

## **Prototype Saklar Otomatis Berbasis Android Dan Arduino**

**Muhammad Ridha<sup>1\*</sup>, Boy Abidin Rozany<sup>2</sup>**

<sup>1,2</sup>Program Studi Teknik Informatika, STMIK Banjarbaru  
 Jl. A. Yani Km. 33,3 Banjarbaru, Telp (0511) 4782881  
<sup>2</sup>boy.abidin@gmail.com

\* *Corresponding Author*: ridharemixdj@gmail.com

### **Abstrak**

Pada umumnya menyalakan peralatan elektronik seperti lampu, kipas, AC (*Air Conditioning*) dan lain-lain harus menekan saklar secara manual, hal ini tidak efisien terutama bagi penyandang cacat dan orang tua lanjut usia. Jumlah saklar yang banyak menyebabkan terbuangnya waktu dan tenaga untuk menyalakan saklar-saklar tersebut. Ketika seseorang lupa mematikan saklar lampu dipagi hari karena berbagai hal seperti buru-buru kekantor dan lain sebagainya maka terjadi pemborosan penggunaan listrik. Artikel ini menyajikan model saklar berbasis arduino yang dapat digunakan untuk mengontrol langsung saklar peralatan elektronik dari jarak jauh melalui sistem android, dan dapat membuat saklar bekerja secara otomatis. Sistem Android digunakan sebagai *remote* pengontrol, pengatur waktu otomatis, dan sebagai monitor untuk mengetahui status aktif/mati saklar. Kit Arduino uno digunakan sebagai controller untuk mengatur perintah sesuai dengan data yang diterima dari sistem android. Adapun Sensor RTC DS3231 digunakan sebagai pewaktuan pada arduino, dan *bluetooth* HC-05 digunakan sebagai jaringan komunikasi antara android dan arduino. Terdapat juga komponen Relay yang digunakan sebagai output dari perintah yang diberikan arduino, dan terhubung dengan lampu dan AC. Hasil pengujian *Black box* menunjukkan alat dapat berfungsi sebagai saklar otomatis, dan efektif mengendalikan hidup-mati peralatan elektronik dalam jumlah yang banyak sesuai dengan waktu yang telah ditentukan.

**Kata kunci:** Saklar Otomatis, Mikrokontroler, Arduino Uno, Berbasis Android

### **Abstract**

*In general, turning on electronic equipment such as lights, fans, air conditioning (AC) and others must press the switch manually, this is not efficient especially for people with disabilities and elderly people. The large number of switches causes the waste of time and energy to turn on the switches. When someone forgets to turn off the light switch in the morning because of various things such as rush to the office and so on, there is a waste of electricity usage. This article presents an arduino-based switch model that can be used to directly control electronic equipment switches remotely through the Android system, and can make the switch work automatically. The Android system is used as a remote controller, automatic timer, and as a monitor to find out the status of the on / off switch. The Arduino Uno Kit is used as a controller to manage commands according to data received from the Android system. The DS3231 RTC Sensor is used as a timing for Arduino, and Bluetooth HC-05 is used as a communication network between Android and Arduino. There is also a Relay component that is used as the output of the commands given arduino, and is connected to lights and AC. Black box test results show the tool can function as an automatic switch, and effectively control the life and death of electronic equipment in large quantities in accordance with a predetermined time.*

**Keywords:** Automatic Switch, Microcontroller, Arduino Uno, Based on Android

### **1. Pendahuluan**

Umumnya orang dalam menyalakan peralatan elektronik seperti lampu, kipas, AC dan lain-lain harus menekan saklar secara manual, hal ini tentu saja sangat tidak efisien terutama bagi penyandang cacat dan orang tua lanjut usia, selain itu jika jumlah saklar yang dinyalakan ada banyak tentu saja menyebabkan terbuangnya waktu dan tenaga untuk menyalakan saklar-

saklar tersebut, kemudian jika seseorang lupa mematikan saklar lampu dipagi hari karena berbagai hal seperti buru-buru kekantor dan lain sebagainya maka tentu saja terjadi pemborosan terhadap penggunaan listrik. Oleh sebab itu di perlukanlah sebuah sistem saklar otomatis yang dapat mempermudah pengontrolan peralatan elektronik tersebut.

Arduino Uno adalah sebuah rangkaian sistem kendali multifungsi yang dikembangkan dari mikrokontroler berbasis ATmega328. Berbagai kelebihan yang dapat ditemukan pada sistem Arduino diantaranya adalah tidak perlu perangkat chip programmer karena didalamnya sudah ada bootloader yang akan menangani upload program dari komputer, Arduino sudah memiliki sarana komunikasi USB, sehingga pengguna laptop yang tidak memiliki port serial/RS232 bisa menggunakannya. Bahasa pemrograman relatif mudah karena *software* Arduino dilengkapi dengan kumpulan library yang cukup lengkap, dan Arduino memiliki modul siap pakai (*shield*) yang bisa ditancapkan pada *board* Arduino. Misalnya shield GPS, Ethernet, SD Card, dan lain-lain [1]. Arduino telah digunakan sebagai sistem pengendali pada berbagai bidang. Silvia, Haritman, & Muladi [2], Reswandi, Pangaribuan, & Atmaja [3], dan Sari, Hardyanto [4], telah menguji penggunaan Arduino sebagai sistem pengendali berbagai jenis Pintu. Rinaldy, Chtistianti, & Supriyadi [5], Birdayansyah, Soedjarwanto, & Zebua [6] telah menguji Arduino untuk sistem kendali perputaran motor. Arduino juga telah diujicoba penggunaannya sebagai bagian dari sistem peringatan dini [7][8] dan sebagai fungsi saklar [9][10][11].

Artikel ini menyajikan model saklar berbasis arduino yang dapat digunakan untuk mengontrol langsung saklar peralatan elektronik dari jarak jauh melalui sistem android, dan dapat membuat saklar bekerja secara otomatis

## 2. Tinjauan Pustaka

Penelitian mengenai sistem saklar pengendali peralatan rumah tangga telah banyak dilakukan. Penelitian yang dilakukan oleh Liliana, Welman J mengenai *prototype* penerangan rumah otomatis berbasis mikrokontroler ATmega8535, pada penelitian tersebut dirancangnya sebuah sistem penerangan secara otomatis menggunakan sensor LDR atau cahaya dan juga sensor PIR atau gerak yang mana semuanya dikontrol melalui sebuah mikrokontroler ATmega8535. Terdapat perbedaan antara penelitian tersebut dengan penelitian ini yaitu: pengontrolan yang digunakan berbeda, pada penelitian ini, pengontrolan dilakukan melalui smartphone android dan menggunakan sensor RTC [12].

Penelitian dilakukan oleh Sutono mengenai perancangan sistem otomatisasi lampu penerangan menggunakan sensor gerak dan sensor cahaya berbasis arduino uno ATmega 328, pada penelitian tersebut dirancang sistem saklar otomatis untuk mengoperasikan beban lampu penerangan suatu ruangan menggunakan masukan sensor kehadiran orang jenis *passive infrared* PIR dan sensor intensitas cahaya jenis *light dependent resistor* LDR. Terdapat perbedaan antara penelitian tersebut dengan penelitian ini yaitu: pengontrolan yang digunakan berbeda, pada penelitian ini, pengontrolan dilakukan melalui smartphone android dan menggunakan sensor RTC [13].

Penelitian dilakukan oleh Eka Desyantoro mengenai sistem pengendali peralatan elektronik dalam rumah secara otomatis menggunakan sensor PIR, LM35, dan sensor LDR, pada penelitian tersebut dirancang sistem pengendali peralatan listrik dengan sensor PIR yang berfungsi untuk mendeteksi objek bergerak, sensor LM35 yang berfungsi mendeteksi suhu, dan sensor LDR yang berfungsi mendeteksi cahaya yang dihubungkan dengan ATmega16 sebagai pengendali sistem dan LCD untuk menampilkan data sensor dan mengatur relay, Terdapat perbedaan antara penelitian tersebut dengan penelitian ini yaitu: pengontrolan yang digunakan berbeda, pada penelitian ini, pengontrolan dilakukan melalui smartphone android dan menggunakan sensor RTC [14].

Konsep yang diajukan pada artikel ini adalah penggunaan sistem android sebagai remote pengontrol pada saklar, pengatur *timer* waktu aktif/mati saklar secara otomatis, dan indikator untuk mengetahui status saklar. Konsepnya untuk mode manual android melakukan pengiriman data ke arduino melalui koneksi bluetooth, kemudian setelah data diterima, arduino akan mengirimkan data kembali ke android untuk menyalakan indikator saklar pada aplikasi dan memberikan perintah kepada relay (saklar) yang sudah terhubung dengan peralatan elektronik seperti lampu baik itu aktif ataupun mati. Untuk mode otomatis konsepnya hampir sama, pertama android mengirimkan data *timer* waktu aktif/mati saklar ke arduino, kemudian setelah data diterima, program arduino menginterupsi atau menghentikan program manual dan masuk

kemode otomatis, selanjutnya jika data waktu pada sensor RTC syaratnya sudah terpenuhi dengan data waktu aktif/mati yang sudah diset dan disimpan pada memory arduino sebelumnya, maka arduino akan memberikan perintah ke relay baik itu aktif maupun mati.

**3. Metode Penelitian**

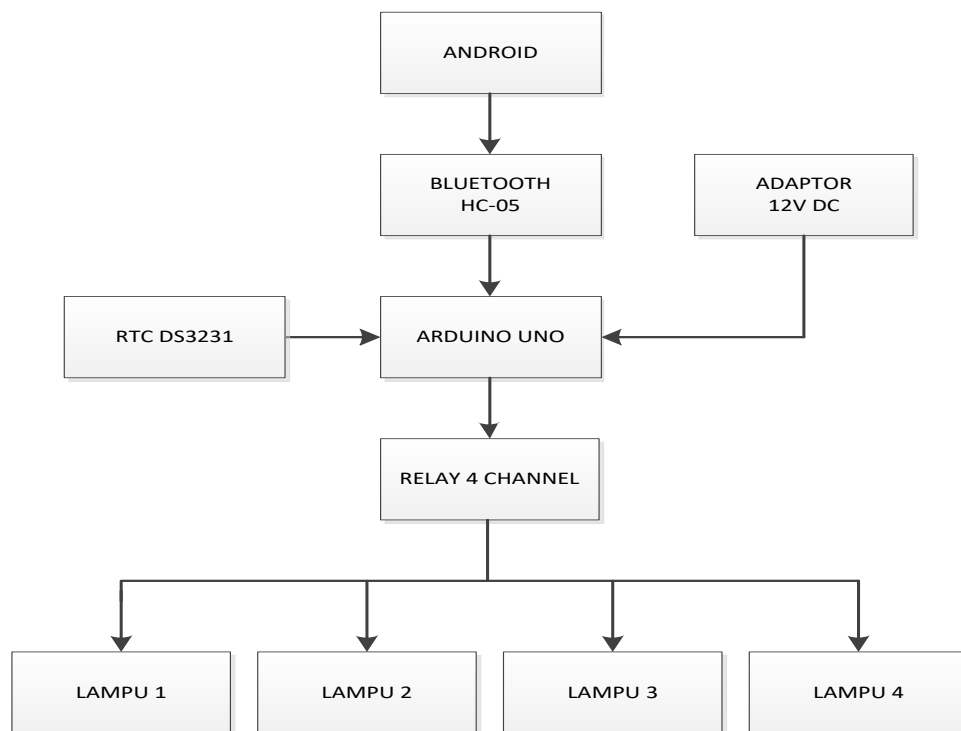
**3.1. Kebutuhan Sistem**

Komponen utama *hardware* yang digunakan adalah arduino uno, bluetooth HC-05, sensor RTC DS3231, relay, terminal block dan adaptor. kemudian software yang digunakan untuk memprogram hardware adalah Arduino IDE, sedangkan untuk memprogram aplikasi adalah App Inventor. Pada *prototype* ini Android digunakan sebagai remote pengontrol, pengatur waktu otomatis, dan sebagai monitor untuk mengetahui status aktif/mati saklar. Arduino uno digunakan sebagai kontroller untuk mengatur perintah sesuai dengan data yang diterima dari android. Sensor RTC DS3231 digunakan sebagai pewaktuan pada arduino. *Bluetooth* HC-05 digunakan sebagai jaringan komunikasi antara android dan arduino, Relay digunakan sebagai output dari perintah yang diberikan arduino, dan terhubung dengan lampu AC, terakhir Adaptor digunakan sebagai catu daya untuk menyalakan *Prototype*.

**3.2 Rancangan Sistem**

1) Arsitektur Sistem

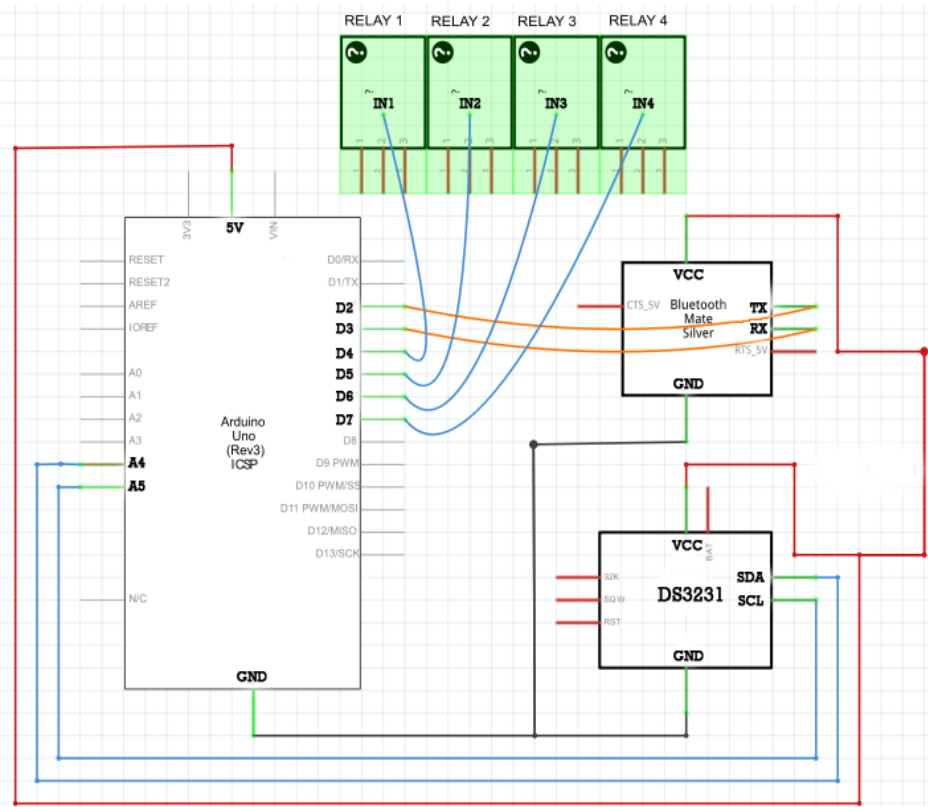
Secara umum Arsitektur sistem yang dibuat pada penelitian ini dapat dilihat pada diagram blok berikut:



Gambar 1 Arsitektur Sistem Saklar Otomatis

Pada blok diagram gambar 1 dapat dilihat proses pengendalian sistem saklarnya, dimulai dari android mengirimkan data ke arduino melalui komunikasi *bluetooth* kemudian arduino memproses data dan memberikan perintah untuk menyalakan ataupun mematikan relay 1, relay 2, relay 3, dan relay 4 sesuai dengan perintah data yang dikirim melalui android tadi, selanjutnya untuk pengendalian saklar berbasis waktu, pertama data jam dan saklar di atur melalui *smartphone*, dan data pengaturan tersebut dikirimkan ke arduino melalui bluetooth, setelah data diterima selanjutnya data pengaturan jam dan saklar tadi disimpan pada EEPROM yang ada pada arduino, jika data jam yang ada pada sensor RTC DS3231 syaratnya sudah sesuai dengan data pengaturan jam dan saklar yang tersimpan pada EEPROM maka saklar pun akan menyala atau mati sesuai dengan kondisi yang telah diatur.

## 2) Rancangan Skematik Sistem



Gambar 2 Rancangan Skematik Sistem

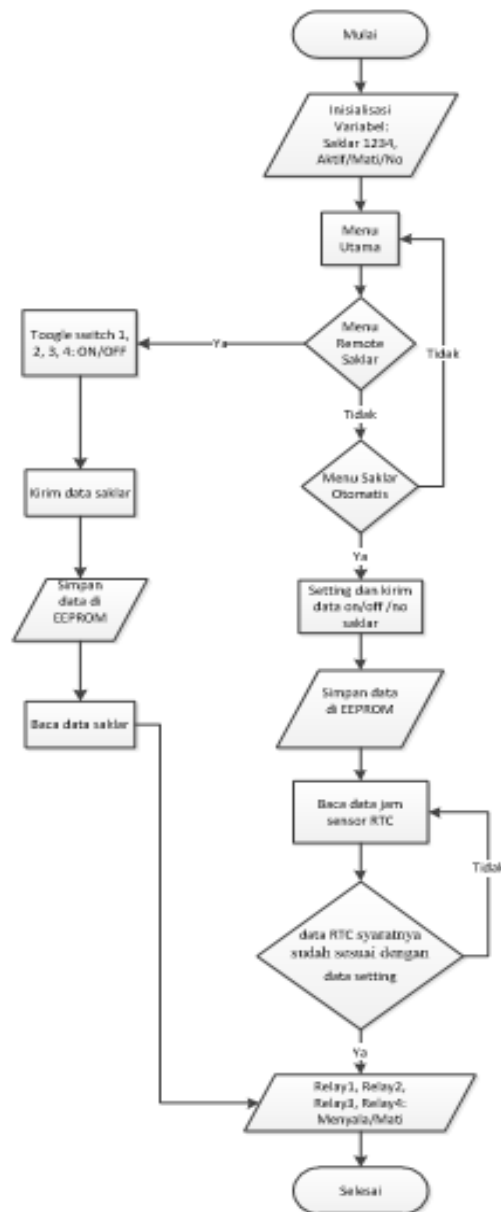
Pada desain skematik rancangan dapat dilihat jalur-jalur kabel untuk menghubungkan antar pin/jalur kabel kelistrikan komponen yang ada. Untuk lebih jelasnya hubungan antar pin/jalur kabel kelistrikan pada hardware dapat dilihat pada tabel berikut

Tabel 1. Data Pin Komponen

No	Input Pin Arduino	Output Pin
1	Pin Digital 2	Pin Tx Bluetooth
2	Pin Digital 3	Pin Rx Bluetooth
3	Pin Digital 4	In 1 Relay
4	Pin Digital 5	In 2 Relay
5	Pin Digital 6	In 3 Relay
6	Pin Digital 7	In 4 Relay
7	Pin Analog A4	SDA RTC DS3231
8	Pin Analog A5	SCL RTC DS3231
9	Pin 5V	Pin VCC Bluetooth
10	Pin 5V	Pin VCC Relay
11	Pin 5V	Pin VCC RTC DS3231
12	Pin Ground	Pin Ground Bluetooth
13	Pin Ground	Pin Ground Relay
14	Pin Ground	Pin Ground RTC DS3231

Pada tabel 1 diatas pin digital 2, dan 3 pada arduino dihubungkan dengan pin tx dan rx yang ada pada bluetooth HC-05, kemudian pin digital 4,5,6,7 pada arduino dihubungkan dengan pin In 1,2,3,4 relay, selanjutnya pin analog A4,A5 arduino dihubungkan dengan pin SDA, SCL pada RTC DS3231, terakhir untuk tegangannya pin 5v dan ground pada arduino dihubungkan atau dirangkai secara paralel kemasing-masing pin vcc dan ground komponen lain dalam hal ini komponen tersebut ialah bluetooth HC-05, RTC DS3231, dan relay.

3) Model Logik Sistem

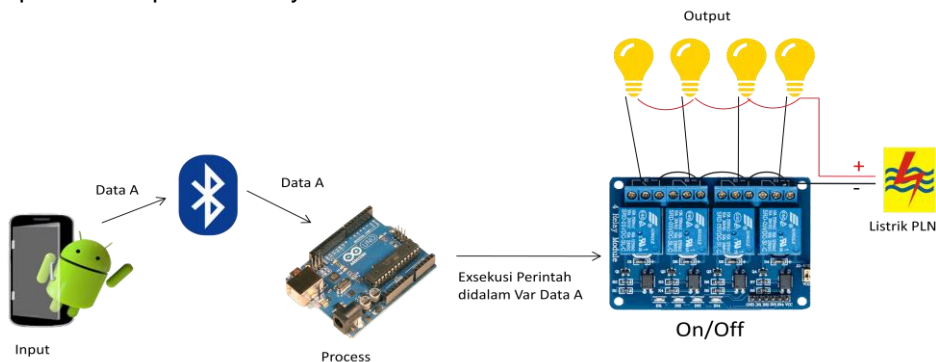


Gambar 3 Model Logik Sistem Saklar Otomatis

Pada diagram alir gambar 3, di mulai dengan menginisialisasi variabel yaitu saklar 1,2,3,4 aktif, mati, no, variabel ini fungsinya sebagai pengenalan agar arduino dapat membaca perintah yang dikirimkan oleh android, setelah variabel di inialisasi selanjutnya masuk kemenu utama apabila user memilih menu remote saklar maka akan muncul 4 buah *switch*, jika *switch* tadi ditekan maka android akan mengirimkan data ke arduino, selanjutnya arduino menyimpan data tersebut di EEPROM dan membaca data, lalu mengeksekusi program sesuai dengan data yang diterima dari android dan menghasilkan *output* aktif atau mati pada relay. Selanjutnya apabila user memilih menu saklar otomatis maka akan muncul interface untuk melakukan pengaturan saklar aktif atau mati secara otomatis, setelah data diatur kemudian data tersebut dikirim dari android ke arduino, dan arduino menyimpan data di EEPROM, selanjutnya arduino akan melakukan perbandingan jika data pengaturan waktu yang ada pada RTC syaratnya sudah terpenuhi dengan data pengaturan waktu yang tersimpan pada EEPROM maka relay akan aktif atau mati secara otomatis tetapi jika tidak maka arduino akan kembali melakukan perbandingan, secara berulang-ulang sampai syarat kondisinya terpenuhi.

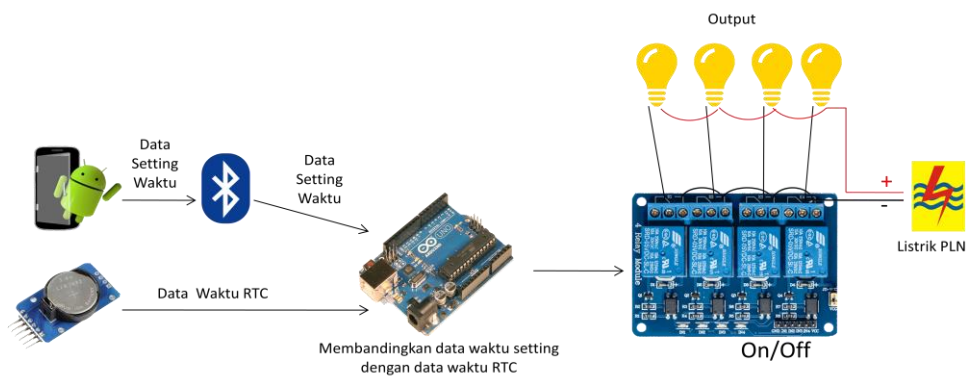
#### 4) Mekanisme Kerja Sistem

Pada mekanisme sistem akan dijelaskan alur kerja yang terjadi didalam sistem. Berisikan tentang alur kerja hubungan antara sistem kendali mikrokontroler, android, sensor dan komponen-komponen lainnya.



Gambar 4. Mekanisme Sistem Remote Saklar

Pada gambar 4, android melakukan pengiriman data variabel ke arduino melalui bluetooth HC-05 kemudian data variabel tersebut dibaca dan diproses oleh arduino, selanjutnya arduino akan menjalankan perintah untuk menyalakan atau mematikan relay,1,2,3 atau 4 sesuai dengan isi perintah yang telah diprogram pada arduino.

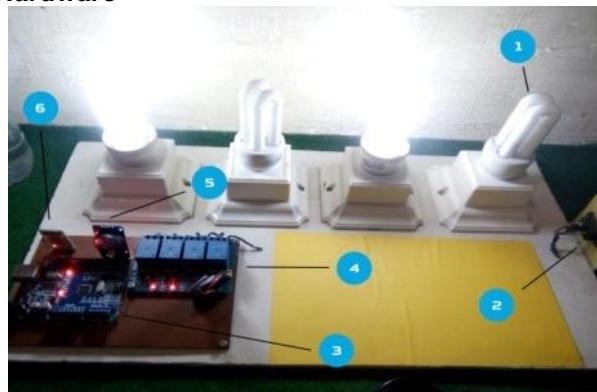


Gambar 5. Mekanisme Sistem Saklar Otomatis

Pada gambar 5, android melakukan pengiriman data pengaturan waktu ke arduino melalui *bluetooth* HC-05 kemudian arduino membandingkan data tersebut dengan data waktu yang diambil dari sensor RTC DS3231, apabila datanya bernilai benar maka arduino akan menjalankan perintah menyalakan atau mematikan relay yang terhubung dengan lampu.

## 4. Hasil dan Pembahasan

### 4.1 Tampilan Model *Hardware*



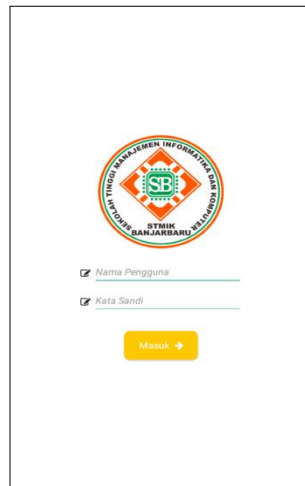
Gambar 7 *Hardware Prototype* Saklar Otomatis

Keterangan:

1. Lampu AC ukuran 5 watt
2. Jack AC sebagai output penghubung kelistrikan
3. Mikrokontroler arduino sebagai pengendali utama alat
4. Relay 4 channel sebagai penghubung dan pemutus arus listrik ke lampu
5. Sensor RTC sebagai jam atau waktu pada hardware
6. Bluetooth HC-05 sebagai penghubung komunikasi antara android dan hardware

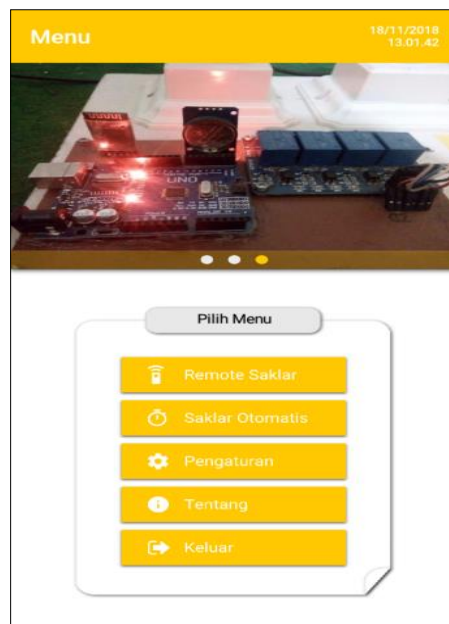
**4.2 Tampilan Model Software**

Antarmuka sistem kendali saklar menggunakan teknologi berbasis mobile (sistem android). Teknologi berbasis mobile memungkinkan efektivitas pergerakan dan jangkauan user [15], sehingga juga ramah terhadap user penyandang disabilitas.



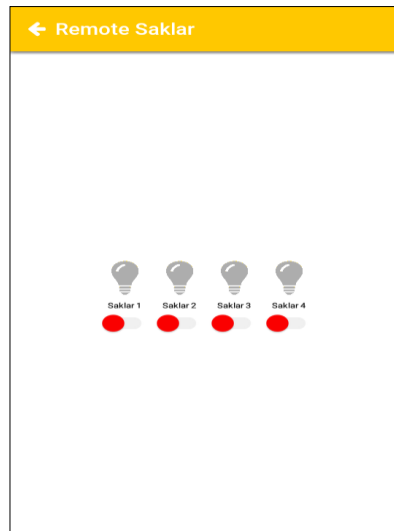
Gambar 8 Interface Login

Interface login diakses pertama kali ketika aplikasi dibuka, Interface ini berfungsi sebagai pengamanan agar perangkat hardware tidak bisa diakses semua orang, untuk dapat mengakses perangkat hardware diperlukan inputan nama pengguna dan sandi yang sesuai kemudian tekan tombol login untuk masuk ke menu utama, jika nama pengguna dan sandi yang dimasukan sudah sesuai maka interface pada aplikasi akan beralih ke interface menu utama.



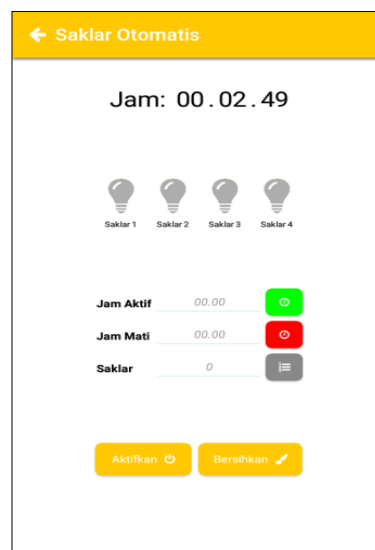
Gambar 9. Interface Menu Utama

Pada *interface* menu terdapat lima tombol yaitu *Remote* Saklar, Saklar Otomatis, Pengaturan, Tentang, dan Keluar jika tombol *remote* saklar ditekan maka akan beralih ke *interface* remote saklar, jika tombol saklar otomatis ditekan maka akan beralih ke *interface* saklar otomatis, jika tombol pengaturan ditekan maka akan beralih ke *interface* pengaturan, jika tombol tentang ditekan maka akan beralih ke *interface* tentang, jika tombol keluar atau tombol *backpressed* pada perangkat android ditekan maka akan muncul konfirmasi “apakah anda ingin keluar dari aplikasi?”, jika ditekan ya maka aplikasi akan keluar, jika ditekan tidak maka aplikasi akan tetap berada di *interface* menu.



Gambar 10. *Interface Remote Saklar*

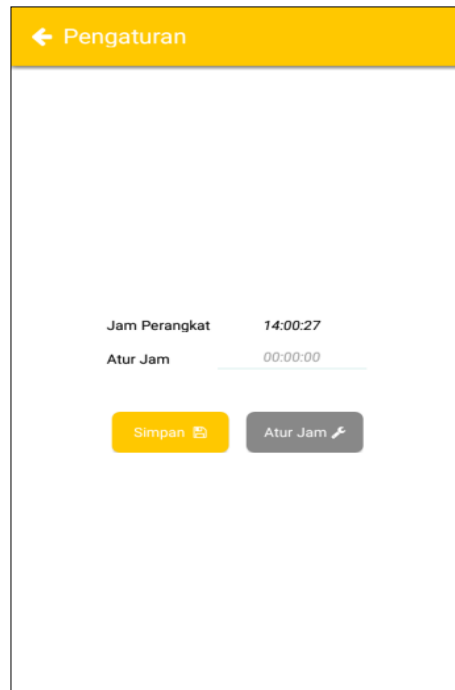
*Interface remote* saklar digunakan untuk melakukan pengontrolan saklar secara manual, pada *interface* ini terdapat 4 buah *switch* dan 4 buah icon lampu, yang mana 4 buah *switch* ini posisi awalnya dalam keadaan mati, berwarna merah, sedangkan untuk icon lampu awalnya berwarna abu-abu. *Switch* ini berfungsi untuk melakukan pengontrolan saklar baik itu aktif ataupun mati, jika *switch* dalam keadaan mati ditekan maka *switch* akan berubah warna menjadi hijau, icon lampu berubah warna menjadi kuning, dan aplikasi akan mengirim data *string* ke perangkat *hardware*, kemudian lampu atau saklar pada perangkat *hardware* akan menyala. Sebaliknya jika *switch* dalam keadaan aktif ditekan maka *switch* akan berubah warna menjadi merah, icon lampu berubah warna menjadi abu-abu, dan aplikasi akan mengirim data *string* ke perangkat *hardware*, kemudian lampu atau saklar pada perangkat *hardware* akan mati.



Gambar 11 *Interface Saklar Otomatis*



*Interface* saklar otomatis digunakan untuk melakukan pengontrolan saklar secara otomatis dengan *timer*, pada interface ini terdapat label text untuk menampilkan jam dari sensor RTC DS3231, kemudian terdapat 4 icon lampu berwarna abu-abu yang dapat berubah warna menjadi kuning digunakan sebagai indikator untuk mengetahui saklar mana yang sedang menyala dan saklar mana yang sedang mati, terdapat 2 buah *timepicker* untuk mengatur jam aktif saklar dan jam mati saklar, terdapat 1 buah *combobox* untuk memilih saklar mana yang akan diotomatiskan, terdapat 3 buah *textbox* 2 diantaranya untuk menampilkan data pengaturan jam saklar dari *timepicker* dan 1 untuk menampilkan data pengaturan nomor saklar dari *combobox*, terdapat juga tombol untuk mengaktifkan/mematikan mode otomatis, dan tombol bersihkan untuk menghapus inputan data saklar yang telah diatur pada aplikasi.



Gambar 12 *Interface* Pengaturan

*Interface* pengaturan digunakan untuk melakukan pengaturan terhadap waktu atau jam yang ada pada perangkat *hardware*, didalam interface ini terdapat 2 buah label *text* yaitu label *text* jam perangkat dan label *text* atur jam, juga terdapat 2 buah *textbox*, 1 digunakan untuk menampilkan data jam pada *hardware*, dan 1 nya digunakan untuk menampilkan hasil pengaturan data jam, selain itu terdapat juga 2 buah tombol, yaitu tombol atur jam yang mana bila tombol ini ditekan maka akan memanggil sebuah *timepicker*, kemudian tombol yang kedua yaitu tombol simpan digunakan untuk melakukan penyimpanan pengaturan jam pada *hardware*.

### 4.3 Pengujian Sistem

#### 1) Pengujian *Blackbox* Komunikasi Perangkat

Untuk melakukan pengujian saklar aktif dan mati secara otomatis maka pada sistem diberikan beberapa kondisi yaitu:

A = Jam Hidup < Jam Mati

B = Jam Hidup > Jam Mati

X = ((Jam Sekarang >= Jam Hidup) && (Jam Sekarang < Jam Mati))

Y = ((Jam Sekarang <= Jam Hidup) && (Jam Sekarang < Jam Mati))

|| (Jam Sekarang >= Jam Hidup))

Keterangan:

A = Kondisi A

B = Kondisi B

X = Kondisi saklar diaktifkan pada kondisi A

Y = Kondisi saklar diaktifkan pada kondisi B

Tabel 2. Hasil Pengujian *Blackbox* Kondisi A saklar mati

No	Objek	Inputan waktu	Parameter			Perhitungan	Hasil Total Detik	Kondisi	Status saklar
			Jam (hh)	Menit (mm)	Detik (ss)				
1	Jam Hidup	18:00:00	18	0	0	$(18 \times 3600) + 0 + 0$	64800	A	Mati
2	Jam Mati	19:00:00	19	0	0	$(19 \times 3600) + 0 + 0$	68400		
3	Jam Sekarang	17:00:00	17	0	0	$(17 \times 3600) + 0 + 0$	61200		

Dari hasil pengujian diatas dapat disimpulkan berdasarkan inputan nilai jam hidup lebih kecil dari pada jam mati maka program saklar otomatis masuk kedalam kondisi A dan dikarenakan kondisi X yang ada didalam kondisi A tidak terpenuhi maka status saklar dalam keadaan mati.

Tabel 3. Hasil Pengujian *Blackbox* Kondisi A saklar hidup

No	Objek	Inputan waktu	Parameter			Perhitungan	Hasil Total Detik	Kondisi	Status saklar
			Jam (hh)	Menit (mm)	Detik (ss)				
1	Jam Hidup	18:00:00	18	0	0	$(18 \times 3600) + 0 + 0$	64800	A	Hidup
2	Jam Mati	19:00:00	19	0	0	$(19 \times 3600) + 0 + 0$	68400		
3	Jam Sekarang	18:30:00	18	30	0	$(18 \times 3600) + (30 \times 60) + 0$	66600		

Dari hasil pengujian diatas dapat disimpulkan berdasarkan inputan nilai jam hidup lebih kecil dari pada jam mati maka program saklar otomatis masuk kedalam kondisi A dan dikarenakan kondisi X yang ada didalam kondisi A terpenuhi maka status saklar dalam keadaan hidup.

Tabel 5 Hasil Pengujian *Blackbox* Kondisi B saklar mati

No	Objek	Inputan waktu	Parameter			Perhitungan	Hasil Total Detik	Kondisi	Status saklar
			Jam (hh)	Menit (mm)	Detik (ss)				
1	Jam Hidup	23:00:00	23	00	00	$(23 \times 3600) + 0 + 0$	82800	B	Mati
2	Jam Mati	01:00:00	1	0	0	$(1 \times 3600) + 0 + 0$	3600		
3	Jam Sekarang	22:00:00	22	00	0	$(22 \times 3600) + 0 + 0$	79200		

Dari hasil pengujian diatas dapat disimpulkan berdasarkan inputan nilai jam hidup lebih besar dari pada jam mati maka program saklar otomatis masuk kedalam kondisi B dan dikarenakan kondisi Y yang ada didalam kondisi B tidak terpenuhi maka status saklar dalam keadaan mati.

Tabel 6 Hasil Pengujian *Blackbox* Kondisi B saklar hidup

No	Objek	Inputan waktu	Parameter			Perhitungan	Hasil Total Detik	Kondisi	Status saklar
			Jam (hh)	Menit (mm)	Detik (ss)				
1	Jam Hidup	23:00:00	23	59	59	$(23 \times 3600) + 0 + 0$	82800	B	Hidup
2	Jam Mati	01:00:00	1	0	0	$(1 \times 3600) + 0 + 0$	3600		
3	Jam Sekarang	00:00:00	23	40	0	0	0		

Dari hasil pengujian diatas dapat jelaskan bahwa berdasarkan inputan nilai jam hidup lebih besar dari pada jam mati maka program saklar otomatis masuk kedalam kondisi B dan dikarenakan kondisi Y yang ada didalam kondisi B terpenuhi maka status saklar dalam keadaan hidup.

## 5. Kesimpulan

Setelah pengujian *Prototype* Saklar Otomatis berbasis Android dan Arduino, Berdasarkan penelitian yang dilakukan selama pengujian dapat disimpulkan:

- 1) Model yang dikembangkan dapat membantu dalam hal penyalaan saklar terutama jika jumlah saklarnya ada banyak.
- 2) Alat yang dikembangkan dapat mengontrol saklar dari jarak jauh
- 3) Dengan mengimplementasikan modul *bluetooth* HC-05 maka alat ini bisa melakukan komunikasi data dengan aplikasi android.
- 4) Dengan mengimplementasikan relay dan modul RTC DS3231 maka alat ini bisa melakukan pengontrolan saklar secara otomatis dengan waktu.

**DAFTAR REFERENSI**

- [1] Wibowo, H., Somantri, Y., dan Haritman, E. Rancang Bangun Magnetic Door Lock Menggunakan Keypad Dan Solenoid Berbasis Mikrokontroler Arduino Uno, *Jurnal Electrans*. 2013; 12: 39-48
- [2] Silvia, A. F., Haritman, E., & Mulyadi, Y. Rancang bangun akses kontrol pintu gerbang berbasis arduino dan android. *Electrans*. 2014; 13(1): 1-10.
- [3] Reswandi, J., Pangaribuan, P., & Atmaja, R. D. Rancang Bangun Prototype Kendali Pintu Gerbang Parkir Berbasis Pelat Nomor Polisi Dan Barcode Menggunakan Pengolahan Citra Digital. *eProceedings of Engineering*. 2015; 2(1): 1-9
- [4] Sari, M. W., & Hardyanto, H. Implementasi Aplikasi Monitoring Pengendalian Pintu Gerbang Rumah Menggunakan App Inventor Berbasis Android. *Jurnal Eksplorasi Karya Sistem Informasi dan Sains*. 2016; 9(1): 20-28
- [5] Rinaldy, R., Christianti, R. F., & Supriyadi, D. Pengendalian Motor Servo yang terintegrasi dengan webcam berbasis internet dan arduino. *Jurnal infotel*. 2013; 5(2); 17-23.
- [6] Birdayansyah, R., Soedjarwanto, N., & Zebua, O. Pengendalian Kecepatan Motor DC Menggunakan Perintah Suara Berbasis Mikrokontroler Arduino. *Electrician*. 2015; 9(2): 97-108.
- [7] Bahar, B., & Purwanto, A. Model Sistem Peringatan Dini Banjir Di Kecamatan Satu Menggunakan Sensor Kapasitif Aluminium Foil. *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*. 2015; 3(2): 545-552
- [8] Amelia, G. R., & Aprilianto, H. Sistem Deteksi Kebocoran Gas LPG Berbasis Mikrokontroler Atmega16. *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*. 2017; 5(1): 907-916
- [9] Fatoni, A., & Rendra, D. B. Perancangan Prototype Sistem Kendali Lampu Menggunakan Handphone Android Berbasis Arduino. *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*. 2014; 1: 23-29
- [10] Samsugi, S., Ardiansyah, A., & Kastutara. Arduino dan Modul Wifi ESP8266 sebagai Media Kendali Jarak Jauh dengan antarmuka Berbasis Android. *Jurnal Teknoinfo*. 2018; 12(1): 23-27.
- [11] Taufiq, T., & Fadlani, A. Pengendali Level Air Tandon Otomatis Menggunakan Gelombang Ultrasonik. *Progresif: Jurnal Ilmiah Komputer*. 2019; 14(1): 1-12
- [12] Liliana, & J, W. Prototype Penerangan Rumah Otomatis Berbasis Mikrokontroler ATmega8535. *Jurnal Sains, Teknologi dan Industri*. 2014; 11(2): 273-281
- [13] Sutono. Perancangan Sistem Otomatisasi Lampu Penerangan Menggunakan Sensor Gerak Dan Sensor Cahaya Berbasis Arduino Uno ATmega 328. *Majalah Ilmiah Unikom*. 2014; 12(2): 223-232
- [14] Desyantoro, E. Sistem Pengendali Peralatan Elektronik Dalam Rumah Secara Otomatis Menggunakan Sensor PIR, LM35, Dan Sensor LDR. *Jurnal Teknologi dan Sistem Komputer*. 2015; 3(3): 405-411
- [15] Bahar, Soegiarto. Development of Instructional Media Based on Mobile Technology to Enriching Teaching Material for Primary School Students in Indonesia Post-Learning in The Classrooms. *International Journal of Scientific & Technology Research*. 2020; 9(1): 94-98