BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Sistem yang dibuat merupakan berbasis *augmented reality* yaitu pengenalan tokoh wayang golek. Aplikasi yang dibuat seolah-olah pengguna dapat berinteraksi dengan wayang golek menggunakan magic card dengan obyek-obyek 3 dimensi yang dibuat.

Aplikasi ini dibuat dengan mengambil latar dilingkungan nyata yang kemudian gabungkan dengan obyek-obyek 3D melalui kamera. Orientasi dan posisi *marker* akan dideteksi lewat *frame-frame* yang ditangkap oleh kamera.

3.1.1 Analisis Masalah

Berkurangnya minat dan ketertarikan masyarakat terhadap kebudayaan daerah terutama wayang golek ini dikarenakan kemajuan teknologi[8] oleh karena itu untuk dapat mengingatkan dan mendapatkan perhatian, pelestariannya harus dapat mengikuti perkembangan jaman dengan masuk dan dikembangkan kedalam teknologi. *Augmented reality* merupakan perkembangan teknologi yang dapat menjadi media untuk pelestarian kebudayaan daerah wayang golek ini. Bentuk virtual dari wayang golek dapat digabungkan dengan dunia nyata. Marker sebagai alat bantu dalam *augmented reality* sebagai penanda objek virtual dapat diterapkan kedalam bentuk kartu atau disebut magic card[9].

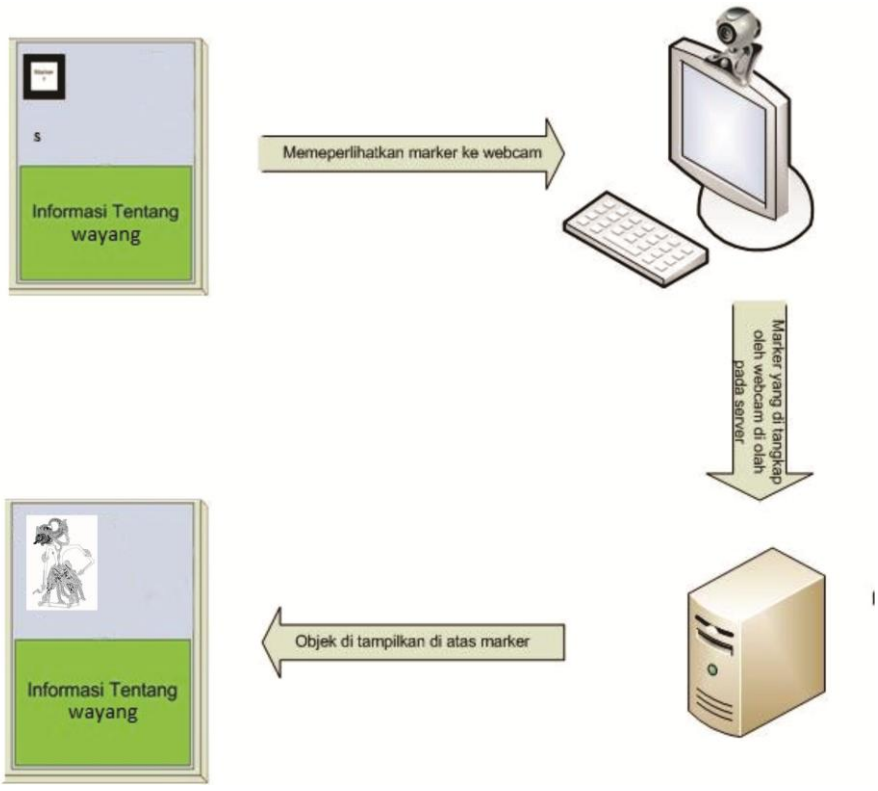
Analisis masalah adalah suatu gambaran masalah yang diangkat dalam penulisan skripsi tentang mengimplentasikan teknologi *augmented reality* pada aplikasi yang dapat meningkatkan dan mendekatkan wayang golek dengan teknologi maka dibangun aplikasi yang memberikan pengenalan secara lebih interaktif. Agar 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 wayang golek dapat ditingkatkan.

3.1.2 Analisis Arsitektur Sistem

Pada arsitektur aplikasi yang akan dibangun terdiri dari beberapa komponen, yaitu *user* yang menggunakan aplikasi pengenalan tokoh wayang golek berbasis augmented reality, *user* mengarahkan marker/magic card 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 obyek-obyek 3D yang digunakan dalam aplikasi.

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



Gambar 3.1 Arsitektur Aplikasi [4]

3.1.3 Analisis Cerita

Pada Aplikasi ini pengenalan tokoh dan karakter wayang golek akan dikemas didalam sebuah cerita untuk membantu pengguna agar lebih mengerti mengenai tokoh wayang golek tersebut. Ceritanya berjudul "Kematian gatotkaca".

"Kematian Gatotkaca"

Ketika Gatotkaca lahir tali pusarnya tidak dapat diputus dengan jenis senjata apapun kecuali senjata Kunta Wijayandanu. Sehingga Batara Narada membawakan senjata Kunta Wijayandanu yang akan diberikan kepada Arjuna untuk memotong tali pusar Gatotkaca. Tetapi diperjalanan Batara Narada dikecoh dan diperdayai oleh Karna dan ayahnya (Batara Surya) sehingga Kunta Wijayandanu berhasil direbut oleh Karna. Setelah mengetahui hal itu, Arjuna segera mengejar Karna. Sehingga terjadilah perebutan Kunta Wijayandanu. Setelah pertarungan yang sangat sengit akhirnya Karna melarikan diri dengan membawa busur panah Kunta Wijayandanu, sedangkan Arjuna hanya mendapatkan kerangka/sarung senjatanya.

Sehingga sarung Kunta Wijayandanu inilah yang digunakan untuk memotong tali pusar gatotkaca. Anehnya setelah digunakan untuk memotong tali pusar, sarung senjata ini masuk kedalam pusar gatotkaca, menyatu dengan tubuhnya. Lalu Dewa menitahkan bahwa nanti yang dapat membunuh Gatotkaca hanya Karna karena Wijayandanu akan mencari sarungnya yang terdapat didalam pusar gatotkaca.

Ketika itu di Kahyangan sedang terjadi huru-hara akibat sebuan Patih Sekipu dan rajanya Kala Pracona. Gatotkaca yang masih bayi dimasukan ke kawah Candradimuka. Setelah muncul dari kawah Candradimuka, Gatotkaca telah berbaju besi dan badannya menjadi sangat kuat. Akhirnya Gatotkaca dapat membunuh kedua raksasa tersebut. Setelah menang Gatotkaca dipertemukan dengan ayah dan ibunya beserta para Pandawa yang disambut dengan suka cita. Setelah dewasa Gatotkaca menjadi raja di Negara ibunya yaitu Pringgadani.

Pada perang Baratayudha Gatotkaca menjadi senapati. Tak satu pun prajurit Kurawa yang dapat menandingi kesaktian Gatotkaca. Kurawa kocar-kacir, banyak prajurit yang tewas. Karena Kurawa akan kalah, kemudian Karna diangkat

menjadi senapati. Gatotkaca tahu dan ingat perintah Batara Narada. Setiap berhadapan dengan Karna, Gatotkaca selalu bersembunyi di balik awan. Karena Karna tidak bisa terbang, Karna melepaskan senjata Kunta Wijayandanu. Tetapi panah itu tidak dapat menjangkau Gatotkaca, namun tiba – tiba arwah sukma Kalabendana muncul sambil menyampaikan berita dari Kahyangan bahwa ajal Gatotkaca telah ditetapkan malam itu. Gatotkaca yang pasrah terhadap takdirnya berpesan supaya mayatnya bisa digunakan untuk membunuh musuh. Kalibendana setuju, Kemudian Kalabendana menangkap panah Kunta Wijayandanu dan meneruskan untuk bisa menjangkau Gatotkaca. Akhirnya senjata Kunta Wijayadanu pun menusuk perut Gatotkaca dan melebur dengan sarungnya, yang tersimpan di dalam pusar Gatotkaca. Setelah Gatotkaca gugur, Kalabendana melemparkan jenazah Gatotkaca kearah Karna. Karna berhasil melompat sehingga lolos dari maut. Namun keretanya hancur berkeping - keping akibat tertimpa tubuh Gatotkaca. Pecahan kereta tersebut melesat ke segala arah dan menewaskan para prajurit Kurawa[5].

3.1.3.1 Analisis Tokoh Wayang Golek

Analisis Tokoh Wayang Golek merupakan penjelasan mengenai siapa dan bagaimana karakter tokoh wayang golek yang akan dikenalkan di dalam aplikasi ini. Analisis dari tokoh yang akan dijadikan objek 3D dalam aplikasi ini dapat dilihat pada tabel 3.1 [5].

Tabel 3.1 Analisis Tokoh

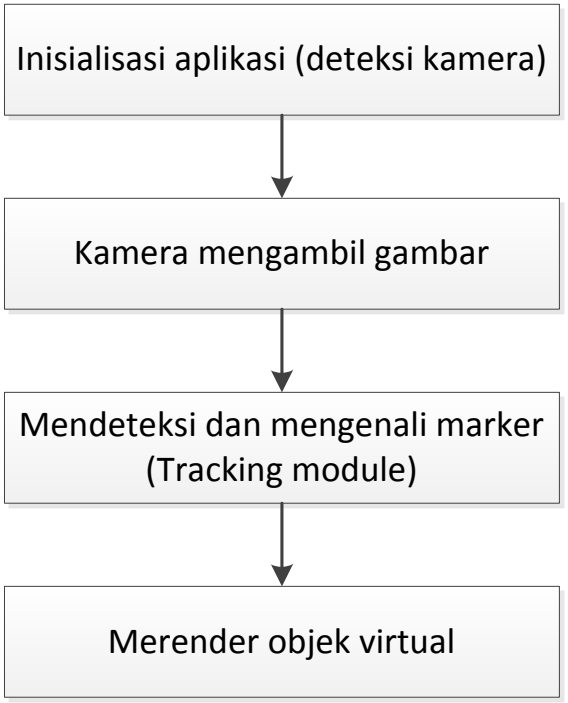
ASPEK	TOKOH	
	Gatotkaca	Arjuna
Ayah	Bima	Prabu Pandudewanata
Ibu	Dewi Arimbi	Dewi Kunti
Saudara Kandung	-	Yudistira Bima
Saudara Tiri	Seayah : Antareja Antasena	Seayah : Nakula Sadewa
Istri	1. Dewi Pregiwa berputra ▪ Sasi Kirana	1. Dewi Subadra, berputra ▪ Raden Abimanyu 2. Dewi Sulastri,

		berputra ▪ Raden Sumitra 3. Dewi Larasati, berputra ▪ Raden Bratalaras 4. Dewi Ulupi atau Palupi, berputra ▪ Bambang Irawan 5. Dewi Jimambang, berputra ▪ Kumaladewa ▪ Kumalasakti 6. Dewi Ratri, berputra ▪ Bambang Wijanarka 7. Dewi Dresanala, berputra ▪ Raden Wisanggeni 8. Dewi Wilutama, berputra ▪ Bambang Wilugangga 9. Dewi Manuhara, berputra ▪ Endang Pregiwa ▪ Endang Pregiwati 10. Dewi Supraba, berputra ▪ Raden Prabakusuma 11. Dewi Antakawulan, berputra ▪ Bambang Antakadewa 12. Dewi Juwitaningrat, berputra ▪ Bambang Sumbada 13. Dewi Maheswara 14. Dewi Retno Kasimpar 15. Dewi Dyah Sarimaya 16. Dewi Srikandi
Nama lain	Bambang Tutuka Kacanegara Senaputra Bimasuta Arimbiyatmaja Krincing Wesi	Janaka Permadi Danajaya Indratanaya Palguna Jahnawi Kumbang ali-ali Suparta

		Margana Pamade
Julukan	**"Otot Kawat Tulang Besi"**	**"Dananjaya"**
Senjata / Pusaka	Kotang Antrakusuma Caping Basunada Terumpah Padakacerma	Keris Pulanggeni Keris Kalanadah Panah Pasopati Panah Sarotama
Watak	Gagah berani Tangguh Teguh Cerdik Gesit Gemar menolong sesama Bekerja tanpa pamrih Berjiwa kesatria	Gagah berani Mulia Tangguh Beriman kuat Bersunguh-sungguh Gemar menolong sesama Tidak sombong Senang bertapa Berjiwa kesatria

3.1.4 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.

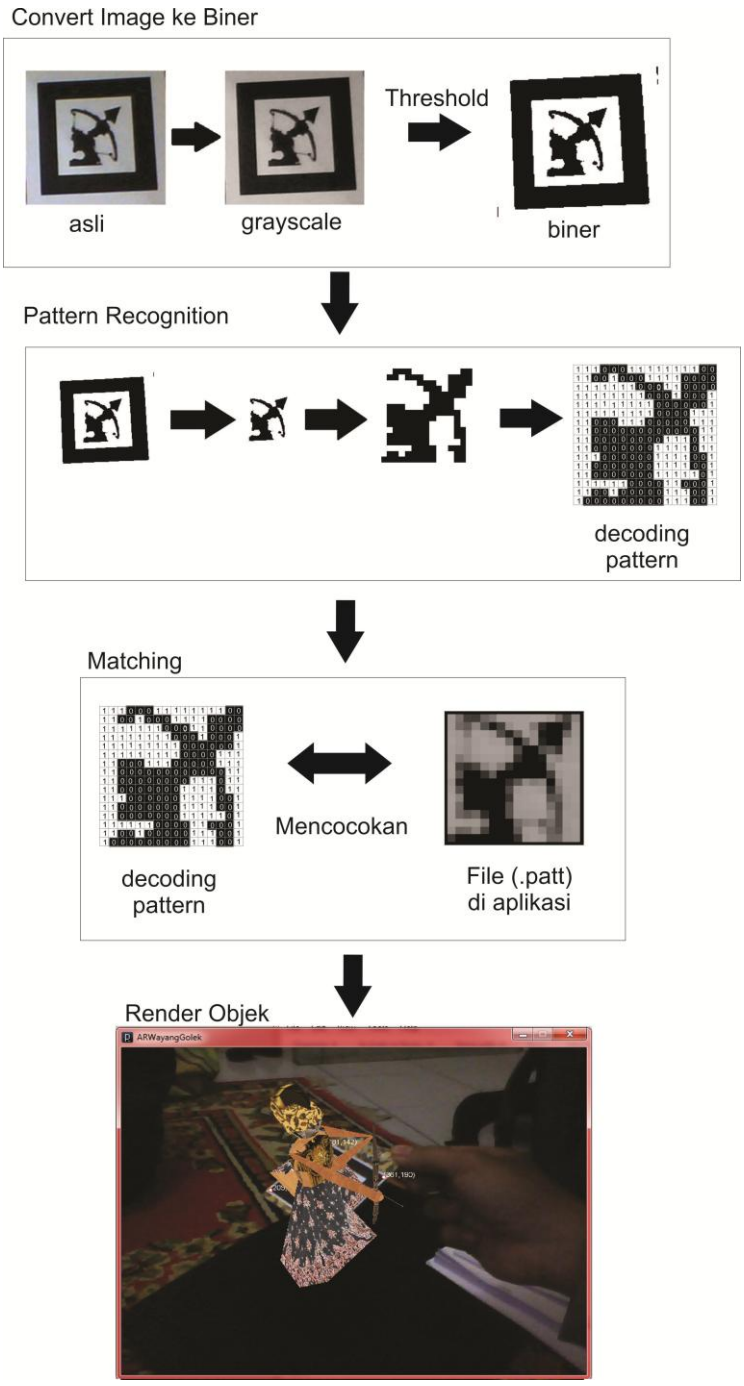


Gambar 3.2 Flowchart AR Sistem

Pada Gambar 3.2 Flowchart AR sistem, 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 mencocokan dengan file (dot)patt di aplikasi bila cocok maka akan merender objek virtualnya.

3.1.4.1 Mendeteksi dan Mengenali Marker

Pada proses mendeteksi dan mengenali marker (Tracking module) sistem memerlukan intensitas gambar (greyscale Image) sehingga gambar yang di dapatkan dari kamera akan dikonversi menjadi grayscale lalu menjadi binary image dengan menggunakan ambang batas *Threshold*. Setelah itu pola bagian dalamnya akan diambil dan dicocokan (Pattern Recognition) lagi dengan *template* pola pada database. Langkah pemrosesannya dapat dilihat pada gambar 3.3.

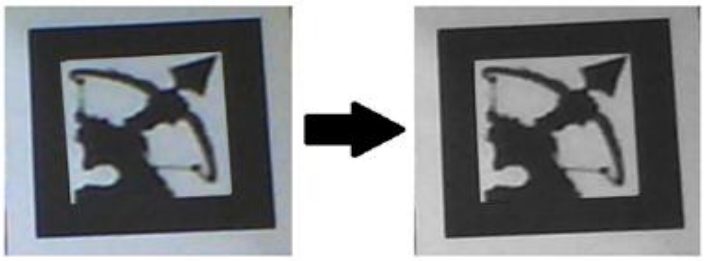


Gambar 3.3 Gambar mendeteksi dan mengenali marker

3.1.4.2 Binary image

Binary image dapat digunakan untuk mendapatkan semua informasi yang dibutuhkan tentang posisi dan bentuk objek. Threshold merupakan metode yang digunakan untuk segmentasi citra menjadi binary image yang baik untuk

mendeskripsikan foreground (objek) dari background.

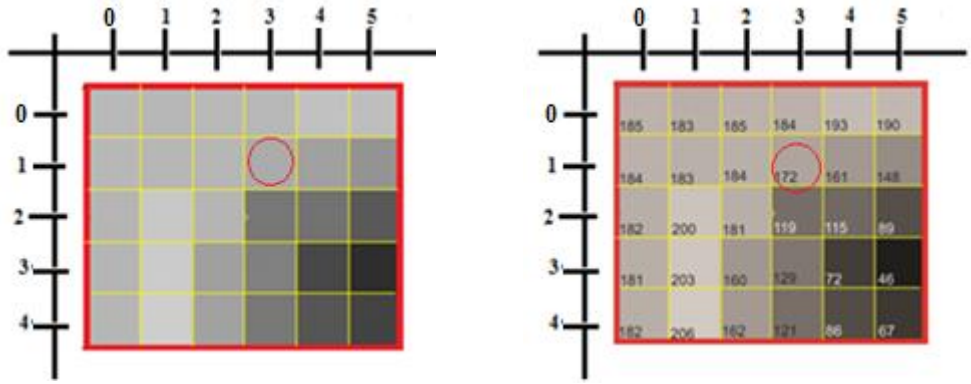


(a) Gambar asli **(b) Gambar Grayscale**

Gambar 3.4 Gambar asli menjadi Gambar Grayscale

Pada gambar 3.4 grayscale yang didapatkan oleh kamera dari dunia nyata, tidak akan menjadi benar-benar hitam atau putih (hitam atau putih murni), maka tidak akan dapat menemukan foreground atau background. Sehingga diperlukan penentuan Threshold grey-level antara dua tingkat dominan, yang berfungsi sebagai ambang batas untuk membedakan foreground(objek) dan background yaitu piksel yang akan menjadi benar-benar hitam atau putih.

Mayoritas piksel milik objek akan memiliki gray-level yang relatif rendah , sedangkan piksel latar belakang akan memiliki gray-level yang relatif tinggi . (Hitam diwakili oleh gray level 0 , dan Putih oleh grey level 255). Nilai dari penentuan ambang threshold didapatkan dari pemindaian piksel demi piksel dan mengetes setiap piksel itu dengan ketentuan ambang threshold itu. Jika $f(x,y) > T$, maka piksel merupakan background piksel. Jika $f(x,y) \leq T$ maka piksel merupakan foreground piksel. Nilai T yang ideal untuk keadaan pencahayaan yang cukup supaya mendapatkan gambar bentuk sebenarnya untuk meningkatkan keakuratan pendeteksian marker ditentukan Default Nilai T berkisar antara 110-120 [10]. Hal ini dapat diringkas dalam definisi berikut, di mana b(x,y) merupakan piksel citra biner thresholded.



Gambar 3.5 Tingkat nilai ke-abuan

Pada Gambar 3.6 terlihat level ke-abuan pada setiap *pixel*. Nilai Treshold 120, maka persamaannya sesuai dengan rumus (2-4) menjadi seperti :

$$b(x,y) = \begin{cases} 255, \text{if } f(x,y) > \mathbf{120} \ \ldots\ldots \mathbf{(1)} \\ 0, \quad \text{if } f(x,y) \leq \mathbf{120} \ \ldots\ldots \mathbf{(2)} \end{cases}$$

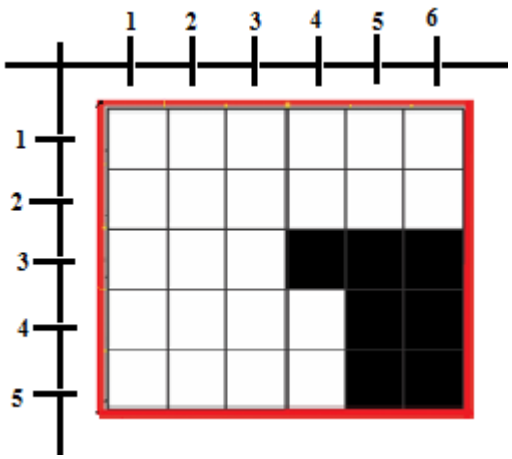
Pada persamaan (1) untuk setiap piksel pada gambar grayscale dengan nilai ke-abuan lebih dari 120, maka akan bernilai 255 (putih) di setiap piksel pada gambar biner. Pada persamaan (2) untuk piksel pada gambar grayscale dengan nilai ke-abuan kurang dari atau sama dengan 120, maka akan bernilai 0 (hitam) untuk piksel pada gambar biner.

Misalnya pada f(3,1) dengan nilai keabuan 172 bila dimasukan ke dalam persamaan

$$b(3,1) = \begin{cases} 255, \text{if } 172 > 120 \\ 0, \quad \text{if } 172 \leq 120 \end{cases}$$

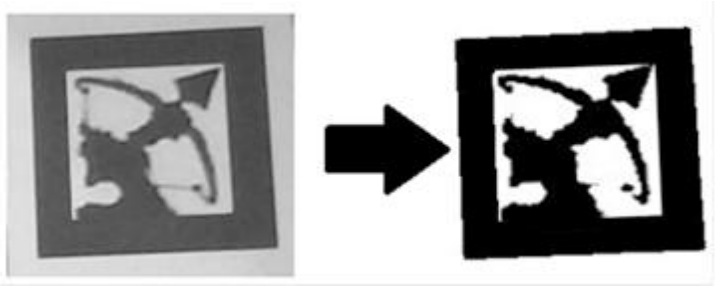
Dengan persamaan nilai keabuan f(3,1) = 172 , nilai b(3,1) akan menjadi 255 atau akan berwarna putih.

Sehingga hasil dari nilai ke-abuan pada gambar 3.5 menjadi gambar biner dengan nilai Threshold(T) = 120 dapat dilihat pada gambar 3.6.



Gambar 3.6 Gambar Biner T=120

Hasil Konversi Gambar grayscale dapat dilihat pada gambar 3.7.



(a) Gambar Grayscale **(b) Gambar Biner**

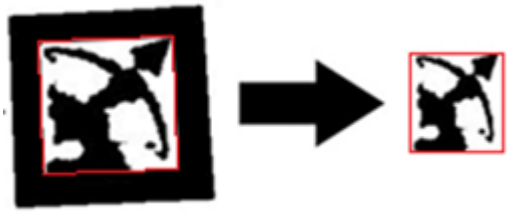
Gambar 3.7 Gambar Grayscale menjadi Gambar Biner

3.1.4.3 Pattern Recognition

Pattern Recognition adalah mendeteksi wilayah di dalam marker setelah marker dideteksi untuk dibandingkan kecocokannya dengan pattern di dalam database sebagai penanda objek virtual.

Proses ini merupakan proses setelah thresholding untuk mendapatkan gambar biner, sehingga gambar yang diproses sudah dalam bentuk biner. Penskalaan pengambilan gambar pattern dari marker dimulai dari piksel yang paling dekat dengan garis terdalam marker seperti terlihat pada gambar 3.8.

Bentuk pattern pada marker dapat berbentuk apa saja, selama tidak terlalu detail seperti foto high definition. Pattern dapat juga berbagai warna, tapi harus asimetris untuk menghasilkan hasil yang baik. Jika pattern cocok 70% dengan pattern yang berada didalam database, objek 3D wayang akan tampil.

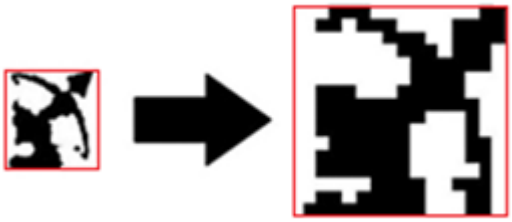


(a) Marker gambar biner (b) Pattern Marker

Gambar 3.8 Identifikasi Marker Pattern

3.1.4.4 Proses Penskalaan Pattern

Penskalaan pattern dirubah menjadi resolusi 16x16 piksel supaya sama dengan resolusi *template* pattern pada tahap *template matching* berikutnya.



(a) Pattern Marker dari gambar biner (b) Pattern Marker 16x16 piksel

Gambar 3.9 Penskalaan Pattern

Langkah penskalaan dilakukan dengan proses :

1. Gambar 3.10 adalah piksel yang dilengkapi dengan koordinatnya sebelum piksel tersebut mengalami proses penskalaan.

(0,0)	(1,0)	(2,0)	(3,0)	(4,0)

Gambar 3.10 Piksel awal sebelum proses penskalaan

2. Gambar 3.11 adalah piksel hasil proses penskalaan

(0,0)	(1,0)	(2,0)

Gambar 3.11 Piksel hasil proses penskalaan

3. Tabel 3.2 adalah perhitungan proses penskalaan dengan menggunakan persamaan (2-7) dan (2-8).

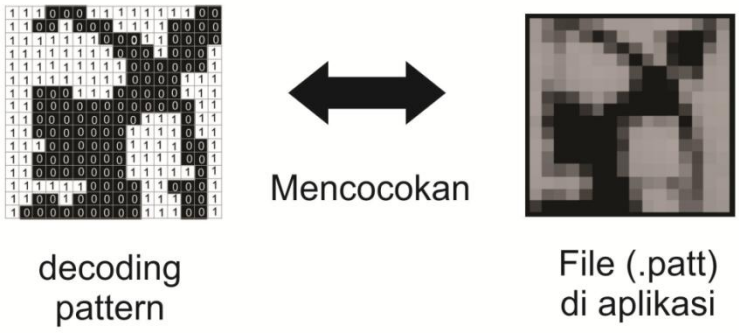
Tabel 3.2 Perhitungan proses penskalaan

Koordinat Pixel Awal		Faktor Pengali		Koordinat Pixel Target	
I	j	dx	dy	x=round(i*dx)	y=round(j*dy)
0	0	0.5	0.5	0	0
1	0	0.5	0.5	1	0
2	0	0.5	0.5	1	0
3	0	0.5	0.5	2	0
4	0	0.5	0.5	2	0

4. Koordinat *pixel* target x dan y adalah koordinat hasil pemetaan dari *pixel* lama terhadap *pixel* target.

3.1.4.5 *Template* matching

Template matching merupakan tahap selanjutnya setelah pattern recognition proses pencocokan gambar hasil penangkapan dari kamera yang telah dikonversi dengan *template* pada aplikasi. Pada proses ini bila *template* cocok dengan pattern objek maka aplikasi akan merender objek 3D wayang golek sedangkan bila *template* cocok dengan pattern aksi maka aplikasi akan memanggil aksi dari wayang golek tersebut ini dapat terlihat bila terdeteksi bersamaan dengan marker objek.

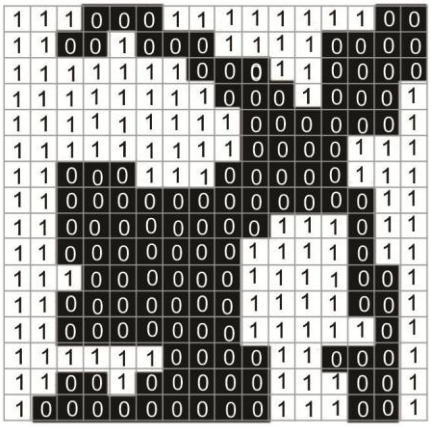


Gambar 3.12 *Template matching*

Pengenalan pola dengan menggunakan metode *template matching* dilakukan dengan cara membandingkan citra masukan dengan citra *template*.

Citra masukan dihitung berdasarkan banyaknya titik yang sesuai dengan citra *template*. *Pixel* citra biner ditelusuri mulai dari kiri atas hingga ke kanan bawah. Decoding Citra biner dengan *pixel* berwarna hitam akan direpresentasikan

dengan nilai 1. Sedangkan *pixel* citra yang berwarna putih akan direpresentasikan dengan nilai 0. Gambar 3.13 adalah gambar yang mengilustrasikan angka 1 dan 0 yang mewakili nilai *pixel* citra.



Gambar 3.13 Contoh decoding pattern

Deretan angka biner pada citra masukan akan dihitung dengan deretan angka biner pada citra *template*. *Template* dengan nilai eror terkecil merupakan *template* citra yang paling sesuai dengan citra masukan.

Penerapan pengenalan marker dengan menggunakan metode *template matching*.

Tabel 3.3 menunjukan deretan angka biner baris pertama citra *template* pada aplikasi, deretan angka biner baris pertama citra masukan dari gambar biner, dan Perhitungan *Error* dengan menggunakan persamaan (2-9).

Tabel 3.3 Perhitungan *Error*

Deretan Angka biner citra *template*	1 1 1 1 0 0 1 1 1 1 1 1 1 0 0 0
Deretan Angka biner citra masukan	1 1 1 0 0 0 1 1 1 1 1 1 1 1 0 0
Perhitungan *Error*	(1-1)^2 + (1-1)^2 + (1-1)^2 + **(1-0)^2** + (0-0)^2 + (0-0)^2 + (1-1)^2 + (1-1)^2 + (1-1)^2 + (1-1)^2 + (1-1)^2 + (1-1)^2 + (1-1)^2 + **(0-1)^2** + (0-0)^2 + (0-0)^2 = **2**

Dari tabel 3.3 ditemukan 2 piksel yang tidak sama (error) sehingga sebesar 2/16 * 100% = 12.5% terjadi error, 14/16 * 100% = 87.5% citra template dan citra masukan cocok atau sesuai.

3.1.5 Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen ataupun komponen-komponen apa saja yang dibutuhkan sistem yang akan dibangun sampai sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang dibutuhkan sistem. Pada analisis kebutuhan sistem non-fungsional ini dijelaskan mengenai analisis perangkat keras (*hardware*), perangkat lunak (*software*) dan pengguna (*user*) .

Aplikasi pengenalan tokoh wayang golek ini merupakan aplikasi berbasis *Augmented reality*. *Aplikasi* penganalan tokoh wayang golek berbasis *Augmented reality* dapat dijadikan alternatif cara mengenalkan wayang golek sebagai budaya indonesia agar dapat dilestarikan, dilindungi, dan dikenal bukan hanya oleh masyarakat indonesia sendiri tetapi seluruh dunia. Karena a*plikasi* pengenalan wayang golek adalah produk berbasis AR yang menggunakan *marker* sebagai pola pelacakan atau *tracking marker*, maka dibutuhkan komputer dan perangkat lunak sebagai pengolah citra dan *marker*. Berikut ini adalah perangkat lunak yang dibutuhkan *aplikasi* pengenalan wayang golek.

3.1.6 Analisis Kebutuhan Perangkat Keras

Aplikasi penganalan wayang golek 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 1 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* terhadap *webcam*. Resolusi minimal webcam 640x480 *pixel*.
7. *Marker*

   Marker dibutuhkan oleh *tracking library* untuk menempatkan model virtual di dunia nyata dengan cara menentukan koordinat *marker* terhadap kamera.

3.1.7 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 CS3
4. Autodesk 3D max
5. JAVA

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

1) Sistem operasi Windows 7
2) JAVA

3.1.8 Analisis Pengguna

Analisis pengguna dimaksudkan untuk mengetahui siapa saja pengguna yang terlibat sistem. Adapun karakteristik dari pengguna adalah sebagai berikut : .

a. Jenjang pendidikan dimulai dari anak sekolah (SD, SMP, SMA), mahasiswa.
b. Pengguna dapat membaca.
c. Aplikasi ini dapat digunakan oleh pengguna yang buta warna.
d. Umur pengguna mulai dari 6 tahun.

3.1.9 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.

3.2 Pemodelan Sistem

Pemodelan sistem dilakukan dengan metode berorientasi Obyek dengan meng0gunakan standarisasi UML, dengan tahap – tahap penentuan siapa saja actor yang terlibat dan perancangan *use case specification, Interaksi diagram (sequence diagram).*

3.2.1 Use Case Diagram

Use case diagram ini digunakan untuk menggambarkan hubungan sejumlah external actor dengan use case yang terdapat dalam sistem. *Use case* diagram ini hanya menggambarkan keadaan lingkungan sistem yang dapat dilihat dari luar oleh actor. Pada use case ini terdapat dua actor, yaitu *user* dan sistem. Penjelasan mengenai dua actor dapat dilihat pada tabel 3.2

Tabel 3.4 Aktor dalam *Use Case*

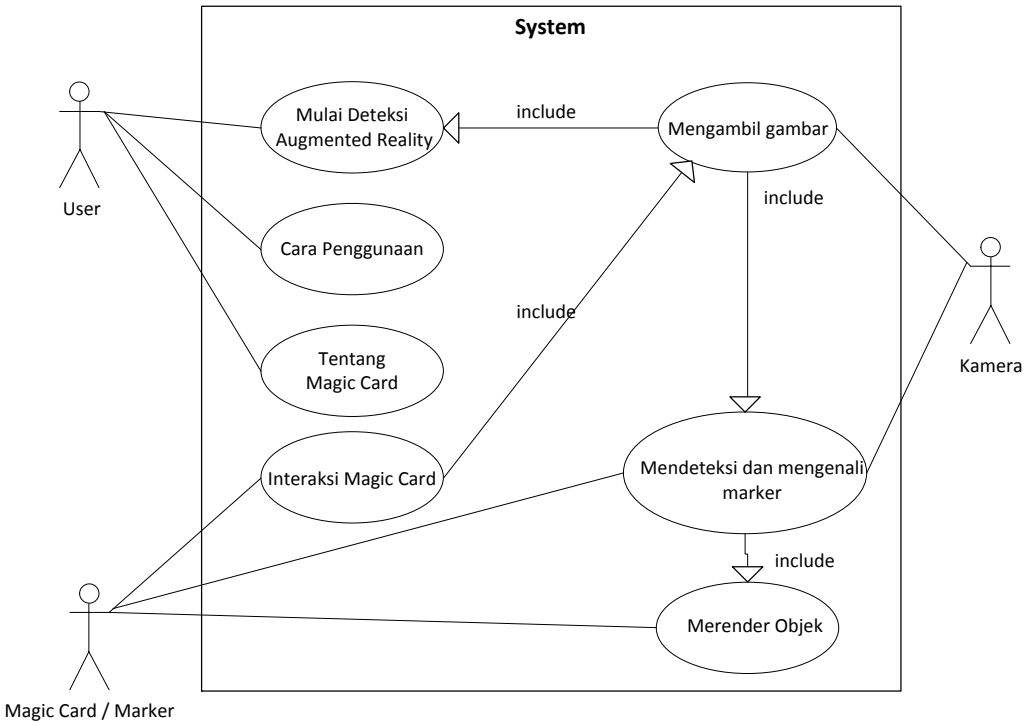
Nama Aktor	Definisi
User	Orang yang berinteraksi dengan sistem, yaitu pengguna yang mengakses semua fungsi yang disediakan sistem.
Marker	Marker adalah alat bantu yang digunakan sebagai penanda dimana obyek virtual 3D akan ditampilkan.
Kamera	Kamera adalah alat bantu yang digunakan sebagai pendeteksi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

3.2.2 Skenario Use Diagram

Use case pada aplikasi berhubungan dengan aplikasi dan penggunaan *augmented reality* pada permaianan. Dimana *augmented reality* digunakan dalam memanipulasi obyek-obyek maya dengan berinteraksi dengan obyek fisik di lingkungan nyata. Agar dapat melihat detail *uses case* dapat lihat pada table 3.3

Diagram *use case* pada gambar 3.5 dilakukan oleh actor dengan *User* dimana diagram *use case* ini lebih bagaimana actor dapat menjalankan fungsi- fungsi yang terdapat pada aplikasi.

a. Jenjang pendidikan dimulai dari anak sekolah (SD, SMP, SMA), mahasiswa.
b. Pengguna dapat membaca.
c. Aplikasi ini dapat digunakan oleh pengguna yang buta warna.
d. Umur pengguna mulai dari 6 tahun.



Gambar 3.14 Use case Diagram

Bagian ini menjelaskan masing-masing deskripsi dari Use *Case* diagram yang dilakukan oleh pengguna. Mulai dari proses memiih masuk kedalam

aplikasi, cara menggunakan aplikasi, dan tentang Magic Card.

Tabel 3.5 Deskripsi Use Case Diagram

NO	Kode *Use Case*	Nama *Use Case*	Keterangan
1	UC-P01	Mulai Deteksi *Augmented reality*	Proses untuk memulai aplikasi AR Wayang Golek
2	UC-P02	Cara Penggunaan	Sistem akan menampilkan Cara Penggunaan Aplikasi.
3	UC-P03	Tentang Magic Card	Sistem akan menampilkan Tentang Magic Card.
4	UC-P04	Interaksi Magic Card	Sistem akan mendeteksi marker aksi yang digerakan dan merender gambar,tekstur dan model geometri sesuai dengan aksi yang dideteksi.
5	UC-P05	Mengambil gambar	Sistem akan mengambil gambar yang ditangkap oleh kamera.
6	UC-P06	mendeteksi dan mengenali marker	Sistem akan mendeteksi dan mengenali marker dari gambar biner.
7	UC-P07	Merender Objek	Sistem akan merender objek 3D atau menjalankan aksi wayang golek bila pattern dan *template* cocok diatas marker.

Tabel 3.6 Spesifikasi Skenario : Mulai Deteksi *Augmented reality*

Nama *Use Case*	Mulai Deteksi *Augmented reality*	
Nomor	UC-P01	
Aktor	*User*	
Kondisi Awal	*User* berada di tampilan awal aplikasi	
Aksi Aktor		**Aksi Sistem**
1. *User* menjalankan Aplikasi *Augmented reality*.		2. Sistem menjalankan Aplikasi *Augmented reality* 3. Sistem mendeteksi ketersediaan kamera. 4. Kamera terdeteksi.
Kondisi Akhir	Aplikasi menampilkan jendela aplikasi *Augmented reality* dan kamera terdeteksi.	

Scenario Alternatif	1. Bila kamera tidak terdeteksi, layar akan berwarna putih dan *user* mencoba mengulang membuka aplikasi. 2. Menampilkan pesan kamera tidak terdeteksi 3. Memilih menu yang lainnya.

Tabel 3.7 Spesifikasi Skenario : Cara Penggunaan

Nama *Use Case*	Cara Penggunaan	
Nomor	UC-P02	
Aktor	*User*	
Kondisi Awal	*User* berada di tampilan awal aplikasi	
Aksi Aktor		**Aksi Sistem**
1. *User* menjalankan Cara Penggunaan		2. Sistem menjalankan Cara Penggunaan
Kondisi Akhir	Aplikasi menampilkan jendela Cara Penggunaan	
Scenario Alternatif	1. Memilih menu yang lainnya.	

Tabel 3.8 Spesifikasi Skenario : Tentang Magic Card

Nama *Use Case*	Tentang Magic Card	
Nomor	UC-P03	
Aktor	*User*	
Kondisi Awal	*User* berada di tampilan Tentang Magic Card	
Aksi Aktor		**Aksi Sistem**
1. *User* menjalankan Tentang Magic Card		2. Sistem menjalankan Tentang Magic Card
Kondisi Akhir	Aplikasi menampilkan jendela Tentang Magic Card	
Scenario Alternatif	1. Memilih menu yang lainnya.	

Tabel 3.9 Spesifikasi Skenario : Interaksi Magic Card

Nama *Use Case*	Interaksi Magic Card	
Nomor	UC-P04	
Aktor	Magic Card/Marker	
Kondisi Awal	*User* sudah menjalankan aplikasi *augmented reality*. Sistem menampilkan jendela *Augmented reality*.	
Aksi Aktor		**Aksi Sistem**

1. Magic Card ditampilkan dihadapan kamera 4. Marker berputar.	2. Sistem mendeteksi magic card. 3. Magic Card terdeteksi . 5. Sistem merender aksi objek 3D wayang golek.
Kondisi Akhir	Objek 3D wayang golek melakukan aksi
Scenario Alternatif	1. Magic Card aksi tidak terdeteksi maka objek tidak melakukan aksi.

Tabel 3.10 Spesifikasi Skenario : Mengambil gambar

Nama *Use Case*	Mengambil gambar
Nomor	UC-P05
Aktor	Kamera
Kondisi Awal	Kamera sudah terdeteksi
Aksi Aktor	**Aksi Sistem**
1. Kamera mengambil gambar.	2. Sistem menerima gambar hasil tangkapan kamera.
Kondisi Akhir	Sistem mendapatkan gambar dari kamera.
Scenario Alternatif	1. Tidak ada gambar diambil menampilkan pesan error.

Tabel 3.11 Spesifikasi Skenario : Mendeteksi dan Mengenali Marker

Nama *Use Case*	Mendeteksi dan Mengenali Marker
Nomor	UC-P07
Aktor	Magic card/Marker, Kamera
Kondisi Awal	Sistem mempunyai gambar dari kamera
Aksi Aktor	**Aksi Sistem**

1. Magic Card/Marker dihadapkan pada kamera 2. Kamera mendapatkan gambar dan mengirim ke system. 10. Marker terdeteksi.	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 9. Sistem mencocokan decoding pattern dengan *template* file (.patt) di aplikasi *(template matching).*
Kondisi Akhir	Marker terdeteksi
Scenario Alternatif	1. Marker tidak terdeteksi, tidak terjadi apa-apa.

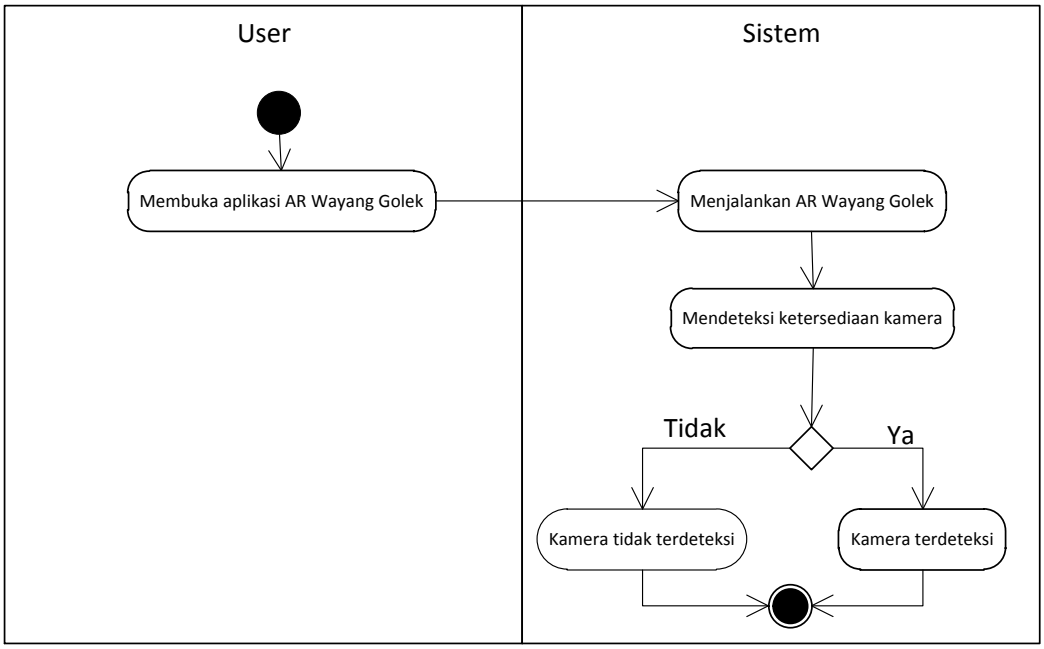
Tabel 3.12 Spesifikasi Skenario: Merender Objek

Nama *Use Case*	Merender Objek	
Nomor	UC-P08	
Aktor	Magic card/Marker	
Kondisi Awal	Marker terdeteksi	
Aksi Aktor		**Aksi Sistem**
2. Objek 3D atau animasi tampil pada bagian atas marker.		1. Merender objek 3D atau animasi yang sesuai dengan *template* dan pattern yang dicocokan.
Kondisi Akhir	Objek 3D wayang golek tampil pada bagian atas marker atau menampilkan animasi wayang golek.	
Scenario Alternatif	1. Objek tidak terender, muncul pesan aplikasi error.	

3.2.3 Activity Diagram

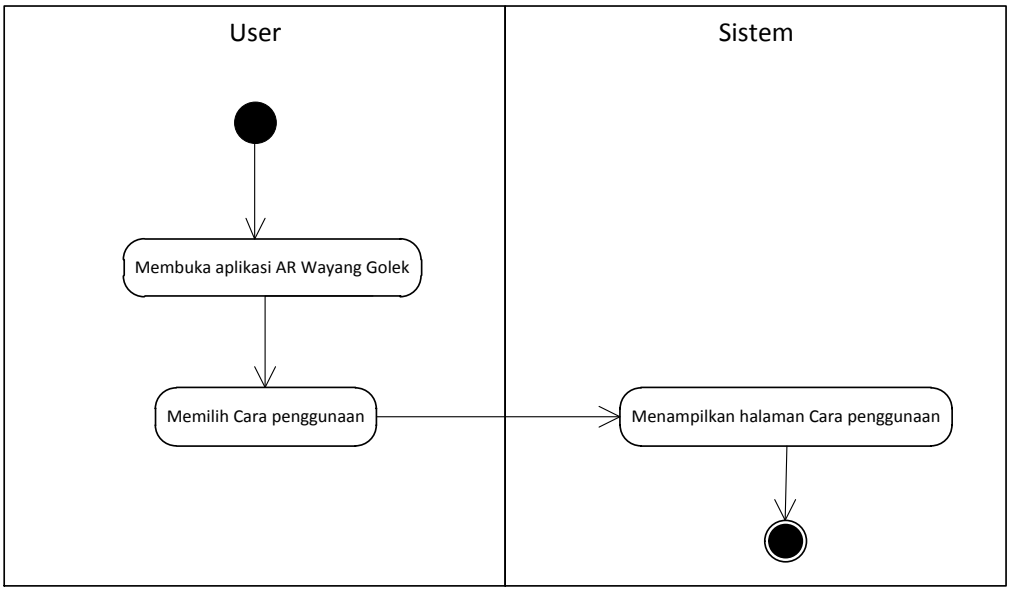
Diagram activity menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi dan bagaimana berakhir. Penggambaran *activity* diagram memiliki kemiripan dengan *flowchart diagram. Activity diagram* memodelkan *event-event* yang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

1. *Activity* diagram terkait dengan Skenario mulai deteksi *augmented reality* dapat dilihat pada gambar 3.12



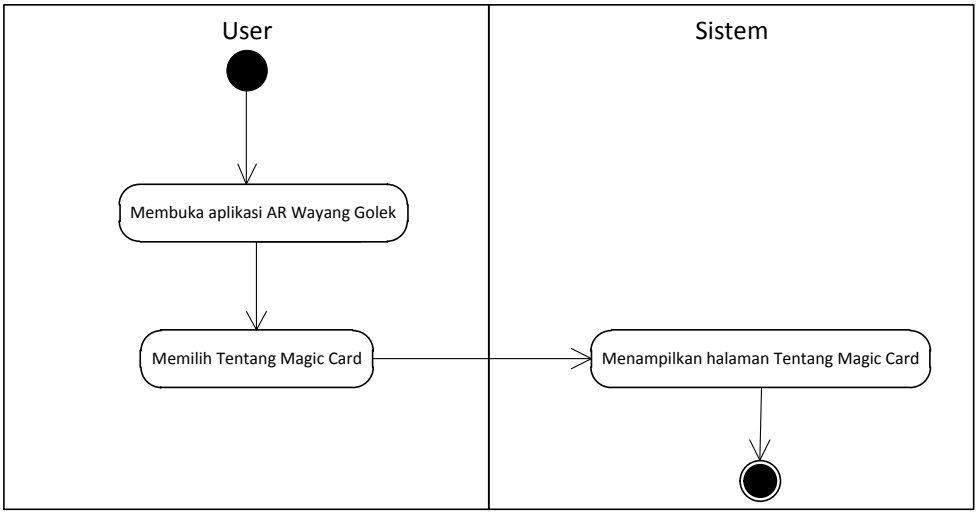
Gambar 3.15 Activity Diagram Mulai Deteksi Augmented reality

2. *Activity* diagram terkait dengan Skenario cara penggunaan dapat dilihat pada gambar 3.13



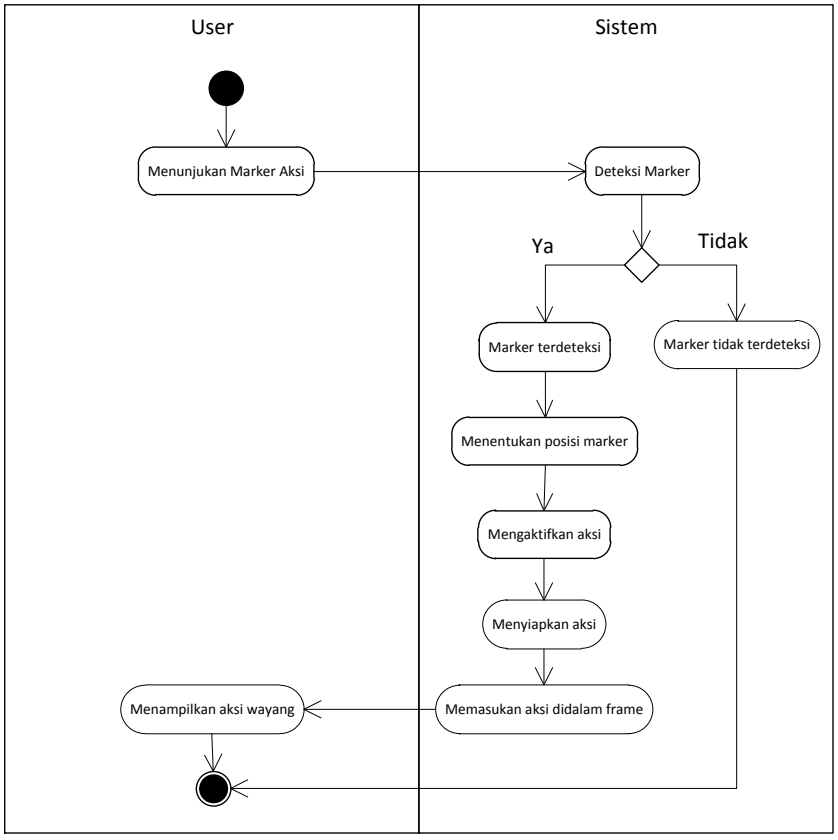
Gambar 3.16 Activity Diagram Cara Penggunaan

3. *Activity* diagram terkait dengan Skenario tentang magic card dapat dilihat pada gambar 3.14



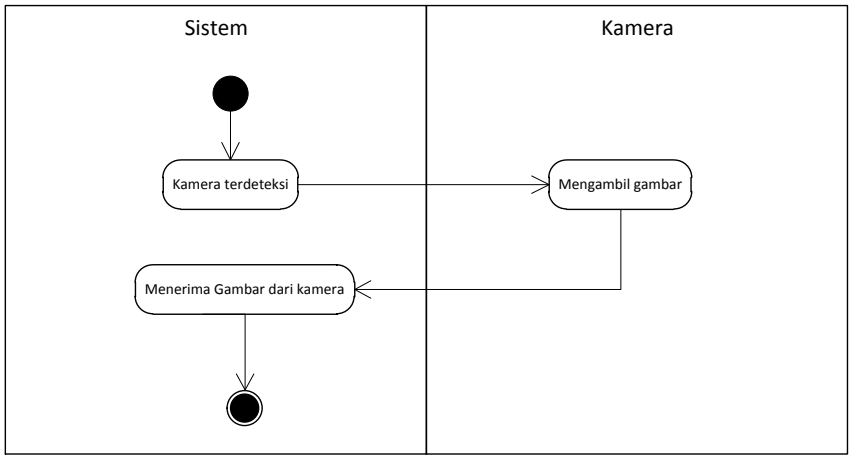
Gambar 3.17 Activity Diagram Tentang Magic Card

4. *Activity* diagram terkait dengan Skenario interaksi magic card dapat dilihat pada gambar 3.15



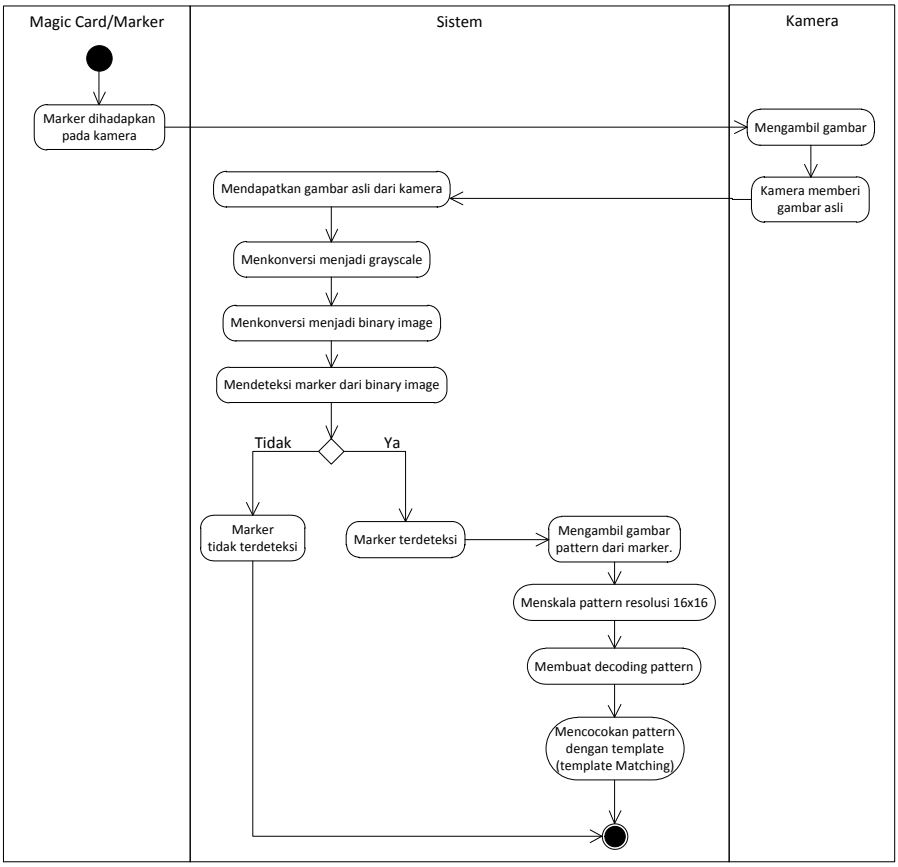
Gambar 3.18 Activity Diagram Interaksi Magic Card

5. *Activity* diagram terkait dengan Skenario mengambil gambar dapat dilihat pada gambar 3.16



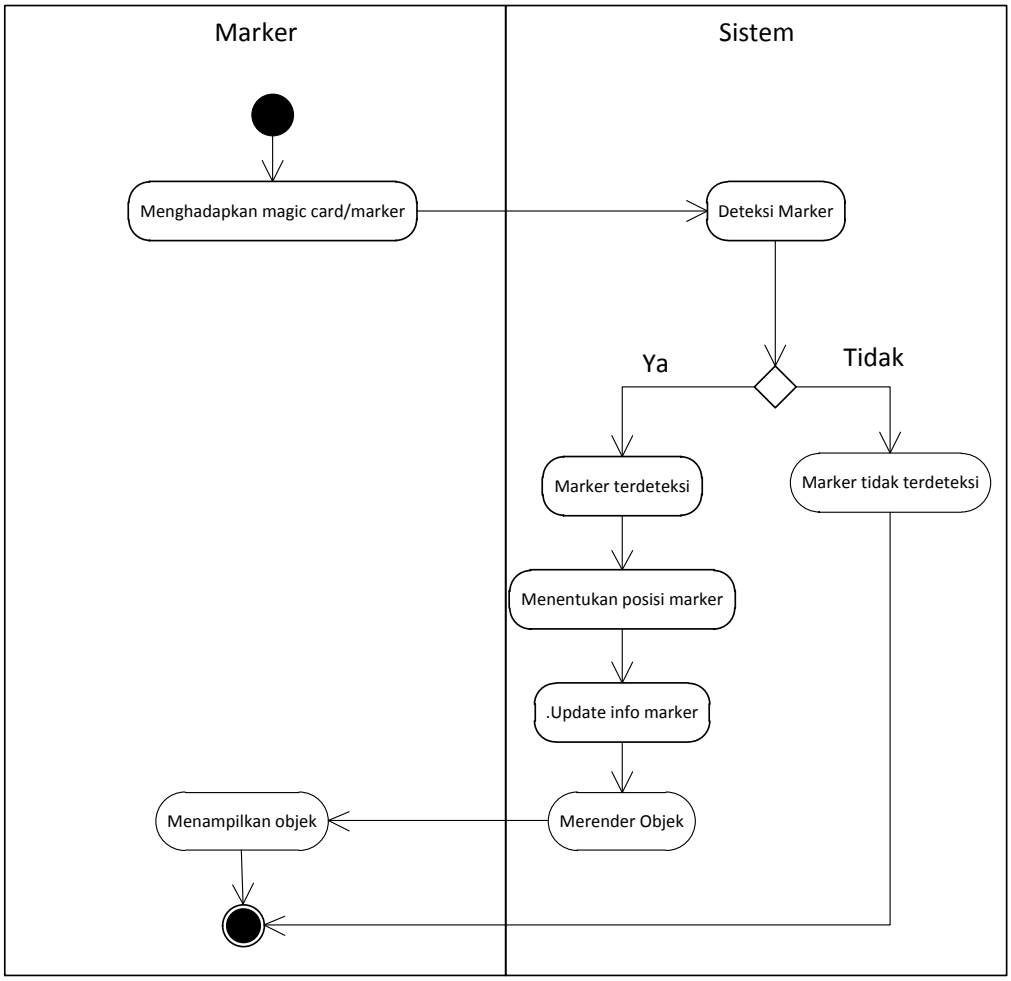
Gambar 3.19 Activity Diagram Mengambil gambar

6. *Activity* diagram terkait dengan Skenario mendeteksi dan mengenali Marker dapat dilihat pada gambar 3.18



Gambar 3.20 Activity Diagram mendeteksi dan mengenali Marker

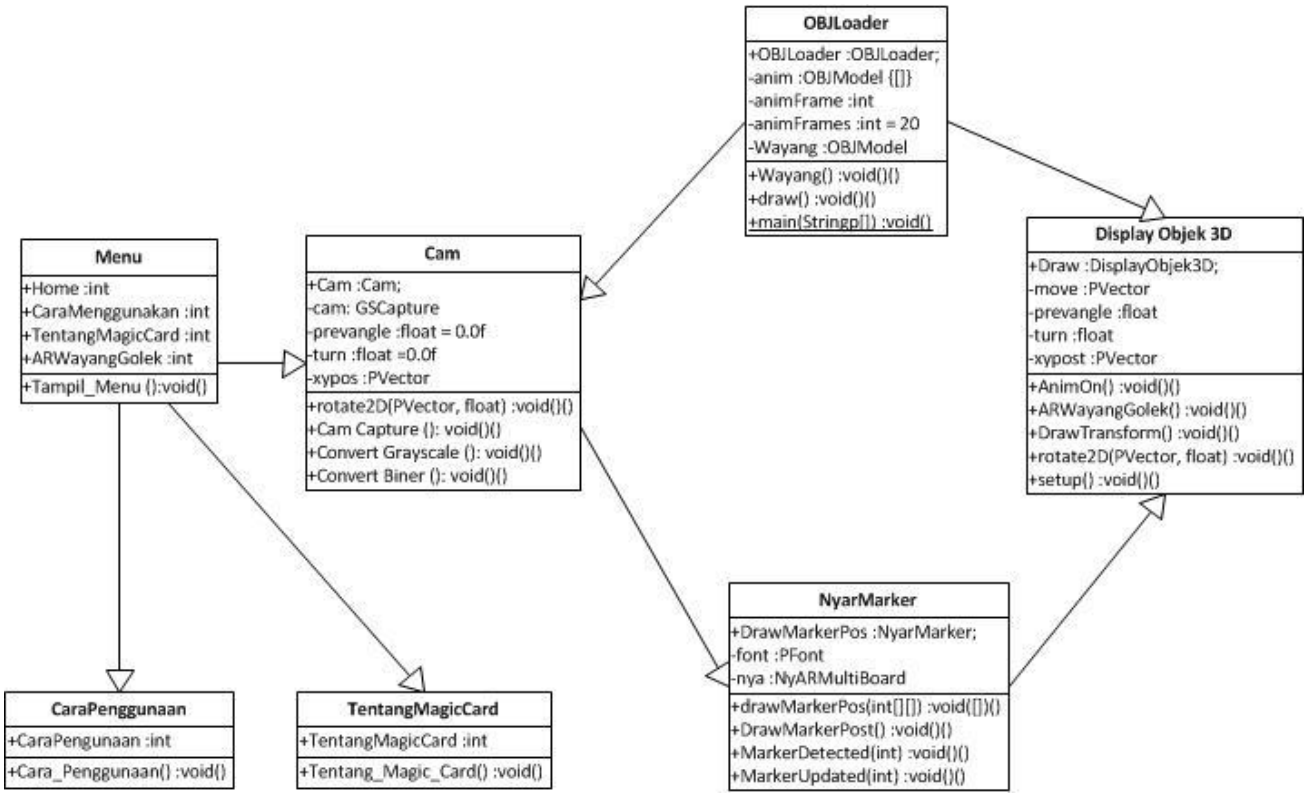
7. *Activity* diagram terkait dengan Skenario Merender Objek dapat dilihat pada gambar 3.19



Gambar 3.21 Activity Diagram Merender Objek

3.2.4 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). Gambar 3.20 adalah kelas diagram dari Implementasi Teknologi Augmented reality Pada Aplikasi Pengenalan tokoh wayang golek.



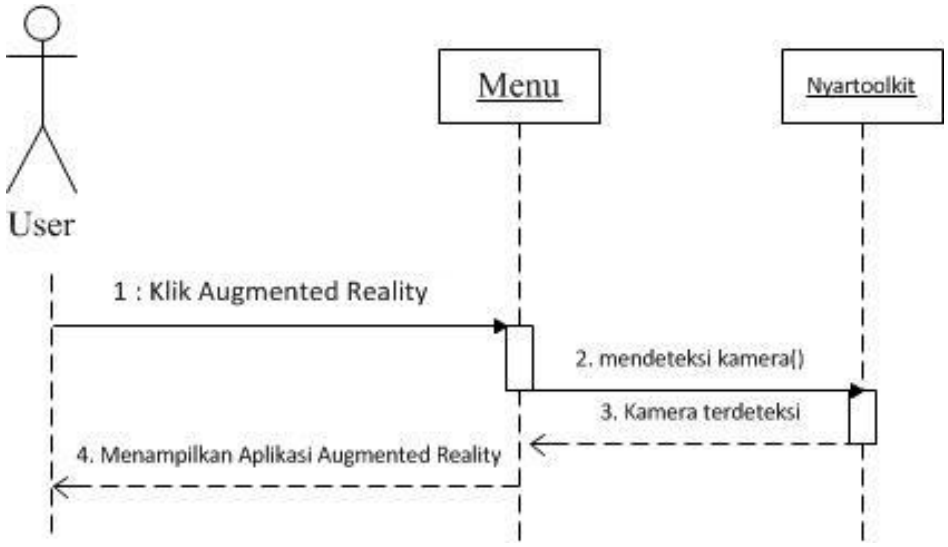
Gambar 3.22 Class Diagram Aplikasi

3.2.5 Sequence Diagram

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

1. *Sequence Diagram* Mulai Deteksi *Augmented reality*..

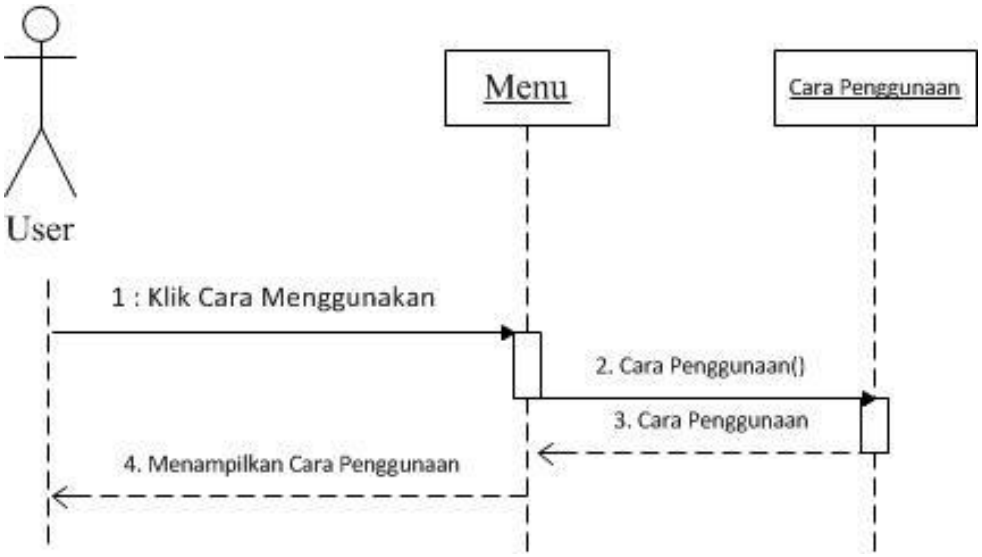
Gambar *Sequence Diagram* Mulai Deteksi *Augmented reality* dapat dilihat pada gambar 3.21



Gambar 3.23 Sequence Diagram Mulai Deteksi *Augmented reality*.

2. *Sequence Diagram Cara Penggunaan*..

Gambar *Sequence Diagram* Cara Penggunaan dapat dilihat pada gambar 3.22.



Gambar 3.24 Sequence Diagram Cara Penggunaan.

3. *Sequence Diagram Tentang Magic Card.*

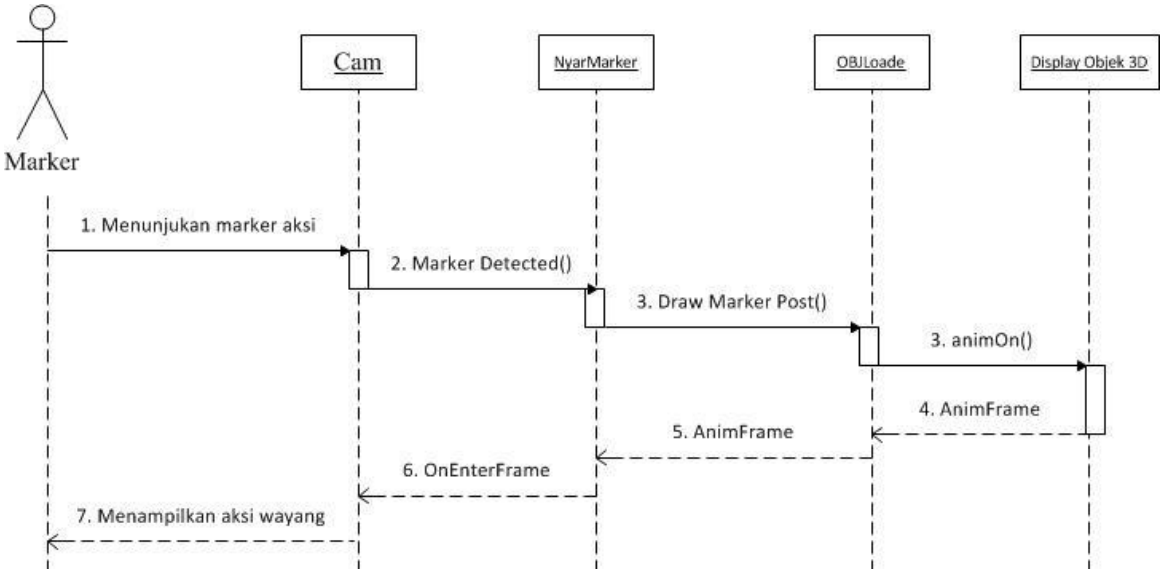
Gambar *Sequence Diagram* Tentang Magic Card dapat dilihat pada gambar 3.23



Gambar 3.25 Sequence Diagram Tentang Magic Card.

4. *Sequence Diagram Interaksi Magic Card.*

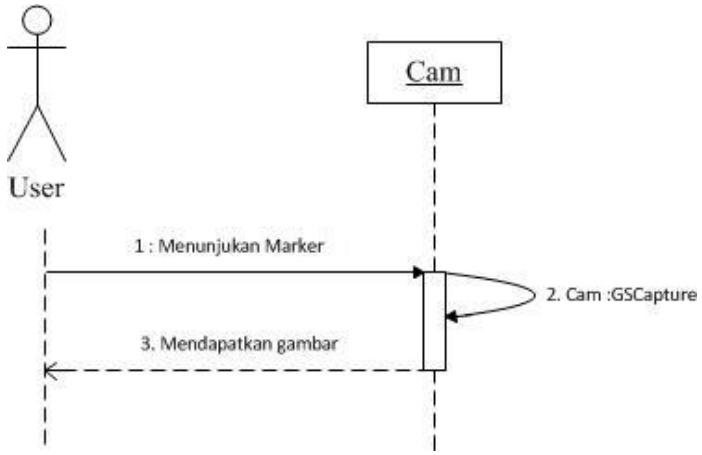
Gambar *Sequence Diagram* Tentang Magic Card dapat dilihat pada gambar 3.24



Gambar 3.26 Sequence Diagram Interaksi Magic Card.

5. *Sequence Diagram Mengambil Gambar.*

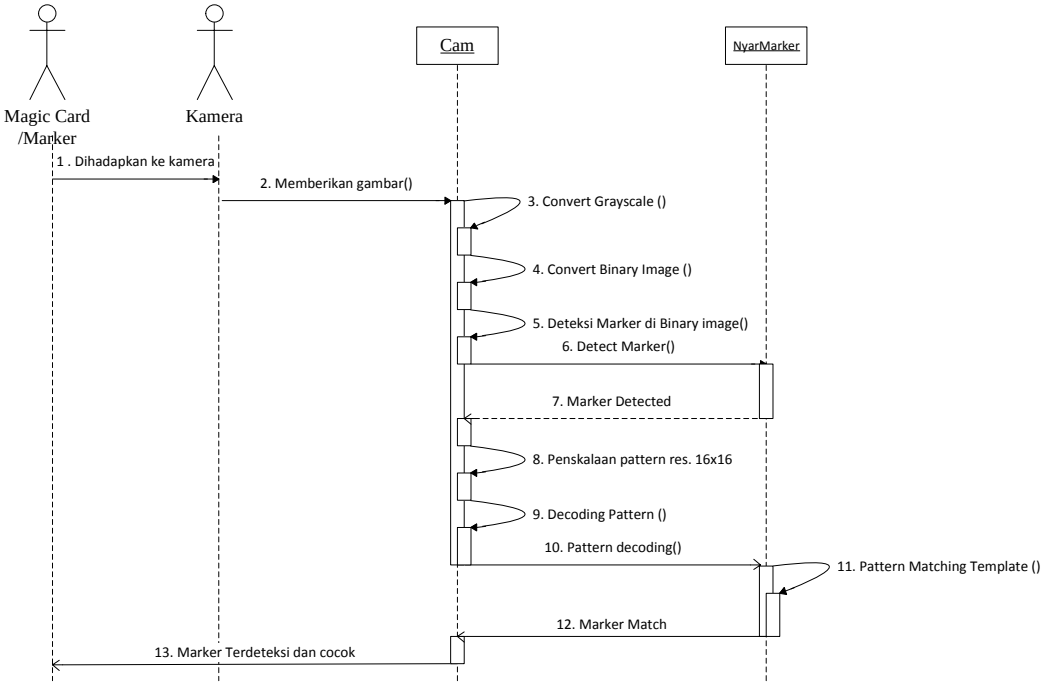
Gambar *Sequence Diagram* Mengambil Gambar dapat dilihat pada gambar 3.25



Gambar 3.27 Sequence Diagram Mengambil Gambar.

6. *Sequence Diagram Mendeteksi dan Mengenali Marker.*

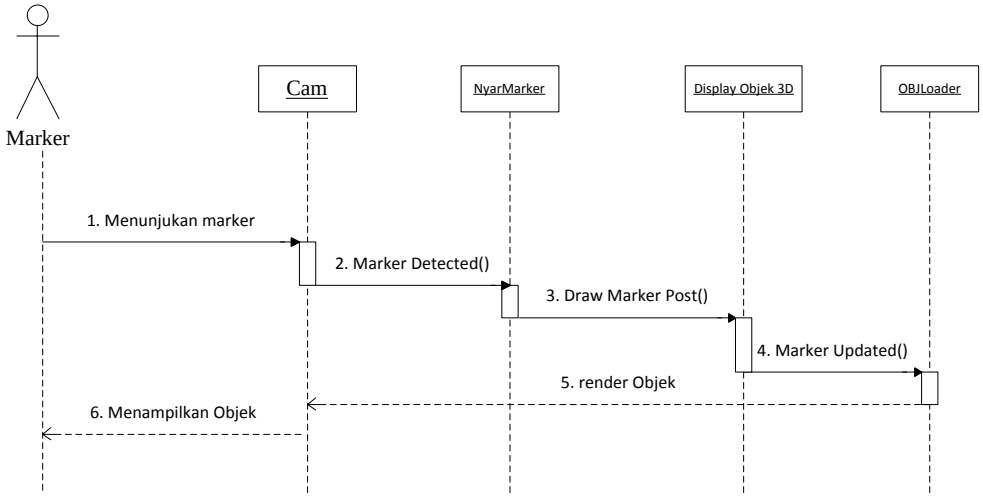
Gambar *Sequence Diagram Mendeteksi dan Mengenali Marker* dapat dilihat pada gambar 3.27



Gambar 3.28 Sequence Diagram Mendeteksi dan Mengenali Marker

7. *Sequence Diagram Merender Objek.*

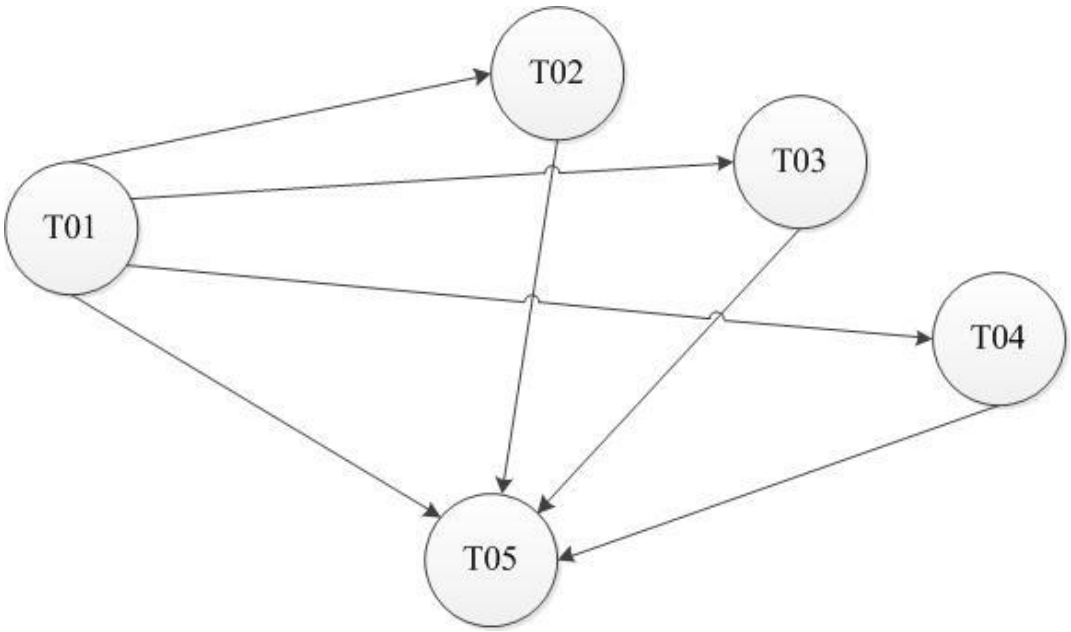
Gambar *Sequence Diagram* Merender Objek dapat dilihat pada gambar 3.28



Gambar 3.29 Sequence Diagram Merender Objek.

3.2.6 Jaringan Semantik

Jaringan semantik merupakan jaringan data dan informasi, yang menunjukan hubungan antar berbagai objek. dapat dilihat pada gambar 3.29



Gambar 3.30 Jaringan Semantik

Keterangan :

T01= Home

T02= Cara Pemakaian

T03= Tentang Magic Card

T04= Aplikasi *Augmented reality*

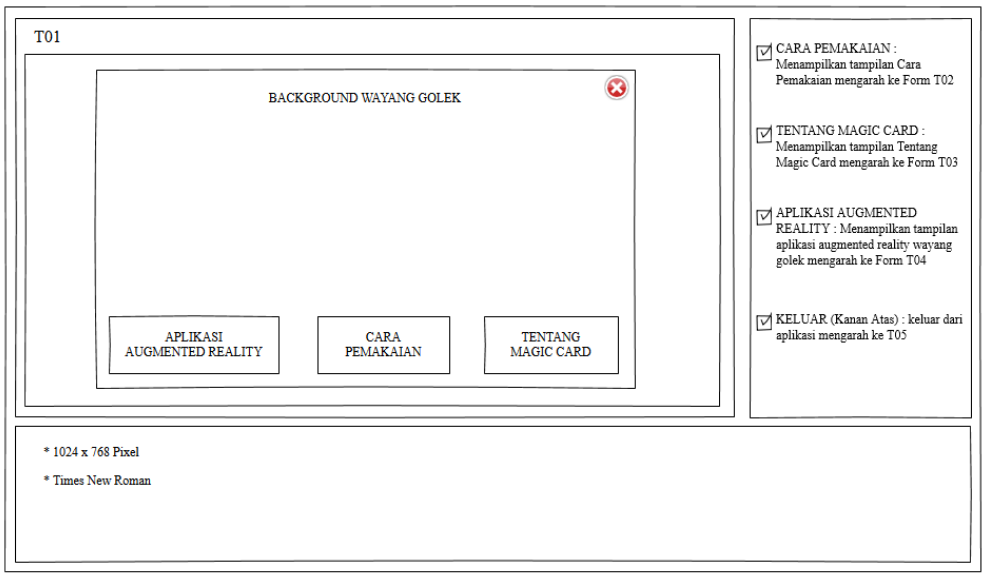
T05= Keluar

3.2.7 Perancangan Sistem

Perancangan sistem akan dimulai setelah tahap analisis terhadap sistem selesai dilakukan. Perancangan dapat di definisikan sebagai proses aplikasi berbagai teknik dan prinsip bagi tujuan pendefinisian suatu perangkat, suatu proses atau sistem dalam detail yang memadai untuk memungkinkan merealisasikan objek. Perancangan digambarkan sebagai proses multi-langkah dimana representasi menu antarmuka dan aplikasi.

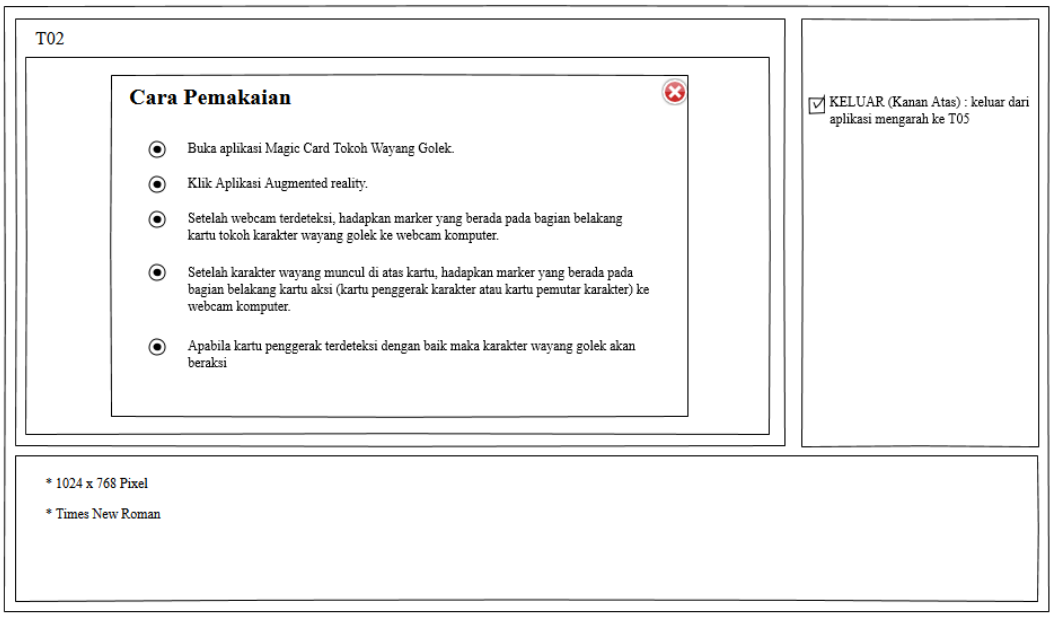
3.2.8 Perancangan Antarmuka

Desain perancangan antarmuka aplikasi dapat dilihat pada gambar 3.30 sampai gambar 3.33



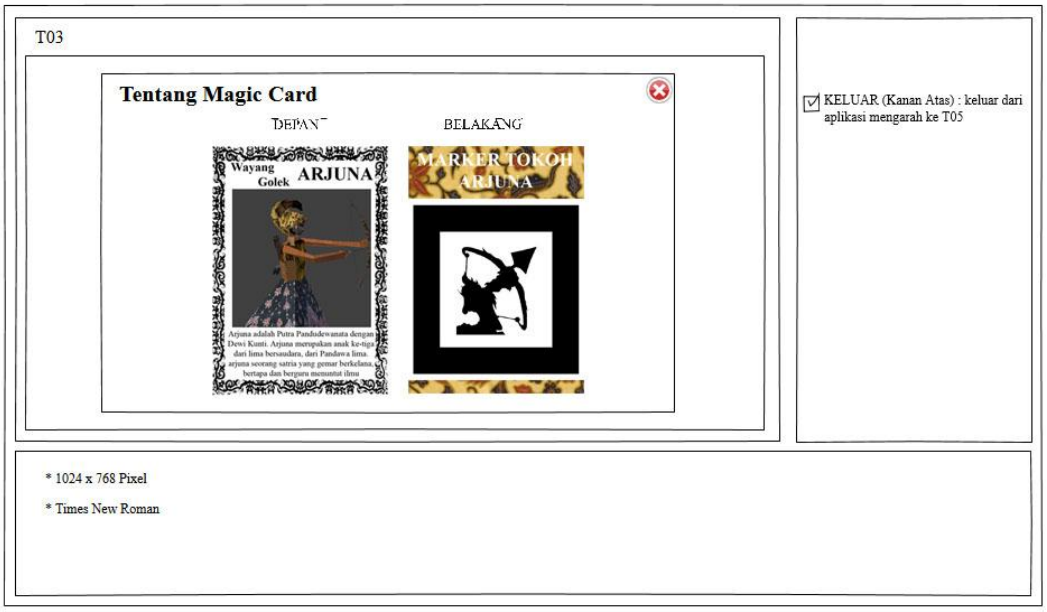
Gambar 3.31 Perancangan Antarmuka Home

Perancangan antarmuka cara menggunakan aplikasi dapat dilihat pada gambar 3.31



Gambar 3.32 Perancangan Antarmuka Cara Menggunakan Aplikasi

Perancangan antarmuka tentang Magic Card dapat dilihat pada gambar 3.32



Gambar 3.33 Perancangan Antarmuka Tentang Magic Card

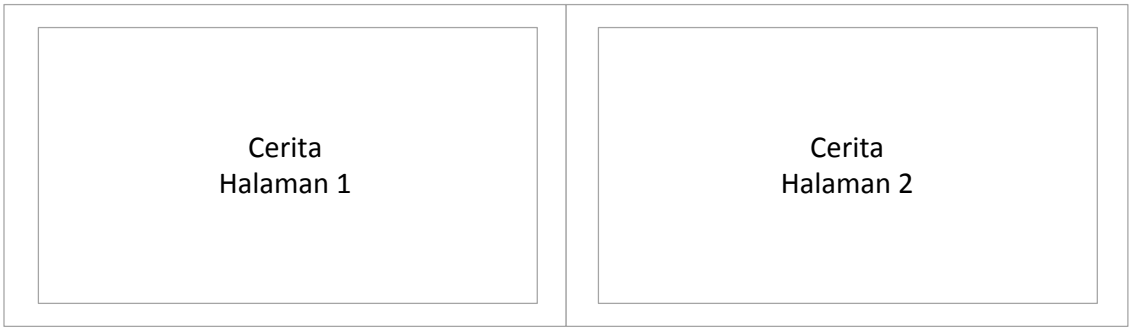
Perancangan antarmuka aplikasi Magic Card Tokoh Wayang Golek *Augmented reality* dapat dilihat pada gambar 3.33



Gambar 3.34 Perancangan Antarmuka *Augmented reality* Wayang Golek

3.2.9 Perancangan Buku Magic Card

Desain perancangan isi buku magic card dapat dilihat pada gambar 3.34 sampai gambar 3.35



Gambar 3.35 Perancangan Buku Magic Card Cerita Wayang Golek

Perancangan isi buku magic card tokoh wayang golek dapat dilihat pada gambar 3.35



Gambar 3.36 Perancangan Buku Magic Card Tokoh Wayang Golek

3.2.10 Perancangan Aplikasi

Daerah area aplikasi berada dalam *marker* yang telah dibuat. Dalam area aplikasi ini terdapat obyek-obyek wayang dalam model 3 dimensi dan dapat mengeluarkan suara apabila marker dihadapkan pada kamera. Aplikasi ini memiliki tujuan yaitu, agar anak dapat balajar dan mengetahui berbagi macam jenis wayang berdasarkan jenis makanan. Untuk mencapai tujuan itu *user* bertindak sebagai pihak yang mengarahkan *marker* ke webcam.

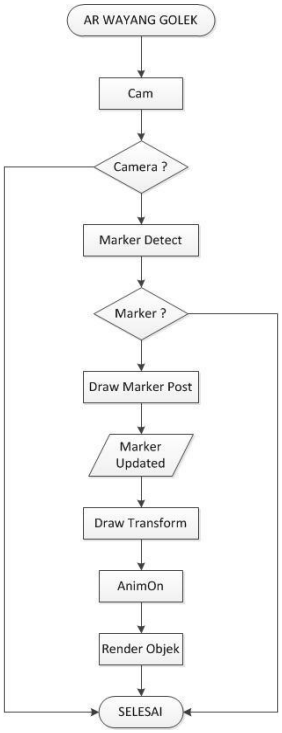
Area aplikasi ini berbentuk sebuah kotak, dimana setiap marker tersebut terdapat suatu obyek wayang. Semakin besar marker maka akan semakin besar pula obyek yang akan di tampilkan begitu pun dengan semakin jauh jarak marker dengan webcam maka semakin kecil pula objek yang akan di tampilkan.

3.2.11 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 wayang golek *augmented reality*. Adapun method-method yang terdapat dalam aplikasi yang akan dibangun adalah sebagai berikut :

1 Perancangan methode AR Wayang Golek

Method cara menggunakan seperti terlihat pada gambar 3.36



Gambar 3.37 Gambar method AR Wayang Golek