

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Internet of Things (IoT)

*Internet of Things* atau dikenal juga dengan singkatan IoT merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif.



**Gambar 2.1** Ilustrasi dari *Internet of Things*

*Internet of things* dapat didefinisikan kemampuan berbagai device yang bisa saling terhubung dan saling bertukar data melalui jaringan internet. IoT merupakan sebuah teknologi yang memungkinkan adanya sebuah pengendalian, komunikasi, kerjasama dengan berbagai perangkat keras, data melalui jaringan internet. Sehingga bisa dikatakan bahwa *internet of things* (IoT) adalah ketika kita menyambungkan sesuatu (*things*) yang tidak dioperasikan oleh manusia ke internet.

*Internet of Things* bukan hanya terkait dengan pengendalian perangkat melalui perangkat jauh, tapi juga bagaimana berbagai data, memvirtualisasikan segala hal nyata ke dalam bentuk internet dan lain-lain. Internet menjadi sebuah penghubung antara sesama mesin secara otomatis. Selain itu juga ada *user* yang bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung. Manfaatnya menggunakan teknologi IoT yaitu pekerjaan yang dilakukan oleh manusia menjadi lebih cepat, muda, dan efisien.[1]

### **2.1.1 Contoh penerapan IOT**

Ada banyak bidang-bidang internet of things baik yang sudah berjalan maupun masih dalam tahap pengembangan dan perencanaan. Adapun diantaranya yaitu:

1. *Smart Helmet Visit*, Baru-baru ini muncul salah satu inovasi helm dimana pengguna dapat mengakses GPS dan map langsung dari helm. Melalui layar kecil yang tepat berada di depan mata, pengguna bisa mengetahui rute perjalanan dan jarak tempuh. Bahkan helm yang terintegrasi dengan *internet of things* ini dapat memberikan informasi seputar kondisi mesin kendaraan.
2. Manajemen dan pemantauan peralatan otomatis jarak jauh, Ini memungkinkan sistem terpusat untuk mengontrol dan memantau semua proses perusahaan. Memberi perusahaan kemampuan yang belum pernah terjadi sebelumnya untuk mengawasi kemajuan produksi mereka secara *real-time*, sekaligus dapat menganalisis data historis yang diperoleh sehubungan dengan proses produksi perusahaan.
3. Kontrol kualitas, Kemampuan untuk memantau kualitas produk manufaktur pada setiap tahap, mulai dari bahan mentah yang digunakan dalam proses, hingga cara pengangkutannya (melalui aplikasi pelacakan cerdas), hingga ke reaksi pelanggan setelah produk diterima.
4. Transportasi, Manajemen dan informasi lalu lintas. Setiap pengguna dapat berbagi informasi tentang rute yang mereka lalui pada pengendara lain. Dengan demikian, titik padat tidak akan memburuk menjadi macet karena pengendara akan mengambil jalur lain yang lebih lapang.

5. Kesehatan, seperti *Healthmap* memberi tahu data terbaru dimana dan seberapa cepat virus berkembang. Kemudian contoh peranan *internet of things* dapat mengidentifikasi dan mendeteksi kemungkinan adanya kemungkinan suatu virus atau penyakit dalam tubuh seseorang. Memantau dan memberikan informasi pasien dari jarak jauh di suatu rumah sakit.

### 2.1.2 Sistem pada IoT

Teknologi *Internet of Things* adalah konsep perangkat yang mampu mentransfer data tanpa perlu terhubung dengan manusia, melainkan internet sebagai mediana. Sederhananya manusia tidak perlu mengontrol benda/perangkat *IoT* tersebut secara langsung. Melainkan manusia bisa mengontrol benda tersebut dari jarak jauh. Sistem dasar dari IoT terdiri dari cara kerja mengacu pada 3 elemen utama yaitu:

1. Hardware / fisik (*Things*)
2. Koneksi Internet
3. *Cloud data center* sebagai tempat untuk menyimpan data base dan menjalankan aplikasi.

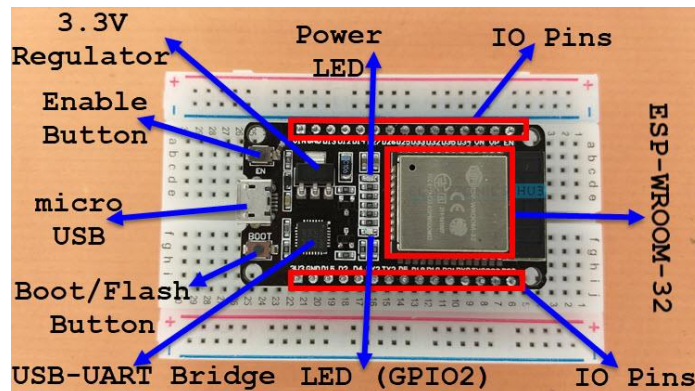


**Gambar 2.2** Skema IoT

## 2.2 NodeMCU ESP32

Dengan pengembangan teknologi IoT yang begitu besar baik hanya sekedar hobi atau kebutuhan komersial, sehingga muncul pengembangan terhadap board-board yang kecil dan multifungsi serta memiliki harga yang terjangkau, seperti Arduino UNO, ESP8266, ESP32, dll. Jika ingin menambahkan konektivitas Wi-Fi dalam sebuah proyek, maka ESP8266 adalah pilihan yang bagus. Tetapi jika ingin membangun sistem yang lengkap dengan konektivitas Wi-Fi, konektivitas Bluetooth, ADC, DAC, Konektivitas Serial, dan banyak fitur lainnya, maka ESP32 adalah pilihan utama.

ESP32 adalah Mikrokontroler System on Chip (SoC) berbiaya rendah dari Espressif Systems, yang juga sebagai pengembang dari SoC ESP8266 yang terkenal dengan NodeMCU. ESP32 adalah penerus SoC ESP8266 dengan menggunakan Mikroprosesor Xtensa LX6 32-bit Tensilica dengan Wi-Fi dan Bluetooth yang terintegrasi. Hal yang baik tentang ESP32, seperti ESP8266 adalah komponen RF terintegrasi seperti Power Amplifier, Low-Noise Receive Amplifier, Antena Switch, dan Filter. Hal ini membuat perancangan hardware pada ESP32 menjadi sangat mudah karena hanya memerlukan sedikit komponen eksternal. Hal penting yang perlu diketahui tentang ESP32 adalah diproduksi menggunakan teknologi 40 nm ultra-low-power TSMC. Jadi, dapat dioperasikan dengan baterai yang umum seperti yang sudah digunakan pada perangkat perlengkapan audio, monitoring, smartwatch.[4]



**Gambar 2.3** keterangan papan nodemcu32

### 2.2.1 Spesifikasi ESP32

ESP32 memiliki lebih banyak fitur daripada ESP8266. Memulai dengan ESP32 ini. Berikut ini daftar beberapa spesifikasi penting dari ESP32. Tetapi untuk spesifikasi lengkap, dapat melihat pada Datasheet.

- 1) Single or Dual-Core 32-bit LX6 Microprocessor with clock frequency up to 240 MHz.
- 2) 520 KB of SRAM, 448 KB of ROM and 16 KB of RTC SRAM.
- 3) Supports 802.11 b/g/n Wi-Fi connectivity with speeds up to 150 Mbps.
- 4) Support for both Classic Bluetooth v4.2 and BLE specifications.
- 5) 34 Programmable GPIOs.
- 6) Up to 18 channels of 12-bit SAR ADC and 2 channels of 8-bit DAC
- 7) Serial Connectivity include 4 x SPI, 2 x I<sup>2</sup>C, 2 x I<sup>2</sup>S, 3 x UART.
- 8) Ethernet MAC for physical LAN Communication (requires external PHY).
- 9) 1 Host controller for SD/SDIO/MMC and 1 Slave controller for SDIO/SPI.
- 10) Motor PWM and up to 16-channels of LED PWM.
- 11) Secure Boot and Flash Encryption.
- 12) Cryptographic Hardware Acceleration for AES, Hash (SHA-2), RSA, ECC and RNG.

### 2.2.2 Software Pendukung

Hardware yang baik seperti ESP32 akan lebih *user friendly* jika dapat diprogram lebih dari satu software pendukung (*Cross Platform*). ESP32 mendukung beberapa environment pemrograman.

Beberapa environment pemrograman yang umum digunakan adalah:

- 1) Arduino IDE
- 2) PlatformIO IDE (VS Code)
- 3) LUA
- 4) MicroPython
- 5) Espressif IDF (IoT Development Framework)
- 6) JavaScript

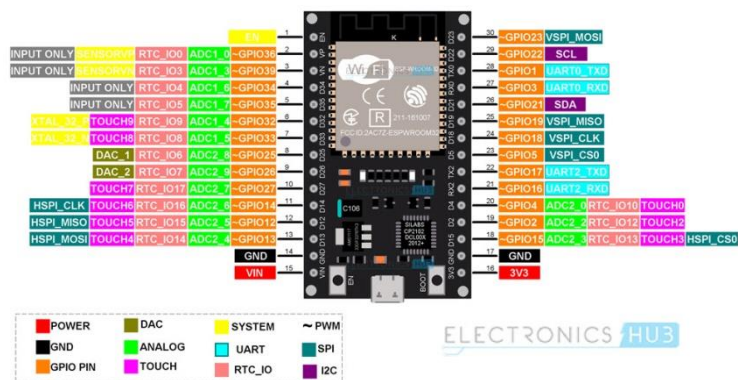
### 2.2.3 Layout

Board ESP32 memiliki 30 Pin (15 pin di setiap sisi). Ada beberapa board dengan 36 Pin dan beberapa dengan Pin yang lebih sedikit. Jadi, periksa kembali pin sebelum membuat koneksi atau bahkan menyalakan boardnya. Board ESP32 terdiri dari:

- 1) ESP-WROOM-32 Module
- 2) Two rows of IO Pins (with 15 pins on each side)
- 3) CP2012 USB – UART Bridge IC
- 4) micro–USB Connector (for power and programming)
- 5) AMS1117 3.3V Regulator IC
- 6) Enable Button (for Reset)
- 7) Boot Button (for flashing)
- 8) Power LED (Red)
- 9) User LED (Blue – connected to GPIO2)
- 10) Some passive components

Hal yang menarik tentang IC USB-to-UART adalah bahwa pin DTR dan RTS-nya digunakan untuk mengatur ESP32 secara otomatis ke mode pemrograman (bila diperlukan) dan juga mengistirahatkan board setelah pemrograman.

### 2.2.4 Pinout ESP32 Board



Gambar 2.4 pinout esp32 board

## 2.3 *Wireless Fidelity (WiFi)*

### 2.3.1 **Pengertian Wifi**

Wifi adalah sebuah media penghantar komunikasi data tanpa kabel yang bisa digunakan untuk komunikasi atau mentransfer program dan data dengan kemampuan yang sangat cepat. Wifi juga dapat diartikan sebagai teknologi yang memanfaatkan peralatan elektronik untuk bertukar data dengan menggunakan gelombang radio (nirkabel) melalui sebuah jaringan komputer termasuk koneksi.[5]

**Tabel 2.1** Spesifikasi Wifi

Spesifikasi Wifi			
Spesifikasi	Kecepatan	Frekuensi Band	cocok dengan
802.1 b	11 Mb/s	-2,4 GHz	b
802.1 1a	54 Mb/s	-5 GHz	a
802.1 1g	54 Mb/s	-2,4 GHz	b,g
802.1 1n	54 Mb/s	-2,4 GHz	b,g,n

### 2.3.2 **Fungsi Wifi**

Berikut berbagai macam kegunaan Wifi selain digunakan untuk internetan dari berbagai sumber:

#### **1. Mensinkronisasikan telepon dengan pc tanpa USB**

Wifi bisa digunakan untuk sinkronisasi hp dengan laptop/komputer. Sehingga memudahkan untuk tidak repot menggunakan kabel data ataupun kabel USB hanya dengan melalui aplikasi.

#### **2. Mentransfer foto dari kamera digital**

Kartu Eye-fi adalah kartu memori tanpa nirkabel dan saat kamera sedang dalam jangkauan jaringan tertentu, foto dan video akan tertransfer ke komputer / laptop.

### 3. Streaming film ke TV

Media server dapat streaming video melalui wifi untuk setiap HTPC lainnya. *Xbox/ Playstation* diaktifkan TV di rumah

#### 2.3.3 Mode Akses Koneksi Wifi

Dalam Wifi, terdapat 2 mode akses koneksi yaitu:

##### 1. Ad-Hoc

Mode koneksi ini adalah mode dimana beberapa komputer terhubung secara langsung, atau lebih dikenal dengan istilah *peer-to-peer*. Keuntungannya, lebih murah dan praktis bila yang terkoneksi hanya 2 atau 3 komputer, tanpa harus membeli *access point*.

##### 2. Infrastruktur

Menggunakan *access Point* yang berfungsi sebagai pengatur lalu lintas data, sehingga memungkinkan banyak *client* dapat saling terhubung melalui jaringan.

### 2.4 Arduino IDE

Arduino IDE adalah sebuah perangkat lunak pemrograman yang digunakan untuk membuat sketsa pemrograman atau dengan kata lain arduino IDE sebagai media untuk pemrograman pada papan yang ingin diprogram. Arduino IDE berfungsi untuk memasukkan program ke dalam sirkuit board modul Arduino salah satunya Modul ESP8266. IDE merupakan kependekan dari *Integrated Development Environment*, atau merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui software dibenamkan melalui sintaks pemrograman. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (*Sketch*) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *complier* Arduino dengan mikrokontroler. Arduino IDE dibuat dari bahasa pemrograman



JAVA. Arduino IDE juga dilengkapi dengan *library* C/C++ yang biasa disebut *Wiring* yang membuat operasi input dan output menjadi lebih mudah.[6]

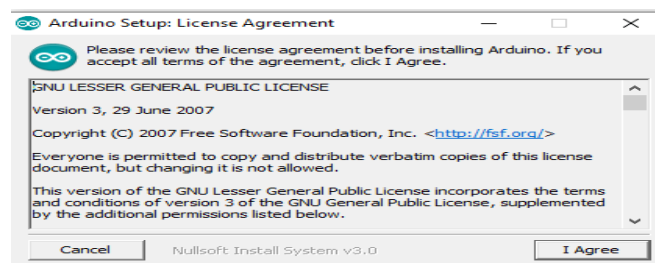
#### 2.4.1 Cara Install Software Arduino IDE

1. Cara Menginstall Arduino IDE yaitu dengan cara Download di <https://www.arduino.cc/en/Main/Software> For Windows XP and up



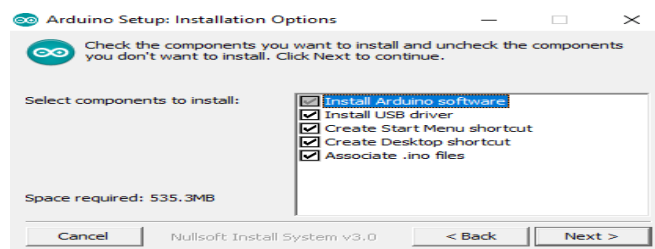
**Gambar 2.5 Download Aplikasi Arduino IDE for Windows**

2. Setelah itu akan muncul persetujuan Installasi Software Arduino seperti dibawah ini, kemudian Klik I Agree.



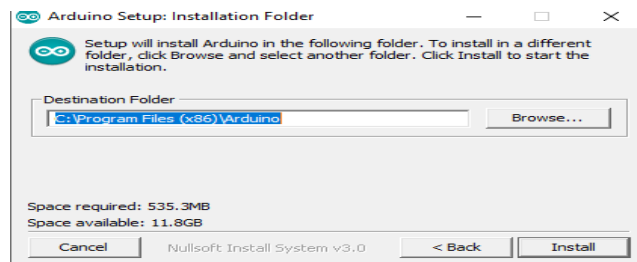
**Gambar 2.6 Persetujuan Installasi Arduino**

3. Kemudian Untuk Installation Option pilih semua option dan klik tombol Next.



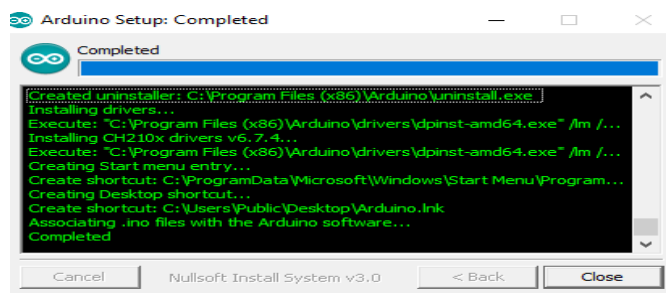
**Gambar 2.7 Opsi Instalasi Arduino IDE**

4. Tentukan folder tempat menyimpan program arduino dan klik tombol install untuk memulai proses instalasi software, setelah itu klik Install dan tunggu prosesnya hingga selesai.



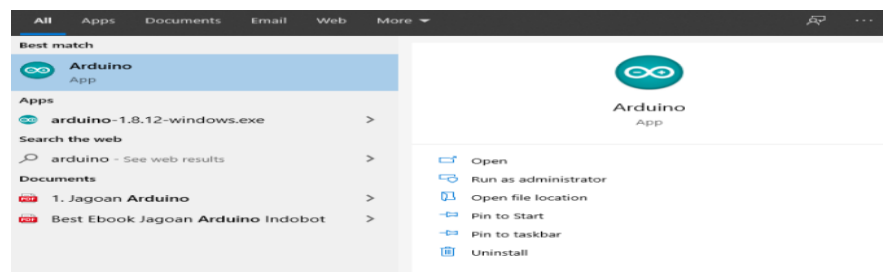
**Gambar 2.8 Pilihan instalasi folder**

5. Setelah selesai klik Close, dan software Arduini sudah mulai bisa dibuka

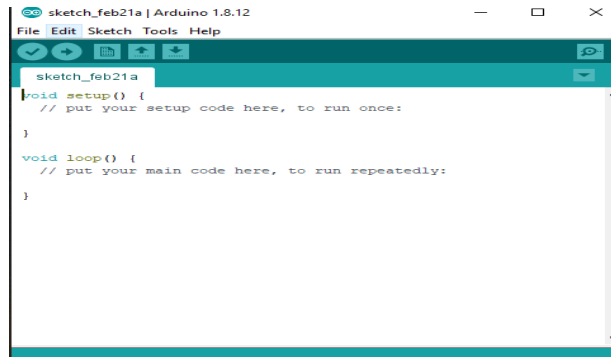


**Gambar 2.9 Proses instalasi selesai**

6. Proses instalasi Software Arduino sudah selesai terinstal di windows. Cek di Desktop Windows atau Start Menu untuk menjalankan Software IDE Arduino






**Gambar 2.10 Start Software IDE Arduino**






**Gambar 2.11** Tampilan awal Arduino IDE

*Arduino integrated Development Environment* atau *Arduino Software* (IDE) merupakan software yang digunakan sebagai media untuk memprogram di arduino, dengan kata lain *Arduino IDE* sebagai media untuk memprogram *board* Arduino. *Arduino IDE* ini berguna sebagai text editor untuk membuat, mengedit, dan juga mevalidasi kode program. bisa juga digunakan untuk meng-upload ke board Arduino. Kode program yang digunakan pada Arduino disebut dengan istilah Arduino “sketch” atau disebut juga *source code* arduino, dengan ekstensi file source code.

**Tabel 2.1** simbol di arduino IDE

Symbol	Keterangan
	Verify berfungsi untuk melakukan checking kode yang kamu buat apakah sudah sesuai dengan kaidah pemrograman yang ada atau belum
	Upload Berfungsi untuk melakukan kompilasi program atau kode yang kamu buat menjadi bahasa yang dapat dipahami oleh mesih alias si Arduino
	<b>New Sketch</b> Berfungsi membuat sketch baru

	<p>Open</p> <p>Membuka file yang sudah ada</p>
	<p>Save</p> <p>Berfungsi untuk menyimpan <i>Sketch</i> yang telah kamu buat</p>
	<p>Serial Monitor</p> <p>Berfungsi untuk membuka serial monitor. Serial monitor disini merupakan jendela yang menampilkan data apa saja yang dikirimkan atau dipertukarkan antara arduino dengan sketch pada port serialnya. Serial monitor ini dapat digunakan untuk menampilkan nilai proses, nilai pembacaan, bahkan pesan error.</p>

## 2.4.2 Pemrograman Arduino

Pemrograman arduino menggunakan struktur Bahasa C. Mekanisme pemrograman arduino sama dengan mikrokontroler pada umumnya. Mulai dari membuat sketch Program, meng-*compile*, selanjutnya proses upload pada papan arduino. Pengisian program dengan metode upload ialah mengisi papan arduino dengan program yang sudah berbentuk Hex atau hasil *compile* dari Bahasa C ke Bahasa mesin.

Program Arduino dapat dibagi dalam tiga bagian utama: Stuktur, nilai-nilai(variabel dan konstanta), dan fungsi.

### 1. Stuktur utama

#### 1) Setup()

Fungsi setup() dipanggil ketika sektsa program dimulai. Fungsi ini digunakan untuk menginisialisasi variabel, mode pin, penggunaan *library*, dll. Fungsi *setup()* hanya akan berjalan sekali, setelah power arduino dinyalakan atau saat mereset papanArduino.

## **2) Loop()**

Setelah membuat fungsi setup(), maka berikutnya adalah fungsi loop(). Fungsi loop akan melakukan loop berturut-turut dimana program akan dijalankan terus menerus secara berurutan dan loop untuk mengontrol papan Arduino.

## **3) //Komentar**

Komentar digunakan untuk memberikan keterangan pada program yang dibuat. Komentar ini tidak dieksekusi maka komentar tidak menambah ukuran file hasil compile.

## **2. Ekspresi Bilangan**

Dalam pemrograman bahasa C pada Arduino, bilangan dapat diekspresikan dalam beberapa format yaitu:

### **1. Biner**

Ditulis dengan awalan huruf '0b'. Contoh : b11110010

### **2. Desimal**

Ditulis biasa tanpa awalan. Contoh : 350

### **3. Oktal**

Ditulis dengan awalan angka '0'. Contoh : 0550

### **4. Heksadesimal**

Diawali dengan '0x'. contoh : 0x5A

## **3. Stuktur Kontrol**

Setiap program yang dibuat membutuhkan suatu kontrol. Tak hanya perulangan namun suatu eksekusi dengan syarat tertentu juga diperlukan. Pengujian kondisi sebagai berikut:

### **1) if**

Digunakan untuk mengecek suatu kondisi, jika benar maka perintah didalam if akan dieksekusi

## 2) **if - else**

Seperti dengan if, hanya saja ada 2 pilihan pernyataan / perintah. Jika kondisi benar maka perintah didalam if akan dikerjakan, jika kondisinya salah maka pernyataan di dalam else yang akan dikerjakan.

## 3) **if - else if**

Untuk melakukan pengecekan suatu lebih dari satu kondisi maka bisa menggunakan if-else if

## 4) **Switch case**

Pernyataan ini digunakan untuk memilih kondisi yang sesuai untuk kemudian mengerjakan perintahnya. Bedanya adalah kondisi yang diuji berupa sebuah nilai variabel, `switch(variabel){// variabel yang diuji`. Jika variabel memenuhi syarat dari salah satu case maka dia akan mengerjakan pernyataan/perintah tersebut. Misal nilai variabel = 2 maka dia akan mengerjakan pernyataan/perintah 2. Jika tidak memenuhi maka secara otomatis yang akan dieksekusi adalah default.

## 4. **Perulangan**

### 1) **While ()**

Perulangan ini digunakan untuk membuat perulangan yang tidak terbatas selama kondisi didalamnya terpenuhi. Program akan terus melakukan perulangan hingga nilai a mencapai angka 100

### 2) **do...while()**

Perintah ini berfungsi untuk mengeksekusi suatu pernyataan atau perintah lalu melihat kondisi di dalam while. Apabila kondisinya sesuai maka pernyataan akan dieksekusi kembali. Dan jika tidak maka pernyataan tak akan dieksekusi. Perulangan pertambahan  $a+1$  akan dilakukan sampai nilai  $a = 100$

### 3) **For ()**

Fungsi for yaitu untuk melakukan perulangan yang sifatnya terbatas. Dalam perintah for wajib disertakan nilai awal, kondisi, dan penambahan (increment) atau pengurangan (decrement)

#### 4) **goto**

Perintah ini digunakan untuk melompat secara langsung atau menuju perintah yang telah diberi label

#### 5) **return**

untuk memberikan nilai balik dari suatu fungsi

#### 6) **continue**

digunakan untuk melewati perulangan yang tersisa dari struktur *looping*. Baik itu do, for, maupun while

#### 7) **break**

digunakan untuk memaksa sebuah perulangan berhenti sebelum waktunya. Baik itu dalam perulangan for, while, maupun do..while

### 5. **Syntax**

Dalam bahasa pemrograman, syntax diartikan sebagai suatu aturan dalam penulisan kode program agar kode tersebut bisa dimengerti komputer. Beberapa sintak dalam pemrograman Arduino antara lain:

#### 1) **Titik koma ; (*semicolon*)**

Digunakan untuk mengakhiri sebuah pernyataan

#### 2) **Kurung kurawal {} (*curly braces*)**

Bagian utama dalam bahasa pemrograman C yang digunakan dalam beberapa konstruksi yang berbeda dalam beberapa fungsi.

#### 3) **#define**

Komponen C yang berguna yang memungkinkan *programmer* untuk memberi nama untuk nilai konstan sebelum program dikompilasi.

#### 4) **#include**

Digunakan untuk memasukan perpustakaan atau *library* di luar di sketsa program.

## 6. Operasi Aritmatika

**Tabel 2.3** Operasi Aritmatika

<b>Operasi Aritmatika Operator</b>	<b>Keterangan</b>
=	Pemberian Nilai
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Sisa Bagian

## 7. Operator Perbandingan

**Tabel 2.4** Operator Perbandingan

<b>Operator</b>	<b>Keterangan</b>
==	Persamaan. Jika kedua nilai yang dibandingkan sama maka hasilnya <i>'true'</i>
!=	Pertidaksamaan. Jika kedua nilai yang dibandingkan tidak sama hasilnya <i>'true'</i>
>	Lebih Besar
<	Lebih Kecil
>=	Lebih Besar atau Sama Dengan
<=	Lebih Kecil atau Sama Dengan



## 8. Operator Boolean

**Tabel 2.5** Operator Boolean

<b>Operator</b>	<b>Keterangan</b>
&&	AND
//	OR
!	NOT

## 9. Operator Bitwise

Bitwise operator = digunakan untuk operasi bit per bit pada nilai integer. Terdiri dari operator NOT,AND,OR,XORShl,Shr. Type: int atau char Bitwise operator, dari namanya sudah jelas diketahui pasti berhubungan dengan bit. Biasanya digunakan untuk memanipulasi data bertipe bit dari tipe data integer atau byte.

**Tabel 2.6** Operator Bitwise

<b>Operator</b>	<b>Keterangan</b>
<<	Geser Kiri
>>	Geser Kanan
&	AND
/	OR
^	XOR
	NOT

## 10. Operator Pertambahan dan Pengurangan

Tabel 2.7 Operator Pertambahan dan Pengurangan

Operator	Keterangan	Contoh	Keterangan
++	Pertambahan 1/ <i>increment</i>	a++	A = a + 1
--	Pengurangan 1/ <i>decrement</i>	a--	a = a - 1

## 11. Variabel

Variabel adalah suatu wadah untuk menyimpan atau menampung data. Nama variabel dibebaskan namun ada peraturan tersendiri seperti tidak boleh ada spasi, maksimal 32 karakter dan tidak boleh menggunakan istilah baku dalam bahasa C arduino karena dapat tersaji program yang error. Cara mendeklarasikan variable sebelum digunakan yaitu sebagai berikut:[tipe data][spasi][nama variabel][=][nilai int nilai\_1];

## 12. Tipe Data

Tipe data yang berbeda- beda memiliki kapasitas penyimpanan yang berbeda - beda pula. Berikut tipe data tersebut :

Tabel 2.8 Tipe-tipe Data

Tipe Data	Lebar Data	Jangkauan
Char	1 byte	-128 s/d 127
Unsigned char	1 byte	0 s/d 255
Byte	1 byte	0 s/d 255
Word	2 byte	0 s/d 65535
Int	2 byte	-32768 s/d 32767

Unsigned int	2 byte	0 s/d 65535
Long	4 byte	-2147438648 s/d 2147438647
Unsigned Long	4 byte	0 s/d 4294967295
Float	4 byte	-3.4028235E+38 s/d 3.4028235E+38

### 13. Time

#### 1) Milis()

Menghitung dengan satuan millisecond sejak papan arduino mulai menjalankan program hingga 50 hari setelah itu akan kembali ke nol begitu selanjutnya.

#### 2) Micros()

Menghitung dengan satuan microseconds sejak papan Arduino mulai menjalankan program hingga 70 menit setelah itu akan kembali ke nol.

#### 3) Delay

Jeda program untuk jumlah waktu (dalam mikrodetik). (Ada 1000 milidetik dalam satu detik)

#### 4) Delay Microseconds()

Jeda program untuk jumlah waktu (dalam mikrodetik). Ada seribu mikrodetik di milidetik, dan satu juta mikrodetik dalam detik.

### 2.5 Aplikasi Massachusetts Institute of Technology (MIT).

MIT app adalah Sistem berbasis web dimana aplikasi Android dapat digunakan tanpa perlu tahu bagaimana cara meng-code-nya. Sistem ini telah dihentikan oleh google tapi dirilis kembali oleh google sebagai proyek open-source dan saat ini dikelola oleh Massachusetts Institute of Technology (MIT). Dengan app inventor, pengguna bisa melakukan pemrograman komputer untuk menciptakan aplikasi perangkat lunak dengan sistem operasi berbasis android. App inventor ini berbasis visual block programming karena memungkinkan pengguna bisa menggunakan, melihat, menyusun dan men-drag and drops block

yang merupakan simbol perintah dan fungsi event handler untuk menciptakan sebuah aplikasi yang bisa berjalan di sistem android.



**Gambar 2.12 MIT app inventor**

MIT App Inventor merupakan platform untuk memudahkan proses pembuatan aplikasi sederhana tanpa harus mempelajari atau menggunakan bahasa pemrograman yang terlalu banyak. Kita dapat mendesain aplikasi android sesuai keinginan dengan menggunakan berbagai macam layout dan komponen yang tersedia. App Inventor memungkinkan pengguna baru untuk memprogram komputer untuk menciptakan aplikasi perangkat lunak bagi sistem operasi Android. App Inventor menggunakan antarmuka grafis, serupa dengan antarmuka pengguna pada Scratch dan StarLogo TNG, yang memungkinkan pengguna untuk men-drag-and-drop objek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. Dalam menciptakan App Inventor, Google telah melakukan riset yang berhubungan dengan komputasi edukasional dan menyelesaikan lingkungan pengembangan online Google.

### **2.5.1 Area Kerja MIT App Inventor**

Hal penting yang harus diketahui sebelum memulai menggunakan aplikasi App Inventor 2 adalah mengenal, mempelajari, dan memahami area kerja App Inventor. Secara umum area kerja app inventor terbagi atas 2

yaitu: designer yang berfungsi untuk mendesain user interface aplikasi dan area block editor yang berfungsi untuk menyusun blok-blok kode program.


### 1. Halaman Designer














Pada halaman *designer* terdapat beberapa jendela seperti *Palette*, *Viewer*, *Components*, *Media*, dan *Properties*. *Tools* tersebut berfungsi untuk mendesain tampilan aplikasi android sesuai keinginan.[7]

1. *Palette* merupakan jendela tempat mengambil komponen-komponen yang dikategorikan dalam beberapa kategori untuk dimasukkan dalam aplikasi yang dibuat. Terdapat kategori *User Interface*, *Layout*, *Media*, *Drawing and Animation*, *Maps*, *Sensors*, *Social*, *Storage*, *Connectivity*, *LEGO MINDSTORMS*, *Experimental*, dan *Extension*.
2. *Viewer* merupakan tempat untuk mengatur tampilan komponen pada aplikasi nantinya.
3. *Components* merupakan tempat untuk mengatur komponen-komponen yang telah diletakkan di *viewer*, seperti misalnya mengganti nama komponen, dan menghapus komponen.
4. *Properties* merupakan tempat untuk mengatur properti layar, dan komponen-komponen yang digunakan pada aplikasi yang sedang dibuat seperti lebar, tinggi, warna latar, besar huruf, dll.
5. *Media* merupakan tempat untuk mengunggah gambar untuk digunakan pada aplikasi yang sedang dibuat.

Pada bagian *Palette* terdapat komponen penyusun untuk mendesain aplikasi, berikut adalah komponen penyusunnya:






#### 1. User Interface

Simbol	Nama	Fungsi
	<i>Button</i>	Dapat mendeteksi ketukan, <i>hold down</i> ketika pengguna menekan tombol, atau ketika pengguna melepas tombol. Ketika <i>button</i> mendeteksi salah satu dari hal tersebut, <i>button</i> akan menjalankan perintah.







	<i>Checkbox</i>	Dapat mendeteksi ketukan dari pengguna dan mengganti state-nya menjadi <i>true/false</i> (boolean).
	<i>Date Picker</i>	Memunculkan kalender untuk memilih tanggal.
	<i>Image</i>	Memasukkan gambar dalam aplikasi.
	<i>Label</i>	Menampilkan teks dalam aplikasi.
	<i>List Picker</i>	Menampilkan <i>list</i> yang dapat dipilih oleh pengguna ketika pengguna menekan <i>list</i> .
	<i>List View</i>	Menampilkan <i>list</i> .
	<i>Notifier</i>	Memunculkan pesan/peringatan <i>pop-up</i> pada aplikasi. Pesan dapat hilang secara otomatis, atau harus menerima input dari pengguna terlebih dahulu baru menghilang.
	<i>Password Text Box</i>	Menyediakan <i>textbox</i> yang menyembunyikan teks yang dimasukkan.
	<i>Slider</i>	Menyediakan progress bar yang dapat digeser.
	<i>Spinner</i>	Menampilkan <i>pop-up list</i> dengan elemen yang dapat dipilih ketika ditekan.
	<i>Text Box</i>	Menyediakan area untuk mengetik teks.
	<i>Time Picker</i>	Memunculkan jam untuk memilih waktu.
	<i>Web Viewer</i>	Menyediakan area yang dapat menampilkan laman web.





## 2. Layout

<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
---------------	-------------	---------------




	<i>Horizontal Arrangement</i>	Menyusun komponen secara horizontal.
	<i>Horizontal Scroll Arrangement</i>	Menyusun komponen secara horizontal, namun jika lebar komponen melebihi lebar komponen ini, maka dapat digeser ( <i>scroll</i> ).
	<i>Table Arrangement</i>	Membuat tabel.
	<i>Vertical Arrangement</i>	Menyusun komponen secara vertikal.
	<i>Vertical Scroll Arrangement</i>	Menyusun komponen secara vertikal, namun jika tinggi komponen melebihi tinggi komponen ini, maka dapat digeser ( <i>scroll</i> ).

### 3. Media




<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Camcorder</i>	Mengaktifkan kamera HP dan merekam.
	<i>Camera</i>	Mengaktifkan kamera HP dan memotret.
	<i>Image Picker</i>	Menampilkan galeri pada aplikasi. Pengguna dapat memilih gambar yang akan disimpan dari galeri yang ditampilkan.
	<i>Player</i>	Memainkan musik/audio dan mengatur vibrasi HP.
	<i>Sound</i>	Memainkan musik/audio dan mengatur vibrasi HP dalam interval waktu tertentu.
	<i>Sound Recorder</i>	Merekam suara.

	<i>Speech Recognizer</i>	Mengkonversi suara menjadi teks menggunakan fitur speech recognition pada HP.
	<i>Text to Speech</i>	Mengkonversi teks menjadi suara. Pitch dan kecepatan pembacaan dapat diatur.
	<i>Video Player</i>	Memutar video serta menampilkan pengaturannya.
	<i>Yandex Translate</i>	Menerjemahkan teks dari satu bahasa ke bahasa yang lainnya.





#### 4. Drawing and Animation

<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Ball</i>	Sprite bola yang dapat berinteraksi pada sentuhan dan tarikan, serta berinteraksi dengan sprite lainnya pada kanvas.
	<i>Canvas</i>	Kanvas tempat sprite berinteraksi dengan pengguna atau dengan sprite lainnya, atau tempat pengguna menggambar.
	<i>Image Sprite</i>	Sprite gambar yang dapat berinteraksi pada sentuhan dan tarikan, serta berinteraksi dengan sprite lainnya pada kanvas.



#### 5. Maps

<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Circle</i>	Memvisualisasikan lingkaran dengan radius tertentu (dalam meter) pada koordinat tertentu. Dapat juga digunakan untuk menjalankan program ketika pengguna berada di suatu area.
	<i>Feature Collection</i>	Mengelompokkan fitur-fitur Maps. Event yang terjadi pada salah satu komponen pada kelompok tersebut, akan memicu event pada komponen-komponen lainnya.
	<i>Line String</i>	Menggambar garis pada peta.








	<i>Map</i>	Menampilkan peta pada layar aplikasi.
	<i>Marker</i>	Memberi tanda pada suatu lokasi di peta.
	<i>Polygon</i>	Membuat perimeter pada peta.
	<i>Rectangle</i>	Merupakan polygon dengan garis lintang dan garis bujur untuk batas utara, selatan, timur, dan barat. Jika sudut segiempat dipindahkan, maka informasi koordinat akan diperbaharui.



#### 6. Sensors



<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Accelerometer Sensor</i>	Mendeteksi goyangan dan percepatan dalam m/s <sup>2</sup> .
	<i>Barcode Scanner</i>	Membaca barcode.
	<i>Gyroscope Sensor</i>	Mengukur kecepatan sudut dalam derajat/detik.
	<i>Location Sensor</i>	Menyediakan informasi geografis seperti koordinat, ketinggian, kecepatan, dan alamat. Dapat juga mengkonversi alamat menjadi koordinat.
	<i>Near Field</i>	Memungkinkan aplikasi mengaktifkan fitur komunikasi jarak dekat (NFC).
	<i>Orientation</i>	Menentukan orientasi HP.
	<i>Pedometer</i>	Menentukan banyaknya langkah, dan jarak yang ditempuh.
	<i>Proximity Sensor</i>	Mengukur jarak antara objek dengan layar HP.

### 7. Social





<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	<i>Contact Picker</i>	Menampilkan list berisi kontak ketika ditekan.
	<i>Email Picker</i>	Menampilkan list berisi email ketika pengguna mengetikkan kata kunci ataupun email dari orang yang dituju.
	<i>Phone Call</i>	Melakukan panggilan telepon.
	<i>Phone Number Picker</i>	Menampilkan list berisi nomor kontak pada layar HP.
	<i>Sharing</i>	Memungkinkan pertukaran berkas atau pesan antara satu aplikasi dengan aplikasi lainnya yang terdapat pada HP.
	<i>Texting</i>	Mengirim pesan teks ke suatu nomor kontak.
	<i>Twitter</i>	Memungkinkan komunikasi dengan Twitter sehingga pengguna dapat membuat tweet, melihat tweet, dll.

### 8. Storage



<b>Simbol</b>	<b>Nama</b>	<b>Fungsi</b>
	File	Menerima atau mengirim berkas. Dapat juga digunakan untuk membaca atau menulis berkas.
	Fusiontables Control	Memungkinkan komunikasi dengan Google Fusion Table. Google Fusion Table dapat menyimpan, membagikan, query, dan memvisualisasikan tabel data.

	TinyDB	Menyimpan data untuk aplikasi sehingga ketika aplikasi ditutup, dan kemudian dibuka kembali, data masih tersimpan.
	TinyWebDB	Menyimpan data pada web.

### 9. Connectivity

Simbol	Nama	Fungsi
	<i>Activity Starter</i>	Memanggil aplikasi lain.
	<i>Bluetooth Client</i>	Membuat aplikasi sebagai Bluetooth client.
	Bluetooth Server	Membuat aplikasi sebagai Bluetooth server.
	Web	Memungkinkan aplikasi untuk berinteraksi dengan web dan menyediakan fungsi untuk HTTP GET, POST, PUT, dan DELETE.

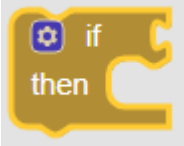
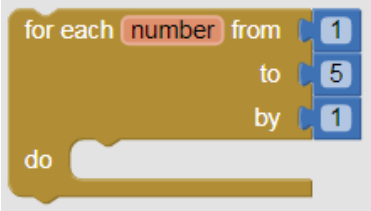
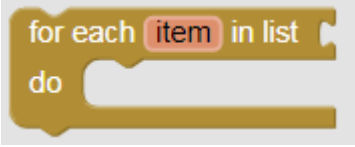
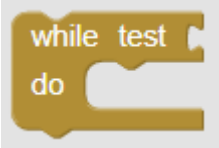
### 10. Experimental

Simbol	Nama	Fungsi
	CloudDB	Memungkinkan pengguna aplikasi untuk saling bertukar data pada database dengan software Redis.
	FirebaseDB	Memungkinkan pengguna aplikasi untuk saling bertukar data pada database dengan Firebase.



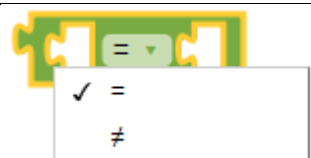
## 2. Halaman *Blocks*

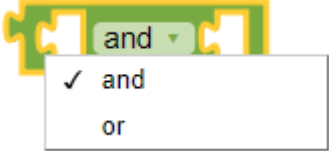
Pada halaman *blocks* terdapat beberapa *code block* yang berfungsi untuk memprogram aplikasi android sesuai yang diinginkan. Pada halaman *block* terdapat beberapa komponen seperti *Control*, *Logic*, *Math*, *Text*, *Lists*, *Colors*, *Variables*, dan *Procedures*. Berikut adalah komponen yang terdapat pada halaman *blocks*:

## 1. Control


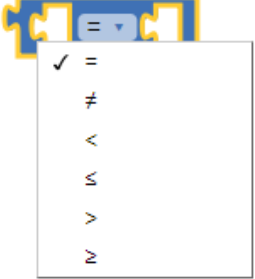
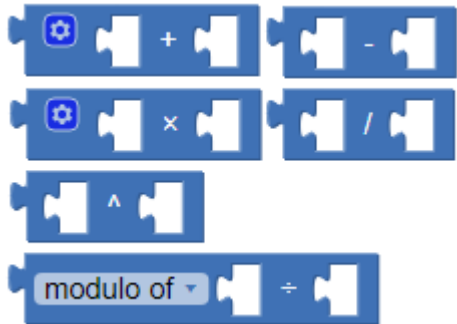

Blok Kode	Fungsi
	If kondisional. Jika “if” memenuhi syarat, maka blok yang ada setelah “then” dieksekusi.
	Looping dari angka pertama hingga angka terakhir dengan suatu interval. Maka gambar disamping berarti loop dari angka 1 hingga 5 dengan interval 1: 1, 2, 3, 4, 5.
	For bertingkat. Untuk setiap objek dalam list, dilakukan looping.
	Jika nilai test true, maka loop while berjalan.


## 2. Logic

Blok Kode	Fungsi
	Boolean true/false.
	Jika dipasangkan dengan true/false maka true menjadi false, dan false menjadi true.
	Memeriksa apakah satu objek sama dengan/tidak sama dengan objek di kanan. Jika sesuai kriteria, maka blok akan bernilai true, dan false jika tidak sesuai.


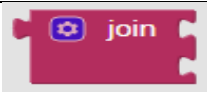
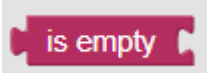
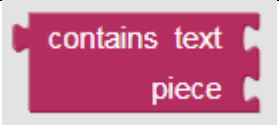
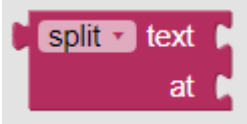
	<p>Pada logika and jika kedua syarat terpenuhi, maka nilainya menjadi true. Sedangkan pada logika or, jika salah satu syarat atau keduanya terpenuhi, maka nilainya menjadi true.</p>
---	---

### 3. Math


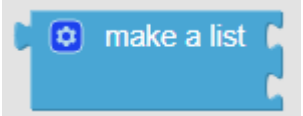
Blok Kode	Fungsi
	<p>Digunakan untuk menginput angka.</p>
	<p>Digunakan untuk membandingkan dua angka. Perbandingan dapat berupa sama dengan, tidak sama dengan, lebih dari, kurang dari, lebih dari sama dengan, dan kurang dari sama dengan.</p>
	<p>Operasi matematika dasar, yaitu tambah, kurang, kali, bagi, pangkat, dan modulus.</p>
	<p>Mengambil nilai integer secara acak dari range yang ditentukan.</p>

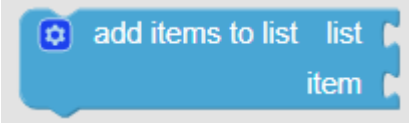
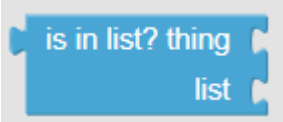
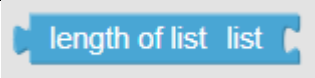

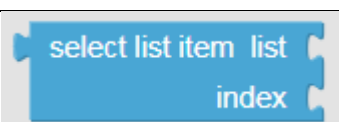
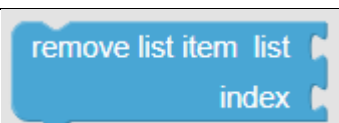
	<p>Operasi trigonometri sin, cos, tan.</p>
---	--

#### 4. Text


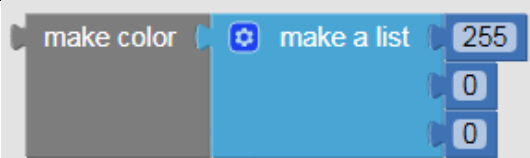
Blok Kode	Fungsi
	<p>Teks kosong.</p>
	<p>Menggabungkan dua atau lebih teks.</p>
	<p>Memeriksa jika teks kosong atau tidak.</p>
	<p>Memeriksa apakah ada bagian tertentu dalam suatu teks.</p>
	<p>Memisahkan teks pada penanda tertentu dan membuatnya menjadi list.</p>

#### 5. List

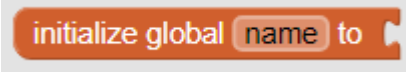

Blok Kode	Fungsi
	<p>Membuat list kosong.</p>
	<p>Membuat list.</p>

	Menambahkan elemen pada list.
	Memeriksa apakah ada elemen tertentu pada list.
	Memeriksa banyaknya elemen pada list.
	Memeriksa apakah list kosong atau tidak.
	Mengambil elemen dari suatu list.
	Menghapus elemen dari suatu list.

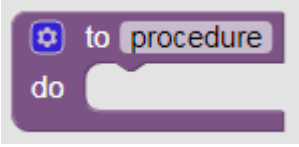
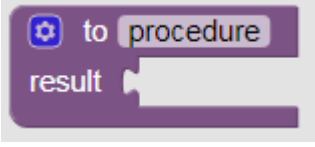
## 6. Colors

Blok Kode	Fungsi
	Pilihan warna yang disediakan MIT AI.
	Membuat warna sendiri dengan menggunakan kode RGB warna.

## 7. Variables

Blok Kode	Fungsi
	Membuat suatu variabel global.
	Mengambil variabel global.
	Memodifikasi isi dari variabel global.

## 8. Procedure

Blok Kode	Fungsi
	Membuat suatu prosedur.
	Membuat suatu fungsi yang mengembalikan hasil tertentu.

### 2.6 IFTTT ( *If This Than That* )

IFTTT adalah singkatan dari *If This Then That*. IFTTT memungkinkan anda untuk terhubung ke layanan *cloud* dan perangkat berkemampuan Internet untuk membuat beberapa tindakan otomatis yang berguna untuk membuat kehidupan digital anda lebih mudah. IFTTT ( *If This Than That* ) adalah sebuah aplikasi yang memungkinkan kamu untuk menghubungkan dua aplikasi web menjadi satu. Linden Tibbets, pembuat aplikasi ini memiliki versi untuk memungkinkan data digital seperti data fisik, dimana pengguna dapat



menggabungkan beberapa hal untuk membuat hal baru dengan mudah, kapan dan dimana saja. [8]



**Gambar 2.21 ifttt**

## **2.7 Adafruit.IO**



**Gambar 2.22 adafruit.io**

Adafruit.IO adalah layanan *cloud* atau *platform* yang berguna untuk memantau, merespon, dan berinteraksi input atau output suatu sensor dll secara real time dan gratis. Adafruit IO adalah platform IOT yang dibangun menggunakan prinsip Protokol MQTT (Message Queue Telemetry Transport). MQTT adalah protokol ringan yang memungkinkan beberapa perangkat untuk terhubung ke server. [9]

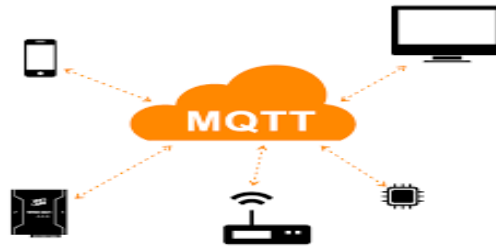
Adapun 2 layanan dan menu utama meliputi sebagai berikut:

1. Dasbor (*Dashboard*) adalah fitur yang diintegrasikan ke dalam Adafruit.IO yang memungkinkan untuk membuat grafik, mengukur, mencatat, dan menampilkan data secara *real-time*.

## 2. Feed

Umpan(*feed*) adalah inti dari Adafruit.IO dimana dalam halaman tersebut berfungsi untuk menyimpan data yang akan di unggah dan data keluaran (*output*) sensor. Misalnya tanggal dan waktu dapat diunggah atau GPS yang berfungsi untuk mengkoordinasikan dari mana data berasal. Dalam menu ini juga telah diinerasikan berbagai umpan (*feed*) dengan mengontrol privasi yang disesuaikan untuk mmbatasi privasi akses data.

### 2.13 MQTT (*Message Gueuing Telemetry Transport*)



**Gambar 2.23 mqtt**

MQTT adalah singkatan dari *Message Gueuing Telemetry Transport*. MQTT diciptakan pada tahun 1999 oleh DR.Andy Stanford-Clark dari IBM dan Arlen Nipper dari Arcom. MQTT merupakan sebuah protokol yang diterapkan pada IOT. Protokol ini sangat mendukung untuk jaringan WAN, karena WAN mencakup area yang luas. Protokol MQTT mempunyai kelebihan yaitu dapat bekerja dengan energi dan media penyimpanan yang minimum. [10]

MQTT adalah protokol konektivitas *machine-to machine* (M2M) / *Internet of Things* (IoT) yang berbasis *open source* dengan standar terbuka (OASIS) yang dirancang untuk perangkat terbatas dan *bandwith* rendah, dengan *latency* tinggi atau berjalan pada jaringan yang tidak dapat diandalkan. MQTT sangat ideal untuk perangkat yang terhubung dan aplikasi *mobile* di era M2M/IoT dimana *bandwith* dan daya baterai menjadi pertimbangan utama. Protokol MQTT menggunakan prinsip *publish subscribe*. Komponen (biasanya sensor) yang menghasilkan info tertentu dan menerbitkan info tersebut *publisher*. *Client* yang tertarik untuk mendapatkan info tertentu mendaftarkan diri

minat dari info tertentu proses ini disebut *subscribe*, *client* yang berminat disebut *subscriber*. Selain *publisher* dan *subscriber* ada juga *broker* yang menjamin subscriber mendapatkan info yang diinginkan dari *publisher*.

#### 2.14 *Android*

*Android* merupakan salah satu sistem operasi atau *operating system* berbasis *mobile* yang sangat banyak di gunakan sekarang ini. Utamanya pada telepon. pintar (smartphone) ataupun tablet. *Android* adalah sistem operasi yang dirancang oleh *Google* dengan basis kernel *Linux* untuk mendukung kinerja perangkat elektronik layar sentuh, seperti tablet atau smartphone. Jadi, *android* digunakan dengan sentuhan, gesekan ataupun ketukan pada layar *gadget* anda. *Android* bersifat *open source* atau bebas digunakan, dimodifikasi, diperbaiki dan didistribusikan oleh para pembuat ataupun pengembang perangkat lunak. Dengan sifat *open source* perusahaan teknologi bebas menggunakan OS ini diperangkatnya tanpa lisensi alias gratis.

Di awal pembuatannya, *android* ditargetkan bagi penggunaan perangkat kamera digital. Akan tetapi, para pencipta *android*, yaitu Andy Rubin, Chris White, dan Nick Sears berpendapat bahwa pasar untuk kamera digital tidak terlalu besar. Maka dari itu, sistem operasi ini kemudian dialihkan penggunaannya pada ponsel pintar<sup>[10]</sup>. Pada tahun 2004, *android* mulai dipasarkan dan berhadapan dengan saingan smartphone berbasis sistem operasi *Symbian* dan *Windows Mobile*. Di awal pemasarannya ini, Andy Rubin dan partner-nya sulit mendapatkan investor. Hingga akhirnya, android berhasil mendapatkan suntikan dana sebesar 10.000 dolar Amerika dari Steve Perlman, seseorang yang kala itu ingin membantu Andy Rubin. Di bulan Juli 2005, *Google* mengakuisisi *Android Inc.* dengan uang sebesar 50 juta dolar. Para pendiri android kemudian bergabung dengan *Google* dan memimpin proyek ini. Setelah *Google* akhirnya berkompetisi juga dalam perangkat ponsel pintar yang dibelinya, yaitu *android*, *Google* akhirnya membuat prototipe. Prototipe tersebut merupakan *smartphone* yang memiliki *keyboard*, seperti milik

*Blackberry*, Hingga Desember 2006, berita mengenai prototipe Android ini terus tersiar. Tanpa disangka-sangka, pada tahun 2007, perusahaan *Apple* merilis *iPhone* dengan desain *smartphone* yang hampir seluruh permukaannya menggunakan layar sentuh. Mulai dari situ, Google memikirkan bagaimana perkembangan *smartphone android* untuk ke depannya, mengingat prototipe awalnya menggunakan *keyboard* tanpa layar sentuh sama sekali. Untuk menyaingi *iPhone*, *Nokia* dan *Blackberry* merilis ponsel dengan layar sentuh di tahun 2008. Tak ingin kalah dengan kompetitornya, *Google* juga merilis ponsel dengan layar sentuh, yaitu *HTC Dream* atau *T-Mobile G1*. Namun, tak hanya layar sentuh saja, *smartphone* ini juga tidak meninggalkan penggunaan *keyboard*. [11]

#### **A. Kelebihan Android**

##### 1. Merupakan Sistem Operasi *Open Source*

Siapa saja bisa menggunakannya secara gratis. Para *developer* atau pengembang dimudahkan untuk mengoptimalkan dan mengembangkan OS ini untuk *smartphone* yang dibuatnya.

##### 2. Harganya Beragam

Ada yang terbilang cukup terjangkau, ada pula yang memiliki harga jual tinggi. Sehingga, *smartphone android* bisa menjangkau semua kalangan. Namun, semakin tinggi harga, semakin mumpuni pula spesifikasinya.

##### 3. Memiliki Banyak Dukungan Aplikasi

Hal ini juga tidak lepas dari sifat android yang merupakan sistem operasi *Open Source*. Pengembang pun diizinkan untuk mengembangkan aplikasi berbasis *source code* dan *android*. Oleh karena itu, jika anda masuk ke *Play Store*, akan ditemukan banyak sekali ribuan aplikasi yang sesuai dengan kebutuhan pengguna.

##### 4. Mudah dimodifikasi

Banyak komponen yang bisa anda atur ulang atau dimodifikasi, mulai dari ROM hingga *custom overclock* pada sistem operasi. Hal ini bisa berpengaruh terhadap performa ponsel pintar berbasis *android* agar bisa bekerja lebih cepat dan sesuai dengan keinginan.

## **B. Kekurangan Android**

### 1. Kerja sistemnya cukup berat

Hal ini menyebabkan banyak memori yang dibutuhkan, baik RAM maupun ROM. Bagi *smartphone* yang memiliki RAM dan ROM berkapasitas kecil, tentu ini akan menghambat performanya.

### 2. Hasil modifikasi sering menyebabkan sistem bekerja tidak stabil dan kurang optimal

Adakalanya hasil modifikasi mengakibatkan OS menjadi sedikit lelet dan kurang responsif nantinya, bisa berpengaruh pada *hardware* sehingga menjadi cepat panas dan kapasitas memori lebih mudah bocor.

### 3. Kurang responsif jika disandingkan dengan spesifikasi *hardware* yang tidak baik

Hal ini terjadi ketika kapasitas penyimpanan RAM atau kecepatan *processor* yang digunakan rendah.