

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Masalah

Dengan beralihnya fungsi sepeda motor yang bukan lagi hanya menjadi alat transportasi namun juga menjadi bagian dari gaya hidup masyarakat membuat kebutuhan untuk mempercantik sepeda motor pun meningkat. Salah satu cara untuk mempercantik sepeda motor adalah dengan cara memodifikasi sepeda motor tersebut, dimana para pengguna sepeda motor akan merasa lebih percaya diri setelah sepeda motor mereka di modifikasi karena akan menimbulkan kesan unik.

Akan tetapi berdasarkan hasil kuesioner, para pengguna sepeda motor mengeluhkan kendala di saat akan modifikasi motor. Salah satunya adalah tidak dapat memvisualisasikan barang modifikasi yang akan di pasang pada motornya, sedangkan belum ada aplikasi yang dapat membantu permasalahan tersebut.

3.2. Analisis Aplikasi Sejenis

Analisis aplikasi sejenis merupakan sebuah analisis yang membahas konten, fitur-fitur, serta komponen-komponen apa saja yang ada di dalam aplikasi yang sejenis agar menjadi acuan aplikasi yang akan di bangun.

3.2.1. Yamaha MyGarage

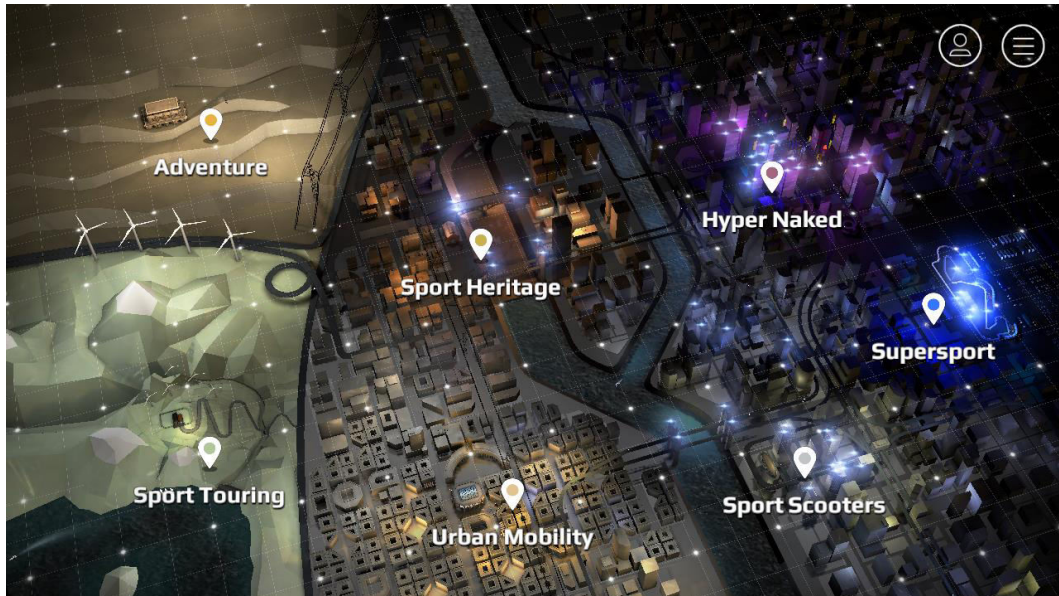
Yamaha MyGarage sebuah aplikasi Android yang di kembangkan oleh YAMAHA MOTOR Europe dimana pengguna dapat mencoba aksesoris sepeda motor dengan render 3 dimensi.

3.2.1.1. Komponen Yamaha MyGarage

1. Menu Utama

Di menu utama dari aplikasi ini pengguna langsung disuguhkan dengan pemilihan tipe motor yang akan di modifikasi. Tipe-tipe motor yang tersedia

adalah: *Adventure*, *Sport Touring*, *Sport Heritage*, *Urban Mobility*, *Hyper Naked*, *Supersport*, *Sport Scooters*.



Gambar 3.1 Tampilan Menu Utama MyGarage

2. Menu pemilihan Motor

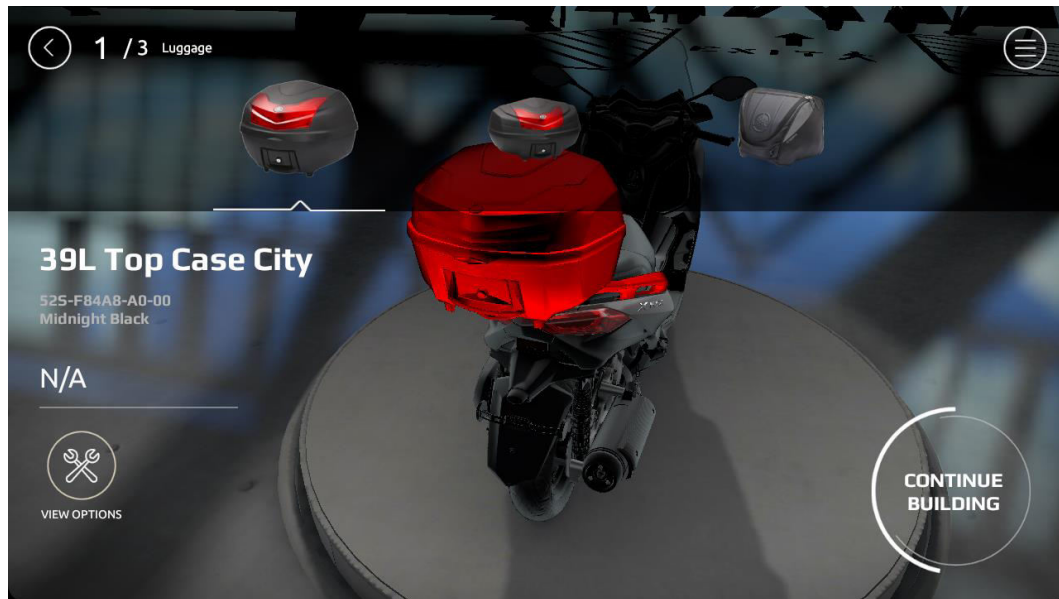
Setelah memilih tipe motor, maka pengguna dapat memilih motor apa yang akan di modifikasi



Gambar 3.2 Menu Pemilihan Motor MyGarage

3. Modifikasi Sepeda Motor

Setelah memilih sepeda ,motor maka pengguna dapat memulai modifikasi sepeda motor. Menu modifikasi yang tersedia adalah *colour*, *functional*, *luggage*, *comfort*, *screens*, dan *cosmetics*.



Gambar 3.3 Menu Modifikasi MyGarage

3.2.1.2. Hasil Analisis Aplikasi MyGarage

Berdasarkan analisis dari aplikasi Yamaha MyGarage dapat diperoleh beberapa poin yang penting terkait aplikasi tersebut. Fungsi utama aplikasi ini adalah memudahkan pengguna mendesain sepeda motornya.

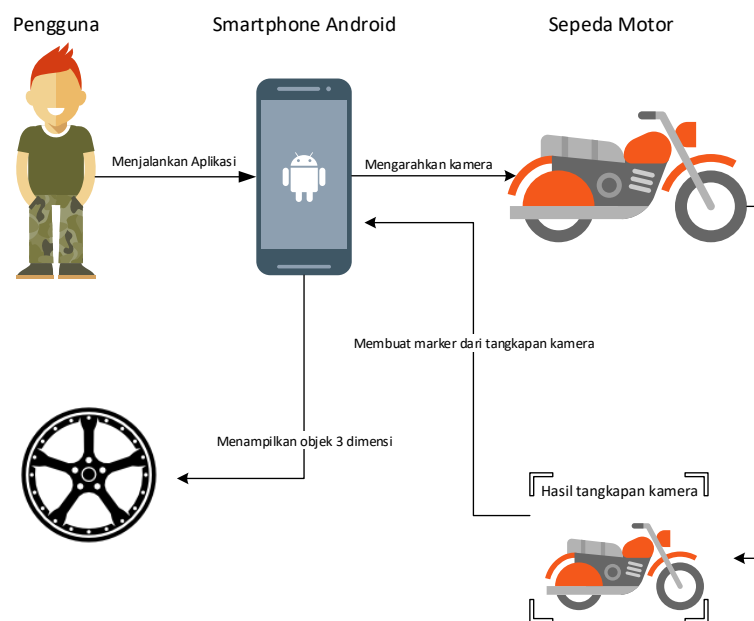
Kelebihan dari aplikasi ini adalah Yamaha MyGarage dapat mendesain modifikasi dalam bentuk 3 dimensi sehingga dapat membandingkan hasil modifikasi yang ada pada aplikasi dengan sepeda motornya sendiri.

3.3. Analisis Sistem

Analisis sistem merupakan identifikasi dari komponen-komponen dalam suatu sistem informasi agar dapat mengevaluasi permasalahan-permasalahan yang terjadi dan kebutuhan-kebutuhan yang diharapkan.

3.3.1. Deskripsi Sistem

Sistem yang akan dibangun adalah aplikasi *mobile* untuk *smartphone* dengan *platform* Android dengan menggunakan teknologi *Augmented Reality*. Penggunaan *platform* Android adalah karena sebagian besar pengguna *smartphone* menggunakan Android, sehingga aplikasi yang akan dibangun mudah tersedia di *smartphone* pengguna. Teknologi *Augmented Reality* digunakan agar pengguna dapat secara langsung menampilkan objek modifikasi kepada sepeda motornya masing-masing bukan kepada sepeda motor objek 3 dimensi agar pengguna dapat lebih memvisualisasikan objek modifikasi pada motornya. Untuk dapat menampilkan objek 3 dimensi pada *Augmented Reality* digunakan metode *markerless-based* dengan teknik *User Defined Target*. Input yang dimasukkan ke dalam sistem adalah gambar hasil tangkapan kamera yang di dapat dari pengguna. Proses yang dilakukan sistem menggunakan *Library Vuforia SDK*. Pemilihan *Librari Vuforia* ini dilakukan karena *Library* ini menawarkan fitur *Image Tracking* menggunakan *User Defined Target* dimana objek 3 dimensi akan muncul di hadapan gambar yang di tankap oleh kamera pengguna. Hal ini mendukung solusi visualisasi objek modifikasi secara *real-time* terhadap sepeda motor pengguna. Deskripsi sistem dapat dilihat pada Gambar 3.4.



Gambar 3.4 Deskripsi Sistem

Pengguna menjalankan aplikasi, kemudian sistem aplikasi akan membuka kamera untuk di arahkan pengguna pada sepeda motor. Setelah pengguna mengarahkan ke bagian tertentu pada sepeda motor kemudian pengguna dapat memilih objek apa yang akan di tampilkan lalu pengguna menekan tombol tangkap gambar. Setelah pengguna menangkap gambar kemudian system akan membuat gambar tersebut menjadi *marker* yang akan digunakan sebagai *tracking* objek 3 dimensi.

3.3.2. Analisis Metode Pembangunan Perangkat Lunak

3.3.2.1. Konsep

Konsep aplikasi yang akan dibangun merupakan aplikasi multimedia yang memanfaatkan teknologi augmented reality berbasis android yang dapat digunakan untuk membantu pengguna memvisualisasikan desain modifikasi sepeda motor.

Tabel 3.1 Pembangunan Konsep

Unsur yang Membangun	Deskripsi
Judul	Perancangan Aplikasi Desain Modifikasi Sepeda Motor Menggunakan Augmented Reality
Pengguna	Publik
Objek 3 Dimensi	Objek 3 dimensi yang digunakan adalah objek nyata dari aksesoris modifikasi sepeda motor yang dibuat kembali kedalam objek 3 dimensi.

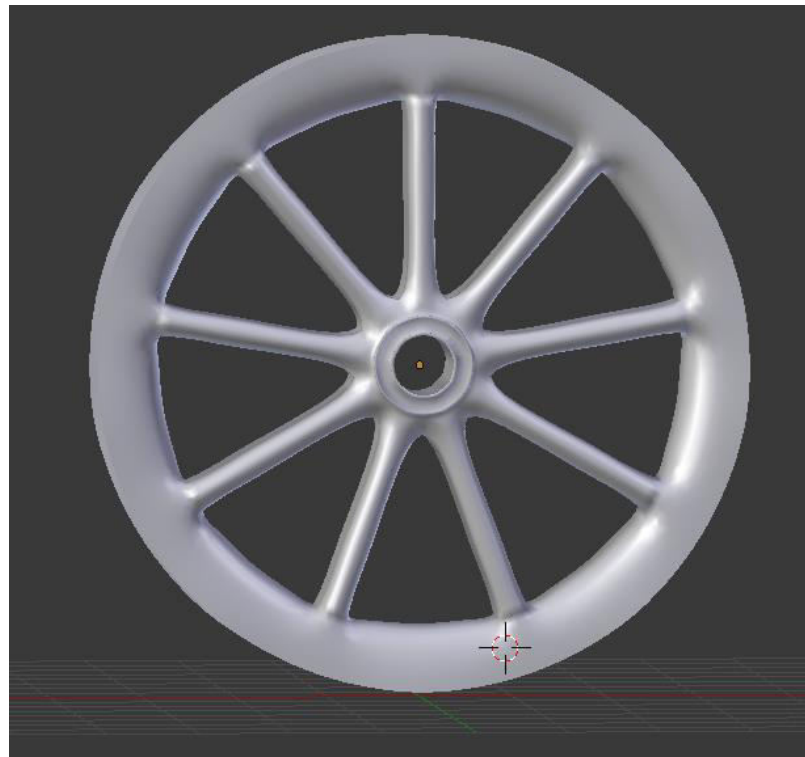
3.3.2.2. Desain

Desain dari aplikasi ini menjelaskan tentang objek yang menjadi objek 3 dimensi yang dibuat dengan aplikasi blender. Data yang di dapat berupa gambar aksesoris modifikasi. Contoh desain objek modifikasi dapat dilihat pada Gambar 3.5.



Gambar 3.5 Contoh Gambar Asli Velg

Dari Gambar 3.5 kemudian di visualisasikan kedalam objek 3 dimensi dengan menggunakan aplikasi blender.



Gambar 3.6 Hasil Modelling Objek

Setelah dibentuk objek 3 dimensi kemudian di beri tekstur agar warnanya sesuai dengan objek nyata.



Gambar 3.7 Objek 3 Dimensi Setelah *Texturing*

3.3.2.3. Material Collecting

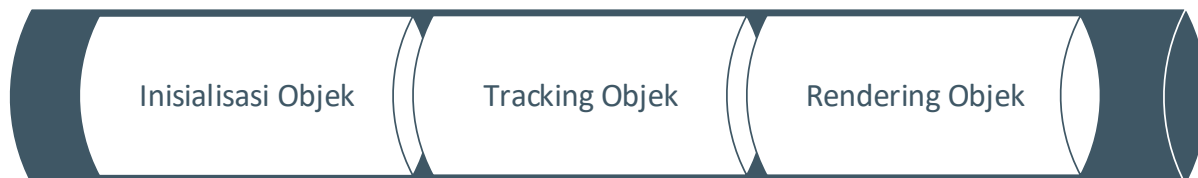
Analisis pengumpulan material membahas mengenai material atau bahan-bahan yang digunakan untuk aplikasi yang akan dibangun, serta sumber diperoleh. Material yang digunakan dapat dilihat pada Tabel 3.2 Pengumpulan Material.

Tabel 3.2 Pengumpulan Material

Material	Jenis File	Sumber
Velg	Objek 3D	Dibuat dengan menggunakan blender
Box	Objek 3D	Dibuat dengan menggunakan blender
Handle	Objek 3D	Dibuat dengan menggunakan blender
Shockbreaker	Objek 3D	Dibuat dengan menggunakan blender

3.3.3. Analisis Alur Sistem

Analisis alur sistem merupakan yang mendeskripsikan proses berjalannya sistem. Alur sistem dapat dilihat pada Gambar 3.8.



Gambar 3.8 Alur sistem

1. Inisialisasi Objek

Objek yang akan digunakan sistem akan di *load* terlebih dahulu saat awal sistem berjalan, hal ini dilakukan agar sistem tidak perlu memuat model setiap pengguna akan berganti objek.

2. Tracking Objek

Tracking objek adalah proses dimana aplikasi akan melakukan proses deteksi atau pelacakan sebuah marker dengan tujuan untuk menampilkan objek *Augmented Reality* berupa objek modifikasi 3 dimensi.

Objek yang dimunculkan akan bergerak mengikuti marker yang telah di tangkap oleh pengguna saat akan memunculkan objek modifikasi.

3. Rendering Objek

Hasil tangkapan marker oleh pengguna akan menjadi acuan untuk menampilkan objek *Augmented Reality* yang telah dipilih pengguna. Proses menampilkan objek *Augmented Reality* itu sendiri sudah di sediakan oleh Library Vuforia SDK.

3.4. Analisis Augmented Reality

Aplikasi yang dibangun menggunakan metode User Defined Target dimana artinya marker yang didapat adalah berdasarkan *input* dari pengguna. User Defined Target ini adalah turunan dari metode Image Target.

3.4.1. Image Target

Adalah dimana system mencari sebuah marker berupa gambar. Gambar yang akan dijadikan marker di tangkap oleh kamera yang ada pada perangkat pengguna.



Gambar 3.9 Image Target

3.4.2. Pemrosesan Image Target

Vuforia mencari *features* berdasarkan banyaknya *edge* pada gambar, semakin banyak *features* maka semakin bagus pula gambar tersebut menjadi *marker*. Berikut adalah proses pendeteksian *features* dari sebuah gambar.

1. Membangun *Scale Space*

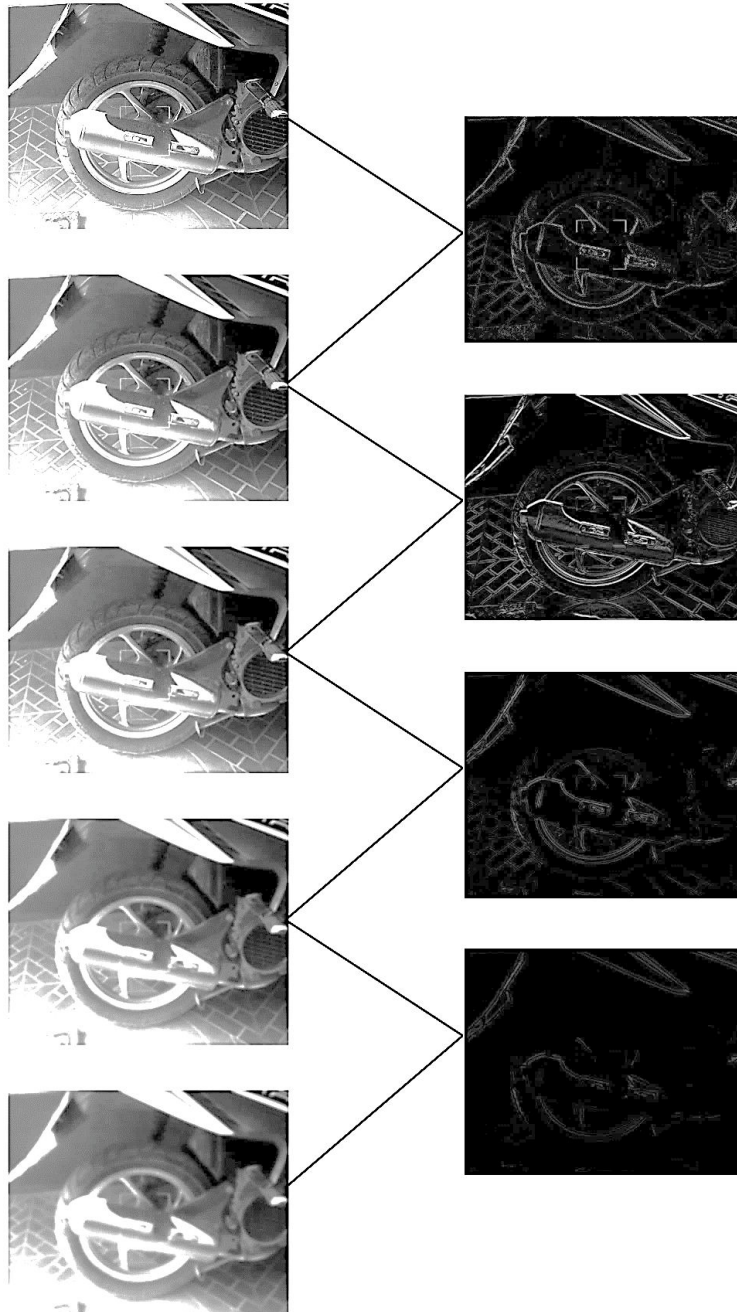
Sebagai tahap awal dari pencarian *features* dilakukan pembangunan *scale space* untuk mengeliminasi tepian palsu pada gambar dengan melakukan *Gaussian blur* pada gambar target.



Gambar 3.10 Pembangunan *Scale Space*

2. Perhitungan *Laplacian of Gaussian*

Proses ini dilakukan dengan cara men-*subtract* gambar yang telah di hasilkan oleh proses pembangunan *scale space* sehingga terbentuk tepian dari gambar target.

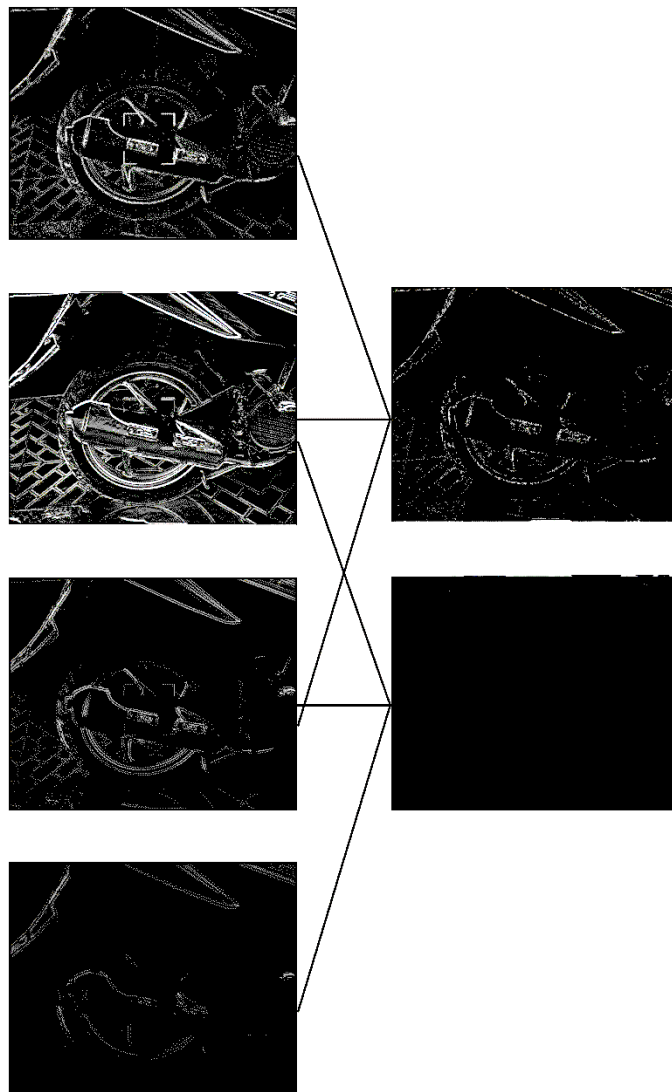


Gambar 3.11 Proses *Laplacian of Gaussian*

3. Mencari *keypoints*

Pencarian *keypoints* ini dilakukan dengan menemukan *minimal/maxima* dari gambar target dengan cara membandingkan setiap nilai *pixel* pada gambar hasil perhitungan *Laplacian of Gaussian* dengan 8 *pixel* tetangganya pada skala yang sama dan 9 *pixel* di atas dan di bawah skala gambar. Kemudian ditemukan nilai *minima* yaitu nilai *pixel* yang terkecil dari sebuah iterasi dan nilai *maksima* yaitu nilai yang terbesar dari sebuah iterasi.

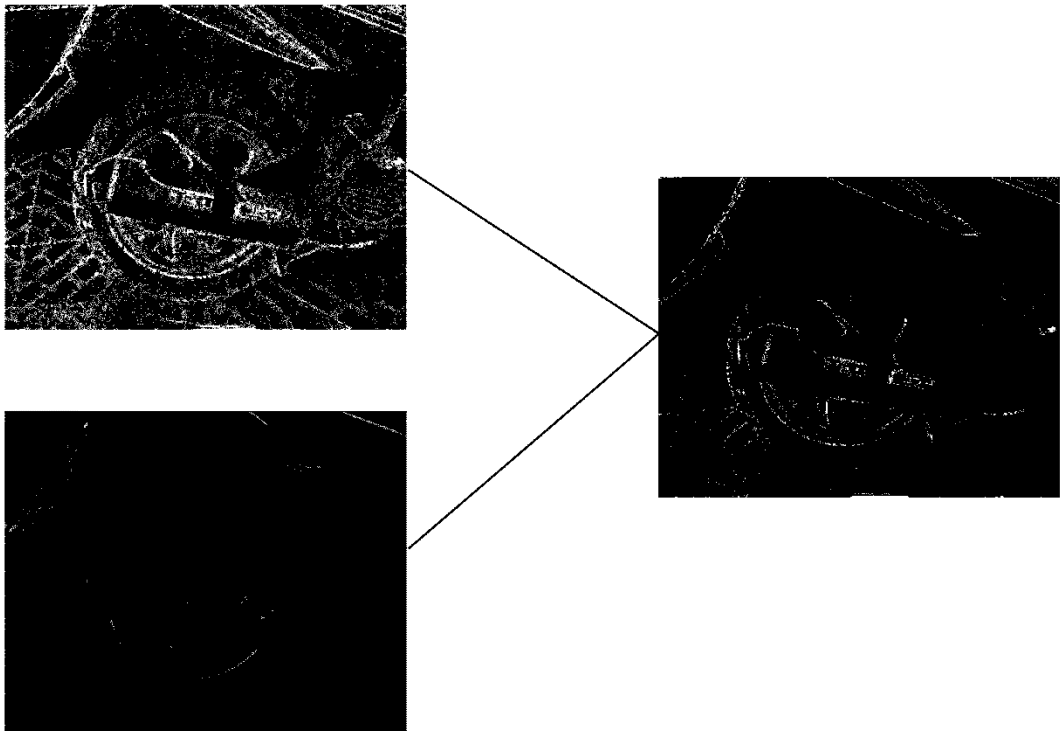
Keypoints adalah *pixel* yang berada di antara *maxima* dan *minima*.



Gambar 3.12 Hasil pencarian *keypoints*

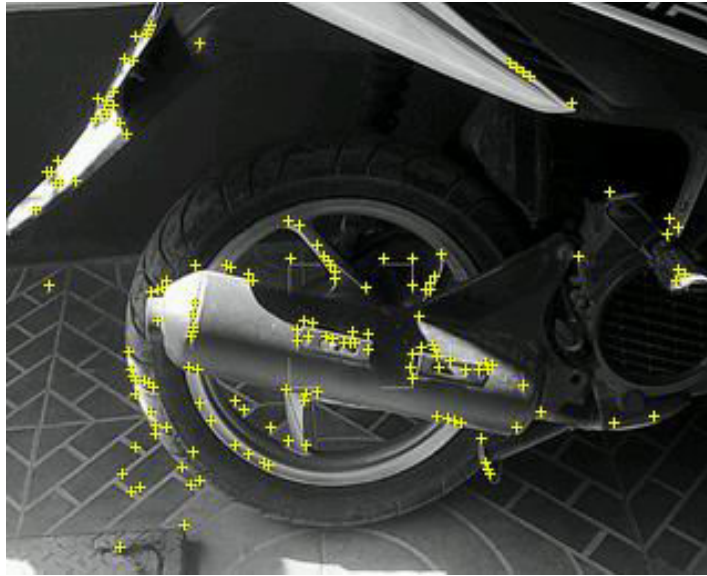
4. Mendapatkan *features*

Tahap ini dilakukan dengan cara membandingkan kontras gambar hasil pencarian *keypoints*. Jika ada *keypoints* dengan tingkat kontras kurang dari nilai tertentu maka *keypoints* tersebut akan dihilangkan. Hasilnya adalah *features* dari gambar target.



Gambar 3.13 Proses pencarian *features*

Jika titik-titik *features* yang telah di dapat di aplikasikan pada gambar target maka hasilnya dapat terlihat pada Gambar 3.14.



Gambar 3.14 *Features* pada gambar target

3.4.3. Menampilkan Objek

Setelah hasil tangkapan menjadi marker maka langkah selanjutnya adalah menampilkan objek yang menyesuaikan dengan *marker* yang telah di tangkap.



Gambar 3.15 Menampilkan Objek pada *Marker*

3.5. Analisis Kebutuhan Non Fungsional

Berikut ini adalah hasil analisis kebutuhan non fungsional pada perancangan Aplikasi desain sepeda motor menggunakan augmented reality. Spesifikasi kebutuhan non-fungsional dapat dilihat pada Tabel 3.3.

Tabel 3.3 Spesifikasi Kebutuhan Non-Fungsional

Nomor SKPL	Spesifikasi Kebutuhan non-Fungsional
SKPL-NF-01	Sistem yang dibangun menggunakan Android
SKPL-NF-02	Sistem yang dibangun harus memenuhi spesifikasi perangkat keras minimum
SKPL-NF-03	Sistem dibangun dengan Unity dan menggunakan Bahasa pemrograman C#

3.5.1. Analisis Perangkat Lunak

Perangkat lunak pengembang dapat dilihat di Tabel 3.4.

Tabel 3.4 Kebutuhan Perangkat Lunak Pengembang

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Microsoft Windows 10
2	Tool Pembangun	Unity 2017.3.1
3	Tool Desain	Blender 2.97

3.5.2. Analisis Perangkat Keras

Berikut ini adalah kebutuhan perangkat keras pengembang yang ditunjukkan oleh Tabel 3.5.

Tabel 3.5 Kebutuhan Perangkat Keras Pengembang

No	Perangkat Keras	Spesifikasi
Personal Computer		
1	Processor	AMD FX-6300 3.5GHz
2	Monitor	17"
3	Graphic Card	2 GB
4	Memori	DDR3 8GB
5	Harddisk Drive	Free Space 10GB
6	Webcam	0.3 MP
7	Keyboard dan Mouse	Standar
Smartphone Android		
1	Sistem Operasi	Android Nougat 7.1.2
2	Processor	8x ARM Cortex-A53 1.40GHz
3	Memori	RAM 3GB
4	Penyimpanan	Free Space 50 MB

Aplikasi yang akan di bangun mempunyai spesifikasi minimum untuk dijalankan. Kebutuhan perangkat keras pengguna dapat dilihat di Tabel 3.6.

Tabel 3.6 Kebutuhan Perangkat Keras Pengguna

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Android Jellybean 4.1
2	Processor	1 GHz
3	Memori	RAM 1GB
4	Penyimpanan	Free Space 50 MB
5	Camera	5MP

3.5.3. Analisis Kebutuhan Pengguna

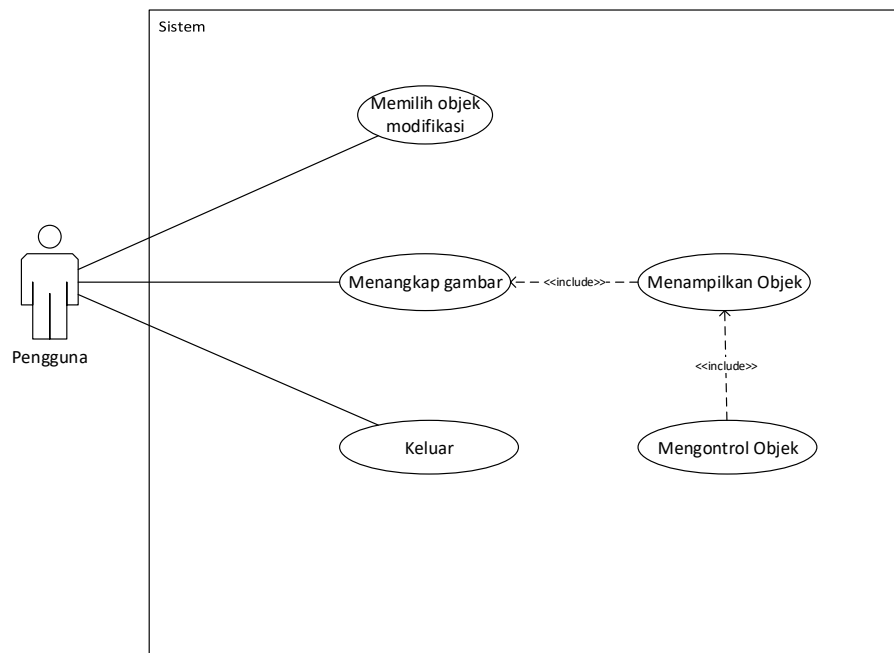
Kebutuhan dari penggunaan aplikasi desain sepeda motor dengan *augmented reality* ini antara lain pengguna mempunyai dan mengerti cara memasang aplikasi pada perangkat dengan spesifikasi minimum yang terdapat pada Tabel 3.6.

3.6. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional adalah analisis terhadap kebutuhan secara fungsional baik dalam aliran data ataupun informasi. Analisis kebutuhan fungsional digambarkan dalam analisis struktur yang akan digambarkan didalam sub bab berikutnya.

3.6.1. Diagram Use Case

Diagram *use-case* merupakan salah satu diagram untuk memodelkan aspek perilaku sistem. Masing-masing diagram *use-case* menunjukkan sekumpulan *use-case*, aktor dan hubungannya [13]. *Use case* dari aplikasi desain modifikasi sepeda motor menggunakan *Augmented Reality* dapat di lihat di Gambar 3.16.



Gambar 3.16 Diagram Use Case

3.6.1.1. Skenario Use Case

Skenario *use-case* merupakan narasi tentang aktivitas dalam suatu *use case* diagram pada sistem. Berikut proses – proses yang terdapat dalam skenario *use case*.

3.6.1.1.1. Skenario *Use Case* Memilih Objek Modifikasi

Berikut merupakan skenario *use-case* memilih objek modifikasi yang ditunjukkan oleh Tabel 3.7.

Tabel 3.7 Skenario *Use Case* Memilih Objek Modifikasi

Use-case name	Memilih Objek Modifikasi	
Related requirements	-	
Goal in context	Pengguna dapat memilih objek modifikasi	
Precondition	-	
Successful end condition	Pengguna dapat memilih objek modifikasi yang sesuai	
Fail end condition	Pengguna gagal memilih objek modifikasi yang sesuai	
Primary Actors	Pengguna	
Secondary Actors	Tidak ada	
Main flow	Step	Action
	1	Sistem menampilkan menu objek modifikasi
	2	Pengguna memilih objek modifikasi yang akan di tampilkan
	3	Sistem menampilkan pesan objek terpilih

3.6.1.1.2. Skenario *Use Case* Menangkap Gambar

Berikut merupakan skenario *use-case* menangkap gambar yang ditunjukkan oleh Tabel 3.8.

Tabel 3.8 Skenario *Use Case* Menangkap Gambar

Use-case name	Menangkap Gambar	
Related requirements	-	
Goal in context	Pegguna berhasil menangkap gambar untuk dijadikan marker	
Precondition	-	
Successful end condition	Pegguna dapat menangkap gambar dari kamera	
Fail end condition	Pegguna gagal menangkap gambar dari kamera	
Primary Actors	Pegguna	
Secondary Actors	Tidak ada	
Main flow	Step	Action
	1	Sistem membuka kamera
	2	Pegguna mengarahkan kamera ke target
	3	Pegguna menekan tombol tangkap gambar
	4	Sistem menampilkan pesan tangkapan berhasil
Extension	Step	Action
	4.1	Sistem menampilkan pesan tangkapan gagal

3.6.1.1.3. Skenario *Use Case* Menampilkan Objek

Berikut merupakan skenario *use-case* menampilkan objek yang ditunjukkan oleh Tabel 3.9 Skenario *Use Case* Menampilkan Objek.

Tabel 3.9 Skenario *Use Case* Menampilkan Objek

Use-case name	Menampilkan Objek	
Related requirements	-	
Goal in context	Pengguna berhasil memunculkan objek	
Precondition	Pengguna telah menangkap gambar untuk di jadikan marker	
Successful end condition	Sistem dapat memunculkan objek berdasarkan marker	
Fail end condition	Sistem tidak dapat memunculkan objek	
Primary Actors	Pengguna	
Secondary Actors	Tidak ada	
Main flow	Step	Action
	1	Pengguna mengarahkan kamera ke target
	2	Sistem mendeteksi marker
	3	Sistem memunculkan objek 3 dimensi

3.6.1.1.4. Skenario *Use Case* Mengontrol Objek

Berikut merupakan skenario *use-case* mengontrol objek yang ditunjukkan oleh Tabel 3.10 Skenario *Use Case* Mengontrol Objek.

Tabel 3.10 Skenario *Use Case* Mengontrol Objek

Use-case name	Mengontrol Objek	
Related requirements	-	
Goal in context	Pengguna berhasil mengontrol objek	
Precondition	Sistem telah memunculkan objek 3 dimensi	
Successful end condition	Pengguna dapat menggeser atau mengubah ukuran objek	
Fail end condition	-	
Primary Actors	Pengguna	
Secondary Actors	Tidak ada	
Main flow	Step	Action
	1	Pengguna menggeserkan objek dengan input sentuhan
	2	Sistem mengontrol objek berdasarkan input sentuhan pengguna

3.6.1.1.5. Skenario *Use Case* Keluar

Berikut merupakan skenario *use-case* keluar yang ditunjukkan oleh Tabel 3.11 Skenario *Use Case* Keluar.

Tabel 3.11 Skenario *Use Case* Keluar

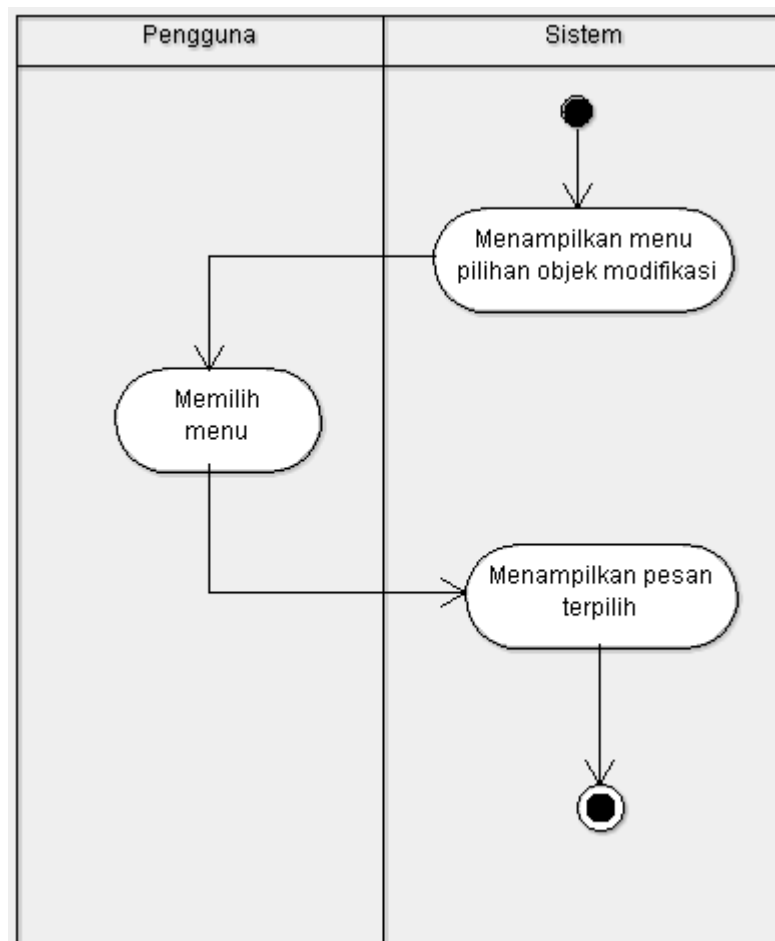
Use-case name	Keluar	
Related requirements	-	
Goal in context	Pengguna berhasil keluar dari aplikasi	
Precondition	-	
Successful end condition	Sistem menghentikan aplikasi	
Fail end condition	-	
Primary Actors	Pengguna	
Secondary Actors	Tidak ada	
Main flow	Step	Action
	1	Pengguna menekan tombol keluar
	2	Sistem menghentikan aplikasi

3.6.2. Activity Diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis dan menu yang ada pada perangkat lunak.

3.6.2.1. Activity Diagram Memilih Objek Modifikasi

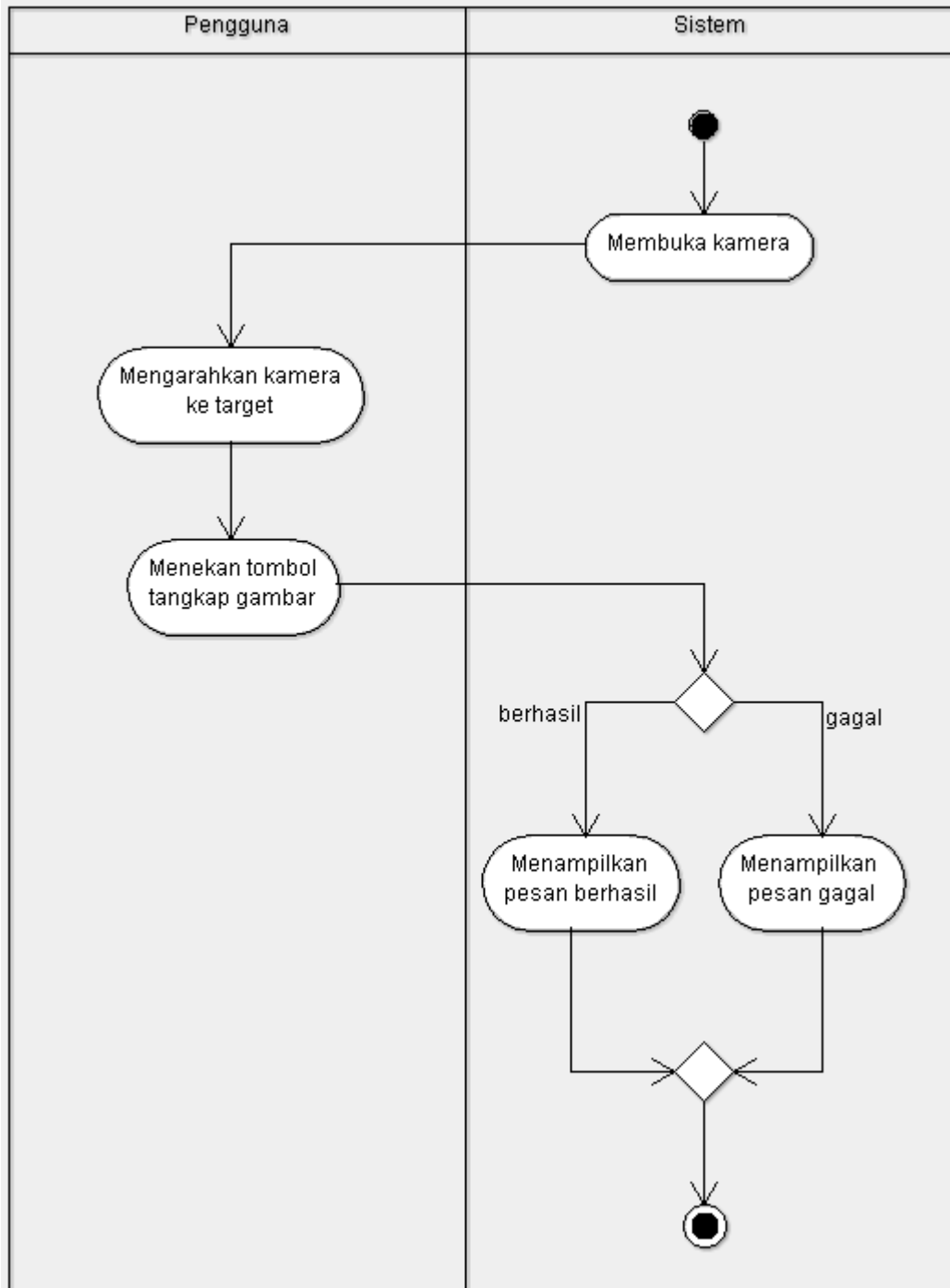
Berikut merupakan *activity diagram* dari *use case* memilih objek modifikasi ditunjukkan oleh Gambar 3.17.



Gambar 3.17 Activity Diagram Memilih Objek Modifikasi

3.6.2.2. Activity Diagram Menangkap Gambar

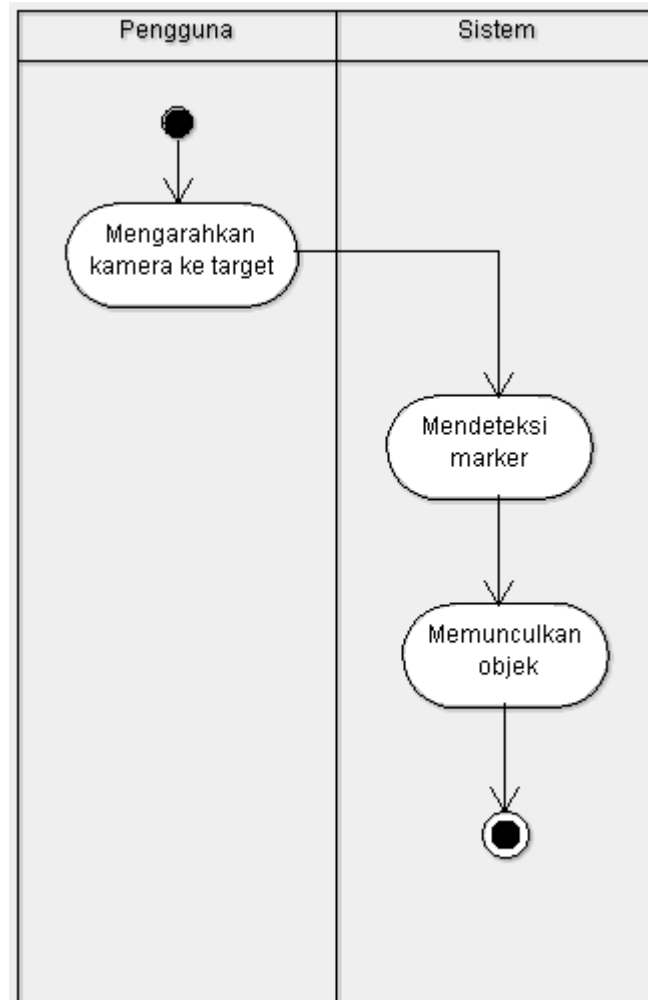
Berikut merupakan *activity diagram* dari *use case* menangkap gambar ditunjukkan oleh Gambar 3.18.



Gambar 3.18 Activity Diagram Menangkap Gambar

3.6.2.3. Activity Diagram Menampilkan Objek

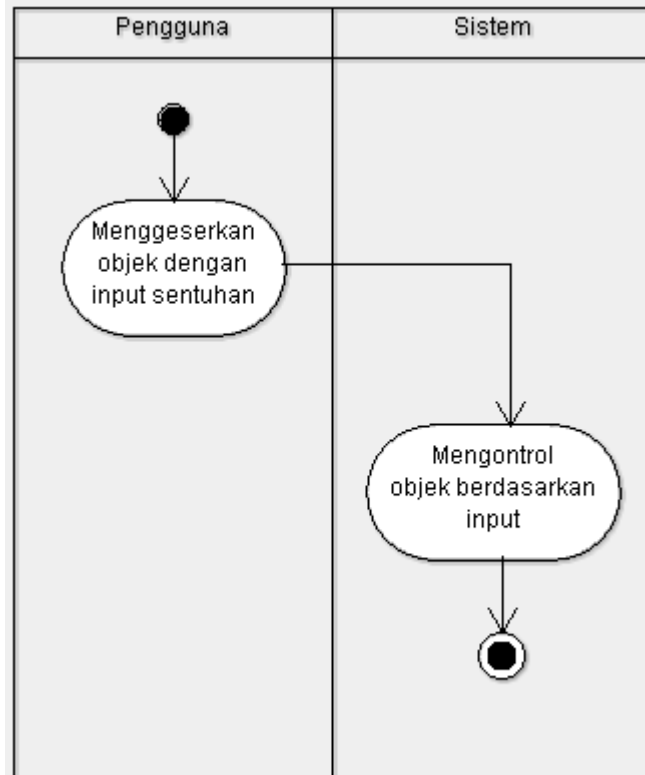
Berikut merupakan *activity diagram* dari *use case* menampilkan objek ditunjukkan oleh Gambar 3.19.



Gambar 3.19 Activity Diagram Menampilkan Objek

3.6.2.4. Activity Diagram Mengontrol Objek

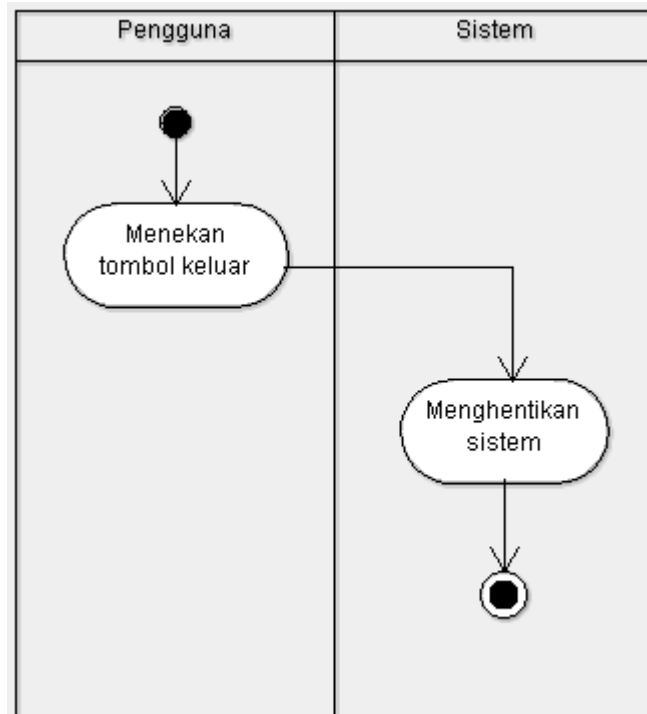
Berikut merupakan *activity diagram* dari *use case* mengontrol objek ditunjukkan oleh Gambar 3.20.



Gambar 3.20 Activity Diagram Mengontrol Objek

3.6.2.5. Activity Diagram Keluar

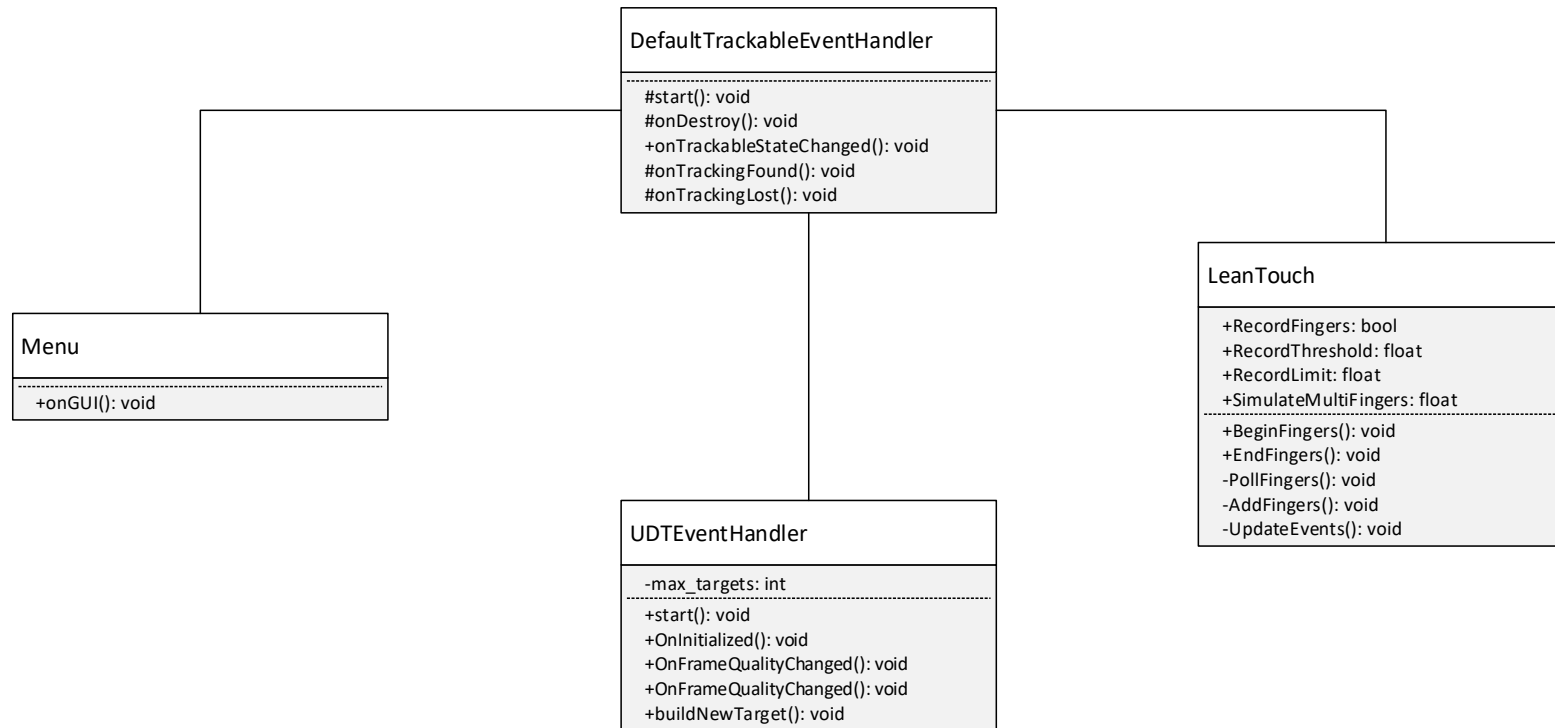
Berikut merupakan *activity diagram* dari *use case* keluar ditunjukkan oleh Gambar 3.21.



Gambar 3.21 Activity Diagram Keluar

3.6.3. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang dibuat untuk membangun sistem. Berikut adalah *class diagram* aplikasi desain modifikasi sepeda motor ditunjukkan oleh Gambar 3.22



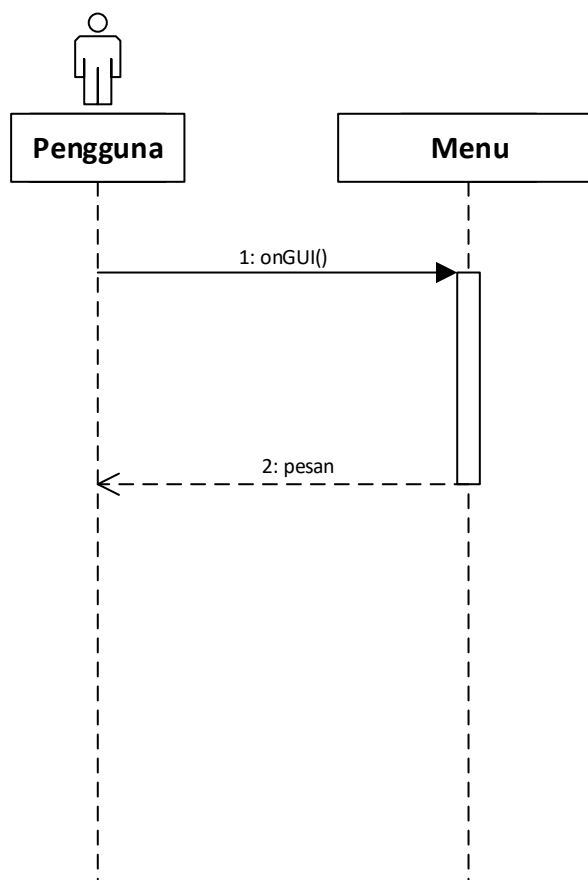
Gambar 3.22 Class Diagram

3.6.4. Sequence Diagram

Sebuah *sequence diagram*, diagram, secara khusus, menjabarkan behavior sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek-objek ini didalam *use case*.

3.6.4.1. Sequence Diagram Memilih Objek Modifikasi

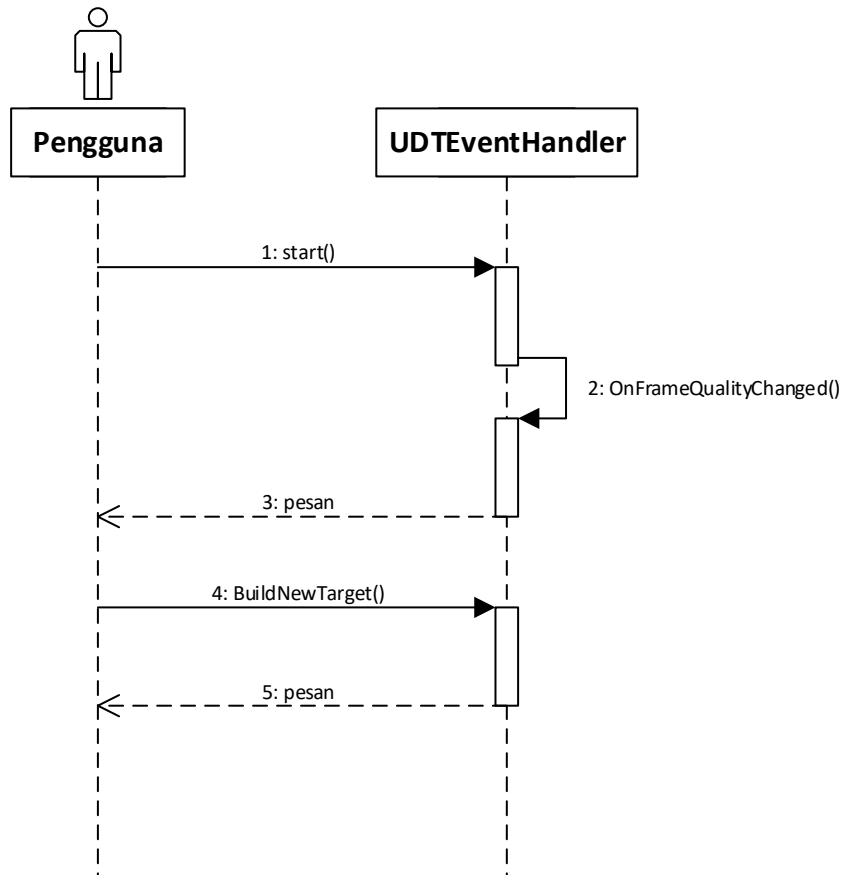
Berikut ini merupakan *sequence diagram* dari *use case* memilih objek yang ditampilkan pada Gambar 3.23.



Gambar 3.23 Sequence Diagram Memilih Objek Modifikasi

3.6.4.2. Sequence Diagram Menangkap Gambar

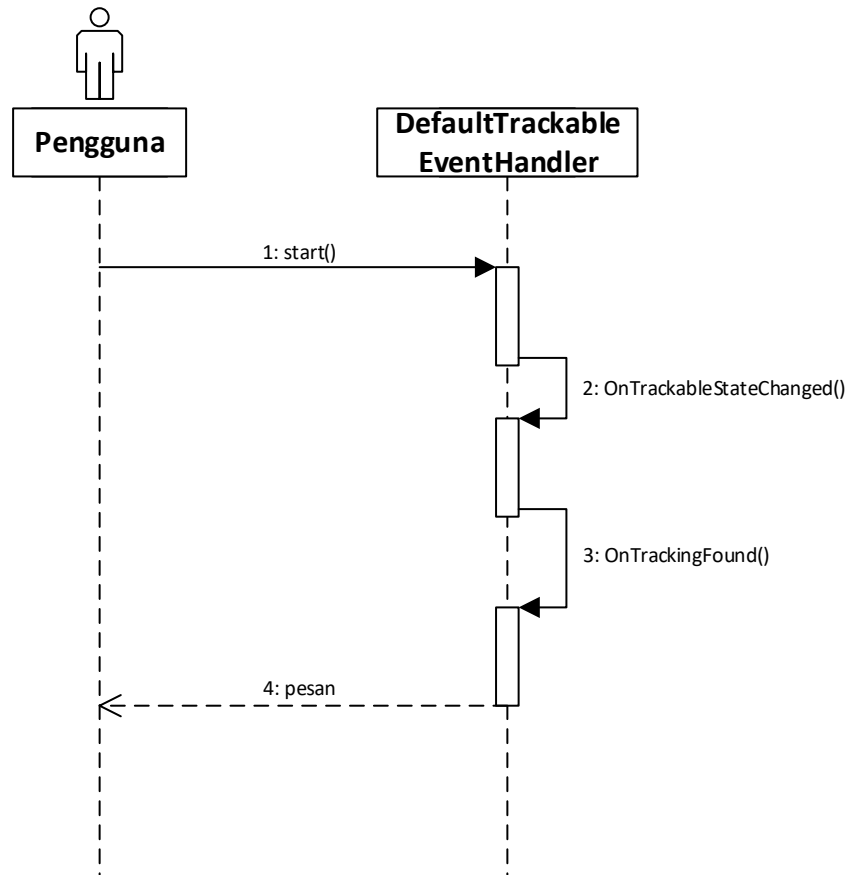
Berikut ini merupakan *sequence diagram* dari *use case* menangkap gambar yang ditampilkan pada Gambar 3.24.



Gambar 3.24 Sequence Diagram Menangkap Gambar

3.6.4.3. Sequence Diagram Menampilkan Objek

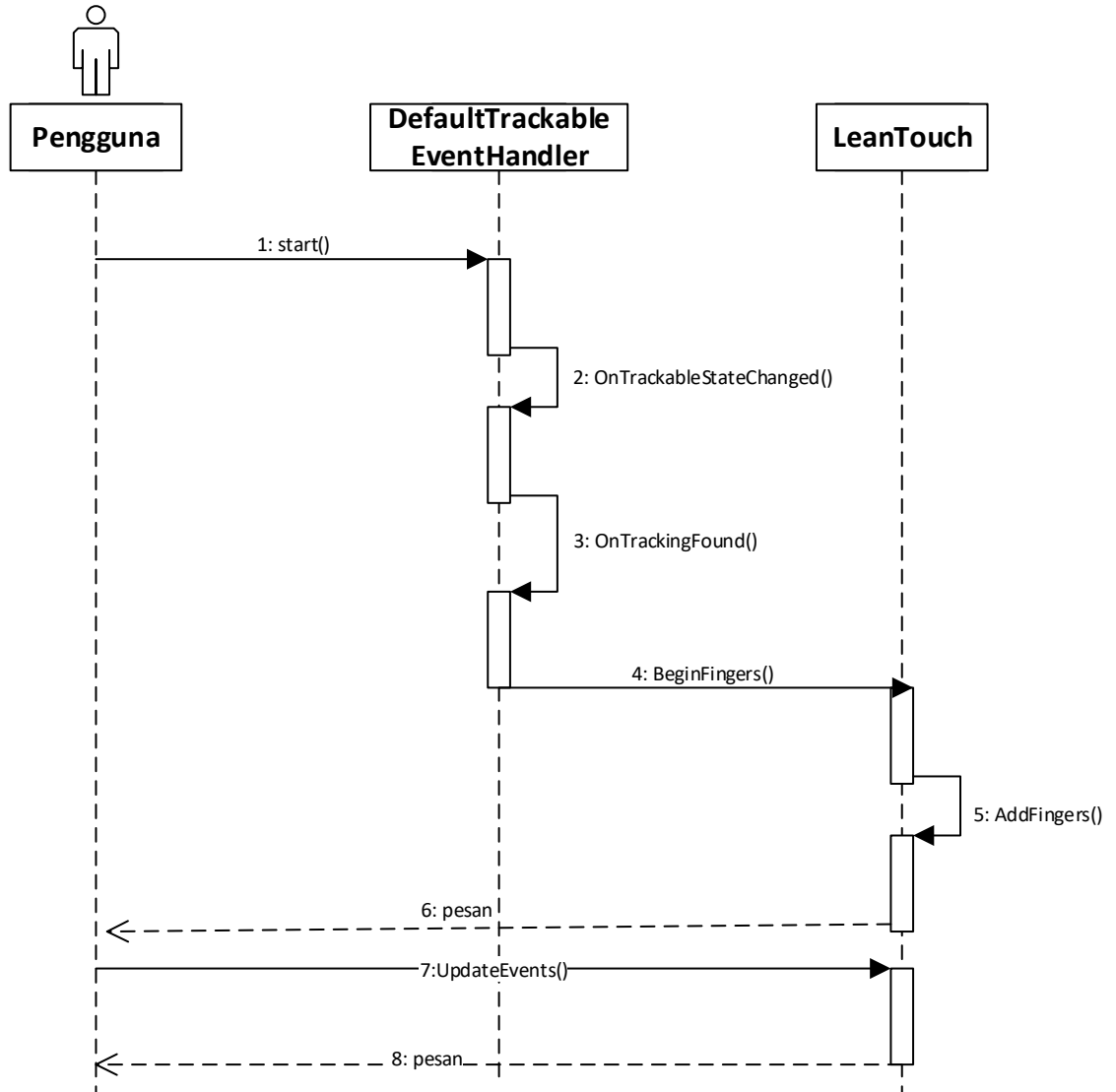
Berikut ini merupakan *sequence diagram* dari *use case* menampilkan objek yang ditampilkan pada Gambar 3.25.



Gambar 3.25 Sequence Diagram Menampilkan Objek

3.6.4.4. Sequence Diagram Mengontrol Objek

Berikut ini merupakan *sequence diagram* dari *use case* menampilkan objek yang ditampilkan pada Gambar 3.26



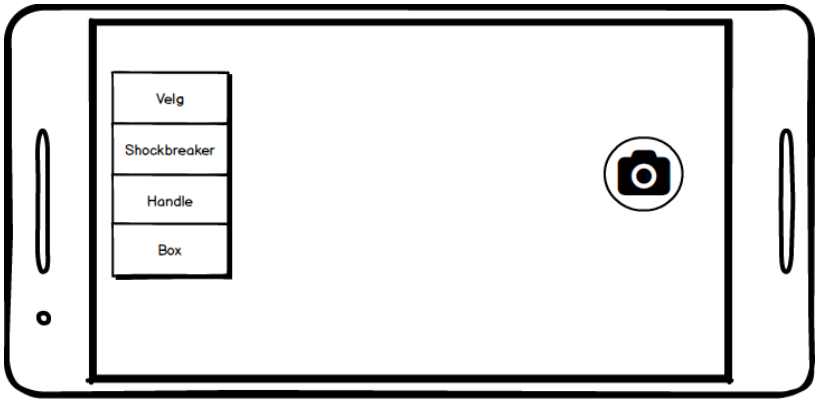
Gambar 3.26 Sequence Diagram Mengontrol Objek

3.6.5. Perancangan Antar Muka

Spesifikasi antarmuka merupakan suatu bentuk tampilan dari program yang akan dibuat untuk kebutuhan antarmuka dengan pengguna.

3.6.5.1. Perancangan Antar Muka Utama

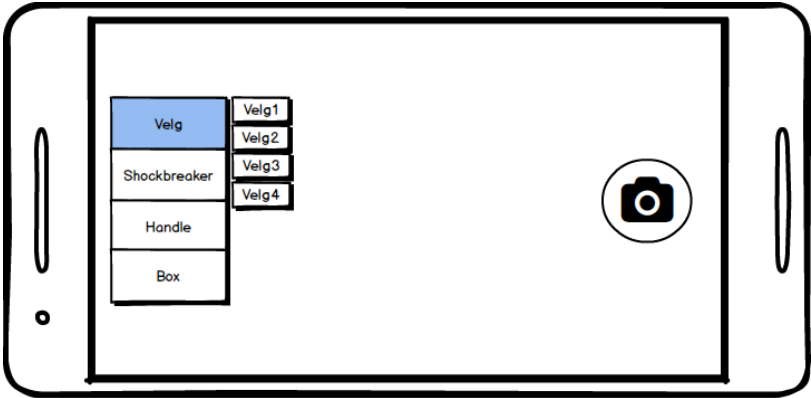
Berikut adalah antar muka menu utama yang ditunjukkan oleh Gambar 3.27.

T01	Antarmuka Utama	
		<ul style="list-style-type: none"> -Tekan tombol velg untuk menuju ke T02 -Tekan tombol shockbreaker untuk menuju ke T03 -Tekan tombol Box untuk menuju ke T04
<p>Ukuran resolusi: 854x480</p> <p>Font: Calibri 12 pt</p> <p>Warna disesuaikan</p>		<ul style="list-style-type: none"> -Tekan tombol handle untuk menuju ke T05 - Tekan tombol kamera untuk menampilkan pesan M17 atau M18

Gambar 3.27 Antar Muka Utama

3.6.5.2. Perancangan Antar Muka Menu *Velg*

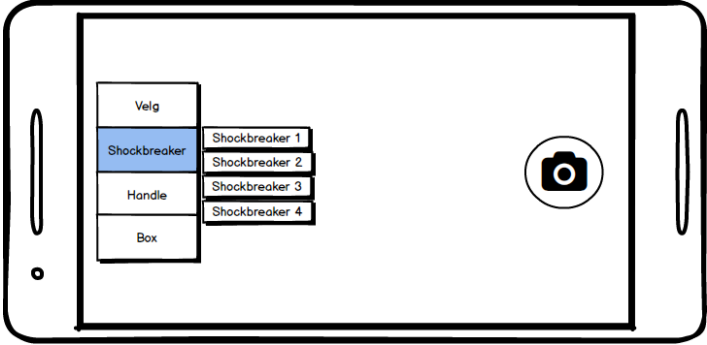
Berikut adalah antar muka menu *velg* yang ditunjukkan oleh Gambar 3.28.

T02	Antarmuka Menu <i>Velg</i>
	<ul style="list-style-type: none"> -Tekan tombol velg untuk menuju ke T01 -Tekan tombol shockbreaker untuk menuju ke T03 -Tekan tombol Box untuk menuju ke T04 -Tekan tombol handle untuk menuju ke T05 -Tekan tombol velg 1 untuk menampilkan pesan M01 -Tekan tombol velg 2 untuk menampilkan pesan M02 -Tekan tombol velg 3 untuk menampilkan pesan M03 -Tekan tombol velg 4 untuk menampilkan pesan M04 -Tekan tombol kamera untuk menampilkan pesan M17 atau M18
<p>Ukuran resolusi: 854x480</p> <p>Font: Calibri 12 pt</p> <p>Warna disesuaikan</p>	

Gambar 3.28 Antar Muka Menu *Velg*

3.6.5.3. Perancangan Antar Muka Menu *Shockbreaker*

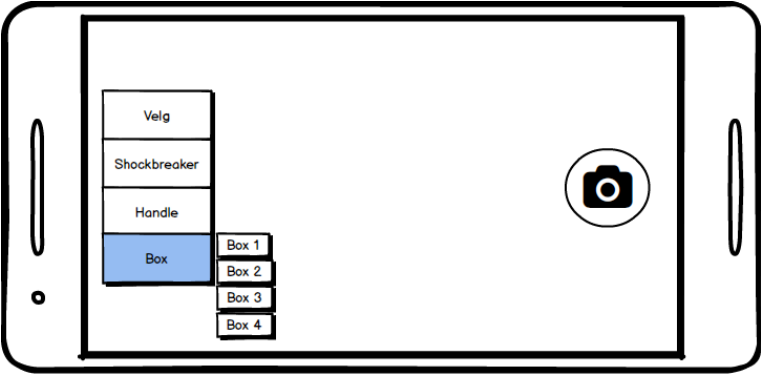
Berikut adalah antar muka menu *shockbreaker* yang ditunjukkan oleh Gambar 3.29.

T03	Antarmuka Menu Shockbreaker	
		<ul style="list-style-type: none"> -Tekan tombol velg untuk menuju ke T02 -Tekan tombol shockbreaker untuk menuju ke T01 -Tekan tombol Box untuk menuju ke T04 -Tekan tombol handle untuk menuju ke T05 -Tekan tombol shockbreaker 1 untuk menampilkan pesan M05 -Tekan tombol shockbreaker 2 untuk menampilkan pesan M06 -Tekan tombol shockbreaker 3 untuk menampilkan pesan M07 -Tekan tombol shockbreaker 4 untuk menampilkan pesan M08 - Tekan tombol kamera untuk menampilkan pesan M17 atau M18
<p>Ukuran resolusi: 854x480 Font: Calibri 12 pt Warna disesuaikan</p>		

Gambar 3.29 Antar Muka Menu *Shockbreaker*

3.6.5.4. Perancangan Antar Menu *Box*

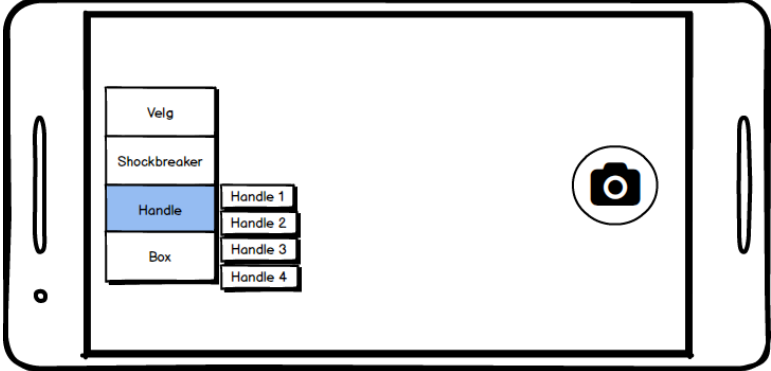
Berikut adalah antar muka menu *box* yang ditunjukkan oleh Gambar 3.30.

T04	Antarmuka Menu <i>Box</i>
 <p>The diagram shows a mobile phone screen with a menu on the left side. The menu items are: Velg, Shockbreaker, Handle, Box (highlighted in blue), Box 1, Box 2, Box 3, and Box 4. On the right side of the screen, there is a camera icon. The phone has a home button at the bottom and volume buttons on the sides.</p>	<ul style="list-style-type: none"> -Tekan tombol velg untuk menuju ke T02 -Tekan tombol shockbreaker untuk menuju ke T03 -Tekan tombol Box untuk menuju ke T01 -Tekan tombol handle untuk menuju ke T05 -Tekan tombol Box 1 untuk menampilkan pesan M09 -Tekan tombol Box 2 untuk menampilkan pesan M10 -Tekan tombol Box 3 untuk menampilkan pesan M11 -Tekan tombol Box 4 untuk menampilkan pesan M12 - Tekan tombol kamera untuk menampilkan pesan M17 atau M18
<p>Ukuran resolusi: 854x480 Font: Calibri 12 pt Warna disesuaikan</p>	

Gambar 3.30 Antar Muka Menu *Box*

3.6.5.5. Perancangan Antar Muka *Handle*

Berikut adalah antar muka menu *handle* yang ditunjukkan oleh Gambar 3.30.

T05	Antarmuka Menu Handle	
		<ul style="list-style-type: none"> -Tekan tombol velg untuk menuju ke T02 -Tekan tombol shockbreaker untuk menuju ke T03 -Tekan tombol Box untuk menuju ke T04 -Tekan tombol handle untuk menuju ke T01 -Tekan tombol handle 1 untuk menampilkan pesan M13 -Tekan tombol handle 2 untuk menampilkan pesan M14 -Tekan tombol handle 3 untuk menampilkan pesan M15 -Tekan tombol handle 4 untuk menampilkan pesan M16 -Tekan tombol kamera untuk menampilkan pesan M17 atau M18
<p>Ukuran resolusi: 854x480 Font: Calibri 12 pt Warna disesuaikan</p>		

Gambar 3.31 Antar Muka Menu *Handle*

3.6.5.6. Perancangan Pesan

Berikut adalah perancangan pesan yang ditunjukkan oleh Tabel

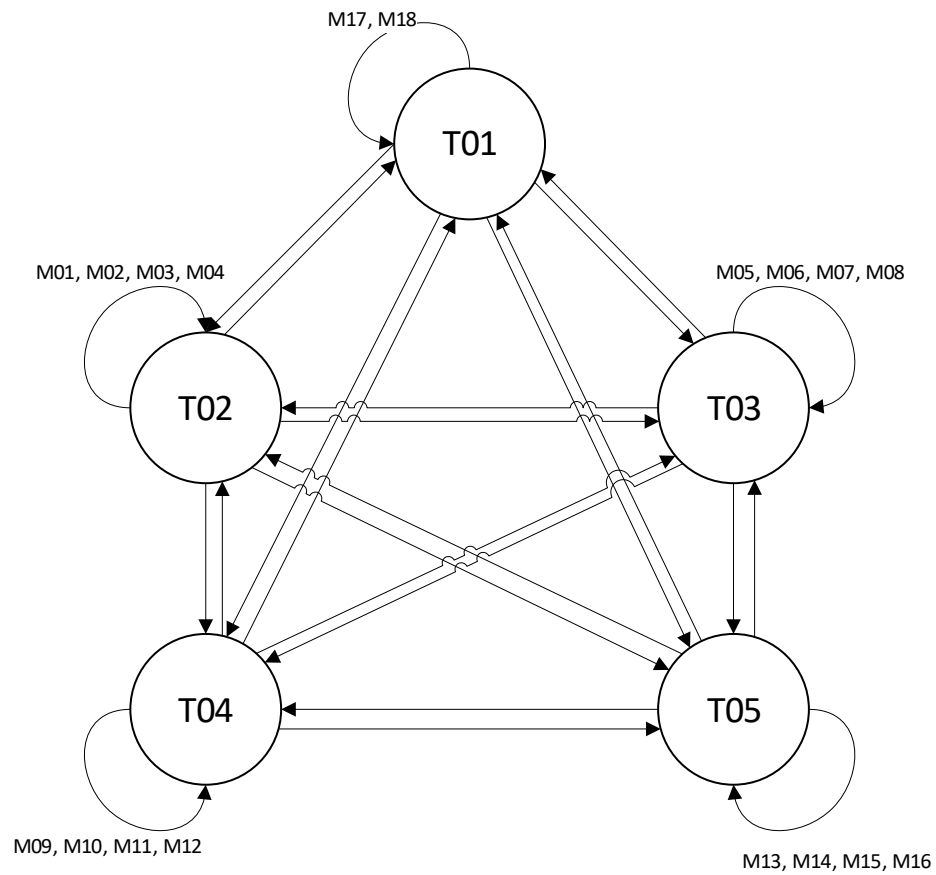
3.12

Tabel 3.12 Perancangan Pesan

No. Pesan	Isi Pesan	Terdapat Pada Form
M01	Anda memilih velg 1	T02
M02	Anda memilih velg 2	T02
M03	Anda memilih velg 3	T02
M04	Anda memilih velg 4	T02
M05	Anda memilih shockbreaker 1	T03
M06	Anda memilih shockbreaker 2	T03
M07	Anda memilih shockbreaker 3	T03
M08	Anda memilih shockbreaker 4	T03
M09	Anda memilih box 1	T04
M10	Anda memilih box 2	T04
M11	Anda memilih box 3	T04
M12	Anda memilih box 4	T04
M13	Anda memilih handle 1	T05
M14	Anda memilih handle 2	T05
M15	Anda memilih handle 3	T05
M16	Anda memilih handle 4	T05
M17	Kualitas gambar rendah	T01,T02,T03,T04,T05
M18	Objek berhasil ditampilkan	T01,T02,T03,T04,T05

3.6.6. Jaringan Semantik

Jaringan semantic dari perancangan antar muka dapat dilihat pada Gambar 3.32.



Gambar 3.32 Jaringan Semantik