000);

  /* LDRReading = analogRead(LDR);
  Serial.print("LDR READING:");
  Serial.println(LDRReading);

  if (LDRReading >threshold_val){
  lEDBrightness = map(LDRReading, 0, 1023, 0, 255);
  Serial.print("LED BRIGHTNESS:");
  Serial.println(lEDBrightness);

  analogWrite(LED, lEDBrightness);
  analogWrite(LED2, lEDBrightness);
  analogWrite(LED3, lEDBrightness);
  }
  else{
  analogWrite(LED, 0);
  analogWrite(LED2, 0);
  analogWrite(LED3, 0);
```

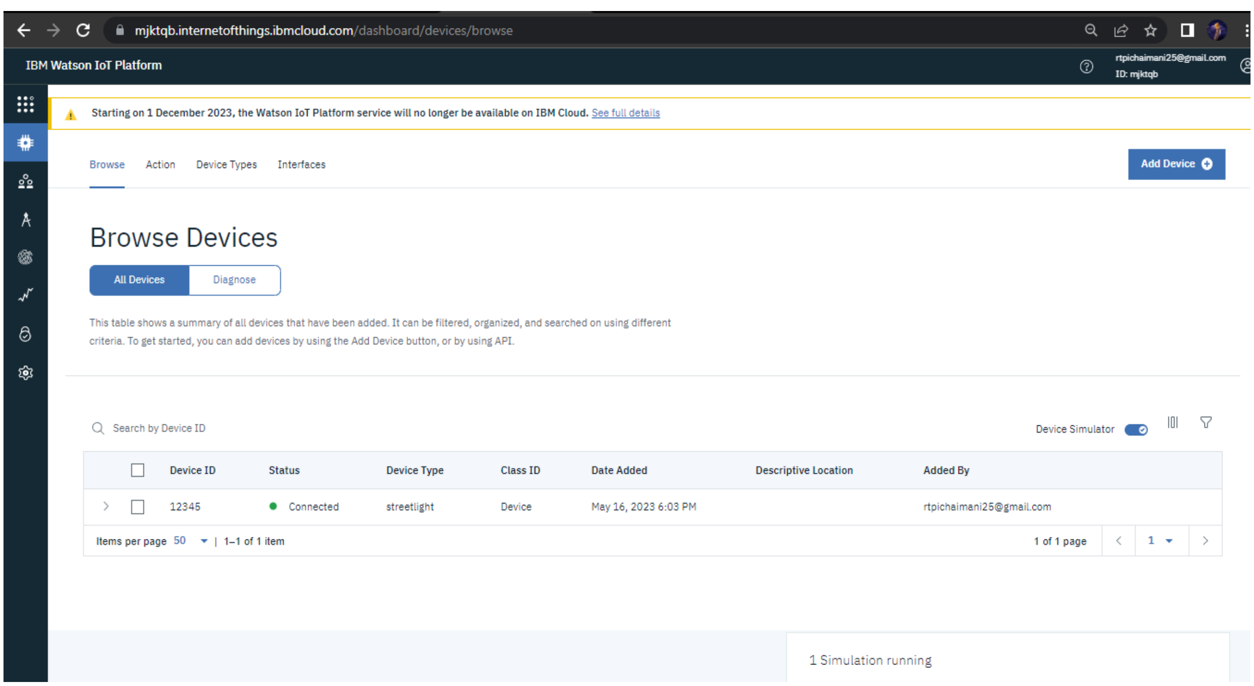
```
  }

  delay(300);*/

  if (!client.loop()) {
    mqttconnect();
  }
}



/*....................................retrieving to Cloud...............................*/

/*void PublishData(float temp, float humid) {
  mqttconnect();//function call for connecting to ibm*/
  /*
     creating the String in in form JSon to update the data to ibm cloud
  */
  /*String payload = "{\"temperature\":";
  payload += temp;
  payload += "," "\"humidity\":";
  payload += humid;
  payload += "}";


  Serial.print("Sending payload: ");
  Serial.println(payload);


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it
will print publish ok in Serial monitor or else it will print publish failed
  } else {
    Serial.println("Publish failed");
  }

}  */
void mqttconnect() {
  if (!client.connected()) {
```

```
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!!!client.connect(clientId, authMethod, token)) {
     Serial.print(".");
     delay(500);
    }

     initManagedDevice();
     Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the
connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
```

```
  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);

  for (int i = 0; i < payloadLength; i++) {
    //Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }

  Serial.println("data: "+ data3);
  if(data3=="lighton1")
  {
Serial.println(data3);
digitalWrite(LED,HIGH);

  }

  else if(data3=="lightoff1")
  {
Serial.println(data3);
digitalWrite(LED,LOW);


  }
  else if(data3=="lighton2")
  {
Serial.println(data3);
digitalWrite(LED2,HIGH);

  }

  else if(data3=="lightoff2")
  {
Serial.println(data3);
digitalWrite(LED2,LOW);

  }
  else if(data3=="lighton3")
```

```
  {
Serial.println(data3);
digitalWrite(LED3,HIGH);

  }

  else if(data3=="lightoff3")
  {
Serial.println(data3);
digitalWrite(LED3,LOW);

  }
data3="";



}
```

OUTPUT:



Github link: https://github.com/naanmudhalvan-SI/PBL-NT-GP--6246-1680874189

Project demo link: https://drive.google.com/file/d/1GsIrn15wCaRT77M-aOW4J6fVQtWSd-Ij/view?usp=drivesdk