

**IMPLEMENTASI TEKNOLOGI AUGMENTED REALITY  
DALAM MEMPERKENALKAN KESENIAN WAYANG  
GOLEK JAWA BARAT SECARA MARKERLESS  
BERBASIS ANDROID**

**TUGAS AKHIR**

Diajukan untuk Memenuhi Salah Satu Syarat Kelulusan  
Program Pendidikan Sarjana

Oleh:

Felicia Yolanda Santosa  
2019130005



**JURUSAN INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER-LIKMI  
BANDUNG  
2022**

**IMPLEMENTASI TEKNOLOGI AUGMENTED REALITY  
DALAM MEMPERKENALKAN KESENIAN WAYANG  
GOLEK JAWA BARAT SECARA MARKERLESS  
BERBASIS ANDROID**

Oleh:  
Felicia Yolanda Santosa  
2019130005

Bandung, 18 Mei 2022  
Menyetujui,

Kezia Stefani, S.T., M.Kom.  
Pembimbing

Dhanny Setiawan, S.T., M.T.  
Ketua Jurusan

**JURUSAN INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER-LIKMI  
BANDUNG  
2022**

## ABSTRAK

Pesatnya perkembangan teknologi saat ini sangat mempengaruhi sebagian besar aspek kehidupan masyarakat di berbagai bidang, khususnya dibidang Pendidikan salah satunya adalah penggunaan media pengenalan dan pembelajaran dengan *augmented reality*. *Augmented reality* memungkinkan pengguna untuk melihat dunia nyata dengan objek maya. Oleh karena itu, *Augmented Reality* ini dapat digunakan sebagai media pengenalan yaitu alat bantu dalam meningkatkan minat generasi muda dalam melestarikan kesenian tradisional dan sebagai media pembelajaran khususnya pada wayang golek Jawa Barat serta memudahkan masyarakat, pengunjung-pengunjung dari luar atau dalam dapat melihat dan mengenal wayang golek lebih efektif dan efisien tanpa harus mengalami kesulitan.

Penyusunan perancangan *augmented reality* menggunakan metode *Markerless* yang merupakan metode yang tidak memerlukan marker untuk menampilkan objek virtual. Dengan menggunakan metode SLAM (*Simultaneous Location and Mapping*) yang berarti *user* atau pengguna dapat mengakses dan menggunakan aplikasi dimana saja dan kapan saja.

Peneliti menggunakan pendekatan SDLC *prototyping* dengan membuat beberapa diagram UML (*Unified Modeling Language*) untuk menguraikan cara kerja dalam aplikasi *augmented reality* yang terdiri dari 3 jenis diagram, yaitu *Use Case Diagram*, *Skenario Use Case*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*. Dalam penelitian ini peneliti menggunakan *software Unity3D* untuk perancangan aplikasi. Peneliti juga menggunakan Blender sebagai desain objek 3D wayang golek. Hasil dari penelitian ini adalah aplikasi tersebut dapat di install pada perangkat android serta tampilan pada aplikasi ini mudah untuk digunakan.

Kata Kunci: *Augmented Reality*, *Markerless*, Wayang Golek, Android, SLAM (*Simultaneous Location and Mapping*)

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yesus Kristus atas segala berkat, kasih karunia, penyertaannya dan kebaikannya sehingga penulis dapat menyelesaikan penyusunan tugas akhir yang berjudul **“Implementasi Teknologi *Augmented Reality* Dalam Memperkenalkan Kesenian Wayang Golek Jawa Barat Secara *Markerless* Berbasis Android”**

Penyusunan Tugas Akhir ini dilakukan untuk memenuhi syarat kelulusan yang diajukan dalam program pendidikan Sarjana jurusan Teknik Informatika, dengan bidang minat grafis dan multimedia di STMIK LIKMI Bandung. Terdapat banyak kekurangan dalam Tugas Akhir ini, tetapi penulis berharap, Tugas Akhir ini dapat memberikan manfaat bagi seluruh pembaca.

Dalam penyusunan tugas akhir ini, penulis juga ingin mengucapkan terima kasih sebesar-besarnya kepada orang-orang yang telah membantu, memberikan masukan, mendukung, memberikan semangat dan membimbing serta mendoakan baik secara langsung dan tidak langsung dalam tugas akhir ini, terutama kepada:

1. Ibu Kezia Stefani, S.T., M.Kom. selaku dosen pembimbing dalam memberikan arahan, memberikan masukan, bimbingan, dan meluangkan waktu kepada penulis yang berguna bagi penulis untuk menyelesaikan Tugas Akhir ini.
2. Bapak Dhanny Setiawan, S.T., M.T., Bapak Sudimanto, S.T., M.Kom. selaku penguji, serta seluruh dosen STMIK LIKMI lainnya yang telah banyak membantu dan memberikan banyak ilmu untuk penulis, untuk staf-staf STMIK LIKMI yang telah banyak membantu penulis, penulis mengucapkan terima kasih banyak.
3. Kepada Papi dan Mami selaku orang tua, Andreas Andi dan Luciana, serta Cici, Cynthia Jocelyne Santosa selaku saudara yang telah mendampingi, mendoakan dan memberikan semangat, serta mendukung selama penulis mengerjakan Tugas Akhir ini.

4. Dan juga orang yang paling dekat dan berarti bagi penulis yaitu Levin Martinus Budiarto yang telah banyak membantu, menghibur, memberikan semangat, mendukung, menemani penulis, penulis ucapkan banyak terimakasih.
5. Sahabat penulis selama di kuliah yaitu Claresta Octaviani yang telah membantu, memberikan semangat, dorongan, dan motivasi.
6. Teman-teman dari SMA (A6) yaitu Vira, Chele, Nia, Denisha, dan Yulia dan beberapa teman SMA lainnya yang telah memberikan semangat kepada penulis.
7. Teman-teman likmi yaitu Kevin, Carel sebagai tim fast track yang telah bersama-sama dalam mengerjakan tugas akhir, dan juga kepada Laurent, Regina, Fedora serta teman-teman lainnya yang sesama jurusan IT yaitu yang sering membantu, memberikan semangat.
8. Teman-teman seperjuangan dan teman-teman STMIK LIKMI yang selalu mendukung, memberikan semangat dan membantu penulis.
9. Semua pihak yang telah membantu secara langsung dan tidak langsung dan yang tidak dapat disebutkan satu per satu oleh penulis.

Penulis menyadari akan adanya kekurangan pada tugas akhir ini. Oleh karena itu, penulis dapat menerima setiap kritik dan saran yang dapat berguna untuk memperbaiki kekurangan pada tugas akhir ini. Penulis berharap agar tugas akhir ini dapat bermanfaat dan berguna bagi setiap rekan dan pembaca sekalian.

Akhir kata, penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah berperan penting dalam membantu penulis dalam menyelesaikan tugas akhir ini.

Bandung, 18 Mei 2022

Penulis

## DAFTAR ISI

ABSTRAK.....	II
KATA PENGANTAR .....	III
DAFTAR ISI.....	V
DAFTAR GAMBAR .....	VIII
DAFTAR TABEL .....	IX
DAFTAR LAMPIRAN .....	X
DAFTAR SIMBOL .....	XI
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian.....	3
1.4 Pembatasan Masalah.....	4
1.5 Kegunaan Hasil.....	4
1.6 Sistematika Penulisan.....	5
BAB II LANDASAN TEORI.....	7
2.1 Rekayasa Perangkat Lunak.....	7
2.2 Metode Penelitian.....	10
2.2.1 Karakteristik metodologi berorientasi objek .....	12
2.2.2 <i>Unified modelling language</i> (UML).....	15
2.3 Model Pengembangan.....	17
2.4 Teori Khusus Topik Tugas Akhir.....	21
2.4.1 <i>Arsitektur Augmented Reality</i> .....	21
2.4.2 <i>Markerless AR</i> .....	23

2.4.3	<i>Markerless SLAM (Simultaneous Localization and Mapping)</i>	24
2.4.4	Wayang Golek	25
2.5	Perangkat Lunak untuk Analisis dan Perancangan Berorientasi Objek	28
2.5.1	Unity 3D	28
2.5.2	Vuforia	30
2.5.3	Android Studio	32
2.5.4	Blender	34
2.5.5	Android	34
2.5.6	Star UML	36
BAB III	ANALISIS DAN PERANCANGAN PERANGKAT LUNAK	37
3.1	Gambaran Umum Perangkat Lunak	37
3.2	Spesifikasi Kebutuhan Perangkat Lunak	38
3.2.1	Kebutuhan <i>Fungsional</i>	38
3.2.2	Kebutuhan <i>non-Fungsional</i>	38
Diagram-Diagram Perancangan Perangkat Lunak		
3.3	<i>Use Case Diagram</i>	39
3.4	Skenario <i>Use Case</i>	40
3.4.1	Skenario <i>Use Case Tracking</i>	40
3.4.2	Skenario <i>Use Case Start AR Wayang Golek</i>	41
3.4.3	Skenario <i>Use Case Help</i>	42
3.4.4	Skenario <i>Use Case Quit</i>	42
3.5	<i>Class Diagram</i>	44
3.6	<i>Sequence Diagram</i>	45
3.7	<i>Activity Diagram</i>	47
3.7.1	<i>Activity Diagram Use Case Tracking</i>	47
3.7.2	<i>Activity Diagram Use Case Start AR Wayang Golek</i>	49
3.7.3	<i>Activity Diagram Use Case Help</i>	51

3.7.4	<i>Activity Diagram Use Case Quit</i> .....	52
3.8	Rancangan Antar Muka.....	53
BAB IV IMPLEMENTASI DAN PENGUJIAN PERANGKAT LUNAK.....		58
4.1	Spesifikasi Kebutuhan Perangkat Keras.....	58
4.2	Pengujian Antar Muka.....	58
4.3	Pengujian Fungsi.....	65
4.3.1	Hasil Pengujian Fungsi Menu Utama.....	65
4.3.2	Hasil Pengujian Fungsi <i>Help</i> .....	65
4.3.3	Hasil Pengujian Fungsi <i>Start AR</i> .....	66
4.3.4	Hasil Pengujian terhadap Lingkungan.....	67
4.3.5	Hasil Pengujian dengan Tipe <i>Smartphone</i> yang Berbeda.....	68
BAB V KESIMPULAN DAN SARAN.....		70
5.1	Kesimpulan.....	70
5.2	Saran.....	71
DAFTAR PUSTAKA.....		72
LAMPIRAN 1 CONTOH TAMPILAN 3D WAYANG GOLEK.....		74
LAMPIRAN 2 LISTING PROGRAM.....		82

## DAFTAR GAMBAR

Gambar 2. 1 UML Diagram 2.5.....	15
Gambar 2. 2 Paradigma Pemodelan <i>Prototype</i> .....	20
Gambar 2. 3 Skema <i>Virtuality Continuum</i> .....	22
Gambar 3. 1 <i>Use Case Diagram</i> .....	39
Gambar 3. 2 <i>Class Diagram</i> .....	44
Gambar 3. 3 <i>Sequence Diagram Start AR</i> .....	45
Gambar 3. 4 <i>Sequence Diagram Help dan Quit</i> .....	46
Gambar 3. 5 <i>Activity Diagram Use Case Tracking</i> .....	48
Gambar 3. 6 <i>Activity Diagram Use Case Start AR Wayang Golek</i> .....	50
Gambar 3. 6 <i>Activity Diagram Use Case Help</i> .....	51
Gambar 3. 7 <i>Activity Diagram Use Case Quit</i> .....	52
Gambar 3. 8 Rancangan Antar Muka <i>Welcome Page</i> .....	53
Gambar 3. 9 Rancangan Antar Muka Menu Utama.....	54
Gambar 3. 10 Rancangan Antar Muka <i>Help</i> .....	55
Gambar 3. 11 Rancangan Antar Muka <i>Start AR</i> .....	56
Gambar 3. 12 Rancangan Antar Muka <i>Start AR– button “Profile”</i> .....	57
Gambar 4. 1 UI <i>Welcome Page</i> .....	59
Gambar 4. 2 UI Menu Utama.....	60
Gambar 4. 3 UI <i>Help</i> .....	61
Gambar 4. 4 UI <i>Start AR</i> .....	63
Gambar 4. 5 Contoh <i>Tracking Wayang Golek AR</i> .....	64
Gambar 4. 6 UI <i>Start AR– button “Profile”</i> .....	64

## DAFTAR TABEL

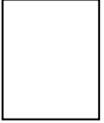
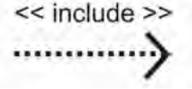
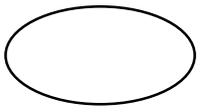
Tabel 2. 1 <i>SDLC Phases and Related Deliverables</i> .....	18
Tabel 3. 1 Skenario <i>use case Tracking</i> .....	40
Tabel 3. 2 Skenario <i>use case Start AR Wayang Golek</i> .....	41
Tabel 3. 3 Skenario <i>use case Help</i> .....	42
Tabel 3. 4 Skenario <i>use case Quit</i> .....	43
Tabel 4. 1 Hasil Pengujian Fungsi Menu Utama.....	65
Tabel 4. 2 Hasil Pengujian Fungsi <i>Help</i> .....	65
Tabel 4. 3 Hasil Pengujian Fungsi <i>Start AR</i> .....	66
Tabel 4. 4 Hasil Pengujian terhadap Lingkungan .....	67
Tabel 4. 5 Hasil Pengujian pada Tipe <i>Smartphone</i> yang Berbeda.....	68

## DAFTAR LAMPIRAN

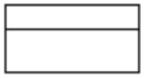
LAMPIRAN 1: CONTOH TAMPILAN 3D WAYANG GOLEK.....	74
1. Objek 3D Wayang Golek Semar.....	74
2. Objek 3D Wayang Golek Cepot.....	76
3. Objek 3D Wayang Golek Hanoman.....	79
LAMPIRAN 2: CONTOH <i>LISTING PROGRAM</i> .....	82
1. <i>AppManager.cs</i> .....	82
2. <i>ArManager.cs</i> .....	82
3. <i>Backpress.cs</i> .....	83
4. <i>FadeEffectScene.cs</i> .....	84
5. <i>RotateObject.cs</i> .....	84
6. <i>ShowHideInfo.cs</i> .....	85
7. <i>ShowHideObject.cs</i> .....	86
8. <i>SliderMenuAnim.cs</i> .....	86
9. <i>SplashFade.cs</i> .....	86
10. <i>ZoomObj.cs</i> .....	87

## DAFTAR SIMBOL

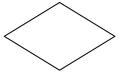
### 1. Simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Association</i>	Garis yang menghubungkan antara objek satu dengan objek lainnya.
3		<i>System Boundary</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5		<i>Use Case</i>	Menspesifikasikan sebuah fungsi tertentu yang disediakan oleh sistem dimana terjadi pertukaran pesan antar anggota sistem.

### 2. Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2		<i>Association</i>	Garis yang menghubungkan antara objek satu dengan objek lainnya.
3		<i>Directed Association</i>	Garis yang menghubungkan antara objek satu dengan objek lainnya di mana objek yang dituju tidak mengetahui relasi tersebut.
4		<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya

### 3. Simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
2		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
3		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Decision</i>	Suatu titik pada <i>activity diagram</i> yang mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi
5		<i>Transition</i>	Menunjukkan kegiatan berikutnya dan arah jalur sistem.

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Indonesia adalah negara yang memiliki suku, kesenian, bahasa, dan budayanya budayanya seperti tarian, batik, musik tradisional, wayang, gamelan yang sangat beragam dari sabang sampai merauke (Widiastuti, 2013). Kebudayaan itu sendiri merupakan hak cipta, karsa, dan rasa, dan berarti mengolah sehingga mempengaruhi tingkat pengetahuan, sistem ide maupun gagasan yang terdapat dalam pikiran manusia. Adanya budaya lokal pada sisi lain yang mengalami kemajuan sangat pesat, dalam bidang pengembangan ilmu pengetahuan dan teknologi tentunya berdampak bagi kehidupan masyarakat Indonesia.

Adanya kebudayaan asing yang bersifat negatif memasuki kehidupan bangsa Indonesia, terutama para generasi muda tanpa adanya upaya pelestarian nilai-nilai budaya maka dikhawatirkan Bangsa Indonesia akan kehilangan jati dirinya. Bukti lemahnya masyarakat Indonesia dalam upaya pelestarian budaya terlihat dari minimnya minat untuk mempelajari kesenian tradisional atau daerah yang saat ini sudah hampir dilupakan oleh generasi muda. Khususnya kaum muda jaman sekarang lebih suka kepada kebudayaan asing dibanding dengan menikmati berbagai seni pertunjukan tradisional yang sebelumnya akrab dengan kehidupan mereka dan tanpa disadari kebudayaan asing malah lebih mendominasi. Dan dari banyaknya keberagaman budaya Indonesia ada beberapa budaya yang hampir punah hampir ditinggalkan oleh masyarakat contohnya seperti wayang. Wayang adalah suatu karya seni rupa yang mempunyai makna, lambang dan simbol bagi falsafah hidup bagi anggota masyarakat pendukungnya (Sunarto, 1989).

Wayang Golek adalah salah satu kesenian tradisional Sunda provinsi Jawa Barat, khususnya sangat populer di wilayah tanah Pasundan, yang merupakan karya seni adiluhung sebagai perpaduan dari berbagai karya seni, ada seni ukir, seni tutur, seni peran, dan seni pertunjukan. Wayang Golek terbuat dari kayu yang dimainkan oleh manusia yang disebut dalang. Kesenian ini dipentaskan pada acara tertentu, dipentaskan salah satu contohnya di sebuah Padepokan.

Pada setiap media untuk pembelajaran memiliki karakteristik yang berbeda oleh karena itu perlu dipahami dan dipilih media mana yang sesuai dengan kebutuhan dan kondisi yang ada. Maka untuk meningkatkan minat generasi muda dalam melestarikan kesenian tradisional dan sebagai media pembelajaran khususnya pada wayang oleh karena itu dibuatlah aplikasi untuk memperkenalkan tokoh wayang berbasis Augmented Reality (AR) secara markerless dengan menggunakan android.

*Augmented Reality* merupakan suatu teknologi yang menggabungkan atau melengkapi benda dua dimensi dan tiga dimensi (Bacca, et al,2014) pada dunia nyata dengan lingkungan virtual yang dibuat oleh komputer secara real time (Azuma, 1997). Dan pada saat ini terdapat berbagai macam teknik *Markerles Tracking*, seperti *Face Tracking*, *3D Object Tracking*, dan *Motion Tracking*. Berjalanya waktu teknologi *Augmented Reality* (AR) berkembang begitu cepat dan ada enam aplikasi Augmented Reality yang telah dikembangkan (Azuma, 1997), contohnya seperti untuk Medis, Manufaktur dan Reparasi, Anotasi dan Visualisasi, Perencanaan Jalur Robot, Hiburan, dan Navigasi dan Penargetan Pesawat Militer.

Oleh karena itu penulis memiliki solusi dalam mengenalkan wayang golek kepada pengunjung dan sebagai media pembelajaran dengan melihat sebuah objek 3D wayang golek Jawa Barat dari Smartphone Android menggunakan teknologi *Augmented Reality* (AR). Menggunakan Android karena peranan Android saat ini sangat penting bagi sebagian masyarakat Indonesia. Dan dengan adanya penggunaan teknologi Android, dapat dimanfaatkan untuk mengenalkan *Augmented Reality* (AR) wayang golek 3D kepada masyarakat pada umumnya dan pengunjung khususnya.

Maka dari itu, dalam pembuatan tugas akhir ini penulis melakukan analisa terhadap permasalahan yang ada pada kesenian wayang golek Jawa Barat yang sudah hampir punah hampir ditinggalkan oleh masyarakat dan untuk memperkenalkan kesenian wayang kepada pengunjung-pengunjung yang berdatangan ke Indonesia, Jawa Barat. Pada penelitian ini, penulis akan membangun sebuah sistem atau aplikasi *Augmented Reality* secara *real-time* dan berbasis android yang tanpa menggunakan marker atau disebut dengan *markerless augmented reality* dengan metode SLAM (*Simultaneous*

*Localization and Mapping*) agar masyarakat dan pengunjung dapat melihat dan mengenal wayang golek lebih efektif dan efisien tanpa harus mengalami kesulitan. Oleh karena itu, dirancang suatu sistem aplikasi "**Implementasi Teknologi *Augmented Reality* Dalam Memperkenalkan Kesenian Wayang Golek Jawa Barat Secara *Markerless* Berbasis Android**".

## **1.2 Rumusan Masalah**

Hasil analisa dan pengamatan yang dilakukan penulis pada kesenian wayang golek Jawa Barat, maka penulis dapat menemukan permasalahan yang timbul di kesenian wayang golek ini seperti:

1. Bagaimana cara meningkatkan minat dan meningkatkan pelestarian masyarakat serta kaum generasi muda dalam mengenal kesenian wayang golek Jawa Barat?
2. Bagaimana cara agar masyarakat, kaum generasi muda dan pengunjung-pengunjung dari dalam atau luar tidak mengalami kesulitan dalam mengenal dan melihat bentuk wayang golek?
3. Terkadang dari internet hanya menampilkan objek berbentuk 2D. Bagaimana caranya dalam meningkatkan visualisasi realitas wayang golek Jawa Barat?

## **1.3 Tujuan Penelitian**

Tujuan dari penelitian ini adalah untuk merancang serta membangun sebuah aplikasi untuk memperkenalkan Wayang Golek dengan menggunakan *augmented reality* berbasis android:

1. Dapat melestarikan tradisi dan kebudayaan dengan memanfaatkan teknologi saat ini dan menarik minat masyarakat untuk lebih mengetahui dan mengenal tentang Wayang Golek.
2. Masyarakat dan pengunjung dapat melihat dan mengenal wayang golek lebih efektif dan efisien tanpa harus mengalami kesulitan, memudahkan masyarakat yang mengunjungi kota Jawa Barat atau tanpa mengunjungi kota Jawa Barat dalam memperoleh informasi budaya Wayang Golek. Membuat kesenian wayang golek Jawa

Barat semakin luas dan semakin dikenal khalayak ramai mulai dari pengunjung dari luar maupun dalam.

3. Dengan adanya aplikasi ini dan pengimplementasian *Augmented Reality Markerless* ini dapat menampilkan objek berbentuk 3D dan dapat memvisualisasi realitas wayang golek.

#### **1.4 Pembatasan Masalah**

Batasan yang ada dalam sistem pada tugas akhir ini adalah:

1. Wayang yang digunakan berasal dari Sunda provinsi Jawa Barat, Pasundan,
2. Wayang yang akan dibuat dalam aplikasi ini 3 (Cepot, Semar, Hanoman)
3. Teknologi *Markerless Augmented Reality* dengan mendeteksi dari lensa camera yang dilihat sudut camera terhadap bidang datar dari jarak yang paling dekat dengan bidang datar, lalu lama-lama menjauh untuk menentukan besar kecilnya wayang golek yang ditracking.
4. Metode pendeteksian objek menggunakan *markerless augmented reality* yang ditampilkan berupa gambar 3d.

#### **1.5 Kegunaan Hasil**

Pengenalan wayang golek dengan *Augmented Reality secara Markerless* ini diharapkan dapat memberikan manfaat kepada pengguna aplikasi. Diantaranya manfaatnya antara lain:

1. Dengan menggunakan aplikasi *Augmented Reality* yang memanfaatkan teknologi saat ini dapat menarik minat masyarakat untuk lebih mengetahui, belajar dan mengenal tentang wayang golek.
2. Membuat kesenian wayang golek Jawa Barat semakin luas dan semakin dikenal khalayak ramai mulai dari pengunjung dari luar maupun dalam secara efektif dan efisien.

## 1.6 Sistematika Penulisan

Tugas akhir ini tersusun dalam beberapa bab, yang pada dasarnya setiap bab membahas dan menguraikan bahasan yang berbeda-beda. Penulis melakukan dan membuat sistematika penulisan pada tugas akhir ini sebagai berikut:

### BAB I PENDAHULUAN

Bab ini berisikan mengenai latar belakang yang dilakukan penulis dalam bab ini berisi latar belakang, identifikasi masalah, tujuan penelitian, pembatasan masalah, metode penelitian.

### BAB II LANDASAN TEORI

Bab ini berisikan tinjauan pustaka menguraikan landasan teori, kerangka pikir dari acuan yang mendasari dalam kegiatan penelitian pada tugas akhir ini. Berisikan tentang teori dan konsep terkait penelitian, yaitu tentang Wayang, *Augmented Reality*, *Metode Markerless Augmented Reality*, *Metode SLAM*, dan *Android*.

### BAB III ANALISIS DAN PERANCANGAN

Bab ini berisikan tentang analisis suatu permasalahan dan perancangan struktur aplikasi, membuat diagram-diagram perancangan perangkat lunak yaitu *use case* diagram, skenario *use case*, *activity* diagram, *class* diagram serta rancangan antar muka.

### BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisikan tentang uraian tentang hasil penelitian dan pembahasan terhadap hasil penelitian yang dilakukan dan akan memaparkan hasil terhadap uji coba pendekatan yang telah dilakukan dalam membangun aplikasi *Augmented Reality* secara *Markerless* untuk pengenalan wayang golek berbasis android.

### BAB V KESIMPULAN DAN SARAN

Bab ini berisikan tentang kesimpulan dari keseluruhan penelitian yang telah dilakukan, serta saran dari penulis untuk kegiatan penelitian selanjutnya terkait dengan topik yang sedang dibahas dan saran yang dapat diusulkan untuk pengembangan suatu sistem.

### DAFTAR PUSTAKA

Berisikan daftar buku – buku yang menjadi gambaran penulis dalam pembuatan *Augmented Reality Markerless* untuk pengenalan wayang golek berbasis android.

## BAB II

### LANDASAN TEORI

#### 2.1 Rekayasa Perangkat Lunak

Menurut deskripsi dari buku "*Software Engineering: A Methodical Approach*", pengertian rekayasa perangkat lunak ialah: "*Software engineering is the process by which software systems are investigated, planned, modeled, developed, implemented and managed.*" (Foster, 2014)

Berdasarkan deskripsi tersebut dapat diterjemahkan bahwa rekayasa perangkat lunak ialah suatu proses di mana sistem perangkat lunak yang diselidiki, direncanakan, dibuat modelnya, dikelola, dan dikembangkan.

Menurut Ivan Marsic, definisi dari rekayasa perangkat lunak merupakan: "*Software engineering is a discipline for solving business problems by designing and developing software-based systems.*" (Marsic, 2012)

Berdasarkan definisi tersebut dapat diterjemahkan bahwa rekayasa perangkat lunak itu adalah disiplin untuk memecahkan permasalahan pekerjaan dengan membuat, merancang dan mengelola pengembangan suatu sistem perangkat lunak.

Menurut Rod Stephens, definisi formal dari rekayasa perangkat lunak adalah : "*An organized, analytical approach to the design, development , use, and maintenance of software*" (Stephens, 2015) .

Berdasarkan definisi tersebut diterjemahkan bahwa pengertian formal dari rekayasa perangkat lunak seperti pendekatan analitis yang terorganisir untuk desain, pengembangan, penggunaan, dan pemeliharaan perangkat lunak.

Perangkat lunak itu sendiri adalah suatu kumpulan prosedur, program dan dokumen yang saling berkesinambungan yg membahas masalah yang ada di dunia nyata dan dikonfigurasi dalam bentuk aplikasi yang dikerjakan komputer. Perangkat lunak ialah seluruh perintah yang digunakan untuk memproses informasi (Hasanah & Untari, 2020)

RPL merupakan disiplin ilmu yg merepresentasikan aspek-aspek produksi perangkat lunak, berasal dari tahap awal spesifikasi sistem hingga pemeliharaan sistem.

Perangkat lunak yang dirancang harus: tepat waktu, tepat dalam masalah biaya / anggaran, meningkatkan suatu kinerja, mengoperasikan prosedur sistem dengan benar. RPL itu lebih terfokus pada pengoperasian dan pengembangan perangkat lunak serta mengirimkan perangkat lunak yang berguna untuk pelanggan. Sehingga sistem perangkat lunak yang baik merupakan sistem yang mampu memenuhi kebutuhan pelanggan atau berorientasi pada pelanggan atau pemakai perangkat lunak, bukan berorientasi pada pembuat atau pengembang (Hasanah & Untari, 2020)

Jadi, berdasarkan definisi RPL secara keseluruhan disimpulkan rekayasa perangkat lunak itu merupakan prosedur dan program yang saling berhubungan untuk dikonfigurasi pada sebuah aplikasi, yang lebih terfokus pada praktik pengembangan perangkat lunak dan perangkat lunak yang baik adalah yang bermanfaat dan memenuhi kebutuhan *user*.

Berdasarkan definisi dari para ahli yang telah dibahas sebelumnya, maka ada beberapa jenis dengan ruang lingkup RPL diantaranya adalah:

1. *Software requirements* berurusan pada spesifikasi kebutuhan serta kualifikasi tentang perangkat lunak.
2. *Software design* meliputi proses tentang arsitektur, komponen, *design* antarmuka dari perangkat lunak.
3. *Software construction* merupakan detail peningkatan perangkat lunak, seperti algoritma, pengkodean, serta pengujian.
4. *Software testing* mencakup pengujian dengan seluruh perilaku perangkat lunak.
5. *Software maintenance* yaitu meliputi usaha-usaha pemeliharaan pada saat pengoperasian perangkat lunak.
6. *Software configuration management* berkaitan tentang perubahan konfigurasi perangkat lunak dalam memenuhi kebutuhan.
7. *Software engineering management* berkaitan dengan pengelolaan, pengukuran, serta *planning* proyek RPL.
8. *Software engineering tools and methods* meliputi analisis teoritis mengenai alat bantu serta metode RPL.

9. *Software engineering process* berkaitan terhadap pengertian, implementasi, pengukuran, pengelolaan, pengembangan serta perbaikan proses RPL.

10. *Software quality* mengutamakan suatu kualitas serta siklus hidup perangkat lunak.

(Hasanah & Untari, 2020)

Rekayasa perangkat lunak ini mengutamakan bagaimana membuat perangkat lunak dengan kriteria-kriteria seperti:

1. Mampu selalu dikelola sehabis perangkat lunak selesai diciptakan beriringan dengan majunya teknologi, lingkungan, dan sistem perangkat lunak mampu dibuat sedemikian rupa agar mudah diubah apabila terjadi perubahan kebutuhan perangkat lunak oleh pemesan. (*Maintainability*)
2. Bisa diandalkan pada proses bisnis dijalankan perubahan yang terjadi, tidak mengakibatkan kerugian yang besar jika terjadi kegagalan pada sistem, dan perangkat lunak wajib aman dari pengguna yang tidak memiliki hak akses. (*Dependability Robust*)
3. Berdaya guna pada aspek sumber daya dan penerapan perangkat lunak tidak boleh menggunakan memory dan processor cycles secara berlebihan serta mampu melakukan proses dalam waktu singkat. (*Efficiency*).
4. Kemampuan yang digunakan sesuai dengan kebutuhan (*Usability*). Berdasarkan kriteria tersebut disimpulkan perangkat lunak yang baik yaitu yang mampu melengkapi kebutuhan pelanggan (*customer*) ataupun *user* (pemakai perangkat lunak), tidak berorientasi terhadap pembuat atau pengembang perangkat lunak.

Berdasarkan beberapa penjelasan diatas, dapat ditarik kesimpulan yaitu rekayasa perangkat lunak ialah tentang proses yang lebih fokus pada maintenance yang berkembang seiring lingkungan dan teknologi, lalu berfokus juga pada efisiensi sumber daya, dapat diandalkan pada proses bisnisnya dan bermanfaat dan memenuhi kebutuhan *user*.(Hasanah & Untari, 2020)

Menurut Rosa dan Shalaluddin dalam buku "Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek", RPL merupakan perancangan dengan memakai konsep, prinsip rekayasa agar menghasilkan perangkat lunak bernilai ekonomis yang

dapat dipercaya dan bekerja secara efisien menggunakan mesin. (Salahuddin & Rosa, 2011)

Maka mampu ditarik kesimpulan sebenarnya rekayasa perangkat lunak ialah suatu prosedur pembuatan dan pengembangan perangkat lunak dimulai dari tahap perancangan hingga tahapan implementasi memakai pendekatan sistematis, disiplin, dapat diukur sehingga menghasilkan suatu perangkat lunak yang handal, ekonomis, dan juga dapat seefisien mungkin penggunaannya.

## 2.2 Metode Penelitian

Dalam pembuatan Aplikasi *Augmented Reality* secara *Markerless* berbasis Android untuk Memperkenalkan Kesenian Wayang Golek berbasis android ini, digunakan pengembangan berorientasi objek yang biasa disebut OOP (*Object Oriented Programming*).

Dan Clark dalam bukunya yang berjudul *Beginning C# Object-Oriented Programming* mendefinisikan pemrograman berorientasi objek atau OOP:

*Object-oriented programming is an approach to software development in which the structure of the software is based on objects interacting with each other to accomplish a task.* (Clark & Sanders, 2011)

Berdasarkan penjelasan tersebut, dapat dijelaskan bahwa *Object Oriented Programming* atau OOP ialah sebuah pendekatan dalam pengembangan PL dimana sebuah perintah atau tugas dieksekusi pada interaksi menggunakan *variable* dan *method*. (Clark & Sanders, 2011). *Object Oriented Programming* atau OOP memiliki beberapa konsep dasar diantaranya: (Mall, 2018)

### 1. Class

*Class* merupakan sekumpulan objek dengan mempunyai atribut serupa. *Class* terdiri atas sekumpulan objek misalnya class mamalia terdiri dari anjing, kucing, dan kuda.

### 2. Objects

Dalam *Object Oriented Programming*, *objects* akan saling berinteraksi untuk menyelesaikan suatu tugas. Objek memiliki data-data dan bisa diakses oleh objek lainnya. Data yang tersimpan disebuah objek disebut dengan *attributes*.

### 3. *Methods*

Data pada OOP memiliki *methods* yang mengizinkan sebuah data untuk mengakses data dari objek lainnya. (Mall, 2018)

### 4. *Abstraction*

*Abstraction* adalah sebuah konsep OOP dimana program akan menyelesaikan aspek-aspek yang penting saja dan mengabaikan aspek-aspek lain yang dan berhubungan.

### 5. *Polymorphism*

*Polymorphism* adalah konsep dimana suatu kelas dapat menggunakan metode, nama, ataupun argumen yang sama seperti kelas lainnya tetapi melakukan proses yang berbeda.

### 6. *Encapsulation*

Dalam OOP sebuah kelas bisa memilih data mana yang akan ditunjukkan pada user atau data mana yang harus disembunyikan guna untuk menghindari perubahan data tidak diinginkan. (Hillar, 2015)

Dikarenakan konsep OOP berfokus pada objek untuk menyelesaikan sebuah tugas, maka terdapat juga hubungan antar objek diantaranya: (Mall, 2018)

#### 1. *Inheritance*

Dalam OOP, sebuah kelas bisa melakukan ekstensi dengan mewariskan atribut pada *subclass*. *Class* yang melakukan ekstensi disebut dengan *superclass*. *Subclass* memiliki atribut serupa dengan *superclass* dengan tambahan atribut baru. (Hillar, 2015: 78)

#### 2. *Agregasi*

Dalam OOP data tidak bisa diakses secara langsung. Data harus diakses melalui object yang berkewajiban pada data tersebut. (Clark & Sanders, 2011)

#### 3. *Dependencies*

#### 4. *Asosiasi*

Pemrograman berorientasi objek ialah sebuah proses penerapan dimana program disusun dari kumpulan objek yang saling berkolaborasi.

### 2.2.1 Karakteristik metodologi berorientasi objek

Metodologi berorientasi objek ini tidak dapat bekerja sendiri, tetapi metodologi ini perlu adanya dukungan dari sebuah konsep, karakter, serta kumpulan objek yang berisi data dan operasi sebagai dasar dari perancangan sistem berorientasi objek.

Berikut merupakan beberapa konsep yang dapat dan harus dipahami dari metodologi pengembangan sistem berorientasi objek yaitu:

#### 1. Kelas (*Class*)

Kelas juga memiliki sebuah atribut, metode, hubungan, dan juga memiliki arti. Kelas merupakan sebuah bentuk yang ada pada kode program dengan memakai metodologi berorientasi objek.

#### 2. Objek (*object*)

Objek dapat menjadi sebuah entitas mampu dalam menampung sebuah informasi dan sebuah kelakuan atau metode yang bisa mempengaruhi status pada objek. Contohnya seperti benda, satuan organisasi, manusia, status, struktur, kejadian dan beberapa hal lainnya yang abstrak.

#### 3. Metode (*method*)

Metode merupakan sebuah kelas yang dapat memiliki 1 atau lebih metode atau operasi agar dapat berfungsi memanipulasi objeknya sendiri. Aktivitas, aksi suatu keadaan, dan kelakuan di dalam dunia nyata merupakan asal dari sebuah metode. Contohnya adalah *Copy*, *Write*, *Read*, *Cut*, dan sebagainya. Metode ada 2 yaitu Metode *get* dan *set*. Fungsi metode *get* adalah agar suatu atribut dapat diakses oleh kelas lain, dan metode *set* mengisi dan menjaga atribut agar tidak mudah di akses oleh kelas lain secara langsung.

#### 4. Atribut (*attribute*)

Atribut sebuah kelas memiliki suatu variabel tertentu atau global. Atribut sebuah objek memiliki beberapa nilai dan beberapa elemen data. Atribut harus bersifat *private* agar terjaganya konsep enkapsulasi. Atribut sebuah objek contohnya adalah nama, jenis, dan lainnya.

#### 5. Abstraksi (*abstraction*)

Sebuah hal yang kompleks terdapat di dunia nyata dapat terjadi sesuatu bentuk model yang sangat sederhana menggunakan cara mengabaikan beberapa aspek yang tidak termasuk dan sesuai permasalahan.

6. Enkapsulasi

Objek yang mempunyai penggabungan atribut dan layanan data agar penerapan disembunyikan sehingga objek-objek lainnya tidak dapat memahami proses kerjanya.

7. Pewarisan (*inheritance*)

Sebuah hal mampu mewariskan separuh atau seluruhnya kepada bagian dari objek lain.

8. Antarmuka (*interface*)

Memiliki kemiripan dengan kelas, tetapi antarmuka ini tidak mempunyai atribut kelas serta mempunyai sebuah metode yang tidak mempunyai isi. Metode tersebut oleh kelas lain mampu mengimplementasikan. Satu dan lebih antarmuka mampu diimplementasikan oleh satu kelas dengan mendeklarasikan suatu metode yang dibutuhkan kelas, dan isinya didefinisikan kepada kode program kelas.

9. *Reusability*

Objek yang telah dibuat dan digunakan pada sebuah masalah dapat digunakan kembali untuk sebuah permasalahan lain. Contohnya adalah ketika meminjam buku di perpustakaan harus memerlukan sebuah kelas Anggota, dan saat ingin meminjam buku lagi, kelas anggota dapat dipergunakan kembali dengan perbedaan sedikit pada aplikasi peminjaman buku.

10. Generalisasi dan Spesialisasi

Sesuatu hal (kelas serta objek) yang umum memiliki hubungan dengan sesuatu hal (kelas dan objek) yang khusus. Contohnya adalah generalisasi (umum) yaitu kendaraan beroda dan spesialisasi (khusus), yaitu mobil, sepeda dan motor.

11. Komunikasi Antar Objek

Komunikasi antara sebuah objek dengan objek lainnya dengan menggunakan pengiriman pesan.

12. Polimorfisme (*polymorphism*)

Sebuah objek yang dapat melakukan penghematan baris program dengan cara membuat persamaan nama.

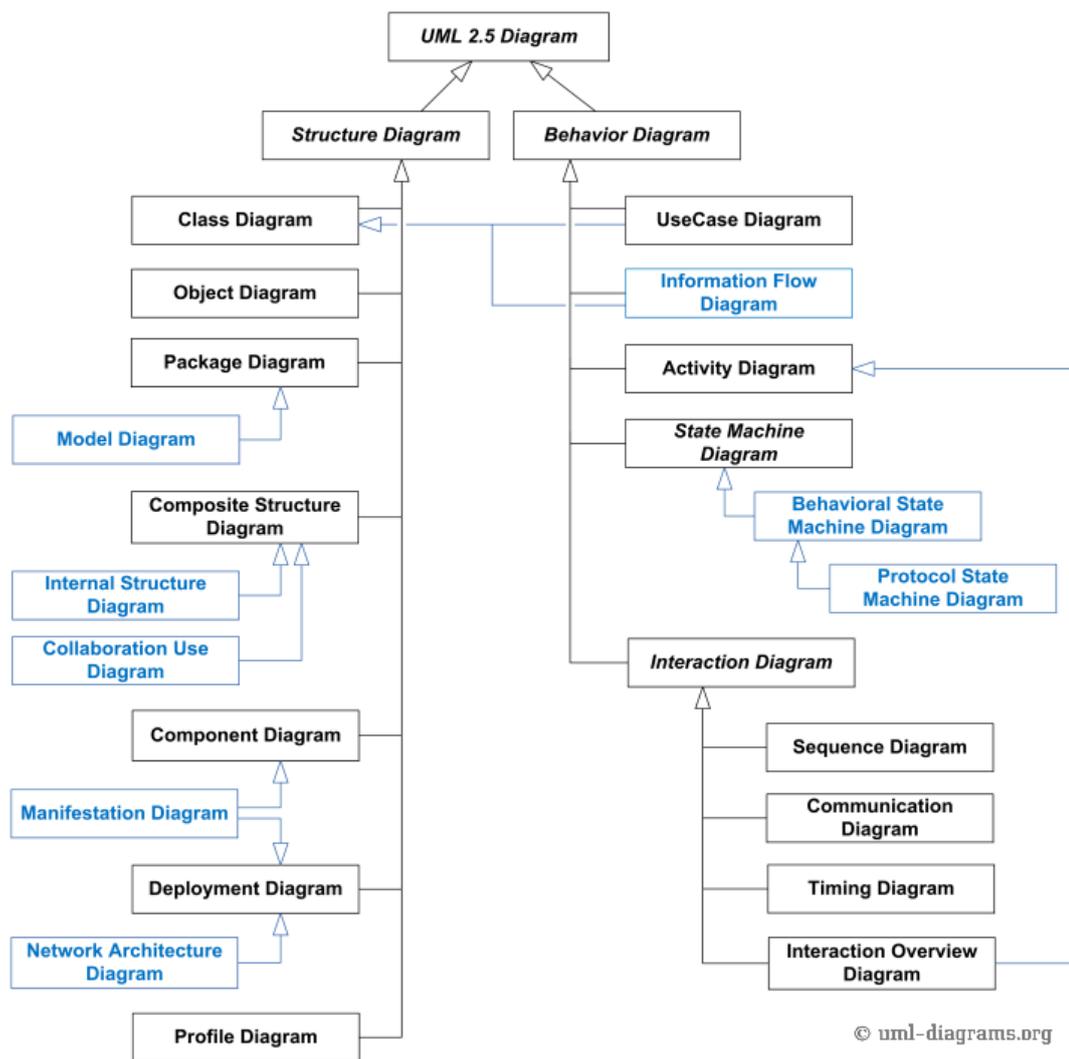
13. *Package*

Melakukan penggabungan atau pengelompokan sejumlah kelas yang sama disimpan didalam *package* lain atau berbeda.

(Salahuddin & Rosa, 2011)

## 2.2.2 Unified modelling language (UML)

*Unified Modeling Language* dikembangkan oleh Grady Booch, Ivar Jacobson, dan James Rumbaugh pada tahun 1995, Grady Booch, Ivar Jacobson, dan James Rumbaugh sebelumnya telah mengembangkan notasi dan metodologi masing-masing. Objektif utama dari *Unified Modeling Language* adalah memberikan kamus yang umum dari istilah-istilah *object-oriented* dan teknik pembuatan diagram untuk pemodelan proyek pengembangan sistem mulai dari tahap analisis hingga implementasi. (Dennis et al., 2015)



Gambar 2. 1 UML Diagram 2.5

(Sumber: <https://www.uml-diagrams.org/uml-25-diagrams.html>)

UML gambar 2.1 adalah versi terbaru, menyediakan lima belas jenis diagram yang secara garis besar bisa di kelompokkan menjadi dua yaitu *structure* diagram dan *behavior* diagram. Keduanya memiliki fokus dan fungsi berbeda. *Structure* diagram biasanya

digunakan untuk mendefinisikan komponen dan hubungan antar komponen. Sedangkan *behavior* diagram digunakan untuk menggambarkan dinamika dari sebuah sistem. *Behavior* diagram berfokus pada interaksi antara entitas yang berbeda. (Cha et al., 2019)

Diagram-diagram UML yang akan digunakan dalam laporan tugas akhir ini adalah:

1. *Use Case Diagram*

ialah suatu sarana dalam melaksanakan pengorganisasian spesifikasi kebutuhan pengguna menggunakan cara yang sederhana untuk diatur serta dapat dipahami da setiap pengguna. (Nugroho, 2010)

2. *Scenario Use Case Diagram*

Setiap *use case* akan dilengkapi dengan skenario. Yang merupakan suatu alur proses *use case* yang telah dibuat berdasarkan sisi aktor serta sistem. Skenario dibuat untuk setiap *use case* mulai dari yang terkecil, misalnya untuk generalisasi oleh karena itu skenario yang telah diciptakan ialah *use case* khusus. Skenario normal adalah skenario bila sistem berjalan normal tanpa ada kesalahan seperti error. Sedangkan skenario alternatif adalah skenario bila sistem tidak berjalan normal, alau mengalami error Skenario normal dan skenario alternatif dapat dapat dibuat lebih dan satu. Alur dan skenario inilah nantinya menjadi dasar pembuatan diagram sekuen

(Salahuddin & Rosa, 2011)

3. *Class Diagram*

Merupakan diagram yang berguna untuk mendeskripsikan hubungan antar class. *Class* diagram memiliki fungsi berbeda dalam tahap peningkatan perangkat lunak. Dalam tahap analisis, *class* diagram berguna untuk menyusun konsep dunia nyata. Pada tahap perancangan dan implementasi, *class* diagram digunakan untuk menggambarkan struktur suatu sistem. (Rumpe, 2016)

4. *Sequence Diagram*

*Sequence* diagram merupakan salah satu yang menjelaskan bagaimana suatu operasi itu dilakukan. Objek-objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut. *Sequence*

Diagram mendeskripsikan bagaimana entitas dalam sistem berinteraksi, termasuk pesan yang digunakan saat interaksi.

#### 5. *Activity Diagram*

Adalah suatu langkah dari tampilan fungsional karena menggambarkan logika proses, atau fungsi, diimplementasikan menggunakan kode. Setiap proses menggambarkan urutan dari tugas dan keputusan yang mengatur kapan dan bagaimana tugas tersebut dilakukan. *Activity diagram* juga dapat digunakan untuk memodelkan implementasi metode yang kompleks. (Yunanto, 2015)

### 2.3 Model Pengembangan

Dalam peningkatan perangkat lunak, terdapat beberapa macam model pengembangan, diantaranya : (Foster, 2014)

#### 1. *Waterfall*

Model ini merupakan model tradisional yang berurutan atau bertahap, pada setiap tahapannya akan dilakukan pengecekan hingga suatu tahapan selesai baru tahapan selanjutnya dapat dieksekusi. Dalam model ini setiap persyaratan akan didokumentasi dengan jelas.

#### 2. *Phased Prototype Model*

Pada model ini tahapan pertama adalah melakukan investigasi dan menentukan sistem yang akan dibuat lalu mengidentifikasi komponen utama yang dibutuhkan. Dan dibuatlah model dan dikembangkan sesuai keinginan pengguna.

#### 3. *Iterative Development Model*

Model ini ialah pengembangan dari model *phased prototype*. Model ini tersusun dari berbagai tahapan (investigasi, analisis, perancangan, pengembangan, pengujian, dan implementasi) yang dilakukan secara berulang dalam waktu yang lebih singkat.

#### 4. *Rapid Prototype Model*

Model peningkatan perangkat lunak dengan menggunakan perencanaan yang minimal tetapi mendukung serta mampu membuat *prototype* lebih cepat. *Prototype* adalah model kerja yang secara fungsional setara dengan produk yang dibuat. Istilah lain yang sering

digunakan juga yaitu RAD (*rapid application development*). Model ini pada umumnya hanya mengeluarkan biaya yang kecil dan tidak memakan waktu yang lama untuk mengembangkannya.

#### 5. *Formal Transformation Model*

Dalam model ini kualitas adalah bagian terpenting pada peningkatan perangkat lunak. Terdapat beberapa tahapan ketat yang harus dilalui untuk menjaga keandalan serta keunggulan perangkat lunak.

#### 6. *Component-Based Model*

Pada model ini perangkat lunak yang dikembangkan berasal dari komponen-komponen yang didapat dari perangkat lunak lainnya yang telah teruji. Dikarenakan perangkat lunak yang ditingkatkan lebih stabil serta diharapkan digunakan di lingkungan yang lebih luas.

#### 7. *Agile Development Model*

Model ini pun mengambil konsep ide dari model *phased prototyping*, *iterative development*, dan *rapid prototyping* yang lebih inti terhadap hasil dari pada proses pencapaiannya. Lebih menekankan pada pengembangan fungsi yang cepat dan bisa segera ditunjukkan hasilnya, serta minim dokumentasi.

Terlepas dari proses model yang digunakan, pengembangan perangkat lunak melalui lima buah tahap yaitu *Software Development Life Cycle* atau disingkat SDLC. Tabel 2.1 adalah tahapan-tahapan dalam SDLC berikut hasil yang dikeluarkan.

(Foster, 2014)

Tabel 2. 1 SDLC Phases and Related Deliverables

Tahapan SDLC	Hasil yang Diharapkan
Penelitian dan Analisis	Kebutuhan awal sistem, Spesifikasi kebutuhan sistem
Desain (Modeling)	Spesifikasi desain
Pembuatan Perangkat Lunak	Sistem perangkat lunak, Dokumentasi produk
Implementasi	
Managemen	Sistem perangkat lunak dan dokumentasi yang sudah diperbaharui.

Dalam penelitian tugas akhir ini penulis akan menggunakan model pengembangan *Phased Prototype Model*.

Menurut Elvis C. Foster Model ini:

*Also referred to as the evolutionary development model. Advantages of phased prototype model are: The user gets a feel of the system before its completion. Improved user participation over the waterfall model. The likelihood of producing an acceptable system is enhanced. The disadvantages of the phased prototype model include: The increased likelihood of a poorly documented system. The system may be poorly integrated. The system will therefore be more difficult to maintain. (Foster, 2014)*

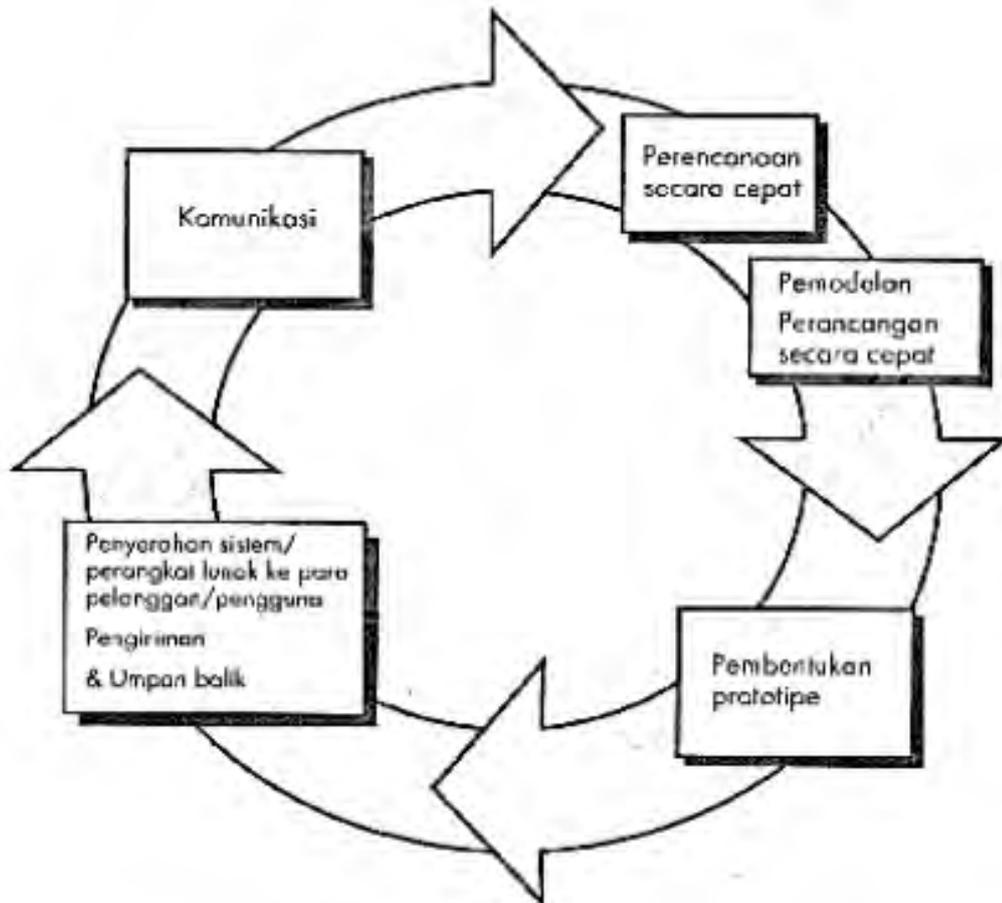
Berdasarkan definisi tersebut diterjemahkan bahwa Model *Prototype* ialah model pengembangan evolusioner dan memiliki: Kegunaan pada model prototipe bertahap adalah Pengguna / *Client* bisa langsung merasakan penggunaan sistem sebelum selesai, kemungkinan dapat menghasilkan sistem yang dapat diterima dan ditingkatkan, *Developer* dapat mengembangkan aplikasi berdasarkan saran atau *feedback* langsung *client*.

Kerugian dari model prototipe bertahap meliputi: Kemungkinan peningkatan sistem yang didokumentasikan dengan buruk, adanya sistem mungkin tidak terintegrasi dengan benar, oleh karena itu sistem menjadi lebih sulit untuk dipelihara.

Menurut Setiyani (Setiyani, 2019) , menjelaskan bahwa pengguna perangkat lunak kadang mendefinisikan kebutuhannya secara umum, dan tidak dapat mendefinisikan kebutuhan secara rinci terkait fitur-fitur atau fungsi-fungsi yang nantinya akan dikembangkan. Dengan melihat kasus ini maka pengembang tidak memiliki kepastian terkait efisiensi algoritma dimanfaatkan pada peningkatan perangkat lunak yang berguna untuk pengguna. Pada pengulasan diatas, pendekatan yang paling baik adalah menggunakan paradigma pembuatan *prototype*. Dalam hal ini paradigma pembuatan prototipe sering membantu tim pengembangan serta para stakeholder untuk lebih memahami kebutuhan perangkat lunak yang akan ditingkatkan, Pembuatan *prototype* dimulai dengan komunikasi pada tim pengembangan dengan pelanggan, kemudian menetapkan sasaran pengembangan secara menyeluruh dan mengenali spesifikasi kebutuhan.

Pembuatan *prototype* direncanakan dan dirancang dengan cepat rancangan kemudian di konstruksi pada pembentukan *prototype*. *Prototype* kemudian diserahkan kepada stakeholder dan mereka melakukan evaluasi terhadap *prototype* tersebut, yang pada akhirnya memberikan umpan balik dalam memperbaiki atau mengembangkan detailnya kebutuhan perangkat lunak. Pengulasan pada saat *prototype* diperbaiki

memenuhi kebutuhan stakeholder, serta pada waktu bersamaan memungkinkan pengembang memahami lebih dalam kebutuhan dari perangkat lunak dikerjakan. Semakin banyak pengembangan, maka dari itu perangkat lunak berevolusi memenuhi kebutuhan pengguna. Paradigma pembuatan *prototype* digambarkan:



Gambar 2. 2 Paradigma Pemodelan Prototype

(Setiyani, 2019)

Menurut Rosa dan Shalahuddin (Salahuddin & Rosa, 2011) maka dapat ditarik kesimpulan model *prototype* cocok digunakan untuk menjabarkan kebutuhan pelanggan secara lebih detail karena pelanggan sering kali kesulitan dalam menyampaikan kebutuhan secara spesifik dengan tidak melihat gambaran dengan jelas. Untuk mengantisipasi bahwa proyek yang sedang berjalan dapat sesuai dengan target waktu dan biaya di awal, maka sebaiknya spesifikasi persyaratan sistem harus disetujui oleh pengembang dengan pelanggan secara tertulis. Dokumen tersebut merupakan ukuran sehingga spesifikasi persyaratan sistem berada di dalam lingkup proyek.

## 2.4 Teori Khusus Topik Tugas Akhir

### 2.4.1 Arsitektur *Augmented Reality*

*Augmented Reality* (AR) adalah variasi dari *Virtual Environment* (VE) atau *Virtual Reality* (VR). AR berguna untuk user dapat menonton dunia nyata dan menambah realitas, menggunakan objek virtual dengan penggabungan dengan dunia nyata. Pencapaiannya, user mampu melihat benda virtual serta nyata yang muncul bersamaan pada tempat yang sama.

AR ialah suatu teknologi yang menjalani kemajuan pesat sekali. Oleh karena itu, kegunaannya yang luas, *augmented reality* dapat dipakai untuk beberapa bidang misalnya pendidikan, pembelajaran, desain, game, dan masih banyak lagi. AR memiliki perbedaan dengan *Virtual Reality* atau VR yaitu cara bagaimana user berinteraksi dengan dunia maya. Dalam VR, pengguna harus menggunakan alat yang memasukkan pengguna ke dalam dunia maya.

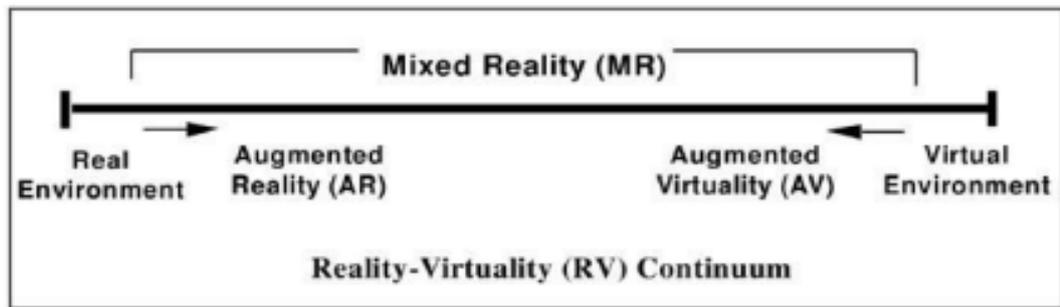
Dalam buku berjudul *Virtual and Augmented Reality for Dummies*, Paul Mealy mendefinisikan *augmented reality* yaitu:

*Augmented reality is a way of viewing the real world (either directly or via a device such as a camera creating a visual of the real world) and "augmenting" that real world visual with computer-generated input such as still graphics, audio, or videos. AR is different from VR in that AR augments (adds to) a real-world or existing scene instead of creating something new from scratch. (Mealy, 2018)*

Craig menganggap AR Sebagai sebuah medium yang digunakan untuk menyampaikan informasi digital (Craig, 2013) dengan memasukkan informasi tersebut pada dunia nyata. (Peddie, 2017)

Menurut Azuma ada tiga karakteristik yang wajib dimiliki oleh *augmented reality* yaitu: (Azuma, 1997)

1. Menggabungkan dunia nyata dan virtual.
2. Berinteraksi dalam waktu nyata.
3. Berbentuk tiga dimensi.



Gambar 2.3 Skema Virtuality Continuum

Sumber (Furht, 2011)

Gambar 2.3, menunjukkan bedanya pada *Augmented Reality* (AR) dengan *Virtual Reality* (VR). Dengan AR, objek virtual terdiri dari teks, objek 3D, gambar, animasi, atau video dapat disatukan pada lingkungan nyata (*real environment*). Sedangkan untuk VR, manusia akan sepenuhnya berinteraksi dengan computer dalam lingkungan maya atau *virtual environment*.

Secara garis besar, terdapat dua langkah utama dalam aplikasi *augmented reality* ialah: (Craig, 2013)

1. Aplikasi harus menentukan kondisi dunia nyata serta dunia maya saat itu.
2. Aplikasi harus menampilkan dunia maya dalam keselarasan dengan dunia nyata dimana partisipan bisa merasakan elemen dunia maya sebagai bagian dari dunianya dan kemudian kembali ke langkah pertama dan dilanjutkan ke langkah berikutnya.

Dalam pengimplementasian *augmented reality*, terdapat tiga materi utama ialah:

1. Sensor

Sensor memiliki fungsi untuk mengetahui posisi dari hardware. Sensor sendiri bisa dibagi menjadi tiga kategori yang memiliki fungsinya masing-masing

- a. Sensor untuk *tracking*

Dalam proses *tracking*, sensor berguna untuk menentukan posisi dari perangkat di dunia nyata. Beberapa contoh sensor untuk *tracking* yaitu kamera, GPS. Kamera sebagai sensor berperan dalam pengambilan gambar dunia nyata serta GPS berguna untuk menentukan lokasi seperti halnya pada *game augmented reality* PokemonGO yang mengandalkan GPS untuk mengetahui lokasi perangkat.

- b. Sensor yang berfungsi untuk mengumpulkan data

- c. Sensor untuk pengumpulan input
2. Prosesor

Prosesor bertugas untuk memproses informasi yang didapat dan penjelasan dapat dilihat. Untuk bisa menampilkan informasi secara *real-time*, sebuah perangkat harus memiliki kekuatan komputasi yang memadai agar bisa memproses informasi-informasi yang ada.

3. Display

Untuk perkembangan *augmented reality* terdapat dua metode yaitu *marker-based* dan *markerless*. Untuk analisis ini, metode yang akan digunakan ialah *metode markerless*.

#### **2.4.2 Markerless AR**

*Augmented Reality* mempunyai 2 jenis ialah dengan menggunakan marker dan tidak menggunakan marker. Marker merupakan suatu alat bantu agar mampu mendeteksi objek virtual yang dimunculkan. Teknologi *Augmented Reality* dapat menggunakan metode *Markerless*, pada metode ini *user* tidak lagi menggunakan suatu marker agar dapat menampilkan *realtime* objek virtual. Saat ini perusahaan Total Immersion dan Qualcomm mengembangkan teknologi *Augmented Reality*, terdapat berbagai macam metode AR pada *markerless Tracking* seperti *face tracking*, *3D object tracking*, *gps-based tracking*, dan *motion tracking*.

Objek virtual diantisipasi dari bergantung pada sebagian sekitar sebagai tujuannya. (Arief et al., 2019)

Metode-metode ini mampu dimanfaatkan untuk:

1. *Face tracking*

Cara ini memanfaatkan suatu perhitungan khusus dimana dari perhitungan ini kamera bisa melihat wajah manusia mulai dari bisa melihat keadaan hidung, mata, mulut, yang kemudian kamera melihat benda-benda di sekelilingnya misalnya lemari, kursi, pohon dan benda lain yang berbeda.

2. *3D object tracking*

Teknik *3D object tracking* adalah metode *markerless augmented reality* mampu melihat berbagai macam objek pada dunia nyata seperti meja, kendaraan, lemari, dan lain-lain.

### 3. *Motion tracking*

Dengan cara ini kamera bisa menangkap pergerakan dalam kenyataan. Teknik ini digunakan untuk pembuatan berbagai film yang menciptakan berbagai gerakan. Fitur untuk membedakan gerakan menyerupai membuat film yang menyimulasi Gerakan.

### 4. *Gps-based tracking*

Metode berikut menggunakan fitur GPS dan kompas yang terdapat pada ponsel. Dengan menggunakan fitur ini, aplikasi AR yang dibuat dapat memulihkan informasi dari GPS dan kompas yang kemudian menampilkan arah secara bertahap. Representasinya juga bisa dalam 3 dimensi (3D). Memakai GPS dan sensor kompas yang kemudian menunjukkan suatu object kedalam bentuk arah yang dibutuhkan.

Dengan adanya metode ini, proses *Augmented Reality* tidak lagi terbatas pada marker saja, namun gambar visual, objek 3D, GPS atau wajah mampu dijadikan sebagai target deteksi. Bedanya metode *marker-based* dengan *marker-less* yaitu terjadi pada teknik *tracking* posisi kamera serta orientasi kamera yang diperhitungkan pada marker yang sudah ditentukan. Maka pada *marker-less* menghitung posisi serta orientasi kamera dengan dunia nyata dengan tidak ada ketentuan tertentu, hanya menggunakan fitur alami seperti edge, corner, garis ataupun model 3D.

#### **2.4.3 *Markerless SLAM (Simultaneous Localization and Mapping)***

Dalam metode *markerless*, perangkat lebih mengandalkan sensor untuk menampilkan sebuah sensor. Untuk penggunaan pada smartphone, umumnya AR akan menggunakan GPS untuk menentukan lokasi dari perangkat, inertial unit (*gyroscope, magnetometer, dan accelerometer*), kamera, dan bahkan sensor tambahan. (Arnaldi et al., 2018)

Untuk metode *markerless* sendiri bisa di bagi menjadi dua metode yaitu dengan GPS dan *SLAM*. Metode *markerless* dengan GPS sebelumnya sudah dipakai pada banyak aplikasi atau permainan contohnya seperti PokemonGO dimana pemain harus berjalan ke tempat tertentu untuk bisa menangkap pokemon, bertarung, bahkan menyimpan pokemon mereka. Bahkan, setiap negeri memiliki pokedex langka yang berbeda-beda. Untuk bisa

mengetahui posisi dari pengguna, perangkat harus terlebih dahulu mengakses GPS agar bisa mengatur posisi dari perangkat lalu informasi akan terus diperbaharui sehingga posisi pemain bisa berganti-ganti sesuai tempat mereka saat itu.

Secara garis besar SLAM atau *simultaneous localization and mapping*, bekerja dengan dasar membuat peta 3D berdasarkan gambar dari kamera dan menempatkan objek di peta tersebut. Dikutip dari buku *Virtual Reality and Augmented Reality Myths and Realities* dijelaskan cara kerja dari metode SLAM yaitu:

*The system is based, first of all, on recognizing the environment a 3D map of space is image mixed with the information from the inertial unit make it possible to find the position and orientation of the headset in this space.*  
(Peddie, 2017)

Menggambarkan suatu proses yang bersangkutan dalam SLAM. Pertama-tama, modul *acquisition* mengambil gambar dengan kamera kemudian modul *tracking* akan menghitung lokasi dan orientasi untuk virtual *overlay*. Gambar asli dan komponen virtual akan digabungkan oleh modul rendering dengan pose yang telah dihitung dan kemudian merender gambar augmented pada layar.

Penggunaan metode SLAM bermanfaat sekali pada aplikasi-aplikasi yang mengharuskan menyediakan banyak informasi gambar sehingga pengguna tidak perlu repot membawa atau mencari marker terlebih dahulu. Tetapi SLAM mempunyai banyak keterbatasan atau kekurangan yaitu *requirement* yang tinggi baik untuk pengguna maupun developer sendiri.

#### **2.4.4 Wayang Golek**

Wayang Golek merupakan salah satu kesenian tradisional yang berasal dari Provinsi Jawa Barat. Seperti dikutip dari [www.pdwi.org](http://www.pdwi.org), Wayang Golek adalah salah satu bentuk seni pertunjukan yang tumbuh dan berkembang di daerah Jawa Barat, yang terbuat dari kayu. Kesenian Wayang Golek ini sangat populer di Jawa Barat khususnya di wilayah tanah pasundan. Wayang Golek mempunyai tiga dimensi layaknya boneka. Wayang Golek terlihat lebih real daripada Wayang Kulit dan Wayang Klithik. Wayang merupakan suatu seni pertunjukan yang edipeni-adiluhung, yang artinya merupakan seni dengan terdapat nilai-nilai keindahan serta dengan adanya ajaran moral spiritual. (Cahaya, 2019)

Menurut beberapa sumber, sejarah Wayang Golek di mulai pada abad 17. Awalnya, kesenian Wayang Golek lahir dari wilayah pesisir utara pulau Jawa. Yang selalu digunakan oleh Sunan Kudus dalam penyebaran agama Islam di lingkungan masyarakat. Pada zaman dahulu masa itu, pertunjukan Wayang Golek masih menggunakan bahasa Jawa untuk dialog-dialognya, yang mulai berkembang di Jawa Barat pada waktu ekspansi kesultanan Mataram. Dan pada masa perkembangannya Wayang Golek mulai menggunakan bahasa Sunda untuk dialognya. Lalu selain sebagai media penyebaran agama, Wayang Golek berguna sebagai pelengkap acara syukuran ataupun ruwatan. Wayang Golek semakin maju dan berkembang budayanya dengan mulai menggunakan iringan sinden pada 1920an. Hingga saat ini Wayang Golek terus menerus berkembang sebagai hiburan masyarakat terutama berada di tanah Sunda.

Dalam pertunjukan Wayang Golek lakon serta cerita di mainkan oleh seorang dalang dengan bahasa Sunda pada dialog yang di bawakan. Pakem dan jalan cerita Wayang Golek hampir sama dengan wayang kulit, seperti pada cerita Ramayana dan Mahabharata. Tetapi perbedaannya ialah pada tokoh punakawan, penamaan dan bentuk dari punakawan memiliki versi tersendiri adalah dalam versi Sunda. Pada tokoh-tokoh wayang mempunyai sifat serta karakter sendiri-sendiri, secara umum dikelompokkan dalam 2 kelompok besar, yaitu tokoh yang berkarakter baik, lalu yang berkarakter buruk. Pada pertunjukan wayang, kedua kelompok tokoh yang berbeda karakter ini senantiasa dihadapkan sebagai lawan satu sama lain.

Dalam dunia pedalangan ada dua sumber cerita pewayangan, yaitu ada yang perang Mahabharata, yang menghadapkan tokoh-tokoh dari Negara Amarta dengan tokoh-tokoh Negara Astina yang dihadapkan sebagai dua kelompok yang berlawanan. Tokoh-tokoh Negara Amarta dengan tokoh Pandawanya dikelompokkan dalam golongan protagonis, sedangkan tokoh-tokoh Negara Astina dengan tokoh-tokoh Kurawanya digolongkan sebagai kelompok antagonis. Tokoh yang baik (protagonis) selalu tampil dari sisi sebelah kanan, sedangkan lawannya (antagonis), sebelah kiri.

Menurut seorang informan, ada kurang lebih 120 karakter tokoh wayang, yang terdiri dari karakter protagonis serta antagonis. Secara visual, karakter masing-masing

tokoh pada pewayangan digambarkan dengan perbedaan bentuk serta warna kulit muka. Karakteristik para tokoh digolongkan menjadi 4 golongan utama, diantaranya: Satria, Ponggawa, Buta, dan Panakawan. (Dhari, 2019).

Jadi dapat disimpulkan bahwa wayang golek ialah hasil cipta, rasa, dan karsa yang adalah kesenian Wayang Golek ini populer sekali di Jawa Barat khususnya di wilayah Tanah Pasundan.

Wayang yang dibuat dalam aplikasi ini terdiri dari 3 yaitu: (Cepot, Semar, Hanoman). Berikut penjelasan tentang ketiga wayang golek tersebut:

#### 1. Wayang Golek Cepot

Wayang Golek Cepot yang mempunyai nama Astrajingga dengan dua suku kata, yaitu sastra yang berarti tulisan, dan jingga yang berarti merah sebagai lambang dari kelakuan yang buruk. Namun dibalik warna merahnya, tokoh si Cepot memiliki makna, nilai, dan gambaran yang bersifat berani. Cepot merupakan anak sulung dari tiga bersaudara dari pasangan Semar Badranaya dan Sutiragen. Nama cepot saya Sastra Jingga, karakternya yang humoris, terkadang sedikit mengganggu, selalu mampu membuat suasana cerita yang menegangkan menjadi lebih cair dan penuh tawa. Dengan humornya memberi saran dan kritik. Dan kini telah menjadi 'ikon' dalam pertunjukan wayang. Cepot memiliki wajah merah dan gigi bawah besar yang menonjol. Warna wajah merah dimaknai oleh kitab pewayangan sebagai representasi sifat buruk.

#### 2. Wayang Golek Semar

Wayang Golek Semar ialah tokoh penasihat dan pegasuh bagi para pandawa. Semar itu sebagai perwujudan Batara Ismaya yang merupakan kakak kandung dari Batara guru, raja dari para dewa. Dalam kisah perwayangan terutama Wayang Golek, tokoh semar bisa dibilang sebagai pemecah suasana pada saat-saat yang menegangkan. Semar juga mempunyai Keris sebagai alat untuk membantunya di saat genting, keris ini disimpan di dalam endong (semacam tas) yang ada di belakang punggungnya sehingga bisa ia bawa kemanapun. Semar digambarkan mempunyai tubuh pendek,

rambut pendek, wajah putih, bokong besar dan perut buncit. Semar merupakan perpaduan rakyat kecil sekaligus dewa dari kahyangan.

### 3. Wayang Golek Hanoman

Wayang Golek Hanoman atau nama lengkapnya Anoman Perbanca Suta yaitu kera berambut putih, anak Batara Bayu dan Anjani. Hanoman pernah menjadi senapati kerajaan Mahespati, melayani Batara Rama dalam cerita Ramayana. Hanoman diberitahu untuk memiliki umur yang sangat panjang, hanoman bertanggung jawab untuk menjaga jiwa Rahwana di pangkuannya. Dan juga Hanoman dikenal memiliki kekuatan magis, kekuatan yang luar biasa dan sakti mandraguna serta segala macam mantra agar hanoman kebal dari serangan musuh dan bisa mengangkat gunung. Hanoman adalah tokoh protagonis dalam cerita wayang Ramayana. Dapat berbicara dengan penuh sopan santun layaknya manusia. Dan merupakan tokoh penting yang gagah berani dan menebarkan kebaikan dalam seni wayang kisah Ramayana.

## 2.5 Perangkat Lunak untuk Analisis dan Perancangan Berorientasi Objek

Adapun perangkat lunak dapat digunakan dalam pengembangan software adalah Unity 3D, Vuforia, Visual Studio, Blender, dan Sistem Operasi Android.

### 2.5.1 Unity 3D

Unity 3d merupakan salah satu software yang bagus digunakan untuk mengembangkan game 3D , *integrated development environment (IDE)* dan selain itu juga merupakan software yang interaktif atau dapat juga digunakan untuk membuat animasi 3D. Unity merupakan suatu software dalam pembuatan video game lebih seringnya disebut juga game engine. (<http://unity3d.com/>) Pembuatan aplikasi yang dibuat dalam Unity 3D yang semua objek di export dalam software ini. Penulisan scrip dilakukan dalam editor yang terpisah didalam Unity 3D yang disebut juga *MonoDevelop*. Unity ialah suatu tool terintegrasi dalam pembuatan game, arsitektur bangunan ataupun simulasi. Unity juga dapat digunakan untuk games PC serta games Online. Untuk games Online membutuhkan plugin yaitu *Unity Web Player*, terjadi sama pada *Flash Player* dengan Browser. Kekuatan terbesar Unity ialah kemampuan sejumlah besar platform yang dibuat dengan mudah.

Unity mampu dibuat untuk membuat game pada Windows, OS X, iOS, Android, Web Plugin, Flash, Xbox 360, PlayStation 3, dan Wii U. yang merupakan peluang yang besar dalam mengembangkan penggunaan Unity. (Cushman & Habbak, 2013)

Bahasa pemrograman yang dipakai bermacam-macam, mulai dari Javascript, C#, dan Boo. Pada proyek game yang sama, penggabungan script dengan bahasa yang berbeda diperbolehkan, tetapi tetap dianjurkan agar memakai satu bahasa saja di seluruh proyek agar menghindari konflik serta memudahkan dalam membaca dan memahami.

Pada unity, tidak dapat melakukan desain dan modelling, karena unity bukan tool untuk mendesain. Jadi untuk mendesain, harus memerlukan 3D editor lainnya contohnya seperti 3dsmax atau Blender. Beberapa yang bisa dilakukan di unity, adanya *fitur audio reverb zone, particle effect, Sky Box*, dan lainnya.

1. Memakai 3 bahasa pemrograman yaitu: JavaScript, C#, dan Boo.
2. *Flexible and EasyMoving, rotating, dan scaling objects* hanya dibutuhkan sebaris kode. Sama halnya dengan *Duplicating, removing, serta changing properties*.
3. Multi Platform Game mampu disebar di PC, Mac, Wii, iPhone, iPad serta browser, android.
4. *Visual Properties Variables* dapat diartikan menggunakan *scripts* yang ditampilkan pada *Editor*. Dapat digeser-geser, *drag and drop*, bisa memilih warna atau *color picker*.
5. Berbasis .NET → Penjalanan program dilakukan menggunakan Open Source .NET platform, Mono.

Unity ialah *game engine cross-platform* yang dibentuk oleh *Unity Technologies*. *Game engine* yang dibangun oleh IDE dengan kemampuan membuat di berbagai platform. Lebih dari satu juta pengembang, sampai saat ini membuat game terkenal menggunakan Unity. Hal ini dibentuk agar memudahkan *user* serta produktivitas yang tinggi. Serta dengan cara belajar yang relatif mudah serta dengan aplikasi versi gratis, membuat beberapa sekolah terdorong dalam mengajarkan Unity sebagai pengantar dalam mengembangkan game. (Cushman & Habbak, 2013)

### 2.5.2 Vuforia

Menurut (Fernando, 2013) Vuforia merupakan software yang berguna pada *augmented reality* yang dikembangkan oleh Qualcomm, menggunakan sumber yang konsisten mengenai computer vision berfokus pada image recognition. Yang berguna membantu para developer dalam pembuatan aplikasi-aplikasi *Augmented Reality* (AR) di mobile phones (iOS, Android). Vuforia memakai teknologi *computer vision* dalam mengenali dan melacak *marker* atau *image* target dan objek 3D sederhana, seperti kotak, secara *real-time*. (Grubert & Grasset, 2013)

Vuforia adalah *Augmented Reality Software Development Kit* (SDK) untuk perangkat mobile yang memungkinkan pembuatan aplikasi AR. SDK Vuforia tersedia untuk digabungkan dengan unity yaitu Vuforia AR Extension for Unity. Vuforia memiliki banyak fitur serta kemampuan membantu pengembang dalam merealisasikan pemikiran mereka yang tidak terbatas secara teknis. Target vuforia ialah obyek dunia nyata yang mampu dideteksi oleh kamera, serta dapat memunculkan obyek virtual. Jenis-jenis target pada vuforia yaitu:

1. *Image targets*, seperti: foto, papan permainan, halaman majalah, sampul buku, kemasan produk, poster, kartu ucapan.
2. *Frame markers*, merupakan tipe frame gambar 2D dengan pattern khusus untuk digunakan sebagai permainan.
3. *Object targets*, adalah obyek 3D fisik yang menjadi markernya. Contohnya: buku, mainan, dan lain sebagainya.
4. *Multi-target*, contohnya kemasan produk mulai dari produk yang berbentuk kotak ataupun persegi. Yang dapat menampilkan gambar sederhana *Augmented 3D*.

*Vuforia object scanner* ialah aplikasi android yang dipakai dalam memindai obyek 3D. *Object scanner* mengeluarkan hasil data obyek file dengan isi data sumber yang diperlukan dalam menentukan obyek target di target manager. *Vuforia object scanner* mampu menghasilkan, menguji serta mengedit file data obyek yang memberikan visualisasi fitur obyek dari target obyek 3D. Vuforia SDK membutuhkan komponen-komponen penting untuk dapat bekerja dengan baik. Komponen-komponennya adalah:

#### 4. Kamera

Kamera diperlukan dalam memastikan jika setiap *frame* ditangkap serta dilanjutkan dengan efisien pada tracker. Developer hanya memberitahu kamera kapan mulai menangkap serta berhenti.

#### 2. *Image Converter*

Mengkonversi format kamera (misalnya YUV21) dalam format yang dapat dideteksi oleh OpenGL (misalnya RGB567) serta untuk *tracking* (misalnya *luminance*).

#### 3. *Tracker*

Mengandung algoritma *computer vision* mampu mendeteksi serta melacak objek dunia nyata di video kamera. Kamera algoritma yang berbeda bertugas dalam mendeteksi *trackable* baru, serta mengevaluasi *virtual button*. Hasilnya akan disimpan pada *state object* yang dipakai pada *video background renderer* serta dapat diakses dari *application code*.

#### 4. *Video Background Renderer*

Merender gambar dari kamera yang tersimpan pada *state object*. Performa dari video background *renderer* bergantung sekali terhadap *device* yang dipakai.

#### 5. *Application Code*

Menginstalasi semua komponen diatas serta melaksanakan tiga tahapan penting pada *application code* yaitu :

- a. *Query state object* terhadap target baru terdeteksi atau *Maker*.
- b. *Update* logika aplikasi masing-masing input baru diinputkan.
- c. *Render* grafis yang ditambahkan (*augmented*).

#### 6. *Target Resources*

Dibuat menggunakan *online Target Management System*. *Assets* diunduh dengan adanya konfigurasi xml – *config.xml* – yang mengharuskan developer untuk mengkonfirmasi beberapa fitur dan *trackable* serta *binary file* dengan berisi database *trackable*.

AR Vuforia mewariskan cara berinteraksi dengan menggunakan kamera mobile phones yang dipakai pada perangkat masukan, sebagai mata elektronik dalam mengenali

penanda tertentu, sehingga di layar bisa ditampilkan perpaduan antara dunia nyata dengan dunia yang digambarkan oleh aplikasi. Berikut adalah kelebihan vuforia:

1. Karena dengan menggunakan teknologi AR, kertas bergambar biasa jika diproyeksikan pada kamera gadget yang mempunyai teknologi Vuforia di dalamnya, maka layar gadget yang awalnya latar kertas kosong berubah menjadi adanya penambahan tampilan grafis 3D yang sensasional.
2. Bebas dalam memilih background game favorit. Adanya game yang dimainkan menggunakan teknologi AR. Karenanya dengan adanya teknologi Vuforia, oleh karena itu akan memudahkan dalam memainkan game dengan latar belakang permainan yang berbeda.
3. Berinteraksi dengan bebas. Adanya teknologi Vuforia, maka mampu berinteraksi secara luas dibandingkan permainan yang hanya menggunakan teknologi 2D atau 3D, karenanya game menggunakan AR akan membuat lupa dalam batas antara dunia nyata dengan dunia teknologi.
4. *Fun*. Yang mempunyai pengertian serta makna yang luas, dengan tujuan yang tetap sama yaitu untuk membahagiakan orang yang memainkannya. Vuforia ialah suatu teknologi *fun*, yang membuat pemain game dapat merasakan kebahagiaan secara luas.
5. *Vallue Added*. Adanya suatu teknologi Vuforia dari Qualcomm, semua developer AR mampu meningkatkan kemampuan untuk menciptakan banyak teknologi dengan mempunyai *vallue added* yang tinggi.

### **2.5.3 Android Studio**

*Android Studio* adalah digunakan untuk mengembangkan aplikasi *Android*. *Android Studio* mempunyai beberapa keunggulan yang membuatnya populer dan banyak penggunanya. Dibawah ini terdapat keunggulan dari *Android Studio*:

1. *Interface* yang mudah dalam penggunaannya
2. Terbuka untuk dikembangkan
3. Selalu dilakukan *upgrade*

4. Banyak aplikasi dan *game* gratis

Berikut kekurangan *Android Studio*:

1. Beberapa dari ponsel android tidak mendapatkan pembaruan
2. Banyaknya merk dan *type* pada *android*
3. *Lag* dan Lemot

Berikut antarmuka *Android Studio*:

1. *Toolbar*

Menunjukkan untuk melakukan atau menjalankan aktivitas dengan cepat. Di jendela ini Anda dapat menangani *SDK Manager*, *AVD Manager*, menjalankan hasil program, *build program*, hingga *debuging*. dibuat sedemikian rupa dengan simbol-simbol yang mudah dikenali sehingga memudahkan para engineer untuk melakukan aktivitas.

2. *Navigation Bar*

Tampilan yang bermanfaat dalam menyelidiki semua aktivitas. Disini dapat dilakukakan tindakan seperti membuat proyek baru, mengedit proyek atau membuka proyek yang sudah jadi. Sebagian kapasitas pada rute dapat diakses padapada *Toolbar* di nomor 1.

3. *Windows Editor*

Dalam tampilan ini pengembang bisa menyesuaikan atau membuat kode. Semua kode dalam desain registri tugas dapat diubah di jendela ini.

4. *Windows Toolbar*

Dapat memperkecil dan memperbesar jendela. Dapat juga untuk menentukan jendela yang akan di tampilkan, serta jendela yang ditutupi.

5. *Windows Tool*

Dapat dilakukan pengaturan manajemen proyek, pencarian dan *version control*. Dapat juga memperbesar serta memperkecil jendela dan juga mampu menentukan jendela yang ingin di tampilkan.

6. *Status Bar*

Memperlihatkan status *project*, proyek dapat berjalan normal, ada kesalahan atau ada pemberitahuan.

#### 2.5.4 Blender

Menurut situs [www.blender.org](http://www.blender.org), blender adalah aplikasi *open source* dan gratis yang mampu dimanfaatkan dalam pembuatan model 3D. Aplikasi blender mempunyai beberapa kelebihan adalah:

1. *Open source* dan *free*

Blender ialah aplikasi *open source* dan *free*, dimana kita bisa dengan bebas memodifikasi dan menggunakan *source code* untuk tujuan apapun, termasuk secara pribadi atau komersial. Kebebasan ini ditentukan oleh *General Public License Blender*.

2. *Multi platform*

Blender dapat dijalankan diberbagai sistem operasi, contohnya Linux, Windows, dan Macintosh.

3. Lengkap

Blender juga memiliki fitur yang lengkap, diantaranya fitur pengeditan video (*video editing*), pembuatan game (*game engine*), animasi, simulasi, *compositing* serta *motion tracking*.

(Artra, 2012)

Jadi blender mempunyai suatu *video compositor* serta *intergrated game engine*. Karya yang dihasilkan dapat dipublikasikan baik free ataupun untuk dikomersilkan. Blender ialah sebuah program modeling 3D serta Animation yang memiliki kelebihan tersendiri daripada program-program modeling 3D lainnya, yaitu kelebihanannya mampu menciptakan game dengan tidak menggunakan program tambahan lainnya, karena Blender sudah memiliki Game Engine sendiri dengan menggunakan bahasa pemograman Python. Pembuatan objek bangunan, robot, penerapan texture, dan animasi yang dihasilkan dari penggunaan Blender.

#### 2.5.5 Android

Android ialah suatu sistem operasi dengan didistribusikan melalui open source dengan Google, pengelolaan *operating system* ini dapat dilakukan oleh berbagai pihak dengan tidak membutuhkan lisensi khusus. (Lee, 2012) Sistem operasi yang diperuntukan

bagi smartphone ini berbasis sistem operasi Linux. API (*Application Programming Interface*) merupakan sejumlah perintah, fungsi, serta protokol yang dipakai oleh programmer dalam pembangunan perangkat lunak pada sistem operasi tertentu, API mengharuskan programmer dalam memanfaatkan fungsi standar untuk berinteraksi dengan sistem operasi. (Lee, 2012)

Android merupakan sistem operasi mobile platform berdasarkan Linux kernel versi 2.6 serta tersedia tidak terbatas pada penggunaan commercial dan non-commercial yang bersifat open source. Beberapa kemudahan dalam menggunakan platform pada android adalah:

1. Android merupakan *open platform*, diartikan android membebaskan dalam mengakses atau apa yang mampu dilakukan pada android.
2. Android ialah *mobile platform* yang sangat mudah berkembang dengan pesat, sangat banyak orang mengunduh serta memainkan game.

Android ialah sebuah sistem operasi open source yang mampu dikembangkan banyak orang dalam membuat aplikasinya sendiri. Setiap munculnya versi android terbaru disertakan dengan nama makanan, contohnya: Cupcake, Gingerbread, Donat, Ice Cream Sandwich, Jelly Bean, Kitkat, Lollipop, Marshmallow, dan yang terakhir rilis pada tanggal 22 Agustus 2016 adalah Nougat (Android Versi 7).

Android Inc. diciptakan pertama kali di Palo Alto, California oleh Rich Miner, Nick Sears, Chris White, dan Andy Rubin. Android Operating System merupakan *operating system* berbasis Linux yang bersifat *open source*. (Callaham, 2018) Dalam mengembangkan aplikasi, Android menggunakan bahasa pemrograman Java sebagai bahasa utamanya namun Android masih tetap mendukung beberapa bahasa pemrograman lainnya, seperti C#, Kotlin, dan Python. Meier dan Ian menjelaskan pada buku *Professional Android®, Fourth Edition*:

*In Android, system, bundled, and all third-party applications are written with the same APIS and executed on the same run time These APIS feature hardware access, video recording, location-based services, support for background services, maps, notifications, sensors, relational databases, inter application communication, Bluetooth, NFC, and 2D and 3D graphics.* (Meier & Lake, 2018)

Menurut kutipan yang diambil di atas, API dalam Android memiliki fungsi penting dalam berjalannya sebuah aplikasi yaitu agar dapat memberikan akses hardware, perekam video. Service berbasis lokasi, dukungan untuk *background services*, peta, notifikasi, sensor, *database*, *inter application communication*, *Bluetooth*, NFC, serta gambar dua dimensi serta tiga dimensi.

### 2.5.6 Star UML

Star UML adalah proyek *open source* untuk mengembangkan platform *Unified Modeling Language* (UML) atau *Model DrivenArchitecture* (MDA) yang cepat, fleksibel, mampu diperluas, dan terdapat banyak fitur. StarUML dikembangkan menggunakan Bahasa Pemrograman Delphi. StarUML merupakan proyek yang *multi-lingual* dan tidak bergantung pada bahasa pemrograman yang spesifik, sehingga bahasa pemrograman lainnya mampu dipakai dalam mengembangkan StarUML, seperti C/C++, Java, Visual Basic, Delphi, Jscript, VBScript, C#, VB.NET, dan lainnya. (Iswari, 2015)

StarUML dilengkapi dengan 13 jenis diagram yang berbeda, mendukung notasi UML dengan dukungan sketsa UML. Star UML juga dapat mengoptimalkan produktivitas dan kualitas dari suatu proyek software. Peranan Star UML adalah sangat penting yaitu sebagai pembuatan gambar-gambar dan skema untuk memenuhi analisis kebutuhan yang akan diciptakan. Seperti didalamnya membuat *usecase* diagram, *activity* diagram, serta *class* diagram, maupun *sequence* diagram.

Jadi mampu ditarik kesimpulan *Unified Modeling Language* (UML) ialah bahasa spesifikasi standar dalam mendokumentasikan, menspesifikasikan, serta membangun sistem perangkat lunak, dan juga adalah suatu himpunan struktur maupun teknik yang berguna dalam pemodelan desain program berorientasi objek (OOP) dan aplikasinya. Penggunaan UML berguna juga dalam membuat model pada semua jenis aplikasi piranti lunak, yang aplikasinya mampu dijalankan pada piranti keras, sistem operasi serta jaringan apapun, dan menggunakan bahasa pemrograman apapun.

## BAB III

### ANALISIS DAN PERANCANGAN PERANGKAT LUNAK

#### 3.1 Gambaran Umum Perangkat Lunak

Aplikasi yang akan dibuat merupakan aplikasi *augmented reality* yang dapat digunakan sebagai media pengenalan dan pembelajaran kesenian wayang golek Jawa Barat, yang akan menampilkan objek-objek 3D, animasi dan dapat mengeluarkan suara. Aplikasi ini akan menggunakan metode SLAM yang berarti *user* atau pengguna dapat mengakses dan menggunakan aplikasi dimana saja dan kapan saja. *User* bisa menggunakan aplikasi untuk memvisualisasikan dalam pengenalan wayang golek.

Perangkat lunak ini akan dipasang pada perangkat *mobile* berbasis Android dan menggunakan layar sentuh. Perangkat lunak ini menggunakan kamera perangkat *mobile* itu sendiri, dan tampilan untuk media pengenalan dan pembelajaran ini memakai layar portrait.

Untuk bisa menggunakan aplikasi, *user* harus terlebih dahulu menangkap dasar bidang datar dengan kamera. Dalam aplikasi terdapat beberapa menu, yaitu:

##### 1. Start AR Wayang Golek

*User* bisa melihat memvisualisasikan wayang golek Jawa Barat untuk pengenalan dan pembelajaran, dan dapat memilih wayang golek mana yang ingin ditampilkan dan dilihat, dapat diperbesar dan diperkecil serta dirotasi. Di menu ini terdapat berbagai wayang golek berbentuk 3D yg dapat ditampilkan dan dapat mengeluarkan suara. Serta dalam menu ini juga dapat melihat keterangan-keterangan atau penjelasan mengenai wayang yang dipilih.

##### 2. Help

Melalui menu ini, *user* dapat melihat instruksi atau cara menggunakan aplikasi ini untuk dapat menampilkan atau memvisualisasikan wayang golek tersebut agar mempermudah *user* dalam menggunakan aplikasi ini.

##### 3. Quit

Menu *quit* ini, dipakai untuk *user* keluar dari aplikasi *Augmented Reality Markerless* Wayang Golek Jawa Barat ini.

### 3.2 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak adalah komponen-komponen apa saja yang akan dibutuhkan sistem dalam membangun sebuah perangkat lunak sampai yang pada akhirnya diimplementasikan, oleh karena itu spesifikasi kebutuhan perangkat lunak dibagi menjadi dua bagian, yaitu : kebutuhan fungsional dan kebutuhan non-fungsional.

#### 3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah deskripsi atas apa yang bisa *user* lakukan dalam aplikasi. Adapun kebutuhan fungsional dalam aplikasi adalah sebagai berikut:

1. Aplikasi dapat menampilkan peragaan wayang golek berupa objek 3D dengan posisi yang diinginkan *user*.
2. Wayang Golek yang dipilih dapat mengeluarkan suara.
3. Membuat *user* atau pengguna dapat memakai aplikasi untuk media pembelajaran dan pengenalan, dan terdapat penjelasan-penjelasan di setiap wayang goleknya.

#### 3.2.2 Kebutuhan non-Fungsional

Kebutuhan non-fungsional merupakan perangkat lunak maupun perangkat keras yang dibutuhkan untuk menggunakan aplikasi:

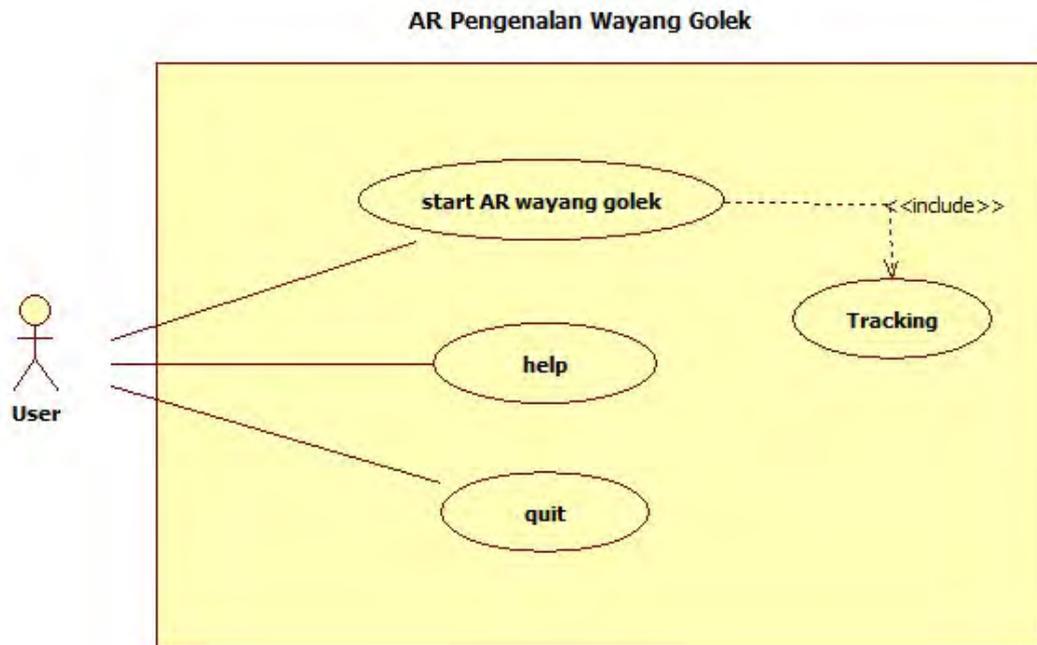
1. Perangkat Keras
  - a. *Smartphone* yang cocok dengan Google ARCore serta fungsi AR.
  - b. RAM minimal 3 GB.
  - c. Kamera 5 MP(Mega Pixel)
2. Perangkat Lunak
  - a. Android sebagai sistem operasi : Android *version* 5.1

#### Diagram-Diagram Perancangan Perangkat Lunak

Diagram-diagram perancangan perangkat lunak memiliki hubungan yang kuat dengan peneliti dalam perancangan sebuah software yang berorientasi objek dan *user* sebagai pemain. Dalam pembuatan desain UML dibagi menjadi 4 diagram yaitu *Use Case* Diagram, *Activity* Diagram, dan *Class* Diagram.

### 3.3 Use Case Diagram

Use case diagram digunakan untuk menggambarkan kebutuhan fungsional aplikasi yang dipresentasikan dengan use case. Gambar 3.1 adalah use case yang mendefinikan kebutuhan fungsional dari aplikasi yang akan dikembangkan:



Gambar 3. 1 Use Case Diagram

Penjelasan untuk use case diagram pada gambar 3.1 adalah sebagai berikut:

#### 1. Tracking

Tracking merupakan proses dimana sistem akan memindai sekelilingnya dengan kamera. Beberapa hal yang akan dipindai adalah sudut, pola.

#### 2. Start AR Wayang Golek

Dapat menampilkan Wayang Golek AR dengan adanya *button* "Start Tracking" dan *button* "Stop Tracking" untuk menampilkan/memunculkan wayang dan memberhentikan tracking wayang. Dan juga ada *button* "Menu wayang" yang berisi wayang apa saja yang ada di dalam aplikasi yang nantinya diklik / dipilih user dan user dapat melihat wayang golek yang dipilih dan yang ingin ditampilkan, dapat di perbesar dengan *button* "Zoom In" dan diperkecil dengan *button* "Zoom Out" serta dirotasi dengan *button* "Rotate". Di menu ini terdapat berbagai wayang golek berbentuk 3D yg dapat di tampilkan dan terdapat *button* "Voice" yang digunakan agar wayang dapat

mengeluarkan suara. Serta dalam menu ini juga dapat melihat keterangan-keterangan atau penjelasan mengenai wayang yang dipilih dengan memilih *button* “*Profile*”.

3. Help

*Use Case help* merupakan *use case* dimana *user* bisa melihat bantuan berupa instruksi untuk penggunaan aplikasi pada *button start AR Wayang Golek*.

4. Quit

Pada *Use case quit* merupakan *use case* yang dimana untuk keluar dari aplikasi *Augmented Reality Markerless Wayang Golek Jawa Barat* ini.

**3.4 Skenario Use Case**

Skenario *use case* berguna untuk menguraikan apa saja yang dapat terjadi dalam sebuah *use case* baik dalam skenario normal atau alternatif. Skenario akan menunjukkan interaksi antara *user* dan *system* dengan detail secara bertahap.

**3.4.1 Skenario Use Case Tracking**

*Use case : Tracking*

Aktor : *User*

Kondisi awal : *User* sudah masuk ke dalam aplikasi

Skenario normal : *User* memberikan akses kamera pada sistem untuk bisa melakukan *tracking*.

Tabel 3. 1 Skenario *use case Tracking*

<b>User</b>	<b>Sistem</b>
	1. Akses kamera perangkat.
	2. Terdapat <i>button</i> “ <i>Start Tracking</i> ” untuk memulai <i>tracking</i> dan memunculkan wayang yang dipilih oleh <i>user</i> .
3. <i>User</i> menekan <i>button</i> “ <i>Start Tracking</i> ”.	
4. Lalu <i>User</i> mengarahkan kamera mulai dari dasar terdekat lantai / bidang datar lalu pelan-pelan menjauh untuk bisa melihat wayang secara lebih utuh.	
	5. Memunculkan karakter Wayang Golek AR 3D yang dipilih <i>user</i> .
Skenario alternatif : <i>User</i> ingin memberhentikan/ stop <i>tracking</i>	
	5. Akses kamera perangkat.

<b>User</b>	<b>Sistem</b>
	6. Terdapat <i>button</i> “ <i>Stop Tracking</i> ” untuk memberhentikan <i>tracking</i> dan menghilangkan pemunculan wayang yang dipilih oleh <i>user</i> .
7. <i>User</i> menekan <i>button</i> “ <i>Stop Tracking</i> ”.	
	8. Aplikasi memberhentikan <i>tracking</i> dan memberhentikan pemunculan Wayang Golek AR 3D yang dipilih <i>user</i> .

### 3.4.2 Skenario Use Case Start AR Wayang Golek

Use case : Start AR Wayang Golek

Aktor : *User*

Kondisi awal : *User* sudah menekan tombol *Start*

Skenario normal : *User* berhasil menampilkan dan mengenal Wayang Golek AR secara 3D.

Tabel 3. 2 Skenario use case Start AR Wayang Golek

<b>User</b>	<b>Sistem</b>
	1. Akses kamera perangkat.
2. Menekan <i>button</i> “ <i>Menu wayang</i> ” yang terdapat di pinggir kiri dan terdapat wayang apa saja yang ada di dalam aplikasi. Dan <i>user</i> memilih / mengklik wayang yang dipilih dan yang diinginkan. Dan klik <i>button</i> “ <i>Start Tracking</i> ”.	
	3. <i>Find</i> dan <i>get</i> objek sesuai dari data aset.
	4. Render objek 3D wayang golek
	5. Memunculkan wayang golek yang diinginkan.
6. Dapat menekan <i>button</i> “ <i>Voice</i> ”, untuk memunculkan suara.	
	7. Mengeluarkan <i>output</i> suara dari wayang yang sedang dipilih dan ditampilkan
8. Dapat juga menekan <i>button</i> “ <i>Profile</i> ”, untuk melihat dan membaca keterangan wayang yang dipilih.	
	9. Menampilkan keterangan-keterangan atau penjelasan mengenai wayang yang dipilih
10. Dapat juga menekan <i>button</i> “ <i>Zoom In</i> ” untuk memperbesar wayang yang dipilih.	
	11. Menampilkan wayang golek yang dipilih dengan tampak lebih besar.

<b>User</b>	<b>Sistem</b>
12. Dapat juga menekan <i>button</i> "Zoom Out" untuk memperkecil wayang yang dipilih.	
	13. Menampilkan wayang golek yang dipilih dengan tampak lebih kecil.
14. User juga dapat juga menekan <i>button</i> "Rotate" untuk membuat wayang yang dipilih berputar agar user dapat melihat sisi belakang depan samping.	
	15. Akan menampilkan wayang golek yang dipilih dengan berputar.
Skenario alternatif : Wayang golek tidak muncul.	
	1. Akses kamera perangkat.
2. Menekan <i>button</i> "Menu wayang" yang terdapat di pinggir kiri dan terdapat wayang apa saja yang ada di dalam aplikasi. Dan <i>user</i> memilih / mengklik wayang yang dipilih dan yang diinginkan. Dan klik <i>button</i> "Start Tarcking".	
	3. Aplikasi gagal meretrieve objek dari data aset.

### 3.4.3 Skenario Use Case Help

Use case : Help

Aktor : User

Kondisi awal : User sudah masuk dalam aplikasi

Skenario normal : Instruksi untuk penggunaan aplikasi berhasil ditampilkan.

Tabel 3. 3 Skenario use case Help

<b>User</b>	<b>Sistem</b>
1. Pilih <i>button</i> "Help" pada layar Menu Utama.	
	2. Memunculkan instruksi penggunaan aplikasi.
Skenario alternatif: Instruksi gagal ditampilkan	
1. Pilih <i>button</i> "Help" pada layar Menu Utama.	
	2. System tidak melakukan apapun, tetap berada di layar Menu Utama.

### 3.4.4 Skenario Use Case Quit

Use case : Quit

Aktor : User

Kondisi awal : User sudah masuk dalam aplikasi

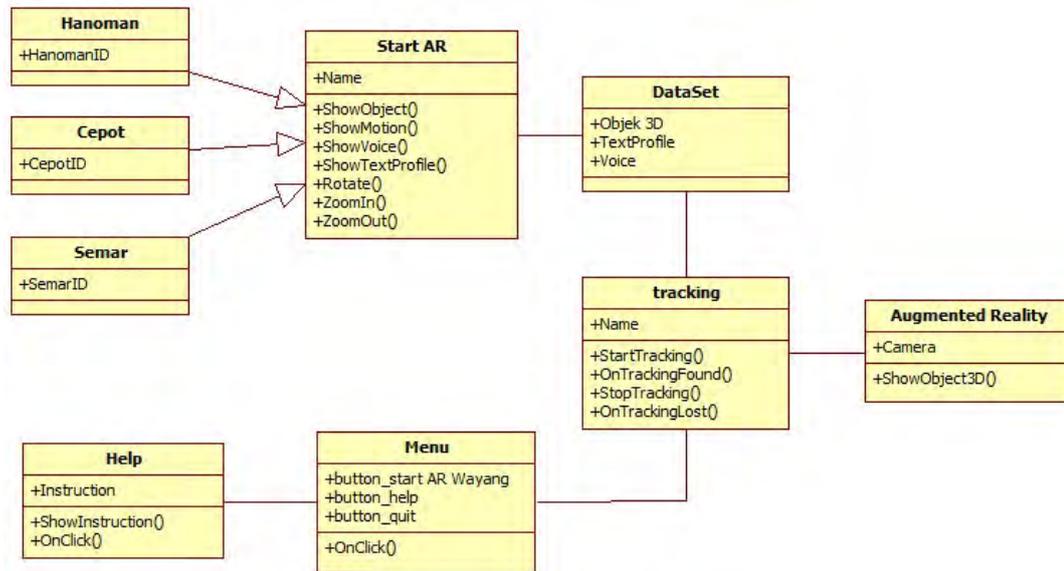
Skenario normal : *User* berhasil keluar dari aplikasi.

Tabel 3. 4 Skenario use case *Quit*

<b><i>User</i></b>	<b><i>Sistem</i></b>
1. Pilih <i>button</i> “ <i>Quit</i> ” pada layar Menu Utama.	
	2. Menutup aplikasi <i>Augmented Reality Markerless</i> Wayang Golek Jawa Barat dan juga dari halaman Menu Utama.
Skenario alternatif: Instruksi gagal ,gagal keluar dari aplikasi.	
1. Pilih <i>button</i> “ <i>Quit</i> ” pada layar Menu Utama.	
	2. <i>System</i> tidak melakukan apapun , dan tidak keluar dari aplikasi <i>Augmented Reality Markerless</i> Wayang Golek Jawa Barat dan tetap berada di halaman Menu Utama.

### 3.5 Class Diagram

*Class diagram* berguna untuk menggambarkan kelas-kelas yang ada di dalam aplikasi serta hubungan antar kelas. Berikut adalah *class diagram* untuk aplikasi yang akan dikembangkan. Gambar 3.2 Menampilkan *class diagram* untuk aplikasi.



Gambar 3. 2 Class Diagram

Gambar 3.2 menunjukkan *class diagram* yang mempresentasikan kelas-kelas yang ada di dalam aplikasi. Berikut penjelasan untuk setiap kelas pada gambar 3.2:

#### 1. Start AR

Pada kelas *Start AR* ini memiliki atribut *name*. Dan memiliki *method* seperti *ShowObject*, *ShowMotion*, *ShowVoice*, *ShowTextProfile*, *Rotate*, *ZoomIn*, *ZoomOut*. Dan terdapat kelas *hanoman*, *cepot*, dan *semar* yang merupakan bagian lebih spesifik/ khusus yang terdapat dalam *Start AR*.

#### 2. DataSet

Kelas *dataset* memiliki atribut objek 3D, *text profile*, dan *voice*.

#### 3. Tracking

Kelas *tracking* ini memiliki atribut *name* dengan *method* *StartTracking*, *OnTrackingFound*, *StopTracking* dan *OnTrackingLost*.

#### 4. Augmented Reality

Kelas *Augmented Reality* ini memiliki atribut *camera* dan memiliki *method ShowObject3D*.

#### 5. Menu

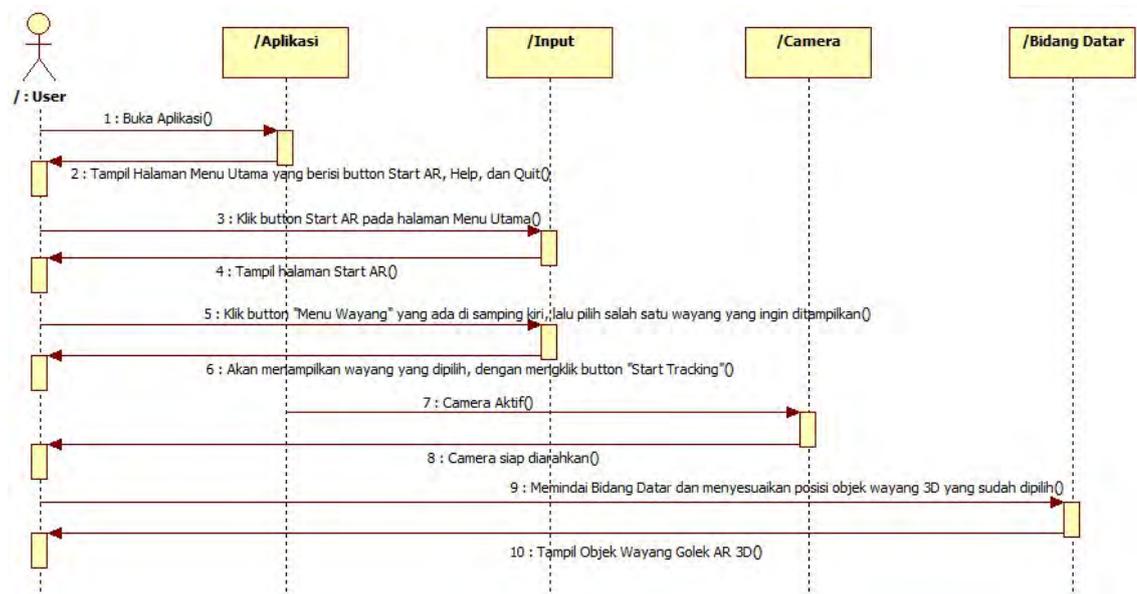
Kelas Menu memiliki tiga atribut yaitu *button\_start* AR wayang, *button\_quit* dan *button\_help*. Serta *method* untuk menjalankannya yaitu dengan *onClick*.

#### 6. Help

Kelas *help/bantuan* ini memiliki atribut *instruction* pemakaian aplikasi dan *button\_home* menu serta *method ShowInstruction* dan *onClick*.

### 3.6 Sequence Diagram

*Sequence* diagram berguna untuk menggambarkan skenario atau rangkaian langkah – langkah yang mendeskripsikan bagaimana entitas dalam sistem berinteraksi. Berikut adalah *sequence* diagram untuk aplikasi yang akan dikembangkan. Gambar 3.3 dan gambar 3.4 menampilkan *sequence* diagram untuk aplikasi.

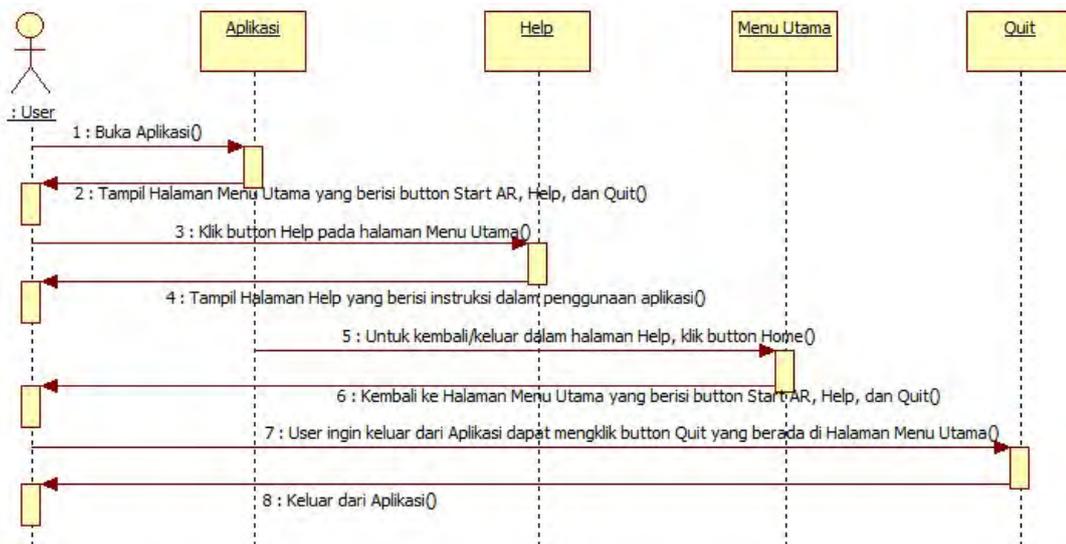


Gambar 3.3 *Sequence Diagram Start AR*

Gambar 3.3 menunjukkan *sequence* diagram *Start AR* yang menggambarkan skenario di dalam aplikasi. Berikut penjelasan untuk setiap kelas pada gambar 3.3. Pertama-tama *user* dapat membuka aplikasi “AR Wayang Golek Jabar”, lalu kemudian

sistem aplikasi merespon dengan menampilkan halaman Menu Utama yang berisi button *Start AR* untuk menampilkan objek-objek wayang AR, *Help* untuk melihat instruksi, dan *Quit* untuk keluar dari aplikasi. Kemudian ketika *user* menekan *button Start AR* pada halaman Menu Utama maka sistem aplikasi akan menampilkan halaman *Start AR*.

*User* menekan simbol *button* “Menu Wayang” yang ada di halaman *Start AR* yang berada di samping kiri, lalu *user* dapat melihat wayang apa saja yang ada dalam aplikasi dan dapat memilih untuk ditampilkan. Setelah *user* menekan dan memilih wayang yang ingin ditampilkan, maka sistem akan merespon dengan menampilkan wayang yang dipilih oleh *user* dengan mengklik “*Start Tracking*” untuk memunculkan dan memulai pentrackingan. Kemudian kamera akan aktif setelah menekan “*Start Tracking*” tersebut. *User* mulai mengarahkan kamera dimulai dari dasar terdekat lantai/ bidang datar lalu pelan-pelan menjauh untuk bisa melihat wayang secara lebih utuh. Setelah itu sistem mulai mendeteksi/ memindai permukaan dan posisi, dan bila sudah terdeteksi, maka akan muncul Objek Wayang Golek AR secara 3D yang sudah dipilih oleh *user*.



Gambar 3. 4 *Sequence Diagram Help dan Quit*

Gambar 3.4 menunjukkan *sequence diagram Help dan Quit* yang menggambarkan skenario di dalam aplikasi. Berikut penjelasan untuk setiap kelas pada gambar 3.4. Pertama-tama *user* dapat membuka aplikasi “AR Wayang Golek Jabar”, lalu kemudian sistem aplikasi merespon dengan menampilkan halaman Menu Utama yang berisi button

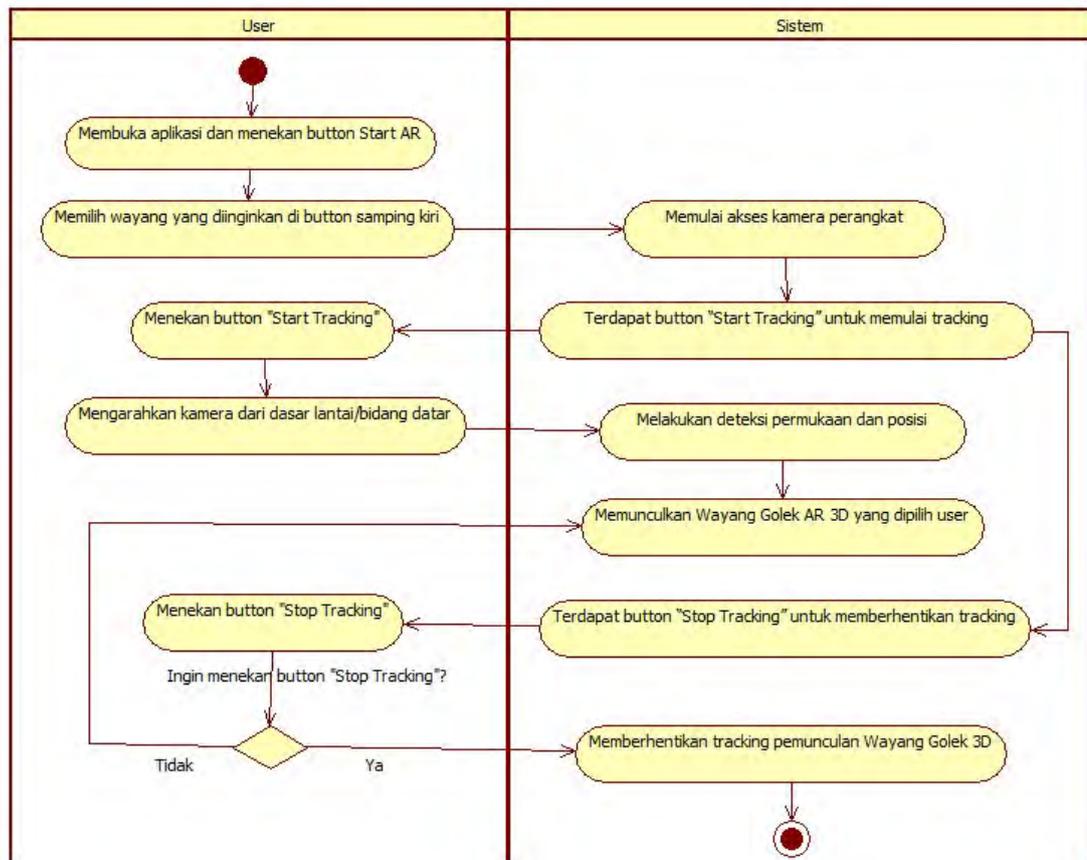
*Start AR* untuk menampilkan objek-objek wayang AR, *Help*, dan *Quit*. Kemudian *user* ingin menekan *button Help* pada halaman Menu Utama maka sistem aplikasi akan menampilkan halaman *Help* yang berisi instruksi dalam penggunaan aplikasi “AR Wayang Golek Jabar”. Dan di halaman *Help* terdapat *button Home* untuk kembali ke halaman Menu Utama, *User* dapat mengklik *button Home* untuk kembali ke Halaman Menu Utama yang berisi *button Start AR*, *Help*, dan *Quit* tersebut. Kemudian jika *user* ingin keluar dari aplikasi “AR Wayang Golek Jabar” maka dapat menekan *button Quit* yang berada di halaman Menu Utama.

### **3.7 Activity Diagram**

*Activity diagram* adalah diagram yang digunakan untuk melanjutkan aktivitas interaksi *user* dan *system* dalam sebuah *use case*. Dalam pembuatannya, *activity diagram* menggunakan skenario *use case* sebagai referensi atas aktivitas apa saja yang terjadi.

#### **3.7.1 Activity Diagram Use Case Tracking**

Dalam *use case Tracking* akan ada interaksi antara *user* dengan *system* dimana *user* akan menentukan apakah sistem dapat mengakses kamera atau tidak.



Gambar 3. 5 Activity Diagram Use Case Tracking

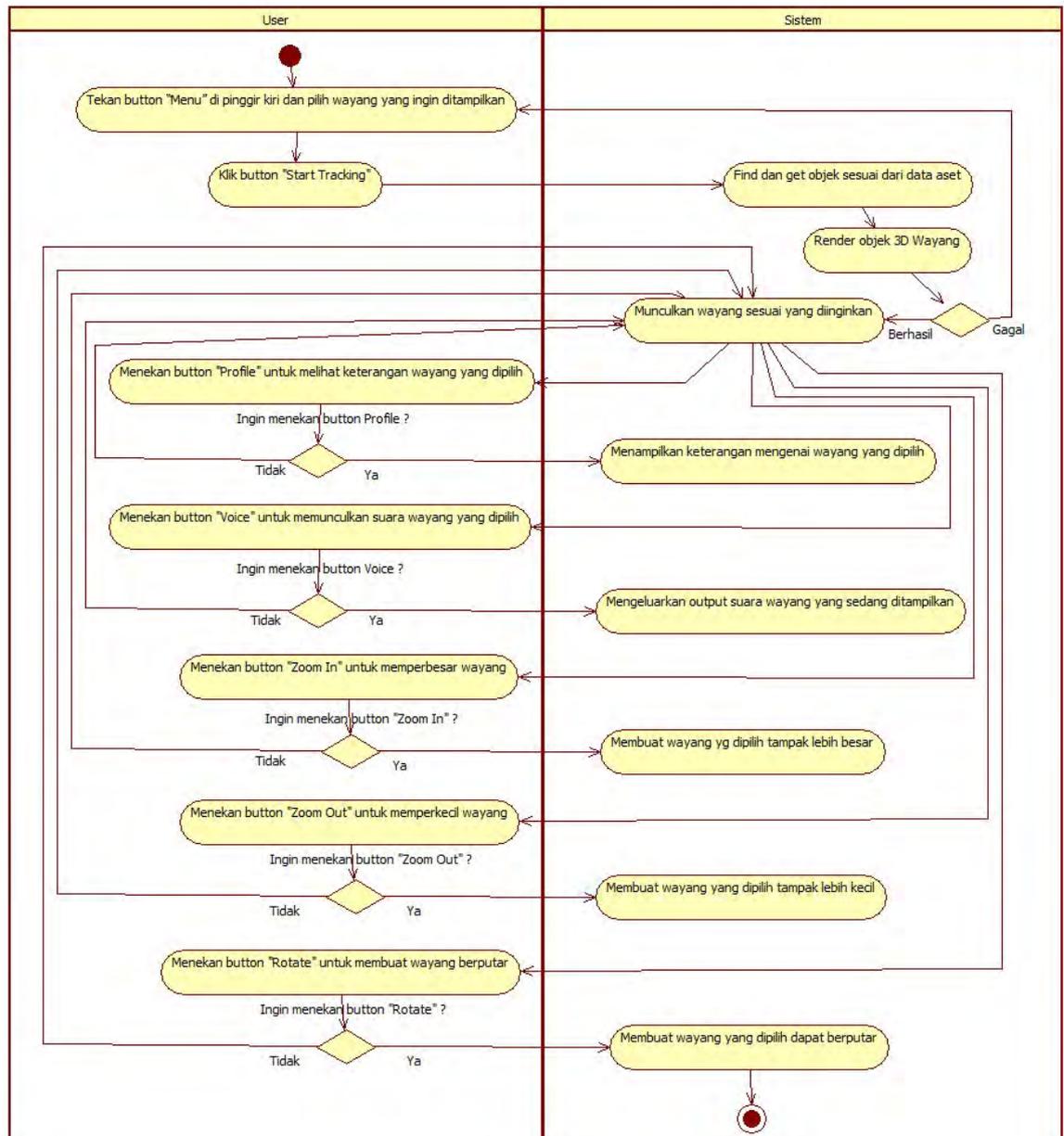
Gambar 3.5 adalah *activity diagram* dari *use case Tracking* dimana *user* bisa mengakses kamera dalam aplikasi.

Pertama-tama, setelah *user* membuka aplikasi dengan menekan *button* "Start AR", Lalu *user* memilih wayang dengan menekan *button* "menu wayang" yang berada di samping kiri, dan memilih / mengklik wayang yang diinginkan. Dan sistem mulai mengakses kamera perangkat, lalu dibagian bawah terdapat *button* "Start Tracking" untuk memulai *tracking*/memunculkan wayang 3D. Setelah *user* menekan *button* "Start Tracking" dan *user* mulai mengarahkan kamera mulai dari dasar terdekat lantai/ bidang datar lalu pelan-pelan menjauh untuk bisa melihat wayang secara lebih utuh. Setelah itu sistem mulai mendeteksi permukaan dan posisi, dan bila sudah terdeteksi, maka akan muncul Wayang Golek AR secara 3D yang dipilih *user*. Kemudian setelah *Start Tracking*, dalam sistem terdapat *button* "Stop Tracking" juga yang berada dibagian bawah, jika *user* ingin memberhentikan *Tracking* pemunculan wayang golek AR maka *tekan button* "Stop

*Tracking*” tersebut. Dan jangan tekan *button* “*Stop Tracking*” jika user masih ingin tetap menampilkan wayang golek AR maka sistem akan tetap menampilkan wayang golek AR 3D yang sedang dipilih *user*.

### **3.7.2 Activity Diagram Use Case Start AR Wayang Golek**

Dalam *use case Start AR Wayang Golek* akan ada interaksi antara *user* dengan sistem dimana *user* dapat menekan *button* agar sistem dapat merender objek 3D wayang dan dapat menampilkan wayang yang diinginkan oleh *user*. Dan *User* dapat menekan *button-button* agar sistem dapat mengeluarkan *output* suara serta menampilkan keterangan dan penjelasan mengenai wayang yang dipilih, bisa memperbesar, memperkecil dan juga dapat merotasi wayang yang dipilih oleh *user*.



Gambar 3. 6 Activity Diagram Use Case Start AR Wayang Golek

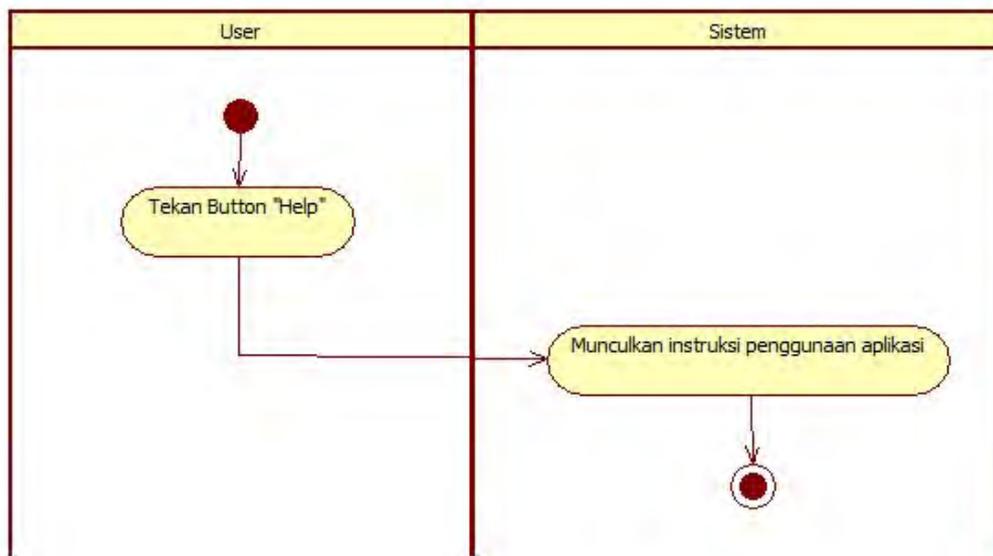
Penjelasan untuk gambar 3.6 adalah sebagai berikut:

Pertama-tama, *user* terlebih dahulu untuk memilih wayang yang ingin ditampilkan, dengan menekan *button* “Menu Wayang” yang ada disamping kiri yang berisi wayang apa saja yang ada dalam aplikasi, lalu *user* menekan *button* “Start Tracking” yang berada di bawah. Setelah *user* memilih wayang dan *start tracking*, maka sistem akan menemukan dari data aset lalu menampilkan dan *user* bisa melihat render objek wayang 3D AR. Tetapi jika gagal, sistem tidak akan memberi respon apapun dan *user* harus memilih ulang wayang yang ingin ditampilkan dengan menekan *button* “Menu Wayang” kembali.

Lalu setelah *user* bisa melihat render objek wayang dalam 3D, setelah itu juga *user* juga bisa menekan *button* “*Profile*” yang digunakan agar *user* dapat melihat, membaca, dan mengenal keterangan wayang yang dipilih. Kemudian *user* juga bisa menekan *button* “*Voice*” untuk memunculkan suara wayang tersebut, lalu sistem mengeluarkan *output* suara dari wayang yang sedang dipilih dan ditampilkan. Tidak hanya *Profile* dan *Voice*, *user* juga dapat menekan *button* “*ZoomIn*” yang digunakan untuk memperbesar wayang, lalu ada juga *button* “*ZoomOut*” jika *user* ingin memperkecil kembali wayang yang sedang ditampilkan, dan juga setelah itu terdapat juga *button* “*Rotate*” yang digunakan untuk membuat wayang dapat berputar dan *user* dapat melihat sisi depan, belakang, dan samping dari wayang yang sedang dipilih *user*.

### 3.7.3 Activity Diagram Use Case Help

Dalam *use case help*, *user* menekan *button* “*Help*” yang akan ditampilkan oleh sistem. *Activity diagram use case* Bantuan menggambarkan aktivitas *user* dan sistem untuk melihat instruksi pemakaian aplikasi.



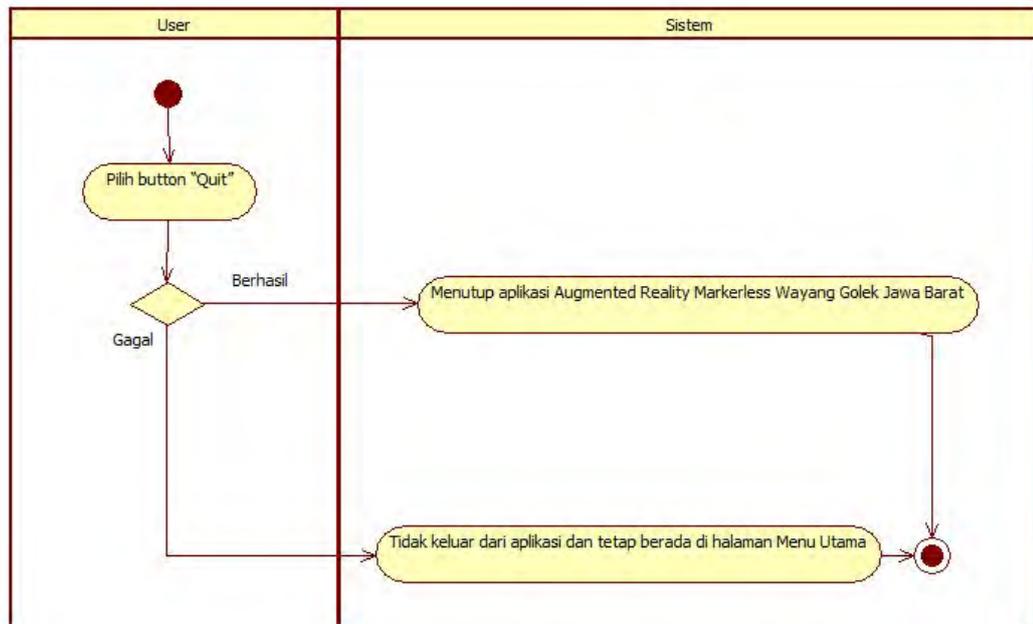
Gambar 3. 7 Activity Diagram Use Case Help

Penjelasan untuk gambar 3.7 adalah sebagai berikut:

Pertama-tama, *user* terlebih dahulu mengakses menu bantuan dengan menekan tombol “*Help*” di halaman Menu Utama. Setelah itu, sistem akan menampilkan instruksi dalam pemakaian aplikasi.

### 3.7.4 Activity Diagram Use Case Quit

Dalam *use case quit* akan ada interaksi antara *user* dengan sistem dimana *user* dapat menekan *button* “*Quit*” dan sistem akan memproses. *Activity diagram use case* Quit adalah aktivitas *user* dan sistem yang dimana untuk keluar dari aplikasi *Augmented Reality Markerless* Wayang Golek Jawa Barat ini.



Gambar 3. 8 Activity Diagram Use Case Quit

Penjelasan untuk gambar 3.8 adalah sebagai berikut:

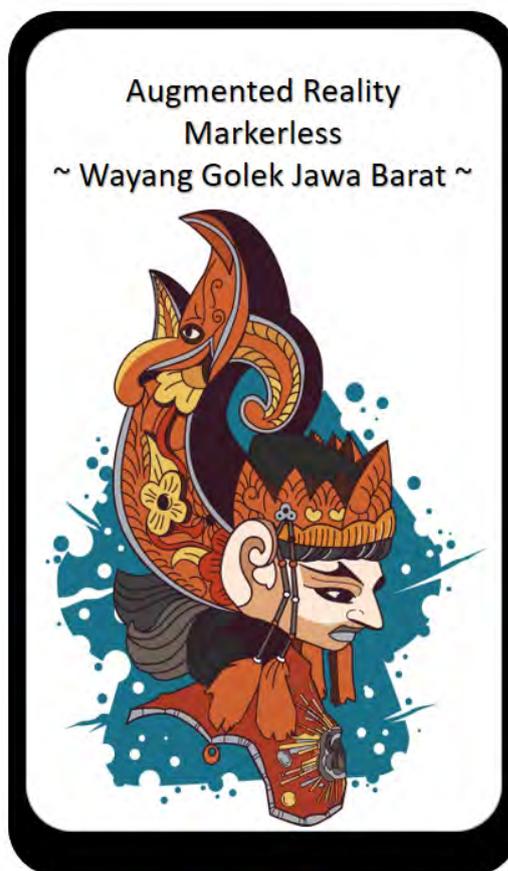
Pertama-tama, *user* terlebih dahulu mengakses menu keluar dengan menekan tombol “*Quit*” di halaman Menu Utama. Setelah itu, sistem akan menutup aplikasi *Augmented Reality Markerless* Wayang Golek Jawa Barat. Tetapi jika gagal, maka sistem tidak akan keluar dari aplikasi dan tetap berada di halaman Menu Utama.

### 3.8 Rancangan Antar Muka

Dalam sub bab ini, dijelaskan rancangan antar muka atau UI yang merupakan sketsa awal bagaimana antarmuka akan dibuat. Terdapat beberapa kategori antar muka antara lain UI *welcome page*, UI menu utama, UI *help*, serta UI *Start AR*.

#### 1. Rancangan antar muka *Welcome Page*

Gambar 3.9 ini merupakan rancangan antar muka *Welcome Page* yang merupakan gambar awal, Logo, dan Judul Aplikasi.



Gambar 3. 9 Rancangan Antar Muka *Welcome Page*

#### 2. Rancangan antar muka Menu Utama

Gambar 3.10 adalah rancangan antar muka yang terdiri dari beberapa *button* seperti *button "Start AR"*, *button "Help"*, dan *button "Quit"*. Penjelasan lebih lanjut sebagai berikut:

a. *Button Start AR*

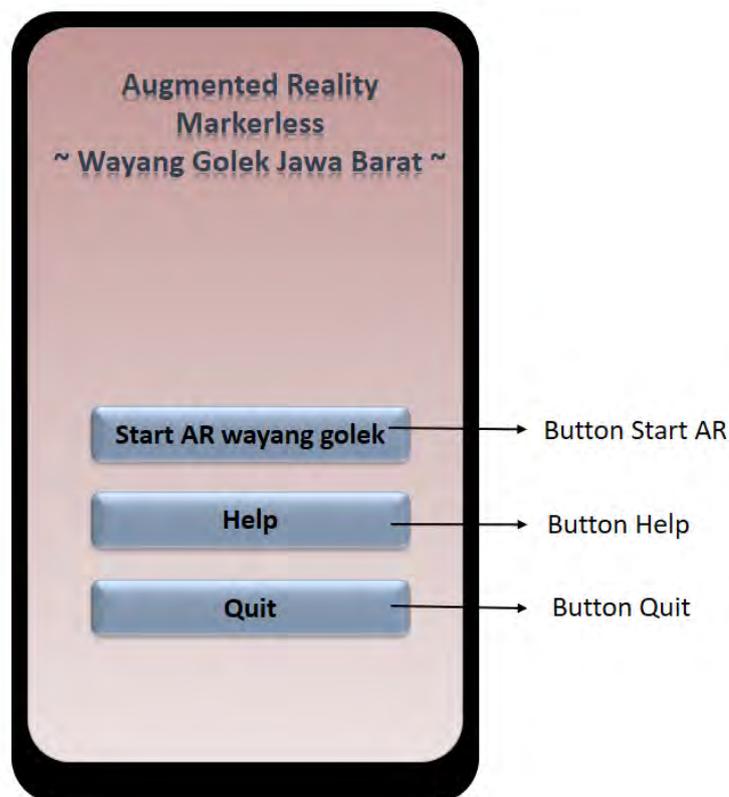
User bisa melihat memvisualisasikan wayang golek Jawa Barat untuk pengenalan dan pembelajaran, dan dapat memilih wayang golek mana yang ingin ditampilkan dan dilihat, dapat diperbesar dan diperkecil serta dirotasi. Dan dapat mengeluarkan suara. Serta dalam menu ini juga dapat melihat keterangan-keterangan atau penjelasan mengenai wayang yang dipilih.

b. *Button Help*

Melalui *button* ini, *user* dapat melihat instruksi atau cara menggunakan aplikasi ini untuk dapat menampilkan atau memvisualisasikan wayang golek tersebut untuk mempermudah *user* dalam menggunakan aplikasi ini.

c. *Button Quit*

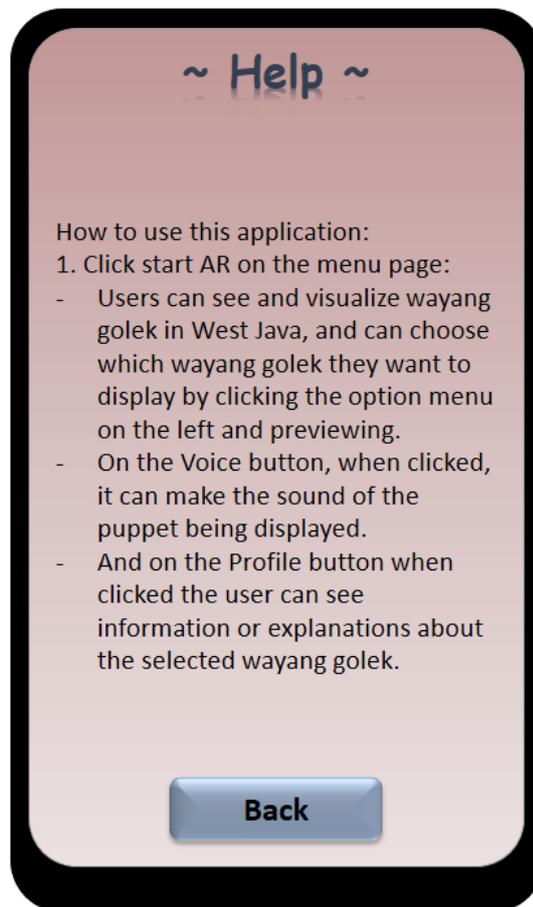
*Button quit* ini , dipakai *user* untuk keluar dari aplikasi *Augmented Reality Markerless Wayang Golek Jawa Barat* ini.



Gambar 3.10 Rancangan Antar Muka Menu Utama

### 3. Rancangan antar muka *Help*

Gambar 3.11 adalah rancangan antar muka *Help* untuk UI menampilkan instruksi. Untuk instruksi, *user* hanya bisa mengakses fungsi tersebut di awal aplikasi yaitu di Menu Utama saja. Pada *button* ini, *user* dapat melihat instruksi atau cara menggunakan aplikasi ini untuk dapat menampilkan atau memvisualisasikan wayang golek tersebut untuk mempermudah *user* dalam menggunakan aplikasi ini. Dan juga di UI *Help* ini terdapat *button* “back” yang digunakan untuk menutup halaman UI *Help* dan kembali ke halaman Menu Utama, lalu bisa memilih / menekan *button* *Start AR* atau *Quit*.



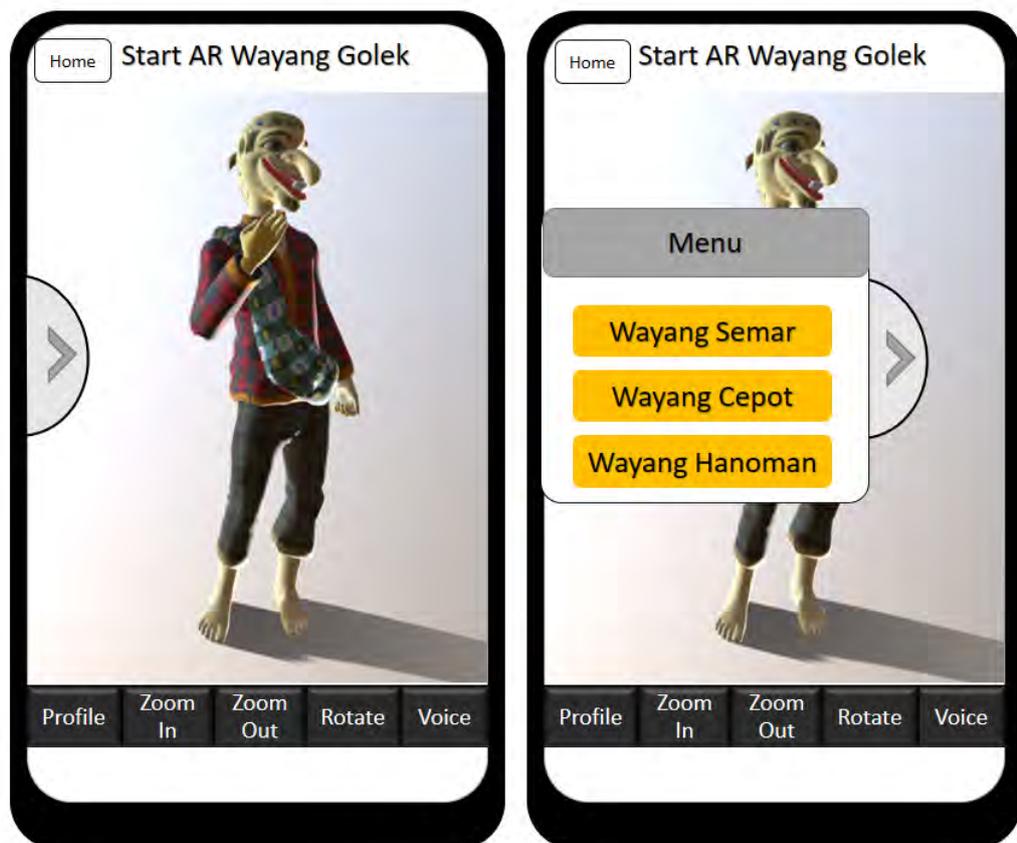
Gambar 3. 11 Rancangan Antar Muka *Help*

### 4. Rancangan antar muka *Start AR*

Pada Gambar 3.12 merupakan rancangan antar muka *Start AR*. Pada UI *Start AR* dapat menampilkan Wayang Golek AR, ada *button* “Menu Wayang”, *button* “*Profile*”,

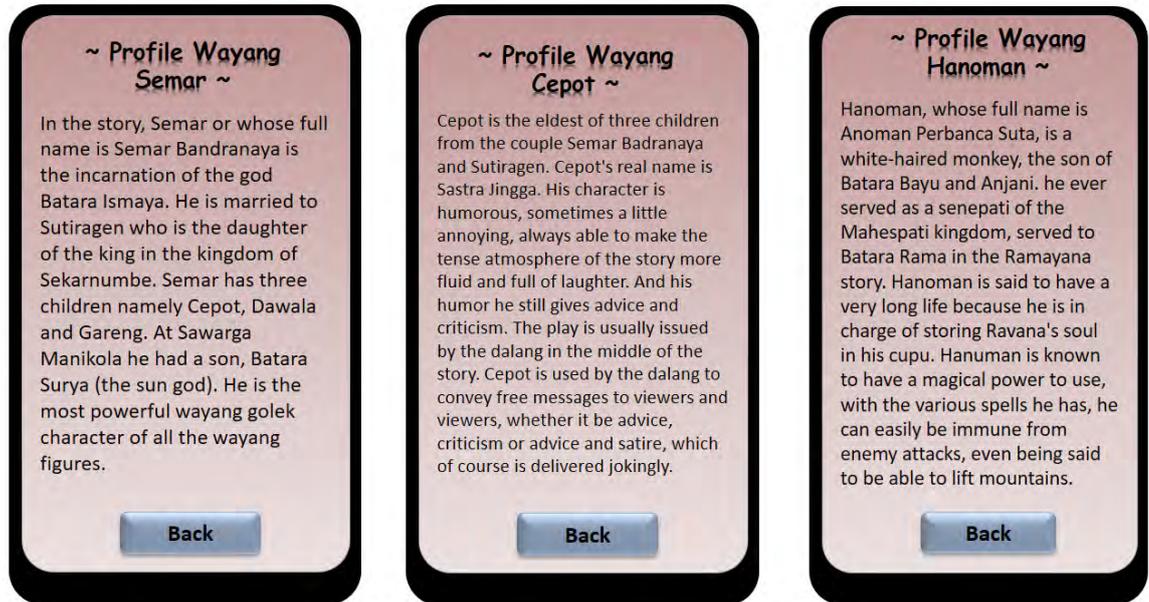
button “Zoom In”, button “Zoom Out”, button “Rotate” serta button “Voice”. Penjelasan lebih lanjut sebagai berikut:

Pada button “Menu Wayang” akan berisi wayang apa saja yang ada di dalam aplikasi yang nantinya diklik / dipilih user dan user dapat melihat wayang golek yang dipilih dan yang ingin ditampilkan. Lalu pada button “Profile” ini juga dapat melihat keterangan-keterangan atau penjelasan mengenai wayang yang dipilih. Dan juga button “Zoom In” yang dapat digunakan memperbesar wayang golek yang sedang ditampilkan, lalu button “Zoom Out” yang dapat memperkecil wayang golek yang sedang ditampilkan. Ada juga button “Rotate” digunakan agar wayang golek yang sedang ditampilkan itu bergerak memutar sehingga user dapat melihat sisi depan, belakang dan samping. Terdapat juga button “Voice” yang digunakan agar wayang dapat mengeluarkan suara, dan untuk button “Home” yang berada di kiri pojok atas, digunakan untuk user jika ingin kembali ke halaman Menu Utama.



Gambar 3. 12 Rancangan Antar Muka Start AR

Dan juga disini terdapat Isi *button* “Profile” tiap wayang:



Gambar 3. 13 Rancangan Antar Muka *Start AR*– *button* “Profile”

## BAB IV

### IMPLEMENTASI DAN PENGUJIAN PERANGKAT LUNAK

#### 4.1 Spesifikasi Kebutuhan Perangkat Keras

Pada pembuatan aplikasi “Memperkenalkan Kesenian Wayang Golek Jawa Barat dengan *Augmented Reality* secara *Markerless* Berbasis Android” untuk perangkat mobile ini dirancang menggunakan sebuah laptop yang memiliki spesifikasi:

1. RAM 8 GB
2. NVIDIA GeForce GTX 1050 TI
3. Ruang penyimpanan SSD 256 GB

Untuk menguji apakah fungsi dalam aplikasi yang telah dibuat berjalan sesuai hasil yang diharapkan, dibutuhkan perangkat keras *smartphone* dengan kriteria berikut:

1. Perangkat yang menjalankan OS Android minimal Android *version* 5.1
2. *Smartphone* yang cocok dengan Google ARCore serta fungsi AR.
3. RAM minimal 3 GB.
4. Minimal kamera 5 MP (Mega Pixel)

#### 4.2 Pengujian Antar Muka

Tampilan antar muka akan berfokus pada hasil akhir UI berdasarkan rancangan sebelumnya. Dalam aplikasi terdapat beberapa UI yaitu:

##### 1. *Welcome Page*

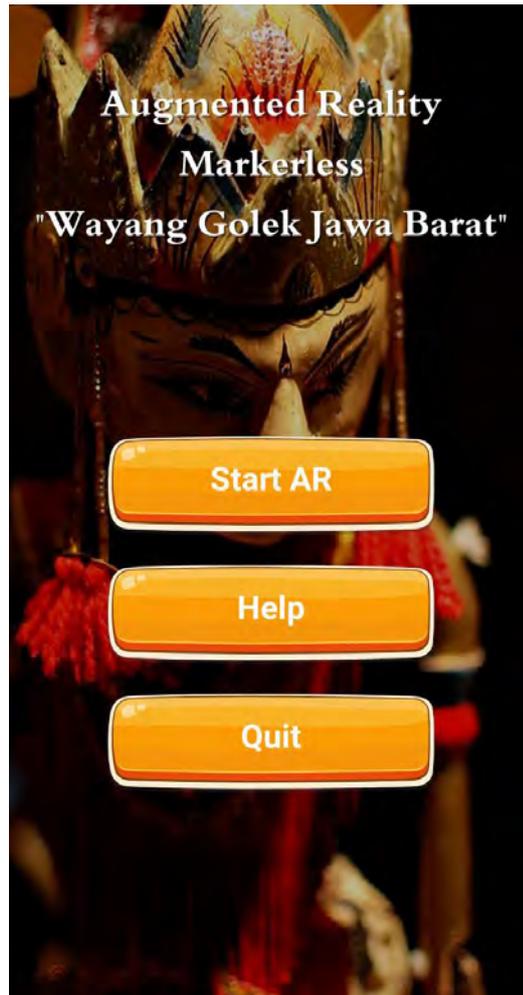
Gambar 4.1 menunjukkan hasil akhir rancangan antar muka untuk *welcome page*. Pada gambar 4.1 ini merupakan tampilan awal pada saat membuka aplikasinya.



Gambar 4. 1 *UI Welcome Page*

## 2. Menu Utama

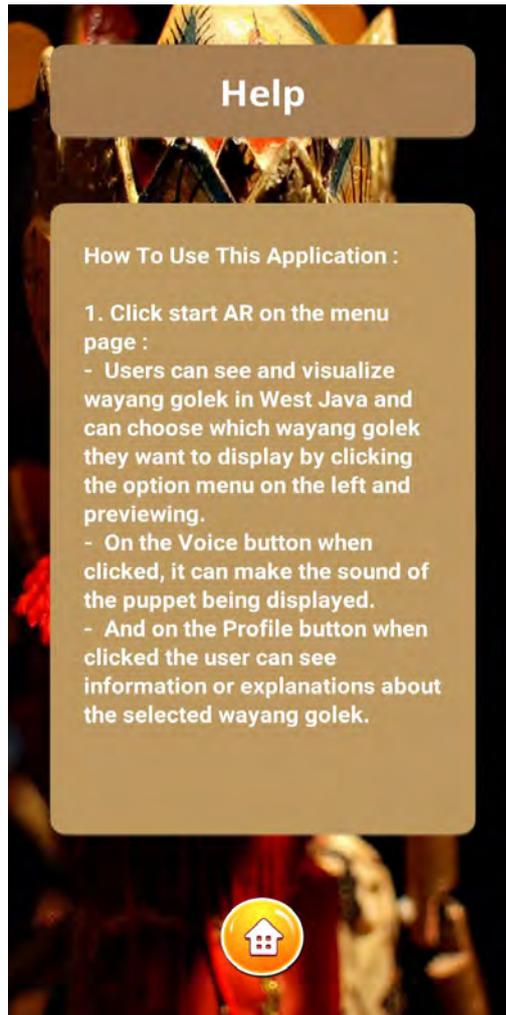
Dan untuk hasil akhir rancangan antar muka untuk Menu Utama. Pada gambar 4.2 ini terdapat tiga buah *button* yaitu *button* “*Start AR*”, *button* “*Help*”, dan *button* “*Quit*”. *Button* “*Start AR*” berguna untuk menampilkan Wayang Golek AR yang di dalamnya terdapat tombol untuk memilih menu wayang golek mana yang ingin ditampilkan. Sedangkan untuk *button* “*Help*” berguna untuk mengakses instruksi pemakaian aplikasinya, dan untuk *button* “*Quit*” yaitu untuk keluar dari aplikasinya.



Gambar 4. 2 UI Menu Utama

### 3. *Help*

Pada UI *Help* di gambar 4.3 *user* dapat melihat instruksi pemakaian aplikasi. Dan untuk menutup UI *Help* , *user* harus menekan button *Back/ Home* maka akan kembali ke halaman Menu Utama.



Gambar 4. 3 UI Help

#### 4. Start AR

Pada UI *Start AR* , sistem akan menampilkan AR hasil wayang yang dipilih dari *button* yang ada disamping untuk mengetahui ada apa saja wayang yang ada di aplikasi serta memilih wayang mana yang ingin ditampilkan, lalu setelah itu *user* klik *button* “*Start Tracking*” untuk memunculkan Wayang Golek AR 3D , dan untuk memberhentikan /menghilangkan wayangnya klik *button* “*Stop Tracking*” dan terdapat juga beberapa *button* antara lain *button profile*, *button zoom in*, *button zoom out*, *button rotate*, dan *button voice*.

Pada gambar 4.4 dan gambar 4.5 terdapat beberapa komponen *button*.

##### a. *Button* Menu Wayang

Pada *button* “*Menu Wayang*” yang terdapat di samping kiri berisi wayang apa saja yang ada di dalam aplikasi yang nantinya diklik / dipilih *user* dan *user* dapat melihat

wayang golek yang dipilih dan yang ingin ditampilkan. Didalam Menu Wayang ini terdapat *button\_wayang semar*, *button\_wayang cepot*, dan *button\_wayang hanoman*, yang misalnya user mengklik *button\_wayang semar* yang terdapat di *button Menu Wayang*, maka akan muncul dan akan ditampilkan wayang semar 3D AR.

b. *Button Start Tracking*

Pada *button* ini user menekan *button "Start Tracking"* untuk mentracking dan memunculkan Wayang Golek 3D AR.

c. *Button Stop Tracking*

Dan *button "Stop Tracking"* untuk memberhentikan /menghilangkan Wayang Golek 3D AR yang sedang ditampilkan.

d. *Button Profile*

Dalam *button "Profile"* ini juga dapat melihat keterangan-keterangan atau penjelasan mengenai wayang yang dipilih seperti di gambar 4.6. Dan untuk menutup halaman *Profile* ini terdapat *button "Home"*.

e. *Button Zoom In*

Pada *button* ini dapat memperbesar wayang golek yang sedang dipilih dan ditampilkan.

f. *Button Zoom Out*

Dan juga dalam *button* ini dapat memperkecil wayang golek yang sedang dipilih dan ditampilkan.

g. *Button Rotate*

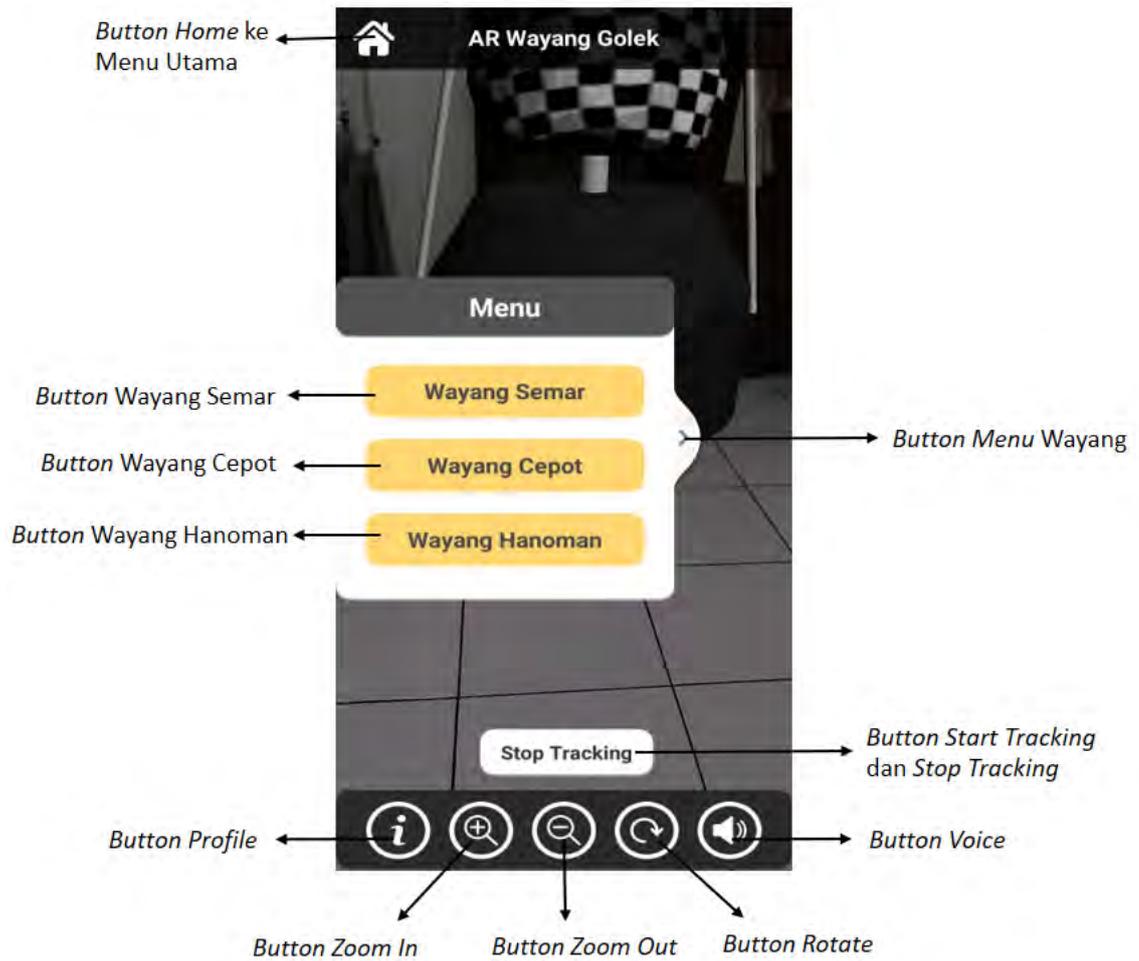
Serta dalam *button* ini dapat membuat wayang golek yang sedang dipilih dan ditampilkan itu bergerak memutar sehingga user dapat melihat sisi depan, belakang dan samping.

h. *Button Voice*

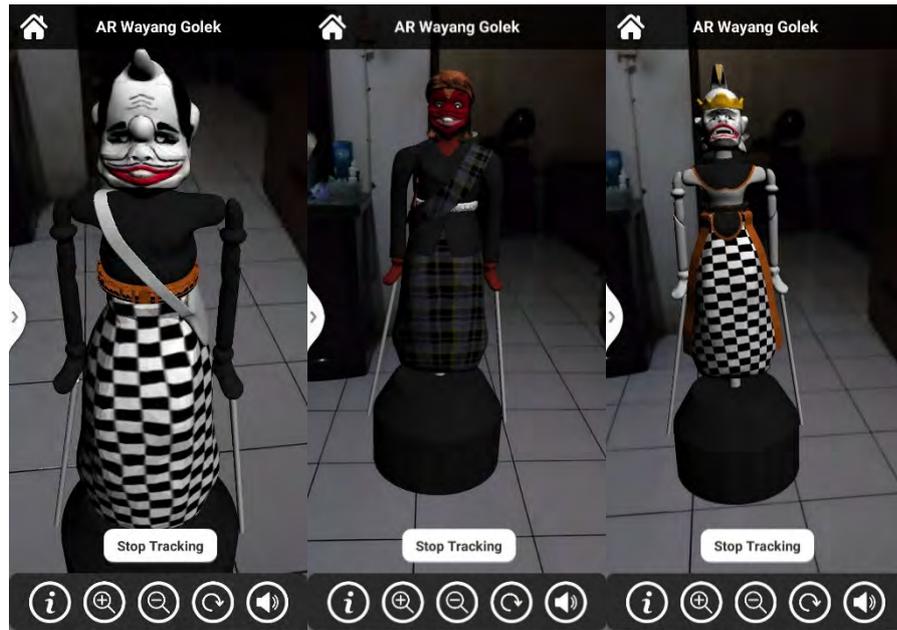
Dan juga terdapat *button "Voice"* yang digunakan agar wayang dapat mengeluarkan suara.

i. *Button Home*

Untuk *button* "Home" ini yang berada di kiri pojok atas, digunakan untuk *user* jika ingin kembali ke halaman Menu Utama.

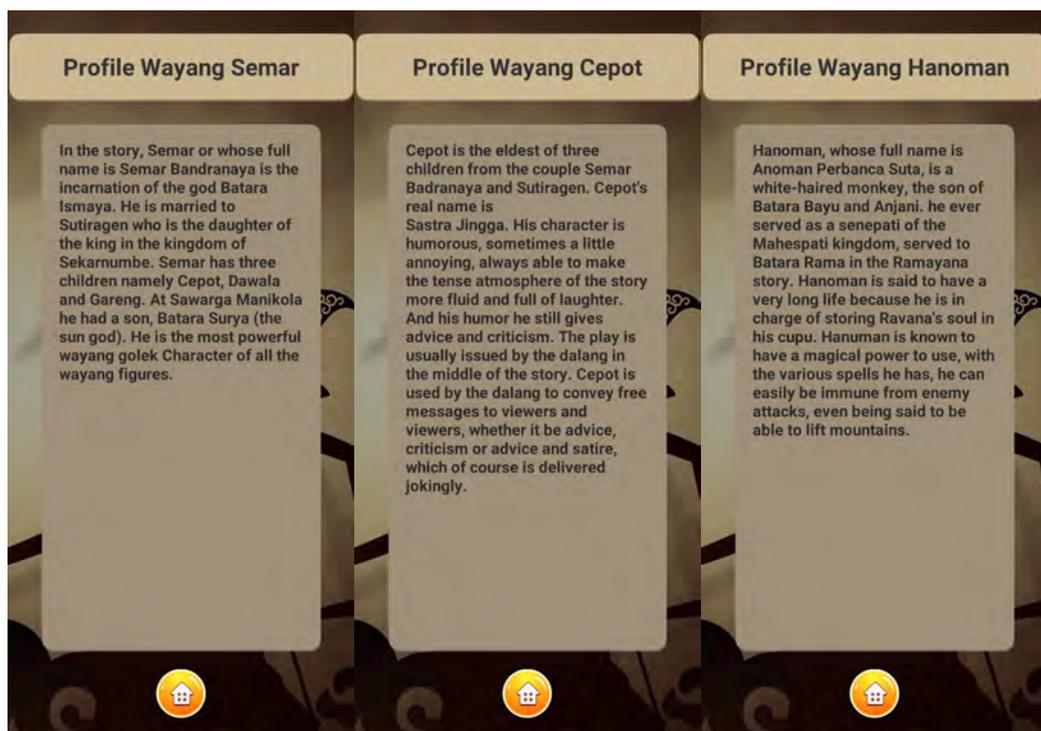


Gambar 4. 4 UI Start AR



Gambar 4. 4 Contoh *Tracking Wayang Golek AR*

Dan juga disini terdapat isi *button "Profile"* tiap wayang:



Gambar 4. 5 *UI Start AR- button "Profile"*

### 4.3 Pengujian Fungsi

Pengujian fungsi akan dilakukan dengan metode *black box* dengan tipe pengujian fungsionalitas dimana pengujian akan berfokus pada kebutuhan fungsionalitas aplikasi. Pengujian dilakukan dengan maksud untuk melihat apakah fungsi-fungsi perangkat lunak sudah berjalan dengan yang diharapkan atau belum. Berikut ini adalah tabel-tabel dari fungsi perangkat lunak yang diuji :

#### 4.3.1 Hasil Pengujian Fungsi Menu Utama

Tabel 4. 1 Hasil Pengujian Fungsi Menu Utama

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>button Start AR</i>	Sistem akan masuk ke <i>scene Start AR</i>	Sistem masuk ke <i>sence Start AR</i>	Berhasil
2	Klik <i>button Help</i>	Sistem akan masuk ke <i>scene Help</i>	Sistem masuk ke <i>sence Help</i>	Berhasil
3	Klik <i>button Quit</i>	Sistem akan keluar dari aplikasi AR Wayang Golek <i>Markerless</i>	Sistem keluar dari aplikasi AR Wayang Golek <i>Markerless</i>	Berhasil

#### 4.3.2 Hasil Pengujian Fungsi *Help*

Tabel 4. 2 Hasil Pengujian Fungsi *Help*

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>button Home</i>	Sistem akan keluar dari <i>sence Help</i> dan kembali ke halaman Menu Utama	Sistem keluar dari <i>sence Help</i> dan kembali ke halaman Menu Utama	Berhasil

### 4.3.3 Hasil Pengujian Fungsi *Start AR*

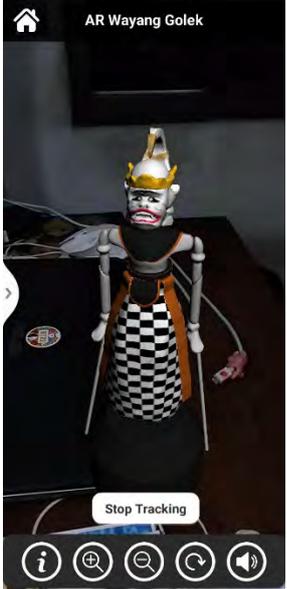
Tabel 4. 3 Hasil Pengujian Fungsi *Start AR*

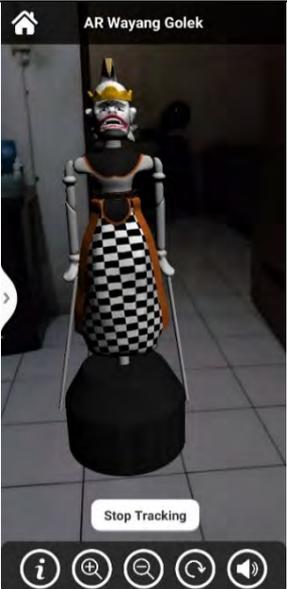
No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>button</i> "Menu Wayang"	Sistem akan menampilkan menu-menu wayang/ ada wayang apa saja dalam aplikasi	Sistem menampilkan menu-menu wayang/ada wayang apa saja dalam aplikasi	Berhasil
2	Klik <i>button</i> "Wayang Semar"	Sistem akan menampilkan wayang golek semar 3D AR	Sistem menampilkan wayang golek semar 3D AR	Berhasil
3	Klik <i>button</i> "Wayang Cepot"	Sistem akan menampilkan wayang golek cepot 3D AR	Sistem menampilkan wayang golek cepot 3D AR	Berhasil
4	Klik <i>button</i> "Wayang Hanoman"	Sistem akan menampilkan wayang golek hanoman 3D AR	Sistem menampilkan wayang golek hanoman 3D AR	Berhasil
5	Klik <i>button</i> "Start Tracking"	Sistem akan menampilkan / mentracking wayang yang sedang dipilih	Sistem menampilkan / mentracking wayang yang sedang dipilih	Berhasil
6	Klik <i>button</i> "Stop Tracking"	Sistem akan memberhentikan tracking pemunculan wayang yang sedang dipilih	Sistem memberhentikan tracking pemunculan wayang yang sedang dipilih	Berhasil
7	Klik <i>button</i> "Profile"	Sistem akan masuk ke halaman Profile wayang yang sedang dipilih	Sistem masuk ke halaman Profile wayang yang sedang dipilih	Berhasil
8	Klik <i>button</i> "Home" di dalam tampilan "Profile"	Sistem akan menutup halaman <i>Profile</i> dan kembali ke halaman <i>Start AR</i> yang sedang menampilkan wayang yang dipilih	Sistem menutup halaman <i>Profile</i> dan kembali ke halaman <i>Start AR</i> yang sedang menampilkan wayang yang dipilih	Berhasil
9	Klik <i>button</i> "Zoom In"	Sistem akan menampilkan wayang yang dipilih secara lebih besar	Sistem menampilkan wayang yang dipilih secara lebih besar	Berhasil
10	Klik <i>button</i> "Zoom Out"	Sistem akan menampilkan wayang yang dipilih secara lebih kecil	Sistem menampilkan wayang yang dipilih secara lebih kecil	Berhasil
11	Klik <i>button</i> "Rotate"	Sistem akan menampilkan wayang yang dipilih dengan berputar	Sistem menampilkan wayang yang dipilih dengan berputar	Berhasil

No	Aksi	Hasil yang diharapkan	Output	Keterangan
12	Klik <i>button</i> "Voice"	Sistem akan mengeluarkan suara wayang sedang dipilih	Sistem mengeluarkan suara wayang sedang dipilih	Berhasil
13	Klik <i>button</i> "Home"	Sistem akan keluar dari halaman <i>Start AR</i> dan kembali ke halaman Menu Utama	Sistem keluar dari halaman <i>Start AR</i> dan kembali ke halaman Menu Utama	Berhasil

#### 4.3.4 Hasil Pengujian terhadap Lingkungan

Tabel 4. 4 Hasil Pengujian terhadap Lingkungan

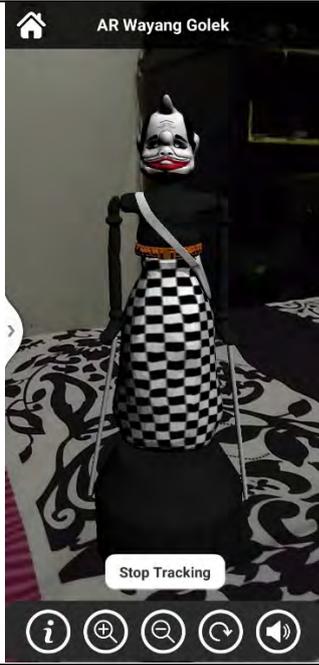
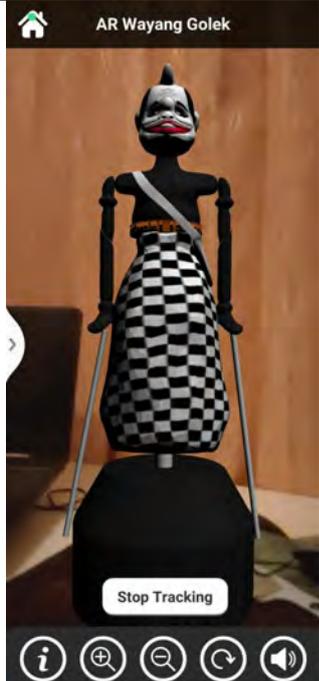
No	Lingkungan	Hasil
1	Lingkungan dengan lampu 4W	
2	Lingkungan dengan lampu 5W	

3	Lingkungan dengan lampu 8W	
---	----------------------------	--

#### 4.3.5 Hasil Pengujian dengan Tipe Smartphone yang Berbeda

Tabel 4. 5 Hasil Pengujian pada Smartphone

No	Tipe Smartphone	Hasil
1	<p>Oppo A7</p> <p>Dengan spesifikasi smartphone:</p> <p>Sistem Operasi : Android 8.1, ColorOS 5.2</p> <p>ROM/ Memori Internal : 64GB 4GB RAM</p> <p>Kamera : 13MP, LED flash, HDR</p> <p>CPU/ Prosesor : Octa-core 1.8 GHz Cortex-A53</p>	

2	<p>Oppo A5 2020</p> <p>Dengan spesifikasi smartphone:</p> <p>Sistem Operasi : Android 9.0, ColorOS 6.1</p> <p>ROM/ Memori Internal : 64GB 4GB RAM</p> <p>Kamera : 12MP, LED flash, HDR</p> <p>CPU/ Prosesor : Octa-core (4x2.0 GHz Kryo 260 Gold &amp; 4x1.8 GHz Kryo 260 Silver)</p>	
3	<p>Poco x3 pro</p> <p>Dengan spesifikasi smartphone:</p> <p>Sistem Operasi : MIUI 12 berbasis POCO</p> <p>Penyimpanan &amp; RAM : 6+128 GB</p> <p>Kamera : 48MP</p> <p>CPU/ Prosesor : Qualcomm Kryo 485, CPU octa-core, hingga 2,96GHz</p>	

Pengujian fungsi dengan metode *black box* yang sudah dilakukan penulis dapat dilihat pada tabel-tabel diatas, dimana semua tombol-tombol yang penulis sudah uji semuanya berjalan dengan sebagaimana yang diharapkan. Pembuatan aplikasi "Memperkenalkan Kesenian Wayang Golek Jawa Barat dengan *Augmented Reality* secara *Markerless* Berbasis Android" dapat di lakukan dan berjalan baik dengan menggunakan *software* Unity dan akses kamera pada smartphone berjalan baik.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah melakukan perancangan perangkat lunak dilakukan dalam penelitian ini, dan hasil pengujian, dapat diambil beberapa kesimpulan sebagai berikut:

1. Dalam meningkatkan pelestarian dan meningkatkan minat masyarakat dan kaum muda mengenai kesenian Wayang Golek yaitu dengan cara pembuatan aplikasi “Memperkenalkan Kesenian Wayang Golek Jawa Barat dengan *Augmented Reality* secara *Markerless* Berbasis Android” dengan bentuk 3D *Augmented Reality* ini membuat masyarakat dan kaum muda lebih tertarik dalam mempelajari dan lebih mengenal kesenian Wayang Golek di Jawa Barat.
2. Karena pengenalan wayang golek yang masih belum menggunakan aplikasi, sehingga pengunjung-pengunjung dari luar atau dalam terkadang mengalami kesulitan dalam mengenal dan melihat bentuk wayang golek, menempuh jarak yang di tuju terlalu jauh sehingga mereka harus ke museum atau tempat-tempat pengrajin wayang. Maka dibuatlah pengimplementasian *Augmented reality* yang berbasis Android ini sebagai media pengenalan dan pembelajaran yang bersifat *portable*, memberikan kemudahan dalam penggunaannya serta dapat digunakan dimana saja. Dengan kemudahan itu, maka aplikasi ini dibuat. Adapun keuntungan-keuntungan yang ada seperti pengguna dapat mengontrol objek 3D yang ditampilkan, dapat mengeluarkan suara, dapat melihat penjelasan-penjelasan dari wayang yang dipilih dan user dapat mengontrol objek 3D dengan *zoom in* , *zoom out*, dan *rotate*. Aplikasi diimplementasikan menggunakan metode *markerless* AR yaitu SLAM (*Simultaneous Location and Mapping*) yang berarti *user* atau pengguna dapat mengakses dan menggunakan aplikasi dimana saja dan kapan saja.
3. Karena terkadang dari internet hanya menampilkan objek berbentuk 2D yang kurang akan visualisasi realitasnya. Maka dibuatlah aplikasi yang dapat menampilkan Wayang Golek Jawa Barat dengan bentuk 3D yang dibuat dengan aplikasi blender dan dengan *software* Unity3D karena Unity merupakan *game engine multi platform*

yang salah satunya untuk pembuatan sebuah aplikasi berbasis Android serta untuk pembuatan UI serta pengembangan fungsionalitas untuk komponen-komponen.

## 5.2 Saran

Perancangan aplikasi “Memperkenalkan Kesenian Wayang Golek Jawa Barat dengan *Augmented Reality* secara *Markerless* Berbasis Android” ini masih banyak memiliki kekurangan, oleh karena itu penulis dapat menyampaikan saran-saran untuk tugas akhir ini, agar aplikasi dapat dikembangkan menjadi lebih baik dengan penambahan sebagai berikut:

1. Menambah objek-objek 3d Wayang Golek Jawa Barat lainnya seperti Arjuna, Gatotkaca, Gareng, Dawala, dan lain sebagainya.
2. Membuat wayang golek 3d yang sedang di tracking lebih fokus, lebih konsisten berdiri ditempat yang ditracking dan lebih bisa menyesuaikan ukuran objek pada saat diletakan di permukaan.
3. Menambah efek dan variasi animasi pada objek-objek 3d wayang golek AR agar lebih menarik.
4. Software dan hardware yang dipakai dan digunakan kedepannya lebih baik.

## DAFTAR PUSTAKA

- Arief, U. M., Wibawanto, H., & Nastiti, A. L. (2019). Membuat Game Augmented Reality (AR) dengan Unity 3D. *Yogyakarta: Andi*.
- Arnaldi, B., Guitton, P., & Moreau, G. (2018). *Virtual reality and augmented reality: Myths and realities*. John Wiley & Sons.
- Artra, W. (2012). Augmented Reality Objek 3 Dimensi dengan Perangkat Artoolkit dan Blender. *Jurnal Teknologi Informasi DINAMIK Volume, 17(2)*, 107–117.
- Cahaya, C. (n.d.). Nilai, Makna, dan Simbol dalam Pertunjukan Wayang Golek sebagai Representasi Media Pendidikan Budi Pekerti. *Panggung, 26(2)*, 298246.
- Callaham, J. (2018). The history of Android OS: its name, origin and more. *Android Authority*.
- Cha, S., Taylor, R. N., & Kang, K. (2019). *Handbook of software engineering*. Springer.
- Clark, D., & Sanders, J. (2011). *Beginning C# object-oriented programming*. Springer.
- Craig, A. B. (2013). *Understanding augmented reality: Concepts and applications*. Newnes.
- Cushman, D., & Habbak, H. E. L. (2013). *Developing ar games for ios and android*. Packt Publishing Ltd.
- Dennis, A., Wixom, B., & Tegarden, D. (2015). *Systems analysis and design: An object-oriented approach with UML*. John wiley & sons.
- Dhari, Y. W. (2019). Pewarisan Keahlian Mendalang pada Keluarga Dalang Wayang Golek Abah Sunarya. *Umbara, 4(2)*, 130–140.
- Fernando, M. (2013). Membuat aplikasi android augmented reality menggunakan vuforia sdk dan unity.
- Foster, E. C. (2014). *Software Engineering: A Methodical Approach*. Kindle Edition. A Press.
- Furht, B. (2011). *Handbook of augmented reality*. Springer Science & Business Media.
- Grubert, J., & Grasset, R. (2013). *Augmented reality for Android application development*. Packt Publishing Ltd.
- Hasanah, F. N., & Untari, R. S. (2020). Buku Ajar Rekayasa Perangkat Lunak. *Umsida*

Press, 1–119.

- Hillar, G. C. (2015). *Learning Object-oriented programming*. Packt Publishing Ltd.
- Iswari, N. M. S. (2015). Review Perangkat Lunak StarUML Berdasarkan Faktor Kualitas McCall. *Ultimatics: Jurnal Teknik Informatika*, 7(1).
- Lee, W.-M. (2012). *Beginning android 4 application Development*. John Wiley & Sons.
- Mall, R. (2018). *Fundamentals of software engineering*. PHI Learning Pvt. Ltd.
- Marsic, I. (2012). *Software Engineering*. EBook.
- Mealy, P. (2018). *Virtual & augmented reality for dummies*. John Wiley & Sons.
- Meier, R., & Lake, I. (2018). *Professional Android*. John Wiley & Sons.
- Nugroho, A. (2010). *Rekayasa perangkat lunak berorientasi objek dengan metode USDP*. Penerbit Andi.
- Peddie, J. (2017). *Augmented reality: Where we will all live*. Springer.
- Rumpe, B. (2016). *Modeling with UML*. Springer.
- Salahuddin, M., & Rosa, A. S. (2011). *Rekayasa Perangkat Lunak. Bandung: Modula*.
- Setiyani, L. (2019). *Research Methods Information Technology*.
- Stephens, R. (2015). *Beginning software engineering*. John Wiley & Sons.
- Yunanto, R. (2015). *Buku Uml*.

<https://www.uml-diagrams.org/uml-25-diagrams.html> diakses tanggal 30 November 2021

Pukul 21:20 WIB

<http://unity3d.com/> diakses tanggal 30 November 2021 Pukul 23:30 WIB

[www.blender.org](http://www.blender.org) diakses tanggal 30 November 2021 Pukul 22:40 WIB

**LAMPIRAN 1**  
**CONTOH TAMPILAN 3D WAYANG GOLEK**

1. Lampiran objek 3d Wayang Golek Semar :

-Tampak depan



-Tampak samping



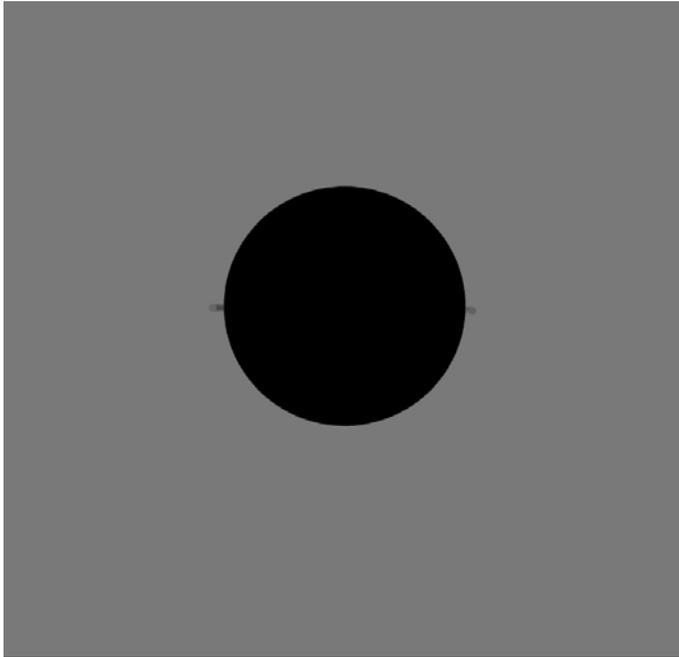
-Tampak belakang



-Tampak atas



-Tampak bawah



2. Lampiran Skesta 3d Wayang Golek Cepot :

-Tampak depan



-Tampak samping kiri



-Tampak samping kanan



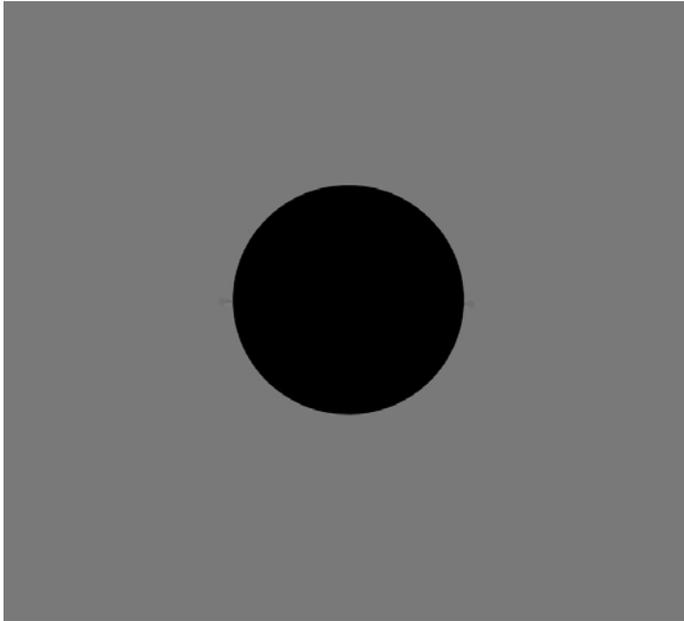
-Tampak belakang



-Tampak atas



-Tampak bawah



3. Lampiran Skesta 3d Wayang Golek Hanoman :

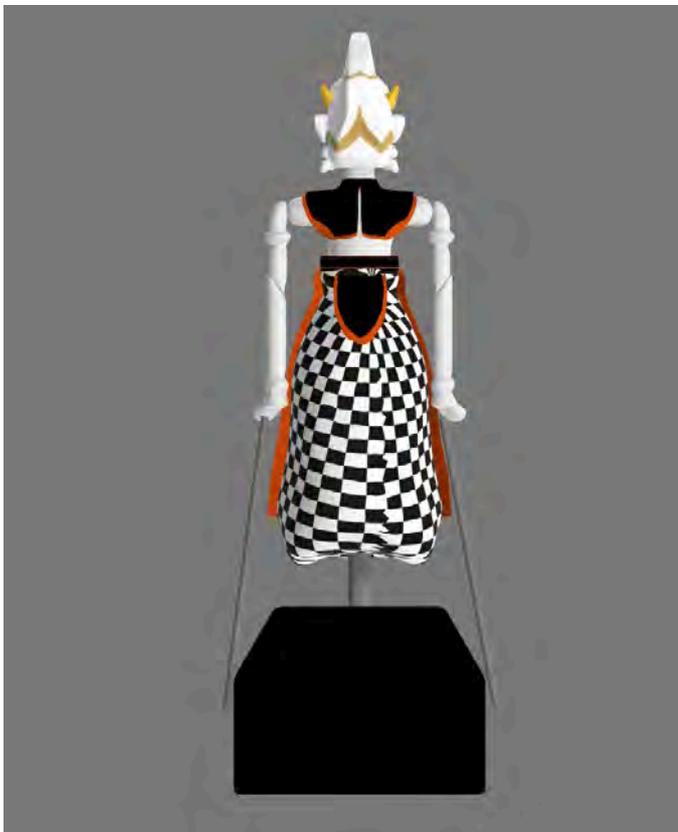
-Tampak depan



- Tampak samping



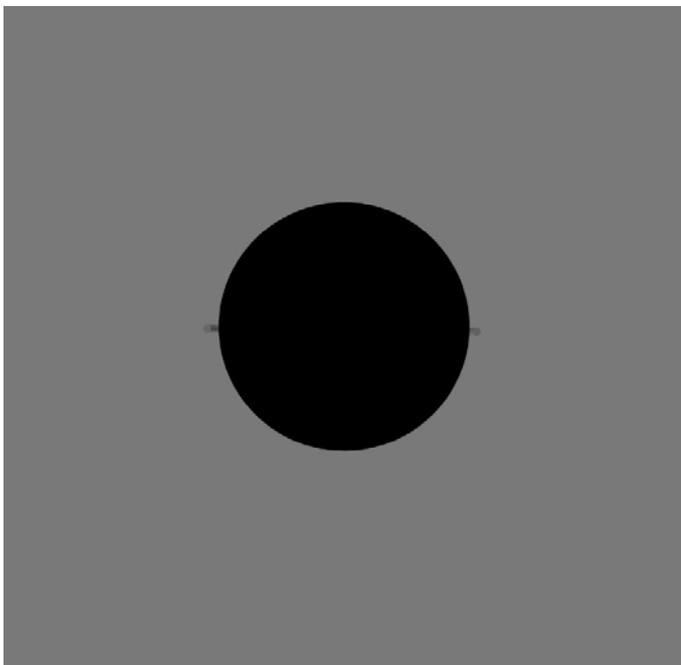
- Tampak belakang



- Tampak atas



- Tampak bawah



## LAMPIRAN 2

### LISTING PROGRAM

#### 1. *AppManager.cs*

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
public class AppManager : MonoBehaviour
{
    public GameObject MainPanel, HelpPanel;

    // Start is called before the first frame update
    void Start()
    {
        MainPanel.SetActive(true);
        HelpPanel.SetActive(false);
    }

    public void ArMenuScene()
    {
        SceneManager.LoadScene("ARscene");
    }

    public void MainPanelActive()
    {
        MainPanel.SetActive(true);
        HelpPanel.SetActive(false);
    }

    public void HelpPanelActive()
    {
        MainPanel.SetActive(false);
        HelpPanel.SetActive(true);
    }

    public void QuitApps()
    {
        Application.Quit();
    }
}
```

#### 2. *ArManager.cs*

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ArManager : MonoBehaviour
{
    public GameObject[] Wayang;
    public GameObject[] BtnProfil;
    // Start is called before the first frame update
    void Start()
    {
        Wayang[0].SetActive(true);
        Wayang[1].SetActive(false);
        Wayang[2].SetActive(false);
    }
}
```

```

        BtnProfil[0].SetActive(true);
        BtnProfil[1].SetActive(false);
        BtnProfil[2].SetActive(false);
    }

    public void WayangSemarPicked()
    {
        Wayang[0].SetActive(true);
        Wayang[1].SetActive(false);
        Wayang[2].SetActive(false);
        BtnProfil[0].SetActive(true);
        BtnProfil[1].SetActive(false);
        BtnProfil[2].SetActive(false);
    }

    public void WayangCepotPicked()
    {
        Wayang[0].SetActive(false);
        Wayang[1].SetActive(true);
        Wayang[2].SetActive(false);
        BtnProfil[0].SetActive(false);
        BtnProfil[1].SetActive(true);
        BtnProfil[2].SetActive(false);
    }

    public void WayangHanomanPicked()
    {
        Wayang[0].SetActive(false);
        Wayang[1].SetActive(false);
        Wayang[2].SetActive(true);
        BtnProfil[0].SetActive(false);
        BtnProfil[1].SetActive(false);
        BtnProfil[2].SetActive(true);
    }
}

```

### 3. *Backpress.cs*

```

using UnityEngine;
using System.Collections;
using UnityEngine.SceneManagement;

public class backpress : MonoBehaviour {

    public string MainMenu;

    // Update is called once per frame

    void Update()
    {
        if (Input.GetKeyDown(KeyCode.Escape))
        {
            SceneManager.LoadScene(MainMenu);
        }
    }

    public void BackToMenu()
    {

```

```

        SceneManager.LoadScene(MainMenu);
    }

    public void QuitApps()
    {
        Application.Quit();
    }
}

```

#### 4. *FadeEffectScene.cs*

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FadeEffectScene : MonoBehaviour {

    public Texture2D fadeOutTexture;
    public float fadeSpeed = 0.8f;

    private int drawDepth = -1000;
    private float alpha = 1.0f;
    private int fadeDir = -1;

    void OnGUI()
    {
        alpha += fadeDir * fadeSpeed * Time.deltaTime;
        alpha = Mathf.Clamp01(alpha);

        GUI.color = new Color(GUI.color.r, GUI.color.g, GUI.color.b,
alpha);
        GUI.depth = drawDepth;
        GUI.DrawTexture(new Rect(0, 0, Screen.width, Screen.height),
fadeOutTexture);
    }

    public float BeginFade(int direction)
    {
        fadeDir = direction;
        return (fadeSpeed);
    }

    void OnLevelWasLoaded()
    {
        BeginFade(-1);
    }
}

```

#### 5. *RotateObject.cs*

```

using UnityEngine;
using System.Collections;

public class RotateObject : MonoBehaviour
{

```

```

public GameObject objectRotate;

public float rotateSpeed = 50f;
bool rotateStatus = false;

public void Rotasi()
{
    if (rotateStatus == false)
    {
        rotateStatus = true;
    }
    else
    {
        rotateStatus = false;
    }
}

void Update()
{
    if (rotateStatus == true)
    {
        objectRotate.transform.Rotate(Vector3.up, rotateSpeed *
Time.deltaTime);
    }
}
}

```

## 6. ShowHideInfo.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ShowHideInfo : MonoBehaviour {

    public GameObject InfoText;
    private bool ShowInfo = false;

    public void showhideInfo()
    {
        if (!ShowInfo)
        {
            InfoText.SetActive(true);
            ShowInfo = true;
        }
        else
        {
            InfoText.SetActive(false);
            ShowInfo = false;
        }
    }

    public void ShowPanelInfo()
    {
        InfoText.SetActive(true);
    }
}

```

### 7. *ShowHideObject.cs*

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ShowHideObject : MonoBehaviour {

    public GameObject Object;
    private bool showObject = false;

    public void showhideObject()
    {
        if (!showObject)
        {
            Object.SetActive(false);
            showObject = true;
        }
        else
        {
            Object.SetActive(true);
            showObject = false;
        }
    }
}
```

### 8. *SliderMenuAnim.cs*

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class SliderMenuAnim : MonoBehaviour
{
    public GameObject PanelMenu;

    public void ShowHideMenu()
    {
        if(PanelMenu != null)
        {
            Animator animator = PanelMenu.GetComponent<Animator>();
            if(animator != null)
            {
                bool isOpen = animator.GetBool("show");
                animator.SetBool("show", !isOpen);
            }
        }
    }
}
```

### 9. *SplashFade.cs*

```
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;
```

```

using System.Collections;

public class SplashFade : MonoBehaviour
{
    public Image splashImage;
    public string loadLevel;

    IEnumerator Start()
    {
        splashImage.canvasRenderer.SetAlpha(0.0f);

        FadeIn();
        yield return new WaitForSeconds(3.5f);
        FadeOut();
        yield return new WaitForSeconds(3.5f);
        SceneManager.LoadScene(loadLevel);
    }

    void FadeIn()
    {
        splashImage.CrossFadeAlpha(1.0f, 1.5f, false);
    }

    void FadeOut()
    {
        splashImage.CrossFadeAlpha(0.0f, 2.5f, false);
    }
}

```

#### 10. ZoomObj.cs

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ZoomObj : MonoBehaviour
{
    public GameObject Object;

    private bool _ZoomIn;
    private bool _ZoomOut;

    //object scale speed
    public float Scale = 0.1f;

    // Update is called once per frame
    void Update()
    {
        if (_ZoomIn)
        {
            //make a bigger object
            Object.transform.localScale += new Vector3(Scale, Scale,
Scale);
        }

        if (_ZoomOut)
        {

```

```

        //make a small object
        Object.transform.localScale -= new Vector3(Scale, Scale,
Scale);
    }
}

//Make object scaled big
public void OnPressZoomIn()
{
    _ZoomIn = true;
}

public void OnReleaseZoomIn()
{
    _ZoomIn = false;
}

//Make object scaled small
public void OnPressZoomOut()
{
    _ZoomOut = true;
}

public void OnReleaseZoomOut()
{
    _ZoomOut = false;
}
}

```