

**PREDIKSI MAINTENANCE PADA SOLAR CELL MENGGUNAKAN
AMAZON WEB SERVICE**



TUGAS AKHIR

*Disusun dalam rangka memenuhi salah satu persyaratan
Untuk menyelesaikan program Strata-1 Departemen Teknik Informatika
Fakultas Teknik Universitas Hasanuddin
Makassar*

Disusun Oleh :

FATHUR RIZQI

D421 15 312

**DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
MAKASSAR**

2022

LEMBAR PENGESAHAN SKRIPSI
PREDIKSI MAINTENANCE PADA SOLAR CELL MENGGUNAKAN
AMAZON WEB SERVICE

Disusun dan diajukan oleh

FATHUR RIZQI
D42115312

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas Teknik Universitas Hasanuddin pada tanggal 25 Agustus 2022 dan dinyatakan telah memenuhi syarat kelulusan.

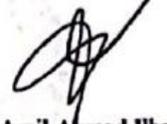
Menyetujui,

Pembimbing Utama,



Dr. Adnan, S.T., MT.
Nip. 197404262003121002

Pembimbing Pendamping,



Dr. Amil Ahmad Ilham, S.T., M.IT.
Nip. 197310101998021001



Dr. Amil Ahmad Ilham, S.T., M.IT.
Nip. 197310101998021002

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : FATHUR RIZQI
NIM : D421 15 312
Program Studi : TEKNIK INFORMATIKA
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

PREDIKSI MAINTENANCE PADA SOLAR CELL MENGGUNAKAN AMAZON WEB SERVICE

Adalah karya tulis saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 04 Agustus 2022

Yang Menyatakan


FATHUR RIZQI

ABSTRAK

Pentingnya pemeliharaan terhadap suatu benda maupun mesin menjadi hal yang sangat fundamental untuk keberlangsungan hidup sebuah benda ataupun mesin. Dengan melihat permasalahan diatas, maka di dalam penelitian ini, kami akan melakukan sebuah upaya untuk pendeteksian *maintenance* lebih dini terhadap kinerja Panel Surya dengan metode peramalan (*forecasting*) dari data yang dikumpulkan dalam rentang waktu tertentu dengan perangkat *Internet of Things* berupa sensor-sensor seperti sensor Arus & Tegangan, Sensor Debu, Sensor Cahaya dan sensor Suhu & Kelembaban yang terhubung dengan mikrokontroller ESP 32 hingga membentuk dataset yang di organisir di dalam konsol Amazon Web Service.

Penelitian ini bertujuan untuk mengetahui lebih dini kemungkinan terjadinya kerusakan pada suatu peralatan. Dengan mengetahui kondisi peralatan tersebut, maka dapat dilakukan tindakan untuk mencegah peralatan tersebut mengalami kerusakan.

Kata kunci: *Internet of Things*, Panel Surya, Prediksi, *Maintenance*, *Amazon Web Service*

ABSTRACT

The importance of preservation of an object as well as a machine becoming very fundamental thing for the viability of an object as a machine. Accordingly, through this study, we will find a way to detect ahead of time the performance of solar panel with the forecasting method of a collected data within certain time span with IOT devices in the form of sensors such as voltage and current sensor, dust sensor, light sensor, and temperature and humidity sensor which were connected with ESP 32 microcontroller to form an organized dataset in the Amazon Web Service console. this study aims to determine early on the damage possibility of an equipment. by knowing the condition of the equipment, actions can be taken to prevent the equipment from being damaged.

Keyword: *Internet of Things, Solar Cell, Prediction, Maintenance, Amazon Web Service*

KATA PENGANTAR

Bismillahirrahmanirrahim

Assalamualaikum Warahmatullahi Wabarakatuh

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya sehingga tugas akhir yang berjudul “PREDIKSI MAINTENANCE PADA PANEL SURYA MENGGUNAKAN AMAZON WEB SERVICE” ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari bahwa dalam penyusunan dan penulisan laporan tugas akhir ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan tugas akhir. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada:

1. Kedua Orang tua penulis, Alm. Dr. Muh. Nadjib, M.Ed., M.Lib dan Ibu Harnawati Abdullah yang selalu memberikan dukungan, doa, dan semangat serta selalu sabar dalam mendidik penulis sejak kecil;
2. Bapak Adnan, ST., M.T., Ph.D, selaku pembimbing I dan Bapak Dr. Amil Ahmad Ilham, ST., M.IT., selaku pembimbing II yang selalu menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir;
3. Bapak Dr. Amil Ahmad Ilham, ST., M.IT., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin atas bimbingannya selama masa perkuliahan penulis;

4. Teman-teman Hyperv15or atas dukungan dan semangat yang diberikan selama ini;
5. Segenap Staf dan Dosen Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu penulis.
6. Orang-orang berpengaruh lainnya yang tanpa sadar telah menjadi inspirasi bagi penulis.

Akhir kata, penulis berharap semoga Allah SWT. berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga Tugas Akhir ini dapat memberikan manfaat bagi pengembangan ilmu. Aamiin.

Wassalam

Makassar, 4 Agustus 2022

DAFTAR ISI

KATA PENGANTAR	ii
ABSTRAK	Error! Bookmark not defined.
DAFTAR ISI.....	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	ix
BAB I PENDAHULUAN.....	1
1.1 LATAR BELAKANG	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Manfaat Penelitian	3
1.5 Batasan Masalah Penelitian.....	3
1.6 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA.....	5
2.1 Internet of Things	5
2.2 Predictive Maitenance.....	7
2.3 Mikrokontroler	9

2.3.1 Esp 32.....	10
2.4 Sensor-sensor	12
2.4.1 Sensor suhu dan kelembaban (DHT11)	13
2.4.2 Sensor cahaya (BH1750).....	14
2.4.3 Sensor debu (GP2Y1010AU0F Optical Dust Sensor)	15
2.4.4 Sensor INA 219 dc current monitor	16
2.5 Amazon Web Service.....	17
2.5.1 IoT Core	17
2.5.2 IoT Analytics.....	19
2.6 Arima	24
2.6.1 Klasifikasi Model ARIMA.....	25
BAB III METODOLOGI PENELITIAN.....	33
3.1 Tahapan Penelitian	33
3.2 Waktu dan Lokasi Penelitian	34
3.3 Instrumen Penelitian.....	34
3.4 Pengambilan data	35
3.5 Perancangan sistem	35
3.5.1 Perancangan Panel Surya	36

3.5.2 Perancangan Perangkat Keras (Hardware).....	38
3.5.3 Penulisan program.....	40
3.5.4 Penggunaan tools Amazon Web Service (AWS).....	42
3.6 Pengujian Sistem.....	45
BAB IV Hasil & Pembahasan.....	46
4.1 Hasil Penelitian.....	46
4.2 Pembahasan.....	53
BAB V PENUTUP.....	60
5.1 Kesimpulan	60
5.2 Saran.....	60
DAFTAR PUSTAKA	61
LAMPIRAN.....	62

DAFTAR GAMBAR

Gambar 2. 1 ESP 32	12
Gambar 2. 2 DHT 11	13
Gambar 2. 3 Sensor Cahaya BH1750	14
Gambar 2. 4 Sensor Debu	15
Gambar 2. 5 Sensor Arus & Tegangan	16
Gambar 2. 6 IoT Core	18
Gambar 2. 7 IoT Analytics.....	20
Gambar 3. 1 Rangka & Panel Surya	36
Gambar 3. 2 Skema Charge Controller	37
Gambar 3. 3 Rangkaian Mikrokontroler & Sensor	38
Gambar 3. 4 Rules AWS.....	43
Gambar 3. 5 Dataset AWS	43
Gambar 3. 6 export dataset ke Jupyter Notebook	44
Gambar 4. 1 Grafik Awal Debu	46
Gambar 4. 2 Grafik Awal Tegangan.....	47
Gambar 4. 3 Grafik Awal Arus	47
Gambar 4. 4 Gambar Dataset	48

Gambar 4. 5 Decompose Data Tegangan.....	54
Gambar 4. 6 Shifting Data Tegangan.....	55
Gambar 4. 7 Forecast Tegangan.....	55
Gambar 4. 8 Decompose Data Debu.....	56
Gambar 4. 9 Shifting Data Debu.....	57
Gambar 4. 10 Forecast Debu.....	57
Gambar 4. 11 Decompose Arus	58
Gambar 4. 12 Shifting Arus	59
Gambar 4. 13 Forecast Arus.....	59

DAFTAR TABEL

Tabel 4. 1 Prediksi Tegangan.....	49
Tabel 4. 2 Prediksi Arus.....	50
Tabel 4. 3 Prediksi Debu.....	51

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Dewasa ini, kita mulai menyadari akan pentingnya pemeliharaan terhadap suatu benda maupun mesin untuk keberlangsungan hidup maupun efisiensi dari sebuah benda, namun terkadang kita tidak mengetahui mengenai gejala yang diperlihatkan oleh benda tersebut, kapan benda tersebut membutuhkan pemeliharaan khusus dsb. Oleh karena itu, hal tersebutlah yang melatarbelakangi untuk melakukan pendeteksian lebih dini mengenai pemeliharaan sebuah mesin yang pada kasus ini ialah Panel Surya agar dapat bekerja lebih efektif.

Panel surya ialah suatu komponen yang dapat digunakan untuk mengubah energy cahaya Matahari menjadi energy Listrik dengan menggunakan prinsip yang disebut efek photovoltaic. Efek photovoltaic itu sendiri adalah suatu fenomena di mana muncul tegangan listrik karena adanya suatu hubungan atau kontak dari dua elektroda, dimana keduanya dihubungkan dengan sistem padatan atau cairan saat mendapatkan energi cahaya. Oleh karena itulah, solar cell sering disebut juga dengan sel *photovoltaic* (PV). Efek photovoltaic ini ditemukan pertama kali oleh Henri Becquerel (wikipedia.org) pada tahun 1839. Sehingga kita perlu

memperhatikan beberapa factor dalam menjaga kinerja Panel Surya agar tetap berjalan dengan baik.

Metode yang digunakan kali ini ialah *Predictive Maintenance* yang merupakan cara terbaik dalam melakukan perawatan, dengan adanya **Predictive Maintenance** bertujuan untuk memprediksi waktu untuk melakukan Maintenance (Perbaikan) pada Panel Surya. Sehingga, kinerja mesin anda akan tetap berjalan dengan lancar dan efisien. Cara kerja metode ini diawali dengan (1) Mengumpulkan data Panel Surya menggunakan beberapa Sensor dan di Integrasikan dengan microcontroller ESP 32, (2) Menganalisis data – data Panel Surya yang diterima oleh Sensor, kemudian (3) Menghasilkan hasil Prediksi untuk melakukan Maintenance pada Panel Surya (solar cell).

Diharapkan dari penelitian ini dapat memberi Manfaat praktis agar bisa melakukan prediksi apakah panel surya masih berfungsi secara optimal, dan kapan panel surya tersebut memerlukan maintenance agar kinerjanya tetap maksimal, dan juga dapat menjadi rujukan dalam Ilmu Teknik Informatika, khususnya dalam mengembangkan Maintenance system pada panel surya dengan memanfaatkan model dan metode yang tepat dan efisien.

1.2 Rumusan Masalah

Kapan waktu yang tepat untuk melakukan maintenance pada Panel Surya?

1.3 Tujuan Penelitian

Untuk mengetahui kapan waktu yang tepat untuk melakukan Maintenance (perbaikan) pada Panel Surya.

1.4 Manfaat Penelitian

Manfaat dari penelitian ini adalah: (1) Manfaat praktis agar bisa melakukan prediksi apakah panel surya masih berfungsi secara optimal, dan kapan panel surya tersebut memerlukan maintenance agar kinerjanya tetap maksimal. (2) Manfaat teoretis, hasil penelitian akan menjadi rujukan dalam Ilmu Teknik Informatika, khususnya dalam mengembangkan maintenance system pada panel surya dengan memanfaatkan model dan metode yang tepat dan efisien.

1.5 Batasan Masalah Penelitian

Pembahasan penelitian ini hanya terbatas dengan menggunakan mikrokontroller dengan jenis ESP 32 dan menggunakan 4 sensor yaitu Sensor Suhu dan Kelembaban, Sensor Arus & Tegangan, Sensor Debu, Sensor Cahaya. Serta menggunakan Amazon Web Service untuk mengolah data mengenai Panel Surya.

1.6 Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi tulisan secara keseluruhan, maka akan diuraikan beberapa tahapan dari penulisan secara sistematis, yaitu :

BAB I PENDAHULUAN

Bab ini menguraikan secara umum mengenai hal yang menyangkut latar belakang, perumusan masalah dan batasan masalah, tujuan, manfaat dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi teori-teori terkait hal-hal yang mendasari dan berhubungan dengan perancangan alat Predictive Maintenance pada Panel Surya (Solar Cell), termasuk Panel Surya, Mikrokontroler ESP 32, Sensor, dan metode-metode yang digunakan dalam penelitian.

BAB III METODOLOGI PENELITIAN

Bab ini berisi tentang perencanaan dan perancangan alat maupun sistem dalam pembuatan alat Predictive Maintenance pada Solar Cell serta metode-metode dalam pengolahan data, mulai dari perancangan sistem hingga analisis data.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang hasil pengolahan dan analisis data serta pembahasan hasil penelitian.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan yang didapatkan berdasarkan hasil penelitian yang telah dilakukan serta saran-saran untuk pengembangan lebih lanjut.

BAB II

TINJAUAN PUSTAKA

2.1 Internet of Things

Internet of Thing (IoT) adalah sebuah konsep dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer. IoT telah berkembang dari konvergensi teknologi nirkabel, *micro-electromechanical systems* (MEMS), dan Internet. Penelitian pada IoT masih dalam tahap perkembangan. Oleh karena itu, tidak ada definisi dari *Internet of Things*. Berikut adalah beberapa definisi alternatif dikemukakan untuk memahami *Internet of Things* (IoT), diantaranya:

Menurut Ashton: “definisi awal IoT adalah *Internet of Things* memiliki potensi untuk mengubah dunia seperti pernah dilakukan oleh Internet, bahkan mungkin lebih baik”. Pernyataan tersebut diambil dari artikel sebagai berikut: “Hari ini komputer dan manusia, hampir sepenuhnya tergantung pada Internet untuk segala informasi yang semua terdiri dari sekitar 50 petabyte (satu petabyte adalah 1.024 terabyte) data yang tersedia pada Internet dan pertama kali digagas dan diciptakan oleh manusia. Dari mulai magnetik, menekan tombol rekam, mengambil gambar digital atau memadai kode bar.

Diagram konvensional dari Internet meninggalkan router menjadi bagian terpenting dari semuanya. Masalahnya adalah orang memiliki waktu, perhatian dan

akurasi terbatas. Mereka semua berarti tidak sangat baik dalam menangkap berbagai data tentang hal di dunia nyata.

Dari segi fisik dan begitu juga lingkungan kita. Gagasan dan informasi begitu penting, tetapi banyak lagi hal yang penting. Namun teknologi informasi saat ini sangat tergantung pada data yang berasal dari orang-orang sehingga komputer kita tahu lebih banyak tentang semua ide dari hal-hal tersebut”

Sementara definisi IoT Menurut Casagras (*Coordinator and support action for global RFID-related activities and standardisation*): “sebagai sebuah infrastruktur jaringan global, yang menghubungkan benda-benda fisik dan virtual melalui eksploitasi data *capture* dan kemampuan komunikasi. Infrastruktur terdiri dari jaringan yang telah ada dan internet berikut pengembangan jaringannya. Semua ini akan menawarkan identifikasi obyek, sensor dan kemampuan koneksi sebagai dasar untuk pengembangan layanan dan aplikasi ko-operatif yang independen. Ia juga ditandai dengan tingkat otonomi data capture yang tinggi, event transfer, konektivitas jaringan dan interoperabilitas”

Internet of Things (IoT) juga dapat diartikan sebagai konsep komputasi tentang objek sehari-hari yang terhubung ke internet dan mampu mengidentifikasi diri ke perangkat lain. Jadi, *Internet of Things* sebenarnya adalah konsep yang cukup sederhana, yang artinya menghubungkan semua objek fisik di kehidupan sehari-hari ke Internet.

Cara Kerja Internet of Things yaitu dengan memanfaatkan sebuah argumentasi pemrograman yang dimana tiap-tiap perintah argumennya itu menghasilkan sebuah interaksi antara sesama mesin yang terhubung secara otomatis

tanpa campur tangan manusia dan dalam jarak berapa pun. Internetlah yang menjadi penghubung di antara kedua interaksi mesin tersebut, sementara manusia hanya bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung

2.2 Predictive Maitenance

Predictive maintenance (PdM) adalah servis peralatan ketika diperkirakan layanan diperlukan, dalam toleransi tertentu. PdM digunakan di jalur kereta api, peralatan industri, pabrik dan pengolahan minyak dan gas hingga penerapannya pada Panel Surya

Memelihara mesin dan elektronik merupakan solusi paling hemat biaya jika dilakukan saat dibutuhkan. Untuk itu, sistem perawatan prediktif dirancang untuk memastikan bahwa servis tidak terjadi terlalu cepat, membuang-buang uang untuk pekerjaan yang tidak perlu, atau terlambat, setelah aus dan waktu telah menyebabkan kerusakan yang tidak semestinya. Sistem PDM juga dapat membantu merencanakan inventaris untuk suku cadang pengganti dan memberikan masukan pada sistem yang memerlukan peningkatan desain karena kinerja yang tidak dapat diterima.

Memelihara mesin dan elektronik paling efisien, ketika dilakukan saat dibutuhkan. Untuk itu, sistem perawatan prediktif dirancang untuk memastikan bahwa servis tidak terjadi terlalu cepat, membuang-buang uang untuk pekerjaan yang tidak perlu, atau terlambat, setelah aus dan waktu telah menyebabkan kerusakan yang tidak semestinya. Sistem PDM juga dapat membantu merencanakan inventaris untuk suku cadang pengganti dan memberikan masukan pada sistem yang memerlukan peningkatan desain karena kinerja yang tidak dapat diterima.

PdM diaktifkan oleh kemajuan teknologi sensor dan komunikasi yang merupakan bagian dari tren otomatisasi dan Internet of Things (IoT) yang sedang berlangsung, dan khususnya industry IoT. Kemajuan tersebut memungkinkan pemantauan dan analisis data yang berkelanjutan untuk kondisi mekanik dan listrik, efisiensi operasional, dan indikator kinerja lainnya.

PdM bekerja melalui penggunaan sensor, seringkali terikat dengan perangkat lunak PdM pada sensor nirkabel dan jaringan aktuator (dikenal : *wireless sensor and actuator network* –WSAN). Perangkat lunak memperhitungkan waktu akun, jarak tempuh atau penggunaan dan perubahan yang terukur, kadang-kadang dari sensor yang dapat mengindikasikan kebutuhan akan layanan sebelum penurunan kinerja atau degradasi peralatan terjadi. Pengukuran ini sering dilakukan pada mesin saat sedang berjalan untuk mengurangi dampak pada produksi

Predictive maintenance ini merupakan perawatan yang dilakukan bersifat prediksi, dalam hal ini merupakan evaluasi dari perawatan berkala. Pendeteksian ini dapat di evaluasi dari indikator-indikator yang terpasang pada instalasi suatu alat dan juga dapat melakukan analisa vibrasi dan alignment untuk menambah data dan tindakan perbaikan selanjutnya. Predictive maintenance lebih mirip dengan preventive maintenance akan tetapi predictive maintenance ini tidak terjadwal secara teratur. Predictive maintenance menganalisa suatu kondisi peralatan dari trend perilaku peralatan. Metode ini merupakan efisiensi dari metode sebelumnya (preventive maintenance), yaitu melakukan persiapan maintenance sebelum waktu kerusakan terjadi namun dengan prediksi waktu yang lebih tepat. Metode ini merupakan efisiensi dari metode sebelumnya (preventive maintenance), yaitu

melakukan persiapan maintenance sebelum waktu kerusakan terjadi namun dengan prediksi waktu yang lebih tepat.

Tujuan dari Predictive maintenance ini adalah untuk mengetahui lebih dini kemungkinan terjadinya kerusakan pada suatu peralatan. Dengan mengetahui kondisi peralatan tersebut kita dapat melakukan tindakan untuk mencegah peralatan tersebut mengalami kerusakan fatal saat sedang beroperasi yang dapat force outage

2.3 Mikrokontroler

Mikrokontroler atau dalam bahasa Indonesia berarti Pengendali Mikro adalah sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronik dan umunya dapat menyimpan program didalamnya. Mikrokontroler adalah sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronik dan umunya dapat menyimpan program did umumnya terdiri dari CPU (Central Processing Unit), memori, I/O tertentu dan unit pendukung seperti *Analog-to-Digital Converter* (ADC) yang sudah terintegrasi di dalamnya. Kelebihan utama dari mikrokontroler ialah tersedianya RAM dan peralatan I/O pendukung sehingga ukuran board mikrokontroler menjadi sangat ringkas.

Semua peralatan yang berhubungan dengan aktivitas kita hampir semua nya memiliki mikrokontroler, contohnya: Handphone yang selalu kita gunakan untuk berkomunikasi, layar LCD, mobil, motor, kamera digital serta masih banyak peralatan yang lain tapi intinya adalah setiap perangkat elektronik yang mempunyai “remote control” hampir pasti mengandung mikrokontroler. Meskipun dalam kehidupan sehari-hari kita selalu berhubungan dengan alat ini, masih banyak orang-

orang yang belum mengetahui apa itu mikrokontroler dan bagaimana alat ini bekerja. Pada dasarnya, mikrokontroler adalah suatu perangkat yang mengintegrasikan sejumlah komponen dari sistem mikroprosesor ke dalam sebuah microchip tunggal. Ada tiga komponen utama dari *mikrokontroler*, yaitu: processor CPU, memory dan input/output(I/O).

2.3.1 Esp 32

ESP32 merupakan mikrokontroler keluaran espressif yang mana disebutkan dalam websitenya sebagai “*A feature-rich MCU with integrated Wi-Fi and Bluetooth connectivity for a wide-range of application*”. Atau dengan kata lain ESP32 adalah *microcontroller unit* (MCU) yang kaya fitur dengan WiFi dan konektivitas Bluetooth terintegrasi untuk beragam aplikasi.

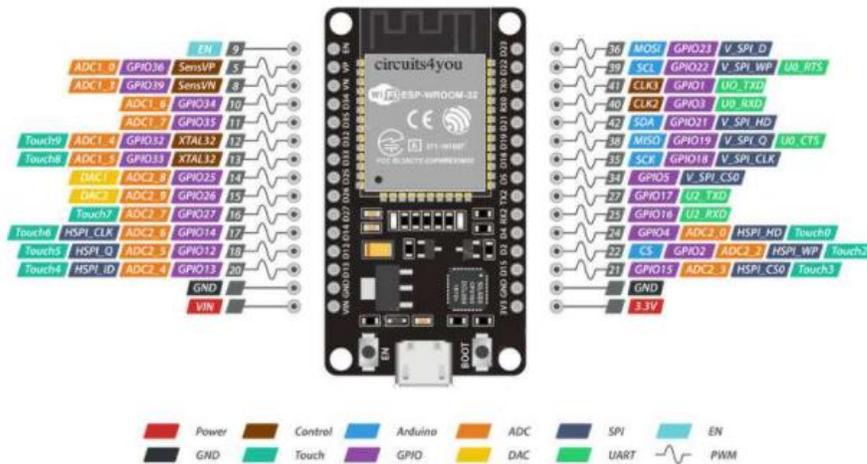
Website Espressif juga menggambarkan kemampuan ESP32, yaitu:

1. *Robust design* : ESP32 mampu berfungsi dengan andal di lingkungan industri, dengan suhu pengoperasian berkisar antara -40°C hingga 125°C. Didukung oleh sirkuit kalibrasi canggih, ESP32 secara dinamis dapat menghapus ketidaksempurnaan sirkuit eksternal dan beradaptasi dengan perubahan kondisi eksternal.
2. *Ultra-Low Power Consumption* : Didesain untuk perangkat *mobile*, perangkat elektronik yang dapat dipakai dan aplikasi IoT, ESP32 mencapai konsumsi daya sangat rendah dengan kombinasi beberapa jenis perangkat lunak berpemilik (*proprietary software*). ESP32 juga mencakup fitur

canggih, seperti gating jam berbutir halus (*fine-grained clock gating*), berbagai mode daya dan penskalaan daya dinamis.

3. *High Level of Integration* : ESP32 sangat terintegrasi, dengan sakelar antena built-in, balun RF, *amplifier* daya, *low-noise receive amplifier*, filter, dan modul manajemen daya. ESP32 menambahkan fungsionalitas dan keserbagunaan yang tak ternilai dengan persyaratan *minimal Printed Circuit Board (PCB)*.
4. *Hybrid Wi-Fi and Bluetooth Chip* : ESP32 dapat berfungsi sebagai sistem mandiri yang lengkap atau sebagai perangkat *slave* untuk *MCU host*, mengurangi *overhead* tumpukan komunikasi pada prosesor utama. ESP32 dapat berinteraksi dengan sistem lain untuk menyediakan fungsi Wi-Fi dan Bluetooth melalui antarmuka SPI / SDIO atau I2C / UART.

Gambar 2.1 menunjukkan mikrokontroler ESP32 sebagai penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul WiFi dan bluetooth sehingga sangat mendukung untuk membuat sistem aplikasi *Internet of Things*. Memiliki 18 ADC (*Analog Digital Converter*), 2 DAC, 16 PWM, 10 sensor sentuh, 2 jalur antarmuka UART, pin antarmuka I2C, I2S, dan SPI. (G, 2019)



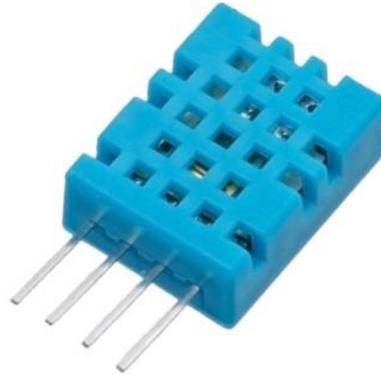
Gambar 2. 1 ESP 32

2.4 Sensor-sensor

Sensor ialah perangkat yang digunakan untuk mendeteksi perubahan besaran fisik seperti tekanan, gaya, besaran listrik, cahaya, gerakan, kelembaban, suhu, kecepatan hingga fenomena-fenomena lingkungan lainnya. Setelah mengamati terjadinya perubahan, Input yang terdeteksi tersebut akan dikonversi mejadi Output yang dapat dimengerti oleh manusia baik melalui perangkat sensor itu sendiri ataupun ditransmisikan secara elektronik melalui jaringan untuk ditampilkan atau diolah menjadi informasi yang bermanfaat bagi penggunaanya

Sensor sering didefinisikan sebagai “perangkat yang menerima dan menanggapi sinyal atau stimulus.” Definisi tersebut masih terlihat cukup luas, sebagai contoh mata manusia yang kemudian dapat digunakan untuk memicu suatu tindakan tertentu dalam hal ini ialah menangkap Cahaya

2.4.1 Sensor suhu dan kelembaban (DHT11)



Gambar 2. 2 DHT 11

Gambar 2.2 menunjukkan DHT11 yang merupakan sensor temperatur dan kelembaban dengan *output* sinyal digital yang dikalibrasi. Dengan menggunakan teknik akuisisi sinyal digital eksklusif dan teknologi pengindraan suhu & kelembaban, memastikan keandalan tinggi dan stabilitas jangka panjang yang sangat baik. Sumber daya listrik yang dibutuhkan oleh DHT11 adalah 3 V – 5,5 V DC. Format *single-bus data* digunakan untuk komunikasi antara MCU dan sensor DHT11. Satu proses komunikasi sekitar 4 ms. Data terdiri dari bagian desimal dan integral. Transmisi data yang lengkap adalah 40 *bit*, dan sensor mengirimkan *bit* data yang lebih tinggi terlebih dahulu. Format data: data RH integral 8 *bit* + data RH desimal 8 *bit* + data T integral 8 *bit* + data T desimal 8 *bit* + jumlah cek 8 *bit*. Jika transmisi data benar, *check-sum* harus menjadi 8 *bit* terakhir dari "data RH 8 *bit* integral + data RH desimal 8 *bit* + data T integral 8 *bit* + data T desimal 8 *bit*".

Sensor DHT11 ini adalah salah satu sensor yang dapat mengukur dua parameter lingkungan sekaligus, yakni suhu dan kelembaban udara (*humidity*). Dalam sensor ini

terdapat sebuah *thermistor* tipe NTC (*Negative Temperature Coefficient*) untuk mengukur suhu, sebuah sensor kelembaban tipe resistif dan sebuah mikrokontroler 8 bit yang mengelola kedua sensor tersebut dan mengirim hasilnya ke pin *output* dengan format *single-wire bi-directional* (kabel tunggal dua arah). (Ajie, 2016)

2.4.2 Sensor cahaya (BH1750)



Gambar 2. 3 Sensor Cahaya BH1750

GY-302 Digital Light Intensity Sensor Module adalah sebuah modul sensor cahaya berbasis IC BH1750 seperti pada **gambar 2.3**. BH1750 adalah sebuah IC sensor cahaya dengan antarmuka IC. Modul ini memberikan nilai output digital melalui IC bus, sehingga Anda tidak perlu lagi menambahkan konverter ADC

BH1750FVI adalah sebuah IC sensor yang digunakan untuk mengukur intensitas cahaya sekitar dalam ukuran atau satuan lux. Sensor ini menggunakan protokol I2C untuk komunikasi dengan mikrokontroler

Modul sensor intensitas cahaya BH1750 adalah sensor cahaya digital yang memiliki keluaran sinyal digital, sehingga tidak memerlukan perhitungan yang rumit. Sensor BH1750 ini lebih akurat dan lebih mudah digunakan jika dibandingkan

dengan sensor lain seperti foto diode dan LDR yang memiliki keluaran sinyal analog dan perlu melakukan perhitungan untuk mendapatkan data intensitas. Sensor cahaya digital BH1750 ini dapat melakukan pengukuran dengan keluaran lux (lx) tanpa perlu melakukan perhitungan terlebih dahulu.

Jangkauan deteksi sensor ini cukup lebar yaitu antara 1 – 65535 lux. 1 lux artinya besaran intensitas cahaya 1 lumen pada area seluas 1 meter persegi atau kalau dituliskan dalam persamaan menjadi :

$$1 \text{ Lux} = 1 \text{ Lm} / \text{m}^2$$

2.4.3 Sensor debu (GP2Y1010AU0F Optical Dust Sensor)



Gambar 2. 4 Sensor Debu

GP2Y1010AU0F Optical Dust Sensor pada **gambar 2.4** merupakan sensor debu yang berbasis inframerah. Sensor ini sangat efektif dalam mendeteksi partikel yang sangat halus seperti debu atau asap rokok, dan umumnya digunakan dalam sistem pembersih udara. Sensor GP2Y1010AU0F adalah sensor debu yang

memanfaatkan hamburan cahaya atau disebut dengan sistem penginderaan optic. Prinsip kerja dari sensor ini ialah dengan mendeteksi debu ataupun partikel yang lain kemudian akan di pantulkan cahaya ke bagian penerima. Cahaya dicerminkan pada partikel melewati keseluruhan permukaan, kemudian oleh photodiode diubah menjadi tegangan

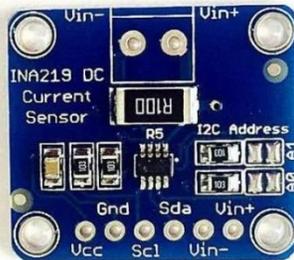
Tegangan harus diperkuat untuk dapat membaca perubahan. Output dari sensor adalah tegangan analog sebanding dengan kepadatan debu yang terukur, dengan sensitivitas 0.5V/0.1 mg/m³. Untuk dapat mengkonversi data analog menjadi data digital 10 bit maka digunakan persamaan (1) sebagai berikut

$$\text{calcVoltage} = \text{voMeasured} * (5.0/1024)$$

Selanjutnya untuk mengkonversi data digital menjadi suatu nilai konsentrasi paparan debu digunakan persamaan (2) sebagai berikut.

$$\text{dustDensity} = 0.17 * \text{calcVoltage} - 0.1$$

2.4.4 Sensor INA 219 dc current monitor



Gambar 2. 5 Sensor Arus & Tegangan

Gambar 2.5 menunjukkan Adafruit INA 219 dc current monitor yang merupakan sensor untuk mengukur tegangan hingga 26V dan arus hingga 3,2A. dan sensor ini juga dapat merasakan tegangan, arus dan daya pada saat yang sama, sehingga sangat cocok untuk dipakai pada aplikasi energi monitoring seperti pada solar cell atau battery charger Dengan konektifitas menggunakan I2C, dapat dengan mudah terhubung dengan Arduino atau pun dengan microcontroller lainnya yang memiliki I2C interface

2.5 Amazon Web Service

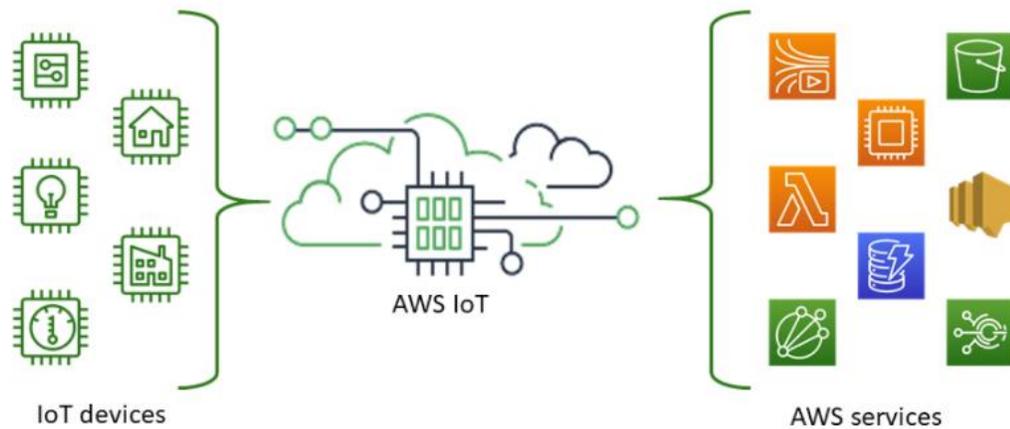
Amazon Web Services atau disingkat AWS ialah Platform cloud paling komprehensif dan digunakan secara luas di dunia. AWS menawarkan sangat banyak layanan, kurang lebih menawarkan 175 layanan unggulan yang cukup lengkap dari pusat data secara global. Beberapa Startup besar Dunia dengan pertumbuhan tercepat hingga lembaga pemerintah yang terkemuka juga ikut menggunakan Amazon Web Services (AWS) untuk memangkas biaya operasional, menjadi lebih sigap dan adaptif dan mendorong inovasi untuk menjadi lebih cepat.

2.5.1 IoT Core

AWS IoT core ialah layanan cloud dari AWS yang menghubungkan perangkat IoT yang kita miliki dengan perangkat lain dan layanan cloud dari AWS itu sendiri. AWS IoT core menyediakan perangkat lunak yang dapat membantu kita dalam mengintegrasikan perangkat yang kita miliki ke dalam solusi berbasis IoT dalam AWS. Jika perangkat Anda dapat terhubung ke AWS IoT, maka AWS IoT dapat menghubungkannya ke layanan cloud yang disediakan oleh AWS

AWS IoT Core juga menyediakan komunikasi dua arah yang aman untuk perangkat yang terhubung ke Internet (seperti sensor, aktuator, perangkat tertanam, perangkat nirkabel, dan peralatan pintar) untuk terhubung ke AWS Cloud melalui MQTT, HTTPS, dan LoRaWAN.

Layanan dari AWS IoT Core ini menghubungkan perangkat IoT yang kita miliki ke layanan AWS IoT dan layanan AWS lainnya seperti pada **gambar 2.6**. AWS IoT Core juga menyertakan gateway perangkat dan perantara pesan, yang menghubungkan dan memproses pesan antara perangkat IoT Anda dan cloud



Gambar 2. 6 IoT Core

AWS IoT core memungkinkan kita untuk memilih teknologi apa yang paling tepat dan terkini untuk solusi yang kita butuhkan yang bertujuan untuk membantu dalam mengelola dan mendukung perangkat IoT yang kita miliki.

2.5.2 IoT Analytics

AWS IoT Analytics dapat digunakan untuk mengotomatisasi langkah-langkah yang diperlukan untuk menganalisis data dari perangkat IoT. AWS IoT Analytics juga dapat memfilter, mengubah, dan memperkaya data IoT sebelum menyimpannya di penyimpanan data deret waktu atau dikenal dengan *time-series data store* untuk dapat di analisis nantinya. Kita dapat memilih servis layanan untuk mengumpulkan data yang hanya kita butuhkan dari perangkat kita sendiri, mulai dari menerapkan transformasi matematika untuk memproses data, dan memperkaya data dengan metadata khusus perangkat seperti jenis dan lokasi perangkat sebelum menyimpan data tersebut. Kemudian, kita juga dapat menganalisis data dengan menjalankan kueri menggunakan mesin kueri SQL bawaan, atau melakukan analisis yang lebih kompleks dan inferensi pembelajaran mesin atau dikenal dengan *Machine Learning*. AWS IoT Analytics juga memungkinkan kita untuk dapat mengeksplorasi data tingkat lanjut dengan mengintegrasikannya dengan Jupyter Notebook. AWS IoT Analytics juga memungkinkan kita untuk melakukan visualisasi data dengan Amazon QuickSight.

Alat analitik tradisional dirancang untuk memproses data terstruktur. Sementara data IoT sering kali bersifat mentah (belum terstruktur) karena berasal dari perangkat yang merekam data yang kurang terstruktur (seperti data mentah dari suhu, gerakan, atau suara) dalam hal ini ialah Sensor-sensor yang kita gunakan. Akibatnya data dari perangkat ini dapat memiliki celah yang signifikan, ataupun pesan yang rusak, dan pembacaan yang salah yang harus diperbaiki terlebih dahulu sebelum analisis dapat dilakukan.

AWS IoT Analytics juga memungkinkan kita untuk dapat mengatasi masalah ini dan mengumpulkan data perangkat dalam jumlah besar, memproses pesan, dan kemudian menyimpannya. Kita juga dapat membuat data tersebut menjadi data yang terstruktur kemudian menganalisisnya.

Cara kerja AWS IoT Analytics seperti pada **gambar 2.7** .



Gambar 2. 7 IoT Analytics

Yg pertama ialah *Collect* (Mengumpulkan Data dari berbagai format)

- Terintegrasi dengan AWS IoT Core. AWS IoT Analytics terintegrasi penuh dengan AWS IoT Core sehingga dapat menerima pesan dari perangkat yang terhubung
- Gunakan API batch untuk menambahkan data dari sumber mana pun. AWS IoT Analytics dapat menerima data dari sumber mana pun melalui HTTP. Artinya, perangkat atau layanan apa pun yang terhubung ke internet dapat mengirim data ke AWS IoT Analytics.

- Kumpulkan data yang hanya ingin kita simpan dan analisis. Kita dapat menggunakan konsol AWS IoT Analytics untuk mengonfigurasi AWS IoT Analytics untuk menerima pesan dari perangkat melalui filter topik MQTT dalam berbagai format dan frekuensi. AWS IoT Analytics memvalidasi bahwa data berada dalam parameter tertentu yang Anda tentukan dan buat saluran. Kemudian, layanan merutekan saluran ke jalur pipa yang sesuai untuk pemrosesan pesan, transformasi, dan pengayaan

Yg kedua ialah Process (memproses data)

- Bersihkan dan filter. AWS IoT Analytics memungkinkan kita untuk dapat menentukan fungsi AWS Lambda yang dipicu saat AWS IoT Analytics mendeteksi data yang hilang, sehingga kita dapat menjalankan kode untuk memperkirakan dan mengisi celah. Anda juga dapat menentukan filter maksimum dan minimum serta memberi batasan untuk menghapus outlier dalam data Anda
- Transform. AWS IoT Analytics dapat mengubah pesan menggunakan logika matematis atau kondisional yang telah kita tetapkan, sehingga memungkinkan kita untuk melakukan penghitungan umum, contohnya seperti konversi Celsius ke Fahrenheit
- Memperkaya. AWS IoT Analytics dapat memperkaya data dengan sumber data eksternal seperti prakiraan cuaca, lalu merutekan data ke penyimpanan data AWS IoT Analytics

Yg ketiga ialah Store (menyimpan data)

- Penyimpanan data *time-series*. AWS IoT Analytics menyimpan data perangkat di penyimpanan data *time-series* yang dioptimalkan untuk pengambilan dan analisis yang lebih cepat. Kita juga dapat mengelola izin akses, menerapkan kebijakan penyimpanan data, dan mengekspor data Anda ke titik akses eksternal
- Simpan data yang diproses dan data mentah. AWS IoT Analytics menyimpan data yang diproses dan juga secara otomatis menyimpan data mentah yang diserap sehingga memungkinkan kita untuk memprosesnya di lain waktu

Yg keempat ialah Analyze (menganalisis data)

- Menjalankan query SQL Ad-hoc. AWS IoT Analytics menyediakan mesin kueri SQL sehingga kita bisa menjalankan kueri ad-hoc dan mendapatkan hasil dengan cepat. Layanan ini memungkinkan kita untuk dapat menggunakan kueri SQL standar untuk mengekstrak data dari penyimpanan data untuk menjawab pertanyaan seperti jarak rata-rata yang ditempuh untuk armada kendaraan yang terhubung atau berapa banyak pintu di gedung pintar yang dikunci setelah jam 7 malam. Kueri ini dapat digunakan kembali meskipun perangkat yang terhubung, ukuran armada, dan persyaratan analitik berubah
- Analisis rangkaian waktu. AWS IoT Analytics mendukung analisis rangkaian waktu sehingga memungkinkan kita untuk dapat menganalisis kinerja perangkat dari waktu ke waktu dan memahami bagaimana dan di mana perangkat tersebut digunakan, terus memantau data perangkat untuk

memprediksi masalah pemeliharaan, dan memantau sensor untuk memprediksi dan bereaksi terhadap keadaan lingkungan

- Notebook yang dihosting untuk analitik canggih dan pembelajaran mesin. AWS IoT Analytics menyertakan dukungan untuk notebook yang dihosting, dalam hal ini ialah Jupyter Notebook untuk analisis statistik dan pembelajaran mesin. Layanan ini mencakup satu set template notebook yang berisi model dan visualisasi pembelajaran mesin yang ditulis AWS. Anda dapat menggunakan template untuk memulai kasus penggunaan IoT yang terkait dengan profil kegagalan perangkat, memperkirakan peristiwa seperti penggunaan rendah yang mungkin menandakan pelanggan akan meninggalkan produk, atau mengelompokkan perangkat berdasarkan tingkat penggunaan pelanggan (misalnya pengguna berat, pengguna akhir pekan) atau kesehatan perangkat. Setelah Anda menulis buku catatan, kemudian kita bisa menyimpan dan menjalankannya pada jadwal yang telah kita tentukan.
- Prediksi. Kita bisa melakukan klasifikasi statistik melalui metode yang disebut regresi logistik. Kita juga dapat menggunakan Long-Short-Term Memory (LSTM), yang merupakan teknik jaringan saraf yang kuat untuk memprediksi output atau status proses yang bervariasi dari waktu ke waktu. Template notebook yang dibuat sebelumnya juga mendukung algoritme pengelompokan K-means untuk segmentasi perangkat, yang mengelompokkan perangkat kita ke dalam kelompok perangkat yang serupa. Template ini biasanya digunakan untuk profil kesehatan perangkat dan status

perangkat seperti unit HVAC di pabrik cokelat atau keausan blade pada turbin angin. Sekali lagi, template buku catatan ini dapat dimuat dan dijalankan sesuai jadwal yang kita tentukan.

2.6 Arima

Model *Auto-Regressive Integrated Moving Average* (ARIMA) adalah model yang secara penuh mengabaikan independen variabel dalam pembuatan peramalan. ARIMA menggunakan nilai masa lalu dan sekarang dari variabel dependen untuk menghasilkan peramalan jangka pendek yang akurat yang bertujuan untuk menentukan hubungan statistik yang baik antar variabel yang diramal dengan nilai historis variabel tersebut sehingga peramalan dapat dilakukan dengan model tersebut.

Deret berkala atau *time series* merupakan rangkaian observasi atau pengamatan yang diambil pada waktu yang berurutan. Waktu pengambilan data dapat berupa periode detik, hari, tahun, atau satuan waktu yang lainnya. Contoh dari data deret berkala ialah jumlah barang yang dikirim dari pabrik setiap bulan, jumlah kecelakaan di jalan setiap minggunya, jumlah curah hujan harian dan pengamatan per jam pada proses kimia dan lain sebagainya.

Proses peramalan dilakukan jika data deret berkala telah stasioner. Data telah bersifat stasioner apabila varian, rata-rata, dan *auto-correlation* bersifat konstan dari waktu ke waktu. Penggunaan stasioneritas data dapat dilakukan dengan menggunakan *Augmented Dickey Fuller Test* atau ADF test.

Kelas model paling umum untuk peramalan data deret berkala yang dapat dibuat menjadi stasioner dengan melakukan diferensiasi dan mungkin bersamaan

dengan transformasi nonlinier seperti *logging* atau *deflating* jika diperlukan. *Lags* dari deret stasioner dalam persamaan peramalan disebut dengan istilah “*autoregressive*”, *Lags* dari kesalahan peramalan disebut dengan istilah “*moving average*”. Dan sebuah deret berkala yang membutuhkan proses pembedaan untuk membuat deret tersebut menjadi stasioner disebut sebagai bersi “*integrated*” dari deret stasioner. Model *randome-walk*, *radom trend*, *autoregressive*, dan *exponential smoothing* adalah kasus-kasus model ARIMA (Nau, 2018).

2.6.1 Klasifikasi Model ARIMA

Model ARIMA dibagi dalam 3 unsur, yaitu: model autoregresif(AR), moving average(MA), dan Integreted(I). ketiga unsur ini bisa dimodifikasi sehingga membentuk model baru. misalnya model autoregresif dan moving average (ARMA). namun, apabila mau dibuat dalam bentuk umumnya menjadi ARIMA(p,d,q). p menyatakan ordo AR, d menyatakan ordo Integreted dan q menyatakan ordo moving avirage. apabila modelnya menjadi AR maka model umumnya menjadi ARIMA(1,0,0). untuk lebih jelasnya berikut dijelaskan untuk masingmasing unsur :

- Aoutoregresif (AR)

Bentuk umum dari model autoregresif dengan ordo p (AR(p)) atau model ARIMA(P,0,0) dinyatakan sebagai berikut:

$$X_t = \mu' + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + e_t$$

Dimana : μ' = Suatu konstanta

ϕ_p = Parameter autoregresif ke-p

e_t = nilai kesalahan pada t

Maksud dari autoregresif yaitu nilai X dipengaruhi oleh nilai x periode sebelumnya hingga periode ke-p. Jadi variabel adalah hal yang sangat berpengaruh.

- Moving Average (MA)

Bentuk umum dari model moving average dengan ordo q (MA(q)) atau model ARIMA(0,0,q) dinyatakan sebagai berikut:

$$X_t = \mu' + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \dots - \theta_q e_{t-k}$$

Dimana : μ' = Suatu konstanta

θ_1 sampai θ_q adalah parameter-parameter *moving average*

e_{t-k} = nilai kesalahan pada saat $t - k$

Maksud dari moving average yaitu nilai variabel x dipengaruhi oleh error dari variabel x tersebut.

- Integrated (I)

Bentuk umum dari model integrated dengan ordo d (I(d)) atau model ARIMA(0,d,0). integrated disini adalah menyatakan difference dari data. maksudnya bahwa dalam membuat model ARIMA syarat keharusan yang harus dipenuhi adalah stasioneritas data. apabila data stasioner pada level maka ordonya sama dengan 0, namun apabila stasioner pada different pertama maka ordonya 1, dst.

Model ARIMA dibagi dalam 2 bentuk, yaitu model ARIMA tanpa musiman dan model ARIMA musiman. model ARIMA tanpa musiman merupakan model ARIMA yang tidak dipengaruhi oleh faktor waktu musim. bentuk umum dapat dinyatakan dalam persamaan berikut:

$$(\mathbf{1} - \mathbf{B})(\mathbf{1} - \phi_1 \mathbf{B})X_t = \mu' + (\mathbf{1} - \theta_1 \mathbf{B})e_t$$

Sedangkan ARIMA musiman merupakan model ARIMA yang dipengaruhi oleh faktor waktu musim. Model ini biasa disebut Seasonal ARIMA(SARIMA). bentuk umum dinyatakan sebagai berikut:

$$(\mathbf{1} - \mathbf{B})(\mathbf{1} - \mathbf{B}^{12})X_t = (\mathbf{1} - \theta_1 \mathbf{B})(\mathbf{1} - \Theta_1 \mathbf{B}^{12})e_t$$

Tapi untuk menggunakannya dalam peramalan mengharuskan dilakukan suatu penjabaran dari perasamaan tersebut dan menjadikannya sebuah persamaan regresi yang lebih umum. Untuk model diatas bentuknya adalah:

$$X_t = X_{t-1} + X_{t-12} - X_{t-13} + e_t - \theta_1 e_{t-12} + \theta_1 \Theta_1 e_{t-13}$$

Untuk meramalkan satu periode ke depan, yaitu X_{t+1} maka seperti pada persamaan berikut:

$$X_{t+1} = X_t + X_{t-11} - X_{t-12} + e_{t+1} - \theta_1 e_{t-11} + \theta_1 \Theta_1 e_{t-12}$$

2.6.2 Tahap pembuatan model ARIMA dan SARIMA

ARIMA dan SARIMA merupakan metode analisis time series, hal yang penting yang perlu diperhatikan dalam menganalisis data time series adalah ketepatan model. Meskipun ARIMA & SARIMA tampaknya lebih modern dibandingkan metode lainnya, karena model persamaannya yang complicated dan terkesan high class, namun persamaan yang diperoleh tetap harus dibandingkan dengan analisis lainnya. Model yang baik adalah model yang memiliki nilai MSE atau MSD yang paling kecil.

Adapun tahap-tahapan pembuatan model ARIMA adalah sebagai berikut:

- Identifikasi

Identifikasi data yang akan kita proses apakah mengandung tren, atau seasonal. Arima Sarima mensyaratkan data yang diproses harus stasioner. Stasioner dalam istilah saya artinya data berfluktuatif dalam nilai tertentu, tidak menunjukkan tren kenaikan atau penurunan. Jika ternyata data menunjukkan adanya tren, kita lakukan differencing atau pembedaan. Pembedaan pertama atau jika diperlukan pembedaan kedua. Tujuannya? Agar data menjadi stasioner.

- Pendugaan parameter (Estimasi)

Setelah data kita pastikan stasioner, kita mulai mengestimasi model Arima atau Sarima. Cara estimasinya dengan cara mengamati ACF dan PACF. Apa itu? ACF merupakan singkatan dari Autocorrelation Function, yang mengindikasikan nilai autoregressive. PACF merupakan singkatan Partial Autocorrelation Function, yang mengindikasikan nilai Moving Averagenya. Dalam menentukan data mengandung Autoregressive (AR) atau Moving Average (MA) adalah melihat pola atau perilaku ACF dan PACFnya. Hal yang perlu dipahami dalam bagian ini adalah anda harus mengerti mana pola yang dikatakan *Cut Off*, dan mana pola yang dikatakan *dying down*.

Pola cut off terjadi apabila data mendekati nilai 0 pada lag – lag awal atau terlihat gambar yang langsung menurun drastis (cut off). Sedangkan dying down biasanya terlihat menurun perlahanlahan mendekati nilai 0. Belum ada kepastian berapa lag yang termasuk dalam lag awal, ada yang menjelaskan 5 data awal, ada juga yang menjelaskan 10 data awal. Namun biasanya kita melihat pola dari gambar sudah dapat diketahui apakah itu menurun perlahan atau langsung drastis/cut off.

Biasanya pada pola cut off, nilai $|T|$ langsung tidak signifikan pada lag 2 atau 3, sedangkan pola *dying down* nilai $|T|$ signifikan pada lag-lag awal. Data signifikan apabila nilai $|T| > 2$ untuk Arima, dan $|T| > 1.25$ untuk Sarima.

Pola ACF dan PACF juga mencerminkan kestasioneran data. Data yang belum stasioner biasanya memiliki pola *dying down* dengan nilai $|T|$ yang signifikan di hampir semua lag (dominan). Maka bila menjumpai data keduanya (PACF dan ACF) *dying down* dengan nilai $|T|$ yang signifikan, sebaiknya melakukan identifikasi ulang dan melakukan tahap pembedaan / *differencing*.

- Cek Diagnostik (Evaluasi Model)

Setelah menduga parameter, langkah selanjutnya adalah menguji model apakah modelnya sudah baik untuk digunakan. Untuk melihat model yang baik bisa dilihat dari residualnya. Jika residualnya white noise, maka modelnya dapat dikatakan baik dan sebaliknya.

Salah satu cara untuk melihat white noise dapat diuji melalui korelogram ACF dan PACF dari residual. Bila ACF dan PACF tidak signifikan, ini mengindikasikan residual white noise artinya modelnya sudah cocok.

Selain itu dapat dilakukan dengan test Ljung- Box untuk mengetahui white noisenya. Apabila hipotesis awalnya diterima maka residual memenuhi syarat white noise. Sebaliknya jika hipotesis awalnya ditolak maka residual tidak white noise. Statistik uji Ljung-Box sebagai berikut:

$$Q^* = n(n+2) \sum_{k=1}^K \frac{\hat{r}_k^2}{n-k}$$

Dimana : $n' = n-(n+SD)$

d = ordo pembedaan bukan faktor musiman

D = ordo pembedaan faktor musiman

S = jumlah periode permusim

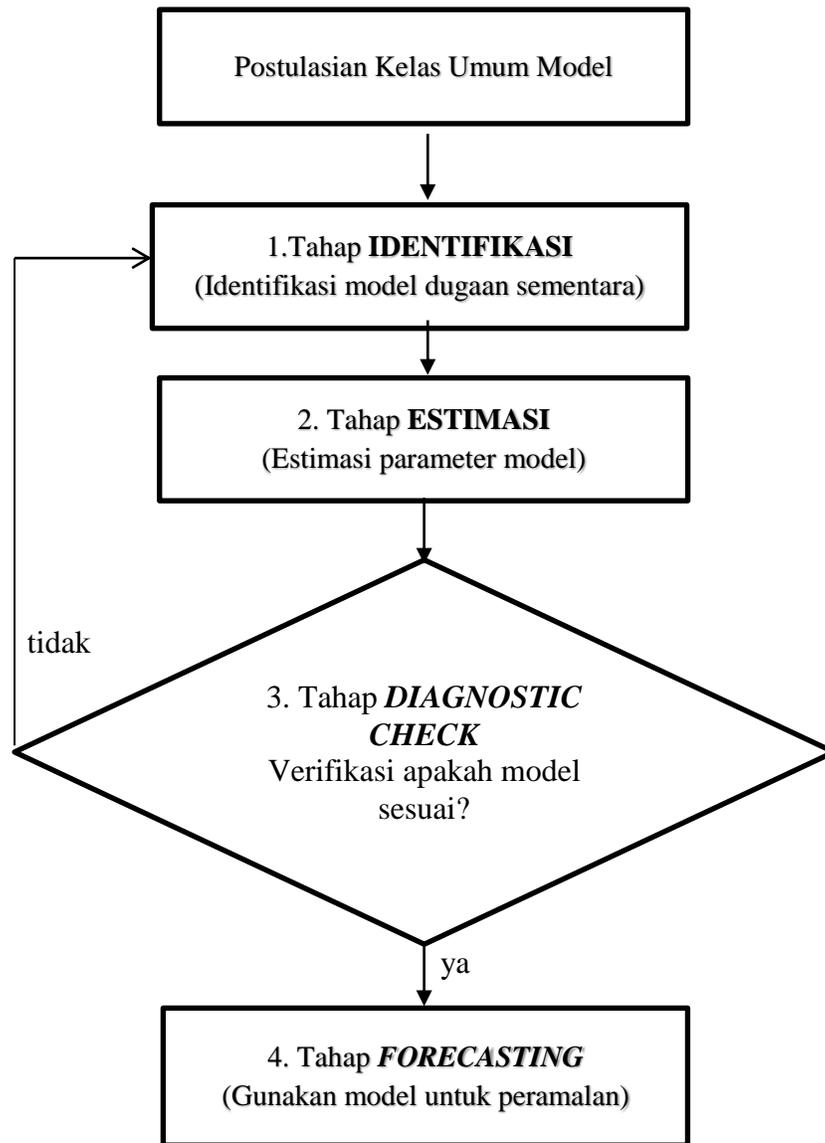
m = lag waktu maksimum

r_k = auto korelasi untuk time lag 1,2,3,....., k

Dari hasil tersebut mungkin saja ada beberapa model yang baik digunakan. Sehingga langkah selanjutnya dengan memilih model terbaik dengan melihat beberapa indicator lain, seperti AIC, SIC, R2adjusted.

- *Forecasting*

Setelah ketiga tahap itu dilewati maka dapat dilakukan peramalan. Setelah mengetahui model, maka langkah terakhir sesuai tujuan penelitian adalah melakukan peramalan data. Peramalan ini juga sesungguhnya merupakan penjabaran dari persamaan berdasarkan koefisien-koefisien yang didapat, sehingga kita dapat menentukan kondisi di masa yang akan datang. Peramalan tidak hanya dilakukan untuk masa mendatang saja, melainkan bisa juga digunakan untuk menentukan berapa nilai seharusnya jika menggunakan model yang kita peroleh. Yang bertujuan untuk menentukan nilai error dalam kasus tertentu.



Sebuah peramalan akan selalu mengandung ketidakpastian. Penyimpangan hasil peramalan selain disebabkan adanya *error* juga dapat disebabkan karena tidak ada metode peramalan yang mampu menghasilkan peramalan dengan sangat akurat. Hal ini juga dapat disebabkan karena metode atau model yang digunakan tidak dapat

memprediksi dengan baik komponen *trend*, musiman ataupun siklus yang terdapat pada deret data (Sungkawa & Megasari, 2011). *Mean Square Error* (MSE) adalah salah satu metode yang dapat digunakan untuk mengevaluasi model dengan rumus:

$$MSE = \frac{\sum_{t=1}^n (At - Ft)^2}{N}$$

Dimana : At = Nilai Aktual Permintaan

Ft = Nilai hasil peramalan

N = banyaknya data