BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan. Setelah implementasi maka dilakukan pengujian sistem dimana akan dilihat kekurangan-kekurangan pada aplikasi untuk pengembangan sistem selanjutnya.

4.1 Implementasi Sistem

Setelah menganalisis dan merancang sistem yang akan dibangun secara rinci, maka selanjutnya menuju tahap implementasi.

Tujuan implementasi adalah untuk mengkonfirmasikan modul program perancangan kepada para pengguna sistem, sehingga pengguna dapat memberikan masukan terhadap pembangunan sistem.

4.1.1 Kebutuhan Perangkat Keras

Spesifikasi perangkat keras (*hardware)* yang digunakan dalam pembangunan *game* edukasi pintar struktur bahasa dapat dilihat pada tabel berikut :

Tabel 4.1 Perangkat Keras Yang Digunakan

No	Perangkat Keras	Keterangan
1	Processor	Intel Core 2 Duo T6600 @2.20GHz
2	Memory	2GB RAM
3	VGA card	256 MB
4	Monitor	14 ‘‘
5	Mouse dan Keyboard	1 buah

4.1.2 Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak (*software)* yang digunakan dalam pembangunan *game* edukasi pintar struktur bahasa dapat dilihat pada tabel berikut :

Tabel 4.2 Perangkat Lunak Yang Digunakan

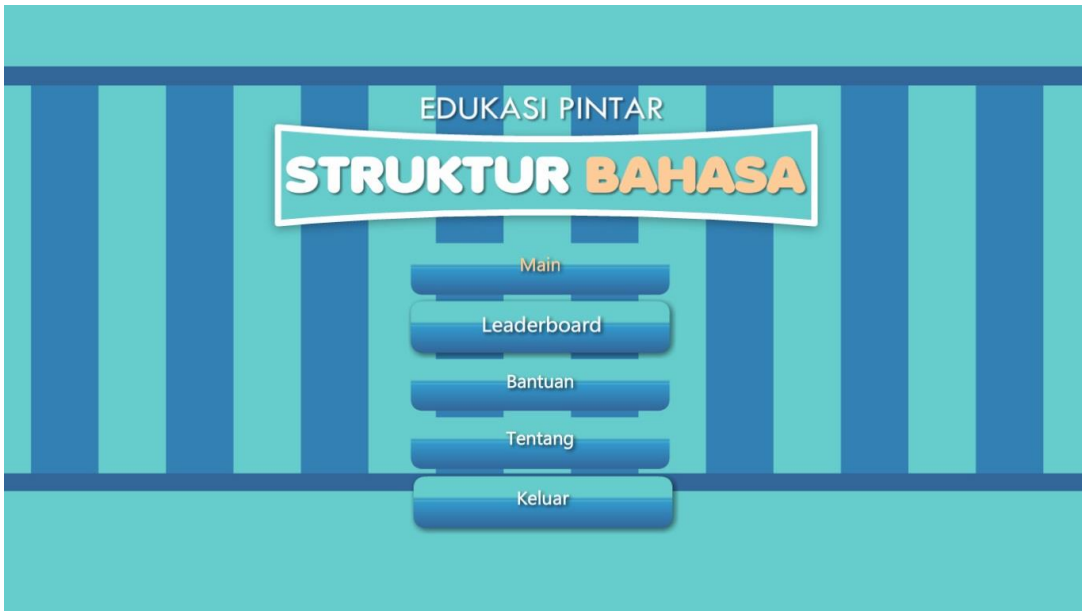
No	Perangkat Lunak	Keterangan
1	Windows 7	Sebagai sistem Operasi
2	Adobe Flash CS6	Sebagai program pembangun
3	Action Script	Sebagai bahasa pemrogranan
4	Adobe Photoshop	Untuk mengolah gambar
5	Adobe illustrator	Untuk mengolah gambar
6	Microsoft Word	Untuk membuat laporan
7	Star uml	Untuk membuat perancangan antarmuka
8	Adobe soundboth	Untuk mengolah suara

4.1.3 Implementasi Antar Muka

Implementasi antar muka adalah bagian yang menunjukan bentuk tampilan setiap antar muka pada aplikasi yang sudah dibangun. Implementasi antar muka pada *game* ini adalah sebagai berikut :

1. Antarmuka Menu Utama

   Antarmuka menu utama ditunjukan pada gambar 4.1. Tombol menu yang tersedia adalah tombol main yang berfungsi sebagai akses menuju permainan, tombol skor tertinggi yang berfungsi sebagai akses menuju nilai tertinggi, tombol bantuan yang berfungsi sebagai akses menuju bantuan permainan, tombol tentang yang berfungsi menampilkan profil pembuat aplikasi, dan tombol keluar yang berfungsi sebagai akses menuju keluar permainan.



Gambar 4.1 Antarmuka Menu Utama

2. Antarmuka main

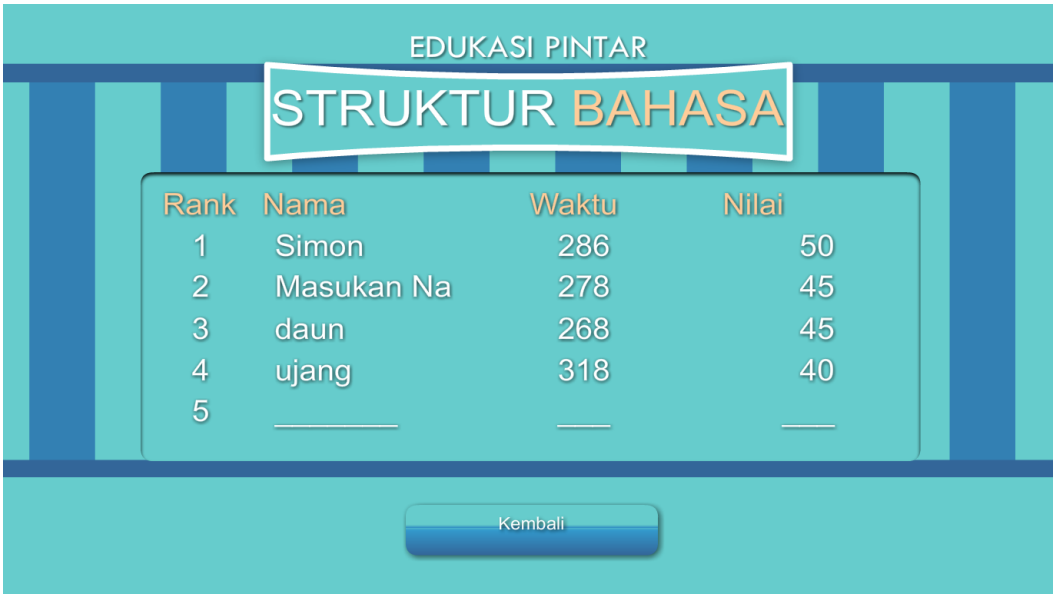
Antarmuka menu main ditunjukan pada gambar 4.2. antarmuka menu main sebagai akses menuju permainan menyusun kata dan mencocokkan kata. Berikut tampilan antarmuka halaman utama permainan menyusun kata dan mencocokkan kata.



Gambar 4.2. Antarmuka main

3. Antarmuka skor tertinggi

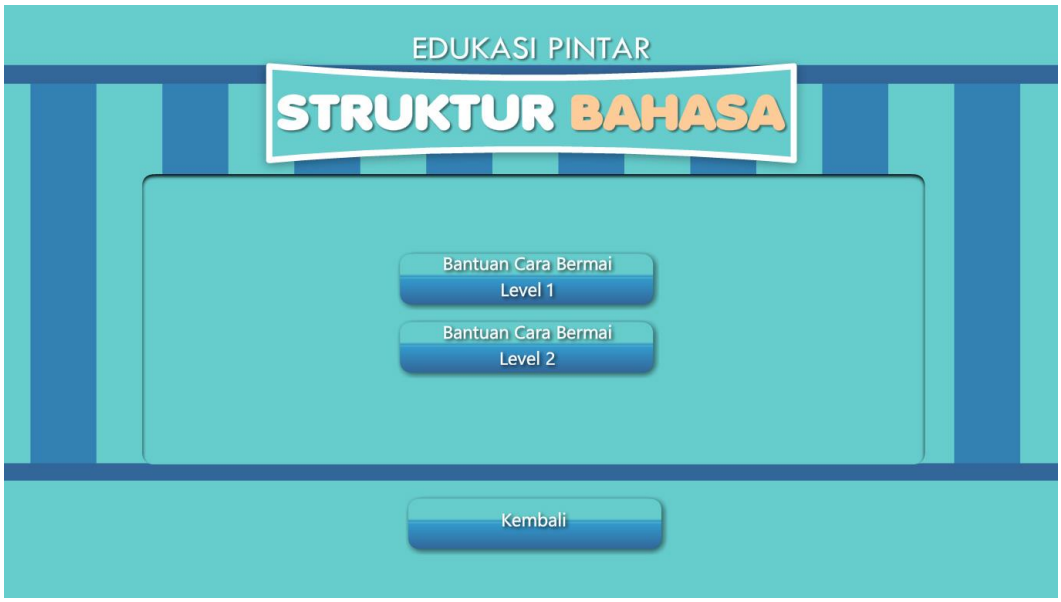
   Antarmuka skor tertinggi ditunjukan pada gambar 4.3. antarmuka menu skor tertinggi sebagai akses untuk menuju nilai tertinggi dan tombol kembali untuk kembali ke menu utama.



Gambar 4.3. Antarmuka skor tertinggi

4. Antarmuka bantuan

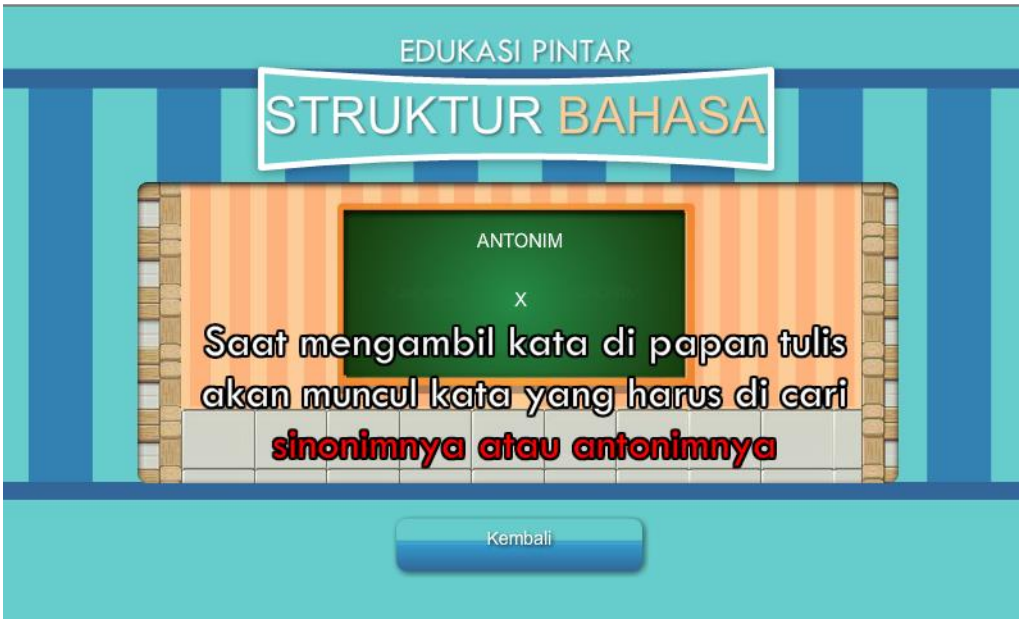
   Antarmuka bantuan ditunjukan pada gambar 4.4. 4.5. 4.6. antarmuka menu bantuan sebagai akses untuk menuju bantuan cara bermain dan tombol kembali untuk kembali ke menu utama.



Gambar 4.4. Antarmuka bantuan



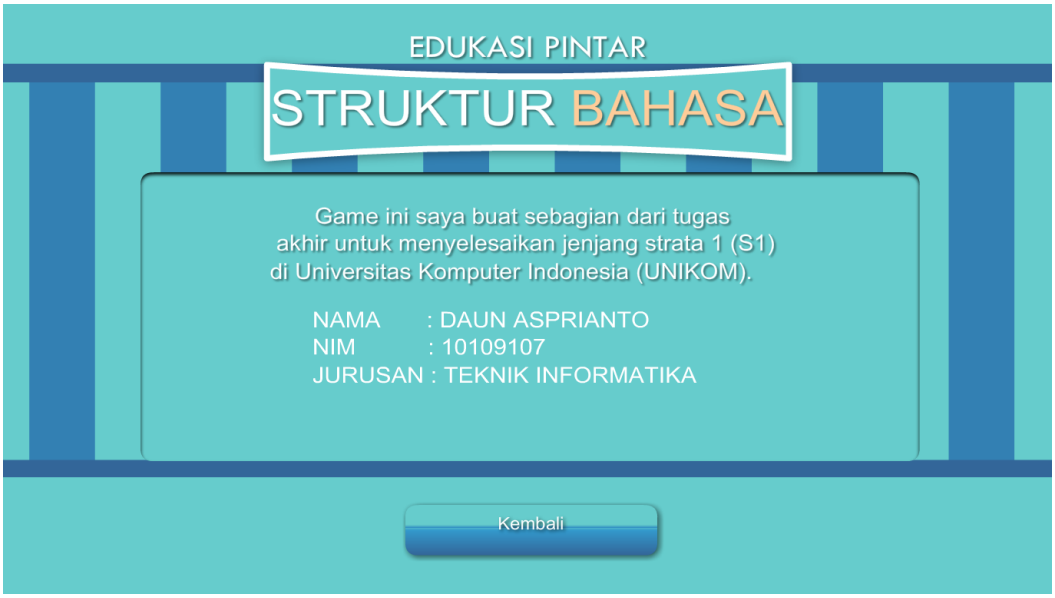
Gambar 4.5. Antarmuka bantuan *game* menyusun kata



Gambar 4.6. Antarmuka bantuan *game* mencocokkan kata

5. Antarmuka tentang

   Antarmuka tentang ditunjukan pada gambar 4.7. antarmuka menu tentang sebagai akses untuk menuju tentang pembuat aplikasi dan tombol kembali untuk kembali ke menu utama



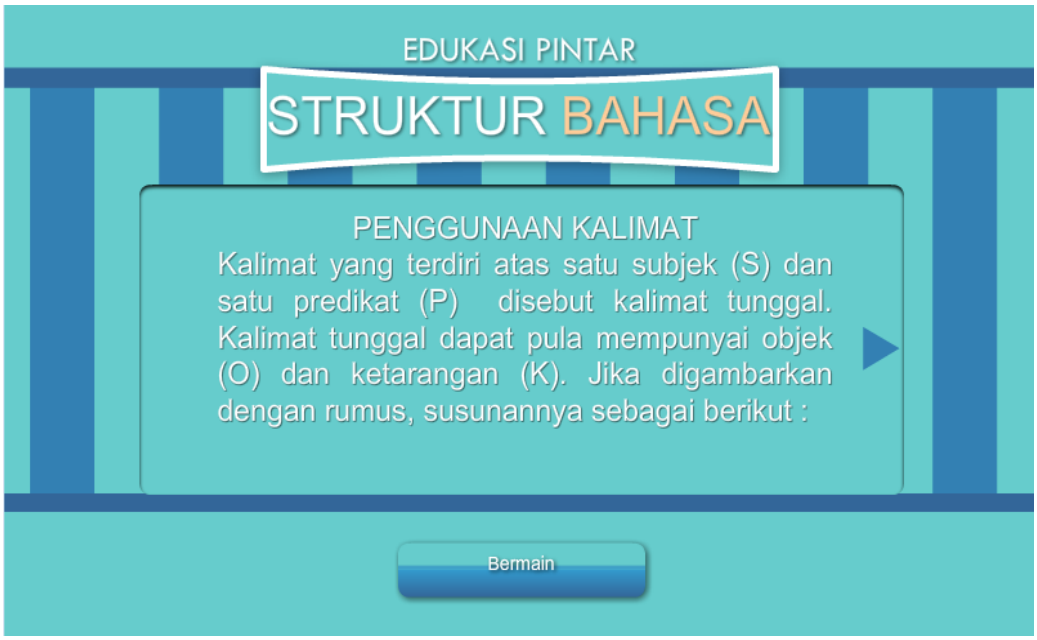
Gambar 4.7. Antarmuka tentang

6. Antarmuka *game* menyusun kata

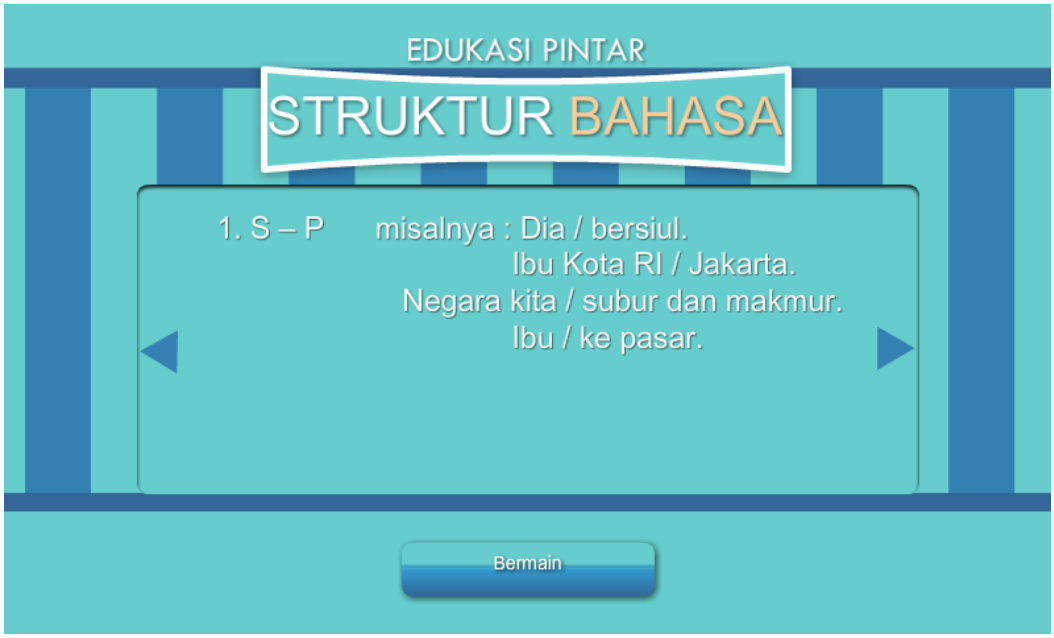
   User atau pemain memilih menu main yang berada dimenu utama, kemudian masuk ke halaman utama permainan dan pemain memilih gedung perpustakaan untuk memulai permainan menyusun kata.



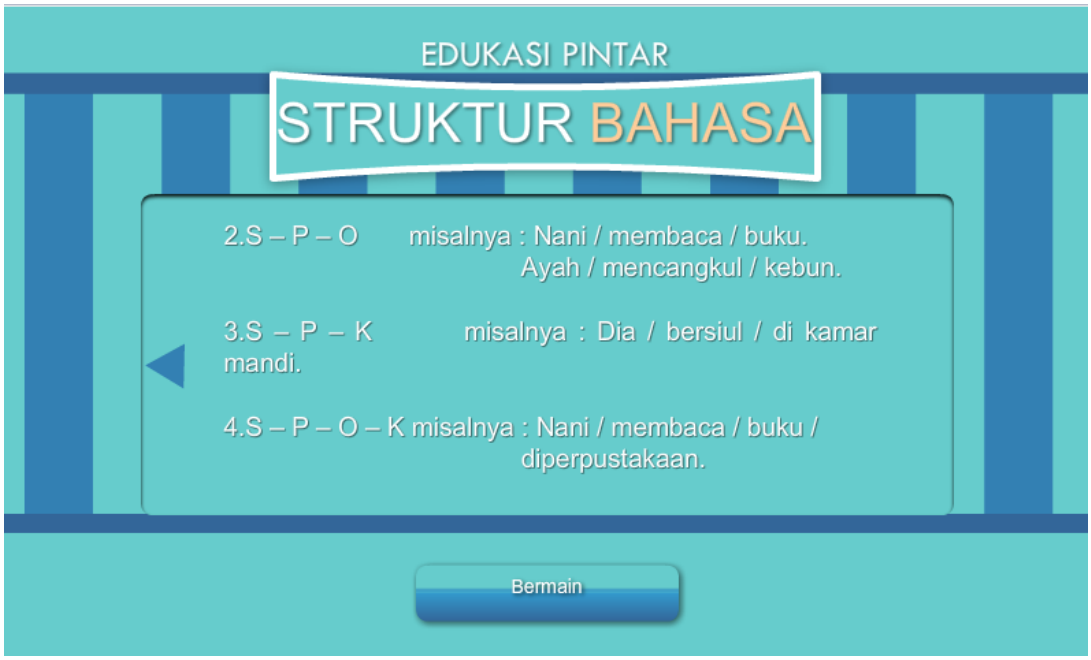
Gambar 4.8. Antarmuka *game* menyusun kata



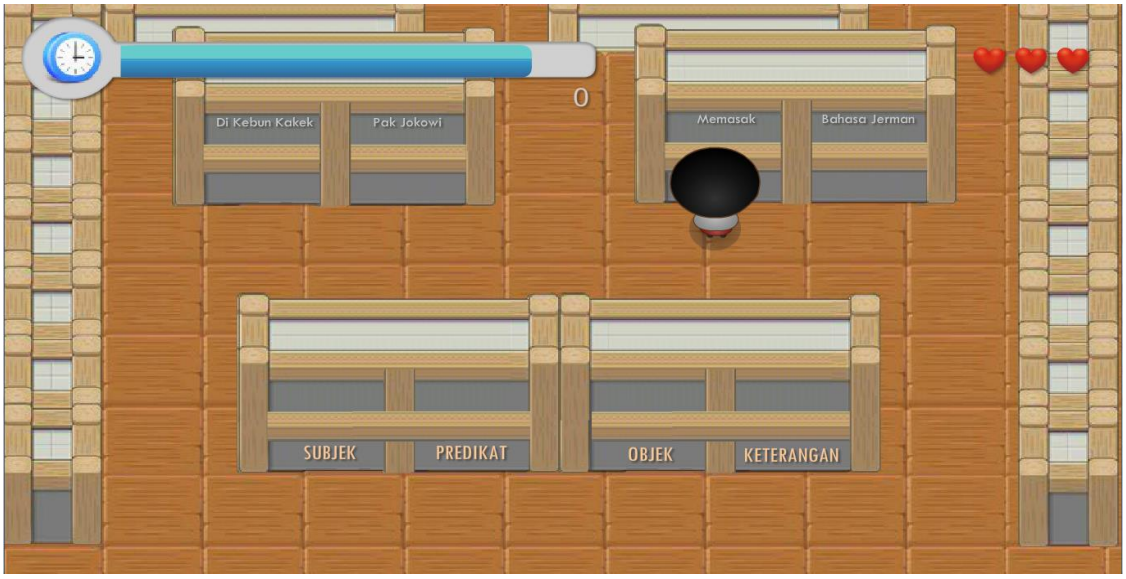
Gambar 4.9. Antarmuka materi *game* menyusun kata



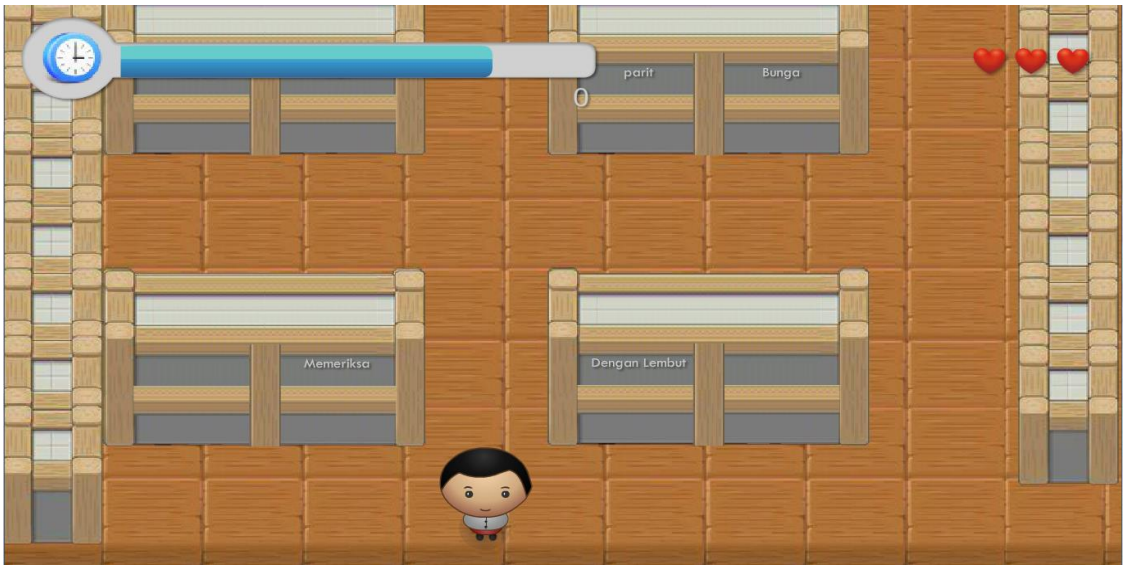
Gambar 4.10. Antarmuka materi *game* menyusun kata



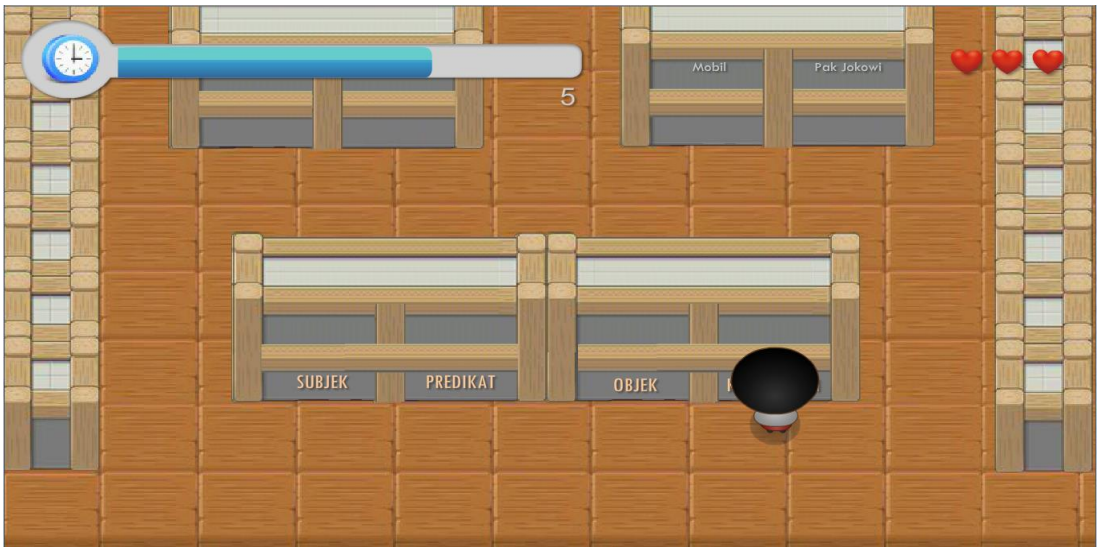
Gambar 4.11. Antarmuka materi *game* menyusun kata



Gambar 4.12. Antarmuka *game* menyusun kata



Gambar 4.13. Antarmuka *game* menyusun kata



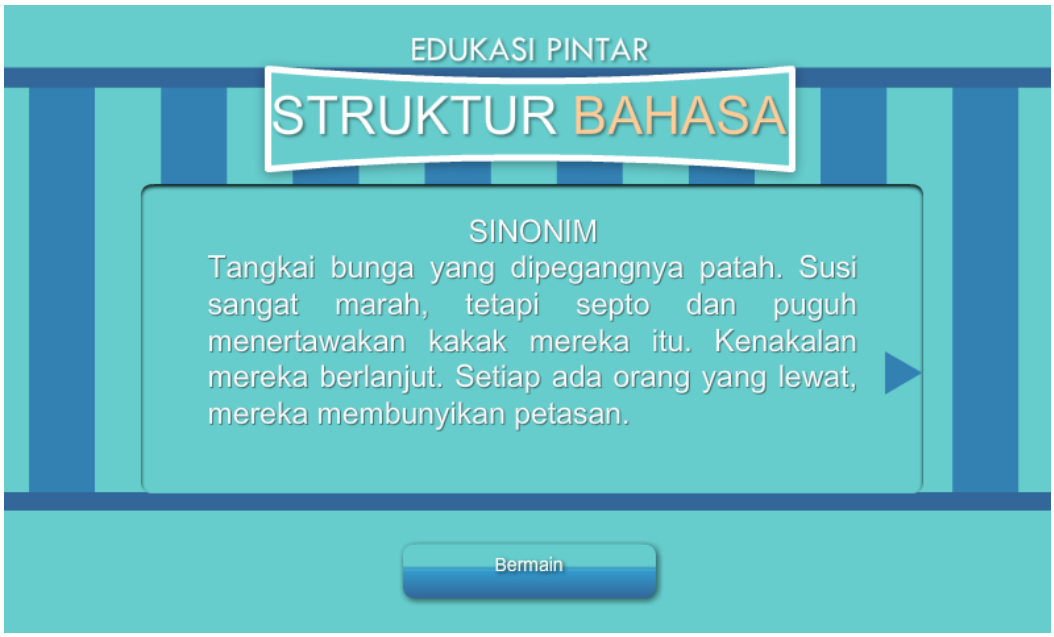
Gambar 4.14. Antarmuka *game* menyusun kata

7. Antarmuka *game* mencocokkan kata

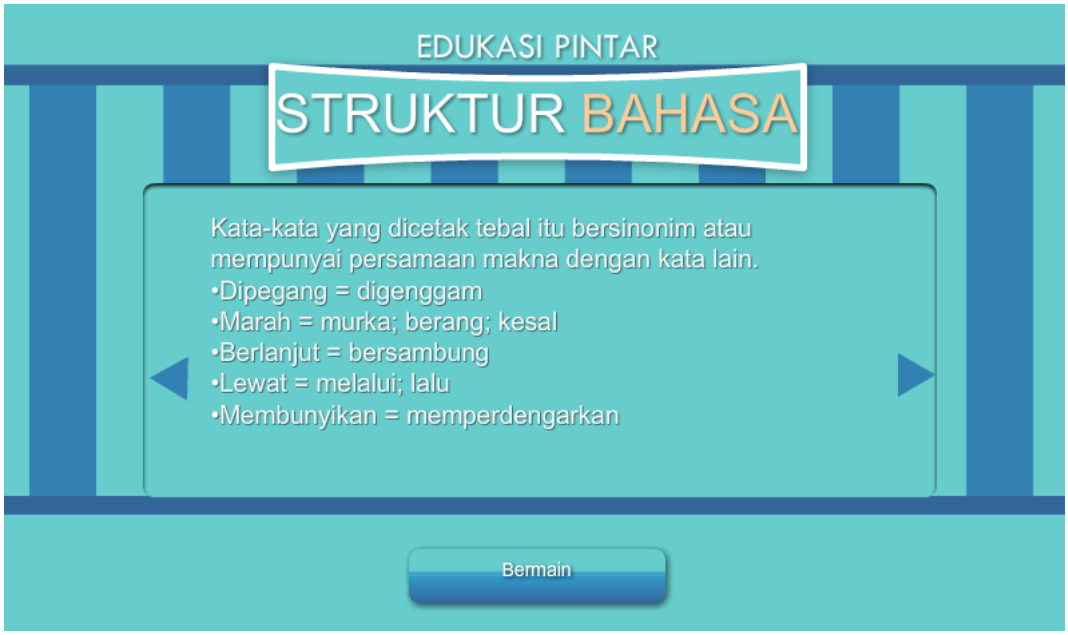
   User atau pemain memilih menu main yang berada dimenu utama, kemudian masuk ke halaman utama permainan dan pemain memilih gedung kelas bahasa untuk memulai permainan mencocokkan kata.



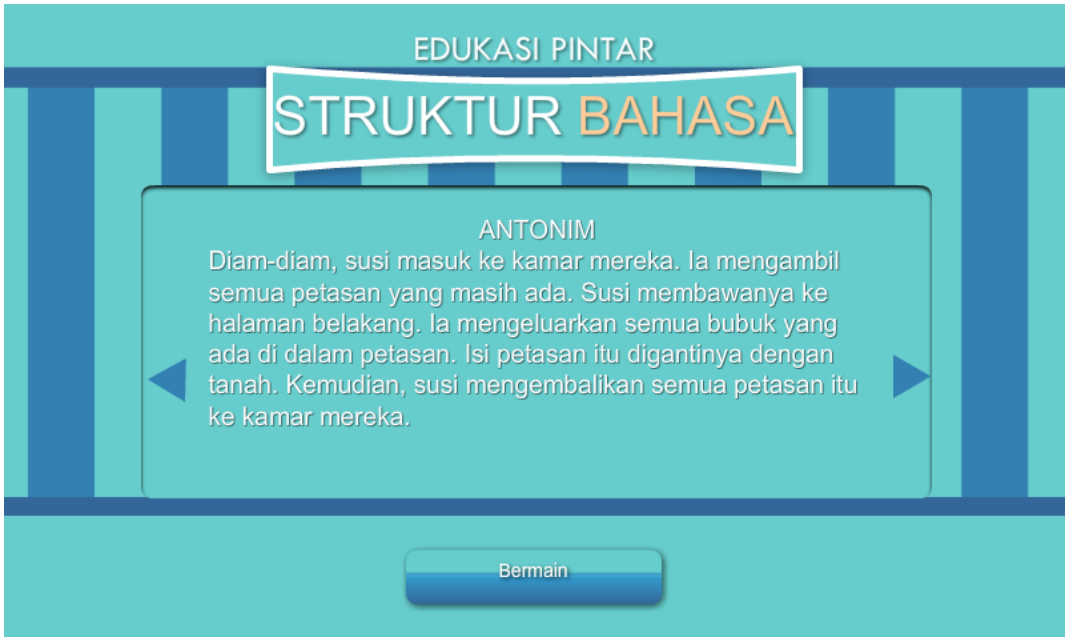
Gambar 4.15. Antarmuka *game* mencocokkan kata



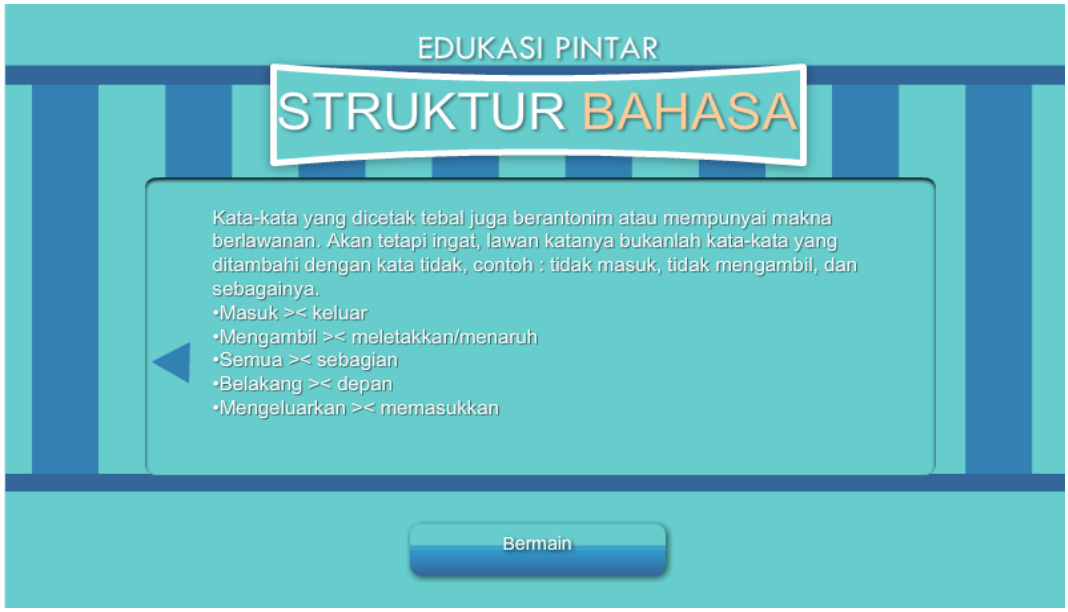
Gambar 4.16. Antarmuka materi *game* mencocokkan kata



Gambar 4.17. Antarmuka materi *game* mencocokkan kata



Gambar 4.18. Antarmuka materi *game* mencocokkan kata



Gambar 4.19. Antarmuka materi *game* mencocokkan kata



Gambar 4.20. Antarmuka *game* mencocokkan kata



Gambar 4.21. Antarmuka *game* mencocokkan kata

4.1.4 Implementasi Pesan

Implementasi pesan merupakan bagian tampilan pada *game* yang membutuhkan penggunanya untuk melakukan suatu keputusan. Berikut ini adalah implementasi pesan pada *game* Pintar Struktur Bahasa.

Tabel **Error! No text of specified style in document.**.3 Implementasi Pesan *Game*

No	Pesan	Keterangan
1	Pesan Keluar Game	Merupakan sebuah pesan yang memberikan pengguna suatu bentuk pilihan untuk melakukan keputusan apakah YA untuk keluar dari permainan atau TIDAK untuk tetap pada permainan.

4.2 Pengujian Sistem

Dalam melakukan pengujian terhadap aplikasi *game* yang dibangun dilakukan dengan dua macam teknik, yaitu teknik pengujian *black box* dan teknik pengujian *white box*. Pengujian *black box* berfokus pada menemukan kesalahan program, sedangkan pengujian *white box* berfokus kepada struktur internal (*source code*) program untuk mengetahui apakah masih terjadi *error*. Pengujian dilakukan dengan dua tahap, yaitu tahap pengujian *alpha* (fungsional) dan tahap pengujian *beta*.

4.2.1 Rencana Pengujian

Pengujian akan dilkakukan melalui dua tahap, yaitu pengujian *alpha* dan pengujian *beta.*

1. Pengujian *Alpha*

   Metode yang digunakan dalam pengujian *alpha* adalah metode *blackbox* yang fokus pada persyaratan fungsional dari perangkat lunak yang dibangun.

2. Pengujian *Beta*

   Metode yang digunakan dalam pengujian *beta* merupakan pengujian yang dilakukan secara objektif terhadap pengguna.

4.2.2 Pengujian *Alpha*

Pada tahap ini, pengujian dilakukan untuk memastikan aplikasi *game* yang dibangun telah berjalan dengan benar dan sesuai dengan kebutuhan yang diinginkan. Pengujian *alpha* berfokus pada persyaratan fungsional perangkat lunak.

4.2.2.1 Skenario Pengujian

Skenario pengujian menjelaskan pengujian terhadap sistem yang ada pada aplikasi *game* yang dibangun. Skenario pengujian dari aplikasi *game* yang dibangun dapat dilihat pada tabel berikut.

Tabel 4.4. Skenario Pengujian *game* edukasi pintar struktur bahasa

No	Item uji	Detail Pengujian	Jenis Pengujian
1	Menu utama	Menampilkan menu main	*Black box*
		Menampilkan menu skor tertinggi	*Black box*
		Menampilkan menu bantuan	*Black box*
		Menampilkan menu tentang	*Black box*
2	Menu main	Menampilkan halaman utama permainan	*Black box*
		Memilih gedung perpustakaan dan gedung kelas bahasa	*Black box*
	Permainan	Memilih gedung perpustakaan	*Black box*
		Menampilkan materi	*Black box*
		Menekan tombol bermain	*Black box*

3	menyusun kata	Menampilkan permainan menyusun kata	*Black box*
4	Permainan mencocokkan kata	Memilih gedung kelas bahasa	*Black box*
		Menampilkan materi	*Black box*
		Menekan tombol bermain	*Black box*
		Menampilkan permainan mencocokkan kata	*Black box*
6	Menu skor tertinggi	Menampilkan nilai tertinggi	*Black box*
		Menekan tombol kembali	*Black box*
7	Menu bantuan	Menampilkan pilihan bantuan	*Black box*
		Menekan tombol bantuan permainan 1	*Black box*
		Menekan tombol bantuan permainan 2	*Black box*
		Menekan tombol kembali	*Black box*
8	Menu tentang	Menampilkan tentang pembuat aplikasi	*Black box*
		Menekan tombol kembali	*Black box*

4.2.2.2 Hasil Pengujian (*Black box)*

Berikut ini adalah kasus untuk menguji perangkat lunak yang telah dibangun dengan menggunakan metode *Black box.*

Tabel 4.5. Hasil Pengujian *Black box*

No	**Kasus / Uji**	**Skenario Uji**	**Hasil yang diHarapkan**	**Hasil Pengujian**
1	Menu Utama	Memilih menu main	Masuk ke halaman utama permainan	[√] Berhasil [ ] Tidak Berhasil
		Memilih menu skor tertinggi	Masuk dan menampilkan nilai tertinggi	[√] Berhasil [ ] Tidak Berhasil
		Memilih menu bantuan	Masuk dan menampilkan pilihan bantuan permainan	[√] Berhasil [ ] Tidak Berhasil
		Memilih menu tentang	Masuk dan menampilkan tentang pembuat aplikasi	[√] Berhasil [ ] Tidak Berhasil
2	Menu Main	Memilih menu main	Masuk ke halaman utama permainan	[√] Berhasil [ ] Tidak Berhasil
3	Permainan menyusun kata	Memilih gedung perpustakaan	Masuk ke halaman materi	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol	Masuk ke permainan	[√] Berhasil [ ] Tidak

		bermain	menyusun kata	Berhasil
4	Permainan mencocokkan kata	Memilih gedung kelas bahasa	Masuk ke halaman materi	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol bermain	Masuk ke permainan mencocokkan kata	[√] Berhasil [ ] Tidak Berhasil
5	Menu skor tertinggi	Memilih menu skor tertinggi	Masuk dan menampilkan nilai tertinggi	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol kembali	Kembali ke menu utama	[√] Berhasil [ ] Tidak Berhasil
6	Menu bantuan	Memilih menu bantuan	Masuk dan menampilkan pilihan menu bantuan	[√] Berhasil [ ] Tidak Berhasil
		Memilih tombol bantuan permainan 1	Masuk dan menampilkan bantuan permainan 1	[√] Berhasil [ ] Tidak Berhasil
		Memilih tombol bantuan permainan 2	Masuk dan menampilkan bantuan permainan 2	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol kembali	Kembali ke menu utama	[√] Berhasil [ ] Tidak Berhasil

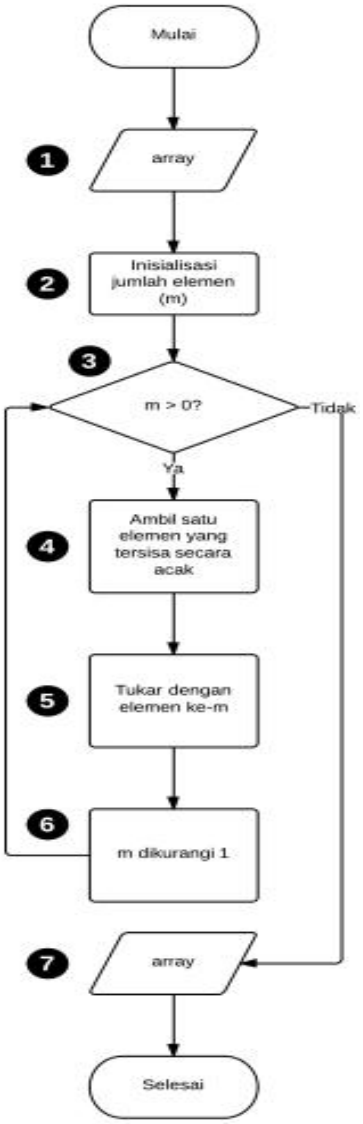
7	Menu tentang	Memilih menu tentang	Masuk dan menampilkan tentang pembuat aplikasi	[√] Berhasil [ ] Tidak Berhasil
		Menekan tombol kembali	Kembali ke menu utama	[√] Berhasil [ ] Tidak Berhasil

4.2.2.3 Hasil Pengujian (*White box)*

Pengujian *white box* merupakan pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur *control* dari desain aplikasi secara prosedural untuk membagi pengujian ke dalam beberapa kasus pengujian untuk mengetahui apakah setiap node pada *independent paths* dikerjakan. Serta untuk menguji alur logika, perulangan, dan validasi parameter masukannya.

1. *Flow Chart Fisher – Yates Shuffle*

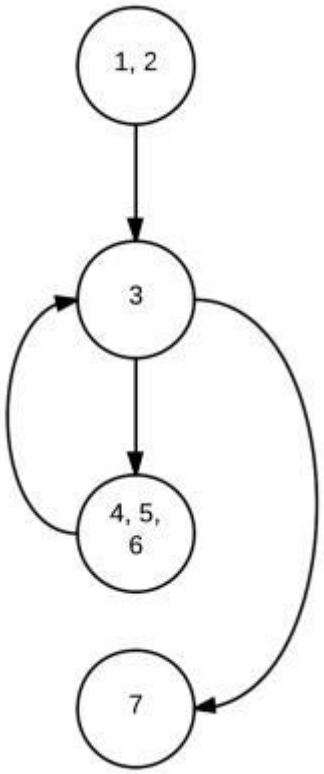
   Pertama – tama *Flow Chart* dari algoritma yang akan diuji, diberi nomor sesuai urutannya. Untuk lebih jelasnya lihat gambar 4.22.



Gambar 4.22. *Flow Chart Fisher-Yates Shuffle*

2. *Flow* Graph

Kemudian dari *Flow Chart* tadi dibuat *Flow Graph* seperti yang bisa dilihat pada gambar 4.23.



Gambar 4.23. *Flow Graph Fisher-Yates Shuffle*

3. *Cyclomatic* Complexity

   E = 4, N = 4

   V(G) = E – N + 2

   V(G) = 4 – 4 + 2

   V(G) = 2

   Keterangan :

   E : Jumlah busur pada flowgraph

   N : Jumlah simpul pada flowgraph

   V(G) : Cyclomatic Complexity

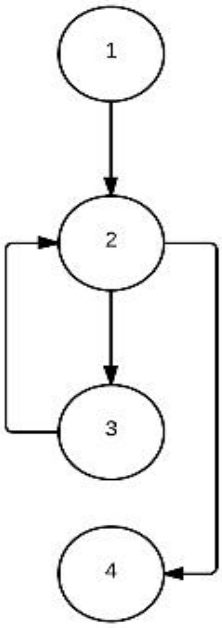
4. *Independent* Path

   Path 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7

   Path 2 : 1 – 2 – 3 – 7

5. Graph Matrices Testing

Sebelum dibuat tabel *graph matrices testing*, *flowgraph* sebelumnya disederhanakan dahulu dengan penomoran ulang seperti pada gambar 4.24.



Gambar 4.24. *Flow Graph* setelah penomoran ulang

Setelah *flowgraph* diberi nomor baru, lalu dibuat tabel *graph matrice* seperti pada tabel 4.6 berikut :

Tabel 4.6 Tabel *Graph Matrices Testing*

	1	2	3	4	Perhitungan
1		1			0
2			1	1	1
3		1			0
4					0
Jumlah					1

V(G) = Jumlah *graph matrices* + 1

V(G) = 2

6. Setelah dilakukan pengujian hasil yang didapat bisa dilihat pada tabel 4.7 berikut:

Tabel 4.7 Tabel Hasil Pengujian *White Box*

Path no	Node (n)	Hasil yang diharapkan	Hasil sesuai uji kasus	keterangan
1	1 – 2 – 3 – 4 – 5 – 6 – 7	Menghasilkan *array* yang sudah di-*shuffle*	Menghasilkan *array* yang sudah di-*shuffle*	[√] alur terlewati [ ] alur tidak terlewati
2	1 – 2 – 3 – 7	*Array* tidak di-*shuffle* karena array kosong	*Array* tidak di-*shuffle* karena array kosong	[√] alur terlewati [ ] alur tidak terlewati

4.2.2.4 Kesimpulan Hasil Pengujian *Black box dan White box*

Hasil pengujian *black-box* dengan pengujian diatas dapat ditarik kesimpulan bahwa pembangunan *game* edukasi pintar struktur bahasa tidak terdapat kesalahan proses dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

Dari hasil pengamatan dan pengujian *white-box* dan *black-box.* Pengujian *white-box* dapat disimpulkan bahwa pengacakan kata dalam *game* edukasi pintar struktur bahasa sudah sesuai dengan yang diharapkan.

4.2.3 Pengujian *Beta*

Pengujian *beta* merupakan pengujian yang dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas sistem, apakah telah memenuhi harapan atau belum. Untuk itu dalam pengujian *beta* dilakukan penelitian terhadap 30 responden atau calon pengguna sistem dengan melakukan pengumpulan data menggunakan kuesioner. 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 untuk setiap jawabannya seperti pada tabel berikut.

Tabel 4.8 Bobot nilai kuesioner

Kategori Jawaban	**Bobot**
Setuju	2
Biasa Saja	1
Tidak Setuju	0

2. Kualifikasi Skor

   Persentase skor = (jumlah skor / jumlah bobot nilai tertinggi) x 100%

   Berikut adalah kriteria penilaian intrepretasi skor, kualifikasi persentase = (bobot jawaban / bobot nilai maks) x 100% :

   a. Persentase nilai Setuju = (3/3) x 100% = 100%
   b. Persentase nilai Biasa Saja = (2/3) x 100% = 66.66%
   c. Persentase nilai Tidak Setuju = (1/3) x 100% = 33.33%



Gambar 4.25. Kualifikasi Skala Pengukuran

Berikut ini hasil perhitungan persentasi dari jawaban yang dipilih oleh 30 responden terpilih untuk dijadikan sampel penelitian *game* edukasi pintar struktur bahasa.

1. Pertanyaan Nomor 1

(1) Menurut adik – adik, apakah tampilan *game* ini menarik?

Tabel 4.9 Perhitungan Pertanyaan Nomor 1

Pertanyaan	Kategori Jawaban	Bobot	Frekuensi	Skor
1	Setuju	2	28	56
	Biasa Saja	1	2	2
	Tidak Setuju	0	-	-
Jumlah				58

Persentase = (58 / 60) x 100% = 97%



Gambar 4.26. Hasil Pengukuran Pertanyaan nomor 1

Berdasarkan persentase di atas maka dapat disimpulkan bahwa penilaian terhadap tampilan *game* edukasi pintar struktur bahasa adalah sebagian besar responden setuju.

2. Pertanyaan Nomor 2

(2) Menurut adik – adik, apakah *game* ini mudah digunakan?

Tabel 4.10 Perhitungan Pertanyaan Nomor 2

Pertanyaan	Kategori Jawaban	Bobot	Frekuensi	Skor
2	Setuju	2	26	52
	Biasa Saja	1	4	4
	Tidak Setuju	0	-	-
Jumlah				56

Persentase = (56 / 60) x 100% = 93%



Gambar 4.27. Hasil Pengukuran Pertanyaan nomor 2

Berdasarkan persentase di atas maka dapat disimpulkan bahwa penilaian anak mudah menggunakan *game* edukasi pintar struktur bahasa adalah sebagian besar responden setuju.

3. Pertanyaan Nomor 3

(3) Apakah adik – adik tertarik untuk memainkan *game* ini lagi?

Tabel 4.11 Perhitungan Pertanyaan Nomor 3

Pertanyaan	Kategori Jawaban	Bobot	Frekuensi	Skor
3	Setuju	2	25	50
	Biasa Saja	1	5	5
	Tidak Setuju	0	-	-
Jumlah				55

Persentase = (55 / 60) x 100% = 92%



Gambar 4.28. Hasil Pengukuran Pertanyaan nomor 3

Berdasarkan persentase di atas maka dapat disimpulkan bahwa penilaian ketertarikan anak memainkan *game* edukasi pintar struktur bahasa lagi adalah sebagian besar responden setuju.

4. Pertanyaan Nomor 4

(4) Setelah memainkan *game* ini, apakah adik – adik mudah mempelajari kosa kata kalimat?

Tabel 4.12 Perhitungan Pertanyaan Nomor 4

Pertanyaan	Kategori Jawaban	Bobot	Frekuensi	Skor
4	Setuju	2	25	50
	Biasa Saja	1	5	5
	Tidak Setuju	0	-	-
Jumlah				55

Persentase = (55 / 60) x 100% = 92%



Gambar 4.29. Hasil Pengukuran Pertanyaan nomor 4

Berdasarkan persentase di atas maka dapat disimpulkan bahwa penilaian anak setelah memainkan *game* edukasi pintar struktur bahasa anak dapat mudah mempelajari tentang kosa kata kalimat adalah sebagian besar responden setuju.

5. Pertanyaan Nomor 5

(5) Setelah memainkan *game* ini, apakah adik – adik mudah membedakan antonim dan sinonim kata?

Tabel 4.13 Perhitungan Pertanyaan Nomor 5

Pertanyaan	Kategori Jawaban	Bobot	Frekuensi	Skor
5	Setuju	2	27	54
	Biasa Saja	1	3	3
	Tidak Setuju	0	-	-
Jumlah				57

Persentase = (55 / 60) x 100% = 95%



Gambar 4.30. Hasil Pengukuran Pertanyaan nomor 5

Berdasarkan persentase di atas maka dapat disimpulkan bahwa penilaian anak setelah memainkan *game* edukasi pintar struktur bahasa anak dapat mudah mempelajari dan membedakan tentang antonim dan sinonim kata adalah sebagian besar responden setuju.

4.2.3.1 Kesimpulan Pengujian *Beta*

Berdasarkan pengujian *beta* yang telah didapat dari hasil kuesioner yang telah dijawab oleh para responden, maka bisa diambil kesimpulan bahwa *game* "Pintar Struktur Bahasa" ini :

1. Tampilan *game* menarik.
2. *Game* mudah digunakan.
3. Menarik responden anak dalam memainkannya lagi.
4. Anak mengerti pembelajaran kosa kata kalimat dalam *game.*
5. Anak mengerti pembelajaran antonim dan sinonim dalam *game.*