BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi

Tahap implementasi merupakan tahap dimana aplikasi yang sudah dibangun dan dirancang siap dioperasikan. Tahapan ini juga akan menjelaskan mengenai implementasi perangkat keras, implementasi perangkat lunak dan implementasi program serta antarmuka aplikasi.

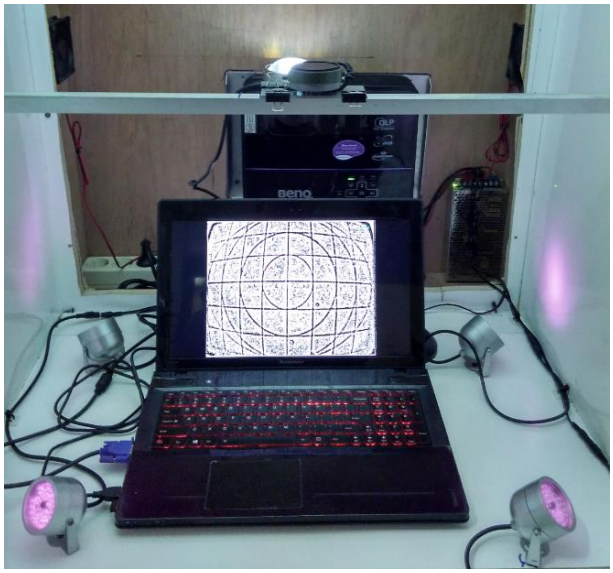
4.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk membangun teknologi *multitouch* pada aplikasi pengenalan batu artefak dapat dilihat pada tabel 4.1.

Tabel 4.1 Implementasi Perangkat Keras

NO	Perangkat Keras	Spesifikasi
1	*Processor*	Intel Core i-7 2,4 GHz
2	*Memory* RAM	2 GB
3	VGA *Card*	2 GB
4	Harddisk	500 GB
5	Monitor	15”
6	Keyboard	*Wireless*
7	Mouse	*Wireless*
8	Proyektor	*Short Throw*
9	4x Infra merah	1 IR 48 Led
10	Kamera PS3 Eye	640x480 Pixel
11	Lensa IR	720 nm 52mm
12	Box	80cmx80cmx90cm
13	Kaca + layar Proyeksi	7mm

Pada gambar 4.1 merupakan bentuk perangkat keras yang dimplementasikan dan digunakan pada aplikasi koleksi batu artefak menggunakan *marker* dengan teknologi *multitouch.*



Gambar 4.1 Implementasi Perangkat keras

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun teknologi *multitouch* pada aplikasi pengenalan batu artefak dapat dilihat pada tabel 4.2.

Tabel 4.2 Implementasi Perangkat Lunak

NO	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Windows 8.1
2	Reactivision	Versi 1.5
3	Adobe Flash	Versi CS6
4	*Library* TUIO	TUIO FLASH

4.1.3 Implementasi Class

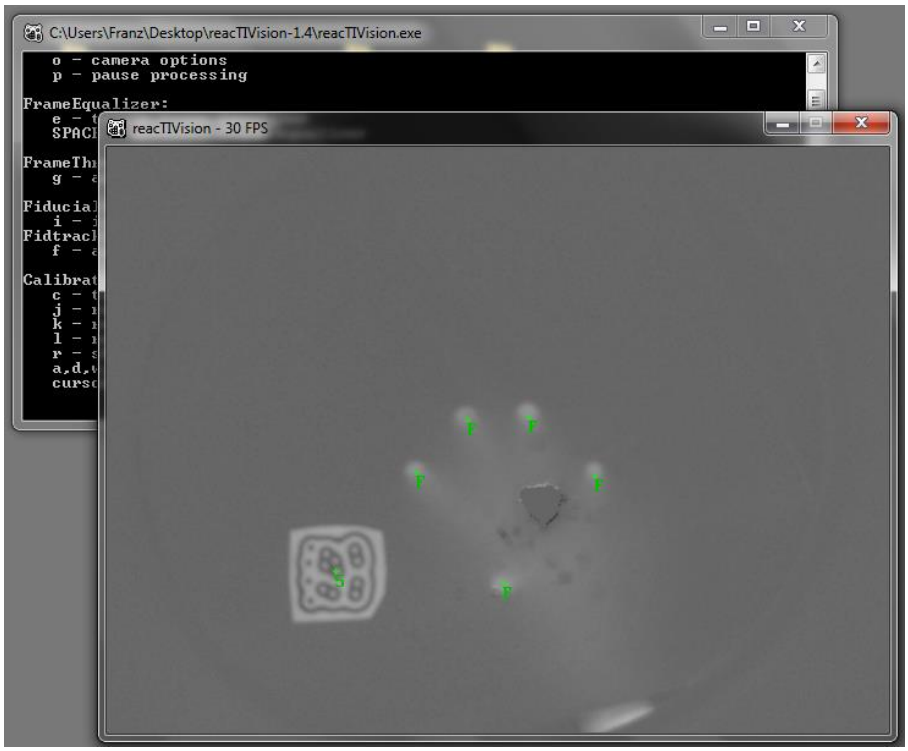
Implementasi *class* pada aplikasi pengenalan koleksi batu artefak menggunakan *marker* dengan teknologi *multitouch* merupakan implementasi dari perancangan *class diagram* yang terbentuk dari *use case diagram* kedalam bentuk fisik *file-file* dengan ekstensi .as dan yang digunakan untuk menjalankan proses dalam sistem. Implementasi *class* dapat dilihat pada tabel 4.3.

Tabel 4.3 Implementasi Class

NO	Nama Kelas	Nama file fisik
1	TUIO	TUIO.as
2	TouchEvent	TouchEvent.as
3	UI	UI.as
4	OSCConnectionEvent	OSCConnectionEvent.as
5	MovieClip	MovieClip.as
6	MainTimeline	MainTimeline.as

4.1.4 Implementasi Reactivision

Reactivision digunakan untuk membaca *marker* dan sentuhan jari pada perangkat keras *multitouch* yang akan digunakan sebagai *input* pada aplikasi *artefak stones* seperti gambar 4.2.



Gambar 4.2 Implementasi Reactivision

Pada tabel 4.4 merupakan konfigurasi yang *reactivision* yang digunakan pada aplikasi *artefak stones*, namun konfigurasi tersebut bukan menjadi patokan, karena dalam implementasi pembuatan *multitouch* berbeda-beda dari segi teknik pembuatannya, dari tingkat pencahayaannya, tingkat kualitas kamera dan elemen-elemen yang terdapat dalam pembuatan *multitouch*.

Tabel 4.4 Konfigurasi *Reactivision*

Fitur	Nilai
Tile Size	4
Gradient gate	0
Finger Size	24
Finger Sensitivity	143
Invert x-axis	0
Invert y-axis	0

4.1.5 Implementasi Antarmuka Sistem

Implementasi antarmuka yang dilakukan pada aplikasi pengenalan koleksi batu artefal menggunakan *marker* dengan teknologi *multitouch* terdiri dari beberapa tampilan. Berikut ini merupakan beberapa tampilan antar muka yang telah diimplementasikan.

1. Antarmuka animasi

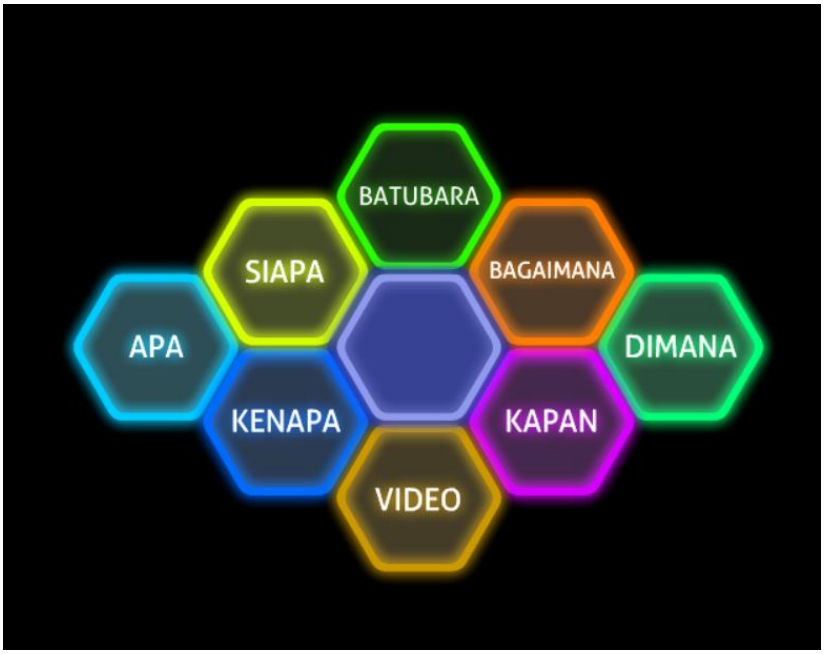
   Antarmuka animasi merupakan tampilan awal pada saat aplikasi pertama kali dibuka. Selain itu, pada saat posisi *idle* (tidak ada interaksi) animasi akan kembali dijalankan berulang-ulang sampai ada interkasi. Gambar 4.3 merupakan tampilan animasi dari aplikasi.



Gambar 4.3 Implementasi Animasi

2. Antarmuka menu utama

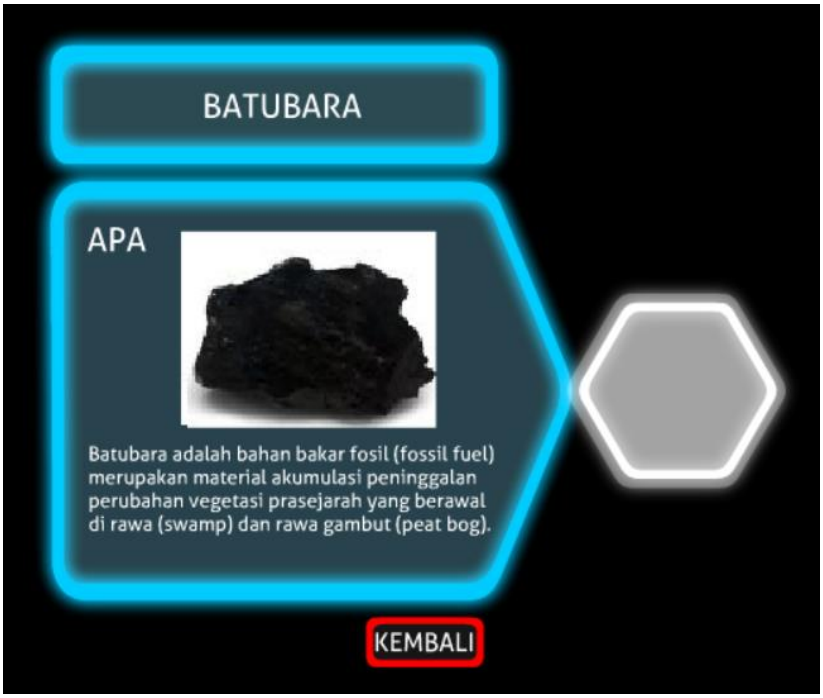
   Antarmuka menu utama merupakan tampilan yang akan muncul setelah pengguna meletakan *marker* diatas layar *multitouch*. Terdapat beberapa menu didalamnua, seperti menu apa, siapa, kenapa, bagaimana, kapan, dimana dan video. Gamabar 4.4 merupakan tampilan menu utama.



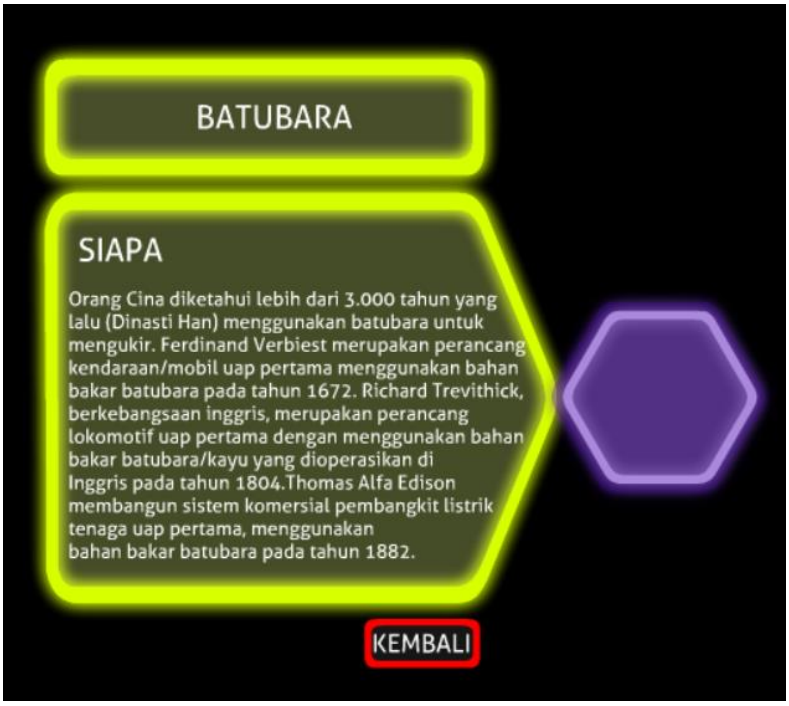
Gambar 4.4 Implementasi menu utama

3. Antarmuka Dalam Menu

   Antarmuka dalam menu merupakan tampilan yang akan ditampilkan ketika pengguna menekan tombol pada tampilan menu utama. Tampilan dalam menu dapat dilihat pada gambar 4.5.



Gambar 4.5 Implementasi tampilan menu apa

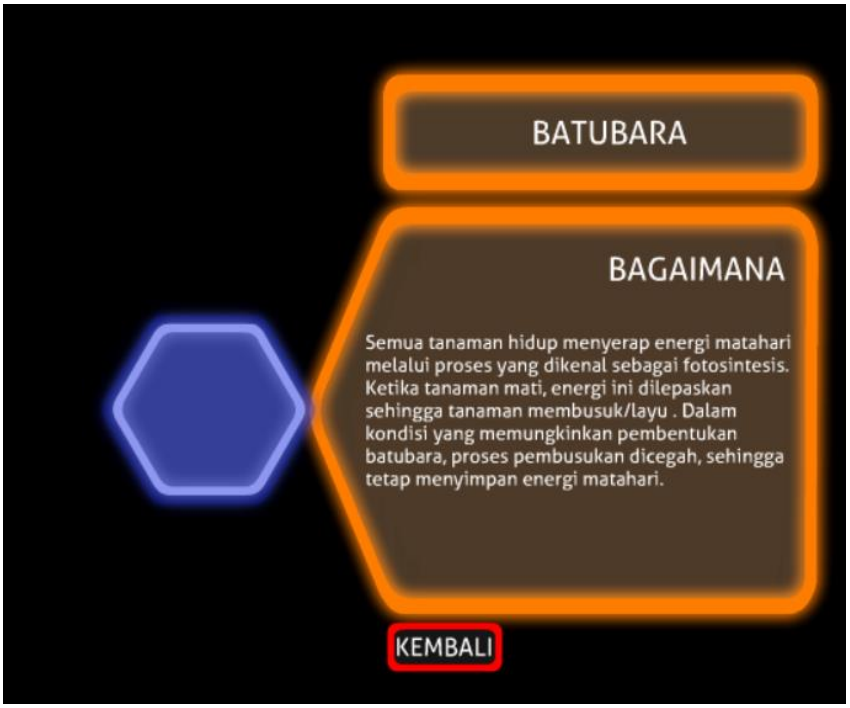


Gambar 4.6 Implementasi tampilan menu siapa

Pada gambar 4.6 merupakan implementasi tampilan dari menu siapa dan gambar 4.7 merupakan tampilan dari menu kenapa.

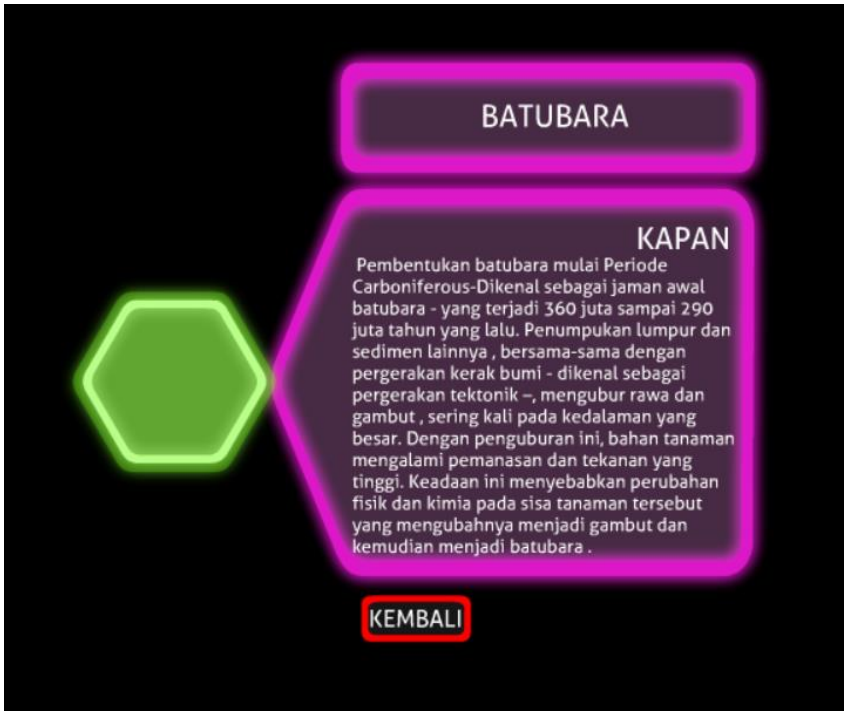


Gambar 4.7 Implementasi tampilan menu kenapa

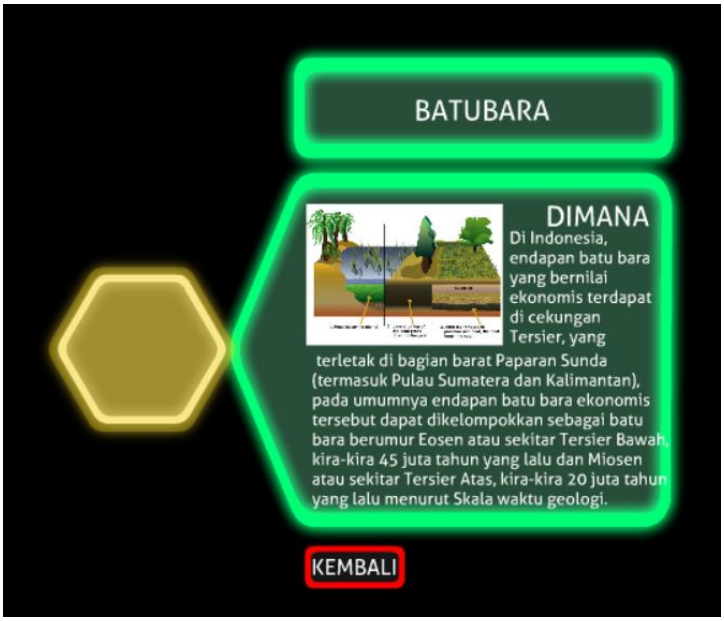


Gambar 4.8 Implementasi tampilan menu bagaimana

Pada gambar 4.8 merupakan implementasi tampilan dari menu bagaimana dan gambar 4.9 merupakan tampilan dari menu kapan.



Gambar 4.9 Implementasi tampilan menu kapan



Gambar 4.10 Implementasi tampilan menu dimana

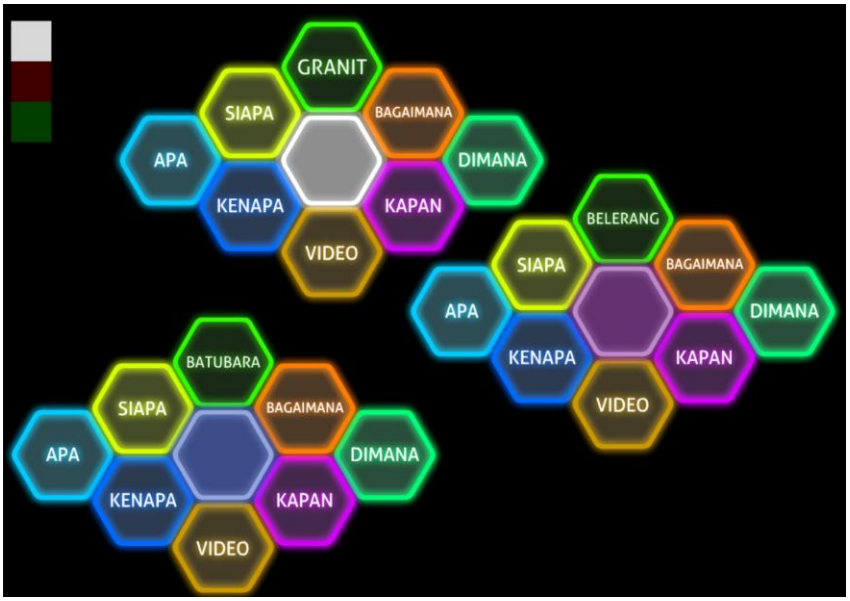
Pada gambar 4.10 merupakan implementasi tampilan menu dimana dan gambar 4.11 merupakan implementasi tampilan dari menu video. Video yang di tampilkan berukuran 300px x300px, dengan *mode* tanpa suara (*silent*). Jika pengguna memutar lebih dari satu video secara bersamaan suara yang dihasilkan tidak bertabrakan (*noise*).



Gambar 4.11 Implementasi tampilan menu video

4. Antarmuka menampilkan lebih dari satu menu

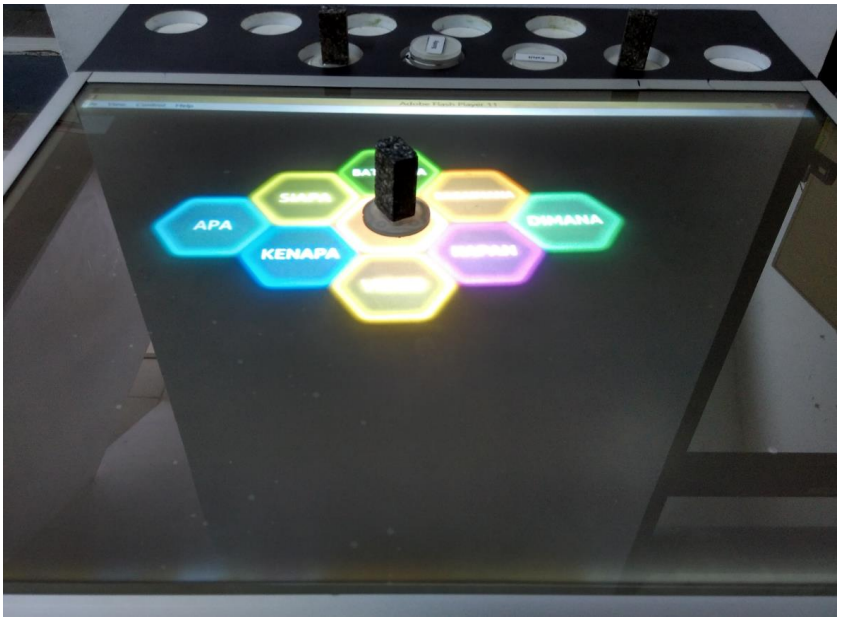
   Pada antarmuka ini merupakan gambaran tampilan menu jika pengguna membuka lebih dari satu informasi yang berbeda seperti gambar 4.12.



Gambar 4.12 Implementasi tampilan lebih dari satu menu

5. Implementasi Aplikasi *artefact Stones*

   Pada bagian ini merupakan tampilan aplikasi yang sudah terintegrasi dengan permukaan layar sentuhan seperti gambar 4.13.



Gambar 4.13 Implementasi aplikasi *artefact Stones*

4.2 Pengujian Sistem

Pengujian dilakukan dengan tujuan untuk menemukan kesalahan dan kekurangan pada perangkat lunak yang diuji. Pengujian tersebut dimaksudkan untuk mengetahui perangkat lunak yang dibuat apakah sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak. Dalam penelitian ini dilakukan pengujian terhadap sistem secara fungsional. Pengujian dilakukan dengan teknik pengujian *black box* yang lebih berfokus pada menemukan kesalahan program secara fungsional.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan terhadap sistem, merupakan pengujian terhadap fungsi-fungsi yang ada di dalam aplikasi, apakah fungsional dari aplikasi berfungsi sesuai yang diharapkan atau tidak. Rencana pengujian dari aplikasi dapat dilihat pada tabel 4.5

Tabel 4.5 Rencana Pengujian

NO	Komponen Yang diuji	Skenario Pengujian	Jenis Pengujian
1	Animasi	*Openning Application*	*Black Box*
		Idle Activity	
2	Menu Utama	Menampilkan Menu	*Black Box*
		Menampilkan lebih dari 1 menu	
		Memilih Menu Apa	
		Memilih Menu Siapa	
		Memilih Menu Kenapa	
		Memilih Menu Bagaimana	
		Memilih Menu Kapan	
		Memilih Menu Dimana	
		Memilih Menu Video	
		Menghilangkan Menu	
3	Interaksi *content*	Rotasi *content*	*Black Box*
		Geser *content*	
4	Pembacaan *Marker*	Jarak dan sudut antara kamera dengan *marker*	*Black Box*
5	Deteksi Sentuhan (*touch*)	Jumlah banyaknya sentuhan yang dapat dideteksi oleh kamera.	*Black Box*
6	Uji Performa Sistem	Dapat membaca *marker* ,sentuhan sebanyak mungkin dan seberapa cepat reaksi yang direspon oleh sistem	*Black Box*

4.2.2 Kasus dan Hasil Pengujian (Black Box Testing)

Kasus dan hasil uji perangkat lunak yang sudah dibangun dengan menggunakan teknik *black box* berdasarkan pengamatan yang dilakukan dan menghasilkan kesimpulan pengujian tersebut adalah sebagai berikut:

1. Pengujian Animasi

Pengujian ini merupakan proses dimana sistem menampilkan animasi pada saat aplikasi pertama kali dibuka dan pada saat tidak ada *marker* yang terbaca. Hasil dari proses pengujian dapat dilihat pada tabel 4.6.

Tabel 4.6 Pengujian Animasi

Kasus dan Hasil Uji			
Parameter	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Opening Application	Menjalankan Animasi	Sistem menjalankan animasi	[✓] Berjalan [ ] Gagal
idle	Menjalankan Animasi	Sistem menjalankan animasi	[✓] Berjalan [ ] Gagal

2. Pengujian Interaksi *Content*

Pengujian ini merupakan proses dimana *content* dapat menampilkan informasi, dapat memutar video, dapat berputar dan bergeser sesuai keinginan pengguna. Hasil dari proses pengujian dapat dilihat pada tabel 4.7.

Tabel 4.7 Pengujian Interaksi *Content*

Kasus dan Hasil Uji			
Parameter	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Rotasi *Marker*	*Content* dapat berputar sesuai posisi *marker*	*Content* dapat berputar sesuai posisi *marker*	[✓] Berjalan [ ] Gagal
Geser *Marker*	*Content* dapat bergeser sesuai posisi *marker*	*Content* dapat bergeser sesuai posisi *marker*	[✓] Berjalan [ ] Gagal

3. Pengujian Menu Utama

Pengujian ini merupakan proses dimana sistem menampilkan menu dan pengguna menyentuh atau memilih menu pada saat aplikasi pertama kali dibuka dan pada saat tidak ada *marker* yang terbaca. Hasil dari proses pengujian dapat dilihat pada tabel 4.8.

Tabel 4.8 Pengujian Menu Utama

Kasus dan Hasil Uji			
Parameter	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menampilkan Menu	Sistem menampilkan menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
Menampilkan lebih dari 1 menu	Sistem dapat menampilkan lebih dari satu menu	Sistem dapat menampilkan lebih dari satu menu	[✓] Berjalan [ ] Gagal
Memilih Menu Apa	Sistem dapat menampilkan menu apa	Sistem dapat menampilkan menu apa	[✓] Berjalan [ ] Gagal
Memilih Menu Siapa	Sistem dapat menampilkan menu siapa	Sistem dapat menampilkan menu siapa	[✓] Berjalan [ ] Gagal
Memilih Menu Kenapa	Sistem dapat menampilkan menu kenapa	Sistem dapat menampilkan menu kenapa	[✓] Berjalan [ ] Gagal
Memilih Menu Bagaimana	Sistem dapat menampilkan menu bagaimana	Sistem dapat menampilkan menu bagaimana	[✓] Berjalan [ ] Gagal
Memilih Menu Kapan	Sistem dapat menampilkan menu kapan	Sistem dapat menampilkan menu kapan	[✓] Berjalan [ ] Gagal
Memilih Menu Dimana	Sistem dapat menampilkan menu dimana	Sistem dapat menampilkan menu dimana	[✓] Berjalan [ ] Gagal
Memilih Menu Video	Sistem dapat menampilkan video	Sistem dapat menampilkan video	[✓] Berjalan [ ] Gagal
Menghilangkan Menu	Sistem dapat menghilangkan menu	Sistem dapat menghilangkan menu	[✓] Berjalan [ ] Gagal

4. Pembacaan *Marker*

Pembacaan *marker* merupakan yang dilakukan untuk mengukur jarak dan sudut antara kamera dengan *marker* yang diletakan pada layar *multitouch*.

Tabel 4.9 Pengujian Pembacaan Marker

Kasus dan Hasil Uji				
Parameter		**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Jarak	**Sudut**			
20 cm	90^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	60^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	45^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	30^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
30 cm	90^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	60^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	45^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	30^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
40 cm	90^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	60^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	45^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal
	30^0	Menampilkan Menu	Sistem menampilkan menu	[✓] Berjalan [ ] Gagal

Tabel 4.9 merupakan hasil pengujian dengan jarak dan sudut sebagai *parameter* masukannya. Jarak yang diuji antara *marker* dengan kamera adalah 20cm, 30cm, 40cm, sedangkan untuk sudut yang diuji adalah 90^o, 60^o, 45^o, dan 30^o. Dengan jarak 40 cm semua layar sentuhan dapat tertangkap kamera, namun memiliki kekurangan yaitu dibagian sudut-sudut layar sentuhan untuk pembacaan *marker* tidak stabil. Pembacaan *marker* tidak stabil karena lensa yang digunakan lensa *width* atau *fish eye*, menyebabkan beberapa *node* pada *marker* bentuknya tidak lagi presisi sehingga tidak dapat terbaca oleh kamera dengan sempurna.

5. Deteksi Sentuhan

Deteksi Sentuhan merupakan pengujian yang dilakukan untuk mengukur seberapa banyak sistem dapat mendeteksi atau membaca sentuhan pada layar *multitouch*. Hasil dari proses pengujian dapat dilihat pada tabel 4.10.

Tabel 4.10 Pengujian Deteksi Sentuhan

Kasus dan Hasil Uji			
Jumlah Sentuhan	**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
40	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[ ] Berjalan [✓] Gagal
35	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[ ] Berjalan [✓] Gagal
30	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[ ] Berjalan [✓] Gagal
25	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[ ] Berjalan [✓] Gagal
20	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[✓] Berjalan [ ] Gagal
15	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[✓] Berjalan [ ] Gagal
10	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[✓] Berjalan [ ] Gagal
5	Mendeteksi sentuhan	Sistem dapat mendeteksi sentuhan	[✓] Berjalan [ ] Gagal

Tabel 4.10 merupakan hasil pengujian sentuhan dengan jumlah sentuhan sebagai parameter. Dengan jumlah 5 – 20 sentuhan secara bersamaan pada layar sentuhan dapat terdeteksi dengan baik. Namun diatas 20 sentuhan yang dapat dideteksi tidak stabil, terutama dibagian sisi-sisi permukaan sentuhan, dikarenakan konfigurasi dan kalibrasi yang belum benar. Selain itu karena metode dari *reactivision*, tidak memeberikan nilai batas minimal dan maksimal suatu objek yang dikategorikan suatu sentuhan, menyebabkan pembacaan suatu sentuhan tidak stabil dan dibeberapa titik posisi *marker* dan sentuhan tidak sesuai dengan yang ditampilkan.

6. Uji Peforma

Uji performa merupakan pungujian yang dilakukan untuk mengukur seberapa banyak sistem dapat membaca *marker* dan *touch* yang terdapat pada layar sentuhan dan seberapa cepat reaksi sistem. Hasil dari proses pengujian performa dapat dilihat pada tabel 4.11.

Tabel 4.11 Pengujian performa

Kasus dan Hasil Uji				
Parameter		**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Jumlah Marker	**Jumlah Sentuhan**			
10	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
9	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Berjalan [✓] Gagal
8	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
7	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[ ] Cepat [✓] Lambat
6	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
5	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat

Tabel 4.11 Pengujian performa (Lanjutan)

Kasus dan Hasil Uji				
Parameter		**Keluaran yang diharapkan**	**Pengamatan**	**Kesimpulan**
Jumlah Marker	**Jumlah Sentuhan**			
4	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
3	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
2	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
1	20	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	15	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	10	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat
	5	Respon dari sistem terhadap *marker* dan sentuhan	Respon yang diberikan sistem	[✓] Cepat [ ] Lambat

Tabel 4.11 merupakan hasil pengujian peforma sistem terhadap pembacaan *marker* dan sentuhan pada layar. Dengan jumlah 7-10 *marker* dengan jumlah sentuhan 5 - 20 sentuhan secara bersamaan pada layar sentuhan dapat terdeteksi namun reaksi dari sistem melambat. Dengan jumlah 1-6 *marker* dengan jumlah sentuhan 5 - 20 sentuhan secara bersamaan pada layar sentuhan dapat terdeteksi dengan baik dan reaksi dari sistem stabil, terutama dibagian tengah permukaan sentuhan. Selain dikarenakan konfigurasi dan kalibrasi yang belum benar, kecepatan kamera yang digunakan yaitu 30fps, sehingga menyebabkan

pembacaan *marker* dan sentuhan kurang maksimal karena keterbatasan kecepatan kamera yang digunakan.



Gambar 4.14 Uji Peforma aplikasi *artefact Stones*

4.2.3 Kesimpulan Pengujian Black Box

Berdasarkan hasil pengujian *black box* terhadap aplikasi yang dilakukan maka dapat disimpulkan bahwa, aplikasi pengenalan koleksi batu artefak menggunakan *marker* dengan teknologi *multitouch* yang dibangun secara fungsional berjalan sesuai dengan yang diharapkan. Dari hasil pengujian performa reaksi dan kecepatan sistem terhadap pembacaan *marker* dan sentuhan mampu membaca 6 *maker* dan 20 sentuhan di posisi tengah layar sentuhan, namun selain di posisi tengah layar sentuhan pembacaan *marker* dan pendeteksian sentuhan belum stabil.