

Implementasi LoRaWAN Menggunakan Platform Chirpstack pada Proses Pengiriman Data Pada Perangkat Tracking Milik Transtrack

1st Mikael Esaputra Wibowo
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

mikaelewibowo@student.telkomuniversity.ac.id

2nd Periyadi
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

periyadi@telkomuniversity.ac.id

2nd Henry Rossi Andrian
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

henryrossiandrian@telkomuniversity.ac.id

Abstrak — PT. Indo Trans Teknologi adalah perusahaan yang bergerak di bidang penyediaan layanan manajemen logistik dan tracking kendaraan. Produk-produk *hardware* yang ditawarkan memiliki fitur *tracking* dan ekstensi *port i/o* agar dapat diintegrasikan dengan perangkat lain. Dalam mengirimkan data produk ini mengandalkan sinyal GSM sebagai media pengirimannya. Ini menjadi sebuah kelemahan karena produk ini tidak dapat terpasang di daerah yang tidak terdapat menara GSM. Maka dari itulah peran LoRaWAN dapat menjadi solusi jika alat ini harus dipasang ditempat yang tidak terjangkau menara GSM. Sistem LoRaWAN yang dibuat akan menggunakan *chip* LoRa-E5-Hf sebagai mikrokontroler pada *end-node*, SenseCap M2 Gateway sebagai *gateway* LoRaWAN, dan platform ChirpStack sebagai *server* LoRaWAN. Hasinya implementasi LoRaWAN menggunakan platform Chirpstack berhasil dilakukan, *end-node* dapat melakukan *uplink* ke Chirpstack, dan dengan menggunakan *chip* LoRa-E5-Hf, dan SenseCap M2 gateway, data dapat terkirim sejauh maksimum 415m pada daerah perkotaan dan perkantoran.

Kata kunci— LoRa, LoRaWAN, ChirpStack, LoRa-E5-Hf, SenseCap M2 Gateway

I. PENDAHULUAN

PT. Indo Trans Teknologi (Transtrack) merupakan sebuah perusahaan yang menyediakan layanan solusi pada manajemen operasi armada transportasi, optimalisasi *supply chain integrator* untuk meningkatkan produktivitas dan menekan biaya, serta memberikan solusi pada masalah keamanan pada kendaraan. Produk yang ditawarkan perusahaan ini sudah mengimplementasikan teknologi *Internet of Things* (IoT) agar perangkat dapat dianalisis, *di-monitoring*, maupun dilacak dari jarak jauh menggunakan aplikasi *Fleet Management System* (FMS) yang telah disediakan[1].

Transtrack menawarkan perangkat *tracking*[2] yang mengandalkan modul GSM untuk mengirim data ke *server* mereka, yang dimana alat ini bergantung pada infrastruktur menara GSM. Ketergantungan ini dapat menyebabkan masalah jika perangkat berada di luar jangkauan menara, mengakibatkan data seperti posisi dan nilai sensor tidak

terkirim dengan akurat dan mempengaruhi biaya operasional. Sebagai alternatif, disarankan untuk menggunakan LoRa (*Long Range*), yaitu teknologi berbasis *Radio Frequency* pada *Low-Power Wide Area Network* (LPWAN)[3]. LoRa distandarisasi oleh Semtech dan diatur oleh LoRa Alliance, menawarkan jangkauan yang lebih luas (hingga 15 kilometer di area rural), konsumsi daya rendah, dan skalabilitas yang baik, menjadikannya opsi yang lebih handal dan efisien dibandingkan GSM untuk pengiriman data nirkabel.

Maka dari itu perusahaan akan mengadakan project merancang sistem LoRaWAN untuk mengirimkan data-data perangkat *tracking* milik mereka sebagai opsi lain selain penggunaan GSM. Project ini berisikan 4 orang magang sebagai IoT Engineer termasuk penulis. Penulis diberi tugas dari perusahaan untuk membuat dan merancang sistem LoRaWAN lalu melaporkan keefektifannya dalam pengiriman data[4].

II. KAJIAN TEORI

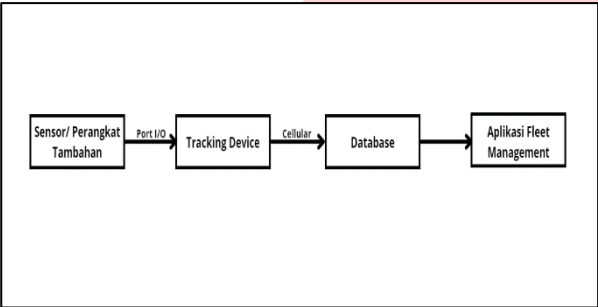
Transtrack adalah sebuah perusahaan yang bergerak di bidang Teknologi IoT pada transportasi dan logistik. Kebanyakan produk perangkat keras milik Transtrack memiliki fitur *tracker* untuk melacak posisi perangkat tersebut. Perangkat – perangkat ini biasanya dilengkapi dengan *port I/O* untuk mengintegrasikan perangkat atau sensor lain, dan modul GSM untuk mengirimkan data ke database perusahaan[2]. Karena penggunaan GSM menjadi satu-satunya media pengiriman data, terkadang ada suatu kasus dimana disebuah tempat tidak terdapat infrastruktur untuk menunjang kebutuhan modul GSM, contohnya di area tambang[5]. Client perusahaan tambang membutuhkan fitur *tracker* milik Transtrack untuk melacak semua alat-alat berat milik perusahaan tersebut di sekitaran area tambang mereka.

Maka dari itu Transtrack membutuhkan alternatif lain dalam pengiriman data pada perangkat mereka. Maka diajukanlah sistem komunikasi LoRaWAN sebagai media pengiriman data alternatif. LoRaWAN menjadi pilihan alternatif pengiriman data selain GSM dikarenakan LoRaWAN memiliki keunggulan yaitu konsumsi energi yang rendah serta jangkauan radius sinyal LoRa yang cukup jauh[6].

Namun ada beberapa kelemahan pada LoRaWAN, yaitu ketika disebuah gateway terdapat banyak end-node maka ini akan mempengaruhi jumlah *packet loss* yang diterima oleh *server* LoRaWAN[7]. LoRaWAN merupakan protokol yang baru, maka masih terdapat *vulnerability* pada proses *join request* sampai pengiriman data[8].

Ada beberapa contoh pembuatan alat GPS *tracker* berbasis LoRaWAN yang pernah dibuat dan juga komparasi performa modul LoRa yang dapat menjadi referensi pada proyek kali ini[9]. Pada kesempatan kali ini penulis menggunakan microcontroller STM32WLE5JC yang didalamnya sudah terintegrasi modul LoRa milik Semtech. Desain *end-node* akan menjadi kompak dan kecil sehingga menghemat biaya produksi PCB dan menghemat tempat untuk komponen komponen lainnya yang diperlukan.

III. METODE



GAMBAR 1
Gambaran Sistem Saat Ini

Gambar 1 merupakan gambaran secara umum sistem dan cara kerja produk perangkat keras milik Transtrack. *Tracking Device* merupakan perangkat yang dipasang di suatu tempat dengan tujuan utama untuk membaca data lokasi global lalu mengirimkannya ke database milik Transtrack menggunakan GSM. Selain itu perangkat ini juga di lengkapi dengan *port i/o* untuk mendapatkan data tambahan dari sensor atau perangkat tambahan lain tergantung tujuan pemasangan perangkat. Data data ini juga dikirimkan melalui modul GSM bersamaan dengan data lokasi global. Lalu pada *server database* data yang diterima dari *tracking device* akan di dekripsi berdasarkan protokol yang digunakan *tracking device* saat mengirimkan data. Data yang telah didekripsi akan disimpan dan ditampilkan pada aplikasi *fleet management* dalam bentuk laporan *log* dan *chart* kepada *client*.

Pada sistem saat ini perangkat *tracking* hanya mengandalkan *cellular/GSM* sebagai pengiriman data. Hal ini membuat perangkat hanya dapat bekerja pada area yang terjangkau sinyal GSM seperti di area tambang. Maka dari itu dibutuhkan LoRaWAN sebagai opsi lain untuk pengiriman data. Untuk membangun sistem LoRaWAN dibutuhkan hal sebagai berikut :

TABEL 1
Kebutuhan Fungsional

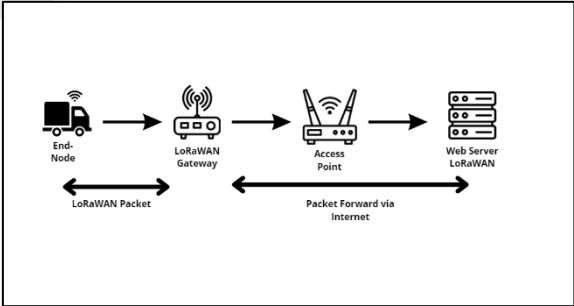
No	Kebutuhan Fungsional
1.	Sistem dapat mendaftarkan Join Eui, Dev Eui, dan Appkey pada <i>end-node</i> yang akan dipasang.
2.	<i>End-node</i> yang telah terdaftar dapat <i>join</i> ke jaringan LoRaWAN.
3.	<i>End-node</i> dapat mentransmisikan data ke <i>gateway</i> sejauh mungkin, lalu di-forward ke server LoRaWAN.
4.	Sistem dapat mengabaikan atau menolak <i>join</i> dari <i>end-node</i> yang <i>join</i> eui,dev eui, dan appkey-nya tidak terdaftar pada server LoRaWAN.

TABEL 2
Kebutuhan Non-fungsional

No	Kebutuhan Non-Fungsional
1.	Sistem harus memastikan keamanan data yang dikirim dan diterima oleh perangkat dengan mengenkripsi data tersebut.
2.	<i>Packet loss</i> saat pengiriman data menggunakan LoRa harus seminim mungkin.
3.	Sistem harus dapat mendeteksi <i>end-node</i> sejauh mungkin dengan konsumsi daya seminimal mungkin.
4.	Sistem dapat menangani banyak <i>end-node</i> dan <i>gateway</i>

Dari penjelasan diatas maka dibangunlah sebuah sistem LoRaWAN dengan rancangan sebagai berikut :

A. Rancangan Sistem

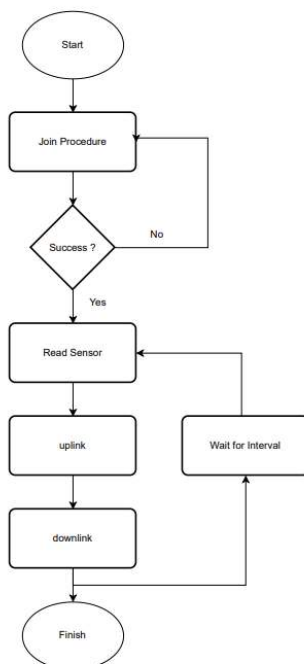


GAMBAR 2
Rancangan Sistem LoRaWAN

Gambar 2 merupakan diagram sistem LoRaWAN yang dibangun Penulis. Terdapat perangkat perangkat seperti :

- *End-node* adalah perangkat yang akan dipasang di lapangan yang bertujuan untuk mengambil data data sensor seperti temperatur, kelembapan, lokasi, dan lain sebagainya untuk dikirim ke *web server* LoRaWAN melalui LoRaWAN gateway.
- LoRaWAN gateway adalah perangkat yang diletakan *stationary* di suatu tempat yang bertujuan sebagai tempat berkumpulnya data-data yang dikirim dari *end-node*, kemudian diteruskan ke *web server* LoRaWAN melalui *access point*.
- *Access Point* adalah perangkat yang berperan sebagai perpanjangan tangan dari LoRaWAN gateway agar data dari perangkat tersebut dapat diteruskan ke *web server* LoRaWAN melalui *internet*.
- *Web server* LoRaWAN merupakan tempat penyimpanan data dan pengolahan jaringan LoRaWAN.

B. Flowchart End-Node



GAMBAR 3
Flowchart End-Node

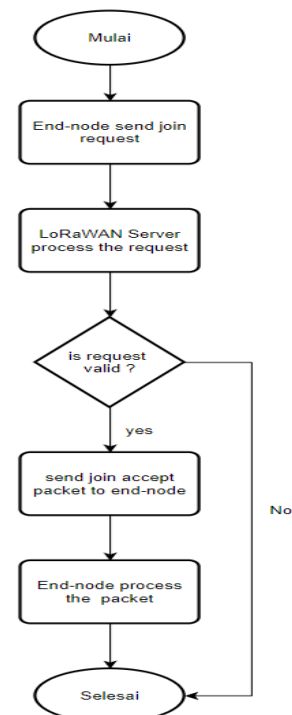
End-node adalah perangkat pada jaringan LoRaWAN yang terpasang di lapangan. *End-node* memiliki fungsi utama yaitu membaca data sensor lalu mengirimkannya menggunakan *uplink* ke *server* LoRaWAN. Gambar 3 merupakan diagram alur pada *firmware end-node*, berikut penjelasan setiap tahapnya :

- Pertama, sebelum melakukan transmisi data *end-node* terlebih dahulu melakukan proses *join*. Proses ini berfungsi agar *end-node* teridentifikasi oleh *server* LoRaWAN. Hasil dari proses ini adalah sebuah jawaban dari server

LoRaWAN yang berisi *NwkSKey*(*Network Sesion Key*) dan *AppSKey*(*Application Sesion Key*) yang digunakan untuk mengenkripsikan/mengdekripsikan data selama transmisi, *NetID*(*Network ID*) sebagai identitas *end-node* pada jaringan LoRaWAN, dan juga parameter konfigurasi *interval* Rx, Tx selama *uplink/downlink* pada *end-node* untuk mensinkronisasi komunikasi LoRaWAN.

- *End-node* akan terus melakukan proses *join* hingga berhasil. Jika proses *join* berhasil maka *end-node* akan memulai proses pengiriman data. Pada perancangan sistem ini *end-node* menggunakan class A, sehingga setiap kali melakukan *uplink*, *end-node* akan melakukan *downlink* dua kali selama waktu yang sudah ditentukan server LoRaWAN.
- Pada proses pengiriman data, *end-node* akan membaca data sensor yang digunakan. Lalu data data tersebut di-*uplink* ke *server* LoRaWAN, setelah itu *end-node* akan membuka jendela *downlink* sebanyak 2 kali untuk menerima jawaban dari *server*. Jika *uplink* berhasil mencapai server maka server akan membalas pesannya dengan nilai RSSI, SNR, dan juga data yang berhasil terkirim.
- Proses pengiriman data ini terjadi berulang ulang sampai *end-node* diterminasi.

C. Algorithma Join Procedure



GAMBAR 4
Flowchart Join Procedure

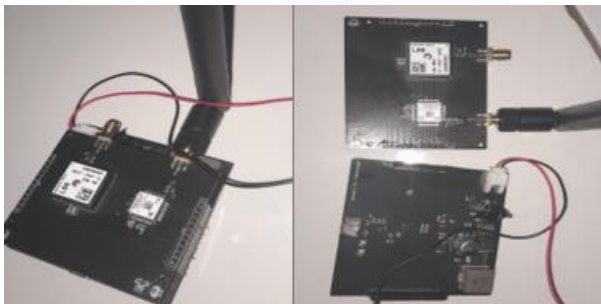
Join Procedure adalah proses bergabungnya *end-node* ke jaringan LoRaWAN ketika pertama kali perangkat dinyalakan. Berikut penjelasan alur *join procedure* :

- Pertama, *end-node* akan mengirimkan *packet join request* yang berisi JoinEui, DevEui, DevNonce, dan *Message Integrity Code* (MIC) ke LoRaWAN server.
- Setelah itu *join request* diterima oleh *server* lalu dicek apakah identitas yang terdapat pada *packet* tersebut terdaftar pada *server* LoRaWAN. Jika identitas terdaftar maka *server* LoRaWAN akan mengirimkan *packet* balasan yang berisi Appnonce, NetID, DevAddr, DLSetting, RXDelay, CFList. *Packet* ini akan dienkripsi dengan AppKey sebelum dikirim kembali ke *end-node*. Namun jika *packet* yang diterima tidak terdaftar atau MIC tidak sesuai maka *server* akan mengabaikan *request* tersebut.
- *Join accept packet* diterima oleh *end-node* lalu di dekrip menggunakan AppKey yang sama. *End-node* akan menerapkan konfigurasi DLSetting, dan RXDelay pada dirinya sendiri, lalu menyimpan data NwkSKey, AppSKey, NetID, dan DevAddr untuk keperluan *uplink* dan *downlink* data.

IV. HASIL DAN PEMBAHASAN

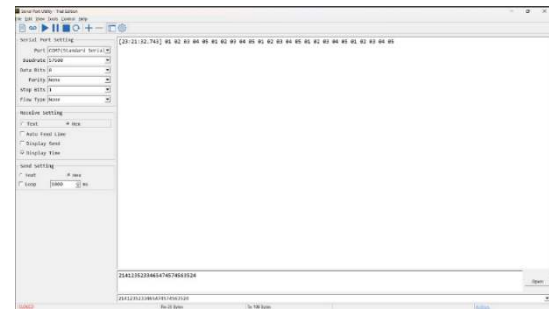
Sebelum melakukan pengujian, perangkat *end-node* dan *gateway* terlebih dahulu di konfigurasi dan didaftarkan ke *platform* chirpstack.

A. Pemasangan Komponen Perangkat *End-node* dan *Testing Peer to Peer*



GAMBAR 5
Module End-Node

Komponen dan pcb yang diperlukan dirangkai dan disolder sehingga membentuk 2 modul yaitu modul LoRa sebagai main modul untuk pengiriman data sensor yang dimodulasi menggunakan LoRaWAN, dan modul *power supply* untuk menurunkan sekaligus melindungi komponen lain yang ada di PCB dari loncatan arus/ tegangan. Kemudian, kedua modul tersebut dipasang secara vertikal dan jadilah bentuk akhir dari perangkat *end-node* seperti gambar 5. Untuk mengetahui apakah perangkat ini



GAMBAR 6
Testing Peer to Peer

dapat berjalan dengan semestinya, penulis mengupload *firmware* pengiriman data *peer to peer* ke *end-node* dimana salah satu *node* akan berperan sebagai *reciever* dan *node* yang lain sebagai *transmitter*. Data yang dikirim dari *transmitter* bertipe data *unsigned integer* dalam bentuk array dengan isi datanya {1,2,3,4,5}. Data tersebut dikirim dengan frekuensi 923Mhz. Hasilnya pada gambar 6 data yang dikirimkan *transmitter* berhasil diterima oleh *reciever*.

B. Konfigurasi Gateway



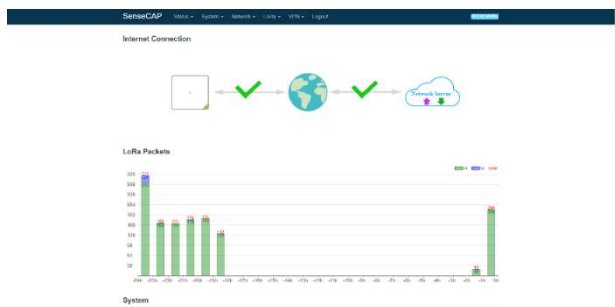
GAMBAR 7
Interface Gateway Sensecap LoRAWAN

Gateway membutuhkan sambungan *internet* untuk terkoneksi ke *server* LoRaWAN, oleh karena itu *gateway* perlu dikonfigurasi agar *gateway* dapat terkoneksi ke *access point*. untuk mengkonfigurasi *gateway*, *gateway* perlu masuk ke mode konfigurasi dengan menahan button selama 5 detik sampai LED WLAN menyala biru seperti gambar 7. gunakan laptop, lalu pada menu wifi terdapat SSID SenseCAP_XXXXXX, koneksikan laptop dengan SSID tersebut. Pastikan IP laptop pada *interface* WLAN di-setting dinamis, lalu akses 192.168.168.1 untuk masuk ke menu antarmuka, lalu masukan *id password* sesuai dengan yang tertera pada *gateway* seperti gambar 8



GAMBAR 8
Identitas Sensecap Gateway

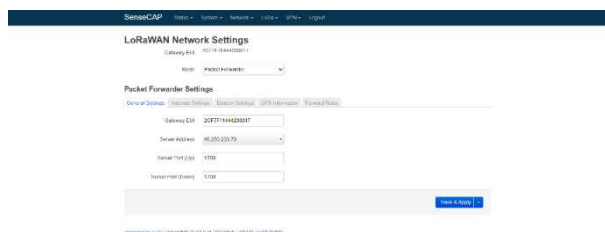
Jika berhasil maka akan masuk ketampilan awal seperti gambar 9.



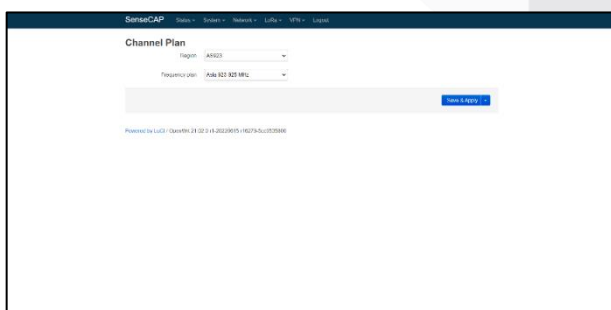
GAMBAR 9

Menu Awal Tampilan Konfigurasi Sensecap Gateway

Untuk dapat menkoneksi gateway ke access point pergi ke *network > wireless* lalu *scan* sinyal WiFi yang tersedia, lalu *join network* pada SSID access point, masukan *password*-nya lalu *save*. Jika berhasil terkoneksi ke internet maka pada menu awal akan ada tanda centang seperti gambar 9. setelah itu gateway perlu dikonfigurasi untuk *forward packet* LoRaWAN dari gateway ke *web-server* LoRaWAN. Untuk mengkonfigurasi hal tersebut pertama pergi ke *LoRa > Channel Network* dan akan memunculkan tampilan seperti gambar 10.

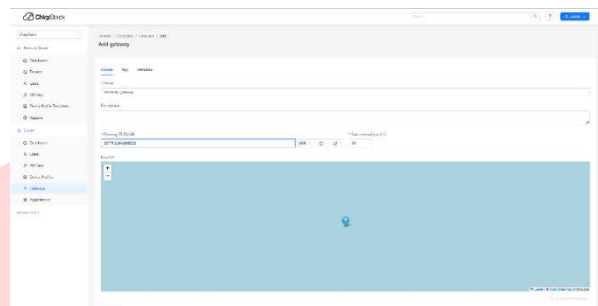
Gambar 10
Menu Channel Network

Pada menu ini, gateway perlu di-setting ke *mode packet forwarder*, diberikan gateway EUI, serta mengatur target alamat *server* LoRawan beserta *port*-nya. Setelah itu atur frekuensi gateway padata menu *LoRa>Channel plan*, lalu pilih frekuensi AS923 seperti gambar 11. untuk menentukan gateway eui dapat menggunakan angka acak menggunakan bantuan *random eui generator* yang tersedia di *internet*.

GAMBAR 11
Menu Channel Plan

C. Pendaftaran *End-node* dan *Gateway* pada *Web-server* Chipstack

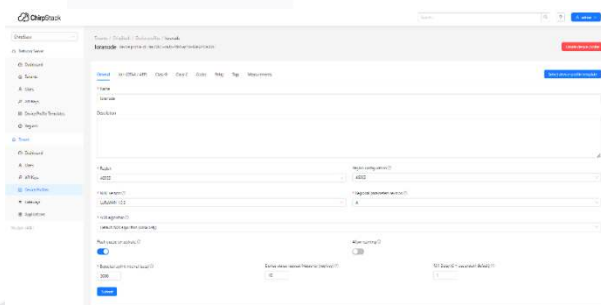
Agar sistem LoRaWAN dapat berjalan, perangkat *end-node* memerlukan DEVEui, JoinEui, dan Appkey. Untuk mendapatkan 3 hal ini, penulis mendaftarkan gateway ke *web server* LoRaWAN terlebih dahulu. Untuk mendaftarkan gateway pada *web server* ini, pilih menu *gateway>add gateway* lalu isi kolom *name*, *gateway id* sesuai dengan yang telah diatur pada menu konfigurasi gateway sebelumnya, dan *set interval* data seperti gambar 12, lalu klik *submit*.



GAMBAR 12

Pendaftaran Gateway pada Chirpstack

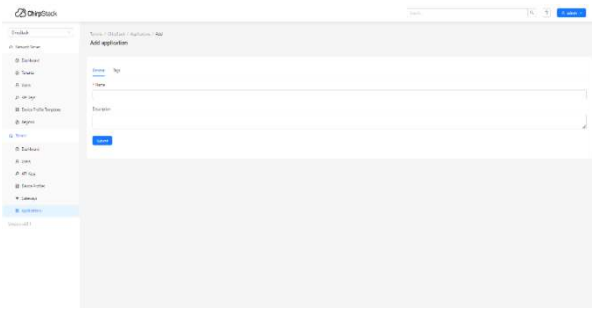
Jika gateway berhasil didaftarkan, selanjutnya penulis mendaftarkan *Device Profile*. *Device profile* ini berfungsi untuk menentukan frekuensi LoRaWAN yang digunakan pada perangkat *end-node*. Karena *end-node* yang digunakan menggunakan menggunakan frekuensi AS923, maka pada menu ini penulis mengatur pengaturan *device profile* seperti gambar 13.



GAMBAR 13

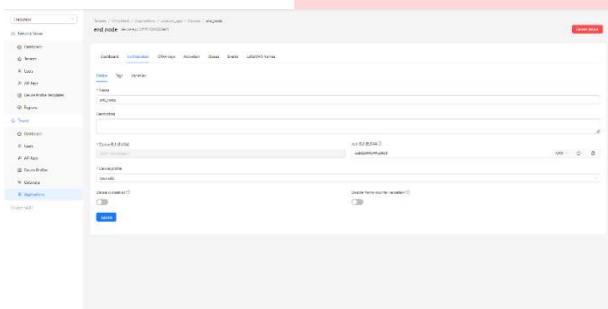
Kofigurasi Profile Device pada Chirpstack

Setelah *device profiles* berhasil dibuat, maka selanjutnya penulis mendaftarkan *end-node* dengan membuat profil aplikasi terlebih dahulu dengan memilih tombol *add application* lalu masukan *name*, dan *description*(opsional).



GAMBAR 14
Mendaftarkan Aplikasi pada Chirpstack

Setelah aplikasi dibuat, selanjutnya penulis mendaftarkan perangkat *end-node* pada menu *device* di aplikasi yang sebelumnya dibuat. pada pengaturan *end-node*, *join eui* pada menu konfigurasi dapat dibuat dengan *generate* angka acak dengan menekan tombol tanda panah



GAMBAR 15
Tampilan Konfigurasi Device pada Chirpstack

melingkar, begitu pula dengan *Application key* pada menu *OTAA keys*, *Network Session Key*, dan *Application Session Key* pada menu *activation*. *Join EUI* adalah identitas milik *end-node* ketika melakukan *join procedure* ke server LoRaWAN, *AppKey* adalah kunci untuk mengenkripsi data ketika *end-node* melakukan *join procedure* hasil dari proses ini *end-node* mendapatkan *application session key* (AppSKey) dan *network session key* (NwkSKey). AppSKey dan NwkSKey digunakan untuk mengenkripsi data *uplink* dan *downlink*.

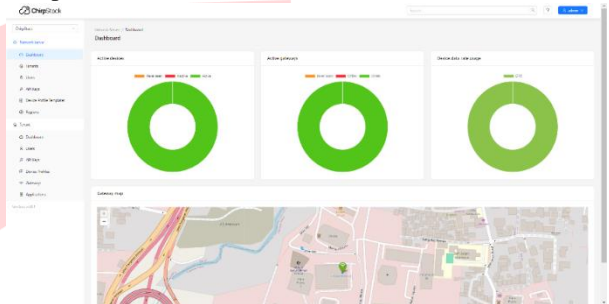
D. Konfigurasi Firmware LoRaWAN End-Node

Setelah mendaftarkan *end-node* pada *web-server* chirpstack, *firmware* pada perangkat *end-node* perlu diatur ulang khususnya pada bagian *Device EUI*, *Join EUI*, *AppKey* dan *NwkKey*. Bagian bagian ini diatur ulang mengikuti pengaturan yang sudah didaftarkan di chirpstack. Untuk DevEUI menggunakan *Gateway ID* pada pengaturan *gateway* sebelumnya, *Join EUI* menggunakan *JoinEui* yang terdaftar pada Chirpstack, dan *App Key* serta *Nwk Key* keduanya menggunakan *App key* pada *device* yang sudah didaftarkan pada Chirpstack.



GAMBAR 16
Konfigurasi Identitas End-Node

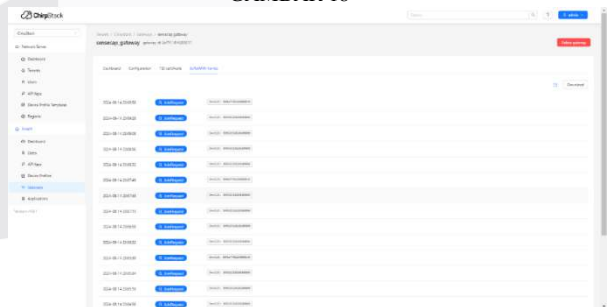
E. Tampilan antarmuka Server Lorawan



GAMBAR 17
Tampilan Dashboard Chirpstack

Gambar 17 adalah tampilan awal pada *web-server* LoRaWAN menggunakan chirpstack. *Web* ini dapat digunakan untuk memantau status, pendaftaran perangkat, dan juga mengawasi lalu lintas data yang masuk ke *end-node* atau *gateway*. Pada gambar 17 melaporkan status perangkat *end-node* dan *gateway* yang sudah didaftarkan pada *web-server* tersebut. Selain itu terdapat posisi lokasi *gateway* yang terdaftar pada *web* tersebut. *Web* ini menyediakan menu *log* data pada *gateway* dan *end-node* yang terdaftar. Menu *log* ini berfungsi untuk memantau lalu lintas data pada perangkat LoRaWAN yang terdaftar di *web* tersebut.

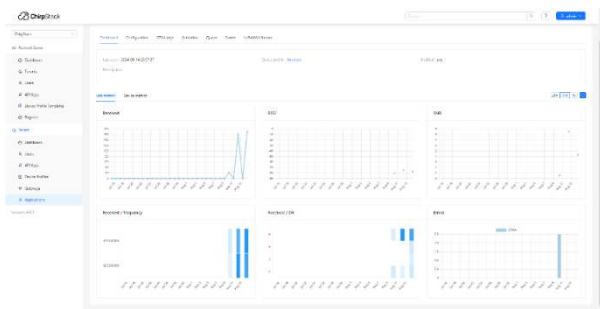
GAMBAR 18



Log Data Gateway

Contohnya pada gambar 18 merupakan tampilan *log* data pada *gateway*. Terdapat beberapa *end-node* yang mengirim *join request* yang masuk namun diabaikan oleh server karena DevEUI milik *node* tersebut tidak terdaftar di server. Selain itu *web* ini juga menyediakan dashboard aktifitas data yang melewati perangkat *gateway* atau *end-node* yang telah terdaftar. Contohnya pada gambar 19 yang menunjukkan laporan seperti jumlah data *receive*, *RSSI*, *SNR*, *Frequency* yang digunakan, *DR* (Data Rate),

dan *data error* yang diterima oleh *end-node* yang sudah didaftarkan.



GAMBAR 19
Dashboard End-Node

Setelah *gateway* dan *end-node* berhasil terkoneksi dan dikonfigurasi dan *end-node* dapat berkomunikasi dengan *server chirpstack* maka akan dilakukan pengujian pengiriman data sebagai berikut:

A. Lokasi

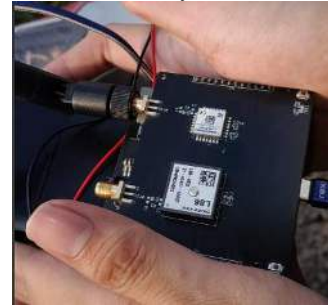


GAMBAR 20
Lokasi Menara 165

Pengujian alat akan dilakukan di sekitar kantor PT Indo Trans Teknologi yang berlokasi di Menara 165, Jakarta Selatan. Lokasi ini terletak di kawasan padat dengan banyak gedung tinggi dan permukiman padat penduduk, yang dapat menyebabkan tantangan dalam pengujian sinyal, seperti *multipath fading* dan interferensi. Hambatan fisik seperti dinding dan pepohonan serta interferensi dari perangkat elektronik lain di sekitar area juga dapat mempengaruhi kualitas sinyal dan jangkauan efektif perangkat yang diuji, terutama untuk teknologi seperti LoRaWAN yang bergantung pada komunikasi nirkabel jarak jauh.

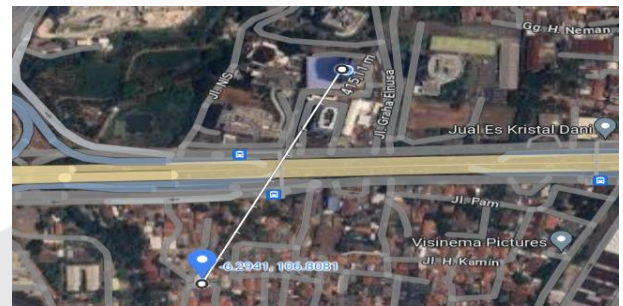
Untuk mengatasi tantangan ini, *gateway* akan ditempatkan di dekat jendela di kantor Transtrack, agar sinyal dapat lebih mudah menembus keluar dan mencapai perangkat IoT di luar gedung. Penempatan ini bertujuan untuk meminimalkan hambatan yang disebabkan oleh struktur bangunan dan mengoptimalkan penerimaan serta pengiriman sinyal antara perangkat dan *gateway*. Dengan lokasi yang strategis ini, diharapkan jalur komunikasi yang lebih terbuka dapat mengurangi efek penyerapan dan pantulan sinyal, sehingga meningkatkan kinerja alat yang diuji.

B. Uji Coba Menemukan Sinyal LoRaWAN



GAMBAR 21
End-Node Melakukan Join Procedure

Uji coba kali ini penulis melakukan penelusuran ke tempat-tempat di sekitar area kantor untuk menemukan tempat yang dapat mengirim sinyal LoRa dari *end-node* menuju *gateway*. Penulis menggunakan *end-node* untuk mendeteksi kemungkinan sinyal LoRa dapat terkirim ke *server gateway*. Pada *end-node* terdapat mekanisme *Join Procedure* dimana *end-node* mengirimkan *join request* ke *web-server* LoRaWAN terus menerus hingga *server* menerima pesan dan membalas *request* tersebut dengan mengirimkan *session key*. Selama melakukan *join procedure*, LED pada perangkat *end-node* akan berkedip berkali-kali hingga *end-node* mendapatkan jawaban dari *web-server*. Hasilnya sebagian besar area jangkauan LoRaWAN *gateway* adalah titik buta. Ini dikarenakan daerah sekitar kantor TransTrack dipenuhi oleh gedung-gedung tinggi sehingga sinyal LoRa tidak dapat sampai ke *gateway*. Meski begitu terdapat beberapa tempat yang dimana *end-node* berhasil melakukan *join procedure* seperti contohnya titik pada gambar 22.



GAMBAR 22
Titik Terjauh Pengiriman LoRaWAN

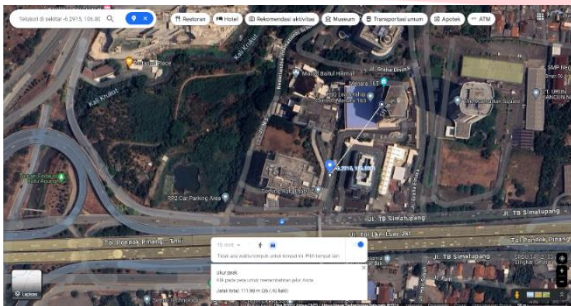
Pada titik ini *end-node* dapat mengirimkan *packet data* LoRaWAN, dan merupakan titik terjauh yang didapat oleh penulis selama melakukan penelusuran sinyal. Ini bisa terjadi karena pada titik ini, *end-node* berada di daerah yang ideal dalam pengiriman data LoRaWAN karena *end-node* berada di area jarak pandang dari lokasi ruangan pemasangan *gateway*. Maka dari itu, data tidak akan terhalang oleh gedung apapun. Bisa terlihat pada gambar 23, Kantor Transtrack yang berada di lantai 6 masih dapat terlihat dari titik ini.



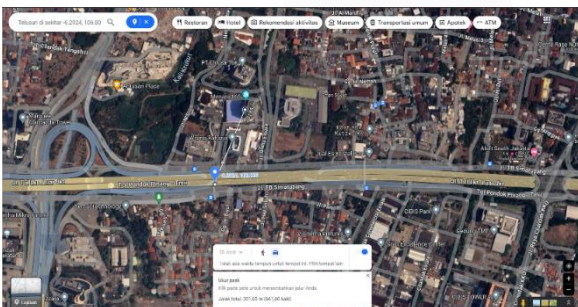
GAMBAR 23
Lingkungan Titik Terjauh Pengiriman LoRaWAN

C. Uji Coba Kekuatan Sinyal dan *Packet Loss*

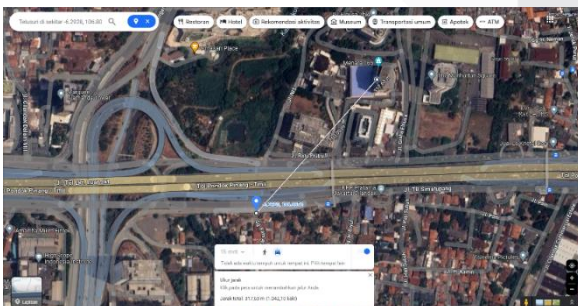
Setelah menemukan titik lokasi terjauh, dan menyimpulkan bahwa lokasi pengiriman paket yang ideal harus berada di area jarak pandang dengan *gateway* maka penulis menentukan 3 tempat lainnya yang masih berada di daerah jarak pandang *gateway* yang berjarak 100m, 200m, dan 300m dari *gateway* untuk dilakukan uji coba pengukuran kekuatan sinyal dan *packet loss*. 3 lokasi itu berada di titik gambar dibawah ini :



GAMBAR 24
Titik 100m dari Gateway



GAMBAR 25
Titik 200m dari Gateway



GAMBAR 26
Titik 300m dari Gateway

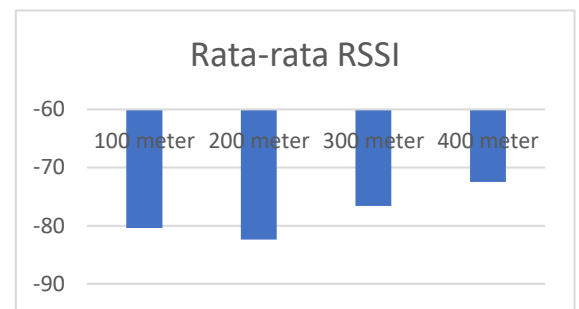
Dari 4 titik yang telah ditentukan tersebut penulis melakukan *logging* data sebanyak 10 kali *uplink* data pada *end-node*. *Logging* pada *end-node* akan menggunakan *Serial Port Utility*, sedangkan pada *web-server* akan menggunakan menu

logging pada menu *end-node*. Terdapat 2 hal yang akan diukur yaitu kekuatan sinyal yang diukur dengan parameter rata rata nilai RSSI dan SNR pada data yang berhasil diterima dan *packet loss*. Hasil pengukuran :

1. Pengukuran RSSI

Tabel 3
Data RSSI yang Terbaca di Chirpstack

Jarak	100m	200m	300m	400m
RSSI	-76	-85	-75	-64
	-83	-85	-75	-67
	-81	-84	-75	-74
	-79	-85	-72	-69
	-81	-81	-78	-77
	-80	-80	-83	-79
	-78	-79	-76	-77
	-83	-85	-79	-73
	-82	-80	-	-
	-81	-80	-	-
rata - rata	-80,4	-82,4	-76,625	-72,5



GAMBAR 27
Chart Rata-Rata RSSI

RSSI (*Received Signal Strength Indicator*) adalah satuan pengukuran kekuatan sinyal ketika data berhasil terkirim ke *receiver*. Semakin mendekati 0 semakin kuat kekuatan sinyal. Pada gambar 27 menunjukkan bahwa nilai RSSI naik ketika berada di 200 meter. Namun pada jarak 300 meter dan 400 meter RSSI menunjukkan penurunan nilai. Ini bisa terjadi karena pada jarak 100 meter dan 200 meter lokasi pengiriman data berada di dekat gedung gedung besar.



GAMBAR 28
Lingkungan pada Jarak 200m dan 400m

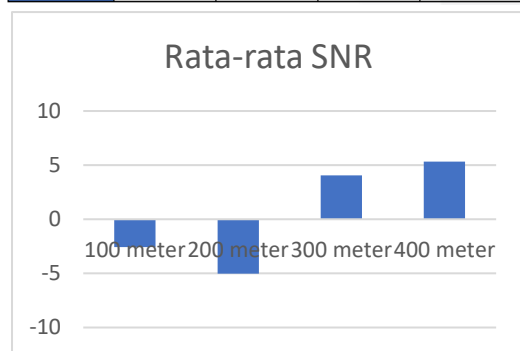
Kemungkinan besar di daerah ini terdapat banyak sekali sinyal yang menginterferensi sinyal LoRa. Sedangkan pada

jarak 300 meter dan 400 meter nilai RSSI menurun karena lokasi pengiriman data berada di daerah perumahan sehingga sedikit sekali sinyal yang akan menginterferensi sinyal LoRa.

2. Pengukuran SNR

TABEL 4
Data SNR yang Terbaca ke Chirpstack

Jarak	100m	200m	300m	400m
RSSI	-3,5	-11,2	3	6,5
	0,2	-6	6,8	5,2
	-4,8	-5	-4	1,8
	-2,2	-2,2	5,2	2,2
	-6	-1,5	7	6,8
	-0,2	-6,5	7	8
	1,2	0,5	5,2	7
	-6,8	-2,2	2,2	5,2
	-6,2	-12	-	-
	2,5	-4	-	-
rata - rata	-2,58	-5,01	4,05	5,3375



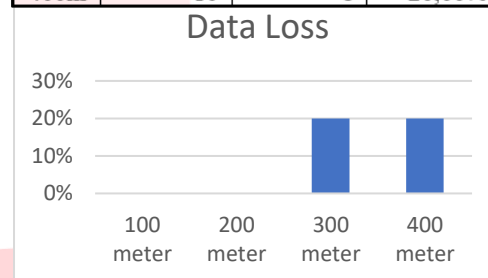
GAMBAR 29
Chart Rata-Rata SNR

SNR (*Signal to Noise Ratio*) adalah sebuah rasio antara sinyal dan *noise* pada sebuah sinyal. Semakin positif nilai SNR, rasio sinyal semakin besar, sebaliknya jika semakin negatif nilai SNR maka semakin besar rasio *noise*-nya. Dari gambar 29 bisa disimpulkan jika terdapat banyak *noise* pada jarak 100 meter dan 200 meter, dan lebih banyak sinyal pada jarak 300 meter dan 400 meter. Ini memperkuat dugaan diatas mengapa nilai RSSI pada jarak tersebut mengalami kenaikan dibanding jarak setelahnya.

3. Pengukuran Data Loss

TABEL 5
Hasil Data Loss

jarak	data terkirim	data diterima	data loss
100m	10	10	0,00%
200m	10	10	0,00%
300m	10	8	20,00%
400m	10	8	20,00%



GAMBA 30
Chart Data Loss

Data loss adalah peristiwa ketika sebuah data berhasil ditransmisikan (*end-node*) namun data tidak berhasil diterima pada tempat tujuan data tersebut (*server* LoRaWAN). Pada gambar 30 terlihat dimana *data loss* mulai terjadi pada jarak 300 meter. ini terjadi karena semakin jauh *end-node* mentransmisikan data semakin sedikit data yang berhasil masuk pada *web-server* LoRaWAN dikarenakan terdapat beberapa *packet* data yang kehilangan informasinya selama ditransmisikan di udara sehingga ketika sampai di server LoRaWAN data yang diterima tidak lengkap dan diabaikan.

V. KESIMPULAN

Sistem LoRaWAN yang berusaha dibangun oleh penulis sudah dapat berjalan dengan baik. *End-node* sudah dapat melakukan *uplink* untuk mengirimkan data data ke ChirpStack, begitu juga ChirpStack sudah bisa menyajikan data dan informasi tentang *gateway* dan *end-node* yang terdaftar.

Peletakan *gateway* pada ruang tertutup sangat tidak dianjurkan dikarenakan pengiriman data akan sangat terganggu dikarenakan data terhalang oleh tembok atau dinding. Disarankan peletakan *gateway* harus di tempat terbuka dengan tiang pemancar yang diletakan tinggi agar dapat menjangkau area yang lebih luas.

Berdasarkan hasil pengujian pengiriman data ke chirpstack, sistem LoRaWAN yang dibangun sangat tidak cocok untuk pengiriman data jarak jauh pada daerah perkotaan dikarenakan banyaknya *noise* di daerah tersebut, namun masih memungkinkan untuk daerah perumahan.

REFERENSI

- [1] “Perusahaan Sistem Manajemen Telematik Armada | TransTRACK.” Accessed: Aug. 06, 2024. [Online]. Available: <https://www.transtrack.co/id/tentang-kami>
- [2] I. Intyas and T. H. Rini, “Perancangan Alat Untuk Tracking Ekspedisi Berbasis GPS (Google Maps) Via SMS,” *Journal ICT*, vol. 9, no. 17, 2018.
- [3] “What are LoRa and LoRaWAN? | The Things Network.” Accessed: Aug. 05, 2024. [Online]. Available: <https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/>
- [4] A. Parashar and F. Deebea, “Licensed Under Creative Commons Attribution CC BY GPS Tracking System Using LoRaWan,” *International Journal of Science and Research*, doi: 10.21275/SR201126115727.
- [5] “Cakupan 3G / 4G / 5G dalam Indonesia - nPerf.com.” Accessed: Aug. 07, 2024. [Online]. Available: <https://www.nperf.com/id/map/ID/-/5119.Telkomsel/signal?ll=-4.696879026871413&lg=116.37763571621278&zom=6>
- [6] H. M. Rakha, J. D. Setiawan, and P. Paryanto, “PENGUJIAN SISTEM MONITORING KONDISI LINGKUNGAN PERAIRAN DENGAN MEMPERHATIKAN KONSUMSI DAYA DAN JARAK PENGIRIMAN DATA DENGAN MENGGUNAKAN WAHANA BUOY,” *JURNAL TEKNIK MESIN*, vol. 10, no. 1, pp. 103–116, 2022.
- [7] F. Muhammad, A. Bhawiyuga, and D. P. Kartikasari, “Analisis Kinerja Protokol Lorawan Untuk Transmisi Data Pada Skenario Urban Area,” *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, vol. 3, no. 9, pp. 9054–9060, 2019.
- [8] P. de Moraes and A. F. da Conceição, “A systematic review of security in the lorawan network protocol,” *arXiv preprint arXiv:2105.00384*, 2021.
- [9] R. Muhendra, N. I. Kreshnaviyanto, and A. Amin, “Jaringan sensor nirkabel: studi dan evaluasi kinerja lora transmitter dan long range radio frekuensi (RF) pada luar ruang,” *Jurnal Jaring SainTek*, vol. 3, no. 1, pp. 6–12, 2021.