

Penggunaan Node-RED pada Sistem Monitoring dan Kontrol Green House berbasis Protokol MQTT

Sri Mulyono, Muhammad Qomaruddin, Muhammad Syaiful Anwar
Jurusan Teknik Informatika, Universitas Islam Sultan Agung

Correspondence Author: sri.m@unissula.ac.id

Abstrak

Sistem *Internet of Things* (IoT) memungkinkan pengguna untuk mencapai lebih dalam terhadap otomatisasi, analisis, dan integrasi sistem serta meningkatkan jangkauan daerah dan akurasi sistem, IoT memanfaatkan teknologi penginderaan, jaringan dan robotika. Pembangunan *green house* belum sepenuhnya sesuai dengan iklim dimana *green house* tersebut dibuat atau dibangun, sehingga harapan pemenuhan kuantitas, kualitas dan kontinuitas produksi belum terealisasi dengan baik/optimal, oleh karenanya upaya-upaya dalam perbaikan kualitas sebuah *green house* sangat diperlukan, salah satunya adalah peningkatan kontrol pada *green house* itu sendiri seperti *monitoring* and *controlling*, kontrol yang dimaksudkan seperti suhu udara, kelembaban udara, kelembaban tanah dan kontrol distribusi air. Oleh karena itu melalui penelitian ini akan dibuat sebuah sistem penyiram tanaman yang mampu mengkondisikan suhu udara, kelembaban tanah dan kelembaban udara pada tanaman secara otomatis, sistem tersebut akan dibangun berbasis IoT menggunakan Node-RED dengan protokol MQTT, metode penelitian dengan urutan proses pengambilan data, perancangan permodelan, pembuatan alat, pembuatan sistem, kalibrasi alat dan hasil analisa data, sehingga sistem dapat dimonitor dan dikontrol secara realtime. Dengan adanya sistem automasi berbasis IoT dengan protokol MQTT pada penyiram tanaman maka dapat meningkatkan efisiensi waktu, tenaga dan energi, serta mengoptimalkan proses perkembangan dan pertumbuhan tanaman.

Keyword: *Internet of Things*, penyiraman, *real time*, MQTT, *Green House*

1. PENDAHULUAN

Green house merupakan sebuah bangunan kontruksi yang berfungsi untuk menghindari atau memanipulasi kondisi lingkungan agar tercipta kondisi lingkungan yang dikehendaki dalam pemeliharaan tanaman, seiring berkembangnya agribisnis dan pendukung bidang pertanian lainnya peranan *green house* sangat dibutuhkan, hal ini dilakukan dalam rangka meningkatkan kualitas hasil panen. Namun pembangunan *green house* belum sepenuhnya sesuai dengan keadaan iklim yang cenderung sulit untuk diprediksi secara langsung, sehingga harapan pemenuhan kuantitas, kualitas dan kontinuitas produksi belum terealisasi dengan baik atau optimal, oleh karenanya upaya-upaya dalam perbaikan kualitas sebuah *green house* sangat diperlukan.

Perkembangan teknologi digital yang serba modern ini menuntut masyarakat untuk lebih cenderung mengikuti proses perkembangannya dan beradaptasi guna menuju masyarakat yang kreatif, inovatif, dan mandiri serta mampu memanfaatkan iptek dan sumber daya lokal untuk menghasilkan produk berdaya saing tinggi. Salah satu upaya dalam hal ini yaitu dengan memanfaatkan teknologi *internet of things* (IoT) untuk membantu proses perawatan pada tanaman di dalam *green house*, padahal dengan menggunakan *internet of things* dapat mempermudah petani dalam hal *monitoring* dan *controlling*, *monitoring* yang dimaksudkan antara lain suhu udara, kelembaban tanah dan kelembaban udara serta *controlling* distribusi air dan pupuk.

Dulu, alat-alat komunikasi elektronik yang hanya dapat dipakai untuk menelepon dan mengirim pesan melalui SMS. Kini, *handphone* sudah semakin maju sehingga dikenal sebagai telepon pintar atau *smartphone*. Semua itu tidak terlepas dari perkembangan *mobile application* yang disematkan dalam *handphone* itu sendiri. Kemudian, teknologi komputer juga berkembang begitu pesat, banyak perusahaan besar memanfaatkan komputer untuk pengolahan data, administrasi hingga operasional sistem produksi.

Oleh karena itu, pada penelitian ini penulis akan membuat sistem perawatan tanaman dengan konsep teknologi *internet of things* menggunakan Node-RED dan protokol MQTT, sehingga informasi pada tanaman *green house* dapat secara langsung diterima dalam sistem aplikasi *smartphone* dan web, disisi lain petani bisa mengukur, memonitor dan mendeteksi dari dini kekurangan komponen-komponen utama dalam *green house*. Sehingga secara efisien, petani bisa memantau perkembangan tanamannya, mengelola dan mengontrol kebutuhan energi yang digunakan secara *real time*.

2. TINJAUAN PUSTAKA

2.1. *Internet of Things (IoT)*

Istilah "*Internet of Things*" (IoT) pertama kali digunakan di 1999 oleh teknologi inggris pelopor kevin ashton untuk menggambarkan sebuah sistem di mana objek dalam dunia fisik dapat terhubung ke internet oleh sensor.

Ashton menciptakan istilah tersebut untuk menggambarkan kekuatan penghubung *Radio-Frequency Identification (RFID)* digunakan dalam rantai pasokan perusahaan ke internet untuk menghitung dan melacak jumlah barang tanpa membutuhkan campur tangan manusia, *Internet of Things* telah menjadi istilah populer untuk menggambarkan scenario di mana konektivitas internet dan kemampuan komputasi meluas ke berbagai benda, perangkat, sensor, dan item sehari-hari.

Banyak implemensi menggunakan model komunikasi teknis lainnya , masing-masing dengan karakteristik tersendiri .Empat model komunikasi umum yang dideskripsikan oleh *Internet Architecture Board* include yaitu : *device-to-device arsitektur* , *device-to-cloud* , *device-to-gateway* , dan *back-end data-sharing*. Model ini menyoroti fleksibilitas dengan cara perangkat IO dapat terhubung kemudian memberikan nilai kepada pengguna.

Sistem IoT memungkinkan pengguna untuk mencapai lebih dalam terhadap otomatisasi, analisis, dan integrasi sistem. Mereka yang ingin meningkatkan jangkauan daerah dan akurasi mereka. IoT memanfaatkan teknologi penginderaan, Jaringan dan Robotika baru dan yang sudah ada [1].

2.2. WeMos D1 R2 WiFi Uno Based ESP8266 Shield

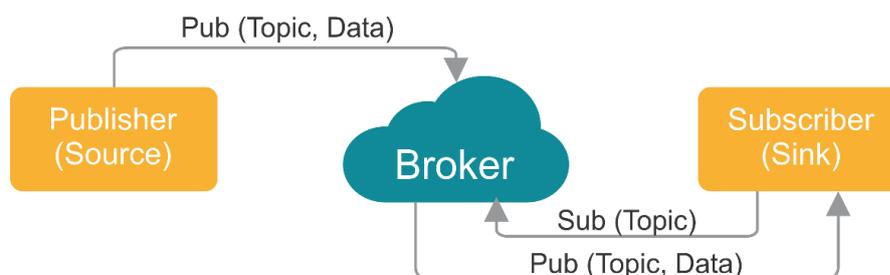
WeMos D1 R2 merupakan salah satu arduino *compatible development board* yang dirancang khusus untuk keperluan IoT. Wemos menggunakan chip SoC WiFi yang cukup terkenal saat ini yaitu ESP8266. Cukup banyak modul WiFi yang menggunakan SoC ESP8266. Namun Wemos memiliki beberapa kelebihan tersendiri yang menurut saya sangat cocok digunakan untuk Aplikasi IoT.

Beberapa alasan yang membuat Wemos cukup menarik untuk di oprek adalah sebagai berikut [2]:

1. Arduino compatible, artinya dapat diprogram menggunakan Arduino IDE dengan *sintaks* program dan *library* yang banyak terdapat di internet.
2. *Pinout* yang *compatible* dengan Arduino uno, Wemos D1 R2 merupakan salah satu produk yang memiliki bentuk dan *pinout* standar seperti arduino uno. Sehingga memudahkan kita untuk menghubungkan dengan arduino shield lainnya.
3. Wemos dapat *running stand alone* tanpa perlu dihubungkan dengan mikrokontroler. Berbeda dengan modul WiFi lain yang masih membutuhkan mikrokontroler sebagai pengontrol, Wemos dapat *running stand alone* karena didalamnya sudah terdapat CPU yang dapat diprogram melalui *Serial port* ataupun via OTA (*Over The Air*) atau transfer program secara wireless.
4. *High Frequency CPU*, dengan *processor* utama 32bit berkecepatan 80MHz Wemos dapat mengeksekusi program lebih cepat dibandingn dibandingkan mikrokontroler 8 bit yang digunakan di Arduino.
5. Dukungan *High Level Language*, Selain menggunakan Arduino IDE Wemos juga dapat diprogram menggunakan bahasa Python dan Lua. Sehingga memudahkan bagi *network programmer* yang belum terbiasa menggunakan Arduino.

2.3. MQTT

MQTT atau *Message Queuing Telemetry Transport* merupakan protokol transport dengan sifat *clientserverpublish/subscribe*. MQTT merupakan protokol *transport* dengan karakteristik sederhana, terbuka dan ringan yang dirancang agar mudah diimplementasikan. Sehingga MQTT dapat digunakan di banyak situasi, termasuk penggunaannya dalam komunikasi *machine-to-machine (M2M)* dan *Internet of Things (IoT)*. Protokol MQTT berjalan dengan menggunakan TCP/IP. Sehingga protokol ini membutuhkan transportasi guna menjalankan perintah MQTT, *bytestream* dari *client to server* atau *server to client* [3], berikut gambaran bagaimana protokol MQTT bekerja.



Gambar 1. MQTT Real Protocol

Keterangan :

1. *Topic* (UTF-8) merupakan kanal untuk melakukan subscribe (klien) yang juga berfungsi sebagai *filter* untuk *broker* dalam mengirimkan pesan ke setiap klien.
2. *Broker (cloud)* berfungsi untuk menangani *publish* dan *subscribe* data agar sumber pengirim data (*publisher*) dan penerima data (klien) tidak saling mengetahui (*space decoupling*).

Pada gambar 1 terdapat dua tipe klien yaitu *publisher* dan *subscriber* yang mana kedua klien tersebut dapat saling terhubung dengan sebuah *topic* tertentu melalui *broker*.

2.4. Node-RED

Node-RED adalah sebuah *tool* berbasis *browser* untuk membuat aplikasi *Internet of Things (IoT)* yang mana lingkungan pemrograman visualnya mempermudah penggunaannya untuk membuat aplikasi sebagai “*flow*”.

Lanskap bahasa pemrograman sangatlah luas dan meliputi berbagai jenis gaya dan paradigma pemrograman. Bahasa imperatif berorientasi objek saat ini menguasai dunia pemrograman, namun begitu sebetulnya ada alternatif untuk pengembangan atau produksi software dan juga untuk membuat prototipe ide dengan cepat.

Node-RED mengambil jalur alternatif tersebut untuk pengembangan software. Pertama, ia adalah bahasa pemrograman visual. Daripada membuat aplikasi sebagai barisan kodingan, Node-RED fokus ke program sebagai *flow*.



Gambar 2 Contoh *flow* di dalam lingkungan pengembangan Node-RED

Flow ini terbentuk dari node-node yang saling berhubungan di mana tiap node melakukan tugas tertentu (lihat Gambar 1). Walaupun Node-RED didesain untuk *Internet of Things (IoT)*, ia juga dapat digunakan untuk keperluan umum dan untuk berbagai macam jenis aplikasi [4].

Node-RED menyediakan berbagai jenis *node* yang dapat membuat developer langsung menjadi produktif, seperti:

1. Menampilkan *input node* dan *output node* yang mana mengizinkan subskripsi dan tanda terima dari topik *MQ Telemetry Transport (MQTT)* dan keluaran dari topik *MQTT* ke sebuah *broker*
2. Mengembangkan layanan web melalui permintaan *HTTP* (beserta pembuatan balasan *HTTP*); dan *TCP* level rendah dan layanan *User Datagram Protocol* yang dapat membuat server, menerima input, dan menghasilkan output.

Membuat *node* dengan fungsi tersendiri (dalam *JavaScript*), menghasilkan pesan dengan pemicunya berbasis waktu, dan menunda pesan untuk menilai batas *flow*.

2.5. Green House

Green house merupakan sebuah bangunan yang berkerangka atau dibentuk menggelembung, diselubungi bahan bening atau tembus cahaya yang dapat meneruskan cahaya secara optimum untuk produksi dan melindungi tanaman dari kondisi iklim yang merugikan bagi pertumbuhan tanaman.

Tujuan dalam pembuatan *green house* pada pertanian adalah untuk mempertahankan agar tanaman-tanaman yang ada di dalamnya dapat tetap hidup (tidak mati), kaca dalam *green house* akan menghalangi keluarnya panas matahari yang masuk di dalamnya. Mekanisme terperangkapnya panas matahari akibat peningkatan konsentrasi gas rumah kaca di atmosfer itulah yang disebut dengan efek rumah kaca [5].

3. METODE PENELITIAN

3.1. Desain Penelitian

Desain penelitian adalah kerangka kerja yang digunakan untuk melaksanakan riset pemasaran. Desain penelitian memberikan prosedur untuk mendapatkan informasi yang diperlukan untuk menyusun atau menyelesaikan masalah dalam penelitian. Desain penelitian merupakan dasar dalam melakukan penelitian. Oleh sebab itu, desain penelitian yang baik akan menghasilkan penelitian yang efektif dan efisien. Tujuan dalam perancangan desain penelitian ini yaitu untuk menyelidiki suatu masalah atau situasi untuk mendapatkan pengetahuan dan pemahaman yang baik. Disisi lain, desain penelitian ini juga bertujuan untuk menggambarkan suatu pokok masalah yang akan dianalisa, sehingga proses dalam perancangan sistem mampu menggambarkan pernyataan yang jelas mengenai permasalahan yang dihadapi secara hipotesis, spesifik mencakup informasi detail yang dibutuhkan.

Pada perancangan desain penelitian dapat digambarkan dalam bentuk desain penelitian pada gambar 3 di bawah ini.



Gambar 3. Desain Penelitian

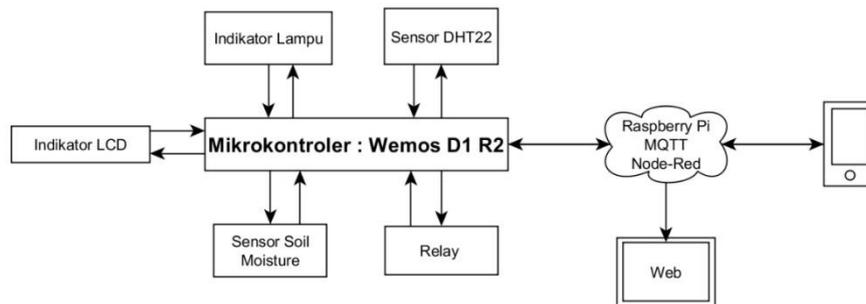
3.2. Desain Sistem

Sistem monitoring dan kontrol pada *green house* terdiri dari beberapa komponen yaitu:

1. Wemos D1 R2 WiFi Uno Based ESP8266 Shield
2. Sensor *humidity* khusus udara.
3. Sensor *humidity* khusus Tanah.
4. *Relay*.
5. *Cloud* atau server Raspbery Pi 3.
6. *Androiddevice*
7. *Web Client php mysql*

Pengendali atau kontrol bekerja dengan menggunakan sistem *mikrokontroller*, pada modul wemos D1 R2 uno based terdapat sebuah *chip* mikrokontroler ESP8266 yang berperan sebagai akses internet untuk mengirim dan menerima data pada sensor DHT22 udara serta sensor kelembaban tanah untuk kemudian dihubungkan ke *cloud* sebagai server Raspbery pi 3 yang nantinya dapat digunakan sebagai lalu lintas penyimpanan data dan pengelolaan informasi.

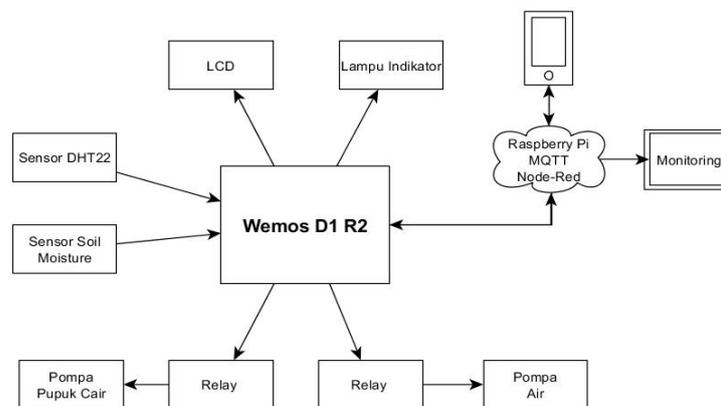
Dalam hal ini, maka pengguna dapat memantau kondisi serta mengendalikan proses sistem penyiraman pada tanaman *green house* melalui aplikasi android yang terintegrasi dengan *cloud* dengan bantuan jaringan koneksi internet, sehingga nantinya jika terdapat perubahan pada kondisi suhu serta kelembaban pada ruang *green house* pengguna bisa dapat mengetahui secara *realtime* melalui aplikasi Blynk dengan sistem operasi android dan web php MySQL, berikut adalah blok dari alur diagram desain sistem pada gambar 4.



Gambar 4. Desain Sistem

3.3. Desain Alur Sistem

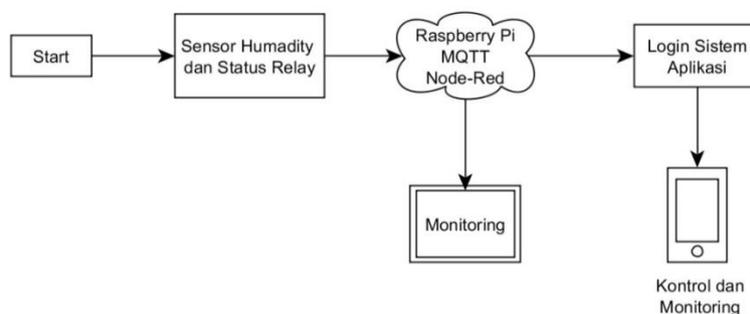
Pertama pada perangkat wemos d1 r2 menjadi pusat kontrol dan *monitoring* pada *greenhouse* dimana perangkat terkoneksi dengan sensor dht22 dan *soil moisture* serta dua perangkat *relay* yang saling terhubung dengan kabel dimana sumber energi listrik menggunakan *supply* dari adaptor, sedangkan proses perintah serta pengiriman informasi menggunakan server (*cloud*) Raspberry Pi 3 yang terhubung dengan perangkat aplikasi Blynk dan web php mysql dengan bantuan protokol MQTT seperti yang terlihat pada gambar 4.



Gambar 4. Desain Alur Sistem

3.4. Monitor

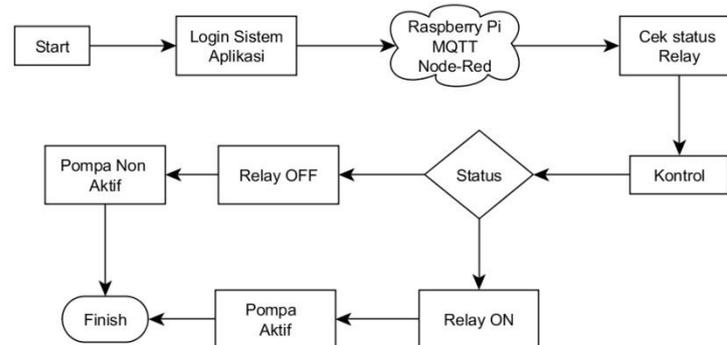
Pada gambar 5 menggambarkan proses pengiriman data suhu dan kelembaban dimana proses pengiriman data diolah secara *realtime* kemudian nilai diteruskan ke *cloud* atau *server* raspberry pi 3 untuk kemudian dihubungkan ke sistem aplikasi menggunakan *tool* Node-RED, sehingga pengguna dapat mengontrol dan memonitor melalui aplikasi *mobile* dan web tanpa ada batasan waktu.

Gambar 5. Blok Diagram *Monitoring*

3.5. Kendali

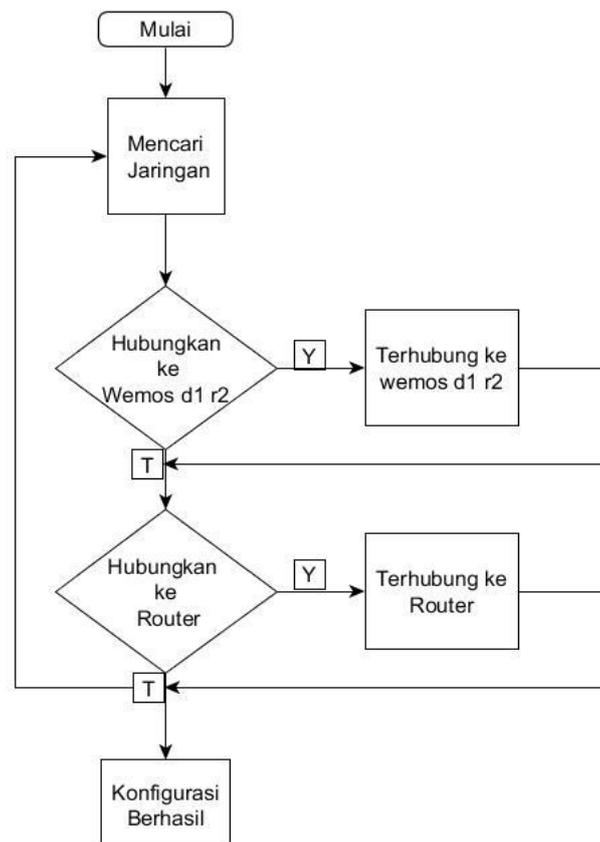
Pada gambar 6 dijelaskan bahwa sistem kendali atau kontrol pada perangkat ini diawali dengan membuka aplikasi dan mengecek apakah jaringan Sudah terhubung dengan peangkat dan sistem, kemudian pengguna dapat mengecek status kondisi dalam *greenhouse*.

Setelah pengguna mengetahui kondisi suhu dan kelembaban pada *green house*, pengguna dapat menghidupkan (on) atau mematikan (off) penyiraman dan pemupukan tanaman melalui antarmuka sistem yang terhubung dengan kontroler. Proses ini berjalan ketika data yang diterima oleh kontroler diteruskannya ke *relay*, dengan asumsi jika nilai diterima sama dengan on/1 maka pompa hidup, sedangkan jika data yang diterima sama dengan off/0 maka pompa mati, dalam penelitian ini juga menggunakan model *timer* pada *relay*, sehingga *relay* akan mati secara otomatis selama 10 detik.



Gambar 6. Blok Diagram Proses Kendali

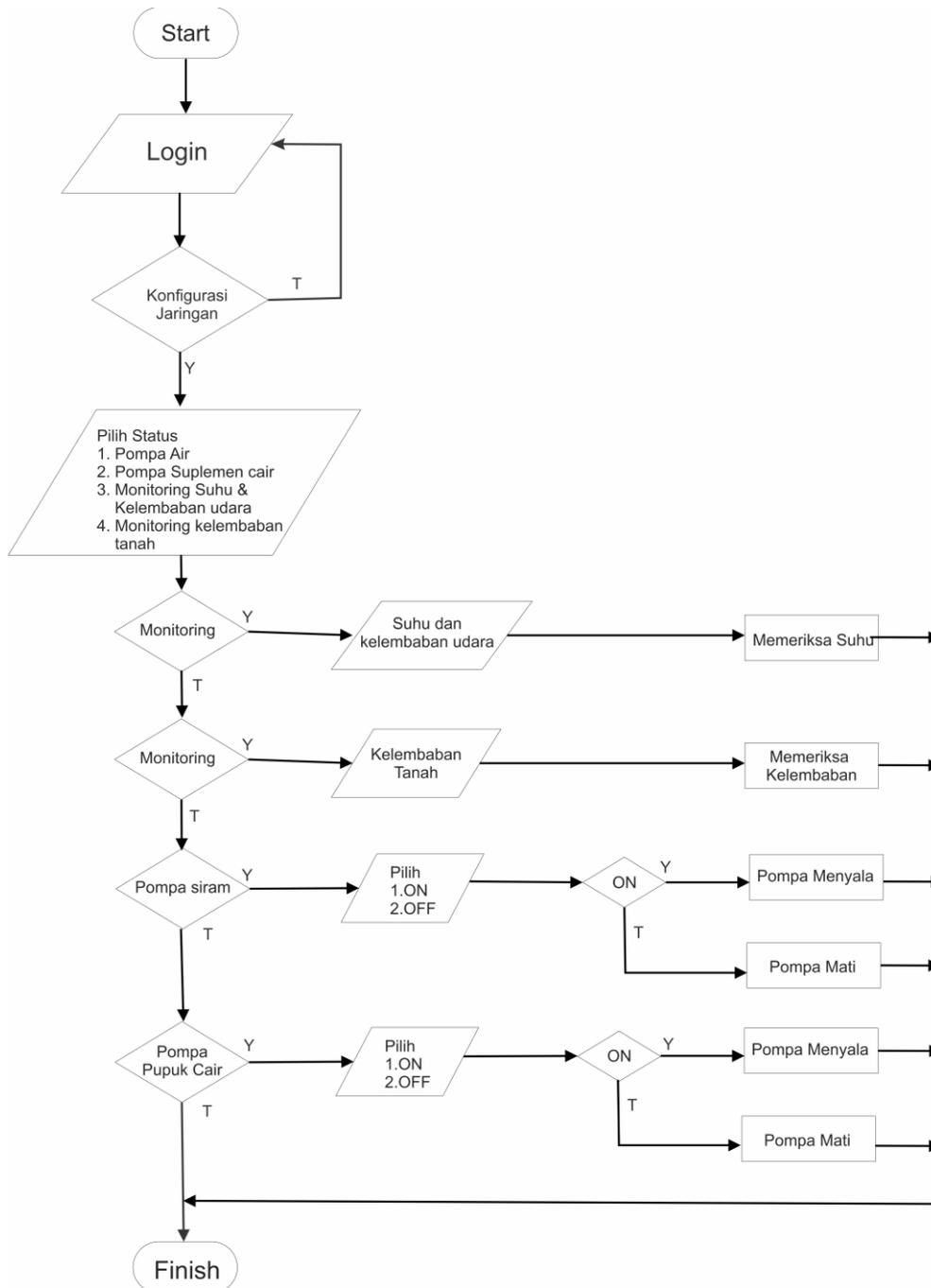
3.6. Flowchart Konfigurasi Wemos D1 D2



Gambar 7. Proses Konfigurasi Wemos d1 r2

Pada gambar 7 merupakan proses konfigurasi yang menggambarkan rangkaian proses koneksi jaringan, dimana terdapat konfigurasi wifi dari wemos d1 r2 jika konfigurasi jaringan wemos berhasil maka tahap berikutnya adalah proses konfigurasi koneksi ke router, dimana dalam router sudah terdapat server mini yaitu Raspberry Pi 3, jika konfigurasi semua telah berhasil maka sistem secara otomatis terkoneksi dengan perangkat, jika proses konfigurasi tidak berhasil maka perlu untuk dilakukan proses konfigurasi dari awal.

3.7. Flowchart Sistem IoT



Gambar 8. Flowchart Sistem IoT

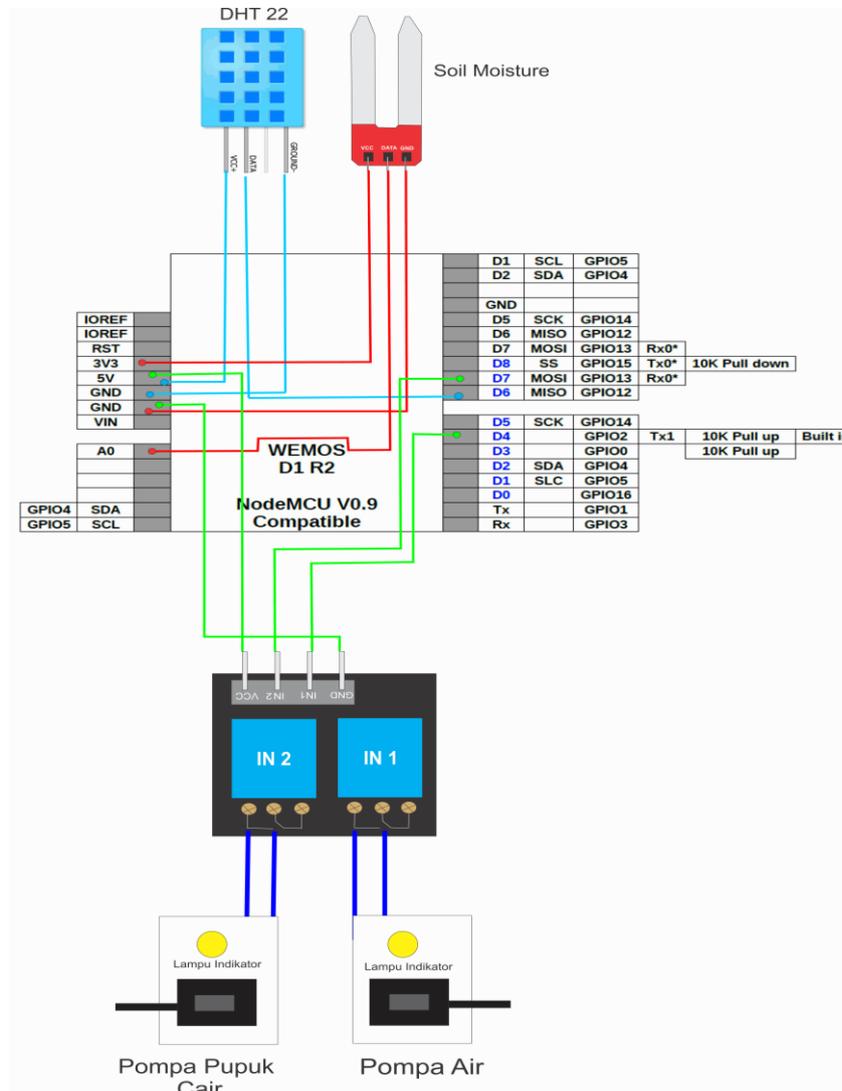
Pada gambar 8. menggambarkan alur *flowchart* sistem yang terdiri dari 3 proses dan 8 entitas, diantaranya yaitu:

1. Proses *client* pada sistem aplikasi, pada proses ini *user* dapat melakukan *request* data - data untuk proses kontrol dan *monitoring* pada *greenhouse* kemudian sistem merespon dengan mengirimkan data-data yang diperlukan *user*, sehingga pada dasarnya proses pada sistem aplikasi dilakukan oleh *client* atau *user*.
2. Proses *server*, proses ini merupakan lalu lintas pertukaran informasi yang tersimpan pada *database* pada perangkat Raspberry Pi 3 dengan menggunakan *tool* Node-RED untuk kemudian dikirimkan berupa data-data yang diperlukan *client* atau *user*.

3. Proses *client* kontrol dan monitoring suhu udara serta kelembaban tanah, proses ini merupakan proses inti dan paling banyak melakukan proses entitas dan dilakukan juga proses pengontrolan dan pengambilan data-data yang nantinya diolah didalam sistem dan perangkat untuk kemudian diteruskan ke *server*.

4. HASIL DAN ANALISA

4.1. Rangkaian Perancangan Perangkat Wemos D1 R2

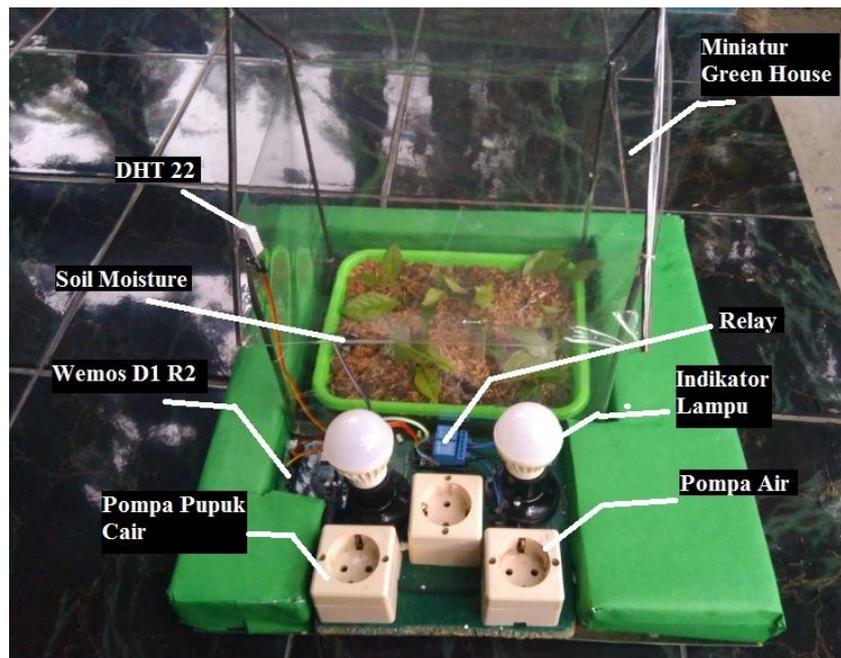


Gambar 9. Rangkaian Perancangan Perangkat

Pada gambar 9 menjelaskan mengenai alur dan desain dalam perancangan komponen-komponen yang nantinya saling terhubung menjadi sebuah perangkat piranti pintar yang terintegrasi melalui protokol MQTT.

4.2. Media Simulasi

Simulasi pada penelitian ini menggunakan prototipe miniatur *green house* sebagai gambaran atau pengganti *green house* yang asli seperti pada *green house* di UNISSULA. Simulasi dibuat agar semirip mungkin dengan aslinya sehingga akan memberikan gambaran mengenai bagaimana peneliti melakukan pengujian seolah seperti petani melakukan aktivitas perawatan tanaman di dalam *greenhouse*, gambar 10 Media Simulasi miniatur *green house*.



Gambar 10. Media Simulasi

4.3. Proses Integrasi dan Pengkodean

Proses integrasi perangkat lunak atau pengkodean pada protipe ini terbagi menjadi tiga bagian, yaitu pengkodean terhadap *server*, mikrokontroler, dan aplikasi.

1. Server

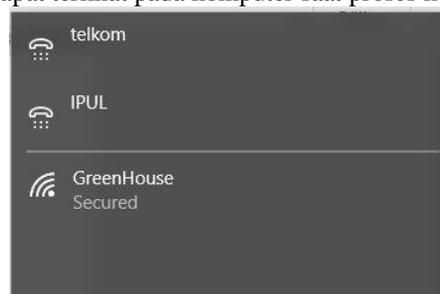
Proses konfigurasi server agar dapat terhubung dengan jaringan wifi ESP8266 pada wemos d1 r2, perangkat elektronik jenis wifi module dengan output serial (TTL) dilengkapi dengan beberapa GPIO (*General Purpose Input/Output Interface*), agar jaringan wifi ESP8266 dapat terlihat, terlebih dahulu dalam perangkat wemos dilakukan pemrograman konfigurasi jaringan koneksi seperti gambar 11:

```
//fetches ssid and pass and tries to connect
//if it does not connect it starts an access point with the specified name
//and goes into a blocking loop awaiting configuration
if (!wifiManager.autoConnect("GreenHouse", "ipulsukses")) {
  Serial.println("failed to connect and hit timeout");
  delay(3000);
  //reset and try again, or maybe put it to deep sleep
  ESP.restart();
  delay(5000);
}

//if you get here you have connected to the WiFi
Serial.println("connected...yeey :)");
```

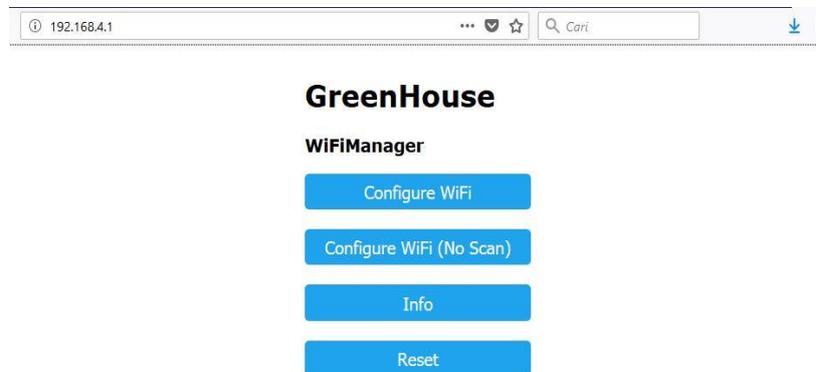
Gambar 11. Konfigurasi Koneksi

Dari sini maka wifi module dapat terlihat pada komputer saat proses konfigurasi akan dilakukan.



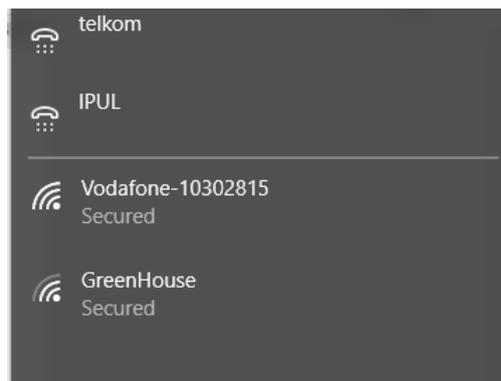
Gambar 12. Tampilan Jaringan GreenHouse

Setelah berhasil terkoneksi dengan jaringan “GreenHouse” seperti pada gambar 4. selanjutnya user secara otomatis akan membuka browser untuk dilakukan proses konfigurasi wifi seperti pada gambar 13 berikut:



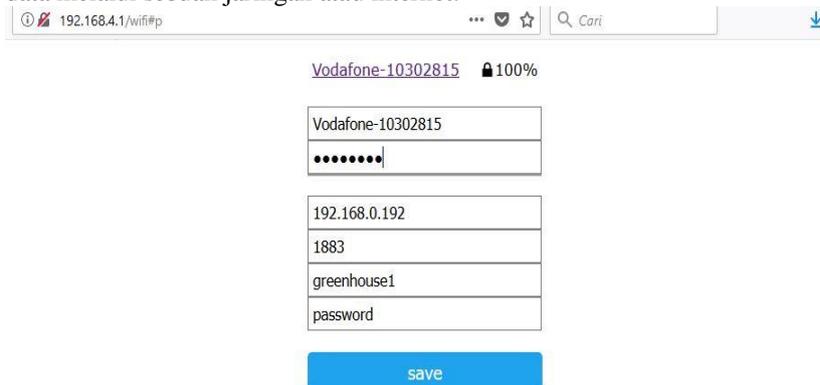
Gambar 13. Tampilan Konfigurasi

Konfigurasi dilakukan dengan meng-klik menu *Configure Wifi* secara otomatis akan masuk ke tahap akhir konfigurasi.



Gambar 14. Tampilan Jaringan Vodafone

Sebelum konfigurasi selesai, tahapan berikutnya yaitu dengan memasang router sebagai perangkat untuk mengirimkan paket data melalui sebuah jaringan atau internet.



Gambar 15. Konfigurasi Vodafone

Hasil akhir konfigurasi wemos d1 r2 yaitu dengan cara meng-klik jaringan Vodafone dengan disertai *password* dan IP yang sudah terprogram dan *dicompile* dalam perangkat wemos d1 r2.

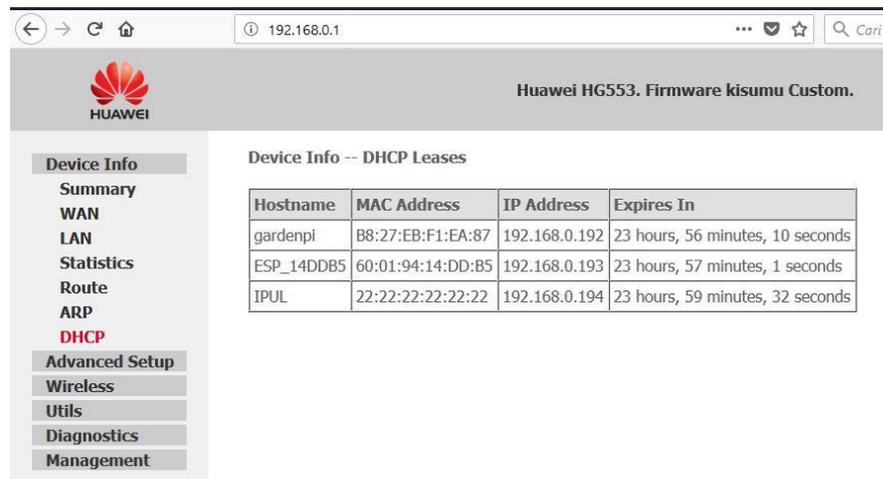
```

mounting FS...
mounted file system
reading config file
opened config file
{"mqtt_server":"192.168.0.192","mqtt_port":"1883","mqtt_user":"greenhouse1","mqtt_pass":"password"}
parsed json
*WM: Adding parameter

```

Gambar 16. Tampilan Compile Pada Arduino IDE

Sehingga akan didapat satu jaringan yang saling terhubung dalam router Vodafone. Setelah konfigurasi server telah selesai maka secara otomatis akan membuka menu browser tampilan informasi koneksitas router, disini dapat terlihat informasi beberapa jaringan yang telah terkoneksi serta alamat IP, seperti pada gambar 17:

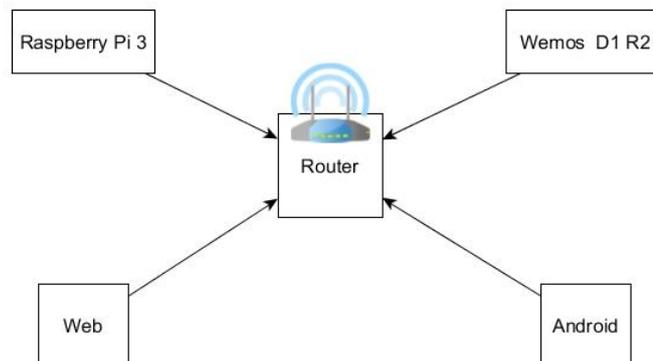


The screenshot shows the Huawei HG553 router's web interface. The browser address bar displays '192.168.0.1'. The page title is 'Huawei HG553. Firmware kisumu Custom.'. On the left, there is a navigation menu with options: Device Info (selected), Summary, WAN, LAN, Statistics, Route, ARP, DHCP, Advanced Setup, Wireless, Utils, Diagnostics, and Management. The main content area is titled 'Device Info -- DHCP Leases' and contains a table with the following data:

Hostname	MAC Address	IP Address	Expires In
gardenpi	B8:27:EB:F1:EA:87	192.168.0.192	23 hours, 56 minutes, 10 seconds
ESP_14DDB5	60:01:94:14:DD:B5	192.168.0.193	23 hours, 57 minutes, 1 seconds
IPUL	22:22:22:22:22:22	192.168.0.194	23 hours, 59 minutes, 32 seconds

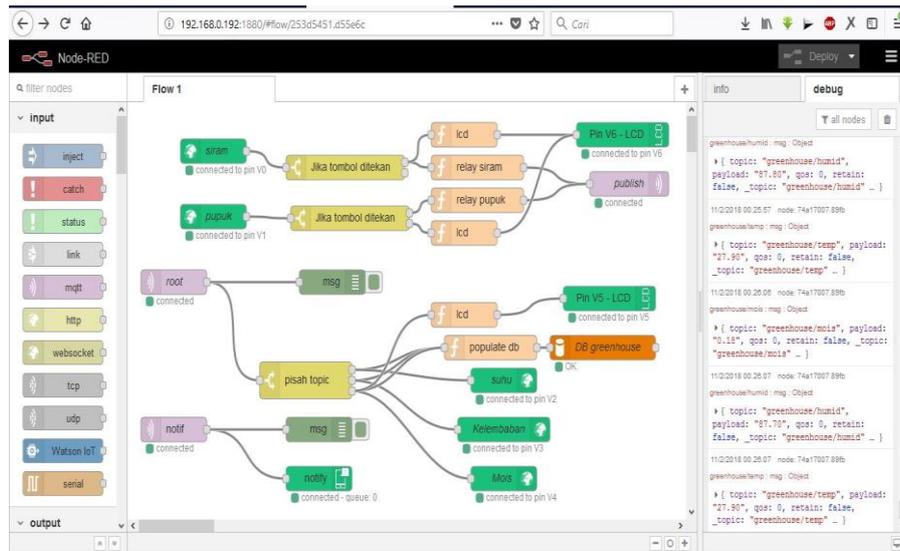
Gambar 17. Device Info Router

Sehingga akan tampak 4 jaringan dalam satu koneksi yaitu Raspberry Pi 3, wemos d1 r2, web dan android.



Gambar 18. Lalu Lintas Jaringan

Pada gambar 18. menjelaskan proses lalu lintas menggunakan jaringan lokal dimana sebelumnya dalam perangkat Raspberry Pi 3 telah ter-upload beberapa tool dan kode program seperti MQTT mosquito, Node.js, Node-RED, nginx dan php MySQL, kemudian proses untuk mengirimkan data dari wemos menuju ke aplikasi menggunakan tool Node-RED, seperti gambar 19.



Gambar 19. Tampilan Tools Node-RED

Gambar 19 menjelaskan mengenai lalu lintas pengiriman data suhu dan kelembaban bermula dari Wemos kemudian dikirim ke Raspberry Pi 3 dimana selanjutnya akan diolah dalam tool Node-RED kemudian data monitoring dihubungkan ke MySQL database dan aplikasi, berbeda dengan proses pengiriman pesan perintah atau kontrol dimana pesan bermula dari aplikasi android baru kemudian dikirim menuju Wemos melalui Raspberry Pi 3 dan Node-RED.

2. Mikrokontroler

Mikrokontroler WeMos D1 R2 WiFi Uno Based ESP8266 Shield harus diunggah dengan kode – kode program menggunakan software Anrduino IDE agar perangkat mikrokontroler dapat bekerja atau berjalan sesuai kebutuhan sistem, seperti pada gambar 20.

```

SmartGreenHouse | Arduino 1.8.1
File Edit Sketch Tools Help
SmartGreenHouse $
#include <FS.h> //this needs to be first, or it all crashes and burns...
#include <ESP8266WiFi.h> //https://github.com/esp8266/Arduino
#include <DNSServer.h>
#include <ESP8266WebServer.h>
#include <WiFiManager.h> //https://github.com/tzapu/WiFiManager
#include <ArduinoJson.h> //https://github.com/bblanchon/ArduinoJson
#include <PubSubClient.h>
#include "DHT.h"

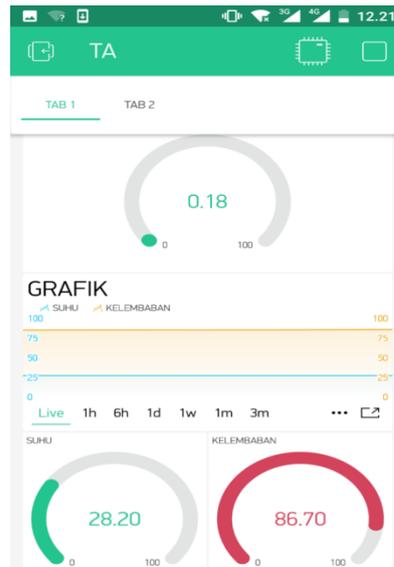
```

Gambar 20. Program Jaringan Pada Wemos D1 R2

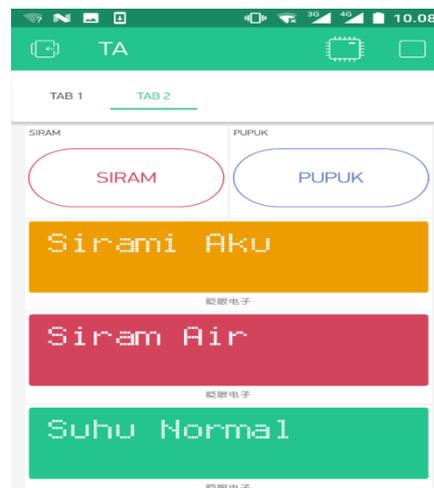
Selanjutnya agar program dapat ter-*upload* dengan baik kedalam perangkat wemos, maka diperlukan beberapa *Include Library Manager* yang mendukung sistem jaringan internet seperti WiFiManager.h, ArduinoJason.h, PubSubClient.h dan ESP8266webServer.h.

3. Android Device

Aplikasi android *device* yang digunakan menggunakan aplikasi Blynk berbasis android, jenis aplikasi tersebut dapat di unduh pada *play store* dan digunakan secara gratis sehingga dalam penggunaan android peneliti tidak perlu membuat aplikasi berbasis android baru, melainkan hanya merancang android tersebut agar dapat terhubung dan digunakan kedalam system. Pada *androiddevice*, data – data berasal dari wemos kemudian dikirim menuju server kemudian diolah menjadi sebuah informasi berupa grafik agar user lebih nyaman saat menggunakan sistem ini. *Androiddevice* dapat dilihat pada gambar 21.



Gambar 21. Tampilan Android Blynk Tab 1 Monitoring

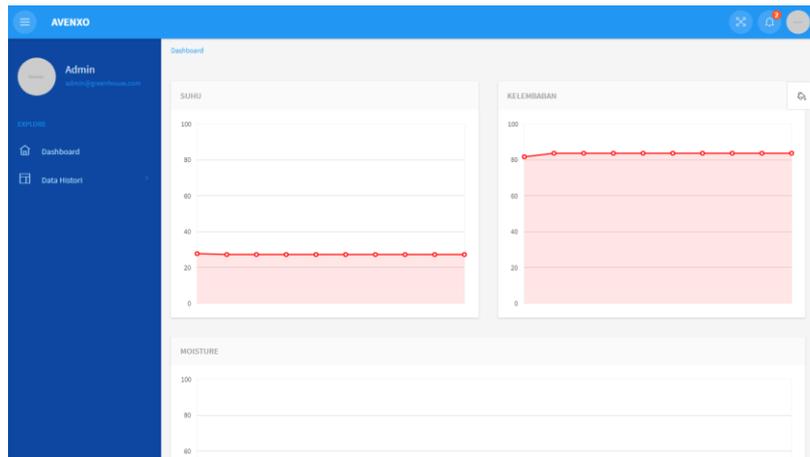


Gambar 22. Tampilan Android Blynk Tab 2 Kontrol

Aplikasi Blynk mendukung kinerja sistem berbasis *internet of things* dimana terdapat beberapa fitur yang memudahkan peneliti dalam melakukan proses pengiriman dan penerimaan data untuk dapat diolah ke dalam sistem, pada aplikasi Blynk inilah peneliti melakukan proses *subscribe* dan *publish* yaitu monitoring suhu, kelembaban udara dan kelembaban tanah, serta proses pengiriman pesan yaitu kontrol dua relay, relay pompa air dan relay pupuk cair.

4. Web Client

Aplikasi *web client* yang digunakan menggunakan aplikasi web berbasis php myql, berbeda dengan palikas Blynk berbasis android, aplikasi berbasis web ini perlu terlebih dahulu dilakukan pemrograman menggunakan bahasa pemrograman php mysql yang mendukung aplikasi berbasis web. Aplikasi *webclient* dibuat untuk mendukung kinerja sistem berbasis *internet of things* dimana *database* yang tersimpan akan dikirim dan diproses untuk nantinya diolah menjadi data berupa grafik dan *history*, pada aplikasi web inilah peneliti melakukan proses penerimaan data yaitu monitoring suhu, kelembaban udara dan kelembaban tanah.



Gambar 23. Tampilan Aplikasi *Web Client*

5. KESIMPULAN

Kesimpulan dari penelitian ini menghasilkan bahwa telah dibangun sebuah sistem perawatan tanaman *green house* secara *real time* sebagai solusi *memonitoring* meliputi suhu udara, kelembaban udara, kelembaban tanah dan control dua relay meliputi control penyiraman air dan pupuk cair dengan menggabungkan aplikasi android, web, mikrokontroller wemos, sensor dan beberapa rangkaian elektronik. Konsep IoT dengan menggunakan protokol MQTT mosquito dapat bekerja dengan baik pada kondisi lingkungan *greenhouse* dari jarak jangkauan *router* secara *realtime* dengan ketentuan adanya koneksi jaringan internet.

DAFTAR PUSTAKA

- [1] T. I. Society, "The Internet of Things: An Overview," 2015. [Online]. Available: https://www.internetsociety.org/sites/default/files/ISOC-IoT-Overview-20151014_0.pdf.
- [2] R. E., "Cara Memprogram Wemos D1 R2 Mini ESP8266 Dengan Arduino," 2016. [Online]. Available: <http://ekorudiawan.com/cara-memprogram-wemos-esp8266-dengan-arduino/>.
- [3] G.O. Satria, G.B. Satrya, dan A. Herutomo., *Implementasi Protokol Mqtt Pada Smart Building Berbasis Openmtc*. Universitas Telkom Bandung, 2014.
- [4] M. Tim J, "Develop with Node-RED," 2016. [Online]. Available: https://software.intel.com/en-us/articles/developing-with-node-red?utm_source=teknojurnal.com&utm_medium=Syndication&utm_campaign=IoT_indonesia_APAC_ContentSyndication_2016&cmp=tj162.
- [5] S.Guntoro, *Saatnya Menerapkan Pertanian Tekno-Ekologis Topan Nixo*. Jakarta: PT. Agromedia Pustaka, 2011.