BAB IV
IMPLEMENTASI DAN PENGUJIAN

IV.1. Implementasi Sistem

Implementasi sistem adalah tahap pengujian kelayakan pada sistem yang dibuat sehingga aplikasi yang dibuat tidak menyimpang dari perancangan sistemnya. Game ini dibuat menggunakan tools pembangunan perangkat lunak Eclipse. Perangkat keras yang digunakan untuk mengimplementasikan dan menguji game ini memiliki spesifikasi sebagai berikut :

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

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

1. Windows 7 Ultimate sebagai sistem operasi
2. Eclipse Indigo sebagai perangkat lunak pembangun game

IV.1.1. Implementasi Antarmuka

Implementasi antarmuka adalah bagian yang menunjukan bentuk tampilan setiap antarmuka pada aplikasi yang sudah dibangun. Implementasi antarmuka pada game ini ditunjukan pada gambar berikut ini.

1. Antar muka Mulai Permainan

   Antarmuka mulai permainan ditunjukan pada gambar IV.1 tombol yang tersedia adalah tombol mulai permainan yang berfungsi sebagai akses menuju menu utama permainan.



Gambar IV. 1 Antarmuka mulai permainan

2. Antarmuka Menu Utama

   Antar muka menu utama ditunjukan pada gambar IV.2. tombol yang tersedia adalah tombol mulai baru sebagai akses menuju pilih karakter, tombol *load game* sebagai akses memulai permainan sesuai *save point*, tombol *credit* sebagai akses melihat tampilan *credit*, tombol keluar sebagai akses untuk keluar dari permainan.



Gambar IV. 2 Antarmuka menu utama

3. Antarmuka Pilih Karakter

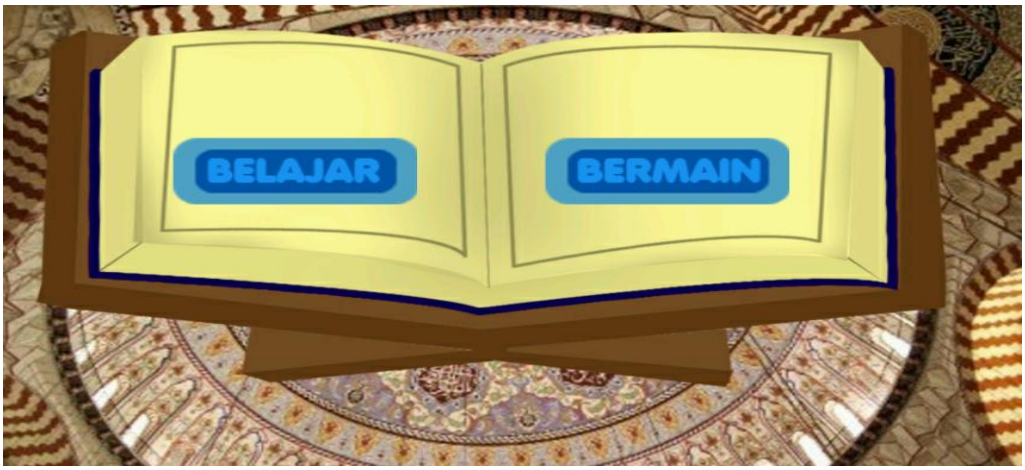
   Antarmuka pilih karakter adalah pengembangan dari form antarmuka menu utama, ditunjukan pada gambar IV.3. Antarmuka ini merupakan antarmuka untuk memilih karakter yang akan dimainkan.



Gambar IV. 3 Antarmuka pilih karakter

4. Antarmuka Menu Pilihan Belajar Dan Bermain

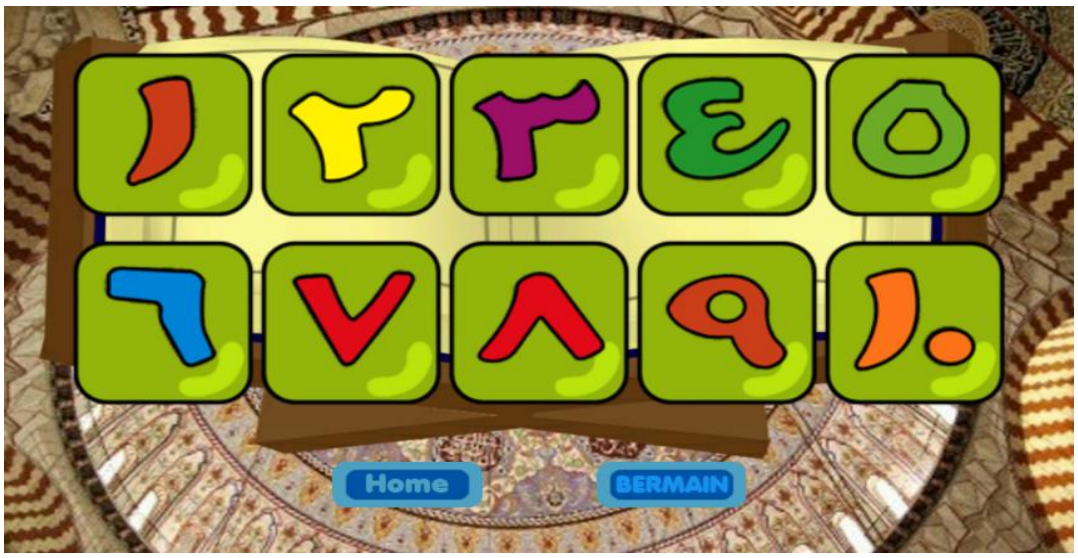
   Antarmuka pilihan belajar dan bermain ditunjukan pada gambar IV.4. tombol yang tersedia adalah tombol belajar sebagai akses menuju stage 1 atau stage 2 belajar, tombol bermain sebagai akses menuju stage 1 atau stage 2 bermain.



Gambar IV. 4 Antarmuka menu pilihan belajar dan bermain

5. Antarmuka Belajar

   Antarmuka belajar stage terdiri dari antarmuka belajar stage 1 yang ditunjukan pada gambar IV.5, dan antarmuka belajar stage 2 yang ditunjukan pada gambar IV.6. tombol yang tersedia adalah tombol home yang berfungsi sebagai akses untuk kembali ke menu utama, tombol bermain sebagai akses untuk bermain pada stage 1 atau bermain pada stage 2, tombol huruf dan angka sebagai akses untuk menuju antarmuka angka dan huruf secara detail.



Gambar IV. 5 Antarmuka Belajar Stage 1



Gambar IV. 6 Antarmuka Belajar Stage 2

6. Antarmuka Pilih Jalan

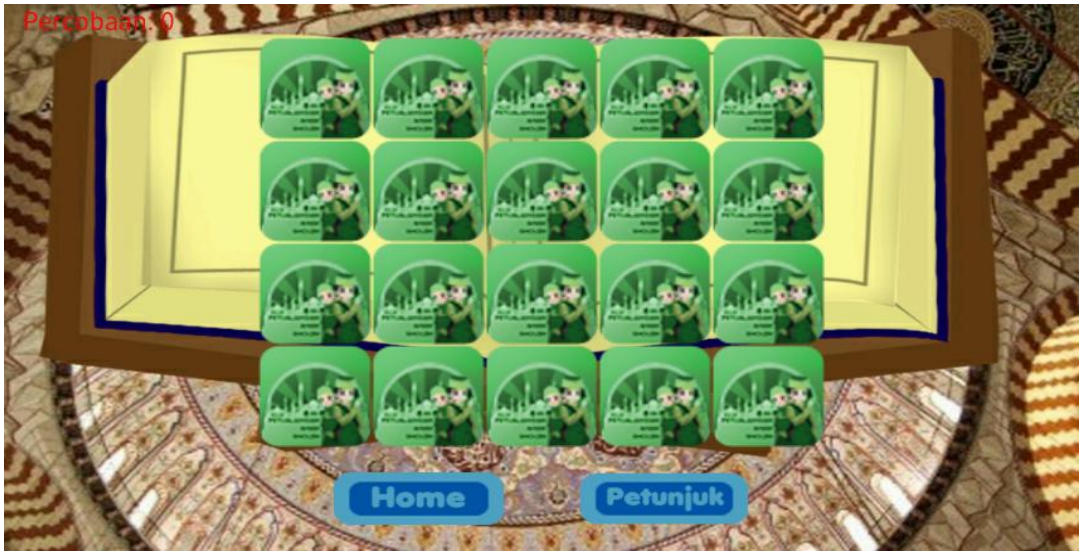
   Antarmuka pilih jalan ditunjukan pada gambar IV.7. antarmuka ini merupakan antarmuka untuk memilih jalan yang akan dilewati, ada dua tombol yaitu tombol jalan kiri sebagai akses cerita permainan pada arah kiri, dan tombol jalan kanan sebagai akses cerita permainan pada arah kanan.



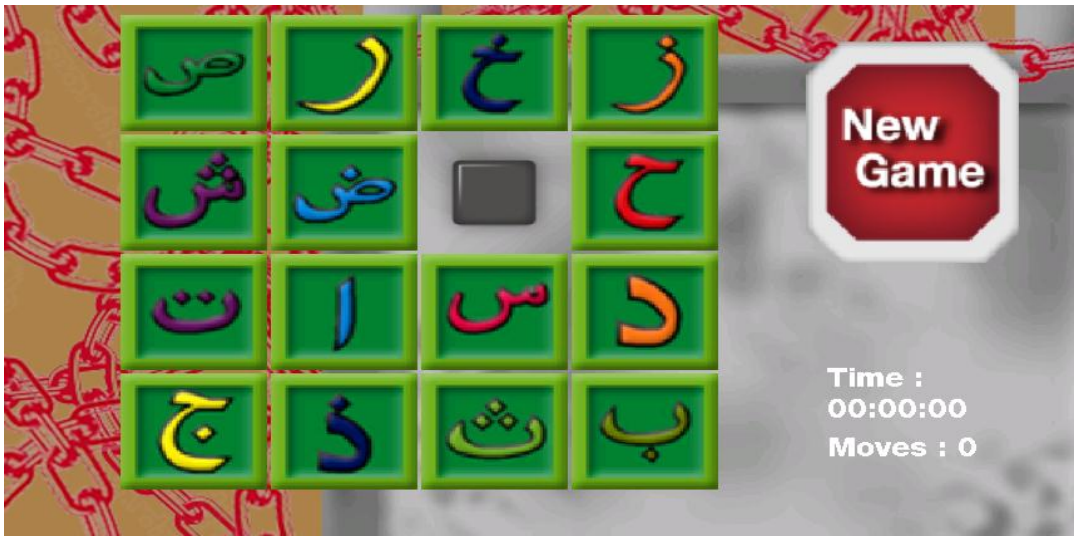
Gambar IV. 7 Antarmuka Pilih Jalan

7. Antarmuka Stage

   Antarmuka stage terdiri dari antarmuka stage 1 yang ditunjukan pada gambar IV.8, antarmuka stage 2 yang ditunjukan pada gambar IV.9, dan antarmuka stage 3 ditunjukan pada gambar IV.10. tombol yang tersedia adalah tombol home sebagai akses untuk kembali ke menu utama dan tombol petunjuk sebagai akses untuk menuju antarmuka petunjuk.



Gambar IV. 8 Antarmuka Stage 1



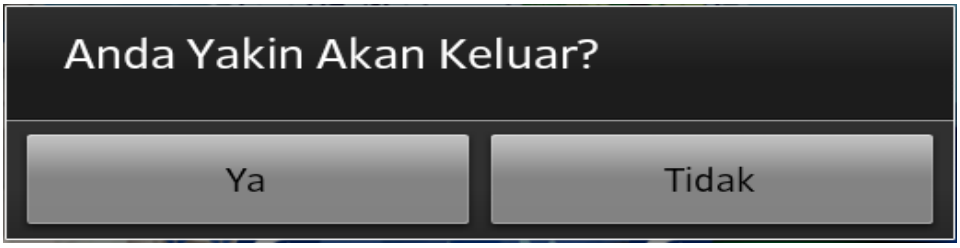
Gambar IV. 9 Antarmuka Stage 2

IV.1.2. Implementasi Pesan

Implementasi pesan adalah bagian yang menunjukan bentuk tampilan setiap pesan pada aplikasi yang sudah dibangun. Implementasi antarmuika pada game ini ditunjukan pada gambar IV.11 sampai dengan gambar IV.14.

1. Pesan keluar permainan

   Pesan keluar permainan ditunjukan pada gambar IV.11, pesan keluar permainan berfungsi sebagai validasi jika pemain ingin keluar dari permainan.



Gambar IV. 10 Pesan Keluar Permainan

2. Pesan *save game*

   Pesan *save game* ditunjukan pada gambar IV.12. pesan *save game* berfungsi sebagai pemberi informasi apakah user akan melakukan *save game* atau tidak.

3. Pesan kembali ke home

   Pesan kembali ke home ditunjukan pada gambar IV.13. pesan kembali ke home berfungsi sebagai validasi jika pemain ingin keluar dari permainan dan kembali ke menu utama.

IV.2. Pengujian Sistem

Pengujian sistem adalah bagian penelusuran dalam rangka menemukan kesalahan pada sistem yang dibangun. Tujuan pengujian sistem 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 tereksekusi dan kesalahan *typography* yang sulit ditemukan jika tidak dijalankan. Pada umumnya pengujian ini dilakukan hanya pada lingkungan 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 antarmuka, inisialisasi 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 table IV.1.

Tabel IV. 1 Deskripsi Rencana Pengujian

No.	Kelas Uji	Detail Pengujian	Jenis Pengujian
	Uji Algoritma	Pengujian pada proses penentuan nilai kebenaran pada game puzzle	*White Box*
	Uji antarmuka mulai permainan	Menekan tombol mulai permainan	*Black box*
	Uji antarmuka menu utama	Menekan tombol mulai baru	*Black box*
		Menekan tombol load game	*Black box*
		Menekan tombol credit	*Black box*
		Menekan tombol keluar	*Black box*
	Uji antarmuka pilih karakter	Menekan tombol karakter 1	*Black box*
		Menekan tombol karakter 2	*Black box*
	Uji antarmuka menu pilihan belajar dan bermain	Menekan tombol belajar	*Black box*
		Menekan tombol bermain	*Black box*
	Uji antarmuka belajar	Menekan tombol home	*Black box*
		Menekan tombol bermain	*Black box*

		Menekan tombol angka	*Black box*
		Menekan tombol huruf	*Black box*
	Uji antarmuka pilih jalan	Menekan tombol jalan kiri	*Black box*
		Menekan tombol jalan kanan	*Black box*
	Uji antarmuka stage	Menekan tombol home	*Black box*
		Menekan tombol petunjuk	*Black box*
		Menekan tombol gambar angka	*Black box*
		Menekan pilihan jawaban	*Black box*

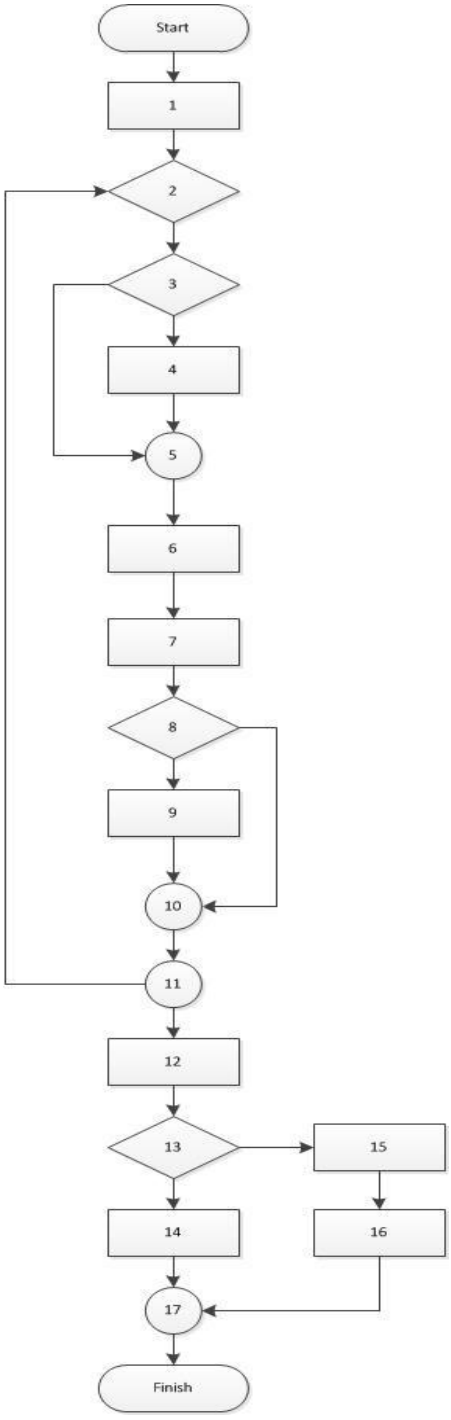
IV.2.2. Pengujian *White box*

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 yang digunakan dan dilakukan dengan beberapa tahap, diantaranya :

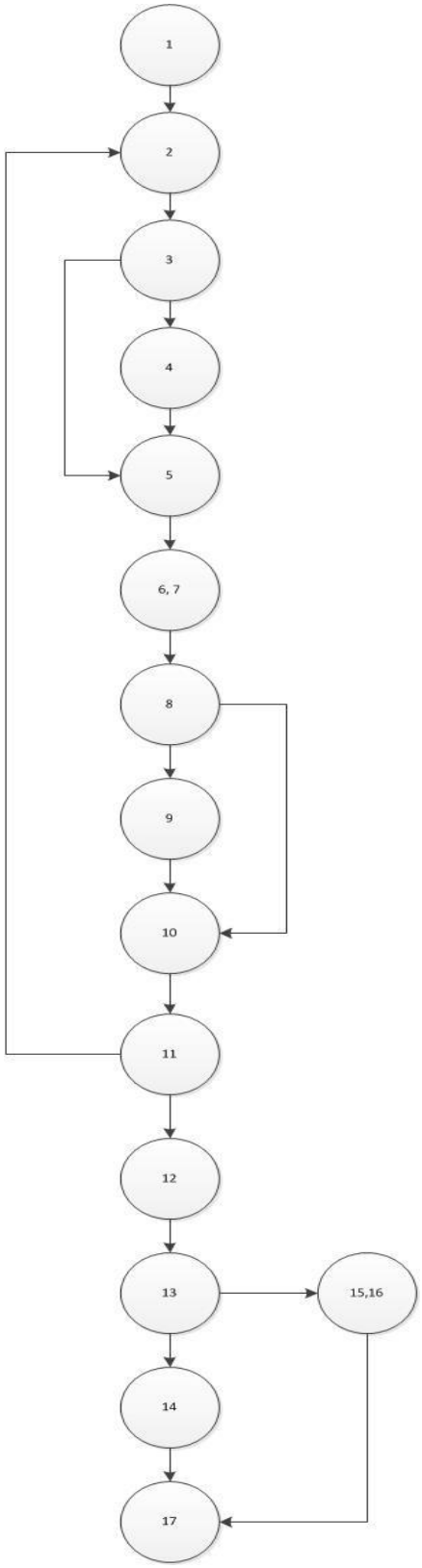
1. Pengubahan pseudocode menjadi *flowchart* kemudian menjadi *flowgraph.*

```
1   var n<-0;
2      for i<-1 to I <- 16 do
3            if (ContentPanel.Children[i] = StackPanel) AND false then
4                  continue;
5            endif
6                  num1<-FindItemValueByPosition(i)
7                  num2<-FindItemValueByPosition(I – 1)
8             if num1 > num2 then
9                   n<-n+1
10            endif
11     endfor
12              emptyPos <- FindEmptyItemPosition()
13           if n%2 = (emptyPos + emptyPos / 4) % 2 then
14                 return true
15              else
16                 return false
17           endif
```

Gambar IV. 11 Pseudocode algoritma



Gambar IV. 12 Flowchart Algoritma



Gambar IV. 13 Flowgraph Algoritma

2. *Cyclometic complexity*

   V(G) = E – N + 2
   = 17 – 14 + 2
   = 5

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

3. Jumlah *region*

   Jumlah Region = jmlah kurva tertutup + 1 (Region Terluar)
   Jumlah Region = 4 + 1
   Jumlah Region = 5

4. *Independent path*

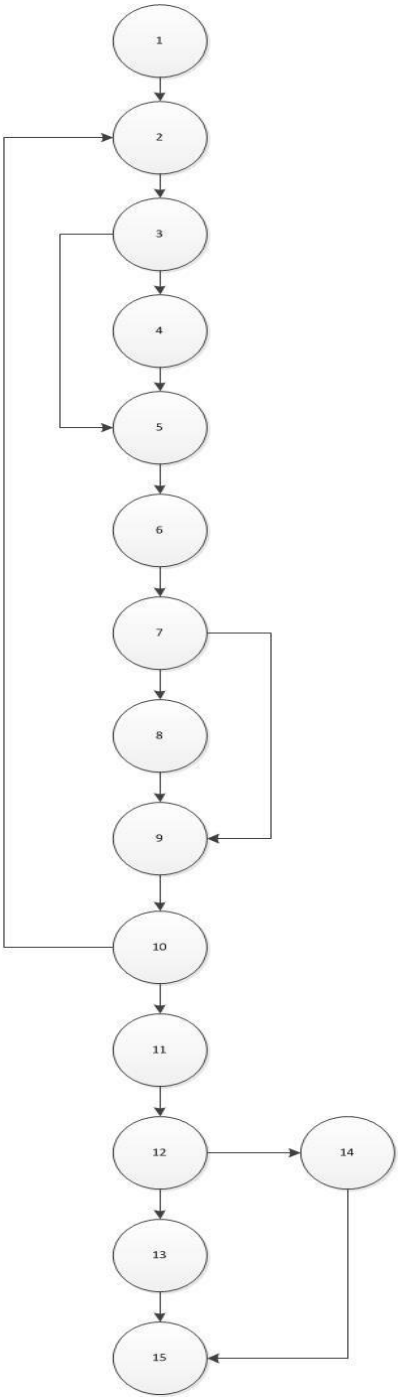
   Path 1 : 1-2-3-4-5-6-7-8-9-10-11-12-13-14-17
   Path 2 : 1-2-3-6-7-8-9-10-11-12-13-14-17
   Path 3 : 1-2-3-4-5-6-7-8-11-12-13-14-17
   Path 4 : 1-2-3-4-5-6-7-8-9-10-11-12-13-15-16-17

5. Jumlah *predicated node*

   Predicated node = 3,8,11,13

   V(G) = P + 1
   = 4 + 1
   = 5

6. Penomoran ulang *flowgraph*



Gambar IV. 14 Penomoran Ulang Flowgraph

7. Graph Matriks

Tabel IV. 2 Graph Matriks

simpul	1	2	3	4	5	6	7	8	9	10	11	12	13	14	Jumlah
1		1													0
2			1						1						1
3				1	1										1
4					1										0
5						1									0
6							1								0
7								1	1						1
8									1						0
9										1					0
10											1				0
11												1			0
12														1	1
13														1	0
14															0
Jumlah (x)															4

V(g) = x + 1
V(g) = 4 + 1
V(g) = 5

8. Kesimpulan

Berdasarkan pengujian pada setiap metode, dihasilkan nilai *Cyclometic Complexity* yang sama yaitu 5. 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 table IV.2.

Tabel IV. 3 Pengujian Blackbox

No.	Kelas Uji	Detail Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1.	Mulai Permainan	Menekan tombol mulai permainan	Menampilkan menu utama	[√] Berhasil [ ] Tidak Berhasil
2.	Menu Utama	Menekan tombol mulai baru	Menampilkan form pilih karakter	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol load game	Menampilkan form load game	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol credit	Menampilkan form credit	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol keluar	Keluar dari game	[√] Berhasil [ ] Tidak Berhasil
3	Pilih Karakter	Menekan tombol gambar karakter 1	Menampilkan form menu pilihan belajar dan bermain	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol gambar karakter 2	Menampilkan form menu pilihan belajar dan bermain	[√] Berhasil [ ] Tidak Berhasil
4	Menu Pilihan Belajar dan Bermain	Menekan tombol belajar	Menampilkan form belajar	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol bermain	Menampilkan form bermain	[√] Berhasil [ ] Tidak Berhasil
5	Belajar	Menekan tombol home	Menampilkan form menu utama	[√] Berhasil [ ] Tidak Berhasil

		Menekan tombol bermain	Menampilkan form bermain	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol gambar angka dan huruf	Menampilkan form detail gambar angka dan huruf	[√] Berhasil [ ] Tidak Berhasil
6	Pilih Jalan	Menekan tombol gambar jalan kiri	Menampilkan form cerita jalan kiri	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol gambar jalan kanan	Menampilkan form cerita jalan kanan	[√] Berhasil [ ] Tidak Berhasil
7	Stage	Menekan tombol home	Menampilkan form menu utama	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol petunjuk	Menampilkan form petunjuk	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol gambar angka	Gambar angka bisa terbuka dan tertutup	[√] Berhasil [ ] Tidak Berhasil
		Menekan pilihan jawaban	Pilihan jawaban bisa dipilih	[√] Berhasil [ ] Tidak Berhasil

IV.2.4. Pengujian *Beta*

Pengujian *beta* adalah pengujian pada lingkungan pengguna. Pengujian ini menggunakan metode pengumpulan data melalui kuesioner. 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. Teknik pengujian yang digunakan dalam pengujian beta ini adalah teknik *skala likert*.

1. Penafsiran Bobot Pengujian

   Pengujian menggunakan 3 kategori jawaban dengan bobot yang berbeda pada setiap jawabannya seperti pada table berikut.

Tabel III. 1 Bobot Jawaban

Pernyataan	Tidak	Cukup	Ya
Bobot	1	2	3

2. Keusioner Terhadap Anak Tingkat Sekolah Dasar

   Data hasil kuesioner yang telah diberikan dapat dicari presentasinya dari masing-masing jawaban dengan menggunakan rumus :

$$P = \frac{S}{skorideal} x100\%$$

Keterangan :

P = Nilai persentase yang dicari

S = Jumlah frekuensi dikalikan dengan nilai yang di tetapkan jawaban

Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel

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

Hasil Kuesioner:

1. Apakah dengan menggunakan aplikasi ini kamu bisa mengenal angka dan huruf hijaiyah ?

Tabel IV. 4 Perhitungan Persentase Pertanyaan Kesatu

Pertanyaan	Jawaban	Skor	Frekuensi	S
1	Tidak	1	0	0
	Cukup	2	8	16
	Ya	3	22	66
Jumlah				82

$$P = \frac{82}{90} x100 = 91\%$$

2. Apakah dengan aplikasi ini menambah minat kamu untuk belajar angka dan huruf hijaiyah ?

Tabel IV. 5 Perhitungan Persentase Pertanyaan kedua

Pertanyaan	Jawaban	Skor	Frekuensi	S
2	Tidak	1	0	0
	Cukup	2	16	32
	Ya	3	14	42
Jumlah				74

$$P = \frac{74}{90} x100 = 82\%$$

3. Apakah tampilan game ini menarik ?

Tabel IV. 6 Perhitungan Persentase Pertanyaan ketiga

Pertanyaan	Jawaban	Skor	Frekuensi	S
3	Tidak	1	0	0
	Cukup	2	8	16
	Ya	3	22	66
Jumlah				82

$$P = \frac{82}{90} x100 = 91\%$$

4. Apakah aplikasi game ini mudah untuk digunakan ?

Tabel IV. 7 Perhitungan Persentase Pertanyaan keempat

Pertanyaan	Jawaban	Skor	Frekuensi	S
4	Tidak	1	0	0
	Cukup	2	17	34
	Ya	3	13	39
Jumlah				73

$$P = \frac{73}{90} x100 = 81\%$$

Kesimpulan dari pengujian beta ini adalah

1. Sesuai dengan hasil kuesioner banyaknya anak-anak yang bisa mengenal angka dan huruf hijaiyah setelah memainkan game pesantren anak soleh ini.
2. Game petualangan anak soleh cukup menambah minat anak-anak untuk belajar angka dan huruf hijaiyah.
3. Dengan menggunakan tampilan yang menarik anak-anak menjadi tertarik untuk memainkan game pada perangkat mobile.