

Rancang Bangun Pendeteksi Kebakaran Dan Monitoring Berbasis IoT Dengan *Microcontroller* NodeMCU

Arafa Sudarta ^{1,*}, Ferdiansyah Ferdiansyah ¹, Rimhot Richson Siahaan ¹,
Maruloh Maruloh ¹

¹ Sistem Informasi; Universitas Nusa Mandiri; Jatiwaringin Raya No. 2, Jakarta Timur 13620,
(021) 2853447 / 28534390; e-mail: arafa.sudarta@gmail.com, ferdy4unsur@gmail.com,
rimhotrnsiahaan@gmail.com, maruloh.mru@nusamandiri.ac.id

* Korespondensi: e-mail: arafa.sudarta@gmail.com

Diterima: 21 Mei 2022; Review: 23 Juni 2022; Disetujui: 23 Juni 2022

Cara sitasi: Sudarta A, Ferdiansyah F, Siahaan RR, Maruloh M. 2022. Rancang Bangun Sistem Pendeteksi Kebakaran dan Monitoring Berbasis IoT dengan *Microcontroller* NodeMCU. Bina Insani ICT Journal. Vol. 9 (1): 22-32.

Abstrak: Setiap rumah ataupun gedung kemungkinan atau berpotensi untuk mengalami kebakaran, yang disebabkan oleh banyak faktor, baik itu dari kelalaian pemilik ataupun disengaja oleh orang yang tidak bertanggung jawab. Dari hal-hal kemungkinan tersebut maka dibutuhkan suatu alat deteksi kebakaran yang akan memberi tahu pemilik bangunan sehingga dapat ditanggulangi lebih dini agar kejadian kebakaran dapat diminimalisir. Peralatan pendeteksi yang sudah beredar pun bervariasi, dari harga yang murah sampai dengan harga yang sangat mahal. Pada penelitian ini akan dibuat Rancang Bangun Pendeteksi Kebakaran Berbasis IoT Menggunakan Mikrokontroler NodeMCU untuk mendeteksi api dan asap untuk mengetahui terjadinya kebakaran menggunakan *IR Flame* Sensor dan MQ-2 Sensor, dimana inputan data berasal dari pembacaan nilai sensor tersebut dan data dikirimkan ke platform cloud service thinger.io dan data tersebut ditampilkan secara *live monitoring*. Dari rancang bangun yang dibuat, hasil yang didapat dari Rancang Bangun Pendeteksi Kebakaran Berbasis IoT Menggunakan Mikrokontroler NodeMCU untuk mendeteksi api dan asap untuk mengetahui terjadinya kebakaran.

Kata kunci: *cloud service, IoT, micro controller, MQ-2, NodeMCU*

Abstract: Every house or building has the possibility or potential to experience a fire, which is caused by many factors, be it from the negligence of the owner or intentionally by irresponsible people. From these possibilities, a fire detection device is needed that will notify the owner of the building so that it can be handled early so that fire incidents can be minimized. The detection equipment that has been circulating also varies, from cheap prices to very expensive prices. In this study, an IoT-Based Fire Detection Design using NodeMCU Microcontroller will be made to detect fire and smoke to determine the occurrence of fire using *IR Flame* Sensor and MQ-2 Sensor, where data input comes from reading the value of the sensor and the data is sent to the cloud service platform thinger.io and the data is displayed in live monitoring. From the design made, the results obtained from the IoT-Based Fire Detection Design Using NodeMCU Microcontroller to detect fire and smoke to determine the occurrence of fire.

Keywords: *cloud service, IoT, micro controller, MQ-2, NodeMCU*

1. Pendahuluan

Dari riset data yang diambil dari Dinas Penanggulangan Kebakaran dan Penyelamatan Provinsi DKI Jakarta terdapat total 5.043 kasus kebakaran dan penyelamatan di Provinsi DKI

Jakarta pada tahun 2020[1]. Dari sekian banyak kasus kejadian yang ada, kasus penyelamatan paling banyak juga berada di wilayah Jakarta Selatan yaitu 1.098 [2] kasus dan disusul dari wilayah Jakarta Timur dengan 1.013 [1],[2] kasus. Dari data yang dikeluarkan oleh Dinas Penanggulangan Kebakaran [1],[2] dan Penyelamatan Provinsi DKI Jakarta, kejadian kasus kebakaran yang disebabkan oleh lilin dan puntung rokok terdapat 43 kasus kejadian[1],[2].

Umumnya kasus kejadian kebakaran diketahui oleh pemilik atau masyarakat saat kondisi api sudah mulai membesar, asap hitam yang keluar dari suatu bangunan atau gedung, ataupun terjadi ledakan dan akhirnya dapat menimbulkan kerugian yang sangat besar bahkan memberikan dampak trauma psikologis bagi korbannya.

Untuk membuat *system* deteksi kebakaran, penulis melakukan riset untuk mencari referensi dari penelitian-penelitian sebelumnya yang berkaitan dengan latar belakang masalah. Referensi diambil dari berbagai jurnal atau karya ilmiah dan penelitian terdahulu yang berhubungan judul yang diambil. Dibawah ini penulis mengambil 3 penelitian terbaru.

Penelitian yang ditulis dan dilakukan oleh Ilham Santoso, Miftah Farid Adiwisatra, Bambang Kelana Simpony, Deddy Supriadi, Dini Silvi Purnia [3] pada tahun 2021 dengan judul "Implementasi Nodemcu Dalam *Home Automation* Dengan Sistem Kontrol Aplikasi *Blynk*" [3]. Penelitian ini bertujuan untuk membuat sebuah teknologi home automation berbasis IoT dengan menggunakan metode survey PPDIOO [3]. Sistem Otomatisasi rumah dengan memanfaatkan beberapa perangkat seperti modul NodeMCU ESP8266 sebagai controller, relay sebagai aktuator, sensor DHT11, sensor MQ-2, buzzer sebagai alarm, aplikasi Blynk sebagai pengolah data [3].

Penelitian yang dilakukan oleh Dian Indriani, Muhammad Subhan, Eka Rahmawati, 2021, "Sistem Alarm Kebakaran Berbasis Arduino Menggunakan Flame Detector Dan Sensor MQ-2" [5]. Hasil rangkaian sistem pendeteksi kebakaran bekerja berdasarkan objek proteksi kebakaran dan gas sebagai input yang terhubung dengan sebuah micro controller Arduino. Sistem yang digunakan adalah sensor flame detector dan sensor MQ-2[6]. Berdasarkan hasil penelitian, flame detector dapat mendeteksi objek berupa api hingga jarak 40 cm [5], dan sensor MQ-2 dapat mendeteksi gas hingga PPM > 1,20[5] pada jarak 20 cm. <1,20 pada jarak 40 cm [5]. Saat sensor mendeteksi kebakaran dan gas melebihi batas yang ditentukan, buzzer berbunyi, LED berbunyi dan layar LCD menampilkan pesan "Ada kebakaran". Semakin jauh dari sensor, semakin lama waktu yang dibutuhkan sensor untuk merespon, dan semakin jauh, semakin rendah konsentrasi gas yang direspon oleh sensor MQ-2 [5].

Referensi penelitian yang penulis diambil untuk menjadi referensi adalah penelitian yang dilakukan oleh Muhammad Immanuddin, Zulwisli, 2019, "Sistem Alarm Dan Monitoring Kebakaran Rumah Berbasis NodeMCU Dengan Komunikasi Android" [6]. *Monitoring system* menggunakan teknologi WiFi sebagai *early warning* kebakaran. Sensor DHT11 merupakan alat untuk membaca suhu udara dengan unit (°C) yang beroperasi dengan tegangan input 5volt. Keluaran diubah oleh *stabilizer voltage* [7] sehingga supply power terhadap perangkat yang terkoneksi dengan NodeMCU menjadi stabil dan dapat diakomodir. *Micro controller* dengan modul ESP8266 pada NodeMCU merupakan penerima dan pengirim data atau file yang telah diproses kemudian data tersebut dikirim ke *webserver* dan data tersebut dikirimkan ke ponsel untuk menampilkan hasil *monitoring*. Penginderaan suhu akan bekerja jika suhu di atas 37°C sehingga Android system akan memberikan peringatan bahwa telah terjadi kebakaran. Pada suhu 2°C, modul relay akan beroperasi dan menyalakan pompa air untuk memadamkan api. Sistem ini 5 menit lebih cepat dari sistem yang dikembangkan sebelumnya, yang membutuhkan waktu sekitar 30 menit untuk memberi tahu pemangku kepentingan [6]. Waktu deteksi dan pengiriman data yang begitu lama, kebakaran pun tidak dapat diantisipasi karena informasi yang dikirimkan sangat lambat [6].

Dari masalah kebakaran yang sudah diuraikan di atas dan dari hasil penelitian sebelumnya, sangat diperlukan sebuah sistem pendeteksi dini kebakaran secara *real time* dan memberikan peringatan dini kepada pemilik gedung, rumah atau bangunan lainnya sehingga kebakaran besar dapat dikurangi. Informasi deteksi dini kebakaran ini bisa memanfaatkan teknologi Internet.

Teknologi deteksi dini kebakaran ini bisa memanfaatkan *Internet Of Things* agar user bisa mendapatkan informasi dan monitoring ruangan gedung atau bangunan secara *real time* [9].

Internet of Things [10],[12] telah menjadi *buzzword* yang sering diucapkan oleh para pengguna internet baik itu didalam negeri atau luar negeri. *Internet of Thing* (IoT) [7],[12]

merupakan sebuah konsep dimana suatu objek berpotensi memiliki kemampuan untuk mengirimkan data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer [13].

Penelitian ini ditujukan untuk membuat sebuah prototype pendeteksi dini kebakaran dengan menggunakan *Internet of Thing* (IOT) dengan judul Rancang Bangun Pendeteksi Kebakaran dan Monitoring Berbasis *Internet of Things* [14],[15] Menggunakan *Micro Controller* Node MCU. Prototype sistem ini menggunakan 2 (dua) sensor yaitu, MQ-2 sebagai pendeteksi asap dan gas, *Flame Sensor* untuk mendeteksi api. Pemberitahuan kebakaran akan dikirim melalui email dan telegram, sebagai aktuator pada sistem deteksi dini kebakaran adalah *relay* yang akan berfungsi untuk menyalakan alarm, dan menyalakan pompa sebagai *water sprinkle* untukantisipasi awal. Selain itu *log* deteksi dan event kejadian, baik dari sensor ataupun *actuator* akan disimpan pada data *bucket cloud thinger.io* [8] yang dapat dijadikan sebagai analisa data ataupun data investigasi kejadian kebakaran.

2. Metode Penelitian

Untuk membuat rancang bangun pendeteksi kebakaran berbasis *internet of things* dengan NodeMCU diperlukan beberapa metode sebagai berikut:

Metode Research and Development

Metode penelitian yang penulis gunakan dalam melakukan penelitian ini yaitu menggunakan [16] salah satu metode yaitu metode penelitian dan pengembangan atau lebih dikenal dengan *Research & Development* [16]. Selain dari metode R&D [16] ini, metode pengembangan sistem *Prototype* untuk menghasilkan produk dalam bentuk *Prototype* sangat tepat, karena sebuah *Prototype* akan menghasilkan sebuah gambaran tentang cara kerja *system* yang akan berfungsi dalam bentuk lengkapnya [17].

Sebelum memulai tahapan RnD, penulis melakukan studi literatur untuk mendapatkan data. Studi literatur ini merupakan salah satu tahapan pra penelitian, dalam pengertiannya studi literatur adalah cara untuk menyelesaikan persoalan dengan cara menelusuri sumber tulisan yang pernah dibuat [16]. Dalam hal ini tentunya seorang peneliti harus memiliki wawasan yang luas terkait objek yang akan diteliti [16]. Sumber sumber penelitian pun harus jelas dan bisa dipertanggungjawabkan [18].

Wawancara

Wawancara dilakukan untuk menambahkan masukan terkait pengembangan rancang bangun pendeteksi kebakaran berbasis IoT. Dalam hal ini wawancara dilakukan untuk mengetahui karakteristik sistem pendeteksi kebakaran yang sudah terpasang atau yang sudah beredar dipasaran. Untuk menentukan narasumber, penulis melakukan korespondensi dengan beberapa narasumber ahli dibidang *fire alarm system*.

Tabel 1. Narasumber Wawancara Ahli dibidang *Fire Alarm System* Bandara Internasional Soekarno Hatta

Nama	Perusahaan	Lisensi	Jabatan
Dwi Kristiyanto	PT. Angkasa Pura II (Persero)	AHLI (BAF, FAS, FAS) Yang mengeluarkan Direktort Bandar Udara	Senior Officer Maintenance Quality Control
Iron Nur Pramudya	PT. Angkasa Pura II (Persero)	AHLI (IFS) Yang mengeluarkan Direktort Bandar Udara	Supervisor
Fachruri M. Noer	PT. Angkasa Pura II (Persero)	AHLI (IFS) Yang mengeluarkan Direktort Bandar Udara	Supervisor
Keterangan	IFS: <i>Information System</i> BAF: <i>Building Automation System (BAS) dan Fire Alarm System</i> P3B: Peralataan Pendeteksi Pemeriksaan Barang FAS: <i>Facility Aviation Security</i>		

Sumber: Hasil Penelitian (2021)

Metode Pengembangan System

Setelah dilakukan studi literatur dan wawancara dengan ahli dibidangnya, penulis mulai mengembangkan system dengan metode *prototype*. Ada beberapa tahapan dalam prototipe ini memiliki enam tahapan [19].

Tahapan Requirements getting and analysis

Pada tahapan ini system didefinisikan secara rinci apa saja kebutuhan yang akan digunakan pada system deteksi kebakaran ini, baik meliputi *hardware* maupun *software*.

Quick Design

Tahapan kedua ini adalah membuat desain sederhana [17] yang nantinya akan memberikan gambaran secara singkat bagaimana system yang akan dibuat dan tentunya berdasarkan dari diskusi pada langkah pertama. Untuk tahapan ini gambaran system dibuat dalam bentuk *use case diagram*.

Build a Prototype

Desain cepat telah dibuat maka proses pembangunan prototipe [19] sebenarnya dijadikan bahan rujukan penulis untuk mengembangkan *system* tersebut.

Initial User Evaluation

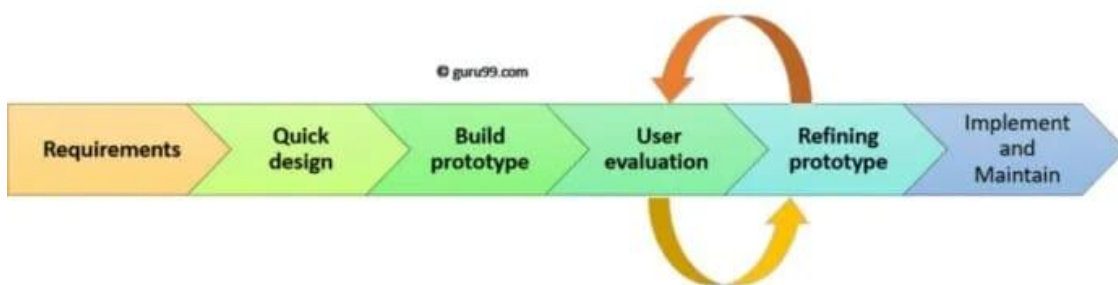
System yang telah dibuat kemudian di tampilkan kepada user atau pengguna awam dan narasumber ahli dibidang *fire alarm system*, dengan harapan ada *feedback* terhadap *system* yang telah dibuat

Refining Prototype

Jika pengguna ataupun narasumber ahli dibidangnya memberikan catatan perbaikan maka perbaikan system akan terus berulang pada tahap *initial user* dan *refining prototype*. Apabila tidak ada catatan maka tahapan berikutnya dapat dilakukan.

Implement Product and Maintain

Pada tahapan ini penulis membangun system berdasarkan prototipe akhir yang dianggap tidak ada catatan. Dan selanjutnya system telah dibuat akan dilakukan penjadwalan *preventive maintenance*.



Sumber: guru99.com (2021)

Gambar 1. Tahapan Metode Prototype

3. Hasil dan Pembahasan

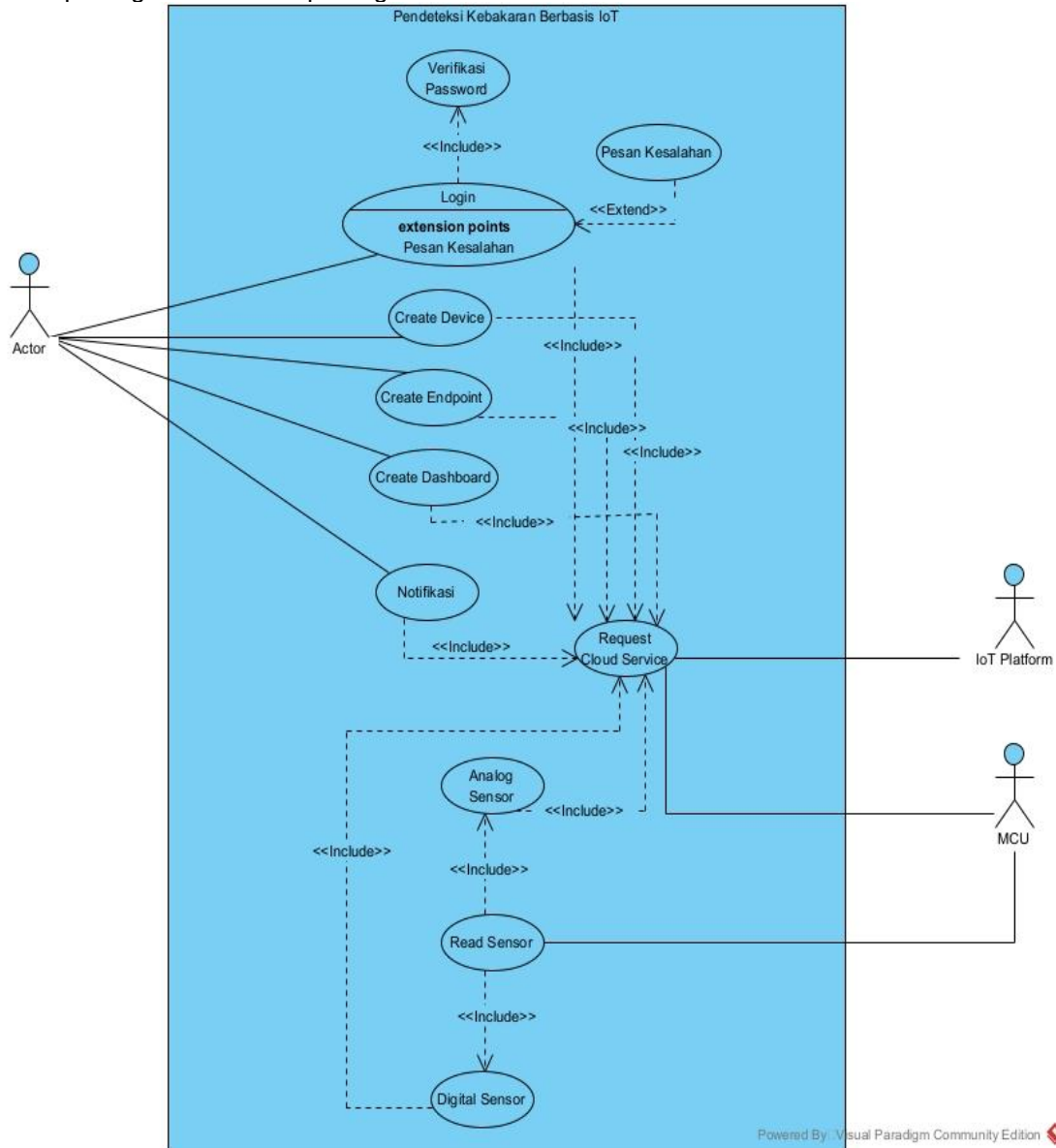
Hasil Rancangan

Dari Rancang Bangun Pendeteksi Kebakaran dan Monitoring Berbasis *Internet of Things* Menggunakan *Micro Controller Node MCU* [3], [20],[22] diharapkan mendapatkan hasil luaran sebagai berikut: 1) Memudahkan pengguna dalam memantau situasi pada suatu gedung atau bangunan secara *real time*, 2) Memudahkan pengguna atau pemilik bangunan/gedung dalam mendeteksi dini adanya kemungkinan terjadinya kebakaran secara tepat dan cepat, 3) Memberikan informasi pemetaan lokasi titik api secara *real time*, 4) Dapat diakses secara cepat karena menggunakan berbagai platform sebagai monitoring dan notifikasi, 5) Data sensor dari *Cloud IoT Platform* dapat digunakan untuk analisa data dan investigasi ketika terjadinya kebakaran.

Hasil dari pembuatan prototype akan digambarkan secara lengkap dengan UML (*Unified Modelling Language*) baik itu dari *use case*, *activity diagram*, *entity relationship diagram*, *sequence diagram*, *deployment* yang sesuai dengan kebutuhan rancang bangun

sistem. Pada pembahasan ini juga akan ditampilkan *user interface* sistem monitoring deteksi dini kebakaran dan alat pendeteksi kebakaran.

Sebelum melanjutkan tahap implemementasi tentunya tahap desain harus dibuat terlebih dahulu. Pada tahap inilah gambaran atau merepresentasikan dunia abstrak sistem kedalam suatu model yang sesuai dengan kebutuh pemakai atau pengguna yang merupakan kombinasi antara perangkat keras dan perangkat lunak.



Sumber: Hasi Penelitian (2021)

Gambar 2. Use Case Diagram

Use Case Login

Pengguna dapat melakukan *login* ke *Cloud service thinger.io* [8] untuk menambah *device*, membuat *trigger endpoint*, cetak data *bucket*, membuat *dashboard*, dan menambahkan lokasi peralatan.

Use Case Create Device dan create endpoint

Pada *use case* ini pengguna dapat menambahkan *device* baru dan setting credential. Yang nantinya akan dimasukkan pada *NodeMCU* sebagai komunikasi *MCU* dengan *Cloud Service*. Pengguna dapat menambahkan *endpoint* yang berfungsi sebagai *trigger notifikasi* kepada pengguna ketika terjadi kondisi yang sudah dibuat pada *MCU*.

Use Case Cetak Log Sensor

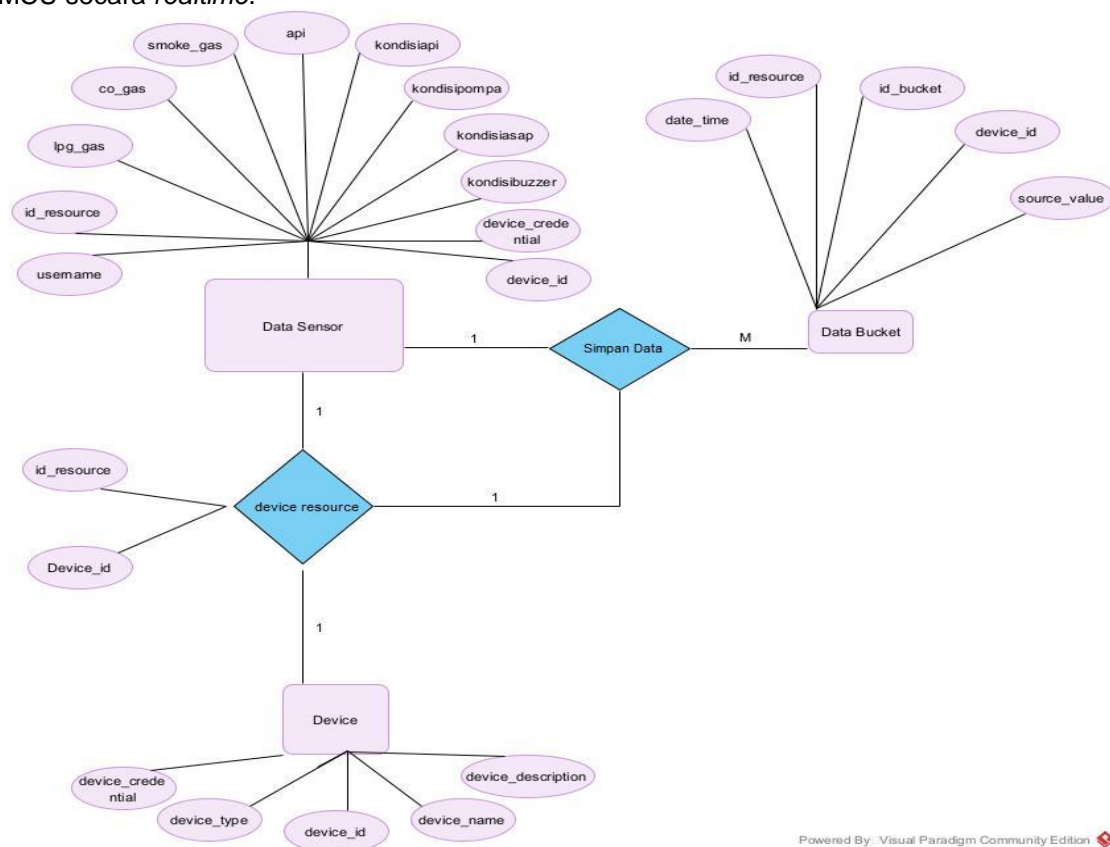
Pada use case ini digambarkan bahwa pengguna dapat memanggil atau mencetak data sensor yang dikirimkan oleh MCU kepada *cloud service*, yang akhirnya data tersebut dapat digunakan sebagai analisa kejadian kebakaran.

Use Case Dashboard Live Monitoring Deteksi Kebakaran

Pengguna dapat melakukan banyak hal pada dashboard monitoring, pengguna dapat menambahkan widget data resource yang diterima dari MCU, membuat *dashboard* menjadi *private* atau *public*.

Use Case Notifikasi

Pengguna dapat menerima notifikasi tanpa harus melihat *live dashboard* monitoring yang dikirimkan secara otomatis oleh *cloud service* yang tentunya data tersebut didapatkan dari MCU secara *realtime*.



Sumber: Hasil Penelitian (2021)

Gambar 3. Entity Relationship Diagram Pendeteksi Kebakaran berbasis IoT

Use Case Read Sensor dan Use Case Control Actuator

MCU dapat mengaktifkan dan membaca nilai sensor baik itu analog dan digital. Disisi lain MCU juga dapat melakukan kontrol terhadap *actuator*, pada kondisi ini yang disebut *aktuator* adalah *relay*, *relay* akan mengaktifkan *buzzer* dan pompa *sprinkle*.

Use Case Request Service Cloud

Cloud Service akan memberikan dan menerima request kepada MCU dan pengguna, karena *cloud service* ini memiliki kemampuan komunikasi secara *biderictional* atau komunikasi 2 (dua) arah.

Activity Diagram Deteksi Api, Notifikasi dan Data Logging

Activity Diagram Deteksi Api, *Notifikasi* dan *Data Logging* menjelaskan proses inisiasi sensor analog dan digital untuk melakukan pembacaan nilai sensor. Jika nilai sensor melebihi nilai ambang batas, maka MCU akan melakukan kontrol terhadap aktuator baik itu berupa *buzzer* atau mengaktifkan *sprinkle water*. Dari aktifitas ini juga dapat diketahui proses komunikasi antara pengguna, *cloud service*, MCU.

Entity Relationship Diagram

ERD merupakan suatu model digunakan untuk menjelaskan mengenai hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang memiliki hubungan antar relasi dan biasanya dapat digambarkan dengan notasi serta simbol. Berikut ini merupakan ERD dari Rancang Bangun Pendeteksi Kebakaran Berbasis IoT dengan *Microcontroller NodeMCU* [21], [22].

Hasil Pengujian

Blackbox testing[23] merupakan tool yang digunakan dalam pengujian penelitian ini, metode pengujian pada *system*, baik pada sisi *hardware* maupun *software*. Dari sisi *hardware*, terdapat beberapa tes fungsional pengujian yaitu fungsional koneksi NodeMCU ke jaringan internet, fungsi sensor Analog, fungsi sensor digital, fungsi *Relay*, fungsi LCD Display 16x2 I2C.

Untuk pengujian fungsional *software* yang diuji adalah *website optimized connection* dengan tool *webpagetest.org*, *form login*, *Create Device*, *create endpoint*, *create dashboard*, *data bucket report*.

Pengujian Konektivitas Node MCU dan Cloud Service

Pada Pengujian merupakan tahap awal dari system deteksi kebakaran, karena langkah awal untuk melakukan koneksi antara NodeMCU dan *Cloud Service Thinger.io*, selain itu tahapan inilah menjadi hal yg sangat penting dari system tersebut. Dibawah ini merupakan tabel pengujian.

Tabel 2. Tabel pengujian Node MCU dan Cloud Service

No	Pengujian	Hasil Harapan	Hasil Pengujian	Kesimpulan
1	Pengguna membuat device baru	Device baru dibuat	Sesuai harapan	Valid
2	Pengguna memasukan <i>username, id device, credential</i> ke <i>sketch arduino</i>	Device dan <i>Cloud service</i> dapat berkomunikasi	Sesuai harapan	Valid
3	Pengguna memasukan <i>username</i> saja ke <i>sketch arduino</i>	Status Device MCU pada <i>cloud service disconnected</i>	Sesuai harapan	Valid
4	Pengguna memasukan <i>ID Device</i> dan <i>Credential</i> saja ke <i>sketch arduino</i>	Status Device MCU pada <i>cloud service disconnected</i>	Sesuai harapan	Valid
5	Pengguna membuat device baru	Device baru dibuat	Sesuai harapan	Valid
6	Pengguna memasukan <i>username, id device, credential</i> ke <i>sketch arduino</i>	Device dan <i>Cloud service</i> dapat berkomunikasi	Sesuai harapan	Valid
7	Pengguna memasukan <i>username</i> saja ke <i>sketch arduino</i>	Status Device MCU pada <i>cloud service disconnected</i>	Sesuai harapan	Valid

Sumber: Hasil Penelitian (2021)

Proses ini merupakan tahap awal MCU dapat berkomunikasi dengan *thinger.io*. Ketika perangkat MCU tidak dapat terhubung dengan *thinger.io* maka proses penyimpanan data, menampilkan resource nilai sensor, *control actuator* tidak dapat dilakukan. Hasil pengujian dapat dilakukan dengan tingkat keberhasilan 100%.

Pengujian Aktuator

Tujuan dari pengujian ini adalah untuk mengetahui apakah relay yang digunakan dapat bekerja dengan baik. Pengujian dilakukan dengan cara untuk memicu sensor, analog atau digital, *Buzzer Relay* bekerja ketika *flame* sensor mendeteksi kebakaran, sensor MQ2 dapat membaca atau mendeteksi kadar Asap, CO, LPG dengan nilai sensor status harga lebih besar atau sama dengan 300 *ppm/part per milion*.

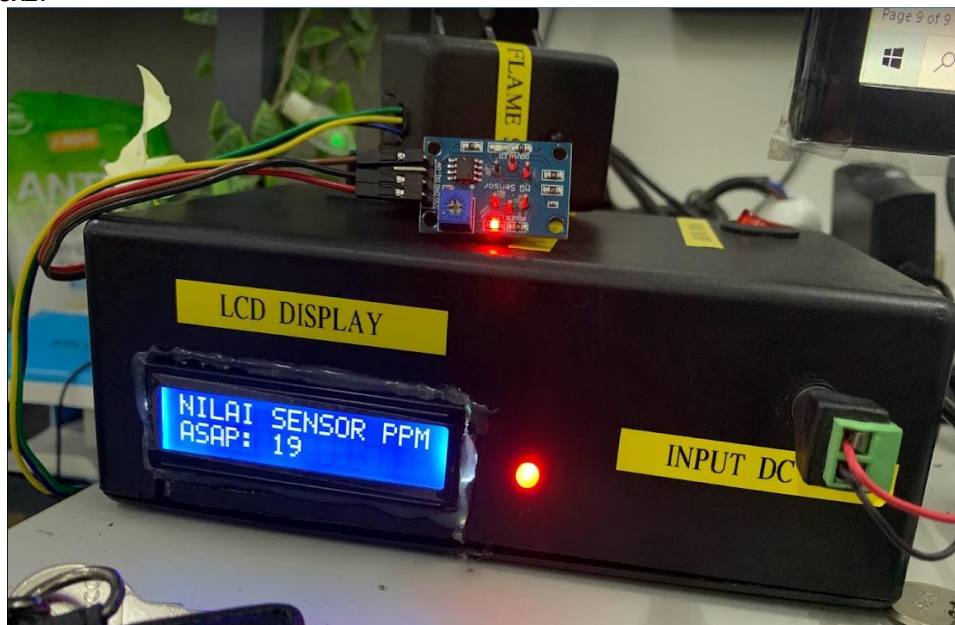
Pengujian berikutnya dengan memberikan *trigger* pada sensor MQ-2 dengan kondisi nilai sensor MQ-2 diatas atau sama dengan 1000ppm maka *Relay pompa sprinkle* dan *buzzer* aktif. Data hasil pengujian *Relay* pada tabel di atas menunjukkan bahwa *Relay 2 channel* bekerja dengan baik dan sesuai dengan perintah kondisi dengan tingkat keberhasilan 100%.

Tabel 3. Tabel pengujian aktuatur

No	Pengujian	Hasil Harapan	Hasil Pengujian	Kesimpulan
1	Relay Buzzer	Relay Buzzer aktif ketika mendeteksi api	Sesuai harapan	Valid
2	Relay Buzzer	Relay Buzzer aktif ketika mendeteksi asap diatas atau sama dengan 300ppm	Sesuai harapan	Valid
3	Relay Buzzer	Relay Buzzer aktif ketika mendeteksi LPG diatas atau sama dengan 300ppm	Sesuai harapan	Valid
4	Relay Buzzer	Relay Buzzer aktif ketika mendeteksi CO diatas atau sama dengan 300ppm	Sesuai harapan	Valid
5	Relay Pompa Sprinkle	Relay Buzzer dan pompa sprinkle aktif ketika mendeteksi asap diatas atau sama dengan 1000ppm	Sesuai harapan	Valid
6	Relay Pompa Sprinkle	Relay Buzzer dan pompa sprinkle aktif ketika mendeteksi LPG diatas atau sama dengan 1000ppm	Sesuai harapan	Valid
7	Relay Pompa Sprinkle	Relay Buzzer dan pompa sprinkle aktif ketika mendeteksi CO diatas atau sama dengan 1000ppm	Sesuai harapan	Valid

Sumber: Hasil Penelitian (2021)

Berikut ini merupakan hasil tampilan nilai sensor MQ-2 yang ditampilkan pada *Display LCD 16x2*.



Sumber: Hasil Penelitian (2021)

Gambar 4. Perangkat Pendeteksi Kebakaran

Pengujian Performa dan Kecepatan

Pengujian berikutnya adalah pengujian terhadap performa dan kecepatan *live dashboard monitoring* yang diberikan oleh *thinger.io*.

Pengujian dengan GTMetrix

Untuk menguji kecepatan dan efisiensi dari sebuah website yang diuji dengan menggunakan *analyze engine google page speed* dan *Yahoo Yslow*. Pada tes ini akan ditampilkan hasil serta rekomendasi yang harus dilakukan dan hasil akhir dari tes ini akan diberikan poin penilaian sebuah website dengan grade A sampai F.

Pengujian dengan Webpagetest.org

Merupakan *tools* untuk pengetestan dan memastikan performa sebuah web secara *real*. Pada tes ini *webpagetest.org* akan memberikan *grade* pada akhir tes. *Test* ini juga akan memberikan pemilik sebuah *website* rekomendasi penggunaan *image* untuk *optimasi load* pada sebuah website.

Pengujian dengan *BlazeMeter*

Pengujian terakhir adalah dengan menguji tingkat stress sebuah *website*, dimana sebuah *website* akan diuji dan dilihat respon *time*, data *transfer*, mengetahui tingkat *error* yang terjadi pada saat penerimaan data. *Live dashboard monitoring* akan diuji dengan 25 VU (*Virtual User*). Pada hasil akhir pengujian *live dashboard monitoring* ini dapat menangani permintaan *user* sebanyak 20 VU secara bersamaan.

Potensi

Rancang bangun pendeteksi kebakaran berbasis IoT dengan NodeMCU ini dikembangkan atas dasar bagaimana sebuah perangkat pendeteksi kebakaran dapat kita lihat dan baca secara *real time*, bagaimana pengguna dapat menerima notifikasi kebakaran sebagai *early warning* sehingga kerugian jiwa dan materil dapat ditekan seminimal mungkin, dan tentunya adalah produk konvensional pendeteksi kebakaran mahal.

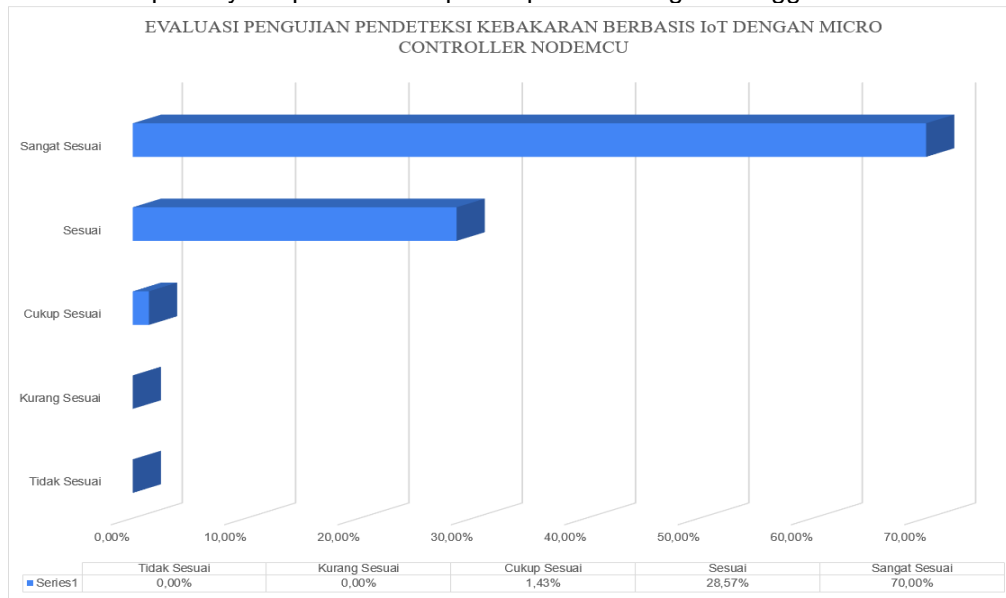
Kelebihan dari perangkat ini adalah kita dapat menambahkan sensor-sensor pada *microcontroller* sesuai dengan kebutuhan kita sehingga dapat dikembangkan menjadi sebuah perangkat IoT secara *kompleks*.

Tantangan

Tantangan *Internet of Things* [13],[15] kedepan yaitu semakin banyak perangkat atau hardware yang akan terkoneksi, maka akan semakin banyak pula aplikasi dan layanan yang berjalan. Terdapat berbagai macam aplikasi, hardware yang nantinya akan bekerja secara bersamaan untuk menghasilkan data-data tersebut. Di baliknya, tersembunyi tantangan [13],[15],[24] yang memiliki potensi ancaman terhadap keamanan data, keamanan fisik, keamanan dari perangkat, regulasi, privasi, enkripsi, otentikasi, dan segudang ancaman lain yang perlu ditangani agar kendala tersebut tidak mengganggu di kemudian harinya dan masih banyak lagi tantangan lainnya.

Hasil Implementasi

Dibawah ini merupakan hasil survey dari kuesioner yang sudah disebar untuk melihat dari hasil evaluasi tingkat efektivitas implementasi perangkat pendeteksi kebakaran. Kuesioner ini terdiri dari 27 pertanyaan pada 70 sampel responden dengan menggunakan skala *likert*.



Sumber: Hasil Penelitian (2022)

Gambar 5. Grafik jawaban keuisioner

Skala *Likert* merupakan pengukuran skala persepsi, sikap, atau pendapat seseorang atau kelompok terhadap suatu peristiwa atau fenomena sosial, berdasarkan definisi operasional yang dikembangkan oleh peneliti. Pertanyaan penelitian sering diterapkan dalam kuesioner dan

sering digunakan untuk penelitian diadakan. dalam bentuk survei, termasuk dalam bentuk survei penelitian deskriptif [25].

Pencipta skala *Likert* [25],[28] bernama Rensis Likert yang berasal dari Amerika Serikat, menerbitkan laporan yang menjelaskan penggunaannya [26],[28]. Ketentuan skala pada pertanyaan dibuat dari 1 sampai 5 dengan rumus sebagai berikut:

$P = T \times P_n$	Keterangan :
Rumus Index	T = Total jumlah responden yang memilih
$\% = \text{Total Skor} / Y \times 100$	P _n = Pilihan angka skor <i>likert</i>
	Y = skor tertinggi <i>likert</i> x jumlah responden
	X = skor terendah <i>likert</i> x jumlah responden

Pada grafik di atas dapat diketahui bahwa dari 70 responden dengan 27 pertanyaan dapat disimpulkan bahwa responden menjawab cukup sesuai adalah 1,43%, sesuai 28,57 %, dan sangat sesuai 70% selebihnya 0% dengan jawaban tidak sesuai dan kurang sesuai.

4. Kesimpulan

Berdasarkan rancang bangun pendeteksi kebakaran dan *monitoring* berbasis *Internet of Things* menggunakan *Micro Controller Node MCU* untuk mengontrol atau memantau kebakaran pada suatu gedung berdasarkan data yang didapat dari sensor asap MQ-2 dan *flame sensor*, maka telah menghasilkan beberapa kesimpulan. Melalui *IoT Cloud sistem*[29] system pendeteksi kebakaran ini dapat memberikan peringatan dini akan kejadian kebakaran dengan hasil uji pada sensor dan *actuator* yang berfungsi secara baik. Dari hasil pengujian alat, hasil pembacaan sensor MQ-2 [30] dapat mendeteksi adanya kebocoran gas, asap, dan CO, sedangkan untuk *actuator* dan seluruh kejadian yang dideteksi melebihi dari ambang batas yang telah ditentukan maka *thinger.io* mengirimkan notifikasi melalui email dan telegram yang. Untuk system monitoringnya sendiri sudah dilakukan test dengan 3 (tiga) *tools* yaitu *webpagetest.org*, *blazemeter*, dan *GT Metric* dan hasil yang didapatkan adalah Grade B pada *GT Metric* dengan maksimal user yang dapat mengakses 20 (dua puluh) User. Cetak Log Sensor dapat memanggil atau mencetak data sensor yang dikirimkan oleh MCU [20],[22] kepada *cloud service* [4],[29],[31] yang akhirnya data tersebut dapat digunakan sebagai analisa kejadian kebakaran. Membuat tampilan *dashboard* menarik yang berfungsi untuk memudahkan pengguna melihat nilai sensor pada *device* atau *control device* secara *real time*. Hasil *evaluasi* terhadap pengujian alat adalah responden menjawab cukup sesuai adalah 1,43%, sesuai 28,57 %, dan sangat sesuai 70% selebihnya 0% dengan jawaban tidak sesuai dan kurang sesuai.

Referensi

- [1] DKI, "Data Jumlah Peristiwa Kebakaran Menurut Bulan dan Kabupaten Kota Di Provinsi DKI Jakarta Tahun 2019 - Open Data Jakarta." <https://data.jakarta.go.id/dataset/data-jumlah-peristiwa-kebakaran-menurut-bulan-dan-kabupaten-kota-di-provinsi-dki-jakarta/resource/e78c7340a80d5cbe0fa3c4210e5a4656> (accessed Jun. 21, 2022).
- [2] DKI, "Kejadian Kebakaran di DKI Jakarta Tahun 2020 - Unit Pengelola Statistik." <https://statistik.jakarta.go.id/kejadian-kebakaran-di-dki-jakarta-tahun-2020/> (accessed Nov. 28, 2021).
- [3] I. Santoso, M. Farid Adiwisastro, B. Kelana Simpony, D. Supriadi, and D. Silvi Purnia, "Implementasi NodeMCUDalam Home Automation Dengan Sistem Kontrol Aplikasi Blynk," *Jurnal Swabumi*, vol. 9, no. 2, p. 2021, 2021.
- [4] A. L. Bustamante, M. A. Patricio, and J. M. Molina, "Thinger.io: An open source platform for deploying data fusion applications in IoT environments," *Sensors (Switzerland)*, vol. 19, no. 5, 2019.
- [5] D. Indriani, "Sistem Alarm Kebakaran Berbasis Arduino Menggunakan Flame Detector Dan Sensor MQ-2," *Pedagogos (Jurnal Pendidikan)*, no. Vol 3 No 2 (2021): Pedagogos: Jurnal Pendidikan, pp. 16–23, 2021.
- [6] M. Imamuddin and Z. Zulwisli, "Sistem Alarm Dan Monitoring Kebakaran Rumah Berbasis NodeMCU Dengan Komunikasi Android," *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, vol. 7, no. 2, pp. 40–45, Jun. 2019.
- [7] M. Iman, "Penggunaan Ekosistem IoT di Indonesia Diprediksi Bakal Meningkat." <https://www.goodnewsfromindonesia.id/2021/02/27/penggunaan-iot-di-indonesia-pada-2021-diprediksi-meningkat> (accessed Jan. 13, 2022).

- [8] Thinger, "Thinger.io – Open Source IoT Platform." <https://thinger.io/> (accessed Jan. 05, 2022).
- [9] J. M. S. Waworundeng, "Desain Sistem Deteksi Asap dan Api Berbasis Sensor, Mikrokontroler dan IoT," *CogITo Smart Journal*, vol. 6, no. 1, 2020.
- [10] F. Behmann and K. Wu, *Collaborative internet of things (C-IOT): For future smart connected life and business*. 2015. doi: 10.1002/9781118913734.
- [11] Dataflair, "How IoT Works - 4 Main Components of IoT System - DataFlair." <https://dataflair.training/blogs/how-iot-works/> (accessed Jan. 11, 2022).
- [12] IoT, "IoT Explained — How Does an IoT System Actually Work? | by Calum McClelland | IoT For All | Medium." <https://medium.com/iotforall/iot-explained-how-does-an-iot-system-actually-work-e90e2c435fe7> (accessed Jan. 11, 2022).
- [13] O. M. Prabowo, "Pembatasan Definisi Things Dalam Konteks Internet of Things Berdasarkan Keterkaitan Embedded System dan Internet Protocol," *Journal of Information Technology*, vol. 1, no. 2, 2019.
- [14] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.
- [15] Oracle, "What Is the Internet of Things (IoT)?" <https://www.oracle.com/internet-of-things/what-is-iot/> (accessed Nov. 21, 2021).
- [16] I. Prasetyo, "Teknik Analisis Data Dalam Research and Development," *UNY: Fakultas Ilmu Pendidikan*, vol. 6, 2014.
- [17] Salamadian, "Metode Prototype: Kelebihan, Kekurangan & Tahapan Model | Salamadian." <https://salamadian.com/metode-prototype-prototipe-adalah/> (accessed Jun. 02, 2022).
- [18] Awwaabiin, "Studi Literatur: Pengertian, Ciri-Ciri, dan Teknik Pengumpulan Datanya," <https://Penerbitdeepublish.Com/Studi-Literatur>. 2021.
- [19] Guru99, "Prototyping Model in Software Engineering: Methodology, Process, Approach." <https://www.guru99.com/software-engineering-prototyping-model.html> (accessed Jan. 11, 2022).
- [20] T. Saputro, "Mengenal NodeMCU: Pertemuan Pertama-embeddednesia.com." <https://embeddednesia.com/v1/tutorial-nodemcu-pertemuan-pertama/> (accessed Nov. 21, 2021).
- [21] T.E. Projects, "Introduction to NodeMCU V3 - The Engineering Projects." <https://www.theengineeringprojects.com/2018/10/introduction-to-nodemcu-v3.html> (accessed Nov. 24, 2021).
- [22] Components, "NodeMCU ESP8266 Pinout, Specifications, Features & Datasheet." <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet> (accessed Nov. 28, 2021).
- [23] Guru99, "What is BLACK Box Testing? Techniques, Example & Types." <https://www.guru99.com/black-box-testing.html> (accessed Jun. 21, 2022).
- [24] Telkomseliot, "2021: The Rise of Internet of Things in Indonesia," Aug. 04, 2021. <https://telkomseliot.com/en/news-insight/2021-the-rise-of-internet-of-things-in-indonesia> (accessed Jan. 13, 2022).
- [25] V. H. Pranatawijaya, W. Widiatry, R. Priskila, and P. B. A. A. Putra, "Penerapan Skala Likert dan Skala Dikotomi Pada Kuesioner Online," *Jurnal Sains dan Informatika*, vol. 5, no. 2, 2019.
- [26] D. Kho, "Pengertian Skala Likert dan Cara Menggunakan Skala Likert," *Teknik Elektronika*, 2018.
- [27] D. Kho, "Pengertian Skala Likert," *teknikelektronika.com*. 2021.
- [28] W. Budiaji, "Skala Pengukuran dan Jumlah Respon Skala Likert (The Measurement Scale and The Number of Responses in Likert Scale)," *Ilmu Pertanian dan Perikanan*, vol. 2, no. 2, 2013.
- [29] N. Jadidya and B. TV, *Cloud of Things Architecture with Application to Smart Home Scenario*. 2016.
- [30] Components101, MQ2 Gas Sensor Pinout, Features, Equivalents & Datasheet." <https://components101.com/sensors/mq2-gas-sensor> (accessed Jan. 10, 2022).
- [31] Thinger, "Thinger.io Professional Tools – Thinger.io." <https://www.thinger.io/thinger-io-professional-tools/> (accessed Jan. 11, 2022).