

INTEGRASI PERANGKAT-PERANGKAT PADA *SMART HOME* MENGGUNAKAN *SMARTPHONE*

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh:

Ayu Anki NIRM : 0031705

Deswanto NIRM : 0031735

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2020**

LEMBAR PENGESAHAN

JUDUL PROYEK AKHIR

Integrasi Perangkat-perangkat pada *Smart Home*
Menggunakan *Smarthphone*

Oleh :

Ayu Anki /0031705

Deswanto /0031735

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



Indra Dwisaputra, M.T.

Pembimbing 2



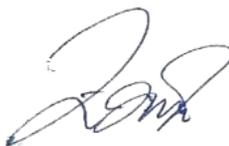
Ocsirendi, M.T.

Penguji 1



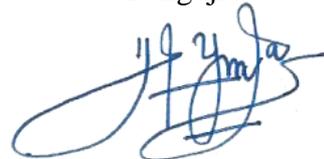
Aan Febriansyah, M.T

Penguji 2



Zanu Saputra, M.Tr.T

Penguji 3



Yang Agita R, M.Eng

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa 1 : Ayu Anki NIRM : 0031705

Nama Mahasiswa 2 : Deswarto NIRM : 0031735

Dengan Judul : Integrasi Perangkat-perangkat pada *Smart Home*
Menggunakan *Smartphone*

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 19 Agustus 2020

Nama Mahasiswa
1. Ayu Anki
2. Deswarto

Tanda Tangan


.....

.....

ABSTRAK

Perangkat elektronik yang digunakan secara manual membuat waktu serta finansial menjadi tidak efisien. Untuk mengatasi masalah tersebut pada proyek akhir ini munculah ide untuk memanfaatkan sistem smart home. Agar pengguna lebih nyaman dalam menggunakan perangkat elektronik. Perangkat akan diintegrasikan ke smartphone. Perangkat-perangkat yang akan diintegrasikan yaitu stopkontak, fitting lampu, pompa air, gerendel, dan termometer. Perangkat dan smartphone dapat terhubung dengan aplikasi berbasis android. Perangkat dihubungkan dengan smartphone menggunakan hotspot. Pengguna cukup memasukkan IP address yang ada pada NodeMCU kedalam menu SET IP. Setelah itu pengguna bisa memilih menu ON/OFF untuk mengaktifkan atau menonaktifkan perangkat, menu TIMER untuk mengaktifkan atau menonaktifkan perangkat dengan pengaturan waktu, dan menu DATA untuk monitoring data yang dibaca oleh sensor. Aplikasi ini dapat menampilkan data stopkontak, data fitting lampu dan data dari perangkat termometer. Perangkat dapat terhubung dengan aplikasi android secara bersamaan namun tetap memiliki waktu antrian data selama 1-2 detik, tergantung koneksi hotspot. Persentase perbandingan eror daya beban lampu 8 watt berada di bawah 4,8% dan perbandingan eror daya beban lampu 15 watt berada di bawah 7,7%.

Kata kunci: *Android, Smart Home, Smartphone*

ABSTRACT

Electronic devices used manually can turn out to be time-consuming and financially inefficient. To solve the problem, this final project requires an idea to take advantage of the smart home system. It aims to make users more comfortable using electronic devices. The device will be integrated into a smartphone. The devices to be integrated are plugs, light fittings, water pumps, bolt and thermometer. Devices and smartphone could be connected with android based on applications. Connecting the device could be use the hotspot on the smartphone. The device only needs to enter the IP address one NodeMCU to the IP SET menu. After that, the user can select the ON /OFF menu to activate or deactivate the device, thhe Timer menu to activate or deactivate the device by setting the time, and save DATA to monitor the data read by the sensor. This application can display plugs data, light fitting data and data froma thermometer device. Device can be connected to android app simultaneously, yet it has a queue time for about 1-2 seconds, depends on the hotspot connection. The percentage of error comparative 8 watt lamp power is below of 4,8% while the percentage of error comparative 15 watt lamp power is below of 7,7%.

Keywords : Android, Smart Home, Smartphone

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Puji dan syukur penulis panjatkan atas kehadiran Allah SWT yang telah memberi rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan laporan serta proyek akhir ini dengan baik dan tepat waktu.

Adapun judul proyek akhir ini adalah “Integrasi Perangkat-Perangkat pada *Smart Home* Menggunakan *Smarthphone*”. Tujuan dari pembuatan laporan proyek akhir ini yaitu sebagai salah satu persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program pendidikan Diploma III di Politeknik Manufaktur Negeri Bangka Belitung.

Dalam pelaksanaan pembuatan proyek akhir penulis ingin menyampaikan rasa terimakasih yang sebesar-besarnya kepada:

1. Bapak I Made Andik Setiawan, M. Eng, Ph. D., selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
2. Bapak Aan Febriansyah, M. T., selaku Ketua Jurusan Teknik Elektro dan Informatika Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Eko Sulistyono, M. T., selaku Ketua Program Studi Diploma III Teknik Elektronika Politeknik Manufaktur Negeri Bangka Belitung.
4. Bapak Indra Dwisaputra, M.T., selaku Dosen Pembimbing I yang telah membantu memberikan arahan dan bimbingan dalam penulisan serta pembuatan alat pada penyelesaian proyek akhir ini.
5. Bapak Ocsirendi, M.T., selaku Dosen Pembimbing II yang telah membantu memberikan arahan dan bimbingan dalam penulisan serta pembuatan alat pada penyelesaian proyek akhir ini.
6. Seluruh Dosen, Instruktur dan Staff pengajar Jurusan Teknik Elektro dan Informatika Politeknik Manufaktur Negeri Bangka Belitung yang telah mengajarkan banyak ilmu pengetahuan.
7. Kepada keluarga yang telah memberikan dukungan kepada penulis berupa dukungan moril maupun materi.

8. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung dalam pembuatan proyek akhir.

Penulis menyadari masih banyak kekurangan dalam penulisan laporan proyek akhir ini dikarenakan keterbatasan ilmu yang penulis miliki. Untuk itu dengan tidak mengurangi rasa hormat, dengan segala kerendahan hati, penulis sangat mengharapkan saran atau kritik yang sifatnya membangun dan bermanfaat untuk kesempurnaan proyek akhir ini.

Selanjutnya penulis berharap semoga proyek akhir ini dapat bermanfaat bagi para pembaca umumnya dan dapat dikembangkan untuk memperlancar dalam melaksanakan tugas yang berkaitan dengan proyek akhir ini.

Sungailiat, 19 Agustus 2020

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	ii
PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Proyek Akhir	2
BAB II.....	3
DASAR TEORI	3
2.1 <i>Smart Home</i>	3
2.2 Sensor	4
2.2.1 Sensor <i>PIR</i>	4
2.2.2 Sensor <i>Water Level</i>	5
2.2.3 Sensor <i>PZEM 004T</i>	7
2.2.4 Sensor <i>DHT11</i>	7
2.3 Topologi.....	8
2.4 Teori Daya, Arus, & Tegangan.....	9
2.4.1 Daya Listrik	10
2.4.1.1 Daya Aktif.....	10
2.4.1.2 Daya Reaktif	11

2.4.1.3	Daya Semu	11
2.4.1.4	Penggunaan Daya	11
2.4.2	Arus Listrik	11
2.4.3	Tegangan Listrik	11
2.5	<i>Android</i>	12
2.6	<i>MIT App Inventor</i>	12
2.7	<i>Hotspot</i>	13
BAB III		14
METODE PENELITIAN		14
3.1	Metode Pelaksanaan Pembuatan Proyek Akhir	14
3.2	Rancangan Algoritma Kerja <i>Hardware</i>	16
3.3	Komunikasi Antara Perangkat dan <i>Smartphone</i>	17
3.4	Rancangan <i>Hardware</i>	18
3.4.1	Rancangan <i>Hardware</i> Fiting Lampu	18
3.4.2	Rancangan <i>Hardware</i> Stopkontak	19
3.4.3	Rancangan <i>Hardware</i> Saklar Pompa Air	21
3.4.4	Rancangan <i>Hardware</i> Saklar Gerendel	22
3.4.5	Rancangan <i>Hardware</i> Termometer	24
3.5	Rancangan Aplikasi	25
3.5.1	Sub Menu <i>ON/OFF</i>	26
3.5.2	Sub Menu Data	27
3.5.3	Sub Menu <i>Timer</i>	27
3.5.4	Sub Menu Set IP	28
BAB IV		30
PEMBAHASAN		30
4.1	Pengujian Fiting Lampu	30
4.1.1	Pengujian ON/OFF Fiting Lampu Manual	30
4.1.2	Pengujian ON/OFF Fiting Lampu Otomatis	31
4.1.3	Pengujian Monitoring Data Pada Fiting Lampu	32
4.1.4	Pengujian Pengaturan Timer Pada Fiting Lampu	34
4.2	Pengujian Stopkontak	35

4.2.1 Pengujian Stopkontak ON/OFF	35
4.2.2 Pengujian Monitoring Data Pada Stopkontak.....	36
4.2.3 Pengujian Pengaturan Timer Pada Stopkontak.....	38
4.3 Pengujian Pompa Air	38
4.3.1 Pengujian ON/OFF Pompa Air Manual	39
4.3.2 Pengujian ON/OFF Pompa Air Otomatis	40
4.4 Pengujian Gerendel.....	40
4.5 Pengujian Termometer.....	41
BAB V.....	43
KESIMPULAN DAN SARAN.....	43
5.1 Kesimpulan.....	43
5.2 Saran	43
DAFTAR PUSTAKA	44

DAFTAR TABEL

Tabel 2. 1 Konfigurasi Sensor <i>PIR</i>	5
Tabel 2. 2 Spesifikasi Sensor <i>Water Level</i>	6
Tabel 2. 3 Tabel Percobaan Jarak Komunikasi	9
Tabel 4. 1 Hasil Pengujian Fiting Lampu Mode Manual.....	31
Tabel 4. 2 Hasil Pengujian Fiting Lampu Mode Otomatis	32
Tabel 4. 3 Data Perbandingan Tegangan	33
Tabel 4. 4 Data Perbandingan Arus	33
Tabel 4. 5 Data Perbandingan Daya.....	34
Tabel 4. 6 Hasil Pengaturan <i>Timer</i> Fiting Lampu.....	34
Tabel 4. 7 Hasil Pengujian <i>ON / OFF</i> Stopkontak	35
Tabel 4. 8 Data Perbandingan Tegangan	36
Tabel 4. 9 Data Perbandingan Arus	37
Tabel 4. 10 Data Perbandingan Daya.....	38
Tabel 4. 11 Hasil Pengujian Pengaturan Timer Stopkontak	38
Tabel 4. 12 Hasil Pengujian Pompa Air Mode Manual	39
Tabel 4. 13 Hasil Pengujian <i>ON / OFF</i> Pompa Air Mode Otomatis	40
Tabel 4. 14 Hasil Pengujian <i>ON / OFF</i> Gerendel	41
Tabel 4. 15 Hasil Pengujian <i>Monitoring</i> Pada Termometer	42

DAFTAR GAMBAR

Gambar 2. 1 Gambar Sistem Rumah Pintar (<i>Smart Home</i>)	3
Gambar 2. 2 Gambar Sensor <i>PIR</i>	4
Gambar 2. 3 Gambar Sensor <i>Water Level</i>	6
Gambar 2. 4 Gambar Sensor <i>PZEM 004T</i>	7
Gambar 2. 5 Gambar Sensor <i>DHT11</i>	8
Gambar 2. 6 Ilustrasi Topologi <i>Star</i>	9
Gambar 2. 7 Gambar Versi <i>Android</i> dan Logonya	12
Gambar 2. 8 Logo <i>MIT App Inventor</i>	13
Gambar 2. 9 Logo <i>Hotspot</i>	13
Gambar 3. 1 <i>Flowchart</i> Pembuatan Proyek Akhir.....	14
Gambar 3. 2 <i>Flowchart</i> Rancangan Algoritma Kerja Alat	16
Gambar 3. 3 Ilustrasi Antara <i>Smartphone</i> Dengan Masing-Masing Perangkat..	17
Gambar 3. 4 Gambar Rancangan <i>Hardware</i> Fiting Lampu.....	18
Gambar 3. 5 Gambar Skematik Fiting Lampu	19
Gambar 3. 6 Gambar Rancangan <i>Hardware</i> Stopkontak.....	20
Gambar 3. 7 Gambar Skematik Stopkontak.....	21
Gambar 3. 8 Rancangan Saklar Pompa Air	21
Gambar 3. 9 Gambar Skematik Saklar Pompa Air	22
Gambar 3. 10 Rancangan Saklar Gerendel	23
Gambar 3. 11 Gambar Skematik Saklar Gerendel	23
Gambar 3. 12 Rancangan <i>Hardware</i> Termometer	24
Gambar 3. 13 Gambar Skematik Termometer	25
Gambar 3. 14 Gambar Menu Pada Aplikasi <i>Smartphone</i>	25
Gambar 3. 15 Gambar Sub Menu <i>ON/OFF</i>	26
Gambar 3. 16 Gambar Sub Menu Dari Data.....	27
Gambar 3. 17 Gambar Sub Menu Dari Timer.....	28
Gambar 3. 18 Gambar Sub Menu Dari Set IP.....	29
Gambar 4. 1 (a) Menyalakan Lampu (b) Mematikan Lampu	31

Gambar 4. 2 (a) Menyalakan Stopkontak (b) Mematikan Stopkontak	35
Gambar 4. 3 (a) Menyalakan Pompa Air (b) Mematikan Pompa Air.....	39
Gambar 4. 4 (a) Gerendel Kondisi Aktif (b) Gerendel Kondisi Tidak Aktif.....	41

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Riwayat Hidup

Lampiran 2 : Program Aplikasi

Lampiran 3 : Program *Nodemcu*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan zaman membuat perangkat elektronik menjadi kebutuhan bagi setiap orang. Namun pengguna sering lalai dalam mengoperasikan alat-alat elektronik tersebut. Pengguna sering lupa dalam mematikan lampu, mengunci pintu, dan mematikan pompa air. Hal tersebut jika sering terjadi akan berdampak tidak efisiennya waktu serta finansial yang digunakan [1]. Oleh karena itu dengan tujuan efisiensi agar penghuni menjadi nyaman dan aman dalam menggunakan perangkat elektronik. Maka muncul ide untuk membuat inovasi perangkat-perangkat elektronik menggunakan sistem rumah cerdas (*smart home*).

Sebelumnya telah dibuat alat proyek akhir yang berjudul “Stopkontak Pintar Berbasis *Android*” [2] dan “Kontrol dan *Monitoring* Fiting Lampu Berbasis *Android*” [3]. Pada proyek akhir berjudul “Stopkontak Pintar Berbasis *Android*” alat ini mampu *memonitoring* penggunaan daya listrik, biaya penggunaan dan mengontrol stopkontak melalui *Android*. Perangkat ini memiliki kekurangan yaitu hanya bisa digunakan dengan jarak maksimal 10 meter serta masing-masing stopkontak tidak bisa berbagi informasi satu sama lain sehingga harus melakukan pengecekan dengan mengkoneksikan ke masing-masing stopkontak [2], sedangkan pada proyek akhir yang berjudul “Kontrol dan *Monitoring* Fiting Lampu Berbasis *Android*” mempunyai kekurangan seperti hanya bisa menggunakan 2 fitting lampu serta alat yang diaplikasikan kedalam sistem *smart home* hanya fitting lampu [3].

Dari beberapa perangkat yang telah dibuat sebelumnya munculah ide untuk mengembangkan dan memodifikasi proyek akhir sebelumnya, sehingga penulis ingin membuat sistem yang menjamin penggunaannya agar lebih aman dan nyaman serta bersifat lebih efisien baik secara waktu maupun finansial, dengan itu penulis membuat proyek akhir baru yang berjudul “Integrasi Perangkat-perangkat *Smart Home* menggunakan *Smartphone*”. Perangkat-perangkat ini dapat *dimonitoring*

dan dikontrol dengan mudah dengan *smartphone*. Beberapa alat juga memiliki sistem otomatis dan manual yang bisa diatur oleh penggunanya karena alat-alat ini diintegrasikan kedalam sebuah aplikasi yang berbasis *android*.

1.2 Rumusan Masalah

Dari latar belakang yang ada, maka ada beberapa rumusan masalah yang akan dikaji oleh penulis dalam laporan proyek akhir ini adalah :

1. Bagaimana cara mengkoneksikan *smartphone* dengan perangkat-perangkat *smart home*.
2. Bagaimana cara mengirim data-data pada perangkat *smart home* ke dalam *smarthphone*.
3. Bagaimana cara membuat aplikasi pada *smartphone* yang bisa terhubung ke perangkat *smart home*.
4. Bagaimana membuat fitting lampu dan pompa air agar bisa bekerja secara otomatis dan manual.

1.3 Batasan Masalah

Untuk menghindari ruang lingkup permasalahan yang bersifat cukup kompleks, maka dalam pembuatan proyek akhir ini dibuat beberapa batasan masalah, yaitu :

1. Aplikasi hanya tersedia dalam bentuk *smartphone*, tidak dalam bentuk web, dan aplikasi laptop.
2. Setiap perangkat yang dibuat hanya 1 alat saja.

1.4 Tujuan Proyek Akhir

Adapun tujuan dalam penulisan laporan proyek akhir yang berjudul “Integrasi Perangkat-perangkat pada *smart home* menggunakan *smartphone*” adalah sebagai berikut :

1. Membuat perangkat stopkontak, fitting lampu dengan sistem otomatis, sistem otomatis pada pompa air, *smartlock*, dan alat *monitoring* suhu ruangan.
2. Membuat sistem aktivasi mode manual dan otomatis untuk lampu dan pompa air.

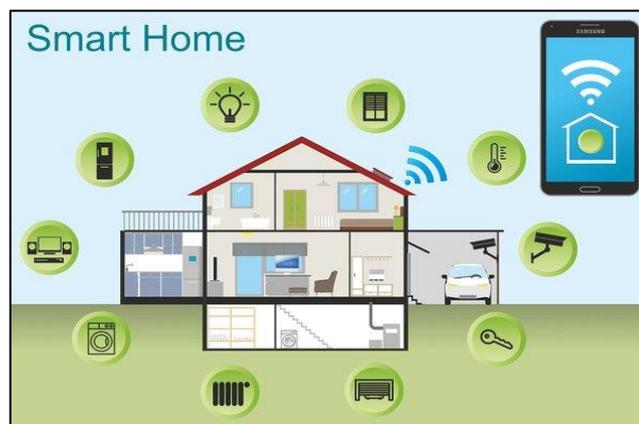
BAB II

DASAR TEORI

2.1 *Smart Home*

Smart home merupakan sebuah aplikasi yang terprogram melalui komputer yang bisa memberikan kenyamanan, keamanan, serta penghematan energi secara otomatis sesuai dengan kendali pengguna [4]. Pada referensi yang berjudul “implementasi aplikasi rumah pintar berbasis *android* dengan *arduino microcontroller*” sistem ini dibuat dengan tujuan mempermudah penggunaanya dalam mengendalikan perangkat elektronik yang terdapat di dalam rumah tersebut [5].

Rumah pintar dapat diartikan sebagai sebuah sistem gabungan antar teknologi dan pelayanan yang dibuat dengan maksud meningkatkan keamanan, efisiensi, dan kenyamanan penghuninya [6]. Selain mengontrol peralatan elektronik, sistem ini juga bisa *monitoring* data perangkat elektronik. Seperti pada penelitian yang berjudul “penggunaan *DHT11* dan *arduino* sebagai pendeteksi suhu pada laptop” pada penelitian ini bertujuan untuk mendeteksi suhu pada laptop yang digunakan untuk mencegah *overheating* [7]. Untuk gambar rumah pintar (*smart home*) akan ditunjukkan pada Gambar 2.1 berikut ini [8].



Gambar 2. 1 Gambar Sistem Rumah Pintar (*Smart Home*)

2.2 Sensor

Sensor merupakan alat yang berfungsi merubah besaran fisik menjadi listrik. Sensor biasanya dapat mendeteksi sebuah gejala atau sinyal yang berasal dari sebuah energi listrik, energi fisika, energi kimia, energi biologi, dan energi lainnya. Sensor yang baik harus memiliki linieritas, sensitivitas, dan tanggapan waktu [9].

2.2.1 Sensor *PIR*

Sensor *PIR* (*Passive infrared receiver*) adalah sebuah sensor yang dapat mendeteksi radiasi *infrared* pada suatu benda. Sesuai dengan namanya sensor ini bersifat pasif karena hanya dapat mendeteksi pancaran dari *infrared* [3]. Dengan menggunakan sensor ini maka perangkat dapat mendeteksi keberadaan manusia jika ada pergerakan yang melewati sensor *PIR* ini. Untuk gambar snsor *PIR* akan ditunjukkan pada Gambar 2.2 berikut ini [10].



Gambar 2. 2 Gambar Sensor *PIR*

Pada gambar dapat dilihat sensor *PIR* memiliki bagian seperti setengah bola. Bagian tersebut berguna untuk memancarkan sensor agar jangkauan sensor bisa lebih luas.

Untuk bisa mengkoneksikan sensor *PIR* dengan mikrokontroler ada beberapa konfigurasi yang harus disesuaikan. Konfigurasi dari sensor *PIR* dapat dilihat dari Tabel 2.1 berikut ini [10].

Tabel 2. 1 Konfigurasi Sensor *PIR*

NO	Nama Bagian	Keterangan
1	Pengatur waktu jeda	Digunakan untuk Mengatur lama pulsa <i>high</i> setelah terdeteksi terjadinya gerakan dan setelah gerakan berakhir
2	Pengatur sensitivitas	Pengatur tingkat sensitivitas sensor <i>PIR</i>
3	Regulator 3VDC	Penstabil tegangan menjadi 3VDC
4	Dioda Pengaman	Mengamankan sensor jika terjadi salah pengkabelan antara <i>VCC</i> dan <i>ground</i>
5	<i>DC Power</i>	Input tegangan dengan (3-12) VDC (Direkomendasikan dengan menggunakan input 5VDC)
6	Output Digital	Output digital sensor
7	<i>Ground</i>	Hubungkan dengan <i>ground</i>
8	<i>BIS0001</i>	<i>IC</i> sensor <i>PIR</i>
9	Pengatur <i>Jumper</i>	Untuk mengatur output dari pin digital

2.2.2 Sensor *Water Level*

Water level merupakan sensor yang digunakan untuk mendeteksi ketinggian air atau untuk mendeteksi kosong atau tidaknya air pada suatu tempat. Sensor *water level* ini bisa mengukur ketinggian air menjadi sebuah sinyal analog. Nilai sinyal analog tersebut bisa digunakan secara langsung dalam mode program [11]. Sensor ini juga memiliki sensitivitas yang begitu tinggi dan untuk gambar sensor ini juga akan ditunjukkan pada Gambar 2.3 berikut ini [12].



Gambar 2. 3 Gambar Sensor *Water Level*

Dapat dilihat bentuk sensor *water level* yang digunakan berbentuk seperti pelampung. Hal ini untuk memudahkan sensor bekerja. Karena air akan menggerakkan pelampung tersebut.

Untuk bisa menyesuaikan sensor dengan mikrokontroler agar bisa bekerja dengan baik. Maka harus disesuaikan dengan spesifikasi sensor. Berikut spesifikasi dari sensor *water level* akan ditunjukkan pada Tabel 2.2. [12].

Tabel 2. 2 Spesifikasi Sensor *Water Level*

Panjang kabel	36 inci
Jenis	Baik
Beban maksimum	50 watt
Tegangan switching maksimum	100V DC
Tegangan minimum	250V DC
Arus switching maksimum	0,5A
Arus beban maksimum	1.0
Resistansi kontak maks	0,4 Ohm
Suhu pengoprasian	-20 ~+80°C
Berat bersih	12g
Bahan	PP

2.2.3 Sensor *PZEM 004T*

Sensor *PZEM 004T* merupakan sebuah sensor yang bisa mengukur arus, tegangan, dan daya listrik *AC (Alternatif Current)* yang dikeluarkan melalui komunikasi serial [3]. Namun sensor ini tidak mampu membaca arus dengan ketelitian *mili ampere*. Sensor ini juga memiliki sensor tegangan dan sensor arus yang sudah terintegrasi [13]. Untuk gambar sensor *PZEM 004T* akan ditunjukkan pada Gambar 2.4 berikut ini [14].

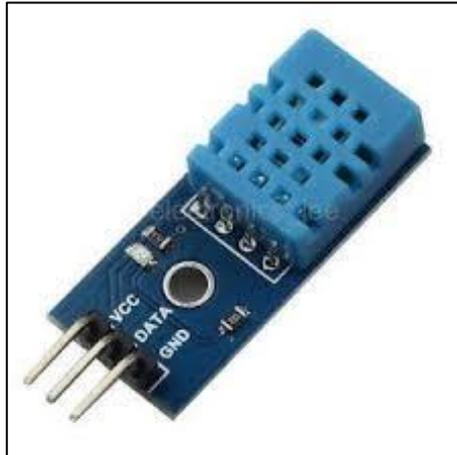


Gambar 2. 4 Gambar Sensor *PZEM 004T*

2.2.4 Sensor *DHT11*

Sensor *DHT11* adalah sebuah sensor yang dapat membaca suhu (*temperature*) dan kelembaban (*humidity*) suatu ruangan. Stabilitas yang dimiliki oleh sensor *DHT11* ini sangat baik. Sensor ini memiliki stabilitas yang baik serta fitur kalibrasi yang akurat [15].

Sensor ini jika dilihat dari respon, pembacaan data yang cepat, dan kemampuan anti-*interference* maka termasuk kedalam kualitas terbaik. Sensor ini memiliki respon yang cepat dalam pembacaan data. Sensor ini memiliki transmisi sinyal hingga 20 meter. Oleh karena itulah sensor ini banyak digunakan dalam aplikasi-aplikasi suhu dan kelembaban. Untuk Gambar 2.5 adalah contoh fisik sensor *DHT11* [16].



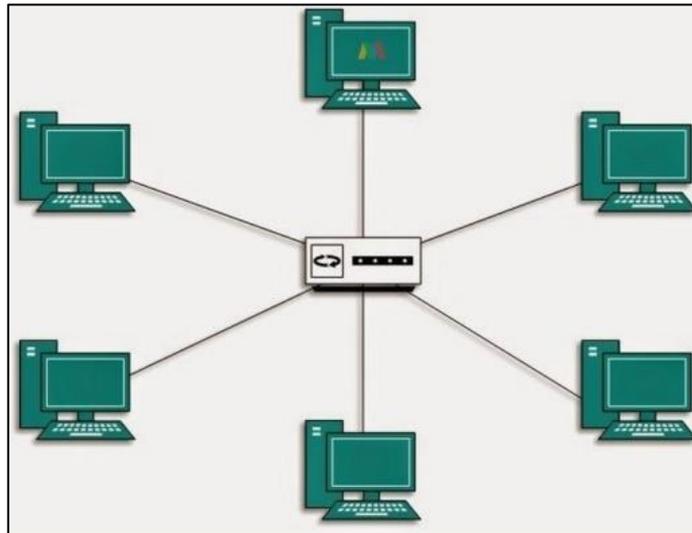
Gambar 2. 5 Gambar Sensor *DHT11*

2.3 Topologi

Topologi merupakan sebuah sistem jaringan yang membentuk struktur jaringan yang bisa menghubungkan perangkat antar jaringan dengan menggunakan kabel ataupun tanpa kabel (*nirkabel*) [17]. Topologi memiliki beberapa jenis yaitu [18] :

- Topologi bus
- Topologi *star*
- Topologi *ring*
- Topologi *mesh*
- Topologi *hybrid*
- Topologi *peer to peer*
- Topologi *linier*

Untuk topologi yang digunakan dalam proyek akhir ini adalah topologi *star*. Topologi *star* merupakan topologi yang setiap perangkat memiliki akses jaringan tersendiri untuk terhubung ke perangkat pusat [3]. Berikut merupakan ilustrasi topologi *star* yang akan ditunjukkan oleh Gambar 2.6 [3].



Gambar 2. 6 Ilustrasi Topologi *Star*

Dalam penggunaan topologi *star* telah dilakukan pengujian jarak komunikasi efektif antara perangkat dan smartphone. Berikut hasil pengujian yang dilakukan akan ditunjukkan pada Tabel 2.3 berikut ini.

Tabel 2. 3 Tabel Percobaan Jarak Komunikasi

NO	Jarak	Tanpa Penghalang	Dengan penghalang
1	5 Meter	Terhubung	Terhubung
2	10 Meter	Terhubung	Terhubung
3	15 Meter	Terhubung	Tidak Terhubung
4	20 Meter	Tidak Terhubung	Tidak Terhubung
5	25 Meter	Tidak Terhubung	Tidak Terhubung
6	30 Meter	Tidak Terhubung	Tidak Terhubung

2.4 Teori Daya, Arus, & Tegangan

Daya (watt), arus (ampere), dan Tegangan (volt) adalah 3 istilah yang paling sering didengar. Ketiga hal tersebut mempunyai hubungan yang saling berkaitan. Contohnya ketika kita menggunakan perangkat elektronik. Untuk menggunakan suatu perangkat elektronik di rumah, maka daya, arus dan tegangan harus

terpenuhi. Berikut persamaan yang menunjukkan keterkaitan antara 3 hal tersebut akan ditunjukkan pada persamaan 2.1 berikut ini [19] :

$$P = V \times I \quad \dots\dots\dots (2.1)$$

$$V = P/I$$

$$I = P/V$$

Keterangan :

- P : Power (Daya Listrik) dalam satuan Watt
- V : Voltage (Tegangan Listrik) dalam satuan Volt
- I : Intersity (Arus Listrik) dalam satuan Ampere

Tegangan (volt) merupakan hal yang pertama kali harus dihasilkan, karena jika tidak terdapat tegangan maka daya dan arus tidak akan muncul. Ketika tegangan mulai dialirkan kepada suatu alat elektronika maka akan dihasilkan arus yang sering disebut dengan satuan ampere. Besaran yang mengalir dari sumber listrik menuju alat elektronika ini dipengaruhi oleh seberapa besarnya nilai tahanan pada suatu alat elektronika tersebut, hal ini pada umumnya disebut daya listrik yang memiliki satuan Watt [20].

2.4.1 Daya Listrik

Daya atau bisa juga disebut *power* merupakan banyaknya energi yang dihasilkan atau digunakan dalam suatu alat elektronika dalam satuan waktu [20].

2.4.1.1 Daya Aktif

Daya aktif adalah daya yang sudah terpakai untuk melakukan energi sebenarnya [21]. Daya aktif memiliki satuan Watt. Daya aktif dapat dihitung menggunakan persamaan 2.2 berikut :

$$P = V \times I \times \text{Cos } \varphi \quad \dots\dots\dots (2.2)$$

Keterangan :

- V = Tegangan, Volt
- I = Arus, Ampere
- φ = Faktor daya

2.4.1.2 Daya Reaktif

Daya reaktif adalah jumlah daya yang dibutuhkan untuk membentuk medan magnet [21]. Daya reaktif dapat dihitung menggunakan persamaan 2.3 berikut :

$$Q = V \times I \times \sin \phi \quad \dots\dots\dots(2.3)$$

Keterangan :

- V = Tegangan, Volt
- I = Arsus, Ampere
- ϕ = Faktor daya

2.4.1.3 Daya Semu

Daya semu merupakan daya yang didapat dari penjumlahan tegangan rms dan arus rms dalam suatu daya [21]. Satuan daya semu adalah VA(Voltampere).Daya semu dapat dihitung menggunakan persamaan 2.3 berikut :

$$S = V \times I \quad \dots\dots\dots(2.3)$$

Keterangan :

- V = Tegangan, Volt
- I = Arus, Ampere

2.4.1.4 Penggunaan Daya

Dalam penggunaan daya, energi yang digunakan oleh listrik adalah daya dikalikan dengan waktu penggunaan alat tersebut [3]. Untuk mengukur satu jam pemakaian maka dihitung menggunakan persamaan 2.5 berikut :

$$\text{Daya per – jam} = \frac{V \times I}{1000} \quad \dots\dots\dots(2.5)$$

2.4.2 Arus Listrik

Arus listrik adalah banyaknya muatan listrik yang mengalir dalam sebuah alat elektronika dalam satuan waktu [20]. Satuan besaran arus listrik yang mengalir biasanya menggunakan satuan ampere(A). Arus juga biasanya dikenal dengan sebutan intensity(I).

2.4.3 Tegangan Listrik

Tegangan listrik atau *voltage* (V) merupakan sumber listrik yang memiliki nilai potensial antara dua titik yang berbeda. Satuan yang digunakan tegangan

adalah *volt* (V). Satuan *volt* disini menyatakan besaran yang dibutuhkan untuk mengaktifkan alat elektronika tersebut.) [19].

2.5 *Android*

Android merupakan sebuah sistem operasi berbasis *linux* yang bersifat *open source*. *Android* menyediakan platform terbuka agar para pengembang mudah untuk menciptakan aplikasi mereka sendiri. Sistem *android* ini dirancang untuk *smartphone* atau komputer *tablet* [3]. Seperti *software* lainnya *android* juga memiliki berbagai macam versi dan pengembangan dengan nama julukan masing-masing pada setiap versi. Terdapat beberapa kelebihan yang dimiliki *android* [22].

- *Android* memiliki sistem *user interface* / sistem antarmuka.
- *Android* bersifat *open source*.
- *Android* memiliki banyak fitur aplikasi yang dibutuhkan.

Sejak pertama kali *android* dikembangkan selalu menggunakan awalan sesuai dengan urutan huruf alfabet dengan penamaan makanan manis, kecuali pada *android* 1.1, *android* 1.2, dan *android* 10. [3] Berikut versi serta logo *android* yang akan ditunjukkan oleh Gambar 2.7 berikut ini.



Gambar 2. 7 Gambar Versi *Android* dan Logonya

2.6 *MIT App Inventor*

MIT App Inventor merupakan *tool* pemrograman untuk memudahkan pembuatan sebuah aplikasi yang dapat diinstal pada *android*, pembuatan desain

tampilan dan penyusunan blok pada algoritmanya menggunakan metode *click* dan *drag*. Pembuatan aplikasi *android* ini dapat diakses melalui browser internet. *MIT App Inventor* ini dikembangkan oleh *google* dan *MIT* [3]. Berikut logo *MIT App Inventor* yang akan ditunjukkan oleh Gambar 2.8 [23].

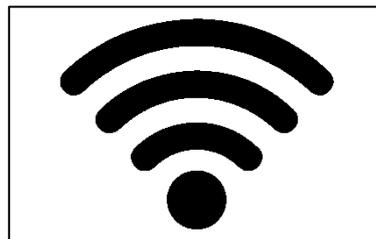


Gambar 2. 8 Logo *MIT App Inventor*

2.7 *Hotspot*

Hotspot merupakan sebuah area dimana orang atau *user* dapat menggunakan jaringan internet. *Hotspot* berfungsi sebagai koneksi internet. *Hotspot* dapat diakses melalui PC, laptop, handphone, notebook dan lainnya. Dalam penggunaan *hotspot* terdapat beberapa kelebihan. Berikut beberapa kelebihan *hotspot* : [24]

- Mudah diakses.
- Dapat digunakan pada beberapa perangkat dalam waktu bersamaan.
- Minat masyarakat yang tinggi
- Tidak ada batas waktu penggunaan

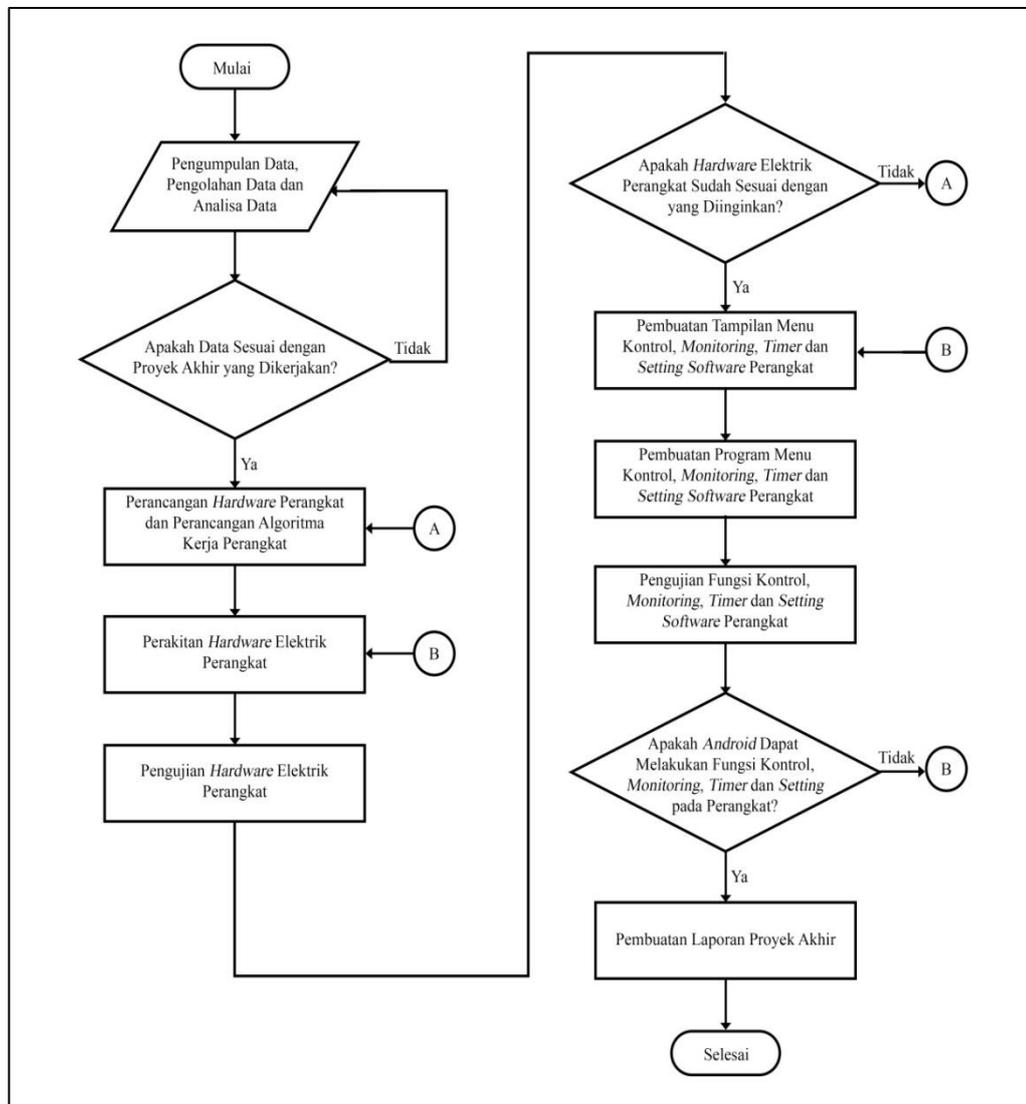


Gambar 2. 9 Logo *Hotspot*

BAB III METODE PENELITIAN

3.1 Metode Pelaksanaan Pembuatan Proyek Akhir

Proses dari mulai sampai selesainya pembuatan proyek akhir dijelaskan pada *flowchart* pembuatan proyek akhir. *Flowchart* pembuatan proyek ditunjukkan pada Gambar 3.1 berikut.



Gambar 3. 1 *Flowchart* Pembuatan Proyek Akhir

Gambar 3.1 merupakan *flowchart* pembuatan proyek akhir yang menjelaskan alur dari proses pengerjaan proyek akhir. Hal yang pertama dilakukan yaitu pengumpulan data yang diperlukan dalam menyelesaikan proyek akhir. Proses pengumpulan data dapat dilakukan secara langsung (primer) dan secara tidak langsung (sekunder). Pengumpulan data secara langsung merupakan pengumpulan data yang diperoleh dari konsultasi dengan dosen pembimbing sedangkan untuk pengumpulan data secara tidak langsung (sekunder) merupakan data yang diperoleh dari membaca makalah proyek akhir tahun sebelumnya yang berjudul “kontrol dan *monitoring* perangkat *smart home* berbasis *android*” dan dari referensi – referensi dari penelitian yang masih berhubungan dengan proyek akhir yang ingin dibuat. Data - data yang didapatkan dari proses pengumpulan data selanjutnya diolah dan dianalisa. Data yang telah diolah dan dianalisa dikumpulkan lalu dipilih untuk menjadi referensi dan acuan dalam pembuatan proyek akhir. Perancangan *hardware* perangkat *smart home* dan perancangan algoritma kerja alat dibuat agar *hardware* dan prinsip kerja alat sesuai dengan spesifikasi proyek akhir. Setelah selesai perakitan *hardware* elektrik, maka dilakukan pengujian, tujuannya untuk mengetahui apakah *hardware* elektrik perangkat *smart home* sudah sesuai dengan spesifikasi yang diinginkan.

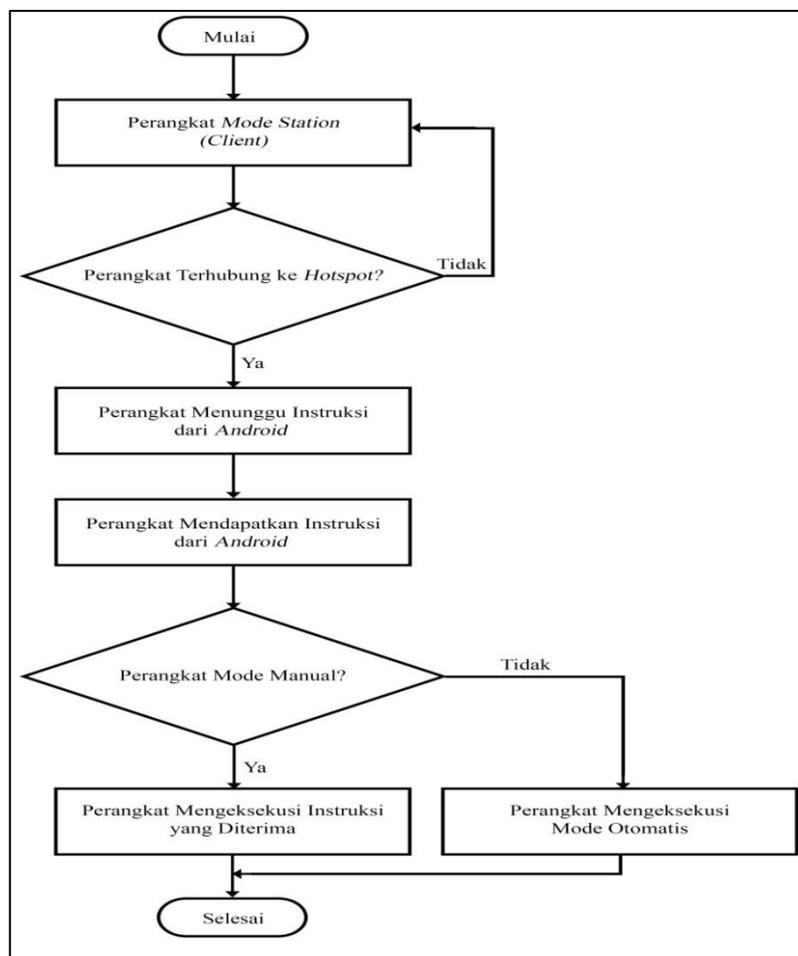
Pembuatan *software* menggunakan *MIT App Inventor 2* yang diakses menggunakan browser internet. Pembuatan tampilan menu dilakukan pada bagian designer *MIT App Inventor 2*. Pembuatan tampilan menu kontrol, *monitoring*, timer dan setting harus mudah dipahami serta digunakan dan sesuai fungsi dari perangkat yang akan dibuat. Pembuatan program menu kontrol, *monitoring*, timer dan *setting software* perangkat *smart home* dilakukan pada bagian *bloks MIT App Inventor 2*. 20 Pengujian fungsi kontrol, *monitoring*, timer dan *setting software* perangkat *smart home* dilakukan dengan cara menginstal *software* yang telah dibuat ke *Android*. Pengujian fungsi dilakukan secara keseluruhan, baik dari segi kontrol, *monitoring*, timer dan *setting software* harus dapat bekerja sesuai dengan fungsinya.

Proses pembuatan laporan merupakan proses terakhir dalam pembuatan proyek akhir. Pembuatan laporan bertujuan merangkum keseluruhan yang

berhubungan dengan proyek akhir, seperti latar belakang, tujuan, rumusan masalah, batasan masalah, teori dasar, metode penelitian, pembahasan serta kesimpulan dan saran.

3.2 Rancangan Algoritma Kerja *Hardware*

Rancangan algoritma kerja *hardware* merupakan rancangan dari proses cara kerja *hardware* untuk memenuhi spesifikasi yang diinginkan. Berikut adalah *flowchart* rancangan algoritma kerja *hardware* ditunjukkan pada Gambar 3.2 berikut.



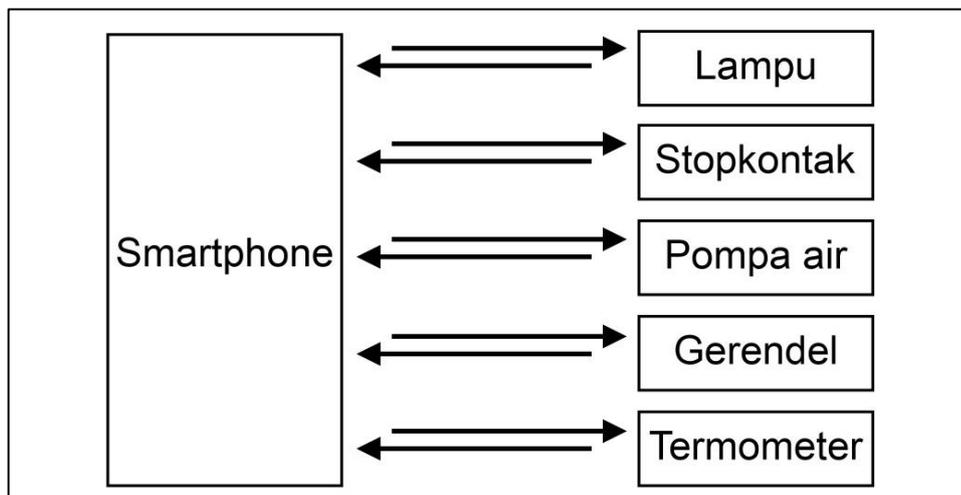
Gambar 3. 2 *Flowchart* Rancangan Algoritma Kerja Alat

Pada Gambar 3.2 merupakan *flowchart* rancangan algoritma kerja alat yang digunakan dalam mencapai target penyelesaian sesuai dengan spesifikasi yang diinginkan. Dalam keadaan normal, maka perangkat *smart home* berfungsi

sebagai *station (client)*. Saat dalam mode *client* perangkat *smart home* bertugas mencari dan menghubungkan ke *hotspot* yang sesuai dengan *SSID (Service Set Identifier)* dan kata sandi yang telah diatur sesuai dengan program yang telah dibuat. Proses pencarian akan berhenti saat perangkat *smart home* telah terhubung ke *hotspot* yang *SSID* dan kata sandinya sesuai dengan program, namun jika hubungan antara perangkat *smart home* dan *hotspot* terputus, maka perangkat *smart home* akan melakukan pencarian *hotspot* sampai berhasil terhubung kembali ke *hotspot* yang sesuai. Ketika perangkat *smart home* terhubung ke *hotspot*, maka perangkat *smart home* akan menunggu instruksi yang akan dikirim dari *smartphone* yang telah terhubung. Setelah mendapatkan instruksi perangkat mengeksekusi instruksi yang diberikan.

3.3 Komunikasi Antara Perangkat dan *Smartphone*

Pada pembuatan proyek *smarthphone* akan melakukan komunikasi data dengan masing-masing perangkat *smart home* yang akan ditunjukkan pada ilustrasi 3.3 berikut ini.



Gambar 3. 3 Ilustrasi Antara *Smarthphone* Dengan Masing-Masing Perangkat

Dari gambar dapat disimpulkan *smarthphone* berfungsi sebagai perangkat yang bisa menerima serta mengirim data yang didapat dari masing-masing perangkat *smart home*. Data-data tersebut akan diolah di NodeMCU terlebih

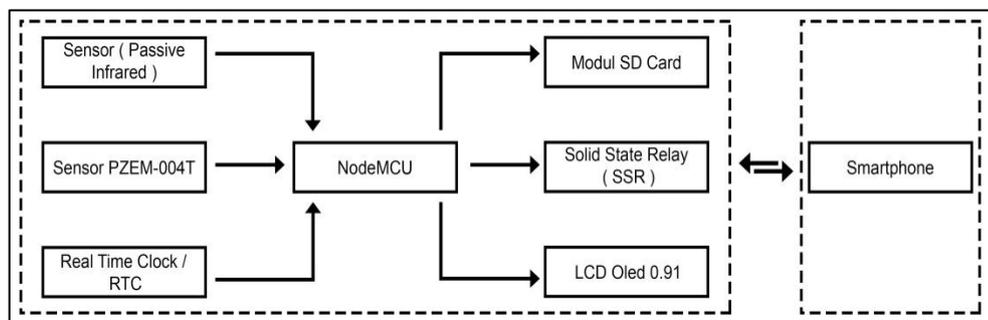
dahulu sebelum dikirim dan diterima oleh setiap perangkat. Berikut uraian penjelasan pada masing-masing perangkat *smart home*.

3.4 Rancangan *Hardware*

Pada pembuatan proyek akhir yang berjudul “Integrasi Perangkat-perangkat Pada *smart home* Menggunakan *smartphone*” terdapat rancangan *hardware* untuk masing-masing perangkat.

3.4.1 Rancangan *Hardware* Fiting Lampu

Dalam proses pembuatan fitting lampu terdapat rancangan *hardware* untuk memenuhi spesifikasi pembuatan alat yang diinginkan. Berikut adalah gambar rancangan *hardware* pada fitting lampu yang akan ditunjukkan pada Gambar 3.4.

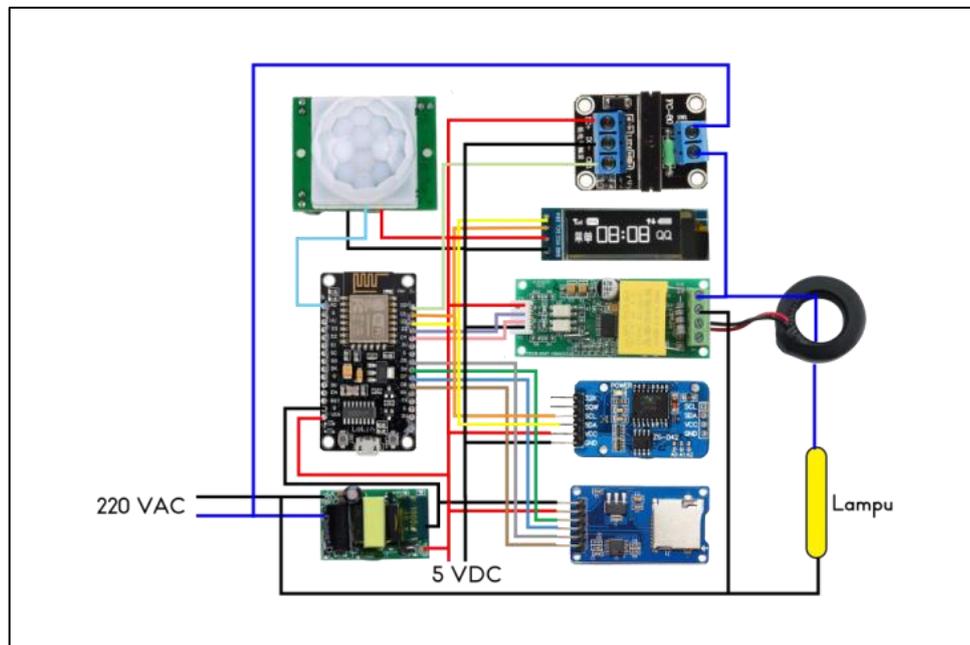


Gambar 3. 4 Gambar Rancangan *Hardware* Fiting Lampu

Pada rancangan *hardware* fitting lampu terdapat beberapa komponen yang terbagi menjadi input, proses, dan output. Pada input terdapat sensor *PIR* yang berfungsi sebagai alat pendeteksi keberadaan manusia jika terdapat gerakan yang melewati sensor tersebut. Sensor ini dimanfaatkan untuk membuat fitting lampu bersifat otomatis. Selain sensor *PIR* pada input juga terdapat sensor *PZEM 004T* yang berfungsi untuk mengukur tegangan, arus, dan daya listrik pada fitting lampu tersebut, sehingga pengguna yang menggunakan fitting lampu ini dapat mengetahui berapa besar daya yang telah digunakan secara *real time*. Komponen terakhir yang terdapat pada input adalah *RTC* sebagai alat untuk pengaturan waktu. Kemudian pada proses terdapat komponen yang bernama *NodeMCU* yang berfungsi sebagai pengolah data, data yang diolah didapat dari sensor *PIR*, sensor *PZEM 004T*, dan *RTC*. Pada output terdapat modul *SD Card* yang berfungsi

sebagai modul untuk membaca dan menulis data baik ke *SD Card* maupun dari *SD Card*. Selain *SD Card* pada output juga terdapat *solid state* relay yang berfungsi sebagai saklar. Output yang terakhir adalah *LCD Oled 0.91*. LCD ini berfungsi sebagai penampil pengaturan *timer*.

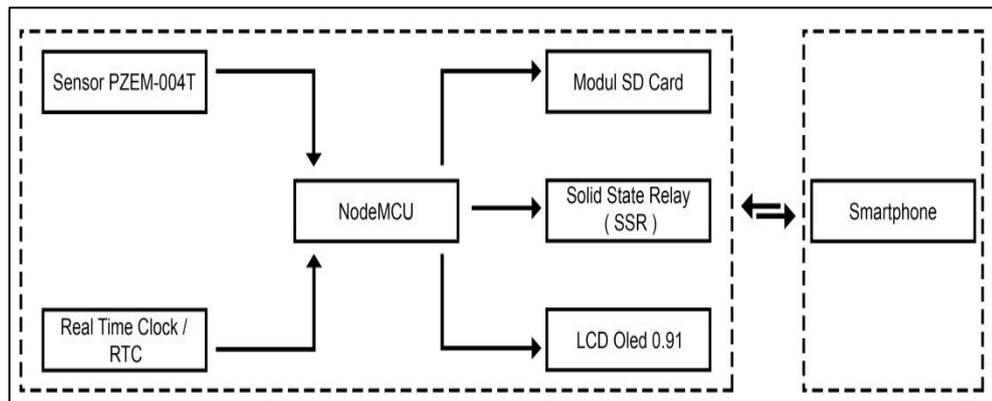
Dari hasil pengolahan data yang dikirimkan pada aplikasi pada *android* yang telah dibuat dengan *software App Inventor* melalui jaringan *WiFi* dengan menggunakan kontroler *NodeMCU ESP8266*. *Android* dapat menerima data serta mengirimkan perintah pada *NodeMCU ESP8266* untuk men-*trigger solid state relay* sehingga dapat mematikan dan menghidupkan lampu yang digunakan. Berikut skematik fitting lampu dapat ditunjukkan pada Gambar 3.5 berikut.



Gambar 3. 5 Gambar Skematik Fiting Lampu

3.4.2 Rancangan *Hardware* Stopkontak

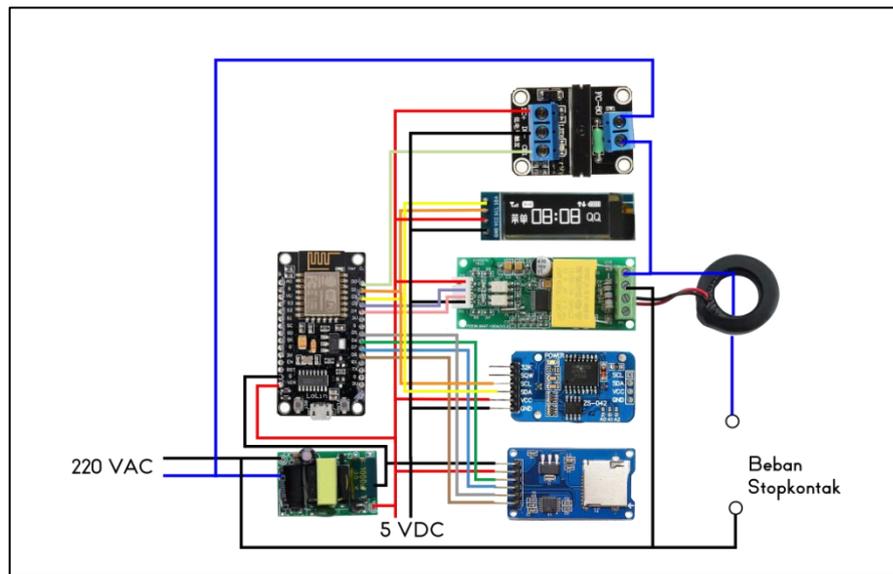
Selain proses pembuatan *hardware* fitting lampu terdapat juga rancangan *hardware* stopkontak pada proyek akhir kali ini. Pembuatan rancangan *hardware* stopkontak ini dibuat untuk memenuhi spesifikasi pembuatan alat yang diinginkan. Berikut adalah rancangan *hardware* pada stopkontak yang akan ditunjukkan pada Gambar 3.6.



Gambar 3. 6 Gambar Rancangan *Hardware* Stopkontak

Pada rancangan *hardware* kali ini terdapat perbedaan yang terletak pada input. Inputnya hanya terdapat sensor *PZEM 004T* dan *RTC* saja. Dengan fungsinya yang masih sama yaitu yang berfungsi untuk mengukur tegangan, arus, dan daya listrik pada stopkontak tersebut, sehingga pengguna yang menggunakan stopkontak ini dapat mengetahui berapa besar arus, tegangan, dan daya yang telah digunakan secara *real time*. Komponen selanjutnya yang digunakan pada input adalah *RTC* sebagai alat untuk pengaturan waktu seperti hari, tanggal, dan jam secara *real time*. Kemudian pada proses terdapat komponen yang bernama *NodeMCU* berfungsi sebagai pengolah data, data yang diolah didapat dari sensor *PZEM 004T*, dan *RTC*. Pada output terdapat modul *SD Card* yang berfungsi sebagai modul untuk membaca dan menulis data baik ke *SD Card* maupun dari *SD Card*, *SD Card* juga berfungsi untuk menyimpan data yang sudah diolah oleh *NodeMCU*. Selain *SD Card* pada output juga terdapat *solid state relay* yang berfungsi sebagai saklar. Output yang terakhir adalah *LCD Oled 0.91*. LCD ini berfungsi sebagai penampil pengaturan *timer*.

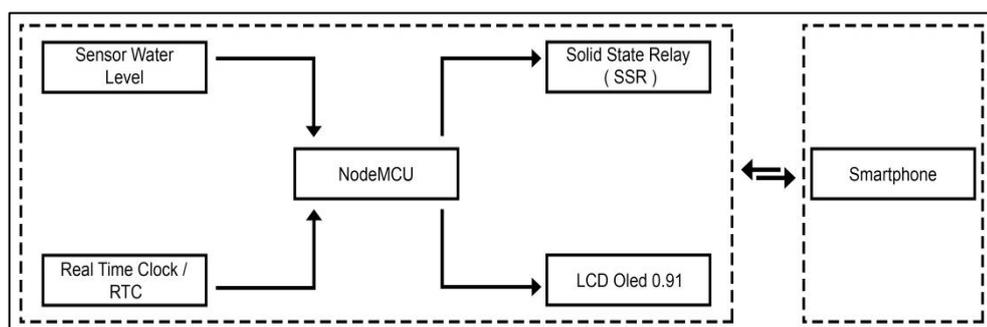
Dari hasil pengolahan data yang didapat selanjutnya dikirimkan pada aplikasi *android* yang telah dibuat dengan *software App Inventor* melalui jaringan *WiFi* dengan menggunakan kontroler *NodeMCU ESP8266*. *Android* dapat menerima data serta mengirimkan perintah pada *NodeMCU ESP8266* untuk *trigger solid state relay* sehingga dapat mematikan dan menghidupkan stopkontak yang digunakan. Berikut skematik stopkontak dapat ditunjukkan pada Gambar 3.7 berikut.



Gambar 3. 7 Gambar Skematik Stopkontak

3.4.3 Rancangan *Hardware* Saklar Pompa Air

Rancangan *hardware* selanjutnya adalah rancangan *hardware* Saklar pompa air. Pembuatan rancangan *hardware* Saklar Pompa Air ini dibuat untuk memenuhi spesifikasi pembuatan alat yang diinginkan. Berikut adalah gambar rancangan *hardware* saklar pada pompa air yang ditunjukkan pada Gambar 3.8.

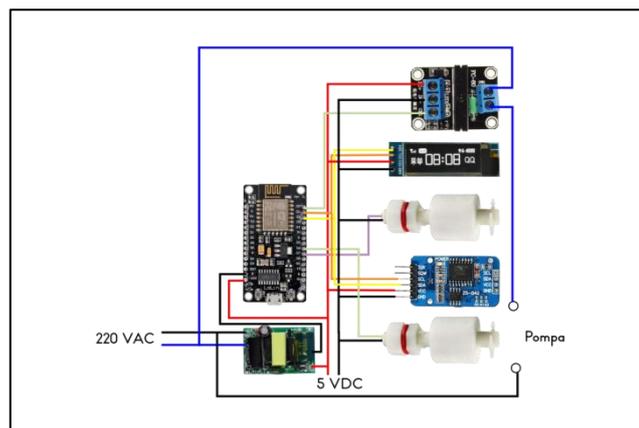


Gambar 3. 8 Rancangan Saklar Pompa Air

Pada rancangan *hardware* kali ini sensor yang digunakan adalah sensor *water level* yang berfungsi sebagai alat untuk mendeteksi ada atau tidaknya air didalam suatu tempat. Komponen selanjutnya yang digunakan pada input masih sama seperti rancangan *hardware* sebelumnya yaitu adalah *RTC* sebagai alat untuk pengaturan waktu seperti hari, tanggal, dan jam secara *real time*. Kemudian pada proses terdapat komponen yang bernama *NodeMCU* dengan fungsi yang

masih sama sebagai pengolah data, data yang diolah didapat dari sensor *water level*, dan *RTC*. Pada output sedikit berbeda karena tidak terdapat modul *SD Card*. Namun terdapat *solid state relay* yang berfungsi sebagai saklar. Output yang terakhir adalah *LCD Oled 0.91* yang berfungsi sebagai penampil data yang diinginkan.

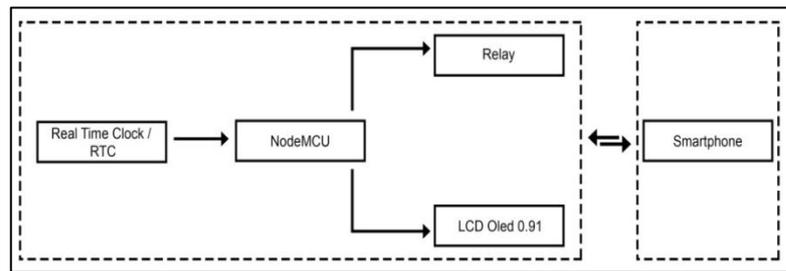
Data dikirimkan pada aplikasi *android* yang telah dibuat dengan software *App Inventor* melalui jaringan *WiFi* dengan menggunakan kontroler *NodeMCU ESP8266*. *Android* dapat menerima data serta mengirimkan perintah pada *NodeMCU ESP8266* untuk *men-trigger solid state relay* sehingga dapat mematikan dan menghidupkan pompa air yang digunakan. Berikut skematik saklar pompa air dapat ditunjukkan pada Gambar 3.9 berikut.



Gambar 3. 9 Gambar Skematik Saklar Pompa Air

3.4.4 Rancangan *Hardware* Saklar Gerendel

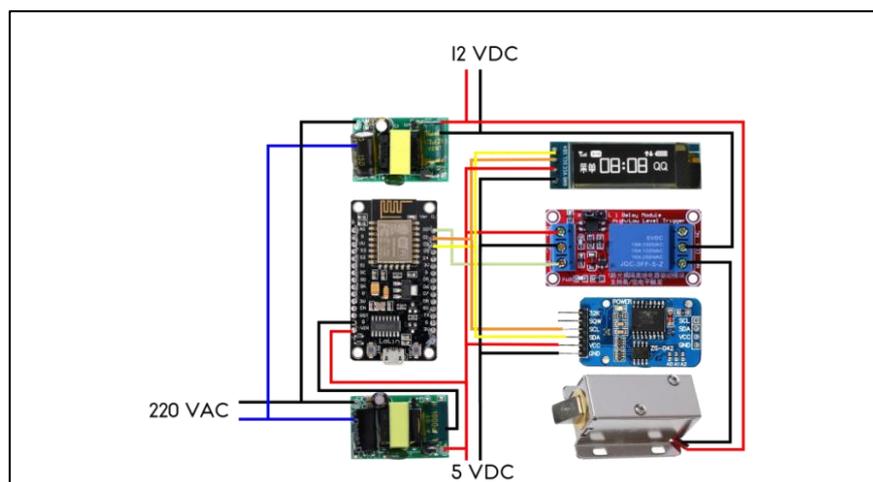
Rancangan *hardware* selanjutnya adalah rancangan *hardware* saklar gerendel. Pembuatan rancangan *hardware* gerendel ini dibuat untuk memenuhi spesifikasi pembuatan alat yang diinginkan. Berikut adalah rancangan *hardware* saklar gerendel yang akan ditunjukkan pada Gambar 3.10.



Gambar 3. 10 Rancangan Saklar Gerendel

Pada rancangan *hardware* kali ini terletak perbedaan pada input yang hanya terdapat *RTC* sebagai alat untuk pengaturan waktu seperti hari, tanggal, dan jam secara *real time*. kemudian pada proses terdapat komponen yang bernama *NodeMCU* dengan fungsi yang masih sama sebagai pengolah data, data yang diolah didapat dari perintah *smartphone*. Pada output hanya terdapat dua komponen yaitu *relay* dan *LCD Oled 0.91*. *Relay* berfungsi sebagai pemutus dan penghubung arus yang menuju ke Gerendel. Outputnya adalah *LCD Oled 0.91*. *LCD* ini berfungsi sebagai penampil data yang diinginkan.

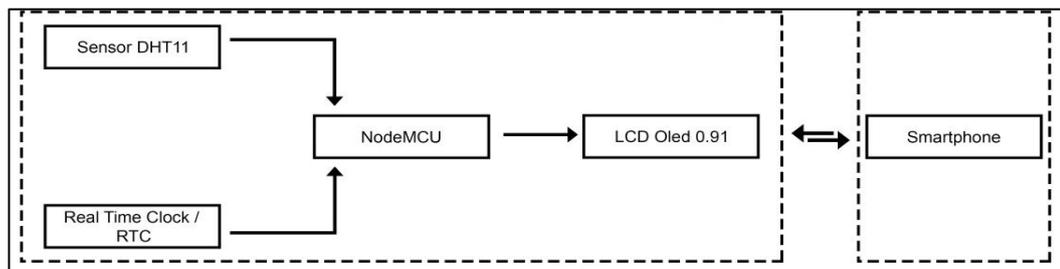
Dari hasil pengolahan data yang dikirimkan pada aplikasi *android* yang telah dibuat dengan *software App Inventor* melalui jaringan *WiFi* dengan menggunakan kontroler *NodeMCU ESP8266*. *Android* dapat menerima data serta mengirimkan perintah pada *NodeMCU ESP8266* untuk *men-trigger relay* sehingga dapat mematikan dan menghidupkan *solenoid* yang digunakan. Berikut skematik saklar grendel dapat ditunjukkan pada Gambar 3.11 berikut.



Gambar 3. 11 Gambar Skematik Saklar Gerendel

3.4.5 Rancangan *Hardware* Termometer

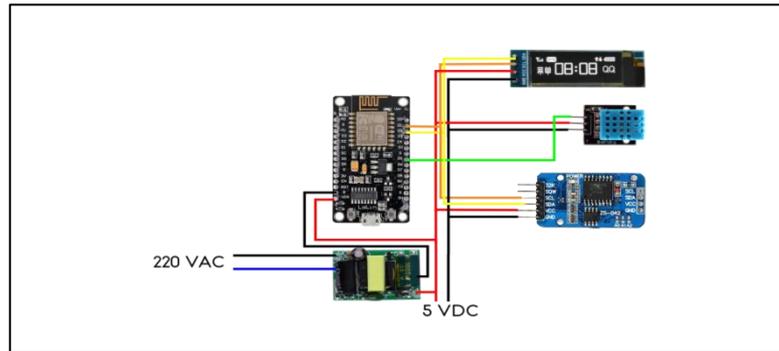
Pembuatan rancangan *hardware* termometer ini dibuat untuk memenuhi spesifikasi pembuatan alat yang diinginkan. Berikut adalah rancangan *hardware* pada termometer yang akan ditunjukkan pada Gambar 3.12 berikut ini.



Gambar 3. 12 Rancangan *Hardware* Termometer

Terdapat beberapa komponen yang telah dibagi dalam input, proses dan output. Inputnya hanya terdapat sensor *DHT11* dan *RTC* saja. Sensor *DHT11* berfungsi untuk membaca suhu atau kelembaban suatu ruangan secara *real time*. Komponen selanjutnya yang digunakan pada input adalah *RTC* sebagai alat untuk pengaturan waktu seperti hari, tanggal, dan jam secara *real time*. Kemudian pada proses terdapat komponen yang bernama *NodeMCU* dengan fungsi yang masih sama sebagai pengolah data, data yang diolah didapat dari sensor *DHT11* dan *RTC*. Pada output hanya terdapat *LCD Oled 0.91* yang berfungsi sebagai penampil data yang diinginkan.

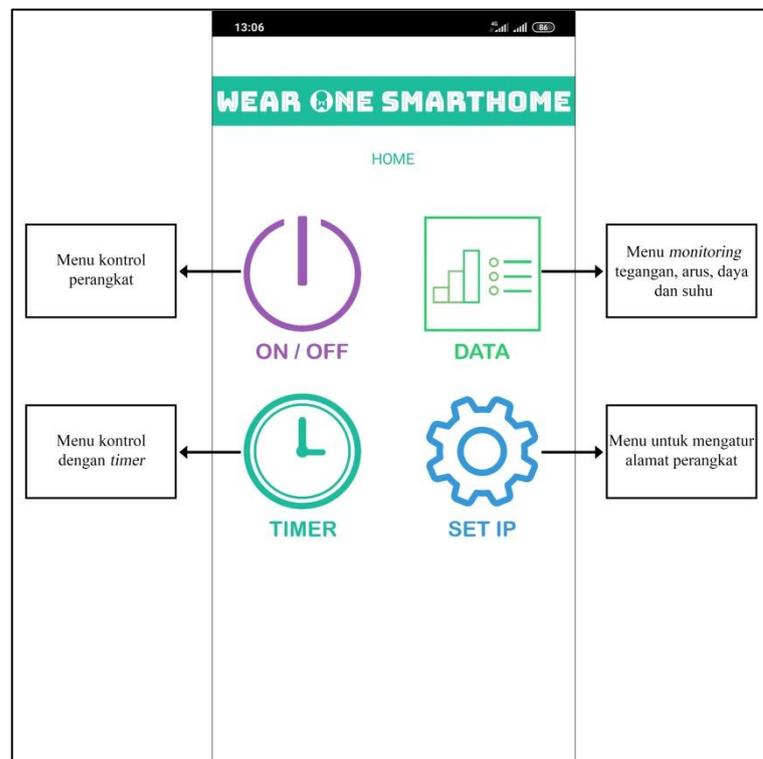
Dari hasil pengolahan data dikirimkan pada aplikasi *android* yang telah dibuat dengan *software App Inventor* melalui jaringan *WiFi* dengan menggunakan kontroler *NodeMCU ESP8266*. *Android* dapat menerima data serta mengirimkan perintah pada *NodeMCU ESP8266* untuk *men-trigger sensor* sehingga dapat membaca suhu pada lingkungan tersebut. Berikut skematik termometer dapat ditunjukkan pada Gambar 3.13 berikut.



Gambar 3. 13 Gambar Skematik Termometer

3.5 Rancangan Aplikasi

Pada pembuatan proyek akhir yang berjudul “Integrasi Perangkat-perangkat Pada *smart home* Menggunakan *smartphone*” memiliki rancangan aplikasi untuk masing-masing alat. Aplikasi ini dirancang menggunakan *App MIT Inventor*. Aplikasi yang telah dibuat dapat diinstal melalui *android*. Berikut gambar pada aplikasi akan ditunjukkan pada Gambar 3.14.

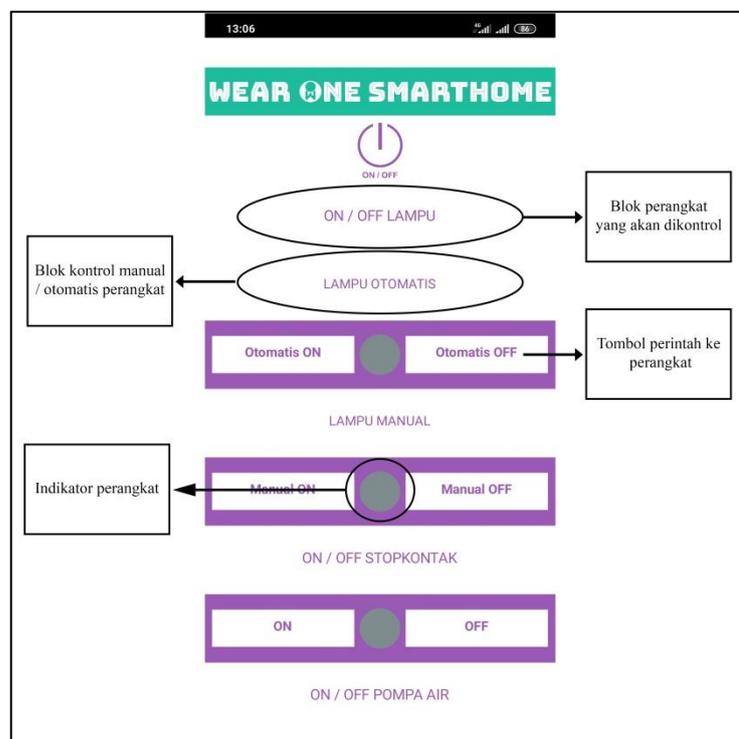


Gambar 3. 14 Gambar Menu Pada Aplikasi *Smartphone*

Pada Aplikasi tersebut terdapat 4 menu berupa menu *ON/OFF*, set IP, data, dan *TIMER*. Didalam menu-menu tersebut masih terdapat sub menu yang berisi fungsi yang sesuai dengan fungsi-fungsi dari perangkat-perangkat *smart home*.

3.5.1 Sub Menu *ON/OFF*

Sub menu ini berfungsi sebagai pengatur unntuk mengaktifkan atau tidak mengaktifkan perangkat-perangkat tersebut dengan kendali *smartphone*. Berikut gambar dari sub menu *ON/OFF* yang akan ditunjukkan oleh Gambar 3.15.

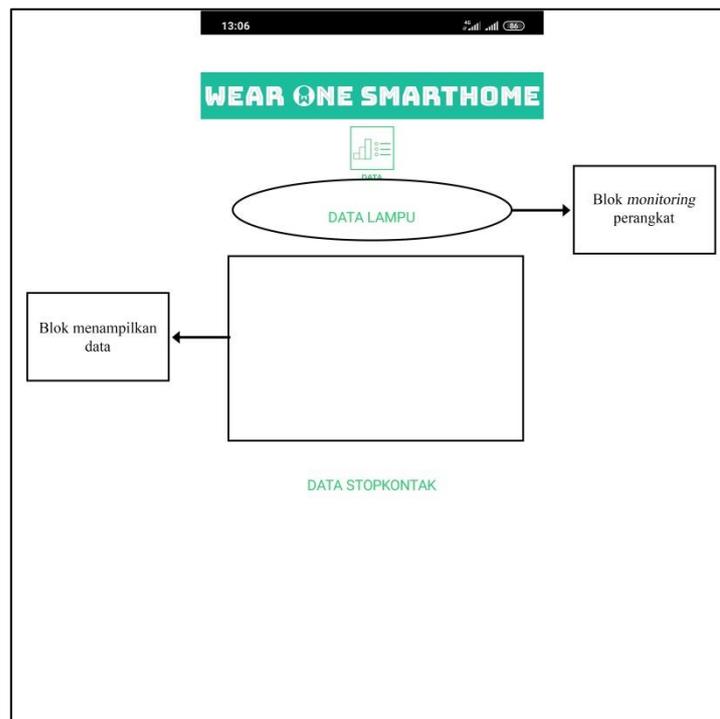


Gambar 3. 15 Gambar Sub Menu *ON/OFF*

Terdapat enam fungsi yang dapat diaktifkan atau tidak diaktifkan melalui sub menu ini yaitu *ON/OFF* lampu otomatis, *ON/OFF* lampu manual, *ON/OFF* Stopkontak, *ON/OFF* pompa air otomatis, *ON/OFF* pompa air manual, *ON/OFF* gerendel. Pada setiap sub menu jika menekan tombol *ON* maka perangkat akan aktif dengan ditandai indikator berwarna hijau pada *smartphone* muncul. Jika menekan tombol *OFF* maka perangkat akan mati dengan ditandai lampu indikator warna merah pada *smartphone* muncul.

3.5.2 Sub Menu Data

Sub menu ini berfungsi melihat data-data pada perangkat-perangkat tersebut dengan kendali *smartphone*. Berikut gambar dari sub menu Data yang akan ditunjukkan pada Gambar 3.16.

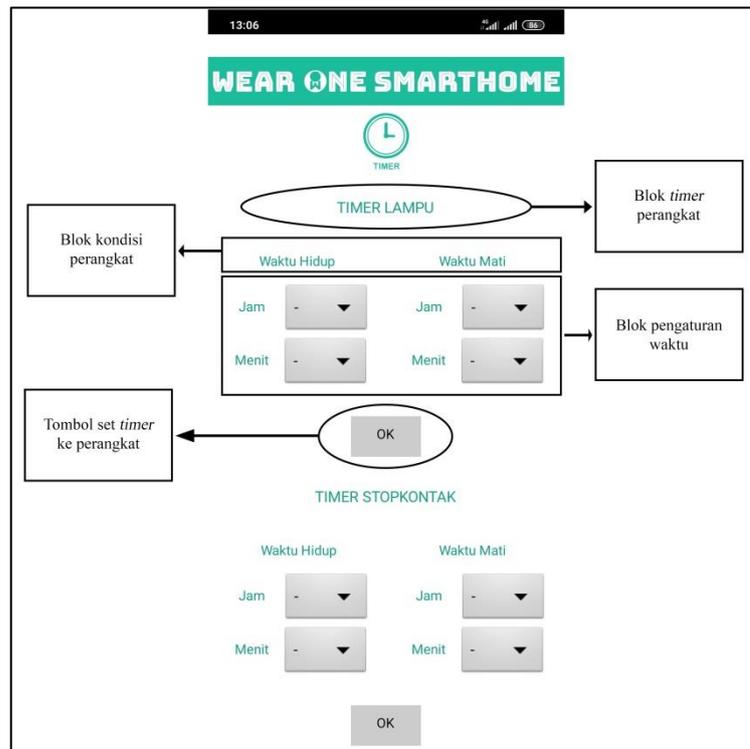


Gambar 3. 16 Gambar Sub Menu Dari Data

Menu ini dapat melihat data arus, tegangan dan daya, serta data suhu ruangan. Terdapat tiga data perangkat pada sub menu ini yaitu data lampu, data stopkontak, data termometer. Untuk melihat data, perangkat harus terhubung ke aplikasi. Sehingga data akan tampil pada layar *smartphone*.

3.5.3 Sub Menu *Timer*

Sub menu *timer* berfungsi sebagai pengatur untuk mengaktifkan atau menonaktifkan perangkat dengan pengaturan waktu yang ditentukan. Berikut gambar pada sub menu *timer* pada Gambar 3.17.

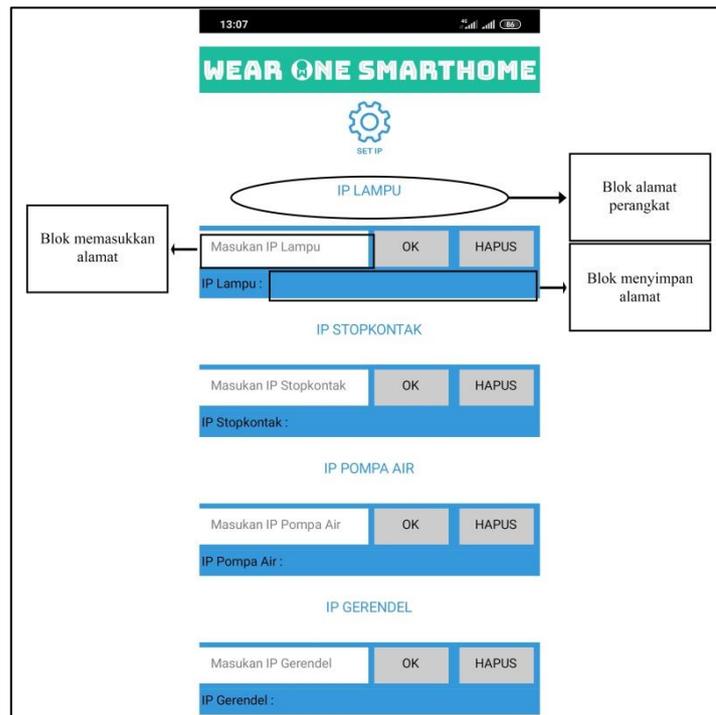


Gambar 3. 17 Gambar Sub Menu Dari Timer

Terdapat dua buah perangkat yang dapat diatur *timernya* melalui sub menu ini yaitu *Timer* untuk lampu dan *Timer* untuk stopkontak. Untuk menampilkan tampilan *timer* pada sub menu ini. Pertama-tama klik perangkat mana yang akan diatur waktunya, kemudian akan muncul tampilan dengan tulisan “waktu hidup” yang berfungsi sebagai pengatur kapan waktu perangkat tersebut akan dihidupkan dan tulisan “waktu mati” yang berfungsi sebagai pengatur kapan perangkat tersebut harus dimatikan.

3.5.4 Sub Menu Set IP

Sub menu ini berfungsi sebagai pengatur alamat jaringan dan perangkat yang akan dihubungkan melalui kendali *smartphone*. Berikut gambar sub menu Set IP pada Gambar 3.18.



Gambar 3. 18 Gambar Sub Menu Dari Set IP

Terdapat lima buah perangkat yang dapat diatur set IPnya melalui sub menu ini yaitu Set IP lampu, Set IP Stopkontak, Set IP Pompa Air, Set Gerendel, Set Termometer. Untuk mengatur set IP maka perlu mengklik set IP perangkat mana yang akan diatur. Setelah muncul tampilan dari sub menu set IP selanjutnya masukan alamat IP dari perangkat-perangkat *smart home* tersebut, sehingga perangkat-perangkat tersebut dapat terhubung dengan *smartphone*.

BAB IV

PEMBAHASAN

Untuk memastikan fungsi alat yang telah dibuat sesuai dengan fungsi yang diinginkan, maka pada bab ini akan membahas pengujian perangkat-perangkat yang akan digunakan dalam proyek akhir yang berjudul “Integrasi Perangkat-perangkat *Smart Home* Menggunakan *Smartphone*“. Adapun untuk melakukan pengujian pada aplikasi yang dibuat dapat dilakukan dengan langkah-langkah sebagai berikut:

1. Hidupkan *hotspot* pada perangkat *Android*.
2. Tunggu hingga perangkat terhubung (*connected*) pada *android*.
3. Buka aplikasi “*WEAR ONE SMART HOME*”.
4. Masuk ke menu *setting* lalu setting IP masing-masing perangkat. Tekan tombol *OK* lalu kembali ke menu utama.
5. Masuk ke menu *ON/OFF*. Untuk mengontrol perangkat.
6. Setelah menghidupkan perangkat masuk ke menu *monitoring* untuk *monitoring* arus, tegangan, daya dan suhu yang terbaca oleh sensor.
7. Masuk ke menu *timer* untuk mengatur jadwal hidup atau mati pada perangkat.

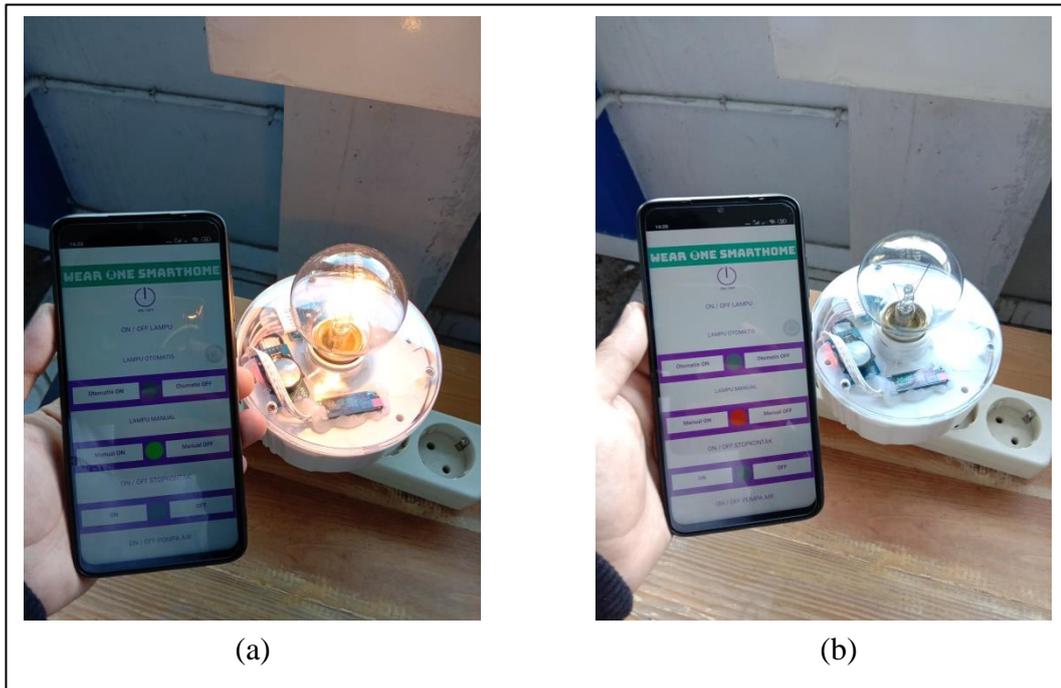
Adapun hasil dari pengujian yang dilakukan adalah sebagai berikut :

4.1 Pengujian Fiting Lampu

Pada perangkat fiting lampu, yang akan diuji adalah *ON/OFF* fiting lampu manual, *ON/OFF* fiting lampu otomatis, *monitoring* data, dan *ON/OFF* dengan *timer*. Berikut hasil pengujian pada perangkat fiting lampu :

4.1.1 Pengujian *ON/OFF* Fiting Lampu Manual

Pada pengujian ini perintah yang diberikan adalah menyalakan dan mematikan lampu dengan mode manual. Setelah dilakukan pengujian maka kondisi lampu yang didapatkan dapat dilihat pada Gambar 4.1 berikut ini.



Gambar 4. 1 (a) Menyalakan Lampu (b) Mematikan Lampu

Adapun kondisi fitting lampu manual yang telah diuji akan ditampilkan dalam bentuk tabel. Berikut pengujian fitting lampu mode manual yang akan ditunjukkan oleh Tabel 4.1 berikut ini.

Tabel 4. 1 Hasil Pengujian Fiting Lampu Mode Manual

NO	Perintah	Kondisi Lampu
1	Manual <i>ON</i>	Menyala
2	Manual <i>OFF</i>	Tidak Menyala

Ketika perintah manual *ON* dikirimkan dari *smartphone* maka kondisi lampu menyala. Ketika perintah manual *OFF* dikirimkan dari *smartphone* maka kondisi lampu tidak menyala.

4.1.2 Pengujian *ON/OFF* Fiting Lampu Otomatis

Adapun kondisi fitting lampu yang telah diuji akan ditampilkan dalam bentuk tabel. Berikut pengujian fitting lampu mode otomatis yang akan ditunjukkan oleh Tabel 4.2 berikut ini.

Tabel 4. 2 Hasil Pengujian Fiting Lampu Mode Otomatis

NO	Perintah	Sensor	Kondisi Lampu
1	Otomatis <i>ON</i>	Ada orang	Menyala
2	Otomatis <i>ON</i>	Tidak ada orang	Tidak menyala
3	Otomatis <i>OFF</i>	Ada orang	Tidak menyala
4	Otomatis <i>OFF</i>	Tidak ada orang	Tidak menyala

Ketika perintah otomatis *ON* dikirimkan dari *smartphone* maka perangkat akan menyimpan perintah tersebut setelah itu perangkat akan membaca data dari sensor. Setelah kedua kondisi tersebut terpenuhi maka lampu akan menyala. Apabila ada salah satu kondisi atau perintah tidak terpenuhi maka lampu tidak menyala.

4.1.3 Pengujian *Monitoring Data* Pada Fiting Lampu

Pengujian pada fitting lampu tidak hanya pengujian pada penggunaan sistem otomatis dan manual saja dalam mengendalikan fitting lampu. Pada fitting lampu juga terdapat data yang dapat di *monitoring* melalui *smartphone*. Data yang dikirimkan adalah data yang dibaca oleh sensor *PZEM-004T*. Data dari sensor dibaca dengan program. Berikut adalah *list* program pengujian pada sensor *PZEM-004T*.

```
void loop()
{
    float voltage = pzem.voltage();
    float current = pzem.current();
    float power = pzem.power();

    Serial.print(voltage); Serial.println("V");
    Serial.print(current); Serial.println("A");
    Serial.print(power); Serial.println("W");
}
```

Data dari fitting lampu yang telah diuji akan ditampilkan dalam bentuk tabel. Data juga dibandingkan dengan data yang didapat dari hasil pengukuran. Berikut tabel perbandingan data tegangan yang akan ditunjukkan pada Tabel 4.3 berikut ini.

Tabel 4. 3 Data Perbandingan Tegangan

NO	Beban	Percobaan	Tegangan (V)		Perbandingan (%)
			Perangkat	Pengukuran	
1	Lampu 8 Watt	1	220,60	220,2	0,18
		2	219,50	220,2	0,31
		3	219,90	220,2	0,13
2	Lampu 15 Watt	1	219,80	220,3	0,22
		2	219,10	220,3	0,54
		3	218,90	220,3	0,63

Dari data tabel didapatkan persentase perbandingan tegangan beban lampu 8 watt berada dibawah 0,4% sedangkan perbandingan tegangan beban lampu 15 watt berada dibawah 0,7%. Sehingga dapat disimpulkan pembacaan data tegangan dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

Selain pengujian data tegangan, selanjutnya dilakukan pengujian data arus pada beban. Berikut tabel perbandingan data arus yang akan ditunjukkan pada Tabel 4.4 berikut ini.

Tabel 4. 4 Data Perbandingan Arus

NO	Beban	Percobaan	Arus (A)		Perbandingan (%)
			Perangkat	Pengukuran	
1	Lampu 8 Watt	1	0,06	0,06	0
		2	0,06	0,06	0
		3	0,06	0,06	0
2	Lampu 15 Watt	1	0,07	0,07	0
		2	0,07	0,07	0
		3	0,07	0,07	0

Dari tabel persentase perbandingan arus antara data yang dikirimkan ke *smartphone* dan data yang diukur dengan alat ukur didapatkan persentase perbandingan arus beban lampu 8 watt sebesar 0% sedangkan perbandingan arus

beban lampu 15 watt sebesar 0%. Sehingga dapat disimpulkan pembacaan data arus dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

Setelah pengujian data tegangan dan arus, selanjutnya dilakukan pengujian data daya pada beban. Data daya yang ditampilkan di *smartphone* akan dibandingkan dengan data dari alat ukur yang dihitung menggunakan persamaan 2.2 agar mendapatkan data daya aktif. Berikut tabel perbandingan data daya yang akan ditunjukkan pada Tabel 4.5 berikut ini.

Tabel 4. 5 Data Perbandingan Daya

NO	Beban	Percobaan	Daya (Watt)		Perbandingan (%)
			Perangkat	Perhitungan	
1	Lampu 8 Watt	1	8,00	7,92	1,01
		2	7,90	7,92	0,25
		3	8,00	7,92	1,01
2	15 Watt	1	15,80	14,95	5,68
		2	15,80	14,95	5,68
		3	15,80	14,95	5,68

Dari data tabel didapatkan persentase perbandingan daya beban lampu 8 watt berada dibawah 1,1% sedangkan perbandingan daya beban lampu 15 watt berada dibawah 5,7%. Sehingga dapat disimpulkan pembacaan data daya dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

4.1.4 Pengujian Pengaturan *Timer* Pada Fiting Lampu

Adapun pengaturan *timer* yang telah diuji akan ditampilkan dalam bentuk tabel. Berikut pengujian pengaturan *timer* fitting lampu yang akan ditunjukkan oleh Tabel 4.6 berikut ini.

Tabel 4. 6 Hasil Pengaturan *Timer* Fiting Lampu

NO	Atur waktu	Waktu sekarang	Lampu
1	14.20	14.20	Menyala
2	14.25	14.25	Tidak menyala

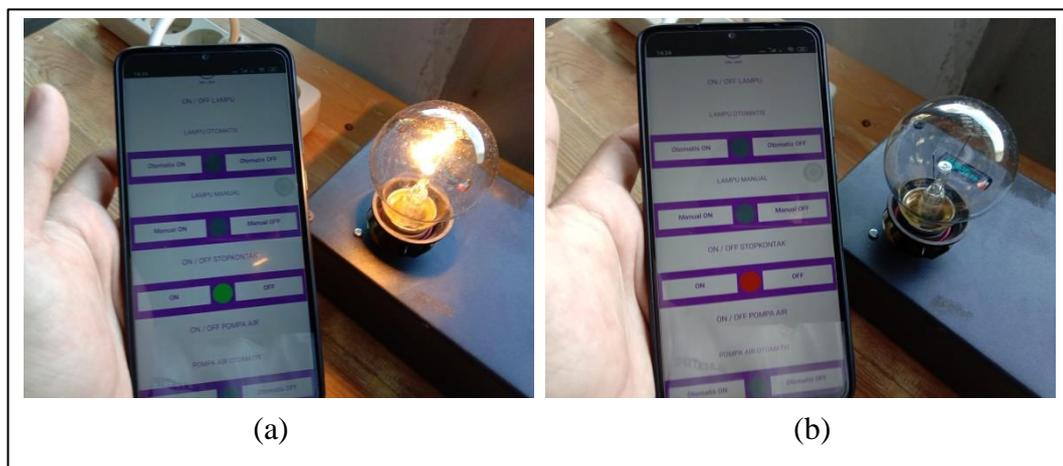
Dari data tabel didapatkan kesimpulan ketika *timer* disetting dari *smartphone* maka data *timer* akan disimpan pada perangkat. Ketika kondisi *timer* waktu hidup terpenuhi maka apabila lampu dalam kondisi tidak menyala menjadi menyala. Ketika *timer* waktu mati terpenuhi kondisi lampu tidak menyala.

4.2 Pengujian Stopkontak

Pada perangkat stopkontak, yang akan diuji adalah pengujian *ON / OFF* pada stopkontak, pengujian *monitoring* data pada stopkontak, pengujian *ON / OFF* dengan *timer* pada stopkontak. Berikut uraian pengujian yang telah dilakukan :

4.2.1 Pengujian Stopkontak *ON/OFF*

Pada pengujian ini akan dilakukan menyalakan lampu menggunakan mode manual dan juga mematikan lampu menggunakan mode manual. Berikut gambar stopkontak pada saat kondisi *ON/OFF* akan ditunjukkan pada Gambar 4.2 berikut.



Gambar 4. 2 (a) Menyalakan Stopkontak (b) Mematikan Stopkontak

Berikut pengujian *ON/OFF* stopkontak yang akan ditunjukkan oleh Tabel 4.7 berikut ini.

Tabel 4. 7 Hasil Pengujian *ON / OFF* Stopkontak

NO	Perintah	Kondisi Stopkontak
1	<i>ON</i>	Aktif
2	<i>OFF</i>	Tidak aktif

Ketika perintah *ON* dari *smartphone* dikirimkan maka stopkontak akan aktif dan jika perintah *OFF* dari *smartphone* dikirimkan maka stopkontak tidak aktif.

4.2.2 Pengujian *Monitoring Data Pada Stopkontak*

Pengujian selanjutnya adalah pengujian *monitoring* data pada stopkontak. Pada pengujian ini yang akan ditampilkan adalah data tegangan, arus, dan daya. Data yang dikirimkan adalah data yang dibaca oleh sensor *PZEM-004T*. Data dari sensor dibaca dengan program. Berikut adalah *list* program pengujian pada sensor *PZEM-004T*.

```
void loop()
{
  float voltage = pzem.voltage();
  float current = pzem.current();
  float power = pzem.power();
  Serial.print(voltage); Serial.println("V");
  Serial.print(current); Serial.println("A");
  Serial.print(power); Serial.println("W");
}
```

Adapun data dari stopkontak yang telah diuji akan ditampilkan dalam bentuk tabel. Data dari perangkat juga dibandingkan dengan data yang didapat dari hasil pengukuran dengan alat ukur. Berikut tabel perbandingan data tegangan yang akan ditunjukkan pada Tabel 4.8 berikut ini.

Tabel 4. 8 Data Perbandingan Tegangan

NO	Beban	Percobaan	Tegangan (V)		Perbandingan (%)
			Perangkat	Pengukuran	
1	Lampu 8 Watt	1	220,10	220,2	0,04
		2	219,50	220,2	0,31
		3	219,30	220,2	0,40
2	15 Watt	1	220,20	220,3	0,04
		2	219,10	220,3	0,54
		3	219,40	220,3	0,40

Dari data tabel didapatkan persentase perbandingan tegangan beban lampu 8 watt berada di bawah 0,5% sedangkan perbandingan tegangan beban lampu 15 watt berada di bawah 0,6%. Sehingga dapat disimpulkan pembacaan data tegangan dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

Selain pengujian data tegangan, selanjutnya dilakukan pengujian data arus pada beban. Pengujian data arus menggunakan alat ukur menggunakan rangkaian untuk membaca arus. Berikut tabel perbandingan data arus yang akan ditunjukkan pada Tabel 4.9 berikut ini.

Tabel 4. 9 Data Perbandingan Arus

NO	Beban	Percobaan	Arus (A)		Perbandingan (%)
			Perangkat	Pengukuran	
1	Lampu 8 Watt	1	0,06	0,06	0
		2	0,06	0,06	0
		3	0,06	0,06	0
2	Lampu 15 Watt	1	0,07	0,07	0
		2	0,07	0,07	0
		3	0,07	0,07	0

Dari tabel persentase perbandingan arus antara data yang dikirimkan ke *smartphone* dan data yang diukur dengan alat ukur didapatkan persentase perbandingan arus beban lampu 8 watt sebesar 0% sedangkan perbandingan arus beban lampu 15 watt sebesar 0%. Sehingga dapat disimpulkan pembacaan data arus dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

Setelah pengujian data tegangan dan arus, selanjutnya dilakukan pengujian data daya pada beban. Data daya yang ditampilkan di *smartphone* akan dibandingkan dengan data dari alat ukur yang dihitung menggunakan persamaan 2.2 supaya mendapatkan data daya aktif. Perbandingan data daya menggunakan perhitungan karena alat untuk mengukur daya aktif tidak dapat digunakan mengukur beban yang digunakan. Berikut tabel perbandingan data daya yang akan ditunjukkan pada Tabel 4.10 berikut ini.

Tabel 4. 10 Data Perbandingan Daya

NO	Beban	Percobaan	Daya (Watt)		Perbandingan (%)
			Perangkat	Perhitungan	
1	Lampu 8 Watt	1	8,20	7,92	3,53
		2	8,20	7,92	3,53
		3	8,30	7,92	4,79
2	15 Watt	1	16,00	14,95	7,02
		2	15,90	14,95	6,35
		3	16,10	14,95	7,69

Dari tabel persentase perbandingan daya antara data yang dikirimkan ke *smartphone* dan data yang dihitung didapatkan persentase perbandingan daya beban lampu 8 watt berada di bawah 4,8% sedangkan perbandingan daya beban lampu 15 watt berada di bawah 7,7%. Sehingga dapat disimpulkan pembacaan data daya dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

4.2.3 Pengujian Pengaturan *Timer* Pada Stopkontak

Berikut pengujian pengaturan *timer* pada stopkontak yang akan ditunjukkan oleh Tabel 4.11 berikut ini.

Tabel 4. 11 Hasil Pengujian Pengaturan *Timer* Stopkontak

NO	Atur waktu	Waktu sekarang	Stopkontak
1	14.10	14.10	Aktif
2	14.15	14.15	Tidak aktif

Dari data tabel didapatkan kesimpulan ketika *timer* disetting dari *smartphone* maka data *timer* akan disimpan pada perangkat. Ketika kondisi *timer* waktu hidup terpenuhi maka apabila stopkontak dalam kondisi tidak aktif menjadi aktif. Ketika *timer* waktu mati terpenuhi kondisi stopkontak menjadi tidak aktif.

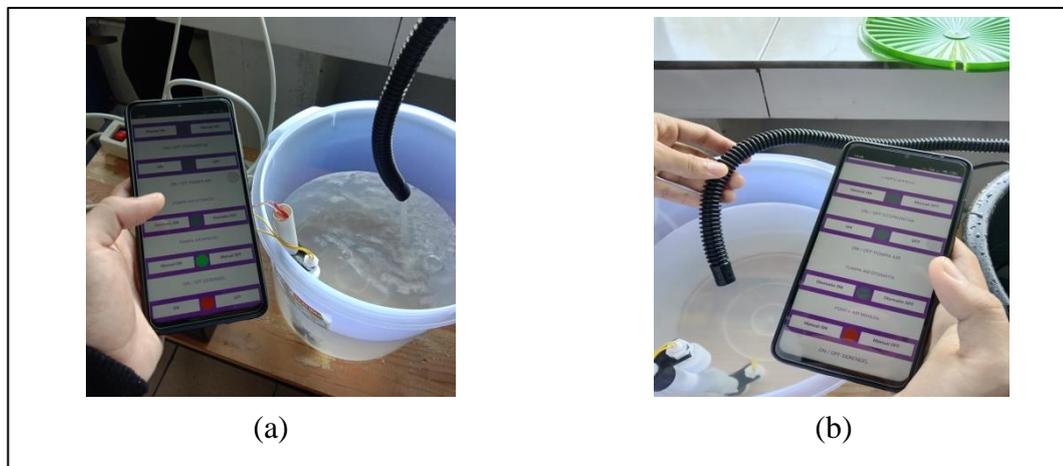
4.3 Pengujian Pompa Air

Pada perangkat pompa air, yang akan diuji adalah *ON / OFF* pompa air manual dan *ON/OFF* pompa air otomatis. Tangki air menggunakan toples ukuran

sedang. Sedangkan sensor yang digunakan bisa diaplikasikan pada tangki air besar. Berikut hasil pengujian pada perangkat pompa air :

4.3.1 Pengujian *ON/OFF* Pompa Air Manual

Pengujian pompa yang akan dilakukan adalah pengujian menghidupkan dan mematikan pompa air secara manual. Kondisi awal pompa air adalah *OFF* kemudian pompa air diaktifkan melalui *smartphone* sehingga kondisinya berubah menjadi *ON* dan juga sebaliknya. Berikut pengujian pompa air mode manual akan ditunjukkan oleh Gambar 4.3 berikut.



Gambar 4. 3 (a) Menyalakan Pompa Air (b) Mematikan Pompa Air

Berikut pengujian saklar *ON/OFF* yang dilakukan pada pompa air mode manual yang akan ditunjukkan oleh Tabel 4.12 berikut ini.

Tabel 4. 12 Hasil Pengujian Pompa Air Mode Manual

NO	Perintah	Kondisi pompa
1	Manual <i>ON</i>	Aktif
2	Manual <i>OFF</i>	Tidak aktif

Ketika perintah manual *ON* dikirimkan dari *smartphone* maka kondisi pompa air aktif. Ketika perintah manual *OFF* dikirimkan dari *smartphone* maka kondisi pompa air tidak aktif. Untuk mengganti mode aktivasi, pengguna harus mematikan mode yang sedang aktif.

4.3.2 Pengujian *ON/OFF* Pompa Air Otomatis

Untuk melihat perubahan indikator pada menu *ON/OFF* pengguna harus keluar dari menu *ON/OFF* lalu masuk lagi ke menu tersebut. Berikut pengujian *ON/OFF* pompa air mode otomatis yang akan ditunjukkan oleh Tabel 4.13 berikut ini.

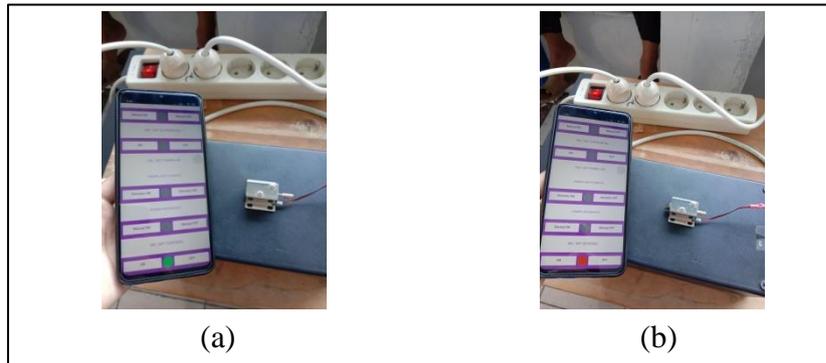
Tabel 4. 13 Hasil Pengujian *ON / OFF* Pompa Air Mode Otomatis

NO	Perintah	Kondisi sensor		Pompa
		Atas	Bawah	
1	Otomatis <i>ON</i>	<i>Low</i>	<i>Low</i>	Aktif
2	Otomatis <i>ON</i>	<i>Low</i>	<i>High</i>	Aktif
3	Otomatis <i>ON</i>	<i>High</i>	<i>High</i>	Tidak aktif
4	Otomatis <i>ON</i>	<i>High</i>	<i>Low</i>	Tidak aktif
5	Otomatis <i>OFF</i>	<i>Low</i>	<i>Low</i>	Tidak aktif
6	Otomatis <i>OFF</i>	<i>Low</i>	<i>High</i>	Tidak aktif
7	Otomatis <i>OFF</i>	<i>High</i>	<i>High</i>	Tidak aktif
8	Otomatis <i>OFF</i>	<i>High</i>	<i>Low</i>	Tidak aktif

Dari tabel dapat disimpulkan pompa hanya akan aktif apabila perintah otomatis *ON* dikirimkan dan kondisi sensor atas *low* dan sensor bawah *low* atau *high*. Selain perintah dan kondisi tersebut maka pompa tidak aktif.

4.4 Pengujian Gerendel

Pengujian selanjutnya adalah pengujian *ON / OFF* gerendel. Pada pengujian ini gerendel akan aktif jika tombol *ON* melalui *smartphone* ditekan. Pengujian gerendel kondisi on akan ditunjukkan oleh Gambar 4.4 berikut ini.



Gambar 4. 4 (a) Gerendel Kondisi Aktif (b) Gerendel Kondisi Tidak Aktif

Berikut pengujian *ON / OFF* gerendel yang akan ditunjukkan oleh Tabel 4.14 berikut ini.

Tabel 4. 14 Hasil Pengujian *ON / OFF* Gerendel

NO	Perintah	Kondisi pompa
1	<i>ON</i>	Aktif
2	<i>OFF</i>	Tidak aktif

Ketika perintah *ON* dikirimkan dari *smartphone* maka kondisi gerendel aktif. Ketika perintah *OFF* dikirimkan dari *smartphone* maka kondisi gerendel tidak aktif.

4.5 Pengujian Termometer

Pengujian pada termometer adalah pengujian *monitoring* data suhu yang ada dilingkungan perangkat. Data yang dibaca oleh sensor diolah di *NodeMCU* setelah itu ditampilkan pada *LCD OLED* secara *realtime* dan data juga dikirimkan ke *smartphone* dan bisa dilihat melalui *smartphone*. Berikut adalah *list* program pengujian pada sensor *DHT 11*.

```
void loop()
{
  delay(5000); } Menunggu selama 5 detik
  float t = dht.readTemperature(); } Membaca data sensor
  Serial.print(t);
  Serial.println("*C"); } Menampilkan data sensor
}
```

Adapun data dari termometer yang telah diuji dengan *smartphone* dan dengan alat ukur akan ditampilkan dalam bentuk tabel. Berikut pengujian *monitoring* data termometer yang akan ditunjukkan pada Tabel 4.15 berikut ini.

Tabel 4. 15 Hasil Pengujian *Monitoring* Pada Termometer

NO	Percobaan	Suhu (°C)		Perbandingan (%)
		Perangkat	Pengukuran	
1	Dalam Ruangan	30,30	30	1
	Luar Ruangan	27,20	27	0,74

Dari tabel persentase perbandingan suhu antara data yang dikirimkan ke *smartphone* dan data pada alat ukur didapatkan persentase perbandingan suhu di dalam ruangan sebesar 1% sedangkan perbandingan suhu di luar ruangan sebesar 0,74%. Sehingga dapat disimpulkan pembacaan data suhu dari perangkat yang dibuat sesuai dengan perancangan yang diinginkan.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pengujian yang telah dilakukan pada masing-masing perangkat pada proyek akhir yang berjudul “Integrasi Perangkat-Perangkat *Smart Home* menggunakan *Smartphone*” maka dapat ditarik kesimpulan sebagai berikut :

1. Data tegangan, arus, daya, dan suhu yang ada pada perangkat dapat ditampilkan pada aplikasi di *android*.
2. Semua perangkat dapat terhubung ke aplikasi secara bersamaan namun tetap memiliki waktu antrian data 1-2 detik, tergantung koneksi *hotspot*.
3. Pengontrolan *ON/OFF* perangkat dengan *smartphone* berfungsi dengan baik.
4. Mode aktivasi otomatis pada perangkat fitting lampu dan saklar pompa air berfungsi dengan baik.
5. Persentase perbandingan eror daya beban lampu 8 watt berada di bawah 4,8% sedangkan perbandingan eror daya beban lampu 15 watt berada di bawah 7,7%.

5.2 Saran

Apabila alat ini akan dikembangkan secara lanjut, maka perlunya beberapa saran agar alat berfungsi lebih baik lagi. saran yang akan diberikan antara lain :

1. Proses komunikasi antara perangkat dan *android* menggunakan topologi jaringan yang lain.
2. Pembuatan perangkat menggunakan spesifikasi komponen yang lebih tinggi.
3. Menggunakan database khusus untuk data.
4. Proses komunikasi menggunakan sistem *online*.

DAFTAR PUSTAKA

- [1] Abdul Jafar Hakim, "PROTOTYPE SMART HOME DENGAN KONSEP INTERNET OF THING (IOT) MENGGUNAKAN ARDUINO BERBASIS WEB," p. 1, September 2015.
- [2] K'K Angrainy and Steven Novaldy F, "Stopkontak Pintar Berbasis Android," *Proyek Akhir*, Mei 2018.
- [3] yogie saputra and habi alqadri, "KONTROL DAN MONITORING FITING LAMPU," *PROYEK AKHIR*, september 2019.
- [4] Fauzan Masykur and Fiqiana Prasetyowati, "APLIKASI RUMAH PINTAR (SMART HOME) PENGENDALI PERALATAN ELEKTRONIK RUMAH TANGGA BERBASIS WEB," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 3, pp. 51-58, Maret 2016.
- [5] Muhamad Muslihudin, Willy Renvillia, Taufiq , Andreas Andoyo, and Fery Susanto Susanto, "IMPLEMENTASI APLIKASI RUMAH PINTAR BERBASIS ANDROID DENGAN," *Jurnal Keteknikan dan Sains (JUTEKS) – LPPM UNHAS*, vol. 1, JUNI 2018.
- [6] fauzan masykur and fiqiana prasetyowati, "Jurnal Teknologi Informasi Komputer," *APLIKASI RUMAH PINTAR(SMART HOME) PENGENDALI PERALATAN ELEKTRONIK RUMAH BERBASIS WEB*, vol. 3, p. 52, MARET 2016.
- [7] Dinda Hardianti, Mutia Rizki Rizki, and Fajri Yanti, "PENGUNAAN DHT11 DAN ARDUINO UNO SEBAGAI PENDETEKSI SUHU," *RELATIVITAS: Jurnal Riset Inovasi Pembelajaran Fisika*, vol. 1, April 2019.
- [8] azhar ilyas. (2018, Desember) AKURAT.CO. [Online]. m.akurat.co

- [9] Adrinta Abdurrazaq, Muhammad Ihsan, Anhari Syahputra, Rasyid Imam Ghani, and Dahlan Sitompul, *SENSOR*, Januari 2017.
- [10] abu daud. (2018, juni) abu daud belajar dan berbagi web site. [Online]. abudaud.wordpress.com
- [11] Puthut Kurniawan , Rozeff Pramana, and Deny Nusyrwan, "PROTOTYPE SISTEM DETEKSI KEBOCORAN AIR DAN PENGURASAN SECARA OTOMATIS PADA KAPAL BERBASIS ARDUINO UNO DAN LABVIEW ," 2017.
- [12] ECLIPSEERA MODUL. (2016) ARDUINO-SHOP website. [Online]. ARDUINO-SHOP.EU
- [13] Fatoni Nur Habibi, Sabar Setiawidayat, and Moh Mukhsim, "Alat Monitoring Pemakaian Energi Listrik Berbasis Android," *Prosiding Seminar Nasional Teknologi Elektro Terapan 2017*, vol. 1, 2017.
- [14] admin. (2019, agustus) NN digital website. [Online]. <https://www.nn-digital.com/>
- [15] Abdul Hafiz, Fardian , and Aulia Rahman, "Rancang Bangun Prototipe Pengukuran dan Pemantauan Suhu, Kelembaban serta Cahaya Secara Otomatis Berbasis Iot pada Rumah Jamur Merang," *KITEKTRO: Jurnal Online Teknik Elektro*, vol. 2, 2017.
- [16] a Najmurrokhman, Amrulloh, and Kusnandar, "PROTOTIPE PENGENDALI SUHU DAN KELEMBABAN UNTUK COLD STORAGE MENGGUNAKAN MIKROKONTROLER ATMEGA328 DAN SENSOR DHT11," *jurnal teknologi*, vol. 10, p. 76, september 2017.
- [17] Satukan Halawa, "PERANCANGAN APLIKASI PEMBELAJARAN TOPOLOGI JARINGAN KOMPUTER UNTUK SEKOLAH MENENGAH KEJURUAN (SMK) TEKNIK KOMPUTER DAN JARINGAN (TKJ)

DENGAN METODE COMPUTER BASED INSTRUCTION ," *Jurnal Riset Komputer (JURIKOM)*, vol. 3, p. 68, februari 2016.

[18] Siti Nur Khasanah, "KEAMANAN JARINGAN DENGAN PACKET FILTERING FIREWALL (STUDI KASUS: PT. SUKSES BERKAT MANDIRI JAKARTA) ," *JURNAL KHATULISTIWA INFORMATIKA*, vol. IV, pp. 183-185, desember 2016.

[19] Siti Nur Habibah and Melda Panjaitan, "Pembelajaran Fisika Dasar dan Elektronika Dasar (Arus, Hambatan, dan Tegangan Listrik) Menggunakan Aplikasi Matlab Metode Simulik," vol. 4, april 2018.

[20] Rahmad Azly. (2018, November) READING & LEARNING TEMPAT KITA BERBAGI ILMU. [Online]. duniaberbagiilmuuntuksemua.blogspot.com

[21] Asep Muhamad Alipudin, Didik Notosudjono, and Dimas Bangun Fiddiansyah, "RANCANG BANGUN ALAT MONITORING BIAYA LISTRIK TERPAKAI BERBASIS INTERNET OF THINGS (IOT)," 2018.

[22] (2019, May) bootUP blog. [Online]. bootuo.ai

[23] (2018, agustus) wikimedia commons. [Online]. commons.wikimedia.org

[24] (2020, maret) Dosen Pendidikan. [Online]. dosenpendidikan.co.id

LAMPIRAN

LAMPIRAN 1

Daftar Riwayat Hidup

DAFTAR RIWAYAT HIDUP

I. Data Pribadi

Nama Lengkap : Ayu Anki
Tempat, Tanggal Lahir : Bandung, 8 April 1999
Alamat : jalan kejaksanaan II
Telp : -
HP : 081271412973
e-mail:
ankiayu08@gmail.com
Jenis kelamin : Perempuan
Agama : Islam



II. Riwayat Pendidikan

Pendidikan	Tahun
SD MUHAMMADIYAH MUNTOK	2011
SMPN 1 MUNTOK	2014
SMAN 1 MUNTOK	2017

III. Pendidikan Non Formal

-

Sungailiat, 19 Agustus 2020

Ayu Anki

DAFTAR RIWAYAT HIDUP

I. Data Pribadi

Nama Lengkap : Deswarto
Tempat, Tanggal Lahir : Bangka Tengah, 30
Desember 1998
Alamat : Dusun Semujur Atas
Telp : -
HP : 083175641002
e-mail:
deswartodeswarto@gmail.com
Jenis kelamin : Laki-laki
Agama : Islam



II. Riwayat Pendidikan

Pendidikan	Tahun
SDN 10 PANGKALAN BARU	2011
SMPN 6 PANGKALPINANG	2014
SMKN 2 PANGKALPINANG	2017

III. Pendidikan Non Formal

-

Sungailiat, 19 Agustus 2020



Deswarto

LAMPIRAN 2

PROGRAM APLIKASI

PROGRAM APLIKASI

PROGRAM ON/OFF



PROGRAM MONITORING

```
when Web1 .GotText
  url responseCode responseType responseContent
do
  if get responseCode = 200
  then set DataLampu . Text to get responseContent

when Web2 .GotText
  url responseCode responseType responseContent
do
  if get responseCode = 200
  then set DataStopkontak . Text to get responseContent

when Web3 .GotText
  url responseCode responseType responseContent
do
  if get responseCode = 200
  then set DataTermometer . Text to get responseContent
```

PROGRAM *TIMER*

```
when OKlampu .Click
do
  set Web1 . Uri to join " http://"
  join IP_LAMPU_OK . Text
  /*
  *
  * if JamlampuON . Selection = "0"
  * then "25"
  * else JamlampuON . Selection
  *
  * if MenitlampuON . Selection = "0"
  * then "61"
  * else MenitlampuON . Selection
  *
  * if JamlampuOFF . Selection = "0"
  * then "25"
  * else JamlampuOFF . Selection
  *
  * if MenitlampuOFF . Selection = "0"
  * then "61"
  * else MenitlampuOFF . Selection
  * #
  */
  set Awallampu . Visible to true
  set KetOKlampu . Visible to false
  call Web1 .Get
```

PROGRAM SET IP

```
when HAPUS_IP_LAMPU .Click
do
  call TinyDB1 .ClearAll
  set IP_LAMPU_OK .Text to join " " IP_LAMPU .Text

when HAPUS_IP_STOPKONTAK .Click
do
  call TinyDB1 .ClearAll
  set IP_STOPKONTAK_OK .Text to join " " IP_STOPKONTAK .Text

when HAPUS_IP_POMPA .Click
do
  call TinyDB1 .ClearAll
  set IP_POMPA_OK .Text to join " " IP_POMPA .Text

when HAPUS_IP_GERENDEL .Click
do
  call TinyDB1 .ClearAll
  set IP_GERENDEL_OK .Text to join " " IP_GERENDEL .Text

when HAPUS_IP_TERMOMETER .Click
do
  call TinyDB1 .ClearAll
  set IP_TERMOMETER_OK .Text to join " " IP_TERMOMETER .Text
```

LAMPIRAN 3

PROGRAM NodeMCU

PROGRAM NODEMCU

PROGRAM ON/OFF

```
    if (req.indexOf("/lampumanualon") != -1)
    {
        Serial.println("HIDUP");
        digitalWrite(relay,LOW);
        String kondisi = "HTTP/1.1 200 OK\r\nContent-
Type: text/html\r\n\r\n<!DOCTYPE HTML>\r\n<html>\r\n ";
        kondisi += "1";
        kondisi += "</html>\n";
        client.print(kondisi);
    }
    else if (req.indexOf("/lampumanualoff") != -1)
    {
        Serial.println("MATI");
        digitalWrite(relay,HIGH);
        String kondisi = "HTTP/1.1 200 OK\r\nContent-
Type: text/html\r\n\r\n<!DOCTYPE HTML>\r\n<html>\r\n ";
        kondisi += "0";
        kondisi += "</html>\n";
        client.print(kondisi);
    }
    else if (req.indexOf("/lampuotomatisation") != -1)
```

```
{  
    Mode=1;  
    String kondisi = "HTTP/1.1 200 OK\r\nContent-  
Type: text/html\r\n\r\n<!DOCTYPE HTML>\r\n<html>\r\n "  
    kondisi += "3";  
    kondisi += "</html>\n";  
    client.print(kondisi);  
}
```

PROGRAM *TIMER*

```
    DateTime now = rtc.now();

    if (now.hour() == JamOn && now.minute() == MenitOn
&& now.second() <=5)

    {

        digitalWrite(relay, LOW);

        Serial.println("LAMPU MENYALA (TIMER)");

    }

    else if(now.hour() == JamOff && now.minute() ==
MenitOff && now.second() <=5)

    {

        digitalWrite(relay, HIGH);

        Serial.println("LAMPU MATI (TIMER)");

    }
```

PROGRAM MONITORING

```
if (req.indexOf("/data") != -1)
{
    sensor();

    String data = "HTTP/1.1 200 OK\r\nContent-Type:
text/html\r\n\r\n<!DOCTYPE HTML>\r\n<html>\r\n ";

    data += "Tegangan= ";

    data += (voltage);

    data += " Volt";

    data += "<br/>";

    data += "Arus= ";

    data += (current);

    data += " Ampere";

    data += "<br/>";

    data += "Daya= ";

    data += (power);

    data += " Watt";

    data += "<br/>";

    data += "Waktu Pemakaian Sekarang= ";

    data += (jam);

    data += ":";

    data += (menit);

    data += ":";
```

```
data += (detik);

data += "<br/>";

data += "Biaya Pemakaian Sekarang= ";

data += (totalbiaya);

data += "<br/>";

data += "Waktu Pemakaian Hari Ini= ";

data += (totaljam);

data += ":";

data += (totalmenit);

data += ":";

data += (totaldetik);

data += "<br/>";

data += "Biaya Pemakaian Hari Ini= ";

data += (biayahrini);

data += "<br/>";

data += "Waktu Pemakaian Kemarin= ";

data += (jamkemarin);

data += ":";

data += (menitkemarin);

data += ":";

data += (detikkemarin);

data += "<br/>";

data += "Biaya Pemakaian Kemarin= ";
```

```
data += (biayakemarin);  
data += "<br/>";  
data += "</html>\n";  
client.print(data); delay(1);}
```

PROGRAM OTOMATIS

```
if(state < 500)
{
    digitalWrite (relay, HIGH);
    Serial.println("Motion absent!");
    String kondisi = "HTTP/1.1 200
OK\r\nContent-Type: text/html\r\n\r\n<!DOCTYPE
HTML>\r\n<html>\r\n ";
    kondisi += "2";
    kondisi += "</html>\n";
    client.print(kondisi);
    io=0;
}
else if (state > 500)
{
    digitalWrite (relay, LOW);
    Serial.println("Motion detected!");
    String kondisi = "HTTP/1.1 200
OK\r\nContent-Type: text/html\r\n\r\n<!DOCTYPE
HTML>\r\n<html>\r\n ";
    kondisi += "3";
    kondisi += "</html>\n";
    client.print(kondisi);
    io=1; }
```

PROGRAM SET IP

```
void ModeAccessPoint()
{
    IPAddress AP(192,168,43,1);
    IPAddress mask = (255, 255, 255, 0);
    WiFi.mode(WIFI_AP);
    WiFi.softAP("lampu12", "habi20190");
    WiFi.softAPConfig(AP, AP, mask);
    server.begin();
    Serial.print("IP: "); Serial.println(WiFi.softAPIP());
    Serial.print("MAC:");
    Serial.println(WiFi.softAPmacAddress());
}
```