

## **GOVERNANCE OF IOT DEVICES USING NODE-RED ORCHESTRATOR AND WEB-BASED DASHBOARD**

**Dwi Atmodjo WP<sup>\*1</sup>, Richad Harianja<sup>2</sup>, Januponsa Dio Firizqi<sup>\*3</sup>, Rido Dwi Kurniawan<sup>4</sup>**

<sup>1,2</sup>Faculty of Information Technology, Perbanas Institute, Indonesia

<sup>3,4</sup>Faculty of Information Technology, Universitas Pradita, Indonesia

Email: <sup>1</sup>[dwiatmodjo@gmail.com](mailto:dwiatmodjo@gmail.com), <sup>2</sup>[richad.harianja01@perbanas.id](mailto:richad.harianja01@perbanas.id), <sup>3</sup>[januponsa.dio@pradita.ac.id](mailto:januponsa.dio@pradita.ac.id),  
<sup>4</sup>[rido.dwi@pradita.ac.id](mailto:rido.dwi@pradita.ac.id)

(Article received: August 18, 2023; Revision: October 09, 2023; published: October 22, 2023)

### **Abstract**

*High electricity bills are often caused by a lack of monitoring the use of electronic equipment in the household. One common problem is forgetting to turn off appliances that are not in use, which causes electricity to continue flowing and increases electricity bills. Currently, although there are several technologies that allow the control of appliances, their features are still limited, especially in lamps. This research focuses on developing Internet of Things (IoT) technology in the field of energy and control. One of the implementations is using a Raspberry Pi microcontroller to monitor the use of electrical energy in households. In order to make the use of IoT components more efficient and easier, a NODE-RED-based system is used which serves as a link between end users and IoT equipment. For monitoring IoT components, a web dashboard using the Laravel framework was also built. This allows users to view the components used and the activities that occur on their IoT equipment. High electricity bills are often caused by a lack of monitoring the use of electronic equipment in the household. One common problem is forgetting to turn off appliances that are not in use, which causes electricity to continue flowing and increases electricity bills. Currently, although there are several technologies that allow the control of appliances, their features are still limited, especially in lamps. This research focuses on developing Internet of Things (IoT) technology in the field of energy and control. One of the implementations is using a Raspberry Pi microcontroller to monitor the use of electrical energy in households. In order to make the use of IoT components more efficient and easier, a NODE-RED-based system is used which serves as a link between end users and IoT equipment. For monitoring IoT components, a web dashboard using the Laravel framework was also built. This allows users to view the components used and the activities that occur on their IoT equipment.*

**Keywords:** *IoT, Laravel Node-RED, Prototype, Raspberry microcontroller, Smarthome.*

## **TATA KELOLA PERANGKAT IOT MENGGUNAKAN ORKESTRATOR NODE-RED DAN DASHBOARD BERBASIS WEB**

### **Abstrak**

Tagihan listrik yang tinggi seringkali disebabkan oleh kurangnya pemantauan penggunaan peralatan elektronik di rumah tangga. Salah satu masalah umum adalah lupa mematikan peralatan yang tidak digunakan, yang menyebabkan arus listrik terus mengalir dan meningkatkan tagihan listrik. Saat ini, meskipun ada beberapa teknologi yang memungkinkan pengendalian peralatan, fiturnya masih terbatas, terutama pada lampu. Penelitian ini fokus pada pengembangan teknologi Internet of Things (IoT) di bidang energi dan kontrol. Salah satu implementasinya adalah menggunakan mikrokontroler Raspberry Pi untuk memantau penggunaan energi listrik di rumah tangga. Dalam rangka membuat penggunaan komponen IoT lebih efisien dan mudah, digunakan sistem berbasis NODE-RED yang berfungsi sebagai penghubung antara pengguna akhir dan peralatan IoT. Untuk pemantauan komponen IoT, juga dibangun sebuah dashboard web menggunakan framework Laravel. Ini memungkinkan pengguna melihat komponen yang digunakan dan aktivitas yang terjadi pada peralatan IoT mereka. Metode prototyping digunakan dalam penelitian ini, memungkinkan pengembang menguji berulang-ulang hingga mencapai prototipe yang dapat diterima. Hasil pengujian menunjukkan bahwa pengguna dapat mengendalikan penggunaan daya listrik dengan mudah melalui dashboard UI yang sederhana. Selain itu, sistem ini memudahkan pengguna dalam membatasi penggunaan listrik tanpa harus mematikan peralatan secara manual. Dengan solusi ini, diharapkan dapat membantu mengatasi masalah penggunaan listrik yang tidak efisien dan memudahkan pengguna yang mungkin kesulitan dengan pemrograman dan bahasa pemrograman terkait IoT. Dengan dashboard berbasis web yang ramah pengguna, pengelolaan peralatan IoT menjadi lebih mudah dan

efisien. Solusi ini memiliki potensi besar untuk membantu rumah tangga mengurangi tagihan listrik yang tinggi dan meningkatkan kesadaran akan penggunaan energi.

**Kata kunci:** *IoT, Laravel, Node-RED, Prototype, Raspberry microcontroller, Smarthome.*

## 1. PENDAHULUAN

Tagihan listrik yang besar atau membengkak pada rumah tangga sering terjadi karena penggunaan alat-alat elektronika yang tidak terpantau dengan baik [1]. Hal yang sangat sering sekali terjadi adalah lupa untuk mematikan daya pada alat elektronika yang tidak digunakan yang menyebabkan arus listrik akan terus mengalir dan akan membuat tagihan listrik membengkak [2]. Penelitian ini mengembangkan teknologi dibidang IoT yang bergerak dibidang energi dan control, salah satu pengembangan alat elektronika adalah membuat implementasi sistem IoT pada pemantauan energi listrik rumah tangga dengan menggunakan microcontroller. Microcontroller sendiri adalah *integrated circuit* yang mampu melakukan penulisan, penghapusan sampai dengan pembacaan suatu perintah [3]. Pemantauan energi listrik rumah tangga ini menggunakan Raspberry pi dengan pemrograman Node-RED untuk memantau energi yang telah digunakan pada rumah. Raspberry pi sendiri merupakan salah satu *single board Computer* (SBC) yang cukup populer dan bersifat *open source* (berbasis Linux), juga bisa dimodifikasi sesuai kebutuhan penggunaannya [4].

*Internet of Things* atau dikenal juga dengan singkatan IoT adalah segala sesuatu atau perangkat elektronik yang dapat berinteraksi langsung dengan pengguna yang digunakan untuk kebutuhan monitoring ataupun kontrol pada perangkat tersebut melalui internet [5]. Dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer [6]. Cara kerja *Internet of Things* yaitu dengan memanfaatkan sebuah argumentasi pemrograman yang dimana setiap perintah yang terdapat pada argumennya menghasilkan sebuah interaksi antara sesama mesin yang terhubung secara otomatis tanpa campur tangan manusia [7]

Terdapat beberapa komponen yang mendukung IoT diantaranya: sensor, motor, actuator dan microprocessor. Perangkat ini berfungsi untuk mengumpulkan data, menjalankan aksi tertentu, dan mengontrol perangkat lain. Secara umum komponen tersebut dikembangkan untuk dapat melakukan suatu fungsi tertentu dan bekerja secara independen [8].

Agar dapat digunakan dan saling bertukar data, perangkat tersebut harus dijalankan dengan menggunakan kode program atau membutuhkan *programming*. *Programming* tersebut berfungsi sebagai penghubung dari aplikasi yang digunakan oleh *end-user* untuk melakukan fungsi-fungsi terhadap alat pada IoT yang terinstall. Jika terjadi

perubahan pada sistem, misalnya penambahan perangkat baru, maka kode program juga harus diperbaharui.

Untuk dapat melakukan monitoring terhadap semua alat IoT juga dibutuhkan sebuah dashboard yang nantinya akan di gunakan oleh *end-user* atau pengguna untuk dapat melihat apa saja komponen yang digunakan dan aktivitas yang terjadi pada komponen-komponen IoT yang terinstall.

Permasalahan diatas juga mendasari penulis untuk mencoba mengembangkan sebuah sistem yang dapat mengintegrasikan setiap perangkat tersebut dengan cara yang lebih sederhana tanpa menggunakan kode program yang rumit, pengguna akan mengatur cara perangkat tersebut bekerja dengan membuatnya dalam bentuk rangkaian flow [9]. Sistem yang dikembangkan akan mampu melakukan otomatisasi observasi atau aksi apapun, sesuai rangkaian flow yang dibuat oleh pengguna dan juga mampu memberikan kemampuan kefleksibelan penggunaan perangkat sesuai kebutuhan pengguna. Dengan kemampuan ini, permasalahan terkait dengan pengubahan kode program jika keadaan sistem berubah akan dapat diselesaikan. Lalu dengan adanya dashboard berbasis web akan mempermudah pengguna [10]. Pengguna akan dimudahkan untuk melihat data komponen IoT yang sedang digunakan dan memantau setiap aktivitas yang dilakukan terhadap komponen-komponen IoT.

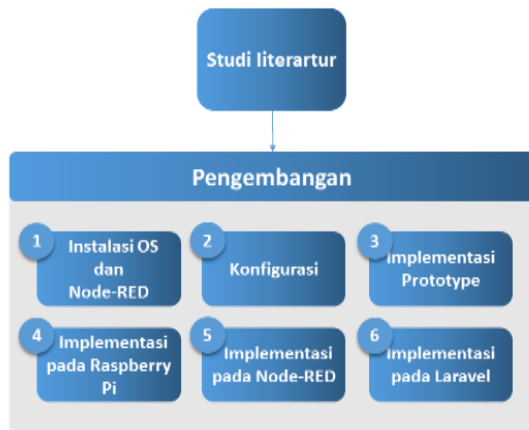
Metode yang digunakan dalam penyelesaian kasus ini menggunakan metode prototyping di mana pengembang membangun sebuah prototype yang akan diuji dan dikerjakan ulang sampai prototype yang dapat diterima tercapai [11]. Metode ini menciptakan basis untuk menghasilkan sistem yang paling baik [12]. Metode ini bersifat iterative atau proses yang dikerjakan berulang dan dinamis [13].

Dengan adanya sistem ini, diharapkan dapat membangun sistem terintegrasi yang dapat mengintegrasikan perangkat IoT yang memiliki API *endpoint*. Sistem akan memberikan kemampuan dalam memudahkan penggunaan perangkat IoT dengan membuatnya sebagai rangkaian *flow*. Mengimplementasikan penggunaan perangkat lunak Node-RED merupakan sebuah tool untuk membuat aplikasi *Internet of Things* (IoT) yang mana lingkungan pemrograman visualnya mempermudah penggunaannya untuk membuat aplikasi sebagai "*flow*" [14]. Sebagai aplikasi untuk membuat flow dan antarmuka pengguna dengan sistem [15]. Memberikan solusi penggunaan perangkat IoT secara lebih mudah melalui rangkaian flow yang dihasilkan melalui dashboard dengan menggunakan node yang ditambahkan *developer* untuk perangkat IoT. Yang

membantu pengguna dalam mengoperasikan perangkat IoT dengan lebih mudah dan sederhana tanpa menggunakan kode program yang rumit dan dapat memantau penggunaan komponen IoT dengan dashboard berbasis web yang *user friendly*.

**2. METODE PENELITIAN**

Metode yang digunakan dalam penelitian dengan menggunakan model prototype. Model prototyping merupakan metode pengembangan sistem dengan mengembangkan sebuah prototype untuk membantu dalam mendapatkan gambaran lebih rinci tentang spesifikasi sistem [16].



Gambar 1. Tahapan Penelitian

Seperti yang terlihat pada gambar 1 tahapan pelaksanaan penelitian dimulai dari beberapa studi literatur terkait penelitian sejenis dan teknologi yang akan digunakan dalam penelitian dan rancangan dari sistem yang akan dikembangkan. Pada setiap tahapan pengembangan akan dilakukan pengujian untuk memastikan sistem yang dikembangkan telah berjalan dengan baik dan benar. Tahapan pengembangan terbagi menjadi:

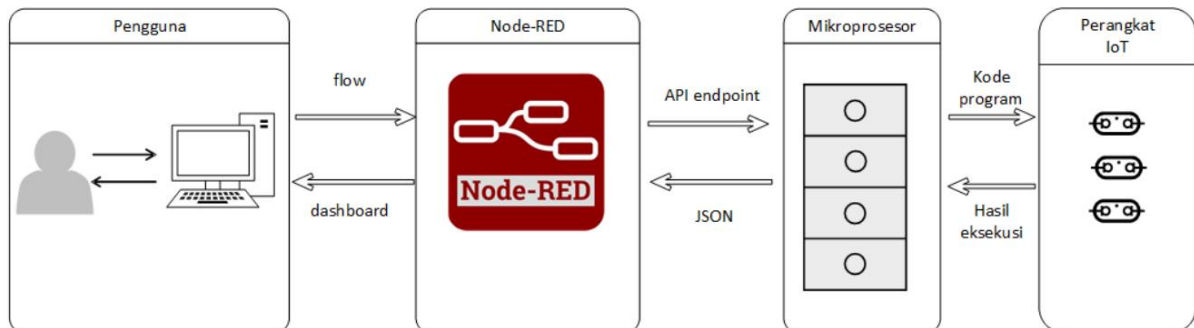
1. Instalasi menjelaskan tentang proses persiapan instalasi yang dilakukan dalam pengerjaan

implementasi IoT. Instalasi yang dilakukan seperti instalasi sistem operasi Raspberry Pi, dan instalasi Node-RED.

2. Konfigurasi menjelaskan tentang konfigurasi yang dilakukan terhadap instalasi yang dilakukan pada proses sebelumnya. Konfigurasi dilakukan agar software yang sudah diinstal dapat berjalan sesuai fungsi dan kebutuhan. Konfigurasi yang dilakukan pada software yang telah diinstal, seperti konfigurasi sistem operasi. Konfigurasi Sistem Operasi Raspberry Pi.
3. Implementasi Prototype menjelaskan tentang implementasi yang dilakukan dalam pembuatan prototype. Pada prototype terdapat sembilan slot, slot tersebut digunakan untuk mempermudah pengguna dalam merangkai perangkat IoT dan tidak harus mengerti dengan GPIO pin pada Raspberry Pi tersebut. Pengguna memilih slot yang sesuai dengan jumlah pin yang dibutuhkan terhadap perangkat IoT tersebut dan menyesuaikan setiap pin dari perangkat IoT terhadap pin yang terdapat pada slot yang ingin digunakan.
4. Implementasi Pada Raspberry Pi menjelaskan tentang implementasi yang dilakukan pada RPi3 pada proses pengerjaan project penelitian ini.
5. Implementasi pada Node-RED menjelaskan tentang implementasi yang dilakukan berupa pembuatan rangkaian perangkat IoT pada Raspberry Pi dan implementasi rangkaian tersebut terhadap Node-RED.
6. Implementasi pada Laravel menjelaskan tentang informasi yang tercakup dalam platform web-based yang digunakan sebagai pengontrol rangkaian listrik otomatis.

**2.1. High Level Sistem Overview**

High level sistem dari Tata Kelola Perangkat IoT Menggunakan orkestrator Node-RED dan Dashboard Berbasis Web secara umum dapat dilihat pada gambar berikut.



Gambar 2. High Level Sistem Overview

Pada gambar 2 dapat dilihat pengguna mengakses dashboard Node-RED melalui web browser pada laptop/pc. Pada dashboard, pengguna dapat menggambarkan flow yang dibutuhkan menggunakan node yang disediakan oleh Node-RED

maupun dengan custom node. Setelah pengguna selesai membuat rangkaian flow dan menyimpannya pada server, server akan mulai menjalankan flow tersebut sesuai dengan flow yang telah didefinisikan pengguna pada dashboard. Node-RED

berkomunikasi dengan perangkat IoT melalui jaringan lokal menggunakan API endpoint yang disediakan oleh Raspberry Pi. Jika ada permintaan untuk mengakses perangkat IoT melalui API endpoint tersebut, Raspberry Pi kemudian mengeksekusi kode program yang sesuai dengan API endpoint yang diminta. Setelah kode program selesai dieksekusi, perangkat yang bersangkutan kemudian mengirimkan data hasil pengukuran maupun status eksekusi kembali ke Raspberry Pi, mengolah nya lalu mengirim respon ke Node-RED berupa data berformat JSON.

Pada sistem yang dibangun memiliki tiga fungsi utama yang saling mendukung, yaitu kontrol flow, transmisi data, dan kontrol perangkat. Proses yang terjadi pada sistem secara umum yaitu:

1. Kontrol Flow

Kontrol flow terjadi antara pengguna dengan Node-RED. Pengguna dapat mengakses dashboard Node-RED menggunakan browser yang mendukung javascript. Pada dashboard Node-RED, pengguna dapat membuat flow dibutuhkan dengan memanfaatkan node yang tersedia pada dashboard, baik dengan node bawaan Node-RED maupun dengan custom node yang telah dibuat. Jika pada flow yang digambarkan memiliki node httprequest yang melakukan request ke API endpoint, proses kemudian masuk ke bagian transmisi data.

2. Transmisi Data

Proses transmisi data terjadi antara Node-RED dengan Raspberry Pi melalui jaringan lokal. Proses akan masuk ke bagian transmisi data jika pada flow terdapat node http-request dan mengarah ke API endpoint perangkat yang telah didefinisikan pada

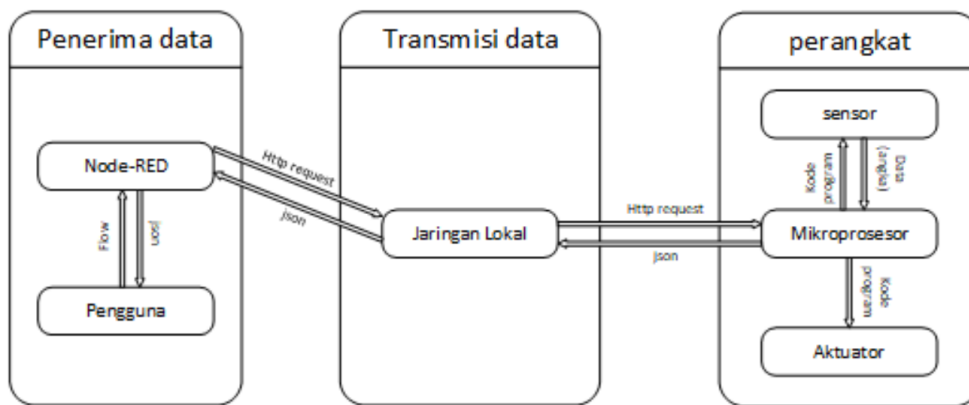
Raspberry Pi. Ketika ada request, Node-RED melalui node http-request mengirimkan request ke API endpoint yang dituju dan menunggu respon dari Raspberry Pi. Respon yang dikirimkan oleh Raspberry Pi berformat json. Jika respon telah dikirimkan oleh Raspberry Pi, respon kemudian diteruskan kembali ke rangkaian flow.

3. Kontrol Perangkat

Proses kontrol perangkat terjadi antara Raspberry Pi dengan perangkat IoT. Terdapat beberapa komponen yang berperan pada bagian kontrol perangkat, yaitu perangkat IoT, serta API endpoint dan kode program, yang ditulis dalam Bahasa program Python. API endpoint berfungsi untuk menerima request dan mengirim respon kembali ke pengirim request. Kode program berfungsi untuk menginstruksikan perangkat untuk menjalankan aksi yang sesuai dengan dengan blok kode yang telah didefinisikan. Proses memasuki bagian ini jika ada permintaan dari Node-RED melalui API endpoint. Ketika ada request, Raspberry Pi kemudian mengidentifikasi alamat API endpoint yang diminta, kemudian mengeksekusi blok kode program yang sesuai dengan perangkat yang diminta. Hasil dari eksekusi kode program tersebut kemudian dikirimkan kembali ke flow Node-RED.

2.2. Deskripsi Pertukaran Data

Deskripsi pertukaran data pada sistem Tata Kelola Perangkat IoT Menggunakan orkestrator Node-RED dan Dashboard Berbasis Web dapat dilihat pada gambar berikut.



Gambar 3. Deskripsi Pertukaran Data

Sistem Tata Kelola Perangkat IoT Menggunakan orkestrator Node-RED dan Dashboard Berbasis Web merupakan sistem terintegrasi yang menggunakan perangkat perangkat sistem tertanam seperti sensor dan actuator, yang dapat digunakan untuk memantau keadaan lingkungan sekitar, dan menjalankan aksi tertentu sesuai fungsi perangkat yang bersangkutan. Perangkat tersebut disematkan pada mikroprosesor agar dapat bekerja. Dalam menjalankan setiap perangkat, mikroprosesor

menggunakan kode program yang telah ditulis dan dioptimalkan untuk dapat berjalan sesuai fungsinya

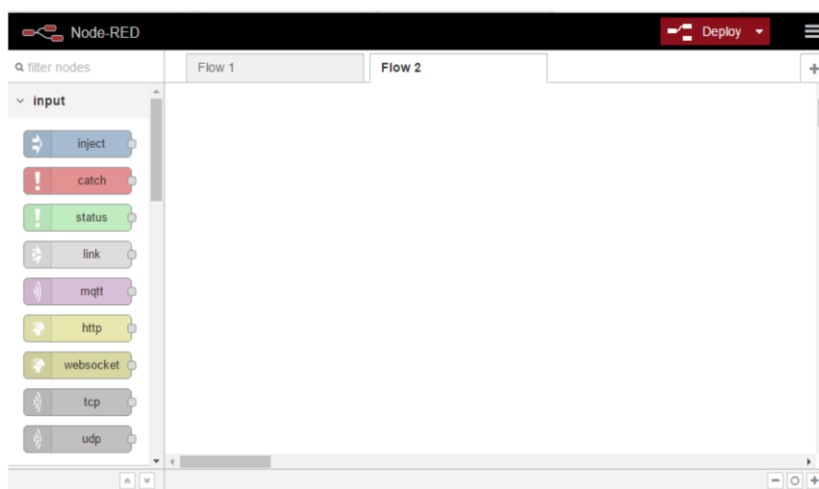
Pada gambar 3 dapat dilihat Node-RED diakses oleh pengguna lalu membuat flow menggunakan node-node yang tersedia pada dashboard Node-RED. Kemudian Node-RED mengirimkan request berupa http request menggunakan metode GET ke mikroprosesor menggunakan jaringan lokal yang digunakan. Jaringan lokal berfungsi sebagai penghubung antara Node-RED dengan

mikroprosesor. Jika http request telah diterima mikroprosesor, maka mikroprosesor akan menjalankan kode program yang diminta oleh pengguna berdasarkan API endpoint yang dikirimkan melalui http request. Jika yang dijalankan adalah sensor maka akan menerima hasil dari kode program yang dijalankan. Contoh pada sensor jarak, jika sensor jarak dijalankan maka pengguna akan menerima data dari pengukuran jarak yang dilakukan oleh sensor tersebut. Jika pada kode program aktuator dijalankan maka akan memberikan berupa respon kepada pengguna. Contoh penggunaan aktuator jenis servo, jika kode program servo dijalankan maka servo akan bergerak. Setelah itu mikroprosesor akan

memberikan hasil berupa format JSON yang akan dikirimkan melalui jaringan local kepada pengguna.

### 2.3. Custom Node pada Node-RED

Dashboard Node-RED pada sistem yang dibangun memiliki 2 fungsi utama yaitu sebagai antarmuka dengan rangkaian flow yang dapat dibuat oleh pengguna, dan sebagai perangkat penengah antara pengguna dengan perangkat IoT. Fungsi antarmuka tersedia melalui dashboard yang disediakan oleh Node-RED untuk membuat flow. Dashboard dapat dilihat pada gambar 4 berikut.



Gambar 4. Dashboard Node-RED

Setiap node yang disediakan oleh Node-RED memiliki fungsi khusus. Selain menggunakan node yang telah disediakan Node-RED, pengguna dapat membuat node baru, sesuai dengan fungsi yang diinginkan pengguna. Dalam membuat node baru dibutuhkan 3 file yaitu:

1. File .js
2. File .html
3. package.json

Pada dashboard Node-RED developer menambahkan node-node baru untuk mengendalikan berbagai perangkat IoT. Pada setiap node akan ada form untuk meng-edit atau mengatur setiap perangkat IoT tersebut agar sesuai dengan rangkaian pada raspberry pi. Adapun node-node yang akan ditambahkan pada dashboard Node-RED, diantaranya:

1. Node ITDX Ping  
Node ITDX Ping digunakan untuk mendeteksi jarak dari suatu objek yang berada didepan sensor ping tersebut.
2. Node ITDX Buzzer  
Node ITDX Buzzer pada Node-RED digunakan untuk mengubah getaran listrik menjadi getaran suara.
3. Node ITDX DHT11  
Node ITDX DHT11 dibangun untuk mendeteksi besar suhu dan kelembapan udara disekitar sensor tersebut.

4. Node ITDX Raspi

Node ITDX Raspi digunakan sebagai mikroprosesor. Node raspi tersebut dapat diintegrasikan beberapa raspberry pi dengan memperkenalkan setiap ip address dari raspberry pi yang akan digunakan.

5. Node ITDX Relay

Node ITDX Relay dibangun sebagai suatu komponen elektronika yang bersifat elektronis serta dapat dikendalikan dari rangkaian elektronik lainnya dengan memanfaatkan tenaga listrik sebagai sumber energinya.

6. Node ITDX Servo

Node ITDX Servo dibangun sebagai aktuator putar.

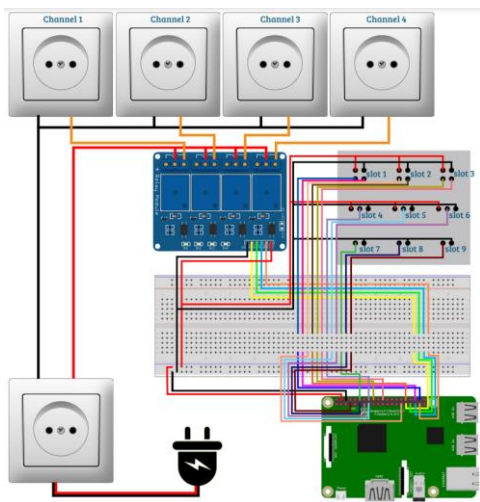
### 2.4. Desain Prototype Board

Dashboard Node-RED pada sistem yang dibangun memiliki 2 fungsi utama yaitu sebagai antarmuka dengan rangkaian flow yang dapat dibuat oleh pengguna, dan sebagai perangkat penengah antara pengguna dengan perangkat IoT. Fungsi antarmuka tersedia melalui dashboard yang disediakan oleh Node-RED untuk membuat flow. Dashboard dapat dilihat pada gambar berikut.

Perangkat IoT dapat bekerja dan berkomunikasi melalui pin GPIO yang terdapat pada board



Raspberry Pi. Agar bekerja dengan baik, pin yang terdapat pada perangkat IoT harus terhubung ke pin GPIO yang benar pada Raspberry. Jika salah memilih pin, perangkat berpotensi mengalami kerusakan. Oleh karena itu, developer mengembangkan sebuah prototype board yang dapat membantu pengguna untuk memilih pin yang sesuai dengan perangkat IoT yang dioperasikan. Prototype yang dibangun menyajikan 4 soket stop kontak untuk menghubungkan beberapa perangkat elektronik, serta 9 slot pin untuk menghubungkan perangkat IoT. Desain prototype board yang dikembangkan dapat dilihat pada gambar 5 berikut.



Gambar 5. Desain Prototype Board

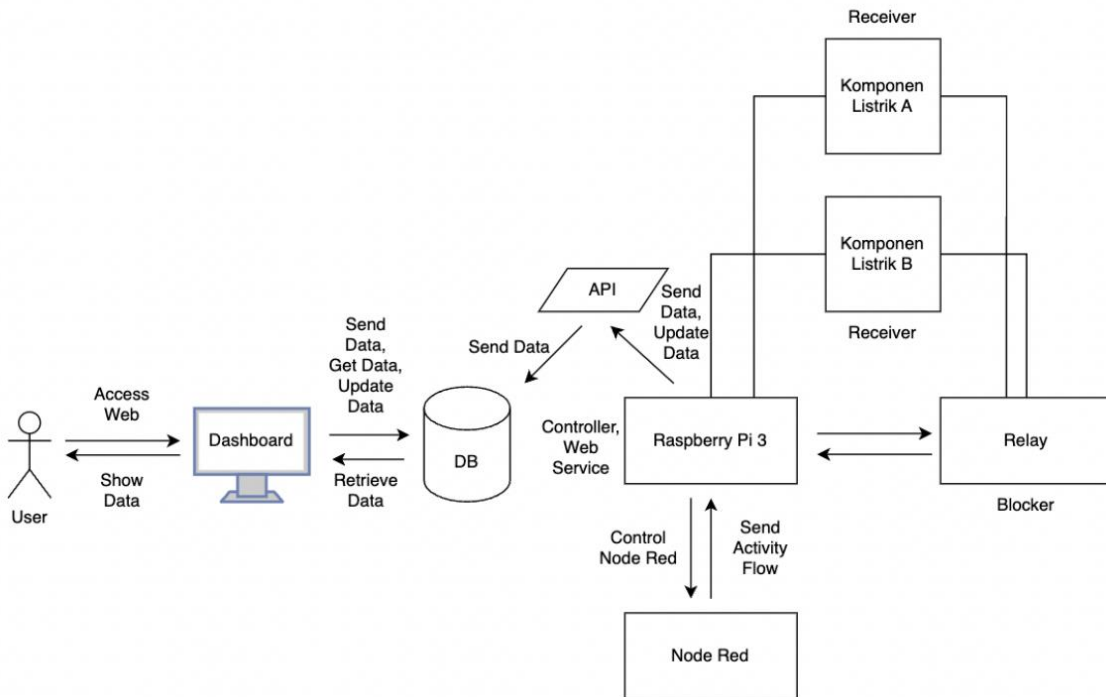
Pada Gambar 4 dapat dilihat bahwa Prototype Board menyediakan 4 soket stopkontak dan 9 slot pin.

Keempat stopkontak terhubung ke relay dimana relay berperan dalam memutuskan dan menyambungkan aliran listrik pada soket stopkontak. Selain itu terdapat juga 9 slot pin yang berguna untuk menyambungkan perangkat IoT dengan perangkat. Kesembilan slot pin tersebut dibagi menjadi 3 kategori slot yaitu:

1. Slot 4 pin yang terdiri dari 1 pin vcc, 1 pin ground, dan 2 pin GPIO
2. Slot 3 pin yang terdiri dari 1 pin vcc, 1 pin ground, dan 1 pin GPIO
3. Slot 2 pin yang terdiri dari 1 pin ground dan 1 pin GPIO

Masing masing channel relay dan slot setidaknya terhubung dengan 1 atau 2 pin GPIO pada Raspberry Pi. Informasi detail mengenai pin yang digunakan pada slot dan relay.

User akan mengakses web monitoring lalu akan mengakses dashboard sistem. Pada saat didalam dashboard sistem, user dapat melakukan proses create new user, melihat list komponen dan melihat list user yang terdaftar dalam sistem. Lalu dashboard akan melakukan get dan retrieve data ke database yang dimana database ini berada atau di install di raspberry pi yang juga bertindak sebagai web server. Selain sebagai web server RPi3 juga bertindak sebagai controller seluruh komponen electric. RPi3 juga menjadi tempat untuk Node-RED di install dan pembuatan design flow komponen listrik nya akan di rangkai didalam Node-RED tersebut. Framework python disini akan bertugas sebagai penghubung antara web server dengan database yang berfungsi untuk melakukan pengolahan data ke dalam database untuk setiap aksi yang dilakukan ke komponen listrik yang terhubung.



Gambar 6. Flow Prototype

Pada Gambar 6 dapat dilihat flow prototype yang akan dibangun pada pengerjaan penelitian kali ini. User akan mengakses web monitoring lalu akan mengakses dashboard system. Pada saat didalam dashboard system, user dapat melakukan proses create new user, melihat list komponen dan melihat list user yang terdaftar dalam system. Lalu dashboard akan melakukan get dan retrieve data ke database yang dimana database ini berada atau di install di raspberry pi yang juga bertindak sebagai web server. Selain sebagai web server RPi3 juga bertindak sebagai controller seluruh komponen electric. RPi3 juga menjadi tempat untuk Node-Red di install dan pembuatan design flow komponen listrik nya akan di rangkai didalam Node-Red tersebut. Framework python disini akan bertugas sebagai penghubung antara web server dengan database yang berfungsi untuk melakukan pengolahan data ke dalam database untuk setiap aksi yang dilakukan ke komponen listrik yang terhubung.

### 3. HASIL DAN PEMBAHASAN

Pada bab ini dijelaskan mengenai implementasi terhadap sistem yang telah dilakukan. Proses implementasi yang dilakukan berupa instalasi, konfigurasi, implementasi prototype, implementasi pada Raspberry Pi, implementasi pada laravel dan implementasi pada Node-RED.

#### 3.1. Instalasi

Terdapat beberapa hal yang perlu untuk diinstalasi, antara lain:

1. Instalasi Node-RED

Node-RED adalah software yang digunakan sebagai server yang akan mengelola data yang diterima dari perangkat IoT dan menyediakan tampilan antarmuka pengguna.

2. Instalasi Raspberry Pi

Raspberry Pi adalah software yang digunakan sebagai mikroprosesor untuk mengoperasikan perangkat IoT dan mengirimkan data ke server.

3. Instalasi Flask

Flask adalah Micro Web Framework yang ditulis menggunakan bahasa pemrograman Python [17].

4. Instalasi Laravel

Laravel adalah *framework* aplikasi web dengan sintaks yang ekspresif dan elegan [18]. Framework Laravel web ini menyediakan struktur dan titik awal untuk membuat aplikasi yang akan kita develop, memungkinkan pembangun untuk fokus menciptakan sesuatu yang luar biasa.

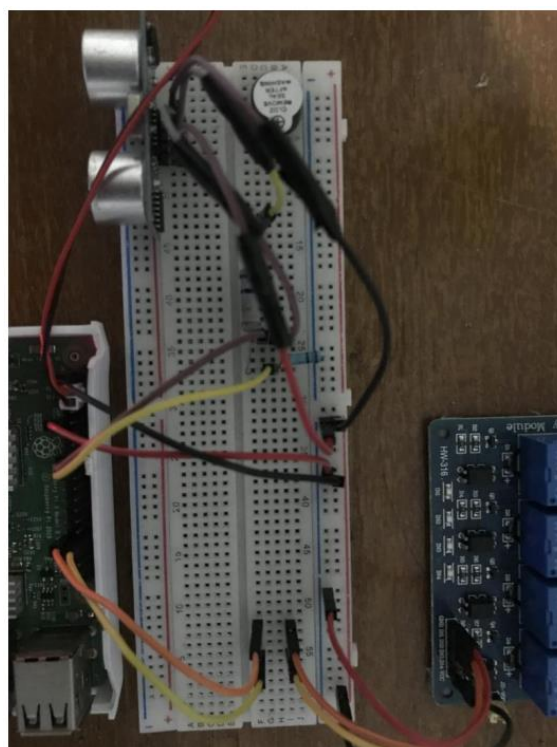
#### 3.2. Konfigurasi

Pada sub bab ini dijelaskan mengenai konfigurasi yang dilakukan terhadap instalasi yang dilakukan pada sub bab sebelumnya. Konfigurasi dilakukan agar software yang sudah diinstal dapat

berjalan sesuai fungsi dan kebutuhan. Konfigurasi yang dilakukan pada software yang telah diinstal, seperti konfigurasi sistem operasi.

#### 3.3. Implementasi Prototype

Pada sub bab ini diuraikan mengenai implementasi yang dilakukan dalam pembuatan prototype. Pada prototype terdapat beberapa slot, slot tersebut digunakan untuk mempermudah pengguna dalam merangkai perangkat IoT dan tidak harus mengerti dengan GPIO pin pada Raspberry Pi tersebut. Pengguna memilih slot yang sesuai dengan jumlah pin yang dibutuhkan terhadap perangkat IoT tersebut dan menyesuaikan setiap pin dari perangkat IoT terhadap pin yang terdapat pada slot yang ingin digunakan. Implementasi prototype dapat dilihat pada gambar 7 dibawah ini.



Gambar 7. Implementasi Prototype

Terdapat satu ekstensi pada prototype tersebut dimana channel 1 merupakan ekstensi yang terhubung ke Relay dan channel 5 digunakan untuk dapat terhubung ke Raspberry Pi.

#### 3.4. Implementasi pada Raspberry Pi

Pada sub bab ini menjelaskan implementasi yang dilakukan pada Raspberry Pi dalam proses pengerjaan Pembangunan prototype. Implementasi yang dilakukan berupa implementasi Raspberry Pi terhubung pada Node-RED.

Setelah selesai melakukan instalasi pada Raspberry Pi yang akan digunakan, maka untuk dapat terhubung terhadap Node-RED pengguna harus menjalankan perintah berikut:

1. Menjalankan perintah ini pada terminal Raspberry Pi.
2. Setelah menjalankan perintah diatas, secara otomatis akan menginstal perintah yang terdapat pada install.sh.
3. Kemudian setelah selesai menjalankan perintah diatas maka Raspberry Pi tersebut telah dikenali pada Node-RED dan telah dapat diakses pada aplikasi Node-RED tersebut.

Raspberry Pi yang telah terhubung pada Node-RED dapat dijalankan oleh pengguna dengan membuat flow menggunakan node yang telah ditambahkan pada dashboard Node-RED dan memperkenalkan ip address dari Raspberry Pi yang digunakan. Raspberry Pi akan dapat menjalankan berbagai perangkat IoT dengan merangkai perangkat IoT pada board yang telah disediakan dan mendisain flow menggunakan node-node yang telah disediakan pada dashboard Node-RED. Jika terjadi perubahan pada sistem, misalnya penambahan perangkat baru, maka pengguna hanya memperbaharui flow yang sesuai dengan perangkat yang diperlukan.

### 3.5. Implementasi pada Node-RED

Pada sub bab ini dijelaskan mengenai implementasi yang dilakukan pada Node-RED dalam proses pengerjaan penelitian. Implementasi yang dilakukan berupa pembuatan rangkaian perangkat IoT pada Raspberry Pi dan implementasi rangkaian tersebut terhadap Node-RED.

#### 3.5.1. Implementasi Node Ping Sensor pada Node-RED

Node Ping digunakan untuk mendeteksi keberadaan suatu objek tertentu yang terdapat didepan sensor Ping. Untuk menambahkan node Ping pada Node-RED. Untuk contoh rangkaian pada Ping sensor yang dijalankan menggunakan NodeRED dapat dilihat pada gambar 8 dibawah ini.

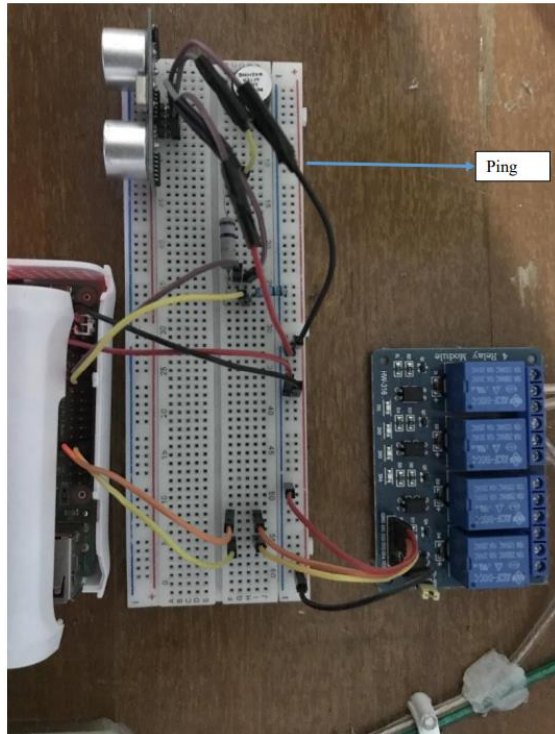
#### 3.5.2. Implementasi Node Raspi pada Node-RED

Node raspi pada Node-RED digunakan sebagai mikroprosesor. Pada node Raspberry dapat diintegrasikan beberapa Raspberry Pi hanya dengan mengetahui ip address dari setiap Raspberry Pi yang akan dihubungkan. Perangkat IoT dirangkai pada board yang telah dibangun oleh developer untuk mempermudah dalam penggunaan Raspberry Pi dan untuk dapat menjalankan perangkat tersebut melalui Node-RED harus mengetahui Ip address dari Raspberry Pi yang akan digunakan. Untuk menambahkan node Raspi pada Node-RED.

#### 3.5.3. Implementasi Node Relay pada Node-RED

Node Relay pada aplikasi yang telah dibuat sebagai suatu komponen elektronika yang bersifat elektronis dan sederhana serta tersusun oleh saklar, lilitan, dan poros besi. Relay merupakan tuas saklar

dengan lilitan kawat pada batang besi atau solenoid di dekat Relay. Ketika solenoid dialiri arus listrik, tuas akan tertarik karena adanya gaya magnet yang terjadi pada solenoid sehingga kontak saklar akan menutup.



Gambar 8. Rangkaian Ping Sensor pada Board

#### 3.5.4. Implementasi Node ITDX Ping pada Node-RED

Node ITDX Ping digunakan untuk mendeteksi keberadaan suatu objek tertentu yang terdapat didepan sensor Ping.

#### 3.5.5. Implementasi Node ITDX Buzzer pada Node-RED

Node ITDX Buzzer pada Node-RED yang telah dibangun merupakan komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Buzzer tersebut dapat dijalankan lebih mudah dengan menggunakan aplikasi Node-RED. Untuk menambahkan node ITDX Buzzer pada Node-RED.

#### 3.5.6. Implementasi Node ITDX DHT11 Sensor pada Node-RED

Node ITDX DHT11 sensor pada Node-RED yang dibangun digunakan untuk mendeteksi besarnya suhu dan kelembapan relatif udara di sekitar sensor tersebut. Dengan menangkap sinyal digital yang ada di lingkungan kemudian diolah dalam mikroprosesor untuk selanjutnya dikonversikan menjadi besaran suhu dan kelembapan.



### 3.6. Desain Database

Untuk penyimpanan data yang diolah pada pembangunan sistem ini menggunakan database. Database yang digunakan adalah mysql dan dapat diakses menggunakan phpMyAdmin. Data yang akan

diolah adalah data User dan Komponen. Nama database yang akan di gunakan adalah database “electricity”. Adapun design rancangan database yang akan di implementasikan dapat dilihat pada gambar 9 dibawah ini.

The image shows four database table schemas for the 'electricity' database:

- electricity users:** id (bigint(20) unsigned), username (varchar(255)), email (varchar(255)), password (varchar(255)), created\_at (timestamp), updated\_at (timestamp).
- electricity components:** id (bigint(20) unsigned), name (varchar(255)), status (varchar(255)), room (varchar(255)), total\_component (varchar(255)), created\_at (timestamp), updated\_at (timestamp).
- electricity migrations:** id (int(10) unsigned), migration (varchar(255)), batch (int(11)).
- electricity personal\_access\_tokens:** id (bigint(20) unsigned), tokenable\_type (varchar(255)), tokenable\_id (bigint(20) unsigned), name (varchar(255)), token (varchar(64)), abilities (text), last\_used\_at (timestamp), expires\_at (timestamp), created\_at (timestamp), updated\_at (timestamp).

Gambar 9. Rancangan Database

### 3.7. Page View Komponen

Seperti yang terlihat pada gambar 10 pada halaman list komponen ini user dapat melihat seluruh komponen yang terdaftar dalam sistem dan terpasang

dalam rumah/ ruangan user. Pada menu list komponen ini juga user dapat melakukan proses edit komponen dan juga menghapus komponen yang sudah tidak digunakan ataupun relevan lagi dalam sistem.

The screenshot shows the Admin Dashboard interface. The main content area displays a table titled 'Table Component List' with the following data:

COMPONENT NAME	LOCATION	TOTAL COMPONENT	STATUS	ACTION
Kipas Angin Cosmos	Kamar Tamu	1	ON	EDIT DELETE
Lampu Kamar Tamu	Kamar Tamu	1	ON	EDIT DELETE

Gambar 10. Tampilan Page List Komponen

### 3.8. Hasil Pengujian

Pada sub bab ini berisi tentang pengujian terhadap implementasi yang telah dilakukan untuk melihat dan memastikan bahwa perangkat yang dibangun dapat berjalan dengan baik. Pengujian yang dilakukan yaitu skenario pengujian terhadap kipas

angin diantaranya skenario kipas angin dihidupkan dan skenario kipas angin dimatikan.

Berikut ini adalah hasil dari beberapa pengujian komponen alat listrik yang di design menggunakan Node-RED dan RPi3.

### 3.8.1. Skenario Pengujian Kipas Angin Dihidupkan



Gambar 11. Pengujian Kipas Angin Dihidupkan

Pengujian ini dilakukan untuk memastikan skenario kipas angin yang telah dirancang dapat berjalan dengan baik. Jika jarak yang dideteksi oleh sensor Ping di bawah 33 cm maka Ping sensor akan mengirimkan besar jarak yang terdeteksi ke Relay untuk memberi respon dengan menghidupkan kipas angina seperti yang terlihat pada gambar 11.

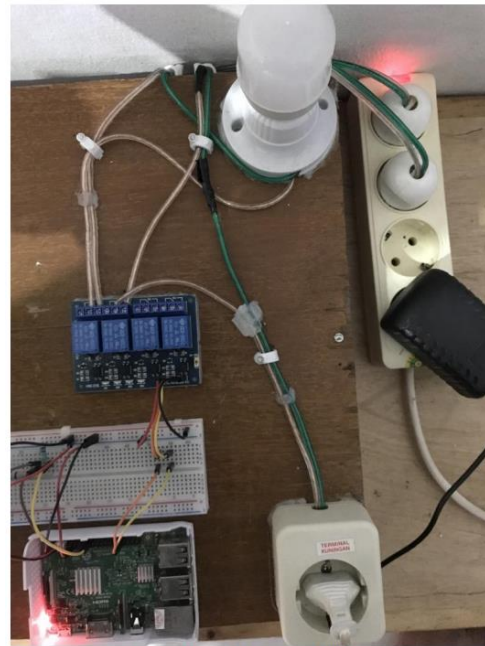
### 3.8.2. Hasil Pengujian Kipas Angin Dimatikan



Gambar 12. Pengujian Kipas Angin Dimatikan

Pengujian ini dilakukan untuk memastikan skenario pada kipas angin yang telah dirancang pada dapat berjalan dengan baik. Pada gambar 12 merupakan pengujian kipas angin yang akan dijalankan, jika jarak yang dideteksi oleh Ping sensor pada jarak di atas 33 cm maka kipas angin akan dimatikan.

### 3.8.3. Hasil Pengujian Lampu Dihidupkan



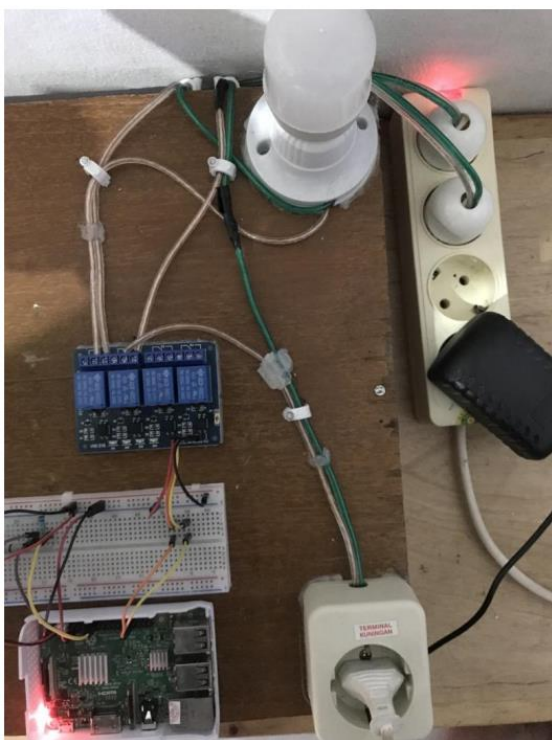
Gambar 13. Pengujian Lampu Dihidupkan

Pengujian pada gambar 13 ini dilakukan untuk memastikan skenario lampu yang telah dirancang dapat berjalan dengan baik. Jjika jarak yang dideteksi oleh sensor Ping di bawah 33 cm maka Ping sensor akan mengirimkan besar jarak yang terdeteksi ke Relay untuk memberi respon dengan menghidupkan lampu.

### 3.8.4. Hasil Pengujian Lampu Dimatikan

Pengujian pada gambar 14 ini dilakukan untuk memastikan skenario pada lampu yang telah dirancang dapat berjalan dengan baik. Jika jarak yang dideteksi oleh Ping sensor pada jarak di atas 33 cm maka lampu akan dimatikan.

Dari berbagai pengujian yang telah dilakukan, maka didapatkan hasil bahwa sistem yang telah dikembangkan dapat berfungsi dengan baik dan data yang didapat dari sistem telah sesuai dengan flow yang telah didesain pada aplikasi Node-RED.



Gambar 14. Pengujian Lampu Dimatikan

#### 4. DISKUSI

Dari hasil penelitian yang sudah dirancang dan diuji ini bahwa pengguna dapat memantau dan mengontrol penggunaan daya listrik dengan efektif melalui antarmuka dashboard yang ramah pengguna. Sistem ini membantu pengguna memantau dan mengendalikan penggunaan daya listrik di rumah dengan mudah melalui dashboard UI yang ramah pengguna. Pengguna dapat membatasi penggunaan listrik tanpa harus mematikan perangkat secara manual, berkat dashboard yang telah dibangun. Penggunaan Node-RED sebagai alat pemrograman memfasilitasi visualisasi hubungan perangkat IoT, memungkinkan pengguna untuk memonitor perangkat IoT tanpa perlu menghadapi kerumitan bahasa pemrograman. Selain itu, pengguna dapat menambahkan perangkat lain sesuai kebutuhan dengan mengembangkan Node kustom di Node-RED.

Integrasi framework Laravel dalam pembuatan dashboard juga membantu pengguna memantau dan mengelola komponen listrik lebih efisien, mencegah lonjakan tagihan yang tak terduga.

Dari beberapa penelitian terdahulu terdapat keunikan dan perbedaan antara penelitian ini dengan penelitian terdahulu. Penulis mengambil 3 penelitian terdahulu sebagai acuan dan pembeda, diantaranya:

1. Rancang Bangun prototype Pengontrolan Lampu Gedung STMIK Amik Riau berbasis IOT Menggunakan raspberry pi 3 model B. Pada penelitian ini semua komponen yang terhubung dengan Raspberry Pi masih belum terintegrasi satu sama lain. Masih membutuhkan pembuatan

kode program yang lebih rumit untuk mengintegrasikan antar komponen.

2. Sistem Absensi Berbasis RFID Menggunakan Raspberry Pi. Pada penelitian tersebut memiliki perbedaan ialah semua komponen yang terhubung dengan Raspberry Pi masih belum terintegrasi satu sama lain. Masih membutuhkan pembuatan kode program yang lebih rumit untuk mengintegrasikan antar komponen. Pada penelitian ini menawarkan solusi untuk mengatasi permasalahan integrasi antar komponen rpi menggunakan Node-RED.
3. Kendali Cerdas Kelistrikan Rumah menggunakan Node-RED Berbasis Raspberry Pi-3 B+ Pada penelitian ini semua komponen tidak terdapat tools yang berguna untuk mengatur dan memantau setiap aktivitas yang sedang berjalan pada komponen rpi. Sehingga diperlukan pembangunan sebuah software berbasis web untuk melakukan hal tersebut.

#### 5. KESIMPULAN

Setelah melakukan perancangan dan pembuatan sistem kemudian dilakukan pengujian. Pengujian menggunakan alat elektronik seperti kipas angin dan lampu. Dengan skenario jika jarak yang dideteksi dibawah 33 cm maka alat elektronik akan menyala dan jika jarak diatas 33 cm maka alat elektronik akan dimatikan dapat berjalan baik. Maka diambil beberapa kesimpulan dari hasil pengujian yang dilakukan penulis menarik kesimpulan bahwa dengan adanya sistem ini dapat mempermudah user dalam memonitor seluruh penggunaan daya listrik dalam rumah. Kemudian user dapat mengendalikan penggunaan daya listrik menggunakan dashboard UI yang user friendly. Memudahkan User dalam membatasi penggunaan listrik tanpa harus mematikan penggunaan perangkat secara manual lagi, karena sudah dapat mengontrolnya menggunakan dashboard yang dibangun. Node-RED merupakan tool pemrograman yang dapat mengkoneksikan secara visual flow-flow lalu mengumpulkan, memproses, dan menyebarkan data. Aplikasi Node-RED mampu memudahkan pengguna untuk memonitoring komponen perangkat IoT tersebut dengan cara yang lebih mudah tanpa khawatir dengan kerumitan bahasa pemrograman.

User dapat mengembangkan costum Node pada Node-RED untuk menambahkan perangkat lain sesuai kebutuhan pengguna. Dengan dashboard yang dibangun menggunakan Laravel *framework* pengguna lebih merasa mudah karena ada alat/ tools yang digunakan untuk monitoring komponen listrik. Dan juga penggunaan listrik dapat lebih dipantau agar tidak menyebabkan pembengkakan tagihan.

#### DAFTAR PUSTAKA

- [1] F. Wijayanti, D. Handaya and N. Putriwibowo, "Monitoring Dan Controlling

- Konsumsi Listrik Skala Rumah Tangga Berbasis Blynk IoT," *Seminar Nasional Teknik Mesin*, vol. 11, no. 1, pp. 317-326, 2021.
- [2] M. R. Tumangke, W. Priharti and D. K. Silalahi, "Rancang Bangun Sistem Pemanas Air Fotovoltaik," *eProceedings of Engineering*, vol. 8, no. 6, pp. 11463-11472, 2021.
- [3] M. Amin, R. Ananda, N. H. Muflih and M. Arif, "Pengenalan Teknologi Microcontroller Dengan Kompetensi Pembuatan Tong Sampah Pintar Pada Siswa Kelas Xi SMK N 2 Tanjungbalai," *Jurnal Pemberdayaan Sosial dan Teknologi Masyarakat*, vol. 2, no. 2, pp. 175-180, 2023.
- [4] R. Friadi and J. , "Sistem Kontrol Intensitas Cahaya, Suhu dan Kelembaban Udara Pada Greenhouse Berbasis Raspberry PI," *JTIS*, vol. 2, no. 2, pp. 30-37, 2019.
- [5] A. Surahman, B. Aditama, M. Bakri and R. , "Sistem Pakan Ayam Otomatis Berbasis Internet Of Things," *Jurnal Teknologi Dan Sistem Tertanam*, vol. 2, no. 1, pp. 13-20, 2021.
- [6] M. . R. Fahlevi and H. Gunawan, "Perancangan Sistem Pendeteksi Banjir Berbasis Internet Of Things," *It (Informatic Technique) Journal*, vol. 8, no. 1, pp. 23-29, 2021.
- [7] T. Rachmadi, *Mengenal apa itu internet of things*, Tiga Ebook, 2020.
- [8] E. Q. Ahmed, I. A. Aljazaery, A. . F. Alzubidi and H. T. S. ALRikabi, "Design and implementation control system for a self-balancing robot based on internet of things by using Arduino microcontroller," *Periodicals of Engineering and Natural Sciences*, vol. 9, no. 3, pp. 409-417, 2021.
- [9] R. Harianja and A. Mailangkay, "IoT Orchestrator Node-RED dan Dashboard History menggunakan Codeignite," *Prosiding Seminar Nasional*, vol. 2, no. 1, pp. 30-33, 2022.
- [10] A. Shollahuddin and H. Saputro, "Sistem Informasi Laboratorium Di Rs Pku Muhammadiyah Mayong Menggunakan Sistem Dashboard Berbasis Web," *Journal of Information System and Computer*, vol. 1, no. 1, pp. 36-42, 2021.
- [11] W. G. Wardhana, I. Arwani and B. Rahayu, "Implementasi Teknologi Restful Web Service Dalam Pengembangan Sistem Informasi Perekaman Prestasi Mahasiswa Berbasis Website (Studi Kasus: Fakultas Teknologi Pertanian Universitas Brawijaya)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 4, no. 2, pp. 680-689, 2020.
- [12] R. Pramuditaa and K. Setyawan, "Sistem Smart Class Berbasis Internet Of Things Dengan Menggunakan Metode Prototype," *SMARTICS Journal*, vol. 8, no. 1, pp. 28-34, 2022.
- [13] R. Subagja, D. Rusmawan and A. S. Adung, "Perancangan Aplikasi Transaksi Dan Pelaporan Di Cv. Anugerah Nurtindo Nusantara (Ann) Berbasis Web," *Telematika*, vol. 3, no. 2, pp. 13-27, 2021.
- [14] F. B. Awi, A. Rabi and W. Dirgantara, "Pengimplementasian Metode Fuzzy Logic pada Kontrol Rumah Jamur Otomatis Berbasis Node-RED," *Jurnal Teknik Elektro dan Komputer TRIAC*, vol. 9, no. 3, pp. 130-134, 2022.
- [15] A. . E. Wijaya and R. B. S. Sukarni, "Sistem Monitoring Kualitas Air Mineral Berbasis Iot (Internet Of Things) Menggunakan Platform Node-Red Dan Metode Saw (Simple Additive Weighting)," *Jurnal Teknologi Informasi dan Komunikasi*, vol. 12, no. 2, pp. 96-106, 2019.
- [16] I. A. Musdar and H. Arfandy, "Rancang bangun Sistem Informasi Pariwisata Sulawesi Selatan Berbasis Android dengan Menggunakan Metode Prototyping," *SINTECH (Science and Information Technology)*, vol. 3, no. 1, pp. 70-76, 2020.
- [17] R. M. R. Clinton and R. Sengkey, "Purwarupa Sistem Daftar Pelanggaran Lalulintas Berbasis Mini-Komputer Raspberry Pi," *Jurnal Teknik Elektro dan Komputer*, vol. 8, no. 3, pp. 181-192, 2019.
- [18] Y. Mulyana, N. Ramsari, A. D. Rachmanto and H. Puspita, "Sistem Informasi Penerimaan Peserta Didik Baru (Ppdb) Menggunakan Framework Laravel 8 (Studi Kasus : Smk Widya Dirgantara)," in *Proceeding Seminar Nasional Ilmu Komputer*, 2022.