

**SISTEM KONTROL DAN MONITORING LAMPU TAMAN  
POLMANBABEL MENGGUNAKAN PANEL SURYA BERBASIS IoT**

**PROYEK AKHIR**

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



Disusun oleh :

Monica

NIM: 0031916

Muhammad Haz Fatahillah A

NIM: 0031917

**POLITEKNIK MANUFAKTUR NEGERI  
BANGKA BELITUNG**

**2022**

**LEMBAR PENGESAHAN**

**SISTEM KONTROL DAN MONITORING LAMPU TAMAN  
POLMANBABEL MENGGUNAKAN PANEL SURYA BERBASIS IoT**

Oleh :

Monica

NPM 0031916

Muhammad Haz Fatahillah A

NPM 0031917

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan  
Program Diploma III Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



Zanu Saputra, M.Tr.T

Pembimbing 2



Surojo, M.T

Penguji 1

Muhammad Iqbal Nugraha, M.Eng

Penguji 2



Dr.Parulian Silalahi, M.Pd

## **PERNYATAAN BUKAN PLAGIAT**

Yang bertanda tangan di bawah ini :

Nama Mahasiswa 1 : Monica NIM : 0031916

Nama Mahasiswa 2 : Muhammad Haz Fatahillah A NIM : 0031917

Dengan judul : SISTEM KONTROL DAN MONITORING LAMPU  
TAMAN POLMANBABEL MENGGUNAKAN PANEL  
SURYA BERBASIS IoT.

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, 8 Agustus 2022

Nama Mahasiswa

Tanda Tangan

1. Monica

.....

2. Muhammad Haz Fatahillah A

.....

## **ABSTRAK**

*Pengontrolan lampu pada fasilitas penerangan taman pada umumnya menggunakan listrik dan saklar, sehingga menyulitkan operator dalam menghidupkan lampu taman. Hal tersebut menjadi dasar pembuatan pengontrolan lampu penerangan tenaga surya menggunakan input tenaga matahari yang berupa panel surya dan Arduino uno sebagai pusat pemrosesan data yang didapat dari sensor arus, sensor tegangan, dan sensor LDR yang digunakan sebagai pembaca keadaan sekitar lampu. Lampu taman menggunakan Arduino uno sebagai kontroler utamanya, NodeMCU yang berfungsi sebagai media komunikasi antara Android dengan lampu taman berbasis panel surya dengan memanfaatkan jaringan Wi-Fi serta menerapkan teknologi IoT. dan monitoring dilakukan melalui aplikasi Android yang dibuat menggunakan software Kodular yang bernama Simpuman, Dari pengujian yang dilakukan dapat disimpulkan bahwa lampu dapat menyala ketika hardware terhubung dengan hotspot portable kecerahan lampu maksimal daya yang dihasilkan 498.2 mWatt.*

*Kata kunci: Internet of Things (IoT), Kodular, kontrol lampu taman, tenaga surya.*

## **ABSTRACT**

*Light control in garden lighting facilities generally uses electricity and switches, making it difficult for operators to turn on garden lights. This is the basis for controlling solar lighting using solar input in the form of solar panels and Arduino Uno as a data processing center obtained from current sensors, voltage sensors, and LDR sensors which are used as readers of the state around the lamp. Garden lights use Arduino uno as the main controller, NodeMCU which functions as a communication medium between Android and solar panel-based garden lights by utilizing the Wi-Fi network and applying IoT technology. and monitoring is carried out through an Android application made using Kodular software called Simpuman. From the tests carried out, it can be concluded that the lights can turn on when the hardware is connected to a portable hotspot, the maximum brightness of the lamp is 498.2 mWatt.*

*Keywords: Internet of Things (IoT), Kodular, garden light control, solar power.*

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat ALLAH Subhanahu Wa Ta'ala atas berkat rahmat dan hidayah-Nya jualah sehingga pada akhirnya penulis dapat menyelesaikan karya tulis proyek akhir ini dengan baik. Karya Tulis Proyek Akhir ini disusun sebagai salah satu persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program pendidikan Diploma III di Politeknik Manufaktur Negeri Bangka Belitung. Penulis mencoba untuk menerapkan ilmu pengetahuan yang telah didapatkan selama tiga tahun mengenyam pendidikan di Politeknik Manufaktur Negeri Bangka Belitung dalam melaksanakan Proyek Akhir ini.

Pada kesempatan ini penulis menyampaikan rasa terimakasih yang sebesar-besarnya kepada orang-orang yang telah berperan sehingga dapat terselesaikannya Proyek Akhir ini, sebagai berikut :

1. Keluarga besar (Ayah, Ibu, Kakak – Kakak Penulis teman-teman dll) yang selalu senantiasa memberikan kasih sayang, doa, dukungan moril maupun materi dan semangat.
2. Bapak I Made Andik Setiawan, M.Eng., Ph.D selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Zanu Saputra, M.Tr.T. selaku pembimbing I dan Bapak Surojo, M.T selaku pembimbing II yang telah meluangkan banyak waktu, tenaga, pikiran dan yang paling penting Motivasi didalam memberikan pengarahan dalam penulisan karya tulis Proyek Akhir ini.
4. Seluruh staf pengajar di Politeknik Manufaktur Negeri Bangka Belitung.
5. Rekan-rekan mahasiswa Politeknik Manufaktur Negeri Bangka Belitung yang telah banyak membantu selama menyelesaikan Proyek Akhir.
6. Rekan-rekan sepermainan di tempat penulis tinggal, yang selalu memberikan semangat mental dan moril disaat penulis jenuh.
7. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung dalam pembuatan Proyek Akhir ini yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa penulisan makalah proyek akhir ini masih jauh dari sempurna dikarenakan penulis adalah manusia biasa yang tidak luput dari kesalahan. Karena yang benar hanya datang dari ALLAH dan yang salah datang dari penulis sendiri. Oleh karena itu, sangat diharapkan segala petunjuk, kritik, dan saran yang membangun dari pembaca agar dapat menunjang pengembangan dan perbaikan penulisan selanjutnya. Akhir kata penulis mohon maaf atas kekurangan dalam penulisan karya tulis Proyek Akhir ini dan penulis dengan senang hati menerima saran dan kritik yang membangun dari pembaca. Besar harapan penulis semoga makalah tugas akhir dan alat yang dibuat dapat memberikan manfaat bagi pihak yang berkepentingan pada khususnya dan bagi perkembangan ilmu teknologi pada umumnya.

Sungailiat, 2022

Penulis

## DAFTAR ISI

<b>LEMBAR PENGESAHAN</b> .....	<b>Error! Bookmark not defined.</b>
<b>PERNYATAAN BUKAN PLAGIAT</b> .....	ii
<b>ABSTRAK</b> .....	iii
<b>ABSTRACT</b> .....	iv
<b>KATA PENGANTAR</b> .....	iv
<b>DAFTAR ISI</b> .....	vii
<b>DAFTAR GAMBAR</b> .....	ix
<b>DAFTAR TABEL</b> .....	x
<b>DAFTAR LAMPIRAN</b> .....	xi
<b>BAB I PENDAHULUAN</b> .....	1
1.1 Latar Belakang Masalah .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Proyek Akhir .....	2
<b>BAB II DASAR TEORI</b> .....	3
2.1 Deskripsi Umum Lampu Taman .....	3
2.2 Panel Surya ( <i>Solar Cell</i> ).....	3
2.3 Baterai ( <i>Accumulator</i> ) .....	5
2.4 <i>Solar Charge Controller</i> .....	6
2.5 Sensor <i>Light Dependent Resistor</i> (LDR).....	7
2.6 <i>Maximum Power Point Tracking</i> (MPPT) .....	8
2.7 Arduino Uno .....	9
2.8 NodeMCU ESP8266 .....	11
2.9 Kodular .....	12
2.9.1 Kelebihan Aplikasi Kodular .....	13
2.9.2 Kekurangan Aplikasi Kodular .....	14
2.9.3 Logo Kodular .....	14
<b>BAB III METODE PELAKSANAAN</b> .....	15
3.1 Metode Pelaksanaan Proyek Akhir .....	15



3.2 Pengumpulan dan Analisis Data.....	16
3.3 Perancangan Konstruksi Lampu Taman.....	16
3.4 Perancangan <i>Hardware</i> Sistem kontrol dan <i>monitoring</i> Lampu Taman.....	17
3.5 Perancangan <i>Software</i> pembuatan aplikasi .....	17
3.6 Rancangan Konstruksi dan <i>Hardware</i> Elektrik Kontrol <i>Monitoring</i> .....	19
3.7 Pembuatan Makalah Proyek Akhir.....	20
<b>BAB IV PEMBAHASAN</b> .....	21
4.1 Deskripsi Alat.....	21
4.1.1 <i>Hardware</i> Elektrik Lampu Taman Polmanbabel Menggunakan Panel Surya Berbasis Iot.....	22
4.2 Blok diagram dan <i>Flowchart</i> .....	23
4.2.1 Blok Diagram <i>Hardware</i> Komponen .....	23
4.2.2 Blok Diagram Alir .....	23
4.2.3 <i>Flowchart</i> .....	24
4.4 Pemilihan NodeMCU ESP8266 .....	26
4.6 <i>Software Monitoring</i> Lampu Berbasis <i>Android</i> .....	27
4.6.1 Aplikasi “Simpuman”.....	27
4.6.2 Pembuatan aplikasi “Simpuman” .....	29
4.6.3 Pengujian Aplikasi “Simpuman” .....	30
4.7 Pengujian Sensor LDR .....	31
4.8 Pengujian <i>Monitoring</i> Tegangan, Arus, dan Daya pada Alat.....	35
4.8.1 Tabel Pengisian Daya Pada Panel Surya: .....	35
4.8.2 Tabel Penggunaan Daya Pada Panel Surya: .....	37
4.8.3 Pengujian sistem penerangan taman dalam 10 Hari .....	38
<b>BAB V PENUTUP</b> .....	42
5.1 Kesimpulan.....	42
5.2 Saran.....	42
<b>DAFTAR PUSTAKA</b> .....	43

## DAFTAR GAMBAR

Gambar 2. 1 Panel Surya ( <i>Solar cell</i> ).....	4
Gambar 2. 2 Sensor LDR.....	7
Gambar 2. 3 <i>Maximum Power Point Tracking</i> (MPPT) .....	9
Gambar 2. 4 Arduino UNO.....	10
Gambar 2. 5 NodeMCU ESP 8266 .....	11
Gambar 2. 6 Logo Kodular .....	14
Gambar 3. 1 Rancangan Kontruksi Lampu.....	16
Gambar 3. 2 <i>Hardware</i> Sistem kontroler.....	17
Gambar 3. 3 <i>Layout</i> utama <i>software</i> Kodular .....	18
Gambar 3. 4 <i>Layout</i> pada bagian <i>blocks software</i> Kodular .....	18
Gambar 3. 5 <i>Layout</i> Simpunan di <i>software</i> Kodular .....	19
Gambar 3. 6 Konstruksi Lampu Penerangan .....	19
Gambar 4. 1 Konstruksi Lampu Penerangan Taman .....	22
Gambar 4. 2 <i>Hardware</i> Elektrik Lampu Taman .....	22
Gambar 4. 3 Blok Diagram Komponen .....	23
Gambar 4. 4 Rancangan Aplikasi Simpunan .....	27
Gambar 4. 5 Tampilan <i>Designer</i> .....	28
Gambar 4. 6 Tampilan Hasil Di Aplikasi.....	28
Gambar 4. 7 Tampilan Blocks Program.....	28
Gambar 4. 8 Tampilan Menu Aplikasi Simpunan .....	29
Gambar 4. 9 Skema Pengujian LDR.....	31
Gambar 4. 10 Kondisi Pada Malam Hari .....	34
Gambar 4. 11 Kondisi Pada Siang Hari .....	34
Gambar 4. 12 Kondisi Pada Tengan Malam .....	34
Gambar 4. 13 Kondisi Pada Sore Hari .....	34

## DAFTAR TABEL

Tabel 2. 1 Spesifikasi Arduino Uno .....	9
Tabel 2. 2 Spesifikasi NodeMCU ESP8266 .....	10
Tabel 4. 1 Pengujian NodeMCU ESP 8266 .....	26
Tabel 4. 2 Pengujian Sensor LDR.....	32
Tabel 4. 3 Hasil Pengujian LDR dibanding dengan alat ukur Luxmeter .....	35
Tabel 4. 4 Tabel Pengisian Daya Panel Surya .....	35
Tabel 4. 5 Tabel Penggunaan Daya Pada Panel Surya.....	37
Tabel 4. 6 Tabel Pengujian Sistem Penerangan.....	38
Tabel 4. 7 Grafik Hasil Pengujian.....	38
Tabel 4. 8 Tabel Pengujian Lampu Sistem dengan Lampu Konvensional .....	38

## DAFTAR LAMPIRAN

- Lampiran 1 Daftar Riwayat Hidup
- Lampiran 2 Kode Program Arduino Uno
- Lampiran 3 Kode Program NodeMCU ESP8266
- Lampiran 4 Kode Program Aplikasi Simpunan



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Energi listrik merupakan kebutuhan utama masyarakat dalam melakukan segala aktifitas sehari – hari. Energi matahari merupakan sumber energi alternatif potensial dan tidak pernah habis energinya [1]. Dalam pemakaian energi pada masa kini masih banyak pengguna peralatan terutama peralatan elektronik yang mengkonsumsi energi listrik secara berlebihan.

Lampu taman adalah salah satu fasilitas penerangan yang ada di setiap taman. Ketika lampu taman padam masih banyak yang ditangani dengan cara manual [2], banyak pengguna lampu taman membiarkan lampu menyala terus menerus sepanjang malam bahkan sepanjang hari dengan tingkat kecerahan maksimal, hal tersebut merupakan pemborosan dalam penggunaan energi listrik [3]. Pemantauan lampu dilakukan dengan mendatangi tempat dimana lampu berada, jika kontrol lampu seperti ini maka sangat banyak kekurangan salah satunya petugas membutuhkan waktu untuk mengetahui kondisi lampu penerangan jalan. Oleh karena itu diperlukan penyesuaian tingkat kecerahan lampu pada jam – jam tertentu, dan diperlukan adanya sistem kontrol dan *monitoring* lampu penerangan taman yang dapat dikendali oleh sistem berbasis *IoT* dan *Android* untuk memudahkan petugas yang mengontrol lampu penerangan taman dari jarak jauh.

Berdasarkan uraian diatas, penulis membuat suatu alat dengan memanfaatkan energi dari matahari sebagai sumber energi listrik. Proyek akhir ini bertujuan untuk memudahkan operator agar dapat mengetahui dengan mudah jika ada kerusakan pada lampu penerangan jalan dari jarak jauh serta *monitoring* tingkat efisiensi kecerahan lampu penerangan jalan umum serta pemeliharaan lampu jalan yang dapat dikontrol kapan waktu menyala pada lampu taman kampus Politeknik Manufaktur Negeri Bangka Belitung yang berbasis *IoT* menggunakan Aplikasi yang dibuat menggunakan *software* Kodular yang penulis beri nama “Simpunan”.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas maka bisa dirumuskan beberapa masalah berikut ini:

1. Bagaimana perancangan *hardware* sistem kontrol dan *monitoring* lampu taman Polmanbabel menggunakan panel surya berbasis *IoT*.
2. Bagaimana cara membuat aplikasi dan penggabungan data pada alat agar sesuai dengan aplikasi.
3. Bagaimana efisiensi perancangan sistem kontrol lampu berbasis *IoT* terhadap lampu konvensional.

## 1.3 Batasan Masalah

Dalam proyek akhir ini yang menjadi Batasan masalah adalah :

1. *Solar cell* yang digunakan merupakan *solar cell* 50WP dengan Baterai aki yang digunakan 12V12Ah.
2. Objek yang di kontrol adalah satu lampu LED DC dengan total 10watt.
3. Jaringan yang digunakan pada proyek ini adalah *Wi-Fi* Polman dan *hotspot portable* dari *smartphone* yang sesuai dengan kode pada program Arduino.

## 1.4 Tujuan Proyek Akhir

Adapun Tujuan yang akan disampaikan melalui proyek akhir ini yaitu:

1. Membuat sistem kontrol berbasis *IoT* pada Aplikasi lampu taman menggunakan *software* Kodular yang dibuat pada aplikasi *Android* yang diberi nama “Simpuman”.
2. Membuat sistem *monitoring* lampu taman untuk mengetahui tegangan, arus, serta daya yang dipakai pada lampu tersebut melalui aplikasi “Simpuman”.
3. Mengetahui tingkat kecerahan lampu taman redup, terang, dan sangat terang.

## **BAB II**

### **DASAR TEORI**

#### **2.1 Deskripsi Umum Lampu Taman**

Lampu penerangan taman tenaga surya merupakan sebuah lampu penerangan yang memerlukan cahaya matahari sebagai sumber energi listriknya. Penerangan jalan tenaga surya adalah bentuk alternatif yang murah dan ekonomis untuk digunakan sebagai sumber listrik penerangan karena menggunakan sumber energi yang gratis dan tak ada habisnya dari alam [4]. Lampu jalan tenaga surya menggunakan modul / *solar cell* dengan waktu pakai sampai 25 tahun yang berfungsi sebagai penerima cahaya (sinar) matahari yang kemudian diubah menjadi listrik. Lampu ini secara otomatis dapat mulai menyala pada waktu sore hari dan padam pada pagi hari dengan perawatan yang mudah dan efisien.

#### **2.2 Panel Surya (*Solar Cell*)**

Menurut (Reza Nandika, Pamor Gunoto, 2018). Modul panel surya dengan kapasitas daya 50Wp tersusun dari 32 sel yang terhubung seri, serta masing – masing sel mempunyai daya keluaran listrik 1.6watt arus dan tegangan 3A 0.53V serta tegangan nominal 16.9V dengan proses perancangan alat panel surya yang digunakan 50Wp disesuaikan dengan beban lampu yang mereka gunakan. dengan ukuran kapasitas panel sel surya beban yang digunakan perhari 36watt dan intensitas matahari rata – rata 8jam maka,  $36\text{watt} \times 12/8 = 54\text{Wp}$  (panel surya yang digunakan 50Wp) .

Menurut (Donny T B Sihombing, 2013). Fotovoltaik merupakan teknologi yang berfungsi sebagai pengubah konversi radiasi matahari menjadi listrik secara langsung, dan panel surya merupakan elemen semikonduktoryang dapat mengkonversi energi surya menjadi listrik dengan efek fotovoltaik. Keunggulan dalam menggunakan panel surya ada beberapa diantaranya: terang, tahan lama, hemat energi listrik, ramah lingkungan, dan hemat dalam perawatan. Pada lampu penerangan yang dibuat memiliki komponen pendukung yaitu modul *solarcell*

*polycrystalline* yang berfungsi mengubah cahaya matahari menjadi energi listrik DC dengan satuan WP. Dalam teknik perencanaan sistem penerangan yang dibuat dapat dianalisa dari instalasi *solar cell* yang terbuat dari silikon yang dapat mengubah intensitas matahari menjadi listrik dan membuat photon bergerak menuju electron dan menghasilkan tegangan dan arus listrik DC, jika panel dengan kapasitas 50Wp mendapatkan cahaya matahari 8 jam maka daya yang dihasilkan yaitu 400 Wh dan dapat menyalakan lampu 25Watt selama 4 jam. Daya yang dihasilkan pada saat mendung yaitu 150WP sedangkan pada saat cerah 250WP.

Menurut (Made Tirta Yasa dan Ivany Sarief, 2021). Panel surya tersusun dari beberapa sel yang dihubungkan secara seri dan paralel dan umumnya terdiri dari 32-40 sel dan tergantung ukuran panel dan gabungan membentuk “Array”, kapasitas panel diperlukan beberapa faktor yang diperhatikan kebutuhan energi total, insolasi cahaya dan faktor penyesuaian sehingga dapat menghitung *losses*.

*Solar cell* atau panel surya merupakan alat yang dapat mengkonversi cahaya matahari menjadi listrik serta dapat menghasilkan listrik arus searah (DC), sel surya terdiri dari beberapa lapisan yaitu elektroda depan belakang, lapisan absorber tipe-p dan lapisan transparan tipe-n. *solar cell* memiliki beberapa jenis yaitu *poly-crystalline*, *mono-crystalline* [5]. Sedangkan yang digunakan dalam pembuatan proyek akhir ini yaitu *solar cell* jenis *poly-crystalline* (polikristal) 50wp yang dapat menghasilkan daya listrik pada saat cuaca mendung.



Panel surya yang penulis gunakan dalam pembuatan proyek akhir ini yaitu



menggunakan panel surya *polycrystalline* dimana modul *solar cell* yang memiliki tingkat efisiensi terbaik diantara jenis lainnya. Dengan jumlah sel surya yang lebih banyak maka *output* daya yang dihasilkan akan lebih stabil dan daya puncak yang dihasilkan berkisar rentang 3%, serta panel surya jenis ini berbahan dasar silikon (dengan multi-kristal). Namun dikarenakan banyaknya kristal pada per sel panel surya *polycrystalline* ini menyebabkan tingkat efisiensi yang sedikit rendah [9].

### 2.3 Baterai (*Accumulator*)

Menurut (Wahyu Anhar, Basri, M. Amin, Dkk, 2018). Baterai merupakan komponen yang berfungsi untuk menghasilkan energi listrik arus DC pada *solar system* energi listrik dalam baterai dihasilkan oleh panel surya, dalam perhitungan kapasitas baterai juga harus mempertimbangkan *deep of discharge (DOD)* yang bernilai 80%.

Menurut (Made Tirta Yasa dan Ivany Sarief, 2021). Baterai adalah komponen yang berfungsi menghasilkan arus DC dalam *solar system*. Ketentuan yang membatasi tingkat kedalaman pengosongan maksimum, *DOD* dalam persentase 80% energi yang dapat digunakan 20% sebagai cadangan.

Baterai adalah media yang dapat menyimpan energi yang dapat dikonversi menjadi daya, didalam baterai terjadi sebuah proses elektrokimia yang *reversible* dengan tingkat efisiensi yang tinggi [4]. *Reversible* pada baterai terjadi proses pengisian dari listrik menjadi kimia serta sebaliknya proses pengosongan yaitu perubahan energi kimia menjadi listrik, kapasitas baterai yaitu arus listrik dialirkan ke rangkaian beban dalam jangka waktu [5].

Tegangan kapasitas baterai (Ah) dapat dinyatakan sebagai berikut:

$$C = I \times T$$

Dimana

C = Kapasitas Baterai(Ah)

I = besar arus yang mengalir (A)

T = Waktu (jam)

Sedangkan untuk menghitung kapasitas baterai Ah (*ampere-hours*) dengan persamaan berikut:

$$P = V \times I \quad (1)$$

$$V = P / I \quad (2)$$

$$I = P / V \quad (3)$$

#### **2.4 Solar Charge Controller**

Menurut (Made Tirta Yasa dan Ivany Sarief,2021). SCC dapat mengatur *overcharging* (kelebihan pengisian) dan kelebihan tegangan dari *solar module*. Dalam menghitung kebutuhan kapasitas SCC, maka yang perlu dilihat yaitu nilai *Isc* (*short circuit current*) dan nilainya dikali dengan jumlah panel surya dan dihasilkan minimal nilai dari charge controller dalam penggunaan menggunakan SCC 2A.

Menurut (Reza Nandika, Pamor Gutomo, 2018). Baterai tersisa 20% - 30%, maka regulator akan memutuskan beban dan mengatur kelebihan pengisian agar tidak terjadi *full discharge* dalam penelitian yang dilakukan menggunakan sel surya 12v, maka SCC yang digunakan 12v dan modul sel surya yang digunakan + 10% dari nilai *Isc*. Panel surya yang digunakan 50Wp, maka nilai *Isc* 3.25A dan SCC yang digunakan 3.6A.

*Solar Charge Controller* merupakan alat yang berfungsi mengisi baterai dengan aliran arus konstan ke tegangan yang dicapai, jika arus sudah tercapai maka arus akan turun ke posisi penahan sehingga indikator menunjukkan baterai telah terisi penuh [5].

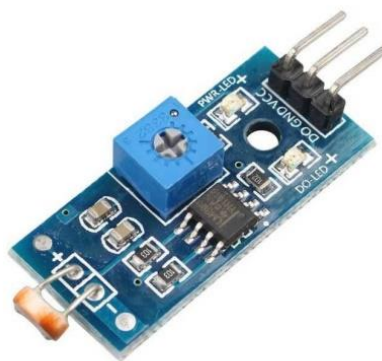
Fungsi dari *solar charge controller* yaitu:

- Dapat mengubah arus DC tegangan tinggi ke tegangan rendah.
- Dapat memaksimalkan transfer daya dari modul fotovoltaik ke baterai dengan MPPT
- Arus balik dari baterai dapat diblokade pada saat malam hari.
- Dapat *monitoring* tegangan, arus, dan daya dari Panel Surya ke baterai.

## 2.5 Sensor Light Dependent Resistor (LDR)

Menurut (Eko Ihsanto, Muhamad Dawud, 2016). Pada sistem yang dibuat setiap lampu dipasangkan dua buah sensor LDR yang berfungsi untuk mendeteksi lampu menyala atau tidak. Dan terhubung ke Arduino Uno R3 yang memiliki fungsi sebagai pengatur jalannya sistem pada sensor (mikrokontroler), satu LDR diletakkan dibawah cahaya lampu dan satu LDR lainnya dipasang diatas tiang PJU dan cara yang digunakan yaitu sensor LDR akan menerima intensitas cahaya jika tidak mencapai batas minimum maka sensor akan menyebut lampu tidak bisa berfungsi.

Sensor *Light Dependent Resistor (LDR)* merupakan sensor yang komponennya berbentuk resistor yang nilai resistansinya berubah-ubah sesuai dengan intensitas cahaya yang mengenai sensor tersebut, Sensor LDR mengeluarkan tegangan analog ketika dihubungkan ke sumber daya 5 volt, semakin besar intensitas cahaya maka semakin besar tegangan yang dikeluarkan oleh sensor LDR. Sensor LDR akan merubah nilai tegangan 0 volt sampe 5 volt menjadi nilai digital dalam kisaran 0 sampai 1023. Kit bernama *flying fish* memadukan sensor LDR dengan IC LM 393, dilengkapi dengan 4 pin yaitu pin *vcc*, *analog out*, *digital out* dan *ground* serta dilengkapi *trigger level* yang berfungsi mengatur keluaran data digital dari sensor LDR [7].



Untuk menghitung tegangan keluaran pada sensor LDR maka menggunakan persamaan berikut ini :

$$V_o = \frac{LDR}{LDR+R_2} V_{CC}$$

Dimana:

$V_o$  = Tegangan *output*

LDR = Resistansi nilai LDR

$R_1$  = Resistor

$V_{CC}$  = Tegangan *input*

## 2.6 Maximum Power Point Tracking (MPPT)

Menurut (Rahmat Ihsani Yuskar, Angga Rusdinar, Irwan Purnama,2018). *Maximum Power Point Tracking (MPPT)* merupakan tahap kontrol penelusur titik maksimum yang dikeluarkan fotovoltaik pada metode kontrol pengisian baterai bekerja secara optimal dan MPPT dapat mengkalkulasi keluaran fotovoltaik dan fibandingkan dengan tegangan pada baterai,serta daya dioptimalkan dalam pengisian baterai.

*Maximum Power Point Tracking (MPPT)* adalah salah satu sistem *solar charge controller* yang terdapat pada sistem panel surya agar dapat menghasilkan daya maksimal. MPPT tidak hanya berfungsi untuk meningkatkan daya dari input panel surya yang dialirkan menuju beban, selain itu juga meningkatkan *lifetime* sistem fotovoltaik [5]. *Controller* MPPT dapat meningkatkan jumlah nilai arus yang masuk ke baterai, nilainya dapat bervariasi jumlahnya tergantung kepada kondisi cuaca hari tersebut, suhu yang terukur, status pengisian daya baterai, dan faktor-faktor lainnya. Fotovoltaik memiliki karakteristik kurva tegangan dan arus yang tidak linier, serta daya maksimum titik puncak koordinat  $V_{mpp}$  dan  $I_{mpp}$ . Tegangan dan arus yang dihasilkan akan diolah datanya melalui *controller* kemudian dihasilkan sinyal PWM dengan nilai *duty cycle* yang sesuai dengan algoritma yang digunakan [10].



Gambar 2. 3 Maximum Power Point Tracking (MPPT)

## 2.7 Arduino Uno

Menurut (Bahrin,2017) dalam pembuatan sistem kontrol otomatis mengatur penggunaan lapu menggunakan arduino uno sebagai pengontrolan proses sistem kerja yang dapat dikendalikan dalam proyek yang digunakannya menggunakan arduino ATmega328 sebagai mikrokontroler utama.

Arduino uno merupakan mikrokontroler yang menggunakan ATmega328 sebagai otak dari sistem kendali, Arduino uno mempunyai 14 pin diantaranya 6 pin *digital output* PWM, 6 pin *digital input analog*, 1 osilator kristal 16MHz, koneksi USB, *power*, IC header, dan *reset button* [6].

Pada *Timer* merupakan bagian microcontroller *clock* internal untuk mengukur waktu. Di *firmware* Arduino timer memiliki konfigurasi frekuensi 1kHz *enable interrupt*. Fungsi *timer* dalam Arduino dengan *output* PWM dalam data *sheet* Arduino terdapat PWM pin dengan contoh OCRxA dengan x merupakan *timer*. Pada board Arduino Uno memiliki 3 *timer* serta 6 pin *output* PWM.

Pada pin Arduino dengan PWM adalah sebagai berikut:

- Arduino pin 5 dan 6, *timer* 0 (8 bit)
- Arduino pin 9 dan 10, *timer* 1 (16 bit)
- Arduino pin 3 dan 11, *timer* 2 (8 bit)

Mode Operasi *timer* ada beberapa yaitu:

- Normal, untuk *delay* dan selang waktu.

- PWM, untuk gelombang PWM dengan resolusi tinggi.
- *Fast PWM*, memberi pulsa PWM frekuensi tinggi.
- CTC, mengembalikan nilai TCNT ke 0

Konfigurasi pin pada Arduino Uno:

- IC ATMEGA328 : mikrokontroler dengan *boothloader* Arduino uno.
- *Digital I/O* : pada pin 3,5,6,9,10,11 dengan tanda (~) berarti juga berfungsi untuk PWM.
- *Analog Input* : *input* data sensor, potensiometer.
- *Power* : 5V, 3.3V, GND.
- *Power Jack* : Tegangan *input*.



Tabel 2. 1 Spesifikasi Arduino Uno

Deskripsi	Spesifikasi
Mikrokontroler	ATMega328
Tegangan	5V
Tegangan <i>Input</i> (dianjurkan)	7 – 12 V
Tegangan <i>Input</i>	6 – 20 V (batas)
Digital I/O	14 Pin (6 dijadikan <i>output</i> PWM)
Analog <i>Input</i>	6 Pin
Arus DC untuk 3.3V	50mA

Arus DC per I/O	40mA
Flash Memory 32kB (ATMega328)	0.5 kB digunakan untuk <i>bootloader</i>
EEPROM	1kB
SRAM	2kB (ATMega328)
Clock speed	16 Hz

## 2.8 NodeMCU ESP8266

NodeMCU yaitu merupakan *platform Internet of Thing (IoT)* yang memiliki sifat *open source*. NodeMCU terdiri dari perangkat keras berupa *system on chip Wi-Fi ESP8266* dari ESP8266 buatan *ESPRESSIF SYSTEM*. NodeMCU ESP8266 menggunakan ESP8266 dengan seri ESP-12E (ESP8266MOD) dengan dilengkapi *driver CH340G*. Pengembangan penggunaan NodeMCU ESP8266 dapat menggunakan *sketch* atau program dengan *software Arduino IDE*[4]. Berikut NodeMCU ESP8266 ditunjukkan dengan gambar.



Spesifikasi dari NodeMCU ESP8266 dapat dilihat pada tabel 2.1 berikut.

Tabel 2. 2 Spesifikasi NodeMCU ESP8266

Deskripsi	Spesifikasi
Mikrokontroler	ESP8266
<i>Input</i> Tegangan	3.3V ~ 5V
Ukuran <i>board</i>	57mm X 30mm
GPIO	13 pin
<i>Flash Memory</i>	4MB
<i>Wireless</i>	802.11 b\g\n <i>standard</i>
<i>USB to serial converter</i>	CH340G

## 2.9 Kodular

Menurut(Ronaldo,2020). Pengembangan aplikasi menggunakan Kodular dan pembuatan aplikasi “*Wonderful of Minangkabau*” alasannya dalam memilih *software* Kodular dalam pembuatan Aplikasi tersebut dikarenakan project *opensource* MIT App Inventordengan banyak fitur hingga *tools* yang dapat digunakan sangat bisa dimengerti mulai dari mengakses *website* Kodular serta mendesain aplikasi dan membuat aplikasi yang diinginkan.

Menurut(Tristi Ardita Rismayani, Nurul Anriani, Sukirwan, 2022) dalam penelitian yang dibuat oleh mereka pengembangan e-modul dalam media pembelajaran siswa belajar secara mandiri dapat dengan Kodular yang berbasis *smartphone* Android dengan model pengembangan ADDIE (*Analysis,Design,Development,Implementation,Evaluation*) adalah valid berdasar pada pengujian dengan persentase rata – rata 83,3%.

Menurut(Heni Ambarwati,2021). Dalam peneltiain yang dilakukanbertujuan untuk mengetahui langkah dan pembelajaran Android Kodular pada materi zakat fikih, adapun dalam pengembangan menggunakan model yang dilakukan membuat berbagai macam aplikasi kepada pengajar agar belajar pada peseta didik dapat lebih efektif salah satu caranya yaitu dalam pembuatan aplikasi menggunakan *software* Kodular dimana banyak kemudahan dalam proses



membuatnya serta dapat dipelajari dengan mudah oleh pengguna dikarenakan *software* Kodular tanpa Coding.

Kodular merupakan sebuah situs berupa *web* yang didalamnya tersedia beberapa *tools* yang diantaranya yaitu, *Mit App Inventor* yang menyediakan sebuah *block programming* untuk membuat sebuah aplikasi tanpa perlu menguasai ataupun menetik kodingan untuk membuat kode program. Kodular sebelumnya memiliki nama *Makeorid*. Pada situs Kodular terdapat *tools* IDE yang bersifat *open source* seperti *App Inventor*. Kodular juga memiliki fitur *widget* banyak dari *tools* IDE. Selain itu, Kodular tidak hanya dapat membuat aplikasi Android, tetapi juga dapat mengunggah hasil ke dalam Kodular *store* dan dapat membuat *widget* yang tidak terdapat dari bawaan dengan ekstensi.

Kodular terus berkembang pada saat proses pembuatan aplikasi *tools* untuk memudahkan *developer* dan aplikasi Android tanpa *coding*. Untuk pembuatan aplikasi Android pada Kodular, hanya mengandalkan *drag and drop* serta menyusun *puzzle blok program* agar program pada aplikasi yang dibuat bisa berfungsi dengan baik.

### 2.9.1 Kelebihan Aplikasi Kodular

- Tampilan dashboard *Design UI/UX*.
- Banyak fitur *widget*.
- Tidak harus *install software* tambahan.
- Menggunakan *web browser*.
- Hanya mengisi parameter dari *program blocks*.
- Tanpa *coding*.
- Aplikasi Android lebih efisien.
- *drag dan drop* pada program blocks yang ada.
- Bisa membuat *database* lokal dengan SQLite atau TinyDB.
- Aplikasi bisa di *publish* atau *extension* ke Kodular *store*.

### 2.9.2 Kekurangan Aplikasi Kodular

- Batasan maksimum ukuran aplikasi Android 10 MB. Jika lebih, terjadi *error* ketika kompilasi.
- Aplikasi Android Tidak bisa dirancang 100% sesuai keinginan pengguna.
- Semakin banyak *project*, maka akan lama *loading*.
- Tidak bisa membuat aplikasi Android *launcher*, *widget*, dan *theme* (tema).
- Harus *online*.

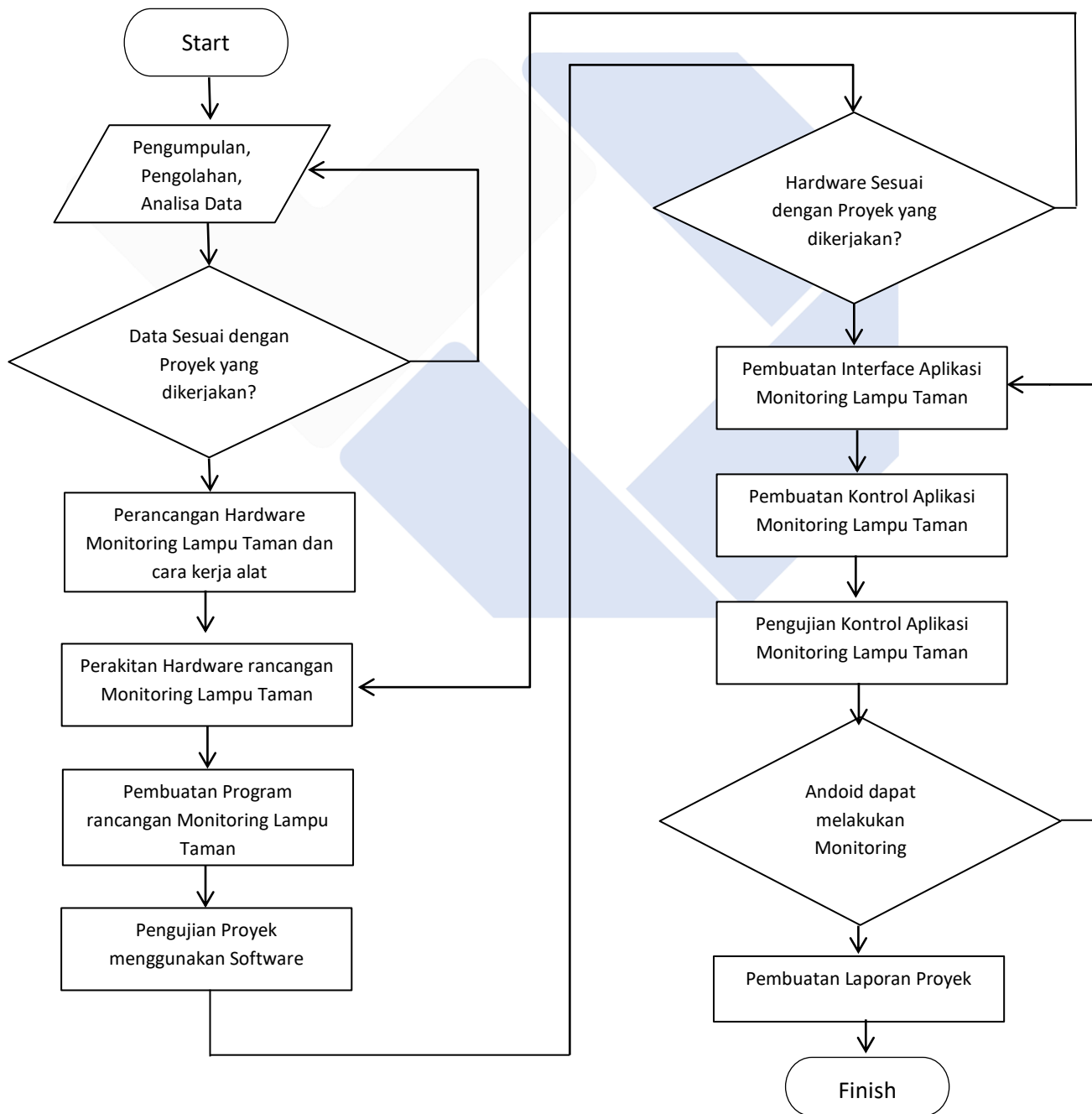
### 2.9.3 Logo Kodular



### BAB III METODE PELAKSANAAN

#### 3.1 Metode Pelaksanaan Proyek Akhir

Dalam pelaksanaan proyek akhir ini, dilakukan tahapan-tahapan penelitian yang bertujuan untuk memudahkan penulis dalam proses pembuatan proyek akhir. Adapun tahap-tahap penelitian yang dilakukan penulis adalah seperti yang terlihat pada *flowchart* berikut:



### 3.2 Pengumpulan dan Analisis Data

Pengumpulan data merupakan proses mencari dan mengumpulkan data-data yang diperlukan dalam proses pembuatan proyek akhir. Selanjutnya, data dianalisis untuk menentukan apa saja yang akan diterapkan pada proyek akhir sehingga proyek akhir ini memiliki kelebihan tersendiri dari kontrol *monitoring* lampu penerangan jalan yang pernah dibuat.

Pada tahap pengumpulan data penulis melakukan beberapa cara diantaranya:

1. Observasi : pengamatan Lampu Taman Polmanbabel konvensional yang berada di Politeknik Manufaktur Negeri Bangka Belitung.
2. Studi Pustaka : pengumpulan data dari buku referensi, jurnal, artikel internet serta prosiding sesuai judul yang diangkat.

### 3.3 Perancangan Konstruksi Lampu Taman

Perancangan konstruksi kontrol *monitoring* lampu taman merupakan proses pembuatan desain konstruksi kontrol *monitoring* lampu taman tersebut untuk meletakkan seluruh komponen yang akan digunakan, lalu melakukan pembuatan konstruksi. Setelah tahap pembuatan konstruksi lampu taman selesai maka tahap selanjutnya adalah melakukan perakitan.

Berikut merupakan rancangan dari proyek yang akan dibuat :



Gambar 3. 1 Rancangan Kontruksi Lampu

### 3.4 Perancangan *Hardware* Sistem kontrol dan *monitoring* Lampu Taman

Perancangan *hardware* pada proses pembuatan rancangan bertujuan untuk menentukan komponen yang akan digunakan seperti Arduino uno, lampu LED DC, sensor arus, sensor tegangan, sensor LDR, NodeMCU ESP8266, dan modul DC *light dimmer*, PWM serta MPPT.

Pada perancangan *hardware* ini rancangan yang pertama yaitu tahap perancangan *controller* dan peletakan kontrol nya, berikut merupakan kontroler yang digunakan :

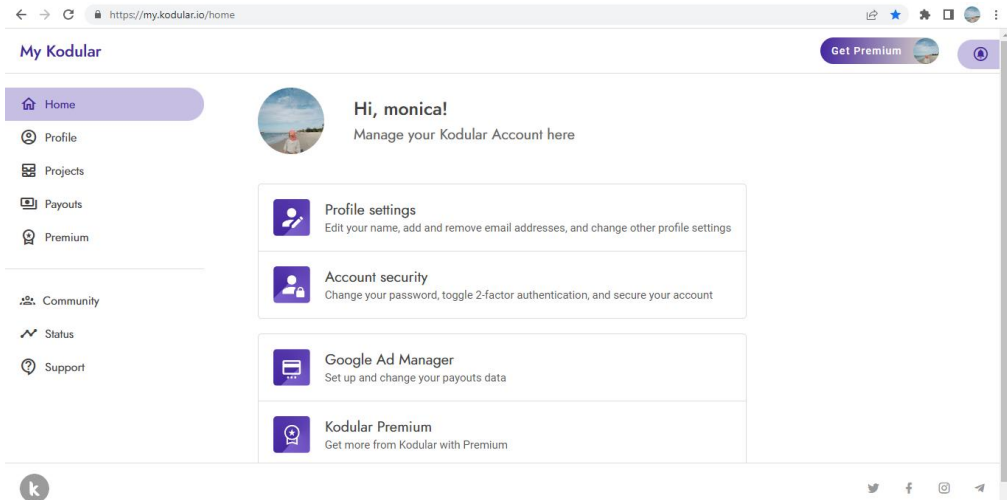


### 3.5 Perancangan *Software* pembuatan aplikasi

Perancangan *software* merupakan proses pembuatan rancangan tampilan aplikasi pada *smartphone* kecerahan lampu dan status aktivitas lampu pada aplikasi menggunakan aplikasi Kodular, perancangan *software* meliputi rancangan tampilan *login*, *history*, *about*, *logout*, serta *device* pada *device* akan ditampilkan *Voltage before*, *voltage after*, *current before*, *current after*, *dimmer PWM*, *status LDR*, serta *time activity* untuk tahap selanjutnya yaitu membuat *software* kontrol *monitoring* lampu taman ini meliputi:

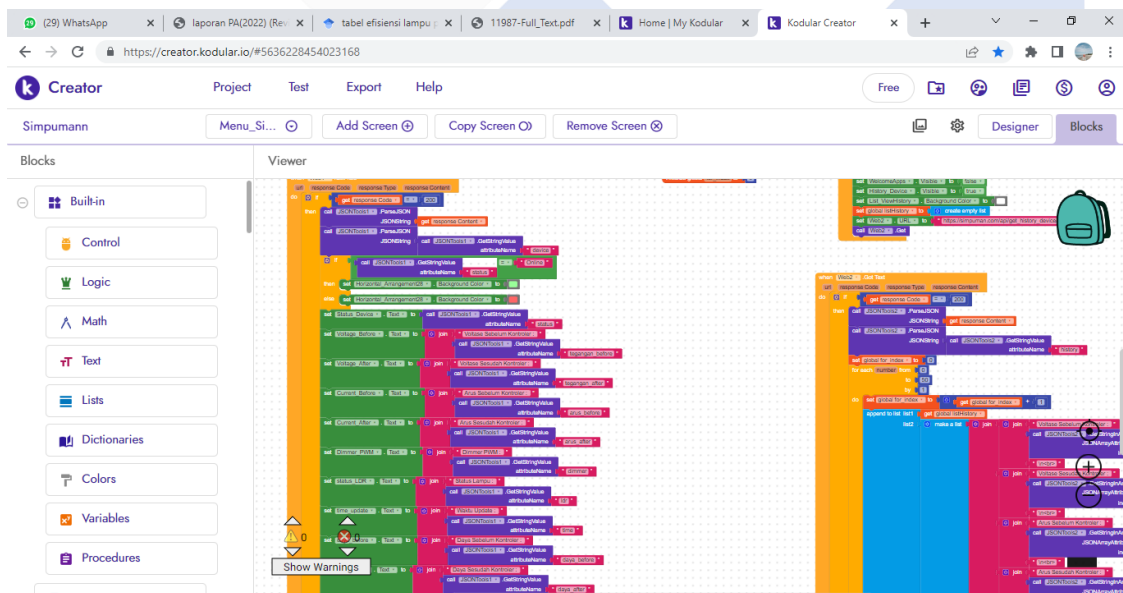
- Pembuatan aplikasi pada *smartphone* dengan menggunakan Kodular
- Pembuatan program keseluruhan pada Arduino Uno
- Pembuatan program keseluruhan pada NodeMCU
- Pembuatan *website* aplikasi Simpunan

- Pada tahap pembuatan aplikasi pertama – tama buka *software* Kodular, maka akan muncul seperti pada gambar 3.3 jika kita sudah pernah *login* dan membuat akun dari *software* Kodular yang ada di laman *website*.



Gambar 3. 3 *Layout* utama *software* Kodular

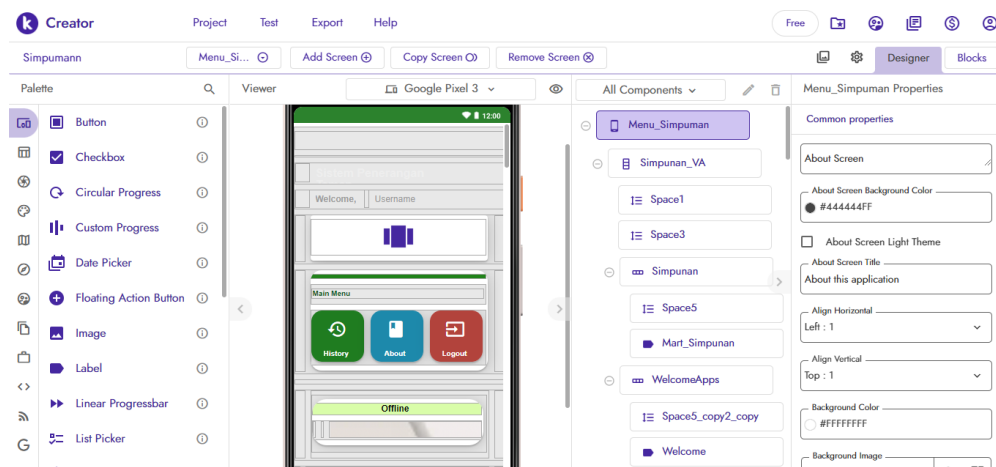
- Setelah itu masuk ke bagian Kodular *creator*, dan mulailah merancang aplikasi *Simpuman* dengan *blocks* pada bagian berikut :



Gambar 3. 4 *Layout* pada bagian *blocks* *software* Kodular

Tampilan diatas merupakan tampilan *blocks program* pada aplikasi yang akan dibuat, dimana program – program yang dimasukkan akan menjadi penyesuaian untuk tampilan aplikasi yang akan penulis buat. Untuk melihat tampilan yang telah terdesain menggunakan *blocks*, maka dapat pergi ke menu bar *design* dan pada menu *design* akan banyak *tools – tools* yang akan menjadi design awal untuk tampilan ketika aplikasi dibuka di Android.

Berikut merupakan rancangan dari aplikasi Simpunan yang di desain menggunakan *software* Kodular :



Gambar 3. 5 Layout Simpunan di *software* Kodular

### 3.6 Rancangan Konstruksi dan *hardware* Elektrik Kontrol Monitoring

Rancangan *hardware* konstruksi dan elektrik dilakukan dengan cara merakit keseluruhan pada setiap bagian dari *hardware* konstruksi dan elektrik digabung menjadi satu kesatuan.



Gambar 3. 6 Konstruksi Lampu Penerangan

### **3.7 Pembuatan Makalah Proyek Akhir**

Pembuatan makalah proyek akhir merupakan tahap terakhir dalam pembuatan proyek akhir yang bertujuan untuk merangkum hasil dari proyek akhir. Dalam makalah proyek akhir ini berisi pendahuluan, landasan teori, pembahasan serta kesimpulan dan saran.





## **BAB IV**

### **PEMBAHASAN**

Pada bab ini akan dibahas mengenai proses serta pengujian alat yang dikembangkan dalam pembuatan Proyek Akhir dengan judul “Sistem Kontrol dan *Monitoring* Lampu Taman Polmanbabel Menggunakan Panel Surya berbasis IoT”, yaitu sebagai berikut:

#### **4.1 Deskripsi Alat**

Sistem kontrol dan *Monitoring* Lampu Taman Polmanbabel Menggunakan Panel Surya Berbasis Iot ini merupakan sistem pengontrolan dan *monitoring* menggunakan Arduino Uno dan NodeMCU ESP8266 sebagai modul Wi-Fi untuk mengirim data ke *Android* dengan Aplikasi yang dibuat menggunakan *software* Kodular yang Penulis beri nama Sistem Penerangan Lampu Taman (Simpuman) yang ditampilkan pada aplikasi tersebut. Pada sistem Lampu ini menggunakan sensor LDR dan *light dimmer* sebagai mikrokontroler dengan media kontrol dan menerima informasi dari kondisi lampu taman melalui aplikasi sebagai kontrol untuk mendeteksi cahaya serta intensitas cahaya, lampu DC dan Modul DC *light dimmer*. Pada saat mengatur lampu melalui Android jika telah menghidupkan *hotspot portable* maka alat akan otomatis tersambung dengan jaringan, maka dengan sendirinya lampu akan hidup menyala terang, dan jika dilihat dari aplikasi Simpuman yang telah dibuat maka pengguna akan dapat melihat nilai tegangan, arus dan, daya yang tampil di aplikasi Simpuman.



Gambar diatas merupakan kontruksi alat pada penerangan lampu taman yang telah dilakukan uji coba dengan cara menghidupkan lampunya ketika perangkat alat yang telah dibuat terhubung ke *hotspot portable* maka lampu akan otomatis hidup dengan *delay* 60 detik dan NodeMCU akan langsung mengirimkan data ke Aplikasi Simpuman.

#### **4.1.1 Hardware Elektrik Lampu Taman Polmanbabel Menggunakan Panel Surya Berbasis Iot**

Proses lain pada proyek akhir ini adalah perancangan, perakitan, dan percobaan terhadap *hardware* elektrik. Pada sub bab ini akan dibahas hardware elektrik yang akan digunakan. Berikut gambar elektrik yang telah dirakit ditunjukkan pada Gambar 4.2

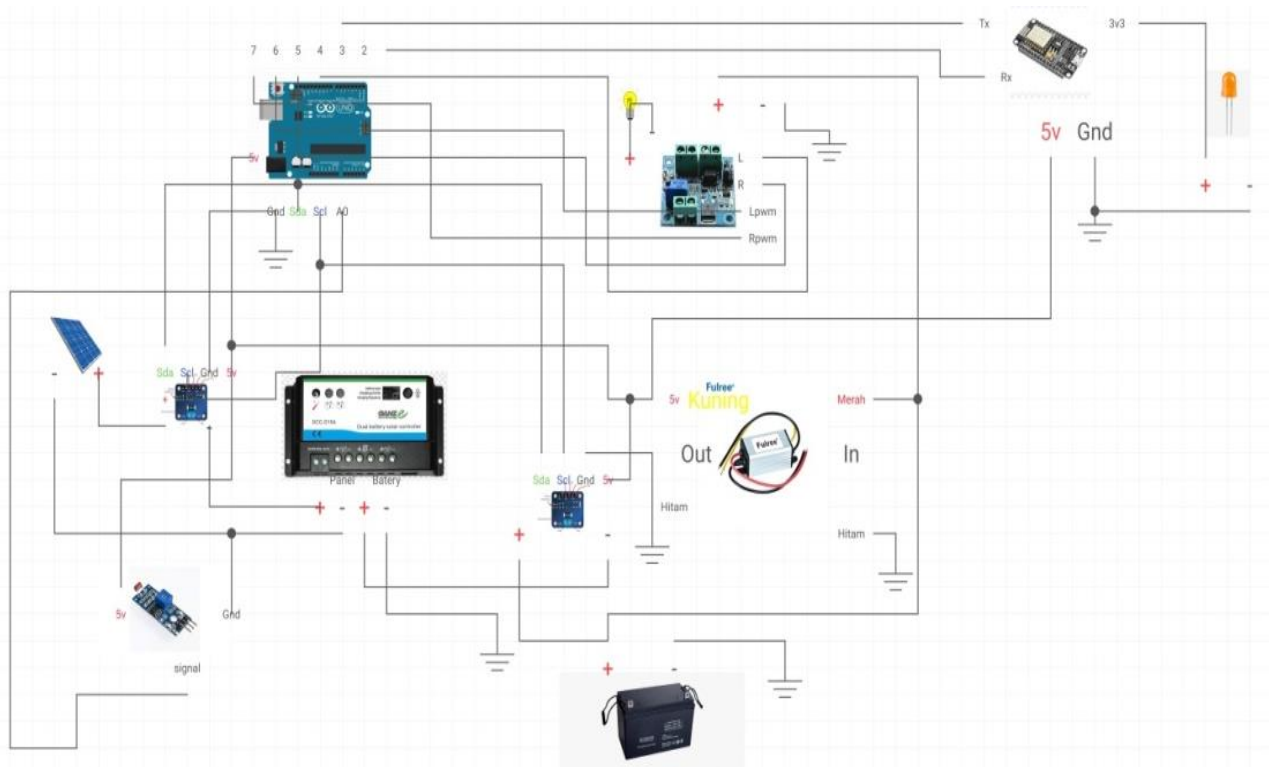


## 4.2 Blok diagram dan *Flowchart*

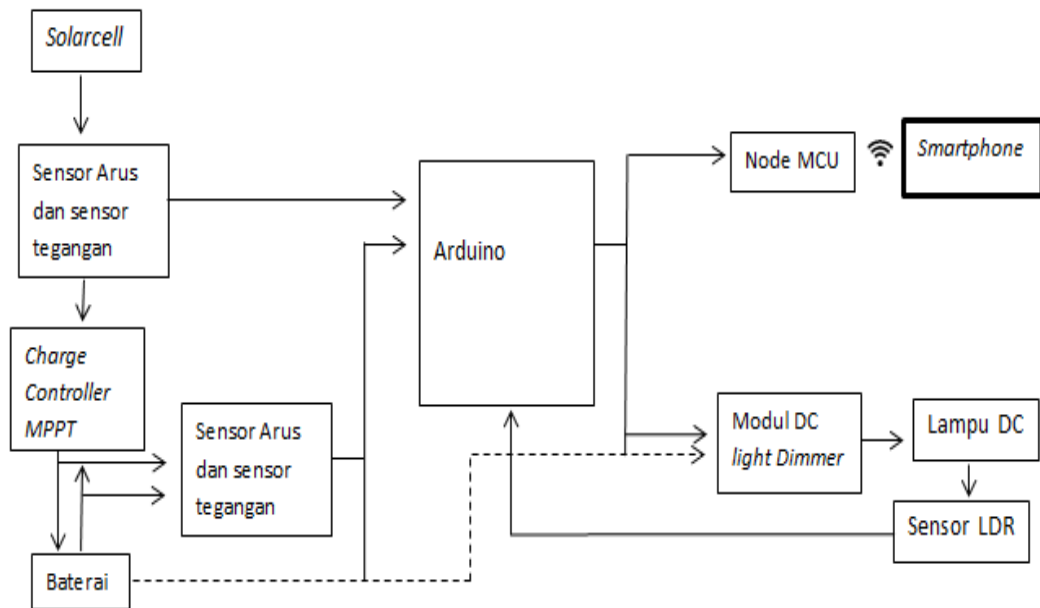
Pada proses perencanaan membuat proyek, penulis membuat *wiring diagram* terlebih dahulu untuk memudahkan proses pembuatan alat dengan melihat alur serta pin pada masing – masing komponen agar tidak salah dalam tahap pembuatan.

### 4.2.1 Blok Diagram *Hardware* Komponen

Pada blok diagram dibawah ini *wiring* diagram agar mempermudah cara pembacaan sistem serta pemilihan alur komponen yang digunakan dan perhubungan kabel – kabel elektriknya.



#### 4.2.2 Blok Diagram Alir



Ket:

Daya cadangan: ----->

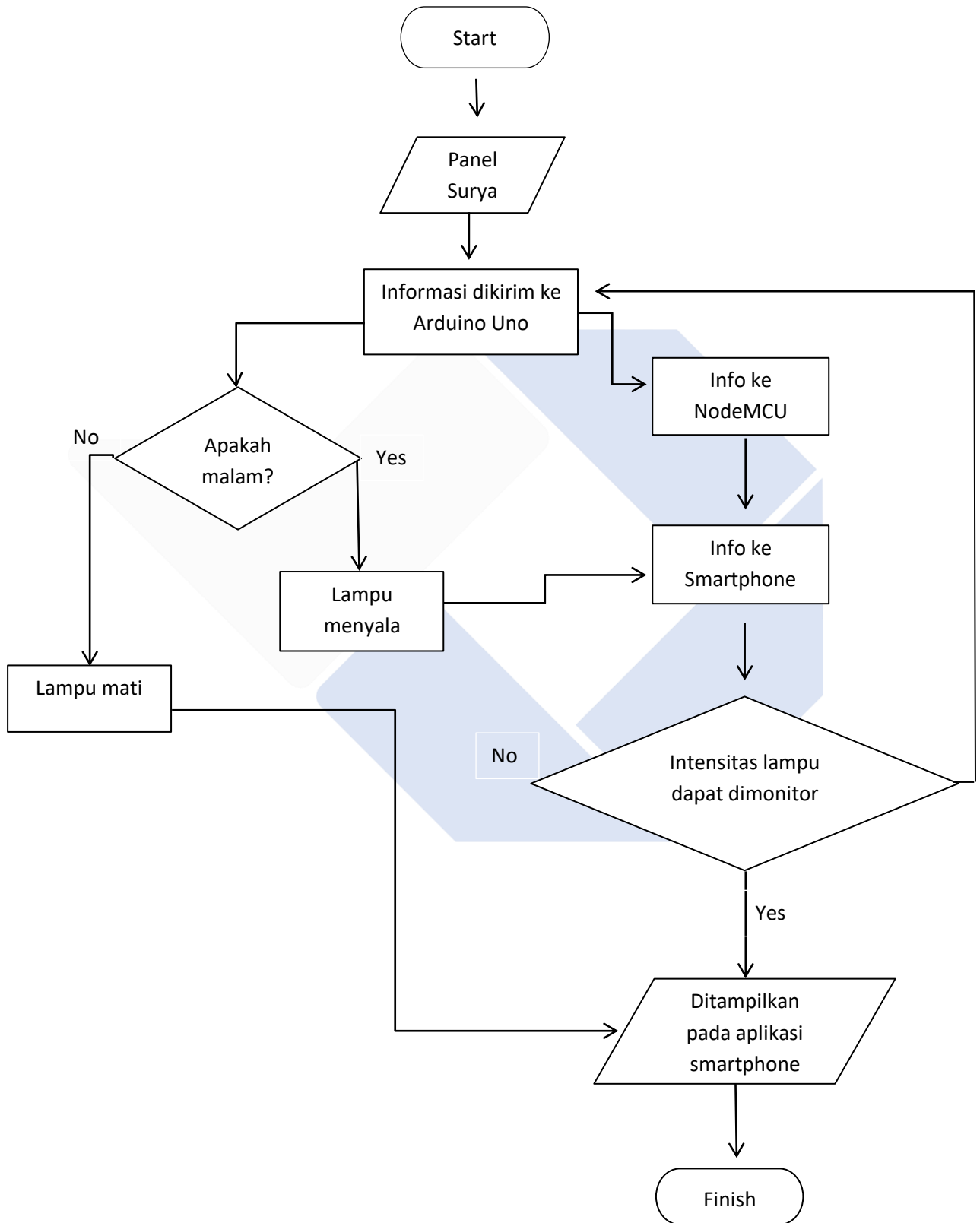
Daya Utama : ———>

Gambar 4.3.1 Blok diagram

Berikut penjelasan dari blok diagram diatas yaitu :

1. *solarcell* / panel surya menyerap cahaya matahari kemudian mengubahnya menjadi energi listrik.
2. Kemudian *solar cell* memberi daya ke sensor arus, sensor tegangan, dan MPPT.
3. Baterai berfungsi untuk menyimpan energi yang diserap oleh panel surya yang berguna sebagai energi cadangan.
4. Arduino Uno dan NodeMCU ESP 8266 berfungsi sebagai kontroler utama.
5. Modul DC light dimmer berfungsi sebagai mengatur intensitas kecerahan lampu.
6. LDR berfungsi sebagai pendeteksi kondisi cahaya lampu terkini.

### 4.2.3 Flowchart



#### 4.4 Penentuan NodeMCU ESP8266

Penentuan kontroler yang digunakan maka dibutuhkan kemampuan pada NodeMCU ESP 8266 dengan kontroler seperti pada Tabel 4.1

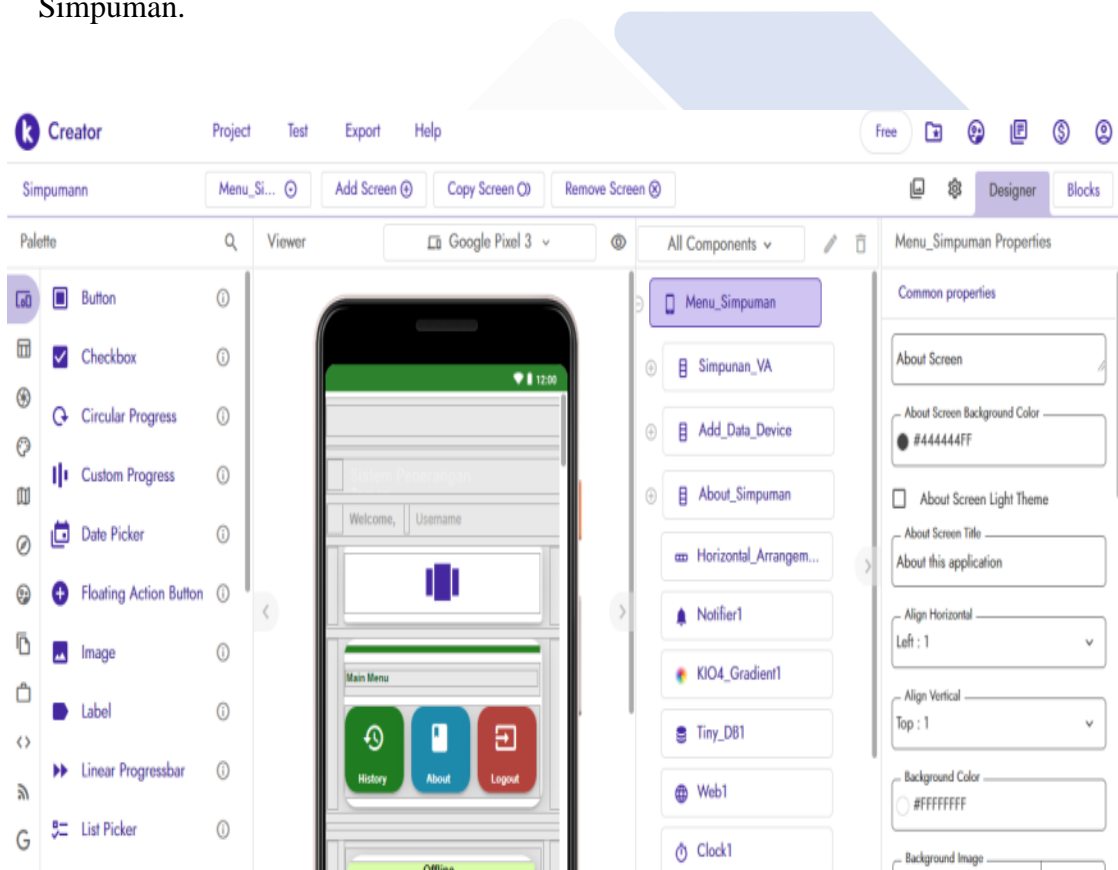
Tabel 4. 1 Pemilihan NodeMCU ESP 8266

No	Deskripsi	Alasan
1	Mempunyai <i>Digital Input/Digital Output</i> (DI/DO) minimal 10.	Sesuai dengan banyaknya pin komponen yang digunakan.
2	<i>Board</i> dengan ukuran kecil.	Agar memudahkan <i>space</i> pada <i>box</i> kontroler.
3	Diutamakan kontroler dengan <i>wireless</i> .	Agar mempunyai kemampuan maksimal dengan piranti yang kecil.
4	Mudah dalam penggunaan.	Agar tidak rumit dalam proses pembuatan proyek akhir.
5	Harga Terjangkau.	Agar hasil alat maksimum tanpa biaya yang besar.

Dalam memilih kontroler yang akan digunakan penulis melakukan perbandingan dengan beberapa kontroler seperti Arduino Nano, Wemos DI ESP8266, dari hasil yang diperoleh penulis akhirnya menggunakan NodeMCU ESP8266. Hal ini karena selain mempunyai kemampuan jaringan *Wi-Fi* NodeMCU ini dapat mendukung perangkat lain yang ingin penulis gunakan. untuk memastikan alat dapat berfungsi dengan baik dan bekerja sesuai dengan hasil akhir yang diinginkan.

#### 4.6 Software Monitoring Lampu Berbasis Android

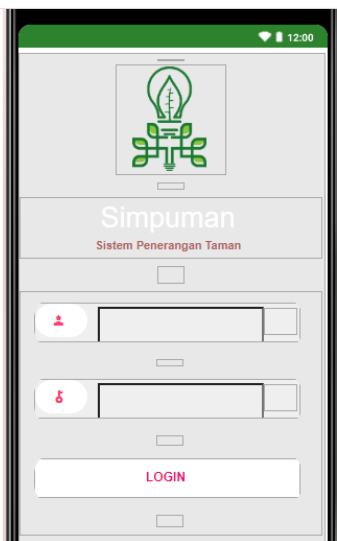
Proyek Akhir “Sistem Kontrol dan Monitoring Lampu Taman Polmanbabel menggunakan Panel Surya Berbasis IoT” menggunakan aplikasi yang dapat digunakan pada Android aplikasi ini digunakan untuk memantau arus, tegangan, daya serta memantau perangkat yang akan dibuat. Dalam hal ini penulis memutuskan membuat aplikasi Android menggunakan *software* Kodular ini mudah digunakan dan dipelajari serta mempunyai fitur yang sesuai dengan aplikasi yang akan dibuat. Berikut merupakan sebagian contoh dari tampilan rancangan *software* Kodular yang digunakan penulis dalam membuat aplikasi Simpunan.



#### 4.6.1 Aplikasi “Simpuman”.

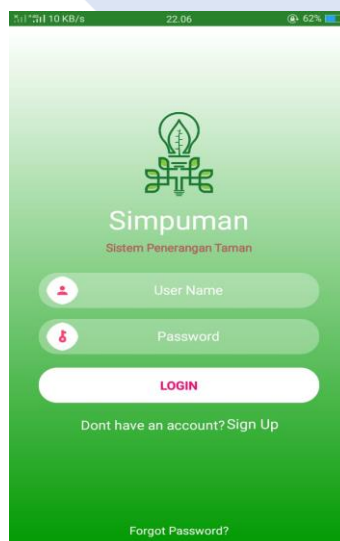
Aplikasi “Simpuman” merupakan aplikasi yang penulis buat untuk memonitoring lampu taman polmanbabel yang dibuat pada proyek akhir ini, aplikasi “Simpuman” ini dibuat dengan *software* Kodular dan untuk pengoperasian yang terhubung dengan perangkat pada NodeMCU ESP8266.

Berikut tampilan utama aplikasi “Simpuman” pada gambar berikut :



```
when Btn_Login Click
do
  set Web1 URL to https://simpuman.com/api/login
  call Web1 Post Text
  join text
  join username
  join User_Text_Box Text
  join Password
  join pass_Text_Box Text
```

```
when Web1 Got Text
do
  call N1Dismiss Dismiss Progress Dialog
  if get response Code = 200
  then
    call JSONTools ParseJSON
    JSONString
    call N1ShowAlert Show Alert
    notice call JSONTools GetStringValue
    attribute Name = message
    if call JSONTools GetStringValue
    attribute Name = status = success
    then
      call N1ShowAlert Show Alert
      notice Login Berhasil
      call T1Store Store Value
      big username
      value To Store User_Text_Box Text
      open another screen screenName Menu_Simpuman
    else
      call N1ShowAlert Show Alert
      notice Server Error
```

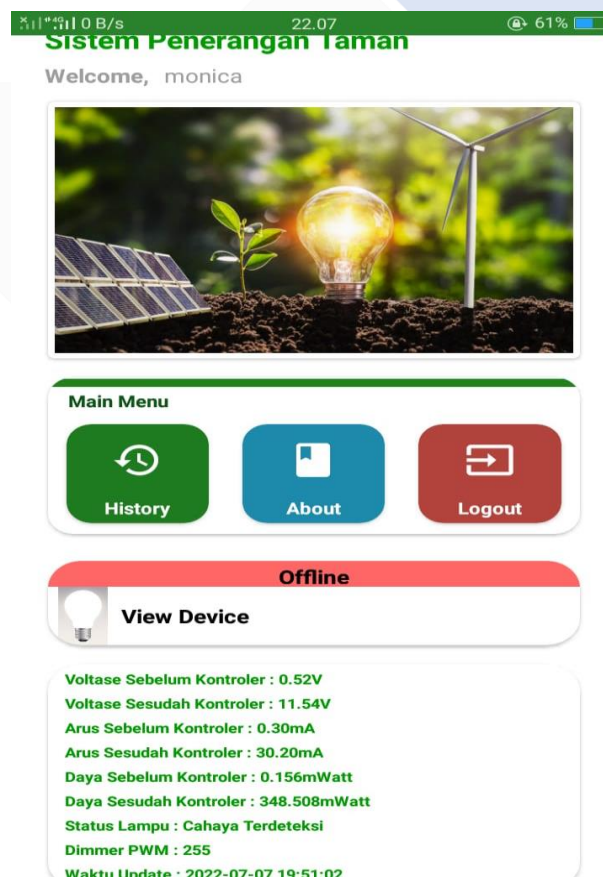




Untuk mengoperasikan aplikasi yang telah dibuat, user harus memasukkan *username* dan *password* yang telah disediakan agar tidak sembarangan pengguna melakukan akses pada aplikasi simpunan tersebut.

#### 4.6.2 Pembuatan aplikasi “Simpunan”

Pembuatan aplikasi “Simpunan” yang digunakan pada *Android* dilakukan dengan menggunakan *software* Kodular.io secara *online* menggunakan komputer. Pembuatan aplikasi *Android* dengan tiga tahap yakni rancangan tampilan, blok editor, dan proses komunikasi. Adapun hasil tampilan yang telah dibuat dapat dilihat pada gambar 4.8 berikut ini:



Gambar 4. 8 Tampilan Menu Aplikasi Simpunan

### 4.6.3 Pengujian Aplikasi “Simpuman”

Pengujian Aplikasi “Simpuman” dilakukan untuk mengetahui apakah alat yang dibuat sesuai dengan fungsi yang diinginkan. Adapun 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 “Simpuman”
4. Masuk ke *menu device* dan lihat apakah status *device online* atau *offline*
5. Jika status *device online* maka perangkat terhubung dengan perangkat keras atau Lampu.
6. Namun, jika status *device offline* maka perangkat keras tidak terhubung dan lakukan pengecekan kabel atau adaptor pada perangkat, lalu masuk kembali ke aplikasi.
7. Setelah *device* terlihat status *online*, maka akan terlihat beberapa *view device* seperti yang telah tertera pada gambar diatas.

Lampu akan tetap menyala dan akan tetap mengirim data ke aplikasi “Simpuman” setiap lima detik setelah *realtime* pada perangkat meskipun *user* keluar dari aplikasi “Simpuman” dikarenakan dalam pengiriman data dari NodeMCU ESP8266 mempunyai waktu kurang lebih selama lima detik, sehingga perubahan data pada Android dapat lebih lama dibandingkan data pada NodeMCU ESP8266 di *serial monitor*. Hal ini juga dilakukan pengaturan waktu yang diberikan penulis pada *list program* yang dibuat untuk mencegah penumpukan data ketika NodeMCU ESP8266 mengirimkan data pada Android.

#### 4.7 Pengujian Sensor LDR

LDR dalam Lampu Taman Polmanbabel menggunakan Panel Surya Berbasis IoT berfungsi untuk mendeteksi keadaan lampu taman terkini yang kemudian data tersebut dikirimkan melalui NodeMCU ke aplikasi Simpuman. Sensor berperan sebagai pengawas lampu apabila terjadi *error* pada lampu taman info akan terkirimkan ke aplikasi Simpuman apabila *user* sedang *online*.

Sensor ini hanya dapat memberikan informasi berupa lampu menyala atau lampu mati saja. *output* yang ditampilkan berupa “Cahaya Terdeteksi” apabila kondisi lampu menyala walaupun redup, dan “Tidak Ada Cahaya” ketika lampu mati ataupun lampu mengalami kerusakan.

Cara kerja LDR yakni akan menerima *output* dari kondisi lampu DC kemudian data tersebut akan dikirimkan arduino ke aplikasi Simpuman dengan NodeMCU sebagai perantaranya. *User* akan menerima informasi tersebut ketika mengecek kondisi lampu melalui aplikasi Simpuman. Pengujian ini menggunakan cara modul sensor LDR dihubungkan ke ArduinoUno.



Pengambilan data dari sensor LDR Lampu Taman Polmanbabel Menggunakan Panel Surya Berbasis IoT berdasarkan 3 situasi yang kemudian data tersebut akan ditampilkan dalam aplikasi Simpuman, berikut data tersebut :

- a. Kondisi lampu menyala

**Status Lampu : Cahaya Terdeteksi**

**Dimmer PWM : 255**

Pada kondisi ini, lampu menyala dalam kondisi tanpa masalah. Dimmer mengkonfirmasi bahwa lampu taman dalam kondisi menyala.

- b. Kondisi lampu menyala namun ada kerusakan

**Status Lampu : Tidak Ada Cahaya**

**Dimmer PWM : 255**

Pada kondisi ini, lampu menyala namun lampu mengalami kerusakan. Dimmer mengkonfirmasi bahwa lampu taman dalam kondisi menyala, namun status menunjukkan bahwa lampu mengkonfirmasi kalau “Tidak Ada Cahaya”.

- c. Kondisi Lampu mati

Kondisi ini merupakan kondisi dimana lampu taman akan padam sesuai dengan jadwalnya. Ketika jadwal padam tiba, Dimmer akan menunjukkan nilainya di angka 0 dan status lampu “Tidak Ada Cahaya”

Berikut merupakan hasil pengujian pada sensor LDR yang tertera pada tabel 4.2 :

Tabel 4. 2 Pengujian Sensor LDR

No.	Waktu Menyala	Dimmer PWM	Status Lampu
1	00:00	255	Cahaya Terdeteksi
2	01:00	255	Cahaya Terdeteksi
3	02:00	255	Cahaya Terdeteksi
4	03:00	255	Cahaya Terdeteksi
5	04:00	255	Cahaya Terdeteksi
6	05:30	50	Cahaya Terdeteksi
7	05:40	25	Cahaya Terdeteksi
8	05:50	1	Cahaya Terdeteksi
9	06:00	0	Tidak Ada Cahaya
10	07:00	0	Tidak Ada Cahaya
11	08:00	0	Tidak Ada Cahaya
12	09:00	0	Tidak Ada Cahaya
13	10:00	0	Tidak Ada Cahaya

---

14	11:00	0	Tidak Ada Cahaya
15	12:00	0	Tidak Ada Cahaya
16	13:00	0	Tidak Ada Cahaya
17	14:00	0	Tidak Ada Cahaya
18	15:00	0	Tidak Ada Cahaya
19	16:00	0	Tidak Ada Cahaya
20	17:30	1	Cahaya Terdeteksi
21	17:40	25	Cahaya Terdeteksi
22	17:50	150	Cahaya Terdeteksi
23	18:00	255	Cahaya Terdeteksi
24	19:00	255	Cahaya Terdeteksi
25	20:00	255	Cahaya Terdeteksi
26	21:00	255	Cahaya Terdeteksi
27	22:00	255	Cahaya Terdeteksi
28	23:00	255	Cahaya Terdeteksi

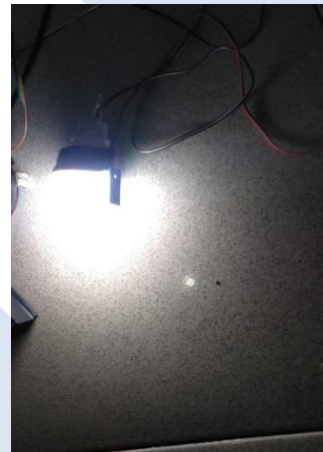
---

Gambar Hasil *output* Dimmer Lampu :



Gambar 4. 11

Kondisi pada siang hari



Gambar 4. 13

Kondisi pada sore hari

Tabel 4.3 Hasil Pengujian LDR dibanding dengan alat ukur Luxmeter

No.	Luxmeter	Sensor LDR
1.	71	75
2.	282	283
3.	303	297
4.	414	427
5.	956	586
6.	988	657
7.	1349	1023

Dari pengujian yang dilakukan diatas dapat dilihat bahwa perbandingan alat ukur luxmeter dengan sensor LDR berbeda dikarenakan sensor LDR memiliki Tahanan 1023 untuk cahaya dengan intensitas cerah sedangkan luxmeter dapat mendeteksi hingga 1349 pada intensitas cerah.

#### 4.8 Pengujian *Monitoring* Tegangan, Arus, dan Daya pada Alat

Untuk melakukan *monitoring* tegangan, arus, daya, dan status cahaya pada *hardware* dan terhubung dengan *Android*. Pengamatan dilakukan sesuai dengan waktu pengecasan dan waktu pemakaian, Berdasarkan hasil pengujian pada *hardware Monitoring* Lampu Taman menggunakan panel surya berbasis IoT, maka dapat dilihat pada tabel 4.1 berikut :

##### 4.8.1 Tabel Pengisian Daya Pada Panel Surya:

Tabel 4. 4 Tabel Pengisian Daya Panel Surya

Saat Pengisian				
Jam	Tegangan	Arus	Daya	Dimmer
09.00 wib	19.90 V	27.70 mA	551.23 mWatt	0
10.00 wib	19.88 V	30.80 mA	612.304 mWatt	0

11.00 wib	19.84 V	58.60 mA	1162.624 mWatt	0
12.00 wib	15.49 V	17.31 mA	26816.288 mWatt	0
13.00 wib	16.42 V	16.76 mA	27531.414 mWatt	0
14.00 wib	16.11 V	17.11 mA	27565.821 mWatt	0

---

Berdasarkan tabel diatas, dengan kondisi cuaca pada pukul 12.00 wib dengan intensitas cahaya matahari telah mencapai titik maksimum maka tegangan, arus, dan daya juga nilainya dapat dilihat dengan persamaan berikut ini, yaitu :

Dengan menggunakan rumus daya (mengambil contoh data pada pukul 12.00 wib) :

$$P = V \times I$$

Dimana,

$$V = 15.49\text{v}$$

$$I = 17.31\text{mA}$$

Maka,

$$P = V \times I$$

$$P = 15.49 \text{ V} \times 17.31 \text{ mA}$$

$$P = 26816.288 \text{ mWatt}$$

Nilai perhitungan diatas sama dengan data hasil pengujian yang dimana cara membacanya terhubungn dengan aplikasi Simpuman, data tersebut tersimpan pada *history* data di *API website* Simpuman.



#### 4.8.2 Tabel Penggunaan Daya Pada Panel Surya

Tabel 4. 5Tabel Penggunaan Daya Pada Panel Surya

Saat Penggunaan				
Jam	Tegangan	Arus	Daya	Dimmer
18.00 wib	13.54 V	- 2034.40 mA	- 27545.776 mWatt	255
19.00 wib	13.53 V	- 2026.70 mA	- 27421.251 mWatt	255
20.00 wib	13.24 V	- 1915.30 mA	- 25358.572 mWatt	255
21.00 wib	11.92 V	- 49.50 mA	- 590.04 mWatt	255
22.00 wib	11.92 V	- 4.00 mA	- 47.68 mWatt	255
23.00 wib	11.95 V	1.10 mA	13.145 mWatt	255

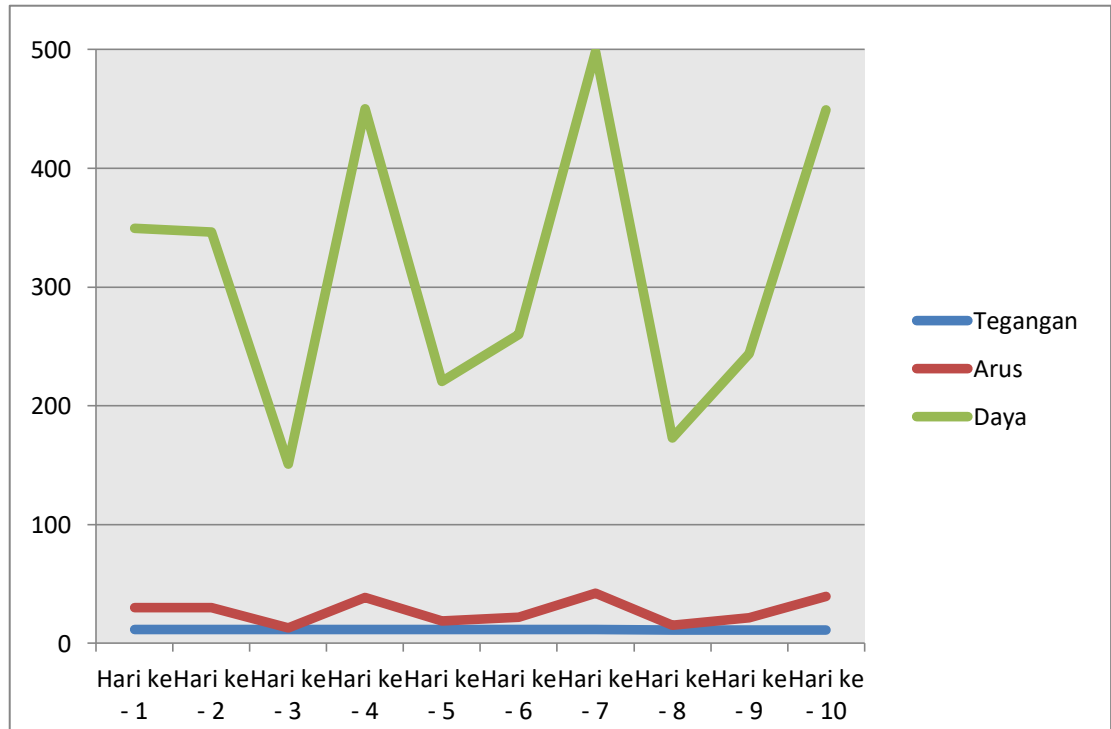
Berdasarkan Tabel diatas, jika pengecasan berlangsung selama pukul 09.00 s/d 14.00 wib (selama kurang lebih 5 jam) maka pengujian menghidupkan lampu mengikuti aturan yang sesuai dengan pengecasan lampu menyala yaitu terjadi Pada pukul 18.00 s/d 23.00 wib (selama kurang lebih 5 jam) maka dapat diambil contoh pada pukul 23.00 wib dimana puncak kecerahan intensitas cahaya lingkungan mulai meredup dengan kecerahan lampu maksimal dengan penggunaan tegangan yang semakin kecil yaitu 11.95 V. maka nilai daya yang keluar pada *output* nya yaitu 13.145 mWatt.

### 4.8.3 Pengujian sistem penerangan taman dalam 10 Hari

Tabel 4. 6 Tabel Pengujian Sistem Penerangan

No.	Hari Ke -	Tegangan (V)	Arus (mA)	Daya (mWatt)
1.	Hari ke - 1	11.54 V	30.30mA	349.662 mWatt
2.	Hari ke - 2	11.58 V	29.90mA	346.242 mWatt
3.	Hari ke - 3	11.60 V	13.00mA	150.8 mWatt
4.	Hari ke - 4	11.60 V	38.80mA	450.08 mWatt
5.	Hari ke - 5	11.66 V	18.90mA	220.374 mWatt
6.	Hari ke - 6	11.76 V	22.10mA	259.896 mWatt
7.	Hari ke - 7	11.75 V	42.40mA	498.2 mWatt
8.	Hari ke - 8	11.38 V	15.20mA	172.976 mWatt
9.	Hari ke - 9	11.39 V	21.40mA	243.746 mWatt
10.	Hari ke - 10	11.40 V	39.40mA	449.16 mWatt

Tabel 4.7 Grafik Hasil Pengujian



Dari grafik hasil pengujian terhadap sistem penerangan taman maka dapat disimpulkan bahwa tingkat intensitas kecerahan lingkungan mempengaruhi hasil lampu yang menyala dikarenakan pada saat pengisian panel surya daya yang dihasilkan menunjukkan status terang redupnya lampu yang menyala.

#### 4.8.4 Pengujian Lampu Sistem Dengan Lampu Konvensional

Tabel 4.8 Pengujian Lampu Sistem dengan Lampu Konvensional

No	Hari	Tegangan (V)		Arus (mA)		Daya (mWatt)	
		Lampu sistem	Konvensional	Lampu sistem	Konvensional	Lampu sistem	Konvensional
1	Ke-1	11.54 V	11.36 V	30.30 mA	11.14 mA	349.62 mWatt	126.55 04 mWatt
2	Ke-2	11.58 V	11.36 V	29.90 mA	11.14 mA	346.242 mWatt	126.55 04 mWatt
3	Ke-3	11.60 V	11.36 V	13.00 mA	11.14 mA	150.8 mWatt	126.55 04 mWatt
4	Ke-4	11.60 V	11.36 V	38.80 mA	11.14 mA	450.08 mWatt	126.55 04 mWatt
5	Ke-5	11.66 V	11.36 V	18.90 mA	11.14 mA	220.37 mWatt	126.55 04 mWatt
6	Ke-6	11.76 V	11.36 V	22.10 mA	11.14 mA	259.89 mWatt	126.55 04 mWatt
7	Ke-7	11.75 V	11.36 V	42.40 mA	11.14 mA	498.2 mWatt	126.55 04 mWatt
8	Ke-8	11.38 V	11.36 V	15.20 mA	11.14 mA	172.97 mWatt	126.55 04 mWatt
9	Ke-9	11.39 V	11.36 V	21.40	11.14	243.74	126.55

				mA	mA	mWatt	04
							mWatt
10	Ke-	11.40 V	11.36 V	39.40	11.14	449.16	126.55
	10			mA	mA	mWatt	04
							mWatt
	<b>Rata-</b>	12.74 V	11.36 V	27.44	11.14	314.11	126.55
	<b>Rata</b>			mA	mA	mWatt	04
							mWatt

Dari data hasil pengujian yang dilakukan selama sepuluh hari dengan sistem pengujian perbandingan antara lampu konvensional dari panel surya langsung menggunakan baterai tanpa ada kontroler dari sistem yang dibuat didapatkan hasil pada tegangan, arus, serta daya yang diperoleh lebih sedikit dibandingkan dengan lampu sesudah kontroler dengan sistem yang dibuat.

Seperti dilihat dari contoh nilai rata – rata daya yang digunakan pada lampu sebesar 314,113 mWatt sedangkan pada lampu konvensional yang digunakan tanpa adanya kontroler hanya memakai daya 126,5504 mWatt.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Berdasarkan hasil pengujian dan analisa terhadap fungsi alat pada proyek akhir dengan judul “Sistem Kontrol Dan Monitoring Lampu Taman Polmanbabel Menggunakan Panel Surya Berbasis IoT” ini maka dapat ditarik kesimpulan sebagai berikut :

1. Pembuatan Aplikasi menggunakan *software* Kodular yang diberi nama “Simpuman” telah berhasil digunakan sehingga dapat *monitoring* tegangan, arus, dan daya pada alat yang dibuat.
2. Nilai tegangan, arus, dan daya dapat ditampilkan pada menu aplikasi Simpuman, sehingga pengguna bisa melihat langsung nilainya pada aplikasi.
3. Rata – rata daya yang dihasilkan dari sistem penerangan lampu yaitu 314,113 mWatt.

#### **5.2 Saran**

Dari proyek akhir yang telah dibuat ada beberapa saran yang disampaikan penulis untuk mengembangkan proyek akhir ini berikutnya, yaitu:

- Untuk kedepannya diharapkan pengembangan aplikasi sistem kontrol bisa dilakukan otomatis dari aplikasi yang telah dibuat agar bisa mengontrol kinerja dari alat tersebut.
- Untuk pengembangan aplikasi diharapkan menggunakan *software Android Studio*, dikarenakan desainnya lebih kompleks sehingga memaksimalkan kinerja dari sistem kontrol dan monitoring lampu taman Polmanbabel menggunakan panel surya berbasis *IoT*.
- Sistem Monitoring dapat juga dilakukan pada beban AC.

## DAFTAR PUSTAKA

- [1] Dyah Nur'ainingsih, Radius Iswanton, and Hartono Siswono, "Lampu Taman Otomatis Menggunakan Solar Tracker Berbasis Mikrokontroler At98s51," *Jurnal Ilmiah FIFO*, p. 124, Volume V/ No. 2/Nov/2013.
- [2] Fadliondi, Didi Cahyadi Riza Samsinar, "System Monitoring dan Perancangan Alat Pendeteksi Kerusakan Lampu Penerangan Jalan Umum (LPJU) Otomatis Berbasis Internet Of Things(IoT)," *RESISTOR(Elektronika Kendali Telekomunikasi Tenaga Listrik Komputer) Vol. 4, No. 2* , p. 169.
- [3] Rijalul Imam,I Gede Putu Wirarama Wedashwara W, Fitri Bimantoro, "Rancang Bangun Sistem Monitoring dan Controlling Penerangan Jalan Umum Berbasis IoT dan Android," *JTIKA. Vol. 2, No. 1*, p. 101, Maret 2020.
- [4] Mugi Alan Prasetya and Rachmat Aulia, "PROTOTYPE PENERANGAN LAMPU TAMAN OTOMATIS MENGGUNAKAN ARDUINO UNO," 2020.
- [5] Rijalul Imam, I W Gede Putu Wirarama Wedashwara, and Fitri Bimantoro, "RANCANG BANGUN SISTEM MONITORING DAN CONTROLLING PENERANGAN JALAN UMUM BERBASIS IOT DAN ANDROID (Design of Monitoring and Controlling for Public Street Lighting Based in IoT and Android)," [Online]. <http://jtika.if.unram.ac.id/index.php/JTIKA/>
- [6] Riyan Wahyu Hidayat, Irma Husnaini, and Jl Hamka Air Tawar, "Perancangan Sistem Kontrol dan Monitoring Lampu Penerangan Tenaga Surya Menggunakan Aplikasi CAYENNE Berbasis IoT," 2021.
- [7] Muharnis Muharnis, Ariadi Ariadi, Jefri Lianda Adam Adam, "Penerapan IoT untuk Sistem Pemantauan Lampu Penerangan Jalan Umum," *ELINVO (Electronics, Informatics, and Vocational Education)*, pp. 32 - 35, Mei, 2020.

- [8] Taufik Akbar, Khairil Anwar Indra Gunawan, "Prototype Sistem monitoring Tegangan Panel Surya (Solar Cell) Pada Lampu Penerangan Jalan Berbasis Web Aplikasi," *Infotek: Jurnal Informatika dan Teknologi*, Vol. 2 No. 2, pp. 73-77, Juli, 2019.
- [9] Muhammad Fitra Zambak, Suwarno Suwarno, Partaonan Harahap Putu Pawitra Teguh Dharma Priatam, "Analisa Radiasi Sinar Matahari Terhadap Panel Surya 50 WP," *RELE (Rekayasa Elektrikal dan Energi) Jurnal Teknik Elektro*, Vol. 4 No. 1, p. 49, Juli, 2021.
- [10] Irma Husnaini Riyan Wahyu Hidayat, "Perancangan Sistem Kontrol dan Monitoring Lampu Penerangan Tenaga Surya Menggunakan Aplikasi CAYENNE Berbasis IoT," *Vol 2 No 2 : JTEIN: Jurnal Teknik Elektro Indonesia*, (2021).
- [11] Muhamad Dawud Eko Ihsanto, "Sistem Kontrol dan MonitoringLapu Penerangan Jalan Umum Menggunakan Mikrokontroler Arduino dan Sensor LDR dengan NotifikaSMS," *Jurnal Teknologi Elektro*, vol. 7. No. 2, no. ISSN:2086-9479, pp. 101 - 104, Mei 2016.
- [12] Basri, M. Amin , Randis, T. Sulistyoyo Wahyu Anhar, "Perhitungan Lampu Penerangan Jalan Berbasis Solar System," *Jurnal Sains terapan*, vol. Vol. 4 No. 1, pp. 34 - 35, Februari 2018.
- [13] Ir. Surya Tarmizi Kasim Msi Donny T B Sihombing, "PERENCANAAN SISTEM PENERANGAN JALAN UMUM DAN TAMAN DI AREAL KAMPUS USU DENGAN MENGGUNAKAN TEKNOLOGI TENAGA SURYA (APLIKASI DI AREAL PENDOPO DAN LAPANGAN PARKIR)," *SINGUDA ENSIKOM*, vol. Vol 3. No. 3, no. 11, pp. 118 - 120, September 2013.
- [14] Ivany Sarief Made Tirta Yasa, "PERENCANAAN PENERANGAN JALAN UMUM TENAGA SURYA (PJUTS) DAN SIMULASI DIALux (STUDI KASUS JALAN KOLONEL MASTURI CIMAHI)," *Jurnal Infotronik*, vol. Vol. 6 No.1, pp. 8 - 10, Juni 2021.



- [15] Pamor Gutomo Reza Nandika, "PEMANFAATAN SEL SURYA 50 Wp PADA LAMPU PENERANGAN RUMAH TANGGA DI DAERAH HINTERLAND," *Sigma Teknika*, vol. Vol. 1 No. 2, pp. 188 - 193, November 2018.
- [16] ANgga RUsdinar, Irwan Purnama Rahmat Iskani Yuskar, "Sistem Penerangan Jalan Umum Bertenaga Surya," *eProceedings of Engineering* , vol. 5 No. 3, no. ISSN: 2355-9365, p. 2, Desember 2018.
- [17] Ardoni Ronaldo, "Pembuatan Aplikasi Mobile "Wonderful Of Minangkabau" Sebagai Gudang Informasi Pariwisata Sumatera Barat melalui Website Kodular," *Jurnal Perpustakaan dan Ilmu Informasi*, vol. 2. No. 1, pp. 90 - 91, 2020.
- [18] Herni Ambarwati Nur Cholid, "Pemngembangan Media Pembelajaran berbasis Android Kodular Materi Zakat Mata Pelajaran Fikih Untuk Meningkatkan Motivasi di Madrasah Ibtidaiyah," vol. 8. No. 2, no. pp.125-136, pp. 125 - 127, Oktober 2021.



**LAMPIRAN 1**

## DAFTAR RIWAYAT HIDUP



### 1. Data Pribadi

Nama Lengkap : Monica  
Tempat/Tanggal Lahir : Bandung, 31 Januari 2002  
Alamat Rumah : JL. Raya Sungailiat Desa Pagarawan 01.  
No. Telepon : 085378398901  
E-mail : [monicatessalonica31@gmail.com](mailto:monicatessalonica31@gmail.com)  
Jenis Kelamin : Perempuan  
Agama : Islam

### 2. Riwayat Pendidikan

SDN 7 PAGARAWAN 2013  
SMPN 2 MERAWANG 2016  
SMAN 4 PANGKALPINANG 2019  
POLITEKNIK MANUFAKTUR NEGERI BANGKA BELITUNG 2022

### 3. Pendidikan Non - Formal

Praktik Kerja lapangan di PT. PJB Services UNIT 4 PLTU SUGE  
BELITUNG.

## DAFTAR RIWAYAT HIDUP

### 1. Data Pribadi

Nama Lengkap : Muhammad Haz Fatahillah Alfaridzi  
Tempat/Tanggal Lahir : Sungailiat , 07 November 1999  
Alamat Rumah : JL. Jend A. Yani, Jalur 2, Gg. Baru,  
Sungailiat.  
No. Telepon : 08982389460  
E-mail : [ayissamsung@gmail.com](mailto:ayissamsung@gmail.com)  
Jenis Kelamin : Laki – Laki  
Agama : Islam

### 2. Riwayat Pendidikan

TK SKB SUNGAILIAT	2005
SDN 25 SUNGAILIAT	2013
MTSN SUNGAILIAT	2016
SMKN 1 SUNGAILIAT	2019
POLITEKNIK MANUFAKTUR NEGERI BANGKA BELITUNG	2022

### 3. Pendidikan Non – Formal

Praktik Kerja lapangan di PT. CITRA PLASTIK MAKMUR, BEKASI



**LAMPIRAN 2**

## 1. Program arduino :

```
#include <SoftwareSerial.h>
#include <Wire.h>
#include <Adafruit_INA219.h>

SoftwareSerial SoftSerial(2, 3);

#define pwmLampu 5
#define ldr A0

int L_EN = 7;
int R_EN = 8;
int nilai;
byte pwmValue;

Adafruit_INA219 sensorBefore(0x40);
Adafruit_INA219 sensorAfter(0x41);

unsigned long lastConnectionTime = 0;
const unsigned long postingInterval = 10000L;
//delay 10 detik tiap pengiriman data sensor

void setup() {
  Serial.begin(9600);
  SoftSerial.begin(9600);

  sensorBefore.begin();
  sensorAfter.begin();

  pinMode(pwmLampu, OUTPUT);
  pinMode(L_EN, OUTPUT);
```

```

pinMode(R_EN, OUTPUT);
digitalWrite(L_EN, HIGH);
digitalWrite(R_EN, HIGH);

analogWrite(pwmLampu, 0);          //lampu off

}

void loop() {
  String DataMasuk = "";
  if (SoftSerial.available() > 0) {
    //ambil data jam dari nodemcu
    DataMasuk = SoftSerial.readStringUntil('%');

    String Jam = parseString(DataMasuk, '#', 0);
    String Menit = parseString(DataMasuk, '#', 1);

    Serial.print("jam : ");
    Serial.println(Jam.toInt());
    Serial.print("menit : ");
    Serial.println(Menit.toInt());

    //jika jam 5 pagi
    if (Jam.toInt() == 5) {
      if (Menit.toInt() == 30) {
        pwmValue = 50;
        analogWrite(pwmLampu, pwmValue);          //255
        full terang - 0 lampu off
        Serial.println("Lampu PWM set 150");
      }
      if (Menit.toInt() == 40) {

```

```
        pwmValue = 25;
        analogWrite(pwmLampu, pwmValue);
        Serial.println("Lampu PWM set 50");
    }
    if (Menit.toInt() == 50) {
        pwmValue = 1;
        analogWrite(pwmLampu, pwmValue);
        Serial.println("Lampu PWM set 10");
    }
}

//jam 6 pagi
if (Jam.toInt() == 6) {
    pwmValue = 0;
    analogWrite(pwmLampu,
                pwmValue);
//lampu off
    Serial.println("Lampu Off");
}
if (Jam.toInt() == 7) {
    pwmValue = 0;
    analogWrite(pwmLampu,
                pwmValue);
//lampu off
    Serial.println("Lampu Off");
}
if (Jam.toInt() == 8) {
    pwmValue = 0;
    analogWrite(pwmLampu,
                pwmValue);
//lampu off
    Serial.println("Lampu Off");
}
if (Jam.toInt() == 9) {
```



```
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 10) {
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 11) {
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 12) {
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 13) {
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 14) {
        pwmValue = 0;
```

```

        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 15) {
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }
    if (Jam.toInt() == 16) {
        pwmValue = 0;
        analogWrite(pwmLampu,                pwmValue);
//lampu off
        Serial.println("Lampu Off");
    }

//jika jam 5 sore
    if (Jam.toInt() == 17) {
        if (Menit.toInt() == 30) {
            pwmValue = 1;
            analogWrite(pwmLampu, pwmValue); //255
full terang - 0 lampu off
            Serial.println("Lampu PWM set 10");
        }
        if (Menit.toInt() == 40) {
            pwmValue = 25;
            analogWrite(pwmLampu, pwmValue);
            Serial.println("Lampu PWM set 50");
        }
        if (Menit.toInt() == 50) {

```

```
        pwmValue = 150;
        analogWrite(pwmLampu, pwmValue);
        Serial.println("Lampu PWM set 150");
    }
}

//jam 6 sore
if (Jam.toInt() == 18) {
    pwmValue = 255;
    analogWrite(pwmLampu, pwmValue);
//lampu on
    Serial.println("Lampu ON Full Terang");
}
if (Jam.toInt() == 19) {
    pwmValue = 255;
    analogWrite(pwmLampu, pwmValue);
//lampu on
    Serial.println("Lampu ON Full Terang");
}
if (Jam.toInt() == 20) {
    pwmValue = 255;
    analogWrite(pwmLampu, pwmValue);
//lampu on
    Serial.println("Lampu ON Full Terang");
}
if (Jam.toInt() == 21) {
    pwmValue = 255;
    analogWrite(pwmLampu, pwmValue);
//lampu on
    Serial.println("Lampu ON Full Terang");
}
```

```
    if (Jam.toInt() == 22) {
        pwmValue = 255;
        analogWrite(pwmLampu,          pwmValue);
//lampu on
        Serial.println("Lampu ON Full Terang");
    }
    if (Jam.toInt() == 23) {
        pwmValue = 255;
        analogWrite(pwmLampu,          pwmValue);
//lampu on
        Serial.println("Lampu ON Full Terang");
    }
    // if (Jam.toInt() == 0) {
    //     pwmValue = 255;
    //     analogWrite(pwmLampu,      pwmValue);
//lampu on
    //     Serial.println("Lampu ON Full Terang");
    // }
    if (Jam.toInt() == 1) {
        pwmValue = 255;
        analogWrite(pwmLampu,          pwmValue);
//lampu on
        Serial.println("Lampu ON Full Terang");
    }
    if (Jam.toInt() == 2) {
        pwmValue = 255;
        analogWrite(pwmLampu,          pwmValue);
//lampu on
        Serial.println("Lampu ON Full Terang");
    }
    if (Jam.toInt() == 3) {
```

```

        pwmValue = 255;
        analogWrite(pwmLampu,                pwmValue);
//lampu on
        Serial.println("Lampu ON Full Terang");
    }
    if (Jam.toInt() == 4) {
        pwmValue = 255;
        analogWrite(pwmLampu,                pwmValue);
//lampu on
        Serial.println("Lampu ON Full Terang");
    }
}

if (millis() - lastConnectionTime > postingInterval) {
    float VoltageBefore, VoltageAfter = 0;
    float currentBefore, currentAfter = 0;
    float powerBefore, powerAfter = 0;

    VoltageBefore = sensorBefore.getBusVoltage_V();
    currentBefore = sensorBefore.getCurrent_mA();
    powerBefore = VoltageBefore * (currentBefore /
1000); // Calculate the Power

    VoltageAfter = sensorAfter.getBusVoltage_V();
    currentAfter = sensorAfter.getCurrent_mA();
    powerAfter = VoltageAfter * (currentAfter / 1000);
// Calculate the Power

    nilai = analogRead(ldr);                //1023 =
tidak ada cahaya terdeteksi

```

```
String flagLDR = "Tidak Ada Cahaya";
if (nilai < 300) {
    flagLDR = "Cahaya Terdeteksi";
}

Serial.print("Nilai LDR : ");
Serial.println(nilai);

Serial.print(" LDR : ");
Serial.println(flagLDR);

Serial.print(" Voltage Before: ");
Serial.print(VoltageBefore);
Serial.println(" V");

Serial.print(" Voltage After: ");
Serial.print(VoltageAfter);
Serial.println(" V");

Serial.print("Current Before: ");
Serial.print(currentBefore);
Serial.println(" mA");

Serial.print(" Current After: ");
Serial.print(currentAfter);
Serial.println(" mA");

Serial.print("Power Before: ");
Serial.print(powerBefore);
```

```

Serial.println(" W");

Serial.print("Power After:          ");
Serial.print(powerAfter);
Serial.println(" W");

Serial.println("");

String DataKeluar = "=1#" + String(VoltageBefore) +
"V#" + String(VoltageAfter) + "V#" +
String(currentBefore) + "mA#" + String(currentAfter) +
"mA#" + String(pwmValue) + "#" + flagLDR + "#Online=%";
Serial.println(DataKeluar);
SoftSerial.print(DataKeluar);

Serial.println("");
Serial.println("");

lastConnectionTime = millis();
}
}

```

```

String parseString(String data, char separator, int
index) {
int found = 0;
int strIndex[] = {0, -1};
int maxIndex = data.length() - 1;
for (int i = 0; i <= maxIndex && found <= index; i++)
{

```

```
    if (data.charAt(i) == separator || i == maxIndex) {
        found++;
        strIndex[0] = strIndex[1] + 1;
        strIndex[1] = (i == maxIndex) ? i + 1 : i;
    }
}
return found > index ? data.substring(strIndex[0],
strIndex[1]) : "";
}
```







**LAMPIRAN 3**

## 2. Program NodeMCU ESP 8266

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <SoftwareSerial.h>
#include <ESP8266HTTPClient.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <TimeLib.h>

ESP8266WiFiMulti wifiMulti;
SoftwareSerial SoftSerial(D6, D5);

String ID;
String VoltageBefore;
String VoltageAfter;
String CurrentBefore;
String CurrentAfter;
String pwmDimmer;
String LDRstatus;
String StatusDevice;

const String Url
  ="http://simpuman.com/api/send_data_device";
const long utcOffsetInSeconds = 25200; // set
  UTC + 7 (jakarta) 7x60x60
// Define NTP Client to get time
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org",
  utcOffsetInSeconds);
```

```
unsigned long lastConnectionTime = 0;           // last
time you connected to the server, in milliseconds
const unsigned long postingInterval = 10000L;
String Hour, Min;

void setup() {
  Serial.begin(9600);
  SoftSerial.begin(9600);

  delay(1000);
  Serial.println("Starting system");

  WiFi.mode(WIFI_STA);
  wifiMulti.addAP("monica", "12345678");

  Serial.println("Connecting Wifi...");
  if (wifiMulti.run() == WL_CONNECTED) {
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());
  }

  timeClient.begin();
}

void loop() {
  timeClient.update();

  if (millis() - lastConnectionTime > postingInterval) {
```

```

    showDateandTime();
    lastConnectionTime = millis();
}

String DataMasuk = "";
if (SoftSerial.available() > 0) {
    DataMasuk = SoftSerial.readStringUntil('%');

    Serial.println(DataMasuk);

    if (DataMasuk.substring(0, 1) == "=") {
        if (DataMasuk.substring(DataMasuk.length() - 1,
DataMasuk.length()) == "=") {
            DataMasuk.remove(0,1);
            DataMasuk.remove(DataMasuk.length() - 1,
DataMasuk.length());
            //Serial.println(DataMasuk);

            ID = parseString(DataMasuk, '#', 0);
            VoltageBefore = parseString(DataMasuk, '#', 1);
            VoltageAfter = parseString(DataMasuk, '#', 2);
            CurrentBefore = parseString(DataMasuk, '#', 3);
            CurrentAfter = parseString(DataMasuk, '#', 4);
            pwmDimmer = parseString(DataMasuk, '#', 5);
            LDRstatus = parseString(DataMasuk, '#', 6);
            StatusDevice = parseString(DataMasuk, '#', 7);

            Serial.print("ID Device : ");
            Serial.println(ID);

            Serial.print("Voltage Before:  ");

```

```
Serial.print(VoltageBefore);
Serial.println(" V");

Serial.print("Voltage After:  ");
Serial.print(VoltageAfter);
Serial.println(" V");

Serial.print("Current Before:  ");
Serial.print(CurrentBefore);
Serial.println(" mA");

Serial.print("Current After:  ");
Serial.print(CurrentAfter);
Serial.println(" mA");

Serial.print("pwm Dimmer : ");
Serial.println(pwmDimmer);

Serial.print("LDR Status : ");
Serial.println(LDRstatus);

Serial.print("Status Device : ");
Serial.println(StatusDevice);

Serial.println();
Serial.println();

if (wifiMulti.run() == WL_CONNECTED) {
  WiFiClient client;
  HTTPClient http;
```

```

        String postData = "id=" + ID +
"&tegangan_before=" + VoltageBefore +
"&tegangan_after=" + VoltageAfter + "&arus_before=" +
CurrentBefore + "&arus_after=" + CurrentAfter +
"&dimmer=" + pwmDimmer + "&ldr=" + LDRstatus +
"&status=" + StatusDevice;

        Serial.println("[HTTP] begin...\n");
        Serial.println(postData);

        http.begin(client, Url);
        http.addHeader("Content-Type", "application/x-
www-form-urlencoded");

        int httpCode = http.POST(postData);
        if (httpCode > 0) {
            Serial.printf("[HTTP] POST... code: %d\n",
httpCode);

            if (httpCode == HTTP_CODE_OK) {
                String payload = http.getString();
                Serial.println(payload);
            } else {
                Serial.print("error ");
                Serial.println(httpCode);
            }
        } else {
            Serial.printf("[HTTP] POST... failed, error:
%s\n", http.errorToString(httpCode).c_str());
        }
    }
}

```

```
        delay(1000);
    } else {
        Serial.println("WiFi not connected!");
        delay(1000);
    }
} else {
    Serial.println("Data Masuk Error");
}
} else {
    Serial.println("Data Masuk Error");
}
}

delay(10);
}

void showDateandTime() {
    Serial.println();
    //Serial.println(timeClient.getEpochTime());
    Serial.println(timeClient.getFormattedTime());
    Serial.printf("Date: %4d-%02d-%02d %02d:%02d:%02d\n",
year(timeClient.getEpochTime()),
month(timeClient.getEpochTime()),
day(timeClient.getEpochTime()),
hour(timeClient.getEpochTime()),
minute(timeClient.getEpochTime()),
second(timeClient.getEpochTime()));

    Hour = String(hour(timeClient.getEpochTime()));
    Min = String(minute(timeClient.getEpochTime()));
```

```
String DataKeluar = Hour + "#" + Min + "%";
Serial.println(DataKeluar);
SoftSerial.print(DataKeluar);
}

String parseString(String data, char separator, int
index) {
    int found = 0;
    int strIndex[] = {0, -1};
    int maxIndex = data.length() - 1;
    for (int i = 0; i <= maxIndex && found <= index; i++)
    {
        if (data.charAt(i) == separator || i == maxIndex) {
            found++;
            strIndex[0] = strIndex[1] + 1;
            strIndex[1] = (i == maxIndex) ? i + 1 : i;
        }
    }
    return found > index ? data.substring(strIndex[0],
strIndex[1]) : "";
}
```

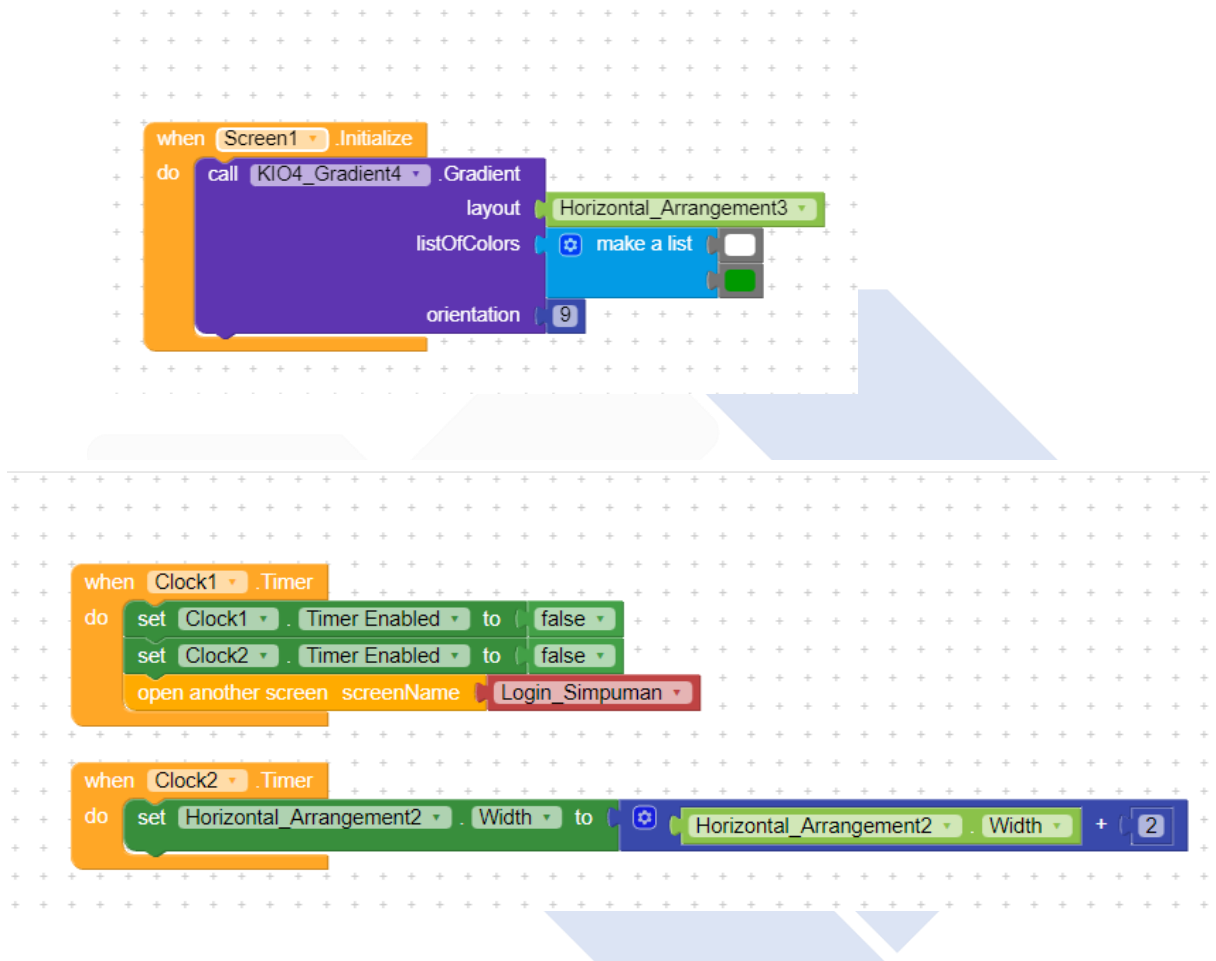




**LAMPIRAN 4**

### 3. Program Aplikasi Simpunan :

#### Menu Utama

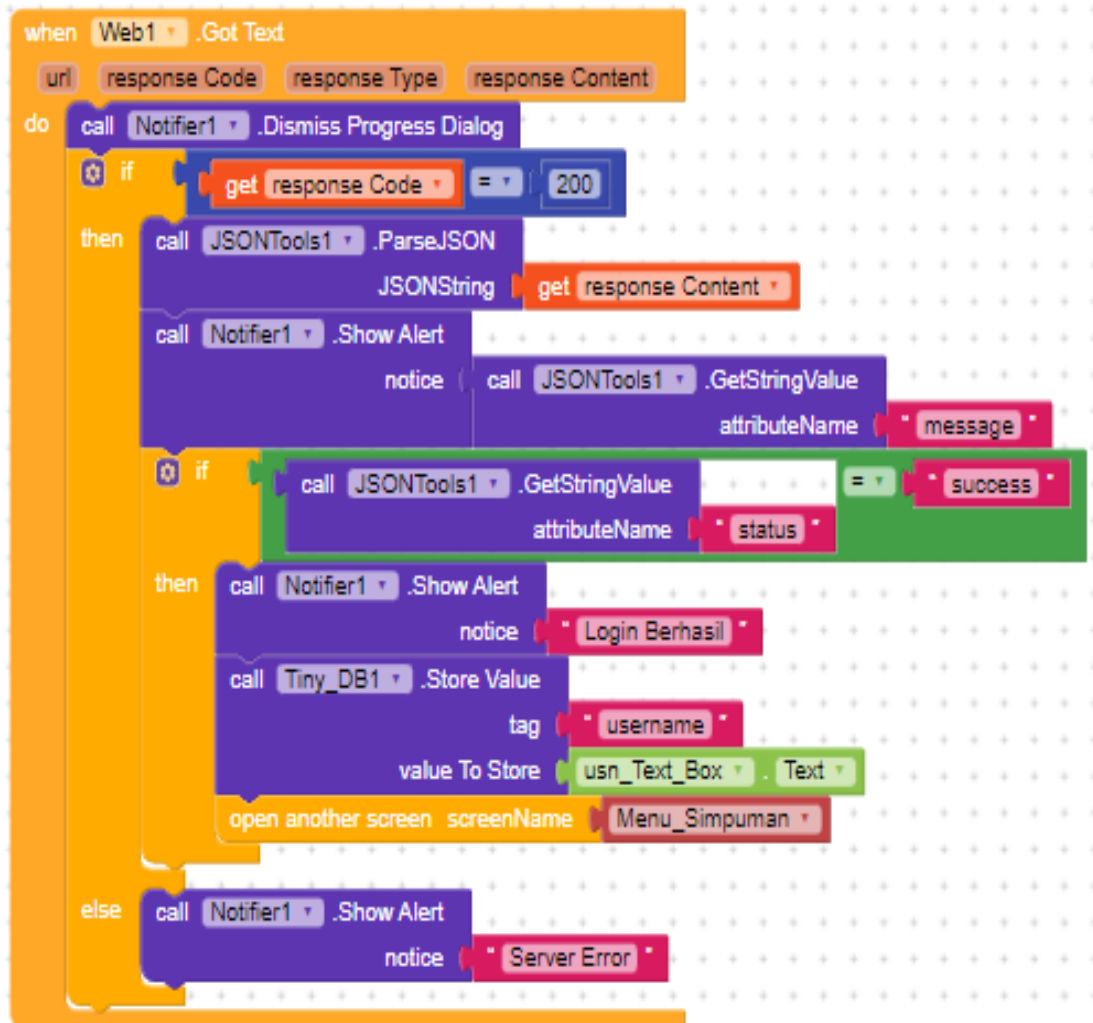


## Menu Login

```
when Login_Simpuman .Back Pressed
do close application
```

```
when Login_Simpuman .Initialize
do call KIO4_Gradient1 .Gradient
    layout Main_var
    listOfColors make a list
    orientation 9
set Login_Simpuman . Navigation Bar Color to
if call Tiny_DB1 .Get Value
    tag username
    value If Tag Not There
then call Notifier1 .Show Alert
    notice Auto Login
open another screen screenName Menu_Simpuman
```

```
when Btn_Login .Click
do set Web1 . URL to https://simpuman.com/api/login
call Web1 .Post Text
    text join join username=
        usn_Text_Box . Text
        join &password=
            pass_Text_Box . Text
```

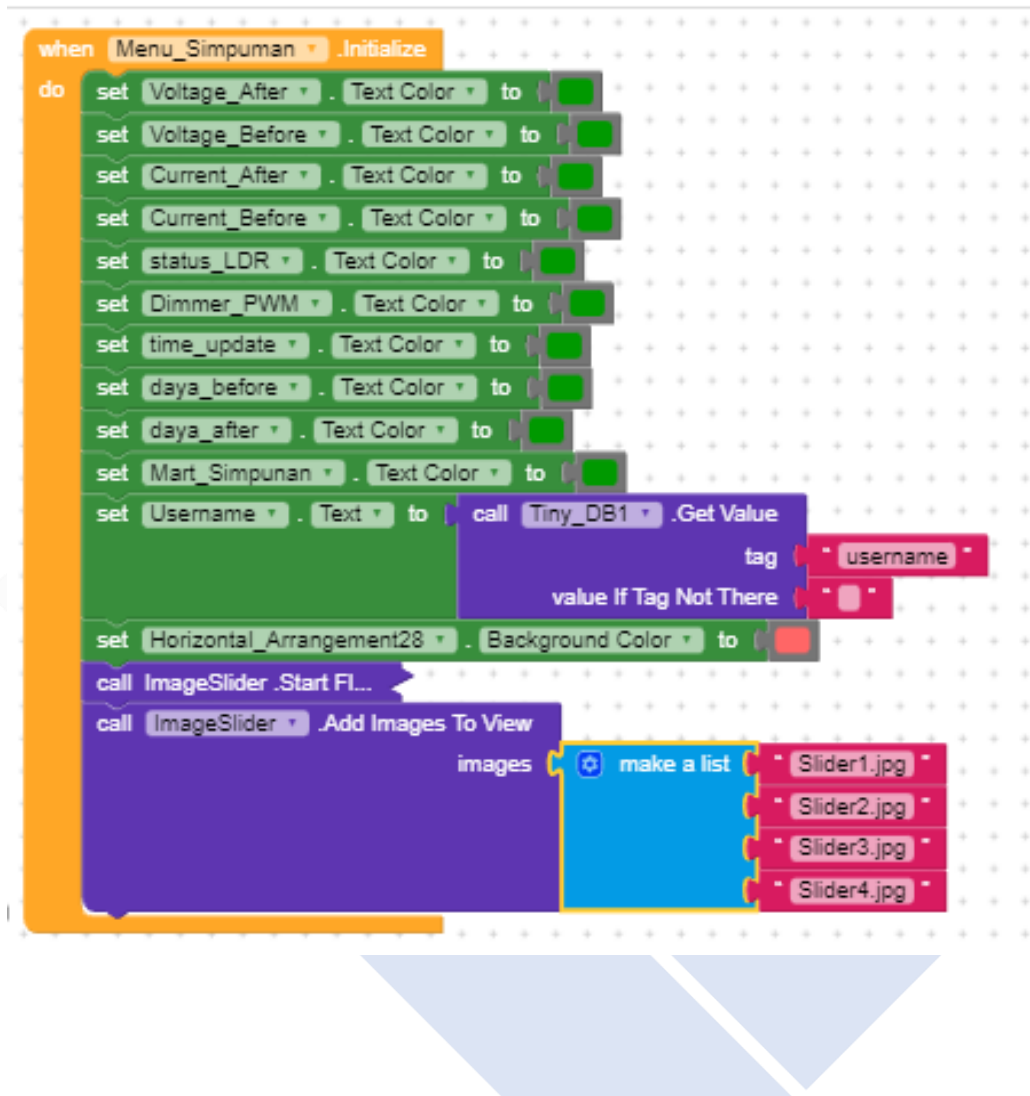


## Menu Logout

```
when Menu_Simpuman ▾ .Back Pressed  
do close application
```

```
when Logout ▾ .Click  
do call Tiny_DB1 ▾ .Clear All  
close application
```

## Menu Tampilan



## Menu About



## Menu History

```
when History.Click
do
  set List_ViewHistory.Text Color to #008000
  set Clock1.Timer Enabled to false
  set Device_Details.Visible to false
  set Main_Menu.Visible to false
  set List_Data_Device.Visible to false
  set Simpunan.Visible to false
  set Slider_Simpunan.Visible to false
  set WelcomeApps.Visible to false
  set History_Device.Visible to true
  set List_ViewHistory.Background Color to #FFFFFF
  set global listHistory to create empty list
  set Web2.URL to "https://simpuman.com/api/get_history_device/1"
  call Web2.Get
```



## Menu Device

