

Desain dan Implementasi Smart Laboratory Berbasis IOT Menggunakan ESP32 dan Thingsboard untuk Meningkatkan Keamanan dan Keselamatan di Laboratorium Teknik Informatika POLNEP

Fitri Wibowo¹, Suheri², Muhammad Diponegoro³, Bangbang Hermanto⁴

^{1,2,3,4}Jurusan Elektro, Politeknik Negeri Pontianak, Pontianak

e-mail ¹fitri.wibowo@polnep.ac.id, ²suheri@polnep.ac.id, ³didi@polnep.ac.id,

⁴bangbang@polnep.ac.id

Abstrak

Tujuan dari penelitian ini adalah untuk merancang dan membangun sebuah sistem berbasis Internet of Things (IoT) yang dapat digunakan untuk memberikan peringatan dini (early warning) pada saat terdeteksi asap atau api di dalam gedung. Sistem juga diharapkan dapat memberikan notifikasi melalui perangkat mobile kepada teknisi dan kepala laboratorium pada saat mendeteksi ketinggian luapan air hujan yang melebihi batas tertentu. Urgensi dari penelitian ini adalah dibutuhkan sebuah sistem dengan memanfaatkan teknologi terkini sebagai upaya untuk meningkatkan keselamatan dan keamanan bagi seseorang yang berada di dalam gedung pada saat terjadi bahaya kebakaran maupun bagi keamanan peralatan yang berada di dalam laboratorium. Metodologi yang digunakan dalam penelitian ini adalah dengan menggunakan pendekatan pengembangan produk yang terdiri dari perangkat lunak (software) dan perangkat keras (hardware) dalam sebuah project. Secara garis besar tahapan-tahapan yang dilakukan adalah: (1) Mendefinisikan project goals; (2) Mendefinisikan project requirements berdasarkan project goals; (3) Mendesain arsitektur sistem; (4) Mendefinisikan alur User Experience (UX); (5) Mengidentifikasi bagian pengembangan; (6) Merakit hardware dan membuat kode program untuk setiap bagian; (7) Mengintegrasikan setiap bagian/tahapan; serta (8) Uji coba dan troubleshooting. Hardware yang akan digunakan adalah microprocessor ESP32 yang diintegrasikan dengan platform IoT Thingsboard. Microcontroller ESP32 telah berhasil digunakan sebagai node IoT yang terhubung ke beberapa sensor diantaranya environmental sensor (suhu, kelembapan) DHT22, sensor intensitas cahaya (LDR), sensor gerakan (motion PIR sensor), sensor magnetic switch reed, sensor detector asap MQ2, dan sensor ketinggian air. Pada sisi server, platform Thingsboard berhasil digunakan sebagai MQTT broker, penyaji data dalam bentuk dashboard interaktif, dan memicu alarm.

Kata kunci: IoT, ESP32, Thingsboard, MQTT

Abstract

The purpose of this research is to design and build an Internet of Things (IoT) based system that can be used to provide early warning when smoke or fire is detected in the building. The system is also expected to be able to provide notifications via mobile devices to technicians and the head of the laboratory when detecting the height of the rainwater overflow that exceeds a certain limit. The urgency of this research is the need for a system by utilizing the latest technology as an effort to improve safety and security for someone who is in the building when there is a fire hazard or for the safety of equipment in the laboratory. The methodology used in this research is to use a product development approach consisting of software and hardware in a project. Broadly speaking, the stages carried out are: (1) Defining project goals; (2) Define

project requirements based on project goals; (3) Designing the system architecture; (4) Define User Experience (UX) flow; (5) Identify the parts/stages of development; (6) Assembling hardware and creating program code for each part/stage; (7) Integrating each part/stage; and (8) testing and troubleshooting. The hardware that will be used is the ESP32 microprocessor which is integrated with the IoT Thingsboard platform. The ESP32 microcontroller has been successfully used as an IoT node that is connected to several sensors including environmental sensors (temperature, humidity) DHT22, light intensity sensors (LDR), motion sensors (motion PIR sensors), magnetic switch reed sensors, MQ2 smoke detector sensors, and sensors. water level. On the server side, the Thingsboard platform has been successfully used as an MQTT broker, presenting data in the form of an interactive dashboard, and triggering alarms.

Keywords: IoT, ESP32, Thingsboard, MQTT

1. PENDAHULUAN

Laboratorium Teknik Informatika Politeknik Negeri Pontianak (POLNEP) adalah laboratorium pendidikan yang berada di Program Studi Teknik Informatika di bawah Jurusan Teknik Elektro Politeknik Negeri Pontianak. Selain dipergunakan untuk menunjang proses belajar mengajar dalam bidang pendidikan dan pengajaran, pada perguruan tinggi laboratorium juga dimanfaatkan untuk mendukung pelaksanaan tri dharma perguruan tinggi lainnya yaitu penelitian dan atau pengabdian pada masyarakat. Oleh karena itu, agar dapat berfungsi secara baik dan optimal suatu laboratorium harus dikelola dengan sistem manajemen yang baik[1]. Saat ini Laboratorium Teknik Informatika POLNEP belum memiliki fitur keselamatan dan keamanan yang memadai. Fitur keselamatan yang dimaksud adalah pendeteksi api atau kebakaran yang dapat memberikan peringatan dini kepada seseorang yang berada di dalam gedung secara *realtime*. Selain itu gedung laboratorium Teknik Informatika POLNEP juga memiliki risiko luapan air yang dapat masuk ke dalam gedung. Dari sisi keamanan, saat ini Laboratorium Teknik Informatika hanya menerapkan kamera pengawas (CCTV). Hal-hal tersebut menimbulkan permasalahan, dari segi keselamatan tentunya menimbulkan risiko bahaya maupun kerusakan baik bagi seseorang yang sedang berada di dalam laboratorium maupun peralatan yang ada di dalam gedung laboratorium.

Tujuan dari penelitian ini adalah untuk merancang dan membangun sebuah sistem yang berbasis Internet of Things (IoT) yang dapat digunakan untuk memberikan peringatan dini (*early warning*) pada saat terdeteksi asap atau api di dalam gedung. Selain itu sistem juga diharapkan dapat memberikan notifikasi melalui perangkat *mobile* kepada teknisi dan kepala laboratorium pada saat mendeteksi ketinggian luapan air hujan yang melebihi batas tertentu. Pada sisi keamanan, sistem dapat digunakan untuk mendeteksi dan memberikan notifikasi kepada kepala laboratorium pada saat terjadi perubahan status pintu di tiga (3) ruangan yaitu pintu depan (akses utama ke dalam gedung), pintu ruang dosen, dan pintu ruang kepala laboratorium.

Urgensi dari penelitian ini adalah dibutuhkannya sebuah sistem dengan memanfaatkan teknologi terkini sebagai upaya untuk meningkatkan keselamatan dan keamanan bagi seseorang yang berada di dalam gedung pada saat terjadi bahaya kebakaran maupun bagi keamanan peralatan yang berada di dalam laboratorium. Terdapat beberapa penelitian serupa dalam 3 tahun terakhir yang pernah dilakukan oleh peneliti lain yang dapat dijadikan rujukan. Diantaranya penelitian yang dilakukan pada tahun 2018 oleh M. Poongothai et.al[2] yaitu desain dan implementasi IoT untuk *smart laboratory*. Latar belakang penelitian tersebut adalah sulitnya memantau dan mengelola penggunaan perangkat elektronik seperti lampu, proyektor, dan pendingin udara di sebuah laboratorium seiring berkembangnya kampus Coimbatore Institute and Technology (CIT) di Coimbatore, India. Tujuan utama dari penelitian tersebut adalah untuk meningkatkan efisiensi penggunaan energi listrik dengan memanfaatkan teknologi berbasis IoT. Pemanfaatan IoT dapat mengurangi effort dan partisipasi manusia dengan menerapkan *automated system* sehingga dapat meningkatkan efisiensi baik dari sisi kinerja manusia maupun penggunaan energi listrik. Pada sisi *hardware* digunakan beberapa komponen utama dalam pembuatan IoT

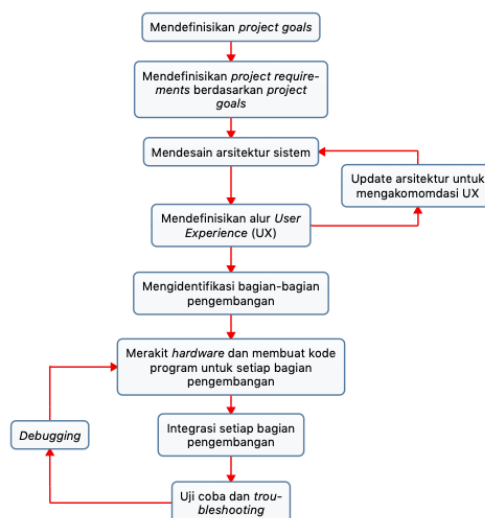
kit diantaranya ESP8266, Arduino UNO, dan Raspberry Pi3. Sedangkan pada sisi *software* penelitian tersebut menggunakan Node-RED sebagai *dashboard*, dan aplikasi Android sebagai antar muka pengguna. Efisiensi yang didapatkan setelah penerapan *smart laboratory* dengan IoT adalah sebesar 30 persen per tahun.

Penelitian selanjutnya dilakukan oleh Radu Nicolae Petraru et.al[3] pada tahun 2019 menyimpulkan penggunaan protokol yang memiliki mekanisme *publish-subscribe* seperti MQTT sangat mungkin digunakan dalam sebuah sistem pemberi peringatan pada sebuah gedung pada saat terjadi bahaya seperti misalkan kebakaran. Dalam penelitian tersebut juga telah diterapkan pemasangan *multiple node* IoT kit yaitu *node* pendeteksi asap/kebakaran, *node* pemantau variabel lingkungan yaitu *temperature* dan kelembaban, serta *node* untuk alarm. Dalam penelitian tersebut juga disimpulkan bahwa untuk *deployment* di dalam gedung, sistem memberikan waktu yang cukup bagi seseorang yang berada di dalam gedung untuk melakukan evakuasi pada saat terjadi bahaya. Adapun hardware yang digunakan adalah ESP8266, Orange Pi One sedangkan pada sisi *software* peneliti menggunakan HiveMQ pada Linux Raspbian 8 sebagai MQTT broker, dan OpenHAB yang dipasang pada sistem operasi Centos 7 sebagai IoT platform. Penelitian ketiga dilakukan oleh Sunu Dias et.al[4] bertujuan untuk mendeteksi gas berbahaya di lingkungan industri. Protokol komunikasi data yang dipakai adalah MQTT yang diintegrasikan dengan MQTT broker pada Thingsboard. Sensor gas *detector* yang digunakan adalah MQ7 dan MQ135.

Dari beberapa penelitian terakhir terkait pembuatan *smart home* maupun *smart building*, *hardware microprocessor* yang digunakan pada umumnya adalah ESP8266. Hal ini sangat wajar jika dilihat dari ketersediaan ESP8266 di pasaran yang mudah dijumpai dengan harga yang ekonomis. Dari sisi perangkat lunak, protokol MQTT lebih banyak digunakan jika dibandingkan dengan protokol lain seperti misalnya HTTP. Sedangkan untuk platform IoT, beberapa penelitian membuktikan bahwa Thingsboard dapat diandalkan sebagai platform untuk mengelola dan mengembangkan IoT[5].

Berdasarkan perbandingan teknologi yang digunakan, metode dan tujuan dari penelitian sejenis yang ada, maka pada penelitian ini penulis mencoba untuk merancang dan menerapkan sebuah *smart laboratory* berbasis IoT guna meningkatkan keamanan dan keselamatan di gedung laboratorium Teknik Informatika Politeknik Negeri Pontianak.

2. METODE



Gambar 1. Metodologi penelitian

Metodologi yang akan digunakan dalam penelitian ini adalah dengan menggunakan pendekatan pengembangan produk yang terdiri dari perangkat lunak (*software*) dan perangkat keras (*hardware*) dalam sebuah project. Gambar 1 menampilkan tahapan-tahapan pada

metodologi yang dimaksud. Jika dibandingkan dengan metodologi pengembangan software, metodologi ini memiliki beberapa langkah tambahan khususnya pada bagian perencanaan[9]. Secara garis besar tahapan-tahapan yang akan dilakukan adalah: (1) Mendefinisikan project goals; (2) Mendefinisikan project requirements berdasarkan project goals; (3) Mendesain arsitektur sistem; (4) Mendefinisikan alur User Experience (UX); (5) Mengidentifikasi bagian-bagian/tahapan-tahapan pengembangan; (6) Merakit hardware dan membuat kode program untuk setiap bagian/tahapan; (7) Mengintegrasikan setiap bagian/tahapan; serta (8) Uji coba dan troubleshooting. Selain tahapan-tahapan tersebut, terdapat aktifitas lain yang bersifat iteratif yang menghubungkan langkah ke – 4 dengan langkah ke – 3, serta langkah ke – 8 dan langkah ke – 6.

2.1 Mendefinisikan *project goals*

Tahapan ini merupakan tahapan pertama dalam metodologi penelitian yang akan digunakan di mana tujuan utama dari penelitian didefinisikan. Tujuan dari penelitian ini adalah untuk mendesain dan menerapkan sebuah smart laboratory dengan menggunakan teknologi berbasis IoT di Laboratorium Teknik Informatika Politeknik Negeri Pontianak. Sesuai dengan roadmap penelitian pada Gambar 2, pada tahun 2021 ini peneliti akan fokus pada pemanfaatan IoT untuk meningkatkan keselamatan dan keamanan di Laboratorium Teknik Informatika. Pada sisi keselamatan, peran dari teknologi IoT nantinya adalah sebagai sebuah pendeteksi jika terdapat anomali seperti misalkan asap, api, atau ketinggian air sehingga diharapkan dapat memberikan peringatan dini (early warning) kepada dosen, teknisi dan mahasiswa yang berada di dalam gedung maupun melakukan otomatisasi seperti memutus aliran listrik. Sedangkan pada sisi keamanan, teknologi IoT nantinya akan dimanfaatkan untuk mendeteksi dan memberi notifikasi secara otomatis kepada kepala laboratorium pada saat pintu laboratorium dibuka atau ditutup dan mendeteksi gerakan di luar waktu kerja.

2.2 Mendefinisikan *project requirement*

Setelah mendefinisikan tujuan dari penelitian, langkah berikutnya adalah menentukan kebutuhan project pada penelitian. Pada tahapan kedua ini, kebutuhan terkait perangkat keras (hardware) maupun perangkat lunak (software). Adapun perangkat mikroprosesor yang akan dimanfaatkan adalah ESP-32 pada sisi hardware, dan Thingsboard Community Edition sebagai perangkat lunak pada sisi server. Selain itu, kebutuhan terkait dengan sensor-sensor yang akan di gunakan untuk mencapai tujuan dari penelitian juga ditentukan. Tabel 1 memuat daftar sensor-sensor yang dibutuhkan di dalam penelitian ini.

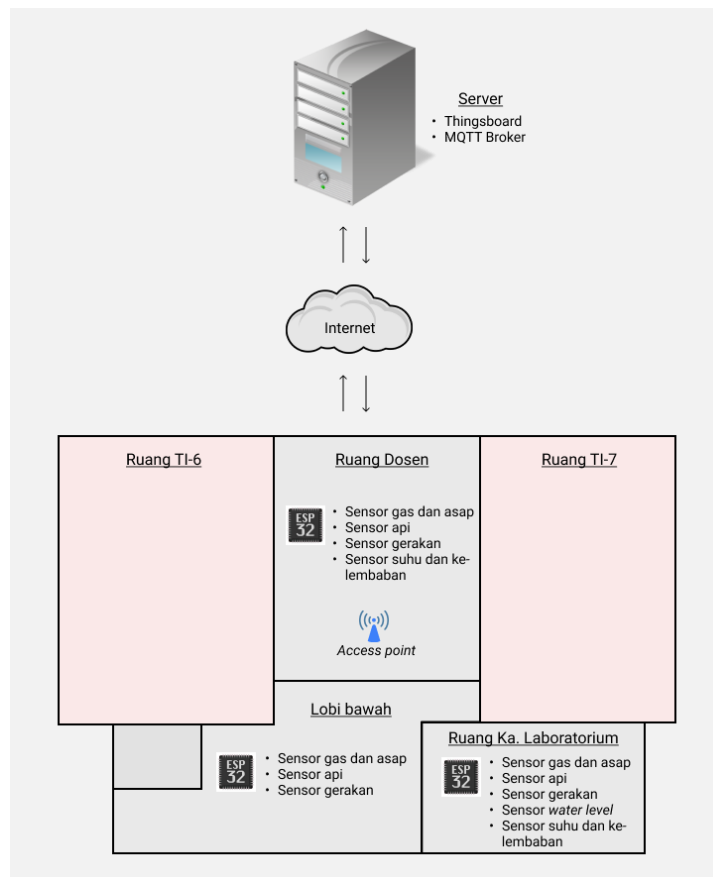
Tabel 1. Kebutuhan sensor

No.	Tipe sensor	Seri/nama produk
1.	Sensor gas dan asap	MQ2
2.	Sensor api	IR <i>flame sensor 5 channel</i>
3.	Sensor ketinggian air	<i>Water liquid level module</i>
4.	Sensor status pintu/jendela	MC38 <i>magnetic door reed switch</i>
5.	Sensor gerakan	HC-SR501 <i>motion sensor</i>
6.	Sensor suhu dan kelembaban	DHT22
7.	Sensor intensitas cahaya	LDR

2.3 Mendesain arsitektur sistem

Tahapan selanjutnya adalah mendesain arsitektur sistem. Pada penelitian ini, uji coba perangkat IoT akan dilakukan pada tiga (3) ruangan yaitu: (1) Lobi bawah Laboratorium Teknik Informatika; (2) Ruang dosen; dan (3) Ruang kepala laboratorium. Data-data yang diperoleh

dari setiap sensor pada perangkat IoT akan dikirimkan ke server Thingsboard dengan menggunakan protokol MQTT. Gambar 4 menampilkan rancangan arsitektur sistem yang saat ini sudah diidentifikasi.



Gambar 2. Rancangan arsitektur sistem

2.4 Mendefinisikan alur *User Experience (UX)*

Tahapan berikutnya adalah pendefinisian alur User Experience. Pada tahapan ini akan dibuat alur penggunaan dari sisi pengguna. Untuk saat ini pengguna yang dimaksud dalam sistem ini adalah kepala laboratorium. Kepala laboratorium memiliki akses pada dashboard aplikasi yang dapat diakses langsung melalui antarmuka pengguna Thingsboard.

2.5 Mengidentifikasi bagian-bagian pengembangan

Tahapan selanjutnya adalah mengidentifikasi bagian-bagian pengembangan pada project yang akan dibuat. Pada tahapan ini project pada penelitian ini akan dibagi menjadi beberapa bagian yang lebih kecil. Bagian-bagian ini kemudian akan diselesaikan secara terpisah dan kemudian diintegrasikan secara keseluruhan.

2.6 Merakit *hardware* dan membuat kode program

Langkah berikutnya adalah merakit *hardware* dan membuat kode program. Pada tahapan ini, bagian-bagian dari project yang telah diidentifikasi selanjutnya akan dirakit dan dibuat kode programnya. Kode program akan diimplementasikan dengan menggunakan bahasa pemrograman C dan Arduino IDE sebagai *Integrated Development Environment (IDE)*.

2.7 Integrasi setiap bagian pengembangan

Tahapan berikutnya adalah integrasi bagian-bagian pengembangan. Pada tahapan ini setiap bagian-bagian pengembangan yang telah dirakit dan dibuat kode programnya akan diintegrasikan.

2.8 Uji coba dan *troubleshooting*

Langkah terakhir adalah uji coba dan *troubleshooting*. Langkah ini dilakukan untuk memastikan bagian-bagian yang telah diintegrasikan berjalan dengan baik dan tidak terjadi permasalahan.

3. HASIL DAN PEMBAHASAN

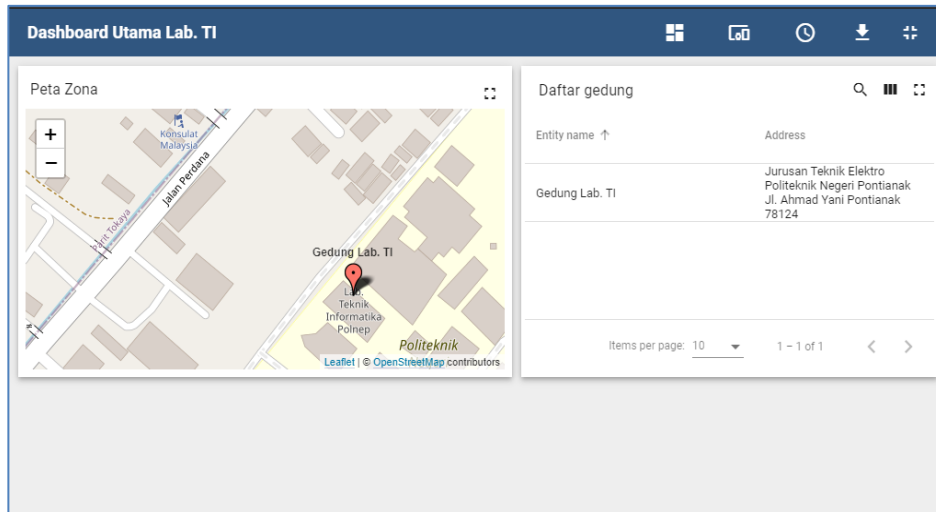
3.1 Pembuatan program dan *dashboard*

Produk *software* yang dimaksud pada penelitian ini adalah berupa kode program yang dijalankan pada microcontroller ESP32. Program tersebut digunakan untuk membaca parameter atau data pada sensor-sensor yang telah ditentukan dan kemudian mengirimkan data-data tersebut melalui internet ke server MQTT. Program tersebut dikembangkan dengan Arduino IDE dengan menambahkan *board* untuk ESP32 serta beberapa pustaka program (*library*) yang digunakan untuk berinteraksi dengan sensor maupun server MQTT. Pada sisi server MQTT, digunakan platform IoT Thingsboard yang berguna sebagai MQTT broker dan penyaji data dalam bentuk dashboard. Platform Thingsboard nantinya juga akan dikonfigurasi untuk dapat meneruskan alarm melalui pesan Telegram/Whatsapp jika node mendeteksi kejadian tertentu seperti adanya indikasi kebakaran (sensor mendeteksi asap), terjadi luapan air yang memasuki area gedung, atau perubahan status pintu diluar jam kerja. Error! Reference source not found. menyajikan daftar *board* dan *library dependencies* program pada Arduino IDE.

Tabel 2. *Board* dan *library dependencies* pembuatan program

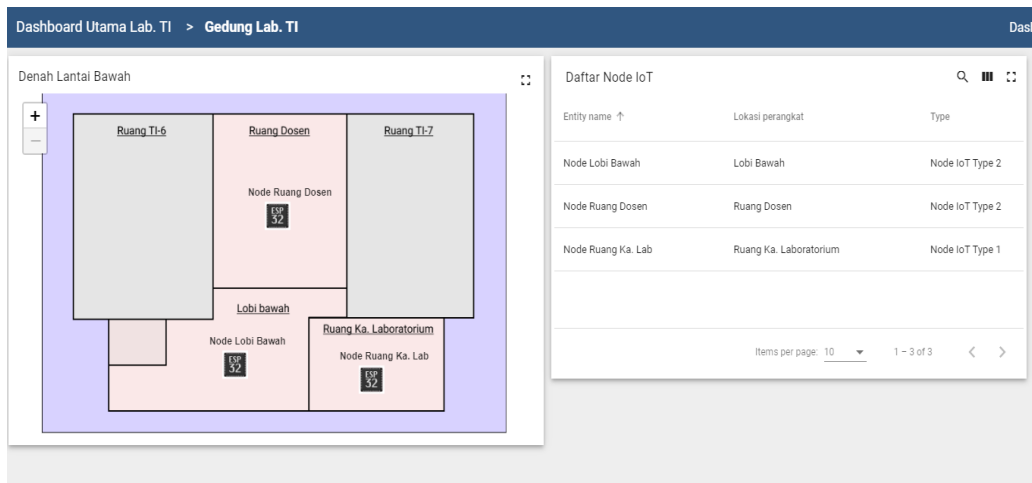
No.	<i>Board/library</i>	URL konfigurasi/repository
1.	ESP32 dev board	https://dl.espressif.com/dl/package_esp32_index.json
2.	Adafruit Unified Sensor	https://github.com/adafruit/Adafruit_Sensor
3.	ArduinoHttpClient	https://github.com/arduino-libraries/ArduinoHttpClient
4.	ArduinoJson	https://github.com/bblanchon/ArduinoJson
5.	DHT Sensor library	https://github.com/adafruit/DHT-sensor-library
6.	PubSubClient	https://github.com/knolleary/pubsubclient
7.	Thingsboard	https://github.com/thingsboard/thingsboard-arduino-sdk

Pada sisi server digunakan sistem operasi Centos Linux versi 7.8. Pada sistem operasi tersebut telah dipasang platform IoT Thingsboard. Thingsboard mendukung beberapa tipe penyimpanan data telemetri dari perangkat IoT[10]. Pada penelitian ini, digunakan media penyimpanan berbasis SQL dengan memanfaatkan *database management system* (DBMS) *open-source* PostgreSQL. Tampilan antarmuka pengguna berupa *dashboard* telah berhasil dibuat untuk menampilkan data-data sensor perangkat IoT yang berada di gedung Laboratorium Teknik Informatika yang dapat diakses melalui alamat URL <http://iot.elektro.polnep.ac.id>. Error! Reference source not found. menampilkan tangkapan layar tampilan awal *dashboard* tersebut.

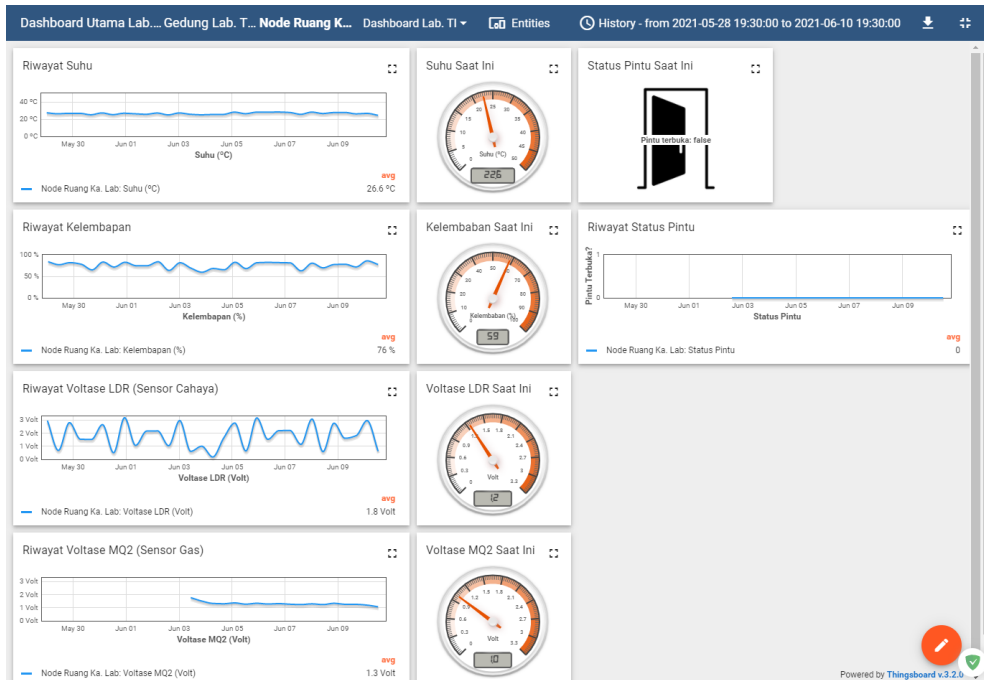


Gambar 3. Tampilan awal dashboard monitoring node IoT gedung Laboratorium Teknik Informatika

Tampilan awal dashboard menyajikan *widget map* dan tabel yang menampilkan lokasi dari gedung Lab. Teknik Informatika. Jika pin peta atau entity name pada *widget* tabel diklik, maka akan menampilkan daftar node IoT yang ada di gedung Lab. Teknik Informatika. Pada tampilan tersebut, digunakan *widget* yang menampilkan gambar denah lantai dan *widget* tabel. Setiap *node* IoT disimbolkan dengan icon ESP32 dan dapat diklik untuk menampilkan detail data pada *node* tersebut. Tampilan *dashboard* tersebut dapat dilihat pada Gambar 4.



Gambar 4. Tampilan dashboard daftar node IoT

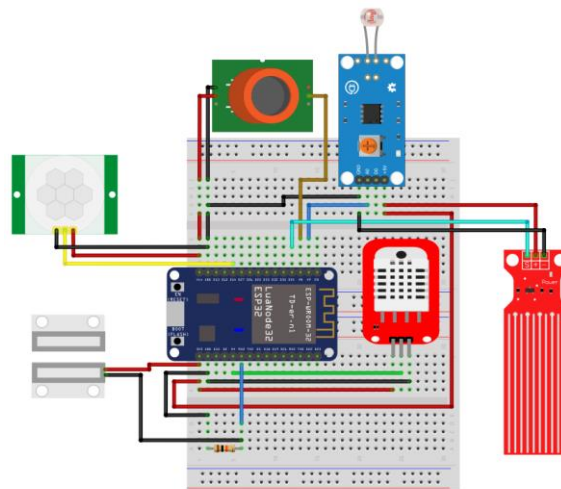


Gambar 5. Dashboard detail node IoT di ruang Kepala Laboratorium Teknik Informatika

Dashboard detail menampilkan data timeseries dari sensor-sensor pada node IoT tersebut. Beberapa tipe widget yang digunakan diantaranya, timeseries plot, dial gauge, dan widget image. Gambar 5 menampilkan dashboard detail node pada IoT kit.

3.2 Perakitan hardware

Adapun produk *hardware* pada penelitian ini adalah berupa IoT kit yang berada pada setiap node/titik lokasi ruangan. Setiap node IoT kit menggunakan sebuah *microcontroller* ESP32 dan dilengkapi dengan sensor-sensor seperti yang disajikan pada gambar 6. Node IoT yang sejauh ini sudah diimplementasikan adalah IoT kit pada ruang Kepala Laboratorium menampilkan wiring sensor pada microcontroller ESP32 node IoT kit ruang Kepala Laboratorium.



Gambar 6. Rancangan wiring sensor dengan ESP32 pada IoT kit

4. KESIMPULAN DAN SARAN

Kesimpulan yang dapat ditarik pada penelitian ini adalah: 1.) Pada penelitian ini microcontroller ESP32 berhasil digunakan sebagai node IoT yang terhubung ke beberapa sensor sekaligus karena memiliki GPIO yang cukup untuk menangani 7 sensor sekaligus. Adapun sensor-sensor yang berhasil dihubungkan sejauh ini adalah environmental sensor (suhu, kelembapan) DHT22, sensor intensitas cahaya (LDR), sensor gerakan (motion PIR sensor), sensor magnetic switch reed, sensor detector asap MQ2, dan sensor ketinggian air; 2.) Pada sisi server, platform Thingsboard berhasil digunakan sebagai MQTT broker dan penyaji data dalam bentuk dashboard interaktif. Selain itu Thingsboard juga dapat digunakan untuk pembuatan alarm berdasarkan nilai data yang dikirimkan dari node IoT. Alarm tersebut selanjutnya dapat dikonfigurasi untuk diteruskan menjadi notifikasi dalam bentuk email maupun pesan singkat Telegram atau Whatsapp.

DAFTAR PUSTAKA

- [1] B. Maftuh, "Pedoman Pemilihan Pranata Laboratorium Pendidikan (PLP) Berprestasi Tahun 2018." Dirjen Sumber Daya IPTEK, dan Pendidikan Tinggi, DIKTI, 2018.
- [2] M. Poongothai, P. M. Subramanian, and A. Rajeswari, "Design and implementation of IoT based smart laboratory," *2018 5th Int. Conf. Ind. Eng. Appl. ICIEA 2018*, pp. 169–173, 2018, doi: 10.1109/IEA.2018.8387090.
- [3] R. N. Pietraru, L. Gabriela Zegrea, and A. D. Ionita, "Publish-Subscribe Deployment Alternatives for Scenarios Related to University Laboratory Safety," *2019 11th Int. Symp. Adv. Top. Electr. Eng. ATEE 2019*, pp. 1–6, 2019, doi: 10.1109/ATEE.2019.8724923.
- [4] S. Dias, W. Kurniadi, M. Hannats, H. Ichsan, and G. E. Setyawan, "Sistem Monitoring Kadar Gas Berbahaya Di Lingkungan Industri Menggunakan Protokol MQTT," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 1, pp. 945–952, 2019.
- [5] Y. E. Windarto, B. M. W. Samosir, and M. R. Assariy, "Monitoring Ruang Berbasis Internet of Things Menggunakan Thingsboard dan Blynk," *Walisongo J. Inf. Technol.*, vol. 2, no. 2, p. 145, 2020, doi: 10.21580/wjit.2020.2.2.5798.
- [6] A. Maier, A. Sharp, and Y. Vagapov, "Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things," *2017 Internet Technol. Appl. ITA 2017 - Proc. 7th Int. Conf.*, pp. 143–148, 2017, doi: 10.1109/ITECHA.2017.8101926.
- [7] Anonim, "Thingsboard Documentation," 2021. <https://thingsboard.io/docs/getting-started-guides/what-is-thingsboard/> (accessed Mar. 01, 2021).
- [8] R. A. Atmoko, R. Riantini, and M. K. Hasin, "IoT real time data acquisition using MQTT protocol," *J. Phys. Conf. Ser.*, vol. 853, no. 1, 2017, doi: 10.1088/1742-6596/853/1/012003.
- [9] P. Desai, *Python Programming for Arduino*. Packt Publishing Ltd., 2015.
- [10] T. T. Authors, "SQL vs NoSQL vs Hybrid database approach," 2021. <https://thingsboard.io/https://thingsboard.io/docs/reference/#sql-vs-nosql-vs-hybrid-database-approach> (accessed Aug. 28, 2021).