

Rancang Bangun dan Implementasi Arsitektur Sistem Kendali Perangkat IoT Berbasis Cloud

Ikrar Bagaskara^{1*}, Annisaa Alam Firdausi², Pramono³

^{1*}Fakultas Ilmu Komputer
Universitas Duta Bangsa Surakarta
¹210103101@mhs.udb.ac.id (penulis
korespondensi)

²Fakultas Ilmu Komputer
Universitas Duta Bangsa Surakarta
²210103047@mhs.udb.ac.id

³Fakultas Ilmu Komputer
Universitas Duta Bangsa
Surakarta
³Pramono@mhs.udb.ac.id

Abstrak— Penelitian ini berfokus pada perancangan dan implementasi arsitektur sistem kendali perangkat IoT berbasis cloud menggunakan AWS IoT Core dan perangkat ESP32 dengan sensor DHT22. Tujuan utamanya adalah mengintegrasikan perangkat IoT dengan layanan cloud untuk memaksimalkan potensi perangkat IoT. Penelitian ini menggunakan metode prototyping untuk mengembangkan dan menguji sistem kendali ini. Hasil pengujian menunjukkan bahwa sistem ini berhasil mengintegrasikan perangkat IoT dengan AWS IoT Core secara efektif, memungkinkan pemantauan dan pengendalian perangkat secara real-time. Sensor DHT22 berhasil mengirimkan data suhu dan kelembaban ke cloud, yang dapat diakses dan dianalisis melalui dashboard AWS IoT. Penggunaan AWS IoT Core memberikan keunggulan seperti skalabilitas, keamanan, dan kemudahan integrasi dengan layanan cloud lainnya. Implementasi ini menunjukkan efisiensi dalam konsumsi daya dan biaya dengan penggunaan ESP32, sambil tetap memberikan kinerja yang andal. Dengan adopsi teknologi ini, terdapat potensi besar dalam memantau dan mengendalikan perangkat IoT dalam jumlah besar secara real-time, membuka peluang untuk pengembangan sistem kendali yang lebih kompleks dan luas jangkauannya. Arsitektur yang diusulkan dapat dijadikan referensi untuk pengembangan sistem kendali IoT lainnya yang membutuhkan integrasi dengan layanan cloud.

Kata kunci— Internet of Things (IoT), Cloud Computing, MQTT, ESP32, AWS IoT Core.

Abstract— This study focuses on the design and implementation of a cloud-based IoT device control system architecture using AWS IoT Core and ESP32 devices equipped with DHT22 sensors. The main objective is to integrate IoT devices with cloud services to maximize their potential. The research employs a prototyping method to develop and test this control system. The test results indicate that the system effectively integrates IoT devices with AWS IoT Core, enabling real-time monitoring and control. The DHT22 sensor successfully transmits temperature and humidity data to the cloud, which can be accessed and analysed through the AWS IoT dashboard. The use of AWS IoT Core offers advantages such as scalability, security, and ease of integration with other cloud services. This implementation demonstrates efficiency in power consumption and cost with the use of ESP32, while still providing reliable performance. The adoption of this technology shows significant potential for monitoring and controlling many IoT devices in real-time, opening opportunities for the development of more complex and extensive control systems. The proposed architecture can serve as a reference for developing other IoT control systems that require cloud service integration.

Keywords— Internet of Things (IoT), Cloud Computing, MQTT, ESP32, AWS IoT Core.

I. PENDAHULUAN

Internet of Things (IoT). IoT adalah ekosistem objek fisik ("benda") yang terhubung ke Internet dan ke benda-benda lain. Benda-benda fisik ini dapat berupa perangkat apa pun yang ditandai dengan sensor (smartphone, peralatan listrik pintar, peralatan kantor pintar, mobil, dan sebagainya) [1]. Jumlah IoT yang terhubung dari seluruh dunia hingga tahun ini sudah mencapai 17.08 miliar perangkat IoT, dan diprediksikan akan mencapai 29.42 miliar perangkat IoT pada tahun 2030 [2]. Penggunaan perangkat IoT terus merangkak dalam jumlah yang signifikan, hal ini dikarenakan banyaknya adopsi yang dilakukan, baik oleh individu, kelompok, hingga perusahaan dengan berbagai bidang, seperti layanan medis, ritel, layanan

pelanggan, rumah pintar, pemantauan lingkungan, dan internet industri. IoT memiliki banyak keuntungan terutama dalam menghemat biaya dan waktu dalam pertukaran data melalui jaringan. Referensi [3] menjelaskan terlepas dari berbagai manfaatnya, IoT juga memiliki risiko, terutama dengan aksesnya ke data pribadi dan non-pribadi yang sangat besar. IoT telah membuat ketersediaan dan akses data menjadi sangat mudah dan data dari berbagai perangkat yang terhubung dapat dengan mudah dikumpulkan, dianalisis, dilaporkan, dan dilihat melalui jaringan untuk meningkatkan efisiensi sistem apa pun di dalam lingkungan yang dibuat.

Dari perspektif kemampuan perangkat keras, desain dari perangkat IoT juga bertujuan agar

penyimpanan dan daya pemrosesan yang lebih kecil nya lebih kecil [4]. Karena sifat itu, banyak perangkat IoT tidak bisa di upgrade *firmware* nya sehingga memunculkan celah keamanan setelah dirilis di publik. Banyak nya perangkat *IoT* akan memunculkan potensi masalah jika tidak di monitor dengan baik. Dengan begitu pentingnya monitoring untuk perangkat IoT. Beberapa manfaat dari monitoring perangkat IoT adalah lebih efisien, Pengontrolan jauh lebih mudah, dan lebih fleksibel. Contoh dari monitoring perangkat IoT terdapat pada penelitian dengan judul sistem monitoring daya listrik dan pengontrolan perangkat elektronik [5].

Hingga karya ilmiah ini dibuat, banyak perusahaan atau *enterprise* yang mulai mengadopsi teknologi *cloud computing* pada sistem mereka. Hal ini dibuktikan dengan perbandingan dimana perusahaan lebih banyak mengeluarkan dana untuk sistem *cloud* mereka daripada data center nya. Pada tahun 2009, 60.7 miliar dollar (\$) dihabiskan untuk operasional data center sedangkan untuk *cloud* hanya 1 miliar dollar (\$), berbanding terbalik dengan tahun 2021 dimana 178 miliar dollar (\$) dihabiskan untuk *cloud* dan 98.05 miliar dollar (\$) untuk data center [6]. Referensi [7] menjelaskan salah satu keuntungan penggunaan teknologi *cloud computing* adalah *on-demand scalability of computing resources* atau kemampuan untuk menyesuaikan hardware atau *resource* sesuai dengan kebutuhan saat ini. Dengan metode pembayaran *pay-as-you-go*, dimana pengguna hanya membayar sesuai *resources* yang digunakan [8]. AWS atau *Amazon Web Service* sebagai *cloud* vendor dengan jumlah pemakai terbesar saat ini, berhasil mencetak *revenue* sejumlah 90,757 juta dollar (\$) pada tahun 2023, angka tersebut meningkat secara signifikan di banding tahun 2020 dimana *revenue* mereka di angka 45,370 juta dollar [9]. AWS mengalami peningkatan *revenue* di karenakan adopsi *cloud computing* secara meluas oleh banyak perusahaan, karena *cloud computing* membuat pemrosesan data menjadi lebih efisien pada beberapa sistem komputasi dan penyimpanan yang aksesnya dilakukan melalui internet [10].

Sehubungan dengan meningkatnya penggunaan IoT secara pesat, para *cloud vendor* pun juga bersaing menyiapkan layanan sebagai wadah untuk mengelola perangkat IoT, di mulai dari layanan

untuk konsumsi data, monitoring perangkat, hingga melakukan analisis data secara otomatis menggunakan *machine learning*. Iot membutuhkan cloud sebagai pusat manajemen karena biasanya masalah jaringan komunikasi terjadi karena CloudIoT melibatkan banyak teknologi jaringan, sehingga pemilihan protokol komunikasi harus mempertimbangkan dukungan oleh banyak perangkat. Sedangkan masalah keamanan terjadi karena banyaknya perangkat di luar sana yang dapat dengan mudah terhubung dengan CloudIoT, sehingga peran autentikasi dan otorisasi diperlukan untuk mengidentifikasi dan memvalidasi perangkat yang mengirimkan data. Tantangan lainnya adalah dengan banyaknya perangkat IoT yang digunakan, dibutuhkan mekanisme untuk manajemen perangkat [11].

Dengan meningkatnya tren penggunaan perangkat IoT yang signifikan, banyak perusahaan mulai mengintegrasikan puluhan perangkat IoT dalam ekosistem pabrik mereka. Hal ini menimbulkan tantangan dalam melakukan pemantauan perangkat IoT secara luas, terutama untuk perangkat yang berada di luar jangkauan jaringan internal perusahaan. Selain itu, penggunaan data center tradisional menyebabkan perusahaan memerlukan lebih banyak waktu untuk mengadopsi teknologi baru, yang mengakibatkan keterlambatan dalam inovasi dan persaingan di pasar. Penelitian ini bertujuan untuk merancang dan mengimplementasikan arsitektur sistem kendali perangkat IoT berbasis cloud yang dapat memfasilitasi pemantauan perangkat IoT secara luas, termasuk perangkat yang berada di luar jangkauan jaringan internal perusahaan. Selain itu, penelitian ini juga bertujuan untuk mempercepat adopsi teknologi dan meningkatkan kemampuan inovasi perusahaan melalui penggunaan teknologi cloud computing

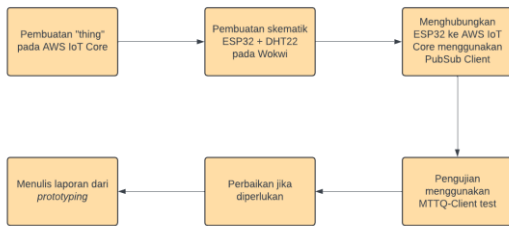
II. METODOLOGI PENELITIAN

2.1 Metode *Prototype*

Menurut [12] metode *prototyping* adalah mengaktualisasikan dan mewujudkan sebuah konsep dengan maksud untuk menganalisis, dengan prototipe sebagai contoh desain yang lebih pendek dari produk akhir. Salah satu keuntungan metode *prototype* adalah mengurangi waktu dan biaya dalam

pembuatan produk, metode ini juga dapat membantu mengirimkan produk lebih cepat ke user sehingga kita dapat segera mendapatkan feedback dari user [13].

2.2 Tahapan Pengembangan Sistem

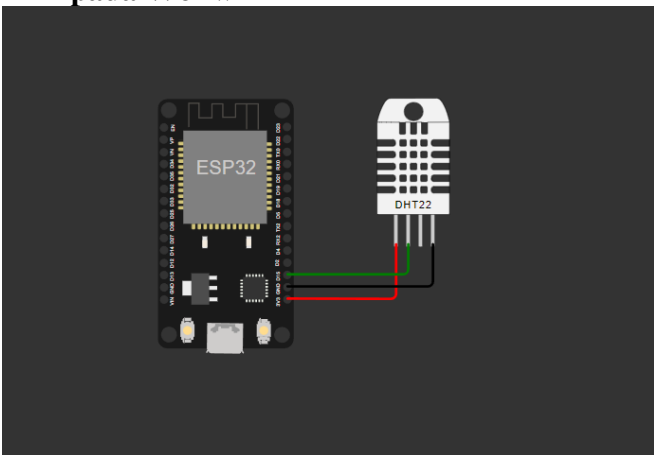


Gambar 1. Tahapan Pengembangan Sistem

1. Pembuatan "thing" pada AWS IoT Core

Membuat "thing" pada service AWS IoT Core dengan memasukkan beberapa parameter seperti nama, sertifikat, dan policy (izin akses antara perangkat IoT dengan AWS IoT Core)

2. Pembuatan skematik ESP32 + DHT22 pada Wokwi



Gambar 2. Skematik ESP32 dengan sensor DHT22

Pada gambar 2 nampak alat IoT yaitu ESP32 yang terhubung ke sensor DHT22 merupakan sensor pendeteksi suhu dan kelembaban yang disimulasikan pada aplikasi web Wokwi. Peneliti juga membuat program sederhana dengan mengambil dari beberapa referensi dan disesuaikan dengan skematik dan kebutuhan.

3. Menghubungkan ESP32 ke AWS IoT Core menggunakan PubSubClient

Tahap ini dilakukan dengan melakukan konfigurasi pada kedua sisi, yaitu AWS IoT Core dan code pada ESP32. Dengan membuat dan menjalankan service Pub/Sub pada AWS yang berfungsi untuk menerima dan mengirimkan pesan dari/ke perangkat IoT yang terhubung. Dan juga menginstall library PubSubClient pada code ESP32 agar dapat terhubung ke service AWS IoT Core.

4. Pengujian menggunakan MQTT-Client test

Pada tahap ini, dilakukan pengujian menggunakan service yang disediakan oleh AWS untuk menguji apakah perangkat ESP32 dapat terhubung ke AWS IoT Core dan melakukan komunikasi, baik mengirim atau menerima pesan.

5. Perbaikan jika diperlukan

Masalah yang ditemukan diperbaiki dan fitur yang diperlukan ditambahkan untuk memastikan bahwa sistem kendali perangkat IoT berbasis cloud berfungsi dengan optimal dan memenuhi semua kebutuhan pengguna.

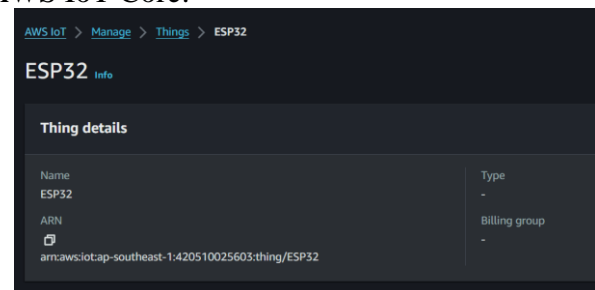
6. Menulis laporan hasil dari prototyping

Merangkum proses pengembangan dan hasil dari evaluasi prototype. Perubahan yang dilakukan serta rekomendasi untuk tahap pengembangan selanjutnya di dokumentasikan untuk memberikan panduan yang jelas dalam pengembangan dan implementasi sistem kendali perangkat IoT berbasis cloud.

III. HASIL DAN PEMBAHASAN

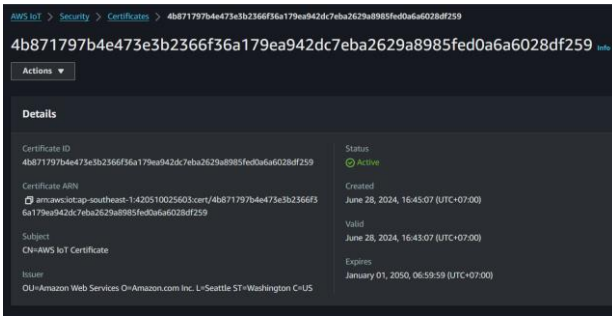
1. Pembuatan "thing" pada AWS IoT Core

Pada tahap konfigurasi untuk thing pada service AWS IoT Core.



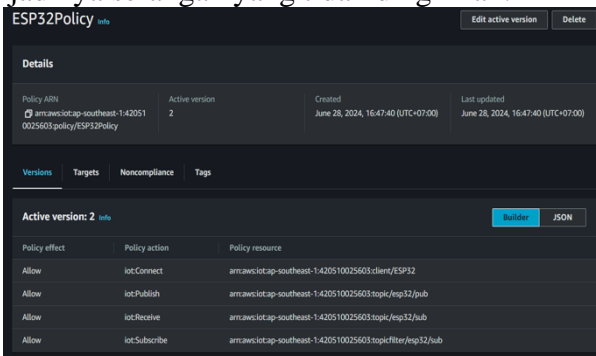
Gambar 3. Nama thing pada AWS IoT Core

Konfigurasi pertama peneliti membuat thing baru pada AWS IoT Core dengan nama thing adalah ESP32.



Gambar 4. Konfigurasi sertifikat pada thing

Kemudian dibuatkan sertifikat yang berfungsi sebagai *authorization* dan *authentication* sehingga perangkat IoT yang terhubung menggunakan koneksi yang aman, dan mengurangi resiko terjadinya serangan yang tidak diinginkan.



Gambar 5. Konfigurasi policy pada thing

Selanjutnya menambahkan policy atau aturan pada thing yang sudah dibuat, policy ini berguna untuk mengatur apa saja yang boleh dan tidak boleh pada thing yang sudah dibuat. Pada gambar 5, peneliti memberikan 4 policy yaitu:

a. Iot:Connect

Mengizinkan koneksi antara thing dengan ESP32.

b. Iot:Publish

Mengizinkan thing untuk mengirimkan pesan ke perangkat ESP32.

c. IoT:Receive

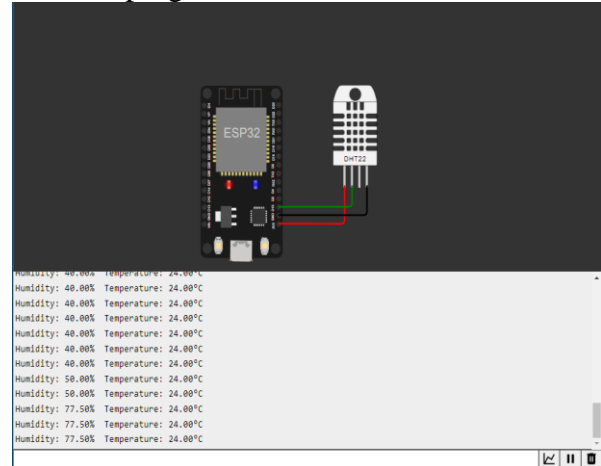
Mengizinkan thing untuk menerima pesan dari perangkat ESP32.

d. IoT:Subscribe

Mengizinkan thing untuk memantau pesan antara thing dengan perangkat ESP32

2. Pembuatan skematik ESP32 + DHT22 pada Wokwi

Di tahap ini peneliti sudah berhasil membuat skematik sesuai dengan gambar 2 dan berhasil mengimplementasikan code sesuai dengan kebutuhan program atau skematik.



Gambar 6. Menampilkan output dari program

Walaupun belum terhubung ke AWS IoT Core tapi rangkaian perangkat ESP32 sudah dapat mengeluarkan output sesuai dengan yang diharapkan, yaitu menampilkan tingkat suhu dan kelembapan.

3. Menghubungkan ESP32 ke AWS IoT Core menggunakan PubSubClient

Pada tahap selanjutnya peneliti berhasil menghubungkan ESP32 ke AWS IoT Core menggunakan PubSubClient dengan menambahkan library *PubSubClient*, kemudian pada code program juga ditambahkan block code untuk memberi tahu apakah ESP32 sudah terhubung dengan baik ke AWS IoT Core.

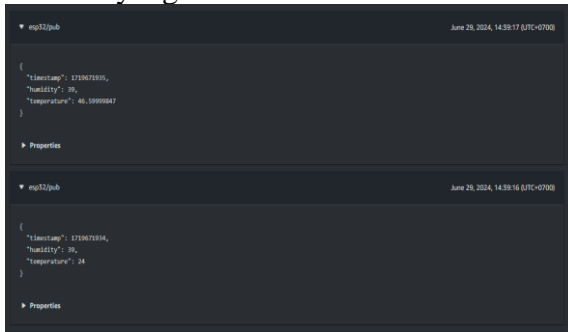


Gambar 7. ESP32 terhubung ke AWS IoT Core

Nampak pada gambar 7, terdapat output “Attempting MQTT connection..connected”, yang menandakan bahwa perangkat sudah terhubung dengan baik dan siap untuk mengirim atau menerima pesan

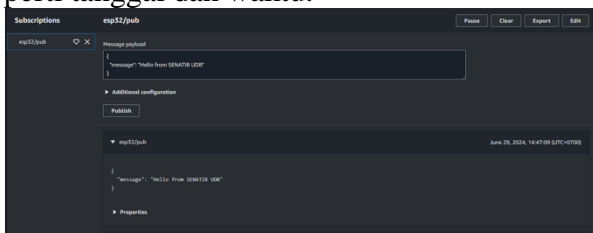
4. Pengujian menggunakan MTTQ-Client test

Pada pengujian ini, peneliti menggunakan MTTQ-Client test yang sudah di sediakan oleh AWS.



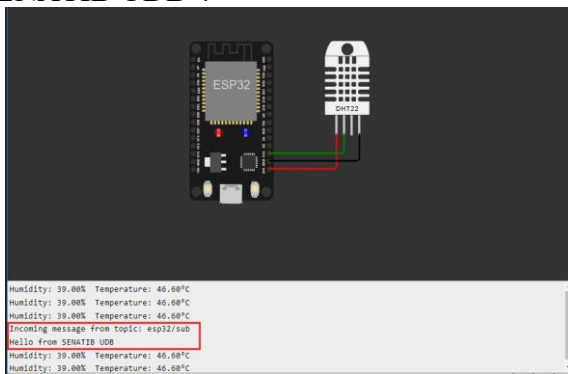
Gambar 8. Menerima pesan dari ESP32

Pada gambar 8 terlihat data atau pesan yang dikirimkan oleh perangkat ESP32 ke thing pada AWS IoT Core beserta dengan informasi tambahan seperti tanggal dan waktu.



Gambar 9. Mengirim pesan ke perangkat ESP32

Pada gambar 9, peneliti mencoba mengirim pesan ke perangkat ESP32 dengan pesan “Hello from SENATIB UDB”.



Gambar 10. Perangkat ESP32 menerima pesan dari AWS IoT Core

Kemudian pada gambar 10, perangkat ESP32 berhasil menerima pesan dari thing yang sudah di konfigurasi sebelumnya, hal ini berarti koneksi antara perangkat ESP32 dengan thing pada servie AWS IoT Core sukses.

IV. KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan arsitektur sistem kendali perangkat IoT berbasis cloud menggunakan AWS IoT Core dan perangkat IoT ESP32 dengan sensor DHT22. Tujuan utama adalah untuk mengintegrasikan perangkat IoT ke cloud guna memaksimalkan potensinya. Hasil pengujian menunjukkan integrasi perangkat IoT dengan layanan cloud yang efektif, memungkinkan pemantauan dan pengendalian real-time melalui AWS IoT Core. Data suhu dan kelembaban dari sensor DHT22 yang terhubung dengan ESP32 berhasil dikirimkan ke cloud dan dianalisis melalui dashboard AWS IoT.

Keunggulan AWS IoT Core meliputi skalabilitas, keamanan, dan kemudahan integrasi dengan layanan cloud lainnya. Implementasi ini membuktikan bahwa ESP32 efisien dalam konsumsi daya dan biaya, sambil tetap memberikan kinerja yang andal. Dengan cloud, perangkat IoT dapat dipantau dan dikontrol dalam jumlah besar secara real-time, membuka peluang pengembangan sistem kendali yang lebih kompleks. Arsitektur ini dapat dijadikan referensi untuk pengembangan sistem kendali IoT lainnya yang membutuhkan integrasi dengan cloud, serta meningkatkan fungsionalitas dan efisiensi sistem kendali IoT berbasis cloud.

REFERENSI

- G. Paolone, D. Iachetti, R. Paesani, F. Pilotti, M. Marinelli, and P. Di Felice, “A Holistic Overview of the Internet of Things Ecosystem,” *IoT*, vol. 3, no. 4, pp. 398–434, 2022, doi: 10.3390/iot3040022.
- L. S. Vailshery, “Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2023, with forecasts from 2022 to 2030 [In billions].” Accessed: Apr. 29, 2024. [Online]. Available: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>
- K. Lawal and H. N. Rafsanjani, “Trends, benefits, risks, and challenges of IoT implementation in residential and commercial buildings,” *Energy and Built Environment*, vol. 3, no. 3, pp. 251–266, Jul. 2022, doi: 10.1016/J.ENBENV.2021.01.009.
- T. Bakhshi, B. Ghita, and I. Kuzminykh, “A Review of IoT Firmware Vulnerabilities and Auditing Techniques,” *Sensors*, vol. 24, no. 2, 2024, doi: 10.3390/s24020708.
- C. Widiyari, F. Rendy, and W. Styorini, “Sistem Monitoring Daya Listrik dan Pengontrolan Perangkat Elektronik Berbasis IoT,” 2020.
- L. S. Vailshery, “Enterprise spending on cloud and data centers by segment from 2009 to 2022 [In billion U.S dollars].” Accessed: Apr. 29, 2024. [Online]. Available: <https://www.statista.com/statistics/1114926/enterprise-spending-cloud-and-data-centers/>
- S. A. Bello *et al.*, “Cloud computing in construction industry: Use cases, benefits and challenges,” *Autom Constr.*, vol. 122, p. 103441, 2021, doi: <https://doi.org/10.1016/j.autcon.2020.103441>.
- R. Islam *et al.*, “The Future of Cloud Computing: Benefits and Challenges,” *International Journal of Communications, Network and System*

- Sciences*, vol. 16, no. 04, pp. 53–65, 2023, doi: 10.4236/ijcns.2023.164004.
- L. S. Vailshery, “Annual revenue of Amazon Web Services (AWS) from 2013 to 2023 [In million U.S. dollars].” Accessed: Apr. 29, 2024. [Online]. Available: <https://www.statista.com/statistics/233725/development-of-amazon-web-services-revenue/>
- L. Golightly, V. Chang, Q. A. Xu, X. Gao, and B. S. C. Liu, “Adoption of cloud computing as innovation in the organization,” *International Journal of Engineering Business Management*, vol. 14, p. 18479790221093990, 2022, doi: 10.1177/18479790221093992.
- O. Bagus Pratama, A. Bhawiyuga, and K. Amron, “Pengembangan Perangkat Lunak IoT Cloud Platform Berbasis Protokol Komunikasi HTTP,” 2018. [Online]. Available: <http://j-ptiik.ub.ac.id>
- L. Kent, C. Snider, J. Gopsill, and B. Hicks, “Mixed reality in design prototyping: A systematic review,” *Des Stud*, vol. 77, p. 101046, 2021, doi: <https://doi.org/10.1016/j.destud.2021.101046>.
- R. Wasik, “The Advantages & Disadvantages of Prototyping - Rapids Reproductions.” Accessed: Apr. 28, 2024. [Online]. Available: <https://rapidsrepro.com/advantages-disadvantages-prototyping/>