

Model Sistem Monitoring Serta Kendali Otomatis Suhu dan Kelembaban Ruangan Pada Budidaya Jamur Tiram Putih Berbasis Internet Of Things

Aldi Maulana¹, Samuel HT¹, Gunawan Tjahjadi¹, Kiki Prawiroredjo¹

¹Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Trisakti, Jakarta.

Informasi Artikel

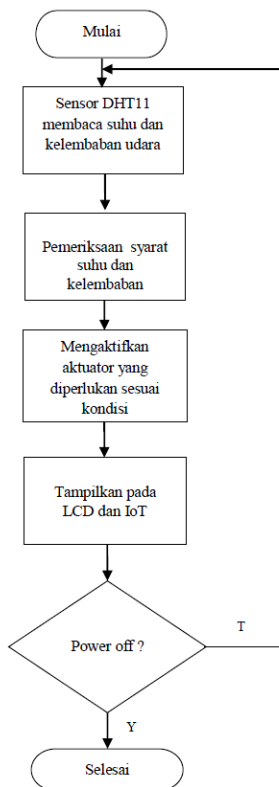
Naskah Diterima : 6 September 2020

Direvisi : 21 Desember 2020

Disetujui : 21 Desember 2020

*Korespondensi Penulis :
kiki.prawiroredjo@trisakti.ac.id

Graphical abstract



Abstract

White oyster mushroom is a very superior food commodity. However, the problem faced is that the temperature and humidity in the lowlands are very unstable and tend to be hot. To get optimal results, oyster mushroom growth requires room temperatures in the range of 26 to 30 °C, room humidity in the range 80% to 90% RH and soil humidity in the range 80 to 100% RH. To maintain and monitor this condition, we need a Monitoring System and Room Temperature and Humidity Automatic Control in white Oyster Mushroom Cultivation Based on the Internet of Things. The system designed regulates temperature, room humidity and soil humidity in real time and displays the data on a LCD in the mushroom cultivation room and send the data to a laptop or a smartphone of the cultivator. A DHT-11 sensor is used to detect room temperature and air humidity, and a soil moisture sensor is used with a probe that is planted into the soil to detect the soil moisture in the cultivation room. Sprayer or mist maker is used as an air conditioner, lamp and fan are used as a temperature and humidity stabilizer for the room and soil. An Arduino Uno with ATmega328P microcontroller is used as a component that controls the whole system. From the test results it is known that the system works properly to maintain the temperature in the range 26° - 30° C and humidity in the range 80% - 90% RH.

Keywords: White Oyster Mushroom, DHT11, Soil Moisture sensor, Internet of Things

Abstrak

Jamur tiram putih merupakan salah satu komoditi pangan yang sangat unggul. Permasalahan yang dihadapi adalah suhu dan kelembaban di dataran rendah sangat tidak stabil dan cenderung panas. Untuk mendapatkan hasil yang optimal pertumbuhan jamur tiram memerlukan suhu pada range 26° - 30° C, kelembaban udara berada pada range 80% - 90% RH dan kelembaban tanah pada range 80 - 100% RH. Untuk menjaga dan memonitor keadaan tersebut dibutuhkan sebuah Sistem Monitoring Serta Kendali Otomatis Suhu dan Kelembaban Ruangan pada Budidaya Jamur Tiram Putih Berbasis Internet of Things. Sistem yang dibuat dapat mengatur suhu, kelembaban ruangan dan kelembaban tanah secara real time dan menampilkan data tersebut pada sebuah LCD di ruang budidaya jamur dan pada laptop atau smartphone pembudidaya. Sebagai pendeteksi suhu dan kelembaban ruangan digunakan sensor DHT-11, sedangkan untuk mendeteksi kelembaban tanah digunakan sensor kelembaban tanah. Sprayer digunakan sebagai pendingin ruangan, lampu dan kipas angin digunakan sebagai penstabil suhu dan kelembaban ruangan dan tanah. Sebagai komponen pengatur kerja sistem secara keseluruhan digunakan Arduino Uno dengan mikrokontroler ATmega328P. Dari hasil pengujian diketahui bahwa sistem dapat berfungsi dengan baik menjaga suhu di daerah range 26° - 30° C, kelembaban ruangan dalam range 80% - 90% RH.

Kata kunci: jamur tiram putih, DHT11, sensor kelembaban tanah, Internet of Things

© 2020 Penerbit Jurusan Teknik Elektro UNTIRTA Press. All rights reserved

1. PENDAHULUAN

Jamur tiram putih (*Pleurotus Ostreatus*) merupakan salah satu jenis jamur kayu yang mempunyai prospek baik untuk dikembangkan sebagai diversifikasi bahan pangan serta kandungan gizinya setara dengan daging dan ikan. Permintaan pasar lokal dan ekspor terbuka lebar, waktu panennya singkat sekitar 1-3 bulan, bahan baku mudah didapat, dan tidak membutuhkan lahan yang luas, oleh karena itu jenis jamur ini mulai banyak dibudidayakan. Khasiat jamur tiram putih untuk kesehatan adalah menghentikan pendarahan dan mempercepat pengeringan luka pada permukaan tubuh, mencegah penyakit diabetes melitus, penyempitan pembuluh darah, menurunkan kolesterol darah, menambah vitalitas dan daya tahan tubuh, serta mencegah penyakit tumor atau kanker, kelenjar gondok, influenza, sekaligus memperlancar buang air besar [1].

Rumah jamur tiram atau kumbung terbuat dari bambu atau kayu, dinding kumbung biasanya terbuat dari gedek atau papan serta atapnya terbuat dari genteng atau sirap. Rumah jamur tiram tidak boleh menggunakan seng atau asbes karena dapat mengakibatkan suhu udara menjadi panas, sedangkan lantainya menggunakan tanah, karena membantu penyerapan air pada saat penyiraman. Hal lain yang harus diperhatikan adalah media tanah atau baglog menggunakan serbuk gergaji [2]. Pada Gambar 1 diperlihatkan gambar jamur tiram putih dalam baglog.



Gambar 1. Jamur Tiram Putih [2]

Budidaya jamur tiram putih saat ini masih menggunakan cara yang konvensional manual dimana bibit jamur tiram akan disemprotkan air menggunakan *sprayer* untuk menjaga kestabilan temperatur udara pada 26 – 30° C, kelembaban udara 80 – 90% RH pada ruangan dan kelembaban tanah 80 – 100% RH. Penyemprotan air harus dilakukan beberapa kali dalam satu hari. Hal ini bertujuan untuk menjaga kesuburan serta kualitas hasil budidaya jamur tiram putih. Jamur tiram putih dapat tumbuh dengan baik pada daerah dataran tinggi berkisar 600 mdpl [3].

Terdapat beberapa penelitian yang pernah dilakukan terkait otomatisasi menggunakan mikrokontroler telah dilakukan oleh Telaumbanua dkk. mengenai pengendalian iklim mikro tanaman sawi dalam *greenhouse* [4]. Penelitian terhadap sistem pemberian nutrisi dan penyiraman tanaman otomatis berdasarkan *real time clock* dan tingkat kelembaban tanah berbasis mikrokontroler Atmega32 [5]. merancang sebuah sistem kendali otomatis untuk mengendalikan temperatur dan kelembaban dalam kumbung jamur tiram berdasarkan perubahan nilai temperatur dan kelembaban ruang berbasis mikrokontroler [6].

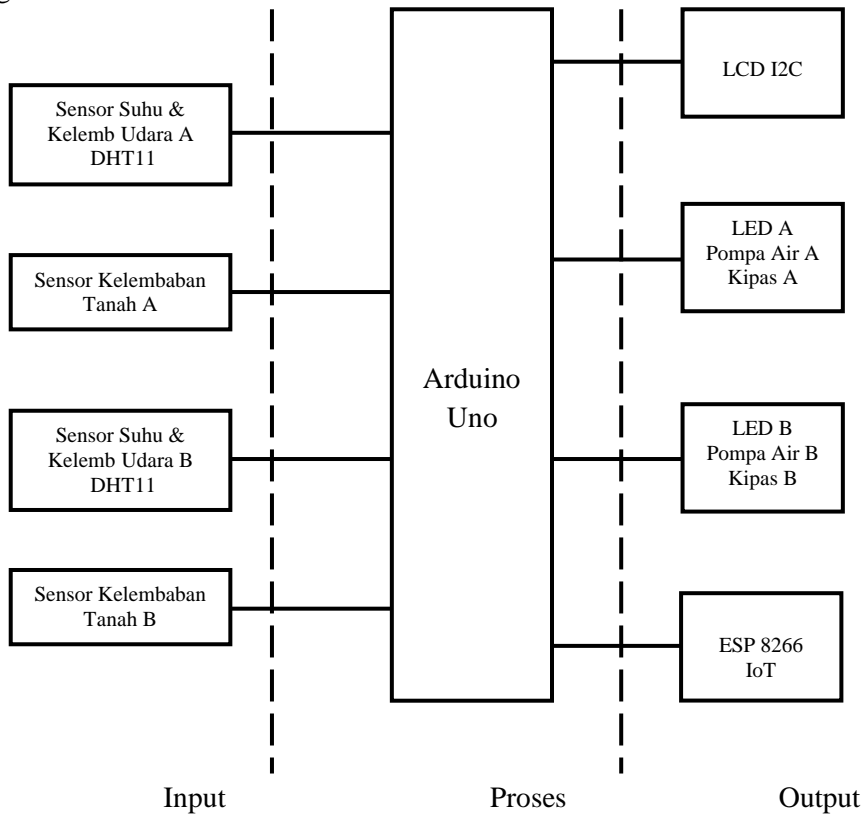
Dalam rangka memudahkan pembudidaya jamur tiram putih dalam membudidayakan tanamannya, dirancang sebuah model sistem yang mampu memonitor kondisi suhu, kelembaban udara dan kelembaban tanah serta melakukan kendali otomatis ketika kondisi ruang tempat budidaya tidak sesuai dengan standar. Monitoring dapat dilakukan dari jarak jauh dengan menggunakan teknologi *web*.

2. METODE PENELITIAN

Pada tahap ini dilakukan identifikasi masalah yang dihadapi dalam membudidayakan jamur tiram putih dan dilanjutkan dengan studi literatur yaitu membaca buku dan jurnal-jurnal yang

membahas tentang budidaya jamur tiram putih. Observasi dilakukan untuk mengetahui keadaan di lapangan tentang budidaya jamur tiram putih dan perancangan yang dilakukan untuk memecahkan masalah pada budidaya jamur tiram putih. Pada tahap perancangan dilakukan desain sistem yang dapat memonitor dan mengendalikan suhu dan kelembaban ruangan budidaya jamur tiram putih. Kemudian dilakukan pemilihan komponen untuk merealisasikan sistem yaitu sensor suhu dan kelembaban DHT11, sensor *soil moisture*, pompa air, lampu, kipas, dan Arduino Uno. Pemilihan sensor DHT11 karena mempunyai keandalan yang tinggi dan stabilitas yang baik untuk jangka panjang [7], sedangkan pemilihan Arduino Uno karena mudah digunakan, harga yang terjangkau, bahasa pemrogramannya mudah, dan kemampuannya yang canggih [8].

2.1 Diagram Blok Sistem

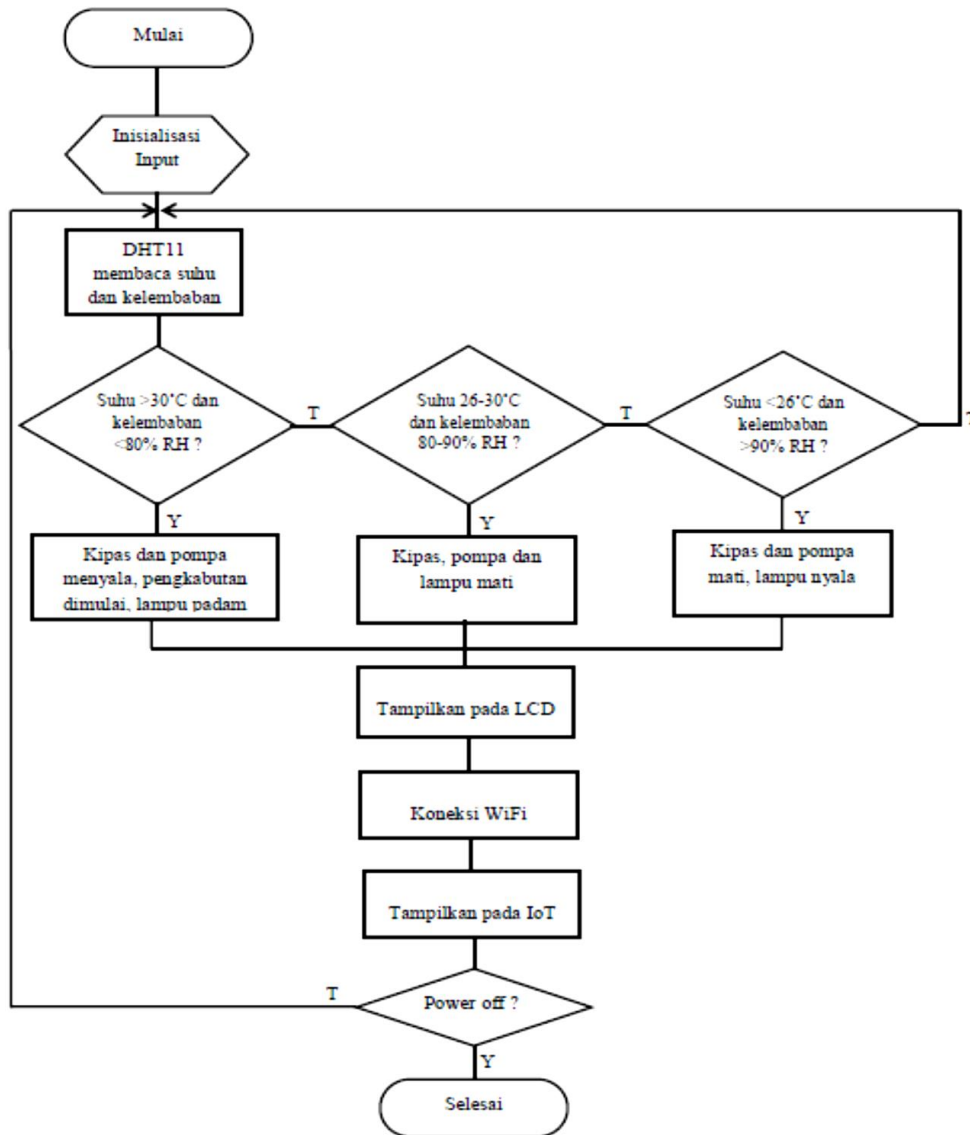


Gambar 2. Diagram Blok Sistem

Diagram blok sistem pada Gambar 2 memperlihatkan bagian input sistem terdiri dari dua buah sensor suhu dan kelembaban udara DHT11 dan dua buah sensor suhu dan kelembaban tanah yang diletakkan di tempat yang berbeda. Bagian pengendali kerja sistem secara keseluruhan adalah Arduino Uno dan bagian output adalah dua buah lampu LED untuk menghangatkan suhu ruangan, dua buah pompa air digunakan untuk memompa air keluar melalui alat penghasil kabut yang berfungsi untuk menjaga kelembaban ruangan dan tanah, dua buah kipas digunakan untuk menurunkan suhu ruangan, rangkaian LCD untuk menampilkan status suhu dan kelembaban ruangan secara *real time* sehingga dapat dimonitor oleh pembudidaya. Rangkaian ESP 8266 sebagai IoT berfungsi untuk mengirim data melalui internet sehingga pembudidaya dapat memonitor kondisi ruangan budidaya dimanapun berada melalui *smartphone* atau *laptop*. ESP8266 telah dilengkapi dengan kemampuan jaringan WiFi [9].

2.2 Perangkat Lunak Sistem

Diagram alir dari perangkat lunak sistem untuk sensor suhu dan kelembaban ruangan terdapat pada Gambar 3.



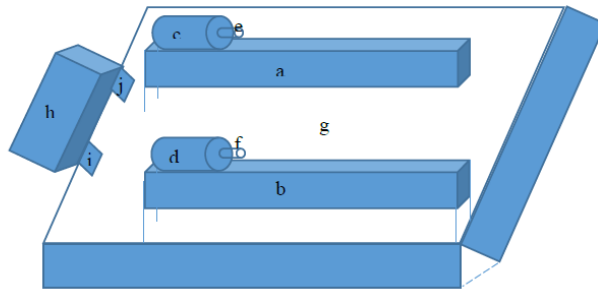
Gambar 3. Diagram Alir Perangkat Lunak Sistem

Dari diagram alir tersebut sensor DHT11 mendeteksi suhu dan kelembaban ruangan rumah jamur tiram. Apabila sensor DHT11 mendeteksi suhu di atas 30° C dan kelembapan berada di bawah 80% RH maka, output akan mengaktifkan kipas dan pompa untuk menyala dan proses pengkabutan dimulai untuk menurunkan suhu dan meningkatkan kelembapan ruangan. Jika sensor DHT11 mendeteksi suhu ruangan di antara 26 sampai dengan 30° C dan kelembapan berada di antara 80 sampai dengan 90% RH maka, kipas angin, lampu dan pompa tidak aktif. Jika sensor DHT11 mendeteksi suhu ruangan di bawah 26° C dan kelembapan berada di atas 90% RH maka, lampu akan menyala sebagai penghangat ruangan.

Setelah aktuator bekerja sesuai kondisi yang diperiksa maka status suhu dan kelembaban akan ditampilkan di LCD kemudian akan diaktifkan koneksi ke WiFi dan ditampilkan pada jarak jauh.

3. Hasil dan Pembahasan

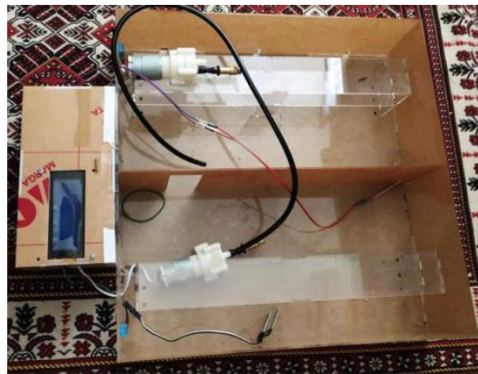
Setelah perancangan sistem dibuat dilakukan perancangan mekanik untuk tata letak komponen yang digunakan. Gambar 4 memperlihatkan tata letak perancangan mekanik sistem, sedangkan Gambar 5 adalah model dari alat yang dibuat dengan ukuran ruangan 40 cm × 40 cm × 10 cm.



Gambar 4. Perancangan Tata Letak Model Sistem

Dimana :

1. a dan b adalah wadah untuk baglog jamur tiram
2. c dan d adalah pompa air yang dilengkapi dengan motor DC 12V
3. e dan f adalah nozzle kuningan.
4. g adalah wadah untuk media tanam
5. h adalah wadah alat pengendali
6. i dan j adalah sensor DHT11



Gambar 5. Model Sistem Tanpa Kipas dan Lampu

3.1 Pengujian Kerja Komponen Output Terhadap Perubahan Suhu dan Kelembaban

Pengujian kerja komponen output dilakukan dengan mengubah suhu dan kelembaban yang bervariasi dan mencatat komponen mana yang aktif dengan kondisi suhu dan kelembaban yang terjadi. Tabel 4 memperlihatkan respon lampu, kipas angin dan pompa terhadap suhu yang konstan dan kelembaban yang bervariasi.

Tabel 4. Respon Alat Terhadap Perubahan Kelembaban

Percobaan ke	Suhu (°C)	Kelembaban (%RH)	Respon		
			Lampu	Kipas	Pompa
1.	26	45	menyala	mati	menyala
2.	26	50	menyala	mati	menyala
3.	26	55	menyala	mati	menyala
4.	26	60	menyala	mati	menyala
5.	26	65	menyala	mati	menyala
6.	26	70	menyala	mati	menyala
7.	26	75	menyala	mati	menyala
8.	26	80	menyala	mati	menyala
9.	26	85	menyala	mati	mati
10.	26	90	menyala	mati	mati
11.	26	95	menyala	menyala	mati

Mula-mula suhu ruangan dijaga konstan 26°C dengan kelembapan 45%RH, maka pompa akan mengaktifkan *sprayer* untuk meningkatkan kelembapan ruangan dan lampu akan menyala untuk meningkatkan suhu ruangan, sedangkan kipas angin akan mati karena suhu berada di level 26°C. Pada saat kelembapan udara 85%RH, maka pompa akan mati sehingga *sprayer* akan berhenti bekerja karena sudah mencapai kelembapan yang telah ditentukan. Ketika kelembapan mencapai 95% maka kipas akan menyala karena berfungsi untuk menurunkan kelembapan ruangan.

Pada Tabel 5 diperlihatkan respon lampu, kipas angin dan pompa terhadap suhu yang bervariasi sedangkan kelembapan dijaga konstan.

Tabel 5. Respon Alat Terhadap Perubahan Suhu

Percobaan ke	Suhu (°C)	Kelembaban (%RH)	Respon		
			Lampu	Kipas	Pompa
1.	26	45	menyala	mati	menyala
2.	27	45	menyala	mati	menyala
3.	28	45	menyala	mati	menyala
4.	29	45	menyala	mati	menyala
5.	30	45	menyala	mati	menyala
6.	31	45	mati	menyala	menyala

Pada pengujian ini kelembapan dijaga konstan sebesar 45%RH dan suhu dibuat bervariasi dari 26 sampai dengan 31° C. Dari pengujian didapat lampu akan menyala untuk meningkatkan suhu ruangan sampai dengan 30° C. Dalam rentang suhu antara 26 sampai dengan 30°C kipas angin akan mati karena pendinginan ruangan tidak diperlukan.

Dan pada saat suhu mencapai 31°C maka lampu akan mati karena sudah melebihi batas suhu yang telah ditentukan dan kipas akan menyala untuk membantu menurunkan suhu ruangan. Pompa air tetap menyala karena kelembapan dijaga tetap 45%.

3.2 Pengujian komunikasi Modul ESP8266 dengan IoT

Tahapan awal ketika menghubungkan Arduino, modul ESP8266 dan *IoT* harus dipastikan *hotspot device* (laptop atau *handphone*) terhubung dengan koneksi internet agar data pada web <https://thingspeak.com/channels/803625> dapat *terupdate*. Langkah selanjutnya mengganti *user hotspot device* menjadi “hotspotp” dan *password hotspot* diganti menjadi “1234567890” agar modul ESP8266 dan device dapat terkoneksi. Program Arduino IDE menggunakan *define* seperti pada Gambar 6 untuk diunggah ke mikrokontroler.

```

/*-----ESP8266 mySerial WiFi Module-----*/
#define SSID "hotspotp" // "SSID-WiFiname"
#define PASS "1234567890" // "password"
#define IP "184.106.153.149"// thingspeak.com ip
String msg = "GET /update?key=J8M696157B4YIXW2"; //change it with your key...
// https://thingspeak.com/channels/803625
/*-----*/
    
```

Gambar 6. *Define* Modul ESP8266 dengan *Hostpot* dan *Thingspeak*

Apabila LCD I2C sudah memunculkan tulisan “Connected” berarti mikrokontroler, *device* dan *IoT* sudah terhubung dengan baik seperti pada Gambar 7.



Gambar 7. Modul ESP8266 Sudah Terhubung dengan *Hotspot*

Ketika semua modul sudah terhubung, pengguna dapat membuka website <https://thingspeak.com/channels/803625> untuk mengetahui kondisi apakah yang sedang terjadi. Apabila pengguna ingin mengupdatenya cukup dengan *refresh* halaman *web* <https://thingspeak.com/channels/803625> dan hasil dari input sensor pada prototype dapat diunduh pada *web* tersebut. Data yang akan diunduh dapat dipilih, bisa menggunakan format JSON, XML ataupun CSV. Tabel 1 memperlihatkan data hasil unduhan CSV.

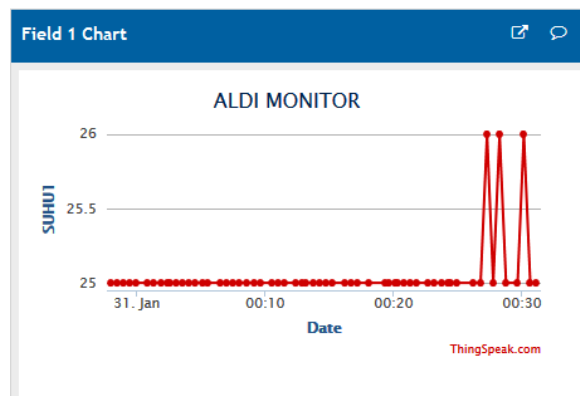
Tabel 1. Format CSV Pada Halaman Web Thingspeak

	A	B	C	D	E	F	G	H
1	created_at	entry_id	field1	field2	field3	field4	field5	field6
2	2019-07-03 19:35:05 UTC	85	28	73	0	28	71	0
3	2019-07-03 19:35:35 UTC	86	28	73	0	28	71	0
4	2019-07-03 19:36:04 UTC	87	28	74	0	28	71	0
5	2019-07-03 19:36:58 UTC	88	28	73	0	28	71	0

Format CSV lebih memudahkan pengguna untuk membaca data yang ada karena ditampilkan dalam baris dan kolom. Komunikasi ESP8266 dengan Arduino menggunakan komunikasi serial (AT + Command).

Untuk dapat menghubungkan *IoT* dengan Arduino dan mendapatkan data seperti diatas dibutuhkan beberapa langkah [10]:

1. Buat akun di *thingspeak* untuk mendapatkan *user api key*.
2. Masukkan *user api key* pada halaman *web thingspeak*.
3. Algoritma di dalam *sketch* Arduino dibagi menjadi dua bagian yaitu pada bagian untuk menghubungkan *router* ke *wifi* dieksekusi pada void setup () dan bagian untuk mengirim data ke *server*.
4. Setelah berhasil *verify* dan *uploading* pada aplikasi Arduino ide maka akan muncul tampilan seperti Gambar 8.



Gambar 8. Tampilan Pada Halaman *Web Thingspeak*

Sumbu vertikal merupakan inputan dari sensor DHT11 kiri sehingga diinisialkan dengan “SUHU1” dan sumbu horisontal merupakan waktu yang sedang berjalan ataupun sudah lampau.

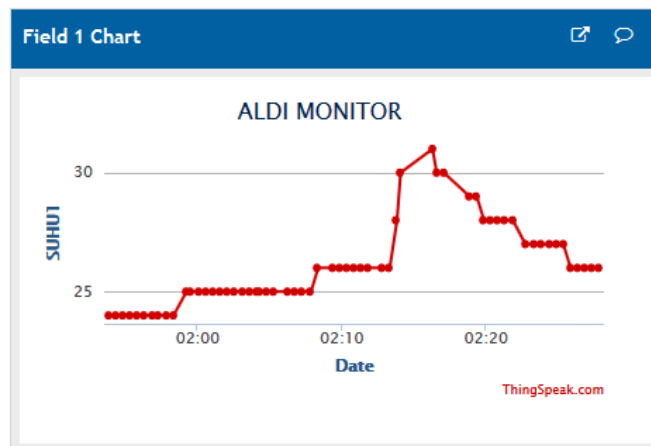
3.3 Pengujian Pengiriman Data Modul DHT11 dan DHT 2 Melalui IoT

Dari modul DHT11 didapat dua data yaitu data perubahan suhu dan perubahan kelembaban udara. Cara melakukan pengujian DHT11 adalah dengan memberikan efek panas menggunakan *hairdryer* yang diatur dengan menggunakan kecepatan rendah dengan jarak 10 cm. Sensor dipanaskan dengan sampai dengan 30° C kemudian *hairdryer* dimatikan, perubahan suhu dapat diunduh dan diperlihatkan pada Tabel 2.

Tabel 2. Data Perubahan Suhu dari DHT11 Melalui IoT

created_at	entry_id	field1
2020-01-30 18:11:05 UTC	266	30
2020-01-30 18:11:35 UTC	267	29
2020-01-30 18:12:05 UTC	268	28
2020-01-30 18:12:35 UTC	269	28
2020-01-30 18:13:13 UTC	270	28
2020-01-30 18:14:31 UTC	271	27
2020-01-30 18:15:01 UTC	272	27
2020-01-30 18:15:30 UTC	273	27
2020-01-30 18:16:09 UTC	274	27
2020-01-30 18:16:39 UTC	275	27
2020-01-30 18:17:09 UTC	276	27
2020-01-30 18:17:47 UTC	277	27
2020-01-30 18:18:26 UTC	278	26

Dari Tabel 2 dapat dilihat kolom *created_at* adalah waktu dimana data diambil, kolom *entry_id* adalah nomor data, dan *field1* adalah data nilai suhu yang didapat. Pada Tabel 2 dapat diketahui bahwa perubahan temperature yang dideteksi sensor DHT11 dapat diperlihatkan melalui IoT dan berfungsi dengan baik ketika dilakukan simulasi pemanasan sensor menggunakan *hairdryer*. Suhu maksimal diberikan hingga mencapai 30° C pada pukul 18:11:05 UTC, jika dilakukan konversi menggunakan waktu Jakarta adalah pukul 01:11:05 WIB pada nomor data 266. Dari tabel dapat dilihat untuk menurunkan suhu dari 30° C menuju 26° C membutuhkan waktu 7 menit 21 detik. Gambar 9 memperlihatkan grafik perubahan suhu terhadap waktu yang dideteksi DHT11 menggunakan *Thingspeak*.



Gambar 9. Tampilan Grafik *Thingspeak* Suhu1 dari DHT11

Pada Tabel 3 diperlihatkan perubahan kelembaban udara yang dideteksi DHT11 terhadap waktu yang diunduh melalui IoT. Dari Tabel 3 dapat dilihat *field2* adalah data kelembaban udara yang disimpan pada server. Dari Tabel 3 diketahui bahwa perubahan kelembaban udara dari 44 % ke 62 % membutuhkan waktu 7 menit 21 detik.

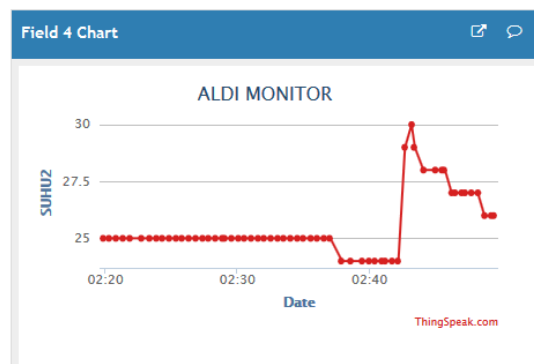
Tabel 3. Data Perubahan Kelembaban Udara dari DHT11 Melalui IoT

created_at	entry_id	field2
2020-01-30 18:11:05 UTC	266	44
2020-01-30 18:11:35 UTC	267	46
2020-01-30 18:12:05 UTC	268	46
2020-01-30 18:12:35 UTC	269	48
2020-01-30 18:13:13 UTC	270	49
2020-01-30 18:14:31 UTC	271	51
2020-01-30 18:15:01 UTC	272	54
2020-01-30 18:15:30 UTC	273	55
2020-01-30 18:16:09 UTC	274	57
2020-01-30 18:16:39 UTC	275	58
2020-01-30 18:17:09 UTC	276	59
2020-01-30 18:17:47 UTC	277	61
2020-01-30 18:18:26 UTC	278	62

Pada Tabel 4 dapat dilihat data perubahan suhu dari DHT11 2 melalui IoT. Field4 pada Tabel 5 menunjukkan perubahan suhu yang dideteksi DHT 11 2. Dari Tabel 4 diketahui bahwa perubahan suhu dari 30° C menuju 26° C membutuhkan waktu 5 menit 31 detik. Respon DHT11 2 terhadap perubahan suhu lebih cepat dari DHT11 1. Gambar 10 memperlihatkan grafik perubahan suhu terhadap waktu yang dideteksi DHT1 2 menggunakan *Thingspeak*.

Tabel 4. Data Perubahan Suhu dari DHT12 Melalui IoT

created_at	entry_id	field4
2020-01-30 19:43:11 UTC	437	30
2020-01-30 19:43:26 UTC	438	29
2020-01-30 19:44:05 UTC	439	28
2020-01-30 19:44:58 UTC	440	28
2020-01-30 19:45:28 UTC	441	28
2020-01-30 19:45:43 UTC	442	28
2020-01-30 19:46:13 UTC	443	27
2020-01-30 19:46:28 UTC	444	27
2020-01-30 19:46:58 UTC	445	27
2020-01-30 19:47:13 UTC	446	27
2020-01-30 19:47:42 UTC	447	27
2020-01-30 19:48:12 UTC	448	27
2020-01-30 19:48:42 UTC	449	26



Gambar 10. Tampilan Grafik *Thingspeak* Suhu2 dari DHT12

Pada Tabel 5 diperlihatkan perubahan kelembaban udara yang dideteksi DHT12 terhadap waktu yang diunduh melalui IoT. Dari Tabel 5 dapat dilihat field5 adalah data kelembaban udara yang

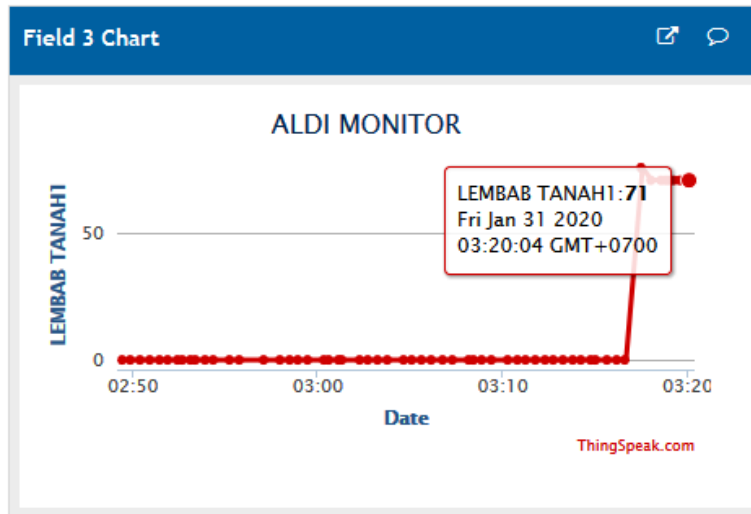
disimpan pada server. Dari Tabel 5 diketahui bahwa perubahan kelembaban udara dari 44 % ke 52 % membutuhkan waktu 5 menit 31 detik.

Tabel 5. Data Perubahan Kelembaban Udara dari DHT12 Melalui IoT

created_at	entry_id	field5
2020-01-30 19:43:11 UTC	437	44
2020-01-30 19:43:26 UTC	438	45
2020-01-30 19:44:05 UTC	439	46
2020-01-30 19:44:58 UTC	440	49
2020-01-30 19:45:28 UTC	441	50
2020-01-30 19:45:43 UTC	442	50
2020-01-30 19:46:13 UTC	443	50
2020-01-30 19:46:28 UTC	444	50
2020-01-30 19:46:58 UTC	445	51
2020-01-30 19:47:13 UTC	446	50
2020-01-30 19:47:42 UTC	447	51
2020-01-30 19:48:12 UTC	448	51
2020-01-30 19:48:42 UTC	449	52

3.4 Pengujian Pengiriman Data Perubahan Kelembaban Tanah Melalui IoT

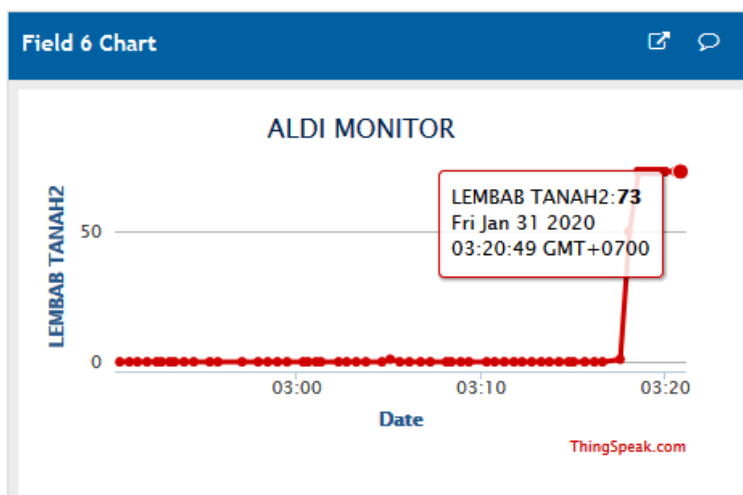
Untuk melakukan pengujian terhadap sensor *soil moisture*1 ataupun 2 adalah dengan menggunakan media serbuk gergaji dengan kondisi mula-mula tanpa menggunakan air dan kemudian diberikan air. Hasil perubahan data kelembaban tanah dari sensor *soil moisture* 1 diperlihatkan pada Gambar 11.



Gambar 11. Perubahan Kelembaban Tanah pada *Soil Moisture* 1

Pada pukul 02:50 kelembaban tanah dengan media serbuk gergaji tidak menggunakan air sehingga kelembabannya adalah 0%RH. Sekitar pukul 03:17 media tanam serbuk gergaji ditambahkan air sehingga nilainya kelembabannya menjadi 71%RH pada pukul 03.20. Kelembaban tanah

dalam



dari 0% ke 71% perubahannya dideteksi waktu sekitar 3 menit.

Gambar 12. Perubahan Kelembaban Tanah pada *Soil Moisture 2*

Pada Gambar 12 dapat dilihat pada pukul 03:00 sensor *soil moisture 2* dengan media tanam serbuk gergaji tidak menggunakan air sehingga kelembabannya adalah 0%RH. Sekitar pukul 03:17 media tanam serbuk gergaji ditambahkan air sehingga nilainya menjadi 73% RH pada pukul 03.20. Dalam waktu 3 menit *soil moisture 2* dapat mendeteksi perubahan kelembaban tanah dari 0% ke 73 %. Selisih kelembaban yang dideteksi *soil moisture1* terhadap *soil moisture2* adalah 2%.

4. KESIMPULAN

Dari hasil pengujian kelembaban ruangan dan suhu dijaga tetap 26° C dapat diketahui bahwa pada saat kelembaban ruangan berada di level 45% sampai dengan 80%RH pompa air menyala untuk mengaktifkan pengkabutan sehingga kelembaban akan naik menuju targetnya. Pada saat kelembaban ruang mencapai 85% maka pompa air akan mati karena kelembaban sudah sesuai perancangan. Pada saat kelembaban ruangan mencapai 95% maka kipas angin akan menyala untuk menurunkan kelembaban ruangan sehingga dapat dijaga kelembaban ruangan dari 80 sampai 90%.

Dari hasil pengujian suhu ruangan dan kelembaban dijaga tetap 45% dapat diketahui bahwa saat suhu dibawah 26° C , maka lampu akan menyala sampai suhu diatas 30° C lampu akan mati sedangkan kipas angin akan menyala untuk menjaga suhu di rentang 26 sampai 30° C.

Dari pengujian respon DHT11 1 diketahui untuk menurunkan suhu dari 30° C menuju 26° C (perubahan 5° C) membutuhkan waktu 7 menit 21 detik sedangkan untuk respon kelembaban udara dalam waktu 7 menit 21 detik dapat mendeteksi perubahan dari 44 % ke 62 % (perubahan 18%). Dari pengujian respon DHT11 2 diketahui untuk menurunkan suhu dari 30° C menuju 26° C (perubahan 5° C) membutuhkan waktu 5 menit 31 detik sedangkan untuk respon kelembaban udara dalam waktu 5 menit 31 detik dapat mendeteksi perubahan dari 44% ke 52% (perubahan 8%).

Dari hasil pengujian respon *soil moisture 1* diketahui untuk mendeteksi kelembaban tanah dari 0% ke 71% dibutuhkan waktu sekitar 3 menit, sedangkan respon *soil moisture 2* mendeteksi kelembaban tanah dari tanah dari 0% ke 73 % dalam waktu sekitar 3 menit.

REFERENSI

- [1] Anonim, "Jamur Tiram", 10 Jun 2019, [Online]. Tersedia :https://id.wikipedia.org/wiki/Jamur_tiram [Diakses 8 Des 2019].
- [2] Anonim, "Cara Budidaya Jamur Tiram Untuk Pemula", 12 Okt 2015, [Online]. Tersedia :<https://satujam.com/budidaya-jamur-tiram/> [Diakses 9 Des 2019].
- [3] Andika A, Soewarto, "Model Pengaturan Suhu dan Kelembapan Pada Ruang Jamur Tiram Menggunakan DHT11 dan Mikrokontroler ATmega328", Skripsi, Universitas Pakuan, 2018.
- [4] Telaumbanua, M., Purwanto, B., & Sutiarso, L. *Rancangbangun Aktuator Pengendali Iklim Mikro di dalam Greenhouse untuk Pertumbuhan Tanaman Sawi (Brassica rapa var.parachinensis L.)*. 2014. agriTECH. Vol. 34, No. 2 Hal: 213-222.
- [5] Abdullah, Masthura. *Sistem Pemberian Nutrisi dan Penyiraman Tanaman Otomatis Berdasarkan Real Time Clock dan Tingkat Kelembaban Tanah Berbasis Mikrokontroler Atmega32*.2018. FISITEK: Jurnal Ilmu Fisika dan Teknologi, Vol. 2, No. 2, Hal: 33 – 4.

- [6] Sri Waluyo dkk. *Pengendalian Temperatur dan Kelembaban dalam Kumbung Jamur Tiram (Pleurotus sp) Secara Otomatis Berbasis Mikrokontroler*. 2018. agriTECH. Vol. 38, No. 3 Hal: 282-288.
- [7] DHT11 Humidity and Temperature Sensor, Tersedia dari : <https://datasheet4u.com/datasheet-parts/DHT11-datasheet.php?id=785590>, [URL dikunjungi pada 2 Agustus 2020].
- [8] Mengenal Arduino : Pengertian, Sejarah, Kelebihan dan Jenis- Jenisnya. Tersedia dari : <https://www.andalanelektro.id/2018/08/mengenal-arduino.html>, [URL dikunjungi pada 6 Agustus 2020].
- [9] ESP8266EX Datasheet, Tersedia dari : https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf, [URL dikunjungi pada 6 Agustus 2020].
- [10] Sapta Aji, “IOT Dengan Arduino dan Thingspeak”, 21 November 2016, [Online]. Tersedia: <http://saptaji.com/2016/11/21/iot-dengan-arduino-dan-thingspeak/>[Diakses 10 Des 2019].