

## Implementasi *Semantic Computing* pada Rumah Cerdas berdasarkan Perintah Pengguna menggunakan *Smartphone*

M Yusuf Effendi<sup>1</sup>, Mochammad Hannats Hanafi Ichsan<sup>2</sup>, Rizal Maulana<sup>3</sup>

Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>yusuf24ef@gmail.com, <sup>2</sup>hanas.hanafi@ub.ac.id, <sup>3</sup>rizal\_lana@ub.ac.id

### Abstrak

Penerapan teknologi IoT (*Internet of Things*) banyak diterapkan salah satunya pada rumah cerdas. Sektor ini merupakan sektor yang paling diminati dikarenakan pendekatannya dekat dengan gaya hidup manusia namun dalam penerapannya terdapat banyak kesulitan yakni salah satunya pengguna kesulitan untuk melakukan konfigurasi dan mengintegrasikan di setiap perangkatnya. Dengan pendekatan semantic yang diterapkan pada penelitian ini dapat mempermudah proses pengenalan perangkat IoT sehingga dapat menghasilkan kemampuan untuk mendefinisikan, mengubah, dan menghasilkan pembacaan data yang lebih baik. Berdasarkan penelitian yang dilakukan, sistem dibangun menggunakan 2 buah sensor yang digunakan sebagai data masukan yaitu sensor LDR (*Light Dependent Resistor*), sensor PIR (*Pasif Infra red*) dan sebuah keluaran berupa aktuator relay yang dapat memberikan nilai 1 (*High*) atau 0 (*Low*). Penelitian ini dilengkapi dengan unit proses yaitu aplikasi *smartphone* yang memiliki fitur kontrol relasi, monitoring, dan menghapus perangkat yang terhubung setiap data yang terhubung dalam firebase. Hasil pengujian diperoleh performa dengan delay 0,999985350 detik untuk proses pengiriman data dari *device* ke aplikasi dan memiliki presentase 100% keberhasilan sistem yang diterapkan pada penelitian implementasi semantik komputing pada rumah cerdas berdasarkan perintah pengguna menggunakan *smartphone*.

**Kata kunci:** Rumah Cerdas, Semantik, Aplikasi

### Abstract

*The application of IoT (Internet of Things) technology is widely applied, one of which is in smart homes. This sector is the most popular sector because its approach is close to human lifestyle, but in its application there are many difficulties, one of which is the difficulty for users to configure and integrate on each device. The semantic approach applied in this research can simplify the process of introducing IoT devices so that it can produce the ability to define, change, and produce better data readings. Based on the research conducted, the system was built using 2 sensors used as input data, namely the LDR (Light Dependent Resistor) sensor, the PIR sensor (Passive Infra red) and an output in the form of a relay actuator that can give a value of 1 (High) or 0 (Low). This research is completed with a process unit that is a smartphone application that has the features of relation control, monitoring, and erasing devices connected to every data connected in firebase. The test results obtained performance with a delay of 0.999985350 seconds for the process of sending data from device to application and has a 100% percentage of the success of the system that is applied to the research implementation of semantic computing in smart homes based on user commands using a smartphone.*

**Keywords:** Smart Home, Semantic, Application

## 1. PENDAHULUAN

*Internet of Thing* memberikan kesempatan untuk mengukur dan mengamati peristiwa yang terjadi pada lingkungan untuk mendapatkan data kontekstual, sehingga *Internet of Thing* sebagai platform masa depan yang dapat mengatasi fakta

dari permasalahan tersebut yang peka terhadap permasalahan fisik yang ada pada lingkungan tertentu. Data yang akan diolah oleh mesin terdiri dari berbagai sumber untuk mendapatkan data yang sesuai. mesin akan melakukan penalaran yang membutuhkan pendekatan secara umum untuk melakukan olah data

tersebut sehingga terbentuk data yang tersusun sesuai dengan kondisi.

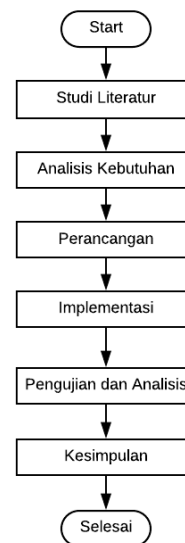
Banyak objek rumah cerdas pada masa mendatang akan terkoneksi dengan jaringan internet ataupun dengan jaringan lokal agar dapat saling bertukar informasi antara objek satu dengan objek lainnya, banyak contoh perangkat rumah cerdas saat ini yang dapat diimplementasikan yakni mengontrol lampu dari intensitas cahaya yang di dapatkan (smarthomeenergy.com, 2013). Dalam penerapannya rumah cerdas pada masa yang akan datang akan di tanamkan perangkat tambahan yakni sensor dan aktuator yang digabungkan pada perangkat atau benda yang akan di control atau monitoring dengan platform lainnya, sehingga setiap perangkat dapat di kendalikan dari jarak jauh (Xu Li, 2011). Masalah yang akan muncul untuk membuat sistem rumah cerdas sendiri adalah dari segi konfigurasi dan konektifitas.

Pada penelitian yang telah dilakukan sebelumnya oleh Agung Prasetyo yang memiliki judul “Implementasi Sistem Semantik Pada Perangkat Lampu Rumah Cerdas Berbasis Agen” kelebihan pada system yang di buat yakni didapatkan hasil yang baik dari segi kecepatan dan data yang di terima sesuai dengan apa yang dikirim. Namun pada penelitian ini memiliki kekurangan yakni penelitian dilakukan tanpa platform yang bertugas sebagai pengolah data yakni database untuk menerima data dan diolah untuk membuat sebuah relasi dengan perintah pengguna menggunakan aplikasi android. Dalam kata lain pengguna tidak memiliki kebebasan untuk memilih perangkat yang akan di hubungkan sehingga perangkat dapat saling melakukan tukar informasi antara satu dengan yang lainnya.

Dari latar belakang yang telah dipaparkan peneliti pada akhirnya melakukan penelitian yang terkait dengan konsep semantic yang di terapkan pada *Smart home* dengan cara yang berbeda yakni “Implementasi *Semantic Computing* pada Rumah Cerdas Berdasarkan Perintah Pengguna Menggunakan *Smartphone*”.

## 2. METODOLOGI PENELITIAN

Metode penelitian akan di tampilkan dalam sebuah diagram pada gambar 1, Gambar 1 merupakan diagram menunjukan langkah - langkah yang akan di lakukan ketika penelitian dilaksanakan.



Gambar 1 Diagram alir metodologi penelitian

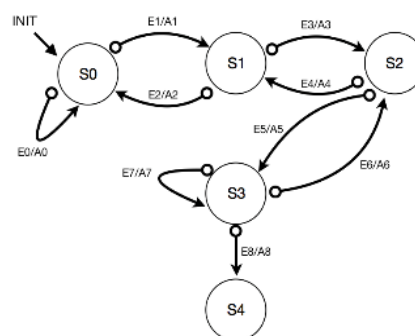
Pertama yaitu menentukan studi literatur yang sesuai dengan penelitian yang akan dilakukan, yang kedua menentukan kebutuhan apa saja yang akan digunakan untuk melakukan penelitian, selanjutnya perancangan dikerjakan untuk mengetahui sistem yang di buat akan terbentuk seperti apa. Pengujian dan analisis, analisis dikerjakan untuk menentukan pengujian yang akan dilakukan akan menjadi seperti apa. Selanjutnya kesimpulan yakni menjawab dari rumusan masalah yang sudah di paparkan.

## 3. PERANCANGAN DAN IMPLEMENTASI

### 3.1 Perancangan Perangkat Lunak

#### 1. State Machine Perangkat Node Sensor

Sistem pada perangkat *node* sensor dijelaskan dengan FSM. FSM sendiri merupakan metode yang digunakan untuk menentukan kinerja sistem atau dengan kata lain gambaran umum dari perangkat dimana pada penerapannya yaitu pada perangkat sensor.



Gambar 2 State machine pada node sensor

Pada gambar 2 merupakan implementasi dari penerapan FSM pada node sensor yang terdiri dari 5 state, 9 event, 9 Action.

Tabel 1 keterangan state machine pada node sensor

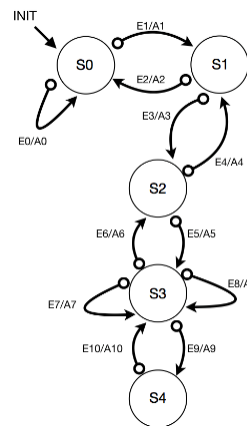
State	Event	Action
S0 Wifi Connected	E0 Wifi Reconnect	A0 Menghubungkan kembali sistem dengan wifi
S1 Firebase Connected	E1 Firebase Connect	A1 Menghubungkan sistem dengan Firebase
S2 Device Ready	E2 Firebase Disconnect	A2 Sistem gagal terhubung dengan Firebase
S3 Device Option	E3 Objek Initialization	A3 Mikrikontroler melakukan inisialisasi file metadata
S4 Off	E4 Objek failed to Initialization	A4 Mikrikontroler gagal melakukan inisialisasi file metadata
	E5 Device Recognition	A5 Mikrikontroler mengirimkan metadata ke Firebase
	E6 Device Filed to Recognition	A6 Mikrikontroler menerima kembalian dari Firebase
	E7 Device Action	A7 Mikrikontroler terhubung dan mengirimkan data ke node sensor
	E8 Device Off	A8 Kesalahan pada sistem

Pada Tabel 1 terdapat state, event, Action. Keseluruhan proses yang akan dilakukan oleh perangkat sensor yakni melewati state 0 menuju ke state1 yakni menghubungkan sensor dengan koneksi internet (wifi) ketika proses konektifitas selesai selanjutnya akan menuju ke state selanjutnya dan apabila proses itu gagal maka akan melakukan looping secara terus menerus hingga kondisi terpenuhi yaitu terhubung dengan wifi, setelah terhubung dengan wifi maka system pada perangkat akan secara otomatis akan melakukan pengiriman data awal atau mendaftarkan perangkatnya akan tetapi ketika ketika system tidak dapat terhubung dengan firebase maka akan terjadi looping terus menerus hingga dapat terhubung ke firebase.

Kodisi selanjutnya yaitu perangkat terhubung dengan firebase dan dapat mengirimkan data yang diakuisisi dari perangkat sensor. Selanjutnya yakni state 2 yang mana perangkat menunggu hingga dapat direlaskan atau di hubungkan dengan perangkat aktuator, setelah menunggu dan melakukan looping terus menerus selanjutnya menuju state 3 yakni perangkat dapat direlaskan dengan perangkat node aktuator yang sudah terhubung dengan firebase. Selanjutnya yaitu state 4 yaitu proses selesai dilakukan dan perangkat pada kondisi off atau mati. Proses tersebut sama halnya dilakukan oleh semua perangkat sensor yang digunakan pada penelitian ini.

## 2. State Machine pada Perangkat Node Aktuator

Dilakukan proses yang sama dengan perangkat sensor yakni menggunakan metode FSM untuk mengetahui perilaku atau system kerja secara umum dari perangkat aktuator.



Gambar 3 State Machine perangkat actuator

Pada gambar 3 yaitu metode state machine yang digunakan memiliki 5 State, 11 Event, 11 Action. Penjelasan dari gambar 3 dapat dilihat pada tabel 2.

Tabel 2 Keterangan State machine perangkat aktuator

State	Event	Action
S0 Wifi Connected	E0 Wifi Reconnect	A0 Menghubungkan kembali sistem dengan wifi
S1 Firebase Connected	E1 Firebase Connect	A1 Menghubungkan sistem dengan Firebase
S2 Device Ready	E2 Firebase Disconnect	A2 Sistem gagal terhubung dengan Firebase
S3 Device Option	E3 Objek Initialization	A3 Mikrikontroler melakukan inisialisasi file metadata
S4 Relation Connected	E4 Objek failed to Initialization	A4 Mikrikontroler gagal melakukan inisialisasi file metadata
	E5 Device Reconnect on	A5 Mikrikontroler mengirimkan metadata ke Firebase
	E6 Device Faild Reconnect on	A6 Mikrikontroler menerima kembalian dari Firebase
	E7 Device Action	A7 Mikrikontroler terhubung dan mengirimkan data memberi aksi terhadap aktuator lampu
	E8 Keep Alive	A8 Ikrokontroler Memeriksa ketersediaan perangkat yang telah terintegrasi
	E9 Aktuator Integration	A9 Mikrikontroler memriksa perangkat sensor yang akan diintegrasikan
	E10 Aktuator Failed to Integration	A10 Mikrikontroler menerima pesan berupa sesor yang terhubung

Pada Tabel 2, dijelaskan terdapat beberapa state, event dan action semua sistem pada state machine perangkat aktuator akan di mulai dari state 0 yaitu perangkat melakukan proses

terhubung dengan jaringan internet (*wifi*), ketika perangkat tidak dapat terhubung dengan internet sistem akan selalu melakukan *looping* untuk terhubung dengan jaringan internet. Setelah terhubung dengan jaringan internet sistem akan lanjut ke *state* 1 yakni menghubungkan perangkat dengan firebase untuk pendaftaran perangkat aktuator. Ketika kondisi dari *state* 1 tidak terpenuhi maka sistem akan melakukan proses *looping* secara terus-menerus hingga perangkat dapat terhubung dengan firebase.

Setelah terhubung dengan firebase, perangkat *node* aktuator sudah siap untuk di hubungkan dengan perangkat sensor yakni di tunjukan pada *state* 2. *Sate* 3 yakni perangkat sudah berhasil di hubungkan dengan perangkat lainnya sehingga dapat saling bertukar informasi. Perangkat aktuator akan menerima data yang diakuisisi oleh perangkat *node* sensor sehingga dapat diajadikan acuan apakah aktuator tersebut akan bernilai 1 atau 0. *State* 4 perangkat off proses pertukaran data selesai.

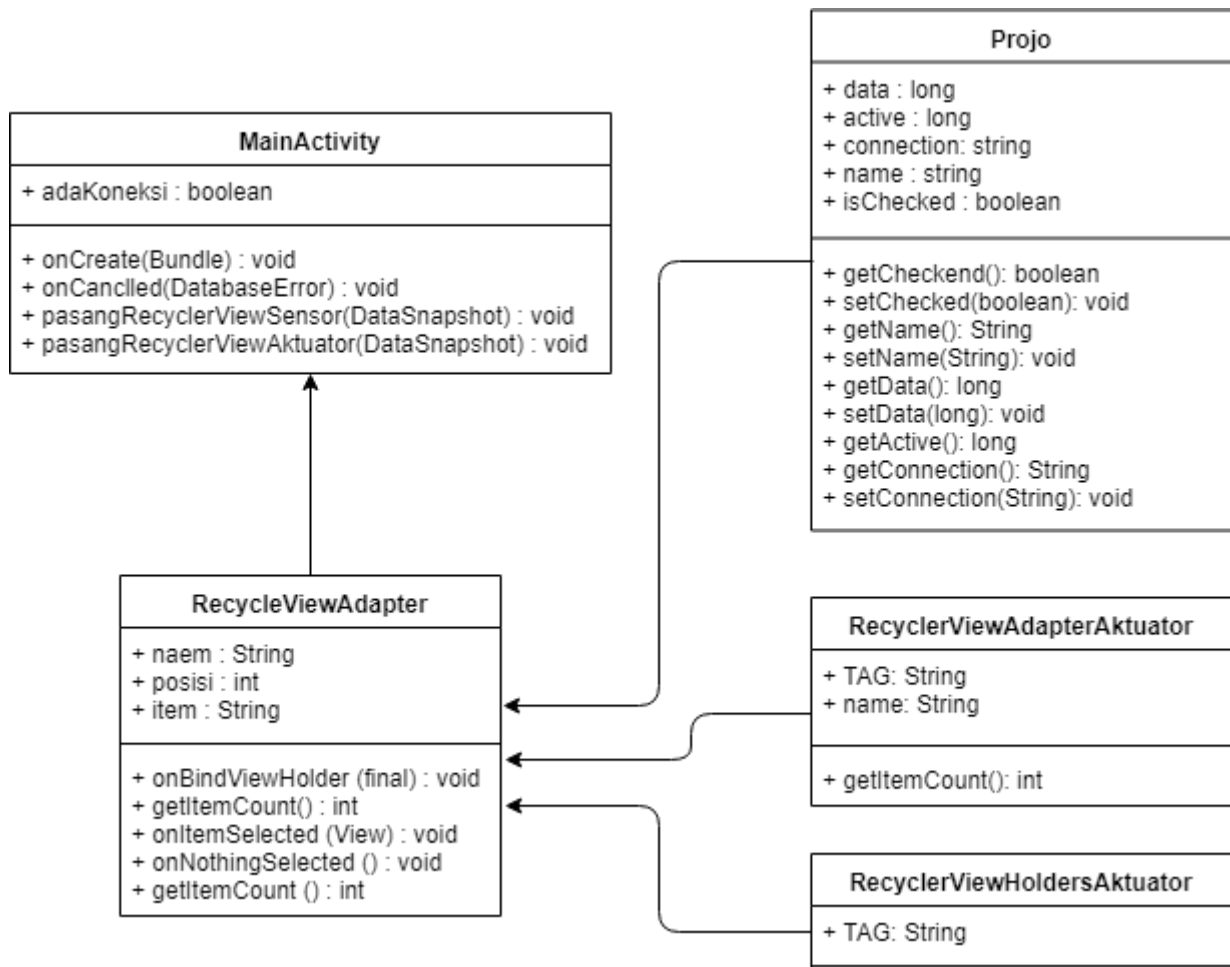
### 3.2 Class Diagram Aplikasi Smartphone

Pada gambar 5 memiliki 5 *calss* yakni *Class Activity*. Diagram MainActivity mempunyai 1 turunan yaitu *class* RecyclerViewAdapter sedangkan 4 *class* yang lainya turunan dari *class* RecyclerViewAdapter yaitu RecyclerViewAdapterAktuator, *projo*, dan RecyclerViewHoldersAktuator.

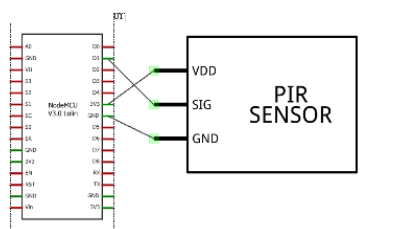
### 3.3 Perancangan Perangkat Keras

#### 1. Perancangan Perangkat Keras Node Sensor PIR

Perancangan perangkat keras pada sensor pir dilakukan dengan membuat skematik sensor pir yang dihubungkan dengan mikrokontroler nodemcu. Sensor pir terdiri dari 3 pin yakni vdd, sig, dan gnd yang masing-masing dihubungkan ke pin mikrokontroler.



Gambar 5 Class diagram aplikasi smartphone

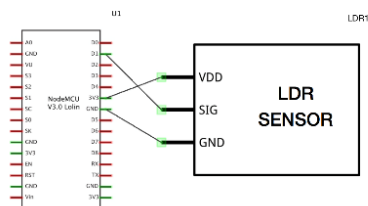


Gambar 4 Konfigurasi pin sensor PIR dengan mikrokontroler nodeMCU

pada Gambar 4 komponen dihubungkan menggunakan kabel jumper sesuai dengan pin sensor.

### 2. Perancangan Perangkat Keras Node Sensor LDR

Perancangan perangkat keras pada *node* sensor LDR memiliki konfigurasi pin yang sama yakni 3 pin yang terdiri dari VDD, SIG, dan GND yang di hubungkan dengan pin mikrokontroler nodemcu.

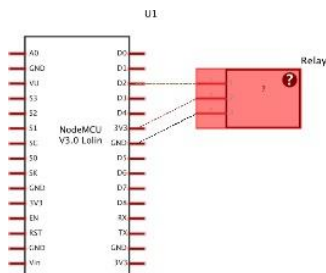


Gambar 6 Konfigurasi pin sensor PIR dengan mikrokontroler nodeMCU

pada Gambar 6 masing-masing pin dihubungkan menggunakan kabel jumper.

### 3. Perancangan Perangkat Keras Node Aktuator Lampu

Perancangan pada perangkat keras *node* aktuator dilakuka dengan membuat skematik. Sesuai dengan gambar 7 perangkat *node* aktuator terdiri dari aktuator relay yang dilengkapi dengan 3 pin yakni VCC, IN, serta pin GND yang masing – masing pin dihubungkan dengan pin mikrokontroler nodemcu.



Gambar 7 Konfigurasi pin aktuator relay dengan mikrokontroler nodeMCU

### 3.4 Implementasi Perangkat Lunak

#### 1. Implementasi State Machine pada Node Sensor

Dari perancangan yang dikerjakan yakni menggunakan metode *state machine* pada *node* sensor, melakukan persiapan untuk dapat terhubung ke jaringan internet dan terhubung ke firebase ditunjukkan pada potongan program tabel 3.

Tabel 3 Library Wifi dan Firebase

No	Kode Program
1	#include <ESP8266WiFi.h>
2	#include <DNSServer.h>
3	#include <ESP8266WebServer.h>
4	#include <WiFiManager.h>
5	#include <FirebaseArduino.h>

Setelah perangkat terhubung dengan jaringan internet dan terhubung dengan firebase selanjutnya perangkat melakukan proses menunggu untuk dapat di relasikan dengan perangkat aktuator, yang ditunjukkan pada potongan program tabel 4.

Tabel 4 Membuat Objek Json dan menunggu relasi

NO	Kode Program
1	JsonObject Aktuator =
2	Firebase.get("Aktuator");
3	JsonVariant variant =
4	Aktuator.getJsonVariant();
5	for( const auto& kv :
6	variant.as<JsonObject>()
7	) {
8	Serial.println(kv.key);
9	String nama_aktuator = kv.key;
10	}
11	if(Firebase.getString("Aktuator
12	/"+
13	nama_aktuator+"/connection") ==
14	"LDR")

#### 2. Implementasi State Machine pada Node Aktuator

Implementasi *state machine* pada *node* aktuator memiliki *library* yang sama dengan *library* yang digunakan pada perangkat *node* sensor yang di jelaskan pada tabel 3.

Tabel 5 Mengambil Data dari Firebase

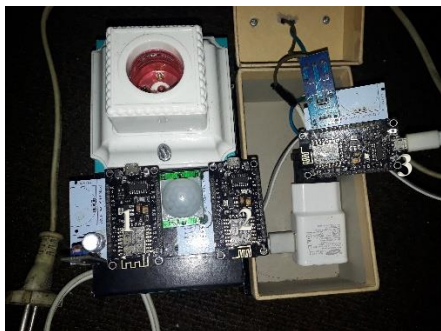
NO	Kode Program
1.	int data
2.	=Firebase.getInt("Aktuator/
3.	Relay1/data");

Tabel 5 merupakan potongan program mengambil data yang ada pada firebase. Perangkat akan selalu melakukan *looping* untuk

mendapatkan update data dari perangkat yang di relasikan dengan perangkat aktuator *relay*.

### 3.5 Implementasi Perangkat Keras

Implementasi pada perangkat keras di jelaskan pada gambar 8 dibawah ini :



Gambar 8 Implementasi Perangkat *Node* sensor dan *Node* Aktuator

1. Implementasi perangkat keras *node* sensor LDR yang di hubungkan dengan mikrokontroler nodemcu.
2. Implementasi perangkat keras *node* sensor PIR yang dilengkapi dengan mikrokontroler nodemcu
3. Implementasi perangkat keras *node* aktuator *relay*.

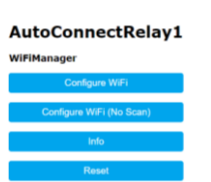
## 4. PENGUJIAN

Pengujian dilakukan menggunakan dua metode yakni *White-Box* dan *Black-Box Testing* untuk menguji perangkat maupun aplikasi dalam penelitian ini.

### 4.1 Pengujian *Black-Box Testing*

Tujuan dari pengujian *black-box testing* adalah untuk mengetahui sistem berjalan sesuai dengan tujuan pengujian yang sudah ditentukan.

1. Pengujian *node* aktuator



Gambar 9 web menghubungkan perangkat dengan wifi

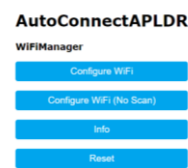
```
*WM: Connecting as wifi client...
*WM: Already connected. Bailing out.
*WM: IP Address:
*WM: 192.168.1.86
connected...yeey :)
connect = 0:1:2
durasi awal =
*WM: freeing allocated params!
1
Durasi = 0:1:5
```

Gambar 10 Hasil Perangkat Aktuator terhubung *wifi*

Tujuan pengujian pada *node* aktuator 1 berfungsi untuk mengontrol kondisi lampu yang dapat menerima nilai 0 dan 1 yakni kondisi mati dan nyala, selanjutnya yaitu perangkat dapat terkoneksi dengan internet (*wifi*) melalui web yang diakses melalui ip perangkat.

### 2. Pengujian *node* sensor

Tujuan pengujian sensor berfungsi untuk akuisisi data dari setiap kondisi yang di dapatkan oleh sensor yang digunakan yaitu sensor PIR dan LDR. Pengujian dilakuka yang peratama yaitu menghubungkan mikrokontroler dengan catu daya yakni 5v, selanjutnya menghubungkan perangkat dengan *wifi* melalui web yang diakses dari ip perangkat.



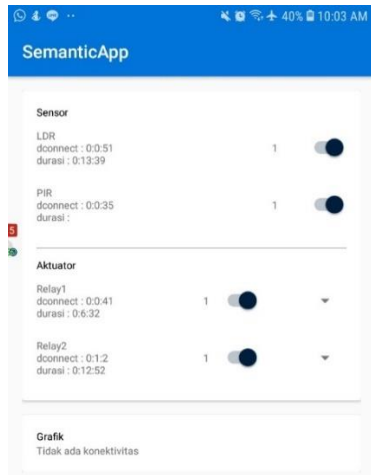
Gambar 11 web menghubungkan Perangkat Sensor dengan *Wifi*

```
*RM: Connection result:
*RM: 3
*RM:
*RM: AutoConnect
*RM: Connecting as wifi client...
*RM: Already connected. Bailing out.
*RM: IP Address:
*RM: 192.168.1.84
 Autoscroll  Show timestamp
```

Gambar 12 Hasil Perangkat Sensor Terhubung *wifi*

### 3. Pengujian aplikasi

Tujuan pengujian pada aplikasi berfungsi sebagai *monitoring* dan *controlling* semua perangkat yang terhubung dengan firebase. Langkah dalam melakukan pengujian pada aplikasi, pertama *monitoring* semua perangkat yang terhung kemudian lakukan kontrol pada setiap perangkat yang terhubung.



Gambar 13 Hasil Semua Perangkat yang terhubung ke Aplikasi

### 4.2 Pengujian White-Box Testing

Pengujian menggunakan metode *white-box testing* yang dilakukan untuk menguji sebuah sistem yang diimplementasikan pada sebuah source code (algoritma). Pengujian *white-box testing* dijalankan sesuai dengan prosedur yang dijalankan.

1. Prosedur Pengujian
  1. Membuat *pseudocode* untuk mengetahui sistem kerja pada algoritma yang akan di buat.
  2. Membuat *basis path* diagram sesuai dengan *pseudocode* sebelumnya.
  3. Menghitung *edge* dan *node* sesuai dengan rumus *Cyclomatic Complexity*.
  4. Menentukan *independent path* sesuai dengan alur yang sudah digambarkan oleh *basis path* dengan kondisi area sesuai hasil yang di dapat setelah melakukan perhitungan pada proses *cyclomatic complexity*.

#### 2. Pengujian Relay

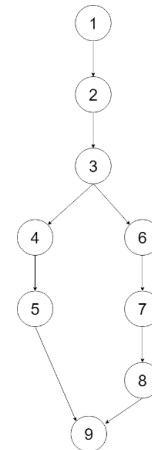
1. *Pseudocode loop* pada relay 1 dan 2

Tabel 3 *pseudocode* pada relay 1 dan 2

No	Keterangan
	Mulai
1.	Mengambil data dari firebase dengan alamat data relay 1
2.	If data samadengan 1
3.	Relay bernilai 1 (High)
4.	Tampilkan nila data yang diambil dari firebase
5.	Else
6.	Relay bernilai 0 (Low)
7.	Tampilkan nila data yang diambil dari firebase
8.	Selesai

2. *Basis path testing*

#### i. Flow diagram



Gambar 14 *Flow Graph* metod loop

#### ii. Cyclomatic complexity

Perhitungan Matematis  

$$V(G) = Edges - Nodes + 2$$

$$= 9 Edges - 9 Nodes + 2$$

$$= 2$$

$V(G) = 2$  Region R1 dan R2

#### iii. Independent path

- 1) 1-2-3-4-5-9
- 2) 1-2-3-6-7-8-9

#### 3. Hasil pengujian node aktuator

Tabel 4 Hasil pengujian metod loop pada node aktuator

Jalur	Test Case	Expected Result	Result	status
1-2-	Pengguna	<i>Device</i>	<i>Device</i>	<b>Valid</b>
3-4-	Menyalak	terhubung	dapat	
5-9	an <i>device</i>	dengan	terhubung	
	dapat	sumber	g	
	terhubung	daya dan	dengan	
	dengan	terhubung	firebase	
	firebase	dengan	mampu	
	dan	firebase	mengiri	
	<i>device</i>	serta	mkan	
	dapat	mengirim	kondisi	
	mengamb	kan	awal	
	il data	kondisi	dan	
	dari	bahwa	mengam	
	firebase	kondisi	bil data	
	untuk	awal aktiv	dari	
	menentuk	,kemudian	firebase.	
	an kondisi	<i>device</i>		
	akuator.	mengambi		
		l data dari		
		firebase		
		untuk		
		kondisi		
		dari		
		aktuator		
		selanjutny		
		a.		

4. Pengujian Sensor LDR

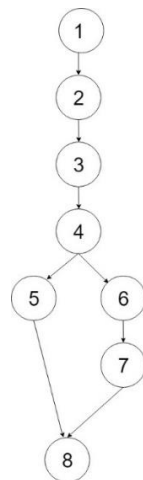
1. Pseudocode sensor LDR

Tabel 5 pseudocode pada sensor LDR

No	Keterangan
1.	Mulai
2.	Isi data
3.	Kirim data
4.	If firebase tidak bisa terhubung
5.	Tampilkan data tidak terkirim
6.	Else
7.	Data dikirim dan tampilkan Uploaded
8.	Selesai

2. Basis path testing

i. Flow diagram



Gambar 15 Flow graph metod sensor LDR

ii. Cyclomatic complexity

Perhitungan Matematis

$$\begin{aligned}
 V(G) &= Edges - Nodes + 2 \\
 &= 8 Edges - 8 Nodes + 2 \\
 &= 2
 \end{aligned}$$

$$V(G) = 2 \text{ Region R1 dan R2}$$

iii. Independent path

- 1) 1-2-3-4-5-8
- 2) 1-2-3-4-6-7-8

3. Hasil pengujian

Tabel 6 hasil pengujian metod loop sensor LDR

Jalur	Test Case	Expected Result	Result	statusn
1-2-	Pengguna	Device	Device	<b>Valid</b>
3-4-	Menyalak	dapat	dapat	
6-7-8	an device, device	terhunbu ng	terhubu ng	
	dapat mengirim	dengan firebase	dengan firebase,	
	kan data ke	dan dapat mengirim	dapat mengak	
	firebase sesuai	data ke firebase	uisisi data	
	dengan	sesuai dengan	dengan baik dan	

kondisi sensor	mengiri m data tersebut ke firebase
----------------	-------------------------------------

5. Pengujian Sensor PIR

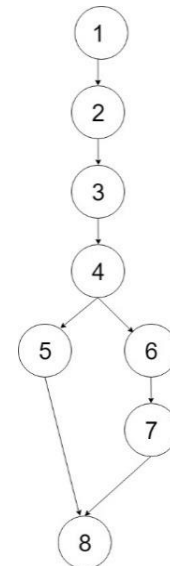
1. Pseudocode loop sensor PIR

Tabel 7 pseudocode loop sensor PIR

No	Keterangan
1.	Mulai
2.	Isi data
3.	Kirim data
4.	If firebase tidak bisa terhubung
5.	Tampilkan data tidak terkirim
6.	Else
7.	Data dikirim dan tampilkan Uploaded
8.	Selesai

2. Basis path testing

i. Flow Diagram



Gambar 16 Flow Graph metod loop sensor PIR

ii. Cyclomatic complexity

Perhitungan Matematis

$$\begin{aligned}
 V(G) &= Edges - Nodes + 2 \\
 &= 8 Edges - 8 Nodes + 2 \\
 &= 2
 \end{aligned}$$

$$V(G) = 2 \text{ Region R1 dan R2}$$

iii. Independent path

- 1) 1-2-3-4-5-8
- 2) 1-2-3-4-6-7-8

3. Hasil pengujian

Tabel 8 hasil pengujian metod loop pada sensor PIR

Jalur	Test Case	Expected Result	Result	status
1-2-	Pengguna	Device	Device	<b>Valid</b>
3-4-	Menyalak	dapat	dapat	
	an device,	terhubung	terhubung	



6-7-8	<i>device</i> dapat mengirim data ke firebase sesuai dengan kondisi sensor	dengan firebase dan dapat mengirim data ke firebase	dengan firebase, dapat mengakuis isi data dengan baik dan mengirim data tersebut ke firebase
-------	--	---	--

1-2-3-4-5-6	Bagaimana a pengguna membuat relasi dari beberapa <i>device</i> .	Sensor dapat mencari semua <i>node</i> aktuator yang terhubung dan ditampilkan.	Sensor terhubung dengan aktuator dan tampilkan data, serta sensor dapat mempengaruhi kondisi aktuator.	<b>Valid</b>
-------------	---	---	--	--------------

6. Pengujian Relasi Node Sensor

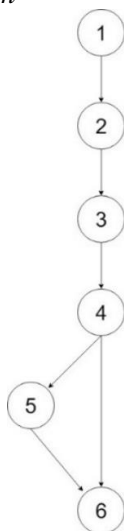
1. *Pseudocode* relasi *node* sensor

Tabel 9 *pseudocode* relasi *node* sensor

No	Keterangan
1.	Mulai
2.	Mencari relasi dari aktuator dan merubah string menjadi json objek
3.	Tampilkan relasi
4.	If LDR terhubung dengan actuator
5.	Kirim data sensor 0 atau 1 untuk mengisi variabel data pada aktuator
6.	selesai

2. *Basis path testing*

i. *Flow diagram*



Gambar 17 *Flow graph* relasi pada *node* sensor

ii. *Cyclomatic complexity*

$$\begin{aligned}
 V(G) &= \text{Edges} - \text{Nodes} + 2 \\
 &= 6 \text{ Edges} - 6 \text{ Nodes} + 2 \\
 &= 2
 \end{aligned}$$

$$V(G) = 2 \text{ Region R1 dan R2}$$

iii. *Independen path*

- 1) 1-2-3-4-5-6
- 2) 1-2-3-4-6

3. Hasil Pengujian

Tabel 10 hasil pengujian relasi

Jalur	Test Case	Expected Result	Result	status
-------	-----------	-----------------	--------	--------

5. PENUTUP

Berdasarkan hasil yang dipaparkan pada tahap perancangan dan pengujian bisa disimpulkan aplikasi dapat mengenali perangkat yang terhubung dengan cara mengambil perangkat sensor dan aktuator yang terhubung dengan firebase. Relasi ditentukan oleh pengguna yaitu melalui sitem yang terdapat pada tampilan aplikasi, sedangkan untuk menentukan relasi pada sistem yaitu aplikasi bertindak sebagai yang menentukan perangkat sensor terhubung dengan perangkat aktuator dengan proses pengambilan nama dari sensor diambil sesuai dengan relasi dari yang ditentukan. Berdasarkan hasil dari pengujian performa pada pengujian didapatkan nilai *delay* 0,9999853603 detik untuk proses pengiriman data ke firebase. Nilai yang didapatkan termasuk *delay* yang sangat kecil untuk proses pengiriman data yang dilakukan suatu perangkat yang terhubung dengan internet.

Berdasarkan apa yang sudah di kerjakan sistem masih memiliki kekurangan yang dapat dijadikan sebagai pengembangan yakni : Penelitian bisa dikembangkan perangkat tidak memerlukan verifikasi lagi ketika perangkat tiba – tiba mati, Yang kedua aplikasi *smartphone* sebagai pengirim data dari perangkat satu dan yang lainnya sehingga perangkat dapat saling bertukar informasi dengan menggunakan database yang lain, sehingga akan didapatkan hasil perbandingan dari database yang berbeda.

DAFTAR PUSTAKA

cook, D. J. & Krisnan, N. C., 2015. Activity Learning: Discovering, Recognizing, and Predicting Human Behavior from Sensor Data. In: Canada: Google Books , p. 12.

Kadam, R., Mahammuni, P. & Parikh, Y., 2015. Smart Home System. *International Journal of Innovative Research in*

*Advanced Engineering (IJIRAE) ISSN: 2349-2163*, 2(1), pp. 81-86.

- Laberg, T., 2005. *Smart Home Technology ; Technology supporting independent living*. Oslo: Health and Social Affairs.
- Maaral, A. I., Su , X. & Riekk, J., 2016. Semantic Reasoning for Context-aware Internet of Things Applications. *IEEE INTERNET OF THINGS JOURNAL*, Volume 3, pp. 1-13.
- Ramparany, F. & Cao, Q. H., 2016. A Semantic Approach to IoT Data Aggregation and Interpretation applied to Home Automation. *2016 International Conference on Internet of Things and Applications (IOTA)* , pp. 23-28.
- Zhang, T., Gao, J., Cheng, J. & Uehara, T., 2015. *Compatibility Testing Service for Mobile Applications*.
- Prasetyo, A., Akbar, S, R., Priyambada, B., 2018. *Implementasi Sistem Semantik pad Perangkat Lampu Rumah Cerdas Berbasis Agen*.Malang.