BAB IV
IMPLEMENTASI DAN PENGUJIAN

IV.1 Implementasi Sistem

Implementasi system adalah tahap pengujian kelayakan pada system yang dibuat sehingga aplikasi yang dibuat tidak menyimpang dari perancangan sistemnya. *Game* ini dibuat menggunakan *Game Maker* 8.0. perangkat keras yang digunakan untuk mengimplementasikan dan menguji *game* ini memiliki spesifikasi sebagai berikut :

1. Processor core i3 dengan kecepatan 2.1GHz
2. Memori 6GB
3. Harddisk 640GB
4. Video Card dengan memori 2GB
5. Monitor
6. Mouse dan Keyboard
7. Speaker

Sedangkan perangkat lunak yang digunakan untuk mengimplimentasikan dan menguji *game* ini adalah :

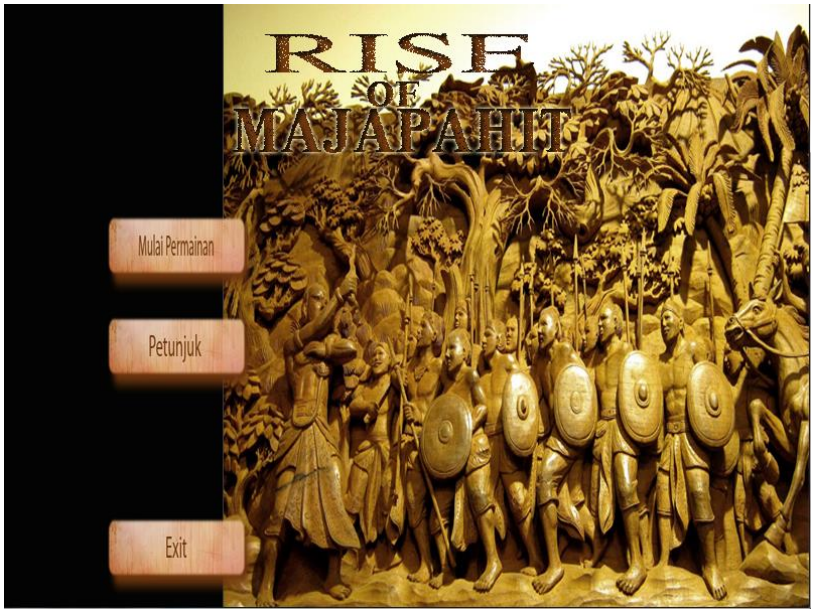
1. Windows 7 Ultimate sebagai system operasi
2. *Game Maker* 8.0 sebagai perangkat lunak pembangun *game*

IV.1.1 Implementasi Antar Muka

Implementasi antar muka adalah bagian yang menunjuk bentuk tampilan setiap antar muka pada aplikasi yang sudah dibangun. Implementasi antar muka pada *game* ini ditunjukkan pada gambar IV.1 sampai gambar IV.6

1. Antarmuka Menu Utama

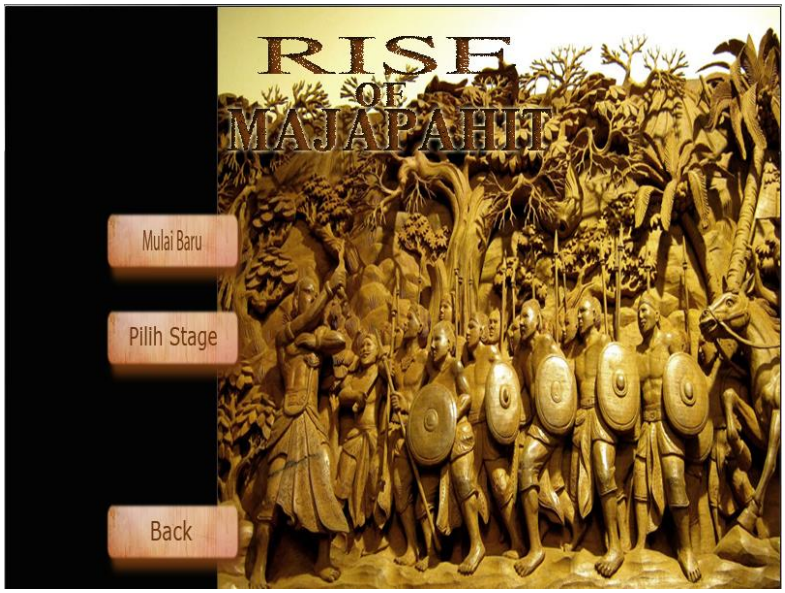
   Antarmuka menu utama ditunjukkan pada Gambar IV.1. Tombol yang tersedia adalah tombol mulai baru yang berfungsi sebagai akses menuju mulai permainan, tombol petunjuk berfungsi sebagai akses menuju petunjuk dan tombol exit berfungsi sebagai akses keluar dari *game*.



Gambar IV. 1 Antarmuka Menu Utama

2. Antarmuka Mulai Permainan

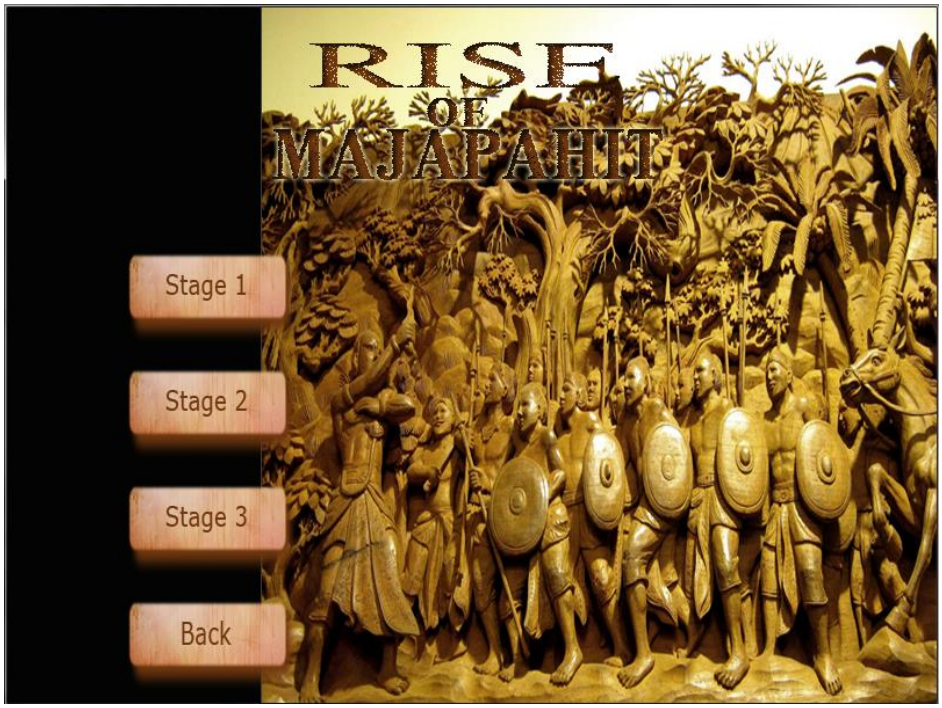
   Antarmuka Mulai Permainan ditunjukkan pada Gambar IV.2. Tombol yang tersedia adalah tombol mulai baru sebagai akses menuju stage 1, tombol stage sebagai akses menuju pilih stage dan tombol back untuk kembali ke menu utama.



Gambar IV. 2 Antarmuka Mulai Permainan

3. Antarmuka Pilih Stage

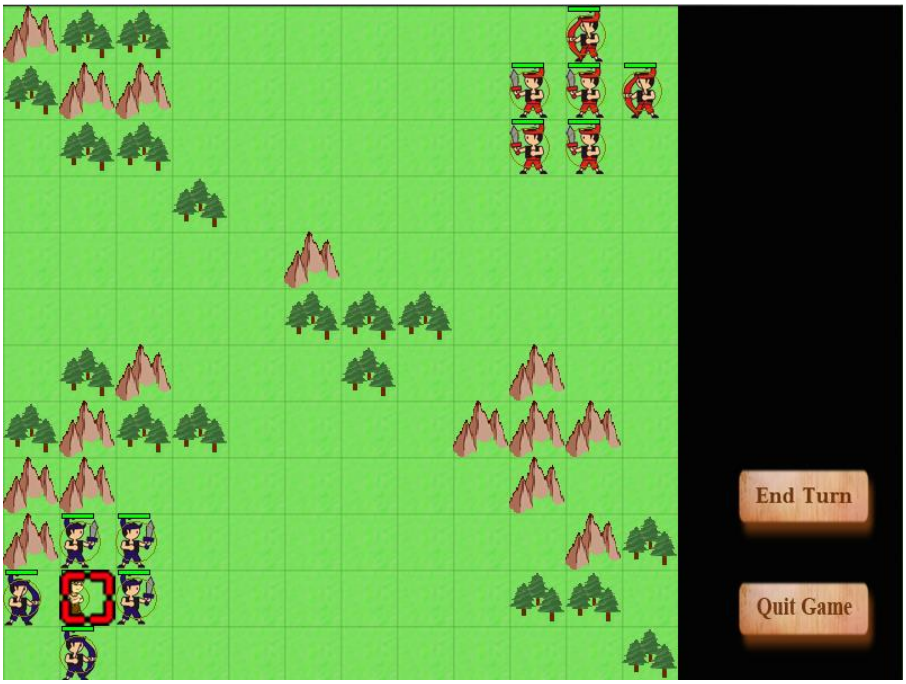
Antarmuka Pilih Stage adalah pengembangan dari form antarmuka mulai permainan, ditunjukkan oleh Gambar IV.3. Tombol yang tersedia adalah tombol stage 1 sebagai akses menuju stage 1, tombol stage 2 sebagai akses menuju stage 2, tombol stage 3 sebagai akses menuju stage 3 dan tombol back untuk kembali ke menu utama.



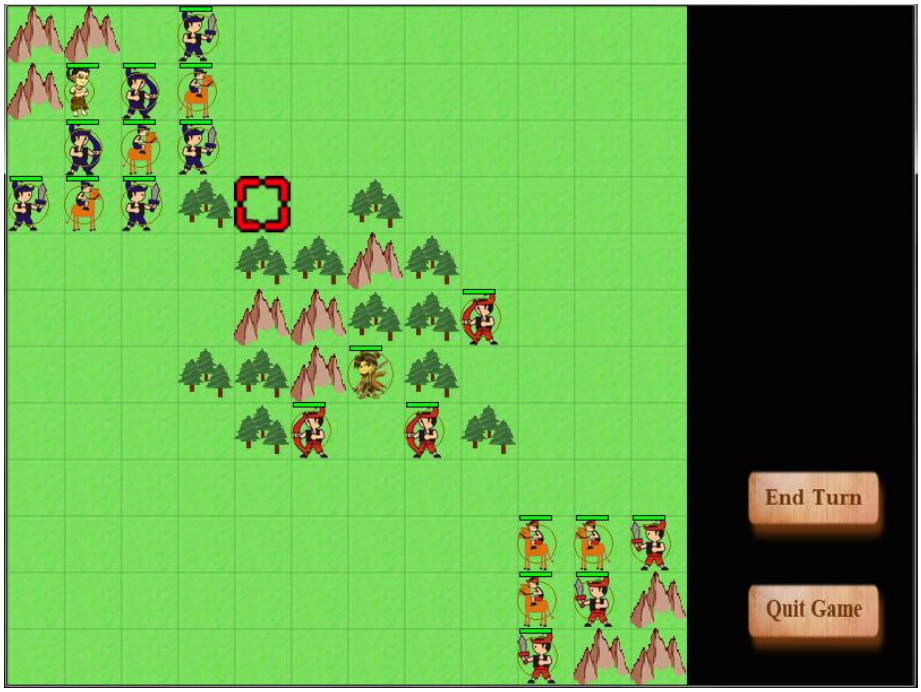
Gambar IV. 3 Antarmuka Pilih Stage

4. Antarmuka Stage

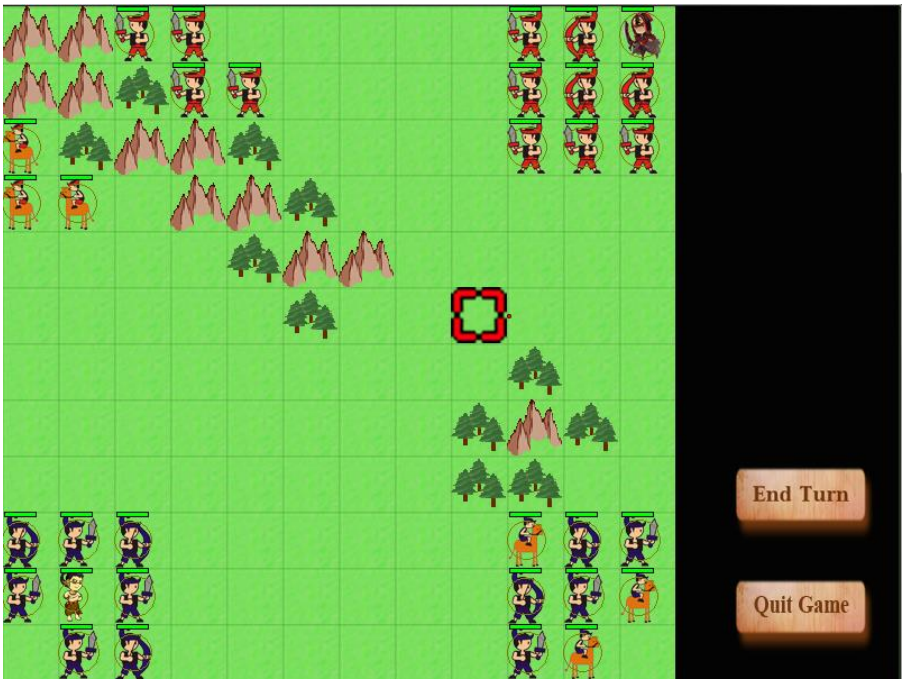
   Antarmuka Stage terdiri dari antarmuka stage 1 yang ditunjukkan pada Gambar IV.4, antarmuka stage 2 yang ditunjukkan pada Gambar IV.5 dan antarmuka stage 3 ditunjukkan pada Gambar IV.6. Tombol yang tersedia adalah tombol end turn yang berfungsi untuk mengakhiri giliran pemain dan tombol quit *game* sebagai akses untuk kembali ke menu utama. Pada antarmuka ini juga terdapat piece yang jumlahnya berbeda di setiap stage. Piece milik pemain ditunjukkan dengan warna biru dan piece milik AI ditunjukkan dengan warna merah.



Gambar IV. 4 Antarmuka Stage 1



Gambar IV. 5 Antarmuka Stage 2



Gambar IV. 6 Antarmuka Stage 3

IV.2 Pengujian Sistem

Pengujian Sistem adalah bagian penelusuran dalamrangka menemukan kesalahan pada sistem yang dibangun. Tujuan pengujian system adalah untuk mencari kesalahan pada sistem yang dibangun sebelum sistem digunakan oleh pengguna.

IV.2.1 Rencana Pengujian

Rencana pengujian perangkat lunak ini dibagi menjadi 2 bagian yaitu pengujian *black box* dan pengujian *white box*.

Pengujian *white box* dilakukan untuk menguji kesalahan logic dan asumsi yang tidak tepat pada kemungkinan eksekusi. Pengembangan kode-kode program selalu memungkinkan terjadinya alur program yang tidak dan tereksekusi dan kesalahan typography yang sulit ditemukan kalau tidak dijalankan. Pada umumnya pengujian ini dilakukan hanya pada lingkungan developer.

Pengujian *black box* dilakukan untuk mencari fungsi-fungsi program yang tidak benar atau hilang. Kesalahan lain yang mungkin terjadi dalam pembuatan

program adalah kesalahan dalam struktur data, kesalahan antar muka, inisialisasi dan akhir program dan kesalahan performansi.

Strategi pengujian akan dilakukan dengan menggunakan pengujian Beta. Pengujian beta adalah pengujian pada lingkungan pengguna. Pengujian ini dilakukan dengan menggunakan metode pengumpulan data melalui kuesioner. Pengguna akan mencoba *game* ini, kemudian pengguna akan memberikan feedback pada kuesioner yang telah disediakan . Deskripsi rencana pengujian dapat dilihat dalam Tabel IV.1.

Tabel IV. 1 Deskripsi Rencana Pengujian

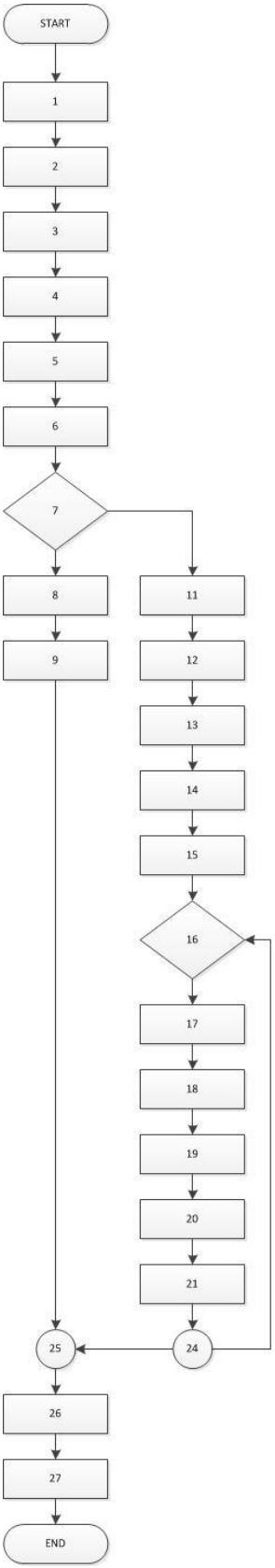
No.	Kelas Uji	Detail Pengujian	Jenis Pengujian
1	Uji algoritma minimax	Pengujian pada proses penentuan nilai keputusan pada algoritma minimax	*White box*
2	Uji antarmuka Menu Utama	Menekan tombol mulai permainan	*Black box*
		Menekan tombol Petunjuk	*Black box*
		Menekan tombol exit	*Black box*
3	Uji antarmuka Mulai Permainan	Menekan tombol Mulai Baru	*Black box*
		Menekan tombol Pilih Stage	*Black box*
		Menekan tombol Back	*Black box*
4	Uji antarmuka Pilih Stage	Menekan tombol Stage 1	*Black box*
		Menekan tombol Stage 2	*Black box*
		Menekan tombol Stage 3	*Black box*
		Menekan tombol Back	*Black box*
5	Uji antarmuka Stage	Menekan tombol end turn	*Black box*
		Menekan tombol quit	*Black box*
		Memilih piece pemain	*Black box*

IV.2.2 Pengujian *White box*

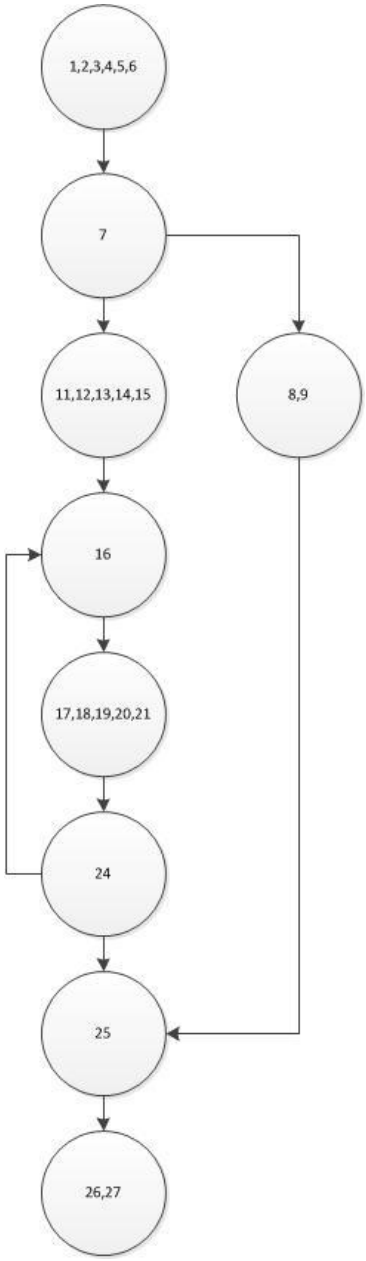
Pengujian *white box* bertujuan untuk mengetahui kinerja logika yang dibuat pada sebuah perangkat lunak apakah berjalan dengan baik atau tidak. Pengujian *white box* akan digunakan pada algoritma minimax. Pengujian *white box* dilakukan dengan beberapa tahap, diantaranya :

1. Pengubahan pseucode menjadi *flowchart* kemudian menjadi *flowgraph*.

```
1   parent ← noone
2   bestmove ← 0
3   move ← 0
4   i ← 0
5   start ← argument0
6   finish ← argument1
7   if  start = finish then
8       path ← ds_stack_create()
9       ds_stack_push(path, start)
10  else
11      open ← ds_priority_create()
12      closed ← ds_list_create()
13      ds_priority_add(open, start, start.bestMove)
14      current ← start
15      n ← ds_list_size(current.neighbors)
16      for i ← 0 to i  n-1 do
17          neighbor ← ds_list_find_value(current.neighbors, i)
18          if  neighbor.move > neighbor.bestMove then
19              neighbor.bestMove ← neighbor.move
20              if neighbor.bestMove > neighbor.move then
21                  ds_priority_add(open, neighbor, bestMove)
22              endif
23          endif
24      endfor
25  endif
26  ds_priority_destroy(open)
27  path ← ds_stack_create()
```



Gambar IV. 7 Flowchart Algoritma Minimax



Gambar IV. 8 Flowgraph Algoritma Minimax

2. Cyclometic complexity

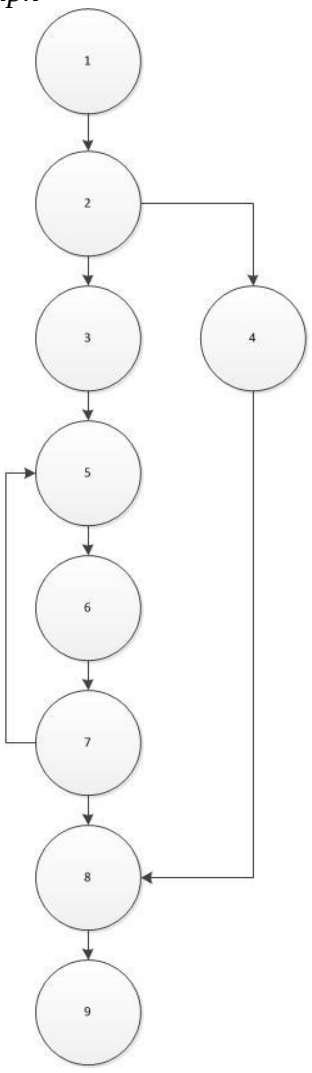
V(G) = E – N + 2
= 9 – 8 + 2
= 3

Dimana :
E : Jumlah busur pada flow graph
N : Jumlah simpul pada flow graph

3. Jumlah region
Jumlah Region = Jumlah Kurva Tertutup + 1 (Region Terluar)
Jumlah Region = 2 + 1
Jumlah Region = 3
4. Independent path
Path 1 : 1-2-3-4-5-6-7-11-12-13-14-15-16-17-18-19-20-21-24-25-26-27
Path 2 : 1-2-3-4-5-6-7-8-9-25-26-27

5. Jumlah *predicated node*
Predicated node = 7,16
V(G) = P + 1
= 2 + 1 = 3

6. Penomoran ulang *flowgraph*



Gambar IV. 9 Penomoran Ulang *Flowgraph*

7. Graph Matrix

Simpul	1	2	3	4	5	6	7	8	9	Jumlah
1		1								0
2			1	1						1
3										0
4					1					0
5						1				0
6							1			0
7					1			1		1
8									1	0
9										0
Jumlah (x)										2

V(g) = x + 1
V(g) = 2 + 1
V(g) = 3

8. Kesimpulan
   Berdasarkan pengujian pada setiap metode, dihasilkan nilai *Cyclometic Complexity* yang sama yaitu 3. Maka dapat disimpulkan bahwa pengujian *white box* pada algoritma minimax berjalan dengan baik, karena setiap pengujian menghasilkan nilai yang sama.

IV.2.3 Pengujian *Black box*

Pengujian *black box* adalah metode pengujian yang berfokus pada persyaratan perangkat lunak. Tujuan pengujian *black box* adalah menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang, menguji hasil keluaran tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut dan mengetahui kemampuan perangkat lunak dalam memenuhi kebutuhan pengguna. Kasus dan hasil pengujian dapat dilihat pada Tabel IV.2.

Tabel IV. 2 Hasil Pengujian

No.	Kelas Uji	Detail Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1	Menu Utama	Menekan tombol mulai permainan	Menampilkan form mulai permainan	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol petunjuk	Menampilkan form petunjuk	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol exit	Keluar dari *game*	[√] Berhasil [ ] Tidak Berhasil
2	Mulai Permainan	Menekan tombol mulai baru	Menampilkan form stage 1	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol pilih stage	Menampilkan form pilih stage	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol back	Menampilkan form menu utama	[√] Berhasil [ ] Tidak Berhasil
4	Pilih Stage	Menekan tombol stage 1	Menampilkan form stage 1	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol stage 2	Menampilkan form stage 2	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol stage 3	Menampilkan form stage 3	[√] Berhasil [ ] Tidak Berhasil
5	Stage	Menekan tombol end turn	Mengakhiri giliran pemain	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol quit	Menampilkan form menu utama	[√] Berhasil [ ] Tidak Berhasil
		Memilih piece pemain	Menampilkan wilayah gerak dan wilayah serang	[√] Berhasil [ ] Tidak Berhasil

IV.2.4 Pengujian Beta

Pengujian beta adalah pengujian pada lingkungan pengguna. Pengujian ini menggunakan metode pengumpulan data melalui kuisioner. Pengguna akan mencoba *game* ini, kemudian pengguna akan memberikan feedback pada kuesioner yang telah disediakan.

Untuk mengetahui tanggapan dan penilaian pengguna terhadap *game* ini. Telah disebarkan kuesioner kepada 30 responden dengan usia berdasarkan pada kebutuhan pengguna. Berikut adalah pertanyaan dan hasil kuesioner yang telah dibagikan dengan menggunakan rumus :

Y = P/Q * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal

Q = Jumlah responden

Y = Nilai prosentase jawaban

Hasil Pengujian Beta adalah sebagai berikut:

Apakah langkah-langkah yang diambil oleh lawan sulit untuk anda prediksi?

	Kategori Jawaban		
	Ya	Biasa	Tidak
Frekuensi Jawaban	22	7	2
Prosentase Nilai	70,9%	22,5%	6,4%

Apakah langkah yang diambil oleh musuh bersifat logis?

	Kategori Jawaban		
	Ya	Biasa	Tidak
Frekuensi Jawaban	18	17	0
Prosentase Nilai	58%	52%	0

Apakah lawan pada *game* ini sulit dikalahkan?

	Kategori Jawaban		
	Ya	Biasa	Tidak
Frekuensi Jawaban	28	2	1
Prosentase Nilai	90,3%	6,4%	3,2%

Apakah kontrol pada *game* ini mudah untuk dioperasikan?

	Kategori Jawaban		
	Ya	Biasa	Tidak
Frekuensi Jawaban	20	10	1
Prosentase Nilai	64,5%	32,2%	3,2%

Apakah informasi pada antarmuka sudah cukup jelas?

	Kategori Jawaban		
	Ya	Biasa	Tidak
Frekuensi Jawaban	21	8	2
Prosentase Nilai	67,7%	25,8%	6,4%

Kesimpulan dari pengujian beta ini adalah :

1. AI menjadi lebih sulit diprediksi langkahnya serta menjadi semakin sulit dikalahkan bagi sebagian pemain ketika algoritma minimax diaplikasikan pada AI.
2. Aturan-aturan yang ada pada *game* ini cukup mudah dimengerti oleh pengguna (pemain).
3. Informasi serta kontrol pada *game* ini ditampilkan dengan cukup baik