

Rancang Bangun Platform IoT OVoRD untuk Aplikasi Sistem Kendali

Immanuel E. P. Pundoko, Vecky C. Poekoel, Feisy D. Kambey
 Jurusan Teknik Elektro, Universitas Sam Ratulangi Manado, Jl. Kampus Bahu, 95115, Indonesia
 ilomon10@gmail.com, vecky.poekoel@unsrat.ac.id, feisykambey@unsrat.ac.id
 Diterima: 16 September 2020; direvisi: 24 November 2020; disetujui: 14 Januari 2021

Abstract — *The rapid development of a system that involves many devices communicating directly and simultaneously. Internet of Things technology is a system that allows devices to be connected directly via the internet, this system has also begun to be implemented in the form of applications, one of which is a webbased application. But to create a system we need to re-create the application of each project that will be developed. Therefore, a platform is created that can create many applications for IoT projects without the need to waste a lot of time and coding burden.*

This platform can make projects more flexible where any data from the device can be used again for other projects with device management and data management features. Users can also build analytics dashboards on each project interactively.

Keyword — *Internet of Things, Cloud Computing, Platform*

Abstrak — Perkembangan pesat suatu sistem yang melibatkan banyak perangkat berkomunikasi secara langsung dan bersamaan. Teknologi Internet of Things adalah salah satu sistem yang memungkinkan perangkat terhubung secara langsung melalui internet, sistem ini juga sudah mulai banyak diterapkan berupa aplikasi, salah satunya aplikasi berbasis Web. Namun untuk membuat suatu sistem tersebut kita perlu membuat ulang aplikasi untuk setiap proyek yang akan dikembangkan. Maka dari itu dibuatlah platform yang dapat membuat banyak aplikasi untuk proyek IoT tanpa perlu banyak membuang waktu dan beban pengkodean. Platform ini dapat membuat proyek yang lebih flexible dimana setiap data dari perangkat bisa di gunakan lagi untuk proyek lainnya dengan fitur manajemen perangkat dan manajemen data. Pengguna juga dapat membangun dashboard analitik di setiap proyek secara interaktif.

Platform ini dapat membuat proyek yang lebih flexible dimana setiap data dari perangkat bisa di gunakan lagi untuk proyek lainnya dengan fitur manajemen perangkat dan manajemen data. Pengguna juga dapat membangun dashboard analitik di setiap proyek secara interaktif.

Kata kunci — *Internet of Things, Komputasi Awan, Platform*

I. PENDAHULUAN

Perkembangan pesat dari suatu peralatan, yang dapat merasakan atau mengeksekusi perintah sederhana secara terus menerus seharian penuh. Perangkat dapat terhubung langsung

ke jaringan Internet yang luas sehingga memungkinkan untuk terhubung ke banyak perangkat sehingga dapat menghasilkan data dan layanan secara real time teknologi tersebut di namakan Internet of Things (IoT). Namun pada Things atau perangkat tersebut terdapat keterbatasan dalam hal komputasi serta penyimpanannya sehingga, performa dan keamanan dapat di khawatirkan. Salah satu contoh, perangkat tidak dapat di menampung data hingga bertahun-tahun lamanya. Hal tersebut dapat diselesaikan dengan mengahlikan proses komputasi dan penyimpanan data ke Cloud Computing Platform. Platform tersebut secara virtual memiliki kemampuan yang tidak terbatas dalam hal penyimpanan dan komputasi [2].

Tentunya sistem tersebut memerlukan adanya sistem dalam bentuk aplikasi untuk memudahkan pengguna untuk mengontrol dan memonitoring data atau perangkat yang terkoneksi. Salah satu teknologi yang sangat populer akhir ini yaitu Web Application atau aplikasi berbasis Web, dan menggunakan Bahasa pemrograman Javascript, dimana aplikasi dapat di jalankan pada web browser seperti Chrome, Firefox dan Edge. Salah satu keunggulan yang besar dari Aplikasi berbasis web ini yaitu, dapat di gunakan pada perangkat smartphone apapun. Dengan menggunakan NodeJS pada sisi server untuk memenuhi kebutuhan seperti sistem real-time.

Seperti yang telah di jelaskan sebelumnya sistem IoT memakan waktu yang lebih saat pengembangan sistem tersebut terlebih lagi memerlukan kemampuan tambahan di bidang Teknologi Informasi (IT) yang mungkin belum di pelajari bagi beberapa pengguna maka dari itu di rancanglah suatu aplikasi yang memangkas pengembang di bagian penyimpanan dan komputasi secara cloud. Fitur yang di sediakan juga terfokus pada fitur yang banyak di implementasikan oleh platform IoT lainnya, yaitu device management, data management, serta monitoring, visualization, analytics, yang bisa dengan mudah di konfigurasi menggunakan Dashboard Builder dan Rules Engine. Seluruh fitur tersebut di sajikan dengan tampilan antarmuka yang interaktif.

A. Internet of Things (IoT)

Internet of Things (IoT) merupakan sebuah konsep dimana suatu objek yang memiliki kemampuan untuk mengirim data

lewat jaringan tanpa memerlukan adanya interaksi dari manusia ke manusia ataupun manusia ke perangkat komputer. IoT sudah berkembang dengan pesat melalui konvergensi teknologi nirkabel *microelectromechanical system (MEMS)* dan juga internet.

IoT bekerja dengan memanfaatkan suatu argumentasi pemrograman, dimana tiap-tiap perintah argumen tersebut bisa menghasilkan suatu interaksi antara mesin secara terus-menerus yang terhubung secara otomatis tanpa campur tangan manusia dan tanpa batasan jangkauan. Dimana benda-benda setiap hari yang bisa dilengkapi dengan mikrokontroler dan pemancar gelombang.

B. Platform IoT

Platform IoT adalah sebuah landasan untuk perangkat elektronik dan sensor terhubung secara terstruktur sehingga perangkat IoT tersebut bisa di management dengan baik. Platform ini juga menyediakan fasilitas komputasi dan penyimpanan sebagai solusi perangkat yang memiliki kecepatan proses dan media penyimpanan yang terbatas. Dengan adanya sistem management pada platform tersebut pengguna bisa dengan mudahnya melihat serta mengolah data yang tersimpan.

C. Cloud Computing

Komputasi awan adalah gabungan pemanfaatan teknologi komputer “komputasi” dan pengembangan berbasis internet “awan”. Awan (cloud) adalah metafora dari internet, sebagaimana awan yang sering digambarkan di diagram jaringan komputer. Sebagaimana awan dalam diagram jaringan komputer tersebut, awan dalam Cloud Computing juga merupakan abstraksi dari infrastruktur kompleks yang tertutup (disembunyikan). Di dalamnya memiliki suatu metoda komputasi di mana kapabilitas teknologi informasi disajikan sebagai suatu layanan, sehingga pengguna dapat mengaksesnya lewat Internet tanpa mengetahui apa yang ada didalamnya, atau memiliki kendali terhadap infrastruktur teknologi terkait. Dalam sebuah makalah tahun 2008 yang dipublikasi IEEE Internet Computing “Cloud Computing adalah suatu paradigma dimana informasi secara permanen tersimpan di server internet dan tersimpan secara sementara di komputer pengguna (client).

Cloud Computing adalah suatu konsep umum yang mencakup SaaS, Web 2.0, dan tren teknologi terbaru lain yang dikenal luas, dengan tema umum berupa ketergantungan terhadap Internet untuk memberikan kebutuhan komputasi pengguna, contohnya, Google Suite menyediakan aplikasi bisnis umum secara daring yang diakses melalui suatu penjelajah web (browser) dengan perangkat lunak dan data yang tersimpan di server.

D. Aplikasi Web

Dalam rekayasa perangkat lunak, suatu aplikasi web adalah suatu aplikasi yang diakses menggunakan penjelajah web (web browser) melalui suatu jaringan seperti Internet atau intranet. Ia juga merupakan suatu aplikasi perangkat lunak

komputer yang dikodekan dalam bahasa yang didukung penjelajah web (seperti JavaScript dan PHP) dan bergantung pada penjelajah tersebut untuk menampilkan aplikasi.

E. NodeJS

Node.js adalah platform software pada sisi server yang memungkinkan untuk mengeksekusi program atau kode Javascript pada mesin server. Untuk bisa menjalankan Javascript, Node.js dibangun dengan engine javascript V8 milik Google. Disamping itu, platform ini memiliki library sendiri untuk menyiapkan layanan web-server sehingga tidak perlu lagi menggunakan program web-server seperti Nginx dan Apache.

F. Rule Engine menggunakan algoritma rete

Algoritma Rete adalah algoritma pencocokan pola untuk mengimplementasikan sistem berbasis aturan (rule-based system). Algoritma ini dirancang oleh Charles L. Forgy dalam tesis pada 1982 di Carnegie Mellon University untuk menerapkan banyak aturan (rules) atau pola ke banyak objek, atau fakta, dengan dasar pengetahuan.

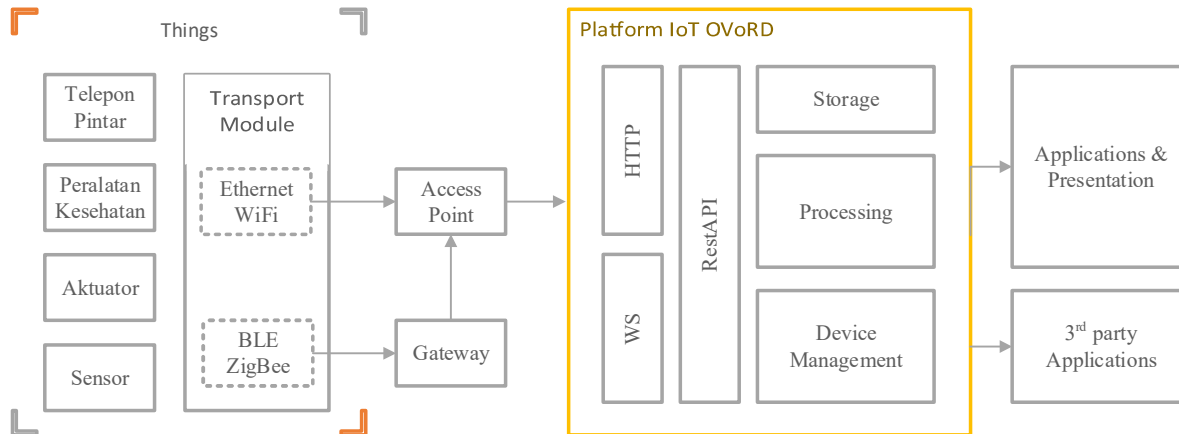
Algoritma Rete menyediakan deskripsi logis yang digeneralisasikan dari sebuah implementasi fungsionalitas yang bertujuan untuk mencocokkan kumpulan data (“fact”) dengan aturan (“rules”) dalam sistem produksi pencocokan pola (kategori rule engine). Sebuah produksi terdiri dari satu atau lebih kondisi dan serangkaian tindakan yang dapat dilakukan untuk setiap kumpulan fact yang sesuai dengan kondisi tersebut. Kondisi menguji atribut fact termasuk pengenalan jenis fact.

Algoritma rete memiliki karakteristik utama sebagai berikut: Mengurangi atau menghilangkan beberapa pemborosan melalui penggunaan suatu sharing node; Menyimpan sebagian kecocokan saat melakukan penggabungan antara jenis fact yang berbeda. Hal ini memungkinkan sistem produksi untuk menghindari beban proses ulang secara keseluruhan semua fact setiap kali ada perubahan pada memori produksi. Sebaliknya, sistem produksi hanya perlu memroses perubahan pada perubahan memori tersebut; Memungkinkan penghapusan elemen memori secara efisien ketika fact ditarik dari memori.

II. METODE PENELITIAN

Pada penelitian perancangan platform IoT, ada beberapa elemen penting yaitu berupa perangkat keras (hardware) dan perangkat lunak (software), namun kali ini kita berfokus pada perancangan sistem yang memudahkan pengguna IoT untuk membangun sistem atau proyek dengan lebih cepat, maka dari itu kita akan berfokus sisi software.

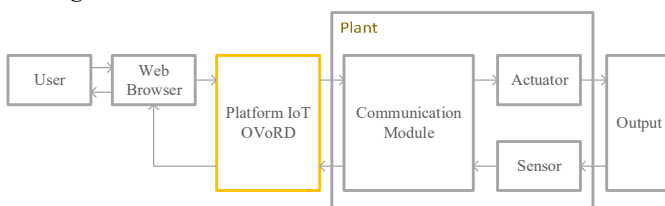
A. Arsitektur sistem



Gambar 1. Arsitektur Platform IoT

Bisa kita lihat beberapa contoh perangkat IoT (things) yang bisa di terapkan pada platform dengan menggunakan modul komunikasi tertentu untuk beberapa kasus sehingga perangkat tersebut bisa berkomunikasi secara langsung ataupun tidak langsung melalui access point/gateway, kemudian platform handle koneksi menggunakan protocol TCP/IP dalam hal ini penulis akan menggunakan protocol HTTP dan WebSocket (untuk kemampuan realtime) lalu penulis menggunakan sistem RestAPI untuk manajemen permintaan-permintaan (request) yang dilakukan perangkat. Sistem seperti ini pastinya memiliki perangkat IoT yang terhubung bukan hanya satu saja, jadi kita memerlukan manajemen perangkat supaya data-data yang di terima bisa diproses, simpan dan bisa di ambil kembali dengan teratur.

B. Diagram blok kendali



Gambar 2. Diagram blok kendali

Pengguna bisa melihat dan mengontrol perangkatnya melalui web browser manapun dengan mudah. Setiap perubahan yang terdapat pada perangkat (plant) bisa di monitoring dan di kontrol secara realtime (untuk perangkat yang telah mendukung koneksi realtime). Dari semua konsep yang telah di tulis sebelumnya, penulis memberikan nama platform ini dengan nama Online Value of Realtime Data atau dapat disingkat menjadi OVoRD.

C. Daftar Use Case

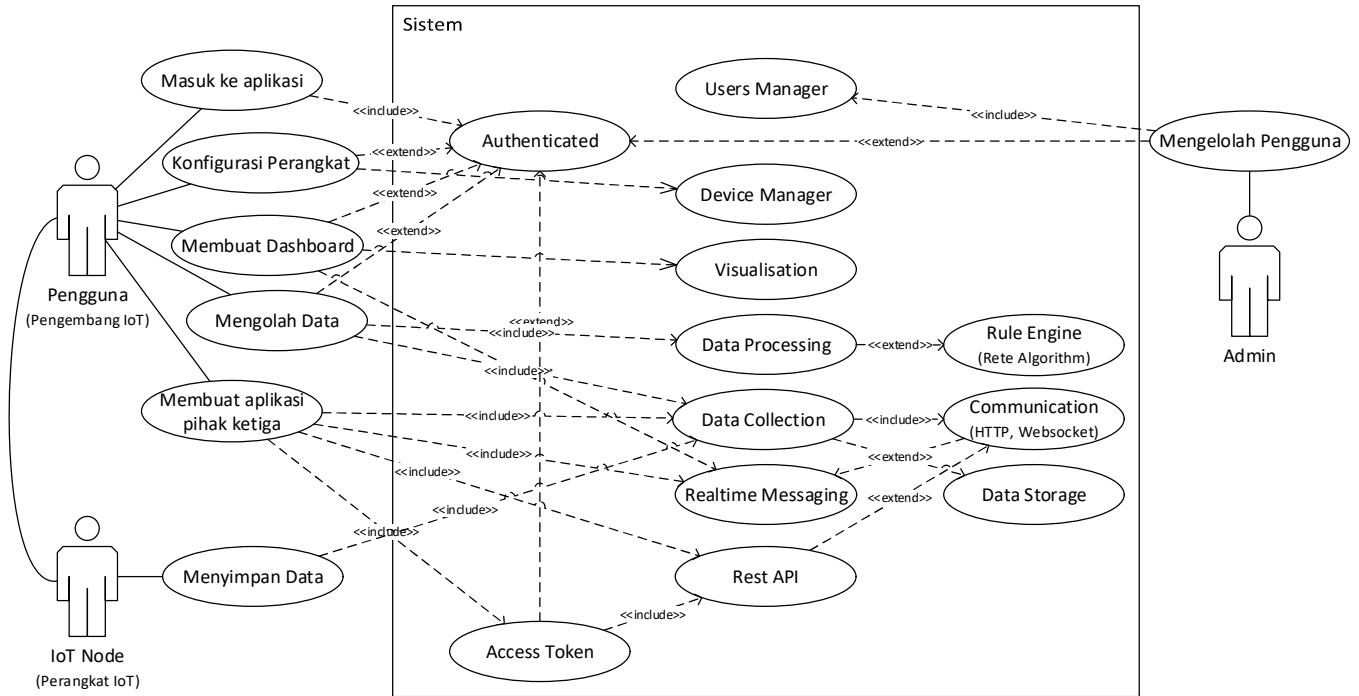
Ada beberapa kasus penggunaan platform IoT yang akan di simulasikan pada pembahasan untuk menguji software yang telah di kembangkan:

- 1) Masuk ke aplikasi, Pada kasus ini pengguna akan melakukan autentifikasi dengan memverifikasikan

nama pengguna (username) dan kata sandi (password) yang telah terdaftar pada form masuk ke aplikasi;

- 2) Menambahkan Pengguna, pada kasus ini admin akan menambahkan pengguna baru;
- 3) Mengelola Pengguna, pada kasus ini admin akan memodifikasi pengguna yang terdaftar;
- 4) Menghapus Pengguna, pada kasus ini admin akan menghapus akun pengguna yang sudah tidak lagi di pakai;
- 5) Mendaftarkan perangkat baru, pada kasus ini pengguna akan menambahkan atau mendaftarkan perangkat yang baru;
- 6) Konfigurasi Perangkat, pada kasus ini pengguna akan menambahkan atau mengurangi field perangkat yang terdaftar.
- 7) Menghapus perangkat, pada kasus ini pengguna akan menghapus perangkat yang telah terdaftar.
- 8) Membuat *access token*, pada kasus ini pengguna akan membuat *access token*.
- 9) Menghapus *access token*, pada kasus ini pengguna akan menghapus akses melalui *access token*.
- 10) Menyimpan data, pada kasus ini pengguna akan menyimpan data yang telah di kumpulkan pada perangkat IoT milik pengguna ke platform.
- 11) Mengolah data, pada kasus ini pengguna akan mengolah data dari perangkat secara realtime sebelum data tersebut di simpan.
- 12) Membuat *dashboard* baru, pada kasus ini pengguna akan membangun dashboard yang bisa mengontrol perangkat dan melihat data secara grafis.
- 13) Membangun tampilan *dashboard*, pada kasus ini pengguna akan menambahkan atau memodifikasi tampilan (*widget*) pada *dashboard*.
- 14) Mengakses data platform dari aplikasi pihak ketiga, pada kasus ini pengguna akan mengakses data yang akan di simpan atau yang telah terimban pada platform melalui aplikasi pihak ketiga yang telah di buat.

D. Use Case Diagram



Gambar 3. Use Case Diagram platform IoT

Kebutuhan yang sudah didefinisikan dalam pada daftar use case, kemudian dimodelkan menggunakan metode unified modeling language (UML) berupa use case diagram dan dilakukan penentuan scenario yang terjadi pada setiap kebutuhan tersebut. Gambar 3 adalah use case diagram dari penggunaan platform IoT.

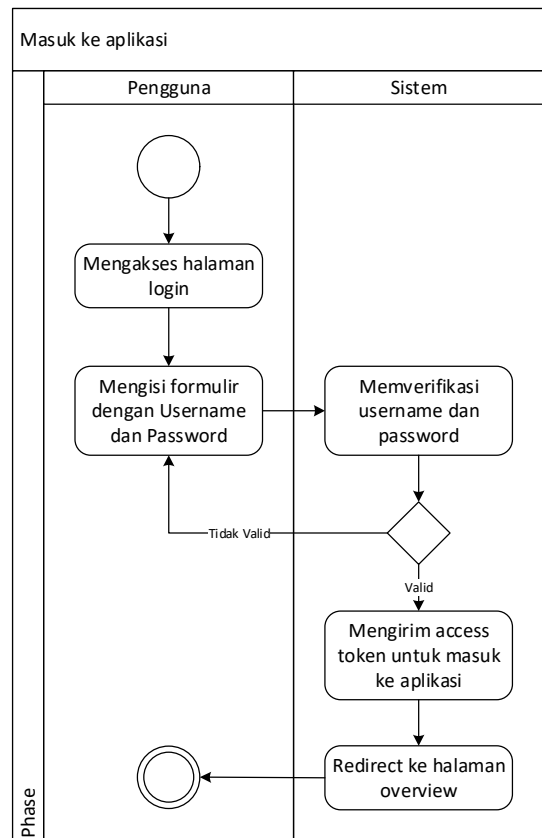
III. HASIL DAN PEMBAHASAN

A. Deskripsi umum aplikasi

OVORD (Online Value of Realtime Data) adalah sebuah platform IoT yang dirancang khusus untuk pelaku IoT (pengguna atau developer) yang memungkinkan pengguna tersebut bisa menghemat waktu saat melakukan pengembangan project IoT. Platform ini menyediakan fasilitas penyimpanan untuk setiap data dari perangkat-perangkat IoT (Things) yang terhubung dan bisa dengan mudah menghapus ataupun menambahkan data dari masing-masing perangkat menggunakan fitur Device Manager, data dari perangkat bisa secara langsung di tampilkan pada dashboard yang bisa di bangun tanpa beban pengkodean dengan fitur Dashboard Builder, data itu juga bisa diproses atau diolah sebelum di tampilkan dengan fitur Rule Engine yang memungkinkan pengguna membangun suatu aturan khusus secara visual.

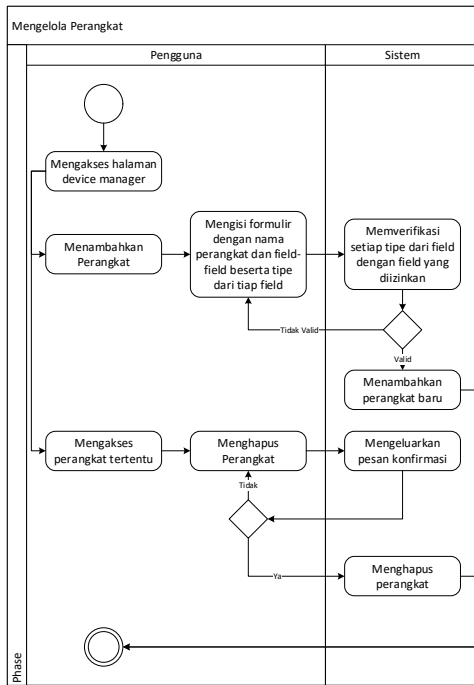
B. Activity Diagram

Berdasarkan use case yang telah dibuat dalam daftar use case bab II, maka proses dan urutan aktivitas sistem setiap kasus dapat di gambarkan dengan diagram aktifitas agar mudah di pahami.



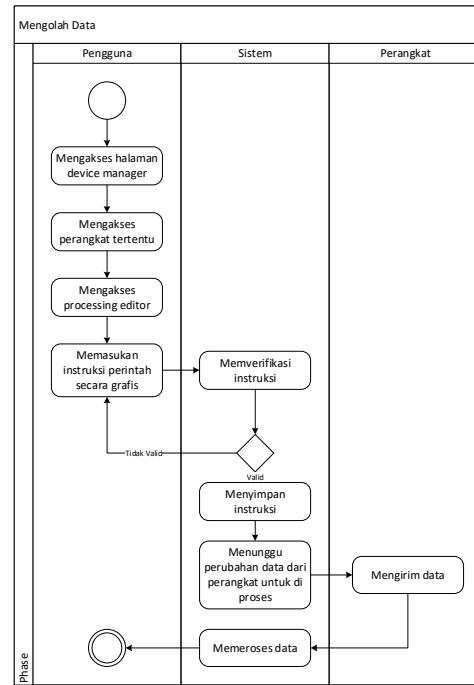
Gambar 4. Activity diagram Masuk ke dalam aplikasi

Memerlukan username/email dan password yang telah terdaftar untuk masuk ke dalam aplikasi danm mengoperasikannya.



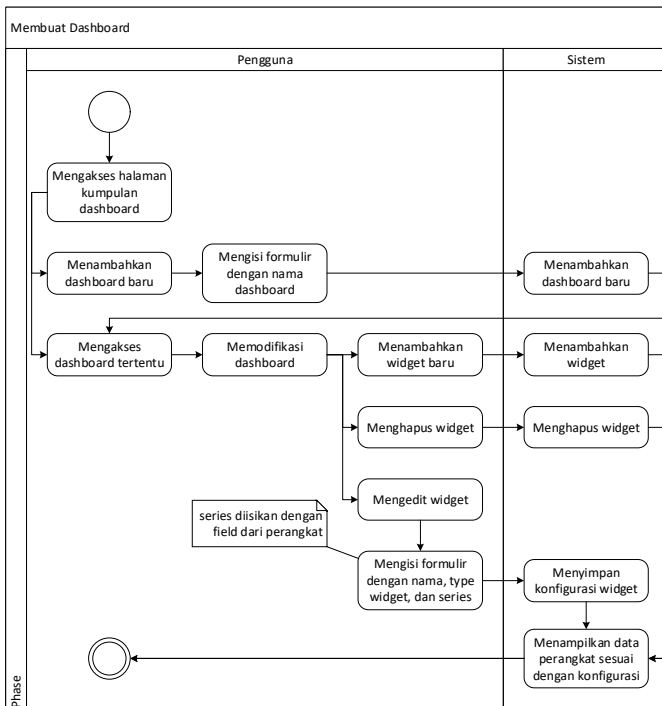
Gambar 5. Activity Diagram Mengelola Perangkat

Pada bagian ini pengguna diharuskan untuk masuk dulu ke aplikasi setelah itu pengguna bisa mengakses halaman device manager (manager perangkat). Pengguna dapat menambahkan maupun mengurangi perangkat yang belum atau telah ia tambahkan ke sistem.



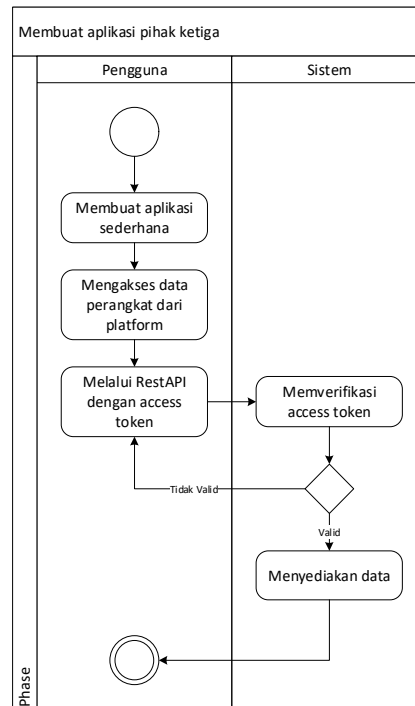
Gambar 7. Activity Diagram Mengelola Data

Untuk mengolah data yang akan dikumpul ke perangkat dapat dilakukan dengan mengakses processing editor pada perangkat tertentu kemudian, upload instruksi yang akan di terapkan, instruksi tersebut bisa langsung di rangkai pada halaman tersebut.



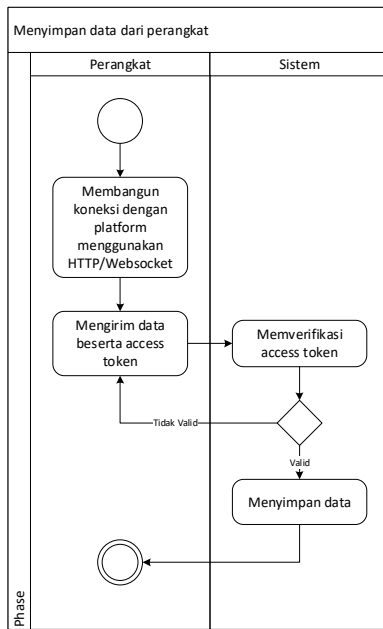
Gambar 6. Activity Diagram Mengelola Dashboard

Mengolah dashboard dapat dilakukan dengan memilih dashboard yang telah dibuat. Jika belum ada satupun dashboard, pengguna dapat membuat satu dengan mengisi formulir dengan nama. Kemudian dashboard dapat di rancang dengan menambahkan widget baru dan mengisi formulir dengan nama widget, tipe widget, dan series.



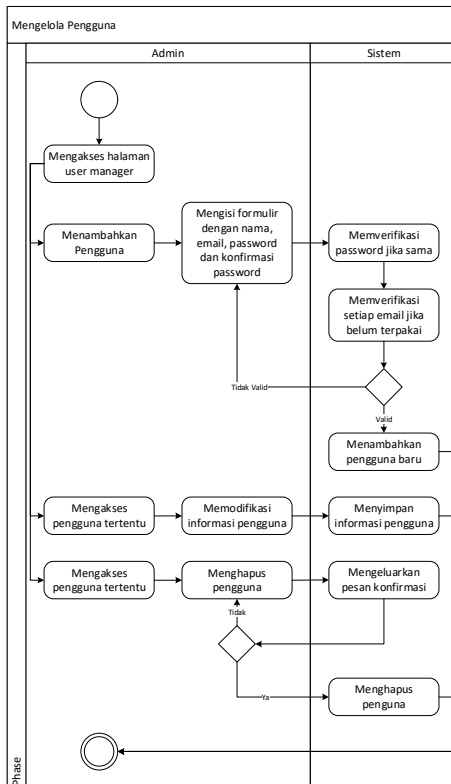
Gambar 8. Activity Diagram Membuat aplikasi pihak ketiga

Pengguna dapat mengambil data dari sistem dengan menyediakan *access token* kemudian lakukan permintaan (*request*) http ke server.



Gambar 9. Activity Diagram Menyimpan data dari perangkat

Perangkat menyimpan data melalui koneksi protocol HTTP ataupun Websocket dengan melampirkan *access token* pada *header* permintaan kemudian sistem akan menyimpan data yang akan di simpan.



Gambar 9. Activity Diagram Menyimpan data dari perangkat

Dengan menggunakan akses Admin, pengguna dapat mengolah akun yang terdaftar pada sistem, mulai dari menambahkan, memodifikasi informasi, dan menghapus akun pengguna. Untuk menambahkan akun, pengguna harus menyiapkan nama, email, dan password untuk akun yang akan di daftarkan.

C. Analisis class diagram

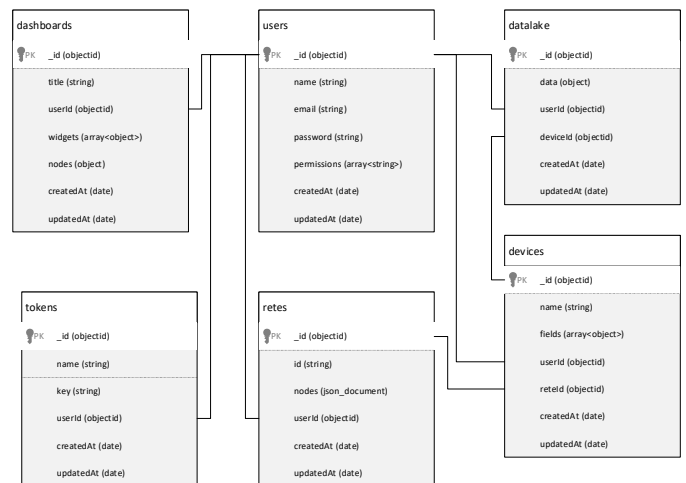
Class diagram yang di dapatkan berdasarkan alur activity diagram yang telah di buat.

TABEL I
TABEL ANALISIS CLASS DIAGRAM

Nama	ATRIBUT	Operasi ^a
Pengguna	Nama Email Password Role	Baca Pengguna Tambah Pengguna Hapus Pengguna Edit Pengguna
Perangkat	Nama Fields Penggua	Baca Perangkat Tambah Perangkat Hapus Perangkat Edit Fields Filter Pengguna
Data Collection	Data Pengguna Perangkat CreatedAt	Baca Data Tambah Data Hapus Data Filter Pengguna Filter Perangkat
Dashboard	Nama Widget Pengguna	Baca Dashboard Tambah Dashboard Hapus Dashboard Edit Widget Filter Pengguna
Access Token	Nama Token Pengguna	Baca Token Tambah Token Hapus Token Filter Pengguna
Processing	Instruksi Perangkat	Tambah Proses Hapus Proses Filter Perangkat
Login System	Email Password Role	Masuk
Access Token System	Access Token	validate

D. Implementasi Basis Data

Berikut ini implementasi basis data yang di tetapkan atau yang saat ini digunakan

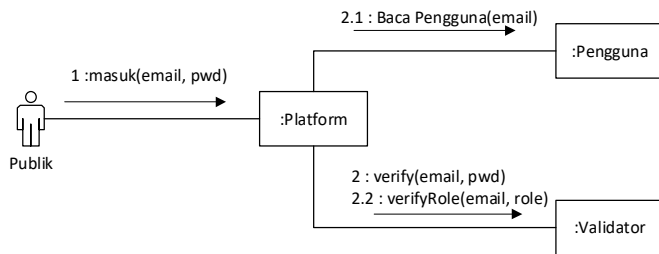


Gambar 10. Implementasi Basis Data

Terdapat enam basis data utama yaitu users yang berisikan data akun pengguna, dashboards yang berisikan data dan komposisi widget, tokens yang berisikan tiket untuk tiap akun, retes yang berisikan instruksi tiap perangkat, datalakes berisikan nilai-nilai data perangkat yang terkumpul, devices

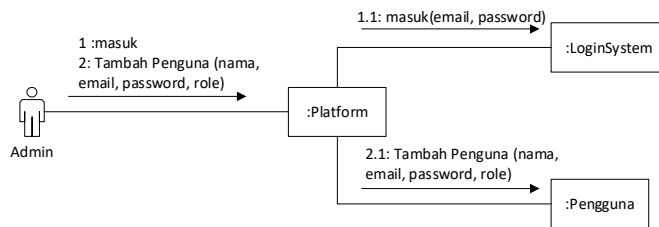
berisikan data umum perangkat yang terdaftar.

E. Communication Diagram



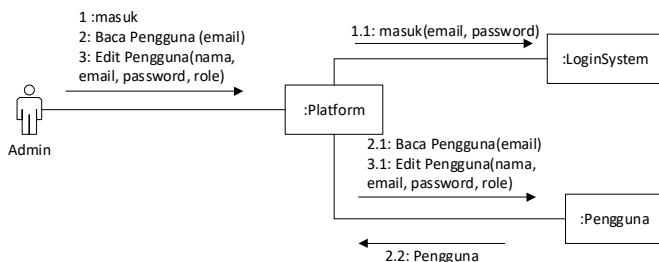
Gambar 11. Communication Diagram Masuk ke aplikasi

Untuk masuk ke aplikasi pengguna perlu mengirimkan email dan password melalui halaman login kemudian sistem akan mencari dan memvalidasi email dan password tersebut.



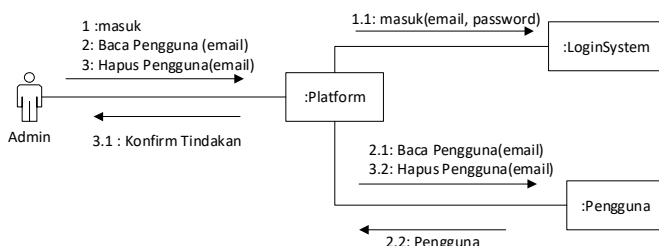
Gambar 12. Communication Diagram Tambah pengguna

Untuk menambahkan akun, pengguna perlu menyiapkan informasi dasar yaitu nama, email, dan password. Namun sebelum itu pengguna harus mendapatkan akses admin terlebih dahulu.



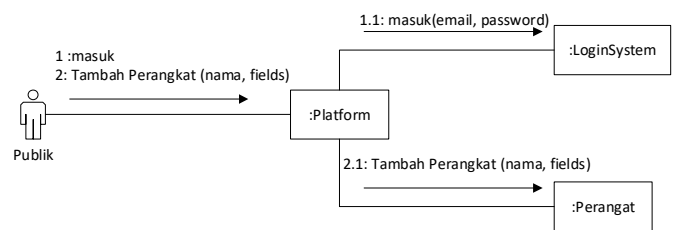
Gambar 13. Communication Diagram Mengelola Pengguna

Untuk mengolah akun, pengguna perlu mendapatkan akses admin kemudian edit formulir dengan memilih pengguna yang akan di modifikasi informasinya.

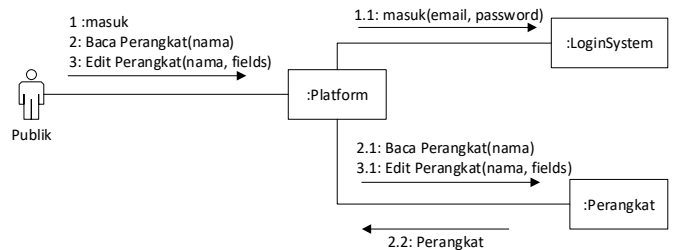


Gambar 14. Communication Diagram Menghapus Pengguna

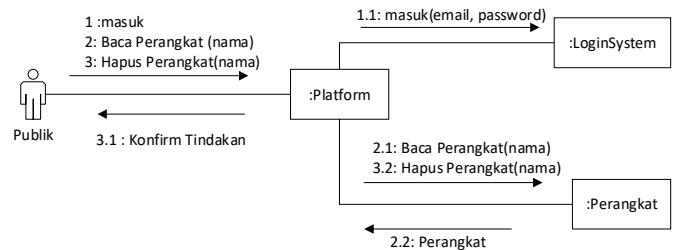
Untuk menghapus pengguna, perlu mendapatkan akses admin kemudian pilih akun yang akan di hapus.



Gambar 15. Communication Diagram Mendaftarkan perangkat

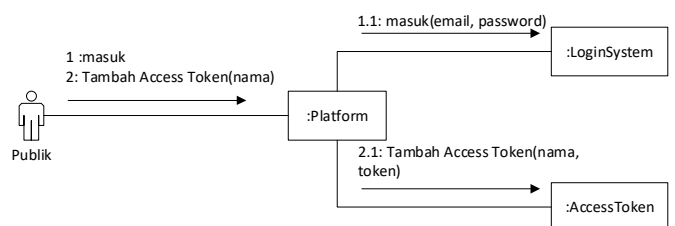


Gambar 16. Communication Diagram Konfigurasi Perangkat

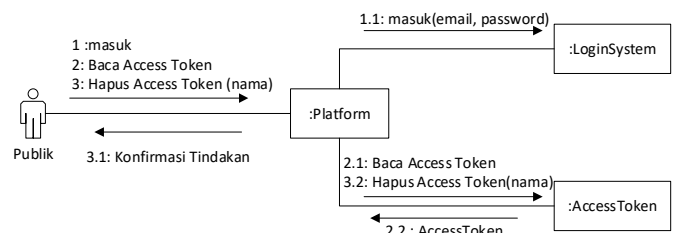


Gambar 17. Communication Diagram Menghapus perangkat

Untuk mendaftarkan perangkat, kirimkan nama dan *fields* seperti pada Gambar 15, kemudian konfigurasi dengan memilih perangkat yang akan di konfig kemudian modifikasi seperti pada Gambar 16, juga dapat menghapus perangkat tersebut.

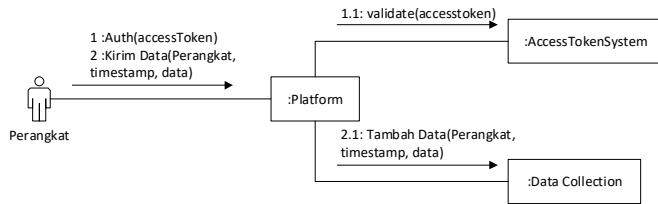


Gambar 18. Communication Diagram Menambahkan access token



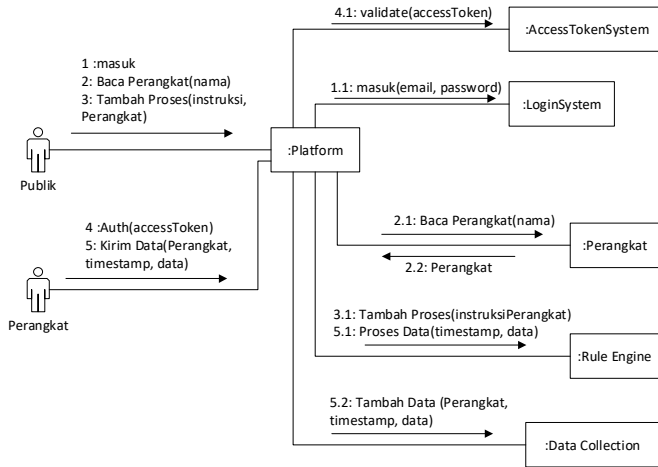
Gambar 19. Communication Diagram Menghapus access token

Untuk menambahkan token, hanya mengirimkan nama token yang akan dibuat (Gambar 18). Dan menghapus token pilih token yang akan dihapus.



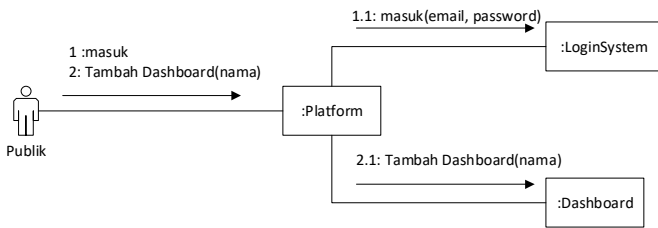
Gambar 20. Communication Diagram Menyimpan data

Untuk menyimpan data dari perangkat siapkan *access token* kemudian kirimkan data beserta token tersebut, kemudian sistem akan menyimpa data tersebut jika tokennya valid.



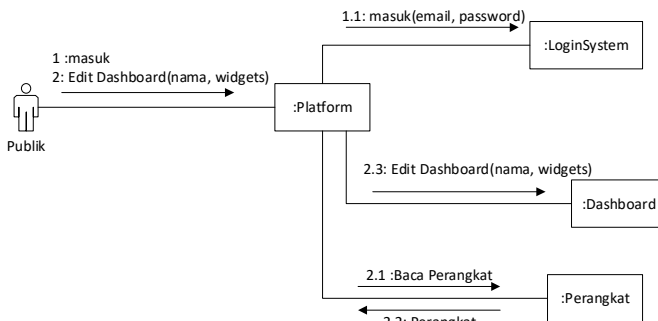
Gambar 21. Communication Diagram Mengolah data

Untuk mengolah data perangkat tambahkan proses instruksi pada perangkat yang diinginkan kemudian kirim data (Gambar 21) dari perangkat maka data tersebut akan langsung di proses sesuai instruksi yang di terapkan.



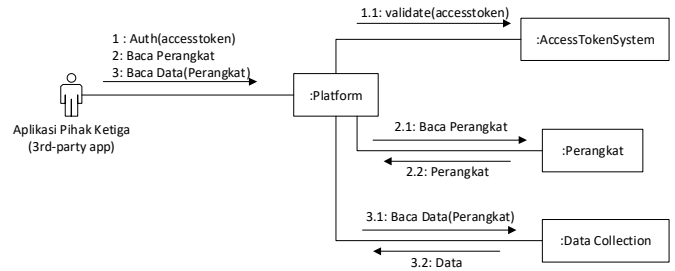
Gambar 22. Communication Diagram Membuat dashboard baru

Untuk membuat dashboard baru hanya menyiapkan nama dashboard kemudian sistem akan menambahkan dashboard tersebut.



Gambar 23. Communication Diagram Membangun tampilan dashboard

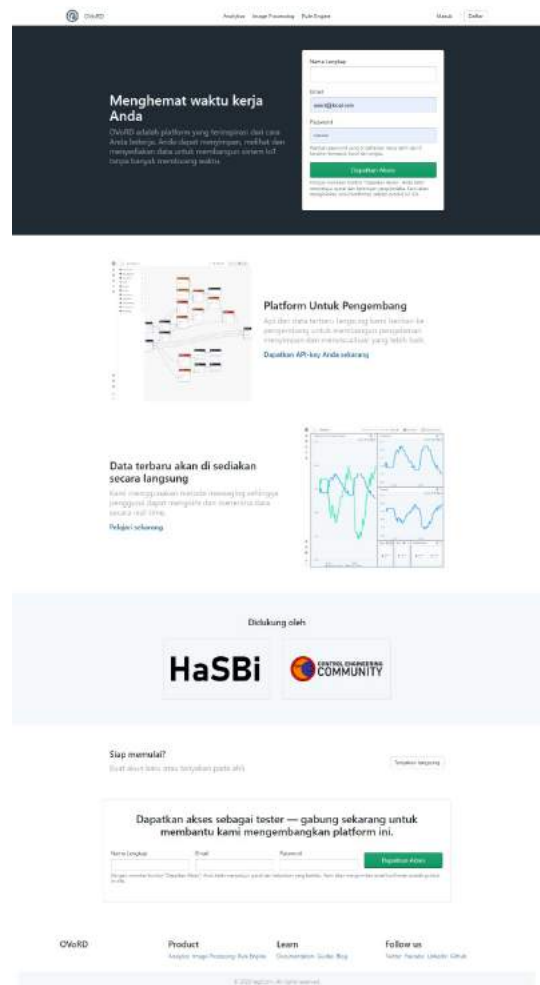
Pengguna dapat memodifikasi dashboard, dan juga dapat membangun tampilan dashboard dengan mengedit widget yang ada atau yang baru.



Gambar 24. Communication Diagram Mengakses data platform dari aplikasi pihak ketiga

Untuk mengakses data platform dari aplikasi tambahan (3rd party), perlu menyiapkan *access token* kemudian lakukan permintaan.

F. Implementasi Antarmuka



Gambar 25. Implementasi Antarmuka halaman Landing

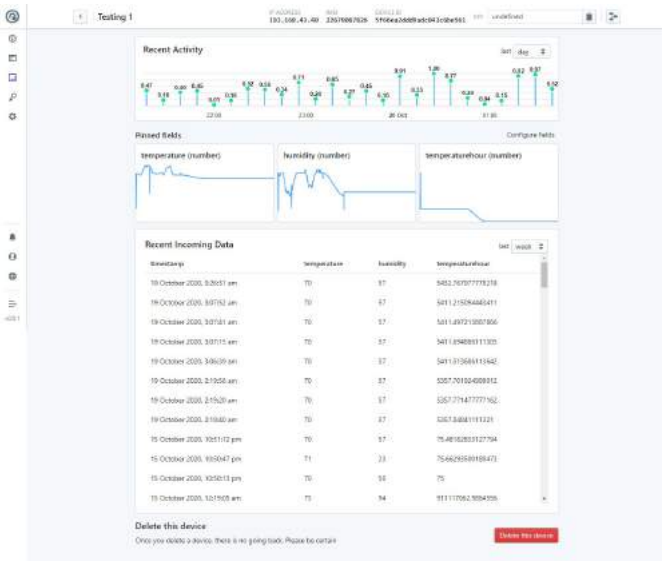
Implementasi antarmuka ini menampilkan deskripsi umum platform OVoRD ini serta navigasi untuk masuk atau mendaftar ke aplikasi.

Berikut ini implementasi antarmuka untuk setiap fitur yang di rancang.



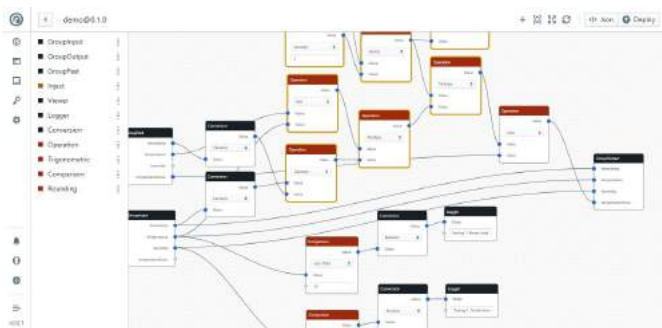
Gambar 26. Implementasi Antarmuka halaman Dashboard

Halaman ini menampilkan data secara visual dan bisa di modifikasi sesuai keinginan menekan tombol pengaturan (cog)



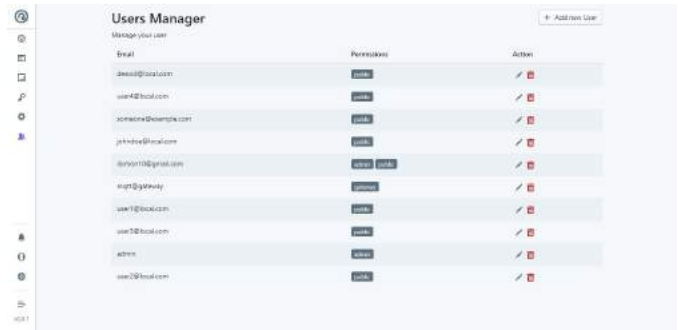
Gambar 27. Implementasi Antarmuka halaman Manajemen Device

Halaman ini menampilkan rangkuman data berdasarkan range waktu tertentu, dalam bentuk grafik maupun tabel. Pada halaman ini juga terdapat tombol delete jika pengguna ingin menghapus perangkat tersebut nanti akan muncul pesan konfirmasi.



Gambar 28. Implementasi Antarmuka halaman editor rule engine

Implementasi antarmuka ini menampilkan editor rule engine secara visual.



Gambar 29. Implementasi Antarmuka halaman Dashboard

Implementasi antarmuka ini menampilkan daftar akun pengguna, pada halaman ini pengguna dengan akses admin bisa mengolah pengguna mulai dari menambahkan, memodifikasi informasi, serta menghapus akun pengguna secara permanen.

G.Laporan Pengujian Black-box

Pengujian akan mengacu pada use case yang telah ditetapkan pada bagian “Daftar Use Case”. Kemudian akan ada pengujian pada setiap halaman aplikasi.

TABEL II
HASIL PENGUJIAN “USE CASE MASUK KE APLIKASI”

No	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “Masuk” pada menu.	Muncul halaman login dengan formulir email dan password	Y	
2.1	Mengirim formulir email dan password yang tidak-valid.	Tidak bisa mengakses sistem dan muncul pesan error.	Y	
2.2	Mengirim formulir email dan password yang valid	Muncul halaman overview pengguna.	Y	
3.	Keluar dari aplikasi dengan memilih tombol “Logout” pada sidebar.	Muncul halaman login dengan formulir email dan password	Y	

TABEL III
HASIL PENGUJIAN “USE CASE MENAMBAHKAN PENGGUNA”

No	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.1	Mengirim formulir email dan password akun publik.	Muncul halaman overview pengguna dan tidak ada pilihan menu “User Manager”.	Y	
1.2	Mengirim formulir email dan password akun admin.	Muncul halaman overview pengguna dan ada pilihan menu “User Manager”.	Y	
2.	Memilih tombol “User Manager” pada sidebar.	Muncul halaman daftar pengguna.	Y	
3.	Memilih tombol “Add new user”.	Muncul formulir name, email, password, dan permissions.	Y	
4.1	Mengirim data pengguna yang sama dengan data pengguna yang terdaftar.	Muncul pesan error.	Y	
4.2	Mengirim data pengguna yang baru.	Daftar pengguna bertambah satu pengguna baru.	Y	

TABEL IV
HASIL PENGUJIAN “USE CASE MENGELOLA PENGGUNA”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “edit” salah satu pengguna.	Muncul halaman edit user profile dan formulir name, email, permissions, dan password.	Y	
2.1	Memodifikasi data pengguna yang sama dengan data pengguna yang terdaftar.	Muncul pesan error.	Y	
2.2	Memodifikasi data pengguna dengan data yang baru.	Informasi pengguna berubah.	Y	

TABEL IV
HASIL PENGUJIAN “USE CASE MENGHAPUS PENGGUNA”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “edit” salah satu pengguna.	Muncul halaman edit user profile dan formulir name, email, permissions, dan password.	Y	
2	Memilih tombol “Delete user”	Muncul pesan konfirmasi tindakan.	Y	
3	Mengonfirmasi tindakan.	Muncul halaman daftar user dan pengguna yang di hapus telah hilang.	Y	

TABEL V
HASIL PENGUJIAN “USE CASE MENDAFTARKAN PERANGKAT BARU”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “Devices”	Muncul halaman daftar perangkat.	Y	
2	Memilih tombol “Register new one”	Muncul formulir nama, dan fields beserta tipe field.	Y	
3	Mengirim data perangkat baru.	Daftar perangkat bertambah satu perangkat baru.	Y	

TABEL VI
HASIL PENGUJIAN “USE CASE KONFIGURASI PERANGKAT”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih salah satu perangkat	Muncul halaman manajemen perangkat.	Y	
2	Memilih tombol “Configure fields”	Muncul daftar fields beserta tipe field.	Y	
3	Mengirim data perangkat baru.	Field perangkat berubah.	Y	

TABEL VII
HASIL PENGUJIAN “USE CASE MENGHAPUS PERANGKAT”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih salah satu perangkat	Muncul halaman manajemen perangkat.	Y	
2	Memilih tombol “Delete this device”	Muncul pesan konfirmasi tindakan.	Y	
3	Mengonfirmasi tindakan.	Perangkat yang dihapus hilang dalam daftar perangkat.	Y	

TABEL VIII
HASIL PENGUJIAN “USE CASE MEMBUAT ACCESS TOKEN”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “Tokens”	Muncul halaman daftar access token.	Y	
2	Memilih tombol “Generate new token”	Muncul formulir nama token.	Y	
3	Mengirim data perangkat baru.	Daftar perangkat bertambah satu perangkat baru.	Y	

TABEL IX
HASIL PENGUJIAN “USE CASE MENGHAPUS ACCESS TOKEN”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1	Memilih tombol “Delete” salah satu token.	Muncul pesan konfirmasi tindakan.	Y	
2	Mengonfirmasi tindakan.	Token yang dihapus hilang dalam daftar token.	Y	

TABEL X
HASIL PENGUJIAN “USE CASE MENYIMPAN DATA DARI PERANGKAT”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1	Mengirim data dengan format JSON beserta access token	Mendapatkan kembali data yang di tambahkan dari server.	Y	
2	Mengirim data dengan format JSON dan access token yang tidak sesuai	Mendapatkan pesan error dari server.	Y	

TABEL XI
HASIL PENGUJIAN “USE CASE MENGOLAH DATA DARI PERANGKAT”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1	Mengakses rule editor salah satu perangkat.	Muncul halaman editor <i>rules engine</i> .	Y	
2	Menambahkan instruksi dan memilih tombol deploy.	Intruksi tersimpan.	Y	
3	Perangkat mengirim data.	Mendapatkan kembali data yang telah diolah dari server.	Y	

TABEL XII
HASIL PENGUJIAN “USE CASE MEMBUAT DASHBOARD BARU”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “Dashboards” pada sidebar	Muncul halaman daftar <i>dashboard</i> .	Y	
2	Memilih tombol “Make new one”	Muncul formulir nama <i>dashboard</i> .	Y	
3	Mengirim data <i>dashboard</i> baru.	Daftar <i>dashboard</i> bertambah satu <i>dashboard</i> baru.	Y	

TABEL XIII
HASIL PENGUJIAN “USE CASE MEMBANGUN TAMPILAN DASHBOARD”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih salah satu <i>dashboard</i>	Muncul halaman <i>dashboard</i> yang di pilih.	Y	
2	Menambah widget baru denga memilih tombol “Add new window”	Muncul satu window baru dengan nama “New Widget”	Y	
3	Memilih tombol “Setup widget” pada widget kosong	Muncul formulir title, type, series untuk konfigurasi widget.	Y	
4	Memilih tipe widget “Time series graph”	Muncul formulir baru yaitu “option” untuk mengonfigurasi tampilan widget.	Y	
5	Memilih salah satu field pada salah satu perangkat dibagian series	Muncul formulir series baru yang opsional.	Y	
6	Mengirim data widget baru.	Widget disimpan dan tampilan window widget di isikan dengan tampilan widget yang telah di sesuaikan.	Y	

TABEL XIV
HASIL PENGUJIAN “USE CASE MENGAkses DATA PLATFORM DATI APLIKASI PIHAK KETIGA”

No.	NAMA	Luaran yang diharapkan	Validitas	
			Y	T
1.	Memilih tombol “Tokens” pada sidebar	Muncul halaman manajemen access token.	Y	
2	Memilih tombol “copy” di salah satu token.	Kode access token di simpan pada clipboard dan bisa di paste.	Y	
3	Mengirim data dengan format JSON beserta access token.	Mendapatkan kembali data yang dikirim dari server.	Y	
4	Mengambil daftar data.	Mendapatkan data yang tersimpan sebelumnya.	Y	

H.Laporan Pengujian Beban

Pengujian beban ini menggunakan aplikasi Apache JMeter. Berikut ini konfigurasi yang dipakai untuk menguji beban menyimpan data

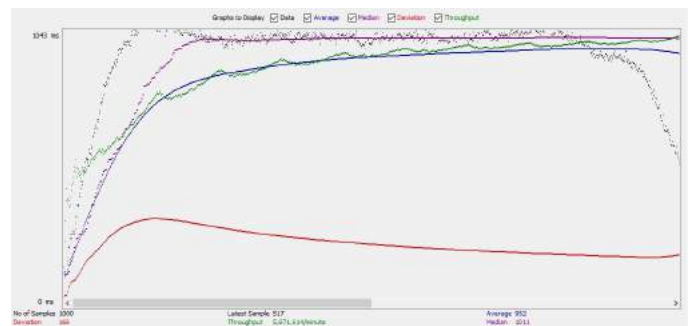
TABEL XV
KONFIGURASI YANG DIGUNAKAN UNTUK PENGUJIAN BEBAN

Path	/DATA-LAKE?X-DEVICE-KEY=
Method	2BCDF8A11245826A7248556300171963 POST
Data (JSON)	{ "deviceId": "5f8e27171251f001af3fa6b0", "data": { "voltage": 73, "current": 50 } }
Sampel	100
Selang waktu	1 detik
Pengulangan	10

TABEL XVI
HASIL REQUEST LOAD TESTING KE SERVER

# Sampel	1000
Rata-rata	1045
Median	1097
Min	80
Max	1085
Std. Dev.	171.18983689460018
Error %	0.00%
Throughput	86.69267446900737/detik
Ukuran data yang diterima (Kb)	65.27348829648895/detik
Ukuran data yang dikirim (Kb)	30.393232553099264/detik
Avg. Bytes	771.0

Berdasarkan sampel yang dikirimkan mendapatkan hasil yang baik tanpa adanya error yang artinya masih bisa menangani data yang lumayan banyak di satu waktu.



Gambar 30. Grafik performa transaksi

Ket:

Hitam Total jumlah sampel yang dikirim saat itu

Biru Rata-rata dari semua sampel yang dikirim saat itu.

Merah Deviasi standar saat ini.

Hijau Jumlah permintaan per menit yang sukses ditangani server.

Throughput adalah parameter terpenting. Parameter tersebut

mewakili kemampuan server untuk menangani beban berat. Semakin tinggi throughput, semakin baik kinerja server.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil dan pengujian yang telah dilakukan dapat disimpulkan bahwa, sistem yang di buat dapat menyimpan data dari sensor kemudian data tersebut dapat di tampilkan secara visual pada dashboard pengguna dengan baik. Dashboard itu juga dapat di rangkai dengan cepat menggunakan fitur Dashboard Builder tanpa perlu repot untuk menulis baris kode yang rumit, semua tersedia dengan pengaturan secara visual. Sistem ini juga dapat memproses data secara realtime sesuai instruksi yang telah di rangkai juga secara visual menggunakan fitur Rule Editor.

Sistem berjalan sesuai dengan yang di rencanakan tanpa ada error pada saat melakukan permintaan pengiriman 100 sampel data dengan 10 kali pengulangan dan interval tiap 1 detik secara bersamaan, rata-rata pengiriman data 1045. Kecepatan penanganan (handle) data mencapai 86.6 per detik dengan deviasi standart yang rendah.

B. Saran

Sistem yang telah di rancang ini masih memerlukan beberapa peningkatan untuk menstabilkan di antaranya:

1. Sebaiknya menggunakan library yang telah support WebGL untuk menangani visualisasi data yang banyak untuk mendapatkan performa yang baik.
2. Pada sistem ini belum menyediakan layanan analitik yang maksimal maka dari itu sebaiknya dapat menerapkan platform analitik contohnya Cube.js.
3. Setiap request data masih belum di batasi jadi ada kemungkinan beban request tidak akan seimbang maka dari itu sebaiknya tambahkan manajemen koneksi.

V. KUTIPAN

- [1] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M., 2015. "Internet of things: A survey on enabling technologies, protocols, and applications". IEEE communications surveys & tutorials, 17(4), pp.2347-2376.
- [2] Botta, A., De Donato, W., Persico, V. and Pescapé, A., 2016. "Integration of cloud computing and internet of things: a survey". Future generation computer systems, 56, pp.684-700.
- [3] Habibi, M.W., Bhawiyuga, A. and Basuki, A., 2018. "Rancang Bangun IOT Cloud Platform Berbasis Protokol Komunikasi MQTT". Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN, 2548, p.964X. Rizkyana, M.A. and Amalga, R.S.G., 2015, July. "Rancangan Arsitektur Aplikasi Pengumpulan Tugas dengan Push Notification Real-Time Menggunakan NodeJs". In Seminar Nasional Informatika (SEMNASIF) (Vol. 1, No. 1).
- [4] Mustaqbal, M. Sidi; Firdaus, Roeri Fajri; Rahmadi, Hendra., 2015. "Pengujian aplikasi menggunakan black box testing boundary value analysis (studi kasus: Aplikasi prediksi kelulusan smnptn)". Jurnal Ilmiah Teknologi Infomasi Terapan, 1.3.
- [5] Ontowirjo, F.Y.Q. 2018. "Implementasi IoT Pada Sistem Monitoring Suhu dan Kelembaban Pada Ruangan Pengeri Berbasis Web". Skripsi Program S1 Teknik Elektro Universitas Sam Ratulangi. Manado.

- [6] Octavian, B.Y. 2019. "Sistem Monitoring Intensitas Cahaya dan Kcepatan Angin Berbasis Web di Kawasan Relokasi Korban Banjir Pandu". Skripsi Program S1 Teknik Elektro Universitas Sam Ratulangi. Manado.



Imanuel E. P. Pundoko (M'76–SM'81–F'87) penulis adalah anak ke dua dari dua bersaudara dalam keluarga Pundoko Sangian, lahir di Manado pada tanggal 07 Mei 1999. Penulis menempuh pendidikan pertama di sekolah Taman Kanak-kanak GMIM Getsemani Sea pada tahun 2003 sampai 2004, kemudian melanjutkan pendidikan di Sekolah Dasar Katolik St. Yosep Kec. Pineleng pada tahun 2004 sampai 2010, setelah itu masuk ke Sekolah Menengah Pertama Advent 2 Sario Manado pada tahun 2010 sampai 2013, kemudian melanjutkan pendidikan di Sekolah Menengah Kejuruan Negeri 1 Manado 2013 hingga lulus di tahun 2016 dengan menempuh kompetensi Teknik Komputer dan Jaringan. Sekarang penulis telah menyelesaikan pendidikan di Universitas Sam Ratulangi Manado.

Selama menempuh Pendidikan di SMKN 1 Manado penulis pernah mengikuti program magang di pusat servis Great Power selama 4 bulan, kemudian melaksanakan kegiatan non-akademik kerja praktek di Unit Layanan Pembangkit Listrik Tenaga Air Tanggari selama 2 bulan saat masih menjadi mahasiswa. Penulis pernah menjadi relawan beberapa projek dari pemerintahan untuk pembuat beberapa aplikasi layanan masyarakat.