BAB 3
ANALISIS DAN PERANCANGAN SISTEM

Tahap analisis dan perancangan merupakan tahap sistematis untuk menyesuaikan kegunaan dan tujuan pada aplikasi. Tahap awal pada tahap analisis dimulai dari analisis masalah, menganalisis sistem, analisis fungsional dan nonfungsional. Sedangkan untuk tahap perancangan dimulai dengan melakukan perancangan sistem yang mencakup perancangan antar muka dan perancangan struktur menu yang nantinya akan digunakan untuk diterapkan pada aplikasi.

3.1 Analisis Sistem

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya yang dimaksudkan 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.

3.1.1 Analisis Masalah

Analisis masalah adalah tahap penjabaran masalah yang ada sebelum aplikasi dibangun dan bertujuan untuk membantu pembangunan aplikasi ini. Analisis masalah yang ada meliputi hal-hal sebagai berikut:

1. Pengenalan kesenian Jawa Barat ke masyarakat masih menggunakan metode konvensional yaitu dengan melakukan pertunjukan seni di beberapa tempat salah satunya di Saung Angklung Udjo. Sehingga orang-orang harus mendatangi tempat tersebut secara langsung.
2. Masih minimnya penggunaan teknologi dalam proses pengenalan kesenian ke masyarakat. Penggunaan teknologi masih hanya sebatas pendataan saja dan dokumentasi kegiatan kesenian yang diselenggarakan. Berdasarkan

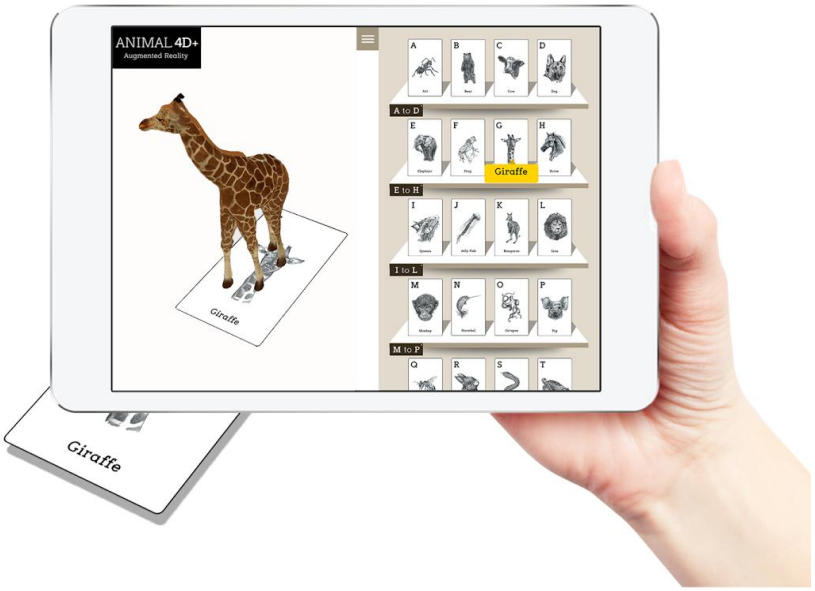
jurnal "Aplikasi AR-Gamelan Sebagai Media Pembelajaran Mengenal Gamelan Jawa Berbasis *Augmented Reality* pada Perangkat Mobile Android", oleh Fata Anshori mengenai pengaplikasian alat musik gamelan ke teknologi AR dan jurnal "Alat Musik Perkusi Augmented Reality Berbasis Android", oleh Mochamad Fathoni, Eko Budi Cahyono, S.Kom, MT, Wahyu Andhyka Kusuma, S.Kom mengenai pengaplikasian alat musik perkusi ke teknologi AR, dapat disimpulkan jika teknologi AR dinilai cocok untuk pengenalan kesenian alat musik, termasuk juga dalam mengenalkan kesenian Jawa Barat ke masyarakat luas. Teknologi AR dapat menampilkan objek kesenian secara 3D dan didukung juga dengan penggunaan animasi dan suara sehingga akan terasa lebih nyata dan menarik.

3.1.2 Analisis Aplikasi Sejenis

Analisis Aplikasi Sejenis merupakan analisis terhadap aplikasi yang sejenis dengan aplikasi yang akan dibangun. Dalam analisis ini akan dibahas mengenai konten apa saja yang terdapat dalam aplikasi tersebut beserta kelebihan dan kekurangannya. Analisis ini bertujuan sebagai gambaran terhadap aplikasi yang akan dibangun dan juga dapat sebagai acuan agar aplikasi yang akan dibangun dapat semaksimal mungkin. Aplikasi yang akan dianalisis adalah *Animal 4D+* dan *AR Concert with Miku.*

3.1.2.1 Animal 4D+

Animal 4D+ merupakan aplikasi pengenalan hewan yang memanfaatkan teknologi AR. Aplikasi ini menggunakan marker berupa kartu hewan sebagai perantara untuk menampilkan animasi 4D ke pengguna sesuai dengan kartu hewan yang disorot dengan kamera pengguna. Fitur-fitur lainnya antara lain adalah fitur untuk menampilkan informasi dari hewan tersebut, dan juga fitur *interaction chain,* fitur ini berfungsi apabila kita menyorot dua kartu secara bersamaan, contohnya kartu gajah dan kartu rumput jika disorot bersamaan maka gajah akan memakan rumput tersebut. Berikut contoh tampilan gambar dari Aplikasi *Animal 4D+* seperti ditunjukkan gambar 3.1



Gambar 3.1 Aplikasi *Animal 4D+*

Kelebihan dari aplikasi *Animal 4D+* adalah sebagai berikut :

1. User Interface dari aplikasi *Animal 4D+* terbilang menarik.
2. Aplikasi tidak hanya memunculkan objek animasi dari masing-masing hewan tapi juga disertai dengan informasi masing-masing hewan tersebut.
3. Terhadap Fitur *Interaction Chain*, sehingga dapat membuat dua objek saling berinteraksi
4. Proses *Scanning Marker* relatif cepat

Sedangkan kekurangan dari aplikasi *Animal 4D+* adalah jumlah hewan yang ditampilkan masih terbatas (hanya satu hewan tiap abjad)

3.1.2.2 AR Concert With Miku

AR Concert With Miku merupakan aplikasi yang menampilkan tarian secara 3D dari Miku. Miku sendiri adalah karakter non fiksi dari Jepang yang berupa seorang penyanyi dan penari wanita. Meski hanya sebuah karakter non fiksi, di jepang sendiri karakter miku sangat terkenal bahkan sering diadakan

konsernya disana. Aplikasi ini menggunakan objek-objek seperti kartu, buku, ataupun lantai sebagai markernya, sehingga aplikasi ini tidak memerlukan marker khusus untuk menampilkan objek 3D. Cara kerja aplikasi ini adalah kita hanya perlu mengarahkan kamera ke objek-objek seperti kartu, buku dll sebagai marker kemudian sistem akan menampilkan objek 3D miku. Objek 3D dapat bergerak sesuai dengan lagu yang akan kita pilih. Berikut contoh tampilan gambar dari *Aplikasi AR Concert With Miku* seperti ditunjukkan gambar 3.2



Gambar 3.2 Aplikasi *AR Concert With Miku*

Kelebihan dari aplikasi *AR Concert with Miku* adalah :

1. Aplikasi tidak memerlukan marker khusus untuk menampilkan objek 3D
2. Gerakan tarian dari miku terlihat *smooth* dan suara yang dihasilkan juga jernih.
3. Terdapat fitur Non AR sehingga tidak perlu lagi melakukan *scanning* marker

Sedangkan kekurangan dari aplikasi *AR Concert with Miku* adalah sebagai berikut:

1. Ukuran aplikasi cukup berat (sekitar 120MB)
2. Jika posisi kamera bergeser dari marker, maka objek akan hilang. Untuk memunculkannya kembali harus melakukan *reset/clear* terlebih dahulu.

Berdasarkan analisis dari kedua aplikasi yang sejenis tersebut dapat diambil kesimpulan bahwa kedua aplikasi tersebut memiliki kelebihan dan kekurangan masing-masing yang dapat digunakan sebagai referensi untuk pembangunan aplikasi AR alat musik dan tarian khas Jawa Barat ini. Adapun kelebihan yang dapat diambil untuk pembangunan aplikasi ini kedepannya adalah sebagai berikut

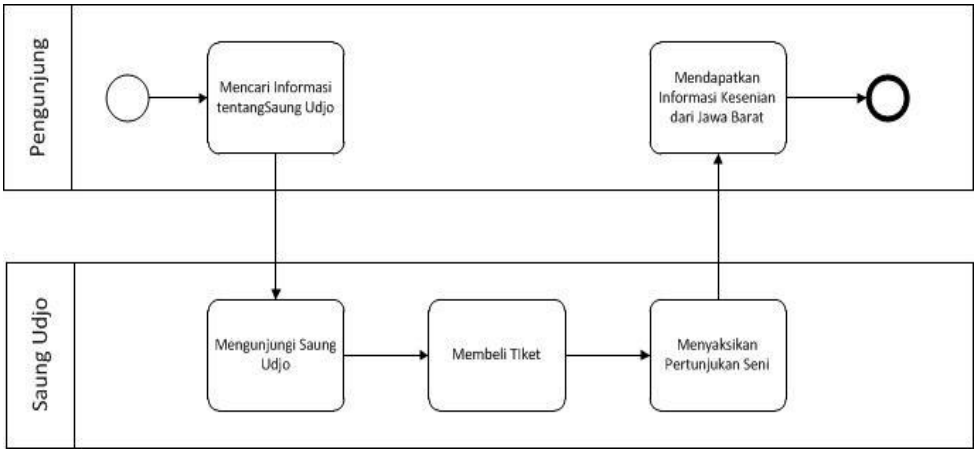
1. *User Interface* aplikasi yang akan dibangun haruslah menarik dan User Friendly tentunya
2. Aplikasi tidak hanya menampilkan objek 3D saja tetapi juga dilengkapi dengan informasi mengenai objek yang ditampilkan
3. Proses *Scanning Marker* harus relatif cepat
4. Gerakan dari objek yang ditampilkan harus lancar dan suara yang dihasilkan juga jernih.
5. Ukuran aplikasi diharapkan dapat seringan mungkin.

3.1.3 Analisis Sistem yang Sedang Berjalan

Analisis sistem yang sedang berjalan merupakan kegiatan menganalisis prosedur-prosedur kerja yang terjadi pada sistem yang ada saat ini. Prosedur kerja yang dimaksud adalah proses pengenalan kesenian Jawa Barat ke masyarakat. Dikarenakan pengenalan aplikasi ini ke masyarakat akan dilakukan di Saung Angklung Udjo, maka kita ambil proses bisnis dari Saung Angklung Udjo sebagai pembanding. Berikut adalah proses bisnis dari prosedur yang saat ini diterapkan oleh Saung Angklung Udjo seperti pada Gambar 3.3. Adapun alur prosesnya adalah sebagai berikut :

1. Pengunjung terlebih dahulu mencari informasi mengenai Saung Angklung Udjo
2. Pengunjung mendatangi Saung Angklung Udjo
3. Pengunjung Membeli tiket masuk
4. Pengunjung melihat-lihat kesenian dari Jawa Barat yang berada di Saung Angklung Udjo dan menyaksikan pagelaran kesenian jika ada.

5. Pengunjung mendapatkan informasi mengenai kesenian dari Jawa Barat

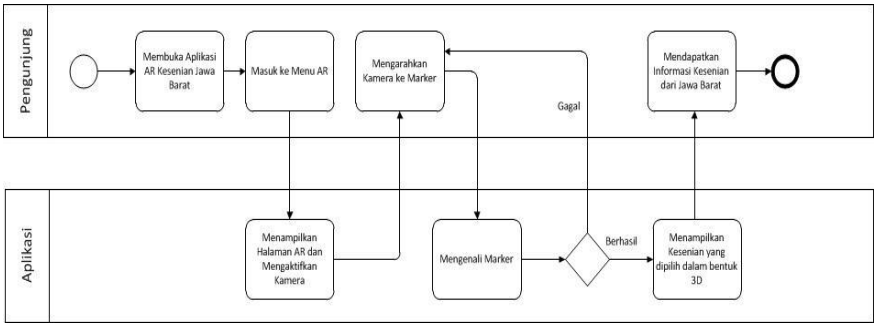


Gambar 3.3 *BPMN* Proses Bisnis dari Saung Angklung Udjo

3.1.4 Analisis Aplikasi yang Akan Dibangun

Aplikasi yang akan dibangun ini menggunakan teknologi *Augmented Reality*. Teknologi AR ini memerlukan sebuah marker sebagai media perantara untuk memunculkan gambar berupa animasi 3D. Proses bisnisnya dapat dilihat pada Gambar 3.4. Adapun alurnya sebagai berikut :

1. Pengunjung membuka aplikasi *Augmented Reality* Seni Tari dan Alat Musik Jawa Barat
2. Pengunjung masuk ke salah satu menu AR
3. Aplikasi akan menampilkan halaman AR yang dipilih sekaligus mengaktifkan kamera
4. Pengunjung mengarahkan kamera ke marker dan memilih kesenian yang ingin ditampikan
5. Aplikasi akan mengenali pola marker, jika cocok maka aplikasi akan menampikan kesenian sesuai marker yang diarahkan ke kamera.



Gambar 3.4 *BPMN* Proses Bisnis dari Aplikasi yang akan dibangun

3.1.5 Spesifikasi Kebutuhan Perangkat Lunak

Perancangan sistem dan perangkat lunak melibatkan identifikasi dan deksripsi abstarksi sistem perangkat lunak yang mendasar dan hubungan-hubungannya. Tahap perancangan fungsional, dan perancangan non-fungsional. Berdasarkan aturan bisnis yang ada, spesifikasi kebutuhan perangkat lunak dari sistem yang akan dibangun dijelaskan dalam Tabel 3.1.

Tabel 3.1 Spesifikasi Kebutuhan Perangkat Lunak Fungsional

No.	Kode Kebutuhan	Deskripsi Kebutuhan
1	SKPL-F001	Aplikasi dapat membaca marker
2	SKPL-F002	Aplikasi dapat mendeteksi marker
3	SKPL-F003	Aplikasi dapat menampilkan objek 3D
4	SKPL-F004	Aplikasi dapat mengeluarkan audio
5	SKPL-F005	Aplikasi dapat menampilkan informasi seni tari dan musik dari Jawa Barat

Analisis spesifikasi kebutuhan perangkat lunak non fungsional dapat dilihat pada tabel 3.2

Tabel 3.2 Spesifikasi Kebutuhan Perangkat Lunak Non Fungsional

No.	Kode Kebutuhan	Deskripsi Kebutuhan
1	SKPL-NF001	Aplikasi memerlukan kamera dengan resolusi yang cukup untuk mengenali marker
2	SKPL-NF002	Aplikasi memerlukan cahaya yang cukup untuk

		mendeteksi marker
3	SKPL-NF003	Aplikasi memerlukan sudut yang cukup untuk membaca marker

3.1.6 Analisis Kebutuhan Non Fungsional

Analisis Kebutuhan Non-Fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Adapun kebutuhan non-fungsional untuk membangun aplikasi ini antara lain meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan penggunaan sistem yang akan memakai aplikasi. Analisis kebutuhan non-fungsional bertujuan agar aplikasi yang akan dibangun dapat digunakan sesuai dengan kebutuhan.

3.1.6.1 Analisis Kebutuhan Perangkat Keras

Analisis Perangkat keras atau *hardware* merupakan salah satu hal yang penting karena tanpa *hardware* yang memenuhi syarat, aplikasi yang akan dibuat tidak akan dapat berjalan. Analisis kebutuhan perangkat keras ini dibagi menjadi dua bagian, diantaranya seperti:

A. Analisis Kebutuhan Perangkat Keras Pembangunan Aplikasi

Kebutuhan perangkat keras yang digunakan untuk membangun aplikasi ini dapat dilihat pada Tabel 3.3.

Tabel 3.3 Analisis Kebutuhan Perangkat Keras Pembuatan Aplikasi

No.	Perangkat Keras	Spesifikasi
1	Processor	AMD Phenom II X4 BE 3,2 Ghz
2	Monitor	Monitor Resolusi 1920x1080
3	VGA	VGA 1GB
4	Harddisk	2 TB
5	Memory	RAM 4GB

6	Mouse	Standar
7	Keyboard	Standar
8	Speaker	Standar

B. Analisis Kebutuhan Perangkat Keras Pengguna

Kebutuhan perangkat keras Android yang digunakan untuk menjalankan aplikasi ini dapat dilihat pada Tabel 3.4

Tabel 3.4 Analisis Kebutuhan Perangkat Keras Untuk User

No.	Perangkat Keras	Spesifikasi
1	Processor	1,5 Ghz
2	Monitor	Touchscreen
3	Memory RAM	2 GB
4	Camera	8 MP

3.1.6.2 Analisis Kebutuhan Perangkat Lunak

Analisis perangkat lunak atau *software* merupakah 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. Analisis kebutuhan perangkat lunak ini dibagi menjadi dua bagian, diantaranya seperti :

A. Analisis Kebutuhan Perangkat Lunak Pembangunan Aplikasi

Kebutuhan perangkat keras yang digunakan untuk membangun aplikasi ini dapat dilihat pada Tabel 3.5

Tabel 3.5 Analisis Kebutuhan Perangkat Lunak Pembuatan Aplikasi

No.	Perangkat Lunak	Kegunaan
1	Windows 10 64 bit	Sebagai Sistem Operasi dari komputer yang digunakan
2	Unity 3D 5.1	Digunakan untuk membuat objek 3D
3	Vuforia Android SDK	Digunakan sebagai library pendukung AR

4	Adobe Photoshop CS 6	Digunakan untuk desain gambar marker
5	StarUML 2.0	Digunakan untuk desain UML

B. Analisis Kebutuhan Perangkat Lunak Pengguna

Kebutuhan perangkat keras Android yang digunakan untuk menjalankan aplikasi ini dapat dilihat pada Tabel 3.6

Tabel 3.6 Analisis Kebutuhan Perangkat Lunak Untuk User

No.	Perangkat Lunak	Spesifikasi
1	Sistem Operasi Android	OS Android 4.0

3.1.6.3 Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna merupakan analisis terhadap pengguna yang akan menggunakan aplikasi yang akan dibangun. Adapun kebutuhan pengguna yang terlibat dalam penggunaan Aplikasi ini dapat dilihat pada tabel 3.7

Tabel 3.7 Analisis Kebutuhan Pengguna

Pengguna	Hak Akses	Pengalaman Pengguna
Pengunjung	Menggunakan fitur aplikasi	Dapat Menggunakan Smartphone Android

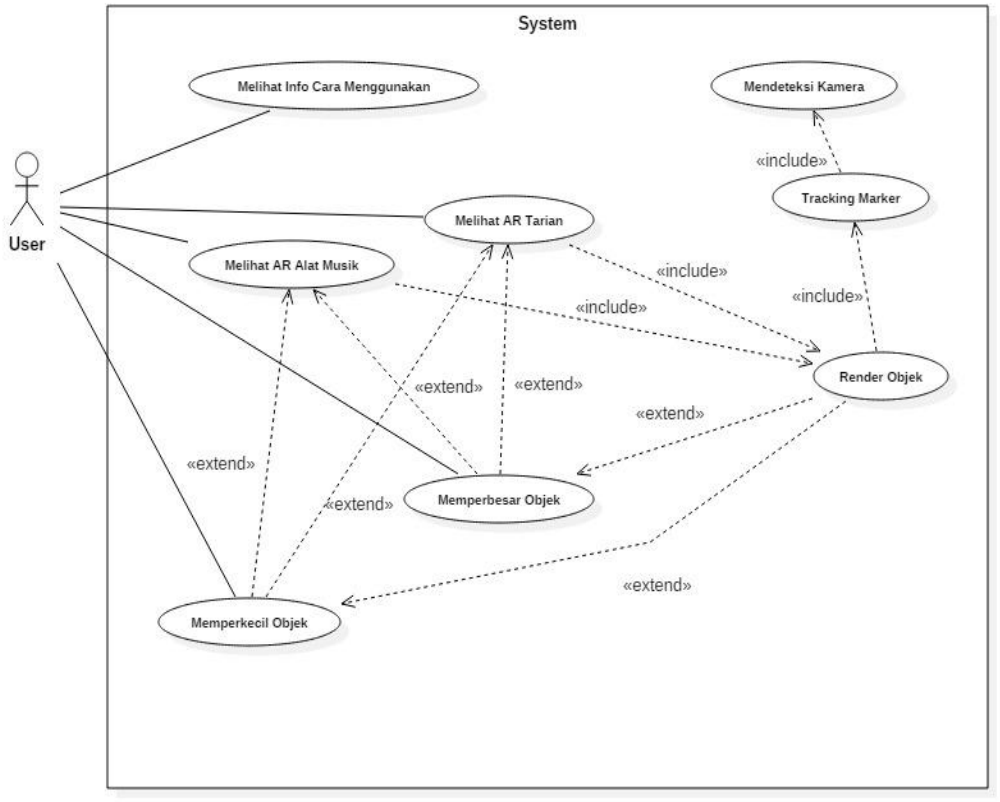
3.1.7 Analisis Kebutuhan Fungsional

Dalam analisis kebutuhan fungsional ini, digunakan konsep Object Oriented Programing. Analisis kebutuhan fungsional ini dimodelkan dengan menggunakan UML (*Unified Modeling Language*). Diagram yang digunakan yaitu *Usecase Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.

3.1.7.1 Use Case Diagram

Use case adalah konstruksi untuk mendeskripsikan bagaimana sistem terlihat di mata pengguna. Sasaran pemodelan *use case* diantaranya adalah

mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan



Gambar 3.5 *Use Case Diagram*

3.1.7.2 Definisi Aktor

Definisi aktor untuk menjelaskan aktor yang terdapat pada *Use Case Diagram*. Definisi aktor dijelaskan pada tabel berikut:

Tabel 3.8 Definisi aktor

No	Aktor	Deskripsi
1	User	Orang yang menggunakan aplikasi

3.1.7.3 *Use Case Identification*

Berikut table indetifikasi use case yang terdapat dalam aplikasi :

Tabel 3.9 Identifikasi Use Case

No	Use Case	Deskripsi
1	Melihat Info cara menggunakan	Fungsionalitas untuk menampilkan info cara menggunakan
2	Melihat AR Alat Musik	Fungsionalitas untuk melihat objek 3D berupa alat musik
3	Melihat AR Alat Tarian	Fungsionalitas untuk melihat objek 3D berupa tarian
4	Mendeteksi kamera	Fungsionalitas mendeteksi keberadaan kamera
5	Tracking Marker	Fungsionalitas untuk mendeteksi keberadaan atau mengenali marker
6	Render Objek	Fungsionalitas untuk menampilkan objek ke layar
7	Memperbesar Objek	Fungsionalitas untuk memperbesar ukuran objek
8	Memperkecil Objek	Fungsionalitas untuk memperkecil ukuran objek

3.1.7.4 *Use Case Scenario*

Skenario *Use Case* berfungsi untuk menjelaskan lebih detail mengenai modul-modul yang terdapat dalam aplikasi. Adapun penjelasan mengenai keterangan tabel skenario *Use Case* sebagai berikut :

1. Skenario *Use Case* Melihat Info cara menggunakan

Tabel 3.10 Skenario *Use Case* Melihat Info cara menggunakan

Identifikasi	
Nomor	UCS-01
Nama	Cara menggunakan
Tujuan	Menampilkan info cara menggunakan aplikasi
Deskripsi	Proses untuk menampilkan info cara menggunakan aplikasi
Aktor	User
Skenario Utama	
Kondisi awal	User berada pada menu utama
Reaksi aktor	**Reaksi sistem**
1. User berada pada menu utama	

Identifikasi	
	2. Menampilkan info cara menggunakan Aplikasi
Kondisi Akhir	Aplikasi menampilkan info cara menggunakan aplikasi

2. Skenario *Use Case* Melihat AR Alat Musik

Tabel 3.11 Skenario *Use Case* Melihat AR Alat Musik

Identifikasi	
Nomor	UCS-02
Nama	Melihat AR Alat Musik
Tujuan	Menampilkan Objek 3D berupa alat musik
Deskripsi	Proses untuk menampilkan Objek 3D berupa alat musik
Aktor	User
Skenario Utama	
Kondisi awal	User berada pada menu utama
Reaksi aktor	**Reaksi sistem**
1. User Memilih Menu AR Alat Musik	
	2. Deteksi kamera, jika terdeteksi maka kamera aktif
3. Kamera aktif	
4. User mengarahkan kamera ke marker	
5. Kamera menangkap gambar	
	6. Deteksi marker, jika terdeteksi maka render objek
	7. Menampilkan objek 3D alat musik
Kondisi Akhir	Menampilkan objek 3D alat musik
Alternatif	Keputusan
	2.1 Kamera tidak terdeteksi
	4.1 Marker tidak terdeteksi

3. Skenario *Use Case* Melihat AR Tarian

Tabel 3.12 Skenario *Use Case* Melihat AR Tarian

Identifikasi	
Nomor	UCS-03
Nama	Melihat AR Tarian
Tujuan	Menampilkan Objek 3D berupa tarian

Identifikasi	
Deskripsi	Proses untuk menampilkan Objek 3D berupa tarian
Aktor	User
Skenario Utama	
Kondisi awal	User berada pada menu utama
Reaksi aktor	**Reaksi sistem**
1. User Memilih Menu AR Alat Musik	
	2. Deteksi kamera, jika terdeteksi maka kamera aktif
3. Kamera aktif	
4. User mengarahkan kamera ke marker	
5. Kamera menangkap gambar	
	6. Deteksi marker, jika terdeteksi maka render objek
	7. Menampilkan objek 3D Tarian
Kondisi Akhir	Menampilkan objek 3D Tarian
Alternatif	Keputusan
	2.1 Kamera tidak terdeteksi
	4.1 Marker tidak terdeteksi

4. Skenario *Use Case* Mendeteksi Kamera

Tabel 3.13 Skenario *Use Case* Mendeteksi Kamera

Identifikasi	
Nomor	UCS-04
Nama	Deteksi Kamera
Tujuan	Mengaktifkan kamera untuk menangkap gambar
Deskripsi	Proses untuk mengaktifkan kamera
Aktor	User
Skenario Utama	
Kondisi awal	User memilih salah satu Menu AR
Reaksi aktor	**Reaksi sistem**
1. User memilih salah satu Menu AR	
	2. Deteksi kamera, jika terdeteksi maka kamera aktif
3. Kamera aktif	
Kondisi Akhir	Kamera dalam kondisi aktif
Alternatif	Keputusan
	2.1 Kamera tidak terdeteksi

5. Skenario *Use Case* Tracking Marker

Tabel 3.14 Skenario *Use Case* Tracking Marker

Identifikasi	
Nomor	UCS-05
Nama	Tracking marker
Tujuan	Mendeteksi keberadaan marker
Deskripsi	Sistem mendeteksi keberadaan marker melalui gambar yang ditangkap oleh kamera
Aktor	User
Skenario Utama	
Kondisi awal	Kamera terdeteksi
Reaksi aktor	**Reaksi sistem**
1. User mengarahkan kamera ke marker	
2. Kamera menangkap gambar	
	3. Sistem mencocokkan pola, jika cocok marker terdeteksi
	4. Marker terdeteksi
Kondisi Akhir	Marker terdeteksi
Alternatif	Keputusan
	3.1 Marker tidak terdeteksi

6. Skenario *Use Case* Render Objek

Tabel 3.15 Skenario Use Case Render Objek

Identifikasi	
Nomor	UCS-06
Nama	Render Objek
Tujuan	Menampilkan Objek di *smartphone*
Deskripsi	Proses untuk menampilkan objek di *smartphone*
Aktor	User
Skenario Utama	
Kondisi awal	Kamera terdeteksi
Reaksi aktor	**Reaksi sistem**
1. User mengarahkan kamera ke marker	
2. Kamera menangkap gambar	
	3. Sistem mencocokkan pola, jika cocok marker terdeteksi
	4. Marker terdeteksi
	5. Menampilkan objek 3D

Kondisi Akhir	Menampilkan objek 3D
Alternatif	Keputusan
	3.1 Marker tidak terdeteksi

7. Skenario *Use Case* Memperbesar Objek

Tabel 3.16 Skenario *Use Case* Memperbesar Objek

Identifikasi	
Nomor	UCS-08
Nama	Memperbesar Objek
Tujuan	Memperbesar ukuran objek
Deskripsi	Proses untuk memperbesar ukuran objek
Aktor	User
Skenario Utama	
Kondisi awal	Objek 3D tampil
Reaksi aktor	**Reaksi sistem**
1. User mengusap layar ke arah luar	
	2. Objek bertambah besar
Kondisi Akhir	Ukuran objek menjadi lebih besar dari pada ukuran sebelumnya

8. Skenario *Use Case* Memperkecil Objek

Tabel 3.17 Skenario *Use Case* Memperkecil Objek

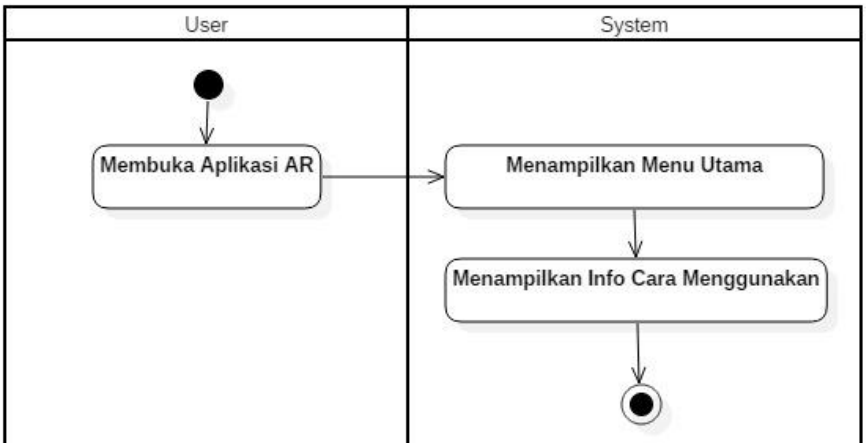
Identifikasi	
Nomor	UCS-09
Nama	Memperkecil Objek
Tujuan	Memperkecil ukuran objek
Deskripsi	Proses untuk memperkecil ukuran objek
Aktor	User
Skenario Utama	
Kondisi awal	Objek 3D tampil
Reaksi aktor	**Reaksi sistem**
1. User mengusap layar ke arah dalam	
	2. Objek bertambah kecil
Kondisi Akhir	Ukuran objek menjadi lebih kecil dari pada ukuran sebelumnya

3.1.7.5 *Activity Diagram*

Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi dan bagaimana mereka berakhir. Penggambaran *activity diagram* memodelkan *event-event* yang terjadi pada *Use Case* dan digunakan untuk pemodelan aspek dinamis dari aplikasi. *Activity Diagram* dimodelkan seperti gambar sebagai berikut :

1. *Activity Diagram* Melihat Info cara menggunakan

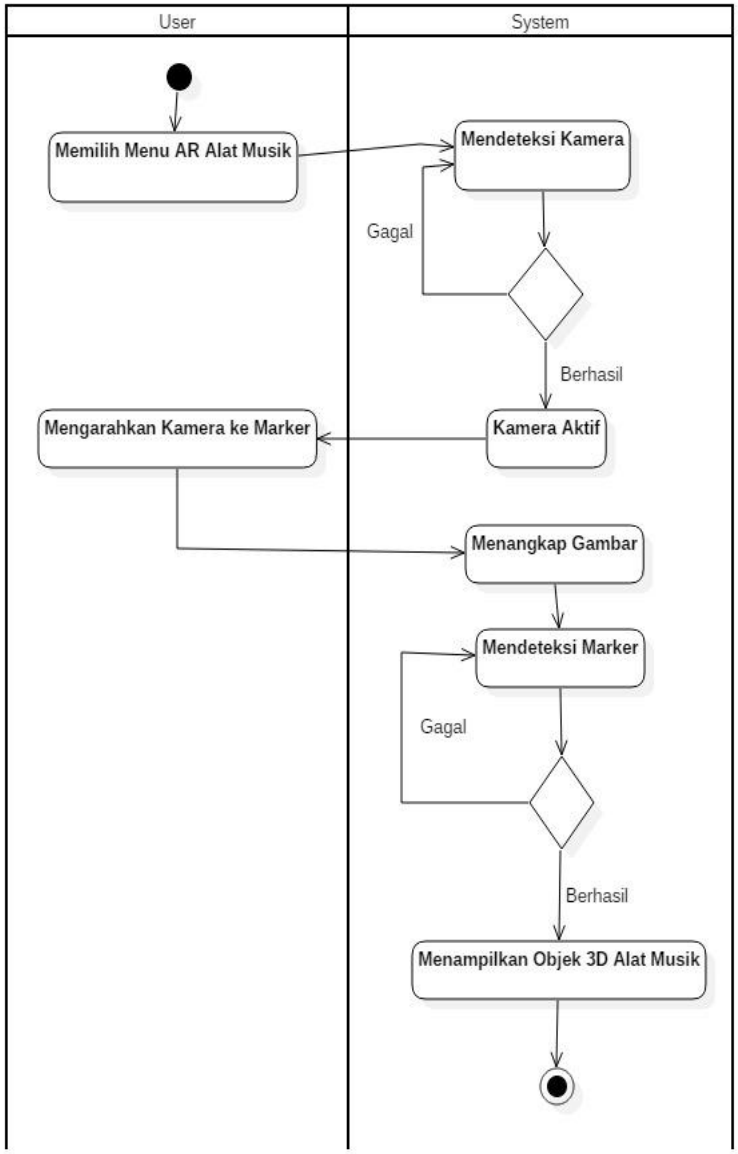
   User membuka aplikasi AR, kemudian sistem akan menampilkan menu utama yang berisi info cara menggunakan aplikasi.



Gambar 3.6 *Activity Diagram* Melihat Info cara menggunakan

2. *Activity Diagram* Melihat AR Alat Musik

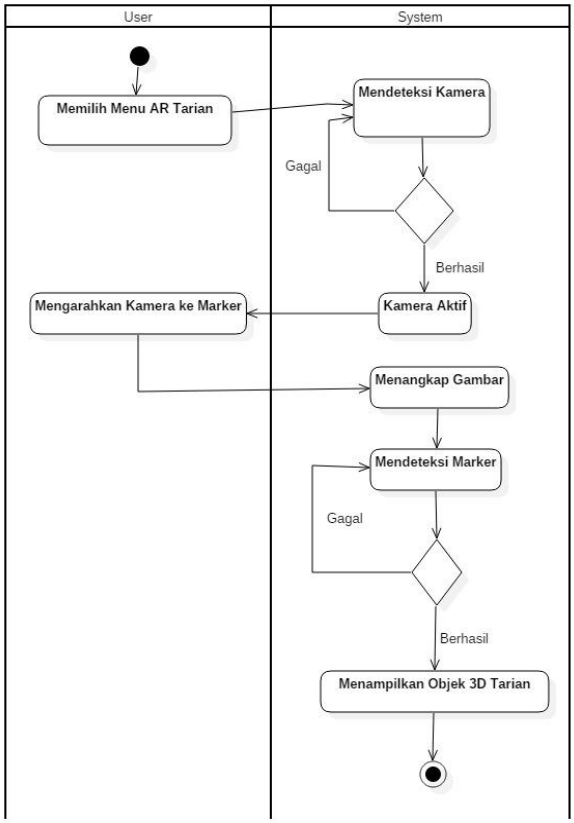
   User memilih menu AR Alat Musik kemudian sistem akan merespon dengan menampilkan objek 3D alat musik.



Gambar 3.7 *Activity Diagram* Melihat AR Alat Musik

3. *Activity Diagram* Melihat AR Tarian

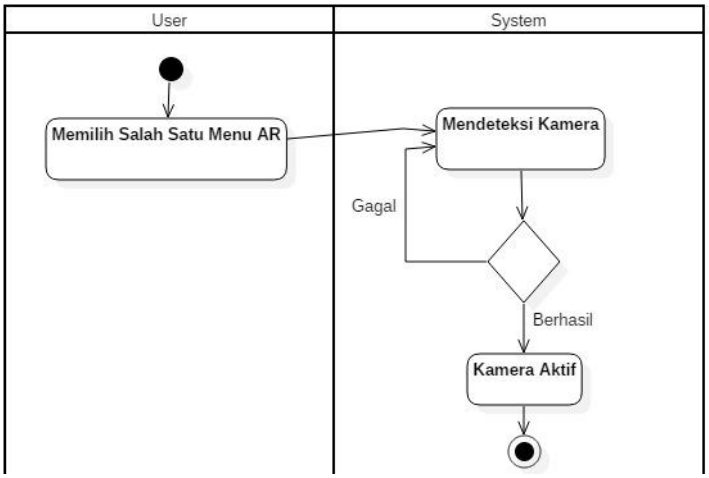
   User memilih menu AR Tarian kemudian sistem akan merespon dengan menampilkan objek 3D tarian.



Gambar 3.8 *Activity Diagram* Melihat AR Tarian

4. *Activity Diagram* Mendeteksi kamera

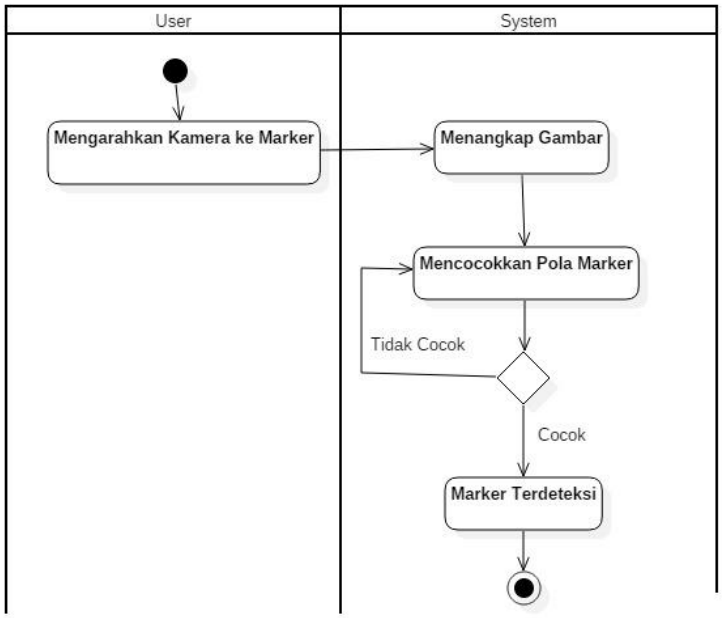
   Setelah *user* memilih salah satu menu AR kemudian sistem akan merespon dengan mendeteksi ketersediaan kamera.



Gambar 3.9 *Activity Diagram* Mendeteksi Kamera

5. *Activity Diagram* Tracking Marker

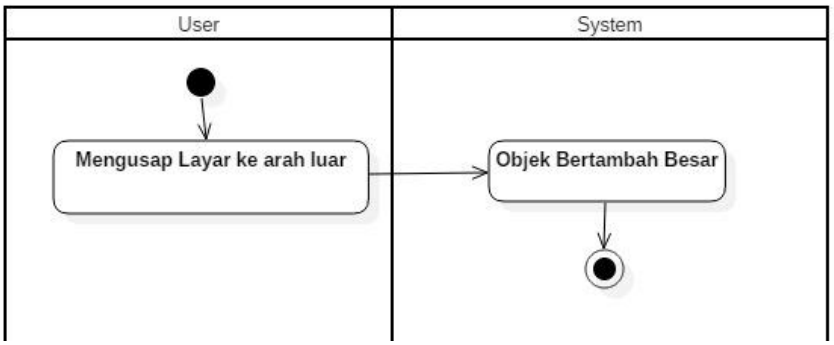
   Setelah kamera terdeteksi, *user* mengarahkan kamera ke marker, maka kamera akan menangkap gambar marker dan selanjutnya diproses oleh sistem.



Gambar 3.10 *Activity Diagram* Tracking Marker

6. *Activity Diagram* Memperbesar Objek

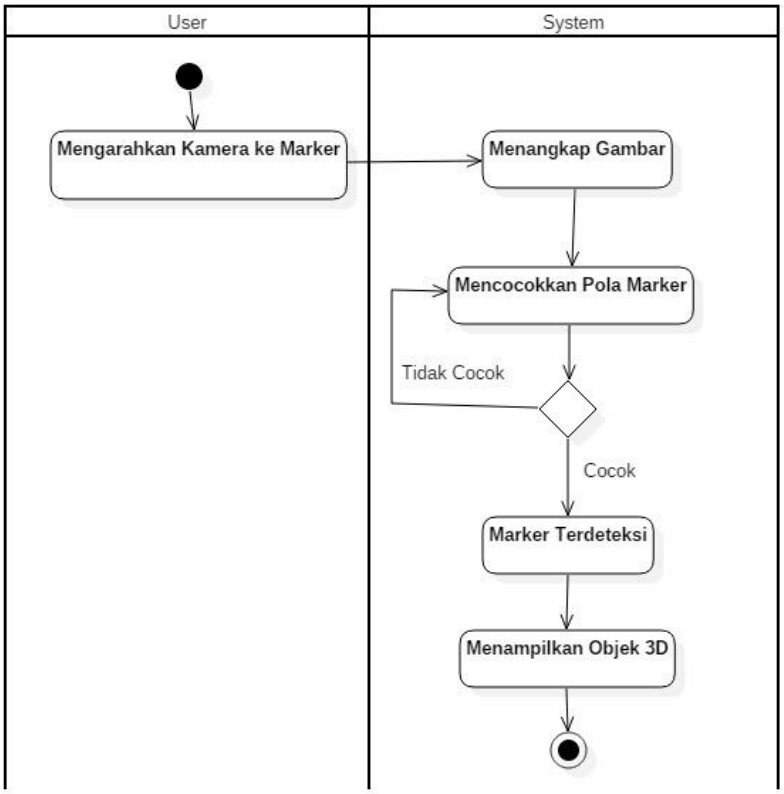
   User dapat memperbesar ukuran objek dengan cara mengusap layar ke arah luar



Gambar 3.11 *Activity Diagram* Memperbesar Objek

7. *Activity Diagram Render* Objek

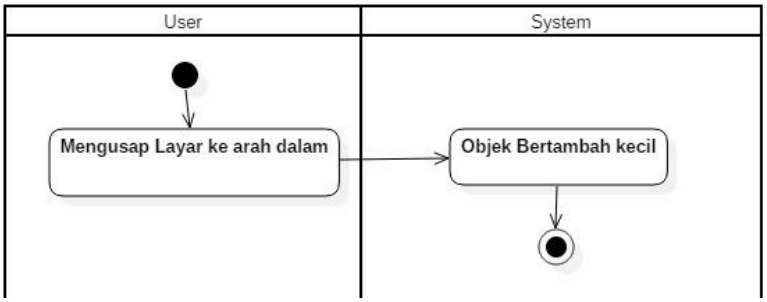
   Setelah melakukan *tracking marker*, sistem akan merender objek dari marker yang sudah terdeteksi.



Gambar 3.12 *Activity Diagram Render* Objek

8. *Activity Diagram* Memperkecil Objek

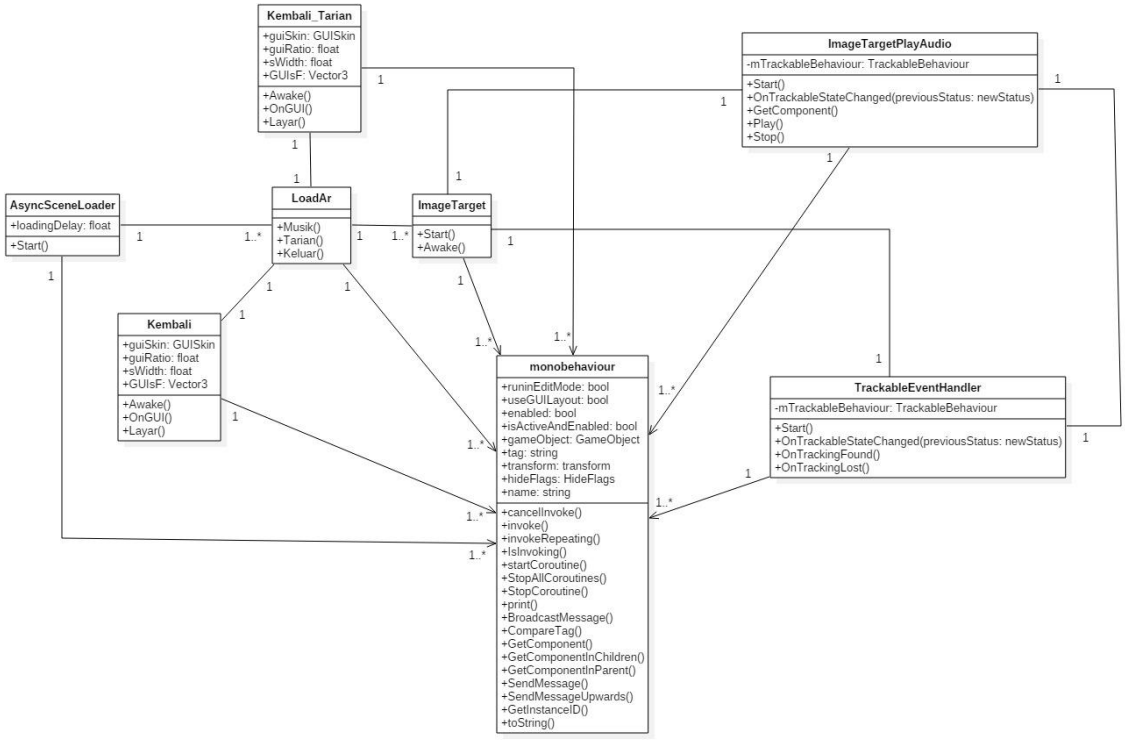
   User dapat memperkecil ukuran objek dengan cara mengusap layar ke arah luar



Gambar 3.13 *Activity Diagram* Memperkecil Objek

3.1.7.6 Class Diagram

Class diagram digunakan untuk memberi gambaran suatu aplikasi. Dapat dilihat dari kelas-kelas yang ada dan cara kerja relasinya. *Class diagram* yang dibuat dapat dilihat pada gambar sebagai berikut :



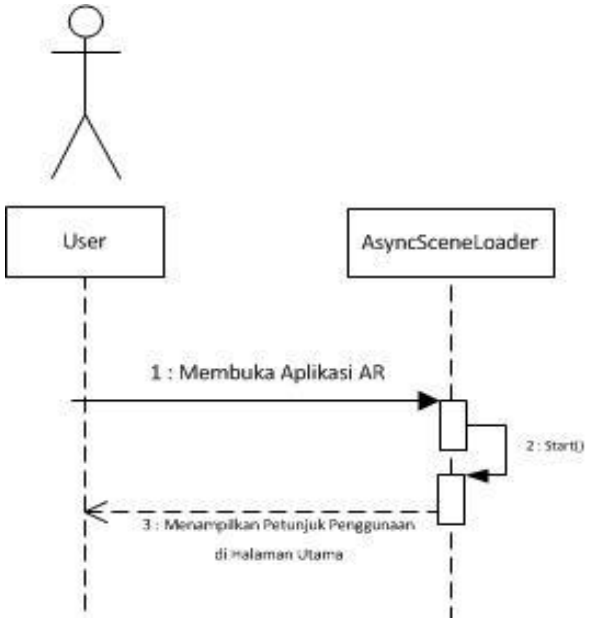
Gambar 3.14 *Class Diagram*

3.1.7.7 Sequence Diagram

Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar objek dan mengindikasikan komunikasi diantara objek-objek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh objek-objek yang melakukan suatu tugas atau aksi-aksi tertentu.

1. *Sequence Diagram* Melihat Info Cara Menggunakan

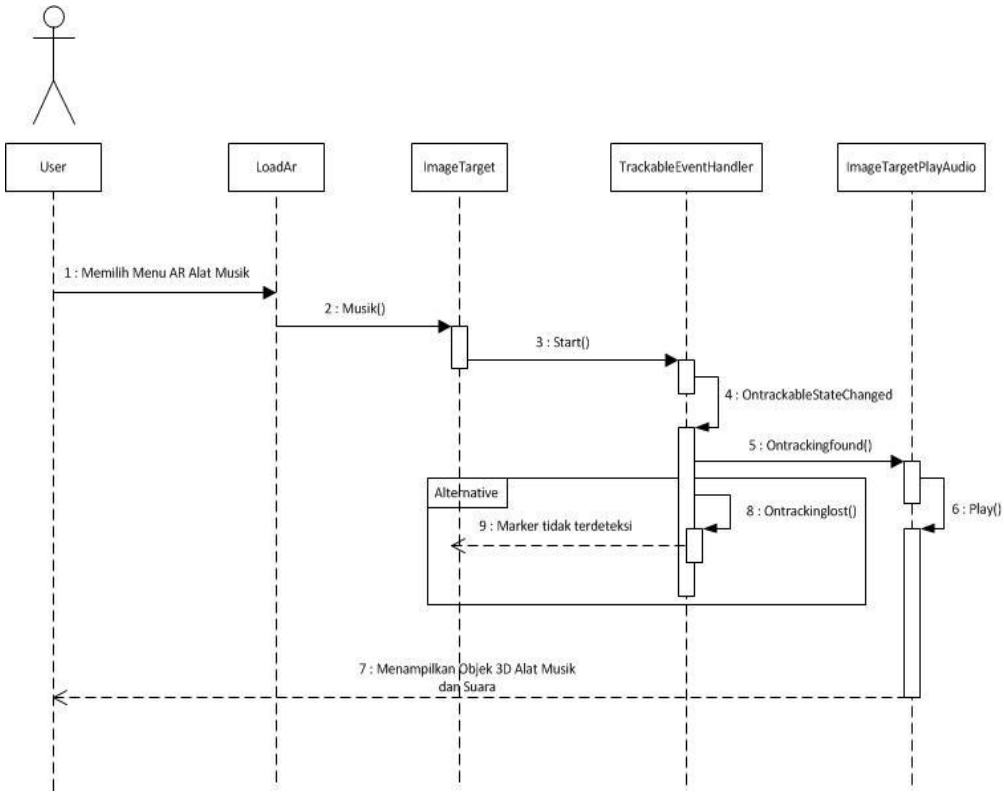
   Sequence Diagram Melihat Info Cara Menggunakan adalah sebagai berikut :



Gambar 3.15 *Sequence Diagram* Melihat Info Cara Menggunakan

2. *Sequence Diagram* Melihat AR Musik

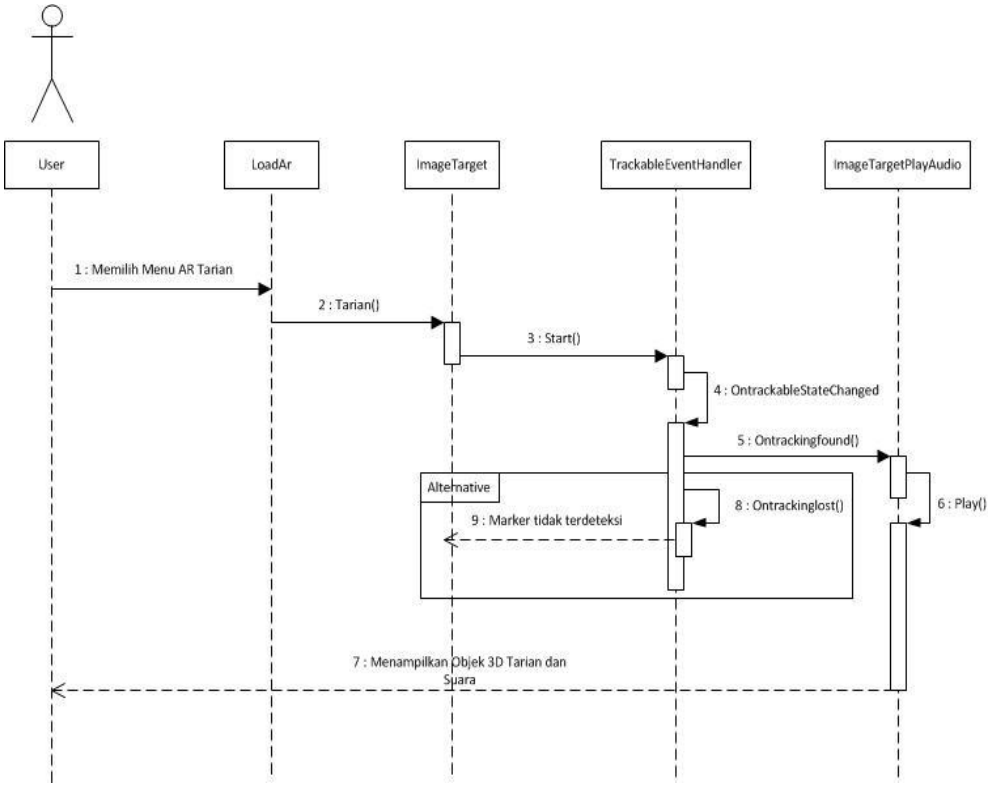
   Sequence Diagram Melihat AR Musik adalah sebagai berikut :



Gambar 3.16 *Sequence Diagram* Melihat AR Alat Musik

3. *Sequence Diagram* Melihat AR Tarian

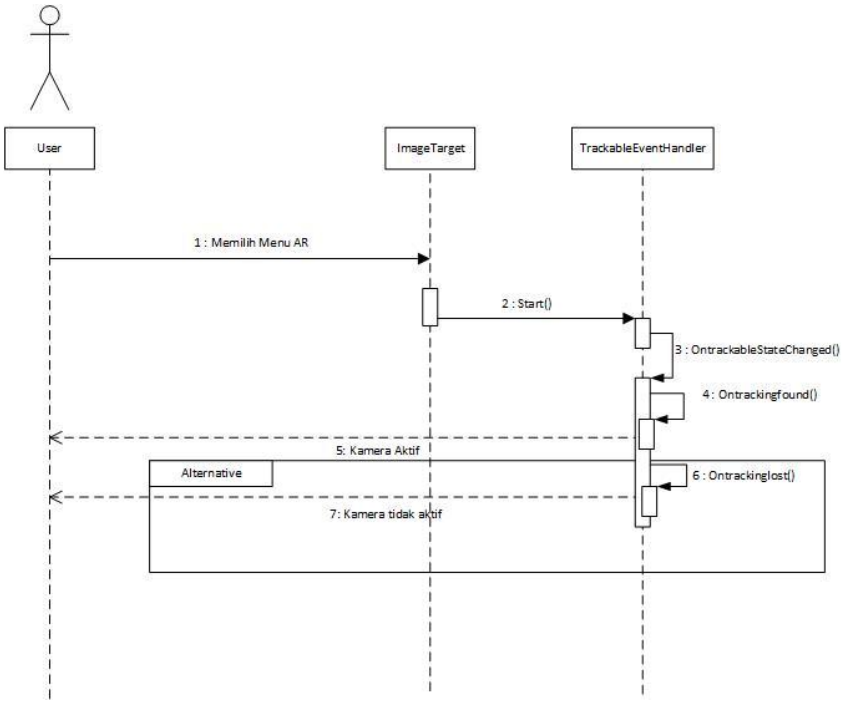
   Sequence Diagram Melihat AR Tarian adalah sebagai berikut :



Gambar 3.17 *Sequence Diagram* Melihat AR Tarian

4. *Sequence Diagram* Mendeteksi Kamera

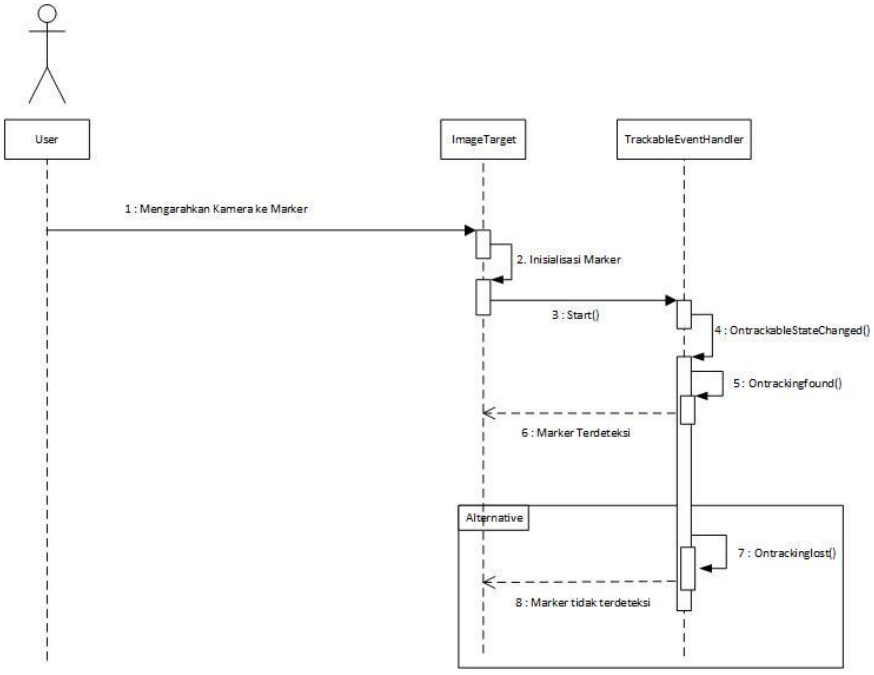
   Sequence Diagram Mendeteksi Kamera adalah sebagai berikut :



Gambar 3.18 *Sequence Diagram* Mendeteksi Kamera

5. *Sequence Diagram Tracking Marker*

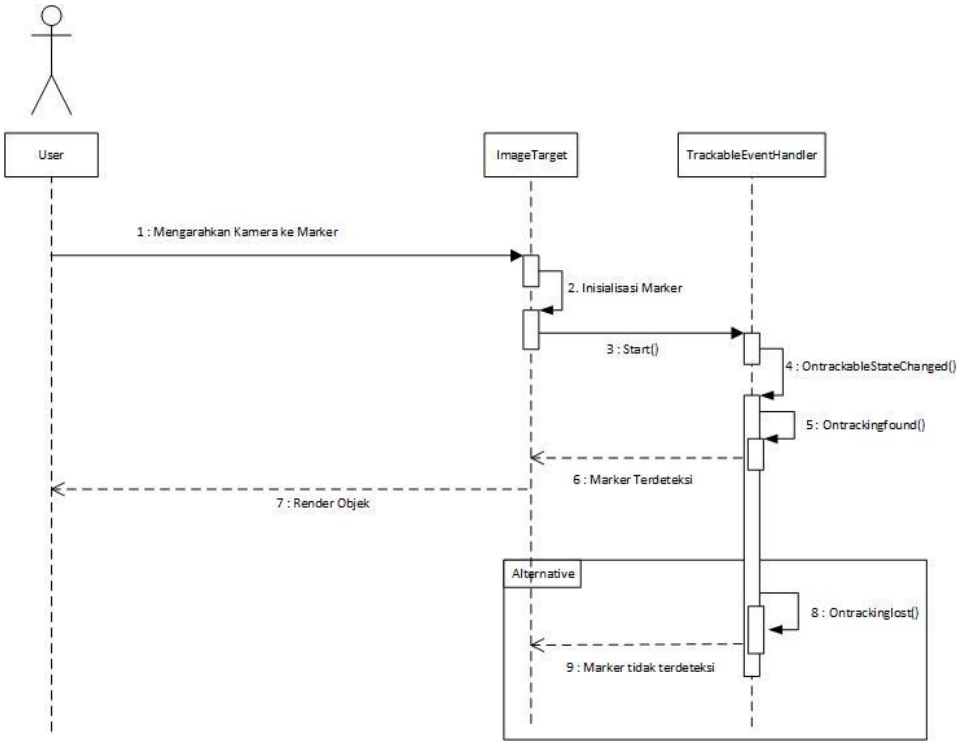
   Sequence Diagram Tracking Marker adalah sebagai berikut :



Gambar 3.19 *Sequence Diagram Tracking Marker*

6. *Sequence Diagram Render* Objek

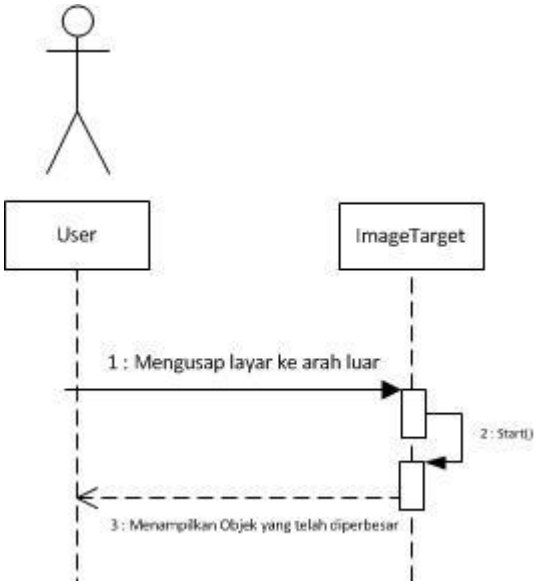
   Sequence Diagram Render Objek adalah sebagai berikut :



Gambar 3.20 *Sequence Diagram* Render Objek

7. *Sequence Diagram* Perbesar Objek

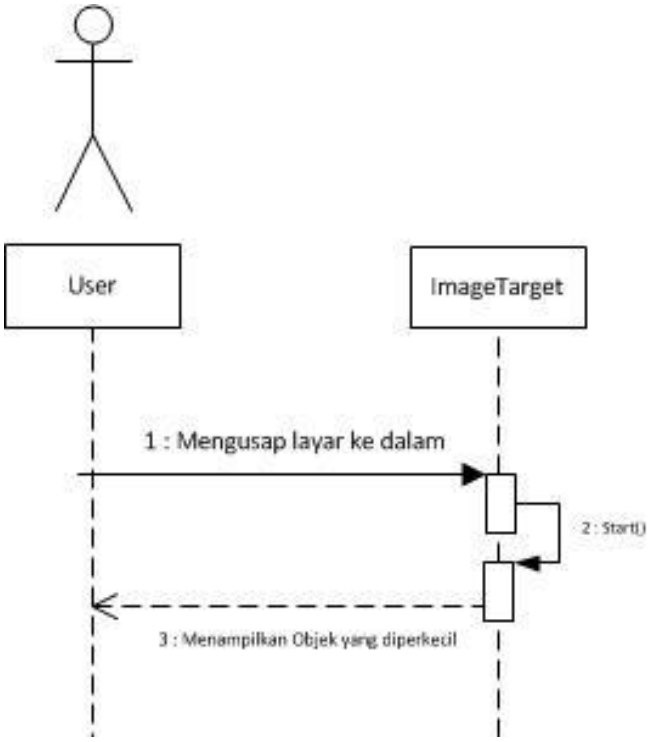
   Sequence Diagram Perbesar Objek adalah sebagai berikut :



Gambar 3.21 *Sequence Diagram* Perbesar Objek

8. *Sequence Diagram* Perkecil Objek

   Sequence Diagram Perkecil Objek adalah sebagai berikut :



Gambar 3.22 *Sequence Diagram* Perkecil Objek

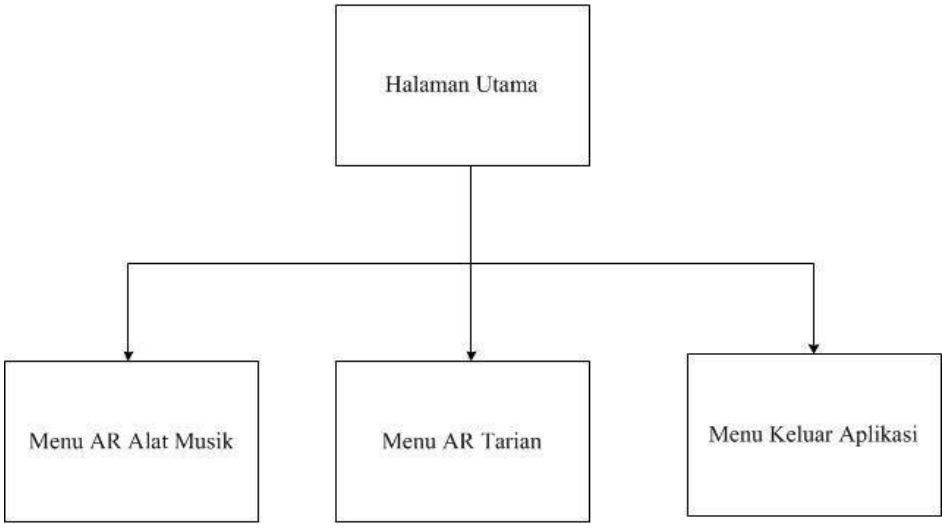
3.2 Perancangan Sistem

Perancangan adalah suatu bagian dari metodologi pembangunan suatu perangkat lunak yang dilakukan setelah tahapan analisis untuk memberikan gambaran secara terperinci. Untuk dapat melakukan perancangan sistem maka dilakukan beberapa langkah yaitu sebagai berikut:

1. Perancangan Struktur Menu
2. Perancangan Antarmuka
3. Perancangan Database Marker
4. Jaringan Semantik

3.2.1 Perancangan Struktur Menu

Struktur Menu adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan aplikasi. Berikut ini adalah perancangan struktur menu Aplikasi yang dapat dilihat pada gambar 3.23



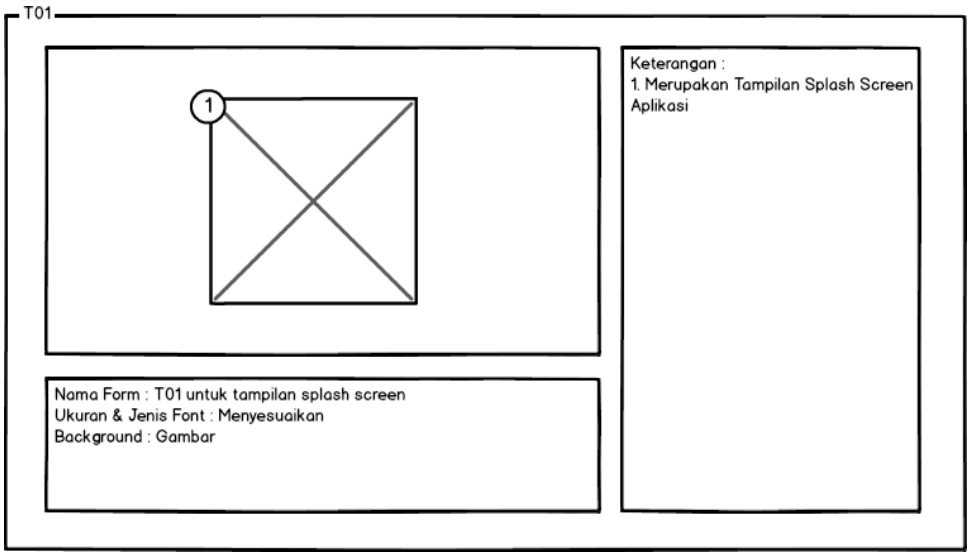
Gambar 3.23 Struktur Menu Aplikasi

3.2.2 Perancangan Antarmuka

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

1. **Perancangan Antarmuka Splash Screen**

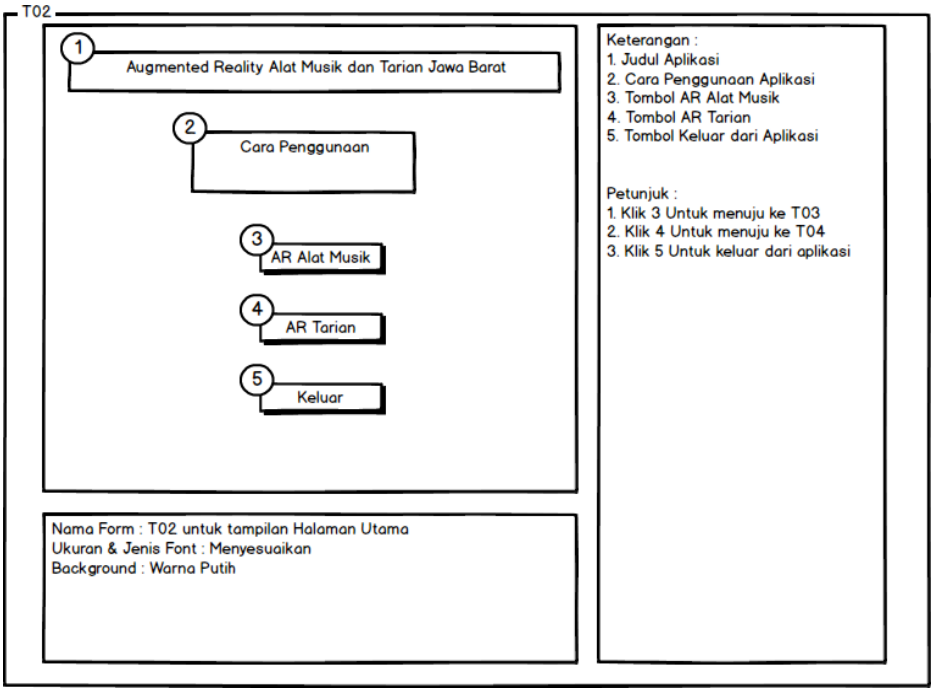
   Perancangan Antarmuka Splash Screen pada aplikasi ini bisa dilihat pada Gambar 3.24



Gambar 3.24 Perancangan Antarmuka Splash Screen

2. Perancangan Antarmuka Halaman Utama

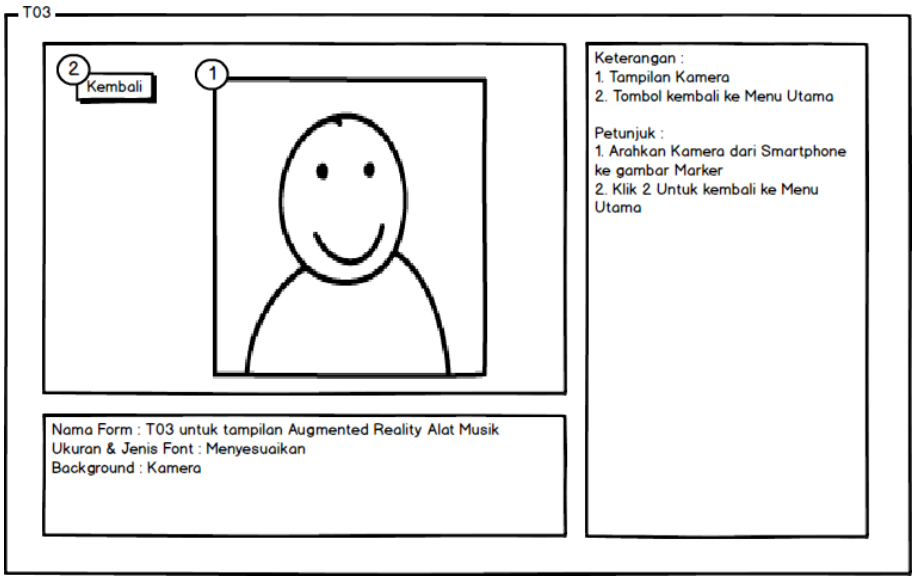
Perancangan Antarmuka Halaman Utama pada aplikasi ini bisa dilihat pada Gambar 3.25



Gambar 3.25 Perancangan Antarmuka Halaman Utama

3. Perancangan Antarmuka Augmented Reality Alat Musik

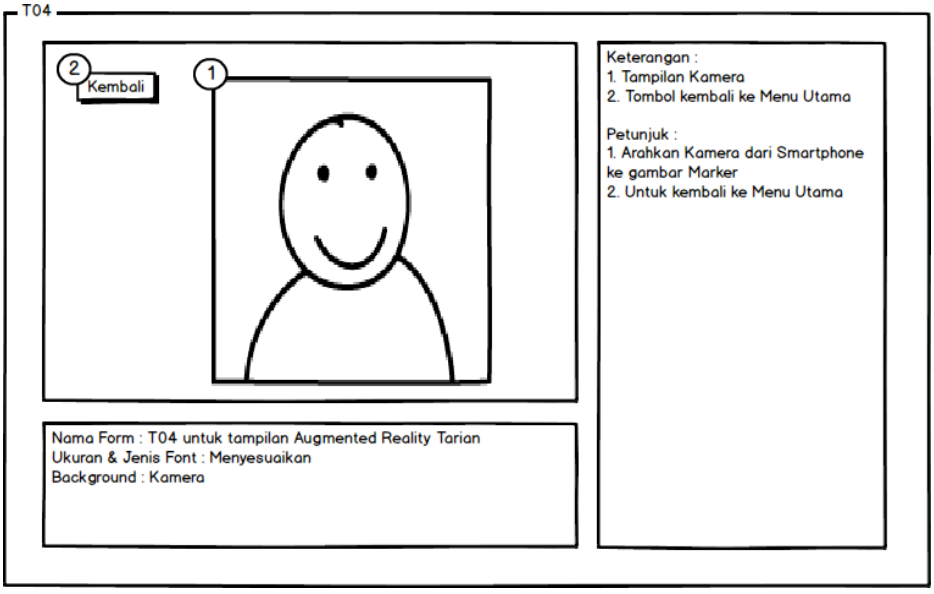
Perancangan Antarmuka Augmented Realiy Alat Musik pada aplikasi ini bisa dilihat pada Gambar 3.26



Gambar 3.26 Perancangan Antarmuka Augmented Reality Alat Musik

4. Perancangan Antarmuka Augmented Reality Tarian

Perancangan Antarmuka Augmented Reality Tarian pada aplikasi ini bisa dilihat pada Gambar 3.27

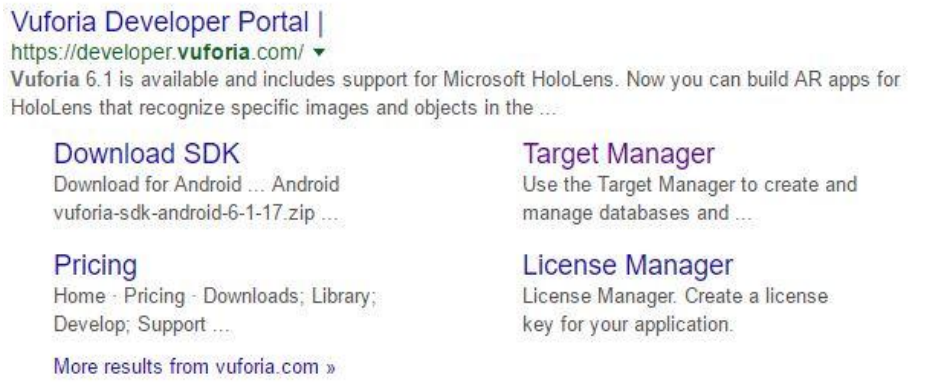


Gambar 3.27 Perancangan Antarmuka Augmented Reality Tarian

3.2.3 Perancangan *Database* Marker

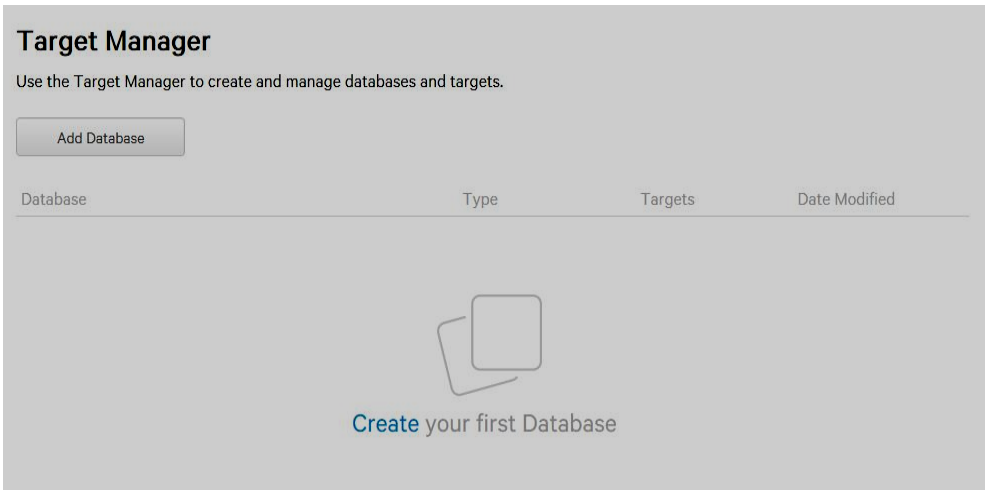
Untuk menampilkan objek, Aplikasi ini membutuhkan sebuah perantara berupa marker. Untuk pembuatan *database* marker dapat dilakukan secara online di website Vuforia. Adapun langkah-langkahnya antara lain :

1. Masuk ke Website Vuforia www.developer.vuforia.com atau ketikkan saja Vuforia di search engine dan kemudian pilih *Target Manager* (Dibutuhkan *Login* terlebih dahulu)



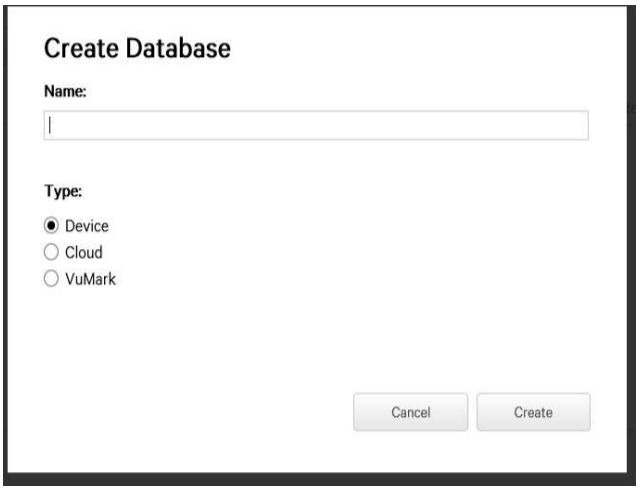
Gambar 3.28 *Target Manager*

2. Pilih Add Database



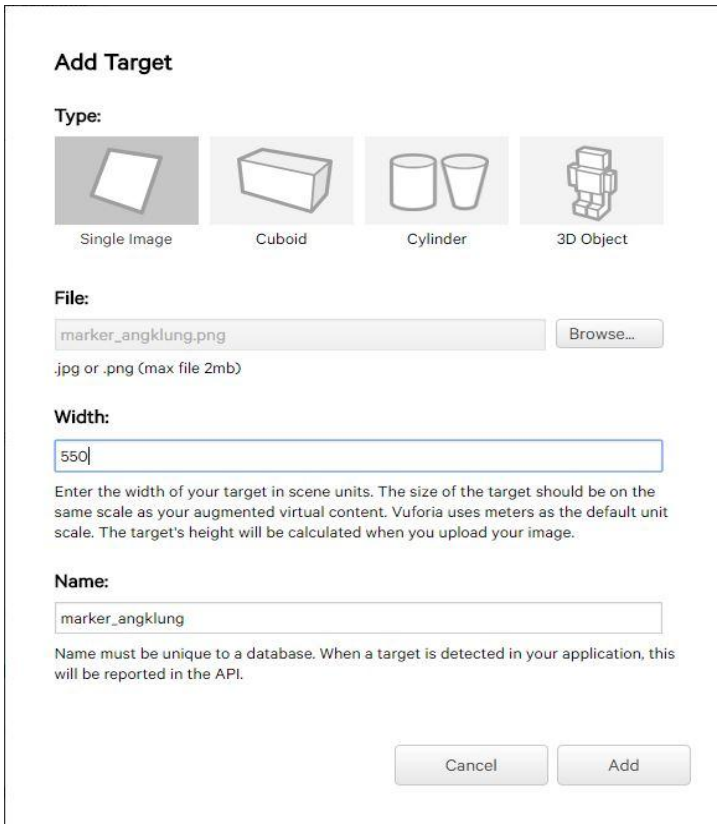
Gambar 3.29 *Add Database*

3. Masukkan nama *Database* kemudian pilih *Create*



Gambar 3.30 *Create Database*

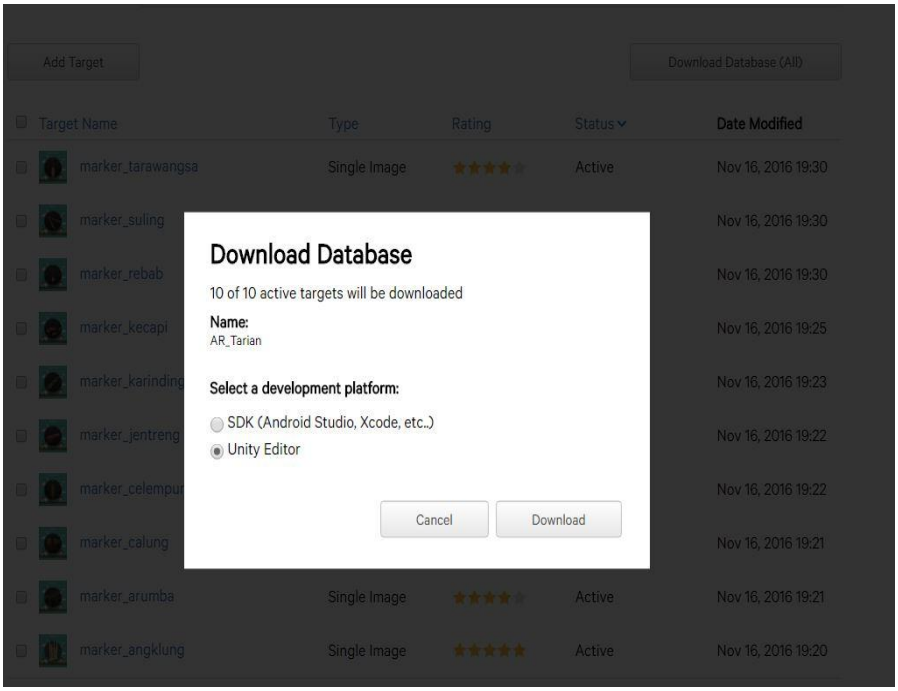
4. Pada *Database*, klik add target kemudian akan ada pop menu sebagai berikut.



Gambar 3.31 *Add Target*

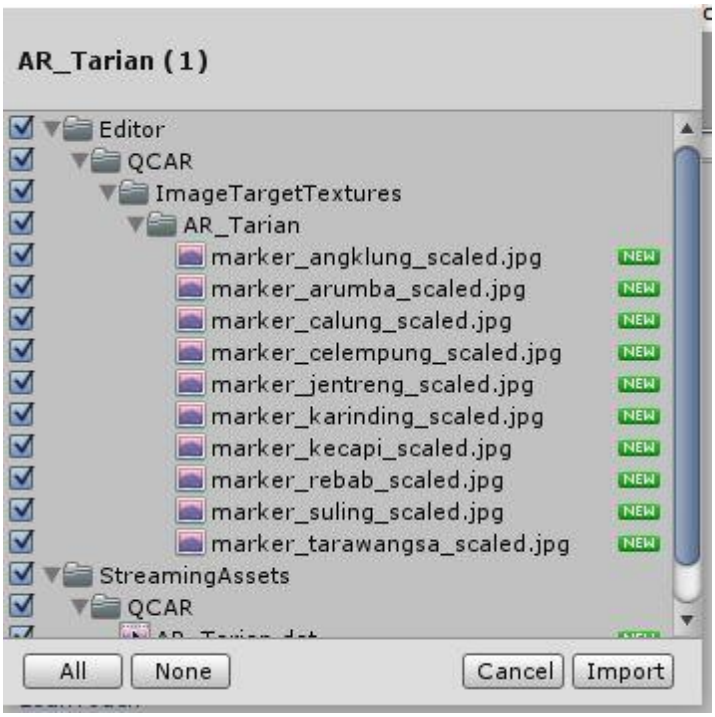
Pilih Single Image, kemudian masukkan file gambar yang akan dijadikan marker. Atur ukurannya, kemudian beri nama sesuai dengan objek yang akan ditampilkan. Jika sudah klik add. Ulangi sampai jumlah marker sesuai dengan objek yang akan ditampilkan

5. Pilih Download database (all), kemudian untuk platform pilih Unity Editor dan klik Download



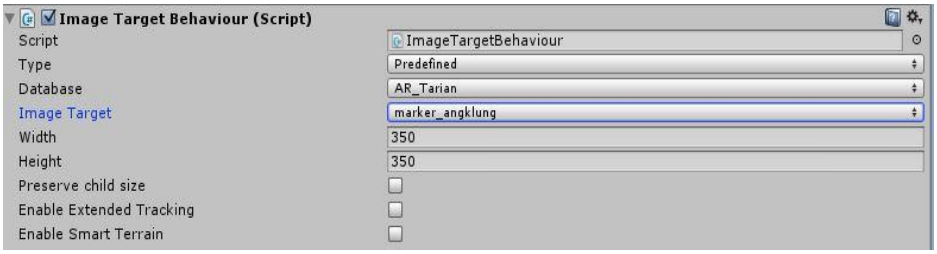
Gambar 3.32 *Download Database*

6. Klik 2x pada file database yang telah didownload kemudian akan muncul tampilan seperti berikut. Klik import



Gambar 3.33 *Import Database*

7. Setelah itu cukup cocokkan marker yang telah dibuat dengan objek yang akan ditampilkan. Contoh : marker_angklung untuk menampilkan objek angklung



Gambar 3.34 Atur *Marker*

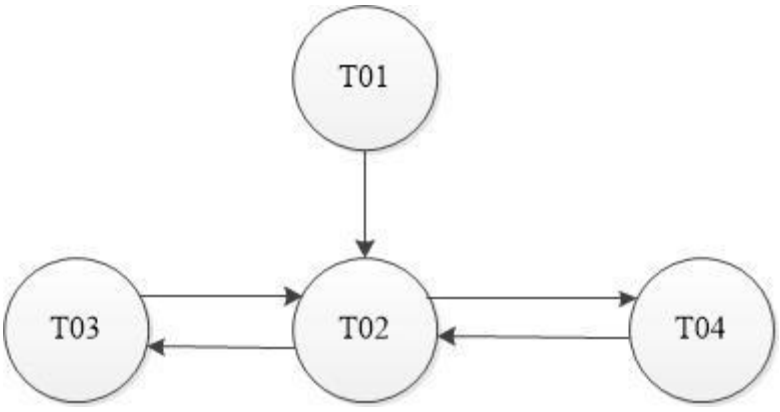
Contoh *Marker* pada aplikasi ini dapat dilihat pada Gambar 3.34



Gambar 3.34 Contoh Gambar *Marker*

3.2.4 Jaringan Semantik

Jaringan semantik merupakan jaringan data yang menunjukan hubungan antar berbagai objek. Jaringan semantik pada ini dapat dilihat pada Gambar 3.35



Gambar 3.35 Jaringan Semantik

Berikut ini adalah keterangan untuk gambar Jaringan Semantik dapat dilihat pada tabel 3.18

Tabel 3.18 Keterangan Gambar Jaringan Semantik

No	Kode	Keterangan
1	T01	Antarmuka Splash Screen
2	T02	Antarmuka Halaman Utama
3	T03	Antarmuka Augmented Realiy Alat Musik
4	T04	Antarmuka Augmented Reality Tarian