

## Arsitektur Serverless Internet of Things untuk Pencatatan Data Perangkat Microclimate Adjuster di Fasilitas Riset PRITA

I Komang Agus Ady Aryanto<sup>1</sup>, I Putu Widia Prasetya<sup>2</sup>, I Kadek Arta Wiguna<sup>3</sup>, I Made Pande Darma Yuda<sup>4</sup>, I Nyoman Kusuma Wardana<sup>5</sup>

<sup>1</sup>Program Studi Teknologi Informasi, Institut Teknologi dan Bisnis STIKOM Bali, Bali

<sup>2,3,4</sup>Program Studi Teknologi Informasi, Universitas Pendidikan Nasional, Bali

<sup>5</sup>Jurusan Teknik Elektro, Politeknik Negeri Bali, Bali

### ABSTRACT

Internet of Things (IoT) is an integral part of the solutions needed to implement the 4.0 industrial revolutions, including in the field of precision agriculture in PRITA greenhouses. To implement IoT technology, of course, a fairly complex infrastructure is needed, given the nature of IoT technology, which is divergent and consists of various sub-systems. In this study, we tried to develop a microclimate adjuster device that was built on previous research, by adding a serverless IoT architecture to get an integrated environmental data recording feature based on cloud computing services. The serverless architecture offered in this study is intended to eliminate all the complexities that exist on the infrastructure side, as well as minimize the effort and cost to run the system. Based on the test results by applying directly to the prototype device, it is known that the architecture offered is capable of running well. Devices can connect to IoT Core properly via SSL encrypted MQTT protocol, device data can be stored in NoSQL databases, and the OTA Update process can now be done from anywhere via the Internet.

**Keywords:** Internet of Things, serverless, microclimate adjuster, MQTT, NoSQL

### ABSTRAK

Internet of Things (IoT) menjadi bagian tak terpisahkan dari solusi yang diperlukan untuk mengimplementasikan revolusi industri 4.0, termasuk pada bidang pertanian presisi di rumah kaca PRITA. Untuk mengimplementasikan teknologi IoT, tentunya diperlukan infrastruktur yang cukup rumit, mengingat sifat dari teknologi IoT adalah divergen dan terdiri dari berbagai macam sub sistem. Pada penelitian ini, peneliti mencoba untuk mengembangkan perangkat microclimate adjuster yang dibangun pada penelitian sebelumnya, dengan menambahkan arsitektur serverless IoT untuk mendapatkan fitur pencatatan data lingkungan yang terintegrasi berbasis layanan cloud computing. Arsitektur serverless yang ditawarkan pada penelitian ini ditujukan untuk menghilangkan segala kerumitan yang ada pada sisi infrastruktur, serta meminimalkan usaha dan biaya untuk menjalankan sistem. Berdasarkan hasil pengujian dengan menerapkan secara langsung pada perangkat purwarupa, diketahui bahwa arsitektur yang ditawarkan mampu berjalan dengan baik. Perangkat dapat terhubung ke IoT Core dengan baik melalui protokol MQTT yang terenkripsi SSL, data perangkat mampu disimpan pada database NoSQL, dan proses OTA Update kini dapat dijalankan dari mana saja melalui Internet.

**Kata kunci:** Internet of Things, serverless, microclimate adjuster, MQTT, NoSQL

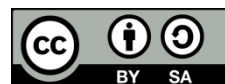
### Info Artikel

Diterima Redaksi : 03-12-2020

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*

Selesai Revisi : 10-12-2020

Diterbitkan Online : 31-12-2020



### Penulis Korespondensi:

I Komang Agus Ady Aryanto  
Program Studi Teknologi Informasi,  
Institut Teknologi dan Bisnis STIKOM Bali,  
Jl. Raya Puputan No.86, Dangin Puri Klod, Kec. Denpasar Tim., Kota Denpasar, Bali 80234  
Email: agus\_ady@stikom-bali.ac.id

## 1. PENDAHULUAN

PRITA (precision agriculture laboratory) merupakan fasilitas riset dalam bidang pertanian presisi. PRITA berada di bawah Program Studi Teknologi Informasi, Fakultas Teknik dan Informatika, Universitas Pendidikan Nasional. PRITA memiliki satu unit rumah kaca yang di dalamnya berisi sistem pertanian berteknologi *hydroponic fragmented media beds* dan *aeroponic vertical farming*. Menurut [1, 2], metode *fragmented media bed* memiliki keunggulan utama dapat meminimalisir penyebaran penyakit dan apabila dipadukan dengan *aeroponic vertical farming* maka didapatkan keunggulan lain berupa penghematan tempat, air, dan hasil yang lebih optimal berkat proses aerasi pada bagian akar tanaman.

Sebagai rumah kaca yang terletak di area topis, tentunya diperlukan suatu sistem terotomatisasi untuk melakukan pengendalian kondisi micro-climate. Sehingga, tanaman yang tumbuh di dalam rumah kaca dapat berkembang dengan baik. Pada penelitian sebelumnya [3], peneliti telah membangun purwarupa perangkat *micro-climate adjuster* berkemampuan *remote monitoring*. Selengkapnya, berdasarkan hasil pengujian dan observasi purwarupa, dapat diketahui purwarupa berjalan sebagaimana mestinya. Pada fitur *remote monitoring*, digunakan teknologi *Internet of Things* (IoT) dengan protokol MQTT. Namun, cakupan penelitian tersebut hanya sebatas implementasi bagian perangkat purwarupa *micro-climate adjuster*. Sedangkan untuk dapat menjalankan sistem pada implementasi riil, diperlukan infrastruktur untuk *Internet of Things* yang memungkinkan perangkat dapat merekam data dan berkomunikasi dengan pengguna [4, 11].

Untuk mendapatkan arsitektur IoT yang baik, perlu dilakukan analisis kebutuhan dan karakteristik sistem. Terkait infrastruktur, komponen dasarnya adalah konektivitas dan IoT Core [5]. Konektivitas berkaitan dengan fleksibilitas sistem, penggunaan daya, dan akan menentukan bagaimana struktur dan desain keseluruhan perangkat keras dan perangkat lunak yang digunakan [6]. Sedangkan IoT Core tentunya berkaitan dengan infrastruktur server yang berkorelasi dengan biaya dan pilihan layanan [7]. Apabila dikaitkan dengan kebutuhan umum untuk tujuan implementasi pada rumah kaca khususnya di fasilitas riset PRITA, maka diperlukan konektivitas dan infrastruktur yang mampu mengakomodir fungsi pengendalian dan monitoring kondisi *microclimate*. Beberapa penelitian telah dilakukan sebelumnya berkaitan dengan implementasi *serverless computing* [8,9,10]. Berdasarkan penelitian tersebut, diketahui bahwa arsitektur *serverless* merupakan pilihan yang paling menjanjikan dari segi efektifitas, efisiensi dan manajemen risiko.

Komputasi *serverless* muncul sebagai paradigma baru [8]. Pada umumnya, jenis komputasi ini dilihat sebagai solusi yang menarik untuk penerapan aplikasi berbasis cloud. Salah satu penyebabnya adalah terjadinya pergeseran arsitektur aplikasi berskala korporat baru-baru ini ke framework *container* dan *microservice* [9, 10]. Dengan menggunakan framework *container*, aplikasi berskala besar dengan tingkat kompleksitas tinggi dapat dipecah-pecah menjadi *microservice*. Aplikasi yang dipecah menjadi *microservice* dalam sebuah *container* selanjutnya dapat dilakukan perawatan, penyesuaian, dan pengembangan yang lebih fleksibel, tanpa harus mendisrupsi keseluruhan layanan. Berangkat dari keuntungan menjanjikan yang ditawarkan oleh komputasi *serverless*, serta beberapa penelitian yang telah dilakukan sebelumnya berkaitan dengan penerapan *serverless computing* untuk tujuan pembangunan aplikasi berbasis teknologi *Internet of Things*, maka pada penelitian ini, peneliti mencoba untuk menerapkan arsitektur *serverless* pada perangkat *microclimate adjuster* yang dipasang pada rumah kaca fasilitas riset PRITA.

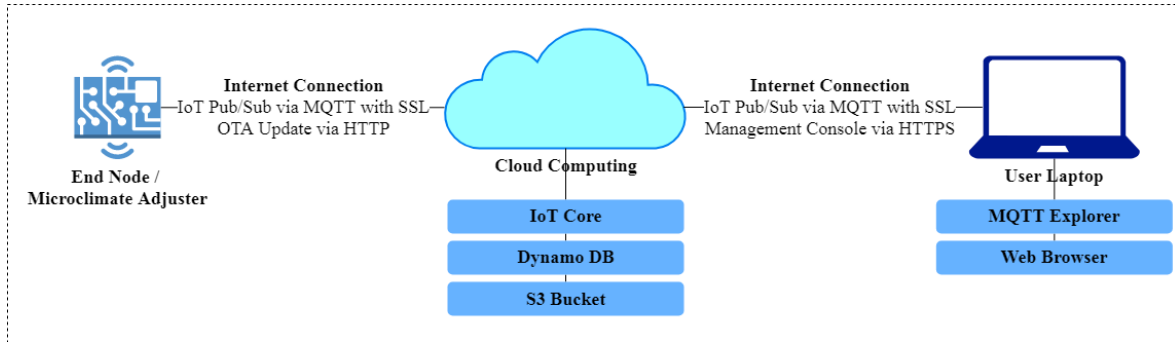
Arsitektur *serverless* yang ditawarkan pada penelitian ini mencakup desain dan implementasi penerapan layanan *cloud computing*, yang secara umum terdiri dari infrastruktur untuk IoT Core dan *database*. Layanan IoT Core berperan sebagai infrastruktur yang mengatur bagaimana perangkat IoT terhubung, termasuk juga pengguna. Kemudian, pada bagian *database*, berperan dalam mengatur bagaimana data disimpan. Untuk mendapatkan wawasan mengenai performa arsitektur yang dibangun, peneliti menerapkan arsitektur ini pada perangkat purwarupa *microclimate adjuster* sebagai *proof-of-concept*. Performa sistem selanjutnya diuji dengan metode *black box unit testing* dan dilakukan analisis fungsionalitas, keandalan, dan performa perangkat dan sistem secara keseluruhan. Pengujian purwarupa dilakukan dengan memasang secara langsung (*field test*) di rumah kaca PRITA selama satu minggu dengan kondisi perangkat terhubung ke layanan IoT.

## 2. METODE PENELITIAN

### 2.1 Desain Arsitektur

Penelitian ini adalah penelitian pengembangan dari penelitian sebelumnya [3], perangkat purwarupa telah dibangun dan telah memiliki *application programming interface* (API) yang mendukung fitur konektivitas IoT. Untuk desain perangkat, keras, perangkat lunak dan lapisan koneksi IEEE 802.11 telah

terimplementasi dengan baik pada penelitian sebelumnya. Sehingga, pada penelitian ini dibahas lebih detail mengenai arsitektur serverless IoT secara keseluruhan yang memungkinkan terciptanya satu sistem utuh terintegrasi untuk mengatur perangkat microclimate adjuster. Baik dari sisi perangkat, hingga infrastruktur dasar dari teknologi IoT. Gambar 1 memperlihatkan bagaimana susunan besar keseluruhan sistem serverless IoT yang ditawarkan pada penelitian ini. Perangkat microclimate adjuster yang terhubung ke sistem serverless IoT bisa diistilahkan sebagai sebuah end node atau things. Pada bagian things ini, untuk dapat terhubung ke serverless IoT, hanya diperlukan beberapa pengaturan yaitu kredensial untuk terhubung ke jaringan WLAN (Wireless Local Area Network) dan kredensial untuk terkoneksi ke bagian IoT Core.



Gambar 1. Gambaran Umum Arsitektur Serverless Internet of Things

Untuk kredensial jaringan WLAN, perlu disetel informasi autentikasi WPA2/PSK berupa SSID dan sandi jaringan ke konfigurasi perangkat. Untuk kredensial IoT Core diperlukan kunci enkripsi SSL yang terdiri dari *client certificate*, *server certificate* dan *private key*. Ketiga kunci ini didapatkan dari *management console* IoT Core saat proses pendaftaran atau penambahan perangkat. Selanjutnya, ketiga file kunci ini diunduh dan ditulis ke memori flash perangkat dan dienkripsi. Pada antarmuka *management console* IoT Core, selain fitur manajemen perangkat, terdapat pula fitur lain seperti pengaturan polisi dan aturan (*rules*). Pada bagian polisi ini ditentukan pengaturan polisi seperti perangkat yang mana saja yang boleh menggunakan sumber daya layanan berdasarkan berbagai macam parameter. Pada penelitian ini, polisi diset ke mode minimum untuk menyederhanakan pengembangan. Perangkat yang telah memiliki kunci SSL yang valid dapat terhubung ke layanan dan melakukan kegiatan publish dan subscribe ke topik mana pun tanpa ada pembatasan. Format topik yang digunakan cukup sederhana dan daftar topik yang digunakan pada arsitektur ini terlihat pada Tabel 1. Pembentukan format topik mempertimbangkan skenario jumlah perangkat yang terhubung lebih dari satu. Setiap topik ditambahkan jenis aksi pada bagian akhir, hal ini sangat penting untuk menghilangkan potensi terjadinya *message loop* yang tidak sengaja terjadi saat melakukan pemanggilan *request response* melalui fungsi *IoT Incoming Message Callback* di sisi *end node*.

Tabel 1. Format Topik IoT untuk Publish Subscribe

No.	Nama Topik Subscribe	Fungsi
1	/ALL/sub	Mensubscribe semua pesan dari setiap things
2	/[GROUP]/sub	Mensubscribe semua pesan hanya dari things yang berada di grup yang sama
3	/[GROUP]/[NAMA_UNIK_PERANGKAT]/sub	Mensubscribe pesan yang hanya ditujukan ke nama perangkat
4	/error	Mensubscribe pesan error dari IoT Core dan Dynamo DB
No.	Nama Topik Publish	Fungsi
1	/ALL/pub	Mempublikasikan pesan ke semua things
2	/[GROUP]/pub	Mempublikasikan pesan ke things pada grup yang sama
3	/[GROUP]/[NAMA_UNIK_PERANGKAT]/pub	Mempublikasikan pesan ke things dengan nama spesifik

Selain IoT Core, juga digunakan layanan Dynamo DB. Dynamo DB digunakan untuk melakukan penyimpanan pesan yang dikirim oleh perangkat. Untuk dapat mengambil data perangkat yang diatur oleh broker pada layanan IoT Core, maka Dynamo DB perlu diberikan hak akses sebagai service dengan ijin membaca topik IoT (IoT Topic Read & Query). Untuk memungkinkan hal ini, Dynamo DB dibuatkan semacam access rules untuk dapat membaca topik melalui layanan IAM (*Identity Access management*). Setelah Dynamo DB memiliki access rules, selanjutnya dibuatkan pemetaan aliran data dari IoT Core ke Dynamo DB. Pada proses pemetaan ini disetel topik mana saja yang akan dialirkan ke DynamoDB dan ditetapkan nama tabel pada Dynamo DB yang digunakan untuk menyimpan data. Dynamo DB merupakan database NoSQL, sehingga tidak terdapat struktur baku. Untuk itu, pada proses pemetaan juga ditetapkan field ID sebagai pembeda antara satu record dengan yang lain dan field DT sebagai penyortir berdasarkan waktu. Untuk field lainnya, secara otomatis akan terbentuk saat terdapat data masuk. Data yang masuk dari IoT Core berbentuk JSON dengan struktur label: data, sehingga setiap terdapat label baru maka secara otomatis pula akan dibuatkan kolom baru sesuai dengan nama label pada tabel Dynamo DB.

Selain untuk tujuan pencatatan data dan manajemen perangkat, tentu diperlukan suatu mekanisme untuk melakukan *maintenance* atau perawatan pada perangkat, seperti misalnya pembaruan perangkat lunak atau firmware. Untuk itu, pada arsitektur ini digunakan metode OTA Update melalui Internet dengan memanfaatkan layanan S3 Bucket. Konsepnya cukup sederhana, pengembang atau pengguna yang berniat untuk melakukan pembaruan cukup mengunggah file firmware terbaru ke folder di S3 Bucket. Selanjutnya, pengembang cukup mengirimkan instruksi update melalui IoT Topik, baik ke semua perangkat maupun ke perangkat tertentu. Segera setelah perangkat menerima instruksi update dari pesan subscribe, maka proses update akan dimulai. Perangkat mengunduh file firmware melalui HTTP dan melakukan *self firmware flashing*.

## 2.2 Teknik Pengujian dan Validasi

Pengujian dan validasi dilakukan secara langsung pada perangkat purwarupa yang telah terpasang di Lab. PRITA. Untuk instalasi perangkat keras tidak terdapat perubahan dengan penelitian sebelumnya, hanya saja pada sisi perangkat lunak telah ditambahkan konfigurasi dan penyesuaian untuk penggunaan infrastruktur serverless IoT menggunakan layanan dari vendor Amazon Web Service. Sehingga, pada penelitian ini yang diobservasi adalah bagaimana performa proses sistem bekerja pada infrastruktur serverless. Untuk detail parameter yang diobservasi beserta validasinya dapat dilihat pada Tabel 2.

Tabel 2. Parameter dan Validasi yang Diujikan dan Diobservasi

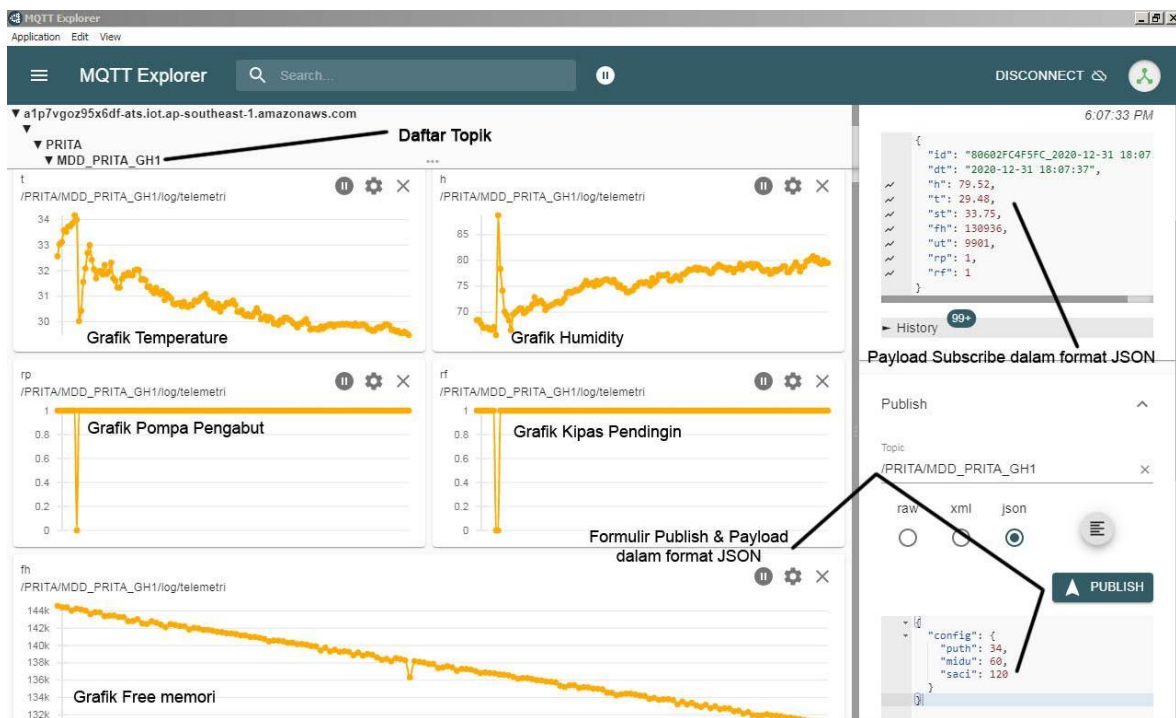
No.	Parameter	Validasi
1	Koneksi ke IoT Core menggunakan MQTT dengan SSL	Pengujian dianggap berhasil apabila perangkat berhasil terkoneksi ke IoT Core. Ditandai dengan lampu indikator perangkat berubah warna menjadi hijau.
2	Proses Publish / Subscribe pada setiap topik	Pengujian dianggap berhasil apabila proses subscribe dan publish pesan berhasil dilakukan. Ditandai dengan ACK dari perangkat yang memastikan data pesan terkirim melalui MQTT Explorer.
3	Integrasi data flow dari IoT Core ke Dynamo DB	Pengujian dianggap berhasil apabila data berhasil masuk ke tabel Dynamo DB dan tidak ada pesan error yang dipublish ke topik /error
4	OTA Update via HTTP dari S3 Bucket	Pengujian dianggap berhasil apabila perangkat dapat melakukan OTA Update. Ditandai dengan tidak ada log error pada perangkat.

## 3. HASIL DAN PEMBAHASAN

*Arsitektur Serverless IoT untuk Pencatatan Data Microclimate Adjuster di PRITA (I Komang Agus Ady Aryanto)*

Berdasarkan hasil pengujian purwarupa, diketahui bahwa perangkat berhasil terhubung ke IoT Core dengan baik dan lancar. Tidak ditemukan kendala apapun untuk kontroler ESP32 dari segi ketersediaan sumber daya untuk menjalankan *stack protocol* TCP/IP yang menggunakan enkripsi SSL. Ketersediaan memori tercatat stabil dan tidak terlalu banyak berbeda dengan pengujian pada penelitian sebelumnya yang menggunakan IoT Broker berbasis layanan *trial* tanpa enkripsi, yaitu berada dikisaran 125 hingga 144 KB. Untuk proses subscribe dan publish juga tidak ditemukan kendala apapun. Perangkat mampu menjalankan routine *IoT Incoming Message Callback* dengan lancar, dibuktikan dengan delay antara request / response dari pesan yang dikirim dan diterima kurang dari satu detik.

Pesan yang berisi konfigurasi perangkat juga mampu diterima dengan baik dan tersimpan dengan baik di perangkat. Gambar 2 memperlihatkan antarmuka MQTT Explorer yang dilengkapi dengan visualisasi data masuk serta formulir untuk mempublikasikan data berupa payload JSON ke IoT Core. Melalui antarmuka ini, pengguna dapat melakukan query ke perangkat, maupun melakukan perubahan konfigurasi termasuk mengirim perintah OTA Update. Pada pengujian OTA Update diketahui proses update berjalan sebagaimana mestinya, namun tetap dengan catatan bahwa memori bebas harus tidak kurang dari 130 KB.



Gambar 2. Antarmuka MQTT Explorer untuk Melakukan Observasi dan Mengirimkan Instruksi ke Perangkat

Untuk integrasi antara IoT Core dan Dynamo DB setelah dilakukan observasi, diketahui tidak terdapat kendala apapun. Integrasi yang dilakukan berhasil dan data yang masuk dapat tersimpan di Dynamo DB. Seperti terlihat pada Gambar 3, terdapat 9 kolom yang terdiri dari id, dt, fh, rf, rp, ut, st, h, dan t, masing-masing berisi data id pesan, timestamp pesan dikirim, *free heap*, status kipas, status pompa, *uptime* dalam detik, temperatur sistem, *humidity*, dan temperatur lingkungan. Pesan dikirim dalam interval 2 menit sekali, dan berdasarkan hasil pengamatan data tercatat dengan baik. Pada kolom *free heap* atau sisa memori RAM, terlihat penggunaan memori yang tidak begitu berbeda dengan hasil observasi penelitian sebelumnya. Berdasarkan itu, dapat diketahui bahwa arsitektur serverless selain membebaskan pengguna dari kerumitan, risiko dan tingginya biaya untuk menjalankan server Internet of Things sendiri, juga cukup *seamless* dalam hal penggunaan sumber daya perangkat dan integrasi dari infrastruktur sebelumnya ke model serverless.

id	dt	fh	rf	rp	ut	st	h	t
80602FC4F5FC_2020-	2020-1...	129164	1	1	11161	33.5	80.31	29.3
80602FC4F5FC_2020-	2020-1...	129432	1	1	10981	33.5	79.95	29.4
80602FC4F5FC_2020-	2020-1...	130408	1	1	10321	33.75	80.31	29.44
80602FC4F5FC_2020-	2020-1...	130512	1	1	10261	33.75	79.85	29.41
80602FC4F5FC_2020-	2020-1...	130676	1	1	10081	33.75	81.07	29.4
80602FC4F5FC_2020-	2020-1...	131236	1	1	9721	33.75	79.31	29.57
80602FC4F5FC_2020-	2020-1...	131820	1	1	9301	34	79.83	29.89
80602FC4F5FC_2020-	2020-1...	133180	1	1	8101	34	77.92	29.89
80602FC4F5FC_2020-	2020-1...	134180	1	1	7561	34	78.88	29.82
80602FC4F5FC_2020-	2020-1...	135088	1	1	6721	34.25	78.44	29.89

Gambar 3. Tabel Dynamo DB Berisi Catatan Data Perangkat

#### 4. KESIMPULAN

Peneliti mencoba untuk mengembangkan hasil implementasi purwarupa microclimate adjuster pada penelitian sebelumnya, dengan menambahkan arsitektur berbasis layanan cloud untuk memungkinkan fitur pencatatan data dan manajemen perangkat IoT yang terintegrasi melalui layanan *serverless computing*. Dengan menerapkan layanan serverless, tentunya proses manajemen perangkat dan pencatatan data dari perangkat microclimate adjuster di Lab. PRITA dapat dipermudah dengan menghilangkan kerumitan pada infrastruktur IoT yang dilengkapi dengan database. Berdasarkan hasil pengujian purwarupa, diketahui bahwa perangkat dapat beroperasi dengan baik sesuai dengan yang diharapkan. Proses *publish* dan *subscribe* yang berjalan secara aman melalui MQTT SSL dapat terjadi dengan baik melalui layanan IoT Core, serta fitur OTA Update juga berjalan lancar melalui file hosting S3 Bucket. Data lingkungan yang dikirim perangkat juga tersimpan dengan baik pada database NoSQL Dynamo DB. Untuk pengembangan selanjutnya, layanan serverless IoT dapat dikembangkan untuk mencakup platform data analytic, yaitu, data yang telah berhasil terekam dapat dianalisis untuk mendapatkan model yang dapat digunakan untuk tujuan mengembangkan sistem kontrol dan prediksi. Sehingga, proses pengendalian kondisi lingkungan di rumah kaca PRITA dapat terjadi dengan lebih baik dan lebih efisien.

#### UCAPAN TERIMAKASIH

Peneliti menyampaikan ucapan terimakasih kepada Bapak Rektor beserta jajaran pengampu kepentingan di Fakultas Teknik dan Informatika khususnya Program studi Teknologi Informasi Universitas Pendidikan Nasional atas bantuan yang diberikan untuk membangun Undiknas Precision Agriculture Laboratory (PRITA) Research Facility. Sehingga, dimungkinkan untuk peneliti melakukan pengembangan ilmu pengetahuan dan teknologi di bidang pertanian presisi.

#### DAFTAR RUJUKAN

- [1] Halveland, J. (2020). Design of a Shallow-Aero Ebb and Flow Hydroponics System and Associated Educational Module for Tri Cycle Farms. Biological and Agricultural Engineering Undergraduate Honors Theses Retrieved from <https://scholarworks.uark.edu/baeguht/76>.
- [2] Eldridge, B.M., Manzoni, L.R., Graham, C.A., Rodgers, B., Farmer, J.R. and Dodd, A.N. (2020), Getting to the roots of aeroponic indoor farming. *New Phytol*, 228: 1183-1192. <https://doi.org/10.1111/nph.16780>.
- [3] Wardana, I.N.K., et. al. (2020). Purwarupa Perangkat Microclimate Adjuster Berkemampuan Remote Monitoring di Fasilitas Riset PRITA. *TIERS Information Technology Journal*. Vol. 1. No. 2. pp: 1-9.
- [4] Salam, A., & Shah, S. (2019). Urban Underground Infrastructure Monitoring IoT: The Path Loss Analysis. 2019 IEEE 5th World Forum on Internet of Things (WF-IoT). doi:10.1109/wf-iot.2019.8767358.
- [5] Ding, J., Nemati, M., Ranaweera, C., & Choi, J. (2020). IoT Connectivity Technologies and Applications: A Survey. *IEEE Access*, 1–1. doi:10.1109/access.2020.2985932.
- [6] Wardana, I Nyoman Kusuma, Ngakan Nyoman Kutha Krisnawijaya, and I Wayan Aditya Suranata. 2018. "Sub-1 GHz Wireless Nodes Performance Evaluation for Intelligent Greenhouse System." *TELKOMNIKA (Telecommunication Computing Electronics and Control)* 16 (6): 2888. <https://doi.org/10.12928/telkomnika.v16i6.11556>.
- [7] Dupont, C., Bures, T., Sheikhalishahi, M., Pham, C., & Rahim, A. (2017). Low-cost IoT, Big Data, and Cloud Platform for Developing Countries. *Lecture Notes in Computer Science*, 285–299. doi:10.1007/978-3-319-68066-8\_23.

- 
- [8] Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., ... Suter, P. (2017). Serverless Computing: Current Trends and Open Problems. *Research Advances in Cloud Computing*, 1–20. doi:10.1007/978-981-10-5026-8\_1.
  - [9] Sewak, M., & Singh, S. (2018). Winning in the Era of Serverless Computing and Function as a Service. 2018 3rd International Conference for Convergence in Technology (I2CT). doi:10.1109/i2ct.2018.8529465.
  - [10] Stigler, M. (2017). Understanding Serverless Computing. *Beginning Serverless Computing*, 1–14. doi:10.1007/978-1-4842-3084-8\_1.
  - [11] Aafreen, R., Neyaz, S. Y., Shamim, R., & Beg, M. S. (2019). An IoT based system for telemetry and control of Greenhouse environment. 2019 International Conference on Electrical, Electronics and Computer Engineering (UPCON). doi:10.1109/upcon47278.2019.8980258.