

BAB II

TINJAUAN PUSTAKA

2.1 Monitoring

Monitoring adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan kontinyu tentang kegiatan atau program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program atau kegiatan itu selanjutnya. (Nely, 2012)

Pengertian monitoring menurut para ahli :

1. Cassely dan Kumar 1987

Monitoring merupakan program yang terintegrasi, bagian penting dipraktek manajemen yang baik dan arena itu merupakan bagian integral di manajemen sehari-hari.

2. Calyton dan Petry 1983

Monitoring sebagai suatu proses mengukur, mencatat, mengumpulkan, memproses dan mengkomunikasikan informasi untuk membantu pengambilan keputusan manajemen program/proyek.

3. Oxfam 1995

Monitoring adalah mekanisme yang sudah menyatu untuk memeriksa yang sudah untuk memeriksa bahwa semua berjalan untuk direncanakan dan memberi kesempatan agar penyesuaian dapat dilakukan secara metodologis.

Dari pengertian-pengertian diatas maka dapat disimpulkan bahwa Monitoring adalah bagian dari kegiatan pengawasan terhadap suatu aktivitas dimana dalam pemantauan pada umumnya bertujuan memeriksa apakah program yang telah berjalan itu sesuai dengan sasaran atau sesuai dengan tujuan dari program tersebut atau tidak.

2.2 Modul Wifi ESP8266 Nodemcu

Modul *wifi* Esp8266 NodeMcu merupakan sebuah *opensource platform* IOT dan pengembangan *kit* yang menggunakan bahasa pemrograman Lua untuk membantu programmer dalam membuat prototipe produk IOT atau bisa memakai sketch dengan arduino IDE untuk memprogramnya. Modul *Wifi* ESP8266 NodeMCU merupakan turunan pengembangan dari modul platform IoT (*Internet of Things*) keluarga esp8266. Pengembangan *Kit* ini mengintegrasikan GPIO, PWM (*Pulse Width Modulation*), I2C, dan ADC (*Analog to Digital Converter*) semua dalam satu board. Nodemcu ini sendiri yaitu Boardnya yang berukuran sangat kecil yaitu panjang 4.83 cm, lebar 2.54 cm, dengan berat 7 gram serta dengan daya yang rendah untuk menyalakanya. walaupun ukurannya yang kecil, board ini sudah dilengkapi dengan fitur *Wi-Fi*. (Ilham, 2018: 3)



Gambar 2.1 Modul *Wifi* Esp8266 Nodemcu

Modul *wifi* esp8266 Nodemcu ini menggunakan bahasa Lua untuk memprogramnya. Bahasa Lua memiliki logika dan susunan pemrograman yang sama dengan C hanya berbeda syntax. Jika menggunakan bahasa Lua maka dapat menggunakan tool Lua loader maupun Lua uploder.

Selain dengan bahasa Lua NodeMCU juga support dengan software Arduino IDE dengan melakukan sedikit perubahan board manager pada Arduino IDE untuk memprogram nya.

Sebelum digunakan Board ini harus di Flash terlebih dahulu agar support terhadap tool yang akan digunakan. Jika menggunakan Arduino IDE menggunakan firmware yang cocok yaitu firmware keluaran dari Ai-Thinker yang support AT *Command*. Untuk penggunaan tool loader Firmware yang di gunakan adalah firmware NodeMCU.

2.3 Library ESP8266 Pada Arduino IDE

ESP8266 merupakan modul wifi yang berfungsi sebagai perangkat tambahan mikrokontroler seperti Arduino agar dapat terhubung langsung dengan wifi dan membuat koneksi TCP/IP.

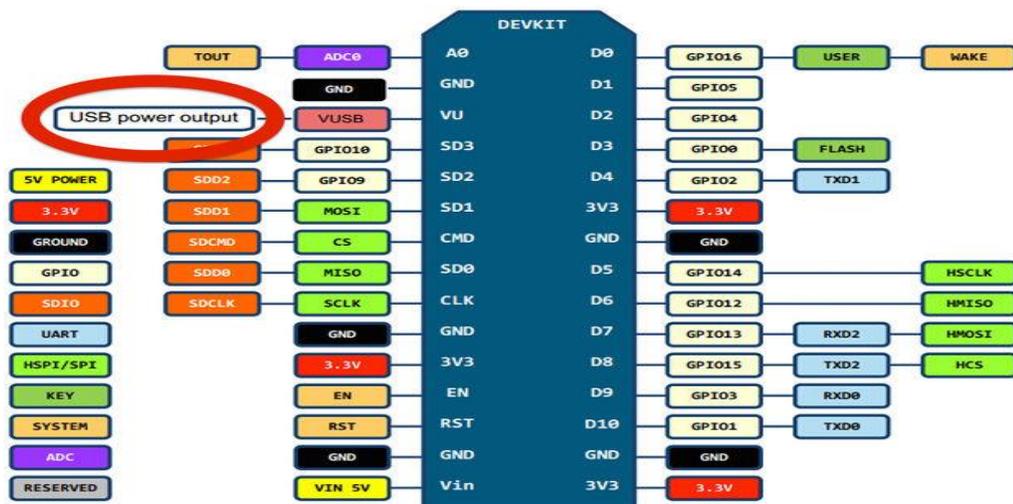
Pustaka atau library merupakan koleksi program yang digunakan untuk membangun dan mengembangkan perangkat lunak yang biasanya mengandung kode program atau data pembantu hal ini mengizinkan kode dan data dapat digunakan bersama-sama dan dapat diubah dengan menggunakan paradigma pemrograman moduler sehingga membuat modul wifi ESP8266 Nodemcu ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler.

Selain itu memprogram perangkat ini menggunakan Arduino IDE. Dengan menambahkan library ESP826 pada board manager sehingga dapat memprogram dengan basic program Arduino Tujuan memasang board ESP8266 adalah supaya board yang support WIFI ESP8266 seperti NodeMCU, Wemos, dan sebagainya bisa di Program pada Arduino IDE. <http://www.esp8266.com>.

Tabel 2.1. Spesifikasi Modul *Wifi* ESP8266 Nodemcu

Spesifikasi	Modul <i>Wifi</i> ESP8266 Nodemcu
Mikrokontroler	Modul Wifi ESP8266 Nodemcu
Ukuran Board	48.3 mm x 25.4 mm
Tegangan Input	3.3 ~ 5V
GPIO	13 Pin
Kanal PWM	10 Kanal
10 bit ADC Pin	1 Pin

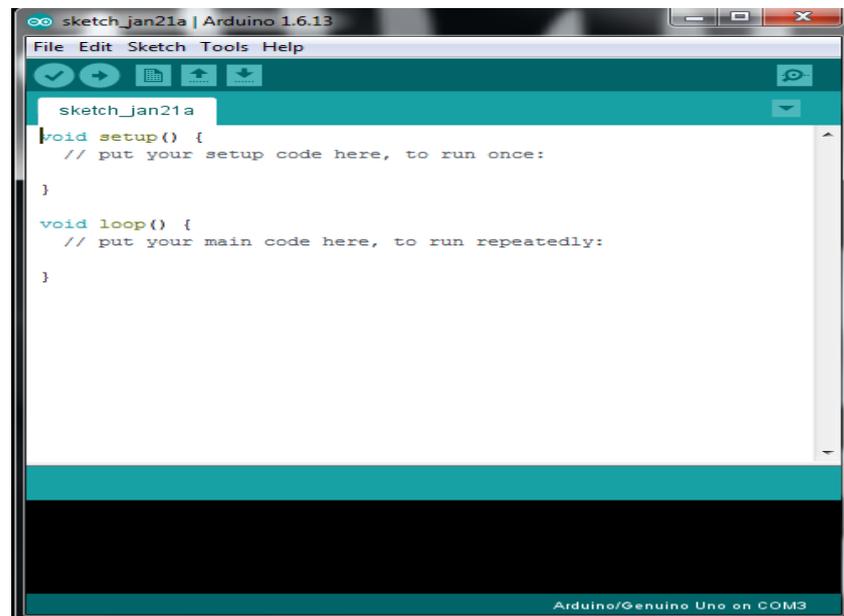
Flash Memori	4 MB
Clock Speed	24/26/40 MHz
WIFI	IEEE 802.11 b/g/n
Frekuensi	2.4 Ghz – 22.5 GHz
USB Port	Micro USB



Gambar 2.2 Posisi Pin Modul Wifi Esp8266 Nodemcu

2.4 Software Arduino IDE

Arduino *Development Environment* terdiri dari editor teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tombol-tombol untuk fungsi yang umum dan beberapa menu. Arduino *Development Environment* terhubung ke ESP8266 board untuk meng-*upload* program dan juga untuk berkomunikasi dengan ESP8266 board.



Gambar 2.3 Tampilan Arduino IDE

Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan *library* C/C++ yang biasa disebut *Wiring* yang membuat operasi input dan output menjadi lebih mudah. Arduino IDE ini dikembangkan dari *software* Processing yang dirombak menjadi Arduino IDE untuk pemrograman dengan Arduino atau perangkat Mikrokontroler lainnya yang telah dilengkapi *library* perangkat yang ingin dihubungkan itu sendiri.

Perangkat lunak yang ditulis menggunakan Arduino *Development Environment* disebut *sketch*. *Sketch* ditulis pada editor teks. *Sketch* disimpan dengan file berekstensi .ino. area pesan memberikan informasi dan pesan error ketika kita menyimpan atau membuka *sketch*. Konsol menampilkan output teks dari Arduino *Development Environment* dan juga menampilkan pesan error ketika kita mengkompilasi *sketch*. Pada sudut kanan bawah jendela Arduino *Development Environment* menunjukkan jenis board dan port serial yang sedang digunakan. (Safaat, 2015)

Tabel 2.2 Fungsi *shortcut button* di Arduino IDE

No	Icon	Nama	Fungsi
1.		Verify	Untuk mengecek error kode program yang telah dibuat.
2.		Upload	Meng- <i>upload</i> program ke Arduino board.
3.		New	Untuk membuat <i>sketch</i> baru.
4.		Open	Membuka <i>sketch</i> program yang telah disimpan.
5.		Save	Menyimpan <i>sketch</i> program yang dibuat.
6.		Serial	Membuka layar serial

2.5 Blynk

Blynk adalah sebuah aplikasi yang dapat digunakan untuk kontrol jarak jauh menggunakan *smartphone*. Blynk dapat diunduh di Google play untuk pengguna android dan Apps Store untuk pengguna ios. Blynk juga mendukung berbagai macam hardware yang dapat digunakan untuk *project Internet of Things*. Penambahan komponen pada Blynk Apps dengan cara *Drag and Drop* sehingga memudahkan dalam penambahan komponen *Input/output* tanpa perlu kemampuan pemrograman Android maupun iOS. (Ilham, 2018)

Blynk diciptakan dengan tujuan untuk *control* dan *monitoring* hardware secara jarak jauh menggunakan komunikasi data internet. Kemampuan untuk menyimpan data dan menampilkan data secara visual baik menggunakan angka, warna ataupun grafis.

Terdapat 3 komponen utama Blynk, yaitu :

1. Blynk Apps

Blynk Apps memungkinkan untuk membuat *project interface* dengan berbagai macam komponen *input/output* yang mendukung untuk pengiriman maupun penerimaan data serta merepresentasikan data sesuai dengan komponen yang dipilih. Representasi data dapat berbentuk visual angka maupun grafik.

Terdapat 4 jenis kategori komponen yang berdatap pada Aplikasi Blynk

- a. *Controller* digunakan untuk mengirimkan data atau perintah ke Hardware.
- b. *Display* digunakan untuk menampilkan data yang berasal dari hardware ke *smartphone*.
- c. *Notification* digunakan untuk mengirim pesan dan notifikasi.
- d. *Interface* Pengaturan tampilan pada aplikasi Blynk dapat berupa menu ataupun tab.
- e. *Others* beberapa komponen yang tidak masuk dalam 3 kategori sebelumnya diantaranya Bridge, RTC, Bluetooth.

2. Blynk Server

Blynk server merupakan fasilitas *Backend Service* berbasis *cloud* yang bertanggung jawab untuk mengatur komunikasi antara aplikasi blynk di *smartphone* dengan lingkungan hardware. Kemampun untuk menangani puluhan hardware pada saat yang bersamaan semakin memudahkan bagi para pengembang sistem IoT. Blynk server juga tersedia dalam bentuk *local server* apabila digunakan pada lingkungan tanpa internet. Blynk server local bersifat open source dan dapat diimplementasikan pada Hardware Raspberry Pi.

3. Blynk Library

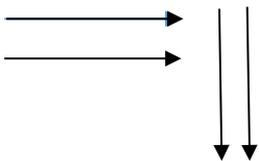
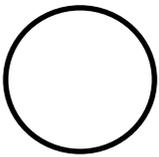
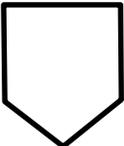
Blynk Library dapat digunakan untuk membantu pengembangan *code*. Blynk library tersedia pada banyak platform perangkat keras sehingga semakin

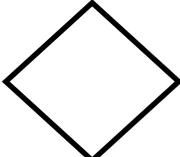
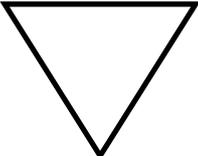
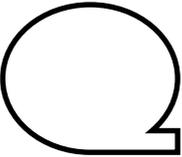
memudahkan para pengembang IoT dengan fleksibilitas hardware yang didukung oleh lingkungan Blynk.

2.6 Flowchart

Flowchart atau diagram alir merupakan representasi grafik dari langkah-langkah yang harus diikuti dalam menyelesaikan suatu permasalahan yang terdiri dari sekumpulan simbol, dimana masing-masing simbol mempresentasikan suatu kegiatan tertentu. Flowchart diawali dengan menerima input, pemrosesan input, dan diakhiri dengan menampilkan output. (Agus, 2014)

Tabel 2.3 Simbol-simbol dan keterangan *Flowchart*

No	Simbol	Keterangan
1.		Simbol arus / <i>flow</i> , yaitu menyatakan jalannya arus suatu program.
2.		Simbol <i>connector</i> , berfungsi menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
3.		Simbol <i>manual</i> , yaitu menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.
4.		Simbol <i>offline connector</i> , menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
5.		Simbol <i>process</i> , menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.

6.		Simbol <i>decision</i> , yaitu menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban ya/tidak.
7.		Simbol <i>terminal</i> , yaitu menyatakan permulaan atau akhir suatu program.
8.		Simbol <i>predefined process</i> , yaitu menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
9.		Simbol <i>keying operation</i> , menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
10.		Simbol <i>offline-storage</i> , menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
11.		Simbol <i>manual input</i> , memasukkan data secara manual dengan menggunakan online keyboard.
12.		Simbol <i>input/output</i> , menyatakan proses masukkan atau keluaran tanpa tergantung jenis peralatannya.
13.		Simbol <i>magnetic tape</i> , menyatakan input berasal dari pita magnetis atau output disimpan ke pita magnetis.
14.		Simbol <i>disk storage</i> , menyatakan input berasal dari disk atau output disimpan ke disk.

15.		Simbol <i>document</i> , mencetak keluaran dalam bentuk dokumen.
16.		Simbol <i>display</i> , mencetak keluaran dalam layar monitor.