

BAB II

TINJAUAN PUSTAKA

2.1 *Internet of Things (IoT)*

2.1.1 *Pengertian Internet of Things (IoT)*

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. Dimana bertujuan memperluas manfaat dan konektivitas internet yang tersambung secara terus-menerus, berada di dunia fisik, bahan pangan, elektronik, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor tertanam dan selalu “on”.

“A *Things*” pada *Internet of Things* dapat didefinisikan sebagai subjek misalkan orang dengan monitor *implant* jantung, hewan peternakan dengan *transponder biochip*, sebuah mobil yang telah dilengkapi *built-in* sensor untuk memperingatkan pengemudi ketika tekanan ban rendah. Sejauh ini, IoT paling erat hubungannya dengan komunikasi *machine-to-machine* (M2M) di bidang manufaktur dan listrik, perminyakan, dan gas. Produk dibangun dengan kemampuan komunikasi M2M yang sering disebut dengan sistem cerdas atau “*smart*”. Sebagai contoh yaitu *smart* kabel, *smart* meter, *smart grid sensor*.

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)*, antara lain (id.wikipedia.org):

Ide awal *Internet of Things* pertama kali dimunculkan oleh Kevin Ashton pada tahun 1999 di salah satu presentasinya. Kini banyak perusahaan besar mulai mendalami *Internet of Things* sebut saja Intel, Microsoft, Oracle, dan banyak lainnya.

Banyak yang memprediksi bahwa *Internet of Things* adalah “*the next big thing*” di dunia teknologi informasi, hal ini karena *Internet of Things* menawarkan banyak potensi yang bisa digali.

Menurut Ashton definisi awal IoT adalah sebagai sebuah teknologi yang memungkinkan adanya pengendalian, komunikasi, kerjasama dengan berbagai pihak perangkat keras, berbagai data, memvirtualisasikan segala hal nyata ke dalam bentuk internet, dan lain-lain melalui jaringan internet. 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”

Menurut Casagras (*Coordinator and support action for global RFID-related activities and standardisation*) mendefinisikan IoT 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 otonom data *capture* yang tinggi, *event transfer*, konektivitas jaringan dan interoperabilitas.

2.1.2 Metode *Internet of Things* (IoT)

Metode yang digunakan oleh *Internet of Things* adalah nirkabel atau pengendalian secara otomatis tanpa mengenal jarak. Pengimplementasian *Internet*

of Things sendiri biasanya selalu mengikuti keinginan si *developer* dalam mengembangkan sebuah aplikasi yang ia ciptakan, apabila aplikasinya itu diciptakan guna membantu monitoring sebuah ruangan maka pengimplementasian *Internet of Things* itu sendiri harus mengikuti alur diagram pemrograman mengenai sensor dalam sebuah rumah, berapa jauh jarak agar ruangan dapat dikontrol, dan kecepatan jaringan internet yang digunakan. Perkembangan teknologi jaringan dan Internet seperti hadirnya IPv6, 4G, dan Wimax, dapat membantu pengimplementasian *Internet of Things* menjadi lebih optimal, dan memungkinkan jarak yang dapat di lewati menjadi semakin jauh, sehingga semakin memudahkan kita dalam mengontrol sesuatu.

2.1.3 Perangkat *Internet of Things* Dengan Komunikasi

Saat ini, di belahan dunia manapun banyak ditemukan peralatan yang sudah terhubung dengan internet. Satu peralatan yang digunakan bisa menghubungkan ke perangkat lainnya hanya dengan menggunakan koneksi internet. Ada beberapa peralatan atau aplikasi yang bisa digunakan untuk menghubungkan dari satu perangkat ke perangkat lainnya. Berikut beberapa contoh perangkat:

1. *Smartphone*



Gambar 2.1 *Smartphone* ^[1]

Saat ini banyak *smartphone* yang bisa menghubungkan kita dengan dunia luar serta perangkat lain. Sebagai contoh *smartphone* yang kita

gunakan bisa mengendalikan televisi, AC dan lain sebagainya. Televisi dan AC bisa kita matikan dengan menggunakan *smartphone* yang kita punya. Dengan begitu kita tidak perlu repot-repot untuk mematikan atau menyalakannya secara manual. *Smartphone* juga bisa digunakan untuk belanja *online* jika terdapat aplikasi dalam *smartphone* tersebut dan terhubung dengan koneksi internet. Jadi kita tidak harus membuang waktu untuk pergi ke toko untuk membeli sesuatu yang kita butuhkan.

2. Ipad



Gambar 2.2 Ipad ^[2]

Sama seperti *smartphone*, *ipad* juga bisa mengendalikan beberapa peralatan. Sebagai contoh kita bisa *mendownload* lagu atau videop dengan menggunakan *ipad* yang tentu saja sudah terkoneksi internet. Kemudian kita juga bisa melakukan transaksi *online* seperti *online shop* dan sebagainya. Kita juga dapat bermain *game*, tidak hanya *offline*, *online* pun bisa asalkan terhubung dengan koneksi internet. *Ipad* juga bisa terkoneksi dengan *smartphone* agar kita bisa mentransfer data dengan mudah.

3. Laptop / Komputer



Gambar 2.3 Laptop dan Komputer ^[3]

Contoh ini masih sama dengan *smartphone* dan *ipad*, laptop / komputer bisa mengendalikan perangkat lain secara cepat. Contohnya kita bisa mengendalikan laptop atau komputer lain dari laptop yang kita punya. Kita juga bisa mengendalikan CCTV yang dipasang pada rumah atau ruangan saat kita sedang melakukan pekerjaan diluar.

2.1.4 Dampak *Internet Of Things*

Dampak besar *Internet of Things* akan terasa pada besarnya data yang dihasilkan pada perangkat tersebut. Bayangkan betapa besarnya informasi personal yang disimpan dan di analisa oleh milyaran perangkat. Itulah yang disebut dengan *big data* yang menimbulkan masalah keamanan.

Implementasi keamanan untuk *Internet of Things* terbilang mahal karena harus melalui tiga lapis: fisik, jaringan dan data. Fisik artinya sebuah perangkat yang memiliki keamanan seperti kunci, seandainya dicuri atau hilang, adapun keamanan jaringan dapat dilakukan melalui *Virtual Private Network* (VPN) atau bentuk enkripsi lainnya.

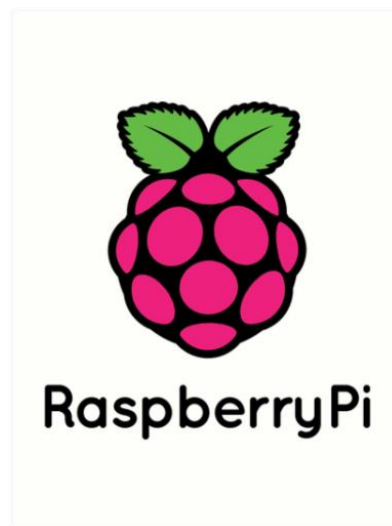
Kemanan paling rentan adalah keamanan dari data itu sendiri. Ketika tersimpan secara *local* di dalam perangkat data (*data at rest*) dan ketika data

tersebut dikirim ke perangkat lain (data on the move) harus tetap di enkripsi supaya terhindar dari peretas atau *hacker*.

Internet of Things sangat bersinggungan dengan *cloud computing*. Tentu saja dapat menyimpan semua data itu di *home storage* di rumah. Disinilah keunggulan dari *cloud computing*, karena itu para analis menilai bahwa *Internet of Things* adalah perpaduan yang sempurna.

2.2 Raspberry Pi

2.2.1 Pengenalan Raspberry Pi



Gambar 2.4 Logo RaspberryPi ^[4]

Raspberry Pi sering disingkat dengan nama Raspi, adalah komputer papan tunggal (*single-board circuit; SBC*) yang seukuran dengan kartu kredit yang dapat digunakan untuk menjalankan program perkantoran, permainan komputer, dan sebagai pemutar media hingga video beresousi tinggi. Raspberry Pi dikembangkan oleh yayasan nirlaba, Raspberry Pi *Foundation*, yang digawangi sejumlah pengembang dan ahli komputer dari Universitas Cambridge, Inggris. Ide dibalik Raspberry Pi diawali dari keinginan untuk mencetak pemrogram generasi baru. Seperti disebutkan dalam situs resmi Raspberry Pi *Foundation*, waktu itu Eben Upton, Rob Mullins, Jack Lang, dan Alan Mycroft, dari Laboratorium Komputer Universitas Cambridge memiliki kekhawatiran melihat kianturnya keahlian dan jumlah siswa yang hendak belajar ilmu komputer. Mereka lantas

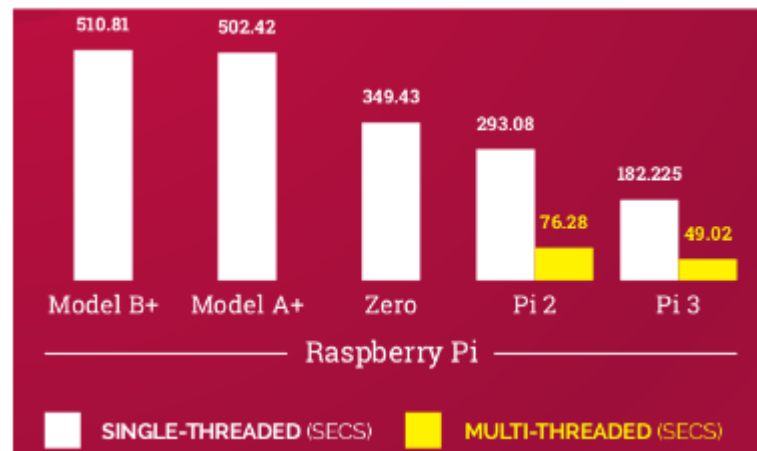
mendirikan yayasan Raspberry Pi bersama dengan Pete Lomas dan David Braben pada 2009. Tiga tahun kemudian, Raspberry Pi Model B memasuki produksi massal. Dalam peluncuran pertamanya pada akhir Februari 2012 dalam beberapa jam saja sudah terjual 100.000 unit. Pada bulan Februari 2016, Raspberry Pi *Foundation* mengumumkan bahwa mereka telah menjual 8 juta perangkat Raspi, sehingga menjadikannya sebagai perangkat paling laris di Inggris.

Sangat jelas sekali, Raspi tidak akan memiliki kekuatan atau tidak se-*powerfull* seperti *desktop* PC. Tapi karena harganya yang jauh lebih murah maka kita bisa mengoprek dan memodifikasi tanpa memikirkan *costnya*.

Raspberry Pi juga bagus dalam melakukan banyak hal yang tidak membutuhkan komputer mahal untuk membuatnya. seperti berjalan sebagai NAS (*Network Attached Storage*), *web server*, *router*, *media center*, *TorrentBox* dan masih banyak lagi.

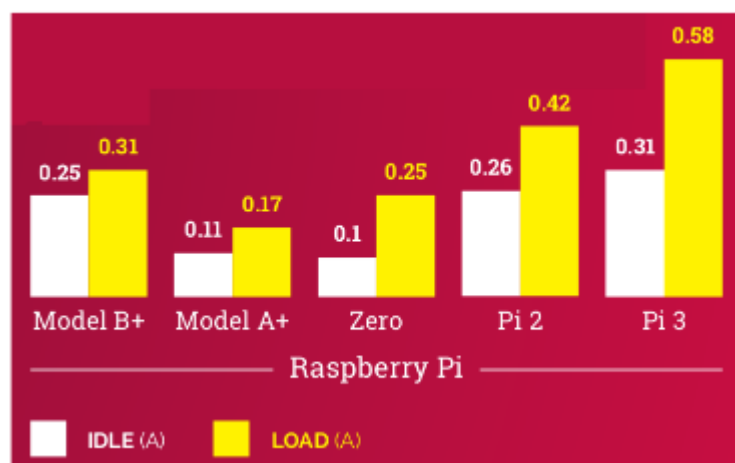
Raspberry Pi memiliki dua model yaitu model A dan model B. Secara umum Raspberry Pi Model B, 512MB RAM. Perbedaan model A dan B terletak pada *memory* yang digunakan, Model A menggunakan memory 256 MB dan model B 512 MB. Selain itu model B juga sudah dilengkapi dengan *ethernet port* (kartu jaringan) yang tidak terdapat di model A. Desain Raspberry Pi didasarkan seputar SoC (*System-on-a-chip*) *Broadcom BCM2835*, yang telah menanamkan prosesor ARM1176JZF-S dengan 700 MHz, *VideoCore IV GPU*, dan 256 *Megabyte* RAM (model B). Penyimpanan data didesain tidak untuk menggunakan *hard disk* atau *solid-state drive*, melainkan mengandalkan kartu SD (*SD memory card*) untuk *booting* dan penyimpanan jangka panjang.

Raspberry Pi terbaru menawarkan dukungan operasi *multi-threaded* dengan mengambil keunggulan dari empat *processing core* pada Pi 2 dan Pi 3. SysBench mengungkapkan pengembangan desain Raspberry Pi dari pertama hingga terakhir, menunjukkan kinerja *single-threaded* telah meningkat pesat, dan keunggulan terbesar pada saat dijalkannya program *multi-threaded*.



Gambar 2.5 *Single dan Multi-Treaded* pada Raspberry Pi ^[4]

Anda tidak bisa mendapatkan kinerja ekstra dari Raspberry Pi tanpa beberapa pengorbanan. Raspberry Pi 3 mengambil daya lebih besar dari versi sebelumnya, baik pada saat mendapat beban kerja maksimum (*load*) maupun pada saat tidak ada beban (*idle*). Jika anda mencari jenis Raspberry yang memiliki daya tahan baterai maksimum, mungkin bisa memilih Model A + Pi Zero sebagai alternatif.



Gambar 2.6 Beban pada Raspberry Pi ^[4]

Raspberry Pi 3 memiliki fitur 40-pin *General-Purpose Input-Output* (GPIO) dengan *header* yang sama seperti semua jenis Raspberry Pi Model B+ dan Model A+. Perangkat keras GPIO yang ada akan bekerja tanpa modifikasi, satu-

satunya perubahan adalah *switch* UART tergabung pada pin GPIO, namun hal ini sudah ditangani secara internal oleh sistem operasi. Pin GPIO Raspberry Pi sering digunakan bersama dengan program berbasis *Python*, meskipun hal ini dapat menyebabkan terjadinya *bottleneck* CPU. Namun pada Raspberry terbaru hal ini sudah teratasi. Dalam tes ini, program sederhana yang menggunakan *library* RPi.GPIO digunakan untuk menghitung seberapa cepat mematikan pin dengan menggunakan *counter* frekuensi.

Raspberry Pi memiliki website di www.raspberrypi.org yang memiliki cukup banyak informasi tentang produk, aksesoris sampai dengan contoh proyek pembelajaran yang dapat dikerjakan dengan Raspberry Pi. Berikut ini adalah tampilan website Raspberry Pi.



Gambar 2.7 Tampilan website resmi Raspberry Pi ^[4]

Berikut 11 Model *Comparison Chart* dari Raspberry Pi yang ada :

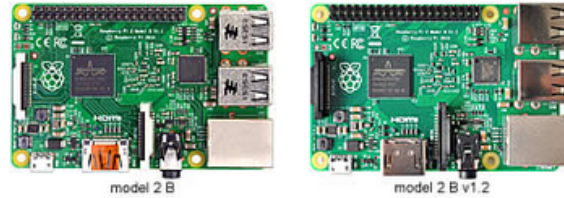
Tabel 2.1 Spesifikasi Raspberry Pi 1 Model B dan B+



Feature	Pi 1 B	Pi 1 B+
Released	Apr 2012	Jul 2014
Architecture	ARMv6Z 32-bit	ARMv6Z 32-bit
SoC	Broadcom BCM2835	Broadcom BCM2835
CPU	700 MHz ARM1176JZF-S	700 MHz ARM1176JZF-S
Cores	1	1
GPU	Broadcom VideoCore IV HD 1080p	Broadcom VideoCore IV HD 1080p
Memory RAM	512 MB	512 MB
Operating System	Primarily Linux based	Primarily Linux based
USB 2.0 Ports	2	4
Camera Input	15-pin CSI (Camera Serial Interface)	15-pin CSI (Camera Serial Interface)
Video Output	Composite 3.5 mm RCA and HDMI	HDMI, Composite, DSI (Display Serial Interface)
Audio Output	Analog 3.5 mm jack, Digital via HDMI	Analog 3.5 mm jack, Digital via HDMI
Storage	SD slot	Micro SD slot
Ethernet	10/100 Mbps	10/100 Mbps
Onboard WiFi	None	None
Onboard Bluetooth®	None	None
Input/Output Pins	26	40
Power (less peripherals)	5v 700 ma	5v 320 ma
Size	85 mm x 56 mm	85 mm x 56 mm

(Sumber : <https://www.mbtechworks.com/hardware/raspberry-pi-model-comparison.html>)

Tabel 2.2 Spesifikasi Raspberry Pi 2 Model B dan B V1.2



Feature	Pi 2 B	Pi 2 B V1.2
Released	Feb 2015	Oct 2016
Architecture	ARMv7-A 32-bit	ARMv8-A 64/32-bit
SoC	Broadcom BCM2836	Broadcom BCM2837
CPU	900 MHz ARM Cortex-A7	900 MHz ARM Cortex-A53
Cores	4	4
GPU	Broadcom VideoCore IV HD 1080p	Broadcom VideoCore IV HD 1080p
Memory RAM	1 GB	1 GB
Operating System	Primarily Linux based	Primarily Linux based
USB 2.0 Ports	4	4
Camera Input	15-pin CSI (Camera Serial Interface)	15-pin CSI (Camera Serial Interface)
Video Output	HDMI, Composite, DSI (Display Serial Interface)	HDMI, Composite, DSI (Display Serial Interface)
Audio Output	Analog 3.5 mm jack, Digital via HDMI	Analog 3.5 mm jack, Digital via HDMI
Storage	Micro SD slot	Micro SD slot
Ethernet	10/100 Mbps	10/100 Mbps
Onboard WiFi	None	None
Onboard Bluetooth ®	None	None
Input/Output Pins	40	40
Power (less peripherals)	5v 750 ma	5v 750 ma
Size	85 mm x 56 mm	85 mm x 56 mm

(Sumber : <https://www.mbtechworks.com/hardware/raspberry-pi-model-comparison.html>)

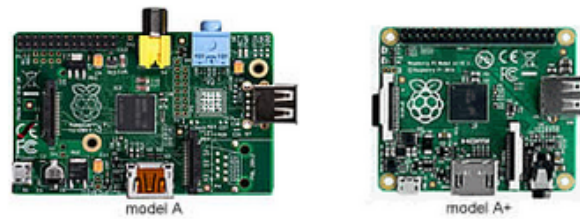
Tabel 2.3 Spesifikasi Raspberry Pi 3 Model B dan B+



Feature	Pi 3 B	Pi 3 B+
Released	Feb 2016	Mar 2018
Architecture	ARMv8-A 64/32-bit	ARMv8-A 64/32-bit
SoC	Broadcom BCM2837	Broadcom BCM2837B0
CPU	1.2 GHz ARM Cortex-A53	1.4 GHz ARM Cortex-A53
Cores	4	4
GPU	Broadcom VideoCore IV HD 1080p	Broadcom VideoCore IV HD 1080p
Memory RAM	1 GB	1 GB
Operating System	Primarily Linux based	Primarily Linux based
USB 2.0 Ports	4	4
Camera Input	15-pin CSI (Camera Serial Interface)	15-pin CSI (Camera Serial Interface)
Video Output	HDMI, Composite, DSI (Display Serial Interface)	HDMI, Composite, DSI (Display Serial Interface)
Audio Output	Analog 3.5 mm jack, Digital via HDMI	Analog 3.5 mm jack, Digital via HDMI
Storage	Micro SD slot	Micro SD slot
Ethernet	10/100 Mbps	10/100/1000 (max 300) Mbps
Onboard WiFi	2.4 Ghz 802.11 b/g/n	2.4 Ghz and 5 Ghz 802.11 b/g/n/ac
Onboard Bluetooth®	4.1 BLE	4.2 BLE
Input/Output Pins	40	40
Power (less peripherals)	5v 850 ma	5v 950 ma
Size	85 mm x 56 mm	85 mm x 56 mm

(Sumber : <https://www.mbtchworks.com/hardware/raspberry-pi-model-comparison.html>)

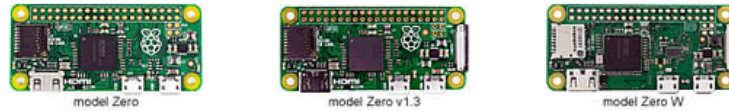
Tabel 2.4 Spesifikasi Raspberry Pi 1 Model A dan A+



Feature	Pi 1 A	Pi 1 A+
Released	Feb 2013	Nov 2014
Architecture	ARMv6Z 32-bit	ARMv6Z 32-bit
SoC	Broadcom BCM2835	Broadcom BCM2835
CPU	700 MHz ARM1176JZF-S	700 MHz ARM1176JZF-S
Cores	1	1
GPU	Broadcom VideoCore IV HD 1080p	Broadcom VideoCore IV HD 1080p
Memory RAM	256 MB	512 MB
Operating System	Primarily Linux based	Primarily Linux based
USB 2.0 Ports	1	1
Camera Input	15-pin CSI (Camera Serial Interface)	15-pin CSI (Camera Serial Interface)
Video Output	HDMI, Composite RCA, DSI	HDMI, Composite TTRS, DSI (Display Serial Interface)
Audio Output	Analog 3.5 mm jack, Digital via HDMI	Analog 3.5 mm jack, Digital via HDMI
Audio Output	Analog 3.5 mm jack, Digital via HDMI	Analog 3.5 mm jack, Digital via HDMI
Storage	SD slot	Micro SD slot
Ethernet	None	None
Onboard WiFi	None	None
Onboard Bluetooth ®	None	None
Input/Output Pins	26	40
Power (less peripherals)	5v 300 ma	5v 200 ma
Size	85 mm x 56 mm	65 mm x 56 mm

(Sumber : <https://www.mbtechworks.com/hardware/raspberry-pi-model-comparison.html>)

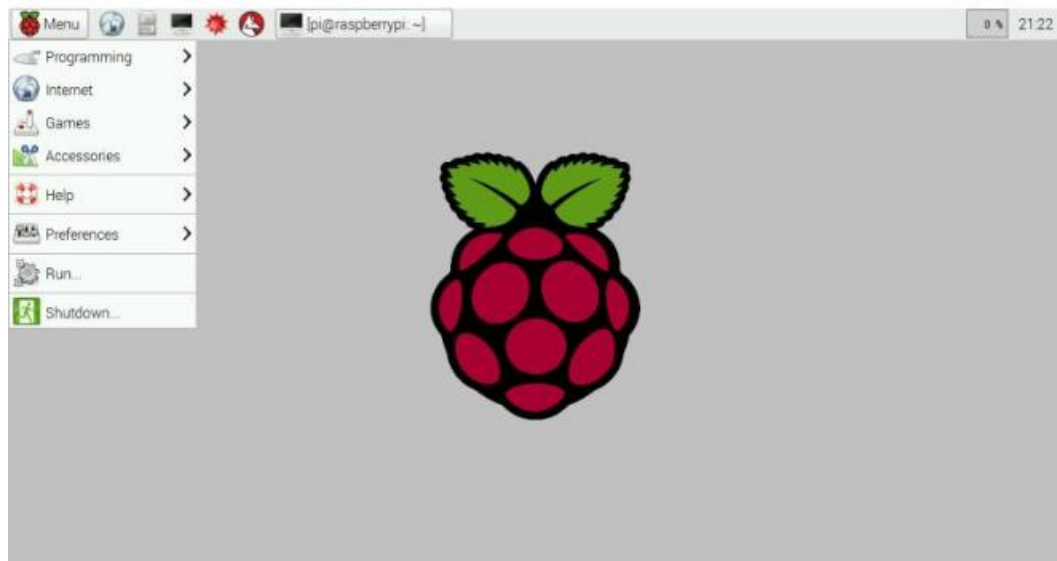
Tabel 2.5 Spesifikasi Raspberry Pi Zero V1.2, V1.3 dan *Wireless*



Feature	Pi Zero V1.2	Pi Zero V1.3	Pi Zero W (Wireless)
Released	Nov 2015	May 2016	Feb 2017
Architecture	ARMv6Z 32-bit	ARMv6Z 32-bit	ARMv6Z 32-bit
SoC	Broadcom BCM2835	Broadcom BCM2835	Broadcom BCM2835
CPU	1 Ghz ARM1176JZF-S	1 Ghz ARM1176JZF-S	1 Ghz ARM1176JZF-S
Cores	1	1	1
GPU	Broadcom VideoCore IV HD 1080p	Broadcom VideoCore IV HD 1080p	Broadcom VideoCore IV HD 1080p
Memory RAM	512 MB	512 MB	512 MB
Operating System	Primarily Linux based	Primarily Linux based	Primarily Linux based
USB 2.0 Ports	1 Micro USB	1 Micro USB	1 Micro USB
Camera Input	None	15-pin CSI (Camera Serial Interface)	15-pin CSI (Camera Serial Interface)
Video Output	Mini HDMI, Composite on PCB	Mini HDMI, Composite on PCB	Mini HDMI, Composite on PCB
Audio Output	Digital via mini-HDMI	Digital via mini-HDMI	Digital via mini-HDMI
Audio Output	Digital via mini-HDMI	Digital via mini-HDMI	Digital via mini-HDMI
Storage	Micro SD slot	Micro SD slot	Micro SD slot
Ethernet	None	None	None
Onboard WiFi	None	None	2.4 Ghz 802.11 b/g/n
Onboard Bluetooth®	None	None	4.1 BLE
Input/Output Pins	40 (pins not mounted)	40 (pins not mounted)	40 (pins not mounted)
Power (less peripherals)	5v 300 ma	5v 300 ma	5v 300 ma
Size	65 mm x 30 mm	65 mm x 30 mm	65 mm x 30 mm

(Sumber : <https://www.mbtechworks.com/hardware/raspberry-pi-model-comparison.html>)

2.2.2 Raspbian (Raspberry Pi + Debian)



Gambar 2.8 Tampilan utama pada Raspbian ^[4]

Raspbian adalah sistem operasi "resmi" dari Raspberry Pi dan oleh karena itu kebanyakan orang ingin memulai dari Raspbian. Raspbian adalah versi Linux yang dibangun khusus untuk Raspberry Pi, berbasis Debian yang dioptimalkan untuk perangkat keras Raspberry Pi. Raspbian tidak berafiliasi dengan Raspberry Pi *Foundation*. Raspbian diciptakan oleh tim pengembang kecil yang merupakan penggemar perangkat keras Raspberry Pi, yang berdedikasi untuk tujuan pendidikan Raspberry Pi *Foundation* dan Proyek Debian.

Raspbian dilengkapi dengan semua perangkat lunak yang Anda perlukan untuk setiap tugas dasar dengan komputer. Anda akan mendapatkan *Libre Office* sebagai perangkat perkantoran (*Office Suite*), *browser web*, program email, dan beberapa alat untuk mengajarkan pemrograman kepada anak-anak (*Scratch*) dan orang dewasa (Python, Java, C/C++). Bahkan didalamnya terdapat permainan *Minecraft*.

Raspbian menyediakan lebih dari sekedar OS biasa, Raspbian hadir dengan lebih dari 35.000 paket, perangkat lunak pra-kompilasi (*Pre-compiled Software*) yang digabungkan dalam format yang bagus untuk memudahkan pemasangan di Raspberry Pi Anda. Paket awal dari lebih dari 35.000 paket Raspbian yang dioptimalkan untuk kinerja terbaik di Raspberry Pi selesai pada

bulan Juni 2012. Namun, Raspbian masih dalam pengembangan aktif dengan penekanan pada peningkatan stabilitas dan kinerja sebanyak mungkin paket Debian.

Raspbian adalah tulang punggung bagi hampir semua proyek *Do-It-Yourself* (DIY) yang ada di luar sana, jadi jika Anda ingin membuat sesuatu, sebaiknya mulai dari Raspbian. Karena sistem operasi ini sangat banyak yang menggunakan, juga mudah menemukan panduan dan tips pemecahan masalah. Namun, jika Anda baru mengenal Linux, Raspbian akan sedikit membingungkan bagi Anda. “*Rpi Beginners Wiki*” atau “*Raspberry Pi Tutorials Channel*” merupakan titik awal belajar yang disarankan, begitu juga berbagai sumber resmi di *website* Raspberry Pi.

2.2.3 Fungsi dari Raspberry Pi

Meskipun ukurannya sangat mini, ternyata RasPi dapat melakukan hal-hal yang tidak terduga. Berikut diantaranya:

1. Sebagai Komputer Desktop Mini

Perkembangan Raspberry Pi kini sudah semakin baik dan canggih. Fitur dan penggunaannya pun bisa di manfaatkan layaknya komputer *desktop* walaupun tetap belum bisa menandingi komputer *desktop* berbasis CPU Intel. Selain itu, Raspberry Pi ini juga di klaim lebih hemat daya.

2. Sebagai File Server

Kita dapat berbagi file film, dokumen, musik atau foto-foto lain dimana saja dan kapan saja. Kemampuan dan kelebihan ini membuat Raspberry Pi mampu seolah-olah memiliki fungsi layaknya file server.

3. Sebagai Download Server

Dari generasi ke generasi, RasPi saat ini bisa digunakan juga sebagai *download server*. Dengan Raspberry, penggunaanya bisa melakukan pengontrolan dan pengelolaan file yang di *download* via web, baik *web browser desktop*, *smartphone* ataupun tablet.

4. Sebagai Access Point

Device Raspberry yang kita miliki saat ini bisa dijadikan sebagai *access point* dengan menancapkan adapter [Wi-Fi](#) yang kompatibel.

5. Sebagai Server DNS

Kita bisa menjadikan server DNS pada Raspberry Pi sebagai pengganti server DNS ISP yang melambat dengan bantuan aplikasi seperti BIND9.

6. Sebagai Multimedia Player

Selain hal-hal diatas, kita bisa memanfaatkan RaspPi sebagai *media player* untuk menonton film, mendengarkan *music*, melihat foto, menonton *youtube* atau bermain social media dengan menggunakan TV lawas kita sebagai monitornya.

2.2.4 Kelebihan Raspberry Pi

Kelebihan utama Raspberry Pi adalah ia dapat melakukan segala hal yang dapat dilakukan oleh komputer/laptop dengan sistem operasi Linux. Misalnya, membuat server, membuat program dengan berbagai macam bahasa, terutama bahasa tingkat tinggi seperti Python. Untuk fungsi sehari-hari, Raspberry dapat menjalankan sistem operasi berbasis GUI, sehingga dapat menggunakannya untuk melakukan pekerjaan standard seperti *browsing*, mendengarkan musik, nonton film, bermain *game*, mengetik dll. Untuk penggunaan tingkat lanjut, Raspberry Pi hampir tidak memiliki batasan. Banyak sekali kemungkinan pengembangan aplikasi yang dapat dilakukan dengan menggunakan Raspberry Pi.

Raspberry Pi seakan menggantikan fungsi komputer, tapi dalam bentuk mini. Raspberry Pi bisa disamakan dengan sebuah komputer mini. Anda dapat menjalankan sistem operasi lengkap, seperti Linux dan *Android* pada Raspberry Pi. Anda juga dapat membuat program pada sistem operasi tersebut yang dapat mengontrol fungsi sistem dan *pin general purpose input-output* yang tersedia. Raspberry Pi didesain untuk digunakan pada level yang lebih tinggi. Dengan

perangkat keras yang telah terintegrasi yang bisa digunakan untuk mengatur peralatan seperti ethernet, video, *audio processing*, jumlah RAM yang besar dan jumlah penyimpanan yang hampir tak terbatas.

Raspberry Pi berjalan menggunakan sistem operasi *open source*, Linux. Raspberry Pi juga dapat dihubungkan dengan ke monitor komputer biasa, dan tambahan port untuk menghubungkannya dengan *mouse* dan *keyboard*. Dan untuk penyimpanan data, Raspberry Pi tidak menggunakan *Hard Disk* namun Raspberry Pi dapat menggunakan *SD Card* untuk menyimpan data, baik itu data *Operating System* ataupun untuk media penyimpanan data jangka panjang. Dengan memanfaatkan teknologi SoC (*System On Chip*), Raspberry Pi berjalan di atas arsitektur ARM11 seperti yang dapat ditemui pada iPhone 3G maupun *smartphone* lain dan dilengkapi dengan videocore 4 GPU yang mampu memutar video dengan kualitas BluRay.

2.2.5 Kekurangan Raspberry

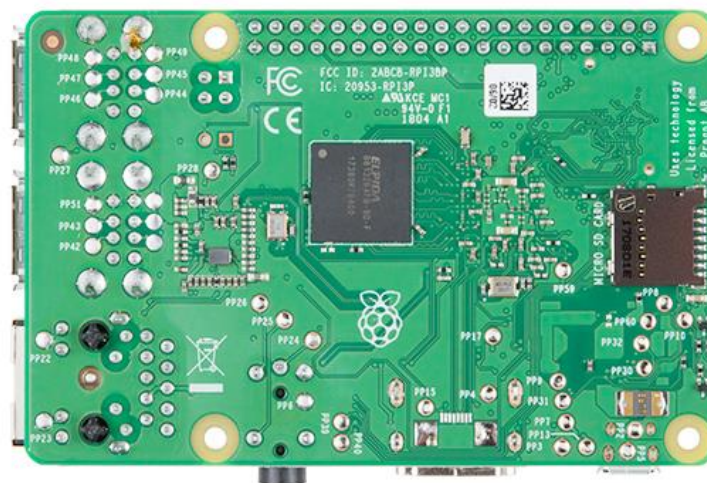
Raspberry Pi dapat membaca sensor digital secara langsung, akan tetapi Raspberry Pi tidak bisa langsung dihubungkan dengan sensor analog. Raspberry Pi yang menggunakan sistem operasi dan *SD card* yang memerlukan prosedur khusus ketika ingin memmatikannya. Jadi harus di-*shutdown* sebagaimana komputer pada umumnya.

Raspberry Pi mungkin akan terdapat sedikit lebih lambat karena kernel Linux pada sistem operasi Raspberry Pi memiliki fungsi prioritas proses seperti yang dimiliki oleh semua sistem operasi. Kernel linux harus menangani banyak proses dengan prioritas yang ditentukan, sehingga proses menggerakkan lengan akan lebih lambat.

2.3 Raspberry Pi 3 Model B+



Gambar 2.9 *Front Side* Raspberry pi 3 Model B+ ^[5]



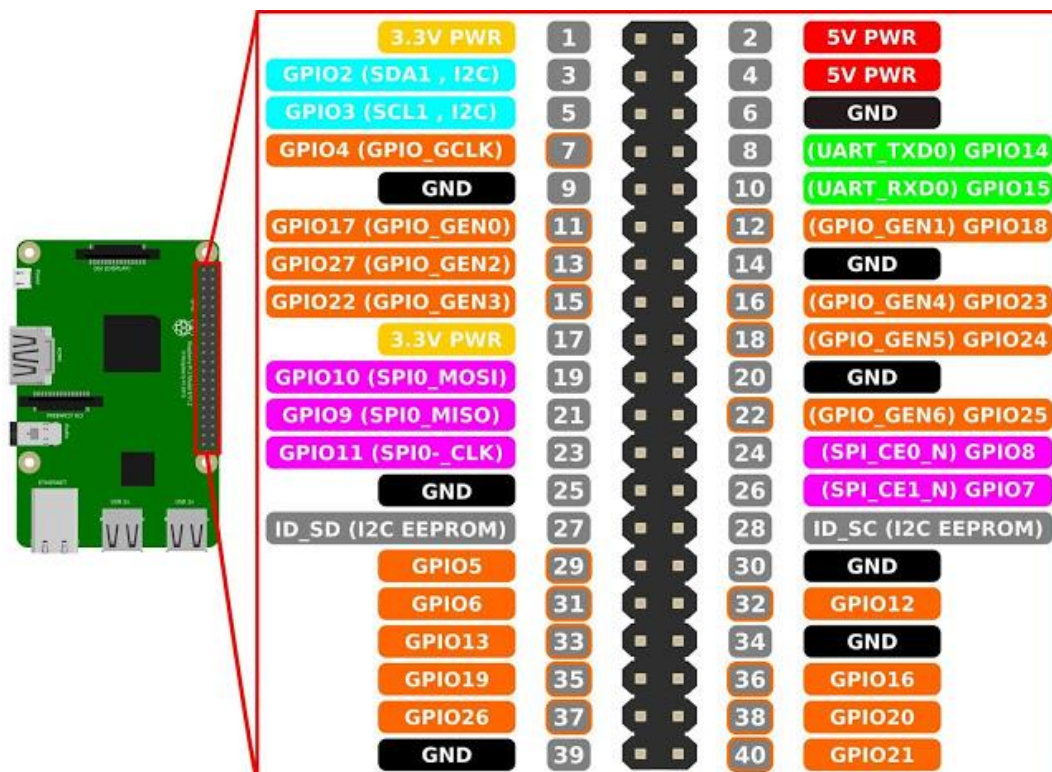
Gambar 2.10 *Back Side* Raspberry Pi 3 Model B+ ^[5]

Beberapa kelebihan dan peningkatan raspberry Pi 3 Model B+ di banding sebelumnya sebagai berikut:

1. Raspberry Pi 3 Model B+ menggunakan *chipset* baru yaitu Broadcom BCM2837B0 Cortex A53 64-bit 1,4GHz, mempunyai kecepatan *Pocessor* jauh lebih cepat.
2. Memiliki kemampuan Jaringan lebih baik dengan Koneksi *Wireless dual band* yang sudah mendukung 802.11ac dan Bluetooth 4.2

3. *Chipset* pada Raspberry pi 3 Model B+ Memiliki manajemen suhu yang lebih baik
4. *Faster Ethernet (Gigabit Ethernet over USB 2.0)*
5. *Power-over-Ethernet support*

KOMUNIKASI PIN GPIO



Gambar 2.11 Raspberry Pi 3 GPIO Pin ^[5]

Pada Raspberry Pi terdapat GPIO Pin yang dapat di gunakan untuk *interface* dengan *device* lain, di antara fungsi GPIO pin di antaranya :

1. Power Pin

Pada GPIO Raspberry Pi sudah tersedia *Power Pin* Gnd, 3.3 V dan 5 V.

2.GPIO

Secara umum GPIO adalah standar pin, yang dapat di gunakan untuk *On/Off* , misalnya pada Led.

3. PWM (*Pulse-Width Modulation*)

Hardware PWM = GPIO12, GPIO13, GPIO18, GPIO19.

4. SPI (*Serial Peripheral Interface Bus*)

- Pin SPI digunakan untuk komunikasi dengan *module* dengan *interface* SPI, misalnya RFID, Untuk detail pin GPIO nya sebagai berikut :
- SPI0:MOSI(GPIO10);MISO(GPIO9);SCLK(GPIO11);CE0(GPIO8),CE1(GPIO7)
- SPI1:MOSI(GPIO20);MISO(GPIO19);SCLK(GPIO21);CE0(GPIO18);CE2(GPIO16)

5. I2C (*Inter-Integrated Circuit*)

Pin I2C dapat di gunakan untuk komunikasi dengan *Module* yang *support* dengan I2C Protokol, misalnya RTC. Untuk Pin GPIO nya sebagai berikut:

- Data: (GPIO2); *Clock* (GPIO3)
- EEPROM Data: (GPIO0); EEPROM *Clock* (GPIO1)

6. Serial

Digunakan untuk *Serial Input* dan *Output*, Komunikasi untuk *peripheral external*, seperti RS232 atau Modbus. Pin *GPIO* nya sebagai berikut :

- TX (GPIO14); RX (GPIO15)

2.4 Modul Kamera Raspberry Pi 5MP



Gambar 2.12 Modul Kamera Raspberry Pi 5MP ^[4]

Modul ini secara khusus dibuat untuk Raspberry . Memiliki konektor yang akan dipasang langsung ke papan Raspberry dan mendukung video berkualitas HD hingga 1080p . Sudut pengambilan video disarankan dipasang tegak lurus terhadap objek atau ± 90 derajat.

2.5 Python Programming Language



Gambar 2.13 Logo Bahasa Pemrograman Python ^[6]

Raspberry Pi dengan sistem operasi Raspbian ataupun sistem operasi lainnya yang berbasis Linux mendukung Python sebagai bahasa pemrograman utama selain bahasa pemrograman seperti C/C++ .Anda bisa menginstall python sebagai sarana mempelajari bahasa pemrograman Python.

Python adalah bahasa pemrograman [interpretatif](#) multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim

sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python juga didukung oleh komunitas yang besar.

Python diciptakan pertama kali oleh Guido van Rossum pada tahun 1991. Sintaks dan fungsi pada Python dipengaruhi oleh beberapa bahasa seperti C, C++, Lisp, Perl dan Java. Oleh karena itu, kita dapat menemui konsep pemrograman prosedural, fungsional dan object-oriented di Python. Python relatif mudah dipelajari bila dibandingkan dengan C++, Java dan PHP karena sintaks Python lebih singkat, lebih jelas dan mudah dipahami oleh programmer pemula.

Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada [pemrograman berorientasi objek](#), pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai [bahasa pemrograman dinamis](#) yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai [bahasa skrip](#) meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai *platform* [sistem operasi](#).

Beberapa fitur yang dimiliki Python adalah:

1. Memiliki kepustakaan yang luas; dalam distribusi Python telah disediakan modul-modul 'siapa pakai' untuk berbagai keperluan.
2. Memiliki tata bahasa yang jernih dan mudah dipelajari.
3. Memiliki aturan *layout* kode sumber yang memudahkan pengecekan, pembacaan kembali dan penulisan ulang kode sumber.
4. Berorientasi objek.
5. Memiliki sistem pengelolaan memori otomatis (*garbage collection*, seperti [java](#))

6. Modular, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa Python maupun [C/C++](#).
7. Memiliki fasilitas [pengumpulan sampah](#) otomatis, seperti halnya pada bahasa pemrograman [Java](#), python memiliki fasilitas pengaturan penggunaan ingatan komputer sehingga para pemrogram tidak perlu melakukan pengaturan ingatan komputer secara langsung.
8. Memiliki banyak fasilitas pendukung sehingga mudah dalam pengoperasiannya.

Meskipun di Indonesia Python tidak sepopuler C/C++ ataupun Java, Python sangat patut untuk dipelajari karena banyak pilihan library Python yang bisa kita gunakan yang biasanya tidak dapat kita temui di bahasa lain. Pada bagian ini akan dijelaskan dasar-dasar dari pemrograman Python. Buka terminal, lalu ketik perintah python untuk membuka console Python.

2.5.1 Membuat Variabel

Variabel di Python tidak menyertakan tipe data secara eksplisit, berbeda dengan C++ atau Java. Semua tipe data sama cara memasukkan nilainya ke dalam variabel.

```
>>> integer = 1
>>> float = 2.5
>>> string = "aku adalah string"
>>> boolean = False
>>> integer
1
>>> float
2.5
>>> string
"aku adalah string"
>>> boolean
False
```

String dapat diapit dengan tandakutip tunggal (' ') atau tandakutip ganda (" "). Kita juga bisa mendeklarasikan string multilinedengan mengapit string dalam tandakutip tunggal atau tanda kutip ganda sebanyak tiga kali (' ' ' ' atau ' ' ' ' ' ' ' ').

```
>>> multiline = '''
... Aku seorang kapiten
... Mempunyai pedang panjang
... Kalau berjalan prok-prok-prok
... Aku seorang kapiten
... '''
...
>>> multiline
'Aku seorang kapiten
mempunyai pedang panjang
kalau berjalan prok-prok-prok
Aku seorang kapiten
'
```

2.5.2 Menulis Komentar

Python menggunakan tanda # untuk komentar. Sebuah komentar merupakan baris teks yang tidak dibaca sebagai kode oleh Python. Baris tersebut hanya diperuntukan untuk dibaca oleh manusia. Komentar membuat program yang Anda buat lebih mudah untuk dimengerti. Ketika Anda melihat kembali kode yang telah dibuat atau orang lain ingin berkolaborasi dengan Anda, maka mereka bisa membaca komentarnya dan dengan mudah mengartikan apa yang kode Anda buat. Contoh komentar satu baris:

```
#Variable berikut digunakan untuk menyimpan nomer misterius
#iniilah gunanya komentar, kita jadi tahu apa fungsi dari variable ini
mysterious variable = 42
```

Tanda # hanya akan mengomentari satu baris saja. Sementara dalam penggunaannya tentunya akan memerlukan komentar dengan panjang lebih dari satu baris, memulai dengan tanda # setiap barisnya, tentunya akan sangat menyusahakan. Makadari itu, untuk komentar multi-baris, Anda bisa memasukan seluruh blok komentar ke dalam satu set tanda yang terdiri dari tiga tanda petik dua ("""). Contoh penggunaan komentar multi-baris pada Python :

```
""" Ini adalah contoh
Komentar multi-baris
pada Python """
my var = 123
```

2.5.3 Membuat Struktur Data : List

Pada Python, kita tidak dapat menemui konsep array. Sebagai gantinya, python menyediakan struktur data list. Inisialisasi untuk list sama seperti variabel biasa.

```
>>> list = [1, 2, 3, 4]
>>> list
[1, 2, 3, 4]
>>> list = [1.5, 2.4, 0.5]
>>> list
[1.5, 2.4, 0.5]
```

Kita juga dapat membuat deret integer pada list secara otomatis dengan fungsi range.

```
>>> list = range(5)
>>> list
[0, 1, 2, 3, 4]
>>> list = range(0,5)
>>> list
[0, 1, 2, 3, 4]
>>> list = range(0,10,2)
>>> list
[0, 2, 4, 6, 8]
```

Berbeda dengan C++ dan Java, pada list kita dapat menampung elemen yang berbeda tipe data dalam satu variabel.

```
>>> list = [1, 'dua', 3.0]
>>> list
[1, 'dua', 3.0]
```

Cara akses elemen pada list sama seperti pada array, yaitu dengan menyertakan indeks elemen. Indeks dimulai dari nilai 0.

```
>>> list[0]
1
>>> list[1]
'dua'
```

List juga bisa mengandung list-list yang lain. Hal ini dikenal sebagai multi-dimensional list.

```
>>> list = [1, [1, 2, 3.0], [1.5, 'dua']]
```

Kita dapat menambahkan atau mengurangi elemen pada list yang sudah dibuat sebelumnya dengan menggunakan fungsi `append()`, `insert()` dan `remove()`

```
>>> list = [0, 1]
>>> list
[0, 1]
>>> list.append(2) # menambahkan elemen pada akhir list
>>> list
[0, 1, 2]
>>> list.insert(1, 'aku ditengah') # menambahkan elemen pada indeks 1
>>> list
[0, 'aku ditengah', 1, 2]
>>> list.remove('aku ditengah') # menghapus elemen yang nilainya 'aku ditengah'
>>> list
[0, 1, 2]
```

Untuk mengambil elemen pertama dari suatu list, kita dapat menggunakan `pop`.

```
>>> list = [0, 1]
>>> list.pop()
0
```

2.5.4 Membuat Struktur Data : Dictionary

Selain list, Python memiliki struktur data dictionary. Mirip seperti konsep hash, tiap elemen pada dictionary memiliki alias atau key.

```
>>> dict = {'satu': 1, 'dua': 2, 'tiga': 3}
```

Berbeda dengan list, cara akses nilai pada dictionary adalah dengan menyertakan nama key-nya.

```
>>> dict['satu']
1
>>> dict['dua']
2
```

Elemen pada dictionary dapat terdiri dari tipe data yang berbeda-beda, bahkan nama key juga tidak harus selalu string. Kita juga bisa memasukkan list sebagai elemen pada *dictionary*.

```

>>> dict = {'satu': 1.0, 'dua': [1, 'dua', [1, 2, 3]], 3: 'tiga'}
>>> dict[3]
'tiga'
>>> dict['dua']
[1, 'dua', [1, 2, 3]]
>>> dict['dua'][1]
['dua']
>>> dict['dua'][2]
[1, 2, 3]
>>> dict['dua'][2][1]
[2]

```

List dan dictionary adalah struktur data yang paling sering digunakan pada Python, namun struktur data pada Python tidak hanya terbatas pada itu saja. Untuk mempelajari struktur data lebih lanjut, Anda dapat membuka tutorial Python lebih lanjut.

2.5.5 Menggunakan If, Elif dan Else

Fungsi if pada Python lebih sederhana penulisannya dibandingkan C++ atau Java. Untuk mengapit pernyataan di dalam if tidak menggunakan kurung kurawal. Penulisan pernyataan yang masuk pada fungsi if lebih menjorok ke dalam (menggunakan whitespace).

```

>>> var = 1
>>> if var < 2:
...     var = var + 1
... elif var < 3:
...     var = var + 2
... else:
...     var = 0
>>> var
2

```

Selain case-sensitive, Python sangat memperhatikan whitespace untuk memisahkan pernyataan dengan deklarasi fungsi. Oleh karena itu Anda harus selalu konsisten dalam memilih whitespace (pilih salah satu antara spasi atau tab). Pastikan juga semua indentasi pada fungsi memiliki jumlah whitespace yang sama.

2.5.6 Mengulang Dengan For

Fungsi for sedikit berbeda dengan C++ atau Java, karena kita tidak menginisialisasi kondisi berhenti iterasi. Agar kita bisa menginisialisasi jumlah perulangan, kita bisa menggunakan range.

```
>>> var = 0
>>> for el in range(1, 10):
...     var = var + 1
...
>>> var
9
```

Untuk menghentikan atau melanjutkan iterasi pada suatu kondisi tertentu, kita dapat menggunakan break dan continue.

```
>>> var = 0
>>> for el in range(1, 10):
...     if var > 5:
...         break
...     elif var == 3:
...         continue
...     var = var + 1
...
>>> var
3
```

Di Python, for akan menjalankan pernyataan di dalamnya sebanyak elemen pada list dan akan berhenti setelah nilai indeks elemen melebihi nilai indeks maksimal pada list.

```
>>> list = ['maine coon', 'munchkin', 'siamese']
>>> var = 0
>>> for el in list:
...     var = var + 1
...
>>> var
2
```

2.5.7 Mengulang Dengan While

Fungsi while akan terus secara iteratif menjalankan pernyataan di dalamnya selama kondisi pada while belum terpenuhi.

```
>>> var = 10
>>> while var > 0:
...     var = var - 1
...
>>> var
0
```

2.5.8 Mencetak Dengan Print

Fungsi print digunakan untuk mencetak string pada terminal.

```
>>> print 'Halo dunia!'
Halo dunia!
```

Kita juga bisa mencetak string multiline dengan menggunakan fungsi yang sama.

```
>>> print '''
... Aku seorang kapiten
... Mempunyai pedang panjang
... Kalau berjalan prok-prok-prok
... Aku seorang kapiten
... '''
Aku seorang kapiten
Mempunyai pedang panjang
Kalau berjalan prok-prok-prok
Aku seorang kapiten
```

Menyambungkan 2 atau lebih string dapat dilakukan dengan menggunakan operator plus(+).

```
>>> print 'Halo ' + 'dunia!'
Halo dunia!
```

Untuk menambahkan keluaran yang bukan string, kita dapat menambahkan koma.

```
>>> print 'Kucing yang kami pelihara di rumah berjenis: ', ['main coon',
'siamese', 'munchkin']
Kucing yang kami pelihara di rumah berjenis: ['main coon', 'siamese', 'munchkin']
```

Selain itu, kita juga bisa memasukkan variabel di tengah-tengah string (formatting) dengan menggunakan tandakurung kurawal {}.

```
>>> x = 1
>>> y = 2
>>> print '{} + {} = {}'.format(x, y, x + y)
1 + 2 = 3
```

Mengubah urutan output dapat dilakukan dengan memberikan indeks di dalam kurung kurawal.

```
>>> print '{1} + {0} = {2}'.format(x, y, x + y)
2 + 1 = 3
```

Indeks dapat diganti dengan key seperti pada dictionary.

```
>>> print '{0} + {1} = {hasil}'.format(x, y, hasil = x + y)
1 + 2 = 3
```

Bagi Andayang familiar dengan format output yang mendeklarasikan tipe data secara eksplisit seperti pada C/C++, Andajuga bisa melakukan hal yang mirip di Python seperti dibawah ini:

```
>>> print '%d + %d = %d' % (x, y, x + y)
1 + 2 = 3
```

2.5.9 Membaca dan Menulis File

Untuk menulis teks pada file, kita dapat menggunakan write() dengan terlebih dahulu membuat file pada direktori.

```
>>> f = open('contoh_file.txt', 'w')
>>> for el in range(1,11):
...     f.write('Ini adalah baris teks {} pada file.\n'.format(el))
...
>>> f.close()
```

Untuk membaca keseluruhan teks pada file, kita dapat menggunakan read().

```
>>> f = open('contoh_file.txt', 'r')
>>> f.read()
Ini adalah baris teks 1 pada file.
Ini adalah baris teks 2 pada file.
Ini adalah baris teks 3 pada file.
Ini adalah baris teks 4 pada file.
Ini adalah baris teks 5 pada file.
>>> f.close()
```

Untuk membaca satu baris teks pada file, kita dapat menggunakan `readline()`.

```
>>> f = open('contoh_file.txt', 'r')
>>> f.readline()
Ini adalah baris teks 1 pada file.
>>> f.close()
```

Untuk membaca satu per satu baris teks pada file, kita dapat menggunakan `for`.

```
>>> f = open('contoh_file.txt', 'r')
>>> for teks in f:
...     print teks
...
Ini adalah baris teks 1 pada file.
Ini adalah baris teks 2 pada file.
Ini adalah baris teks 3 pada file.
Ini adalah baris teks 4 pada file.
Ini adalah baris teks 5 pada file.
```

Apabila kita lihat secara teliti, ada perbedaan cara antara membuka file untuk menulis dan membaca. Parameter kedua yang dimasukkan pada fungsi `open()` adalah mode pembacaan file. Di bawah ini adalah daftar mode yang dapat digunakan pada fungsi `open()`:

1. 'r' hanya untuk membaca file (read)
2. 'w' hanya untuk menulis file (write)
3. 'a' untuk menambahkan teks pada akhir teks di dalam file (append)
4. 'r+' untuk membaca dan menulis file

Selalu gunakan fungsi `close()` setelah Anda membuka file untuk menghindari terjadinya memory leak

Jika kita membuka file dengan pilihan atribut "w", bila file tersebut sudah ada, maka akan file tersebut akan dibuat kembali dan isi dari file tersebut akan otomatis terhapus. Jika kita ingin membuka file yang sudah ada kemudian menambahkan isi file tersebut kita gunakan pilihan atribut "a" (append).

Untuk menyimpan file data log, disarankan menggunakan format CSV (Comma Separated Value). Di dalam Python sudah disediakan modul CSV.

```
import csv
import datetime

temp = sensor()
temp = round(temp, 2)
now = datetime.datetime.now()
file = open("temperature.csv", "a")
writer = csv.writer(file, delimiter = ",")
data = [now, temp]
writer.writerow(data)
file.close()
```

2.5.10 Membuat Fungsi

Deklarasi fungsi pada Python dimulai dengan `def` dan diikuti dengan nama fungsi. Sama seperti deklarasi variabel, kita tidak perlu menambahkan tipe data secara eksplisit pada fungsi. Untuk fungsi yang mengembalikan nilai dapat menggunakan `return`.

```
>>> def sambung_string(str1, str2):
...     sambungan = str1 + ' dan ' + str2
...     return sambungan
```

Untuk memanggil fungsi, tulis nama fungsi setelah itu masukkan nilai parameter fungsinya jika ada.

```
>>> hasil = sambung_string('aku', 'kamu')
>>> print hasil
aku dan kamu
```

Kita juga bisa membuat nilai default pada parameter sehingga kita punya opsi untuk menerima nilai default tersebut atau tidak.

```
>>> def sambung_string(str1, str2='kamu'):
...     sambungan = str1 + ' dan ' + str2
...     return sambungan
...
>>> hasil = sambung_string('aku')
>>> print hasil
aku dan kamu
>>> hasil_lain = sambung_string('aku', 'dia')
>>> print hasil_lain
aku dan dia
```

Semua parameter pada Python adalah `passed by reference`. Sehingga apabila di dalam fungsi ada operasi yang melibatkan variabel dengan nama yang sama

dengan variabel di luar fungsi, maka nilai pada variabel di luar fungsi akan ikut berubah. Contoh:

```
>>> def tambah_elemen(elemen):
...     list.append(elemen)
...     print 'Output di dalam fungsi', list
...
>>> list = [1, 2]
>>> tambah_elemen(3)
Output di dalam fungsi [1, 2, 3]
>>> print 'Output di luar fungsi', list
Output di luar fungsi [1, 2, 3]
```

2.5.11 Membuat Program Object Oriented

Kita dapat mengimplementasikan konsep object oriented pada Python. Inisialisasi kelas pada Python dapat dilihat pada contoh di bawah ini.

```
>>> class Kucing:
...     def __init__(self): # dobel underscore
...         self.jenis = 'domestik'
...         self.bulu = 'kuning'
...         self.buntut = 'pendek'
...         self.nama = 'Ambercat'
...     def deskripsi(self):
...         print 'Hai namaku {}. Aku adalah kucing berjenis {}. Warna buluku {} dan buntutku {}.\n'.format(self.nama, self.jenis, self.bulu, self.buntut)
...
>>>
```

Pada contoh, kelas Kucing juga memiliki dua method, yaitu `__init__()` dan `deskripsi()`. Method `__init__()` adalah constructor yang akan selalu dipanggil setiap objek Kucing dibuat. Dalam contoh ini, `__init__()` akan membuat empat atribut, yaitu jenis, bulu, buntut dan nama. Setiap ingin membuat method, Anda harus selalu memasukkan parameter `self` pada deklarasi method.

Cara membuat objek dari kelas Kucing dapat dilihat pada contoh di bawah ini.

```
>>> meong = Kucing()
```

Cara untuk memanggil method pada objek adalah sebagai berikut.

```
>>> meong.deskripsi()
Hai namaku Ambercat. Aku adalah kucing berjenis domestik. Warna buluku kuning dan buntutku pendek.
```

Supaya kita dapat mengeset nilai atribut jenis, bulu, buntut dan nama, kita dapat memodifikasi kelas Kucing menjadi seperti di bawah ini.

```
>>> class Kucing:
...     def __init__(self, jenis='domestik', bulu='kuning', buntut='pendek',
nama='Ambercat'): # dobel underscore
...         self.jenis = jenis
...         self.bulu = bulu
...         self.buntut = buntut
...         self.nama = nama
...     def deskripsi(self):
...         print 'Hai namaku {}. Aku adalah kucing berjenis {}. Warna buluku
{} dan buntutku {}.\n'.format(self.nama, self.jenis, self.bulu, self.buntut)
... 
```

Untuk mengganti beberapa nilai atribut pada inisialisasi objek, kita dapat menggunakan cara berikut.

```
>>> meong = Kucing(bulu='putih', buntut='panjang')
>>> meong.deskripsi()
Hai namaku Ambercat. Aku adalah kucing berjenis domestik. Warna buluku putih dan
buntutku panjang.
```

Bagian ini belum mencakup seluruh fungsi dan konsep pada Python. Untuk mempelajari lebih lanjut mengenai Python, Anda dapat mengunjungi dokumentasi Python lebih lanjut.

2.6 Open CV Programming



Gambar 2.14 Logo OpenCV ^[7]

OpenCV (*Open Source Computer Vision Library*) adalah sebuah [pustaka perangkat lunak](#) yang ditujukan untuk [pengolahan citra](#) dinamis secara *real-time*, yang dibuat oleh [Intel](#), dan sekarang didukung oleh [Willow Garage](#) dan Itseez.

Program ini bebas dan berada dalam naungan [sumber terbuka](#) dari [lisensi BSD](#). Pustaka ini merupakan pustaka [lintas platform](#). Pustaka memiliki lebih dari 2500 algoritma dioptimasi yang termasuk sekumpulan komprehensif dari algoritma komputer visi pada mesin pembelajaran klasik dan pengembangan terbaru. Algoritma-algoritma itu dapat digunakan mendeteksi dan mengenali wajah-wajah, identifikasi obyek-obyek, klasifikasi aksi manusia dalam video-video, pelacakan pergerakan kamera, pelacakan pergerakan objek, ekstraksi objek-objek dari model-model 3D, menghasilkan titik awan-awan 3D dari kamera stereo, menggabungkan citra-citra untuk menghasilkan sebuah citra resolusi tinggi pada seluruh kejadian, menemukan citra-citra mirip dari sebuah basis data citra, Menghapus mata merah dari gambar yang diambil menggunakan flash, mengikuti pergerakan mata, mengenali pemandangan dan memberikan penanda-penanda untuk hamparan atas itu dengan kenyataan tambahan.

2.7 Face Detection

Face Detection / Deteksi wajah adalah teknologi komputer yang sedang diterapkan untuk berbagai aplikasi yang membutuhkan identifikasi wajah manusia dalam gambar atau video digital. Ini dapat dianggap sebagai kasus khusus deteksi kelas objek, di mana tugasnya adalah menemukan lokasi dan ukuran semua objek dalam gambar yang termasuk kelas tertentu. Teknologi ini mampu mendeteksi wajah frontal atau dekat-frontal dalam foto, terlepas dari orientasi, kondisi pencahayaan, atau warna kulit.

Aplikasi *Face Detection* / Deteksi wajah menggunakan algoritma yang memutuskan apakah suatu gambar adalah gambar positif (gambar wajah) atau gambar negatif (gambar non-wajah). Ini disebut *classifier*. Untuk mengklasifikasikan gambar baru dengan benar, ini dilatih pada ratusan ribu gambar wajah dan non-wajah. Fitur ini menjawab pertanyaan "Di mana wajah-wajah dalam gambar ini?". Untuk setiap wajah yang terdeteksi, Anda mendapatkan analisis lengkap tentang poin-poin penting (*landmark*) di sekitar mata, alis mata, rahang, hidung, dan mulut.

2.7 Aplikasi Telegram



Gambar 2.15 Logo Telegram ^[8]

Telegram adalah sebuah aplikasi layanan [pengirim pesan instan multiplatform](#) berbasis [awan](#) yang bersifat gratis dan nirlaba. [Klien](#) Telegram tersedia untuk perangkat telepon seluler ([Android](#), [iOS](#), [Windows Phone](#), [Ubuntu Touch](#)) dan sistem perangkat komputer ([Windows](#), [OS X](#), [Linux](#)). Para pengguna dapat mengirim pesan dan bertukar foto, video, stiker, audio, dan tipe berkas lainnya. Telegram juga menyediakan pengiriman pesan [ujung ke ujung terenkripsi](#) opsional.

Berbagai kelebihan yang ditawarkan yang sangat berguna pada penelitian ini seperti adanya *cloud* pada server *Telegram Messenger* yang memungkinkan untuk menyimpan data-data seperti percakapan, foto dan video. Fitur *bot* yang memiliki kecerdasan artifisial merupakan fitur yang dapat terintegrasi dengan dengan berbagai layanan melalui internet. Dengan fitur *bot* inilah penulis akan membuat suatu sistem yang dapat terintegrasi pada sistem keamanan rumah.