BAB 3

ANALISIS DAN PERANCANGAN

3.1 Analisis

Pada proses analisis ini mencakup berbagai analisis diantaranya analisis *game*, analisis masukan, analisis algoritma, analisis kebutuhan perangkat lunak yang mencakup analisis kebutuhan fungsional dan nonfungsional serta perancangan-perancangan dalam pembuatan perangkat lunak.

3.1.1 Analisis Masalah

Misi utama dari game ini adalah bertahan hidup. Terkadang *player* tidak dapat bertahan hidup karena *player* masih belum bisa menentukan musuh yang harus diprioritaskan untuk diserang lebih dulu. Algoritma greedy dan logika fuzzy merupakan solusi yang bisa diterapkan dalam game ini. Algoritma greedy sebagai pencari rute terpendek sehingga player akan lebih memprioritaskan musuh terdekat dan logika fuzzy untuk mengatur perilaku menyerang yang mengacu pada nyawa karakter dan musuh serta dan damage yang dimiliki oleh karakter.

3.1.2 Analisis *Game* Sejenis

Pada tahap ini akan dilakukan analisi terhadap *game* yang sejenis, analisis *game* sejenis dilakukan dengan metode observasi terhadap *game* sejenis

3.1.2.1 Pengenalan Game Dead State

Game dead state adalah game yang bertipe *Tactical Role-Playing Game*(T-RPG). Game ini dikembangkan oleh *DoubleBear Production*. Game ini dimulai ketika kota fiksi Splendid dikuasai oleh zombie dan karakter utama harus bertanggung jawab untuk kelangsungan hidup para warga yang belum terkena virus *zombie* tersebut.

Misi dari game *Dead State* adalah bagaimana karakter utama dalam game ini ini mampu mempertahankan kelangsungan hidup dari buruan *zombie* yang sudah menguasai kota. Karakter utama akan berjuang mengumpulkan bahan makanan dan persediaan lainya untuk dapat bertahan hidup dengan cara keluar dari penampungan dan mengumpulkan barang yang tersisah di dalam kota yang telah dipenuhi oleh *zombie.* Di dalam *game Dead State* pemain diminta untuk mengambil keputusan sulit dan menantang untuk menjaga diri mereka sendiri dan sesama korban yang masih hidup.

3.1.2.2 Gameplay Dead State

Game dari Dead state adalah sebagai berikut

1. Karakter utama terdiri dari 3 orang
2. Karakter memiliki *Action-point* (AP) total yang merupakan penggabungan dari AP aksi dan AP senjata
3. Setiap aksi yang dilakukan karakter dipengaruhi oleh AP total
4. Senjata yang digunakan memiliki nilai AP senjata yang berbeda dan ketika player menggunakan senjata maka AP total karakter akan di kurangi Oleh AP senjata.
5. Strategi dalam game ini menggunakan *Turn-Based Strategy* atau bergiliran jadi karakter dan *zombie* akan bergiliran dalam melakukan penyerangan.
6. Karakter harus mengumpulkan bahan makanan, senjata, obat-obatan yang masih terdapat di dalam kota yang sudah dipenuhi oleh *zombie.*

3.1.2.3 Tampilan Game Dead State

Menu utama dalam game Dead State dapat dilihat pada gambar 3.1



Gambar 3.1 Gambar Menu Utama *Game Dead State*

Tampilan karakter saat berhadapan melawan zombie dapat dilihat pada gambar3.2.



Gambar 3.2 Tampilan Karakter Dalam Pertempuran dengan *Zombie*

3.1.2.4 Hasil Analisis Game Dead State

Adapun hasil analisis dari game *Dead State* memiliki kelebihan dan kekurangan, sebagai berikut:

a. Kelebihan Dead State:
    1. Game ini beripe T-RPG.
    2. Grafik sudah menggunakan 3D.
    3. *Story line* non-linear dan memungkinkan pemain untuk mengeksplorasi *game Dead State,* dan menghadapi tantangan baru setiap kali bermain.

b. Kekurangan:
    1. *Player* tidak dapat memprioritaskan zombie yang harus diserang lebih dulu.

3.1.3 Pengenalan Game Me VS Zombie

Game yang dibuat akan mengambil judul "*Me Vs Zombie*". Jenis game yang dibuat adalah bertipe RPG. User akan mengendalikan tiga karakter dalam game untuk melawan musuh atau *zombie.*

3.1.3.1 Story Line

Game *Me vs Zombie* merupakan game yang beripe RPG menceritakan tetang 3 orang karakter yang mepertahankan hidup dari serangan *zombie*.Karakter dan musuh mempunyai nilai *healt-point*(HP) awal yang nilainya 100, HP yang menandakan status dari karakter atau musuh, jika HP kurang dari 60 maka status karakter atau musuh dikatakan lemah.

Untuk *story line* dalam game ini di bagi menjadi 3 level. Untuk kesulitan tiap level dibagi menurut banyaknya musuh yang dihadapi, yaitu:

1. Level 1

    Setelah masuk di level satu maka *player* memulai permainan dengan menekan tombol *play* dan akan mucul secara acak posisi karakter dan musuh. *Player* harus memilih karakter dan musuh yang akan melakukan

penyerangan. Setelah memilih karakter dan musuh maka karakter akan berjalan secara otomatis ke musuh. Musuh akan menyerang ketika karakter telah melakukan *atack* ke musuh. Karakter yang telah dipilih untuk menyerang tidak dapat dipilih lagi kecuali ketiga karakter yang di pakai sudah dipilih maka karakter dapat digunakan kembali. Jika dalam level 1 semua pemain mati maka permainan akan selesai tetapi jika semua musuh mati makan akan masuk ke level 2.

2. Level 2

   Dilevel 2 untuk alur penyerangan masih sama dengan level 1 tetapi pada level 2 kodisi karakter masih tetap sama dengan level 1 dan akan melawan 8 musuh. *Player* akan masuk ke level 3 jika telah membunuh semua musuh.

3. Dilevel 3 alur penyerangan masih sama seperti dilevel 1 dan level 2 tetapi player akan menghadapi 10 musuh dan kondisi HP *player* masih tetap sama seperti level sebelumnya.

3.1.3.2 *Gameplay*

Misi utama dalam game ini adalah karakter harus mampu bertahan hidup dari serangan *zombie.* Untuk menunjang permainan dalam mempertahan hidup game ini terdapat sugesti saran penyerangan terhadap *zombi*e yang dapat dipakai sebagai prioritas penyerangan dari pemain. Di dalam *gameplay* karakter mempunyai 2 kondisi yaitu:

1. Pemain mengikuti saran.
   a. Pemain melakukan penyerangan ketika sampai pada zombie yang dituju. Untuk setiap kali penyerangan akan mengurangi *Healt-Point* dari *zombie*.
   b. Setelah menyerang pemain dapat kabur.
   c. Pemain juga dapat kabur secara bebas jika tidak memungkinkan untuk melakukan penyerangan

2. Menang
    a. Pemain dikatakan menang jika dapat membunuh semua *zombie*.
    b. Pemain dikatakan kalah jika semua karakter mati.

3.1.4 Analisis Algoritma Greedy

Algoritma greedy dalam *game* ini betujuan untuk mencari rute terdekat dari karakter *player* ke musuh.

3.1.4.1 Analisis Masukan

Analisis masukan yang dilakukan dalam penelitian ini adalah menganalisis parameter algoritma greedy pada *Game* yang akan dibuat. Parameter yang inputkan yang dimiliki oleh pemain dan musuh adalah berupa inputan koordinat yang merupakan letak dari pemain dan musuh. Titik koordinat karakter dan musuh akan muncul secara otomatis ketika permainan dimulai. Dari titik – titik koordinat tersebut player akan memilih karakter secara acak, setelah memilih karakter maka langkah selanjutnya adalah menghitung bobot terendah dari karakter dan musuh. Bobot terendah didapatkan dari hasil pengurangan koordinat karakter dan koordinat musuh. Hasil bobot terendah akan menghasilkan jumlah kotak yang harus dilewati oleh karakter dan dari kotak itu akan menujukan titik koordinat tujuan karakter dan merupakan rute terpendek dari karakter yang dipilih ke titik koordinat tujuan. Berikut adalah tahapan algoritma *greedy*:

1. Menentukan himpunan kandidat (C)

    Himpunan kandidat ini berisi elemen-elemen yang memiliki peluang untuk pembentuk solusi. Hipunan kandidat dalam game ini adalah titik koordinat karakter dan musuh.

    C: {…}
2. Menentukan Himpunan Solusi (S)

Himpunan ini berisi solusi permasalahan yang diselesaikan. Himpunan Solusi terdiri dari semua titik koordinat tujuan karakter.

S: {…}

3. Menentukan Fungsi Seleksi

   Fungsi yang pada setiap langkah memilih kandidat paling memungkinkan mencapai solusi optimal. Fungsi seleksi disini akan menghitung bobot terkecil dari semua tujuan karakter. Bobot terkecil akan menghasilkan jumlah langkah yang diambil karakter ke titik koordinat tujuan.

4. Menentukan Fungsi Kelayakan

   Fungsi yang memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak, yakni kandidat tersebut bersama-sama dengan himpunan solusi yang sudah terbentuk tidak melanggar kendala yang ada. Fungsi kelayakan yang diambil adalah titik koordinat tujuan.

5. Menentukan Fungsi Obyektif

   Nilai minimum yang didapat, yaitu jumlah langkah dan titik koordinat tujuan.

3.1.3.2 Analisis Greedy

Algoritma greedy dalam *game* ini hanya digunakan dalam pengambilan jarak terdekat yang dimana jarak tersebut akan menjadi paramter jarak pada logika fuzzy.

Algoritma greedy adalah algoritma yang memecahkan masalah langkah demi langkah dan merupakan salah satu metode dalam masalah optimasi. Algoritma greedy membentuk solusi langkah perlangkah sebagai berikut:

1. Terdapat banyak pilihan yang perlu dipilih pada setiap langkah solusi, oleh karena itu, pada setiap langkah harus dibuat keputusan yang terbaik dalam menentukan pilihan.
2. Pendekatan yang digunakan didalam algoritma *greedy* adalah membuat pilihan yang terlihat memberikan perolehan terbaik yaitu

dengan membuat pilihan *optimum local* pada setiap langkah dan diharapkan akan mendapatkan solusi optimum global

Masalah optimasi dalam konteks algoritma *greedy* disusun oleh elemen-elemen sebagai berikut:

1. Himpunan kandidat (C)

   Himpunan kandidat disini adalah koordinat dari karakter pemain dan musuh. Seperti contoh yang terlihat pada gambar 3.3. *Player* yang aktif adalah player yang kotaknya berwarna merah yaitu P1. Koordinat P1 (2,3). dan Z1 koordinat musuh (14,1), Z2(10,2), Z3(11,3), Z4(7,0), Z5(9,5)

   C:{2,3; 14,1; 10,2; 11,3; 7,0; 9,5}

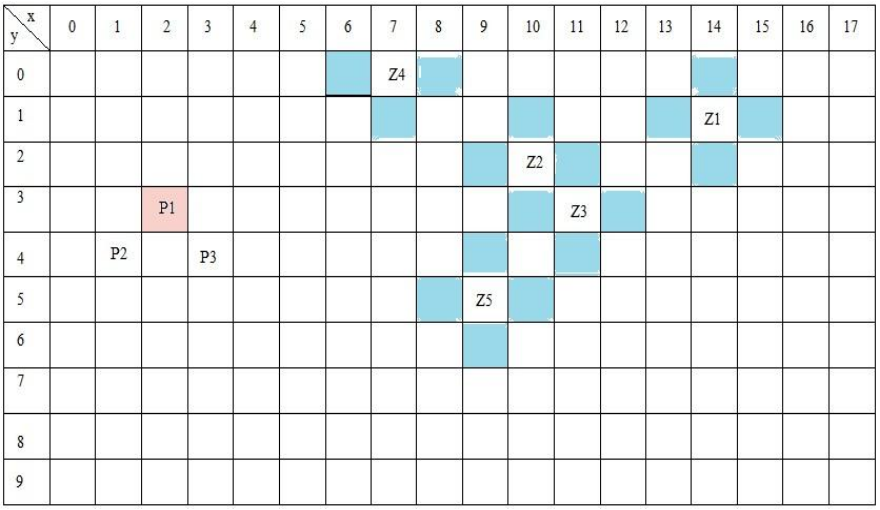
x / y	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0								Z4										
1															Z1			
2											Z2							
3			P1									Z3						
4		P2		P3														
5										Z5								
6																		
7																		
8																		
9																		

Gambar 3.3 Koordinat Karakter *Player* dan koordinat *Zombie*

2. Himpunan solusi (S)

   Himpunan solusi dari permasalaham pada gambar 3.4 adalah koordinat Z1(14,1), (14,0), (15,1),(14,2), Z2(10,2), (10,1), (11,2), (10,3), Z3 (10,3), (11,2), (13,3), (11,4), Z4 (6,0), (8,0), (7,1), Z5 (8,5), (9,4), (10,5), (9,6). Dapat dilihat pada gambar 3.4 yang kotaknya berwana biru:

   S: {14,1; 14,0; 15,1; 14,2; 10,2; 10,1; 11,2; 10,3; 10,3; 11,2; 13,3; 11,4; 6,0; 8,0; 7,1; 8,5; 9,4; 10,5; 9,6}



Gambar 3.4 Titik koordinar tujuan P1

3. Fungsi seleksi

Untuk menentukan fungsi komulatif dapat dihitung dengan koordinat dari pemain dikurang tujuan pemain.

a. Tujuan P1 ke Z1

a) | 2 – 13 | + | 3 – 1 | = 13

b) | 2 – 14 | + | 3 – 0 | = 15

c) | 2 – 15 | + | 3 – 1 | = 15

d) | 2 – 14 | + | 3 – 2 | = 13

b. Tujuan P1 ke Z2

a) | 2 – 9 | + | 3 – 2 | = 8

b) | 2 – 10 | + | 3 – 1 | = 10

c) | 2 – 11 | + | 3 – 1 | = 10

d) | 2 – 10 | + | 3 – 3 | = 8

c. Tujuan P1 ke Z3

a) | 2 – 10 | + | 3 – 3 | = 8

b) | 2 – 11 | + | 3 – 2 | = 10

c) | 2 – 12 | + | 3 – 3 | = 10

d) | 2 – 11 | + | 3 – 4 | = 10

d. Tujuan P1 ke Z4

a) | 2 – 6 | + | 3 – 0 | = 7

b) | 2 – 8 | + | 3 – 0 | = 9

c) | 2 – 7 | + | 3 – 1 | = 7

e. Tujuan P1 ke Z5

a) | 2 – 8 | + | 3 – 5 | = 8

b) | 2 – 9 | + | 3 – 4 | = 8

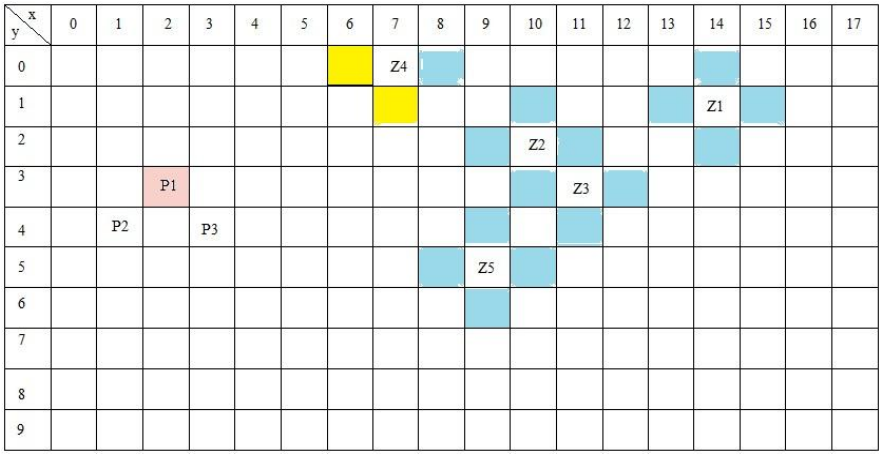
c) | 2 – 10 | + | 3 – 5 | = 10

d) | 2 – 9 | + | 3 – 6 | = 10

Dari hasil perhitungan didapat bobot terendah adalah 7 dengan target tujuan untuk diserang adalah Z4, maka titik koordinat tujuan yang didapatkan adalah (6,0) dan (7,1).

4. Fungsi kelayakan

Fungsi kelayakan yang didapatkan dari perhitungan bobot terendah yang menghasilkan koordinat tujuan yaitu koordinat (6,0) dan (7,1) atau dapat dilihat pada gambar 3.5 yang berwana kuning



Gambar 3.5 Titik koordinat Terdekat

5. Fungsi obyektif

   Nilai minimum atau rute terpendek yang didapat pada permasalahan di atas ada 2 yaitu koordinat (6,0) dan (7,1) dengan jumlah bobot terendahnya adalah 7 langkah.

3.1.5 Analisi Logika Fuzzy

Pada penerapan metode *Fuzzy Tsukamto* dalam *game* ini untuk menentukan prioritas serangan pemain terhadapat musuh. Komponen yang digunakan dalam pembuatan game ini yaitu *Model Base*.

3.1.5.1 Analisis Masukan

Analisis masukan yang dilakukan dalam penenlitian ini adalah menganalisis perameter dalam logika fuzzy pada game *Me Vs Zombie*. Parameter yang dimiliki oleh karakter yaitu *Damage* dan *Healt-Point* (HP) parameter yang dimiliki oleh musuh adalah *Healt-Point* (HP), yang nantinya parameter – parameter tersebut berpengaruh pada prioritas serang yang dilakukan player. Berikut rentang nilai dari parameter karakter dan musuh pada tabel 3.1

Tabel 3.1 Parameter Masukan Dan Rentang Nilai Masukannya

No	Parameter	Rentang Nilai
1	*Damage* Karakter	1 – 20
2	*Healt-Point* (HP) Karakter	10 – 100
3	*Healt-Point* (HP) Musuh	10 – 100

3.1.5.2 Model Base Metode Tsukamoto

Di dalam model base ini, secara umum terdapat 3 langkah untuk menentukan prioritas serangan dengan metode *Tsukamoto*, yaitu mendefinisikan variabel, infrensi, dan *defuzzifikasi* (menentukan output *crisp*).

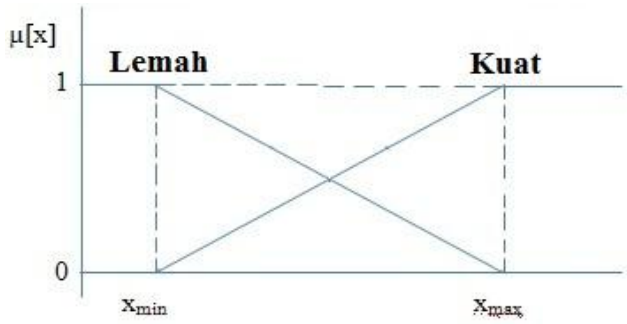
1. Mendefinisikan Variabel *Fuzzy*

Pada tahap ini, nilai keanggotaan himpunan parameter Damage, nyawa, musuh saat ini dicari menggunakan funggsi keanggotaan himpunan fuzzy dengan memperhatikan nilai maksimum dan nilai minimum data 1 periode terakhir dari tiap variabe. Variabel 1 periode terakhir antara antara lain variabel Damage, variabel nyawa dan variabel musuh dan variabel serang. Nilai masukan dari tiap-tiap variabel diambil dari nilai masukan parameter, yaitu:

1. Variabel *Damage* nilai masukannya diambil dari parameter *damage*. Variabel *damage* mempunyai 2 himpunan fuzzy yaitu fuzzy kuat dan fuzzy lemah. Damage lemah jika damage kurang dari 10.
2. Variabel Nyawa nilai masukannya diambil dari parameter HP karakter. Variabel nyawa mempunyai 2 himpunan fuzzy yaitu fuzzy kuat dan fuzzy lemah. Nyawa dikatakan lemah jika HP karakter kurang dari 60.
3. Variabel Musuh nilai masukannya diambil dari parameter HP musuh. Variabel musuh mempunyai 2 himpunan fuzzy yaitu fuzzy kuat dan fuzzy lemah. Musuh dikatakan lemah jika HP musuh kurang dari 60
4. Variabel Serang merupakan hasil yang diambil dari karakter. Serang memiliki 2 himpunan fuzzy yaitu fuzzy menyerang dan fuzzy kabur. Karakter akan kabur jika nilai serangnya krang dari 60.

a. Variabel Damage

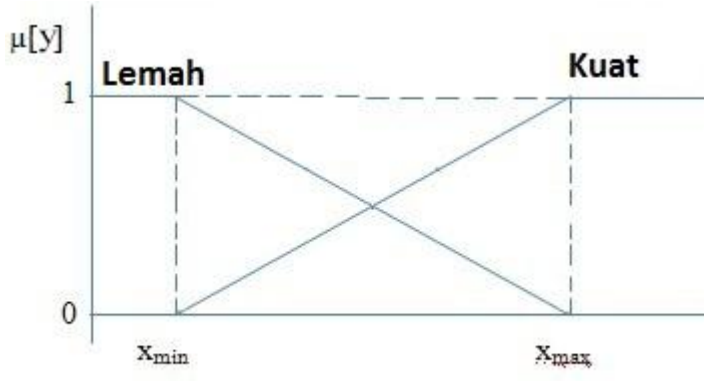
Variabel damage terdiridari 2 himpunan *fuzzy* yaitu LEMAH dan KUAT. Fungsi kenaggotaan himpunan *fuzzy* LEMAH dan KUAT dari variabel Damage direpresentasikan pada gambar 3.6



Gambar 3.6 Fungsi Keangggotaan himpunan Fuzzy LEMAH dan KUAT dari variabel Damage

b. Variabel Nyawa

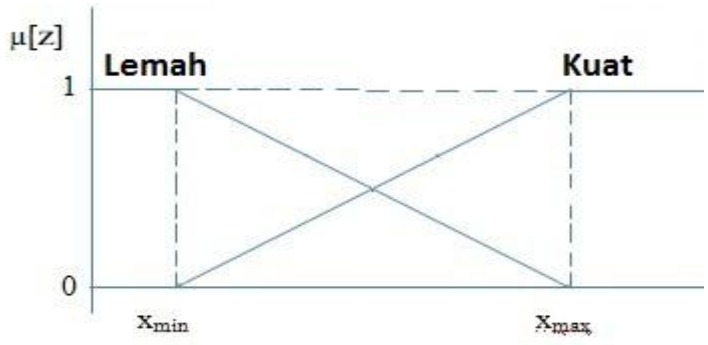
Variabel nyawa terdiri dari 2 himpunan fuzzy yaitu KUAT dan LEMAH. Fungsi kenaggotaan himpunan fuzzy KUAT dan LEMAH dari variabel Damage direpresentasikan pada gambar 3.7



Gambar 3.7 Fungsi keanggotaan fuzzy LEMAH dan KUAT dari variabel Nyawa

c. Variabel Musuh

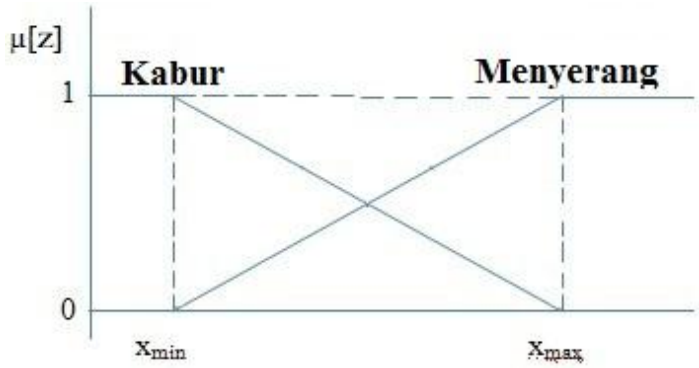
Variabel nyawa terdiri dari 2 himpunan *fuzzy* yaitu KUAT dan LEMAH. Fungsi kenaggotaan himpunan *fuzzy* LEMAH dan KUAT dari variabel Musuh direpresentasikan pada gambar 3.8



Gambar 3.8 Fungsi keanggotaan himpunan fuzzy LEMAH dan KUAT dari variabel Musuh

d. Variabel Serang

Variabel serang terdiri dari 2 himpunan *fuzzy* yaitu KABUR dan MENYERANG. Fungsi kenaggotaan himpunan *fuzzy* KABUR dan MENYERANG dari variabel Serang direpresentasikan pada gambar 3.9



Gamba 3.9 Fungsi keanggotaan himpunan fuzzy LEMAH dan KUAT dari variabel Musuh

2. INFERENSI

Inferesi adalah proses penggabungan banyak aturan berdasarkan data yang tersedia. Dari uraian di atas, telah terbentuk 8 himpunan fuzzy sebagai berikut

damage KUAT, damage LEMAH, nyawa LEMAH, nyawa KUAT, musuh LEMAH, musuh KUAT , serang KABUR, serang MENYERANG.

Dengan mengkombinasikan himpunan-himpunan *fuzzy* tersebut, maka diperoleh delapan aturan *fuzzy* sebagai berikut:

[R1] IF Damage KUAT and Nyawa KUAT and Musuh KUAT then MENYERANG

[R2] IF Damage LEMAH and Nyawa KUAT and Musuh KUAT then KABUR

[R3] IF Damage LEMAH and Nyawa LEMAH and Musuh KUAT then KABUR

[R4] IF Damage KUAT and Nyawa LEMAH and Musuh KUAT then MENYERANG

[R5] IF Damage LEMAH and Nyawa KUAT and Musuh LEMAH then KABUR

[R6] IF Damage KUAT and Nyawa LEMAH and Musuh LEMAH then MENYERANG

[R7] IF Damage KUAT and Nyawa KUAT and Musuh LEMAH then MENYERANG

[R8] IF Damage LEMAH and Nyawa LEMAH and Musuh LEMAH then KABUR

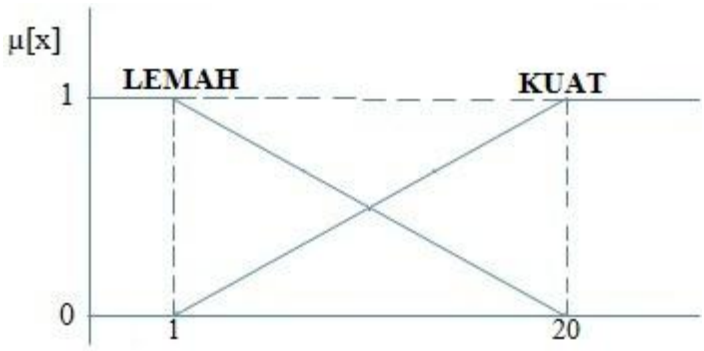
Contoh Kasus Pada Game

Contoh kasus pada game dapat dilihat dari contoh kasus pada gambar 3.3 di atas. P1 akan menyerang ke musuh terdekat yaitu Z4. Misalkan P1 mempunyai Damege (20), Nyawa (80), dan Z4 mempunyai Nyawa (30), maka prioritas yang akan diambil oleh pemain adalah?

a. Memodelkan variabel *fuzzy (Fuzzyfikasi)*

Ada 4 variabel fuzzy yang dimodelkan, yaitu Damage, nyawa, musuh, dan serang

1) Damage, terdiri dari 2 himpunan *fuzzy* yaitu KUAT dan LEMAH. Fungsi keanggotaan Damage direpresentasikan pada Gambar 3.10



Gambar 3.10 Fungsi keanggotaan himpunan *fuzzy* LEMAH dan KUAT dari variabel Damage

Fungsi keanggotaan Himpunan LEMAH dan KUAT dari variabel Damage:

$$\mu_{\text{DamageLemah}}[x] = \begin{cases} 1 & , \quad x \leq 1 \\ \dfrac{20-x}{19} & , 1 \leq x \leq 20 \\ 0 & , \quad x \geq 20 \end{cases}$$

$$\mu_{DamageKuat}[x] = \begin{cases} 0 & , \quad x \leq 1 \\ \dfrac{x-20}{19} & , 1 \leq x \leq 20 \\ 1 & , \quad x \geq 20 \end{cases}$$

Nilai keanggotaan himpunan LEMAH dan KUAT dari variabel DAMAGE bisa di cari dengan

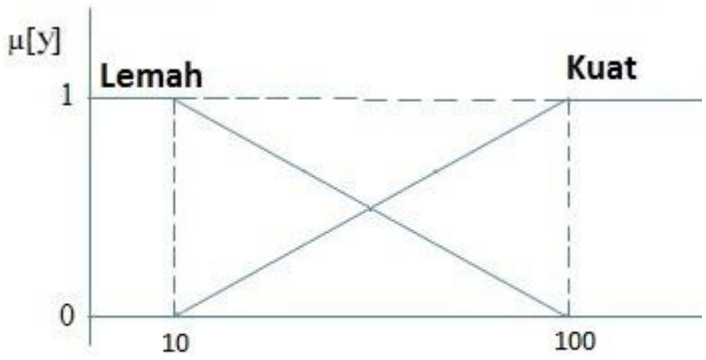
$$\mu_{DamageLemah}[20] = \frac{20-20}{19}$$

$$= 0$$

$$\mu_{DamageKuat}[20] = \frac{20-10}{19}$$

$$= 0{,}52$$

2) Nyawa terdiri dari 2 himpunan fuzzy, yaitu LEMAH dan KUAT. Fungsi keanggotaan himpunan LEMAH dan KUAT dari variabel Nyawa direpresentasikan pada gambar 3.11



Gambar 3.11 Fungsi Keanggotaan himpunan fuzzy LEMAH, KUAT dari variabel Nyawa

Fungsi keanggotaan himpunan fuzzy LEMAH dan KUAT dari variabel Nyawa

$$\mu_{\text{NyawaLemah}}[y] = \begin{cases} 1 & x \leq 10 \\ \dfrac{100 - x}{90}, & 10 \leq x \leq 100 \\ 0 & x \geq 100 \end{cases}$$

$$\mu_{NyawaKuat}[y] = \begin{cases} 0 & x \leq 10 \\ \dfrac{x - 10}{90}, & 10 \leq x \leq 100 \\ 1 & x \geq 100 \end{cases}$$

Nilai keanggotaanhimpunan LEMAH dan KUAT dari variabel Nyawa dapat dicari dengan

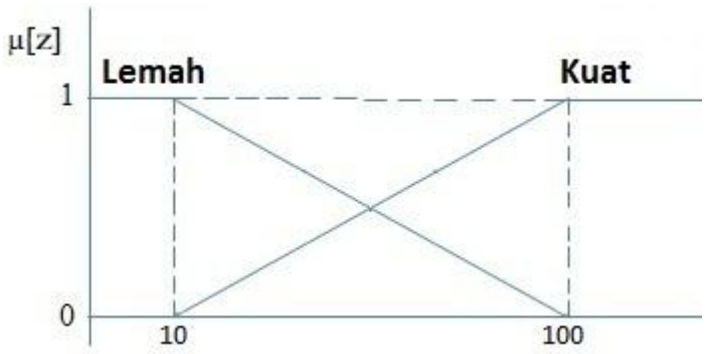
$$\mu_{NyawaLemah}[80] = \frac{100 - 80}{90}$$

$$= 0{,}22$$

$$\mu_{NyawaKuat}[80] = \frac{80 - 10}{90}$$

$$= 0{,}78$$

3) Musuh : terdiri dari 3 himpunan fuzzy, yaitu LEMAH dan KUAT. Fungsi keanggotaan fuzzy LEMAH dan KUAT dari variabel Musuh direpresentasikan pada gambar 3.12



Gambar 3.12 Fungsi keanggotaan Himpunan fuzzy LEMAH dan KUAT dari variabel Musuh

Fungsi keanggotaan himpunan fuzzy LEMAH dan KUAT dari variabel Musuh

$$\mu_{MusuhLemah}[z] = \begin{cases} 1, & x \leq 10 \\ \frac{100-z}{90}, & 10 \leq x \leq 100 \\ 0, & x \geq 100 \end{cases}$$

$$\mu_{MusuhKuat}[z] = \begin{cases} 0, & x \leq 10 \\ \frac{z-10}{90}, & 10 \leq x \leq 100 \\ 1, & x \geq 100 \end{cases}$$

Nilai keanggotaan dari himpunan LEMAH dan KUAT dari variabel Musuh dapat dicari:

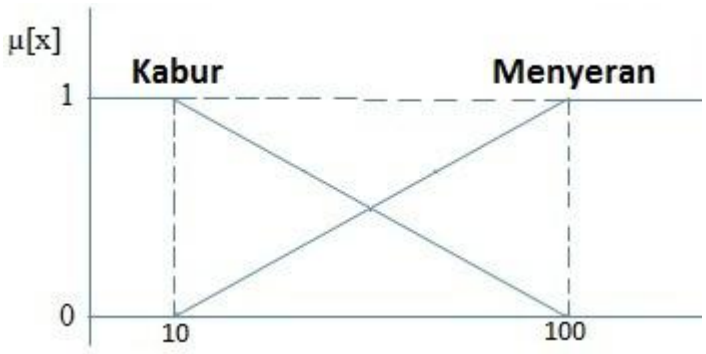
$$\mu_{MusuhLemah}[30] = \frac{100-30}{90}$$

$$= 0{,}78$$

$$\mu_{MusuhLemah}[30] = \frac{30-10}{90}$$

$$= 0{,}22$$

4) Serang: terdiri dari 2 himpunan fuzzy, yaitu MENYERANG dan KABUR. Fungsi keanggotaan himpunan fuzzy MENYERANG dan KABUR dari variabel Serang direpresentasikan pada Gambar 3.13



Gambar 3.13 Fungsi keanggotaan Himpunan Fuzzy MENYERANG dan KABUR dari variabel Serang:

$$\mu_{Menyerang}[x] = \begin{cases} 1 & , \quad x \leq 10 \\ \frac{100 - x}{90} & , 10 \leq x \leq 100 \\ 0 & , \quad x \geq 100 \end{cases}$$

$$\mu_{Kabur}[x] = \begin{cases} 0 & , \quad x \leq 10 \\ \frac{x - 10}{90} & , 10 \leq x \leq 100 \\ 1 & , \quad x \geq 100 \end{cases}$$

b. INFERENSI

[R1] IF Damage KUAT and Nyawa KUAT and Musuh KUAT then MENYERANG;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R1] yang dinotasikan denagn α1 diperoleh rumus berikut:

$\alpha 1 = \mu_{DamageKuat \cap} \mu_{NyawaKuat \cap} \mu_{MusuhKuat}$

$= \min\ (\mu_{\text{DamageKUAT}}\ [0{,}52], \mu_{\text{NyawaKuat}}[0{,}78], \mu_{\text{MusuhKuat}}[0{,}22])$

$= \min\ 0{,}22$

$z1 = z_{max} - \alpha 1(z_{max} - z_{min})$

$z1 = 100 - 0{,}22(100 - 10)$

$z1 = 100 - 19{,}8$

$z1 = 80{,}2$

[R2] IF Damage LEMAH and Nyawa KUAT and Musuh KUAT then KABUR;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R2] yang dinotasikan denagn α2 diperoleh rumus berikut:

$$\alpha 2 = \mu_{DamageLemah} \cap \mu_{NyawaKuat} \cap \mu_{MusuhKuat}$$

$$= \min\left(\mu_{\text{DamageLEMAH}}[0], \mu_{\text{NyawaKuat}}[0{,}78], \mu_{\text{MusuhKuat}}[0{,}22]\right)$$

$$= \min 0$$

$$z2 = z_{max} - \alpha 2(z_{max} - z_{min})$$

$$z2 = 100 - 0(100 - 10)$$

$$z2 = 100 - 0$$

$$z2 = 100$$

[R3] IF Damage LEMAH and Nyawa LEMAH and Musuh KUAT then KABUR;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R3] yang dinotasikan denagn α1 diperoleh rumus berikut:

$$\alpha 3 = \mu_{DamageLemah} \cap \mu_{NyawaLemaht} \cap \mu_{MusuhKuat}$$

$$= \min\left(\mu_{\text{DamageLEMAH}}[0], \mu_{\text{NyawaLemah}}[0{,}22], \mu_{\text{MusuhKuat}}[0{,}22]\right)$$

$$= \min 0$$

$$z3 = z_{max} - \alpha 3(z_{max} - z_{min})$$

$$z3 = 100 - 0(100 - 10)$$

$$z3 = 100 - 0$$

$$z3 = 100$$

[R4] IF Damage KUAT and Nyawa LEMAH and Musuh KUAT then MENYERANG;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R1] yang dinotasikan denagn α1 diperoleh rumus berikut:

$\alpha 4 = \mu_{DamageKuat} \cap \mu_{NyawaLemah} \cap \mu_{MusuhKuat}$

$= \min\ (\mu_{\text{DamageKUAT}}\ [0{,}52], \mu_{\text{NyawaLemah}}\ [0{,}22], \mu_{\text{MusuhKuat}}\ [0{,}22])$

= min 0,22

$z4 = z_{max} - \alpha 4(z_{max} - z_{min})$

$z4 = 100 - 0{,}22(100 - 10)$

$z4 = 100 - 19{,}8$

$z4 = 80{,}2$

[R5] IF Damage LEMAH and Nyawa KUAT and Musuh LEMAH then KABUR;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R5] yang dinotasikan denagn α1 diperoleh rumus berikut:

$\alpha 5 = \mu_{DamagLemah} \cap \mu_{NyawaKuat} \cap \mu_{MusuhLemah}$

$= \min\ (\mu_{\text{DamageLemah}}\ [0], \mu_{\text{NyawaKuat}}\ [0{,}78], \mu_{\text{MusuhLemah}}\ [0{,}78])$

= min 0

$z5 = z_{max} - \alpha 5(z_{max} - z_{min})$

$z5 = 100 - 0(100 - 10)$

$z5 = 100 - 0$

$z5 = 100$

[R6] IF Damage KUAT and Nyawa LEMAH and Musuh LEMAH then MENYERANG;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R6] yang dinotasikan denagn α6 diperoleh rumus berikut:

$\alpha 6 = \mu_{DamageKuat} \cap \mu_{NyawaLemah} \cap \mu_{MusuhLemah}$

$= \min\ (\mu_{\text{DamageKuat}}\ [0{,}52], \mu_{\text{NyawaLemah}}\ [0{,}22], \mu_{\text{MusuhLemah}}\ [0{,}78])$

= min 0,22

$z6 = z_{max} - \alpha 6(z_{max} - z_{min})$

$z6 = 100 - 0{,}22(100 - 10)$

$z6 = 100 - 19{,}8$

$z6 = 80{,}2$

[R7] IF Damage KUAT and Nyawa KUAT and Musuh LEMAH then MENYERANG;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R7] yang dinotasikan denagn α7 diperoleh rumus berikut:

$\alpha 7 = \mu_{DamageKuat} \cap \mu_{NyawaKuat} \cap \mu_{MusuhLemah}$

$= \min\ (\mu_{\text{DamageKuat}}\ [0{,}52], \mu_{\text{NyawaKuat}}[0{,}78], \mu_{\text{MusuhLemah}}[0{,}78])$

= min 0,52

$z7 = z_{max} - \alpha 7(z_{max} - z_{min})$

$z7 = 100 - 0{,}52(100 - 10)$

$z7 = 100 - 46{,}8$

$z7 = 53{,}2$

[R8] IF Damage LEMAH and Nyawa LEMAH and Musuh LEMAH then KABUR;

Nilai kenaggotaan anteseden untuk aturan fuzzy [R8] yang dinotasikan denagn α8 diperoleh rumus berikut:

$\alpha 8 = \mu_{DamageLemah} \mu_{NyawaLemah} \cap \mu_{MusuhLemah}$

$= \min\ (\mu_{\text{DamageLEMAH}}\ [0], \mu_{\text{NyawaLemah}}[0{,}22], \mu_{\text{MusuhLemah}}[0{,}78])$

= min 0

$z1 = z_{max} - \alpha 1(z_{max} - z_{min})$

$z1 = 100 - 0(100 - 10)$

$z1 = 100 - 0$

$z1 = 100$

c. Menentukan Output Crisp (Deffuzzyfikasi)

Pada metode Tsukamoto, untuk menentkan output crisp digunakan defuzifikasi rat-rata terpusat yaitu:

$$Z = \frac{(\alpha 1 \times z1)+(\alpha 2 \times z2)+(\alpha 3 \times z3)+(\alpha 4 \times z4)+(\alpha 5 \times z5)+(\alpha 6 \times z6)+(\alpha 7 \times z7)+(\alpha 8 \times z8)}{\alpha 1+\alpha 2+\alpha 3+\alpha 4+\alpha 5+\alpha 6+\alpha 7+\alpha 8}$$

$$= \frac{(0{,}22 \times 80{,}2) + (0 \times 100) + (0 \times 100) + (0{,}22 \times 80{,}2) + (0 \times 100) + (0{,}22 \times 80{,}2) + (0{,}52 \times 53{,}2) + (0 \times 100)}{0{,}22 + 0 + 0 + 0{,}22 + 0 + 0{,}22 + 0{,}52 + 0}$$

$$= \frac{17{,}644 + 0 + 0 + 17{,}644 + 0 + 17{,}644 + 27{,}664 + 0}{1{,}18}$$

$$Z = \frac{80{,}596}{1{,}18}$$

$$Z = 68{,}34$$

Jadi nilai yang dihasilkan berdasarkan aturan yang sudah dibentuk sebelumnya. Sesuai dengan contoh dari perhitungan yang dihasilkan dari nilai Damage [20], nyawa [80], dan musuh [30] yang dimiliki pemain dengan nilai Z yang didapatkan yaitu 68,34 maka keputusan adalah pemain adalah menyerang karena nilai Z lebih besar dari 60.

3.2 Analisis Kebutuhan Perangkat Lunak

Pada tahap ini meliputi elemen-elemen yang diperlukan dalam mengimplementasikan algortima *greedy* dan logika fuzzy yaitu terdiri dari kebutuhan non-fungsional dan kebutuhan fungsional. Spesifikasi kebutuhan perangkat lunak akan dibagi kedalam dua bagian yaitu Kebutuhan Fungsional dan Spesifikasi Kebutuhan Non-Fungsional.

3.2.1 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional adalah langkah pembangunan perangkat lunak dengan menganalisis sumber daya yang akan dibutuh dan digunakan serta menggunakan perangkat lunak yang digunakan untuk membangun sistem. Perangkat keras dan perangkat lunak yang dimiliki harus sesuai dengan kebutuhan, sehingga dapat ditentukan kompabilitas aplikasi yang dibangun terhadap sumber daya yang ada.

3.2.1.1 Analisis Perangkat Keras

Perangkat keras atau *hardware* merupakan salah satu hal yang penting karena tanpa *hardware* yang memenuhi syarat, sistem yang akan dibuat tidak akan dapat berjalan. Perangkat keras minimum yang digunakan :

1. Processor Dual Core 1,80 Ghz
2. RAM 1 GB
3. VGA 512 MB
4. Hard disk 80 GB
5. Monitor
6. *Keyboard*

3.2.1.2 Analisis Pengguna

Berdasarkan hasil penelitian, spesifikasi minimal bagi pengguna yang dapat memainkan game ini adalah:

a. Peran : *Player*
b. Tingkat Keterampilan : Dapat mengoperasikan komputer
c. Pengalaman : Pernah memainkan *game* PC

3.2.1.3 Analisis Perangkat Lunak

Perangkat lunak atau *software* merupakan hal yang terpenting dalam mendukung kinerja sebuah sistem. Perangkat lunak digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar dapat saling berinteraksi diantara keduanya. Perangkat lunak yang dibutuhkan untuk membangun aplikasi *game* ini adalah sebagai berikut :

1. Sistem operasi Microsoft Windows 7.

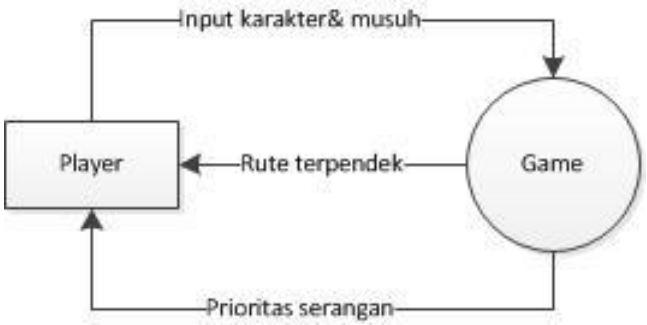
2. Visual Studio sebagai editor.

3.2.2 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional adalah segala bentuk data yang dibutuhkan oleh sistem agar sistem dapat berjalan sesuai dengan prosedur yang dibangun, dengan demikian perlu dilakukan analisis terhadap data yang dibutuhkan dari sistem yang sedang berjalan agar dapat diimplementasikan kedalam program aplikasi yang akan diajukan nantinya.

3.2.2.1 Diagram Konteks

Diagram konteks adalah suatu proses yang menggambarkan dan menjelaskan aliran informasi dari sistem yang diusulkan secara garis besarnya dan dapat dilihat pada gambar 3.14



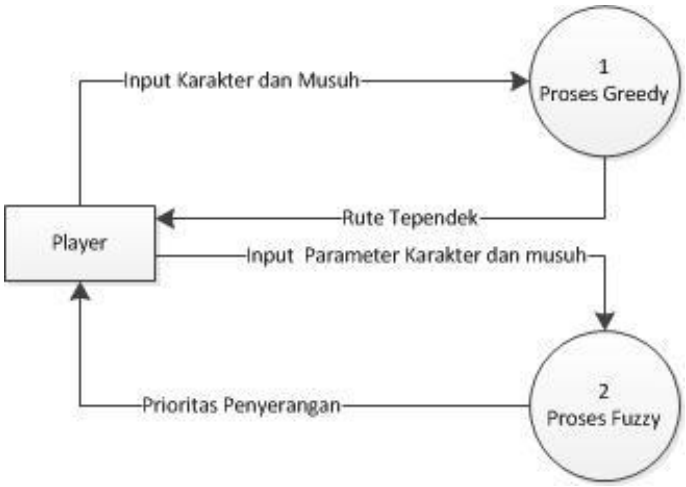
Gambar 3.14 Diagram Konteks Pada Game Yang Dibuat

3.2.2.2 Data *Flow Diagram*

Data Flow Diagram adalah representasi grafik dari sebuah sistem. DFD menggambarkan sebuah sistem yang telah ada atau baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut akan disimpan.

1. **DFD level 1 Game Me vs Zombie**

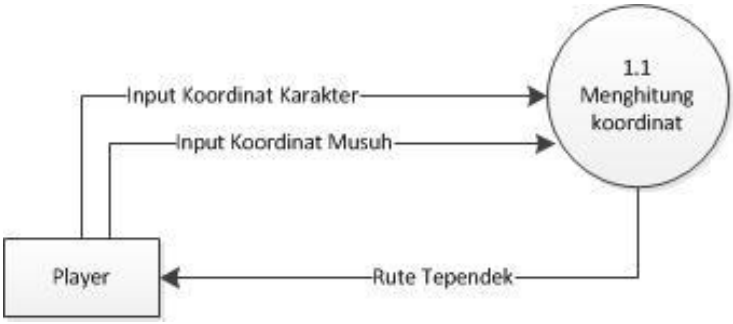
Berikut ini adalah gambar DFD level 1 game Me Vs Zombie 3.14



Gambar 3.15 DFD Level 1 Game Me vs Zombie

2. **DFD level 2 Game Me Vs Zombie level 2 Proses Greedy 1.1**

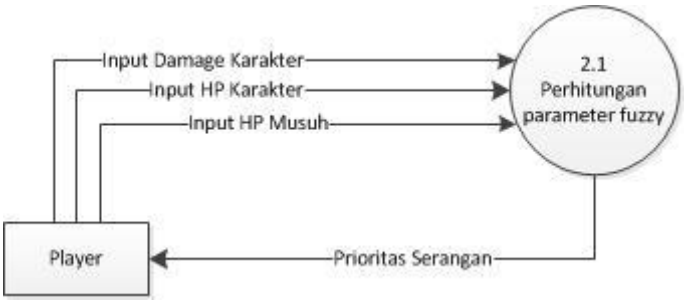
   Berikut ini adalah gambar DFD level 2 pada game Me vs Zombie yang dibuat terlihat pada gambar 3.16



Gambar 3.16 DFD Level 2 Game Me Vs Zombie Proses Greedy 1.1

3. **DFD level 2 Game Me Vs Zombie level 2 Proses Fuzzy 2.1**

   Berikut ini adalah gambar DFD level 2 proses fuzzy pada game Me vs Zombie yang dibuat terlihat pada gambar 3.17



Gambar 3.17 DFD Level 2 Game Me Vs Zombie Proses Fuzzy 2.1

3.2.3 Spesifikasi Proses

Spesifikasi proses merupakan deskripsi dari setiap elemen proses yang terdapat dalam program, yang meliputi nama, proses, input, output dan keterangan dari proses. Berikut ini spesifikasi proses dari Game pengaturan taktik dan prioritas serangan pada Me vs Zombie ditunjukan pada tabel 3.2

Tabel 3.2 Tabel Spesifikasi Proses Game Me vs Zombie

No	Proses	Keterangan
1	No.Proses	1
	Nama	Game Me Vs Zombie
	Source(sumber)	*Player*
	Input	1. Input Musuh 2. Input Karakter
	Output	Mendapatkan target musuh yang diserang
	Logika Proses	1. *Player* menginputkan Musuh 2. *Player* menginputkan karakter pemain 3. Proses Greedy dijalankan 4. Akan ketemu rute terpendek 5. Proses *fuzzy* dijalankan 6. Proses *fuzzy* akan menghasilkan

		target musuh yang akan dituju
2	**No.Proses**	**1.1**
	Nama	Proses Greedy
	Source(sumber)	*Player*
	Input	1. Koordinat Musuh 2. Koordinat Karakter
	Output	Rute terdekat antar karakter dan musuh
	Logika Proses	1. Pemain Input Koordinat dari musuh dan karakter 2. Akan dihitung bobot teredah 3. Akan ketemu rute terdekat antar pemain dan musuh
3	**No.Proses**	**2.1**
	Nama	Proses Fuzzy
	Source(sumber)	*Player*
	Input	1. Nyawa Karakter 2. Damage Karakter 3. Jarak Musuh
	Output	Saran prioritas penyerangan
	Logika Proses	1. Musuh terdekat dengan karakter 2. Menghitung nilai-nilai dari parameter musuh dan kakrakter 3. Pesan prioritas penyerangan

3.3 Perancangan Sistem

Perancangan adalah suatu bagian metodologi pengembangam dari suatu perangkat lunak yang dilakukan setelah tahapan untuk memberikan gambaran secara terperinci. Dalam tahap ini digambarkan rancangan perangkat lunak yang akan dibangun sebelum dilanjutkan ke proses pengkodean (*coding*) ke dalam bahasa pemrograman.

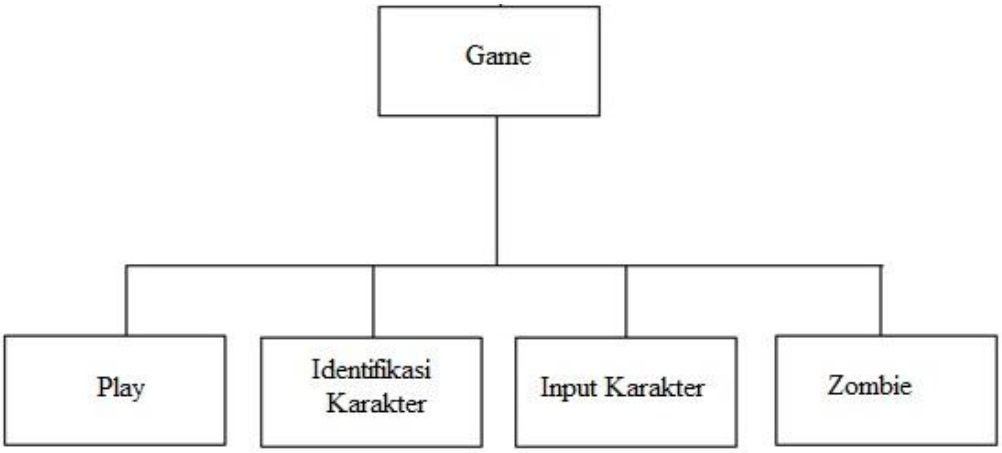
3.3.1 Perancangan Perangkat Lunak

Perancangan perangkat lunak adalah kegiatan untuk merancang aplikasi yang dibangun, tahapannya dimulai dari perancangan menu, perancangan antar muka, jaringan semantik dan perancangan *method*.

3.3.1.1 Struktur Menu

Struktur menu merupakan bentuk utama dari suatu rancangan sistem atau aplikasi yang berfungsi untuk memudahkan dalam pemilihan proses yang akan dikerjakan sesuai dengan kebutuhan dan hak aksesnya.

Berikut ini merupakan gambar perancangan struktur menu pada game *Me Vs Zombie*:



Gambar 3.18 Gambar Struktur Menu

3.3.1.2 Perancangan Karakter dan Musuh

Pada game *Me Vs Zombie* terdapat beberapa karakter dan musuh dalam *game.* Berikut daftar objek yang terdapat dalam game *Me Vs Zombie*:

Tabel 3.3 Tabel Karakter

Gambar	Nama	HP	Status	Damage
	Rangga	100	K (Kuat)	20
	Kerin	100	K (Kuat)	10
	Dilla	100	K (Kuat)	5

Tabel 3.4 Tabel Musuh

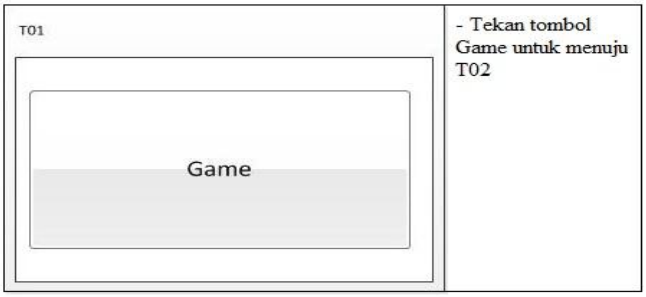
Gambar	Nama	HP	Status	Damage
	Zombie 1	100	K (Kuat)	5
	Zombie 2	100	K (Kuat)	5

	Zombie 3	100	K (Kuat)	5
	Zombie 4	100	K (Kuat)	5
	Zombie 5	100	K (Kuat)	5

3.3.1.3 Perancangan Antar Muka

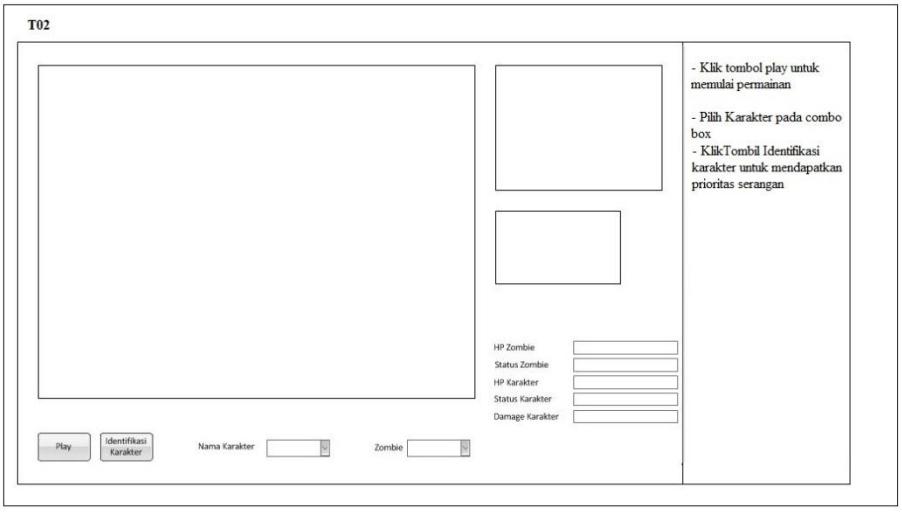
Perancangan antar muka dilakukan untuk mempermudah dalam megimplementasikan sistem yang dibangun. Berikut adalah perancangan antar muka dari aplikasi yang dibangun.

1. Tampilan Menu Utama



Gambar 3.19 Antar Muka Tampilan Menu Utama

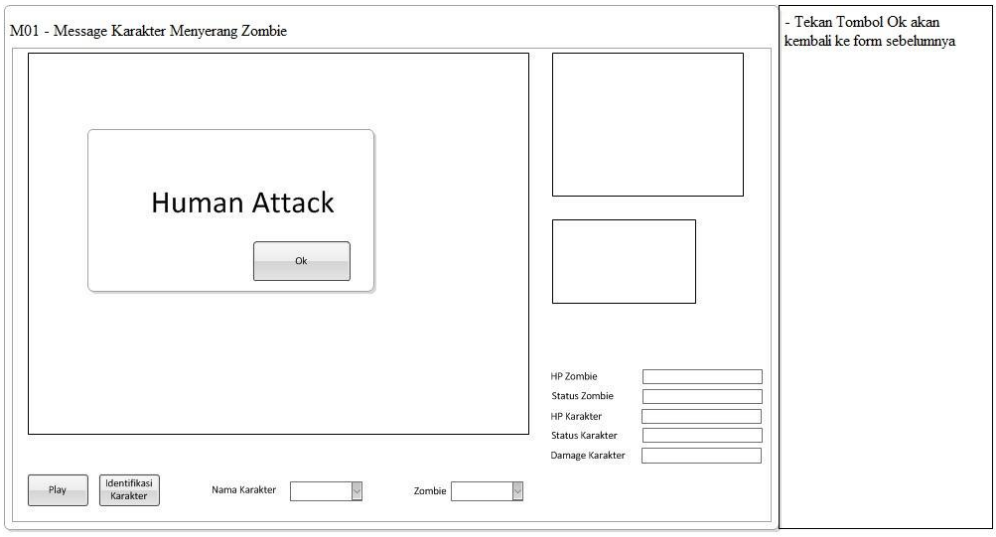
2. Antar muka saat bermain



Gambar 3.20 Antar Muka Untuk Iput Karakter dan Zombie

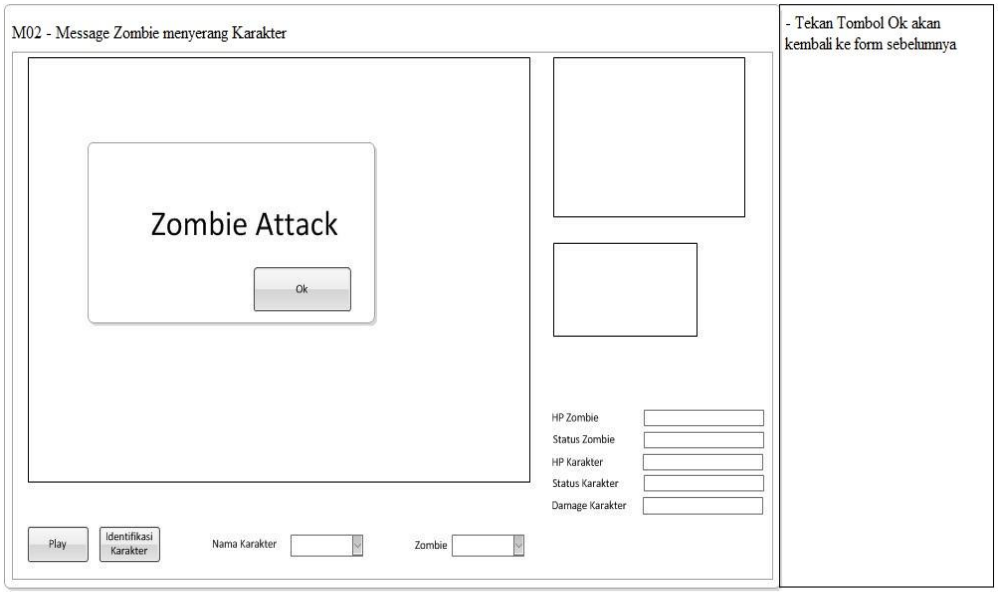
3.2.1.4 Perancangan Pesan

1. Tampilan pesan penyerangan karakter ke *zombie*.



Gambar 3.21 Tampilan Message 01

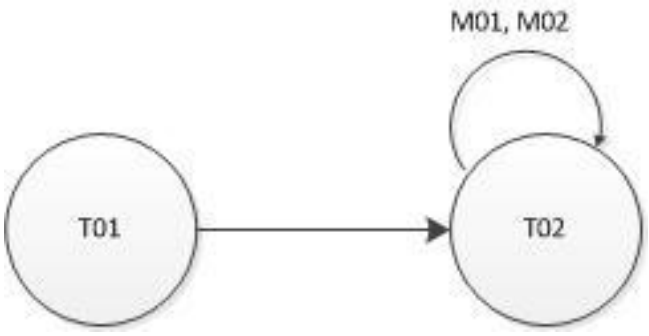
2. Tampilan pesan penyerangan *zombie* ke karakter.



Gambar 3.22 Tampilan Message 02

3.3.2 Jaringan Semantik

Struktur aplikasi dibuat secara moduler, dimana program dipecah menjadi modul-modul kecil yang mudah dibuat, mudah dites dan mudah untuk dimodifikasi. Dalam pembuatan *game* ini menggunakan alat bantu bagan struktural yang disebut jaringan semantik. Berikut adalah jaringan semantik dari *game* yang dibuat pada gambar 3.22.

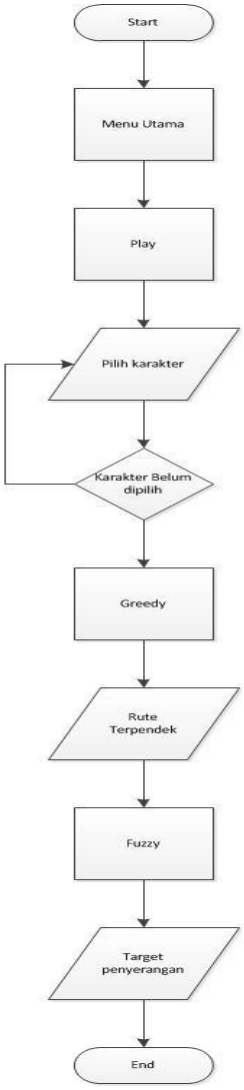


Gambar 3.23 Jaringan Semantik

3.3.3 Perancangan Prosedural

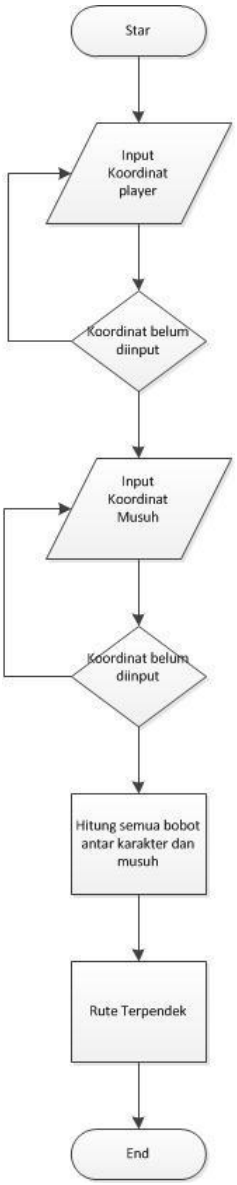
Perancangan prosedural mentransformasi elemen-elemen struktural dari arsitektur program kedalam suatu deskripsi prosedural dari komponen-komponen perangkat lunak. Adapun perancangan prosedural untuk *game* yang akan dibangun adalah sebagai berikut :

a. Prosedur *Game*



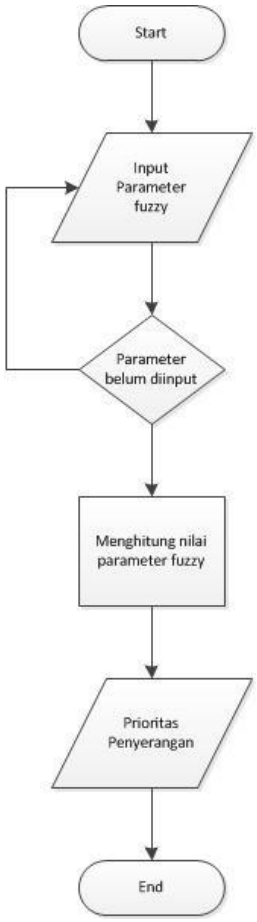
Gambar 3.24 Prosedur Game

b. Posedur Greedy



Gambar 3.25 Prosedure Proses Greedy

c. Prosedur Fuzzy



Gambar 3.24 Prosedur Fuzzy