BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis Masalah

Pada analisis masalah *game* Resistansi Listrik yang akan dibangun yaitu pada aksi cahaya lampu (mati, berkedip, redup, terang dan pecah) dan kecepatan pintu (lambat, cepat dan sangat lambat) harus dapat menyesuaikan dengan tegangan dan arus yang masuk dari *power supply*, Sehingga cahaya lampu harus dapat menyala dengan intensitas cahaya yang tepat dan kecepatan pintu dapat bergerak dengan kecepatan yang tepat sesuai dengan masukan tegangan dan arus, oleh karena itu diperlukan logika *fuzzy* untuk merepresentasikan hal tersebut.

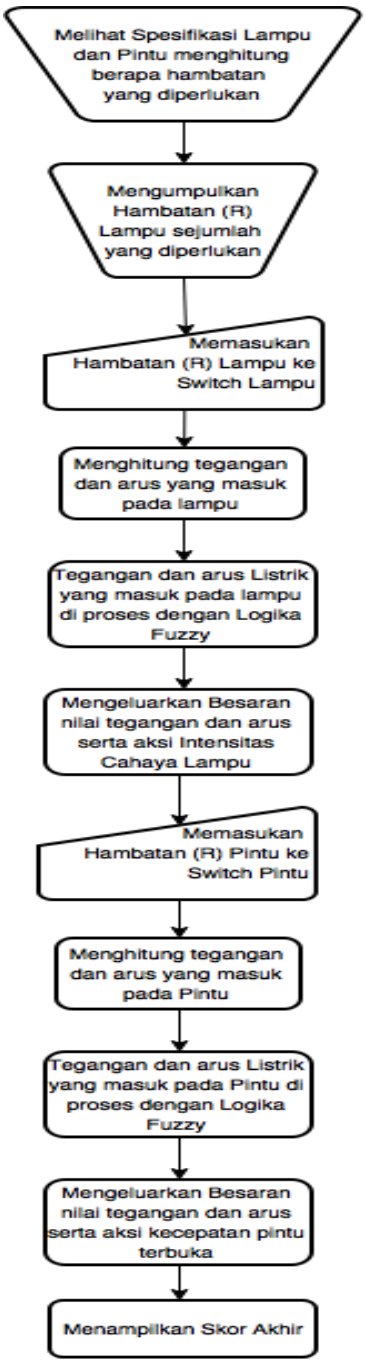
3.2 Analisis Game

Game Resistansi Listrik ini mempunyai latar sebuah ruangan berkarakteristik seperti gudang. Kasus gudang mengalami mati lampu dan pintu *rolling door electrict* tidak dapat beroperasi karena listrik nya mati yang disebabkan resistor atau hambatan listrik yang rusak, maka datang petugas listrik pabrik yang akan menyelesaikan masalah tersebut sampai pada listrik di gudang tersebut menyala dan lampu kembali hidup serta pintu dapat bergerak kembali dengan batas waktu yang ditentukan, apabila waktu habis atau pemain tertangkap musuh maka *game* akan langsung berakhir. Di dalam *game* ini pemain berperan sebagai petugas listrik, dimana pemain di haruskan menyalakan lampu dengan terang dan membuka pintu dengan kecepatan cepat pada sebuah ruangan berdasarkan variabel masukan yaitu Tegangan, Arus, dan Hambatan. Dengan cara pemain harus menghitung hambatan yang ideal bagi lampu dan pintu menggunakan hukum ohm lalu mengumpulkan hambatan yang terdapat pada ruangan area permainan tersebut, lalu hambatan tersebut menjadi penentu seberapa besar tegangan dan arus yang akan masuk pada lampu dan pintu, lalu pada tahap ini logika *fuzzy* dijalankan dan akan menghasilkan perilaku cahaya lampu dan kecepatan pergerakan pintu yang variatif sesuai dengan tegangan dan arus yang masuk, pada kecepatan pergerakan musuh ditentukan oleh aksi dari

intensitas cahaya lampu yang di dapat, apabila cahaya lampu mati dan pecah pergerakan musuh akan lambat, cahaya kedip pergerakan musuh sedang, cahaya redup pergerakan musuh cepat dan cahaya terang pergerakan musuh sangat cepat.

3.2.1 Skenario *Game*

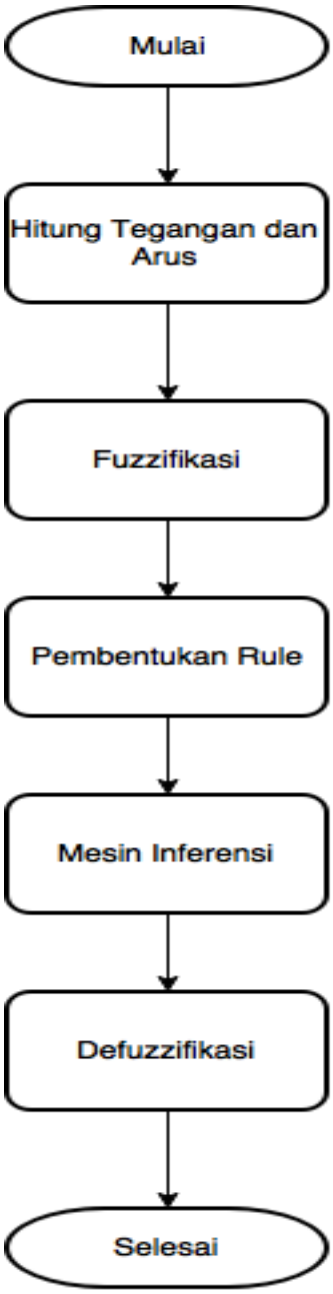
Skenario dalam *game* dapat dilihat alur permainannya sebagai berikut:



Gambar 3.1 Flowchart Alur Permainan

3.2.2 Skenario Logika Fuzzy Pada *Game*

Skenario logika fuzzy pada game dapat dilihat alurnya pada gambar berikut :



Gambar 3.2 Flowchart Logika Fuzzy Pada *Game*

Berikut deskripsikan *game* yang akan dibangun :

Tabel 3.1 *Game* Yang Dibangun

Keterangan	
Desain Grafis	3D
Karakter Utama	Petugas Listrik Pabrik
Pergerakan Karakter	Pergerakan karakter digerakan oleh pemain, pergerakan meliputi : 1. Maju 2. Mundur 3. Bergerak Ke Kanan 4. Bergerak Ke Kiri 5. Mengambil R
Rintangan	Waktu Permainan dan Set ruangan
Target	Mengumpulkan hambatan (R) untuk menyalakan lampu dengan terang dan membuka pintu dengan kecepatan pintu cepat.

3.3 Analisis Masukan

Analisis masukan yang dilakukan dalam penelitian ini adalah menganalisis variabel dalam permainan dan dalam algoritma logika *fuzzy*, variabel tersebut antara lain Hambatan (R), Tegangan(V) dan Arus (I).

3.3.1 Analisis Masukan pada *Game*

Terdapat 1 variabel pada *game* ini, dimana variabel ini akan menentukan hasil *output* pada *game*. Adapun variabel yang digunakan adalah:

3.3.1.1 Variabel Hambatan

Hambatan (R) digunakan untuk membatasi arus dan tegangan listrik yang akan masuk dari tegangan *supply* ke pintu dan lampu, hambatan diperoleh dari perhitungan antara spesifikasi lampu dan spesifikasi pintu dengan tegangan supply menggunakan hukum ohm dengan rumus :

R = (P*ower Supply* - V.(lampu atau pintu) / I.(lampu atau pintu)

Contoh Kasus : Pemain harus mencari nilai hambatan yang diperlukan untuk menyalakan lampu dengan Spesifikasi *Power Supply* dan lampu sebagai berikut :

- *Power Supply* = 220V
- Spesifikasi Lampu = 20V10A

Maka diselesaikan dengan rumus (3.1)

R=(220V – 20V) / 10A

= 200V /10A

= 20 Ω

Jadi dibutuhkan hambatan atau R sejumlah 20 Ω untuk dapat lampu menyala terang.

3.3.1.2 Variabel Waktu

Waktu digunakan untuk membatasi waktu pemain berada di area permainan, sehingga apabila waktu habis maka permainan akan langsung berakhir tanpa skor akhir. Untuk keterangan waktu ada pada tabel 3.2.

Tabel 3.2 Keterangan Variabel

No	Nama	Spesifikasi	Keterangan
1	*PowerSupply*	220V	Tegangan dari sumber listrik yang dapat disalurkan ke objek, nilai sudah ditentukan sistem sesuai spesifikasi. Contoh:220V
2	V.lampu dan I.lampu	Lampu 1: 20V10A Lampu 2: 40V20A Lampu 3: 80V20A	V.Lampu yaitu tegangan lampu dan I.Lampu yaitu Arus lampu, nilai sudah ditentukan sesuai spesifikasi. Contoh:Lampu 1 20V10A
3	V.Pintu dan I.Pintu	Pintu 1 : 120V20A Pintu 2 : 160V20A Pintu 3 : 180V20A	V.Lampu yaitu tegangan lampu dan I.Lampu yaitu Arus lampu, nilai sudah ditentukan sesuai spesifikasi.

			Contoh: 120V10A
4	Hambatan (R)	Nilai diperoleh dari R yang diambil pemain	Hambatan ditentukan pemain, dengan cara dihitung dengan hukum ohm mencari R berdasarkan suplai tegangan dan spesifikasi lampu dan pintu.
5	Hambatan Objek (R.Objek)	Nilai dihitung sistem	R.Objek yaitu hambatan dari objek tersebut (lampu dan pintu),Dihitung dengan tegangan di bagi arus atau V/I sehingga dihasilkan R objek. Contoh : R.objek =4V/2I=2 Ω
6	I.Masuk Objek	Dihitung Sistem	I.Masuk Objek adalah Arus yang masuk ke dalam objek lampu atau pintu dengan rumus $I.Masuk\ Objek = \frac{Power\ Supply}{(R+R\ (objek)}$
7	V.Masuk Objek	Dihitung Sistem	V.Masuk Objek adalah Tegangan yang masuk ke dalam objek lampu atau pintu, dihitung dengan rumus : V.Masuk Objek= I.Objek x R.Objek
8	Waktu	Ditentukan Sistem	Waktu digunakan untuk membatasi waktu pemain berada di area permainan, sehingga apabila waktu habis maka permainan akan langsung berakhir tanpa skor akhir. Contoh : 120 detik

9	Pergerakan Musuh	Lampu Mati : Pergerakan Musuh Lambat Lampu Kedip : Pergerakan Musuh Sedang Lampu Redup : Pergerakan Musuh Cepat Lampu Terang : Pergerakan Musuh Sangat Cepat Lampu Pecah : Pergerakan Musuh lambat	Aksi pergerakan musuh menyesuaikan kecepatan nya dengan intensitas cahaya lampu yang di dapat. Contoh : Apabila pemain memasukan hambatan lampu dan menghasilkan intensitas cahaya lampu terang maka pergerakan musuh akan sangat cepat.

3.3.2 Analisis Masukan pada Logika Fuzzy

Variabel yang digunakan dalam *game* Resistansi Listrik terdapat 4 variabel, Berikut Penjelasan dan masing-masing himpunanya :

1. Arus→Sangat Rendah, Rendah, Cukup, Tinggi dan Sangat Tinggi.

   Variabel ini untuk menunjukan seberapa besar Arus listrik yang akan masuk ke dalam objek (Lampu atau Pintu), Variabel ini dipengaruhi oleh tegangan *Supply*, Hambatan dan Hambatan Objek, dihitung dengan rumus :

   $$I_{MasukObjek} = \frac{Power\ Supply}{(R + R\ (objek)}$$

   Dimana I Masuk Objek adalah Arus Masuk Objek (Lampu dan Pintu), untuk keterangan rumus ada pada Tabel 3.2.

2. Tegangan→Sangat Rendah, Rendah, Cukup, Tinggi dan Sangat Tinggi.

   Variabel ini untuk menunjukan seberapa besar tegangan listrik yang akan masuk ke dalam objek (Lampu atau Pintu), Variabel ini dipengaruhi oleh Arus objek dan Hambatan Objek dengan rumus :

   $$R_{objek} = V_{objek} \,/\, I_{objek}$$

   $$V_{MasukObjek} = I_{masuk\ objek}\ x\ R_{objek}$$

   Dimana V Masuk Objek adalah Tegangan Masuk Objek (Lampu dan Pintu) dan R Objek adalah Hambatan dari Objek (Lampu dan Pintu), Untuk Keterangan Rumus ada pada Tabel 3.2.

3. Intensitas Cahaya Lampu→Mati, Kedip, Redup, Terang dan Pecah.
4. Kecepatan Pergerakan Pintu → Lambat, Cepat dan Sangat Lambat.

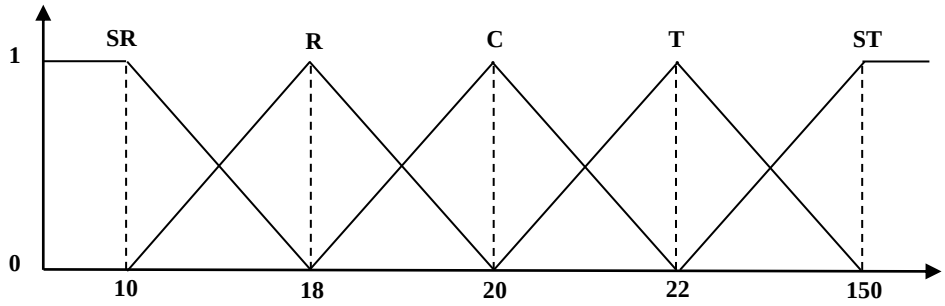
Dari setiap variabel mempunyai domain yang dinamis hal ini dilakukan karena setiap spesifikasi tegangan dan arus dari lampu dan pintu akan berbeda-beda. Untuk Spesifikasi Lampu dan pintu dapat dilihat pada Tabel 3.2 pada nomor 2 dan 3.

Berikut adalah domain-domain yang telah ditentukan berdasarkan setiap spesifikasi lampu dan pintunya.

a. Lampu 1 20V10A

- Tegangan
  - Sangat Rendah : 10V – 18V (Bahu Kiri)
  - Rendah : 10V –18V– 20V (Segitiga)
  - Cukup : 18V–20V–22V (Segitiga)
  - Tinggi : 20V–22V– 150V (Segitiga)
  - Sangat Tinggi : 22V – 150V (Bahu Kanan)

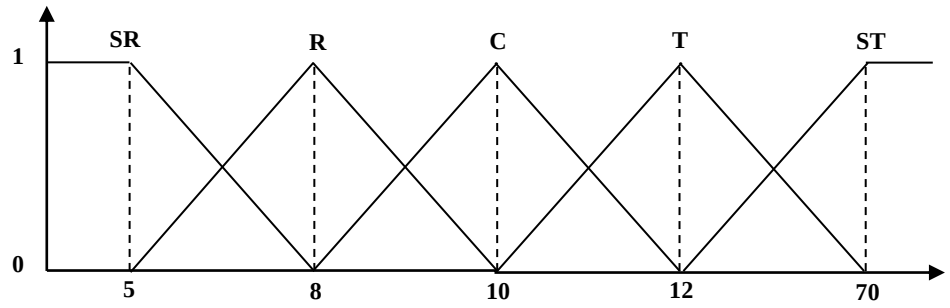
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.3



Gambar 3.3 Fungsi Keanggotaan Tegangan Lampu Kecil

- Arus
  - Sangat Rendah : 5A – 8A (Bahu Kiri)
  - Rendah : 5A – 8A– 10A (Segitiga)
  - Cukup : 8A–10A–12A (Segitiga)
  - Tinggi : 10A–12A– 70A (Segitiga)
  - Sangat Tinggi : 12A – 70A (Bahu Kanan)

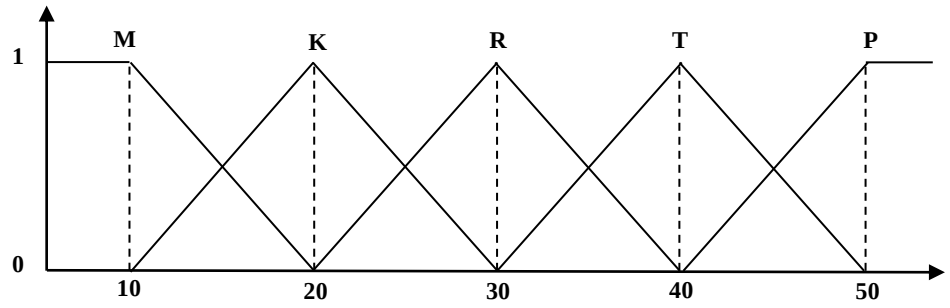
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.34



Gambar 3.4 Fungsi Keanggotaan Arus Lampu 1

- Tingkat Intensitas Cahaya
  - Mati : 10 – 20 (Bahu Kiri)
  - Kedip : 10 – 20– 30 (Segitiga)
  - Redup : 20 – 30 – 40 (Segitiga)
  - Terang : 30 – 40 – 50 (Segitiga)
  - Pecah : 40 – 50 (Bahu Kanan)

Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.5

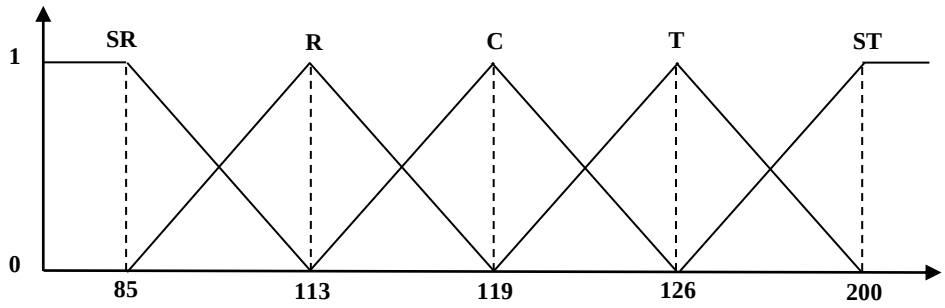


Gambar 3.5 Fungsi Keanggotaan Intensitas Lampu Kecil

b. Pintu 1 120V20A

- Tegangan
  - Sangat Rendah : 85V – 113V (Bahu Kiri)
  - Rendah : 85V – 113V– 119V (Segitiga)
  - Cukup : 113V–119V–126V (Segitiga)
  - Tinggi : 119V–126V– 200V (Segitiga)
  - Sangat Tinggi : 126V – 200V (Bahu Kanan)

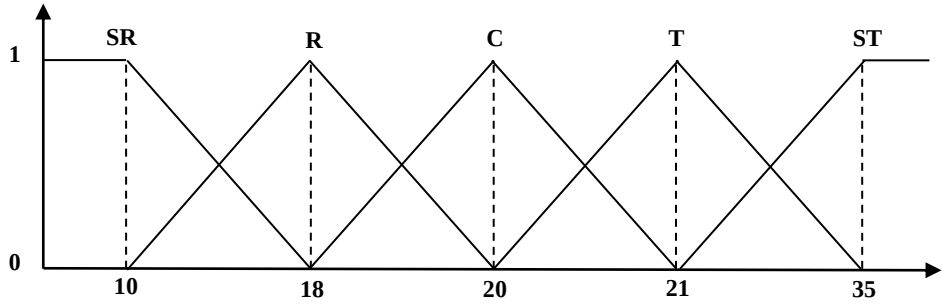
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.6



Gambar 3.6 Fungsi Keanggotaan Tegangan Pintu 1

- Arus
  - Sangat Rendah : 10A – 18A (Bahu Kiri)
  - Rendah : 10A – 18A– 20A (Segitiga)
  - Cukup : 18A–20A–21A (Segitiga)
  - Tinggi : 20A–21A– 35A (Segitiga)
  - Sangat Tinggi : 21A – 35A (Bahu Kanan)

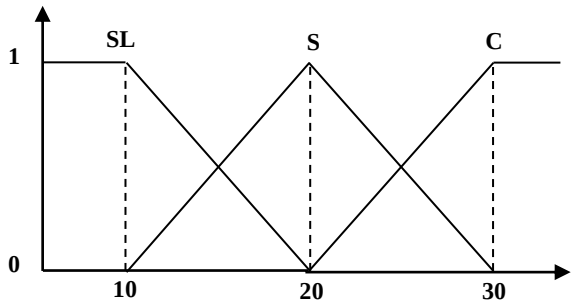
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.7



Gambar 3.7 Fungsi Keanggotaan Arus Pintu 1

- Kecepatan Pintu
  - Lambat : 10 – 20 (Bahu Kiri)
  - Sedang : 10 – 20– 30 (Segitiga)
  - Cepat : 20 – 30 (Bahu Kanan)

Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.8

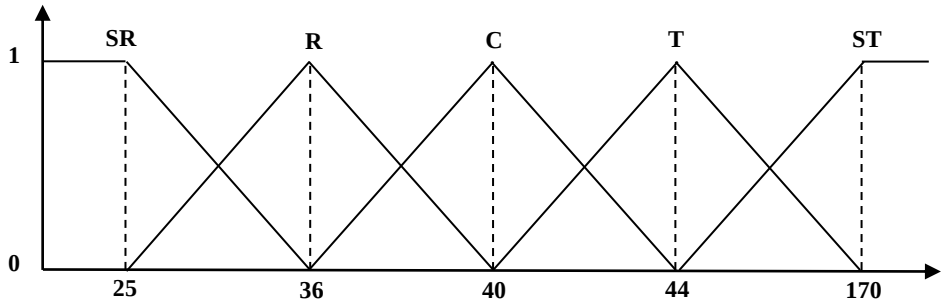


Gambar 3.8 Fungsi Keanggotaan Kecepatan Pintu 1

c. Lampu 2 40V20A

- Tegangan
  - Sangat Rendah : 25V – 36V (Bahu Kiri)
  - Rendah : 25V – 36V– 40V (Segitiga)
  - Cukup : 36V– 40V– 44V (Segitiga)
  - Tinggi : 40V– 44V – 170V (Segitiga)
  - Sangat Tinggi : 44V – 170V (Bahu Kanan)

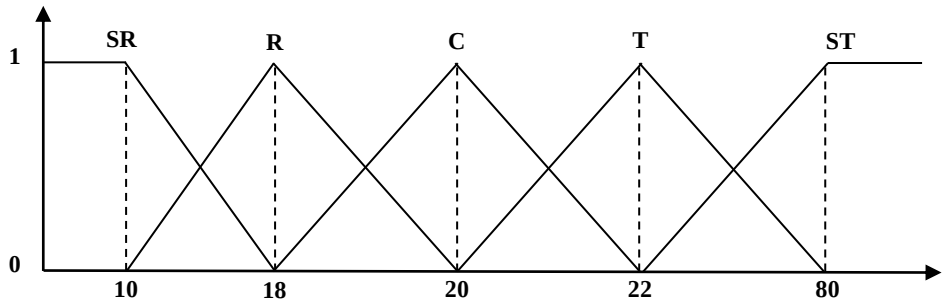
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.9



Gambar 3.9 Fungsi Keanggotaan Tegangan Lampu 2

- Arus
  - Sangat Rendah : 10A – 18A (Bahu Kiri)
  - Rendah : 10A – 18A– 20A (Segitiga)
  - Cukup : 18A–20A–22A (Segitiga)
  - Tinggi : 20A–22A– 80A (Segitiga)
  - Sangat Tinggi : 22A – 80A (Bahu Kanan)

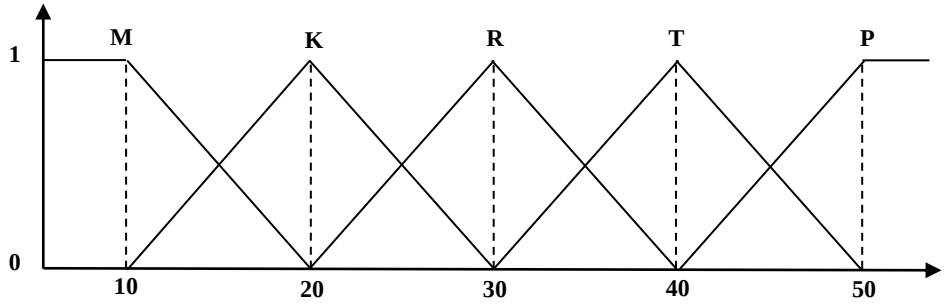
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.10



Gambar 3.10 Fungsi Keanggotaan Arus Lampu 2

- Tingkat Intensitas Cahaya
  - Mati : 10 – 20 (Bahu Kiri)
  - Kedip : 10 – 20– 30 (Segitiga)
  - Redup : 20–30–40 (Segitiga)
  - Terang : 30–40– 50 (Segitiga)
  - Pecah : 40 – 50 (Bahu Kanan)

Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.11

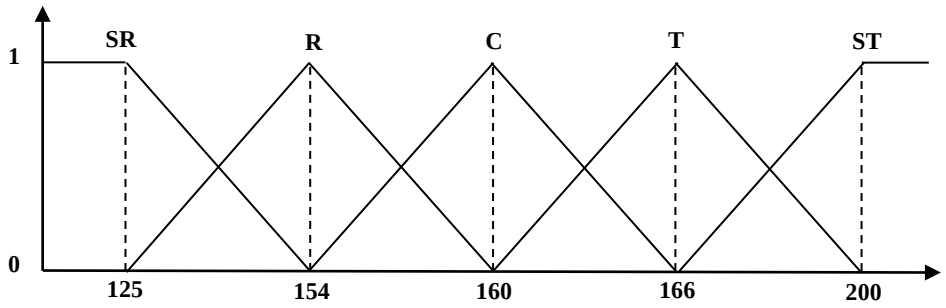


Gambar 3.11 Fungsi Keanggotaan Intensitas Lampu 2

d. Pintu 2 160V20A

- Tegangan
  - Sangat Rendah : 125V – 154V (Bahu Kiri)
  - Rendah : 125V – 154V– 160V (Segitiga)
  - Cukup : 154V–160V–166V (Segitiga)
  - Tinggi : 160V–166V– 200V (Segitiga)
  - Sangat Tinggi : 166V – 200V (Bahu Kanan)

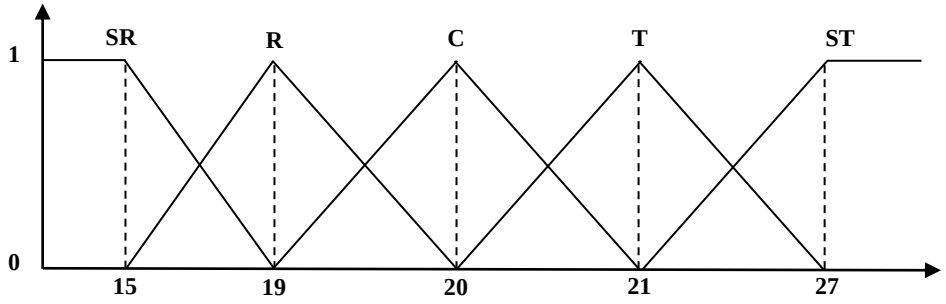
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.12



Gambar 3.12 Fungsi Keanggotaan Tegangan Pintu 2

- Arus
  - Sangat Rendah : 15A – 19A (Bahu Kiri)
  - Rendah : 15A – 19A– 20A (Segitiga)
  - Cukup : 19A–20A–21A (Segitiga)
  - Tinggi : 20A–21A– 27A (Segitiga)
  - Sangat Tinggi : 21A – 27A (Bahu Kanan)

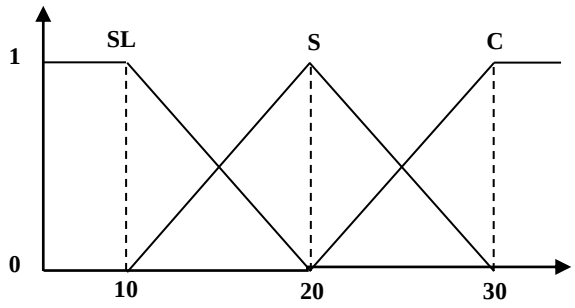
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.13



Gambar 3.13 Fungsi Keanggotaan Arus Pintu 2

- Kecepatan Pintu
  - Lambat : 10 – 20 (Bahu Kiri)
  - Sedang : 10 – 20– 30 (Segitiga)
  - Cepat : 20 – 30 (Bahu Kanan)

Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.14

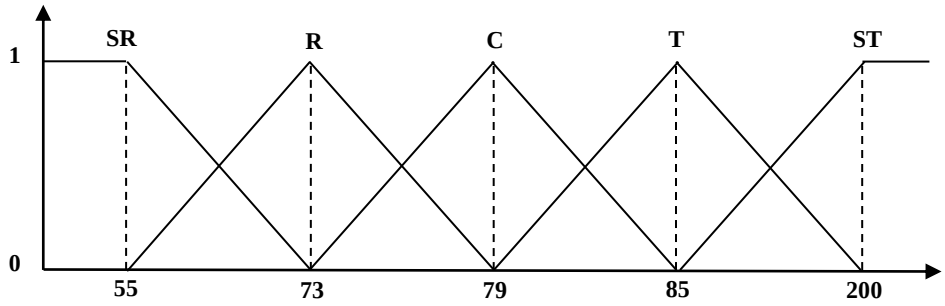


Gambar 3. 14 Fungsi Keanggotaan Kecepatan Pintu 2

e. Lampu 3 80V20A

- Tegangan
  - Sangat Rendah : 55V – 73V (Bahu Kiri)
  - Rendah : 55V – 73V– 79V (Segitiga)
  - Cukup : 73V–79V–85V (Segitiga)
  - Tinggi : 79V–85V– 200V (Segitiga)
  - Sangat Tinggi : 85V – 200V (Bahu Kanan)

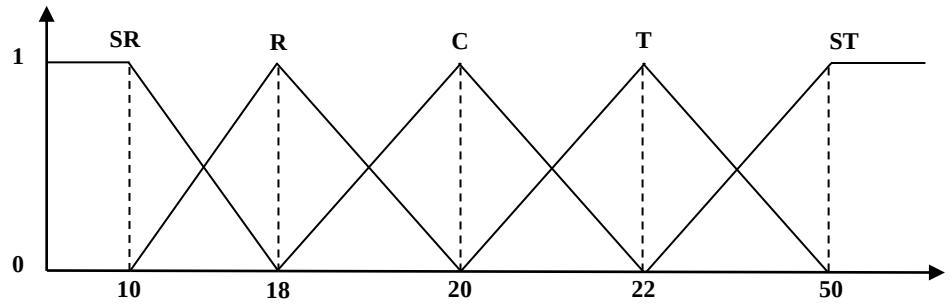
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.15



Gambar 3.15 Fungsi Keanggotaan Tegangan Lampu 3

- Arus
  - Sangat Rendah : 10A – 18A (Bahu Kiri)
  - Rendah : 10A – 18A– 20A (Segitiga)
  - Cukup : 18A–20A–22A (Segitiga)
  - Tinggi : 20A–22A– 50A (Segitiga)
  - Sangat Tinggi : 22A – 50A (Bahu Kanan)

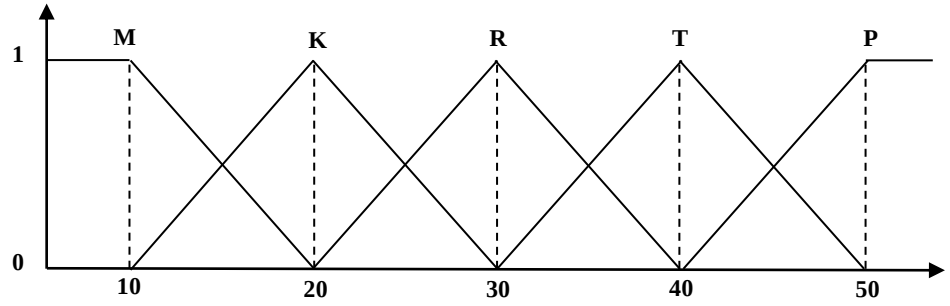
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.16



Gambar 3.16 Fungsi Keanggotaan Arus Lampu 3

- Tingkat Intensitas Cahaya
  - Mati : 10 – 20 (Bahu Kiri)
  - Kedip : 10 – 20– 30 (Segitiga)
  - Redup : 20–30–40 (Segitiga)
  - Terang : 30–40– 50 (Segitiga)
  - Pecah : 40 – 50 (Bahu Kanan)

Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.17

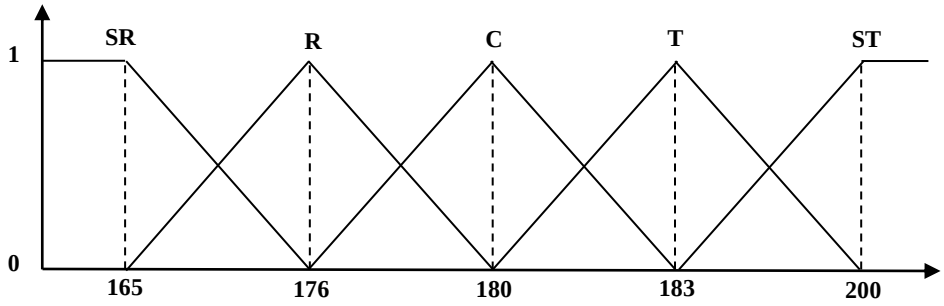


Gambar 3.17 Fungsi Keanggotaan Intensitas Lampu 3

f. Pintu 3 180V20A

- Tegangan
  - Sangat Rendah : 165V – 176V (Bahu Kiri)
  - Rendah : 165V – 176V– 180V (Segitiga)
  - Cukup : 176V–180V–183V (Segitiga)
  - Tinggi : 180V–183V– 200V (Segitiga)
  - Sangat Tinggi : 183V – 200V (Bahu Kanan)

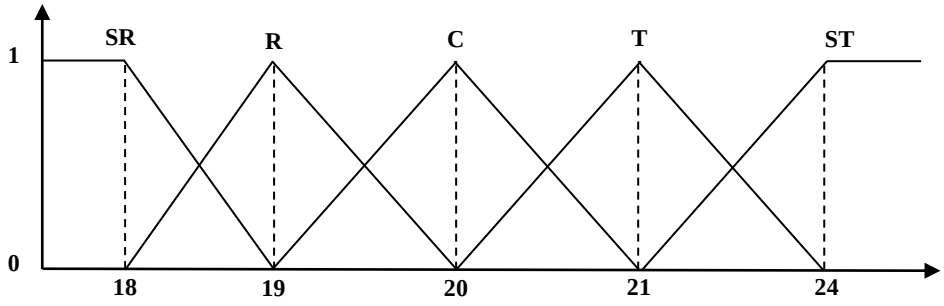
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.18



Gambar 3.18 Fungsi Keanggotaan Tegangan Pintu 3

- Arus
  - Sangat Rendah : 18A – 19A (Bahu Kiri)
  - Rendah : 18A – 19A– 20A (Segitiga)
  - Cukup : 19A–20A–21A (Segitiga)
  - Tinggi : 20A–21A– 24A (Segitiga)
  - Sangat Tinggi : 21A – 24A (Bahu Kanan)

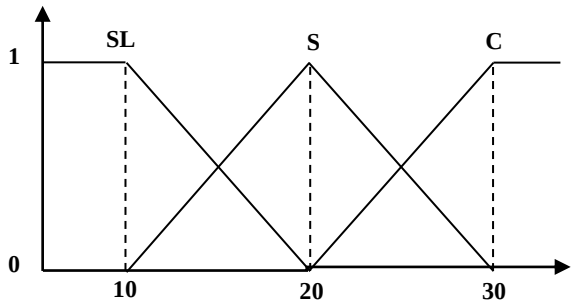
Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.19



Gambar 3.19 Fungsi Keanggotaan Arus Pintu 3

- Kecepatan Pintu
  - Lambat : 10 – 20 (Bahu Kiri)
  - Sedang : 10 – 20– 30 (Segitiga)
  - Cepat : 20 – 30 (Bahu Kanan)

Pada range nilai yang sudah di jelaskan di atas, maka akan membentuk hasil fungsi keanggotaan yang terdapat pada gambar 3.20



Gambar 3.20 Fungsi Keanggotaan Kecepatan Pintu 3

3.4 Analisis *Scoring*

Pada *game* Resistansi Listrik terdapat *Ranking* yang diberikan setiap menyelesaikan permainan, berdasarkan 2 masukan hasil dari intensitas cahaya lampu dan kecepatan pintu bergerak.

Dengan tingkatan *Ranking* : A = Tinggi
B = Sedang
C = Buruk
D = Buruk Sekali

berikut tabel *Rule* untuk penentuan *Score* :

Tabel 3.3 Tabel *Score*

Objek	Pintu			
Lampu	Keterangan	Lambat	Cepat	Sangat Lambat
	Mati	D	C	D
	Kedip	C	B	D
	Redup	C	A	C
	Terang	D	A	C
	Pecah	D	C	D

3.5 Analisis Algoritma Logika *Fuzzy*

Analisis algoritma yang dilakukan dalam penelitian ini adalah meneliti bagaimana cara kerja algoritma logika fuzzy pada lampu dalam menentukan nilai intensitas cahaya dan pintu dalam menentukan kecepatan pintu bergerak di dalam *game* simulasi. Algoritma logika *fuzzy* ini memiliki 2 variabel yaitu variabel Tegangan dan Arus. Serta algoritma logika *fuzzy* memiliki 4 tahapan proses yaitu fuzzifikasi, pembentukan *rule*, mesin inferensi dan defuzzifikasi. Berikut adalah ilustrasi logika *fuzzy* yang di peroleh dari 2 variabel masukan untuk 2 keluaran yang berbeda dapat di lihat pada tabel input dan output algoritma logika *fuzzy*.

Tabel 3.4 Tabel *Rule* Logika *Fuzzy* Lampu

No	Tegangan	Arus	Aksi Lampu
1	Sangat Rendah	Sangat Rendah	Mati
2	Sangat Rendah	Rendah	Mati
3	Sangat Rendah	Cukup	Mati
4	Sangat Rendah	Tinggi	Mati
5	Sangat Rendah	Sangat Tinggi	Mati
6	Rendah	Sangat Rendah	Kedip
7	Rendah	Rendah	Kedip
8	Rendah	Cukup	Kedip
9	Rendah	Tinggi	Redup
10	Rendah	Sangat Tinggi	Redup
11	Cukup	Sangat Rendah	Redup
12	Cukup	Rendah	Redup
13	Cukup	Cukup	Terang
14	Cukup	Tinggi	Redup
15	Cukup	Sangat Tinggi	Redup
16	Tinggi	Sangat Rendah	Pecah
17	Tinggi	Rendah	Pecah
18	Tinggi	Cukup	Pecah

19	Tinggi	Tinggi	Pecah
20	Tinggi	Sangat Tinggi	Pecah
21	Sangat Tinggi	Sangat Rendah	Pecah
22	Sangat Tinggi	Rendah	Pecah
23	Sangat Tinggi	Cukup	Pecah
24	Sangat Tinggi	Tinggi	Pecah
25	Sangat Tinggi	Sangat Tinggi	Pecah

Pada tabel input dan output digunakan untuk menentukan keputusan yang dilakukan oleh karakter utama. Berikut penjelasan dari masing – masing input dan output tersebut adalah:

3.5.1 Rule logika *fuzzy* Lampu

1. **IF** Tegangan is Sangat Rendah **AND** Arus is Sangat Rendah **Then** Lampu Mati.
2. **IF** Tegangan is Sangat Rendah **AND** Arus is Rendah **Then** Lampu Mati.
3. **IF** Tegangan is Sangat Rendah **AND** Arus is Cukup **Then** Lampu Mati.
4. **IF** Tegangan is Sangat Rendah **AND** Arus is Tinggi **Then** Lampu Mati.
5. **IF** Tegangan is Sangat Rendah **AND** Arus is Sangat Tinggi **Then** Lampu Mati.
6. **IF** Tegangan is Rendah **AND** Arus is Sangat Rendah **Then** Lampu Kedip.
7. **IF** Tegangan is Rendah **AND** Arus is Rendah **Then** Lampu Kedip.
8. **IF** Tegangan is Rendah **AND** Arus is Cukup **Then** Lampu Kedip.
9. **IF** Tegangan is Rendah **AND** Arus is Tinggi **Then** Lampu Redup.
10. **IF** Tegangan is Rendah **AND** Arus is Sangat Tinggi **Then** Lampu Redup.
11. **IF** Tegangan is Cukup **AND** Arus is Sangat Rendah **Then** Lampu Redup.

    IF Tegangan is Cukup **AND** Arus is Rendah **Then** Lampu Redup

    IF Tegangan is Cukup **AND** Arus is Cukup **Then** Lampu Terang.

    IF Tegangan is Cukup **AND** Arus is Tinggi **Then** Lampu Redup.

IF Tegangan is Cukup **AND** Arus is Sangat Tinggi **Then** Redup.

IF Tegangan is Tinggi **AND** Arus is Sangat Rendah **Then** Lampu Pecah.

IF Tegangan is Tinggi **AND** Arus is Rendah **Then** Lampu Pecah.

IF Tegangan is Tinggi **AND** Arus is Rendah **Then** Lampu Pecah.

IF Tegangan is Tinggi **AND** Arus is Tinggi Rendah **Then** Lampu Pecah.

IF Tegangan is Tinggi **AND** Arus is Sangat Tinggi Rendah **Then** Lampu Pecah.

IF Tegangan is Sangat Tinggi **AND** Arus is Sangat Rendah**Then** Rendah **Then** Lampu Pecah.

IF Tegangan is Sangat Tinggi **AND** Arus is Rendah **Then** Lampu Pecah.

IF Tegangan is Sangat Tinggi **AND** Arus is Cukup Rendah **Then** Lampu Pecah.

IF Tegangan is Sangat Tinggi **AND** Arus is Tinggi Rendah **Then** Lampu Pecah.

IF Tegangan is Sangat Tinggi **AND** Arus is Sangat Tinggi Rendah **Then** Lampu Pecah.

Tabel 3.5 Tabel *Rule* Logika *Fuzzy* Pintu

No	Tegangan	Arus	Aksi
1	Sangat Rendah	Sangat Rendah	Sangat Lambat
2	Sangat Rendah	Rendah	Sangat Lambat
3	Sangat Rendah	Cukup	Sangat Lambat
4	Sangat Rendah	Tinggi	Sangat Lambat
5	Sangat Rendah	Sangat Tinggi	Sangat Lambat
6	Rendah	Sangat Rendah	Sangat Lambat
7	Rendah	Rendah	Sangat Lambat
8	Rendah	Cukup	Sangat Lambat
9	Rendah	Tinggi	Sangat Lambat
10	Rendah	Sangat Tinggi	Sangat Lambat
11	Cukup	Sangat Rendah	Sangat Lambat

12	Cukup	Rendah	Sangat Lambat
13	Cukup	Cukup	Cepat
14	Cukup	Tinggi	Lambat
15	Cukup	Sangat Tinggi	Lambat
16	Tinggi	Sangat Rendah	Lambat
17	Tinggi	Rendah	Lambat
18	Tinggi	Cukup	Lambat
19	Tinggi	Tinggi	Lambat
20	Tinggi	Sangat Tinggi	Lambat
21	Sangat Tinggi	Sangat Rendah	Lambat
22	Sangat Tinggi	Rendah	Lambat
23	Sangat Tinggi	Cukup	Lambat
24	Sangat Tinggi	Tinggi	Lambat
25	Sangat Tinggi	Sangat Tinggi	Lambat

Pada tabel input dan output digunakan untuk menentukan keputusan yang dilakukan oleh karakter utama. Berikut penjelasan dari masing – masing input dan output tersebut adalah

3.5.2 Rule logika *fuzzy* Pintu

1. **IF** Tegangan is Sangat Rendah **AND** Arus is Sangat Rendah **Then** Kecepatan Pintu Sangat Lambat.
2. **IF** Tegangan is Sangat Rendah **AND** Arus is Rendah **Then** Kecepatan Pintu Sangat Lambat.
3. **IF** Tegangan is Sangat Rendah **AND** Arus is Cukup **Then** Kecepatan Pintu Sangat Lambat.
4. **IF** Tegangan is Sangat Rendah **AND** Arus is Tinggi **Then** Kecepatan Pintu Sangat Lambat.
5. **IF** Tegangan is Sangat Rendah **AND** Arus is Sangat Tinggi **Then** Kecepatan Pintu Sangat Lambat.
6. **IF** Tegangan is Rendah **AND** Arus is Sangat Rendah **Then** Kecepatan Pintu Sangat Lambat.

7. **IF** Tegangan is Rendah **AND** Arus is Rendah **Then** Kecepatan Pintu Sangat Lambat.
8. **IF** Tegangan is Rendah **AND** Arus is Cukup **Then** Kecepatan Pintu Sangat Lambat.
9. **IF** Tegangan is Rendah **AND** Arus is Tinggi **Then** Kecepatan Pintu Sangat Lambat.
10. **IF** Tegangan is Rendah **AND** Arus is Sangat Tinggi **Then** Kecepatan Pintu Sangat Lambat.
11. **IF** Tegangan is Cukup **AND** Arus is Sangat Rendah **Then** Kecepatan Pintu Sangat Lambat.
12. **IF** Tegangan is Cukup **AND** Arus is Rendah **Then** Kecepatan Pintu Sangat Lambat.
13. **IF** Tegangan is Cukup **AND** Arus is Cukup **Then** Kecepatan Pintu Cepat.
14. **IF** Tegangan is Cukup **AND** Arus is Tinggi **Then** Kecepatan Pintu Lambat.
15. **IF** Tegangan is Cukup **AND** Arus is Sangat Tinggi **Then** Kecepatan Pintu Lambat.
16. **IF** Tegangan is Tinggi **AND** Arus is Sangat Rendah **Then** Kecepatan Pintu Lambat.
17. **IF** Tegangan is Tinggi **AND** Arus is Rendah **Then** Kecepatan Pintu Lambat.
18. **IF** Tegangan is Tinggi **AND** Arus is Cukup **Then** Kecepatan Pintu Lambat.
19. **IF** Tegangan is Tinggi **AND** Arus is Tinggi **Then** Kecepatan Pintu Lambat.
20. **IF** Tegangan is Tinggi **AND** Arus is Sangat Tinggi **Then** Kecepatan Pintu Lambat
21. **IF** Tegangan is Sangat Tinggi **AND** Arus is Sangat Rendah **Then** Kecepatan Pintu Sangat Lambat.
22. **IF** Tegangan is Sangat Tinggi **AND** Arus is Rendah **Then** Kecepatan

Pintu Lambat.

23. **IF** Tegangan is Sangat Tinggi **AND** Arus is Cukup **Then** Kecepatan Pintu Lambat t.
24. **IF** Tegangan is Sangat Tinggi **AND** Arus is Tinggi**Then**Kecepatan Pintu Sangat Lambat.
25. **IF** Tegangan is Sangat Tinggi **AND** Arus is Sangat Tinggi **Then** Kecepatan Pintu Lambat.

3.5.3 Contoh Kasus :

Pemain memasukan nilai R lampu sebesar 21 dan R pintu sebesar 5, dengan spesifikasi *Power supply* sebesar 220V, maka:

A. Lampu 1: 20V10A

Maka arus yang masuk ke lampu adalah:

$$I_{Lampu} = \frac{PowerSupply}{\left(R + \left(\frac{20V}{10A}\right)\right)} = \frac{220}{(21 + 2)} = \mathbf{9,565217\ A}$$

sedangkan tegangan yang masuk ke lampu adalah:

$$V_{Lampu} = I_{Lampu}\ x\ R_{Lampu} = \mathbf{9.57}\ A\ x\ 2\ \Omega = \mathbf{19,13043\ V}$$

a. Tahapan Fuzzifikasi

Dengan nilai yang diberikan pada tegangan sebesar $\mathbf{19,13043}$ V dan arus $\mathbf{9,565217}$ A.

Nilai $\mathbf{19,13043}$ berada pada nilai linguistik.

- **Variabel Tegangan**

1. Himpunan Sangat Rendah (SR)

$$\mu SR(x) = \begin{cases} 1; & x \leq 10 \\ \frac{18 - x}{18 - 10}\ untuk & 10 \leq x \leq 18 \\ 0; & x \geq 18 \end{cases}$$

2. Himpunan Rendah (R)

$$\mu R(x) = \begin{cases} \dfrac{x-10}{18-10}; & 10 \le x \le 18 \\ \dfrac{20-x}{20-18}; & 18 \le x \le 20 \\ 0; & x \ge 20 \; atau \; x \le 10 \end{cases} untuk$$

3. Himpunan Cukup (C)

$$\mu C(x) = \begin{cases} 0; & x \ge 21 \; atau \; x \le 18 \\ \dfrac{x-18}{20-18}; & 18 \le x \le 20 \\ \dfrac{22-x}{22-20}; & 20 \le x \le 22 \end{cases} untuk$$

4. Himpunan Tinggi (T)

$$\mu T(x) = \begin{cases} 0; & x \ge 150 \; atau \; x \le 20 \\ \dfrac{x-20}{22-20}; & 20 \le x \le 22 \\ \dfrac{150-x}{150-22}; & 22 \le x \le 150 \end{cases} untuk$$

5. Himpunan Sangat Tinggi (ST)

$$\mu ST(x) = \begin{cases} 1; & x \ge 150 \\ \dfrac{x-22}{150-22}; & 22 \le x \le 150 \\ 0; & x \le 22 \end{cases} untuk$$

Derajat keanggotaan untuk Tegangan 19,13043 V adalah :

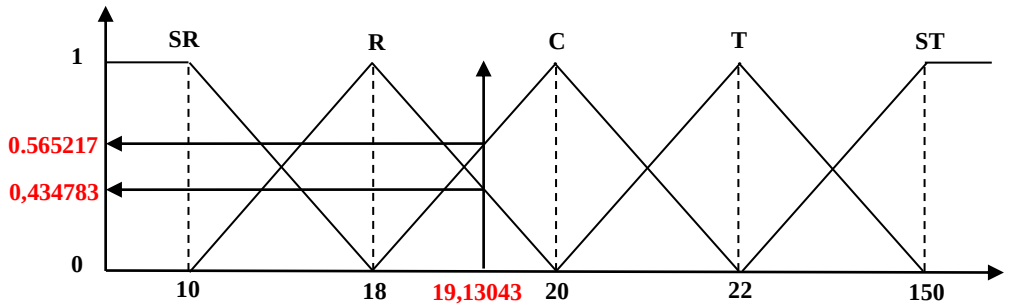
$\mu TeganganSangatRendah\ (19{,}13043\ ) = 0$

$\mu TeganganRendah\ (19{,}13043\ ) = \dfrac{(20-19{,}13043\ )}{(20-18)} = 0{,}434783$

$\mu TeganganCukup\ (19{,}13043\ ) = \dfrac{(19{,}13043\ -18)}{(20-18)} = 0{,}565217$

$\mu TeganganTinggi\ (19{,}13043\ ) = 0$

$\mu TeganganSangatTinggi\ (19{,}13043\ ) = 0$



Gambar 3.21 Fungsi Keanggotaan Tegangan 19,13043V Lampu 1

- **Variabel Arus**

1. Himpunan Sangat Rendah (SR)

$$\mu SR(x) = \begin{cases} 1; & x \leq 5 \\ \dfrac{8-x}{8-5} & untuk\ 5 \leq x \leq 8 \\ 0; & x \geq 8 \end{cases}$$

2. Himpunan Rendah (R)

$$\mu R(x) = \begin{cases} \dfrac{x-5}{8-5}; & 5 \leq x \leq 8 \\ \dfrac{10-x}{10-8}; & untuk \quad 8 \leq x \leq 10 \\ 0; & x \geq 10\ atau\ x \leq 5 \end{cases}$$

3. Himpunan Cukup (C)

$$\mu C(x) = \begin{cases} 0; & x \geq 11\ atau\ x \leq 8 \\ \dfrac{x-8}{10-8}; & untuk \quad 8 \leq x \leq 10 \\ \dfrac{12-x}{12-10}; & 10 \leq x \leq 12 \end{cases}$$

4. Himpunan Tinggi (T)

$$\mu T(x) = \begin{cases} 0; & x \geq 70\ atau\ x \leq 10 \\ \dfrac{x-10}{12-10}; & untuk \quad 10 \leq x \leq 12 \\ \dfrac{25-x}{70-12}; & 12 \leq x \leq 70 \end{cases}$$

5. Himpunan Sangat Tinggi (ST)

$$\mu ST(x) = \begin{cases} 1; & x \geq 70 \\ \dfrac{x-12}{70-12}; & untuk\ 12 \leq x \leq 70 \\ 0; & x \leq 12 \end{cases}$$

Derajat keanggotaan untuk Arus 9,565217 A adalah :

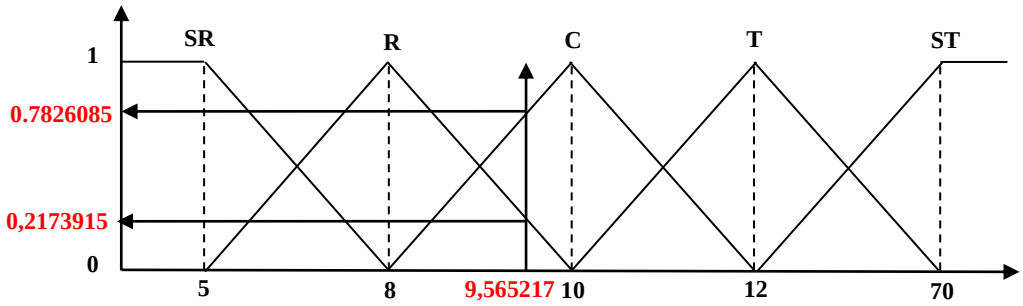
$\mu ArusSangatRendah\ (9{,}565217) = 0$

$\mu ArusRendah\ (9{,}565217) = \dfrac{(10-9{,}565217)}{(10-8)} = 0{,}2173915$

$\mu ArusCukup\ (9{,}565217) = \dfrac{(9{,}565217-8)}{(10-8)} = 0{,}7826085$

$\mu ArusTinggi\ (9{,}565217) = 0$

$\mu ArusSangatTinggi\ (9{,}565217) = 0$



Gambar 3.22 Fungsi Keanggotaan Arus 9,565217A Lampu 1

b. Tahap Pembentukan Rule

Dalam Hal ini, rule-rule dibentuk sesuai dengan yang memenuhi kondisi Soal.

Tabel 3.5 Tahap Pembentukan Rule

No	IF	Tegangan	AND	Arus	THEN	Lampu
1	IF	Rendah	AND	Rendah	THEN	Kedip
2	IF	Rendah	AND	Cukup	THEN	Kedip
3	IF	Cukup	AND	Rendah	THEN	Redup
4	IF	Cukup	AND	Cukup	THEN	Terang

c. Mesin Inferensi

[R1] IF Tegangan Rendah AND Arus Rendah THEN Lampu Kedip

α-pred1 = µRendah ∩ µRendah

=MIN(µRendah [0,434783]; µRendah[0,2173915])

=MIN (0,434783; 0,2173915)

=0,2173915

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel Lampu:

$$\frac{30 - z_1}{30 - 20} = 0{,}2173915$$

$$30 - z_1 = 0{,}2173915 * (30 - 20)$$

$$z_1 = 30 - 2{,}173915$$

$$Maka\ z_1 = 27{,}82608$$

[R2] IF Tegangan Rendah AND Arus Cukup THEN Lampu Kedip

α-pred2 = µRendah ∩ µcukup

= MIN(µRendah [0,434783]; µCukup[0,7826085])

= MIN (0,434783; 0,7826085)

= 0,434783

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel Lampu:

$$\frac{30 - z_2}{30 - 20} = 0{,}434783$$

$$30 - z_2 = 0{,}434783 * (30 - 20)$$

$$z_2 = 30 - 4{,}34783$$

$$Maka\ z_2 = 25.65217$$

[R3]IF Tegangan Cukup AND Arus Rendah THEN Lampu Redup

α-pred3= µCukup ∩ µRendah

= MIN(µCukup [0,565217]; µRendah [0,2173915])

= MIN (0,565217 ; 0,2173915)

= 0,2173915

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel Lampu:

$$\frac{40 - z_3}{40 - 30} = 0{,}2173915$$

$$40 - z_3 = 0{,}2173915 * (40 - 30)$$

$$z_3 = 40 - 2{,}173915$$

$$Maka\ z_3 = 37{,}82608$$

[R4] IF Tegangan Cukup AND Arus Cukup THEN Lampu Terang

α-pred4= µCukup ∩ µCukup

= MIN(µCukup [0,565217]; µCukup [0,7826085])

= MIN (0,565217 ; 0,7826085)

= 0,565217

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel Lampu:

$$\frac{50 - z_3}{40 - 40} = 0{,}565217$$

$$50 - z_4 = 0{,}565217 * (50 - 40)$$

$$z_4 = 50 - 5{,}65217$$

$$Maka\ z_4 = 44{,}34783$$

d. Defuzzifikasi

$$Z = \frac{\Sigma aizi}{ai}$$

αi = Nilai minimal dari derajat keanggotaan

Zi = Nilai domain pada variabel linguistik

Nilai Defuzzifikasi

$$Z = \frac{(0{,}2173915 * 27{,}82608) + (0{,}434783 * 25{,}65217) + (0{,}2173915 * 37{,}82608) + (0{,}565217 * 44{,}34783)}{0{,}2173915 + 0{,}434783 + 0{,}2173915 + 0{,}565217}$$

$$Z = 35{,}19104$$

Kesimpulan

Nilai yang didapat dari proses *defuzzyfication* tersebut selanjutnya digunakan untuk penentuan keputusan dengan aturan sebagai berikut :

Mati : memiliki nilai 5– 14.

Kedip : memiliki nilai 15 - 24.

Redup : memiliki nilai 25 – 34.

Terang : memiliki nilai 35 – 44.

Pecah : memiliki nilai 45 – 55 atau lebih dari 55.

Maka hasil nilai dari 35,19104 adalah Terang.

B. Pintu 1: 120V20A

arus yang masuk ke pintu adalah:

$$I_{Pintu} = \frac{PowerSupply}{\left(R + \left(\frac{120V}{20A}\right)\right)} = \frac{220}{(5 + 6)} = \mathbf{20\ A}$$

sedangkan tegangan yang masuk ke pintu adalah:

$$V_{Pintu} = I_{Pintu}\ x\ R_{Pintu} = \mathbf{20}A\ x\ 6\ \Omega = \mathbf{120\ V}$$

a. Tahapan Fuzzifikasi

Dengan nilai yang diberikan pada tegangan sebesar 120 V dan arus 20 A.

Nilai 120 berada pada nilai linguistik.

- **Variabel Tegangan**

1. Himpunan Sangat Rendah (SR)

$$\mu SR(x) = \begin{cases} 1; & x \leq 85 \\ \frac{113 - x}{113 - 10}\ untuk & 85 \leq x \leq 113 \\ 0; & x \geq 113 \end{cases}$$

2. Himpunan Rendah (R)

$$\mu R(x) = \begin{cases} \frac{x - 85}{113 - 85}; & 85 \leq x \leq 113 \\ \frac{119 - x}{119 - 113}; untuk & 113 \leq x \leq 119 \\ 0; & x \geq 119\ atau\ x \leq 85 \end{cases}$$

3. Himpunan Cukup (C)

$$\mu C(x) = \begin{cases} 0; & x \geq 126 \; atau \; x \leq 113 \\ \dfrac{x-113}{119-113}; & untuk \quad 113 \leq x \leq 119 \\ \dfrac{126-x}{126-119}; & 119 \leq x \leq 126 \end{cases}$$

4. Himpunan Tinggi (T)

$$\mu T(x) = \begin{cases} 0; & x \geq 200 \; atau \; x \leq 119 \\ \dfrac{x-119}{126-119}; & untuk \quad 119 \leq x \leq 126 \\ \dfrac{200-x}{200-126}; & 126 \leq x \leq 200 \end{cases}$$

5. Himpunan Sangat Tinggi (ST)

$$\mu ST(x) = \begin{cases} 1; & x \geq 200 \\ \dfrac{x-126}{200-126}; & untuk \;\; 126 \leq x \leq 200 \\ 0; & x \leq 126 \end{cases}$$

Derajat keanggotaan untuk Tegangan 120 V adalah :

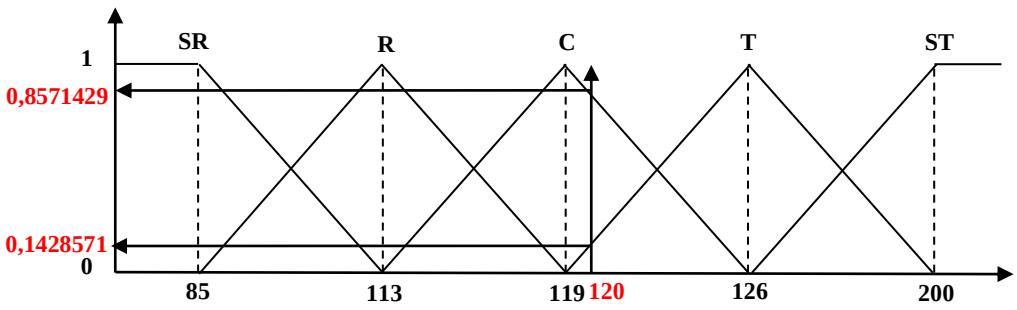
$\mu TeganganSangatRendah\ (120\ ) = 0$

$\mu TeganganRendah\ (120\ ) = 0$

$\mu TeganganCukup\ (120\ ) = \dfrac{(126-120)}{(126-119)} = 0{,}8571429$

$\mu TeganganTinggi\ (120) = \dfrac{(120-119)}{(126-119)} = 0{,}1428571$

$\mu TeganganSangatTinggi\ (120\ ) = 0$



Gambar 3.23 Fungsi Keanggotaan Tegangan 120V Pintu 1

- **Variabel Arus**

1. Himpunan Sangat Rendah (SR)

$$\mu SR(x) = \begin{cases} 1; & x \leq 10 \\ \dfrac{18-x}{18-18} & untuk\ 10 \leq x \leq 18 \\ 0; & x \geq 18 \end{cases}$$

2. Himpunan Rendah (R)

$$\mu R(x) = \begin{cases} \dfrac{x-10}{18-10}; & 10 \leq x \leq 18 \\ \dfrac{20-x}{20-18}; & untuk \quad 18 \leq x \leq 20 \\ 0; & x \geq 20\ atau\ x \leq 10 \end{cases}$$

3. Himpunan Cukup (C)

$$\mu C(x) = \begin{cases} 0; & x \geq 21\ atau\ x \leq 18 \\ \dfrac{x-18}{20-18}; & untuk \quad 18 \leq x \leq 20 \\ \dfrac{21-x}{21-20}; & 20 \leq x \leq 21 \end{cases}$$

4. Himpunan Tinggi (T)

$$\mu T(x) = \begin{cases} 0; & x \geq 35\ atau\ x \leq 20 \\ \dfrac{x-20}{21-20}; & untuk \quad 20 \leq x \leq 21 \\ \dfrac{35-x}{35-21}; & 21 \leq x \leq 35 \end{cases}$$

5. Himpunan Sangat Tinggi (ST)

$$\mu ST(x) = \begin{cases} 1; & x \geq 35 \\ \dfrac{x-21}{35-21}; & untuk\ 21 \leq x \leq 35 \\ 0; & x \leq 21 \end{cases}$$

Derajat keanggotaan untuk Arus 20 A adalah :

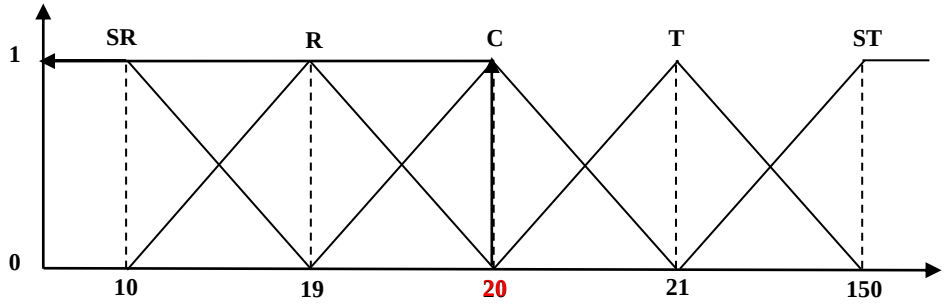
$\mu ArusSangatRendah\ (20) = 0$

$\mu ArusRendah\ (20) = 0$

$\mu ArusCukup\ (20) = 1$

$\mu ArusTinggi\ (20) = 0$

$\mu ArusSangatTinggi\ (20) = 0$



Gambar 3.24 Fungsi Keanggotaan Arus 20A Pintu 1

b. Tahap Pembentukan Rule

Dalam Hal ini, rule-rule dibentuk sesuai dengan yang memenuhi kondisi Soal.

Tabel 3.6 Tahap Pembentukan Rule

No	IF	Tegangan	AND	Arus	THEN	Pintu
1	IF	Cukup	AND	Cukup	THEN	Cepat
2	IF	Cukup	AND	Tinggi	THEN	Lambat
3	IF	Tinggi	AND	Cukup	THEN	Lambat
4	IF	Tinggi	AND	Tinggi	THEN	Lambat

c. Mesin Inferensi

[R1] IF Tegangan Cukup AND Arus Cukup THEN Pintu Cepat

α-pred$_1$ = μCukup ∩ μCukup

=MIN(μCukup [0,8571429]; μCukup [1])

=MIN (0,8571429; 1)

=0,8571429

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel pintu:

$$\frac{30 - z_1}{30 - 20} = 0{,}8571429$$

$$30 - z_1 = 0{,}8571429 * (30 - 20)$$

$$z_1 = 30 - 8{,}571429$$

$$Maka\ z_1 = 21{,}42857$$

[R2] IF Tegangan Cukup AND Arus Tinggi THEN Pintu Lambat

α-pred2 = µCukup ∩ µTinggi
= MIN(µCukup [0,8571429]; µTinggi [0])
= MIN (0,8571429; 0)
= 0

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel pintu:

$$\frac{z_2 - 20}{30 - 20} = 0$$

$$z_2 - 20 = 0 * (30 - 20)$$

$$z_2 = 20 - 0$$

$$Maka\ z_2 = 20$$

[R3]IF Tegangan Tinggi AND Arus Cukup THEN Pintu Lambat

α-pred3= µTinggi ∩ µCukup
= MIN(µTinggi [0,1428571]; µCukup [1])
= MIN (0,1428571 ; 1)
= 0,1428571

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel pintu:

$$\frac{z_3 - 20}{30 - 20} = 0{,}1428571$$

$$z_3 - 20 = 0{,}1428571 * (30 - 20)$$

$$z_3 = 20 - 1{,}428571$$

$$Maka\ z_3 = 18.571429$$

[R4] IF Tegangan Tinggi AND Arus Tinggi THEN Pintu Lambat

α-pred4= μTinggi ∩ μTinggi

= MIN(μTinggi [0,1428571]; μTinggi [0])

= MIN (0,1428571 ; 0)

= 0

Dari sini dapat dihitung suatu nilai tunggal dalam semesta pembicaraan pada variabel pintu:

$$\frac{z_4 - 20}{30 - 20} = 0$$

$$z_4 - 20 = 0 * (30 - 20)$$

$$z_4 = 20 - 0$$

$$Maka\ z_4 = 20$$

d. Defuzzifikasi

$$Z = \frac{\Sigma aizi}{ai}$$

ai = Nilai minimal dari derajat keanggotaan

Zi = Nilai domain pada variabel linguistik

z = Nilai defuzzifikasi

Nilai Defuzzifikasi

$$Z = \frac{(0{,}8571429 * 21{,}42857) + (0 * 20) + (0{,}1428571 * 18{,}571429) + (0 * 20)}{0{,}8571429 + 0 + 0{,}1428571 + 0}$$

$$Z = 21.020407$$

Kesimpulan :

Nilai yang didapat dari proses *defuzzyfication* tersebut selanjutnya digunakan untuk penentuan keputusan dengan aturan sebagai berikut :

Sangat lambat : memiliki nilai 5 – 14 atau kurang dari 5.

Cepat : memiliki nilai 15 - 24.

Lambat : memiliki nilai 25 – 34 atau lebih dari 34.

Maka hasil nilai dari 21,020407 adalah Cepat.

3.6 Analisis Kebutuhan Perangkat Lunak dan Perangkat Keras

Perangkat lunak yang digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar bisa saling berinteraksi diantara keduanya.

1. Pengembang

Spesifikasi perangkat lunak yang digunakan pengembang untuk membangun *game* ini dapat dilihat pada tabel berikut :

Tabel 3.7 Spesifikasi Perangkat Lunak Pengembang

Nama Perangkat Lunak	Spesifikasi
Sistem Operasi	Microsoft Windows 7
Tools	- Unity 3D 5.0 - Blender 2.73 - DirectX 11

2. Pengguna

Spesifikasi minimum perangkat lunak yang dibutuhkan pengguna untuk memainkan *game* ini dapat dilihat pada tabel berikut :

Tabel 3.8 Spesifikasi Minimum Perangkat Lunak Pengguna

Nama Perangkat Lunak	Spesifikasi
Sistem Operasi	Microsoft Windows XP SP2
Tools	- Unity Web Player - DirectX 9

3. Pengembang

Spesifikasi perangkat keras yang digunakan pengembang untuk membangun *game* ini dapat dilihat pada tabel berikut:

Tabel 3.9 Spesifikasi Perangkat Keras Pengembang

Nama Perangkat	Spesifikasi
Processor	Intel Core 2 Duo 2,1739150 GHz
RAM	4 GB
VGA Card	250 MB
Harddisk	500 GB
Monitor	Resolusi 1366 x 768
Mouse dan *keyboard*	Standar

4. Pengguna

Berdasarkan *system requirement* yang ada pada *website* Unity3D spesifikasi minimum perangkat keras bagi pengguna untuk dapat memainkan *game* ini dengan baik dapat dilihat pada tabel berikut :

Tabel 3.10 Spesifikasi Minimum Perangkat Keras Pengguna

Nama Perangkat	Spesifikasi
Processor	Intel Pentium 4
RAM	2 GB
VGA Card	1 GB
Harddisk	Tersedia 100 MB
Monitor	Resolusi 800 x 600
Mouse dan *keyboard*	Standar

Dengan spesifikasi yang lebih rendah maka pengguna tidak dapat memainkan aplikasi *game* dengan maksimal seperti mengalami ketersendatan atau gambar yang kabur.

3.7 Analisis pengguna

Adapun kriteria pengguna yang dapat menguji game ini adalah sebagai berikut:

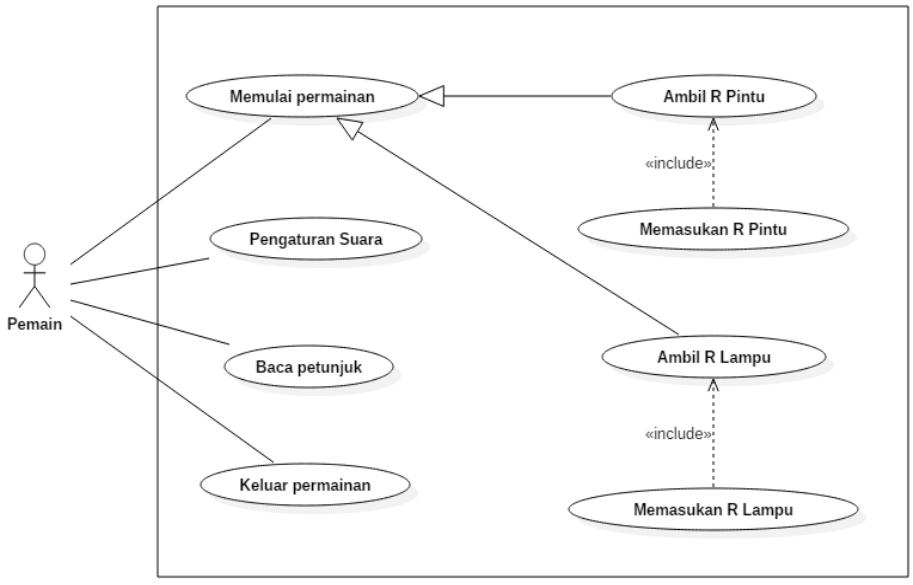
1. Pengguna tidak buta angka.
2. Pengguna tidak buta huruf.
3. Pengguna dapat membaca.

3.8 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik sesuai kebutuhan. Analisis kebutuhan fungsional dimodelkan dengan menggunakan UML *(Unified Modeling Language).*

3.8.1 Use Case Diagram

Diagram *use case* menggambarkan hubungan antara aktor dan sistem yang dibangun. Perancangan proses-proses yang terdapat pada *game* yang akan dibangun akan digambarkan dengan diagram *use case.* Interaksi antara aktor dengan aplikasi *game* yang akan dibangun digambarkan pada *diagram use case* pada Gambar 3.25.



Gambar 3.25 *Use case Diagram Game* yang akan dibangun

3.8.2 Definisi Aktor

Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem sistem yang akan dibangun. Definisi aktor diterangkan pada tabel berikut :

Tabel 3.11 Analisis Aktor

No	Aktor	Deskripsi
1	Pemain	Pemain atau orang yang memainkan game

3.8.3 Definisi *Use case*

Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. Definisi use case diterangkan pada tabel 3.14 :

Tabel 3.12 Definisi Use case

No	*Use case*	Deskripsi
1	Mulai permainan	Proses untuk masuk dalam permainan
2	Ambil Hambatan (R) Lampu	Proses untuk menambah nilai Hambatan lampu dengan mengambilnya dalam bentuk tabung warna merah yang terdapat pada area permainan
3	Masukan Hambatan (R) ke Switch lampu	Proses untuk memasukan hambatan lampu ke dalam Switch lampu.
4	Ambil Hambatan (R) Pintu	Proses untuk menambah nilai Hambatan pintu dengan mengambilnya dalam bentuk kotak warna abu yang terdapat pada area permainan
5	Masukan Hambatan (R) ke Switch Pintu	Proses untuk memasukan hambatan Pintu ke dalam Switch Pintu.
6	Pengaturan Suara	Proses untuk mengatur suara menjadi ON atau OFF
7	Petunjuk	Proses untuk melihat petunjuk atau panduan cara bermain
8	Keluar permainan	Proses untuk keluar dari permainan

3.8.4 Skenario *Use case*

Setiap use case dilengkapi dengan skenario. Skenario use case adalah alur jalannya proses use case dari sisi aktor dan sistem. Berikut adalah skenario jalannya masing-masing use case yang telah didefinisikan sebelumnya:

1. Skenario *Use case* Mulai Permainan

Tabel 3.13 Skenario *Use case* Mulai Permainan

Identifikasi	
No	1
Nama	Mulai Permainan
Tujuan	Untuk memulai permainan
Deskripsi	Proses untuk memulai permainan
Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di Menu Utama
Aksi aktor	**Reaksi sistem**
	1. Menampilkan layar menu utama
2. Memilih menu mulai Permainan	3. Sistem Memulai Permainan
	4. Sistem Menampilkan layar Permainan
Kondisi akhir	Permainan Dimainkan

2. Skenario *Use case* Ambil hambatan (R) Lampu

Tabel 3.14 Skenario *Use case* Ambil R Lampu

Identifikasi	
No	3
Nama	Ambil hambatan (R) Lampu
Tujuan	Untuk mengambil nilai R lampu di dalam R
Deskripsi	Proses untuk mengambil nilai R lampu
Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di area permainan
Aksi aktor	**Reaksi sistem**
1. Mengambil R Lampu	2.Inisialisasi nilai R lampu awal
	3. Update Nilai R Lampu
4.Menampilkan nilai R lampu yang telah di dapat	

Kondisi akhir	Aktor mendapatkan nilai R lampu bertambah

3. Skenario *Use case* Masukan R Lampu ke Switch lampu

Tabel 3.15 Skenario *Use case* Masukan R ke Switch lampu

Identifikasi	
No	4
Nama	Masukan R Lampu ke Switch lampu
Tujuan	Untuk memasukan nilai R lampu ke Switch lampu
Deskripsi	Proses untuk memasukan nilai R lampu ke Switch lampu
Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di area permainan
Aksi aktor	**Reaksi sistem**
1. Memasukan R Lampu	
	2. Proses Hitung tegangan dan arus yang masuk kedalam lampu
	3. Proses Fuzzifikasi
	4. Proses Pembentukan Rule
	5. Proses Inferensi
	6. Proses Defuzzifikasi
	7. Mengeluarkan Nilai Tegangan dan Arus yang masuk
	8. Mengeluarkan Aksi intensitas Lampu
Kondisi akhir	Sistem mengeluarkan Keputusan nilai intensitas cahaya lampu.

8. Skenario *Use case* Ambil R Pintu

Tabel 3.16 Skenario *Use case* Ambil R Pintu

Identifikasi	
No	5
Nama	Ambil R Pintu
Tujuan	Untuk mengambil nilai R pintu di dalam R
Deskripsi	Proses untuk mengambil nilai R pintu di dalam R

Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di dalam permainan
Aksi aktor	**Reaksi sistem**
1. Mengambil R Pintu	2. Inisialisasi nilai R pintu awal
	3. Update Nilai R Pintu
4.Menampilkan nilai R Pintu yang telah di dapat	
Kondisi akhir	Aktor mendapatkan nilai Rpintu bertambah

9. Skenario *Use case* Masukan R Pintu ke Switch Pintu

Tabel 3.17 Skenario *Use case* Masukan R ke Switch Pintu

Identifikasi	
No	6
Nama	Masukan R Pintu ke Switch pintu
Tujuan	Untuk memasukan nilai R pintu ke Switch pintu
Deskripsi	Proses untuk memasukan nilai R pintu ke Switch pintu
Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di dalam permainan
Aksi aktor	**Reaksi sistem**
1. Memasukan R Pintu	
	2. Proses Hitung tegangan dan arus yang masuk kedalam Pintu
	3. Proses Fuzzifikasi
	4. Proses Pembentukan Rule
	5. Proses Inferensi
	6. Proses Defuzzifikasi
	7. Mengeluarkan Nilai Tegangan dan Arus yang masuk
	8. Mengeluarkan Aksi Kecepatan pintu terbuka
	9. .Menampilkan Skor akhir

Kondisi akhir	Sistem mengeluarkan keputusan kecepatan pintu terbuka dan perhitungan skor akhir.

10. Skenario *Use case* Pengaturan suara

Tabel 3.18 Skenario *Use case* Pengaturan Suara

Identifikasi	
No	7
Nama	Pengaturan suara
Tujuan	Untuk mengatur suara pada permainan
Deskripsi	Proses untuk mengatur suara pada permainan
Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di Menu Utama
Aksi aktor	**Reaksi sistem**
	1. Menampilkan layar menu utama
2. Pilih Menu Pengaturan Suara	
	3. Menampilkan Pengaturan Suara
4. Memilih pengaturan On/Off	
	5. Aksi Suara On/Off
6.Menutup layar pengaturan suara	
	7.Menampilkan layar menu utama
Kondisi akhir	Sistem merespon pengaturan suara dari pemain lalu sistem kembali menampilkan menu utama

11. Skenario *Use case* Petunjuk

Tabel 3.19 Skenario *Use case* Baca petunjuk

Identifikasi	
No	8
Nama	Petunjuk
Tujuan	Untuk melihat petunjuk permainan
Deskripsi	Proses untuk melihat petunjuk permainan

Aktor	Pemain
Skenario utama	
Kondisi awal	Aktor berada di Menu Utama
Aksi aktor	**Reaksi sistem**
	1. Menampilkan layar menu utama
2. Memilih menu petunjuk	
	3. Menampilkan petunjuk permainan
4. Menutup tampilan petunjuk	
	5. Menampilkan layar menu utama
Kondisi akhir	Aktor berada di dalam layar menu utama

12. Skenario *Use case* Keluar permainan

Tabel 3.20 Skenario *Use case* Keluar permainan

Identifikasi	
No	9
Nama	Keluar permainan
Tujuan	Untuk keluar permainan
Deskripsi	Proses untuk keluar permainan
Aktor	Pemain
Skenario normal	
Kondisi awal	Aktor berada di menu utama
Aksi aktor	**Reaksi sistem**
	1. Menampilkan layar menu utama
2. Memilih Menu keluar	
	3. Menampilkan konfirmasi keluar
4. Memilih keluar	
	5. keluar dari Permainan
Kondisi akhir	Aktor keluar dari permainan

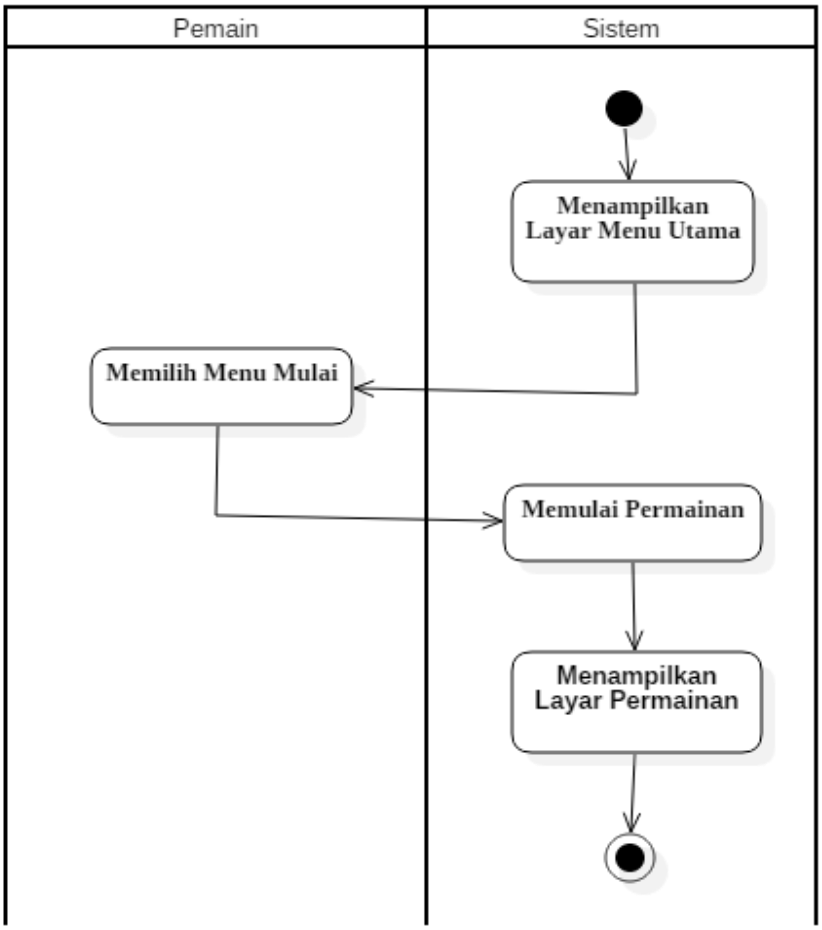
3.8.5 Activity Diagram

Diagram *activity* menggambarkan proses-proses yang terjadi mulai aktivitas dimulai sampai aktivitas berhenti pada sistem yang sedang dirancang.

Bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

1. *Activity Diagram* Mulai Permainan

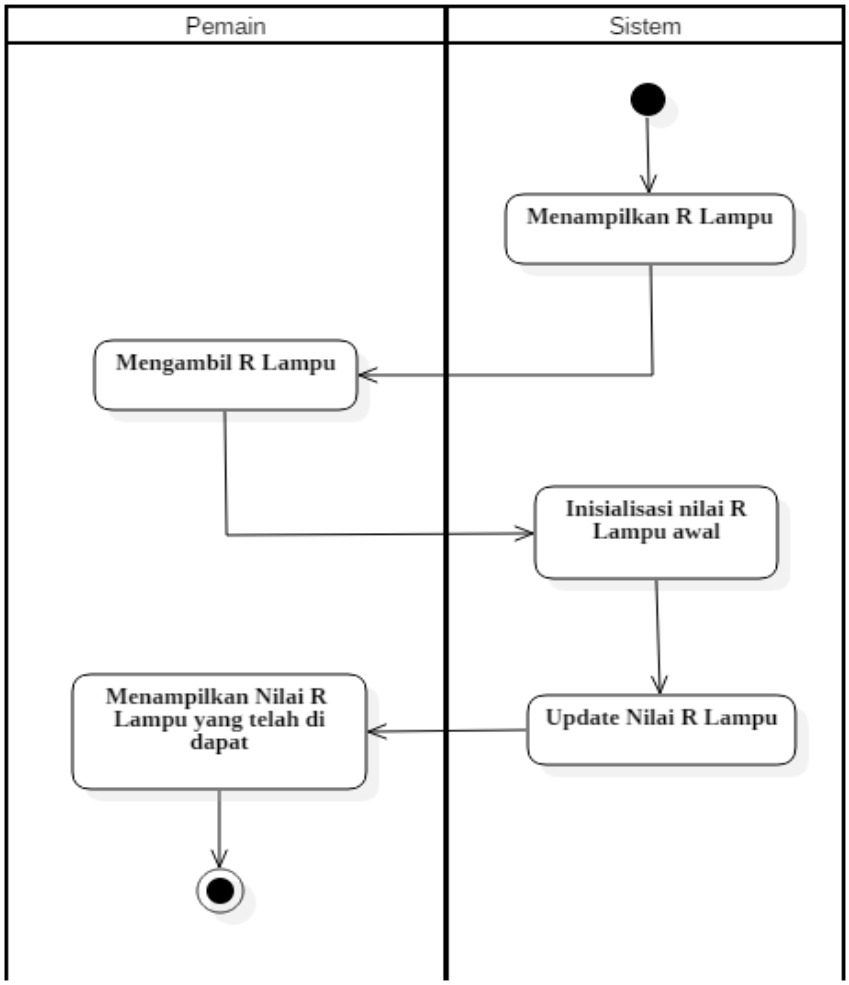
Proses ini terjadi saat pemain memilih *menu* mulai pada *Menu* utama. Selanjutnya,pemain akan masuk ke permainan Berikut adalah *activity diagram* mulai *game* yang terdapat pada gambar 3.26



Gambar 3.26 Activity Diagram Mulai Permainan

2. *Activity Diagram* Ambil Hambatan (R) Lampu

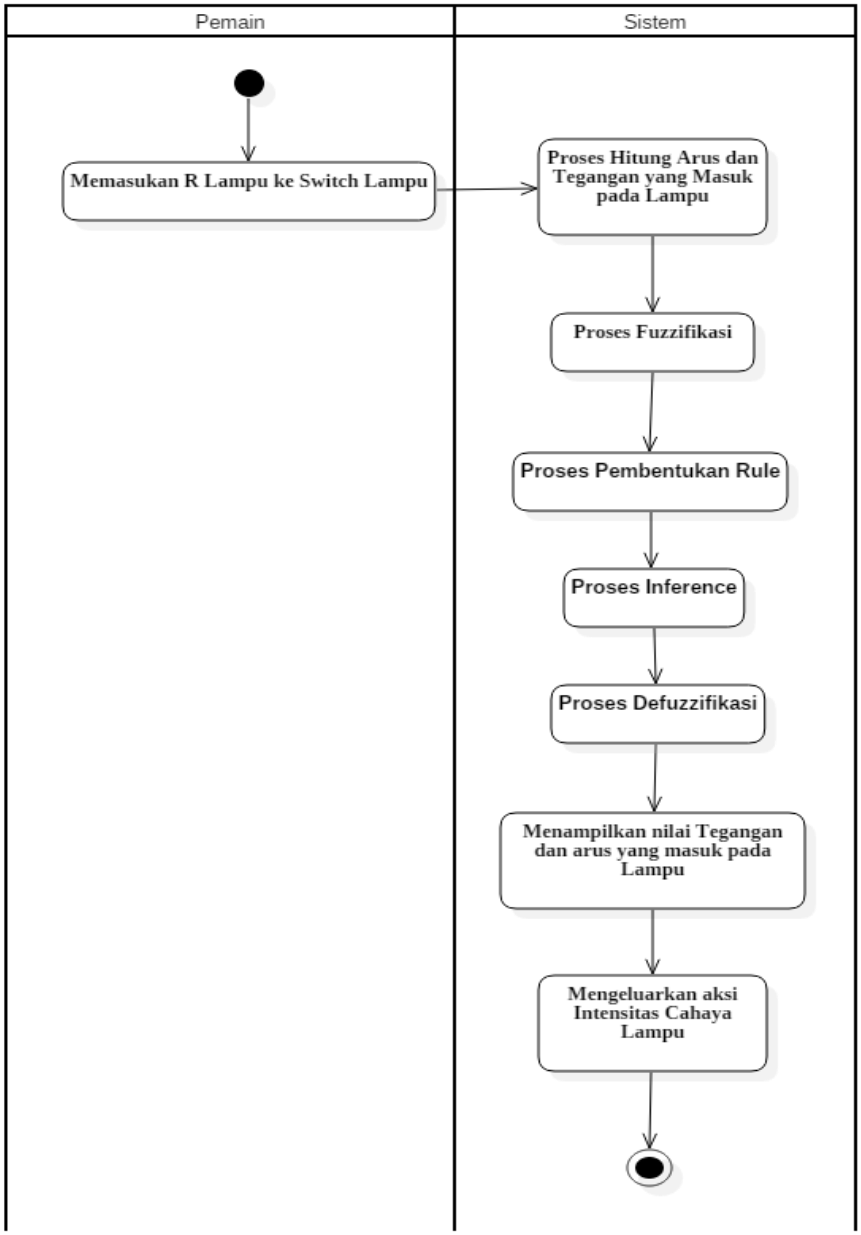
Proses ini terjadi saat pemain Mengambil hambatan (R) Lampu di area permainan. Selanjutnya, sistem akan menambahkan nilai R ketika R diambil. Berikut adalah *activity diagram* Ambil Hambatan (R) Lampu yang terdapat pada gambar 3.27.



Gambar 3.27 Activity Diagram Ambil R Lampu

3. *Activity Diagram* Masukan R Lampu ke Switch Lampu

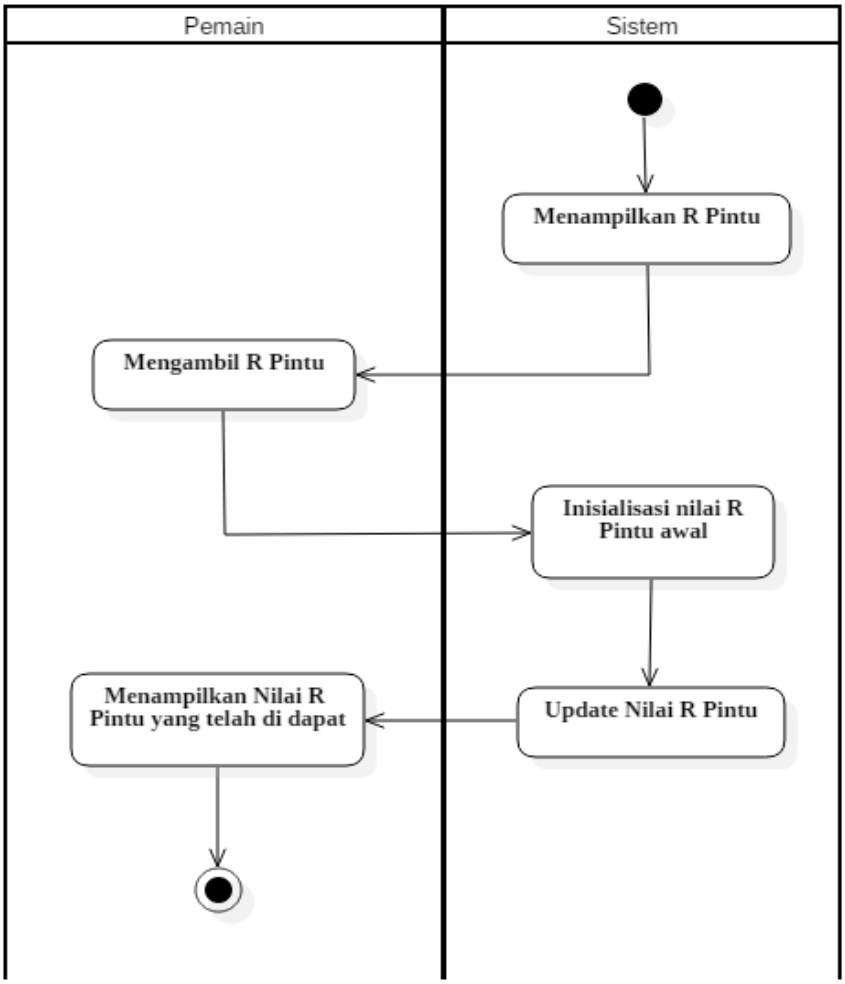
Proses ini terjadi saat pemain memasukan hambatan (R) Lampu ke dalam *Switch* lampu Selanjutnya, sistem akan menghitung tegangan dan arus yang masuk lalu menghitung nilai keanggotaan fuzzy dan mengeluarkan aksi lampu. Berikut adalah *activity diagram* Masukan R Pintu ke Switch lampu yang terdapat pada gambar 3.28.



Gambar 3.28 Activity Diagram Masukan R Lampu ke Switch Lampu

4. *Activity Diagram* Ambil Hambatan (R) Pintu

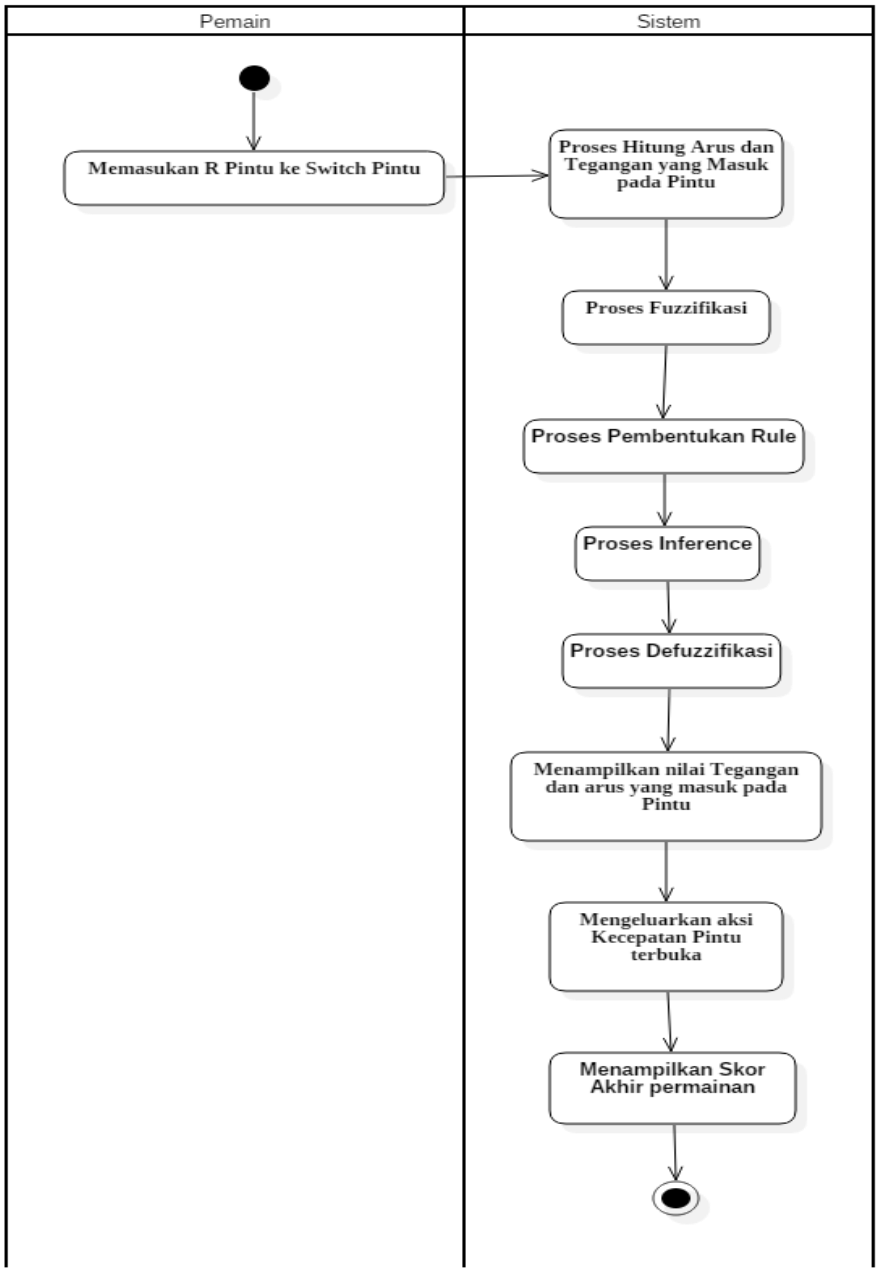
Proses ini terjadi saat pemain Mengambil hambatan (R) Pintu di area permainan. Selanjutnya, sistem akan menambahkan nilai R ketika R diambil. Berikut adalah *activity diagram* Ambil Hambatan (R) Pintu yang terdapat pada gambar 3.29.



Gambar 3.29 Activity Diagram Ambil R Pintu

5. *Activity Diagram* Masukan R Pintu ke Switch Pintu

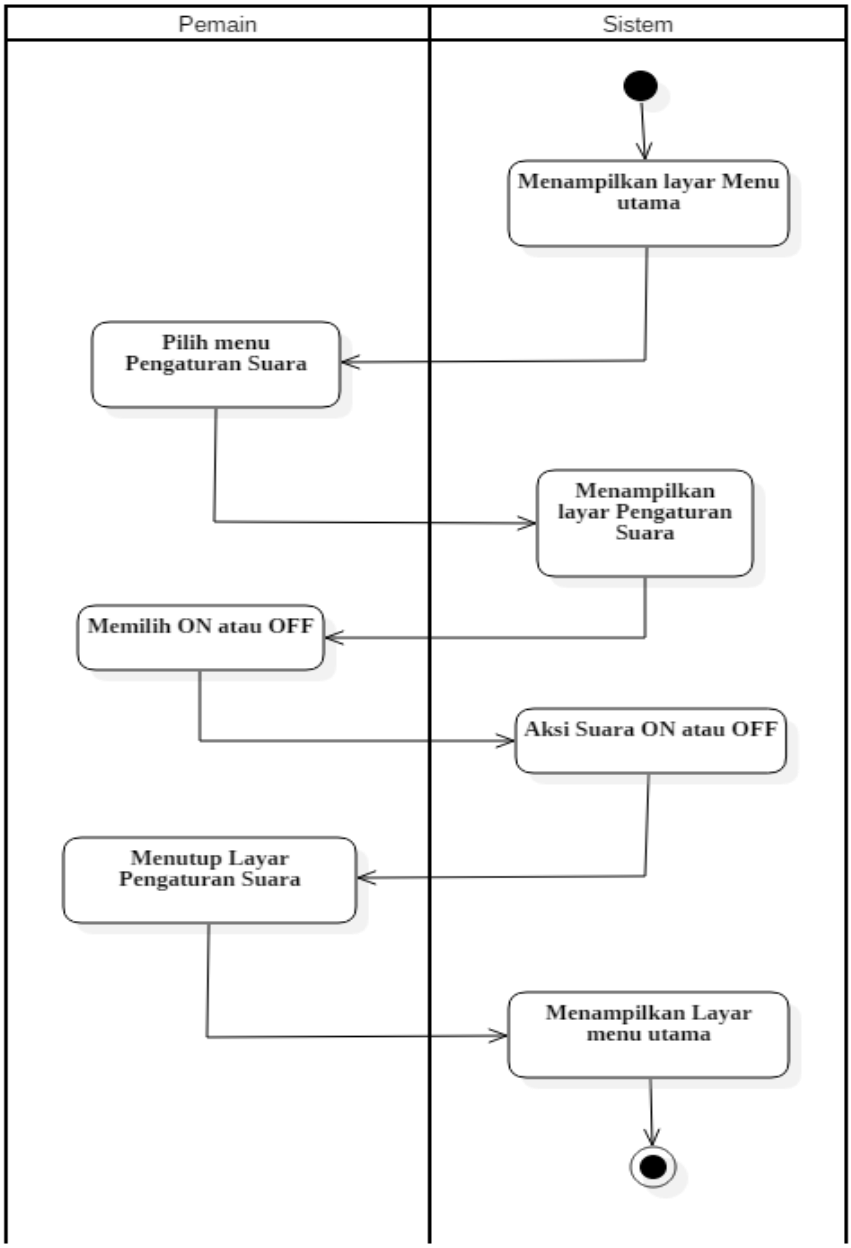
Proses ini terjadi saat pemain memasukan hambatan (R) Pintu ke dalam *Switch* pintu Selanjutnya, sistem akan menghitung tegangan dan arus yang masuk dan menghitung nilai keanggotaan fuzzy dan mengeluarkan aksi Pintu serta skor akhir. Berikut adalah *activity diagram* Masukan R Pintu ke Switch Pintu yang terdapat pada gambar 3.30.



Gambar 3.30 Activity Diagram Masukan R Pintu ke Switch Pintu

6. *Activity Diagram* Pengaturan Suara

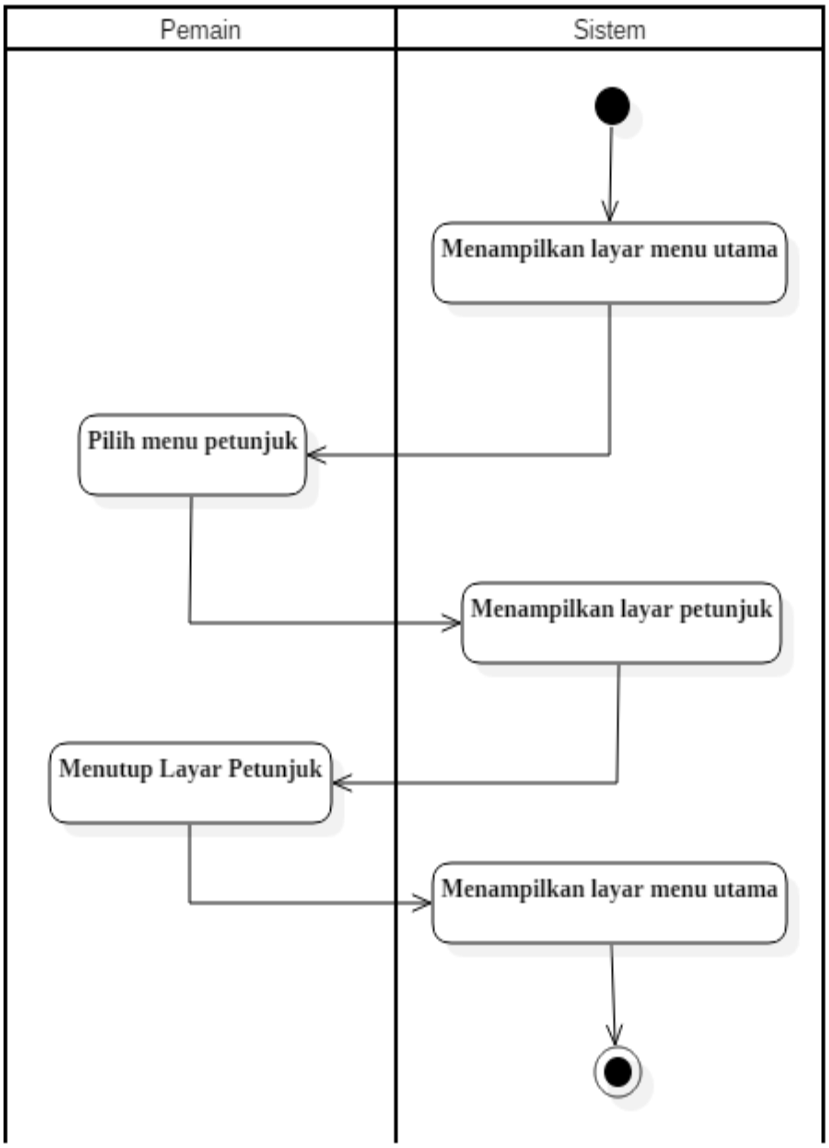
Proses ini terjadi saat pemain memilih pengaturan suara on/off Selanjutnya,sistem akan mengekseskusi pilihan pengaturan. Berikut adalah *activity diagram* Pengaturan Suara yang terdapat pada gambar 3.31.



Gambar 3.31 Activity Diagram Pengaturan Suara

7. *Activity Diagram* Baca Petunjuk

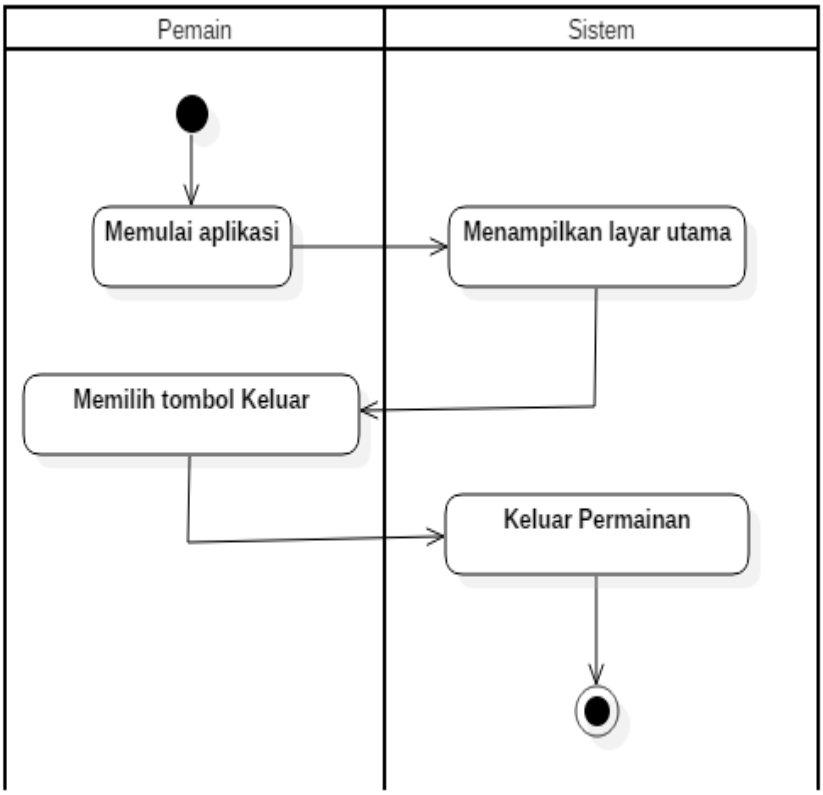
Proses ini terjadi saat pemain memilih Menu petunjuk Selanjutnya, sistem akan menampilkan petunjuk permainan. Berikut adalah *activity diagram* petunjuk yang terdapat pada gambar 3.32.



Gambar 3.32 Activity Diagram Petunjuk

8. *Activity Diagram* Keluar

Proses ini terjadi saat pemain memilih Menu Keluar Selanjutnya, sistem akan mengeluarkan pemain dari sistem. Berikut adalah *activity diagram* petunjuk yang terdapat pada gambar 3.33.



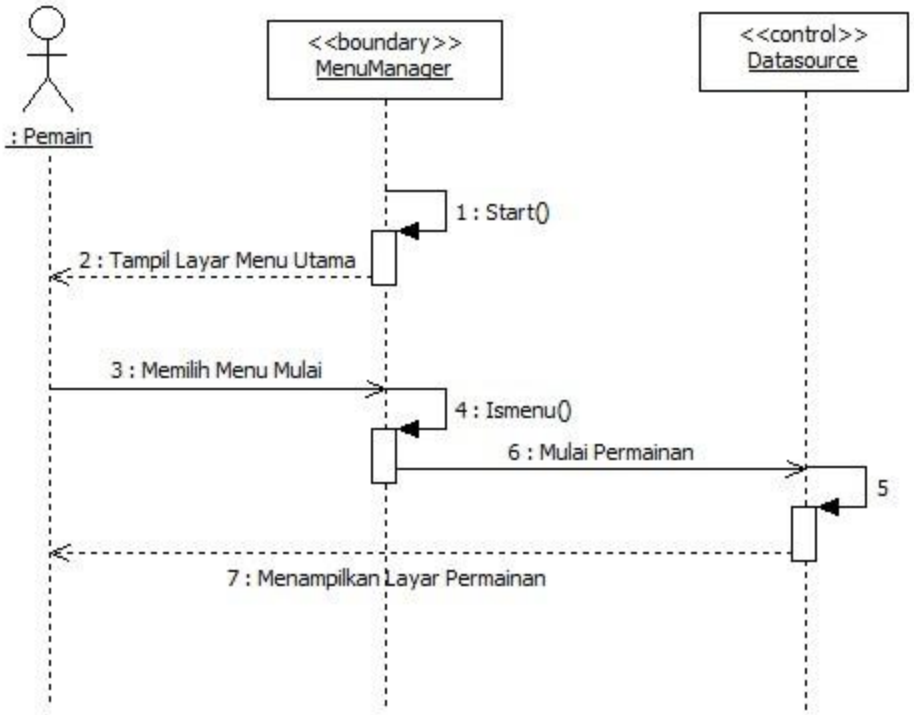
Gambar 3.33 Activity Diagram Keluar

3.8.6 Sequence Diagram

Sequence Diagram merupakan penggambaran dari interaksi antar objek di dalam dan sekitar sistem (termasuk pengguna, *display*, dan sebagainya). *Sequence Diagram* terdiri atas dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait). Berikut adalah beberapa *diagram sequence* dari *game* yang akan dibangun :

1. *Sequence Diagram* Mulai *Game*

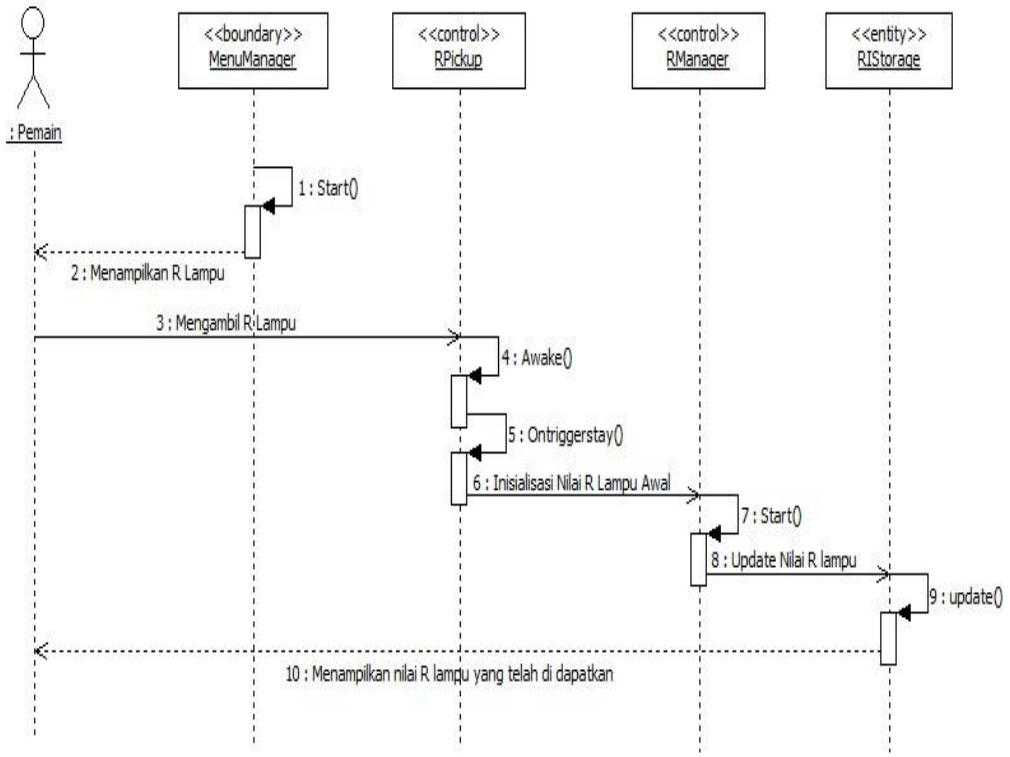
Berikut merupakan sequence diagram dari Mulai *Game* yang berisikan *Class* MenuManager dan DataSource, yang terdapat pada gambar 3.34.



Gambar 3.34 *Sequence Diagram* Mulai *Game*

2. *Sequence Diagram* Mengambil R Lampu

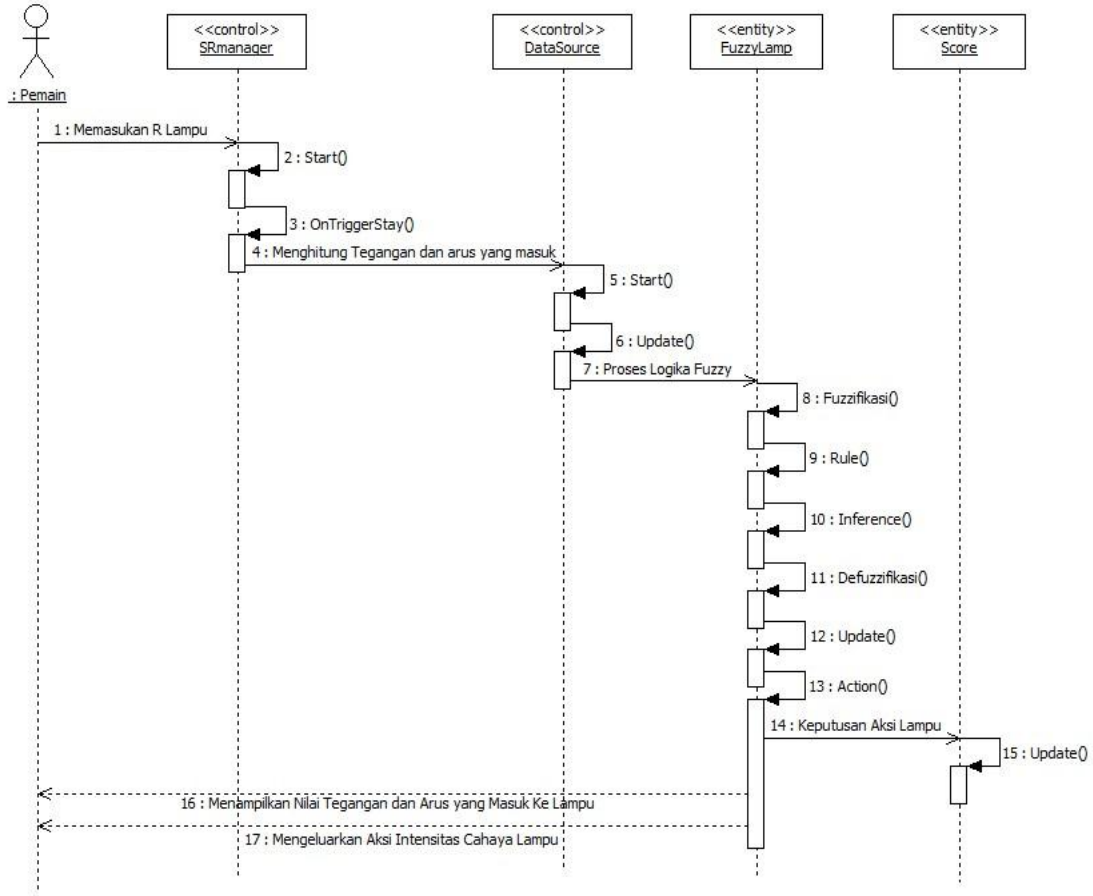
Berikut merupakan sequence diagram dari Mengambil R Lampu yang berisikan *Class* MenuManager, RPickup, RIStorage dan RManager yang terdapat pada gambar 3.35.



Gambar 3.35 *Sequence Diagram* Mengambil R Lampu

3. *Sequence Diagram* Memasukan R Lampu ke Switch Lampu

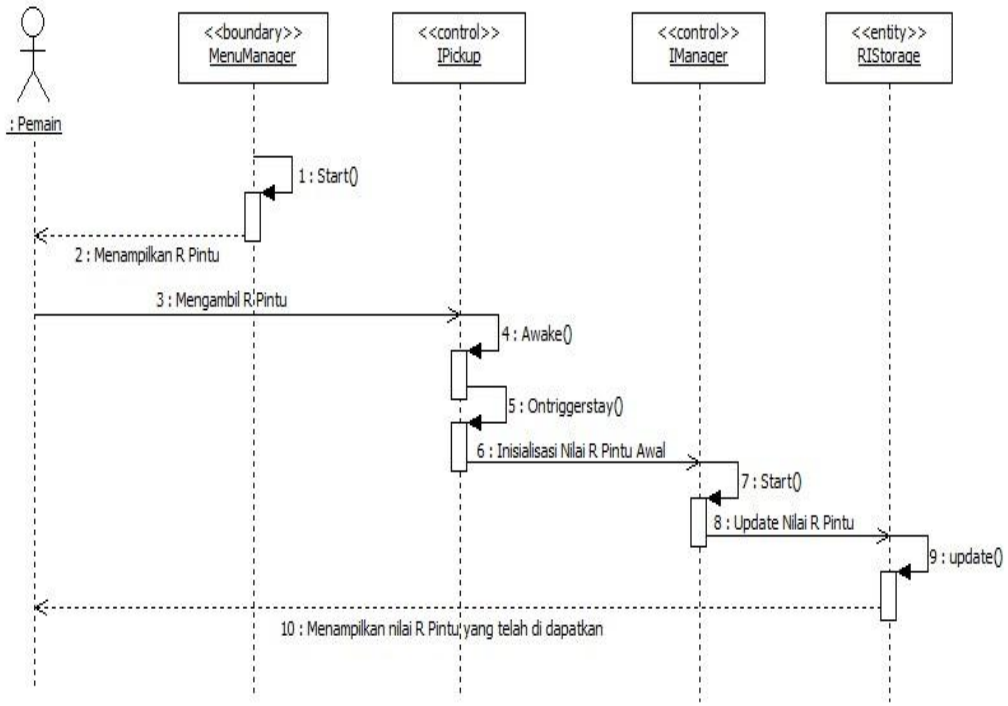
Berikut merupakan sequence diagram dari Memasukan R Lampu ke Switch Lampu yang berisikan *Class* SRManager, DataSource, FuzzyLamp dan Score, yang terdapat pada gambar 3.36.



Gambar 3.36 *Sequence Diagram* Memasukan R Lampu ke Switch Lampu

4. *Sequence Diagram* Mengambil R Pintu

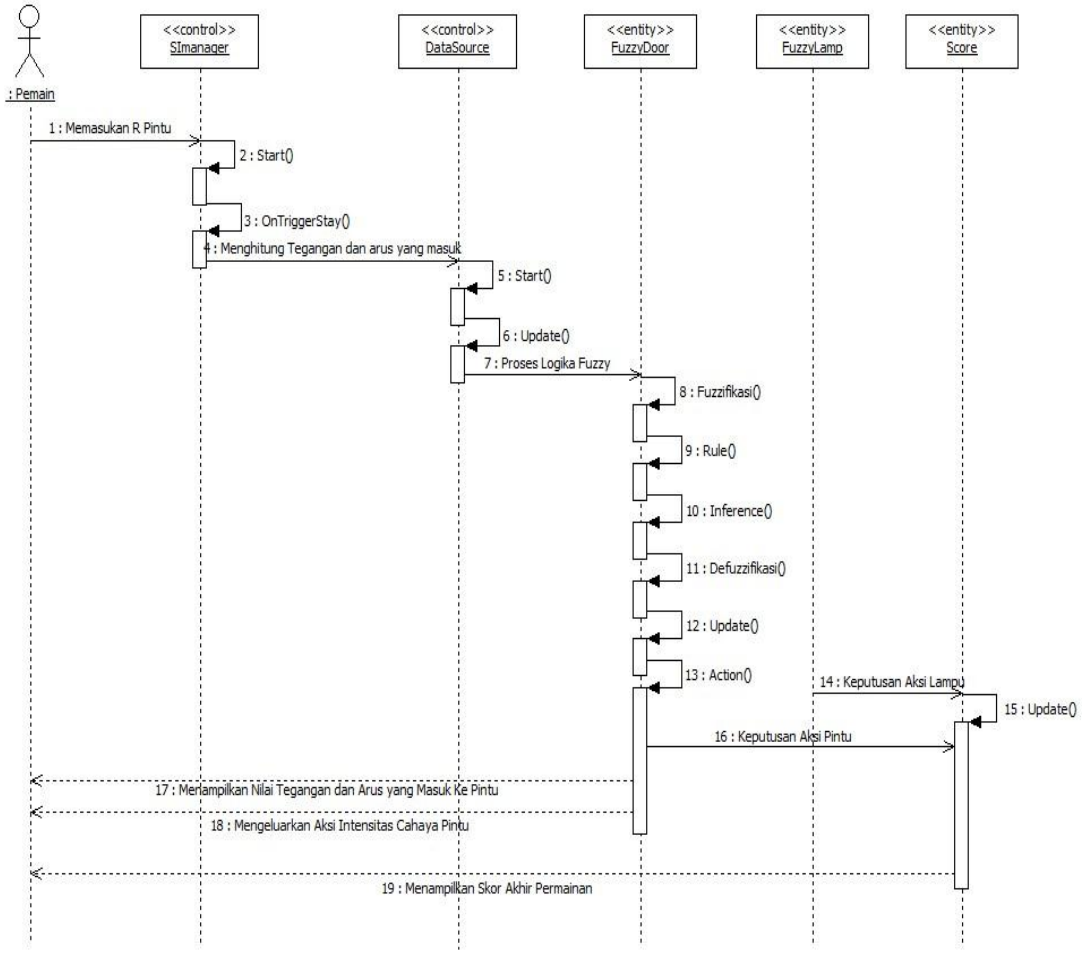
Berikut merupakan sequence diagram dari Mengambil R Pintu yang berisikan *Class* MenuManager, Rpickup, RIStorage dan RManager, yang terdapat pada gambar 3.37.



Gambar 3.37 *Sequence Diagram* Mengambil R Pintu

5. *Sequence Diagram* Memasukan R Pintu ke Switch Pintu

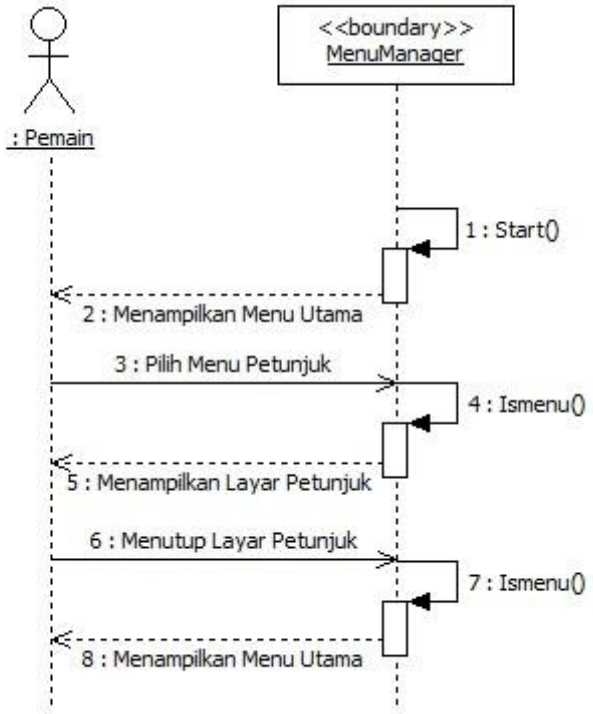
Berikut merupakan sequence diagram dari Memasukan R Pintu ke Switch Pintu yang berisikan *Class* SIManager, DataSource, FuzzyDoor dan Score, yang terdapat pada gambar 3.38.



Gambar 3.38 *Sequence Diagram* Memasukan R Pintu ke Switch Pintu

6. *Sequence Diagram* Baca Petunjuk

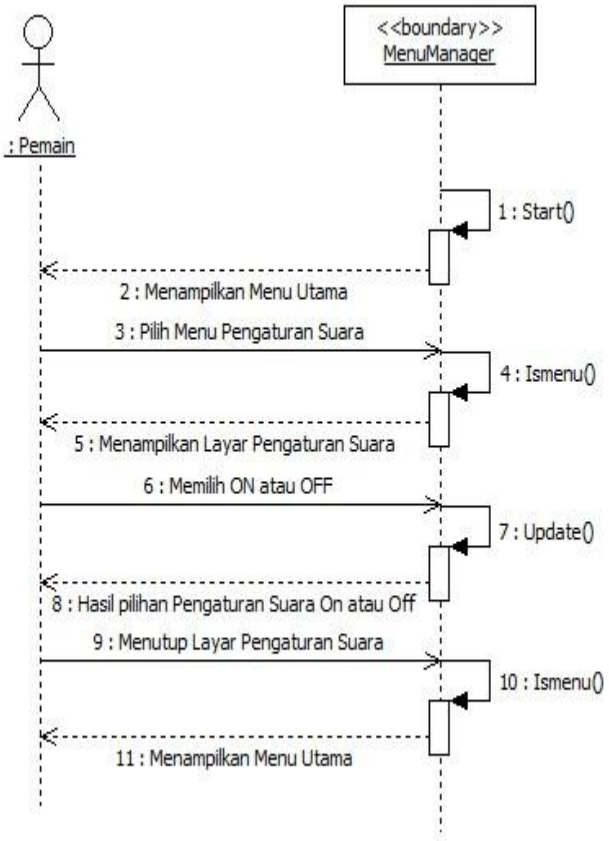
Berikut merupakan sequence diagram dari Baca Petunjuk yang berisikan *Class* MenuManager, yang terdapat pada gambar 3.39.



Gambar 3.39 *Sequence Diagram* Petunjuk

7. *Sequence Diagram* Pengaturan Suara

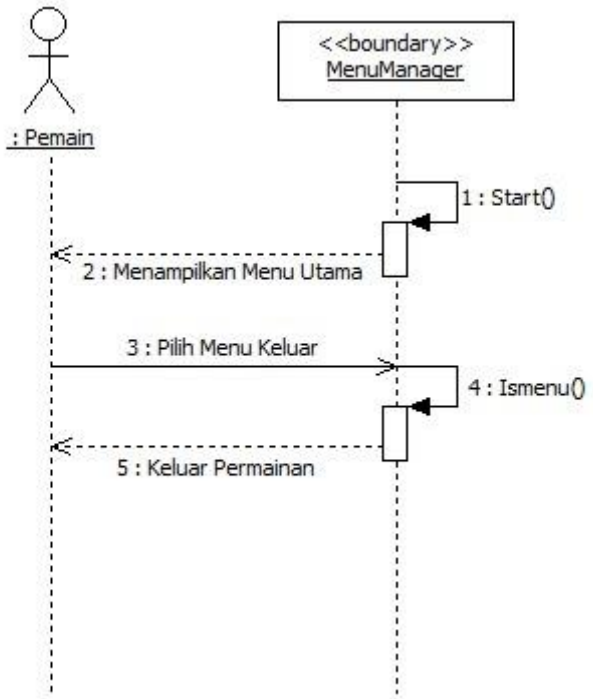
Berikut merupakan sequence diagram dari Pengaturan Suara yang berisikan *Class* MenuManager, yang terdapat pada gambar 3.40.



Gambar 3.40 *Sequence Diagram* Pengaturan Suara

8. *Sequence Diagram* Keluar

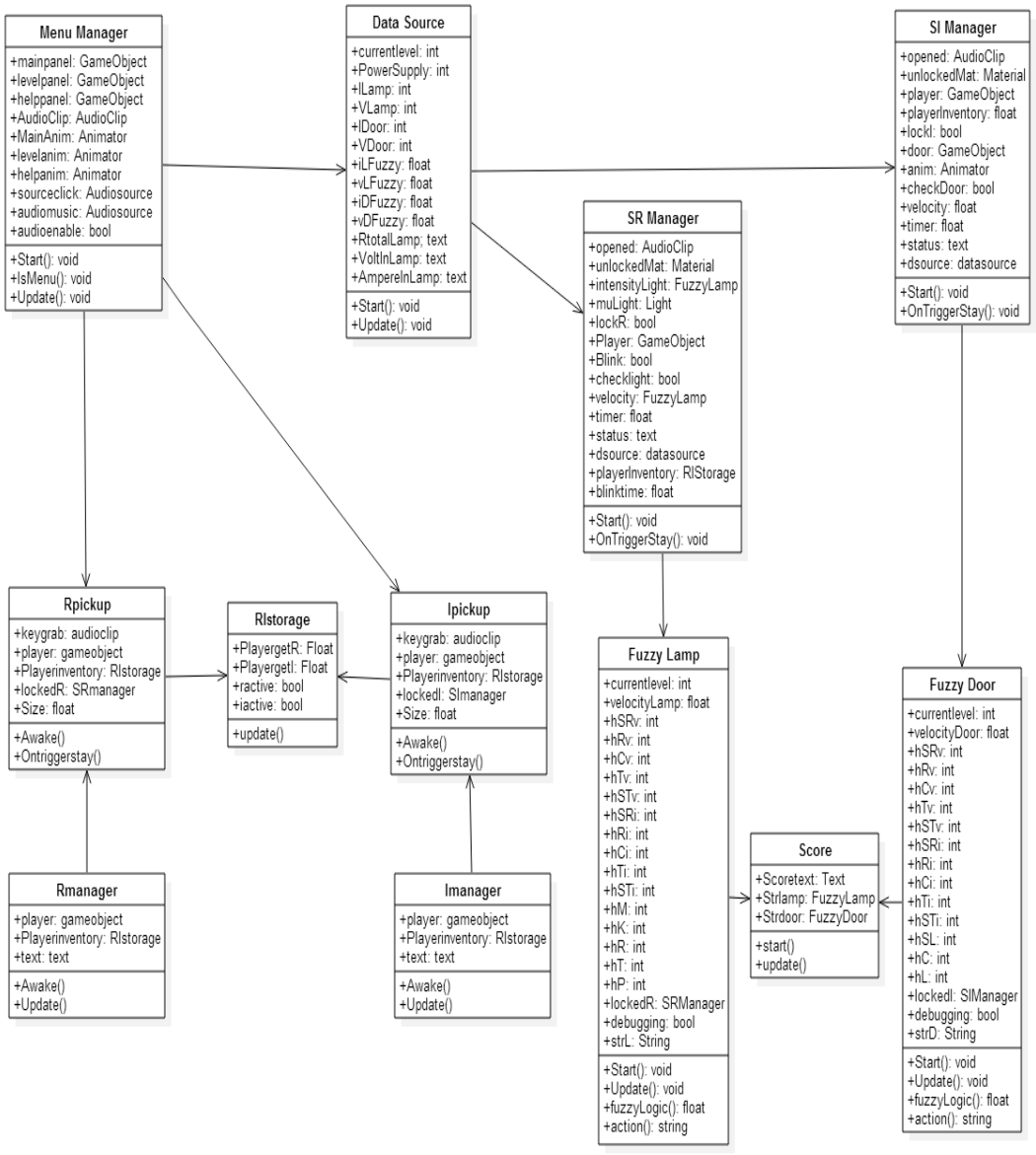
Berikut merupakan sequence diagram dari Keluar yang berisikan *Class* MenuManager, yang terdapat pada gambar 3.41.



Gambar 3.41 *Sequence Diagram* Keluar

3.8.7 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Berikut adalah *Class diagram* yang terdapat pada Gambar 3.42.



Gambar 3.42 *Class Diagram*

3.9 Perancangan Sistem

Perancangan adalah suatu bagian dari metodologi pengembangan suatu perangkat lunak yang dilakukan setelah tahapan untuk memberikan gambaran secara terperinci. Dalam tahap ini digambarkan rancangan sistem yang akan dibangun sebelum dilakukan pengkodean ke dalam suatu bahasa pemrograman.

3.9.1 Perancangan Class

Merupakan kumpulan class yang ada pada sistem yang dirancang, dan berfungsi untuk membangun sistem. Dan berikut ini adalah penjelasan nya.

Tabel 3.21 Deskripsi Class

No	Class	Deskripsi	Tipe Class
1	Menu Manager	*Class* yang berisikan *method* memulai permainan, pengaturan suara, petunjuk dan keluar.	*Boundary*
2	Data Source	*Class* yang berisikan *method* perhitungan tegangan dan arus yang masuk serta spesifikasi lampu dan pintu.	*Control*
3	SR Manager	*Class* yang berisikan *method control* memasukan R Lampu ke dalam *switch* lampu.	*Control*
4	SI Manager	*Class* yang berisikan *method control* memasukan R Pintu ke dalam *switch* Pintu.	*Control*
5	Fuzzy Lamp	*Class* yang berisikan *method* logika *fuzzy lampu*	*Entity*
6	Fuzzy Door	*Class* yang berisikan *method* logika *fuzzy* pintu	*Entity*
7	Score	*Class* yang berisikan *method* menentukan skor akhir	*Entity*
8	Rpickup	*Class* yang berisikan *method* mengambil R Lampu	*Control*
9	Rmanager	*Class* yang berisikan *method* menampilkan nilai R yang di dapat dari mengambil R Lampu	*Control*
10	RIstorage	*Class* yang berisikan *method* inisialisasi nilai R awal	*Control*

11	Ipickup	*Class* yang berisikan *method* mengambil R Pintu	*Control*
12	Imanager	*Class* yang berisikan *method* menampilkan nilai R yang di dapat dari mengambil R pintu	*Control*

Dalam *Class* diagram terdapat beberapa atribut di dalam nya. Adapun penjelasan nya adalah sebagai berikut :

1. *Class* MenuManager

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* MenuManager pada Tabel 3.22.

Tabel 3.22 Perancangan Class MenuManager

No	Nama Atribut	Tipe Data	Visibility
1	mainpanel	*GameObject*	*Public*
2	levelpanel	*GameObject*	*Public*
3	helppanel	*GameObject*	*Public*
4	audioclip	*audioclip*	*Public*
5	mainanim	*animator*	*Private*
6	levelanim	*animator*	*Private*
7	helpanim	*animator*	*Private*
8	sourceclick	*audiosource*	*Private*
9	audiomusic	*audiosource*	
10	audioenable	bool	*Public*

2. *Class* DataSource

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* DataSource pada Tabel 3.23.

Tabel 3.23 Perancangan Class DataSource

No	Nama Atribut	Tipe Data	Visibility
1	currentlevel	*int*	*Public*
2	powersupply	*int*	*Public*
3	ILamp	*int*	*Private*
4	Vlamp	*int*	*Private*

5	Idoor	*int*	*Private*
6	Vdoor	*int*	*Private*
7	iLFuzzy	*Float*	*Public*
8	vLFuzzy	*Float*	*Public*
9	iDFuzzy	*Float*	*Public*
10	vDFuzzy	*Float*	*Public*
11	RtotalLamp	*Text*	
12	VoltInLamp	*Text*	
13	AmpereInLamp	*Text*	
14	RtotalDoor	*Text*	
15	VoltInDoor	*Text*	
16	AmpereInDoor	*Text*	

3. *Class* FuzzyDoor

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* FuzzyDoor pada Tabel 3.24.

Tabel 3.24 Perancangan Class FuzzyDoor

No	Nama Atribut	Tipe Data	Visibility
1	currentlevel	*int*	*Public*
2	velocityDoor	*Float*	*Public*
3	hSRv	*int*	*Private*
4	hRv	*int*	*Private*
5	hCv	*int*	*Private*
6	hTv	*int*	*Private*
7	hSTv	*int*	*Private*
8	hSRi	*int*	*Private*
9	hRi	*int*	*Private*
10	hCi	*int*	*Private*
11	hTi	*int*	*Private*
12	hSTi	*int*	*Private*

13	hSL	*int*	*Private*
14	hC	*int*	*Private*
15	hL	*int*	*Private*
16	lockedI	*SIManager*	*Private*
17	Debugging	*Bool*	*Private*
18	strD	*String*	*Public*

4. *Class* FuzzyLamp

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* FuzzyLamp pada Tabel 3.25.

Tabel 3.25 Perancangan Class FuzzyLamp

No	Nama Atribut	Tipe Data	Visibility
1	currentlevel	*int*	*Public*
2	intensityLightResult	*Float*	*Public*
3	hSRv	*int*	*Private*
4	hRv	*int*	*Private*
5	hCv	*int*	*Private*
6	hTv	*int*	*Private*
7	hSTv	*int*	*Private*
8	hSRi	*int*	*Private*
9	hRi	*int*	*Private*
10	hCi	*int*	*Private*
11	hTi	*int*	*Private*
12	hSTi	*int*	*Private*
13	hM	*int*	*Private*
14	hK	*int*	*Private*
15	hR	*int*	*Private*
16	hT	*int*	*Private*
17	hP	*int*	*Private*
18	lockedR	*SRManager*	*Private*

19	Debugging	*Bool*	*Private*
20	strL	*String*	*Public*

5. *Class* RIStorage

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* RIStorage pada Tabel 3.26.

Tabel 3.26 Perancangan Class RIStorage

No	Nama Atribut	Tipe Data	Visibility
1	playerGetR	*Float*	*Public*
2	playerGetI	*Float*	*Public*
3	rActive	*Bool*	*Public*
4	iActive	*Bool*	*Public*

6. *Class* Imanager

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* Imanager pada Tabel 3.27.

Tabel 3.27 Perancangan Class Imanager

No	Nama Atribut	Tipe Data	Visibility
1	player	*GameObject*	*Private*
2	playerInventory	*RIStorage*	*Private*
3	Text	*Text*	

7. *Class* Ipickup

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* Ipickup pada Tabel 3.28.

Tabel 3.28 Perancangan Class Ipickup

No	Nama Atribut	Tipe Data	Visibility
1	keygrab	*AudioClip*	*Public*
2	player	*GameObject*	*Private*
3	playerInventory	*RIStorage*	*Private*

4	lockedI	*SIManager*	
5	Size	*Float*	*Public*

8. *Class* Rmanager

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* Rmanager pada Tabel 3.29.

Tabel 3.29 Perancangan Class Rmanager

No	Nama Atribut	Tipe Data	Visibility
1	Player	*GameObject*	*Private*
2	playerInventory	*RIStorage*	*Private*
3	Text	*Text*	

9. *Class* RPickup

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* RPickup pada Tabel 3.30.

Tabel 3.30 Perancangan Class RPickup

No	Nama Atribut	Tipe Data	Visibility
1	keygrab	*AudioClip*	*Public*
2	player	*GameObject*	*Private*
3	playerInventory	*RIStorage*	*Private*
4	lockedI	*SRManager*	
5	Size	*Float*	*Public*

10. *Class* SIManager

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* SIManager pada Tabel 3.31.

Tabel 3.31 Perancangan Class SIManager

No	Nama Atribut	Tipe Data	Visibility
1	Opened	*AudioClip*	*Public*
2	unlockedMat	*Material*	*Public*
3	Player	*GameObject*	*Private*

4	PlayerInventory	RIStorage	*Private*
5	lockI	*Bool*	*Public*
6	Door	*GameObject*	*Public*
7	Anim	*Animator*	*Private*
8	checkDoor	*Bool*	
9	velocity	*FuzzyDoor*	*Private*
10	Timer	*Float*	
11	status	*Text*	
12	dsource	*Datasource*	

11. *Class* SRManager

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* SRManager pada Tabel 3.32.

Tabel 3.32 Perancangan Class SRManager

No	Nama Atribut	Tipe Data	Visibility
1	Opened	*AudioClip*	*Public*
2	unlockedMat	*Material*	*Public*
3	intensityLight	*FuzzyLamp*	*Private*
4	myLight	Light	*Public*
5	lockR	*Bool*	*Public*
6	player	*GameObject*	*Private*
7	Blink	*Bool*	*Private*
8	checkLight	*Bool*	
9	velocity	*FuzzyDoor*	*Private*
10	Timer	*Float*	
11	status	*Text*	
12	dsource	*Datasource*	
13	PlayerInventory	*RIStorage*	*Private*
14	blinkTime	*Float*	

12. *Class* Score

Berikut merupakan deskripsi mengenai nama atribut, tipe data dan *visibility* dari *Class* Score pada Tabel 3.33.

Tabel 3.33 Perancangan Class Score

No	Nama Atribut	Tipe Data	Visibility
1	Scoretext	*Text*	
2	strLamp	*FuzzyLamp*	
3	strDoor	*FuzzyDoor*	

3.9.2 Perancangan Method

Merupakan kumpulan method yang ada pada sistem yang dirancang, dan berfungsi untuk menjalankan sistem. Berikut adalah deskripsi mengenai method pada tabel 3.34.

Tabel 3.34 Deskripsi Method

No	Method	Class	Deskripsi
1	Start()	MenuManager	Method yang berfungsi untuk memulai proses sistem di class MenuManager
2	Ismenu()		Method yang berfungsi untuk akses pemilihan menu dari menu utama
3	Update()		Method yang berfungsi untuk meng *update* pilihan pengaturan suara
4	Awake()	RPickup	Method yang berfungsi untuk memulai proses sistem di class RPickup
5	OntriggerStay()		Method yang berfungsi untuk proses mengambil R Lampu
6	Awake()	IPickup	Method yang berfungsi untuk memulai proses sistem di class IPickup
7	OntriggerStay()		Method yang berfungsi untuk proses mengambil R Pintu
8	Start()	RIStorage	Method yang berfungsi untuk Method yang berfungsi untuk memulai proses sistem dan insialisasi nilai awal R di class RIStorage
9	Awake()	Rmanager	Method yang berfungsi untuk memulai proses sistem di class Rmanager
10	Update()	Rmanager	Method yang berfungsi untuk *update* nilai

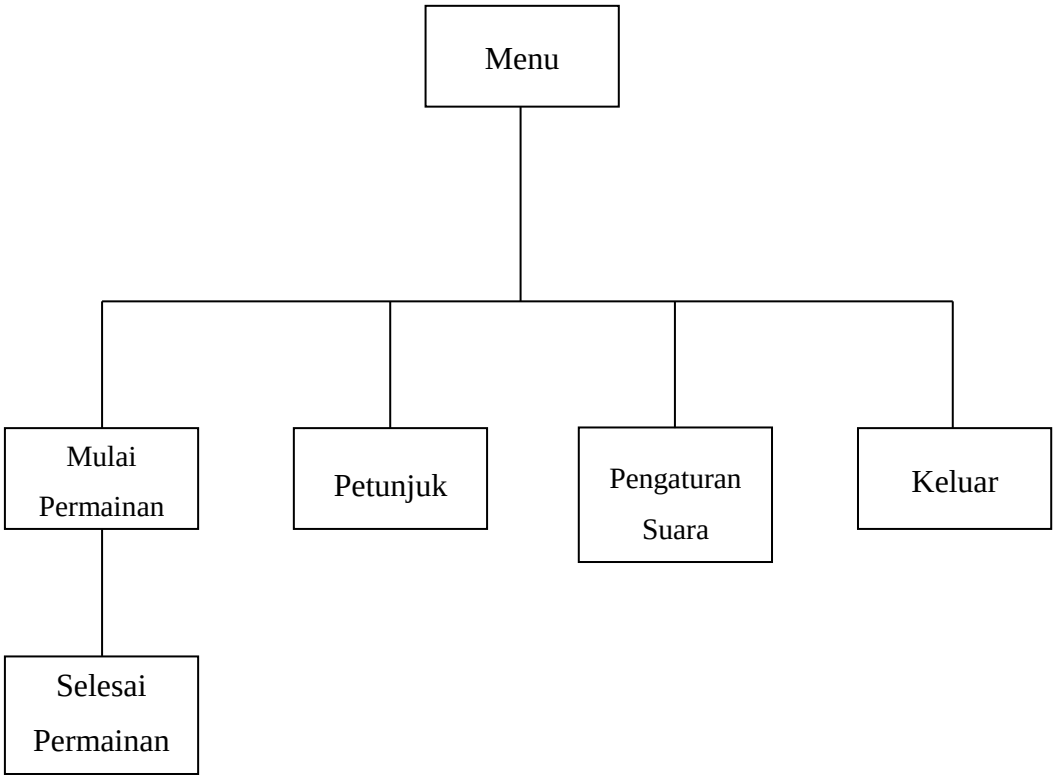
			R lampu
11	Awake()	Imanager	Method yang berfungsi untuk memulai proses sistem di class RPickup
12	Update()		Method yang berfungsi untuk *update* nilai R Pintu
13	Start()	SRmanager	Method yang berfungsi untuk memulai proses sistem di class SRmanager
14	OntriggerStay()		Method yang berfungsi untuk memicu aksi lampu ketika R Lampu dimasukan ke Switch Lampu
15	Start()	SImanager	Method yang berfungsi untuk memulai proses sistem di class SImanager
16	OntriggerStay()		Method yang berfungsi untuk memicu aksi pintu ketika R Pintu dimasukan ke Switch Lampu
17	Start()	DataSource	Method yang berfungsi untuk memulai proses sistem di class DataSource
18	Update()		Method yang berfungsi untuk menghitung tegangan dan arus yang masuk pada lampu atau pintu.
19	Start()	FuzzyLamp	Method yang berfungsi untuk memulai proses sistem di class FuzzyLamp
20	Fuzzifikasi()		Method yang berfungsi untuk proses untuk mengubah suatu masukan dari bentuk tegas (crisp) menjadi fuzzy (variabel linguistik) yang biasanya disajikan dalam bentuk himpunan-himpunan fuzzy dengan suatu fungsi kenggotaannya masing-masing.
21	Rule()		Method yang berfungsi untuk

			Pembentukan rule yang akan di gunakan pada proses inference
22	Inference()		Method yang berfungsi untuk memperhitungkan semua aturan yang ada dalam basis pengetahuan. Hasil dari proses inference direpresentasikan oleh suatu fuzzy set untuk setiap variabel bebas
23	Defuzzifikasi()		Method yang berfungsi untuk proses mengubah output *fuzzy* yang diperoleh dari mesin inferensi menjadi nilai tegas menggunakan fungsi keanggotaan yang sesuai dengan saat dilakukan *fuzzyfikasi*
24	Update()		Method yang berfungsi untuk mengupdate nilai Ztotal Lampu menjadi keputusan nyala lampu
25	Action()	FuzzyLamp	Method yang berfungsi pilihan nilai intensitas cahaya lampu berdasarkan kategori mati, kedip, redup, terang dan pecah
26	Start()	FuzzyDoor	Method yang berfungsi untuk memulai proses sistem di class FuzzyDoor
27	Fuzzifikasi()		Method yang berfungsi untuk memulai proses sistem di class FuzzyLamp
28	Rule()		Method yang berfungsi untuk proses untuk mengubah suatu masukan dari bentuk tegas (crisp) menjadi fuzzy (variabel linguistik) yang biasanya disajikan dalam bentuk himpunan-himpunan fuzzy dengan suatu fungsi kenggotaannya masing-masing.

29	Inference()		Method yang berfungsi untuk Pembentukan rule yang akan di gunakan pada proses inference
30	Defuzzifikasi()		Method yang berfungsi untuk memperhitungkan semua aturan yang ada dalam basis pengetahuan. Hasil dari proses inference direpresentasikan oleh suatu fuzzy set untuk setiap variabel bebas
28	Update()		Method yang berfungsi untuk mengupdate nilai Ztotal Lampu menjadi keputusan kecepatan pintu terbuka
29	Action()		Method yang berfungsi pilihan nilai kecepatan pintu terbuka berdasarkan kategori lambat, cepat dan sangat lambat
30	Start()	Score	Method yang berfungsi untuk memulai proses sistem di class Score
28	Update()		Method yang berfungsi menentukan Skor akhir berdasarkan hasil keputusan nilai intensitas cahaya lampu dan kecepatan pintu terbuka

3.9.3 Perancangan Struktur Menu

Struktur menu bertujuan untuk menjelaskan perancangan menu proses yang dapat digunakan oleh pengguna. Struktur menu secara umum yang dibuat dalam sistem ini terdiri dari menu-menu dan sub menunya. Struktur menu *game* resistansi listrik dapat dilihat pada gambar 3.43.



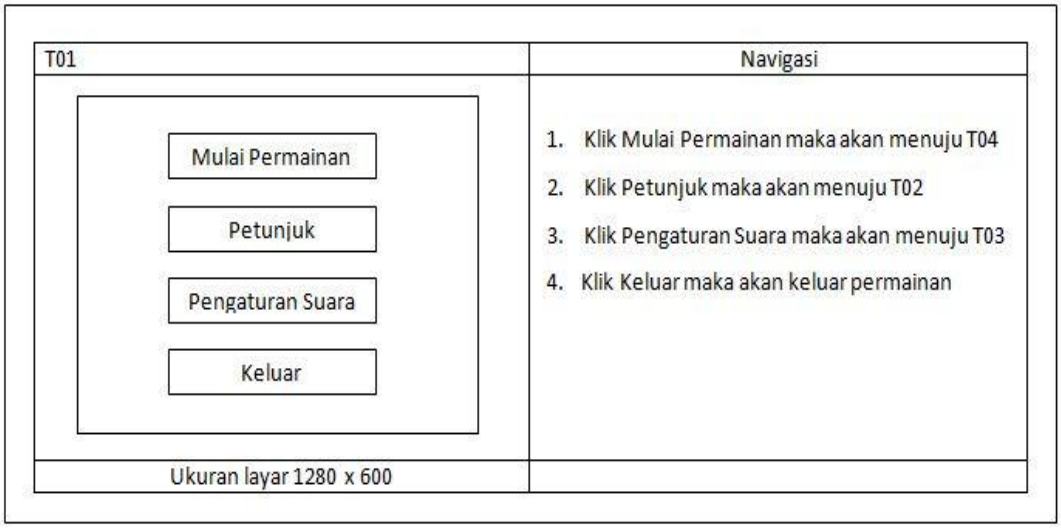
Gambar 3.43 Struktur Menu

3.9.4 Perancangan Antar Muka

Perancangan antarmuka dibuat unutk menggambarkan tampilan program yag akan digunakan oleh pengguna untuk berinteraksi dengan aplikasi yang dibuat. Perancangan dibuat berdasarkan tampilan antarmuka baik masukan (*input*) maupun keluaran (*output*) yang akan dihasilkan saat aplikasi diimplementasikan.

1. **Perancangan Antarmuka Menu Utama**

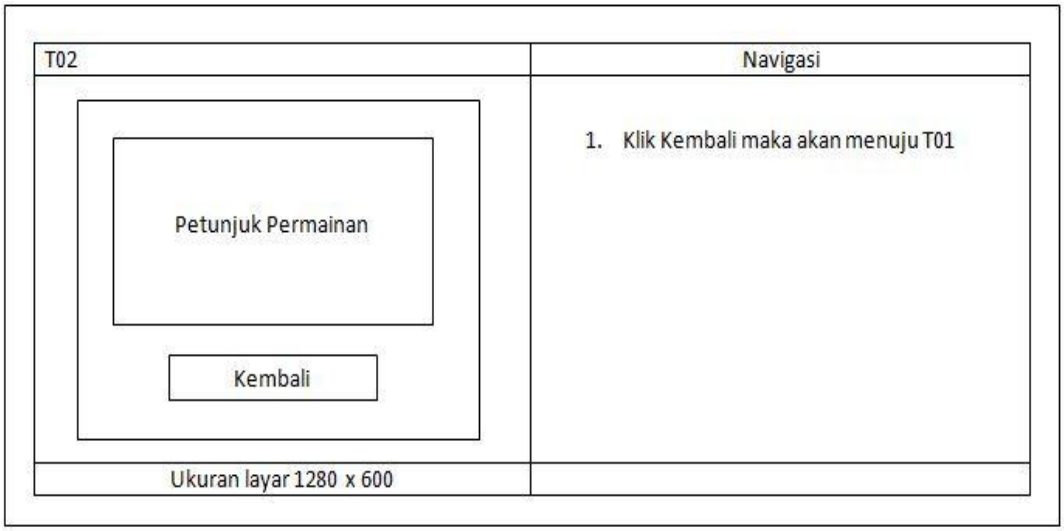
Perancangan antarmuka Menu Utama dapat dilihat pada gambar 3.44.



Gambar 3.44 Perancangan Antarmuka Menu Utama

2. **Perancangan Antarmuka Petunjuk**

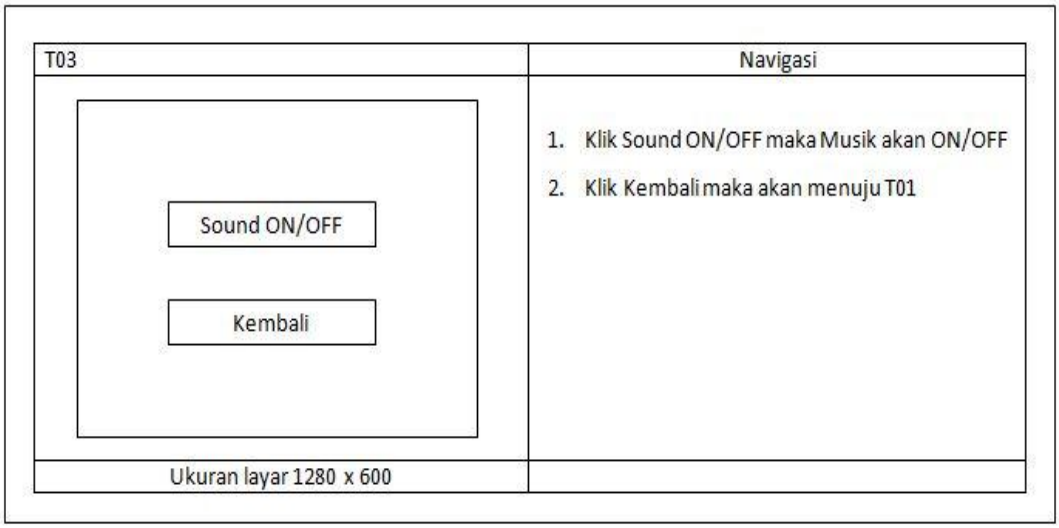
Perancangan antarmuka Petunjuk dapat dilihat pada gambar 3.45.



Gambar 3.45 Perancangan Antarmuka Petunjuk

3. Perancangan Antarmuka Pengaturan Suara

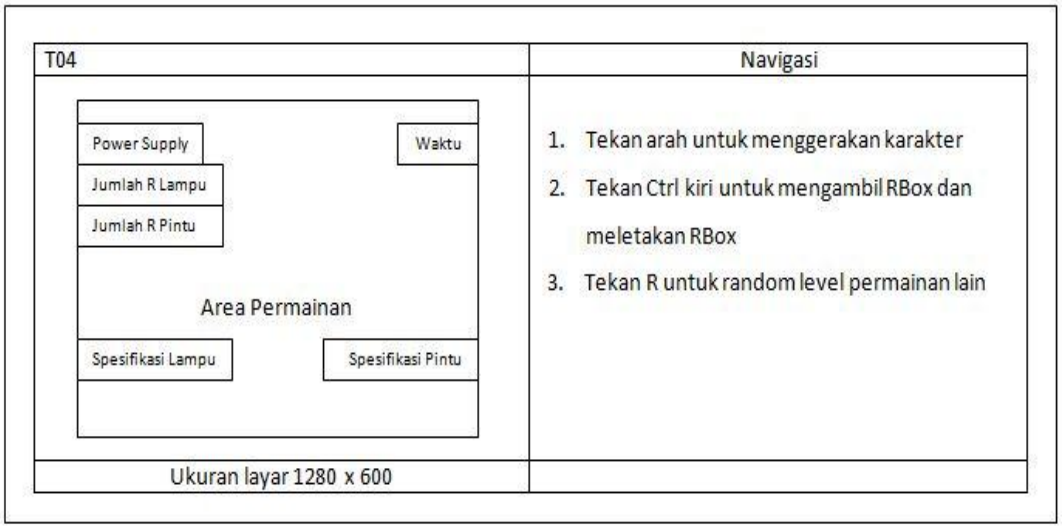
Perancangan antarmuka Pengaturan Suara dapat dilihat pada gambar 3.46.



Gambar 3.46 Perancangan Antarmuka Pengaturan Suara

4. Perancangan Antarmuka Mulai Permainan

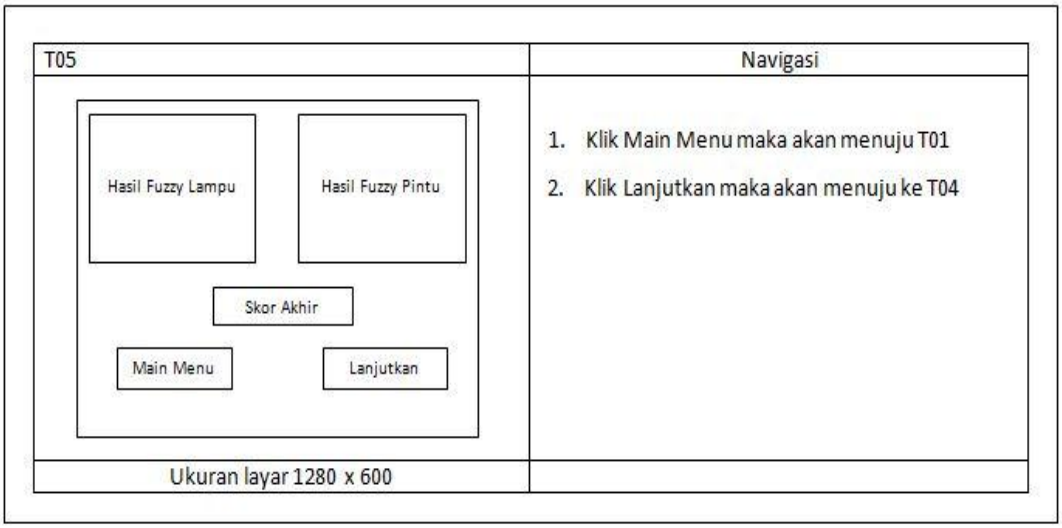
Perancangan antarmuka Mulai Permainan dapat dilihat pada gambar 3.47.



Gambar 3.47 Perancangan Antarmuka Mulai Permainan

5. Perancangan Antarmuka Selesai Permainan

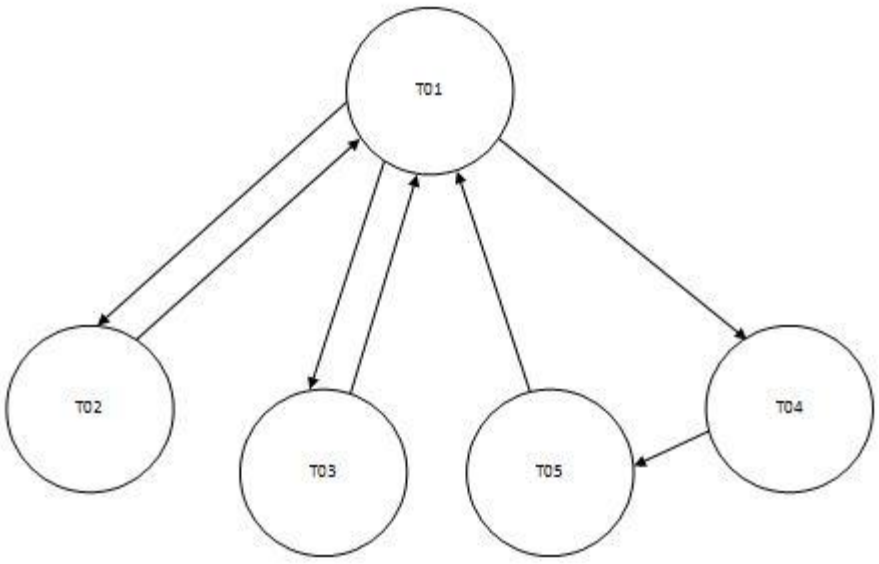
Perancangan antarmuka Selesai Permainan dapat dilihat pada gambar 3.48.



Gambar 3.48 Perancangan Antarmuka Selesai Permainan

3.9.5 Jaringan Semantik

Jaringan semantik dari *game* yang akan dibangun dapat dilihat pada gambar 3.49 .



Gambar 3.49 Jaringan Semantik