BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Implementasi merupakan tahap pembuatan perangkat lunak serta merupakan tahap selanjutnya dari kegiatan perancangan aplikasi sebelumnya, tahap ini merupakan tahap dimana aplikasi siap untuk dioperasikan. Aplikasi ini diimplementasikan dengan tema Petani Membasmi Hama dengan file *extension-nya* .exe.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang dapat digunakan untuk menjalankan aplikasi game ini tidak harus memiliki spesifikasi yang tinggi. Spesifikasi perangkat keras yang digunakan untuk implementasi sebagai berikut :

1. Processor Intel Dual Core T4200 2GHz
2. Memori 2 GB
3. VGA card NVIDIA GEFORCE G102M 512MB
4. Monitor 1366 x 768 pixel
5. Mouse dan keyboard
6. Speaker

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang dapat digunakan untuk menjalankan dan menguji aplikasi *game* ini memiliki spesifikasi sebagai berikut :

1. Sistem Operasi *Windows* 7
2. *Flash Player* 11

4.1.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan menampilkan setiap tampilan sistem yang dibangun dengan pengkodeannya dalam file program. Pada bagian ini

akan ditampilkan implementasi antarmuka menu utama, menu Mulai, menu Pengaturan, Menu Bantuan. Implementasi antar muka menu utama bisa dilihat pada gambar 4.1.



Gambar 4.1. Antarmuka Menu Utama

4.1.3.1 Implementasi Antarmuka Menu Pengaturan

Berikut adalah Implementasi Antarmuka Menu Pengaturan yang ditunjukan oleh gambar 4.2.



Gambar 4.2. Antarmuka Menu Pengaturan

4.1.3.2 Implementasi Antarmuka Menu Bantuan

Berikut adalah Implementasi Antarmuka Menu Bantuan yang ditunjukan oleh gambar 4.3.



Gambar 4.3. Antarmuka Menu Bantuan

4.1.3.3 Implementasi Antarmuka Menu Main

Berikut adalah Implementasi Antarmuka Menu Main yang ditunjukan oleh gambar di bawah ini dimana terdapat antarmuka level 1, antarmuka level 2 dan antarmuka level 3.

a. Antarmuka Level 1

Antarmuka level 1 merupakan tampilan pada saat memainkan permainan level 1. Tampilan antarmuka level 1 dapat dilihat pada gambar 4.4.



Gambar 4.4. Antarmuka Level 1

b. Antarmuka Level 2

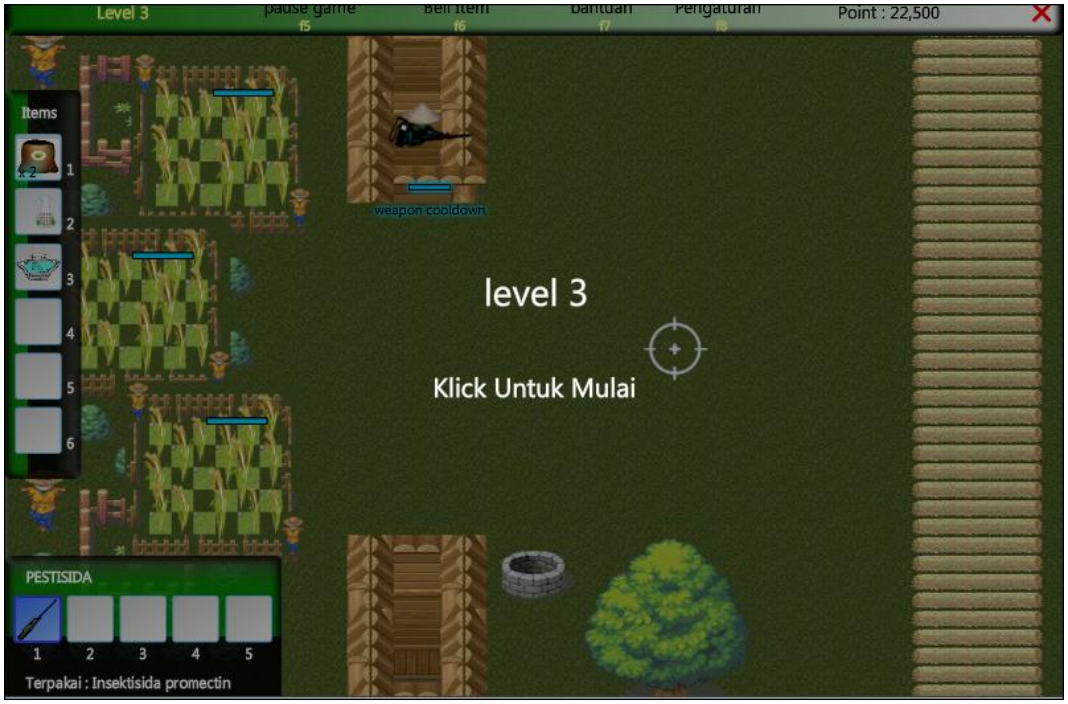
Antarmuka level 2 merupakan tampilan pada saat memainkan level Tampilan antarmuka level 2 dapat dilihat pada gambar 4.5.



Gambar 4.5. Antarmuka Level 2

c. Antarmuka Level 3

Antarmuka level 3 merupakan tampilan pada saat memainkan level 3. Tampilan antarmuka level 3 dapat dilihat pada gambar 4.6.



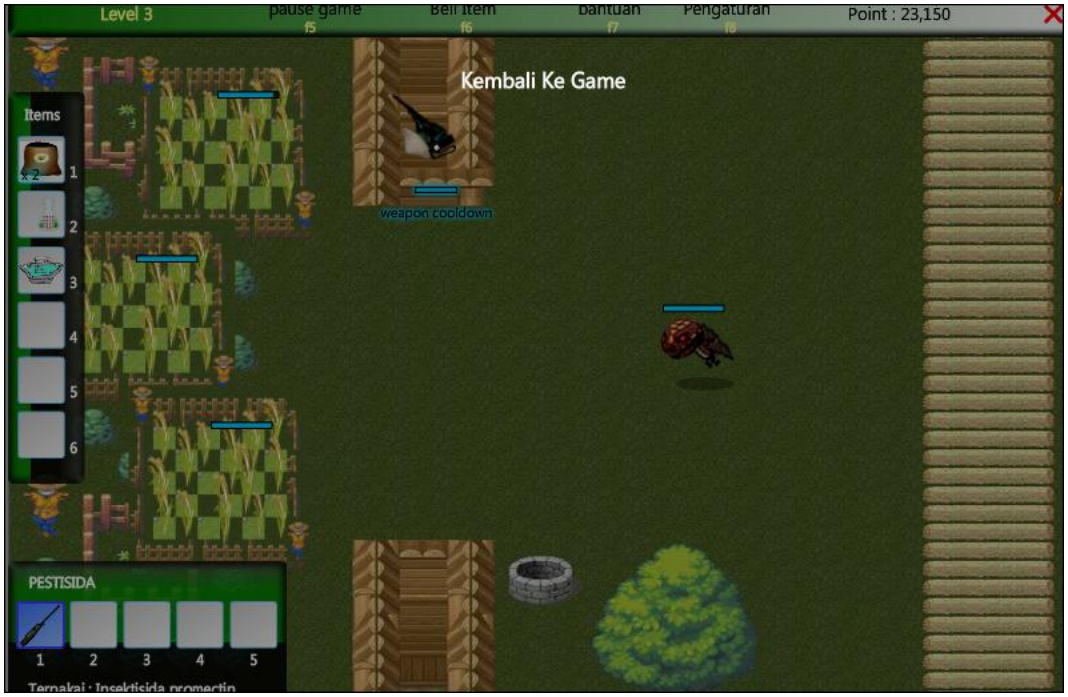
Gambar 4.6. Antarmuka Level 3

4.1.4 Implementasi Pesan

Pada bagian implementasi pesan ini akan ditampilkan implementasi pesan *pause menu, game over* menu ,*level* selesai dan misi selesai.

4.1.4.1 Implementasi Pesan *Pause* Menu

Berikut adalah Implementasi Pesan *Pause* Menu yang ditunjukkan oleh gambar 4.7.



Gambar 4.7. Pesan *Pause* Menu

4.1.4.2 Implementasi Pesan *Game over* Menu

Berikut adalah Implementasi Pesan *Gameover* Menu yang ditunjukkan oleh gambar 4.8.



Gambar 4.8. Pesan *Gameover* Menu

4.1.4.3 Implementasi Pesan *Level* Selesai

Berikut adalah Implementasi Pesan *Level* Selesai yang ditunjukkan oleh gambar 4.9.



Gambar 4.9. Pesan *Level* Selesai

4.1.4.4 Implementasi Pesan *Game* Selesai

Berikut adalah Implementasi Pesan *Game* Selesai yang ditunjukkan oleh gambar 4.10.



Gambar 4.10. Pesan *Game* Selesai

4.2 Pengujian Sistem

Pada tahap pengujian sistem ada 2 (dua) macam teknik pengujian yaitu *white box testing* dan *black box testing*. Pengujian yang menggunakan teknik *white box testing* lebih menguji pada struktur internal (*source code*) program, di mana setiap baris kode yang akan di *compile* dicek satu persatu untuk mengetahui adanya kesalahan atau tidak . Pengujian pada *black box testing* tidak terlalu memperhatikan struktur internal (*source code*) dari program, tetapi lebih kepada menemukan kesalahan program. Pada *game* Petani membasmi hama digunakan pengujian *black box testing*. Cara pengujian dilakukan melalui dua tahap, yaitu pengujian *alpha* (fungsional) dan pengujian *beta*

4.2.1 Pengujian *Alpha*

Pengujian *Alpha* adalah salah satu pengujian fungsional yang dipakai dalam menguji sistem yang telah dibuat dengan metode pengujian. Pengujian *alpha* dilakukan dengan menggunakan metode *white box* dan *black box.* Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Skenario pengujian dari *game* Petani membasmi hama dapat dilihat pada table 4.1

Tabel 4. 1 Skenario pengujian *Alpha*

No	Komponen yang diuji	Skenario pengujian	Hasil Pengujian
1	Menu Utama	Memilih tombol Mulai	*Black Box*
		Memilih tombol Pengaturan	*Black Box*
		Memilih tombol Bantuan	*Black Box*
2	Keluar Dari *Game*	Memilih tombol *close*	*Black box*
3	Memainkan *Level*	Memilih teks mulai *level*	*Black box*
		Karakter bergerak mengikuti bidikan	*Black box*
		Karakter menembak	*Black box*

		Musuh terkena senjata	*Black box*
		Musuh mati	*Black box*
		Memilih Teks Beli Item	*Black box*
		Padi diserang musuh	*Black box*
		Memilih Item Perlengkapan	*Black box*
4	Ganti senjata	Tekan tombol 1+*Shift*	*Black box*
		Tekan tombol 2+*Shift*	*Black box*
		Tekan tombol 3+*Shift*	*Black box*
		Tekan tombol 4+*Shift*	*Black box*
		Tekan tombol 5+*Shift*	*Black box*
5	*Pause game*	Memilih tombol kembali	*Black box*
6	*Game Over*	Memilih tombol kembali	*Black box*
7	Misi Selesai	Memilih tombol kembali	*Black box*
8	Algoritma *Fuzzy logic*	Pergerakan pada Hama	*White box*

4.2.1.1 Kasus dan hasil pengujian *Black box*

Berikut ini adalah hasil dari pengujian perangkat lunak yang sudah dibangun menggunakan metode *black box* berdasarkan pengamatan yang sudah dilakukan dan menghasilkan kesimpulan dari pengujian tersebut.

Tabel 4.2 Hasil Pengujian *Black Box*

No	Kasus yang akan diuji	Skenario dan Hasil Pengujian		
		Skenario uji	**Hasil yang diharapkan**	**Kesimpulan**
1.	Menu Utama	Memilih tombol Mulai	Sistem menampilkan *level* permainan	[√] Berhasil [ ] Tidak berhasil
		Memilih tombol Pengaturan	Sistem menampilkan halaman pengaturan suara	[√] Berhasil [ ] Tidak berhasil

		Memilih tombol Bantuan	Sistem menampilkan halaman bantuan *game*	[√] Berhasil [ ] Tidak berhasil
2	Keluar Dari *Game*	Memilih tombol *close*	Sistem akan menutup aplikasi	[√] Berhasil [ ] Tidak berhasil
3	Memainkan *Level*	Memilih teks mulai *level*	Sistem menjalankan misi *level*	[√] Berhasil [ ] Tidak berhasil
		Karakter bergerak mengikuti bidikan	Sistem menggerakan karakter sesuai gerakan *mouse*	[√] Berhasil [ ] Tidak berhasil
		Karakter menembak	Sistem menampilkan efek tembakan	[√] Berhasil [ ] Tidak berhasil
		Musuh terkena senjata	Sistem manampilkan darah musuh berkurang	[√] Berhasil [ ] Tidak berhasil
		Musuh mati	Sistem menampilkan point bertambah	[√] Berhasil [ ] Tidak berhasil
		Memilih Teks Beli Item	Sistem menampilkan *inventory shop*	[√] Berhasil [ ] Tidak berhasil
		Padi diserang musuh	Sistem menampilkan Kekuatan padi berkurang	[√] Berhasil [ ] Tidak berhasil
		Memilih Item Perlengkapan	Sistem menampikan item perlengkapan terseleksi	[√] Berhasil [ ] Tidak berhasil
4	Ganti senjata	Tekan tombol 1+*Shift*	Karakter menggunakan senjata	[√] Berhasil [ ] Tidak berhasil
		Tekan tombol 2+*Shift*	Karakter menggunakan senjata	[√] Berhasil [ ] Tidak berhasil
		Tekan tombol	Karakter	[√] Berhasil

		3+*Shift*	menggunakan senjata	[ ] Tidak berhasil
		Tekan tombol 4+*Shift*	Karakter menggunakan senjata	[√] Berhasil [ ] Tidak berhasil
		Tekan tombol 5+*Shift*	Karakter menggunakan senjata	[√] Berhasil [ ] Tidak berhasil
5	*Pause game*	Memilih tombol kembali	Menampilkan *game* berjalan kembali	[√] Berhasil [ ] Tidak berhasil
6	*Game Over*	Memilih tombol kembali	Menampilkan *game* kembali ke menu utama	[√] Berhasil [ ] Tidak berhasil
7	Misi Selesai	Memilih tombol kembali	Menampilkan *game* kembali ke menu utama	[√] Berhasil [ ] Tidak berhasil

4.2.1.2 Kesimpulan Pengujian *Black-Box*

Berdasarkan hasil pengujian terhadap butir uji yang telah dilakukan memberikan kesimpulan bahwa proses sudah benar. Pesan kesalahan yang ada sudah tampil cukup maksimal. Secara fungsionalitas *game* sudah dapat menghasilkan *output* yang diharapkan.

4.2.1.3 Kasus dan Hasil Pengujian (*White Box Testing*)

Berikut ini merupakan kasus algoritma *fuzzy logic* untuk menguji perangkat lunak yang sudah dibangun dengan menggunakan metode *white box*, berdasarkan skenario pengujian aplikasi yang sudah dijabarkan dan penerapannya menggunakan bahasa *actionscript 3.0* :

Tabel 4.3. *Source Code* Algoritma *Fuzzy logic*

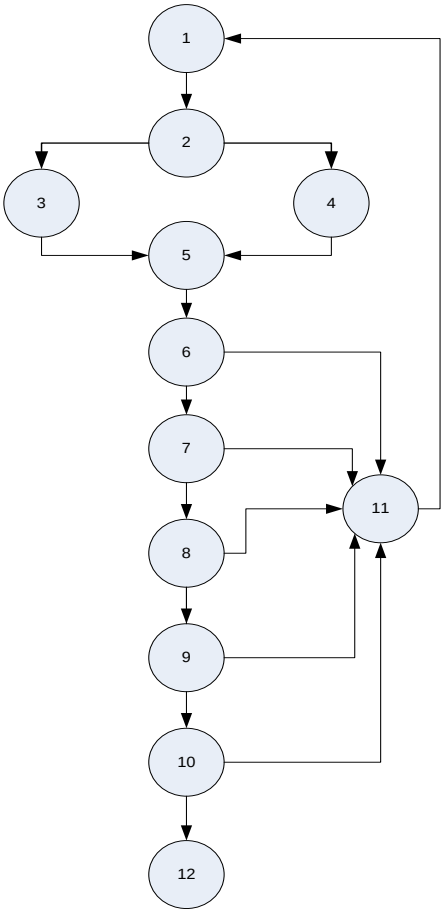
Pseudocode
1. `fuz=new Fuzzyficator();` 2. `manifold=new FuzzyManifold("Distance_to_Target","hp","moveSpd");`

```
3.    var movSpdStatusInput:FuzzyInput=new Fuzzy Input()
      manifold.input = movSpdStatusInput;
      func = factory.create(FuzzyMembershipFunction.enemy0,
      "slow", "0,15,25" );
      manifold.addMember(func);
      func = factory.create(FuzzyMembershipFunction.enemy1,
      "Medium", "30, 45 ,70" );
      manifold.addMember(func);
      func = factory.create(FuzzyMembershipFunction.enemy2,
      "Fast", "80, 90,100",","," );
      manifold.addMember(func);
4.    var hpStatusInput:FuzzyInput=new Fuzzy Input()
      manifold.input = hpStatusInput;
      func = factory.create(FuzzyMembershipFunction.enemy, "low",
      "0,5, 15");
      manifold.addMember(func);
      func = factory.create(FuzzyMembershipFunction.enemy,
      "strong", "80, 90, 100" );
      manifold.addMember(func);
5.    rule = new FuzzyRule( "IF moveSpd IS slow AND hp IS low THEN
      Distance_to_Target IS castle50");
      fuz.addRule(rule);
6.    rule = new FuzzyRule( "IF moveSpd IS slow AND hp IS strong
      THEN Distance_to_Target IS castle100 ");
      fuz.addRule(rule);
7.    rule = new FuzzyRule( "IF moveSpd IS medium AND hp IS low
      THEN Distance_to_Target IS castle50 ");
      fuz.addRule(rule);
8.    rule = new FuzzyRule( " IF moveSpd IS medium AND hp IS
      strong THEN Distance_to_Target IS castle100");
      fuz.addRule(rule);
9.    rule = new FuzzyRule( " IF moveSpd IS fast AND hp IS strong
      THEN Distance_to_Target IS castle75");
      fuz.addRule(rule);
10.   rule = new FuzzyRule( " IF moveSpd IS slow AND hp IS low
      THEN Distance_to_Target IS castle75");
      fuz.addRule(rule);
```

```
11.   fuz.addRule(rule,false);
      hpStatusInput.value = 8;
      moveSodStatusInput.value = 200
      fuz.Fuzzify()
12.   fuzzyManifolds =
      fuz.Defuzzify(DefuzzificationMethod.Average);
      Trace("OUTPUTCOG:"+FuzzyManifold(fuzzyManifolds
      ["Distance_to_Target"]).output);
```

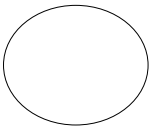
a. Flow Graph

Berikut merupakan *Flow Graph* dari algoritma *fuzzy logic* yang dapat dilihat pada gambar berikut.



Gambar 4.11. *Flow Graph fuzzy logic*

Keterangan :

 = Menggambarkan *node*/simpul

 = *edges*/panah

b. *Cyclomatic Complexity*

E = 17, N = 12

V(G) = E – N + 2

V(G) = 17 – 12 + 2

V(G) = 7

Keterangan :

E : Jumlah busur pada *flowgraph*

N : Jumlah simpul pada *flowgraph*

V(G) : *Cyclomatic Complexity*

c. *Independent Path*

1. *Path* 1 : 1-2-3-4-5-6-7-8-9-10-11-12
2. *Path* 2 : 1-2-3-4-5-11-12
3. *Path* 3 : 1-2-3-4-5-6-11-12
4. *Path* 4 : 1-2-3-4-5-6-7-11-12
5. *Path* 5 : 1-2-3-4-5-6-7-8-11-12
6. *Path* 6 : 1-2-3-4-5-6-7-8-9-11-12
7. *Path* 7 : 1-2-3-4-5-6-7-8-9-10-11-12

d. ***Graph Matriks Testing***

Tabel 4. 4 ***Graph Matriks Algoritma Fuzzy Logic***

N	1	2	3	4	5	6	7	8	9	10	11	12	N E-1
1		1											1-1=0
2			1	1									2-1=1
3					1								1-1=0
4					1								1-1=0
5						1					1		2-1=1
6							1				1		2-1=1
7								1			1		2-1=1
8									1		1		2-1=1
9										1	1		2-1=1
10												1	1-1=0
11												1	1-1=0
12													1-1=0
Sum E+1													6+1=7

Keterangan:

1. Baris dan kolom merepresentasikan simpul.

2. Nilai 1 merepresentasikan adanya keterhubungan antar *node*/simpul. yang dibuat sesuai alur logika dari program tersebut. penggunaan kondisi memengaruhi jumlah *node* dan edge dari *flow graph* tersebut.

a. Baris 1 dan kolom 2 = merepresentasikan adanya keterhubungan antar *node*/simpul 1 dan *node*/simpul 2 dari *flow graph*.
b. Baris 2 dan kolom 3 dan 4 = merepresentasikan adanya keterhubungan antar *node*/simpul 2 dan *node*/simpul 3 dan 4 dari *flow graph*.
c. Baris 3 dan kolom 5 = merepresentasikan adanya keterhubungan antar *node*/simpul 3 dan *node*/simpul 5 dari *flow graph.*

d. Baris 4 dan kolom 5 = merepresentasikan adanya keterhubungan antar *node*/simpul 4 dan *node*/simpul 5dari *flow graph*.
e. Baris 5 dan kolom 6 dan 11 = merepresentasikan adanya keterhubungan antar *node*/simpul 5 dan *node*/simpul 6 dan 11 dari *flow graph*.
f. Baris 6 dan kolom 7 dan 11 = merepresentasikan adanya keterhubungan antar *node*/simpul 6 dan *node*/simpul 7dan 11 dari *flow graph*.
g. Baris 7 dan kolom 8 dan 11 = merepresentasikan adanya keterhubungan antar *node*/simpul 7 dan *node*/simpul 8 dan11 dari *flow graph.*
h. Baris 8 dan kolom 9 dan 11 = merepresentasikan adanya keterhubungan antar *node*/simpul 8 dan *node*/simpul 9 dan 11 dari *flow graph*.
i. Baris 9 dan kolom 10 dan 11 = merepresentasikan adanya keterhubungan antar *node*/simpul 9 dan *node*/simpul 10 dan 11 dari *flow graph.*
j. Baris 10 dan kolom 12 = merepresentasikan adanya keterhubungan antar *node*/simpul 10 dan *node*/simpul 12 dari *flow graph*.
k. Baris 11 dan kolom 12 = merepresentasikan adanya keterhubungan antar *node*/simpul 11 dan *node*/simpul 12 dari *flow graph*.

c. Hasil Uji Algoritma *Fuzzy Logic*.

Tabel 4. 5 Hasil Uji Algoritma *Fuzzy Logic*

No	Kasus Uji	Pengamatan	Yang diharapkan	Kesimpulan
1	`fuz=new Fuzzyficator(); manifold=new FuzzyManifold("Distance_to_Target","hp","moveSpd"); rule = new FuzzyRule( "IF moveSpd IS slow AND hp IS low THEN Distance_to_Target IS castle50"); fuz.addRule(rule);`	Rule kecepatan lambat AND kekuatan lemah	NPC menyerang padi 50 Ton	[√] dikerjakan [ ] tidak dikerjakan

2	`rule = new FuzzyRule( "IF moveSpd IS slow AND hp IS strong THEN Distance_to_Target IS castle100 "); fuz.addRule(rule);`	Rule kecepatan lambat AND kekuatan kuat	NPC menyerang padi 100 Ton	[√] dikerjakan [ ] tidak dikerjakan
3	`rule = new FuzzyRule( "IF moveSpd IS medium AND hp IS low THEN Distance_to_Target IS castle50 "); fuz.addRule(rule);`	Rule kecepatan sedang AND kekuatan lemah	NPC menyerang padi 50 Ton	[√] dikerjakan [ ] tidak dikerjakan
4	`rule = new FuzzyRule( " IF moveSpd IS medium AND hp IS strong THEN Distance_to_Target IS castle100"); fuz.addRule(rule);`	Rule kecepatan sedang AND kekuatan kuat	NPC menyerang padi 100 Ton	[√] dikerjakan [ ] tidak dikerjakan
5	`rule = new FuzzyRule( " IF moveSpd IS fast AND hp IS strong THEN Distance_to_Target IS castle75"); fuz.addRule(rule);`	Rule kecepatan Cepat AND kekuatan kuat	NPC menyerang padi 75 Ton	[√] dikerjakan [ ] tidak dikerjakan
6	`rule = new FuzzyRule( " IF moveSpd IS slow AND hp IS low THEN Distance_to_Target IS castle75"); fuz.addRule(rule);`	Rule kecepatan cepat AND kekuatan lemah	NPC menyerang padi 75 Ton	[√] dikerjakan [ ] tidak dikerjakan

4.2.1.4 Kesimpulan Pengujian *Alpha*

Dari hasil pengujian yang dilakukan, dapat disimpulkan bahwa sistem sudah berjalan seperti yang diharapkan dan secara fungsional sudah dapat menghasilkan keluaran yang diharapkan, akan tetapi pada prosesnya masih

memungkinkan untuk terjadinya kesalahan, namun frekuensi kesalahan masih relatif kecil.

4.2.2 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif, dimana dilakukan pengujian secara langsung terhadap pengguna dengan menggunakan kuesioner mengenai kepuasan pengguna atas aplikasi yang telah dibangun. Adapun metode penilaian pengujian yang digunakan adalah metode kuantitatif berdasarkan data dari pengguna. Kuesioner disebarkan secara acak tanpa memperhatikan strata yang ada dalam populasi. Kuesioner disebarkan kepada 35 orang dari umur 10 tahun ke atas, dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang telah dibangun. Kuesioner terdiri dari 5 pernyataan dengan menggunakan skala 1 sampai 5 (skala likert). Untuk mencari nilai persentase dari masing-masing jawaban kuesioner digunakan rumus skala likert sebagai berikut :

$$P=B/Q*100\% \qquad (4.\ 1)$$

Keterangan :

1). P= Nilai persentase yang dicari.

2). B= Bobot setiap pilihan jawaban.

3). Q= Nilai tertinggi dikalikan dengan jumlah sampel.

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas.

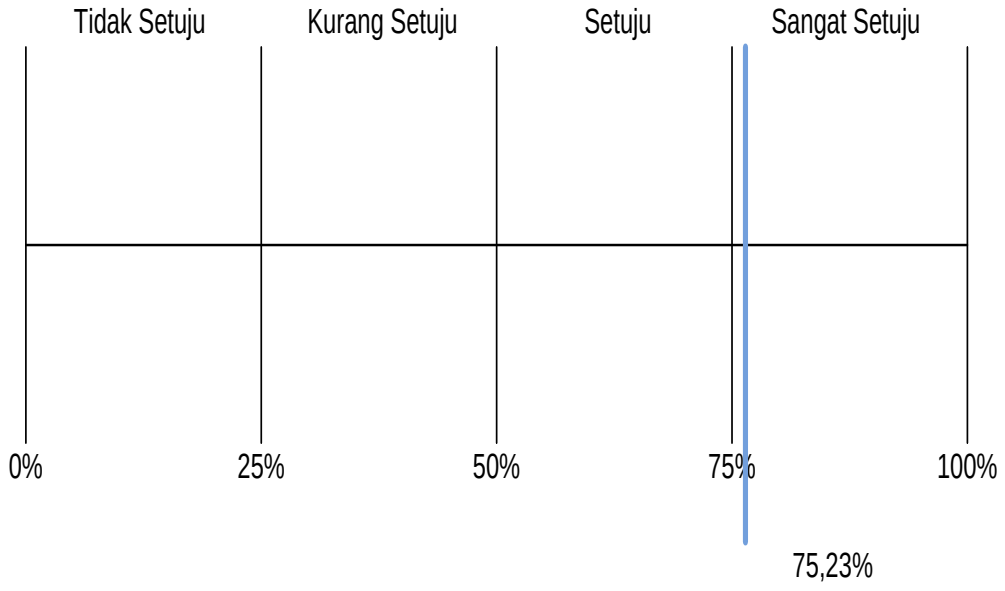
1. Menurut Anda apakah *game* yang mengenalkan tentang perjuangan petani di Indonesia bisa menjadi sarana alternatif dalam mengangkat karakter asal Indonesia ?

Tabel 4. 6 Perhitungan persentase pertanyaan pertama

Pertanyaan	Keterangan	Skor	Responden	Jumlah
1	Sangat setuju	3	13	39
	Setuju	2	19	38
	Kurang setuju	1	2	2
	Tidak Setuju	0	1	0
Jumlah			**35**	**79**

P= 79/105*100 = 75,23%

Dapat disimpulkan bahwa penilai terhadap pertanyaan 1 pada *game* Petani Pembasmi hama adalah **75,23**% dari yang diharapkan 100%, atau dapat dikategorikan sebagai sangat setuju. Untuk lebih jelasnya dapat dilihat pada gambar 4.12.



Gambar 4.12 Interpretasi Skor Pertanyaan No 1

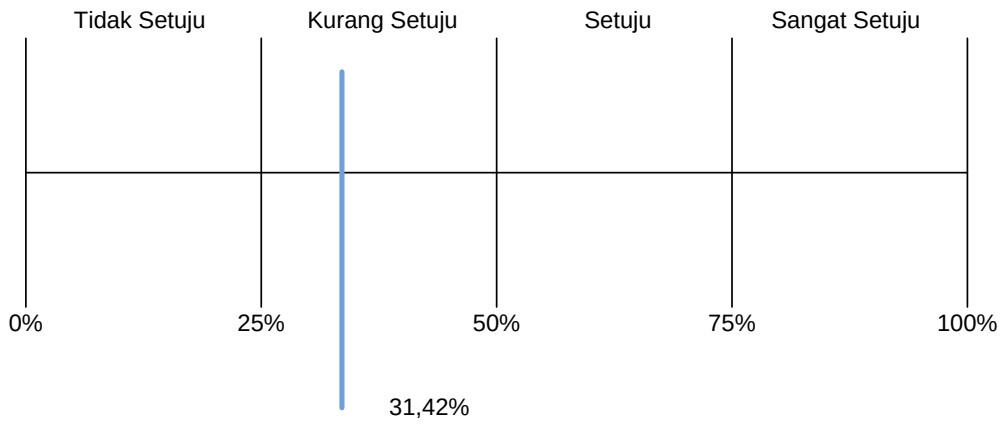
2. Menurut anda apakah *game* tentang pertanian yang sudah ada, lebih menantang ?

Tabel 4. 7 Perhitungan persentase pertanyaan kedua

Pertanyaan	Keterangan	Skor	Responden	Jumlah
2	Sangat setuju	3	9	27
	Setuju	2	1	2
	Kurang setuju	1	4	4
	Tidak Setuju	0	21	0
Jumlah			**35**	**33**

P= 33/105*100 = 31,42%

Dapat disimpulkan bahwa penilai terhadap pertanyaan 2 pada *game* Petani Pembasmi hama adalah **31,42**% dari yang diharapkan 100%, atau dapat dikategorikan sebagai sangat setuju. Untuk lebih jelasnya dapat dilihat pada gambar 4.13.



Gambar 4.13 Interpretasi Skor Pertanyaan No 2

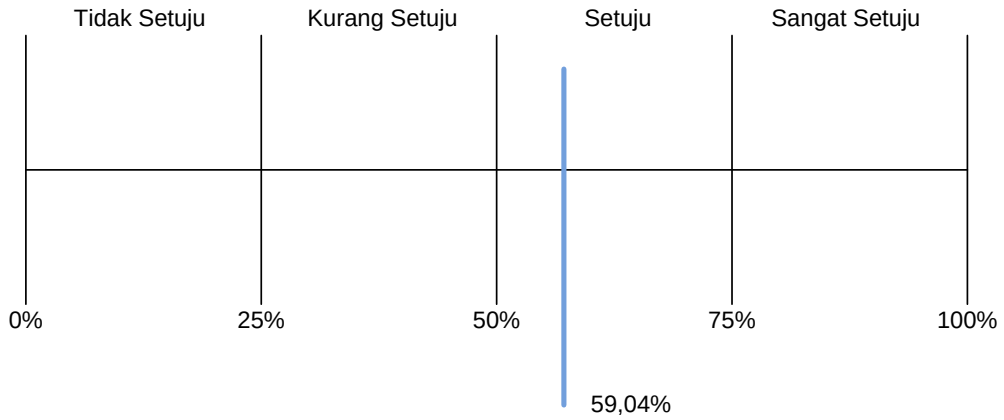
3. Menurut Anda apakah tampilan pada game perjuangan petani Indonesia mudah difahami?

Tabel 4. 8 Perhitungan persentase pertanyaan ketiga

Pertanyaan	Keterangan	Skor	Responden	Jumlah
3	Sangat setuju	3	6	18
	Setuju	2	18	36
	Kurang setuju	1	8	8
	Tidak Setuju	0	3	0
Jumlah			**35**	**62**

P= 62/105*100 = 59,04%

Dapat disimpulkan bahwa penilai terhadap pertanyaan 3 pada *game* Petani Pembasmi hama adalah **59,04**% dari yang diharapkan 100%, atau dapat dikategorikan sebagai setuju. Untuk lebih jelasnya dapat dilihat pada gambar 4.14.



Gambar 4.14 Interpretasi Skor Pertanyaan No 3

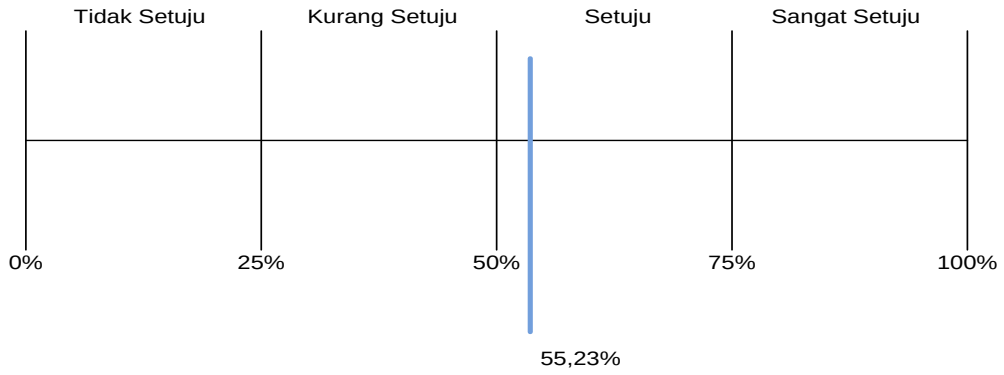
4. Pada game perjuangan petani Indonesia , apakah pergerakan musuhnya menyulitkan Anda?

Tabel 4. 9 Perhitungan persentase pertanyaan empat

Pertanyaan	Keterangan	Skor	Responden	Jumlah
4	Sangat setuju	3	6	18
	Setuju	2	15	30
	Kurang setuju	1	10	10
	Tidak Setuju	0	4	0
Jumlah			**35**	**58**

P= 58/105*100 = 55,23%

Dapat disimpulkan bahwa penilai terhadap pertanyaan 4 pada *game* Petani Pembasmi hama adalah **55,23**% dari yang diharapkan 100%, atau dapat dikategorikan sebagai setuju. Untuk lebih jelasnya dapat dilihat pada gambar 4.15.



Gambar 4.15 Interpretasi Skor Pertanyaan No 4

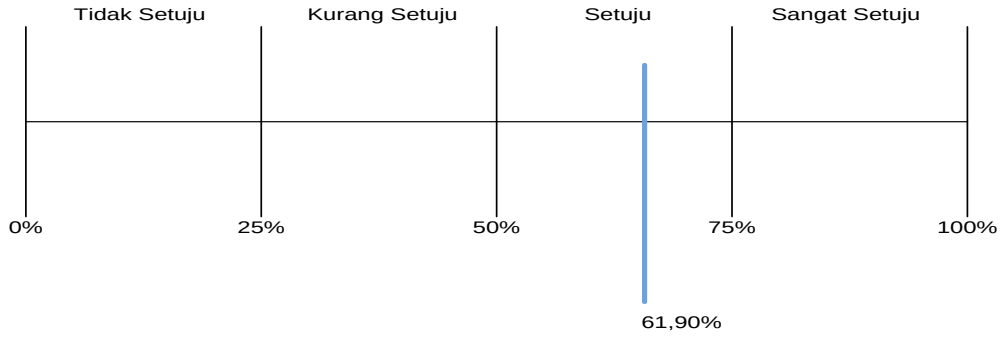
5. Setelah Memainkan game Perjuangan Petani Indonesia, menurut anda apakah game ini perlu dikembangkan ?

Tabel 4. 10 Perhitungan persentase pertanyaan lima

Pertanyaan	Keterangan	Skor	Responden	Jumlah
5	Sangat setuju	3	15	30
	Setuju	2	17	34
	Kurang setuju	1	1	1
	Tidak Setuju	0	2	0
Jumlah			**35**	**65**

P= 65/105*100 = 61.90%

Dapat disimpulkan bahwa penilai terhadap pertanyaan 5 pada *game* Petani Pembasmi hama adalah **61.90**% dari yang diharapkan 100%, atau dapat dikategorikan sebagai sangat setuju. Untuk lebih jelasnya dapat dilihat pada gambar 4.16.



Gambar 4.16 Interpretasi Skor Pertanyaan No 5

4.2.2.1 Kesimpulan keseluruhan Pengujian *Beta*

Berdasarkan dari jawaban yang diberikan oleh responden, bisa ditarik kesimpulan bahwa :

1. *Game* Perjuangan Petani Indonesia Membasmi hama menjadi sarana alternatif dalam mengangkat karakter asal Indonesia dan dinyatakan sangat Setuju oleh responden dengan nilai 75,23%.

2. untuk game bertema pertanian yang sudah ada saat ini apakah sudah menantang permainannya dinyatakan kurang setuju oleh responden dengan nilai 31,42%

3. Tampilan game yang fleksibel dapat memudahkan pemain dalam mengetahui cara memain dapat udah difahami ,hal ini dinyatakan dengan 59,28% responden yang setuju, game yang mudah dimainkan sangat berpengaruh.

4. Pergerakan musuh dalam game perjuangan petani Indonesia dirasa sudah menyulitkan hal ini dinyatakann oleh 55,23% responden yang menjawab.

5. Sebagian besar responden menjawab game ini perlu dikembangkan, hal ini dinyatan oleh responden dengan nilai 61.10%