BAB 3

ANALISIS DAN PERANCANGAN

3.1 Analisis

Analisis bertujuan untuk mengidentifikasi permasalahan – permasalahan yang ada pada sistem serta menentukan kebutuhan dari sistem yang dibangun. Analisis tersebut meliputi analisis masalah, analisis kebutuhan data, analisis sistem. Analisis merupakan tahapan yang paling penting, karena kesalahan dalam tahap ini akan menyebabkan kesalahan di tahap selanjutnya. Aplikasi yang dibuat merupakan program untuk medeteksi sebuah marker sekaligus menampilkan objek tiga dimensi. Aplikasi yang dibuat merupakan edukasi berbasis *augmented reality* yaitu media pembelajaran pengenalan rumah adat tradisional untuk anak sekolah dasar. seolah-olah pengguna berinteraksi langsung dengan objek virtual dalam dunia nyata yang disajikan dalam bentuk buku.

Aplikasi ini dibuat dengan mengambil latar dilingkungan nyata yang kemudian digabungkan dengan objek 3D melalui kamera. Orientasi dan posisi *marker* akan dideteksi lewat *frame-frame* yang ditangkap oleh kamera. Setelah *marker* terdeteksi oleh kamera, maka akan didapatkan matriks transformasi yang dapat digunakan untuk transformasi seluruh objek yang ada dalam aplikasi.

3.1.1 Analisis Masalah

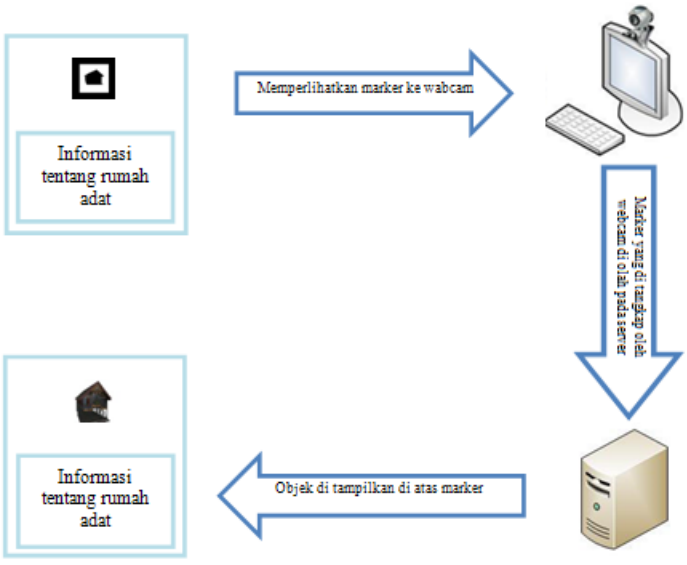
Saat ini pembelajaran mengenai pengenalan rumah adat tradisional hanya melalui buku bacaan dan gambar yang ditempel di dinding kelas. Dimana siswa hanya membaca serta melihat buku atau gambar dan guru hanya berbicara di depan kelas. Sehingga kurangnya interaksi antara siswa dan guru, membuat siswa kurang memperhatikan dan memahami materi yang disampaikan oleh guru.

Agar solusi tersebut dapat dicapai maka diterapkan teknologi *augmented reality* sehingga dapat berinteraksi melalui dunia objek digital secara lebih nyata. Karena keunggulan yang terdapat pada teknologi *augmented reality* tersebut maka peranan teknologi dalam melestarikan dan mengenalkan rumah adat dapat ditingkatkan.

3.1.2 Analisis Gambaran Umum Sistem

Pada arsitektur aplikasi yang akan dibangun terdiri dari beberapa komponen, yaitu ; *user* yang menggunakan aplikasi pengenalan rumah adat berbasis *augmented reality*, *user* mengarahkan marker/magic book sehingga marker dapat tertangkap olah kamera. Kemudian dari gambar yang didapat dari kamera sistem komputer melakukan tracking marker untuk mengidentifikasi marker yang digunakan oleh *user*. Komputer melakukan render objek-objek 3D yang digunakan dalam aplikasi.

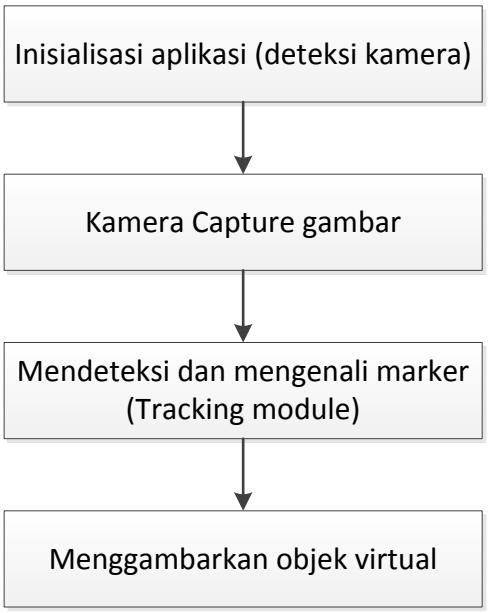
User dapat melihat hasil manipulasi *system* melalui layar komputer/ monitor. Gambaran arsitektur sistem dapat dilihat pada gambar 3.1.



Gambar 3.1 Gambaran umum sistem[8]

3.1.3 Analisis Metode

Marker Based Tracking merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih. Komputer akan mengenali posisi dan orientasi marker dan menciptakan dunia virtual 3D yaitu titik (0,0,0) dan 3 sumbu yaitu X,Y,dan Z. *Marker Based Tracking* ini sudah lama dikembangkan sejak 1980-an dan pada awal 1990-an mulai dikembangkan untuk penggunaan *augmented Reality*. Penggunaan Marker dapat membantu meningkatkan akurasi penempatan objek virtual di dunia nyata sesuai dengan pergerakan kamera. Penggunaan marker juga cukup mudah untuk digunakan merender berbagai objek yang berbeda. Dengan menggunakan kamera dan marker objek virtual dapat muncul diatas gambar marker seperti berada di dunia nyata.



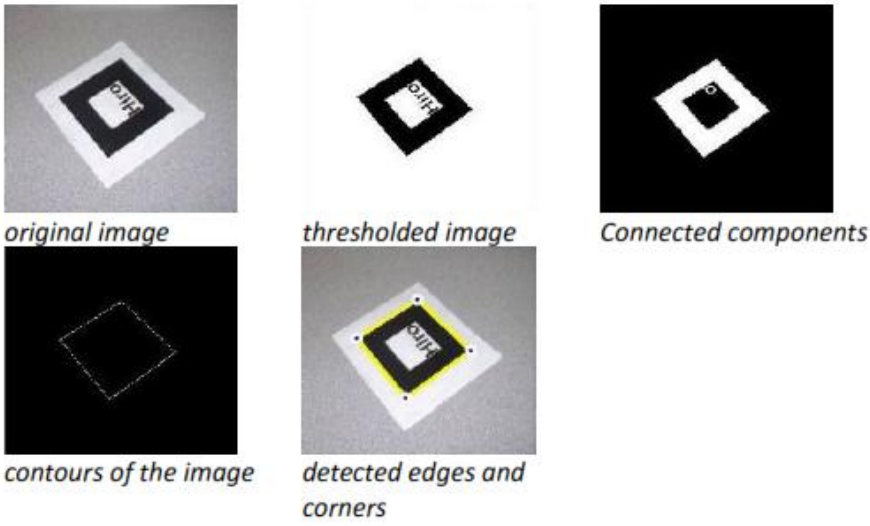
Gambar 3.2 Alur *system augmented reality*

Pada Gambar 3.2 Alur *system augmented reality*, dimulai dengan inisialisasi aplikasi (deteksi kamera) pada tahap ini aplikasi akan mendeteksi ketersediaan kamera pada perangkat keras user, kemudian kamera

mengambil gambar pada tahap ini kamera akan menangkap gambar dari dunia nyata, kemudian akan dilakukan proses mendeteksi dan mengenali marker (*Tracking module*), setelah marker terdeteksi dan dikenali sistem akan menggambarkan objek virtualnya.

3.1.3.1 Mendeteksi dan Mengenali Marker

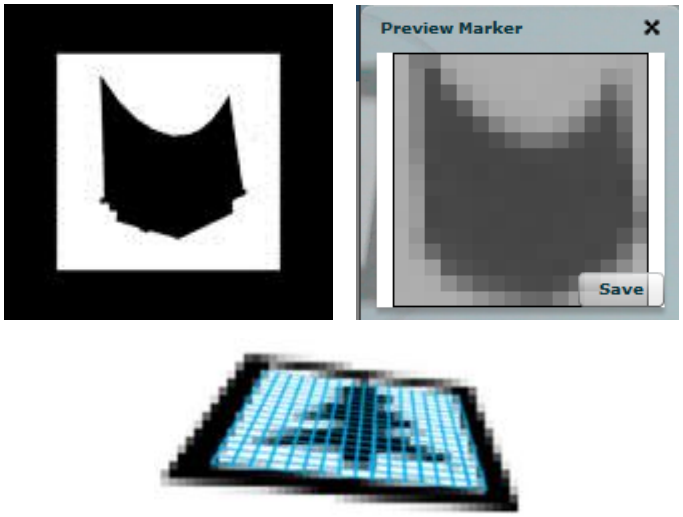
Pada proses mendeteksi dan mengenali marker (*Tracking module*) sistem memerlukan intensitas gambar (*greyscale image*) sehingga sistem akan mengkonversi gambar menjadi greyscale kemudian mencari beberapa bentuk persegi setelah bentuk persegi (Gambar 3.3) ditemukan sistem akan mendeteksi wilayah didalam persegi (Pattern Recognition) dan kemudian dibandingkan kecocokan marker dengan pattern di dalam database.



Gambar 3.3 Contoh proses tracking module

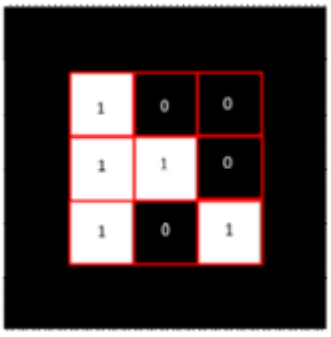
3.1.3.2 Pattern Recognition

Pattern Recognition adalah mendeteksi wilayah di dalam persegi setelah marker untuk dibandingkan kecocokannya dengan pattern di dalam database sebagai penanda objek virtual. Setelah wilayah di dalam marker ditemukan sistem akan merubah dalam ukuran 16x16 (Gambar 3.4) dan diberi nilai biner pada setiap sel atau pixel nya.



Gambar 3.4 Contoh marker (sebelah kiri), marker dalam ukuran 16x16 (tengah), marker terdeteksi dengan sampel grid 16x16 pixel

Marker dikalkulasikan dalam bentuk biner pada setiap sel berdasarkan warna, warna hitam = 0 dan warna putih = 1. Nilai pada setiap sel merupakan nilai pada setiap pixel pada marker. Untuk setiap sel , sistem mendapat nilai biner dan seluruh data marker dapat direpresentasikan sebagai serangkaian nilai-nilai biner atau sebagai salah satu bilangan biner . Dalam sederhana data matriks biner ini (biner) merupakan sebagai ID penanda (Gambar 3.5)



Gambar 3.5 Contoh decoding marker: marker ID 100110101 (=309)

3.1.3.3 Multi Marker

Multi Marker merupakan Teknik *Marker Based Tracking* yang menggunakan dua marker atau lebih untuk memanipulasi satu objek. Ini merupakan salah satu cara interaksi untuk memanipulasi objek virtual yang seakan berada di dunia nyata. Pada implementasinya multi marker terdapat dua tipe yaitu static dan dinamis. Statik marker digunakan untuk objek traking kamera dan dinamik marker lainnya digunakan untuk memanipulasi objek. Seperti pada aplikasi ini akan terdiri dari dua jenis marker, yaitu marker objek dan marker aksi untuk membuat memanipulasi objek pada marker objek.

3.1.4 Perancangan *Marker*

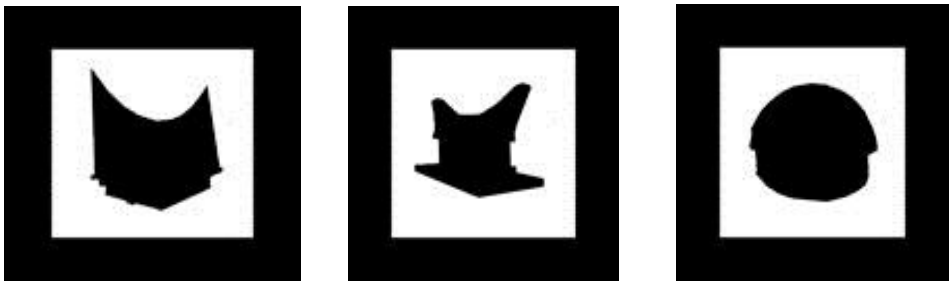
Marker merupakan bagian yang sangat penting. Perancangan marker tidak boleh dilakukan sembarangan, ada aturan yang harus dipenuhi dalam merancang sebuah marker.

1. Dalam kasus ini marker harus berwarna hitam agar lebih mempermudah dalam proses perhitungan pendeteksian marker dan merender objek.
2. Marker yang digunakan harus berbentuk segi empat.
3. Ukuran marker akan berpengaruh terhadap objek yang akan ditampilkan. Semakin besar ukuran marker maka akan semakin jauh jarak yang dibutuhkan untuk merender objek.
4. Ketebalan marker juga sangat diperhatikan dalam membuat sebuah marker. Tebal marker disarankan minimal 25% dari panjang garis tepi marker.

Marker-marker yang digunakan pada *magic book* AR Rumah adat tradisional ini adalah:



(a) *Marker* aksi (b) Rumah Gadang-Sumbar (c) Rumah Aceh-Aceh



(d) Rumah Batak-Sumut (e) Rumah Tongkonan-Sulsel (f) Rumah Honai-Papua



(g) Rumah Limas-Jambi

Gambar 3.6 Marker AR Rumah adat

Keterangan:

Gambar a: Marker aksi rumah adat

Gambar b: Marker untuk objek Rumah adat Gadang - Sumatera Barat

Gambar c: Marker untuk objek Rumah adat Aceh - Aceh

Gambar d: Marker untuk objek Rumah adat Balai Batak Toba - Sumut

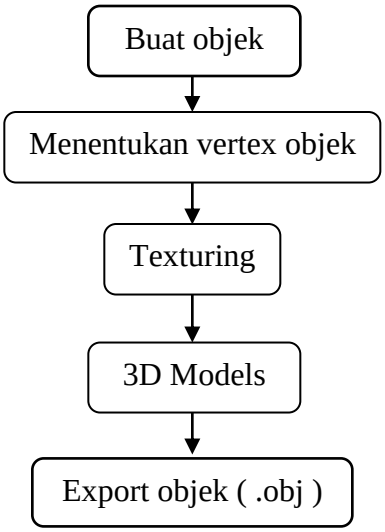
Gambar e: Marker untuk objek Rumah adat Tongkonan – Provinsi Sulsel

Gambar f: Marker untuk objek Rumah adat Honai – Provinsi Papua

Gambar g: Marker untuk objek Rumah adat Limas – Provinsi Jambi

3.1.5 Proses Pembuatan Objek

Objek yang dibuat menggunakan *software* Sketchup. Objek-objek yang dibuat dalam AR Rumah adat ini masih menggunakan *standard primitive*. Dalam proses pemodelan terdiri dari dua langkah. Pertama, kita membuat objek 3D dengan menggunakan Sketchup . Kedua mengeksport objek yang sudah kita buat ke dalam NyARtoolkit. Untuk lebih jelasnya dapat dilihat pada gambar 3.7 dibawah ini.



Gambar 3.7 Tahapan pemodelan

3.1.5.1 Kalibrasi Kamera

Kalibrasi kamera adalah salah satu langkah yang harus dilakukan dalam proses rekonstruksi 3D, dimana proses ini diperlukan untuk mendapatkan informasi parameter kamera yang digunakan untuk melakukan transformasi dari 3D (*world coordinate*) menuju ke 2D (*camera coordinate*). Parameter kamera terdiri dari dua yaitu parameter intrinsik dan parameter ekstrinsik yang nantinya akan digunakan untuk melakukan perhitungan,

sehingga dapat ditentukan letak dari suatu benda dalam ruang 3D. Parameter intrinsik terdiri dari empat unsur yaitu:

a) Nilai focus kamera, yaitu jarak antara lensa kamera dengan bidang gambar.
b) Titik pusat proyeksi, yaitu lokasi titik tengah gambar dalam pixel koordinat.
c) Ukuran *pixel* efektif.
d) Koefisien distorsi, yaitu koefisien tingkat kelengkungan lensa meliputi radial dan tangensial distorsi.

3.1.5.2 Cara Kerja NyarToolkit

NyarToolkit menggunakan teknik visi komputer untuk mengkalkulasikan sudut pandang kamera nyata ke Marker yang nyata. Ada lima langkah, dalam proses kerja NyarToolkit.

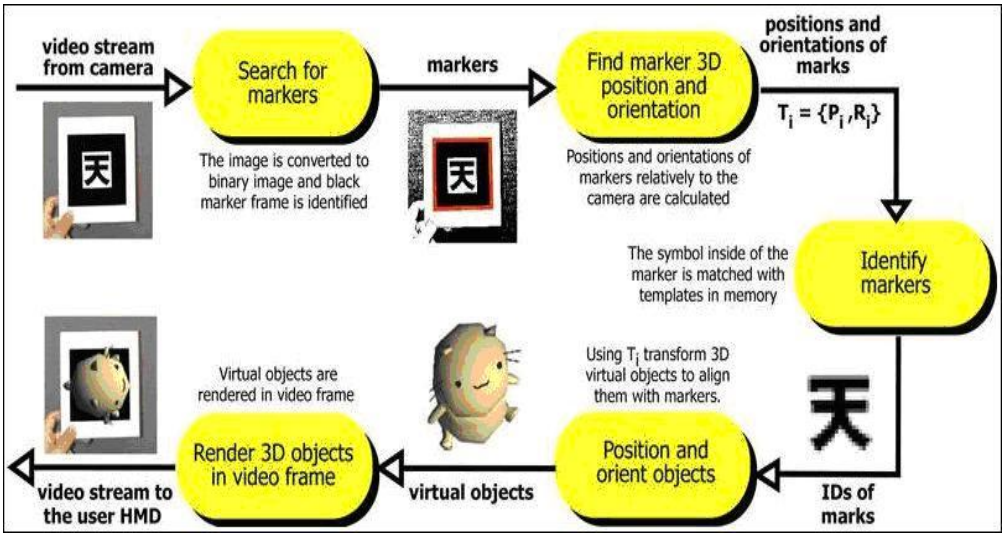
1. Mencari Marker , kemudian Marker yang dideteksi dirubah menjadi binary, kemudian black frame atau bingkai hitam terdeteksi oleh kamera.

2. Kamera menemukan posisi Marker 3D dan dikalkulasikan dengan kamera nyata.

3. Kamera mengindentifikasi Marker, apakah pola Marker sesuai dengan templates memory.

4. Dengan mentransformasikan posisi Marker .

5. Objek 3D di render diatas Marker .



Gambar 3.8 Cara kerja NyarToolkit

Sebelum kamera digunakan, kamera harus dikalibrasi terlebih dahulu. Kalibrasi kamera merupakan bagian yang sangat penting dalam proses pengambilan *input* video. Hal ini disebabkan oleh distorsi pada lensa kamera yang tiap kamera berbeda karakteristiknya. Tujuan dari kalibrasi kamera adalah untuk menghitung tingkat distorsi dari sebuah lensa kamera yang digunakan agar *image* yang dihasilkan mendekati image ideal. Parameter ini nantinya digunakan dalam perhitungan pada proses *Pose and Position Estimation* agar model objek dapat ditampilkan tepat diatas *marker* .

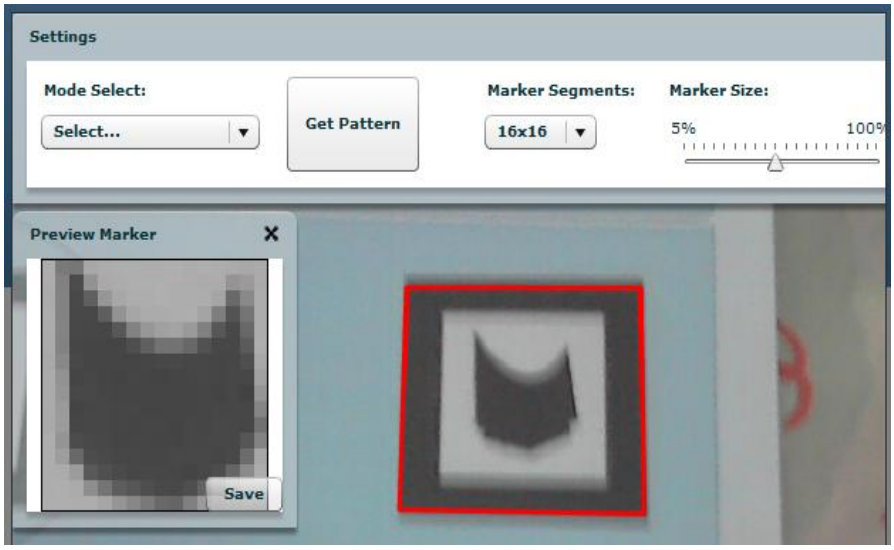


Image ideal Distorsi image

Gambar 3.9 Image ideal dan distorsi image

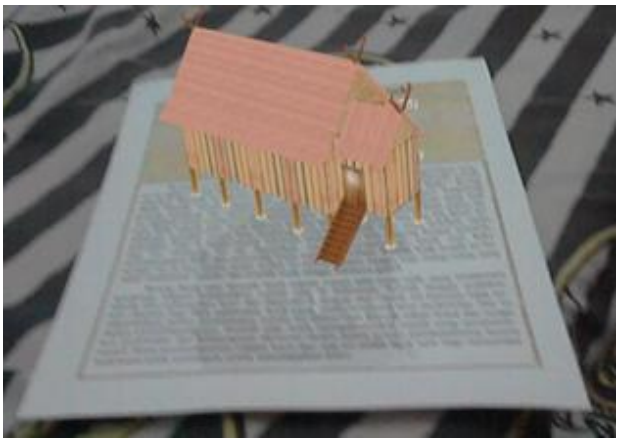
a) Deteksi *Marker*

Deteksi *Marker* merupakan tahap dimana *Marker* akan di identifikasi oleh *webcam* sebagai sasaran untuk penempatan objek yang akan dirender. Deteksi *Marker* ini dengan menggunakan AR toolkit generator



Gambar 3.10 Deteksi *Marker*

Setelah marker di deteksi atau di identifikasi oleh wabcam, marker yang sudah di identifikasi ini akan mengeluarkan output atau akan merender objek yang dikenali oleh sistem, bisa di lihat pada gambar 3.11:



Gambar 3.11 Hasil render objek

3.1.6 Analisis Kebutuhan Non Fungsional

Analisis *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. Analisis

kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan. Pada analisis kebutuhan sistem non-fungsional ini dijelaskan analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras dan analisis pengguna.

3.1.6.1 Analisis Kebutuhan Perangkat Keras

Aplikasi media pembelajaran pengenalan rumah adat adalah program yang membutuhkan grafis video dengan kemampuan relative tinggi. Hal ini karena model-model yang ditampilkan menggunakan teknik *rendering.*

Berdasarkan studi literatur terhadap proyek-proyek pengembangan teknologi AR, maka diperolah spesifikasi minimum perangkat keras yang dibutuhkan oleh pihak pengembang untuk teknologi AR seperti berikut ini:

1. *Processor* Intel core 2 duo atau setara.
2. RAM minimal 2 GB
3. Ruang sisa *hardisk* minimal 4 GB
4. *Motherboard* dengan *chipset* yang kompatibel dengan VGA *card* yang dipakai.
5. Optimum menggunakan VGA *card* dengan kemampuan me-*render* grafis 3D, seperti GeForce 6xxx atau ATI 1xxx series.
6. *Webcam*

Webcam dipakai untuk menangkap citra yang kemudian diproses oleh *tracker library.* Hasil dari proses ini akan menghasilkan matriks transformasi *marker* relative terhadap *webcam.*

3.1.6.2 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak digunakan dalam sebuah sistem merupakan kumpulan perintah-perintah yang diberikan kepada perangkat keras agar saling berinteraksi untuk melakukan suatu tugas. Perangkat lunak yang

dibutuhkan oleh pihak pengembang untuk membangun aplikasi adalah sebagai berikut :

1. Sistem operasi Windows 7
2. Processing 1.2.1
3. Photoshop CS5
4. Autodesk 3D max
5. JAVA
6. Google Chrome
7. Delphi 7

Sedangkan Perangkat lunak yang dibutuhkan oleh pihak pengguna untuk aplikasi ini adalah sebagai berikut :

1) Sistem operasi Windows 7
2) JAVA
3) Delphi 7

3.1.6.3 Analisis Pengguna

Analisis pengguna dimaksudkan untuk mengetahui siapa saja pengguna yang terlibat pada aplikasi. Pengguna yaitu *public* yang dapat mengerti dan memahami komputer sehingga dapat menggunakan aplikasi yang akan dibangun. Aplikasi ini digunakan oleh guru SD kelas VI.

3.1.7 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan.

3.1.7.1 Pemodelan Sistem

Pemodelan sistem dilakukan dengan metode berorientasi obyek dengan menggunakan standarisasi UML, dengan tahap – tahap penentuan siapa saja *actor* yang terlibat dan perancangan *use case specification, Interaksi diagram (sequence diagram).*

3.1.7.2 *Use Case Diagram*

Use case diagram ini digunakan untuk menggambarkan hubungan sejumlah external *actor* dengan *use case* yang terdapat dalam *system*. *Use case diagram* ini hanya menggambarkan keadaan lingkungan *system* yang dapat dilihat dari luar oleh *actor*. Pada *use case* ini terdapat tiga *actor*, yaitu *user*, *marker* dan kemera. Penjelasan mengenai tiga *actor* dapat dilihat pada tabel 3.1

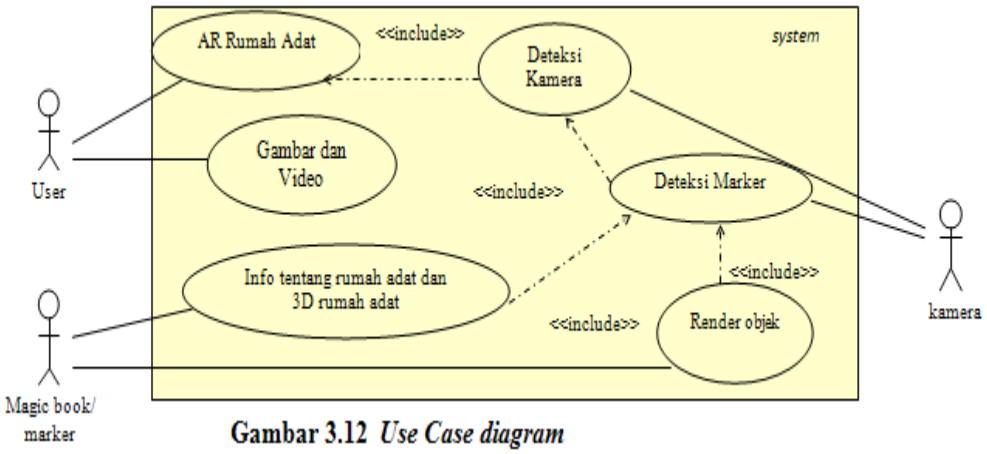
Tabel 3.1 Aktor dalam *Use Case*

Nama Aktor	**Definisi**
User	Orang yang berinteraksi dengan *system*, yaitu pengguna yang mengakses semua fungsi yang disediakan *system*.
Marker	Marker adalah alat bantu yang digunakan sebagai penanda dimana objek virtual 3D akan ditampilkan.
Kamera	Kamera adalah alat bantu yang digunakan sebagai pendeteksi keberadaan marker.

3.1.7.3 *Skenario Use Case Diagram*

Use case pada aplikasi berhubungan dengan aplikasi dan penggunaan *augmented reality* pada *magic book.* Dimana *augmented reality* digunakan dalam memanipulasi objek-objek maya dengan berinteraksi dengan objek fisik di lingkungan nyata. Agar dapat melihat detail *uses case* dapat lihat pada tabel 3.2

Diagram *use case* pada gambar 3.12 dilakukan oleh actor dengan User dimana diagram *use case* ini lebih bagaimana actor dapat menjalankan fungsi-fungsi yang terdapat pada aplikasi.



Gambar 3.12 ***Use Case diagram***

Bagian ini menjelaskan masing-masing deskripsi dari Use *Case* diagram yang dilakukan oleh *user*. Mulai dari proses memiih masuk kedalam aplikasi, cara menggunakan aplikasi, dan tentang *magic book.*

Tabel 3.2 Deskripsi *Use Case Diagram*

NO	Kode *Use Case*	Nama *Use Case*	Keterangan
1	UC-P01	AR Rumah Adat	Proses untuk memulai aplikasi AR Rumah Adat
2	UC-P02	Deteksi *Marker*	Sistem mendeteksi keberadaan marker melalui gambar yang ditangkap oleh kamera

3	UC-P03	Render Objek	Sistem akan melakukan render obyek pada awal aplikasi dijalankan .sistem akan merender gambar,tekstur dan
4	UC-P04	Deteksi Kamera	Sistem akan mendeteksi keberadaan kamera
5	UC-P05	Info tentang rumah adat dan 3D rumah adat	Sistem akan mendeteksi marker aksi yang digerakan dan merender gambar,tekstur dan model geometri sesuai dengan aksi yang di deteksi
6	UC-P06	Gambar dan Video	Sistem akan menampilkan gambar dan video

Tabel 3.3 Spesifikasi Skenario : AR Rumah Adat

Nama *Use Case*	AR Rumah Adat	
Nomor	UC-P01	
Aktor	User	
Kondisi Awal	User berada di tampilan awal aplikasi	
Reaksi Aktor		**Reaksi Sistem**
1. User membuka aplikasi *augmented reality* rumah adat		2. Sistem menjalankan aplikasi *Augmented Reality* Rumah Adat 3. Sistem mendeteksi ketersediaan kamera 5. Kamera terdeteksi
Kondisi Akhir	Aplikasi menampilkan jendela AR Rumah Adat dan mendeteksi kamera	

Skenario Alternatif	1. User memilih menu yang lain. 2. Kamera tidak terdeteksi

Tabel 3.4 Spesifikasi Skenario: Deteksi Marker

Nama *Use*	Deteksi Marker	
Nomor	UC-P02	
Aktor	User, Kamera, *Magic Book*	
Kondisi	Kamera telah terdeteksi	
Reaksi Aktor	**Reaksi Sistem**	**Kamera**
1. Magic book/marker dihadapkan pada kamera	3. Sistem menkonversi gambar yang didapatkan dari asli menjadi grayscale 4. Sistem menkonversi gambar yang didapatkan dari grayscale menjadi binary image. 5. Sistem mendeteksi marker dari gambar biner yang didapatkan. 6. Sistem mengambil gambar pattern dari marker (pattern recognition). 7. Sistem melakukan penskalaan pattern resolusi 16x16. 8. Sistem mendecoding pattern Sistem mencocokan decoding pattern dengan *template* file (.patt) di aplikasi *(template matching).*	2. Kamera mendapatkan gambar dan mengirim ke system.
Kondisi Akhir	Marker terdeteksi	
Skenario Alternatif	Marker tidak terdeteksi 1. Marker tidak terdeteksi, tidak terjadi apa-apa.	

Tabel 3.5 Spesifikasi Skenario: Render Objek

Nama *Use Case*	Render Objek	
Nomor	UC-P03	
Aktor	Marker	
Kondisi Awal	*Marker* terdeteksi	
Reaksi Aktor		**Reaksi Sistem**

1. Menunjukan *marker* 6. Menampilkan objek	2. Deteksi marker. 3. Menentukan posisi marker 4. Update info marker 5. Merender objek
Kondisi Akhir	Semua konten berhasil direnderuntuk digunakan dalam Aplikasi dan menampilkan objek diatas marker.
Skenario Alternatif	1. Marker tidak terdeteksi.

Tabel 3.6 Spesifikasi Skenario: Deteksi Kamera

Nama *Use Case*	Deteksi Kamera
Nomor	UC-P04
Aktor	Kamera
Kondisi Awal	Kamera belum terdeteksi
Reaksi Aktor	**Reaksi Sistem**
2. Mengambil gambar	1. Kamera terdeteksi
Kondisi Akhir	Menerima gambar dari kamera dan Menamplkan jendela AR dan Kamera terdeteksi
Skenario Alternatif	1. Bila kamera tidak terdeteksi, layar akan berwarna putih dan *user* mencoba mengulang membuka aplikasi. 3. Menampilkan pesan kamera tidak terdeteksi memilih menu yang lainnya.

Tabel 3.7 Spesifikasi Skenario : Info tentang rumah adat dan 3D rumah adat

Nama *Use Case*	Info tentang rumah adat dan 3D rumah adat
Nomor	UC-P05
Aktor	User
Kondisi Awal	User berada di dalam aplikasi
Reaksi Aktor	**Reaksi Sistem**

1. Membuka aplikasi 2. Memilih info tentang rumah adat.	3. Menampilkan halaman info tentang rumah adat.
Kondisi Akhir	Aplikasi menampilkan jendela Info tentang rumah adat
Skenario Alternatif	1. Memilih menu lainnya.

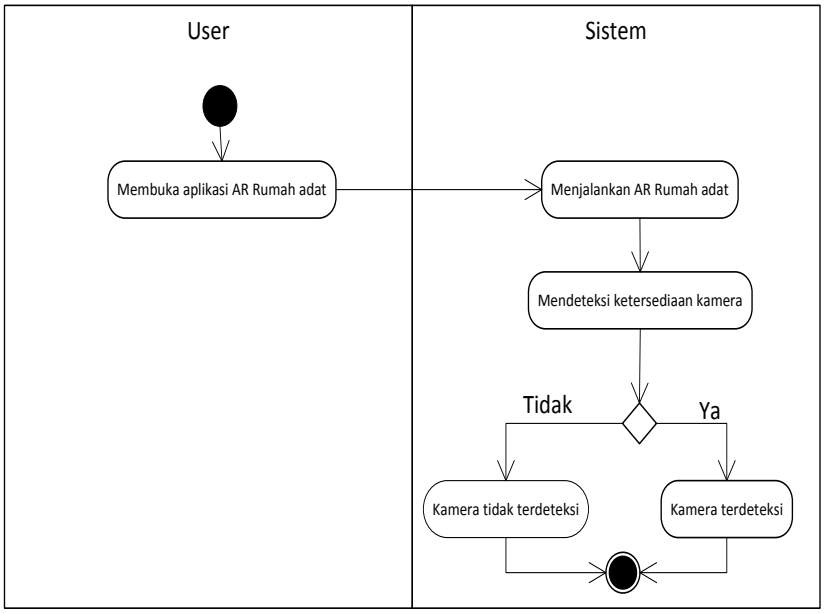
Tabel 3.8 Spesifikasi Skenario : Gambar dan Video

Nama *Use Case*	Gambar dan Video	
Nomor	UC-P06	
Aktor	*User*	
Kondisi Awal	*User* berada di tampilan Gambar dan Video	
Aksi Aktor		**Aksi Sistem**
1. Membuka aplikasi 2. Memilih Gambar dan Video.		3. Sistem menjalankan Gambar dan Video
Kondisi Akhir	Aplikasi menampilkan jendela Gambar dan Video	
Scenario Alternatif	1. Memilih menu yang lainnya.	

3.1.7.4 *Activity Diagram*

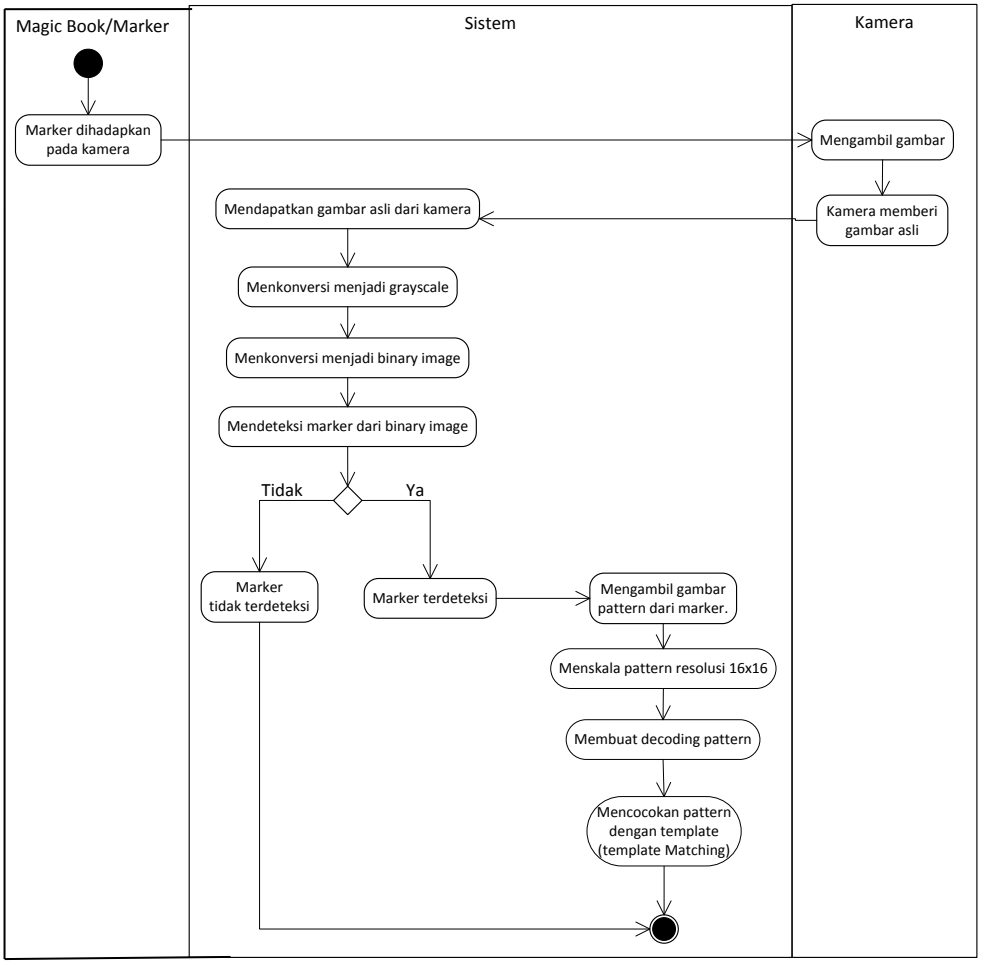
Diagram activity menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi dan bagaimana mereka berakhir. Penggambaran *activity* diagram memiliki kemiripan dengan *flowchart diagram. Activity diagram* memodelkan *event-event* yang terjadi pada *Use Case* dan digunakan untuk pemodelan aspek dinamis dari sistem.

1. *Activity diagram* terkait dengan skenario AR Rumah adat dapat dilihat pada gambar 3.13



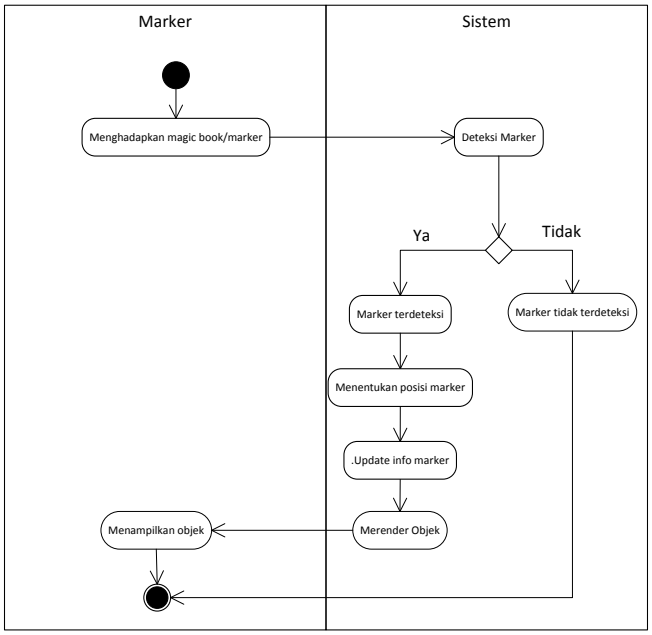
Gambar 3.13 *Activity diagram* AR rumah adat

2. *Activity* diagram terkait dengan skenario deteksi *marker* dapat dilihat pada gambar 3.14.



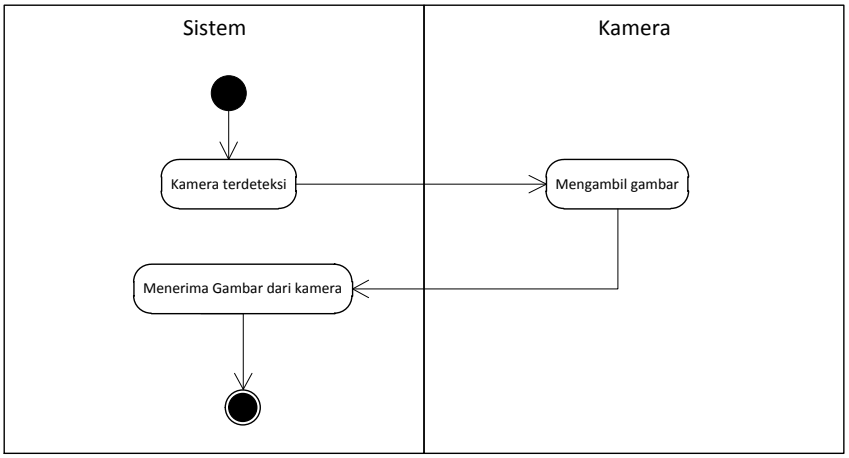
Gambar 3.14 Activity Diagram Deteksi Marker

3. *Activity* diagram terkait dengan skenario render objek dapat dilihat pada gambar 3.15.



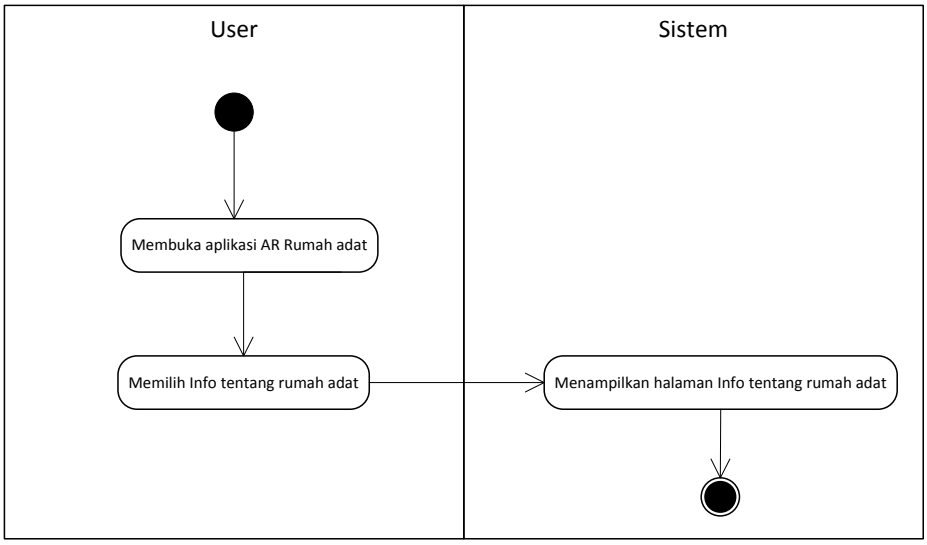
Gambar 3.15 Activity Diagram Render Objek

4. *Activity* diagram terkait dengan skenario deteksi kamera dapat dilihat pada gambar 3.16.



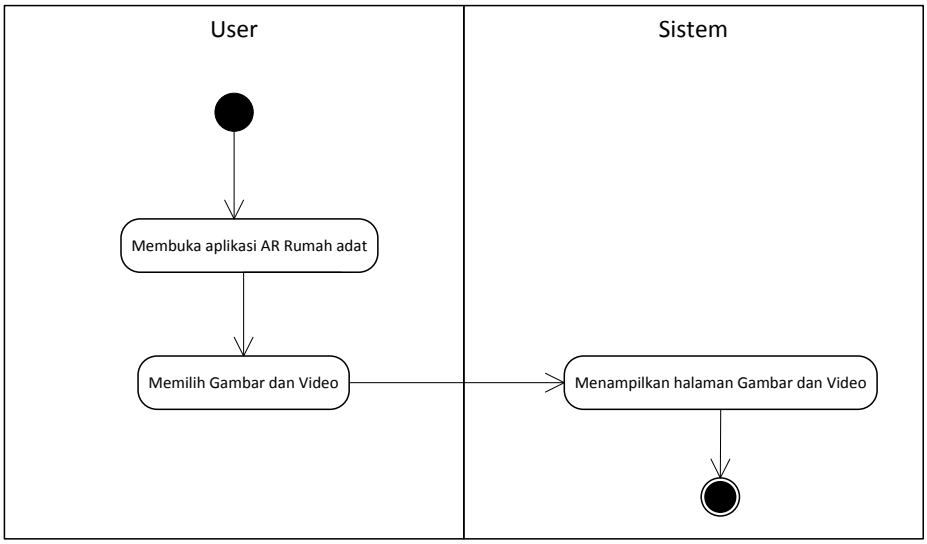
Gambar 3.16 Activity Diagram Deteksi Kamera

5. *Activity* diagram terkait dengan skenario info tentang rumah adat dan 3D rumah adat dapat dilihat pada gambar 3.17.



Gambar 3.17 Activity Diagram Info tentang rumah adat dan 3D rumah adat

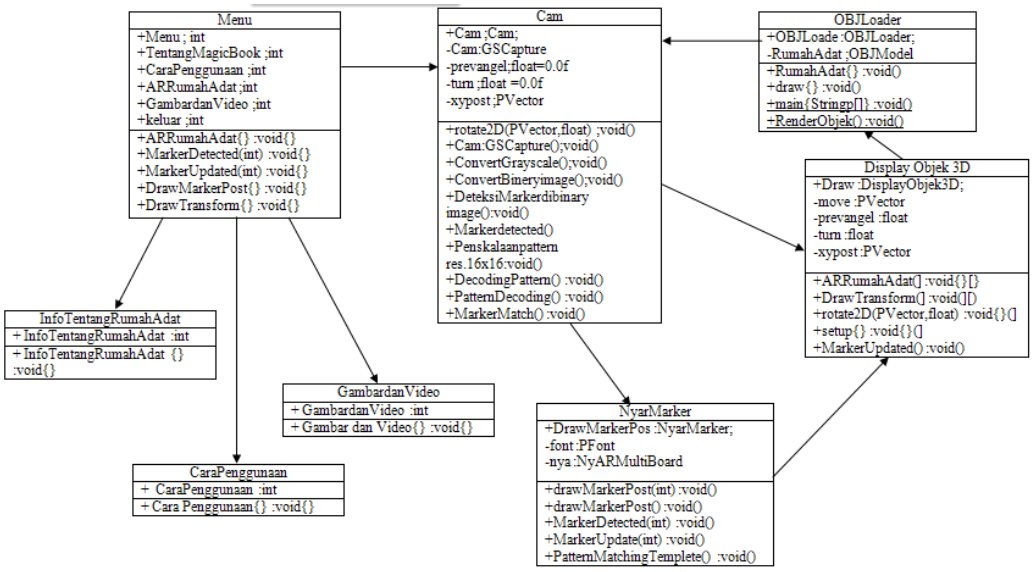
6. Activity diagram terkait dengan skenario gambar dan video dapat dilihat pada gambar 3.18.



Gambar 3.18 Activity Diagram Gambar dan Video

3.1.7.5 *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). Gambar 3.19 adalah kelas diagram dari implementasi teknologi *Augmented Reality* pada aplikasi pengenalan rumah adat tradisional.



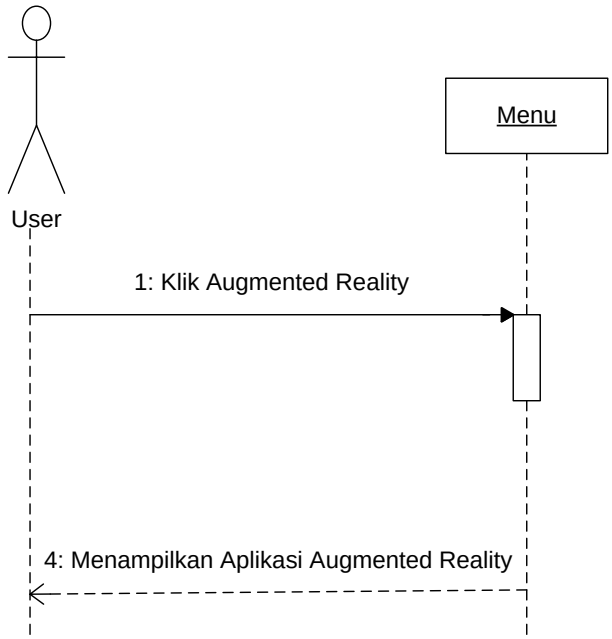
Gambar 3.19 *Class Diagram aplikasi*

3.1.7.6 *Sequence Diagram*

Sequence diagram menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh objek dan *message* (pesan) yang diletakkan diantara objek- objek ini di dalam *use case*. Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segiempat. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan progres vertikal.

1. *Sequence Diagram* AR Rumah adat

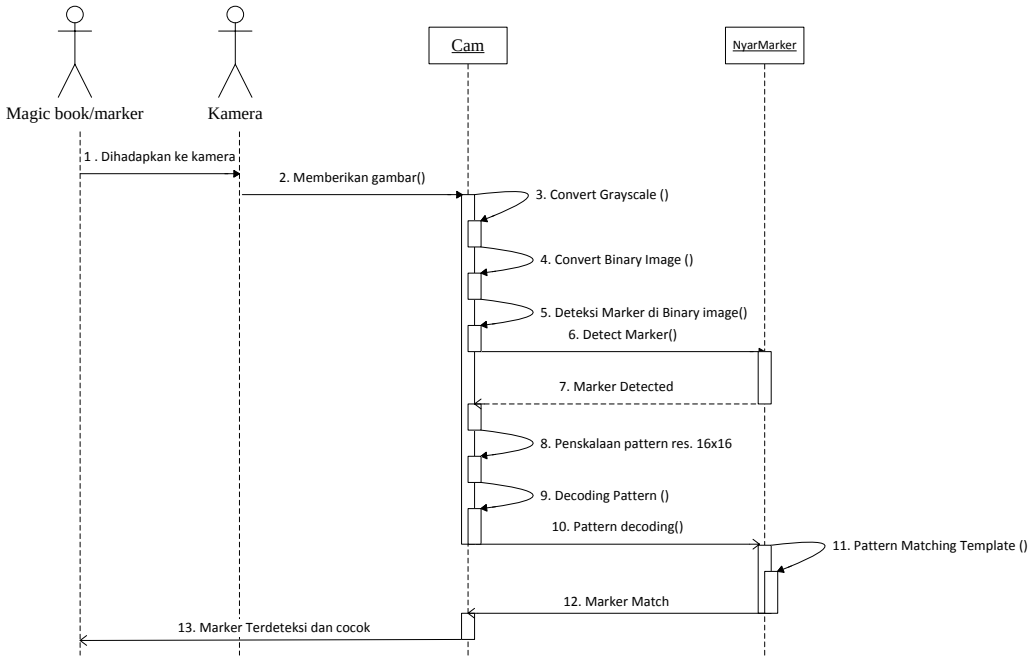
   Gambar *Sequence Diagram* AR Rumah adat dapat dilihat pada gambar 3.20



Gambar 3.20 *Sequence Diagram* AR Rumah adat

2. *Sequence Diagram* Deteksi Marker

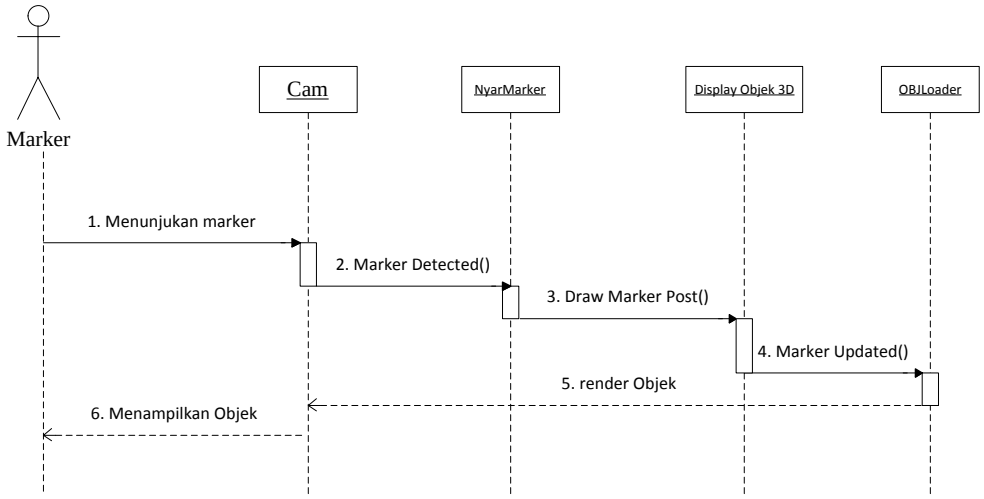
   Gambar *Sequence Diagram* Deteksi Marker aplikasi dapat dilihat pada gambar 3.21



Gambar 3.21 *Sequence Diagram* Deteksi Marker

3. *Sequence Diagram* Render Objek

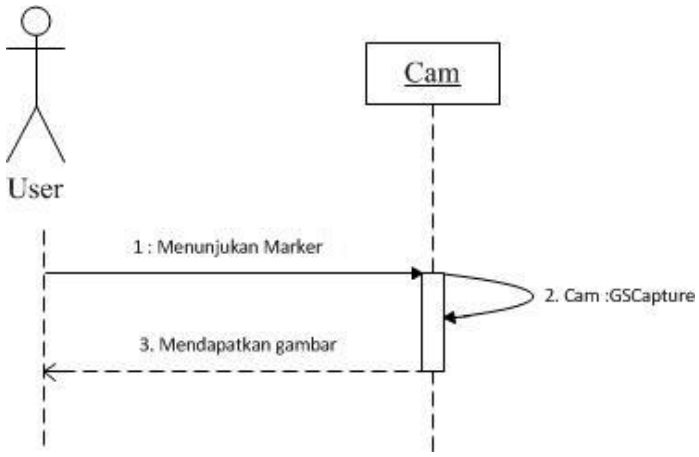
   Gambar *Sequence Diagram* render objek dapat dilihat pada gambar 3.22



Gambar 3.22 *Sequence Diagram* Render Objek

4. *Sequence Diagram* Deteksi kamera

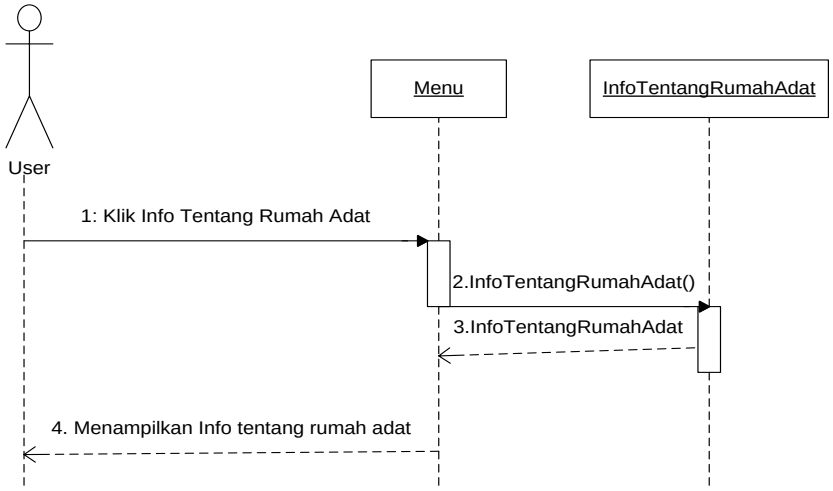
   Gambar *Sequence Diagram* deteksi kamera dapat dilihat pada gambar 3.23



Gambar 3.23 *Sequence Diagram* Deteksi kamera

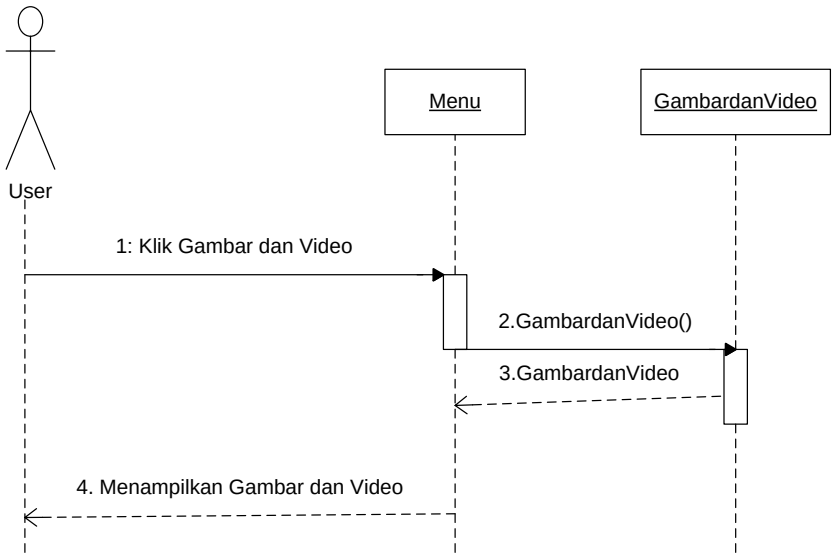
5. *Sequence Diagram* Info tentang rumah adat dan 3D rumah adat

   Gambar *Sequence Diagram* Info tentang rumah adat dan 3D rumah adat dapat dilihat pada gambar 3.24



Gambar 3.24 *Sequence Diagram* Info Tentang Rumah Adat

6. *Sequence Diagram* Gambar dan Video
   Gambar *sequence diagram* Gambar dan Video dapat dilihat pada gambar 3.25



Gambar 3.25 *Sequence Diagram* Gambar dan Video

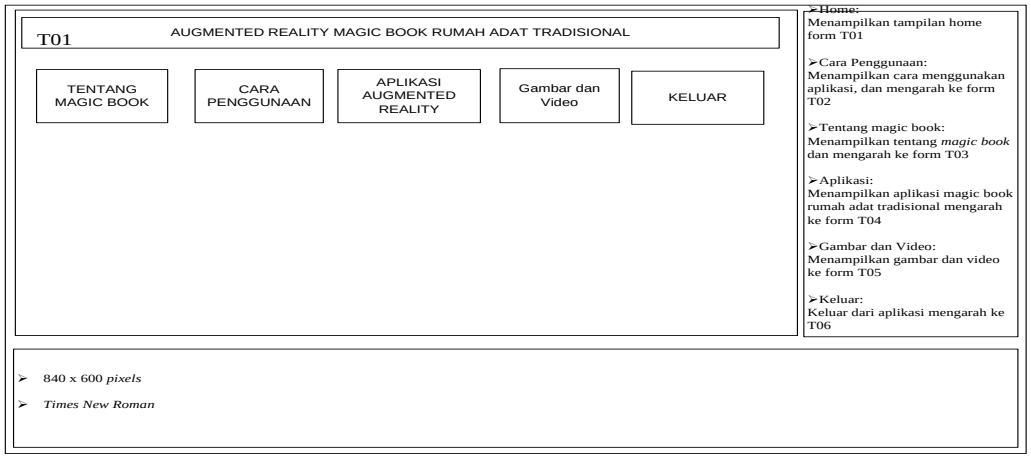
3.2 Perancangan Sistem

Perancangan adalah suatu bagian dari metodologi pengembangan pembangunan suatu perangkat lunak yang dilakukan setelah tahapan untuk memberikan gambaran secara terperinci. Berdasarkan uraian diatas perancangan sistem merupakan tahapan dari siklus pengembangan sistem yang didefinisikan dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perancangan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satukesatuan yang utuh dan berfungsi, juga menyangkut konfigurasi dari komponen-komponen perangkat keras dan perangkat lunak dari suatu system.

3.2.1 Perancangan Antarmuka

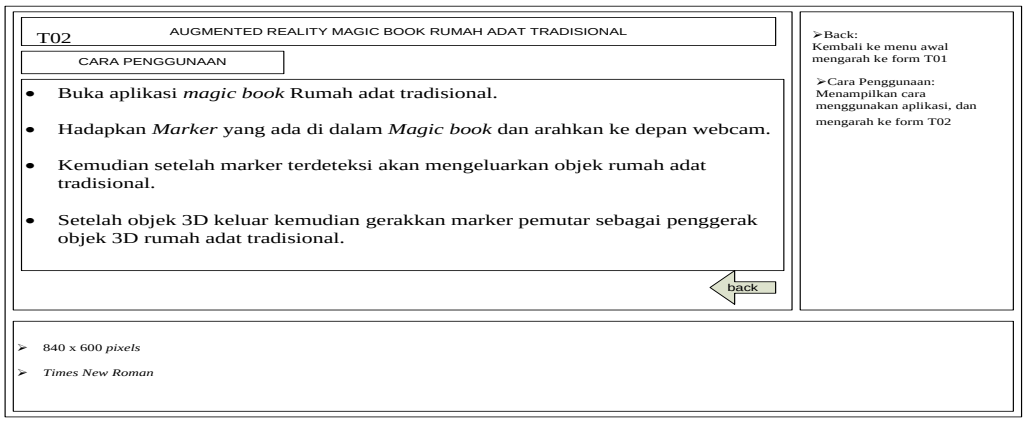
Interface atau antar muka merupakan tampilan dari suatu program aplikasi yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara program dengan pengguna. Aplikasi yang akan dibangun diharapkan menyediakan interface yang mudah dipahami dan digunakan oleh pengguna.

Desain perancangan antarmuka aplikasi dapat dilihat pada gambar 3.26 sampai gambar 3.30



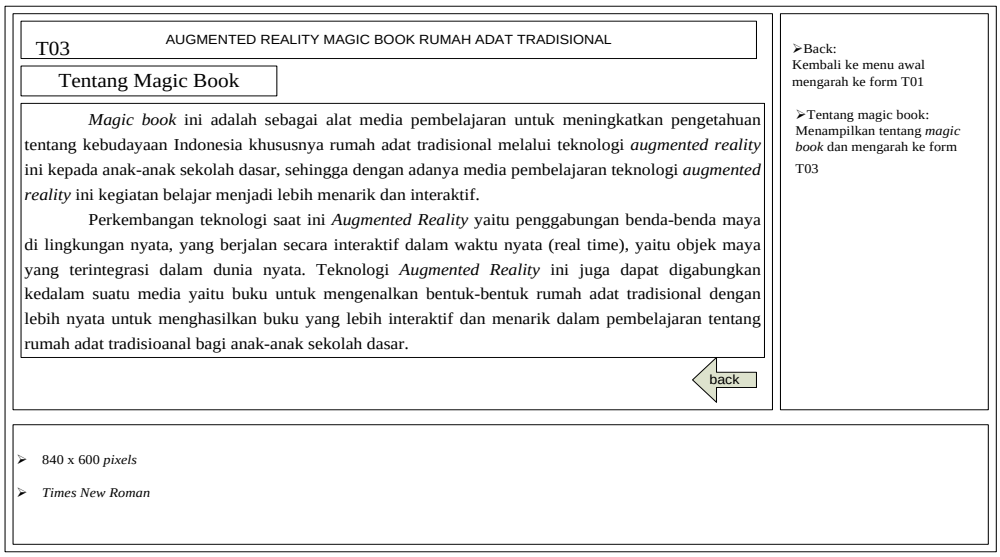
Gambar 3.26 Perancangan Antarmuka Home

Perancangan antarmuka cara menggunakan aplikasi dapat dilihat pada gambar 3.27



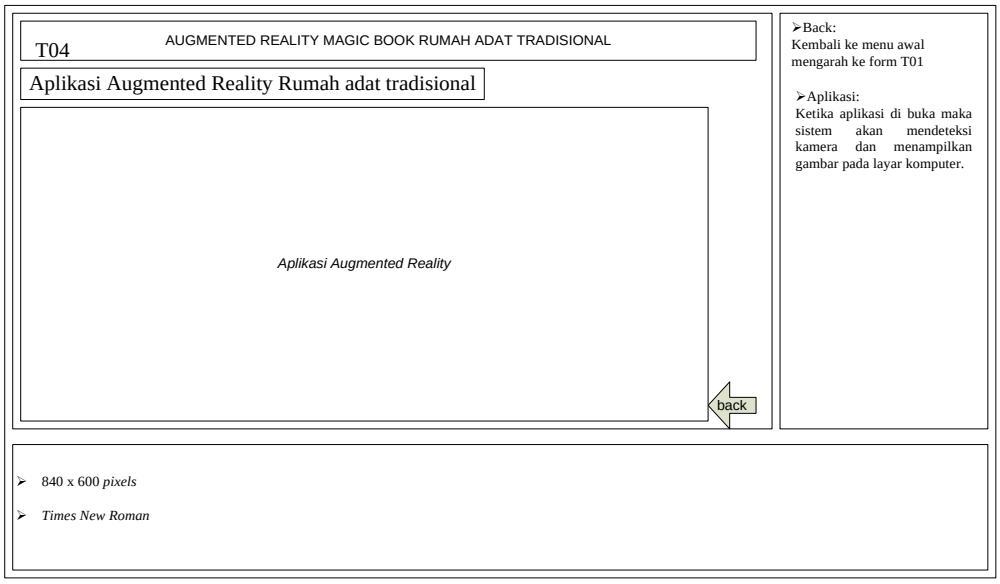
Gambar 3.27 Perancangan Antarmuka Cara Menggunakan Aplikasi

Perancangan antarmuka tentang *magic book* dapat dilihat pada gambar 3.28



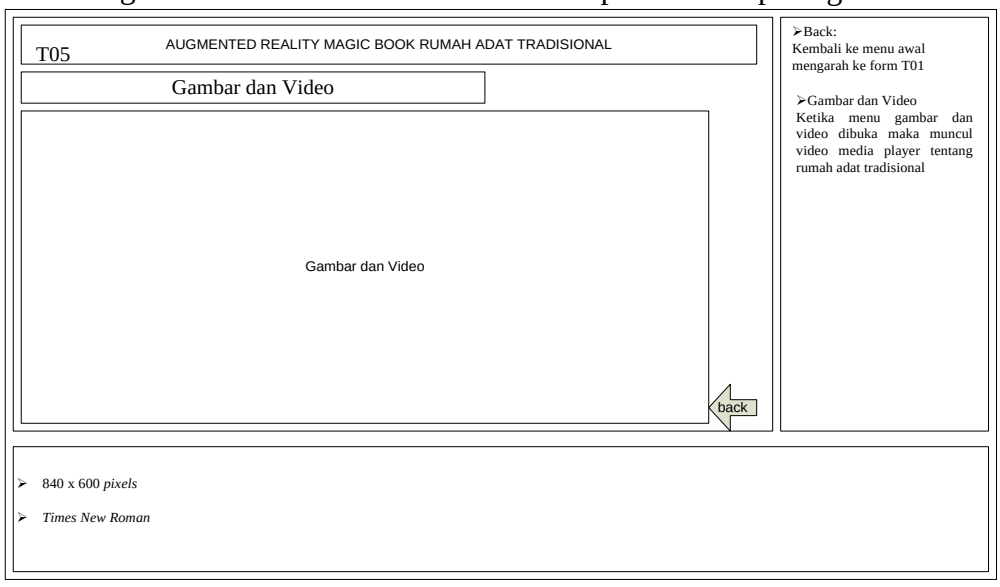
Gambar 3.28 Perancangan Antarmuka Tentang *magic book*

Perancangan antarmuka aplikasi *Augmented Reality* rumah adat tradisional dapat dilihat pada gambar 3.29



Gambar 3.29 Perancangan Antarmuka Aplikasi Augmented Reality Rumah adat

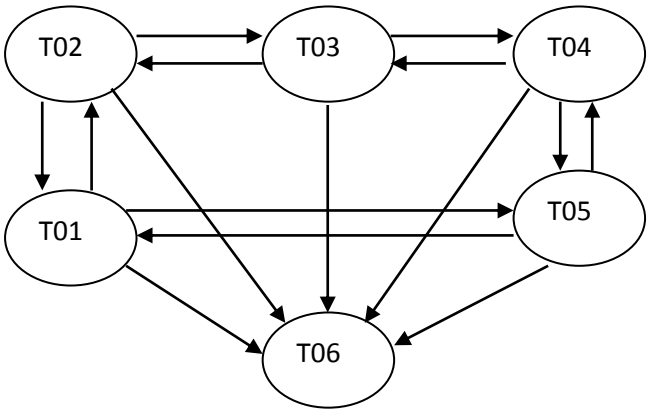
Perancangan antarmuka Gambar dan Video dapat dilihat pada gambar 3.30



Gambar 3.30 Perancangan Antarmuka Gambar dan Video

3.2.2 Jaringan Semantik

Jaringan semantik merupakan jaringan data dan informasi, yang menunjukan hubungan antar berbagai objek. dapat dilihat pada gambar 3.31



Gambar 3.31 Jaringan Semantik

Keterangan :

T01= Home

T02= Cara Menggunakan

T03= Tentang *Magic Book*

T04= Aplikasi Augmented Reality

T05= Gambar dan Video

T06=Keluar

3.2.3 Perancangan Aplikasi

Daerah area aplikasi berada dalam *marker* yang telah dibuat. Dalam area aplikasi ini terdapat objek-objek rumah adat dalam model tiga dimensi. Aplikasi ini memiliki tujuan yaitu, agar anak dapat balajar dan mengetahui berbagi macam jenis rumah adat berdasarkan asal daerahnya. Untuk mencapai tujuan itu user bertindak sebagai pihak yang mengarahkan *marker* ke webcam.

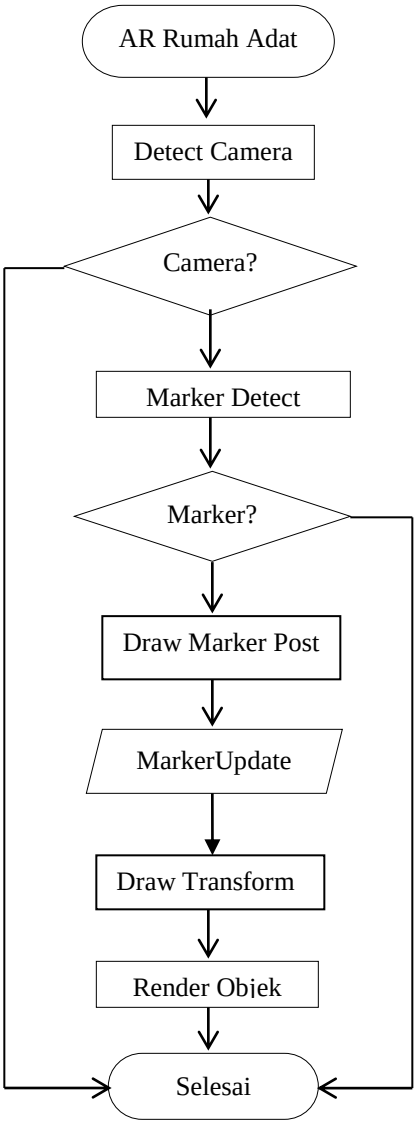
Area aplikasi ini berbentuk sebuah kotak, dimana setiap marker tersebut terdapat suatu objek rumah adat. Semakin besar marker maka akan semakin besar pula objek yang akan di tampilkan begitu pun dengan semakin jauh jarak marker dengan webcam maka semakin kecil pula objek yang akan di tampilkan.

3.2.4 Perancangan Metode

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 *Magic Book* rumah adat tradisional *augmented reality*. Adapun method-method yang terdapat dalam aplikasi yang akan dibangun adalah sebagai berikut :

1. Perancangan metode AR Rumah adat tradisional

   Method cara menggunakan seperti terlihat pada gambar 3.32



Gambar 3.32 Gambar method AR Rumah adat