

BAB II DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian yang dilakukan oleh Raisa Tahseen Hasanat dan Tim yang berjudul “*An IoT based Real-Time Data-centric Monitoring System for Vaccine Cold Chain*” membahas mengenai sistem pemantauan data sentris secara *real time* berbasis *Internet of Things* untuk rantai dingin vaksin di negara berkembang dimana sistem yang dipantau adalah pemantauan suhu kelembaban dan sistem distribusi. Sistem rancang yang dibangun adalah sebuah sistem pemantauan distribusi vaksin yang dapat dilacak dan diawasi oleh admin saat aktif. Di dalam pembawa vaksin dilengkapi dengan sensor suhu dan kelembaban yang memonitor secara terus menerus lalu dikirim ke unit mikrokontroler yang mencatat suhu dan mengirimkan data secara berkala. Selain sensor suhu dan kelembaban, ditempatkan juga sensor LDR (*Light Defendent Resistor*) untuk menentukan apakah pembawa vaksin telah dibuka kapan saja selama perjalanan. Untuk pendeteksian lokasi menggunakan modul GSM-GPS-GPRS untuk memberikan informasi lokasi pembawa vaksin beserta penentuan tanggal dan waktu perjalanan yang dilakukan. Keluaran dari penelitian ini adalah sebuah aplikasi sistem *monitoring* berbasis rantai dingin vaksin [4].

Penelitian yang dilakukan oleh K.Salah dan Tim yang berjudul “*IoT enabled Shipping Container with Environmental Monitoring and Location Tracking*” membahas mengenai wadah kontainer berbasis IoT dengan pemantauan lingkungan dan pelacakan lokasi dimana arsitektur perangkat keras dan perangkat lunak yang berfungsi penuh untuk desain, implementasi dan *prototype* kerja dari sistem wadah kontainer pintar untuk pengiriman barang. Sistem dapat diakses melalui laman (seperti pada perangkat seluler atau *website*) untuk konfigurasi, interaksi pengguna dan pemantauan secara *real time*. Sistem yang di rancang dapat mengintegrasikan data sensorik seperti sensor DHT22, Sensor Vibrasi yang secara keseluruhan terintegrasi otomatis ke dalam *cloud* [5].

Penelitian yang dilakukan oleh Davit Nurhannavi dan Tim yang berjudul “Rancang bangun alat keamanan sepeda motor berbasis IoT menggunakan NodeMcu dan GPS” membahas mengenai sistem keamanan pada sepeda motor berbasis IoT

dimana peneliti menggunakan Nodemcu sebagai mikrokontroler dan GPS sebagai modul pelacak titik lokasi. *Prototype* yang diciptakan adalah kotak kontrol sistem yang terdiri dari 2 buah kotak yaitu yang pertama terdapat kotak untuk menyimpan piranti MiFi Huawei E5377 max dan kotak kedua digunakan sebagai penyimpanan komponen seperti Modul LM2596, NodeMCU, Modul GPS Ublok Neo 6M, Modul getar SW-420 dan Relay DC 4 *Channel*. Selain perancangan kotak kontrol, dilakukan juga perancangan perangkat lunak yang menggunakan *Blynk* sebagai *interface* pengguna [6].

Penelitian yang dilakukan oleh Arfend Atma Maulana Khalifa dan Kiki Prawiroredjo yang berjudul “Model Sistem Pengendalian Suhu dan Kelembaban ruangan produksi obat berbasis NodeMcu ESP32” membahas mengenai rancangan model sistem yang diterapkan pada ruangan produksi obat dengan menggunakan sensor suhu DHT11. Penelitian ini didasarkan pada aturan yang mana dalam proses produksi obat harus mengikuti anjuran suhu dan kelembaban ruangan yang telah ditetapkan. Dari hasil pengujian yang dilakukan dapat diketahui bahwa sistem dapat mengendalikan suhu dan kelembaban ruangan secara otomatis bila ada gangguan sekaligus mencatat data waktu kejadian gangguan untuk melengkapi laporan selama proses produksi obat [7].

Penelitian yang dilakukan oleh Afreen Mohsin dan Dr.Siva S Yellpalli yang berjudul “*IoT Based Cold Chain Logistics Monitoring*” membahas mengenai pemantauan distribusi yang membutuhkan suhu dingin atau biasa disebut rantai dingin (*cold chain*). Perancangan sistem pada alat ini menggunakan AMR7 LPC2148 sebagai mikrokontroler, GPS modul sebagai modul pelacakan lokasi, sensor kelembaban, sensor suhu (LM35). Penelitian ini menghasilkan sebuah tempat penyimpanan yang dapat menjaga kualitas *cold chain* [8].

2.2 DASAR TEORI

2.2.1 *Wireless-Fidelity* (WiFi)

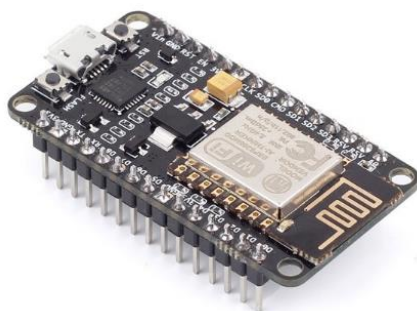
Wireless Fidelity atau yang biasa disingkat dengan WiFi merupakan sebuah teknologi komunikasi nirkabel yang memanfaatkan gelombang radio dalam rentang 2.4 GHz sampai dengan 5 GHz untuk menghubungkan dua perangkat atau lebih untuk dapat saling bertukar informasi. Teknologi WiFi telah banyak digunakan pada perangkat *mobile* seperti *Smartphone* dan Laptop hingga ke perangkat elektronik lainnya. WiFi adalah jaringan area *local* atau LAN (*Local Area Network*) yang tidak memerlukan kabel dengan koneksi kecepatan tinggi. WiFi sering juga disebut sebagai *Wireless Area Network* (WLAN). Untuk dapat melakukan akses internet, perangkat elektronik perlu berada dalam satu titik akses atau *hotspot* jaringan nirkabel sehingga terhubung dengan WiFi. Pada umumnya jaringan WiFi dapat menjangkau hingga 20 meter didalam ruangan dan lebih dari 20 meter di luar ruangan [8][9].



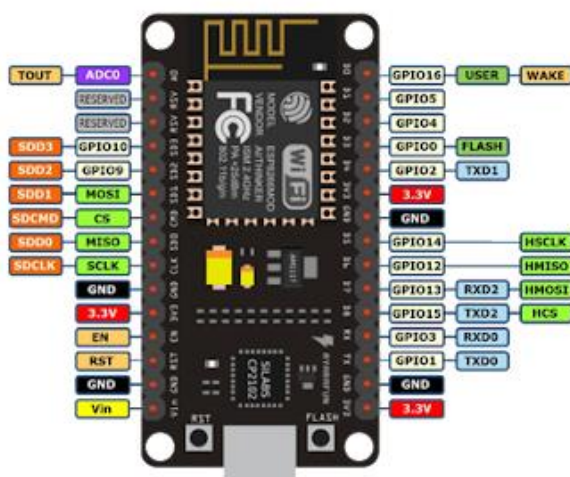
Gambar 2.1 *Wireless Fidelity* [23]

2.2.2 NodeMcu ESP8266 1.0

NodeMcu adalah sebuah *board* elektronik yang berbasis *chip* ESP8266 dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (WiFi). Pada modul ini terdapat beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah alat *monitoring* maupun *controlling* pada proyek IoT. NodeMcu ESP8266 membutuhkan daya sekitar 3.3 V dengan memiliki tiga mode WiFi yaitu *Station*, *Access Point* dan *Both* (keduanya). Sehingga modul ini dapat berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler [10].



Gambar 2.2 NodeMcu 1.0 [24]



Gambar 2.3 Pin Layout NodeMcu ESP8266 V2 [24]

Tabel 2.1 Tabel Spesifikasi NodeMcu ESP 8266 [10]

<i>Microcontroller</i>	Tensilica 32-bit RISC CPU Xtensa LX106
<i>Operating Voltage</i>	3.3V
<i>Input Voltage</i>	7-12V
<i>Digital I/O Pins (DIO)</i>	16
<i>Analog Input Pins (ADC)</i>	1
<i>UARTs</i>	1
<i>SPIs</i>	1
<i>I2Cs</i>	1
<i>Flash Memory</i>	4 MB
<i>SRAM</i>	64 KB
<i>Clock Speed</i>	80 MHz
<i>Antenna</i>	PCB Antenna

2.2.3 Modul GPS NEO6M V2

Modul ini digunakan sebagai perangkat untuk mendeteksi lokasi dengan menangkap dan memproses sinyal navigasi, sistem keamanan pada perangkat seluler, pengumpulan data lokasi dan lain sebagainya [11].



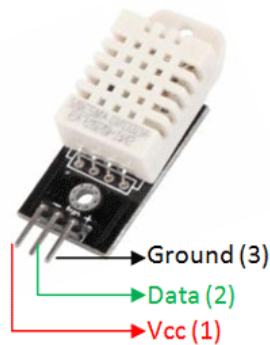
Gambar 2.4 GPS Neo6Mv2 [12]

Tabel 2.2 *Datasheet* performansi GPSNEO6MV2 [12]

<i>Maximum Navigation update rate</i>	5HZ
<i>Default baud rate</i>	9600 bps
<i>Sensitivity</i>	-160 dBm
<i>Supply Voltage</i>	3.6 V
<i>Maximum DC Current at any output</i>	10mA
<i>Operation limits</i>	<i>Gravity-4g, Altitude-50000m, Velocity-500m/s</i>
<i>Operating temperature range</i>	-40°C to 85°C
<i>Time-To-First Fix</i>	<i>Cold start 32s, Warm Start 23s, Hot Start <1s</i>
<i>Standalone GPS Receiver</i>	
<i>Anti-jamming technology</i>	
<i>UART Interface at the output pins</i>	
<i>EEPROM with battery backup</i>	

2.2.4 DHT 22

DHT22 adalah sensor digital kelembaban dan suhu relatif. Sensor DHT22 menggunakan kapasitor dan *thermistor* untuk mengukur udara disekitarnya dan keluar sinyal pada pin data. DHT22 diklaim memiliki kualitas pembacaan yang baik, dinilai dari respon proses akuisisi data yang cepat dan ukurannya yang minimalis. Sensor DHT22 sangat mudah diaplikasikan pada mikrokontroler tipe *Arduino* karena memiliki tingkat stabilitas yang dapat dipercaya dan fitur kalibrasi yang memiliki hasil sangat akurat. Salah satu jenis *Arduino* adalah *Arduino Uno* [13].



Gambar 2.5 Sensor DHT 22 [14]

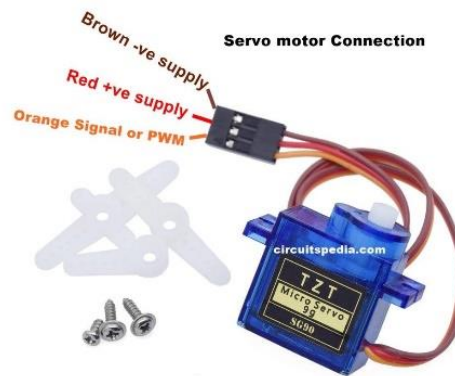
Tabel 2.1 *Datasheet* DHT 22 [14].

<i>Model</i>	DHT 22
<i>Power Supply</i>	3.3 – 6V DC
<i>Output Signal</i>	<i>Digital Signal via Single Bus</i>
<i>Sensing Element</i>	<i>Polymeter Capacitor</i>
<i>Operating Current</i>	0.3 mA (<i>measuring</i>) 60uA (<i>standby</i>)
<i>Accuracy</i>	+0.5°C and +-1%
<i>Resolution</i>	<i>Temperature and Humidity both are 16-bit</i>
<i>Repeatability</i>	<i>Humidity +-1%RH;Temperature +-0.2 Celcius</i>
<i>Humidity Range</i>	0% to 100%
<i>Temperature Range</i>	-40°C to 80°C

<i>Sensing Period</i>	<i>Average: 2s</i>
<i>Interchangeability</i>	<i>Fully Interchangeable</i>
<i>Dimensions</i>	<i>Small size 14*18*5.5 mm; Big Size 22*28*5 mm</i>

2.2.5 Micro Servo SG90

Micro servo (motor servo) adalah sebuah perangkat yang dirancang dengan sistem kontrol umpan balik *loop* (tertutup), sehingga dapat di *set-up* atau diatur untuk menentukan dan memastikan posisi sudut dari poros output motor. *Micro servo* merupakan perangkat yang terdiri dari motor DC, serangkaian *gear*, rangkaian kontrol dan potensiometer [15].



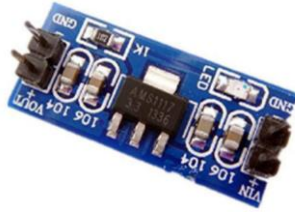
Gambar 2.6 *Micro Servo SG90* [25]

Tabel 2.2 *Datasheet Micro Servo* [16]

<i>Operating Voltage</i>	+5V
<i>Torque</i>	2.5kg/cm
<i>Operating speed</i>	0.1s/60°
<i>Gear Type</i>	<i>Plastic</i>
<i>Rotation</i>	0° - 180°
<i>Weight of motor</i>	9gm
<i>Package includes</i>	<i>Gear horns and screws</i>

2.2.6 AMS1117

Modul AMS1117 merupakan modul *regulator* yang berfungsi sebagai penurun tegangan DC ke DC yang terdiri dari beberapa level regulasi yaitu 2.85V, 3.3V, dan 5V. Dengan kata lain, *output* dari modul *regulator* ini bersifat *fixed* atau tetap. Disebut modul karena AMS1117 telah dikemas dengan beberapa komponen pendukung lainnya dalam satu *board* [17].



Gambar 2.7 AMS1117 [17]

2.2.7 Arduino IDE

IDE adalah kependekan dari *Integrated Development Environment* atau jika dijelaskan menjadi sebuah lingkungan terintegrasi yang dipergunakan untuk melakukan pengembangan. *Arduino* menggunakan bahas pemrograman sendiri yang menyerupai Bahasa C. *Arduino* IDE dibuat dari Bahasa JAVA yang dilengkapi dengan library C/C++ yang biasa disebut *wiring* yang membuat operasi *Input* dan *Output* menjadi lebih mudah [18].



Gambar 2.8 Logo Arduino IDE [26]

2.2.8 Blynk

Blynk merupakan *platform* sistem operasi iOS maupun android sebagai kendali pada modul *Arduino*, Raspberry Pi, ESP8266 dan perangkat sejenis lainnya melalui internet. Aplikasi ini sangat mudah digunakan dan memiliki banyak fitur yang memudahkan pengguna dalam memakainya. Aplikasi ini bisa menggunakan banyak *board* yang dapat disesuaikan dengan kebutuhan masing-masing. Selain mudah

digunakan, *Blynk* juga bisa digunakan sebagai *controlling* untuk perangkat *Internet of Things* dimana pengguna dapat menggunakan *button-button* yang sudah tersedia di *Blynk*. Dalam menghubungkannya, pengguna harus menggunakan *virtual pin* yang harus di masukan di *Arduino IDE* dan di *Blynk* agar komunikasi dapat tersambung. Dalam *Blynk* dikenal juga “Autentikasi Token”. Token ini didapatkan ketika pengguna merancang sistem di *Blynk*, dan token tersebut harus dimasukan ke *Arduino IDE* sebagai identitas dari sistem yang sudah di buat di *Blynk*. Selama *Blynk* dan Perangkat terkoneksi ke internet, maka *Blynk* akan menerima masukan data dari sensor-sensor yang digunakan dalam perangkat [19].



Gambar 2.9 Logo *Blynk* [27]

2.2.9 *Firebase*

Firebase merupakan suatu layanan dari Google yang dapat memberikan kemudahan bagi para *developer* aplikasi dalam mengembangkan aplikasinya.

Firebase terdiri dari berbagai jenis atau fitur yang diantaranya :

- a. *Firebase Analytics* digunakan sebagai koleksi data dan *reporting* untuk aplikasi Android maupun iOS dengan koleksi data yang dapat bervariasi.
- b. *Firebase Cloud Messaging and Notifications* menyediakan koneksi yang handal dan hemat baterai antar *server* maupun *device* .
- c. *Firebase Authentication* digunakan untuk mengautentikasi pengguna ke aplikasi yang dibuat oleh seorang *developer*. *Firebase Authentication* mendukung autentikasi menggunakan nomor telepon, sandi, penyedia identitas gabungan dll.
- d. *Firebase Cloud Firestore* adalah *database* yang bersifat fleksibel dan terukur untuk pengembangan perangkat. *Cloud Firestore* merupakan *database* NoSQL yang di *hosting* di *cloud* dan dapat diakses melalui SDK *real* oleh aplikasi Ios, Android dan Web.

- e. *Firestore Real Time Database* adalah *database* yang di host melalui *cloud*. Data disimpan dan di eksekusi dalam bentuk JSON dan disinkronkan secara *Real Time* ke tiap *user* yang terkoneksi.
- f. *Firestore Hosting* digunakan untuk menayangkan konten melalui koneksi yang begitu aman, mengirimkan konten secara cepat dan mendukung semua jenis konten untuk di hosting seperti API.



Gambar 2.10 Logo *Firestore* [28]