

BAB II

LANDASAN TEORI

2.1. *Prototype*

Menurut [7] *Prototype* atau *prototyping* adalah kegiatan yang sangat penting didalam proses physical computing karena pada tahap inilah seorang perancang melakukan eksperimen dan uji coba dari berbagai jenis komponen, ukuran, parameter, program komputer dan sebagainya berulang-ulang kali sampai diperoleh kombinasi yang paling tepat. Dalam hal ini perhitungan angka-angka dan rumus yang akurat bukanlah satu-satunya faktor yang menjadi kunci sukses di dalam mendesain sebuah alat karena ada banyak faktor eksternal yang turut berperan, sehingga proses mencoba dan menemukan/mengoreksi kesalahan perlu melibatkan hal-hal yang sifatnya non-eksakta. *Prototyping* adalah gabungan antara akurasi perhitungan dan seni. pada proses model *prototype* dapat dijelaskan sebagai berikut:

a. *User Requirements*

Pada tahap ini *devoleper* dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian – bagian yang akan dibutuhkan berikutnya. Detil kebutuhan mungkin tidak dibicarakan pada tahap ini.

b. *Develope Prototype*

Pada tahap ini dilakukan perancangan sistem *prototype* oleh *developer*, perancangan sistem dilakukan secara cepat dan rancangan diusahakan mewakili semua aspek *softwarei* yang telah diketahui.

c. *Revise Prototype*

Pada tahap ini dilakukan evaluasi *prototype system* oleh klien. Apabila klien merasa *protoype system* telah dikembangkan sesuai dengan keinginannya maka *prototype* tersebut dapat digunakan, akan tetapi jika *prototype* tersebut tidak sesuai, maka *prototype*, tersebut akan dilakukan revisi dan digunakan sebagai acuan dalam memperjelas kebutuhan

software dan kemudian dikembangkan *prototype* selanjutnya. Model ini terdapat 5 fase atau tahapan yang terdiri dari sebagai berikut:

- a. Pengumpulan Kebutuhan
Pelanggan dan pengembang bersama – sama mendefinisikan format dan kebutuhan keseluruhan perangkat lunak. Mengidentifikasi semua semua kebutuhan dan garis besar sistem yang akan dibuat.
- b. Membangun *Prototype*
Membangun *prototype* dengan membuat perancangan sementara yang berpusat pada penyajian kepada pelanggan (misalkan dengan membuat *input* dan contoh *output* nya).
- c. Mengkodekan Sistem
Dalam tahap ini *prototyping* yang sudah disepakati diterjemahkan kedalam bahasa pemrograman yang sesuai.
- d. Menguji Sistem
Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites terlebih dahulu sebelum digunakan. Pengujian ini dilakukan dengan *white box*, *black box*, basis path, pengujian arsitektur dan lain – lain.
- e. Evaluasi *Prototype*
Evaluasi ini dilakukan oleh pelanggan apakah *prototype* yang sudah dibangun atau sudah selesai dengan keinginan pelanggan. Jika sudah sesuai maka langkah keempat akan diambil.

2.2. Sistem

Sistem adalah suatu himpunan dari berbagai bagian atau elemen, yang saling berhubungan secara terorganisasi berdasar fungsi-fungsinya menjadi satu kesatuan. Berdasarkan definisi sebelumnya [8]. Maka dapat saya simpulkan sistem adalah sekelompok unsur yang saling terhubung satu sama lain yang berfungsi bersama-sama untuk mencapai suatu tujuan tertentu.

2.2.1. Karakteristik Sistem

Menurut [9] bahwa karakteristik sistem adalah sebagai berikut:

- a. **Komponen Sistem (*Component*)**
Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar atau sering disebut supra sistem.
- b. **Batasan Sistem (*Boundary*)**
Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau dengan lingkungan lainnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan
- c. **Lingkungan Luar Sistem (*Environment*)**
Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.
- d. **Penghubung (*Interface*)**
Media yang menghubungkan sistem dengan subsistem yang lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu *sub* sistem ke *sub* sistem lain. Bentuk keluaran dari satu *sub* sistem akan menjadi masukan untuk *sub* sistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.
- e. **Masukan Sistem (*Input*)**
Energi yang dimasukkan ke dalam sistem disebut masukan sistem yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

f. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini dapat menjadi masukan bagi *sub* sistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi *input* bagi *sub* sistem lain.

g. Pengolah Sistem (Proses)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran (*Objective*)

Suatu sistem mempunyai tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.3. Rumah Pintar (*SmartHome*)

Rumah Pintar (*Smart Home*) adalah aplikasi gabungan antara teknologi dan pelayanan yang dikhususkan pada lingkungan ramah dengan fungsi tertentu yang bertujuan meningkatkan keamanan efisiensi dan kenyamanan penghuninya. Sistem rumah pintar (*Smart Home*) biasanya terdiri dari perangkat monitoring, perangkat kontrol dan otomatis ada beberapa perangkat yang dapat di akses menggunakan komputer [3]. *SmartHome* bisa dibidang sebagai pengenalan pada penelitian Internet of Things (IoT). Hal ini dikarenakan pengimplementasiannya terdapat efisiensi dan otomatis penggunaan teknologi lainnya [10]. Rumah pintar (*Smart Home*) bisa disebut aplikasi gabungan antara teknologi dan pelayanan yang dikhususkan pada lingkungan rumah atau gedung dengan fungsi tertentu yang bertujuan untuk meningkatkan keamanan, efisiensi dan kenyamanan penghuninya. Sistem rumah pintar (*smart home*) biasanya terdiri dari perangkat monitoring, perangkat kontrol dan otomatis ada beberapa perangkat yang dapat di akses menggunakan computer[11]. *SmartHome* juga merupakan suatu perangkat yang terintegrasi dengan konsep *Internet of Things* yang

memungkinkan untuk mengkoleksi data, memproses data, dan merubah data. *SmartHome* memberikan beberapa kemungkinan bagi pengguna untuk mengontrol, dan menyesuaikan rumah berdasarkan keinginan pemilik [12]. Rumah pintar atau lebih dikenal dengan istilah *SmartHome* sebuah tempat tinggal atau kediaman yang menghubungkan jaringan komunikasi dengan peralatan listrik yang dimungkinkan dapat dikontrol, dimonitor atau diakses dari jarak jauh. Smart home juga dapat meningkatkan efisiensi, kenyamanan dan keamanan dengan menggunakan teknologi secara otomatis[13]. Secara garis besar, *SmartHome* diciptakan untuk memudahkan pemilik atau penghuni rumah dalam mengatur dan mengontrol beberapa perangkat elektronik supaya dapat bekerja secara otomatis serta dapat diakses dari jarak jauh.

2.4. *Internet of Things (IoT)*

Internet of Things adalah konsep yang muncul dimana semua alat dan layanan terhubung satu dengan yang lain dengan mengumpulkan, bertukar dengan memproses data untuk beradaptasi secara dinamis[14]. *Internet of Things (IoT)* merupakan kumpulan benda-benda (things), berupa perangkat fisik (hardware /embedded system) yang mampu bertukar informasi antar sumber informasi, operator layanan ataupun perangkat lainnya yang terhubung kedalam sistem sehingga dapat memberikan kemanfaatan yang lebih besar. Perangkat fisik (hardware/embedded system) dalam infrastruktur *Internet of Things* merupakan *hardware* yang tertanam (embedded) dengan elektronik, perangkat lunak, sensor dan juga konektivitas[15]. *Internet of Things* dalam penerapannya juga dapat mengidentifikasi, menemukan, melacak, memantau objek dan memicu *event* terkait secara otomatis dan *real time*, Pengembangan dan penerapan komputer, Internet dan teknologi informasi dan komunikasi lainnya (TIK) membawa dampak yang besar pada masyarakat manajemen ekonomi, operasi produksi, sosial manajemen dan bahkan kehidupan pribadi[14].

2.5. Android

Android adalah sebuah *Operating System* berbasis linux yang dipakai oleh perangkat seluler seperti Smartphone dan tablet [16]. Android merupakan suatu Operating System atau OS yang sampai saat ini masih dalam tahap perkembangan, OS ini seperti OS lainnya seperti, Symbian, IOS di I-Phone, dan lain sebagainya[17]. Berdasarkan uraian pendapat diatas dapat diartikan bahwa pengertian dari android adalah suatu sistem operasi pada *SmartPhone* atau tablet yang mempunyai banyak fitur didalamnya untuk mempermudah kehidupan manusia dan sampai sekarang terus berkembang semakin canggih.

2.6. Mikrokontroler

Mikrokontroler adalah mikroprosesor yang dikhususkan untuk instrumentasi dan kendali. Mikroprosesor merupakan suatu alat elektronik digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus[18]. Mikrokontroler berisikan bagian – bagian utama yaitu CPU (*Central Processing Unit*), RAM (*Random Access Memory*), ROM (*Read Only Memory*) dan port 1/0 (Input/Output). Mikrokontroler tertentu bahkan menyertakan ADC(*Analog Digital Converter*), USB *controller*, CAN (*Contoller Area Network*) dan lain – lain[10].

2.6.1. Sejarah Singkat Mikrokontroler

Menurut [12] menyatakan bahwa mikrokontroler pertama kali dikenalkan oleh *Texas Instrument* dengan seri TMS 1000 pada tahun 1974 yang merupakan mikrokontroler 4bit pertama mikrokontroler ini mulai dibuat sejak 1971, yang merupakan mikro komputer dalam sebuah *chip* lengkap dengan RAM dan ROM. Kemudian pada tahun 1976 intel mengeluarkan mikrokontroler yang kelak menjadi populer dengan nama 8748 yang merupakan mikrokontroler 8bit, yang merupakan mikrokontroler dari keluarga MCS 48. Menurut [19] bahwa Tahun 1835, Charles Babbage mengusulkan komputer digital (*Digital Computer*) pertama didunia menggunakan punched card untuk data dan instruksi, serta program control (*looping and branching*) dengan unit aritmatik dan unik

penyimpanan. Saat ini, mikrokontroler yang banyak dikeluarkan oleh Atmel dengan seri AT89Sxx, dan mikrokontroler AVR yang merupakan mikrokontroler RISC dengan seri ATMEGA8535 (walaupun varian dari mikrokontroler AVR sangatlah banyak, dengan masing-masing memiliki fitur yang berbeda-beda).

2.6.2. Jenis-Jenis Mikrokontroler

Menurut [18] menyatakan bahwa secara teknis hanya ada dua macam mikrokontroler yaitu RISC dan CISC serta masing-masing mempunyai keturunan atau keluarga sendiri-sendiri. Instruksi yang dimiliki RISC (*Reduced Instruction Set Computer*) terbatas, tetapi memiliki fasilitas yang lebih banyak. Sedangkan instruksi CISC (*Complex Instruction Set Computer*) bisa dikatakan lebih lengkap tetapi dengan fasilitas secukupnya. Berikut adalah jenis-jenis mikrokontroler yang umum digunakan:

a. Keluarga MCS51

Menurut [18] Mikrokontroler ini termasuk dalam keluarga mikrokontroler CISC. Sebagian besar intruksinya dieksekusi dalam 12 siklus *clock*. Salah satu kemampuan dari mikrokontroler ini adalah pemasangan sebuah mesin pemroses *boolean* yang mengizinkan operasi logika *boolean* tingkatan bit dapat dilakukan secara langsung dan secara efisien dalam *register* internal dan RAM.

b. AVR

Menurut [19] Mikrokontroler TinyAVR (ATTiny) adalah mikrokontroler 8 bit. ATTiny merupakan mikrokontroler avr kecil dan memiliki peripheral yang terbatas.

c. PIC (*Programmable Interface Controller*)

Menurut [18] Pada awalnya, PIC merupakan kependekan dari *Programmable Interface Controller*. Tetapi pada perkembangannya berubah menjadi *Programmable Intelligent Computer*. PIC termasuk keluarga mikrokontroler berarsitektur *Harvard* yang dibuat oleh *Microelectronic General Instruments* dengan nama PIC640.

d. Arduino

Menurut [18] Arduino adalah kit elektronik atau papan rangkaian elektronik *opensource* yang didalamnya terdapat komponen utama yaitu sebuah *chip* mikrokontroler dengan jenis AVR dari perusahaan Atmel.

e. Mikrokontroller Xmega

Menurut [19] Mikrokontroller Xmega adalah mikrokontroller 8/16 bit. Xmega memiliki periphal baru dan canggih dengan untuk kerja, sistem monitoring event dan DMA yang ditingkatkan,serta merupakan pengembangan keluarga AVR untuk pasar low power dan high performance. Dengan adanya fitur DMA (*Direct Memory Access*) dapat mengurangi kemungkinan terjadinya kemacetan pada saat transfer data. Xmega mendukung kriptografi AES dan DES.

2.7. Sekilas Tentang NodemMCU ESP

NodeMCU adalah suatu perangkat keras akses terbuka *platform* dari sistem kerja Internet of Things. Pengembangan Kit menggunakan Bahasa pemrograman LUA sebagai pembantu programmer ketika membuat contoh alat produk internet of things. Kit ini di kembangkan berdasarkan modul ESP8266 yang terintegrasi dengan gpio, ICC, I-Wire, dan ADC semua dalam satu board[7]. ESP 8266 dari seri ESP besutan *Espressif System*, juga *firmware* yang digunakan merupakan bahasa pemrograman scripting Lua. Istilah Node MCU secara default sebenarnya mengacu pada *firmware* yang digunakan dari pada perangkat keras *development kit*, dan Node MCU juga bisa diartikan sebagai board arduino-nya ESP 8266. Selain dengan bahasa Lua NodeMCU juga *support* dengan software Arduino IDE dengan melakukan sedikit perubahan pada board manager di dalam software Arduino IDE yaitu dengan menambahkan URL untuk mengunduh board khusus NodeMCU pada board manager[20]. Dengan kata lain NodeMCU merupakan sebuah papan mikrokontroler yang berbasis modul wireless ESP8266, sehingga sesuai untuk diterapkan di bidang *Internet of Things*, *SmartHome* kontrol atau aplikasi pengendalian tanpa kabel lainnya.

tersedia di versi 0.9, ESP8266 menggunakan WiFi 2,4 GHz, mendukung WPA/WPA2[21].

1. Kelebihan
 - a. NodeMCU memiliki fitur WiFi yang terintegrasi.
 - b. Kapasitas flash memory, RAM dan clock NodeMCU jauh lebih tinggi disbanding dengan Arduino Uno.
 - c. Harga board NodeMCU lebih murah.
2. Kekurangan
 - a. Dukungan library NodeMCU belum banyak yang dimiliki oleh Arduino Uno.
 - b. Komunitas pemakai NodeMCU lebih sedikit dari pada pemakai Arduino Uno.
3. Spesifikasi :
 - a. Tipe ESP8266 ESP-12E
 - b. Vendor Pembuat LoLin
 - c. USB port Micro USB
 - d. GPIO 13 Pin
 - e. ADC 1 pin (10 bit)
 - f. Usb to Serial Converter CH340G
 - g. Power Input 5 Vdc



Gambar 2.1 Bentuk fisik NodeMCU ESP8262

Sumber: <https://cyberblogspot.com/wp-content/uploads/2020/01/nodemcu-v3-pinout-complete-final-1024x839.png>

2.8. Jenis – jenis NodeMCU

Menurut [22] saat ini terdapat setidaknya tiga produsen NodeMCU yang produknya kini beredar di pasaran, antara lain Amica, DOIT, dan Lolin/WeMos. Dengan beberapa varian board yang diproduksi yakni V1, V2 dan V3 seperti pada gambar 2.2



Gambar 2.2 Bentuk fisik jenis – jenis NodeMCU

Sumber : <https://i2.wp.com/www.nyebarilmu.com/wp-content/uploads/2017/07/Versi-Nodemcu.png?resize=640%2C244&ssl=1>

- a. NodeMCU Versi 0.9 (v1)
Pada versi ini (v0.9) merupakan versi pertama yang memiliki memori flash 4 MB sebagai (System on Chip) SoC-nya dan ESP8266 yang digunakan yaitu ESP-12.
- b. NodeMCU Versi 1.0 (v2)
- c. Versi ini merupakan pengembangan dari versi 0.9. Dan pada versi 1.0 ini ESP8266 yang digunakan yaitu tipe ESP-12E yang dianggap lebih stabil dari ESP-12. Selain itu ukuran papan modulnya diperkecil sehingga dapat digunakan membuat prototipe proyek di breadboard. Serta terdapat pin yang dikhususkan untuk komunikasi SPI (Serial Peripheral Interface) dan PWM (Pulse Width Modulation) yang tidak tersedia di versi 0.9.
- d. NodeMCU Versi 1.0 (Unofficial Board) (v3)
Generasi ketiga NodeMCU ini biasa disebut v3 Lolin Dikatakan unofficial board dikarenakan produk modul ini diproduksi secara tidak resmi terkait

persetujuan dari Developer Official NodeMCU. Setidaknya sampai tulisan ini dibuat, belum ada versi resmi untuk v3 NodeMCU. Generasi v3 hanyalah versi yang diciptakan oleh produsen LoLin dengan perbaikan minor terhadap v2, untuk perbedaannya dapat dilihat pada Tabel 2.2. Diklaim memiliki antarmuka USB yang lebih cepat dan penambahan V USB *power output*.

Tabel 2.1 Resume untuk jenis – jenis NodeMCU

Spesifikasi	Versi NodeMCU		
	Versi 0.9	Versi 0.1 (<i>Officesial Board</i>)	Versi 0.1 (<i>Unofficesial Board</i>)
Vendor Pembuat	Amica	Amica	Lolin
USB Port	Micro USB	Micro USB	Micro USB
GPIO Pin	11	13	13
ADC	1 Pin (10 bit)	1 Pin (10 bit)	1 Pin (10 bit)
USB to Serial converter	CH340G	CCP2102	CH340G

Tabel 2.2 Fungsi Pinout ESP8266

No	Nama Pin	Deskripsi
1	Micro-USB	Pasti semuanya sudah tau bagian ini ya. Fungsinya sebagai power yang dapat terhubung dengan USB port. Selain itu, biasanya juga digunakan untuk melakukan pengiriman sketch atau memantau data serial dengan serial monitor di aplikasi Arduino IDE
2	3.3V	Digunakan sebagai tegangan untuk device lainnya. ada 3 tempat untuk 3.3v. Biasanya juga dituliskan hanya 3V (Sebenarnya tetap 3,3V)
3	GND	Ground. Sebagai tegangan 0 atau nilai negatif untuk mengalirkan arus.

4	Vin	Sebagai External Power yang akan mempengaruhi Output dari seluruh pin. Cara menggunakannya yaitu dengan menghubungkannya dengan tegangan 7 hingga 12vol
5	EN, RST	Pin yang digunakan untuk reset program di mikrokontroler.
6	A0	Analog pin, digunakan untuk membaca input secara analog.
7	GPIO 1 – GPIO 16	Pin yang dapat digunakan sebagai <i>Input</i> dan <i>Output</i> . Pin ini dapat melakukan pembacaan dan pengiriman data secara analog juga.
8	SD1,CMD, SD0,CLK	SPI Pin untuk komunikasi SPI (Serial Peripheral Interface) dimana kita akan menggunakan clock untuk sinkronisasi deteksi bit pada receiver.
9	TXD0, RXD0, TXD2, RXD2	Sebagai <i>interface</i> UART, Pasangannya adalah TXD0 dengan RXD0 dan TXD2 dengan RXD2. TXD1 digunakan untuk upload <i>firmware</i> /program.
10	SDA, SCL (I2C Pins)	Digunakan untuk device yang membutuhkan I2C

2.8. Bahasa Pemrograman NodemMCU ESP8266

Menurut [12] bahwa banyak bahasa yang bisa digunakan untuk program mikrokontroler, misalnya Bahasa *assembly*. Namun dalam pemrograman Arduino bahasa yang dipakai adalah bahasa C. Akar bahasa C adalah bahasa BCPL yang dikembangkan oleh Martin Richards pada tahun 1967. Bahasa C adalah bahasa *standart*, artinya suatu program yang ditulis dengan versi bahasa C tentu akan dapat dikomplikasikan dengan versi bahasa C yang lain dengan sedikit modifikasi. Beberapa alasan mengapa bahasa C banyak digunakan, diantaranya adalah sebagai berikut:

1. Bahasa C tersedia hamper disemua jenis komputer.
2. Kode bahasa C bersifat *portable*.

3. Bahasa C hanya menyediakan sedikit kata-kata kunci.
4. Proses *executable program* bahasa C lebih cepat.

2.9. Aplikasi Blynk

Menurut [23] Blynk adalah IOT *Cloud platform* untuk aplikasi iOS dan Android yang berguna untuk mengontrol Arduino, Raspberry Pi, dan board-board sejenisnya melalui Internet. Blynk adalah dashboard digital di mana kita dapat membangun sebuah antarmuka grafis untuk alat yang telah dibuat hanya dengan menarik dan menjatuhkan sebuah *widget*. Blynk sangat mudah dan sederhana untuk mengatur semuanya dan hanya dalam waktu kurang dari 5 menit.



2.10. Power Suply

Semua peralatan elektronika menggunakan sumber tenaga untuk beroperasi, sumber tenaga tersebut bermacam-macam ada yang dari baterai, Accu, ada juga yang langsung menggunakan tegangan listrik jala-jala PLN, untuk konsumsi tegangan yang berasal dari tegangan listrik untuk alat-alat elektronika tertentu tidak bisa langsung dikonsumsi akan tetapi harus disesuaikan dengan tegangan yang diperlukan oleh peralatan tersebut. Penyesuaian tegangan ini dilakukan oleh sebuah alat yang dinamakan *Power Supply* atau adaptor.



Gambar 2.4 Bentuk fisik *Power Suply* 12V 3A

Sumber: <https://id-test-11.slatic.net/p/c641a78bb0f51cecb235082d415fde51.png>

2.11. *Module Relay 8 Channel 5V*

Menurut [13] *Module relay* merupakan suatu piranti yang menggunakan elektromagnetik untuk mengoperasikan seperangkat kontak saklar. Susunan sederhana *module relay* terdiri dari kumparan kawat penghantar yang dililitkan pada inti besi. Bila kumparan diberi energi, medan magnet yang terbentuk menarik amatur berporos yang digunakan sebagai pengungkit mekanisme saklar. Adapun lebih jelas bentuk fisik kabel *Relay* seperti digambar 2.6;



Gambar 2.5 Bentuk fisik Relay 8 channel 12V

Sumber: (https://1.bp.blogspot.com/-LfyBINUY0qU/XKYiy_HGWII/AAAAAAAAATEs/I7_qSBV30C0XbmmDKCxsrmJK15CB6s-DtQCLcBGAs/w1200-h630-p-k-no-nu/relay%2B8ch.jpg)

2.12. Kabel USB Standar A-B

Menurut [12] bahwa kabel USB ini digunakan untuk menghubungkan antara papan arduino dengan komputer. Fungsinya untuk komunikasi antara arduino dan komputer juga salah satu sumber daya pada arduino. Kabel ini sama dengan kabel printer sambungan US. Adapun lebih jelas bentuk fisik kabel USB standart seperti digambar 2.7;

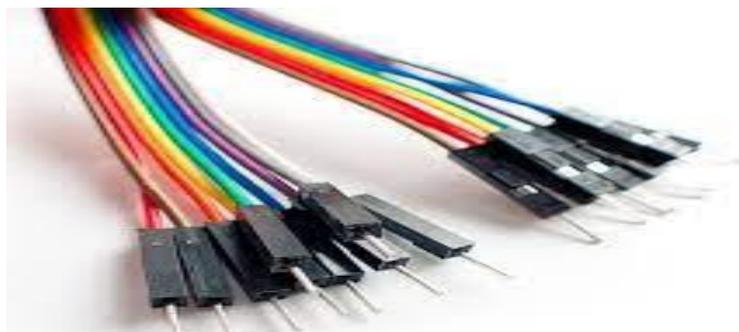


Gambar 2.6 Bentuk fisik kabel USB A-B

Sumber : (<https://blog.dimensidata.com/mengenal-macam-jenis-tipe-kabel-usb-dan-versi-kabel-usb/>)

2.13. Kabel Jumper (*JumpWire*)

Menurut [12] bahwa kabel jumper adalah kabel – kabel pendek digunakan untuk menghubungkan antara komponen yang satu dengan komponen yang lainnya pada breadboard sehingga terdapat hubungan listrik dan terbentuk rangkaian elektronik. Adapun lebih jelas bentuk fisik *board jumper wire* seperti digambar 2.8



Gambar 2.7 Bentuk fisik kabel *Jumper Wire*

Sumber: (<https://www.aldyrazor.com/2020/04/kabel-jumper-arduino.html>)

2.14. *Solenoid DoorLock*

Menurut [23] *Solenoid doorlock* adalah salah satu solenoid pengunci otomatis yang di fungsikan khusus sebagai solenoid untuk pengunci pintu. Tegangan solenoid pengunci pintu ini rata – rata 12 vol. Pada kondisi normal tua solenoid memanjang atau terkunci. Jika terjadi tegangan tuas akan terbuka mendadak. Adapun lebih jelas bisa dilihat dari gambar 2.9



Gambar 2.8 Bentuk fisik *Solenoid DoorLock*

Sumber: (https://1.bp.blogspot.com/LfyBINUY0qU/XKYiy_HGWII/AAAAAATEs/17_qSBV30C0XbmmDKCxsMJkI5CB6s-DtOCLcBGAs/w1200-h630-p-k-no-nu/relay%2B8ch.jpg)

2.15. Definisi Tools Pengembang Perangkat Lunak

2.15.1. UML (unified Modeling Language)

Menurut [24] bahwa UML (*unified Modeling Language*) adalah salah satu tool/model untuk merancang pengembangan software yang berbasis object oriented. UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. [25] mengatakan bahwa Setiap sistem yang kompleks seharusnya bisa dipandang dari sudut yang berbeda-beda sehingga kita bisa mendapatkan pemahaman secara menyeluruh. Untuk upaya tersebut UML menyediakan 8 jenis diagram yang dapat dikelompokkan berdasarkan sifat-sifatnya (statis dan dinamis). Ke-8 jenis diagram dalam UML tersebut adalah:

a. *Use Case Diagram*

Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem.

b. *Sequence Diagram*

Sequence diagram bersifat dinamis. Diagram urutan ini adalah siagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu. *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu. Diawali dari apa yang mentrigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan.

c. *Activity diagram*

Activity diagram bersifat dinamis. Diagram ini adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya disebuah sistem. *Activity diagram* merupakan state diagram khusus, dimana sebagian besar state adalah *action* dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah (sistem dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

d. *Component Diagram*

Component Diagram bersifat statis. Diagram ini memperlihatkan organisasi serta kebergantungan pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan ke dalam satu atau lebih kelas-kelas, antarmuka, serta kolaborasi-kolaborasi.

e. *Deployment Diagram*

Deployment Diagram bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Diagram ini memuat simpul-simpul (node) beserta komponen-komponen yang ada di dalamnya. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada mesin (*distributed computing*).

2.15.2. *Flowchart*

Menurut [26] bahwa *flowchart* adalah bagan (*chart*) yang menunjukkan alir (*flow*) didalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. [27] Mengatakan *Flowchart* ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan poses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Setelah *flowchart* selesai disusun, selanjutnya programmer menerjemahkan nya ke bentuk program dengan bahasa pemrograman. Terbagi 2 jenis *flowchart* sebagai berikut :

a. *System Flowchart*

System Flowchart merupakan diagram alir yang menggambarkan suatu sistem peralatan komputer yang digunakan dalam proses pengolahan data serta hubungan antar peralatan tersebut. Sistem *flowchart* tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah , tetapi hanya untuk menggambarkan prosedur dalam sistem yang dibentuk.

b. *Program Flowchart*

Program Flowchart merupakan bagan alir yang menggambarkan urutan logika dari suatu prosedur pemecahan masalah. Untuk menggambarkan *flowchart* program telah tersedia simbolsimbol standart.

2.16. *Black Box Testing*

Menurut [22] bahwa *blackbox testing* adalah tipe testing yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para tester memandang perangkat lunak seperti layaknya sebuah kotak

hitam yang tidak penting dilihat isinya, tetapi cukup dikenal proses testing dibagian luar. Jenis testing ini hanya memandangi perangkat lunak dari sisi spesifikasi dan kebutuhan yang telah di definisikan pada saat awal perancangan. Sebagai contoh jika terdapat sebuah perangkat lunak yang merupakan sebuah sistem informasi *inventory* di sebuah perusahaan. Maka pada jenis *whitebox testing* perangkat lunak tersebut akan dibongkar listing programnya untuk kemudian di tes menggunakan teknik – teknik yang telah dijelaskan sebelumnya. Sedangkan pada jenis *blackbox testing*, perangkat lunak tersebut akan dieksekusi kemudian berusaha di tes apakah telah memenuhi kebutuhan pengguna yang didefinisikan pada saat awal tanpa harus membongkar *listing* programnya.

2.17. Penelitian Terdahulu

Berikut beberapa penelitian yang digunakan sebagai referensi untuk membantu mempermudah melakukan penelitian.

Tabel 2.3 Penelitian Terdahulu

No	Nama Penelitian	Judul Penelitian	Hasil Penelitian
1	Bekti Maryuni Susanto, Ery Setiyawan Jullev Atmadji, Willy Laurent Brenkman (2018)	Implementasi Mqtt Protocol Pada Smart Home Security Berbasis Web	Pada MQTT sendiri mempunyai keunggulan yaitu dapat mengirimkan data dengan bandwidth yang ringan, konsumsi listrik yang sedikit, latensi serta konektivitas yang sangat tinggi, ketersediaan variable yang banyak serta jaminan pengiriman data yang dapat dinegosiasikan. Paper ini membahas tentang implementasi MQTT protocol pada smart home security berbasis web. Topik ini dipilih karena keamanan rumah merupakan permasalahan yang sangat penting, apalagi

			saat kita meninggalkan rumah.
2	Harrizki Arie Pradana, Risna Risna, (2014)	RANCANG BANGUN APLIKASI MONITORING PENGGUNAAN AIR PDAM BERBASIS MIKROKONTROLER ARDUINO UNO	Hasil yang dicapai adalah meningkatkan aspek kenyamanan dan kemudahan yang umumnya digunakan pada masyarakat awam dalam monitoring penggunaan debit air setiap bulannya, dimana yang sebelumnya menggunakan meteran yang tidak semua orang bisa membacanya, dan rangkaian ini dapat bekerja dengan water flow sensor pada mikrokontroler arduino uno[28].
3	Setiawan, A., & Purnamasari, A. I. (2019)[Pengembangan Smart Home Dengan <i>Microcontrollers</i> ESP32 Dan MC-38 Door <i>Magnetic Switch Sensor</i> Berbasis <i>Internet of Things</i> (IoT) Untuk Meningkatkan Deteksi Dini Keamanan Perumahan	Alasan atau tujuan dari penelitian ini adalah untuk mengembangkan perangkat smarthome yang terhubung dengan internet atau <i>Internet of Things</i> (IoT). Dengan perangkat smarthome berbasis <i>Internet of Things</i> (IoT) diharapkan dapat memberikan solusi untuk mendeteksi secara dini keamanan perumahan melalui internet atau bisa diaplikasikan melalui smartphone.
4	Ahmad Khalid Azzam (2019)	Implementasi Pengontrolan <i>Smart Home</i> Menggunakan Voice Command Pada Facebook Messenger	Pada implementasi ini diujikan peralatan elektronik seperti lampu dan kipas, fungsi pada lampu yaitu untuk menyalakan dan mematikan dan fungsi pada kipas yaitu untuk menyalakan dengan pelan, menyalakan dengan kencang dan mematikan. Untuk menghubungkan aplikasi perangkat lunak dan keras, penelitian ini memanfaatkan mini komputer Raspberry Pi

			agar lampu dan kipas bisa dikontrol. Perintah suara yang diterima pada chat facebook messenger dikirim menuju Heroku agar bahasa perintahnya diproses wit.ai, dikirimkan ke database pada Firebase dan perintah menuju perangkat Raspberry pi yang terhubung dengan relay yang mengalirkan listrik menuju perangkat lampu dan kipas.
5	Zainal Abidin, Susmini Indriani Lestarinigati (2014)	Sistem Keamanan dan Monitoring Rumah Pintar secara Online Menggunakan Perangkat Mobile	Hasil dari penelitian ini adalah dapat pengontrolan peralatan rumah tangga seperti lampu dan kunci pintu perangkat yang dikembangkan telah dapat memantau (memonitoring) keadaan dan kondisi melalui IP Camera dan juga dapat memberikan peringatan ketika kunci pintu dibuka secara paksa.
6	Fauzan Masykur, Fiqiana Prasetyuwati (2016)	Aplikasi Rumah Rumah Pintar <i>Smart Home</i> pengendalian Peralatan Elektronik Rumah Tangga Berbasis Web	Hasil penelitian ini adalah Aplikasi yang digunakan untuk mengendalikan peralatan elektronik rumah tangga yang berupa lampu, AC, TV pengendalian ini dapat dijalankan oleh pengguna melalui interface web yang akan terhubung dengan mikrokontroler, mikrokontroler yang digunakan dalam aplikasi ini adalah raspberry pi dan aplikasi rumah pintar ini yang di kendalikan LedLed.
7	Bayu Prakasa A, Muhammad Syahril Qiram, Dedy Hermanto (2014)	Automatisasi <i>Smart Home</i> Dengan Raspberry Pi Dan <i>Smartphone Android</i>	Penelitian ini untuk merancang rumah sistem dengan menggunakan media komunikasi wireless dengan menggunakan wifi dan mikrokontroler Raspberry Pi sebagai pengendali dan

			<p>pengolah data dari user yang akan memberikan perintah <i>on</i> atau <i>off</i> kepada lampu. Pemakaian wifi mengharuskan pengguna melakukan pemasangan router windows atau wifi didalam rumah.</p>
8	Ageng Setiani Rafika, Meidy Surya Hadi Putra, Winda Larasati (2014)	<i>Smart Home Automatic Menggunakan Media Bluetooth Berbasis Mikrokontroler Atmega 328</i>	<p>Penelitian ini disebut merancang <i>smart home system</i> dengan menggunakan media komunikasi <i>bluetooth</i> dan mikrokontroler atmega 328 yang berfungsi mengelolah data masukan dari user yang akan memberikan kondisi <i>on</i> atau <i>off</i> pada perangkat elektronik yang dilakukan dengan menggunakan aplikasi bluesistem yang menginputkan beberapa kode sebagai perintah <i>on</i> atau <i>off</i> contohnya kode "A" pintu sudah terbuka "B" pintu sudah tertutup.</p>
9	Angger Dimas Bayu Sadewo, Edita Rosana Widasari, Adharul Muttaqin (2017)	Perancangan Pengendalian Rumah Menggunakan Smartphone Android Dengan Konektivitas <i>Bluetooth</i>	<p>Hasil dari penelitian ini adalah smartphone dapat mengendalikan beberapa peralatan elektronik yaitu lampu, kipas dan penguncian pintu secara otomatis. Sistem ini juga dapat mengirimkan notifikasi jika mengalami sebuah kondisi darurat. Dengan menggunakan ini dapat diharapkan membuat <i>user</i> lebih mudah melakukan aktivitas didalam rumah serta meningkatkan kenyamanan dan keamanan dalam menjalani aktivitas didalam rumah.</p>
10	Harrizki Arie Pradana (2022), Jukardi	Implementasi <i>Smart Home</i> Dengan Konsep <i>Internet Of Things (Iot)</i>	<p>Hasil dari penelitian ini adalah dapat pengontrollan peralatan rumah tangga</p>

	Afriyansyah	Berbasis <i>Smartphone</i>	seperti lampu, kipas angin, dan alat elektronik lainnya dengan menggunakan <i>smartphone</i> agar membuat <i>user</i> lebih mudah melakukan aktivitas didalam rumah serta meningkatkan kenyamanan dan kemudahan didalam rumah[29].
--	-------------	----------------------------	--

