

BAB 3

ANALISIS DAN PERANCANGAN

3. 1. Analisis Sistem

Analisis sistem merupakan penguraian dari sistem utama ke dalam sub-sub sistem dengan tujuan untuk mengidentifikasi dan mengevaluasi segala permasalahan dan hambatan yang terjadi serta kebutuhan yang diharapkan dapat menjadi acuan untuk diusulkannya perbaikan-perbaikan.

Tahap analisis sistem ini sangat penting karena apabila terjadi kesalahan dalam tahap ini akan mengakibatkan kesalahan pada tahap selanjutnya, untuk itu diperlukan tingkat ketelitian dan kecermatan yang tinggi untuk dapat mendapatkan kualitas kerja sistem yang baik. Untuk dapat melakukan analisis sistem maka dilakukan beberapa buah langkah yaitu sebagai berikut:

1. Analisis masalah
2. Analisis Materi
3. Analisis sistem yang sedang berjalan
4. Analisis aplikasi sejenis
5. Analisis arsitektur sistem
6. Analisis aplikasi yang dibangun
7. Analisis Alur Sistem
8. Analisis Marker
9. Analisis Metode
10. Analisis Kebutuhan Non Fungsional
11. Analisis kebutuhan fungsional

3.1.1. Analisis Masalah

Analisis masalah adalah sebuah proses untuk memecahkan sesuatu ke dalam bagian-bagian yang saling berkaitan satu sama lainnya. Analisis masalah akan menjelaskan tentang masalah yang terjadi pada proses pembelajaran *Chord* gitar sebelum melakukan pembangunan aplikasi multimedia.

Analisis masalah yang telah dilakukan terdapat beberapa masalah, yaitu:

1. Masih banyak orang yang ingin belajar alat musik gitar yang masih belum mampu untuk menerapkan metode pembelajaran *chord* gitar menggunakan buku-buku musik maupun video tutorial gitar, oleh sebab itu maka dirancang sebuah aplikasi sebagai media yang akan dengan mudah dan interaktif untuk digunakan.
2. Metode belajar menggunakan gambar *chord* dianggap kurang menarik sehingga metode tersebut cukup membosankan, sehingga dirancang sebuah model 3D dengan animasi agar memberi pengalaman baru bagi pengguna yang akan belajar *chord* gitar
3. Penempatan dan posisi jari meneka senar gitar tidak ada dalam gambar *chord* bahkan dalam video yang ada hanya terlihat dari satu arah sehingga bagi pemula akan begitu sulit untuk mengimplementasikannya, oleh karena itu pada sistem yang akan dibangun, dirancang model 3D yang dapat dilihat dari sudut yang berbeda, dan juga fitur suara yang akan mengarahkan pengguna dan memberi informasi penggunaan jari dan posisi *fret* pada sebuah *chord* gitar yang ditampilkan.

3.1.2. Analisis Materi

Analisis Materi akan menjelaskan mengenai materi pembelajaran chord gitar yang diambil dari referensi buku dengan judul buku “Trik Cepat Belajar Gitar Bolong tanpa Guru” yang ditulis oleh Adi Jarot Pamungkas. Materi yang diaplikasikan dalam sistem yang dibangun adalah sebagai berikut :

Tabel 3.1 Tabel Analisis materi

Materi	Tujuan yang ingin dicapai	Penerapan pada aplikasi
Bagian bagian gitar	<i>User</i> mengenali bagian-bagian apa saja yang terdapat pada sebuah gitar akustik serta fungsinya.	Gitar akan dimodelkan dalam bentuk 2D yang dilengkapi dengan nama dan penjelasan bagian tersebut.
Teori Musik	1. <i>User</i> mengenali <i>chord</i> mayor dan minor	1. Dalam aplikasi yang akan dibangun akan

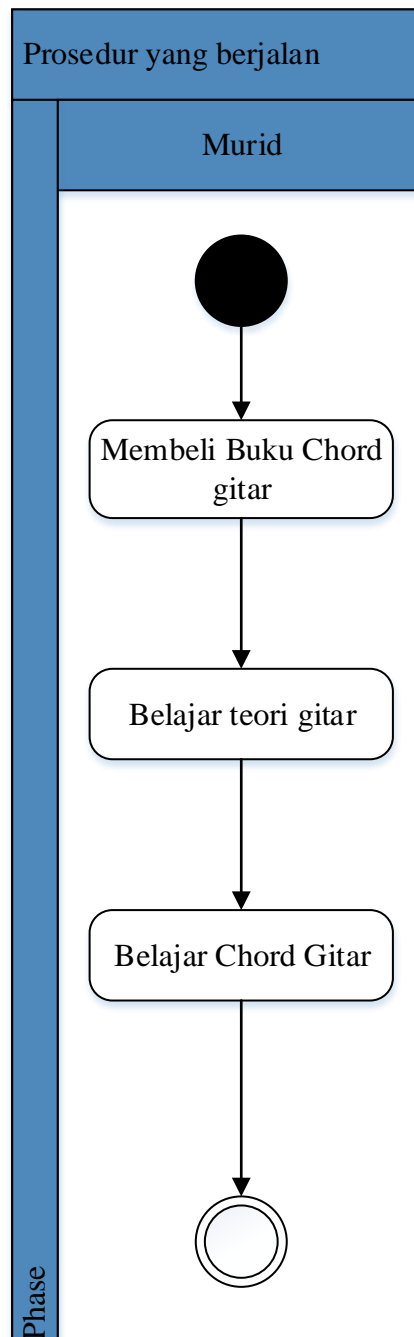
	<p>beserta bentuk <i>chord</i> tersebut</p> <p>2. <i>User</i> mengetahui posisi sebuah <i>chord</i> terletak pada senar dan <i>fret</i> beberapa</p>	<p>disertai dengan gambar <i>chord</i> dalam bentuk tabulasi gitar</p> <p>2. Pada gambar <i>chord</i> yang dibuat akan disertai dengan angka penunjuk posisi <i>chord</i></p>
Teknik Dasar	User mengetahui cara penempatan jari untuk menekan sebuah <i>chord</i> , posisi pada <i>fret</i> dan ditekan dengan pada senar beberapa	Untuk mempermudah dalam penerapan sebuah <i>chord</i> maka akan dibangun model 3D dengan animasi dan suara yang akan mengarahkan pengguna dalam belajar <i>chord</i> gitar

3.1.3. Analisis Sistem yang berjalan

Analisis sistem yang sedang berjalan akan menjelaskan mengenai langkah-langkah kerja dan dokumen-dokumen yang berjalan dalam langkah-langkah tersebut yang bertujuan untuk mengetahui lebih jelas mengenai bagaimana cara kerja prosedur yang sudah berjalan tersebut, sehingga dapat diketahui kekurangan dan kelebihan yang terdapat pada prosedur yang sudah berjalan.

Untuk belajar *chord* gitar maka orang yang akan belajar harus melakukan tahapan sebagai berikut :

1. Membeli buku Pelajaran musik secara khusus alat musik Gitar
2. Siswa harus memulai Belajar dari dasar musik terlebih dahulu sebelum memasuki teori *chord* gitar
3. Belajar *chord* gitar



Gambar 3.1 Prosedur Pembelajaran Chord Gitar

3.1.4. Analisis sistem atau Aplikasi sejenis

Analisis sistem atau aplikasi sejenis merupakan analisis yang akan membahas mengenai sistem atau aplikasi yang menjadi acuan dalam pembangunan sebuah aplikasi. Dalam aplikasi sejenis akan dibahas mengenai berbagai hal yang ada didalam aplikasi sejenis diantaranya konten, cara menggunakan dan komponen-

komponen apa saja yang ada di dalam aplikasi tersebut sehingga dapat menjadi acuan bagi kebutuhan yang akan dibangun.

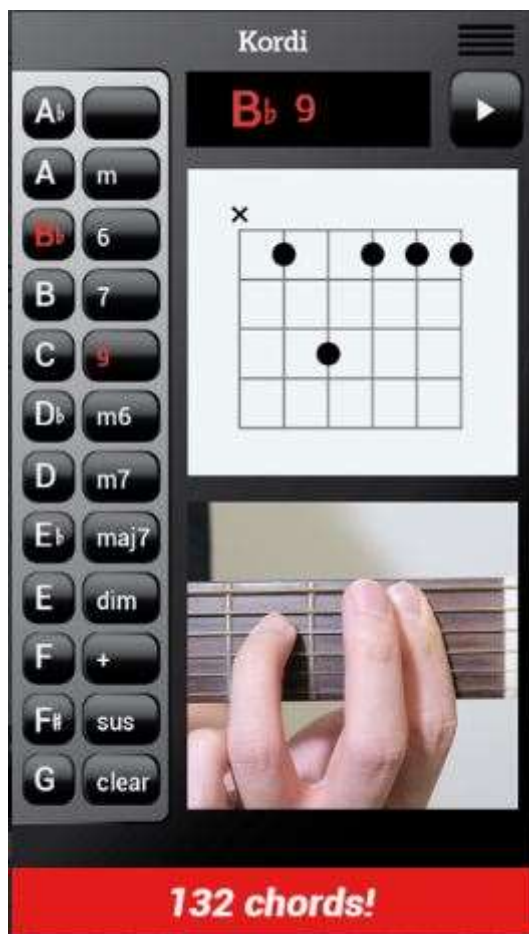
Tabel 3.2 Perbandingan aplikasi sejenis dengan aplikasi yang dibangun

Aplikasi Sejenis 1	Aplikasi Sejenis 2	Aplikasi yang akan dibangun
Kordi Gitar <i>Chord</i>	Kumpulan Kunci Gitar Indonesia	Perancangan <i>Augmented Reality</i> untuk Pembelajaran <i>Chord</i> Gitar berbasis Android
Kelebihan	Kelebihan	Kelebihan
<ol style="list-style-type: none"> 1. Memiliki <i>Bank Chord</i> yang banyak yaitu 132 <i>Chord</i> 2. Ada <i>chord</i> Mayor 7 dan 6 serta Halfdiminish (<i>chord jazz</i>) 	<ol style="list-style-type: none"> 1. Penerapan <i>chord</i> pada aplikasi langsung pada lagu 2. Ada <i>chord</i> 7 (<i>seven chord</i>) 3. Menyediakan beberapa tutorial <i>chord</i> gitar untuk beberapa genre lagu 	<ol style="list-style-type: none"> 1. Menampilkan <i>Chord</i> Mayor dan Minor 2. Visualisasi dalam bentuk 3D 3. Ada fitur Suara setiap <i>chord</i> yang akan dipelajari 4. Menampilkan teori singkat pembentukan <i>chord</i> gitar
Kekurangan	Kekurangan	Kekurangan
<ol style="list-style-type: none"> 1. Visualisai hanya dalam bentuk gambar tangan dan tabulasi. 	<ol style="list-style-type: none"> 1. Visualisai hanya gambar <i>chord</i> dalam bentuk tabel <i>chord</i> 	

1. Kordi Guitar *Chord*

Pada aplikasi ditampilkan sebanyak 132 Akord gitar yang terdiri dari Mayor, Minor, *Seven Chord*, Diminish dll. Pengguna dapat melihat beberapa bentuk bentuk chord gitar yang ingin dipelajari.

Berikut adalah tampilan Aplikasi Kordi Guitar *Chord*



Gambar 3.2 Tampilan Kordi Guitar *Chord*

2. Kumpulan Kunci Gitar Indonesia

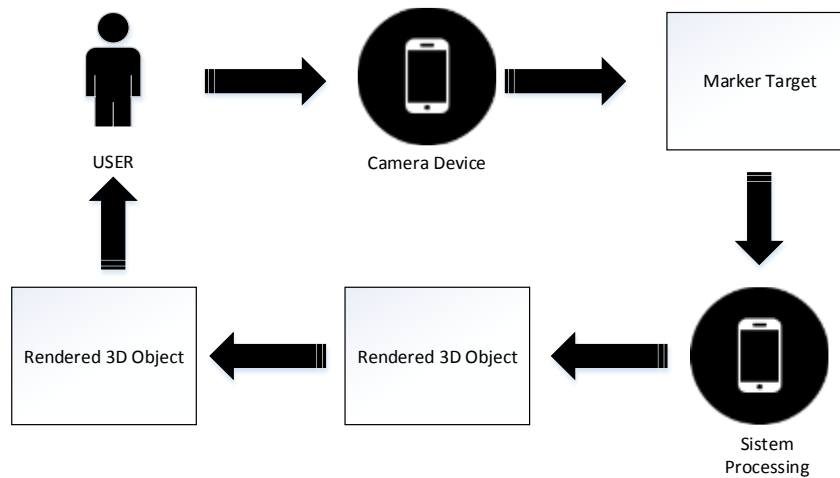
Pada aplikasi Kumpulan Kunci Gitar Indonesia ini tersedia beberapa gambar *offline* berupa tabel-tabel kunci gitar dimana penempatan nada-nada yang akan ditekan dengan jari digambarkan sebagai titik-titik pada garis-garis yang diibaratkan sebagai senar gitar, dan kolom-kolom atau *fret* gitar.



Gambar 3.3 Tampilan Chord pada aplikasi kumpulan kunci gitar Indonesia

3.1.5. Analisis Arsitektur Sistem

Arsitektur sistem bertujuan untuk mengidentifikasi yang akan dibangun pada aplikasi *Augmented Reality* untuk pembelajaran *Chord* gitar yang akan dibangun berbasis Android.



Gambar 3.4 Arsitektur Sistem

Aplikasi ini membantu untuk Pembelajaran *Chord* Gitar. Arsitektur sistem yang dibangun terdiri dari beberapa komponen yaitu :

1. *User* adalah pengguna yang akan menggunakan aplikasi, *user* mengarahkan kamera ke *marker*,
2. *Marker* yang digunakan berupa gambar yang dicetak dalam kartu ,
3. kemudian kamera akan melakukan *tracking* pada *marker* untuk mengidentifikasi *marker* yang digunakan oleh *user*.
4. Sistem akan melakukan render objek 3D diatas *marker* yang telah teridentifikasi.
5. Kemudian sistem akan menampilkan Objek 3D yang sudah di *render*, sehingga dapat dilihat oleh *user*

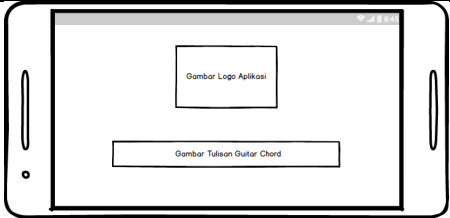
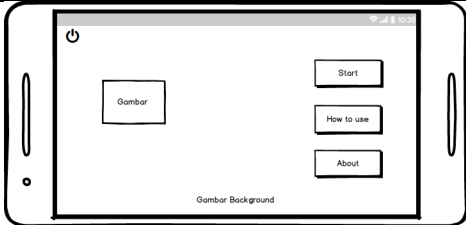
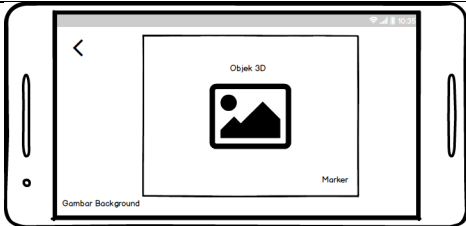
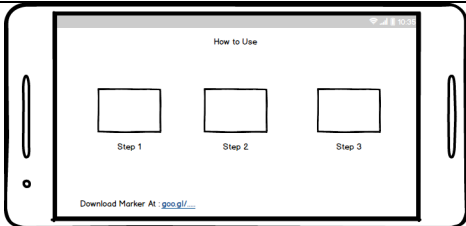
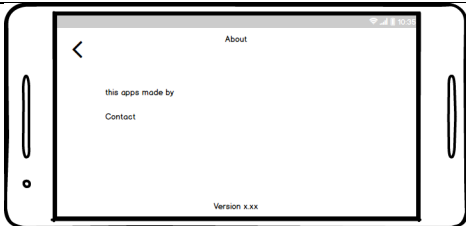
3.1.6. Analisis Aplikasi yang dibangun

Perancangan *Augmented Reality* untuk pembelajaran chord gitar memiliki tujuan yaitu untuk Membuat suatu simulasi multimedai 3D interaktif perakitan sepeda yang dapat membantu pengguna aplikasi untuk melakukan pembelajaran *chord* pada gitar sehingga pengguna dapat melihat gambaran secara rinci posisi dan penempatan jari dengan visual dari *chord* yang dipilih oleh pengguna.

3.1.6.1. Perancangan Storyboard Aplikasi

Storyboard pada pembangunan aplikasi ini akan dijelaskan pada tabel Dibawah:

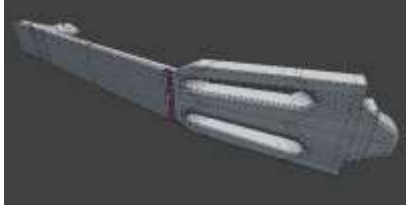





Tabel 3.3 Storyboard aplikasi

No.	Visual	Deskripsi	Scene
1		Menampilkan halaman Splash Screen Aplikasi GuitAR Guide	<i>Splash Screen</i>
2		Menampilkan Menu utama, yang terdiri dari menu <i>Start</i> , <i>How to use</i> dan <i>About</i>	Halaman utama
3		Menampilkan Halaman <i>Start</i> , dimana kamera diaktifkan dan gambar 3D muncul	Halaman <i>Start</i>
4		Menampilkan gambar cara penggunaan aplikasi.	Halaman <i>How to use</i>
5		Menampilkan halaman Tentang aplikasi	Halaman <i>About</i>

3.1.6.2. Pemodelan Objek 3D

Pembuatan model objek 3d menggunakan aplikasi yaitu *blender*. Objek yang dibuat sesuai dengan anatomi gitar yang telah dijelaskan. Objek 3d yang dibuat hanya sebatas modelnya saja untuk penggabungan objek bagian-bagian gitar dan texture sebagai representasi bahan yang digunakan.



1. Pembuatan *Freatboard* gitarTabel 3.4 Tabel bagian *Freatboard* gitar


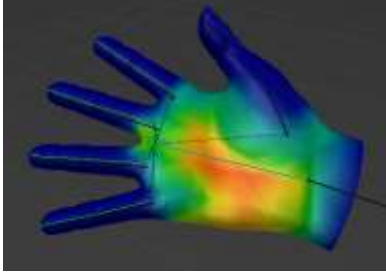
No.	Nama	Model 3D
1	Bagian Tangkai Utama Gitar <i>/ Fretboard</i>	
2	Bagian pembatas nada pada <i>Freatboard/ Fret</i>	
3	Bagian <i>Tuning Machines</i>	
4	Senar Gitar	
5	Bagian depan <i>Fretboard</i>	
6	Bagian belakang <i>Fretboard</i>	

Model 3D yang dibuat adalah bagian *Fretboard* gitar sampai ke *tuning machine*, yang terdiri dari pembuatan papan jari gitar, pembuatan pembatas 22 kolom *fret*, pembuatan senar gitar yang terdiri dari 6 tali dawai yang akan ditekan oleh jari sehingga menghasilkan nada, pembuatan *tuning machine* yang pada gitar berfungsi untuk menyetem senar gitar. Model gitar yang dibuat adalah model gitar nylon dengan 6 Senar. Standar *tuning* nada pada masing-masing senar yang akan dibuat adalah *chromatic E-A-D-G-B-E*.

2. Pembuatan Bagian Tangan

Tabel 3.5 Tabel bagian tangan

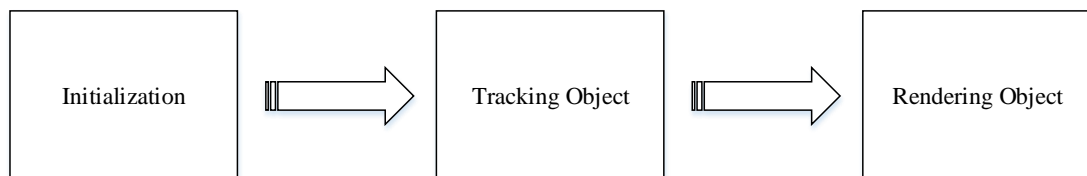
No.	Nama	Model 3D
1	<i>Shape</i> dasar tangan	
2	Memperhalus shape dasar menggunakan modivier subdivision surface	

3	Pembentukan tulang (<i>rigging</i>)	
4	menentukan titik berat dari setiap ruas tulang (<i>weight paint</i>)	

Pada bagian tangan, Model 3D yang dibuat adalah model 3D tangan kiri dengan jari-jari yang akan menekan senar gitar pada *fretboard*. Jari-jari yang akan menekan senar adalah jari kelingking sampai jari telunjuk.

3.1.7. Analisis Alur Sistem

Analisis alur sistem merupakan analisis yang mendeskripsikan bagaimana proses augmented reality dari awal inisialisasi, *tracking marker*, sampai dengan proses *rendering* objek 3D. Secara garis besarnya, dalam perancangan ada tiga bagian utama yaitu sebagai berikut



Gambar 3.5 Alur Sistem

Secara garis besarnya, dalam perancangan ada tiga bagian utama yaitu :

1. Insialisasi gambar marker
2. *Tracking* marker
3. *Rendering* objek

3.1.8. Analisis Marker

Marker merupakan perangkat keras lainnya yang merupakan salah satu bagian dari *Augmented Reality* (AR) yang digunakan untuk membuat suatu aplikasi *Augmented Reality* (AR), pada umumnya marker merupakan sebuah persegi hitam dan di tengahnya terdapat bentuk atau ciri yang membedakan antara satu marker dengan marker lain yang akan menjadi keypoint, marker yang terdapat ditengah bisa berbentuk gambar, angka, huruf, atau apa saja, didalam pola marker yang merupakan ilustrasi hitam dan putih dengan batas hitam yang tebal dan latar belakang putih digunakan agar komputer mengenali posisi dan orientasi marker dan menciptakan dunia virtual 3D yaitu pada titik (0,0,0) dari 3 sumbu yaitu X,Y, dan Z.

Pada aplikasi ini, objek *marker* yang digunakan berupa objek gambar berupa kartu yang didalamnya terdapat gambar Nama *Chord* Mayor atau Minor yang akan discan. Objek-objek tersebut akan merepresentasikan objek 3D yang akan ditampilkan dalam proses *augmented reality*. Penanda Minor atau mayor ini pula yang akan menjadi Pembeda saat marker discan dan menampilkan *augmented reality* dari chord tersebut.



Gambar 3.6 Marker Chord Mayor dan Minor

Tahapan selanjutnya, setelah menentukan gambar yang akan digunakan sebagai marker maka perlu dilakukannya proses meng-upload gambar tersebut ke situs developer vuforia untuk kemudian diproses agar gambar tersebut dapat dikenali dan dimasukkan ke dalam sistem yang akan dibangun. Setelah selesainya proses pengenalan gambar, tahap selanjutnya adalah men-download gambar tersebut dengan format *.unitypackage*.

3.1.9. Analisis Metode NFT (*Natural Feature Tracking*)

Natural Feature Tracking atau yang biasa disebut NFT merupakan metode yang digunakan di dalam Library Vuforia Qualcomm untuk mendeteksi pola gambar. Sebelum dilakukannya pendeteksian fitur-fitur pada gambar, terlebih dahulu dilakukan proses identifikasi marker. Identifikasi marker adalah proses untuk menangkap gambar yang ditangkap oleh kamera. Gambar yang diperoleh itu dianggap marker terlepas ada atau tidaknya marker yang sesungguhnya pada gambar. Setelah marker diperoleh maka proses selanjutnya adalah mengubah marker menjadi keabu-abuan atau proses grayscale. Marker yang sudah diubah menjadi keabu-abuan kemudian dideteksi fiturnya menggunakan metode NFT. Proses ini dilakukan untuk memudahkan dalam pendeteksian fitur pada marker. Adapun algoritma yang digunakan di dalam NFT itu sendiri adalah SIFT dan FERNS.

Komponen-komponen dalam proses deteksi fitur yang dilakukan adalah sebagai berikut :

1. Trackable Type
 - a. UNKNOWN_TYPE : Pelacakan yang tidak diketahui
 - b. IMAGE_TARGET : Pelacakan berdasarkan gambar
 - c. MULTI_TARGET : Pelacakan lebih dari 1 target
 - d. MARKER : Pelacakan marker
2. Trackable Name Pelacakan menggunakan kumpulan karakter unik yang disusun sebagai target identifikasi pelacakan. Untuk penulisan nama hanya diperbolehkan maksimal 64 karakter dan hanya mengandung karakter (a-z, A-Z, 0-9, [-_.]).

3. *Trackable* Status Informasi status pelacakan yang dilakukan per-frame dan akan selalu diupdate secara real-time. Status-status yang dapat terjadi pada saat pelacakan adalah sebagai berikut :
- UNKNOWN : gambar marker pada pelacakan tidak diketahui biasanya dikembalikan sebelum tracker initialization
 - UNDEFINE : gambar *marker* tidak didefinisikan
 - NOT_FOUND: gambar *marker* tidak ditemukan pada library yang dituju
 - DETECTED : gambar *marker* terdeteksi dalam frame
 - TRACKED : pelacakan telah terlacak dalam frame
 - Beberapa fungsi yang digunakan pada vuforia dalam proses tracking marker adalah sebagai berikut:

```
CameraDevice
```

Fungsi kelas ini untuk menyediakan akses ke kamera dan properti.

```
QCARBehaviour
```

Merupakan kelas untuk menangani pelacakan dan memicu tampilan *background rendering* video pada kamera.

```
DataSetLoadBehaviour
```

atau lebih *data set* ketika *startup* pada *image target*.

```
TrackableBehaviour.UnregisterTrackableEventHandler
```

Berfungsi sebagai *unregisters* sebuah *Tracker Event Handler* mengembalikan nilai “*false*” jika *event handler* tidak ada.

```
String TrackableBehaviour.TrackableName [get]
```

Pelacakan nama yang ada pada *TrackableBehaviour*.

```
void TrackableBehaviour.OnTrackerUpdate (Status newStatus) [inline, virtual]
```

Dipicu oleh *TrackerBehaviour* setelah itu diperbarui.

```
DefaultTrackableEventHandler.OnTrackableStateChanged
```

Sebuah *handler custom* yang mengimplementasikan antarmuka

```
ITrackableEventHandler.DefaultInitializationErrorHandler
```

Fungsi untuk menginisialisasi eror pada *DefaultTrackableEventHandler*

3.1.9.1. Pengambilan tekstur Marker

Tahap ini merupakan tahap dimana tekstur dari marker yang telah diwarnai akan dideteksi untuk mengetahui batas wilayah/region dari marker tersebut untuk kemudian di-capture agar diperoleh citra tekstur marker. Citra tekstur yang diambil ini kemudian diproses/dirender untuk mendapatkan citra tekstur yang dapat diaplikasikan pada objek 3D yang akan ditampilkan.

Pada tahap awal sistem akan membaca dan mendeteksi lokasi serta koordinat marker yang akan digunakan sebagai batas dari pengambilan citra tekstur *marker*. Setelah marker terdeteksi dan lokasi serta batas marker didapat, maka dilakukan proses pengambilan citra tekstur *marker*. Dari proses tersebut akan didapat citra tekstur marker yang kemudian akan diproses menjadi Texture2D yang merupakan *Game Object* yang dapat diaplikasikan sebagai tekstur pada objek 3D yang akan ditampilkan.

3.1.10. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional menggambarkan kebutuhan luar sistem yang diperlukan untuk menjalankan aplikasi yang dibangun. Adapun kebutuhan non-fungsional pada aplikasi meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan pengguna sistem yang akan memakai aplikasi.

3.1.10.1. Analisis Kebutuhan Perangkat Keras

Analisis Perangkat keras atau *hardware* merupakan salah satu hal yang penting karena tanpa *hardware* yang memenuhi syarat, program yang akan dibuat tidak akan dapat berjalan. Spesifikasi perangkat keras yang dapat dipergunakan untuk membangun aplikasi ini adalah sebagai berikut :

Tabel 3.6 Kebutuhan Perangkat Keras Pengembang

No.	Perangkat Keras	Spesifikasi
Laptop		
1	Processor	Intel(R) Core (TM) i5-2430M CPU @2.40GHz (4CPUs), ~2GHz

2	Monitor	14'' (inch)
3	VGA	AMD Radeon HD 6700M 2Gb
4	RAM	4GB
5	Hard disk	500GB
Mobile		
1	Layar	4'' (inch)
2	RAM	2 GB
3	Kamera	13 MP

Rekomendasi minimum spesifikasi kebutuhan perangkat keras yang digunakan untuk menjalankan aplikasi teknologi *augmented reality* ini pada perangkat *mobile* adalah sebagai berikut :

Tabel 3.7 Kebutuhan Perangkat Keras Pengguna

No.	Perangkat Keras	Spesifikasi
1	Layar	4'' (inch)
2	RAM	1 GB
3	SD Card	2 GB
4	Kamera	3 MP

3.1.10.2. Analisis Kebutuhan Perangkat Lunak

Analisis perangkat lunak atau *software* merupakan hal yang terpenting dalam mendukung kinerja sebuah sistem. Perangkat lunak digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar dapat saling berinteraksi diantara keduanya. Perangkat lunak yang dapat digunakan untuk membangun aplikasi teknologi *augmented reality* ini adalah sebagai berikut :

Tabel 3.8 Kebutuhan Perangkat Lunak Pengembang

No.	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Windows 10
2	Tool Pembangunan	Unity 3D
3	Tool Design	Blender
4	Library	Vuforia SDK

Rekomendasi minimum spesifikasi kebutuhan perangkat lunak yang digunakan untuk menjalankan aplikasi teknologi *augmented Realty* ini pada perangkat *mobile* adalah sebagai berikut :

Tabel 3.9 Kebutuhan Perangkat Lunak Pengguna

No.	Perangkat Lunak	Spesifikasi
1	Operating System (OS)	Android
2	Versi Android	6.0.1 Marshmallow

3.1.10.3. Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna merupakan analisis terhadap *user* yang akan menggunakan sistem yang akan dibangun. Adapun kebutuhan pengguna yang terlibat dalam penggunaan aplikasi teknologi Augmented Reality ini dapat dilihat pada tabel 3.5 berikut :

Tabel 3.10 Analisis Kebutuhan Pengguna

Pengguna	Calon pemain gitar
Hak Akses	Menggunakan Aplikasi
User Experience	Dapat menggunakan perangkat <i>mobile</i> terutama Android.

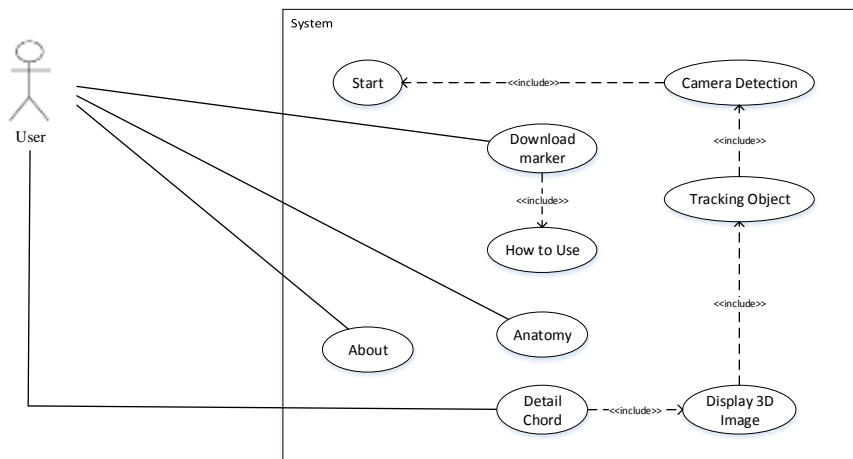
3.1.11. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan kebutuhan sistem yang akan dibangun pada aplikasi teknologi *augmented reality* ini. Adapun kebutuhan fungsional pada aplikasi yang akan dibangun ini dengan pemodelan berorientasi objek. Perangkat lunak ini dimodelkan menggunakan UML (*Unified Modeling*

Language). Tahap-tahap pemodelan dalam analisis tersebut antara lain *use case diagram*, *skenario*, *sequence diagram*, *activity diagram*, *class diagram*.

3.1.11.1. Use case diagram

Use Case merupakan gambaran skenario dari interaksi antara *user* dengan sistem. Sebuah diagram *Use Case* menggambarkan hubungan antara aktor dan kegiatan yang dapat dilakukannya terhadap aplikasi.



Gambar 3.7 Use case diagram

3.1.11.1.1. Defenisi Actor

Definisi *Actor* berfungsi untuk menjelaskan *Actor* yang terdapat pada *Use Case Diagram*. Definisi *Actor* diterangkan pada Tabel 3.8.

Tabel 3.11 Defenisi Actor

No.	Actor	Deskripsi
1	<i>User</i>	Orang yang berinteraksi dengan system, yaitu pengguna yang mengakses semua fungsi yang disediakan sistem.

3.1.11.1.2. Defenisi Use case

Definisi *Use Case* berfungsi untuk menjelaskan fungsi *Use Case* yang terdapat pada *Use Case Diagram*. Definisi *Use Case* diterangkan pada Tabel 3.9.

Tabel 3.12 Defensi Use case

No.	Use Case	Deskripsi
1	<i>Start</i>	Fungsionalitas untuk memulai aplikasi AR
2	<i>How to use</i>	Fungsionalitas untuk menampilkan info cara menggunakan
3	<i>About</i>	Fungsionalitas untuk menampilkan deskripsi singkat tentang aplikasi
4	Deteksi Kamera	Fungsionalitas mendeteksi ketersediaan kamera.
5	<i>Tracking Object</i>	Fungsionalitas untuk mendeteksi keberadaan atau mengenali obyek
6	Menampilkan Gambar 3D	Fungsionalitas untuk menampilkan denah berupa obyek 3D pada layar <i>device</i> / perangkat mobile

1. Scenario *Use case* menggunakan AR

Tabel 3.10 di bawah ini adalah Skenario *Use Case* untuk memulai aplikasi AR Bangunan.

Tabel 3.13 Skenario use case Start

Identifikasi	
Nomor	UCS-01
Nama	<i>Start</i>
Tujuan	Menjalankan menu menggunakan AR
Deskripsi	Proses untuk menampilkan obyek 3D dari Kunci gitar.
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Pengguna berada pada menu utama
Aksi Aktor	Reaksi Sistem
1. Pengguna masuk menu menggunakan AR	

	2. Sistem mendeteksi kamera 3. Kamera Aktif
4. Pengguna mengarahkan kamera pada <i>marker</i> .	
	5. Sistem menampilkan obyek ruangan 3D.
Kondisi Akhir	Aplikasi menampilkan objek ruangan 3D dari <i>Chord</i> Gitar

2. Skenario *Usecase How to use*

Tabel 3.9 di bawah ini adalah Skenario *Usecase How to use* untuk cara menggunakan aplikasi AR *Chord* gitar.

Tabel 3.14 Usecase How to use

Identifikasi	
Nomor	UCS-02
Nama	<i>How to use</i>
Tujuan	Menampilkan info cara menggunakan aplikasi
Deskripsi	Proses untuk menampilkan Info cara penggunaan Aplikasi
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Pengguna berada pada menu utama
Aksi Aktor	Reaksi Sistem
1. Pengguna masuk pada menu <i>How to use</i>	
	2. Menampilkan informasi cara penggunaan aplikasi

Kondisi Akhir	Aplikasi menampilkan info cara menggunakan aplikasi
----------------------	---

3. *Usecase Download Marker*

Tabel 3.14 di bawah ini adalah *Skenario download Marker* untuk membuka *link* unduh *marker*.

Tabel 3.15 Skenario Download Marker

Identifikasi	
Nomor	UCS-03
Nama	<i>Download Marker</i>
Tujuan	Menjalankan menu unduk <i>marker</i>
Deskripsi	Proses untuk membuka <i>link download marker</i>
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Pengguna berada pada menu utama
Aksi Aktor	Reaksi Sistem
1. Pengguna memilih halaman How to use	
	2. Sistem menampilkan halaman cara penggunaan dan tombol download marker
3. Pengguna memilih tombol <i>download marker</i>	
	4. Sistem membuka <i>link</i> untuk mengunduh <i>marker</i>
Kondisi Akhir	Aplikasi menampilkan <i>link</i> unduh <i>marker</i>

4. *Usecase Anatomy*

Tabel 3.15 di bawah ini adalah *Skenario anatomy* untuk menampilkan materi penjelasan bagian-bagian gitar.

Tabel 3.16 Skenario Anotomy

Identifikasi	
Nomor	UCS-04
Nama	<i>Anatomy</i>
Tujuan	Menampilkan info bagian bagian gitar
Deskripsi	Proses untuk menampilkan Info materi bagian-bagian gitar
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Pengguna berada pada menu utama
Aksi Aktor	Reaksi Sistem
1. Pengguna memilih <i>Anatomy</i>	
	2. Menampilkan informasi bagian-bagian gitar
Kondisi Akhir	Aplikasi menampilkan info bagian-bagian gitar

5. *Use case About*

Tabel 3.16 di bawah ini adalah *Skenario Use Case About* pada aplikasi AR Bangunan.

Tabel 3.17 Use case diagram About

Identifikasi	
Nomor	UCS-05
Nama	<i>About</i>
Tujuan	Menampilkan halaman deskripsi singkat aplikasi

Deskripsi	Proses untuk info deskripsi singkat aplikasi
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Pengguna berada pada menu utama
Aksi Aktor	Reaksi Sistem
1. <i>Pengguna memilih menu About</i>	
	2. Menampilkan info deskripsi singkat aplikasi
Kondisi akhir	Aplikasi menampilkan info deskripsi singkat tentang aplikasi

6. Skenario *Use Case* Deteksi Kamera

Tabel 3.17 di bawah ini adalah *Skenario Use Case* deteksi kamera pada aplikasi pengenalan rangka dan bagian tumbuhan.

Tabel 3.18 Use case deteksi kamera

Identifikasi	
Nomor	UCS-06
Nama	Mendeteksi Kamera
Tujuan	Mengaktifkan kamera
Deskripsi	Proses mengaktifkan kamera
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Kamera dalam kondisi belum aktif
Aksi Aktor	Reaksi Sistem
1. Pengguna memilih menu menggunakan AR	
	2. Sistem memeriksa ketersediaan kamera
	3. Kamera Aktif

Kondisi akhir	Kamera dalam kondisi aktif
----------------------	----------------------------

7. Skenario *Use Case Tracking Objek*

Tabel 3.18 di bawah ini adalah *Skenario Use Case tracking* obyek pada aplikasi.

Tabel 3.19 Use case tracking objek

Identifikasi	
Nomor	UCS-07
Nama	Tracking Objek
Tujuan	Mendeteksi keberadaan objek
Deskripsi	Sistem mendeteksi keberadaan obyek melalui gambar yang ditangkap oleh kamera
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Kamera Aktif
Aksi Aktor	Reaksi Sistem
1. Pengguna memilih menu <i>Start</i>	
	2. Sistem memproses masukan gambar yang ditangkap oleh kamera
	3. Sistem mencocokkan pola
	4. Obyek dikenali
Kondisi Akhir	Obyek dikenali

8. Skenario *Use case* menampilkan objek 3D

Tabel 3.19 di bawah ini adalah *Skenario Use Case* Menampilkan Denah 3D pada aplikasi AR.

Tabel 3.20 Use case menampilkan Objek 3D

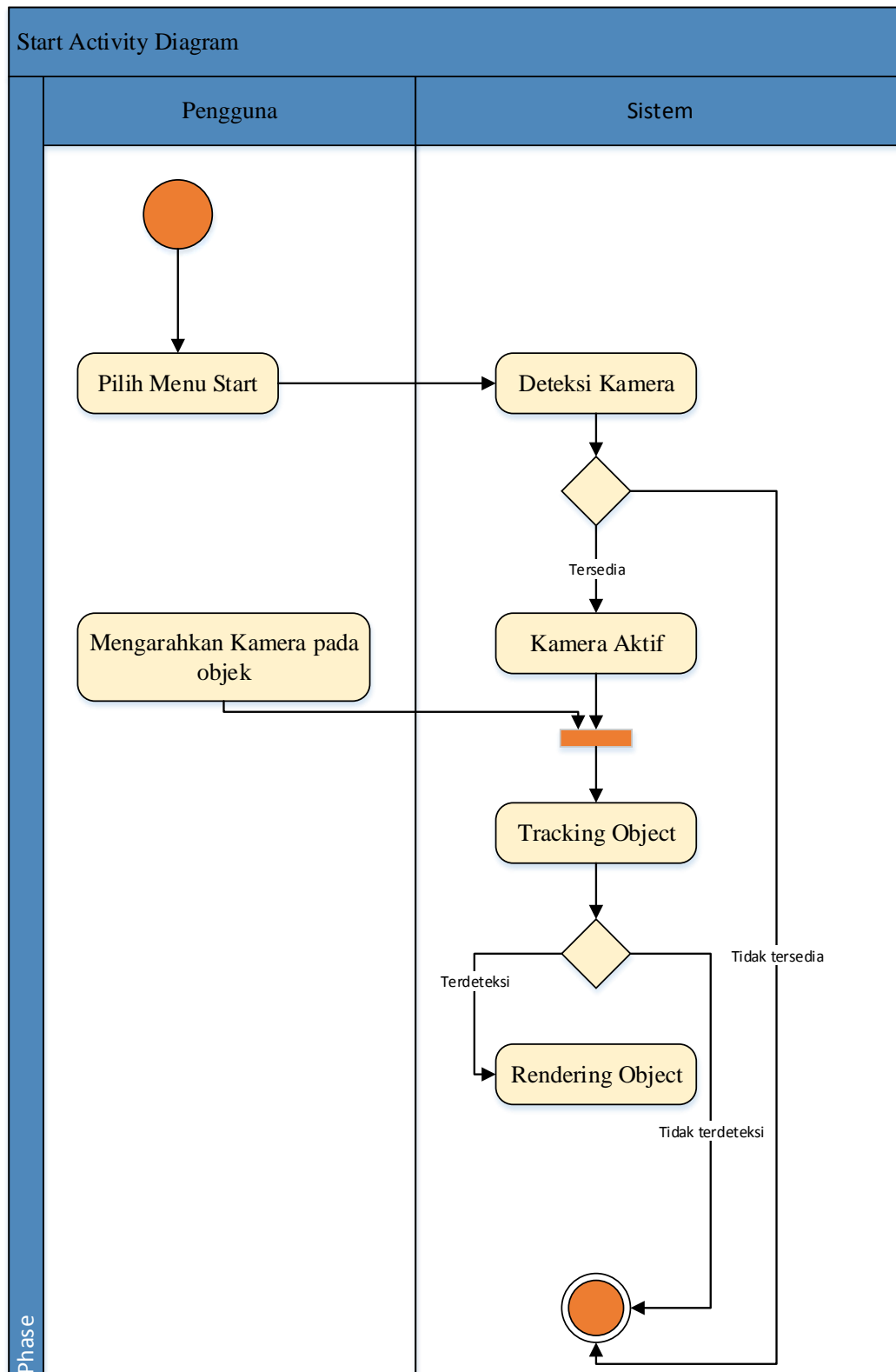
Identifikasi	
Nomor	UCS-08
Nama	Menampilkan objek 3D
Tujuan	Menampilkan objek 3D pada layar handphone
Deskripsi	Proses untuk menampilkan obyek 3D pada layar <i>handphone</i>
Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Kamera Aktif
Aksi Aktor	Reaksi Sistem
1. Pengguna mengarahkan kamera pada objek	
	2. Sistem memproses masukan gambar yang ditangkap oleh kamera 3. Sistem mencocokkan pola 4. Objek dikenali 5. Objek 3D tampil di layar <i>handphone</i>
Kondisi akhir	Objek 3D tampil dilayar handphone

3.1.11.2. Activity diagram

Activity diagram menggambarkan urutan aktivitas dalam sebuah proses. Struktur diagram ini mirip *flowchart* atau *Data Flow Diagram* pada perancangan terstruktur. Berikut ini merupakan *activity diagram* pada sistem yang akan dibangun.

3.1.11.2.1. Activity Diagram Start

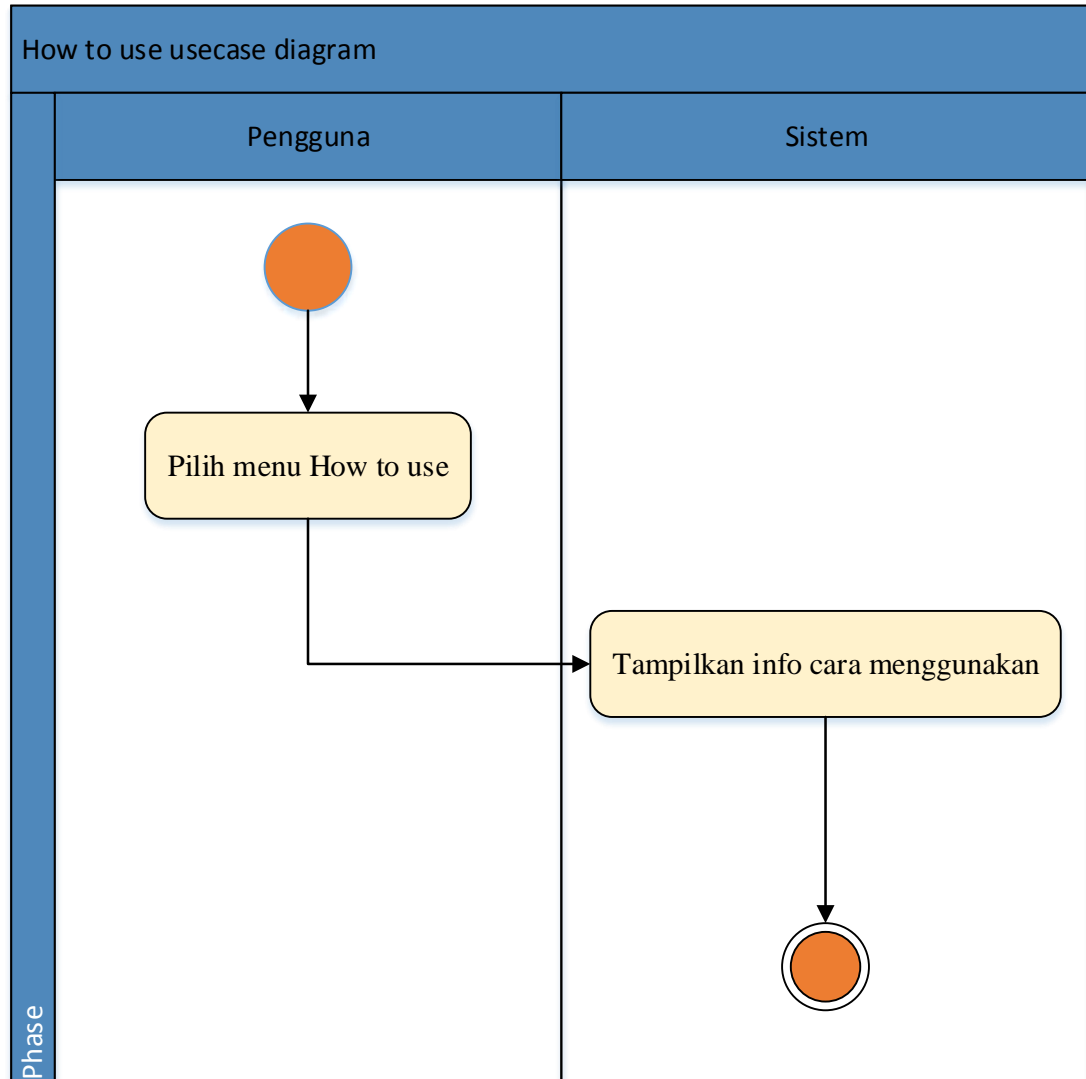
Activity Diagram terkait skenario Start memiliki alur proses seperti digambarkan sebagai berikut :



Gambar 3.8 Activity Diagram start

3.1.11.2.2. Activity diagram How to use

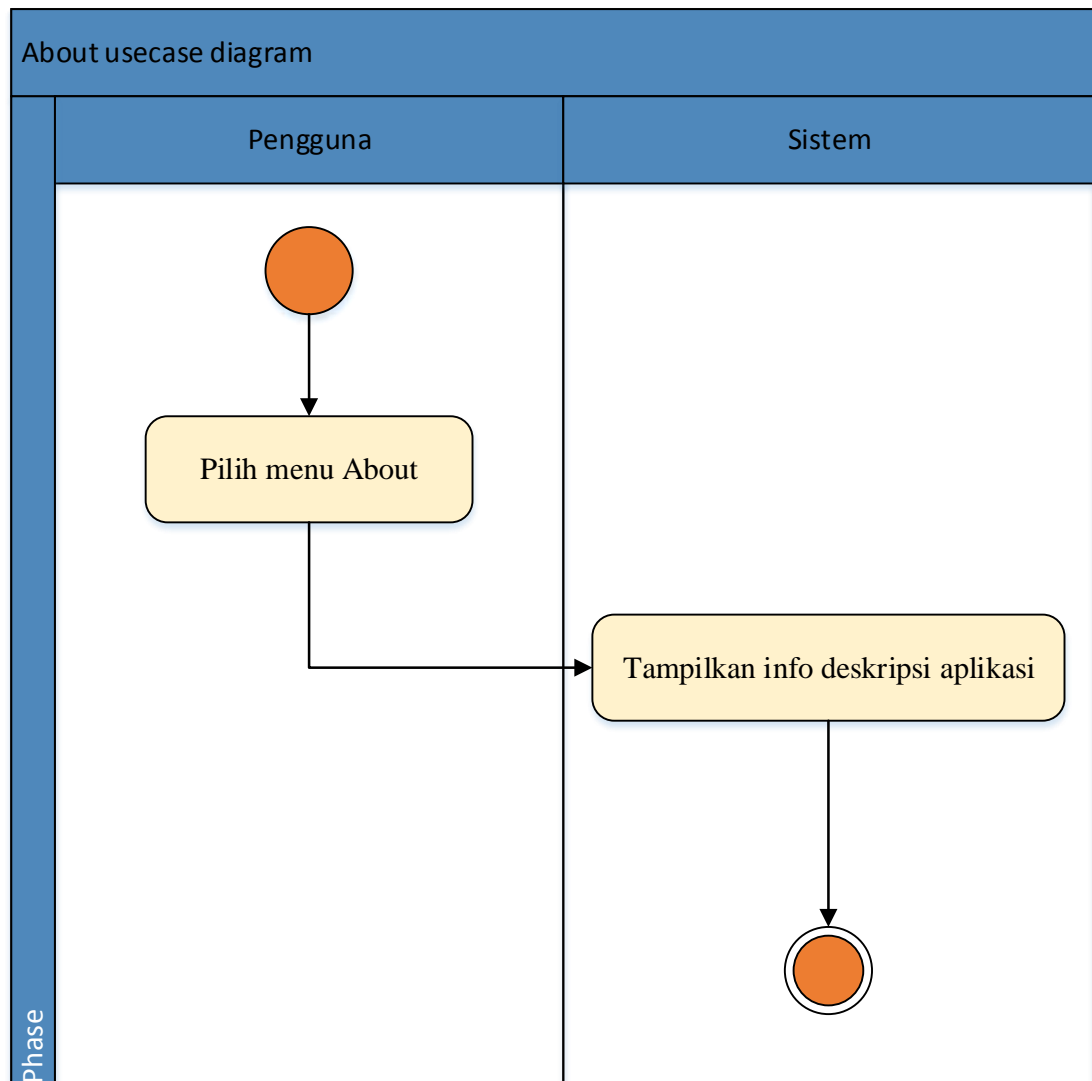
Activity Diagram terkait skenario *How to use* memiliki alur proses seperti digambarkan sebagai berikut :



Gambar 3.9 activity diagram How to use

3.1.11.2.3. Activity diagram About

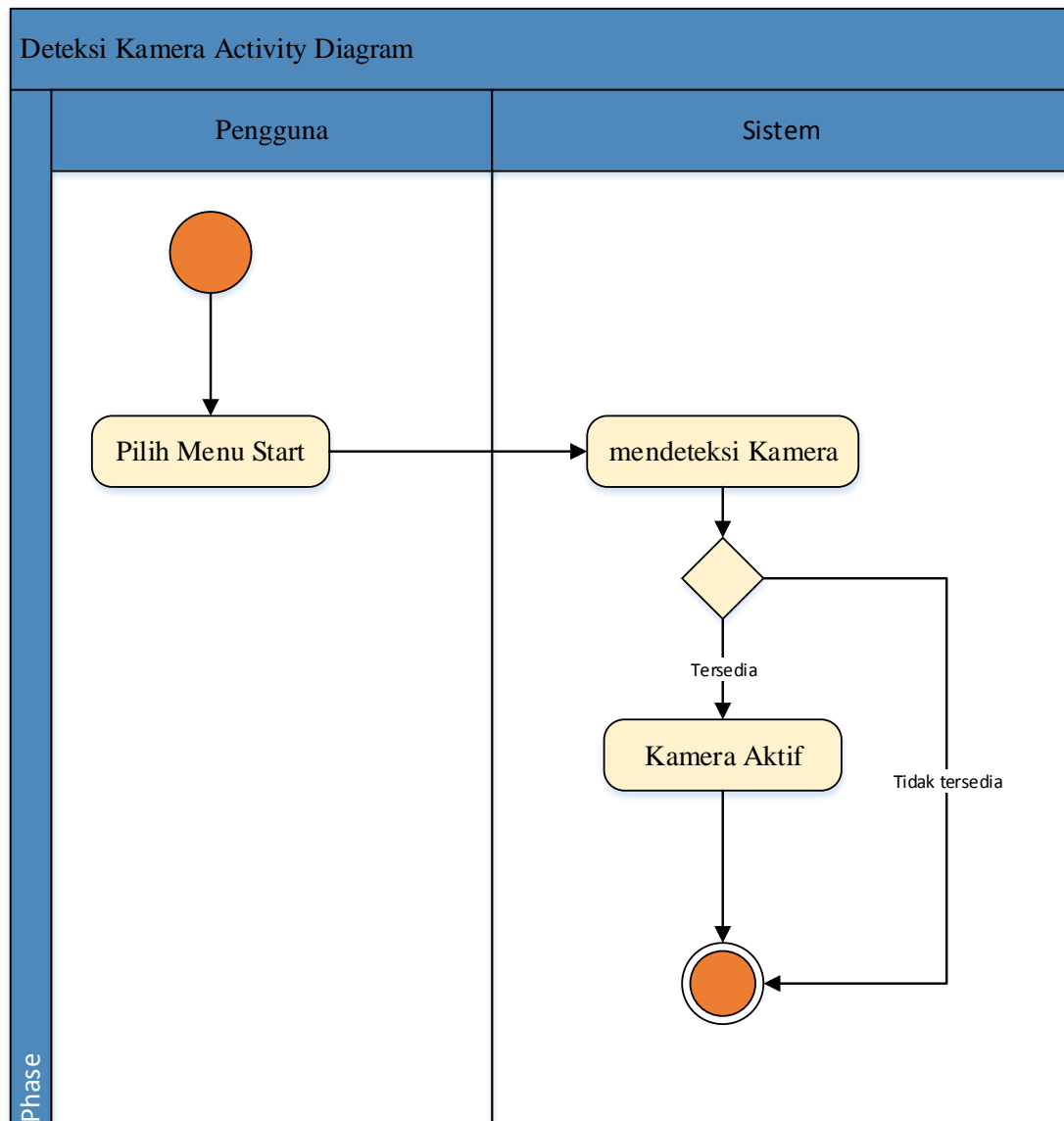
Activity Diagram terkait skenario *About* memiliki alur proses seperti digambarkan sebagai berikut :



Gambar 3.10 Activity Diagram About

3.1.11.2.4. Activity diagram Camera detection

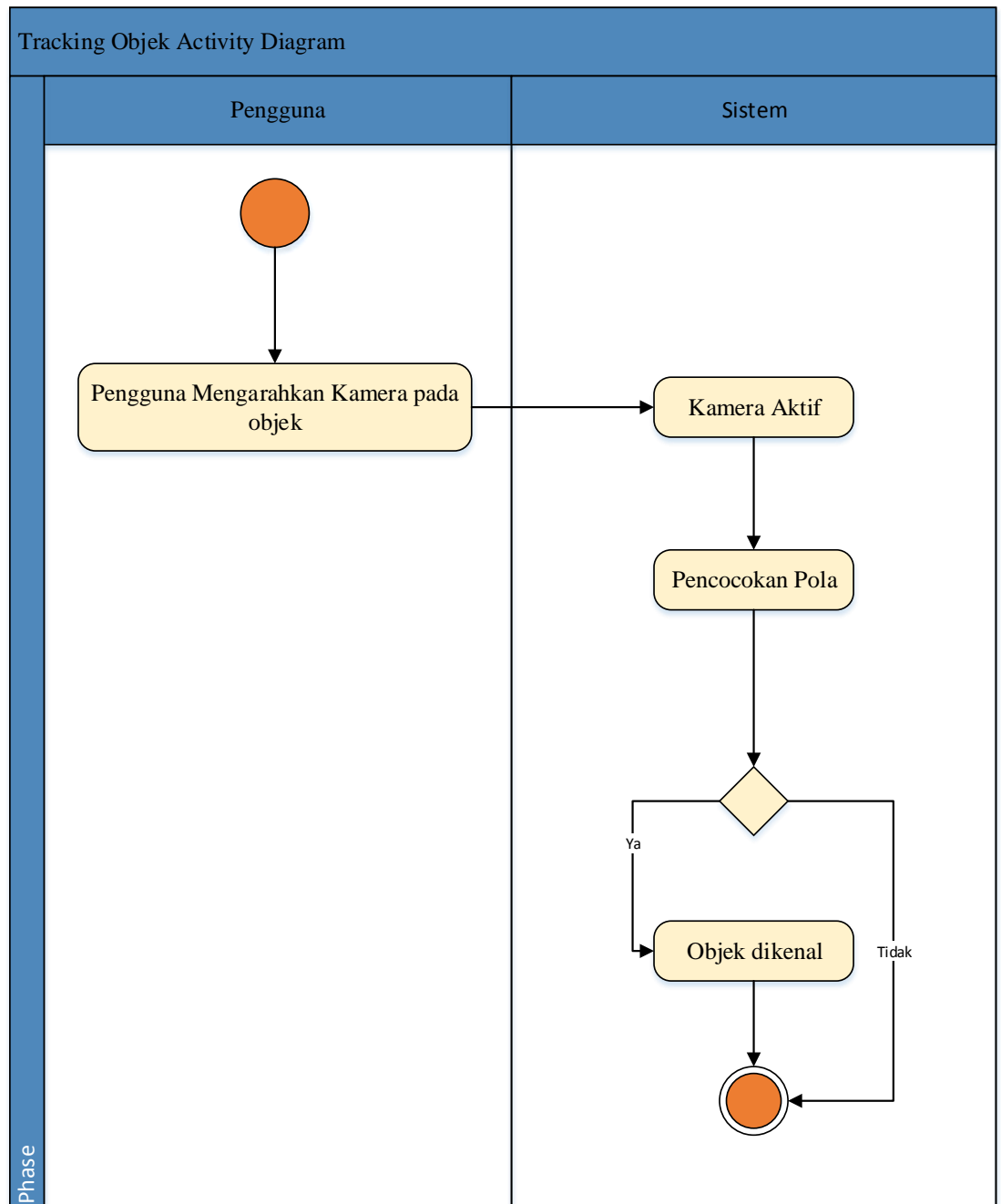
Activity Diagram terkait skenario Deteksi Kamera memiliki alur proses seperti digambarkan sebagai berikut :



Gambar 3.11 Activity diagram Deteksi Kamera

3.1.11.2.5. Activity diagram Tracking Objek

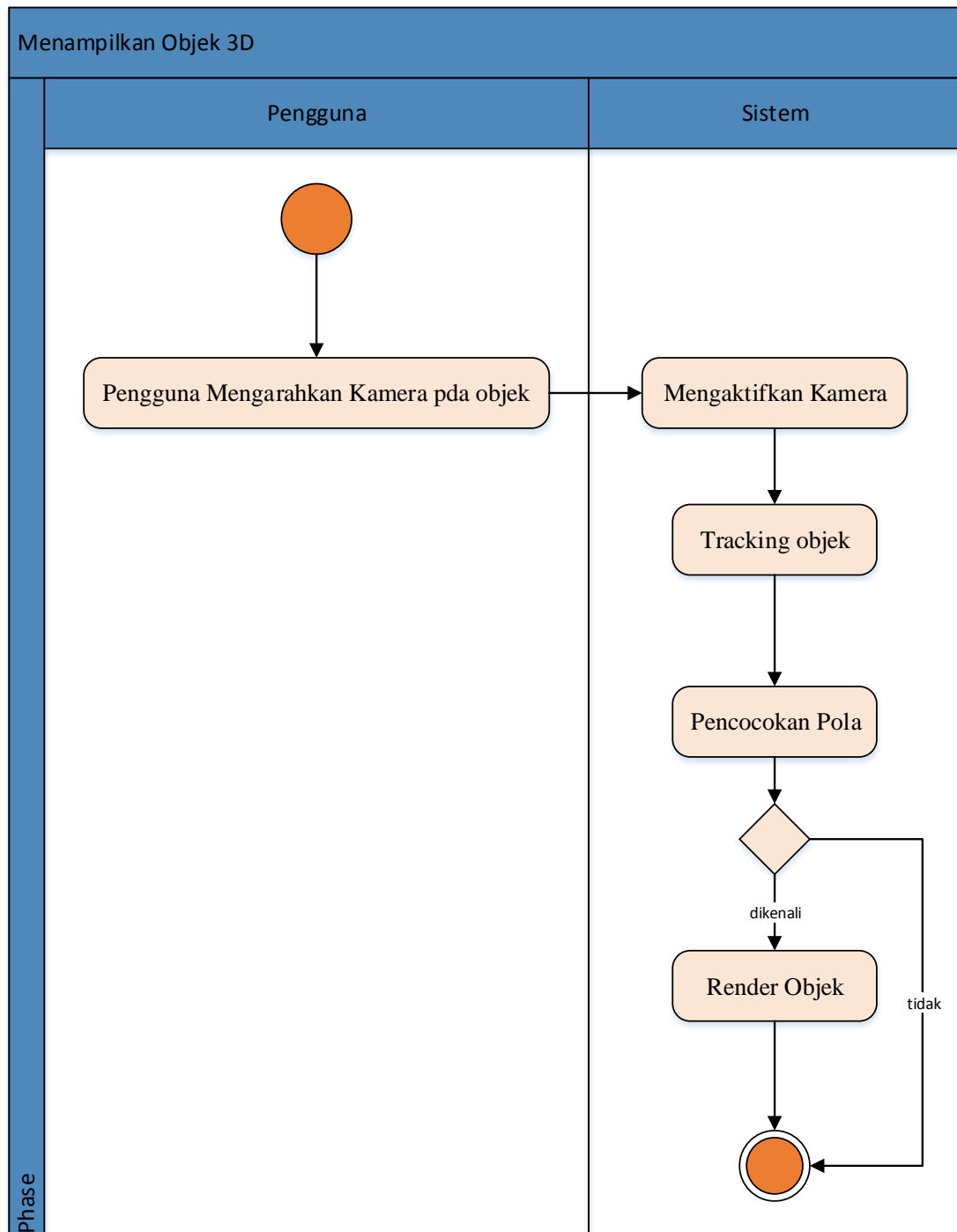
Activity Diagram terkait skenario Deteksi Kamera memiliki alur proses seperti digambarkan sebagai berikut :



Gambar 3.12 Activity Diagram Tracking Objek

3.1.11.2.6. Activity diagram Menampilkan Objek 3D

Activity Diagram terkait skenario Menampilkan Objek 3D memiliki alur proses seperti digambarkan sebagai berikut :



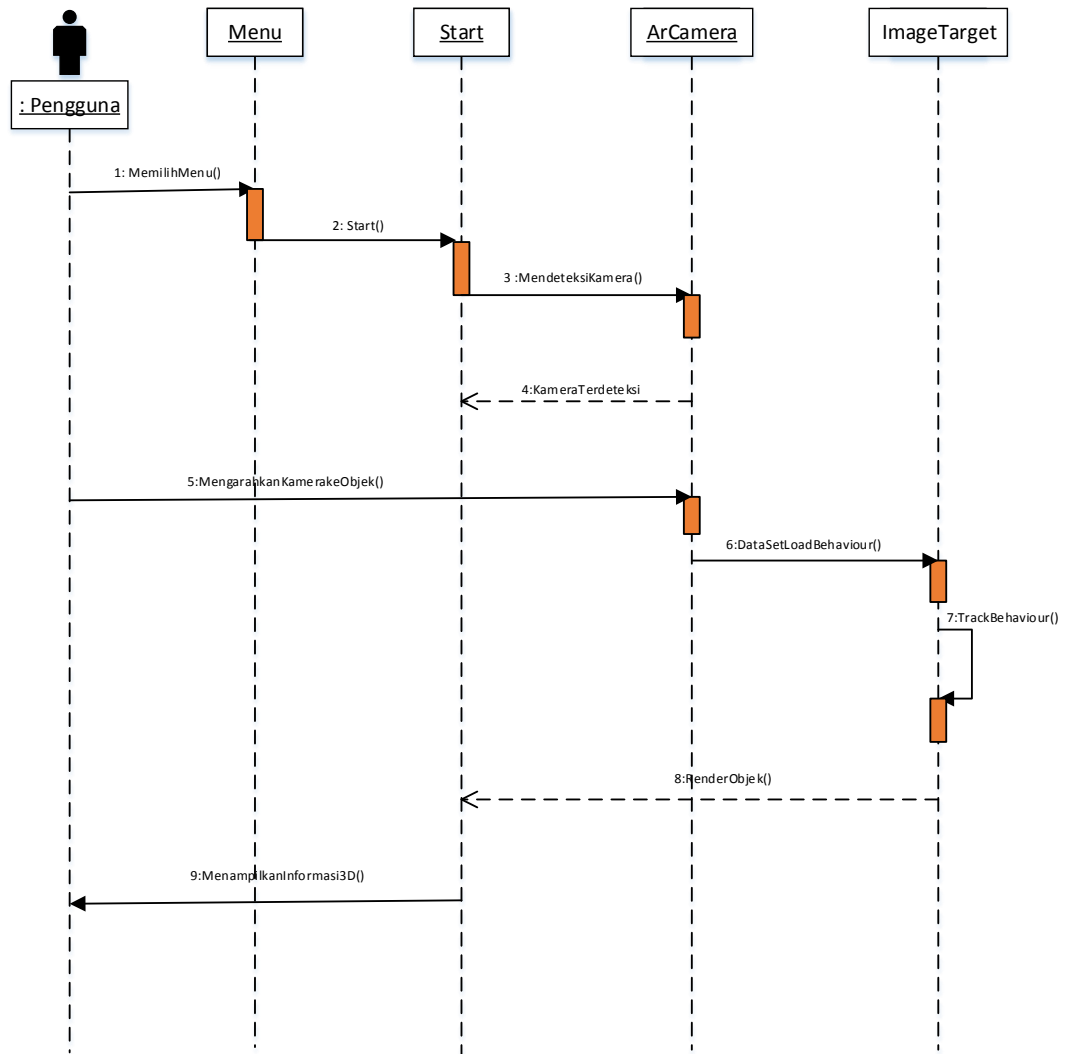
Gambar 3.13 Activity diagram menampilkan objek 3D

3.1.11.3. Sequence Diagram

Sequence diagram menjelaskan gambaran interaksi antar obyek dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar obyek-obyek

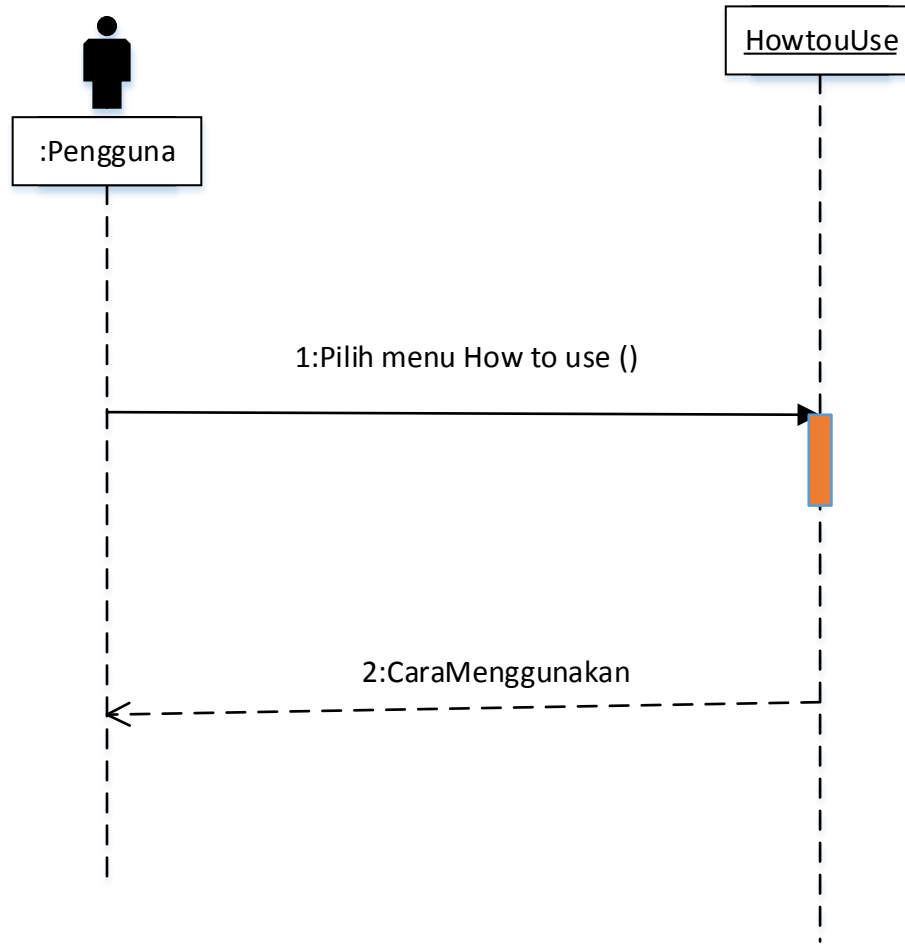
yang saling berinteraksi. *Sequence diagram* pada aplikasi ini terdiri dari *Sequence diagram* menu utama, mulai *game*, pengaturan dan instruksi.

3.1.11.3.1. *Sequence diagram Start*



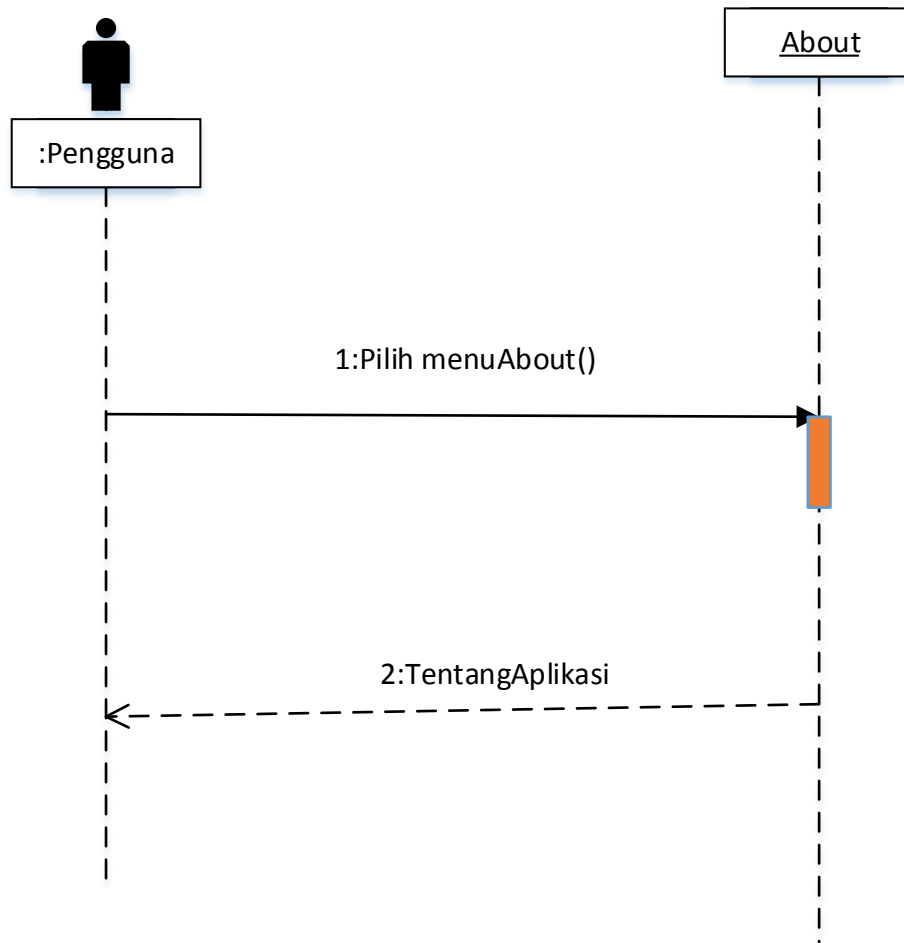
Gambar 3.14 *Sequence Diagram Start*

3.1.11.3.2. Sequence diagram How to use

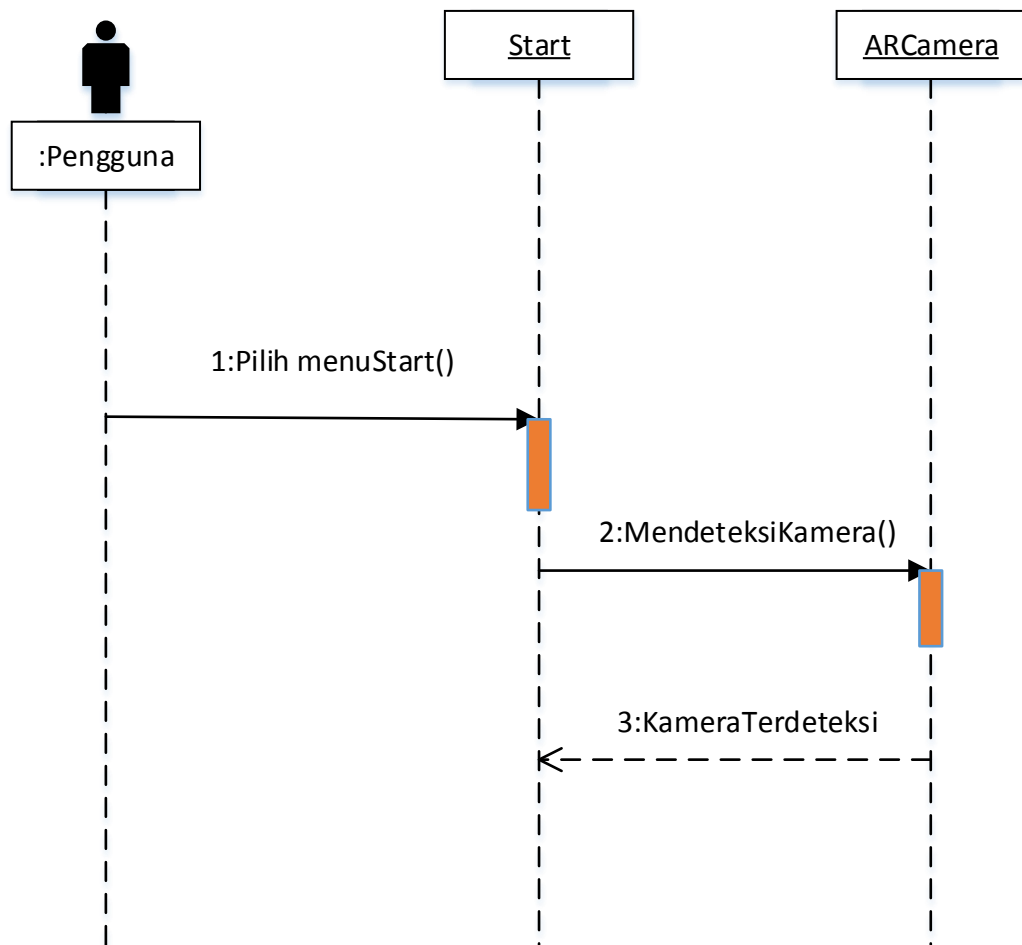


Gambar 3.15 Sequence diagram How to use

3.1.11.3.3. Sequence diagram About

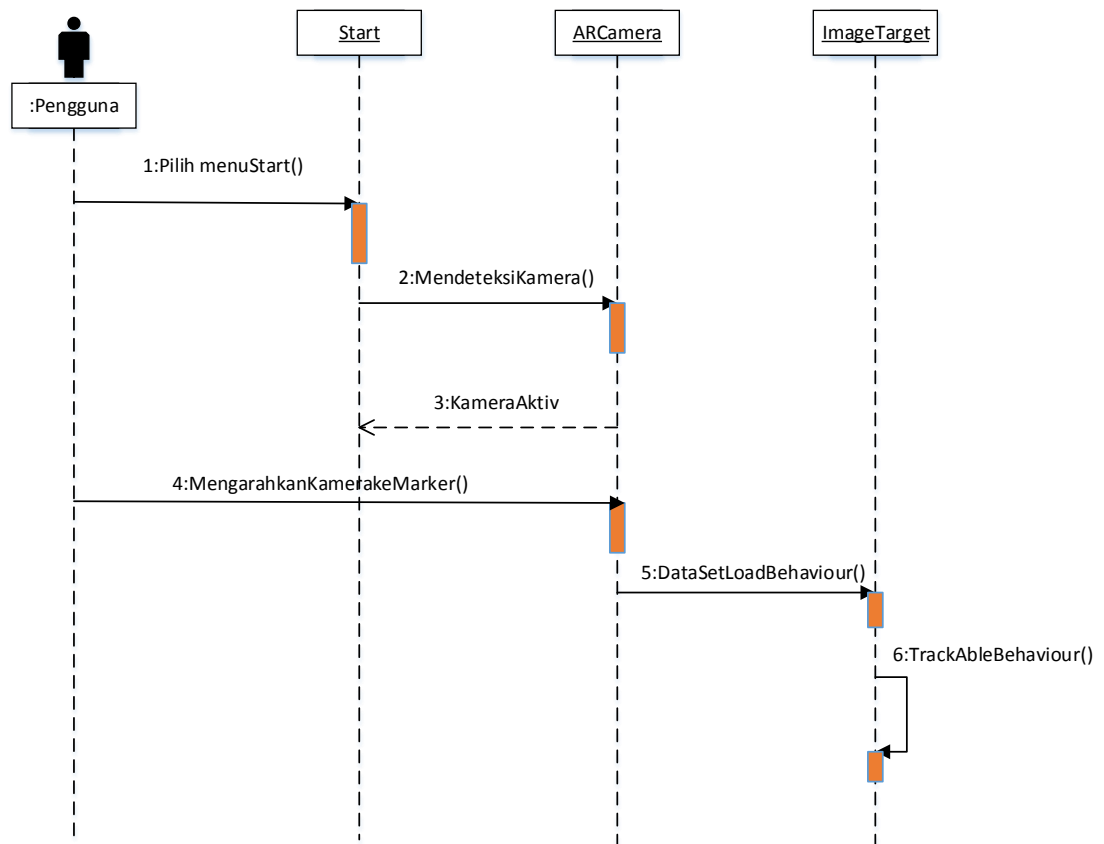


Gambar 3.16 Sequence Diagram About

3.1.11.3.4. *Sequence diagram camera detection*

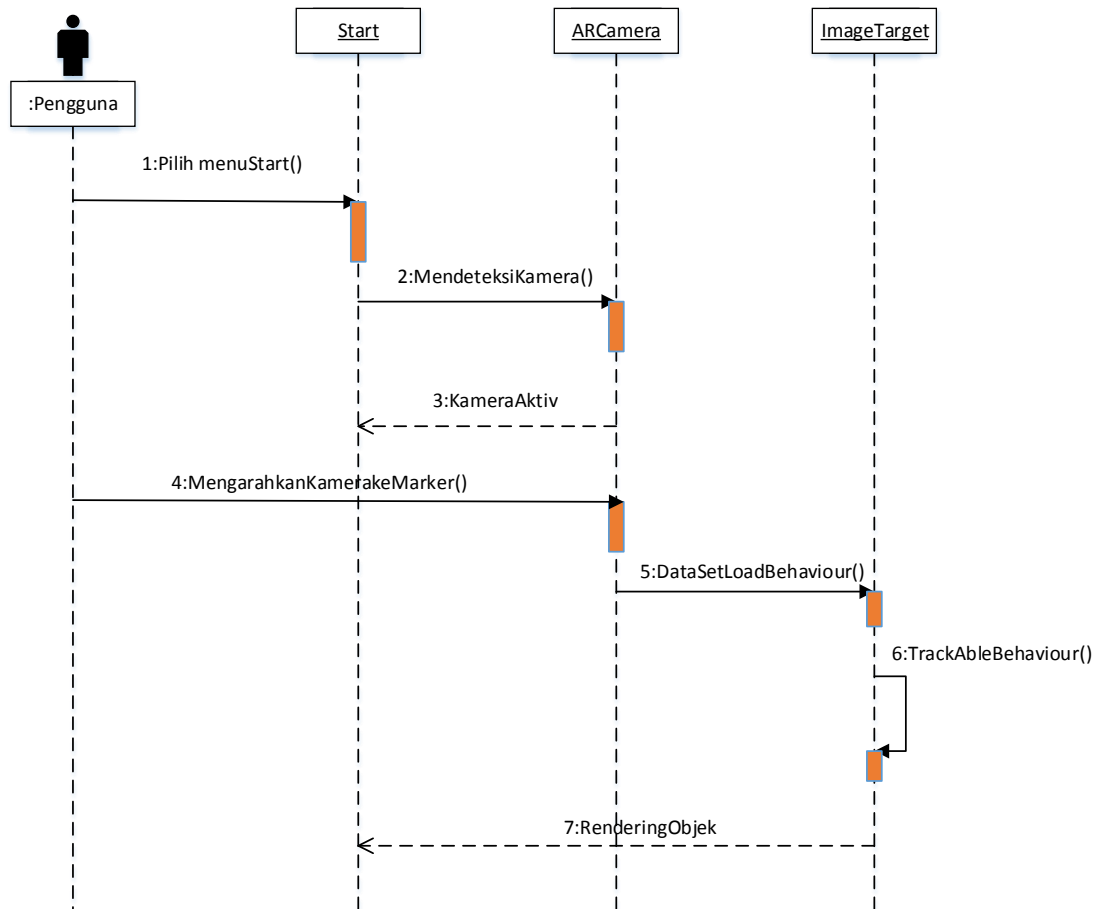
Gambar 3.17 Sequence diagram camera detection

3.1.11.3.5. Sequence diagram Tracking Objek



Gambar 3.18 Sequence diagram tracking objek

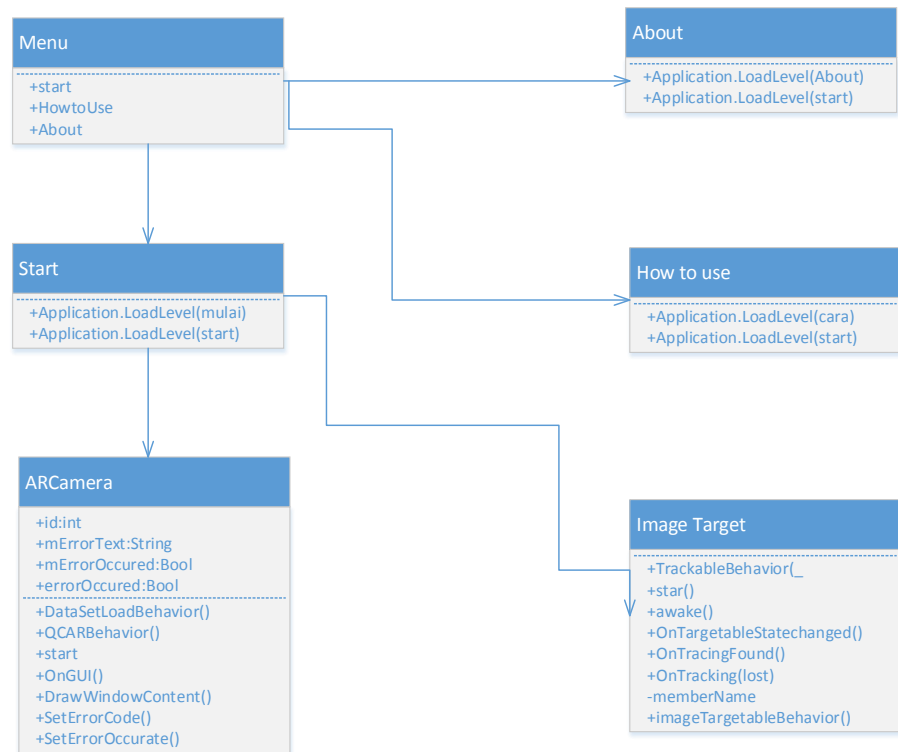
3.1.11.3.6. Sequence diagram menampilkan Objek 3D



Gambar 3.19 Sequence diagram Menampilkan Objek 3D

3.1.11.4. Class Diagram

Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (*attribut* atau *property*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda atau fungsi). Berikut adalah *class diagram* dari aplikasi denah AR.



Gambar 3.20 Class Diagram

3.2. Perancangan Sistem

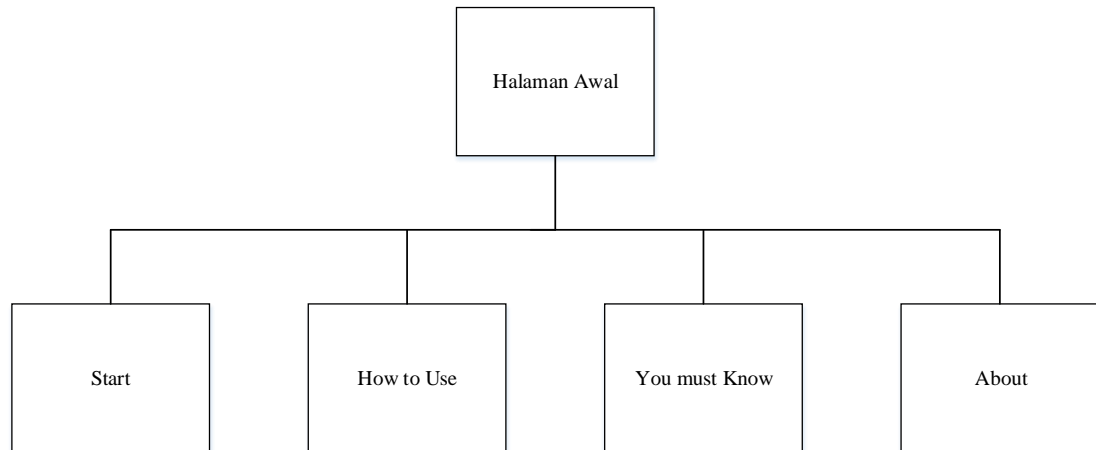
Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakukan setelah melalui tahapan analisis. Untuk dapat melakukan perancangan sistem maka dilakukan beberapa buah langkah yaitu sebagai berikut :

1. Perancangan Struktur *Menu*
2. Perancangan Antarmuka
3. Perancangan Marker
4. Perancangan Jaringan Semantik
5. Perancangan Method

3.2.1. Perancangan Struktur Menu

Struktur *menu* adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer. Saat menjalankan program, pengguna tidak mengalami kesulitan dalam memilih *menu-menu* yang

diinginkan. Berikut ini perancangan struktur *menu* pada aplikasi Pembelajaran chord gitar yang ditunjukkan pada gambar berikut ini.

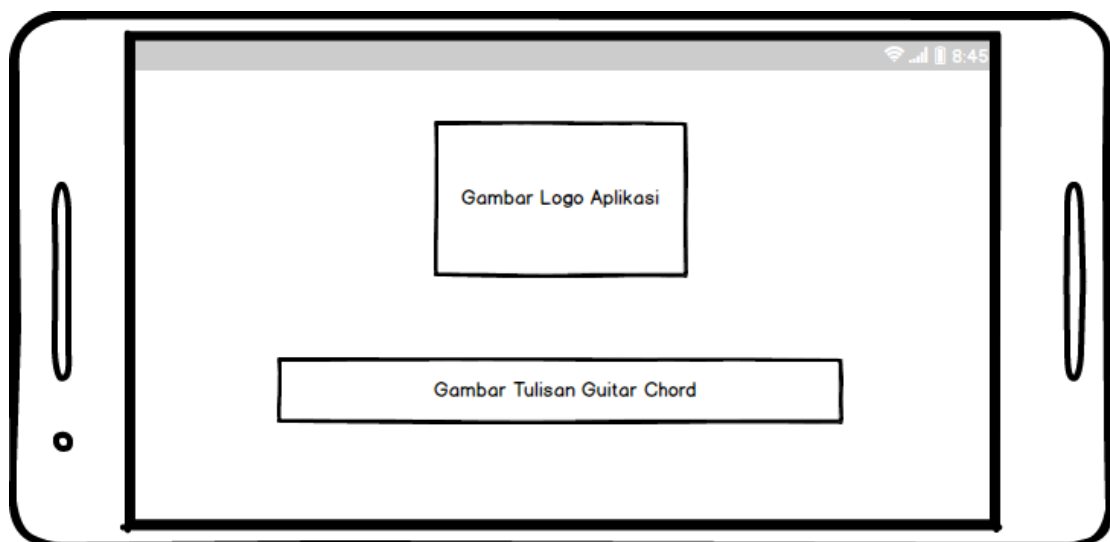


Gambar 3.21 Struktur Menu

3.2.2. Perancangan Antar Muka

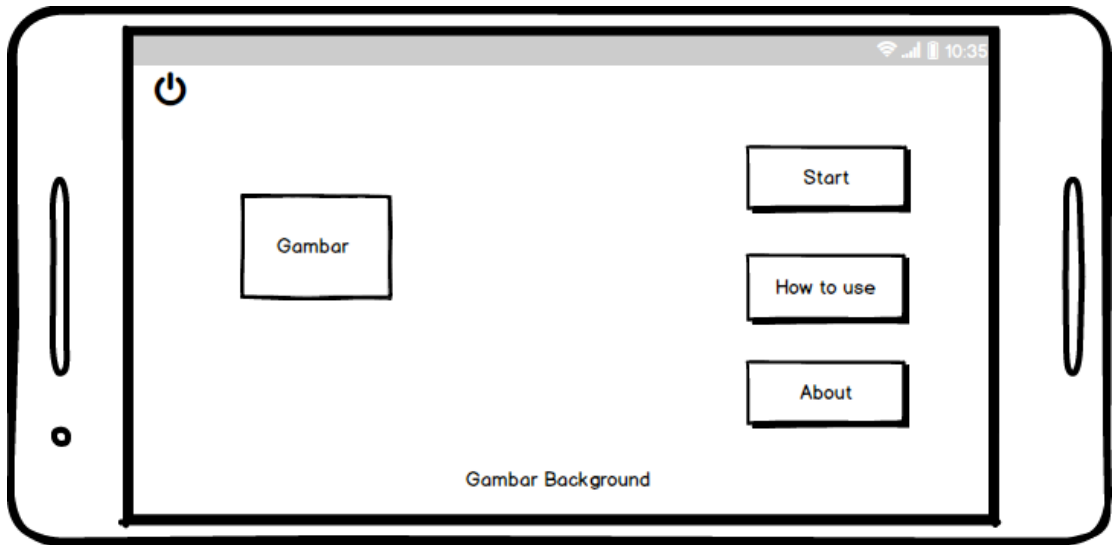
Perancangan antarmuka dibutuhkan untuk mewakili keadaan sebenarnya dari aplikasi yang akan dibangun. Berikut ini contoh perancangan antarmuka yang ada dari aplikasi yang akan dibangun yaitu :

1. Perancangan Antar muka Splash Screen



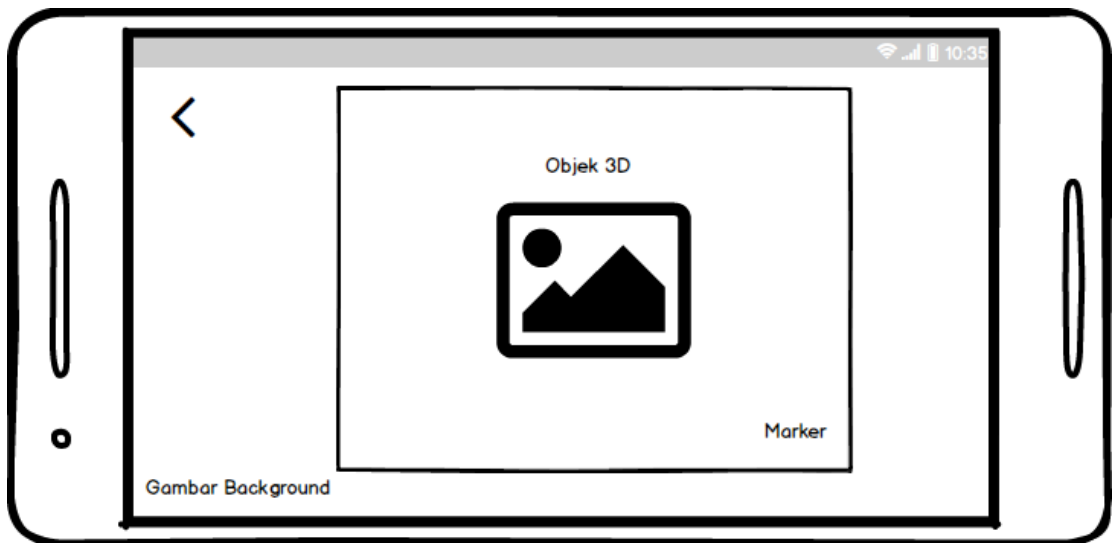
Gambar 3.22 Splash Screen

2. Perancangan Antar Muka Menu Utama



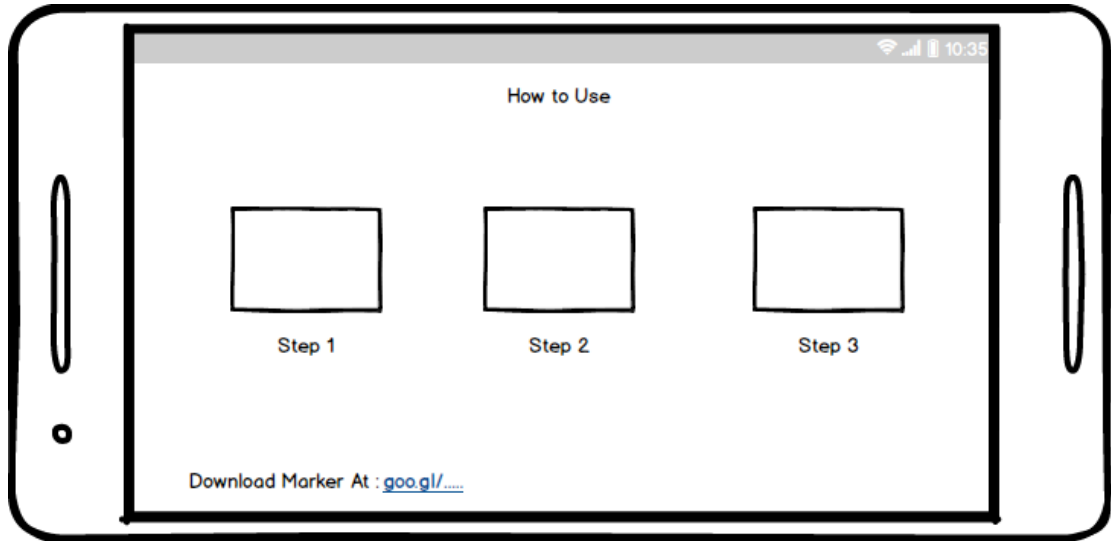
Gambar 3.23 Menu Utama

3. Perancangan Antar muka menu Start



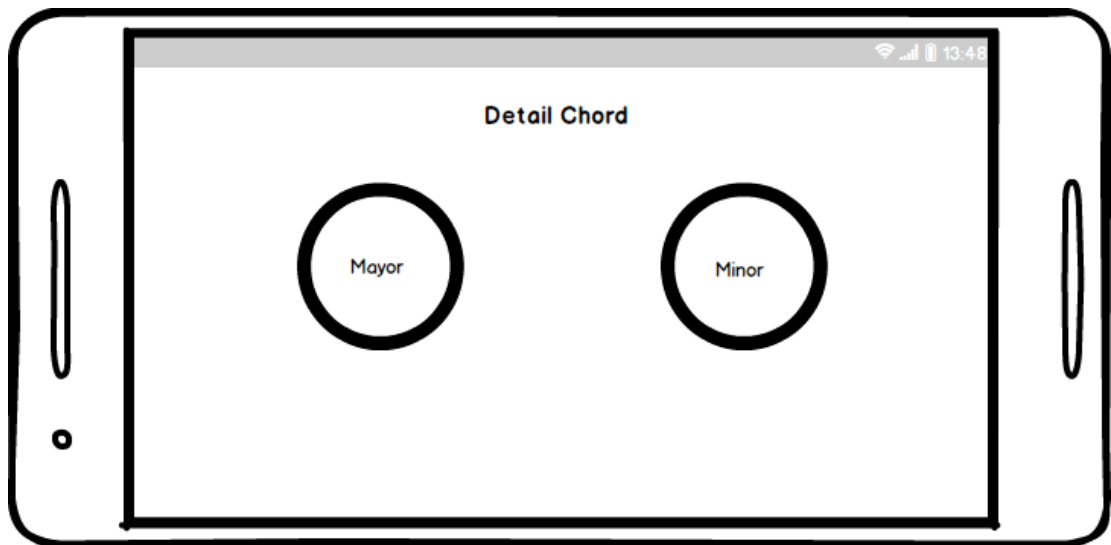
Gambar 3.24 Menu Start

4. Perancangan Antar muka menu *How to use*



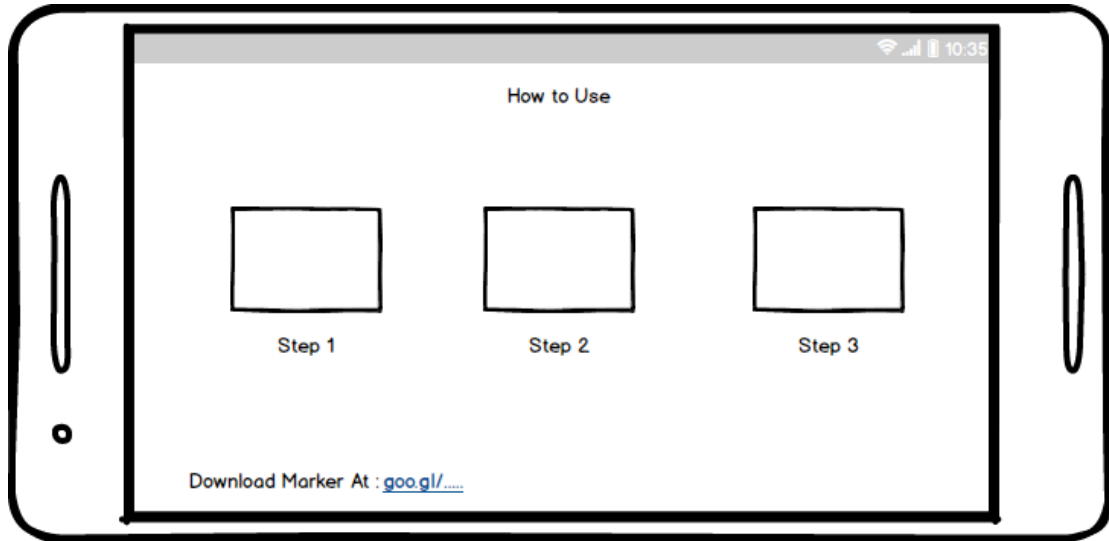
Gambar 3.25 Menu How to use

1. Perancangan Antar muka Detail *Chord*



Gambar 3.26 Detail Chord

2. Perancangan Antar muka menu *How to use*



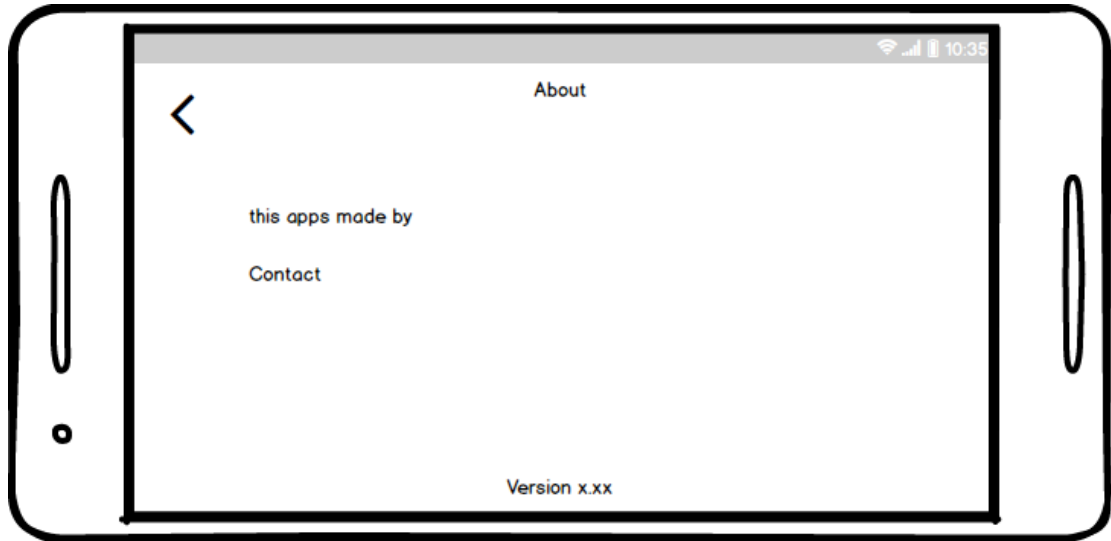
Gambar 3.27 Menu How to use

3. Percangan antar muka Menu *About*



Gambar 3.28 Menu Anatomy

4. Perancangan antar muka Menu *About*



Gambar 3.29 Menu About




3.2.3. Perancangan *Marker*



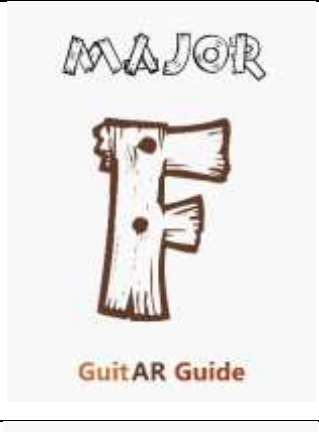

Perancangan *Marker* dibutuhkan untuk mewakili keadaan sebenarnya dari *marker* yang akan dibangun. *Marker* tersebut akan di print dalam bentuk kartu berukuran 8,7 cm x 6,2 cm. Berikut ini contoh perancangan *marker* yang ada dari aplikasi yang akan dibangun yaitu :

3.2.3.1. Perancangan *Marker Chord Mayor*

Marker pada *chord* mayor dirancang dalam bentuk kartu, yang didalamnya terdapat gambar yang akan dideteksi oleh kamera sehingga aplikasi akan menampilkan model 3D *chord* mayor pada *smartphone*. Gambar yang terdapat pada *marker chord* mayor adalah huruf dari nama *chord* tersebut, misalnya *chord A*, dalam *marker* tersebut terdapat huruf dalam font berwarna coklat, pada bagian atas terdapat tulisan “MAJOR” berwarna hitam dan bagian bawah terdapat tulisan nama aplikasi yaitu “GuitAr Guide”

Tabel 3.21 Tabel Marker Chord Mayor



No.	Gambar Marker	Nama
1	 <p>The image shows a hand holding a guitar neck with the A major chord shape. The word 'MAJOR' is written in a stylized font above the hand. Below the hand, the text 'GuitAR Guide' is written in a smaller font.</p>	<i>Marker Chord A Mayor</i>
2	 <p>The image shows a hand holding a guitar neck with the B major chord shape. The word 'MAJOR' is written in a stylized font above the hand. Below the hand, the text 'GuitAR Guide' is written in a smaller font.</p>	<i>Marker Chord B Mayor</i>
3	 <p>The image shows a hand holding a guitar neck with the C major chord shape. The word 'MAJOR' is written in a stylized font above the hand. Below the hand, the text 'GuitAR Guide' is written in a smaller font.</p>	<i>Maker Chord C Mayor</i>


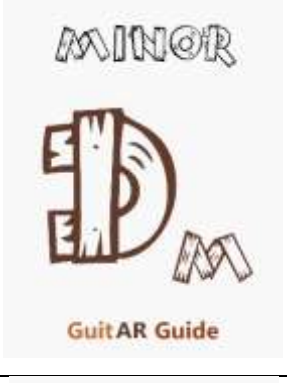


4	 The image shows the word "MAJOR" in a stylized, hand-drawn font at the top. Below it is a large, stylized letter "D" that resembles a piece of wood with several circular holes. At the bottom, the text "GuitAR Guide" is written in a simple, sans-serif font.	<i>Marker Chord D Mayor</i>
5	 The image shows the word "MAJOR" in a stylized, hand-drawn font at the top. Below it is a large, stylized letter "E" that resembles a piece of wood with several circular holes. At the bottom, the text "GuitAR Guide" is written in a simple, sans-serif font.	<i>Marker Chord E Mayor</i>
6	 The image shows the word "MAJOR" in a stylized, hand-drawn font at the top. Below it is a large, stylized letter "F" that resembles a piece of wood with several circular holes. At the bottom, the text "GuitAR Guide" is written in a simple, sans-serif font.	<i>Marker Chord F Mayor</i>
7	 The image shows the word "MAJOR" in a stylized, hand-drawn font at the top. Below it is a large, stylized letter "G" that resembles a piece of wood with several circular holes. At the bottom, the text "GuitAR Guide" is written in a simple, sans-serif font.	<i>Marker Chord G Mayor</i>


3.2.3.2. Perancangan *Marker Chord Minor*

Marker pada *chord* minor dirancang dalam bentuk kartu, yang didalamnya terdapat gambar yang akan dideteksi oleh kamera sehingga aplikasi akan menampilkan model 3D *chord* minor pada *smartphone*. Gambar yang terdapat pada *marker chord* minor adalah huruf dari nama *chord* tersebut, misalnya *chord* A minor, dalam *marker* tersebut terdapat huruf dalam *font* berwarna coklat yaitu “Am”, pada bagian atas terdapat tulisan “MINOR” berwarna hitam dan bagian bawah terdapat tulisan nama aplikasi yaitu “GuitAr Guide”

Tabel 3.22 Tabel *Marker Chord Minor*

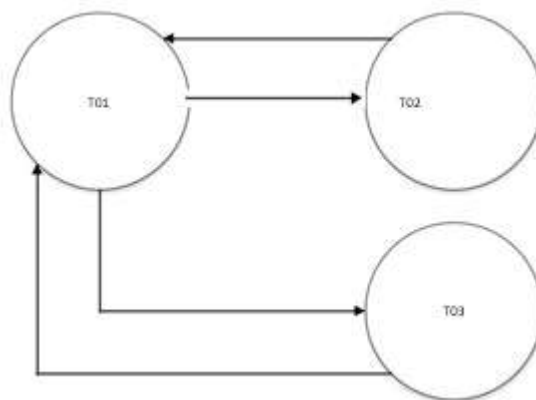
No.	Gambar Marker	Nama
1		<i>Marker Chord A Minor</i>
2		<i>Marker Chord B Minor</i>

3	 <p>MINOR</p> <p>C_m</p> <p>GuitAR Guide</p>	<i>Marker Chord C Minor</i>
4	 <p>MINOR</p> <p>D_m</p> <p>GuitAR Guide</p>	<i>Maker Chord D Minor</i>
5	 <p>MINOR</p> <p>E_m</p> <p>GuitAR Guide</p>	<i>Marker Chord E Minor</i>
6	 <p>MINOR</p> <p>F_m</p> <p>GuitAR Guide</p>	<i>Maker Chord F Minor</i>

7		Marker Chord G Minor
---	---	----------------------

3.2.4. Perancangan Jaringan Semantik

Jaringan semantik ini menunjukkan form-form yang bisa diakses oleh pengguna seperti ditunjukkan pada Gambar 3.30



Gambar 3.30 Jaringan Semantik

Keterangan :

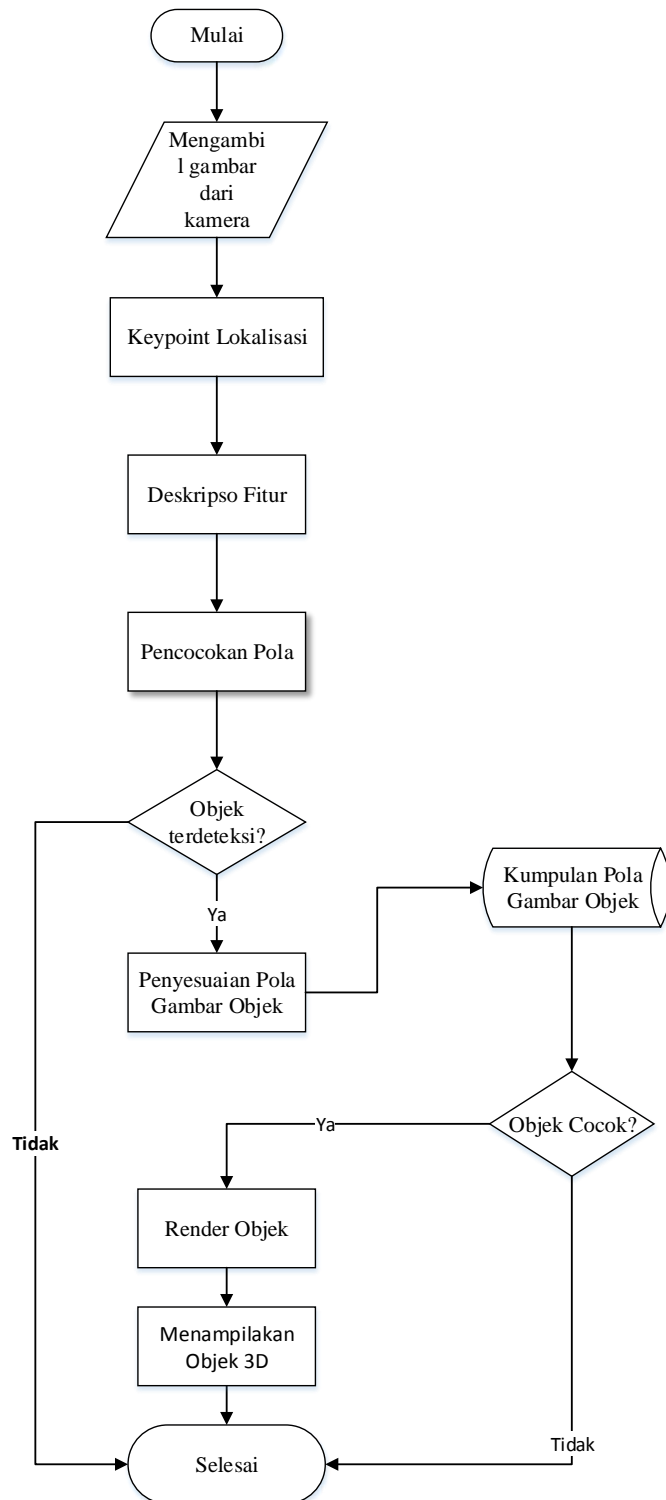
T01 : Menu Utama

T03 : How to use

T02 : Start

3.2.5. Perancangan Method

Perancangan komponen method merupakan perancangan yang dibuat setelah perancangan jaringan semantik dan antarmuka. Perancangan ini berfungsi untuk mendeskripsikan method-method yang berada di dalam aplikasi pengenalan rangka dan bagian tumbuhan menggunakan teknologi augmented reality. Adapun method-method yang terdapat dalam aplikasi yang akan dibangun adalah sebagai berikut :

1. Perancangan *Method Augmented Reality*

Gambar 3.31 *Method Augmented Reality*