

BAB II

TINJAUAN PUSTAKA

2.1 Perbandingan Penelitian

Tabel 2.1 Perbandingan Penelitian Sebelumnya

NO	Judul Referensi	Nama Peneliti dan Tahun	Keunggulan	Perbandingan
1.	Perancangan Alat Pengaduk Dodol Semi Otomatis [10].	Ajie Pangestu Sukirno /2018	-Operator tidak memerlukan banyak tenaga. -Posisi tidak membungkuk. - Rangka yang sudah berbahan dasar plat besi.	- Belum menggunakan teknologi IoT.
2.	Rancang Bangun Mesin Pengaduk Adonan Donat [11].	Adi Santoso Pribadi /2019	-Dapat merancang serta mewujudkan mesin pengaduk adonan yang sederhana untuk industri kecil. - Mengetahui kebutuhan mesin. Mempercepat produksi donat pada industri kecil.	- Belum menggunakan teknologi IoT.

3.	Rancang Bangun Alat Pengaduk Adonan Bubur Organik Kapasitas 7 liter Untuk Industri UMKM [12].	Ahmad Yunus Nasution /2018	- Dapat merancang serta mewujudkan mesin pengaduk adonan yang sederhana untuk UMKM kecil maupun besar.	- Belum menggunakan teknologi IoT.
4.	Pengaduk adonan Otomatis Abon Lele Berbasis <i>Internet of Things (IoT)</i> .	Muhammad Dirgantara JR & Yulian Ibnu Maliq /2022	- Operator tidak memerlukan banyak tenaga. -sudah berbasis IOT. -Rangka yang sudah berbahan dasar plat besi. -Serta menciptakan alat sederhana untuk UMKM yang akan memproduksi makanan seperti ini.	- Sudah memiliki teknologi IoT.

2.2 Perangkat Lunak (*Software*)

Menurut Roger S. Pressman Perangkat lunak atau *software* adalah sebuah perintah program dalam computer, yang jika diberi perintah oleh pengguna maka akan memberikan fungsi unjuk kerja seperti yang diinginkan oleh penggunanya. Pernyataan ini menunjukkan bahwa perangkat lunak atau *software* ini berfungsi untuk memberi perintah pada komputer, agar komputer berfungsi secara optimal.

Software dibuat menggunakan bahasa pemrograman yang diciptakan oleh programmer yang kemudian dikompilasi menggunakan aplikasi kompilator sehingga menjadi sebuah kode yang dapat dikenali oleh mesin *hardware* [5].

2.2.1 Fungsi Perangkat Lunak (*Software*)

Fungsi-fungsi dari *software* antara lain:

1. *Software* menyediakan fungsi dasar sebagai kebutuhan komputer yang dapat dibagi menjadi sistem operasi atau sistem pendukung.
2. *Software* berfungsi dalam mengatur berbagai *hardware* untuk bekerja secara bersama-sama.
3. Sebagai penerjemah dalam setiap intruksi-intruksi ke dalam bahasa mesin agar dapat diterima oleh *hardware* Untuk mengidentifikasi program.

2.2.2 Pembagian Perangkat Lunak (*Software*)

Software dapat dibedakan menjadi beberapa bagian, antara lain sebagai berikut:

1. Sistem Operasi, adalah perangkat lunak yang mengorganisasikan semua komponen mesin computer. Contoh-contoh sistem operasi:
 1. Windows
 2. Linux
 3. Mac OS
2. Program Aplikasi, adalah suatu program yang ditulis dalam bahasa pemrograman tertentu untuk diterapkan pada bidangnya. Program aplikasi dibedakan dalam beberapa jenis aplikasi antara lain:
 1. Pengolah kata (*word processor*)
 2. Pengolah angka (*Spread Sheet*)
 3. Pengolah data (*database*)
3. Program Bantu (*Utility*), adalah suatu program yang berfungsi untuk membantu sistem operasi. Contoh-contoh program bantu yaitu:
 1. Perangkat internet (WEB)
 2. Anti Virus
 3. PC Tools
 4. Windows Media Player (WMP)
4. Bahasa Pemrograman, adalah suatu program yang berbentuk *assembler compiler* atau *interpreter*. Contoh-contoh bahasa

pemograman adalah sebagai berikut:

1. Bahasa Pemograman HTML (*Hyper Text Markup Language*)
2. PHP
3. ASP (*Active Server Pages*)
4. XML (*Extensible Markup Language*)
5. WML (*Wireless Markup Language*)
6. PERL
7. CFM
8. *Javascript*
9. CSS (*Cascading Style Sheets*)

2.3 *Internet of Things (IoT)*



Gambar 2.1 Ilustrasi dari penggunaan IoT ^[1]

Internet Of Things atau sering disebut IoT adalah sebuah gagasan dimana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung. Misalnya CCTV yang terpasang di sepanjang jalan dihubungkan dengan koneksi internet dan disatukan di rung kontrol yang jaraknya mungkin puluhan kilometer. Atau sebuah rumah cerdas yang dapat dimanage lewat smartphone dengan bantuan koneksi internet. Pada dasarnya perangkat IoT terdiri dari sensor sebagai media pengumpul data,sambungan internet sebagai media komuniikasi dan server sebagai pengumpul informasi yang diterima sensor [1].

2.3.1 Teknologi *Internet of Things* (IoT)

Lima teknologi IoT yang banyak digunakan dalam penyebaran produk dan layanan berbasis IoT yang sukses Antara lain:

1. *Radio Frequency Identification* (RFID) / Identifikasi Frekuensi Radio

Radio Frequency Identification (RFID) memungkinkan identifikasi otomatis dan pengambilan data menggunakan radio gelombang, tag, dan pembaca. Tag dapat menyimpan lebih banyak data daripada barcode tradisional. Tag tersebut berisi data berupa *Electronic Product Code* (EPC), sistem identifikasi barang berbasis RFID global yang dikembangkan oleh *Auto-ID* Center. Tag aktif dapat berisi sensor eksternal untuk memantau suhu, tekanan, bahan kimia, dan kondisi lainnya [13].

Tag RFID aktif digunakan di pabrik, laboratorium rumah sakit, dan manajemen aset TI penginderaan jauh. Semi-pasif Tag RFID menggunakan baterai untuk memberi daya pada microchip saat berkomunikasi dengan menarik daya dari pembaca. Tag RFID aktif dan semi-pasif lebih mahal daripada tag pasif [13].

2. *Wireless Sensor Networks* (WSN) / Jaringan Sensor Nirkabel

Jaringan sensor nirkabel (WSN) terdiri dari perangkat yang dilengkapi sensor otonom yang didistribusikan secara spasial untuk memantau kondisi fisik atau lingkungan dan dapat bekerja sama dengan sistem RFID untuk melacak dengan lebih baik status hal-hal seperti lokasi, suhu, dan pergerakannya. WSN memungkinkan topologi jaringan dan komunikasi multihop yang berbeda. Kemajuan teknologi terkini dalam sirkuit terpadu berdaya rendah dan komunikasi nirkabel telah menyediakan perangkat mini berdaya rendah yang efisien, murah, untuk digunakan dalam aplikasi WSN [14].

3. *Middleware* / Perangkat Tengah

Middleware adalah lapisan perangkat lunak yang disisipkan di antara aplikasi perangkat lunak untuk memudahkan pengembang perangkat lunak melakukan komunikasi dan *input/output*. Fitur menyembunyikan detail

berbagai teknologi sangat penting untuk membebaskan pengembang IoT dari layanan perangkat lunak yang tidak relevan secara langsung ke aplikasi IoT tertentu. *Middleware* mendapatkan popularitas pada 1980-an karena peran utamanya dalam menyederhanakan integrasi teknologi lama menjadi yang baru. Ini juga memfasilitasi pengembangan layanan baru di lingkungan komputasi terdistribusi. Infrastruktur IoT terdistribusi yang kompleks dengan banyak perangkat heterogen memerlukan penyederhanaan pengembangan aplikasi dan layanan baru, sehingga penggunaan *middleware* sangat cocok dengan pengembangan aplikasi IoT [15].

Misalnya, *Global Sensor Networks (GSN)* adalah *platform middleware* sensor *open source* yang memungkinkan pengembangan dan penyebaran layanan sensor dengan hampir nol upaya pemrograman. Sebagian besar arsitektur *middleware* untuk IoT mengikuti pendekatan berorientasi layanan untuk mendukung topologi jaringan yang tidak dikenal dan dinamis.

4. *Cloud Computing* / Komputasi Awan

Komputasi awan adalah model untuk akses sesuai permintaan ke kumpulan sumber daya yang dapat dikonfigurasi bersama (misalnya, komputer, jaringan, server, penyimpanan, aplikasi, layanan, perangkat lunak) yang dapat disediakan sebagai *Infrastructure as a Service (IaaS)* / Infrastruktur sebagai Layanan atau *Software as a Service (SaaS)* / Perangkat Lunak sebagai Layanan. Salah satu hasil terpenting dari IoT adalah sejumlah data besar yang dihasilkan dari perangkat yang terhubung ke Internet. Banyak aplikasi IoT memerlukan penyimpanan data yang sangat besar, kecepatan pemrosesan yang sangat besar untuk memungkinkan pengambilan keputusan waktu nyata, dan jaringan *broadband* berkecepatan tinggi untuk mengalirkan data, audio, atau video. Komputasi awan memberikan solusi *back-end* yang ideal untuk menangani aliran data yang besar dan memprosesnya untuk jumlah perangkat IoT dan manusia yang belum pernah ada sebelumnya secara *real time* [16].

5. IoT Applications / Aplikasi IoT

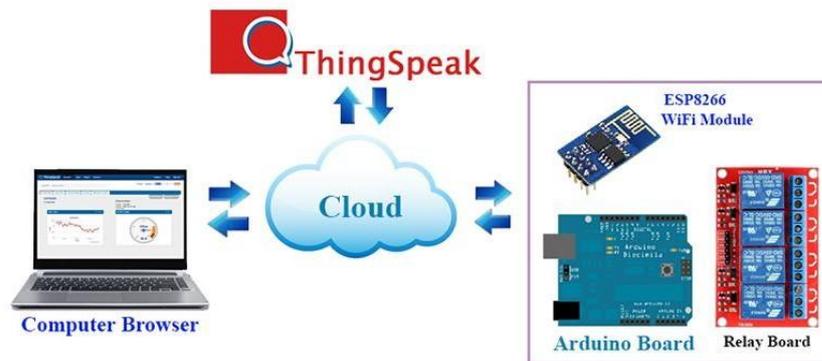
IoT memfasilitasi pengembangan berbagai aplikasi IoT yang berorientasi industri dan spesifik pengguna. Sementara perangkat dan jaringan menyediakan konektivitas fisik, aplikasi IoT memungkinkan interaksi perangkat-ke-perangkat dan manusia-ke-perangkat dengan cara yang andal dan kuat. Aplikasi IoT pada perangkat perlu memastikan bahwa data/pesan telah diterima dan ditindaklanjuti dengan benar pada waktu yang tepat [1].

Misalnya, aplikasi transportasi dan logistik memantau status barang yang diangkut seperti buah-buahan, produk segar, daging, dan produk susu. Selama transportasi, status konservasi (misalnya, suhu, kelembaban, guncangan) dipantau secara konstan dan tindakan yang tepat diambil secara otomatis untuk menghindari kerusakan saat sambungan berada di luar jangkauan. Misalnya, FedEx menggunakan *SenseAware* untuk mengawasi suhu, lokasi, dan tanda-tanda vital lain dari sebuah paket, termasuk saat dibuka dan apakah paket tersebut dirusak di sepanjang jalan [1].

2.4 *ThingSpeak* Server

ThingSpeak adalah sebuah wadah *open source* berbentuk *website* yang menyediakan layanan untuk kebutuhan IoT (*Internet of Things*) dan dapat menyimpan dan menerima data menggunakan protokol HTTP melalui Internet. *ThingSpeak* dapat digunakan untuk pengaplikasian *sensor logging*, *location tracking*, dan lain-lain. Dalam arti lain *ThingSpeak* merupakan sebuah *platform* IoT yang mampu digunakan untuk mengumpulkan, menyimpan, menganalisa, memvisualisasikan, dan bertindak sesuai data dari sensor atau aktuator, seperti Arduino, *Raspberry*, dan perangkat keras lainnya. Elemen utama pada aktivitas *ThingSpeak* adalah *channel* yang berisi *data fields*, *location fields*, dan *status field* [7].

ThingSpeak secara original diluncurkan pada tahun 2010 oleh ioBridge sebagai sebuah layanan untuk mendukung pengaplikasian IoT [7]. *ThingSpeak* dapat digunakan secara gratis namun dengan beberapa batasan yang diberikan, yaitu hanya dapat menerima data setiap 12 detik sekali.



Gambar 2.2 *ThingSpeak* Sebagai Cloud Server^[7]

2.5 Arduino IDE (*Integrated Development Environment*)



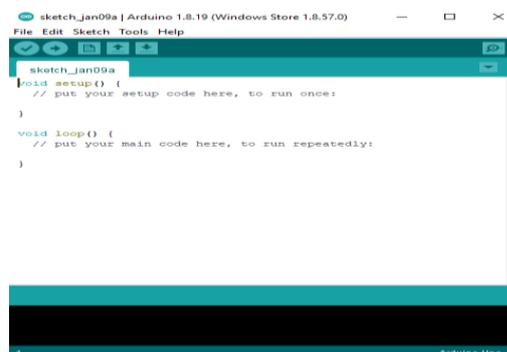
Gambar 2.3 Logo Arduino IDE^[2]

Arduino IDE (*Integrated Development Environment*) adalah *software* open source yang dikembangkan oleh Arduino untuk memrogram Arduino. Arduino IDE ini berguna sebagai *text* editor untuk membuat, mengedit, dan juga memvalidasi 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*. *ino*. 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 *compiler* Arduino dengan mikrokontroler Arduino IDE dibuat dari bahasa pemrograman *JAVA* [2].

Arduino IDE juga dilengkapi dengan *library* C/C++ yang biasa disebut Wiring yang membuat operasi input dan output menjadi lebih mudah. Arduino IDE ini dikembangkan dari *software Processing* yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino. Arduino IDE terdiri dari:

1. Editor program, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa C++ yang disederhanakan, yang merupakan turunan dari proyek *open source* Wiring [2].
2. Compiler, sebuah modul yang mengubah kode program menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa C++. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini [2].
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memori di dalam papan Arduino [2].



Gambar 2.4 Tampilan Awal *Software* Arduino IDE [2]

2.6 Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka.

Android merupakan generasi baru *platform mobile*, *platform* yang memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkannya. Sistem operasi yang mendasari Android dilisensikan dibawah GNU, *General Public Lisensi* Versi 2 (GPLv2), yang sering dikenal dengan istilah “*copyleft*” lisensi di mana setiap perbaikan pihak ketiga harus terus jatuh di bawah terms. Komersialisasi pengembang (produsen *handset* khususnya) dapat memilih untuk meningkatkan *platform* tanpa harus memberikan perbaikan mereka ke masyarakat *open source*. Sebaliknya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lisensi apapun yang mereka inginkan. Pengembang aplikasi Android diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apapun yang mereka inginkan [3].

Berikut ini adalah daftar urutan versi android mulai dari terlama sampai dengan versi terbaru. Tercatat hingga saat ini android memiliki 12 versi yang dirilis secara resmi (versi 1.0 – versi 12) dengan 19 kode penamaan (Android Astro – Android 12).

Tabel 2.2 Versi-versi Android [3]

VERSI	PENAMAAN ANDROID	TANGGAL LIRIS RESI
1.0	Astro Boy	23 September 2008
1.1	Bender	9 Februari 2009
1.5	Cupcake	27 April 2009
1.6	Donut	15 September 2009
2.0-2.1	Éclair	26 Oktober 2009
2.2	Froyo (Frozen Yoghurt)	20 Mei 2010
2.3	Gingerbread	6 Desember 2010

3.0-3.2	Honeycomb	22 Februari 2011
4.0	Ice Cream Sandwich	19 Oktober 2011
4.1-4.3	Jelly Bean	9 Juli 2012
4.4	KitKat	31 Oktober 2013
5.0-5.1	Lollipop	12 November 2014
6.0	Marshmallow	5 Oktober 2015
7.0-7.1	Nougat	22 Agustus 2016
8.0-8.1	Oreo	21 Agustus 2017
9	Pie	6 Agustus 2018
10	Android 10	3 September 2019
11	Android 11	9 September 2020
12	Android 12	18 Mei 2021

2.7 App Inventor



Gambar 2.5 Logo App Inventor ^[6]

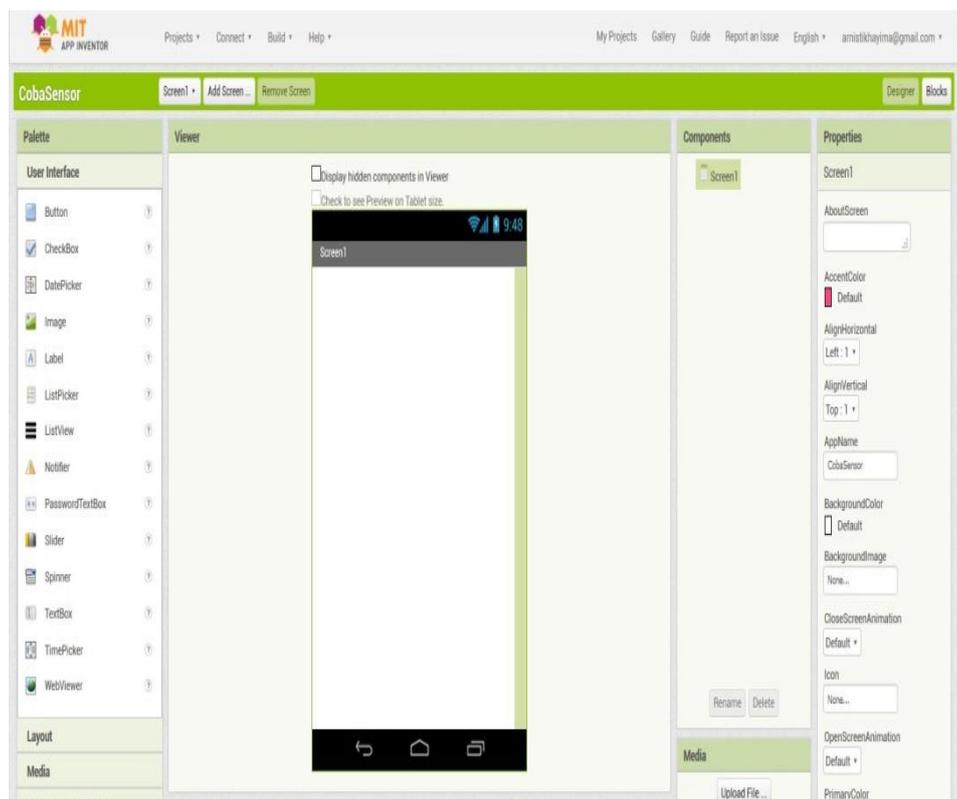
App Inventor adalah sebuah aplikasi builder untuk membuat aplikasi yang berjalan di sistem operasi Android yang disediakan oleh google *labs* dan untuk membuat Aplikasi android haruslah bermodalkan koneksi internet dan browser. Proyek yang dibuat akan di simpan secara online yang membantu kita untuk mengerjakan proyek ini secara bertahap. Meskipun App Inventor ini terbatas tapi banyak banget aplikasi yang powerfull , misall game dengan Grafis 2D dan Animasi ,utility,Aplikasi Multimedia,GPS,Navigasi dan masih banyak lagi [6].

Disini jika ingin menjalankan program App Inventor harus punya account google dulu untuk bisa masuk ke home app inventor. App Inventor ini menggunakan teknik visual programming,

Berbentuk seperti susunan puzzle-puzzle yang memiliki logika tertentu. Kalian bisa baca-baca tutorial dan contoh-contoh aplikasinya di page Explore [6].

2.7.1 Tampilan MIT App Inventor

Pada MIT App Inventor terdapat dua halaman utama, yaitu halaman designer dan halaman blocks. Halaman designer digunakan untuk mendesain tampilan aplikasi dengan berbagai komponen dan layout yang disediakan sesuai dengan keinginan. Sedangkan halaman blocks digunakan untuk memprogram jalannya aplikasi android sesuai dengan tujuan [6].



Gambar 2.6 Tampilan Halaman Designer [6]



Gambar 2.7 Tampilan Halaman Blocks ^[6]

2.7.2 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 [6].

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.

- 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

Tabel 2.3 User Interface [6]

Simbol	Nama	Fungsi
	<i>Button</i>	Dapat mendeteksi ketukan, hold down ketika pengguna menekan tombol, atau ketika pengguna melepas tombol. Ketika button mendeteksi salah satu dari hal tersebut, button akan menjalankan perintah.
	<i>Checkbox</i>	Dapat mendeteksi ketukan dari pengguna dan mengganti state-nya menjadi true/false (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 list yang dapat dipilih oleh pengguna ketika pengguna menekan list.
	<i>List View</i>	Menampilkan list.
	<i>Notifier</i>	Memunculkan pesan/peringatan pop-up pada aplikasi. Pesan dapat hilang secara otomatis, atau harus menerima input dari pengguna terlebih dahulu baru menghilang.
	<i>Password Text Box</i>	Menyediakan textbox yang menyembunyikan teks yang dimasukkan.
	<i>Slider</i>	Menyediakan progress bar yang dapat digeser.

	<i>Spinner</i>	Menampilkan pop-up list 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

Tabel 2.4 Layout [6]

Simbol	Nama	Fungsi
	<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 (scroll).
	<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 (scroll).

3. Media

Tabel 2.5 Media [6]

Simbol	Nama	Fungsi
	<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. Sosial

Tabel 2.6 Sosial [6]

Simbol	Nama	Fungsi
	<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.
	Twitter	Memungkinkan komunikasi dengan Twitter sehingga pengguna dapat membuat tweet, melihat tweet, dll.

5. Storage

Tabel 2.7 Storage [6]

Simbol	Nama	Fungsi
	<i>File</i>	Menerima atau mengirim berkas. Dapat juga digunakan untuk membaca atau menulis berkas.
	<i>Fusion Tables Control</i>	Memungkinkan komunikasi dengan Google Fusion Table. Google Fusion Table dapat menyimpan, membagikan, query, dan memvisualisasikan tabel data.
	<i>TinyDB</i>	Menyimpan data untuk aplikasi sehingga ketika aplikasi ditutup, kemudian dibuka kembali, data masih tersimpan.
	<i>TinyWeb DB</i>	Menyimpan data pada web.

6. Connectivity

Tabel 2.8 Connectivity [6]

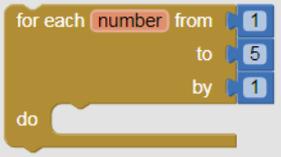
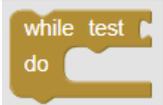
Simbol	Nama	Fungsi
	<i>Activity Starter</i>	Memanggil aplikasi lain.
	Bluetooth <i>Client</i>	Membuat aplikasi sebagai Bluetooth <i>client</i> .
	Bluetooth <i>Server</i>	Membuat aplikasi sebagai Bluetooth <i>server</i> .
	Web	Memungkinkan aplikasi untuk berinteraksi dengan web dan menyediakan fungsi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

2.7.3 Halaman Block

Pada halaman blocks terdapat beberapa codeblock 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

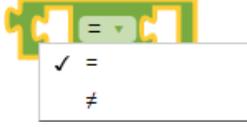
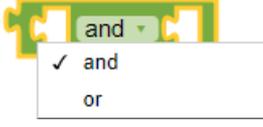
Tabel 2.9 Control [6]

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

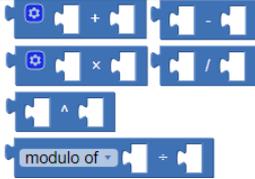
Tabel 2.10 Logic [6]

Blok kode	Fungsi
	Boolean true/false.
	Jika dipasangkan dengan true/false maka true menjadi false, dan false menjadi true.

	<p>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>
	<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>

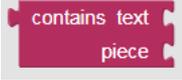
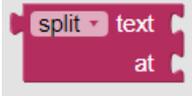
3. Math

Tabel 2.11 Math [6]

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>

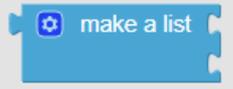
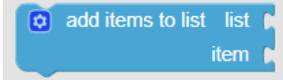
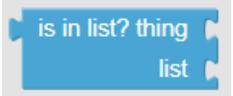
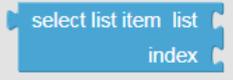
4. Text

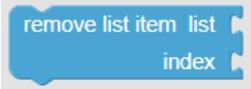
Tabel 2.12 Text [6]

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

5. List

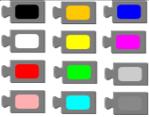
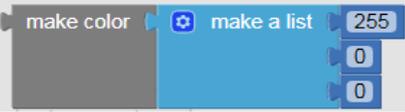
Tabel 2.13 List [6]

Blok Kode	Fungsi
	Membuat list kosong.
	Membuat list.
	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.
---	-----------------------------------

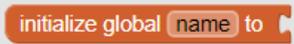
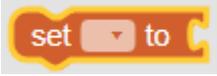
6. Colors

Tabel 2.14 Colors [6]

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

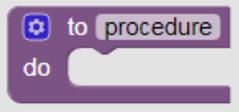
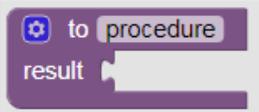
7. Variables

Tabel 2.15 Variables [6]

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

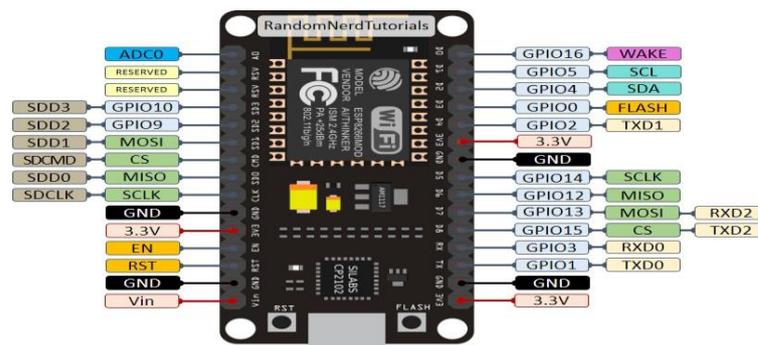
8. Procedure

Tabel 2.16 Procedure [6]

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

2.8 NodeMCU ESP8266

ESP8266 adalah Wifi Serial *Transceiver Module* sebuah komponen chip terintegrasi yang didesain untuk keperluan dunia masa kini yang serba tersambung. Chip ini menawarkan solusi networking Wi-Fi yang lengkap dan menyatu, yang dapat digunakan sebagai penyedia aplikasi atau untuk memisahkan semua fungsi networking Wi-Fi ke pemroses aplikasi lainnya.



Gambar 2.8 NodeMCU ESP8266 [4]

ESP8266 memiliki kemampuan *on-board* prosesing dan penyimpanan yang memungkinkan chip tersebut untuk diintegrasikan dengan sensor-sensor atau dengan aplikasi alat tertentu melalui pin input output hanya dengan pemrograman singkat. Dengan level yang tinggi berupa on-chip yang terintegrasi memungkinkan external sirkuit yang ramping, termasuk modul sisi depan, didesain untuk menempati area PCB yang kecil [4].

Pengembangan kit ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC, 1-Wire dan ADC (Analog to Digital Converter) semua dalam satu board. GPIO NodeMCU ESP8266 seperti Gambar 2.7.

NodeMCU berukuran panjang 4.83cm, lebar 2.54cm, dan berat 7 gram. Board ini sudah dilengkapi dengan fitur WiFi dan Firmwarena yang bersifat *opensource*.

Spesifikasi yang dimiliki oleh NodeMCU sebagai berikut :

1. Board ini berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan onboard USB to TTL. Wireless yang digunakan adalah IEEE 802.11b/g/n.
2. 2 tantalum kapasitor 100 micro farad dan 10 micro farad.
3. 3.3v LDO regulator.
4. Blue led sebagai indikator.
5. Cp2102 usb to UART bridge.
6. Tombol reset, port usb, dan tombol flash.
7. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX
8. 3 pin ground.
9. S3 dan S2 sebagai pin GPIO
10. S1 MOSI (Master Output Slave Input) yaitu jalur data dari master dan masuk ke dalam slave, sc cmd/sc.
11. S0 MISO (Master Input Slave Input) yaitu jalur data keluar dari slave dan masuk ke dalam master.
12. SK yang merupakan SCLK dari master ke slave yang berfungsi sebagai clock.
13. Pin Vin sebagai masukan tegangan.
14. Built in 32-bit MCU.



Gambar 2.9 GPIO NodeMCU ESP8266 v3 ^[4]

1. RST : berfungsi mereset modul
2. ADC: Analog Digital Converter. Rentang tegangan masukan 0-1v, dengan skop nilai digital 0-1024
3. EN: Chip Enable, Active High
4. IO16 :GPIO16, dapat digunakan untuk membangunkan chipset dari mode deep sleep
5. IO14 : GPIO14; HSPI_CLK
6. IO12 : GPIO12: HSPI_MISO
7. IO13: GPIO13; HSPI_MOSI; UART0_CTS
8. VCC: Catu daya 3.3V (VDD)
9. CS0 :Chip selection
10. MISO : Slave output, Main input
11. IO9 : GPIO9
12. IO10 GBIO10
13. MOSI: *Main output slave input*
14. SCLK: *Clock*
15. GND: *Ground*
16. IO15: GPIO15; MTDO; HSPICS; UART0_RTS
17. IO2 : GPIO2;UART1_TXD
18. IO0 : GPIO0
19. IO4 : GPIO4
20. IO5 : GPIO5
21. RXD : UART0_RXD; GPIO3
22. TXD : UART0_TXD; GPIO1

2.9 Kabel Junper

Kabel *jumper* adalah suatu istilah kabel yang ber-diameter kecil yang di dalam dunia elektronika digunakan untuk menghubungkan dua titik atau lebih dan dapat juga untuk menghubungkan 2 komponen elektronika [8].

2.9.1 Jenis Kabel Jumper

Ada beberapa jenis kabel *jumper* yang dibedakan berdasarkan konektor kabelnya, yaitu :

1. *Male-male*

Kabel *jumper* jenis ini digunakan untuk koneksi *male to male* pada kedua ujung kabelnya.



Gambar 2.10 Kabel *jumper male – male* ^[8]

2. *Male-female*

Kabel *jumper* jenis ini digunakan untuk koneksi *male to female* dengan salah satu ujung kabel dikoneksi *male* dan satu ujungnya lagi dengan koneksi *female*.



Gambar 2.11 Kabel *jumper male – female* ^[8]

3. Kabel Jumper Female to Female

Jenis kabel jumper yang terakhir adalah kabel female to female. Kabel ini sangat cocok untuk menghubungkan antar komponen yang memiliki header male. contohnya seperti sensor ultrasonik HC-SR04, sensor suhu DHT, dan masih banyak lagi.



Gambar 2.12 Kabel *jumper female – female* ^[8]

2.10 Module Relay

Relay adalah komponen elektronik berupa saklar elektronik yang digerakkan oleh arus listrik. Secara prinsip, *relay* merupakan tuas saklar dengan lilitan kawat pada batang besi (solenoid) di dekatnya, ketika solenoid dialiri arus listrik, tuas akan tertarik karena adanya gaya magnet yang terjadi pada solenoid sehingga kontak saklar akan menutup. Pada saat arus dihentikan, gaya magnet akan hilang, tuas akan kembali keposisi semula dan kontak saklar kembali terbuka. *Relay* biasanya digunakan untuk menggerakkan arus/tegangan yang besar (misalnya peralatan listrik 4 A/AC 220V) dengan memakai arus/tegangan yang kecil (misalnya 0.1 A/12 volt DC).

Relay adalah komponen listrik yang bekerja berdasarkan prinsip induksi medan elektromagnetis. Jika sebuah penghantar dialiri oleh arus listrik, maka disekitar penghantar tersebut timbul medan magnet. Medan magnet yang dihasilkan oleh arus listrik tersebut selanjutnya diinduksikan ke logam ferromagnetis. Penemu *relay* pertama kali adalah Joseph Henry pada tahun 1835 [9].



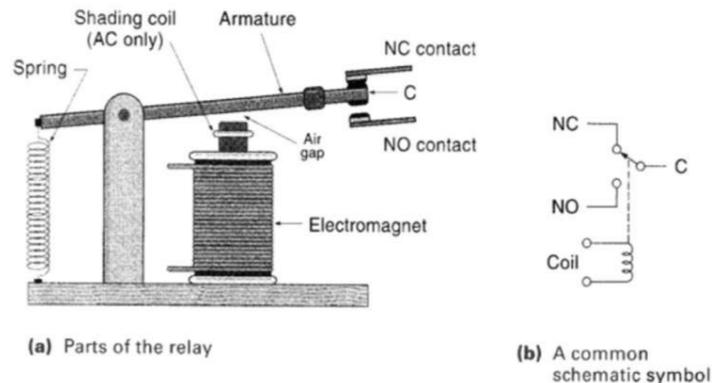
Gambar 2.13 Module Relay 5V tipe 2Channel ^[9]

2.10.1 Prinsip Kerja Module Relay

Prinsip kerja sama dengan kontraktor magnet yaitu sama-sama berdasarkan kemagnetaan yang dihasilkan oleh kumparan coil, jika kumparan coil tersebut diberi sumber listrik. Berdasarkan sumber listrik yang masuk maka relay dibagi menjadi 2 macam yaitu relay DC dan relay AC, besar tegangan DC yang masuk pada coil relay bervariasi sesuai dengan ukuran yang tertera pada body relay tersebut diantaranya relay dengan tegangan 6 Volt, 12 Volt, 24 Volt, 48 Volt, sedangkan untuk tegangan AC sebesar 220 Volt.

Relay terdiri dari coil dan contact, coil adalah gulungan kawat yang mendapat arus listrik, sedangkan contact adalah sejenis saklar yang pergerakannya tergantung dari ada tidaknya arus listrik di coil. Contact ada 2 jenis : Normally Open (kondisi awal sebelum diaktifkan *open*), dan *Normally Closed* (kondisi awal sebelum diaktifkan *close*).

Secara sederhana berikut ini prinsip kerja dari relay : ketika coil mendapat listrik (*energized*), akan timbul gaya elektromagnet yang akan menarik armature yang berpegas, dan contact akan menutup [9].



Gambar 2.14 Prinsip Kerja Relay ^[9]

Adapun spesifikasi dari module relay 2 channel, sebagai berikut :

- 1) Menggunakan tegangan rendah, 5V, sehingga dapat langsung dihubungkan pada sistem mikrokontroler.
- 2) Tipe relay adalah SPDT (Single Pole Double Throw): 1 COMMON, 1 NC (Normally Close), dan 1 NO (Normally Open).
- 3) Memiliki daya tahan sampai dengan 10A.
- 4) Pin pengendali dapat dihubungkan dengan port mikrokontroler mana saja, sehingga membuat pemrogram dapat leluasa menentukan pin mikrokontroler yang digunakan sebagai pengendali.
- 5) Dilengkapi rangkaian penggerak (*driver*) relay dengan level tegangan TTL sehingga dapat langsung dikendalikan oleh mikrokontroler.
- 6) Driver bertipe “*active high*” atau kumparan relay akan aktif saat pin pengendali diberi logika “1”.
- 7) Driver dilengkapi rangkaian peredam GGL induksi sehingga tidak akan membuat reset sistem mikrokontroler.

Connection:

1. VCC connect to 5V
2. GND connect to GND
3. 1N1-1N2 relay control interface connected MCU's IO port.