

# Design Of Weather Condition Monitoring System Based On Internet Of Things


Perancangan Sistem Monitoring Kondisi Cuaca Berbasis *Internet of Things*


Alifia Sekar Ratri, Vecky C. Poekoel, Arthur M. Rumagit

Jurusan Teknik Elektro, Universitas Sam Ratulangi, Jl. Kampus Bahu, 95115, Indonesia

e-mails : [alifiasekarratri@gmail.com](mailto:alifiasekarratri@gmail.com), [vecky.poekoel@unsrat.ac.id](mailto:vecky.poekoel@unsrat.ac.id), [arthur\\_rumagit@unsrat.ac.id](mailto:arthur_rumagit@unsrat.ac.id)

Received: [date]; revised: [date]; accepted: [date]

  
Vecky C. Poekoel

  
Arthur M. Rumagit

*Abstract — Weather forecasts are currently unpredictable because they often change rapidly and continuously over time. Weather in one area with other areas has different weather parameters. Therefore, weather conditions are indispensable information and are widely used to monitor weather changes that are constantly changing in controlled areas such as homes, industries, campuses, and others. However, a problem occurs when an accurate weather report for the current time is required. With the weather reporting system, all weather parameter sensors will be controlled by the ESP32 microcontroller as a server that will send all the data collected by the sensors to the database. In this final project, actual weather information can be obtained through the development of technology-based weather monitoring technology that can be accessed without having to be directly in the desired area using Internet of Things (IoT) technology. The purpose of this design is to make it easier to get weather information via the internet and can directly analyze changes in rainfall, temperature, humidity and air pressure in certain areas. The process of sending data can be seen through the OVoRD platform, so that information can be received more quickly in real-time. Weather parameters that can be monitored include temperature, rainfall, humidity, light intensity, wind direction and wind speed. The results of the tool test state that the tool has been operating well, can transmit data in real-time and can be used to monitor the weather based on the Internet of Things. The OVoRD platform has been integrated with tools where the program has entered the API code so that the interface on the OVoRD platform can display sensor data.*

**Key words**— *IoT; Microcontroller; Temperature Sensor; Weather Monitoring.*

Ramalan cuaca akhir-akhir ini tidak bisa diprediksi karena sering berubah-ubah secara cepat dan terus menerus setiap waktu. Cuaca pada suatu daerah dengan daerah lainnya memiliki parameter cuaca yang berbeda-beda. Oleh karena itu, kondisi cuaca merupakan suatu informasi yang sangat diperlukan dan banyak digunakan untuk memantau perubahan cuaca yang terus berubah di area tekontrol seperti rumah, industri, kampus, dan lain- lain. Namun, masalah terjadi saat dibutuhkan laporan cuaca yang akurat untuk waktu saat ini. Dengan cuaca sistem pelaporan semua sensor parameter cuaca akan dikendalikan oleh mikrokontroler ESP32 sebagai server yang akan mengirim semua data yang dikumpulkan oleh

sensor ke database. Dalam tugas akhir ini informasi cuaca yang aktual dapat diperoleh melalui pengembangan teknologi pemantauan cuaca berbasis ilmu teknologi yang dapat diakses tanpa harus berada langsung pada daerah yang diinginkan menggunakan teknologi *Internet of Things* (IoT). Tujuan dari perancangan ini untuk memudahkan dalam mendapatkan informasi cuaca melalui jaringan internet dan dapat langsung menganalisis perubahan curah hujan, suhu, kelembaban dan tekanan udara pada daerah tertentu. Proses pengiriman datanya dapat dilihat melalui platform OVoRD, sehingga informasi dapat lebih cepat diterima secara *real-time*. Parameter cuaca yang dapat dimonitor antara lain suhu, curah hujan, kelembaban udara, intensitas cahaya, arah angin dan kecepatan angin. Hasil dari pengujian alat menyatakan bahwa alat telah beroperasi dengan baik, dapat mengirimkan data secara *real-time* dan dapat digunakan untuk memantau cuaca berbasis *Internet of Things*. Platform OVoRD sudah terintegrasi dengan alat dimana pada program sudah dimasukkan kode API agar interface pada platform OVoRD dapat menampilkan data sensor.

**Kata kunci** — *IoT; Mikrokontroler; Monitoring Cuaca; Sensor Suhu.*

## I. PENDAHULUAN

Informasi mengenai suatu kondisi cuaca saat ini sudah menjadi kebutuhan umum bagi masyarakat karena banyak aktifitas yang bergantung pada kondisi cuaca. Kondisi cuaca dapat berubah-ubah secara cepat dan ekstrim sewaktu-waktu dapat menyebabkan masyarakat kurang tanggap terhadap dampak yang diakibatkan [1]. Unsur cuaca yang akan diamati dapat dijadikan sebagai bahan prakirakan cuaca pada waktu mendatang. Informasi cuaca yang paling umum digunakan adalah suhu, kelembaban udara, curah hujan, kecepatan angin, arah angin, dan cahaya[2].

Saat ini semakin mudah untuk mendapatkan suatu informasi dengan adanya jangkauan internet yang luas. Semua pengguna internet dapat mendapatkan informasi apapun itu dalam waktu yang singkat. Selain menjadi sarana untuk berbagi suatu informasi, internet juga dapat digunakan untuk mengontrol atau mengendalikan suatu objek melalui *Internet of Things* (IoT). *Internet of Things* merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus[3]. Dengan memanfaatkan *Internet of Things*, seseorang dapat membangun

sebuah sistem pemantauan cuaca yang dapat digunakan didaerah atau area yang diinginkan, untuk mengamati keadaan cuaca karena sering terjadi perbedaan cuaca antara suatu kawasan dengan kawasan lainnya.

Untuk melakukan pemantauan perubahan kondisi cuaca secara *real-time* dan mendapatkan informasi data mengenai kondisi cuaca dapat dilakukan dengan membangun sebuah perangkat keras yang terhubung dengan sistem pemantauan kondisi cuaca. Berdasarkan latar belakang tersebut, penulis berkeinginan untuk merancang sistem monitoring kondisi cuaca berbasis IoT.

#### A. Penelitian Terkait

- 1) Penelitian yang dilakukan oleh Tri Rahajoeningoem, Ivan Heru Saputra [1] mengenai Sistem Monitoring Cuaca dan Deteksi Banjir pada Android Berbasis *Internet of Things* (IoT). Pada penelitian ini akan dibuat suatu sistem monitoring keadaan cuaca dan ketinggian level air yang dapat memberikan peringatan bila terjadi perubahan cuaca yang ekstrim maupun banjir di wilayah hulu sehingga nantinya masyarakat di wilayah hilir dapat mempersiapkan diri untuk menghindarinya.
- 2) Penelitian yang dilakukan oleh Faza Ulya, Muhammad Kamal, Azhar [2] mengenai Rancang Bangun Sistem Monitoring Cuaca Dengan Tampilan *Thingspeak*. Penelitian ini bertujuan untuk memudahkan dalam mendapatkan informasi cuaca melalui jaringan internet dan dapat langsung menganalisis perubahan curah hujan, suhu, kelembaban dan tekanan udara pada daerah tertentu.
- 3) Penelitian yang dilakukan oleh Totok Sugiyanto, Arif Fahmi, Razki Nalandari [4] mengenai Rancang Bangun Sistem Monitoring Cuaca Berbasis *Internet Of Things* (IoT), tujuan dibuatnya sistem ini untuk memberikan sebuah informasi perubahan kondisi cuaca secara *real-time* kepada masyarakat dan informasi tersebut bisa di akses dengan mudah oleh semua orang melalui *web*.
- 4) Penelitian yang dilakukan oleh Ferdy Erwan, Abdul Muid, Irma Nirmala [5] mengenai Rancang Bangun Sitem Pengukur Cuaca Otomatis Menggunakan arduino dan terintegrasi dengan *Website*. Pada Fakultas MIPA Universitas Tanjungpura terdapat kuliah lapangan yang memerlukan data cuaca sebagai bahan penelitian. Biasanya terdapat masalah dalam melakukan pengumpulan data cuaca karena tidak memiliki sarana informasi untuk memenuhi kebutuhan tersebut. Tujuan penelitian ini adalah merancang sistem pengukur cuaca otomatis untuk mengukur suhu dan kelembaban, sensor BMP180 untuk mengukur tekanan udara, sensor BH1750 untuk mengukur intensitas cahaya, sensor kecepatan angin, sensor arah angin, dan sensor curah hujan. Data yang terkumpul dikirim ke *server* yang kemudian ditampilkan pada *website*.

#### B. Internet Of Things (IoT)

*Internet of Things* atau dikenal juga dengan singkatan IoT, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang sudah meluas. Cara kerja IoT yaitu dengan memanfaatkan sebuah algoritma pemrograman yang menghasilkan argumen yang dimana tiap-tiap perintah argumennya itu menghasilkan sebuah interaksi

antara sesama perangkat yang terhubung secara otomatis dalam jarak berapa pun. Internetlah yang menjadi media penghubung di antara kedua interaksi perangkat tersebut, sementara peran manusia hanya bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung.

#### C. Sensor

Sensor adalah perangkat yang merespon setiap perubahan dalam fenomena fisik atau variabel lingkungan, seperti cahaya, gaya, suhu, tekanan, listrik, gerakan, kelembapan, dan kecepatan. Perubahan ini memengaruhi sifat fisik, kimia, atau elektromagnetik dari sensor yang selanjutnya diproses ke bentuk yang lebih dapat digunakan dan dibaca oleh suatu rangkaian elektronik [6]. Setelah mengamati perubahan, *Input* yang terdeteksi akan dikonversikan menjadi *output* yang diubah menjadi tampilan yang mudah dibaca dan dipahami oleh pengguna, baik melalui perangkat sensor itu sendiri atau di transmisikan secara elektronik melalui jaringan untuk dibaca atau diproses menjadi data yang bermanfaat bagi pengguna. Sensor digunakan untuk pendeteksian saat melakukan suatu aktivitas seperti pengukuran dan pengendalian.

- 1) Sensor *rain gauge* yang bisa dihubungkan dengan mikrokontroler arduino maupun mikrokontroler lainnya sebagai sensor pembaca curah hujan baik dalam satuan *inch of rain* maupun dalam satuan *mm of rain*. Curah hujan merupakan salah satu unsur cuaca yang datanya diperoleh dengan cara mengukurnya dengan menggunakan alat penakar hujan, sehingga dapat diketahui jumlahnya dalam satuan milimeter (mm) [7]. Ini digunakan untuk mendeteksi hujan. Ini juga bisa digunakan untuk mengukur intensitas hujan. Ini memiliki keluaran digital juga hasil analog. Modul ini mengukur kelembaban melalui pin keluaran analog dan ketika ambang kelembaban melebihi batas banyak itu memberikan keluaran digital. Lebih air atau resistansi yang lebih rendah berarti lebih rendah tegangan keluaran.
- 2) Sensor kecepatan angin atau *anemometer* adalah sebuah perangkat untuk mengukur kecepatan angin dan untuk mengukur arah angin. Anemometer merupakan salah satu instrumen yang sering digunakan oleh Badan Meterologi Klimatologi dan Geofisika (BMKG) [8]. *Anemometer* harus ditempatkan di daerah terbuka.
- 3) Sensor arah angin adalah sensor yang mendeteksi 8 arah angin antara lain arah angin utara, timur laut, timur, tenggara, selatan, barat daya, barat, dan barat laut. Saat mendeteksi sensor (logika 0) dan jika tidak mendeteksi sesor (logika 1)[9]
- 4) ESP32 *WIFI* merupakan modul *wifi* yang berfungsi sebagai perangkat tambahan mikrokontroler seperti arduino agar dapat terhubung langsung dengan *wifi* dan membuat koneksi TCP/IP. Modul ini membutuhkan daya sekitar 3.3v dengan memiliki tiga mode *wifi* yaitu *station*, *access point* dan *both* (Keduanya). Modul ini juga dilengkapi dengan prosesor, memori dan GPIO dimana jumlah pin bergantung dengan jenis esp32 yang kita gunakan. Sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler. Pada alat penyiraman otomatis, pin esp32 yang digunakan ada

3 yaitu pin *supply*, pin GPIO, dan pin sistem minimum (*pull up*)[10]. Arduino Mega 2560 adalah *board* mikrokontroler yang berbasis pada ATmega2560. Memiliki 54 pin *input / output digital* (15 dapat digunakan sebagai *output PWM*), 16 *input analog*, 4 UART (*port serial perangkat keras*), osilator kristal 16 MHz, koneksi USB, colokan listrik, *header ICSP*, dan tombol *reset*. Berisi semua yang dibutuhkan untuk mendukung mikrokontroler, cukup hubungkan ke komputer dengan kabel USB atau nyalakan dengan adaptor AC-ke-DC atau baterai untuk memulai.

- 5) GY-302 *Ambient Light Sensor* adalah modul sensor cahaya yang menggunakan *chip* sensor BH1750FVI dengan koneksi serial I2C. Sensor ini mempunyai sensitivitas yang tinggi, rentang pengukuran cahaya yang luas dan tidak terpengaruh oleh sinar infra merah.
- 6) DHT22- sensor digital untuk mendeteksi suhu dan kelembaban di lingkungan. Tidak seperti pendahulunya DHT11, ia memiliki tingkat akurasi pengukuran yang tinggi dan dapat digunakan di jalan.

#### D. OVoRD

Platform OVoRD adalah suatu sistem yang menangani masalah manajemen perangkat iot hingga memvisualisasikan suatu kumpulan data.

## II. METODE

Waktu dan tempat penelitian perancangan alat ini dilakukan selama beberapa bulan. Penelitian dimulai pada bulan maret 2021. Tempat penelitian tugas akhir ini dilakukan di Program Studi Teknik Informatika, Universitas Sam Ratulangi.

Alat dan bahan penelitian yang akan digunakan yaitu sensor curah hujan, sensor arah angin, sensor kecepatann angin, sensor intensitas cahaya GY-302, sensor suhu dan kelembapan udara DHT 22, mikrocontroller ESP32 untuk mengolah data sensor dan mengirimkan data via WiFi.

#### A. Tahap – Tahap Penelitian

Dalam Melakukan penelitian ini ada beberapa tahap – tahap yang akan dilakukan oleh peneliti untuk menyelesaikan penelitian, tahap-tahap yang dilakukan yaitu :

##### 1). Studi Literatur

Studi Literatur adalah proses pengumpulan data atau pencarian referensi yang sesuai melalui media buku, *paper riset*, jurnal dan skripsi. Pada tahap ini akan dilakukan penelusuran serta pembelajaran terhadap berbagai macam literatur seperti buku, jurnal, skripsi dan referensi melalui internet yang berkaitan dengan kondisi cuaca, arah mata angin, curah hujan, intensitas cahaya, suhu dan kelembapan udara, *Internet of Things*, deteksi sensor dan pengiriman data, serta pemrograman bahasa C/C++ dalam penggunaan pemrograman arduino, pembacaan sensor, serta pengiriman data menggunakan modul *mikrocontroller* ESP32.

##### 2). Observasi

Observasi ini merupakan metode pengumpulan data dengan mengamati secara langsung dilokasi atau pengamatan secara

langsung terhadap objek penelitian. Dalam tahap ini peneliti akan melakukan pengumpulan data dengan mengambil bukti-bukti berupa foto gambar, membuat tabel percobaan, serta mengamati keadaan lingkungan secara langsung.

Dalam tahap ini objek penelitian adalah keadaan cuaca, maka akan dilakukan pengamatan secara menyeluruh terhadap pembacaan sensor pada keadaan cuaca pada situasi cuaca normal dan tidak normal.

##### 3). Analisa Kebutuhan

Dalam tahap ini merupakan proses menganalisis kebutuhan mengenai apa saja yang akan diperlukan dan digunakan untuk melakukan penelitian. Seperti dalam pembacaan data pada sensor dan pengiriman data ke *web* melalui modul ESP-32.

##### 4). Perancangan dan Pembuatan

Pada tahap ini akan dilakukan perancangan algoritma pada pembacaan sensor dan pengiriman data dari objek penelitian, penulisan kode sumber program dari sensor-sensor dan pengiriman data dari objek penelitian akan dibuat dalam *software* Arduino IDE dan kemudian setelah penulisan kode sumber program baca sensor dan pengiriman data sudah selesai di buat, maka akan di tampilkan pada *dashboard website* tahap ini merupakan proses menganalisis kebutuhan mengenai apa saja yang akan diperlukan dan digunakan untuk melakukan penelitian. Seperti dalam pembacaan data pada sensor dan pengiriman data ke *web* melalui modul ESP-32.

##### 5). Uji Coba

Dalam tahap ini akan dilakukan proses pengujian sensor-sensor dan juga pengujian pengiriman data yang sudah di program pada *software* Arduino IDE secara otomatis. akan di uji apakah data pembacaan sensor dapat di terima oleh *website*.

##### 6). Penulisan Skripsi

Dalam tahap penulisan skripsi ini maka semua tahapan-tahapan penelitian yang sudah dilakukan oleh peneliti akan ditulis berupa laporan karya tulis penelitian yang didalamnya akan berisikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitan, manfaat penelitian, landasan teori, metodologi penelitian, hasil pembahasan dan kesimpulan dari penelitian.

#### B. Konsep Dasar Perancangan Alat

Konsep dasar dalam penelitian pembuatan aplikasi sistem monitoring kondisi cuaca ini menggunakan dua unsur utama, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*). Konsep dasar ini menjadi pedoman untuk merancang sesuatu, dimana konsep itu sendiri terdapat langkah-langkah dan petunjuk yang menunjang dalam desain. Berikut ini tahapan dari perancangan sistem monitoring. Berikut adalah bagan alur sistem dari monitoring kondisi cuaca yang akan dibuat.

Pada gambar 1 bagan sistem monitoring kondisi cuaca diatas dapat dilihat masukkan tersebut merupakan sensor, sensor-sensor ini digunakan untuk memonitoring kondisi cuaca. Sensor ini di hubungkan ke mikrokontroler ESP32 dengan menggunakan kabel jumper, setelah mikrokontroler membaca masukkan data dari sensor, kemudian data di kirim ke *server* untuk diolah. Kemudian data yang di terima di tampilkan ke

dalam halaman platform sehingga data dapat di lihat oleh pengguna.

Pada gambar 2 merupakan diagram pengiriman data, dimana hasil dari pembacaan sensor dalam mikrokontroler dikirim kedalam platform OVORD melalui modul WiFi. Platform yang akan dibuat nantinya berisi data real dari sensor, yang nanti nilainya akan ditampilkan dalam angka yang mudah dipahami.

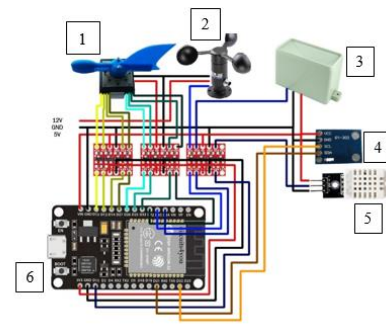
Pada gambar 3 rangkaian dari komponen sensor kecepatan angin, sensor arah angin, sensor curah hujan, sensor intensitas cahaya, sensor suhu dan kelembapan udara, yang terhubung ke dalam ESP-32, kemudian untuk pengiriman data menggunakan jaringan WiFi yang terkoneksi ke ESP-32, sehingga data-data dari pembacaan sensor dapat di kirimkan ke server.

Pada gambar 4 perancangan sistem monitoring kondisi cuaca ini menghubungkan beberapa komponen sebagai berikut, komponen sensor arah angin komponen sensor kecepatan angin, komponen sensor curah hujan, komponen sensor intensitas cahaya GY-302, komponen sensor suhu dan kelembaban DHT-22, komponen sistem pengiriman data WiFi ESP32 development kit

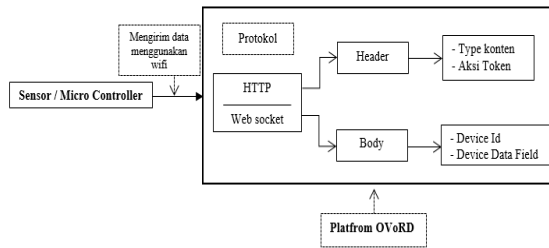
Sistem pengkabelan dari sistem monitoring kondisi cuaca ini terhubung dengan suplai listrik tegangan DC yang berasal dari baterai yang terhubung dengan regulator step up/down untuk memberikan suplai tegangan pada rangkaian.

Flowchart perancangan sistem menggunakan alat berbasis microcontroller yang telah dilengkapi sensor kecepatan angin, sensor arah angin, sensor curah hujan, sensor intensitas cahaya, sensor suhu dan kelembapan udara. Alat ini akan dipasang di titik tertentu, alat ini akan menerima data dari sensor dan mengirimkan data tersebut setiap 10 detik ke server. Server yang dimaksud di sini adalah website.

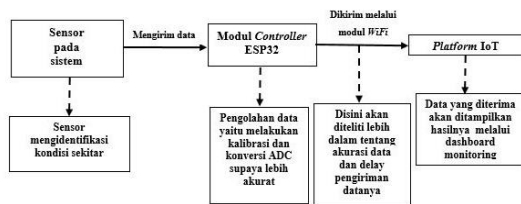
Data yang dikirimkan ke website akan membantu menyimpan data ke OVORD. Data ini nantinya akan diolah dan hasil perhitungannya akan ditampilkan ke dalam bentuk tampilan angka yang mudah dipahami.



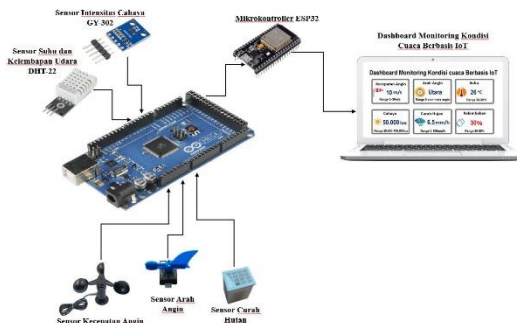
Gambar 4. Sistem Pengkabelan dari Komponen yang Dibutuhkan



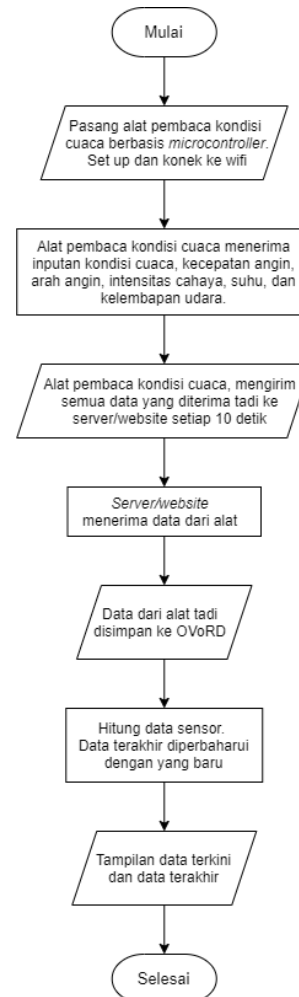
Gambar 1. Bagian Sistem Monitoring Kondisi Cuaca



Gambar 2. Diagram Blok Pengiriman Data Pada Platform OVORD



Gambar 3. Desain Komponen



Gambar 5. Flowchart Sistem



### III. HASIL DAN PEMBAHASAN

#### A. Lingkungan Implementasi

Perancangan sistem monitoring kondisi cuaca ini menggunakan beberapa perangkat sebagai berikut:

##### 1). Implementasi Perangkat Keras

Pada gambar 6 Sistem monitoring kondisi cuaca ini menggunakan perangkat keras sebagai berikut:

ESP32 *development kit* sebagai *board* yang digunakan merancang aplikasi IoT. Sensor DHT22 sebagai sensor yang digunakan untuk pengukuran suhu dan kelembaban dengan *range* pengukuran suhu -400C – 800C dan *range* pengukuran kelembaban 0 – 100% RH. Sensor GY-302 sebagai sensor yang digunakan untuk pengukuran intensitas cahaya berbasis IC BH1750 dengan *range* pengukuran 0 – 65535 lux.

Sensor ukur kecepatan angin dengan model JL-FS2 Waterproof sebagai sensor yang digunakan untuk pengukuran kecepatan angin dengan *range* pengukuran 0 – 60m/s. Sensor curah hujan v1.2 dengan tipe *tipping bucket* menggunakan *hall effect* sebagai sensor yang digunakan untuk pengukuran curah hujan dengan *range* pengukuran per tip bernilai 0.053 *inchi of rain* atau 1.346 mm of rain.

##### 2). Implementasi Perangkat Lunak

Implementasi perangkat lunak pada perancangan sistem monitoring kondisi cuaca berbasis *Internet of Things* ini menggunakan perangkat lunak *Microsoft Windows 10 x 64* sebagai sistem operasinya.



Gambar 5. Perangkat Keras Sistem Monitoring Kondisi Cuaca

#### B. Pengujian Sistem

##### 1) Pengujian Mikrokontroler ESP-32

Mikrokontroler ESP32 berfungsi sebagai otak atau pengendali dari rangkaian yang mudah dihubungkan dan digunakan karena memiliki semua pin yang terbuka, serta memiliki modul *WiFi* sehingga sistem dapat tersambung ke jaringan. Pengujian dilakukan dengan membuat dan mengeksekusi program untuk menghidupkan dan mematikan LED *internal* yang terdapat pada mikrokontroler. Tujuan pengujian adalah untuk menguji program yang bekerja pada mikrokontroler tersebut.

##### 2) Pengujian Sensor Kecepatan Angin

Pengujian sensor kecepatan angin dilakukan untuk mencari data hasil kalibrasi oleh mikrokontroler dan membandingkannya dengan alat ukur *anemometer*. Dengan mengetahui selisih atau *error* maka dapat dicari persentase *error* alat yang dibuat.

Keterangan Tabel 1 merupakan hasil pengujian perbandingan nilai kecepatan angin yang telah diukur menggunakan sensor kecepatan angin pada alat dan *anemometer digital*. Data menunjukkan adanya selisih antara alat ukur dengan *output* sensor.

Namun selisih atau persentase *error* masih dalam batas toleransi dengan demikian dapat dikatakan sensor bekerja dengan baik. Persentase *error* pengukuran didapatkan dari pembagian nilai selisih pembacaan dengan nilai *anemometer digital* kemudian dikalikan 100%.

$$Error = \frac{\text{Selisih nilai pembacaan}}{\text{Nilai anemometer}} \times 100\% \quad (1)$$

Berdasarkan rumus (1) di atas, hasil perhitungan yang diperoleh adalah sebagai berikut :

$$\begin{aligned} Error &= \frac{0,06}{0,96} \times 100\% \\ &= 0,062 \times 100\% \\ &= 6,2\% \end{aligned}$$

TABEL 1  
 HASIL PENGUJIAN SENSOR KECEPATAN ANGIN

Waktu	Kecepatan Angin Alat (m/s)	Anemometer Digital (m/s)	Selisih	Error
12:10	0,9	0,96	0,06	6,2%
12:20	1,1	1,15	0,05	4,3%
12:30	0,5	0,57	0,07	12,2%
12:40	0,9	0,96	0,06	6,2%
12:50	0,9	0,96	0,06	6,2%
13:00	1,3	1,34	0,04	2,9%
Rata-rata error (%)				6,3%

### 3) Pengujian Sensor Temperatur dan Kelembapan Udara DHT-22

Sensor untuk mendeteksi kelembapan dan temperatur udara adalah sensor DHT22. DHT22 mempunyai keandalan yang sangat tinggi dan stabilitas jangka panjang yang sangat baik. Hal ini dikarenakan teknologi modul akuisisi digital yang tertanam di dalam sensor tersebut. Di dalam sensor tersebut terdapat sensor kelembapan kapasitif dan sensor suhu NTC yang dapat dihubungkan dengan mikrokontroler. Sensor merupakan sensor yang memiliki *output digital* yang telah terkalibrasi, sehingga dalam pengujian ini dilakukan pembacaan sensor dan membandingkannya dengan alat ukur *thermometer* dan *humidity* pada Lutron LM-8102 .

Untuk menguji sensor tersebut maka harus dibuat program di mikrokontroler ESP32 kemudian dibaca melalui serial hh. Pengujian sensor DHT22 ini bertujuan untuk mengetahui kemampuan sensor dalam menerima perubahan suhu dan kelembapan pada ruang terbuka. Hasil pengujian sensor suhu dan kelembapan adalah sebagai berikut :

$$Error = \frac{\text{Selisih nilai pembacaan}}{\text{Nilai thermometer}} \times 100\% \quad (2)$$

Berdasarkan rumus (2) di atas, hasil perhitungan yang diperoleh adalah sebagai berikut :

$$\begin{aligned} Error &= \frac{0,2}{31,1} \times 100\% \\ &= 0,0064 \times 100\% \\ &= 0,64\% \end{aligned}$$

Keterangan tabel 2 dibawah merupakan hasil pengujian yang dilakukan dengan membandingkan suhu dari *thermometer* dengan data suhu DHT-22. Terlihat perbedaan pengukuran tetapi tidak terlalu besar, dengan demikian dapat dikatakan sensor bekerja dengan baik dalam mendeteksi suhu dan kelembapan udara.

Persentase *error* pengukuran didapatkan dari pembagian nilai selisih pembacaan dengan nilai *thermometer* kemudian dikalikan 100%.

TABEL 2  
HASIL PENGUJIAN SENSOR KELEMBAPAN DHT-22

Waktu	Kelembapan Alat	Humidity Digital	Selisih	Error
12:10	69,5%	65,4%	4,1%	6,2%
12:20	70%	68,9%	1,1%	1,5%
12:30	68%	64,6%	3,4%	5,2%
12:40	67%	64%	3%	4,6%
12:50	66,6%	65,2%	1,4%	2,1%
13:00	65,8%	64,8%	1%	1,5%
		Rata-rata error (%)		3,5%

$$Error = \frac{\text{Selisih nilai pembacaan}}{\text{Nilai humidity digital}} \times 100\% \quad (3)$$

Berdasarkan rumus (3) di atas, hasil perhitungan yang diperoleh adalah sebagai berikut :

$$\begin{aligned} Error &= \frac{4,1}{65,4} \times 100\% \\ &= 0,062 \times 100\% \\ &= 6,2\% \end{aligned}$$

### 4) Pengujian Sensor Intensitas Cahaya

Pengujian sensor intensitas cahaya disini akan dilakukan untuk menguji apakah sensor tersebut bisa mendeteksi intensitas cahaya dengan baik dan sesuai dengan alat light meter yang sudah ada. GY-302 *digital light intensity sensor module* adalah sebuah modul sensor cahaya berbasis IC BH1750. BH1750 adalah sebuah IC sensor cahaya dengan antarmuka I<sup>2</sup>C. Modul ini memberikan nilai *output digital* melalui I<sup>2</sup>C bus, sehingga tidak perlu lagi menambahkan konverter ADC.

Pengujian sensor ini akan dilakukan dengan menghubungkan sensor ke ESP32 dan akan menggunakan cahaya dari ruang terbuka, lalu kita melihat hasil yang dibaca oleh sensor intensitas cahaya pada alat dan *light meter digital* kemudian membandingkannya dengan cara menghitung presentase *error* dari kedua parameter tersebut. Hasil pengukuran dapat dilihat pada tabel 3.

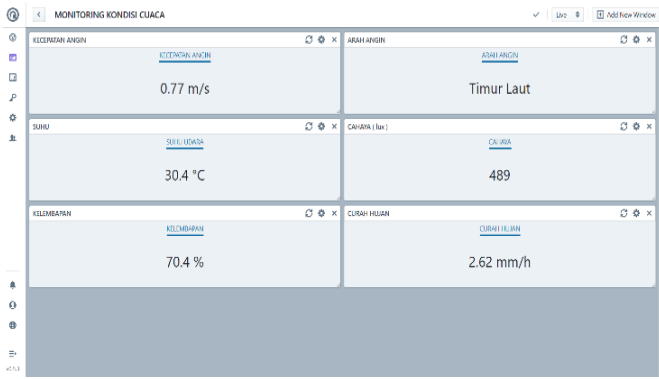
Keterangan tabel 3 dibawah ini merupakan hasil pengujian yang telah dilakukan dengan membandingkan intensitas cahaya pada alat dan *light meter digital*, terlihat bahwa sensor bekerja dengan baik. Pengukuran dibawah ini terdapat selisih antara alat ukur dengan *output* sensor.

Persentase *error* pengukuran didapatkan dari pembagian nilai selisih pembacaan dengan nilai *light meter* kemudian dikalikan 100%.

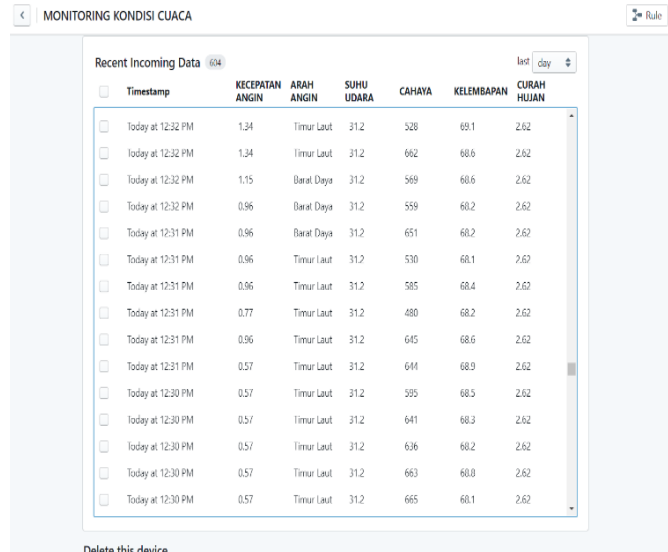
$$Error = \frac{\text{Selisih nilai pembacaan}}{\text{Nilai light meter}} \times 100\% \quad (4)$$

TABEL 3  
HASIL PENGUJIAN SENSOR INTENSITAS CAHAYA

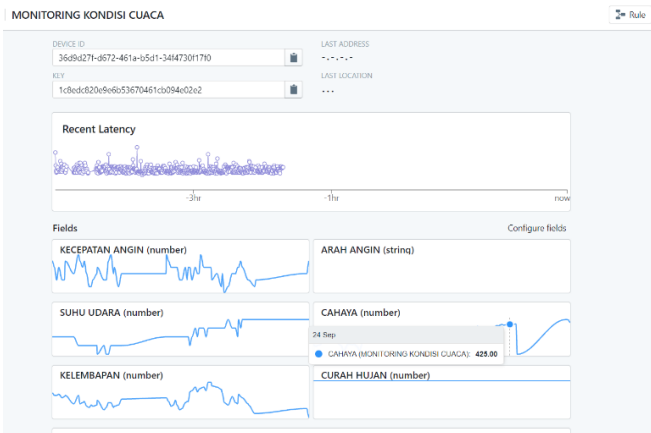
Waktu	Intensitas Cahaya	Light Meter	Selisih	Error
12:10	494	505	11	2,1%
12:20	547	550	3	0,5%
12:30	670	679	9	1,3%
12:40	660	671	11	1,6%
12:50	799	811	12	1,4%
13:00	898	908	10	1,1%
		Rata-rata error (%)		1,3%



Gambar 6. Tampilan *Dashboard* OVoRD



Gambar 8. Tampilan *Timestamp* OVoRD



Gambar 7. Tampilan *Fields* OVoRD

Berdasarkan rumus (4) di atas, hasil perhitungan yang diperoleh adalah sebagai berikut :

$$\begin{aligned}
 Error &= \frac{11}{505} \times 100\% \\
 &= 0,021 \times 100\% \\
 &= 2,1\%
 \end{aligned}$$

Pengujian secara keseluruhan dilakukan setelah semua komponen berhasil dipasang pada rangkaian utama yaitu mikrokontroler. Pengujian dilakukan dengan cara menjalankan sistem kemudian mengamati fungsi kerja dari sistem selama beberapa waktu. Untuk pengujian ini dibutuhkan sebuah *hotspot* agar sistem dapat berhubungan dengan jaringan lokal melalui sambungan *WiFi*. Nama *hotspot* telah disesuaikan dengan program yang dibuat dengan *password* tertentu. Saat diaktifkan Node Mcu ESP32 akan terhubung dengan *hotspot* yang telah diatur dan terkoneksi. Jika koneksi berhasil maka data akan siap untuk dikirim pada saat ada permintaan dan hasil data akan diperbarui setiap 10 detik. Untuk itu, *user* dapat mengakses data melalui *platform OVoRD* yang diatur dan hasil data akan berupa seperti pada gambar 6.

Pada gambar 6 merupakan tampilan *dashboard* OVoRD. Pada bagian ini berisi tampilan kecepatan angin (m/s), suhu (°C), kelembapan(%Rh), curah hujan (mm/h), intensitas cahaya (lux), arah angin.

TABEL 4  
HASIL PENGUKURAN CUACA  
HARI JUMAT, 24 SEPTEMBER 2021

Waktu	(m/s)	(°C)	(%Rh)	(lux)	Arah Angin	(mm/h)
12:10	0,96	30,9	69,5	494	Timur Laut	2.62
12:20	1,15	31	70	547	Timur Laut	2.62
12:30	0,57	31,2	68	670	Timur Laut	2.62
12:40	0,96	31,3	67	660	Timur Laut	2.62
12:50	0,96	31,6	66,6	799	Timur Laut	2.62
13:00	1,34	31,5	65,8	898	Timur Laut	2.62

Pada gambar 7 menampilkan tampilan grafik pengambilan data secara terus menerus setiap 10 detik secara *realtime* yang ditampilkan pada *platform OVoRD*.

Pada gambar 8 menampilkan seluruh data yang telah berhasil tersimpan pada *database* OVoRD. Parameter cuaca yang tersimpan berupa kecepatan angin, arah angin, suhu udara, intensitas cahaya, kelembapan, dan curah hujan yang diperbarui setiap 10 detik.

Pada tabel 4 merupakan hasil dari pengujian sistem monitoring kondisi cuaca berbasis *Internet of Things* yang dilakukan pada hari jumat, tanggal 24 september 2021 selama satu jam sesuai dengan permintaan dan hasil data yang diambil setiap 10 menit.

5) Pengujian Normalisasi Data Menggunakan SPSS

a. Metode Uji *Shapiro-Wilk*

Pengujian menggunakan metode uji *Shapiro-wilk* pada data kecepatan angin, intensitas cahaya,

## Tests of Normality

	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
KECEPATAN	.283	6	.143	.921	6	.514
PEMBANDING	.287	6	.135	.919	6	.500

a. Lilliefors Significance Correction

Gambar 9. Test Of Normality Kecepatan menggunakan Shapiro-Wilk

## Tests of Normality

	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
CAHAYA	.188	6	.200*	.960	6	.816
PEMBANDING	.188	6	.200*	.953	6	.764

\*. This is a lower bound of the true significance.

a. Lilliefors Significance Correction

Gambar 10. Test of Normality Intensitas menggunakan Shapiro-Wilk

temperatur, kelembapan dengan alat pembanding. Jika nilai Sig. hitung  $> 0,05$  (Sig. acuan) maka dapat dikatakan bahwa data berasal dari populasi yang berdistribusi normal. Sedangkan, jika nilai Sig. hitung  $< 0,05$  (Sig. acuan) maka dapat dikatakan bahwa data berasal dari populasi yang berdistribusi tidak normal. Data yang berdistribusi normal adalah data yang memiliki nilai  $> 0.05$  pada taraf signifikan 5%. Gambar diatas ini merupakan hasil perhitungan uji normalitas menggunakan uji *shapiro-wilk* dengan aplikasi SPSS terhadap data alat kondisi cuaca dan alat pembanding, sebagai berikut :

Berdasarkan gambar 9 hasil perhitungan uji normalitas pada data hasil kecepatan angin menggunakan uji *shapiro-wilk* diatas, dari diketahui bahwa kecepatan memperoleh nilai signifikansi 0.514 dan pembanding memperoleh nilai 0.500. Hasil tersebut menunjukkan bahwa nilai signifikansi kecepatan dan pembanding  $> 0.05$ , maka dapat disimpulkan bahwa data hasil Kecepatan Angin dan pembanding berdistribusi normal. Selanjutnya akan dilakukan pengujian menggunakan uji *t-test*.

Berdasarkan gambar 10 hasil perhitungan uji normalitas pada data hasil intensitas cahaya menggunakan uji *shapiro-wilk* diatas, dari diketahui bahwa cahaya memperoleh nilai signifikansi 0.816 dan pembanding memperoleh nilai 0.764. Hasil tersebut menunjukkan bahwa nilai signifikansi kecepatan dan pembanding  $> 0.05$ , maka dapat disimpulkan bahwa data hasil intensitas cahaya dan pembanding berdistribusi normal. Selanjutnya akan dilakukan pengujian menggunakan uji *t-test*.

Berdasarkan gambar 11 hasil perhitungan uji normalitas pada data hasil temperatur menggunakan uji *shapiro-wilk* diatas, dari diketahui bahwa temperatur memperoleh nilai signifikansi 0.801 dan pembanding memperoleh nilai 0.020. Hasil tersebut menunjukkan bahwa nilai signifikansi kecepatan dan pembanding  $< 0.05$ , maka dapat disimpulkan bahwa data hasil temperatur dan pembanding berdistribusi tidak normal.

Berdasarkan gambar 12 hasil perhitungan uji normalitas pada data hasil temperatur menggunakan uji *shapiro-wilk* diatas, dari diketahui bahwa kelembapan memperoleh nilai signifikansi 0.638 dan pembanding memperoleh nilai 0.028.

## Tests of Normality

	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
TEMPERATUR	.153	6	.200*	.958	6	.801
PEMBANDING	.315	6	.063	.750	6	.020

\*. This is a lower bound of the true significance.

a. Lilliefors Significance Correction

Gambar 11. Test of Normality Temperatur menggunakan Shapiro-Wilk

## Tests of Normality

	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
KELEMBAPAN	.188	6	.200*	.937	6	.638
PEMBANDING	.352	6	.019	.765	6	.028

\*. This is a lower bound of the true significance.

a. Lilliefors Significance Correction

Gambar 12. Test of Normality Kelembapan menggunakan Shapiro-Wilk

Hasil tersebut menunjukkan bahwa nilai signifikansi kecepatan dan pembanding  $< 0.05$ , maka dapat disimpulkan bahwa data hasil kelembapan dan pembanding berdistribusi tidak normal. Maka selanjutnya akan dilakukan pengujian menggunakan uji *wilcoxon*.

b. Metode Uji Shapiro-Wilk

Uji *shapiro wilk* adalah sebuah metode atau rumus perhitungan sebaran data yang dibuat oleh *shapiro* dan *wilk*. Metode *shapiro wilk* adalah metode uji normalitas yang efektif dan *valid* digunakan untuk sampel berjumlah kecil. Metode *shapiro wilk* menggunakan data dasar yang belum diolah dalam tabel distribusi frekuensi. Metode *shapiro wilk* menggunakan data dasar yang belum diolah dalam tabel distribusi frekuensi. Data diurut, kemudian dibagi dalam dua kelompok untuk dikonversi dalam *shapiro wilk*. Cara baca hasil perhitungan uji *shapiro wilk* adalah dengan melihat nilai *shapiro wilk* hitung dan tingkat Signifikansinya.

Uji *paired sample t test* merupakan bagian dari analisis statistik parametik. Oleh karena itu, sebagaimana aturan dasar dalam analisis statistik parametik, maka persyaratan utamanya adalah data penelitian haruslah berdistribusi normal. Untuk mengetahui apakah data yang dipakai dalam pengujian *paired sample t test* ini berdistribusi normal atau tidak, tentunya perlu melakukan uji normalitas terlebih dahulu menggunakan uji *shapiro wilk*.

Untuk Kecepatan angin dan intensitas cahaya sudah diuji menggunakan metode uji *shapiro wilk* hasilnya berdistribusi normal, kemudian akan dilanjutkan dengan uji *paired sample t test*.

Pada output gambar 13 terlihat bahwa hasil statistik deskriptif dari kedua sampel yang diteliti yakni nilai kecepatan dan perbandingan. Untuk nilai kecepatan diperoleh rata-rata kecepatan sebesar 0.9333. Sedangkan untuk nilai perbandingan sebesar 0.9900. jumlah data yang digunakan sebagai sampel penelitian adalah sebanyak 6 data. Untuk nilai Std. Error Mean untuk kecepatan sebesar 0.10853 dan untuk perbandingan 0.10443.



**T-Test**

Paired Samples Statistics					
	Mean	N	Std. Deviation	Std. Error Mean	
Pair 1	KECEPATAN	.9333	6	.26583	.10853
	PERBANDINGAN	.9900	6	.25581	.10443

Gambar 13. Paired Samples Statistics Kecepatan Angin

Paired Samples Correlations					
	N	Correlation	One-Sided p	Two-Sided p	
Pair 1	KECEPATAN & PERBANDINGAN	6	1.000	<.001	<.001

Gambar 14. Paired Samples Correlations Pada Kecepatan Angin

Paired Samples Test										
	Mean	Std. Deviation	Paired Differences			t	df	Significance		
			Std. Error Mean	95% Confidence Interval of the Difference Lower	Upper			One-Sided p	Two-Sided p	
Pair 1	KECEPATAN- PERBANDINGAN	-.05667	.01033	.00422	-.06751	-.04593	-13.440	5	<.001	<.001

Gambar 15. Paired Samples Test Pada Kecepatan Angin

Karena nilai rata-rata kecepatan pada gambar 13 adalah  $0.9333 < \text{perbandingan } 0.9900$ , maka itu artinya secara deskriptif ada perbedaan rata-rata hasil kecepatan dengan perbandingan. Selanjutnya untuk membuktikan apakah perbedaan tersebut benar-benar nyata (signifikan) atau tidak, maka perlu menafsirkan hasil uji *paired sample t test* yang terdapat pada tabel output “Paired Samples Test”.

Output gambar 14 diatas menunjukkan hasil uji korelasi atau hubungan antara kedua data atau hubungan variabel kecepatan dengan perbandingan. Berdasarkan output diatas diketahui nilai koefisien korelasi (*Correlation*) sebesar 1.000 dengan nilai signifikansi (Sig.) sebesar  $< 0.001$ . karena nilai Sig. kurang dari  $0,001 < \text{probabilitas } 0.05$ , maka dapat dikatakan bahwa ada hubungan antara variabel kecepatan dengan perbandingan.

Rumusan hipotesis penelitian sebagai berikut :

$H_0$  = Tidak ada perbedaan rata-rata antara data kecepatan dengan perbandingan yang artinya tidak ada pengaruh pada data sebelum dan sesudah

$H_a$  = Ada perbedaan rata-rata antara data kecepatan dengan perbandingan yang artinya ada pengaruh pada data sebelum dan sesudah.

Menurut Singgih Santoso (2014: 265). Pedoman pengambilan keputusan dalam uji *paired sample t-test* berdasarkan nilai signifikansi (Sig.) hasil output SPSS, adalah sebagai berikut:

Jika nilai Sig. (2-tailed)  $< 0.05$ , maka  $H_0$  ditolak dan  $H_a$  diterima dan sebaliknya, jika nilai Sig. (2-tailed)  $> 0.05$ , maka  $H_0$  diterima dan  $H_a$  ditolak

Berdasarkan gambar 15 pada ouput “Paired Sample Test” di atas, diketahui nilai Sig. (2-tailed) adalah sebesar  $0.001 < 0.05$ , maka  $H_0$  ditolak dan  $H_a$  diterima. Sehingga dapat disimpulkan bahwa ada perbedaan rata-rata antara data kecepatan dengan perbandingan yang artinya ada pengaruh pada data sebelum dan sesudah.

Berdasarkan pengujian pada gambar 16 “Paired Sample Test” di atas, diketahui nilai Sig. (2-tailed) adalah sebesar  $0.001 < 0.05$ , maka  $H_0$  ditolak dan  $H_a$  diterima. Sehingga dapat disimpulkan bahwa ada perbedaan rata-rata antara data cahaya dengan pembeding yang artinya ada pengaruh pada data sebelum dan sesudah.

**T-Test**

Paired Samples Statistics					
	Mean	N	Std. Deviation	Std. Error Mean	
Pair 1	CAHAYA	678.0000	6	151.29177	61.76461
	PEMBANDING	687.3333	6	152.59707	62.29749

Gambar 16. Paired Samples Statistics Intensitas Cahaya

Paired Samples Correlations					
	N	Correlation	One-Sided p	Two-Sided p	
Pair 1	CAHAYA & PEMBANDING	6	1.000	<.001	<.001

Gambar 17. Paired Samples Correlations Intensitas Cahaya

Paired Samples Test										
	Mean	Std. Deviation	Paired Differences			t	df	Significance		
			Std. Error Mean	95% Confidence Interval of the Difference Lower	Upper			One-Sided p	Two-Sided p	
Pair 1	CAHAYA- PEMBANDING	-9.33333	9.26589	1.33333	-12.76078	-5.90589	-7.000	5	<.001	<.001

Gambar 18. Paired Samples Test Pada Intensitas Cahaya

Pada output gambar 16 terlihat bahwa hasil *statistic* deskriptif dari kedua sampel yang diteliti yakni nilai intensitas cahaya dan pembeding. Untuk nilai intensitas cahaya diperoleh rata-rata cahaya sebesar 678.0000. Sedangkan untuk nilai pembeding sebesar 687.3333. Jumlah data yang digunakan sebagai sampel penelitian adalah sebanyak 6 data. Untuk nilai *Std. Error Mean* untuk cahaya sebesar 61.76461 dan untuk pembeding 62.29749. Karena nilai rata-rata cahaya  $678.0000 < \text{pembeding } 687.3333$ , maka itu artinya secara deskriptif ada perbedaan rata-rata hasil cahaya dengan pembeding. Selanjutnya untuk membuktikan apakah perbedaan tersebut benar-benar nyata (signifikan) atau tidak, maka perlu menafsirkan hasil uji *paired sample t test* yang terdapat pada tabel output “Paired Samples Test”.

Output gambar 17 diatas menunjukkan hasil uji korelasi atau hubungan antara kedua data atau hubungan variabel cahaya dengan pembeding. Berdasarkan output diatas diketahui nilai koefisien korelasi (*Correlation*) sebesar 1.000 dengan nilai signifikansi (Sig.) sebesar  $< 0.001$ . karena nilai Sig. kurang dari  $0,001 < \text{probabilitas } 0.05$ , maka dapat dikatakan bahwa ada hubungan antara variabel cahaya dengan pembeding.

Rumusan hipotesis penelitian sebagai berikut :

$H_0$  = Tidak ada perbedaan rata-rata antara data cahaya dengan pembeding yang artinya tidak ada pengaruh pada data sebelum dan sesudah

$H_a$  = Ada perbedaan rata-rata antara data cahaya dengan pembeding yang artinya ada pengaruh pada data sebelum dan sesudah.

Menurut Singgih Santoso (2014: 265). Pedoman pengambilan keputusan dalam uji *paired sample t-test* berdasarkan nilai signifikansi (Sig.) hasil output SPSS, adalah, jika nilai Sig. (2-tailed)  $< 0.05$ , maka  $H_0$  ditolak dan  $H_a$  diterima dan sebaliknya, jika nilai Sig. (2-tailed)  $> 0.05$ , maka  $H_0$  diterima dan  $H_a$  ditolak.

Berdasarkan ouput gambar 18 “Paired Sample Test” di atas, diketahui nilai Sig. (2-tailed) adalah sebesar  $0.001 < 0.05$ , maka  $H_0$  ditolak dan  $H_a$  diterima. Sehingga dapat disimpulkan bahwa ada perbedaan rata-rata antara data cahaya dengan pembeding yang artinya ada pengaruh pada data sebelum dan sesudah.

#### IV. KESIMPILAN DAN SARAN

##### A. Kesimpulan

Berdasarkan hasil penelitian dari tugas akhir ini, maka dapat ditarik beberapa kesimpulan yaitu sistem kerja dari monitoring kondisi cuaca berbasis *internet of things* selalu terhubung dengan *web server* apabila terkoneksi dengan internet dan arus. Data dari hasil pembacaan sensor dan ESP-32 dikirimkan ke OVoRD sebagai *platform* IoT. *Platform* OVoRD sudah terintegrasi dengan alat dimana pada program sudah dimasukkan kode API agar *interface* pada *platform* OVoRD dapat menampilkan data sensor. Pengujian pengiriman data dapat dikirim dengan baik secara realtime. Peningkatan kesalahan pengiriman data pada sistem dapat disebabkan perbedaan jarak antara ESP-32 dengan modem. Jarak ideal pada ruang terbuka antara ESP-32 dengan modem ialah kurang lebih 10 meter sehingga meminimalisir tingkat kesalahan dalam pengiriman data. Monitoring kondisi cuaca dapat dilakukan melalui aplikasi *web* yang digunakan yaitu *platform* IoT OVoRD, yang menampilkan data kecepatan angin, arah angin, curah hujan, suhu, kelembaban, dan intensitas cahaya.

##### B. Saran

Berdasarkan kesimpulan diatas, dalam pembuatan skripsi ini penelitian dan pengembangan dibutuhkan agar sistem yang dirancang dapat diterapkan secara lebih lengkap sebagai alat bantu untuk mengetahui keadaan cuaca dengan kondisi apapun dilapangan.

Tipe Thermal,” pp. 1–6, 2011.

- [9] H. Muliadi, “Desain dan Implementasi Sistem Pemantau Cuaca dengan Aplikasi Telegram Design and Implementation of Weather Monitoring Systems with Telegram Applications,” *Desain Dan Implementasi Sist. Pemantau Cuaca Dengan Apl. Telegr.*, 2018.
- [10] N. Ulpah, L. Kamelia, T. Prabowo, T. Elektro, T. Elektro, and F. Sains, “Rancang Bangun Penyiraman Otomatis Berbasis IoT Menggunakan Smartphone,” *Tek. Elektro, Fak. Sains dan Teknol. UIN Sunan Gunung Djati*, no. November 2020, pp. 279–286, 2020.



**Alifia Sekar Ratri** penulis adalah anak pertama dari dua bersaudara. Anak dari Heru Cahyono (ayah) dan Hidayah Salam (ibu). Lahir di Manado, pada tanggal 30 Oktober 2000. Penulis menempuh pendidikan pertama di TK Bunda Bestari tahun 2004-2005, kemudian melanjutkan ke Sekolah Dasar Negeri 03 Manado tahun 2005-2011, setelah itu melanjutkan Sekolah Menengah Pertama Negeri 01 Manado tahun 2011-2014, kemudian melanjutkan pendidikan di Sekolah Menengah Atas Negeri 01 Manado tahun 2014-2017. Tahun 2017 penulis melanjutkan studi S1 di Fakultas Teknik Jurusan Teknik Elektro, Program Studi Teknik Informatika, Universitas Sam Ratulangi. Selama perkuliahan penulis tergabung dalam organisasi Himpunan Mahasiswa Elektro dan juga organisasi Badan Tadzkir Fakultas Teknik.

#### V. KUTIPAN

- [1] T. Rahajoeningoem and I. H. Saputra, “Sistem Monitoring Cuaca dan Deteksi Banjir pada Android Berbasis Internet of Things (IoT),” *Pros. SAINTIKS FTIK UNIKOM*, pp. 33–40, 2017.
- [2] F. Ulya and M. Kamal, “Rancang Bangun Sistem Monitoring Cuaca Dengan Tampilan Thingspeak,” *J. Tektro*, vol. 1, no. 1, pp. 23–28, 2017.
- [3] Y. Efendi, “Internet Of Things (Iot) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile,” *J. Ilm. Ilmu Komput.*, vol. 4, no. 2, pp. 21–27, 2018, doi: 10.35329/jiik.v4i2.41.
- [4] H. Fuad, “Rancang bangun sistem monitoring ternak ayam berbasis internet of things (IoT),” *Inst. Teknol. Telkom Purwokerto*, vol. 02, no. 01, pp. 1–5, 2019.
- [5] F. Erwan, A. Muid, and I. Nirmala, “Rancang Bangun Sistem Pengukur Cuaca Otomatis Menggunakan Arduino Dan Terintegrasi Dengan Website,” *J. Coding, Sist. Komput. Untan*, vol. 06, no. 03, pp. 255–264, 2018.
- [6] B. A. B. Ii, “Universitas Sumatera Utara 6,” vol. 1, no. 2, pp. 6–38, 2001.
- [7] N. L. Mufidah, “Sistem Informasi Curah Hujan Dengan Nodemcu Berbasis Website,” *Ubiquitous Comput. its Appl. J.*, vol. 1, pp. 25–34, 2018, doi: 10.51804/ucaiaj.v1i1.25-34.
- [8] M. Yusuf and L. , Alrijadjis, “Desain Sensor Kecepatan Angin Dengan Kontrol Adaptif Untuk Anemometer