

PENDEKATAN IOT-METH DALAM PERANCANGAN SISTEM KENDALI DAN MONITORING PENYIRAMAN JAMUR OTOMATIS BERBASIS IOT MENGGUNAKAN APPSHEET

Yoga Ferry Pradhana¹, Moch Yusuf Asyhari², Juwari³

Teknik Informatika, Universitas PGRI Madiun, Madiun

Email: yogaferrypradhana@gmail.com

Abstract.

Oyster mushroom cultivation requires high humidity and temperature stability within the growing environment (kumbung). Farmers often face difficulties maintaining these conditions due to manual watering processes, time limitations, and lack of real-time environmental data. This research aims to design and implement an automatic mushroom watering control and monitoring system using the Internet of Things (IoT) and AppSheet, based on the IoT-METH (Collins, 2017) methodology. This methodology includes six stages: generate ideas, refine ideas, conceptualize the project, design the architecture, prototype, and deploy the IoT system. The system uses a DHT22 sensor to measure temperature and humidity, ESP32 microcontroller as the controller, and a 12V water pump for the watering mechanism. Data is transmitted in real-time to AppSheet, enabling remote monitoring and control via mobile application, both automatically and manually. The results show that the system performs well in maintaining kumbung conditions and provides flexibility and ease of use for farmers.

Keywords: *Internet of Things; Automatic Watering; Mushroom Farming; IoT-METH; ESP32; AppSheet*

Abstrak

Budidaya jamur tiram membutuhkan stabilitas suhu dan kelembaban yang tinggi di dalam kumbung. Petani sering mengalami kesulitan dalam menjaga kondisi tersebut karena proses penyiraman yang masih manual, keterbatasan waktu, dan tidak adanya data lingkungan secara real-time. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem kendali dan monitoring penyiraman jamur otomatis berbasis Internet of Things (IoT) menggunakan AppSheet dengan pendekatan metodologi IoT-METH (Collins, 2017). Metodologi ini meliputi enam tahapan: menghasilkan ide, menyempurnakan ide, konseptualisasi proyek, merancang arsitektur, membuat prototipe, dan menerapkan sistem IoT. Sistem menggunakan sensor DHT22 untuk membaca suhu dan kelembaban, mikrokontroler ESP32 sebagai pengendali utama, dan pompa air 12V untuk mekanisme penyiraman. Data dikirim secara real-time ke AppSheet, sehingga memungkinkan monitoring dan kontrol jarak jauh baik secara otomatis maupun manual. Hasil pengujian menunjukkan bahwa sistem ini bekerja dengan baik dalam menjaga kondisi kumbung dan memberikan fleksibilitas serta kemudahan bagi petani.

1. Pendahuluan

Budidaya jamur tiram (*Pleurotus ostreatus*) saat ini masih menghadapi tantangan besar, terutama dalam menjaga kondisi suhu dan kelembaban yang stabil di dalam kumbung. Permasalahan ini sangat relevan di wilayah dengan iklim panas dan kering seperti Kabupaten Magetan, Jawa Timur, di mana kondisi lingkungan tidak mendukung pertumbuhan optimal jamur tiram yang membutuhkan kelembaban 80–90% dan suhu berkisar 22–28°C [1]. Ketidaksesuaian iklim lokal dengan kebutuhan jamur tiram memaksa petani melakukan penyiraman manual, yang rentan terhadap inkonsistensi dan kesalahan. Akibatnya, kelembaban dalam kumbung menjadi tidak stabil, menurunkan produktivitas, memperpanjang waktu panen, dan menurunkan kualitas jamur yang dihasilkan [2]. [3]Mengembangkan Sistem monitoring kelembaban baglog jamur tiram dan penyiraman otomatis berbasis Internet of

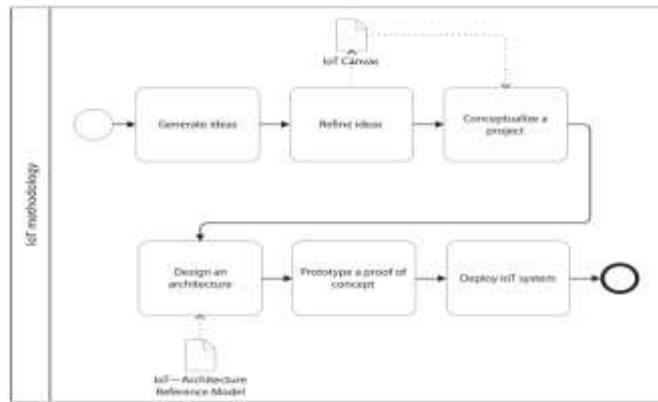
Things (IoT)[4] menggunakan esp8266. Sistem IoT dengan aplikasi Blynk digunakan untuk memantau kelembapan dan mengoperasikan alat penyemprot jamur secara real-time. Meski penyemprot hanya menjangkau titik tertentu, sistem ini tetap efisien dalam penggunaan waktu dan sumber daya. Pemantauan berbasis cloud memungkinkan kontrol jarak jauh dan pengambilan keputusan cepat guna menjaga kelembapan optimal bagi pertumbuhan jamur.

Kini tersedia alternatif yang lebih canggih dan fleksibel, yaitu ESP32[5] sebagai mikrokontroler dengan kemampuan dual-core serta konektivitas yang lebih stabil, dan appsheet sebagai platform pengembangan aplikasi yang terintegrasi dengan layanan Google, khususnya Google spreadsheet[6]. Aplikasi ini dapat dikoneksikan secara langsung melalui Google apps script[7] dan terhubung ke appsheet, sehingga memungkinkan proses pemantauan dan pengendalian sistem Internet of Things (IoT) secara real-time berbasis cloud. Dalam penelitian ini, penulis menggunakan ESP32, apps script, dan appsheet sebagai fondasi utama dalam pengembangan alat, guna meningkatkan efisiensi, keterhubungan data, dan kemudahan akses pengguna melalui platform digital. Potensi pemanfaatan platform AppSheet[8] sebagai antarmuka sistem monitoring dan kendali berbasis Internet of Things (IoT) menjadi solusi menarik yang belum banyak dieksplorasi. AppSheet memiliki keunggulan dalam kemudahan akses, fleksibilitas pengembangan, serta kompatibilitas dengan layanan cloud seperti Google Spreadsheet dan App Script. Oleh karena itu, konsep yang ditawarkan dalam penelitian ini adalah membangun sistem kendali dan monitoring alat penyemprot jamur otomatis berbasis IoT dengan pendekatan metodologi IoT-METH [9] dan diintegrasikan dengan platform AppSheet sebagai sarana pemantauan dan pengendalian jarak jauh.

Tujuan dari penelitian ini adalah merancang dan mengimplementasikan sistem penyemprotan otomatis yang mampu menjaga kelembaban optimal kumbung jamur secara konsisten, serta menyediakan antarmuka pemantauan yang mudah digunakan bagi petani jamur. Penerapan pendekatan IoT-METH diharapkan dapat menghasilkan sistem yang terstruktur, efektif, dan sesuai dengan kebutuhan pengguna.

2. Metode

Penelitian ini menggunakan pendekatan metodologi IoT-METH yang terdiri dari enam tahapan yang sistematis dan interaktif. Setiap tahapan memiliki peran spesifik dalam membangun solusi berbasis Internet of Things (IoT) secara efisien dan adaptif. Metode dijelaskan secara runtut agar pembaca dapat memahami serta mereplikasi proses yang dilakukan.



Gambar 1. Metode IoT-METH (Collins, 2017)

Setiap langkah dalam pendekatan IoT-METH digunakan untuk membangun sistem kendali dan monitoring alat penyemprot jamur otomatis berbasis IoT menggunakan AppSheet. Pada tahap *Co-create* (menghasilkan ide), diidentifikasi permasalahan petani jamur di Desa Pojoksari, Kecamatan Sukomoro, Kabupaten Magetan, yaitu penyemprotan yang masih manual dan tidak efisien, khususnya dalam hal waktu dan pengawasan kelembaban kumbung. Hasil identifikasi ini divisualisasikan dalam bentuk use case diagram sebagai dasar perancangan sistem[10]. Ideate (Menyempurnakan Ide) Ide awal yang telah dirumuskan kemudian disempurnakan melalui diskusi. *Internet of Things (IoT) canvas* yang digunakan

untuk mengidentifikasi aktor utama sebagai petani jamur, perangkat fisik sensor DHT22[11] dan modul relay perangkat pemrosesan ESP32, serta integrasi dengan layanan cloud melalui *google spreadsheet*, *apps script* dan *appsheet*. Sketsa awal antarmuka pengguna dapat dilihat pada yang dirancang dalam tahap ini untuk memastikan kemudahan penggunaan sistem.

Conceptualize Project (Konseptualisasi Proyek) Dilakukan analisis teknis terhadap semua komponen yang telah dirancang. Persyaratan sistem divalidasi kembali, seperti kebutuhan pembacaan suhu dan kelembaban, pengendalian pompa penyemprot, serta monitoring *real-time* melalui *appsheet*[12]. Design an Architecture (Mendesain Arsitektur) Seluruh komponen yang telah ditentukan dipetakan ke dalam arsitektur sistem berbasis *IoT Architecture Reference Model*. Perangkat keras seperti ESP32 dan sensor DHT22 berada pada lapisan *endpoint*, komunikasi data menggunakan koneksi Wi-Fi pada lapisan *connectivity*, sedangkan *middleware* dibangun melalui *google apps script*[13] untuk menjembatani ESP32 dan *google spreadsheet*. Layanan *Internet of Things* (IoT)[14] berupa sistem kendali otomatis, dan lapisan aplikasi dikembangkan menggunakan *appsheet* untuk mengontrol dan memonitor alat dari *smartphone*. Prototype (Membuat Prototipe) ESP32 dikodekan untuk membaca data dari sensor DHT22, mengaktifkan relay[15], saat kondisi kelembaban tidak sesuai, serta mengirim data ke *spreadsheet*. Data dari sistem IoT ditampilkan dan dikendalikan melalui antarmuka AppSheet. Tahap akhir meliputi penerapan langsung di lokasi budidaya jamur, dengan membandingkan kondisi sebelum dan sesudah penggunaan alat. Hasil evaluasi digunakan untuk penyempurnaan sistem secara bertahap agar sesuai dengan kebutuhan lapangan. Berkat penggunaan alat yang terjangkau dan mudah diakses, sistem ini juga berpotensi untuk direplikasi pada pertanian serupa.

3. Hasil dan Pembahasan

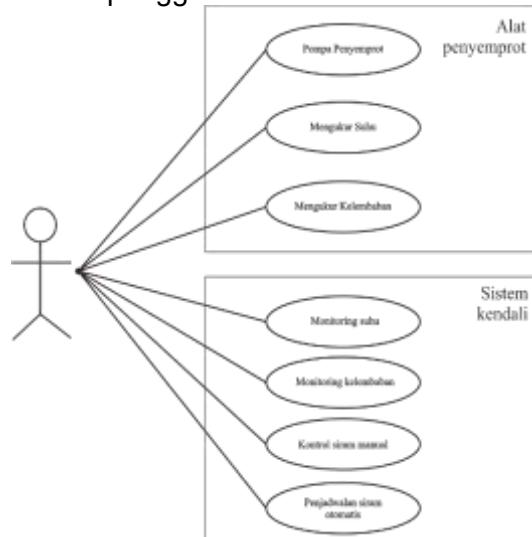
Hasil dan pembahasan akan menjelaskan pendekatan IoT-METH dalam perancangan sistem kendali dan monitoring penyiraman jamur otomatis berbasis IoT menggunakan AppSheet. Petani bisa memonitoring dan melakukan penyiraman jamur secara otomatis menggunakan alat dan mengurangi penggunaan tenaga manusia.

3.1. Penyajian Hasil

Setiap langkah dalam Pendekatan IoT-METH memerlukan proses tertentu yang digunakan untuk membuat sistem kendaali dan monitoring alat penyemprot jamur otomatis berbasis *Internet of Things* (IoT) menggunakan AppSheet. Berikut penggunaan IoT-METH dalam proses perancangan.

3.1.1. Co-create (Menghasilkan Ide)

Mengidentifikasi Permasalahan utama yang dihadapi petani jamur di Desa Pojoksari, Kecamatan Sukomoro, Kabupaten Magetan adalah proses penyemprotan yang masih manual dan kurang efisien, terutama dalam hal waktu dan pengawasan kelembaban. Identifikasi masalah ini kemudian divisualisasikan dalam bentuk use case diagram (Gambar 2) untuk menggambarkan interaksi antara pengguna dan sistem.



Gambar 2. Usecase Diagram Sistem

3.1.2. *Ideate* (Menyempurnakan Ide)

Fasilitasi diskusi dilakukan menggunakan artefak berupa *IoT Canvas* yang membantu proses curah pendapat untuk merumuskan dan memvalidasi kebutuhan sistem secara menyeluruh.



Gambar 3. IoT canvas

IoT canvas membantu memahami dan memformulasikan komponen-komponen utama dari sistem *Internet of Things* (IoT), mulai dari permasalahan yang ingin diselesaikan hingga aspek teknis seperti sensor, aktuator, *middleware*, dan antarmuka pengguna.

3.1.3 Conceptualize project (Konseptualisasi Proyek)

Konseptualisasi bertujuan untuk menjembatani kesenjangan antara gagasan dan pelaksanaan implementasi sistem. Peneliti melakukan validasi atas kebutuhan fungsional dan nonfungsional, serta melakukan analisis lebih dalam terhadap domain permasalahan.

Tabel 1. Konseptualisasi Sistem

Koseptualisasi Sistem

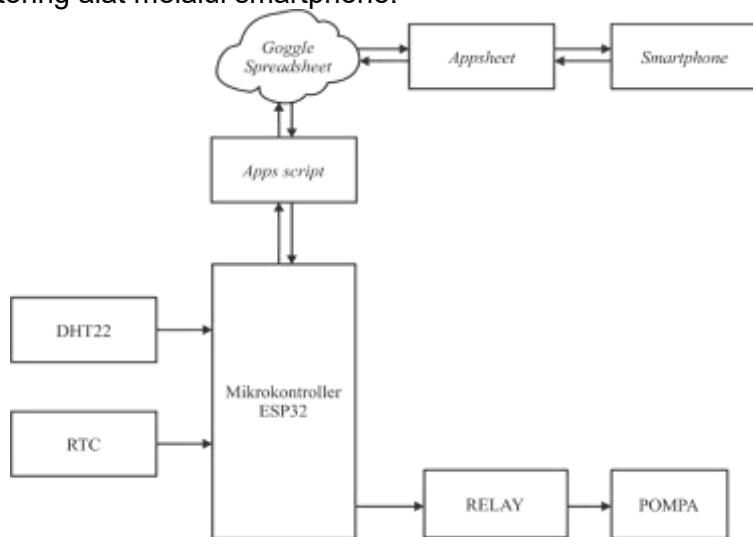
1	Sistem berbasis Internet of Things (IoT) menggunakan ESP32.
2	Integrasi aplikasi AppSheet yang terhubung dengan Google Spreadsheet untuk kontrol dan monitoring data.
3	Penggunaan sensor DHT22, RTC DS3231, dan modul relay untuk mendukung sistem otomatisasi penyiraman.
4	Sistem akan menampilkan data suhu, kelembaban, dan status penyiraman secara real-time.
5	Data akan disimpan secara otomatis dalam Google Spreadsheet dan dapat diakses dari perangkat seluler.
6	Pengujian akan dilakukan untuk mengukur akurasi, efisiensi, dan keandalan sistem.

Dalam proses ini menghasilkan dokumen persetujuan persyaratan sistem sebagai dasar pengembangan sistem yang sudah di jelaskan di atas.

3.1.4 Design an architecture (Mendesain Arsitektur)

Seluruh komponen sistem dipetakan ke dalam IoT Architecture Reference Model. Perangkat keras seperti ESP32 dan sensor DHT22 berada di lapisan endpoint, dengan komunikasi data melalui Wi-Fi pada lapisan konektivitas. Middleware menggunakan Google Apps Script untuk menghubungkan ESP32 dengan Google Spreadsheet. Pada lapisan

aplikasi, sistem kendali otomatis dikembangkan menggunakan AppSheet, memungkinkan kontrol dan monitoring alat melalui smartphone.



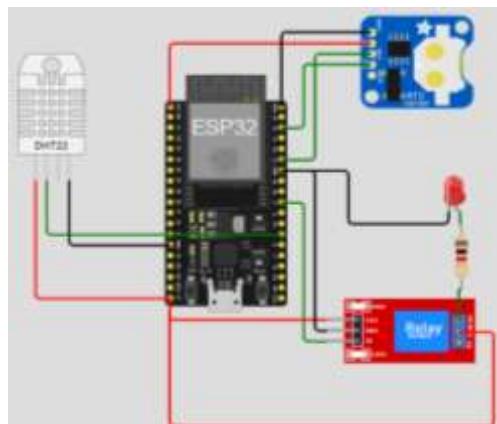
Gambar 4. Blok Diagram



Gambar 5. Anantarmuka Sistem

3.1.5 Prototype (Membuat Prototipe)

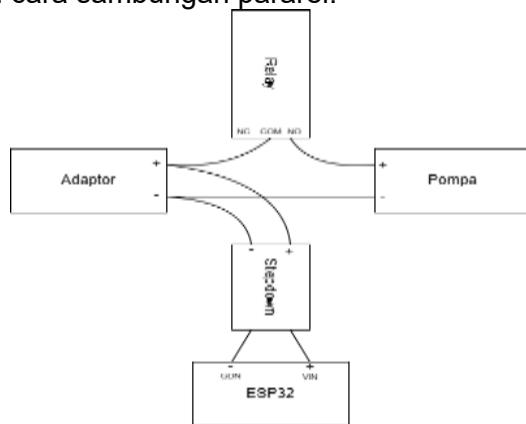
Prototipe yang direncanakan menggunakan skematik sistem yang di desain dan disimulasikan menggunakan wokwi.com untuk memperkecil presentasi kegagalan dalam perakitan alat.



Gambar 6. Skematik Sistem

ESP32 dikodekan untuk membaca data dari sensor DHT22, mengaktifkan relay saat kondisi kelembaban tidak sesuai, serta mengirim data ke spreadsheet. Data ini kemudian ditampilkan dan dikendalikan melalui antarmuka appsheet. Untuk meringkas tampilan alat

peneliti melakukan penyambungan kabel untuk sumber tegangan pada pompa dan ESP32 dilakukan dengan dengan cara sambungan pararel.



Gambar 7. Rangkaian Paralel

Kabel negatif pompa disambungkan ke kabel negatif dari adaptor, kabel positif dari pompa di sambungkan ke NO (*Normally Open*) relay, Kabel positif adaptor di sambungkan ke COM (*Common*) Relay, dan Kabel Positif dan negatif dari adaptor disambungkan ke *stepdown* yang berfungsi merubah dari 12V ke 5V serta disambungkan ke GDN dan VIN pada ESP32. Dikarenakan pompa menerima tegangan sebesar 12V dan ESP32 menerima tegangan 5V dengan cara ini alat bisa menggunakan 1 adaptor tegangan untuk 2 input yang berbeda.



Gambar 8. Rangkaian Alat Penyemprot Jamur Otomatis

3.1.6 Deploy (Menerapkan Sistem IoT)

Tahap akhir adalah menerapkan sistem ke lingkungan nyata, yaitu ke lokasi budidaya jamur.



Gambar 9. Box Elektronik Dengan Rangkaian

Untuk menguji fungsional dari alat untuk menyelesaikan permasalahan yang sudah ditentukan di awal. Fungsional dari pompa, relay, DHT22, dan sistem kendali yaitu appsheet.



Gambar 10. Lokasi Kumbung Jamur

3.2. Pembahasan

Penelitian terdahulu [3] masih memanfaatkan mikrokontroler ESP8266 sebagai dasar dalam pengembangan sistem berbasis Internet of Things (IoT). Namun, seiring dengan perkembangan teknologi, kini tersedia alternatif yang lebih canggih dan fleksibel, yaitu ESP32. ESP8266 adalah mikrokontroler dengan prosesor single-core 32-bit dan kecepatan hingga 160 MHz, serta RAM sekitar 160 KB. Sementara itu, ESP32 merupakan versi yang lebih canggih dengan prosesor dual-core 32-bit, kecepatan hingga 240 MHz, dan RAM mencapai 520 KB. Dari segi fitur, ESP32 unggul karena mendukung Bluetooth 4.2 + BLE, memiliki lebih banyak pin GPIO (hingga 34), kanal ADC yang lebih banyak dan lebih presisi (18 kanal 12-bit dibanding 1 kanal 10-bit pada ESP8266), serta dilengkapi DAC dan sensor internal seperti Hall sensor dan touch sensor. Kedua modul mendukung konektivitas WiFi 802.11 b/g/n dan komunikasi seperti SPI, I2C, UART, dan PWM, namun ESP32 menawarkan jumlah dan fleksibilitas yang lebih besar. Kapasitas flash memory keduanya tergantung modul, umumnya hingga 16 MB. ESP8266 lebih hemat daya dan murah (sekitar Rp20.000–Rp40.000), sedangkan ESP32 sedikit lebih mahal (sekitar Rp40.000–Rp70.000) dengan konsumsi daya yang lebih tinggi. Dalam hal kemampuan real-time, ESP32 lebih unggul karena mendukung RTOS seperti FreeRTOS, cocok untuk aplikasi multitasking dan proyek IoT yang kompleks. Singkatnya, ESP8266 cocok untuk proyek IoT dasar, sedangkan ESP32 ideal untuk sistem yang membutuhkan performa tinggi dan fitur lengkap.

Penelitian terdahulu [3] sistem monitoring dan kontrol berbasis IoT menggunakan aplikasi **Blynk** sebagai antarmuka utama karena kemudahan integrasinya dengan mikrokontroler seperti **ESP8266** dan **ESP32**, serta telah digunakan secara luas oleh banyak pengembang dan peneliti. Namun terdapat alternatif lain yang mulai digunakan, yaitu **AppSheet**, terutama pada sistem yang berorientasi pada pengolahan dan penyimpanan data berbasis **Google Spreadsheet**. AppSheet memiliki kelebihan dalam hal kemudahan pembuatan aplikasi tanpa perlu menulis kode (no-code), integrasi langsung dengan berbagai sumber data seperti Google Sheets, serta fitur otomatisasi dan tampilan antarmuka yang fleksibel untuk keperluan input maupun visualisasi data. Blynk adalah platform yang digunakan untuk mengontrol dan memantau perangkat IoT seperti ESP32 dan ESP8266 melalui antarmuka interaktif berupa tombol, grafik, dan slider. Meskipun mudah digunakan untuk kontrol real-time, Blynk tidak terintegrasi langsung dengan Google Spreadsheet, sehingga memerlukan middleware tambahan seperti Node-RED atau webhook untuk mengirim data ke spreadsheet. Sebaliknya, AppSheet dirancang khusus untuk membangun aplikasi berbasis data dan memiliki integrasi langsung (native) dengan Google Spreadsheet, memungkinkan pengguna membaca dan menulis data secara langsung tanpa perlu perantara tambahan. Blynk dan AppSheet sama-sama mendukung Android, iOS, dan web, tetapi memiliki pendekatan berbeda. Blynk mengandalkan komunikasi langsung antara perangkat IoT dan server untuk menampilkan data

di antarmuka pengguna, sedangkan AppSheet menggunakan spreadsheet sebagai basis data yang terhubung langsung ke UI aplikasi. AppSheet tidak mendukung koneksi langsung dengan perangkat IoT seperti ESP32, namun unggul dalam otomatisasi, workflow, serta kemudahan penggunaan berkat pendekatan no-code dan fleksibilitas tampilan seperti tabel, form, grafik, atau galeri.

Dari segi keamanan, Blynk menggunakan sistem login berbasis cloud miliknya, sedangkan AppSheet mendukung autentikasi Google dan kontrol akses pengguna yang lebih canggih. Keduanya menawarkan versi gratis dengan keterbatasan serta opsi berbayar untuk fitur lanjutan. Secara umum, Blynk lebih tepat untuk proyek kontrol perangkat IoT secara langsung, sementara AppSheet lebih cocok untuk aplikasi berbasis data dengan kebutuhan tampilan kompleks dan otomatisasi tinggi. Dalam konteks pengembangan sistem IoT, pemilihan metode kerja juga penting. Metode Prototipe menekankan iterasi cepat dan pengujian berulang, cocok untuk eksplorasi awal. Sementara itu, metode IoT-METH (Collins, 2017) menawarkan pendekatan yang lebih terstruktur, dirancang khusus untuk proyek IoT agar lebih sistematis dan terarah. Memahami perbedaan kedua metode ini membantu menentukan strategi yang paling efektif sesuai kebutuhan proyek.

Tabel 2. Perbandingan Methode

Aspek	Metode Prototipe	Pendekatan IoT-METH (Collins, 2017)
Tujuan Utama	Menghasilkan sistem fungsional awal secara cepat untuk diuji pengguna	Membangun sistem IoT secara terstruktur dan sistematis
Sifat Pengembangan	Iteratif dan fleksibel	Tahapan terstruktur dan menyeluruh
Jumlah Tahapan	Umumnya 3–4 tahapan utama (analisis, desain, uji, revisi)	6 tahapan: Cocreate, Ideate, Conceptualize, Design Architecture, Prototype, Deploy
Fokus Pendekatan	Fokus pada pengujian cepat dan perbaikan berulang	Fokus pada pemetaan kebutuhan, arsitektur sistem, dan validasi menyeluruh
Keterlibatan Pengguna	Tinggi pada tahap pengujian purwarupa	Tinggi sejak awal (co-creation) hingga evaluasi akhir
Cocok untuk	Proyek kecil hingga menengah dengan kebutuhan belum jelas	Proyek IoT kompleks yang butuh perencanaan matang dan dokumentasi lengkap
Kelebihan	Cepat menghasilkan purwarupa, responsif terhadap umpan balik pengguna	Terstruktur, sesuai untuk sistem IoT multi-komponen dan real-time
Kekurangan	Kurang sistematis, dokumentasi awal sering minim	Membutuhkan waktu dan pemahaman metodologi lebih dalam
Contoh Penggunaan	Aplikasi umum, sistem sederhana, pengujian antarmuka	Sistem kendali & monitoring berbasis IoT, seperti sistem penyiraman otomatis

Dalam pengembangan proyek IoT, mayoritas pendekatan yang digunakan adalah metode prototipe karena bersifat iteratif dan memungkinkan pengujian langsung terhadap fungsionalitas sistem. Namun demikian, pendekatan **IoT Methodology** yang dikemukakan oleh **Collins (2017)** juga layak dipertimbangkan karena menawarkan langkah-langkah yang lebih terstruktur dan spesifik, mulai dari identifikasi kebutuhan, desain sistem, integrasi perangkat, hingga evaluasi performa IoT secara menyeluruh.

Tabel 3. Skenario Penerapan IoT-METH

Tahapan	Skenario	Target	Hasil	Catatan
---------	----------	--------	-------	---------

Cocreate (Menghasilkan Ide)	Wawancara ke petani jamur	Mengetahui permasalahan	Menghasilkan ide dan terciptanya konsep sistem melalui usecase	Petani jamur memiliki kendala tentang penyemprotan kumbung yang manual, dan tidak bias memonitoring suhu dan kelembaban secara mudah.
Ideate (Menyempurnakan Ide)	Merumuskan kebutuhan sistem dan alat	Membuat rumusan (IOT Canvas)	Terciptanya iot canvas untuk dijadikan acuan utama sistem	IOT canvas akan mempermudah dalam perancangan sistem agar tidak ada kesalahan dan kekurangan dalam implementasinya
Conceptualize Project (Konseptualisasi Proyek)	Melakukan validasi atas kebutuhan fungsional dan nonfungsional	Memvalidasi dengan pihak ahli tentang alat dan komponen yang akan digunakan	Menghasilkan persetujuan persyaratan sistem sebagai dasar pengembangan sistem	Dokumen ini akan dijadikan sebagai dasar untuk melakukan pengembangan sistem lanjutan
Design an Architecture (Mendesain Arsitektur)	Memetakan ke dalam arsitektur sistem berbasis IoT Architecture Reference Model	Terpetakan 5 point yaitu titik akhir, konektivitas, middleware, layanan IoT, dan aplikasi sesuai dengan keperluan	Menghasilkan blok sistem yang menggambarkan 5 point dan rancangan antarmuka sistem kendali dan monitoring	Blok diagram digunakan untuk mempresentasikan struktur dan komponen yang digunakan dari sistem serta antarmuka merupakan proses perancangan cara interaksi antara pengguna dan sistem atau perangkat
Prototype (Membuat Prototipe)	Membuat simulasi perakitan alat menggunakan website online dan perancangan arus tegangan	Mendapatkan skematik sistem gambaran yang valid, perakitan dan perancangan jalur arus tegangan	Mendapatkan rangkiran jalur tegangan dan perakitan alat	Skematik sistem mengurangi presentase kerusakan alat ketika perakitan dan rangkaian jalur tanganan di fungsikan untuk membuat alat lebih rapi dan profesional
Deploy (Menerapkan Sistem IoT)	Melakukan penyempurnaan alat dan pengujian sistem sesuai di lapangan (kumbung)	Alat dapat berjalan dengan baik	Alat berfungsi dengan baik dan mengatasi permasalahan di awal	Penerapan di kumbung jamur yang ada di desa pojoksari kecamatan sukomoro kabupaten magetan dengan di dampingi petani

4. Kesimpulan

Pendekatan IoT-METH (Collins, 2017) terbukti efektif dalam perancangan sistem kendali dan monitoring penyiraman jamur otomatis berbasis IoT. Dengan mengikuti enam tahapan sistematis — mulai dari *cocreate*, *ideate*, *conceptualize project*, *design architecture*, *prototype*, hingga *deploy* — proses pengembangan sistem menjadi lebih terstruktur, terukur, dan sesuai kebutuhan lapangan. Setiap tahap berkontribusi penting dalam menghasilkan solusi yang adaptif dan relevan dengan permasalahan petani jamur, khususnya dalam menjaga kelembaban kumbung secara otomatis dan akurat.

Integrasi antara ESP32, sensor DHT22, pompa air 12V, dan platform AppSheet memungkinkan pengumpulan data lingkungan secara real-time serta pengendalian alat dari

jarak jauh melalui antarmuka mobile. AppSheet berperan sebagai jembatan antarmuka yang mudah diakses dan dikelola oleh pengguna tanpa perlu penguasaan teknis pemrograman lanjutan. Dengan demikian, penggunaan pendekatan IoT-METH tidak hanya meningkatkan efektivitas teknis sistem, tetapi juga mendukung keterlibatan pengguna akhir dalam seluruh proses pengembangan dan penggunaan sistem secara berkelanjutan. Pengembangan lebih lanjut bisa ditambahkan kamera untuk memantau kondisi kumbung dan mendeteksi jamur siap panen atau belum.

5. Referensi

- [1] Y. Hu *et al.*, "Effects and Mechanism of the Mycelial Culture Temperature on the Growth and Development of Pleurotus ostreatus (Jacq.) P. Kumm," *Horticulturae*, vol. 9, 2023, doi: 10.3390/horticulturae9010095.
- [2] M. J. Saputra and R. R. Suryono, "Technology Implementation Drip Irrigation on Plants Corn uses Soil Moisture Sensor and Esp 32 Microcontroller Implementasi Teknologi Irigasi Tetes pada Tanaman Jagung Menggunakan Sensor Soil Moisture dan Mikrokontroler Esp 32," vol. 5, 2025.
- [3] A. S. Yudhanto, P. Studi, and T. Informatika, "Sistem Monitoring Kelembapan Baglog Jamur Tiram Dan Penyiraman Otomatis Berbasis Iot Menggunakan," 2024.
- [4] Muhamad Maksum Hidayat, Nur Fitrianingsih Hasan, Intan Maya, and Martalina Wakerwa, "Sistem Kontrol Suhu Dan Kelembapan Otomatis Pada Budidaya Jamur Tiram Berbasis Iot Untuk Mendukung Smart Farming System," *Tek. Teknol. Inf. dan Multimed.*, vol. 4, no. 2, pp. 190–195, 2023, doi: 10.46764/teknimedia.v4i2.130.
- [5] R. Hidayati, K. Sari, Y. Halawa, A. T. A., and M. A. Harits, "Sistem Pemantauan Kualitas Udara Secara Real-Time Menggunakan Esp32 Dan Teknologi Iot," vol. 5, 2024, doi: 10.46576/djtechno.
- [6] A. J. Rusdi, F. Putri, and N. A. T. D. M. Wulan, "Implementasi Pendaftaran Online Pasien Rawat Jalan di RSAUDR. M. Munir Malang," vol. 4, 2024.
- [7] A. I. Asry, "Implementation of Google App Script in Cloud-Based Data Search Application," *JEAT J. Electr. Autom. Technol.*, vol. 1, 2022, [Online]. Available: <https://doi.org/10.61844/jeat.v1i2.405>
- [8] D. Dinh-Cuong, N. The-Vinh, and B. Anh-Tu, "Enhancing student admissions management efficiency through digital transformation: A case study using a no-code development platform," vol. 4, 2024, [Online]. Available: <https://doi.org/10.5281/zenodo.10971980>
- [9] Q. F. Hassan, A. ur R. Khan, and S. A. Madani, *Internet of Things Challenges, advances, and applications*. 2018. [Online]. Available: <https://civilica.com/doc/927474>
- [10] Siska Narulita, Ahmad Nugroho, and M. Zakki Abdillah, "Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS)," *Bridg. J. Publ. Sist. Inf. dan Telekomun.*, vol. 2, 2024, doi: 10.62951/bridge.v2i3.174.
- [11] R. Hermansyah and D. Wijayanto, "Sistem Monitoring Suhu dan Kelembapan Berbasis DHT22 dengan Metodologi Rapid Application Development," vol. 2, 2024.
- [12] M. K. Hassan, M. H. M. Rusli, and N. A. M. Salleh, "Desarrollo de un sistema de procesamiento de pedidos utilizando Google Sheets y AppSheet para una fabrica de PYME automotriz de Malasia Deposito," *J. Mech. Eng.*, vol. 20, 2023.
- [13] T. Wijaya, "Pemanfaatan Google App Script dalam Merancang Aplikasi Web Guna Meningkatkan Efisiensi Biaya Perusahaan," 2024.
- [14] Syahid, D. F. Saputro, A. Z. Sarianto, N. D. DPP, and N. Azmi, "Sistem Budidaya Jamur Berbasis Internet Of Things Guna Meningkatkan Produktivitas Petani Jamur Di Kabupaten Semarang," *J. Rekayasa Teknol. Nusa Putra*, vol. 18, 2022, doi: 10.52005/rekayasa.v5i1.96.
- [15] P. Paradva, P. A. M. Patel, and M. K. Chavada, "Power System Analysis and Relay Coordination for an Industrial Distribution System," *Ijireeice*, vol. 10, 2022, doi: 10.17148/ijireeice.2022.10440.

