

BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi yang dibangun. Tahapan ini dilakukan setelah analisis dan perancangan selesai dilakukan dan selanjutnya akan diimplementasikan kedalam bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap aplikasi. Aplikasi yang telah dibangun akan diimplementasikan untuk mengetahui apakah aplikasi tersebut dapat berjalan sesuai dengan tujuannya atau tidak.

4.1 Implementasi

Tujuan implementasi aplikasi adalah untuk menjelaskan tentang manual modul kepada semua user yang akan menggunakan aplikasi. Sehingga user tersebut dapat merespon apa yang ditampilkan di aplikasi dan memberikan masukan kepada pembuat aplikasi untuk dilakukan perbaikan agar aplikasi lebih baik lagi.

4.1.1 Implementasi Perangkat pembangun

Perangkat lunak yang digunakan pada aplikasi komputer yang digunakan untuk implementasi teknologi *augmented reality* pada aplikasi adalah sebagai berikut:

1. Perangkat Keras

Perangkat keras yang digunakan untuk pembuatan aplikasi dijelaskan pada **Tabel 4-1**.

Tabel 4-1 Perangkat Keras yang Digunakan

Komputer	Spesifikasi Perangkat Keras
Prosesor	Intel(R) Core(TM) i7-2600 3,40GHz
Layar	1366 x 768
Monitor	LCD 14 inc
Memori	Memori 8 GB
Hardisk	500 GB
Kamera	1,3 Mega Pixel

2. Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan aplikasi dijelaskan pada **Tabel 4-2**.

Tabel 4-2 Perangkat Lunak yang Digunakan

Komputer	Spesifikasi Perangkat Lunak
Aplikasi Operasi	Microsoft Windows 7
Bahasa Pemrograman	C#
Engine	Unity
SDK	<i>Vuforia</i>

4.1.2 Implementasi Antarmuka

Pada tahap ini dilakukan penerapan hasil perancangan antarmuka ke dalam aplikasi yang dibangun dengan menggunakan perangkat lunak yang telah dipaparkan pada sub bab implementasi perangkat lunak, yang tercantum pada gambar di bawah ini dan untuk antarmuka selanjutnya dapat dilihat pada :

1. Tampilan Utama Eksekusi aplikasi awal akan menampilkan tampilan antarmuka menu utama. Seperti yang terlihat pada **Gambar 4-1**.



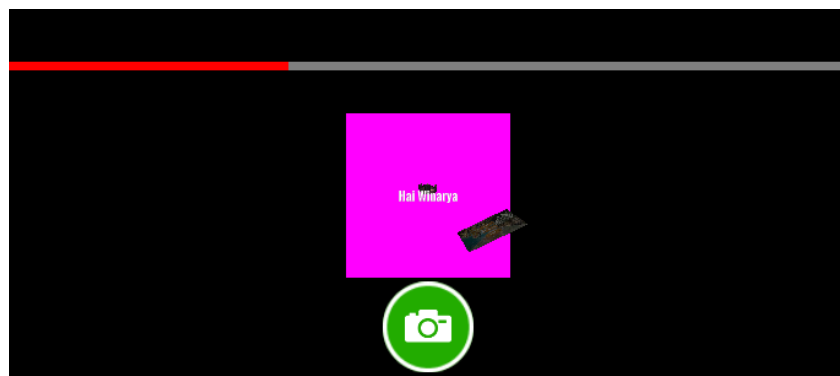
Gambar 4-1 tampilan antarmuka menu utama

2. Tampilan antarmuka Konten Tambahan. Seperti yang terlihat pada **Gambar 4-2**.



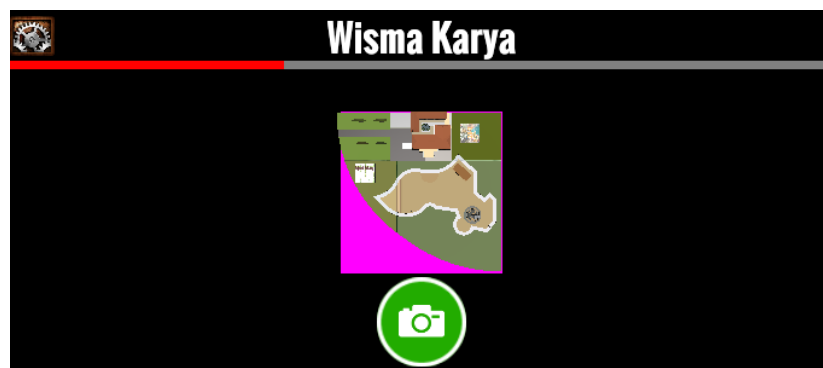
Gambar 4-2 tampilan antarmuka tambahan konten

3. Tampilan antarmuka *Introduction* Seperti yang terlihat pada **Gambar 4-3**.



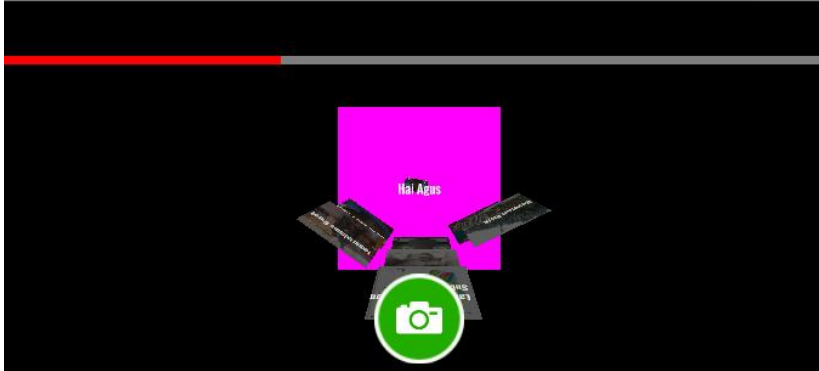
Gambar 4-3 tampilan antarmuka *introduction*

4. Tampilan antarmuka *Main_Konten* Seperti yang terlihat pada **Gambar 4-4**.



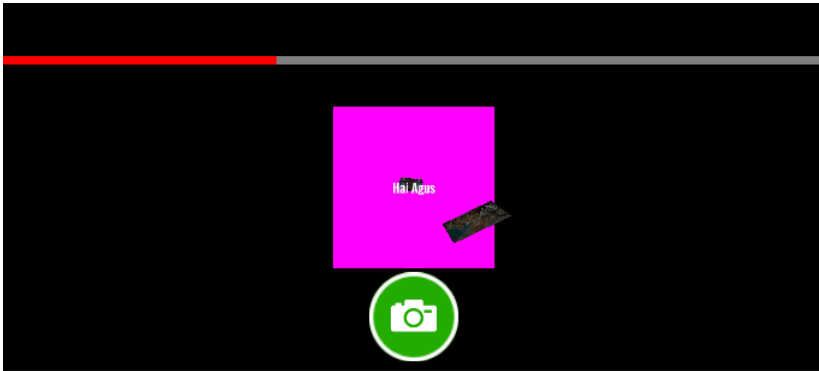
Gambar 4-4 tampilan antarmuka *main* konten

5. Tampilan antarmuka Sejarah Subang Seperti yang terlihat pada **Gambar 4-5** .



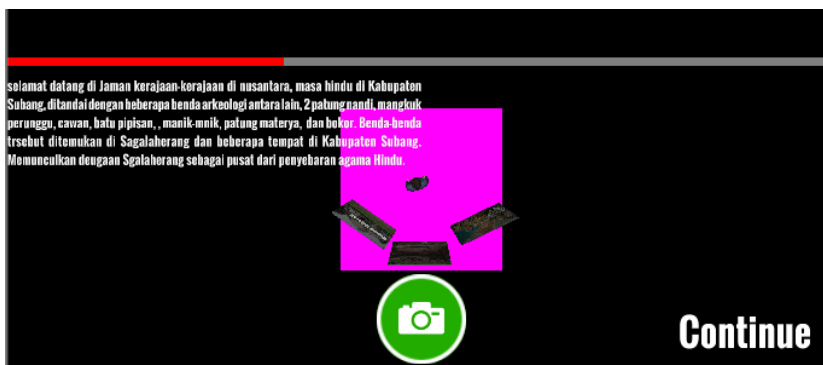
Gambar 4-5 tampilan antarmuka sejarah subang

6. Tampilan antarmuka Prasejarah Seperti yang terlihat pada **Gambar 4-6**.



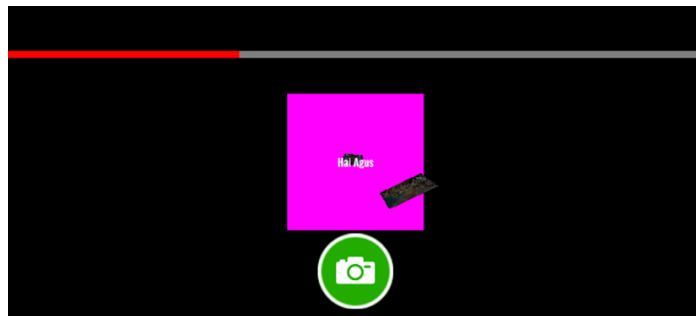
Gambar 4-6 tampilan antarmuka prasejarah

7. Tampilan antarmuka Hindu, Budha, dan islam Seperti yang terlihat pada **Gambar 4-7**.



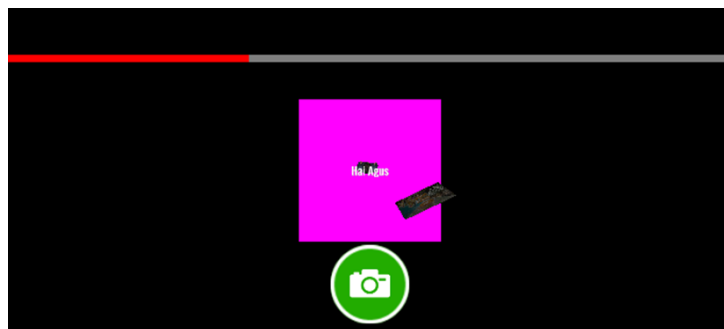
Gambar 4-7 tampilan antarmuka Hindu, Budha dan Islam

8. Tampilan antarmuka Imperialisme Seperti yang terlihat pada **Gambar 4-8**.



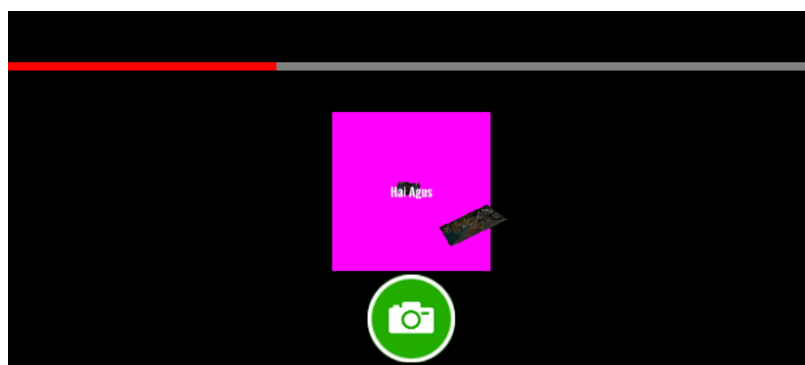
Gambar 4-8 tampilan antamuka imperialisme

9. Tampilan antarmuka Pergerakan Nasional dan Pendudukan Jepang Seperti yang terlihat pada **Gambar 4-9**.



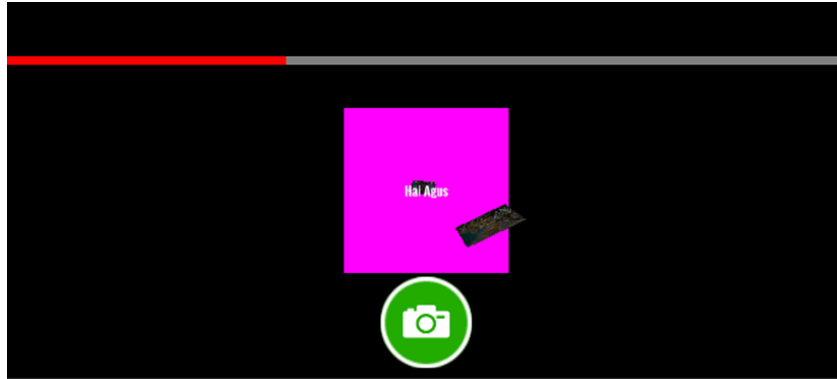
Gambar 4-9 tampilan antarmuka pergerakan nasional dan pendudukan Jepang

10. Tampilan antarmuka Revolusi Fisik Seperti yang terlihat pada **Gambar 4-10**.



Gambar 4-10 tampilan antamuka revolusi fisik

11. Tampilan antarmuka Kabupaten Subang Seperti yang terlihat pada **Gambar 4-11**.



Gambar 4-11 tampilan antarmuka Kabupaten Subang

4.2 Pengujian

Pengujian aplikasi dilakukan bertujuan untuk menemukan kesalahan atau kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

Dalam penelitian ini pengujian yang dilakukan terhadap aplikasi yaitu pengujian secara fungsional (alpha) dan beta. Metode yang digunakan dalam pengujian ini adalah pengujian blackbox yang berfokus pada persyaratan fungsional dari aplikasi yang dibangun.

4.2.1 Pengujian Blackbox

Pengujian blackbox dilakukan pada sisi pengembangan yang merekam semua kesalahan dan masalah pemakaian. Pengujian blackbox dilakukan pada sebuah lingkungan yang terkendali.

Rencana Pengujian

Rencana pengujian adalah pengujian terhadap fungsi-fungsi yang ada di dalam aplikasi yang dibangun, apakah fungsional dari aplikasi berfungsi sesuai yang diharapkan atau tidak. Berikut ini **Tabel 4-3** rencana pengujian dari aplikasi yang dibangun.

Tabel 4-3 Tabel Rencana Pengujian

No	Komponen Yang Diuji	Butir Pengujian	Jenis Pengujian
1	Menu	Memilih Tombol Mulai, Memilih Tombol Tambah Konten0, Memilih Tombol <i>build</i> (kamera), Memilih Tombol Keluar	Blackbox
2	Pengujian use case	Menjalankan proses, menyajikan konten, Menjalankan proses mendeteksi <i>feature</i> , Menjalankan proses membentuk <i>marker</i> , Menjalankan proses mencocokkan <i>marker</i> , Menjalankan proses Merender informasi,	Blackbox

4.2.2 Kasus dan Hasil Pengujian Blackbox

Kasus dan hasil pengujian berisi pemaparan dari rencana pengujian yang telah disusun pada skenario pengujian. Pengujian ini dilakukan secara *black box* dengan hanya memperhatikan masukan ke dalam aplikasi dan keluaran dari masukan tersebut. Berdasarkan rencana pengujian, maka dapat dilakukan pengujian blackbox pada aplikasi sebagai berikut:

4.2.2.1 Pengujian Tampilan Menu

Pengujian tampil menu merupakan pengujian fungsionalitas untuk menampilkan menu yang telah diterapkan dalam aplikasi, seperti terlihat pada **Tabel 4-4** berikut :

Tabel 4-4 Pengujian Tampilan Menu

No	Skenario Uji	Hasil yang diharapkan	Hasil Pengujian
1	Memilih Tombol Mulai	Ketika user menekan tombol mulai, maka aplikasi membuka kamera dan memulai <i>menscan frame</i> untuk mendapatkan kualitas <i>frame</i> berdasarkan kekayaan <i>feature</i>	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
2	Memilih Tambahan Konten	Menampilkan scene DLC dan dapat mendownload asset bundles	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
3	Memilih tombol buid (kamera)	Membuat target baru dan menyimpannya di dataset sebagai penyimpanan sementara yang dihancurkan ketika berpindah scene	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
4	Memilih tombol keluar	Menghentikan aplikasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

4.2.2.2 Pengujian Use Case

Pengujian use case adalah pengujian yang dilakukan pada proses-proses yang dapat dilakukan setelah menerima intruksi oleh pengguna, aplikasi akan melaksanakan proses tersebut hingga selesai dan menghasilkan keluaran yang diharapkan, ditunjukkan pada **Tabel 4-5**.

Tabel 4-5 Pengujian use case

No	Kasu/ Diuji	Skenario uji	Hasil yang diharapkan	Hasil Pengujian
1	Membuka Kamera	Aplikasi aplikasi dimulai	Aplikasi membuka kamera, menscan <i>frame</i> hasil calon <i>marker</i> dan mendeteksi <i>frame</i> serta mengirimkan feedback berupa warna <i>plane</i>	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
2	Membentuk <i>marker</i>	User menyentu h tombol <i>build</i> (kamera)	Aplikasi mendeteksi <i>feature</i> dari calon <i>marker</i> dan menyimpannya dalam penyimpanan dataset berupa array yang kemudian di cocokkan dengan masukan kamera secara runtime dan merender informasi	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
3	Menambah konten	User menye ntuh tombol downl oad pada menu tamba	Aplikasi melakukan koneksi melalui protokol http dan melakukan request untuk mendownload file aset bundle yang dimaksud	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

		han konten		
4	Mengukur Kualitas <i>Frame</i>		Aplikasi mengukur kualitas <i>frame</i> berdasarkan standar <i>vuforia</i> SDK untuk dijadikan <i>marker</i> . Mengimplementasikannya sebagai warna pada <i>plane</i>	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
5	Mengekstrak <i>Feature</i>		Aplikasi dapat mengukur kualitas <i>Frame</i> dan mencocokkan <i>marker</i> dengan mengekstrak <i>feature</i>	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
6	Merender Informasi		Aplikasi merender informasi yang telah ditanam dalam scene setelah menjalankan proses mencocokkan <i>marker</i>	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
7	Mencocokkan <i>Marker</i>		Aplikasi mencocokkan <i>feature</i> pada calon <i>marker</i> dan yang telah disimpan	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

			dalam dataset setelah proses membentuk <i>marker</i> berhasil	
--	--	--	--	--

4.2.3 Pengujian Jarak

Pada pengujian jarak penggunaan *marker*, semakin dekat jarak *marker* dengan kamera akan mengakibatkan ukuran *marker* yang terdeteksi semakin besar, sehingga bisa tertangkap dengan baik. Namun ketika jarak kamera dengan *marker* semakin jauh maka ukuran *marker* yang tertangkap kamera semakin kecil, sehingga pola *marker* menjadi tidak jelas dan mengakibatkan *marker* tidak terdeteksi.

Salah satu permasalahan dari jarak antara kamera dan *marker* adalah tingkat kefokusan dari gambar yang ditangkap oleh kamera. Kualitas kamera yang memiliki fitur autofocus maka deteksi *marker* akan berjalan dengan baik. Berikut **Tabel 4-6** adalah hasil pengujian *marker*.

Tabel 4-6 Pengujian Jarak

No	Jarak (Centimeter)	Hasil Pengujian Jarak
1	100	Terdeteksi, namun objek 3D tidak tampil
2	60	Terdeteksi, namun objek 3D tidak stabil
3	50	Terdeteksi, namun objek 3D tidak stabil
4	40	Terdeteksi, namun objek 3D tidak stabil
5	30	Terdeteksi dengan baik objek 3D stabil
6	20	Terdeteksi dengan baik objek 3D stabil

4.2.4 Pengujian Sudut

Pada pengujian sudut penggunaan *marker*, semakin tegak lurus sudut terhadap *marker* dengan kamera akan mengakibatkan hasil render yang terdeteksi semakin baik, sehingga bisa menghasilkan persepsi pengguna dengan baik. Namun ketika sudut kamera terhadap *marker* sejajar jauh maka area *marker* yang tertangkap

kamera semakin kecil, sehingga pola *marker* menjadi tidak jelas dan mengakibatkan *marker* tidak terdeteksi. Berikut **Tabel 4-7** adalah hasil pengujian sudut.

Tabel 4-7 Pengujian Sudut

No	Sudut (Derajat)	Hasil Pengujian Sudut
1	180	Tidak Terdeteksi, objek 3D tidak tampil
2	270	Terdeteksi dengan baik, namun objek 3D stabil
3	90	Terdeteksi dengan baik, objek 3D stabil
4	45	Terdeteksi dengan baik, objek 3D stabil
5	30	Terdeteksi dengan baik, objek 3D stabil
6	20	Terdeteksi dengan baik, objek 3D tidak stabil

4.2.5 Pengujian Pengguna

Pengujian ini dilakukan dengan melakukan kuisisioner kepada masyarakat Kabupaten Subang yang memiliki ketertarikan kepada sejarah lokal atau daerah. Pengujian ini bertujuan untuk mengetahui respon pengguna terhadap sistem yang telah dibangun.

4.2.5.1 Pengujian Kuisisioner

Pengujian ini menggunakan metode User Acceptance Test, pengujian ini bertujuan untuk mencoba aplikasi dan mengetahui apakah aplikasi yang dibangun sudah memenuhi tujuannya. Pada pengujian ini, kuisisioner diberikan kepada 35 responden, dengan 5 pertanyaan yang mewakili tujuan dari pembangunan aplikasi ini, dan dihitung menggunakan skala Likert, dengan skor dari skala masing-masing jawaban yang dijelaskan pada **Tabel 4-8**.

Tabel 4-8 Skala Likert

Skala	Skor
Sangat Setuju (SS)	5
Setuju (S)	4
Cukup (C)	3
Tidak Setuju (TS)	2
Sangat Tidak Setuju (STS)	1

4.2.5.2 Skenario Pengujian Kuisioner

Skenario pengujian yang dilakukan kepada masyarakat Kabupaten Subang yang berminat untuk mempelajari dan melestarikan sejarah lokal atau daerah Kabupaten Subang, dijelaskan pada Tabel **Tabel 4-9**.

Tabel 4-9 Skenario Pengujian Kuisioner

No	Tujuan	Pertanyaan
1	Membantu menarik minat masyarakat Kabupaten Subang untuk mengunjungi pusat-pusat pengetahuan sejarah Kabupaten Subang..	1. Apakah anda setuju, aplikasi <i>retrokognitor</i> memberikan anda informasi pengetahuan sejarah lokal Kabupaten Subang ? 2. Apakah anda setuju, aplikasi <i>retrokognitor</i> membuat anda ingin mempelajari sejarah Kabupaten Subang lebih dalam lagi ?
2	Membuat media informasi yang membantu memaksimalkan jangkauan penyampaian informasi sejarah Kabupaten Subang.	3. Apakah anda setuju, informasi yang anda dapatkan dari <i>retrokognitor</i> merupakan informasi yang tepat ?
3	Menyediakan media informasi menyampaikan informasi mengenai sejarah	4. Apakah anda setuju, aplikasi ini membantu jangkauan

	Kabupaten Subang secara urut utuh dan berkelanjutan	penyebaran informasi situs bersejarah dan sejarah Kabupaten Subang ? 5. Apakah anda setuju, aplikasi ini menyuguhkan materi sejarah secara berurutan ?
--	---	---

4.2.5.3 Perhitungan Skala Likert

Perhitungan skala *likert* berdasarkan jawaban dari para responden, memiliki beberapa langkah, sebagai berikut :

1. Menentukan skor maksimal, yaitu :

$$\begin{aligned} \text{Skor maksimal} &= \text{skor terbesar} \times \text{jumlah responden} \\ &= 25 \times 35 \\ &= 875 \end{aligned}$$

2. Menentukan total skor minimal, yaitu :

$$\begin{aligned} \text{Total skor maksimal} &= \text{skor terkecil} \times \text{jumlah responden} \\ &= 17 \times 35 \\ &= 595 \end{aligned}$$

3. Menentukan nilai median, yaitu :

$$\begin{aligned} \text{Nilai Median} &= (\text{total skor maksimal} + \text{total skor minimal}) / 2 \\ &= (875 + 595) / 2 \\ &= 74 \end{aligned}$$

4. Menentukan nilai kuartil 1, yaitu :

$$\begin{aligned} \text{Kuartil 1} &= (\text{total skor minimal} + \text{nilai median}) / 2 \\ &= (595 + 74) / 2 \\ &= 35 \end{aligned}$$

5. Menentukan nilai kuartil 3, yaitu :

$$\text{Kuartil 3} = (\text{total skor maksimal} + \text{nilai median}) / 2$$

$$= (875 + 74) / 2$$

$$= 475$$

2. Setelah menentukan skor, buat skala dengan memasukkan skor minimal sampai skor maksimal yang sudah diperoleh dari perhitungan sebelumnya, seperti **Gambar 4- 12.**



Gambar 4- 12 Skala

3. Membuat batas skor berdasarkan perhitungan sebelumnya, dan dijelaskan pada Tabel 4.15.
- Kategori Sangat Positif, yaitu daerah yang dibatasi oleh kuartil 3 dan skor maksimal ($\text{kuartil } 3 \leq x \leq \text{skor maksimal}$).
 - Kategori Positif, yaitu daerah yang dibatasi oleh median dan kuartil ketiga ($\text{median} \leq x < \text{kuartil } 3$).
 - Kategori Negatif, yaitu daerah yang dibatasi oleh kuartil 1 dan median ($\text{kuartil } 1 \leq x < \text{median}$).
 - Kategori Sangat Negatif, yaitu daerah yang dibatasi oleh skor minimal dan kuartil 1 ($\text{skor minimal} \leq x < \text{kuartil } 1$).

Tabel 4-10 Tabel Kategori Sikap

Kategori	Batas Skor
Sangat Positif	$875 \leq x \leq 595$
Positif	$594 \leq x \leq 475$
Negatif	$474 \leq x \leq 74$
Sangat Negatif	$73 \leq x \leq 35$

4.2.5.4 Hasil Pengujian Kuisisioner

Hasil dari setiap jawaban kuisisioner, yang diisi oleh para responden, ditunjukkan pada Tabel 4-11.

Tabel 4-11 Hasil Pengujian Kuisisioner

Responden	Pernyataan					Skor
	1	2	3	4	5	
1	4	5	3	4	5	21
2	4	5	3	4	5	21
3	4	3	3	5	4	19
4	4	5	3	3	4	19
5	4	5	3	4	5	21
6	4	3	3	4	5	19
7	5	4	3	3	4	19
8	5	4	4	3	5	21
9	4	4	3	4	4	19
10	5	4	4	3	5	21
11	5	4	4	4	5	22
12	4	4	5	4	5	22
13	4	4	3	4	5	20
14	4	4	4	4	4	20
15	5	4	3	2	4	18
16	5	4	5	4	5	23
17	5	5	4	3	4	21
18	4	5	4	5	4	22
19	4	3	4	4	4	19
20	4	5	4	3	5	21
21	5	4	4	5	5	23
22	5	4	3	4	5	21
23	5	3	2	3	5	18
24	5	4	5	4	5	23

25	4	3	2	3	4	16
26	4	3	5	4	5	21
27	4	3	5	3	5	20
28	4	5	3	4	5	21
29	4	5	5	5	5	24
30	3	4	3	4	3	17
31	5	4	5	3	4	21
32	5	4	4	3	4	20
33	5	5	4	3	4	21
34	4	4	4	4	4	20
35	4	4	3	4	5	20
Jumlah						714

Berdasarkan skor yang didapat dari hasil kuisioner kepada 35 orang responden, didapatkan total skor 714. Berdasarkan kategori sikap, maka skor 714 masuk ke dalam kategori sikap sangat positif.

4.2.5.5 Kesimpulan Pengujian Kuisioner

Berdasarkan hasil pengujian secara keseluruhan, maka dapat ditarik kesimpulan, bahwa pengujian fungsionalitas aplikasi *retrokognitor* ini sudah sesuai dengan kebutuhan pengguna. Selanjutnya dalam pengujian kepada pengguna, responden dapat memahami materi pada aplikasi *retrokognitor*. Sehingga, dapat disimpulkan bahwa sistem yang dibangun telah memenuhi tujuan pembangunannya.

