

Phase 4 Development part 2

ENVIRONMENT MONITORING IN PARKS

- In the phase 3, we initiated to develop our idea of the problem statement and it ended up with the technologies to capture the real time environmental data from the parks using IOT devices.
- In this phase, we are going to develop the requirements for data transmission to the cloud services and the exposure of the data to the public.
- For data transmission, we are going to use an endpoint and the data transmission is done by Wi-Fi on IoT and the transmission protocol used here is HTTP.

Technologies Used:-

For our project, we used technologies like,

- **Data Transmission:**

For data transmission, we ended up with Wi-Fi technology. We know that is a high power consuming implementation but for high coverage distances, Wi Fi is the best choice.

- **Communication Protocol:**

We chose **HTTP** protocol for communication to the server because of its simplicity of coding and it is secure for data transmission.

- **Data Storage:**

We decided to use the endpoint for the data storage from the IoT devices that can be implemented in the parks. The name of the endpoint domain is **Beeceptor**. It provides the user to create mock API's for testing. We can change the data storage idea for future implementation. Our beeceptor endpoint is <https://smartenvironment.free.beeceptor.com>

- **Idea of Data Sharing:**

The data monitored and received from the IoT project kit(i.e Temperature and Humidity) can be exposed to

the people by integrating the created API to a website that will be hosted on the network for information sharing.

Features of the website:

The idea of website is chosen because of its simplicity and people can use it on from any client server(i.e Browser). Our designed website has features like as follows:

- ❖ It has the Suggestion bar for giving suggestion to the people about their planning of their outdoor activities.
- ❖ It has the Refresh button for receiving the data from the endpoint at instantaneous time when the people wants to know the weather condition.
- ❖ Other than that, we can add up some details in future like the name of the park, timing, etc.,

Implementation of Wi-Fi in IoT devices:

In wokwi platform, we have to develop the code for simulating our device that works on the wi-fi for the data transmission. We send the data to our beceptor endpoint so that the code must be in the form of both using wi-fi and the http protocol. Here is the code:

```
#include <Arduino.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include <DHT.h>

const char* ssid = "Wokwi-GUEST";
const char* password = "";
const char* beceptorEndpoint = "https://smartenvironment.free.beeceptor.com";
// endpoint url
#define DHTPIN 4
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(5000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");
```

```

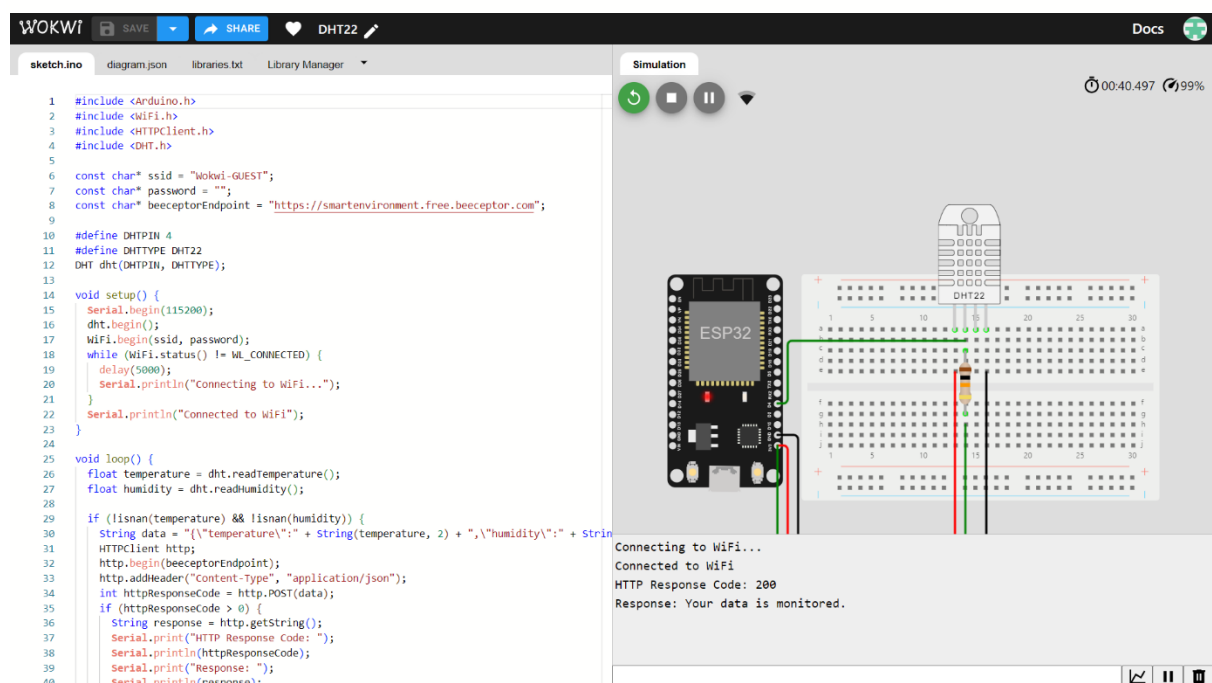
}

void loop() {
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

  if (!isnan(temperature) && !isnan(humidity)) {
    String data = "{\"temperature\": " + String(temperature, 2) +
    ", \"humidity\": " + String(humidity, 2) + "}";
    HTTPClient http;
    http.begin(beeceptorEndpoint);
    http.addHeader("Content-Type", "application/json");
    int httpResponseCode = http.POST(data);
    if (httpResponseCode > 0) {
      String response = http.getString();
      Serial.print("HTTP Response Code: ");
      Serial.println(httpResponseCode);
      Serial.print("Response: ");
      Serial.println(response);
    }
    http.end();
  }

  delay(60000);
}

```

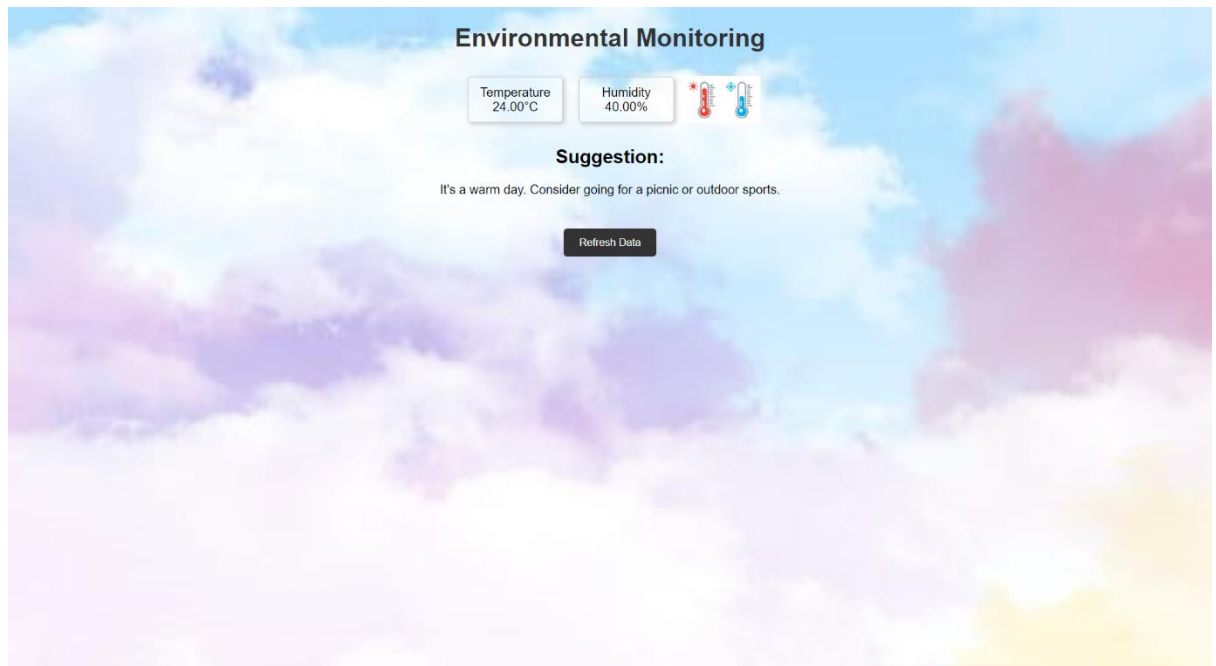


This is the simulated result image attached above. The data has to be sent to the endpoint and the from the endpoint the message “Your data is monitored” is received successfully.

Website Implementation and creation:

Our website is created by the programs based on coding languages like html for index, css for website design and java script for website information. The code has to be tested with programming tool. We used Visual Studio Code for Testing purposes. We have to configure the endpoint mock rules posting the data to the website we created.

The outlook of the website is attached below:



This is the user interface of the website that we created for exposing the temperature and humidity values along with the suggestion for the people to planning their outdoor activities effectively.

Website Hosting:

Our website is hosted on a free domain named Netlify since it is simple to host static websites like ours and its completely no cost for hosting.

Our website link hosted on Netlify platform is

<https://environmentalmonitoringibm.netlify.app/>

All the information about the website and the mocking rules of the endpoint on beeceptor will be explained in the documentation of phase 5.