

## VISUALISASI DATA SUHU DAN KELEMBABAN RUANGAN ADMISI UTDI BERBASIS IOT MENGGUNAKAN GOOGLE SPREADSHEET DAN GOOGLE LOOKER STUDIO

*Kuindra Iriyanta<sup>1</sup>, Hendra<sup>2</sup>, Indra Yatini Buryadi<sup>3</sup>*

<sup>1,3</sup>Universitas Teknologi Digital Indonesia

Jl. Raya Janti (Majapahit) No. 143, Karangjambe, Yogyakarta

<sup>2</sup>Institut Teknologi dan Bisnis Adias

Jl. Tegalmilati No. 22, Petarukan, Pemalang, Jawa Tengah

*Email : <sup>1</sup>kuindra@gmail.com, <sup>2</sup>camp.hendra@gmail.com, <sup>3</sup>indrayatini@utdi.ac.id*

### ABSTRAK

Visualisasi data adalah salah satu cara untuk menyajikan informasi atau laporan agar menarik dan mudah dipahami. Sehingga dengan melihat visualisasi tersebut dapat diperoleh informasi yang dibutuhkan dan kemudian diambil sebuah kesimpulan. Dashboard adalah sebuah tampilan visual dari informasi atau laporan yang informatif dan juga mudah untuk dipahami. Pemanfaatan teknologi saat ini mendorong setiap lembaga atau instansi menggunakan dashboard untuk menyampaikan laporan. Salah satu dashboard yang sering dipakai adalah Google Looker Studio, yang dapat dimanfaatkan untuk memvisualkan data suhu dan kelembaban ruangan kerja yang terukur dan tersajikan secara realtime. Data suhu diambil langsung dari sensor yang dipasang di ruangan kerja. Dengan Google Looker Studio yang berfungsi sebagai media visualisasi data dan terkoneksi dengan Google Spreadsheet sebagai pengolah data dan IFTTT sebagai pengambil data dari sensor, maka visualisasi data akan dapat disajikan dengan jelas, tepat dan akurat.

**Keywords:** *Google Looker Studio, Spreadsheet, IFTTT, NodeMCU ESP32, IoT.*

### ABSTRACT

*Data visualization is one way to present information or reports so that they are interesting and easy to understand. So by looking at the visualization, the information needed can be obtained and then a conclusion can be drawn. Dashboard is a visual display of information or reports that are informative and easy to understand. The current use of technology encourages every institution or agency to use dashboards to submit reports. One of the dashboards that is often used is Google Looker Studio, which can be used to visualize the measured and presented temperature and humidity data in the workspace in real time. Temperature data is taken directly from sensors installed in the workspace. With Google Looker Studio which functions as a data visualization medium and connected to Google Sheets as a data processor and IFTTT as a data collector from sensors, data visualization will be able to be presented clearly, precisely and accurately.*

**Keywords:** *Google Looker Studio, Spreadsheet, IFTTT, NodeMCU ESP32, IoT.*

## 1. PENDAHULUAN

Dashboard adalah cara penyajian data dan atau informasi penting menggunakan grafik, tabel, gambar dan lainnya sehingga menarik dan mudah dipahami oleh semua pihak. Dengan adanya dashboard yang memvisualkan data apapun itu, maka ketertarikan orang untuk melihat, kemudian memahami akan meningkat, sehingga data yang telah divisualkan akan lebih informatif. Visualisasi atau Visualisasi data adalah istilah umum yang menggambarkan setiap upaya untuk membantu orang memahami signifikansi data dengan menempatkan data dalam konteks visual [1]. Di era teknologi modern, dimana setiap lini kehidupan telah mengadopsi internet, salah satunya adalah keberadaan internet of things, penggunaan dashboard ini tidak dapat dipungkiri menjadi sebuah solusi.

Penelitian ini akan merancang dan merealisasikan sistem IoT yang terhubung ke Google Spreadsheet menggunakan IFTTT dengan komponen utama ESP32 lalu Dashboard Google Looker Studio digunakan

untuk visualisasi data suhu dan kelembaban. Google Spreadsheet digunakan untuk menyimpan data historis suhu dan kelembaban dimana Google Spreadsheet memiliki kemampuan yang cukup baik dalam menyimpan data dan memiliki kemampuan untuk diakses dari luar sehingga tidak diperlukan program database yang khusus untuk menyimpan data historis seperti InfluxDB yang juga memiliki kemampuan yang sama yaitu menyimpan data historis [2]. Dashboard Google Looker Studio yang memiliki fungsi utama sebagai alat interpretasi visual yang ramah pengguna digunakan untuk merepresentasikan kumpulan data yang kompleks [3]. Keunggulan yang dimiliki oleh Google Looker Studio ialah kemampuan untuk kolaborasi dalam membuat visualisasi data serta banyaknya fungsi yang ditawarkan secara gratis jika dibandingkan dengan pesaingnya. Google Looker Studio lebih praktis digunakan jika dibandingkan aplikasi Tableau, tetapi masih kurang dalam hal mengkostumisasi dashboard [4] [5]. Dengan Dashboard Google Looker Studio, informasi tentang suhu dan kelembaban ruangan dapat diakses secara luas dengan menggunakan browser yang tersedia pada perangkat smartphone maupun menggunakan komputer.

## 2. TINJAUAN PUSTAKA

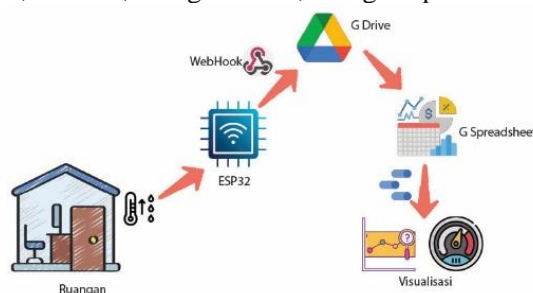
Penggunaan internet of things untuk melakukan monitoring lingkungan semakin hari semakin banyak diimplementasikan, misalnya monitoring suhu dan kelembaban ruangan dimana Heru Purnadi dalam penelitiannya, melakukan visualisasi suhu dan kelembaban ruangan menggunakan Google Looker Studio. Monitoring suhu dan kelembaban dilakukan namun proses pencatatannya masih dilakukan manual dengan cara mengetikkan pada Google Spreadsheet setiap harinya. Ini tentunya menjadi masalah ketika terjadi keabsenan petugas dalam memasukkan data [6]. Sedangkan Lidia Hana Sinaga dalam penelitiannya telah membangun sistem monitoring suhu dan kelembaban udara dimana visualisasinya menggunakan web dan menggunakan NodeMCU sebagai mikrokontroler [7]. Lebih jauh tentang penggunaan IoT adalah adanya proses yang dilakukan bukan hanya menampilkan data, tetapi juga melakukan prediksi terhadap kemungkinan data yang akan muncul pada periode selanjutnya menggunakan metode regresi linear sehingga dapat diambil keputusan terhadap prediksi data yang didapatkan [8].

## 3. METODE PENELITIAN

Metode penelitian meliputi observasi, kemudian implementasi. Observasi dilakukan dengan mengamati ruangan Humas dan Admisi Universitas Teknologi Digital Indonesia (UTDI) untuk dapat menentukan peletakkan sensor. Implementasi dilakukan dengan membangun sistem monitoring dan kemudian hasilnya akan divisualisasikan. Waktu penelitian dilakukan sejak 25 Maret 2022 – 29 Maret 2022.

## 4. PEMBAHASAN

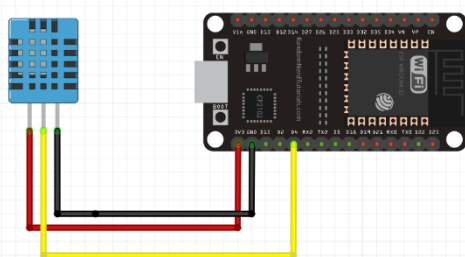
Gambar 1 merupakan gambaran umum dari sistem yang akan dikembangkan yang melibatkan 5 komponen utama yaitu ruangan, ESP 32, Google Drive, Google Spreadsheet, dan visualisasi.



Gambar 1. Rancangan Sistem

#### 4.1. Merancang Perangkat Keras

Data yang dibutuhkan dalam penelitian ini adalah data yang diperoleh dari sensor yang terpasang di lokasi penelitian. Sensor yang dipakai untuk mendapatkan data suhu dan kelembaban ruangan adalah sensor DHT11. DHT11 merupakan sebuah sensor yang mampu mengukur dua parameter sekaligus, yakni suhu udara (temperature) dan kelembaban udara (humidity). Didalam sensor ini terdapat sebuah thermistor tipe NTC (Negative Temperature Coefficient) untuk mengukur suhu, sebuah sensor kelembaban tipe resisitif dan sebuah mikrokontroler 8-bit yang mengolah kedua sensor tersebut dan mengirim hasilnya ke pin output dengan format single-wire bi-directional (kabel tunggal dua arah). Sensor ini memiliki 3 kaki yaitu VCC, GND dan data. Kaki kaki pada sensor DHT11 ini dihubungkan dengan ESP32 sebagai prosesor yang akan membaca nilai sensor. ESP32 adalah sebuah mikrokontroler yang dibuat oleh sebuah perusahaan bernama Espressif Systems. Kelebihan yang dimiliki oleh mikrokontroler ini adalah sudah terdapat Wi-Fi dan Bluetooth di dalamnya, sehingga sangat memudahkan digunakan untuk membangun sebuah sistem berbasis IoT. Kaki-kaki ESP32 memiliki pin masukan yang dapat dimanfaatkan untuk membaca data dari sensor DHT11. Hubungan sensor DHT11 ke ESP32 seperti terlihat pada gambar 2, dimana jalur data yang digunakan adalah pin D4 sebagai jalur pengiriman data dari sensor ke DHT11.



Gambar 2. Pengkabelan antara DHT11 dan ESP32

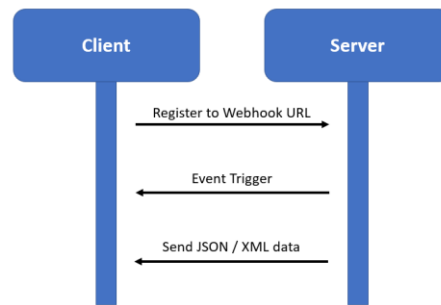
#### 4.2. IFTTT

IFTTT (If This, Than That) adalah sebuah layanan otomasi yang menggabungkan beberapa layanan internet menjadi satu. IFTTT memungkinkan untuk men-share sebuah post pada facebook juga ter-post di Instagram dan twitter. Ketika ada panggilan masuk di Android, nomor tersebut bisa tersimpan otomatis di Google Spreadsheet. IFTTT lebih menyerupai dengan suatu mekanisme otomasi. Hanya dengan 1 aksi akan didapatkan berbagai reaksi. Sehingga dengan menggunakan IFTTT kita dapat menghubungkan antara sensor dengan Google Spreadsheet [9].

#### 4.3. Koneksi Google Spreadsheet dengan IFTTT

Google Spreadsheet merupakan salah satu tools yang telah disediakan oleh Google secara gratis, dapat diakses dimana saja dan kapan saja, melalui ponsel, tablet, atau komputer bahkan saat tidak ada koneksi internet [10]. Untuk menghubungkan antara Google Spreadsheet dengan IFTTT, pertama yang dilakukan adalah membuat akun di <https://ifttt.com/>. Pembuatan akun ini bersifat gratis, namun jika diinginkan layanan yang lebih baik maka dianjurkan untuk menggunakan layanan berbayar.

Setelah membuat akun, maka selanjutnya yang dilakukan adalah membuat applet. Applet digunakan untuk menghubungkan device atau aplikasi yang berbeda. Applet adalah semacam sistem atau jembatan yang dapat menghubungkan antara aplikasi 1 dengan aplikasi lainnya, sehingga dapat diotomatisasi melalui layanan IFTTT. Setelah membuat applet, selanjutnya adalah memilih servis yang akan digunakan yaitu webhooks. Webhooks adalah salah satu sarana berkomunikasi antar aplikasi atau sistem dengan lebih efektif. Komunikasi yang terjadi saat menggunakan sarana tersebut berbasis event. Artinya, transfer informasi baru terjadi saat ada input atau tindakan, yang kemudian memicu tindakan lain [11] [12].



Gambar 3. Skema sederhana webhook

Webhooks tidak sama dengan API meskipun sama-sama dapat memberikan data. Letak perbedaannya adalah jika API hanya akan memberikan data yang diminta, maka webhooks akan memberikan semua data tanpa diminta. Selanjutnya harus ditentukan trigger yang akan digunakan. Trigger yang dipilih adalah receive web request karena sistem ini akan memanfaatkan web request untuk mendapatkan data dari sensor. Gambar 3 dan gambar 4 merupakan proses membuat Trigger dan proses konfigurasi Trigger yang mana akan diaktifkan setiap kali layanan menerima permintaan web untuk tentang suatu peristiwa [13].

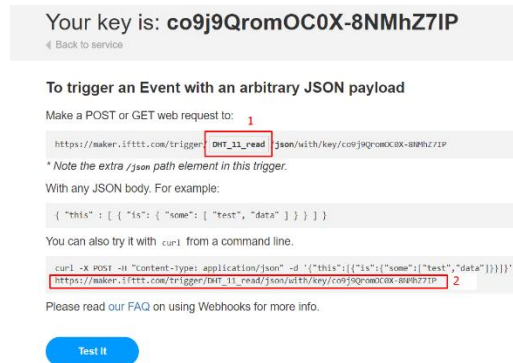
Gambar 4. Membuat Trigger

Gambar 5. Konfigurasi Trigger

Selanjutnya adalah membuat nama trigger yang akan digunakan, sebagai contoh digunakan nama DHT\_11\_read. Sampai pada titik ini maka telah diselesaikan langkah if this, Dan selanjutnya adalah then that. Then that adalah reaksi yang dilakukan bila ada permintaan dari web, dalam hal ini adalah dari sensor. Than that ini nantinya akan menghubungkan dengan Google Spreadsheet. Pilih menu than that kemudian cari Google Spreadsheet. Dengan memilih Google Spreadsheet, maka data hasil dari pembacaan sensor akan di simpan di Google Spreadsheet. Untuk itu perlu dilakukan beberapa konfigurasi, yaitu akun yang digunakan, kemudian nama Spreadsheet, format baris serta alamat file. Akun ini akan terpilihkan secara otomatis sesuai dengan akun yang digunakan untuk mendaftar.

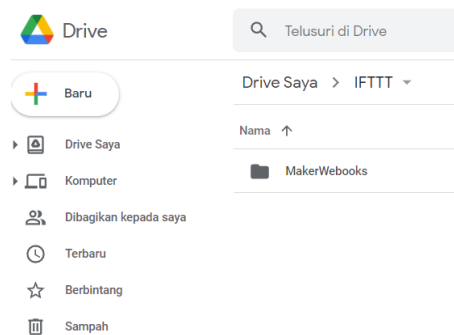
Spreadsheet name ini adalah nama file Spreadsheet yang dikehendaki, dapat diisi sesuai dengan keinginan. Formatted row berisi dengan tanggal pengambilan data yang akan ter-stored dalam format teks, event name berisi nama even yang dipakai yaitu DHT\_11\_read. Value1 adalah variable untuk suhu sedangkan Value2 adalah untuk variable kelembaban udara. Sedangkan Drive Folder Path adalah letak Spreadsheet disimpan dalam Google Drive.

Setelah selesai mengkonfigurasi applet, maka selanjutnya ada menguji koneksi antara applet dengan Google Spreadsheet seperti yang ditampilkan pada gambar 5 dimana proses dilakukan melalui halaman documentation. Pada halaman ini akan menunjukkan kunci API yang akan digunakan sebagai variable IFTTT URL resource pada program Arduino IDE.



Gambar 6. API key IFTTT

Selain API key, pada gambar 5 harus didefinisikan nama trigger event yang digunakan yaitu DHT\_11\_read agar dapat mengenali trigger yang telah dibuat. Kemudian juga IFTTT URL resource yang ditunjukkan oleh angka 2 pada gambar 5. URL ini digunakan pada program Arduino IDE sebagai variable IFTTT URL resource. Kemudian tombol test it digunakan untuk menguji apakah applet telah berfungsi yang ditunjukkan dengan adanya folder IFTTT pada google drive dan juga file Spreadsheet yang telah di create secara otomatis. Spreadsheet ini nanti yang akan menampung pembacaan data dari sensor.



Gambar 7. Folder di Google Drive

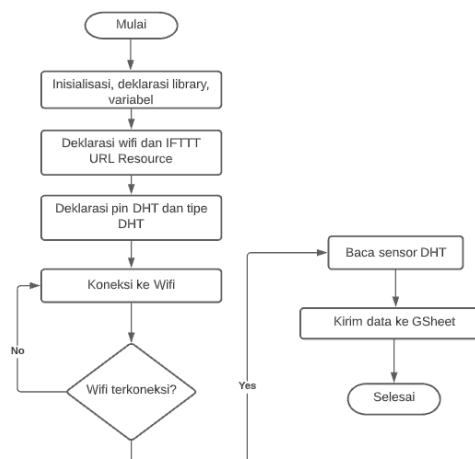
	A	B	C	D
1	March 25, 2022 at 08:30AM	DHT_11_read	Suhu	Kelembaban
2	March 25, 2022 at 08:30AM	DHT_11_read	Suhu	Kelembaban
3	March 25, 2022 at 08:30AM	DHT_11_read	Suhu	Kelembaban
4				
5				
6				
7				
8				
9				

Gambar 8. Google Sheet hasil pengujian Applet

Dari gambar 7 dan gambar 8 terlihat bahwa pengujian applet menghasilkan folder dan file spreadsheet dengan 4 kolom yaitu timestamp, event trigger, suhu dan kelembaban. Nilai suhu dan kelembaban ini belum berupa nilai hasil pembacaan karena belum terhubung ke sensor.

#### 4.4. Merancang Program di Arduino IDE

Untuk memprogram ESP32, diperlukan aplikasi Arduino IDE (Integrated Development Environment) bawaan dari Arduino. Arduino adalah mikrokontroler open source yang dapat dengan mudah diprogram, dihapus dan diprogram ulang setiap saat [14]. Diperkenalkan pada tahun 2005, platform Arduino dirancang untuk menyediakan cara yang murah dan mudah bagi penghobi, pelajar, dan profesional untuk membuat perangkat yang berinteraksi dengan lingkungan mereka menggunakan sensor dan aktuator [15]. Software ini berguna untuk membuka, membuat, mengedit source code pada Arduino dengan sebutan “sketches”. Sketch merupakan source code yang berisi logika dan algoritma yang akan diunggah ke dalam IC mikrokontroler (Arduino). Didalam sketch Arduino ini dikonfigurasi bagaimana menginisialisasi sensor yang ada, konfigurasi wifi, konfigurasi IFTTT hingga proses pembacaan sensor. Secara garis besar, digambarkan dalam flowchart seperti gambar 9 dibawah ini.



Gambar 9. Flowchart Sistem Program pada ESP32

Bahasa pemrograman yang digunakan adalah bahasa C/C++. Program pada Arduino sendiri dibagi menjadi 3 yaitu (i) Structure, (ii) Values (berisi variable dan konstanta) dan yang terakhir (iii) function. Structure kode pada arduino yaitu berisi fungsi `setup()`, yang dipanggil pertama kali ketika menjalankan sketch yang digunakan sebagai tempat inisialisasi variable, pin mode, penggunaan library dan lainnya. Fungsi ini dijalankan sekali ketika board dinyalakan atau di reset dan `loop()` yang akan melakukan perulangan, memungkinkan program untuk mengubah dan menanggapi. Loop digunakan untuk mengontrol board Arduino. Sedangkan function adalah segmentasi kode yang memungkinkan programmer untuk membuat potongan-potongan modular kode yang melakukan tugas khusus, misalnya fungsi konversi ke persen atau fungsi pembulatan. Program untuk mendeklarasikan library, wifi, IFTTT, waktu jeda dan sensor dapat dilihat pada cuplikan source code berikut ini.

```

// Deklarasi Library
#include <WiFi.h>
#include <DHT.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>

// Deklarasi SSID and Password
const char* ssid = "AkakomJanti (T.3.1)";
const char* password = "";

// Deklarasi IFTTT URL resource

```

```
const char* resource = "https://maker.ifttt.com/trigger/DHT_11_read/with/key/co9j9QromOC0X-8NMhZ7IP";

// Maker Webhooks IFTTT
const char* server = "maker.ifttt.com";

// Deklarasi Waktu Jeda
uint64_t uS_TO_S_FACTOR = 1000000; // konversi mikrodetik ke detik
// sleep for 30 minutes = 1800 seconds
uint64_t TIME_TO_SLEEP = 60; //jeda 60 detik

//Deklarasi Sensor dan PIN DHT11
#define DHTPIN 4 //Pin data dari DHT terhubung ke pin D4 ESP32
#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);
```

Penggunaan DHT11 sebagai sensor tidak serta merta dapat langsung di aplikasikan, karena membutuhkan library tambahan yaitu DHT.h. Library adalah kumpulan kode program Arduino dasar yang dikemas sedemikian agar dapat memberikan perintah terhadap suatu komponen sehingga mampu berkerja sesuai fungsinya. Library berguna untuk memudahkan pengguna dalam penulisan sketch atau program. Selain penambahan library DHT.h, perlu juga ditambahkan library lain. Pada bagian deklarasi ini juga dideklarasikan SSID dan password agar ESP32 dapat terhubung ke wifi. Deklarasi IFTTT URL resource yang didapat saat membuat koneksi pada IFTTT.com dituliskan pada bagian ini. Deklarasi jeda waktu disini dimaksudkan untuk memberikan jeda pembacaan selama 60 detik. Tipe sensor DHT11 dan pin yang digunakan juga dideklarasikan pada bagian ini.

```
void setup() {
  Serial.begin(115200);
  delay(1000);
  dht.begin();
  initWifi();
  makeIFTTTRequest();

  // Jeda mode for 3600 detik
  Serial.println("Going to sleep now");
  ESP.deepSleep(TIME_TO_SLEEP * uS_TO_S_FACTOR);
}
```

Perintah Serial.begin(115200) digunakan untuk menentukan kecepatan pembacaan data melalui port serial. Delay(1000) memberikan jeda waktu selama 1 detik. Pada potongan program yang ditunjukkan dilakukan inisialisasi untuk pembacaan data melalui sensor DHT serta inisialisasi untuk koneksi ke wifi dan IFTTT.

```
void loop() {
  float t = dht.readTemperature(); //membaca DHT11
}

// Melakukan koneksi ke Wifi
void initWifi() {
  Serial.print("Menghubungkan ke: ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);

  int timeout = 10 * 4; // 10 seconds
  while(WiFi.status() != WL_CONNECTED && (timeout-- > 0)) {
    delay(250);
    Serial.print(".");
  }
  Serial.println("");

  if(WiFi.status() != WL_CONNECTED) {
    Serial.println("Gagal koneksi, coba lagi");
  }
  Serial.print("WiFi terhubung : ");
  Serial.print(millis());
  Serial.print(", Alamat IP: ");
  Serial.println(WiFi.localIP());
}
```

```
// Melakukan HTTP request ke the IFTTT web service
void makeIFTTRequest() {
  Serial.print("Menghubungkan ke ");
  Serial.print(server);

  WiFiClient client;
  int retries = 5;
  while(!client.connect(server, 80) && (retries-- > 0)) {
    Serial.print(".");
  }
  Serial.println();
  if(!client.connected()) {
    Serial.println("Gagal terhubung...");
  }
  Serial.print("Meminta resource URL: ");
  Serial.println(resource);
}
```

Script diatas menunjukkan program untuk membaca sensor DHT11. Selain itu juga dilakukan koneksi ke wifi dengan menggunakan SSID dan password yang telah dideklarasikan pada bagian deklarasi. Selanjutnya juga dilakukan koneksi ke server IFTTT dengan perintah makeIFTTRequest dimana alamat URL nya telah dideklarasikan sebelumnya.

```
// Mengirimkan data suhu dan kelembaban
String jsonObject = String("{\"value1\":" + dht.readTemperature() + "\",\"value2\":" +
+ dht.readHumidity() + "\"}");
client.println(String("POST ") + resource + " HTTP/1.1");
client.println(String("Host: ") + server);
client.println("Koneksi : close\r\nContent-Type: application/json");
client.print("Panjang Koneksi: ");
client.println(jsonObject.length());
client.println();
client.println(jsonObject);
int timeout = 5 * 10; // 5 seconds
while(!client.available() && (timeout-- > 0)){
  delay(100);
}
if(!client.available()) {
  Serial.println("Tidak ada respon...");
}
while(client.available()){
  Serial.write(client.read());
}
Serial.println("\nmenghentikan koneksi");
client.stop();
}
```

Kemudian pada potongan script diatas menunjukkan program untuk mengirimkan data suhu dan kelembaban ke IFTTT. Setelah selesai menuliskan program maka selanjutnya dilakukan verify dan unggah kedalam ESP32. Jika program berhasil diunggah maka hasil pembacaan data akan ditampilkan ke spreadsheet yang telah dikonfigurasi. Hasil dari pembacaan terlihat pada tabel 1 data pembacaan menampilkan waktu pembacaan data, nama event, suhu udara dan kelembaban udara. Data dibaca dan ditampilkan dengan jeda 60 detik.

Tabel 1. Hasil pembacaan sensor yang tersimpan di Google Spreadsheet

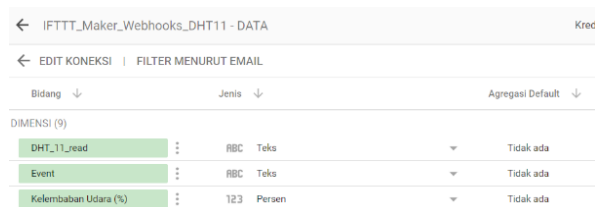
TimeStamp	Event	Suhu	Kelembaban
March 29, 2022 at 08:17PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:19PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:20PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:21PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:22PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:23PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:24PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:25PM	DHT_11_read	27.60	67.00



March 29, 2022 at 08:26PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:27PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:28PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:29PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:30PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:31PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:32PM	DHT_11_read	27.60	67.00
March 29, 2022 at 08:34PM	DHT_11_read	27.60	67.00

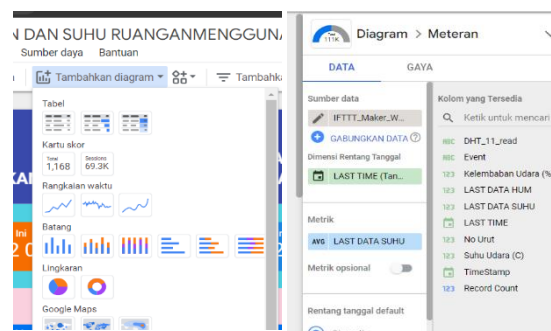
#### 4.5. Visualisasi Data

Setelah data sensor masuk ke dalam Spreadsheet, data tersebut akan dihubungkan dengan Google Looker Studio untuk ditampilkan dalam dashboard yang lebih informatif. Penulis menghubungkan antara Google Looker Studio dengan Google Spreadsheet yang telah dihasilkan pada proses membuat applet. Setelah terhubung, yang perlu dilakukan adalah melakukan konfigurasi terhadap tipe data seperti yang terlihat pada gambar 10 berikut ini.



Gambar 10. Konfigurasi data pada Google Looker Studio

Setelah selesai melakukan konfigurasi, maka proses ini dapat diselesaikan dengan meng-klik tombol selesai pada pojok kanan atas. Maka akan terlihat tampilan dashboard kosong yang berwarna putih. Pada halaman inilah akan divisualisasikan hasil pembacaan suhu dan kelembaban udara dengan menambahkan grafik melalui menu “tambah diagram” sehingga muncul pilihan grafik yang akan digunakan.



Gambar 11. Menambahkan Grafik dan Field

Penulis memilih menggunakan tipe grafik meteran yang terletak dipaling bawah untuk menampilkan data suhu dan kelembaban terakhir. Untuk menampilkan suhu dan data terakhir, perlu dilakukan sedikit modifikasi pada Spreadsheet, yaitu dengan cara menambahkan 1 kolom untuk memberikan nomor urut data. Pemberian ini diperlukan karena timestamp yang masuk ke spreadsheet bukan dalam format tanggal namun teks, sehingga tidak dapat dilakukan pengurutan berdasarkan waktu. Setelah memberikan nomor urut secara otomatis dengan menggunakan fungsi tertentu, maka data tersebut di pindahkan ke sheet lain menggunakan perintah Query pada Spreadsheet. Query ini berfungsi sama halnya dengan Query pada SQL, yaitu menampilkan data pada rentang tertentu dan mengurutkannya. Dengan demikian data terakhir bisa ditampilkan.

The screenshot shows a Google Sheet with two tables. The first table contains sensor data for temperature and humidity over time. The second table is a summary of the last time, temperature, and humidity readings.

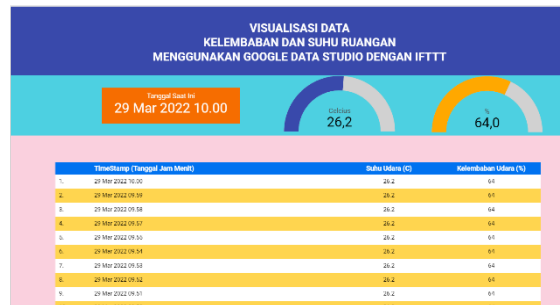
A	B	C	D	E
March 26, 2022 at 08:04PM	DHT_11_read	20.50	72.00	1961
March 26, 2022 at 08:05PM	DHT_11_read	20.50	72.00	1962
March 26, 2022 at 08:06PM	DHT_11_read	20.50	72.00	1963
March 26, 2022 at 08:07PM	DHT_11_read	20.50	72.00	1964
March 26, 2022 at 08:08PM	DHT_11_read	20.50	72.00	1965
March 26, 2022 at 08:09PM	DHT_11_read	20.50	72.00	1966
March 26, 2022 at 08:10PM	DHT_11_read	20.50	72.00	1967
March 26, 2022 at 08:11PM	DHT_11_read	20.50	72.00	1968
March 26, 2022 at 08:12PM	DHT_11_read	20.50	72.00	1969

F	G	H
LAST TIME	Last Tem	Last Hum
29/03/2022 9:47:58	26.20	64.00

Gambar 12. Hasil pemberian nomor, penggunaan query dan hasil data terakhir

Setelah berhasil menambahkan grafik informasi untuk suhu pada dashboard, maka hal yang sama bisa dilakukan untuk kelembaban. Selain itu bisa juga ditambahkan hasil pembacaan selama 10 menit terakhir menggunakan tabel. Kemudian pemberian judul terhadap dashboard ini juga dilakukan dengan menambahkan teks pada bagian atas dari dashboard.



Gambar 13. Dashedboard hasil visualisasi data

Agar dashboard ini dapat diakses melalui perangkat lain secara luas, maka dashboard ini harus di share melalui menu “bagikan” yang terletak di pojok kanan atas. Pada menu ini dapat diatur bahwa siapa saja yang boleh melihat dashboard ini. URL yang diperoleh dari proses ini dapat di gunakan untuk mengakses dashboard dari manapun kapanpun selama terdapat koneksi internet. Dashboard ini belum mampu melakukan auto refresh, sehingga data yang masuk hanya akan terekam pada Spreadsheet namun tidak ditampilkan di Google Looker Studio, sehingga pada browser harus ditambahkan add-on Looker Studio Auto Refresh yang di tanamkan pada browser Chrome. Add-on ini di set untuk dapat melakukan auto refresh setiap 60 detik.

#### 4.6. Pengujian Sistem

Pengujian dilakukan terhadap data yang masuk dan juga terhadap perubahan dashboard. Pengujian pertama adalah dengan mengakses dashboard dari lain selama 5 menit. Kemudian pengujian yang kedua adalah pengaksesan dengan menggunakan smartphone juga selama 5 menit. Dari hasil pengujian terlihat bahwa data yang masuk akan selalu diperbaharui setiap 1 menit.



Gambar 14. Hasil pengujian dengan smartphone

## 5. KESIMPULAN

Dari percobaan diatas dapat disimpulkan bahwa melakukan monitoring suhu dan kelembaban ruangan berbasis IoT dapat juga divisualkan menggunakan Google Looker Studio. Proses visualisasi ini melibatkan aplikasi pihak ketiga yaitu IFTTT. Visualisasi yang dihasilkan menjadi lebih menarik karena disajikan dalam bentuk grafis pada dashboard. Namun masih terdapat beberapa kendala yaitu timestamp pengambilan data yang bukan ber-format waktu, sehingga membutuhkan kolom khusus untuk dapat mendapatkan data terakhir. Data yang terekam di Google Spreadsheet menjadikan pengolahan data dapat dilakukan dengan mudah seperti layaknya pengolahan data pada spreadsheet lain. Selain itu penggunaan IFTTT yang relatif sederhana memudahkan untuk menghubungkan antara piranti IoT dengan Spreadsheet. IFTTT ini tidak sepenuhnya gratis, karena pada dasarnya layanan ini layanan berbayar. Namun setidaknya, untuk visualisasi data suhu dan kelembaban ruangan berbasis IoT ini dapat teratasi dengan baik.

## DAFTAR PUSTAKA

- [1] D. Apriani, M. Aan, and W. E. Saputra, "Data Visualization Using Google Data Studio," *Int. J. Cyber IT Serv. Manag.*, vol. 2, no. 1, 2022, doi: 10.34306/ijcitsm.v2i1.68.
- [2] Hendra and W. Andriyani, "Studi Komparasi Menyimpan Dan Menampilkan Data Histori Antara Database Terstruktur Mariadb Dan Database Tidak Terstruktur Influxdb," *J. Teknol. Technoscientia*, vol. 12, no. 2, pp. 168–174, 2020.
- [3] M. Mucchetti, "Google Data Studio," in *BigQuery for Data Warehousing*, 2020.
- [4] F. . Aditya, "Perancangan Dashboard Sebagai Pendukung Keputusan Berdasarkan Data Antrean Pada Aplikasi Antribos (Studi Kasus: Medical Center ITS)," 2018.
- [5] P. Khadapkar, "How to build a BI dashboard using Google Data Studio and BigQuery | Google Cloud Big Data and Machine Learning Blog | Google Cloud Platform," *Google Cloud Platform*, 2017. .
- [6] Heri Purnadi, "PEMANFAATAN GOOGLE SPREADSHEET DAN GOOGLE DATA STUDIO SEBAGAI DASHBOARD SUHU DAN KELEMBABAN DI LABORATORIUM," *Insa. Metrol. PPSDK*, vol. 1, no. 1, 2021, doi: 10.55101/ppsdk.v1i1.639.
- [7] L. H. Sinaga, "Monitoring Suhu Dan Kelembaban Udara Menggunakan Iot (Internet of Things) Berbasis Nodemcu Esp8266," *Monit. Suhu Dan Kelembaban Udar. Menggunakan Iot (Internet Things) Berbas. Nodemcu Esp8266*, vol. 19, no. 2, 2020.
- [8] K. Iriyanta, B. P. D. Putranto, and W. Andriyani, "IOT BASED SOIL MOISTURE MONITORING AND SOIL MOISTURE PREDICTION USING LINEAR REGRESSION (CASE STUDY OF VINCA PLANTS)," *J. Intell. Softw. Syst.*, vol. 2, no. 1, 2023, doi: 10.26798/jiss.v2i1.929.
- [9] M. Saeidi, M. Calvert, A. W. Au, A. Sarma, and R. B. Bobba, " If This Context Then That Concern : Exploring users' concerns with IFTTT applets ," *Proc. Priv. Enhancing Technol.*, vol. 2022, no. 1, 2022, doi: 10.2478/popets-2022-0009.
- [10] Q. Aini, U. Rahardja, I. Handayani, M. Hardini, and A. Ali, "Utilization of google spreadsheets as activity information media at the official site alphabet incubator," 2019.
- [11] I. Bafadal *et al.*, "Designing Consultation Chatbot Using Telegram API and Webhook-based NodeJS Applications," *Proc. 7th Int. Conf. Educ. Technol. (ICET 2021)*, vol. 601, 2021.
- [12] E. A. Kavats and A. A. Kostenko, "ANALYSIS OF CONNECTION METHODS OF TELEGRAM ROBOTS WITH SERVER PART," *Syst. Technol.*, vol. 3, no. 122, 2019, doi: 10.34185/1562-9945-3-122-2019-03.
- [13] T. Istiana, R. Indra A, G. S. Budhi Dharmawan, and B. Prakoso, "Pengembangan Sistem Diseminasi Prakiraan Cuaca Menggunakan Aplikasi Bot Telegram dengan Metode Webhook," *Elektron J. Ilm.*, vol. 12, no. 1, 2020, doi: 10.30630/eji.12.1.159.
- [14] M. Vital Carrillo, "Arduino Introduction," *Publicación Semest.*, vol. 9, no. 17, 2021.
- [15] L. Louis, "Working Principle of Arduino and Using it as a Tool for Study and Research," *Int. J. Control. Autom. Commun. Syst.*, vol. 1, no. 2, 2016, doi: 10.5121/ijcacs.2016.1203.