

BAB II TINJAUAN UMUM

2.1 *Internet of Things*

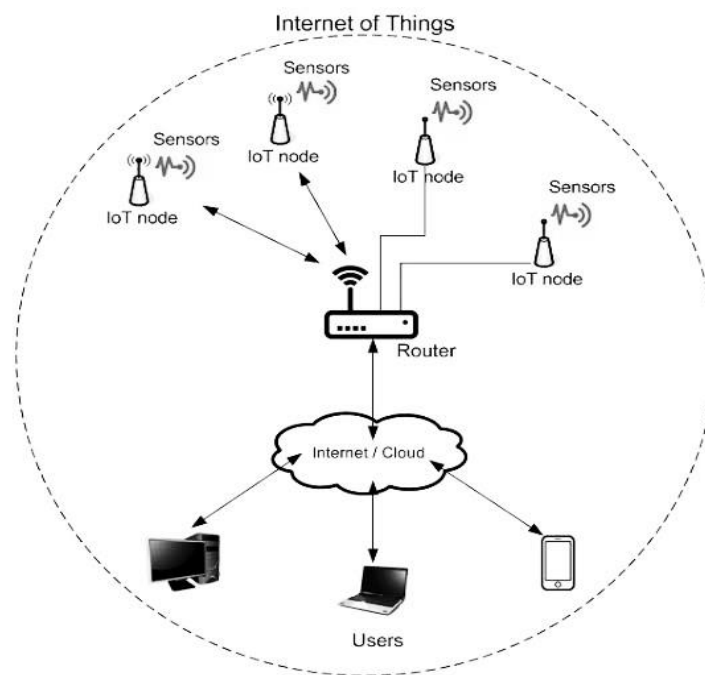
Internet of Things (IoT) adalah sebuah konsep/skenario dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer. Konsep dasar IoT adalah adanya perangkat *node* berupa sistem cerdas tertanam (*embedded*) yang akan mensensor dan mengendalikan berbagai objek fisik di sekitar manusia tersebut. *Node* tersebut terkoneksi satu sama lain melalui jaringan internet sehingga dapat saling berinteraksi dan bekerjasama untuk mencapai tujuan bersama. Sejauh ini, IoT paling erat hubungannya dengan komunikasi machine-to-machine (M2M). Produk dibangun dengan kemampuan komunikasi M2M yang sering disebut dengan sistem cerdas atau "smart".

Istilah IoT (*Internet of Things*) mulai dikenal tahun 1999 yang saat itu disebutkan pertama kalinya dalam sebuah presentasi oleh Kevin Ashton, cofounder and *executive director of the Auto-ID Center* di MIT. Penemuan ini disebut sebagai sebuah lompatan besar dalam *commercializing IoT*[10]. *Internet of things* menjadi sebuah bidang penelitian tersendiri semenjak berkembangnya teknologi internet dan media komunikasi lain semakin berkembang keperluan manusia tentang teknologi, maka semakin banyak penelitian yang akan hadir, *internet of things* salah satu hasil pemikiran para peneliti yang mengoptimasi beberapa alat seperti media sensor, *radio frequency identification (RFID)*, *wireless sensor network* serta *smart object* lain yang memungkinkan manusia mudah berinteraksi dengan semua peralatan yang terhubung dengan jaringan internet[11].

Internet of Things merupakan perkembangan keilmuan yang sangat menjanjikan untuk mengoptimalkan kehidupan berdasarkan sensor cerdas dan peralatan pintar yang bekerjasama melalui jaringan internet[12]. *Internet of Things* adalah revolusi teknologi yang mewakili masa depan komputasi dan komunikasi.

Internet of Things adalah paradigma di mana objek dapat melakukan pengidentifikasian, penginderaan dan kemampuan pemrosesan yang akan memungkinkan mereka untuk berkomunikasi satu sama lain dengan perangkat maupun layanan lain melalui internet untuk mencapai beberapa tujuan[13].

Secara umum, IoT mengacu pada interkoneksi jaringan benda sehari-hari, yang dilengkapi dengan kecerdasan di mana-mana. IoT akan meningkatkan keberadaan internet dengan mengintegrasikan setiap objek untuk berinteraksi melalui sistem tertanam. Berkat kemajuan dalam teknologi yang mendasari, IoT membuka peluang luar biasa untuk sejumlah besar aplikasi baru yang menjanjikan untuk meningkatkan kualitas hidup kita. Dalam beberapa tahun terakhir, IoT telah mendapatkan banyak perhatian dari para peneliti dan praktisi dari seluruh dunia[14].



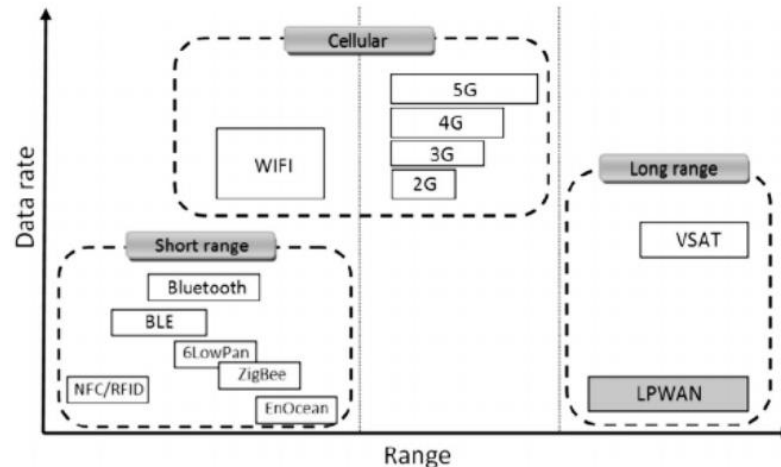
Gambar 2.1 Konsep Komunikasi IoT[14]

IoT adalah jaringan perangkat yang berkomunikasi sendiri menggunakan konektivitas IP tanpa campur tangan manusia. Ekosistem *Internet of Things* (IoT) terdiri atas objek pintar, perangkat cerdas, Smartphone & tablet dll. IoT menggunakan *Radio-frequency identification (RFID)*, *Quick Response (QR) codes*,

sensor atau teknologi nirkabel yang memungkinkan terjadinya komunikasi antar perangkat[15].

2.2 Low Power Wide Area Network (LPWAN)

LPWAN merupakan teknologi komunikasi nirkabel yang memiliki *bit rate* rendah untuk jarak yang jauh dan biaya rendah untuk karakteristik konsumsi daya yang lebih baik[5]. LPWAN semakin mendapatkan popularitas di industri dan komunitas penelitian karena daya yang rendah, jarak jauh, dan karakteristik komunikasi murah. LPWAN menyediakan komunikasi jarak jauh hingga 10–40 km di zona pedesaan dan 1–5 km di zona perkotaan [16]. Sesuai dengan karakteristiknya, tujuan dari teknologi ini adalah jarak jauh, memperpanjang umur baterai dan biaya yang rendah dengan banyak kasus penggunaan yang hanya membutuhkan kecepatan data rendah. Kecepatan transmisi data yaitu 0.3 kbps hingga 5.5 kbps[17]



Gambar 2.2 Perbandingan Data Rate dan Range pada Teknologi Komunikasi Radio[18]

2.3 LoRa

LoRa, akronim dari *long range*, merupakan produk modul teknologi konektivitas nirkabel yang utamanya ditujukan untuk sistem IoT. Modul LoRa dipatenkan dan diproduksi tunggal oleh Semtech Corporation. Meskipun secara

produksi tertutup, namun pengembangan dan implementasinya berstandar terbuka dan standarisasi protokol-protokolnya dikeluarkan oleh asosiasi LoRa Alliance[8].

LoRa merupakan teknologi *transceiver* jaringan nirkabel yang dirancang khusus untuk komunikasi jarak jauh dengan konsumsi daya yang rendah. LoRa merupakan teknologi nirkabel berdaya rendah dengan rentang komunikasi jauh. LoRa menggunakan modulasi *Chirp-spread-spectrum* (CSS) untuk mempertahankan karakteristik daya rendah untuk kepentingan meningkatkan jangkauan komunikasi. CSS telah digunakan dalam komunikasi jarak jauh oleh militer dan badan antariksa karena kemampuannya untuk menahan gangguan[19]. LoRa memiliki keuntungan yaitu jangkauan komunikasi yang jauh seperti seluler namun berdaya rendah seperti Bluetooth, sehingga penggunaannya sangat cocok untuk perangkat sensor yang dioperasikan tahunan dengan sumber daya baterai dan pada cakupan area yang luas[17].

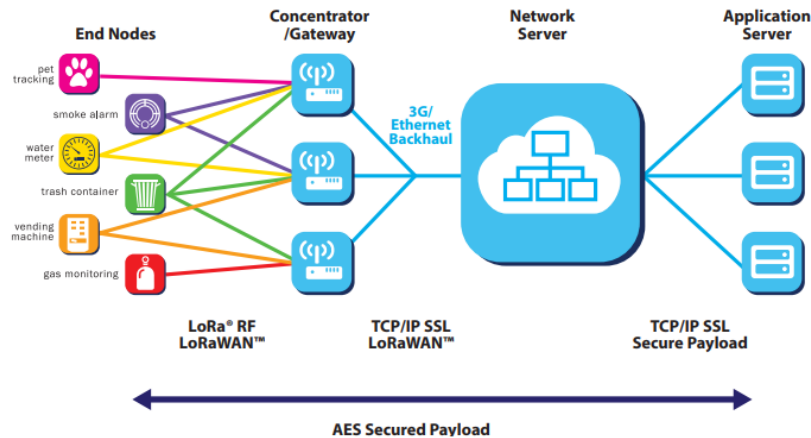


Gambar 2.3 Logo LoRa [20]

LoRa beroperasi pada pita frekuensi *Industrial, Scientific, and Medical* (ISM) tidak berlisensi seperti pada frekuensi 2,4 GHz, 868MHz, 915MHz, 433MHz tergantung pada regulasi masing - masing wilayah[21].

2.3.1 LoRaWAN

LoRaWAN mendefinisikan protokol komunikasi dan arsitektur sistem untuk jaringan sementara lapisan fisik LoRa yang memungkinkan jangkauan komunikasi jarak jauh. Protokol dan arsitektur jaringan memiliki pengaruh paling besar dalam menentukan masa pakai baterai dari sebuah node, kapasitas jaringan, kualitas layanan, keamanan, dan beragam aplikasi yang dilayani oleh jaringan[20].



Gambar 2.4 Arsitektur LoRaWAN [20]

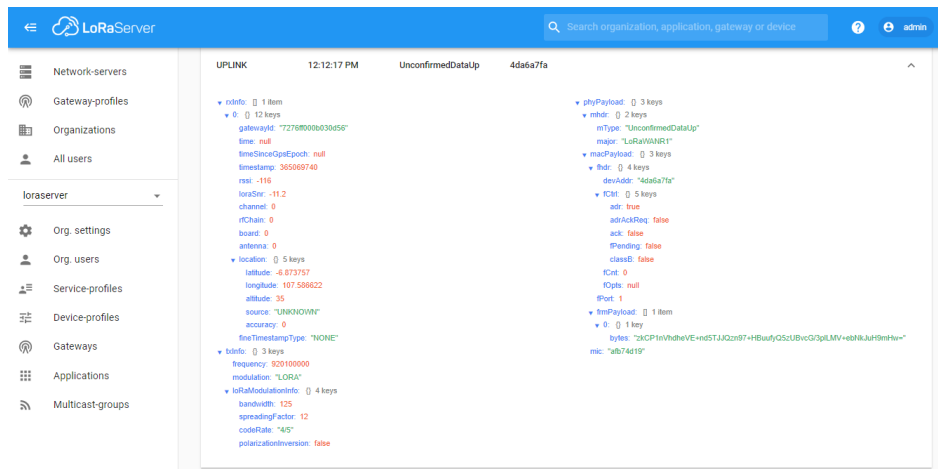
Jaringan LoRa bertopologi star dengan *gateway* sebagai jembatan penghubung. *Gateway* dihubungkan dengan banyak *node* disatu sisi dan network server disisi lainnya. Jaringan LoRa secara garis besar terdiri dari empat bagian: perangkat, *gateway*, *network server* dan aplikasi pengguna[8].

Entitas yang didefinisikan dalam LoRaWAN[22]:

1. *End-device* (atau *end-node*), yaitu node yang melakukan uplink (UL) atau menerima lalu lintas downlink (DL).
2. *gateway*, yaitu perangkat yang mengumpulkan paket yang datang dari *end-device* dan dapat meneruskannya. Gateway juga dapat mengirimkan paket ke *end-device*.
3. Network Server (NS), yang memainkan peran sentral koordinator dan pengontrol jaringan LoRa.

2.3.2 LoRa Server

Pada Gambar 2.6 LoRa server adalah *open-source* LoRaWAN *network server*. Berisi *frame uplink* dan *downlink* transmisi data. Dalam frame pengiriman berisi informasi mengenai kualitas pengiriman data dan detail lokasi serta waktu transmisi data[23].



Gambar 2.5 LoRa Server

2.4 Received Signal Strength Indicator

Received Signal Strength Indicator (RSSI) merupakan parameter yang menunjukkan daya terima dari seluruh sinyal pada band frekuensi saluran yang digunakan. Nilai RSSI semakin dekat dengan 0, maka semakin baik sinyal tersebut. Nilai RSSI dinyatakan dalam dBm dan merupakan nilai negatif, nilai minimum RSSI adalah -120 dBm[23].



Gambar 2.6 Kategori RSSI [17]

Untuk menentukan *Received Signal Strength Indicator* (RSSI) digunakan *log-normal shadowing model* dengan *path loss* referensi yang digunakan dalam keadaan *free space*. Nilai RSSI hasil perhitungan teoritis akan dibandingkan dengan RSSI hasil pengujian di lapangan.

2.4.1 Free Space Path Loss

Free space path loss model adalah redaman ruang bebas yang dimana jalur dalam keadaan *line of sight* (LOS) sehingga tidak terdapat hambatan antara transmitter dan receiver. *Pathloss* dari model *free space* merupakan kasus transmisi ideal. Beberapa faktor yang mempengaruhi propagasi sinyal, seperti multipath,

rintangan dan sebagainya, diabaikan dalam model ini[24]. Persamaan *free space path loss* dapat dituliskan sebagai berikut:

$$PL_{fs} = 3,44 + 10n \log f(\text{MHz}) + 10n \log d_0(\text{Km}) \quad (1)$$

Dimana :

PL_{fs} : Rugi-rugi propagasi di udara dalam *free space* (dB)

f : Frekuensi (MHz)

d : Jarak antara Tx dan Rx (km)

n : *path loss exponent* (untuk free space, n = 2)

2.4.2 Path Loss Exponent

Path loss secara umum didefinisikan sebagai penurunan kuat medan secara menyeluruh sesuai bertambah jauhnya jarak antara pemancar dan penerima. Pengaruhnya sangat kuat, sehingga menimbulkan penurunan level daya pada sinyal yang diterima[25].

Path loss rata-rata untuk skala besar ($\overline{PL}(d)$) dapat dihitung menggunakan fungsi jarak dengan menggunakan *path loss exponent* berikut:

$$PL(d) \propto \left(\frac{d}{d_0}\right)^n \quad (2)$$

$$\overline{PL}(d)[\text{dB}] = PL(d_0) + 10n \log\left(\frac{d}{d_0}\right) \quad (3)$$

Dimana:

$\overline{PL}(d)$: *Path loss*

$PL(d_0)$: *Free space path loss* (*path loss* referensi)

d : Jarak antara Tx dan Rx

d_0 : Jarak referensi

n : *path loss exponent*

Parameter n yang sangat berpengaruh dalam menentukan batas kritis dari cakupan wilayah dan kapasitas sistem. n adalah *path loss exponent*, yang disesuaikan dengan kondisi lingkungan pengujian.

Tabel 2.1 *Path loss Exponents* [28]

Lingkungan	Path Loss Exponent, n
Free space	2
Urban area cellular radio	2,7 – 3,55
Shadowed urban cellular radio	3 – 5
In building LOS	1,6 – 1,8
Obstructed in building	4 – 6
Obstructed in factories	2 – 3

2.4.3 *Log-Normal Shadowing Model*

Log-normal shadowing model merupakan salah satu metode untuk memprediksi kekuatan sinyal yang diterima dengan memperhitungkan *path loss*. Model ini sangat cocok untuk lingkungan indoor dan outdoor. Model ini menyediakan sejumlah parameter yang dapat dikonfigurasi sesuai dengan lingkungan yang berbeda[27]. *log-normal shadowing model* berguna ketika berhadapan dengan situasi yang lebih realistis, dimana *efek shadowing* yang kemungkinan besar terjadi dilapangan juga diperhitungkan[28]. Karena itu PL(d) dihitung dengan *log-normal distribution* berikut:

$$PL(d)[dB] = \overline{PL}(d) + X\sigma \quad (4)$$

$$PL(d)[dB] = PL(d_0) + 10n \log\left(\frac{d}{d_0}\right) + X\sigma \quad (5)$$

$$Pr(d) = Pt(d) - PL(d) \quad (6)$$

Berdasarkan persamaan di atas, perhitungan *path loss* pada jarak d ditambahkan dengan efek *shadowing* $X\sigma$. $X\sigma$ adalah variabel acak yang terdistribusi gaussian dengan zero mean dan standar deviasi σ (dalam dB). Nilai n dan standar deviasi σ akan disesuaikan kondisi lingkungan pengujian. Disini juga memperhitungkan *gain* antena pengirim serta penerima. Sehingga, RSSI:

$$RSSI = Pt(d) + Gt - (PL(d_0) + 10n \log\left(\frac{d}{d_0}\right) + X\sigma) + Gr \quad (7)$$

Dimana:

$PL (d_0)$: <i>Free space path loss (path loss referensi)</i>
d	: Jarak antara Tx dan Rx
d_0	: Jarak referensi
n	: <i>path loss exponent</i>
G_t	: <i>Gain</i> antena pengirim
G_r	: <i>Gain</i> antena penerima

2.5 *Signal to Noise Ratio*

Signal to Noise Ratio (SNR) didefinisikan sebagai rasio antara daya sinyal yang diinginkan dengan daya derau (noise), dan secara luas digunakan sebagai standar ukuran kualitas sinyal untuk sistem komunikasi. Suatu sinyal informasi sebagai media komunikasi akan mengalami banyak gangguan oleh derau (noise), sehingga dapat merusak sinyal informasi itu. Sinyal yang mengalami gangguan ini mengalami penurunan kualitasnya. Kualitas sinyal ini dapat ditentukan nilai dari nilai Signal to Noise Ratio (SNR) yang diukur dalam satuan desibel (dB)[29].

2.6 *Packet Loss*

Packet Loss merupakan banyaknya paket yang gagal mencapai tempat tujuan saat pengiriman paket. Jika paket gagal dikirim maka paket tersebut tidak akan dikirim kembali, atau dengan kata lain paket tersebut hilang[23].

Rumus untuk menentukan *Packet Loss*:

$$Packet\ Loss = \frac{Paket\ yang\ dikirim - paket\ yang\ diterima}{paket\ yang\ dikirim} \times 100\% \quad (8)$$

2.7 *Delay*

Delay merupakan banyaknya waktu yang diperlukan sebuah paket untuk melakukan perjalanan dari sumber ke tujuan. Bersama dengan *delay*, mendefinisikan kecepatan dan kapasitas dalam jaringan[23].

Rumus untuk menentukan *Delay*:

$$Delay = Waktu\ paket\ diterima - waktu\ paket\ dikirim \quad (9)$$

Tabel 2.2 Standarisasi Delay oleh THIPON

Kategori Delay	Besarnya Delay	Indeks
Sangat Bagus	< 150 ms	4
Bagus	150 s/d 300 ms	3
Sedang	300 s/d 450 ms	2
Jelek	> 450 ms	1

2.8 ESP-32 Espressif

Pada akhir 2014, komunitas pembuat menemukan modul WiFi kecil yang bagus yang disebut ESP8266. Pengembang di mana-mana bersukacita pada perangkat ajaib ini yang menempatkan "*Internet*" di *Internet of Things*. ESP32 adalah penerus ESP8266 yang sangat dinanti[30].

ESP32 merupakan mikrokontroler yang sudah dilengkapi modul Wi-Fi dan *Bluetooth* yang berfungsi sebagai perangkat tambahan. Modul ini juga dilengkapi dengan prosesor, memori dan GPIO (*General Purpose Input Output*) dimana jumlah pin bergantung dengan jenis ESP32 yang digunakan sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun lagi karena sudah memiliki perlengkapan layaknya mikrokontroler. Selain itu, ada beberapa *Firmware Software Development Kit* (SDK) yang digunakan oleh perangkat ini berdasarkan *open source* yang diantaranya adalah sebagai berikut :

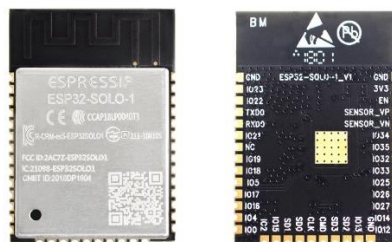
1. NodeMCU dengan menggunakan *basic programming* luar
2. MicroPhyton dengan menggunakan *basic programming phyton*
3. AT Command dengan menggunakan *basic programming AT Command*

ESP32 juga memiliki kapasitas memori lebih besar jika dibandingkan dengan ESP8266, adapun spesifikasi ESP32 adalah sebagai berikut :

1. 448 KB ROM untuk *booting* dan *core function*
2. 520 KB SRAM untuk data dan instruksi
3. 8 KB SRAM pada RTC yang dapat digunakan ketika dalam keadaan *deepsleep*
4. 1 Kbit *Efuse*

5. Mendukung *multiple external* QSPI (*Quad Serial Peripheral Interface*) flash dan SRAM chip

Untuk pemrogramannya sendiri dapat menggunakan ESPlorer untuk *Firmware* berbasis NodeMCU dan menggunakan *putty* sebagai terminal kontrol untuk AT Command. Selain itu, kita bisa memprogram perangkat ini menggunakan Arduino IDE dan kita dapat dengan mudah memprogram dengan dasar-dasar program arduino. Bentuk fisik dari ESP32 bisa dilihat pada gambar 2.5



Gambar 2.7 ESP32 Espressif [30]

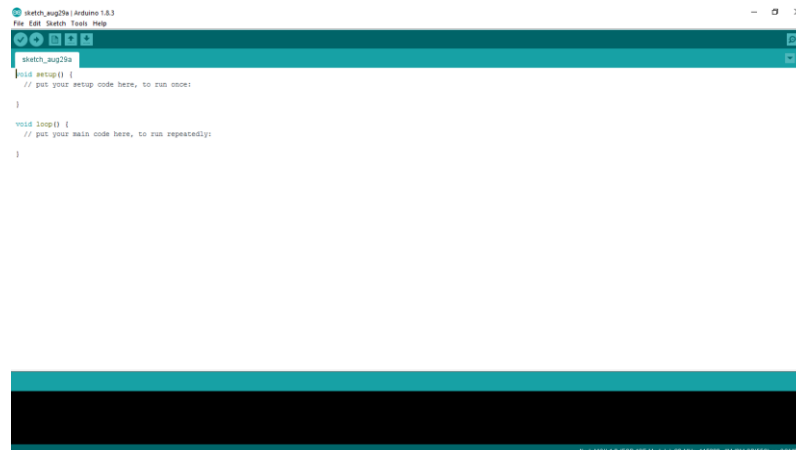
2.9 Arduino IDE

Modul Arduino merupakan open-source platform prototyping[31]. Arduino merupakan sistem mikrokontroler yang relatif mudah dan cepat dalam membuat aplikasi elektronika maupun robotika[32].

Pendekatan pemrograman yang paling umum adalah dengan menggunakan Arduino IDE, yang memanfaatkan bahasa pemrograman C. Arduino IDE adalah *software* yang ditulis menggunakan java[31]. Arduino IDE terdiri dari:

1. *Editor program*, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.
2. *Verify/Compiler*, sebuah modul yang mengubah kode program (bahasa *processing*) menjadi kode biner.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memori mikrokontroler di dalam papan arduino.

Berikut ini adalah tampilan software Arduno IDE:



Gambar 2.8 Tampilan Arduino IDE

2.10 Sensor

Dalam kaitannya dengan sistem elektronis, Sensor pada dasarnya dapat dipandang sebagai sebuah perangkat atau device yang berfungsi mengubah suatu besaran fisik menjadi besaran listrik, sehingga keluarannya dapat diolah dengan rangkaian listrik atau sistem digital. Dewasa ini, hampir seluruh peralatan modern memiliki sensor di dalamnya. Terkait dengan perkembangan teknologi yang begitu luar biasa, pada saat ini, banyak sensor telah dipabrikasi dengan ukuran sangat kecil hingga orde nanometer sehingga menjadikan sensor sangat mudah digunakan dan dihemat energinya[33]

2.10.1 Sensor MQ-7

Karbon Monoksida (CO) adalah gas yang tak berwarna, tak berbau, dan tak berasa. Ia terdiri dari satu atom karbon yang secara kovalen berikatan dengan satu atom oksigen. Dalam ikatan ini, terdapat dua ikatan kovalen dan satu ikatan kovalen koordinasi antara atom karbon dan oksigen. Gas CO sebagian besar berasal dari sisa pembakaran dan emisi dan kendaraan bermotor. Gas CO sangat berbahaya apabila terhirup oleh manusia, terutama masalah pernapasan bahkan dapat menyebabkan kematian. Sensor CO yang digunakan pada sistem ini adalah sensor MQ-7.

2.10.2 Sensor Particulate Matter (PM)

Particulate matter (PM) adalah istilah untuk partikel padat atau cair yang ditemukan di udara [3]. Partikel dengan ukuran besar atau cukup gelap dapat dilihat sebagai jelaga atau asap. Sedangkan partikel yang sangat kecil dapat dilihat dengan mikroskop electron. Partikel berasal dari berbagai sumber baik mobile dan stasioner (diesel truk, woodstoves, pembangkit listrik, dan lain lain). Sensor PM yang digunakan pada sistem ini adalah sensor Sharp GP2Y1010AU0F[34].

2.10.3 Sensor Suhu dan Kelembapan

DHT11 adalah keluaran sinyal digital dari sensor suhu dan kelembapan yang sudah dikalibrasi. Menggunakan sensor teknologi akuisisi digital serta suhu dan kelembapan teknologi sensor. Sensor termasuk kelembapan resistif mengukur komponen dan sensor suhu NTC komponen. Koneksi dengan kinerja tinggi 8 bit Mikrokontroler chip tunggal memberikan sensor kualitas yang sangat baik, respon super cepat, kemampuan anti-jamming, hemat biaya. DHT11 menawarkan port I / O untuk terhubung dengan msp430. Mengirimkan 40 bit data kelembapan dan suhu ke mikrokontroler setiap saat, data diperiksa oleh CRC. Jadi ketelitian data dijamin. Konsumsi daya DHT11 cukup rendah, tegangan catu daya 5 V dan rata-rata maksimum saat ini sekitar 0,5 mA. Pengkoneksian DHT11 sensor suhu dan kelembapan digital sederhana, hanya perlu satu port I / O untuk terhubung ke MCU[35].

2.11 Platform

Dalam ilmu komputer, *platform* merupakan kombinasi antara sebuah arsitektur perangkat keras dengan sebuah kerangka kerja perangkat lunak (kerangka kerja aplikasi). Kombinasi tersebut memungkinkan sebuah perangkat lunak, khusus perangkat lunak aplikasi, dapat berjalan. *Platform* yang umum sudah menyertakan arsitektur, sistem operasi, bahasa pemrograman dan antarmuka yang terkait untuk komputer.

Platform adalah unsur yang penting dalam pengembangan perangkat lunak. *Platform* mungkin dapat didefinisikan secara sederhana sebagai tempat untuk menjalankan perangkat lunak. Penyelenggara *platform* menyediakan

pengembangan perangkat lunak dengan kesepakatan serangkaian kode logika yang akan berjalan secara konsisten sepanjang *platform* ini berjalan di atas *platform* yang lainnya. Kode logika ini mencakup bytecode, kode sumber dan kode mesin. Dengan demikian, pelaksanaan program tidak dibatasi oleh jenis sistem operasi yang tersedia. *Platform* telah menggantikan sebagian besar bahasa mesin independen. Beberapa *platform* perangkat lunak mengemulasikan *platform* perangkat keras secara menyeluruh, seperti pada virtualisasi sistem.

Jadi secara detailnya *Platform* adalah arsitektur hardware/fondasi/standar bagaimana sebuah sistem dimana aplikasi/program dapat berjalan, atau bisa juga dikatakan *Platform* adalah dasar dari teknologi dimana teknologi yang lain atau proses-proses dibuat. Sebuah *Platform* terdiri dari sistem operasi yaitu program sistem koordinasi komputer yang memberikan perintah-perintah kepada prosesor dan hardware untuk melakukan operasi-operasi logis dan mengatur pergerakan data di komputer. Banyak orang beranggapan bahwa *Platform* dan Sistem Operasi adalah sama, namun pada kenyataannya tidak. *Platform* merupakan dasar atau tempat dimana sistem operasi bekerja, tanpa *platform* sistem operasi tidak akan berjalan.

Fungsi utama *platform* adalah untuk menjalankan sebuah perangkat lunak agar dapat digunakan atau berjalan serta dapat memudahkan dalam pengoperasian aplikasi atau *software* dan lain sebagainya yang berhubungan dengan teknologi, memberikan keamanan dan membantu dalam hal perencanaan[36].

Platform Layanan IoT dianggap sebagai landasan solusi IoT yang sukses. *Platform* Layanan IoT bertanggung jawab untuk banyak hal. Fungsi *Platform* Layanan IoT mencakup kemampuan untuk menggunakan, mengkonfigurasi, memecahkan masalah, mengamankan, mengelola, dan memantau perangkat IoT. Dan juga mencakup kemampuan untuk mengelola aplikasi dalam hal instalasi perangkat lunak / *firmware*, penambalan, mulai / berhenti., debugging, dan pemantauan[37].

Jadi fungsi utama *Platform* IoT adalah tempat atau fasilitas yang digunakan untuk menampilkan monitoring dan control dari jarak jauh yang dapat terhubung dengan menggunakan koneksi internet, antara *End device* dan *brainware*.

2.11.1 ANTARES Platform

Platform IoT ANTARES adalah *platform* IoT yang dikembangkan oleh PT. Telekomunikasi Indonesia (Persero) Tbk., yang digarap oleh bidang *Infrastructure Research and Standarization* (IRS) pada Divisi Digital Service (DDS)[40].

ANTARES merupakan singkatan dari *Application and Technology Platform as your Reliable Solution*. Dan merupakan *middleware* yang menghubungkan *developer* dengan *end user*. *End User* yang berlangganan ke *developer* tidak perlu berhubungan langsung dengan ANTARES. *Developer* dapat melakukan *subscribe* data yang tersimpan pada ANTARES ke *website/aplikasi* yang dikembangkan untuk tujuan komersil.



Gambar 2.9 Tampilan Depan Antares.id [38]

Platform ANTARES sendiri merupakan generic *platform* yang dapat menyimpan berbagai tipe data. UI yang simple dan user friendly menjadikan keunggulan dari *platform* ini pula.

2.12 Tinjauan Mutakhir

Penelitian ini mengacu terhadap beberapa referensi yang terkait dengan penelitian sebelumnya, dimana masing-masing peneliti menggunakan metode yang berbeda sesuai dengan permasalahan yang dibahas. Berikut ini beberapa referensi yang digunakan untuk membedakan pembahasan yang dibahas pada penelitian yang telah ada.

Tabel 2.3 Tinjauan Mutakhir

Penulis (Tahun)	Judul	Metode/Parameter	Fitur	Kelebihan	Kekurangan
Eko Murdyantoro, et. al (2009)	Studi Performansi Jarak Jangkauan LoRa Olg01 Sebagai Infrastruktur Konektivitas Nirkabel Iot	RSSI (<i>Received Signal Strength Indicator</i>)	RSSI dan jangkauan LoRa.	- Jarak jangkauan LoRa yang diperoleh terpapar dengan jelas dan mudah dipahami. - RSSI yang diperoleh cukup besar.	- Jarak jangkauan hanya hingga 400 m.
Pinky Devi Dama Istianti, et. al (2019)	Analisis Performansi Teknologi Akses LPWAN LoRa Antares Untuk Komunikasi Data End Node	<i>Received Signal Strength Indicator</i> (RSSI), <i>Signal to Noise Ratio</i> (SNR), dan <i>Success Rate</i>	Teori, Penjelasan, Perbandingan.	- Hasil pengukuran setiap parameter ditunjukkan dengan jelas.	- Tidak ada perbandingan hasil pengukuran dengan hasil perhitungan secara teoritis.

Z A Tan, et. al. (2019)	<i>Analysis on LoRa RSSI in Urban, Suburban, and Rural Area for Handover Signal Strength-Based Algorithm.</i>	RSSI (<i>Received Signal Strength Indicator</i>)	Parameter	- Jangkauannya sangat luas yaitu hingga 4,6 km untuk area urban dengan RSSI kurang lebih berkisar pada -90 dBm hingga -140dBm	- Metode pengambilan data tidak terlalu jelas.
Pascal Jörke, et. Al. (2017)	<i>Urban Channel Models for Smart City IoT-Networks Based on Empirical Measurements of LoRa-links at 433 and 868 MHz</i>	<i>Packet Delivery Rates</i> , RSSI dan SNR	Parameter	- Hasil Pengujian ditunjukkan dengan jelas. - Pada 869,5 MHz RSSI berkisar -101 dBm hingga -132 dBm. SNR sebesar 6 dB hingga -21 dB -	- Tidak ada pembanding dengan perhitungan teoritis

Mauricio A. Sanchez, et. al (2015)	Peningkatan Akurasi Estimasi Jarak Rssi Dengan Model <i>Log Normal</i> Menggunakan Metode <i>Kalman Filter</i> Pada Bluetooth Low Energy	Perhitungan RSSI dengan <i>Log-normal shadowing</i>	Metode perhitungan	- Cara perhitunga dijelaskan secara jelas sehingga mudah dipahami.	- Penentuan nilai standar deviasi tidak dicantumkan
------------------------------------	--	---	--------------------	--	---